

BACHELOR PROJECT

EFFICIENT SIMULATION OF DEUTSCH'S ALGORITHM.

DEPARTMENT OF MATHEMATICS, LINKÖPING UNIVERSITY

Niklas Johansson

LITH-MAT-EX - - 2014/05 - - SE

BACHELOR PROJECT

EFFICIENT SIMULATION OF DEUTSCH'S ALGORITHM.

DEPARTMENT OF MATHEMATICS, LINKÖPING UNIVERSITY

Niklas Johansson

LITH-MAT-EX - - 2014/05 - - SE

Project:	16 hp
Level:	G2
Supervisor:	Jan-Åke Larsson, Department of Electrical Engineering, Linköping University.
Examiner:	Jan-Åke Larsson, Department of Electrical Engineering, Linköping University.

Linköping, September 2014

Abstract

It is shown that Deutsch's algorithm and the Deutsch-Jozsa algorithm for two and three qubits can be efficiently implemented on a classical computer. This is done by analysing the structure of the oracles contained in these algorithms, followed by an implementation in the toy theory proposed by Robert W. Spekkens which uses 2 classical bits to simulate a qubit. Further discussion is given about how to analyse, and possible implementations, of the Deutsch-Jozsa for higher number of qubits in the toy theory.

Acknowledgement

I would like to thank my supervisor, Jan-Åke Larsson, for helpful discussions where useful insights were gained, and for introducing me to this highly interesting topic.

Contents

1	Introduction	1
1.1	Background	1
1.2	Quantum Theory	2
1.2.1	Vector Formalism	3
1.2.2	Operators and Measurement	3
1.2.3	Composite Systems	5
1.3	Quantum Computation	5
1.3.1	Qubit	6
1.3.2	Quantum Gates	7
1.3.3	Quantum Circuitry	8
1.4	Deutsch's Algorithm	9
1.5	Contextuality	13
1.6	Hypothesis	14
2	Spekkens' Toy Theory	15
2.1	Transformations	17
2.2	Measurement	18
2.3	Composite Systems	19
3	Deutsch's and the Peres-Mermin Square	21
3.1	Dismantling the Oracle	21
3.2	Simulation of Deutsch's algorithm	23
4	Spekkens and the Deutsch-Jozsa	25
4.1	Simulation of the Deutsch-Jozsa algorithm	27

5 Conclusion	33
Bibliography	35
A Number of Balanced Functions	37

Chapter 1

Introduction

This document considers the subject of *quantum computation* in correlation to the *contextuality* of a quantum measurement, both which will be briefly explained in the first part of the document. The later sections cover a study of this correlation and an outlook for further studies.

This text aims at being comprehensible to any technical graduate, regardless of their field of study.

1.1 Background

A quantum computer is a device believed to have superior computational power compared with today's modern computers, which in this text is referred to as classical computers, since they obey the same logic as classical physics. A realization of a quantum computer in the same extent as today's classical computer will yield great leaps forward in different fields of science and technology.

The theory of quantum computation emerges from the creation of hypothetical algorithms based on quantum mechanical "reasoning". It is therefore not always known whether one needs a quantum computer

to evaluate these algorithms. However, for some problems these algorithms offers an exponential speed-up compared with their known solutions in classical computation. Little is known about which quantum mechanical property that gives raise to this speed-up, but a good candidate seems to be the *contextuality* dependence of a quantum measurement. By studying this, and thereby gaining more insight in how a quantum computer works, and what advantages it offers, one will take one step closer to realizing such a device. Also there is the possibility of discovering more of these fast algorithms than what is known to this day.

1.2 Quantum Theory

This chapter will try to motivate why the theory of classical physics is not sufficient to explain the observed nature, and that the new theory is only allowed to make statistical predictions. After this motivation we will skip directly to the results of quantum theory needed throughout the text.

As explained by [1], in classical physics, true or false statements posed to a system form a Boolean algebra. However, this is not true for a quantum mechanical system, and more specific, it is the distributive law that fails

$$(A \text{ or } B) \text{ and } C \Leftrightarrow (A \text{ and } C) \text{ or } (B \text{ and } C). \quad (1.1)$$

As an example, consider the double-slit experiment in which one directs a beam of light onto a barrier with two slits, and the light that passes through the slits will be detected on a screen. If one consider a single photon passing through slit $(A \text{ or } B)$ and then arriving at the screen C we have the case to the left of (1.1). A large ensemble of these experiments will produce a wave-like interference pattern on the screen. If one utilizes a detector to decide if the photon passes through slit A or slit B , and thus switching to the case $(A \text{ and } C)$ or $(B \text{ and } C)$, the interference pattern disappears. This indicate a particle-like behaviour. Hence, the equivalence of (1.1) cannot hold for this system.

The above experiment also shows wave-like behaviour of other particles, and these phenomena, among others, led to the development of quantum mechanics. It postulates that the state of a physical system can be described by a wave function ψ that obey the superposition principle, but unlike a classical wave ψ is an abstract quantity with the probabilistic measure $\|\psi\|^2$.

1.2.1 Vector Formalism

It is useful to represent the wave function with an abstract state vector $|\psi\rangle$ [2]. The notation $|\psi\rangle$ indicating that it is a column vector and its dual $\langle\psi|$ is a row vector. More explicitly $\langle\psi| = |\psi\rangle^\dagger$ where † is the Hermitian conjugate, which in vector formalism corresponds to the transpose taken along with the complex conjugate.

The inner and outer products are then given by $\langle\psi_1|\psi_2\rangle$ and $|\psi_1\rangle\langle\psi_2|$ respectively. Further the condition $\langle\psi|\psi\rangle = 1$ expresses that the probabilities of all possible outcomes sums up to one, and is also known as the normalisation condition.

The mathematical space in which the state of our system resides, is therefore a complete inner product space (Hilbert space), and under the normalization condition this is called a projective Hilbert space \mathcal{H} [1]. Also, in this representation we have constrained ourself to a finite-dimensional space.

1.2.2 Operators and Measurement

Physical quantities such as, position, momentum, energy, etc. are referred to as *observables*. Information about these observables are stored as truth values in \mathcal{H} , and the truth values are subspaces to \mathcal{H} . We can now *generate* statements P_i about this information by creating orthogonal projections from \mathcal{H} onto these subspaces. Such a projection is given by a *projection operator* P_V , which project a vector space \mathbb{W} onto a subspace \mathbb{V} . Suppose that \mathbb{W} is a d -dimensional vector space and \mathbb{V} a k -dimensional subspace of \mathbb{W} . It is possible to produce an orthonormal set $\{|\phi_1\rangle, |\phi_2\rangle, \dots, |\phi_d\rangle\}$ acting as a basis for \mathbb{W} , such that

$\{|\phi_1\rangle, |\phi_2\rangle, \dots, |\phi_k\rangle\}$ forms an orthonormal basis for \mathbb{V} [3]. The projection of \mathbb{W} onto \mathbb{V} is then given by

$$P_{\mathbb{V}} = \sum_{n=1}^k |\phi_n\rangle\langle\phi_n|. \quad (1.2)$$

To acquire knowledge about whether our statement is true or false, one has to *announce* the statement and not just generate it; one has to measure. This is done by letting the projection operator act on our state. By not allowing the measurement to change the state of the system, we end up with

$$A|\psi\rangle = a|\psi\rangle \quad (1.3)$$

where A is now the operator to which we relate the observable, and a is a scalar labelling the value of the observable. Equation (1.3) has non-trivial solutions only when a is an eigenvalue of A , and $|\psi\rangle$ its corresponding eigenvector. If A is unitary, and the state $|\psi\rangle$ is described in a n -dimensional vector space, then A has n discrete eigenvalues $\{\lambda_j\}_{j=1}^n$ with eigenvectors $\{|\phi_j\rangle\}_{j=1}^n$ that forms an orthonormal set.

The state of the observed system can then be described by

$$|\psi\rangle = \sum_{j=1}^n c_j |\phi_j\rangle = \left(\sum_{j=1}^n P_j \right) |\psi\rangle \quad \text{hence} \quad I = \sum_{j=1}^n P_j. \quad (1.4)$$

This is called *the spectral resolution of the identity*. By recognizing that $|\phi_j\rangle$ is an eigenvector to A , one gets $A|\phi_j\rangle = \lambda_j|\phi_j\rangle$ and therefore

$$A|\psi\rangle = \sum_{j=1}^n \lambda_j c_j |\phi_j\rangle = \left(\sum_{j=1}^n \lambda_j P_j \right) |\psi\rangle \quad \text{hence} \quad A = \sum_{j=1}^n \lambda_j P_j. \quad (1.5)$$

Where P_j is the projection onto the eigenspace of A , and the eigenvalues λ_j are the possible outcomes of the measurement. This is the spectral representation of A [4]. The Born rule tells us that the probability of

measuring λ_j is given by $|c_j|^2$ and the average of a large ensemble of measurements can be written

$$\langle A \rangle = \langle \psi | A | \psi \rangle = \sum_m \sum_n c_m^* c_n \lambda_n \langle \phi_m | \phi_n \rangle = \sum_n |c_n|^2 \lambda_n. \quad (1.6)$$

If the outcome λ_j occurred, the state of the system immediately after the measurement is $|\phi_j\rangle$. A measurement under these conditions is called a *projective measurement*.

With the word observable one also implies that its value is real and produced by a Hermitian operators. For an operator A to be Hermitian the condition $A = A^\dagger$ has to be fulfilled. Also A is said to be unitary if $AA^\dagger = A^\dagger A = I$, where I is the identity operator.

Two operators A and B are said to commute if

$$[A, B] = (AB - BA) = 0 \quad (1.7)$$

which implies that their observables can be measured simultaneously.

1.2.3 Composite Systems

A system composed by two or more subsystems, where the state of the subsystems are $|\psi_1\rangle, |\psi_2\rangle, \dots, |\psi_n\rangle$ then the state of the total system is the tensor product between the subsystems

$$|\psi\rangle = |\psi_1\rangle \otimes |\psi_2\rangle \otimes \dots \otimes |\psi_n\rangle \equiv (\{|\psi_j\rangle\}_{j=1}^n)^{\otimes n} \quad (1.8)$$

and in general, superposition between these subsystem makes up the space \mathcal{H} in which $|\psi\rangle$ is represented.

1.3 Quantum Computation

The use of quantum theory for solving computational problems possibly offers an advantage over classical computation. This have however

not been proven strictly, and there is a possibility that quantum computers are no more powerful than a classical computer. A tool for analysing the difference between classical and quantum computation is the *computational complexity theory* where computational problems are categorized by their difficulty into *complexity classes*. Examples of complexity classes are: **P** the set of problems that can be efficiently solved on a classical computer; **NP** the set of problems whose solutions can be easily verified, and **PSPACE** is the set of all problems that can be solved with resources that are few in spatial size. It is known that $\mathbf{P} \subseteq \mathbf{NP}$, and **PSPACE** is believed to be bigger than **NP**. The complexity class of problems which can be efficiently solved by a quantum computer, can be shown to contain **P**, but not outside of **PSPACE**. However, how the class fits with respect to these two and **NP** is not known. Quantum algorithms have shown to efficiently solve some problems that are believed to be outside of **P**, but a better understanding of which principles that governs quantum algorithms advantage over its classical counterpart is still needed.

One principle of quantum computation without classical counterpart is *quantum parallelism*, in which one evaluation of the algorithm superimposes all answers on the initial state. However, readout can only be done of one of these answers, since a measurement will reduce the state to obtain only the result that was actually measured.

We say that a quantum algorithm can be efficiently simulated if it can be implemented on a classical computer, without the time and physical resources needed to evaluate the algorithm, growing exponentially as the problem grows.

The rest of this chapter will focus on the basic framework in which quantum algorithms are explained.

1.3.1 Qubit

A classical bit can take on the values 0 or 1 while a qubit $|\psi\rangle$ can take on the values $|0\rangle$ or $|1\rangle$ and any linear combination thereof

$$|\psi\rangle = c_0|0\rangle + c_1|1\rangle \tag{1.9}$$

under the normalisation condition $|c_0|^2 + |c_1|^2 = 1$. One can effectively write

$$|\psi\rangle = \cos \frac{\theta}{2} |0\rangle + e^{i\varphi} \sin \frac{\theta}{2} |1\rangle \quad (1.10)$$

which have a geometrical representation of a three-dimensional unit sphere, often called the *Bloch sphere*.

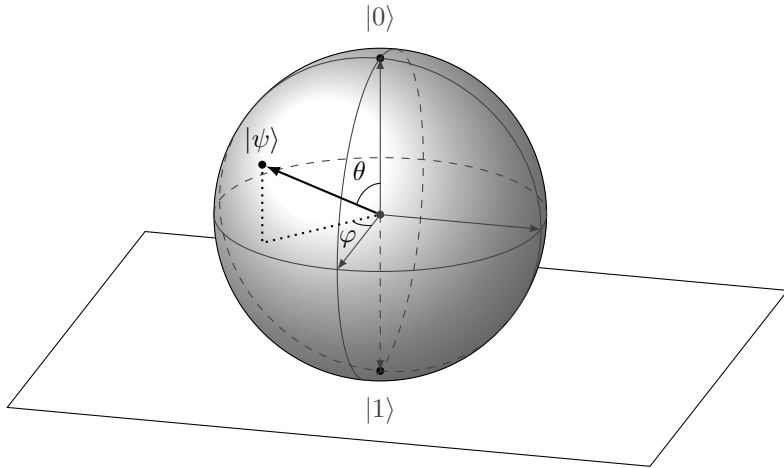


Figure 1.1: Graphical representation of the Bloch sphere.

Each point on the Bloch sphere represents a state of the qubit, however, a measurement can only yield 1 or 0 with the qubit ending up in state $|1\rangle$ or $|0\rangle$ respectively.

1.3.2 Quantum Gates

A quantum gate is a unitary transformation of the state that one puts through the gate. Moreover, any unitary transformation specifies a valid quantum gate [3].

Some specific and important single qubit gates are the Pauli matrices

$$\sigma_x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \equiv \text{---} \boxed{X} \text{---} \quad (1.11)$$

$$\sigma_z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \equiv \text{---} \boxed{Z} \text{---} \quad (1.12)$$

$$\sigma_y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} \equiv \text{---} \boxed{Y} \text{---} \quad (1.13)$$

and the *Hadamard* gate

$$H = \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \equiv \text{---} \boxed{H} \text{---} \quad (1.14)$$

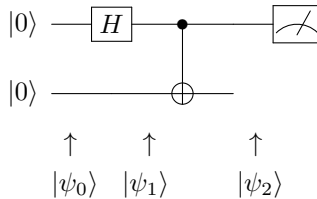
Multiple qubit gates are gates that can take two or more qubits as an argument, or rather an unitary transformation that can act on multiple qubits. For our purpose the most important one is the *controlled-NOT* gate (*CN*)

$$\begin{array}{ccc} |x\rangle & \text{---} \bullet & |x\rangle \\ & | & \\ |y\rangle & \text{---} \oplus & |y \oplus x\rangle \end{array} \quad (1.15)$$

which actually apply σ_x to the target qubit $|y\rangle$ if the controll qubit $|x\rangle = |1\rangle$. In (1.15) \oplus is addition modulo-2.

1.3.3 Quantum Circuitry

A sequence of quantum gates acting on a set of qubits is called a quantum circuit. A schematic representation can for example look like



and is evaluated from left to right. The initial state $|\psi_0\rangle$ is prepared as $|0\rangle \otimes |0\rangle$, and is often abbreviated as $|00\rangle$. Continuing the evaluation gives

$$|\psi_1\rangle = (H \otimes I)|\psi_0\rangle = \frac{|00\rangle + |10\rangle}{\sqrt{2}}$$

$$|\psi_2\rangle = CN(H \otimes I)|\psi_0\rangle = \frac{|00\rangle + |11\rangle}{\sqrt{2}}$$

and at last one measures the first qubit with an outcome of 1 or 0, both with probability $\frac{1}{2}$.

Another important part of many circuits is when a set of qubits is all put through individual Hadamard gates

$$\begin{array}{c} |x_1\rangle \text{ --- } \boxed{H} \text{ ---} \\ \vdots \qquad \qquad \vdots \\ |x_n\rangle \text{ --- } \boxed{H} \text{ ---} \end{array}$$

and it is shown in [3] that it can be written like

$$H^{\otimes n}|x\rangle^{\otimes n} = \frac{1}{\sqrt{2^n}} \sum_z (-1)^{x \cdot z} |z\rangle \quad (1.16)$$

where $x \cdot z$ is the bitwise product of x and z , summed modulo 2.

1.4 Deutsch's Algorithm

Consider the quantum circuit shown in Figure 1.2 as described by [3]

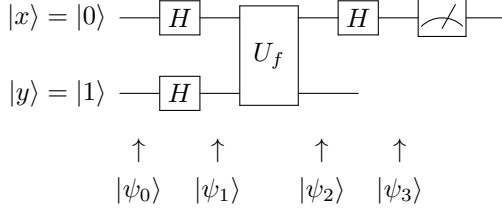


Figure 1.2: Quantum circuit performing Deutsch's algorithm.

where U_f is an *oracle* performing a unitary transformation defined by the mapping $|x, y\rangle \rightarrow |x, y \oplus f(x)\rangle$, with the condition that $f(x) \in \{0, 1\}$ is *constant* for all values of x or else it is *balanced* (returning 0 exactly as many times as it returns 1). The initial state is prepared as

$$|\psi_0\rangle = |0\rangle \otimes |1\rangle \equiv |0\rangle|1\rangle \equiv |01\rangle \quad (1.17)$$

applying the Hadamard gates to both qubits gives

$$|\psi_1\rangle = \left(\frac{|0\rangle + |1\rangle}{\sqrt{2}} \right) \left(\frac{|0\rangle - |1\rangle}{\sqrt{2}} \right) = \frac{|00\rangle - |01\rangle + |10\rangle - |11\rangle}{2}. \quad (1.18)$$

Next one applies U_f to obtain

$$|\psi_2\rangle = \frac{|0, f(0)\rangle - |0, 1 \oplus f(0)\rangle + |1, f(1)\rangle - |1, 1 \oplus f(1)\rangle}{2} \quad (1.19)$$

$$= \begin{cases} \pm \left(\frac{|0\rangle + |1\rangle}{\sqrt{2}} \right) \left(\frac{|0\rangle - |1\rangle}{\sqrt{2}} \right), & f(0) = f(1) \\ \pm \left(\frac{|0\rangle - |1\rangle}{\sqrt{2}} \right) \left(\frac{|0\rangle - |1\rangle}{\sqrt{2}} \right), & f(0) \neq f(1) \end{cases} \quad (1.20)$$

Applying the last Hadamard on the first qubit leaves us with

$$|\psi_3\rangle = \begin{cases} \pm |0\rangle \left(\frac{|0\rangle - |1\rangle}{\sqrt{2}} \right), & f(0) = f(1) \\ \pm |1\rangle \left(\frac{|0\rangle - |1\rangle}{\sqrt{2}} \right), & f(0) \neq f(1) \end{cases} \quad (1.21)$$

A measurement of $|x\rangle$ will yield 1 if $f(0) \neq f(1)$ and 0 if $f(0) = f(1)$. Hence, $f(0) \oplus f(1)$ can be determined by just one evaluation, compared with the classical solution which require at least two evaluations of $f(x)$.

Deutsch's algorithm is a special case of a more general algorithm, that [3] refers to as the Deutsch-Jozsa algorithm. Consider now again a balanced or constant function $f(x)$, with a domain

$$x \in \{0, 1, 2, 3, \dots, 2^n - 1\}. \quad (1.22)$$

To classically determine if $f(x)$ is constant with certainty, one needs to make $2^n/2 + 1$ queries to $f(x)$, while Deutsch-Jozsa require only one, and hence enabling an exponential speed-up of the classical solution.

The Deutsch-Jozsa algorithm is evaluated as the circuit shown in Figure 1.3.

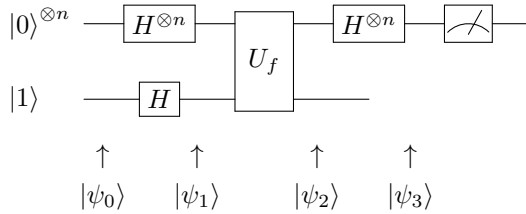


Figure 1.3: Quantum circuit performing the Deutsch-Jozsa algorithm.

The prepared state $|0\rangle^{\otimes n}|1\rangle$ is put through Hadamard gates yielding

$$|\psi_1\rangle = \left(\frac{1}{\sqrt{2^n}} \sum_x |x\rangle \right) \left(\frac{|0\rangle - |1\rangle}{\sqrt{2}} \right). \quad (1.23)$$

Followed by the oracle

$$|\psi_2\rangle = \left(\frac{1}{\sqrt{2^n}} \sum_x |x\rangle \right) \left(\frac{|f(x)\rangle - |1 \oplus f(x)\rangle}{\sqrt{2}} \right). \quad (1.24)$$

Since $f(x) \in \{0, 1\}$ this is equivalent with

$$|\psi_2\rangle = \left(\frac{1}{\sqrt{2^n}} \sum_x (-1)^{f(x)} |x\rangle \right) \left(\frac{|0\rangle - |1\rangle}{\sqrt{2}} \right). \quad (1.25)$$

Applying Hadamards to the n first qubits under the convention that $x \cdot z$ is addition modulo 2 of the bitwise product

$$\begin{aligned} |\psi_3\rangle &= H^{\otimes n} \left(\frac{1}{\sqrt{2^n}} \sum_x (-1)^{f(x)} |x\rangle \right) \left(\frac{|0\rangle - |1\rangle}{\sqrt{2}} \right) \\ &= \left(\frac{1}{\sqrt{2^n}} \sum_x (-1)^{f(x)} \left[\frac{1}{\sqrt{2^n}} \sum_z (-1)^{x \cdot z} |z\rangle \right] \right) \left(\frac{|0\rangle - |1\rangle}{\sqrt{2}} \right) \\ &= \left(\sum_z \frac{1}{2^n} \sum_x (-1)^{f(x) + x \cdot z} |z\rangle \right) \left(\frac{|0\rangle - |1\rangle}{\sqrt{2}} \right). \end{aligned} \quad (1.26)$$

At last we measure the n first qubits along $|0\rangle^{\otimes n}$, that is, we make the statement "*the first n qubits are unchanged under one evaluation*". This is a reasonable statement since we seemingly just alter the $(n+1)$:th qubit. The state after the measurement becomes

$$\left(\frac{1}{2^n} \sum_{x=0}^{2^n-1} (-1)^{f(x)} \right) |0\rangle^{\otimes n} \left(\frac{|0\rangle - |1\rangle}{\sqrt{2}} \right) \quad (1.27)$$

and we see that the probability for the statement being true is

$$\left| \frac{1}{2^n} \sum_{x=0}^{2^n-1} (-1)^{f(x)} \right|^2 = \begin{cases} 1, & f(x) \text{ is constant} \\ 0, & f(x) \text{ is balanced} \end{cases}. \quad (1.28)$$

If $f(x)$ is balanced then the probability amplitudes undergo total destructive interference (the state is being projected onto the nullspace of the subspace related to our statement) and our statement is therefore false. A constant function, on the other hand, will result in the amplitudes interfering constructively; answering that the statement is true with certainty.

1.5 Contextuality

As explained in [5], one first assumes that a measurement of an operator A depends only on the choice of the operator, and the system that the operator acts on. If A commutes with the operators B and C , then A can be measured simultaneously with B or C . The result of measuring A does not depend on whether A is measured alone or together with either B or C . However, this assumption is in contrary to what actually happens.

A measurement of a function $f(A, B)$ of the operators A and B , must also produce the outcome $f(\alpha, \beta)$ if α and β are the supposed outcomes of A and B respectively. As an example, take the Mermin-Peres array below,

$I \otimes \sigma_z$	$\sigma_z \otimes I$	$\sigma_z \otimes \sigma_z$
$\sigma_x \otimes I$	$I \otimes \sigma_x$	$\sigma_x \otimes \sigma_x$
$\sigma_x \otimes \sigma_z$	$\sigma_z \otimes \sigma_x$	$\sigma_y \otimes \sigma_y$

(1.29)

each of the nine operators have eigenvalues ± 1 . The operators in each row and in each column commute, and their product is the identity operator except for the third column which multiplies according to

$$(\sigma_z \otimes \sigma_z)(\sigma_x \otimes \sigma_x)(\sigma_y \otimes \sigma_y) = -I. \quad (1.30)$$

It is therefore not possible to assign a measurement outcome of $+1$ or -1 to the operators in (1.29), and also produce the result of measuring the functions that multiplies the three operators of each column and row. Or another example, to clarify further, lets define the operators in (1.29) as A_{ij} , where j is the number of the column and i the row. Now consider the function that is adding each product along the rows and columns, except the last column that is subtracted.

$$f(A_{ij}) = A_{11}A_{12}A_{13} + A_{21}A_{22}A_{23} + A_{31}A_{32}A_{33} + A_{11}A_{21}A_{31} + A_{12}A_{22}A_{32} - A_{31}A_{32}A_{33} \quad (1.31)$$

A measurement of this function has an upper bound of 6, but if we

instead try to address a value $\alpha_{ij} = \pm 1$ to each individual operator; simulating individual measurement, the upper bound becomes 4.

The conclusion being that, in general, measurement of an operator cannot be independent of the context of the measurement.

1.6 Hypothesis

Our hypothesis is that the exponential speed-up of Deutsch's and the Deutsch-Jozsa algorithm, compared with their classical counterpart is achieved by utilizing a process that is violating non-contextuality, and that their oracles, in this way can be associated with operators that have the same properties as those in the Mermin-Peres array. By finding these operators we hope to show that quantum contextuality is the resource used in these algorithms. If this is not the case, one should be able to efficiently simulate these algorithms in a toy theory by Robert W Spekkens [6] which uses $2N$ classical bits to simulate N qubits. Spekkens' toy theory is able to reproduce phenomena such as interference and noncommutativity, but is unable to reproduce contextuality. Therefore if a simulation in Spekkens' toy theory is possible, contextuality can then instead be excluded from being the resource utilized by these algorithms.

Chapter 2

Spekkens' Toy Theory

The toy theory, described in [6] by Spekkens, centers on a principle that restrict the amount of knowledge an observer can have about the system, called *the knowledge balance principle*. The maximal knowledge one can have about the state of a system equals the amount that is unknown. One defines knowledge so that it can be measured; knowledge is the minimal canonical set of answered yes/no questions that is enough to fully specify the real state of the system (*ontic state*). According to the knowledge balance principle this set is assumed to contain an even amount of elements, otherwise one could not have an equal amount of answered and unanswered questions. The simplest case is when the canonical set only contains two questions, which gives a system with the ability of possessing one of four ontic states. This system is called an *elementary system*. To clarify, labelling the ontic states 1, 2, 3, 4 and asking the questions $1 \vee 2$ and $1 \vee 3$ to the system (\vee read as 'or') would determine the systems ontic state. Theses two questions are then by definition a canonical set to the system. However, according to the knowledge balance principle only one of them can be answered to an observer.

The states of maximal knowledge that can be perceived by an observer (*epistemic states*); specified as disjunctions of the ontic states of an elementary system are

$$1 \vee 2 \equiv \blacksquare\blacksquare\square\square \equiv \tilde{0} \quad (2.1a)$$

$$3 \vee 4 \equiv \square\square\blacksquare\blacksquare \equiv \tilde{1} \quad (2.1b)$$

$$1 \vee 3 \equiv \blacksquare\square\blacksquare\square \equiv \tilde{0} +_1 \tilde{1} \quad (2.1c)$$

$$2 \vee 4 \equiv \square\blacksquare\square\blacksquare \equiv \tilde{0} +_2 \tilde{1} \quad (2.1d)$$

$$1 \vee 4 \equiv \blacksquare\square\square\blacksquare \equiv \tilde{0} +_4 \tilde{1} \quad (2.1e)$$

$$2 \vee 3 \equiv \square\blacksquare\blacksquare\square \equiv \tilde{0} +_3 \tilde{1}. \quad (2.1f)$$

for which one defines a graphical representation. The four cells in the graphical representation denote the ontic states, and filled cells represent the observers knowledge about which ontic state the system possesses.

The operations denoted by $+_1 +_2 +_3$ and $+_4$ are called *coherent binary operations* and are analogue to coherent superposition of states in quantum theory. The first operation can be described as defining a new epistemic state by keeping the first ontic state of both epistemic states included in the operation. In the same way $+_2$ is keeping the second ontic state of the epistemic states. These operations are thought of as a combination between $\tilde{0}$ and $\tilde{1}$ with equal weights, but with a relative phase of the second term to the first by $0, \pi, \pi/2$ and $3\pi/2$ respectively.

A single elementary system in the toy theory is analogous to a qubit in quantum theory. The six epistemic states of maximal knowledge, shown above, are analogue to the following six qubit states

$$\blacksquare\blacksquare\square\square \Leftrightarrow |0\rangle \quad (2.2a)$$

$$\square\square\blacksquare\blacksquare \Leftrightarrow |1\rangle \quad (2.2b)$$

$$\blacksquare\square\blacksquare\square \Leftrightarrow \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \quad (2.2c)$$

$$\square\blacksquare\square\blacksquare \Leftrightarrow \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) \quad (2.2d)$$

$$\blacksquare\square\square\blacksquare \Leftrightarrow \frac{1}{\sqrt{2}}(|0\rangle - i|1\rangle) \quad (2.2e)$$

$$\square\blacksquare\blacksquare\square \Leftrightarrow \frac{1}{\sqrt{2}}(|0\rangle + i|1\rangle) \quad (2.2f)$$

Note that the epistemic states associated with $|0\rangle, |+\rangle$ and $|+i\rangle$ are chosen by convention and are eigenstates to σ_z, σ_x and σ_y . In the same way as a qubit can be graphically represented by the Bloch sphere, an elementary system in the toy theory can be represented in a similar manner.

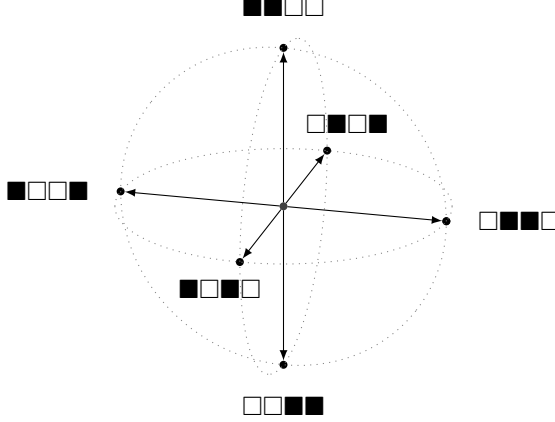


Figure 2.1: Graphical representation of an elementary system, and its six possible epistemic states.

2.1 Transformations

Transformations of an elementary system, that are allowed by the knowledge balance principle, are one-to-one maps and one-to-many maps. As an example consider the many-to-one map that transforms the ontic states 1 and 2 to the ontic state 3. This is clearly a violation of the knowledge balance principle, and all many-to-one maps will cause a violation. The transformations of our interest are the one-to-one maps which are just permutations of the ontic states. These transformations can be graphically depicted with arrows between the ontic states.

$$\begin{array}{c} \text{Diagram showing a transformation from two ontic states to one} \end{array} \quad (2.3)$$

$$\begin{array}{c} \text{Diagram showing a transformation from one ontic state to two} \end{array} \quad (2.4)$$

The first transformation (2.3) is an anti-unitary map in Hilbert space. The second (2.4) is a unitary transformation, specifically analogue to the Pauli matrix σ_x . Anti-unitary maps do not represent possible transformations of a quantum system, since they are assumed to be continuous in time. Transformation analogous to anti-unitary maps arises from the fact that the toy theory is discrete.

2.2 Measurement

The knowledge balance principle also restricts the sort of *reproducible* measurements that can be implemented; reproducible in the sense that if measured twice on the same system it will yield the same result. The fewest possible ontic states that one can associate with a single measurement are two. Therefore, the valid measurements are those that separate the ontic states into sets of two ontic states.

$$\{1 \vee 2, 3 \vee 4\} \equiv \blacklozenge\blacklozenge\lozenge\lozenge \quad (2.5)$$

$$\{1 \vee 3, 2 \vee 4\} \equiv \blacklozenge\lozenge\blacklozenge\lozenge \quad (2.6)$$

$$\{1 \vee 4, 2 \vee 3\} \equiv \blacklozenge\lozenge\lozenge\blacklozenge. \quad (2.7)$$

These sets are analogous to the three bases in quantum theory

$$\{1 \vee 2, 3 \vee 4\} \Leftrightarrow \{|0\rangle, |1\rangle\} \quad (2.8)$$

$$\{1 \vee 3, 2 \vee 4\} \Leftrightarrow \{|+\rangle, |-\rangle\} \quad (2.9)$$

$$\{1 \vee 4, 2 \vee 3\} \Leftrightarrow \{|+i\rangle, |-i\rangle\}. \quad (2.10)$$

As an example, take the epistemic state

$$\blacksquare\blacksquare\square\square \quad (2.11)$$

and perform the measurement that distinguish the states $1 \vee 2$ and $3 \vee 4$

$$\blacklozenge\blacklozenge\lozenge\lozenge. \quad (2.12)$$

Then the outcome \blacklozenge will occur with certainty. While if one performs the measurement that distinguish the states $1 \vee 3$ and $2 \vee 4$

$$\blacklozenge\lozenge\lozenge\lozenge \quad (2.13)$$

the outcome is not determined. Over a large number of such measurements the outcomes, \blacklozenge and \lozenge , will occur an equal amount of times in average. If the outcome of this measurement is \blacklozenge , then one knows that the system was in the ontic state $\blacksquare\square\square\square$ before the measurement. However, in order not to violate the knowledge balance principle, and keep the measurement reproducible, the state is updated to be in $\blacksquare\square\blacksquare\square$ after measurement.

2.3 Composite Systems

Every system is assumed to be composed by elementary systems. For a pair of elementary systems there are four questions in its canonical set, and sixteen possible ontic states. For n systems there are 4^n possible ontic states and $2n$ questions in the canonical set. For composite systems the principle will impose more constraint, since it needs to be upheld not only for the whole system, but also for each individual part of the system.

A pair of elementary systems can be graphically represented with a 4×4 array. ‘ \cdot ’ reads as ‘*and*’; then consider the epistemic state $(3 \vee 4) \cdot (3 \vee 4)$. This can be represented graphically as follows:

$$\begin{array}{cccc} 4 & \square & \square & \blacksquare & \blacksquare \\ 3 & \square & \square & \blacksquare & \blacksquare \\ 2 & \square & \square & \square & \square \\ 1 & \square & \square & \square & \square \\ & 1 & 2 & 3 & 4 \end{array} \quad (2.14)$$

Chapter 3

Deutsch's and the Peres-Mermin Square

The oracle transformation of Deutsch's algorithm clearly creates a separation between constant and balanced functions. Getting an outcome of a measurement with the value 1 we know that the function applied were constant, while measuring a value of -1 the function were balanced. The initial thought where that one could identify the realizations of the oracle with rotations of the operators in the Mermin-Peres square (1.29). Rotations in such a way that they pair together the eigenvalues and states in the same way as described above.

3.1 Dismantling the Oracle

The oracle performing the mapping $|x, y\rangle \rightarrow |x, y \oplus f(x)\rangle$ is realized with four unitary transformations, since the domain and range of $f(x)$ are defined only for two discrete values. These mappings are

$$f(x) = 0 : |x, y\rangle \rightarrow |x, y \oplus 0\rangle = |x, y\rangle \quad (3.1a)$$

$$f(x) = 1 : |x, y\rangle \rightarrow |x, y \oplus 1\rangle \quad (3.1b)$$

$$f(x) = x : |x, y\rangle \rightarrow |x, y \oplus x\rangle \quad (3.1c)$$

$$f(x) = x \oplus 1 : |x, y\rangle \rightarrow |x, y \oplus (x \oplus 1)\rangle. \quad (3.1d)$$

Figure 3.1 show the circuits with oracles that realize these mappings.

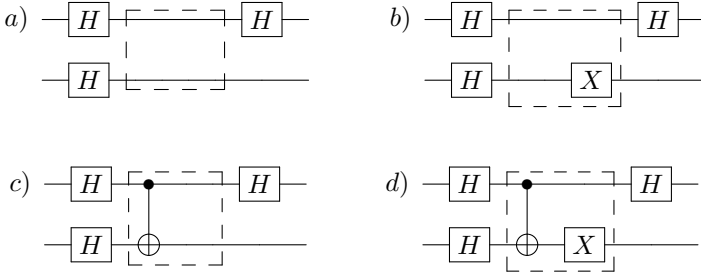


Figure 3.1: Quantum circuit performing Deutsch's algorithm with realizations of the oracle highlighted. *a), b), c)* and *d)* shows the circuits with an oracle performing (3.1a), (3.1b), (3.1c) and (3.1d) respectively.

Mappings (3.1a) and (3.1b) differs from each other only by a controlled-NOT, which results only in a global phase-shift in the output state. Since a statistical measurement depend on the square of a state, global phase will have no observable effect. These mappings can therefore be thought of as the same transformation, or at least to produce equal states. The same argument can be made for the mappings (3.1c) and (3.1d). This leaves us with only two distinguishable transformations that can be identified with transformations in the Peres-Mermin square (1.29). The identification is done by rotating the eigenspaces of the operators in the square in such a way that the eigenvalues $+1$ and -1 correspond to $|0\rangle$ and $|1\rangle$ on the first qubit respectively. By doing this one can identify that the transformations of Figure 3.1 *a)* or *b)* is the required rotation for the operator $\sigma_z \otimes I$, while the transformation of Figure 3.1 *c)* or *d)* is the rotation to $\sigma_z \otimes \sigma_z$. However, these two

transformations are insufficient to produce the same properties as those of the Mermin-Peres square, since we need at least three distinguishable transformations.

To confirm that contextuality is not being responsible for the speed-up, we set out to simulate these algorithms with a model that lacks the phenomenon of contextuality, namely Spekkens' toy theory.

3.2 Simulation of Deutsch's algorithm

We start by obtaining the operations analogous to Hadamard, Pauli-X and the control-NOT gates. The control-NOT analogue transformation CN is given in [6] as:


(3.2)

and it can be easily verified that the Pauli-X analogue is given by


 $\equiv \tilde{\sigma}_x.$
(3.3)

A Hadamard analogue is hard to find by the means of permutations. However, a rotation around the axis connecting the two eigenstates of $\tilde{\sigma}_y$ in (Figure 2.1) is given by


 $\equiv \tilde{H}.$
(3.4)

and this is exactly what we need to evolve the state before arguing it to the oracle.

To reproduce the realizations of Figure 3.1 in the toy theory we start by preparing a system composed by two elementary system in the epistemic state $(1 \vee 2) \cdot (3 \vee 4)$ (analogue to $|01\rangle$)


(3.5)

and then applying \tilde{H} to both elementary systems

$$\begin{array}{|c|c|c|c|} \hline \square & \square & \square & \square \\ \hline \square & \square & \blacksquare & \square \\ \hline \square & \square & \blacksquare & \square \\ \hline \square & \square & \blacksquare & \square \\ \hline \end{array} \xrightarrow{\tilde{H} \cdot \tilde{H}} \begin{array}{|c|c|c|c|} \hline \square & \square & \square & \square \\ \hline \square & \blacksquare & \square & \square \\ \hline \square & \square & \square & \square \\ \hline \square & \blacksquare & \square & \square \\ \hline \end{array}. \quad (3.6)$$

The unitary analogues to the mappings (3.1a) - (3.1d) will produce the same result as the quantum oracle, except from global phase. Global phase; as previous stated, have no observable effects. Applying the oracle; followed by the inverse permutation of \tilde{H} (lets call it \tilde{H}^\dagger) on the first elementary system yields:

$$\begin{array}{|c|c|c|c|} \hline \square & \square & \square & \square \\ \hline \square & \blacksquare & \square & \square \\ \hline \square & \square & \square & \square \\ \hline \square & \blacksquare & \square & \square \\ \hline \end{array} \xrightarrow{\tilde{I} \cdot \tilde{I}} \begin{array}{|c|c|c|c|} \hline \square & \square & \square & \square \\ \hline \square & \blacksquare & \square & \square \\ \hline \square & \square & \square & \square \\ \hline \square & \blacksquare & \square & \square \\ \hline \end{array} \xrightarrow{\tilde{H}^\dagger \cdot \tilde{I}} \begin{array}{|c|c|c|c|} \hline \square & \square & \square & \square \\ \hline \square & \square & \square & \square \\ \hline \square & \blacksquare & \square & \square \\ \hline \square & \blacksquare & \square & \square \\ \hline \end{array} \quad (3.7a)$$

$$\begin{array}{|c|c|c|c|} \hline \square & \square & \square & \square \\ \hline \square & \blacksquare & \square & \square \\ \hline \square & \square & \square & \square \\ \hline \square & \blacksquare & \square & \square \\ \hline \end{array} \xrightarrow{\tilde{I} \cdot \tilde{\sigma}_x} \begin{array}{|c|c|c|c|} \hline \square & \square & \square & \square \\ \hline \square & \blacksquare & \square & \square \\ \hline \square & \square & \square & \square \\ \hline \square & \blacksquare & \square & \square \\ \hline \end{array} \xrightarrow{\tilde{H}^\dagger \cdot \tilde{I}} \begin{array}{|c|c|c|c|} \hline \square & \square & \square & \square \\ \hline \square & \square & \square & \square \\ \hline \square & \blacksquare & \square & \square \\ \hline \square & \blacksquare & \square & \square \\ \hline \end{array} \quad (3.7b)$$

$$\begin{array}{|c|c|c|c|} \hline \square & \square & \square & \square \\ \hline \square & \blacksquare & \square & \square \\ \hline \square & \square & \square & \square \\ \hline \square & \blacksquare & \square & \square \\ \hline \end{array} \xrightarrow{\tilde{C}N} \begin{array}{|c|c|c|c|} \hline \square & \blacksquare & \square & \square \\ \hline \square & \square & \square & \square \\ \hline \square & \blacksquare & \square & \square \\ \hline \square & \square & \square & \square \\ \hline \end{array} \xrightarrow{\tilde{H}^\dagger \cdot \tilde{I}} \begin{array}{|c|c|c|c|} \hline \square & \blacksquare & \square & \square \\ \hline \square & \blacksquare & \square & \square \\ \hline \square & \square & \square & \square \\ \hline \square & \square & \square & \square \\ \hline \end{array} \quad (3.7c)$$

$$\begin{array}{|c|c|c|c|} \hline \square & \square & \square & \square \\ \hline \square & \blacksquare & \square & \square \\ \hline \square & \square & \square & \square \\ \hline \square & \blacksquare & \square & \square \\ \hline \end{array} \xrightarrow{\tilde{C}N(\tilde{I} \cdot \tilde{\sigma}_x)} \begin{array}{|c|c|c|c|} \hline \square & \blacksquare & \square & \square \\ \hline \square & \square & \square & \square \\ \hline \square & \blacksquare & \square & \square \\ \hline \square & \square & \square & \square \\ \hline \end{array} \xrightarrow{\tilde{H}^\dagger \cdot \tilde{I}} \begin{array}{|c|c|c|c|} \hline \square & \blacksquare & \square & \square \\ \hline \square & \blacksquare & \square & \square \\ \hline \square & \square & \square & \square \\ \hline \square & \square & \square & \square \\ \hline \end{array} \quad (3.7d)$$

Performing the measurement $\blacklozenge\blacklozenge\lozenge\lozenge$ on the first elementary system; the one depicted vertically, will clearly outcome \lozenge if the function was balanced; \blacklozenge if constant. Thus allowing a determination of this property by query the oracle only once.

This shows that Deutsch's algorithm can be efficiently implemented on a classical computer through Spekkens toy theory. Deutsch's algorithm is therefore not a valid example for how a quantum computer has an advantage over classical computers.

Chapter 4

Spekkens and the Deutsch-Jozsa

In the general case, with a data register of n bits, one gets functions $f(x)$ with a domain of 2^n . The amount of non-equivalent balanced functions grow as

$$\frac{(2^n)!}{(\frac{2^n}{2})!(\frac{2^n}{2})!} \tag{4.1}$$

which grows as 6, 70, 12870, 601080390 which is referred to in the online encyclopedia of integer sequences as the *central binomial coefficients for powers of 2*.

A general implementation of the n -qubit oracle is shown in Figure 4.1 where the $f(x)$ -gates are given by the Pauli-X or identity operators if $f(x)$ is 1 or 0 respectively.

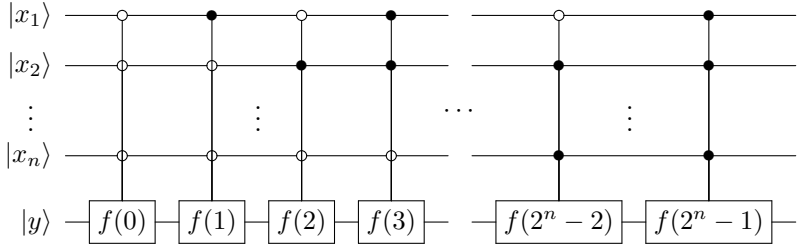


Figure 4.1: A general implementation of the oracle in the Deutsch-Jozsa algorithm; with $|x_1\rangle$ as the least significant qubit and $|x_n\rangle$ as the most significant qubit.

The oracle in Figure 4.1 is illustrative, but not the most efficient. For $n = 2$ we get functions with four numbers in its domain. Then instead of two constant functions and two balanced functions one has to realize two constant functions and *six* balanced functions. The resulting oracles can be simplified to only contain control-NOT and Pauli-X gates. This is because in a two bit binary system the individual bits along with their Boolean inverse, and combination of these by addition modulo 2, can create all balanced functions. The oracles are given by

- Control-NOT from the first qubit to the target qubit $|y\rangle$;
- Control-NOT from the second qubit to the target;
- Two control-NOTs from both qubit to the target;

and these along with a Pauli-X gate at the target qubit.

For $n = 3$ we get functions with a domain of eight bits, which yields a number of 70 non-equivalent balanced function, all of which can be generated by control-NOT, Toffoli and Pauli-X gates. This is shown in Appendix A.

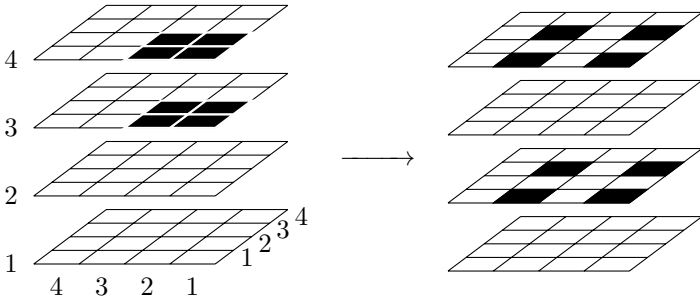
4.1 Simulation of the Deutsch-Jozsa algorithm

We will now verify that an oracle performing the balanced functions, generated by a two qubit register, can be simulated in the toy theory by:

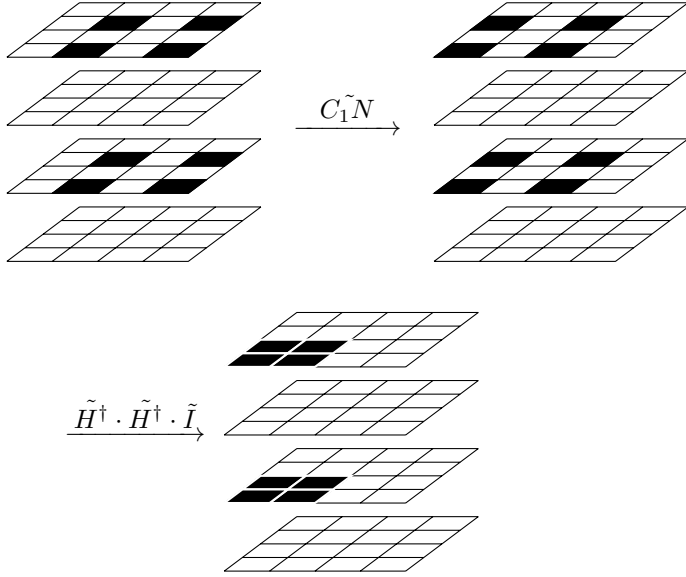
- $\tilde{C}N$ from the first elementary system to the target; $C_1\tilde{N}$
- $\tilde{C}N$ from the second elementary system to the target; $C_2\tilde{N}$
- $\tilde{C}N$ from both elementary systems to the target; $C_1\tilde{N} + C_2\tilde{N}$

and those three modulated with $\tilde{\sigma}_x$ on the target.

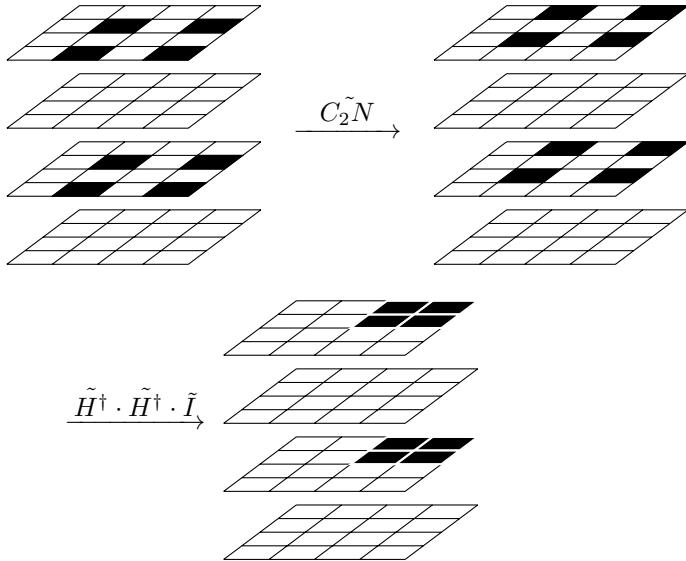
Following is a composite of three elementary system first prepared in the state $\tilde{0} \cdot \tilde{0} \cdot \tilde{1}$, and then individual permutations \tilde{H} on all three systems:



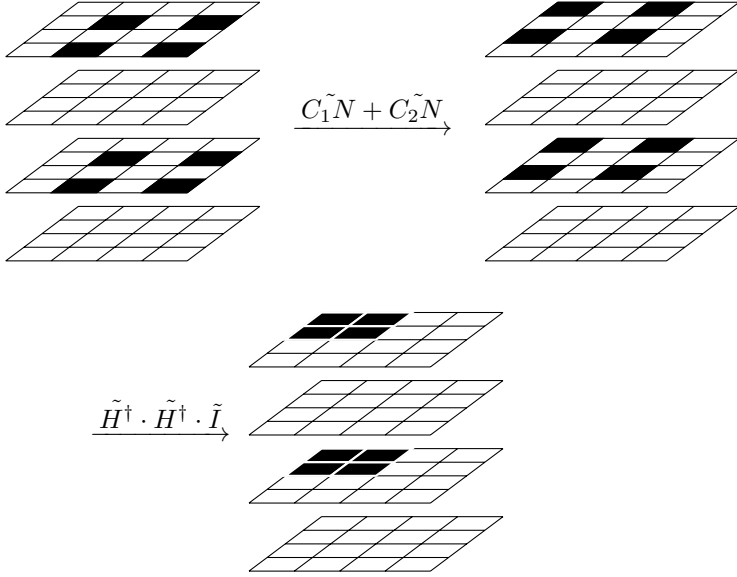
The system is then in a state invariant to $\tilde{\sigma}_x$ we need therefore only consider ourselves with the three above listed permutations. Starting with $\tilde{C}N$ from the first elementary system to the target



continuing with $\tilde{C}N$ from the second elementary system to the target:



and last we got the case of two $C\tilde{N}$ from the first and second elementary system to the target:



Any statement claiming that the first and second system (depicted horizontally) are unchanged in the above three examples, will always be false. In the case of a constant function the oracle is performed by the unity map and the unity map along with $\tilde{\sigma}_x$ (for which the state is invariant), and it is obvious that in these cases the statement will be true.

For an oracle with $n = 3$ qubit register, as shown in section 2.2, all 70 balanced functions can be generated from control-NOT, Toffoli and Pauli-X gates. We need now a transformation analogue to the Toffoli, and what works is

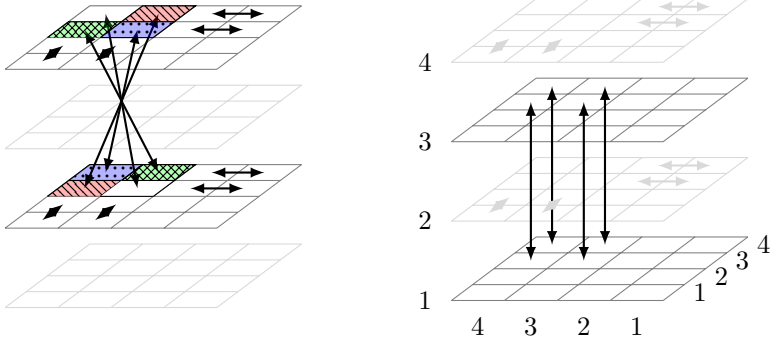


Figure 4.2: Transformation analogue to the Toffoli gate with the control systems in the horizontal plane, and the target system along the vertical axis. To clarify, the transformation is split; to the left the target system ontic states 2 and 4 ; to the right the target system ontic states 1 and 3.

As an example we will simulate this in the toy theory for the balanced function $(f(x))_{x=1}^8 = (0, 0, 0, 1, 1, 1, 1, 0)$ generated by the following oracle

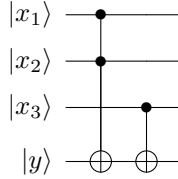
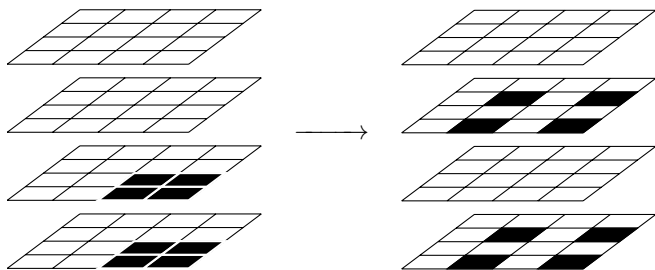


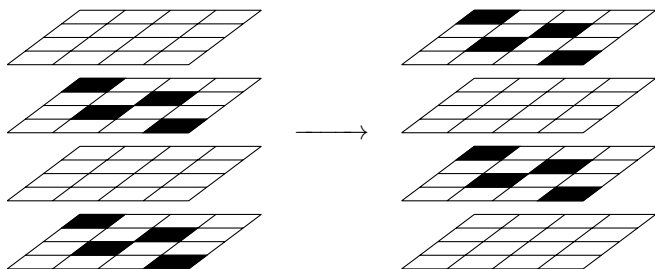
Figure 4.3: An oracle performing the balanced function $(f(x))_{x=1}^8 = (0, 0, 0, 1, 1, 1, 1, 0)$.

However, we will only depict the three first elementary systems, the first and second in the horizontal plane, and the third along the vertical. Keep in mind that the target system is in the epistemic state $3 \vee 4$ initially and in $2 \vee 4$ after the first transformation.

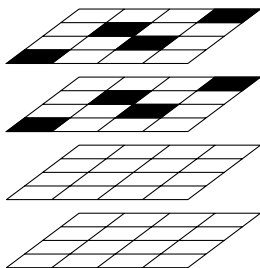
We start by preparing the systems in the state $\tilde{0} \cdot \tilde{0} \cdot \tilde{0} \cdot \tilde{1}$ and apply \tilde{H} to all four systems.



Applying the Toffoli transformation in Figure 4.2 followed by the $\tilde{C}\tilde{N}$ we get



At last we apply \tilde{H}^\dagger to the first three systems and get.



This state is clearly disjunct from the initial state so any statement made about the initial state being unchanged is false.

This shows that the Deutsch-Jozsa algorithm for two and three qubits can also be efficiently simulated on a classical computer, and we assume that this is valid for the general case with an arbitrary number of qubits. This conjecture is based on the fact that the generalized Toffoli, or n -Toffoli gates can be decomposed into Toffoli gates [7].

Chapter 5

Conclusion

Since N elementary systems in the toy theory can be implemented by $2N$ classical bits, defining the ontic states, we have shown that Deutsch's algorithm and the Deutsch-Jozsa algorithm for two and three qubits can be efficiently implemented on a classical computer. Because of this, the resources that these algorithms utilize are not quantum resources. These algorithms should instead be viewed upon as new, and more efficient classical algorithms brought forward by the counter-intuitive thinking of quantum mechanics; made intuitive by the epistemic view in the toy theory. To analyse the Deutsch-Jozsa algorithm for four qubits and more, one need to obtain a transformation analogue to the generalized Toffoli gate, and we leave this as an open problem. However, if the Toffoli transformation that we have provided here (Figure 4.2) takes all possible input states into the correct epistemic states, that is, states corresponding to quantum mechanical states given that a Toffoli is argued with the analogue input state, then a generalized Toffoli gate can be composed by single Toffoli gates in the way described by [7].

It would also be interesting to analyse other quantum algorithms in this manner. The algorithm to be next in line should be Simon's algorithm, since Deutsch-Jozsa is a special case of Simon's. Also Simon's algorithm and its classical counterpart are both stochastic algorithms, meaning that they will produce the correct result with a bounded probability,

compared with The Deutsch-Jozsa and its classical solution who are deterministic algorithms.

In the toy theory the available states are $|0\rangle$, $|1\rangle$ and four *equally* weighted states of those two. Many quantum algorithms depends on *unequally* weighted states, one being Grover's search algorithm that searches an unsorted database. Another important set of quantum algorithms are those dependent on a subroutine known as quantum Fourier transform, which uses a continuum of equally weighted states. It should therefore be interesting to analyse the above algorithms with a toy theory extended to produce a continuum of states.

Bibliography

- [1] G. B. Folland. *Quantum field theory : a tourist guide for mathematicians*. Mathematical surveys and monographs: 149. Providence, R.I. : American Mathematical Society, cop, 2008.
- [2] B. H. Bransden, C. J. Joachain, and B. H. Bransden. *Quantum mechanics*. Harlow : Prentice Hall, 2000.
- [3] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information*. Cambridge University Press, New York, NY, USA, 10th edition, 2011.
- [4] Erwin Kreyszig. *Introductory functional analysis with applications*. New York : Wiley, cop, 1978.
- [5] Asher Peres. *Quantum theory : concepts and methods*. Fundamental theories of physics: 57. Dordrecht : Kluwer, cop, 1993.
- [6] Robert W. Spekkens. Evidence for the epistemic view of quantum states: A toy theory. *Phys. Rev. A*, 75:032110, Mar 2007.
- [7] Adriano Barenco, Charles H. Bennett, Richard Cleve, David P. DiVincenzo, Norman Margolus, et al. Elementary gates for quantum computation. *Phys.Rev.*, A52:3457, 1995.

Appendix A

Number of Balanced Functions

Following is a function that returns a matrix and the number of all unique balanced functions, with a domain of 8 bits, that can be generated by CNOT, Toffoli and Pauli-X gates.

```
function [Number Matrix] = NumOfBalanced

%Defining the funtions generated by single CNOT and Toffoli.
SingleCNOT = [0 1 0 1 0 1 0 1;
              0 0 1 1 0 0 1 1;
              0 0 0 0 1 1 1 1];

SingleToffoli = [0 0 0 1 0 0 0 1;
                 0 0 0 0 0 0 1 1;
                 0 0 0 0 0 1 0 1];

% Creating funtions generated by combinations of CNOT.
CombinedCNOT = [
    mod(SingleCNOT(1,:)+SingleCNOT(2,:),2);
    mod(SingleCNOT(1,:)+SingleCNOT(3,:),2);
    mod(SingleCNOT(2,:)+SingleCNOT(3,:),2)
    mod(SingleCNOT(1,:)+SingleCNOT(2,:)+SingleCNOT(3,:),2)];
```

```

% Creating funtions generated by combinations of Toffoli.
CombinedToffoli = [
    mod(SingleToffoli(1,:)+SingleToffoli(2,:),2);
    mod(SingleToffoli(1,:)+SingleToffoli(3,:),2);
    mod(SingleToffoli(2,:)+SingleToffoli(3,:),2)
    mod(SingleToffoli(1,:)+SingleToffoli(2,:)+
SingleToffoli(3,:),2)];

% Creating funtions generated by combinations between
% CNOTs and Toffoli.
CNOT = [SingleCNOT;
        CombinedCNOT];

Toffoli = [SingleToffoli;
           CombinedToffoli];

[M Domain] = size(Toffoli);
[N Domain] = size(CNOT);

Mat = [CNOT;Toffoli];

for n = 1 : N
    for m = 1 : M
        func = mod(Toffoli(m,:)+CNOT(n,:),2);
        Partial(m,:) = func;
    end
    Mat = [Mat; Partial];
end

% Appending the alterations from adding a Pauli-X
BoolInverse = mod(Mat + ones(size(Mat)),2);
Mat = [Mat ;BoolInverse];

% Discarding all unbalanced functions
m = 1;
[K Domain] = size(Mat);
for k = 1 : K
    if sum(Mat(k,:)) == Domain/2
        Balanced(m,:) = Mat(k,:);
        m = m + 1;
    end
end

```

```
end

% Sorting out all unique functions
Matrix = unique(Balanced , 'rows');

[Number Domain] = size(Matrix);
```

```
ans =
```

```
70
```



LINKÖPINGS UNIVERSITET

Copyright

The publishers will keep this document online on the Internet - or its possible replacement - for a period of 25 years from the date of publication barring exceptional circumstances. The online availability of the document implies a permanent permission for anyone to read, to download, to print out single copies for your own use and to use it unchanged for any non-commercial research and educational purpose. Subsequent transfers of copyright cannot revoke this permission. All other uses of the document are conditional on the consent of the copyright owner. The publisher has taken technical and administrative measures to assure authenticity, security and accessibility. According to intellectual property law the author has the right to be mentioned when his/her work is accessed as described above and to be protected against infringement. For additional information about the Linköping University Electronic Press and its procedures for publication and for assurance of document integrity, please refer to its WWW home page: <http://www.ep.liu.se/>

Upphovsrätt

Detta dokument hålls tillgängligt på Internet - eller dess framtida ersättare - under 25 år från publiceringsdatum under förutsättning att inga extraordinära omständigheter uppstår. Tillgång till dokumentet innebär tillstånd för var och en att läsa, ladda ner, skriva ut enstaka kopior för enskilt bruk och att använda det oförändrat för ickekommersiell forskning och för undervisning. Överföring av upphovsrätten vid en senare tidpunkt kan inte upphäva detta tillstånd. All annan användning av dokumentet kräver upphovsmannens medgivande. För att garantera äktheten, säkerheten och tillgängligheten finns det lösningar av teknisk och administrativ art. Upphovsmannens ideella rätt innefattar rätt att bli nämnd som upphovsman i den omfattning som god sed kräver vid användning av dokumentet på ovan beskrivna sätt samt skydd mot att dokumentet ändras eller presenteras i sådan form eller i sådant sammanhang som är kränkande för upphovsmannens litterära eller konstnärliga anseende eller egenart. För ytterligare information om Linköping University Electronic Press se förlagets hemsida <http://www.ep.liu.se/>

© 2014, Niklas Johansson