

# QUANTUM CELLULAR AUTOMATA

**Wim van Dam**

Master's thesis (387)  
Computing Science Institute  
University of Nijmegen, The Netherlands  
(current email-address: [wimvdam@cwj.nl](mailto:wimvdam@cwj.nl))

*written under the supervision of*  
prof. dr. ir. P.M.B. Vitányi (C.W.I. & University of Amsterdam)  
*and*  
prof. C.H.A. Koster (University of Nijmegen)

Summer 1996



# Acknowledgments

*Writing a master's thesis can sometimes be a lonesome activity, I therefore feel privileged to be able to thank the following people.*

*First of all I want to thank Kees Koster and Paul Vitányi for their experienced guidance and interest during the writing of this thesis. I also thank André Berthiaume for giving me a 'first hand' introduction in quantum computing, his knowledge of this field prevented me from making a lot of typical beginner's mistakes. Sjoerd Stallinga and Paul Bastiaansen should be mentioned for explaining me the essentials of quantum mechanics. The hospitality of Christoph Dürr and Huong Lê Thanh made an inspiring and enjoyable visit to Paris possible in the summer of '95.*

*In 1996, the Institute for Scientific Interchange Foundation enabled me to go to the 'Torino workshop on Quantum Computing'. During this workshop I enjoyed the interesting discussions with Tommaso Toffoli, Seth Lloyd, Norman Margolus, Adriano Barenco and David Meyer.*

# Contents

<b>1</b>	<b>Quantum Physics</b>	<b>9</b>
1.1	The Two-Slit Experiment . . . . .	9
1.2	The Bra-ket notation . . . . .	9
1.3	The Mathematics of Quantum Physics . . . . .	11
1.4	The Superposition of Basis States . . . . .	12
1.5	Evolution of Quantum Mechanical Systems . . . . .	13
1.6	Expanding the State Space . . . . .	14
1.6.1	Notational Customs . . . . .	15
1.7	The Effects of Measurement . . . . .	15
<b>2</b>	<b>Quantum Computing</b>	<b>17</b>
2.1	Quantum Memory . . . . .	17
2.1.1	Quantum Bits and Quantum Registers . . . . .	17
2.1.2	Entanglement . . . . .	17
2.2	Quantum Gates . . . . .	18
2.2.1	Proper Quantum Gates . . . . .	19
2.3	Quantum Gate Circuits . . . . .	20
2.3.1	A Universal Quantum Gate . . . . .	20
2.4	Quantum Turing Machines . . . . .	21
2.4.1	Well Formed QTMs . . . . .	22
2.5	Algorithms for Quantum Computers . . . . .	23
<b>3</b>	<b>Classical Cellular Automata</b>	<b>25</b>
3.1	Cellular Automata . . . . .	25
3.2	Characteristics of Cellular Automata . . . . .	26
3.3	Formal definition of Cellular Automata . . . . .	27
3.3.1	The Cellular Automata Model . . . . .	28
3.4	Reversible Cellular Automata . . . . .	28
<b>4</b>	<b>Quantum Cellular Automata</b>	<b>30</b>
4.1	The Size of the State Space . . . . .	30
4.2	Quantum Cellular Automata . . . . .	30
4.2.1	Normalized QCA . . . . .	31
4.2.2	Well-formed QCA . . . . .	32
4.3	Proving Well-formedness . . . . .	33
4.3.1	A definition of Balancedness . . . . .	35
4.4	Quiescent Quantum Cellular Automata . . . . .	36
4.5	Partitioned Quantum Cellular Automata . . . . .	37

<b>5</b>	<b>Quantum Gate Cellular Automata</b>	<b>39</b>
5.1	Description of QGCA . . . . .	39
5.1.1	Formal Definition of QGCA . . . . .	41
5.2	Preliminaries . . . . .	41
5.2.1	The Shift Neighborhood Scheme . . . . .	41
5.2.2	Two state QGCA . . . . .	42
5.2.3	Combining neighborhood schemes . . . . .	43
5.3	Every QGCA describes a QCA . . . . .	43
<b>6</b>	<b>QCA, PQCA, and QGCA are Equivalent</b>	<b>45</b>
6.1	A Definition of Equivalence . . . . .	45
6.2	Some Preliminary Results . . . . .	47
6.3	Shift Neighborhood Scheme is Universal . . . . .	48
6.3.1	Periodic QGCA . . . . .	48
6.3.2	Simulating Neighborhood Schemes . . . . .	48
6.3.3	Simulating Periodic QGCA . . . . .	49
6.4	Every QCA can be simulated by a QGCA . . . . .	51
6.5	QGCA and PQCA are Equivalent . . . . .	52
6.6	Conclusion . . . . .	53
<b>7</b>	<b>A Universal Quantum Cellular Automaton</b>	<b>54</b>
7.1	Simulating QCA with QTMS . . . . .	54
7.2	A Universal Quantum Cellular Automaton . . . . .	55
7.2.1	Periodic Universal Gate Arrays . . . . .	55
7.2.2	Simulating the Wiring . . . . .	55
7.2.3	The Universal Quantum Cellular Automaton . . . . .	57
7.3	Conclusions . . . . .	59
<b>A</b>	<b>Unitary Transformations</b>	<b>60</b>
A.1	Finite dimensional Transformations . . . . .	60
A.2	Infinite dimensional Transformations . . . . .	60
A.3	Exponential Expressions . . . . .	61
<b>B</b>	<b>Proving Well-Formedness</b>	<b>62</b>
<b>C</b>	<b>Simulating Neighborhood Schemes</b>	<b>65</b>
<b>D</b>	<b>Mapping the Calibrations</b>	<b>67</b>
<b>E</b>	<b>Sources of Information</b>	<b>68</b>

# List of Figures

1.1	Young's two-slit experiment . . . . .	10
2.1	A small quantum gate circuit . . . . .	21
3.1	Example of a one-dimensional cellular automaton . . . . .	26
3.2	Space/time behavior of a classical one-dimensional cellular automaton . . . . .	27
4.1	Simple quantum cellular automaton . . . . .	31
4.2	A partitioned quantum cellular automaton . . . . .	38
5.1	A quantum gate cellular automaton . . . . .	40
5.2	The Shift neighborhood scheme . . . . .	42
6.1	Simulation of a neighborhood scheme . . . . .	49
7.1	A periodic universal gate array . . . . .	56
7.2	A periodic <i>IXU</i> array . . . . .	56

# Index

- balanced QCA, 32
- basis state, 9
- bra, 6
- bra-ket notation, 6
- bracket notation, 6
  
- contiguous neighborhood scheme, 28
  
- entangled states, 11
- equivalent QCA, 43
- equivalent sets of QCA, 43
  
- general function of QCA, 30
  
- hermitian conjugate, 10
- Hilbert space, 9
- Hilbertian space, 8
  
- inner product, 9
- interference, 6
  
- ket, 6
  
- neighborhood scheme, 36
- norm, 9
- norm preserving, 10
- normalization constraint, 10
- normalized neighborhood scheme, 28
  
- partitioned QCA, 34
- periodic  $IXU$  array, 52
- periodic QGCA, 45
- periodic  $U$  gate array, 52
- PQCA, 34
- probability, 7
- probability amplitude, 6
- product state, 11
- proper behavior, 29
- proper QCA, 29
- proper state, 29
  
- QCA, 27
- QGCA, 38
- QQCA, 33
- quantum cellular automata, 27
- quantum gate cellular automata, 38
  
- quantum Turing machine, 18
- quiescent quantum cellular automata, 33
- quiescent state, 33
  
- requested behavior, 16
  
- shift equivalent automata, 29
- shift equivalent states, 29
- shift transformation, 28
- shift-dynamical systems, 24
- simple transformation, 42
- simulating QCA, 43
- simulating sets of QCA, 43
- State space, 9
- superposition, 9
  
- tensor product, 11
- translation invariant systems, 24
  
- $\mathcal{U}$  automaton, 54
- unitary matrix, 10
- unitary QQCA, 33
- unitary transformation, 10
- universal automaton, 54
  
- well-formed general function, 30
- well-formed QCA, 29
- well-formed QQCA, 33

# Introduction

*“I’m not happy with all the analyses that go with just the classical theory, because nature isn’t classical, dammit”*

This brusque statement was made by Richard Feynman in his keynote speech ‘Simulating Physics with Computers’ in 1981 and heralded the new field of *Quantum Computation*. He stressed the fact that computers whose behaviour depend on quantum mechanical phenomena are perhaps more powerful than ordinary, ‘classical’ computers. This has proven to be true and has raised a growing interest among both physicists and computer scientists in the theory of Quantum Computing.

In 1994 Peter Shor showed the existence of an algorithm *for a quantum computer* that can factor any number within polynomial time. This is a remarkable result because it is generally believed that *on a classical computer* this problem takes up an exponential amount of time, which is the reason that the difficulty of factoring is used in most modern cryptography protocols. More recently, Lov Grover demonstrated that a quantum computer can find an entry in an unsorted list of size  $N$  with a time-complexity proportional to only  $\sqrt{N}$ . It is not known how to do this for any computer as-we-know-it-today.

We will apply the paradigm-shift from classical to quantum computation to the model of *Cellular Automata*. Cellular Automata are commonly used to describe discrete systems with a parallel and uniform time-evolution. ‘Conway’s Game of Life’ is a well-know example of such an automaton on a two dimensional grid. Just as Turing-machines are appropriate when considering sequential computation, cellular automata provide us with a theoretical abstraction of massive parallel computation. Cellular automata are also used in physics, biology and other areas to describe systems such as fluids, gases and DNA-sequences.

After investigating some of the typical characteristics of this model, it will be shown that there exists a *Universal Quantum Cellular Automaton* that can simulate any other automaton with only linear slowdown. This result may be of use for the actual construction of a quantum computer because several authors have suggested that it may be easier to construct a Quantum Cellular Automaton than a Quantum Turing-machine.



# Chapter 1

## Quantum Physics

We shortly describe some concepts and laws of quantum mechanics. We will restrict ourselves to the fundamentals which are necessary and sufficient to understand quantum computation.

### 1.1 The Two-Slit Experiment

As a start we will look at Young's two-slit experiment which shows us the basic properties of quantum physics (Figure 1.1). After describing this well-known experiment in words, the bracket notation will be introduced. The mathematical background of this notational tool will be explained. We refer to the standard introductory books for a more thorough and detailed explanation of quantum physics [10, 14, 24].

The set-up of the two-slit experiment is as follows. A source  $S$  sends a photon through a wall with two-slits  $A$  and  $B$  on a projection screen  $P$ . The probability of a photon arriving at position  $x$  on this screen will depend on the value of  $x$ . When we repeat this experiment for a large number of photons this probability-distribution on  $P$  will produce an interference pattern which can be explained by the wave-like properties of light.

The problem is that this interference also occurs when the source emits photons at such a low rate that only one photon at the time travels from  $S$  to  $P$ . This means that a photon can interfere with itself. To understand this phenomenon we have to look more precise to what is happening.

For every photon that arrives at  $x$  there are two possibilities for its history: it either went through slit  $A$  or slit  $B$  (from the source  $S$ ). After the photon has been observed at  $x$  it is impossible to determine if it used  $A$  or  $B$  to pass the wall. We therefore say that the two routes "from  $S$ , through  $A$ , to  $x$ " and "from  $S$ , through  $B$ , to  $x$ " are *indistinguishable*. This is a necessary feature of the experiment because *when we do* track down the way the photon goes, the interference pattern vanishes. (This can be done by setting up two detectors at the two slits.) In short: a photon that goes through specifically one slit cannot interfere with itself.

### 1.2 The Bra-ket notation

We now know that the interference depends on the set of possible paths a photon can travel from  $S$  to  $x$ . This can be described by the *bra-ket* notation. This is the conventional way of describing quantum mechanical events and was introduced by Paul Dirac [17]. The mathematical basis of this formalism will be explained in the next section.

When it is possible to go from a configuration  $X$  to configuration  $Y$  we can describe this with a non-zero *probability amplitude*  $\langle Y|X\rangle$ . This amplitude is a complex number with a norm  $\leq 1$ . The  $\langle Y|$  part of this expression is called a *bra*, the  $|X\rangle$  part is a *ket*. The whole term is therefore called a *bra-ket*. Inside this bracket are descriptions (note: *not* mathematical expressions). A bracket should be read from right to left to understand its meaning. If it is not possible to go

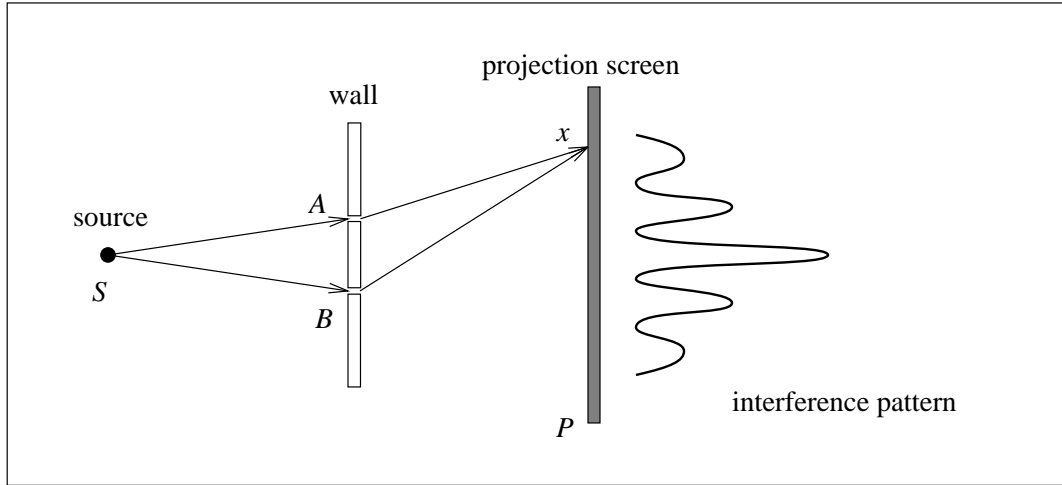


Figure 1.1: Young's two-slit experiment. The source  $S$  transmits photons through a wall with two slits ( $A$  and  $B$ ) on a projection screen  $P$ . The arrival of a single photon at the screen  $P$  is described as the superposition of the two possible trajectories allowed by the two slits. Because the probability amplitude of arriving at position  $x$  via  $A$  differs (in general) from the amplitude of arriving via  $B$ , an interference pattern will appear. If a measurement is performed in order to determine through which slit the photon has traveled, the interference pattern at  $P$  will disappear. This is because the two trajectories are now made distinguishable which destroys the initial superposition of the photon. In reality distance between the slits  $A$  and  $B$  has to be much smaller to get a noticeable interference pattern.

from  $X$  to  $Y$  the amplitude  $\langle Y|X \rangle$  will be 0. We say that  $X$  and  $Y$  are mutual *orthogonal* when  $\langle Y|X \rangle = 0$ .

By multiplying the amplitude  $\langle Z|Y \rangle$  with the amplitude  $\langle Y|X \rangle$ , we get the amplitude of going from situation  $X$  to  $Z$ , thus:  $\langle Z|X \rangle = \langle Z|Y \rangle \langle Y|X \rangle$ .

A probability amplitude tells us the probability  $|\langle Y|X \rangle|^2$  of measuring  $Y$  given the initial configuration  $X$ . The amplitude  $\langle Y|X \rangle$  cannot be measured directly.

Because the labels  $X$  and  $Y$  describe situations they can sometimes be decomposed into more specific descriptions which are mutual exclusive. If we take the example for which  $|X \rangle = |X_1 \text{ or } X_2 \rangle$  then the amplitude of going from state  $|X \rangle$  to state  $\langle Y|$  can also be decomposed. If the parts  $|X_1 \rangle$  and  $|X_2 \rangle$  are *indistinguishable* we can *add* the amplitudes such that:

$$\langle Y|X \rangle = \langle Y|X_1 \text{ or } X_2 \rangle = \langle Y| \{ |X_1 \rangle + |X_2 \rangle \} = \langle Y|X_1 \rangle + \langle Y|X_2 \rangle$$

When the parts are distinguishable we cannot add the amplitudes. The same reasoning goes for the bra, for example:  $\langle Y|X \rangle = \langle Y_1 \text{ or } Y_2|X \rangle = \langle Y_1|X \rangle + \langle Y_2|X \rangle$ .

Let us now restate the two-slit experiment with the use of this notation.

**EXAMPLE 1.1** *The amplitude of an emitted photon to reach position  $x$  at the screen is described by:*

$$\langle \text{arrives at } x \mid \text{emitted from } S \rangle \equiv \langle x|S \rangle$$

As we have seen this trajectory can be divided into two indistinguishable routes "going through slit  $A$ " and "going through  $B$ ", therefore:

$$\begin{aligned} \langle x|S \rangle &= \langle x| \text{ from } A \rangle \langle \text{through } A|S \rangle + \langle x| \text{ from } B \rangle \langle \text{through } B|S \rangle \\ &\equiv \langle x|A \rangle \langle A|S \rangle + \langle x|B \rangle \langle B|S \rangle \end{aligned}$$

The probability of a photon to reach  $x$  from  $S$  is thus calculated by:

$$|\langle x|S\rangle|^2 = |\langle x|A\rangle\langle A|S\rangle + \langle x|B\rangle\langle B|S\rangle|^2$$

This shows how the amplitudes  $\langle x|A\rangle\langle A|S\rangle$  and  $\langle x|B\rangle\langle B|S\rangle$  can interact and thereby increase or decrease the value  $|\langle x|S\rangle|^2$ . If the  $A$  and  $B$ -part have the same phase, the probability is amplified. Take for example the values:  $\langle x|A\rangle\langle A|S\rangle = \langle x|B\rangle\langle B|S\rangle = 1/2$ . The probability of a photon to reach  $x$  via  $A$  or  $B$  individually equals  $|1/2|^2 = 1/4$ . But the total probability is  $|1/2 + 1/2|^2 = 1$ .

The two parts ‘cancel’ each other out if they have opposite phases. To see this we have to change  $\langle x|B\rangle\langle B|S\rangle$  into  $-1/2$  such that  $|\langle x|S\rangle|^2 = 0$ . This phase-dependent behavior explains the interference pattern in our experiment. The phase of a photon to go from a slit to a position  $x$  on the screen depends on the distance between the slit and  $x$ . Because the distances  $A - x$  and  $B - x$  are different for every  $x$ , the two amplitudes  $\langle x|A\rangle$  and  $\langle x|B\rangle$  sum up differently for every position  $x$ .

When we make the  $A$  and  $B$  parts distinguishable (for example by using a photon-detector at the slits) this expression changes. Because we cannot add the amplitudes  $\langle x|A\rangle\langle A|S\rangle$  and  $\langle x|B\rangle\langle B|S\rangle$  anymore, the probability of measuring a photon at  $x$  now equals:

$$\begin{aligned} |\langle x|S\rangle|^2 &= |\langle \text{arrives at } x \text{ through } A | S\rangle|^2 + |\langle \text{arrives at } x \text{ through } B | S\rangle|^2 \\ &= |\langle x|A\rangle\langle A|S\rangle|^2 + |\langle x|B\rangle\langle B|S\rangle|^2 \end{aligned}$$

We simply have to add the two probabilities.

The difference between the two calculations lies in the fact that amplitudes are complex-valued numbers but probabilities are always in the domain  $[0, 1] \subset \mathbb{R}$ .  $\diamond$

### 1.3 The Mathematics of Quantum Physics

We will now make the bracket notation more precise and meaningful in a mathematical sense. This can be done by defining a complex valued vector space such that the  $|\cdot\rangle$  and  $\langle\cdot|$  expressions are vectors with an inner product  $\langle\cdot|\cdot\rangle$ . For our purposes it will be sufficient to use the following two definitions of a *Hilbertian space* and a *state space*.

**DEFINITION 1.1 (Hilbertian space)** For every set of basis states  $B$ , the Hilbertian space  $\ell_2(B)$  is the complex-valued linear space on the domain  $B$  with a bounded norm. In other words, for every

$$X = \sum_{\xi \in B} \alpha_\xi \cdot \xi$$

with  $\alpha_\xi \in \mathbb{C}$ , the vector  $X$  is an element of  $\ell_2(B)$  if and only if  $(\alpha^*$  is the complex conjugate of  $\alpha \in \mathbb{C})$

$$\sum_{\xi \in B} \alpha_\xi \alpha_\xi^* < \infty$$

This space is equipped with an inner product  $\langle\cdot|\cdot\rangle : \ell_2(B) \times \ell_2(B) \rightarrow \mathbb{C}$  and a norm  $\|\cdot\| : \ell_2(B) \rightarrow \mathbb{R}$  defined by:

$$\langle X, Y \rangle = \left\langle \left( \sum_{\xi \in B} \alpha_\xi \cdot \xi \right), \left( \sum_{\xi \in B} \beta_\xi \cdot \xi \right) \right\rangle = \sum_{\xi \in B} \alpha_\xi \beta_\xi^*$$

and  $\|X\| = \sqrt{\langle X, X \rangle}$  for every  $X, Y \in \ell_2(B)$ . By definition all the vectors in  $B$  are mutually orthogonal and have norm 1.

Note that we do not claim that this is the formal definition of a *Hilbert space*. The above described  $\ell_2(\cdot)$  space *is only an example* of a Hilbert-space. It can be shown that the *inner product* and *norm* have the properties:

$$\begin{aligned} \langle X, \alpha Y + \beta Z \rangle &= \alpha \langle X, Y \rangle + \beta \langle X, Z \rangle & \|X\| &\geq 0 \\ \langle X, Y \rangle &= \langle Y, X \rangle^* & \|X\| &= 0 \text{ if and only if } X = 0 \\ \langle X, X \rangle &\geq 0 & \|X\| + \|Y\| &\geq \|X + Y\| \\ \langle X, X \rangle &= 0 \text{ if and only if } X = 0 & \|\alpha X\| &= |\alpha| \|X\| \end{aligned}$$

for every  $\alpha, \beta \in \mathbb{C}$  and  $X, Y, Z \in \ell_2(B)$ . Which are exactly the properties we want them to have.

The complex-values in combination with the notation of the inner product already forecasts the next definition which formalizes the bra-ket notation.

**DEFINITION 1.2 (State space)** *Given a set of basis states  $B$ , the state space  $\mathcal{H}_B$  equals the Hilbertian space  $\ell_2(B)$  restricted to the vectors with norm 1. The bracket notation enables us to describe vectors by  $\langle X|$  and  $|Y\rangle \in \mathcal{H}_B$ . If we restrict the inner product on  $\ell_2(B)$  to the domain  $\mathcal{H}_B$ , we define with this notation:*

$$\langle X|Y\rangle \equiv \langle\langle X|, |Y\rangle\rangle$$

The Hilbertian space  $\ell_2(B)$  is a complex, linear space spanned by the basis set  $B$ . Every vector is in  $\ell_2(B)$  is therefore a linear combination of basis states  $\in B$ . Because the state space  $\mathcal{H}_B$  is a subset of  $\ell_2(B)$ , every state  $|X\rangle \in \mathcal{H}_B$  is also described by a linear combination of basis states. A state does not *determine* its basis set. It is therefore possible to have two different basis sets  $A \neq B$  with  $\mathcal{H}_A = \mathcal{H}_B$ .

## 1.4 The Superposition of Basis States

The possible configurations of a physical system are described by the state space  $\mathcal{H}_B$  which is spanned by a set of basis states  $B$ . The number of basis states (which equals the dimension of  $\mathcal{H}_B$ ) can be infinite and even uncountable (like the arriving of a photon on a position  $x \in \mathbb{R}$  in our example). From now on we will only look at finite dimensional state spaces. A basis state  $\xi \in B$  will be denoted by  $|\xi\rangle$  to preserve the bracket notation.

As a consequence of this, every state  $|X\rangle$  can be described by a linear combination on the basis states  $B$ :

$$|X\rangle = \sum_{\xi \in B} \alpha_\xi |\xi\rangle$$

with  $\alpha \in \mathbb{C}$ . It is said that  $X$  is in a *superposition of basis states  $B$* .

Because  $B$  is an orthogonal basis set with all vectors norm 1 we have for every  $\xi, \chi \in B$ :

$$\langle \chi | \xi \rangle = \begin{cases} 1 & \text{if } \chi = \xi \\ 0 & \text{if } \chi \neq \xi \end{cases}$$

By multiplying the linear combination of  $X$  on the left with a basis state  $\langle \chi |$  we therefore get:

$$\langle \chi | X \rangle = \sum_{\xi \in B} \alpha_\xi \langle \chi | \xi \rangle = \alpha_\chi$$

for every  $\chi \in B$ . This shows us how the inner product in the Hilbertian space  $\ell_2(B)$  relates to the probability amplitudes in quantum mechanics and vice versa. By definition it holds that

$$|X\rangle = \sum_{\xi \in B} |\xi\rangle \langle \xi | X \rangle$$

for every state  $|X\rangle$  in the state space  $\mathcal{H}_B$ . The probability of measuring a state  $|X\rangle$  in the basis state  $\xi$  equals  $|\langle\xi|X\rangle|^2$ . By the ‘norm 1’ constraint on the state space  $\mathcal{H}_B$  we see that the overall probability of measuring a configuration  $X$  in a basis state is 1:

$$\sum_{\xi \in B} |\langle\xi|X\rangle|^2 = 1$$

for every  $|X\rangle \in \mathcal{H}_B$ . This is the *normalization restriction* which will play an eminent role in this thesis. Only states and transformations which respect this restriction are sensible from a physical point of view. We say that a state space and its transformations have to be *proper* or *well-formed*. This normalization condition for  $\mathcal{H}_B$  is a restriction on the Hilbertian space  $\ell_2(B)$ . The classical state space  $B$  is a subset of  $\mathcal{H}_B$  where every amplitude has the value 0 or 1. We therefore can write:

$$B \subsetneq \mathcal{H}_B \subsetneq \ell_2(B)$$

## 1.5 Evolution of Quantum Mechanical Systems

If we want to be more precise about the (time)-evolution a system undergoes, we have to use *transformations* which describe this evolution in the state space. The transformations are conventionally called *operators* or *time operators*. If under the influence of an operator  $U$  a system evolves from state  $|X\rangle$  to a state  $|Y\rangle$ , we denote this by:

$$U|X\rangle = |Y\rangle \quad \text{or} \quad \langle Y|U|X\rangle = 1 \quad \text{or} \quad |X\rangle \xrightarrow{U} |Y\rangle$$

If we want this operator to obey the laws of quantum mechanics  $U$  has to be both *linear* and *norm preserving*. It has to respect the normalization condition and the superposition principle of the state space  $\mathcal{H}_B$ . We therefore can write

$$U|X\rangle = U \left( \sum_{\xi \in B} \alpha_\xi |\xi\rangle \right) = \sum_{\xi \in B} \alpha_\xi \cdot U|\xi\rangle$$

for every  $|X\rangle \in \mathcal{H}_B$ . Because  $U$  is a linear transformation it can be described by a matrix  $M_U$ . If we want this matrix to be norm preserving it has to be a unitary matrix. In order to define unitarity we first have to define the *hermitian conjugate* of a matrix.

**DEFINITION 1.3 (Hermitian conjugate)** *The hermitian conjugate of a matrix  $M$  is the matrix  $M^\dagger$  with  $[M^\dagger]_{ij} \equiv [M]_{ji}^*$  for every index  $i$  and  $j$ .*

This enables us to define:

**DEFINITION 1.4 (Unitary matrix)** *A unitary matrix  $M$  is a complex valued matrix such that the hermitian conjugate  $M^\dagger$  is the inverse of  $M$  and vice versa. Therefore:*

$$M \cdot M^\dagger = M^\dagger \cdot M = I$$

A *unitary transformation* is thus defined by a unitary matrix. From now on we will make no distinction between a function  $U$  and the corresponding matrix  $U$ .

**DEFINITION 1.5 (Unitary transformation)** *A transformation  $U : \mathcal{H}_B \rightarrow \mathcal{H}_B$  of the state space  $\mathcal{H}_B$  is unitary if and only if it corresponds to a unitary matrix  $U \in \mathbb{C}^{d \times d}$  with  $d$  the dimension of  $\mathcal{H}_B$ .*

Because every unitary matrix  $U$  has an inverse  $U^\dagger$  it defines a bijective or *reversible* transformation. The classical unitary transformations are a subset of the unitary transformations and are described by the unitary matrices  $U$  with  $[U]_{ij} \in \{0, 1\}$  for every  $i, j$ . See the appendix for the definition of a unitary transformation on an infinite dimensional state space.

## 1.6 Expanding the State Space

Consider two state spaces  $\mathcal{H}_V$  and  $\mathcal{H}_W$  spanned by  $V = \{v_1, \dots, v_n\}$  and  $W = \{w_1, \dots, w_m\}$ . If we join these spaces we get an expanded space which will be expressed as the *tensor product*  $\otimes$  of  $\mathcal{H}_V$  and  $\mathcal{H}_W$ . This gives us a new state space  $\mathcal{H}_Z$  with dimension  $nm$ :

$$\mathcal{H}_Z = \mathcal{H}_{(V \times W)} = \mathcal{H}_V \otimes \mathcal{H}_W$$

This space is spanned by the basis vectors  $z_{ij} = v_i \otimes w_j$ . In the bracket notation this is usually denoted as:

$$|v_i\rangle \otimes |w_j\rangle = |v_i, w_j\rangle = |z_{ij}\rangle$$

for every  $v_i \in V$ ,  $w_j \in W$  and  $z_{ij} \in Z$ . The tensor product  $\otimes$  is a linear function which determines all the vectors of  $\mathcal{H}_Z$  by:

$$\begin{aligned} \text{If } |X\rangle &= \sum_{i=1}^n \alpha_i |v_i\rangle \quad \text{and} \quad |Y\rangle = \sum_{j=1}^m \beta_j |w_j\rangle \\ \text{then } |X\rangle \otimes |Y\rangle &= \sum_{i,j=1}^{n,m} \alpha_i \beta_j |v_i, w_j\rangle \end{aligned}$$

In vector notation this can be visualized as:

$$|X\rangle \otimes |Y\rangle = \begin{pmatrix} \alpha_1 |Y\rangle \\ \alpha_2 |Y\rangle \\ \vdots \\ \alpha_n |Y\rangle \end{pmatrix} = \begin{pmatrix} \alpha_1 \beta_1 \\ \alpha_1 \beta_2 \\ \vdots \\ \alpha_2 \beta_1 \\ \vdots \\ \alpha_n \beta_m \end{pmatrix} \in \mathbb{C}^{nm}$$

If the states  $|X\rangle$  and  $|Y\rangle$  are known states, we will denote the joint state  $|X\rangle \otimes |Y\rangle$  also by  $|X, Y\rangle$ . This is called a *product state*. (Because  $X$  and  $Y$  are just labels this is not a mathematical rule, just a notational custom.) The set of product states of a state space  $\mathcal{H}_V \otimes \mathcal{H}_W = \mathcal{H}_Z$  is a proper subset of  $\mathcal{H}_Z$ . The states which are not product states are called *entangled states* because there is correlation between the  $V$ -part and the  $W$ -part of the vector.

If a state space  $\mathcal{H}_Z$  can be decomposed into two (or more) spaces  $\mathcal{H}_V$  and  $\mathcal{H}_W$  such that  $\mathcal{H}_Z = \mathcal{H}_V \otimes \mathcal{H}_W$ , we refer to  $\mathcal{H}_Z$  as a product space. Now  $V$  and  $W$  define the ‘subspaces’ or ‘subsystems’ of  $\mathcal{H}_Z$ . We will encounter these subsystems later on, when we are discussing the notion of entanglement.

If  $P$  is an operator on  $\mathcal{H}_V$  and  $Q$  is an operator on  $\mathcal{H}_W$ , then their combined behavior in  $\mathcal{H}_Z$  is also a tensor product:

$$P|X\rangle \otimes Q|Y\rangle = (P \otimes Q)|X, Y\rangle$$

Because  $P$  and  $Q$  are matrices, the tensor product  $P \otimes Q$  can be calculated by:

$$P \otimes Q = \begin{pmatrix} p_{11}Q & p_{12}Q & \cdots & p_{1n}Q \\ p_{21}Q & p_{22}Q & \cdots & p_{2n}Q \\ \vdots & \vdots & \ddots & \vdots \\ p_{n1}Q & p_{n2}Q & \cdots & p_{nn}Q \end{pmatrix} \in \mathbb{C}^{nm \times nm}$$

The most important properties of the tensor product are (assuming that the dimensions of  $P, Q, R$  and  $S$  are compatible):

$$\begin{aligned} \text{If } \mu \in \mathbb{C} \text{ then } (\mu P) \otimes Q &= P \otimes (\mu Q) = \mu(P \otimes Q) & P \otimes (Q \otimes R) &= (P \otimes Q) \otimes R \\ (P + Q) \otimes R &= P \otimes R + Q \otimes R & (P \otimes Q)(R \otimes S) &= (PR \otimes QS) \\ P \otimes (Q + R) &= P \otimes Q + P \otimes R & (P \otimes Q)^\dagger &= (P^\dagger \otimes Q^\dagger) \end{aligned}$$

Tensor products for matrices are also called “direct products” or “Kronecker products” [25, 33, 47]. The tensor product should not be confused with the Cartesian product “ $\times$ ”. A tensor product of two spaces  $\mathcal{H}_A$  and  $\mathcal{H}_B$  is *induced* by the Cartesian product of the two basis sets  $A$  and  $B$ .

### 1.6.1 Notational Customs

The following notational customs will be used in this paper. When necessary, a brief explanation will be given.

**repeated tensor products** By  $P^{[n]}$  (with  $n \in \mathbb{N}^+$ ) we mean  $P \otimes P \otimes \cdots \otimes P$  ( $n$  times) and by definition  $P^{[0]} = 1 \in \mathbb{C}$ . Because a tensor product does not commute, we have to define explicitly:

$$\bigotimes_{i \in \mathbb{Z}_r} P_i = \bigotimes_{i=0}^{i=r-1} P_i = P_0 \otimes P_1 \otimes \cdots \otimes P_{r-1}$$

for  $r \in \mathbb{N}^+$ , and because  $P^{[0]} = 1$  also:

$$\bigotimes_{i=r}^{r-1} P_i = 1 \in \mathbb{C}$$

**state space** If we use the basis set  $B$ , the corresponding state space will be denoted by  $\mathcal{H}_B$ .

**basis set/states** The set  $B$  is will be treated as a subset of  $\mathcal{H}_Q$ . The basis states in  $B$  will also be named *canonical states*.

**ket notation** For every basis state  $\xi \in B$  we will write  $|\xi\rangle$  when  $\xi$  is used in the context of state space  $\mathcal{H}_B$ .

**long ket notation** Because  $|x\rangle \otimes |y\rangle$  implies that the  $x$  and  $y$  part are not entangled, it is erroneous to state  $|x, y\rangle \equiv |x\rangle \otimes |y\rangle$ . In general –if entanglement is involved– we have to describe the state by one large ket. To avoid cumbersome expressions, the following notation is introduced:

$$|x_a, \dots, x_b\rangle = \left| [x_i]_{i=a}^b \right\rangle = |x_{a:b}\rangle$$

## 1.7 The Effects of Measurement

A quantum mechanical system is fully described by its state vector. This state however, cannot be measured directly. That is, we cannot observe the amplitudes of the different base vectors. Only some specific information of this state can be observed. These *observables* are characterized in the following way.

Consider a state space  $\mathcal{H}_Z$  with a state  $|\Psi\rangle \in \mathcal{H}_Z$ . An observable corresponds to a set of subspaces  $\mathcal{H}_{W_1}, \mathcal{H}_{W_2}, \dots, \mathcal{H}_{W_k} \subseteq \mathcal{H}_Z$  with:

- for every  $i, j$ , if  $i \neq j$  then  $\mathcal{H}_{W_i} \perp \mathcal{H}_{W_j}$
- for every state  $|\Psi\rangle \in \mathcal{H}_Z$ :  $|\Psi\rangle = \sum_{i=1}^k \alpha_i |\psi_i\rangle$  with  $|\psi_i\rangle \in \mathcal{H}_{W_i}$  for every  $i$

An observation along these subspaces will give us a result ‘ $j$ ’ which corresponds to a single subspace  $\mathcal{H}_{W_j}$ . This obeys the following rules:

- The probability of measuring  $j$  equals  $|\alpha_j|^2$ .
- If  $j$  is measured, the state *collapses* according to  $|\Psi\rangle \rightsquigarrow |\psi_j\rangle$  (the new state vector is normalized again)

Because of the second rule all the information about the amplitudes  $\alpha$  is lost. This shows that measuring a quantum-mechanical system is in general an irreversible process because the superposition is disturbed (hence the “ $\rightsquigarrow$ ” symbol).

**EXAMPLE 1.2** Take a two spin system  $\mathcal{H}_Z$  with  $Z = \{|\uparrow\uparrow\rangle, |\uparrow\downarrow\rangle, |\downarrow\uparrow\rangle, |\downarrow\downarrow\rangle\}$  with a state  $|\Psi\rangle$  in the superposition:  $|\Psi\rangle = \frac{1}{3}|\uparrow\uparrow\rangle + \frac{2}{3}|\uparrow\downarrow\rangle + \frac{2}{3}|\downarrow\uparrow\rangle$ . A measurement on the first spin divides  $\mathcal{H}_Z$  into the two subspaces  $\mathcal{H}_{W_1}$  and  $\mathcal{H}_{W_2}$ , with  $W_1 = \{|\uparrow\uparrow\rangle, |\uparrow\downarrow\rangle\}$  and  $W_2 = \{|\downarrow\uparrow\rangle, |\downarrow\downarrow\rangle\}$ . The  $W_1$  space corresponds to measuring “up” and  $W_2$  with measuring “down”.

Along these subspaces, the state vector is described by:

$$|\Psi\rangle = \frac{\sqrt{5}}{3} \left\{ \frac{1}{\sqrt{5}}|\uparrow\uparrow\rangle + \frac{2}{\sqrt{5}}|\uparrow\downarrow\rangle \right\} + \frac{2}{3} \{|\downarrow\uparrow\rangle\} = \frac{\sqrt{5}}{3}|\psi_1\rangle + \frac{2}{3}|\psi_2\rangle$$

with  $|\psi_1\rangle \in \mathcal{H}_{W_1}$  and  $|\psi_2\rangle \in \mathcal{H}_{W_2}$ . As a result, the outcome of the measurement has two possibilities:

1. first spin is “up” with probability  $\frac{5}{9}$ ; the new state vector becomes:  $|\Psi'\rangle = \frac{1}{5}\sqrt{5}\{|\uparrow\uparrow\rangle + 2|\uparrow\downarrow\rangle\}$
2. first spin is “down” with probability  $\frac{4}{9}$ ; the new state vector becomes:  $|\Psi'\rangle = |\downarrow\downarrow\rangle$

◇

This example shows that if a state  $|X\rangle \in \mathcal{H}_A \otimes \mathcal{H}_B$  has entanglement between the two subsystems, an observation on the  $A$ -part will also influence the  $B$ -part of the state.

Our first lemma tells us how to calculate the amplitudes of a subsystem  $\mathcal{H}_Q$  in an expanded state space  $\mathcal{H}_{Q^n}$ .

**LEMMA 1.1** Given an expanded state space  $\mathcal{H}_{Q^n}$  whose states are described by  $|x_1, \dots, x_n\rangle \in \mathcal{H}_{Q^n}$ . For every  $X \in \mathcal{H}_{Q^n}$ , the probability amplitude to have  $|Y\rangle \in \mathcal{H}_Q$  on the  $x_1$  position of the state  $|X\rangle$  equals:

$$\langle x_1 = Y | X \rangle = \sum_{Z \in Q^{n-1}} \langle x_1 = Y, x_{2:n} = Z | X \rangle$$

PROOF. Let  $Q'$  be a basis set with  $\mathcal{H}_Q = \mathcal{H}_{Q'}$  and  $Y \in Q'$ . Because  $Q'$  is an orthogonal set, we have for every  $q \in Q'$ :

$$\langle x_1 = Y | x_1 = q, \dots \rangle = \begin{cases} 1 & \text{if } q = Y \\ 0 & \text{if } q \neq Y \end{cases}$$

The state space  $\mathcal{H}_{Q' \times Q^{n-1}}$  equals the state space  $\mathcal{H}_{Q^n}$  and therefore (by summation over the new basis set  $Q' \times Q^{n-1}$ ):

$$\begin{aligned} \langle x_1 = Y | X \rangle &= \sum_{q \in Q'} \sum_{Z \in Q^{n-1}} \langle x_1 = Y | x_1 = q, x_{2:n} = Z \rangle \langle x_1 = q, x_{2:n} = Z | X \rangle \\ &= \sum_{Z \in Q^{n-1}} \langle x_1 = Y, x_{2:n} = Z | X \rangle \end{aligned}$$

□



## Chapter 2

# Quantum Computing

In this chapter we will describe the basics of quantum computation. This is done by defining quantum bits, quantum registers and quantum gates. The model of quantum Turing-machines will also be mentioned.

## 2.1 Quantum Memory

### 2.1.1 Quantum Bits and Quantum Registers

Consider a two-state system  $B$  for which we have labeled the basis states “0” and “1”. The state set  $\mathcal{H}_B$  is therefore be described by the configurations

$$|X\rangle = \alpha_0|0\rangle + \alpha_1|1\rangle$$

for every  $\alpha_0, \alpha_1 \in \mathbb{C}$  with  $|\alpha_0|^2 + |\alpha_1|^2 = 1$ . Such a state  $|X\rangle$  will be called a *quantum-bit* or *qubit*. Because the canonical states  $|0\rangle$  and  $|1\rangle$  will have their natural meaning, this systems is usually indicated by  $\mathcal{H}_{\{0,1\}}$ .

If we expand this state space to that of a system whose basis set is described by  $\{0,1\}^n$  we get the definition of a  $n$ -qubit system. The possible configurations of such a *quantum register* are covered by the expressions

$$|X\rangle = \sum_{\xi \in \{0,1\}^n} \alpha_\xi |\xi\rangle$$

which obey the normalization restriction. The state space of an  $n$ -qubit system equals the tensor product of  $n$  separate qubit systems:

$$\mathcal{H}_{\{0,1\}^n} = \underbrace{\mathcal{H}_{\{0,1\}} \otimes \mathcal{H}_{\{0,1\}} \otimes \cdots \otimes \mathcal{H}_{\{0,1\}}}_{n \text{ times}} = \mathcal{H}_{\{0,1\}}^{[n]}$$

Qubits and quantum registers are used to describe the memory of quantum computers. The canonical configurations of an  $n$ -qubit system are the ‘classical configurations’  $\{0,1\}^n$ . The canonical configurations are a basis set for the system  $\mathcal{H}_{\{0,1\}^n}$ .

### 2.1.2 Entanglement

When the amplitudes of an  $n$ -qubit configuration define a correlation between the individual qubits, we say that the qubits are *entangled*. This correlation can effect the outcome of measurements performed on the system in a way that can not be understood in a classical sense.

**EXAMPLE 2.1** Take the following configuration of a two-qubit system:

$$|\Psi\rangle = \frac{1}{\sqrt{2}}\sqrt{2}\{|0,0\rangle + |1,1\rangle\}$$

When we measure the value of the first qubit, the wave function will collapse to one of the two possible states:

$$|\Psi_0\rangle = |0,0\rangle \quad \text{or} \quad |\Psi_1\rangle = |1,1\rangle$$

This means that the value of the second bit is determined by the outcome of our measurement on the first bit: there is a correlation between the bits. This is not always the case. If the initial two-qubit system is a tensor product of two individual qubits, then a measurement on the first bit does not affect the second bit. For example:

$$|\Psi\rangle = \frac{1}{\sqrt{2}}\{|0,0\rangle + |0,1\rangle + |1,0\rangle + |1,1\rangle\} = \frac{1}{\sqrt{2}}\{|0_1\rangle + |1_1\rangle\} \otimes \{|0_2\rangle + |1_2\rangle\}$$

An observation on the first bit changes the system into one of the configurations:

$$|\Psi_0\rangle = |0_1\rangle \otimes \frac{1}{\sqrt{2}}\sqrt{2}\{|0_2\rangle + |1_2\rangle\} \quad \text{or} \quad |\Psi_1\rangle = |1_1\rangle \otimes \frac{1}{\sqrt{2}}\sqrt{2}\{|0_2\rangle + |1_2\rangle\}$$

which does not affect the second bit. ◇

The notion of entanglement is important because it illustrates the influence of a measurement on the system and is typical for quantum-mechanical systems. It also enables the occurrence of *interference* in a quantum computer which defines the fundamental difference between probabilistic and quantum computing.

Entanglement can not always be seen by the probability of measuring certain basis states. As a counterexample: the two qubits in the following state are entangled:

$$|\Psi\rangle = \frac{1}{\sqrt{2}}\{|0,0\rangle + |0,1\rangle + |1,0\rangle - |1,1\rangle\}$$

although there is no ‘correlation’ to be found when measuring the bits.

## 2.2 Quantum Gates

We will now look at gates which operate on qubits: *quantum gates*. In the previous chapter we discussed unitary transformations operating on a state space. A quantum gate is a system that performs such a proper transformation on a register of qubits. Every proper quantum gate that operates on a  $n$ -qubit system is therefore described by a unitary matrix  $M \in \mathbb{C}^{d \times d}$  (with  $d = 2^n$  the dimension of the state space  $\mathcal{H}_{\{0,1\}^n}$ ). An example of a gate that operates on one qubit is the NOT-gate.

**EXAMPLE 2.2** The NOT-gate is described by the unitary matrix which operates on the two dimensional state space  $\mathcal{H}_{\{0,1\}}$  with

$$\begin{pmatrix} 1 \\ 0 \end{pmatrix} \equiv |0\rangle \quad \text{and} \quad \begin{pmatrix} 0 \\ 1 \end{pmatrix} \equiv |1\rangle$$

The matrix  $M_{\text{NOT}}$  therefore equals

$$M_{\text{NOT}} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

The ‘traditional’ behavior  $\text{NOT}(0) = 1$  and  $\text{NOT}(1) = 0$  is shown by the matrix multiplications:

$$M_{\text{NOT}}|0\rangle = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} = |1\rangle$$

and

$$M_{\text{NOT}}|1\rangle = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} = |0\rangle$$

In general we have for this gate:

$$M_{\text{NOT}}\{\alpha_0|0\rangle + \alpha_1|1\rangle\} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} \alpha_0 \\ \alpha_1 \end{pmatrix} = \begin{pmatrix} \alpha_1 \\ \alpha_0 \end{pmatrix} = \alpha_1|0\rangle + \alpha_0|1\rangle$$

with  $\alpha_0, \alpha_1 \in \mathbb{C}$  and  $|\alpha_0|^2 + |\alpha_1|^2 = 1$ . ◇

The last equation in this example shows that a quantum gate has to respect the possible superposition of a quantum register. This enables us to define gates whose behavior is impossible with classical gates. An example of such a true quantum gate is the following:

**EXAMPLE 2.3** Define a one-qubit gate  $Q$  by:

$$Q = \frac{1}{2} \begin{pmatrix} 1-i & 1+i \\ 1+i & 1-i \end{pmatrix}$$

When applying this gate to the canonical states  $|0\rangle$  and  $|1\rangle$ , we get: ( $\lambda = \frac{1}{2} - \frac{i}{2}$ ;  $|\lambda|^2 = \frac{1}{2}$ )

$$Q|0\rangle = \lambda|0\rangle + \lambda^*|1\rangle \quad \text{and} \quad Q|1\rangle = \lambda^*|0\rangle + \lambda|1\rangle$$

In both cases the result is a perfect ‘fifty-fifty’ mixture of  $|0\rangle$  and  $|1\rangle$  (there is only a phase difference between  $Q|0\rangle$  and  $Q|1\rangle$ ). If we apply the  $Q$ -gate a second time on this result we get:

$$Q\{Q|0\rangle\} = |1\rangle \quad \text{and} \quad Q\{Q|1\rangle\} = |0\rangle$$

which is the NOT-function again. This is affirmed by the matrix of  $Q^2$ :

$$Q^2 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

We therefore say that  $Q$  is the ‘square root of NOT’. Although the input and output of  $Q^2$  ‘classical’ it is not possible to have a classical gate  $Q$  with the same behavior. This is proven by:  $\det(Q)^2 = \det(\text{NOT}) = -1$  and thus  $\det(Q) \in \{i, -i\}$  which is impossible for a classical gate. ◇

The reversible gates with classical behavior are a proper subset of all possible quantum gates. Non-reversible gates are not proper because the corresponding matrix has to be an invertible matrix ( $M \cdot M^\dagger = 1$ ).

### 2.2.1 Proper Quantum Gates

It is not always obvious if a gate is well-formed or not. We therefore will use the following ‘rule of thumb’ to certify that a gate with a certain *requested behavior* is proper.

**DEFINITION 2.1 (Requested behavior)** The requested behavior of a gate  $M$  is described by a finite list of input and output values  $\langle (x_1, y_1), \dots, (x_k, y_k) \rangle$  such that  $M(x_i) = y_i$  for every  $i$ .

This definition is used in the next lemma about proper quantum gates.

**LEMMA 2.1** *If the requested behavior of a gate  $M : \mathcal{H}_{B^n} \rightarrow \mathcal{H}_{B^n}$  is described by a list*

$$\langle (x_1, y_1), \dots, (x_k, y_k) \rangle$$

*with  $x_i \perp x_j$  and  $y_i \perp y_j$  for every  $i \neq j$ , then there exists a proper quantum gate with  $M(x_i) = y_i$  for every  $1 \leq i \leq k$ .*

PROOF. Let the set of basis states be denoted by  $B^n = \{\xi_1, \dots\}$ . Because  $x_1, \dots, x_k$  are mutually orthonormal, there exists a unitary transformation  $P : \mathcal{H}_{B^n} \rightarrow \mathcal{H}_{B^n}$  with  $P(\xi_i) = x_i$ , for every  $1 \leq i \leq k$ . For the same reason there also exists a unitary  $R$  such that  $R(\xi_i) = y_i$ . If we define  $M = R \cdot P^\dagger$ ,  $M$  will also be unitary, with for every  $1 \leq i \leq k$ :  $M(x_i) = R(P^\dagger(x_i)) = R(\xi_i) = y_i$ .  $\square$

## 2.3 Quantum Gate Circuits

An acyclic circuit of quantum gates is called a *quantum (gate) circuit* or *quantum gate array*. The general behavior of a quantum circuit can be calculated with the matrix formalism in a straightforward way.

Consider two gates  $A$  and  $B$  that operate on two disjoint subsystems of  $n$  and  $m$  bits:

$$A : \mathcal{H}_{\{0,1\}^n} \rightarrow \mathcal{H}_{\{0,1\}^n} \quad \text{and} \quad B : \mathcal{H}_{\{0,1\}^m} \rightarrow \mathcal{H}_{\{0,1\}^m}$$

The combination of  $A$  and  $B$  defines a transformation on the state space  $\mathcal{H}_{\{0,1\}^n} \otimes \mathcal{H}_{\{0,1\}^m} = \mathcal{H}_{\{0,1\}^{n+m}}$  described by the tensor product  $A \otimes B$

$$(A \otimes B) : \mathcal{H}_{\{0,1\}^{n+m}} \longrightarrow \mathcal{H}_{\{0,1\}^{n+m}}$$

with

$$(A \otimes B)|X, Y\rangle \longmapsto A|X\rangle \otimes B|Y\rangle$$

for every  $X \in \{0, 1\}^n$  and  $Y \in \{0, 1\}^m$ .

When two layers of gates act as a sequence on an  $n$ -qubit system, the general transformation is calculated by the product of the two defining matrices. This rule was already used in the example of the square root of NOT gate. To summarize:

1. When two gates are sequential, multiply the matrices
2. When two gates are parallel, take the tensor product of the matrices.

See Figure 2.1 for an example of a small quantum circuit. The characteristics of the quantum circuit model were first investigated in articles by David Deutsch [16] and Andrew Chi-Chih Yao [60].

### 2.3.1 A Universal Quantum Gate

Because quantum arrays define unitary transformations on qubit systems, it is natural to ask “What kind of quantum gates do we need to implement an arbitrary unitary transformation?” For classical non-reversible computation we know that it is sufficient to have AND, OR and NOT-gates. The quantum case is a more difficult problem to solve.

It is impossible to construct every unitary transformation *exactly* with a finite set of basic gates because the number of possible transformations is uncountable. We therefore have to approximate the transformation within a bounded error. This means that given a transformation  $U$  and an arbitrary small number  $\varepsilon > 0$  we can make a circuit  $U'$  which simulates  $U$  within the allowed error  $\varepsilon$ , that is  $\|U - U'\| \leq \varepsilon$ . If there exists a quantum gate by which we can construct any transformation within a bounded error, we will call this gate a *universal gate*.

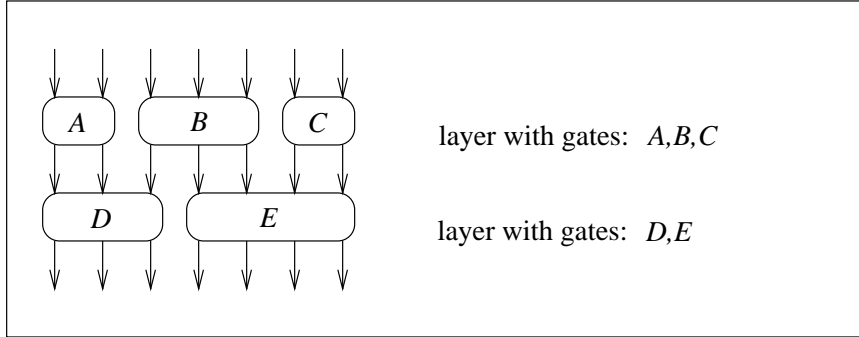


Figure 2.1: A small quantum gate circuit with two layers and five gates. The action of the first layer is described by the tensor product  $A \otimes B \otimes C$ . The second layer equals:  $D \otimes E$ . The transformation defined by this circuit is therefore described by:  $(D \otimes E) \cdot (A \otimes B \otimes C)$ .

It was proven by David Deutsch that there exists a universal gate that operates on three bits [16]. After that, Adriano Barenco[3] and David DiVincenzo[18] showed the existence of a two bit gate which is universal in its computational power. For a detailed expose of this subject the reader is best referred to the famous ‘article by nine authors’ by Adriano Barenco *et al.* The quantum circuit model has proven to be the most convenient model to study quantum computing.

## 2.4 Quantum Turing Machines

The quantum Turing machine provides us with an alternative model for quantum computing. There is strong relationship between the characteristics of such QTMs and reversible Turing machines [5, 15]. The following definition has its origin by Ethan Bernstein and Umesh Vazirani [6].

**DEFINITION 2.2 (Quantum Turing machine)** A Quantum Turing Machine  $M$  is defined by the tuple  $\langle K, \Sigma, \delta \rangle$ , with  $\Sigma$  a finite alphabet set (including a blank symbol  $\square$ ),  $K$  a finite internal state set and  $\delta : K \times \Sigma \times \{\leftarrow, \rightarrow\} \times K \times \Sigma \rightarrow \mathbb{C}$  the finite state control.

The set of basis states is described by the states  $|\Psi\rangle = |k, h, \psi\rangle \in \ell_2(K \times \mathbb{N} \times Q^*) = \mathcal{S}$ , where  $k$  indicates the internal state of the QTM;  $h$  defines the head-position on the one-way infinite tape and  $\psi$  describes the tape-content (with only a finite number of non-blank symbols). The global transition function  $M : \mathcal{S} \rightarrow \mathcal{S}$  now obeys (for every  $k, h, \psi, q, h', k'$  and  $\psi'$ ):

$$\begin{aligned} \langle k', h-1, \psi_{0:h-1} \times q \times \psi_{h+1:\dots} | M | k, h, \psi \rangle &= \delta(k, \psi_h, \leftarrow, q, k') \\ \langle k', h+1, \psi_{0:h-1} \times q \times \psi_{h+1:\dots} | M | k, h, \psi \rangle &= \delta(k, \psi_h, \rightarrow, q, k') \\ \langle k', h', \psi' | M | k, h, \psi \rangle &= 0 \quad \text{otherwise} \end{aligned}$$

(Note that by each computational step the head *must* move.) By superposition on the set of basis states,  $M$  describes a transformation on the Hilbertian space  $\mathcal{S}$ . If  $M$  describes a *unitary* transformation, the QTM  $M$  is said to be *well formed* or *proper*.

**NOTATION 2.1** We will use the following notational conventions:  $(|X\rangle, |Y\rangle) \in \mathcal{S}$  and  $M$  a well formed QTM):

- Instead of  $|k, h, \psi\rangle$  we may also write  $|k, \psi_0, \dots, \psi_{h-1}, \underline{\psi}_h, \psi_{h+1}, \dots\rangle$ .
- By  $|X\rangle \xrightarrow{M} |Y\rangle$  we mean:  $M|X\rangle = |Y\rangle$
- If there exists a  $t \in \mathbb{N}$  with  $M^t|X\rangle = |Y\rangle$  we denote this by  $|X\rangle \xrightarrow{*M} |Y\rangle$

- The internal state is often described by a function-value  $\lceil \dots \rceil$ . For example, with  $\lceil \text{start} \rceil$  and  $\lceil \text{halt} \rceil$  we mean the starting and halting state  $\in K$  of the QTM. This is done without further specification.

### 2.4.1 Well Formed QTMs

Because QTMs will not play an important role in this paper, we will only summarize the main characteristics of quantum Turing machines. First of all we have to take a closer look at the well formedness issue. To handle this restriction on the *infinite* dimensional Hilbertian space  $\mathcal{S}$ , Ethan Bernstein and Umesh Vazirani translated it into the following three constraints.

**LEMMA 2.2** *A quantum Turing machine is well formed if and only if the finite state control  $\delta$  obeys the following local conditions (where  $\sum_{k,h,\psi} |k, h, \psi\rangle$  describes the summation on the basis states of  $\mathcal{S}$ ):*

1. For every basis state  $X$ , the transformation  $M$  is norm preserving:

$$\sum_{k,h,\psi} |\langle k, h, \psi | M | X \rangle|^2 = 1$$

2. If  $X$  and  $Y$  are different basis states with identical head positions, then  $M(X)$  and  $M(Y)$  are orthogonal:

$$\sum_{k,h,\psi} \langle Y | M | k, h, \psi \rangle \langle k, h, \psi | M | X \rangle = 0$$

3. For every basis state  $X$  with head position  $h$  the ‘ $\leftarrow$ ’ and ‘ $\rightarrow$ ’-part are mutual orthogonal:

$$\sum_{k,\psi} \langle k, h+1, \psi | M | X \rangle \langle X | M | k, h-1, \psi \rangle = 0$$

PROOF. See the original paper [6]. □

A QTM with a deterministic head position always satisfies the third constraint. We will use this in the following lemma which shows us that any finite dimensional unitary transformation can be simulated by a quantum Turing machine.

**LEMMA 2.3** *For every unitary transformation  $M$  on  $\mathcal{H}_{\{0,1\}^n}$  there exists a well formed QTM  $T$  such that for every  $|\psi\rangle \in \mathcal{H}_{\{0,1\}^n}$  we have:*

$$|\lceil \text{start} \rceil, 0\rangle \otimes |\psi\rangle \xrightarrow{*}_T |\lceil \text{halt} \rceil, 0\rangle \otimes M|\psi\rangle$$

PROOF. We will restrict the definition to the values  $\psi \in \{0, 1\}^n$  (by superposition this will impose the desired transformation on the whole state space  $\mathcal{H}_{\{0,1\}^n}$ ). Because there is only a finite number ( $2^n$ ) of possibilities, we will use an exhaustive look-up table. First we will read the value from left to right (thereby simultaneously erasing the tape contents). After this part, the QTM ‘splits’ into the desired amplitudes and writes the various basis states on the tape. After a small ‘shuffle’ (the head is not allowed to stay stationary), the machine halts. With  $m_{\xi\psi} = \langle \xi | M | \psi \rangle$  this is formally described by the sequence:

$$|\lceil \text{start} \rceil, 0\rangle \otimes |\psi\rangle \longrightarrow_T |\lceil \psi_0 \rceil, 1\rangle \otimes |0, \psi_{1:n-1}\rangle$$

First we *replace* the  $n$  bits of the input string  $\psi$  to a corresponding state of the QTM ...

$$\xrightarrow{*}_T |\lceil \psi \rceil, n\rangle \otimes |0, \dots, 0\rangle$$

after which the output value is calculated by a finite look-up table. This produces a superposition on the internal state of the QTM with amplitudes  $m$ .

$$\longrightarrow_T \sum_{\xi \in \{0,1\}^n} m_{\xi\psi} |\ulcorner \psi, \xi \urcorner, n-1 \rangle \otimes |0, \dots, 0\rangle$$

By moving the head backwards, this output string is written on the tape of the Turing machine.

$$\longrightarrow_T \sum_{\xi \in \{0,1\}^n} m_{\xi\psi} |\ulcorner \psi, \xi_{0:n-2} \urcorner, n-2 \rangle \otimes |0, \dots, 0, \xi_{n-1}\rangle$$

By doing so, the superposition of the machine  $T$  is transported to a superposition on the tape,

$$\xrightarrow{*}_T \sum_{\xi \in \{0,1\}^n} m_{\xi\psi} |\ulcorner \psi, \xi_0 \urcorner, 0 \rangle \otimes |0, \xi_{1:n-1}\rangle$$

Finally, the QTM is returns to one basis state ‘halting ... ’ which shuffles the head one last time.

$$\begin{aligned} &\longrightarrow_T \sum_{\xi \in \{0,1\}^n} |\ulcorner \text{halting } \dots \urcorner, 1 \rangle \otimes m_{\xi\psi} |\xi\rangle \\ &\longrightarrow_T |\ulcorner \text{halt} \urcorner, 0 \rangle \otimes M|\psi\rangle \end{aligned}$$

With lemma 2.2 and the unitarity of  $M$  we can verify the well formedness of this QTM. Because  $M$  is norm preserving, the  $m$  values will also be proper (requirement 1). The  $\ulcorner \dots \urcorner$ -function will be defined in such a way that  $\psi \neq \psi'$  implies  $|\ulcorner \psi \urcorner\rangle \perp |\ulcorner \psi' \urcorner\rangle$ . The transformation  $M$  is angle preserving, and therefore  $M|\psi\rangle \perp M|\psi'\rangle$  which certifies the second restriction. Because this QTM has deterministic head position, the third requirement is trivial.  $\square$

It was shown by Ethan Bernstein and Umesh Vazirani[6] that there exists a Universal QTM that can simulate any other QTM with only polynomial slowdown. The equivalence of the quantum circuit model and QTM-model was proven by Andrew Chi-Chih Yao[60].

## 2.5 Algorithms for Quantum Computers

What are quantum computers good for? When we want to simulate a quantum mechanical system a quantum computer could be very useful. On a classical computer we would have to compute the evolution for every basis state after which the final result is obtained by a summation of the calculated amplitudes for each basis state. For an  $n$  dimensional state space this calculation has a time complexity proportional to  $n$ . Because the dimension of the state space grows exponentially in the size of the system, this is a time consuming procedure. With a quantum computer this problem of large state spaces is solved by allowing the quantum computer to enter an equally sized superposition. By doing this we do not have to do the calculation for every basis state because the computer does this in parallel. This ‘quantum parallelism’ is also the main idea behind the algorithms that have been constructed to solve traditional computational problems in a faster way than is possible (or known to be possible) on a classical computer.

**EXAMPLE 2.4** Assume a function  $f : \{0,1\}^n \rightarrow \{0,1\}$  and a quantum algorithm  $M$  with

$$M|\xi, 0\rangle = |\xi, f(\xi)\rangle$$

for every  $\xi \in \{0,1\}^n$ . If we provide this algorithm with an equally distributed superposition of input states, we can calculate the superposition of outputs with the same time and space complexity by:

$$M \left( \sum_{\xi \in \{0,1\}^n} 2^{-n/2} |\xi, 0\rangle \right) = \sum_{\xi \in \{0,1\}^n} 2^{-n/2} |\xi, f(\xi)\rangle$$

*A measurement on this superposition gives us one of the basis states  $|\xi, f(\xi)\rangle$  with a probability of  $2^{-n}$ . Because we cannot control the probabilities of a measurement it would be misleading to say that we really ‘know’ all the possible outcomes of a function. In general it is impossible to force the superposition into a basis state with  $f(\xi) = 1$ . In order to take advantage of this superposition we have to use the interference phenomenon on the amplitudes of the basis states [7, 22, 30, 51]. If this is possible and how to do this depends on the function  $f$  we have calculated.  $\diamond$*

In 1994 Peter Shor showed how a quantum computer could be used to factor numbers with only polynomial time complexity [49, 50]. Not only does this algorithm decide if a number is prime or not but it will also give the prime factors of a composite number. It is generally believed but not known that on a classical computer this problem requires an exponential amount of time to solve. Moreover, because of this intractability, factorization is commonly used in modern encryption schemes such as RSA.

Another result was achieved more recently by Lov Grover [26, 9]. He proved that for any function  $f : \mathbb{Z}_n \rightarrow \{0, 1\}$  it is possible to find a number  $i$  with  $f(i) = 1$  (if such a number exists) with a time complexity proportional to  $\sqrt{n}$ . This quantum searching algorithm does not assume any knowledge about the function  $f$ . Although this is not an exponential speed-up it is likely that every deterministic or probabilistic algorithm will have a time complexity proportional to  $n$  for this task.



## Chapter 3

# Classical Cellular Automata

In this section we will give a brief description of cellular automata and its characteristics. Special attention will be paid to the subclass of reversible cellular automata which play an important role in the theory of quantum cellular automata.

### 3.1 Cellular Automata

The model of cellular automata (CA) is used to describe the behavior of *discrete systems* with a *uniform* and *parallel space/time behavior*. Given a space  $S$  and a local state set  $Q$ , a CA will describe a function  $F_S : Q^S \rightarrow Q^S$  which defines the time-evolution of an initial configuration  $X \in Q^S$ . Consequently, the configuration at time  $t \in \mathbb{N}$  is described by the expression  $F^t(X)$ .

Before giving a formal definition we will first look at a typical example of a one-dimensional cellular automaton to make ourselves familiar with some important characteristics.

**EXAMPLE 3.1** Consider a one-dimensional, two-state CA  $F$  such that the successor  $X'$  of a configuration  $X \in \{0, 1\}^{\mathbb{Z}}$  is determined by a local function  $f$  according to:

$$\begin{aligned} X' &= F_{\mathbb{Z}}(X) \\ &= F_{\mathbb{Z}}\left(\bigotimes_{i \in \mathbb{Z}} X_i\right) \\ &= \bigotimes_{i \in \mathbb{Z}} f(X_{i-1}, X_i, X_{i+1}) \end{aligned}$$

This local function  $f : Q^3 \rightarrow Q$  is defined by a finite table:

$$\begin{array}{ll} f(0, 0, 0) = 0 & f(0, 0, 1) = 1 \\ f(0, 1, 0) = 0 & f(0, 1, 1) = 1 \\ f(1, 0, 1) = 0 & f(1, 0, 0) = 1 \\ f(1, 1, 1) = 0 & f(1, 1, 0) = 1 \end{array}$$

which implements the addition-modulo-two-rule:  $f(x_{-1}, x_0, x_1) = x_{-1} \oplus x_1$ .

If we unfold the time parameter  $t$  of the global function  $F_{\mathbb{Z}}^t$  as an additional space-dimension, we get a  $(1 + 1)$ -dimensional structure which shows us the space/time behavior  $\mathbb{Z} \times \mathbb{N} \rightarrow Q$  of the CA on an initial configuration  $X$ :

$$(i, t) \mapsto [F_{\mathbb{Z}}^t(X)]_i$$

Figure 3.1 shows us the result of this transformation. With the convention that time is going downwards each time-step can be identified as a layer in such a structure.

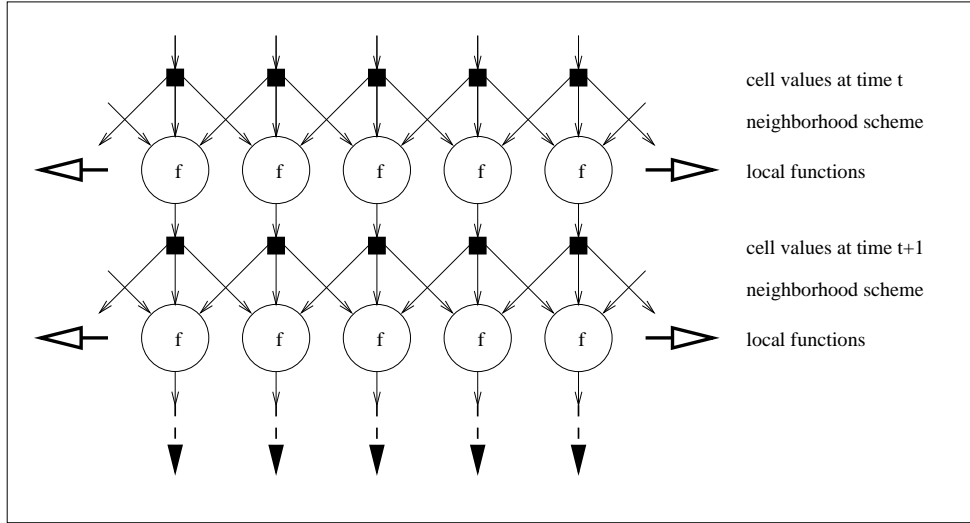


Figure 3.1: A part of the automaton in Example 3.1. The local states  $X_i$  at time  $t$  are the input values for the local functions  $f$  which determine the next configuration at time  $t + 1$ . The mapping between the *cell-values* and the *local functions* is described by the *neighborhood scheme* of the automaton.

To continue this example we take the ‘single seed’ initial configuration:  $X = \dots 00100 \dots$  and look at the time/space evolution of this CA. A calculation by hand shows that this will produce the triangle of Pascal modulo 2:

$$\begin{array}{rcl}
 X & = & \dots \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ \dots \\
 F(X) & = & \dots \ 0 \ 0 \ 0 \ 0 \ 0 \ 1 \ 0 \ 1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ \dots \\
 F^2(X) & = & \dots \ 0 \ 0 \ 0 \ 0 \ 1 \ 0 \ 0 \ 0 \ 1 \ 0 \ 0 \ 0 \ 0 \ 0 \ \dots \\
 F^3(X) & = & \dots \ 0 \ 0 \ 0 \ 1 \ 0 \ 1 \ 0 \ 1 \ 0 \ 1 \ 0 \ 0 \ 0 \ 0 \ \dots \\
 F^4(X) & = & \dots \ 0 \ 0 \ 1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1 \ 0 \ 0 \ 0 \ \dots \\
 F^5(X) & = & \dots \ 0 \ 1 \ 0 \ 1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 1 \ 0 \ 1 \ 0 \ 0 \ \dots \\
 F^6(X) & = & \dots \ 1 \ 0 \ 0 \ 0 \ 1 \ 0 \ 0 \ 0 \ 1 \ 0 \ 0 \ 0 \ 1 \ \dots \\
 & \dots & \dots \qquad \qquad \qquad \dots
 \end{array}$$

When computed on a larger scale (Figure 3.2) this reveals a fractal structure also known as ‘Sierpinski’s triangle’ [59].  $\diamond$

## 3.2 Characteristics of Cellular Automata

The above example mentioned some terms and characteristics of cellular automata which we will now make more explicit.

**discrete space:** The space  $S$  which determines the dimension and the size of the configurations is discrete. In this thesis we will concentrate on the one-dimensional case  $S = \mathbb{Z}$  (two-way infinite) and  $S = \mathbb{Z}_k$  (periodic boundaries with size  $k$ ).

**finite state set:** The state set  $Q$  is always finite and typical  $Q = \{0, 1\}$  which defines a binary CA.

**cell/cell-value:** Each space-position  $i \in S$  identifies a cell which contains a cell-value  $X_i \in Q$ .

**configuration:** A configuration  $X$  is the concatenation of all cell-values on the space  $S$  at time  $t$ , and therefore an element of  $Q^S$ .

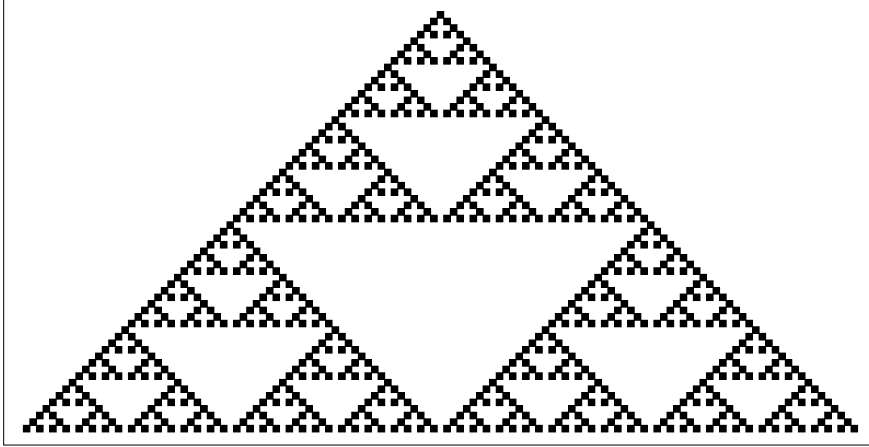


Figure 3.2: The space/time behavior of the one-dimensional cellular automaton in Example 3.1 on an initial configuration with only one cell non-zero. The cell-values 1 are colored black, the zero-values white; time is going downwards. This self-similar structure is also known as the Sierpinski-triangle and has a Hausdorff–Besicovitch or fractal dimension of  $\log(3)/\log(2) \approx 1.59$ .

**local function:** The time evolution of the different cells is described by a local function  $f$ . Like the space of a CA, the time-parameter is also considered discrete.

**neighborhood scheme:** The outcome of the local function at position  $i$  only depends on a finite set of cell-values in the vicinity of  $i$ . This set is defined by the neighborhood scheme  $N$  of the CA.

**global function:** Given a space  $S$  (compatible with the neighborhood scheme), the local function  $f$  imposes a global function  $F_S$  on the set of configurations. This global function determines the *space/time behavior* of the cellular automaton on an initial configuration.

The most stringent and typical characteristic of the CA-model is the restriction that the local function does not depend on the time  $t$  or the place  $i$ : a cellular automaton has *homogeneous* space/time behavior. It is for this reason that CA are sometimes referred to as *shift-dynamical* or *translation invariant* systems.

### 3.3 Formal definition of Cellular Automata

After this informal introduction to one dimensional cellular automata, we are now ready for a formal definition of this model.

**DEFINITION 3.1 (One dimensional CA)** A one-dimensional, classical cellular automaton is defined by the tuple  $\langle Q, N, f \rangle$  with  $Q$  the finite state set,  $N$  the neighborhood scheme and  $f$  the local function. The scheme  $N = \{N_1, \dots, N_{|N|}\}$  is a finite set of  $\mathbb{Z}$ -values with  $N_1 < N_2 < \dots < N_{|N|}$ .

For every appropriate space  $S$  this definition gives us a global function  $F_S : Q^S \rightarrow Q^S$  according to:

$$\begin{aligned} F_S(X) &= F_S \left( \bigotimes_{i \in S} X_i \right) \\ &= \bigotimes_{i \in S} f(X_{i+N_1}, X_{i+N_2}, \dots, X_{i+N_{|N|}}) \end{aligned}$$

To simplify the above notation we introduce the following short hands for the terms appearing in the local function.

**NOTATION 3.1** If  $N = (N_1, \dots, N_{|N|})$  is a neighborhood scheme it will be understood that:

$$(X_{i+N}) \equiv (X_{i+N_1}, X_{i+N_2}, \dots, X_{i+N_{|N|}})$$

and

$$(X_{a:b}) \equiv (X_a, X_{a+1}, \dots, X_b)$$

This leads us to the the brief and elementary expression:

$$F \left( \bigotimes_{i \in S} X_i \right) = \bigotimes_{i \in S} f(X_{i+N})$$

Notice that a CA  $F = \langle Q, N, f \rangle$  does not determine the space  $S$  it is supposed to act on.

**EXAMPLE 3.2** The CA of Example 3.1 is described by the tuple  $F = \langle \{0, 1\}, (-1, 0, 1), f \rangle$  with  $f(x, y, z) = x \oplus z$ .  $\diamond$

Because the definition of a neighborhood scheme  $N$  does not determine the space  $S$  it acts on, we can apply the same CA  $F = \langle Q, N, f \rangle$  on different structures compatible with the set  $\mathbb{Z}$ . Our main interest goes to the simple one-dimensional cases  $\mathbb{Z}$  and  $\mathbb{Z}_k$  which results in the functions  $F_{\mathbb{Z}} : Q^{\mathbb{Z}} \rightarrow Q^{\mathbb{Z}}$  and  $F_k : Q^k \rightarrow Q^k$  with  $k \in \mathbb{N}^+$ .

If the neighborhood  $N$  does not allow any interaction between the cell-values, the CA will have trivial behavior. Typically this is the case if  $|N| = 1$ . On the other hand it can be shown that every CA  $F$  can be redefined as a CA with  $N = \{0, 1\}$  without loss of generality. This will be proven in Lemma 6.5.

### 3.3.1 The Cellular Automata Model

Cellular automata were first used as such by John von Neumann [12, 46]. He showed the possibility of a universal two dimensional CA. Although not hard to prove, we will only refer the standard literature for a proof that one-dimensional CA can simulate Turing machines and are therefore computational universal. Because of their simple and homogeneous and parallel structure, cellular automata are frequently used to model physical systems such as gases, liquids et cetera. This, in combination with its use as a mathematical abstraction of parallel computation, makes it a unique combination of physics and theoretical computer science. The last decade there has been a growing interest in theory of cellular automata by computer scientists and physicists because of the possibility to simulate large CA with the use of modern computer technology. The articles by Stephen Wolfram [58, 59], Tommaso Toffoli [53, 23], Norman Margolus [54], and Howard Gutowitz [27] are a good starting point to investigate this modern use of cellular automata.

## 3.4 Reversible Cellular Automata

Given a global function  $F : Q^{\mathbb{Z}} \rightarrow Q^{\mathbb{Z}}$ , we say that the CA  $F$  is *global reversible* if and only if every configuration  $Y \in Q^{\mathbb{Z}}$  has *one predecessor*  $X$  such that:  $F(X) = Y$ . A simple reversible cellular automaton (RCA) is described by:  $k = 2$ ,  $N = \{0, 1\}$  and the local function  $f$  with:  $f(x_0, x_1) = x_0$  for every  $x_0, x_1 \in Q$ . This RCA is just the identity CA with  $F(X) = X$  for every  $X \in Q^{\mathbb{Z}}$ . By this example we see that it is possible to have a RCA with a non-reversible local function (which is always the case with  $k \geq 2$ ). More complex RCA are also possible but appear to be very rare among the set of plain cellular automata.

The characteristics of reversible cellular automata are extensively described in a review article by Tommaso Toffoli and Norman Margolus [27, 52]. For a long time it was unknown if there exists

a reversible cellular automata capable of embedding a general purpose computer. In 1976 however, Tommaso Toffoli proved that any  $d$ -dimensional cellular automaton could be simulated by a  $d + 1$ -dimensional reversible cellular automaton [52]. By the existence of a universal Turing machine embedded in a 1-dimensional cellular automaton, this proves that there exists a 2-dimensional reversible cellular automaton which is capable of doing the same. After that Morita and Harao proved the existence of a one dimensional RCA which is universal in its computational power [19, 44, 45]. They did this by defining a special subclass of CA which are reversible by definition: *partitioned cellular automata*. This type of CA will also be discussed in this thesis.

The ‘modern approach’ to cellular automata with the use of computer simulations seems to suggest that the old articles on this subject are outdated. Especially for the model of reversible automata this not the case. The theory of cellular automata as a computational model is a *mathematical theory*. To avoid ‘re-inventing the wheel’ one should also pay attention to the more abstract articles by Serafino Amoroso *et al.* [2], Gustav Hedlund [28], and Daniel Richardson [48].

## Chapter 4

# Quantum Cellular Automata

In this section we will define the model of one dimensional quantum cellular automata. This model is a straightforward extension of the classical model that exists for one dimensional CA.

### 4.1 The Size of the State Space

Before we describe the various models a note must be made on the size of the QCA we are considering. If we take a two way infinite automaton, the number of basis states will be infinite and even uncountable. Because we want to describe the general behavior by a transformation matrix this is highly impractical. We solve this problem by restricting ourselves to finite, (size  $k$ ) *circular bounded* automata (also called *periodic* QCA). This means that the set of basis states corresponds with the finite set of functions:  $\mathbb{Z}_k \rightarrow Q$  instead of the uncountable set of functions  $\mathbb{Z} \rightarrow Q$  ( $Q$  is the set of states for each cell).

This is somewhat different with the common approach to this problem. Most authors so far (John Watrous [57] and Christoph Dürr, Houg Lê Thanh and Miklos Santha [20, 21, 29]) define their QCA with a *quiescent* (non-active) state. Only configurations with a finite number of non-quiescent cells are then taken into account. From now on we will only look at one-dimensional structures.

### 4.2 Quantum Cellular Automata

With quantum cellular automata (QCA) we refer to the general description of a one dimensional quantum cellular automaton. It is a natural extension of the classical definition.

**DEFINITION 4.1 (One dimensional Quantum cellular automaton)** A QCA  $F$  is defined by the tuple:  $\langle Q, N, f \rangle$ , where:

- $Q$  describes the finite set of states
- $N \subset \mathbb{Z}$  defines the finite neighborhood scheme with  $N = \{N_0, \dots, N_{|N|}\}$  and  $N_0 < \dots < N_{|N|}$
- $f$  is the local function  $f : Q^N \rightarrow \mathcal{H}_Q$

In this last definition we see the quantum-mechanical aspect of the automaton. For every  $k \in \mathbb{N}^+$  this  $F$  gives us a transformation  $F_k : \ell_2(Q^k) \rightarrow \ell_2(Q^k)$  described by

$$\begin{aligned} F_k(X) &= F_k \left( \sum_{\xi \in Q^k} \alpha_\xi \cdot |\xi\rangle \right) \\ &= \sum_{\xi \in Q^k} \alpha_\xi \cdot F_k |\xi\rangle \end{aligned}$$

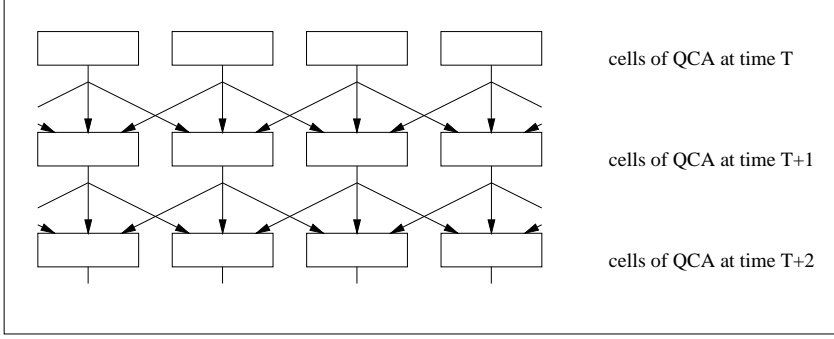


Figure 4.1: A typical one dimensional quantum cellular automaton with neighborhood scheme  $N = \{-1, 0, 1\}$ . Notice the identical structure when compared to the CA in Figure 3.1.

for every  $X \in \ell_2(Q^k)$  with  $\alpha_\xi \in \mathbb{C}$ . The values  $F_k|\xi\rangle$  are defined as a tensor product of the local values  $\xi_i$  of the basis state  $\xi$  computed by  $f$ :

$$\begin{aligned} F_k|\xi\rangle &= \bigotimes_{j \in \mathbb{Z}_k} f(\xi_{j+N_0}, \dots, \xi_{j+N_{|N|}}) \\ &= \bigotimes_{j \in \mathbb{Z}_k} f(\xi_{j+N}) \in (\mathcal{H}_Q)^k \end{aligned}$$

Note that the addition “ $j+N$ ” is in  $\mathbb{Z}_k$  and not in  $\mathbb{Z}$  and also that the range of  $F_k(Q^k)$  is a subset of the proper non-entangled states. The relation between the basis states, the proper non-entangled states, the proper states and all possible states in the Hilbertian space is described by:

$$Q^k \subsetneq (\mathcal{H}_Q)^k \subsetneq \mathcal{H}_{Q^k} \subsetneq \ell_2(Q^k)$$

Figure 4.1 is a picture of a QCA with neighborhood scheme  $N = \{-1, 0, 1\}$ .

### 4.2.1 Normalized QCA

The neighborhood set  $N$  defines the possible interaction between the cell values at at each computational step. Automata with large neighborhood sets allow more ‘internal communication’ than QCA with small ‘neighborhood sets’. Before proving that a neighborhood scheme of  $\{0, 1\}$  is sufficient to implement any possible QCA, we will first have to give the following definitions.

**DEFINITION 4.2 (Contiguous neighborhood scheme)** A QCA is *contiguous* if its neighborhood scheme is described by a closed interval on  $\mathbb{Z}$ :  $N = \{N_0, N_0 + 1, \dots, N_{|N|}\}$ .

**DEFINITION 4.3 (Normalized neighborhood scheme)** A QCA is *normalized* if its neighborhood scheme is a contiguous with  $N_0 = 0$ , therefore:  $N = \{0, 1, \dots, |N| - 1\}$ .

**DEFINITION 4.4 (Shift)** A *shift transformation*  $S$  on  $\mathcal{H}_{Q^k}$  is a transformation, defined by:

$$S(X) = S\left(\sum_{\xi \in Q^k} \alpha_\xi |\xi\rangle\right) = \sum_{\xi \in Q^k} \alpha_\xi \cdot S|\xi\rangle = \sum_{\xi \in Q^k} \alpha_\xi \cdot \left(\bigotimes_{j \in \mathbb{Z}_k} \xi_{j+1}\right)$$

for every  $X \in \mathcal{H}_{Q^k}$ .

We will often write  $S|x_0, \dots, x_{k-1}\rangle = |x_1, \dots, x_{k-1}, x_0\rangle$  to describe the behavior of the shift-transformation. By  $S^d$  we mean the transformation  $S$ ,  $d$ -times repeated. A negative  $d$  indicates the use of the inverse translation  $S^{-1}$ .

**DEFINITION 4.5 (Shift equivalence of states)** Two configurations  $X$  and  $Y \in \mathcal{H}_{Q^k}$  are *shift-equivalent* if there exists a  $d \in \mathbb{Z}$  such that:  $X = T^d(Y)$ .

Because we allow negative shifts this is an equivalence relation.

**DEFINITION 4.6 (Shift equivalence of automata)** Two QCA  $F$  and  $F'$  (both with state-set  $Q$ ) are *shift-equivalent* if there exists a  $d \in \mathbb{Z}$  such that for every  $X \in \mathcal{H}_{Q^k}$  we have:  $F_k(X) = S^d(F'_k(X))$ . A shorthand for this relation is  $F = S^d \circ F'$  or  $F \equiv F'$ .

Note that the value  $d$  does not depend on the input  $X$  or even the size  $k$  of the structure. With these definitions we can describe the following lemma.

**LEMMA 4.1** For every QCA  $F$  there exists a normalized QCA  $F'$  which is shift-equivalent with  $F$ .

PROOF. (by construction) Given the QCA  $F = \langle Q, N, f \rangle$  we define  $F'$  with the tuple  $\langle Q', N', f' \rangle$  in the following way: an equal state set,  $Q' = Q$ , a neighborhood scheme  $N' = \{0, 1, \dots, N_{|N|} - N_1\}$  and the local function:  $f' : Q^{N'} \rightarrow \mathcal{H}_Q$  with:

$$f'(X_0, X_1, \dots, X_{|N'|-1}) = f(X_0, X_{N_2-N_1}, X_{N_3-N_1}, \dots, X_{|N'|-1})$$

for every  $X \in Q^{N'}$  With this construction we have  $F = S^{N_1} \circ F'$ .  $\square$

This  $F'$  is called the *normalized version* of  $F$ . If  $F' = S^d \circ F$  we have for every  $t \in \mathbb{N}$ :  $F^t = S^{ts} \circ F'^t$  (this is because  $F \circ S = S \circ F$  for every QCA  $F$ ).

## 4.2.2 Well-formed QCA

We start the discussion of well-formed QCA with a counter example.

**EXAMPLE 4.1** Take a QCA defined by  $\langle \{0, 1\}, \{0, 1\}, f \rangle$ , with  $f$  defined by:

$$\begin{aligned} f(0, 0) &= |0\rangle & f(0, 1) &= \frac{1}{2}\sqrt{2}\{|0\rangle + |1\rangle\} \\ f(1, 0) &= \frac{1}{2}\sqrt{2}\{|0\rangle - |1\rangle\} & f(1, 1) &= |1\rangle \end{aligned}$$

This gives us the following evolution from the initial state  $|0, 0, 1\rangle \in \mathcal{H}_{Q^3}$ :

$$\begin{aligned} |0, 0, 1\rangle &\xrightarrow{F_3} f(0, 0) \otimes f(0, 1) \otimes f(1, 0) \\ &= |0\rangle_0 \otimes \left\{ \frac{1}{2}\sqrt{2}\{|0_1\rangle + |1_1\rangle\} \right\} \otimes \left\{ \frac{1}{2}\sqrt{2}\{|0_2\rangle - |1_2\rangle\} \right\} \\ &= \frac{1}{2}\{|0, 0, 0\rangle - |0, 0, 1\rangle + |0, 1, 0\rangle - |0, 1, 1\rangle\} \\ &\xrightarrow{F_3} \frac{1}{4}\{2|0, 0, 0\rangle + |0, 0, 1\rangle - 3|0, 1, 0\rangle + 2|0, 1, 1\rangle + |1, 0, 0\rangle - 2|1, 1, 0\rangle + |1, 1, 1\rangle\} \end{aligned}$$

This last vector is not proper: its norm equals  $\sqrt{22}/4 \approx 1.17$ . Because this contradicts the laws of quantum mechanics it is said that this QCA is not well-formed: the transformation  $F_3$  is not unitary.  $\diamond$

This shows that our definition of QCA allows automata which are not proper. What we mean by well-formed QCA is defined as follows.

**DEFINITION 4.7 (Well-formed QCA)** A QCA  $F$  is *well-formed* if and only if for every  $k \in \mathbb{N}^+$  the corresponding  $F_k$  is a unitary transformation.



This means that  $F_k$  applied to the basis states  $Q^k$  has to be both norm and angle preserving in the Hilbertian space  $\ell_2(Q^k)$ . Because the range of  $F_k(Q^k)$  consists of proper non-entangled states, it holds for every  $\chi$  and  $v \in Q^k$  that  $F_k|\xi\rangle$  and  $F_k|v\rangle$  can be written as a tensor product:

$$F_k|\chi\rangle = \bigotimes_{i \in \mathbb{Z}_k} x_i \quad \text{and} \quad F_k|v\rangle = \bigotimes_{i \in \mathbb{Z}_k} y_i$$

with  $x_i, y_i \in \mathcal{H}_Q$ . The inner-product of these two vectors can therefore be calculated by the multiplication of the inner products of the individual cell-values;

$$\langle F_k|\chi\rangle, F_k|v\rangle \rangle = \left\langle \bigotimes_{i \in \mathbb{Z}_k} x_i, \bigotimes_{i \in \mathbb{Z}_k} y_i \right\rangle = \prod_{i \in \mathbb{Z}_k} \langle x_i, y_i \rangle$$

For every  $x_i \in \mathcal{H}_Q$  we know that  $\langle x_i, x_i \rangle = 1$ . This means that for every QCA the  $F_k$  by definition preserves the norm of the basis states:

$$\langle F_k|\chi\rangle, F_k|\chi\rangle \rangle = \left\langle \bigotimes_{i \in \mathbb{Z}_k} x_i, \bigotimes_{i \in \mathbb{Z}_k} x_i \right\rangle = \prod_{i \in \mathbb{Z}_k} \langle x_i, x_i \rangle = 1$$

If we want  $F_k$  to be angle-preserving we must have for every  $\xi \neq v \in Q^k$  (and therefore  $\langle \xi, v \rangle = 0$ )

$$\langle F_k|\chi\rangle, F_k|v\rangle \rangle = \left\langle \bigotimes_{i \in \mathbb{Z}_k} x_i, \bigotimes_{i \in \mathbb{Z}_k} y_i \right\rangle = \prod_{i \in \mathbb{Z}_k} \langle x_i, y_i \rangle = 0$$

This means that there must exist a  $i \in \mathbb{Z}_k$  for which  $\langle x_i, y_i \rangle = 0$ . The following lemma is easy to prove.

**LEMMA 4.2** For every QCA  $F$ , if  $F'$  is the normalized QCA with  $F = S^d(F')$  (as described in lemma 4.1) the well-formedness of  $F$  equals the well-formedness of  $F'$ .

PROOF.  $S$  is a norm and angle-preserving transformation on  $\mathcal{H}_{Q^k}$ . □

### 4.3 Proving Well-formedness

In this chapter we will show that the well-formedness restriction can be translated into a local constraint. This enables us to formulate an algorithm that decides if a given QCA is well-formed or not.

**DEFINITION 4.8 (General function of a QCA)** Given a QCA  $F = \langle Q, N, f \rangle$  we define the general function  $F_{\mathbb{Z}} : Q^{\mathbb{Z}} \rightarrow (\mathcal{H}_Q)^{\mathbb{Z}}$  by:

$$F_{\mathbb{Z}}(X) = \bigotimes_{i \in \mathbb{Z}} f(X_{i+N}) \in (\mathcal{H}_Q)^{\mathbb{Z}}$$

for every  $X \in Q^{\mathbb{Z}}$ .

This function describes only the first time step of a QCA on the two-way infinite basis states. Using a well-formedness definition for  $F_{\mathbb{Z}}$  we can investigate the unitarity of the periodic functions  $F_k$  for all values of  $k$ .

**DEFINITION 4.9 (Well-formed general function)** The function  $F_{\mathbb{Z}}$  is well-formed if and only if for every  $X \neq Y \in Q^{\mathbb{Z}}$  there exists an  $i \in \mathbb{Z}$  such that:

$$\langle F_{\mathbb{Z}}(X)_i, F_{\mathbb{Z}}(Y)_i \rangle = 0$$

We will now prove the equivalence of both definitions of well-formedness for QCA.

**LEMMA 4.3** *A QCA  $F$  is well-formed if and only if its corresponding general function  $F_{\mathbb{Z}}$  is well-formed.*

PROOF. **(If)** If  $F = \langle Q, N, f \rangle$  is not well-formed then there exists a  $k$  such that  $F_k$  is not angle-preserving. This means that there exists a  $\chi$  and  $v \in Q^k$  with  $\chi \perp v$  and  $F_k|\chi\rangle \not\perp F_k|v\rangle$ . If we define the two-way infinite states  $X, Y \in Q^{\mathbb{Z}}$  by:

$$X = \bigotimes_{i \in \mathbb{Z}} \chi \quad \text{and} \quad Y = \bigotimes_{i \in \mathbb{Z}} v$$

it follows that for every  $i \in \mathbb{Z}$  we have:

$$F_{\mathbb{Z}}(X)_i = (F_k|\chi\rangle)_{i \bmod k} \quad \text{and} \quad F_{\mathbb{Z}}(Y)_i = (F_k|v\rangle)_{i \bmod k}$$

Because  $X \neq Y$  and  $F_{\mathbb{Z}}(X)_i \not\perp F_{\mathbb{Z}}(Y)_i$  for every  $i \in \mathbb{Z}$ ,  $X$  and  $Y$  violate the well-formedness definition of  $F_{\mathbb{Z}}$ .

**(Only if)** Without loss of generality we assume  $F$  to be normalized. If  $F_{\mathbb{Z}}$  is not well-formed then there exist two basis states  $X, Y \in Q^{\mathbb{Z}}$  with  $X_0 \neq Y_0$  and

$$\langle F_{\mathbb{Z}}(X)_i, F_{\mathbb{Z}}(Y)_i \rangle \neq 0$$

for every  $i \in \mathbb{Z}$ . In Appendix B it is shown that with the states  $X$  and  $Y$  we can construct a  $k \leq 2|Q|^{2|N|} + |N|$  and two periodic states  $X', Y' \in Q^k$  with  $|X'\rangle \perp |Y'\rangle$  and  $F_k|X'\rangle \not\perp F_k|Y'\rangle$ . This proves that the QCA  $F$  is not well-formed.  $\square$

In the above lemma, the proof is more important the conclusion: with little extra effort, we can deduce the following *local constraint* for well-formed QCA. This lemma will be the backbone of this thesis.

**LEMMA 4.4** *For every well-formed QCA  $F = \langle Q, N, f \rangle$  there exist two values  $p, q \in \mathbb{Z}$  with:*

$$-|Q|^{2(N|N|-N_1+1)} - N_1 \leq p \leq q < |Q|^{2(N|N|-N_1+1)} + N|N| - 2N_1 + 1$$

such that for every  $X, Y \in Q^{\mathbb{Z}}$  with  $X_0 \neq Y_0$  we have:

$$\prod_{i=p}^q \langle F_{\mathbb{Z}}(X)_i, F_{\mathbb{Z}}(Y)_i \rangle = 0$$

PROOF. Because Lemma 4.3 is ‘if and only if’, it follows (see Appendix B) that for every normalized, well-formed QCA  $F = \langle Q, \{0, \dots, |N| - 1\}, f \rangle$  there exists a  $p \geq -|Q|^{2|N|}$  and a  $q < |Q|^{2|N|} + |N|$ , such that for every  $X, Y \in Q^{\mathbb{Z}}$  (with  $X_0 \neq Y_0$ ), we have:

$$\prod_{i=p}^q \langle F_{\mathbb{Z}}(X)_i, F_{\mathbb{Z}}(Y)_i \rangle = 0$$

The constructions in Lemma 4.1 and Lemma 4.2 generalizes this to the desired result.  $\square$

One of the important consequences of this lemma can now be proven directly.

**LEMMA 4.5** *The well-formedness property of a QCA  $F = \langle Q, N, f \rangle$  is decidable.*

PROOF. If  $F$  is not well-formed, there exists a  $k \in \mathbb{N}^+$  such that the finite dimensional transformation  $F_k$  is not unitary. Because  $k$  is bounded by

$$k \leq 2|Q|^{2(N|N|-N_1+1)}$$

this is decidable in finite time.  $\square$

The importance of of this bounded restriction lies in the fact that it does not depend on the local function  $f$  of a QCA. For classical reversible CA the range  $[p, q]$  is also known as the *inverse neighborhood* of a CA. Previous work by Jarkko Kari [31] suggests that the bound in Lemma 4.4 can be made smaller.

If the domain of a well-formed QCA is a proper state-space, the range will also be a proper state-space. In that case we can write  $F_k : \mathcal{H}_{Q^k} \rightarrow \mathcal{H}_{Q^k}$ . If we want to respect the physical laws, only a fraction of the possible QCA may be considered meaningful. This resembles the situation with classical CA where reversibility is a strong restriction on the automata [2, 48] From now on it will be understood that by QCA we mean well-formed QCA.

### 4.3.1 A definition of Balancedness

Recently, Christoph Dürr *et al.* [21] raised the question about a definition of ‘balancedness’ in the case of quantum CA. Here we give a generalization of the classical definition used by Amoroso and Patt [2] (which differs from the one used by Maruoka and Kimura [38]).

**DEFINITION 4.10 (Balanced QCA)** A QCA  $F = \langle Q, f, N \rangle$  will be called *balanced* if and only if

$$\sum_{x \in Q^N} |\langle q, f(x) \rangle|^2 = |Q|^{|N|-1}$$

for every  $q \in \mathcal{H}_Q$ ,

The following lemma shows the validity of this definition.

**LEMMA 4.6** Every well-formed QCA  $F = \langle Q, f, N \rangle$  is balanced.

PROOF. If we take  $k = N_{|N|} - N_1 + 1$ , the summation of  $f(x)$  is ‘encapsulated’ in the summation on the set of basis states  $Q^k$ :

$$\sum_{x \in Q^N} |\langle q, f(x) \rangle|^2 = |Q|^{|N|-k} \sum_{X \in Q^k} |\langle q, (F_k|X) \rangle_0|^2$$

With Lemma 1.1 we can derive the equality:

$$\begin{aligned} |Q|^{|N|-k} \sum_{X \in Q^k} |\langle q, (F_k|X) \rangle_0|^2 &= |Q|^{|N|-k} \sum_{X \in Q^k} \left( \sum_{Y \in Q^{k-1}} |\langle q \otimes Y, (F_k|X) \rangle|^2 \right) \\ &= |Q|^{|N|-k} \sum_{Y \in Q^{k-1}} \left( \sum_{X \in Q^k} |\langle q \otimes Y, (F_k|X) \rangle|^2 \right) \end{aligned}$$

Because  $F_k$  is a unitary transformation such that  $\mathcal{H}_{F_k(Q^k)} = \mathcal{H}_{Q^k}$  with  $q \otimes Y$  a vector with norm 1 in  $\mathcal{H}_{Q^k}$  we reach:

$$|Q|^{|N|-k} \sum_{y \in Q^{k-1}} \left( \sum_{X \in Q^k} |\langle q \otimes y, (F_k|X) \rangle|^2 \right) = |Q|^{|N|-k} \sum_{y \in Q^{k-1}} 1 = |Q|^{|N|-1}$$

which proves the lemma. □

This lemma tells us that a QCA can only be well-formed if the output values of the local function  $f$  are uniformly distributed on the state space  $\mathcal{H}_Q$ . This is not a sufficient restriction for properness, there exist balanced QCA which are not well-formed (see Example 4.1).

## 4.4 Quiescent Quantum Cellular Automata

In order to relate this thesis to some earlier results about QCA, we will give a short description of QCA with *quiescent states*: Quiescent QCA.

**DEFINITION 4.11 (Quiescent QCA)** A QQCA  $F$  is a QCA defined by the tuple  $\langle Q, \square, N, f \rangle$ , with a quiescent state  $\square \in Q$ , such that  $f(\square^N) = |\square\rangle$ .

This definition is used to overcome the problems of describing the behavior of a QCA in an uncountable infinite state space  $\mathcal{H}_{Q^z}$ . The only configurations that are allowed for a QQCA are those with only a finite number of non-quiescent states. Because the left and right tail of such a configuration  $(\cdots \square \square x_0 x_1 \cdots x_N \square \square \cdots)$  will remain quiescent under the action of a QCA, only a countable subset of basis states will be necessary to describe the evolution of the system. The set of basis states of these finite configurations is denoted by  $Q^*$ .

For every QQCA  $F$  the time operator on the superposition of finite configurations  $F_* : \ell_2(Q^*) \rightarrow \ell_2(Q^*)$  is defined by

$$\begin{aligned} F_*(X) &= F_* \left( \sum_{\xi \in Q^*} \alpha_\xi |\xi\rangle \right) \\ &= \sum_{\xi \in Q^*} \alpha_\xi \cdot F_* |\xi\rangle \end{aligned}$$

for every  $X \in \ell_2(Q^*)$ . The function  $F_* |\xi\rangle$  on the basis states  $Q^*$  is determined by:

$$F_* |\xi\rangle = \sum_{j \in \mathbb{Z}} f(\xi_{j+N}) \quad (\in \ell_2(Q^*))$$

The well-formedness issue of the function  $F_*$  is subtle and has some potential pitfalls. Here is a first attempt of definition.

**DEFINITION 4.12 (Well-formed QQCA)** Given a QQCA  $F = \langle Q, \square, N, f \rangle$ , the function  $F_*$  is well-formed if and only if for every  $X, Y \in Q^*$ : the norm of  $F_* |X\rangle$  equals 1 and  $X \perp Y$  implies  $F_* |X\rangle \perp F_* |Y\rangle$ .

Because the domain of the local function  $f$  is a subset of  $\mathcal{H}_Q$  the norm of  $F_*(X)$  will always be 1. The following example will show that there exist well-formed QQCA which are not unitary.

**EXAMPLE 4.2** Take the QQCA  $F = \langle \{\square, \bullet\}, \square, \{0, 1\}, f \rangle$ , with:

$$f(\square, \square) = f(\bullet, \bullet) = |\square\rangle \quad \text{and} \quad f(\square, \bullet) = f(\bullet, \square) = |\bullet\rangle$$

The function  $F_*$  is well-formed but not injective and therefore not unitary. This is shown by the equation  $F_*(X) = |\dots \square \square \bullet \square \square \dots\rangle$  which can not be satisfied for  $X \in \mathcal{H}_{Q^*}$ .  $\diamond$

This example indicates that we have to restrict definition 4.12 to the injective QQCA.

**DEFINITION 4.13 (Unitary QQCA)** Given a QQCA  $F = \langle Q, \square, N, f \rangle$ , the function  $F_*$  is unitary if and only if the function  $F_*$  is well-formed and for every  $Y \in Q^*$  there exists a  $X \in \mathcal{H}_{Q^*}$  with  $F_*(X) = |Y\rangle$ .

Because QQCA are a subset of QCA as defined in Definition 4.1 we can relate the above definition of unitary QQCA to that of well-formed QCA.

**LEMMA 4.7** If a QQCA  $F = \langle Q, \square, N, f \rangle$  is well-formed as described in Definition 4.7 then the  $F_*$  is a unitary function.

PROOF. (by contradiction) If the function  $F_*$  is not unitary there are two possibilities:

1. there exist  $X \perp Y \in Q^*$  with  $F_*|X\rangle \not\perp F_*|Y\rangle$
2. there exists an  $X \in Q^*$  with no predecessor in the domain  $\mathcal{H}_{Q^*}$ .

Because  $X, Y$  have a left and right tail which is quiescent, it is possible to take  $k \in \mathbb{N}^+$  sufficiently large, such that the same possibility holds for  $F_k$ . This shows that  $F_k$  is not unitary and therefore proves  $F$  not to be well-formed.  $\square$

The converse of this lemma would be: “If a QCCA has a unitary  $F_*$ , then the QCA  $F$  is well-formed.” The following example shows that this is not the case.

**EXAMPLE 4.3** Take the QCCA  $F = \langle \{\square, 0, 1\}, \square, \{0, 1\}, f \rangle$ , with  $f$  defined by:

$$\begin{aligned} f(\square, \square) &= |\square\rangle & f(\square, x) &= |\square\rangle \\ f(x, \square) &= |x\rangle & f(x, y) &= |x \oplus y\rangle \end{aligned}$$

for every  $x, y \in \{0, 1\}$ . Although the function  $F_*$  is unitary, the QCA  $F = \langle \{\square, 0, 1\}, N, f \rangle$  is not well-formed. This is shown by  $F_2|0, 0\rangle = F_2|1, 1\rangle = |0, 0\rangle$ .  $\diamond$

We can summarize the above lemma’s and examples in the following hierarchy with  $F$  a QCCA:

$$\boxed{\begin{array}{l} F_k \text{ is unitary for every } k \in \mathbb{N}^+ \\ F_Z \text{ is well-formed} \end{array}} \begin{array}{l} \implies \\ \not\Leftarrow \end{array} \boxed{F_* \text{ is unitary}} \begin{array}{l} \implies \\ \not\Leftarrow \end{array} \boxed{F_* \text{ is well-formed}}$$

It is still unknown how to define well-formedness for non-quiescent state QCA on a two-way infinite structure with an uncountable dimensional state space  $\mathcal{H}_{Q^z}$ .

## 4.5 Partitioned Quantum Cellular Automata

The results of the previous sections showed how to decide if a QCA is proper or not. What we still do not know is how to construct proper QCA with non-trivial behavior. This situation is no different with the problem of non-trivial reversible classical CA. A solution to this problem is (partly) given by *partitioned* CA [19, 44, 45]. Partitioned CA are constructed in such a way that the well-formedness of the automaton is implied by their definition. John Watrous applied this method to construct a QCA which can simulate a quantum Turing machines [57].

**DEFINITION 4.14 (Partitioned QCA)** A *partitioned quantum cellular automaton* (PQCA) is defined by a tuple  $\langle Q_L, Q_C, Q_R, g \rangle$ , with  $Q_L, Q_C$  and  $Q_R$  finite state sets and  $g : \mathcal{H}_{Q_L \times Q_C \times Q_R} \rightarrow \mathcal{H}_{Q_L \times Q_C \times Q_R}$  a unitary transformation.

A PQCA defines a special type of QCA  $F = \langle Q, \{-1, 0, 1\}, f \rangle$  with  $Q = Q_L \times Q_C \times Q_R$ . This product state set reflects the idea to decompose the cell-values into three disjoint parts Left, Center and Right. Each state  $x \in Q$  is therefore identified by three sub-states:  $x = x^l \times x^c \times x^r$ . This is used to translate the unitary function  $g$  into the local function  $f : Q^3 \rightarrow \mathcal{H}_Q$  by:

$$f(x_{-1}, x_0, x_1) = g(x_1^l \times x_0^c \times x_{-1}^r)$$

for every  $x_{-1}, x_0, x_1 \in Q$ . Because  $g$  is a unitary transformation, the QCA  $F$  will have a local behavior which is also unitary. From this it follows that  $F$  will be a well-formed QCA. See Figure 4.2 for an example of such a PQCA. With the use of PQCA we can define QCA in a straightforward way without having to worry about the well-formedness constraint. This has proven to be a fruitful concept in the theory of reversible CA and proper QCA. It assures us that the model of QCA is a non-trivial way of describing parallel quantum computation.

**LEMMA 4.8** *There exists a proper QCA which can simulate a Universal quantum Turing machine with polynomial time complexity.*

PROOF. See the original paper by John Watrous [57] which combines the earlier results on reversible CA and quantum Turing Machines.  $\square$

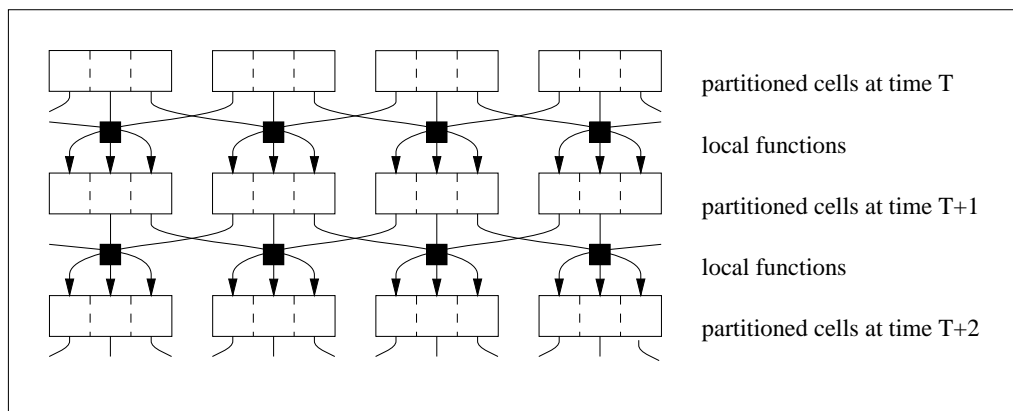


Figure 4.2: A partitioned quantum cellular automaton. The local function corresponds to a unitary transformation on the three sub-parts of a cell-value. The well-formedness of the local function gives us a global behavior of a well-formed QCA.

## Chapter 5

# Quantum Gate Cellular Automata

The definition of the quantum gate cellular automata (QGCA) is strongly related to the notion of quantum gate arrays. By replacing the local function of a QCA by a proper quantum gate we get a QCA which is proper by definition. This resembles the construction of partitioned QCA. Because of its central role in this thesis an extensive description will be given.

### 5.1 Description of QGCA

A quantum gate cellular automaton (QGCA) is a parallel and uniform circuit with identical quantum gates. See Figure 5.1 for an example of such a structure.

To avoid the problems of infinite sized systems, we will only consider QGCA with a *circular bounded* structure. The number of input/output values will be called the *fan* of the gates. For a proper quantum gate the fan-in has to equal to the fan-out, so no confusion is possible with this terminology. If we have  $k$  gates with fan  $n$  the number of cell-values will be  $nk$ . For clarity we use two indices to indicate an individual cell on the structure  $\mathbb{Z}_{nk}$ . Every basis state  $\xi$  of such a system can be described by the tensor product of the individual cell values:

$$|\xi\rangle = x_0^0 \otimes x_0^1 \otimes \cdots \otimes x_0^{n-1} \otimes x_1^0 \otimes \cdots \otimes x_1^{n-1} \otimes \cdots \otimes x_{k-1}^0 \otimes \cdots \otimes x_{k-1}^{n-1}$$

which equals

$$|\xi\rangle = \bigotimes_{j \in \mathbb{Z}_k} \left( \bigotimes_{i \in \mathbb{Z}_n} x_j^i \right) = \bigotimes_{j \in \mathbb{Z}_k} x_j$$

with  $x_j^i \in Q$  for every  $i \in \mathbb{Z}_n$  and  $j \in \mathbb{Z}_k$ .

Because the gates are in a parallel position, the behavior of a layer of gates is described by a unitary matrix  $U$  which is the  $k$ -fold tensor product of the individual gates  $M$ . Therefore  $U = M^{[k]}$ .

The *neighborhood scheme* describes the ‘wiring’ between the gates. Each gate has the same neighborhood scheme which does not depend on the size  $k$  of the circuit.

**DEFINITION 5.1 (Neighborhood scheme)** A neighborhood scheme  $P$  for gates with fan  $n \in \mathbb{N}^+$  is defined by a tuple  $\langle n, \sigma, \phi \rangle$ . Both  $\sigma : \mathbb{Z}_n \rightarrow \mathbb{Z}_n$  and  $\phi : \mathbb{Z}_n \rightarrow \mathbb{Z}$  are functions, with  $\sigma$  a bijection on  $\mathbb{Z}_n$ .

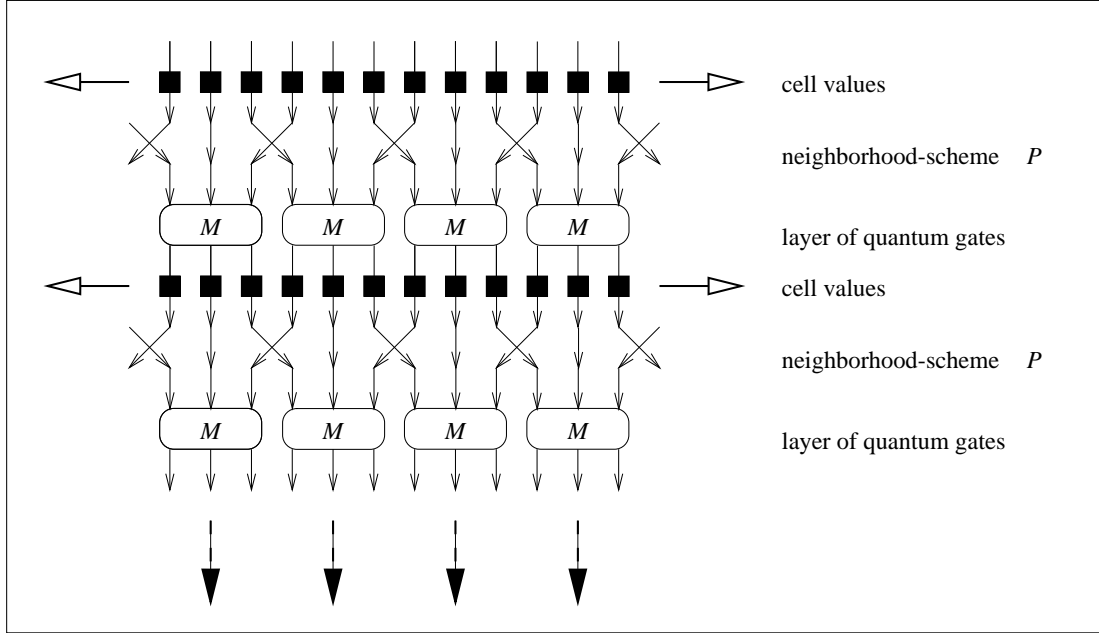


Figure 5.1: A part of a quantum gate cellular automaton. All the gates  $M$  are identical. The neighborhood scheme  $P$  defines the wiring between the gates. Because  $M$  is proper gate, the global transformation function of this system will also be proper. We therefore know that this structure resembles a well-formed QCA.

A neighborhood scheme  $P$  permutes the values of a basis state  $\xi \in Q^{nk}$  in the following way.

$$\begin{aligned} P|\xi\rangle &= P\left(\bigotimes_{j \in \mathbb{Z}_k} \left(\bigotimes_{i \in \mathbb{Z}_n} |\xi_j^i\rangle\right)\right) \\ &= \bigotimes_{j \in \mathbb{Z}_k} \left(\bigotimes_{i \in \mathbb{Z}_n} |\xi_{j+\phi(i)}^{\sigma(i)}\rangle\right) \end{aligned}$$

This shows that  $P$  corresponds to a function  $\mathbb{Z}_{nk} \rightarrow \mathbb{Z}_{nk}$  and also that the description of  $P$  does not depend on the involved set of states  $Q$ . Because  $\sigma$  is a bijection,  $P$  will also be a bijection for every  $k \in \mathbb{N}^+$ . This is necessary because we want  $P$  to be a proper transformation. If  $P$  acts on a superposition, the behavior of  $P$  will respect this superposition:

$$P|X\rangle = P\left(\sum_{\xi \in Q^{nk}} \alpha_\xi |\xi\rangle\right) = \sum_{\xi \in Q^{nk}} \alpha_\xi P|\xi\rangle$$

for every  $|X\rangle \in \mathcal{H}_{Q^{nk}}$ .

**EXAMPLE 5.1** The neighborhood scheme as shown in Figure 5.1 is described by  $\langle n = 3, \sigma, \phi \rangle$  with:

$$\begin{aligned} \sigma(0) &= 2 & \phi(0) &= -1 \\ \sigma(1) &= 1 & \text{and } \phi(1) &= 0 \\ \sigma(2) &= 0 & \phi(2) &= 1 \end{aligned}$$

◇

The reader is encouraged to check the above example, to get a better insight in the used formalism. Note that the neighborhood scheme acts *before* the layer of gates is applied.



### 5.1.1 Formal Definition of QGCA

We are now ready to give a formal definition of a QGCA.

**DEFINITION 5.2 (Quantum gate cellular automata)** A QGCA  $F$  is defined by  $\langle Q, n, M, P \rangle$  with  $Q$  a finite state set,  $n \in \mathbb{N}^+$ ,  $M : \mathcal{H}_{Q^n} \rightarrow \mathcal{H}_{Q^n}$  a proper quantum gate with fan  $n$  and  $P$  a neighborhood scheme  $\langle n, \sigma, \phi \rangle$ .

As with QCA this definition does not specify the size  $k$  of the array. A *layer* of a QGCA corresponds to the sequential combination of a neighborhood scheme  $P$  and a row of gates  $M^{[k]}$ . One layer can be identified by one time step of the QGCA. The global behavior of a QGCA  $F = \langle Q, n, M, P \rangle$  of size  $k$  is denoted by  $F_k$  and obeys:

$$\begin{aligned} F_k(X) &= M^{[k]} \circ P \left( \sum_{\xi \in Q^{nk}} \alpha_\xi |\xi\rangle \right) \\ &= \sum_{\xi \in Q^{nk}} \alpha_\xi M^{[k]} \circ P |\xi\rangle \end{aligned}$$

for every  $X \in \mathcal{H}_{Q^{nk}}$ . The effect on the basis states  $\xi$  is defined by:

$$\begin{aligned} M^{[k]} \circ P |\xi\rangle &= M^{[k]} \circ P \left( \bigotimes_{j \in \mathbb{Z}_k} \left( \bigotimes_{i \in \mathbb{Z}_n} \xi_j^i \right) \right) \\ &= M^{[k]} \left( \bigotimes_{j \in \mathbb{Z}_k} \left( \bigotimes_{i \in \mathbb{Z}_n} \xi_{j+\phi(i)}^{\sigma(i)} \right) \right) \\ &= \bigotimes_{j \in \mathbb{Z}_k} \left( M \left( \bigotimes_{i \in \mathbb{Z}_n} \xi_{j+\phi(i)}^{\sigma(i)} \right) \right) \end{aligned}$$

The function of a sequence of  $t$  layers  $F_k$  will be described by  $F_k^t$ .

## 5.2 Preliminaries

Before continuing with the general theory of quantum gate arrays, we will take a closer look at a particular neighborhood scheme which plays an important role: the Shift neighborhood scheme.

### 5.2.1 The Shift Neighborhood Scheme

In this thesis we will concentrate on the Shift neighborhood scheme  $S$ , which simply ‘moves’ all the values one cell leftwards (see Figure 5.2 for an example). With the fan-value  $n$  this gives us the tuple  $\langle n, i+1, 1_{n-1}(i) \rangle$  with: “ $i+1$ ” an addition in  $\mathbb{Z}_n$  and  $1_c$  the function with  $1_c(x) = 1$  if  $x = c$  and  $1_c(x) = 0$  otherwise.

**EXAMPLE 5.2** Because of their size, it is not practical to actually write out the transformation-matrices. As an example of the matrix  $S$  for the simple two state system with  $kn = 3$  the equation

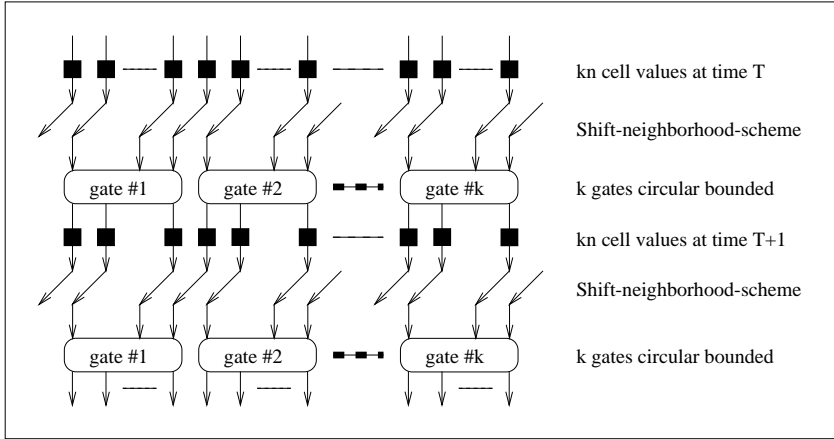


Figure 5.2: The shift neighborhood scheme. The neighborhood scheme  $S$  describes a leftwards translation on the cell values.

$S|0, 1, 1\rangle = |1, 1, 0\rangle$  looks like:

$$S|0, 1, 1\rangle = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 0_{|000\rangle} \\ 0_{|001\rangle} \\ 0_{|010\rangle} \\ 1_{|011\rangle} \\ 0_{|100\rangle} \\ 0_{|101\rangle} \\ 0_{|110\rangle} \\ 0_{|111\rangle} \end{pmatrix} = \begin{pmatrix} 0_{|000\rangle} \\ 0_{|001\rangle} \\ 0_{|010\rangle} \\ 0_{|011\rangle} \\ 0_{|100\rangle} \\ 0_{|101\rangle} \\ 1_{|110\rangle} \\ 0_{|111\rangle} \end{pmatrix} = |1, 1, 0\rangle$$

◇

From now on when using  $S$  in the context of QGCA, we will mean the shift neighborhood scheme. The inverse of  $S$  (a shift rightwards) is denoted by  $S^{-1}$ .

### 5.2.2 Two state QGCA

The following lemma shows us how to simulate a QGCA with an arbitrary state set  $Q$  by a QGCA which uses qubits.

**LEMMA 5.1** *Every QGCA  $F = \langle Q, n, M, P \rangle$  can be simulated by a two state QGCA*

$$G = \langle \{0, 1\}, n', M', P' \rangle$$

with an equal number of computational steps.

PROOF. (by construction) First expand the set  $Q$  to a set  $R \supseteq Q$  such that  $|R| = 2^\lambda$  with  $\lambda = \lceil \log_2 |Q| \rceil$ . Now we can use a bijection  $\phi : R \rightarrow \{0, 1\}^\lambda$  which, induced to the  $n$  cells of each gate, gives us a bijection  $\Phi : R^n \rightarrow \{0, 1\}^{n\lambda}$ . With this  $\Phi$  we define the new quantum gate  $M'$  on  $\mathcal{H}_{\{0,1\}^{n\lambda}}$  by: for every  $x \in R^n$  by:

$$M'(\Phi(x)) = \begin{cases} \Phi^{-1}(M(x)) & \text{if } x \in Q^n \\ \Phi^{-1}(x) & \text{otherwise} \end{cases}$$

for every  $x \in R^n$ . The new neighborhood scheme  $P' = \langle n\lambda, \sigma', \phi' \rangle$  is obtained by splitting each 'wire' into  $\lambda$  parts:

$$\sigma'(i\lambda + t) = \sigma(i) \quad \text{and} \quad \phi'(i\lambda + t) = \phi(i)$$

for every  $i \in \mathbb{Z}_n$  and  $t \in \mathbb{Z}_\lambda$ .

This QGCA  $G = \langle \{0, 1\}, n\lambda, M', \langle n\lambda, \sigma', \phi' \rangle \rangle$  simulates the original QGCA according to the bijection  $\phi$ . If an illegal state is encoded ( $x \notin Q^n$ ) the  $M'$ -gate behaves like the identity. This shows that  $M'$  is a proper transformation.  $\square$

### 5.2.3 Combining neighborhood schemes

Here we will show how to express a neighborhood scheme  $R \circ P$  which is a combination of two initial schemes  $P$  and  $R$ . Given the two neighborhood schemes  $P = \langle n, \sigma, \phi \rangle$  and  $R = \langle n, \varsigma, \varphi \rangle$ , we can define the action of  $P$  by:

$$P \left( \bigotimes_{j \in \mathbb{Z}_k} \left( \bigotimes_{i \in \mathbb{Z}_n} x_j^i \right) \right) = \bigotimes_{j \in \mathbb{Z}_k} \left( \bigotimes_{i \in \mathbb{Z}_n} x_{j+\phi(i)}^{\sigma(i)} \right) = \bigotimes_{j \in \mathbb{Z}_k} \left( \bigotimes_{i \in \mathbb{Z}_n} y_j^i \right)$$

If we use the  $y$  expressions to describe the effect of  $R$  we get:

$$R \left( \bigotimes_{j \in \mathbb{Z}_k} \left( \bigotimes_{i \in \mathbb{Z}_n} y_j^i \right) \right) = \bigotimes_{j \in \mathbb{Z}_k} \left( \bigotimes_{i \in \mathbb{Z}_n} y_{j+\varphi(i)}^{\varsigma(i)} \right)$$

The combination of  $R$  and  $P$  therefore obeys (using the equation  $y_j^i = x_{j+\phi(i)}^{\sigma(i)}$ ):

$$R \circ P \left( \bigotimes_{j \in \mathbb{Z}_k} \left( \bigotimes_{i \in \mathbb{Z}_n} x_j^i \right) \right) = \bigotimes_{j \in \mathbb{Z}_k} \left( \bigotimes_{i \in \mathbb{Z}_n} x_{j+\varphi(i)+\phi(\varsigma(i))}^{\sigma(\varsigma(i))} \right)$$

This shows us that we can write:

$$R \circ P = \langle n, \varsigma(i), \varphi(i) \rangle \circ \langle n, \sigma(i), \phi(i) \rangle = \langle n, \sigma \circ \varsigma(i), \varphi(i) + \phi \circ \varsigma(i) \rangle$$

With regard to the shift neighborhood scheme we have:  $S = \langle n, i + 1, 1_{n-1}(i) \rangle$  and

$$S^n \circ P = \langle n, i, 1 \rangle \circ \langle n, \sigma(i), \phi(i) \rangle = \langle n, \sigma(i), 1 + \phi(i) \rangle = P \circ S^n$$

## 5.3 Every QGCA describes a QCA

If we look at the figures 4.2 and 5.1 we see a great resemblance. We already know that PQCA are a special kind of QCA. We will now prove the same holds for QGCA.

**LEMMA 5.2** *Every proper QGCA  $F = \langle Q, n, M, P \rangle$  defines a well-formed QCA  $QCA F = \langle Q^n, N, f \rangle$ .*

**PROOF.** The correspondence between the values of the QGCA and those of the QCA is expressed by:

$$x_j = x_j^0 \otimes \cdots \otimes x_j^{n-1}$$

for every  $x_j \in Q^n$ . The neighborhood scheme  $P = \langle n, \sigma, \phi \rangle$  gives us the neighborhood set of  $F$  according to:  $N = \{\phi(i) \mid i \in \mathbb{Z}_n\}$ . The local function  $f : (Q^n)^N \rightarrow \mathcal{H}_{Q^n}$  now obeys:

$$f(x_N) = M \left( \bigotimes_{i \in \mathbb{Z}_n} x_{\phi(i)}^{\sigma(i)} \right)$$

By this construction, for every  $k \in \mathbb{N}^+$  it holds that:

$$\begin{aligned}
G_k \left( \bigotimes_{j \in \mathbb{Z}_k} \left( \bigotimes_{i \in \mathbb{Z}_n} x_j^i \right) \right) &= M^{[k]} \cdot P_k \left( \bigotimes_{j \in \mathbb{Z}_k} \left( \bigotimes_{i \in \mathbb{Z}_n} x_j^i \right) \right) \\
&= M^{[k]} \left( \bigotimes_{j \in \mathbb{Z}_k} \left( \bigotimes_{i \in \mathbb{Z}_n} x_{j+\phi(i)}^{\sigma(i)} \right) \right) \\
&= \bigotimes_{j \in \mathbb{Z}_k} \left( M \left( \bigotimes_{i \in \mathbb{Z}_n} x_{j+\phi(i)}^{\sigma(i)} \right) \right) \\
&= \bigotimes_{j \in \mathbb{Z}_k} f(x_{j+N}) \\
&= F_k \left( \bigotimes_{j \in \mathbb{Z}_k} x_j \right)
\end{aligned}$$

□

As with PQCA the well-formedness of a QGCA  $\langle Q, n, M, P \rangle$  is certified by the well-formedness of the gate and the neighborhood scheme.

## Chapter 6

# QCA, PQCA, and QGCA are Equivalent

In this chapter will prove the equivalence of all three models for quantum cellular automata. By ‘equivalence’ we mean that every QCA can be simulated by a QGCA, every QGCA by a PQCA, every PQCA by a QCA etc. All these simulations have at most linear slowdown. Our main problem is to prove that every proper QCA can be simulated by a QGCA.

### 6.1 A Definition of Equivalence

Suppose we have two QCA  $F$  and  $G$ . If we say that  $G$  simulates  $F$ , we mean that there are two ‘simple transformations’  $\Phi_t$  and  $\Psi_t$  such that  $\Phi_t \circ G^{\lambda t} \circ \Psi_t$  equals  $F^t$ . The  $G^\lambda$ -expression indicates a simulation with only linear (proportional to  $\lambda$ ) slowdown. The transformations must be ‘simple’ because we want  $G$  to do the *actual* calculation. Or in the words of Marvin Minsky[43]:

( ... ) for if one is permitted an arbitrary partial-recursive computation to do the encoding ( ... ), then one could use as the code the result of the Turing-machine computation itself, and this would surely be considered a cheat!

With this warning in mind we pose the following (inductive) definition:

**DEFINITION 6.1 (Simple transformation)** *Let  $P, Q, R$  be state sets and  $a, b, c, d \in \mathbb{N}^+$ . Simple transformations depending on  $t \in \mathbb{Z}$ , are constructed by the following three procedures:*

1. If  $\phi : P^a \rightarrow Q^b$  is a total function, then  $\Phi(t) : P^{a\mathbb{Z}} \rightarrow Q^{b\mathbb{Z}}$  defined by:

$$\Phi(t) \left( \bigotimes_{i \in \mathbb{Z}} x_i \right) = \bigotimes_{i \in \mathbb{Z}} [\phi(x_{i\ell}, \dots, x_{i\ell+t-1})]_{im:(im+m-1)}$$

is a simple transformation.

2. If  $\Phi(t)$  is a simple transformation, then the transformations  $S \circ \Phi(t)$ ,  $S^{-1} \circ \Phi(t)$ ,  $S^t \circ \Phi(t)$  and  $S^{-t} \circ \Phi(t)$  are also simple (with  $S$  the shift translation).
3. If  $\Phi(t) : P^{a\mathbb{Z}} \rightarrow Q^{b\mathbb{Z}}$  and  $\Psi(t) : Q^{c\mathbb{Z}} \rightarrow R^{d\mathbb{Z}}$  are simple transformations, then the combination  $\Theta(t) : P^{ac\mathbb{Z}} \rightarrow R^{bd}$  described by  $\Theta(t) = \Psi(t) \circ \Phi(t)$  is also a simple transformation.

A simple transformation  $\Phi(t) : P^{a\mathbb{Z}} \rightarrow Q^{b\mathbb{Z}}$  defines for every  $k \in \mathbb{N}^+$  the simple transformation  $\Phi_k(t) : \mathcal{H}_{P^{ka}} \rightarrow \mathcal{H}_{Q^{kb}}$  which obeys the superposition principle of the state space.

Now we can formulate the definition for simulation of quantum cellular automata.

**DEFINITION 6.2 (Simulating QCA)** A QCA  $G = \langle P, N', g \rangle$  can simulate a QCA  $F = \langle Q, N, f \rangle$ , if there exists a tuple  $\langle \lambda, \Phi, \Psi \rangle$  with  $\Phi(t) : Q^{a\mathbb{Z}} \rightarrow P^{b\mathbb{Z}}$  and  $\Psi(t) : P^{b\mathbb{Z}} \rightarrow Q^{a\mathbb{Z}}$  simple transformations and  $\lambda \in \mathbb{N}$  such that for every  $t, k \in \mathbb{N}^+$  and  $X \in \mathcal{H}_{Q^{ka}}$  we have:

$$F_{ka}^t(X) = \Phi_{ka}(t) \circ G_{kb}^{\lambda t} \circ \Psi_{kb}(t)(X)$$

Or in other words:

$$F_{ka}^t = \Phi_{ka}(t) \circ (G_{kb})^{\lambda t} \circ \Psi_{kb}(t)$$

If no misunderstanding is possible we can shorten this to:  $F^t = \Phi_t \circ G^{\lambda t} \circ \Psi_t$ .

When  $F$  can be simulated by  $G$ , we will write “ $\{F\} \preceq \{G\}$ ”. This relation is *certified* by  $\langle \lambda, \Phi, \Psi \rangle$ , this tuple will also be called the *encoding* of  $F$  into  $G$ .

With this definition we can define the notion of equivalence for quantum cellular automata.

**DEFINITION 6.3 (Equivalent QCA)** Two QCA  $F$  and  $G$  will be called *equivalent* if and only if  $F$  can simulate  $G$  and  $G$  can simulate  $F$ .

Consequently we will denote this by “ $\{F\} \simeq \{G\}$ ”. For every QCA  $F, G$  and  $H$  we now have:

1.  $\{F\} \preceq \{F\}$
2. If  $\{F\} \preceq \{G\}$  and  $\{G\} \preceq \{H\}$ , then  $\{F\} \preceq \{H\}$
3. If  $\{F\} \preceq \{G\}$  and  $\{G\} \preceq \{F\}$ , then  $\{F\} \simeq \{G\}$

If we take the ‘identity-encoding’  $\langle 1, I, I \rangle$ , we see that 1 holds, whereas 3 holds by definition. The second rule needs more explanation.

If  $\{F\} \preceq \{G\}$  is certified by  $\langle \lambda, \Phi, \Psi \rangle$  and  $\{G\} \preceq \{H\}$  by  $\langle \mu, \Delta, \Lambda \rangle$ , we have (in shorthand):  $F^t = \Phi_t \circ G^{\lambda t} \circ \Psi_t$  and  $G^j = \Delta_j \circ H^{\mu j} \circ \Lambda_j$  and therefore:

$$F^t = \Phi_t \circ \Delta_{\lambda t} \circ H^{\mu \lambda t} \circ \Lambda_{\lambda t} \circ \Psi_t$$

By defining  $\Xi_t = \Phi_t \circ \Delta_{\lambda t}$  and  $\Upsilon_t = \Lambda_{\lambda t} \circ \Psi_t$ , we see by Definition 6.1 that  $\Xi$  and  $\Upsilon$  are simple transformations. This shows that  $\langle \mu \lambda, \Xi, \Upsilon \rangle$  certifies the  $\{F\} \preceq \{H\}$ -relation.

Because the relations “ $\preceq$ ” and “ $\simeq$ ” are defined on *sets* of QCA we can extend their domain in the following way:

**DEFINITION 6.4 (Simulating sets of QCA)** If  $\mathcal{A}$  and  $\mathcal{B}$  are sets of quantum cellular automata, we say that  $\mathcal{B}$  can simulate  $\mathcal{A}$  if and only if for every  $F \in \mathcal{A}$  there exists a  $G \in \mathcal{B}$  which simulates  $F$ . This will be denoted by “ $\mathcal{A} \preceq \mathcal{B}$ ”.

**DEFINITION 6.5 (Equivalent sets of QCA)** If  $\mathcal{A}$  and  $\mathcal{B}$  are sets of quantum cellular automata, we say that  $\mathcal{A}$  and  $\mathcal{B}$  are equivalent if and only if both  $\mathcal{A} \preceq \mathcal{B}$  and  $\mathcal{B} \preceq \mathcal{A}$  holds. This equivalence is denoted by  $\mathcal{A} \simeq \mathcal{B}$ .

With this definition we have for every set of quantum cellular automata  $\mathcal{A}, \mathcal{B}$  and  $\mathcal{C}$ :

1.  $\mathcal{A} \preceq \mathcal{A}$
2. If  $\mathcal{A} \preceq \mathcal{B}$  and  $\mathcal{B} \preceq \mathcal{C}$ , then  $\mathcal{A} \preceq \mathcal{C}$
3. If  $\mathcal{A} \preceq \mathcal{B}$  and  $\mathcal{B} \preceq \mathcal{A}$ , then  $\mathcal{A} \simeq \mathcal{B}$
4.  $\mathcal{A} \simeq \mathcal{A}$
5. If  $\mathcal{A} \simeq \mathcal{B}$ , then  $\mathcal{B} \simeq \mathcal{A}$
6. If  $\mathcal{A} \simeq \mathcal{B}$  and “ $\mathcal{B} \simeq \mathcal{C}$ ”, then  $\mathcal{A} \simeq \mathcal{C}$

These rules show that  $\preceq$  is a partial ordering and  $\simeq$  is an equivalence relation. A few concluding remarks about the above definitions:

- $\mathcal{A} \simeq \mathcal{B}$  does not necessarily mean that for every  $F \in \mathcal{A}$  there exists a  $G \in \mathcal{B}$ , with  $F$  and  $G$  equivalent.
- The constant “ $\lambda$ ” indicates the linear slowdown of the simulation.
- If  $\mathcal{A} \subseteq \mathcal{B}$  then automatically  $\mathcal{A} \preceq \mathcal{B}$ .
- If  $F$  is shift equivalent with  $G$  then  $F$  and  $G$  are also equivalent.
- An interesting question is whether the  $\preceq$ -relation is *total* or not. Does for every QCA  $F$  and  $G$  it holds that  $\{F\} \preceq \{G\}$  or  $\{G\} \preceq \{F\}$ ?

## 6.2 Some Preliminary Results

The strategy of this chapter is to reduce the different sets of QCA, PQCA and QGCA to each other in a step-by-step way. Let us first summarize the relations we already know. We begin by using the fact that QGCA and PQCA are subsets of QCA.

**LEMMA 6.1** PQCA  $\preceq$  QCA.

PROOF. Partitioned QCA are just a special type of QCA, therefore PQCA  $\subseteq$  QCA.  $\square$

**LEMMA 6.2** QGCA  $\preceq$  QCA.

PROOF. Lemma 5.2 proves QGCA  $\subseteq$  QCA.  $\square$

To reduce the class of QCA we use the following lemma.

**LEMMA 6.3** The subset of normalized QCA is equivalent with the general set of QCA.

PROOF. Normalized QCA are a subset of QCA. Lemma 4.1 proves that every QCA is shift-equivalent with a normalized QCA, thus QCA  $\simeq$  QCA<sub>normalized</sub>.  $\square$

Now we will prove a new result which will simplify our reasoning about simulating QCA considerably.

**LEMMA 6.4** The set of QCA with neighborhood set  $N = \{0, 1\}$  is equivalent with the set of normalized QCA.

PROOF.(by construction) For every normalized QCA  $F$  defined by  $\langle Q, \{0, \dots, |N| - 1\}, f \rangle$  there exists a QCA  $G$  with neighborhood set  $N_G = \{0, 1\}$  which simulates  $F$  with no slowdown. In order to get a neighborhood size 2 we expand the set of states:  $G$  is described by the tuple  $\langle Q^{r-1}, \{0, 1\}, g \rangle$ . The local function  $g : (Q^{|N|-1})^2 \rightarrow \mathcal{H}_{Q^{|N|-1}}$  in this definition obeys

$$g(x_{1:(|N|-1)}, x_{|N|:(2|N|-2)}) = f(x_{1:|N|}) \otimes f(x_{2:(|N|+1)}) \otimes \dots \otimes f(x_{(|N|-1):(2|N|-2)})$$

with  $x_i \in Q$  and therefore  $x_{a:(a+|N|-2)} \in Q^{|N|-1}$ .

The encoding of  $F$  into  $G$  is established by the simple transformations:

$$\begin{aligned} \Psi : (Q)^{|N|-1} &\rightarrow (Q)^{|N|-1} && \text{with } \Psi(x_{1:|N|-1}) = x_1 \otimes \dots \otimes x_{|N|-1} \\ \Phi : (Q^{|N|-1}) &\rightarrow (Q)^{|N|-1} && \text{with } \Phi(x_1 \otimes \dots \otimes x_{|N|-1}) = x_{1:|N|-1} \end{aligned}$$

The tuple  $\langle 1, \Phi, \Psi \rangle$  now certifies  $\{F\} \preceq \{G\}$  with  $N_G = \{0, 1\}$ .  $\square$

A simple extension of this lemma will be the last of our preliminary results.

**LEMMA 6.5** The set of QCA with neighborhood  $N = \{0, 1\}$  is equivalent with the set of QCA.

PROOF. Combine the results: QCA  $\simeq$  QCA<sub>normalized</sub> and QCA<sub>normalized</sub>  $\simeq$  QCA <sub>$N=\{0,1\}$</sub> .  $\square$

### 6.3 Shift Neighborhood Scheme is Universal

In this section we will prove that every neighborhood scheme can be simulated using the shift-scheme  $S$ . It will be understood that  $k$  is the number of gates,  $n$  is the fan of the gates,  $P$  and  $R$  are neighborhood schemes and  $M_1, M_2, \dots$  are proper quantum gates.

#### 6.3.1 Periodic QGCA

A *periodic* QGCA is a non-homogeneous QGCA. It is build by a periodic pattern of different gates and neighborhood schemes.

**DEFINITION 6.6 (Periodic QGCA)** A periodic QGCA  $F$  of  $\mu$  layers is defined by a tuple

$$F = \langle Q, n, \mu, M_0, \dots, M_{\mu-1}, P_0, \dots, P_{\mu-1} \rangle$$

with  $M_i$  proper quantum gates and  $P_i$  proper neighborhood schemes for  $0 \leq i < \mu$ . The unitary operator of the first  $\mu$  layers of this QCA is defined by:

$$F_k = \prod_{i=\mu-1}^0 \left( (M_i)^{[k]} \cdot P_i \right)$$

(Notice the order of the index  $i$  and the fact that  $F_k$  is defined as the operator of *several* layers.)

**EXAMPLE 6.1** With the 3-periodic QGCA  $F = \langle Q, n, 3, A_0, A_1, A_2, P_0, P_1, P_2 \rangle$ ; the behavior of the first  $3j$  layers is described by:

$$F_k^j = \left( A_2^{[k]} \cdot P_2 \cdot A_1^{[k]} \cdot P_1 \cdot A_0^{[k]} \cdot P_0 \right)^j$$

◇

A periodic QGCA with  $\mu$  layers will be call a  $\mu$ -periodic QGCA. An 1-periodic QGCA is again a regular QGCA. This definition is a side-step from the regular classes QCA, PQCA, and QGCA to allow a more flexible way of defining quantum cellular automata. After that it will be proven that every periodic QGCA corresponds to a QGCA.

#### 6.3.2 Simulating Neighborhood Schemes

By allowing different layers in a QGCA we can construct a periodic QGCA that mimics the behavior of a neighborhood scheme. This is shown in the following lemma.

**LEMMA 6.6** For every QGCA  $F = \langle Q, n, M, P \rangle$  there exists a periodic QGCA  $G$  defined by

$$G = \langle Q, n, \mu, M_0, \dots, M_{\mu-1}, S, \dots, S \rangle$$

which is shift equivalent with  $F$ .

PROOF. See Appendix C for a proof by construction of this lemma. □

Figure 6.1 shows us an example of such a simulation by a periodic QGCA.

**EXAMPLE 6.2** The QGCA  $F = \langle Q, 3, M, P \rangle$  with a neighborhood scheme  $P = \langle 3, \sigma, \phi \rangle$  and

$$\begin{array}{ll} \sigma(0) = 2 & \phi(0) = -1 \\ \sigma(1) = 1 & \phi(1) = 1 \\ \sigma(2) = 0 & \phi(2) = 0 \end{array}$$



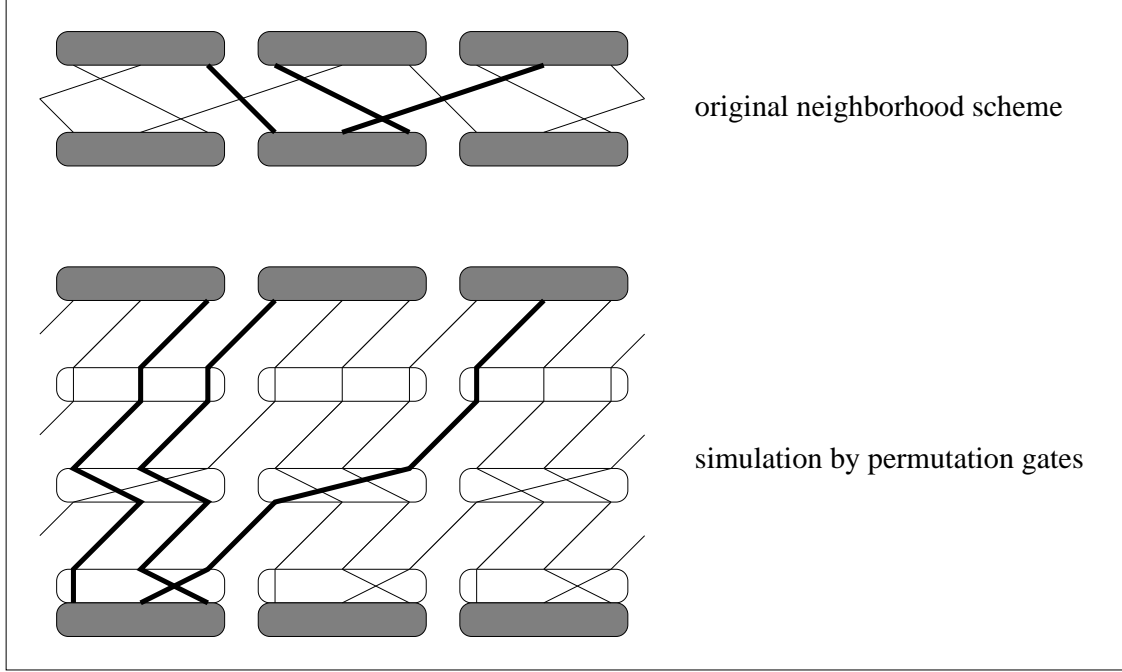


Figure 6.1: Simulation of the neighborhood scheme  $P$  in Example 6.2. The two additional layers of the periodic QGCA  $G$  consist of permutation gates which mimic the wiring of the original neighborhood scheme. The last permutation gate is combined with the gate  $M$  of the QGCA  $F$ . Because the periodic QGCA only uses the Shift scheme this simulation will be equivalent with initial QGCA  $F$  up to a  $S^3$  translation for every layer of  $F$ .

can be simulated by a 3-periodic QGCA

$$G = \langle Q, 3, 3, M_0, M_1, M_2, S, S, S \rangle.$$

The gates of this automaton are described by:

$$M_0|x, y, z\rangle = |x, y, z\rangle \quad M_1|x, y, z\rangle = |z, x, y\rangle \quad M_2|x, y, z\rangle = M|x, z, y\rangle$$

for every  $x, y, z \in Q$ . See Figure 6.1 for an illustration of this simulation by  $G$ . Notice how the last permutation gate and the  $M$  gate are combined to one gate  $M_2$ .  $\diamond$

### 6.3.3 Simulating Periodic QGCA

The use of periodic QGCA will only be temporary because we can simulate any periodic QGCA by a regular QGCA.

**LEMMA 6.7** For every periodic QGCA  $F = \langle Q, n, \mu, M_0, \dots, M_{\mu-1}, S, \dots, S \rangle$  there exists a QGCA  $G = \langle Q, n + \Delta, M', S \rangle$  with  $\Delta = \lceil \log(\mu) / \log(|Q|) \rceil$  which can simulate  $F$ .

PROOF. Let  $\lceil \dots \rceil$  be an injective function:  $\mathbb{Z}_\mu \rightarrow Q^\Delta$ . Define  $M' : \mathcal{H}_{Q^{n+\Delta}} \rightarrow \mathcal{H}_{Q^{n+\Delta}}$  for every  $j \in \mathbb{Z}_\mu$  by:

$$M'|x_0, \dots, x_{n-2}; \lceil j \rceil; x_{n-1}\rangle = M_j|x_0, \dots, x_{n-1}\rangle \otimes |\lceil j + 1 \rceil\rangle$$

We will use the simple transformations:  $\Psi : Q^n \rightarrow Q^{n+\Delta}$  and  $\Phi : Q^{n+\Delta} \rightarrow Q^n$  with

$$\Psi(x_0, \dots, x_{n-1}) = (x_0, \dots, x_{n-1}) \otimes (\lceil 0 \rceil)$$

and

$$\Phi(x_0, \dots, x_{n-1}, x'_1, \dots, x'_\Delta) = (x_0, \dots, x_{n-1})$$

Now  $\langle \lambda = \mu, \Phi, \Psi \rangle$  certifies the  $\{F\} \preceq \{G\}$  relation according to:

$$\Phi_{k(n+\Delta)} \circ G_{k(n+\Delta)}^{\lambda t} \circ \Psi_{kn} = F_{kn}^t$$

□

An example will show us how this procedure can be applied to a small periodic QGCA.

**EXAMPLE 6.3** *If a 2-periodic QGCA  $F = \langle \{0, 1\}, 3, 2, M_0, M_1, S, S \rangle$  has the following behavior for the circuit size  $k = 2$ :*

$$\begin{aligned} F_2(X) &= M_1^{[2]} \cdot S \cdot M_0^{[2]} \cdot S |x_0^0, x_0^1, x_0^2, x_1^0, x_1^1, x_1^2\rangle \\ &= M_1^{[2]} \cdot S \cdot M_0^{[2]} |x_0^1, x_0^2, x_1^0, x_1^1, x_1^2, x_0^0\rangle \\ &= M_1^{[2]} \cdot S (M_0 |x_0^1, x_0^2, x_1^0\rangle \otimes M_0 |x_1^1, x_1^2, x_0^0\rangle) \\ &= M_1^{[2]} \cdot S (|y_0^0, y_0^1, y_0^2, y_1^0, y_1^1, y_1^2\rangle) \\ &= M_1^{[2]} (|y_0^1, y_0^2, y_1^0, y_1^1, y_1^2, y_0^0\rangle) \\ &= M_1 |y_0^1, y_0^2, y_1^0\rangle \otimes M_1 |y_1^1, y_1^2, y_0^0\rangle \\ &= |z_0^0, z_0^1, z_0^2, z_1^0, z_1^1, z_1^2\rangle \\ &= (Z) \end{aligned}$$

Then, with

$$\Psi(x_0^0, x_0^1, x_0^2) = (x_0^0, x_0^1, x_0^2; \ulcorner 0 \urcorner) \quad \text{and} \quad \Phi(x_0^0, x_0^1, x_0^2; x'_0) = (x_0^0, x_0^1, x_0^2)$$

the tuple  $\langle \lambda = 2, \Phi, \Psi \rangle$  certifies the simulation of  $F$  by  $G = \langle \{0, 1\}, 4, M', S \rangle$  as is illustrated by:

$$\begin{aligned} \Phi_8 \circ (M'^{[2]} \cdot S)^2 \circ \Psi_6(X) &= \Phi_8 \circ M'^{[2]} \cdot S \cdot M'^{[2]} \cdot S |x_0^0, x_0^1, x_0^2, \ulcorner 0 \urcorner, x_1^0, x_1^1, x_1^2, \ulcorner 0 \urcorner\rangle \\ &= \Phi_8 \circ M'^{[2]} \cdot S \cdot M'^{[2]} |x_0^1, x_0^2, \ulcorner 0 \urcorner, x_1^0, x_1^1, x_1^2, \ulcorner 0 \urcorner, x_0^0\rangle \\ &= \Phi_8 \circ M'^{[2]} \cdot S (M' |x_0^1, x_0^2, \ulcorner 0 \urcorner, x_1^0\rangle \otimes M' |x_1^1, x_1^2, \ulcorner 0 \urcorner, x_0^0\rangle) \\ &= \Phi_8 \circ M'^{[2]} \cdot S (M_0 |x_0^1, x_0^2, x_1^0\rangle \otimes |\ulcorner 1 \urcorner\rangle \otimes M_0 |x_1^1, x_1^2, x_0^0\rangle \otimes |\ulcorner 1 \urcorner\rangle) \\ &= \Phi_8 \circ M'^{[2]} \cdot S |y_0^0, y_0^1, y_0^2, \ulcorner 1 \urcorner, y_1^0, y_1^1, y_1^2, \ulcorner 1 \urcorner\rangle \\ &= \Phi_8 \circ M'^{[2]} |y_0^1, y_0^2, \ulcorner 1 \urcorner, y_1^0, y_1^1, y_1^2, \ulcorner 1 \urcorner, y_0^0\rangle \\ &= \Phi_8 \circ M' |y_0^1, y_0^2, \ulcorner 1 \urcorner, y_1^0\rangle \otimes M' |y_1^1, y_1^2, \ulcorner 1 \urcorner, y_0^0\rangle \\ &= \Phi_8 \circ M_1 |y_0^1, y_0^2, y_1^0\rangle \otimes |\ulcorner 0 \urcorner\rangle \otimes M_1 |y_1^1, y_1^2, y_0^0\rangle \otimes |\ulcorner 0 \urcorner\rangle \\ &= \Phi_8 |z_0^0, z_0^1, z_0^2, \ulcorner 0 \urcorner, z_1^0, z_1^1, z_1^2, \ulcorner 0 \urcorner\rangle \\ &= |z_0^0, z_0^1, z_0^2, z_1^0, z_1^1, z_1^2\rangle \\ &= (Z) \end{aligned}$$

◇

With the use of above lemmas we can now state the following theorem about the simulation of periodic QGCA.

**THEOREM 6.1** *Every periodic QGCA  $F = \langle Q, n, \mu, M_0, \dots, M_{\mu-1}, P_0, \dots, P_{\mu-1} \rangle$  can be simulated by a regular QGCA  $G = \langle Q, n', M', S \rangle$  with only linear slowdown. This can be abridged by the statement:  $\text{QGCA}_{\text{periodic}} \simeq \text{QGCA}_S$ .*

PROOF. If apply the proof of lemma 6.6 for every neighborhood scheme  $P_i$  we have a periodic QGCA which only uses the shift neighborhood scheme. With lemma 6.7 we can construct a regular QGCA  $G$  which has the same behavior and uses the shift neighborhood scheme.  $\square$

This is a strong equivalence relation within the class of QGCA. We are now able to define our QGCA in terms of periodic QGCA because we know that eventually we can change those into a regular QGCA again.

## 6.4 Every QCA can be simulated by a QGCA

Using the above results we will prove that every well-formed QCA can be simulated by a QGCA with only linear slowdown.

**LEMMA 6.8** *Every proper QCA  $F$  can be simulated by a periodic QGCA  $G$ .*

PROOF. We already know by Lemma 6.5 that we can restrict this proof to the QCA with neighborhood set  $N = \{0, 1\}$ , we therefore assume  $F = \langle Q, \{0, 1\}, f \rangle$ .

Lemma 4.4 shows us that there exist  $a \leq 0 \leq b \in \mathbb{Z}$  and a function  $g : R \rightarrow Q$  with  $R \subset \mathcal{H}_{Q^{b-a+1}}$  which calculates the local inverse of  $F$ . This function obeys:  $\xi \perp \chi$  if  $g(\xi) \neq g(\chi)$  for every  $\xi, \chi \in R$ . The function  $g$  is called the local inverse because with  $x_i \in Q$  and  $|y_i\rangle = f(x_i, x_{i+1})$  the function

$$|y_a, y_{a+1}, \dots, y_{-1}, x_0, y_0, y_1, \dots, y_b\rangle \xleftarrow{g} |y_a, y_{a+1}, \dots, y_{-1}, y_0, y_0, y_1, \dots, y_b\rangle$$

defines a proper transformation which respects the inner product. This proves the well-formedness of the transformation

$$\begin{aligned} |y_a, \dots, y_{-1}, x_0, y_0, y_1, \dots, y_b\rangle &\xrightarrow{f_0} |y_a, \dots, y_0, y_0, y_1, \dots, y_b\rangle \\ &\equiv |y_a, \dots, y_{-1}\rangle \otimes \{\alpha|0, 0\rangle + \beta|1, 1\rangle\} \otimes |y_1, \dots, y_b\rangle \end{aligned}$$

for every  $x_i \in Q$ , and  $f(x_i, x_{i+1}) = |y_i\rangle$  with  $f(x_0, x_1) = |y_0\rangle = \alpha|0\rangle + \beta|1\rangle$ .

We define the following  $A, B$  and  $C$  gates and their ‘requested behavior’ (see Lemma 2.1) with  $x_j \in Q$ ;  $y_j = f(x_j, x_{j+1}) \in \mathcal{H}_Q$  and  $r = b - a + 1$ . The  $A$ -gate calculates the  $r - 1$  values of  $y$  on the left-side:

$$A \left( \bigotimes_{j=0}^{r-1} |x_j, x_j\rangle \right) = \left( \bigotimes_{j=0}^{r-2} |x_j, y_j\rangle \right) \otimes |x_{r-1}, x_{r-1}\rangle$$

The  $B$ -gate calculates the remaining value of  $y$ :

$$B \left( \left( \bigotimes_{j=1}^{r-2} |x_j, y_j\rangle \right) \otimes |x_{r-1}, x_{r-1}\rangle \otimes |x_r, y_r\rangle \right) = \bigotimes_{j=1}^r |x_j, y_j\rangle$$

We use the  $C$ -gates to replace the  $x$  values by the new  $y$  values. Each  $C$  gate replaces one  $x$  value. The well-formedness of this transformation is affirmed by the existence of the inverse function  $g$ . For every  $0 \leq t \leq r - 1$  and  $z_i \in \{x_i, y_i\}$  and  $|y_{t-a}\rangle = \alpha|0\rangle + \beta|1\rangle$ :

$$\begin{aligned} C_t &\left( \left( \bigotimes_{j=t}^{t-a-1} |z_j, y_j\rangle \right) \otimes |x_{t-a}, y_{t-a}\rangle \otimes \left( \bigotimes_{j=t-a+1}^{t+r-1} |z_j, y_j\rangle \right) \right) \\ &= \left( \bigotimes_{j=t}^{t-a-1} |z_j, y_j\rangle \right) \otimes |y_{t-a}, y_{t-a}\rangle \otimes \left( \bigotimes_{j=t-a+1}^{t+r-1} |z_j, y_j\rangle \right) \\ &\equiv \left( \bigotimes_{j=t}^{t-a-1} |z_j, y_j\rangle \right) \otimes \{\alpha|0, 0\rangle + \beta|1, 1\rangle\} \otimes \left( \bigotimes_{j=t-a+1}^{t+r-1} |z_j, y_j\rangle \right) \end{aligned}$$

By the indices of the  $x$  and  $y$  values in the above definition it follows that:

$$\begin{aligned}
& \prod_{t=0}^{r-1} \left( C_t^{[k]} \cdot S^2 \right) \cdot S^{2(r-3)} \cdot B^{[k]} \cdot S^2 \cdot A^{[k]} \cdot S^0 \left( \bigotimes_{j \in \mathbb{Z}_{kr}} |x_j, x_j\rangle \right) \\
&= \prod_{t=0}^{r-1} \left( C_t^{[k]} \cdot S^2 \right) \left( \bigotimes_{j \in \mathbb{Z}_{kr}} |x_{r-2+j}, y_{r-2+j}\rangle \right) \\
&\quad \vdots \\
&= \bigotimes_{j \in \mathbb{Z}_{kr}} |y_j, y_j\rangle \\
&= \bigotimes_{j \in \mathbb{Z}_{kr}} |f(x_i, x_{i+1}), f(x_i, x_{i+1})\rangle
\end{aligned}$$

By Lemma 2.1 and the existence of the function  $g$  we know that  $A$ ,  $B$  and  $C$  are all proper gates (note that all  $C$ -gates are identical). The proper  $(r+2)$ -periodic QGCA

$$G = \langle Q, 2r, r+2, A, B, C_{r-1}, \dots, C_0, S^0, S^2, S^{2(r-2)}, S^2, \dots, S^2 \rangle$$

therefore simulates the QCA  $F$ . This is certified by the encoding  $\langle 1, \Phi : Q^{2r} \rightarrow Q^r, \Psi : Q^r \rightarrow Q^{2r} \rangle$  with  $\Psi(x_0, \dots, x_{r-1}) = (x_0, x_0, x_1, x_1, \dots, x_{r-1}, x_{r-1})$  and  $\Phi = \Psi^{-1}$ .  $\square$

We use this lemma to proof that the class of QCA can be simulated by the the class of QGCA.

**THEOREM 6.2** *Every well-formed QCA  $F$  can be simulated by a QGCA  $G$ .*

PROOF. A well-formed QCA  $F$  can be simulated by a periodic QGCA (see the above lemma) and every such periodic QGCA can be simulated by a QGCA  $G$  (Theorem 6.1). We therefore know that there exists a QGCA  $G$  with  $\{F\} \preceq \{G\}$ .  $\square$

## 6.5 QGCA and PQCA are Equivalent

Because every PQCA is a QCA (which can be simulated by a QGCA), we only have to prove that every QGCA can be converted into a PQCA with the same behavior. Again we will restrict ourselves to the shift-scheme which covers all possible QGCA.

**LEMMA 6.9** *Every proper QGCA  $F$  can be simulated by a PQCA  $G$ .*

PROOF. The QGCA  $F$  is defined by  $\langle Q, n, M, S \rangle$ . We construct a PQCA  $G$  described by

$$G = \langle Q' = Q'_L \times Q'_C \times Q'_R, 3, \{-1, 0, 1\}, g \rangle$$

with  $Q'_L = Q$ ,  $Q'_C = Q^{n-1}$ , and  $Q'_R = \{\text{null}\}$ . The simple transformations are  $\Psi : Q^n \rightarrow Q'$  and  $\Phi : Q' \rightarrow Q^n$ , with  $\Psi(x_0, \dots, x_{n-1}) = x_0 \times x_{1:n-1} \times \text{null}$  and  $\Phi = \Psi^{-1}$ . The local function  $g : (Q')^3 \rightarrow \mathcal{H}_{Q'}$  obeys:

$$g(x, y, z) = g'(z_L \times y_C \times x_R) = M(y_C \times z_L) \times \text{null} \in \mathcal{H}_{Q \times Q^{n-1} \times \{\text{null}\}}$$

with  $x, y, z \in Q'$ . Because  $M$  is a proper gate,  $g'$  will be a unitary transformation.  $G$  is therefore a proper PQCA, which simulates the original QGCA with no slowdown. This is certified by  $\langle 1, \Phi, \Psi \rangle$ ,

as is shown by ( $t = 1$  and  $k \in \mathbb{N}^+$ ):

$$\begin{aligned}
\Phi_k \circ G_k \circ \Psi_{kn} \left( \bigotimes_{j \in \mathbb{Z}_k} \left( \bigotimes_{i \in \mathbb{Z}_n} x_j^i \right) \right) &= \Phi_k \circ G_k \bigotimes_{j \in \mathbb{Z}_k} (x_j^0 \times x_j^1 \times \dots \times x_j^{n-1} \times \text{null}) \\
&= \Phi_k \bigotimes_{j \in \mathbb{Z}_k} (M(x_j^1, \dots, x_j^{n-1}, x_{j+1}^0) \times \text{null}) \\
&= \bigotimes_{j \in \mathbb{Z}_k} M(x_j^1, \dots, x_j^{n-1}, x_{j+1}^0) \\
&= M^{[k]} \cdot S_k \left( \bigotimes_{j \in \mathbb{Z}_k} \left( \bigotimes_{i \in \mathbb{Z}_n} x_j^i \right) \right)
\end{aligned}$$

Because  $\text{QGCA} \simeq \text{QGCA}_S$  we may conclude:  $\text{QGCA} \simeq \text{PQCA}$  □

## 6.6 Conclusion

**THEOREM 6.3**  $\text{QCA} \simeq \text{PQCA} \simeq \text{QGCA}$ .

PROOF. A summary of the results of this chapter gives us:

1.  $\text{QCA} \preceq \text{QGCA}_{\text{periodic}}$  (Lemma 6.8)
2.  $\text{QGCA}_{\text{periodic}} \preceq \text{QGCA}_S$  (Theorem 6.1)
3.  $\text{QGCA}_S \preceq \text{PQCA}$  (Lemma 6.9)
4.  $\text{PQCA} \preceq \text{QCA}$  and  $\text{QGCA} \preceq \text{QCA}$  (Lemma 6.1 and Lemma 6.2)

□

The theory of quantum gates and quantum networks assumes:  $Q = \{0, 1\}$ . For this reason we state the following additional lemma:

**LEMMA 6.10** *Every well-formed QCA  $F = \langle Q, r, N, f \rangle$  can be simulated by a QGCA  $G = \langle \{0, 1\}, n, M, S \rangle$ .*

PROOF. For every QCA  $F$  there exists a QGCA  $F' = \langle Q, n', M', S \rangle$  with  $\{F\} \preceq \{F'\}$ .  $F'$  can be simulated by a QGCA  $F'' = \langle \{0, 1\}, n'', M'', P \rangle$  (lemma 5.1). By theorem 6.1 there exists a QGCA  $G = \langle \{0, 1\}, n, M, S \rangle$  with  $\{F''\} \preceq \{G\}$ . Therefore:  $\{F\} \preceq \{G\}$ . □

This last lemma enables us to use the known results on quantum gates [4, 16, 18, 60] when analyzing QCA, without loss of generality. By doing this, the next section shows us the existence of a universal QCA and answers a question about simulating QCA on Quantum Turing Machines [21].

The above theorems and lemmas also hold when restricted to classical CA. Theorem 6.3 therefore proves (in the case of 1d-CA) the conjecture made by Toffoli and Margolus [55] about the structural invertibility of (classical) reversible CA. An independent proof of this conjecture has been made by Kari [32], which also holds for 2d-CA.

## Chapter 7

# A Universal Quantum Cellular Automaton

In this last chapter we will prove that there exists a QCA which can simulate any other QCA with only a linear time slowdown. This QCA will therefore be called a Universal quantum cellular automaton.

### 7.1 Simulating QCA with QTMs

With the results of the previous chapter, we are now able to solve a well-known problem: “Is it always possible to simulate a QCA on a quantum Turing machine?” The answer will be affirmative.

**LEMMA 7.1** *For every well formed QGCA  $F = \langle \{0, 1\}, n, M, S \rangle$  there exists a well formed QTM  $T = \langle \Sigma, K, \delta \rangle$  which simulates  $F$ . The simulation of the function  $F_k^t$  will have  $\mathcal{O}(\| \cdot \|)$  space complexity and  $\mathcal{O}(\| \cdot \|)$  time complexity.*

PROOF. Let the alphabet set  $\Sigma$  equal  $\{0, 1, \square\}$ . Given the unitary transformation  $M$  on  $\mathcal{H}_{\{0,1\}^n}$ , we first define two QTMs:  $T_S$  and  $T_M$ .

The  $T_S$  simulates the shift neighborhood scheme of the QGCA, it is defined by:

$$|\ulcorner \text{start} \urcorner; \square, x_0^0, x_0^1, \dots, x_{k-1}^{n-1}, \square, \dots \rangle \xrightarrow{*}_{T_S} |\ulcorner \text{halt} \urcorner; \square, \underline{x}_0^1, x_0^2, \dots, x_{k-1}^{n-1}, x_0^0, \square, \dots \rangle$$

The time/space complexity of this well formed QTM will be  $\mathcal{O}(k)$ . The description of  $T_S$  does not depend on  $k$  (we want  $T$  to be valid for any  $k$ ).

The second QTM,  $T_M$ , has to simulate the unitary transformation  $M^{[k]}$  which corresponds to a concatenation of  $k$  transformations  $M$ . Its behavior is described by the following sequence which will be explained afterwards.

$$\begin{aligned} |\ulcorner \text{start} \urcorner; \square, \underline{y}_0^0, y_0^1, \dots, y_{k-1}^{n-1}, \square, \dots \rangle &\xrightarrow{*}_{T_M} |\ulcorner \text{start} \urcorner; \square \rangle \otimes M|y_0\rangle \otimes |\underline{y}_1^0, \dots, \square, \dots \rangle \\ &\xrightarrow{*}_{T_M} |\ulcorner \text{start} \urcorner; \square \rangle \otimes M|y_0\rangle \otimes M|y_1\rangle \otimes |\underline{y}_2^0, \dots, \square, \dots \rangle \\ &\vdots \\ &\xrightarrow{*}_{T_M} |\ulcorner \text{start} \urcorner; \square \rangle \otimes M|y_0\rangle \cdots \otimes M|y_{k-1}\rangle \otimes |\square, \dots \rangle \\ &\xrightarrow{*}_{T_M} |\ulcorner \text{halt} \urcorner; \square \rangle \otimes M|y_0\rangle \cdots \otimes M|y_{k-1}\rangle \otimes |\square, \dots \rangle \end{aligned}$$

This QTM applies  $M$  to the first  $n$  bits of the input string after which the head moves  $n$  places to the right. This process is repeated until the  $\square$ -symbol is read. Now the head returns to the leftmost  $\square$ -symbol and the QTM halts. Because  $M$  is a unitary transformation such a well formed  $T_M$  exists (Lemma 2.3) Again the QTM is independent of  $k$  and has time/space complexity of  $\mathcal{O}(k)$ .

This shows that  $T_M \circ T_S$  simulates one layer of the QGCA  $F$ . By repeating this algorithm  $t$  times we simulate the global function  $F_k^t$  of the QCA  $F$ . Because the time complexity will be  $\mathcal{O}(kt)$ , whereas the space complexity remains  $\mathcal{O}(k)$ , the simulation has linear slowdown.  $\square$

With the use of the above lemma we can prove that every well formed QCA can efficiently be simulated by a well formed QTM, a question raised by Daniel Simon [51], John Watrous[57], and Christoph Dürr *et al.* [20, 21].

**THEOREM 7.1** *Every well formed QCA  $F$  can efficiently be simulated by a well formed QTM  $T$ .*

PROOF. For every QCA  $F$  there exists a QGCA  $G = \langle \{0, 1\}, n, M, S \rangle$  which simulates  $F$  with linear slowdown (Lemma 6.10). Therefore, by Lemma 7.1, there exists a well formed QTM  $T$  which simulates  $F_k^t$  with space-complexity  $\mathcal{O}(k)$  and time-complexity  $\mathcal{O}(kt)$ .  $\square$

## 7.2 A Universal Quantum Cellular Automaton

By now we have proven that a QTM can simulate a QCA and vice versa. We also know that there exists a universal QTM. This leads us to the conclusion that there exists a QCA that can simulate any other well-formed QCA. Still this is not the end of our investigations. The disadvantage of the above described construction is that we do not use the parallel computational power of a QCA when it has to simulate another parallel QCA. This is obviously not an efficient way of simulating. Here it will be shown that it is indeed possible to construct a QCA which can mimic the behavior of any other QCA *in parallel*.

Our approach will be to use the known results about quantum circuits and universal quantum gates to construct a QGCA that we can ‘program’ with the input to simulate other possible QGCA. Because there are uncountable many QGCA we have to be satisfied with a simulation within a bounded error. This resembles the situation with the universal quantum gate. We will use this universal two bit gate  $U$  to approximate the  $M$  gate of a QGCA  $\langle \{0, 1\}, n, M, S \rangle$ . The resulting circuit after this transformation will be periodic both in the space and time direction. The universal QCA  $\mathcal{U}$  will be a QCA which can be programmed to simulate any such circuit.

### 7.2.1 Periodic Universal Gate Arrays

Our aim is to simulate a QGCA  $G = \langle \{0, 1\}, n, M, S \rangle$ . For the simulation of the quantum gate  $M$  we will use a universal two bit gate  $U$ . A circuit that only consists of  $U$  gates will be called a *U gate array*. The first transformational step will be to replace every  $M$  gate with a  $U$  gate array which approximates the initial gate  $M$ . Because of the symmetrical structure of QGCA, the resulting  $U$  gate array will be periodic in both space and time dimensions. Such a circuit will there be called a *periodic U gate array*. See Figure 7.1 for an example. In order to translate the ‘wiring’ of this array, another model will be introduced.

### 7.2.2 Simulating the Wiring

We proceed by defining two gates  $I$  and  $X$  that will mimic the wiring of a periodic  $U$  gate array. The gate  $I$  corresponds with the two-qubit identity gate whereas  $X$  describes a cross-over:

$$I|x, y\rangle = |x, y\rangle \quad \text{and} \quad X|x, y\rangle = |y, x\rangle$$

for every  $x, y \in \{0, 1\}$ . By combining the  $I$ ,  $X$  and  $U$  gates in a wall-like, periodic structure, we can mimic any periodic  $U$  gate array. An of example such an *periodic IXU array* of this is shown in Figure 7.2. To describe these structures we use the following definition.

**DEFINITION 7.1 (periodic IXU array)** *A periodic IXU array  $F$  is defined by  $\langle m, n, A_j^i, B_j^i \rangle$  with  $A_j^i, B_j^i \in \{I, X, U\}$  for every  $j \in \mathbb{Z}_m$  and  $i \in \mathbb{Z}_n$ .*

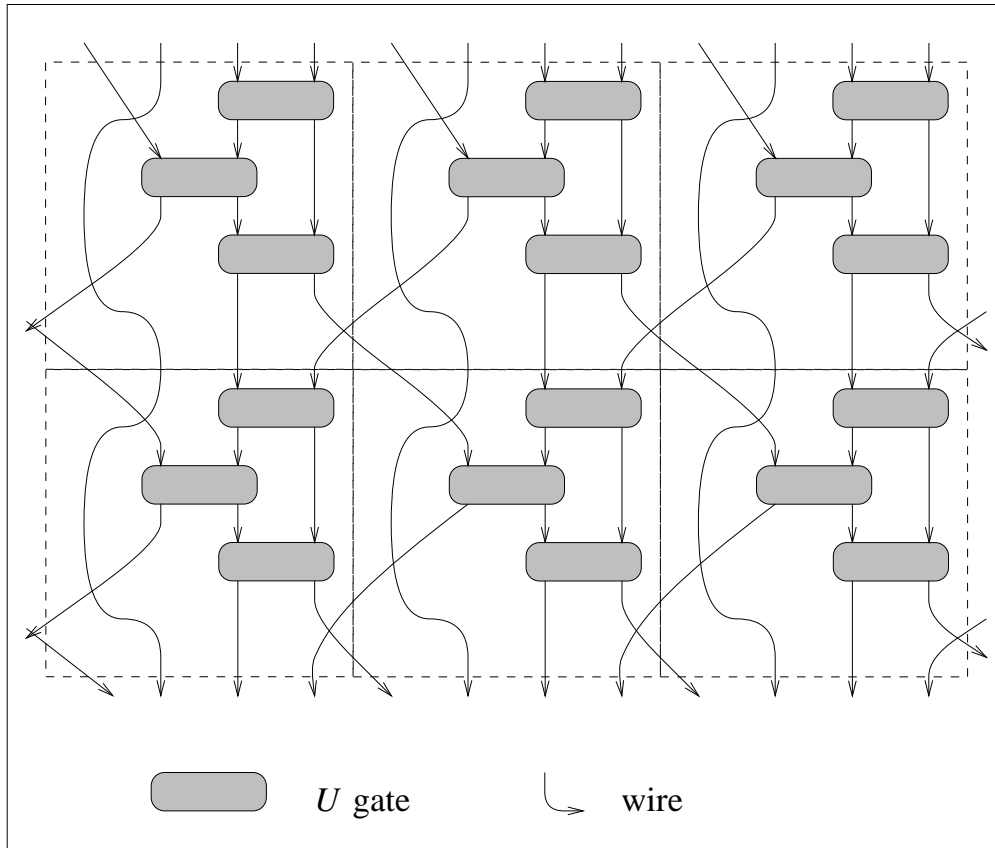


Figure 7.1: Part of a periodic  $U$  gate array which simulates a QGCA. Each box contains a  $U$  gate array that simulates a gate of the QGCA. The part shown here contains two layers of three original gates. The periodic pattern of the resulting  $U$  gate array is apparent.

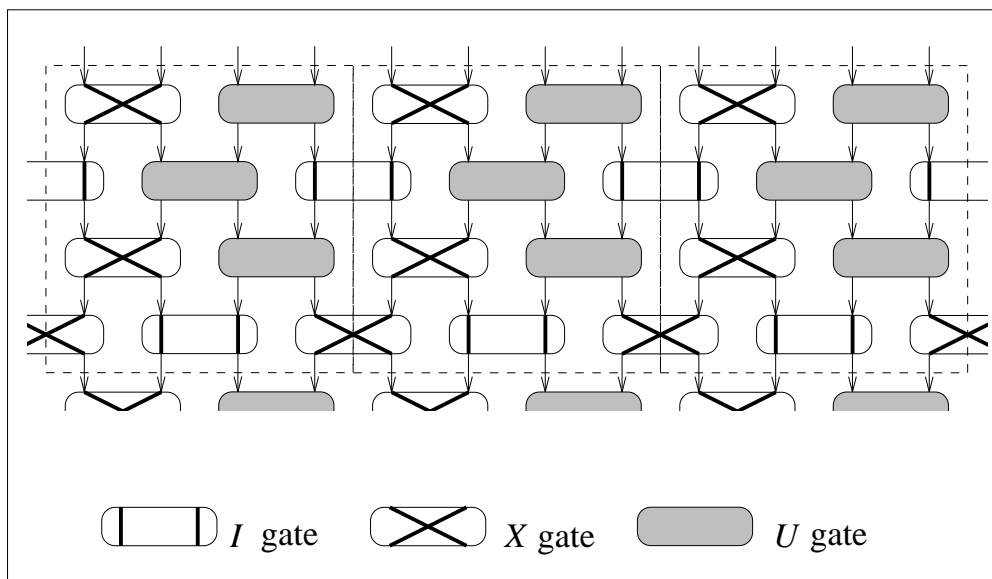


Figure 7.2: Example of a periodic  $IXU$  array. The indicated areas correspond to the boxes of the periodic  $U$  gate array of Figure 7.1. This figure shows the simulation of one layer of three gates of the original QGCA.



A periodic  $IXU$  array is build by identical boxes of  $IXU$  arrays. The number  $m$  indicates the horizontal size (in terms of gates) of this box whereas  $n$  defines the number of *double layers* in each box. Each double layer consists of an  $A$  layer and a  $B$  layer. The function  $F_{2mk}$  describes the behavior of a row  $\mathbb{Z}_k$  of  $k$  boxes which acts on  $2mk$  bits and uses  $2n$  time steps (measured in individual layers). This function is defined by:

$$F_{2mk} = \prod_{i=n-1}^0 \left[ S^\dagger \cdot \left( \bigotimes_{j \in \mathbb{Z}_{mk}} B_{j \bmod m}^i \right) \cdot S \cdot \left( \bigotimes_{j \in \mathbb{Z}_{mk}} A_{j \bmod m}^i \right) \right]$$

**EXAMPLE 7.1** The periodic  $IXU$  array of Figure 7.2 is defined by  $\langle 2, 2, A_j^i, B_j^i \rangle$  with

$$\begin{array}{ll} A_0^0 = X & A_1^0 = U \\ B_0^0 = U & B_1^0 = I \\ A_0^1 = X & A_1^1 = U \\ B_0^1 = I & B_1^1 = X \end{array}$$

◇

Without going into full detail we can state the following lemma.

**LEMMA 7.2** For every QGCA  $F = \langle \{0, 1\}, n, M, S \rangle$  there exists a periodic  $IXU$  array  $G = \langle n, m, A_j^i, B_j^i \rangle$  (with  $\gcd(m, n) = 1$ ) which simulates  $F$ . This simulation is within an arbitrary small error.

PROOF. Use the above described transformation from QGCA to periodic  $U$  gate arrays to periodic  $IXU$  arrays. The  $\gcd(m, n) = 1$  constraint is satisfied by adding a sufficient number of identity-layers which only consist of  $I$ -gates, thereby increasing the value of  $n$  until the desired number is reached. □

This lemma tells us that for every  $k \in \mathbb{N}^+$  the function  $G_{2mk}$  will simulate  $F_k$ . If  $\varepsilon$  indicates the allowed error for simulating one  $M$  gate, the simulation of  $F_k$  will have an error of  $k\varepsilon$ . Because  $\varepsilon$  can be made arbitrary small,  $k\varepsilon$  can be made arbitrary small for a fixed value of  $k$ . The linear slowdown is expressed by the ‘vertical size’ of each block which equals  $2n$ . If the simulation of the  $M$  gate requires some additional work bits for the  $U$  gate array, the space complexity will grow with a factor proportional to  $k$ . The reason for the  $\gcd(m, n) = 1$  constraint will become evident in the proof in Appendix D.

If we express the cell values of the periodic  $IXU$  array at time  $t$  by  $|x_0^t, y_0^t, \dots, x_{km-1}^t, y_{km-1}^t\rangle$ , the general behavior of the  $IXU$  array is calibrated by the equations:

$$A_{j \bmod m}^{i \bmod n} |x_j^{2i}, y_j^{2i}\rangle = |x_j^{2i+1}, y_j^{2i+1}\rangle \quad (7.1)$$

$$B_{j \bmod m}^{i \bmod n} |y_j^{2i+1}, x_{j+1}^{2i+1}\rangle = |y_j^{2i+2}, x_{j+1}^{2i+2}\rangle \quad (7.2)$$

for every  $i \in \mathbb{N}$  and  $j \in \mathbb{Z}_{km}$ . We will use this ‘calibration’ to prove that the universal automaton  $\mathcal{U}$  embeds the same behavior.

### 7.2.3 The Universal Quantum Cellular Automaton

The universal quantum cellular automaton  $\mathcal{U}$  is described by the following definition (an explanation of its semantics will be given below):

**DEFINITION 7.2 ( $\mathcal{U}$  automaton)** The  $\mathcal{U}$  automaton is a QGCA  $\langle Q, n, M, P \rangle$  with:

**state set**  $Q = \{0, 1, \triangleleft, \square, \lceil I \rceil, \lceil X \rceil, \lceil U \rceil\}$  and  $n = 4$

**quantum gate**  $M : \mathcal{H}_{Q^4} \rightarrow \mathcal{H}_{Q^4}$  that obeys:

$$\begin{aligned} M|\square, \square, \ulcorner A \urcorner, \square\rangle &= |\square, \square, \ulcorner A \urcorner, \square\rangle & M|\square, x, \ulcorner A \urcorner, \square\rangle &= |\square, x, \ulcorner A \urcorner, \square\rangle \\ M|\square, \square, \ulcorner A \urcorner, x\rangle &= |x, \square, \ulcorner A \urcorner, \square\rangle & M|\square, \square, \ulcorner A \urcorner, \triangleleft\rangle &= |\triangleleft, \square, \ulcorner A \urcorner, \square\rangle \\ M|x, \square, \ulcorner A \urcorner, \triangleleft\rangle &= |\triangleleft, x, \ulcorner A \urcorner, \square\rangle & M|\square, x, \ulcorner A \urcorner, y\rangle &= |x', \square, \ulcorner A \urcorner, y'\rangle \end{aligned}$$

for every  $x, y \in \{0, 1\}$  and for every  $A \in \{I, X, U\}$ :

$$A|x, y\rangle = |x', y'\rangle$$

**neighborhood-scheme**  $P = \langle n, \sigma, \phi \rangle$  defined by

$$\begin{array}{cccc} \sigma(0) = 3 & \sigma(1) = 1 & \sigma(2) = 2 & \sigma(3) = 0 \\ \phi(0) = -1 & \phi(1) = 0 & \phi(2) = 0 & \phi(3) = 1 \end{array}$$

If we supply this automaton with the appropriate input, it will simulate a periodic  $IXU$  array. This is shown in the following analysis.

For every  $N, K \in \mathbb{N}^+$  the input:

$$\left| \left[ \square x_J^0 \pi_J^0 \square, y_J^0 \pi_J^1 \square, \square \square \pi_J^2 \square, \triangleleft \square \pi_J^3 \square, [\square \square \pi_J^i \square]_{i=4}^{2N-1} \right]_{J \in \mathbb{Z}_K} \right\rangle$$

with  $\pi \in \{\ulcorner I \urcorner, \ulcorner X \urcorner, \ulcorner U \urcorner\}$  and  $x, y \in \{0, 1\}$ , will evolve after two layers of the automaton  $\mathcal{U}_{2NK}$  to the configuration:

$$\left| \left[ \square \square \pi_J^0 \square, \triangleleft y_J^1 \pi_J^1 \square, [\square \square \pi_J^i \square]_{i=2}^{2N-2}, x_{J+1}^1 \square \pi_J^{2N-1} \square \right]_{J \in \mathbb{Z}_K} \right\rangle$$

with  $\Pi_J^0 |x_J^0, y_J^0\rangle = |x_J^1, x_J^1\rangle$  (the gate  $\Pi$  is determined by  $\pi_J^0 = \ulcorner \Pi_J^0 \urcorner$ ). At this point the  $x$ -values will travel step-by-step to the left, while the  $y$ -values have moved one place to the right and are now stationary (this is established by the “ $\triangleleft$ ”-symbol).

The next important situation occurs after  $\Theta(N)$  layers when the  $x$  and  $y$  values ‘meet again’ at the  $\pi_J^1$  gates, which are then to be simulated. This will result in the configuration:

$$\left| \left[ y_J^2 \square \pi_J^0 \square, \square \square \pi_J^1 \square, \triangleleft x_{J+1}^2 \pi_J^2 \square, [\square \square \pi_J^i \square]_{i=2}^{2N-1} \right]_{J \in \mathbb{Z}_K} \right\rangle$$

with  $\Pi_J^1 |y_J^1, x_{J+1}^1\rangle = |y_J^2, x_{J+1}^2\rangle$  and  $\pi_J^1 = \ulcorner \Pi_J^1 \urcorner$ .

By now the  $y$ -values go left in order to collide at the  $\pi_J^2$  gates with the  $x$ -variables. This process will be repeated in the obvious way.

A careful examination of the behavior of  $\mathcal{U}_{2NK}$  on this input reveals that the overall time evolution is calibrated by the equations:

$$\Pi_{J+\lfloor \frac{2i}{2N} \rfloor \bmod K}^{2i \bmod 2N} |x_{J+i}^{2i}, y_{J+i}^{2i}\rangle = |x_{J+i}^{2i+1}, y_{J+i}^{2i+1}\rangle \quad (7.3)$$

$$\Pi_{J+\lfloor \frac{2i+1}{2N} \rfloor \bmod K}^{2i+1 \bmod 2N} |y_{J+i}^{2i+1}, x_{J+i+1}^{2i+1}\rangle = |y_{J+i}^{2i+2}, y_{J+i+1}^{2i+2}\rangle \quad (7.4)$$

for every  $i \in \mathbb{N}$ ,  $J \in \mathbb{Z}_K$ , and  $\pi_i = \ulcorner \Pi_i \urcorner$ .

By using the calibrations 7.1, 7.2, 7.3, and 7.4, we can prove the following lemma.

**LEMMA 7.3** Every periodic  $IXU$  array  $F = \langle m, n, A_j^i, B_j^i \rangle$  with  $\gcd(m, n) = 1$  can be simulated by the QGCA  $\mathcal{U}$ . The time-complexity of the simulation is linear and does not depend on the size  $k$  of the periodic  $IXU$  array  $F_{2mk}$ .

PROOF. If a ‘mapping’ between the four calibrations is possible, it follows that the two automata have the same evolution. Because  $\gcd(m, n) = 1$ , there exists a positive integer  $\alpha$  such that

$\alpha n \bmod m = 1$ . With  $N = \alpha n$  and  $K = km$  every  $F_{2mk}$  can be simulated by  $\mathcal{U}_{2NK}$ . This is shown by the existence of a mapping which satisfies (assuming  $J = j - i$ ):

$$A_{j \bmod m}^{i \bmod n} = \Pi_{j-i+\lfloor \frac{2i}{2N} \rfloor \bmod K}^{2i \bmod 2N} \quad B_{j \bmod m}^{i \bmod n} = \Pi_{j-i+\lfloor \frac{2i+1}{2N} \rfloor \bmod K}^{2i+1 \bmod 2N}$$

for every  $i \in \mathbb{N}$  and  $j \in \mathbb{Z}$ . See Appendix D for the existence proof of such a mapping. Because  $\alpha$  does not depend on  $k$ , the time-complexity is bounded by  $\Theta(Ni) = \Theta(i)$  for the simulation of  $2i$  layers of  $F$ .  $\square$

With this lemma we have reached the final conclusion of this thesis.

**THEOREM 7.2** *There exists a proper universal quantum cellular automaton  $\mathcal{U}$  which can simulate any well-formed QCA  $F$  with a bounded error. The extra costs of the simulation of  $F_k^t$  has time-complexity  $\Theta(t)$  and space-complexity  $\Theta(k)$ .*

PROOF. This follows from the results of Lemma 6.10, Lemma 7.2 and Lemma 7.3.  $\square$

## 7.3 Conclusions

We have defined a class of well-formed quantum cellular automata which are one-dimensional and circular bounded. It has been shown that every proper QCA resembles a periodic quantum gate array. This assures us that the *global unitarity* of a QCA can be reduced to the *local unitarity* of the gates of a QGCA. There exists a QCA  $\mathcal{U}$  which can simulate any such array (by using a universal two-qubit gate  $U$ ). The time-complexity of this simulation does not depend on the size  $k$  of the initial QCA. This proves that  $\mathcal{U}$  is a *universal quantum cellular automaton*.

The existence of a *classical* universal cellular automata was shown by Jürgen Albert *et al.* [1] and Bruno Martin [37]. Both constructions did applied to irreversible CA. Because the classical reversible cellular automata (RCA) are a subset of QCA, we know that every RCA can be simulated by the *non-classical*  $\mathcal{U}$  automaton. If we want a universal RCA, we have to adapt Definition 7.2 in such a way that it uses a universal *three-bit* gate (the Toffoli-gate for example). This is because there does not exist a classical two-bit gate  $U$  that is universal in its computational power [3, 4, 16, 18]. The author is unknown of an earlier construction of such an automaton.

Theorem 7.2 shows that the model of QCA is a useful model of parallel quantum computing on its own. Because of their uniform structure, cellular automata are particular useful to bridge the gap between physics and computational theory. From this intermediate standpoint there are at least two directions possible, each with its own merits. By going from QCA to physics one obtains a powerful model to describe quantum mechanical systems. Recent work by David Meyer [39, 40, 41, 42] goes along this pathway.

The other direction is to try to actually *construct* a controllable QCA. Several authors have suggested that QCA-like systems are more likely to be build than quantum Turing machines oriented structures [8, 34, 35, 36]. If such a construction would indeed be possible in the future, we would have equipped ourselves with a new remarkable tool. A tool whose computational power we are just beginning to unravel [11, 22, 56].

# Appendix A

## Unitary Transformations

Because of their central role in quantum computing, we will summarize the properties of unitary transformations [13, 25, 33, 47]. A linear transformation  $F$  is determined by its state set  $S$  and the corresponding transformational matrix  $M_F \in \mathbb{C}^{S \times S}$ . We will only look at countable state sets.

**DEFINITION A.1** *A transformation  $F$  on  $S$  is unitary if and only if its corresponding matrix  $M_F$  obeys:  $M_F^\dagger M_F = M_F M_F^\dagger = 1_S$ , with  $1_S$  the identity on the set  $S$ .*

### A.1 Finite dimensional Transformations

If  $M_F$  is finite dimensional (e.g.  $S$  is finite)  $M_F^\dagger M_F = 1_S$  always implies  $M_F M_F^\dagger = 1_S$ . We therefore can simplify the above definition.

**DEFINITION A.2** *A finite dimensional transformation  $F$  on  $S$  is unitary if and only if its corresponding matrix  $M_F$  obeys:  $M_F^\dagger M_F = 1_S$ , with  $1_S$  the identity on the set  $S$ .*

### A.2 Infinite dimensional Transformations

If we take the the infinite transformation  $F(i) = i + 1$  on  $S = \mathbb{N}$  we obtain the following ‘inverse’  $F^\dagger : \mathbb{N}^+ \rightarrow \mathbb{N}$ , with  $F^\dagger(i+1) = i$  for every  $i \in \mathbb{N}$ . We therefore have:  $F^\dagger \circ F = I$  but not  $F \circ F^\dagger = I$  ( $F \circ F^\dagger(0)$  is not defined). In matrix notation this is illustrated by:

$$M_F^\dagger \cdot M_F = \begin{pmatrix} 0 & 1 & 0 & \cdots \\ 0 & 0 & 1 & \cdots \\ 0 & 0 & 0 & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{pmatrix} \cdot \begin{pmatrix} 0 & 0 & 0 & \cdots \\ 1 & 0 & 0 & \cdots \\ 0 & 1 & 0 & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & \cdots \\ 0 & 1 & 0 & \cdots \\ 0 & 0 & 1 & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{pmatrix}$$

but:

$$M_F \cdot M_F^\dagger = \begin{pmatrix} 0 & 0 & 0 & \cdots \\ 1 & 0 & 0 & \cdots \\ 0 & 1 & 0 & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{pmatrix} \cdot \begin{pmatrix} 0 & 1 & 0 & \cdots \\ 0 & 0 & 1 & \cdots \\ 0 & 0 & 0 & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 & \cdots \\ 0 & 1 & 0 & \cdots \\ 0 & 0 & 1 & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{pmatrix}$$

A closer look at  $F$  reveals that  $F$  could not be unitary because  $F$  is not surjective. This is a necessary and sufficient restriction for infinite transformations with  $M_F^\dagger M_F = 1_S$  to be unitary:

**DEFINITION A.3** *An infinite dimensional transformation  $F$  on  $S$  is unitary if and only if its corresponding matrix  $M_F$  obeys:  $M_F^\dagger M_F = 1_S$  (with  $1_S$  the identity on the set  $S$ ) and  $F$  is surjective.*

### A.3 Exponential Expressions

**DEFINITION A.4** Let  $H$  be a square matrix, by  $e^H$  we mean  $I + H + \frac{1}{2}H^2 + \dots$ . Therefore for every  $t \in \mathbb{C}$ :

$$\exp(t \cdot H) = \sum_{k=0}^{\infty} \frac{1}{k!} \cdot t^k \cdot H^k$$

**EXAMPLE A.1**

$$\exp\left(t \cdot \begin{pmatrix} 0 & i \\ i & 0 \end{pmatrix}\right) = \begin{pmatrix} \cos(t) & i \sin(t) \\ i \sin(t) & \cos(t) \end{pmatrix}$$

◇

**DEFINITION A.5** A matrix  $A$  is called hermitian if and only if  $A^\dagger = A$ .

Finite unitary and hermitian matrix correspond to each other in the following way.

1. For every hermitian matrix  $H$ ,  $\exp(iH)$  is a unitary matrix.
2. For every unitary matrix  $U$ , there exists a hermitian matrix  $H$  such that:  $U = \exp(iH)$  (in general this  $H$  is not unique).

For this reason the analogy of hermitian matrices to real numbers and unitary matrices to the complex numbers with norm one is made (except for commutativity of multiplication).

## Appendix B

# Proving Well-Formedness

In this appendix we will prove that if the general function  $F_{\mathbb{Z}}$  of a normalized QCA is not well-formed, there exists a  $K \in \mathbb{N}^+$  such that  $F_k$  is not unitary. This result is used in Lemma 4.3.

**LEMMA B.1** *Let  $F = \langle Q, N, f \rangle$  a normalized QCA. If there exist two values  $X \neq Y \in Q^{\mathbb{Z}}$  such that*

$$\langle F_{\mathbb{Z}}(X)_i, F_{\mathbb{Z}}(Y)_i \rangle \neq 0$$

for every  $i \in \mathbb{Z}$ , then there exists a  $k \in \mathbb{N}^+$  with  $F_k$  not unitary.

PROOF. We use  $x_i$  and  $y_i \in \mathcal{H}_Q$  defined by

$$F_{\mathbb{Z}}(X) = F_{\mathbb{Z}}\left(\bigotimes_{i \in \mathbb{Z}} X_i\right) = \bigotimes_{i \in \mathbb{Z}} x_i \quad \text{and} \quad F_{\mathbb{Z}}(Y) = F_{\mathbb{Z}}\left(\bigotimes_{i \in \mathbb{Z}} Y_i\right) = \bigotimes_{i \in \mathbb{Z}} y_i$$

with  $X_i, Y_i \in Q$  and  $X_0 \neq Y_0$  such that

$$\text{for every } i \in \mathbb{Z} : \quad \langle x_i, y_i \rangle \neq 0 \tag{B.1}$$

We will now prove that, given these  $X$  and  $Y$ , we are able to construct a  $k \in \mathbb{N}^+$  and a  $X', Y' \in Q^k$ , with  $X' \perp Y'$  and  $F_k|X' \rangle \not\perp F_k|Y' \rangle$ .

Define  $r = |N|$  the size of the normalized neighborhood  $N = \{0, 1, \dots, r-1\}$ . Take  $a, b, c$  and  $d$  (with  $a < b \leq 0 \leq c < d$ ), such that:

$$\begin{aligned} X_{a:(a+r-1)} &= X_{b:(b+r-1)} & Y_{a:(a+r-1)} &= Y_{b:(b+r-1)} \\ X_{c:(c+r-1)} &= X_{d:(d+r-1)} & Y_{c:(c+r-1)} &= Y_{d:(d+r-1)} \end{aligned}$$

Because there are ‘only’  $|Q|^{2r}$  different combinations of  $[X_{i:(i+r-1)}; Y_{i:(i+r-1)}]$ , this is always possible for  $a \geq -|Q|^{2r}$  and  $d \leq |Q|^{2r}$ . With these values we define:

$$\begin{aligned} X_L &= X_{a:(b-1)} & X_R &= X_{c:(d-1)} \\ Y_L &= Y_{a:(b-1)} & Y_R &= Y_{c:(d-1)} \end{aligned}$$

This can be visualized by:

$$\begin{array}{ccccccc} X & = & \cdots & \underbrace{X_a \otimes \cdots \otimes X_{b-1}}_{X_L} & \otimes X_b \otimes \cdots \otimes X_0 \otimes \cdots \otimes & \underbrace{X_c \otimes \cdots \otimes X_{d-1}}_{X_R} & \otimes X_d \cdots \\ Y & = & \cdots & \underbrace{Y_a \otimes \cdots \otimes Y_{b-1}}_{Y_L} & \otimes Y_b \otimes \cdots \otimes Y_0 \otimes \cdots \otimes & \underbrace{Y_c \otimes \cdots \otimes Y_{d-1}}_{Y_R} & \otimes Y_d \cdots \end{array}$$

Now we distinguish the following three possibilities:

1.  $X_L \neq Y_L$ : Take  $X' = X_L, Y' = Y_L$  both elements of  $Q^k$  with  $k = b - a$  (thus  $X' \perp Y'$ ). For  $0 \leq i \leq r - 1$  we have:

$$X'_{k+i} = X'_i = X_{a+i} = X_{b+i}$$

and therefore:

$$\begin{aligned} F_k(X') &= \bigotimes_{j \in \mathbb{Z}_k} f \left( \bigotimes_{i \in \mathbb{Z}_r} X'_{j+i} \right) \\ &= \bigotimes_{j \in \mathbb{Z}_k} f \left( \bigotimes_{i \in \mathbb{Z}_r} X_{a+j+i} \right) \\ &= (F_{\mathbb{Z}}(X))_{a:(b-1)} \end{aligned}$$

and for the same reasons also:  $F_k(Y') = (F_{\mathbb{Z}}(Y))_{a:(b-1)}$ . This leads us to:

$$\langle F_k(X'), F_k(Y') \rangle = \left\langle (F_{\mathbb{Z}}(X))_{a:(b-1)}, (F_{\mathbb{Z}}(Y))_{a:(b-1)} \right\rangle = \prod_{i=a}^{b-1} \langle x_i, y_i \rangle$$

By (B.1) we now know that  $F_k(X') \not\perp F_k(Y')$ , which shows that  $F_k$  is not a unitary transformation (with  $k \leq |Q|^{2r}$ ).

2.  $X_R \neq Y_R$ : With  $X' = X_R, Y' = Y_R$  and  $k = d - c$ , the same reasoning as with  $X_L \neq Y_L$  holds (again  $k \leq |Q|^{2r}$ ).
3.  $X_L = Y_L$  and  $X_R = Y_R$ : First we will prove  $X_{a:(b+r-1)} = Y_{a:(b+r-1)}$ . We already know:  $X_{a:(b-1)} = Y_{a:(b-1)}$ . With induction (given  $0 \leq i \leq r - 1$  and  $X_{a:(b+i-1)} = Y_{a:(b+i-1)}$ ) it follows that  $X_{b+i} = X_{a+i} = Y_{a+i} = Y_{b+i}$  holds. Likewise we can prove:  $X_{c:(d+r-1)} = Y_{c:(d+r-1)}$ . Now take  $X' = X_{b:(c+r-1)}$  and  $Y' = Y_{b:(c+r-1)} \in Q^k$ , with  $k = c + r - b$ . Because  $b \leq 0 \leq c + r - 1$  and  $X_0 \neq Y_0$ , we have  $X' \perp Y'$ . Below we will use: (because  $X_{b:(b+r-1)} = Y_{b:(b+r-1)}$  and  $X_{c:(c+r-1)} = Y_{c:(c+r-1)}$ ):

$$\text{for every } 0 \leq i \leq r - 1 : \quad X'_{c-b+i} = X_{c+i} = Y_{c+i} = Y'_{c-b+i}$$

and also for every  $r \leq i \leq 2r - 2$ :

$$X'_{c-b+i} = X'_{k+i-r} = X'_{i-r} = X_{b+i-r} = Y_{b+i-r} = Y'_{i-r} = Y'_{k+i-r} = Y'_{c-b+i}$$

To summarize:  $X'_{c-b+i} = Y'_{c-b+i}$ ; if  $0 \leq i \leq 2r - 2$ . This gives us:

$$\begin{aligned} F_k(X') &= \bigotimes_{j \in \mathbb{Z}_k} f \left( \bigotimes_{i \in \mathbb{Z}_r} X'_{j+i} \right) \\ &= \left( \bigotimes_{j=0}^{c-b-1} f \left( \bigotimes_{i \in \mathbb{Z}_r} X'_{j+i} \right) \right) \otimes \left( \bigotimes_{j=c-b}^{k-1} f \left( \bigotimes_{i \in \mathbb{Z}_r} X'_{j+i} \right) \right) \\ &= \left( \bigotimes_{j=b}^{c-1} f \left( \bigotimes_{i \in \mathbb{Z}_r} X_{j+i} \right) \right) \otimes \left( \bigotimes_{j=c-b}^{k-1} f \left( \bigotimes_{i \in \mathbb{Z}_r} Y'_{j+i} \right) \right) \end{aligned}$$

If we combine this with:

$$F_k(Y') = \left( \bigotimes_{j=b}^{c-1} f \left( \bigotimes_{i \in \mathbb{Z}_r} Y_{j+i} \right) \right) \otimes \left( \bigotimes_{j=c-b}^{k-1} f \left( \bigotimes_{i \in \mathbb{Z}_r} Y'_{j+i} \right) \right)$$

the inner-product of  $F_k(X')$  and  $F_k(Y')$  equals:

$$\begin{aligned}
\langle F_k(X'), F_k(Y') \rangle &= \left\langle \left( \bigotimes_{j=b}^{c-1} f \left( \bigotimes_{i \in \mathbb{Z}_r} X_{j+i} \right) \right), \left( \bigotimes_{j=b}^{c-1} f \left( \bigotimes_{i \in \mathbb{Z}_r} Y_{j+i} \right) \right) \right\rangle \\
&= \left\langle (F_{\mathbb{Z}}(X))_{b:(c-1)}, (F_{\mathbb{Z}}(Y))_{b:(c-1)} \right\rangle \\
&= \prod_{i=b}^{c-1} \langle x_i, y_i \rangle
\end{aligned}$$

Again, (B.1) shows us that  $F_k(X') \not\perp F_k(Y')$ , which shows that  $F_k$  is not a unitary transformation (with  $k \leq 2|Q|^{2r} + r$ ).

All possibilities are covered by 1, 2 and 3. This proves that there exists a  $k \leq 2|Q|^{2r} + r$ , for which  $F_k$  is not unitary: the QCA  $F$  is not well-formed as defined in (4.7).  $\square$



## Appendix C

# Simulating Neighborhood Schemes

In this appendix we give the proof of Lemma 6.6 about the simulation of arbitrary neighborhood schemes by periodic QGCA which only use the Shift neighborhood scheme.

**LEMMA C.1** *For every QGCA  $F = \langle Q, n, M, P \rangle$  there exists a periodic QGCA  $G$  defined by*

$$G = \langle Q, n, \mu, M_0, \dots, M_{\mu-1}, S, \dots, S \rangle$$

*which is shift equivalent with  $F$ .*

PROOF. Given the neighborhood scheme  $P = \langle n, \sigma, \phi \rangle$ , we will use the labels:

- $\Phi = \{\phi(i) \mid i \in \mathbb{Z}_n\}$
- $r = \max(\Phi) - \min(\Phi)$
- $\psi_i = \{\sigma(t) \mid t \in \mathbb{Z}_n \text{ and } \phi(t) = \min(\Phi) + i\}$  for every  $0 \leq i \leq r$
- $m(P) = \sum_{i=0}^r i \cdot |\psi_i|$

(Therefore if  $m(P) = 0$  we have  $\psi_0 = \mathbb{Z}_n$  and  $m(P) = 1$  implies  $|\psi_0| = n - 1$  and  $|\psi_1| = 1$ .) For every neighborhood scheme  $P = \langle n, \sigma, \phi \rangle$  and value  $c \in \mathbb{Z}_n$ , we define the following standard procedure:

Construct an alternative neighborhood scheme  $R = \langle n, \varsigma, \varphi \rangle$  by

$$\varsigma(i) = \sigma(i) \quad \text{and} \quad \varphi(i) = \phi(i) - 1_c(i)$$

Next, we define two 'permutation gates'  $\Pi : \mathbb{Z}_n \rightarrow \mathbb{Z}_n$  and  $\pi : \mathbb{Z}_n \rightarrow \mathbb{Z}_n$  by:

$$\Pi(i) = \begin{cases} n-1 & \text{if } i = c \\ c & \text{if } i = n-1 \\ i & \text{otherwise} \end{cases}$$

and  $\pi(i) = \Pi(i-1)$  (such that  $\pi(\Pi(i)+1) = i$  for every  $i \in \mathbb{Z}_n$ ). We now have for every  $k \in \mathbb{N}^+$ :

$$\begin{aligned} \Pi^{[k]} \circ S_k \circ \pi^{[k]} \circ R_k &= \langle n, \Pi(i), 0 \rangle \circ \langle n, i+1, 1_{n-1}(i) \rangle \circ \langle n, \pi(i), 0 \rangle \circ \langle n, \varsigma(i), \varphi(i) \rangle \\ &= \langle n, \Pi(i), 0 \rangle \circ \langle n, i+1, 1_{n-1}(i) \rangle \circ \langle n, \varsigma \circ \pi(i), \varphi \circ \pi(i) \rangle \\ &= \langle n, \Pi(i), 0 \rangle \circ \langle n, \varsigma \circ \pi(i+1), 1_{n-1}(i) + \varphi \circ \pi(i+1) \rangle \\ &= \langle n, \varsigma \circ \pi(\Pi(i)+1), 1_{n-1} \circ \Pi(i) + \varphi \circ \pi(\Pi(i)+1) \rangle \\ &= \langle n, \sigma(i), \phi(i) \rangle \\ &= P_k \end{aligned}$$

This shows that the  $\mu+1$ -periodic QGCA  $G' = \langle Q, n, \mu+1, \pi, M_0 \circ \Pi, \dots, M_{\mu-1}, R, S, \dots, S \rangle$  is identical with the  $\mu$ -periodic QGCA  $G = \langle Q, n, \mu, M_0, \dots, M_{\mu-1}, P, S, \dots, S \rangle$ , and thus  $G = G'$ .

We will use this procedure to ‘transform’ the QGCA  $F$  into the desired periodic QGCA. For every  $\mu$ -periodic QGCA  $G = \langle Q, n, \mu, M_0, \dots, M_{\mu-1}, P, S, \dots, S \rangle$ , we distinguish the following (mutual exclusive) possibilities:

1.  $\{0\} = \psi_0$  and  $n = 1$ : The neighborhood scheme  $P = \langle n, \sigma, \phi \rangle$  has  $\sigma(0) = 0$  and  $\phi(0) = t$  (with  $t \in \mathbb{Z}$ ). We therefore have  $S \circ S^{t-1} = P_k$  for every  $k \in \mathbb{N}^+$ . This proves the  $\mu$ -periodic QGCA  $G' = \langle Q, n, \mu, M_0, \dots, M_{\mu-1}, S, \dots, S \rangle$  to be shift-equivalent with the initial QGCA  $G$ ; in other words  $G' \equiv G$ .
2.  $\{0\} = \psi_0$  and  $n > 1$ : Take  $c$  such that  $\sigma(c) = \max(\Psi)$  and apply the procedure. The new neighborhood scheme  $R$  has  $m(R) = m(P) - 1$ .
3.  $\{0\} \subsetneq \psi_0$ : Take  $c$  with  $\sigma(c) \neq 0$  and  $\phi(c) = \min(\Psi)$  and apply the procedure to get the neighborhood scheme  $R$ . Now we have  $\psi_0^R = \{\zeta(c)\} = \{\sigma(c)\} \not\supseteq \{0\}$  combined with  $m(R) = m(P) + n - 1$ .
4.  $\{0\} \not\subseteq \psi_0$  and  $m(P) > 1$ : Take  $c$  such that  $\sigma(c) = \max(\psi_r)$  and apply the procedure. For the neighborhood scheme  $R$  we have  $m(R) = m(P) - 1$  and (again)  $\{0\} \not\subseteq \psi_0^R$ .
5.  $\{0\} \not\subseteq \psi_0$  and  $m(P) = 1$ : This implies:  $\psi_0 = \{1, \dots, n-1\}$ ;  $\psi_1 = \{0\}$ ;  $\Phi = \{t, t+1\}$ ;  $\phi(c) = t+1$ ;  $\sigma(c) = 0$ , with  $t \in \mathbb{Z}$  and  $c \in \mathbb{Z}_n$  (therefore  $\phi(i) = t + 1_c(i)$ ). Define the permutation-gate  $\Pi(i) = \sigma(i) - 1$ . It now holds for every  $k \in \mathbb{N}^+$ :

$$\begin{aligned}
\Pi^{[k]} \circ S_k \circ S_k^{tn} &= \langle n, \Pi(i), 0 \rangle \circ \langle n, i+1, 1_{n-1}(i) \rangle \circ \langle n, i, t \rangle \\
&= \langle n, \Pi(i) + 1, 1_{n-1}(\Pi(i)) \rangle \circ \langle n, i, t \rangle \\
&= \langle n, \sigma(i), 1_c(i) + t \rangle \\
&= P_k
\end{aligned}$$

This shows that if we define a  $\mu$ -periodic QGCA

$$G' = \langle Q, n, \mu, M_0 \cdot \Pi, M_1, \dots, M_{\mu-1}, S, \dots, S \rangle$$

we have finally reached the shift-equivalence  $G \equiv G'$ .

If we start with the initial QGCA  $F$  and apply the above transformation recursively, we will always reach (after a finite amount of steps) possibility 5. This means that we have constructed the desired  $\mu$ -periodic QGCA  $G$ , with  $F \equiv G$ .  $\square$

## Appendix D

# Mapping the Calibrations

We will prove the existence of a mapping as used in Lemma 7.3.

**LEMMA D.1** *For every  $m, n \in \mathbb{N}^+$  with  $\gcd(m, n) = 1$ , there exists an  $\alpha \in \mathbb{N}^*$  and a mapping  $\Pi$  of  $A$  and  $B$  which respects the equations (with  $N = \alpha n$  and  $K = km$ )*

$$A_{j \bmod m}^{i \bmod n} = \Pi_{j-i+\lfloor \frac{2i}{2N} \rfloor \bmod K}^{2i \bmod 2N} \qquad B_{j \bmod m}^{i \bmod n} = \Pi_{j-i+\lfloor \frac{2i+1}{2N} \rfloor \bmod K}^{2i+1 \bmod 2N}$$

for every  $i, k \in \mathbb{N}$  and  $j \in \mathbb{Z}$ .

PROOF. In order to ensure a correct mapping between the  $A$  values and  $\Pi$ , we have to prove that if:

$$\begin{aligned} 2i \bmod 2N &\equiv 2p \bmod 2N \\ j - i + \left\lfloor \frac{2i}{2N} \right\rfloor \bmod K &\equiv q - p + \left\lfloor \frac{2p}{2N} \right\rfloor \bmod K \end{aligned}$$

then:

$$\begin{aligned} i \bmod n &\equiv p \bmod n \\ j \bmod m &\equiv q \bmod m \end{aligned}$$

for every  $i, p \in \mathbb{N}$  and  $j, q \in \mathbb{Z}$ .

By  $K = km$  and defining  $\alpha$  such that  $N = \alpha n \equiv 1 \bmod m$ , the initial conditions can be restated to:

$$\begin{aligned} 2(i-p) \bmod 2\alpha n &\equiv 0 \\ (j-q) - (i-p) + \left\lfloor \frac{2i}{2\alpha n} \right\rfloor - \left\lfloor \frac{2p}{2\alpha n} \right\rfloor \bmod km &\equiv 0 \end{aligned}$$

The first condition leads to  $p = i + \lambda \alpha n$  (with  $\lambda \in \mathbb{Z}$ ). As a result the second condition becomes:

$$(j-q) + \lambda(\alpha n - 1) \bmod km \equiv 0$$

Therefore the the restrictions  $(i-p) \bmod n \equiv 0$  and  $(j-q) \bmod m \equiv 0$  are satisfied for every  $i, p \in \mathbb{N}$  and  $j, q \in \mathbb{Z}$ .

The same reasoning holds if we want to prove the existence of a consistent mapping between the  $B$  values and  $\Pi$ .  $\square$

## Appendix E

# Sources of Information

The writing of this thesis has benefited greatly from the information which is nowadays available on the internet. The pace of the recent developments in quantum computing is unthinkable without the existence of archives such as `quant-ph` or `qcomp`. What follows is a personal selection of the *sites* and *home-pages* which I found particular useful.

<http://xxx.lanl.gov/archives/quant-ph>

The e-print archive of the Los Alamos National Laboratory has shown to be the center of publications regarding quantum computation.

<http://babbage.sissa.it/quant-ph/>

The European mirror-site of `quant-ph`.

<http://feynman.stanford.edu/qcomp/>

The Quantum Computation Archive at the Stanford University provides a large but well-selected database of articles solely about quantum computation.

<http://eve.physics.ox.ac.uk/QC/home.html>

The home-page of the Quantum Information Group at the University of Oxford. With on-line elementary introductions about quantum cryptography, cryptoanalysis, computation and communication by Artur Ekert and David Deutsch.

<http://vesta.physics.ucla.edu/~smolin/index.html>

The colorful Quantum Information Page of John Smolin at University of California, Los Angeles.

<http://www.cwi.nl/~berthiau/>

André Berthiaume's home-page with his introduction in quantum computation [7].

[http://www.iro.umontreal.ca/labs/theorique/index\\_en.html](http://www.iro.umontreal.ca/labs/theorique/index_en.html)

The Laboratory for Theoretical & Quantum Computing at the Université de Montréal. Gilles Brassard's extensive *Bibliography of Quantum Cryptography* can be found at the English home-page of Claude Crépeau.

<http://www.research.ibm.com/quantuminfo/>

The research group at IBM with an on-line article about quantum teleportation.

<http://www.physik.uni-ulm.de/~sam/home.html>

Starting point of the tutorial on quantum computation by Samuel Braunstein.

<http://publish.aps.org/PRA/prahome.html>

Site of *Physical Review A*, the journal where the field of quantum computation has found its niche.

<http://alife.santafe.edu/alife/topics/cas/ca-faq/ca-faq.html>

Frequently Asked Questions about cellular automata; edited by Howard Gutowitz. Keeps also track of the ongoing discussions in the newsgroup `comp.theory.cell-automata`.

<http://cs.ua.edu/graduate/lusth/qca/>

The Quantum Coupled Architecture Home Page which deals with a number of subjects closely related to QCA.

<http://aerodec.anu.edu.au/~qc/index.html>

The Quantum Computing home page of the Australian National University, including a list of paper reviews.

<http://altavista.digital.com/>

*"If all else fails, try consulting Alta Vista."* Current statistics (Summer 1996):

- "cellular automata": 5037 documents matching the query
- "quantum computation": 990 documents matching the query
- "quantum cellular automata": 40 documents matching the query

# Bibliography

- [1] Jürgen Albert and Karel Culik II. A simple universal cellular automaton and its one-way and totalistic version. *Complex Systems*, 1:1–16, 1987.
- [2] Serafino Amoroso and Yale N. Patt. Decision procedures for surjectivity and injectivity of parallel maps for tessellation structures. *Journal of Computer and System Sciences*, 6(5):448–464, October 1972.
- [3] Adriano Barenco. A universal two-bit gate for quantum computation. *Proceedings of the Royal Society of London A*, 449:669–677, 1995. <http://xxx.lanl.gov/abs/quant-ph/9505016>.
- [4] Adriano Barenco, Charles H. Bennett, Richard Cleve, David P. DiVincenzo, Norman Margolus, Peter Shor, Tycho Sleator, John Smolin, and Harald Weinfurter. Elementary gates for quantum computation. *Physical Review A*, 52(5):3457–3467, 1995. <http://xxx.lanl.gov/abs/quant-ph/9503016>.
- [5] Charles H. Bennett. Logical reversibility of computation. *IBM Journal of Research and Development*, 17:525–532, November 1973.
- [6] Ethan Bernstein and Umesh Vazirani. Quantum complexity theory. In *Proceedings of the 25th Annual ACM Symposium on Theory of Computing*, pages 11–20, San Diego, California, May 1993. ACM Press. [ftp://math.berkeley.edu/pub/Preprints/Bob\\_Solovay/Quantum/](ftp://math.berkeley.edu/pub/Preprints/Bob_Solovay/Quantum/).
- [7] André Berthiaume. Quantum computation. In *Complexity Theory Retrospective II*. Springer-Verlag, 1996. To appear. <http://www.cwi.nl/~berthiau/seminar.html>.
- [8] Michael Biafore. Can quantum computers have simple Hamiltonians? [http://dynamics.bu.edu/InterJournal/papers/\[3\]\\_120994152541\\_.html](http://dynamics.bu.edu/InterJournal/papers/[3]_120994152541_.html), November 1994. Presented at Physics and Computation 1994.
- [9] Michel Boyer, Gilles Brassard, Peter Høyer, and Alain Tapp. Tight bounds on quantum searching. <http://xxx.lanl.gov/abs/quant-ph/9605034>, May 1996. To be appear in the conference proceedings of Physics and Computation 1996.
- [10] Brian H. Bransden and Charles J. Joachain. *Introduction to Quantum Mechanics*. Longman Scientific & Technical, New York, 1989.
- [11] Gilles Brassard. Quantum computing: The end of classical cryptography? *SIGACT News*, 25(4):15–21, December 1994. Cryptology column.
- [12] Arthur W. Burks, editor. *Essays on Cellular Automata*. University of Illinois Press, Urbana, 1970.
- [13] Richard Courant and David Hilbert. *Methods of Mathematical Physics*, volume 1. Interscience Publishers, New York, 1953.
- [14] Ashok Das and Adrian C. Melissinos. *Quantum Mechanics: a modern introduction*. Gordon and Breach Science Publishers, New York, 1986.

- [15] 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.
- [16] David Deutsch. Quantum computational networks. *Proceedings of the Royal Society of London A*, 425:73–90, 1989.
- [17] Paul A.M. Dirac. *The Principles of Quantum Mechanics*. Clarendon Press, Oxford, fourth edition, 1958.
- [18] David P. DiVincenzo. Two-bit gates are universal for quantum computation. *Physical Review A*, 51(2):1015–1022, 1995. <http://xxx.lanl.gov/abs/cond-mat/9407022>.
- [19] Jean-Christophe Dubacq. How to simulate non-reversible turing machines by one-dimensional cellular automata. Technical Report URA CNRS 1398, École Normale Supérieure de Lyon, Laboratoire d’informatique du Parallélisme, March 1995.
- [20] Christoph Dürr and Miklos Santha. A decision procedure for unitary linear quantum cellular automata. In *Proceedings of the 37th Annual Symposium on Foundations of Computer Science*. IEEE Computer Society Press, October 1996. <http://xxx.lanl.gov/abs/quant-ph/9604007>.
- [21] Christoph Dürr, Huong Lê Thanh, and Miklos Santha. A decision procedure for well-formed quantum linear cellular automata. In Claude Puech and Rüdiger Reischuk, editors, *13th Annual Symposium on Theoretical Aspects of Computer Science*, volume 1046 of *Lecture Notes in Computer Science*, pages 281–292. Springer, February 1996.
- [22] Artur Ekert. Quantum computation. <http://feynman.stanford.edu/qcomp/ekert/index.html>, 1994. Proceedings of the ICAP meeting, Boulder.
- [23] Doyne Farmer, Tommaso Toffoli, and Stephen Wolfram, editors. *Cellular Automata*, volume 10(1,2) of *Physica D*, North-Holland, Amsterdam, March 1983. Los Alamos National Laboratory, Elsevier Science Publishers.
- [24] Richard P. Feynman. *The Feynman lectures on physics*, volume III: Quantum Mechanics. Addison-Wesley, 1965.
- [25] Joel N. Franklin. *Matrix Theory*. Prentice-Hall, 1968.
- [26] Lov G. Grover. A fast quantum mechanical algorithm for database search. In *Proceedings of the 28th Annual Symposium on the Theory of Computing*, pages 212–219, Philadelphia, Pennsylvania, May 1996. STOC’96. <http://xxx.lanl.gov/abs/quant-ph/9605043>.
- [27] Howard A. Gutowitz, editor. *Cellular automata: Theory and Experiment*, volume 45(1,2,3) of *Physica D*, North-Holland, Amsterdam, September 1990. The Center for Nonlinear Studies; Los Alamos National Laboratory, Elsevier Science Publishers.
- [28] Gustav A. Hedlund. Endomorphisms and automorphisms of the shift dynamical system. *Mathematical Systems Theory*, 3(4):320–375, 1969.
- [29] Peter Høyer. Note on linear quantum cellular automata. <http://www.imada.ou.dk/~u2pi/>, February 1996.
- [30] Richard Jozsa. Characterizing classes of functions computable by quantum parallelism. *Proceedings of the Royal Society of London A*, 435:563–574, 1991.
- [31] Jarkko Kari. On the inverse neighborhoods of reversible cellular automata. In Grzegorz Rozenberg and Arto Salomaa, editors, *Lindenmayer Systems, Impacts on theoretical Computer Graphics, and Developmental Biology*, pages 477–495. Springer-Verlag, 1992.

- [32] Jarkko Kari. Representation of reversible cellular automata with block permutations. *Mathematical Systems Theory*, 29(1):47–61, January 1996.
- [33] Peter Lancaster. *Theory of Matrices*. Academic Press, New York, 1969.
- [34] Seth Lloyd. A technologically feasible quantum computer. <http://feynman.stanford.edu/qcomp/lloyd/index.html>.
- [35] Seth Lloyd. A potentially realizable quantum computer. *Science*, 261:1569–1571, 1993.
- [36] Norman Margolus. Parallel quantum computation. In Wojciech H. Zurek, editor, *Complexity, Entropy, and the Physics and Information*, volume VIII of *SFI Studies in the Sciences of Complexity*, pages 273–287. Addison–Wesley, Redwood City, 1990. <http://feynman.stanford.edu/qcomp/margolus/index.html>.
- [37] Bruno Martin. A universal cellular automaton in quasi-linear time and its S–m–n form. *Theoretical Computer Science*, 123:199–237, 1994.
- [38] Akira Maruoka and Masayuki Kimura. Condition for injectivity of global maps for tessellation automata. *Information and Control*, 32:158–162, 1976.
- [39] David A. Meyer. From quantum cellular automata to quantum lattice gases. <http://xxx.lanl.gov/abs/quant-ph/9604003>, March 1996. To appear in *Journal of Statistical Physics*.
- [40] David A. Meyer. On the absence of homogeneous scalar quantum cellular automata. <http://xxx.lanl.gov/abs/quant-ph/9604011>, April 1996. Submitted to *Physics Letters A*.
- [41] David A. Meyer. Quantum mechanics of lattice gas automata I. One particle plane waves and potentials. Private communication, July 1996. Submitted to *Physical Review E*.
- [42] David A. Meyer. Unitarity in one dimensional nonlinear quantum cellular automata. <http://xxx.lanl.gov/abs/quant-ph/9605023>, April 1996. Submitted to *Communications in Mathematical Physics*.
- [43] Marvin Minsky. *Computation: finite and infinite machines*. Prentice-Hall International, London, 1967.
- [44] Kenichi Morita. Computation-universality of one-dimensional one-way reversible cellular automata. *Information Processing Letters*, 42:325–329, July 1992.
- [45] Kenichi Morita and Masateru Harao. Computation universality of one-dimensional reversible (injective) cellular automata. *Transactions of the IEICE*, E72(6):758–762, June 1989.
- [46] John von Neumann. *Theory of Self-Reproducing Automata*. University of Illinois Press, Urbana, 1966. Edited and completed by Arthur W. Burks.
- [47] Marshall C. Pease III. *Methods of Matrix Algebra*. Mathematics in Science and Engineering 16. Academic Press, New York, 1965.
- [48] Daniel Richardson. Tessellations with local transformations. *Journal of Computer and System Sciences*, 6(5):373–388, October 1972.
- [49] Peter W. Shor. Algorithms for quantum computation: Discrete logarithms and factoring. In Shafi Goldwasser, editor, *Proceedings of the 35th Annual Symposium on Foundations of Computer Science*, pages 124–134. IEEE Computer Society Press, November 1994. <http://feynman.stanford.edu/qcomp/shor/index.html>.
- [50] Peter W. Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. <http://xxx.lanl.gov/abs/quant-ph/9508027>, August 1995. Expanded version of [49].



- [51] Daniel R. Simon. On the power of quantum computation. In Shafi Goldwasser, editor, *Proceedings of the 35th Annual Symposium on Foundations of Computer Science*, pages 116–123. IEEE Computer Society Press, November 1994. <http://feynman.stanford.edu/qcomp/simon/index.html>.
- [52] Tommaso Toffoli. Computation and construction universality of reversible cellular automata. *Journal of Computer and System Sciences*, 15:213–231, 1977.
- [53] Tommaso Toffoli. Cellular automata as an alternative to (rather than an approximation of) differential equations in modeling physics. *Physica D*, 10:117–127, 1984. Contribution in [23].
- [54] Tommaso Toffoli and Norman Margolus. *Cellular automata machines: a new environment for modeling*. MIT Press, Cambridge, 1987.
- [55] Tommaso Toffoli and Norman Margolus. Invertible cellular automata: a review. *Physica D*, 45:229–253, 1990. Contribution in [27].
- [56] Paul Vitányi. Physics and the new computation. In Jiri Wiedermann and Petr Hájek, editors, *Mathematical Foundations of Computer Science 1995, Proceedings of the 20th International Symposium*, volume 969 of *Lecture Notes in Computer Science*, pages 106–128. Springer, August 1995. <http://www.cwi.nl/~paulv/publications.html>.
- [57] John Watrous. On one-dimensional quantum cellular automata. In *Proceedings of the 36th Annual Symposium on Foundations of Computer Science*, pages 528–537, Milwaukee, Wisconsin, October 1995. IEEE Computer Society Press.
- [58] Stephen Wolfram. *Theory and Applications of Cellular Automata*, volume 1 of *Advanced series on complex systems*. World Scientific Publishing Co Pte Ltd., 1986.
- [59] Stephen Wolfram. *Cellular Automata and Complexity*. Addison-Wesley, 1994. Collected papers.
- [60] Andrew Chi-Chih Yao. Quantum circuit complexity. In *Proceedings of the 34th IEEE Symposium on Foundations of Computer Science*, pages 352–361, 1993. <http://feynman.stanford.edu/qcomp/yao/index.html>.