

IMPERIAL COLLEGE LONDON

DEPARTMENT OF COMPUTING

Quantum Computing
and the Hidden Subgroup Problem

Author

Edward PEARCE-CRUMP

Supervisor

Professor William KNOTTENBELT

Submitted in partial fulfillment of the requirements for the MSc degree in
Computing Science of Imperial College London

September 2020

ad matrem amandam meam et patrem

Abstract

We look at how the Quantum Fourier Transform can be used to solve a number of computationally difficult problems on a quantum computer. In the first half of this thesis, we show how a model for Quantum Computing can be constructed from the postulates of Quantum Mechanics, focusing on the unitary nature of operations in a quantum computer. We describe a procedure that replicates any classical computation on a quantum computer using only Toffoli and quantum NOT gates. In addition, we study the important quantum algorithms for the quantum system \mathbb{Z}_{2^n} : in particular, we look at a procedure that solves the Phase Estimation Problem for a unitary operator and see how it is implemented in Shor's Algorithm to factor integers into their prime decomposition. In the second half, we generalise our approach to quantum systems that are the group algebra of a finite group over \mathbb{C} . We look at whether there exists an efficient quantum algorithm that solves the Hidden Subgroup Problem for an arbitrary group. We show how the irreducible representations of a group can be used in a procedure to solve the problem in the case where the group is abelian. We see how Shor's Algorithm and the Discrete Logarithm Problem are specific instances of this procedure. Finally, we investigate the Hidden Subgroup Problem in the case where the group is not abelian, which is an open problem in Computer Science. We show how weak Fourier sampling makes it possible to find the hidden subgroup in the case where it is normal in the group.

This is my own work unless otherwise stated.

Acknowledgements

To Professor William Knottenbelt, thank you for supervising me during this Master's Thesis. In particular, I am grateful that you allowed me to switch my research topic to Quantum Computing. I look forward to working under your supervision again in the near future.

To Ellie, thank you for your encouragement, kindness and enthusiasm during the last few months. The quantum circuits are dedicated to you.

To Andrew and Alexandra, thank you for your support over the years. You are the greatest gifts my parents could ever give me.

To Mum and Dad, thank you for all the love that you've given me and for the many sacrifices that you've made for me. I wouldn't have been able to write this thesis without you. I hope I've done you proud.

Notation

Linear Algebra

Symbol	Description
--------	-------------

\dagger	Adjoint
$*$	Complex Conjugate
\otimes	Tensor Product
\oplus	Direct Sum; sometimes addition modulo 2
tr	Trace operator
rank	Rank of a Matrix
$(*, *)$	Inner Product
$\langle * * \rangle$	Inner Product; a “braket”
$(*, *)_H$	Inner Product restricted to a group H
$(A^*)^T$	Adjoint of the matrix A
δ_{ij}	Kronecker Delta; 1 if $i = j$, 0 otherwise
CNOT	Controlled-NOT gate
H	Hadamard operator on a qubit
$H^{\otimes n}$	Hadamard operator on n qubits
F_{2^n}	Quantum Fourier Transform for a quantum system of size 2^n
F_G	Quantum Fourier Transform for an arbitrary group G
F_G^\dagger	Inverse Quantum Fourier Transform for an arbitrary group G
I	Identity operator
I_V	Identity operator on a vector space V
I_d	Identity matrix of dimension d
M_i	Measurement Operator indexed by outcome i
Q	Quantum System, a complex Hilbert Space
R_k	Rotation gate
S	Phase gate; same as R_2 gate
Toffoli	Toffoli gate
U	Arbitrary Unitary operator
U_f	Oracle evaluating the function f
X	Quantum NOT gate
$ \psi\rangle$	State of a quantum computer
ρ	Density Operator/Density Matrix
$ v\rangle$	A vector in Dirac Notation; a “ket”
$\langle v $	Defined to be $ v\rangle^\dagger$; a “bra”
$ w\rangle \langle v $	Outer Product Operator
$ v\rangle \otimes w\rangle$	Tensor Product of $ v\rangle$ and $ w\rangle$
$ v\rangle w\rangle$	Abbreviated notation for the Tensor Product of $ v\rangle$ and $ w\rangle$
$ vw\rangle$	Abbreviated notation for the Tensor Product of $ v\rangle$ and $ w\rangle$
$\{ 0\rangle, 1\rangle\}$	Computational Basis of a qubit
$\{ 0\rangle, 1\rangle\}^n$	Computational Basis of an n -qubit quantum system
$ 0\rangle^{\otimes n}$	Initial State of an n -qubit quantum computer
$ \tilde{x}\rangle$	Element of the Fourier basis
$ x + H\rangle$	Coset State in an abelian group G
$ xH\rangle$	Coset State in any group G
ϕ	Phase of an eigenvalue of a unitary operator
$x \cdot y$	Dot Product

Group Theory

Symbol	Description
G	Symbol for an arbitrary group
$ G $	Order of the group G
1	Identity element of an arbitrary or non-abelian group
0	Identity element of an abelian group
(C_N, \times)	Cyclic group of order N , multiplicative notation
$(\mathbb{Z}_N, +)$	Cyclic group of order N , additive notation
(\mathbb{Z}_N^*, \times)	Multiplicative group of integers modulo N
$(\mathbb{C}^\times, \times)$	Group of units of \mathbb{C}
$\mathbb{C}[G]$	Group Algebra of G over \mathbb{C}
\mathbb{C}^G	Set of functions $G \rightarrow \mathbb{C}$
\widehat{G}	Dual group of G
G/H	Quotient Group
$[G : H]$	Index of H in G ; size of the Quotient group G/H
$GL(V)$	Group of invertible linear maps $V \rightarrow V$
H	Hidden Subgroup in the Hidden Subgroup Problem
H^\perp	Orthogonal subgroup of \widehat{G}
L	Left Regular Representation
R	Right Regular Representation
$ord(x)$	Order of the element $x \in G$
ρ	Representation ρ of a group
d_ρ	Dimension of a representation ρ
χ_ρ	Character of a representation ρ
χ	Abbreviated notation for χ_ρ
$ker(\phi)$	Kernel of a group homomorphism ϕ
$im(\phi)$	Image of a group homomorphism ϕ
$\mathbb{1}_G$	Trivial representation of a group G
σ	Irreducible representation of a group
\leq	Subgroup
\trianglelefteq	Normal Subgroup
\cong	Isomorphic groups
\sim	Isomorphic representations
Ψ_x	Pontryagin Duality, evaluate character at $x \in G$
ω_N	N^{th} root of unity, defined to be $e^{\frac{2\pi i}{N}}$
ω	Abbreviated notation for ω_N

Miscellaneous

Symbol	Description
\wedge	Logical AND
\neg	Logical NOT
\sum	Sum
\prod	Product
\int	Integral
gcd	Greatest Common Divisor
λ	Scalar in \mathbb{C}
\log_2	Base-2 Logarithm

$\phi(N)$	Euler's Totient Function
$[a_0, \dots, a_n]$	n^{th} convergent of a simple continued fraction
$L^2([0, 1])$	Space of L^2 functions on $[0, 1]$
$L^2(\mathbb{Z})$	Space of L^2 functions on \mathbb{Z}
\hat{f}	Fourier Transform of a function f
$P(i)$	Probability of outcome i occurring
$\Omega(N)$	Asymptotic Lower Bound for an algorithm of input size N
$O(N)$	Asymptotic Upper Bound for an algorithm of input size N
$\Theta(N)$	Asymptotic Lower and Upper Bound for an algorithm of input size N
\mathbb{S}^1	Unit Circle

Contents

1	Introduction	1
2	The Quantum Computing Model	7
2.1	Quantum Systems and their States	7
2.2	Transformations of Quantum States	8
2.3	Measurement of Quantum States	19
2.4	Quantum Algorithms	21
2.5	Density Operators	22
2.6	Reversible Operations	24
3	Applications of the Quantum Fourier Transform	29
3.1	Quantum Fourier Transform, version 1	29
3.2	Phase Estimation	34
3.3	Order-Finding Algorithm	41
3.4	Factoring Integers	46
4	Quantum Computing with Groups	50
4.1	The Hidden Subgroup Problem	50
4.2	Group Representation Theory	50
4.3	The Characters of Finite Abelian Groups	57
5	The Quantum Fourier Transform for Abelian Groups	65
5.1	Quantum Fourier Transform, version 2	65
5.2	Quantum Fourier Transform over a General Finite Abelian Group	68
6	The Abelian Hidden Subgroup Problem	70
6.1	A Procedure for the Abelian Hidden Subgroup Problem	70
6.2	The Discrete Logarithm Problem	73
7	The General Hidden Subgroup Problem	76
7.1	Quantum Fourier Transform, version 3	76
7.2	Weak Fourier Sampling	82
7.3	Normal Subgroups	86
8	Conclusion	89

Appendix A Literature Overview	91
Appendix B Research Approach	92
References	93

1 Introduction

Joseph Fourier (1768 – 1830), in his attempts to solve the heat equation [22]

$$\frac{\partial u}{\partial t} = \alpha^2 \frac{\partial^2 u}{\partial x^2} \quad (1.1)$$

for some function $u(x, t)$ representing the temperature of a metal rod at position x and time t , realised that certain complex-valued functions f could be represented as a linear combination of sine and cosine functions.

In particular, if f is a 1-periodic function mapping the real line \mathbb{R} to the complex plane \mathbb{C} , then f can be expressed in the form

$$f(x) = \sum_{n \geq 0} a_n \cos(2\pi n x) + \sum_{n \geq 1} b_n \sin(2\pi n x) \quad (1.2)$$

with coefficients $a_n, b_n \in \mathbb{C}$ for all $n \in \mathbb{Z}$.

It is possible to rewrite Equation (1.2) using complex exponentials instead as

$$f(x) = \sum_{n=-\infty}^{\infty} c_n e^{2\pi i n x} \quad (1.3)$$

with coefficients $c_n \in \mathbb{C}$.

It can be shown that the coefficients c_n take the form

$$c_n = \int_0^1 e^{-2\pi i n x} f(x) dx \quad (1.4)$$

Relabelling c_n as $\hat{f}(n)$, Equation (1.3) becomes

$$f(x) = \sum_{n=-\infty}^{\infty} \hat{f}(n) e^{2\pi i n x} \quad (1.5)$$

Expressions of this form are called Fourier series and the coefficients $\hat{f}(n)$ are called Fourier coefficients.

If we consider instead functions f in the vector space $L^2([0, 1])$, which are complex-valued functions defined on the interval $[0, 1]$ satisfying

$$\|f\|^2 := \int_0^1 |f(x)|^2 dx < \infty \quad (1.6)$$

then not only can it be shown that the integral

$$\hat{f}(n) := \int_0^1 e^{-2\pi i n x} f(x) dx \quad (1.7)$$

exists for all $n \in \mathbb{Z}$, but also that the limit

$$\lim_{N \rightarrow \infty} \left\| \sum_{n=-N}^N \hat{f}(n) e^{2\pi i n x} - f(x) \right\| \quad (1.8)$$

exists and equals 0, where $\| * \|$ is defined in Equation (1.6).

Hence we say that $f(x)$ has a Fourier series which we write in the form

$$\sum_{n=-\infty}^{\infty} \widehat{f}(n) e^{2\pi i n x} \quad (1.9)$$

The vector space $L^2([0, 1])$ was of particular interest to Fourier because its functions come with a lot more structure than simple periodic functions.

In particular, we can define a function $(*, *) : L^2([0, 1]) \times L^2([0, 1]) \rightarrow \mathbb{C}$ to be

$$(f, g) := \int_0^1 f(x)^* g(x) dx \quad (1.10)$$

for all $f, g \in L^2([0, 1])$, where $*$ denotes the complex conjugate. This function can be seen to satisfy the properties of a complex inner product as given in Definition 2.1.

As a result, we have that

$$(f, f) = \int_0^1 f(x)^* f(x) dx = \int_0^1 |f(x)|^2 dx = \|f\|^2 \quad (1.11)$$

Furthermore, we can show that the set of functions $B := \{e_n\}_{n \in \mathbb{Z}}$ where

$$e_n(x) := e^{2\pi i n x} \quad \text{for all } x \in [0, 1] \quad (1.12)$$

is an orthonormal set in $L^2([0, 1])$ with respect to the inner product $(*, *)$ defined in Equation (1.10).

Proof. Consider the inner product (e_m, e_n) :

$$(e_m, e_n) = \int_0^1 (e^{2\pi i m x})^* e^{2\pi i n x} dx = \int_0^1 e^{-2\pi i m x} e^{2\pi i n x} dx = \int_0^1 e^{2\pi i (n-m)x} dx \quad (1.13)$$

Splitting into cases:

Case 1: $m \neq n$

Equation (1.13) becomes

$$\frac{1}{2\pi i (n-m)} e^{2\pi i (n-m)x} \Big|_0^1 = \frac{1}{2\pi i (n-m)} [e^{2\pi i (n-m)} - e^0] = \frac{1}{2\pi i (n-m)} [1 - 1] = 0 \quad (1.14)$$

Case 2: $m = n$

Equation (1.13) becomes

$$\int_0^1 1 dx = 1 \quad (1.15)$$

Hence

$$(e_m, e_n) = \delta_{mn} \quad (1.16)$$

where δ is the Kronecker delta. □

With more work, it can, in fact, be shown that the orthonormal set B of complex exponentials is an orthonormal basis of $L^2([0, 1])$ with respect to $(*, *)$.

As a result, it is easy to see that

$$(e_n, f) = \int_0^1 e_n(x)^* f(x) dx = \int_0^1 e^{-2\pi i n x} f(x) dx = \widehat{f}(n) \quad (1.17)$$

and so, from Equation (1.9), functions $f \in L^2([0, 1])$ can be written as a linear combination of the orthonormal basis elements of $L^2([0, 1])$

$$f = \sum_{n=-\infty}^{\infty} (e_n, f) e_n \quad (1.18)$$

This allows us to prove the following identity, which is often referred to as Parseval's Identity, although sometimes it is also called Rayleigh's Identity:

$$\int_0^1 |f(x)|^2 dx = \sum_{n=-\infty}^{\infty} |\widehat{f}(n)|^2 \quad (1.19)$$

Proof. We have that

$$\begin{aligned} \int_0^1 |f(x)|^2 dx &= (f, f) \\ &= \left(\sum_{m=-\infty}^{\infty} (e_m, f) e_m, \sum_{n=-\infty}^{\infty} (e_n, f) e_n \right) \\ &= \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} (e_m, f)^* (e_n, f) (e_m, e_n) \\ &= \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} (e_m, f)^* (e_n, f) \delta_{mn} \\ &= \sum_{n=-\infty}^{\infty} (e_n, f)^* (e_n, f) \\ &= \sum_{n=-\infty}^{\infty} |(e_n, f)|^2 \\ &= \sum_{n=-\infty}^{\infty} |\widehat{f}(n)|^2 \end{aligned}$$

as required. □

If we consider the vector space of functions $L^2(\mathbb{Z})$ which are complex-valued functions F defined on \mathbb{Z} satisfying

$$\sum_{n=-\infty}^{\infty} |F(n)|^2 \leq \infty \quad (1.20)$$

and we define a function $\langle * | * \rangle : L^2(\mathbb{Z}) \times L^2(\mathbb{Z}) \rightarrow \mathbb{C}$ to be

$$\langle F | G \rangle := \sum_{n=-\infty}^{\infty} F(n)^* G(n) \quad (1.21)$$

for all $F, G \in L^2(\mathbb{Z})$, then it can be shown using Definition 2.1 that $\langle *|* \rangle$ is an inner product for $L^2(\mathbb{Z})$. Consequently, we can rewrite Parseval's Identity in terms of inner products:

$$(f, f) = \langle \widehat{f} | \widehat{f} \rangle \quad (1.22)$$

In fact, we claim that Parseval's Identity actually says that the operator $U : L^2([0, 1]) \rightarrow L^2(\mathbb{Z})$, which maps $f \mapsto \widehat{f}$, is a unitary operator, that is, it is surjective and it preserves the inner product.

We call U the Fourier Transform on $L^2([0, 1])$.

Proof. We know from Equation (1.16) that $(e_m, e_n) = \delta_{mn}$.

Writing $\widehat{f}_i := Ue_i$ for $i = m, n$, we consider $\langle \widehat{f}_m | \widehat{f}_n \rangle$.

From Equation (1.9), setting $f(x) = e_n(x)$, we can see that

$$\widehat{f}_n(k) = \delta_{kn} \quad (1.23)$$

for all $k \in \mathbb{Z}$.

Similarly, we can see that $\widehat{f}_m(k) = \delta_{km}$ for all $k \in \mathbb{Z}$.

Therefore

$$\langle \widehat{f}_m | \widehat{f}_n \rangle = \sum_{k=-\infty}^{\infty} \widehat{f}_m(k)^* \widehat{f}_n(k) = \sum_{k=-\infty}^{\infty} \delta_{km}^* \delta_{kn} = \delta_{mn} = (e_m, e_n) \quad (1.24)$$

Since $\{e_n\}_{n \in \mathbb{Z}}$ forms an orthonormal basis of $L^2([0, 1])$, we have that $\{\widehat{f}_n\}_{n \in \mathbb{Z}}$ forms an orthonormal basis of $L^2(\mathbb{Z})$, from which we can easily see that U is surjective; moreover, we have that

$$(f, g) = \langle Uf | Ug \rangle \quad (1.25)$$

for all f, g in $L^2([0, 1])$, which can easily be seen by expressing f, g, Uf, Ug as linear combinations of the basis elements of their respective vector spaces and applying the properties of the inner product.

We have therefore shown that the Fourier Transform on $L^2([0, 1])$ is unitary. \square

This result made it possible for Fourier to solve the heat equation given in Equation (1.1) when $u(x, t)$ was a function of position and time such that $u(x, t) \in L^2([0, 1]) \times \mathbb{R}_{\geq 0}$ satisfying $u(0, t) = u(1, t)$, for he could then express $u(x, t)$ as a Fourier series

$$u(x, t) = \sum_{n=-\infty}^{\infty} c_n(t) e^{2\pi i n x} \quad (1.26)$$

where

$$c_n(t) = \int_0^1 e^{-2\pi i n x} u(x, t) dx \quad (1.27)$$

By differentiating this expression with respect to t , substituting in Equation (1.1), and using integration by parts twice, he would have obtained a differential equation in $c_n(t)$ whose solution is

$$c_n(t) = c_n(0) e^{-4\alpha^2 \pi^2 n^2 t} \quad (1.28)$$

By assuming some additional boundary conditions, namely that

$$u(x, 0) = f(x) \quad (1.29)$$

where $f \in L^2([0, 1])$, he would have found that $c_n(0)$ is

$$c_n(0) = \int_0^1 e^{-2\pi i n x} u(x, 0) dx = \int_0^1 e^{-2\pi i n x} f(x) dx = \widehat{f}(n) \quad (1.30)$$

the n^{th} Fourier coefficient of f .

Hence he would have shown that the general solution to the heat equation under these conditions is

$$u(x, t) = \sum_{n=-\infty}^{\infty} \widehat{f}(n) e^{-4\alpha^2 \pi^2 n^2 t} e^{2\pi i n x} \quad (1.31)$$

Now while this thesis is neither a study of the heat equation, nor a study of the Fourier Transform on $L^2([0, 1])$, it is a study of a different type of Fourier Transform known as the Quantum Fourier Transform, which has many of the same properties that the Fourier Transform on $L^2([0, 1])$ has, and which often forms the key to solving many difficult problems in Quantum Computing.

In particular, we will see that the Quantum Fourier Transform is also a unitary operator on the complex Hilbert space that it operates on, thus preserving the inner product on vectors in the Hilbert space. This is important because Quantum Computing is entirely based upon unitary operators and how they are used to change the state of a quantum computer in order to output some information which solves a computational problem. In the same way that applying the Fourier Transform to elements of $L^2([0, 1])$, expressed as a linear combination of orthonormal basis elements $\{e_n\}_{n \in \mathbb{Z}}$, allowed Fourier to solve the heat equation, we will see that applying the Quantum Fourier Transform to elements of the group algebra over \mathbb{C} for some group G , $\mathbb{C}[G]$, expressed as a linear combination of an orthonormal basis whose elements are indexed by the group elements of G , will allow us to solve a range of computational problems on a quantum computer, all of which happen to be specific instances of a much greater problem known as the Hidden Subgroup Problem.

As we have briefly mentioned, a quantum computer has a state at any point in time, similar to its classical counterpart. But while the state of a classical computer can be observed without being affected by the observation, the state of a quantum computer could, and is likely to, change upon it being observed. This is because the model for Quantum Computing is founded upon the postulates of Quantum Mechanics, the third of which says that the state of the computer changes probabilistically according to the so-called ‘‘measurement operators’’ that are used to observe it. Furthermore, while the classical computer’s state is a composition of bits, whose values take on either 0 or 1, the quantum computer’s state is typically a (tensored) composition of qubits, which are unit vectors that can be expressed as linear combinations of some basis vectors in the complex Hilbert space \mathbb{C}^2 . Normally, this basis is an orthonormal basis known as the computational basis, which can be written as $\{|0\rangle, |1\rangle\}$ in Dirac notation, and the measurement operators are those that ‘‘collapse’’ the state of a qubit into one of two quantum states, $|0\rangle$ or $|1\rangle$, that are analogous to the possible states of a bit. However, neither the basis nor the measurement operators necessarily have to be these. Used well, the extra range of states that are possible during a quantum computation prior to an observation can result in quantum algorithms that are much more powerful than classical ones.

One of the major issues that occurs in Quantum Computing but not in Classical Computing as a consequence of the previous paragraph is that the act of observing the state of a quantum computer can lead to a level of ‘‘destructive interference’’, where the resulting state suffers a loss of information which was, in some sense, held in the state before it was observed. We will come to see that the Quantum Fourier Transform is very useful in this respect because, in the case where it is applied to the group algebra of an abelian group G , $\mathbb{C}[G]$, it is a transformation that maps the computational basis of $\mathbb{C}[G]$ to an orthonormal basis of the group algebra of the dual group of G , $\mathbb{C}[\widehat{G}]$, known as

the Fourier basis. It just so happens that $\mathbb{C}[\widehat{G}]$ is isomorphic to $\mathbb{C}[G]$, which we will come to see is crucial in the context of unitary operators in Quantum Computing. The elements of the Fourier basis are themselves linear combinations of a natural basis of $\mathbb{C}[\widehat{G}]$ indexed by the one-dimensional irreducible representations of G . By applying the Quantum Fourier Transform to a state of the quantum system $\mathbb{C}[G]$, we obtain a state in the other, isomorphic quantum system, $\mathbb{C}[\widehat{G}]$, expressed as a linear combination of elements from the Fourier basis. The key point is that when we choose measurement operators that collapse this new state into one of the irreducible representations of G and observe the new state, it results in the preservation of the information that was held in the original state of the former quantum system $\mathbb{C}[G]$, under the right conditions. This is very similar to how the Fourier Transform on $L^2([0, 1])$ made it possible for functions f in this space to be transformed into L^2 functions \widehat{f} on the dual group of $[0, 1]$, the integers \mathbb{Z} , which could then be worked with in order to obtain a solution to the heat equation in the original $L^2([0, 1])$ space.

Our efforts to understand the Quantum Fourier Transform will take us on a journey across a wide spectrum of topics in Mathematics, Physics and Computer Science: we will see how Quantum Mechanics is used to inform the basis of a model of computation for Quantum Computing; we will study a number of concepts from linear algebra in order to express our thoughts and ideas; we will see that quantum computers need all of its operators to be reversible, in direct contrast to Classical Computing; we will find ourselves venturing into number theory in order to analyse a quantum algorithm to factor integers on a quantum computer, before generalising our results and algorithms by investigating the application of the Quantum Fourier Transform to the Hidden Subgroup Problem. We will need to delve into the foundational results of Group Representation Theory, in particular, the Characters of Finite Abelian Groups, before we are able to give an algorithm that solves the Hidden Subgroup Problem in the case where the group is abelian. We will conclude with an investigation into what remains an open problem to this day: the Hidden Subgroup Problem in the case where the group is not abelian.

In the final presentation of this thesis, I have had to leave out my investigation into the nature of single qubits owing to a need for brevity. In particular, I have had to leave out the Bloch Sphere representation of a qubit, the influence of the Pauli matrices on single qubits, how there exists a unitary rotation operator $R_{\widehat{n}}(\theta)$ that acts on the state of a qubit in an equivalent manner to a rotation by an angle θ about the \widehat{n} axis of the Bloch Sphere, as well as a full characterisation of the local nature of unitary operators on the state of a composite quantum system formed from n qubits. In doing so, I believe that the final presentation, despite its length, is more compact, and I hope that it reads as a coherent representation of my foray into Quantum Computing.

Edward Pearce-Crump
Imperial College London
September 2020

2 The Quantum Computing Model

We begin by introducing the foundational concepts needed to study Quantum Computing. Quantum Computing can be thought of as a model for performing computations that is founded upon the postulates of Quantum Mechanics. The model describes how the state of a quantum system can be transformed through time in order to generate information that can be used to solve some sort of computational task. The model itself is often referred to as a quantum computer and the solution to a computational task is often called a quantum algorithm. This chapter is based upon the material found in [11], [14] and [21].

2.1 Quantum Systems and their States

The postulates of Quantum Mechanics are expressed in the language of linear algebra. We review some basic concepts from linear algebra before stating what the first postulate is.

Definition 2.1. An inner product space is a choice of vector space Q over \mathbb{C} augmented by a function $(*, *) : Q \times Q \rightarrow \mathbb{C}$ that satisfies the following:

1. $(*, *)$ is linear in the second argument, that is

$$\left(|v\rangle, \sum_i \lambda_i |w_i\rangle \right) = \sum_i \lambda_i (|v\rangle, |w_i\rangle) \quad (2.1)$$

where $\lambda_i \in \mathbb{C}$ for all i .

2. $(|v\rangle, |w\rangle)^* = (|w\rangle, |v\rangle)$, where $*$ denotes the complex conjugate.
3. $(|v\rangle, |v\rangle) \geq 0$ with equality if and only if $|v\rangle = 0$.

From this definition, we can show that any inner product is conjugate-linear in the first argument.

Remark 2.2. We will be studying finite-dimensional complex inner product spaces Q . Since they are exactly the same as complex Hilbert spaces and the postulates of Quantum Mechanics are expressed in the language of complex Hilbert spaces, we will write “complex Hilbert space” instead of “finite-dimensional complex inner product space” from now on. We will also assume that all vector spaces are finite-dimensional going forward.

Furthermore, if $|v\rangle, |w\rangle$ are two vectors in a complex Hilbert space Q , their inner product $(|v\rangle, |w\rangle)$ is often written instead as a “braket” $\langle v|w\rangle$. Hence, we will write $\langle *|*$ instead of $(*, *)$ where possible to denote the inner product associated with Q , reverting to the $(*, *)$ notation only when it is absolutely necessary.

Nevertheless, when we introduce a complex Hilbert space Q , often we won’t explicitly state what its accompanying inner product is when it is easily understood.

Definition 2.3. Let Q be a complex Hilbert space with inner product $\langle *|*$.

- Two vectors $|v\rangle, |w\rangle$ are said to be orthogonal in Q if $\langle v|w\rangle = 0$.
- The norm $\|*\|$ of a vector $|v\rangle \in Q$ is defined to be $\| |v\rangle \| := \sqrt{\langle v|v\rangle}$. A vector is called a unit vector if its norm equals 1.

- A set of vectors $\{|i\rangle\}_{i \in I}$ indexed by an index set I are said to be orthonormal in Q if each vector in the set is a unit vector and the relation $\langle i|j\rangle = \delta_{ij}$ holds for any two vectors $|i\rangle, |j\rangle$ in the set.

Definition 2.4. A qubit is a unit vector $|\psi\rangle$ in the Hilbert space $Q = \mathbb{C}^2$ with inner product $\langle *|*$.

Using Dirac notation, Q has an orthonormal basis $\{|0\rangle, |1\rangle\}$, which means that $|\psi\rangle$ can be written as $\alpha|0\rangle + \beta|1\rangle$ for some so-called “amplitudes” $\alpha, \beta \in \mathbb{C}$ such that $|\alpha|^2 + |\beta|^2 = 1$.

The qubit $|\psi\rangle$ is said to be in a superposition of the “basic” quantum states $|0\rangle$ and $|1\rangle$, which are analogues of the possible states of a bit, the fundamental unit of classical computing.

Theorem 2.5 (First Postulate of Quantum Mechanics). A quantum system is a complex Hilbert space Q with inner product $\langle *|*$ which has a quantum state described by a unit vector $|\psi\rangle$ in Q .

Q is usually the tensor product of some n quantum systems, each of which has a quantum state that is a qubit. We call quantum systems of this form either “typical” or “composite” quantum systems.

Therefore, Q is typically the Hilbert space $\bigotimes_{i=1}^n \mathbb{C}^2$ with an inner product defined by

$$\left\langle \bigotimes_{i=1}^n |\psi_i\rangle \left| \bigotimes_{i=1}^n |\phi_i\rangle \right. \right\rangle := \prod_{i=1}^n \langle \psi_i | \phi_i \rangle \quad (2.2)$$

where $|\psi_i\rangle, |\phi_i\rangle$ are qubits in \mathbb{C}^2 for all i .

Consequently, the quantum state $|\psi\rangle$ of a composite quantum system is a unit vector in Q whose state is held in n qubits.

In addition, Q has a standard orthonormal basis $\{|x_1\rangle \otimes \cdots \otimes |x_n\rangle \mid x_i \in \{0, 1\}\}$, which is called the computational basis of Q .

Since we can also write this basis set as $\{|x_1 \dots x_n\rangle \mid x_i \in \{0, 1\}\}$, it means that the computational basis of Q is the set of binary strings of length n .

Furthermore, since $\bigotimes_{i=1}^n \mathbb{C}^2 \cong \mathbb{C}^{2^n}$, we can also label the computational basis of Q as $\{|i\rangle\}_{i=0}^{2^n-1}$. We often find ourselves using this latter form to express the computational basis of a quantum system, but the first two are used where appropriate.

Ultimately, this means that the quantum state $|\psi\rangle$ of a typical quantum system Q can be expressed as a superposition of the computational basis

$$\sum_{i=0}^{2^n-1} \alpha_i |i\rangle \quad (2.3)$$

such that the amplitudes $\alpha_i \in \mathbb{C}$ satisfy $\sum_{i=0}^{2^n-1} |\alpha_i|^2 = 1$.

2.2 Transformations of Quantum States

We would like to understand what sort of transformations we can perform on the state of a quantum system and, in doing so, understand how the state changes with time. Again, we need more concepts from linear algebra before we can declare what the second postulate is. We first define a very important notation for representing linear operators on complex Hilbert spaces.

Definition 2.6 (Outer Product Operator). Let $|v\rangle$ be a vector in a complex Hilbert space V and let $|w\rangle$ be a vector in a complex Hilbert space W . Denoting the inner product of V by $\langle *|*_V$, we can define a linear operator, called the outer product operator, $|w\rangle \langle v| : V \rightarrow W$ as follows:

$$(|w\rangle \langle v|) |x\rangle := |w\rangle \langle v|x\rangle_V = \langle v|x\rangle_V |w\rangle \quad (2.4)$$

for all $|x\rangle \in V$, since $\langle v|x\rangle_V \in \mathbb{C}$.

Consequently, we can take linear combinations of outer product operators which defines an operator $V \rightarrow W$ as follows:

$$\left(\sum_i a_i |w_i\rangle \langle v_i| \right) |x\rangle := \sum_i a_i \langle v_i|x\rangle_V |w_i\rangle \quad (2.5)$$

With this, we obtain an important equality for the identity operator I_V on a complex Hilbert space V .

Example 2.7 (Completeness Relation). Let V be a complex Hilbert space, and let $\{|i\rangle\}$ be an orthonormal basis for V .

Then a vector $|v\rangle \in V$ can be written as a linear combination $\sum_i \alpha_i |i\rangle$ of the basis vectors such that $\sum_i |\alpha_i|^2 = 1$, with $\alpha_i \in \mathbb{C}$.

Consider now the linear operator $\sum_i |i\rangle \langle i| : V \rightarrow V$.

By the orthonormality of the basis, $\langle i|v\rangle = \alpha_i$ for all i , and so we have that

$$\left(\sum_i |i\rangle \langle i| \right) |v\rangle = \sum_i |i\rangle \langle i|v\rangle = \sum_i \alpha_i |i\rangle = |v\rangle \quad (2.6)$$

Because this is true for all vectors $|v\rangle \in V$, we must have that

$$\sum_i |i\rangle \langle i| = I_V \quad (2.7)$$

where $I_V : V \rightarrow V$ is the identity operator on V .

Equation (2.7) is called the Completeness Relation, which is a way of expressing the identity operator as a linear combination of outer product operators.

As a result, we can represent any linear operator in the outer product notation, as follows:

Example 2.8 (Outer Product Representation). Let $A : V \rightarrow W$ be a linear operator on two complex Hilbert spaces V and W . Suppose that $\{|v_i\rangle\}$ is an orthonormal basis of V and $\{|w_j\rangle\}$ is an orthonormal basis of W .

Then we can say that

$$A = I_W A I_V = \left(\sum_j |w_j\rangle \langle w_j| \right) A \left(\sum_i |v_i\rangle \langle v_i| \right) = \sum_{i,j} \langle w_j|A|v_i\rangle |w_j\rangle \langle v_i| \quad (2.8)$$

where $\langle w_j|A|v_i\rangle$ uses the inner product of W .

Hence A has been expressed as a linear combination of outer product operators.

Another important concept that will be used often is the following:

Definition 2.9 (Adjoint Operator). Let $A : V \rightarrow W$ be a linear operator between two complex Hilbert spaces V and W . Then the unique linear operator $A^\dagger : W \rightarrow V$ such that the following equality holds

$$(|w\rangle, A|v\rangle)_W = (A^\dagger|w\rangle, |v\rangle)_V \quad (2.9)$$

for all vectors $|v\rangle \in V$, $|w\rangle \in W$ is called the adjoint of A .

For a vector $|v\rangle \in V$, we define $|v\rangle^\dagger := \langle v|$.

Because we can prove that $(AB)^\dagger = B^\dagger A^\dagger$ from Equation (2.9), we also have that $(A|v\rangle)^\dagger = \langle v| A^\dagger$.

We prove some very important results involving adjoints that will be used often in later sections.

Exercise 2.10 (Adjoint of the Outer Product). If V and W are complex Hilbert spaces and $|v\rangle \in V$, $|w\rangle \in W$ are any two vectors in their respective vector spaces, show that

$$(|w\rangle \langle v|)^\dagger = |v\rangle \langle w| \quad (2.10)$$

as an operator $W \rightarrow V$.

Proof. Let $|a\rangle$ be any vector in W and let $|b\rangle$ be any vector in V . Then

$$\begin{aligned} & ((|w\rangle \langle v|)^\dagger |a\rangle, |b\rangle) \\ &= (|a\rangle, (|w\rangle \langle v|) |b\rangle) \\ &= (|a\rangle, \langle v|b\rangle |w\rangle) \\ &= \langle v|b\rangle (|a\rangle, |w\rangle) \\ &= \langle v|b\rangle \langle a|w\rangle \\ &= \langle a|w\rangle (|v\rangle, |b\rangle) \\ &= (\langle w|a\rangle |v\rangle, |b\rangle) \\ &= ((|v\rangle \langle w|) |a\rangle, |b\rangle) \end{aligned}$$

where we have used that the inner product is conjugate-linear in the first argument and linear in the second argument. We have also used the definition of the outer product too.

Hence, by the uniqueness of the adjoint, we obtain the result. \square

Exercise 2.11 (Anti-Linearity of the Adjoint). Show that the adjoint operation is anti-linear, that is,

$$\left(\sum_i a_i A_i \right)^\dagger = \sum_i a_i^* A_i^\dagger \quad (2.11)$$

where the $A_i : V \rightarrow W$ are linear operators between complex Hilbert spaces V and W and $a_i \in \mathbb{C}$ for all i .

Proof. Let $|v\rangle$ be any vector in V and let $|w\rangle$ be any vector in W . Then

$$\begin{aligned} & \left(\left(\sum_i a_i A_i \right)^\dagger |w\rangle, |v\rangle \right) \\ &= \left(|w\rangle, \sum_i a_i A_i |v\rangle \right) \\ &= \sum_i a_i (|w\rangle, A_i |v\rangle) \\ &= \sum_i a_i (A_i^\dagger |w\rangle, |v\rangle) \\ &= \left(\left(\sum_i a_i^* A_i^\dagger \right) |w\rangle, |v\rangle \right) \end{aligned}$$

where we have used that the inner product is conjugate-linear in the first argument and linear in the second argument.

Hence we obtain the result by the uniqueness of the adjoint. \square

Exercise 2.12. Show that $(A^\dagger)^\dagger = A$.

Proof. Let $|v\rangle$ be any vector in V and let $|w\rangle$ be any vector in W . Then

$$((A^\dagger)^\dagger |v\rangle, |w\rangle) = (|v\rangle, A^\dagger |w\rangle) = \langle v|A^\dagger|w\rangle = (A|v\rangle, |w\rangle) \quad (2.12)$$

Again, by the uniqueness of the adjoint, we get the result. \square

We can also apply the adjoint operator to a matrix, as follows:

Definition 2.13 (Adjoint of a Matrix). When we have a matrix $A \in \mathbb{C}^{m \times n}$ - for example, when looking at the matrix representation of a linear operator in some bases - we can say that its adjoint, $A^\dagger \in \mathbb{C}^{n \times m}$, is the conjugate transpose of A , that is, $(A^*)^T$.

We now come to one of the most important definitions in Quantum Computing.

Definition 2.14 (Unitary Operators). Let V be a complex Hilbert space.

A linear operator U on V is said to be unitary if $U^\dagger U$ equals the identity operator I_V .

With this definition, we can now state the second postulate of Quantum Mechanics.

Theorem 2.15 (Second Postulate of Quantum Mechanics). Any change in the state of a quantum system Q over time is described by a unitary operator. Hence, if $|\psi_i\rangle$ is the state of the quantum system at time t_i for $i = 1, 2$, then there is a unitary operator U such that

$$|\psi_2\rangle = U |\psi_1\rangle \quad (2.13)$$

Such operators U are also called quantum gates, or just gates.

Because the evolution of the quantum system is unitary, we need to provide some additional properties of unitary operators.

The first property is the following:

Lemma 2.16. Unitary operators U on a complex Hilbert space V preserve the inner product between vectors.

Proof. Let $|v\rangle, |w\rangle$ be vectors in V .

Then

$$(U|v\rangle, U|w\rangle) = \langle v|U^\dagger U|w\rangle = \langle v|I_V|w\rangle = \langle v|w\rangle \quad (2.14)$$

as required. \square

We obtain the outer product representation for unitary operators from this result, which we use often in what follows.

Lemma 2.17 (Outer Product Representation of Unitary Operators). Let V, W be complex Hilbert spaces of the same dimension.

1. Let $\{|v_i\rangle\}$ be any orthonormal basis of V . Suppose that $U : V \rightarrow W$ is a unitary operator, and define $|w_i\rangle := U |v_i\rangle$. Then $\{|w_i\rangle\}$ is an orthonormal basis of W , and hence $U = \sum_i |w_i\rangle \langle v_i|$.
2. Suppose instead that $\{|v_i\rangle\}$ and $\{|w_i\rangle\}$ are two orthonormal bases. Then $U := \sum_i |w_i\rangle \langle v_i|$ is a unitary operator $V \rightarrow W$.

Proof. 1. It is clear that $\{|w_i\rangle\}$ is an orthonormal basis of W , as unitary operators preserve inner products by Lemma 2.16.

Also, it is clear, for any basis element $|v_j\rangle$, that

$$\left(\sum_i |w_i\rangle \langle v_i| \right) |v_j\rangle = \sum_i \langle v_i | v_j \rangle |w_i\rangle = \sum_i \delta_{ij} |w_i\rangle = |w_j\rangle \quad (2.15)$$

which is exactly the action of U on the orthonormal basis of V .

2. We show that $U := \sum_i |w_i\rangle \langle v_i|$ is a unitary operator. Indeed, we have that

$$\begin{aligned} U^\dagger U &= \left(\sum_j |w_j\rangle \langle v_j| \right)^\dagger \left(\sum_i |w_i\rangle \langle v_i| \right) \\ &= \sum_i \sum_j |v_j\rangle \langle w_j | w_i \rangle \langle v_i| \\ &= \sum_i \sum_j \delta_{ij} |v_j\rangle \langle v_i| \\ &= \sum_i |v_i\rangle \langle v_i| \\ &= I_V \end{aligned}$$

where we have used the Completeness Relation (Equation (2.7)) in the last equation. □

Exercise 2.18. Show that all eigenvalues of a unitary matrix have modulus 1, that is, they can be written in the form $e^{i\theta}$ for some $\theta \in \mathbb{R}$.

Proof. Let the unitary matrix be denoted by U , operating on a vector space V . Then U is a unitary operator on V that satisfies $U^\dagger U = U U^\dagger = I$, and so it is also a normal operator. Hence it has a spectral decomposition - see [21] for details.

This means that there is an orthonormal basis $\{|i\rangle\}$ of V in which U is a diagonal matrix, and so it can be expressed as an operator in the form

$$U = \sum_i \lambda_i |i\rangle \langle i| \quad (2.16)$$

for some $\lambda_i \in \mathbb{C}$.

Now

$$\begin{aligned}
U^\dagger U &= \left(\sum_j \lambda_j |j\rangle \langle j| \right)^\dagger \left(\sum_i \lambda_i |i\rangle \langle i| \right) \\
&= \left(\sum_j \lambda_j^* |j\rangle \langle j| \right) \left(\sum_i \lambda_i |i\rangle \langle i| \right) \\
&= \sum_i \sum_j \lambda_i \lambda_j^* |j\rangle \langle j|i\rangle \langle i| \\
&= \sum_i \sum_j \lambda_i \lambda_j^* \delta_{ij} |j\rangle \langle i| \\
&= \sum_i |\lambda_i|^2 |i\rangle \langle i|
\end{aligned}$$

But $U^\dagger U = I = \sum_i |i\rangle \langle i|$ by the Completeness Relation, and so $|\lambda_i|^2 = 1$ for all i .

Hence the eigenvalues of U have modulus 1, that is, they are equal to $e^{i\theta}$ for some $\theta \in \mathbb{R}$. \square

The first postulate of Quantum Mechanics (Theorem 2.5) said that quantum systems are often the tensor product of quantum systems each of which has a quantum state that is a qubit. Therefore, we need to be able to apply unitary operators to these spaces too. The following looks at tensor product spaces and operators on them, using the definitions given in [6] and [21].

Definition 2.19. Suppose that V, W are complex Hilbert spaces.

If $\{|v_i\rangle\}$ and $\{|w_j\rangle\}$ are bases of V and W respectively, then $\{|v_i\rangle \otimes |w_j\rangle\}$ is a basis of $V \otimes W$.

Moreover, $V \otimes W$ is a complex Hilbert space with inner product

$$\left(\sum_i a_i [|v_i\rangle \otimes |w_i\rangle], \sum_j b_j [|v'_j\rangle \otimes |w'_j\rangle] \right) := \sum_i \sum_j a_i^* b_j \langle v_i | v'_j \rangle \langle w_i | w'_j \rangle \quad (2.17)$$

Furthermore, suppose that A is a linear operator on V and B is a linear operator on W .

Then we can define a linear operator $A \otimes B$ on $V \otimes W$ by

$$(A \otimes B) \left(\sum_i a_i [|v_i\rangle \otimes |w_i\rangle] \right) := \sum_i a_i [A |v_i\rangle \otimes B |w_i\rangle] \quad (2.18)$$

Exercise 2.20. Show that the adjoint operation distributes over the tensor product, that is

$$(A \otimes B)^\dagger = A^\dagger \otimes B^\dagger \quad (2.19)$$

as a map $V' \otimes W' \rightarrow V \otimes W$ for complex Hilbert spaces V, V', W and W' .

Proof. Let $|v\rangle \otimes |w\rangle$ be an arbitrary element of $V \otimes W$ and let $|v'\rangle \otimes |w'\rangle$ be an arbitrary element of $V' \otimes W'$. Writing these as $|vw\rangle, |v'w'\rangle$, we have that

$$\begin{aligned}
((A \otimes B)^\dagger |v'w'\rangle, |vw\rangle) &= (|v'w'\rangle, (A \otimes B) |vw\rangle) \\
&= (|v'w'\rangle, A |v\rangle \otimes B |w\rangle) \\
&= \langle v' | A |v\rangle \langle w' | B |w\rangle \\
&= \langle v' | (A^\dagger)^\dagger |v\rangle \langle w' | (B^\dagger)^\dagger |w\rangle \\
&= (A^\dagger |v'\rangle \otimes B^\dagger |w'\rangle, |vw\rangle) \\
&= ((A^\dagger \otimes B^\dagger) |v'w'\rangle, |vw\rangle)
\end{aligned}$$

where we have applied Definition 2.19 and Exercise 2.12.

Hence, by the uniqueness of the adjoint, we obtain the result. \square

Exercise 2.21. Show that the tensor product of two unitary operators on complex vector spaces V and W is unitary on the complex tensor product Hilbert space $V \otimes W$.

Proof. Let U_V, U_W be unitary operators on V and W respectively.

Let $|v\rangle \otimes |w\rangle$ be an arbitrary element of $V \otimes W$.

Then

$$\begin{aligned} (U_V \otimes U_W)^\dagger (U_V \otimes U_W) (|v\rangle \otimes |w\rangle) &= (U_V \otimes U_W)^\dagger (U_V |v\rangle \otimes U_W |w\rangle) \\ &= (U_V^\dagger \otimes U_W^\dagger) (U_V |v\rangle \otimes U_W |w\rangle) \\ &= U_V^\dagger U_V |v\rangle \otimes U_W^\dagger U_W |w\rangle \\ &= I_V |v\rangle \otimes I_W |w\rangle \\ &= (I_V \otimes I_W) (|v\rangle \otimes |w\rangle) \\ &= I_{V \otimes W} (|v\rangle \otimes |w\rangle) \end{aligned}$$

where we have applied Exercise 2.20 to $(U_V \otimes U_W)^\dagger$.

Hence $U_V \otimes U_W$ is unitary on $V \otimes W$, as required. \square

We can also consider matrix representations of linear operators on complex tensor product Hilbert spaces.

Definition 2.22 (Kronecker Product). Let V_i, W_i be complex Hilbert spaces and suppose that $L_i : V_i \rightarrow W_i$ are linear operators for $i = 1, 2$.

Let A_i be the $\dim W_i \times \dim V_i$ matrix representation of L_i for some bases of V_i and W_i , for $i = 1, 2$.

Consider now the linear operator $L_1 \otimes L_2 : V_1 \otimes W_1 \rightarrow V_2 \otimes W_2$

Then the matrix representation of $L_1 \otimes L_2$ in the bases naturally formed for $V_1 \otimes V_2$ and $W_1 \otimes W_2$ is given by $A_1 \otimes A_2$ and is defined to be the block matrix

$$A_1 \otimes A_2 = (a_{ij}^1 A_2) \tag{2.20}$$

where $A_1 = (a_{ij}^1)$. \otimes on matrices is called the Kronecker Product.

The matrix $A_1 \otimes A_2$ has size $(\dim W_1 \dim W_2) \times (\dim V_1 \dim V_2)$.

We are now in a position to provide a number of examples that shows the sorts of operations we can perform on the states of various quantum systems.

We first look at some unitary operators on qubits, before studying the most important unitary operators on composite quantum systems.

Example 2.23. We consider operators on qubits, typically defined on the computational basis $\{|0\rangle, |1\rangle\}$ of $Q = \mathbb{C}^2$.

1. The gate I , also called the identity gate, leaves the state of a qubit unchanged. Its outer representation, using Equation (2.8), is $I = |0\rangle\langle 0| + |1\rangle\langle 1|$, and its matrix representation in the computational basis is

$$\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \tag{2.21}$$

It is clear that $I^\dagger I = I^2 = I$, and so I is unitary.

2. The gate X , also called the NOT gate, maps $|0\rangle \mapsto |1\rangle$ and $|1\rangle \mapsto |0\rangle$, and can be extended linearly to describe its operation on a superposition state $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$. Its outer product representation is $X = |0\rangle\langle 1| + |1\rangle\langle 0|$, and its matrix representation in the computational basis is

$$\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \quad (2.22)$$

It is clear that $X^\dagger X = X^2 = I$, and so X is unitary.

3. The gate Z maps $|0\rangle \mapsto |0\rangle$ and $|1\rangle \mapsto -|1\rangle$. The -1 factor is also known as a phase factor. Its outer product representation is $Z = |0\rangle\langle 0| - |1\rangle\langle 1|$, and its matrix representation in the computational basis is

$$\begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \quad (2.23)$$

It is clear that $Z^\dagger Z = Z^2 = I$, and so Z is unitary.

4. The rotation gate, R_k , for some $k \in \mathbb{Z}$, changes the phase of a state; in particular, it maps $|0\rangle \mapsto |0\rangle$ and $|1\rangle \mapsto e^{\frac{2\pi i}{2^k}}|1\rangle$. Its matrix representation in the computational basis is therefore

$$\begin{pmatrix} 1 & 0 \\ 0 & e^{\frac{2\pi i}{2^k}} \end{pmatrix} \quad (2.24)$$

It is easy to show that $R_k^\dagger R_k = I$, and so R_k is unitary.

Furthermore, note that sometimes the gate R_2 (i.e $k = 2$) is labelled as S instead, and has a matrix representation

$$\begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix} \quad (2.25)$$

5. The Hadamard gate, H , is a very important unitary operator in Quantum Computing. It maps

$$|0\rangle \mapsto \frac{|0\rangle + |1\rangle}{\sqrt{2}} \quad |1\rangle \mapsto \frac{|0\rangle - |1\rangle}{\sqrt{2}} \quad (2.26)$$

and so its outer product representation is

$$H = \frac{1}{\sqrt{2}} [(|0\rangle + |1\rangle)\langle 0| + (|0\rangle - |1\rangle)\langle 1|] \quad (2.27)$$

with matrix representation in the computational basis

$$\frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \quad (2.28)$$

It is easy to show that $H^\dagger H = H^2 = I$, and so H is unitary.

It is common to draw diagrams to represent all of these gates except for the identity gate, and so the above gates are given by

$$\text{---} \boxed{X} \text{---} \quad \text{---} \boxed{Z} \text{---} \quad \text{---} \boxed{R_k} \text{---} \quad \text{---} \boxed{S} \text{---} \quad \text{---} \boxed{H} \text{---} \quad (2.29)$$

The most important gate on a composite quantum system is the following:

Exercise 2.24. Show that the Hadamard operation on n qubits, $H^{\otimes n}$, may be written as

$$H^{\otimes n} = \frac{1}{\sqrt{2^n}} \sum_{x,y \in \{0,1\}^n} (-1)^{x \cdot y} |y\rangle \langle x| \quad (2.30)$$

The diagram for this gate is

$$\text{---}^n \boxed{H^{\otimes n}} \text{---} \quad (2.31)$$

where the “dash n ” indicates that there are n wires going in and coming out of the gate, representing its action on an n qubit state.

Proof. From Equation (2.26) we can see that

$$H |z_i\rangle = \frac{1}{\sqrt{2}} \sum_{y_i \in \{0,1\}} (-1)^{z_i y_i} |y_i\rangle \quad (2.32)$$

for $z_i \in \{0,1\}$.

Since $H^{\otimes n}$ is an operator on $Q := \bigotimes_{i=1}^n \mathbb{C}^2$, we can write the elements $|z\rangle, |y\rangle$ of the computational basis of Q as $|z\rangle = |z_1 z_2 \dots z_n\rangle$ and $|y\rangle = |y_1 y_2 \dots y_n\rangle$, where $z_i, y_i \in \{0,1\}$ for all i .

Hence, we have that

$$\begin{aligned} H^{\otimes n} |z\rangle &= \bigotimes_{i=1}^n H |z_i\rangle \\ &= \bigotimes_{i=1}^n \left(\frac{1}{\sqrt{2}} \sum_{y_i \in \{0,1\}} (-1)^{z_i y_i} |y_i\rangle \right) \\ &= \frac{1}{\sqrt{2^n}} \sum_{y \in \{0,1\}^n} (-1)^{\sum_{i=1}^n z_i y_i} |y\rangle \\ &= \frac{1}{\sqrt{2^n}} \sum_{y \in \{0,1\}^n} (-1)^{z \cdot y} |y\rangle \end{aligned} \quad (2.33)$$

Now, since

$$\begin{aligned} \left(\frac{1}{\sqrt{2^n}} \sum_{x,y \in \{0,1\}^n} (-1)^{x \cdot y} |y\rangle \langle x| \right) |z\rangle &= \frac{1}{\sqrt{2^n}} \sum_{x,y \in \{0,1\}^n} (-1)^{x \cdot y} \langle x|z\rangle |y\rangle \\ &= \frac{1}{\sqrt{2^n}} \sum_{x,y \in \{0,1\}^n} (-1)^{x \cdot y} \delta_{x,z} |y\rangle \\ &= \frac{1}{\sqrt{2^n}} \sum_{y \in \{0,1\}^n} (-1)^{z \cdot y} |y\rangle \\ &= H^{\otimes n} |z\rangle \end{aligned} \quad (2.34)$$

we obtain the result, as required. \square

Example 2.25. In particular, we can express the matrix representation of the unitary operator $H^{\otimes 2}$ for the computational basis of $\mathbb{C}^2 \otimes \mathbb{C}^2$ explicitly, using Equation (2.28):

$$H^{\otimes 2} = \frac{1}{\sqrt{2}} \begin{pmatrix} H & H \\ H & -H \end{pmatrix} = \frac{1}{2} \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{pmatrix} \quad (2.35)$$

Example 2.26. Other important gates on two qubits, that is, on $\mathbb{C}^2 \otimes \mathbb{C}^2$, are defined on the computational basis $\{|00\rangle, |01\rangle, |10\rangle, |11\rangle\}$, where, by Theorem 2.5, each digit in a computational basis state can be and is referred to as a bit. For example, for the state $|10\rangle$, the first bit is 1 and the second bit is 0.

1. The controlled-NOT, or CNOT gate, performs the NOT gate on the second bit if and only if the first bit is 1. That is, it maps $|x, y\rangle \mapsto |x, x \oplus y\rangle$, where \oplus is addition modulo 2. Hence its matrix representation is

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \quad (2.36)$$

We will draw this gate in either of the following equivalent forms



$$\quad (2.37)$$

2. More generally, if U is a unitary operator on a qubit, then the controlled- U gate performs the U gate on the second bit if and only if the first bit is 1, that is, it maps $|x, y\rangle \mapsto |x\rangle U^x |y\rangle$. The diagram for this gate is



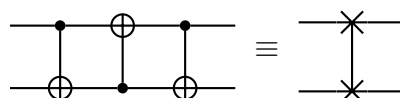
$$\quad (2.38)$$

3. Another important operation is the SWAP gate on two qubits, that is, the map $|x, y\rangle \mapsto |y, x\rangle$. We can implement this as a series of controlled-NOT gates:

$$|x, y\rangle \mapsto |x, x \oplus y\rangle \mapsto |x \oplus (x \oplus y), x \oplus y\rangle = |y, x \oplus y\rangle \mapsto |y, y \oplus (x \oplus y)\rangle = |y, x\rangle \quad (2.39)$$

Note that in the second mapping, we have used a version of the controlled-NOT gate that performs the NOT gate on the first bit if and only if the second bit is 1. This is okay to do; we just need to define which bit in a computational basis state is the “control” bit and which is the “target” bit when performing a controlled-NOT operation.

We can draw the SWAP gate in either of the following equivalent forms:



$$\quad (2.40)$$

Before looking at gates on quantum systems with n qubits, we look at a special gate on a quantum system with three qubits:

Example 2.27. The Toffoli gate on three qubits, that is, on $\mathbb{C}^2 \otimes \mathbb{C}^2 \otimes \mathbb{C}^2$, is defined on the computational basis as follows:

$$\text{Toffoli} : |x_1, x_2, x_3\rangle \rightarrow |x_1, x_2, x_3 \oplus x_1x_2\rangle \tag{2.41}$$

where $x_i \in \{0, 1\}$ for all i and \oplus is, once again, addition modulo 2.

The first two bits of a computational basis state are called control bits, and the third bit is called a target bit. The names come from the fact that if $x_1 = x_2 = 1$, then the third bit in the output of the Toffoli gate becomes $|\neg x_3\rangle$ (logical NOT), otherwise the third bit is unchanged by the Toffoli gate.

This is easily shown to be a unitary operator which is also its own inverse.

The matrix representation of this gate in the computational basis is 8×8 , so we won't state it here. However, the diagram is shown below:



We can also change around the order of the operation, that is, say, make the first and third bits the control bits and the second the target bit, in which case the mapping is

$$|x_1, x_2, x_3\rangle \rightarrow |x_1, x_2 \oplus x_1x_3, x_3\rangle \tag{2.43}$$

on the computational basis, and the diagram of the gate becomes



We still call this a Toffoli gate. When using these gates, we simply need to specify which two bits of a computational basis state are the control bits and which one is the target bit.

Example 2.28. We are able to give a few more examples of unitary operators on quantum systems of n qubits, $Q = \bigotimes_{i=1}^n \mathbb{C}^2$. We have already seen one example in Exercise 2.24, the Hadamard gate on n qubits, $H^{\otimes n}$.

1. One of the most important ideas in Quantum Computing is the ability to construct an n -qubit gate by tensoring a number of one, two or three qubit gates together with the identity gate.

For example, if we wish to apply the Hadamard gate to the first qubit and the NOT gate to the third qubit of the current state, leaving the rest unchanged, we can apply the operator $H \otimes I \otimes X \otimes I^{\otimes(n-3)}$ to achieve the desired result.

2. A more general controlled- U gate exists than the one defined on two qubits in Example 2.26. If we have a unitary operator U that operates on $n - 1$ qubits, then we can have an extra qubit which acts as a control qubit in the following sense:

We define a unitary operator U' on n qubits as follows: U' maps $|0\rangle |v\rangle \mapsto |0\rangle |v\rangle$ and $|1\rangle |v\rangle \mapsto |1\rangle U |v\rangle$, where $|v\rangle$ is a state on $n - 1$ qubits.

The diagram for this gate is



We've already seen an example of this sort of gate. If we let $n = 3$ and let U be the CNOT gate, then U' is the Toffoli gate of Example 2.27.

We have seen the first two postulates of Quantum Mechanics and looked at unitary operators in some depth. We study the third and final postulate in the next section.

2.3 Measurement of Quantum States

We know that a quantum system has a state which can be evolved throughout time by unitary operators. If the quantum system is composite and made up of n qubits, then we know from Equation (2.3) that its state is given by

$$\sum_{i=0}^{2^n-1} \alpha_i |i\rangle \quad (2.46)$$

such that the amplitudes $\alpha_i \in \mathbb{C}$ satisfy $\sum_{i=0}^{2^n-1} |\alpha_i|^2 = 1$.

Suppose that we now want to observe the system to derive some sort of information from it. The third postulate of Quantum Mechanics tells us what happens:

Theorem 2.29 (Third Postulate of Quantum Mechanics). There is a collection of measurement operators $\{M_i\}$ whose index describes all the possible measurement outcomes that may occur from observing the state $|\psi\rangle$ of the quantum system (as given in Equation (2.46)).

In particular, they must satisfy the Completeness Relation of Example 2.7 in the following sense

$$\sum_i M_i^\dagger M_i = I \quad (2.47)$$

The measurement operators are defined by the basis in which we wish to observe the quantum system.

Remark 2.30. There are two types of measurement operators that we will typically use, unless stated otherwise.

1. The first type are those that measure the entire quantum system in the computational basis (that is, measure all n qubits at once), and they are given by the outer product operators

$$\{M_i := |i\rangle \langle i|_{i=0}^{2^n-1}\} \quad (2.48)$$

Note that $M_i^\dagger M_i = M_i$ for all i and so they satisfy the Completeness Relation.

A probability distribution is induced by measuring the state $|\psi\rangle$ in these operators, and is given by

$$P(i) = \langle \psi | M_i^\dagger M_i | \psi \rangle = |\alpha_i|^2 \quad (2.49)$$

The state of the system, after measuring index i with probability $p(i)$, collapses to the state $|i\rangle$.

2. The second type are those that perform a partial measurement of the quantum system in the computational basis. It is enough to consider measurement operators that measure only one qubit of Equation (2.46) because it can be shown that performing two measurements one after the other is the same as performing a single measurement - see [21] for more details.

We can describe these measurement operators using the following

$$M_{0j} = I^{\otimes j-1} \otimes |0\rangle \langle 0| \otimes I^{\otimes n-j} \quad \text{and} \quad M_{1j} = I^{\otimes j-1} \otimes |1\rangle \langle 1| \otimes I^{\otimes n-j} \quad (2.50)$$

where the second index means measure qubit j only (which has only two possible outcomes, $|0\rangle$ and $|1\rangle$.)

Clearly, $M_{ij}^\dagger M_{ij} = I$, the identity operator on the n -qubit system, for $i = 0, 1$, and so their sum over i satisfies the Completeness Relation.

As before, a probability distribution is induced by measuring $|\psi\rangle$ in these operators, and it is given by

$$P_j(i) = \langle \psi | M_{ij}^\dagger M_{ij} | \psi \rangle \quad (2.51)$$

for $i = 0, 1$.

After measuring, the state collapses to a superposition of those basis elements in $|\psi\rangle$ whose j th qubit is the measurement outcome i with probability $P_j(i)$. This new superposition is renormalised so that the state remains a unit vector in the quantum system.

Example 2.31. We give some examples of measuring states of quantum systems using the two types of measurement operators given in Remark 2.30.

1. Suppose that we have a qubit in the state $|\psi\rangle := \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$ and suppose that we wish to measure this state. Our measurement operators are $M_0 = |0\rangle\langle 0|$ and $M_1 = |1\rangle\langle 1|$, and so by applying Equation (2.49), the state collapses to either $|0\rangle$ or $|1\rangle$ with an equal probability of $\frac{1}{2}$.

2. Suppose instead that we have a state $|\psi\rangle := \frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|11\rangle$ in a two-qubit quantum system. Suppose that we want to measure the second qubit only. Then the measurement operators are $M_{02} = I \otimes |0\rangle\langle 0|$ and $M_{12} = I \otimes |1\rangle\langle 1|$.

By applying Equation (2.51), we obtain a 0 with probability $\frac{1}{2}$, with the state collapsing in this case to $|00\rangle$, and we obtain a 1 with probability $\frac{1}{2}$, with the state collapsing in this case to $|11\rangle$.

3. Suppose now that we have the state $|\psi\rangle := \frac{1}{\sqrt{2}}|000\rangle + \frac{1}{2}|110\rangle + \frac{1}{2}|101\rangle$ in a three-qubit quantum system.

Suppose that we wish to measure the third qubit. Then the measurement operators are $M_{03} = I^{\otimes 2} \otimes |0\rangle\langle 0|$ and $M_{13} = I^{\otimes 2} \otimes |1\rangle\langle 1|$.

By applying Equation (2.51), we obtain a 0 with probability $\frac{3}{4}$, with the state collapsing in this case to $\frac{\sqrt{2}}{\sqrt{3}}|000\rangle + \frac{1}{\sqrt{3}}|110\rangle$ (where we renormalised the superposition resulting from the measurement), and we obtain a 1 with probability $\frac{1}{4}$, with the state collapsing in this case to $|101\rangle$.

Suppose that we want to measure the the second qubit after the measurement of the third qubit. Then the measurement operators are $M_{02} = I \otimes |0\rangle\langle 0| \otimes I$ and $M_{12} = I \otimes |1\rangle\langle 1| \otimes I$ and we obtain

- A 0 in the second qubit and a 0 in the third qubit with probability $\frac{3}{4} \times \frac{2}{3} = \frac{1}{2}$, and the state collapses to $|000\rangle$.
- A 1 in the second qubit and a 0 in the third qubit with probability $\frac{3}{4} \times \frac{1}{3} = \frac{1}{4}$, and the state collapses to $|110\rangle$.
- A 0 in the second qubit and a 1 in the third qubit with probability $\frac{1}{4} \times 1 = \frac{1}{4}$, and the state collapses to $|101\rangle$.
- A 1 in the second qubit and a 1 in the third qubit with probability $\frac{1}{4} \times 0 = 0$.

Notice that performing the measurement of the third qubit followed by the measurement of the second qubit could have been performed in one measurement by using the measurement operators $\{L_{ij}\}$ defined by

$$\begin{aligned} L_{00} &:= I \otimes |0\rangle\langle 0| \otimes |0\rangle\langle 0| = M_{02}M_{03} \\ L_{01} &:= I \otimes |0\rangle\langle 0| \otimes |1\rangle\langle 1| = M_{02}M_{13} \\ L_{10} &:= I \otimes |1\rangle\langle 1| \otimes |0\rangle\langle 0| = M_{12}M_{03} \\ L_{11} &:= I \otimes |1\rangle\langle 1| \otimes |1\rangle\langle 1| = M_{12}M_{13} \end{aligned}$$

on the original state $|\psi\rangle$.

2.4 Quantum Algorithms

We are now in a position to summarise the model for Quantum Computing, and describe what a quantum algorithm is in terms of this model.

Theorem 2.32 (Model of Quantum Computing). The model of Quantum Computing is as follows:

1. It is taken as given that we can always prepare $|0\rangle^{\otimes n}$ to be the initial state of a composite quantum system consisting of n qubits.

If we want a different state $|x\rangle = |x_1 \dots x_n\rangle$, where $x_i \in \{0, 1\}$ for all i , to be the initial state instead, we can apply quantum NOT gates to the various qubits i of $|0\rangle^{\otimes n}$ where we want x_i to be 1. We would still consider $|x\rangle$ to be the initial state of the quantum system even though we have applied some NOT gates to $|0\rangle^{\otimes n}$.

For example, if we want the input state $|0 \dots 01\rangle$ (sometimes also just written as $|1\rangle$), then we apply the operator $I^{\otimes n-1} \otimes X$ to $|0\rangle^{\otimes n}$ to prepare this as the initial state.

2. The evolution of the state of the quantum system through time is given by the Second Postulate of Quantum Mechanics (Theorem 2.15). This says that the change in state is described by a unitary operator U .
3. A measurement of some or all of the qubits in the quantum state is performed according to the Third Postulate of Quantum Mechanics (Theorem 2.29). From this we derive some information that can be used to obtain the solution to some computational task.

We sometimes refer to the model of Quantum Computing as a “quantum computer”.

Remark 2.33. It is worth noting that sometimes we split the initial state of a quantum system up into a number of so-called “quantum registers” and study the evolution of the registers through time.

For example, we could split the initial state of an n -qubit quantum system up into two quantum registers, the first register consisting of t qubits in the initial state $|0\rangle^{\otimes t}$ and the second register consisting of L qubits in the initial state $|1\rangle = |0 \dots 01\rangle$ such that $t + L = n$.

Here we would say that this quantum system consists of two registers, a t -qubit register and an L -qubit register, and we would write the initial state of the system as $|0\rangle^{\otimes t} |1\rangle$.

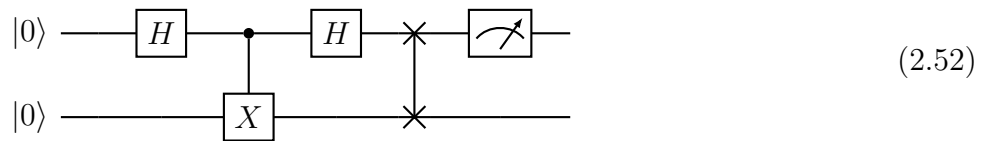
Definition 2.34 (Quantum Algorithm). A quantum algorithm is defined to be a series of computational steps that can be performed on a quantum computer, as defined in Theorem 2.32.

Definition 2.35. A quantum circuit is a diagram that consists of lines representing so-called “quantum wires” carrying qubits and so-called “quantum gates” representing unitary operators that are applied to quantum states. A quantum circuit provides a visualisation of the steps of a quantum algorithm.

Not every quantum algorithm can be described in terms of a quantum circuit. However, if a quantum algorithm can be described by a quantum circuit, then such a circuit is called an implementation of the quantum algorithm.

The operations given in quantum circuits are performed from left to right, showing how the state of a quantum system evolves through time by the unitary operators that are applied to it. They cannot merely be looked at to understand what algorithm they represent; each operator at every time step needs to be performed and explicitly calculated in order to avoid making any computational errors.

Example 2.36. For example, the quantum circuit



describes how the initial state $|0\rangle|0\rangle$ of a two-qubit quantum system evolves through time.

In the first time step, the Hadamard gate H is applied to the first qubit, giving the state $\frac{1}{\sqrt{2}}|0\rangle|0\rangle + \frac{1}{\sqrt{2}}|1\rangle|0\rangle$.

A controlled-NOT gate is then applied to this state, giving $\frac{1}{\sqrt{2}}|0\rangle|0\rangle + \frac{1}{\sqrt{2}}|1\rangle|1\rangle$.

Then the Hadamard gate H is again applied to the first qubit, giving the state $\frac{1}{2}|0\rangle|0\rangle + \frac{1}{2}|0\rangle|1\rangle + \frac{1}{2}|1\rangle|0\rangle - \frac{1}{2}|1\rangle|1\rangle$.

The first qubit is then swapped with the second qubit. In this case, the operation is superfluous as the state remains unchanged.

Finally the first qubit is measured.

The output state of this quantum algorithm is either $\frac{1}{\sqrt{2}}|0\rangle|0\rangle + \frac{1}{\sqrt{2}}|0\rangle|1\rangle$ with probability $\frac{1}{2}$ or $\frac{1}{\sqrt{2}}|1\rangle|0\rangle - \frac{1}{\sqrt{2}}|1\rangle|1\rangle$ with probability $\frac{1}{2}$.

Definition 2.37 (Efficient Implementation). We want to find and study quantum algorithms whose implementations have a number of gates that is polynomial in the size of the input qubits. Such implementations are called efficient.

2.5 Density Operators

There is an equivalent formulation of the postulates of Quantum Mechanics that uses a so-called “density operator” instead of a state vector. It is particularly useful when the state of a quantum system is in a superposition of computational basis states. We occasionally use this approach to describe some quantum algorithms, particularly in Chapter 7.

Definition 2.38 (Density Operator). Suppose that a quantum system is in one of a number of states $\{|\psi_i\rangle\}_{i \in I}$, where I is an index set, with probabilities $\{p_i\}_{i \in I}$.

The pairs $\{(p_i, |\psi_i\rangle)\}_{i \in I}$ are called an ensemble of pure states.

The density operator for the quantum system is defined by the equation

$$\rho := \sum_{i \in I} p_i |\psi_i\rangle \langle \psi_i| \quad (2.53)$$

The density operator is sometimes also referred to as the density matrix.

If a quantum system is in a state $|\psi\rangle$ that is known exactly, then it is said to be a pure state, and the equivalent density operator is $\rho = |\psi\rangle \langle \psi|$. Otherwise, ρ is said to be in a mixed state.

Example 2.39. Suppose that a quantum state $|\psi\rangle$ of a typical quantum system Q of n qubits is in some superposition of the computational basis, that is

$$|\psi\rangle = \sum_{i=0}^{2^n-1} \alpha_i |i\rangle \quad (2.54)$$

such that the amplitudes $\alpha_i \in \mathbb{C}$ satisfy $\sum_{i=0}^{2^n-1} |\alpha_i|^2 = 1$.

Then Q is in one of the states $\{|i\rangle\}_{i=0}^{2^n-1}$ with probabilities $\{|\alpha_i|^2\}_{i=0}^{2^n-1}$, and so the density operator for the system is

$$\rho = \sum_{i=0}^{2^n-1} |\alpha_i|^2 |i\rangle \langle i| \quad (2.55)$$

Lemma 2.40 (Evolution of Density Operators). Suppose that the evolution of a quantum system is described by the unitary operator U .

Then the evolution of the density operator is described by the map $\rho \mapsto U\rho U^\dagger$.

Proof. We provide the proof as given in [21].

If the system were in one of the pure states $|\psi_i\rangle$ in the ensemble with probability p_i , then it would be evolved to the state $U|\psi_i\rangle$ with probability p_i under the unitary operator U . Hence

$$\rho = \sum_{i \in I} p_i |\psi_i\rangle \langle \psi_i| \mapsto \sum_{i \in I} p_i U|\psi_i\rangle \langle \psi_i| U^\dagger = U \left[\sum_{i \in I} p_i |\psi_i\rangle \langle \psi_i| \right] U^\dagger = U\rho U^\dagger \quad (2.56)$$

□

Lemma 2.41 (Measurement of Density Operators). Suppose that the measurement operators for performing a measurement on a density operator ρ are given by $\{M_j\}$ for some index j that specifies the possible measurement outcomes.

Then the probability of observing outcome j , $P(j)$, is given by

$$P(j) = \text{tr}(M_j^\dagger M_j \rho) \quad (2.57)$$

where tr is the trace of $M_j^\dagger M_j \rho$, considered as a linear operator.

Proof. We provide the proof as given in [21].

Indeed, for a state $|\psi_i\rangle$ of a quantum system Q and a linear operator A , we have that

$$\text{tr}(A |\psi_i\rangle \langle \psi_i|) = \sum_k \langle k|A|\psi_i\rangle \langle \psi_i|k\rangle = \langle \psi_i|A|\psi_i\rangle \quad (2.58)$$

where we have extended $|\psi_i\rangle$ to an orthonormal basis of Q via the Gram-Schmidt procedure.

Therefore, if the original state of the system was some $|\psi_i\rangle$ from the ensemble of pure states, then the probability of obtaining result j , $P(j|i)$, is

$$P(j|i) = \langle \psi_i | M_j^\dagger M_j | \psi_i \rangle = \text{tr}(M_j^\dagger M_j | \psi_i \rangle \langle \psi_i |) \quad (2.59)$$

by Equation (2.58).

Hence

$$\begin{aligned} P(j) &= \sum_i p_i P(j|i) \\ &= \sum_i p_i \text{tr}(M_j^\dagger M_j | \psi_i \rangle \langle \psi_i |) \\ &= \text{tr} \left(M_j^\dagger M_j \left[\sum_i p_i | \psi_i \rangle \langle \psi_i | \right] \right) \\ &= \text{tr}(M_j^\dagger M_j \rho) \end{aligned}$$

by the Law of Total Probability. □

Remark 2.42. It can be shown that if a measurement results in outcome j with probability $P(j)$, then the density operator ρ collapses to

$$\rho_j = \frac{M_j \rho M_j^\dagger}{\text{tr}(M_j^\dagger M_j \rho)} \quad (2.60)$$

See [21] for details.

Remark 2.43. The postulates of Quantum Mechanics can therefore be reformulated in the language of density operators. By studying the density operator further and providing a full characterisation of it, we could reformulate the postulates without referring to the notion of a state vector in a quantum system. We won't do this as we won't have any use for it, but see [21] for more details.

2.6 Reversible Operations

In this section we show that any classical computation can be replicated on a quantum computer reversibly using only Toffoli and quantum NOT gates. We have used [7] to motivate the content of this section, although we provide a different and more intuitive implementation than the one that is given in [7].

We know that any computation on a classical computer can be built out of three basic components: the classical AND gate, the classical NOT gate, and the FANOUT operation, which is the operation that creates a copy of a bit in another wire. Each of these components is an operator on bits that evolves their state. We would like to replicate these operations on a quantum computer.

However, the second postulate (Theorem 2.15) says that quantum evolution needs to be unitary. In particular, this means that operators on quantum states need to be reversible, for if the unitary operator U evolves the state $|\psi_1\rangle$ to $|\psi_2\rangle$, then U^\dagger recovers the original state $|\psi_1\rangle$ from $|\psi_2\rangle$.

So while the classical NOT gate has a direct unitary counterpart in the quantum NOT gate (X), the classical AND gate does not. In particular, the classical AND gate is not reversible, since it can be

seen from the truth table that the original input bits cannot be recovered by reversing the operation on the output bit. In addition, Wootters and Zurek [27] showed that quantum states cannot be directly copied without being modified in the attempt to copy them.

We saw in Example 2.27 the definition of the Toffoli gate. It performed the operation

$$\text{Toffoli} : |x_1, x_2, x_3\rangle \rightarrow |x_1, x_2, x_3 \oplus x_1x_2\rangle \quad (2.61)$$

on the computational basis of $\mathbb{C}^2 \otimes \mathbb{C}^2 \otimes \mathbb{C}^2$, where $x_i \in \{0, 1\}$ for all i and \oplus is addition modulo 2.

We can use this gate to implement both the classical AND operation and the FANOUT operation reversibly through the use of extra ‘‘ancilla’’ qubits, which are extra qubits that help in performing an operation reversibly. Often we obtain the result with extra ‘‘garbage’’ qubits, which are qubits in the output whose states are not needed in the computation.

The classical AND operation can be implemented by applying the Toffoli gate to the state $|a, b, 0\rangle$ for $a, b \in \{0, 1\}$ and looking at the result in the third qubit of the output state, since

$$\text{Toffoli} |a, b, 0\rangle = |a, b, ab\rangle = |a, b, a \wedge b\rangle \quad (2.62)$$

Note that the third qubit in the input is an ancilla qubit, and that the first two qubits in the output of this operation become garbage qubits.

The FANOUT operation can be implemented in a similar way by applying the Toffoli gate to the state $|a, 1, 0\rangle$ since it outputs the state $|a, 1, a\rangle$. The second and third qubits in the input are ancilla qubits, and the second qubit in the output becomes a garbage qubit.

We’ve already seen that the classical NOT gate can be implemented using the unitary quantum NOT gate.

Therefore, since these three classical operations can be implemented reversibly on a quantum computer, and any classical computation can be constructed out of these three operations, it means that we are able to implement any classical operation on a quantum computer using just Toffoli and quantum NOT gates. We now show how to do this in detail.

Let C be a classical circuit that takes some x expressed in some n input bits to some output $y := C(x)$ expressed in k bits. This means that, on a quantum computer, the input can be held in n qubits and the output can be held in k qubits, since we know from Theorem 2.5 that the computational basis of a n -qubit quantum system, $\bigotimes_{i=1}^n \mathbb{C}^2$, can be written as $\{|x_1 \dots x_n\rangle \mid x_i \in \{0, 1\}\}$, and similarly for a k -qubit quantum system.

We can therefore write the classical x as the quantum state $|x\rangle = |x_1 \dots x_n\rangle$ and the classical y as the quantum state $|y\rangle = |C(x)\rangle = |y_1 \dots y_k\rangle$.

We show how to construct a quantum circuit \widehat{C} that maps

$$\widehat{C} : |x\rangle |0\rangle^{\otimes k} \mapsto |x\rangle |C(x)\rangle \quad (2.63)$$

in a reversible fashion using only Toffoli and quantum NOT gates. This quantum circuit clearly replicates the classical circuit C .

Since C can be constructed out of classical AND, NOT and FANOUT operations, it means that we can construct a quantum gate C' corresponding to the classical operator C which reversibly maps

$$C' : |x\rangle |0\rangle^{\otimes k} |0\rangle^{\otimes m} \mapsto |x\rangle |y\rangle |g(x)\rangle \quad (2.64)$$

where $|0\rangle^{\otimes m}$ are some m extra ancilla qubits in the input and $|g(x)\rangle$ is the garbage, dependent on the input $|x\rangle$, which is left over from performing the quantum counterparts to the classical AND, NOT and FANOUT operations.

The register containing the garbage is problematic because there could be interference between the first and third register when the circuit C' is performed on states that are not in the computational basis of $\bigotimes_{i=1}^n \mathbb{C}^2$. Any type of measurement on the third register could change the state of the first register, which is not what we want.

However, we can use the “trick of uncomputation” which uses the reversibility of C' to erase the garbage from the third register by applying $(C')^\dagger$ to the first three registers. But we need a way to retain the result $y = C(x)$ first before applying this, as uncomputation will also remove $|y\rangle$ if we haven't stored the result elsewhere first.

The following procedure constructs the quantum circuit \widehat{C} from this quantum gate C' :

1. Start with five registers in the state

$$|x\rangle |0\rangle^{\otimes k} |0\rangle^{\otimes m} |0\rangle^{\otimes k} |0\rangle \quad (2.65)$$

where the last three registers all hold ancilla qubits.

2. Apply C' to the first three registers to give

$$|x\rangle |y\rangle |g(x)\rangle |0\rangle^{\otimes k} |0\rangle \quad (2.66)$$

3. Apply the quantum NOT gate X to the fifth register only, which gives

$$|x\rangle |y\rangle |g(x)\rangle |0\rangle^{\otimes k} |1\rangle \quad (2.67)$$

4. Apply k Toffoli gates to copy the bits that make up y , where the qubits in the second and fifth registers are control qubits and the qubits in the fourth register are the target qubits.

More explicitly, using k Toffoli gates which each perform the map $|z_2, z_4, z_5\rangle \mapsto |z_2, z_4 \oplus z_2 z_5, z_5\rangle$, we have that $|y_i, 0, 1\rangle \mapsto |y_i, y_i, 1\rangle$ for all i from 1 to k .

This gives

$$|x\rangle |y\rangle |g(x)\rangle |y\rangle |1\rangle \quad (2.68)$$

5. Apply $(C')^\dagger$ to the first three registers to erase the garbage $g(x)$. This is the trick of uncomputation as described above. The state of the system is now

$$|x\rangle |0\rangle^{\otimes k} |0\rangle^{\otimes m} |y\rangle |1\rangle \quad (2.69)$$

6. To get $|y\rangle$ in the second register again, we need to apply two sets of k Toffoli gates.

The first set has the qubits in the fourth and fifth register as control qubits and the qubits in the second register as target qubits, that is, they each perform the map $|z_2, z_4, z_5\rangle \mapsto |z_2, z_2 \oplus z_4 z_5, z_5\rangle$ to give

$$|x\rangle |y\rangle |0\rangle^{\otimes m} |y\rangle |1\rangle \quad (2.70)$$

since $|0, y_i, 1\rangle \mapsto |y_i, y_i, 1\rangle$ for all i from 1 to k .

The second set are the same as those used in Step 4, where the qubits in the second and fifth registers are control qubits and the qubits in the fourth register are the target qubits. These

gates erase the bits that make up y from the fourth register, since $|y_i, y_i, 1\rangle \mapsto |y_i, 0, 1\rangle$ for all i from 1 to k .

The state of the system evolves to

$$|x\rangle |y\rangle |0\rangle^{\otimes m} |0\rangle^{\otimes k} |1\rangle \tag{2.71}$$

7. Finally, apply the quantum NOT gate to the fifth register to give

$$|x\rangle |y\rangle |0\rangle^{\otimes m} |0\rangle^{\otimes k} |0\rangle \tag{2.72}$$

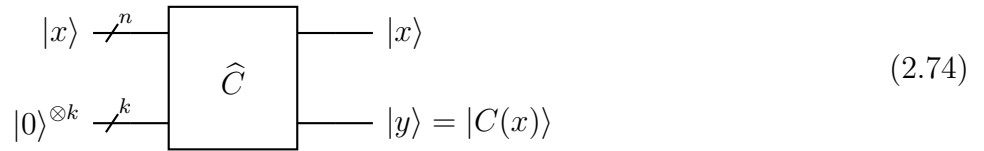
Hence this procedure has mapped

$$|x\rangle |0\rangle^{\otimes k} |0\rangle^{\otimes m} |0\rangle^{\otimes k} |0\rangle \mapsto |x\rangle |C(x)\rangle |0\rangle^{\otimes m} |0\rangle^{\otimes k} |0\rangle \tag{2.73}$$

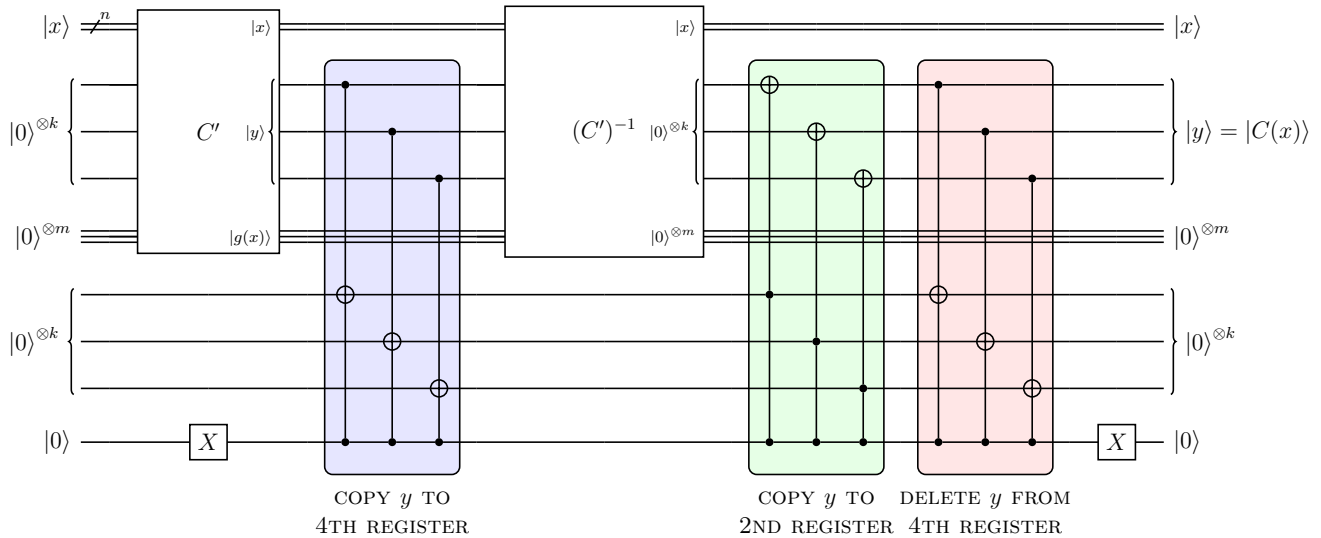
using only Toffoli and NOT gates.

Dropping the last three registers of ancilla qubits gives the final reversible circuit \widehat{C} corresponding to the classical circuit C which maps $|x\rangle |0\rangle^{\otimes k} \mapsto |x\rangle |C(x)\rangle$, as claimed.

The quantum circuit for this operator, in its succinct form, is given by



or, in greater detail, by



where we have emphasised the copying and deleting operations that ultimately move $|y\rangle$ into the second register by highlighting them. Note that in moving $|y\rangle$ we are really copying bits, hence we have not disobeyed the result of Woiters and Zurek [27] stated above.

Remark 2.44. The procedure to construct \widehat{C} used five registers since we wanted to construct the circuit using only Toffoli and NOT gates.

It is worth noting that we could construct the same circuit using only four registers if we also allowed ourselves to use CNOT gates, by changing the purple, green and red k -sets of Toffoli gates given in Steps 4 and 6 into their CNOT counterparts and deleting the fifth register from the quantum circuit above.

Lemma 2.45. We claim that we can construct a quantum circuit that “evaluates an oracle” U_f defined by

$$U_f : |x\rangle |y\rangle \mapsto |x\rangle |y \oplus f(x)\rangle \quad (2.75)$$

where \oplus is addition modulo 2, and f is assumed to be a Boolean function that maps n bits to k bits.

Proof. Note that we can apply a similar quantum gate to C' defined in Equation (2.64) which maps $|x\rangle |0\rangle^{\otimes k} |0\rangle^{\otimes m} \mapsto |x\rangle |f(x)\rangle |g(x)\rangle$, where $|g(x)\rangle$ is garbage in the output state. Denote this reversible gate by U' .

The following procedure implements the operator U_f .

1. Start with four registers in the state

$$|x\rangle |0\rangle^{\otimes k} |0\rangle^{\otimes m} |y\rangle \quad (2.76)$$

where the second and third registers contain ancilla qubits.

2. Apply U' defined above to the first three registers to get

$$|x\rangle |f(x)\rangle |g(x)\rangle |y\rangle \quad (2.77)$$

3. Noting Remark 2.44, apply k CNOT gates to the qubits of the second and fourth registers, where the qubits in the second register are the control qubits and the qubits in the fourth are the target qubits

This gives

$$|x\rangle |f(x)\rangle |g(x)\rangle |y \oplus f(x)\rangle \quad (2.78)$$

4. Apply $(U')^\dagger$ to the first three registers to perform the trick of uncomputation, giving the state

$$|x\rangle |0\rangle^{\otimes k} |0\rangle^{\otimes m} |y \oplus f(x)\rangle \quad (2.79)$$

5. Finally drop the second and third registers.

This procedure implements $U_f : |x\rangle |y\rangle \mapsto |x\rangle |y \oplus f(x)\rangle$. □

Remark 2.46. Typically U_f is applied to the state $|x\rangle |0\rangle$, giving $|x\rangle |f(x)\rangle$.

This is especially useful if we have a superposition of the form

$$\sum_{i=0}^{2^n-1} \alpha_i |i\rangle |0\rangle^{\otimes k} \quad (2.80)$$

since if f is a function that maps n bits to k bits, then U_f evolves this state to

$$\sum_{i=0}^{2^n-1} \alpha_i |i\rangle |f(i)\rangle \quad (2.81)$$

which is a superposition of all the values of the function f .

This is often called “quantum parallelism”. It is one of the major reasons why certain quantum algorithms are significantly quicker than their classical counterparts, because we are able to calculate all the function values with only one call to the operator U_f .

3 Applications of the Quantum Fourier Transform

We now come to one of the most important unitary operators in Quantum Computing, the Quantum Fourier Transform. It appears in many quantum algorithms, some of which we will study in this chapter. We start by defining a version of the Quantum Fourier Transform that we will show can be implemented on a quantum computer. However, in the chapters that follow, we will introduce “improved” versions of this operator depending on our needs. Despite this, we will still say “the” Quantum Fourier Transform when referring to the operator as the version to which we are referring should be clear from the context. The material for this chapter is based upon [3] and [21].

3.1 Quantum Fourier Transform, version 1

Definition 3.1 (Quantum Fourier Transform). Let Q be a quantum system over \mathbb{C} of size 2^n . Writing the computational basis of this system as $\{|0\rangle, |1\rangle, \dots, |2^n - 1\rangle\}$, we can define a linear operator F_{2^n} on this orthonormal basis as follows:

$$F_{2^n} : Q \longrightarrow Q$$

$$|j\rangle \mapsto \frac{1}{\sqrt{2^n}} \sum_{k=0}^{2^n-1} \omega_{2^n}^{jk} |k\rangle \quad (3.1)$$

where $\omega_{2^n} := e^{\frac{2\pi i}{2^n}}$. We will often write ω instead of ω_{2^n} for clarity of notation.

Remark 3.2. A useful way to write the operator F_{2^n} is to use the outer product notation involving the computational basis, as follows

$$F_{2^n} = \frac{1}{\sqrt{2^n}} \sum_{x,k=0}^{2^n-1} \omega^{xk} |k\rangle \langle x| \quad (3.2)$$

The equality holds because, for a basis state $|j\rangle$, we have that

$$\left(\frac{1}{\sqrt{2^n}} \sum_{x,k=0}^{2^n-1} \omega^{xk} |k\rangle \langle x| \right) |j\rangle = \frac{1}{\sqrt{2^n}} \sum_{x,k=0}^{2^n-1} \omega^{xk} |k\rangle \langle x|j\rangle = \frac{1}{\sqrt{2^n}} \sum_{x,k=0}^{2^n-1} \omega^{xk} \delta_{xj} |k\rangle = \frac{1}{\sqrt{2^n}} \sum_{k=0}^{2^n-1} \omega^{jk} |k\rangle$$

So far we have merely stated that the Quantum Fourier Transform is unitary without proving it: we address this next.

Exercise 3.3. Give a direct proof that the Quantum Fourier Transform as given by Equation (3.1) is a unitary operator on a quantum system Q of size 2^n .

Proof. We show that $F_{2^n}^\dagger F_{2^n} = I_Q$.

We have that

$$\begin{aligned}
F_{2^n}^\dagger F_{2^n} &= \left(\frac{1}{\sqrt{2^n}} \sum_{l,m=0}^{2^n-1} \omega^{lm} |l\rangle \langle m| \right)^\dagger \left(\frac{1}{\sqrt{2^n}} \sum_{j,k=0}^{2^n-1} \omega^{jk} |k\rangle \langle j| \right) \\
&= \frac{1}{2^n} \sum_{l,m=0}^{2^n-1} \sum_{j,k=0}^{2^n-1} \omega^{-lm} \omega^{jk} |m\rangle \langle l|k\rangle \langle j| \\
&= \frac{1}{2^n} \sum_{l,m=0}^{2^n-1} \sum_{j,k=0}^{2^n-1} \omega^{-lm} \omega^{jk} \delta_{lk} |m\rangle \langle j| \\
&= \frac{1}{2^n} \sum_{m=0}^{2^n-1} \sum_{j,k=0}^{2^n-1} \omega^{(j-m)k} |m\rangle \langle j| \tag{3.3}
\end{aligned}$$

where we have used the antilinearity of the adjoint from Exercise 2.11 and the orthonormality of the computational basis from Theorem 2.5.

We now consider

$$\frac{1}{2^n} \sum_{k=0}^{2^n-1} \omega^{(j-m)k} \tag{3.4}$$

by splitting into cases:

Case 1: $j = m$

Then Equation (3.4) equals

$$\frac{1}{2^n} \sum_{k=0}^{2^n-1} 1 = 1 \tag{3.5}$$

Case 2: $j \neq m$

Then Equation (3.4) is the sum of a geometric series, which equals

$$\frac{1}{2^n} \left[\frac{1 - \omega^{2^n(j-m)}}{1 - \omega^{(j-m)}} \right] = \frac{1}{2^n} \left[\frac{1 - 1}{1 - \omega^{(j-m)}} \right] = 0 \tag{3.6}$$

Therefore Equation (3.3) becomes

$$\frac{1}{2^n} \sum_{j,k=0}^{2^n-1} |j\rangle \langle j| = \sum_{j=0}^{2^n-1} |j\rangle \langle j| = I_Q \tag{3.7}$$

by the Completeness Relation of Example 2.7. Hence F_{2^n} is unitary. \square

We will also be able to show that the Quantum Fourier Transform is a unitary operator by constructing a quantum circuit for it. For that, we need another way to represent it.

Proposition 3.4 (Product Representation). The mapping given by Equation (3.1) can be written in the form

$$|j_1 j_2 \dots j_n\rangle \mapsto \frac{1}{\sqrt{2^n}} (|0\rangle + e^{2\pi i 0 \cdot j_n} |1\rangle) (|0\rangle + e^{2\pi i 0 \cdot j_{n-1} j_n} |1\rangle) \dots (|0\rangle + e^{2\pi i 0 \cdot j_1 j_2 \dots j_n} |1\rangle) \tag{3.8}$$

where the basis state $|j\rangle$ is being expressed in its binary form, that is, as $j = j_1 j_2 \dots j_n$, where $j = j_1 2^{n-1} + j_2 2^{n-2} + \dots + j_n 2^0$.

Proof. Let $\frac{1}{\sqrt{2^n}} \sum_{k=0}^{2^n-1} \omega^{jk} |k\rangle$ be the result of mapping the basis state $|j\rangle$. Then, using a binary representation for each $|k\rangle$, we have that this result equals

$$\begin{aligned}
&= \frac{1}{\sqrt{2^n}} \sum_{k_1=0}^1 \sum_{k_2=0}^1 \cdots \sum_{k_n=0}^1 e^{2\pi i j (\sum_{l=1}^n k_l 2^{-l})} |k_1 k_2 \dots k_n\rangle \\
&= \frac{1}{\sqrt{2^n}} \sum_{k_1=0}^1 \sum_{k_2=0}^1 \cdots \sum_{k_n=0}^1 \bigotimes_{l=1}^n e^{2\pi i j k_l 2^{-l}} |k_l\rangle \\
&= \frac{1}{\sqrt{2^n}} \bigotimes_{l=1}^n \left[\sum_{k_l=0}^1 e^{2\pi i j k_l 2^{-l}} |k_l\rangle \right] \\
&= \frac{1}{\sqrt{2^n}} \bigotimes_{l=1}^n \left[|0\rangle + e^{2\pi i j 2^{-l}} |1\rangle \right] \\
&= \frac{1}{\sqrt{2^n}} (|0\rangle + e^{2\pi i 0 \cdot j_n} |1\rangle) (|0\rangle + e^{2\pi i 0 \cdot j_{n-1} j_n} |1\rangle) \cdots (|0\rangle + e^{2\pi i 0 \cdot j_1 j_2 \dots j_n} |1\rangle)
\end{aligned}$$

where we have used that $\alpha\beta |k_1 k_2\rangle = \alpha |k_1\rangle \otimes \beta |k_2\rangle$ in the second line.

The step to give the final line is not trivial. It comes from firstly considering

$$\begin{aligned}
\frac{j}{2^l} &= \frac{j_1 2^{n-1} + j_2 2^{n-2} + \cdots + j_n 2^0}{2^l} \\
&= j_1 2^{n-1-l} + \cdots + j_{n-l} 2^0 + j_{n-l+1} 2^{-1} + \cdots + j_n 2^{-l} \\
&= \alpha_l + 0 \cdot j_{n-l+1} \cdots j_n
\end{aligned}$$

where $\alpha_l := j_1 2^{n-1-l} + \cdots + j_{n-l} 2^0 \in \mathbb{Z}$.

Then we have that

$$\begin{aligned}
e^{2\pi i j 2^{-l}} &= e^{2\pi i (\alpha_l + 0 \cdot j_{n-l+1} \dots j_n)} \\
&= e^{2\pi i \alpha_l} e^{2\pi i 0 \cdot j_{n-l+1} \dots j_n} \\
&= e^{2\pi i 0 \cdot j_{n-l+1} \dots j_n}
\end{aligned}$$

Applying this for each l gives the result, as required. \square

Exercise 3.5. Compute the Quantum Fourier Transform of the n qubit state $|00 \dots 0\rangle$ in Q (also written as the element $|0\rangle$ of the computational basis of Q .)

Proof. From the definition of F_{2^n} , we have that

$$\begin{aligned}
|00 \dots 0\rangle &\mapsto \frac{1}{\sqrt{2^n}} \sum_{k=0}^{2^n-1} |k\rangle \\
&= \frac{1}{\sqrt{2^n}} (|0\rangle + |1\rangle)^{\otimes n}
\end{aligned} \tag{3.9}$$

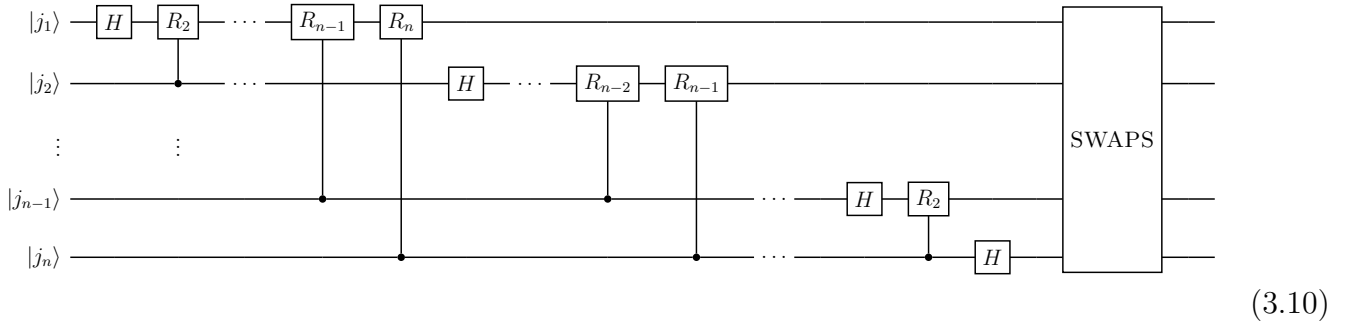
where the $|0\rangle, |1\rangle$ in Equation (3.9) are the elements of the computational basis of a single qubit quantum system.

This form aligns nicely with the Product Representation form given in Equation (3.8). \square

We come to the main result of this Section.

Theorem 3.6 (Implementation of the Quantum Fourier Transform F_{2^n}). When $n = 1$, the Quantum Fourier Transform is the Hadamard gate H .

For $n \in \mathbb{Z}_{>1}$, the quantum circuit for this form of the Quantum Fourier Transform is



where $|j_1 j_2 \dots j_n\rangle$ is a computational basis state $|j\rangle$ of Q given in its binary form, that is, $j_i \in \{0, 1\}$ for all i , and the SWAPS gate swaps qubits 1 and n , 2 and $n - 1$ etc.

Proof. We adapt the proof given in [21].

When $n = 1$, the Quantum Fourier Transform F_2 is the Hadamard gate H , since, by Equation (3.1), F_2 maps $|0\rangle$ to $\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ and $|1\rangle$ to $\frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$, which is exactly the action of H on the computational basis of \mathbb{C}^2 .

To show that the quantum circuit gives the desired implementation for $n \in \mathbb{Z}_{>1}$, we can look at what happens to the state $|j_1\rangle$ (ignoring $|j_2 \dots j_n\rangle$ for now.)

- By applying H first, we obtain the state $\frac{1}{\sqrt{2}}(|0\rangle + e^{2\pi i 0 \cdot j_1} |1\rangle)$, since j_1 is either 0 or 1.
- We then look to apply a controlled- R_2 gate to this state, with the control bit being given by $|j_2\rangle$.
 - If $j_2 = 0$, then R_2 is not applied to the state, and given that $e^{2\pi i 0 \cdot 0 j_2} = 1$ under this condition, we can rewrite our current state as $\frac{1}{\sqrt{2}}(|0\rangle + e^{2\pi i 0 \cdot j_1 j_2} |1\rangle)$.
 - If $j_2 = 1$, then R_2 is applied to the state, and since $R_2 |0\rangle = |0\rangle$ and $R_2 |1\rangle = e^{2\pi i 0 \cdot 0 j_2} |1\rangle$, by linearity we also obtain the state $\frac{1}{\sqrt{2}}(|0\rangle + e^{2\pi i 0 \cdot j_1 j_2} |1\rangle)$.
- Hence we are left in the state $\frac{1}{\sqrt{2}}(|0\rangle + e^{2\pi i 0 \cdot j_1 j_2} |1\rangle)$ after the action of the controlled- R_2 gate in either case.

In a similar fashion, each of the controlled- R_k gates adds an extra bit to the binary fraction part of the phase of $|1\rangle$, resulting in the state $\frac{1}{\sqrt{2}}(|0\rangle + e^{2\pi i 0 \cdot j_1 j_2 \dots j_n} |1\rangle)$ after all the gates have been applied to $|j_1\rangle$.

By applying the same argument to $|j_2\rangle, \dots, |j_n\rangle$, and noting that we do not apply any gates to $|j_{l+1}\rangle$ before we've applied all the appropriate gates to $|j_l\rangle$, we get the state

$$\frac{1}{\sqrt{2^n}} \left((|0\rangle + e^{2\pi i 0 \cdot j_1 j_2 \dots j_n} |1\rangle) \dots (|0\rangle + e^{2\pi i 0 \cdot j_{n-1} j_n} |1\rangle) (|0\rangle + e^{2\pi i 0 \cdot j_n} |1\rangle) \right) \quad (3.11)$$

This looks like the Product Representation of the Quantum Fourier Transform, except the order is reversed. We need to apply swap gates to reverse the order (swap qubits 1 and n , 2 and $n - 1$ etc.), each of which is performed by using three controlled-NOT gates (see Example 2.26).

As a result, we end up with an implementation of the Product Representation of the Quantum Fourier Transform, as given in Equation (3.8). \square

Remark 3.7. The circuit is efficient, according to Definition 2.37, because it uses $n + (n-1) + \dots + 1 = \frac{n(n+1)}{2}$ gates to make the initial computations and then uses at most $\frac{n}{2}$ swaps, each of which can be performed using three controlled-NOT gates. Hence the circuit is a $\Theta(n^2)$ implementation of this version of the Quantum Fourier Transform.

Example 3.8. The 2-qubit Quantum Fourier Transform $F_4 : Q \rightarrow Q$ has a circuit of the form



since $R_2 = S$. We can also give its matrix representation when we choose the computational basis for both Q in the mapping, as follows:

$$\begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & i & -1 & -i \\ 1 & -1 & 1 & -1 \\ 1 & -i & -1 & i \end{pmatrix} \quad (3.13)$$

since the matrix entry in row k , column j when the computational basis is chosen for both Q is ω^{jk} , where we start the row and column indices from 0.

Exercise 3.9. 1. Show that the Inverse Quantum Fourier Transform is given by

$$F_{2^n}^\dagger : |j\rangle \mapsto \frac{1}{\sqrt{2^n}} \sum_{k=0}^{2^n-1} \omega^{-jk} |k\rangle \quad (3.14)$$

2. Give a quantum circuit to perform the Inverse Quantum Fourier Transform

Proof. 1. There are many ways to prove this. We give an approach that uses the adjoint of the outer product representation of F_{2^n} that we have already seen in the proof of Exercise 3.3.

From Equation (3.2) we have that the outer product representation of F_{2^n} is

$$F_{2^n} = \frac{1}{\sqrt{2^n}} \sum_{x,y=0}^{2^n-1} \omega^{xy} |y\rangle \langle x|$$

Applying the adjoint to each side, we get that

$$\begin{aligned} F_{2^n}^\dagger &= \left(\frac{1}{\sqrt{2^n}} \sum_{x,y=0}^{2^n-1} \omega^{xy} |y\rangle \langle x| \right)^\dagger \\ &= \frac{1}{\sqrt{2^n}} \sum_{y,x=0}^{2^n-1} \omega^{-xy} |x\rangle \langle y| \end{aligned} \quad (3.15)$$

Then, for a basis state $|j\rangle$, we have that

$$\left(\frac{1}{\sqrt{2^n}} \sum_{y,x=0}^{2^n-1} \omega^{-xy} |x\rangle \langle y| \right) |j\rangle = \frac{1}{\sqrt{2^n}} \sum_{y,x=0}^{2^n-1} \omega^{-xy} |x\rangle \langle y|j\rangle = \frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} \omega^{-jx} |x\rangle$$

Relabelling the sum over x as a sum over k gives the result.

2. Since $(AB)^\dagger = B^\dagger A^\dagger$, the circuit is the same as the one given in Equation (3.10) but in reverse order, with all the gates replaced by their adjoints. Note that $H^\dagger = H$ and $R_k^\dagger = \begin{pmatrix} 1 & 0 \\ 0 & e^{-\frac{2\pi i}{2^k}} \end{pmatrix}$.

□

3.2 Phase Estimation

We are interested in the Phase Estimation problem because it appears as a key subroutine in many of the quantum algorithms that we'll study. It's a natural problem to consider because the Inverse Quantum Fourier Transform, $F_{2^n}^\dagger$, is used as a gate in a quantum circuit that solves this problem. We can formulate the problem and the goal as follows:

- Problem: We have a unitary operator U that has an eigenvector $|u\rangle$ with eigenvalue $e^{2\pi i\phi}$, but we don't know the phase ϕ .
- Goal: Estimate ϕ .

We note that the problem formulation makes sense as all eigenvalues of a unitary operator have norm 1 by Exercise 2.18.

In order to make use of the quantum circuit that provides a solution to the Phase Estimation problem in other quantum algorithms, we make two promises:

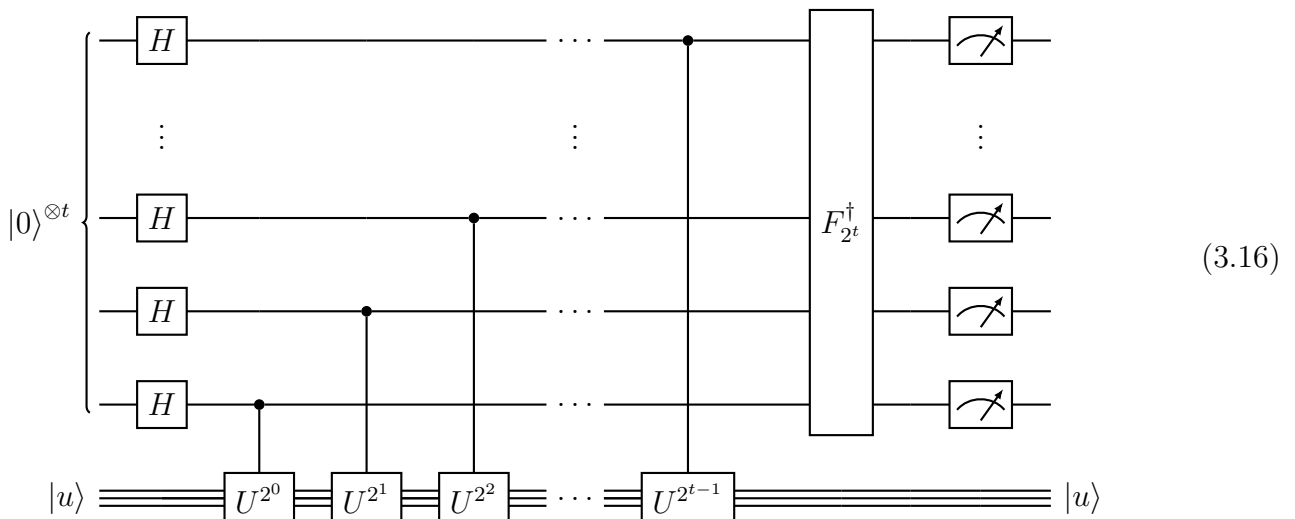
1. we are able to prepare the eigenstate $|u\rangle$ in a L -qubit register - for some L to be chosen by the implementer - as an initial state to be input into the circuit.
2. we can implement "oracles" (also known as "black boxes") which perform the operator U^{2^j} for non-negative integers j .

Neither of these are trivial matters in practice.

But, assuming that this is possible, we can build a circuit that uses two registers: the first is a t -qubit register whose initial state is $|0\rangle^{\otimes t}$, and the second is an L -qubit register whose initial state is $|u\rangle$. We will analyse the choice of t more fully later, but for now, it depends on two things:

- how accurately we wish to estimate ϕ .
- the rate of error that we wish to consider acceptable in applying this procedure.

The quantum circuit that gives the solution to this problem is as follows:



We break down each step of the circuit below:

1. After applying $H^{\otimes t} \otimes I^{\otimes L}$ to $|0\rangle^{\otimes t} |u\rangle$, we obtain the state $\frac{1}{\sqrt{2^t}}(|0\rangle + |1\rangle)^{\otimes t} |u\rangle$.
2. Next we apply the series of controlled- U^{2^j} operations to the second register $|u\rangle$, which leaves it unchanged, but changes the overall state to

$$\left(\frac{1}{\sqrt{2^t}} (|0\rangle + e^{2\pi i 2^{t-1} \phi} |1\rangle) (|0\rangle + e^{2\pi i 2^{t-2} \phi} |1\rangle) \cdots (|0\rangle + e^{2\pi i 2^0 \phi} |1\rangle) \right) |u\rangle \quad (3.17)$$

This is true because, for example:

$$\begin{aligned} (I \otimes U^{2^j})(|1\rangle \otimes |u\rangle) &= I |1\rangle \otimes U^{2^j} |u\rangle \\ &= |1\rangle \otimes e^{2\pi i 2^j \phi} |u\rangle \\ &= e^{2\pi i 2^j \phi} |1\rangle \otimes |u\rangle \end{aligned}$$

3. Equation (3.17) is equivalent to

$$\frac{1}{\sqrt{2^t}} \sum_{k=0}^{2^t-1} e^{2\pi i k \phi} |k\rangle |u\rangle \quad (3.18)$$

by reconstituting the product of the phases as a binary expansion.

4. Now we apply the crucial step $F_{2^t}^\dagger \otimes I^{\otimes L}$ (that is, the Inverse Quantum Fourier Transform to the first register), the result of which we want to be the state

$$|\tilde{\phi}\rangle |u\rangle \quad (3.19)$$

where $\tilde{\phi}$ is a t -bit estimate of ϕ .

5. Measuring the first register in the computational basis gives $\tilde{\phi}$.

Remark 3.10. Clearly, we need to analyse Step 4 more closely in order to be able to assert correctly that the result of applying the Inverse Quantum Fourier Transform to the first register is $|\tilde{\phi}\rangle |u\rangle$.

Remark 3.11. Note that if the phase ϕ can be expressed exactly as a t -bit binary fraction, that is, as $\phi = 0.\phi_1\phi_2 \dots \phi_t$, then Step 4 above gives the state $|\phi_1\phi_2 \dots \phi_t\rangle |u\rangle$.

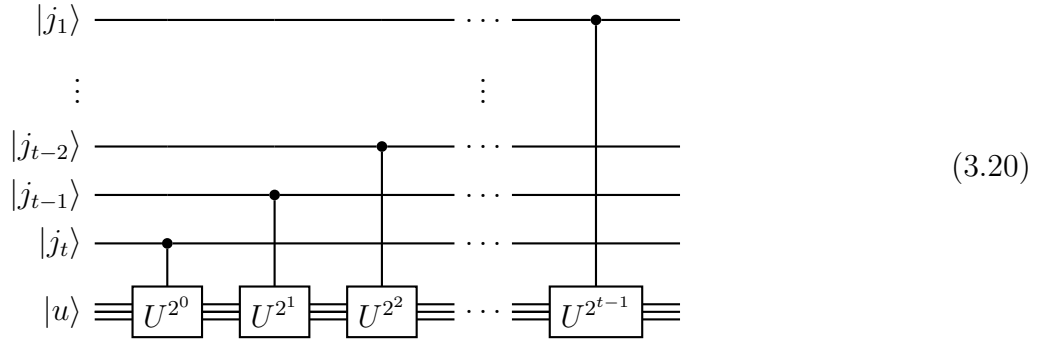
It can be seen that the result of Step 2, given by Equation (3.17), is exactly the same as the result of applying $F_{2^n} \otimes I^{\otimes L}$ to $|\phi_1\phi_2 \dots \phi_t\rangle |u\rangle$, presented in its Product Representation form. Hence applying $F_{2^t}^\dagger \otimes I^{\otimes L}$ to this result (Step 4) returns the original state, since F_{2^n} is unitary.

Therefore, applying Step 5 in this case gives us ϕ exactly.

We can also provide a less explicit implementation of Steps 1 and 2 above by reconsidering what it means to “apply the series of controlled- U^{2^j} operations to the second register $|u\rangle$.”

Exercise 3.12. Show that the effect of the sequence of controlled- U^{2^j} operations applied to the second register $|u\rangle$ is to take the state $|j\rangle |u\rangle \mapsto |j\rangle U^j |u\rangle$.

Proof. Writing $|j\rangle$ as $|j_1 j_2 \dots j_t\rangle$, where $j_k \in \{0, 1\}$ for each k , and considering the quantum circuit below



we have that

$$\begin{aligned} |j\rangle |u\rangle &\mapsto |j_1 j_2 \dots j_t\rangle U^{2^{t-1} j_1} U^{2^{t-2} j_2} \dots U^{2^0 j_t} |u\rangle \\ &= |j_1 j_2 \dots j_t\rangle U^{2^{t-1} j_1 + 2^{t-2} j_2 + \dots + 2^0 j_t} |u\rangle \\ &= |j\rangle U^j |u\rangle \end{aligned}$$

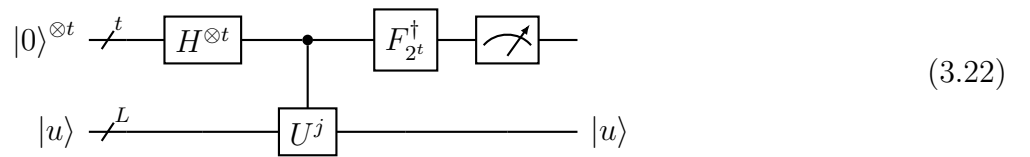
as required. \square

Since we can rewrite the result of Step 1 as

$$\frac{1}{\sqrt{2^t}} \sum_{j=0}^{2^t-1} |j\rangle |u\rangle \quad (3.21)$$

we can apply U^j to the second register $|u\rangle$ for each j in the sum and obtain Equation (3.18) immediately.

This reformulation allows us to present a more compact version of the quantum circuit for Phase Estimation that was given in Equation (3.16):



We are left with 3 major questions:

1. What do we actually get by applying the Inverse Quantum Fourier Transform in Step 4 above, and if it is $|\tilde{\phi}\rangle |u\rangle$, how good an estimate is $\tilde{\phi}$ of ϕ ?
2. How can we choose t such that the algorithm fails with probability less than some arbitrary ϵ ?
3. In which algorithms can we fulfil the two promises that we've made in order to make use of a fully-implemented version of this procedure?

We can answer questions 1. and 2. with the following analysis. We follow the argument given in [21], but we have changed some of the calculations and the order in which the original analysis was presented, adding many more details, in order to be clearer in our reasoning.

By Remark 3.11, we can assume that ϕ has more than t -bits when expressed as a binary fraction, that is, that

$$\phi = 0.\phi_1\phi_2\dots\phi_t + \delta \quad (3.23)$$

where $0 < \delta < \frac{1}{2^t}$. Defining $b := \phi_1 2^{t-1} + \phi_2 2^{t-2} + \dots + \phi_t 2^0$ (and so $b \in \mathbb{Z}$ such that $b \in [0, 2^t - 1]$), we have that

$$\phi = \frac{b}{2^t} + \delta \quad (3.24)$$

Hence b is the largest integer such that $\frac{b}{2^t}$ is less than ϕ .

Beginning at Equation (3.18), we can see that the true result of Step 4, that is, applying $F_{2^n}^\dagger$ to the first register, is

$$\frac{1}{2^t} \sum_{l=0}^{2^t-1} \sum_{k=0}^{2^t-1} e^{2\pi i k \phi} e^{\frac{-2\pi i k l}{2^t}} |l\rangle |u\rangle \quad (3.25)$$

Now define $\beta_l := \frac{1}{2^t} \sum_{k=0}^{2^t-1} e^{2\pi i (\phi - \frac{l}{2^t}) k}$. Note that β_l is periodic in 2^t , that is, $\beta_{l+2^t} = \beta_l$.

We can therefore rewrite Equation (3.25) as

$$\sum_{l=0}^{2^t-1} \beta_l |l\rangle |u\rangle \quad (3.26)$$

This is the true result of applying Step 4 in the procedure above.

Step 5 of the procedure states that the first register of Equation (3.26) is measured in the computational basis. From this measurement, we will obtain some basis state $|l\rangle$ with probability $|\beta_l|^2$.

Claim 3.13. But we want to obtain a state $|l\rangle$ such that it satisfies $|l - b| \leq e$, that is, l is some tolerance for error e close to the integer b , and, furthermore, choose e such that $e < 2^{t-n} - 1$ for an arbitrarily chosen n , because we will then be able to approximate ϕ to an accuracy of n bits.

Proof. Assume that we measure the first register of Equation (3.26) to obtain a basis state $|l\rangle$ such that $|l - b| \leq e$. Then we have that

$$|l - b| \leq e \iff \delta - 2^{-t}e \leq \phi - 2^{-t}l \leq \delta + 2^{-t}e \quad (3.27)$$

which we get by multiplying through by -2^{-t} , adding ϕ , and applying the definition of δ as given by Equation (3.24).

This is important because the error in the approximation of ϕ under this measurement is $\Delta := \phi - 2^{-t}l$, and Equation (3.27) says we can bound this error from above:

$$\Delta \leq \delta + 2^{-t}e < 2^{-t} + 2^{-t}e \quad (3.28)$$

since $0 < \delta < \frac{1}{2^t}$.

By choosing a tolerance for error e such that $e < 2^{t-n} - 1$, we get that $\Delta < 2^{-n}$, and so we can approximate ϕ to an accuracy of n bits for an arbitrarily chosen n . \square

Therefore, we need to understand what is the probability of measuring the first register such that $|l\rangle$ satisfies $|l - b| \leq e$, where $e < 2^{t-n} - 1$ for an arbitrarily chosen n .

We look instead at the probability of $|l - b| > e$ and find a max bound as a function of e for it.

Note that we can restrict $l \bmod 2^t$ by the periodicity of β_l .

Recalling that $b \in \mathbb{Z}$ such that $b \in [0, 2^t - 1]$, we can define $l' := l - b$, and consider it mod 2^t .

Then the range of l' can be $-2^{t-1} + 1 \leq l' \leq 2^{t-1}$, since the two sets $\{0, 1, \dots, 2^t - 1\}$ and $\{-2^{t-1} + 1, -2^{t-1} + 2, \dots, 0, 1, \dots, 2^{t-1}\}$ are precisely the same modulo 2^t , and both $-e$ and $+e$ lie in the range of l' .

Hence, we can say that

$$P(|l - b| > e) = \sum_{l'=-2^{t-1}+1}^{-e-1} |\beta_{l'+b}|^2 + \sum_{l'=e+1}^{2^{t-1}} |\beta_{l'+b}|^2 \quad (3.29)$$

By the definition of β_l , we have that

$$\beta_{l'+b} = \frac{1}{2^t} \sum_{k=0}^{2^t-1} e^{2\pi i \left(\phi - \frac{l'+b}{2^t}\right)k} \quad (3.30)$$

Setting $r := e^{2\pi i \left(\phi - \frac{l'+b}{2^t}\right)}$, we can see that equation (3.30) is a geometric series, with result

$$\begin{aligned} \beta_{l'+b} &= \frac{1}{2^t} \frac{1 - r^{2^t}}{1 - r} \\ &= \frac{1}{2^t} \frac{1 - e^{2\pi i (2^t \phi - (l'+b))}}{1 - e^{2\pi i \left(\phi - \frac{l'+b}{2^t}\right)}} \\ &= \frac{1}{2^t} \frac{1 - e^{2\pi i (2^t \delta - l')}}{1 - e^{2\pi i \left(\delta - \frac{l'}{2^t}\right)}} \end{aligned}$$

with the final step coming from substituting in Equation (3.24).

Since $|1 - e^{ix}| \leq |1| + |-e^{ix}| = 2$ for any $x \in \mathbb{R}$ by the Cauchy-Schwarz inequality, we get that

$$|\beta_{l'+b}| \leq \frac{1}{2^t} \frac{2}{\left|1 - e^{2\pi i \left(\delta - \frac{l'}{2^t}\right)}\right|} \quad (3.31)$$

Next, we show that $-\pi \leq 2\pi \left(\delta - \frac{l'}{2^t}\right) \leq \pi$. We had that

$$\begin{aligned} -2^{t-1} + 1 &\leq l' \leq 2^{t-1} \\ \Rightarrow \frac{-1}{2} + \frac{1}{2^t} &\leq \frac{l'}{2^t} \leq \frac{1}{2} \\ \Rightarrow \frac{-1}{2} &\leq \frac{-l'}{2^t} \leq \frac{1}{2} - \frac{1}{2^t} \\ \Rightarrow \frac{-1}{2} &\leq \delta - \frac{l'}{2^t} \leq \frac{1}{2} \end{aligned}$$

where the last line comes from $0 < \delta < \frac{1}{2^t}$. Multiplying through by 2π gives the result.

The following Lemma helps to continue the proof.

Lemma 3.14. We have that

$$|1 - e^{i\theta}| \geq \frac{2|\theta|}{\pi} \quad (3.32)$$

for all $-\pi \leq \theta \leq \pi$.

Proof. We won't show all the steps, but we note that $|1 - e^{i\theta}| = 2|\sin(\frac{\theta}{2})|$, and so it is enough to show that $|\sin(\frac{\theta}{2})| \geq \frac{|\theta|}{\pi}$ for all $-\pi \leq \theta \leq \pi$.

Since these functions are even on this range, it is sufficient to show that $\sin(\frac{\theta}{2}) \geq \frac{\theta}{\pi}$ for all $0 \leq \theta \leq \pi$, which can be done by proving that $\int_0^\pi \sin(\frac{\theta}{2}) - \frac{\theta}{\pi} d\theta \geq 0$. \square

Applying this to Equation (3.31), we get that

$$|\beta_{l'+b}| \leq \frac{1}{2^t} \frac{2}{\frac{2}{\pi} |2\pi(\delta - \frac{l'}{2^t})|} = \frac{1}{2} \frac{1}{|2^t\delta - l'|} \quad (3.33)$$

Hence, ultimately, substituting this into Equation (3.29), we get that

$$P(|l - b| > e) \leq \frac{1}{4} \left[\sum_{l'=-2^{t-1}+1}^{-e-1} \frac{1}{(l' - 2^t\delta)^2} + \sum_{l'=e+1}^{2^{t-1}} \frac{1}{(l' - 2^t\delta)^2} \right]$$

Since $0 \leq 2^t\delta \leq 1$, the right hand side is

$$\begin{aligned} &\leq \frac{1}{4} \left[\sum_{l'=-2^{t-1}+1}^{-e-1} \frac{1}{(l')^2} + \sum_{l'=e+1}^{2^{t-1}} \frac{1}{(l' - 1)^2} \right] \\ &\leq \frac{1}{4} \left[\sum_{l'=e+1}^{2^{t-1}-1} \frac{1}{(l')^2} + \sum_{l'=e}^{2^{t-1}-1} \frac{1}{(l')^2} \right] \\ &\leq \frac{1}{2} \sum_{l'=e}^{2^{t-1}-1} \frac{1}{(l')^2} \\ &\leq \frac{1}{2} \int_{e-1}^{2^{t-1}-1} \frac{1}{(l')^2} dl' \\ &\leq \frac{1}{2(e-1)} \end{aligned} \quad (3.34)$$

We can use this result to fully understand the probability that the $|l\rangle$ we measure in Step 5 satisfies $|l - b| \leq e$, where $e < 2^{t-n} - 1$ for an arbitrarily chosen n .

Claim 3.15. We can bound this probability $\geq 1 - \epsilon$ for some ϵ if we choose $t := n + \lceil \log_2(2 + \frac{1}{2\epsilon}) \rceil$.

Proof. Our proof of Claim 3.13 showed that if $e < 2^{t-n} - 1$, then $|l - b| \leq e$. Hence we must have that if $|l - b| > e$, then $e \geq 2^{t-n} - 1$.

Applying this to Equation (3.34), we can see that

$$P(|l - b| > e) \leq \frac{1}{2(e-1)} \leq \frac{1}{2(2^{t-n} - 2)} \quad (3.35)$$

In order that $P(|l - b| > e) < \epsilon$, we need $\frac{1}{2(2^{t-n} - 2)} < \epsilon$, which holds if and only if $t > n + \log_2(2 + \frac{1}{2\epsilon})$.

Hence $P(|l - b| \leq e) \geq 1 - \epsilon$ if $t := n + \lceil \log_2(2 + \frac{1}{2\epsilon}) \rceil$. \square

We can combine Claim 3.13 and Claim 3.15 into the following theorem.

Theorem 3.16 (Phase Estimation). Given a unitary operator U with eigenvector $|u\rangle$ and eigenvalue $e^{2\pi i\phi}$, we can approximate the phase ϕ accurate to n bits, which we denote by $\tilde{\phi}$, with success at least $1-\epsilon$ if we apply the quantum algorithm given by the quantum circuit (3.16) with $t := n + \lceil \log_2(2 + \frac{1}{2\epsilon}) \rceil$.

Remark 3.17. It is trivial to extend Theorem 3.16 to superpositions of eigenvectors of U : suppose that $|\psi\rangle := \sum_u c_u |u\rangle$, where each eigenvector has eigenvalue $e^{2\pi i\phi_u}$, and the Phase Estimation procedure takes as input the state $|0\rangle^{\otimes t} |\psi\rangle$ with t chosen as in Theorem 3.16. Then the probability of measuring ϕ_u accurate to n bits is at least $|c_u|^2(1 - \epsilon)$.

Exercise 3.18. Let U be a unitary transform with eigenvalues ± 1 , which acts on a state $|\psi\rangle$. Using the Phase Estimation procedure, construct a quantum circuit to collapse $|\psi\rangle$ into one or the other of the two eigenspaces of U , giving also a classical indicator as to which space the final state is in.

Proof. Let the eigenstates of U be $|u_1\rangle$ and $|u_{-1}\rangle$ with their (obvious) eigenvalues. Then we can write $|\psi\rangle$ as a superposition of these eigenstates, namely

$$|\psi\rangle = c_1 |u_1\rangle + c_{-1} |u_{-1}\rangle \quad (3.36)$$

where $|c_1|^2 + |c_{-1}|^2 = 1$.

Setting $t := 1$, and applying Steps 1 and 2 of the Phase Estimation procedure to the state $|0\rangle |\psi\rangle$, we obtain the state

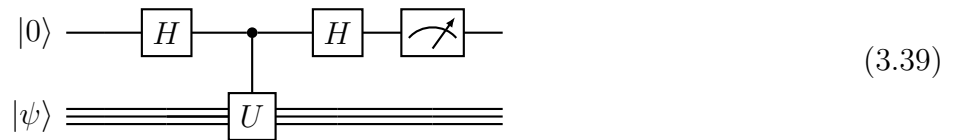
$$\frac{c_1}{\sqrt{2}} [|0\rangle + |1\rangle] |u_1\rangle + \frac{c_{-1}}{\sqrt{2}} [|0\rangle - |1\rangle] |u_{-1}\rangle \quad (3.37)$$

Since $F_2^\dagger = H$, we can apply H to the first register to obtain

$$c_1 |0\rangle |u_1\rangle + c_{-1} |1\rangle |u_{-1}\rangle \quad (3.38)$$

Measuring the first register in the computational basis of \mathbb{C}^2 results in being in the eigenspace with eigenvalue $+1$ with probability $|c_1|^2$ and being in the eigenspace with eigenvalue -1 with probability $|c_{-1}|^2$.

The quantum circuit for this procedure looks as follows:



where we have substituted in for the quantum circuit given in (3.16). □

The remaining major question that we are left with is the third question, which said “in which algorithms can we fulfil the two promises that we’ve made in order to make use of a fully-implemented version of this procedure?” We take a look at our first application in the next section.

3.3 Order-Finding Algorithm

We can formulate the problem and the goal as follows:

- Problem: let $x, N \in \mathbb{Z}_{\geq 0}$ such that $x < N$ and $\gcd(x, N) = 1$.
- Goal: find the smallest positive integer r such that $x^r \equiv 1 \pmod{N}$. r is said to be the order of x modulo N .

Our first result is rather simple:

Exercise 3.19. Show that the order of x satisfies $r \leq N$.

Proof. Since $\gcd(x, N) = 1$, we have that x, N are coprime.

As $x \leq N$, we have that $x \in (\mathbb{Z}_N^*, \times)$, the multiplicative group of integers modulo N which are coprime to N .

The order of this group is $\phi(N)$, where ϕ denotes Euler's totient function, which is defined to be the number of positive integers strictly less than N which are coprime to N .

As the order of x in \mathbb{Z}_N^* is r , we have that $r \mid \phi(N)$ by Lagrange's Theorem, and so $r \leq \phi(N)$.

But $\phi(N) \leq N$ by definition of ϕ , and so $r \leq N$ as required. \square

In order to use the Phase Estimation procedure as part of a quantum algorithm to solve this problem, we need to do the following:

1. reformulate the problem in terms of a unitary operator U and look at its eigenvectors and eigenvalues.
 - For this, it helps to consider N in its binary form, and so we define $L := \lceil \log_2(N) \rceil$ as the number of bits needed to express N in binary.
2. satisfy the two promises, which were
 - to prepare an eigenstate $|u\rangle$ of U in an L -qubit register to be input into the circuit.
 - to provide an implementation of the oracles which perform the operator U^{2^j} for non-negative integers j .

We look at 1. first with the following definition:

Definition 3.20. Let Q be a quantum system over \mathbb{C} of size 2^L . Writing the computational basis of this system as $|0\rangle, |1\rangle, \dots, |2^L - 1\rangle$, we can define a linear operator U on this basis as follows:

$$U : Q \longrightarrow Q$$

$$|y\rangle \mapsto \begin{cases} |xy \bmod N\rangle, & \text{for } 0 \leq y \leq N - 1 \\ |y\rangle, & \text{for } N \leq y \leq 2^L - 1 \end{cases} \quad (3.40)$$

Exercise 3.21. Show that U is a unitary operator.

Proof. We show that U permutes the computational basis. Since $U|y\rangle = |y\rangle$ for $N \leq y \leq 2^L - 1$, it is enough to show that the two sets, $\{|0\rangle, |1\rangle, \dots, |N-1\rangle\}$ and $\{U|0\rangle, U|1\rangle, \dots, U|N-1\rangle\}$ are equivalent. It's clear that if $0 \leq y \leq N-1$, then $U|y\rangle = |xy \bmod N\rangle$ is equivalent to some $|z\rangle$, $0 \leq z \leq N-1$. We therefore need to show that if $U|y\rangle = U|w\rangle$ for $0 \leq y \leq w \leq N-1$, then $y = w$.

Indeed, if $U|y\rangle = U|w\rangle$ for $0 \leq y \leq w \leq N-1$, then $xy \bmod N = xw \bmod N$. Since x has an inverse in (\mathbb{Z}_N^*, \times) , we get that $y \bmod N = w \bmod N$. But $0 \leq y \leq w \leq N-1$, and so $y = w$ as required. \square

We complete 1. by investigating the following claim:

Claim 3.22. The following states are eigenvectors of U with eigenvalues $e^{\frac{2\pi is}{r}}$ for $0 \leq s \leq r-1$.

$$|u_s\rangle := \frac{1}{\sqrt{r}} \sum_{k=0}^{r-1} e^{\frac{-2\pi isk}{r}} |x^k \bmod N\rangle \quad (3.41)$$

Proof. Indeed, we have that

$$U|u_s\rangle = \frac{1}{\sqrt{r}} \sum_{k=0}^{r-1} e^{\frac{-2\pi isk}{r}} |x^{k+1} \bmod N\rangle = e^{\frac{2\pi is}{r}} |u_s\rangle$$

using that when $k = r-1$, $x^{k+1} \equiv 1 \bmod N$. \square

We look at 2. next. We need to fulfil the two promises of the Phase Estimation procedure in order to use a fully implemented version of it in the quantum algorithm for Order-Finding. We start by looking at the first promise.

We are unable to use a single eigenstate $|u_s\rangle$ for some s directly because that would mean that we would have to know what the value of r is. But we can prove the following two exercises, which will lead to showing that $|1\rangle$ is a superposition of the $|u_s\rangle$, and we can prepare this in the second register as input into the Phase Estimation procedure. Note that $|1\rangle$, when using the L -bit binary representation of the computational basis, is really $|00\dots 01\rangle$, and so we can easily prepare this state by applying the operator $I^{\otimes(L-1)} \otimes X$ to $|0\rangle^{\otimes L}$. We already know that $|0\rangle^{\otimes L}$ is a state that can always be prepared for input into a quantum circuit by the model for Quantum Computing given in Theorem 2.32.

Exercise 3.23. Prove that $\sum_{s=0}^{r-1} e^{\frac{-2\pi isk}{r}} = r\delta_{k0}$

Proof. If $k = 0 \bmod r$ then

$$\sum_{s=0}^{r-1} e^{\frac{-2\pi isk}{r}} = \sum_{s=0}^{r-1} 1 = r$$

Else, we have a geometric series, and

$$\sum_{s=0}^{r-1} e^{\frac{-2\pi isk}{r}} = \frac{1 - e^{-2\pi ik}}{1 - e^{\frac{-2\pi ik}{r}}} = 0$$

as required. \square

Exercise 3.24. Prove that

$$\frac{1}{\sqrt{r}} \sum_{s=0}^{r-1} e^{\frac{2\pi isk}{r}} |u_s\rangle = |x^k \bmod N\rangle \quad (3.42)$$

Proof. Substituting in the definition of $|u_s\rangle$ and applying Exercise 3.23, we have that

$$\begin{aligned} \frac{1}{\sqrt{r}} \sum_{s=0}^{r-1} e^{\frac{2\pi i s k}{r}} |u_s\rangle &= \frac{1}{r} \sum_{s=0}^{r-1} \sum_{t=0}^{r-1} e^{\frac{-2\pi i s(t-k)}{r}} |x^t \bmod N\rangle \\ &= \frac{1}{r} \sum_{t=0}^{r-1} r \delta_{(t-k)0} |x^t \bmod N\rangle \\ &= |x^k \bmod N\rangle \end{aligned}$$

as required. \square

Corollary 3.25. Setting $k = 0$ in Equation (3.42) gives

$$\frac{1}{\sqrt{r}} \sum_{s=0}^{r-1} |u_s\rangle = |1\rangle \quad (3.43)$$

and so $|1\rangle$ is a uniform superposition of eigenstates of U .

The second promise asks us to provide an implementation of the oracles which perform the operator U^{2^j} for non-negative integers j . As Exercise 3.12 shows, it is enough to provide an implementation of the transformation $|j\rangle |y\rangle \mapsto |j\rangle U^j |y\rangle = |j\rangle |x^j y \bmod N\rangle$, where $j \in \{0, 1, \dots, 2^t - 1\}$ (where t is the same t that is used to define the number of qubits in the first register of the Phase Estimation procedure, the choice of which is yet to be made explicit).

We can construct a series of reversible circuits that perform these operators, using a third register of L ancilla qubits in the state $|0\rangle^{\otimes L}$, as follows:

$$\begin{aligned} |j\rangle |y\rangle |0\rangle^{\otimes L} &\xrightarrow{\widehat{C}} |j\rangle |y\rangle |x^j \bmod N\rangle \\ &\xrightarrow{M} |j\rangle |x^j y \bmod N\rangle |x^j \bmod N\rangle \\ &\xrightarrow{\widehat{C}^{-1}} |j\rangle |x^j y \bmod N\rangle |0\rangle^{\otimes L} \end{aligned}$$

with implementations of \widehat{C} and M to follow. \widehat{C}^{-1} reverses the calculation of the modular exponential using the trick of uncomputation.

The classical algorithm for computing $x^j \bmod N$, the modular exponential, has two steps.

- Since j can be expressed in binary as $j_{t-1}j_{t-2}\dots j_0$ with $j_i \in \{0, 1\}$ for all i , we need to calculate $x^{2^i} \bmod N$ for all such i . This can be done by starting from $x \bmod N$ and repeatedly squaring, getting $x^2 \bmod N$, $x^{2^2} \bmod N$, etc
- Then we obtain $x^j \bmod N$ by performing the following calculation with these stored modular exponentials

$$x^j \bmod N = \prod_{i=0}^{t-1} \left(x^{j_i 2^i} \bmod N \right) \quad (3.44)$$

which is at most $t - 1$ multiplications (as some of the j_i may be 0).

From Section 2.6, we know that any classical algorithm can be implemented reversibly on a quantum computer, and so we denote its reversible implementation as \widehat{C} . Similarly, the classical algorithm for

multiplying two numbers together also has a reversible implementation on a quantum computer by the same logic, and so we denote this by M .

With this, we have fulfilled both promises, and so we can now use the Phase Estimation procedure in a quantum algorithm for Order-Finding.

In order to use this procedure successfully, we need to make a choice of $t = n + \lceil \log_2(2 + \frac{1}{2\epsilon}) \rceil$ that will provide a “good enough” estimate of the phase. Since we are preparing $|1\rangle$, a uniform superposition of eigenstates $|u_s\rangle$, in the second register, when we apply the Phase Estimation procedure, upon measuring the first register we will obtain an estimate $\tilde{\phi}_s$ of the phase $\frac{s}{r}$ accurate to n bits for some $0 \leq s \leq r-1$, with probability at least $\frac{1}{r}(1-\epsilon)$ and with s unknown. The question therefore becomes “what should we choose n to be so that we can find r ?”

We can answer that question by applying the next theorem, which uses the Continued Fractions Algorithm to compute $\frac{s}{r}$ (with one slight technical proviso) from our estimate $\tilde{\phi}_s$ under certain conditions.

But first, a definition:

Definition 3.26 (Continued Fraction). A finite simple continued fraction is defined by a finite collection a_0, \dots, a_N of positive integers

$$[a_0, \dots, a_N] := a_0 + \frac{1}{a_1 + \frac{1}{a_2 + \frac{1}{\dots + \frac{1}{a_N}}}} \quad (3.45)$$

The n^{th} convergent for some $0 \leq n \leq N$ of this continued fraction is defined to be $[a_0, \dots, a_n]$.

Theorem 3.27. Suppose $\frac{s}{r}$ is a rational number, with s and r both L bit integers, such that

$$\left| \frac{s}{r} - \tilde{\phi}_s \right| \leq \frac{1}{2r^2} \quad (3.46)$$

Then $\frac{s}{r}$ is a convergent of the continued fraction for $\tilde{\phi}_s$, and it can be computed in $O(L^3)$ operations using the Continued Fractions Algorithm.

Proof. See [21, p. 637] for the details. □

Remark 3.28. In order to make use of Theorem 3.27, we need to choose n such that the bound given by Equation (3.46) holds.

Since we know that $r \leq N \leq 2^L$ from Exercise 3.19, we have that $\frac{1}{2^{2L+1}} \leq \frac{1}{2r^2}$.

Hence picking $n = 2L + 1$ means that the result of the Phase Estimation procedure, $\tilde{\phi}_s$, will be an approximation of $\frac{s}{r}$ accurate to $2L + 1$ bits, and so Theorem 3.27 will hold for this n .

We can therefore apply the Continued Fractions Algorithm and will obtain the value of $\frac{s}{r}$ as a convergent in this algorithm.

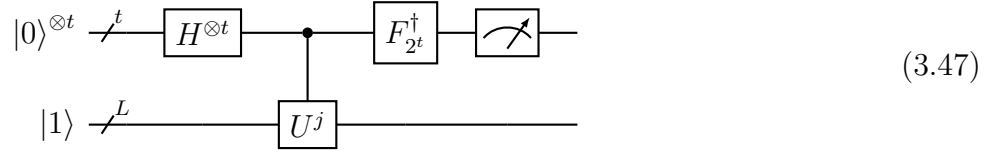
Remark 3.29. The technical proviso mentioned above is that the Continued Fractions Algorithm will give the value of $\frac{s}{r}$, that is, it will return a fraction $\frac{s_1}{r_1} = \frac{s}{r}$ such that $\gcd(s_1, r_1) = 1$.

So we now know s_1 and r_1 . This could be a problem as r_1 may or may not be equal to r .

We can test this by calculating $x^{r_1} \bmod N$.

- If it equals 1 mod N , then we know that $r_1 = r$ and so it is the order of x mod N , as required.
- If it is not equal to 1 mod N , then we know that $r_1 \neq r$. We address how to proceed when we analyse what can be done when the algorithm fails.

The quantum circuit for the Order-Finding procedure is as follows:



When does the algorithm fail?

The obvious situation for when the Order-Finding Algorithm fails is when the Phase Estimation procedure does not produce an estimate of some $\frac{s}{r}$ accurate to $2L + 1$ bits. However, this happens with probability at most $1 - \sum_{s=0}^{r-1} \frac{1}{r} (1 - \epsilon) = \epsilon$, and this can be made negligibly small by increasing the value of t chosen in the Phase Estimation procedure.

The other case, which we've already alluded to, occurs when the Continued Fractions Algorithm returns $\frac{s_1}{r_1}$ with $r_1 \neq r$. We can ignore the situation where we obtain $s_1 = 0$, as it occurs with probability $\frac{1}{r}$, and by repeating the procedure a few times we will obtain an estimate $\tilde{\phi}_s$ of some other $\frac{s}{r}$ with $s \neq 0$.

Ignoring this situation, we therefore get that r_1 is a factor of r . We can then run the procedure again to try to find the order of $x^{r_1} \bmod N$, which we now know will have order $\frac{r}{r_1}$. From the algorithm, we will obtain some r_2 . If we get that $x^{r_1 r_2} \equiv 1 \bmod N$, then we know that $r_1 r_2 = r$ is the order of x , else r_2 is a factor of $\frac{r}{r_1}$, and so we can repeat the procedure to find the order of $x^{r_1 r_2} \bmod N$. We can continue in this way until we find the order of r , which will take at most $O(\log_2 r) = O(L)$ iterations, as we obtain a proper factor of r each time we run the procedure (excluding a constant number of $s = 0$ cases).

See [21, p. 231] for another method to obtain r using $O(1)$ repetitions of the procedure.

What is the runtime for this algorithm?

Given that $t = 2L + 1 + \lceil \log_2(2 + \frac{1}{2\epsilon}) \rceil = O(L)$, we can break down the resources used at each step of the algorithm, as follows:

- Applying the operator $H^{\otimes t}$ to the first register uses $t = O(L)$ gates.
- The modular exponentiation operation was split into two parts:
 - calculating all the $x^{2^i} \bmod N$ from 0 to $t - 1$ requires $t - 1 = O(L)$ squaring operations each at a cost of $O(L^2)$, so $O(L^3)$ in total.
 - calculating $x^j \bmod N$ requires at most $t - 1 = O(L)$ multiplications each at a cost of $O(L^2)$, so $O(L^3)$ in total.

Hence the modular exponentiation operation is $O(L^3)$.

- The Inverse Quantum Fourier Transform, $F_{2^t}^\dagger$, uses $O(t^2) = O(L^2)$ gates.
- The application of Theorem 3.27 to find r_1 uses $O(L^3)$ operations.

Hence one iteration of the Order-Finding Algorithm takes $O(L^3)$ operations. As our analysis showed, we need to repeat it $O(L)$ times in order to guarantee finding r , and so we have an $O(L^4)$ algorithm to find the order of $x \bmod N$.

We can immediately improve it to $O(L^3)$ by using the method found in [21, p. 231]. Shor [26], who discovered this algorithm when he studied the application of Order-Finding to factoring integers, the topic of our next section, obtained an $O(N^2 \log(N) \log(\log(N)))$ algorithm by using faster multiplication algorithms to perform the modular exponentiation than the ones we used above.

3.4 Factoring Integers

We begin as usual by formulating the problem and the goal:

- Problem: We have a positive, composite integer N .
- Goal: To find a non-trivial factor of N .

Shor [26] was able to take the Order-Finding Algorithm of the previous section and use it as a subprocedure within his algorithm to factor integers into their prime decomposition. It is easiest to provide the details of his algorithm first before proving each of the steps:

Theorem 3.30 (Shor's Algorithm). The following procedure is a $O((\log N)^3)$ algorithm to factor a positive composite integer N .

1. If $N \equiv 0 \pmod{2}$, then return 2.
2. Else, randomly choose $x \in [2, N - 1]$, and calculate $\gcd(x, N)$ using Euclid's Algorithm. If $\gcd(x, N) \neq 1$, return $\gcd(x, N)$.
3. Else, apply the Order-Finding Algorithm to find the order r of $x \bmod N$.
4. If r is odd or $x^{\frac{r}{2}} \equiv -1 \pmod{N}$, the algorithm has failed, so return to Step 1.
5. Otherwise compute $\gcd(x^{\frac{r}{2}} - 1, N)$ and $\gcd(x^{\frac{r}{2}} + 1, N)$. At least one of these divides N non-trivially.

The only step that requires a quantum algorithm is Step 3, which we know runs in $O((\log N)^3)$ time if we use the improved method of the Order-Finding Algorithm stated above. Step 1 is a simple division test, which is $O(1)$. All of the other steps use Euclid's Algorithm, which, given that it is a classical algorithm, can be replicated on a quantum computer by Section 2.6 if need be, but can be computed classically in $O((\log N)^3)$ [21, p. 629].

To prove Step 5, we need the following Lemma:

Lemma 3.31. Let N be an L -bit composite number, and suppose that y is a non-trivial solution to the equation $y^2 \equiv 1 \pmod{N}$ such that $1 < y < N - 1$. (Then $y \not\equiv \pm 1 \pmod{N}$.)

Then at least one of $\gcd(y - 1, N)$ and $\gcd(y + 1, N)$ is a non-trivial factor of N that can be computed using $O(L^3)$ operations.

Proof. We give the proof that can be found in [21].

As $y^2 \equiv 1 \pmod{N}$, we have that $N \mid y^2 - 1 = (y - 1)(y + 1)$. Hence N must have a factor $d > 1$ such that either $d \mid y - 1$ or $d \mid y + 1$.

However, $1 < y < N - 1$ implies that $y - 1 < y + 1 < N$, and so $d < N$.

Hence either $1 < d \leq \gcd(y - 1, N) < N$ or $1 < d \leq \gcd(y + 1, N) < N$, and by computing $\gcd(y - 1, N)$ and $\gcd(y + 1, N)$ using Euclid's Algorithm, we obtain a non-trivial factor of N using $O(L^3)$ operations. \square

Remark 3.32. Clearly setting $y = x^{\frac{r}{2}}$ with r even and $x^{\frac{r}{2}} \not\equiv -1 \pmod{N}$ in Lemma 3.31 gives Step 5. Note also that $x^{\frac{r}{2}} \not\equiv 1 \pmod{N}$, since r is the order of $x \pmod{N}$.

We now need to understand how often the algorithm fails in Step 4. First we present the following Lemma:

Lemma 3.33. Let p be an odd prime and suppose that 2^d is the largest power of 2 dividing $\phi(p^\alpha)$. Then if r is the order of a randomly chosen element x in $\mathbb{Z}_{p^\alpha}^*$, the probability that $2^d \mid r$ is $\frac{1}{2}$.

Proof. We give the proof that can be found in [21].

We have that the only integers less than p^α which are not coprime to p^α are the multiples of p , and there are $p^{\alpha-1} - 1$ such multiples, hence $\phi(p^\alpha) = (p^\alpha - 1) - (p^{\alpha-1} - 1) = p^{\alpha-1}(p - 1)$. Since p is odd we also get that $\phi(p^\alpha)$ is even, and hence $d \geq 1$.

We use that $\mathbb{Z}_{p^\alpha}^*$ is a cyclic group [21, p. 632], and so $x \equiv g^k \pmod{p^\alpha}$ for some generator g of the group and for some $k \in \{1, 2, \dots, \phi(p^\alpha)\}$. Furthermore, since r is the order of x in this group, we have that $g^{kr} \equiv 1 \pmod{p^\alpha}$.

Now, if k is odd, then $\phi(p^\alpha) \mid kr$ and hence $2^d \mid r$.

But if k is even, then

$$g^{\frac{k\phi(p^\alpha)}{2}} \equiv 1^{\frac{k}{2}} \equiv 1 \pmod{p^\alpha}$$

Hence $r \mid \frac{\phi(p^\alpha)}{2}$ and so 2^d does not divide r (if it did then 2^{d+1} would divide $\phi(p^\alpha)$, which is a contradiction by the construction of d .)

The probability that k is odd is exactly $\frac{1}{2}$, since $\phi(p^\alpha)$ is even, and hence the probability that 2^d divides the order of a randomly chosen element in $\mathbb{Z}_{p^\alpha}^*$ is $\frac{1}{2}$, as required. \square

With this result, we can now bound the probability of the algorithm failing in Step 4, that is, choosing an $x \in [2, N - 1]$ such that its order r is odd or $x^{\frac{r}{2}} \equiv -1 \pmod{N}$.

Proposition 3.34. Let $\prod_{i=1}^m p_i^{a_i}$ be the prime factorisation of an odd, composite positive integer N . Suppose also that $x \in \mathbb{Z}_N^*$ is chosen uniformly at random and let r be its order.

Then the probability that r is even and $x^{\frac{r}{2}} \not\equiv -1 \pmod{N}$ is greater than or equal to $1 - \frac{1}{2^{m-1}}$.

Proof. This is my own proof as the proof given in [21] is incorrect.

We consider the probability that x is a member of the set

$$S := \{y \in \mathbb{Z}_N^* \mid \text{ord}(y) \text{ is odd or } y^{\frac{\text{ord}(y)}{2}} \equiv -1 \pmod{N}\} \quad (3.48)$$

where $\text{ord}(y)$ is the order of $y \in \mathbb{Z}_N^*$.

The Chinese Remainder Theorem gives us an isomorphism $\mathbb{Z}_N^* \longrightarrow \mathbb{Z}_{p_1^{a_1}}^* \times \cdots \times \mathbb{Z}_{p_m^{a_m}}^*$ defined by

$$y \mapsto (y_1, y_2, \dots, y_m) \quad (3.49)$$

such that $y \equiv y_i \pmod{p_i^{a_i}}$ for all i .

In fact, we can say something stronger:

For any $y \in \mathbb{Z}_N^*$, if we define d to be the largest index such that $2^d \mid \text{ord}(y)$, then this is equivalent under this isomorphism to choosing y_1, y_2, \dots, y_m such that not only does $y \equiv y_i \pmod{p_i^{a_i}}$ hold for all i , but also that there exist d_i that are the largest indices such that $2^{d_i} \mid \text{ord}(y_i)$.

We now look to use Lemma 3.33, as follows:

Let z be the largest integer such that $2^z \mid \phi(N)$, the order of \mathbb{Z}_N^* . Since $\phi(N) = \prod_{i=1}^m \phi(p_i^{a_i})$, we can also define z_i to be the largest integer such that $2^{z_i} \mid \phi(p_i^{a_i})$, where, clearly, by construction, $2^z = \prod_{i=1}^m 2^{z_i}$. It is worth noting that we know what z and the z_i are.

Lemma 3.33 now gives us that $2^{z_i} \mid \text{ord}(y_i)$ with probability exactly $\frac{1}{2}$. But since $2^{d_i} \leq 2^{z_i}$ by construction, and d_i is the largest index such that $2^{d_i} \mid \text{ord}(y_i)$, this Lemma really gives us that $z_i = d_i$ with probability $\frac{1}{2}$, and hence also that $z_i \neq d_i$ with probability $\frac{1}{2}$.

We claim that for all i , $\text{ord}(y_i) \mid \text{ord}(y)$.

Since $y^{\text{ord}(y)} \equiv 1 \pmod{N}$, we have that $y^{\text{ord}(y)} \equiv 1 \pmod{p_i^{a_i}}$ for all i . As $y \equiv y_i \pmod{p_i^{a_i}}$, this implies that $y_i^{\text{ord}(y)} \equiv 1 \pmod{p_i^{a_i}}$ for all i . But $\text{ord}(y_i)$ is the order of y_i in $\mathbb{Z}_{p_i^{a_i}}^*$, and so $\text{ord}(y_i) \mid \text{ord}(y)$ for all i .

Now the event “ $\text{ord}(y)$ is odd or $y^{\frac{\text{ord}(y)}{2}} \equiv -1 \pmod{N}$ ” can either occur when “ $\text{ord}(y)$ is odd” or when “ $\text{ord}(y)$ is even and $y^{\frac{\text{ord}(y)}{2}} \equiv -1 \pmod{N}$.” We consider each case:

- If $\text{ord}(y)$ is odd, then $d = 0$. Since $\text{ord}(y_i) \mid \text{ord}(y)$ for all i , the $\text{ord}(y_i)$ are also odd, and hence d_i must also be 0 for all i .
- If $\text{ord}(y)$ is even and $y^{\frac{\text{ord}(y)}{2}} \equiv -1 \pmod{N}$, then we have that $y^{\frac{\text{ord}(y)}{2}} \equiv -1 \pmod{p_i^{a_i}}$, and so $\text{ord}(y_i)$ does not divide $\frac{\text{ord}(y)}{2}$.
 - For if it did, then writing $\frac{\text{ord}(y)}{2} = \text{ord}(y_i)k$ for some k , we’d have that $y^{\frac{\text{ord}(y)}{2}} \equiv y^{\text{ord}(y_i)k} \equiv 1 \pmod{p_i^{a_i}}$. But since N is odd, $p_i^{a_i} \neq 2$ for all i , and so $-1 \not\equiv 1 \pmod{p_i^{a_i}}$, giving a contradiction.

We show that $d_i = d$ for all i .

Since $2^{d_i} \mid \text{ord}(y_i) \mid \text{ord}(y)$ and d is the max index such that $2^d \mid \text{ord}(y)$, this implies $d_i \leq d$.

Writing $\text{ord}(y) = 2^d s$, with $s \in \mathbb{Z}$ odd, we must have that $\text{ord}(y) = \text{ord}(y_i)k = 2^{d_i} s$ for some $k \in \mathbb{Z}$.

Since $\text{ord}(y_i)$ does not divide $\frac{\text{ord}(y)}{2}$, this must mean that $\frac{k}{2} \notin \mathbb{Z}$. Hence k is odd too.

Writing $\text{ord}(y_i) = 2^{d_i} t$, with $t \in \mathbb{Z}$ odd, we have that $2^d s = kt2^{d_i}$. Hence $2^d \mid kt2^{d_i}$, but since k, t are odd this implies that $2^d \mid 2^{d_i}$. Hence $d \leq d_i$ for all i , as required.

Hence

$$S \equiv \{(y_1, y_2, \dots, y_m) \in \mathbb{Z}_{p_1^{a_1}}^* \times \cdots \times \mathbb{Z}_{p_m^{a_m}}^* \mid d = d_1 = d_2 = \cdots = d_m\} \quad (3.50)$$

We now bound the size of S from above:

When we choose some $x \in \mathbb{Z}_N^*$ at random, this is us making a free choice of some index d (as each x comes with a d by construction) which either cannot be assigned to any of its d_i (which it gets from the isomorphism given by Equation (3.49)) or can be arbitrarily assigned to (at least) one of the d_i . (We have abused notation here by repeating d and d_i but we do so with the hope of keeping track of the ideas already presented in this proof.)

It is necessary that for x to be an element of S , it must be assigned to at least one of the d_i , so we let it be assigned to d_1 without loss of generality.

Notably, at most, every x in the set \mathbb{Z}_N^* has a d that can be assigned to one of the d_i , so we count under this restriction (thus calculating an upper bound for the size of S .)

Furthermore, it is also required that $d_2 = d_1 = d$, $d_3 = d_1 = d$, \dots , $d_m = d_1 = d$ for x to be an element of S , but there is now no longer any free choice over these equalities.

We know from above that, for all i , either $z_i = d_i$ or $z_i \neq d_i$ with equal probability. Hence, for all i , either $z_i = d$ or $z_i \neq d$ under the new restriction.

We need to choose those elements such that $z_i = d$ for all i in order that $d = d_1 = d_2 = \dots = d_m$. Since we've already fixed d_1 , this means that the size of S is at most $\prod_{i=2}^m \frac{1}{2} |\mathbb{Z}_N^*| = \frac{1}{2^{m-1}} |\mathbb{Z}_N^*|$.

Hence

$$P(r \text{ is odd or } x^{\frac{r}{2}} \equiv -1 \pmod{N}) = \frac{|S|}{|\mathbb{Z}_N^*|} \leq \frac{1}{2^{m-1}} \quad (3.51)$$

as required. □

Remark 3.35. In total, this means that Shor's Factoring Algorithm obtains a factor of a positive, composite integer N with $O((\log N)^3)$ resources with high - $O(1)$ - probability.

Exercise 3.36. Shor's Algorithm is an efficient factoring algorithm that comes from having an efficient Order-Finding Algorithm.

Show instead that if we had an efficient factoring algorithm, then we could find the order modulo N of any x that is coprime to N .

Proof. Suppose that we had an efficient factoring algorithm to factor N as $\prod_{i=1}^m p_i^{a_i}$, and let x be coprime to N . By the same factoring algorithm, we can factor $\phi(N)$ into

$$\phi(N) = \prod_{i=1}^m p_i^{a_i-1} (p_i - 1) \quad (3.52)$$

Since the order r of x can be at most $\phi(N)$, and $\phi(N)$ is finite, Equation (3.52) allows us to construct an increasing list of factors of $\phi(N)$

$$1 = f_1 < f_2 < \dots < f_k = \phi(N) \quad (3.53)$$

for some finite k . We can then find the order r by starting at $i = 1$ and calculating $x^{f_i} \pmod{N}$, stopping at the lowest i such that $x^{f_i} \equiv 1 \pmod{N}$. This f_i is the order r of x . □

4 Quantum Computing with Groups

In the previous chapter, we looked the Quantum Fourier Transform when applied to quantum systems of size 2^n , giving an algorithm to estimate the phase of an eigenvector of a unitary operator, an algorithm to find the order of an integer coprime to some integer N as well as an algorithm to factor a composite integer into a product of prime numbers.

In the second half of this thesis, we turn our attention to a famous problem in Quantum Computing which has its roots in Group Theory. It is called the Hidden Subgroup Problem. We will see that we can make use of an improved version of the Quantum Fourier Transform when studying this problem, one which is applied to a quantum system that is the group algebra for some finite group G . We will define this improved version in the next chapter.

For now, we only state what the Hidden Subgroup Problem is, before looking at some more advanced mathematical concepts from Group Representation Theory that we will need in order to understand under what conditions a quantum computer can be used to solve the problem.

4.1 The Hidden Subgroup Problem

Definition 4.1 (The Hidden Subgroup Problem). Let $f : G \rightarrow S$ be a function from a finite group G to a finite set S which can be computed efficiently.

In addition, suppose that f has the following property on some unknown subgroup $H \leq G$:

$$f(g) = f(g') \iff g^{-1}g' \in H$$

The goal is to find H , or equivalently, to find a generating set for H .

Remark 4.2. We note the following:

1. The word “efficiently” in the statement of the problem means that f can be computed in time polynomial to the number of bits needed to represent an input for f . The number of bits is $\log_2 |G|$ because every finite group G has a generating set of size $\leq \log_2 |G|$.
2. The function f is called a hiding function of H .

In fact, we have that f is constant on the left cosets of H in G , and distinct on different cosets.

For if we take some $g \in G$ such that $g \notin H$, we have that

$$f(g) = f(g') \iff g' \in gH \iff gH = g'H \tag{4.1}$$

4.2 Group Representation Theory

The topics addressed in this section and the next form the foundational material needed to study the Hidden Subgroup Problem. They are a collation of definitions, statements and results that can be found in one of [3], [5], [14], [24] and [25]. I have proved many of the results myself, although I reference those which come from one of these sources.

All groups G are assumed to be finite in this section and the next. However, we explicitly use the word finite in definitions where it is particularly important that the group is finite. We use additive notation when the group is abelian, most notably in Section 4.3.

Definition 4.3. A representation ρ of a group G is a choice of vector space V over \mathbb{C} and a homomorphism $\rho : G \rightarrow GL(V)$, that is, $\rho(x)\rho(y) = \rho(xy)$ for all $x, y \in G$.

The dimension d_ρ of the representation ρ is $\dim V$, the dimension of the vector space V .

If we choose a basis for V , then each $\rho(x) : V \rightarrow V$ becomes an invertible matrix in $\mathbb{C}^{d_\rho \times d_\rho}$.

Remark 4.4. Writing 1 as the identity element of G , it is clear that $\rho(1) = I_V$, the identity map $V \rightarrow V$, and $\rho(x^{-1}) = \rho(x)^{-1}$.

Remark 4.5. We are able to assume that our representations are unitary without loss of generality, that is, for all $x \in G$, $\rho(x)^{-1} = \rho(x)^\dagger$, because every representation of a finite group is isomorphic to a unitary representation [25, p. 6].

Furthermore, for each $x \in G$, there is some basis in which $\rho(x)$ is diagonalisable, since we can apply the Spectral Theorem for Normal Operators, as a unitary operator is also normal [21, p. 72].

We will often make use of these two statements in what follows.

Definition 4.6. Let $\rho_1 : G \rightarrow GL(V)$ and $\rho_2 : G \rightarrow GL(W)$ be two representations of G .

A G -linear map between ρ_1 and ρ_2 is a linear map $f : V \rightarrow W$ such that, for all $x \in G$

$$f \circ \rho_1(x) = \rho_2(x) \circ f \quad (4.2)$$

Furthermore, suppose that the G -linear map f is also an isomorphism between the vector spaces V and W .

Then we can say that ρ_1 and ρ_2 are isomorphic representations, and we denote this by $\rho_1 \sim \rho_2$.

Definition 4.7. A subrepresentation of a representation $\rho : G \rightarrow GL(V)$ is a vector subspace $W \subseteq V$ such that $\rho(x)(w) \in W$ for all $x \in G$, $w \in W$. We say that W is an invariant subspace of this representation ρ .

Remark 4.8. It is clear that the zero subspace and the subspace V of a representation $\rho : G \rightarrow GL(V)$ are always invariant subspaces.

Definition 4.9. If the zero subspace and V are the only invariant subspaces of a representation $\rho : G \rightarrow GL(V)$, then ρ is said to be irreducible.

With this, it is natural to try to find the irreducible representations of a group. We can use the following theorem:

Theorem 4.10 (Maschke's Theorem). Let $\rho : G \rightarrow GL(V)$ be a representation and let U_1 be an invariant subspace of ρ . Then there exists another invariant subspace U_2 of ρ that is complementary to U_1 , that is, $V = U_1 \oplus U_2$.

In fact, we can write $\rho = \rho_1 \oplus \rho_2$, where $\rho_i(x)$ is $\rho(x)$ restricted to U_i for $i = 1, 2$ and for all $x \in G$.

Proof. See [24, p. 23] for details. □

Corollary 4.11. Every representation $\rho : G \rightarrow GL(V)$ can be written as a direct sum

$$\rho = \rho_1 \oplus \rho_2 \oplus \cdots \oplus \rho_r \quad (4.3)$$

of irreducible (sub)representations $\rho_i : G \rightarrow GL(U_i)$, not necessarily unique.

Equivalently, we have that

$$V = U_1 \oplus U_2 \oplus \cdots \oplus U_r \quad (4.4)$$

Remark 4.12. Corollary 4.11 implies that we can choose a basis of V in which every $\rho(x)$ is block diagonal, where the i^{th} block corresponds to the i^{th} irreducible subrepresentation in the above decomposition.

Another useful Lemma involving G -linear maps and irreducible representations is the following:

Theorem 4.13 (Schur's Lemma). Let $\rho_1 : G \rightarrow GL(V)$ and $\rho_2 : G \rightarrow GL(W)$ be two irreducible representations of G .

1. Let $f : V \rightarrow W$ be a G -linear map. Then either f is an isomorphism or f is the zero map.
2. Let $f : V \rightarrow V$ be a G -linear map. Then $f = \lambda I_V$ for some $\lambda \in \mathbb{C}$.

Proof. We present the proof given in [24].

1. If f is not the zero map, then
 - $\ker(f) \subseteq V$ is an invariant subspace of ρ_1 that must be the zero subspace as ρ_1 is irreducible, and so f is injective.
 - $\text{im}(f) \subseteq W$ is an invariant subspace of ρ_2 that must be W as ρ_2 is irreducible, and so f is surjective.

Hence f is an isomorphism, and $\rho_1 \sim \rho_2$.

2. As V is a complex vector space, by the Fundamental Theorem of Algebra, $f : V \rightarrow V$ has an eigenvalue $\lambda \in \mathbb{C}$. Then $f - \lambda I_V$ is G -linear, since, for all $x \in G$, $v \in V$

$$\begin{aligned} (f - \lambda I_V)(\rho_1(x)(v)) &= f(\rho_1(x)(v)) - \lambda \rho_1(x)(v) \\ &= \rho_1(x)(f(v)) - \rho_1(x)(\lambda v) \\ &= \rho_1(x)((f - \lambda I_V)(v)) \end{aligned} \tag{4.5}$$

Furthermore, as $f - \lambda I_V$ has a non-zero kernel, it cannot be an isomorphism.

Hence $f - \lambda I_V$ must be the zero map by part 1, that is, $f = \lambda I_V$, as required. \square

Another important definition is the following:

Definition 4.14. Let $\rho : G \rightarrow GL(V)$ be a representation. Then the character of ρ is the function $\chi_\rho : G \rightarrow \mathbb{C}$ defined by $\chi_\rho(x) := \text{tr}(\rho(x))$.

Remark 4.15. Note that the trace of $\rho(x)$ is well-defined since the trace is independent of the basis chosen for V .

For if M_1 is the matrix representation of $\rho(x)$ in one basis, and M_2 is the matrix representation of $\rho(x)$ in another basis, then there exists a unitary change of basis matrix U such that $UM_1U^\dagger = M_2$, but $\text{tr}(M_2) = \text{tr}(UM_1U^\dagger) = \text{tr}(M_1UU^\dagger) = \text{tr}(M_1)$.

Proposition 4.16. Let $\rho : G \rightarrow GL(V)$ be a representation of dimension d_ρ and let χ_ρ be its character. Then

1. If x and y are conjugate in G , then $\chi_\rho(x) = \chi_\rho(y)$.
2. $\chi_\rho(1) = d_\rho$.

3. For any x in G , $\chi_\rho(x^{-1}) = \chi_\rho(x)^*$, the complex conjugate of $\chi_\rho(x)$.

Furthermore, the character $\chi_{\mathbb{1}_d}$ of the trivial representation $\mathbb{1}_d$ of G of dimension d (which maps every $x \in G$ to the identity matrix I_d) is d for every element $x \in G$.

Proof. We prove point 3, since it is the only non-obvious result.

Fix some $x \in G$, and suppose it has some order r .

As there exists a basis in which $\rho(x)$ is a diagonal matrix by Remark 4.5, in this basis, the diagonal entries are the eigenvalues of $\rho(x)$, $\{\lambda_i\}_{i=1}^{d_\rho}$.

Furthermore, each λ_i is an r^{th} root of unity in \mathbb{C} , either by Exercise 2.18, or, more directly, by the fact that, in this basis, $\rho(x)^r = \rho(x^r) = \rho(1) = I_{d_\rho}$.

Therefore, $\rho(x^{-1})$ is a diagonal matrix with entries $\{\lambda_i^{-1}\}_{i=1}^{d_\rho}$ in this basis; moreover, we have that $\lambda_i^{-1} = \lambda_i^*$, since $\rho(x^{-1}) = \rho(x)^\dagger$ by Remark 4.5.

Hence

$$\chi_\rho(x^{-1}) = \sum_{i=1}^{d_\rho} \lambda_i^{-1} = \sum_{i=1}^{d_\rho} \lambda_i^* = \left(\sum_{i=1}^{d_\rho} \lambda_i \right)^* = \chi_\rho(x)^* \quad (4.6)$$

as required. \square

Proposition 4.17. For $i = 1, 2$, let $\rho_i : G \rightarrow GL(U_i)$ be representations of G with characters χ_{ρ_i} . Then we have that

1. $\chi_{\rho_1 \oplus \rho_2} = \chi_{\rho_1} + \chi_{\rho_2}$
2. $\chi_{\rho_1 \otimes \rho_2} = \chi_{\rho_1} \chi_{\rho_2}$

Proof. Both statements can be proved by considering the matrix representations of the $\rho_i(x)$ in some bases of the U_i for all $x \in G$.

1. Fix some $x \in G$. Then we can pick bases of U_1, U_2 such that $(\rho_1 \oplus \rho_2)(x)$ is block diagonal in the basis of $U_1 \oplus U_2$ naturally formed from these bases, with the first block corresponding to $\rho_1(x)$ and the second block corresponding to $\rho_2(x)$.

Clearly $\text{tr}((\rho_1 \oplus \rho_2)(x)) = \text{tr}(\rho_1(x)) + \text{tr}(\rho_2(x))$ in this basis, and hence $\chi_{\rho_1 \oplus \rho_2}(x) = \chi_{\rho_1}(x) + \chi_{\rho_2}(x) = (\chi_{\rho_1} + \chi_{\rho_2})(x)$.

2. Again, fix some $x \in G$. Then picking bases of U_1, U_2 , we obtain a basis of $U_1 \otimes U_2$ such that $\rho_1 \otimes \rho_2$ has the matrix representation

$$M = \begin{pmatrix} \rho_1(x)_{11} \rho_2(x) & \rho_1(x)_{12} \rho_2(x) & \cdots & \rho_1(x)_{1d_{\rho_1}} \rho_2(x) \\ \rho_1(x)_{21} \rho_2(x) & \rho_1(x)_{22} \rho_2(x) & \cdots & \rho_1(x)_{2d_{\rho_1}} \rho_2(x) \\ \vdots & \vdots & \ddots & \vdots \\ \rho_1(x)_{d_{\rho_1}1} \rho_2(x) & \rho_1(x)_{d_{\rho_1}2} \rho_2(x) & \cdots & \rho_1(x)_{d_{\rho_1}d_{\rho_1}} \rho_2(x) \end{pmatrix} \quad (4.7)$$

using the Kronecker Product of Definition 2.22, where $\rho_1(x)_{ij} \rho_2(x)$ is a block of size $d_{\rho_2} \times d_{\rho_2}$ for all i, j .

Then

$$\begin{aligned}
\chi_{\rho_1 \otimes \rho_2}(x) &= \text{tr}(M) \\
&= \text{tr} \left(\sum_{i=1}^{d_{\rho_1}} [\rho_1(x)_{ii} \rho_2(x)] \right) \\
&= \text{tr} \left(\left(\sum_{i=1}^{d_{\rho_1}} \rho_1(x)_{ii} \right) \rho_2(x) \right) \\
&= \left(\sum_{i=1}^{d_{\rho_1}} \rho_1(x)_{ii} \right) \text{tr}(\rho_2(x)) \\
&= \text{tr}(\rho_1(x)) \text{tr}(\rho_2(x)) \\
&= \chi_{\rho_1}(x) \chi_{\rho_2}(x)
\end{aligned} \tag{4.8}$$

as required. \square

Corollary 4.18. We can naturally extend Proposition 4.17 to n representations of G , namely

1. $\chi_{\bigoplus_{i=1}^n \rho_i} = \sum_{i=1}^n \chi_{\rho_i}$
2. $\chi_{\bigotimes_{i=1}^n \rho_i} = \prod_{i=1}^n \chi_{\rho_i}$

We now consider functions $G \rightarrow \mathbb{C}$ in the following way:

Definition 4.19. Let \mathbb{C}^G be the set of functions $G \rightarrow \mathbb{C}$. Then \mathbb{C}^G is a vector space over \mathbb{C} , with linear combinations of elements defined in the obvious manner by

$$\begin{aligned}
(\lambda_1 f_1 + \lambda_2 f_2) : G &\rightarrow \mathbb{C} \\
x &\mapsto \lambda_1 f_1(x) + \lambda_2 f_2(x)
\end{aligned} \tag{4.9}$$

Furthermore, we can augment this vector space with the function $(*, *) : \mathbb{C}^G \times \mathbb{C}^G \rightarrow \mathbb{C}$

$$(f_1, f_2) := \frac{1}{|G|} \sum_{x \in G} f_1(x)^* f_2(x) \tag{4.10}$$

which can be shown to be an inner product using Definition 2.1.

Note that $(*, *)$ is conjugate linear in the first argument, and linear in the second argument.

A related construct is the following:

Definition 4.20. Let G be any group.

Then we can define \widehat{G} to be the set of homomorphisms $G \rightarrow \mathbb{C}^\times$. (Note that $\widehat{G} \subset \mathbb{C}^G$, as sets.)

If we augment \widehat{G} with the following binary operation, the pointwise multiplication of functions:

$$(f_1 f_2)(x) = f_1(x) f_2(x) \quad \text{for all } f_1, f_2 \in \widehat{G}, x \in G \tag{4.11}$$

then \widehat{G} can be shown to be a group, which is called the dual group.

Note that \widehat{G} is an abelian group since multiplication in \mathbb{C}^\times is commutative.

But, for now, using the inner product of \mathbb{C}^G , it is possible to show the following:

Theorem 4.21. Let $\chi_{\rho_1}, \chi_{\rho_2}, \dots, \chi_{\rho_r}$ be the irreducible characters of G . Then

$$(\chi_{\rho_i}, \chi_{\rho_j}) = \delta_{ij} \quad (4.12)$$

which means that the irreducible characters form a linearly independent set of orthonormal vectors in \mathbb{C}^G .

In fact, we can say more:

Theorem 4.22. Let $\rho : G \rightarrow GL(V)$ be a representation, and let $\rho_1, \rho_2, \dots, \rho_r$ be the irreducible representations of G . Then, because we can decompose ρ as a direct sum of the irreducible representations of G by Corollary 4.11, by applying Corollary 4.18, we can say that

$$\chi_\rho = m_1\chi_{\rho_1} + m_2\chi_{\rho_2} + \dots + m_r\chi_{\rho_r} \quad (4.13)$$

where m_i is the number of copies of ρ_i in ρ .

But we can find the m_i , since

$$m_i = (\chi_\rho, \chi_{\rho_i}) \quad (4.14)$$

by Theorem 4.21.

Corollary 4.23. We can therefore deduce a test for irreducibility from Theorem 4.22:

A representation $\rho : G \rightarrow GL(V)$ is irreducible if and only if $(\chi_\rho, \chi_\rho) = 1$.

In giving these last two results, we have presupposed that we are able to find out exactly what all the irreducible representations of a finite group G are, and we have assumed that there are a finite number of them, up to isomorphism. We now give a method for finding all of them for an arbitrary finite group G .

Definition 4.24. Let G be a finite group, and let V_L be a vector space that has a basis enumerated by the elements of G , which we denote by $\{|y\rangle \mid y \in G\}$.

Then $L : G \rightarrow GL(V_L)$ is a representation of dimension $|G|$ defined by

$$L(x)|y\rangle = |xy\rangle \quad (4.15)$$

for all $x \in G$.

L is called the left regular representation of G .

We can also define a representation $R : G \rightarrow GL(V_R)$ of dimension $|G|$, where V_R is a vector space with the same basis as V_L , by

$$R(x)|y\rangle = |yx^{-1}\rangle \quad (4.16)$$

for all $x \in G$.

R is called the right regular representation of G .

An important fact is the following:

Theorem 4.25. The left regular representation of G contains every irreducible representation of G , and the number of irreducible representations of G is finite.

In fact, if $\{\rho_i : G \rightarrow GL(U_i)\}_{i=1}^r$ are the irreducible representations of G with dimensions $\{d_i\}_{i=1}^r$, then

$$V_L = \bigoplus_{i=1}^r U_i^{\oplus d_i} \quad (4.17)$$

and so the left regular representation contains every irreducible representation exactly d_i times.

Equivalently, we can say that

$$L = \bigoplus_{i=1}^r d_i \rho_i \quad (4.18)$$

which we will prove to be isomorphic to the operator

$$L \cong \bigoplus_{i=1}^r [\rho_i \otimes I_{d_{\rho_i}}] \quad (4.19)$$

in Proposition 7.6.

Corollary 4.26. By counting the dimensions of Equation (4.17), we have that

$$\sum_{i=1}^r d_i^2 = |G| \quad (4.20)$$

The following result is also useful:

Lemma 4.27. For any $x \in G$, $x \neq 1$, we have that

$$\sum_{i=1}^r d_{\rho_i} \chi_{\rho_i}(x) = 0 \quad (4.21)$$

where the sum is over the irreducible representations of G .

Proof. Fix $x \in G - \{1\}$. Then $xy \neq y$ for all $y \in G$, and so the matrix representation of $L(x)$ in the basis $\{|y\rangle \mid y \in G\}$ of V_L has a 0 in every entry along the diagonal.

Hence $\chi_L(x) = 0$, and so, using Equation (4.19) and Corollary 4.18, we have that

$$0 = \chi_L(x) = \chi_{\left(\bigoplus_{i=1}^r [\rho_i \otimes I_{d_{\rho_i}}]\right)}(x) = \sum_{i=1}^r \chi_{(\rho_i \otimes I_{d_{\rho_i}})}(x) = \sum_{i=1}^r \chi_{\rho_i}(x) \chi_{I_{d_{\rho_i}}}(x) = \sum_{i=1}^r d_{\rho_i} \chi_{\rho_i}(x) \quad (4.22)$$

as required. \square

Theorem 4.28. We can perform a similar analysis for the right regular representation of G , and get that

$$R \cong \bigoplus_{i=1}^r [I_{d_{\rho_i}} \otimes \rho_i^*] \quad (4.23)$$

We prove this result in Proposition 7.6.

In Group Representation Theory, we often consider a vector space that is isomorphic to V_L , since it also has a basis indexed by the elements of a group G , but we use it in a rather different way.

Definition 4.29. Let G be a finite group.

Then we can form a vector space over \mathbb{C} which has the group elements of G as a basis, denoted by $\{|x\rangle \mid x \in G\}$, such that

$$\sum_{x \in G} \lambda_x |x\rangle + \sum_{x \in G} \mu_x |x\rangle := \sum_{x \in G} (\lambda_x + \mu_x) |x\rangle \quad (4.24)$$

where $\lambda_x, \mu_x \in \mathbb{C}$ for all $x \in G$.

We denote this vector space by $\mathbb{C}[G]$.

In fact, $\mathbb{C}[G]$ is often called the group algebra, because the vector space can also be augmented with a bilinear map $\mathbb{C}[G] \times \mathbb{C}[G] \rightarrow \mathbb{C}[G]$ defined on this basis by $(|x\rangle, |y\rangle) \mapsto |xy\rangle$ for all $x, y \in G$ and extended bilinearly, which turns it into an algebra.

Another useful result relating to the orthogonality of irreducible representations is the following:

Theorem 4.30. For any two irreducible representations ρ and σ of G , we have that

$$\frac{d_\rho}{|G|} \sum_{x \in G} \rho(x)_{i,j}^* \sigma(x)_{k,l} = \delta_{\rho\sigma} \delta_{ik} \delta_{jl} \quad (4.25)$$

where $\delta_{\rho\sigma}$ is 1 if ρ is isomorphic to σ , and 0 otherwise.

We end this section with the following remark:

Remark 4.31. If $\rho : G \rightarrow GL(V)$ is a representation of G , and H is any subgroup of G , then ρ restricted to H is a representation of H .

Irreducible representations of G may not be irreducible when restricted to H .

We write the inner product given in Equation (4.10) as $(*, *)_H$ when representations of G are being considered as representations of H .

4.3 The Characters of Finite Abelian Groups

In order to study the Hidden Subgroup Problem in the case where the group is abelian, we need to understand more about the characters of finite abelian groups. We remind the reader that we will use additive notation for the abelian groups. We begin with the following Proposition:

Proposition 4.32. Every irreducible representation of a finite abelian group G has dimension 1.

Proof. We present an amended proof of the one given in [24].

Let $\rho : G \rightarrow GL(V)$ be an irreducible representation of G , and pick some arbitrary $x \in G$. Then $\rho(x) : V \rightarrow V$ is a G -linear map, since, for all $g \in G$, $v \in V$, we have that

$$\rho(x)(\rho(g)(v)) = \rho(xg)(v) = \rho(gx)(v) = \rho(g)(\rho(x)(v)) \quad (4.26)$$

Applying Schur's Lemma (Theorem 4.13) gives $\rho(x) = \lambda_x I_V$. Now, fixing $v \in V$, we have that

$$\rho(x)(\mu v) = \lambda_x(\mu v) \quad (4.27)$$

for some $\mu \in \mathbb{C}$, and so $\text{span}(v)$ is an invariant subspace of ρ of dimension 1 (since the choice of $x \in G$ was arbitrary).

But since ρ is irreducible, we must have that $\text{span}(v) = V$, and so V has dimension 1, as required. \square

Example 4.33. We can list the irreducible representations of a cyclic group $G = (\mathbb{Z}_N, +)$ of order N in terms of the generator $1 \in G$.

Since a representation of G of dimension 1 is a homomorphism $\rho : G \rightarrow \mathbb{C}^\times$ (as $GL(\mathbb{C}) \cong \mathbb{C}^\times$) we have that it is fully determined by the value of $\rho(1) \in \mathbb{C}^\times$ such that $\rho(1)^N = 1$.

Hence $\rho(1) = \omega_N^k$ for some $k \in \{0, 1, \dots, N-1\}$, where $\omega_N := e^{\frac{2\pi i}{N}}$.

This gives N distinct irreducible representations

$$\begin{aligned} \rho_k : G &\rightarrow \mathbb{C}^\times \\ j &\mapsto \omega_N^{jk} \end{aligned} \quad (4.28)$$

for $k \in \{0, 1, \dots, N-1\}$.

These are all the irreducible representations of G since G has N irreducible representations, which can be seen by considering the left regular representation of G and applying Proposition 4.32 to Equation (4.17).

It is worth noting also that $\rho_k \rho_l = \rho_{k+l}$.

Example 4.34. We can extend Example 4.33 to all finite abelian groups $(G, +)$ of order $|G| = N$, as follows:

We know that a finite abelian group G of order N is a direct product of cyclic groups:

$$G \cong \mathbb{Z}_{N_1} \times \mathbb{Z}_{N_2} \times \cdots \times \mathbb{Z}_{N_r} \quad (4.29)$$

As before, we consider the irreducible representations of G and apply Proposition 4.32 to state that an irreducible representation must be a homomorphism $\rho : G \rightarrow \mathbb{C}^\times$.

As G is a direct product of cyclic groups, it has r generators $\{\gamma_i\}_{i=1}^r$ of the form

$$\gamma_i = (0, 0, \dots, 0, 1_i, 0, \dots, 0) \quad (4.30)$$

(where 1_i is the element $1 \in \mathbb{Z}_{N_i}$) such that $N_i \gamma_i$ is the identity 0_G of G .

Hence ρ is determined entirely by the set of numbers $\{\rho(\gamma_i)\}_{i=1}^r$ such that $\rho(\gamma_i)^{N_i} = 1$ for all i . Therefore $\rho(\gamma_i) = \omega_{N_i}^{k_i}$ for some $k_i \in \{0, 1, \dots, N_i - 1\}$.

Since every element of G is of the form $(\sum_{i=1}^r j_i \gamma_i)$ for $j_i \in \{0, 1, \dots, N_i - 1\}$, $i = 1$ to r , we can label the irreducible representations of G as:

$$\begin{aligned} \rho_{k_1, k_2, \dots, k_r} : G &\rightarrow \mathbb{C}^\times \\ \left(\sum_{i=1}^r j_i \gamma_i \right) &\mapsto \prod_{i=1}^r \omega_{N_i}^{j_i k_i} \end{aligned} \quad (4.31)$$

for $k_i \in \{0, 1, \dots, N_i - 1\}$. This gives us $N = \prod_{i=1}^r N_i$ distinct irreducible representations.

As before, these are all the irreducible representations of G , for the same reasoning as when G was merely a cyclic group.

Also as before, denoting the tuple (k_1, \dots, k_r) by k , we can see that $\rho_k \rho_l = \rho_{k+l}$.

Remark 4.35. Using the notation of Examples 4.33 and 4.34, we can see that the representations of the abelian group $(G, +)$ are just the products of the representations of the individual cyclic groups \mathbb{Z}_{N_i} . That is,

$$\rho_k(j) = \prod_{i=1}^r \rho_{k_i}(j_i) \quad (4.32)$$

under the isomorphism $G \cong \mathbb{Z}_{N_1} \times \mathbb{Z}_{N_2} \times \cdots \times \mathbb{Z}_{N_r}$, where we have defined $j := (\sum_{i=1}^r j_i \gamma_i) \in G$.

Remark 4.36. It is useful to note that a representation ρ of dimension 1 of a group G is exactly the same function as the character χ_ρ .

Hence the irreducible representations of an abelian group G are precisely the irreducible characters of G ; moreover, they are orthonormal in the Hilbert space \mathbb{C}^G under the inner product stated in Equation (4.10) by Theorem 4.21.

Therefore, we will proceed using the language of irreducible characters in the rest of this section.

Note that this means that we can rewrite Equation (4.32) as

$$\chi_k(j) = \prod_{i=1}^r \chi_{k_i}(j_i) \quad (4.33)$$

where we have defined the character of ρ_k to be χ_k and the characters of the ρ_{k_i} to be χ_{k_i} .

Remark 4.37. We proved in Exercise 4.34 that a finite abelian group G had $|G|$ irreducible representations, which are the irreducible characters of G by Remark 4.36.

Hence, for a finite abelian group G , we have that

$$|G| = |\widehat{G}| \quad (4.34)$$

where \widehat{G} is the dual group of G , as given in Definition 4.20.

Our aim is to show something stronger: that if G is abelian, then $G \cong \widehat{\widehat{G}}$.

First, consider the following:

Theorem 4.38. If G is a cyclic group, then $G \cong \widehat{G}$.

Proof. We adjust the proof given in [5].

Suppose that $|G| = N$. Using additive notation, we have that $G \cong (\mathbb{Z}_N, +)$. It is sufficient to show that $\mathbb{Z}_N \cong \widehat{\mathbb{Z}_N}$

Considering the generator 1 of \mathbb{Z}_N , we claim that

$$\begin{aligned} \chi_1 : \mathbb{Z}_N &\rightarrow \mathbb{C}^\times \\ j &\mapsto \omega_N^j \end{aligned} \quad (4.35)$$

as defined in Remark 4.33 (using the notation of characters instead of representations, by Remark 4.36) is a generator of $\widehat{\mathbb{Z}_N}$.

Indeed, let f be any element of $\widehat{\mathbb{Z}_N}$. Then $f = \chi_k$ for some $k \in \{0, 1, \dots, N-1\}$, also by Remark 4.33, and so

$$f(j) = \chi_k(j) = \omega_N^{jk} = \chi_1(j)^k \quad (4.36)$$

Hence $f = \chi_1^k$, and so χ_1 generates $\widehat{\mathbb{Z}_N}$.

Since $|\mathbb{Z}_N| = |\widehat{\mathbb{Z}_N}|$ by Equation (4.34), we must have that $\mathbb{Z}_N \cong \widehat{\mathbb{Z}_N}$ (via the isomorphism $1 \mapsto \chi_1$). \square

Lemma 4.39. If $(G_1, +)$ and $(G_2, +)$ are finite abelian groups, then $\widehat{G_1 \times G_2} \cong \widehat{G_1} \times \widehat{G_2}$.

Proof. We revise the proof given in [5].

For a character χ of $(G_1 \times G_2, +)$, define χ_1 and χ_2 to be characters on G_1 and G_2 respectively by

- $\chi_1(x) = \chi(x, 0)$ for all $x \in G_1$.
- $\chi_2(y) = \chi(0, y)$ for all $y \in G_2$.

Then because

$$\chi(x, y) = \chi((x, 0) + (0, y)) = \chi_1(x)\chi_2(y) \quad (4.37)$$

for all $(x, y) \in G_1 \times G_2$, we get a map

$$\begin{aligned} \phi : \widehat{G_1 \times G_2} &\rightarrow \widehat{G_1} \times \widehat{G_2} \\ \chi &\mapsto (\chi_1, \chi_2) \end{aligned} \quad (4.38)$$

which is easily shown to be a group homomorphism.

To show that ϕ is an isomorphism, we firstly prove that ϕ is injective and then prove that the groups $\widehat{G_1 \times G_2}$ and $\widehat{G_1} \times \widehat{G_2}$ are of the same order.

Indeed, to prove that ϕ is injective, it is sufficient to show that $\ker(\phi)$ is trivial:

If $\chi \in \ker(\phi)$, then χ_1 and χ_2 must be the trivial characters, and so $\chi_1(x) = 1 = \chi_2(y)$ for all $x \in G_1, y \in G_2$. But $1 = \chi_1(x)\chi_2(y) = \chi(x, y)$ for all $(x, y) \in G_1 \times G_2$, and so χ must be the trivial character in $\widehat{G_1 \times G_2}$.

Furthermore, because

$$|\widehat{G_1 \times G_2}| = |G_1 \times G_2| = |G_1||G_2| = |\widehat{G_1}||\widehat{G_2}| = |\widehat{G_1} \times \widehat{G_2}| \quad (4.39)$$

by Equation (4.34), we have shown that ϕ is an isomorphism of groups, as required. \square

Theorem 4.40. If G is an abelian group, then $G \cong \widehat{\widehat{G}}$.

Proof. We amend the proof given in [5].

Since G is abelian, it is isomorphic to a direct product of cyclic groups:

$$G \cong \mathbb{Z}_{N_1} \times \mathbb{Z}_{N_2} \times \cdots \times \mathbb{Z}_{N_r} \quad (4.40)$$

Hence, first by Lemma 4.39 and then by Theorem 4.38, we have that

$$\widehat{G} \cong (\mathbb{Z}_{N_1} \times \mathbb{Z}_{N_2} \times \cdots \times \mathbb{Z}_{N_r})^\wedge \cong \widehat{\mathbb{Z}_{N_1}} \times \widehat{\mathbb{Z}_{N_2}} \times \cdots \times \widehat{\mathbb{Z}_{N_r}} \cong \mathbb{Z}_{N_1} \times \mathbb{Z}_{N_2} \times \cdots \times \mathbb{Z}_{N_r} \cong G \quad (4.41)$$

as required. \square

Another interesting theorem regarding the characters of finite abelian groups is the following:

Theorem 4.41 (Pontryagin Duality). Let G be a finite abelian group. There is a natural isomorphism between G and the double-dual group of G , $\widehat{\widehat{G}}$ (the dual group of \widehat{G}), defined by

$$\begin{aligned} \phi : G &\rightarrow \widehat{\widehat{G}} \\ x &\mapsto \Psi_x \end{aligned} \quad (4.42)$$

where

$$\begin{aligned}\Psi_x : \widehat{G} &\rightarrow \mathbb{C}^\times \\ \chi &\mapsto \chi(x)\end{aligned}\tag{4.43}$$

is the function that evaluates $\chi \in \widehat{G}$ at $x \in G$.

Remark 4.42. Note that for $x \in G$, Ψ_x is a representation of \widehat{G} of dimension 1, because

$$\Psi_x(\chi_1\chi_2) = (\chi_1\chi_2)(x) = \chi_1(x)\chi_2(x) = \Psi_x(\chi_1)\Psi_x(\chi_2)\tag{4.44}$$

where we have used the definition of the binary operation of \widehat{G} in the second equality.

Since \widehat{G} is abelian, we have that Ψ_x is a character of \widehat{G} by Remark 4.36, and so $\Psi_x \in \widehat{\widehat{G}}$.

To prove Theorem 4.41, we need the following Lemma:

Lemma 4.43. Let G be a finite abelian group. If x is not the identity of G , then $\chi(x) \neq 1$ for some character $\chi \in \widehat{G}$.

Proof. Suppose that $\chi(x) = 1$ for all irreducible characters of G . We know that there are $|G|$ of them, by Equation (4.34).

As the irreducible characters of G are the same as the irreducible representations of G by Remark 4.36, we have that $\rho(x) = 1$ for all irreducible representations ρ of G , and they are all of dimension 1.

If we consider the left regular representation L , then Equation (4.18) implies that

$$L(x) = \bigoplus_{i=1}^{|G|} \rho_i(x) = I_{V_L}\tag{4.45}$$

The only element of G that L maps to the identity matrix is the identity element, and hence $x = 1$, as required. \square

Proof of Pontryagin Duality. We know that G and $\widehat{\widehat{G}}$ have the same size by Equation (4.34), and $\ker(\phi)$ is trivial since if $x \in \ker(\phi)$, then every character of G maps x to $1 \in \mathbb{C}$, and so x is the identity of G by Lemma 4.43.

Hence ϕ is an injection between groups of the same size, and so ϕ must be an isomorphism. \square

In order to develop another equation that looks like the inner product of \mathbb{C}^G given in Equation (4.10), we need the following Lemma:

Lemma 4.44. Let G be a finite group. Then

$$\sum_{x \in G} \chi(x) = \begin{cases} |G| & \text{if } \chi = \mathbb{1}_G \\ 0 & \text{if } \chi \neq \mathbb{1}_G \end{cases}\tag{4.46}$$

Proof. Indeed, if $\chi = \mathbb{1}_G$, then $\mathbb{1}_G(x) = 1$ for all $x \in G$, and so $\sum_{x \in G} \chi(x) = |G|$.

Otherwise, there exists some x_0 such that $\chi(x_0) \neq 1$. Then

$$\chi(x_0) \left(\sum_{x \in G} \chi(x) \right) = \sum_{x \in G} \chi(x_0x) = \sum_{x \in G} \chi(x)\tag{4.47}$$

and so $\sum_{x \in G} \chi(x) = 0$. \square

Theorem 4.45. Let G be a finite abelian group. Then, for elements $x_1, x_2 \in G$, we have that

$$\frac{1}{|G|} \sum_{\chi \in \widehat{G}} \chi(x_1)^* \chi(x_2) = \begin{cases} 1 & \text{if } x_1 = x_2 \\ 0 & \text{if } x_1 \neq x_2 \end{cases} \quad (4.48)$$

where the sum is over the irreducible characters χ of G .

Proof. We first consider

$$\sum_{\chi \in \widehat{G}} \chi(x) \quad (4.49)$$

By Pontryagin Duality, we have that

$$\sum_{\chi \in \widehat{G}} \chi(x) = \sum_{\chi \in \widehat{G}} \Psi_x(\chi) \quad (4.50)$$

and so, by Lemma 4.44, we have that

$$\sum_{\chi \in \widehat{G}} \Psi_x(\chi) = \begin{cases} |G| & \text{if } \Psi_x = \mathbb{1}_{\widehat{G}} \\ 0 & \text{if } \Psi_x \neq \mathbb{1}_{\widehat{G}} \end{cases} \quad (4.51)$$

since Ψ_x is a character of the abelian group \widehat{G} and the sum is over the elements of \widehat{G} .

But Lemma 4.43 states that

$$\Psi_x = \mathbb{1}_{\widehat{G}} \iff \chi(x) = 1 \text{ for all } \chi \in \widehat{G} \iff x = 1 \quad (4.52)$$

Hence

$$\sum_{\chi \in \widehat{G}} \chi(x) = \begin{cases} |G| & \text{if } x = 1 \\ 0 & \text{if } x \neq 1 \end{cases} \quad (4.53)$$

Setting $x = x_1^{-1}x_2$ and applying Proposition 4.16, we get that

$$\sum_{\chi \in \widehat{G}} \chi(x_1)^* \chi(x_2) = \begin{cases} |G| & \text{if } x_1 = x_2 \\ 0 & \text{if } x_1 \neq x_2 \end{cases} \quad (4.54)$$

Dividing through by $|G|$ gives the result. \square

To end this section, we introduce an important subgroup of \widehat{G} that will be used often in the Hidden Subgroup Problem.

Theorem 4.46 (Orthogonal Subgroup of the Dual Group). Let G be a finite abelian group, and suppose that H is a subgroup of G .

Then we can define the set

$$H^\perp = \{\chi \in \widehat{G} \mid \chi(h) = 1 \text{ for all } h \in H\} \quad (4.55)$$

Then H^\perp is a subgroup of \widehat{G} which is isomorphic to $\widehat{G/H}$, and so $|H^\perp| = [G : H]$.

Proof. It is easy to show that H^\perp is a subgroup, so we will prove that $H^\perp \cong \widehat{G/H}$.

Again, we use additive notation for the abelian groups, and write 1 as a generator of G .

Since G is abelian, so is G/H , and so $|\widehat{G/H}| = |G/H| = [G : H]$ by Equation (4.34).

Hence we can define a mapping

$$\begin{aligned} \phi : \widehat{G/H} &\rightarrow \widehat{G} \\ \chi_k &\rightarrow \chi_k \circ f := \psi_k \end{aligned} \quad (4.56)$$

where $f : G \rightarrow G/H$ is the canonical homomorphism and χ_k is an irreducible character in $\widehat{G/H}$ for $k \in \{0, 1, \dots, [G : H] - 1\}$

This is easily seen to be a homomorphism of groups as a consequence of the binary operation of the dual group being the pointwise multiplication of functions.

We claim that $im(\phi) \subseteq H^\perp$:

Indeed, for some $\chi_k \in \widehat{G/H}$, we have that

$$\psi_k(j) = (\chi_k \circ f)(j) = \chi_k(j + H) = \omega_{[G:H]}^{\bar{j}k} \quad (4.57)$$

where $\bar{j} := j + H$ (and can be thought of as $j \bmod |H|$ in the exponent of $\omega_{[G:H]}$.)

Hence

$$\psi_k(h) = \chi(\bar{0}) = 1 \quad \text{for all } h \in H \quad (4.58)$$

and so $\psi_k \in H^\perp$.

We also claim that $ker(\phi)$ is trivial:

If $\chi_k \in \widehat{G/H}$ then

$$\begin{aligned} \chi_k \circ f &= \mathbb{1}_G \\ \Rightarrow \chi_k(\bar{j}) &= 1 \quad \text{for all } \bar{j} \in G/H \\ \Rightarrow \omega_{[G:H]}^{\bar{j}k} &= 1 \quad \text{for all } \bar{j} \in G/H \\ \Rightarrow k &= 0 \end{aligned}$$

Hence $\chi_k = \mathbb{1}_{G/H}$.

Therefore, by the First Isomorphism of Groups, we have that $\widehat{G/H} \cong im(\phi)$ and so

$$|\widehat{G/H}| = |im(\phi)| \leq |H^\perp| \quad (4.59)$$

since $im(\phi) \subseteq H^\perp$.

To prove that $H^\perp \cong \widehat{G/H}$, it is enough to show that $|H^\perp| \leq |\widehat{G/H}|$, because then we will have that $H^\perp = im(\phi)$.

To do this, we construct another homomorphism of groups, as follows:

Let $\chi \in H^\perp$. Define $\psi : G/H \rightarrow \mathbb{C}^\times$ by $x + H \mapsto \chi(x)$, where x is a representative of the coset $x + H$ in G .

ψ is a well defined mapping, since if we choose another coset representative $g \in x + H$, then

$$\psi(x + H) = \chi(g) = \chi(x + h) = \chi(x)\chi(h) = \chi(x) \quad (4.60)$$

since $\chi \in H^\perp$.

Therefore ψ is a character of G/H .

We can now define the map

$$\begin{aligned} \alpha : H^\perp &\rightarrow \widehat{G/H} \\ \chi &\rightarrow \psi \end{aligned} \quad (4.61)$$

which is easily shown to be a homomorphism, again by the binary operation of the dual group.

We claim that $\ker(\alpha)$ is trivial:

Indeed, if $\chi \in \ker(\alpha)$, then $\chi(x) = \psi(xH) = \mathbb{1}_{G/H}(x) = 1$ for all $x \in G$, and so $\chi = \mathbb{1}_G$.

Therefore, $|H^\perp| \leq |\widehat{G/H}|$, and so we are done. □

5 The Quantum Fourier Transform for Abelian Groups

In Chapter 3, on the Applications of the Quantum Fourier Transform, we stated that we would introduce “improved” versions of the Quantum Fourier Transform as and when we needed them. In that chapter, we looked at how we could implement the Quantum Fourier Transform when the quantum system was of size 2^n for some $n \in \mathbb{Z}_{\geq 1}$. We now need to generalise our approach by looking at quantum systems that are composed of linear combinations of a basis whose elements are indexed by the elements of an abelian group G . We therefore need a new version of the Quantum Fourier Transform which can be applied to states in such a quantum system, which is where we begin this chapter. We find our motivation for this chapter in [3], [4] and [8].

5.1 Quantum Fourier Transform, version 2

Definition 5.1 (Quantum Fourier Transform). Let G be a finite abelian group of size $|G|$.

Consider a quantum system over \mathbb{C} with the computational basis labelled by the elements of G , namely $\{|j\rangle \mid j \in G\}$.

Note that this quantum system is $\mathbb{C}[G]$, the group algebra from Definition 4.29, considered as a vector space over \mathbb{C} , augmented with the inner product $\langle * | * \rangle : \mathbb{C}[G] \times \mathbb{C}[G] \rightarrow \mathbb{C}$ that is defined on the computational basis by $\langle i | j \rangle = \delta_{ij}$ and extended conjugate-linearly in the first argument and linearly in the second argument.

We can define a linear operator F_G on this basis as follows:

$$F_G : \mathbb{C}[G] \rightarrow \mathbb{C}[\widehat{G}]$$

$$|j\rangle \mapsto \frac{1}{\sqrt{|G|}} \sum_{k \in \widehat{G}} \chi_k(j) |k\rangle \quad (5.1)$$

where \widehat{G} is the dual group of G , and the sum is over the irreducible characters of G that forms a basis of $\mathbb{C}[\widehat{G}]$. Note that they are all one-dimensional by Example 4.34.

F_G is called the Quantum Fourier Transform over the finite abelian group G .

Remark 5.2. As before, we can write F_G using outer product notation, namely

$$F_G = \frac{1}{\sqrt{|G|}} \sum_{x \in G} \sum_{k \in \widehat{G}} \chi_k(x) |k\rangle \langle x| \quad (5.2)$$

with the proof being very similar to that of Remark 3.2.

Furthermore, because there is an isomorphism of groups $\widehat{\widehat{G}} \cong G$ by Theorem 4.40, this implies that $\mathbb{C}[\widehat{\widehat{G}}] \cong \mathbb{C}[G]$ is an isomorphism of vector spaces over \mathbb{C} .

Therefore, in Equation (5.2), we can change the sum over the elements of \widehat{G} to a sum over the elements of G should it be needed, and consider F_G as an operator $\mathbb{C}[G] \rightarrow \mathbb{C}[G]$ where appropriate.

Exercise 5.3. Verify that F_G is a unitary operator, using the orthogonality of operators as given by Equation (4.48).

Proof. We show that $F_G^\dagger F_G = I$.

But first, we need to know what F_G^\dagger is:

$$F_G^\dagger = \left(\frac{1}{\sqrt{|G|}} \sum_{w \in G} \sum_{z \in \widehat{G}} \chi_z(w) |z\rangle \langle w| \right)^\dagger = \frac{1}{\sqrt{|G|}} \sum_{w \in G} \sum_{z \in \widehat{G}} \chi_z(w)^* |w\rangle \langle z| \quad (5.3)$$

where we have used the anti-linearity of the adjoint from Exercise 2.11.

Therefore

$$\begin{aligned} F_G^\dagger F_G &= \frac{1}{|G|} \sum_{x, w \in G} \sum_{k, z \in \widehat{G}} \chi_z(w)^* \chi_k(x) |w\rangle \langle z| k\rangle \langle x| \\ &= \frac{1}{|G|} \sum_{x, w \in G} \sum_{k, z \in \widehat{G}} \chi_z(w)^* \chi_k(x) \delta_{zk} |w\rangle \langle x| \\ &= \frac{1}{|G|} \sum_{x, w \in G} \sum_{k \in \widehat{G}} \chi_k(w)^* \chi_k(x) |w\rangle \langle x| \end{aligned} \quad (5.4)$$

Since

$$\frac{1}{|G|} \sum_{k \in \widehat{G}} \chi_k(w)^* \chi_k(x) = \delta_{wx} \quad (5.5)$$

by Theorem 4.45, we have that Equation (5.4) equals

$$\sum_{x, w \in G} \delta_{wx} |w\rangle \langle x| = \sum_{x \in G} |x\rangle \langle x| = I \quad (5.6)$$

using the Completeness Relation from Example 2.7.

Hence F_G is a unitary operator, as required. \square

We should show that F_G is indeed a new and improved version of “version 1” of the Quantum Fourier Transform, F_{2^n} .

Example 5.4. Consider the group $G = (\mathbb{Z}_{2^n}, +)$.

Then, by Example 4.33, we have that the irreducible characters of \mathbb{Z}_{2^n} are

$$\begin{aligned} \chi_k : \mathbb{Z}_{2^n} &\rightarrow \mathbb{C} \\ |j\rangle &\mapsto \omega_{2^n}^{jk} \end{aligned} \quad (5.7)$$

for $k \in \{0, 1, \dots, 2^n - 1\}$, where $\omega_{2^n} := e^{\frac{2\pi i}{2^n}}$.

Hence, by Equation (5.1), $F_{\mathbb{Z}_{2^n}}$ maps

$$|j\rangle \mapsto \frac{1}{\sqrt{2^n}} \sum_{k=0}^{2^n-1} \omega_{2^n}^{jk} |k\rangle \quad (5.8)$$

which is exactly the same as the definition of F_{2^n} given in Equation (3.1).

In order to keep our notation consistent, going forward we will write $F_{\mathbb{Z}_{2^n}}$ instead of F_{2^n} .

Another very nice example of this form of the Quantum Fourier Transform is the following, which we have seen before in Exercise 2.24, and have used in the Phase Estimation procedure:

Example 5.5. Consider the group $G = (\mathbb{Z}_2^n, +)$, and let $x = (x_1, x_2, \dots, x_n)$ and $y = (y_1, y_2, \dots, y_n)$ be arbitrary elements of G .

Then we claim that the irreducible characters of this group are $\chi_y(x) = (-1)^{x \cdot y}$, where $x \cdot y := \sum_{i=1}^n x_i y_i$.

Indeed, since the irreducible characters of \mathbb{Z}_2 are

$$\begin{array}{ll} \chi_0 : 0 \mapsto 1 & \chi_1 : 0 \mapsto 1 \\ & 1 \mapsto -1 \end{array} \quad (5.9)$$

that is, $\chi_{y_i}(x_i) = (-1)^{x_i y_i}$, the irreducible characters $\chi_y(x)$ of \mathbb{Z}_2^n are the products of the irreducible characters of the individual cyclic groups by Equation (4.33).

Therefore, we have that

$$\chi_y(x) = \prod_{i=1}^n \chi_{y_i}(x_i) = \prod_{i=1}^n (-1)^{x_i y_i} = (-1)^{\sum_{i=1}^n x_i y_i} = (-1)^{x \cdot y} \quad (5.10)$$

Substituting this into Equation (5.2), the Quantum Fourier Transform for \mathbb{Z}_2^n is

$$F_{\mathbb{Z}_2^n} = \frac{1}{\sqrt{2^n}} \sum_{x, y \in \mathbb{Z}_2^n} (-1)^{x \cdot y} |y\rangle \langle x| = H^{\otimes n} \quad (5.11)$$

where the last equality is a consequence of Exercise 2.24.

Remark 5.6. Example 5.5 suggests that the Quantum Fourier Transform over an arbitrary abelian group G is the generalisation of the Hadamard operation $H^{\otimes n}$ applied to n qubits.

In particular, it allows us to create a uniform superposition over group elements when starting with a register of dimension $|G|$ in the state $|0\rangle$, because

$$F_G |0\rangle = \left(\frac{1}{\sqrt{|G|}} \sum_{x \in G} \sum_{k \in \widehat{G}} \chi_k(x) |k\rangle \langle x| \right) |0\rangle = \frac{1}{\sqrt{|G|}} \sum_{k \in G} |k\rangle \quad (5.12)$$

since $\chi_k(0) = 1$ for all $\chi_k \in \widehat{G}$.

Note that the sum is over the group elements in G instead of over the characters of \widehat{G} , but we stated in Remark 5.2 that we could write the sum in this way if we wanted to.

Also, it is worth noting that we have stated the size of the register in terms of its dimension as a vector space as opposed to giving its size in terms of the number of qubits that it is formed from. This is because it is often not possible to provide an implementation, that is, draw a quantum circuit, of the quantum algorithms that follow where we need to create superpositions of group elements. Describing the size in terms of qubits in these situations would be rather meaningless.

Despite this, if $|G| = 2^n$ for some $n \in \mathbb{Z}_{\geq 0}$, then we know that it is possible to provide an implementation in terms of n qubits, as we have shown explicitly with the quantum circuit given in Equation (3.10).

5.2 Quantum Fourier Transform over a General Finite Abelian Group

We want to be able to implement the Quantum Fourier Transform over an arbitrary cyclic group \mathbb{Z}_N

$$F_{\mathbb{Z}_N} = \frac{1}{\sqrt{N}} \sum_{t,y \in \mathbb{Z}_N} \omega_N^{ty} |y\rangle \langle t| \quad (5.13)$$

The quantum circuit given in Equation (3.10) is only a valid implementation of the Quantum Fourier Transform when N is a power of 2.

But we are able to provide an approximate implementation of $F_{\mathbb{Z}_N}$, as follows:

Let U be the unitary operator which subtracts 1 mod N from each basis element of \mathbb{Z}_N , that is,

$$U := \sum_{z \in \mathbb{Z}_N} |z-1\rangle \langle z| \quad (5.14)$$

Then the eigenstates of U are the states $\{F_{\mathbb{Z}_N} |x\rangle \mid x \in \mathbb{Z}_N\}$ with corresponding eigenvalues $\{\omega_N^x\}$, since

$$\begin{aligned} U(F_{\mathbb{Z}_N} |x\rangle) &= U\left(\frac{1}{\sqrt{N}} \sum_{y \in \mathbb{Z}_N} \omega_N^{xy} |y\rangle\right) \\ &= \left(\sum_{z \in \mathbb{Z}_N} |z-1\rangle \langle z|\right) \left(\frac{1}{\sqrt{N}} \sum_{y \in \mathbb{Z}_N} \omega_N^{xy} |y\rangle\right) \\ &= \frac{1}{\sqrt{N}} \sum_{y,z \in \mathbb{Z}_N} \omega_N^{xy} |z-1\rangle \langle z|y\rangle \\ &= \frac{1}{\sqrt{N}} \sum_{y \in \mathbb{Z}_N} \omega_N^{xy} |y-1\rangle \\ &= \frac{1}{\sqrt{N}} \sum_{y \in \mathbb{Z}_N} \omega_N^{x(y+1)} |y\rangle \\ &= \omega_N^x F_{\mathbb{Z}_N} |x\rangle \end{aligned}$$

Writing $|\tilde{x}\rangle := F_{\mathbb{Z}_N} |x\rangle$, we can use the Phase Estimation procedure given by Theorem 3.16 with this unitary operator U to implement the transformation T defined by

$$T : |0\rangle^{\otimes t} |\tilde{x}\rangle \rightarrow |x\rangle |\tilde{x}\rangle \quad (5.15)$$

which will give an estimate of x accurate to $n := \lceil \log_2 N \rceil$ bits if we set $t := n + \lceil \log_2(2 + \frac{1}{2\epsilon}) \rceil$.

Note that if N is a power of 2 then this will give x in exact form, as stated in Remark 3.11.

So, in order to implement the Quantum Fourier Transform over \mathbb{Z}_N approximately, we perform the following procedure:

1. Start with a quantum system in the state $|x\rangle |0\rangle^{\otimes n}$, where $x \in \mathbb{Z}_N$, such that the first register is of size $t = n + \lceil \log_2(2 + \frac{1}{2\epsilon}) \rceil$ qubits, where $n = \lceil \log_2 N \rceil$, and the second register is of size n qubits.

We can create $|x\rangle$ in the first register by applying the quantum NOT gate X to the appropriate qubits of the initial state $|0\rangle^{\otimes t}$ such that we obtain $x \in \mathbb{Z}_N$ when expressed in its binary form.

2. Apply the transformation $U_{F_{\mathbb{Z}_N}}$ to this state, using the method of reversible computation given in Lemma 2.45, which therefore maps $|x\rangle |0\rangle \mapsto |x\rangle F_{\mathbb{Z}_N} |x\rangle = |x\rangle |\tilde{x}\rangle$.
3. Run the operation of T in reverse, that is, apply

$$T^\dagger : |x\rangle |\tilde{x}\rangle \rightarrow |0\rangle^{\otimes t} |\tilde{x}\rangle \quad (5.16)$$

to erase $|x\rangle$.

4. Finally, drop the first register to obtain $|\tilde{x}\rangle$.

Since we can now implement the Quantum Fourier Transform over \mathbb{Z}_N (approximately), we can also implement the Quantum Fourier Transform over an arbitrary finite abelian group G (approximately).

As G is abelian, we know that G is isomorphic to a direct product of cyclic groups

$$G \cong \mathbb{Z}_{N_1} \times \mathbb{Z}_{N_2} \times \cdots \times \mathbb{Z}_{N_r} \quad (5.17)$$

Moreover,

$$\mathbb{C}[G] \cong \mathbb{C}[\mathbb{Z}_{N_1} \times \mathbb{Z}_{N_2} \times \cdots \times \mathbb{Z}_{N_r}] \cong \mathbb{C}[\mathbb{Z}_{N_1}] \otimes \mathbb{C}[\mathbb{Z}_{N_2}] \otimes \cdots \otimes \mathbb{C}[\mathbb{Z}_{N_r}] \quad (5.18)$$

is an isomorphism of Hilbert spaces over \mathbb{C} because we can map one computational basis to another by

$$|(x_1, x_2, \dots, x_r)\rangle \mapsto |x_1\rangle \otimes |x_2\rangle \otimes \cdots \otimes |x_r\rangle \quad (5.19)$$

for $x_i \in \mathbb{Z}_{N_i}$.

Therefore, the Quantum Fourier Transform over a direct product of cyclic groups is simply the tensor product of the Quantum Fourier Transforms over the individual groups, that is

$$F_G = \bigotimes_{i=1}^r F_{\mathbb{Z}_{N_i}} \quad (5.20)$$

6 The Abelian Hidden Subgroup Problem

6.1 A Procedure for the Abelian Hidden Subgroup Problem

We are now in a position to consider the Hidden Subgroup Problem in the case where the group G is a finite abelian group. As for the previous chapter, we find our motivation for this chapter in [3], [4] and [8]. We state the definition of the Hidden Subgroup Problem once again, but this time in the case where G is a finite abelian group:

Definition 6.1 (The Abelian Hidden Subgroup Problem). Let f be a function $f : G \rightarrow S$ from a finite abelian group G to a finite set S which can be computed efficiently.

In addition, suppose that f has the following property on some unknown subgroup $H \leq G$:

$$f(g) = f(g') \iff g^{-1}g' \in H$$

The goal is to find H , or equivalently, to find a generating set for H .

Remark 6.2. It is worth noting that H must be abelian in this case of the Hidden Subgroup Problem.

We will continue to use additive notation to denote binary operations on group elements of the abelian group G .

We can construct a procedure to find H , as follows:

1. Start with a quantum system in the state $|0\rangle|0\rangle$
 - where the first register has dimension $|G|$ and the second register has dimension $\#S$ (which is possible as both G and S are finite).
2. Apply F_G to the first register:
 - Since $\chi_k(0) = 1 \in \mathbb{C}$ for all $\chi_k \in \widehat{G}$, and $G \cong \widehat{\widehat{G}}$ by Theorem 4.40, we can sum over the elements of G rather than over the irreducible characters of G , obtaining a uniform superposition of the elements of G :

$$\frac{1}{\sqrt{|G|}} \sum_{g \in G} |g\rangle |0\rangle \tag{6.1}$$

3. Apply the “black-box” unitary transformation $U_f : |g\rangle |s\rangle \mapsto |g\rangle |s \oplus f(g)\rangle$ to this state, where \oplus is an appropriate binary function for the set S such that $0 \oplus s = s$ for all $s \in S$, giving

$$\frac{1}{\sqrt{|G|}} \sum_{g \in G} |g\rangle |f(g)\rangle \tag{6.2}$$

4. Measure the second register and drop it: this gives some value $f(x)$ for some $x \in G$, and collapses the first register to a uniform superposition over the elements of G that have the same value in S under f , namely the elements of the coset $x + H$.

$$\frac{1}{\sqrt{|H|}} \sum_{h \in H} |x + h\rangle \tag{6.3}$$

This state is also known as a “coset state”, which we can denote by $|x + H\rangle$.

5. Now apply F_G to this state, which gives:

$$\begin{aligned}
F_G |x + H\rangle &= \left(\frac{1}{\sqrt{|G|}} \sum_{t \in G} \sum_{k \in \hat{G}} \chi_k(t) |k\rangle \langle t| \right) \left(\frac{1}{\sqrt{|H|}} \sum_{h \in H} |x + h\rangle \right) \\
&= \frac{1}{\sqrt{|G||H|}} \sum_{h \in H} \sum_{k \in \hat{G}} \chi_k(x + h) |k\rangle \\
&= \sqrt{\frac{|H|}{|G|}} \sum_{k \in \hat{G}} \chi_k(x) \left[\frac{1}{|H|} \sum_{h \in H} \chi_k(h) \right] |k\rangle
\end{aligned} \tag{6.4}$$

We claim that

$$\frac{1}{|H|} \sum_{h \in H} \chi_k(h) = \begin{cases} 1 & \text{if } \chi_k \in H^\perp \\ 0 & \text{if } \chi_k \notin H^\perp \end{cases} \tag{6.5}$$

Proof. Indeed, if $\chi_k \in H^\perp$, that is, $\chi_k(h) = 1$ for all $h \in H$, then

$$\frac{1}{|H|} \sum_{h \in H} \chi_k(h) = \frac{1}{|H|} \sum_{h \in H} 1 = 1 \tag{6.6}$$

If $\chi_k \notin H^\perp$, then $\chi_k(h_0) \neq 1$ for some $h_0 \in H$. Then

$$\chi_k(h_0) \left(\frac{1}{|H|} \sum_{h \in H} \chi_k(h) \right) = \frac{1}{|H|} \sum_{h \in H} \chi_k(h_0 + h) = \frac{1}{|H|} \sum_{h \in h_0 + H} \chi_k(h) = \frac{1}{|H|} \sum_{h \in H} \chi_k(h) \tag{6.7}$$

since $h_0 + H = H$, and so we have that

$$\frac{1}{|H|} \sum_{h \in H} \chi_k(h) = 0 \tag{6.8}$$

as required. \square

Applying this result to Equation (6.4) gives us that

$$F_G |x + H\rangle = \sqrt{\frac{|H|}{|G|}} \sum_{k: \chi_k \in H^\perp} \chi_k(x) |k\rangle \tag{6.9}$$

which is a uniform superposition of the elements of H^\perp , as $|H^\perp| = [G : H]$ by Theorem 4.46 and $|\chi_k(x)| = 1$ for all $\chi_k \in H^\perp, x \in G$ by Example 4.34.

6. Perform a measurement of this register in the computational basis. This gives some character χ_k that maps the hidden subgroup H to 1.

We can then consider the following subgroup of G :

$$\ker(\chi_k) := \{g \in G : \chi_k(g) = 1\} \tag{6.10}$$

which is very useful since $H \subseteq \ker(\chi_k)$ and it restricts what can possibly be in H .

We claim that if we repeat this procedure $O(\log |G|)$ times, then H is the intersection of the kernels we obtain from each iteration with very high probability (almost 1).

Proof. Suppose we are about to begin a new iteration of this procedure. We can denote the intersection of the kernels that we've obtained up to this point as K , which is a subgroup of G that we assume is not H . (If it were H then we wouldn't begin this new iteration.)

We want to find the probability of measuring a χ_k that helps in obtaining H , which we do by looking at the probability of measuring one that does not help.

The probability that we measure a χ_k such that $K \leq \ker(\chi_k)$ in Step 6 is

$$\frac{|H|}{|G|} \times \#\{\chi_k \in \widehat{G} \mid K \leq \ker(\chi_k)\} \quad (6.11)$$

by the uniformity of the superposition given in Equation (6.9).

But because

$$\{\chi_k \in \widehat{G} \mid K \leq \ker(\chi_k)\} = \{\chi_k \in \widehat{G} \mid \chi_k(x) = 1 \forall x \in K\} = K^\perp \quad (6.12)$$

and $|K^\perp| = \frac{|G|}{|K|}$ by Theorem 4.46, we have that

$$P(\text{Measuring a } \chi_k \text{ such that } K \leq \ker(\chi_k)) = \frac{|H|}{|G|} \frac{|G|}{|K|} = \frac{|H|}{|K|} \leq \frac{1}{2} \quad (6.13)$$

with the last equality coming from Lagrange's Theorem (as $H < K, H \neq K$).

Therefore, the probability of observing a χ_k such that $K \not\leq \ker(\chi_k)$ is at least $\frac{1}{2}$.

Moreover, for this χ_k , $K \cap \ker(\chi_k)$ will be a proper subgroup of K , and so, by Lagrange's Theorem again, we have that

$$|K \cap \ker(\chi_k)| \leq \frac{|K|}{2} \quad (6.14)$$

This means that our newly updated subgroup of the intersection of all the kernels obtained up to the end of this iteration, $K \cap \ker(\chi_k)$, is at least half the size of the subgroup at the beginning of the iteration, and it contains H .

Therefore, if we perform $O(\log |G|)$ iterations of this procedure, we will obtain H almost certainly. As a result, this procedure is an efficient quantum algorithm that solves the Abelian Hidden Subgroup Problem. \square

Remark 6.3. In order to use this procedure to solve applications of the HSP, we need to be able to

- provide implementations of quantum states of dimension $|G|$ and $\#S$ respectively in order to prepare the quantum system as given in Step 1.
- implement the transformation U_f that was merely a "black-box" in this procedure, and provide an appropriate binary operation \oplus on elements of S .

One such application is the following problem:

Example 6.4 (Reformulation of the Order-Finding Problem). We saw that the Order-Finding Problem considered $x \in \mathbb{Z}_N^*$ with the goal of finding the smallest positive integer r such that $x^r \equiv 1 \pmod{N}$.

We can reformulate this problem as an example of the Abelian Hidden Subgroup Problem, as follows:

Let G be the abelian group $(\mathbb{Z}, +)$, and let S be the finite set $\{x^j \mid j \in \mathbb{Z}_r \text{ and } x^r = 1\}$.

Then we can let $f : G \rightarrow S$ be the function which maps $j \mapsto x^j$, and f has the property that

$$f(j) = f(j') \iff j \equiv j' \pmod{r} \iff j - j' \in r\mathbb{Z} \quad (6.15)$$

Our goal, therefore, is to find $H := r\mathbb{Z}$, which is a subgroup that has a single generator, the solution to the original problem.

However, in order to use the procedure above, we need to set the first register up such that it has dimension $|G|$. Clearly, $(\mathbb{Z}, +)$ is an infinite group.

Our first thought might be to restrict the domain of f to the group $(\mathbb{Z}_{\phi(N)}, +)$ because we know that $r \mid \phi(N)$ and so $r \leq \phi(N)$. The uniform superposition in Step 5 would be over the elements of H^\perp , which, in this case, has the form

$$H^\perp = \left\{ \chi_k \in \widehat{\mathbb{Z}_{\phi(N)}} \mid e^{\frac{2\pi i k h}{\phi(N)}} = 1 \text{ for all } h \in H \right\} = \left\{ \chi_k \in \widehat{\mathbb{Z}_{\phi(N)}} \mid \frac{kr}{\phi(N)} \in \{0, 1, \dots, r-1\} \right\} \quad (6.16)$$

since $h \in H = r\mathbb{Z}$.

Step 6 would therefore give a k that is some integer multiple of $\frac{\phi(N)}{r}$.

However, this approach is problematic, because we do not know how to implement the Quantum Fourier Transform for $\mathbb{Z}_{\phi(N)}$ exactly, only approximately. It is also not assumed that the algorithm has knowledge of the value of $\phi(N)$ in the first place.

But we do know how to implement the Quantum Fourier Transform exactly for the abelian group $G := (\mathbb{Z}_{2^n}, +)$ for some $n \in \mathbb{Z}$, which we have shown before in Equation (3.10). If we choose n such that $N^2 \leq 2^n \leq 2N^2$, then it can be shown that by applying the procedure for the Abelian Hidden Subgroup Problem for this G we will obtain a k that is “very close” to $\frac{2^n}{r}$. We can then apply a result that is very similar to Theorem 3.27 to obtain r by the Continued Fractions Algorithm.

We won’t look into the details of this calculation because they are very similar to those given for the original Order-Finding Problem of Section 3.3. However, they are provided in full in [8, p. 37].

Consequently, we can see that Shor’s Factoring Algorithm is a specific instance of this algorithm to solve the Abelian Hidden Subgroup Problem, by the one-to-one link between order-finding and factoring composite integers as seen in Section 3.4.

6.2 The Discrete Logarithm Problem

We are able to use the Abelian Hidden Subgroup Problem procedure to solve another problem that we have not seen before, called the “Discrete Logarithm Problem”, which is stated as follows:

- Problem: Suppose that we have a cyclic group (C_N, \times) with a generator g of order N , which is known. Suppose that we consider an element x in this group.
- Goal: to find the smallest non-negative integer r such that $g^r = x$.

We can formulate this problem as an instance of the Abelian Hidden Subgroup Problem, following [3]:

Let G be the group $(\mathbb{Z}_N \times \mathbb{Z}_N, +)$, and let S be the group C_N , considered as a finite set of elements.

Then we can define a function $f : \mathbb{Z}_N \times \mathbb{Z}_N \rightarrow C_N$ which maps

$$(\alpha, \beta) \mapsto x^\alpha g^\beta \quad (6.17)$$

Now, since $g^r = x$ for some non-negative integer r , we also have that $f(\alpha, \beta) = g^{r\alpha + \beta}$, and so f is a constant function on the sets where $r\alpha + \beta$ is some constant γ for varying pairs $(\alpha, \beta) \in \mathbb{Z}_N \times \mathbb{Z}_N$.

We can index these sets by the constant γ itself, namely

$$S_\gamma = \{(\alpha, \beta) \in \mathbb{Z}_N \times \mathbb{Z}_N \mid r\alpha + \beta = \gamma\} \quad (6.18)$$

Hence

$$f(\alpha, \beta) = f(\alpha', \beta') \iff r\alpha + \beta = r\alpha' + \beta' \iff (\alpha - \alpha', \beta - \beta') \in S_0 \quad (6.19)$$

and so $S_0 = \{(\alpha, -r\alpha) \mid \alpha \in \mathbb{Z}_N\}$ is the hidden subgroup H that we are looking to find.

We now apply the procedure for the Abelian Hidden Subgroup Problem as follows:

1. We start in the state $|0\rangle |0\rangle |0\rangle$, where the first and second registers each have dimension N (as $|\mathbb{Z}_N| = N$) and the third register has dimension $\#C_N = N$ also.
 - We have used Equation (5.18) here to say that $\mathbb{C}[\mathbb{Z}_N \times \mathbb{Z}_N] \cong \mathbb{C}[\mathbb{Z}_N] \otimes \mathbb{C}[\mathbb{Z}_N]$. This allows us to start with three registers, the first two for the tensored group algebras and the third for the set C_N .
2. We use Equation (5.20) to say that $F_{\mathbb{Z}_N \times \mathbb{Z}_N} = F_{\mathbb{Z}_N} \otimes F_{\mathbb{Z}_N}$ and apply this operator to the first and second registers to create a uniform superposition over $\mathbb{Z}_N \times \mathbb{Z}_N$.

$$\frac{1}{N} \sum_{\alpha, \beta \in \mathbb{Z}_N} |\alpha\rangle |\beta\rangle |0\rangle \quad (6.20)$$

3. We apply the unitary transformation U_f which computes f in the third register:

$$\frac{1}{N} \sum_{\alpha, \beta \in \mathbb{Z}_N} |\alpha\rangle |\beta\rangle |f(\alpha, \beta)\rangle \quad (6.21)$$

This transformation can be implemented using the techniques of reversible computation via the repeated squaring of the group element $g \in C_N$, as seen before in Section 3.3.

4. We now measure the third register and drop it. We will observe g^δ for some δ in this register. The first two registers therefore collapse to a uniform superposition over the elements of $\mathbb{Z}_N \times \mathbb{Z}_N$ which also map to g^δ under f .

These elements are the set S_δ , but we can actually write this set as a coset of S_0 , namely the coset

$$(0, \delta) + S_0 = \{(\alpha, \delta - r\alpha) \mid \alpha \in \mathbb{Z}_N\} = S_\delta \quad (6.22)$$

Therefore, the state we are left with after measuring the third register is:

$$\frac{1}{\sqrt{N}} \sum_{\alpha \in \mathbb{Z}_N} |\alpha\rangle |\delta - r\alpha\rangle \quad (6.23)$$

5. We apply $F_{\mathbb{Z}_N} \otimes F_{\mathbb{Z}_N}$ to this state.

By Equation (5.13), we have that

$$F_{\mathbb{Z}_N} = \frac{1}{\sqrt{N}} \sum_{x,y \in \mathbb{Z}_N} \omega_N^{xy} |y\rangle \langle x| \quad (6.24)$$

and so

$$F_{\mathbb{Z}_N} \otimes F_{\mathbb{Z}_N} = \frac{1}{N} \left(\sum_{\mu,x \in \mathbb{Z}_N} \omega_N^{x\mu} |\mu\rangle \langle x| \right) \otimes \left(\sum_{\nu,y \in \mathbb{Z}_N} \omega_N^{y\nu} |\nu\rangle \langle y| \right) \quad (6.25)$$

Applying this to the state given by Equation (6.23), we have that this equals

$$\begin{aligned} &= \frac{1}{N^{\frac{3}{2}}} \sum_{\alpha \in \mathbb{Z}_N} \left(\sum_{\mu,x \in \mathbb{Z}_N} \omega_N^{x\mu} |\mu\rangle \langle x|\alpha \right) \otimes \left(\sum_{\nu,y \in \mathbb{Z}_N} \omega_N^{y\nu} |\nu\rangle \langle y|\delta - r\alpha \right) \\ &= \frac{1}{N^{\frac{3}{2}}} \sum_{\alpha,\mu,\nu \in \mathbb{Z}_N} \omega_N^{\alpha\mu + (\delta - r\alpha)\nu} |\mu\rangle |\nu\rangle \\ &= \frac{1}{N^{\frac{3}{2}}} \sum_{\mu,\nu \in \mathbb{Z}_N} \omega_N^{\delta\nu} \left[\sum_{\alpha \in \mathbb{Z}_N} \omega_N^{\alpha(\mu - r\nu)} \right] |\mu\rangle |\nu\rangle \\ &= \frac{1}{N^{\frac{3}{2}}} \sum_{\mu,\nu \in \mathbb{Z}_N} \omega_N^{\delta\nu} [N\delta_{(\mu - r\nu),0}] |\mu\rangle |\nu\rangle \end{aligned} \quad (6.26)$$

$$= \frac{1}{\sqrt{N}} \sum_{\nu \in \mathbb{Z}_N} \omega_N^{\delta\nu} |r\nu\rangle |\nu\rangle \quad (6.27)$$

where the step given in Equation (6.26) is a direct consequence of Exercise 3.23.

6. We now measure this state in the computational basis of $\mathbb{C}[\mathbb{Z}_N] \otimes \mathbb{C}[\mathbb{Z}_N]$, obtaining some pair $(r\nu, \nu)$ uniformly at random.

Since we now know ν , we can check to see if it has a multiplicative inverse mod N . If it does, then we can apply it to the result of the first register to obtain r .

If it does not, then we can perform another iteration of the procedure.

Given that there are $\phi(N)$ elements of \mathbb{Z}_N which have a multiplicative inverse mod N , each iteration has a probability of success $\frac{\phi(N)}{N}$.

This is $\Omega\left(\frac{1}{\log \log N}\right)$ - see [16] for details. Hence we will obtain r with a few iterations, at most.

7 The General Hidden Subgroup Problem

We now look at a method that provides an insight into the open problem of finding an efficient quantum algorithm for the most general form of the Hidden Subgroup Problem: the case where the group G is not abelian. It generalises the approach used to solve the problem when the group is abelian by using a sampling method called weak Fourier sampling. Weak Fourier sampling applies a set of measurement operators indexed by the elements of \widehat{G} to a block diagonal density operator that represents a coset state in $\mathbb{C}[G]$ which has been transformed by the Quantum Fourier Transform to its equivalent state in an isomorphic complex Hilbert space. We show that this method, sometimes called the “standard method”, solves the Hidden Subgroup Problem in the general case efficiently when the hidden subgroup H is normal in G , a result of [15]. However, we follow the presentation given in [3], filling out many of the details that they left out in their paper. To begin, we need another improved version of the Quantum Fourier Transform.

7.1 Quantum Fourier Transform, version 3

Definition 7.1 (Quantum Fourier Transform). Let G be any finite group of size $|G|$.

Consider the group algebra $\mathbb{C}[G]$ as a quantum system over \mathbb{C} with the computational basis labelled by the elements of G , $\{|x\rangle \mid x \in G\}$. This vector space is augmented with the inner product $\langle *|* \rangle : \mathbb{C}[G] \times \mathbb{C}[G] \rightarrow \mathbb{C}$ that is defined on the computational basis by $\langle i|j \rangle = \delta_{ij}$ and then extended conjugate-linearly in the first argument and linearly in the second argument to the whole of $\mathbb{C}[G] \times \mathbb{C}[G]$.

Then G has a set of irreducible representations $\sigma : G \rightarrow GL(\mathbb{C}^{d_\sigma})$, each of dimension d_σ , satisfying

$$\sum_{\sigma \in \widehat{G}} d_\sigma^2 = |G| \quad (7.1)$$

by Corollary 4.26, and for all $x \in G$, $\sigma(x) : \mathbb{C}^{d_\sigma} \rightarrow \mathbb{C}^{d_\sigma}$ is a unitary map of dimension d_σ^2 by Remark 4.5.

We now consider the complex Hilbert space $\bigoplus_{\sigma \in \widehat{G}} \mathbb{C}^{d_\sigma^2}$, which is a direct sum of complex Hilbert spaces over the irreducible representations of G , with its inner product $\langle *|* \rangle : \bigoplus_{\sigma \in \widehat{G}} \mathbb{C}^{d_\sigma^2} \times \bigoplus_{\sigma \in \widehat{G}} \mathbb{C}^{d_\sigma^2} \rightarrow \mathbb{C}$ defined to be

$$\langle f|g \rangle := \sum_{\sigma \in \widehat{G}} \langle f_\sigma|g_\sigma \rangle_\sigma \quad (7.2)$$

for all $|f\rangle, |g\rangle \in \bigoplus_{\sigma \in \widehat{G}} \mathbb{C}^{d_\sigma^2}$, where $\langle *|* \rangle_\sigma$ is the Frobenius inner product for the complex Hilbert space $\mathbb{C}^{d_\sigma^2}$ given by

$$\langle A|B \rangle_\sigma := \text{tr}(A^\dagger B) \quad (7.3)$$

for all matrices $A, B \in \mathbb{C}^{d_\sigma^2}$.

The complex Hilbert space $\mathbb{C}^{d_\sigma^2}$ clearly has dimension d_σ^2 , and so the direct sum complex Hilbert space $\bigoplus_{\sigma \in \widehat{G}} \mathbb{C}^{d_\sigma^2}$ has dimension $|G|$ by Equation (7.1).

If we fix some element $x \in G$, then for each irreducible $\sigma \in \widehat{G}$ and unitary map $\sigma(x) : \mathbb{C}^{d_\sigma} \rightarrow \mathbb{C}^{d_\sigma}$, we can choose bases of both \mathbb{C}^{d_σ} . Then, for these bases, the $\sigma(x)$ become $d_\sigma \times d_\sigma$ unitary matrices over \mathbb{C} , that is, elements of $\mathbb{C}^{d_\sigma^2}$. (For the rest of this definition, we will abuse our notation by considering the $\sigma(x)$ as unitary matrices for the bases chosen.)

So if there are n irreducible representations of G , we can use the index set $\{1, 2, \dots, n\}$ to enumerate them and say that $\bigoplus_{\sigma \in \widehat{G}} \mathbb{C}^{d_\sigma^2} = \bigoplus_{j=1}^n \mathbb{C}^{d_{\sigma_j}^2}$. We can then consider the following element in this Hilbert space:

$$|\tilde{x}\rangle := \left(\frac{d_{\sigma_1}}{\sqrt{|G|}} |\sigma_1(x)\rangle, \frac{d_{\sigma_2}}{\sqrt{|G|}} |\sigma_2(x)\rangle, \dots, \frac{d_{\sigma_n}}{\sqrt{|G|}} |\sigma_n(x)\rangle \right) \quad (7.4)$$

where we have labelled each of the states $|\sigma_i(x)\rangle$ as unit vectors in $\mathbb{C}^{d_{\sigma_i}^2}$, that is, they correspond to the matrix $\frac{1}{\sqrt{d_{\sigma_i}}} \sigma_i(x)$.

The $|\sigma_i(x)\rangle$ are unit vectors in $\mathbb{C}^{d_{\sigma_i}^2}$ because the Frobenius inner product of the matrix $\sigma_i(x)$ with itself in $\mathbb{C}^{d_{\sigma_i}^2}$ is equal to the trace of $\sigma_i(x)^\dagger \sigma_i(x)$ by Equation (7.3), which equals the trace of $I_{d_{\sigma_i}}$ (the $d_{\sigma_i} \times d_{\sigma_i}$ identity matrix) as $\sigma_i(x)$ is unitary; that is, the trace is d_{σ_i} . Hence, for the state $|\sigma_i(x)\rangle$, $\langle \sigma_i(x) | \sigma_i(x) \rangle_{\sigma_i} = 1$.

The norm of $|\tilde{x}\rangle \in \bigoplus_{j=1}^n \mathbb{C}^{d_{\sigma_j}^2}$ is 1 as a result of applying the definition of the inner product given in Equation (7.2) to $\langle \tilde{x} | \tilde{x} \rangle$, using the fact that the $|\sigma_i(x)\rangle$ are unit vectors in $\mathbb{C}^{d_{\sigma_i}^2}$ and using the equality given in Equation (7.1).

We now define $|\sigma_i\rangle \in \bigoplus_{j=1}^n \mathbb{C}^{d_{\sigma_j}^2}$ to be the element that has entry $I_{d_{\sigma_i}}$ in the i th position of Equation (7.4), and the zero matrix of the appropriate dimension otherwise.

Therefore, by the uniqueness of representing elements of a direct sum vector space, we can rewrite the element $|\tilde{x}\rangle$ as

$$|\tilde{x}\rangle = \sum_{i=1}^n \frac{d_{\sigma_i}}{\sqrt{|G|}} |\sigma_i\rangle |\sigma_i(x)\rangle = \sum_{\sigma \in \widehat{G}} \frac{d_\sigma}{\sqrt{|G|}} |\sigma\rangle |\sigma(x)\rangle \quad (7.5)$$

As a result, we can define a linear operator F_G on the computational basis of $\mathbb{C}[G]$ as follows:

$$\begin{aligned} F_G : \mathbb{C}[G] &\rightarrow \bigoplus_{\sigma \in \widehat{G}} \mathbb{C}^{d_\sigma^2} \\ |x\rangle &\mapsto |\tilde{x}\rangle = \sum_{\sigma \in \widehat{G}} \frac{d_\sigma}{\sqrt{|G|}} |\sigma\rangle |\sigma(x)\rangle \end{aligned} \quad (7.6)$$

and extend linearly over linear combinations of computational basis elements.

It is often useful to apply the isomorphism

$$\mathbb{C}^{d_\sigma^2} \cong \mathbb{C}^{d_\sigma} \otimes \mathbb{C}^{d_\sigma} \quad (7.7)$$

by rewriting the state of each $|\sigma(x)\rangle$ as

$$|\sigma(x)\rangle = \sum_{j=1}^{d_\sigma} \sum_{k=1}^{d_\sigma} \frac{\sigma(x)_{j,k}}{\sqrt{d_\sigma}} |j\rangle |k\rangle \quad (7.8)$$

since each of the states $|\sigma(x)\rangle$ are normalised in $\mathbb{C}^{d_\sigma^2}$ and correspond to the matrix $\frac{\sigma(x)}{\sqrt{d_\sigma}}$.

Therefore, we can say that

$$\bigoplus_{\sigma \in \widehat{G}} \mathbb{C}^{d_\sigma^2} \cong \bigoplus_{\sigma \in \widehat{G}} (\mathbb{C}^{d_\sigma} \otimes \mathbb{C}^{d_\sigma}) \quad (7.9)$$

consider F_G as a linear operator

$$F_G : \mathbb{C}[G] \rightarrow \bigoplus_{\sigma \in \widehat{G}} (\mathbb{C}^{d_\sigma} \otimes \mathbb{C}^{d_\sigma}) \quad (7.10)$$

and write $|\tilde{x}\rangle$ as

$$|\tilde{x}\rangle = \sum_{\sigma \in \widehat{G}} \sqrt{\frac{d_\sigma}{|G|}} \sum_{j,k=1}^{d_\sigma} \sigma(x)_{j,k} |\sigma\rangle |j\rangle |k\rangle \quad (7.11)$$

Remark 7.2. From this definition, we can look at what F_G does when G is abelian.

Since $d_\sigma = 1$ for all irreducible representations $\sigma \in \widehat{G}$, we have that

$$\bigoplus_{\sigma \in \widehat{G}} \mathbb{C}^{d_\sigma^2} = \bigoplus_{\sigma \in \widehat{G}} \mathbb{C} \cong \mathbb{C}[\widehat{G}] \quad (7.12)$$

where the isomorphism holds via a mapping of the basis $|\sigma_i\rangle \in \bigoplus_{\sigma \in \widehat{G}} \mathbb{C} \mapsto |\sigma_i\rangle \in \mathbb{C}[\widehat{G}]$ for some enumeration i of the one-dimensional irreducible representations of G .

Since σ is a representation of dimension 1, we must have that $|\sigma(x)| = 1$ for all $x \in G$, and because $\sigma(x) = \chi_\sigma(x)$ in this case by Remark 4.36, we can consider $|\sigma(x)\rangle$ instead to be the complex scalar $\chi_\sigma(x) \in \mathbb{S}^1 \subset \mathbb{C}$, which merely alters the phase of the state $|\sigma\rangle$.

Hence F_G maps the computational basis of $\mathbb{C}[G]$ as follows:

$$\begin{aligned} F_G : \mathbb{C}[G] &\rightarrow \mathbb{C}[\widehat{G}] \\ |x\rangle &\mapsto \frac{1}{\sqrt{|G|}} \sum_{\sigma \in \widehat{G}} \chi_\sigma(x) |\sigma\rangle \end{aligned} \quad (7.13)$$

which is exactly the same as Equation (5.1), as expected.

Remark 7.3. Using the notation of Equation (7.11), we can write F_G using the outer product notation

$$F_G = \sum_{x \in G} |\tilde{x}\rangle \langle x| = \sum_{x \in G} \sum_{\sigma \in \widehat{G}} \sqrt{\frac{d_\sigma}{|G|}} \sum_{j,k=1}^{d_\sigma} \sigma(x)_{j,k} |\sigma\rangle |j\rangle |k\rangle \langle x| \quad (7.14)$$

which is not unique because we have chosen a basis for each unitary map $\sigma(x)$, for all $x \in G$.

Exercise 7.4. We still need to show that F_G is a unitary transformation.

Proof. We show that $F_G^\dagger F_G = I$.

It is clear from Equation (7.14) that $F_G^\dagger = (\sum_{x \in G} |\tilde{x}\rangle \langle x|)^\dagger = \sum_{x \in G} |x\rangle \langle \tilde{x}|$.

Therefore

$$\begin{aligned} F_G^\dagger F_G &= \sum_{x,y \in G} |y\rangle \langle \tilde{y}| \langle \tilde{x}| \langle x| \\ &= \sum_{x,y \in G} \sum_{\sigma, \tilde{\sigma} \in \widehat{G}} \frac{d_\sigma^* d_\sigma}{|G|} \langle \tilde{\sigma} | \sigma \rangle \langle \tilde{\sigma}(y) | \sigma(x) \rangle |y\rangle \langle x| \\ &= \sum_{x,y \in G} \sum_{\sigma \in \widehat{G}} \frac{d_\sigma^2}{|G|} \langle \sigma(y) | \sigma(x) \rangle |y\rangle \langle x| \end{aligned} \quad (7.15)$$

where we have used that $d_\sigma \in \mathbb{R}$ and the orthogonality of irreducible representations in Equation (7.15).

We claim that $\langle \sigma(y) | \sigma(x) \rangle = \frac{\chi_\sigma(y^{-1}x)}{d_\sigma}$

Indeed, we have that

$$\begin{aligned} \langle \sigma(y) | \sigma(x) \rangle &= \sum_{j,k=1}^{d_\sigma} \sum_{l,m=1}^{d_\sigma} \frac{\overline{\sigma(y)_{j,k}}}{\sqrt{d_\sigma}} \frac{\sigma(x)_{l,m}}{\sqrt{d_\sigma}} \langle j | l \rangle \langle k | m \rangle \\ &= \sum_{j,k=1}^{d_\sigma} \frac{\overline{\sigma(y)_{j,k}} \sigma(x)_{j,k}}{d_\sigma} \\ &= \frac{1}{d_\sigma} \sum_{k=1}^{d_\sigma} \left[\sum_{j=1}^{d_\sigma} \sigma^\dagger(y)_{k,j} \sigma(x)_{j,k} \right] \end{aligned} \quad (7.16)$$

$$\begin{aligned} &= \frac{1}{d_\sigma} \sum_{k=1}^{d_\sigma} (\sigma^\dagger(y) \sigma(x))_{k,k} \\ &= \frac{\text{tr}(\sigma^\dagger(y) \sigma(x))}{d_\sigma} \\ &= \frac{\text{tr}(\sigma(y^{-1}x))}{d_\sigma} \end{aligned} \quad (7.17)$$

$$= \frac{\chi_\sigma(y^{-1}x)}{d_\sigma} \quad (7.18)$$

where we have used the definition of unitary operators in Equation (7.16), and used the fact that we have restricted ourselves to the use of unitary representations by Remark 4.5 to obtain Equation (7.17).

Substituting Equation (7.18) into Equation (7.15) gives

$$F_G^\dagger F_G = \sum_{x,y \in G} \sum_{\sigma \in \widehat{G}} \frac{d_\sigma}{|G|} \chi_\sigma(y^{-1}x) |y\rangle \langle x| \quad (7.19)$$

We now consider

$$\sum_{\sigma \in \widehat{G}} \frac{d_\sigma}{|G|} \chi_\sigma(y^{-1}x) \quad (7.20)$$

by splitting into cases.

If $x = y$, then $\chi_\sigma(y^{-1}x) = \chi_\sigma(1) = d_\sigma$, and so Equation (7.20) is equal to

$$\sum_{\sigma \in \widehat{G}} \frac{d_\sigma^2}{|G|} = 1 \quad (7.21)$$

If $x \neq y$, then $\chi_\sigma(y^{-1}x) = \chi_\sigma(z)$ for some $z \in G, z \neq 1$, and so Equation (7.20) becomes

$$\frac{1}{|G|} \sum_{\sigma \in \widehat{G}} d_\sigma \chi_\sigma(z) = 0 \quad (7.22)$$

by Lemma 4.27.

Hence Equation (7.20) equals δ_{xy} , and so Equation (7.19) becomes

$$F_G^\dagger F_G = \sum_{x,y \in G} \delta_{xy} |y\rangle \langle x| = \sum_{x \in G} |x\rangle \langle x| = I \quad (7.23)$$

and so F_G is unitary, as required. \square

Remark 7.5. In particular, Exercise 7.4 shows that

$$\mathbb{C}[G] \cong \bigoplus_{\sigma \in \widehat{G}} \mathbb{C}^{d_\sigma^2} \cong \bigoplus_{\sigma \in \widehat{G}} (\mathbb{C}^{d_\sigma} \otimes \mathbb{C}^{d_\sigma}) \quad (7.24)$$

since F_G is a unitary transformation between complex Hilbert spaces of the same dimension.

An important result for our purposes is the following:

Proposition 7.6. F_G is the G -linear map that results in an isomorphism of representations

$$L \cong \bigoplus_{\sigma \in \widehat{G}} (\sigma \otimes I_{d_\sigma}) \quad R \cong \bigoplus_{\sigma \in \widehat{G}} (I_{d_\sigma} \otimes \sigma^*) \quad (7.25)$$

where L denotes the left regular representation of G , and R denotes the right regular representation of G .

Proof. We show this for the right regular representation R . See [3, p. 50] for the proof of the left regular representation L .

Consider $R(x)$ as a map $\mathbb{C}[G] \rightarrow \mathbb{C}[G]$, instead of as a map $V_R \rightarrow V_R$, for all $x \in G$. We can do this because $\mathbb{C}[G]$ and V_R are isomorphic by Definition 4.29.

Let us define $\tilde{R}(x) := F_G R(x) F_G^\dagger : \bigoplus_{\sigma \in \widehat{G}} \mathbb{C}^{d_\sigma^2} \rightarrow \bigoplus_{\sigma \in \widehat{G}} \mathbb{C}^{d_\sigma^2}$.

Since F_G is an isomorphism between $\mathbb{C}[G]$ and $\bigoplus_{\sigma \in \widehat{G}} \mathbb{C}^{d_\sigma^2}$ by Remark 7.5, by Definition 4.6, it is enough to show that $\tilde{R}(x) = \bigoplus_{\sigma \in \widehat{G}} (I_{d_\sigma} \otimes \sigma^*(x))$.

Since $R(x) |y\rangle = |yx^{-1}\rangle$, we have that

$$\begin{aligned} \tilde{R}(x) &= F_G R(x) F_G^\dagger \\ &= \left(\sum_{z \in G} |\tilde{z}\rangle \langle z| \right) R(x) \left(\sum_{y \in G} |y\rangle \langle \tilde{y}| \right) \\ &= \sum_{z,y \in G} |\tilde{z}\rangle \langle z| yx^{-1} \rangle \langle \tilde{y}| \\ &= \sum_{y \in G} |\widetilde{yx^{-1}}\rangle \langle \tilde{y}| \end{aligned}$$

Substituting in, using Equation (7.14), we get that

$$\tilde{R}(x) = \sum_{y \in G} \sum_{\sigma, \tilde{\sigma} \in \widehat{G}} \sum_{j,k=1}^{d_\sigma} \sum_{l,m=1}^{d_{\tilde{\sigma}}} \frac{\sqrt{d_\sigma d_{\tilde{\sigma}}}}{|G|} \sigma(yx^{-1})_{j,k} \tilde{\sigma}(y)_{l,m}^* |\sigma\rangle |j\rangle |k\rangle \langle \tilde{\sigma}| \langle l| \langle m| \quad (7.26)$$

Since $\sigma(yx^{-1})_{j,k} = \sum_{p=1}^{d_\sigma} \sigma(y)_{j,p} \sigma(x^{-1})_{p,k}$, this gives

$$\tilde{R}(x) = \sum_{y \in G} \sum_{\sigma, \tilde{\sigma} \in \hat{G}} \sum_{j,k,p=1}^{d_\sigma} \sum_{l,m=1}^{d_{\tilde{\sigma}}} \frac{\sqrt{d_\sigma d_{\tilde{\sigma}}}}{|G|} \sigma(y)_{j,p} \sigma(x^{-1})_{p,k} \tilde{\sigma}(y)_{l,m}^* |\sigma\rangle |j\rangle |k\rangle \langle \tilde{\sigma}| \langle l| \langle m| \quad (7.27)$$

By the orthogonality of irreducible representations given by Theorem 4.30,

$$\frac{d_\sigma}{|G|} \sum_{y \in G} \sigma(y)_{j,p} \tilde{\sigma}(y)_{l,m}^* = \delta_{\sigma, \tilde{\sigma}} \delta_{j,l} \delta_{p,m} \quad (7.28)$$

where $\delta_{\sigma, \tilde{\sigma}} = 1$ if σ is an isomorphic representation to $\tilde{\sigma}$, and 0 otherwise.

Substituting this into Equation (7.27) gives

$$\tilde{R}(x) = \sum_{\sigma \in \hat{G}} \sum_{j,k,p=1}^{d_\sigma} \sigma(x^{-1})_{p,k} |\sigma\rangle |j\rangle |k\rangle \langle \sigma| \langle j| \langle p| \quad (7.29)$$

Now, since

$$\sigma(x^{-1})_{p,k} = \sigma(x)_{p,k}^{-1} = \sigma(x)_{p,k}^\dagger = \sigma(x)_{k,p}^* = \sigma^*(x)_{k,p} \quad (7.30)$$

we can rewrite Equation (7.29) as

$$\tilde{R}(x) = \sum_{\sigma \in \hat{G}} \left(|\sigma\rangle \langle \sigma| \otimes \left(\left[\sum_{j=1}^{d_\sigma} |j\rangle \langle j| \right] \otimes \left[\sum_{k=1}^{d_\sigma} \sum_{p=1}^{d_\sigma} \sigma^*(x)_{k,p} |k\rangle \langle p| \right] \right) \right) \quad (7.31)$$

The right hand side is precisely

$$\bigoplus_{\sigma \in \hat{G}} (I_{d_\sigma} \otimes \sigma^*(x)) \quad (7.32)$$

and so we are done. \square

Remark 7.7. F_G is the invertible linear transformation that makes R and $\bigoplus_{\sigma \in \hat{G}} (I_{d_\sigma} \otimes \sigma^*)$ isomorphic representations.

Since $F_G = \sum_{x \in G} |\tilde{x}\rangle \langle x|$, Proposition 7.6 shows that F_G changes the basis of R from the computational basis of $\mathbb{C}[G]$ into the Fourier basis $\{|\tilde{x}\rangle \mid x \in G\}$ of the complex Hilbert space $\bigoplus_{\sigma \in \hat{G}} \mathbb{C}^{d_\sigma^2}$ that is isomorphic to $\mathbb{C}[G]$.

Furthermore, this transformation makes the matrix representation of R in $\bigoplus_{\sigma \in \hat{G}} \mathbb{C}^{d_\sigma^2}$ block diagonal for all $x \in G$, with each block labelled by an irreducible representation $\sigma \in \hat{G}$.

Note also that the Fourier basis is an orthonormal basis of $\bigoplus_{\sigma \in \hat{G}} \mathbb{C}^{d_\sigma^2}$, since for any $|\tilde{x}\rangle, |\tilde{y}\rangle$ in the Fourier basis, we have that

$$\langle \tilde{x} | \tilde{y} \rangle = \sum_{\sigma, \tilde{\sigma} \in \hat{G}} \frac{d_\sigma^* d_\sigma}{|G|} \langle \tilde{\sigma} | \sigma \rangle \langle \tilde{\sigma}(x) | \sigma(y) \rangle = \sum_{\sigma \in \hat{G}} \frac{d_\sigma^2}{|G|} \langle \sigma(x) | \sigma(y) \rangle = \sum_{\sigma \in \hat{G}} \frac{d_\sigma^2}{|G|} \frac{\chi_\sigma(x^{-1}y)}{d_\sigma} = \delta_{xy} \quad (7.33)$$

by Equations (7.18), (7.21) and (7.22).

7.2 Weak Fourier Sampling

We use the statements in Remark 7.7 in what follows to describe the standard method for trying to solve the Hidden Subgroup Problem in the general case where the group is not abelian. In particular, we will transform a coset state expressed in the computational basis of $\mathbb{C}[G]$ using the Quantum Fourier Transform F_G to its equivalent state expressed in the Fourier basis in the isomorphic quantum system $\bigoplus_{\sigma \in \widehat{G}} \mathbb{C}^{d_\sigma^2}$. Since the group is not abelian, we'll use multiplicative notation for the group operation instead of the additive notation used for abelian groups. We will be formulating our approach in the language of density operators which were defined in Section 2.5.

We again start with a superposition of the form

$$\frac{1}{\sqrt{|G|}} \sum_{g \in G} |g\rangle |0\rangle \quad (7.34)$$

and apply the unitary transformation U_f to obtain the state

$$\frac{1}{\sqrt{|G|}} \sum_{g \in G} |g\rangle |f(g)\rangle \quad (7.35)$$

We then measure the second register and drop it, obtaining some coset state

$$|xH\rangle := \frac{1}{\sqrt{|H|}} \sum_{h \in H} |xh\rangle \quad (7.36)$$

for some $x \in G$ measured uniformly at random.

These are the same initial steps that we saw in the procedure for the abelian case.

We can see that

$$|xH\rangle = \frac{1}{\sqrt{|H|}} \sum_{h^{-1} \in H} |xh^{-1}\rangle = \frac{1}{\sqrt{|H|}} \sum_{h^{-1} \in H} R(h) |x\rangle = \frac{1}{\sqrt{|H|}} \sum_{h \in H} R(h) |x\rangle \quad (7.37)$$

We now consider the equivalent density operator formulation of $|xH\rangle$, that is

$$\rho_H := \frac{1}{|G|} \sum_{x \in G} |xH\rangle \langle xH| \quad (7.38)$$

which is equal to

$$= \frac{1}{|G||H|} \sum_{x \in G} \sum_{h, \tilde{h} \in H} R(h) |x\rangle \langle x| R(\tilde{h})^\dagger \quad (7.39)$$

$$= \frac{1}{|G||H|} \sum_{h, \tilde{h} \in H} R(h) R(\tilde{h})^\dagger \quad (7.40)$$

$$= \frac{1}{|G||H|} \sum_{h, \tilde{h} \in H} R(h\tilde{h}^{-1}) \quad (7.41)$$

$$= \frac{1}{|G|} \sum_{h \in H} R(h) \quad (7.42)$$

where we have used the Completeness Relation, $\sum_{x \in G} |x\rangle \langle x| = I$, in Equation (7.40), together with the fact that our representations are unitary in Equation (7.41).

Equation (7.42) can be seen as follows. Assume that H has some arbitrary n elements. Then we can calculate its Cayley table as

$$\begin{array}{c|cccc} \times & h_1 & h_2 & \cdots & h_n \\ \hline h_1 & h_1h_1 & h_1h_2 & \cdots & h_1h_n \\ h_2 & h_2h_1 & h_2h_2 & \cdots & h_2h_n \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ h_n & h_nh_1 & h_nh_2 & \cdots & h_nh_n \end{array} \quad (7.43)$$

Each row of the Cayley table contains the same elements, and each row (as a set) is precisely the elements of H itself.

As there are $|H|$ such rows, we have that

$$\sum_{h, \tilde{h} \in H} R(h\tilde{h}^{-1}) = \sum_{h, \tilde{h}^{-1} \in H} R(h\tilde{h}) = \sum_{h, \tilde{h} \in H} R(h\tilde{h}) = |H| \sum_{h \in H} R(h) \quad (7.44)$$

as required.

We can now apply the transformation $\rho \mapsto F_G \rho F_G^\dagger$ to Equation (7.42) (which is equivalent to applying F_G to the coset state $|xH\rangle$ by Lemma 2.40) and use the result of Proposition 7.6, which states that the right regular representation is block-diagonal in the Fourier basis, to give

$$\tilde{\rho}_H := F_G \rho_H F_G^\dagger \quad (7.45)$$

$$= \frac{1}{|G|} \sum_{h \in H} F_G R(h) F_G^\dagger \quad (7.46)$$

$$= \frac{1}{|G|} \sum_{h \in H} \bigoplus_{\sigma \in \hat{G}} (I_{d_\sigma} \otimes \sigma^*(h)) \quad (7.47)$$

$$= \frac{1}{|G|} \bigoplus_{\sigma \in \hat{G}} \left(I_{d_\sigma} \otimes \left(\sum_{h \in H} \sigma^*(h) \right) \right) \quad (7.48)$$

We can now perform what is known as weak Fourier sampling, which means measuring the density operator $\tilde{\rho}_H$ with respect to a set of measurement operators labelled by all the irreducible representations of G to obtain a specific irreducible representation $\sigma \in \hat{G}$ of dimension d_σ^2 . It is “weak” because we are not following this measurement up with a measurement of the row and column indices j, k corresponding to the matrix representation of $I_{d_\sigma} \otimes (\sum_{h \in H} \sigma^*(h))$, when considered in a form similar to that given by Equation (7.8). (Such a measurement is called strong Fourier sampling.)

The key point is that this measurement will not destructively interfere with the elements of the ensemble of pure states of $F_G |xH\rangle$, that is, the Fourier basis elements

$$\left\{ |\tilde{x}\rangle = \sum_{\sigma \in \hat{G}} \frac{d_\sigma}{\sqrt{|G|}} |\sigma\rangle |\sigma(x)\rangle \mid x \in G \right\} \quad (7.49)$$

since its density operator equivalent $\tilde{\rho}_H$ is block diagonal in the irreducible representations σ of G .

By Lemma 2.41, the probability of obtaining an irreducible representation σ is

$$P(\sigma) = \text{tr}(M_\sigma^\dagger M_\sigma \tilde{\rho}_H) \quad (7.50)$$

where the measurement operator M_σ is defined to be $|\sigma\rangle \langle \sigma| \otimes I_{d_\sigma} \otimes I_{d_\sigma}$.

Since $M_\sigma^\dagger M_\sigma = M_\sigma$, we have that

$$\begin{aligned}
P(\sigma) &= \text{tr}(M_\sigma \tilde{\rho}_H) \\
&= \text{tr} \left(\left[|\sigma\rangle \langle \sigma| \otimes I_{d_\sigma} \otimes I_{d_\sigma} \right] \left\{ \frac{1}{|G|} \sum_{\tilde{\sigma} \in \tilde{G}} \left[|\tilde{\sigma}\rangle \langle \tilde{\sigma}| \otimes I_{d_{\tilde{\sigma}}} \otimes \left(\sum_{h \in H} \tilde{\sigma}^*(h) \right) \right] \right\} \right) \\
&= \text{tr} \left(\frac{1}{|G|} \sum_{\tilde{\sigma} \in \tilde{G}} \left[|\sigma\rangle \langle \sigma| \tilde{\sigma} \rangle \langle \tilde{\sigma}| \otimes I_{d_\sigma} I_{d_{\tilde{\sigma}}} \otimes I_{d_\sigma} \left(\sum_{h \in H} \tilde{\sigma}^*(h) \right) \right] \right) \\
&= \frac{1}{|G|} \text{tr} \left(|\sigma\rangle \langle \sigma| \otimes I_{d_\sigma} \otimes \left(\sum_{h \in H} \sigma^*(h) \right) \right) \\
&= \frac{1}{|G|} \text{tr}(|\sigma\rangle \langle \sigma|) \text{tr}(I_{d_\sigma}) \text{tr} \left(\sum_{h \in H} \sigma^*(h) \right)
\end{aligned}$$

Now, since $\text{tr}(|\sigma\rangle \langle \sigma|) = 1$ and $\text{tr}(I_{d_\sigma}) = d_\sigma$, this implies that

$$\begin{aligned}
P(\sigma) &= \frac{d_\sigma}{|G|} \text{tr} \left(\sum_{h \in H} \sigma^*(h) \right) \\
&= \frac{d_\sigma}{|G|} \sum_{h \in H} \text{tr}(\sigma^*(h)) \\
&= \frac{d_\sigma}{|G|} \sum_{h \in H} \text{tr}(\sigma(h))^* \\
&= \frac{d_\sigma}{|G|} \sum_{h \in H} \chi_\sigma(h)^*
\end{aligned} \tag{7.51}$$

since σ is unitary.

If we restrict χ_σ to be a character of H , we can use the orthogonality of characters given in Theorem 4.21 and the fact that the one-dimensional, irreducible trivial representation $\mathbb{1}_H$ of H maps every element of H to $1 \in \mathbb{C}$ to say that

$$\sum_{h \in H} \chi_\sigma(h)^* = |H| \left(\frac{1}{|H|} \sum_{h \in H} \chi_\sigma(h)^* \chi_{\mathbb{1}_H}(h) \right) = |H| (\chi_\sigma, \chi_{\mathbb{1}_H})_H \tag{7.52}$$

Hence

$$P(\sigma) = \frac{d_\sigma |H|}{|G|} (\chi_\sigma, \chi_{\mathbb{1}_H})_H \tag{7.53}$$

Note that $(\chi_\sigma, \chi_{\mathbb{1}_H})_H$ is the number of times the irreducible trivial representation of H appears in σ when restricted as a representation of H by Theorem 4.22.

Consequently, we can see that the probability of measuring σ is independent of the coset of H that we obtain when measuring the second register of Equation (7.35), that is, it only depends on the subgroup H itself. In fact, the calculation above shows that this probability is entirely determined by the linear operator $\sum_{h \in H} \sigma(h)$.

This operator has a very nice structure, which we describe in the following Proposition:

Proposition 7.8. Let σ be an irreducible representation of G .

Then, for any subgroup $H \leq G$, we have that

$$\frac{1}{|H|} \sum_{h \in H} \sigma(h) : \mathbb{C}^{d_\sigma} \rightarrow \mathbb{C}^{d_\sigma} \quad (7.54)$$

is a projection operator onto the direct sum space $\mathbb{C}^{\oplus m_1}$, where $m_1 := (\chi_\sigma, \chi_{\mathbb{1}_H})_H$.

Proof. Since $\sigma \in \widehat{G}$ is being applied only to the elements of H , we can consider σ as a representation of H . Note that σ may not be irreducible when restricted as a representation of H by Remark 4.31.

Therefore we can apply Corollary 4.11 to write σ as a direct sum

$$\sigma = \rho_1 \oplus \rho_2 \oplus \cdots \oplus \rho_r \quad (7.55)$$

of irreducible representations $\rho_i : H \rightarrow GL(U_i)$, not necessarily unique.

Equivalently, we have that

$$\mathbb{C}^{d_\sigma} = U_1 \oplus U_2 \oplus \cdots \oplus U_r \quad (7.56)$$

Remark 4.12 says that we can choose a basis B of \mathbb{C}^{d_σ} in which every $\sigma(h)$ is block diagonal, where the i^{th} block corresponds to the i^{th} irreducible representation in the decomposition given in Equation (7.55).

Therefore, in this basis B , we can consider the linear operator $\frac{1}{|H|} \sum_{h \in H} \sigma(h)$ as a block diagonal matrix

$$M_B = \begin{pmatrix} \frac{1}{|H|} \sum_{h \in H} \rho_1(h) & 0 & \cdots & 0 \\ 0 & \frac{1}{|H|} \sum_{h \in H} \rho_2(h) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \frac{1}{|H|} \sum_{h \in H} \rho_r(h) \end{pmatrix} \quad (7.57)$$

By Lemma 4.44, we have that

$$\sum_{h \in H} \rho_i(h) = \begin{cases} |H| & \text{if } \rho_i = \mathbb{1}_H \\ 0 & \text{if } \rho_i \neq \mathbb{1}_H \end{cases} \quad (7.58)$$

that is, each sum on the diagonal of the matrix M_B is 1 when the irreducible representation is the trivial representation of H , and 0 otherwise. In particular, $M_B^2 = M_B$.

Since the linear operator $\frac{1}{|H|} \sum_{h \in H} \sigma(h)$ in any basis \tilde{B} is given by

$$UM_B U^\dagger \quad (7.59)$$

where U is a unitary change of basis operator from B to \tilde{B} , we have that

$$\left(\frac{1}{|H|} \sum_{h \in H} \sigma(h) \right)^2 = (UM_B U^\dagger)(UM_B U^\dagger) = UM_B U^\dagger = \frac{1}{|H|} \sum_{h \in H} \sigma(h) \quad (7.60)$$

Therefore $\frac{1}{|H|} \sum_{h \in H} \sigma(h)$ is a projector operator, which is clearly onto the direct sum space $\mathbb{C}^{\oplus m_1}$ by Equations (7.56) and (7.58). \square

Let us now define $\sigma(H) := \sum_{h \in H} \sigma(h)$ for an irreducible representation σ of G .

Then, from Proposition 7.8 and Equation (7.52), we can see that

$$\begin{aligned} (\chi_\sigma, \chi_{1_H})_H &= \frac{1}{|H|} \sum_{h \in H} \chi_\sigma(h)^* \\ &= \text{tr} \left(\frac{1}{|H|} \sum_{h \in H} \sigma(h) \right)^* \\ &= \text{tr} \left(\frac{1}{|H|} \sum_{h \in H} \sigma(h) \right) \\ &= \text{rank}(M_B) \\ &= \text{rank} \left(\frac{1}{|H|} \sigma(H) \right) \end{aligned}$$

since $\text{tr} \left(\frac{1}{|H|} \sum_{h \in H} \sigma(h) \right) \in \mathbb{R}$.

Therefore, we can also write the probability of measuring σ using weak Fourier sampling as

$$P(\sigma) = \frac{d_\sigma |H|}{|G|} \text{rank} \left(\frac{1}{|H|} \sigma(H) \right) \quad (7.61)$$

that is, a scalar multiple of the rank of a projection operator.

Similar to the abelian case, we would like to perform a polynomial in $\log |G|$ number of iterations of this procedure to determine the hidden subgroup H , by taking the intersection of the normal subgroups $\ker(\sigma) := \{g \in G \mid \sigma(g) = I_{d_\sigma}\}$ for all the irreducible representations σ obtained in the iterations. The next section gives an example of where this is possible.

7.3 Normal Subgroups

We show that performing weak Fourier sampling in a polynomial of $\log |G|$ number of iterations of the standard method when the subgroup H is normal in G will lead to determining H .

First, the critical reason why weak Fourier sampling works when the subgroup H is normal:

Proposition 7.9. Consider $\sigma(H) = \sum_{h \in H} \sigma(h)$ for some irreducible representation $\sigma : G \rightarrow GL(\mathbb{C}^{d_\sigma})$ of G . Suppose $H \trianglelefteq G$.

Show that $\sigma(H) = \lambda I_{d_\sigma}$ for some $\lambda \in \mathbb{C}$.

Proof. It is enough to show that $\sigma(H)$, which is a linear operator $\mathbb{C}^{d_\sigma} \rightarrow \mathbb{C}^{d_\sigma}$, is G -linear, that is

$$\sigma(H) \circ \sigma(g) = \sigma(g) \circ \sigma(H) \quad (7.62)$$

for all $g \in G$, because we can then apply Schur's Lemma (Theorem 4.13) to get the result.

Indeed, we have that

$$\begin{aligned}\sigma(H) \circ \sigma(g) &= \sum_{h \in H} \sigma(h)\sigma(g) \\ &= \sum_{\tilde{h} \in H} \sigma(x\tilde{h}x^{-1})\sigma(g)\end{aligned}\tag{7.63}$$

$$\begin{aligned}&= \sum_{\tilde{h} \in H} \sigma(x\tilde{h}x^{-1}g) \\ &= \sum_{\tilde{h} \in H} \sigma(gy\tilde{h}y^{-1})\end{aligned}\tag{7.64}$$

$$\begin{aligned}&= \sigma(g) \sum_{\tilde{h} \in H} \sigma(y\tilde{h}y^{-1}) \\ &= \sigma(g) \sum_{h \in H} \sigma(h)\end{aligned}\tag{7.65}$$

$$= \sigma(g) \circ \sigma(H)\tag{7.66}$$

where we have used, in Equation (7.63), that, for some $x \in G$ and for all $h \in H$, there exists \tilde{h} such that $x\tilde{h}x^{-1} = h$, by the normality of H . In Equation (7.64), we set $y^{-1} = x^{-1}g$, which is still an element of G , and used the normality of H in G again in Equation (7.65). \square

Proposition 7.10. Suppose that H is a normal subgroup of G . Then

$$P(\sigma) = \begin{cases} \frac{d_\sigma^2 |H|}{|G|} & \text{if } H \leq \ker(\sigma) \\ 0 & \text{if } H \not\leq \ker(\sigma) \end{cases}\tag{7.67}$$

that is, the rank of the projector $\frac{1}{|H|}\sigma(H)$ is of full rank, d_σ , when $H \leq \ker(\sigma)$, and 0 otherwise.

Proof. This is a very similar proof to that given in Step 5 of the procedure for the Abelian Hidden Subgroup Problem.

Suppose $H \leq \ker(\sigma)$. Then, since $\sigma(h) = I_{d_\sigma}$ for all $h \in H$, we have that $\frac{1}{|H|}\sigma(H) = I_{d_\sigma}$ also, and so its rank is d_σ .

Hence, by Equation (7.61), we have that

$$P(\sigma) = \frac{d_\sigma |H|}{|G|} \text{rank} \left(\frac{1}{|H|}\sigma(H) \right) = \frac{d_\sigma^2 |H|}{|G|}\tag{7.68}$$

as expected.

Suppose instead that $H \not\leq \ker(\sigma)$ and consider $\sigma(H)$. Since $\sigma(h_0) \neq I_{d_\sigma}$ for some $h_0 \in H$, we have that

$$\sigma(h_0)\sigma(H) = \sum_{h \in H} \sigma(h_0h) = \sum_{h \in H} \sigma(h) = \sigma(H)\tag{7.69}$$

hence

$$(\sigma(h_0) - I_{d_\sigma})\sigma(H) = 0\tag{7.70}$$

Since $\sigma(h_0) - I_{d_\sigma} \neq 0$ and $\sigma(H)$ is a scalar multiple of the identity by Proposition 7.9, it must mean that that scalar is 0. Hence $\sigma(H)$ is the zero map.

Therefore, $\text{rank} \left(\frac{1}{|H|}\sigma(H) \right) = 0$, and so $P(\sigma) = 0$ by Equation (7.61), as required. \square

Theorem 7.11. Combining Proposition 7.9 and Proposition 7.10, we can see that $\tilde{\rho}_H$ is block diagonal in the Fourier basis, where each block is labelled by the irreducible representations σ of G and is some $\lambda_\sigma \in \mathbb{C}$ scalar multiple of the identity matrix $I_{d_\sigma^2}$.

Note that the identity matrix is of size d_σ^2 , because $I_{d_\sigma} \otimes (\sum_{h \in H} \sigma^*(h))$ is a $d_\sigma^2 \times d_\sigma^2$ square matrix by the Kronecker Product of Definition 2.22.

Moreover, we can say that

$$\lambda_\sigma = \begin{cases} \frac{d_\sigma |H|}{|G|} & \text{if } H \leq \ker(\sigma) \\ 0 & \text{if } H \not\leq \ker(\sigma) \end{cases} \quad (7.71)$$

So $\tilde{\rho}_H$ has non-zero blocks of full rank only when $H \leq \ker(\sigma)$; furthermore, $\tilde{\rho}_H$ is not just block-diagonal but diagonal in the Fourier basis.

This is why weak Fourier sampling works in the case when H is normal, because the diagonalisation of the coset state allows us to weak Fourier sample only those irreducible representations of G whose kernel contains H .

In addition, for the σ that we measure, the projector $\frac{1}{|H|}\sigma(H)$ is $d_\sigma/|G|$ times the identity matrix I_{d_σ} , and so its full rank means that all of the available information about H is given up when enough iterations of the sampling procedure are performed. The case where G was abelian, as seen in Chapter 6, is a specific instance of this case, since all subgroups of an abelian group are normal.

We can show this explicitly by following the same steps as for the case where G is abelian:

Theorem 7.12. If we repeat this procedure $O(\log |G|)$ times, then H is the intersection of the $\ker(\sigma)$ for the irreducible representations σ we obtain from each iteration, with high probability.

Proof. Suppose that we are about to begin a new iteration of this procedure. We can denote the intersection of the kernels that we've obtained up to this point as K , which is a subgroup of G that we can assume is not H .

The probability of measuring an irreducible representation σ such that $K \leq \ker(\sigma)$ is

$$\sum_{\sigma \in \widehat{G} \mid K \leq \ker(\sigma)} P(\sigma) = \sum_{\sigma \in \widehat{G} \mid K \leq \ker(\sigma)} \frac{d_\sigma^2 |H|}{|G|} = \frac{|H|}{|G|} \sum_{\sigma \in \widehat{G}} d_\sigma^2 = \frac{|H|}{|G|} \frac{|G|}{|K|} = \frac{|H|}{|K|} \leq \frac{1}{2} \quad (7.72)$$

by the fact that representations of G that map K to the identity can be associated with representations of $\frac{G}{K}$, and by using Lagrange's Theorem to obtain the inequality.

Therefore, the probability of observing a σ such that $K \not\leq \ker(\sigma)$ is at least $\frac{1}{2}$. Similar to Equation (6.14) in the abelian case, we get that

$$|K \cap \ker(\sigma)| \leq \frac{|K|}{2} \quad (7.73)$$

by Lagrange's Theorem again.

Hence each iteration of the procedure has greater than half chance of reducing the intersection of all the kernels at least in half, and so $O(\log |G|)$ iterations will give H almost certainly. In fact, $4 \log_2 |G|$ iterations are enough - see [15] for details. \square

8 Conclusion

We have been able to cover a number of interesting topics in Quantum Computing in this thesis. In the first half, we were able to show how it is possible to replicate any classical algorithm on a quantum computer, coming up with our own explicit implementation using five registers. We looked at some applications of the Quantum Fourier Transform over the complex group algebra of \mathbb{Z}_{2^n} for some $n \in \mathbb{Z}_{\geq 1}$. We saw how being able to implement this version of the Quantum Fourier Transform on a quantum computer makes it possible to solve the Order-Finding Problem for any element in the group \mathbb{Z}_N^* . In turn, we saw how this leads to an efficient quantum algorithm, first discovered by Shor [26], which determines the prime factors of any composite integer in bounded probability polynomial time. We were able to give our own proof to one of the important results that makes up a step in this algorithm.

In the second half, we turned our attention to generalising the approach of Shor by studying a more general problem from Group Theory, the Hidden Subgroup Problem. After reviewing the key results from Group Representation Theory, we first described a quantum algorithm that finds the hidden subgroup in a polynomial of $O(\log |G|)$ time in the case where the group G is abelian. In doing so, we generalised the Quantum Fourier Transform to the complex group algebra over G , seeing how mapping its computational basis to the Fourier basis of the isomorphic group algebra of its dual group \hat{G} makes it possible to measure irreducible characters of G whose kernels all contain the hidden subgroup. We then saw how Shor's Algorithm and the Discrete Logarithm Problem were specific instances of this algorithm.

In the previous chapter, we went even further, studying the case where the group G is not abelian, which is an open problem in Quantum Computing. We were successful in describing the standard method in detail, which generalises the procedure for the case where the group is abelian. Again, we saw that mapping the computational basis of the complex group algebra of G to the Fourier basis of an isomorphic direct sum Hilbert space, indexed by the irreducible representations of G , leads a block diagonal density operator equivalent of a transformed coset state whose blocks are indexed by the irreducible representations of G . We wanted to understand whether we could recover the hidden subgroup from this density operator by performing a method known as weak Fourier sampling, which measures only the irreducible label itself, a polynomial of $O(\log |G|)$ number of times. We concluded by showing how this method gives a positive result when the hidden subgroup is normal.

With more time, it would have been good to take this one step further. For example, Hallgren et al. [15] show that weak Fourier sampling fails to determine the hidden subgroup when a group G has two distinct conjugate subgroups. This is because conjugate subgroups give the same probabilities for each irreducible representation of G , and so the intersection of the kernels of the irreducible representations obtained from repeating the procedure won't distinguish between distinct conjugate subgroups. One could try to go beyond this by performing strong Fourier sampling instead, which measures either the rows $|j\rangle$ or the columns $|k\rangle$ of the irreducible representation σ that is obtained from weak Fourier sampling, with the notation as in Equation (7.8). In fact, Grigni et al. [13] show that not only is there no point in measuring the rows $|j\rangle$, as they give up no more information, but also that in most cases strong Fourier sampling as a technique to find the hidden subgroup is not useful, even when coupled with the idea of picking "random bases" for each of the irreducible representations of the group when defining the Quantum Fourier Transform in the first place. The idea of picking random bases was considered because the Quantum Fourier Transform is defined uniquely only up to a choice of basis, as stated in Remark 7.3.

One problem of major interest is the Hidden Subgroup Problem in the case where the group is the symmetric group S_n . As S_n has distinct conjugate subgroups, weak Fourier sampling fails to give a

solution to this problem. Moore et al. [20] show that strong Fourier sampling cannot give a solution to this problem either. It is known [15], however, that finding a solution to this problem would automatically generate a solution to another open problem in Computer Science known as the Graph Isomorphism Problem. This problem asks whether it can be determined in polynomial time if two graphs of finite size are isomorphic to each other, that is, whether there exists a bijection between their vertex sets that induces a bijection between their edge sets. It has important applications to practical problems such as image processing, computer systems and information networks. A solution to the Hidden Subgroup Problem for the symmetric group S_n would therefore lead to a major breakthrough in a number of important topics in Mathematics and Computer Science.

A Literature Overview

The books on Quantum Computing range from the less technical, such as Bernhardt’s [2], to the very technical, such as the “bible” of the field written by Nielsen and Chuang [21]. Somewhere in the middle lies Rieffel and Polak’s book [23], which offers, in their own words, a “gentle” introduction to Quantum Computing. They are adamant in using Dirac notation to introduce all the linear algebra that is needed to be able to read technical papers in the field. Lipton and Regan’s book [19], however, takes the opposite approach, actively rejecting any use of Dirac notation. This is contrary to most approaches to the subject; despite this, their chapter on implementing Boolean functions on a quantum computer is very good and is not seen elsewhere.

Of the major papers, Deutsch’s 1985 seminal paper [9] laid out a quantum algorithm using only one oracle call to solve the problem of understanding whether a function applied to one bit was balanced or constant. An n -bit extension to this algorithm was given by Deutsch and Jozsa [10] in 1992. The No-Cloning Theorem was discovered by Wootters and Zurek [27] in 1982. Feynman wrote some of the early papers on Quantum Computing, and many of his ideas - especially on the infamous “billiard-ball computer” - can be found in [12]. Shor’s algorithm for factoring integers in bounded probability polynomial time [26] was published in 1994 and is the most famous paper in the field.

The notes written by Crommie and Vazirani [7] provide an excellent introduction to reversible circuits. The lecture notes written by Childs [3] form a useful guide to many of the algebraic quantum algorithms. They were created from his paper with van Dam [4]. de Wolf’s lecture notes [8] on Quantum Computing offer an easier introduction to the Hidden Subgroup Problem, describing a procedure for the problem in the case where the group is abelian. Hallgren, Russell and Ta-Shma’s paper [15] contains a solution to the Hidden Subgroup Problem when the hidden subgroup is normal. It builds on Hallgren’s Ph.D. thesis at the University of California, Berkeley [14] where he studied the weak Fourier sampling approach to the Hidden Subgroup Problem in both the abelian and non-abelian cases. Moore, Russell and Schulman [20] prove that strong Fourier sampling cannot give a solution to the Hidden Subgroup Problem for the symmetric group S_n . Grigni, Schulman and Vazirani’s paper [13] offers an overview of the standard method to the Hidden Subgroup Problem, as well as detailing other failed approaches to try to solve the problem where the group is not abelian.

The quantum circuits in this thesis were drawn using the `quantikz` package, with help from the tutorial written by Kay [18].

B Research Approach

I found myself reading Bernhardt's book [2] initially to establish a brief understanding of the topics in Quantum Computing, before mostly using the Nielsen and Chuang book [21] in the first half of the research time to build a technical understanding of those topics. I supplemented any misunderstandings with Edalat's lecture notes [11] and with Rieffel and Polak's book [23]. I used Conrad's notes on the Tensor Product [6] to add to my understanding of this key algebraic structure in Quantum Computing.

In Nielsen and Chuang's book, I found the chapter on linear algebra very comprehensive and specific to the study of Quantum Computing. Having worked through the exercises, I was able to use that chapter to understand the postulates of Quantum Mechanics. I worked through the chapter on single qubits, solving many of the exercises related to the Bloch Sphere, before studying the Quantum Fourier Transform and its applications to Phase Estimation and the Order-Finding Problem in the case where the quantum system is of size 2^n . I studied their appendices on Number Theory and Group Theory in order to understand the key results for Shor's Algorithm, working out my own proof of Proposition 3.34 as the statement and proof given in the book were incorrect. They also mention the Hidden Subgroup Problem, without going into much depth. I briefly looked at Grover's search algorithm for querying large databases before deciding that in the second half of the research time I wanted to study the Hidden Subgroup Problem instead.

The major source of inspiration for the Hidden Subgroup Problem came from Childs [3]. His notes lay out the bare bones of the argument, skipping over most of the details. He gave a description of the outer product representation of the Quantum Fourier Transform when the group is abelian, which made things easier to understand, especially when deriving the Inverse Quantum Fourier Transform. The outer product approach for describing operators in Quantum Computing is far better than the alternative of looking at the underlying matrix representation in the computational basis, because working at the operator level takes advantage of the Dirac notation in which these operators are expressed and makes the resulting layout more compact. I focused on working through the outline argument given by Childs [3] in minute depth, filling in all of the gaps, which meant proving most of the missing details myself. It has proven to be a very good use of my time, as the results of this work forms the basis of the second half of this thesis.

It quickly became clear that I needed to improve my understanding of Group Representation Theory, and so I used Segal's lecture notes [24] and Serre's book [25] to study the underlying theory. I also used Conrad's notes on the Characters of Finite Abelian Groups [5] to understand the dual group \widehat{G} and the orthogonal subgroup H^\perp , before generalising his work in the final presentation to fit in with the more general results of Segal and Serre. I used de Wolf's lecture notes [8] to gain an initial understanding of the Hidden Subgroup Problem, working through his procedure for the Abelian Hidden Subgroup Problem, before turning my attention to Childs' notes. During this time, I stumbled upon the Ph.D. thesis of Hallgren [14], which helped to consolidate my understanding, even though I found the density operator approach of Childs more intuitive. Hallgren's short summary on the key results of Group Representation Theory proved to be a very useful guide when it came to writing up Chapter 4, but it took a good amount of work to merge all the various sources into one coherent chapter, as they each took different approaches and came from different starting points.

I finished my research with almost two full notebooks of exercises and proofs as well as a number of annotated papers. Quantum Computing has been an exciting and challenging topic to study as I ended up needing to go far beyond my existing knowledge of Mathematics and Computer Science to make the progress that I have made. I very much enjoyed these last few months researching the topic. I hope you enjoyed reading my thesis.

References

- [1] Bell, J.S., 1964. On the Einstein Podolsky Rosen Paradox. *Physics Publishing Company*.
- [2] Bernhardt, C., 2019. *Quantum Computing for Everyone*. The MIT Press.
- [3] Childs, A.M., 2017. *Lecture Notes on Quantum Algorithms*. University of Maryland.
- [4] Childs, A.M. and van Dam, W., 2010. Quantum Algorithms for Algebraic Problems. *Reviews of Modern Physics* 82, 1-52.
- [5] Conrad, K., n.d. *Characters of Finite Abelian Groups*. Unpublished. Expository Paper.
- [6] Conrad, K., n.d. *Tensor Products II*. Unpublished. Expository Paper.
- [7] Crommie, M. and Vazirani, U., 2003. *C/CS/Phys191: Qubits, Quantum Mechanics and Computers*. University of California, Berkeley.
- [8] de Wolf, R., 2019. *Quantum Computing: Lecture Notes*. University of Amsterdam.
- [9] Deutsch, D., 1985. Quantum Theory, the Church-Turing Principle and the Universal Quantum Computer. *Proceedings of the Royal Society of London. Series A, Mathematical and Physical Sciences*.
- [10] Deutsch, D. and Jozsa, R., 1992. Rapid Solution of Problems by Quantum Computation. *Proceedings of the Royal Society of London. Series A, Mathematical and Physical Sciences*.
- [11] Edalat, A. *Quantum Computing*. Unpublished. Lecture Notes for the Course at Imperial College London.
- [12] Feynman, R.P., 1996. *Feynman Lectures on Computation*. Addison-Wesley.
- [13] Grigni, M., Schulman, L.J. and Vazirani, U., 2000. Quantum Mechanical Algorithms for the Nonabelian Hidden Subgroup Problem. *Combinatorica*.
- [14] Hallgren, S., 2000. *Quantum Fourier Sampling, the Hidden Subgroup Problem, and Beyond*. Ph.D. thesis. University of California, Berkeley.
- [15] Hallgren, S., Russell, A. and Ta-Shma, A., 2003. The Hidden Subgroup Problem and Quantum Computation using Group Representations. *Society for Industrial and Applied Mathematics*.
- [16] Hardy, G. and Wright, E., 1979. *An Introduction to the Theory of Numbers*. Oxford University Press.
- [17] Jozsa, R., 2000. Quantum Factoring, Discrete Logarithms, and the Hidden Subgroup Problem. [arXiv:quant-ph/0012084](https://arxiv.org/abs/quant-ph/0012084).
- [18] Kay, A., 2020. Tutorial on the Quantikz Package. [arXiv:1809.03842](https://arxiv.org/abs/1809.03842).
- [19] Lipton, R.J. and Regan, K.W., 2014. *Quantum Algorithms via Linear Algebra*. The MIT Press.
- [20] Moore, C., Russell, A. and Schulman, L.J., 2005. The Symmetric Group Defies Strong Fourier Sampling: Part I. [arXiv:quant-ph/0501056](https://arxiv.org/abs/quant-ph/0501056).
- [21] Nielsen, M.A. and Chuang, I.L., 2010. *Quantum Computation and Quantum Information*. Cambridge University Press.

-
- [22] Osgood, B., 2003. *Lecture Notes for EE 261: The Fourier Transform and its Applications*. Stanford University.
- [23] Rieffel, E. and Polak, W., 2014. *Quantum Computing: A Gentle Introduction*. The MIT Press.
- [24] Segal, E., 2014. *Group Representation Theory*. Imperial College London.
- [25] Serre, J.P., 1977. *Linear Representations of Finite Groups*. Springer.
- [26] Shor, P.W., 1994. Algorithms for Quantum Computation: Discrete Log and Factoring. *Proceedings of Foundations of Computer Science*.
- [27] Wootters, W.K. and Zurek, W.H., 1982. A Single Quantum Cannot be Cloned. *Nature*.