On Quantum Computation Theory

Wim van Dam

On Quantum Computation Theory

ILLC Dissertation Series 2002-04

INSTITUTE FOR LOGIC, LANGUAGE AND COMPUTATION

For further information about ILLC-publications, please contact

Institute for Logic, Language and Computation
Universiteit van Amsterdam
Plantage Muidergracht 24
1018 TV Amsterdam
phone: +31-20-525 6051
fax: +31-20-525 5206
e-mail: `illc@wins.uva.nl`
homepage: `http://www.illc.uva.nl/`

On Quantum Computation Theory

ACADEMISCH PROEFSCHRIFT

ter verkrijging van de graad van doctor aan de
Universiteit van Amsterdam
op gezag van de Rector Magnificus
prof. mr. P.F. van der Heijden
ten overstaan van een door het college voor
promoties ingestelde commissie, in het openbaar
te verdedigen in de Aula der Universiteit
op woensdag 9 oktober 2002, te 14.00 uur

door

Willem Klaas van Dam

geboren te Breda.

Promotor:
Prof. dr. P.M.B. Vitányi

Overige leden promotiecommissie:
prof. dr. H.M. Buhrman
prof. dr. A.K. Ekert
prof. dr. B. Nienhuis
dr. L. Torenvliet

Faculteit der Natuurwetenschappen, Wiskunde en Informatica

Second Version, February 2004

ISBN: 90–5776–091–6

“ . . . Many errors have been made in the world which today, it seems, even a child would not have made. How many crooked, out-of-the-way, narrow, impassable, and devious paths has humanity chosen in the attempt to attain eternal truth, while before it the straight road lay open, like the road leading to a magnificent building destined to become a royal palace. It is wider and more resplendent than all the other paths, lying as it does in the full glare of the sun and lit up by many lights at night, but men have streamed past it in blind darkness. And how many times even when guided by understanding that has descended upon them from heaven, have they still managed to swerve away from it and go astray, have managed in the broad light of day to get into the impassable out-of-the-way places again, have managed again to throw a blinding mist over each other’s eyes and, running after will-o’-the-wisps, have managed to reach the brink of the precipice only to ask themselves afterwards with horror: ‘Where is the way out? Where is the road?’ The present generation sees everything clearly, it is amazed at the errors and laughs at the folly of its ancestors, unaware that this chronicle is shot through with heavenly fires, that every letter in it cries out aloud to them, that from everywhere, from every direction an accusing finger is pointed at it, at the present generation; but the present generation laughs and proudly and self-confidently enters on a series of fresh errors at which their descendants will laugh again later on.”

from “Dead Souls” by Nikolai Gogol
(translated by David Magarshack)

Abstract

The content of the Chapters 3 through 7 of this Ph.D. thesis corresponds with the following articles written by the author.

Chapter 3: Quantum Oracle Interrogation

- “Quantum Oracle Interrogation: Getting all information for almost half the price”, Wim van Dam, in *Proceedings of the 39th Annual IEEE Symposium on Foundations of Computer Science,* pages 362–367 (1998); quant-ph report no. 9805006

Chapter 4: Bounded Quantum Queries

- “Two Classical Queries versus One Quantum Query”, Wim van Dam, quant-ph report no. 9806090
- “Bounded Quantum Query Complexity”, Harry Buhrman and Wim van Dam, in *Proceedings of the 14th Annual IEEE Conference on Computational Complexity,* pages 149–156 (1999); quant-ph report no. 9903035

Chapter 5: Quantum Algorithms and Combinatorics

- “Quantum Algorithms for Weighing Matrices and Quadratic Residues”, Wim van Dam, quant-ph report no. 0008059; to appear in Algorithmica

Chapter 6: Self-Testing of Quantum Gates

- “Self-Testing of Universal and Fault-Tolerant Sets of Quantum Gates”, Wim van Dam, Frédéric Magniez, Michele Mosca and Miklos Santha, in *Proceedings of the 32nd Annual ACM Symposium on Theory of Computing,* pages 688–696 (2000); quant-ph report no. 994108

Chapter 7: Quantum Kolmogorov Complexity

- “Quantum Kolmogorov Complexity”, André Berthiaume, Wim van Dam and Sophie Laplante, in *Proceedings of the 15th Annual IEEE Conference on Computational Complexity,* pages 240–249 (2000); *Journal of Computer and Systems Sciences,* Volume 63, No. 2, pages 201–221 (2001); quant-ph report no. 0005018

Contents

Abstract **vii**

Acknowledgments **xiii**

1 Theory of Quantum Mechanics **1**

1.1 Modeling Information 1
1.2 Quantum Information 2
1.3 Time Evolution of Quantum Bits 3
1.4 Measurements 3
1.5 Limitations of Dirac's Notation 4
1.6 Density Matrices 4
1.7 Separated Systems 6
1.8 Von Neumann Entropy and Fidelity 6
1.9 Operations on Mixed States 8
1.10 Operator Sum Representation 9

2 Quantum Information and Computation **11**

2.1 Some Elementary Operations 11
2.2 Fault Tolerant and Universal Quantum Gates 12
2.3 Quantum versus Classical Query Complexity 13
2.4 Earlier Results in Quantum Computing 14
2.5 More Classic Quantum Results 15
2.6 Notation 17

3 Quantum Oracle Interrogation **19**

3.1 Introduction 19
3.2 Known Quantum Query Complexity Bounds 20
3.3 Definition of the Interrogation Problem 20
3.4 The Quantum Algorithm 21

3.5 Comparison with Classical Algorithms 24
3.6 Approximate Interrogation 24
3.7 Classical Approximate Interrogation 24
3.8 Quantum Approximate Interrogation 25
3.9 The Expected Number of Correct Bits 25
3.10 Interrogation with One Quantum Query 26
3.11 Interrogation with Many Queries 27
3.12 Conclusions 27

4 Quantum Bounded Queries **29**
4.1 Introduction 29
4.2 Classical Complexity Theory 31
4.3 Quantum Complexity Classes 32
4.4 Decision Problems 33
4.5 Functions computable with queries to NP Oracles 35
4.6 Terseness, and other Complexity Classes 37
4.7 Conclusions and Open Problems 38

5 Quantum Algorithms and Combinatorics **39**
5.1 Combinatorics, Hadamard and Weighing Matrices 39
5.2 Quantum Algorithms for Weighing Matrices 41
5.3 Quadratic Residues of Finite Fields 44
5.4 Finite Field Factoids 44
5.5 Multiplicative Characters over Finite Fields 44
5.6 The shifted Legendre Symbol Problem 46
5.7 Conclusion 48

6 Self-Testing of Quantum Gates **49**
6.1 Introduction 49
6.2 The Bloch Ball representation 51
6.3 Norm and Distance 53
6.4 Norms on Superoperators 54
6.5 Properties of CPSOs 55
6.6 Characterization of CPSO Families 57
6.7 Characterization of CNot gates 61
6.8 Robustness 62
6.9 Quantum Self-Testers 64

7 Quantum Kolmogorov Complexity **67**
7.1 Introduction 67
7.2 Desired Properties 68
7.3 Classical Kolmogorov complexity 69
7.4 Quantum Information Theory 70

7.5 Symmetric Subspaces 71
7.6 Accumulation of Errors 72
7.7 Quantum Kolmogorov Complexity 73
7.8 Input/Output Conventions 73
7.9 Defining Quantum Kolmogorov Complexity 74
7.10 Invariance 75
7.11 Properties of Quantum Kolmogorov Complexity 77
7.12 Correspondence for Classical Strings 77
7.13 Quantum Incompressibility 77
7.14 The Complexity of Copies 79
7.15 Subadditivity 80
7.16 The Complexity of Correlations 81
7.17 Extensions and Future Work 82

A Complexity Classes and Reductions **85**
A.1 Complexity Classes 85
A.2 Reductions 86
A.3 Query Complexity 86

B Properties of Matrices **87**
B.1 Properties and Transformations 87
B.2 Decompositions 88

C Norms and Distances **89**
C.1 Norms and Distances on Vectors and Matrices 89
C.2 Norms on Superoperators 91

D Approximate Interrogation **93**

Bibliography **97**

Samenvatting **107**

Acknowledgments

The first person I want to thank is my promotor Paul Vitányi. He let me work on my thesis in complete freedom while strongly supporting my academic endeavors, no matter where they took me. It is no accident that almost all of Paul's students end up enjoying successful scientific careers while having poor relations with the bureaucratic parts of society. His view is that research should be world-class and that the pencil-pushers who distract from this goal should be dealt with in the most time-efficient manner. It clearly serves him and his students well.

It was a pleasure working at C.W.I., and this is mostly due to the wonderful people who have sojourned there over the past years. I especially thank my office mates and neighbors Barbara Terhal, Harry Buhrman, John Tromp, Peter Grünwald, Ronald de Wolf, Peter Gács, Louis Salvail, Richard Cleve, Ronald Cramer, Lance Fortnow, Dieter van Melkebeek, and Hein Röhrig for many conversations that, above all, were fun.

A significant part of my Ph.D. career was also spent at the 'quantum schmantum' group of Artur Ekert at the University of Oxford. I thank him for showing me that an academic career should be enjoyable and that it is your own responsibility to make it so. During those two years I certainly had my fair share of good times, especially with the likes of Mike Mosca, Rasmus Hansen, Dik Bouwmeester, Vlatko Vedral (and Ivona), Lucien Hardy, Patrick Hayden, Ernesto Galvão, Jason Semitecolos, Simon Benjamin, Holly Cummins, Hitoshi Inamori and Leah Henderson.

With the risk of forgetting people, I also want to thank David Deutsch, Andrew Steane, Mike Nielsen, John Watrous, Chiara Macchiavello, Frédéric Magniez, Mauro D'Ariano, Miklos Santha, Umesh Vazirani, Sophie Laplante, André Berthiaume, Peter Høyer and Alain Tapp for discussions, advice and joint work.

Tenslotte wil ik mijn ouders, familie en vrienden bedanken voor al die redenen die nooit in een proefschrift worden beschreven.

The last line is of course reserved for Heather, without whom my life would be incomplete (it would be missing this now-finished Ph.D. thesis, for example).

Wim van Dam
San Francisco, September 2002

Chapter 1

Theory of Quantum Mechanics

This chapter contains a standard introduction to quantum information theory. Topics that will be discussed are: the Hilbert space formalism for quantum states, unitary transformations and the probability rules for measurement outcomes. Also the theory of mixed states, density matrices and completely positive operators is discussed.

1.1 Modeling Information

The term ‘bit’ stands for ‘binary digit’, which reflects the fact that it can be described and implemented by a two-level system. Conventionally, these two levels are indicated by the labels “zero” and “one”, or “0” and “1”. If we want to capture more than two possibilities, more bits are needed: with k bits we have 2^k different labels.

The abstraction from k two-level systems to the set $\{0,1\}^k$ of size 2^k takes us away from the physical details of the implementation of a piece of memory in a computer, and instead focuses on a more mathematical description of information. This ‘physics independent’ approach to standard information theory has been extremely successful in the past decades: it enables a general understanding of computational and communicational processes that is applicable to all the different ways of implementing these processes. It is for this reason that the Turing machine model of computation gives an accurate description of both the mechanical computer suggested by Charles Babbage and the latest Silicon based Pentium IV processors, despite their obvious physical differences. This does not mean that Turing’s model ignores the physical reality of building a computer, on the contrary. The observation that it would be unphysical to assume an infinite or unbounded precision in the components of a computer is expressed by Turing’s rule that per time-step only a fixed, finite amount of computational work can be done.[99] The proper analysis of algorithms in the theory of computational complexity relies critically on the exclusion of computational models that are not realistic. Such models often give the wrong impression that certain complicated tasks are easy. (A good example of this is the result that the factorization of integers can be done in polynomial time if we assume that addition, multiplication and division of arbitrary big numbers can be done in constant time. (See Chapter 4.5.4, Exercise 40 in [63] and [88].) There is, however, also a danger with this axiomatization of the physical assumptions in information theory: believing that the assumptions are true. This is what happened with the traditional view on information; forgotten were the implicit classical assumptions that ignore the possibilities of quantum mechanics. The realization that quantum physics describes a world where information behaves differently than in classical theory led to the blossoming of several fields—quantum information, quantum computing, quantum communication, et cetera. In this thesis we will focus on the differences in query complexity between classical and a quantum computation (Chapters 3–5), the possibility of 'self-testing' a quantum computer (Chapter 6) and a definition of quantum Kolmogorov complexity (Chapter 7). Before doing so, it is necessary to define what we mean by quantum information and computation.

1.2 Quantum Information

At the heart of quantum mechanical information theory lies the *superposition principle.* Where a classical bit is either in the state "zero" or "one", a quantum bit is allowed to be in a superposition of the two states. A qubit with the label q is therefore described in Dirac's bra-ket notation by the linear combination:

$$|q\rangle \;=\; \alpha|\text{“zero”}\rangle + \beta|\text{“one”}\rangle,$$

where for the complex valued amplitudes α, $\beta \in \mathbb{C}$, the normalization restriction $|\alpha|^2 + |\beta|^2 = 1$ applies. Here $|\alpha|$ denotes the *norm* of α: if $\alpha = a + b\mathrm{i}$, then $|\alpha| := \sqrt{a^2 + b^2}$. Alternatively we can write $|\alpha| := \sqrt{\alpha\alpha^*}$, where α^* is the *complex conjugate* $a - b\mathrm{i}$ of the complex value $\alpha = a + b\mathrm{i}$. In this formalism, the state space of a single qubit is built up by the unit vectors in the two-dimensional Hilbert space $\mathcal{H}_2$. For k qubits, there are 2^k basis states and hence the corresponding superposition is a linear combination of all 2^k possible strings of k bits:

$$|q_1 \cdots q_k\rangle \;=\; \sum_{i \in \{0,1\}^k} \alpha_i |i\rangle.$$

Again it is required that the amplitudes α_i obey the normalization condition: $\sum_i |\alpha_i|^2 = 1$. (In Section 1.4 we will see the reason behind this stipulation.) The state space of k qubits is the k-fold tensor product of the state space of a single qubit. This space is identical with a single 2^k-dimensional Hilbert space:

$$|q_1 \cdots q_k\rangle \;\in\; \mathcal{H}_2 \otimes \cdots \otimes \mathcal{H}_2 = \mathcal{H}_{2^k}.$$

For our purposes we will only use finite sets of quantum bits, so there is no need to look at infinite-dimensional Hilbert spaces.

1.3 Time Evolution of Quantum Bits

Quantum mechanics only allows transformations of states that are linear and respect the normalization restriction. When acting on an n-dimensional Hilbert space, these are the $n \times n$ complex valued rotation matrices that are norm preserving: the unitary matrices of $\mathsf{U}(n)$. It is easy to show that this corresponds exactly to the requirement that the inverse of U is the conjugate transpose U^* of the matrix.

The effect of a unitary transformation U on a state x is exactly described by the corresponding rotation of the vector $|x\rangle$ in the appropriate Hilbert space. For this reason, "U" stands both for the quantum mechanical transformation as well as for the unitary rotation:

$$|U(x)\rangle \;=\; U|x\rangle \;=\; U\left(\sum_i \alpha_i|i\rangle\right) \;=\; \sum_i \alpha_i U|i\rangle \;=\; \sum_i \alpha_i \sum_j U_{ji}|j\rangle,$$

where U_{ji} denotes the matrix element of U positioned at the j-th row and the i-th column. It follows from the associativity of matrix multiplication that the effect of two consecutive transformation U and W is the same as the single transformation $(W \cdot U)$. Just as matrix multiplication does not commute, so does the order of a sequence of unitary transformations matter: in general $WU \neq UW$. We can restate this in a more intuitive way by saying that it makes a difference if we first do U and then W, or the other way around. A typical example of this phenomenon is given by the matrices

$$W = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \quad \text{and} \quad U = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \tag{1.1}$$

with clearly $WU \neq UW$.

1.4 Measurements

When measuring the state $|x\rangle = \sum_i \alpha_i|i\rangle$, the probability of observing the outcome "i" equals $|\alpha_i|^2$. This explains the normalization restriction on the amplitudes: the different probabilities have to add up to one. But what exactly is a 'measurement' and an 'observation', and how do we describe this mathematically? These are thorny issues that this thesis will leave untouched. Here we will only give a formal description of the measurement process and a short explanation of why this is such a problematic part of quantum mechanics.

The possible outcomes "i" of x correspond to a set of orthogonal vectors $\{|m_i\rangle\}$ of the measuring device. This device can be our own eye or some kind of machine, but the crucial point is that 'measuring x' implies 'interacting with x'. The *effect on* x of such a measurement is that the state *collapses* according to the outcome "m_i" of our observation. This is described by the transformation:

$$\sum_i \alpha_i|i\rangle \quad \underset{\text{outcome } m_i}{\longmapsto} \quad |i\rangle. \tag{1.2}$$

The above described collapse is a non-unitary transformation. This is typical when we try to describe the behavior of x as it interacts with a system that lies outside of the state. (We say that x is an 'open system'.) When we view x and the measurement device *together* during the observation, the evolution becomes unitary again. Our current example is then described by the transformation:

$$\sum_i \alpha_i|i\rangle \otimes |\text{measurement device}\rangle \quad \longmapsto \quad \sum_i \alpha_i|i\rangle|\text{outcome } m_i\rangle.$$

The problem with this last description is that it no longer specifies the specific outcome "i" that we seem to observe. It is here where the debate on the *measurement problem* starts and our discussion ends.

For the purposes of this thesis it is more convenient to use the terminology of the collapsing quantum state. We will therefore describe the effect of a measurement as in Equation 1.2 for practical reasons. (This does not imply that the author really thinks that there is such a collapse, but these issues are outside the scope of this text. They concern the interpretation of quantum mechanics, which is irrelevant for the purposes of this thesis.)

We just described the traditional 'Von Neumann measurement' where we observe the state x in a canonical basis spanned by the basis vectors i. Other, more subtle, measurement procedures are also possible by choosing an in- or over-complete basis. We will postpone the description of these two options to the point when we discuss the density matrix formalism, which is more suitable for the general theory of interacting quantum mechanical systems.

1.5 Limitations of Dirac's Notation

The braket notation that we discussed above is tailor-made for the description of closed quantum mechanical systems. By this we mean the evolution of states that do not interact with an exterior environment. When we also want to consider the behavior of open systems, the ket-notation becomes less suitable. This was already obvious in the discussion of the measurement procedure where we had to expand the set of unitary operations with a probabilistic procedure that 'collapses' the quantum state to one of the basis states. One cannot help but feel uncomfortable about this sudden change of rules: is it not possible to deal with open and closed quantum systems in the same way? Luckily, we find in the formalism of density matrices a positive answer to this question.

1.6 Density Matrices

An n-dimensional pure state x can be expressed as a normalized vector $|x\rangle$ in the Hilbert space $\mathcal{H}_n$. The complex conjugate $|x\rangle^*$ of this vector is the bra $\langle x|$, which is an element of the adjoint space $\mathcal{H}_n^*$. By taking the direct product between the ket $|x\rangle$ and the bra $\langle x|$, we thus obtain an $n \times n$ complex valued, Hermitian matrix: the *density matrix* of x.

As an example, for the state $|x\rangle = \sum_i \alpha_i|i\rangle$, the density matrix is:

$$|x\rangle\langle x| \;=\; \left(\sum_i \alpha_i|i\rangle\right)\left(\sum_j \alpha_j^*\langle j|\right) \;=\; \sum_{i,j} \alpha_i\alpha_j^*|i\rangle\langle j|.$$

In the case of a single qubit with the ket description $|q\rangle = \alpha|0\rangle + \beta|1\rangle$, this leads to the 2×2 matrix in the standard basis

$$|q\rangle\langle q| \;=\; \begin{bmatrix} |\alpha|^2 & \alpha\beta^* \\ \alpha^*\beta & |\beta|^2 \end{bmatrix}.$$

From now on, the density matrix of the state x will be denoted by the same symbol x, and the fact that a matrix is a density matrix will be indicated by its square brackets.

The great advantage of this formalism is that it also allows the description of an *ensemble* of pure quantum states. If we have such a state ρ, which is a probabilistic mixture of the pure states $|x_t\rangle$ with probabilities p_t, then the matrix ρ is the weighted linear combination of the corresponding pure states matrices,

$$\rho \;=\; \sum_t p_t \cdot |x_t\rangle\langle x_t|,$$

with $p_t \geq 0$ and $\sum_t p_t = 1$.

Every density matrix that can be written as such a convex combination of pure states is a legal, or 'allowed', state, where allowed means: "allowed by the laws of quantum physics". It follows from linear algebra that this restriction coincides with the requirement that the matrix is a Hermitian, positive semidefinite matrix with unit trace.

The *spectral decomposition* of a proper density matrix ρ is done in terms of its eigenvalues λ_t and eigenvectors $|\omega_t\rangle$, by the equality

$$\rho \;=\; \sum_t \lambda_t|\omega_t\rangle\langle\omega_t|. \tag{1.3}$$

This shows that we can interpret ρ as the mixture $\{(\lambda_t, |\omega_t\rangle)\}_t$, where the states ω_t are pure and mutually orthogonal.

The above decomposition gives a convenient way of assigning a mixture to a given density matrix. It is important to realize, however, that a density matrix corresponds to a whole family of possible mixtures. Take, for example, the ensembles $\{(\frac{1}{2}, |0\rangle), (\frac{1}{2}, |1\rangle)\}$ and $\{(\frac{1}{2}, \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)), (\frac{1}{2}, \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle))\}$, which have the same density matrix:

$$\begin{aligned} \frac{1}{2}\begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} + \frac{1}{2}\begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} \;&=\; \begin{bmatrix} \frac{1}{2} & 0 \\ 0 & \frac{1}{2} \end{bmatrix} \\ &=\; \frac{1}{2}\begin{bmatrix} \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} \end{bmatrix} + \frac{1}{2}\begin{bmatrix} +\frac{1}{2} & -\frac{1}{2} \\ -\frac{1}{2} & +\frac{1}{2} \end{bmatrix}. \end{aligned}$$

We shall see that this implies that these two mixtures are indistinguishable from each other; it is therefore more accurate and less confusing to consider them as equivalent mixtures.

The density matrix of a *qubit* ρ in the standard basis is always of the form

$$\rho(p,\alpha) \quad = \quad \begin{bmatrix} p & \alpha^* \\ \alpha & 1-p \end{bmatrix},$$

with the probability p between 0 and 1 and the 'off-diagonal term' $|\alpha|^2 \leq p(1-p)$. If $|\alpha|^2 = p(1-p)$ then ρ is a pure state with $|\rho\rangle = \sqrt{p}|0\rangle + \frac{\alpha}{\sqrt{p}}|1\rangle$ (or $|\rho\rangle = |1\rangle$ if $p = 0$); otherwise the qubit ρ corresponds to a mixture.

1.7 Separated Systems

We need the formalism of density matrices to be able to describe the evolution of an open system. By 'open' we mean that there is a possible interaction between the quantum mechanical state and its environment. An example of such a situation was already mentioned when we saw how a qubit changed into a probabilistic mixture after it interacted with a measurement device outside the qubit system. An important operation in this context is the 'tracing out' operation that describes how we can ignore a part of a quantum system.

Definition 1 (Partial trace) *Let $\mathcal{H}_{AB}$ be the combination of the two Hilbert spaces $\mathcal{H}_A$ and $\mathcal{H}_B$, with the respective bases $\{|a_i\rangle\}$ and $\{|b_j\rangle\}$. The* partial trace tr_B *of a state ρ in $\mathcal{H}_{AB}$ is defined by*

$$\mathrm{tr}_B(\rho) \quad := \quad \sum_j \langle b_j|\rho^{AB}|b_j\rangle,$$

where $\langle x|\rho|y\rangle$ expresses the inner product of the row vector $\langle x|$, the matrix ρ and the column vector $|y\rangle$.

When we are dealing with a general state ρ and we want to describe its content for the subsystem $\mathcal{H}_A$, we indicate this by the notation "ρ^A" Hence in terms of the above definition we would write $\rho^A := \mathrm{tr}_B(\rho^{AB})$. Conversely, we also have $\rho^B := \mathrm{tr}_A(\rho^{AB})$.

1.8 Von Neumann Entropy and Fidelity

The eigenvalues λ_i of a density matrix are always nonnegative and sum up to one. If we decompose a mixture into a linear combination of orthogonal pure states, then the λ's will correspond to the probabilities of the respective eigenvectors. (See Equation 1.3.) Although the eigenvectors of a density matrix are not always unique, its eigenvalues are. This allows us to unambiguously define the *Von Neumann entropy* $S(\rho)$ of a state, which reflects how ‘mixed’ or random ρ is. As a result, pure states will have zero entropy.

Definition 2 (Von Neumann entropy) *The Von Neumann entropy of a mixed state ρ is defined as*

$$S(\rho) \;=\; S\left(\sum_i p_i|\phi_i\rangle\langle\phi_i|\right) \;:=\; -\sum_i p_i \log p_i,$$

where $\sum_i p_i|\phi_i\rangle\langle\phi_i|$ is a spectral decomposition of ρ in its eigenvectors.

If we understand the logarithm of the matrix ρ to be the standard Taylor expansion: $(\rho - I) - \frac{1}{2}(\rho - I)^2 + \frac{1}{3}(\rho - I)^3 - \cdots$, then the above definition can also be written as $S(\rho) := -\mathrm{tr}(\rho \log_2 \rho)$. It should be clear that the Von Neumann entropy equals the Shannon entropy of the eigenvalues of the density matrix ρ.

A source $\mathcal{E} = \{(\rho_i, p_i)\}$ has an associated Von Neumann entropy $S(\rho)$ of the average state $\rho = \sum_i p_i\rho_i$. Schumacher’s noiseless coding theorem [83] shows how to obtain an encoding with average letter-length $S(\rho)$ for a source of pure states, where the fidelity of the encoding goes to 1 as the number of letters emitted by the source goes to infinity. (A survey can be found in Preskill’s lecture notes [78, page 190], Nielsen’s thesis [73, Chapter 7], or the standard book by Nielsen and Chuang [74].)

How close two mixed states ρ and σ are, can be expressed by the fidelity between the two density matrices. This notion generalizes the inner product between two Hilbert space vectors for pure states. The matrix ρ represents a pure state if and only if $\rho^2 = \rho$, in which case we can also say $\sqrt{\rho} = \rho$. In general, the *square root* of a mixed state is defined by

$$\sqrt{\rho} \;=\; \sqrt{\sum_i p_i|\phi_i\rangle\langle\phi_i|} \;:=\; \sum_i \sqrt{p_i}|\phi_i\rangle\langle\phi_i|.$$

We will use this root in the following definition.

Definition 3 (Fidelity) *The fidelity* F*(,b)etween two density matrices ρ and σ is defined by*

$$\mathrm{F}(\rho,\sigma) \;:=\; \mathrm{tr}\left(\sqrt{\sqrt{\rho}\cdot\sigma\cdot\sqrt{\rho}}\right). \tag{1.4}$$

For pure states ϕ and ψ, the above definition coincides again with the familiar $|\langle\phi|\psi\rangle|$ (although some authors use the square of this value). If $\mathrm{F}(\rho,\sigma) = 1$, then $\rho = \sigma$, and vice versa.

1.9 Operations on Mixed States

A unitary transformation U maps the pure state $|x\rangle$ to the new pure state $U|x\rangle$. The latter can be written as the density matrix $U|x\rangle\langle x|U^*$. In the language of density matrices, the corresponding transformation $\boldsymbol{U}$ is therefore calculated by 'sandwiching' the matrix x between U and its conjugate U^*:

$$\boldsymbol{U}\left(|x\rangle\langle x|\right) \;:=\; U|x\rangle\langle x|U^*.$$

If we have a mixed state ρ, then $\boldsymbol{U}$ acts linearly on the eigenvectors of ρ. The following equation shows us that this calculation can be done without having to decompose ρ, and that our sandwich expression therefore also holds for mixed states:

$$\begin{aligned}
\boldsymbol{U}\left(\rho\right) &= \boldsymbol{U}\left(\sum_t \lambda_t|\omega_t\rangle\langle\omega_t|\right) \\
&= \sum_t \lambda_t \cdot \boldsymbol{U}\left(|\omega_t\rangle\langle\omega_t|\right) \\
&:= \sum_t \lambda_t \cdot U|\omega_t\rangle\langle\omega_t|U^* \\
&= U\left(\sum_t \lambda_t \cdot |\omega_t\rangle\langle\omega_t|\right)U^* \\
&= U\cdot\rho\cdot U^*.
\end{aligned}$$

It is clear that the positive eigenvalues λ_t of ρ remain unchanged, and that $\boldsymbol{U}$ only rotates the eigenvectors $|\omega_t\rangle$ to the new eigenstates $U|\omega_t\rangle$.

Unitary operations are an example of completely-positive, trace preserving maps: every positive semidefinite matrix is mapped to (another) positive semidefinite matrix, and the trace of the matrix remains unaltered. Complete-positivity, in combination with the preservation of the trace, assures us that the result of a transformation will be a proper state if we started with a proper one.

Besides the unitary functions, there are other transformations that are possible in quantum mechanics. Just as mixed states are composed of pure states, so can a positive map be a linear combination of matrix multiplications similar to the ones we discussed above. An example of such a non-unitary mapping is the mapping $\boldsymbol{P}$, corresponding to a measurement of a qubit in the standard basis $\{0,1\}$. This function consists of two 'projectors' $P_0 = |0\rangle\langle 0|$ and $P_1 = |1\rangle\langle 1|$ that transform a qubit ρ into a probabilistic mixture of the states 0 and 1. Explicitly:

$$\begin{aligned}
\boldsymbol{P}(\rho) &= \boldsymbol{P}\left(\begin{bmatrix} p & \alpha^* \\ \alpha & 1-p \end{bmatrix}\right) \\
&= \boldsymbol{P_0}\left(\begin{bmatrix} p & \alpha^* \\ \alpha & 1-p \end{bmatrix}\right) + \boldsymbol{P_1}\left(\begin{bmatrix} p & \alpha^* \\ \alpha & 1-p \end{bmatrix}\right) \\
&= \begin{bmatrix} p & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 1-p \end{bmatrix} \\
&= \begin{bmatrix} p & 0 \\ 0 & 1-p \end{bmatrix}.
\end{aligned}$$

We see that the eigenvalues of the new density matrix are p and $1-p$ with the corresponding eigenvectors $|0\rangle\langle 0|$ and $|1\rangle\langle 1|$. In general, the eigenvalues of ρ will change under this transformation and hence there is no unitary operation that can establish the above mapping. In the next section we will give a formal description of all transformations, such as the above $\boldsymbol{P}$, that are allowed by quantum physics.

1.10 Operator Sum Representation

The following requirements for an operator $\boldsymbol{E}$ are necessary and sufficient for $\boldsymbol{E}$ to be a proper quantum mechanical transformation:

1. The mapping $\boldsymbol{E}$ can be written as a set of matrices $\{E_i\}_i$ with which it maps a state ρ to the linear combination $\sum_i E_i \cdot \rho \cdot E_i^*$.

2. The set of operators $\{E_i\}$ has to obey the identity restriction $\sum_k E_i^* \cdot E_i = \mathrm{I}$. (Note the change of order of E and E^* in the multiplication.)

These two requirements exactly describe the set of *completely-positive, trace preserving maps.* Complete-positivity means that we require both $\boldsymbol{E}$ as well its trivial extensions $\boldsymbol{E} \otimes \mathbf{I}$ to higher dimensions to be positive. This is a stronger condition than positivity. An example of a positive but not completely-positive map is the partial transpose $\boldsymbol{T}$, which is defined by $\boldsymbol{T}(\rho) = \rho^T$.

We have properly extended the set of unitary transformations and measurements by the above 'operator sum' formalism. An example of this is the mapping that erases a qubit and replaces it with the value zero. This non-unitary function is the combination of two operators

$$\boldsymbol{E} = \left\{\begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}\right\},$$

and has the same effect on every qubit ρ, namely

$$\begin{aligned}
\boldsymbol{E}(\rho) &= \boldsymbol{E}\left(\begin{bmatrix} p & \alpha^* \\ \alpha & 1-p \end{bmatrix}\right) \\
&= \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}\begin{bmatrix} p & \alpha^* \\ \alpha & 1-p \end{bmatrix}\begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} + \\
&\quad \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}\begin{bmatrix} p & \alpha^* \\ \alpha & 1-p \end{bmatrix}\begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix} \\
&= \begin{bmatrix} p & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} 1-p & 0 \\ 0 & 0 \end{bmatrix} \\
&= |0\rangle\langle 0|.
\end{aligned}$$

We previously argued that a measurement has a non-unitary effect on a state because we ignored its interaction with an outside system (the measurement device). This lesson holds for all allowed transformations:

> *Every completely-positive, trace preserving transformation $\boldsymbol{E}$ of a system $\mathcal{H}_A$ can be viewed as a part of unitary mapping $\boldsymbol{U}_E$ on a bigger system $\mathcal{H}_A \otimes \mathcal{H}_B$. That $\boldsymbol{E}$* by itself *appears to be non-unitary is due to the fact that we ignore the space $\mathcal{H}_B$.*

It can be shown that for the extension of the system it is sufficient to assume that the dimension of the appended space $\mathcal{H}_B$ is twice as large as that of $\mathcal{H}_A$, and that its initial state is $|0\cdots 0\rangle$. Hence, for every allowed quantum mechanical transformation $\boldsymbol{E}$ that acts on an n-dimensional system, there exists a unitary matrix $U_E \in \mathsf{U}(n^3)$ such that

$$\boldsymbol{E}(x) \;=\; \mathrm{tr}_B\left[U_E(x \otimes |0^B\cdots 0^B\rangle\langle 0^B\cdots 0^B|)U_E^*\right]$$

for all x. This is, in more general terms, the difference that we encountered between the Equations 1.2 and 1.3. The non-unitary 'collapse' associated with an observation, or any other kind of interaction, is again a unitary transformation when we incorporate the measurement device into the description of the event.

The converse of the earlier statement also holds: every mapping that can be written as a traced-out, unitary transformation on a larger Hilbert space is a completely-positive, trace preserving mapping.

In the literature on quantum information theory the linear functions on density matrices are sometimes called 'super operators'. We thus have the following definition.

Definition 4 (Completely positive super operator/CPSO) *A transformation $\boldsymbol{E}$ is a* completely positive super operator, *or* CPSO, *if only if $\boldsymbol{E}$ is linear, trace-preserving, and completely positive.*

The reader is referred to the standard book by Asher Peres[77] or the article by Benjamin Schumacher[84] for a more extended and rigorous treatment of this 'operator sum representation'.

Chapter 2

Quantum Information and Computation

In the previous chapter we described the foundations of quantum information and the quantum mechanical transformations that are possible with it. The central idea of computational complexity theory is to assign different 'costs' to different operations. Typically, a fixed set of elementary operations is used to construct all other transformations. The computational cost is then expressed as the minimal number of elementary operations that is necessary to establish the desired transformation.

2.1 Some Elementary Operations

In quantum computing and communication we look at the possibilities of transforming information as is allowed by the laws of quantum mechanics. We usually decompose such quantum algorithms in a series of small elementary steps that consist of one and two qubit operations. The following elementary unitary gates will be used throughout the rest of the thesis.

Definition 5 (Some elementary quantum gates) The Not gate: *This is the gate that we know in classical computation with the additional characteristic that it respects the superposition of a qubit:*

$$\mathrm{Not}(\alpha|0\rangle + \beta|1\rangle) \quad = \quad \beta|0\rangle + \alpha|1\rangle.$$

Phase Flip: *The* Flip *gate changes the phase of a qubit conditional on its value:*

$$\mathrm{Flip}(\alpha|0\rangle + \beta|1\rangle) \quad = \quad \alpha|0\rangle - \beta|1\rangle.$$

Phase Rotation: *A more general phase rotation is provided by the* Phase *operation, which has a free parameter* ϕ *that determines the angle of the phase change:*

$$\mathrm{Phase}_\phi(\alpha|0\rangle + \beta|1\rangle) \quad = \quad \alpha|0\rangle + \mathrm{e}^{\mathrm{i}\phi}\beta|1\rangle.$$

(Note: $\mathrm{Flip} = \mathrm{Phase}_\pi$*.)*

Hadamard transform: *This transformation* H *maps the zero and one state to the following superpositions of the two basis states:*

$$\mathrm{H}|0\rangle \;=\; \tfrac{1}{\sqrt{2}}(|0\rangle+|1\rangle) \quad \textit{and} \quad \mathrm{H}|1\rangle \;=\; \tfrac{1}{\sqrt{2}}(|0\rangle-|1\rangle).$$

The Hadamard is its own inverse ($\mathrm{H}^2=\mathrm{I}$*).*

General Rotation: *The general rotation* R *with angles* α,θ,ϕ *is the unitary one qubit transformation with eigenvectors* $|\psi\rangle = \cos(\frac{\theta}{2})|0\rangle + \mathrm{e}^{\mathrm{i}\phi}\sin(\frac{\theta}{2})|1\rangle$ *and* $|\psi^\perp\rangle = \sin(\frac{\theta}{2})|0\rangle - \mathrm{e}^{\mathrm{i}\phi}\cos(\frac{\theta}{2})|1\rangle$*. The corresponding eigenvalues are indicated by the equalities*

$$\mathrm{R}_{\alpha,\theta,\phi}|\psi\rangle = |\psi\rangle \quad \textit{and} \quad \mathrm{R}_{\alpha,\theta,\phi}|\psi^\perp\rangle = \mathrm{e}^{\mathrm{i}\alpha}|\psi^\perp\rangle,$$

and are therefore 1 *and* $\mathrm{e}^{\mathrm{i}\alpha}$*.*

Controlled-Not: *The controlled-not is a two-qubit operation that applies the* Not *gate to the target bit if the control bit equals "*1*"; otherwise it leaves the target unchanged:*

$$\mathrm{CNot}|x,y\rangle \;=\; |x,y\oplus x\rangle,$$

for all $x,y\in\{0,1\}$*.*

Controlled-Flip: *The controlled-flip is, like the* CNot*, a two-qubit operation. It applies the* Flip *gate if both bits equals "*1*"; otherwise it leaves the state unchanged:*

$$\mathrm{CFlip}|x,y\rangle \;=\; (-1)^{xy}|x,y\rangle,$$

for all $x,y\in\{0,1\}$*.*

2.2 Fault Tolerant and Universal Quantum Gates

It has been shown that there exists finite sets of quantum gates that are universal in the following sense. Consider the networks that can be constructed from a countable set of gates $\{G_1,G_2,\dots\}$. Each network will implement a unitary transformation, and we want to consider if any finite-dimensional unitary transformation can be implemented in such a way. Clearly, because the set of networks is countable, we cannot hope that we can construct every element of $\mathsf{U}(n)$ exactly. Hence, we will have to aim for the approximation (within an arbitrary small error) of every such element. It has been proven that there are indeed universal sets of quantum gates with which this can be achieved, and these sets can be remarkable simple. The following collection was described in [26] and has the additional useful feature that the gates are 'fault-tolerant' [74].

Fact 1 (Universal, Fault Tolerant Sets of Quantum Gates [26]) *With the Hadamard gate* H*, the controlled-not* CNot *and the* $\frac{\pi}{4}$ *phase gate* $\mathrm{R}_{\pi/4,0,0}$ *any other unitary transformation can be approximated within an arbitrary small error (with respect to some distance measure on the set of operators). Also, these three gates can be implemented in a fault-tolerant way.*

2.3 Quantum versus Classical Query Complexity

The theory of quantum computation investigates if, and if so, how, we can use quantum mechanical effects to solve computational problems more efficiently than we can do by classical means. So far, the strongest indication that there maybe such a difference in computational power between quantum and classical computing is provided by Peter Shor's factoring algorithm[91]. Unfortunately, the result by Shor does not prove that there is a superpolynomial separation between the two models of computation. This is because the classical time complexity of factoring and discrete logarithms is still unknown, despite more than two thousand years of effort, starting with Eratosthenes's sieve in 300 B.C.

A complexity measure for which we do have rigorous results is provided by the the black-box, or oracle, model of computation. The algorithms of Deutsch [38], Deutsch & Jozsa [39], Berthiaume & Brassard [23], Bernstein & Vazirani [22], Simon [92], Grover [48], and Buhrman & van Dam [28] give examples of problems for which we have a quantum reduction in the query complexity of a problem, whereas the lower bounds of Jozsa [59], Bennett *et al.* [19], and Beals *et al.* [10] show that there are also limits to the advantage that quantum computation can give us. The general picture that has emerged from these results is that we can only expect a superpolynomial difference between classical and quantum computation if we can use the specific structure of the problem that we try to solve. The promise on the function of Simon's problem is a typical example of such a structure that establishes an exponential quantum improvement over the classical complexity.[92] To find more structured problems that allow such a gain is one of the quests for researchers in quantum complexity theory.

Consider a problem that is defined in terms of n (unknown) values $f(1),\ldots,f(n)$. The *(probabilistic) query complexity* of such a problem is the minimum number of times that an algorithm has to 'consult' the string $f(1),\ldots,f(n)$ to solve the problem (with high probability). A typical example of this setting is the calculation of the OR of n bit values: the question whether there is an index i with $f(i)=1$. The classical query complexity of this task is n, whereas in the quantum setting we only need $O(\sqrt{n})$calls to f to solve the problem. We therefore say that we have a 'quadratic' separation between the classical and the quantum query complexity of the OR function. The question is which tasks allow a quantum reduction in the query complexity, and if so, how much.

The reason why quantum algorithms sometimes require less queries starts with the *superposition principle* of quantum mechanics. A single call "i" to the function f

establishes the evolution $|i\rangle|b\rangle \mapsto |i\rangle|f(i) \oplus b\rangle$ (where $\oplus$ denotes addition modulo two), which in classical computation is the best we can expect from an f-query. But by the rules of quantum mechanics, we can also consult f in superposition. Hence, with a single call we can create a state that depends on several values $f(i)$:

$$\sum_i |i\rangle \otimes (\alpha_i|0\rangle + \beta_i|1\rangle) \quad \underset{\text{one } f\text{-query}}{\longmapsto} \quad \sum_i |i\rangle \otimes (\alpha_i|f(i)\rangle + \beta_i|f(i) \oplus 1\rangle).$$

It is this ‘parallelism’ in combination with the quantum mechanical phenomenon of *interference* that allows us to solve some problems more efficiently than is possible with classical protocols.

2.4 Earlier Results in Quantum Computing

This thesis uses, and builds on, a combination of earlier results in quantum computation. We are especially concerned with the query complexity of procedures that prepare a state that depends on a black-box function. For example, how often do we have to read out the bit values $f(i)$ if we want to create the state $\sum_i (-1)^{f(i)}\alpha_i|i\rangle$? The following fact shows us that this can be done with the minimum of a single query.

Fact 2 (Phase-kick-back trick [31]) *If we can query the function f in quantum mechanical fashion as follows:*

$$|i\rangle \otimes |b\rangle \quad \longmapsto \quad |i\rangle \otimes |b \oplus f(i)\rangle$$

with $f(i), b \in \{0, 1\}$, then the phase changing transition

$$\sum_i \alpha_i|i\rangle \quad \longmapsto \quad \sum_i (-1)^{f(i)}\alpha_i|i\rangle$$

can be established with only one call to the unknown bit values of f.

Proof: First, we append to the superposition of $|i\rangle$ states the qubit $\frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$. Then, in superposition, we add (modulo two) the function value $f(i)$ to this bit. For a specific value of i, this yields the evolution

$$|i\rangle \otimes \tfrac{1}{\sqrt{2}}(|0\rangle - |1\rangle) \quad \longmapsto \quad |i\rangle \otimes \tfrac{1}{\sqrt{2}}(|0 \oplus f(i)\rangle - |1 \oplus f(i)\rangle) \tag{2.1}$$

$$= \quad \begin{cases} +|i\rangle \otimes \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) & \text{if } f(i) = 0 \\ -|i\rangle \otimes \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) & \text{if } f(i) = 1 \end{cases} \tag{2.2}$$

Hence, by the superposition principle, this gives the desired evolution with only one query to the function f. □

Using this fact, we can easily prove the following core result.

Fact 3 (Single Query Parity Trick[31, 38]) *Let $f : \{0,1\} \to \{0,1\}$. There exists a deterministic quantum algorithm that computes the parity bit $f(0) \oplus f(1)$ with one query to the function f. This algorithm works in constant time.*

Proof: Construct the following initial state:

$$|\text{Initial}\rangle \quad = \quad \tfrac{1}{2}(|0\rangle + |1\rangle) \otimes (|0\rangle - |1\rangle). \tag{2.3}$$

Next, we add (modulo two) the bit values $f(i)$ to the rightmost bit, where the index $i \in \{0,1\}$ is described by the first bit of the initial state. Note that by the superposition of this rightmost bit, both values $f(0)$ and $f(1)$ are also queried in superposition. Applying f in such a way establishes the following evolution on the two qubits:

$$|i\rangle \otimes |b\rangle \quad \longmapsto \quad |i\rangle \otimes |b \oplus f(i)\rangle,$$

for $b \in \{0,1\}$. This results in the following outcome when applied to the initial state mentioned in the beginning of the proof:

$$\begin{array}{rll} \frac{1}{2}(|0\rangle + |1\rangle) \otimes (|0\rangle - |1\rangle) & \text{if} & f(0) = f(1) = 0 \\ \frac{1}{2}(|0\rangle - |1\rangle) \otimes (|0\rangle - |1\rangle) & \text{if} & f(0) = 0, f(1) = 1 \\ -\frac{1}{2}(|0\rangle - |1\rangle) \otimes (|0\rangle - |1\rangle) & \text{if} & f(0) = 1, f(1) = 0 \\ -\frac{1}{2}(|0\rangle + |1\rangle) \otimes (|0\rangle - |1\rangle) & \text{if} & f(0) = f(1) = 1. \end{array}$$

Hence, if we apply a Hadamard transformation to the first register, we obtain

$$|\text{Final}\rangle \quad = \quad (-1)^{f(0)}|f(0) \oplus f(1)\rangle \otimes (|0\rangle - |1\rangle).$$

Observing the first bit of this final state yields the correct answer $f(0) \oplus f(1)$ without error. □

2.5 More Classic Quantum Results

In 1993 Bernstein & Vazirani gave the following example of a family of functions that are more easily distinguished with quantum queries than with classical ones.

Fact 4 (Bernstein & Vazirani's inner-product problem [22, 31]) *Let the black-box function $g_s : \{0,1\}^n \to \{0,1\}$ be defined by*

$$g_s(x) \quad = \quad (x,s) \quad = \quad \sum_{i=1}^{n} s_i x_i \bmod 2, \tag{2.4}$$

where $s = s_1 \dots s_n \in \{0,1\}^n$ is an unknown n-bit mask. A quantum computer can determine the value s with one call to the function g_s, whereas any probabilistic, classical algorithm needs at least n queries to g_s to perform the same task.

Proof: (See [22] for the original proof, and [31] for the single query version of it.) First, initialize the $(n+1)$-qubit register

$$|\mathrm{start}\rangle \;=\; \tfrac{1}{\sqrt{2^n}} \sum_{x\in\{0,1\}^n} |x\rangle \otimes \tfrac{1}{\sqrt{2}}(|0\rangle - |1\rangle).$$

By XOR-ing the rightmost bit with the function value $g_s(x)$ (*cf.* Fact 2), we obtain the state

$$\tfrac{1}{\sqrt{2^n}} \sum_{x\in\{0,1\}^n} (-1)^{(s,x)}|x\rangle \otimes \tfrac{1}{\sqrt{2}}(|0\rangle - |1\rangle), \tag{2.5}$$

with only one g_s-call. The bit string s is then easily obtained with an n-fold Hadamard transform on the first n bits:

$$\tfrac{1}{\sqrt{2^n}} \sum_{x\in\{0,1\}^n} (-1)^{(s,x)}|x\rangle \quad \underset{\mathrm{H}^{\otimes n}}{\longmapsto} \quad |s\rangle, \tag{2.6}$$

which concludes the quantum algorithm.

For the classical lower bound we observe that every traditional query will only give (maximally) one bit of information about the n bits of s. □

The above result uses the unitarity of $\mathrm{H}^{\otimes n}$ and its connection with the inner-product function. In Chapter 5 we will derive a similar result for a different family of unitary matrices and the Legendre function that it uses.

Because the Hadamard is its own inverse we have, in fact, the following 'bi-directional statement' about this transform

$$\frac{1}{\sqrt{2^n}} \sum_{x\in\{0,1\}^n} (-1)^{(x,s)}|x\rangle \;\longleftarrow \mathrm{H}^{\otimes n} \longrightarrow\; |s_1 s_2 \cdots s_n\rangle. \tag{2.7}$$

The above leads to the observation that if we want to know the string $s_1 \cdots s_n$, it is sufficient to have a superposition with phase values of the form $(-1)^{(x,s)}$, for every $x \in \{0,1\}^n$. This is a well-known result in quantum computation and has been used several times to underline the differences between quantum and classical information processing.[22, 31, 49, 97]

Another key result in quantum computation is the square-root speed-up that one can obtain when querying a database for a specific element.

Fact 5 (Grover's search algorithm [48]) *Let $f(1),\ldots,f(n)$ be a string of $n-1$ zeros and one entry $f(s) = 1$. With a quantum computer the unknown value s can be determined exactly with only $\lceil \frac{\pi}{4}\sqrt{n}\rceil$ queries to the function f.*

Proof: See the original article by Lov Grover[48], or better yet, the excellent analysis of it by Boyer *et al.*[25] □

2.6 Notation

We use $x, y, \ldots$ to denote finite, classical Boolean strings. When we write $|x\rangle$, we mean the quantum state vector in the standard basis that corresponds to the classical string x. In general we use $\phi, \psi, \ldots$ to denote pure quantum states. Mixed states are represented by the letters ρ, σ et cetera. We also use uppercase letters $X, Y, \ldots$ for (mixed) quantum states that are strings of qubits. The terms quantum state, qubit string, and quantum register are used interchangeably (sometimes to emphasize the purpose of the quantum state at hand). Lower-case letters i, j, k, l, m, n denote integer indices or string lengths.

For classical strings over the alphabet $\{0, 1\}$, $\ell(x)$ denotes the length of the string. For finite sets A, $|A|$ denotes the cardinality of the set. Concatenation of x, y is written as the juxtaposition xy, and the n-fold concatenation of x is written x^n.

For Hilbert spaces, we write $\mathcal{H}_d$ for the d-dimensional Hilbert space and $\mathcal{H}^m$ for the m-fold tensor product space $\mathcal{H} \otimes \cdots \otimes \mathcal{H}$. A pure quantum state ϕ represented as a vector in such a Hilbert space is denoted by the ket $|\phi\rangle$.

We slightly abuse notation by sometimes letting the state symbols $\phi, \rho, \ldots$ also stand for the corresponding density matrices. Hence, a pure state ϕ as a Hilbert space vector is denoted by $|\phi\rangle$, whereas its density matrix $|\phi\rangle\langle\phi|$ can also be indicated by ϕ.

An ensemble $\mathcal{E}$ is a specific distribution $p_1, p_2, \ldots$ over a set of (mixed) states $\rho_1, \rho_2, \ldots$. We denote this by $\mathcal{E} = \{(\rho_i, p_i)\}$. The average state of such an ensemble $\mathcal{E}$ is $\rho = \sum_i p_i \rho_i$. An average state corresponds to several different ensembles. When an ensemble is used to produce a sequence of states ρ_i according to the probabilities p_i, we speak of a *source* $\mathcal{E}$.

The length of a quantum state is denoted by $\ell(X)$, by which we mean the smallest ℓ for which X sits in the 2^ℓ-dimensional Hilbert space (in the standard basis).

A transformation $\boldsymbol{S}$ on the space of density matrices is allowed by the laws of quantum mechanics if and if only it is a completely positive, trace preserving mapping.

Throughout this thesis, results that were already known are indicated as 'facts'.

Chapter 3

Quantum Oracle Interrogation

In this chapter we discuss the quantum query complexity of the 'oracle interrogation' problem: For a black-box function $z : \{1,\ldots,n\} \to \{0,1\}$, how many queries are necessary to recover (with high probability) the n unknown bits $z_1 \cdots z_n$? First, we will describe a *quantum interrogation algorithm* that — with high probability— obtains the n bits using only $\frac{n}{2} + \sqrt{n}$ black box queries. Next, an 'approximating version' of interrogation is discussed. It is shown how with k black box queries one can produce an approximation of z that gets $\frac{n}{2} + \sqrt{k(n-k)}$ bits (expected) of $z_1 \cdots z_n$ correct.

3.1 Introduction

Consider a quantum computer in combination with a black-box function z that describes an n bit string $z_1 \cdots z_n$. We will show how $\frac{n}{2} + \sqrt{n}$ calls to the oracle are sufficient to guess the whole content of the oracle (being an n bit string) with probability greater than 95%. This contrasts the power of classical computers, which require n calls to achieve the same task. From this result it follows that any function with the n bits of z as input, can be calculated using $\frac{n}{2} + \sqrt{n}$ queries to z provided that we allow a small probability of error. It is also shown that an error probability ε can be established by $\frac{n}{2} + O(\log(\frac{1}{\varepsilon}))\sqrt{n}$ oracle queries.

In the second part of the chapter, 'approximate interrogation' is discussed. This is when only a certain fraction of the n bits of z are requested. Also for this scenario does the quantum algorithm outperform the classical protocols. An example is given where a quantum procedure with $\frac{n}{10}$ queries returns a string of which 80% of the bits are correct. Any classical protocol would need $\frac{3n}{5}$ queries to establish such a correctness ratio.

3.2 Known Quantum Query Complexity Bounds

Various articles [10, 42, 72] have determined several lower bounds on the capability of quantum computers to outperform classical computers in the black-box setting. These bounds refer to the required amount of queries to a black-box or oracle (with a domain size n) in order to decide some general property of this black-box. For example, if we want to know (with bounded error) the parity of the n values, then it is still necessary for a quantum computer to call the black-box $\frac{n}{2}$ times[10, 72]. It has also been shown that for the *exact* calculation of certain functions (the bitwise OR for example) all n calls are required[10].

Here, we present an *upper bound* on the number of black-box queries that is sufficient to compute any function over the n bits provided that we allow a small probability of error. More specifically, it will be shown that for every unknown black-box, there is a potential speed-up of almost a factor of two if we want to know everything there is to know about the oracle function. By this the following is meant. If the domain of the oracle has size n, a classical computer will have to apply n calls in order to know all n bits describing the oracle. Below, it will be proven that a quantum computer can perform the same task with high probability using only $\frac{n}{2}+\sqrt{n}$ queries. From this result it immediately follows that *any* function F on the domain $\{0,1\}^n$ can be calculated with a small two-sided error using only $\frac{n}{2}+\sqrt{n}$ calls.

The factor-of-two gain can be increased by going to approximating interrogation procedures. If we do not longer require to know *all* of the n bits but are instead already satisfied with a certain percentage of correct bits, then the difference between classical and quantum computation becomes bigger. An example of this occurs when we want to guess the string such that we can expect 80% of the bits to be correct. A quantum computer can do this with one-sixth of the queries that a classical computer requires: $\frac{n}{10}$ quantum calls versus $\frac{3n}{5}$ classical calls. This also illustrates that the procedure described here is not a ‘superdense coding-in-disguise’, which would allow a reduction by only a factor of two[21].

3.3 Definition of the Interrogation Problem

The setting for this chapter is as follows. We try to investigate the potential differences between a quantum and a classical computer when both cases are confronted with an oracle z. The only thing known in advance about this z is that it is a binary-valued function with a domain of size n. We will view this oracle $z:\{1,\dots,n\}\rightarrow\{0,1\}$ as the n-bit string it defines: $z=z_1\cdots z_n\in\{0,1\}^n$. The goal for both computers is to obtain the complete string z with high probability with as few oracle calls to z as possible. The phrase “with high probability” means that for every possible z the final answer of the algorithm should be *exactly* z with probability at least 95%. (The probability is thus taken over the runs of the algorithm if we would repeat the protocol for a specific z.) Note that we are primarily concerned with the complexity of the algorithm in terms of oracle calls, both the time and space requirements of the algorithms are not considered when analyzing the complexity differences. The model of an oracle as it used here goes also under the name of black-box, or database-query model.

Definition 6 (Interrogation Task) *Consider an unknown black-box containing n bits $z = z_1 \cdots z_n$. The* interrogation task *is to recover the whole string z (with high probability).*

We call this problem interrogating black-box because afterwards, every possible question about z can be answered correctly (with high probability).

3.4 The Quantum Algorithm

The algorithm that we will present here is an approximation of the procedure described in the Equation 2.6. Instead of calculating the phase values $(-1)^{(x,z)}$ for *all* $x \in \{0,1\}^n$, we will do this only for the strings $x_1 \cdots x_n$ that do not have a Hamming weight $\|x\|_1$ (the number of ones in a bit string) above a certain threshold k. By doing so, we can reduce the number of necessary oracle calls while obtaining an outcome that still has a high fidelity with the 'perfect state' of Equation 2.6. The drawback is this procedure is not exact anymore: with a small probability we obtain a string different from z.

As stated in Fact 4, the value (x,z) corresponds to the parity of a subset of bits z_i, where this set is determined by the ones in the string $x_1 \cdots x_n$. To calculate the parity we can perform a sequence of additions modulo 2 of the relevant z_i values, where each z_i has to be (and can be) obtained by one oracle call. Therefore, the Hamming weight $\|x\|_1$ equals the 'oracle call complexity' of the procedure (for an arbitrary bit $b \in \{0,1\}$):

$$|x\rangle|b\rangle \underset{\|x\|_1 \text{ oracle calls}}{\longmapsto} |x\rangle|b \oplus (x,z)\rangle. \tag{3.1}$$

Since the number of z-queries will be limited by a threshold number k, this implies that we can only compute the parity value (x,z) if the Hamming weight of x is less than or equal to k. The algorithm that performs this conditional parity calculation is denoted by A_k and its behavior is thus defined by:

$$A_k|x\rangle|b\rangle \quad := \quad \begin{cases} |x\rangle|b \oplus (x,z)\rangle & \text{if } \|x\|_1 \leq k, \\ |x\rangle|b\rangle & \text{if } \|x\|_1 > k, \end{cases} \tag{3.2}$$

which can be done with at most k oracle calls for every $x_1 \cdots x_n$. Because A_k is reversible and does not induce any undesired phase changes it follows from the superposition principle that we can apply A_k also to a superposition of different x strings. This will allow us to prove the following theorem.

Theorem 1 (Quantum Interrogation) *Consider an unknown n-bit black box $z = z_1 \cdots z_n \in \{0,1\}^n$. There exist a quantum algorithm with query complexity $\frac{n}{2} + \sqrt{n}$ that recovers the whole string z with 95% probability of success.*

Proof: We exhibit the algorithm in detail. Prepare the state Ψ_k, which is an equally weighted superposition of bit strings of size n with Hamming weight $\|x\|_1$ less than or equal to k, and an additional qubit in the state $\frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$ attached to it:

$$|\Psi_k\rangle \otimes \tfrac{1}{\sqrt{2}}(|0\rangle - |1\rangle) \;:=\; \frac{1}{\sqrt{M_k}} \left(\sum_{x\in\{0,1\}^n}^{\|x\|_1 \le k} |x\rangle \right) \otimes \tfrac{1}{\sqrt{2}}(|0\rangle - |1\rangle), \tag{3.3}$$

with M_k the appropriate normalization factor calculated by the number of x strings that have Hamming weight less than or equal to k:

$$M_k \;:=\; \sum_{i=0}^{k} \binom{n}{i}. \tag{3.4}$$

Applying the above-described protocol A_k (Equation 3.2) to this state yields (requiring k oracle calls):

$$A_k|\Psi_k\rangle \otimes \tfrac{1}{\sqrt{2}}(|0\rangle - |1\rangle) \;=\; \frac{1}{\sqrt{M_k}} \left(\sum_{x\in\{0,1\}^n}^{\|x\|_1 \le k} (-1)^{(x,z)}|x\rangle \right) \otimes \tfrac{1}{\sqrt{2}}(|0\rangle - |1\rangle) \tag{3.5}$$

Here we see how the phases of the state $A_k|\Psi_k\rangle$ contain a part of the desired information about $z_1 \cdots z_n$ similar to Equation 2.6.

If we set k to its maximum $k = n$, then applying an n-fold Hadamard to the first n qubits of $A_k|\Psi_k\rangle$ would give us exactly the state $|z_1 \cdots z_n\rangle$. The minimum value $k = 0$ leads to a state that does not reveal anything about z. For all the other possible values of $0 < k < n$ there we have the situation that applying $\mathrm{H}^{\otimes n}$ to the x-register of $A_k|\Psi_k\rangle$ gives a state that is close to $|z_1 \cdots z_n\rangle$, but not exactly. For a given n, this *fidelity* (statistical correspondence) between the acquired state and z depends on k: as k gets bigger, the fidelity increases.

The n qubits that should give $z_1 \cdots z_n$ after the $\mathrm{H}^{\otimes n}$ transformation, is described by (see Equation 3.5):

$$|\Psi'_k\rangle \;=\; \frac{1}{\sqrt{M_k}} \sum_{x\in\{0,1\}^n}^{\|x\|_1 \le k} (-1)^{(x,z)}|x\rangle. \tag{3.6}$$

The probability that this state gives the correct string of z-bits equals the square of its fidelity with the perfect state $|\Psi'_n\rangle$:

$$\mathrm{Prob}(A_k \text{ outputs } z) \;=\; |\langle\Psi'_k|\Psi'_n\rangle|^2 \tag{3.7}$$

The signs of the amplitudes of $|\Psi'_k\rangle$ and $|\Psi'_n\rangle$ will be the same for all registers x with $\|x\|_1 \leq k$, whereas for the other strings with $\|x\|_1 > k$ the amplitudes of $|\Psi'_k\rangle$ are zero. The fidelity between the two states can therefore be calculated in a straightforward way, yielding for the correctness probability (using Equation 3.4):

$$\mathrm{Prob}(A_k \text{ outputs } z) \quad = \quad \frac{M_k}{2^n} \quad = \quad \frac{1}{2^n}\sum_{i=0}^{k}\binom{n}{i}. \tag{3.8}$$

This equality shows the reason why the algorithm also works for values of k around $\frac{n}{2}+\sqrt{n}$. For large n the binomial distribution approaches the Gaussian distribution. The requirement that the correctness probability has some value significantly greater than $\frac{1}{2}$, translates into the requirement that k has to be bigger than the average $\frac{n}{2}$ by some multiple of the standard deviation $\frac{1}{2}\sqrt{n}$ of the Hamming weights over the set of bit strings $\{0,1\}^n$. Because less that 5% of the binomial distribution is concentrated in the right tail that is at least two standard deviations away from the middle, it can be shown that

$$\mathrm{Prob}(A_{\lfloor\frac{n}{2}+\sqrt{n}\rfloor} \text{ outputs } z) \quad > \quad 0.95 \tag{3.9}$$

for every value of n.

This proves that the following algorithm will give us the requested n oracle values $z_1 \cdots z_n$ with an error-rate of less than 5%, using only $\lfloor\frac{n}{2}+\sqrt{n}\rfloor$ queries to the oracle.

1. **Initial state preparation:** Prepare a register of $n+1$ qubits in the state

   $$\Psi_{\lfloor\frac{n}{2}+\sqrt{n}\rfloor} \otimes \tfrac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$$

   as in Equation 3.3.

2. **Oracle calls:** Apply the A_k procedure of Equation 3.2, for $k = \lfloor\frac{n}{2}+\sqrt{n}\rfloor$ oracle queries.

3. **Hadamard transformation:** Perform n Hadamard transforms to the first n qubits on the register (the state $|\Psi'_k\rangle$ in Equation 3.6).

4. **Final observation:** Observe the same first n qubits in the standard basis $|0\rangle, |1\rangle$. The outcome of this observation is our guess for the oracle description $z_1 \cdots z_n$. This estimation of z will be correct for all n bits with error probability less than 5%.

□

An expected error-rate of significantly less than 5% can easily be obtained if we increase the threshold k with a multiple of the standard deviation $\frac{1}{2}\sqrt{n}$. With the use of the Chernoff bound, we can thus show that

$$\mathrm{Prob}_{\text{error}}(k = \tfrac{n}{2}+\lambda\sqrt{n}) \quad \leq \quad \mathrm{e}^{-\frac{2}{3}\lambda^2}.$$

Hence we conclude that an error rate of ε or less can be established with

$$k \leq \frac{n}{2} + \left(\sqrt{\tfrac{3}{2}\log(\tfrac{1}{\varepsilon})}\right)\sqrt{n}$$

queries to the oracle z.

3.5 Comparison with Classical Algorithms

Consider now a classical computer B_k that is allowed to query the oracle k times. This implies that after the procedure $n-k$ bits of z are still unknown. Under the uniform distribution $\mathrm{Prob}(z) = 2^{-n}$, we thus have a probability of 2^{k-n} of guessing the remaining $n-k$ bit correctly. Hence, the probability of recovering the n-bit string $z_1\cdots z_n$ by a classical algorithm is:

$$\mathrm{Prob}(B_k \text{ outputs } z) \leq \frac{1}{2^{n-k}}. \tag{3.10}$$

This establishes the following lemma.

Lemma 1 (Classical Interrogation) *For an error probability of less than $\frac{1}{2}$, the classical, probabilistic, query complexity of the interrogation problem is n.*

The space complexity of the quantum and the classical algorithms is in both cases linear in n.

3.6 Approximate Interrogation

In this section we ask ourself what happens if we want to know only a certain *fraction* of the n unknown bits. In other words: Given a threshold of k oracle-queries, what is the maximum expected number of correct bits c that we can obtain via an 'approximate interrogation' procedure if we assume the uniform distribution $\mathrm{Prob}(z) = 2^{-n}$ over the strings $z \in \{0,1\}$?

3.7 Classical Approximate Interrogation

In the classical setting the analysis is again straightforward. If we query k out of n bits, then we know k bits with certainty and we have to randomly guess the other $n-k$ bits of which we can expect 50% to be correct. The total number of correct bits will therefore be

$$c_k^{\text{clas}} = \frac{n}{2} + \frac{k}{2}, \tag{3.11}$$

which shows a linear relation between k and c.

3.8 Quantum Approximate Interrogation

The quantum procedure for approximate interrogation will be the same algorithm that we used in the first part of this chapter, but with a different initial state Ψ_k. We now allow the amplitudes α_j of Ψ_k to depend on the Hamming weight of the bit strings x:

$$|\Psi_k\rangle \;=\; \sum_{j=0}^{k} \frac{\alpha_j}{\sqrt{\binom{n}{j}}} \sum_{x\in\{0,1\}^n}^{\|x\|_1=j} |x\rangle, \tag{3.12}$$

with the normalization restriction $\sum_j |\alpha_j^2| = 1$.

After the preparation of this state Ψ_k, the algorithm is continued with an application of the k query procedure A_k, in the same way as described in Section 3.4. The n bits outcome of this protocol will correspond to a certain degree with the interrogated bit string $z_1\cdots z_n$. This degree depends on k and the amplitudes α_j.

3.9 The Expected Number of Correct Bits

In this section we will calculate how many bits we can expect to be correct for the quantum interrogation procedure with the initial state Ψ_k of Equation 3.12. We do this by assuming that the unknown bit string consists of zeros only: $z = 0\cdots0$. The expected number of correct bits for the algorithm equals therefore the expected number of zeros of the observed output string y. Because we can make the assumption $z = 0\cdots 0$ without loss of generality, we then conclude that this number will the expected number of correct bits for any $z \in \{0,1\}^n$.

The inner-product between x and z will be zero for every x, hence applying A_k to Ψ_k will not change the initial state:

$$A_k|\Psi_k\rangle \;=\; \sum_{j=0}^{k} \alpha_j \cdot \frac{1}{\sqrt{\binom{n}{j}}} \sum_{x\in\{0,1\}^n}^{\|x\|_1=j} |x\rangle. \tag{3.13}$$

After this A_k, we perform the n Hadamard transforms on all n qubits, yielding a new state:

$$\begin{aligned} \mathrm{H}^{\otimes n}A_k|\Psi_k\rangle \;&=\; \sum_{j=0}^{k} \frac{\alpha_j}{\sqrt{\binom{n}{j}}} \sum_{x\in\{0,1\}^n}^{\|x\|_1=j} \mathrm{H}^{\otimes n}|x\rangle & (3.14)\\ &=\; \frac{1}{\sqrt{2^n}} \sum_{y\in\{0,1\}^n} \sum_{j=0}^{k} \frac{\alpha_j}{\sqrt{\binom{n}{j}}} \sum_{x\in\{0,1\}^n}^{\|x\|_1=j} (-1)^{(y,x)}|y\rangle & (3.15) \end{aligned}$$

Because the above state is invariant under permutation, the probability of observing a certain string y depends only on its Hamming weight $\|y\|_1$. In the Appendix of this thesis it is shown that this gives us the following equality for the expected number of zeros:

$$\begin{aligned}
\mathbf{E}[\#\text{zeros}(\mathrm{H}^{\otimes n}A_k|\Psi_k\rangle)] &= \sum_{t=0}^{n} t\cdot\binom{n}{t}|\langle 0^t1^{n-t}|\mathrm{H}^{\otimes n}A_k|\Psi_k\rangle|^2 \\
&= \frac{1}{2^n}\sum_{t=0}^{n} t\cdot\binom{n}{t}\left|\sum_{j=0}^{k}\frac{\alpha_j}{\sqrt{\binom{n}{j}}}\sum_{x\in\{0,1\}^n}^{||x||_1=j}(-1)^{(0^t1^{n-t},x)}\right|^2 \\
&= \frac{1}{2^n}\sum_{t=0}^{n} t\cdot\binom{n}{t}\left|\sum_{j=0}^{k}\frac{\alpha_j}{\sqrt{\binom{n}{j}}}\sum_{i=0}^{j}(-1)^i\binom{n-t}{i}\binom{t}{j-i}\right|^2 \\
&= \frac{n}{2}+\sum_{j=0}^{k-1}\mathrm{Re}(\alpha_j\alpha_{j+1}^*)\sqrt{j+1}\sqrt{n-j}.
\end{aligned}$$

We can therefore conclude that the expected number c_k of correctly guessed bits for the quantum protocol will be (for given k and α_j):

$$c_k^{\text{quant}} = \frac{n}{2}+\sum_{j=0}^{k-1}\mathrm{Re}(\alpha_j\alpha_{j+1}^*)\sqrt{j+1}\sqrt{n-j}. \qquad (3.16)$$

This equation allows us to optimize the α_j amplitudes such that c_k will be as big as possible. (Note that for such an optimal solution we can always assume $\alpha_j \in \mathbb{R}$ without loss of generality.) Two examples of such optimizations will be given below, both of them showing an improvement over the classical algorithm.

3.10 Interrogation with One Quantum Query

If we allow the quantum computer to ask only one query ($k = 1$) to the oracle, then Equation 3.16 is maximized by choosing $\alpha_0 = \alpha_1 = \frac{1}{\sqrt{2}}$, thus giving for the expected number of correct bits

$$c_1^{\text{quant}} = \frac{n}{2}+\frac{\sqrt{n}}{2}. \qquad (3.17)$$

When we compare this with Equation 3.11, we see that a classical algorithm would require $k = \sqrt{n}$ queries to match the power of a single quantum query.

3.11 Interrogation with Many Queries

Let us assume that k is a square with $0 \leq \frac{k}{n} \leq \frac{1}{2}$. We can then define the amplitudes $\alpha_j \in \mathbb{R}$ according to

$$\alpha_j = \begin{cases} 0 & \text{if } 0 \leq j \leq k - \sqrt{k} \\ \frac{1}{\sqrt[4]{k}} & \text{if } k - \sqrt{k} < j \leq k \end{cases} \tag{3.18}$$

Using Equation 3.16, this gives for the expected *ratio* of correct bits

$$\begin{aligned} \frac{c_k^{\text{quant}}}{n} &= \frac{1}{2} + \frac{1}{n\sqrt{k}} \sum_{j=k-\sqrt{k}+1}^{k-1} \sqrt{j+1}\sqrt{n-j} & (3.19) \\ &= \frac{1}{2} + \sqrt{\frac{k}{n}\left(1 - \frac{k}{n}\right)} - O(\tfrac{1}{\sqrt{n}}). & (3.20) \end{aligned}$$

From this analysis it follows that for big enough n and all values $k \leq \frac{n}{2}$, we can ignore the $O(\frac{1}{\sqrt{n}})$ term in the above equation. For k bigger than $\frac{n}{2}$, we can always adopt the same interrogation scheme that we used to reach the perfect correctness rate $c^q_{n/2} \approx n$. This gives us the following theorem.

Theorem 2 *For big enough n and k queries, the above described algorithm has an expected correctness rate c/n of*

$$\frac{c_k^{\text{quant}}}{n} = \begin{cases} \frac{1}{2} + \sqrt{\frac{k}{n}\left(1 - \frac{k}{n}\right)} & \textit{if } 0 \leq k \leq \frac{n}{2} \\ 1 & \textit{if } k > \frac{n}{2} \end{cases} \tag{3.21}$$

Lemma 2 (Classical Approximate Interrogation) *In the same setting as the previous section, the classical fraction of correct bits is*

$$\frac{c_k^{\text{clas}}}{n} = \frac{1}{2} + \frac{k}{2n}. \tag{3.22}$$

This result is summarized in Figure 3.1 and gives a clear example of a quantum reduction in the query complexity of the approximate interrogation problem. This improvement is especially significant for small values of $\frac{k}{n}$. For example, if we allot the quantum protocol $\frac{n}{10}$ queries, then we can expect 80% of the bits to be correct. Any classical algorithm would need six times as much ($k = \frac{3n}{5}$) queries to obtain such a ratio.

3.12 Conclusions

The model of quantum computation does not permit a general significant speed-up of the existing classical algorithms.[10] Instead, we have to investigate for each different kind of problem whether there is a possible gain by using quantum algorithms or not.

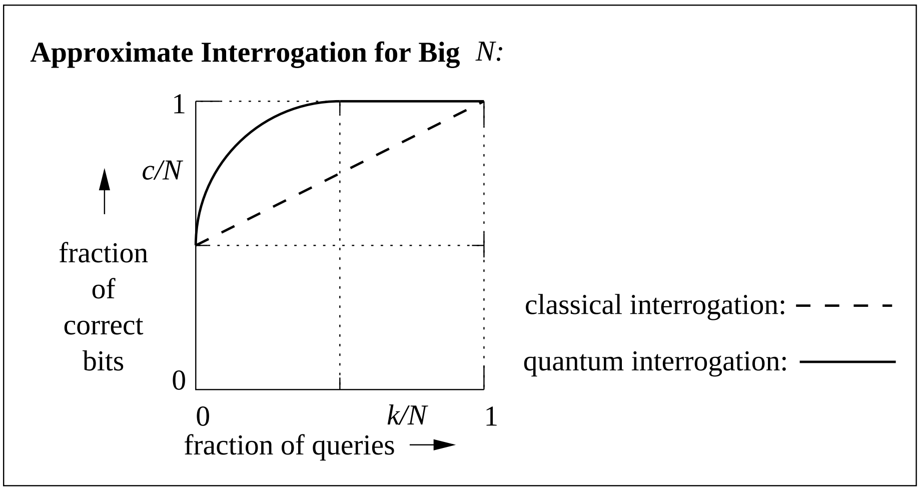

Figure 3.1: Comparison of the interrogation effectiveness between classical and quantum computers.

Here it has been shown that for every binary function $z : \{0,1\}^n \to \{0,1\}$ we can obtain the full description of the function with high probability while querying z only $\frac{n}{2} + \sqrt{n}$ times. A classical computer always requires n calls to determine $z_1 \cdots z_n$ with the same kind of success probability.

The lower bounds on PARITY (with bounded error) and OR (with no allowed error) for black-boxes[10, 42] show us that any quantum algorithm *must* use at least $\frac{n}{2}$ calls to obtain z with bounded error, and that the full n queries are necessary to determine the string without error, respectively. Furthermore, it has been shown by Farhi *et al.*[40] that the $\frac{n}{2} + \sqrt{n}$ of this chapter cannot be reduced any further: it is a tight bound for the interrogation task (up to a constant in front of the $\sqrt{n}$ term).

The term 'approximate interrogation' was used for the scenario where we are interested in obtaining a certain fraction of the n unknown bits. Again we could see how a quantum procedure outperforms the possible classical algorithms (Figure 3.1).

Chapter 4

Quantum Bounded Queries

It is known that that a super-polynomial quantum improvement can only be obtained if we consider problems that are more structured than those in the black-box model of computation.[10] In this chapter we look at the query complexity of problems that can be computed in polynomial time with the help of, for example, an oracle for the SAT problem. It is shown how in this setting a quantum computer requires less queries than a classical computer, provided that standard complexity assumptions like $\mathsf{P} \neq \mathsf{NP}$ are true.

4.1 Introduction

We combine the classical notions and techniques for bounded query classes with those developed in quantum computing. We give strong evidence that quantum queries to an oracle in the class NP does indeed reduce the query complexity of decision problems. Under traditional complexity assumptions, we obtain an exponential speed-up between the quantum and the classical query complexity of function classes.

For decision problems and function classes we obtain the following results (see the appendix of this thesis for a brief overview of these complexity classes):

- $\mathsf{P}_{\shortparallel}^{\mathsf{NP}[2k]} \subseteq \mathsf{EQP}_{\shortparallel}^{\mathsf{NP}[k]}$
- $\mathsf{P}_{\shortparallel}^{\mathsf{NP}[2^{k+1}-2]} \subseteq \mathsf{EQP}^{\mathsf{NP}[k]}$
- $\mathsf{FP}_{\shortparallel}^{\mathsf{NP}[2^{k+1}-2]} \subseteq \mathsf{FEQP}^{\mathsf{NP}[2k]}$
- $\mathsf{FP}_{\shortparallel}^{\mathsf{NP}} \subseteq \mathsf{FEQP}^{\mathsf{NP}[O(\log n)]}$

For sets A that are many-one complete for PSPACE or EXP we show that $\mathsf{FP}^A \subseteq \mathsf{FEQP}^{A[1]}$. Sets A that are many-one complete for PP have the property that $\mathsf{FP}_{\shortparallel}^{A} \subseteq \mathsf{FEQP}^{A[1]}$. In general we prove that for any set A there is a set X such that $\mathsf{FP}^A \subseteq \mathsf{FEQP}^{X[1]}$, establishing that no set is superterse in the quantum setting.

The query complexity of a function is the minimum number of queries (to some oracle) that are needed to compute one value of this function. With *bounded* query complexity we look at the set of functions that can be calculated if we put an upper bound on the number of queries that we allow the computer to ask the oracle. This notion has been extensively studied both in the resource bounded setting [2, 4, 5, 13, 12, 11, 17, 60, 75, 104] and in the recursive setting[15, 16]. This notion and its variants has lead to a series of techniques and tools that are used throughout complexity theory.

In this chapter we combine some of the bounded query notions with quantum computation. The main goal is to further—as was done by Fortnow and Rogers [43]—the incorporation of quantum computation complexity classes into standard classical complexity theory. We feel that the synthesis of quantum computation and classical complexity theory serves two purposes. First, it is important to know the limits of feasible quantum computation and these can be clarified by expressing them in the framework of classical computation. Second, the insights of quantum computation can be useful for classical complexity theory in turn.

We start out with the class of sets (or decision problems) that are computable in polynomial time with bounded queries to a set in NP. We consider the setting where the queries are adaptive (i.e., a query may depend on the answers to previous ones), as well as where they are non-adaptive. Classically, it is known that any decision problem that can be solved in polynomial time with k adaptive queries to a set in NP (the class $\mathsf{P}^{\mathsf{NP}[k]}$) can also be solved with 2^k-1 non-adaptive queries (the class $\mathsf{P}_{\shortparallel}^{\mathsf{NP}[2^k-1]}$, where "$\shortparallel$" indicates the parallel or non-adaptive queries), and vice-versa [13]. In other words: $\mathsf{P}^{\mathsf{NP}[k]} = \mathsf{P}_{\shortparallel}^{\mathsf{NP}[2^k-1]}$. Moreover, there is strong evidence that this trade-off is optimal in the sense that every non-adaptive class $\mathsf{P}_{\shortparallel}^{\mathsf{NP}[k]}$ is different for different values of k. For example if $\mathsf{P}_{\shortparallel}^{\mathsf{NP}[2]} \subseteq \mathsf{P}^{\mathsf{NP}[1]}$, then the polynomial hierarchy collapses [60] (see also [27, 52]).

We will see that if we allow the query machine to make use of quantum mechanical effects such as superposition and interference the situation changes. In the non-adaptive case we will show that $2k$ classical queries can be simulated with only k non-adaptive ones on a quantum computer and in the adaptive case we show how to simulate $2^{k+1}-2$ classical queries with only k quantum queries. The natural quantum analog of P is the class EQP, which stands for *exact quantum polynomial time.* This is the class of sets or decision problems that is computable in polynomial time with a quantum computer that makes no errors (i.e., is exact). Then, our results are that

$$\mathsf{P}_{\shortparallel}^{\mathsf{NP}[2k]} \subseteq \mathsf{EQP}_{\shortparallel}^{\mathsf{NP}[k]} \quad \text{and} \quad \mathsf{P}_{\shortparallel}^{\mathsf{NP}[2^{k+1}-2]} \subseteq \mathsf{EQP}^{\mathsf{NP}[k]}.$$

In particular it follows from this result that $\mathsf{P}_{\shortparallel}^{\mathsf{NP}[2]} \subseteq \mathsf{EQP}^{\mathsf{NP}[1]}$ (see also [36]).

In order to prove these results we combine the classical mind-change technique [13] with the one query version (see [31]) of the first quantum algorithm developed by David Deutsch [38].

Next, we turn our attention to *functions* that are computable with bounded queries to a set in NP. Compared to the decision problems there is probably no nice trade-off between adaptive and non-adaptive queries for functions. This is because the following is known [17]: for any k the inclusion $\mathsf{FP}_{\shortparallel}^{\mathsf{NP}[k]} \subseteq \mathsf{FP}^{\mathsf{NP}[k-1]}$ implies that $\mathsf{P} = \mathsf{NP}$. Moreover, if $\mathsf{FP}_{\shortparallel}^{\mathsf{NP}} \subseteq \mathsf{FP}^{\mathsf{NP}[O(\log n)]}$ then the polynomial time hierarchy collapses [12, 87, 98].

When the adaptive query machine is a quantum computer, things are different and we seem to get a trade-off between adaptiveness and query complexity. We show the following:

$$\mathsf{FP}_{\shortparallel}^{\mathsf{NP}[2^{k+1}-2]} \subseteq \mathsf{FEQP}^{\mathsf{NP}[2k]} \quad \text{and} \quad \mathsf{FP}_{\shortparallel}^{\mathsf{NP}} \subseteq \mathsf{FEQP}^{\mathsf{NP}[O(\log n)]}.$$

Here $\mathsf{FEQP}^{\mathsf{NP}[k]}$ is the class of functions that is computable by an exact quantum Turing machine that runs in polynomial time and is allowed to make k queries to a set in NP. The proofs of these results use our previous results on decision problems and a quantum algorithm developed by Deutsch-Jozsa [39] and Bernstein-Vazirani [22].

Using the same ideas we are able to show that for any set A there exists a set X such that $\mathsf{FP}^A \subseteq \mathsf{FEQP}^{X[1]}$, establishing that no set is 'superterse'. Also because the complexity of X is not much harder than that of A (the problem X is Turing reducible to A), we get quite general theorems for complete sets of complexity classes.

For a complexity class C that is closed under Turing reductions, and a problem $A \in \mathsf{C}$ that is many-one complete for the class C, the inclusion $\mathsf{FP}^{\mathsf{C}} \subseteq \mathsf{FEQP}^{A[1]}$ is proven. This holds in particular for the set QBF of the *true quantified Boolean formulae* which is a PSPACE complete problem, and the complete sets for the class EXP. If C is a class that is closed under truth-table reductions, then it holds that $\mathsf{FP}_{\shortparallel}^{\mathsf{C}} \subseteq \mathsf{FEQP}^{A[1]}$. The Theta levels of the polynomial hierarchy and PP are examples of such classes.

The ingredients for all our results are standard quantum algorithms combined with well known techniques from complexity theory. Nevertheless we feel that this combination gives a new point of view on the nature of bounded query classes and the structure of complete sets in general.

4.2 Classical Complexity Theory

We assume the reader to be familiar with basic notions of complexity theory such as the various complexity classes and types of reducibility as can be found in many textbooks in the area [6, 7, 46, 58]. The essentials for this chapter are mentioned below.

For a set (decision problem) A we will identify A with its characteristic function. Hence for a string x we have $A(x) \in \{0,1\}$, and $A(x) = 1$ if and only if $x \in A$. A class C consists of a set of decision problems. A problem A is many-one poly-time, or $\leq_m^p$-complete for a class C if for any problem $B \in \mathsf{C}$, there exists a polynomial-time computable function or "Karp-reduction" τ such that $x \in B$ if and only if $\tau(x) \in A$. The typical example of such a complete problem is SAT (the set of satisfiable Boolean formulae) which is $\leq_m^p$-complete for the class NP. The class FP indicates the set of *functions* that can be calculated on a polynomial time, deterministic Turing machine.

An oracle Turing machine is *non-adaptive*, if it can produce a list of all of the oracle queries it is going to make before it makes the first query. For any set A, the elements of the class $\mathsf{P}^{A[k]}$ ($\mathsf{FP}^{A[k]}$) are the languages (functions) that are computable by polynomial time Turing machines that accesses the oracle A at most k times on each input. The class $\mathsf{P}_{\shortparallel}^{A[k]}$ and $\mathsf{FP}_{\shortparallel}^{A[k]}$ allow only non-adaptive access to A. The notation $\mathsf{P}^{\mathsf{NP}[q(n)]}$ is used to indicate algorithms that might require $q(n)$ calls to an NP oracle, where q is a function of the input size n.

The class NP can be generalized by defining the *polynomial time hierarchy*. We start with the definitions $\Delta_1^p = \mathsf{P}$ and $\Sigma_1^p = \mathsf{NP}$, and then for the higher levels continue in an inductive fashion according to $\Delta_{i+1}^p = \mathsf{P}^{\Delta_i^p}$ and $\Sigma_{i+1}^p = \mathsf{NP}^{\Sigma_i^p}$ for $i = 2, 3, \ldots$ Many complexity theorists conjecture that this polynomial time hierarchy is infinite, i.e., $\Sigma_{i+1}^p \neq \Sigma_i^p$ for all i.

A class C of languages is closed under Turing (truth-table) reduction if any decision problem that can be solved with a polynomial time Turing machine and (non-adaptive) queries to a set in C, is itself also an element of C. Examples of such classes are PSPACE, EXP, and the Delta levels Δ_{i+1}^p. The classes PP and $\Theta_{i+1}^p = \mathsf{P}_{\shortparallel}^{\Sigma_i^p}$ (Theta levels of the polynomial hierarchy) are for example closed under this truth-table-reduction.

4.3 Quantum Complexity Classes

The class EQP is the collection of those sets that can be computed by a quantum Turing machine that runs in polynomial time and accepts every string j with probability 1 or 0. Likewise, we define the class of functions FEQP as the class of functions that can be computed exactly by some quantum Turing machine that runs in polynomial time. The output of the Turing machine is the function value (rather than a single decision bit).

We model oracle computation as follows (see also [19]). An oracle Turing machine has a special query tape, and during the computation the Turing machine may enter a special pre-query state to make a query to the oracle set A. Suppose the query tape contains the state $|i\rangle|b\rangle$ (i represents the query and b is a bit meant to receive the answer to the query). The result of this operation is that after the call the machine will go into a special state called the post-query state and that the query tape has changed into $|i\rangle|A(i) \oplus b\rangle$, where $\oplus$ is the EXCLUSIVE OR. We will denote this unitary operation by U_A. Note that U_A only changes the contents of the special query answer bit b, and leaves all the other registers unchanged.

As with classical oracle computation, we make the distinction between adaptive and non-adaptive quantum oracle machines. We call a quantum oracle machine non-adaptive if on every computation path a list of all the oracle queries (on this path) is generated before the first query is made.

The class $\mathsf{EQP}^{A[k]}$ are the sets recognized by an exact quantum Turing machine that runs in polynomial time and makes at most k adaptive queries to the oracle for

A. Likewise, we define classes like $\mathsf{EQP}_{\shortparallel}^{A[q(n)]}$, $\mathsf{FEQP}^{A[q(n)]}$, and $\mathsf{FEQP}_{\shortparallel}^{A[q(n)]}$, for non-adaptive decision, adaptive function, and non-adaptive function classes respectively (with $q(n)$ a function that gives an upper bound on the number of queries and n the size of the input string).

4.4 Decision Problems

In this section we will investigate the extra power that a polynomial time, exact quantum computer yields compared to classical deterministic computation when querying a set in the class NP. In the case of deterministic computation the following equality between adaptive and non-adaptive queries to NP is well known.

Fact 6 *[13, 29, 104]*

1. *For all $k \geq 0$ we have $\mathsf{P}_{\shortparallel}^{\mathsf{NP}[2^k-1]} = \mathsf{P}^{\mathsf{NP}[k]}$.*

2. *For any polynomial $q(n) > 1$ the equality $\mathsf{P}_{\shortparallel}^{\mathsf{NP}[q(n)]} = \mathsf{P}^{\mathsf{NP}[O(\log(q(n)))]}$ holds.*

Proof: Both items are proved in a similar way which has two parts. The first part shows that computing a function in $\mathsf{P}_{\shortparallel}^{\mathsf{NP}[2^k-1]}$ can be reduced to computing the *parity* of $2^k - 1$ other queries to NP. The second part then proceeds by showing that using binary search one can compute the parity of $2^k - 1$ NP-queries with k *adaptive* queries to $\textsc{sat}$. On the other hand, it is trivial to see that any computation with k adaptive queries can be simulated exhaustively with $2^k - 1$ non-adaptive oracle calls. □

There is also strong evidence that the above trade-off is tight (see [14, 60]). It follows for example that if $\mathsf{P}_{\shortparallel}^{\mathsf{NP}[2]} = \mathsf{P}^{\mathsf{NP}[1]}$ then the polynomial hierarchy collapses [60]. (See [27] for the latest developments with respect to this question.)

Perhaps surprisingly the situation changes when the query machine is quantum mechanical. Using the one-call-parity trick of Fact 3, we will show that a quantum Turing machine can compute decision problems with half the number of non-adaptive queries.

Theorem 3 *For all $k \geq 0$ we have the inclusion $\mathsf{P}_{\shortparallel}^{\mathsf{NP}[2k]} \subseteq \mathsf{EQP}_{\shortparallel}^{\mathsf{NP}[k]}$.*

Proof: Without loss of generality we will assume that the queries are made to $\textsc{sat}$, and that the predicate that is computable with $2k$ queries to $\textsc{sat}$ is $f(x)$. Let $\psi_1, \psi_2, \ldots, \psi_{2k}$ be the queries that the computation of $f(x)$ makes. We will use the proof technique of Fact 6 (also called mind-change technique) which enables us to compute $f(x)$ by calculating the single bit $\textsc{sat}(\phi_1) \oplus \cdots \oplus \textsc{sat}(\phi_{2k})$. Here the new formulae $\phi_1, \ldots, \phi_{2k}$ can be computed in polynomial time from $\psi_1, \ldots, \psi_{2k}$, f, and x, but without having to consult $\textsc{sat}$.

Next, we use Fact 3 to compute the parity $\textsc{sat}(\phi_i) \oplus \textsc{sat}(\phi_{i+1})$ for odd i ($1 \leq i < 2k$) with k non-adaptive queries to $\textsc{sat}$. Finally we compute the parity of these answers, thus obtaining the necessary information for calculating $f(x)$. □

Lemma 3 $\mathsf{P}_{\shortparallel}^{\mathsf{NP}[2]} \subseteq \mathsf{EQP}^{\mathsf{NP}[1]}$ *(see [36]).*

We do not know whether this is tight. It would be interesting to either improve this result to $\mathsf{P}^{\mathsf{NP}[2]} \subseteq \mathsf{EQP}^{\mathsf{NP}[1]}$ or to show as a consequence of this that the polynomial time hierarchy collapses.

Fact 6 relates adaptive query classes to non-adaptive ones, thereby establishing an exponential gain in the number of queries ($2^k - 1$ versus k queries). We will now show how to use the Deutsch trick to improve this result slightly in the quantum case.

Theorem 4 $\mathsf{P}_{\shortparallel}^{\mathsf{NP}[2^{k+1}-2]} \subseteq \mathsf{EQP}^{\mathsf{NP}[k]}$ *for all* $k \geq 0$.

Proof: The proof is by induction on k. For $k = 1$ we return to the situation of Lemma 3. Let the predicate $f(x)$ be computable with $2^{k+1} - 2$ non-adaptive queries to SAT. As in the proof of Theorem 3 we reduce the $2^{k+1} - 2$ queries ψ_i that $f(x)$ makes, to the calculation of the parity-bit $\text{SAT}(\phi_1) \oplus \cdots \oplus \text{SAT}(\phi_{2^{k+1}-2})$. Next, we construct $2^{k+1} - 2$ new formulae $\chi_1, \ldots, \chi_{2^{k+1}-2}$ according to:

$$\chi_i \text{ is satisfiable} \iff |\{\phi_1, \ldots, \phi_{2^{k+1}-2}\} \cap \text{SAT}| \geq i.$$

The construction of each such χ_i can be done in polynomial time. Consider the non-deterministic polynomial time Turing machine M that on input $\langle i, \phi_1, \ldots, \phi_{2^{k+1}-2}\rangle$, accepts if and only if it can find for i of the formulae a satisfying assignment. Cook and Levin [34, 66] —proving that SAT is $\leq_m^p$ -complete for NP— showed that any polynomial time non-deterministic Turing machine computation $M(x)$ in polynomial time can be transformed into a formula that is satisfiable if and only if $M(x)$ has an accepting computation. Let χ_i be the result of this Cook-Levin reduction.

Note the following two properties of those formulae χ_i:

1. The parity $\text{SAT}(\phi_1) \oplus \cdots \oplus \text{SAT}(\phi_{2^{k+1}-2})$ is the same as the parity $\text{SAT}(\chi_1) \oplus \cdots \oplus \text{SAT}(\chi_{2^{k+1}-2})$.
2. For every i we have $\text{SAT}(\chi_i) \geq \text{SAT}(\chi_{i+1})$.

Now we are ready to make the first query. We compute the parity of $\chi_{2^{k-1}}$ and $\chi_{2^{k-1}+2^k-1}$. This can be done in one query using Fact 3. By doing this we have at the cost of one query reduced the question of computing the parity of $2^{k+1} - 2$ formulae to computing the parity of $2^k - 2$. These we can solve using $k-1$ queries using the induction hypothesis. To see this observe the following. For convenience set $a = 2^{k-1}$ and $b = 2^{k-1} + 2^k - 1$.

Suppose the parity of χ_a and χ_b is odd, with $a < b$. From the second property above, it follows that $\chi_a = 1$ and $\chi_b = 0$, and hence that $\chi_1, \ldots, \chi_a$ are all satisfiable and $\chi_b, \ldots, \chi_{2^{k+1}-2}$ are all unsatisfiable. Also note that a is even, so the parity of $\chi_1, \ldots, \chi_{2^{k+1}-2}$ is the same as the parity of $\chi_{a+1}, \ldots, \chi_{b-1}$ (these are $2^k - 2$ many formulae).

On the other hand assume that the parity of χ_a and χ_b is even. This means (again using property 2 above) that $\chi_a, \ldots, \chi_b$ are all either satisfiable or unsatisfiable and hence have even parity. So again the question reduces to the parity of the remaining formulae: $\chi_1, \ldots, \chi_{a-1}$ and $\chi_{b+1}, \ldots, \chi_{2^k+1-2}$. Which happen to be $2^k - 2$ many formulae. □

In essence the above technique seems to boil down to searching in an ordered list $\chi_1, \ldots, \chi_{2^{k+1}-2}$. In [56] it has been shown that this can not be done with less than $\frac{\log n}{\pi \log \mathrm{e}} - O(1)$ queries. On the other hand, results by Farhi *et al.* [41] and [56] indicate that the query complexity of the ordered search problem is upper bounded by $\frac{1}{\alpha}\log n + O(1)$, with α at least $1.88\ldots$. Using these results it is likely that we can strengthen the above theorem to $\mathsf{P}_{\|}^{\mathsf{NP}[2^{\alpha k - O(1)}]} \subseteq \mathsf{EQP}^{\mathsf{NP}[k]}$.

4.5 Functions computable with queries to NP Oracles

Now we turn our attention to function classes where the algorithm can output bit *strings* rather than single bits. We will see that in this scenario the difference between classical and quantum computation becomes more pronounced.

We start out by looking at functions that are computable with queries to a complete set for the class NP. Classically the situation is not as well understood as the class of decision problems. There is strong evidence that the analog of Fact 6 is not true.

Fact 7 *The following holds for the classical, exact computation of functions:*

1. *If for some $k \geq 0$ we have $\mathsf{FP}_{\|}^{\mathsf{NP}[k+1]} \subseteq \mathsf{FP}^{\mathsf{NP}[k]}$, then $\mathsf{P} = \mathsf{NP}$ [17].*
2. *If for all polynomials $q(n)$ (with n the size of the input string): $\mathsf{FP}_{\|}^{\mathsf{NP}[q(n)]} \subseteq \mathsf{FP}^{\mathsf{NP}[O(\log n)]}$, then $\mathsf{NP} = \mathsf{R}$ (and the polynomial hierarchy collapses) [12, 87, 98].*

When we allow the adaptive query machine to be quantum mechanical the picture becomes again quite different. We will show for example that the inclusion $\mathsf{FP}_{\|}^{\mathsf{NP}[q(n)]} \subseteq \mathsf{FEQP}^{\mathsf{NP}[2\log(q(n))]}$ holds (and this does not imply $\mathsf{NP} = \mathsf{R}$ as far as we know). In order to do so we will use Fact 4.

Let us turn back now to our setting of bounded query classes. Using the quantum tricks of Sections 2.4 and 2.5 we can establish the following result.

Theorem 5 *For exact function calculation with the use of an oracle in* NP *it holds that*

1. *$\mathsf{FP}_{\|}^{\mathsf{NP}[2^{k+1}-2]} \subseteq \mathsf{FEQP}^{\mathsf{NP}[2k]}$ for any $k \geq 0$,*
2. *$\mathsf{FP}_{\|}^{\mathsf{NP}} \subseteq \mathsf{FEQP}^{\mathsf{NP}[O(\log n)]}$.*

Proof: Fix $k \geq 0$, the input z of length m and let g be the function in $\mathsf{FP}_{\|}^{\mathrm{SAT}[2^{k+1}-2]}$. Suppose that $g(z) = (a_1 \cdots a_n) = a$ with $n = m^c$ for some c depending on g. The goal is to obtain the state:

$$|\text{Output}\rangle \;=\; \frac{1}{\sqrt{2^n}} \sum_{x\in\{0,1\}^n} (-1)^{(x,a)}|x\rangle, \tag{4.1}$$

since with this state one application of $\mathrm{H}^{\otimes n}$ will give us $a = g(z)$ (*cf.* 2.6). Clearly, we can obtain this state if we have access to a function f with the property

$$f_z(x) \;=\; \sum_{i=1}^{n} a_i x_i \bmod 2, \tag{4.2}$$

for every $x \in \{0,1\}^n$.

The goal thus is to transform the function we *have* access to—SAT in our case—into one that resembles the one in Equation 4.2. The way to do this is to make use of a quantum subroutine. Observe the following: the binary function $f_z(x) = (x,a)$ is in $\mathsf{P}_{||}^{\mathrm{SAT}[2^{k+1}-2]}$ because we can first compute $g(z) = a$ with $2^{k+1} - 2$ queries to SAT and then determine (x,a). By Theorem 4 this function is computable in $\mathsf{EQP}^{\mathrm{SAT}[k]}$. Hence, when we use this adaptive EQP algorithm in superposition we have the desired function f. There is however one problem with this approach. The algorithm that comes out of Theorem 4 leaves several of the registers in states depending on the input x and SAT. For example the algorithm that computes the parity of two function calls in one generates a phase of (-1) depending on the value of the first function call (see Equation 2.4). These changes in registers and phase shifts obstruct our base quantum machine and as a consequence the sum computed in Equation 2.6 does not work out the way we want (*i.e.*, the interference pattern is different and terms do not cancel out as nice as before.)

The solution to this kind of 'garbage' problem is as follows:

1. Compute $f_z(x)$ with k queries to SAT.

2. Copy the outcome onto an extra auxiliary qubit (by setting the auxiliary bit b to the EXCLUSIVE OR of b and the outcome).

3. Reverse the computation of $f_z(x)$ making another k queries to SAT.

Observe that when we compute $f_z(x)$ in this way, all the phase changes and registers are reset and are in the same state as before computing f, except for the auxiliary qubit that contains the answer. Since the subroutine was exact (*i.e.*, in EQP) the answer bit is a classical bit and will not interfere with the rest of the computation. Note that this corresponds exactly to one oracle call to f. Thus we simulated 1 call to f with $2k$ queries to SAT and hence have established a way of producing the desired state of Equation 4.1.

The second part of the theorem is proved in a similar way now using part 2 of Fact 6. □

4.6 Terseness, and other Complexity Classes

The quantum techniques described above are quite general and can be applied to sets outside of NP. Classically the following question has been studied (see [12] for more information). For any set A define the function $F_n^A(x_1,\ldots,x_n) = (A(x_1)\cdots A(x_n))$ which is an n bit vector telling which of the x_i's is in A and which ones are not. A basic question now is: how many queries to A do we need to compute F_n^A? Sets for which F_n^A can not be computed with less than n queries to A (*i.e.*, $F_n^A \notin \mathsf{FP}^{A[n-1]}$) are called P-*terse*. We call the decision problem A P-*superterse* if $F_n^A \notin \mathsf{FP}^{X[n-1]}$ for any set X. The next theorem shows that this last notion is not useful in the quantum setting.

Theorem 6 *Let A be a subset of $\mathbb{N}$ and let the function $F_n^A : \mathbb{N}^n \to A^n$ be defined by $F_n^A(x_1,\ldots,x_n) := (A(x_1),\ldots,A(x_n))$, where $A(x) = 0$ if $x \notin A$ and $A(x) = 1$ if $x \in A$. For any set A there exists a set $X \subseteq \mathbb{N}$ such that for all n we have $F_n^A \in \mathsf{FEQP}^{X[1]}$.*

Proof: Let X be the following set:

$$X = \{\langle z_1\cdots z_n, x_1\cdots x_n\rangle | (F_n^A(z_1,\ldots,z_n), x_1\cdots x_n) = 1\}.$$

Using the same approach as the proof of Theorem 5 it is not hard to see that F_n^A can be computed relative X with only a single query. □

Using the same idea we can prove the following general theorem about oracles for complexity classes other than NP.

Theorem 7 *Let C be a complexity class and the set A $\leq_m^p$-complete for C.*

1. *If C is closed under $\leq_T^p$-reductions then $\mathsf{FP}^{\mathsf{C}} = \mathsf{FP}^A \subseteq \mathsf{FEQP}^{A[1]} = \mathsf{FEQP}^{\mathsf{C}[1]}$.*

2. *If C is closed under $\leq_{tt}^p$-reductions then $\mathsf{FP}_{\shortparallel}^{\mathsf{C}} = \mathsf{FP}_{\shortparallel}^A \subseteq \mathsf{FEQP}^{A[1]} = \mathsf{FEQP}^{\mathsf{C}[1]}$.*

Proof: Let f be the function we want to compute relative to A. Without loss of generality we assume that $\ell(f(z)) = \ell(z)^c$ for some c depending only on f. As before we construct the following set:

$$X = \{\langle z, y\rangle | (f(z), y) = 1, \text{and } \ell(y) = \ell(z)^c = \ell(f(z))\}.$$

As in Theorem 6 it follows that $f(z)$ is computable with one quantum query to X. Since C is closed under $\leq_T^p$-reductions and $X \leq_T^p A$, it follows that $X \in \mathsf{C}$. Furthermore, since A is $\leq_m^p$-complete for C it also follows that $X \leq_m^p A$. Thus the quantum query can be made to A itself instead of X. The proof of the second part of the theorem is analogous to the first. □

This last theorem gives us immediately the following two lemmas about quantum computation with oracles for some known complexity classes.

Lemma 4

$$\begin{aligned}\mathsf{FP}^{\mathsf{PSPACE}} &\subseteq \mathsf{FEQP}^{\mathsf{PSPACE}[1]}\\ \mathsf{FP}^{\mathsf{EXP}} &\subseteq \mathsf{FEQP}^{\mathsf{EXP}[1]}\\ \mathsf{FP}^{\Delta_i^p} &\subseteq \mathsf{FEQP}^{\Delta_i^p[1]}\end{aligned}$$

for the Delta levels Δ_i^p in the polynomial time hierarchy.

Lemma 5

$$\begin{aligned}\mathsf{FP}_{\shortparallel}^{\mathsf{PP}} &\subseteq \mathsf{FEQP}^{\mathsf{PP}[1]}\\ \mathsf{FP}_{\shortparallel}^{\Theta_i^p} &\subseteq \mathsf{FEQP}^{\Theta_i^p[1]}\end{aligned}$$

with $\Theta_{i+1}^p = \mathsf{P}_{\shortparallel}^{\Sigma_i^p}$.

The first lemma holds in particular for $A = \textsc{qbf}$ (the set of true quantified Boolean formulae) which is PSPACE-complete. Observe also that the situation is quite different in the classical setting, since for EXP-complete sets the above is simply not true.

4.7 Conclusions and Open Problems

We have combined techniques from complexity theory with some of the known quantum algorithms. In doing so we showed that a quantum computer can compute certain functions with fewer queries than classical deterministic computers. Many question however remain. Is it possible to get trade-off results between the adaptive class $\mathsf{EQP}^{\mathsf{NP}[k]}$ and the non-adaptive $\mathsf{EQP}_{\shortparallel}^{\mathsf{NP}[2^k-1]}$ for quantum machines? Are the results we present here optimal? (Especially the recent results on exact searching in an ordered list[41] and [56] deserve further analysis as they seem to suggest a reduction of the quantum query complexity of Theorems 4 and 5 by a factor of two.)

What can one deduce from the assumption that $\mathsf{P}^{\mathsf{NP}} \subseteq \mathsf{EQP}^{\mathsf{NP}[1]}$? Is it true that for any set A we have $\mathsf{P}^A \subseteq \mathsf{EQP}^{A[1]}$ or are there sets where this is not true? A random set would be a good candidate where more than one quantum query is necessary.

Chapter 5

Quantum Algorithms and Combinatorics

In this chapter we investigate how we can employ the structure of combinatorial objects like Hadamard matrices and weighing matrices to devise new quantum algorithms. We show how the properties of a weighing matrix can be used to construct a problem for which the quantum query complexity is significantly lower than the classical one. It is pointed out that this scheme captures both Bernstein & Vazirani's inner-product protocol, as well as Grover's search algorithm.

In the second part we consider Paley's construction of Hadamard matrices to design a more specific problem that uses the Legendre symbol χ (which indicates if an element of a finite field $\mathbb{F}_{p^k}$ is a quadratic residue or not). It is shown how for a shifted Legendre function $f_s(x) = \chi(x+s)$, the unknown $s \in \mathbb{F}_{p^k}$ can be obtained exactly with only two quantum calls to f_s. This is in sharp contrast with the observation that any classical, probabilistic procedure requires at least $k \log p$ queries to solve the same problem.

5.1 Combinatorics, Hadamard and Weighing Matrices

The matrix H associated with the Hadamard transform is—in the context of quantum computation—called the 'Hadamard matrix'. This terminology is perhaps unfortunate because the same term has already been used in combinatorics to cover a much broader concept. (See the 1893 article by Jacques Hadamard[50] for the origin of this term.)

Definition 7 (Hadamard matrix in combinatorics) *A matrix $M \in \{-1,+1\}^{n\times n}$ is called a* Hadamard matrix *if and only if $M \cdot M^T = n \cdot \mathrm{I}_n$, where "$T$" denotes the transpose of a matrix.*

Obviously, when M is a Hadamard matrix, then $\frac{M}{\sqrt{n}} \in \mathsf{U}(n)$ is a unitary matrix . The following two standard results are easy to verify.

- If M is a Hadamard matrix, then the dimension of M will be $1, 2$ or divisible by 4.

- If M_1 and M_2 are Hadamard matrices, then their tensor product $M_1 \otimes M_2$ is a Hadamard matrix as well.

It is a famous open problem whether or not there exists a Hadamard matrix for every dimension $4k$.

The $\mathrm{H}^{\otimes n}$ matrices, which we encountered before, form only a small subset of all the Hadamard matrices that we know in combinatorics. Instead, the matrices $\sqrt{2^n} \cdot \mathrm{H}^{\otimes n}$ should perhaps be called "Hadamard matrices of the Sylvester kind" after the author who first discussed this specific family of matrices.[96]

The properties of Hadamard matrices (especially the above mentioned $4k$-question) is an intensively studied topic in combinatorics, and its complexity is impressive given the simple definition.[33, 51, 85, 86, 93] In 1933, Raymond Paley proved the existence of two families of Hadamard matrices that are very different from Sylvester's 2^n-construction.

Fact 8 (Paley construction I and II) *I: For every prime p with $p = 3 \bmod 4$ and every integer k, there exists a Hadamard matrix of dimension $(p^k+1) \times (p^k+1)$. II: For every prime p with $p = 1 \bmod 4$ and every integer k, there exists a Hadamard matrix of dimension $(2p^k+2) \times (2p^k+2)$.*

Proof: See the original article [76]. □

For here it suffices to say that Paley's construction uses the theory of quadratic residues over finite fields $\mathbb{F}_{p^k}$. We will discuss this topic in Section 5.3 in order to acquire the necessary tools for the construction of the quantum algorithm of Theorem 9.

One can extend the notion of Hadamard matrices by allowing three possible matrix elements $\{-1, +1, 0\}$, while still requiring the $M \cdot M^T \propto \mathrm{I}_n$ restriction. We thus reach the following definition.

Definition 8 (Weighing matrix [33, 85]) *A matrix $M \in \{-1, 0, +1\}^{n \times n}$ is called a* weighing matrix *if and only if $M \cdot M^T = k \cdot \mathrm{I}_n$ for some $0 \leq k \leq n$. The set of such matrices is denoted by $\mathsf{W}(n,k)$.*

By looking at a row of a matrix $M \in \{-1, 0, +1\}^{n \times n}$, we see that $M \cdot M^T = k \cdot I_n$ implies that this row has has $n-k$ zeros, and k entries "$+1$" or "-1". As a result, $\mathsf{W}(n,n)$ are the Hadamard matrices again, whereas $\mathsf{W}(n, n-1)$ are called *conference matrices.* The identity matrix I_n is an example of a $\mathsf{W}(n,1)$ matrix. If $M_1 \in \mathsf{W}(n_1,k_1)$ and $M_2 \in \mathsf{W}(n_2,k_2)$, then their tensor product $M_1 \otimes M_2$ is an element of $\mathsf{W}(n_1n_2, k_1k_2)$. This implies that for every weighing matrix $M \in \mathsf{W}(n,k)$ we have in fact a whole family of matrices $M^{\otimes t} \in \mathsf{W}(n^t, k^t)$, indexed by $t \in \mathbb{N}$.

Example 1

$$\begin{pmatrix} +1 & +1 & +1 & 0 \\ +1 & -1 & 0 & +1 \\ +1 & 0 & -1 & -1 \\ 0 & +1 & -1 & +1 \end{pmatrix}^{\otimes t} \quad \textit{is a } \mathsf{W}(4^t, 3^t) \textit{ weighing matrix.}$$

The observation that for every $M \in \mathsf{W}(n,k)$ the matrix $\frac{1}{\sqrt{k}} \cdot M \in \mathsf{U}(n)$ is a unitary matrix makes the connection between combinatorics and quantum computation that we explore in this chapter. In the next section we will see how the mutually orthogonal basis of such a matrix can be used for a query efficient quantum algorithm. The classical lower bound for the same problem is proven using standard, decision tree arguments.

5.2 Quantum Algorithms for Weighing Matrices

In this section we will describe a general weighing-matrix-problem and its quantum solution. But before doing so, we first mention the following state-construction lemma which follows directly from earlier results on Grover's search algorithm.

Lemma 6 (State construction lemma) *Let $f : \{1,\dots,n\} \to \{-1,0,+1\}$ be a black-box function. If we know that k of the function values are "$+1$" or "-1", and the remaining $n-k$ entries are "0", then the preparation of the state*

$$|f\rangle \ = \ \frac{1}{\sqrt{k}} \sum_{i=1}^{n} f(i)|i\rangle,$$

requires no more than $\lceil \frac{\pi}{4}\sqrt{\frac{n}{k}} \rceil + 1$ quantum evaluations of the black-box function f. When $k = n$, a single query is sufficient.

Proof: First, we use the amplitude amplification process of Grover's search algorithm [48] to create, *exactly,* the state

$$\frac{1}{\sqrt{k}} \sum_{\substack{i=1 \\ f(i)\neq 0}}^{n} |i\rangle$$

with $\leq \lceil \frac{\pi}{4}\sqrt{\frac{n}{k}} \rceil$ queries to f. (See the article by Boyer *et al.* [25] for a derivation of this upper bound. Obviously, no queries are required if $k = n$.) After that, following Fact 2, one additional f-call is sufficient to insert the proper amplitudes, yielding the desired state $|f\rangle$. □

We will now define the central problem of this chapter, which assumes the existence of a weighing matrix.

Definition 9 (Weighing matrix problem) *Let M be a $\mathsf{W}(n,k)$ weighing matrix. Define a set of n functions $f_s^M : \{1,\dots,n\} \to \{-1,0,+1\}$ for every $s \in \{1,\dots,n\}$ by*

$$f_s^M(i) \ = \ M_{si}.$$

Given a function f_s^M in the form of a black-box, we want to calculate the parameter s. The (probabilistic) query complexity of the weighing matrix problem is the minimum number of calls to the function f that is necessary to determine the value s (with high probability).

With the quantum protocol of Lemma 6 we can solve this problem in a straightforward way.

Theorem 8 (Quantum algorithm for the weighing matrix problem) *Given a matrix $M \in \mathsf{W}(n,k)$ with the corresponding query problem of Definition 9, there exists a quantum algorithm that exactly determines s with $\lceil \frac{\pi}{4}\sqrt{\frac{n}{k}} \rceil + 1$ queries to f^M_s. (When $n = k$, the problem can be solved with one query to the function.)*

Proof: First, prepare the state $|f^M_s\rangle = \frac{1}{\sqrt{k}}\sum_{i=1}^n f^M_s(i)|i\rangle$ with $\lceil \frac{\pi}{4}\sqrt{\frac{n}{k}} \rceil + 1$ queries to the function f. Then, measure the state in the basis spanned by the vectors $|f^M_1\rangle, |f^M_2\rangle$, $\dots$,$|f^M_n\rangle$. Because M is a weighing matrix, this basis is orthogonal and hence the outcome of the measurement gives us the value s (via the outcome f^M_s) without error. □

For every possible weighing matrix, this result establishes a separation between the quantum and the classical query complexity of the problem, as is shown by the following classical lower bound.

Lemma 7 (Classical lower bounds for the weighing matrix problem) *Consider the problem of Definition 9 for a weighing matrix $M \in \mathsf{W}(n,k)$. Let d be the number of queries used by a classical algorithm that recovers s with an error probability of ε. Then, this query complexity is bounded from below by*

$$\begin{aligned} d &\geq \log_3(1-\varepsilon) + \log_3 n, \\ d &\geq (1-\varepsilon)\tfrac{n}{k} - \tfrac{1}{k}, \\ d &\geq \log((1-\varepsilon)n + n - k) - \log(n-k+1). \end{aligned}$$

(For the case where $k = n$, this lower bound equals $d \geq \log(1-\varepsilon) + \log n$.)

Proof: We will prove these bounds by considering the decision trees that describe the possible classical protocols. The procedure starts at the root of the tree and this node contains the first index i that the protocol queries to the function f. Depending on the outcome $f(i) \in \{-1, 0, +1\}$, the protocol follows one of the (three) outgoing edges to a new node x, which contains the next query index i_x. This routine is repeated until the procedure reaches one of the leaves of the tree. At that point, the protocol guesses which function it has been querying. With this representation, the depth of such a tree reflects the number of queries that the protocol uses, while the number of leaves (nodes without outgoing edges) indicates how many different functions the procedure can distinguish.

For a probabilistic algorithm with error probability ε, we need to have decision trees with at least $(1-\varepsilon)n$ leaves. Because the number of outgoing edges cannot be bigger than 3, a tree with depth d has maximally 3^d leaves. This proves the first lower bound via $3^d \geq (1-\varepsilon)n$.

For the second and third bound we have to analyze the maximum size of the optimal decision tree as it depends on the values k and n. We know that for every index i_x, there are only k different functions with $f(i_x) \neq 0$. This implies that at every node x the joint number of leaves of the two subtrees (associated with the outcomes $f(i_x) = -1$ and $+1$) cannot be bigger than k. Hence, by considering the path (starting from the root) along the edges that correspond to the answers $f(i_x) = 0$, we see that a decision tree with d queries, can distinguish no more than $dk + 1$ functions. (Consider for example the case where $k = 1$.) Similarly, we can use the observation that there are exactly $n - k$ functions with $f(i_x) = 0$ for every node x. This tells us that a tree with depth d has a maximum number of leaves of $2^d + (2^d - 1)(n - k)$. □

The above bounds simplify significantly when we express them as functions of (big enough) n. This gives us the following table (note that the quantum complexity holds for the exact solution with $\varepsilon = 0$):

k	quantum upper bound	classical lower bound
$o(n)$	$\lceil \frac{\pi}{4}\sqrt{\frac{n}{k}} \rceil + 1$	$(1-\varepsilon)\frac{n}{k} - O(1)$
$\Theta(n)$	$O(1)$	$\log_3 n + \log_3(1-\varepsilon)$
n	1	$\log n + \log(1-\varepsilon)$

Note that the n-dimensional identity matrix is a $\mathsf{W}(n,1)$ weighing matrix, and that for this I_n the previous theorem and lemma are just a rephrasing (with $k = 1$) of the results on Grover's search algorithm for exactly one matching entry. The algorithm of Bernstein & Vazirani is also captured by the above as the case where k has the maximum value $k = n$ (with the weighing matrices $(\sqrt{2} \cdot \mathrm{H})^{\otimes t} \in \mathsf{W}(2^t, 2^t)$). Hence we can think of those two algorithms as the extreme instances of the more general weighing matrix problem.

As we phrased it, a weighing matrix $M \in \mathsf{W}(n,k)$ gives only a input-size specific problem for which there is a classical/quantum separation, but not a problem that is defined for every input size N, as is more customary. We know, however, that for every such matrix M, the tensor products $M^{\otimes t}$ are also $W(n^t, k^t)$ weighing matrices (for all $t \in \mathbb{N}$). We therefore have the following direct consequence of our results.

Lemma 8 *Every weighing matrix $M \in \mathsf{W}(n,k)$ leads—via the set of matrices $M^{\otimes t} \in \mathsf{W}(n^t, k^t)$—to a weighing matrix problem for $N = n^t$ and $K = k^t = N^{\log_n k}$. By defining $\gamma = 1 - \log_n k$ we have, for every suitable N, a quantum algorithm with query complexity $\frac{\pi}{4}\sqrt{N^\gamma}$ for which there is a classical, probabilistic lower bound of $(1-\varepsilon) \cdot N^\gamma$.*

Example 2 *Using the $\mathsf{W}(4^t, 3^t)$ weighing matrices of Example 1, we have $\gamma = 1 - \frac{1}{2}\log 3 \approx 0.21$, and hence a quantum algorithm with query complexity $\frac{\pi}{4}N^{0.10\dots}$. The corresponding classical probabilistic, lower bound of this problem is $(1-\varepsilon) \cdot N^{0.21\dots}$.*

A legitimate objection against the weighing-matrix-problem is that it does not seem to be very useful (besides the known boundary cases $k = 1$ and $k = n$). In order to obtain more natural problems one can try to look into the specific structure that constitutes the weighing matrix or matrices. An example of such an approach will be given in the next two sections via Paley's construction of Hadamard matrices. We will see how this leads to the definition of a problem about quadratic residues of finite fields with a quantum solution that is more efficient than any classical protocol.

5.3 Quadratic Residues of Finite Fields

This section describes some standard results about quadratic residues and Legendre symbols over finite fields. Readers familiar with this topic can safely skip the next paragraphs and continue with Section 5.6. For more background information one can look up references like [32] or [57].

5.4 Finite Field Factoids

From now on p denotes an odd prime. It is known that there always exists a generator ζ for the multiplicative group $\mathbb{F}^\star_{p^k} = \mathbb{F}_{p^k} \setminus \{0\}$. [32, 57] This means that the sequence $\zeta, \zeta^2, \zeta^3, \dots$ will generate all non-zero elements of $\mathbb{F}_{p^k}$. As this is a set of size $p^k - 1$, it follows that $\zeta^{p^k} = \zeta$, and hence $\zeta^{(p^k-1)} = 1$. Hence we have the equality

$$\zeta^i = \zeta^j \quad \text{if and only if} \quad i = j \bmod (p^k - 1) \tag{5.1}$$

for every integer i and j.

We now turn our attention to the definition of the *generalized Legendre symbol*.[32]

Definition 10 (Legendre symbol over finite fields) *For every finite field $\mathbb{F}_{p^k}$, with p an odd prime, the Legendre symbol-function $\chi : \mathbb{F}_{p^k} \to \{-1, 0, +1\}$ indicates if a number is a quadratic residue or not, and is thus defined by*

$$\chi(x) \quad := \quad \begin{cases} 0 & \textit{if } x = 0 \\ +1 & \textit{if } \exists y \neq 0 : y^2 = x \\ -1 & \textit{if } \forall y : y^2 \neq x. \end{cases}$$

By Equation 5.1, the quadratic expression $(\zeta^j)^2 = \zeta^{2j} = \zeta^i$ is correct if and only if $2j = i \bmod p^k - 1$. As p is odd, $p^k - 1$ will be even, and hence there can only exists a j with $(\zeta^j)^2 = \zeta^i$ when i is even. Obviously, if i is even, then ζ^j with $j = \frac{i}{2}$ gives a solution to our quadratic equation. This proves that 50% of the elements of $\mathbb{F}^\star_{p^k}$ are a quadratic residue with $\chi(x) = +1$, while the other half has $\chi(x) = -1$. In particular, $\chi(\zeta^i) = (-1)^i$, and hence for the total sum of the function values: $\sum_x \chi(x) = 0$.

5.5 Multiplicative Characters over Finite Fields

The rule $\chi(\zeta^i) \cdot \chi(\zeta^j) = \chi(\zeta^{i+j})$, in combination with $\chi(0) = 0$, shows that the Legendre symbol χ is a *multiplicative character* with $\chi(x) \cdot \chi(y) = \chi(xy)$ for all $x, y \in \mathbb{F}_{p^k}$.

Definition 11 (Multiplicative characters over finite fields) *The function $\chi : \mathbb{F}_{p^k} \to \mathbb{C}$ is a* multiplicative character *if and only if $\chi(xy) = \chi(x)\chi(y)$ for all $x, y \in \mathbb{F}_{p^k}$. The constant function $\chi(x) = 1$ is called the trivial character. (We do not consider the other trivial function $\chi(x) = 0$.)*

See [32, 57] for the usage of multiplicative characters in number theory. They have the following elementary properties, which we present without proof:

- $\chi(1) = 1$,
- for all nonzero x, the value $\chi(x)$ is a $(p^k - 1)$th root of unity,
- if χ is nontrivial, we have $\chi(0) = 0$,
- the inverse of nonzero x obeys $\chi(x^{-1}) = \chi(x)^{-1} = \chi(x)^*$,
- $\sum_x \chi(x) = 0$ for nontrivial χ.

The remainder of this section is used to prove a 'near orthogonality' property, typical for nontrivial characters, which will be the crucial ingredient of the quantum algorithm of the next section.

Lemma 9 (Near orthogonality of shifted characters) *Consider a nontrivial character $\chi : \mathbb{F}_{p^k} \to \mathbb{C}$. For the 'complex inner product' between two χ-s that are shifted by s and $r \in \mathbb{F}_{p^k}$ it holds that*

$$\sum_{x \in \mathbb{F}_{p^k}} \chi(x+r)^*\chi(x+s) \quad = \quad \begin{cases} p^k - 1 & \text{if } s = r \\ -1 & \text{if } s \neq r. \end{cases}$$

Proof: Rewrite

$$\sum_{x \in \mathbb{F}_{p^k}} \chi(x+r)^*\chi(x+s) \quad = \quad \sum_{x \in \mathbb{F}_{p^k}} \chi(x)^*\chi(x+\Delta)$$

with $\Delta = s - r$. If $s = r$ this sum equals $p^k - 1$. Otherwise, we can use the fact that $\chi(x)^*\chi(x+\Delta) = \chi(1 + x^{-1}\Delta) = \chi(\Delta)\chi(\Delta^{-1} + x^{-1})$ (for $x \neq 0$) to reach

$$\sum_{x \in \mathbb{F}_{p^k}} \chi(x)^*\chi(x+\Delta) \quad = \quad \chi(\Delta) \sum_{x \in \mathbb{F}^\star_{p^k}} \chi(\Delta^{-1} + x^{-1}).$$

Earlier we noticed that $\sum_x \chi(x) = 0$, and therefore in the above summation (where the value $x = 0$ is omitted) we have $\sum_x \chi(x^{-1} + \Delta^{-1}) = -\chi(\Delta^{-1})$. This confirms that indeed

$$\chi(\Delta) \sum_{x \in \mathbb{F}^\star_{p^k}} \chi(x^{-1} + \Delta^{-1}) \quad = \quad -1,$$

which finishes the proof. □

We will use this lemma in the setting where the character is the earlier described Legendre symbol.

5.6 The shifted Legendre Symbol Problem

Raymond Paley used the near orthogonality property of the Legendre symbol for the construction of his Hadamard matrices.[76] Here we will use the same property to describe a problem that, much like the above weighing matrix problem, has a gap between its quantum and its classical query complexity. In light of Theorem 8 and Lemma 7 the results of this section are probably not very surprising. Rather, we wish to give an example of how we can borrow the ideas behind the construction of combinatorial objects to design new quantum algorithms. In this case this is done by stating a problem that uses the Legendre symbol over finite fields.

Definition 12 (Shifted Legendre Symbol Problem) *Assume that we have a black-box for a shifted Legendre function $f_s : \mathbb{F}_{p^k} \to \{-1, 0, +1\}$ that obeys*

$$f_s(x) \quad = \quad \chi(x+s),$$

with the—for us unknown—shift parameter $s \in \mathbb{F}_{p^k}$. (Recall Definition 10 for a description of χ.) The task is to determine the value s with a minimum number of calls to the function f.

First we will prove a lower bound for the classical query complexity of this problem. This proof is almost identical to the lower bounds of Lemma 7 for the weighing matrix problem.

Lemma 10 (Classical lower bound for the SLS problem) *Assume a classical algorithm that tries to solve the shifted Legendre symbol problem over a finite field $\mathbb{F}_{p^k}$. To determine the requested value s with a maximum error rate ε, requires more than $k \log p + \log(1-\varepsilon) - 1$ queries to the function f_s.*

Proof: For every index i_x there is exactly one function with $f(i_x) = 0$. For the decision tree of a classical protocol this implies that every node x can only have two proper subtrees (corresponding to the answers $f(i) = 1$ and -1) and one deciding leaf (the case $f_{(-i)}(i) = 0$). Hence, a decision tree of depth d can distinguish no more than $2^{d+1} - 1$ different functions. In order to be able to differentiate between $(1-\varepsilon)p^k$ functions, we thus need a depth d of at least $\log((1-\varepsilon)p^k - 1)$. □

The next theorem shows us how—with a quantum computer—we can recover s exactly with only two queries.

Theorem 9 (Two Query Quantum Algorithm for the SLS Problem) *For any finite field $\mathbb{F}_{p^k}$, the problem of Definition 12 can be solved exactly with two quantum queries to the black-box function f_s.*

Proof: We exhibit the quantum algorithm in detail. We start with the superposition

$$|\text{start}\rangle \quad = \quad \frac{1}{\sqrt{p^k+1}} \left(\sum_{x \in \mathbb{F}_{p^k}} |x\rangle|0\rangle \right) + \frac{1}{\sqrt{p^k+1}} |\text{dummy}\rangle|1\rangle.$$

(The reason for the "dummy" part of state that we use will be clear later in the analysis.) The first oracle call is used to calculate the different χ values for the non-dummy states, giving

$$\begin{aligned}|\text{start}\rangle \quad & \underset{f_s}{\longmapsto} \quad \frac{1}{\sqrt{p^k+1}}\left(\sum_{x\in\mathbb{F}_{p^k}} |x\rangle|f_s(x)\rangle\right) + \frac{1}{\sqrt{p^k+1}}|\text{dummy}\rangle|1\rangle \\ & = \quad \frac{1}{\sqrt{p^k+1}}\left(\sum_{x\in\mathbb{F}_{p^k}} |x\rangle|\chi(x+s)\rangle\right) + \frac{1}{\sqrt{p^k+1}}|\text{dummy}\rangle|1\rangle.\end{aligned}$$

At this point, we measure the rightmost register to see if it contains the value "zero". If this is indeed the case (probability $\frac{1}{p^k+1}$), the state has collapsed to $|-s\rangle|0\rangle$ which directly gives us the desired answer s. Otherwise, we continue with the now reduced state

$$\frac{1}{\sqrt{p^k}}\left(\sum_{x\in\mathbb{F}_{p^k}\setminus\{-s\}} |x\rangle|\chi(x+s)\rangle\right) + \frac{1}{\sqrt{p^k}}|\text{dummy}\rangle|1\rangle, \tag{5.2}$$

on which we apply a conditional phase change (depending on the χ values in the rightmost register). We finish the computing by 'erasing' this rightmost register with a second call to f_s. (For the dummy part, we just reset the value to "zero".) This gives us the final state ψ, depending on s, of the form

$$|\psi_s\rangle|0\rangle \quad = \quad \frac{1}{\sqrt{p^k}}\left(\sum_{x\in\mathbb{F}_{p^k}} \chi(x+s)|x\rangle\right)|0\rangle + \frac{1}{\sqrt{p^k}}|\text{dummy}\rangle|0\rangle.$$

(Notice how the $\chi(x+s)$ amplitude is zero for the missing entry $x = -s$ in the summation over $\mathbb{F}_{p^k}$.)

What is left to show is that $\{|\psi_s\rangle | s \in \mathbb{F}_{p^k}\}$ forms a set of orthogonal vectors. Lemma 9 tells us that for the inner product between two states ψ_s and ψ_r it holds that

$$\begin{aligned}\langle\psi_r|\psi_s\rangle \quad & = \quad \frac{1}{p^k}\left(\sum_{x\in\mathbb{F}_{p^k}} \chi(x+r)^*\chi(x+s)\right) + \frac{1}{p^k} \\ & = \quad \begin{cases} 1 & \text{if } s = r \\ 0 & \text{if } s \neq r.\end{cases}\end{aligned}$$

In other words, the states ψ_s for $s \in \mathbb{F}_{p^k}$ are mutually orthogonal. Hence, by measuring the final state in the ψ-basis, we can determine without error the shift factor $s \in \mathbb{F}_{p^k}$ after only two oracle calls to the function f_s. □

More recently, Peter Høyer has shown the existence of a one query protocol for the same problem.[private communication]

The above algorithm only reduces the *query complexity* to f_s. The *time complexity* of the protocol is another matter, as we did not explain how to perform the final measurement along the ψ axes in a time-efficient way. In a recent article [37] it is shown how one can implement the unitary mapping

$$|s\rangle \quad \longleftrightarrow \quad \frac{1}{\sqrt{p^k}}\left(\sum_{x\in\mathbb{F}_{p^k}} \chi(x+s)|x\rangle\right) + \frac{1}{\sqrt{p^k}}|\text{dummy}\rangle$$

with an efficient quantum circuit of depth polylog(p^k).

5.7 Conclusion

We have established a connection between the construction of weighing matrices in combinatorics, and the design of new quantum algorithms. It was shown how every weighing matrix leads to a query problem that has a more efficient quantum solution than is possible classically.

Using the structure of quadratic residues over finite fields, we gave an explicit example of a task with constant quantum query complexity, but logarithmic classical query complexity.

The implicit goal of this chapter was to suggest new possibilities for the construction of useful quantum algorithms. Other results on Hadamard matrices that are especially interesting in this context are, for example, the complex Hadamard matrices of Turyn[100] and the Hadamard matrices of the dihedral group type[61, 90].

Chapter 6

Self-Testing of Quantum Gates

This chapter concerns the problem how to test the behavior of a quantum gate. If we think that we have a Hadamard gate H, how can we be sure that H behaves indeed correctly on all possible input qubits $\alpha|0\rangle + \beta|1\rangle$? How can we test this without having to rely on other quantum mechanical components that can be equally unreliable? These questions concern the *self-testability* of quantum gates.

We show how some gates or families of gates are self-testable whereas others are not. These self-testing procedures are also "robust". By this we mean that the error during the test-procedure and the error of the gate are proportional: If we detect a small error during the testing-procedure, then this will always correspond to a small error in the gate. The method is also extended to two-qubit gates.

6.1 Introduction

We consider the design of self-testers for quantum gates. A self-tester for the gates $\boldsymbol{F}_1, \ldots, \boldsymbol{F}_m$ is a classical procedure that, given any gates $\boldsymbol{G}_1, \ldots, \boldsymbol{G}_m$, decides with high probability if each $\boldsymbol{G}_i$ is close to $\boldsymbol{F}_i$. This decision has to rely only on measuring in the computational basis the effect of iterating the gates on the classical states. It turns out that instead of individual gates, we can only design procedures for families of gates. To achieve our goal we borrow some elegant ideas of the theory of program testing: we characterize the gate families by specific properties, we develop a theory of robustness for them, and show that they lead to self-testers. In particular we prove that the universal and fault-tolerant set of gates consisting of a Hadamard gate, a CNot gate, and a phase rotation gate of angle $\frac{\pi}{4}$ is self-testable.

The idea of self-testing in quantum devices is implicit in the work of Adleman, Demarrais and Huang[1]. They have developed a procedure by which a quantum Turing machine is able to estimate its internal angle by its own means under the hypothesis that the machine is unitary. In the context of quantum cryptography Mayers and Yao[71] have designed tests for deciding if a photon source is perfect. These tests guarantee that if source passes them then it is adequate for the security of the Bennett-Brassard[20] quantum key distribution protocol.

Here we develop the theory of self-testing of quantum gates by classical procedures. Given a completely positive super operator (CPSO) $\boldsymbol{G}$ for n qubits, and a family $\mathcal{F}$ of unitary CPSOs, we would like to decide if $\boldsymbol{G}$ belongs to $\mathcal{F}$. Intuitively, a self-tester is a procedure that answers the question "$\boldsymbol{G} \in \mathcal{F}$?" by interacting with the CPSO $\boldsymbol{G}$ in a purely classical way. More precisely, it will be a probabilistic algorithm that is able to access $\boldsymbol{G}$ as a black box in the following sense: it can prepare the classical states $w \in \{0,1\}^n$, iterate $\boldsymbol{G}$ on these states, and afterwards, measure in the computational basis. The access must be seen as a whole, performed by a specific, experimental oracle for $\boldsymbol{G}$: once the basis state w and the number of iterations k have been specified, the program in one step gets back one of the possible probabilistic outcomes of measuring the state of the system after $\boldsymbol{G}$ is iterated k-times on w. The intermediate quantum states of this process cannot be used by the program, which cannot perform any other quantum operations either. For $0 \leq \delta_1 \leq \delta_2$, such an algorithm will be a (δ_1, δ_2)-tester for $\mathcal{F}$ if for every CPSO $\boldsymbol{G}$, whenever the distance of $\boldsymbol{G}$ and $\mathcal{F}$ is at most δ_1 (in some norm), it accepts with high probability, and whenever the same distance is greater than δ_2, it rejects with high probability, where the probability is taken over the measurements performed by the oracle and by the internal coin tosses of the algorithm. Finally we will say that $\mathcal{F}$ is *testable* if for every $\delta_2 > 0$, there exists $0 < \delta_1 \leq \delta_2$ such that there exists a (δ_1, δ_2)-tester for $\mathcal{F}$. These definitions can be extended to several classes of CPSOs.

The study of self-testing programs is a well-established research area which was initiated by the work of Blum, Luby and Rubinfeld[24], Rubinfeld[79], Lipton[69] and Gemmel *et al.* [47]. The purpose of a self-tester for a function family is to detect by simple means if a program which is accessible as an oracle computes a specific function from the given family. This clearly inspired the definition of our self-testers which have the peculiarity that they should test quantum objects that they can access only in some restricted manner. The analogy with self-testing does not stop with the definition. One of the main tools in self-testing of function families is the characterization of these families by robust properties. Informally, a property is robust if whenever a function satisfies the property approximately, then it is close to a function which satisfies it exactly. The concept of robustness was introduced and its implication for self-testing was first studied by Rubinfeld and Sudan[81] and by Rubinfeld[80]. It will play a crucial role in our case as well.

We note in the Preliminaries that for any real ϕ the states $|1\rangle$ and $\mathrm{e}^{\mathrm{i}\phi}|1\rangle$ are experimentally indistinguishable. This implies that if both the input states and the measurement basis vectors are the classical states $|0\rangle$ and $|1\rangle$, then there are 'families' of CPSOs which are mutually indistinguishable. For example, let the CPSO $\mathbf{H}$ be the well-known Hadamard gate with

$$|0\rangle \mapsto \tfrac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \quad \text{and} \quad |1\rangle \mapsto \tfrac{1}{\sqrt{2}}(|0\rangle - |1\rangle),$$

and let $\mathbf{H}_\phi$ be the same gate expressed in the basis $(|0\rangle, \mathrm{e}^{\mathrm{i}\phi}|1\rangle)$, hence

$$|0\rangle \mapsto \tfrac{1}{\sqrt{2}}(|0\rangle + \mathrm{e}^{\mathrm{i}\phi}|1\rangle) \quad \text{and} \quad |1\rangle \mapsto \tfrac{1}{\sqrt{2}}(\mathrm{e}^{-\mathrm{i}\phi}|0\rangle - |1\rangle),$$

for $\phi \in [0, 2\pi)$. Any experiment that uses $\mathbf{H}$ and starts with the state $|0\rangle$ or $|1\rangle$ will produce the outcomes "0" and "1" with the same probabilities as the same experiment with the $\mathbf{H}_\phi$ gate. Thus, no experiment that uses this quantum gate alone can distinguish $\mathbf{H}$ from $\mathbf{H}_\phi$. Indeed, as stated later in Fact 15, we will have to consider a testable 'family' $\mathcal{F} = \{\mathbf{H}_\phi | \phi \in [0, 2\pi)\}$ containing all $\mathbf{H}_\phi$ gates.

The main result is Theorem 15 which states that for several sets of unitary CPSOs, in particular, the Hadamard gates family, Hadamard gates together with CNot gates, and Hadamard gates with CNot and phase rotation gates of angle $\pm\frac{\pi}{4}$, are testable. This last family is of particular importance since every triplet in the family forms a universal and fault-tolerant set of gates for quantum computation[26].

For the proof we will define the notion of experimental equations which are functional equations for CPSOs corresponding to the properties of the quantum gate that a self-tester can approximately test. These tests are done via the interaction with the experimental oracle. The proof itself contains three parts. In Theorems 10, 11, and 12 we will exhibit experimental equations for the families of unitary CPSOs we want to characterize. In Theorem 13 we will show that actually all experimental equations are robust; in fact, the distance of a CPSO from the target family is polynomially related to the error tolerated in the experimental equations. Finally Theorem 14 gives self-testers for CPSO families which are characterized by a finite set of robust experimental equations.

In some cases, we are able to calculate explicitly the polynomial bound in the robustness of experimental equations. Such a result will be illustrated in Lemma 14 for the equations characterizing the Hadamard family $\{\mathbf{H}_\phi\}$.

Technically, these results will be based on the representation of one-qubit states and CPSOs in $\mathbb{R}^3$, where they are respectively vectors in the unit ball of $\mathbb{R}^3$, and particular affine transformations. This correspondence is known as the Bloch Ball representation.

6.2 The Bloch Ball representation

Specific for the one-qubit case there is a very appealing way of describing both the states and its unitary transformations in 3 dimensional Euclidean space, known as the Bloch ball picture. This representation relies on the isomorphism between the group $\mathsf{U}(2)/\mathsf{U}(1)$ and the special rotation group $\mathsf{SO}(3)$, the set of 3×3 orthogonal matrices with determinant 1. This allows us to view one-qubit states as vectors in the unit ball of $\mathbb{R}^3$, and unitary superoperators as rotations on $\mathbb{R}^3$. We will now describe exactly this correspondence.

The *Bloch Ball* $\mathcal{B}$ (respectively *Bloch Sphere* $\mathcal{S}$) is the ball (sphere) with radius 1 of the Euclidean affine space $\mathbb{R}^3$. Any point $\vec{u} \in \mathbb{R}^3$ determines a vector with the same coordinates which we will also denote by $\vec{u}$. The inner product of $\vec{u}$ and $\vec{v}$ will be denoted by $(\vec{u}, \vec{v})$, and the Euclidean norm of $\vec{u}$ by $||\vec{u}||$.

Using spherical coordinates, we can characterize each point $\vec{u} \in \mathbb{R}^3$ by its norm $r \geq 0$, its latitude $\theta \in [0, \pi]$, and its longitude $\phi \in [0, 2\pi)$. The *latitude* is the angle between the z-axis and the vector $\vec{u}$, and the *longitude* is the angle between the x-axis and the orthogonal projection of $\vec{u}$ in the plane defined by $z = 0$. If $\vec{u} = (x, y, z)^T$, then these parameters satisfy $x = r \sin\theta \cos\phi$, $y = r \sin\theta \sin\phi$ and $z = r\cos\theta$. For every $(x, y, z)^T \in \mathcal{B} \subset \mathbb{R}^3$ there exists a unique density matrix such that

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \frac{1}{2}\overrightarrow{\begin{bmatrix} 1+z & x-\mathrm{i}y \\ x+\mathrm{i}y & 1-z \end{bmatrix}} = \vec{\rho}.$$

This mapping is a bijection that also obeys

$$\begin{aligned} \overrightarrow{\rho(p,\alpha)} &= \overrightarrow{\begin{bmatrix} p & \alpha^* \\ \alpha & 1-p \end{bmatrix}} \\ &= \begin{pmatrix} \alpha+\alpha^* \\ \mathrm{i}\alpha^* - \mathrm{i}\alpha \\ 2p-1 \end{pmatrix}. \end{aligned}$$

In this formalism, the pure states are nicely characterized in $\mathcal{B}$ by their norm.

Fact 9 *A density matrix ρ represents a pure state if and only if $\vec{\rho} \in \mathcal{S}$, that is, $||\vec{\rho}|| = 1$.*

Also, if $\theta \in [0, \pi]$ and $\phi \in [0, 2\pi)$ are respectively the latitude and the longitude of $\vec{\psi} \in \mathcal{S}$, then the corresponding density matrix represents a pure state and satisfies $|\psi\rangle = \cos(\frac{\theta}{2})|0\rangle + \sin(\frac{\theta}{2})\mathrm{e}^{\mathrm{i}\phi}|1\rangle$. Observe that the pure states $|\psi\rangle$ and $|\phi\rangle$ are orthogonal if and only if $\vec{\psi} = -\vec{\phi}$. We will use the following notation for the six pure states along the x, y and z axes: $|\zeta_x^{\pm}\rangle = \frac{1}{\sqrt{2}}(|0\rangle \pm |1\rangle)$, $|\zeta_y^{\pm}\rangle = \frac{1}{\sqrt{2}}(|0\rangle \pm \mathrm{i}|1\rangle)$, $|\zeta_z^{+}\rangle = |0\rangle$, and $|\zeta_z^{-}\rangle = |1\rangle$, with the respective coordinates $(\pm 1, 0, 0)$, $(0, \pm 1, 0)$ and $(0, 0, \pm 1)$ in $\mathbb{R}^3$.

For each CPSO $\boldsymbol{G}$, there exists a unique affine transformation $\vec{\boldsymbol{G}}$ over $\mathbb{R}^3$, which maps the ball $\mathcal{B}$ into $\mathcal{B}$ and is such that, for all density matrices ρ, $\vec{\boldsymbol{G}}(\vec{\rho}) = \overrightarrow{\boldsymbol{G}(\rho)}$. Unitary superoperators have a nice characterization in $\mathcal{B}$.

Fact 10 *The map between* $\mathsf{U}(2)/\mathsf{U}(1)$ *and* $\mathsf{SO}(3)$ *that sends A to $\vec{\boldsymbol{A}}$, is an isomorphism.*

For $\alpha \in (-\pi, \pi]$, $\theta \in [0, \frac{\pi}{2}]$, and $\phi \in [0, 2\pi)$, we will define the unitary transformation $\mathrm{R}_{\alpha,\theta,\phi}$ over $\mathcal{H}_2$. If $|\psi\rangle = \cos(\frac{\theta}{2})|0\rangle + \mathrm{e}^{\mathrm{i}\phi}\sin(\frac{\theta}{2})|1\rangle$ and $|\psi^{\perp}\rangle = \sin(\frac{\theta}{2})|0\rangle - \mathrm{e}^{\mathrm{i}\phi}\cos(\frac{\theta}{2})|1\rangle$ then by definition $\mathrm{R}_{\alpha,\theta,\phi}|\psi\rangle = |\psi\rangle$ and $\mathrm{R}_{\alpha,\theta,\phi}|\psi^{\perp}\rangle = \mathrm{e}^{\mathrm{i}\alpha}|\psi^{\perp}\rangle$. If $\boldsymbol{A}$ is a unitary superoperator then we have $\boldsymbol{A} = \mathbf{R}_{\alpha,\theta,\phi}$ for some α, θ, and ϕ. In $\mathbb{R}^3$ the transformation $\vec{\mathbf{R}}_{\alpha,\theta,\phi}$ is the rotation of angle α whose axis cuts the sphere $\mathcal{S}$ in the opposite points $\vec{\psi}$

and $-\vec{\psi} = \overrightarrow{\psi^\perp}$. Note that for $\theta = 0$ the CPSO $\mathbf{R}_{\alpha,0,\phi}$ does not depend on ϕ. We will denote this phase rotation by $\mathbf{R}_\alpha$.

The affine transformation in $\mathcal{B}$ which corresponds to the Von Neumann measurement in the computational basis is the orthogonal projection to the z-axis. Therefore it maps $\vec{\rho} = (x, y, z)$ into $(0, 0, z)$, the point which corresponds to the density matrix $\frac{1+z}{2}|0\rangle\langle 0| + \frac{1-z}{2}|1\rangle\langle 1|$. Thus $\mathrm{Prob}^0[\rho] = \frac{1+z}{2}$.

6.3 Norm and Distance

Consider the space of $n \times n$ dimensional, complex valued matrices. We define the *trace norm* for this space as follows.

Definition 13 (Trace norm) *Let $A \in M_n(\mathbb{C})$ be a complex valued matrix, the trace norm is defined by*

$$\begin{aligned} ||A||_{\mathrm{tr}} &:= \mathrm{tr}\left(\sqrt{A \cdot A^*}\right) \\ &= \sum_{i=1}^{n} \sigma_i, \end{aligned}$$

where A^* is the conjugate transpose of A and $\sigma_1, \sigma_2, \ldots$ are the singular values *of A. (See the appendix of thesis or [54] for more information on these terms.)*

Definition 14 (Euclidean norm) *For $A \in M_n(\mathbb{C})$ a complex valued matrix, its* Euclidean norm *is define by*

$$||A||_2 := \sqrt{\sum_{i,j=1}^{n} |A_{ij}|^2} \tag{6.1}$$

$$= \sum_{i=1}^{n} \sigma_i^2, \tag{6.2}$$

with σ_i the singular values of the matrix A.

Both norms are *matrix norms* because they obey the following properties (see Chapter 5 in [54] for much more on this topic):

1. nonnegative: $||A|| \geq 0$
2. positive: $||A|| = 0$ if and only if $A = 0$
3. homogeneous: $||\alpha A|| = |\alpha| \cdot ||A||$ for all $\alpha \in \mathbb{C}$
4. triangle inequality: $||A + B|| \leq ||A|| + ||B||$

5. submultiplicative: $||AB|| \leq ||A|| \cdot ||B||$.

In addition, for the tensor product between two matrices, we also have the equality

- $||A \otimes B|| = ||A|| \cdot ||B||$.

A very useful relation between the trace and the Euclidean norm is easily established by the inequalities $\sqrt{\sum_i \sigma_i^2} \leq \sum_i \sigma_i \leq \sqrt{n}\sqrt{\sum_i \sigma_i^2}$ with the summation over the n singular values. We thus have

$$||A||_2 \quad \leq \quad ||A||_{\mathrm{tr}} \quad \leq \quad \sqrt{n}||A||_2 \tag{6.3}$$

for all $A \in M_n(\mathbb{C})$.

The trace norm has several advantages when we consider the difference between two quantum states ρ_1 and ρ_2. Given a measurement setting $\mathcal{P} = \{P_i\}$ (with the normalization restriction $\sum_i P_i^* P_i = I$), a density matrix ρ induces a probability distribution $\mathrm{Prob}(P_i|\rho)$ over the different projectors P_i. It can be shown that in this setting, the trace norm of the difference $\rho_1 - \rho_2$ is the *maximal total variation distance* between the two states:

$$||\rho_1 - \rho_2||_{\mathrm{tr}} \quad = \quad \max_{\mathcal{P}} \left(\sum_{P_i \in \mathcal{P}} |\mathrm{Prob}(\text{“}\rho_1 = P_i\text{”}) - \mathrm{Prob}(\text{“}\rho_2 = P_i\text{”})| \right),$$

where the maximization is taken over all measurement settings $\mathcal{P}$. This result suggests that the expression $||\rho_1 - \rho_2||_{\mathrm{tr}}$ is a natural way of measuring the difference between the two states ρ_1 and ρ_2. The following Fact strengthens this belief.

Fact 11 *The trace-norm distance between two qubit states ρ_1 and ρ_2 is identical to the Euclidean distance between $\overrightarrow{\rho_1}$ and $\overrightarrow{\rho_2}$ in the Bloch ball representation:*

$$||\rho_1 - \rho_2||_{\mathrm{tr}} \quad = \quad ||\overrightarrow{\rho_1} - \overrightarrow{\rho_2}||_2.$$

For the density matrices $\rho(p,\alpha)$ and $\rho(q,\beta)$ this value is explicitly expressed by

$$||\rho(p,\alpha) - \rho(q,\beta)||_{\mathrm{tr}} \quad = \quad 2\sqrt{(p-q)^2 + |\alpha - \beta|^2}.$$

6.4 Norms on Superoperators

Definition 15 (Trace Induced Superoperator Norm) *For superoperators, the norm induced by the trace norm is defined as*

$$|||\boldsymbol{G}|||_{\mathrm{tr}} \quad := \quad \max_{X \neq 0} \left\{ \frac{||\boldsymbol{G}(X)||_{\mathrm{tr}}}{||X||_{\mathrm{tr}}} \right\}.$$

We will denote by $\mathrm{dist}_{\mathrm{tr}}$ the natural induced distance by the norm $\|\cdot\|_{\mathrm{tr}}$:

$$\begin{aligned}\mathrm{dist}_{\mathrm{tr}}(\boldsymbol{F},\boldsymbol{G}) &:= \|\boldsymbol{F}-\boldsymbol{G}\|_{\mathrm{tr}}\\ &= \max_{X\neq 0}\left\{\frac{\|\boldsymbol{F}(X)-\boldsymbol{G}(X)\|_{\mathrm{tr}}}{\|X\|_{\mathrm{tr}}}\right\}.\end{aligned}$$

As $\|\cdot\|_{\mathrm{tr}}$ is a norm, the usual properties like $\|\boldsymbol{F}+\boldsymbol{G}\|_{\mathrm{tr}} \leq \|\boldsymbol{F}\|_{\mathrm{tr}} + \|\boldsymbol{G}\|_{\mathrm{tr}}$ and $\|\alpha\boldsymbol{F}\|_{\mathrm{tr}} = |\alpha|\|\boldsymbol{F}\|_{\mathrm{tr}}$ hold. Furthermore, we also have for every power $k\in\mathbb{N}$:

$$\mathrm{dist}_{\mathrm{tr}}(\boldsymbol{F}^k,\boldsymbol{G}^k) \leq k\cdot\mathrm{dist}_{\mathrm{tr}}(\boldsymbol{F},\boldsymbol{G}),$$

which we will use later in this chapter.

6.5 Properties of CPSOs

Here we will establish the properties of CPSOs that we will need for the characterization of our CPSO families.

Fact 12 (Monotonicity of the trace-norm distance [82]) *Let $\boldsymbol{G}$ be a completely positive, trace preserving transformation (a CPSO). The trace-norm distance between two states is non-increasing under the action of $\boldsymbol{G}$:*

$$\|\boldsymbol{G}(\rho_1)-\boldsymbol{G}(\rho_2)\|_{\mathrm{tr}} \leq \|\rho_1-\rho_2\|_{\mathrm{tr}},$$

for all quantum states ρ_1 and ρ_2.

Proof: First we rewrite the Hermitian difference matrix $\rho_1-\rho_2$ according to its spectral decomposition $\rho_1-\rho_2 = \lambda_1\sigma_1-\lambda_2\sigma_2$ with $\lambda_1,\lambda_2\geq 0$, and σ_1 and σ_2 two unit trace, Hermitian matrices that obey $\sigma_1\sigma_2 = 0$. Because the trace of the matrix $(\rho_1-\rho_2)$ is zero and $\|\sigma_1-\sigma_2\|_{\mathrm{tr}} = 2$, we have $\lambda_1=\lambda_2=\frac{1}{2}\|\rho_1-\rho_2\|_{\mathrm{tr}}$. We conclude the proof by using the triangle inequality and the homogeneity of the norm $\|\cdot\|_{\mathrm{tr}}$, in combination with the requirement that the $\boldsymbol{G}$ is a completely positive, trace preserving linear superoperator:

$$\begin{aligned}\|\boldsymbol{G}(\rho_1-\rho_2)\|_{\mathrm{tr}} &= \|\boldsymbol{G}(\lambda_1\sigma_1+(-\lambda_1)\sigma_2)\|_{\mathrm{tr}}\\ &\leq \lambda_1\cdot\|\boldsymbol{G}(\sigma_1)\|_{\mathrm{tr}}+\lambda_1\cdot\|\boldsymbol{G}(\sigma_2)\|_{\mathrm{tr}}\\ &= \|\rho_1-\rho_2\|_{\mathrm{tr}}.\end{aligned}$$

□

Definition 16 (Constant transformation) *A transformation is* constant *if it maps all states to the same output state.*

Fact 13 *If the mixture $p\rho_1+(1-p)\rho_2$ (with the non-degenerate probability $0<p<1$) is a pure state φ, then both ρ_1 and ρ_2 are identical to φ as well.*

Lemma 11 *If a CPSO $\boldsymbol{G} : M_n(\mathbb{C}) \to M_m(\mathbb{C})$ maps the totally mixed state $\frac{1}{n}\mathrm{I}_n$ to a pure state φ, then $\boldsymbol{G}$ is constant.*

Proof: Take an n-dimensional state ρ. The density matrix $\rho' = \frac{n+1}{n^2}\mathrm{I}_n - \frac{1}{n}\rho$ will represent a proper state, and by linearity we know that $\frac{1}{n+1}\boldsymbol{G}(\rho) + \frac{n}{n+1}\boldsymbol{G}(\rho') = \boldsymbol{G}(\frac{1}{n}\mathrm{I}_n) = |\varphi\rangle\langle\varphi|$. This is only possible if $\boldsymbol{G}(\rho) = |\varphi\rangle\langle\varphi|$ (in combination with $\boldsymbol{G}(\rho') = |\varphi\rangle\langle\varphi|$). □

Lemma 12 *Let $\boldsymbol{G}$ be a quantum mechanical transformation of a single qubit, and let ρ be a qubit state. If $\boldsymbol{G}$ is not constant and $\boldsymbol{G}(\rho)$ is a pure state, then ρ has to be a pure state.*

Proof: Let ρ be a mixed qubit and $\boldsymbol{G}(\rho)$ a pure state φ. We can decompose ρ always as $\lambda|\psi\rangle\langle\psi| + (1-\lambda)|\psi^\perp\rangle\langle\psi^\perp|$, with $\frac{1}{2} \leq \lambda < 1$ and $|\psi\rangle$ orthogonal to $|\psi^\perp\rangle$. By linearity, it follows that $\boldsymbol{G}(\rho) = (2\lambda-1)\boldsymbol{G}(\psi) + (2-2\lambda)\boldsymbol{G}(\frac{1}{2}\mathrm{I}_2)$ equals the pure state φ. Because $0 \leq 2\lambda - 1 < 1$ and $0 < 2 - 2\lambda \leq 1$, we can conclude that $\boldsymbol{G}$ maps the total mixture $\frac{1}{2}\mathrm{I}_2$ to the pure state. By the previous lemma this implies that $\boldsymbol{G}$ is constant. □

The space of $2^n \times 2^n$ matrices has dimension 4^n, hence every n qubit CPSO is uniquely defined by the images of 4^n independent states. However, the following lemma shows that for unitary transformations it is sometimes sufficient to know only 3^n images.

Lemma 13 *Let ρ_1, ρ_2, and ρ_3 be three distinct qubit density matrices representing pure states, such that there is a convex combination $\lambda_1\rho_1 + \lambda_2\rho_2 + \lambda_3\rho_3$ that represents the totally mixed qubit $\frac{1}{2}\mathrm{I}_2$. If $\boldsymbol{G}$ is a CPSO for n qubits that acts as the identity on the set $\{\rho_1, \rho_2, \rho_3\}^{\otimes n}$, then $\boldsymbol{G}$ is the identity mapping $\mathbf{I}_{2^n}$.*

Proof: Let P be the set of convex combinations of the three density matrices: $P = \{\lambda_1\rho_1 + \lambda_2\rho_2 + \lambda_3\rho_3 | \lambda_1 + \lambda_2 + \lambda_3 = 1; \lambda_1, \lambda_2, \lambda_3 \in [0,1]\}$. To simplify the discussion, we suppose without loss of generality that P contains the states $\zeta_z^\pm$ and $\zeta_x^\pm$. By linearity of $\boldsymbol{G}$, we know that it acts as the identity on all the states $\varrho_1 \otimes \cdots \otimes \varrho_n$ as long as $\varrho_i \in P$ for all $1 \leq i \leq n$. It will be sufficient to show that $\boldsymbol{G}$ is the identity on density matrices representing non-entangled pure states, since they form a basis for all density matrices.

For every k, let A_k be the set of density matrices representing k-qubit non-entangled pure states, and let $B_{n-k} = \{\zeta_x^\pm, \zeta_z^\pm\}^{\otimes n-k}$. We will show by induction on k that, for every $0 \leq k \leq n$, the CPSO $\boldsymbol{G}$ acts as the identity on $A_k \otimes B_{n-k}$. The case $k = 0$ follows by the hypothesis of the lemma.

Suppose the statement is true for some k. Fix $\sigma \in A_k$ and $\tau \in B_{n-k-1}$. For every one-qubit density matrix ρ let $\tilde{\rho}$ denote the n-qubit density matrix $\sigma \otimes \rho \otimes \tau$.

We now prove that $G(\tilde{\rho}) = \tilde{\rho}$, for every $\rho \in A_1$. For this, we use the fact that the density matrix Ψ^+ representing the entangled EPR state $\frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$, can be written in terms of tensor products of the ζ states:

$$\Psi^+ \;=\; \tfrac{1}{2}(\zeta_x^+ \otimes \zeta_x^+ + \zeta_x^- \otimes \zeta_x^- + \zeta_z^+ \otimes \zeta_z^+ + \zeta_z^- \otimes \zeta_z^- - \zeta_y^+ \otimes \zeta_y^+ - \zeta_y^- \otimes \zeta_y^-).$$

This can be generalized for the pure state $|\tilde{\Psi}^+\rangle = \frac{1}{\sqrt{2}}(|\tilde{0}\rangle|\tilde{0}\rangle + |\tilde{1}\rangle|\tilde{1}\rangle)$:

$$\tilde{\Psi}^+ \;=\; \tfrac{1}{2}(\tilde{\zeta}_x^+ \otimes \tilde{\zeta}_x^+ + \tilde{\zeta}_x^- \otimes \tilde{\zeta}_x^- + \tilde{\zeta}_z^+ \otimes \tilde{\zeta}_z^+ + \tilde{\zeta}_z^- \otimes \tilde{\zeta}_z^-) - \tfrac{1}{2}(\tilde{\zeta}_y^+ \otimes \tilde{\zeta}_y^+ + \tilde{\zeta}_y^- \otimes \tilde{\zeta}_y^-).$$

If we apply the superoperator $\mathbf{I}_{2^n} \otimes \boldsymbol{G}$ to the state $\tilde{\Psi}^+$ we get:

$$\begin{aligned}(\mathbf{I}_{2^n} \otimes \boldsymbol{G})(\tilde{\Psi}^+) \;=\;& +\tfrac{1}{2}(\tilde{\zeta}_x^+ \otimes \tilde{\zeta}_x^+ + \tilde{\zeta}_x^- \otimes \tilde{\zeta}_x^-)\\ & +\tfrac{1}{2}(\tilde{\zeta}_z^+ \otimes \tilde{\zeta}_z^+ + \tilde{\zeta}_z^- \otimes \tilde{\zeta}_z^-)\\ & -\tfrac{1}{2}(\tilde{\zeta}_y^+ \otimes \boldsymbol{G}(\tilde{\zeta}_y^+) + \tilde{\zeta}_y^- \otimes \boldsymbol{G}(\tilde{\zeta}_y^-)).\end{aligned}$$

If $|\varphi\rangle$ and $|\varphi^\perp\rangle$ are orthogonal n-qubit pure states, then let $\Phi^-_{\varphi\varphi^\perp} = \frac{1}{\sqrt{2}}(|\varphi\rangle|\varphi^\perp\rangle - |\varphi^\perp\rangle|\varphi\rangle)$. Since $\Phi^-_{\varphi\varphi^\perp}$ is orthogonal to all symmetric $2n$-qubit pure states of the form $\psi \otimes \psi$, by projecting $(\mathbf{I}_{2^n} \otimes \boldsymbol{G})(\tilde{\Psi}^+)$ to $\Phi^-_{\varphi\varphi^\perp}$ we obtain:

$$\begin{aligned}\langle\Phi^-_{\varphi\varphi^\perp}|(\mathbf{I}_{2^n} \otimes \boldsymbol{G})(\tilde{\Psi}^+)|\Phi^-_{\varphi\varphi^\perp}\rangle \;=\;& -\tfrac{1}{2}\langle\Phi^-_{\varphi\varphi^\perp}|\tilde{\zeta}_y^+ \otimes \boldsymbol{G}(\tilde{\zeta}_y^+)|\Phi^-_{\varphi\varphi^\perp}\rangle\\ & -\tfrac{1}{2}\langle\Phi^-_{\varphi\varphi^\perp}|\tilde{\zeta}_y^- \otimes \boldsymbol{G}(\tilde{\zeta}_y^-)|\Phi^-_{\varphi\varphi^\perp}\rangle.\end{aligned}$$

Since $\boldsymbol{G}$ is a completely positive, the left-hand side of this equality has to be non-negative and in the right-hand side both terms are non-positive. Therefore, for every orthogonal n-qubit pure states $|\varphi\rangle$ and $|\varphi^\perp\rangle$, we get

$$\langle\Phi^-_{\varphi\varphi^\perp}|\tilde{\zeta}_y^+ \otimes \boldsymbol{G}(\tilde{\zeta}_y^+)|\Phi^-_{\varphi\varphi^\perp}\rangle \;=\; \langle\Phi^-_{\varphi\varphi^\perp}|\tilde{\zeta}_y^- \otimes \boldsymbol{G}(\tilde{\zeta}_y^-)|\Phi^-_{\varphi\varphi^\perp}\rangle \;=\; 0.$$

A straightforward calculation then shows that $\boldsymbol{G}(\tilde{\zeta}_y^\pm) = \tilde{\zeta}_y^\pm$. Therefore $\boldsymbol{G}$ acts as the identity on density matrices $\tilde{\zeta}_z^\pm$, $\tilde{\zeta}_x^+$ and $\tilde{\zeta}_y^+$, which generate all density matrices, and thus $\boldsymbol{G}(\tilde{\rho}) = \tilde{\rho}$. □

We also use the property that for CPSOs unitarity and invertibility are equivalent.

Fact 14 *Let $\boldsymbol{G}$ be a CPSO for n qubits. If there exists a CPSO $\boldsymbol{F}$ for n qubits such that $\boldsymbol{F} \circ \boldsymbol{G}$ is the identity mapping, then $\boldsymbol{G}$ is a unitary superoperator.*

Proof: See, for example, Chapter 3.8 in [78]. □

6.6 Characterization of CPSO Families

In this section, every CPSO will be for one qubit. First we define the notion of experimental equations, and then we show that several important CPSO families are characterizable by them.

Definition 17 (Experimental equation) *An* experimental equation *in one CPSO variable, is an equation of the form*

$$\mathrm{Prob}^0[\boldsymbol{G}^k(|b\rangle\langle b|)] \;=\; r, \tag{6.4}$$

where k is a nonnegative integer, $b \in \{0, 1\}$, and $0 \le r \le 1$.

We will call the left-hand side of the equation the *probability term*, and the right-hand side the *constant term*. The *size* of this equation is k. A CPSO $\boldsymbol{G}$ will “almost” satisfy the equations if, for example, it is the result of adding small systematic and random errors (independent of time) to a CPSO that does. For $\varepsilon \geq 0$, the CPSO $\boldsymbol{G}$ *ε-satisfies* Equation 6.4 if $|\mathrm{Prob}^0[\boldsymbol{G}^k(|b\rangle\langle b|)] - r| \leq \varepsilon$, and when $\varepsilon = 0$ we will just say that $\boldsymbol{G}$ *satisfies* Equation 6.4. Let $\{E\}$ be a finite set of experimental equations. If $\boldsymbol{G}$ ε-satisfies all equations in $\{E\}$ we say that $\boldsymbol{G}$ ε-satisfies $\{E\}$. If some $\boldsymbol{G}$ satisfies $\{E\}$ then $\{E\}$ is *satisfiable*. The set $\{\boldsymbol{G} : \boldsymbol{G}$ satisfies $\{E\}\}$ will be denoted by $\mathcal{F}_{\{E\}}$. A family $\mathcal{F}$ of CPSOs is *characterizable* if it is $\mathcal{F}_{\{E\}}$ for some finite set $\{E\}$ of experimental equations. In this case we say that $\{E\}$ *characterizes* $\mathcal{F}$.

All these definitions generalize naturally for m-tuples of CPSOs for $m \geq 2$. In what follows we will need only the case $m = 2$. An *experimental equation* in two CPSO variables is an equation of the form

$$\mathrm{Prob}^0[\boldsymbol{F}^{k_1} \circ \boldsymbol{G}^{l_1} \circ \cdots \circ \boldsymbol{F}^{k_t} \circ \boldsymbol{G}^{l_t}(|b\rangle\langle b|)] \;=\; r,$$

where $k_1, \ldots, k_t, l_1, \ldots, l_t$ are nonnegative integers, $b \in \{0,1\}$, and $0 \leq r \leq 1$.

We discuss now the existence of finite sets of experimental equations in one variable that characterize unitary superoperators, that is, the operators $\mathbf{R}_{\alpha,\theta,\phi}$, for $\alpha \in (-\pi, \pi]$, $\theta \in [0, \frac{\pi}{2}]$, and $\phi \in [0, 2\pi)$. First observe that due to the restrictions of experimental equations, there are unitary superoperators that they cannot distinguish.

Fact 15 *Let $\alpha \in [0,\pi]$, $\theta \in [0, \frac{\pi}{2}]$, and $\phi_1, \phi_2 \in [0, 2\pi)$ such that $\phi_1 \neq \phi_2$. Let $\{E\}$ be a finite set of experimental equations in m variables. If*

$$(\mathbf{R}_{\alpha,\theta,\phi_1}, \boldsymbol{G}_2, \ldots, \boldsymbol{G}_m) \textit{ satisfies } \{E\}$$

then there exist $\boldsymbol{G}'_2, \ldots, \boldsymbol{G}'_m$ and $\boldsymbol{G}''_2, \ldots, \boldsymbol{G}''_m$ such that

$$(\mathbf{R}_{-\alpha,\theta,\phi_1}, \boldsymbol{G}'_2, \ldots, \boldsymbol{G}'_m) \textit{ and } (\mathbf{R}_{\alpha,\theta,\phi_2}, \boldsymbol{G}''_2, \ldots, \boldsymbol{G}''_m) \textit{ both satisfy } \{E\}.$$

In the Bloch Ball formalism this corresponds to the following degrees of freedom in the choice of the orthonormal basis of $\mathbb{R}^3$. Since experimental equations contain exactly the states $|0\rangle\langle 0|$ and $|1\rangle\langle 1|$ there is no freedom in the choice of the z-axis, but there is complete freedom in the choice of the x and y axes. The indistinguishability of the latitude ϕ corresponds to the freedom of choosing the oriented x-axis, and the indistinguishability of the sign of α corresponds to the freedom of choosing the orientation of the y-axis.

We introduce the following notations. Let $\mathcal{R}_{\alpha,\theta}$ denote the superoperator family $\{\mathbf{R}_{\pm\alpha,\theta,\phi} | \phi \in [0, 2\pi)\}$. For $\phi \in [0, 2\pi)$, let the Not_ϕ transformation be defined by $\mathrm{Not}_\phi|0\rangle = \mathrm{e}^{\mathrm{i}\phi}|1\rangle$ and $\mathrm{Not}_\phi(\mathrm{e}^{\mathrm{i}\phi}|1\rangle) = |0\rangle$, and recall that the Hadamard transformation H_ϕ obeys $\mathrm{H}_\phi|0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + \mathrm{e}^{\mathrm{i}\phi}|1\rangle)$ and $\mathrm{H}_\phi(\mathrm{e}^{\mathrm{i}\phi}|1\rangle) = \frac{1}{\sqrt{2}}(|0\rangle - \mathrm{e}^{\mathrm{i}\phi}|1\rangle)$. Observe that $\mathbf{H}_\phi = \mathbf{R}_{\pi,\frac{\pi}{4},\phi}$ and $\mathbf{Not}_\phi = \mathbf{R}_{\pi,\frac{\pi}{2},\phi}$, for $\phi \in [0, 2\pi)$. Finally, let $\{\mathbf{H}_\phi\} = \{\mathbf{H}_\phi | \phi \in [0, 2\pi)\}$, and $\{\mathbf{Not}_\phi\} = \{\mathbf{Not}_\phi | \phi \in [0, 2\pi)\}$.

Since the sign of α cannot be determined, we will assume that α is in the interval $[0, \pi]$. We will also consider only unitary superoperators such that $\frac{\alpha}{\pi}$ is rational. This choice is good enough since these superoperators form a dense subset of all unitary superoperators. For such a unitary superoperator, let n_α be the smallest positive integer n for which $n\alpha = 0 \bmod 2\pi$. Then either $n_\alpha = 1$, or $n_\alpha \geq 2$ and there exists $t \geq 1$ which is coprime with n_α such that $\alpha = (\frac{t}{n_\alpha})2\pi$. Observe that the case $n_\alpha = 1$ corresponds to the identity superoperator.

Our first theorem shows that almost all families $\mathcal{R}_{\alpha,\theta}$ are characterizable by some finite set of experimental equations. In particular $\{\mathbf{H}_\phi\}$ is characterizable.

Theorem 10 *Let* $(\alpha, \theta) \in (0, \pi] \times (0, \frac{\pi}{2}] \backslash \{(\pi, \frac{\pi}{2})\}$ *be such that* $\frac{\alpha}{\pi}$ *is rational. Let* $z_k(\alpha, \theta) = \cos^2\theta + \sin^2\theta \cos(k\alpha)$. *Then the following experimental equations characterize* $\mathcal{R}_{\alpha,\theta}$*:*

$$\mathrm{Prob}^0[\boldsymbol{G}^{n_\alpha}(|1\rangle\langle 1|)] = 0 \quad \textit{and} \quad \mathrm{Prob}^0[\boldsymbol{G}^k(|0\rangle\langle 0|)] = \tfrac{1}{2} + \tfrac{1}{2} z_k(\alpha, \theta),$$

for $k \in \{1, \ldots, n_\alpha\}$.

Proof: First observe that every CPSO in $\mathcal{R}_{\alpha,\theta}$ satisfies the equations of the theorem since the z-coordinate of $\overrightarrow{\mathbf{R}^k_{\alpha,\theta,\phi}(|0\rangle\langle 0|)}$ is $z_k(\alpha, \theta)$ for every $\phi \in [0, 2\pi)$. Let $\boldsymbol{G}$ be a CPSO that satisfies these equations. We will prove that $\boldsymbol{G}$ is a unitary superoperator. Then, Fact 16 implies that $\boldsymbol{G} \in \mathcal{R}_{\alpha,\theta}$.

Since $z_1(\alpha, \theta) \neq \pm 1$, we know $\boldsymbol{G}(|0\rangle\langle 0|) \notin \{|0\rangle\langle 0|, |1\rangle\langle 1|\}$. Observing that $\boldsymbol{G}^{n_\alpha}(|0\rangle\langle 0|) = |0\rangle\langle 0|$, Lemma 12 implies that $\boldsymbol{G}(|0\rangle\langle 0|)$ is a pure state. Thus $|0\rangle\langle 0|$, $|1\rangle\langle 1|$, and $\boldsymbol{G}(|0\rangle\langle 0|)$ are distinct pure states, and since $\boldsymbol{G}^{n_\alpha}$ acts as the identity on them, by Lemma 13 it is the identity mapping. Hence by Fact 14 $\boldsymbol{G}$ is a unitary superoperator. □

Fact 16 *Let* $\alpha \in (0, \pi]$, $\theta \in (0, \frac{\pi}{2}]$, $\alpha' \in (-\pi, \pi]$, $\theta' \in (0, \frac{\pi}{2}]$, *with* $\frac{\alpha}{\pi}$ *a rational and* n_α *the smallest positive integer such that* $n_\alpha \alpha = 0 \bmod 2\pi$, *and let* z_k *be the function* $z_k(\alpha, \theta) = \cos^2\theta + \sin^2\theta\cos(k\alpha)$. *If* $z_k(\alpha, \theta) = z_k(\alpha', \theta')$, *for* $k \in \{1, \ldots, n_\alpha\}$, *then* $|\alpha'| = \alpha$ *and* $\theta' = \theta$.

The remaining families $\mathcal{R}_{\alpha,\theta}$ for which $\frac{\alpha}{\pi}$ is rational are $\{\mathbf{R}_{-\alpha}, \mathbf{R}_\alpha\}$, for $\alpha \in [0, \pi]$, and $\{\mathbf{Not}_\phi\}$. Let us recall that $\boldsymbol{M}$ is the CPSO that represents the Von Neumann measurement in the computational basis. Since $\boldsymbol{M}$ satisfies exactly the same equations as $\mathbf{R}_{\pm\alpha}$, and $\mathbf{Not}_0 \circ \boldsymbol{M}$ satisfies exactly the same equations as $\mathbf{Not}_\phi$, for every $\phi \in [0, 2\pi)$, these families are not characterizable by experimental equations in one variable. Nevertheless it turns out that together with the family $\{\mathbf{H}_\phi\}$ they become characterizable. This is stated in the following theorem.

Theorem 11 *The family* $\{(\mathbf{H}_\phi, \mathbf{Not}_\phi) | \phi \in [0, 2\pi)\} \subset \{\mathbf{H}_\phi\} \times \{\mathbf{Not}_\phi\}$ *is character-*

ized by the experimental equations in two variables $(\boldsymbol{F},\boldsymbol{G})$:

$$\begin{cases} \mathrm{Prob}^0[\boldsymbol{F}(|0\rangle\langle 0|)] = \frac{1}{2} \\ \mathrm{Prob}^0[\boldsymbol{F}^2(|0\rangle\langle 0|)] = 1 \\ \mathrm{Prob}^0[\boldsymbol{F}^2(|1\rangle\langle 1|)] = 0 \\ \\ \mathrm{Prob}^0[\boldsymbol{G}(|0\rangle\langle 0|)] = 0 \\ \mathrm{Prob}^0[\boldsymbol{G}(|1\rangle\langle 1|)] = 1 \\ \\ \mathrm{Prob}^0[\boldsymbol{F}\circ\boldsymbol{G}\circ\boldsymbol{F}(|0\rangle\langle 0|)] = 1. \end{cases}$$

If $\frac{\alpha}{\pi}$ is rational, then the family $\{\mathrm{H}_\phi\}\times\{\mathbf{R}_{\pm\alpha}\}$ is characterized by the experimental equations in two variables $(\boldsymbol{F},\boldsymbol{G})$:

$$\begin{cases} \mathrm{Prob}^0[\boldsymbol{F}(|0\rangle\langle 0|)] = \frac{1}{2} \\ \mathrm{Prob}^0[\boldsymbol{F}^2(|0\rangle\langle 0|)] = 1 \\ \mathrm{Prob}^0[\boldsymbol{F}^2(|1\rangle\langle 1|)] = 0 \\ \\ \mathrm{Prob}^0[\boldsymbol{G}(|0\rangle\langle 0|)] = 1 \\ \mathrm{Prob}^0[\boldsymbol{G}(|1\rangle\langle 1|)] = 0 \\ \\ \mathrm{Prob}^0[\boldsymbol{F}\circ\boldsymbol{G}^{n_\alpha}\circ\boldsymbol{F}(|0\rangle\langle 0|)] = 1 \\ \mathrm{Prob}^0[\boldsymbol{F}\circ\boldsymbol{G}\circ\boldsymbol{F}(|0\rangle\langle 0|)] = \frac{1}{2}+\frac{1}{2}\cos\alpha. \end{cases}$$

Proof: By the previous theorem, H_ϕ is characterized by the first three experimental equations involving $\boldsymbol{F}$. Because of this we know that $\boldsymbol{F}|0\rangle\langle 0|$ corresponds to the pure state $|\zeta_x^\phi\rangle = \frac{1}{\sqrt{2}}(|0\rangle + \mathrm{e}^{\mathrm{i}\phi}|1\rangle)$.

In combination with the knowledge that $\boldsymbol{G}\circ\boldsymbol{F}|0\rangle\langle 0|$ also yields the state ζ_x^ϕ, this tells us that $\boldsymbol{G}$ acts as the identity on ζ_x^ϕ. Consider now the combined CPSO $\mathbf{Not}_\phi\circ\boldsymbol{G}$. This operator acts as the identity on the three density matrices $|0\rangle\langle 0|$, $|1\rangle\langle 1|$, ζ_x^ϕ, which, following Lemma 13, implies that $\mathbf{Not}_\phi\circ\boldsymbol{G}$ is indeed $\mathbf{I}_2$. This is only possible if $\boldsymbol{G}$ equals $\mathbf{Not}_\phi$.

For the second part of the theorem, we employ a proof method of similar vein. Because $\boldsymbol{F}|0\rangle\langle 0| = \zeta_x^\phi$ and $\boldsymbol{F}\circ\boldsymbol{G}^{n_\alpha}\circ\boldsymbol{F}(|0\rangle\langle 0|) = |0\rangle\langle 0|$, We know that $\boldsymbol{G}^{n_\alpha}$ acts as the identity on the pure state ζ_x^ϕ, and hence (using $\boldsymbol{G}|0\rangle\langle 0| = |0\rangle\langle 0|$ and $\boldsymbol{G}|1\rangle\langle 1| = |1\rangle\langle 1|$) that $\boldsymbol{G}^{n_\alpha}$ is $\mathbf{I}_2$, which is only possible if $\boldsymbol{G}$ is unitary. The eigenvectors of the $\mathsf{U}(2)/\mathsf{U}(1)$ rotation associated with $\boldsymbol{G}$ are $|0\rangle$ and $|1\rangle$, and because the n_α-th power of $\boldsymbol{G}$ is the identity, its two eigenvalues have to obey $\lambda_0^{n_\alpha} = \lambda_1^{n_\alpha}$. By the probability $\mathrm{Prob}^0[\boldsymbol{F}\circ\boldsymbol{G}\circ\boldsymbol{F}(|0\rangle\langle 0|)] = \frac{1}{2}+\frac{1}{2}\cos\alpha$ it follows that $\lambda_1 = \lambda_0\mathrm{e}^{\pm\mathrm{i}\alpha}$. Hence $\boldsymbol{G}$ equals $\mathbf{R}_\alpha$ or $\mathbf{R}_{-\alpha}$. □

6.7 Characterization of CNot gates

In this section we will extend our theory of characterization of CPSO families for several qubits. In particular, we will show that the family of $\mathbf{CNot}$ gates together with the family $\{\mathbf{H}_\phi\}$ is characterizable. First we need some definitions.

For every $\phi \in [0, 2\pi)$, we define CNot_ϕ as the only unitary transformation over $\mathbb{C}^4$ satisfying $\mathrm{CNot}_\phi(|0\rangle|\psi\rangle) = |0\rangle|\psi\rangle$ and $\mathrm{CNot}_\phi|1\rangle|\psi\rangle = |1\rangle\mathrm{Not}_\phi|\psi\rangle$, for all $|\psi\rangle \in \mathcal{H}_2$.

We extend the definition of the experimental equation for CPSOs given in Equation 6.5 for n qubits. It is an equation of the form

$$\mathrm{Prob}^v[\boldsymbol{F}^{k_1} \circ \boldsymbol{G}^{l_1} \circ \cdots \circ \boldsymbol{F}^{k_t} \circ \boldsymbol{G}^{l_t}(|w\rangle\langle w|)] \quad = \quad r, \tag{6.5}$$

where in addition to the notation of Equation 6.5 $v, w \in \{0,1\}^n$, and Pr^v is the probability of measuring $|v\rangle\langle v|$. For the variables $\boldsymbol{F}$ and $\boldsymbol{G}$ of Equation 6.5, we also allow both the tensor product of two CPSO variables and the tensor product of a CPSO variable with the identity. We now state the characterization.

Theorem 12 *The family* $\{(\mathbf{H}_\phi, \mathbf{CNot}_\phi)|\phi \in [0, 2\pi)\}$ *is characterized by the experimental equations in two variables* $(\boldsymbol{F}, \boldsymbol{G})$*:*

$$\begin{cases}
\mathrm{Prob}^0[\boldsymbol{F}(|0\rangle\langle 0|)] = \frac{1}{2} \\
\mathrm{Prob}^0[\boldsymbol{F}^2(|0\rangle\langle 0|)] = 1 \\
\mathrm{Prob}^0[\boldsymbol{F}^2(|1\rangle\langle 1|)] = 0 \\
\\
\mathrm{Prob}^{00}[\boldsymbol{G}(|00\rangle\langle 00|)] = 1 \\
\mathrm{Prob}^{01}[\boldsymbol{G}(|01\rangle\langle 01|)] = 1 \\
\mathrm{Prob}^{11}[\boldsymbol{G}(|10\rangle\langle 10|)] = 1 \\
\mathrm{Prob}^{10}[\boldsymbol{G}(|11\rangle\langle 11|)] = 1 \\
\\
\mathrm{Prob}^{00}[(\mathbf{I}_2 \otimes \boldsymbol{F}) \circ \boldsymbol{G} \circ (\mathbf{I}_2 \otimes \boldsymbol{F})(|00\rangle\langle 00|)] = 1 \\
\mathrm{Prob}^{10}[(\mathbf{I}_2 \otimes \boldsymbol{F}) \circ \boldsymbol{G} \circ (\mathbf{I}_2 \otimes \boldsymbol{F})(|10\rangle\langle 10|)] = 1 \\
\\
\mathrm{Prob}^{00}[(\boldsymbol{F} \otimes \mathbf{I}_2) \circ \boldsymbol{G}^2 \circ (\boldsymbol{F} \otimes \mathbf{I}_2)(|00\rangle\langle 00|)] = 1 \\
\mathrm{Prob}^{01}[(\boldsymbol{F} \otimes \mathbf{I}_2) \circ \boldsymbol{G}^2 \circ (\boldsymbol{F} \otimes \mathbf{I}_2)(|01\rangle\langle 01|)] = 1 \\
\\
\mathrm{Prob}^{00}[(\boldsymbol{F} \otimes \boldsymbol{F}) \circ \boldsymbol{G} \circ (\boldsymbol{F} \otimes \boldsymbol{F})(|00\rangle\langle 00|)] = 1.
\end{cases}$$

Proof: Let $\boldsymbol{F}$ and $\boldsymbol{G}$ satisfy these equations. By Theorem 10, with $\alpha = \pi$ and $\theta = \frac{\pi}{4}$, the first three equations imply that $\boldsymbol{F} = \mathbf{H}_\phi$, for some $\phi \in [0, 2\pi)$. Using Lemma 13, the remaining equations imply that $\boldsymbol{G}^2 = \mathbf{I}_4$, and it follows from Fact 14 that $\boldsymbol{G}$ is a unitary CPSO. A straightforward verification then shows that indeed $\boldsymbol{G} = \mathbf{CNot}_\phi$.

□

6.8 Robustness

In this section we introduce the notion of robustness for experimental equations which will be the crucial ingredient for proving self-testability. From now on, $\{E\}$ will always denote a such a set of equations.

First we define the notion of the distance between a CPSO and a family of gates.

Definition 18 *The distance between a CPSO $\boldsymbol{G}$ and a set $\mathcal{F}$ of gates is defined by the minimization*

$$\mathrm{dist}_{\mathrm{tr}}(\boldsymbol{G},\mathcal{F}) \quad := \quad \min_{\boldsymbol{F}\in\mathcal{F}} \mathrm{dist}_{\mathrm{tr}}(\boldsymbol{G},\boldsymbol{F}).$$

For a Euclidean metric, this distance would express the length of the shortest line between a point and a set. We use this generalized distance to define a notion of 'robustness' for a set of experimental equations.

Definition 19 (Robustness) *Let $\varepsilon, \delta \geq 0$, and let $\{E\}$ be a set of experimental equations. We say that $\{E\}$ is (ε,δ)*-robust *if whenever a CPSO $\boldsymbol{G}$ ε-satisfies $\{E\}$, we have $\mathrm{dist}_{\mathrm{tr}}(\boldsymbol{G},\mathcal{F}_{\{E\}}) \leq \delta$.*

When a CPSO family is characterized by a finite set of experimental equations $\{E\}$, one would like to prove that $\{E\}$ is robust. The next theorem shows that this is the case for $\delta \in O(\varepsilon^{1/k})$ with k depending on $\{E\}$.

Theorem 13 *Let $\{E\}$ be a finite satisfiable set of experimental equations. Then there exists an integer $k \geq 1$ and a real $C > 0$ such that for all $\varepsilon \geq 0$, $\{E\}$ is $(\varepsilon, C\varepsilon^{1/k})$-robust.*

Proof: We will use basic notions from algebraic geometry for which we refer the reader for example to [18]. In the proof, $\mathbb{C}$ is identified with $\mathbb{R}^2$. Then the set $\mathcal{K}$ of CPSOs for a fixed number of qubits is a real compact semi-algebraic set. Suppose that in $\{E\}$ there are d equations. Let $f : \mathcal{K} \to \mathbb{R}$ be the function that maps the CPSO $\boldsymbol{G}$ to the maximum of the magnitudes of the difference between the probability term and the constant term of the i^{th} equation in $\{E\}$, for $i = 1,\ldots,d$. By definition of f, we get $f^{-1}(0) = \mathcal{F}_{\{E\}}$. Moreover, f is a continuous semi-algebraic function, since it is the maximum of the magnitudes of polynomial functions in the (real) coefficients of $\boldsymbol{G}$.

Let $g : \mathcal{K} \to \mathbb{R}$ defined in $\boldsymbol{G}$ by $g(\boldsymbol{G}) = \mathrm{dist}_{\mathrm{tr}}(\boldsymbol{G},\mathcal{F}_{\{E\}})$. Since $\mathcal{K}$ is a compact semi-algebraic set, g is a continuous semi-algebraic function. Moreover, for all $\boldsymbol{G} \in \mathcal{K}$, we have $f(\boldsymbol{G}) = 0$ if and only if $g(\boldsymbol{G}) = 0$. Then Fact 17 concludes the proof. □

For a proof of the following fact, see for example [18, Prop. 2.3.11].

Fact 17 (Lojasiewicz's inequality) *Let $X \subseteq \mathbb{R}^m$ be a compact semi-algebraic set. Let $f, g : X \to \mathbb{R}$ be two continuous semi-algebraic functions. Assume that for all $x \in X$, if $f(x) = 0$ then $g(x) = 0$. Then there exists an integer $k \geq 1$ and a real $C > 0$ such that, for all $x \in X$, $|g(x)|^k \leq C|f(x)|$.*

In some cases we can explicitly compute the constants C and k of Theorem 13. We will illustrate these techniques with the equations in Theorem 10 for the case $\alpha = \pi$ and $\theta = \frac{\pi}{4}$. Let us recall that these equations characterize the set $\{\mathbf{H}_\phi\}$.

Lemma 14 *For every* $0 \le \varepsilon \le 1$*, the following equations are* $(\varepsilon, 1824\sqrt{\varepsilon})$*-robust:*

$$\mathrm{Prob}^0[\boldsymbol{G}(|0\rangle\langle 0|)] = \tfrac{1}{2}, \quad \mathrm{Prob}^0[\boldsymbol{G}^2(|0\rangle\langle 0|)] = 1, \text{ and } \mathrm{Prob}^0[\boldsymbol{G}^2(|1\rangle\langle 1|)] = 0.$$

Proof: Let $\boldsymbol{G}$ be a CPSO that ε-satisfies the equations. First we will show there is a point $\overrightarrow{\rho} \in \mathcal{S}$ with z-coordinate 0 whose distance from $\overrightarrow{\boldsymbol{G}(|0\rangle\langle 0|)}$ is at most $10\sqrt{\varepsilon}$. The last two equations imply that $\|\boldsymbol{G}^2(|b\rangle\langle b|) - |b\rangle\langle b|\|_{\mathrm{tr}} \le 3\sqrt{\varepsilon}$, for $b = 0, 1$. Therefore $\|\boldsymbol{G}^2(|0\rangle\langle 0|) - \boldsymbol{G}^2(|1\rangle\langle 1|)\|_{\mathrm{tr}} \ge 2 - 6\sqrt{\varepsilon}$, and by Fact 12 we have $\|\boldsymbol{G}(|0\rangle\langle 0|) - \boldsymbol{G}(|1\rangle\langle 1|)\|_{\mathrm{tr}} \ge 2 - 6\sqrt{\varepsilon}$. Thus $\|\overrightarrow{\boldsymbol{G}(|b\rangle\langle b|)}\|_2 \ge 1 - 6\sqrt{\varepsilon}$, for $b = 0, 1$. Let $\tau = \rho(\frac{1}{2}, \alpha)$, where $\boldsymbol{G}(|0\rangle\langle 0|) = \rho(p, \alpha)$. The first equation implies that $\|\overrightarrow{\tau} - \overrightarrow{\boldsymbol{G}(|0\rangle\langle 0|)}\|_2 \le 2\varepsilon$. Therefore, for $\overrightarrow{\rho} = \overrightarrow{\tau}/\|\overrightarrow{\tau}\|_2$ we get $\|\boldsymbol{G}(|0\rangle\langle 0|) - \rho\|_{\mathrm{tr}} \le 10\sqrt{\varepsilon}$.

The point $\overrightarrow{\rho}$ on $\mathcal{S}$ uniquely defines $\phi \in [0, 2\pi)$ such that $\overrightarrow{\mathbf{H}_\phi(|0\rangle\langle 0|)} = \overrightarrow{\rho}$. One can verify that $\mathbf{H}_\phi^{-1} \circ \boldsymbol{G}$ acts as the identity with error at most $19\sqrt{\varepsilon}$ on the four density matrices $|0\rangle\langle 0|$, $|1\rangle\langle 1|$, $\mathbf{H}_\phi(|0\rangle\langle 0|)$, and $\mathbf{H}_\phi(|1\rangle\langle 1|)$. From Lemma 16 we conclude that $|||\boldsymbol{G} - \mathbf{H}_\phi|||_{\mathrm{tr}} \le 1824\sqrt{\varepsilon}$. □

Lemma 15 *Let* $\boldsymbol{G}$ *be a superoperator on* $M_2(\mathbb{C})$*. Let* $0 \le \varepsilon \le 1$ *be such that* $\|\boldsymbol{G}(\zeta_x^\pm) - \zeta_x^\pm\|_{\mathrm{tr}}$, $\|\boldsymbol{G}(\zeta_y^\pm) - \zeta_y^\pm\|_{\mathrm{tr}}$, $\|\boldsymbol{G}(\zeta_z^\pm) - \zeta_z^\pm\|_{\mathrm{tr}} \le \varepsilon$*; then* $|||\boldsymbol{G} - \mathbf{I}_2|||_{\mathrm{tr}} \le \sqrt{10}\varepsilon$.

Proof: Define a four dimensional basis $\{b_i\}$ for the linear space $\mathbb{C}^{2\times 2}$ by $b_1 = \zeta_x^+$, $b_2 = \zeta_x^-$, $b_3 = \zeta_y^+ - \zeta_z^-$ and $b_4 = \zeta_y^- - \zeta_z^-$. Any 2×2 complex valued matrix can now be expressed as $M_\alpha = \sum_i \alpha_i b_i$, with $\alpha_i \in \mathbb{C}$. This implies for the trace norm of the matrix $\|M_\alpha\|_{\mathrm{tr}} \ge \|M_\alpha\|_2 = \sqrt{|\alpha_1|^2 + |\alpha_2|^2 + |\alpha_3|^2 + |\alpha_4|^2}$. By the assumption of the lemma we have $\|(\boldsymbol{G} - \mathbf{I}_2)(b_1)\|_{\mathrm{tr}}, \|(\boldsymbol{G} - \mathbf{I}_2)(b_2)\|_{\mathrm{tr}} \le \varepsilon$, and also $\|(\boldsymbol{G} - \mathbf{I}_2)(b_3)\|_{\mathrm{tr}}, \|(\boldsymbol{G} - \mathbf{I}_2)(b_4)\|_{\mathrm{tr}} \le 2\varepsilon$. Combining these bounds yields

$$\|(\boldsymbol{G} - \mathbf{I}_2)(M_\alpha)\|_{\mathrm{tr}} \quad \le \quad (|\alpha_1| + |\alpha_2| + 2|\alpha_3| + 2|\alpha_4|)\varepsilon.$$

We are thus left to maximize the fraction

$$\frac{\|\boldsymbol{G} - \mathbf{I}_2(M_\alpha)\|_{\mathrm{tr}}}{\|M_\alpha\|_{\mathrm{tr}}} \le \frac{(|\alpha_1| + |\alpha_2| + 2|\alpha_3| + 2|\alpha_4|)\varepsilon}{\sqrt{|\alpha_1|^2 + |\alpha_2|^2 + |\alpha_3|^2 + |\alpha_4|^2}}$$

over all $\alpha_i \in \mathbb{C}$. Clearly, we can assume all α-coefficients to be nonnegative reals and impose the restriction $\sum_i \alpha_i^2 = 1$. With the use of Lagrange multipliers one can now prove without much effort that the above fraction cannot be bigger than $\sqrt{10}\varepsilon$ (which is established by the values $\alpha_1 = \alpha_2 = \sqrt{\frac{1}{10}}$ and $\alpha_3 = \alpha_4 = \sqrt{\frac{2}{5}}$). □

Lemma 16 *Let* u *and* v *represent two pure qubit states (and* $u^\perp$ *and* $v^\perp$ *the respective orthogonal dual states), with* $|\langle u|v\rangle|^2 = \frac{1}{2}$*. If* $\boldsymbol{G}$ *is a one-qubit CPSO such that* $\|\boldsymbol{G}(x) - x\|_{\mathrm{tr}} \le \varepsilon$ *for* $0 \le \varepsilon \le 1$ *and all* $x \in \{u, v, u^\perp, v^\perp\}$*, then* $|||\boldsymbol{G} - \mathbf{I}_2|||_{\mathrm{tr}} \le 96\varepsilon$.

Proof: We can suppose without loss of generality that $u = \zeta_x^+$ and $v = \zeta_z^+$. Consider the state $\rho = \boldsymbol{G}(\zeta_y^+)$, with its three parameters x, y, z in

$$\rho \;=\; \frac{1}{2}\left[\begin{array}{cc} 1+z & x-\mathrm{i}y \\ x+\mathrm{i}y & 1-z \end{array}\right].$$

From Fact 12 it follows that $\|\boldsymbol{G}(\zeta_z^+) - \rho\|_{\mathrm{tr}} \leq \|\zeta_z^+ - \zeta_y^+\|_{\mathrm{tr}} = \sqrt{2}$. By the assumption of this lemma we have that $\|\boldsymbol{G}(\zeta_z^+) - \zeta_z^+\|_{\mathrm{tr}} \leq \varepsilon$, and hence $\|\zeta_z^+ - \rho\|_{\mathrm{tr}} \leq \sqrt{2}+\varepsilon$. The same relation holds also for the other three fixed points ζ_z^-, ζ_x^+, and ζ_x^-. As a result, the three coordinates of ρ have to obey the four inequalities

$$x^2 + y^2 + (z \pm 1)^2 \text{ and } (x \pm 1)^2 + y^2 + z^2 \;\leq\; (\sqrt{2}+\varepsilon)^2 \;\leq\; 2 + 4\varepsilon \qquad (6.6)$$

A second set of restrictions on (x, y, z) comes from the complete positivity of $\boldsymbol{G}$. Like in the proof of Lemma 13 we use the decomposition of the EPR state Ψ^+, to analyze the two-qubit state:

$$\begin{aligned}(\mathbf{I}_2 \otimes \boldsymbol{G})(\Psi^+) \;=\; & +\tfrac{1}{2}(\zeta_x^+ \otimes \boldsymbol{G}(\zeta_x^+) + \zeta_x^- \otimes \boldsymbol{G}(\zeta_x^-)) \\ & +\tfrac{1}{2}(\zeta_z^+ \otimes \boldsymbol{G}(\zeta_z^+) + \zeta_z^- \otimes \boldsymbol{G}(\zeta_z^-)) \\ & -\tfrac{1}{2}(\zeta_y^+ \otimes \boldsymbol{G}(\zeta_y^+) + \zeta_y^- \otimes \boldsymbol{G}(\zeta_y^-)).\end{aligned}$$

Using the hypothesis, the projection of this state onto the anti-symmetrical entangled qubit pair $|\Phi^-\rangle = \frac{1}{\sqrt{2}}(|01\rangle - |10\rangle)$ yields

$$\begin{aligned}\langle\Phi^-|(\mathbf{I}_2 \otimes \boldsymbol{G})(\Psi^+)|\Phi^-\rangle \;\leq\; & 2\varepsilon - \tfrac{1}{2}\langle\Phi^-|\zeta_y^+ \otimes \boldsymbol{G}(\zeta_y^+)|\Phi^-\rangle \\ & -\tfrac{1}{2}\langle\Phi^-|\zeta_y^- \otimes \boldsymbol{G}(\zeta_y^-)|\Phi^-\rangle.\end{aligned}$$

Since $\boldsymbol{G}$ is a CPSO, as in Lemma 13 we get $\langle\Phi^-|\zeta_y^+ \otimes \rho|\Phi^-\rangle \leq 4\varepsilon$. A straightforward calculation shows that this last relation is equivalent with a restriction on the y coordinate: $y \geq 1 - 16\varepsilon$.

This last inequality implies $y^2 \geq 1 - 32\varepsilon$, which combined with the restrictions of Equation 6.6, leads to the conclusion that $(x \pm 1)^2 \leq 2 + 4\varepsilon - y^2 - z^2 \leq 1 + 36\varepsilon$, and similarly $(z \pm 1)^2 \leq 1 + 36\varepsilon$. The x and z coordinates of ρ satisfy $|x|, |z| \leq 18\varepsilon$. Together these bounds imply

$$\|\boldsymbol{G}(\zeta_y^+) - \zeta_y^+\|_{\mathrm{tr}} \;=\; \sqrt{x^2 + (y-1)^2 + z^2} \;\leq\; \sqrt{904}\varepsilon.$$

The same result can be proved for ζ_y^-. Therefore by Lemma 15 we can conclude the proof. □

6.9 Quantum Self-Testers

In this final section we formally define our testers and establish the relationship between robust equations and testability. The *experimental oracle* $\mathcal{O}[\boldsymbol{G}]$ for $\boldsymbol{G}$ is a probabilistic procedure that takes inputs $(b, k) \in \{0, 1\} \times \mathbb{N}$ and generates outcomes from the set $\{0,1\}$ such that for every input bit b and size k

$$\Pr[\mathcal{O}[\boldsymbol{G}](b,k)=0] \quad = \quad \mathrm{Prob}^0[\boldsymbol{G}^k(|b\rangle\langle b|)].$$

An oracle program T with an experimental oracle $\mathcal{O}[\boldsymbol{G}]$ is a program denoted by $T^{\mathcal{O}[\boldsymbol{G}]}$ that can ask queries to the experimental oracle in the following sense. When T presents a query (b,k) to the oracle, it receives the probabilistic outcome of $\mathcal{O}[\boldsymbol{G}]$ in one computational step. A query to the experimental oracle thus captures the notion of a single experimental run of the black-box $\boldsymbol{G}$.

Definition 20 (Tester) *Let $\mathcal{F}$ be a family of CPSOs, and let $0 \leq \delta_1 \leq \delta_2 < 1$. A* (δ_1,δ_2)-tester for *$\mathcal{F}$ is a probabilistic oracle program T such that for every CPSO $\boldsymbol{G}$,*

- *if* $\mathrm{dist}_{\mathrm{tr}}(\boldsymbol{G},\mathcal{F}) \leq \delta_1$ *then* $\Pr[T^{\mathcal{O}[\boldsymbol{G}]}$ *says* `PASS`$] \geq \frac{2}{3}$,
- *if* $\mathrm{dist}_{\mathrm{tr}}(\boldsymbol{G},\mathcal{F}) > \delta_2$ *then* $\Pr[T^{\mathcal{O}[\boldsymbol{G}]}$ *says* `FAIL`$] \geq \frac{2}{3}$,

where the probability is the expectation over the outcomes of the experimental oracle and the internal coin tosses of the program.

Theorem 14 *Let $\varepsilon,\delta > 0$, and let $\{E\}$ be a satisfiable set of d experimental equations such that the size of every equation is at most k. If $\{E\}$ is (ε,δ)-robust then there exists an $(\frac{\varepsilon}{3k},\delta)$-tester for $\mathcal{F}_{\{E\}}$ that makes $O(d\log(d)/\varepsilon^2)$ queries.*

Proof: We will describe a probabilistic oracle program T. Let $\boldsymbol{G}$ be a CPSO. We can suppose that for every equation in $\{E\}$, T has a rational number $\tilde{r}$ such that $|\tilde{r}-r| \leq \frac{\varepsilon}{6}$, where r is the constant term of the equation. By sampling the oracle $\mathcal{O}[\boldsymbol{G}]$, for every equation in $\{E\}$, T obtains a value $\tilde{p}$ such that $|\tilde{p}-p| \leq \frac{\varepsilon}{6}$ with probability at least $1-\frac{1}{3d}$, where p is the probability term of the equation. A standard Chernoff bound argument shows that this is feasible with $O(\log(d)/\varepsilon^2)$ queries for each equation. If for every equation $|\tilde{p}-\tilde{r}| \leq \frac{2\varepsilon}{3}$, then T says `PASS`, otherwise T says `FAIL`. Using the robustness of $\{E\}$ and Lemma 17, one can verify that T is a $(\frac{\varepsilon}{3k},\delta)$-tester for $\mathcal{F}_{\{E\}}$. □

Lemma 17 *Let $\{E\}$ be a finite satisfiable set of experimental equations such that the size of every equation is at most k, and let $\boldsymbol{G}$ be a CPSO. For every $\varepsilon \geq 0$, if $\mathrm{dist}_{\mathrm{tr}}(\boldsymbol{G},\mathcal{F}_{\{E\}}) \leq \varepsilon$ then $\boldsymbol{G}$ $(k\varepsilon)$-satisfies $\{E\}$.*

Proof: Let $\boldsymbol{F}$ be the CPSO in $\mathcal{F}$ such that $\mathrm{dist}_{\mathrm{tr}}(\boldsymbol{G},\boldsymbol{F}) \leq \varepsilon$. Then $\mathrm{dist}_{\mathrm{tr}}(\boldsymbol{G}^j,\boldsymbol{F}^j) \leq j\varepsilon$ for every $j \in \mathbb{N}$. Hence, by the maximum size k of the experimental equations $\{E\}$, the lemma follows. □

Our main result is the consequence of Theorems 10, 11, 12, 13, 14, and Lemma 14.

Theorem 15 *Let $\mathcal{F}$ be one of the following families:*

- $\mathcal{R}_{\alpha,\theta}$ *for* $(\alpha,\theta) \in (0,\pi] \times (0,\frac{\pi}{2}]\backslash\{(\pi,\frac{\pi}{2})\}$ *where* $\frac{\alpha}{\pi}$ *is rational,*

- $\{(\mathbf{H}_\phi, \mathbf{Not}_\phi) | \phi \in [0, 2\pi)\}$*,*
- $\{\mathbf{H}_\phi\} \times \{\mathbf{R}_{\pm\alpha}\}$ *for* $\frac{\alpha}{\pi}$ *rational,*
- $\{(\mathbf{H}_\phi, \mathbf{CNot}_\phi) | \phi \in [0, 2\pi)\}$*,*
- $\{(\mathbf{H}_\phi, \mathbf{R}_{s\pi/4}, \mathbf{CNot}_\phi) | \phi \in [0, 2\pi), s = \pm 1\}$.

Then there exists an integer $k \geq 1$ *and a real* $C > 0$ *such that, for all* $\varepsilon > 0$*,* $\mathcal{F}$ *has an* $(\varepsilon, C\varepsilon^{1/k})$*-tester that makes* $O(1/\varepsilon^2)$ *queries. Moreover, for every* $0 < \varepsilon \leq 1$*,* $\{\mathbf{H}_\phi\}$ *has an* $(\frac{\varepsilon}{6}, 1824\sqrt{\varepsilon})$*-tester that makes* $O(1/\varepsilon^2)$ *queries.*

Note that each triplet of the last family forms a universal and fault-tolerant set of quantum gates[26].

Chapter 7

Quantum Kolmogorov Complexity

In the classical setting, the Kolmogorov complexity of a string is the length of the shortest program that can produce this string as its output, which is a measure of the amount of innate randomness (or information) contained in the string. In this chapter we define the quantum Kolmogorov complexity of a qubit string as the length of the shortest *quantum* input to a universal quantum Turing machine that produces the target qubit string with high fidelity.

In related work, Paul Vitányi [102, 103] proposes to count the amount of *classical* information that is necessary for an approximating scheme of the quantum state, whereas here we consider the necessary amount of *quantum* information for a similar scheme. We argue that our definition is a natural and accurate representation of the amount of quantum information contained in a quantum state. Peter Gács [45] has also proposed two measures of 'quantum algorithmic entropy', which are based on the existence of a universal semi-density matrix. These measures partially correspond, it turns out, to Vitányi's definition and the one presented in this chapter, respectively.

7.1 Introduction

In classical computations, the Kolmogorov complexity of a finite string is a measure of its randomness.[30, 64, 94] The Kolmogorov complexity of x is the length of the shortest program which produces x as its output. It can be seen as a lower bound on the optimal compression that x can undergo, and its expectation in a probabilistic ensemble is close to the Shannon entropy.[35, 89]

Kolmogorov complexity has been shown to have a windfall of applications in fields as diverse as learning theory, complexity theory, combinatorics, graph theory, and analysis of algorithms.[67]

With the advent of quantum computation, it is natural to ask what is a good definition for the Kolmogorov complexity of quantum strings. Our goal is to argue that our definition is a natural and robust measure the amount of quantum information contained in a quantum string, and that it has several appealing properties.

Finding a robust definition for quantum Kolmogorov complexity has been of interest for many years (see for example [95].) Paul Vitányi [102, 103] has also proposed a definition for quantum algorithmic complexity. Our definition differs significantly from Vitányi's: the definition he proposes is a measure of the amount of *classical* information necessary to approximate the quantum state with a penalty depending in the error of the approximation. More recently, Peter Gács [45] has also proposed two definitions for quantum Kolmogorov complexity, both of which are based on the notion of a universal semi-density matrix. One of Gács' definitions is close to ours, while the other is related to Vitányi's.

7.2 Desired Properties

A good definition of quantum Kolmogorov complexity should meet the following fundamental criteria. These are intended to insure that it gives an accurate representation of the information content of a quantum string.

- It should be robust, that is, invariant up to an additive constant under the choice of the underlying universal quantum Turing machine.

- It should bear a strong relationship with quantum information theory.

- It should be closely related to classical complexity on classical strings.

However, quantum Kolmogorov complexity should not be expected to always behave the way classical Kolmogorov complexity does. The reader may want to bear in mind typical non-classical quantum phenomena such as the no-cloning theorem[107], whose consequences we will discuss in Section 7.14.

A first attempt at defining quantum Kolmogorov complexity of a qubit string X is to consider the length of the shortest quantum program that produces X as its output. There are many questions that arise from this 'definition'.

Bits or qubits? The first question to consider is whether we want to measure the amount of algorithmic information of a string in bits, or in qubits. Note that bit strings (programs) are countable, whereas qubit strings are uncountable, so any definition that measures in bits would have to overcome this apparent contradiction. Paul Vitányi [102, 103] considers classical descriptions of qubit strings, whereas we consider qubit descriptions.

Exact or inexact? What does 'produce' mean? Is a minimal program required to produce the string X exactly, or only up to some fidelity? In the latter case, is the fidelity a constant? Otherwise, how is it parameterized? (For exact simulation, we can only hope to simulate a subclass of the Turing machines, say by restricting the set of possible amplitudes. What would be a reasonable choice?) We will use an approximation scheme.

What model of computation? The size of quantum circuits is not an appropriate measure because it is possible to have a large circuit that has nevertheless a small description in terms of a generating computer program. For this reason we choice the Turing model of computation.

What is meant by 'quantum program?' A program for a quantum Turing machine is its input, and if we want to count program length in qubits, we must allow for 'programs' to be arbitrary qubit strings. (These can be viewed as programs whose code may include some auxiliary 'hard-coded' qubit strings.)

One-time description or multiple generation? In the classical setting, the program that prints a string $x \in \{0,1\}^n$ can be run as many times as desired. Because of the no-cloning theorem[107] of quantum physics however, we cannot assume that the shortest program can be run several times to produce several copies of the same string. (This will be due to the fact that it is not possible to recover the original program after it has produced its output.) There is also a second, but related, reason not to choose the multiple generation option. The complex-valued parameters $\alpha, \beta \in \mathbb{C}$ of a qubit $|q\rangle = \alpha|0\rangle + \beta|1\rangle$ can in principle contain an unbounded amount of information. If we would be able to reproduce the state q over and over again and without error, then we would be able to extract this information, and hence we would have to conclude that the single qubit q contains an unlimited amount of information. This contradicts the fact that the quantum mechanical system of a qubit q can only contain one bit of information.[53] For the above two reason, we will not require a 'reusability' condition.

7.3 Classical Kolmogorov complexity

The Kolmogorov complexity of a string, in the classical setting, is the length of the shortest program which prints this string on an empty input.[67]

Formally, this is stated first relative to a partial computable function, which as we know can be computed by a Turing machine.

Definition 21 (Kolmogorov complexity) *Fix a Turing machine T that computes a universal function Φ . For any pair of strings $x, y \in \{0,1\}^*$, the Kolmogorov complexity C of x relative to y (with respect to T) is defined as*

$$C_T(x|y) \quad := \quad \min\{\ell(p) : \Phi(\langle p, y\rangle) = x\}.$$

When y is the empty string, we simply write $C_T(x)$.

The key theorem on which rests the robustness of Kolmogorov complexity is the *invariance theorem*. This theorem states that the length of shortest programs does not depend by more than an additive constant on the underlying Turing machine. In the classical case, this theorem is proven with the existence of a particular type of universal Turing machine. This machine has two inputs: a finite description of the original Turing machine, and the program that this Turing machine executes to output the string.

More formally, the invariance theorem in the classical case can be stated as follows.

Fact 18 *There is a universal Turing machine U such that for every Turing machine T and pair of strings x, y,*

$$C_U(x|y) \quad \leq \quad C_T(x|y) + c_T,$$

where c_T is a constant depending only on T.

Giving an invariance theorem will be key to showing that quantum Kolmogorov complexity is robust.

Since for any string x of length n, $C(x) \leq n + O(1)$, a string which has complexity at least n is called *incompressible*. The existence of incompressible strings is a crucial fact of Kolmogorov complexity, and very useful in applications thereof.

Fact 19 *For every string length n, there is a string x of length n such that $C(x) \geq n$.*

The proof that there exists incompressible strings is a simple application of the pigeonhole principle. By comparing the number of strings of length n (2^n) and the number of programs of length smaller than n ($2^n - 1$ in total), one must conclude that there is at least one string of length n which is not the output of any of the program of length $< n$.

7.4 Quantum Information Theory

In this section we describe the quantum, or Von Neumann, entropy of ensembles, and important properties which will be used in the proofs of our results.

We start the section by defining the 'χ quantity' for ensembles.

Definition 22 (Holevo's chi quantity [53]) *For an ensemble $\mathcal{E} = \{(\rho_i, p_i)\}$, with $\rho = \sum_i p_i \rho_i$, Holevo's chi quantity equals*

$$\chi(\mathcal{E}) \quad := \quad S(\rho) - \sum_i p_i S(\rho_i).$$

Note that the χ quantity depends not only on ρ, but also on the specific pairs (ρ_i, p_i).

The following monotonicity property of Lindblad and Uhlmann will be very useful later on.

Fact 20 (Lindblad-Uhlmann monotonicity [68, 101]) *Let $\mathcal{E} = \{(\rho_i, p_i)\}$ be an ensemble, and $\boldsymbol{S}$ a completely positive, trace preserving mapping. For every such $\mathcal{E}$ and $\boldsymbol{S}$, it holds that: $\chi(\boldsymbol{S}(\mathcal{E})) \leq \chi(\mathcal{E})$, where $\boldsymbol{S}(\mathcal{E})$ is the transformed ensemble $\{(\boldsymbol{S}(\rho_i), p_i)\}$.*

The entropy of finite systems is robust against small changes. This continuity of S over the space of finite dimensional density matrices ρ is also called *insensitivity*, and is expressed by the following lemma.

Fact 21 (Insensitivity of Von Neumann entropy (see Section II.A in [105])) *If a sequence $\rho_1, \rho_2, \ldots$, has $\lim_{k\to\infty} \rho_k = \rho$, then also $\lim_{k\to\infty} S(\rho_k) = S(\rho)$.*

Proof: The convergence of $\rho_1, \rho_2, \ldots$ to ρ is understood to use some kind of norm for the density matrices that is continuous in the matrix entries $\langle i|\rho|j\rangle$. (The operator norm $|\rho| = \mathrm{tr}(\rho\rho^*)$, for example.) The entropy $S(\rho)$ is a continuous function of the finite set of eigenvalues of ρ. These eigenvalues are also continuous in the entries of ρ. □

Further background on these measures of quantum information and their properties can be found in [78, Chapter 5] and [105]. Another good source is Nielsen's thesis [73].

7.5 Symmetric Subspaces

We use the symmetric subspace of the Hilbert space to prove some of our results on copies of quantum states. Let $\mathcal{H}_d$ be a Hilbert space of dimension d with the basis states labeled $|1\rangle, \ldots, |d\rangle$. The *symmetric subspace* $\mathcal{H}_d^{\vee m}$ or $\bigvee^m \mathcal{H}_d$ of the m-fold tensor product space $\mathcal{H}_d^{\otimes m}$ contains the states that are invariant under permutation of its m parts. As a consequence, it is a subspace spanned by as many basis vectors as there are multisets of size m of $\{1, \ldots, d\}$. If $A = \{i_1, \ldots, i_m\}$ is such a multiset of $\{1, \ldots, d\}$, then $|A\rangle$ is the normalized superposition of all the different permutations of $i_1, \ldots, i_m$. The set of the different vectors $|A\rangle$ (ranging over the multisets A) is an orthogonal basis of the symmetric subspace $\mathcal{H}_d^{\vee m}$. This shows that the dimension of the symmetric subspace is $\binom{m+d-1}{d-1}$, because choosing such a multiset is equivalent to splitting a sequence of m zeroes into d (possibly empty) intervals. (If j_i is the size of the of ith interval, then this number also represents that the element $i \in \{1, \ldots, d\}$ appears j_i times in the multiset. The number of ways of splitting a sequence of size m into d intervals is $\binom{m+d-1}{d-1}$.)

The symmetric subspace $\mathcal{H}_d^{\vee m}$ is the smallest subspace of $\mathcal{H}_d^{\otimes m}$ that contains all the pure states of the form $|\phi\rangle^{\otimes m}$ for all $|\phi\rangle \in \mathcal{H}_d$.

As an example, consider the symmetric subspace $\mathcal{H}_2^{\vee 3}$. For every qubit $\alpha|0\rangle + \beta|1\rangle$, we can indeed express any three-fold copy in the four dimensions of $\mathcal{H}_2^{\vee 3}$:

$$\begin{aligned}(\alpha|0\rangle + \beta|1\rangle)^{\otimes 3} &= \alpha^3|000\rangle \ + \ \alpha^2\beta(|001\rangle + |010\rangle + |100\rangle) \ + \\ &\quad \alpha\beta^2(|011\rangle + |101\rangle + |110\rangle) \ + \ \beta^3|111\rangle \\ &= \alpha^3|\{0,0,0\}\rangle \ + \ \alpha^2\beta\sqrt{3}|\{0,0,1\}\rangle \ + \\ &\quad \alpha\beta^2\sqrt{3}|\{0,1,1\}\rangle \ + \ \beta^3|\{1,1,1\}\rangle.\end{aligned}$$

We thus reach the important conclusion that there exists a unitary transformation from the 3 qubits of the symmetric subspace $\mathcal{H}_2^{\vee 3}$ to the two qubits of the space spanned by the vectors $|\{0,0,0\}\rangle$, $|\{0,0,1\}\rangle$, $|\{0,1,1\}\rangle$ and $|\{1,1,1\}\rangle$. The generalization of this compression result for all values d and m is presented in Section 7.14. For more information on the symmetric subspace and its properties, see the paper by Barenco *et al.* [8].

7.6 Accumulation of Errors

The following lemma is used to bound the error introduced when composing two inexact quantum procedures.

Lemma 18 (Fidelity of composition) *Let ρ_1, ρ_2 and ρ_3 be three density matrices.*

$$\textit{If } \mathrm{F}(\rho_1,\rho_2) \geq 1-\delta_1 \quad \textit{and} \quad \mathrm{F}(\rho_2,\rho_3) \geq 1-\delta_2,$$

then $\mathrm{F}(\rho_1,\rho_3) \geq 1-2\delta_1-2\delta_2$.

Proof: We say that a bi-partite, pure state ϕ^{AB} is the 'purification' of the (mixed) state ρ if we obtain ρ by tracing out the B part of ϕ^{AB}: $\rho = \mathrm{tr}_B(\phi^{AB})$. The lemma now follows from Uhlmann's theorem[44], which says that the fidelity between two (mixed) states ρ_1 and ρ_2 equals the maximum 'pure state fidelity' $|\langle\phi_1|\phi_2\rangle|$, with ϕ_i the purifications of ρ_i. □

This lemma is especially powerful in combination with the monotonicity property: the result that the fidelity between two states cannot decrease under a quantum mechanical transformation.[9] It enables us to prove the following result that bounds the error of two consecutive operations.

Lemma 19 (Fidelity after two transformations) *If U_1 and U_2 are two quantum mechanical transformations and ρ_1, ρ_2, ρ_3 are density matrices such that*

$$\mathrm{F}(\rho_2,U_1(\rho_1)) \geq 1-\delta_1 \quad \textit{and} \quad \mathrm{F}(\rho_3,U_2(\rho_2)) \geq 1-\delta_2, \tag{7.1}$$

then, for the combined transformation U_2U_1,

$$\mathrm{F}(\rho_3,U_2\cdot U_1(\rho_1)) \geq 1-2\delta_1-2\delta_2. \tag{7.2}$$

Proof: From $\mathrm{F}(\rho_2,U_1(\rho_1)) \geq 1-\delta_1$, and the nondecreasing property of the fidelity it follows that $\mathrm{F}(U_2(\rho_2),U_2\cdot U_1(\rho_1)) \geq 1-\delta_1$. Lemma 18 concludes the proof. □

In order to give bounds on the complexity of several copies of a state, as we do in Section 7.14, we also need the following bound on the total error in the n-fold tensor product of the approximation of a given state.

Lemma 20 (Fidelity of copies) *Let $\rho_1^{\otimes n}$ and $\rho_2^{\otimes n}$ be the n-fold copies of the mixed states ρ_1 and ρ_2, then $\mathrm{F}(\rho_1^{\otimes n},\rho_2^{\otimes n}) = (\mathrm{F}(\rho_1,\rho_2))^n$. Hence, if $\mathrm{F}(\rho_1,\rho_2) \geq 1-\delta$, then $\mathrm{F}(\rho_1^{\otimes n},\rho_2^{\otimes n}) \geq 1-n\delta$.*

Proof: Apply the matrix properties $A^{\otimes n}B^{\otimes n} = (AB)^{\otimes n}$ and $\mathrm{tr}(A^{\otimes n}) = (\mathrm{tr}(A))^n$ to the definition of Equation 1.4 to obtain:

$$\begin{aligned}\mathrm{F}(\rho_1^{\otimes n}, \rho_2^{\otimes n}) &= \mathrm{tr}\left(\sqrt{\sqrt{\rho_1^{\otimes n}} \cdot \rho_2^{\otimes n} \cdot \sqrt{\rho_1^{\otimes} n}}\right)\\ &= \mathrm{tr}\left(\sqrt{\sqrt{\rho_1} \cdot \rho_2 \cdot \sqrt{\rho_1}}^{\otimes n}\right)\\ &= \mathrm{tr}\left(\sqrt{\sqrt{\rho_1} \cdot \rho_2 \cdot \sqrt{\rho_1}}\right)^n.\end{aligned}$$

□

7.7 Quantum Kolmogorov Complexity

We define the *quantum Kolmogorov complexity* QC of a string of qubits X, relative to a quantum Turing machine M, as the length of the shortest qubit string that, when given as input to M, produces on the output register the qubit string X. (Note that we only allow M that have computable transition amplitudes. See the articles [22, 38], and particularly Definition 3.2.2 in [22], for a further description of this computational model.)

7.8 Input/Output Conventions

First we will specify in more detail what is meant by the 'input' and 'output' of a quantum computation.

We consider quantum Turing machines with two heads on two one-way infinite tapes: one input/work tape, and one output tape. We allow both tapes to be changed because we want to be able to move the input qubits to the output tape.

For a QTM M with a single input, when we say M starts with input Y, we mean that M starts with the quantum state $|Y\$00\cdots\rangle$ on its input tape, and $|00\cdots\rangle$ on the output tape. The $\$$ symbol is a special endmarker (or blank) symbol.

Note that testing for the end of the input can be done without disturbing the input, since we assume that the '\$' state is orthogonal to the '0' and '1' states. (This is analogous to the classical case, where where Turing machine inputs are encoded in a three-letter alphabet; nevertheless we consider the actual input to be encoded only over the characters 0 and 1.) A string is a proper input if there is only one position on the tape where the the endmarker symbol '\$' appears. We dismiss any non-proper inputs as this would allow the endmarker to be in a superposition of several positions, which cannot be checked by the quantum Turing machine.

For a QTM with multiple inputs, we assume that there is a convention for encoding the multiple inputs so that they can be individually recovered. For example, when we write $M(Y_1, Y_2)$, we may assume that the input tape is initialized to

$|1^{\ell(Y_1)}0Y_1Y_2\$00\cdots\rangle$: the sequence of ones $1^{\ell(Y_1)}$ is unambiguously delimited by the leftmost zero in the string, and with the thus obtained value $\ell(Y_1)$ we can separate Y_1 and Y_2 from the remainder of the sequence. Likewise, for multiple outputs, if we write $M(Y_1, Y_2) = (X_1, X_2)$, we mean that X_1 and X_2 must be encoded according to a prearranged convention so that X_1 and X_2 can be recovered individually from the output tape. (We do not define prefix-free complexity in this thesis.)

We let $M^T(X)$ denote the contents of the output tape after T steps of computation. We consider only QTMs that do not modify their output tape after they have 'halted'. (Because of reversibility, they may modify the input/work tape after reaching the halting state.) The output string $M(X)$ equals the content of the output tape at any time after M has stopped changing this tape. We allow the content of the output tape to be entangled with the input/work tape after M has halted. If this is the case, then the output $M(X)$ is the mixed state that one obtains by 'tracing out' the input/work tape. Note that this output does not change when the computer continues to change the input/work tape after is has officially halted.

7.9 Defining Quantum Kolmogorov Complexity

For some fidelity function $f : \mathbb{N} \to [0, 1]$ we will now define the corresponding quantum Kolmogorov complexity.

Definition 23 (Quantum Kolmogorov complexity with fidelity f) *For any quantum Turing machine M and qubit string X, the f-approximation quantum Kolmogorov complexity, denoted $QC_M^f(X)$, is the length of the smallest qubit string P such that for any fidelity parameter k we have* $\mathrm{F}(X, M(P, 1^k)) \geq f(k)$.

Note that we require that the same string P be used for all approximation parameters k. This way we do not have to consider a sequence of programs $P_1, P_2, \ldots$, which may not have a well defined limiting size $\lim_{k\to\infty} \ell(P_k)$.

Note also that we allow both the string X, the program P, and the output $M(P, 1^k)$ to be mixed states for the following reasons. There is no reason why the approximation $M(P, 1^k)$ of a pure state X has to be pure as well. By allowing mixed states we avoid this problem, and, as a bonus, get also a definition for the complexity of mixed states. Because the fidelity and the time evolution of M is properly defined for mixtures this causes no serious problems. (Clearly, the program P_ρ that simply moves ρ from the input to the output tape will have to be mixed as well, which explains the necessity of mixed input strings.)

We will say that program P 'M-computes X with fidelity $f(k)$' if for all k we have $\mathrm{F}(X, M(P, 1^k)) \geq f(k)$. If f is the constant function 1, we thus have the following definition.

Definition 24 (Quantum Kolmogorov complexity with perfect fidelity) *The perfect fidelity quantum Kolmogorov complexity is $QC_M^1(X)$.*

The problem with this definition is that we do not know whether an invariance theorem can be given for this perfect-fidelity Kolmogorov complexity. This is because the invariance theorems that are known for quantum computers deal with *approximating* procedures rather than with exact simulations. We therefore prove an invariance theorem for a weaker, limiting version, where the output of M must have high fidelity with respect to the target string X: $\mathrm{F}(X, M(P, 1^k)) \approx 1$.

Definition 25 (Quantum Kolmogorov complexity with bounded fidelity) *For a imperfect fidelity $\epsilon < 1$, the complexity $QC^{\epsilon}_M(X)$ is the constant-fidelity quantum Kolmogorov complexity.*

Again there are problems with this definition. First, it may be the case that some strings are very easy to describe up to a given constant, but inherently very hard to describe for a smaller error. Second, it may be the case that some strings are easier to describe up to a given constant on one machine, but not on another machine. For these two reasons, this definition does not appear to be robust.

A stronger notion of approximability is the existence of an approximation *scheme.* (See, for example, the book by Garey and Johnson [46, Chapter 6] for more on approximation algorithms and approximation schemes.) For constant-approximability, different algorithms (with different sizes) can exist for different constants. In an approximation scheme, a single program takes as auxiliary input an approximation parameter k, and produces an output that approximates the value we want within the approximation parameter. This is the model we wish to adopt for quantum Kolmogorov complexity.

Definition 26 (Quantum Kolmogorov complexity with fidelity converging to one) *The complexity $QC^{\uparrow 1}_M(X)$ is equal to $QC^{f}_M(X)$, where $f(k) = 1 - \frac{1}{k}$.*

We choose to encode the fidelity parameter in unary, and the convergence function to be $f(k) = 1 - \frac{1}{k}$ so that the model remains robust when polynomial time bounds are added. We discuss this further in Section 7.10.

We may also define $QC^{\uparrow 1}_M(X|Y)$, the complexity of producing X when Y is given as an auxiliary input, in the usual way.

7.10 Invariance

To show that our definition is robust we must show that the complexity of a qubit string is minimized by a particular type of universal machine, and is invariant, up to an additive constant, under the choice of a different Turing machine.

We use the following result, proved in the paper of Bernstein and Vazirani [22]. To be precise, we use the notation $\overline{M}$ to denote the classical description of the quantum Turing machine M. (Recall that we only consider quantum Turing machines whose amplitudes can be computed to arbitrary precision with a finite classical description.)

Fact 22 (Universal quantum Turing machine [22]) *There exists a universal quantum Turing machine U with a finite classical description such that the following holds. For any quantum Turing machine M (which has a finite classical description), for any pure state X, for any approximation parameter k, and any number of time steps T, we have $\mathrm{F}(U(\overline{M}, X, 1^k, T), M^T(X)) \geq 1 - \frac{1}{k}$. (Remember that M^T is the contents of the output tape of M after T time steps.)*

Theorem 16 (Quantum invariance theorem) *There is a universal quantum Turing machine U such that for any quantum Turing machine M and qubit string X,*

$$QC_U^{\uparrow 1}(X) \quad \leq \quad QC_M^{\uparrow 1}(X) + c_M,$$

where c_M is a constant depending only on M.

Proof: The proof of this theorem follows from the existence of a universal quantum Turing machine, as mentioned here in Fact 22. Let U be this UTM. The constant c_M represents the size of the finite description $\overline{M}$ that U requires to calculate the transition amplitudes of the machine M. Let P be the state that witnesses that $QC_M^{\uparrow 1}(X) = \ell(P)$, and hence $\mathrm{F}(X, M(P, 1^k)) \geq 1 - \frac{1}{k}$ for every k.

With the description $\overline{M}$ (with length c_M), U can simulate with arbitrary accuracy the behavior of M. Specifically, U can simulate machine M on input $(P, 1^{4k})$ with a fidelity of $1 - \frac{1}{4k}$. Therefore, by Lemma 18, $\mathrm{F}(X, U(M, P, 1^{4k})) \geq 1 - \frac{1}{k}$. □

The same holds true for the conditional complexity, that is, there exists a UTM U such that for all quantum machines M and quantum strings X, Y we have $QC_U^{\uparrow 1}(X|Y) \leq QC_M^{\uparrow 1}(X|Y) + c_M$.

Henceforth, we will fix a universal quantum Turing machine U and simply write $QC(X)$ instead of $QC_U^{\uparrow 1}(X)$. Likewise we write $QC(X|Y)$ instead of $QC_U^{\uparrow 1}(X|Y)$. We also abuse notation and write M instead of $\overline{M}$ to represent the code of the quantum Turing machine M used as an input to the universal Turing machine.

The simplest application of the invariance theorem is the following lemma.

Lemma 21 *There exists a constant c, such that for any qubit string X, $QC(X) \leq \ell(X) + c$. The value of c depends only on our choice of the underlying universal Turing machine.*

Proof: Consider the quantum Turing machine M that moves its input to the output tape, yielding $QC_M(X) = \ell(X)$. The result follows by invariance. □

We may also define time-bounded QC is the usual way, that is, fix $T : \mathbb{N} \to \mathbb{N}$ a fully-time-computable function. Then $QC^T(X|Y)$ is the length of the shortest program which on input $(Y, 1^k)$, produces X on its output tape after $T(\ell(X) + \ell(Y))$ computation steps. The simulation of Bernstein and Vazirani entails a polynomial time blowup (polynomial in the length $\ell(Y)$ of the input and the length k of the fidelity parameter), so there will be only a polynomial time blowup in the corresponding invariance theorem.

7.11 Properties of Quantum Kolmogorov Complexity

In this part we compare classical and quantum Kolmogorov complexity by examining several properties of both. We find that many of the properties of the classical complexity, or natural analogs thereof, also hold for the quantum complexity. A notable exception is the complexity of m-fold copies of arbitrary qubit strings.

7.12 Correspondence for Classical Strings

We would like to show that for classical states, classical and quantum Kolmogorov complexity coincide, up to a constant additive term.

Lemma 22 *There is a constant c, such that for every finite, classical string x, it holds that $QC(x) \leq C(x) + c$.*

(The constant depends only on the underlying universal Turing machine.)
Proof: This is clear: the universal quantum computer can also simulate any classical Turing machine. □

The converse is also true, as shown by Peter Gács [45].

Fact 23 *(See [45] for the proof.) There is a constant c, such that for every finite, classical string x, it holds that $C(x) \leq QC(x) + c$.*

7.13 Quantum Incompressibility

In this section, we show that there exist quantum-incompressible strings. Our main theorem is a very general form of the incompressibility theorem with some useful special cases as corollaries.

Assume we want to consider the minimal-length programs that describe a set of quantum states. In general, these may be pure or mixed states. We will use the following notation throughout the proof. The mixed states $\rho_1, \ldots, \rho_M$ be the target strings (those we want to produce as output). Their minimal-length programs will be $\sigma_1, \ldots, \sigma_M$, respectively. The central idea is that if the states ρ_i are sufficiently different, then the programs σ_i must be different as well. We turn this into a quantitative statement with the use of the insensitive chi quantity in combination with the monotonicity of quantum mechanics.

Earlier, Michał Horodecki used a similar technique to prove a closely related result [55], which shows that the Holevo quantity is a lower bound for the optimal compression rate for ensemble of mixed states.

Theorem 17 *For any set of strings $\rho_1,\dots,\rho_M$ such that $\forall i, QC(\rho_i) \leq l$, this l is bounded from below by*

$$l \;\geq\; S(\rho) - \tfrac{1}{M}\sum_i S(\rho_i),$$

where ρ is the 'average' density matrix $\rho = \frac{1}{M}\sum_i \rho_i$. (Stated slightly differently, this says that there is an i such that $QC(\rho_i) \geq S(\rho) - \frac{1}{M}\sum_i S(\rho_i)$.)

Proof: Take $\rho_1,\dots,\rho_M$ and their minimal programs $\sigma_1,\dots,\sigma_M$ (and hence $QC(\rho_i) = \ell(\sigma_i)$). Let $\boldsymbol{S}^k$ be the completely positive, trace preserving map corresponding to the universal QTM U with fidelity parameter k. With this, we define the following three uniform ensembles:

- the ensemble $\mathcal{E} = \{(\rho_i, \frac{1}{M})\}$ of the original strings,
- $\mathcal{E}_\sigma$ the ensemble of programs $\{(\sigma_i, \frac{1}{M})\}$, and
- the ensemble of the k-approximations $\tilde{\mathcal{E}}^k = \boldsymbol{S}^k(\mathcal{E}_\sigma) = \{(\tilde{\rho}_i^k, \frac{1}{M})\}$, with $\tilde{\rho}_i^k = \boldsymbol{S}^k(\sigma_i)$.

By the monotonicity of Fact 20 we know that for every k, $\chi(\tilde{\mathcal{E}}^k) \leq \chi(\mathcal{E}_\sigma)$. The chi quantity of the ensemble $\mathcal{E}_\sigma$ is upper bounded by the maximum size of its strings: $\chi(\mathcal{E}_\sigma) \leq \max_i\{\ell(\sigma_i)\} \leq l$. Thus the only thing that remains to be proven is that $\chi(\tilde{\mathcal{E}}^k)$, for sufficiently big k, is 'close' to $\chi(\mathcal{E})$. This will be done by using the insensitivity of the Von Neumann entropy.

By definition, for all i, $\lim_{k\to\infty} \mathrm{F}(\rho_i, \tilde{\rho}_i^k) = 1$, and hence $\lim_{k\to\infty} \tilde{\rho}_i^k = \rho_i$. Because the ensembles $\mathcal{E}$ and $\tilde{\mathcal{E}}^k$ have only a finite number (M) of states, we can use Lemma 21, to obtain $\lim_{k\to\infty} \chi(\tilde{\mathcal{E}}^k) = \chi(\mathcal{E})$. This shows that for any $\delta > 0$, there exists a k such that $\chi(\mathcal{E}) - \delta \leq \chi(\tilde{\mathcal{E}}^k)$. With the above inequalities we can therefore conclude that $\chi(\mathcal{E}) - \delta \leq l$ holds for arbitrary small $\delta > 0$, and hence that $l \geq \chi(\mathcal{E})$. □

The following four lemmas are straightforward with the above theorem.

Lemma 23 *For every length n, there is an incompressible classical string of length n.*

Proof: Apply Theorem 17 to the set of classical strings of n bits: $\rho_x = |x\rangle\langle x|$ for all $x \in \{0,1\}^n$. All ρ_x are pure states with zero Von Neumann entropy, hence the lower bound on l reads $l \geq S(\rho)$. The average state $\rho = 2^{-n}\sum_x |x\rangle\langle x|$ is the total mixture $2^{-n}I$ with entropy $S(\rho) = n$, hence indeed $l \geq n$. □

Lemma 24 *For any set of orthogonal pure states $|\phi_1\rangle,\dots,|\phi_M\rangle$, the smallest l such that for all i, $QC(\phi_i) \leq l$ is at least $\log M$. (Stated differently, there is an i such that $QC(\phi_i) \geq \log M$.)*

Proof: All the pure states have zero entropy $S(\phi_i) = 0$, hence by Theorem 17: $l \geq S(\rho)$. Because all ϕ_is are mutually orthogonal, this Von Neumann entropy $S(\rho)$ of the average state $\rho = \frac{1}{M}\sum_i |\phi_i\rangle\langle\phi_i|$ equals $\log M$. □

Lemma 25 *For every length n, at least $2^n - 2^{n-c}+1$ mutually orthogonal qubit strings of length n have complexity at least $n - c$.*

Lemma 26 *For any set of pure states $|\phi_1\rangle, \dots, |\phi_M\rangle$, the smallest l such that for all i, $QC(\phi_i) \leq l$ is at least $S(\rho)$, where $\rho = \frac{1}{M}\sum_i |\phi_i\rangle\langle\phi_i|$.*

7.14 The Complexity of Copies

It is trivial to copy a classical bit string x to the m-fold state $x^{\otimes m}$. As long as we know the integer m, the complexity of $x^{\otimes m}$ is no bigger than that of the single copy x, or in Kolmogorov complexity terms: $C(x^{\otimes m}|m) \leq C(x) + O(1)$. This no longer holds in the case of quantum information, as it is in general not possible to copy an unknown quantum state.[107] Typically for a quantum state X, the complexity $QC(X^{\otimes m}|m)$ will grow as m gets bigger. This should not surprise us because a large number m of copies enables us to estimate the amplitudes of X more accurately than a single copy would. Hence, we can ‘extract’ more information from $X^{\otimes m}$ if we have more copies of X. An obvious upper bound on the quantum Kolmogorov complexity of $X^{\otimes m}$ is $QC(X^{\otimes m}|m) \leq m \cdot QC(X)$. The two main theorems of this section tell us that, despite the ‘no cloning’ phenomenon of quantum mechanics, it is possible to compress copies of pure states. This result is established with the help of the theory of symmetric subspaces. We start with the general upper bound.

Theorem 18 *There exists a constant c, such that for an arbitrary pure state X and integer m it holds that*

$$QC(X^{\otimes m}|m) \quad \leq \quad \log \binom{m + 2^{QC(X)} - 1}{2^{QC(X)} - 1} + c, \tag{7.3}$$

and hence $QC(X^{\otimes m}) \leq \log \binom{m+2^{QC(X)}-1}{2^{QC(X)}-1} + O(\log m)$.

Proof: First we outline the proof, omitting the effect of the approximation. Consider a pure qubit string X whose minimal-length program is P_X. To produce m copies of X, it suffices to produce m copies of P_X and execute these m programs.

We can always assume that this P_X is a pure state, because for a mixture of programs, any of the pure programs in the mixtures will produce X as well. Let l be the length $QC(X)$ of P_X; we denote the 2^l-dimensional Hilbert space by $\mathcal{H}$. Consider $\mathcal{H}^{\otimes m}$, the m-fold tensor product of $\mathcal{H}$. The symmetric subspace $\mathcal{H}^{\vee m}$ is d-dimensional, where $d = \binom{m+2^l-1}{2^l-1}$. The sequence $P_X^{\otimes m}$ sits in this symmetric subspace, and can therefore be encoded exactly using $\log d + O(\log m)$ qubits, where the m term is used to describe the rotation from the d-dimensional space to the m copies in $\mathcal{H}^{\otimes m}$. Hence, given m, the quantum Kolmogorov complexity of $X^{\otimes m}$ is bounded from above by $\log d + O(1)$ qubits.

For the full proof, we will need to take into account the effect of the imperfect fidelities and prove that we can reach a fidelity not smaller than $1 - \frac{1}{k}$.

The first part of the computation consists of the mapping from the d dimensions to the symmetric subspace $\mathcal{H}^{\vee m}$. This is the transformation $|i\rangle \mapsto |A_i\rangle$ for $1 \leq i \leq d$, which labels all the multisets $A_i \subseteq \{1, \ldots, 2^l\}$ of size m. We approximate this unitary transformation with enough accuracy such that the output has fidelity $\geq 1 - \frac{1}{4k}$ with the perfect state $P_X^{\otimes m}$.

Next, we execute the programs P_X with a fidelity parameter of $4km$. Hence the joint, m-fold evolution $U_2^{\otimes m}$ establishes $\mathrm{F}(X^{\otimes m}, U_2^{\otimes m}(P_X^{\otimes m})) \geq 1 - \frac{1}{4k}$ (Lemma 20).

We finish the proof by employing Lemma 19, which tells us that the overall fidelity-error of the above two transformations cannot be bigger than $\frac{1}{k}$ □

This upper bound is also very close to being tight for some X, as we show in the next theorem.

Theorem 19 (Incompressibility for copies of quantum states) *For every m and n, there is an n-qubit state X such that $QC(X^{\otimes m}) \geq \log \binom{m+2^n-1}{2^n-1}$.*

Proof: Fix m and n and let $\mathcal{H}$ be the 2^n-dimensional Hilbert space. Consider the (continuous) ensemble of all m-fold tensor product states $X^{\otimes m}$: $\mathcal{E} = \{(X^{\otimes m}, \mu)\}$, where $\mu^{-1} = \int_{X \in \mathcal{H}} dX$ is the appropriate normalization factor. The corresponding average state is calculated by the integral $\rho = \mu \int_{X \in \mathcal{H}} X^{\otimes m} dX$. This mixture is the totally mixed state in the symmetric subspace $\mathcal{H}^{\vee m}$ (see Section 3 in [106]), and hence has entropy $S(\rho) = \log \binom{m+2^n-1}{2^n-1}$. Because all $X^{\otimes m}$ are pure states, we can use Lemma 26 to prove the existence of an X for which $QC(X^{\otimes m}) \geq \log \binom{m+2^n-1}{2^n-1}$. □

The results of this section can be viewed as a refinement of the no-cloning theorem, in the following sense. The quantity $QC(X^{\otimes m}|m)$, for any state X, gives a measure of how clonable that particular state is. Theorem 19 tells us that there exist strings that are ‘maximally non-clonable’.

7.15 Subadditivity

Consider the following subadditivity property of classical Kolmogorov complexity.

Fact 24 *For any x and y, $C(x,y) \leq C(x) + C(y|x) + O(\log(C(x)))$.*

In the classical case, we can produce x, and then produce y from x, and print out the combination of x and y. In the quantum case, producing Y from X may destroy X. In particular, with $X = Y$, the immediate quantum analog of Fact 24 would contradict the $m = 2$ case of Theorem 19.

A natural quantum extension of this result is as follows.

Lemma 27 *For any pair of quantum strings X, Y, we have $QC(X,Y) \leq QC(X,X) + QC(Y|X) + O(\log(QC(X)))$.*

7.16 The Complexity of Correlations

In this section we will use quantum Kolmogorov complexity to quantify the complexity of the correlation between two systems. For a bipartite state ρ_{AB} we denote this quantity by $QCor(\rho_{AB})$, which is defined as follows.

Definition 27 (Quantum Kolmogorov Complexity of Correlations) *Consider a bipartite state ρ_{AB} of $n+m$ qubits where n qubits are on A's side and B has the remaining m qubits. The quantum Kolmogorov complexity $QCor$ of the correlation between A and B is defined by*

$$QCor(\rho_{AB}) \quad := \quad QC(\rho_{AB}|\rho_A, \rho_B),$$

where $\rho_A = \mathrm{tr}_B(\rho_{AB})$ and $\rho_B = \mathrm{tr}_A(\rho_{AB})$.

Because the complexity $QCor(\rho_{AB})$ can never be bigger than $QC(\rho_{AB})$, the following general upper bound holds.

Lemma 28 *There exists a constant c such that for every bipartite, $n+m$-qubit state ρ_{AB} we have*

$$QCor(\rho_{AB}) \quad \leq \quad n+m+c. \tag{7.4}$$

Proof: Apply Lemma 21 to the relation $QCor(\rho_{AB}) \leq QC(\rho_{AB})$. □
The gap between the correlation complexity $QCor$ and the Kolmogorov complexity can be made arbitrarily big as is shown by the next lemma.

Lemma 29 *There exists a constant c such that for any combination of lengths n and m, there is an $n+m$-qubit string ρ_{AB} with maximum Kolmogorov complexity $QC(\rho_{AB}) \geq n+m$, combined with a constant lower bound on the complexity of the correlation $QCor(\rho_{AB}) \leq c$.*

Proof: Consider the set of classical strings of length $n+m$. Clearly, these states can be expressed as tensor products $X_{AB} = X_A \otimes X_B$, where X_A (X_B) are n (m) bit strings. By the program of size c that moves the inputs X_A and X_B to the output tape (thus producing X_{AB}) we obtain $QCor(X_{AB}) = QC(X_{AB}|X_A, X_B) \leq c$. On the other hand, by Lemma 23, at least one of these strings X_{AB} also has to obey $QC(X_{AB}) \geq n+m$. □

The central idea behind the definition of $QCor$ is that we consider the complexity of the correlations 'high' when the partial states ρ_A and ρ_B do not contain much information about the total configuration ρ_{AB}. In this sense it is possible that all the complexity of a state is contained in its correlations. The following lemma expresses this result.

Lemma 30 *For every length n, there exists a bipartite, $n+n$-qubit state ρ_{AB} with maximum correlation complexity $QCor(\rho_{AB}) \geq 2n$.*

Proof: First we consider the $n = 1$ case of two distributed qubits. Take the four Bell states $|\phi^1_{AB}\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$, $|\phi^2_{AB}\rangle = \frac{1}{\sqrt{2}}(|00\rangle - |11\rangle)$, $|\phi^3_{AB}\rangle = \frac{1}{\sqrt{2}}(|01\rangle + |10\rangle)$, and $|\phi^4_{AB}\rangle = \frac{1}{\sqrt{2}}(|01\rangle - |10\rangle)$. As these states are mutually orthogonal, we can use the uniform source $\mathcal{E} = \{(\phi^i_{AB}, \frac{1}{4})\}$ to encode two bits of information.[21] It is also straightforward to see that all the partially traced out states are identical to the same totally mixed qubit: $\phi^i_A = \phi^i_B = \frac{1}{2}(|0\rangle\langle 0| + |1\rangle\langle 1|) = \frac{1}{2}I$ for all i. Hence, for one of the ϕ's we must have $QCor(\phi^i_{AB}) = QC(\phi^i_{AB}|\frac{1}{4}I \otimes I) \geq 2$.

This result easily generalizes to the $n + n$-qubit case if we take the n-fold tensor product of the above source. We can use the words of this $\mathcal{E}^{\otimes n}$ to encode $2n$ bits of information, while the partially traced out words all equal the totally mixed n qubit state $2^{-n}I$. This shows that for at least one of the words it must hold that its correlation complexity is not smaller than $2n$. □

It would be incorrect to think that the complexity $QCor$ is 'yet another measure of entanglement'. It is true that tensor product states $X_A \otimes X_B$ have a low correlation complexity, but so have highly entangled states like $(\frac{1}{\sqrt{2}}|0_A 0_B\rangle + \frac{1}{\sqrt{2}}|1_A 1_B\rangle)^{\otimes n}$. Moreover, the definition also covers the complexity of purely classical correlations. Rather than quantifying entanglement, we expect the above definition to be useful in the context of 'communication complexity theory'. The last section of this chapter will explain this point further.

7.17 Extensions and Future Work

We have argued that the QC of Definition 26 is a robust notion of Kolmogorov complexity for the quantum setting. It would be interesting to see if an invariance theorem can be shown for the ideal quantum Kolmogorov complexity of Definition 24. It would also be interesting to see if the invariance theorem (Theorem 16) can be improved in general.

Kolmogorov complexity in the classical setting is a good tool for showing lower bounds in computational complexity. For instance, one can show lower bounds in classical communication by using classical Kolmogorov complexity. A simple example is the following lower bound on the communication complexity of the equality function. Assume that there is a protocol that decides whether two strings of length n are equal, in which t bits are exchanged. Consider an incompressible string x of length n, and simulate the protocol on input (x, x). Let T be the transcript of the communication on that input. Now we argue that the Kolmogorov complexity of the string can be bounded above by a function of t. To print x, we use the transcript and the protocol to find x as follows. Without loss of generality, assume that the second player always decides whether or not to accept the input. For every candidate z for x, simulate the protocol for the second player on input z, and use the transcript to obtain the communication that the second player would have received from the first player. Because the protocol is sound, the simulation will only accept if $z = x$. We output whenever a string is found that causes the protocol to accept. This program which prints x is of size (roughly) t, and therefore we have $n \leq C(x) \leq t$, from which we can conclude that the communication complexity of the equality function is at least n.

Could a similar argument be carried over to the quantum setting? If so, then by applying this framework to other problems in quantum complexity, quantum Kolmogorov complexity could become a powerful new tool in proving lower bounds.

The number of applications of classical Kolmogorov complexity is countless, and it is our hope that this definition will lead to a similar wide variety of applications in quantum complexity theory.

Appendix A

Complexity Classes and Reductions

A.1 Complexity Classes

P: (Classical) polynomial time

NP: (Classical) nondeterministic, polynomial time

EQP: Exact, quantum, polynomial time

FP: Exact, polynomial time functions

FEQP: Exact, quantum, polynomial time functions

EXP: Exponential time

PSPACE: Polynomial space

PP: Probabilistic, polynomial time

C^A: The class C with queries to the set A

$\mathsf{C}_{||}^A$: The class C with non-adaptive queries to the set A

$\mathsf{C}^{A[k]}$: The class C with not more than k queries to the set A

$\mathsf{C}_{||}^{A[k]}$: The class C with not more than k non-adaptive queries to the set A

Sigma classes: $\Sigma_0^p = \mathsf{P}$, and $\Sigma_{i+1}^p = \mathsf{NP}^{\Sigma_i^p}$

Delta classes: $\Delta_{i+1}^p = \mathsf{P}^{\Sigma_i^p}$

Theta classes: $\Theta_{i+1}^p = \mathsf{P}_{||}^{\Sigma_i^p}$

Computable Decision Problems: Σ_0

A.2 Reductions

many-one reducible “$\leq_m^p$”: $B{\leq_m^p}A$, if there exists a poly-time reduction τ such that $x \in B$ if and only if $\tau(x) \in A$.

truth-table reducible “$\leq_{tt}^p$”: $B{\leq_{tt}^p}A$, if there exists an algorithm for B that answers the question “$x \in B$?” with polynomial many non-adaptive queries to A.

Turing reducible “$\leq_T^p$”: $B{\leq_T^p}A$, if there exists an algorithm for B that answers the question “$x \in B$?” with polynomial many queries to A.

A.3 Query Complexity

n-bit black-box An (unknown) function $f : \{1, \ldots, n\} \to \{0, 1\}$. We say that the black-box ‘contains’ the n-bit string $f(1), \ldots, f(n)$.

(Probabilistic) query complexity The number of times, as a function of n, that the black-box f has to be queried to solve a problem (with high probability). The worst-case distribution over all possible black-boxes is assumed.

Unstructured Problem A problem that is defined for all strings $\{0, 1\}^n$, and hence for all black boxes.

Structured Problem A problem that is only defined on a proper subset of $\{0, 1\}^n$. We say that there is a ‘promise’ on the input of the problem.

Appendix B

Properties of Matrices

B.1 Properties and Transformations

For a finite dimensional, complex valued matrix $A \in M_n(\mathbb{C})$, we can define its

set of complex matrices $\boldsymbol{M_n}(\mathbb{C})$: the n by n dimensional matrices with complex valued entries

set of real matrices $\boldsymbol{M_n}(\mathbb{R})$: the n by n dimensional matrices with real valued entries

transpose $\boldsymbol{A^T}$: defined by $(A^T)_{ij} = A_{ji}$

conjugate transpose A^* defined by $(A^*)_{ij} = (A_{ji})^*$

adjoint $\boldsymbol{A^*}$ identical to the *conjugate transpose*

inverse $\boldsymbol{A^{-1}}$ For non-singular matrices $A \in M_n(\mathbb{C})$ the inverse is defined by $A \cdot A^{-1} = \mathrm{I}_n$; otherwise A^{-1} is undefined.

square root $\boldsymbol{\sqrt{A}}$ The square root of A is the matrix such that $\sqrt{A} \cdot \sqrt{A} = A$. For a diagonal matrix D, we thus have $(\sqrt{D})_{ij} = \sqrt{D_{ij}}$. Using the spectral decomposition we can see that the root of normal matrices can be expressed as $\sqrt{A} = \sqrt{U \Lambda U^*} = U \sqrt{\Lambda} U^*$.

trace $\mathtt{tr}(\boldsymbol{A})$ the value $\sum_{i=1}^{n} A_{ii}$

A finite dimensional, complex valued, matrix $A \in M_n(\mathbb{C})$ can have the following properties.

Diagonal: if $A_{ij} = 0$ for every $i \neq j$

Hermitian: if $A = A^*$

Normal: if $A \cdot A^* = A^* \cdot A$

Unitary: if $A \cdot A^* = \mathrm{I}_n$. The set of unitary $n \times n$ matrices is denoted by $\mathsf{U}(n)$.

Special Rotations: if a real-values matrix obeys $A \cdot A^T = \mathrm{I}_n$ and $\det(A) = 1$; the set of these matrices is denoted by $\mathsf{SO}(n)$

Positive definite: if all the eigenvalues of A are positive

Positive semidefinite: if all the eigenvalues of A are nonnegative

B.2 Decompositions

Singular value decomposition Every matrix $A \in M_n(\mathbb{C})$ can be written as the product $A = V\Sigma W^*$, with V and W unitary matrices, and Σ a nonnegative diagonal matrix. The values $\sigma_i = \Sigma_{ii}$ are the *singular values* of A.

Spectral decomposition of normal matrices Any normal matrix A can be decomposed as a product $A = U\Lambda U^*$, with U a unitary matrix, and Λ a diagonal matrix. The diagonal entries Λ_{ii} are the *eigenvalues* λ_i of A and the set $\{\Lambda_{11}, \Lambda_{22}, \ldots\}$ is the *spectrum* of A.

Appendix C

Norms and Distances

C.1 Norms and Distances on Vectors and Matrices

absolute value $|x|$: For a complex value $x \in \mathbb{C}$, its absolute value, or norm, is defined by $|x| = \sqrt{xx^*}$.

Sum norm $||x||_1$: For a complex valued vector $x \in \mathbb{C}^n$, the sum norm is defined by $\|x\|_1 = \sum_i |x_i|$. This norm is also called the ℓ_1, or *Manhattan norm.* For bitvectors $x \in \{0,1\}^n$ the sum norm corresponds with the *Hamming weight* of a bit string: $||x||_1 =$ “number of ones in x”.

Euclidean, or ℓ_2, vector norm $||x||_2$: For a complex valued vector $x \in \mathbb{C}^n$, its norm is defined by $\|x\|_2 = \sqrt{\sum_i x_i x_i^*}$.

Max, or ℓ_∞, norm $||x||_\infty$: For a complex valued vector $x \in \mathbb{C}^n$, the max norm is defined by $\|x\|_\infty = \max_i |x_i|$.

Fidelity: The *fidelity* between two mixed states ρ and σ is defined by

$$\mathrm{F}(\rho,\sigma) \;=\; \mathrm{tr}\left(\sqrt{\sqrt{\rho}\cdot\sigma\cdot\sqrt{\rho}}\right),$$

although the reader should be warned that some authors use the square of this value.

Euclidean matrix norm $||A||_2$: For a complex valued matrix $A \in M_n(\mathbb{C})$, the Euclidean norm is defined by

$$\|A\|_2 \;=\; \sqrt{\sum_{i,j} A_{ij}A_{ij}^*} \;=\; \sqrt{\mathrm{tr}(A\cdot A^*)}.$$

Alternative names are: ℓ_2, *Frobenius, Hilbert-Schmidt,* or *Schur norm.*

We call this norm *unitarily invariant* because $\|U\cdot A\cdot V\|_2 = \|A\|_2$ for unitary $U, V\in \mathsf{U}(n)$. From this invariance it follows, using the SV decomposition, that we have

$$\|A\|_2 \;=\; \sqrt{\sum_i \sigma_i^2},$$

with σ_i the singular values of A, and hence for normal matrices

$$\|A\|_2 \;=\; \sqrt{\sum_i |\lambda_i|^2},$$

where λ_i are the eigenvalues of A.

Trace norm $\|A\|_{\mathrm{tr}}$: For a matrix $A\in M_n(\mathbb{C})$, the trace norm is defined by

$$\|A\|_{\mathrm{tr}} \;=\; \mathrm{tr}\left(\sqrt{A\cdot A^*}\right) \;=\; \sum_i \sigma_i,$$

with σ_i the singular values of A. From this definition it follows that for *normal* matrices the trace norm equals

$$\|A\|_{\mathrm{tr}} \;=\; \sum_i |\lambda_i|,$$

where $\lambda_1, \lambda_2, \ldots$ are the eigenvalues of A.

In the case of positive, semidefinite matrices we thus have $\|A\|_{\mathrm{tr}} = \mathrm{tr}(A)$, hence the name of this norm. (As a consequence, all proper density matrices obey $\|\rho\|_{\mathrm{tr}} = 1$.)

The usefulness of this norm lies in the distance $\|\rho-\sigma\|_{\mathrm{tr}}$ it defines between two density matrices ρ and σ. For any measurement setting $\mathcal{P} = \{P_i\}$ (with $\sum_i P_i^* P_i = I$), the *total variation distance* between ρ and σ is bounded from above by

$$\|\rho-\sigma\|_{\mathrm{tr}} \;\geq\; \sum_{P_i\in\mathcal{P}} |\mathrm{Prob}(\text{“}\rho = P_i\text{”}) - \mathrm{Prob}(\text{“}\sigma = P_i\text{”})|,$$

with $\mathrm{Prob}(\text{“}\rho = P_i\text{”}) = (\mathrm{F}(P_i,\rho))^2$. If we choose the projectors P_i of $\mathcal{P}$ to be the eigenvectors of $\rho-\sigma$, then we obtain the above bound, hence

$$\|\rho-\sigma\|_{\mathrm{tr}} \;=\; \max_{\mathcal{P}}\left(\sum_{P_i\in\mathcal{P}} |\mathrm{Prob}(\text{“}\rho = P_i\text{”}) - \mathrm{Prob}(\text{“}\sigma = P_i\text{”})|\right).$$

Both the Euclidean and the trace norm are *matrix norms* because they obey the following properties (see Chapter 5 in [54] for much more on this topic):

1. nonnegative: $||A|| \geq 0$
2. positive: $||A|| = 0$ if and only if $A = 0$
3. homogeneous: $||\alpha A|| = |\alpha| \cdot ||A||$ for all $\alpha \in \mathbb{C}$
4. triangle inequality: $||A + B|| \leq ||A|| + ||B||$
5. submultiplicative: $||AB|| \leq ||A|| \cdot ||B||$.

In addition, for the tensor product between two matrices, we also have the equality

- $||A \otimes B|| = ||A|| \cdot ||B||$.

A very useful relation between the trace and the Euclidean norm is easily shown by the inequalities $\frac{1}{\sqrt{n}}\sum_i \sigma_i \;\leq\; \sqrt{\sum_i \sigma_i^2} \;\leq\; \sum_i \sigma_i$ for any n nonnegative values $\sigma_1, \ldots, \sigma_n$. If we take these σ_i to be the singular values of A, we see that

$$||A||_2 \;\leq\; ||A||_{\mathrm{tr}} \;\leq\; \sqrt{n} \cdot ||A||_2, \tag{C.1}$$

for all $A \in M_n(\mathbb{C})$.

C.2 Norms on Superoperators

Trace induced superoperator norm: For a superoperator $\boldsymbol{E} : M_n(\mathbb{C}) \to M_m(\mathbb{C})$ we can use the trace norm to define

$$|||\boldsymbol{E}|||_{\mathrm{tr}} \;=\; \max_{A \neq 0} \frac{||\boldsymbol{E}(A)||_{\mathrm{tr}}}{||A||_{\mathrm{tr}}}.$$

If $\boldsymbol{E}$ is a positive, trace preserving mapping, then $|||\boldsymbol{E}|||_{\mathrm{tr}} = 1$. A drawback of this norm is that it can increase if we tensor $\boldsymbol{E}$ with the identity operator. Take for example the one qubit transpose, with $\boldsymbol{T}(A) = A^T$, which has $|||\boldsymbol{T}|||_{\mathrm{tr}} = 1$, but also $|||\boldsymbol{T} \otimes \mathbf{I}_2|||_{\mathrm{tr}} = 2$.

Diamond superoperator norm: Let $\boldsymbol{E} : M_n(\mathbb{C}) \to M_m(\mathbb{C})$ be a linear superoperator, the *diamond norm* can then be defined by

$$|||\boldsymbol{E}|||_{\diamond} \;=\; |||\boldsymbol{E} \otimes \mathbf{I}_n|||_{\mathrm{tr}}.$$

The reader is referred to the original articles [3, 62] by Alexei Kitaev *et al.* for more details. One of the appealing properties of this norm is its robustness: $|||\boldsymbol{E} \otimes \mathbf{I}|||_{\diamond} = |||\boldsymbol{E}|||_{\diamond}$.

If $\boldsymbol{E}$ is a completely positive, trace preserving transformation, then $|||\boldsymbol{E}|||_{\diamond} = 1$.

Euclidean induced superoperator norm: We define a norm for a superoperator $\boldsymbol{E} : M_n(\mathbb{C}) \to M_m(\mathbb{C})$, by the maximization of the Euclidean norm for matrices:

$$\|\boldsymbol{E}\|_2 \;=\; \max_{A\neq 0} \frac{||\boldsymbol{E}(A)||_2}{||A||_2}.$$

It is straightforward to show that this norm is, like the diamond norm, robust: $\|\boldsymbol{E} \otimes \mathbf{I}\|_2 = \|\boldsymbol{E}\|_2$, for the identity operator $\mathbf{I}$.

By the bounds of Equation C.1, we have for any superoperator $\boldsymbol{E} : M_n(\mathbb{C}) \to M_m(\mathbb{C})$

$$\|\boldsymbol{E}\|_2 \;\leq\; \sqrt{n}\|\boldsymbol{E}\|_{\mathrm{tr}} \text{ and } \|\boldsymbol{E}\|_{\mathrm{tr}} \;\leq\; \sqrt{m}\|\boldsymbol{E}\|_2.$$

Because $\|\boldsymbol{E} \otimes \mathbf{I}\|_2 = \|\boldsymbol{E}\|_2$, we thus obtain an upper bound on the diamond norm in terms of the trace norm:

$$\|\boldsymbol{E}\|_\diamond \;=\; \|\boldsymbol{E} \otimes \mathbf{I}_n\|_{\mathrm{tr}} \;\leq\; \sqrt{nm}\|\boldsymbol{E} \otimes \mathbf{I}_n\|_2 \;=\; \sqrt{nm}\|\boldsymbol{E}\|_2 \;\leq\; n\sqrt{m}\|\boldsymbol{E}\|_{\mathrm{tr}},$$

in combination with the trivial lower bound $\|\boldsymbol{E}\|_{\mathrm{tr}} \leq \|\boldsymbol{E}\|_\diamond$.

Appendix D

Approximate Interrogation

In this appendix we calculate the expected number of correct bits for the 'approximate interrogation' procedure of Section 3.6.

We can assume without loss of generality that the queried string is the all zeros string $z_1 \cdots z_n = 0^n$, such that the A_k transformation is the identity operator, and the Hamming weight $||y||$ of a measured outcome $y_1 \cdots y_n$ equals the number of incorrect bits. We thus set out to prove the following lemma.

Lemma 31 *With the state*

$$|\Psi\rangle \;=\; \sum_{x\in\{0,1\}^n} \alpha_{||x||}|x\rangle,$$

the n-fold Hadamard transform of Ψ will have an expected Hamming weight of

$$\begin{aligned}\mathbf{E}[\text{\#}\mathit{ones}(\mathrm{H}^{\otimes n}|\Psi_k\rangle)] &= \sum_{t=0}^{n} t\cdot\binom{n}{t}|\langle 1^t0^{n-t}|\mathrm{H}^{\otimes n}|\Psi_k\rangle|^2\\ &= \frac{n}{2}-\sum_{j=0}^{k-1}\mathrm{Re}(\alpha_j\alpha_{j+1}^*)\sqrt{j+1}\sqrt{n-j}.\end{aligned}$$

Note that we are expressing here the number of incorrect bits, from which the equality

$$\mathbf{E}[\text{\#zeros}(\mathrm{H}^{\otimes n}|\Psi_k\rangle)] \;=\; \frac{n}{2}+\sum_{j=0}^{k-1}\mathrm{Re}(\alpha_j\alpha_{j+1}^*)\sqrt{j+1}\sqrt{n-j}.$$

follows directly.

The proof of this lemma requires some knowledge about the following family of orthogonal polynomials.

Definition 28 (Krawtchouk Polynomials [65]) *For $r, n \in \mathbb{N}$, the Krawtchouk Polynomial $K_r(\cdot, n) : \mathbb{N} \to \mathbb{Z}$ is defined by*

$$K_r(t;n) \quad := \quad \sum_{j=0}^{r} (-1)^j \binom{t}{j}\binom{n-t}{r-j}.$$

From Chapter 5 in [70] we copy the following property of Krawtchouk polynomials:

$$\sum_{t=0}^{n} \binom{n}{t} K_r(t;n) K_s(t;n) \quad = \quad \begin{cases} 2^n \binom{n}{r} & \text{if } r = s, \\ 0 & \text{if } s \neq r. \end{cases} \tag{D.1}$$

Another important result is the following three-term recurrence relation that K satisfies:

$$(n-2t)K_k(t,n) \quad = \quad (k+1)K_{k+1}(t,n) + (n-k+1)K_{k-1}(t,n). \tag{D.2}$$

From this, the fact that $K_k^2(t,n) = K_k^2(n-t,n)$ for all k, n, t, follows easily.

We now proceed with the proof of the lemma.

Proof: By rewriting the state in lemma according to

$$\begin{aligned} \mathrm{H}^{\otimes n}|\Psi_k\rangle &= \sum_{j=0}^{k} \frac{\alpha_j}{\sqrt{\binom{n}{j}}} \sum_{x\in\{0,1\}^n}^{\|x\|_1=j} \mathrm{H}^{\otimes n}|x\rangle \\ &= \frac{1}{\sqrt{2^n}} \sum_{y\in\{0,1\}^n} \sum_{j=0}^{k} \frac{\alpha_j}{\sqrt{\binom{n}{j}}} \sum_{x\in\{0,1\}^n}^{\|x\|_1=j} (-1)^{(y,x)}|y\rangle, \end{aligned}$$

we obtain the following expression for the expected Hamming weight:

$$\begin{aligned} D(\alpha,n,k) &:= \mathbf{E}[\#\text{ones}(\mathrm{H}^{\otimes n}|\Psi_k\rangle)] \\ &= \sum_{t=0}^{n} t\cdot\binom{n}{t} |\langle 1^t 0^{n-t}|\mathrm{H}^{\otimes n}|\Psi_k\rangle|^2 \\ &= \frac{1}{2^n}\sum_{t=0}^{n} t\cdot\binom{n}{t} \left| \sum_{j=0}^{k} \frac{\alpha_j}{\sqrt{\binom{n}{j}}} \sum_{x\in\{0,1\}^n}^{||x||_1=j} (-1)^{(1^t0^{n-t},x)} \right|^2 \\ &= \frac{1}{2^n}\sum_{t=0}^{n} t\cdot\binom{n}{t} \left| \sum_{j=0}^{k} \frac{\alpha_j}{\sqrt{\binom{n}{j}}} \sum_{i=0}^{j} (-1)^i \binom{t}{i}\binom{n-t}{j-i} \right|^2 \\ &= \frac{1}{2^n}\sum_{t=0}^{n} t\cdot\binom{n}{t} \left| \sum_{j=0}^{k} \frac{\alpha_j K_j(t,n)}{\sqrt{\binom{n}{j}}} \right|^2. \end{aligned}$$

It is easy to see that D is a second degree, multivariate polynomial in the variables α_i:

$$D(\alpha,n,k) \;=\; \sum_{i,j=0}^{k} \beta_{ij}\alpha_i\alpha_j^*$$

with $\beta_{ij} \in \mathbb{R}$ and $\beta_{ij} = \beta_{ji}$ for all i, j, such that $D(\alpha,n,k) \in \mathbb{R}$ for all $\alpha \in \mathbb{C}^n$. Our task is thus to determine these β coefficients.

We start by considering the diagonal elements β_{jj}, with:

$$\beta_{jj} \;=\; \frac{1}{2^n}\sum_{t=0}^{n} t\binom{n}{t}\frac{K_j^2(t,n)}{\binom{n}{j}}.$$

Now, because of the symmetry $\binom{n}{t}K_j^2(t,n) = \binom{n}{n-t}K_j^2(n-t,n)$, we can rewrite this summation as (using Equation D.1 for the last line)

$$\begin{aligned}\beta_{jj} &= \frac{1}{2^n\binom{n}{j}}\sum_{t=0}^{n}\frac{n}{2}\binom{n}{t}K_j^2(t,n)\\ &= \frac{n}{2}.\end{aligned}$$

Next, we look at the off-diagonal terms:

$$\beta_{ij} \;=\; \frac{1}{2^n\sqrt{\binom{n}{i}\binom{n}{j}}}\sum_{t=0}^{n} t\binom{n}{t}K_i(t,n)K_j(t,n).$$

By rewriting the t in front of the $\binom{n}{t}$ binomial as $t = \frac{n}{2} - \frac{1}{2}(n-2t)$, and using $i \neq j$ with Equations D.1 and D.2, we get

$$\begin{aligned}\beta_{ij} &= \frac{-1}{2^{n+1}\sqrt{\binom{n}{i}\binom{n}{j}}}\sum_{t=0}^{n}(n-2t)\binom{n}{t}K_i(t,n)K_j(t,n)\\ &= \frac{-1}{2^{n+1}\sqrt{\binom{n}{i}\binom{n}{j}}}\sum_{t=0}^{n}\binom{n}{t}[(i+1)K_{i+1}(t,n)+(n-i+1)K_{i-1}(t,n)]K_j(t,n).\end{aligned}$$

The orthogonality property of K shows that the above term is zero if $i - j \neq \pm 1$; otherwise, we have

$$\beta_{ij} \;=\; \begin{cases} -\frac{1}{2}\sqrt{(n-i)(i+1)} & \text{if } i+1=j,\\ -\frac{1}{2}\sqrt{(i)(n-i+1)} & \text{if } i-1=j.\end{cases}$$

This concludes our proof that indeed (using the normalization restriction $\sum_i |\alpha_i^2| = 1$)

$$\begin{aligned}D(\alpha,n,k) &= \frac{n}{2}\sum_{i=0}^{k}|\alpha_i^2| - \frac{1}{2}\sum_{i=0}^{k-1}\sqrt{(n-i)(i+1)}(\alpha_i\alpha_{i+1}^* + \alpha_i\alpha_{i+1}^*)\\ &= \frac{n}{2} - \sum_{i=0}^{k-1}\sqrt{(n-i)(i+1)}\mathrm{Re}(\alpha_i\alpha_{i+1}^*).\end{aligned}$$

□

Bibliography

[1] Leonard M. Adleman, Jonathan Demarrais, and Ming-Deh A. Huang. Quantum computability. *SIAM Journal on Computing*, 26(5):1524–1540, October 1997.

[2] Manindra Agrawal and V. Arvind. Quasi-linear truth-table reductions to p-selective sets. *Theoretical Computer Science*, 158(1–2):361–370, May 1996.

[3] Dorit Aharonov, Alexei Kitaev, and Noam Nisan. Quantum circuits with mixed states. In *Proceedings of the 30th Annual ACM Symposium on Theory of Computing*, pages 20–30, 1998.

[4] Amihood Amir, Richard Beigel, and William I. Gasarch. Some connections between bounded query classes and nonuniform complexity. In *Proceedings of the 5th Annual Conference on Structure in Complexity Theory*, pages 232–243, 1990.

[5] Amihood Amir and William I. Gasarch. Polynomial terse sets. *Information and Computation*, 77(1):37–56, April 1988.

[6] José L. Balcázar, Josep Díaz, and Joaquim Gabarró. *Structural Complexity*, volume 1. Springer-Verlag, 1988.

[7] José L. Balcázar, Josep Díaz, and Joaquim Gabarró. *Structural Complexity*, volume 2. Springer-Verlag, 1990.

[8] Adriano Barenco, André Berthiaume, David Deutsch, Artur Ekert, Richard Jozsa, and Chiara Macchiavello. Stabilisation of quantum computations by symmetrisation. *SIAM Journal on Computing*, 26(5):1541–1557, 1997.

[9] H. Barnum, C.M. Caves, C.A. Fuchs, R. Jozsa, and B. Schumacher. Noncommuting mixed states cannot be broadcast. *Physical Review Letters*, 76(15):2821–2828, 1996. arXiv:quant-ph/9511010.

[10] Robert Beals, Harry Buhrman, Richard Cleve, Michele Mosca, and Ronald de Wolf. Quantum lower bounds by polynomials. In *Proceedings of the 39th Annual Symposium on Foundations of Computer Science*, pages 352–361, Los Alamitos, California, November 1998. IEEE Computer Society Press. arXiv:quant-ph/9802049.

[11] Richard Beigel. *Query-limited Reducibilities*. Ph.d. dissertation, Department of Computer Science, Stanford University, 1987. Available on the web via: `http://www.eecs.lehigh.edu/~beigel/papers/`.

[12] Richard Beigel. NP-hard sets are P-superterse unless R = NP. Technical report, Johns Hopkins University, 1988. technical report no. 88-4.

[13] Richard Beigel. Bounded queries to SAT and the Boolean hierarchy. *Theoretical Computer Science*, 84(2):199–223, 1991. Available on the web via: `http://www.eecs.lehigh.edu/~beigel/papers/`.

[14] Richard Beigel, R. Chang, and M. Ogiwara. A relationship between difference hierarchies and relativized polynomial hierarchies. *Mathematical Systems Theory*, 26(3):293–310, 1993.

[15] Richard Beigel and William I. Gasarch. On the complexity of finding the chromatic number of a recursive graph I: The bounded case. *Annals of Pure and Applied Logic*, 45(1):1–38, 1989.

[16] Richard Beigel, William I. Gasarch, John Gill, and Jr. James C. Owings. Terse, superterse and verbose sets. *Information and Computation*, 103:68–85, 1993.

[17] Richard Beigel, Martin Kummer, and Frank Stephan. Approximable sets. *Information and Computation*, 120(2):304–314, 1995.

[18] Riccardo Benedetti and Jean-Jacques Risler. *Real algebraic and semi-algebraic sets*. Hermann, 1990.

[19] Charles H. Bennett, Ethan Bernstein, Gilles Brassard, and Umesh Vazirani. Strengths and weaknesses of quantum computing. *SIAM Journal on Computing*, 26(5):1510–1523, October 1997. arXiv:quant-ph/9701001.

[20] Charles H. Bennett and Gilles Brassard. Quantum cryptography: Public key distribution and coin tossing. In *Proceedings of the IEEE International Conference on Computers, Systems, and Signal Processing*, pages 175–179, 1984.

[21] Charles H. Bennett and Stephen Wiesner. Communication via one- and two-particle operators on Einstein-Podolsky-Rosen states. *Physical Review Letters*, 69:2881–2884, 1992.

[22] Ethan Bernstein and Umesh Vazirani. Quantum complexity theory. *SIAM Journal on Computing*, 26(5):1411–1473, October 1997.

[23] André Berthiaume and Gilles Brassard. Oracle quantum computing. *Journal of Modern Optics*, 41(12):2521–2535, 1994.

[24] Manuel Blum, Michael Luby, and Ronitt Rubinfeld. Self-testing/correcting with applications to numerical problems. *Journal of Computer and System Sciences*, 47(3):549–595, 1990.

[25] Michel Boyer, Gilles Brassard, Peter Høyer, and Alain Tapp. Tight bounds on quantum searching. *Fortschritte der Physik*, 46(4–5):493–505, 1998. arXiv:quant-ph/9605034.

[26] P. Oscar Boykin, Tal Mor, Matthew Pulver, Vwani Roychowdhury, and Farrokh Vatan. On universal and fault-tolerant quantum computing: A novel basis and a new constructive proof of universality for Shor's basis. In *Proceedings of the 40th Annual Symposium on Foundations of Computer Science*, pages 486–494, 1999.

[27] Harry Buhrman and Lance Fortnow. Two queries. In *Proceedings of the 13th IEEE Conference on Computational Complexity*, pages 13–19, New York, 1998. IEEE Computer Society Press. Available on the web via `http://www.cs.uchicago.edu/~fortnow/papers/`.

[28] Harry Buhrman and Wim van Dam. Bounded quantum query complexity. In *Proceedings of the 14th Annual IEEE Conference on Computational Comlexity*, pages 149–156, 1999. arXiv:quant-ph/9903035.

[29] Samuel R. Buss and Louise Hay. On truth-table reducibility to SAT. *Information and Computation*, 91(1):86–102, March 1991.

[30] Gregory Chaitin. On the length of programs for computing finite binary sequences. *Journal of the ACM*, 13(4):547–569, 1966.

[31] Richard Cleve, Artur Ekert, Chiara Macchiavello, and Michele Mosca. Quantum algorithms revisited. *Proceedings of the Royal Society of London A*, 454:339–354, 1998. arXiv:quant-ph/9708016.

[32] Henri Cohen. *A Course in Computational Algebraic Number Theory*, volume 138 of *Graduate Texts in Mathematics*. Springer-Verlag, 1993.

[33] Charles J. Colbourn and Jeffrey H. Dinitz, editors. *The CRC Handbook of Combinatorial Designs*. Series on Discrete Mathematics and Applications. CRC Press, 1996.

[34] Stephen A. Cook. The complexity of theorem-proving procedures. In *Proceedings of the 3rd ACM Symposium Theory of Computing*, pages 151–158, Shaker Heights, Ohio, 1971.

[35] Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. Wiley Series in Telecommunications. John Wiley & Sons, Inc., 1991.

[36] Wim van Dam. Two classical queries versus one quantum query. Technical Report 9806090, quant-ph report, 1998.

[37] Wim van Dam and Sean Hallgren. Efficient quantum algorithms for shifted quadratic character problems. Technical report. arXiv:quant-ph/0011067.

[38] David Deutsch. Quantum theory, the Church-Turing principle and the universal quantum computer. *Proceedings of the Royal Society of London A*, 400:97–117, 1985.

[39] David Deutsch and Richard Jozsa. Rapid solution of problems by quantum computation. *Proceedings of the Royal Society of London A*, 439:553–558, 1992.

[40] Edward Farhi, Jeffrey Goldstone, Sam Gutmann, and Michael Sipser. How many functions can be distinguished with k quantum queries? Technical report. arXiv:quant-ph/9901012.

[41] Edward Farhi, Jeffrey Goldstone, Sam Gutmann, and Michael Sipser. Invariant quantum algorithms for insertion into an ordered list. Technical report. arXiv:quant-ph/9901059.

[42] Edward Farhi, Jeffrey Goldstone, Sam Gutmann, and Michael Sipser. A limit on the speed of quantum computation in determining parity. *Physical Review Letters*, 81:5442–5444, 1998. arXiv:quant-ph/9802045.

[43] Lance Fortnow and John Rogers. Complexity limitations on quantum computation. In *Proceedings of the 13th IEEE Conference on Computational Complexity*, pages 202–209, 1998. arXiv:cc.CC/9811023.

[44] Christopher A. Fuchs and Jeroen van de Graaf. Cryptographic distinguishability measures for quantum mechanical states. *IEEE Transactions on Information Theory*, 45(4):1216–1227, 1999.

[45] Peter Gács. Quantum algorithmic entropy. *Journal of Physics A: Mathematical and General*, 34:6859–6880, 2001. arXiv:quant-ph/0011046.

[46] Michael R. Garey and David S. Johnson. *Computers and Intractability: A guide to the theory of NP-completeness*. W.H. Freeman and Company, New York, 1979.

[47] Peter Gemmell, Richard Lipton, Ronitt Rubinfeld, Madhu Sudan, and Avi Wigderson. Self-testing/correcting for polynomials and for approximate functions. In *Proceedings of the 23rd Annual ACM Symposium on Theory of Computing*, pages 32–42, 1991.

[48] Lov K. Grover. A fast quantum mechanical algorithm for database search. In *Proceedings of the 28th Annual ACM Symposium on Theory of Computing*, pages 212–219, Philadelphia, Pennsylvania, May 1996. ACM. arXiv:quant-ph/9605043.

[49] Lov K. Grover. Quantum computers can search arbitrarily large databases by a single query. *Physical Review Letters*, 79(23):4709–4712, December 1997. arXiv:quant-ph/9706005.

[50] Jacques Hadamard. Résolution d'une question relative aux déterminants. *Bulletin des Sciences Mathématiques*, 17(2):240–246, 1893.

[51] Sam Hedayat, Neil Sloane, and John Stufken. *Orthogonal Arrays: Theory and Applications*. Springer Verlag, New York, 1999.

[52] Edith Hemaspaandra, Lane A. Hemaspaandra, and Harald Hempel. A downward translation in the polynomial hierarchy. In *14th Annual Symposium on Theoretical Aspects of Computer Science*, volume 1200 of *Lecture Notes in Computer Science*, pages 319–328, Lübeck, Germany, 1997. Springer.

[53] Alexander S. Holevo. Bounds for the quantity of information transmitted by a quantum communication channel. *Problemy Peredachi Informatsii*, 9(3):3–11, 1973. English Translation in *Problems in Information Transmission*, 9:177–183, 1973.

[54] Roger A. Horn and Charles R. Johnson. *Matrix Analysis*. Cambridge University Press, 1985.

[55] Michał Horodecki. Limits for compression of quantum information carried by ensembles of mixed states. *Physical Review A*, 57(5):3364–3369, May 1998.

[56] Peter Høyer, Jan Neerbek, and Yaoyun Shi. Quantum complexities of ordered searching, sorting, and element distinctness. In *Proceedings of 28th International Colloquium on Automata, Languages, and Programming*, volume 2076 of *Lecture Notes in Computer Science*, pages 346–357. Springer, 2001.

[57] Kenneth Ireland and Michael Rosen. *A Classical Introduction to Modern Number Theory*, volume 84 of *Graduate Texts in Mathematics*. Springer, second edition, 1990.

[58] David S. Johnson. A catalogue of complexity classes. In J. van Leeuwen, editor, *Handbook of Theoretical Computer Science*, volume A, pages 67–161. Elsevier, Amsterdam, 1990.

[59] Richard Jozsa. Characterizing classes of functions computable by quantum parallelism. *Proceedings of the Royal Society of London A*, 435:563–574, 1991.

[60] Jim Kadin. The polynomial time hierarchy collapses if the Boolean hierarchy collapses. *SIAM Journal on Computing*, 17(6):1263–1282, 1988.

[61] Hiroshi Kimura. Hadamard matrices and dihedral groups. *Designs, Codes, and Cryptography*, 9(1):71–77, 1996.

[62] Alexei Kitaev. Quantum computations: Algorithms and error correction. *Russian Mathematical Surveys*, 52(6):1191–1249, 1997. English translation from *Uspekhi Matematicheskikh Nauk*, Volume 52, Number 6, pp. 53–112.

[63] Donald E. Knuth. *The Art of Computer Programming: Seminumerical Algorithms*, volume 2. Addison-Wesley, Reading, Massachusetts, third edition, 1998.

[64] Andrei K. Kolmogorov. Three approaches to the quantitative definition of information. *Problems of Information Transmission*, 1:1–7, 1965.

[65] M. Krawtchouk. Sur une généralisation des polynomes d'Hermite. *Comptes Rendus*, 189:620–622, 1929.

[66] Leonid A. Levin. Universal search problems. *Problemy Peredaci Informacii*, 9(3):115–116, 1973. In Russian, English translation in *Problems of Information Transmission*, 9, 265–266.

[67] Ming Li and Paul Vitányi. *An Introduction to Kolmogorov Complexity and its Applications*. Springer Verlag, second edition, 1997.

[68] Göran Lindblad. Completely positive maps and entropy inequalities. *Communications in Mathematical Physics*, 40:147–151, 1975.

[69] Richard Lipton. New directions in testing. In *Distributed Computing and Cryptography*, volume 2 of *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, pages 191–202. American Mathematics Society, 1991.

[70] Florence J. MacWilliams and Neil J.A. Sloane. *The Theory of Error-Correcting Codes*, volume 16 of *North-Holland Mathematical Library*. Elsevier Science Publishers, New York, 1977.

[71] Dominic Mayers and Andrew Chi-Chih Yao. Quantum cryptography with imperfect apparatus. In *Proceedings of the 39th Annual Symposium on Foundations of Computer Science*, pages 503–509, 1998.

[72] Ashwin Nayak and Felix Wu. The quantum query complexity of approximating the median and related statistics. In *Proceedings of the Thirty-First Annual ACM Symposium on Theory of Computing*, pages 384–393. ACM Press, 1999. arXiv:quant-ph/9804066.

[73] Michael A. Nielsen. *Quantum Information Theory*. Ph.D. dissertation, University of New Mexico, 1998.

[74] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information*. Cambridge University Press, 2000.

[75] Mitsunori Ogihara. Polynomial-time membership comparable sets. *SIAM Journal on Computing*, 24(5):1068–1081, 1995.

[76] Raymond E.A.C. Paley. On orthogonal matrices. *Journal of Mathematics and Physics*, 12:311–320, 1933.

[77] Asher Peres. *Quantum Theory: Concepts and Methods*, volume 72 of *Fundamental Theories of Physics*. Kluwer Academic Publishers, Dordrecht, The Netherlands, 1995.

[78] John Preskill. Quantum computing. URL: `http://www.theory.caltech.edu/people/preskill/ph229/`, 1998. Course notes.

[79] Ronitt Rubinfeld. *A mathematical theory of self-checking, self-testing and self-correcting programs*. Ph.D. dissertation, University of California, Berkeley, 1990.

[80] Ronitt Rubinfeld. On the robustness of functional equations. In *Proceedings of the 35th Annual Symposium on Foundations of Computer Science*, pages 288–299, 1994.

[81] Ronitt Rubinfeld and Madhu Sudan. Robust characterizations of polynomials with applications to program testing. *SIAM Journal on Computing*, 25(2):252–271, 1996.

[82] Mary Beth Ruskai. Beyond strong subadditivity? improved bounds on the contraction of generalized relative entropy. *Reviews in Mathematical Physics*, 6(5a):1147–1161, 1994.

[83] Benjamin Schumacher. Quantum coding. *Physical Review A*, 51(4):2738–2747, 1995.

[84] Benjamin Schumacher. Sending quantum entanglement through noisy channels. *Physical Review A*, 54(4):2614–2628, October 1996. arXiv:quant-ph/9604023.

[85] Jennifer Seberry. A life's work on Hadamard matrices, statistical designs, bent functions and their application to computer and information security and telecommunications. `http://www.cs.uow.edu.au/people/jennie/lifework.html`.

[86] Jennifer Seberry and Mieko Yamada. Hadamard matrices, sequences, and block designs. In Jeffrey H. Dinitz and Douglas R. Stinson, editors, *Contemporary Design Theory: A Collection of Surveys*, Wiley-Interscience series in discrete mathematics and optimization, chapter 11, pages 431–560. John Wiley & Sons, 1992.

[87] Alan L. Selman. A taxonomy of complexity classes of functions. *Journal of Computer and System Sciences*, 48(2):357–381, April 1994.

[88] Adi Shamir. Factoring numbers in $O(\log n)$ arithmetic steps. *Information Processing Letters*, 8(1):28–31, January 1979.

[89] Claude E. Shannon and Warren Weaver. *The mathematical theory of communication*. University of Illinois Press, 1949.

[90] Koichi Shinoda and Mieko Yamada. A family of Hadamard matrices of dihedral group type. *Discrete Applied Mathematics*, 102(1–2):141–150, May 2000.

[91] Peter W. Shor. Algorithms for quantum computation: Discrete logarithms and factoring. *SIAM Journal on Computing*, 26(5):1484–1509, 1997. arXiv:quant-ph/9508027.

[92] Daniel R. Simon. On the power of quantum computation. *SIAM Journal on Computing*, 26(5):1474–1483, 1997.

[93] Neil Sloane. A library of Hadamard matrices. `http://www.research.att.com/~njas/hadamard/index.html`.

[94] Ray Solomonoff. A preliminary report on a general theory of inductive inference. Technical Report ZTB-138, Zator Company, Cambridge, Mass., 1960.

[95] Karl Svozil. Quantum algorithmic information theory. *Journal of Universal Computer Science*, 2:311–346, 1996.

[96] James Joseph Sylvester. Thoughts on inverse orthogonal matrices, simultaneous sign-successions, and tessellated pavements in two or more colours, with applications to Newton's rule, ornamental tile-work, and the theory of numbers. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 34(232):461–475, 1867.

[97] Barbara Terhal and John Smolin. Single quantum querying of a database. *Physical Review A*, 58(3):1822–1826, September 1998. arXiv:quant-ph/9705041.

[98] Seinosuke Toda. On polynomial-time truth-table reducibility of intractable sets to p-selective sets. *Mathematical Systems Theory*, 24:68–82, 1991.

[99] Alan Turing. On computable numbers, with an application to the Entscheidungsproblem. *Proceedings of the London Mathematical Society, Series 2*, 42:230–265, 1937.

[100] Richard J. Turyn. Complex Hadamard matrices. In *Combinatorial structures and their applications: proceedings of the Calgary international conference on combinatorial structures and their applications*, pages 435–437, Calgary, Alberta, Canada, June 1996. University of Calgary, Gordon and Breach.

[101] Armin Uhlmann. Relative entropy and the Wigner-Yanase-Dyson-Lieb concavity in an interpolation theory. *Reviews in Mathematical Physics*, 54:21–32, 1977.

[102] Paul Vitányi. Three approaches to the quantitative definition of information in an individual pure quantum state. In *Proceedings of the 15th Annual Conference on Computational Complexity*, 2000.

[103] Paul Vitányi. Quantum kolmogorov complexity using classical descriptions. *IEEE Transactions on Information Theory*, 47(6):2464–2479, 2001.

[104] Klaus W. Wagner. Bounded query classes. *SIAM Journal on Computing*, 19(5):833–846, October 1990.

[105] Alfred Wehrl. General properties of entropy. *Reviews of Modern Physics*, 50(2):221–260, 1978.

[106] Reinhard F. Werner. Optimal cloning of pure states. *Physical Review A*, 58:1827–1832, 1998.

[107] William K. Wootters and Wojceich H. Zurek. A single quantum cannot be cloned. *Nature*, 299:802–803, October 1982.

Samenvatting

“Over Quantumberekeningen”

Een quantumcomputer is een computers wiens gedrag cruciaal wordt bepaald door de wetten van de quantummechanica. Dit is een ander soort machine dan de traditionele computer die we kennen uit het dagelijkse leven aangezien deze functioneert volgens de regels van de klassieke mechanica. Hoewel men er nog niet in is geslaagd om een werkende quantumcomputer van behoorlijke grootte te bouwen, is het wel mogelijk om de eigenschappen hiervan te onderzoeken. Dit theoretisch werk dat zich op het grensgebied bevindt van de quantummechanica en de theoretische informatica is het onderwerp van dit proefschrift.

In de hoofdstukken 1 en 2 geef ik een samenvatting van de aspecten van de quantummechanische theorie die essentieel zijn om te begrijpen wat quantuminformatie en quantumcomputers zijn. De twee belangrijkste ingrediënten hierbij zijn het zogenaamde *superpositie principe* en het *interferentie fenomeen.*

De toestand van een quantummechanisch systeem is in het algemeen een lineaire combinatie van de eigentoestanden van dit systeem. Dit betekent dat een quantumbit niet alleen “nul” of “één” kan zijn, maar ook een mengeling (superpositie) van deze twee toestanden. Wiskundig wordt dit het best beschreven middels een 2-dimensionale, vector (α,β) van lengte 1, waarbij de complexe waarde α de amplitude is van het “nul”-gedeelte van de quantumbit, en β de complexe amplitude van het “één”-gedeelte van de quantumbit. Als we een quantumbit (α,β) bekijken dan zullen we de waarde “nul” waarnemen met waarschijnlijkheid $|\alpha^2|$, en de waarde “één” met waarschijnlijkheid $|\beta^2|$ (vandaar ook dat de lengte van de vector 1 moet zijn: $|\alpha^2|+|\beta^2|=1$). Zodoende corresponderen $(1,0)$ en $(0,1)$ met een klassieke waardes “nul” en “één”, terwijl $(\frac{3}{5},\frac{4}{5})$ een ‘$(36\%,64\%)$-combinatie’ is van beide. Als we twee quantumbits willen beschrijven dan hebben we een vier-dimensionale vector $(\alpha_{00},\alpha_{01},\alpha_{10},\alpha_{11})$ nodig, waarbij α_{00} de amplitude is voor de waarde “nul, nul”, α_{01} voor de waarde “nul, één”, enzovoorts. In het algemeen beschrijft men dus de toestand van n quantumbits met een een 2^n-dimensionale vector.

De tijdsevolutie van een quantummechanisch systeem kan beschreven worden als een lineaire transformatie van bovengenoemde vectoren. De enige eis waaraan deze functies moeten voldoen is dat ze de lengte van de vectoren niet veranderen. Wiskundig gesproken zijn dit de 'unitaire transformaties'. Voor een n-bits quantumsysteem van dimensie 2^n hebben we dus een unitaire matrix met grootte $2^n \times 2^n$ nodig om deze tijdsevolutie te kunnen beschrijven.

Om informatie (bits) op een quantummechanische manier te bewerken hebben we een quantumcomputer nodig die de gewenste unitaire transformaties kan implementeren. Dit beschrijven we als volgt. In de theoretische informatica abstraheert men computers vaak tot netwerk van elementaire poorten. Voor klassieke computers zijn deze basispoorten de AND, de OR, en de NOT operatie; middels welke we elke andere transformatie kunnen opbouwen. Voor quantumcomputers hebben we een soortgelijke situatie, alleen zijn de basispoorten natuurlijk anders (deze moeten natuurlijk quantummechanisch zijn). De complexiteit van een berekening kan men nu uitdrukken als de minimale grootte van het netwerk (dat is: het minimale aantal van basispoorten), dat nodig is om deze berekening uit te voeren. De klassieke complexiteit is zodoende het minimale aantal klassieke poorten dat men nodig heeft voor de oplossing van een probleem, terwijl de quantumcomplexiteit het minimale aantal quantumpoorten als maatstaf heeft. Uit onderzoek is gebleken dat voor sommige berekeningen de quantumcomplexiteit veel kleiner is dan de bijbehorende klassieke complexiteit. Met andere woorden: quantumcomputers zijn soms efficiënter dan traditionele computers.

De hoofdstukken 3, 4 en 5 hebben als onderwerp het quantummechanisch 'ondervragen' van informatie. In hoofdstuk 3 is deze informatie een rij $x_1, \dots, x_n$ van n onbekende bits. Op de vraag "wat is x_i?" krijgt men als antwoord de waarde van de bit x_i. Ik laat zien dat het, middels een superpositie van vragen, mogelijk is om met grote kans alle n bits te weten te komen met slechts $\frac{n}{2} + \sqrt{n}$ quantumvragen. Klassiek is dit onmogelijk en zal men altijd om dit te bereiken alle n vragen "x_1?", … , "x_n?" moeten stellen. Dit 'quantumvoordeel' wordt verder uitgebuit in de volgende twee hoofdstukken.

In hoofdstuk 4 wordt beschreven wat mogelijk is als men quantumvragen kan stellen aan een 'orakel' dat bepaalde, zeer specifieke computationele problemen kan oplossen. Door nu de juiste superpositie van verschillende vragen aan het orakel te stellen kunnen meer algemene 'meta-vragen' beantwoord worden op een manier waarbij we het orakel veel minder hoeven te consulteren dan dat klassiek vereist is. Hoe groot dit verschil tussen de quantummechanische and klassieke vraagcomplexiteit is hangt af van het soort orakel dat men gebruikt en of men de vragen 'interactief' kan stellen.

Hoe kunnen we andere orakel-problemen construeren waarvoor een quantumcomputer veel minder vragen hoeft te stellen dan een klassieke computer? Deze vraag wordt behandeld in hoofdstuk 5. In de wiskunde van de combinatoriek bestudeert men al meer dan een eeuw lang zogenaamde 'Hadamard– en weegmatrices' die zich kenmerken doordat elke rij in deze matrices maximaal verschilt van alle andere rijen. Ik laat zien dat deze constructies zeer geschikt zijn voor het definiëren van problemen die zich lenen voor een quantumoplossing die efficiňter is dan de klassieke oplossing van hetzelfde probleem.

‘Zelftesten’ refereert aan de mogelijkheid van een apparaat om eigenhandig te controleren of het naar behoren werkt. Quantumpoorten (zoals we die willen gebruiken in een quantumcomputer) kunnen zichzelf inderdaad testen, zo bewijzen we in hoofdstuk 6. Dit is goed nieuws aangezien dit resultaat laat zien dat we de paradoxale situatie kunnen vermijden waarin we de bruikbaarheid van een quantumcomputer alleen kunnen verifiëren met behulp van een reeds werkende quantumcomputer.

In hoofdstuk 7, tenslotte, proberen we een definitie te geven voor de ‘quantum-Kolmogorov-complexiteit’ van quantuminformatie. In de klassieke informatietheorie komt de Kolmogorov-complexiteit van een string $x_1,\ldots,x_n$ overeen met de grootte van het kleinste computerprogramma dat $x_1,\ldots,x_n$ als uitvoer heeft. Zo ziet men dat de Kolmogorov-complexiteit van een string van 1 miljoen nullen veel kleiner zal zijn dan dat van een even grote string dat een adressenbestand beschrijft. Hoe deze definitie te generaliseren voor quantuminformatie is geenszins voor de hand liggend aangezien het niet duidelijk is hoe precies men de amplitudes (α,β) dient te benaderen. De suggestie die we doen in dit laatste hoofdstuk bestaat eruit dat de quantum-Kolmogorov-complexiteit van een rij van quantumbits wordt gedefinieerd als de lengte van het kortste quantumcomputerprogramma dat deze rij met willekeurige accuratesse kan reproduceren, maar niet noodzakelijk perfect.

Titles in the ILLC Dissertation Series:

ILLC DS-1996-01: **Lex Hendriks**
Computations in Propositional Logic

ILLC DS-1996-02: **Angelo Montanari**
Metric and Layered Temporal Logic for Time Granularity

ILLC DS-1996-03: **Martin H. van den Berg**
Some Aspects of the Internal Structure of Discourse: the Dynamics of Nominal Anaphora

ILLC DS-1996-04: **Jeroen Bruggeman**
Formalizing Organizational Ecology

ILLC DS-1997-01: **Ronald Cramer**
Modular Design of Secure yet Practical Cryptographic Protocols

ILLC DS-1997-02: **Nataša Rakić**
Common Sense Time and Special Relativity

ILLC DS-1997-03: **Arthur Nieuwendijk**
On Logic. Inquiries into the Justification of Deduction

ILLC DS-1997-04: **Atocha Aliseda-Llera**
Seeking Explanations: Abduction in Logic, Philosophy of Science and Artificial Intelligence

ILLC DS-1997-05: **Harry Stein**
The Fiber and the Fabric: An Inquiry into Wittgenstein's Views on Rule-Following and Linguistic Normativity

ILLC DS-1997-06: **Leonie Bosveld - de Smet**
On Mass and Plural Quantification. The Case of French 'des'/'du'-NP's

ILLC DS-1998-01: **Sebastiaan A. Terwijn**
Computability and Measure

ILLC DS-1998-02: **Sjoerd D. Zwart**
Approach to the Truth: Verisimilitude and Truthlikeness

ILLC DS-1998-03: **Peter Grünwald**
The Minimum Description Length Principle and Reasoning under Uncertainty

ILLC DS-1998-04: **Giovanna d'Agostino**
Modal Logic and Non-Well-Founded Set Theory: Translation, Bisimulation, Interpolation

ILLC DS-1998-05: **Mehdi Dastani**
Languages of Perception

ILLC DS-1999-01: **Jelle Gerbrandy**
Bisimulations on Planet Kripke

ILLC DS-1999-02: **Khalil Sima'an**
Learning efficient disambiguation

ILLC DS-1999-03: **Jaap Maat**
Philosophical Languages in the Seventeenth Century: Dalgarno, Wilkins, Leibniz

ILLC DS-1999-04: **Barbara Terhal**
Quantum Algorithms and Quantum Entanglement

ILLC DS-2000-01: **Renata Wassermann**
Resource Bounded Belief Revision

ILLC DS-2000-02: **Jaap Kamps**
A Logical Approach to Computational Theory Building (with applications to sociology)

ILLC DS-2000-03: **Marco Vervoort**
Games, Walks and Grammars: Problems I've Worked On

ILLC DS-2000-04: **Paul van Ulsen**
E.W. Beth als logicus

ILLC DS-2000-05: **Carlos Areces**
Logic Engineering. The Case of Description and Hybrid Logics

ILLC DS-2000-06: **Hans van Ditmarsch**
Knowledge Games

ILLC DS-2000-07: **Egbert L.J. Fortuin**
Polysemy or monosemy: Interpretation of the imperative and the dative-infinitive construction in Russian

ILLC DS-2001-01: **Maria Aloni**
Quantification under Conceptual Covers

ILLC DS-2001-02: **Alexander van den Bosch**
Rationality in Discovery - a study of Logic, Cognition, Computation and Neuropharmacology

ILLC DS-2001-03: **Erik de Haas**
Logics For OO Information Systems: a Semantic Study of Object Orientation from a Categorial Substructural Perspective

ILLC DS-2001-04: **Rosalie Iemhoff**
Provability Logic and Admissible Rules

ILLC DS-2001-05: **Eva Hoogland**
Definability and Interpolation: Model-theoretic investigations

ILLC DS-2001-06: **Ronald de Wolf**
Quantum Computing and Communication Complexity

ILLC DS-2001-07: **Katsumi Sasaki**
Logics and Provability

ILLC DS-2001-08: **Allard Tamminga**
Belief Dynamics. (Epistemo)logical Investigations

ILLC DS-2001-09: **Gwen Kerdiles**
Saying It with Pictures: a Logical Landscape of Conceptual Graphs

ILLC DS-2001-10: **Marc Pauly**
Logic for Social Software

ILLC DS-2002-01: **Nikos Massios**
Decision-Theoretic Robotic Surveillance

ILLC DS-2002-02: **Marco Aiello**
Spatial Reasoning: Theory and Practice

ILLC DS-2002-03: **Yuri Engelhardt**
The Language of Graphics

ILLC DS-2002-04: **Wim van Dam**
On Quantum Computation Theory