An Introduction to Quantum: Computing, Communication Complexity Protocols, Nonlocality and Graph Parameters

Lucas Arenstein

A dissertation submitted to the
Institute of Mathematics and Statistics
of the University of São Paulo
for the degree of
Master of Sciences

Department: Computer Science
Advisor: Prof. Dr. Yoshiharu Kohayakawa

The author was financially supported by CAPES

São Paulo, October 2022

An Introduction to Quantum: Computing, Communication Complexity Protocols, Nonlocality and Graph Parameters

Lucas Arenstein

This version of the thesis includes the corrections suggested by the examining committee during the defense of the original version of the work, which took place on December 12, 2022.

A copy of the original version is available at the Institute of Mathematics and Statistics of the University of São Paulo.

Examining Committee:

Prof. Dr. Yoshiharu Kohayakawa (advisor) – IME-USP
Prof. Dr. Marcelo Terra Cunha - IMECC - UNICAMP
Prof. Dr. Gabriel Coutinho - DCC - UFMG

Everybody has a plan until they get punched in the mouth.

Mike Tyson

It ain't about how hard you hit.
It's about how hard you can get hit and keep moving forward.

Rocky Balboa

Acknowledgements

I will be forever grateful to my advisor Yoshiharu Kohayakawa. I would like to thank him for sharing his wisdom, patience and guidance during my master's degree.

I also thank my examining committee Marcelo Terra Cunha and Gabriel Coutinho for their careful reading of my thesis and their useful comments.

Thanks should also go to my professors from IME-USP who taught me so much since my undergrad.

I'm thankful to my new and old friends Jonas Gonçalves, Jared León, Henrique Schechtmann, Las (Vitor Laskowsky) and Zebu (David Berl) for all the fun and nice time we had.

I'm very lucky to have such an amazing family. Many thanks to my sister, brother, niece, nephews, aunts, uncles and cousins.

I'm extremely grateful to Veronica Laminarca for her encouragement, love and the wonderful time we spent together. I'd also like to thank two cats Boo and Marie for bringing me joy (and not dead rats).

Above all, I would like to express my deepest gratitude to my mother Dirce and my father João for their unconditional love and support.

Abstract

Arenstein, L. **An Introduction to Quantum: Computing, Communication Complexity Protocols, Nonlocality and Graph Parameters.** Thesis – Institute of Mathematics and Statistics, University of São Paulo, São Paulo, 2022.

What are the advantages quantum computing can provide when compared to classical computing? In this thesis, we aim to answer this question from different perspectives. To achieve this goal we divided this work into three parts.

In Part I we start by studying the basic principles of quantum computing. Next, we introduce two quantum algorithms: the Deutsch–Jozsa and Grover's algorithms. The first algorithm has a lower query complexity and the second has a lower time complexity when compared with the best classical algorithm for the same problems. The final topic of this initial part is a detailed explanation and example of how to use Grover's algorithm to solve a Boolean formula in conjunctive normal form.

In the second part, we focus on communication complexity problems. These problems are usually stated as follows: two spatially separated parties Alice and Bob receive an input from a referee. Their goal is to compute the value of a function that depends on both of their inputs with the least amount of communication between them. In Part II we will introduce protocols in which the transmission of quantum bits (qubits), instead of bits, can reduce the amount of communication necessary to solve these problems. We also study how Alice and Bob can use quantum entanglement to solve two communication complexity problems without communicating something that can not be done classically.

In Part III we study nonlocal games inspired by standard graph theory parameters. A nonlocal game is usually defined as a game in which players that can share and do computations in an entangled state have some sort of advantage over classical players. We begin this final part by introducing the quantum chromatic number of a graph, which is the minimal number of colors necessary in a nonlocal game in which Alice and Bob can convince a referee with certainty that they have a proper coloring of the graph. We end this thesis by introducing other two quantum graph parameters, one related to graph homomorphism and the other to the independence number of a graph.

Keywords: quantum computing, quantum algorithms, Grover's algorithm, quantum communication complexity, nonlocality, quantum graph parameters, quantum chromatic number.

Resumo

Arenstein, L. **Uma Introdução à Computação Quântica, Protocolos de Complexidade de Comunicação Quânticos, Não-localidade e Parâmetros Quânticos de Grafos.** Tese – Instituto de Matemática e Estatística, Universidade de São Paulo, São Paulo, 2022.

Quais são as vantagens que a computação quântica pode oferecer em relação à computação clássica? O objetivo desta tese é responder a esta pergunta sob diferentes perspectivas, com este intuito, dividimos este trabalho em três partes.

Na primeira parte começamos a estudar os princípios básicos da computação quântica. Em seguida apresentamos dois algoritmos quânticos; o algoritmo de Deutsch–Jozsa e o algoritmo de Grover. O primeiro possui uma menor complexidade de *query*, já o segundo possui uma menor complexidade de tempo quando comparados aos melhores algoritmos clássicos para os mesmos problemas. O último tópico é uma explicação detalhada com um exemplo de como utilizar o algoritmo de Grover para resolver uma fórmula booleana expressa na forma normal conjuntiva.

Na segunda parte abordamos problemas de complexidade de comunicação. Estes problemas normalmente envolvem duas pessoas, Alice e Bob, que em locais separados recebem um input de um juiz. O objetivo deles é calcular o valor de uma função que depende dos seus inputs com a menor quantidade de comunicação entre eles. Nesta parte, apresentamos protocolos nos quais a transmissão de bits quânticos (qubits), ao invés de bits, podem reduzir a quantidade de comunicação necessária para resolver estes tipos de problemas. Também estudamos como Alice e Bob podem utilizar o emaranhamento quântico para resolver dois problemas de complexidade de comunicação sem que haja comunicação entre ambos, algo que não pode ser feito classicamente.

Na Parte III estudamos jogos não-locais inspirados em parâmetros usuais da teoria dos grafos. Jogos não-locais são definidos como jogos em que jogadores que podem compartilhar e operar em um estado emaranhado possuem algum tipo de vantagem sobre jogadores clássicos. Iniciamos esta última parte apresentando o número cromático quântico de um grafo. Esta quantidade representa o menor número de cores necessárias para Alice e Bob convencerem um juiz que possuem uma coloração própria de um grafo ao jogarem um jogo não-local. Terminamos esta tese apresentando outros dois parâmetros quânticos de grafos, um relacionado ao homomorfismo de grafos e o outro ao número de independência de um grafo.

Palavras-chave: computação quântica, algoritmos quânticos, algoritmo de Grover, complexidade de comunicação quântica, não-localidade, parâmetros quânticos de grafos, número cromático quântico.

Contents

Introduction

In the 80s the idea of building a computer that obeyed the laws of quantum mechanics began to be taken more seriously. By that time a few scientists had an intuition that a quantum computer could simulate a quantum system in a feasible amount of time, something that a classical computer could not do. The field of quantum computing was born.

Since then, there has been a great deal of development and discoveries in the field of quantum computing ranging from theoretical results in algorithms and complexity theory to practical implementations of quantum computers.

A recent breakthrough was made by Google in 2019. They announced that using their quantum computer with 53 qubits they manage to solve a problem in 3 minutes against 2.5 days that would take *Summit* the most powerful supercomputer at the time to solve the same problem.

Inspired by the theoretical discoveries and this recent breakthrough we believe that quantum computers powerful enough to solve useful problems that cannot be solved classically will become a reality.

In this thesis, divided into three parts, we are interested in different types of advantages quantum computing can provide when compared to classical computing. In Part I the advantages will be obtained by presenting two quantum algorithms. The first one is the Deutsch–Jozsa algorithm which provides an exponentially lower query complexity (number of calls to an oracle) when compared with the best classical algorithm. The second one is Grover's algorithm which provides a quadratic speed-up in time complexity over the best classical algorithm.

In the second part, we start by describing problems whose input is distributed among two or more physically separated parties. The solution to these problems is related to the parties' input. The field of communication complexity studies the amount of communication between the parties necessary to solve these sorts of problems. In Part II we introduce protocols in which the transmission of quantum bits, instead of bits, can reduce the amount of communication necessary to solve

these problems. Next, we discuss how entanglement can be used to further reduce (or even eliminate) the amount of communication for these communication complexity problems.

In the final part, we investigate how quantum information processing can provide new parameters related to well-investigated topics from graph theory. We introduce nonlocal games in which players that can perform quantum computation on their inputs have some sort of advantage over classical players.

Overview

In this thesis, besides presenting different advantages quantum computing can provide, each part is self-contained. For instance, a reader who is already familiar with quantum computing could go straight to the second or third part. Another aspect that we would like to emphasize is our effort to make this work as a friendly and mathematically rigorous introduction to each one of the three parts. We hope that anyone with a background in linear algebra can follow this thesis.

Part I - Quantum Computing: From Scratch to Grover

We start by introducing the basic principles of quantum computing. First, we define basic operations and notation that are commonly used in the literature. Next, we explain what a quantum computation is and introduce the quantum circuit model.

The first milestone of this part is to present one of the first quantum algorithms, the Deutsch–Jozsa algorithm, that solves the following problem: Suppose we can query a black box function $f : \{0,1\}^n \to \{0,1\}$. We are promised that f is either *constant* ($f(x)$ is the same for all x), or f is *balanced* ($f(x) = 0$ for exactly half of the input strings x, and $f(x) = 1$ for the other half of the inputs). How many queries do we have to make to find out with certainty what type of function f is?

Classically, in the worst case, one might need to query half plus one ($2^{n-1} + 1$) of all the possible inputs to find out what type of function f is. Surprisingly, the Deutsch–Jozsa algorithm solves this problem with just one query.

The final milestone is to introduce Grover's algorithm. The problem this algorithm solves is: Given a black box function $f : \{0,1\}^n \to \{0,1\}$ that on a single marked string $x^\star$ evaluates to 1 and for all the others inputs evaluates to 0, the goal is to determine $x^\star$. Classically, it takes approximately 2^n queries to the function f to find $x^\star$, which is linear in the domain size. Grover's algorithm solves this problem with high probability using only $O(\sqrt{2^n})$ queries to the function f. This quadratic

speed-up only happens when the best-known classical algorithm for solving this kind of problem is to naively search through all the potential solutions, which are typically exponential in the size of the problem instance.

The final topic of this first part is a detailed explanation and example of how to use Grover's algorithm to solve a Boolean formula in conjunctive normal form.

Part II - Quantum Communication Complexity Protocols and Nonlocality

We start the second part by defining basic concepts in the area of communication complexity. The first protocol that we present is the Distributed Deutsch–Jozsa which solves the following problem: Consider two players Alice and Bob who are physically separated from each other. An honest referee will send Alice an n-bit string x and Bob an n-bit string y, where n is a power of two. They are promised that their inputs are equal or they differ in exactly $n/2$ positions. Bob's goal is to determine the relation between x and y after receiving just one message from Alice.

In this scenario, we are interested in comparing how many bits against how many qubits Alice has to send to Bob to solve this problem with certainty. We show that, for this problem, the capability of sending qubits and performing quantum computation on them can save an exponential amount of communication.

For this problem, if we allow the classical protocol to have some small error probability there is no advantage in using qubits instead of bits. Motivated by this fact, we introduce the Hidden Matching problem. For this second problem, sending qubits instead of bits gives an exponential separation between a quantum and any classical protocol even if we allow the latter to have some small error probability.

In the second chapter of Part II, we give a short introduction of quantum nonlocality, one of the most nonclassical manifestations of quantum mechanics. Quantum nonlocality refers to the scenario where the results of local measurements carried out on an entangled system are somehow correlated. To understand some of the concepts of nonlocality we present a so-called nonlocal game in which players that can share and do computations in an entangled state have some sort of advantages over classical players.

In the final chapter of this part, we present nonlocal games derived from communication complexity problems. One of them is a variation of the Distributed Deutsch–Jozsa problem. In this variation, we forbid communication between the parties but allow them to share an entangled state before the game starts.

Part III - Quantum Graph Parameters

To understand the first quantum graph parameter of this final part, the quantum chromatic number, we need first to explain the c-coloring game. Alice and Bob receive a simple graph $G = (V, E)$ from a referee. Before the game starts they agree on a strategy and claim to the referee that they have a proper c-coloring of G. The players move apart and are now forbidden to communicate. The referee will test their claim with a one-round game by sending vertices $a \in V$ to Alice and $b \in V$ to Bob such that $a = b$ or $ab \in E$. After doing any computation they want Alice sends the color c_a and Bob the color c_b to the referee. They win the game if: $a = b$ and $c_a = c_b$ or ab and $c_a \neq c_b$.

We prove that for a graph G the minimum c that classical players can choose to win the c-coloring game with certainty is equal to $\chi(G)$. Surprisingly, if the players are allowed to use quantum resources there are graphs for which Alice and Bob can win the c-coloring game with probability 1 for $c < \chi(G)$. The smallest c such that quantum players can win the c-coloring game is called the quantum chromatic number denoted by $\chi_q(G)$.

In the first chapter of Part III, we present all the preliminary concepts from graph theory necessary to understand the quantum chromatic number. After that, we focus on a special family of graphs called the Hadamard graphs. We are interested in this family of graphs because they exhibit an exponential separation between the classical and quantum chromatic number. For the final topic of this chapter, we present a quantum protocol to win the c-coloring game while playing with the Hadamard graph.

In the second chapter of the third part we present a general quantum strategy for the c-coloring game. We are particularly interested in a graph with 18 vertices (called G_{18}) that presents a smaller quantum chromatic number when compared to the classical parameter. One of our goals is to explicitly formulate a quantum strategy to win the c-coloring with this graph.

The final topics of this thesis are devoted to other quantum graph parameters. They are related to graph homomorphism and the independence number of a graph which are well-investigated topics from graph theory.

Part I

Quantum Computing: From Scratch to Grover

Chapter 1

Preliminaries

In this initial chapter, we will give a (very) brief overview of the history, important discoveries and advances in quantum computing. In the rest of this first chapter we define basic operations and notation that will be used in this thesis.

1.1 A Brief History of Quantum Computing

Since the nineteenth century, physicists made experiments that did not agree with the laws of classical mechanics. The most prominent explanation for those experiments was made in the early 1920s. A new mathematical framework for physics called quantum mechanics was created. This theory describes the physical properties of nature at the scale of atoms and subatomic particles.

In the 80s a few scientists started to wonder if one could build computers that obeyed the laws of this new theory, using *qubits* (quantum bits) as the basic unit of quantum information in place of classical bits, and operate on this *qubits* according to the laws of quantum mechanics. Richard Feynman had an intuition at the time that to simulate an n-particle quantum system using a classical computer would be inefficient, as it would take an amount of time and space exponential in n. He idealized a system that would allow us to do computations using the natural quantum behavior of particles, hoping that this system would be more efficient than classical computers for this task [Fey82].

In 1985 David Deutsch defined the universal quantum Turing machine [Deu85] and up until 1994 there was some sparse activity with the development of quantum algorithms based on query complexity like the Deutsch–Jozsa Algorithm [DJ92] (introduced in Section 3.2) and also Simon's Algorithm [Sim97]. The creation of quantum complexity theory by Bernstein and Vazirani [BV97] was another significant step in quantum computing. What really sparked a tremendous interest in the

field was Peter Shor's quantum algorithm for efficiently factoring integers in 1994 [Sho94], that could theoretically break the RSA cryptosystem. Another impressive result is the quantum search algorithm discovered in 1996 by Lov Grover [Gro96], which we are going to explain in Chapter 4.1.

Since then there has been a great deal of development in the field of quantum computing ranging from theoretical results in algorithms and complexity theory to practical implementations of quantum computers. In 2019 Google announced that they had achieved Quantum Supremacy[1] using a quantum computer with 53 working qubits [AAB+19]. The latest Quantum Supremacy demonstration was in 2020 by the University of Science and Technology of China using 76 photons with their photonic quantum computer [ZWD+20].

1.2 Operations on Binary Strings

Before we start to present quantum computing we need to define some basic operations on binary strings. First we will show how to change between a natural number and a binary string representation of this number, and later introduce two basic operations on binary strings. These definitions are essential when constructing or studying quantum algorithms.

Definition 1.2.1
If d is a natural number then it can be written in the base-2 as

$$d = 2^{k-1}b_{k-1} + \dots 2b_1 + b_0, \tag{1.1}$$

where $b_j \in \{0, 1\}$ for all $0 \leq j \leq k-1$. The binary representation of d is given by the binary string $s = b_{k-1}b_{k-2}\dots b_1b_0$. The length of s is equal to k and its jth element is b_j.

Note that for the binary string representation of a natural number to be unambiguous we must set the length[2] of the binary string before doing the conversion.

For any two binary strings $r = r_0 \dots r_{l-1}$ and $s = s_0 \dots s_{l-1}$ we have the following operations.

[1] It is the goal of demonstrating that a programmable quantum device can solve a problem that no classical computer can solve in any feasible amount of time irrespective of the usefulness of the problem. This term was coined by John Preskill in 2012.

[2] The minimum number of bits required to represent a nonzero natural number d is $\lfloor \log_2 d \rfloor + 1$.

Definition 1.2.2
$r \oplus s$ is the bitwise addition modulo 2 given by

$$r \oplus s = t, \text{ with } t \in \{0,1\}^l \text{ and } t_k = \begin{cases} 0 & \text{if } r_k = s_k \\ 1 & \text{otherwise} \end{cases} \forall k = 0, \ldots, l-1. \quad (1.2)$$

The operator $\oplus$ is also known as the bitwise exclusive-or (XOR).

Definition 1.2.3
$r \bullet s$ is the bitwise dot product calculated as follows

$$r \bullet s = r_1 s_1 \oplus \cdots \oplus r_l s_l = \bigoplus_{k=1}^{l} r_k s_k. \quad (1.3)$$

1.3 Essential Linear Algebra and Dirac Notation

In this work, we are going to work with finite-dimensional complex vector spaces with an inner product denoted by $\mathbb{C}^n$. This type of vector space is a member of a broader class of vector spaces called Hilbert spaces, that is often used in the quantum computing literature. As we do not need any property of the Hilbert spaces in the rest of this paper, all the systems and operators we consider will be in the standard basis of $\mathbb{C}^n$ to be defined in 1.3.1.

Mathematicians are used to write a vector by putting an arrow over the symbol that identifies the vector or by writing this symbol in italic. In the next sections, you will see that when we are working with quantum states we often deal with really sparse vectors, that is why the *bra-ket* (also known as Dirac[3]) notation comes in hand.

A vector $v \in \mathbb{C}^n$ is denoted in the *bra-ket* notation as a *ket* $|v\rangle \in \mathbb{C}^n$. The zero vector will not be written as $|0\rangle$ because of Definition 1.3.1. Instead we will write it as 0, so

$$|v\rangle + 0 \leftrightarrow \begin{pmatrix} v_0 \\ v_1 \\ \vdots \\ v_{n-1} \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix},$$

where each entry of $|v\rangle$ is a complex number.

Next, is the standard basis definition using the *ket* notation.

[3]This notation was invented by the physicist Paul Dirac.

Definition 1.3.1
The standard basis or computational basis for $\mathbb{C}^{2^q}$, which has 2^q elements, is $\left\{ |j\rangle : j \in \{0,1\}^q \right\}$.

It is also possible to use the decimal representation of a binary string when denoting the basis of a vector space. We have the following equivalent ways of representing the basis states of $\mathbb{C}^{2^2}$:

Dirac Notation	Decimal Representation	Vector Notation
$\|00\rangle, \|01\rangle, \|10\rangle, \|11\rangle$	$\|0\rangle, \|1\rangle, \|2\rangle, \|3\rangle$	$\begin{pmatrix}1\\0\\0\\0\end{pmatrix}, \begin{pmatrix}0\\1\\0\\0\end{pmatrix}, \begin{pmatrix}0\\0\\1\\0\end{pmatrix}, \begin{pmatrix}0\\0\\0\\1\end{pmatrix}.$

To understand the *bra* notation we need to recall that for a real or complex vector space V we can define its dual vector space V^* that is the space of all linear functionals[4] on V. The elements of the dual space of $\mathbb{C}^n$ are written as a *bra* $\langle v| \in (\mathbb{C}^n)^*$. In this case a *bra* is a row vector

$$\langle v| \leftrightarrow \begin{pmatrix} v_0 & v_1 & \dots & v_{n-1} \end{pmatrix}.$$

One important property is that for every $|v\rangle \in \mathbb{C}^n$ there is a unique $\langle v| \in (\mathbb{C}^n)^*$ (called the *dual* of $|v\rangle$) obtained by taking the conjugate transpose[5] of $|v\rangle$:

$$|v\rangle = \begin{pmatrix} v_0 \\ v_1 \\ \vdots \\ v_{n-1} \end{pmatrix} \implies |v\rangle^\dagger = \langle v| = \begin{pmatrix} v_0^* & v_1^* & \dots & v_{n-1}^* \end{pmatrix}.$$

The equivalence $\langle v|^\dagger = |v\rangle$ is also true.

Using the *bra-ket* notation the *inner product* between $|v\rangle$ and $|u\rangle$ both in $\mathbb{C}^n$ is given by

$$\langle v|u\rangle = \begin{pmatrix} v_0^* & \cdots & v_{n-1}^* \end{pmatrix} \begin{pmatrix} u_0 \\ \vdots \\ u_{n-1} \end{pmatrix} = \sum_{j=0}^{n-1} v_j^* u_j, \tag{1.4}$$

[4] A linear functional is a linear transformation from a real or complex vector space to its field. If the reader is interested in more details of these definitions we recommend Section 3.5 of [HK71] or Section 3.F of [Axl15].

[5] In quantum computation the conjugate transpose is also referred to as the Hermitian conjugate or Hermitian adjoint and is denoted by a dagger †.

and our *norm* in $\mathbb{C}^n$ is

$$\| |v\rangle \| = \sqrt{\langle v|v\rangle} = \Big(\sum_{j=1}^{n} |v_j|^2 \Big)^{1/2}. \tag{1.5}$$

A quick observation on how to calculate $|v_j|^2$, the square absolute value of a complex number $|z|^2$ with $z = x + iy$ is calculated as follows

$$|z|^2 = zz^* = (x + iy)(x - iy) = x^2 + y^2. \tag{1.6}$$

1.4 Tensor Product

The tensor product is a way of combining vector spaces together to form larger vector spaces. Suppose U and V are complex vector spaces with dimension n and m respectively. Then we can form a new complex vector space which is the tensor product of these two spaces $W = U \otimes V$ with dimension nm. If $\{|u_1\rangle, \ldots, |u_n\rangle\}$ is a basis of U and $\{|v_1\rangle, \ldots, |v_m\rangle\}$ is a basis of V. Then

$$\{|u_i\rangle \otimes |v_j\rangle : 1 \leq i \leq n, 1 \leq j \leq m\}$$

is a basis for W. A vector in W can be expressed as $|w\rangle = \sum_{j,k} w_{j,k} |u_j\rangle \otimes |v_k\rangle$, meaning that the elements of W are linear combinations of tensor products $|u\rangle \otimes |v\rangle$. For simplicity they are often abbreviated as $|u\rangle |v\rangle$ or $|uv\rangle$.

Consider $|u\rangle, |u_1\rangle$ and $|u_2\rangle \in U$, $|v\rangle, |v_1\rangle$ and $|v_2\rangle \in V$ and a scalar $s \in \mathbb{C}$. By definition, the tensor product has the following properties:

(P1) $s(|u\rangle \otimes |v\rangle) = (s|u\rangle) \otimes |v\rangle = |u\rangle \otimes (s|v\rangle)$.

(P2) $(|u_1\rangle + |u_2\rangle) \otimes |v\rangle = |u_1\rangle \otimes |v\rangle + |u_2\rangle \otimes |v\rangle$.

(P3) $|u\rangle \otimes (|v_1\rangle + |v_2\rangle) = |u\rangle \otimes |v_1\rangle + |u\rangle \otimes |v_2\rangle$.

If A is a linear operator on U and B is a linear operator on V then we can define a linear operator $A \otimes B$ in $W = U \otimes V$ by the equation

$$(A \otimes B)(|u\rangle \otimes |v\rangle) \equiv A|u\rangle \otimes B|v\rangle, \text{ for all } |u\rangle \in U \text{ and } |v\rangle \in V. \tag{1.7}$$

Using the matrix notation of vectors and linear operators we can calculate the matrix representation for the tensor product between two vectors or between two linear operators using the Kronecker product.

Definition 1.4.1
If $A \in \mathbb{C}^{m\times n}$ and $B \in \mathbb{C}^{p\times q}$, then the Kronecker product $A \otimes B$ is the block matrix $M \in \mathbb{C}^{mp\times nq}$ defined by

$$M \coloneqq A \otimes B = \begin{pmatrix} a_{11}B & \dots & a_{1n}B \\ a_{21}B & \dots & a_{2n}B \\ \vdots & & \vdots \\ a_{m1}B & \dots & a_{mn}B \end{pmatrix}_{mp\times nq} . \tag{1.8}$$

Consider $|u\rangle = (u_0 u_1 \dots u_{n-1})^T$ and $|v\rangle = (v_0 v_1 \dots v_{m-1})^T$. The tensor product of $|u\rangle$ and $|v\rangle$ in the matrix notation is given by

$$|u\rangle \otimes |v\rangle = \begin{pmatrix} u_0 |v\rangle \\ u_1 |v\rangle \\ \vdots \\ u_{n-1} |v\rangle \end{pmatrix} = \begin{pmatrix} u_0 v_0 \\ u_0 v_1 \\ \vdots \\ u_0 v_{m-1} \\ u_1 v_0 \\ u_1 v_1 \\ \vdots \\ u_{n-1} v_0 \\ u_{n-1} v_1 \\ \vdots \\ u_{n-1} v_{m-1} \end{pmatrix}_{mn\times 1.}$$

A final remark is the notation $M^{\otimes n}$, which means M tensor product with itself n times, where M can be a vector or the matrix representation of a linear operator. This notation can also be used to express a vector space tensor product with itself n times, for instance, $(\mathbb{C}^d)^{\otimes m}$ is the complex vector space of dimension d^m.

Chapter 2

Quantum: Computation and Circuit Model

Our goal for this chapter is to define a quantum computation and also introduce the quantum circuit model. First, we will present the main characteristics of the *qubit* and how to do basic operations on single and multiples *qubits*. After that, we will motivate and discuss the importance of a universal set of quantum gates. These concepts will allow us to build the components of the quantum circuit model. We will end this chapter by introducing the measurement operation that is the last component of the quantum circuit model.

It is still under debate what the best physical implementation of a *qubit* is. If the reader is interested in this discussion we recommend chapter 7 of [NC02]. This discussion does not affect our work because we are going to treat the *qubit* as an abstract mathematical object that represents the basic unit of information on a quantum computer.

2.1 QuBits

The basic unit of quantum information is the qubit (quantum bit). To formally define a qubit we need to present the first postulate of quantum mechanics which tells us how physical states are represented.

State Space Postulate
Associated to any isolated physical system is a complex vector space with inner product known as the *state space* of the system. The system is completely described by its *state vector*, which is a unit vector in the system's state space.

Let $\mathbb{C}^2$ be the state space of a quantum system. Then we can completely describe this space by its state vector a *qubit* a unit vector in $\mathbb{C}^2$.

A qubit can be described by a linear combination of the basis vectors of $\mathbb{C}^2$, choosing the computational basis:

$$|\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle\,, \tag{2.1}$$

where $\alpha, \beta \in \mathbb{C}$ are called amplitudes. These amplitudes satisfy the *normalization condition* $|\alpha|^2 + |\beta|^2 = 1$ implying that $\||\psi\rangle\| = \sqrt{\langle\psi|\psi\rangle} = 1$. In the measurement section we will see that when we measure this qubit $|\psi\rangle$, we either observe the classical bit 0 with probability $|\alpha|^2$ or observe the classical bit 1 with probability $|\beta|^2$.

To combine a system with multiples qubits we have the following postulate.

Composition of Systems Postulate
When two physical systems are treated as one combined system, the state space of the combined physical system is the tensor product space $\mathbb{S}_1 \otimes \mathbb{S}_2$ of the state spaces $\mathbb{S}_1, \mathbb{S}_2$ of the component subsystems. If the first system is in the state $|\psi_1\rangle$ and the second system is in the state $|\psi_2\rangle$, then the state of the combined system is $|\psi_1\rangle \otimes |\psi_2\rangle$.

As mentioned before it is common to omit the $\otimes$ symbol and write the composition of states $|\psi_1\rangle$ and $|\psi_2\rangle$ as $|\psi_1\rangle\,|\psi_2\rangle$ or $|\psi_1\psi_2\rangle$.

According to these two postulates we will define the state of a quantum system with multiples qubits.

Definition 2.1.1
The state of a q-qubit quantum system is a unit vector in $(\mathbb{C}^2)^{\otimes q}$.

By Definition 1.3.1, the 2^q base vectors of $(\mathbb{C}^2)^{\otimes q}$ can be written using the Dirac notation (left) or with their decimal representation (right):

$$\begin{aligned}
\underbrace{|00\ldots00\rangle}_{q \text{ bits}} &= |0\rangle\,,\\
\underbrace{|00\ldots01\rangle}_{q \text{ bits}} &= |1\rangle\,,\\
\underbrace{|00\ldots10\rangle}_{q \text{ bits}} &= |2\rangle\,,\\
&\vdots\\
\underbrace{|11\ldots11\rangle}_{q \text{ bits}} &= |2^q - 1\rangle\,.
\end{aligned}$$

We can compactly represent the vectors on the left column as $|j\rangle$, $j \in \{0,1\}^q$ (binary strings of length q), and the vectors on the right column as $|k\rangle$, $0 \leq k \leq 2^q - 1$.

Using these notations we have the following equivalent ways to represent a q-qubit system

$$|\psi\rangle = \sum_{j\in\{0,1\}^q} \alpha_j |j\rangle , \qquad |\psi\rangle = \sum_{k=0}^{2^q-1} \alpha_k |k\rangle ,$$

$$\sum_{j\in\{0,1\}^q} |\alpha_j|^2 = 1, \qquad \sum_{k=0}^{2^q-1} |\alpha_k|^2 = 1,$$

where $\alpha_j \in \mathbb{C}$. where $\alpha_k \in \mathbb{C}$.

2.2 Basis States, Superposition and Entanglement

One of the key differences between a qubit and a bit is the possibility that a qubit can exist in a superposition state.

Definition 2.2.1
Consider a q-qubit quantum system

$$|\psi\rangle = \sum_{j\in\{0,1\}^q} \alpha_j |j\rangle . \tag{2.2}$$

We can classify the state of $|\psi\rangle$ as:

- *If there is an l such that $\alpha_l = 1$, then $|\psi\rangle$ is in a basis state.*
- *Otherwise $|\psi\rangle$ is in a superposition state.*

The following definition shows that a q-qubit quantum system is not always just the tensor product of q 1-qubit states.

Definition 2.2.2
A quantum state $|\psi\rangle \in (\mathbb{C}^2)^{\otimes q}$ is a product state if it can be expressed as a tensor product $|\psi_1\rangle \otimes \ldots \otimes |\psi_q\rangle$ of q 1-qubit states. Otherwise, it is entangled.

Consider $|\psi_1\rangle = (|00\rangle + |01\rangle + |10\rangle + |11\rangle)/2$. This is a product state because it can be expressed as $|\psi_1\rangle = ((|0\rangle + |1\rangle)/\sqrt{2}) \otimes ((|0\rangle + |1\rangle)/\sqrt{2})$. But $|\psi_2\rangle = (|00\rangle + |11\rangle)/\sqrt{2}$ is an entangled state because it cannot be expressed as a tensor product of two 1-qubit states.

Quantum entanglement has many applications in the field of quantum information theory. Well-known examples are quantum teleportation, superdense coding

and quantum cryptography. If the reader is interested in these topics we recommend Sections 1.3.7, 2.3 and 12.6 from [NC02]. In the second part of this thesis, we study how quantum entanglement can be used to solve communication complexity problems. And in the third part we study nonlocal games inspired by standard graph theory parameters.

With the concepts introduced so far, we can define the first component of the quantum circuit model.

Quantum Circuit Model - 1
1) A quantum computer has an initial state $|\psi\rangle = |b_1\rangle \otimes |b_2\rangle \otimes \cdots \otimes |b_q\rangle$, where $|\psi\rangle \in (\mathbb{C}^2)^{\otimes q}$ is a unit vector and each $b_i \in \{0, 1\}$. This state is stored in a quantum register.

In the next section, we are going to see what kind of operations we can apply to the state of a quantum system.

2.3 Operations on a Single Qubit

The next postulate tells us which type of operation we can apply to the state of our quantum computer.

Evolution Postulate
The evolution of a closed quantum system is described by a unitary operator. That is, the state $|\psi_1\rangle$ of the system is related to the next state of the system $|\psi_2\rangle$ by a unitary operator U:

$$|\psi_2\rangle = U |\psi_1\rangle . \tag{2.3}$$

From the evolution postulate we are going to define what kind of operation we can do on a quantum computer to evolve the state of the system from one state to another.

Definition 2.3.1
Any evolution operation applied by a quantum computer corresponds to a unitary matrix.

A complex square matrix U is unitary if its conjugate transpose $U^\dagger$ is also its inverse $U^\dagger = U^{-1}$, meaning that $U^\dagger U = UU^\dagger = \mathrm{I}$. An important property of the unitary matrix is that they preserve the norm of a vector

$$\|U |v\rangle\|^2 = (U |v\rangle)^\dagger (U |v\rangle) = \langle v| \underbrace{U^\dagger U}_{\mathrm{I}} |v\rangle = \langle v|v\rangle = \|v\|^2 .$$

By Definition 2.3.1 there are two important facts about quantum operations as they are realized by a unitary matrix: they are linear and reversible.

In Section 2.5 we are going into the details of which unitary matrices a quantum computer can physically implement.

Even though it looks like we have some major restrictions on what type of operations we can do on a quantum computer Deutsch [Deu85] proved that a universal quantum computer is Turing-complete, meaning it can simulate a universal Turing machine.

A linear operator is specified completely by its action on a basis. With that information we can construct the matrix for this linear operator on this basis. Considering the computational basis, the quantum NOT gate is a unitary operator that maps $|0\rangle$ to $|1\rangle$ and $|1\rangle$ to $|0\rangle$. The matrix for the NOT gate in the computational basis is

$$\mathrm{NOT} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}. \tag{2.4}$$

It follows by the definition that the NOT gate is a unitary operator because $(\mathrm{NOT}^*)(\mathrm{NOT}) = (\mathrm{NOT})(\mathrm{NOT}^*) = \mathrm{I}$.

Being a linear operator the NOT gate will map a linear combination of inputs to a linear combination of outputs. Consider the 1-qubit state $|\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle$. After applying the NOT gate on $|\psi\rangle$ we get

$$\mathrm{NOT}\,|\psi\rangle = \alpha\,|1\rangle + \beta\,|0\rangle\,. \tag{2.5}$$

Another extremely important 1-qubit quantum gate is the Hadamard gate (H) that on the computational basis acts as shown on the left while its matrix representation is given on the right:

$$\begin{aligned} \mathrm{H}\,|0\rangle &= \frac{|0\rangle + |1\rangle}{\sqrt{2}} \equiv |+\rangle \\ \mathrm{H}\,|1\rangle &= \frac{|0\rangle - |1\rangle}{\sqrt{2}} \equiv |-\rangle \end{aligned} \tag{2.6}$$

$$\mathrm{H} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}. \tag{2.7}$$

The interpretation of the NOT gate is straightforward: it negates (or flips) the basis state of a qubit. We are used to this operation in classical computing. But the Hadamard gate is an intrinsically quantum gate. It creates a superposition if applied to a basis state on the computational basis. We can easily verify a useful property of the Hadamard gate that $\mathrm{HH} = \mathrm{I}$ so $\mathrm{H} = \mathrm{H}^{-1}$.

2.4 Operations on Multiples Qubits

In the last section we presented how operations on a single qubit work. Now we are interested in operations on multiples qubits. Suppose we have a 2-qubit state $|\psi\rangle = (|01\rangle + |10\rangle)/\sqrt{2}$ and we want to apply the Hadamard gate on the first qubit and the NOT gate on the second qubit. The linear operator describing this operation on the composite system is $\mathrm{H} \otimes \mathrm{NOT}$. Let us calculate the state of the system after applying this operator:

$$\begin{aligned}
(\mathrm{H} \otimes \mathrm{NOT})\,|\psi\rangle &= (\mathrm{H} \otimes \mathrm{NOT})\Big(\frac{1}{\sqrt{2}}(|01\rangle - |10\rangle)\Big) \\
&= \frac{1}{\sqrt{2}}\Big(\mathrm{H}\,|0\rangle \otimes \mathrm{NOT}\,|1\rangle - (\mathrm{H}\,|1\rangle \otimes \mathrm{NOT}\,|0\rangle)\Big) && \text{by (1.7)} \\
&= \frac{1}{\sqrt{2}}\left(\frac{|0\rangle + |1\rangle}{\sqrt{2}} \otimes |0\rangle - \Big(\frac{|0\rangle - |1\rangle}{\sqrt{2}} \otimes |1\rangle\Big)\right) && \text{by (2.6) and (2.5)} \\
&= \frac{1}{2}\left(|00\rangle + |10\rangle - |01\rangle + |11\rangle\right).
\end{aligned}$$

We can achieve the same result using matrix multiplication:

$$\begin{aligned}
(\mathrm{H} \otimes \mathrm{NOT})\,|\psi\rangle &= \left(\frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\ 1 & -1\end{bmatrix} \otimes \begin{bmatrix}0 & 1\\ 1 & 0\end{bmatrix}\right)\frac{1}{\sqrt{2}}\begin{bmatrix}0\\ 1\\ -1\\ 0\end{bmatrix} \\
&= \frac{1}{\sqrt{2}}\begin{pmatrix}0 & 1 & 0 & 1\\ 1 & 0 & 1 & 0\\ 0 & 1 & 0 & -1\\ 1 & 0 & -1 & 0\end{pmatrix}\frac{1}{\sqrt{2}}\begin{bmatrix}0\\ 1\\ -1\\ 0\end{bmatrix} \\
&= \frac{1}{2}\begin{bmatrix}1\\ -1\\ 1\\ 1\end{bmatrix} = \frac{1}{2}\left(|00\rangle + |10\rangle - |01\rangle + |11\rangle\right).
\end{aligned}$$

In the examples above we saw how to operate simultaneously on two qubits. The procedure is analogous for any number of qubits we have in our quantum register.

Now, let us introduce the CNOT gate that is an important 2-qubit operation. Just as we saw in Definition 2.2.2 that a 2-qubit entangled state cannot be expressed as a tensor product of two 1-qubit states, it is also possible that a 2-qubit quantum gate cannot be expressed as a tensor product of two 1-qubit quantum gates. This type of gate is called a *entangling* gate. An example is the quantum controlled-NOT

or CNOT gate that on the computational basis acts as shown on the left while its matrix representation is given on the right:

$$\begin{aligned}
&\mathrm{CNOT}\,|00\rangle \to |00\rangle\,,\\
&\mathrm{CNOT}\,|01\rangle \to |01\rangle\,,\\
&\mathrm{CNOT}\,|10\rangle \to |11\rangle\,,\\
&\mathrm{CNOT}\,|11\rangle \to |10\rangle\,.
\end{aligned} \qquad \mathrm{CNOT} = \begin{bmatrix} 1 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 0 & 1\\ 0 & 0 & 1 & 0 \end{bmatrix}.$$

The CNOT gate negates the second qubit if the first qubit is in state $|1\rangle$, and does nothing otherwise.

One way to create a 2-qubit entangled state from a basis state is using the Hadamard and the CNOT gate. Consider the basis state $|00\rangle$, and apply the following operations on this system:

Apply a Hadamard gate on the first qubit

$$(\mathrm{H}\otimes\mathrm{I})(|00\rangle) = \mathrm{H}\,|0\rangle \otimes \mathrm{I}\,|0\rangle = \frac{|0\rangle+|1\rangle}{\sqrt{2}} \otimes |0\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |10\rangle),$$

now apply a CNOT gate

$$\mathrm{CNOT}\frac{1}{\sqrt{2}}(|00\rangle + |10\rangle) = \frac{1}{\sqrt{2}}(\mathrm{CNOT}\,|00\rangle + \mathrm{CNOT}\,|10\rangle) = \frac{|00\rangle+|11\rangle}{\sqrt{2}}.$$

The resulting state of the system is the entangled state $(|00\rangle + |11\rangle)/\sqrt{2}$.

2.5 Universal Set of Quantum Gates

Current quantum computer hardware cannot implement any unitary matrix, but as we will see in this section a discrete set of quantum gates can be used to approximate any unitary matrix. First we are going to give an intuition on what is necessary for a set of quantum gates to be universal, and later present an important and efficient result about the approximation of unitary matrices: the Solovay–Kitaev Theorem [Kit97]. If the reader is interested in a pedagogical review of this theorem we recommend [DN05].

Suppose T and A are two unitary matrices in $\mathbb{C}^{n\times n}$. The matrix T represents the target operator that we want but cannot implement and A represents the operator that is actually implemented. Let us define the error when A is implemented instead of T by

$$E(T,A) \equiv \sup_{|\psi\rangle} \|(T-A)\,|\psi\rangle\|\,, \tag{2.8}$$

where $|\psi\rangle$ ranges over all quantum states in $\mathbb{C}^n$.

We are going to say that a unitary matrix U can be approximated to arbitrary accuracy if given an error tolerance $\epsilon > 0$ we can implement a unitary matrix V such that $E(U, V) < \epsilon$.

Definition 2.5.1
A set of quantum gates $\mathbb{S}$ is said to be universal if, for any integer $q \geq 1$, any q-qubit unitary matrix can be approximated to arbitrary accuracy by only using a finite sequence of products of gates from $\mathbb{S}$.

To give an intuition in which gates would consist of a universal set of quantum gates, we are going to cite three essentials operations that such set of gates must be able to do:

1. They can create superposition.
2. They can create entanglement.
3. They must contain real and complex entries.

A candidate for the first condition is the Hadamard gate the as defined in (2.6), that maps a basis state to a superposition state. In the end of Subsection 2.4 we saw how to create an entangled state using the Hadamard and the CNOT gate.

This set of gates satisfy the conditions (1) and (2) but as they do not have any complex numbers we need to add another gate to our set. The T gate acts on the computational basis as follows

$$\begin{aligned} \mathrm{T}\,|0\rangle &= |0\rangle \\ \mathrm{T}\,|1\rangle &= e^{i\frac{\pi}{4}}\,|1\rangle\,. \end{aligned} \tag{2.9}$$

The set $\mathbb{S} = \{\mathrm{H}, \mathrm{CNOT}, \mathrm{T}\}$ intuitively is a valid candidate for a universal set of quantum gates. The next theorem proves that they are indeed.

Theorem 2.5.2 - Solovay-Kitaev (simplified)
Let $U = U_1 U_2 \cdots U_m$ be the product of a sequence of, $m \geq 1$, arbitrary unitary matrices acting on 1 or 2-qubits. Then we can approximate U with an error tolerance $\epsilon > 0$ using a sequence of products of only $O(m \log^{3.97}(m/\epsilon))$ gates from the set $\mathbb{S} = \{\mathrm{H}, \mathrm{CNOT}, \mathrm{T}\}$.

A better result was achieved by [Sel12] using only $O(m \log(m/\epsilon))$ gates from another universal set of quantum gates. Another key point of the Solovay-Kitaev Theorem is that it also provides a classical algorithm (with the upper bound from their theorem) to transform any unitary matrix into a sequence of unitary matrices

from our universal set of quantum gates. This is of much practical interest because it allows us to use any unitary matrix we want when designing a quantum algorithm.

With this theorem in mind we can better understand the next definition

Definition 2.5.3
A quantum computation is a sequence of unitary operations $U_1, U_2, \ldots, U_k$, $k \geq 1$ where each U_i $(1 \leq i \leq k)$ is the product of basic unitary matrices (acts non trivially on up to 3 qubits). The matrix associated with this computation is given by $U = U_k U_{k-1} \cdots U_1$.

When we realize a quantum computation on a quantum computer the result might be exact or an approximation that depends on which unitary matrices belong to the universal set of quantum gates from this quantum computer.

Using what we presented in the last section we can add another component to our quantum circuit model.

Quantum Circuit Model - 2
1) A quantum computer has an initial state $|\psi\rangle = |b_1\rangle \otimes |b_2\rangle \otimes \cdots \otimes |b_q\rangle$, where $|\psi\rangle \in (\mathbb{C}^2)^{\otimes q}$ is a unit vector and each $b_i \in \{0, 1\}$. This state is contained in a quantum register.
2.1) The state of the quantum computer evolves by applying unitary operations specified in advance in the form of an algorithm.
2.2) The final state of the system is given by the quantum computation (Definition 2.5.3) of all operations of step 2.1 applied to the initial state of the quantum register.

2.6 Measurements

The last component of the quantum circuit model is the measurement operation used to extract information about the actual state of the quantum system. When physically measuring the system we have to use some kind of external measurement apparatus. This implies that the Evolution Postulate is no longer appropriate for describing this operation as the system is no longer closed.

Measurement Postulate

Information on the state of a quantum computer can only be obtained through a measurement. Given a q-qubit state $|\psi\rangle = \sum_{j\in\{0,1\}^q} \alpha_j |j\rangle$, a computational measurement gate on the kth qubit outputs

$$0 \text{ with probability } \sum_{\substack{j\in\{0,1\}^q \\ j_k=0}} |\alpha_j|^2 \quad \text{or} \quad 1 \text{ with probability } \sum_{\substack{j\in\{0,1\}^q \\ j_k=1}} |\alpha_j|^2.$$

Let $x \in \{0,1\}$ be the measured value of the kth qubit. After the measurement the quantum state becomes

$$\sum_{\substack{j\in\{0,1\}^q \\ j_k=x}} \frac{\alpha_j}{\sqrt{\sum_{j:j_k=x} |\alpha_j|^2}} |j\rangle$$

and the original state $|\psi\rangle$ is no longer recoverable.

A computational measurement is also known as a measurement in the computational (or standard) basis or a Z measurement. For simplicity in this work, we are going to refer to this operation as a measurement. Most of the quantum algorithms use this type of measurement. But there are more general kinds of measurement (introduced in Section 9.1) that can be used in other scenarios.

To familiarize the reader with this operation, consider the 2-qubit state

$$|\psi\rangle = \sqrt{\frac{3}{11}} |00\rangle + \sqrt{\frac{5}{11}} |01\rangle + \sqrt{\frac{1}{11}} |10\rangle + \sqrt{\frac{2}{11}} |11\rangle .$$

The probability of obtaining the measurement outcome 0 on the second qubit is equal to

$$\sum_{\substack{j\in\{0,1\}^2 \\ j_2=0}} |\alpha_j|^2 = |\alpha_{00}|^2 + |\alpha_{10}|^2 = \frac{3}{11} + \frac{1}{11} = \frac{4}{11}.$$

After we measure a 0 on the second qubit of $|\psi\rangle$ the system is now in the state

$$|\psi'\rangle = \sum_{\substack{j\in\{0,1\}^q \\ j_2=0}} \frac{\alpha_j}{\sqrt{\sum_{j:j_2=0} |\alpha_j|^2}} |j\rangle = \frac{\sqrt{3/11}}{\sqrt{4/11}} |00\rangle + \frac{\sqrt{1/11}}{\sqrt{4/11}} |10\rangle = \sqrt{\frac{3}{4}} |00\rangle + \sqrt{\frac{1}{4}} |10\rangle .$$

The measurement postulate leads to a simple expression for the probability of observing a given binary string when measuring all the qubits. Consider the q-qubit state $|\psi\rangle = \sum_{j\in\{0,1\}^q} \alpha_j |j\rangle$. Applying a measurement operation to the q qubits in any order yields j with probability $|\alpha_j|^2$ for each $j \in \{0,1\}^q$. The proof is given in Proposition 3.6 of [Nan20].

Comparing the operations given by the evolution postulate (unitary operations) with the measurement postulate (measurement operation) we have the following difference between them.

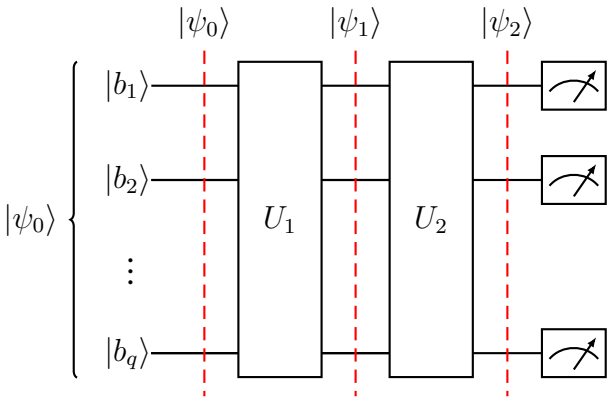
- Unitary operation:
  - **Reversible**, since if we apply a unitary operator U to a quantum state $|\psi\rangle$ we can get back to $|\psi\rangle$ by applying U^* to the state $U|\psi\rangle$.
  - **Deterministic**, there is nothing probabilistic in the evolution process of a quantum system.
- Measurement operation:
  - **Irreversible**, since every information about the system that you did not obtain through a measurement is lost.
  - **Probabilistic**, because the measurement outcomes are random.

Now we can add the final component of the quantum circuit model, the measurement operation.

Quantum Circuit Model - 3
1) A quantum computer has an initial state $|\psi\rangle = |b_1\rangle \otimes |b_2\rangle \otimes \ldots \otimes |b_q\rangle$, where $|\psi\rangle \in (\mathbb{C}^2)^{\otimes q}$ is a unit vector and each $b_i \in \{0, 1\}$. This state is contained in a quantum register.
2.1) The state of the quantum computer evolves by applying unitary operations specified in advance in the form of an algorithm.
2.2) The final state of the system is given by the quantum computation (Definition 2.5.3) of all operations of step 2.1 applied to the initial state of the quantum register.
3) The last step is to measure the final state of the system obtaining with a probability given by the measurement postulate a bit string as output of the circuit.

A graphical representation of the quantum circuit model is shown below. The rectangles represent unitary operations, and the red vertical dashed line represents the state of the quantum computer immediately after the operator on the left of this line.

Suppose $|\psi_0\rangle = |b_1\rangle \otimes |b_2\rangle \otimes \cdots \otimes |b_q\rangle$ is the initial state of the circuit, where each $b_i \in \{0, 1\}$.



Analyzing each step of the quantum circuit we have

$\|\psi_0\rangle$	is the initial state of the quantum register,
$\|\psi_1\rangle = U_1 \|\psi_0\rangle$	is the state after U_1 was applied on $\|\psi_0\rangle$,
$\|\psi_2\rangle = U_2 \|\psi_1\rangle$	is the state after U_2 was applied on $\|\psi_1\rangle$.

The quantum computation associated with this circuit is given by the matrix $U = U_2U_1$. After U_2 we have the measurement operations. One way we could measure the state q-qubit state $|\psi_2\rangle$ is: we start by measuring any one of the q qubits obtaining a bit with probability given by the measurement postulate. Now we update the state of the $(q-1)$-qubits system with the result from the first measurement again using the measurement postulate. For the other $(q-1)$-qubits we repeat this same procedure. After these q single qubit measurements we obtain a bit string of length q with probability equal to the product of the individual measurement probabilities conditioned on the state of the system before this individual measurement. As we mentioned before applying a measurement operation to the q qubits in any order yields j with probability $|\alpha_j|^2$ for each $j \in \{0,1\}^q$.

2.6.1 Global Phase

Consider two q-qubits quantum states $|\psi\rangle$ and $|\phi\rangle$ satisfying $|\psi\rangle = e^{i\theta} |\phi\rangle$ for some $\theta \in \mathbb{R}$. Now, let us apply an arbitrary unitary operator U on $|\phi\rangle$:

$$U |\phi\rangle = \sum_{j\in\{0,1\}^q} \alpha_j |j\rangle . \tag{2.10}$$

Using the relation between $|\psi\rangle$ and $|\phi\rangle$, applying U to $|\psi\rangle$ we have

$$U |\psi\rangle = Ue^{i\theta} |\phi\rangle = \sum_{j\in\{0,1\}^q} e^{i\theta}\alpha_j |j\rangle . \tag{2.11}$$

Let k be an arbitrary q-bit string. If we measure the state given by (2.10), we observe k with probability $|\alpha_k|^2$. If we measure the state given by (2.11), we observe k with probability $|e^{i\theta}\alpha_k|^2 = |\alpha_k|^2$. This result shows us that the probability of obtaining k as the outcome of a measurement after applying an arbitrary unitary matrix U is the same for $|\phi\rangle$ and $|\psi\rangle$. The factor $e^{i\theta}$ is called *global phase* and can be ignored[1]; there is no observable difference between these states.

[1] In the literature the expression "the states are the same up to a global phase" means that the factor $e^{i\theta}$ was used in some way.

Chapter 3

First Quantum Algorithm

David Deutsch in 1985 proposed the first quantum algorithm [Deu85]. In 1992 Deutsch and Jozsa generalized this algorithm [DJ92] that we are going to introduce in this chapter[1]. Other early quantum algorithms were the Bernstein–Vazirani algorithm [BV97] and Simon's algorithm [Sim97]. They were the first quantum algorithms that when analyzed under the query complexity model showed faster solutions than any classic algorithms. In this model we count the number of calls an algorithm makes to an oracle and ignore all the other operations in the algorithm.

In the first section we will define a quantum oracle that will help us understand the Deutsch–Jozsa Algorithm introduced in the second section. Later we will discuss quantum parallelism, a key property in quantum algorithms.

3.1 Oracles

Let $f : \{0,1\}^n \to \{0,1\}$ be a black box function. We have oracle access to f when we can query an input $x \in \{0,1\}^n$ and get $f(x)$ as output. A quantum oracle is a unitary operator U_f that implements f with the mapping

$$U_f \left|x, y\right\rangle = \left|x, y \oplus f(x)\right\rangle, \qquad \text{for all } x \in \{0,1\}^n \text{ and } y \in \{0,1\}. \tag{3.1}$$

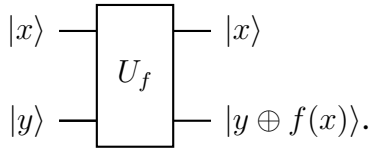
We can verify that U_f is a reversible operation since

$$U_f \left|x, y \oplus f(x)\right\rangle = \left|x, y \oplus f(x) \oplus f(x)\right\rangle = \left|x, y\right\rangle.$$

It is common to set $\left|y\right\rangle$ as $\left|0\right\rangle$, resulting in $U_f \left|x, 0\right\rangle = \left|x, 0 \oplus f(x)\right\rangle = \left|x, f(x)\right\rangle$ to get the result of $f(x)$ on the second register.

[1] We are actually going to introduce the improved version by Cleve, Ekert, Macchiavello and Mosca. [CEMM98] but keep the convention on the name of the Deutsch–Jozsa algorithm.

An oracle constructed in this way is called an XOR Oracle and is represented by the following circuit



From now on an oracle means that we are referring to the quantum case.

We will show that it is possible to write the function value into the phase of the amplitude rather than changing the second qubit value. If we set $|y\rangle$ to $|-\rangle = (|0\rangle - |1\rangle)/\sqrt{2}$ on the operator U_f defined in (3.1), we have

$$\begin{aligned}
U_f |x\rangle |-\rangle &= U_f \left(|x\rangle \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) \right) \\
&= \left(\frac{U_f(|x\rangle |0\rangle) - U_f(|x\rangle |1\rangle)}{\sqrt{2}} \right) \\
&= \left(\frac{|x\rangle |0 \oplus f(x)\rangle - |x\rangle |1 \oplus f(x)\rangle)}{\sqrt{2}} \right) \\
&= |x\rangle \left(\frac{|0 \oplus f(x)\rangle - |1 \oplus f(x)\rangle)}{\sqrt{2}} \right) \qquad \text{whether } f(x) = 0 \text{ or } f(x) = 1 \\
&= |x\rangle (-1)^{f(x)} |-\rangle \qquad \text{associating } (-1)^{f(x)} \text{ with the first qubit} \\
&= (-1)^{f(x)} |x\rangle |-\rangle .
\end{aligned} \tag{3.2}$$

This type of construction is called “Phase Kick-Back”. In the literature, it is common to call an XOR Oracle by Phase Oracle whenever we set $|y\rangle$ to $|-\rangle$ to use a representation such as (3.2).

3.2 Deutsch–Jozsa Algorithm

Suppose we have an n-bits classical oracle $f : \{0,1\}^n \to \{0,1\}$ that is either a:

- Constant function: $f(x) = 0$ or $f(x) = 1$ for all $x \in \{0,1\}^n$, or a
- Balanced function: $f(x) = 0$ for exactly half of $x \in \{0,1\}^n$.

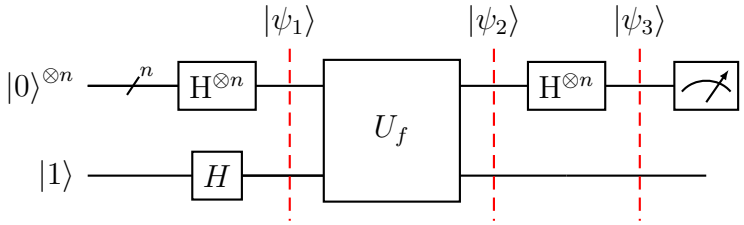
This algorithm solves the problem to determine if f is a constant or balanced function. The classically deterministic solution to this problem will need to make, in the worst case, $2^{n-1} + 1$ queries to the classical oracle as one needs to check half plus one values of the function to find its type. The Deutsch–Jozsa algorithm takes

advantage of being able to query the oracle in superposition[2], solving this problem with just one query. This algorithm is a generalization of the Deutsch algorithm that solves the same problem for $n = 1$.

For this algorithm we have access to a Phase Oracle U_f that implements f

$$U_f \left|x\right\rangle \left|-\right\rangle = (-1)^{f(x)} \left|x\right\rangle \left|-\right\rangle, \text{ where } x \in \{0,1\}^n.$$

Next, let us analyze each step of the quantum circuit of the Deutsch–Jozsa algorithm below:



The initial state of the system is

$$\left|\psi_0\right\rangle = \underbrace{\left|0\right\rangle \ldots \left|0\right\rangle}_{n} \left|1\right\rangle = \left|0\right\rangle^{\otimes n} \left|1\right\rangle.$$

The first operation of this circuit is to apply a Hadamard gate to each one of the $n+1$ qubits, leading to

$$\left|\psi_1\right\rangle = \mathrm{H}^{\otimes(n+1)} \left|\psi_0\right\rangle = \underbrace{\mathrm{H}\left|0\right\rangle \mathrm{H}\left|0\right\rangle \ldots \mathrm{H}\left|0\right\rangle}_{n} \mathrm{H}\left|1\right\rangle \overset{(2.6)}{=} \left|+\right\rangle^{\otimes n} \left|-\right\rangle. \tag{3.3}$$

The next step is to apply the oracle U_f to the state $\left|\psi_1\right\rangle$. As the oracle was defined in the computational basis we first need to rewrite $\left|\psi_1\right\rangle$ in this basis as well. Observe that

$$\begin{aligned} \left|+\right\rangle^{\otimes n} &= \frac{1}{\sqrt{2}}(\left|0\right\rangle + \left|1\right\rangle) \otimes \cdots \otimes \frac{1}{\sqrt{2}}(\left|0\right\rangle + \left|1\right\rangle) \\ &= \frac{1}{\sqrt{2^n}}\big((\left|0\right\rangle + \left|1\right\rangle) \otimes \cdots \otimes (\left|0\right\rangle + \left|1\right\rangle)\big) && \text{expanding the tensor product} \\ &= \frac{1}{2^{n/2}} \sum_{x\in\{0,1\}^n} \left|x\right\rangle && \text{equally weighted superposition of all the bit strings of size } n. \end{aligned} \tag{3.4}$$

We can rewrite $\left|\psi_1\right\rangle$ as

$$\left|\psi_1\right\rangle = \frac{1}{2^{n/2}} \sum_{x\in\{0,1\}^n} \left|x\right\rangle \left|-\right\rangle.$$

[2]In Section 3.3 we will discuss more on this operation.

As the first n qubits are now in the standard basis we can apply U_f to this state

$$|\psi_2\rangle = U_f |\psi_1\rangle = \frac{1}{2^{n/2}} \sum_{x\in\{0,1\}^n} U_f |x\rangle |-\rangle = \frac{1}{2^{n/2}} \sum_{x\in\{0,1\}^n} (-1)^{f(x)} |x\rangle |-\rangle . \tag{3.5}$$

We are going to discuss more of this operation called quantum parallelism in Section 3.3.

To facilitate our analysis before applying the last Hadamard to the first n qubits of $|\psi_2\rangle$ we are going to write the action of the Hadamard in another way. For any $x_1 \in \{0,1\}$

$$\mathrm{H} |x_1\rangle = \frac{1}{\sqrt{2}} \sum_{z_1\in\{0,1\}} (-1)^{x_1 z_1} |z_1\rangle .$$

It is easy to see that this is equivalent to (2.6).

Generalizing for n qubits represented by the bit string $x = x_1 \dots x_n$

$$\begin{aligned}
\mathrm{H}^{\otimes n} |x\rangle &= \mathrm{H} |x_1\rangle \otimes \dots \otimes \mathrm{H} |x_n\rangle \\
&= \frac{1}{2^{n/2}} \left(\sum_{z_1\in\{0,1\}} (-1)^{x_1 z_1} |z_1\rangle \otimes \dots \otimes \sum_{z_n\in\{0,1\}} (-1)^{x_n z_n} |z_n\rangle \right) \\
&= \frac{1}{2^{n/2}} \left(\sum_{z\in\{0,1\}^n} (-1)^{x_1 z_1 + \dots + x_n z_n} |z\rangle \right) \\
&= \frac{1}{2^{n/2}} \sum_{z\in\{0,1\}^n} (-1)^{x\bullet z} |z\rangle ,
\end{aligned} \tag{3.6}$$

where $\bullet$ is the dot product defined in (1.2.3). Using this last equation we can calculate $|\psi_3\rangle$, by applying a Hadamard on the first n qubits and the identity operator on the last one:

$$\begin{aligned}
|\psi_3\rangle &= (\mathrm{H}^{\otimes n} \otimes \mathrm{I}) |\psi_2\rangle \\
&= (\mathrm{H}^{\otimes n} \otimes \mathrm{I}) \left(\frac{1}{2^{n/2}} \sum_{x\in\{0,1\}^n} (-1)^{f(x)} |x\rangle |-\rangle \right) \\
&= \left(\frac{1}{2^{n/2}} \sum_{x\in\{0,1\}^n} (-1)^{f(x)} \mathrm{H}^{\otimes n} |x\rangle \right) \otimes \mathrm{I} |-\rangle \\
&= \frac{1}{2^{n/2}} \sum_{x\in\{0,1\}^n} (-1)^{f(x)} \left(\frac{1}{2^{n/2}} \sum_{z\in\{0,1\}^n} (-1)^{x\bullet z} |z\rangle \right) |-\rangle \qquad \text{by (3.6)} \\
&= \frac{1}{2^n} \sum_{z\in\{0,1\}^n} \sum_{x\in\{0,1\}^n} (-1)^{f(x)+x\bullet z} |z\rangle |-\rangle .
\end{aligned}$$

The last step of the algorithm is to measure $|\psi_3\rangle$. To make our analysis simpler let us rewrite $|\psi_3\rangle$ by separating the state $|0\rangle^{\otimes n}$ from all the others states in the first register

$$|\psi_3\rangle = \frac{1}{2^n}\Big(\sum_{x\in\{0,1\}^n} (-1)^{f(x)+x\bullet 00\ldots 0} |0\rangle^{\otimes n} + \sum_{\substack{z\in\{0,1\}^n \\ z\neq 0^n}} \sum_{x\in\{0,1\}^n} (-1)^{f(x)+x\bullet z} |z\rangle \Big) |-\rangle .$$

The amplitude of the state $|0\rangle^{\otimes n}$ is given by

$$\frac{1}{2^n} \sum_{x\in\{0,1\}^n} (-1)^{f(x)}.$$

Now consider the amplitude of this state when:

- f is constant $\implies \dfrac{1}{2^n} \displaystyle\sum_{x\in\{0,1\}^n} (-1)^{f(x)} = \begin{cases} 1, & \text{if } f(x) = 0 \ \forall x \in \{0,1\}^n \\ -1, & \text{if } f(x) = 1 \ \forall x \in \{0,1\}^n. \end{cases}$
- f is balanced $\implies \dfrac{1}{2^n} \displaystyle\sum_{x\in\{0,1\}^n} (-1)^{f(x)} = \frac{1}{2^n}\Big(\underbrace{\frac{1}{2^{n-1}} \cdot 1}_{\substack{\text{when } f(x)=0 \\ \text{half of the } x \text{ values}}} + \underbrace{\frac{1}{2^{n-1}} \cdot -1}_{\substack{\text{when } f(x)=1 \\ \text{other half of the } x \text{ values}}} \Big) = 0.$

Therefore, after measuring the first n qubits of $|\psi_3\rangle$ if we obtain the outcome $\underbrace{0\ldots 0}_{n}$ we are certain that f is a constant function. If we obtain any other outcome then f is a balanced function.

3.3 Quantum Parallelism

In (3.5) we applied the oracle U_f to a superposition of 2^n states just once and obtained the value of $f(x)$ for all of the 2^n input values. This is a unique quantum mechanical effect called quantum parallelism that we can use to build quantum algorithms. We are going to adapt the explanation given by [dW21].

Suppose we have a classical algorithm that computes some function $f : \{0,1\}^n \to \{0,1\}^m$. Then we can build a quantum circuit U_f (XOR Oracle) that maps $|x\rangle |0\rangle$ to $|x\rangle |f(x)\rangle$ for every $x \in \{0,1\}^n$. If we apply U_f to a superposition of all inputs x (like we did in (3.4)), we get

$$U_f \left(\frac{1}{\sqrt{2^n}} \sum_{x \in \{0,1\}^n} |x\rangle \, |0\rangle \right) = \frac{1}{\sqrt{2^n}} \sum_{x \in \{0,1\}^n} |x\rangle \, |f(x)\rangle \, .$$

We applied U_f just once but the final superposition contains $f(x)$ for all 2^n input values of x. However, this is not very useful by itself and does not give more than classical randomization, because when we measure the final superposition we will get just $|x\rangle \, |f(x)\rangle$ chosen uniformly at random and all other information will be lost. Quantum parallelism needs to be combined with other operations and effects like interference and entanglement in order to get something that is better than classical.

Note that the same explanation is valid for the phase oracle that we used in the Deutsch–Jozsa algorithm (3.5) because as we saw in Section 3.1 we can construct the Phase Oracle from the XOR Oracle.

Chapter 4

Grover's Algorithm

This chapter is dedicated to the study of Grover's algorithm. We begin with an overview of this algorithm. In Section 4.2 we will construct an important unitary operator used by Grover's Algorithm. In the following section, we will calculate how many times we have to run a subroutine to obtain the desired outcome with high probability.

After introducing this important quantum algorithm we will give a detailed explanation and example of how to use Grover's algorithm to solve a Boolean formula in conjunctive normal form. We will end this chapter with some further observations of this algorithm.

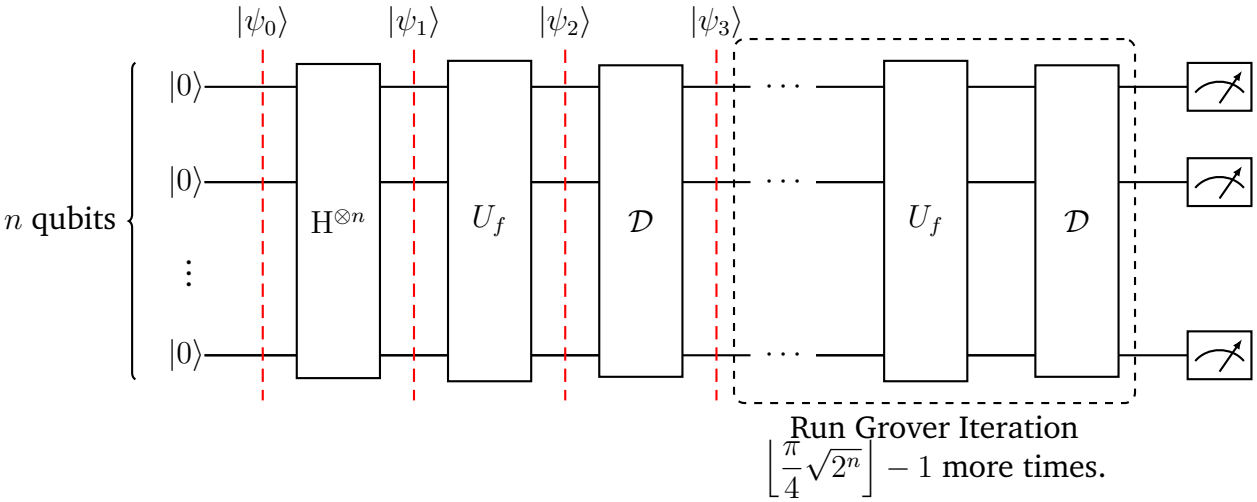
4.1 Algorithm Overview

In 1996, shortly after Shor's algorithm, Lov Grover discovered Grover's algorithm (also known as the quantum search algorithm) that provides a quadratic speed-up over the best-known classical algorithms for a wide class of important problems.

The problem Grover's algorithm solves is: Given a black box function $f : \{0,1\}^n \to \{0,1\}$ that on a single marked string $x^\star$ evaluates to 1 and for all the others inputs evaluates to 0, the goal is to determine $x^\star$. Classically it takes approximately 2^n queries to the function f to find $x^\star$, which is linear in the domain size. Grover's algorithm solves this problem with high probability using only $O(\sqrt{2^n})$ queries to an oracle that implements f. Independently of Grover's work, Bennett et al. (1997) proved that any quantum algorithm for this problem needs to evaluate the oracle $\Omega(\sqrt{2^n})$ times, so Grover's algorithm is asymptotically optimal [BBBV97].

This quadratic speed-up only happens when the best-known classical algorithm for solving this kind of problems is to naively search through all the potential solutions, which are typically exponential in the size of the problem instance.

The circuit of Grover's algorithm is shown below



As a resource we have access to U_f that implements f as before:

$$U_f |x\rangle = (-1)^{f(x)} |x\rangle = \begin{cases} |x\rangle , & \text{if } x \neq x^\star \\ -|x\rangle , & \text{if } x = x^\star . \end{cases} \tag{4.1}$$

We can see that there are some similarities between the Deutsch–Jozsa algorithm with Grover's algorithm, such as the use of quantum parallelism. What we want to emphasize in this section is how the use of interference implemented by a certain operator $\mathcal{D}$ is a crucial step in this algorithm.

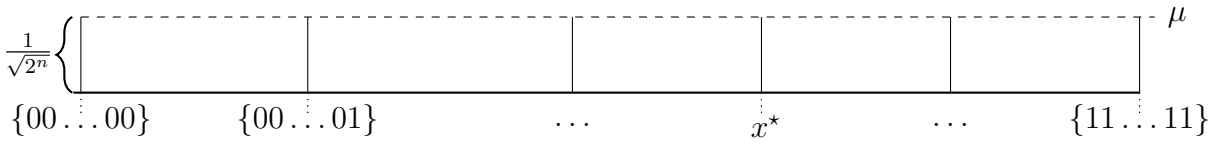
The initial state of the system is

$$|\psi_0\rangle = |0\rangle^{\otimes n} .$$

The first step of the algorithm stands for our complete lack of knowledge of the marked string $x^\star$, so we start with an equal superposition of all the possible input values of f by applying a Hadamard gate to each one of the n qubits, as we did in (3.3), and writing this state $|\psi_1\rangle$ using the same notation as (3.4):

$$|\psi_1\rangle = \mathrm{H}^{\otimes n} |\psi_0\rangle = |+\rangle^{\otimes n} = \frac{1}{\sqrt{2^n}} \sum_{x \in \{0,1\}^n} |x\rangle . \tag{4.2}$$

The current amplitudes of the state of the system $|\psi_1\rangle$ can be visualized with the following diagram,



where μ represents the mean of the amplitudes.

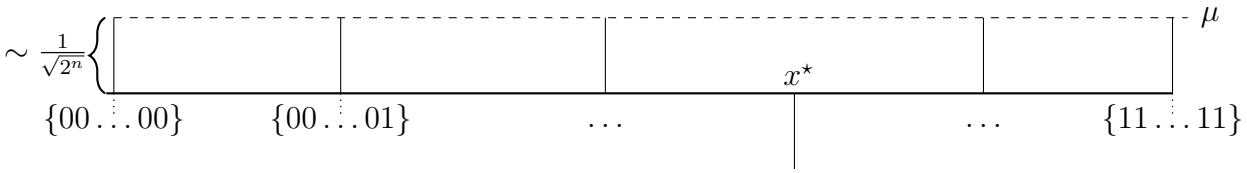
With the goal of increasing the amplitude of the $|x^\star\rangle$ state and decreasing the amplitude of all the other states we will apply the "Grover Iteration" routine $\left\lfloor \frac{\pi}{4}\sqrt{2^n} \right\rfloor$ times:

- Apply U_f.
- Apply the Grover Diffusion Operator $\mathcal{D}$.

After the first application of U_f on $|\psi_1\rangle$ the system is now in

$$|\psi_2\rangle = U_f\,|\psi_1\rangle = -\frac{1}{\sqrt{2^n}}\,|x^\star\rangle + \sum_{\substack{x\in\{0,1\}^n \\ x\neq x^\star}} \frac{1}{\sqrt{2^n}}\,|x\rangle$$

Graphically,



Now that the state $|x^\star\rangle$ is marked (has a negative amplitude) we will apply the Grover Diffusion Operator. This operator will act with a constructive interference on this marked state and with a destructive interference on the other states.

Let μ be the average of the amplitudes of the actual system

$$\mu = \frac{1}{2^n}\sum_{x\in\{0,1\}^n} \alpha_x, \tag{4.3}$$

where α_x is the amplitude of $|x\rangle$ for each $x \in \{0,1\}^n$.

The Grover Diffusion Operator $\mathcal{D}$ implements the mapping

$$\mathcal{D}\left(\sum_{x\in\{0,1\}^n} \alpha_x\,|x\rangle\right) = \sum_{x\in\{0,1\}^n} (2\mu - \alpha_x)\,|x\rangle\,. \tag{4.4}$$

In the next section we will construct $\mathcal{D}$ out of unitary gates showing that the Grover Diffusion Operator is a valid quantum operation.

Before applying $\mathcal{D}$ to the current state of the system $|\psi_2\rangle$ we need to calculate μ:

$$\mu = \frac{1}{2^n}\sum_{x\in\{0,1\}^n} \alpha_x = \frac{1}{2^n}\Big(\underbrace{\frac{2^n-1}{\sqrt{2^n}}}_{\text{all } x \text{ except } x^\star} - \underbrace{\frac{1}{\sqrt{2^n}}}_{x^\star} \Big) = \frac{1}{2^n}\left(\frac{2^n-2}{\sqrt{2^n}}\right) \approx \frac{1}{2^n}\left(\frac{2^n}{\sqrt{2^n}}\right) = \frac{1}{\sqrt{2^n}},$$

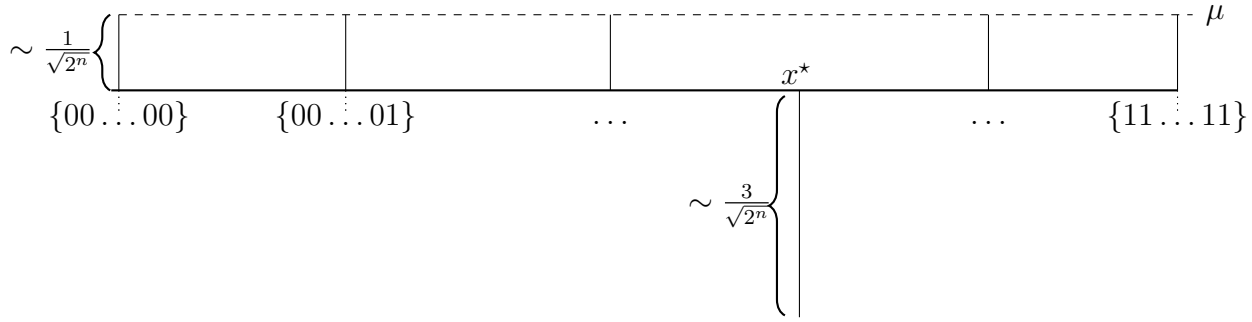
the exact value of μ is $\frac{1}{\sqrt{2^n}}$ minus an irrelevant value that we will not consider.

Let us calculate $|\psi_3\rangle = \mathcal{D}\,|\psi_2\rangle$:

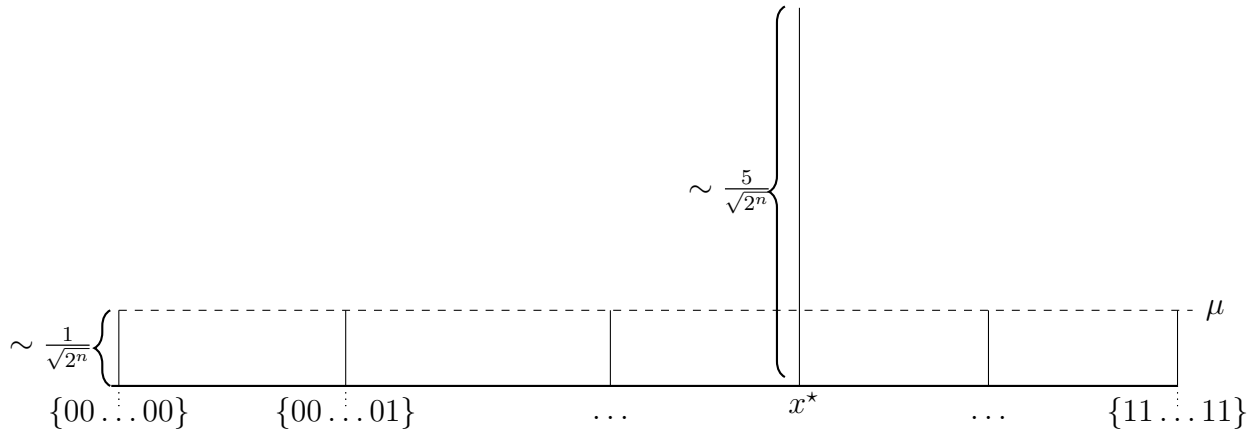
$$
\begin{aligned}
|\psi_3\rangle = \mathcal{D}\,|\psi_2\rangle &= \mathcal{D}\left(-\frac{1}{\sqrt{2^n}}\,|x^\star\rangle\right) + \mathcal{D}\left(\sum_{\substack{x\in\{0,1\}^n \\ x\neq x^\star}} \frac{1}{\sqrt{2^n}}\,|x\rangle\right) \\
&= \left(2\mu - (-\frac{1}{\sqrt{2^n}})\right)|x^\star\rangle + \left(\sum_{\substack{x\in\{0,1\}^n \\ x\neq x^\star}} \left(2\mu - \frac{1}{\sqrt{2^n}}\right)|x\rangle\right) \\
&\approx (\frac{2}{\sqrt{2^n}} + \frac{1}{\sqrt{2^n}})\,|x^\star\rangle + \left(\sum_{\substack{x\in\{0,1\}^n \\ x\neq x^\star}} \left(\frac{2}{\sqrt{2^n}} - \frac{1}{\sqrt{2^n}}\right)|x\rangle\right) \\
&\approx \frac{3}{\sqrt{2^n}}\,|x^\star\rangle + \sum_{\substack{x\in\{0,1\}^n \\ x\neq x^\star}} \frac{1}{\sqrt{2^n}}\,|x\rangle\,.
\end{aligned}
$$

The amplitude of $|x^\star\rangle$ increased to approximately $\frac{3}{\sqrt{2^n}}$ in magnitude, while all the other amplitudes stayed roughly the same.

On the second iteration of the Grover Iteration Operator, after applying U_f to $|\psi_3\rangle$ we have the following amplitudes:



and after applying $\mathcal{D}$ on this state we get another increase on the amplitude of the $|x^\star\rangle$ state:



After $\left\lfloor \pi\sqrt{2^n}/4 \right\rfloor$ executions of the Grover Iteration, we measure the system and obtain the outcome $x^\star$ with high probability, and that is the final step of the algorithm.

In the following section we will see how to construct the Grover Diffusion Operator out of unitary gates, and later prove that the number of iterations used gives us the desired output with high probability.

4.2 Building the Grover Diffusion Operator

In this section we are going to construct the unitary operator that implements $\mathcal{D}$. First we will show that with two gates we can implement the mapping of $\mathcal{D}$ as shown in (4.4). Besides the Hadamard gate, we need a new operator Z_0 defined by

$$\mathrm{Z}_0 \left|x\right\rangle = \begin{cases} \left|x\right\rangle, & \text{if } \left|x\right\rangle = \left|0^n\right\rangle \\ -\left|x\right\rangle, & \text{otherwise.} \end{cases} \tag{4.5}$$

The unitary representation of this operator is $\mathrm{Z}_0 = 2\left|0^n\right\rangle\left\langle 0^n\right| - \mathrm{I}$, where I is the $n \times n$ identity matrix. Let us verify that this representation implements (4.5) correctly. First we apply Z_0 to the $\left|0^n\right\rangle$ state

$$\mathrm{Z}_0 \left|0^n\right\rangle = 2\left|0^n\right\rangle\left\langle 0^n|0^n\right\rangle - \mathrm{I}\left|0^n\right\rangle = 2\left|0^n\right\rangle 1 - \left|0^n\right\rangle = \left|0^n\right\rangle,$$

and now to an arbitrary state $\left|x\right\rangle$ different from $\left|0^n\right\rangle$

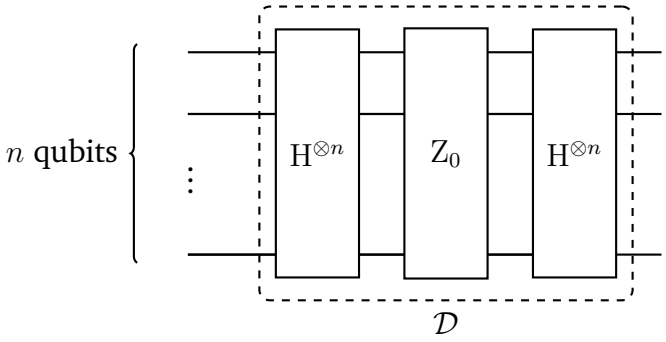
$$\mathrm{Z}_0 \left|x\right\rangle = 2\left|0^n\right\rangle\left\langle 0^n|x\right\rangle - \mathrm{I}\left|x\right\rangle = 2\left|0^n\right\rangle 0 - \left|x\right\rangle = -\left|x\right\rangle,$$

meaning that the unitary representation is correct.

Now, using the Z_0 gate and Hadamard gates we can construct the Grover Diffusion Operator $\mathcal{D}$:

$$
\begin{aligned}
\mathcal{D} &= \mathrm{H}^{\otimes n}\mathrm{Z}_0\mathrm{H}^{\otimes n} \\
&= \mathrm{H}^{\otimes n}(2\,|0^n\rangle\,\langle 0^n| - \mathrm{I})\mathrm{H}^{\otimes n} \\
&= 2\Big(\,\big(\mathrm{H}^{\otimes n}\,|0^n\rangle\big)\,\big(\langle 0^n|\,\mathrm{H}^{\otimes n}\big)\,\Big) - \big(\underbrace{\mathrm{H}^{\otimes n}\mathrm{H}^{\otimes n}}_{\text{involutory}}\big) \\
&= 2\Big(\,|+^n\rangle\,\big(\mathrm{H}^{\otimes n}\,|0^n\rangle\big)^*\,\Big) - \mathrm{I} \\
&= 2\Big(\,|+^n\rangle\,(|+^n\rangle)^*\,\Big) - \mathrm{I} \\
&= 2\Big(\,|+^n\rangle\,\langle +^n|\,\Big) - \mathrm{I}.
\end{aligned}
$$

The circuit representation of $\mathcal{D}$ is



To prove that $\mathcal{D}$ realizes the same mapping defined in (4.4) we are going to apply $\mathcal{D}$ to an arbitrary state $|\psi\rangle = \sum_{x\in\{0,1\}^n} \alpha_x\,|x\rangle$:

$$
\begin{aligned}
\mathcal{D}\,|\psi\rangle &= \big(2\big(\,|+^n\rangle\,\langle +^n|\,\big) - \mathrm{I}\big)\,|\psi\rangle \\
&= 2\big(\,|+^n\rangle\,\langle +^n|\psi\rangle\,\big) - \mathrm{I}\,|\psi\rangle\,.
\end{aligned}
$$

By noting that

$$
\langle +^n| = \left(\frac{\langle 0| + \langle 1|}{\sqrt{2^n}}\right)^{\otimes n} = \sum_{x\in\{0,1\}^n} \frac{1}{\sqrt{2^n}}\,\langle x|\,, \tag{4.6}
$$

so we can rewrite $\langle +^n|\psi\rangle$ using (4.6)

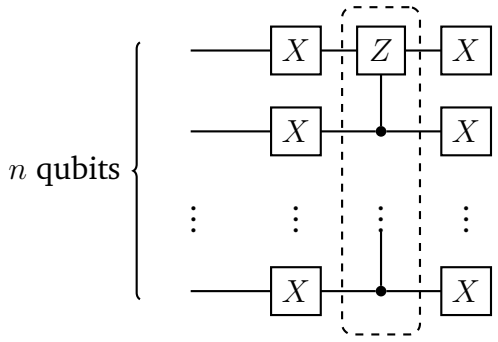
$$
\begin{aligned}
\langle +^n|\psi\rangle &= \sum_{x\in\{0,1\}^n} \frac{\alpha_x}{\sqrt{2^n}}\,\langle x|x\rangle \\
&= \sum_{x\in\{0,1\}^n} \frac{\alpha_x}{\sqrt{2^n}} \\
&= \mu\sqrt{2^n}, \qquad\qquad \text{because } \mu = \frac{1}{2^n}\sum_{x\in\{0,1\}^n} \alpha_x.
\end{aligned}
$$

Using this last equality we can keep simplifying $\mathcal{D}\left|\psi\right\rangle$:

$$\begin{aligned}
\mathcal{D}\left|\psi\right\rangle &= 2\left|+^n\right\rangle\left\langle+^n|\psi\right\rangle - \left|\psi\right\rangle \\
&= 2\left|+^n\right\rangle \mu\sqrt{2^n} - \left|\psi\right\rangle \\
&= 2\left(\frac{\left|0\right\rangle+\left|1\right\rangle}{\sqrt{2^n}}\right)^{\otimes n} \mu\sqrt{2^n} - \left|\psi\right\rangle \\
&= 2\left(\sum_{x\in\{0,1\}^n} \frac{1}{\sqrt{2^n}}\left|x\right\rangle\right) \mu\sqrt{2^n} - \left|\psi\right\rangle \\
&= 2\left(\sum_{x\in\{0,1\}^n} \mu\left|x\right\rangle\right) - \left|\psi\right\rangle \\
&= \left(\sum_{x\in\{0,1\}^n} 2\mu\left|x\right\rangle\right) - \sum_{x\in\{0,1\}^n} \alpha_x\left|x\right\rangle \\
&= \sum_{x\in\{0,1\}^n} (2\mu-\alpha_x)\left|x\right\rangle .
\end{aligned}$$

That is exactly like the mapping defined in (4.4).

Now, for the last part, we are going to show that the circuit below implements Z_0 as defined in (4.5) up to a global phase (as mentioned in Subsection 2.6.1).



Multi-controlled Z Gate.

The only new operation in this circuit is the Multi-controlled Z Gate that applies a Z gate on the first qubit if all the other $n-1$ qubits are in the $\left|1\right\rangle$ state. Let $x \in \{0,1\}^n$ and

$$\mathrm{MCZ}\left|x\right\rangle = \begin{cases} -\left|x\right\rangle, & \text{if } \left|x\right\rangle = \left|1^n\right\rangle \\ \left|x\right\rangle, & \text{otherwise.} \end{cases} \tag{4.7}$$

To verify that the circuit above implements Z_0 up to a global phase of $e^{i\pi} = -1$, we are first going to apply it to the $\left|0^n\right\rangle$ state:

$$\mathrm{NOT}^{\otimes n}\mathrm{MCZ}(\mathrm{NOT}^{\otimes n}\left|0^n\right\rangle) = \mathrm{NOT}^{\otimes n}(\mathrm{MCZ}\left|1^n\right\rangle) = \mathrm{NOT}^{\otimes n}(-\left|1^n\right\rangle) = -\left|0^n\right\rangle,$$

and now to $|x\rangle$ where $x \in \{0,1\}^n$, $x \neq 0^n$:

$$\mathrm{NOT}^{\otimes n}\mathrm{MCZ}(\mathrm{NOT}^{\otimes n}|x^n\rangle) = \mathrm{NOT}^{\otimes n}(\mathrm{MCZ}(\mathrm{NOT}^{\otimes n}|x^n\rangle)) = \mathrm{NOT}^{\otimes n}(\mathrm{NOT}^{\otimes n}|x^n\rangle) = |x^n\rangle\,,$$

as desired.

4.3 Grover Iteration

In this section, we are going to compute the number of times we have to run the Grover Iteration to obtain the outcome $x^\star$ with high probability after measuring the system. We will follow an algebraic approach to find this quantity but there is also an interesting geometric interpretation of the Grover Iteration that can be found in Section 6.1.3 of [NC02]

One key point of this analysis is that repeated applications of the Grover Iteration always keep the system in a 2-dimensional subspace of $(\mathbb{C}^2)^{\otimes n}$ spanned by $\{|x^\star\rangle, |\psi'\rangle\}$ where

$$|\psi'\rangle = \frac{1}{\sqrt{2^n-1}} \sum_{\substack{x\in\{0,1\}^n \\ x\neq x^\star}} |x\rangle\,. \tag{4.8}$$

Let

$$|\overline{\psi}\rangle = \sqrt{\frac{2^n-1}{2^n}}\,|x^\star\rangle - \frac{1}{\sqrt{2^n}}\,|\psi'\rangle\,.$$

We can verify that $|\overline{\psi}\rangle$ is orthogonal to $|\psi_1\rangle = \frac{1}{\sqrt{2^n}}|x^\star\rangle + \sqrt{\frac{2^n-1}{2^n}}|\psi'\rangle$. These two vectors form another basis for this same subspace.

Now let us prove that Grover's algorithm operates entirely within the subspace spanned by $\{|x^\star\rangle, |\psi'\rangle\}$. Considering the action of U_f defined in (4.1) we can see that this operator preserve this subspace

$$U_f|x^\star\rangle = -|x^\star\rangle \text{ and } U_f|\psi'\rangle = |\psi'\rangle\,.$$

Before showing that $\mathcal{D}$ also preserve this subspace we are going to rewrite $\mathcal{D}$ using the equality below that was mentioned before in (4.2)

$$|+^n\rangle = \frac{1}{\sqrt{2^n}} \sum_{x\in\{0,1\}^n} |x\rangle\,,$$

using this equation it is easy to show that $\mathcal{D} = 2|\psi_1\rangle\langle\psi_1| - \mathrm{I}$. Now let us apply $\mathcal{D}$ to the basis vectors and use (4.8) to write our expressions in terms of $|x^\star\rangle$ and $|\psi'\rangle$

$$\mathcal{D}\left|x^{\star}\right\rangle = \left(2\left|\psi_1\right\rangle\left\langle\psi_1\right| - \mathrm{I}\right)\left|x^{\star}\right\rangle = 2\left|\psi_1\right\rangle\left\langle\psi_1|x^{\star}\right\rangle - \left|x^{\star}\right\rangle = \frac{2}{\sqrt{2^n}}\left|\psi_1\right\rangle - \left|x^{\star}\right\rangle$$
$$= \frac{2\sqrt{(2^n-1)}}{2^n}\left|\psi'\right\rangle + \left(\frac{2}{2^n} - 1\right)\left|x^{\star}\right\rangle,$$

$$\mathcal{D}\left|\psi_1\right\rangle = \left(2\left|\psi_1\right\rangle\left\langle\psi'\right| - \mathrm{I}\right)\left|\psi'\right\rangle = 2\sqrt{\frac{2^n-1}{2^n}}\left|\psi_1\right\rangle - \left|\psi'\right\rangle = \left(\frac{2(2^n-1)}{2^n} - 1\right)\left|\psi'\right\rangle + \frac{2\sqrt{2^n-1}}{2^n}\left|x^{\star}\right\rangle$$
$$= -\left(\frac{2}{2^n} - 1\right)\left|\psi'\right\rangle + \frac{2\sqrt{2^n-1}}{2^n}\left|x^{\star}\right\rangle,$$

concluding that $\mathcal{D}$ also preserve the subspace.

Define an angle θ so that $\sin(\theta) = 1/\sqrt{2^n}$ from the trigonometric identity $\sin^2\theta + \cos^2\theta = 1$ we have $\cos(\theta) = \sqrt{2^n-1}/\sqrt{2^n}$. Let us write $\left|x\right\rangle, \left|\psi'\right\rangle, \left|\psi_1\right\rangle$ and $\left|\overline{\psi}\right\rangle$ in function of θ

$$\left|x^{\star}\right\rangle = \sin(\theta)\left|\psi_1\right\rangle + \cos(\theta)\left|\overline{\psi}\right\rangle, \qquad \left|\psi'\right\rangle = \cos(\theta)\left|\psi_1\right\rangle - \sin(\theta)\left|\overline{\psi}\right\rangle, \tag{4.9}$$
$$\left|\psi_1\right\rangle = \sin(\theta)\left|x^{\star}\right\rangle + \cos(\theta)\left|\psi'\right\rangle, \qquad \left|\overline{\psi}\right\rangle = \cos(\theta)\left|x^{\star}\right\rangle - \sin(\theta)\left|\psi'\right\rangle. \tag{4.10}$$

these equations allow us to convert between the two bases.

Next, we want to analyze the action of repeated applications of the Grover Iteration in terms of the four equations above. The state of the system immediately before the first application of U_f is $\left|\psi_1\right\rangle = \sin(\theta)\left|x\right\rangle + \cos(\theta)\left|\psi'\right\rangle$. Applying U_f gives the state

$$U_f\left|\psi_1\right\rangle = -\sin(\theta)\left|x^{\star}\right\rangle + \cos(\theta)\left|\psi'\right\rangle = \cos(2\theta)\left|\psi_1\right\rangle - \sin(2\theta)\left|\overline{\psi}\right\rangle. \tag{4.11}$$

Applying $\mathcal{D}$ to (4.11) we have

$$\begin{aligned}\mathcal{D}(-\sin(\theta)\left|x^{\star}\right\rangle + \cos(\theta)\left|\psi'\right\rangle) &= \sin(3\theta)\left|x^{\star}\right\rangle + \cos(3\theta)\left|\psi'\right\rangle \\ &= \cos(2\theta)\left|\psi_1\right\rangle + \sin(2\theta)\left|\overline{\psi}\right\rangle.\end{aligned} \tag{4.12}$$

To get to the equations (4.11) and (4.12) we used (4.9) and (4.10) to convert between the two bases and also trigonometric identities.

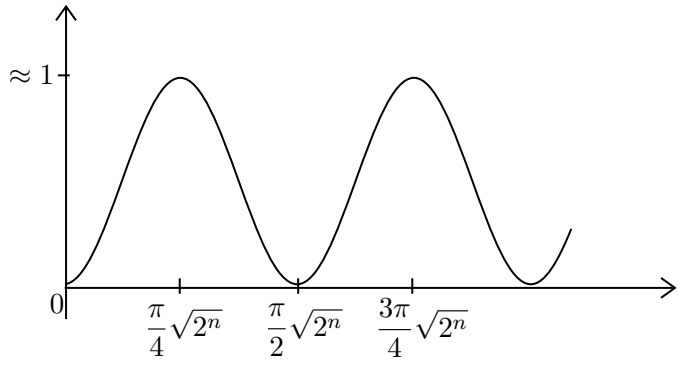
Let $\mathcal{G}$ be the operator that represent the Grover Iteration $\mathcal{G} = \mathcal{D}U_f$, we can verify by induction that after k application on the starting state $\left|\psi_1\right\rangle$ we get to the state

$$\begin{aligned}\mathcal{G}^k\left|\psi_1\right\rangle &= \sin((2k+1)\theta)\left|x^{\star}\right\rangle + \cos((2k+1)\theta)\left|\psi'\right\rangle \\ &= \cos(2k\theta)\left|\psi_1\right\rangle + \sin(2k\theta)\left|\overline{\psi}\right\rangle.\end{aligned} \tag{4.13}$$

When measuring (4.13) we want that the amplitude of the state $|x^\star\rangle$ to be as close to 1 as possible, $\sin((2k+1)\theta) \approx 1$ so we want to find $(2k+1)\theta \approx \pi/2$. For a large n we can use the small angle approximation for $\sin$, in our case $\sin(\theta) \approx \theta$. Calculating k

$$k \approx \frac{\pi}{4\theta} - \frac{1}{2} \approx \frac{\pi}{4}\sqrt{2^n},$$

thus after $\left\lfloor \pi\sqrt{2^n}/4 \right\rfloor$ applications of the Grover Iteration when we measure the state of the system we get the outcome $x^\star$ with a probability close to one. By analyzing (4.13) we can verify that the amplitudes of $|x^\star\rangle$ and $|\psi'\rangle$ are periodic. We can plot the success probability of Grover's algorithm as a function of the number of iterations.



4.4 Solving a CNF formula with Grover's Algorithm

In this section we will show how to solve a Boolean formula expressed in conjunctive normal form (CNF) using Grover's algorithm. We will focus on how to build an oracle for this problem as we already discussed in detail all the other components of Grover's algorithm.

Consider a CNF formula with v variables and c clauses. We use two registers to build this circuit. The first one, with v qubits initialized in the state $|0\rangle$, in the end, will contain the assignment that satisfies the CNF formula. The second register is used exclusively by the oracle and has c qubits also initialized in the state $|0\rangle$ plus one last "check" qubit in the $|1\rangle$ state.

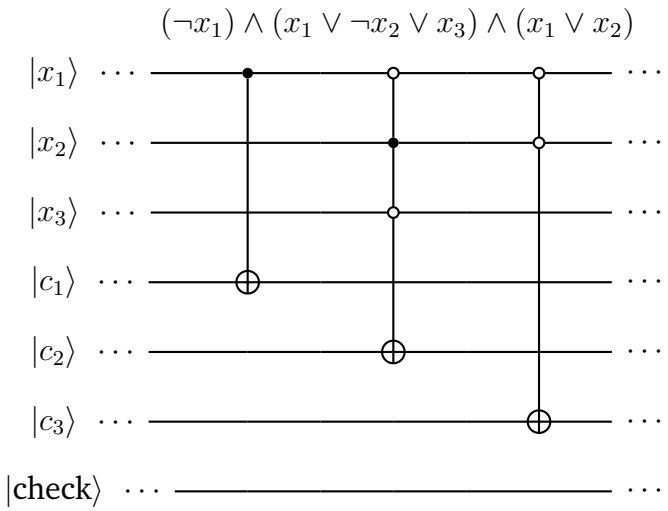
We will explain the construction using a specific formula:

$$(\neg x_1) \wedge (x_1 \vee \neg x_2 \vee x_3) \wedge (x_1 \vee x_2). \tag{4.14}$$

In this case our circuit will need $v + c + 1 = 7$ qubits.

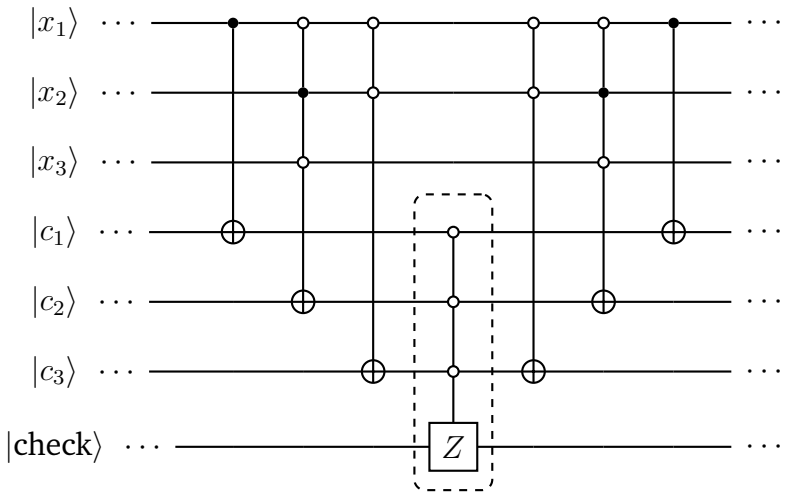
Now, we are going to build the oracle for (4.14). Our oracle will follow Definition (4.1) and with three steps will add a negative amplitude to the state that represents the solution to (4.14).

The Multi-controlled NOT (MCN) gate is quite similar to the MCZ gate. The $\circ$ represents an anti-control (condition on qubit being on state $|0\rangle$), and the $\bullet$ represents a control (condition on qubit being on state $|1\rangle$) and the $\oplus$ represents the NOT gate applied to the target qubit. The first step of the oracle for our CNF formula is shown below.



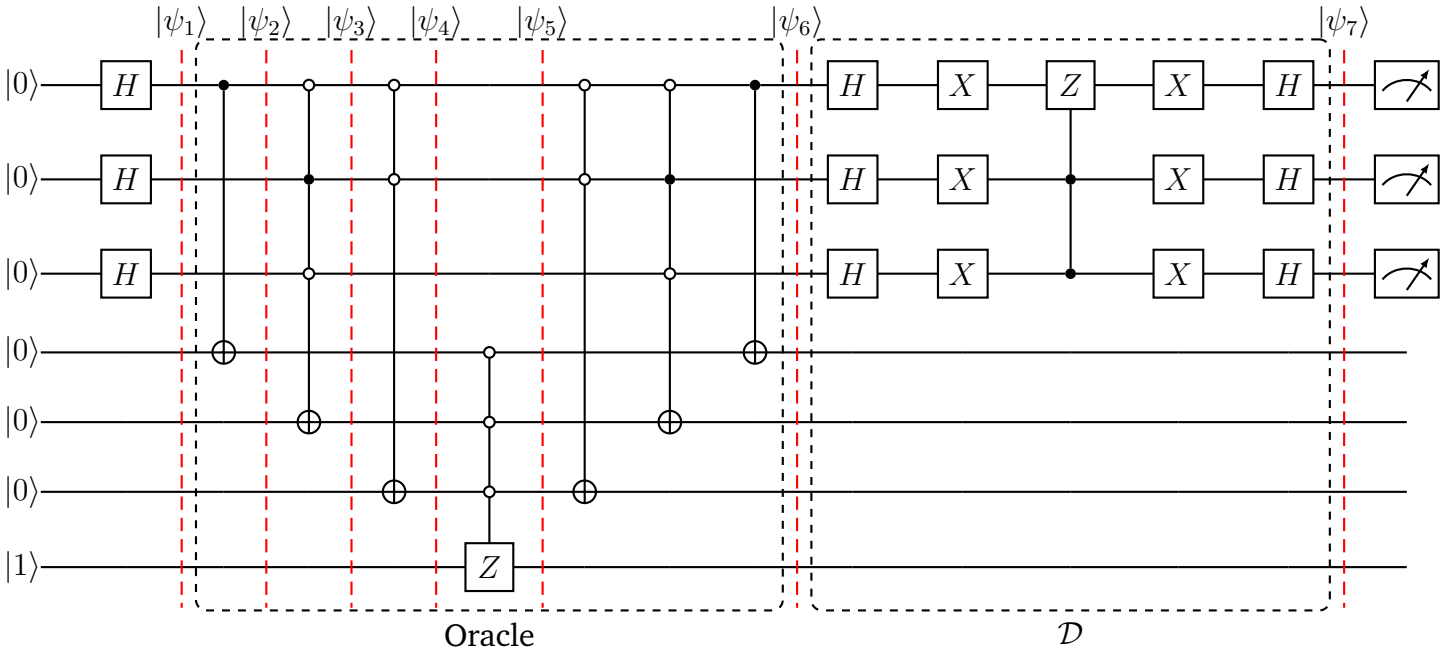
Now for the second step of the oracle, we want to change the amplitude from positive to negative of the states that were not marked in the first step. With a Multi-controlled Z gate we can achieve that by applying a Z gate on the "check" qubit if all the qubits from the second register are in their initial state $|0\rangle$. This implies that the state that satisfies the CNF formula will have a negative amplitude.

The third and final step of the oracle is to "uncompute" the states that were altered by the first step. We have to make this operation because, for the Grover Diffusion Operator to work as expected, the state of the system immediately before this operator must be expressed as a tensor product between the first and second register. We also want to "reset" the qubits from the second register for futures applications of the oracle.



2nd Step

Below is the circuit of Grover's algorithm with one step of the Grover Iteration to solve our sample CNF formula.



Let us explore in detail some important steps of this circuit. The initial state of the system is $|\psi_0\rangle = |000\rangle \otimes |0001\rangle$. Applying a Hadamard transformation on the first register gives the state

$$|\psi_1\rangle = \mathrm{H}^{\otimes 3}|000\rangle \otimes \mathrm{I}^{\otimes 4}|0001\rangle = (\frac{1}{2\sqrt{2}} \sum_{x\in\{0,1\}^3} |x\rangle) \otimes |0001\rangle$$

$$= \frac{1}{2\sqrt{2}}(|0000001\rangle + |0010001\rangle + |0100001\rangle + |0110001\rangle$$

$$+ |1000001\rangle + |1010001\rangle + |1100001\rangle + |1110001\rangle).$$

Now, we will start the first step of the oracle by marking the states that do not satisfy the clauses. For the first clause, we will mark all the states with the first qubit equals to 1 because $(\neg x_1)$ is not satisfied if $x_1 = 1 =$ true.

$$|\psi_2\rangle = \frac{1}{2\sqrt{2}}(|0000001\rangle + |0010001\rangle + |0100001\rangle + |0110001\rangle$$
$$+ |1001001\rangle + |1011001\rangle + |1101001\rangle + |1111001\rangle).$$

For the second clause, the state $|010\rangle$ in the first register represents the assignment $x_1 = 0, x_2 = 1$ and $x_3 = 0$ does not satisfy the clause $(x_1 \vee \neg x_2 \vee x_3)$.

$$|\psi_3\rangle = \frac{1}{2\sqrt{2}}(|0000001\rangle + |0010001\rangle + |0100101\rangle + |0110001\rangle$$
$$+ |1001001\rangle + |1011001\rangle + |1101001\rangle + |1111001\rangle).$$

Similarly, for the third clause, we have

$$|\psi_4\rangle = \frac{1}{2\sqrt{2}}(|0000011\rangle + |0010011\rangle + |0100101\rangle + |0110001\rangle$$
$$+ |1001001\rangle + |1011001\rangle + |1101001\rangle + |1111001\rangle),$$

finishing the first step of the oracle.

As we mentioned earlier in the second step of the oracle we are going to change the amplitude of the state that was not marked in the first step. This state represents an answer to the CNF formula:

$$|\psi_5\rangle = \frac{1}{2\sqrt{2}}(|0000011\rangle + |0010011\rangle + |0100101\rangle - |011 \underbrace{000}_{\substack{\text{unmarked} \\ \text{2nd reg}}} 1\rangle$$
$$+ |1001001\rangle + |1011001\rangle + |1101001\rangle + |1111001\rangle).$$

The third and final step of the oracle is the "uncompute" operation:

$$|\psi_6\rangle = \frac{1}{2\sqrt{2}}(|0000001\rangle + |0010001\rangle + |0100001\rangle - |0110001\rangle$$
$$+ |1000001\rangle + |1010001\rangle + |1100001\rangle + |1110001\rangle)$$
$$= \frac{1}{2\sqrt{2}}(|000\rangle + |001\rangle + |010\rangle - |011\rangle + |100\rangle + |101\rangle + |110\rangle + |111\rangle) \otimes |0001\rangle$$

$$= \left(-\frac{1}{2\sqrt{2}} |011\rangle + \sum_{\substack{x \in \{0,1\}^3 \\ x \neq 011}} \frac{1}{2\sqrt{2}} |x\rangle \right) \otimes |0001\rangle ,$$

ending the oracle phase.

Now we are going to apply the Grover Diffusion Operator on the first register. Calculating μ as defined in (4.3) gives us

$$\mu = \frac{1}{2^3} \sum_{\substack{x \in \{0,1\}^3 \\ y = 0001}} \alpha_{xy} = \frac{1}{2^3} \left(\frac{7}{2\sqrt{2}} - \frac{1}{2\sqrt{2}} \right) = \frac{3}{8\sqrt{2}}.$$

With μ we can apply $\mathcal{D}$ on the first register getting to $|\psi_7\rangle$:

$$\begin{aligned}
|\psi_7\rangle &= (\mathcal{D} \otimes \mathrm{I}^{\otimes 4}) |\psi_6\rangle = \\
&= \left(\mathcal{D} \left(-\frac{1}{2\sqrt{2}} |011\rangle \right) + \mathcal{D} \left(\sum_{\substack{x \in \{0,1\}^3 \\ x \neq 011}} \frac{1}{2\sqrt{2}} |x\rangle \right) \right) \otimes \mathrm{I}(|0001\rangle) \\
&= \left(\left(2\frac{3}{8\sqrt{2}} - \left(-\frac{1}{2\sqrt{2}} \right) \right) |011\rangle + \left(2\frac{3}{8\sqrt{2}} - \frac{1}{2\sqrt{2}} \right) \sum_{\substack{x \in \{0,1\}^3 \\ x \neq 011}} |x\rangle \right) \otimes (|0001\rangle) \\
&= \left(\frac{5}{4\sqrt{2}} |011\rangle + \frac{1}{4\sqrt{2}} \sum_{\substack{x \in \{0,1\}^3 \\ x \neq 011}} |x\rangle \right) \otimes (|0001\rangle).
\end{aligned}$$

When we measure the state $|\psi_7\rangle$ the probability of obtaining the outcome 011, that represents the solution $x_1 = 0, x_2 = 1$ and $x_3 = 1$ to this CNF formula, it is $|5/(4\sqrt{2})|^2 \approx 78\%$. If we execute the Grover Iteration one more time, the probability of obtaining the right outcome increases to about 94%. For this CNF formula with three variables $(n = 3)$, two executions of the Grover Iteration is the ideal number of executions because $\lfloor \pi\sqrt{2^3}/4 \rfloor = 2$.

4.5 Further Observations on Grover's Algorithm

Using an approach like the one presented in the last section we can use Grover's algorithm to solve any 3-SAT formula with n variables in $O(1.414^n)$ time. But, as this problem is not so "unstructured", Schöning's algorithm improved by Rolf [R$^+$03] solves a 3-SAT formula in $O(1.330^n)$ time. Using a generalization of Grover's Algorithm called Amplitude Amplification [BHMT02] we can build a hybrid algorithm that solves a 3-SAT formula in $O(\sqrt{1.330}^{\,n})$ [Amb04, DKW05].

Grover's algorithm can also be used to search for multiple marked strings. If the number of marked strings m out of 2^n possible values is known beforehand, one should apply the Grover Iteration routine $\lfloor(\pi/4)\sqrt{2^n/m}\rfloor$ times before measuring the state of the system. In this scenario all of the m possible solutions have the same amplitude and their sum is the closest to one as possible.

How about when we do not know if there is at least one marked string m, where $0 \leq m \leq 2^n$? In this case we would stick to the following strategy: We start by assuming all strings are marked, running Grover's algorithm with $m = 2^n$ and checking if the measured string is a solution. If that is not the case, we assume half of the strings are marked running Grover's algorithm with $m = 2^{n-1}$ and checking if the measured string is a solution. Following this same procedure with $m = 2^{n-2}$, $m = 2^{n-3}$, and so on until we have found a marked string or have searched unsuccessfully with $m = 1$. Thus, after $O(\sqrt{2^n/m})$ queries we have a high probability of finding a marked string m or concluding that m does not exist.

Part II

Quantum Communication Complexity Protocols and Nonlocality

Chapter 5

Quantum Communication Complexity Protocols

In 1973, many years before quantum computing became an established field, Holevo proved that for any classical message the cost of transmitting it from one party to another in terms of quantum bits, is the same as the cost of transmitting it in terms of classical bits [Hol73a]. After thinking about this result, our intuition that quantum information cannot provide a communication efficiency advantage turns out to be wrong. As we will see in the following sections there are scenarios where the possibility to send and realize operations on qubits can save an exponential amount of communication when compared with a classical scenario.

A communication complexity problem can be described as a game played by Alice and Bob that want to successfully compute a relation with their inputs. Formally, this scenario is usually described by three sets A, B and Z and a relation $\mathcal{R} \subseteq A \times B \times Z$. Alice and Bob are given inputs $a \in A$ and $b \in B$, respectively. None of the players has any information about its partner output. According to a shared protocol, Alice and Bob can make any local computation they want and exchange messages until Bob has sufficient information to announce an output $z \in Z$ s.t $(a, b, z) \in \mathcal{R}$.

The *communication cost* of a protocol is the sum of the lengths of messages (in bits) Alice and Bob exchange on the worst-case inputs a and b. It is important to emphasize that the computations that Alice and Bob can realize locally do not add up to the amount of communication between them. The *deterministic communication complexity* of a problem $\mathcal{R}$ is the cost of the best protocol to compute $\mathcal{R}$ correctly.

We are also interested in the *bounded-error randomized* protocol with error probability $\delta > 0$. Now, the players have access to public random coins and want to

announce an output z that satisfy the relation with probability at least $1-\delta$ for any inputs they are given.

Quantum communication complexity studies scenarios where Alice and Bob have quantum resources, for example, the parties can send each other qubits and perform quantum computations on them. In this chapter, we will introduce two communication complexity problems where quantum players have an advantage over classical players.

5.1 Distributed Deutsch–Jozsa

The first large gap between quantum and classical communication complexity was based on the Deutsch–Jozsa Algorithm3.2.

The Distributed Deutsch–Jozsa solves the following problem:

- Alice receives $x \in \{0,1\}^n$ and Bob receives $y \in \{0,1\}^n$, where n is a power of two.
- Their inputs satisfy the "DJ promise":

  $x = y$ or x and y differ in exactly $n/2$ positions ($d_H(x,y) = n/2$).
- Bob's goal is to determine the relation between x and y after receiving just one message from Alice.

This scenario is called a *one-way* communication complexity problem. Buhrman, Cleve and Wigderson [BCW98] proved, using a combinatorial result of Frankl and Rödl [FR87], that every classical errorless protocol for this problem needs at least $0.007n$ bits of communication. We are going to present a quantum protocol that solves this problem with $\log n$ qubits of communication.

Quantum Protocol for the Distributed Deutsch–Jozsa

(A.0) Alice's initial state is $|\psi_0\rangle = |0\rangle^{\log n} |1\rangle$.

(A.1) Alice applies a Hadamard transform (see (3.3) and (3.4) for reference) in her $\log n + 1$ qubits resulting in

$$|\psi_1\rangle = \mathrm{H}^{(\log n+1)} |0\rangle^{\log n} |1\rangle = \frac{1}{\sqrt{n}} \sum_{j\in\{0,1\}^{\log n}} |j\rangle |-\rangle .$$

It will be useful to express the digits of x with their binary index:

$$x = x_0 \cdots x_{n-1} = x_{\underbrace{00\cdots0}_{\log n}} \cdots x_{\underbrace{11\cdots1}_{\log n}}.$$

For steps A.2 and A.3 we are going to use a phase oracle. More details can be found in Section 3.1.

(A.2) Let U_x be the unitary operator defined by Alice using her input x as:

$$U_x \left|j, q\right\rangle = \left|j, q \oplus x_j\right\rangle, \text{ for all } j \in \{0,1\}^{\log n} \text{ and } q \in \{0,1\}. \tag{5.1}$$

(A.3) Alice applies U_x on $\left|\psi_1\right\rangle$ resulting in

$$\left|\psi_2\right\rangle = U_x \left|\psi_1\right\rangle = \frac{1}{\sqrt{n}} \sum_{j \in \{0,1\}^{\log n}} (-1)^{x_j} \left|j\right\rangle \left|-\right\rangle. \tag{5.2}$$

For reference on this step see the phase kick-back construction in (3.2).

(A.4) Alice sends to Bob the first $\log n$ qubits of $\left|\psi_2\right\rangle$, that is $(\sum_{j \in \{0,1\}^{\log n}} (-1)^{x_j} \left|j\right\rangle)/\sqrt{n}$.

(B.0) Let U_y be the unitary operator defined by Bob, in the same manner, Alice defined U_x in A.2 but using his input y.

(B.1) Using an auxiliary qubit set to $\left|-\right\rangle$, Bob applies U_y on $\left|\psi_2\right\rangle \left|-\right\rangle$ resulting in

$$\begin{aligned}
\left|\psi_3\right\rangle &= U_y \left(\frac{1}{\sqrt{n}} \sum_{j \in \{0,1\}^{\log n}} (-1)^{x_j} \left|j\right\rangle \left|-\right\rangle \right) \\
&= \frac{1}{\sqrt{n}} \sum_{j \in \{0,1\}^{\log n}} (-1)^{x_j} U_y(\left|j\right\rangle \left|-\right\rangle) \\
&= \frac{1}{\sqrt{n}} \sum_{j \in \{0,1\}^{\log n}} (-1)^{x_j \oplus y_j} \left|j\right\rangle \left|-\right\rangle.
\end{aligned}$$

From now on we are going to ignore the auxiliary qubit $\left|-\right\rangle$.

Before moving on to the next step let us remember the result of $\mathrm{H}^{\log n} \left|b\right\rangle$, where $b \in \{0,1\}^{\log n}$:

$$\mathrm{H}^{\log n} \left|b\right\rangle = \frac{1}{\sqrt{n}} \sum_{z \in \{0,1\}^{\log n}} (-1)^{b \bullet z} \left|z\right\rangle.$$

In (3.6) we explained how to get to this equality.

(B.2) Bob applies a Hadamard transform in each one of the qubits of $\left|\psi_3\right\rangle$ resulting in:

$$\left|\psi_4\right\rangle = \mathrm{H}^{\log n} \left|\psi_3\right\rangle = \frac{1}{n} \sum_{j \in \{0,1\}^{\log n}} (-1)^{x_j \oplus y_j} \sum_{z \in \{0,1\}^{\log n}} (-1)^{j \bullet z} \left|z\right\rangle \tag{5.3}$$

The trick of this proof is to analyze the amplitude of the $|0\rangle^{\log n}$ state in $|\psi_4\rangle$. Rearranging the terms in (5.3) give us:

$$|\psi_4\rangle = \frac{1}{n}\Big(\sum_{j\in\{0,1\}^{\log n}} (-1)^{x_j\oplus y_j} |0\rangle^{\log n} + \sum_{j\in\{0,1\}^{\log n}} (-1)^{x_j\oplus y_j} \sum_{\substack{z\in\{0,1\}^{\log n} \\ z\neq|0\rangle^{\log n}}} (-1)^{j\bullet z} |z\rangle \Big).$$

Calculating the amplitude of the state $|0\rangle^{\log n}$ considering the "DJ promise":

- if $x = y \implies \frac{1}{n} \sum_{j\in\{0,1\}^{\log n}} (-1)^{x_j\oplus y_j} = \frac{1}{n} \underbrace{\sum_{j\in\{0,1\}^{\log n}} (-1)^0}_{n} = 1.$
- if $d_H(x,y) = n/2 \implies \frac{1}{n} \sum_{j\in\{0,1\}^{\log n}} (-1)^{x_j\oplus y_j} = \frac{1}{n}\big(\underbrace{\frac{n}{2}}_{x_j=y_j} \underbrace{-\frac{n}{2}}_{x_j\neq y_j} \big) = 0.$

(B.3) For the final step of the protocol Bob measures $|\psi_4\rangle$:

If he obtains the $00\cdots0$ $\log n$-bit string he concludes that $x = y$.

If he obtains any other $\log n$-bit string he concludes that $d_H(x,y) = n/2$.

The only communication that happens in this protocol is on step A.4 when Alice sends a $\log n$ qubits state to Bob. Providing an exponential separation between the quantum and classical protocols.

Nevertheless, as was noted in [BCMdW10], the exponential separation between the quantum and classical communication complexity disappears if we consider a classical protocol on the bounded-error randomized setup. In this other scenario $O(\log n)$ classical bits suffice to determine the relation between x and y (Section 3.4 of [BCMdW10]).

5.2 Hidden Matching Problem

In 2004 Bar-Yossef, Jayram and Kerenidis [BYJK04] discovered a quantum one-way protocol that solves the Hidden Matching problem with Alice sending a single message of size $O(\log n)$ qubits to Bob. They also proved that any classical bounded-error randomized one-way protocol needs $\Omega(\sqrt{n})$ bits of communication. This was the first exponential separation between quantum and bounded-error randomized one-way communication complexity.

As presented in [BYJK04] consider the following problem:

Let n be a positive even integer. In the Hidden Matching Problem, denoted HM_n, Alice is given $x \in \{0,1\}^n$ and Bob is given $M \in \mathcal{M}_n$ (where $\mathcal{M}_n$ denotes the family of all possible perfect matchings on n nodes). Their goal is to output a tuple $\langle i, j, b\rangle$ such that the edge $\{i, j\}$ belongs to the matching M and $b = x_i \oplus x_j$.

Before we introduce the quantum protocol for this problem we are going to define another type of measurement. Let us first note that a square matrix P is a *projection matrix* if $P = P^2$.

Definition 5.2.1 - Complete Projective Measurement
Let $|\psi\rangle$ be a n-qubit state and $\mathbb{B} = \{|b_1\rangle, \ldots, |b_{2^n}\rangle\}$ an orthonormal basis of the n-qubit space. A measurement of $|\psi\rangle$ in basis $\mathbb{B}$ means that we apply the projection operators $P_j = |b_j\rangle\langle b_j|$, $1 \leq j \leq 2^n$ to $|\psi\rangle$. The outcome of this measurement is j and the state of the system collapses to $|b_j\rangle$ with probability $p_j = |\langle\psi|b_j\rangle|^2$.

More details of this type of measurement can be found in Section 9.1.

Now, we will present the quantum protocol for HM_n and later simulate this protocol.

Quantum Protocol for the Hidden Matching Problem:

Input:

Alice receives $x \in \{0,1\}^n$ and Bob receives $M \in \mathcal{M}_n$, where n is a positive even integer.

(A.1) Alice prepares locally the $\log n$ qubit state

$$|\psi\rangle = \frac{1}{\sqrt{n}} \sum_{i=1}^{n} (-1)^{x_i} |i\rangle ,$$

and sends $|\psi\rangle$ to Bob. For reference on how to prepare this state see steps A.0 to A.3 on the previous protocol.

Bob views his input M as an orthogonal decomposition of the space $\mathbb{C}^n$ into $n/2$ 2-dimensional subspaces.

(B.1) Bob performs a projective measurement on $|\psi\rangle$ in the basis $\mathbb{B} = \left\{ \frac{1}{\sqrt{2}}(|k\rangle \pm |l\rangle) \mid \{k, l\} \in M \right\}$.

The probability that the outcome of the measurement is a basis state $(|k\rangle + |l\rangle)/\sqrt{2}$ is

$$\left| \left\langle \psi \middle| \frac{1}{\sqrt{2}}(|k\rangle + |l\rangle) \right\rangle \right|^2 = \frac{1}{2n} \left((-1)^{x_k} + (-1)^{x_l}\right)^2 = \begin{cases} 2/n, & \text{if } x_k \oplus x_l = \mathbf{0} \\ 0, & \text{otherwise.} \end{cases}$$

If that is the case Bob output the tuple $\langle k, l, \mathbf{0} \rangle$.

Similarly, the probability that the outcome of the measurement is a basis state $(|k\rangle - |l\rangle)/\sqrt{2}$ is

$$\left| \left\langle \psi \right| \frac{1}{\sqrt{2}}(|k\rangle - |l\rangle) \right\rangle \Big|^2 = \frac{1}{2n}\left((-1)^{x_k} - (-1)^{x_l}\right)^2 = \begin{cases} 2/n, & \text{if } x_k \oplus x_l = \mathbf{1} \\ 0, & \text{otherwise.} \end{cases}$$

In this case Bob output the tuple $\langle k, l, \mathbf{1} \rangle$.

The only communication in this protocol is on step A.1 when Alice sends a $\log n$ qubits state to Bob.

Simulation of the Quantum Protocol for the Hidden Matching Problem:

Input: $x = 1011$ and $M = \{\{1,3\},\{2,4\}\}$.

(A.1) Alice prepares and sends $|\psi\rangle = (-|1\rangle + |2\rangle - |3\rangle - |4\rangle)/2$ to Bob.
Bob's decomposition of $\mathbb{C}^4$ is $\mathbb{B} = \left\{(|1\rangle \pm |3\rangle)/\sqrt{2}, (|2\rangle \pm |4\rangle)/\sqrt{2}\right\}$.

(B.1) Probability measures $|\psi\rangle$ in basis $\mathbb{B}$ and observe:

$\frac{1}{\sqrt{2}}(|1\rangle + |3\rangle)$ is: $|\langle\psi| \frac{1}{\sqrt{2}}(|1\rangle + |3\rangle)|^2 = \frac{1}{2} \implies x_1 \oplus x_3 = \mathbf{0}$ Bob output $\langle 1, 3, 0\rangle$.

or

$\frac{1}{\sqrt{2}}(|2\rangle - |4\rangle)$ is: $|\langle\psi| \frac{1}{\sqrt{2}}(|2\rangle - |4\rangle)|^2 = \frac{1}{2} \implies x_2 \oplus x_4 = \mathbf{1}$ Bob output $\langle 2, 4, 1\rangle$.

Following Section 2.3.3 of [Sca13] we will explain his classical protocol for the HM_n with added details proving the following theorem:

Theorem 5.2.2 [Sca13]
For every n that is a perfect square, and every positive integer $\sqrt{c} \leq n$, there exists a classical protocol for HM_n with c bits of one-way communication, such that for all inputs x, M,

$$\mathbb{P}(b = x_i \oplus x_j) = \frac{1}{2} + \Omega(\frac{c}{\sqrt{n}}).$$

Proof. Assume for simplicity that c is even and sufficiently large. Using shared random variables that were generated before the game started Alice and Bob define two disjoint subsets S_1 and S_2 of $[n]$ each one of size $\sqrt{n}$:

$$S_1 = \{s_1, \ldots, s_{\sqrt{n}}\},\ S_2 = \{s'_1, \ldots, s'_{\sqrt{n}}\}.$$

Let y and z be two $\sqrt{n}$-bit strings defined as:

$$y = x_{s_1} \cdots x_{s_{\sqrt{n}}} \text{ and } z = x_{s'_1} \cdots x_{s'_{\sqrt{n}}}.$$

Using shared randomness the players produce $2^{c/2}$ random $\sqrt{n}$-bit strings $y^{(1)}, \ldots, y^{(2^{c/2})}$. For each l, $1 \leq l \leq 2^{c/2}$, the distance $d(y, y^{(l)})$ is distributed binomially as the sum of $\sqrt{n}$ fair coin flips.

For the rest of the proof we are interested in the distance between y and $y^{(l)}$ so we will state the following fact about the tail of the binomial distribution: There exists a universal constant $\gamma > 0$ such that if X is the sum of k fair coin flips. Then, for all $0 < \beta < \sqrt{k}/2$ we have

$$\mathbb{P}(X \leq k/2 - \beta\sqrt{k}) \geq 2^{-\gamma(1+\beta^2)}.$$

Using this fact with our notation gives

$$\mathbb{P}(d(y, y^{(l)}) \leq \sqrt{n}/2 - \beta n^{1/4}) \geq 2^{-\gamma(1+\beta^2)}. \tag{5.4}$$

Let us define a *bad event* when $d(y, y^{(l)}) > \sqrt{n}/2 - \beta n^{1/4}$. The probability of a bad event is the complement of (5.4), that it is at most $1 - 2^{-\gamma(1+\beta^2)}$. Considering $y^{(1)}, \ldots, y^{(2^{c/2})}$ we are going to estimate the following probability:

$$\mathbb{P}(\text{all } 2^{c/2} \text{ bit strings } y^{(l)} \text{ result in bad events}) = (1 - 2^{-\gamma(1+\beta^2)})^{2^{c/2}}.$$

Using the fact that $1 - x \leq e^{-x}$ for all $x \in \mathbb{R}$ we can give an upper bound of the probability above:

$$(1 - 2^{-\gamma(1+\beta^2)})^{2^{c/2}} \leq \exp(-(2^{-\gamma(1+\beta^2)})^{2^{c/2}}) = \exp(-(2^{-\gamma(1+\beta^2)+c/2})).$$

Analyzing the term $-\gamma(1 + \beta^2) + c/2$ and choosing $\beta = \Theta(\sqrt{c})$ gives

$$-\gamma(1 + \beta^2) + \frac{c}{2} = -\gamma + \gamma\delta^2 c + \frac{c}{2} \geq -\gamma\delta^2 c + \frac{c}{3},$$

taking $\delta = 1/(2\sqrt{\gamma})$ it follows that

$$-\gamma(1 + \beta^2) + \frac{c}{2} \geq \frac{c}{12}.$$

Therefore, the probability that all of the $2^{c/2}$ bit strings $y^{(l)}$ result in bad events is close to zero.

We can conclude that with probability close to one there will be an l such that y and $y^{(l)}$ are at normalized distance $\leq 1/2 - \Omega(c^{1/2}/n^{1/4})$. We will say that each l that satisfy this inequality is a *good event*. To prove this statement note that:

$$d_N(y, y^{(l)}) \leq (\frac{\sqrt{n}}{2} - \beta n^{1/4}) \underbrace{n^{-1/2}}_{normalization} = \frac{\sqrt{n}}{2} - \frac{\beta}{n^{1/4}} = \frac{\sqrt{n}}{2} - \frac{\delta\sqrt{c}}{n^{1/4}} = \frac{1}{2} - \Omega\left(\frac{c^{1/2}}{n^{1/4}}\right).$$

For the next step of the protocol Alice sends the first l that correspond to a good event to Bob. Otherwise, she communicates to Bob that there is no such l. Now, using the $\sqrt{n}$-bit string z she does the same procedure sending the first l' that correspond to a good event or communicating if there is no such l'. In this step she will send at most $c/2 + c/2 = c$ bits of communication to Bob.

Before we continue the protocol we will use the following proposition: With probability at least $1/2$, Bob's matching M contains an $\{i, j\}$ with $i \in S_1$ and $j \in S_2$. The proof of this proposition is in Appendix A

Bob's Matching Theorem.

Given that Alice sent l and l' to Bob we will calculate the probability that he can predict x_i from $y^{(l)}$, from any $i = s_\alpha$, $1 \leq \alpha \leq \sqrt{n}$:

$$\mathbb{P}\left(y_i^{(l)} = y_i \mid d_N(y^{(l)}, y)\right) = 1 - d_N(y^{(l)}, y) \leq 1 - \left(\frac{1}{2} - \Omega\left(\frac{c^{1/2}}{n^{1/4}}\right)\right) = \frac{1}{2} + \Omega\left(\frac{c^{1/2}}{n^{1/4}}\right).$$

Bob can predict x_j, from any $j = s'_\beta$, $1 \leq \beta \leq \sqrt{n}$ with the same probability.

Finally, we have to show that Bob can predict $x_i \oplus x_j$ with probability $1/2 + \Omega(c/\sqrt{n})$. There are two scenarios where Bob successfully predicts $x_i \oplus x_j$; when his individual predictions of x_i and x_j are both correct and also when they are both wrong. These two scenarios give us the following:

$$\begin{aligned}
\mathbb{P}\left(y_\alpha^{(l)} \oplus z_\beta^{(l')} = y_\alpha \oplus z_\beta\right) &= \mathbb{P}\left(y_\alpha^{(l)} = y_\alpha \text{ and } z_\beta^{(l')} = z_\beta\right) + \mathbb{P}\left(y_\alpha^{(l)} \neq y_\alpha \text{ and } z_\beta^{(l')} \neq z_\beta\right) \\
&= \left(\frac{1}{2} + \Omega\Big(\frac{c^{1/2}}{n^{1/4}}\Big)\right)^2 + \left(1 - \Big(\frac{1}{2} + \Omega(\frac{c^{1/2}}{n^{1/4}})\Big)\right)^2 \\
&= \frac{1}{2} + 2\Omega\left(\frac{c}{n^{1/2}}\right) \\
&= \frac{1}{2} + 2\delta\frac{c}{n^{1/2}},
\end{aligned}$$

taking $\delta = 1/2$ gives the desired probability. In this case Bob's output $y_\alpha^{(l)} \oplus z_\beta^{(l')}$.

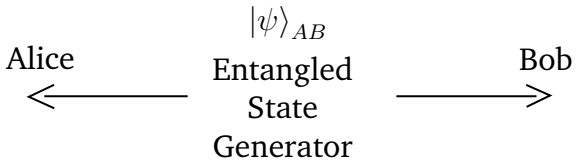
Considering the case where Bob's matching M does not contain an $\{i, j\}$ with $i \in S_1$ and $j \in S_2$ or he did not get a good approximation of y and z. In this case he output a random bit winning with probability $1/2$. Considering these two cases this protocol wins with probability $1/2 + \Omega(c/\sqrt{n})$. ■

Chapter 6

Nonlocality

Quantum nonlocality refers to the scenario where the results of local measurements carried out on an entangled system are somehow correlated. This phenomenon as was proven by Bell [Bel64] violates the principle of locality that states that an object is directly influenced only by its immediate surroundings. Bell's motivation was to prove that quantum mechanics is not a classical theory that depends on hidden variables.

Consider the following experiment that we can realize with current technology:



The entangled state generator produces the state $|\psi\rangle_{AB} = (|0\rangle_A |0\rangle_B + |1\rangle_A |1\rangle_B)/\sqrt{2}$ and proceed to send the first qubit to Alice and the second qubit to Bob that are far away from each other. Now, Alice measures her qubit obtaining a random bit 0 or 1 with the same probability. After this measurement, the system evolves according to the Measurement Postulate (2.6) and whenever Bob measures his qubit he will observe the same result that Alice got on her measurement.

It is important to note that Bell also proved that quantum nonlocality does not allow for faster-than-light communication [Bel64]. Hence, in our experiment (or in any experiment at all) there is no way that Alice can manipulate her qubit to send a message to Bob faster than the speed of light.

Since Bell's work, quantum nonlocality has been the subject of much further theoretical and experimental results. We start our presentation of quantum nonlocality by introducing a nonlocal game (also called pseudo-telepathy game) as they provide an intuitive way to understand the concepts of nonlocality.

6.1 CHSH Game

In the CHSH game two players, Alice and Bob, receive input bits x and y from a referee. Their goal is to send back to the referee bits a and b that satisfy the equation

$$a \oplus b = x \wedge y. \tag{6.1}$$

After the players receive their inputs they cannot communicate with each other.

Now that we have defined the CHSH game we will first study how Alice and Bob can play this game using only classical resources. A *randomized strategy* consists of the following. Before the game starts the players can decide on any protocol and share any random variables they want. After the game starts without any communication the players can use their inputs, shared random variables and have unlimited classical computational power that they can use to compute their outputs. We are going to say that Alice and Bob *win the CHSH game with probability* p if their strategy satisfies Equation (6.1) with probability at least p for all possible inputs. We note that a *deterministic strategy* can be considered as a randomized strategy that the players always play with the same fixed shared "random" variable. This implies that they will always play with the same strategy. For this reason in the rest of this chapter both of these scenarios will be called *classical strategies*.

We will show next that there is no deterministic strategy that always succeeds. Since Alice cannot communicate with Bob her output bit a can only depend on the value of her input bit x. Let a_0 (a_1) be Alice's output when her input bit is 0 (1). Similarly, let b_0 and b_1 be Bob's output depending on his input. Note that these four bits completely characterize any deterministic strategy of Alice and Bob. Equation (6.1) translate into the the equations

$$a_0 \oplus b_0 = 0, \tag{6.2}$$

$$a_0 \oplus b_1 = 0, \tag{6.3}$$

$$a_1 \oplus b_0 = 0, \tag{6.4}$$

$$a_1 \oplus b_1 = 1. \tag{6.5}$$

It is impossible to satisfy all four equations simultaneously since summing them modulo 2 yields $0 = 1$. Hence, for any deterministic strategy there exists an input configuration for which it fails. However, for any three out of the four equations above there is a strategy that satisfies these three equations simultaneously for any input.

Now that we have defined these concepts we will prove the following theorem.

Theorem 6.1.1
No classical strategy allows Alice and Bob to win the CHSH game with probability greater than $3/4$.

Proof. We will start by formalizing a randomized strategy in the following way. Before the game starts the players can share a random B-bit string. After receiving their inputs they can use this bit string and play accordingly. A classical strategy S denoted by $S = (a, b)$ is defined by two functions a and b, let

$$a : \{0,1\} \times \{0,1\}^B \to \{0,1\} \qquad \text{and} \qquad b : \{0,1\} \times \{0,1\}^B \to \{0,1\}.$$

For the next definition we will use the following notation. A bit string r of length B taken uniformly at random from the set $\{0,1\}^B$ will be denoted as $r \in_u \{0,1\}^B$. The notation $S^{(x,y)}$ represents the event that a classical strategy S satisfies Equation (6.1) with the input (x, y). Using these definitions, let

$$\textit{winning probability of } S = \min_{(x,y)\in\{0,1\}^2} \mathbb{P}_{r\in_u\{0,1\}^B}(S^{(x,y)}).$$

The notation $\mathbb{P}_{r\in_u\{0,1\}^B}(S^{(x,y)})$ represents the probability of $S^{(x,y)}$ over all values of r. We will assume that before the game starts Alice and Bob share r.

Now, we will prove two propositions:

(A) There exists an S such that the winning probability of $S \geq 3/4$.

Let $B = 2$ and $x, y \in \{0,1\}$. Consider a strategy $S = (a, b)$, let

$$a(x, 00) = x, \qquad a(x, 01) = 1, \qquad a(x, 10) = x, \qquad a(x, 11) = 0$$

and

$$b(y, 00) = \neg y, \qquad b(y, 01) = \neg y, \qquad b(y, 10) = 0, \qquad b(y, 11) = 0.$$

The winning probability of S for $x = 0$ and $y = 0$ is equal to $3/4$ because this strategy will only lose with this input if $r = 00$. Doing the same analysis for the other tree inputs we get the same probability. Hence, we can conclude that the winning probability of S is greater than or equal to $3/4$.

(B) For all S the winning probability of $S \leq 3/4$.

Because of the structure of the game there are in total 16 deterministic strategies $S_1, \ldots, S_{16}$ that the players can play with. After the players generate their random bit string they are going to play with one of these 16 deterministic

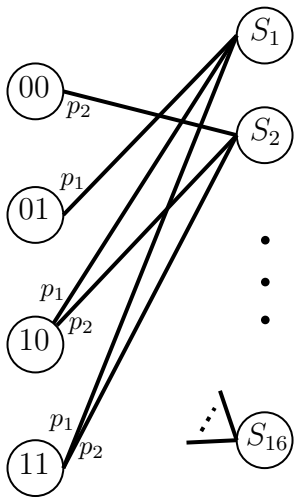
strategies. We can conclude that any probabilistic strategy is a probability distribution over these 16 strategies. Consider the variables $p_1, \ldots, p_{16}$, where

$$0 \leq p_1 \leq 1, i = 1, \ldots, 16 \qquad \text{and} \qquad \sum_{i=1}^{16} p_i = 1.$$

Any probabilistic strategy can be specified by these p_i's that are the probability that this strategy will play with S_i. It follows that for any input (x, y)

$$\mathbb{P}(S^{(x,y)}) = p_1 \mathbf{1}(S_1^{(x,y)}) + \cdots + p_{16} \mathbf{1}(S_{16}^{(x,y)}).$$

We will show that for an input (x, y) the probability above is at most $3/4$. Consider the weighted bipartite graph below:



The edges of the graph were added in the following way. If the strategy S_i satisfies Equation (6.1) with the input (x, y) then there is an edge $e = (S_i, xy)$ with weight $p_i(e)$. We will double count the sum of all the edges in the graph. As each strategy S_i wins the game with at most 3 inputs we have that

$$\sum_{e \in E} p(e) \leq 3(p_1 + \cdots + p_{16}) = 3.$$

Now, counting from the other perspective we have that

$$\sum_{x,y \in \{0,1\}^2} \sum_{e \in \partial(x,y)} p(e) = \mathbb{P}(S^{(0,0)}) + \cdots + \mathbb{P}(S^{(1,1)}).$$

Hence, we can conclude that there is an (x, y) such that the probability that S wins with this input is at most $3/4$. Therefore, from (A) and (B) we can conclude that there is no classical strategy that allows Alice and Bob to win the CHSH game with probability greater than $3/4$. ■

Statements such as Theorem 6.1.1 that prove an upper bound on the optimal success probability of classical strategies for a specific nonlocal game are known as Bell Inequalities. This specific one is called the CHSH inequality. Next, we will see a so-called "violation of a Bell Inequality".

Let us consider a quantum strategy for the CHSH game. Alice and Bob play the same game, but before receiving their inputs they share the 2-qubit entangled state $|\psi\rangle_{AB} = (|0\rangle_A |0\rangle_B - |1\rangle_A |1\rangle_B)/\sqrt{2}$ and carry out the following strategy. Let $R(\theta)$ be the unitary operator that rotates a qubit by an angle θ:

$$R(\theta) = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix}.$$

Depending on the input that Alice receives she applies one of the following rotations: if $x = 0$ she applies $R(-\pi/16)$, if $x = 1$ she applies $R(3\pi/16)$. Let a represent the bit that Alice gets after measuring her qubit. Bob performs the same procedure depending on the value of y. Let b represent the bit he gets after measuring his qubit. Analyzing the state $|\psi\rangle_{AB}$ after Alice rotates her qubit by θ_A and Bob rotates his qubit by θ_B, the entangled state becomes

$$|\psi'\rangle_{AB} = \frac{1}{\sqrt{2}}\big(\cos(\theta_A + \theta_B)(|00\rangle_{AB} - |11\rangle_{AB}) + \sin(\theta_A + \theta_B)(|01\rangle_{AB} - |10\rangle_{AB})\big).$$

We can summarize the probability of success of the four possible inputs when Alice and Bob measure their parts of $|\psi'\rangle_{AB}$ with the following table:

Inputs (s,t)	Correct Output(s) (a,b)	Probability of Success
$0,0$	$0,0$ or $1,1$	$2(\cos(-\pi/8)/\sqrt{2})^2 = \cos^2(\pi/8)$
$1,0$ or $0,1$	$0,0$ or $1,1$	$2(\cos(\pi/8)/\sqrt{2})^2 = \cos^2(\pi/8)$
$1,1$	$0,1$ or $1,0$	$2(\sin(3\pi/8)/\sqrt{2})^2 = \sin^2(3\pi/8) = \cos^2(\pi/8)$

For all inputs the equation $a \oplus b = x \wedge y$ is satisfied with probability $\cos^2(\pi/8) \approx 0.853$. This result shows us that this quantum strategy for the CHSH game succeeds with a higher probability[1] than the best possible classical strategy.

6.2 Structure of Nonlocal Game and Other Examples

The basic structure of nonlocal games can be described as:

[1] Tsirelson [Cir80] proved that a quantum protocol can't achieve a success probability higher than $\cos^2(\pi/8)$ for the CHSH game.

- There is an honest referee that sends the inputs and analyzes the outputs of the players. It is also possible to formulate these games without a referee assuming that the players are honest.
- The player's goal is to compute some sort of relation with their inputs that satisfy some condition.
- Before the game starts the players can decide on any strategy they want.
  - Classical players can share any random variables they want and play accordingly.
  - Quantum players can share any entangled state.
- After the players receive their inputs they cannot communicate with each other.
  - Classical players can make any classical computation using their inputs and random variables.
  - Quantum players can make any quantum operation using their inputs and part of the entangled state.
- The players send their outputs to the referee who computes the result of the game.

In contrast with the CHSH game, other nonlocal games such as the GHZ (Greenberger–Horne–Zeilinger) and the Mermin–Peres Magic Square games have quantum protocols that always succeed. The upper bound on the probability of success of their classical counterparts is shown on the rightmost column.

Game	Players	Input Condition	Output Goal	Best Classical Strategy $\mathbb{P}$ of Success
GHZ	3	bits: s, t, u $s \oplus t \oplus u = 0$	bits: a, b, c $a \oplus b \oplus c = \begin{cases} 0, \text{ if } stu = 000 \\ 1, \text{ if } stu \in \{011, 101, 110\}. \end{cases}$	$3/4$
Mermin–Peres Magic Square	2	$s, t \in \{1, 2, 3\}$	Two 3-bit strings $a_1a_2a_3$ and $b_1b_2b_3$: $\begin{cases} a_1 \oplus a_2 \oplus a_3 = 0 \\ b_1 \oplus b_2 \oplus b_3 = 1 \\ a_t = b_s \end{cases}$	$8/9$

We are going to use the convention that a game in which quantum players by using entanglement have some sort of advantage over classical players is called a nonlocal or pseudo-telepathy game.

Chapter 7

Nonlocal Quantum Communication Complexity Protocols

There are some similarities between nonlocal games and communication complexity problems. In both topics the goal of the players is to provide an output that satisfy some relation given their inputs. On nonlocal games the players have access to an entangled state but are not allowed to communicate after the game starts. On communication complexity problems the players do not have access to an entangled state but can communicate with each other by sending bits or qubits.

From a communication complexity problem we can derive a nonlocal game by forbidding any kind of communication between the players. For instance, we could consider a scenario where before the game starts classical players are allowed to share any kind of random variables and quantum players can share any entangled state. The Nonlocal Distributed Deutsch–Jozsa is an example of this scenario.

Another situation that we could compare is to replace a quantum communication channel with a classical communication channel and allow the quantum players to share an entangled state. In this chapter we will study quantum protocols for these two scenarios.

7.1 Nonlocal Distributed Deutsch–Jozsa

Consider the Distributed Deutsch–Jozsa (Section 5.1), but now Alice and Bob share an entangled state of $\log n$ qubits and, instead of a quantum communication channel, they only have access to a classical communication channel. In this variation they can solve the Distributed Deutsch–Jozsa problem using $\log n$ classical bits of communication.

To understand this other scenario, let us first introduce the Nonlocal Distributed Deutsch–Jozsa problem and the quantum protocol to solve this problem due to Brassard, Cleve and Tapp [BCT99].

- Alice receives $x \in \{0,1\}^n$ and Bob receives $y \in \{0,1\}^n$, where n is a power of two.
- Their inputs satisfy the "DJ promise":

  $x = y$ or x and y differ in exactly $n/2$ positions ($d_H(x,y) = n/2$).
- Their goal is to provide output $a, b \in \{0,1\}^{\log n}$ without using any communication channel. Such that:

  if $x = y$ then $a = b$ or if $d_H(x,y) = n/2$ then $a \neq b$.

Quantum Protocol for the Nonlocal Distributed Deutsch–Jozsa:

Before the game starts Alice and Bob share the entangled state

$$|\psi_0\rangle_{AB} = \frac{1}{\sqrt{n}} \sum_{j\in\{0,1\}^l} \underbrace{|j\rangle\,|-\rangle}_{Alice} \underbrace{|j\rangle\,|-\rangle}_{Bob},$$

where $l = \log n$.

The first $l+1$ qubits of $|\psi_0\rangle$ belongs to Alice and the last $l+1$ qubits of $|\psi_0\rangle$ to Bob. Now, Alice and Bob go to separate locations and cannot communicate with each other.

Game Starts:

Alice receives her input x and applies the unitary U_x as defined in (5.1) to her $l+1$ qubits. Bob does the same operation but using his input y. After this step the entangled state becomes

$$\begin{aligned} |\psi_1\rangle_{A,B} &= \frac{1}{\sqrt{n}} \sum_{j\in\{0,1\}^l} (-1)^{x_j} |j\rangle\,|-\rangle\,(-1)^{y_j} |j\rangle\,|-\rangle \\ &= \frac{1}{\sqrt{n}} \sum_{j\in\{0,1\}^l} (-1)^{x_j\oplus y_j} |j\rangle\,|-\rangle\,|j\rangle\,|-\rangle\,. \end{aligned}$$

For the next step both of them apply a Hadamard transform on their l qubits resulting in:

$$\begin{aligned} |\psi_2\rangle_{A,B} &= \frac{1}{\sqrt{n}} \sum_{j\in\{0,1\}^l} (-1)^{x_j\oplus y_j} \Big(\frac{1}{\sqrt{n}} \sum_{a\in\{0,1\}^l} (-1)^{j\bullet a} |a\rangle\,|-\rangle\Big)\Big(\frac{1}{\sqrt{n}} \sum_{b\in\{0,1\}^l} (-1)^{j\bullet b} |b\rangle\,|-\rangle\Big) \\ &= \frac{1}{n\sqrt{n}} \sum_{a\in\{0,1\}^l} \sum_{b\in\{0,1\}^l} \Big(\sum_{j\in\{0,1\}^l} (-1)^{x_j\oplus y_j\oplus j\bullet(a\oplus b)} |a\rangle\,|-\rangle\,|b\rangle\,|-\rangle\Big). \end{aligned} \tag{7.1}$$

Now, let us calculate the probability that Alice observes an l-bit string a and Bob observes an l-bit string b when each one measures their first l qubits of the entangled state $|\psi_2\rangle_{A,B}$:

$$\mathbb{P}(a,b \mid x,y) = \frac{1}{n^3}\Big(\sum_{j\in\{0,1\}^l} (-1)^{x_j \oplus y_j \oplus j\bullet(a\oplus b)} \Big)^2. \tag{7.2}$$

Therefore,

$$\mathbb{P}(a,b \mid x=y) = \begin{cases} \frac{1}{n}, & \text{if } a=b \\ 0, & \text{if } a\neq b \end{cases} \quad \text{(there are } n \text{ different bit strings of size } \log n)$$

and

$$\mathbb{P}(a,b \mid d_H(x,y)=n/2) = \begin{cases} 0, & \text{if } a=b \\ \frac{1}{n}, & \text{if } a\neq b. \end{cases} \quad \begin{array}{l}\text{(half of the amplitudes will be equal to } 1 \\ \text{and the other half equal to } -1)\end{array}$$

This analysis shows that they always provide the right output, thus concluding the protocol. The authors also proved that there is no way that classical players can win this nonlocal game that is, without communication.

To solve the variation of the Distributed Deutsch–Jozsa presented at the beginning of this section. Alice and Bob first execute this nonlocal protocol. Then, Alice sends the result of her measurement the $\log n$-bit string a to Bob. Now, he could solve the Distributed Deutsch–Jozsa by comparing their bit strings. In this variation the quantum players solved the Distributed Deutsch–Jozsa with $\log n$-bits of communication against the $0.007n$ bits of communication necessary in the classical case.

As presented in [BCMdW10] shared entanglement can sometimes be used to reduce the amount of communication or even eliminate the necessity of a quantum channel between the parties.

7.2 x-Pairs Nonlocal Distributed Deutsch–Jozsa

In this section we are going to show how a variation of the Nonlocal Distributed Deutsch–Jozsa with m players, where m is even, can be solved with the same protocol presented in the previous section.

Consider the following novel problem[1]:

[1]We have not found any reference on the literature to this problem.

- Let $m_1, m_2, \ldots, m_m$ represent the m players that are arranged into $m/2$ pairs: $p_1 = (m_1, m_2),\ p_2 = (m_3, m_4), \ldots,\ p_{m/2} = (m_{m-1}, m_m)$.
- Player m_k receives as input an n-bit string i_k, where n is a power of two, and $1 \leq k \leq m$.
- Let $i_{p_l} = (i_{2j-1}, i_{2j})$, for $1 \leq j \leq m/2$ and $h = m/2$.
- Their inputs satisfy the "x-Pairs DJ promise":

  (C1) $i_{p_1} = \cdots = i_{p_h}$ or
  (C2) $i_{p_1} = \cdots = i_{p_{(h-x)}}$ and $d_H(i_{p_{(h-x+1)}}) = \cdots = d_H(i_{p_h}) = n/2$, $x \in \{1, m/2\}$.
- Their goal is to provide outputs $o_k \in \{0,1\}^{\log n}$, for all $1 \leq k \leq m$, without using any communication channel.
- Such that when:

  (C1) is true $o_1 = o_2, \ldots, o_{m-1} = o_m$,
  (C2) is true $o_1 = o_2, \ldots, o_{(2(h-x-1))} = o_{(2(h-x))}$ and
  $o_{(2(h-x+1))} \neq o_{(2(h-x+2))}, \ldots, o_{(2(h-1))} \neq o_{(2h)}$.

Quantum Protocol for the x-Pairs Nonlocal Distributed Deutsch–Jozsa:

Each pair $p_1, p_2, \ldots, p_{m/2}$ execute the Nonlocal Distributed Deutsch–Jozsa protocol from the previous section. Then, after $m/2$ executions the players will return the correct output.

One might ask if it is possible to adapt the protocol of Section 7.1 to a protocol where the m players play together. We are going to prove that a simple modification of the protocol for two players would not work for the x-Pairs Nonlocal Distributed Deutsch–Jozsa problem.

Before the game starts the players share the following entangled state $|\psi_0\rangle_{p_1 \ldots p_m}$:

$$|\psi_0\rangle_{p_1 \ldots p_m} = \frac{1}{\sqrt{n}} \sum_{j \in \{0,1\}^l} \underbrace{|j\rangle\,|-\rangle}_{m_1} \cdots \underbrace{|j\rangle\,|-\rangle}_{m_m},$$

where $l = \log n$.

Each player will go to separate locations and cannot communicate with each other. When the game start the m players receive their inputs i_k, $1 \leq k \leq m$, and apply the unitary matrix U_{i_k} (defined in (5.1)) to their part of the entangled state:

$$\begin{aligned} |\psi_1\rangle_{p_1 \ldots p_m} &= \frac{1}{\sqrt{n}} \sum_{j \in \{0,1\}^l} (-1)^{i_{1j}} |j\rangle\,|-\rangle \cdots (-1)^{i_{mj}} |j\rangle\,|-\rangle \\ &= \frac{1}{\sqrt{n}} \sum_{j \in \{0,1\}^l} (-1)^{i_{1j} \oplus \cdots \oplus i_{mj}} |j\rangle\,|-\rangle \cdots |j\rangle\,|-\rangle \end{aligned}$$

Next, each player will apply a Hadamard transform on their $l+1$ qubits. The entangled state becomes

$$|\psi_2\rangle_{p_1\ldots p_m} = \frac{1}{n^{(m+1)/2}} \sum_{o_1\in\{0,1\}^l} \cdots \sum_{o_m\in\{0,1\}^l} \Big(\sum_{j\in\{0,1\}^l} (-1)^{i_{1j}\oplus\cdots\oplus i_{mj}\oplus k\bullet(o_1\oplus\cdots\oplus o_m)} |o_1\rangle |1\rangle \cdots |o_m\rangle |1\rangle \Big).$$

For reference on how to get to this state see equation (7.1).

Now, let us calculate the probability that each one of the m players observes an l-bit string o_k, $1 \leq k \leq m$, when each player measures their first l qubits of the entangled state $|\psi_2\rangle_{p_1\ldots p_m}$:

$$\mathbb{P}(o_1, \cdots, o_m \mid i_1, \cdots, i_m) = \frac{1}{(n^{(m+1)/2})^2} \Big(\sum_{k\in\{0,1\}^l} (-1)^{i_{1k}\oplus\cdots\oplus i_{mk}\oplus k\bullet(o_1\oplus\cdots\oplus o_m)} \Big)^2. \quad (7.3)$$

Our task is now to analyze the probability of success of this protocol. If the players input satisfy C1 that is, $i_1 = \cdots = i_m$, each pair is expected to output the same l-bit string. For any input there are $(2^l)^{m/2} = n^{m/2}$ valid outputs. Using equation (7.3) above consider the following probability:

$$\mathbb{P}(o_1 = o_2, o_3 = o_4, \ldots, o_{m-1} = o_m \mid i_1 = i_2 = \cdots = i_m) = \frac{n^2}{(n^{(m+1)/2})^2} = \frac{1}{n^{m-1}}.$$

If $m = 2$ we are in fact in the original Nonlocal Distributed Deutsch–Jozsa setup and this probability agree with the analysis we did using equation (7.2). But, if there are more than 2 players ($m > 2$) as $1/n^{m-1}$ is strictly smaller than $1/n^{m/2}$ there is a chance that some pair(s) will output different l-bit strings when all the players got the same input. Concluding that this simple adaptation of the previous protocol will not work for the x-Pairs Nonlocal Distributed Deutsch–Jozsa. ■

7.3 Multi-party Nonlocal Hidden Matching

The nonlocal version of the hidden matching problem was introduced in [BCMdW10]. Later, Scarpa [Sca13] noted that it is possible to transform the two-party protocol into a multi-party protocol. We are going to formally state this problem and present a quantum protocol that solves this variation.

The multi-party nonlocal hidden matching problem is defined as:

- Let n be a power of two and $\mathcal{M}_n$ the set of all perfect matchings on the set $[n]$.
- Let $m_1, m_2, \ldots, m_m$ represent the m players.
- The first $m-1$ players receives an input $i_k \in \{0,1\}^n$, $1 \leq k \leq m-1$.
- The mth player receives $M \in \mathcal{M}_n$.
- The output of the first $m-1$ players are bit strings $o_k \in \{0,1\}^{\log n}$, $1 \leq k \leq m-1$.
- The output of the mth player is an $(a,b) \in M$ and $c \in \{0,1\}$.
- They win the game if and only if

$$\big((o_1 \oplus \cdots \oplus o_k) \bullet (a \oplus b)\big) \oplus c = i_{1a} \oplus i_{1b} \oplus \cdots \oplus i_{(m-1)a} \oplus i_{(m-1)b}.$$

Quantum Protocol for the Multi-party Nonlocal Hidden Matching Problem:

Initially the m players share the entangled state $|\psi_0\rangle_{p_1 \ldots p_m}$:

$$|\psi_0\rangle_{p_1 \ldots p_m} = \frac{1}{\sqrt{n}} \sum_{j \in \{0,1\}^l} \underbrace{|j\rangle\,|-\rangle}_{p_1} \cdots \underbrace{|j\rangle\,|-\rangle}_{p_{m-1}} \underbrace{|j\rangle}_{p_m},$$

where $l = \log n$.

Now, the first $m-1$ players apply their corresponding unitary matrix U_k (as defined in (5.1)) with their input i_k, $1 \leq k \leq m-1$. The state of the system is now

$$|\psi_1'\rangle_{p_1 \ldots p_m} = \frac{1}{\sqrt{n}} \sum_{j \in \{0,1\}^l} (-1)^{i_{1j}} |j\rangle\,|-\rangle \cdots (-1)^{i_{(m-1)j}} |j\rangle\,|-\rangle\,|j\rangle\,,$$

to improve readability we will ignore all the $|-\rangle$ qubits:

$$|\psi_1\rangle_{p_1 \ldots p_m} = \frac{1}{\sqrt{n}} \sum_{j \in \{0,1\}^l} (-1)^{i_{1j}} |j\rangle \cdots (-1)^{i_{(m-1)j}} |j\rangle\,|j\rangle\,.$$

The mth player performs a projective measurement with projectors $P_{ab} = |a\rangle\langle a| + |b\rangle\langle b|$, with $(a,b) \in M$ on his part of $|\psi_1\rangle_{p_1 \ldots p_m}$. After this measurement, the state collapses to

$$|\psi_2\rangle = \frac{1}{\sqrt{2}}\Big((-1)^{i_{1a}} |a\rangle \cdots (-1)^{i_{(m-1)a}} |a\rangle\,|a\rangle + (-1)^{i_{1b}} |b\rangle \cdots (-1)^{i_{(m-1)b}} |b\rangle\,|b\rangle\Big).$$

For the next step, each one of the m players applies a Hadamard transform on their qubit. The state of the system after the first player applied a Hadamard is

$$|\psi_3\rangle = \frac{1}{\sqrt{2}}\Big(\frac{1}{\sqrt{n}} \sum_{o_1 \in \{0,1\}^l} (-1)^{i_{1a} \oplus o_1 \bullet a} |o_1\rangle \cdots (-1)^{i_{(m-1)a}} |a\rangle\,|a\rangle$$

$$+(-1)^{i_{1b}\oplus o_1\bullet b}\,|o_1\rangle\cdots(-1)^{i_{(m-1)b}}\,|b\rangle\,|b\rangle\Big),$$

and after the mth Hadamard transform we get to

$$|\psi_f\rangle=\frac{1}{\sqrt{2}}\Big(\frac{1}{n^{m/2}}\sum_{o_1\in\{0,1\}^l}(-1)^{i_{1a}\oplus o_1\bullet a}\,|o_1\rangle\cdots\sum_{o_m\in\{0,1\}^l}(-1)^{o_m\bullet a}\,|o_m\rangle$$
$$+(-1)^{i_{1b}\oplus o_1\bullet b}\,|o_1\rangle\cdots(-1)^{i_{(m)b}}\,|o_m\rangle\Big).$$

Notice that in $|\psi_f\rangle$ any state $|o_1\rangle\cdots|o_m\rangle$ with non zero amplitude must satisfy the condition

$$i_{1a}\oplus o_1\bullet a\oplus\cdots\oplus i_{(m-1)a}\oplus o_{(m-1)}\bullet a\oplus o_m\bullet a\oplus i_{1b}\oplus o_1\bullet b\oplus\cdots$$
$$\oplus\, i_{(m-1)b}\oplus o_{(m-1)}\bullet b\oplus o_m\bullet b=0.$$

Rearranging the terms:

$$(o_1+\cdots+o_{m-1})\bullet(a+b)+o_m\bullet(a+b)=i_{1a}\oplus i_{1b}\oplus\cdots\oplus i_{(m-1)a}\oplus i_{(m-1)b}.$$

The condition above implies that all of the possible measurements outcomes will satisfy this equation that is equal to the winning condition of the multi-party non-local hidden matching problem. Finally, for the last step of the protocol, the players measure their part of the system. The first $m-1$ players output their $\log n$-bit strings $o_1,\ldots,o_{m-1}$ and the mth player output $(a,b)\in M$ and a bit $c=o_m\bullet(a+b)$.

Part III

Quantum Graph Parameters

Chapter 8

Quantum Chromatic Number of Hadamard Graphs

This chapter is organized as follows. In the first section, we begin by describing all the preliminary concepts required to understand the quantum chromatic number of Hadamard graphs. In the next section, we will summarize some important results from 1998 to 2005 that contributed to the development of this quantum graph parameter. Finally, in Section 8.3 we present a detailed explanation of the quantum protocol for the c-coloring game played with the Hadamard graph due to Avis, Hasegawa, Kikuchi and Sasaki.

8.1 Preliminary Concepts

A *simple graph* is a pair $G = (V, E)$ where V is a set whose elements are called vertices and E is a subset of $V^{(2)}$ whose elements are called edges. Two vertices $u, v \in V$ are adjacent if $\{u, v\} \in E$. We will represent this edge as uv or $u \sim v$. In this chapter we will only work with simple graphs.

Consider the same graph G, and let $S \subset V$ be any subset of vertices of G. Then the *induced subgraph* $G[S]$ is the graph whose vertex set is S and whose edge set consists of all of the edges in E that have both endpoints in S.

A c-coloring of a graph $G = (V, E)$ is an assignment of c colors to the vertices of G. A coloring c is *proper* if no two adjacent vertices are assigned the same color. The minimum c for which a graph G is c-colorable is called the *chromatic number* denoted by $\chi(G)$.

For the next section we are interested in the Hadamard graph. Let $N = 4k$ for any positive integer k. The *Hadamard graph* H_N is defined as the graph with vertex

set $V_N = \{0,1\}^N$ and edge set $E_N = \{uv \in V_N^{(2)} \mid d_H(u,v) = N/2\}$, where $d_H(u,v)$ is the Hamming distance of u and v.

To understand the quantum chromatic number we need first to define the following nonlocal game:

Definition 8.1.1 *c-coloring game*

- *Alice and Bob receive a graph $G = (V, E)$ from the referee.*
- *The players agree on a protocol to convince the referee that G is c-colorable.*
- *The referee wants to test their claim with a one-round game.*
- *The game starts and Alice and Bob are forbidden to communicate.*
- *The referee sends $a \in V$ to Alice and $b \in V$ to Bob such that $a = b$ or $ab \in E$.*
- *Alice sends the color c_a and Bob sends the color c_b to the referee.*
  - *They win the game if: $a = b$ and $c_a = c_b$ or ab and $c_a \neq c_b$*

Now that we have defined the c-coloring game we can introduce the following theorem:

Theorem 8.1.2
Let G be a graph. Then, the minimum c that classical players can choose to always win the c-coloring game with the graph G is $\chi(G)$.

Proof. Let us first consider any deterministic strategy classical players can use and later we will analyze any probabilistic strategy. Let

$$c^\star = \min\{c \mid \text{classical players can win the } c\text{-coloring game}\}.$$

We are going to prove that $c^\star = \chi(G)$ for any deterministic strategy:

1) $c^\star \geq \chi(G)$.

Any deterministic strategy would consist of two deterministic functions $f_a : V \to [c^\star]$ for Alice and $f_b : V \to [c^\star]$ for Bob. To satisfy the first winning condition the players must always output the same color when asked the same vertex implying that $f_a = f_b$. To satisfy the second winning condition f_a assigns different colors to adjacent vertices and therefore induces a proper coloring of the graph.

2) $c^\star \leq \chi(G)$.

By the same argument above to satisfy the first winning condition Alice and Bob's strategy consists of the same deterministic function $f_a : V \to [c^\star]$. For

the second winning condition the minimum number of colors for which the players can proper color the graph is $\chi(G)$. The function f_a maps each vertex of V to a $\chi(G)$-coloring of G.

To consider any probabilistic strategy we have to consider additional resources. Before the game starts the players can share a random n-bit string. After receiving their inputs they can use this shared n-bit string and also generate private random bit strings and play accordingly. In this scenario a probabilistic strategy is defined by a function $f_a : \{0,1\}^n \times V \to [c^\star]$ such that $f_a(r,v)$ is the color Alice responds when she receives $v \in V$ from the referee given that the shared random bit string took on value r. Let f_b be a function defined by Bob in the same manner. Now, we are going to prove that $c^\star = \chi(G)$ for any probabilistic strategy:

1) $c^\star \geq \chi(G)$.

To satisfy the first winning condition for all $r \in \{0,1\}^n$ and for all $v \in V$ we have $f_a(r,v) = f_b(r,v)$. For the second winning condition let us define $g_r : V \to [c^\star]$ for all $r \in \{0,1\}^n$ by letting $g_r(v) = f_a(r,v)$. To satisfy this condition for all r the function g_r assigns different colors to adjacent vertices and therefore induces a proper coloring of the graph.

2) $c^\star \leq \chi(G)$.

Again to satisfy the first winning condition Alice and Bob have to play with the same function f_a. Now, for the second winning condition let us define $g_r : V \to [\chi(g)]$ for all $r \in \{0,1\}^n$ by letting $g_r(v) = f_a(r,v)$. For all r the function g_r maps each vertex of V to a $\chi(G)$-coloring of G.

Hence, even with additional resources, the best probabilistic strategy would still need $\chi(G)$ colors to win the c-coloring game. ■

If the players are allowed to share an entangled state before the game starts and realize quantum computation and measurements on their part of the state there are graphs for which Alice and Bob can win the c-coloring game with probability 1 for $c < \chi(G)$. We call the smallest c such that Alice and Bob can win the c-coloring game with probability one the *quantum chromatic number* denoted by $\chi_q(G)$. In this scenario, we are also going to say that the graph G is *quantum c-colorable*.

Whenever a graph G has $\chi_q(G) < \chi(G)$ we are going to say that the c-coloring game with G is a *nonlocal game* or a *pseudo-telepathy game*. The first graph that exhibited this difference on the quantum and classical chromatic number was a special case of the Hadamard graph that we are going to introduce in the next section.

8.2 Overview

In this section we will summarize some key results from 1998 to 2005 on the c-coloring game. Buhrman, Cleve and Wigderson in 1998 [BCW98] discovered the first large gaps between quantum and classical communication complexity. They introduced two protocols; the distributed version of Deutsch–Jozsa algorithm (explained in Section 5.1) and also a protocol based on Grover's algorithm.

In 1999, Brassard, Cleve and Tapp [BCT99] were interested in the amount of communication necessary for classical systems with shared randomness to simulate systems with quantum entanglement. Inspired by Buhrman, Cleve and Wigderson result they proved that for the Nonlocal Distributed Deutsch–Jozsa problem (Section 7.1) there is no way that classical players can always win this game without communication.

Implicitly they proved that quantum players can win the c-coloring game with probability 1 with the Hadamard graph H_N and $c = N$, where N is a power of two by using the Nonlocal Distributed Deutsch–Jozsa protocol. For the classical scenario they used a result by Frankl and Rödl [FR87] that for all large N, the chromatic number of the Hadamard graph grows exponentially in N. Combining these two results they implicitly proved that the c-coloring game with H_N is asymptotically a pseudo-telepathy game.

Galliard and Wolf in 2002 [GW02] formally made the connection between the Nonlocal Distributed Deutsch–Jozsa problem and the c-coloring game. A year later Galliard, Tapp and Wolf [GTW03] studied the same restricted case of the Hadamard graph H_N for $N = 2^n$. They proved using a combinatorial argument that for $n = 4$ the c-coloring game with H_{16} is a pseudo-telepathy game. This graph with 65536 vertices is quantum 16-colorable but $\chi(G) > 16$. They also noted that for any n smaller than four, in this restricted case where $N = 2^n$, would not provide a difference between the quantum and classical chromatic number.

The last result of this summary is due to Avis, Hasegawa, Kikuchi and Sasaki [AHKS06] from 2005. The authors discovered a quantum protocol to win the c-coloring game on all Hadamard graphs, removing the restriction that $N = 2^n$ from Galliard, Tapp and Wolf. Next, using a result from Godsil and Newman [GN08] that a Hadamard graph H_N has a chromatic number strictly larger than N whenever $N = 4k > 8$ they obtained the following result:

Theorem 8.2.1 *Adapted from* [AHKS06]
For all $k \geq 3$, $\chi_q(H_{4k}) \leq 4k$ while $\chi(H_{4k}) > 4k$.

We are going to present the prove of this theorem in the next subsection.

After they proved the theorem above they noted that the smallest Hadamard graph H_N such that the c-coloring game is a pseudo-telepathy game with $c = N$ is the H_{12}. Any of its induced subgraphs with 1609 vertices also have this property.

8.3 c-coloring Game Protocol for Hadamard Graphs

In this section we will prove the upper bound on the quantum chromatic number from Theorem 8.2.1 by introducing the protocol due to Avis, Hasegawa, Kikuchi and Sasaki with added details for a self-contained proof. We will show that with this protocol Alice and Bob can win the c-coloring game with certainty for any Hadamard graph H_N using N colors.

For this protocol we are going to use the quantum Fourier transform[1] (QFT). The QFT acts on a quantum state $|x\rangle = \sum_{i=0}^{N-1} x_i |i\rangle$ and maps it to a quantum state $\sum_{i=0}^{N-1} y_i |i\rangle$ according to the formula:

$$y_k = \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} x_j \omega_N^{jk}, \ k = 0, \ldots, N-1,$$

where $\omega_N = \exp(2\pi i/N)$. Note that N is not necessarily a power of two.

We can check that the QFT applied to a qubit $|\psi\rangle = \alpha |0\rangle + \beta |1\rangle$ is equal to the Hadamard transform:

$$y_0 = \frac{1}{\sqrt{2}}(\alpha e^{(2\pi i \frac{(0\cdot 0)}{2})} + \beta e^{(2\pi i \frac{(1\cdot 0)}{2})}) = \frac{1}{\sqrt{2}}(\alpha + \beta)$$
$$y_1 = \frac{1}{\sqrt{2}}(\alpha e^{(2\pi i \frac{(0\cdot 1)}{2})} + \beta e^{(2\pi i \frac{(1\cdot 1)}{2})}) = \frac{1}{\sqrt{2}}(\alpha - \beta).$$

Hence,

$$\mathrm{QFT}\,|\psi\rangle = \frac{1}{\sqrt{2}}(\alpha + \beta)\,|0\rangle + \frac{1}{\sqrt{2}}(\alpha - \beta)\,|1\rangle\,,$$

that is equal to $\mathrm{H}\,|\psi\rangle$.

[1] Quantum analog of the classical discrete Fourier transform applied to the vector of amplitudes of a quantum state.

If we consider the QFT or the QFT^{-1} (inverse of the quantum Fourier transform) applied to a basis state $|b\rangle \in \mathbb{C}^N$, instead of a general state, we get the following mapping:

$$\mathrm{QFT}\,|b\rangle = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} \omega_N^{bk}\,|k\rangle\,, \qquad \mathrm{QFT}^{-1}\,|b\rangle = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} \omega_N^{-bk}\,|k\rangle\,.$$

For instance, applying the QFT to the basis state $|01\rangle$ written as $|1\rangle$ gives:

$$\begin{aligned}\mathrm{QFT}\,|1\rangle &= \frac{1}{2}(\omega_4^{1\cdot 0}\,|0\rangle + \omega_4^{1\cdot 1}\,|1\rangle + \omega_4^{1\cdot 2}\,|2\rangle + \omega_4^{1\cdot 3}\,|3\rangle) \\ &= \frac{1}{2}(|0\rangle + e^{\frac{\pi i}{2}}\,|1\rangle + e^{\pi i}\,|2\rangle + e^{\frac{3\pi i}{2}}\,|3\rangle) \\ &= \frac{1}{2}(|0\rangle + i\,|1\rangle - |2\rangle - i\,|3\rangle).\end{aligned}$$

Now, that we defined this unitary transformation we can introduce the protocol:
Quantum Protocol for the c-coloring Game for Hadamard Graphs

Before the game starts, Alice and Bob get together and receive from the referee a Hadamard graph H_N. Their first step is to prepare the entangled state

$$|\psi_0\rangle_{AB} = \frac{1}{\sqrt{N}} \sum_{i=0}^{N-1} |i\rangle_A\,|-\rangle_A\,|i\rangle_B\,|-\rangle_B\,.$$

To improve readability we will omit both of the $|-\rangle$ states as they will only be used once as auxiliary qubits.

Alice and Bob move apart and are now forbidden to communicate. The game starts and the referee sends the vertex a to Alice and the vertex b to Bob. As mentioned before in the definition of the Hadamard graph the vertices are expressed as N-bit strings.

If N is a power of two each one of the players apply the unitary operator U_x defined in (5.1) using their inputs a and b on their part of $|\psi_0\rangle_{AB}$. If N is not a power of two Alice will apply the unitary operator U'_a defined as:

$$U'_a\,|j\rangle\,|q\rangle = \begin{cases} |j\rangle\,|q \oplus a_j\rangle\,, & j = 0, 1, \ldots, N-1 \text{ and } q \in \{0,1\}. \\ |j\rangle\,|q\rangle\,, & j = N, \ldots, 2^{\lceil \log_2 N \rceil} - 1 \text{ and } q \in \{0,1\}. \end{cases}$$

Bob will act similarly using his input b. As we had mentioned before both of the players have access to an auxiliary qubit in the $|-\rangle$ state. Considering the phase

kick-back trick (see (3.2) for reference) no matter the value of N after this step the resulting state is:

$$|\psi_1\rangle_{AB} = (U_a \otimes U_b)\,|\psi_0\rangle_{AB} = \frac{1}{\sqrt{N}} \sum_{i=0}^{N-1} (-1)^{a_i}\,|i\rangle_A\,(-1)^{b_i}\,|i\rangle_B = \frac{1}{\sqrt{N}} \sum_{i=0}^{N-1} (-1)^{a_i \oplus b_i}\,|i\rangle_A\,|i\rangle_B\,.$$

For the next step Alice applies the QFT and Bob the QFT^{-1} on their part of $|\psi_1\rangle_{AB}$:

$$|\psi_2\rangle_{AB} = (\mathrm{QFT} \otimes \mathrm{QFT}^{-1})\,|\psi_1\rangle_{AB} = \left(\frac{1}{\sqrt{N}}\right)^3 \sum_{c_a=0}^{N-1} \sum_{c_b=0}^{N-1} \sum_{i=0}^{N-1} \omega^{i(c_a - c_b)} (-1)^{a_i \oplus b_i}\,|c_a\rangle_A\,|c_b\rangle_B\,.$$

The last step of the protocol is the measurement operation. Alice and Bob measure their part of $|\psi_2\rangle_{AB}$ in the computational basis (defined in 2.6) and obtain the same string c with probability given by:

$$\mathbb{P}(c, c \mid a, b) = \underbrace{N}_{\text{\# possibilities}} \left(\left(\frac{1}{\sqrt{N}}\right)^3 \sum_{i=0}^{N-1} \omega^{i(c-c)} (-1)^{a_i \oplus b_i} \right)^2 = \frac{1}{N^2} \left(\sum_{i=0}^{N-1} (-1)^{a_i \oplus b_i} \right)^2. \tag{8.1}$$

If $a = b$, $a_i \oplus b_i = 0$ for $i = 0, \ldots, N-1$. Evaluating (8.1) in this case gives

$$\mathbb{P}(c, c \mid a = b) = \frac{1}{N^2} \left(\sum_{i=0}^{N-1} (-1)^0 \right)^2 = \frac{1}{N^2} (N)^2 = 1.$$

When Alice and Bob get the same vertex as input they will always send the same color (expressed by the string c) to the referee.

If $a \neq b$, we know that $ab \in E$ and by definition $d_H(a, b) = N/2$. Evaluating (8.1) in this case:

$$\mathbb{P}(c, c \mid a \neq b) = \frac{1}{N^2} \left(\frac{N}{2} - \frac{N}{2} \right)^2 = 0.$$

Hence, when Alice and Bob get two adjacent vertices as input they will never send the same color to the referee.

These analyses show that this protocol allows Alice and Bob to win the c-coloring game with certainty for any Hadamard graph H_N using N colors. ■

Chapter 9

General Quantum Chromatic Number

In 2006 Cameron, Newman, Montanaro, Severini and Winter [CMN$^+$06] investigated the notion of the quantum chromatic number of any graph. They also proved general facts about this quantum graph parameter. In this chapter, we will study some of their results and give a self-contained explanation of the quantum chromatic number.

To understand this concept we start by giving the required background from quantum information theory that is not usually covered in quantum computing materials. In Section 9.2 we will give an intuition of what two quantum strategies for any graph would look like.

In Section 9.3 we will introduce some important concepts from [CMN$^+$06]. From this point until Section 9.6 our goal will be to explicitly formulate a quantum strategy that enables Alice and Bob to win the c-coloring game on a special graph.

The final topic of this chapter is a short review of the smallest known graph with quantum chromatic number smaller than its classical chromatic number. This is a 2016 result due to Mančinska and Roberson [MR18].

9.1 Other Types of Measurements

In this section, inspired by de Wolf lecture notes [dW21] and also [NC02], we are going to explain the projective measurement operation, describe what is an observable and introduce the POVM measurement.

A projective measurement can be described by an *observable* M, a Hermitian operator on the state of the system being observed. The observable has spectral

decomposition $M = \sum_{j=1}^{m} \lambda_j P_j$, where P_j is the projector onto the eigenspace[1] of M with eigenvalue λ_j. Using this observable the possible outcomes of the measurement correspond to the eigenvalues λ_j of M. To illustrate this new type of measurement consider the following observable:

$$M = \underbrace{1}_{\lambda_1} \underbrace{|+\rangle \langle +|}_{P_1} \underbrace{-1}_{\lambda_2} \underbrace{|-\rangle \langle -|}_{P_2}.$$

Measuring the state $|\phi\rangle = |0\rangle$ with the observable M gives the outcome "2" (label of the projector associated with the eigenvalue -1) with probability $\langle\phi| P_2 |\phi\rangle = 1/2$ and gives the outcome "1" with probability $1/2$. Suppose we got the outcome "2". Then, the state of the system collapses to the new state $|-\rangle$. This is called an X measurement as we are measuring in the X basis $\{|+\rangle, |-\rangle\}$.

Another common way to describe a projective measurement with m possible outcomes is by giving a list $P_1, \ldots, P_m$ of projectors ($P_j = P_j^2,\ 1 \leq j \leq m$) that act of the state of the system being measured and that $\sum_{j=1}^{m} P_j = \mathrm{I}$. With these conditions these projectors are pairwise orthogonal meaning that $P_j P_i = 0$, if $j \neq i$. The projector P_j projects on some subspace V_j of the total space V, and every state $|\psi\rangle \in V$ can be decomposed in a unique way as $|\psi\rangle = \sum_{j=1}^{m} |\psi_j\rangle$, with $|\psi_j\rangle = P_j |\psi\rangle \in V_j$. Because the projectors are orthogonal the subspaces V_j are orthogonal as are the states $|\psi_j\rangle$. When we apply this measurement to the state $|\psi\rangle$ we get the outcome j (label of the projector) with probability $\|P_j |\psi\rangle\|^2 = \mathrm{Tr}(P_j |\psi\rangle \langle\psi|) = \langle\psi| P_j |\psi\rangle$. The measured state collapses to the new state $P_j |\psi\rangle \,/\, \|P_j |\psi\rangle\|$.

To familiarize the reader with this type of measurement consider the 2-qubit state $|\psi\rangle = (|00\rangle + |01\rangle + |11\rangle)/\sqrt{3}$. Let $P_1 = |00\rangle \langle 00|$ and $P_2 = |01\rangle \langle 01| + |10\rangle \langle 10| + |11\rangle \langle 11|$ and carry out a projective measurement on $|\psi\rangle$ with this two projectors. Since $P_1 |\psi\rangle = |00\rangle / \sqrt{3}$ and $P_2 |\psi\rangle = (|01\rangle + |11\rangle)/\sqrt{3}$. We will observe the outcome "1" with probability $\langle\psi| P_1 |\psi\rangle = 1/3$ collapsing the state of the system to $|00\rangle$ and we will observe the outcome "2" with probability $2/3$ collapsing the state of the system to $(|01\rangle + |11\rangle)/\sqrt{2}$.

The most general type of measurement is the *positive operator-valued measure* (POVM). This measurement is commonly used when we only care about the final probability distribution of the possibles outcomes, not about the post-measurement state of the system. A *POVM* is specified by m positive semi-definite[2] matrices

[1] Let $T : V \to V$ be an linear transformation. Given an eigenvalue λ, consider $E = \{|v\rangle : T |v\rangle = \lambda |v\rangle\}$ which is the set of all the eigenvectors associated with λ. E is called the *eigenspace* of T associated with λ.

[2] A Hermitian matrix M is positive semi-definite if the scalar $\langle z| M |z\rangle$ is nonnegative for every nonzero complex $|z\rangle$.

$E_1, \ldots, E_m$ that sum to identity. Each one of these matrices are usually called the *elements* of the POVM. When measuring a state $|\psi\rangle$ with this POVM the probability of observing the outcome j, $1 \leq j \leq m$, is given by $\langle\psi| E_i |\psi\rangle = \mathrm{Tr}(E_i |\psi\rangle \langle\psi|)$. A projective measurement is the special case of a POVM where the elements E_j are projectors.

Peres/Wootters Game

To illustrate a possible use of a POVM we will introduce a quantum detection problem due to Holevo [Hol73b] and Peres/Wootters [Per97]. Suppose we receive a qubit $|\psi_j\rangle$ which is guaranteed to be in one of the following states:

$$|\psi_0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \text{ or } |\psi_1\rangle = \frac{1}{\sqrt{2}}(|0\rangle + \omega |1\rangle) \text{ or } |\psi_2\rangle = \frac{1}{\sqrt{2}}(|0\rangle + \omega^2 |1\rangle),$$

where $\omega = \exp(2i\pi/3)$.

Our task is to answer in which state $|\psi_j\rangle$ was not in. Our valid answers for each one of the possible states we got as input are:

- If $j = 0$ answer 1 or 2.
- If $j = 1$ answer 0 or 2.
- If $j = 2$ answer 0 or 1.

To achieve this goal we are going to build a POVM with three elements. First we need to define the following vectors: $|E'_0\rangle$ orthogonal to $|\psi_0\rangle$, $|E'_1\rangle$ orthogonal to $|\psi_1\rangle$ and $|E'_2\rangle$ orthogonal to $|\psi_2\rangle$:

$$|E'_0\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle), \quad |E'_1\rangle = \frac{1}{\sqrt{2}}(|0\rangle - \omega |1\rangle), \quad |E'_2\rangle = \frac{1}{\sqrt{2}}(|0\rangle - \omega^2 |1\rangle).$$

Now, we can use these vectors to define our POVM elements:

$$E_0 = \frac{2}{3} |E'_0\rangle \langle E'_0|, \quad E_1 = \frac{2}{3} |E'_1\rangle \langle E'_1|, \quad E_2 = \frac{2}{3} |E'_2\rangle \langle E'_2|,$$

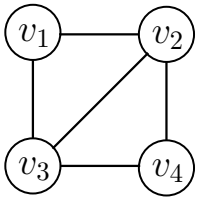
that sum to I as required.

After verifying that $\mathrm{Tr}(E_j |\psi_j\rangle \langle\psi_j|) = 0$ for $i = 0, 1, 2$ one can conclude that this POVM never output the wrong answer. In other words, this POVM always identifies which state we did not get as input.

9.2 Intuition of a Quantum Strategy for the c-coloring game

In Section 8.3 we presented a protocol that wins the c-coloring game on all Hadamard graphs. Our goal for this section is not to present another graph or family of graphs that provide pseudo-telepathy on the c-coloring game but to give an intuition on what two quantum strategies for any graph would look like.

Consider the graph $G = (V, E)$ below:



One way Alice and Bob could win the c-coloring game on the graph G is to play according to the following strategy: Before the game starts they share the entangled state $|\psi\rangle_{AB} = (|00\rangle + |11\rangle + |22\rangle)/\sqrt{3}$ and define four unitary matrices:

$$U_{v_1} = U_{v_4} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \qquad U_{v_2} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}, \qquad U_{v_3} = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}.$$

After moving to separate locations the referee sends Alice a vertex $a \in V$ and Bob a vertex $b \in V$. Alice applies the unitary transformation U_a on her part of $|\psi\rangle_{AB}$, then she measures the resulting state in the computational basis (defined in 2.6) and sends the output to the referee. Bob proceeds in the same manner.

Let us verify that this strategy always wins the c-coloring game on the graph G. Suppose Alice input is the vertex v_3 and Bob got v_2 as input. After Alice applies U_{v_3} on her part of $|\psi\rangle_{AB}$, then the entangled state becomes $|\psi'\rangle_{AB} = (|1\rangle|0\rangle + |2\rangle|1\rangle + |0\rangle|2\rangle)/\sqrt{3}$. Now, let us express with a table: the possible results of Alice's measurement, the state $|\psi'\rangle_{AB}$ collapses given the result of this measurement, and the final state after Bob applies U_{v_2} on his part of the system and Bob's output.

Alice Output	$\|\psi'\rangle_{AB}$ Collapse to $\|\psi''\rangle_{AB}$	After Bob Applies U_{v_2} on $\|\psi''\rangle_{AB}$	Bob Output
0	$\|0\rangle_A \|2\rangle_B$	$\|0\rangle_A \|1\rangle_B$	1
1	$\|1\rangle_A \|0\rangle_B$	$\|1\rangle_A \|2\rangle_B$	2
2	$\|2\rangle_A \|1\rangle_B$	$\|1\rangle_A \|0\rangle_B$	0

After they conclude this protocol the referee received one of the three possible proper coloring of the vertices v_3 and v_2: $2,1$ or $0,2$ or $1,0$. It is straightforward to verify that this strategy always wins the c-coloring game on this graph.

Another way we could formulate a quantum strategy for this game would be to replace the unitary matrices and measurement in the computational basis by a projective measurement. Using the same graph and entangled state $|\psi\rangle_{AB}$ consider these four observables with their respective projectors:

$$\begin{aligned} M_1 = \ M_4 &= -1\underbrace{|0\rangle\langle 0|}_{P_{-1}^{(1)}} + 0\underbrace{|1\rangle\langle 1|}_{P_{0}^{(1)}} + 1\underbrace{|2\rangle\langle 2|}_{P_{+1}^{(1)}}, \\ M_2 &= +0\underbrace{|0\rangle\langle 0|}_{P_{0}^{(2)}} + 1\underbrace{|1\rangle\langle 1|}_{P_{+1}^{(2)}} \underbrace{-1|2\rangle\langle 2|}_{P_{-1}^{(2)}}, \\ M_3 &= +1\underbrace{|0\rangle\langle 0|}_{P_{+1}^{(3)}} - 1\underbrace{|1\rangle\langle 1|}_{P_{-1}^{(3)}} + 0\underbrace{|2\rangle\langle 2|}_{P_{0}^{(3)}}. \end{aligned}$$

With this strategy, Alice and Bob make a projective measurement using one of the four observables according to the vertex they got as input. Suppose Alice got the vertex v_3 as input. Let us express the result of Alice measuring her part of $|\psi\rangle_{AB}$ with the observable M_3 with the table:

Output	$\|\psi\rangle_{AB}$ Collapse to $\|\psi'\rangle_{AB}$
$+1$	$\|00\rangle$
-1	$\|11\rangle$
0	$\|22\rangle$

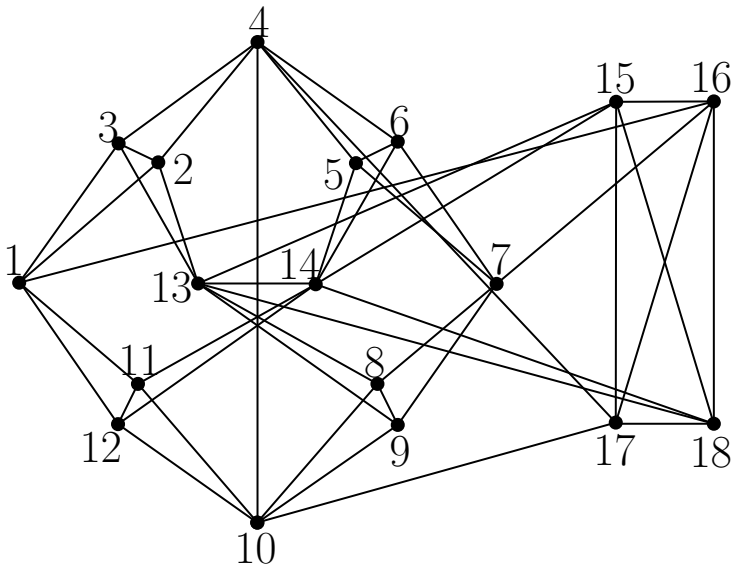
It is easy to check that if Bob got the vertex v_3 as input he would output the same color Alice did. Considering he got the vertex v_1 (or v_4) as input. Measuring the state $|\psi'\rangle_{AB}$ with the observable M_1 would result in the following:

Given Alice Output was	$\|\psi\rangle_{AB}$ Collapse to $\|\psi'\rangle_{AB}$	Bob Output After Measuring $\|\psi'\rangle_{AB}$ with M_1
$+1$	$\|00\rangle$	-1
-1	$\|11\rangle$	0
0	$\|22\rangle$	$+1$

One could check that the above strategy work by testing all the combination of inputs. In the next section, we are going to define two consistency conditions inspired by this strategy with projective measurements that make Alice and Bob win the c-coloring game with certainty.

9.3 G_{18}, General Strategy and Rank-r Version

In 2006, Cameron, Newman, Montanaro, Severini and Winter [CMN$^+$06] proved various general facts about the quantum chromatic number. We are particularly interested in their results involving the G_{18}, a graph with 18 vertices and 44 edges with chromatic number 5 and quantum chromatic number 4.



G_{18}

One of the goals of this chapter will be to use their results to explicitly formulate a quantum strategy for the G_{18} in terms of the unitary matrices Alice and Bob apply to each vertex. To achieve this goal we need first to understand how they defined a general quantum strategy for the c-coloring game and also the rank-r version of the quantum chromatic number.

Definition 9.3.1 [CMN$^+$06]
The most *general strategy* for Alice and Bob to win the c-coloring with probability 1 with c colors for a graph $G = (V, E)$ consists of an entangled state $|\psi\rangle_{AB} \in \mathbb{C}^d \otimes \mathbb{C}^d$ and two families of POVMs. For all $v \in V$, Alice has $\{E_\alpha^v\}_{\alpha=1,\ldots,c}$ and Bob has $\{F_\beta^v\}_{\beta=1,\ldots,c}$. This entangled state and POVMs are set by the players before they get their inputs. When the game starts Alice applies a POVM measurement according to the vertex she got as input and outputs α, the result of this measurement (one of the c possible colors). Bob acts similarly and outputs β. The fact that they win with probability 1 is expressed by the *consistency condition*:

- The players will never output different colors for the same vertex:

$$\forall v \in V, \forall \alpha \neq \beta, \langle\psi| E_\alpha^v \otimes F_\beta^v |\psi\rangle = 0.$$

- The player will never output the same color for any adjacent vertices:

$$\forall vw \in E, \forall \gamma, \langle\psi| E_\gamma^v \otimes F_\gamma^w |\psi\rangle = 0.$$

It is important to emphasize that the dimension d of the entangled state and the rank[3] of the POVMs have no relationship to c. Alice and Bob can use any entangled state and families of POVMs they want. The only quantity we care about is c. In this general strategy c is the number of possible measurements outcomes.

As we mentioned before we call the smallest c such that Alice and Bob can win the c-coloring game the quantum chromatic number. The authors noted that an equivalent definition of the quantum chromatic number is the smallest possible c for which Alice and Bob while playing accordingly to a general strategy, can convince the referee that the consistency condition always holds.

In Section II of [CMN$^+$06] the authors prove that there is simpler way to formulate the general strategy with just one family of POVMs and one consistency condition. Considering the general strategy 9.3.1 we highlight the differences between these two strategies.

[3]The rank of a matrix M is the dimension of the vector space spanned by its columns. This corresponds to the maximal number of linearly independent columns of M.

Definition 9.3.2 [CMN$^+$06]
For all $v \in V$ Alice and Bob have $\{E_\alpha^v\}_{\alpha=1,\ldots,c}$. Alice's strategy is the same but Bob measures his part of the entangled state according to the vertex he got as input with $\{F_\alpha^v\}_{\alpha=1,\ldots,c} = \{\overline{E_\alpha^v}\}_{\alpha=1,\ldots,c}$, for all $v \in V$. The consistency condition can be phrased entirely in terms of Alice's POVMs:

$$\forall vw \in E \text{ and } \forall \alpha \;\; E_\alpha^v E_\alpha^w = 0. \tag{9.1}$$

The notation $\overline{M}$ denotes the complex conjugate of the matrix M.

An interesting observation of this quantum graph parameter is that if we restrict the rank of the POVM elements we can define a restricted case of the quantum chromatic number.

Definition 9.3.3 [CMN$^+$06]
The *rank-r version of the quantum chromatic number* $\chi_q^{(r)}(G)$ is the minimum c such that Alice and Bob can win the c-coloring game for G with an entangled state $|\psi\rangle_{AB} \in \mathbb{C}^{rc} \otimes \mathbb{C}^{rc}$ with the restriction that all the elements of the POVMs have rank r.

By the definition above the authors conclude that $\chi_q^{(r)}(G) \leq \chi_q^{(s)}(G)$ whenever $r \geq s$, and that $\chi_q(G) = \inf_r \chi_q^{(r)}(G)$. In particular, the rank-1 quantum chromatic number is an upper bound of the quantum chromatic number.

9.4 Quantum Chromatic Number as a Graph Parameter

In Section III of [CMN$^+$06] the authors study some properties of the quantum chromatic number as a graph parameter. We are particularly interested in two propositions from this section. We will first introduce the necessary concepts from graph theory and then we will present these two properties.

A *complete graph* is a graph in which every pair of distinct vertices is connected by a unique edge. A complete graph on n vertices is denoted by K_n. A graph $H = (V_h, E_h)$ is a *subgraph* of $G = (V, E)$ if $V_h \subseteq V$ and $E_h \subseteq E$. Using these two concepts we will define a *clique* of a graph G as a complete subgraph of G. The *clique number* of G, denoted by $\omega(G)$ is the size of the largest clique of G. Finally, the last concept we need is graph homomorphism. A homomorphism from a graph G to a graph H is a mapping $\phi : V(G) \to V(H)$ such that

$$uv \in E(G) \implies \phi(u)\phi(v) \in E(H).$$

We write $G \to H$ if there is a homomorphism of G to H.

We are now in a position to prove the following propositions:

Proposition 9.4.1 [CMN$^+$06]
If $G \to H$, then $\chi_q^{(r)}(G) \leq \chi_q^{(r)}(H)$ for all r and hence $\chi_q(G) \leq \chi_q(H)$.

Proof. Consider the graph H. By Definition 9.3.2 of a general strategy, Alice and Bob can share an entangled state $|\psi\rangle_{AB} \in \mathbb{C}^d \otimes \mathbb{C}^d$ and for all $v \in V(H)$ there is a family of POVMs $\{E_\alpha^v\}_{\alpha=1,\ldots,c}$ of rank r such that the consistency condition (9.1) always holds. Let ϕ be a homomorphism from G to H. Using the mapping ϕ and the general strategy for the graph H Alice and Bob can define a general strategy for the graph G in the following manner: They share the same entangled state $|\psi\rangle_{AB}$ and for all $v \in V(G)$ they use the POVM $\{E_\alpha^{\phi(v)}\}_{\alpha=1,\ldots,c}$. It is straightforward to verify that this construction satisfy the consistency condition 9.1. ■

In [CMN$^+$06] the authors state two useful properties that we are going to use to prove the next proposition. First, for any $r \geq 1$, if G has no edges then $\chi_q^{(r)}(G) = \chi_q(G) = 1$. Second, for any $r \geq 1$, if $G = K_n$ then $\chi_q^{(r)}(G) = \chi_q(G) = n$.

Proposition 9.4.2 [CMN$^+$06]
$\omega(G) \leq \chi_q(G) \leq \chi(G)$.

Proof. For the leftmost inequality we note that any graph G contains $K_{\omega(G)}$ as a subgraph hence there is a homomorphism from $K_{\omega(G)}$ to G. By Proposition 9.4.1 and the observation from the last paragraph we have that $\chi_q(K_{\omega(g)}) = \omega(G) \leq \chi_q(G)$. For the rightmost inequality we note that there is a homomorphism from G to $K_{\chi(G)}$, by mapping each vertex of G to the vertex of $K_{\chi(G)}$ corresponding to its color. The results follows by Proposition 9.4.1 and the observation above. ■

9.5 A Strategy for the Rank-1 Quantum Chromatic Number

The restricted case of the quantum chromatic number where all POVMs elements have rank 1 can be modeled in two ways. In the first way Alice and Bob share a *c-dimension maximally entangled state* $|\psi\rangle_{AB} \in \mathbb{C}^c \otimes \mathbb{C}^c$, with $c \geq 2$, defined as:

$$|\psi\rangle_{AB} = \frac{1}{\sqrt{c}} \sum_{j=1}^{c} |j\rangle_A |j\rangle_B , \tag{9.2}$$

and apply the POVMs that correspond to the vertex they got as input that satisfy the consistency condition (9.1). As the elements of the POVMs have rank 1 they are in fact projective measurements.

In this chapter we are particularly interested in another strategy. In Section II of [CMN$^+$06] the authors prove that for this special rank-1 case we can formulate the following strategy in terms of unitary matrices.

Definition 9.5.1
Alice and Bob share a c-dimension maximally entangled state $|\psi\rangle_{AB}$. For each vertex $v \in V$ they define a $c \times c$ unitary matrix U_v. The referee sends the vertex v to Alice and w to Bob. Alice apply the unitary matrix $U_v^\dagger$ on her part of the entangled state and Bob apply the unitary matrix $U_w^\top$ on his part of the entangled state. By definition the system evolves according to the application of $(U_v^\dagger \otimes U_w^\top)$ to $|\psi\rangle_{AB}$. Next, they measure $(U_v^\dagger \otimes U_w^\top)|\psi\rangle_{AB}$ in the computational basis (defined in 2.6). Alice's output is a color $\alpha \in [c]$ and Bob's is a color $\beta \in [c]$. The consistency condition for this strategy is:

$$\forall vw \in E,\ U_v^\dagger U_w \text{ has only zeroes on the diagonal.} \tag{9.3}$$

Let us prove that 9.5.1 provides a valid strategy for the c-coloring game.
Proof:

1) Alice and Bob get the same vertex v as input.

The following facts will be useful before we show that in this case they answer the same color to the referee. Let $c \geq 2$ and

$$U_v = \begin{bmatrix} m_{11} & \cdots & m_{1c} \\ \vdots & \ddots & \vdots \\ m_{c1} & \cdots & m_{cc} \end{bmatrix}.$$

By definition $U_v^\dagger U_v = \mathrm{I}$, so we can conclude that

$$\sum_{p=1}^{c} \overline{m}_{pq} m_{pq} = 1, \text{ for } q = 1, \ldots, c, \tag{9.4}$$

and

$$\sum_{p=1}^{c} \overline{m}_{pq} m_{pr} = 0, \text{ for } q, r = 1, \ldots, c \text{ and } q \neq r. \tag{9.5}$$

Now, we will study the evolution of the system:

$$(U_v^\dagger \otimes U_v^\top)|\psi\rangle_{AB} = (U_v^\dagger \otimes U_v^\top)\frac{1}{\sqrt{c}}(|1\rangle|1\rangle + \cdots + |c\rangle|c\rangle)$$

$$= \frac{1}{\sqrt{c}} \left(\begin{pmatrix} \overline{m}_{11} \\ \vdots \\ \overline{m}_{1c} \end{pmatrix} \otimes \begin{pmatrix} m_{11} \\ \vdots \\ m_{1c} \end{pmatrix} + \cdots + \begin{pmatrix} \overline{m}_{c1} \\ \vdots \\ \overline{m}_{cc} \end{pmatrix} \otimes \begin{pmatrix} m_{c1} \\ \vdots \\ m_{cc} \end{pmatrix} \right)$$
$$= \frac{1}{\sqrt{c}} \begin{pmatrix} \overline{m}_{11}m_{11} + \cdots + \overline{m}_{c1}m_{c1} \\ \vdots \\ \overline{m}_{c1}m_{c1} + \cdots + \overline{m}_{cc}m_{cc} \end{pmatrix},$$

by the summation 9.5 this state vector is equal to

$$= \frac{1}{\sqrt{c}} \big((\overline{m}_{11}m_{11} + \cdots + \overline{m}_{c1}m_{c1}) \left|1\right\rangle \left|1\right\rangle + (\overline{m}_{12}m_{12} + \cdots + \overline{m}_{c2}m_{c2}) \left|2\right\rangle \left|2\right\rangle$$
$$+ \cdots + (\overline{m}_{c1}m_{c1} + \cdots + \overline{m}_{cc}m_{cc}) \left|c\right\rangle \left|c\right\rangle \big),$$

and using the summation 9.4 we get to

$$= \frac{1}{\sqrt{c}} \sum_{j=1}^{c} \left|j\right\rangle \left|j\right\rangle .$$

Because all the possible measurements outcomes are of the form jj, for $j = 1, \ldots, c$, Alice and Bob will answer the same color when they get the same vertex as input.

2) Alice and Bob get adjacent vertices v and w as input.

For this second scenario we need the following fact. Let $c \geq 2$ and

$$U_v = \begin{bmatrix} v_{11} & \cdots & v_{1c} \\ \vdots & \ddots & \vdots \\ v_{c1} & \cdots & v_{cc} \end{bmatrix}, \qquad U_w = \begin{bmatrix} w_{11} & \cdots & w_{1c} \\ \vdots & \ddots & \vdots \\ w_{c1} & \cdots & w_{cc} \end{bmatrix}.$$

The product $U_v^\dagger U_w$ is the $c \times c$ matrix

$$K = \begin{bmatrix} k_{11} & \cdots & k_{1c} \\ \vdots & \ddots & \vdots \\ k_{c1} & \cdots & k_{cc} \end{bmatrix}$$

such that

$$k_{qr} = \sum_{p=1}^{c} \overline{v}_{pq} w_{pr}, \text{ for } q, r = 1, \ldots, c.$$

By the consistency condition 9.3 K has only zeroes on the diagonal so we have that $k_{qr} = 0$ for all $q = r$.

We will show that in this second scenario there is no chance that Alice and Bob answer the same color to the referee. The state of the system evolves according to:

$$
\begin{aligned}
(U_v^\dagger \otimes U_w^\top)\left|\psi\right\rangle_{AB} &= (U_v^\dagger \otimes U_w^\top)\frac{1}{\sqrt{c}}\left(\left|1\right\rangle\left|1\right\rangle + \cdots + \left|c\right\rangle\left|c\right\rangle\right) \\
&= \frac{1}{\sqrt{c}}\left(\begin{pmatrix}\overline{v}_{11}\\ \vdots \\ \overline{v}_{1c}\end{pmatrix} \otimes \begin{pmatrix} w_{11}\\ \vdots \\ w_{1c}\end{pmatrix} + \cdots + \begin{pmatrix}\overline{v}_{c1}\\ \vdots \\ \overline{v}_{cc}\end{pmatrix} \otimes \begin{pmatrix} w_{c1}\\ \vdots \\ w_{cc}\end{pmatrix}\right) \\
&= \frac{1}{\sqrt{c}}\begin{pmatrix}\overline{v}_{11}w_{11} + \cdots + \overline{v}_{c1}w_{c1}\\ \vdots \\ \overline{v}_{c1}w_{c1} + \cdots + \overline{v}_{cc}w_{cc}\end{pmatrix} \\
&= \frac{1}{\sqrt{c}}\left(k_{11},\ \ k_{12},\ \ \cdots,\ \ k_{1c},\ \ k_{21},\ \ k_{22},\ \ \cdots\cdots,\ \ k_{c1},\ \ \cdots,\ \ k_{cc}\right)^\top,
\end{aligned}
$$

because of the analysis we did above the state of the system is equal to

$$
\begin{aligned}
&= \frac{1}{\sqrt{c}}\left(0,\ \ k_{12,}\ \ \cdots,\ \ k_{1c},\ \ k_{21},\ \ 0,\ \ \cdots\cdots,\ \ k_{c1},\ \ \cdots,\ \ 0\right)^\top \\
&= \frac{1}{\sqrt{c}}\sum_{\substack{q,r\in[c]\\ q\neq r}} k_{qr}\left|q\right\rangle\left|r\right\rangle.
\end{aligned}
$$

Because all the possible measurements outcomes are of the form qr, for $q, r \in [c]$ with $q \neq r$, Alice and Bob will never answer the same color when they get adjacent vertices as input. ■

Before we move to our goal to formulate the quantum strategy for the G_{18} in terms of unitary matrices we need to present a proof of the theorem below (adapted from Proposition 12 of [CMN$^+$06]). Our proof that relates an orthogonal representation of a graph in $\mathbb{R}^4$ with the quantum chromatic number will help us find unitary matrices to win the c-coloring game on the G_{18}. An *orthogonal representation*[4] of a graph G is a mapping ϕ from the vertices of G to the non-zero vectors of some vector space whose entries are taken from the set $\{-1, 0, 1\}$, such that if two vertices x and y are adjacent, then $\phi(x)$ and $\phi(y)$ are orthogonal.

Theorem 9.5.2 Adapted from [CMN$^+$06]
Let G be a graph with an orthogonal representation in $\mathbb{R}^4$. Then, $\chi_q^{(1)}(G) \leq 4$.

Proof: Let $G = (V, E)$ be a graph with n vertices and an orthogonal representation in $\mathbb{R}^4$. Then, by definition for every vertex $v \in V$ we can associate a vector l_v from the orthogonal representation such that:

$$\forall vw \in E,\ l_v \text{ and } l_w \text{ are orthogonal.}$$

[4]An orthogonal representation does not have any restriction on the entries of the vectors.

For every $l_v \in \mathbb{R}^4$ we will define a unitary matrix

$$U_v = \frac{1}{\sqrt{\sum_{j=1}^{4} |v_j|}} \begin{bmatrix} v_1 & v_2 & v_3 & v_4 \\ -v_2 & v_1 & -v_4 & v_3 \\ -v_3 & v_4 & v_1 & -v_2 \\ -v_4 & -v_3 & v_2 & v_1 \end{bmatrix}. \tag{9.6}$$

This matrix is unitary because for any $l_v \in \mathbb{R}^4$

$$U_v^\dagger U_v = \frac{1}{\sum_{j=1}^{4} |v_j|} \begin{bmatrix} v_1^2+v_2^2+v_3^2+v_4^2 & 0 & 0 & 0 \\ 0 & v_2^2+v_1^2+v_4^2+v_3^2 & 0 & 0 \\ 0 & 0 & v_3^2+v_4^2+v_1^2+v_2^2 & 0 \\ 0 & 0 & 0 & v_4^2+v_3^2+v_2^2+v_1^2 \end{bmatrix} = \mathrm{I}.$$

Now, let us verify that the consistency condition (9.3) holds. Let vw be any edge of G we will show that $\mathrm{diag}(U_v^\dagger U_w) = (0,0,0,0)$:

$$\begin{aligned} \mathrm{diag}(U_v^\dagger U_w) = \frac{1}{\sqrt{\left(\sum_{j=1}^{4} |v_j|\right)\left(\sum_{j=1}^{4} |w_j|\right)}} (& v_1w_1 + v_2w_2 + v_3w_3 + v_4w_4, \\ & v_2w_2 + v_1w_1 + v_4w_4 + v_3w_3, \\ & v_3w_3 + v_4w_4 + v_1w_1 + v_2w_2, \\ & v_4w_4 + v_3w_3 + v_2w_2 + v_1w_1), \end{aligned}$$

the four elements of the diagonal of $U_v^\dagger U_w$ are equal to the inner product of l_v with l_w. As these vectors are associated with the vertices of an edge of G they are orthogonal. Hence, $\mathrm{diag}(U_v^\dagger U_w) = (0,0,0,0)$ as desired.

In other words, if G has an orthogonal representation in $\mathbb{R}^4$, then Alice and Bob can associate each vector of the orthogonal representation with a unitary matrix of the form (9.6) and follow the strategy mentioned at the beginning of this section. We have proved that they will win the c-coloring game with $c = 4$ for G with certainty implying that $\chi_q^{(1)}(G) \leq 4$. ■

9.5.1 Quantum Chromatic Number of G_{18}

In [CMN+06] the authors verified that the classical chromatic number of the graph G_{18} is 5. For the quantum chromatic number they first provided the following list of vectors:

$$\begin{aligned} l_1, l_2, \ldots, l_{18} = \; & (0,0,1,-1), (1,0,0,0), (0,1,1,1), (0,1,0,-1), (0,0,1,0), \\ & (1,1,0,1), (1,-1,0,0), (0,0,0,1), (1,1,1,0), (1,0,-1,0), \\ & (0,1,0,0), (1,0,1,1), (0,1,-1,0), (1,0,0,-1), (1,1,1,1), \\ & (1,1,-1,-1), (1,-1,1,-1), (1,-1,-1,1), \end{aligned} \tag{9.7}$$

which gives an orthogonal representation of $G_{18} \in \mathbb{R}^4$ implying that $\chi_q^{(1)}(G) \leq 4$.

As we mentioned before the rank-1 quantum chromatic number is an upper bound of the quantum chromatic number. To conclude $\chi_q(G_{18}) = 4$ the authors observed that the G_{18} graph contains a 4-clique (vertices $15-18$). So by Proposition 9.4.2 they proved that $\chi_q(G) = 4$

As we promised at the beginning of this subsection we are going to simulate a round of the c-coloring game with the G_{18} graph. First Alice and Bob receive the graph G_{18} from the referee. They agree on the list of vectors (9.7) which gives an orthogonal representation in $\mathbb{R}^4$ and also on the unitary matrix given in (9.6). Before moving apart they share the entangled state

$$|\psi\rangle_{AB} = \frac{1}{\sqrt{4}} \sum_{j=1}^{4} |j\rangle_A |j\rangle_B ,$$

and inform the referee they have a 4-coloring of the graph. The game starts and the referee sends the vertex v_1 to Alice and the vertex v_3 to Bob.

Alice is going to apply the unitary matrix

$$U_1^\dagger = \frac{1}{\sqrt{2}} \begin{bmatrix} 0 & 0 & -1 & 1 \\ 0 & 0 & -1 & -1 \\ 1 & 1 & 0 & 0 \\ -1 & 1 & 0 & 0 \end{bmatrix}, \tag{9.8}$$

given by l_1 and Bob the unitary matrix

$$U_3^\top = \frac{1}{\sqrt{3}} \begin{bmatrix} 0 & -1 & -1 & -1 \\ 1 & 0 & 1 & -1 \\ 1 & -1 & 0 & 1 \\ 1 & 1 & -1 & 0 \end{bmatrix}, \tag{9.9}$$

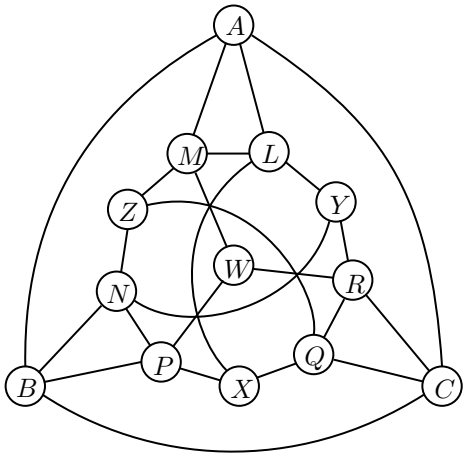
given by l_3. By definition, the system evolves according to:

$$\begin{aligned}
(U_1^\dagger \otimes U_3^\top) |\psi\rangle_{AB} &= (U_1^\dagger \otimes U_3^\top) \frac{1}{2} (|1\rangle |1\rangle + |2\rangle |2\rangle + |3\rangle |3\rangle + |4\rangle |4\rangle) \\
&= \frac{1}{2\sqrt{6}} \big((|3\rangle - |4\rangle)(|2\rangle + |3\rangle + |4\rangle) + (|3\rangle + |4\rangle)(-|1\rangle - |3\rangle + |4\rangle) \\
&\qquad + (-|1\rangle - |2\rangle)(-|1\rangle + |2\rangle - |4\rangle) + (|1\rangle - |2\rangle)(-|1\rangle - |2\rangle + |3\rangle) \big) \\
&= \frac{1}{2\sqrt{6}} \big(|3\rangle |2\rangle + 2|3\rangle |4\rangle - |4\rangle |2\rangle - 2|4\rangle |3\rangle - |3\rangle |1\rangle - |4\rangle |1\rangle \\
&\qquad - 2|1\rangle |2\rangle + |1\rangle |4\rangle + 2|2\rangle |1\rangle + |2\rangle |4\rangle + |1\rangle |3\rangle - |2\rangle |3\rangle \big).
\end{aligned}$$

Because all the possible measurement outcomes are different Alice and Bob send different colors to the referee.

9.6 Another Graph with $\chi > \chi_q$

Mančinska and Roberson in 2016 [MR18] discovered the smallest known example of a graph with quantum chromatic number smaller than the classical chromatic number. They start by considering the G_{13} graph



G_{13}

which has an orthogonal representation in $\mathbb{R}^3$ given by the following list of vectors:

$$\begin{aligned} l_A, l_B, l_C, l_Q, l_R, l_N, l_P, l_L, l_M, l_W, l_Y, l_X, l_Z = \; & (1,0,0), (0,1,0), (0,0,1), (1,1,0), (1,-1,0), \\ & (1,0,1), (1,0,-1), (0,1,1), (0,1,-1), \\ & (1,1,1), (1,1,-1), (1,-1,1), (-1,1,1). \end{aligned}$$

Next, they considered a graph denoted as G_{14} that can be obtained by adding an extra vertex to G_{13} and making it adjacent to all the 13 vertices of G_{13}. This additional vertex is called an *apex* vertex denoted by Ω. If we consider the orthogonal representation of G_{13} we can construct an orthogonal representation of G_{14} by adding a coordinate which is zero and assign the vector $(0,0,0,1)$ to Ω.

First, the authors using Theorem 9.5.2 noted that $\chi_q(G_{13}) \leq 4$. Next, they gave an explicit proof that the quantum chromatic number of G_{13} is 4 which is equal to the classical chromatic number. Again, by Theorem 9.5.2 $\chi_q(G_{14}) \leq 4$ but as $\chi_q(G_{14}) \geq \chi_q(G_{13}) = 4$ they concluded that

$$\chi_q(G_{14}) = \chi_q(G_{13}) = 4.$$

Adding an apex vertex to any graph would increase the chromatic number by one unit. Surprisingly, for the G_{13} the quantum chromatic number remained unchanged after this addition. As a further remark the authors suspected that $\chi(G) = \chi_q(G)$ for any graph G with fewer than 14 vertices.

Chapter 10

Other Quantum Graph Parameters

This final chapter is dedicated to the study of two other quantum graph parameters quantum homomorphism and the quantum independence number. We are interested in two aspects of these parameters: first, presenting the main difference from these parameters with the quantum chromatic number, and second, exhibiting graphs whose classical and quantum parameters differ. For a thorough introduction to these topics, we refer the reader to [MR16] and [Sca13].

10.1 Quantum Homomorphisms

Mančinska and Roberson in 2016 [MR16], besides other results, studied a generalization of the c-coloring game called the (X, Y)-homomorphism game. In this nonlocal game Alice and Bob's goal is to convince a referee that a graph G admits a homomorphism to a graph H with certainty.

Let us formally define this game. In the (X, Y)-homomorphism game Alice and Bob receive a pair of graphs X and Y from a referee. They agree on a strategy to convince the referee that $X \to Y$ who will test their claim with a one-round game. The game starts and the players are not allowed to communicate. The referee sends vertices x_A and x_B of X to Alice and Bob, respectively. Instead of sending two colors, as they did in the c-coloring game, the players will send two vertices of Y to the referee. Alice sends y_A and Bob sends y_B. They win the (X, Y)-homomorphism game if the following conditions are satisfied:

$$\text{if } x_A = x_B, \text{ then } y_A = y_B;$$
$$\text{if } x_A \sim x_B, \text{ then } y_A \sim y_B.$$

To understand why this game is a generalization of the c-coloring game note that a homomorphism from a graph X to the complete graph K_c is equivalent to a

c-coloring of X. As K_c is c-colorable we can assign a color to each vertex x of X with the color assigned to $\phi(x)$ of K_c. This argument implies that when $Y = K_c$ the (X, Y)-homomorphism game reduces to the c-coloring game.

By an argument similar to the classical strategy of the c-coloring game (see Theorem 8.1.2) classical players can only win the (X, Y)-homomorphism game if $X \to Y$.

A general quantum strategy for this game is identical to the one presented for the c-coloring game except for the families of POVMs used by the players. For all $x \in V(X)$, Alice has $\{E_\alpha^x\}_{\alpha \in V(Y)}$ obtaining some outcome $\alpha \in V(Y)$ when she measures her part of the entangled state $|\psi\rangle_{AB} \in \mathbb{C}^d \otimes \mathbb{C}^d$. Bob has $\{F_\beta^x\}_{\beta \in V(Y)}$ obtaining some outcome $\beta \in V(Y)$ when he measures his part of the entangled state. For this game the consistency condition is:

- The players will never output different vertices when they got the same vertex x as input:
$$\forall x \in V(X), \forall \alpha \neq \beta, \langle\psi| E_\alpha^x \otimes F_\beta^x |\psi\rangle = 0.$$
- The player will never output the same vertex for any adjacent vertices $x_A x_B$ they got as input:
$$\forall x_A x_B \in E(X), \forall \alpha \not\sim \beta, \langle\psi| E_\alpha^{x_A} \otimes F_\beta^{x_B} |\psi\rangle = 0.$$

Whenever quantum player can win this game we are going to say that $X \xrightarrow{q} Y$. The same observation we did on Section 9.3 about the notation of the POVM also applies to this consistency condition.

In the c-coloring game we were interested in graphs that quantum players can win with a $c < \chi(G)$. Up until now the only examples of graphs X and Y such that $X \not\to Y$ but $X \xrightarrow{q} Y$ are with one of the graphs being complete or when K_n is a subgraph of Y for some positive integer n [MR16]. For instance the (G_{18}, K_4)-homomorphism game[1] is a nonlocal game because $G_{18} \not\to K_4$ but $G_{18} \xrightarrow{q} K_4$.

To end this section, we note that in the same manner several graph parameters can be defined in terms of homomorphisms it is also possible to define quantum analogs of these parameters. By simply replacing "homomorphism" with "quantum homomorphism" in the definition we have the following:

$$\text{quantum chromatic number:} \qquad \chi_q(G) \coloneqq \min\{n \in \mathbb{N} \mid G \xrightarrow{q} K_n\},$$

[1] We could also say that the (G_{14}, K_4)-homomorphism game is a nonlocal game by the same argument

quantum clique number: $\omega_q(G) \coloneqq \max\{n \in \mathbb{N} \mid K_n \xrightarrow{q} G\}$.

10.2 Quantum Independence Number

Another graph parameter presented in [MR16] was the quantum independence number. Let us first define the classical independence number before we move to the quantum counterpart. An *independent set* is a set of vertices in a graph such that no two elements of this set are adjacent. An independent set is *maximum* if the graph contains no larger independent set. The cardinality of a maximum independent set of G is called the *independence number* and is denoted by $\alpha(G)$.

Another way we could define the independence number would be using the clique number of a graph, for this, we need to define the complement of a graph. The *complement* of a graph G, denoted by $\overline{G}$ is a graph with $V(G) = V(\overline{G})$ such that two distinct vertices of $\overline{G}$ are adjacent if and only if they are not adjacent in G. With these definitions we have that $\alpha(G) \coloneqq \omega(\overline{G})$, and in the quantum framework $\alpha_q(G) \coloneqq \omega_q(\overline{G})$.

Scarpa in [Sca13] proposed the following[2] nonlocal game that gives an alternative definition to $\alpha_q(G)$ that is closely related to the c-coloring game. In the t-independent set game Alice and Bob receive a graph G from the referee. They agree on a strategy to convince the referee that they know a t-tuple $(v_1, \ldots, v_t)$ whose vertices can be used to make an independent set I of G. The referee will test their claim with a one-round game. After the game starts the players are not allowed to communicate. The referee chooses two numbers $x, y \in [t]$ and separately asks Alice to provide the v_x vertex of the t-tuple and Bob to provide the v_y vertex of the t-tuple. They win the t-independent set game if the following consistency condition is satisfied:

$$\text{if } x = y, \text{ then } v_x = v_y;$$
$$\text{if } x \neq y, \text{ then } v_x \not\sim v_y.$$

By an argument similar to the classical strategy of the c-coloring game (see Theorem 8.1.2) classical players cannot win this game with certainty when $t > \alpha(G)$. The general quantum strategy is almost the same as the one presented for the c-coloring

[2] We have in fact made some adaptations to his version of the game.

game. The only difference is in the families of POVMs that are defined using the vertices from I the independent set of G.

Now that we have presented this nonlocal game we can define $\alpha_q(G)$ without quantum homomorphism. For all graphs G, the *quantum independence number* is the maximum number t for which Alice and Bob can convince the referee that the consistency condition above always hold for the t-independent set game played with the graph G.

As mentioned earlier the goal of this section will be to explicitly calculate the independence and quantum independence number of graphs in which these parameters differs. We are interested in graphs G such that $\alpha(G) < \alpha_q(G)$. To achieve this goal we will use Theorem 3.4.8 from [Sca13]. Before we move to this theorem we need to define the Cartesian product of graphs and also three lemmas.

Definition 10.2.1
The Cartesian product of graphs G and H, denoted by $G\square H$ is a graph such that: The vertex set of $G\square H$ is the Cartesian product $V(G) \times V(H)$ and two vertices (u, u') and (v, v') are adjacent in $G\square H$ if and only if either $u = v$ and u' is adjacent to v' in H, or $u' = v'$ and u is adjacent to v in G.

Now that we have defined the Cartesian product of graphs we can move to the following lemmas:

Lemma 10.2.2 [Sca13]
Let G be a graph on n vertices with $\chi(G) > k$. Then we have $\alpha(G\square K_k) < n$.

Proof. Consider the graph $G\square K_k$. By the definition of the Cartesian product the vertex set of $G\square K_k$ can be partitioned into n disjoint cliques of size k. Towards a contradiction, suppose $\alpha(G\square K_k) \geq n$. As we can pick at most one vertex from each clique suppose $\alpha(G\square K_k) = n$ and let I be an independent set of size n. Using the graph $G\square K_k$ we can get a k-coloring of G, as follows: if $(v, k_i) \in I$ we will color $v \in V(G)$ with color i, where $i \in [k]$. This is a proper k coloring of G because by the definition of the Cartesian product of graphs, for all $v \sim u \in E(G)$ we have $((v, k_i) \sim (u, k_i)) \in E(G\square K_k)$, implying that the vertices v and u will not both get color i. This contradicts the assumption that $\chi(G) > k$. ■

The other result we need is:

Lemma 10.2.3 [Sca13]
Let G be a graph on n vertices and $\chi_q(G) \leq k$. Then we have $\alpha_q(G\square K_k) = n$.

We will omit the proof of this lemma (that can be found on Section 3.4.2 of [Sca13]) as it needs further concepts that are out of the scope of this section.

Finally, the last lemma is due to Vizing

Lemma 10.2.4 [Viz63]
For all graphs G, H, the independence number of their Cartesian product satisfies

$$\alpha(G\square H) \leq \min\{\alpha(G)\cdot|V(H)|,\ \alpha(H)\cdot|V(G)|\}$$

Proof. Assume without loss of generality that $\alpha(G)\cdot|V(H)| \leq \alpha(H)\cdot|V(G)|$. Suppose, towards a contradiction that $\alpha(G\square H) > \alpha(G)\cdot|V(H)|$, then there is a $h \in V(H)$ such that there are more than $\alpha(G)$ non-adjacent vertices of $G\square H$ of the form (g,h). This implies the existence of an independent set of G of size larger than $\alpha(G)$, because there is an edge between (v,i) and (w,i) whenever $vw \in E(G)$. ■

Combining these three results we obtain the following.

Theorem 10.2.5 [Sca13]
Let G be a graph on n vertices with $\chi(G) > \chi_q(G) = k$. Then:

1) $\alpha(G\square K_k) < \alpha_q(G\square K_k) = n$.
2) $\alpha(G\square K_k) \leq \alpha(G)\cdot k$.

Proof. For the first bound we note that the graph G with n vertices has $\chi(G) > k$ and $\chi_q(G) = k$ we have from Lemma 10.2.2 that $\alpha(G\square K_k) < n$ and from Lemma 10.2.3 that $\alpha_q(G\square K_k) = n$. Combining these two results gives the desired inequality.

For the second bound it follows from Lemma 10.2.4 that

$$\alpha(G\square K_k) \leq \min\{\alpha(G)\cdot k,\ 1\cdot n\},$$

which gives the desired inequality. ■

Inspired by these results we computed the independence and quantum independence number of the following graphs:

G	$\lvert V(G)\rvert$	$\lvert E(G)\rvert$	$\alpha(G)$	$\alpha_q(G)$
$G_{18}\square K_4$	72	284	17	18
$G_{14}\square K_4$	56	232	13	14

In Section 8.2 we saw that Hadamard graphs exhibit an exponential separation between quantum and classical chromatic number. Now, we are going to build another family of graphs that exhibits a separation in the quantum and classical independence number. Each member of this new family is a Cartesian product of a Hadamard graph with a complete graph.

To show this gap consider the Hadamard graph H_N. Let $N = 4k \geq 12$. By Theorem 8.2.1 $\chi_q(H_N) \leq N$ while $\chi(H_N) > N$. Consider the graph $H_N \square K_N$. By Lemma 10.2.3 if $\chi_q(H_N) \leq N$ then $\alpha_q(H_N \square K_N) = |V(H_N)| = 2^N$. For the classical case from [FR87] it follows that for the same N, there exists an $\epsilon > 0$ such that $\alpha(H_N) \leq (2-\epsilon)^N$. By the second bound of Theorem 10.2.5 we have that $\alpha(H_N \square K_N) \leq (2-\epsilon)^N \cdot N$.

In particular, using a result from [dKP07] that proved that $\alpha(H_{16}) = 2304$ and the discussion above we have that

$$\alpha_q(H_{16} \square K_{16}) = |V(H_{16})| = 2^{16} = 65536,$$

while

$$\alpha(H_{16} \square K_{16}) \leq \alpha(H_{16}) \cdot 16 = 36864.$$

Appendix A

Bob's Matching Theorem

The following theorem allows us to prove that with probability at least $1/2$, Bob's matching M contains an $\{i, j\}$ with $i \in S_1$ and $j \in S_2$.

Bob's Matching Theorem
Let n be an even perfect square and

$$S \in_u \binom{n}{2\sqrt{n}}.$$

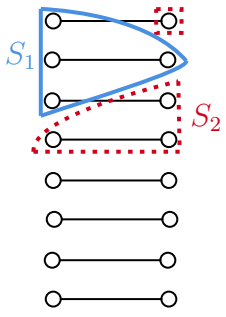
Consider the disjoints subsets

$$S_1 \in_u \binom{S}{\sqrt{n}} \text{ and } S_2 := S \setminus S_1,$$

and let M be a matching taken uniformly at random from the set of all perfect matchings on $[n]$ (a partition into $n/2$ disjoints pairs of $[n]$).
Bob's matching theorem states that:

$$\mathbb{P}(\exists \{i, j\} \in M, i \in S_1 \text{ and } j \in S_2) \geq 1/2.$$

Before we start our proof it is useful to think about the perfect matching M as a graph. For instance, for $n = 16$ we could have the following sets S_1 and S_2



in this example we have exactly two edges with one endpoint in S_1 and one endpoint in S_2.

Proof. Let n, S_1, S_2 and M as defined in the theorem. The total amount of events, expressed by T, is equal to the total amount of sets S_1 and S_2 we can make:

$$T = \binom{n}{\sqrt{n}}\binom{n-\sqrt{n}}{\sqrt{n}}.$$

Our first goal will be to calculate the total possible amount of bad events. Given M, S_1 and S_2 we define a *bad event* as the scenario where there is not a single edge with one endpoint in S_1 and the other endpoint in S_2.

After calculating the total number of bad events we will prove our theorem by showing that

$$\mathbb{P}(\exists\{i,j\} \in M, i \in S_1 \text{ and } j \in S_2) = 1 - \frac{\#\text{ bad events}}{T} \geq \frac{1}{2}.$$

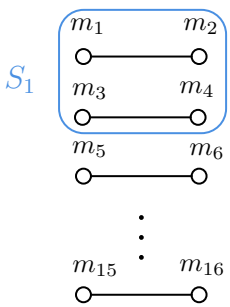
For that we need to define two sets, let

$$D = \{\{i_1, i_2\} \mid i_1, i_2 \in S_1 \text{ and } \{i_1, i_2\} \in M\},$$

and

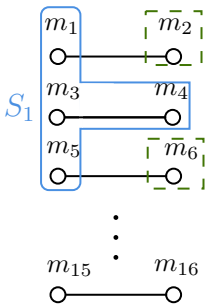
$$B = \{v \in [n] \setminus S_1 \mid \{i, v\} \in M \text{ for some } i \in S_1\}.$$

The members of the set B are the ones that if they belonged to S_1 they would be part of an element of D. Below are three possible examples for $n = 16$:



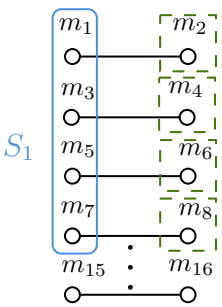
$D = \{m_1m_2, m_3m_4\}$

$B = \emptyset$



$D = \{m_3m_4\}$

$B = \{m_2, m_6\}$



$D = \emptyset$

$B = \{m_2, m_4, m_6, m_8\}$

It is straightforward to see that the size of the set D ranges from zero to $\sqrt{n}/2$ (inclusively). And we can write the size of the set B as $|B| = \sqrt{n} - 2|D|$. With this last equality we can calculate the size of $|S_1 + B| = \sqrt{n} + \sqrt{n} - 2|D| = 2(\sqrt{n} - |D|)$.

Some useful observations are that when $|D| = \sqrt{n}/2$ it is impossible to have a *good event* at least one edge with one endpoint in S_1 and the other endpoint in S_2. But when $|D| = 0$ to have just bad events we must have $|B| = \sqrt{n}$.

The total amount of bad events (T_B) is given by

$$T_B = \sum_{d=0}^{\sqrt{n}/2} \underbrace{\overbrace{\binom{n/2}{d}}^{\#\text{type } d} \overbrace{\binom{n/2-d}{\sqrt{n}-2d}}^{\#\text{ not type } d} \overbrace{2^{\sqrt{n}-2d}}^{\#\text{ possible combinations}}}_{|S_1+B|} \underbrace{\binom{n-2(\sqrt{n}-d)}{\sqrt{n}}}_{|S_2|}.$$

Evaluating T_B/T for different values of n we note that this quantity converges to $1/e$ as n gets bigger. Let $R_n = T_B/T$ for $n = 4^3, 4^4, \ldots, 4^{10}$:

n	R_n	$R_n - R_{n-1}$	$1/e - R_n$
4^3	0.31774		0.050139
4^4	0.34388	0.026145	0.023993
4^5	0.35613	0.012252	0.011741
4^6	0.36207	0.005932	0.005808
4^7	0.36499	0.002919	0.002889
4^8	0.36643	0.001448	0.001440
4^9	0.36715	0.000721	0.000719
4^{10}	0.36751	0.000360	0.000359

With this analysis, we can conclude that the probability of success is $1 - T_B/T \approx 1 - 1/e \approx 0.63$. This completes the proof that

$$\mathbb{P}(\exists\{i,j\} \in M, i \in S_1 \text{ and } j \in S_2) \geq 1/2. \blacksquare$$

Acknowledgement

The authors would like to thank Jared León for his help on this prove.

Bibliography

[AAB+19] Frank Arute, Kunal Arya, Ryan Babbush, Dave Bacon, Joseph C Bardin, Rami Barends, Rupak Biswas, Sergio Boixo, Fernando GSL Brandao, David A Buell, et al. Quantum supremacy using a programmable superconducting processor. *Nature*, 574(7779):505–510, 2019.

[AHKS06] David Avis, Jun Hasegawa, Yosuke Kikuchi, and Yuuya Sasaki. A quantum protocol to win the graph colouring game on all hadamard graphs. *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, 89(5):1378–1381, 2006.

[Amb04] Andris Ambainis. Quantum search algorithms. *ACM SIGACT News*, 35(2):22–35, 2004.

[Axl15] Sheldon Axler. *Linear algebra done right*, volume 2. Springer, 2015.

[BBBV97] Charles H Bennett, Ethan Bernstein, Gilles Brassard, and Umesh Vazirani. Strengths and weaknesses of quantum computing. *SIAM Journal on Computing*, 26(5):1510–1523, 1997.

[BCMdW10] Harry Buhrman, Richard Cleve, Serge Massar, and Ronald de Wolf. Nonlocality and communication complexity. *Reviews of Modern Physics*, 82(1):665–698, Mar 2010. URL: `http://dx.doi.org/10.1103/RevModPhys.82.665`, `doi:10.1103/revmodphys.82.665`.

[BCT99] Gilles Brassard, Richard Cleve, and Alain Tapp. Cost of exactly simulating quantum entanglement with classical communication. *Physical Review Letters*, 83(9):1874, 1999.

[BCW98] Harry Buhrman, Richard Cleve, and Avi Wigderson. Quantum vs. classical communication and computation, 1998. `arXiv:quant-ph/9802040`.

[Bel64] John S Bell. On the Einstein Podolsky Rosen paradox. *Physics Physique Fizika*, 1(3):195, 1964.

[BHMT02] Gilles Brassard, Peter Høyer, Michele Mosca, and Alain Tapp. Quantum amplitude amplification and estimation. *Quantum Computation and Information*, page 53–74, 2002. URL: http://dx.doi.org/10.1090/conm/305/05215, doi:10.1090/conm/305/05215.

[BV97] Ethan Bernstein and Umesh Vazirani. Quantum complexity theory. *SIAM Journal on Computing*, 26(5):1411–1473, 1997.

[BYJK04] Ziv Bar-Yossef, Thathachar S Jayram, and Iordanis Kerenidis. Exponential separation of quantum and classical one-way communication complexity. In *Proceedings of the thirty-sixth annual ACM symposium on Theory of computing*, pages 128–137, 2004.

[CEMM98] Richard Cleve, Artur Ekert, Chiara Macchiavello, and Michele Mosca. Quantum algorithms revisited. *Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, 454(1969):339–354, 1998.

[Cir80] Boris S Cirel'son. Quantum generalizations of Bell's inequality. *Letters in Mathematical Physics*, 4(2):93–100, 1980.

[CMN+06] Peter J Cameron, Ashley Montanaro, Michael W Newman, Simone Severini, and Andreas Winter. On the quantum chromatic number of a graph. *arXiv preprint quant-ph/0608016*, 2006.

[Deu85] David Deutsch. Quantum theory, the Church–Turing principle and the universal quantum computer. *Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences*, 400(1818):97–117, 1985.

[DJ92] David Deutsch and Richard Jozsa. Rapid solution of problems by quantum computation. *Proceedings of the Royal Society of London. Series A: Mathematical and Physical Sciences*, 439(1907):553–558, 1992.

[dKP07] Etienne de Klerk and Dmitrii V Pasechnik. A note on the stability number of an orthogonality graph. *European Journal of Combinatorics*, 28(7):1971–1979, 2007.

[DKW05] Evgeny Dantsin, Vladik Kreinovich, and Alexander Wolpert. On quantum versions of record-breaking algorithms for sat. *SIGACT News*, 36(4):103–108, dec 2005. doi:10.1145/1107523.1107524.

[DN05] Christopher M Dawson and Michael A Nielsen. The Solovay–Kitaev algorithm. *arXiv preprint quant-ph/0505030*, 2005.

[dW21] Ronald de Wolf. Quantum computing: Lecture notes, 2021. `arXiv:1907.09415`.

[Fey82] Richard Feynman. Simulating physics with computers. *International Journal of Theoretical Physics*, 21(6/7):467–488, 1982.

[FR87] Peter Frankl and Vojtěch Rödl. Forbidden intersections. *Transactions of the American Mathematical Society*, 300(1):259–286, 1987.

[GN08] Chris D Godsil and Michael W Newman. Coloring an orthogonality graph. *SIAM Journal on Discrete Mathematics*, 22(2):683–692, 2008.

[Gro96] Lov K Grover. A fast quantum mechanical algorithm for database search. In *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing*, pages 212–219, 1996.

[GTW03] Viktor Galliard, Alain Tapp, and Stefan Wolf. The impossibility of pseudotelepathy without quantum entanglement. In *IEEE International Symposium on Information Theory, 2003. Proceedings.*, page 457. IEEE, 2003.

[GW02] Viktor Galliard and Stefan Wolf. Pseudo-telepathy, entanglement, and graph colorings. In *Proceedings IEEE International Symposium on Information Theory,*, page 101. IEEE, 2002.

[HK71] Kenneth M. Hoffman and Ray Kunze. *Linear Algebra, Second Edition*. Prentice Hall, 1971.

[Hol73a] Alexander Semenovich Holevo. Bounds for the quantity of information transmitted by a quantum communication channel. *Problemy Peredachi Informatsii*, 9(3):3–11, 1973.

[Hol73b] Alexander Semenovich Holevo. Information-theoretical aspects of quantum measurement. *Problemy Peredachi Informatsii*, 9(2):31–42, 1973.

[Kit97] Aleksei Yur'evich Kitaev. Quantum computations: algorithms and error correction. *Uspekhi Matematicheskikh Nauk*, 52(6):53–112, 1997.

[MR16] Laura Mančinska and David E Roberson. Quantum homomorphisms. *Journal of Combinatorial Theory, Series B*, 118:228–267, 2016.

[MR18] Laura Mančinska and David E Roberson. Oddities of quantum colorings. *arXiv preprint arXiv:1801.03542*, 2018.

[Nan20] Giacomo Nannicini. An introduction to quantum computing, without the physics. *SIAM Review*, 62(4):936–981, 2020.

[NC02] Michael A Nielsen and Isaac Chuang. Quantum computation and quantum information, 2002.

[Per97] Asher Peres. *Quantum theory: concepts and methods*. Springer, 1997.

[R+03] Daniel Rolf et al. 3-SAT in RTIME (1.32971ˆn). *citeseerx/10.1.1.13.4171*, 2003.

[Sca13] Giannicola Scarpa. *Quantum entanglement in non-local games, graph parameters and zero-error information theory*. ILLC, 2013.

[Sel12] Peter Selinger. Efficient Clifford+T approximation of single-qubit operators. *arXiv preprint arXiv:1212.6253*, 2012.

[Sho94] Peter W Shor. Algorithms for quantum computation: discrete logarithms and factoring. In *Proceedings 35th Annual Symposium on Foundations of Computer Science*, pages 124–134. IEEE, 1994.

[Sim97] Daniel R Simon. On the power of quantum computation. *SIAM Journal on Computing*, 26(5):1474–1483, 1997.

[Viz63] Vladim G Vizing. The cartesian product of graphs. *Vycisl. Sistemy*, 9(30-43):33, 1963.

[ZWD+20] Han-Sen Zhong, Hui Wang, Yu-Hao Deng, Ming-Cheng Chen, Li-Chao Peng, Yi-Han Luo, Jian Qin, Dian Wu, Xing Ding, Yi Hu, et al. Quantum computational advantage using photons. *Science*, 370(6523):1460–1463, 2020.