6

Quantum Cellular Automata Controlled Self-Organizing Networks

Laszlo Gyongyosi and Sandor Imre
Department of Telecommunications, Budapest University of Technology
Hungary

1. Introduction

Every process of our universe is based on the fundamental elements of nature: information and computation. The basic motivation behind the study of quantum cellular automata (QCA) is the wish to analyze the processes of nature. QCA provide a natural framework within which to describe many classically undescribable and uncomputable physical phenomena, such as the properties of quantum physical systems and the complex background of quantum dynamics. Quantum cellular automata models are based on the working mechanism of classical cellular automata models and use the power of reversible quantum computation. Cellular automata can be used in many fields of science, such as parallel computation, artificial intelligence, image processing, biological systems, simulation of physics systems, hardware design, algorithm theory, and many more.

In the first part of this chapter, we give a brief overview of the basic properties of quantum information processing and analyze the quantum versions of classical cellular automata models. We present all materials in a clear, perspicuous, and comprehensible manner, without using a complex mathematical background. After reviewing physical QCA implementations, we sketch future directions, and then conclude the first part.

In the second part of the chapter, we examine one possible application of QCA, which uses quantum computing to realize real-life based, truly random network organization. This abstract machine is called a Quantum Cellular Machine (QCM), and we design it for controlling a truly random biologically-inspired network, and to integrate quantum learning algorithms and quantum searching into a controlled, self-organizing system. The self-organizing processes in classical systems cannot be truly random. Using our quantum probabilistic QCM unit, we can add truly random behavior to the self-organizing processes of biological networks. A quantum mechanical-based quantum cellular machine (QCM) controls the self-organizing processes of the network and uses a closed, non-classical quantum mechanical-based language inside the QCM. The proposed QCM solution has deep relevance in the evolution of truly random quantum probabilistic self-organizing network structures. The QCM controls the evolution of the system, changes its environment and creates plans without any human interaction, using truly random quantum probabilistic decisions. In a classical system, the classical circuits can only exhibit deterministic behavior. In a quantum probabilistic control system, the quantum circuits can follow both deterministic and quantum probabilistic quantum cellular machine control behaviors. The QCM has classical and quantum communication layers, it uses the classical layer to detect

the network environment. The lower layer of the quantum cellular machine contains the non-deterministic quantum probabilistic decisions and interacts with the classical level. The quantum cellular machine model with the power of quantum computing can be used for the development of a real-life based network organism. We show, that a real, biologically inspired, non-deterministic, truly random network model can be achieved by the discussed QCM model.
In the third part of this chapter, we show that a very efficient quantum searching algorithm can be integrated into the QCM, to find the best solution to a given network input command. We present a quantum searching based method specially designed for quantum probabilistic self-organizing networks, to reduce the complexity of the classical traditional search in the network. The proposed QCM can process both quantum and classical information, and accomplish both deterministic and quantum probabilistic tasks. The information unit is a quantum bit, which can lie in a coherent superposition state of logical states zero and one, and can thus simultaneously store zero and one. Using quantum bits, we can speed up the solutions of classical problems, and even solve some hard problems that classical computers can't solve. The key aspect behind the optimal decisions of a QCM is to design a high-efficiency searching algorithm. A QCM updates the probability amplitudes of its quantum register, according to a given reward value, derived from the network environment. The QCM repeatedly applies a unitary transformation to the quantum states, thus it can enhance, for example, the probability amplitude of the optimal path in the self-organizing network environment, while suppressing the amplitude of all other solutions. The QCM's quantum searching algorithm can help resolve many hard tasks, for example it could be applied to find an optimal logical path, using the effects of quantum mechanics. In the numerical analysis we will show that the quantum communication layer could improve the performance of classical systems.

2. Quantum Cellular Automata

The field of quantum information processing is growing dynamically, and the revolutionary properties of quantum dynamics can be exploited in many fields of science. The classical cellular automata (CA) model uses a discrete and infinite network, equipped with cells. The working mechanism of a CA can be described through the change of states of these discrete cells. The classical automata is deterministic and the state of the cells depends on the states of the neighbouring cells. The state of the automata is determined by the state of all its cells. The quantum version of the classical cellular automata has many advantages over the classical model. The quantum cellular automata (QCA) uses quantum parallelism, which makes it possible to address the cells simultaneously, in parallel, hence the behaviour of a QCA can be controlled globally. The global updating mechanism of the QCA model makes its physical implementation easy in practice, hence the individual manipulation of the quantum bits of the quantum register is not required (Watrous, 1995).

2.1 Related work

In classical computation, the automata—which in general describes parallel processes — could easily become very complex. Using the power of quantum computation and quantum parallelism, the complexity of these structures can be decreased dramatically (Perez-Delgado and Cheung, 2005). As we will see, the phenomena of quantum mechanics can be exploited and can be integrated into the quantum cellular machine. According to the physical attributes of the quantum cellular machine, the evolution of quantum systems can be analysed and discussed by the framework of the QCA, which is a hard task in many physical quantum systems (Dam, 1996).

In a quantum cellular machine, the information processing is realized by quantum operations and transformations. These quantum transformations are called unitary transformations, and these processes are applied on quantum bits, instead of classical bits. Quantum bits represent the fundamental basic unit of quantum information. In practice, quantum states can be realized by photons, electrons, atoms or half-spin particles (Grössing and Zeilinger, 1988). One of the most important properties of quantum states is that these qubits can be in a superposition state, which cannot be imagined for classical bits. This property means that a quantum state can be simultaneously in the logical state of 0 and 1. However, to convert the superponated information into 'useful' information, we have to apply a measurement to the qubit, which converts the quantum information encoded in the position of the state into classical information (Benioff, 1980), (Gyongyosi et al., 2009), (Imre and Balazs, 2005), (Nielsen and Chuang, 2000). The output of the measurement is not completely determined by the position of the qubit, hence its result is not deterministic, but probabilistic (Curtis and Meyer, 2004), (Miller et al., 2006). We will call this property, quantum probabilistic output or behaviour. The measurement destroys the state of the qubit, and changes its state to a "classical" or orthogonal state—according to the basis of the measurement, and the logical value of the classical output. Quantum algorithms, which exploit quantum parallelism, use quantum registers—the collection of superponated quantum states. Using a quantum register, tasks can be computed with an exponential speed up as the number of quantum states in the quantum register increases linearly (Margolus, 1991).

The idea of a cellular automata, or machine, was formulated by von Neumann, a Hungarian mathematician, who showed that a cellular automata based system was capable of self-reproduction (Neumann, 1966). Later, a two-dimensional cellular automata model became extremely popular—later known as the 'Game of Life' (Gardner, 1970). The model uses 'alive' and 'dead' cells, and there are many rules for the state 'dead' and for the state 'alive'. The cells can change between these two states, according to the properties of their neighbours. After the Game of Life had become so popular, new ideas have been presented. As has been shown, the physical properties of the processes of the classical world can be traced back to the fundamental properties of cellular automata behaviour. Later, the one dimensional cellular automata also was introduced (Watrous, 1995). The cellular automata models the world through parallel processes—hence, it is natural to apply the results of quantum information processing to cellular automata models. The idea of a Quantum Cellular Automata (QCA) was first mentioned by Toffoli and Margolus (Toffoli et al., 1990), (Margolus, 1991). Later, Feynman (Feynman, 1982), Grössing and Zeilinger (Grössing and Zeilinger, 1988), and Watrous (Watrous, 1995) have published some formalized results on the subject. The term QCA was first used by Grössing and Zeilinger, and the model of Watrous is based on Feynman's ideas. Based on these works, Richter and Werner (Richter and Werner, 1996) showed some new results in the field.

Parallelism of processes can be exploited dramatically with the help of quantum parallelism, since the transformations of the QCA's are applied in parallel on all the qubits of the quantum register. It also means that the quantum states of the QCA do not have to be addressed separately, since all the quantum states can be used simultaneously in the processes (Vollbrecht and Cirac, 2008). From an engineering viewpoint, the QCA machine can be regarded as a natural extension of the original classical CA model, since the working mechanism of the CA is defined as the combination of parallel processes (Benioff, 1980), (Neumann, 1966), (Miller et al., 2006). Since all the unitary transformations can be realized simultaneously in the model, the quantum states of the quantum register can be handled as an indistinguishable quantum register, where it is not required to control all the qubits in a separate manner, individually (Arrighi et al., 2007), (Toth and Lent, 2001).

2.2 Quantum computing

In this section, we give a brief overview of quantum mechanics, and we introduce the basic definitions of quantum information processing which will be used in the text.

2.2.1 Brief overview of quantum information processing

In quantum information processing, the logical values of classical bits are replaced by state vectors $|0\rangle$ and $|1\rangle$, - called the Dirac notation. Contrary to classical bits, a qubit $|\psi\rangle$ can also be in a linear combination of basis vectors $|0\rangle$ and $|1\rangle$.
The state of a qubit can be expressed as

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

where α and β are complex numbers, which is also called the superposition of the basis vectors, with probability amplitudes α and β. A qubit $|\psi\rangle$ is a vector in a two-dimensional complex space, where the basis vectors $|0\rangle$ and $|1\rangle$ form an orthonormal basis. The orthonormal basis $\{|0\rangle, |1\rangle\}$ forms the computational basis, in Fig. 1 we illustrate the computational basis for the case where the probability amplitudes are real (Imre and Balazs, 2005).

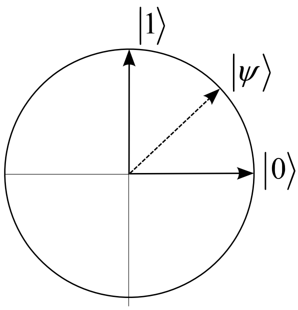

Fig. 1. Computational basis and general representation of a qubit in superposition state.
The vectors or states $|0\rangle$ and $|1\rangle$ can be expressed in matrix representation by

$$|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \text{ and } |1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}. \tag{2}$$

If $|\alpha|^2$ and $|\beta|^2$ are the probabilities, and the number of possible outputs is only two, then for $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ we have $|\alpha|^2 + |\beta|^2 = 1$, and the norm of $|\psi\rangle$ is $\||\psi\rangle\| = \sqrt{|\alpha|^2 + |\beta|^2} = 1$.

The most general transformation of $\| |\psi\rangle \|$ that respects this constraint is a linear transformation U that takes unit vectors to unit vectors.
A *unitary* transformation can be defined as

$$U^{\dagger}U = UU^{\dagger} = I, \tag{3}$$

where $U^{\dagger} = \left(U^{*}\right)^{T}$, hence the adjoint is equal to the transpose of complex conjugate, and I is the identity matrix.
The tensor product has an important role in quantum computation, here we quickly introduce the concept of tensor product. If we have complex vector spaces V and W of dimensions m and n, then the tensor product of $V \otimes W$ is an mn dimensional vector space. The tensor product is non-commutative, thus the notation preserves the ordering. The concept of a linear operator also can be defined over the vector spaces. If we have two linear operators A and B, defined on the vector spaces V and W, then the linear operator $A \otimes B$ on $V \otimes W$ can be defined as $(A \otimes B)(|v\rangle \otimes |w\rangle) = A|v\rangle \otimes B|w\rangle$, where $|v\rangle \in V$ and $|w\rangle \in W$. In matrix representation, $A \otimes B$ can be expressed as

$$A \otimes B = \begin{bmatrix} A_{11}B & \dots & A_{1m}B \\ \vdots & \ddots & \vdots \\ A_{m1}B & \dots & A_{mm}B \end{bmatrix}, \tag{4}$$

where A is an $m \times m$ matrix, and B is an $n \times n$ matrix, hence $A \otimes B$ has dimension $mn \times mn$.
The state $|\psi\rangle$ of an n-qubit quantum register is a superposition of the 2^n states $|0\rangle, |1\rangle, \dots, |2^n - 1\rangle$, thus

$$|\psi\rangle = \sum_{i=0}^{2^n - 1} \alpha_i |i\rangle, \tag{5}$$

with amplitudes α_i constrained by

$$\sum_{i=0}^{2^n - 1} |\alpha_i|^2 = 1. \tag{6}$$

The state of an n-qubit length quantum register is a vector in a 2^n-dimensional complex vector space, hence if the number of the qubits in the quantum register increases linearly, the dimension of the vector space increases exponentially.
A complex vector space V is a Hilbert space $\mathcal{H}$ if there is an *inner product*

$$\langle \psi | \varphi \rangle \tag{7}$$

with $x, y \in \mathbb{C}$ and $|\varphi\rangle, |\psi\rangle, |u\rangle, |v\rangle \in V$ satisfying the rules $\langle \psi | \varphi \rangle = \langle \varphi | \psi \rangle^{*}$, $\langle \varphi | \left(a|v\rangle + b|v\rangle\right)\rangle = a\langle \varphi | u \rangle + b\langle \varphi | v \rangle$, and $\langle \varphi | \varphi \rangle > 0$ if $|\varphi\rangle \neq 0$. If we have vectors

$|\varphi\rangle = a|0\rangle + b|1\rangle$ and $|\psi\rangle = c|0\rangle + d|1\rangle$, then the inner product in matrix representation can be expressed as

$$\langle\varphi|\psi\rangle = \begin{bmatrix} a^* & b^* \end{bmatrix} \begin{bmatrix} c \\ d \end{bmatrix} = a^*c + b^*d. \tag{8}$$

The norm of the vector $|\varphi\rangle$ can be expressed as $\| |\varphi\rangle \| = \sqrt{\langle\varphi|\varphi\rangle}$, and the dual of the vector $|\varphi\rangle$ is denoted by $\langle\varphi|$. The *dual* is a linear operator from the vector space to the complex numbers, defined as $\langle\varphi|(|v\rangle) = \langle\varphi|v\rangle$. The outer product between two vectors $|\varphi\rangle$ and $|\psi\rangle$ can be defined as

$$|\psi\rangle\langle\varphi|, \tag{9}$$

satisfying $(|\psi\rangle\langle\varphi|)|v\rangle = |\psi\rangle\langle\varphi|v\rangle$. The matrix of the outer product $|\psi\rangle\langle\varphi|$ is obtained by usual matrix multiplication of a column matrix by a row matrix, however the matrix multiplication can be replaced by tensor product, since:

$$|\varphi\rangle\langle\psi| = |\varphi\rangle \otimes \langle\psi|. \tag{10}$$

If we have vectors $|\varphi\rangle = a|0\rangle + b|1\rangle$ and $|\psi\rangle = c|0\rangle + d|1\rangle$, the outer product in matrix representation can be expressed as

$$|\varphi\rangle\langle\psi| = \begin{bmatrix} a \\ b \end{bmatrix} \begin{bmatrix} c^* & d^* \end{bmatrix} = \begin{bmatrix} ac^* & ad^* \\ bc^* & bd^* \end{bmatrix}. \tag{11}$$

In Fig 2. we illustrate the general description of a unitary transformation on an *n*-length quantum state, where the input state $|\psi_i\rangle$ is either $|0\rangle$ or $|1\rangle$, generally. After the application of a unitary transformation U on the input states, the state of the quantum register can be given by a state vector $|\psi\rangle$.

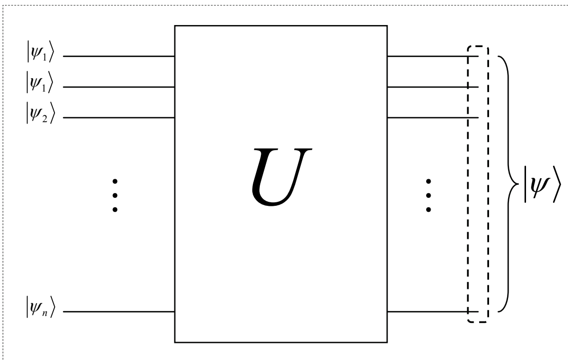

Fig. 2. General sketch of a unitary transformation on an *n*-length quantum register.

The unitary operator U is a $2^n \times 2^n$ matrix, with, in principle, an infinite number of possible operators. The result of the measurement of the state $|\psi\rangle$ results in zeros and ones that form the final result of the quantum computation, based on the n-length qubit string stores in the quantum register.

The quantum circuit of a QCM realizes a reversible operation, and any reversible quantum operation can be expressed as a unitary matrix. For a unitary transformation U, the following property holds:

$$(U^T)^* = U^{-1}, \tag{12}$$

where T denotes transposition and * denotes complex conjugation. The inverse transformation of U also can be expressed by the adjugate $U^\dagger$, which is equal to U^{-1}. One of the most standard quantum gates is the Controlled-NOT (CNOT) gate (Nielsen and Chuang, 2000), (Toffoli et al., 1990).

The CNOT gate is a very important gate in quantum computation, since from the one qubit quantum gates and the CNOT gates every unitary transformation can be expressed, hence these gates are universal. This gate is a two-qubit gate and it contains two qubits, called the control and the target qubit. If the control qubit is $|1\rangle$, then the gate negates the second qubit—called the target qubit. The general CNOT gate is illustrated in Fig. 3.

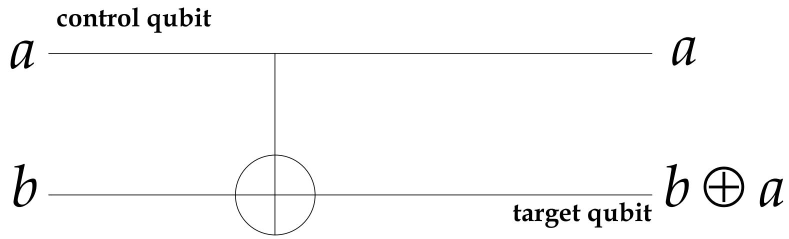

Fig. 3. The Controlled-NOT (CNOT) gate.

As can be verified, the quantum CNOT gate can be regarded as the generalization of the classical XOR transformation, hence $\mathbf{CNOT}|a, b\rangle = |a, b \oplus a\rangle$, which unitary transformation can be expressed in matrix form as follows:

$$\mathbf{CNOT} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}. \tag{13}$$

The controlled behaviour of the CNOT gate can be extended to every unitary transformation, and the generalized control quantum gate can be defined. In Fig. 4, we show a controlled U transformation, the U transformation is applied to the target qubit b only if the control qubit a is in high logical state. We note that the CNOT gate cannot be used to copy a quantum state, while the classical XOR gate can be applied to copy a classical bit.

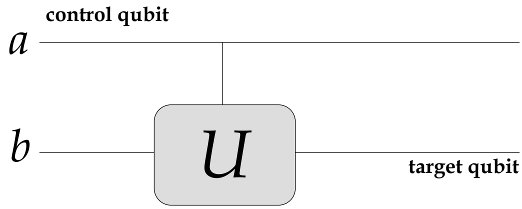

Fig. 4. A controlled-*U* gate.

As can be seen easily, for the CNOT gate, this unitary transformation U is equal to the NOT-transformation, hence the CNOT gate is a controlled-X gate, actually. We can also define the inverse of this transformation as the controlled-$U^{\dagger}$ transformation, as follows:

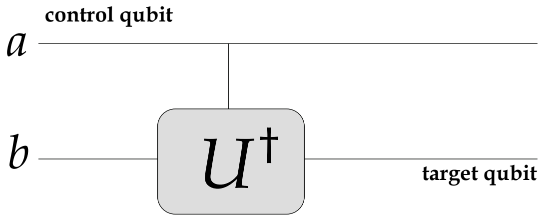

Fig. 5. A controlled inverse *U* gate.

Another important issue in the QCMs quantum circuit is the measurement operator M. The measurement operator converts the quantum information to classical, since after the measurement of a quantum state, the quantum information which is encoded in the quantum state becomes classical, and can be expressed as a logical 0 or 1. The general measurement circuit is illustrated in Fig. 6.

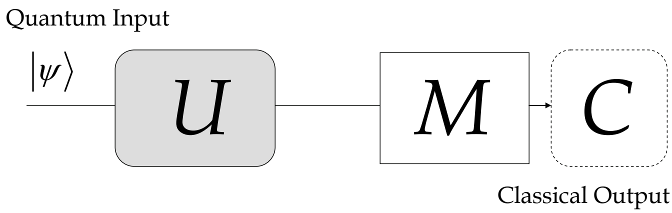

Fig. 6. The measurement of quantum information. The *M* measurement converts the quantum information to classical.

If we measure the quantum state $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$, then the output will be $M = 0$ with probability $|\alpha|^2$ or $M = 1$ with probability $|\beta|^2$.

For the general case, if we measure the *n*-length quantum register $|\psi\rangle = \sum_{i=0}^{2^n-1} \alpha_i |i\rangle$, with possible states $|0\rangle, |1\rangle, \ldots, |2^n - 1\rangle$, then the quantum measurement can be described as a set of $\{M_m\}$ of linear operators. The number of the possible outcomes is n, hence the number *m* of possible measurement operators is between $1 \le m \le n$.

If we measure the quantum register in state $|\psi\rangle = \sum_{i=0}^{2^n-1} \alpha_i |i\rangle$, then the outcome i has a probability of

$$\Pr(i) = \langle \psi | M_i^{\dagger} M_i | \psi \rangle . \tag{14}$$

The sum of the probabilities of all possible outcomes is $\sum_{i=1}^{m} \Pr(i) = \sum_{i=1}^{m} \langle \psi | M_i^{\dagger} M_i | \psi \rangle = 1$, according to the completeness of the measurement operators, since $\sum_{i=1}^{m} M_i^{\dagger} M_i = I$.

After the measurement of outcome i, the state of the quantum register collapses to

$$|\psi'\rangle = \frac{M_i |\psi\rangle}{\sqrt{\langle \psi | M_i^{\dagger} M_i | \psi \rangle}} = \frac{M_i |\psi\rangle}{\sqrt{\Pr(i)}}. \tag{15}$$

Using the previous example, if we have single quantum state $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$, then the measurement operators can be defined as

$$M_0 = |0\rangle\langle 0| \text{ and } M_0 = |1\rangle\langle 1| , \tag{16}$$

since the unknown qubit is defined in the orthonormal basis of $|0\rangle$ and $|1\rangle$.

2.3 Properties of Quantum Cellular Automata

The main idea behind QCA models can be stated as follows: exploit maximally the phenomena of quantum mechanics to outperform the classical model. It seems to be a natural extension to use the properties of quantum information processing, since all the operations of the classical version are parallelized. The QCA models have the advantage, that its can be used to simulate any quantum circuit, or the quantum Turing machine, or can be used to simulate the properties of entanglement transmission. This property is called the *universality* of the quantum cellular automata models, (Curtis and Meyer, 2004), (Grössing and Zeilinger, 1988).

The definition of the QCA model was formalized by Watrous (Watrous, 1995). The QCA can be described as a four-tuple, which consists of a *d*-dimensional construction of cells, a finite set of states, a finite set of neighbours, and a local transaction function. The quantum version of the classical CA uses discrete time and discrete space, however in the quantum case, the transitions are realized by unitary transformations. Moreover, according to quantum parallelism, each of the transformations of the QCA are realized simultaneously on all cells of the QCA. On the other hand, the transformations can be applied only for a small set of local neighbourhoods, in this case, the updating is achieved locally. The updating transformations operate probabilistically on the cells, and for a *well-formed* QCA, all the probabilities of the unitary transformations have to preserve the squared sum of these properties, which is equal to one. As has also been shown, this 'well-formed' property can be verified algorithmically (Dam, 1996), (Perez-Delgado and Cheung, 2005).

In the classical CA model, the processes and the cell updating mechanisms are achieved on classical bits, and the transformations are classical transformation. In a quantum system, the processes are realized by unitary transformations, and the transition functions of these unitary transformations are slightly different from the classical approach. Moreover, according to the no-cloning theorem (Wootters and Zurek, 1982), an unknown quantum state cannot be cloned perfectly, hence the quantum states cannot be split into two quantum registers, and then spliced into one quantum register again. In classical CA, this synchronization method can be applied easily, since the classical bits can be copied freely, an unlimited number of times. The quantum states of the QCA are updated without the possibility of register duplication, using different approaches, such as the partitioning of the quantum register. The partitioning scheme is used to construct a reversible version of the classical cellular automata model. A cellular automata is called *reversible* if there exists only one possible configuration for every actual configuration. The QCA is a reversible automata, however to define the QCA we have to extend the reversibility property and we have to add other properties.

2.3.1 The formal model of the QCA

The transition function of the QCA realizes the map instantaneously and for all the qubits of the quantum register simultaneously (Watrous, 1995). Watrous's one-dimensional QCA consists of the finite set of all possible states, the finite set of the automata's neighborhoods, and a local transition function. The cells of the quantum automata are described in a Hilbert space, and the cells are in superposition states, hence all the possible classical CA states can be handled simultaneously in the QCA model. The results of the transition functions are described as complex numbers in the Hilbert space (Arrighi et al., 2007), (Meyer, 1996).
In a QCA, the transformations are generally achieved by physical processes, hence they can be naturally described as unitary transformations. The unitarity of the one-dimensional QCA can be verified by algorithmic tools, checking whether the sum-squared probabilities of the transformations is equal to 1, or not.

2.3.2 Partitioned and Block-partitioned QCA

The main purpose of the quantum cellular automata model is the realization of the computation of all computable functions in a parallel way, using the fundamental properties of quantum mechanics. After Watrous published the one-dimensional QCA model, the *partitioned* version of this automata was also introduced (Dam, 1996), (Meyer, 1996), (Toffoli et al., 1990).
The partitioned QCA model uses cells, which can be divided into *sub-cells*: these cells are the left, right and centre cells. In a partitioned QCA model, the next state of a cell is determined by the right sub-cell of the left neighbour of the cell, and by the middle sub-cell of itself and by the left sub-cell of the right cell. The cell structure of the partitioned QCA is illustrated in Fig. 7. The local transition function is denoted by μ.
In the partitioned one-dimensional QCA model, the transition function μ can be divided into many permutations of the sub-cells in the neighbourhood of a given cell. The decomposition of μ is based on the fact that transition function can be expressed as unitary transformations, which transformations act on the given cell.
The decomposed transformation function μ can be described as these unitary transformations and by swap transformations between the sub-cells of different cells. A swap transformation between the sub-cells of the neighbouring cells is illustrated in Fig. 8.

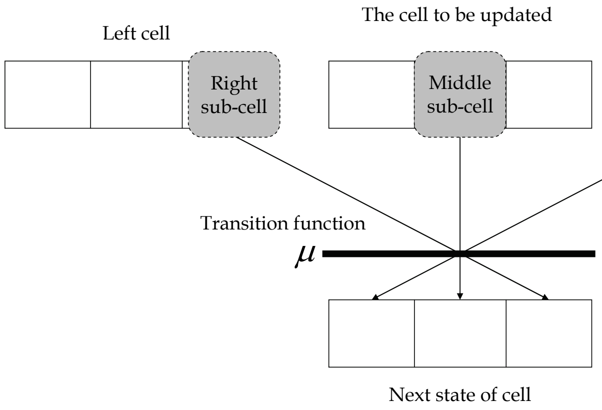

Fig. 7. The one-dimensional QCA. Each cell can be decomposed into three sub-cells.

The cells are divided into three sub-cells. In the updating process, the quantum automata swaps the left and right sub-cells of the neighbouring cells. Next, it updates each cell internally, with the help of a unitary operation, which acts on the three sub-cells of every cell. The partitioned QCAs form a subclass of QCA. There exist several other important subclasses, such as the Block-partitioned QCA (Dam, 1996).

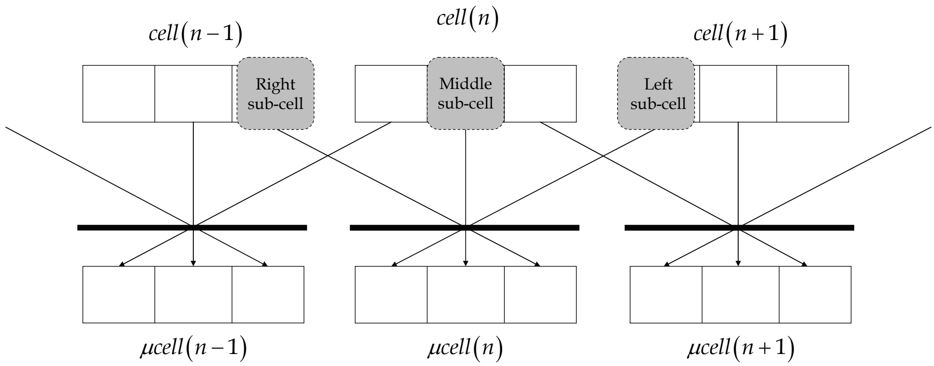

Fig. 8. The updating rule of the QCA. The position of the left and right sub-cells of the neighbouring cells are changed in the next state of the given cell. The updating process is realized by unitary operations applied to each sub-cell of the cells.

The block-partitioned model was introduced by Margolus and Toffoli (Margolus, 1991), (Toffoli et al., 1990) and in this type of model, the cells are divided into blocks, where each block contains two cells. This subclass has deep relevance in practice, since it can be applied to model the properties of physical systems, and other properties also can be explored with this model. The transition function is achieved on *block-level*, hence the maps are realized between the blocks, or the states of the blocks (Dam, 1996), (Meyer, 1996). In the communication process, the blocks can be shifted, and the unitary transformations are applied to the blocks. The rules can be realized by quantum gates as we will see in the second part of this chapter, in which we will define the most important quantum gates.

As a natural generalization of the block-partitioned QCA model, every one-dimensional QCA can be expressed as a set of local unitary operations. According to the Margolus partitioning scheme, a one-dimensional QCA can be expressed by unitary operations (Nielsen and Chuang, 2000), (Watrous, 1995).
In Fig. 9, we show a one-dimensional QCA, partitioned into unitary transformations. The unitary transformations belong to the blocks of the QCA.

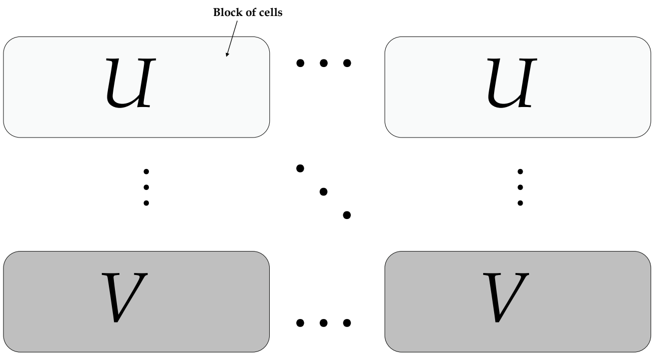

Fig. 9. Block-partitioned QCA. The blocks can be used to realize unitary transformations.

As a result of the decomposition of quantum transformations into elementary quantum gates, every quantum transformation and behaviour can be expressed using one-qubit and two-qubit quantum gates. In the second part of this chapter, we show a *reduction method*, which is able to find the most simple version of a given quantum transformation. It makes the circuits of the quantum cellular automata very easy to implement in practice, using basic quantum devices and tools. The rules of the quantum cellular automata can be viewed as elementary quantum gates applied to the cells, hence the quantum circuit of a QCM can be expressed as the concatenation of different rules, which rules are applied to the cells of the automata model.
Using Margolus's result (Margolus, 1991), in the second part of this chapter we will show, that any quantum probabilistic controller logic can be constructed from elementary unitary quantum transformations. Moreover, an intelligent self-organizing structure can be constructed from these results, as we will see in the second and the third parts of this chapter.
In the next part of this chapter, we show a quantum cellular automata based approach, called the *Quantum Cellular Machine* (QCM), which uses reversible quantum probabilistic quantum circuits to vest quantum probabilistic, truly random behaviour in a self-organized network structure. The QCM combines the basic properties of the block partitioned QCA model and adds many extended functions, according to its quantum circuit realization. The QCM's quantum circuit can realize any unitary transformation, using one- and two-qubit elementary quantum gates, and combines classical and quantum information processing.

2.3.3 Physical QCA Implementations

In this section, we have reviewed the basic properties of Quantum Cellular Automata (QCA). The QCA model refers to the quantum computational analogy with conventional classical models of cellular automata. The physical implementation of a 'classical' cellular automata—called the quantum dot cellular automata—can be exploited by the fundamental properties of the phenomena of quantum mechanics. The quantum dots are nano-structures, which can be constructed from standard semi conductive materials. In these structures, the electrons are trapped inside the dot, however the smaller physical quantum dot requires a higher potential energy for an electron to escape. In a quantum dot QCA, the quantum dots respond to the charge state of their neighbours.

In the quantum dot implementation, the working processes of the logic are based on neighbour interactions, or the movement of charge. The main advantage of these implementations is their very low power, however in these implementations the input and the output is not isolated well.

As in Fig. 10, there are two energetically equivalent ground state polarizations, which can be labelled as logical ‘0’ and logical ‘1’ (Arrighi et al., 2007), (Meyer, 1996), (Toth and Lent, 2001).

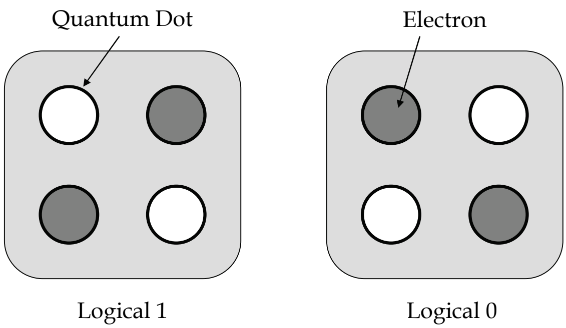

Fig. 10. The basic logic unit of the QCA is the cell. The QCA cell contains four quantum dots.

In experimental realizations, the quantum dot can be described as a nanometre sized structure. The quantum dot has the capability of trapping electrons. To represent the logical zero and one states, the quantum dot confines these electrons, which can escape only if a high potential is available.

The first implementation of physical quantum dot QCA machine was introduced by Toth and Lent (Toth and Lent, 2001). In their method, the state of the system converges to the lowest energy state, which state can be regarded as a uniquely defined state in the system. Later, the construction of a physical QCA model was extended to tunnel junctions and other phase changer nano-technological devices, using correlated magnetic and electrical states, or other magnetic and non-magnetic particles such as the metal tunnel junction QCA or the molecular and magnetic QCA (Arrighi et al., 2007), (Richter and Werner, 1996).

3. Quantum probabilistic control of self-organizing networks

A novel approach to future network communication is the quantum probabilistic self-organizing network structure. The main component of this kind of network structure is the QCM (*Quantum Cellular Machine*). The QCM uses a classical language to communicate with the components of the quantum probabilistic self-organizing networks, and uses a closed, non-classical quantum mechanical based language inside the component model. The proposed QCM model uses quantum computing for the development and the control of a truly random network organism.

In this part, we propose the QCM model to use quantum computing for the development of a real-life based network organism. The QCM model applies quantum mechanics and quantum computing to act on the QCM's internal self model and on its lower quantum layer. The particular quantum effects such as superposition or entanglement can be used to represent a real-life based self-organism and to demonstrate the quantum mechanically based unpredictable logical behaviour. The non-deterministic effects of quantum mechanics can be implemented well in the QCM model by the probabilistic outputs and quantum measurements. The QCM's internal self quantum state cannot be formalized classically; however its internal state machine can be implemented as a cellular automaton. The cells communicate directly by the real inputs and outputs and general transformations are performed on a global level by quantum circuits and quantum algorithms. The proposed quantum model's goal is to demonstrate the quantum mechanical based model's superiority to classical cellular automata based self-organizing networks.

In this part of this chapter, we define the theoretical basis of the QCM's quantum logic; then we give an example of implementation of the QCM module. Finally we conclude with the results and benefits of our quantum mechanical based model.

3.1 Properties of QCM

In the biologically inspired network organization models, the non-repeating output is desired for the same input (Curtis and Meyer, 2004), (Miller et al., 2006) and thus the QCM component behaviour has to be *non-deterministic*. However, the classical cellular automata model has no more dynamical patterns, only the information from the environment and its local state. Thus, if we want to construct a real biologically inspired non-deterministic model, we have to use quantum mechanics and a mapping that allows the QCM to use quantum level communication on a higher, logical level (Gyongyosi et al., 2009). The QCM can redefine the classical-level actions by this lower quantum control level, by applying the rules of the quantum level.

The two layers use two different communication methods. The classical level describes the QCM and network interacting using the classical communication layer. The QCM uses the classical layer to perceive the network environment through the data and information sent by other network components and to execute the given instructions. The lower layer of the QCM represents the non-deterministic quantum probabilistic decisions. The quantum level evolves dynamically, and it represents the properties of the biological organizations and interacts with the classical level.

Our purpose is to implement quantum mechanics based probabilistic decisions in biologically inspired network organizations and network adaptation, and to use them in problems where real non-deterministic behaviour is needed.

The logical output function remaps the network's actual logical state and the internal self model according to either the self model or the logical output function. The logical output function is used by the QCM's self model to modify both the network's state and the internal self model state. The logical output function affects the execution of the network commands, and hence the results of the quantum learning algorithm are converted back to classical ones.

In Fig. 11 we illustrated the hierarchical structure of the QCM. The decisions are based on the internal state and network model, and the output of the QCM is propagated back to the classical layer. The output of the QCM is determined by quantum learning and quantum searching algorithms. The network model and the self model variables are stored as qubits in the quantum layer of the QCM. The other network variables are stored and handled on the classical layer.

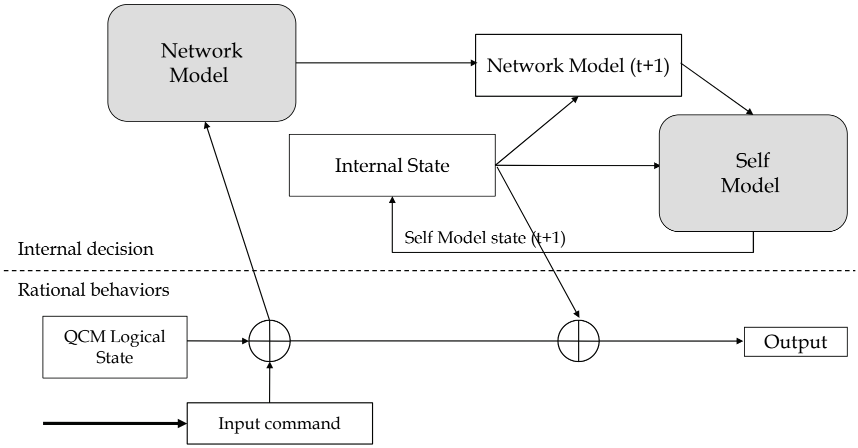

Fig. 11. The hierarchical structure of the QCM model.

In classical systems, the classical network elements can realize only deterministic behaviour. In our quantum probabilistic control system, the quantum circuits can realize both deterministic and quantum probabilistic behaviours. The quantum control based QCM forms a quantum system exploiting the superposition of the state of the QCM with the state of the network environment.
The quantum probabilistic controlled QCM uses automated methods to synthesize quantum behaviours from the examples, the *cares* of the quantum controlled QCM's quantum control table. The minterms not given as examples are the *don't knows*. The minterms not given as examples are converted to output values with various probabilities (Miller et al., 2006). We use the *simplicity principle* by seeking circuits of *reduced complexity*. We extend the QCM's logic synthesis approach to a *learning method* using quantum circuits (Nielsen and Chuang, 2000), (Curtis and Meyer, 2004).

3.1.1 Learning quantum behaviours from network examples

Logic synthesis methods applied to binary functions with many *don't cares* are used as the basis of various learning approaches. The *learning process* creates a circuit description and converts *don't cares* to *cares* trying to satisfy the simplicity. The method of logic synthesis based learning has already been applied to binary and multiple-valued circuits; however it has not been applied to quantum circuits.
In our learning method we use the EPR circuit to realize entanglement. The EPR-generator circuit is illustrated in Fig. 12. The circuit contains a Hadamard gate and a Controlled-NOT (CNOT) gate.
Our QCM *controller's unitary matrix* is the mapping between QCMs inputs and outputs. The unitary mapping is closely related to the behaviour observed on the QCM. The behaviours of the QCM combine quantum probabilistic and deterministic behaviours given by the unitary controller matrix.

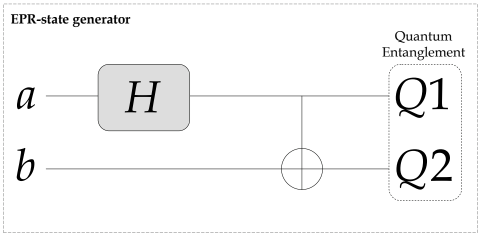

Fig. 12. The EPR-state generator quantum circuit.

The self-organizing processes of the network can be modelled as many cellular automata in parallel, which permute the cells and apply the rules in parallel for many cells. From this viewpoint, the constructed QCM model can be described as a cellular automata model, which uses permutation and parallel quantum transformations on the quantum register. The permutations and unitary transformations are realized in two individual steps, sequentially. The automata repeats the sequences, until the evolution of the quantum system reaches its desired output (Dam, 1996).
The abstract cellular automata-based model of the QCM is illustrated in Fig. 13.

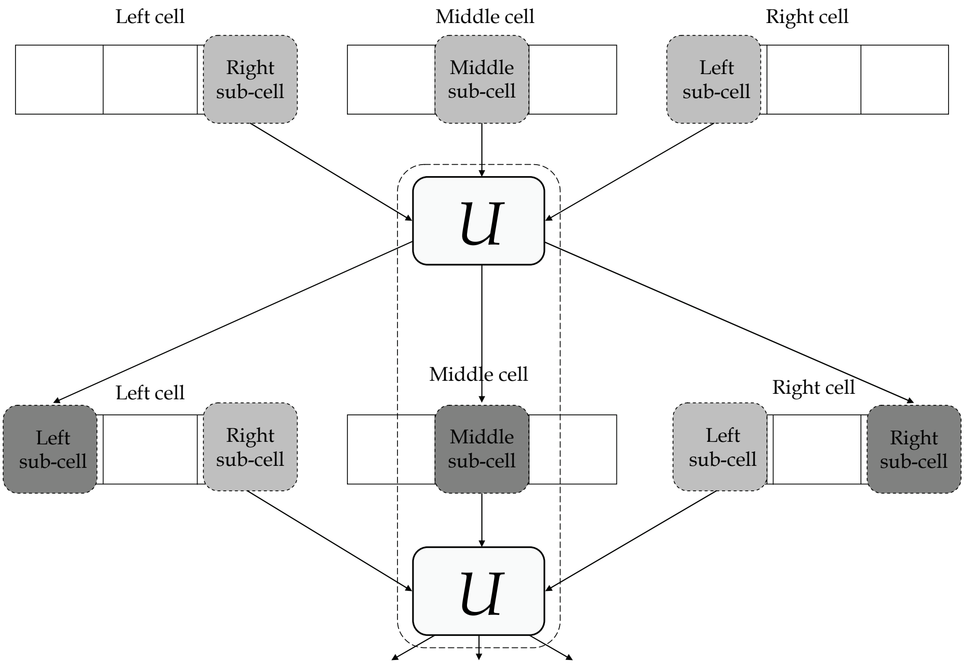

Fig. 13. Extended parallel gate construction. The unitary transformations of the QCA are realized by the quantum gates in parallel.

In this abstract cellular automata representation, the quantum transformations can be changed in every step, and the sequence of the gates can be different in every transformations.

3.1.2 Learning incompletely specified quantum functions

In the quantum probabilistic controlled system, any quantum control function is constructed in the complex Hilbert space $H^{\otimes N}$ from a set of a single-quantum bit and two-quantum bit operators

$$G = \left\{ [I], [controlled - U], \left[controlled - U^{\dagger} \right], [CNOT] \right\}. \tag{17}$$

In general the synthesis problem is to find the simplest circuit for a *Unitary Control Map* (UCM) table with *few given examples* and *many don't cares*. The given examples are mean *cares* in the *UCM* table. Let *G* be a set of *single-qubit* and *two-qubit* unitary operators on complex H^N ; then the process of synthesis can be expressed as a minimization of the given function with respect to the width of the QCM's circuit and the amount of elementary operators used inside the QCM's unitary control mechanism. Thus, this synthesis S_{H^N} can be written as follows (Miller et al., 2006):

$$S_{H^N}(n,G) \xrightarrow{\min} V(n,G), \tag{18}$$

where $V(n,G)$ is the cost of the QCM's circuit constructed of gates from set *G*. We use only single-qubit and two-qubit gates, and thus the QCMs learned by this method are directly implementable in quantum hardware, have low hardware costs, and satisfy the simplicity criteria (Curtis and Meyer, 2004).

3.2 Quantum probabilistic controlling system

We define the quantum probabilistic QCM controlling system. Let I be a set of vectors such that

$$I_k^p = \{i_0, i_1, \ldots, i_n\}, \; n = 2^N \tag{19}$$

is the *k*-th input vector of an *N* quantum bit length state of pattern P or function specification, and

$$f : I \to O \tag{20}$$

is a *reversible* function. The output vector

$$O_k^p = \{o_0, o_1, \ldots, o_n\} \tag{21}$$

is the expected *result vector* for the *input pattern* I_k^p and O is the set of all output vectors. Let $i_k \in \{0,1\}$ and $o_k \in \{0,1\}$ be the elements of the input and output vectors respectively. Let $|\psi\rangle$ be a *three-qubit* quantum state and G be the set of possible operators, and thus *quantum gates* (Curtis and Meyer, 2004). Then, there exists a quantum logic circuit U_f such that for any pair of input patterns I_i^p and output patterns

$$\left(I_i^p, O_i^p\right) \; : \; I_i^p \in I^p, \; O_i^p \in O^p, \tag{22}$$

where $\forall O_i^p \in O \exists I_i^p \in I$ such that $f\left(I_i^p\right) = O_i^p$ is a *one-to-one* mapping. This means that there is a *unitary transform* on a quantum system $U_f|\psi\rangle \to |\psi'\rangle$ for $|\psi\rangle \in I^p$, $|\psi'\rangle \in O^p$. The learning of such a function implies finding the *minimal set* of quantum gates implementing function f and realizing unitary control matrix U_f. The *unspecified* outputs are denoted by *X*; these values represent a *don't care* logic value and correspond to an unknown output. The set of examples is given as a set of pairs

$$P = \{i_k, o_k\}, k = 1, \ldots, n \le 2^N. \tag{23}$$

For the *incompletely* specified functions, the QCM uses a process to *explicitly find* a mapping or function satisfying each pair $\left(I_k^p, O_k^p\right)$ from the given set P such that

$$f\left(I_k^p\right) = O_k^p. \tag{24}$$

The goal of the QCM's learning process is to find a circuit that realizes a *complete* mapping with *incomplete* specifications and that agrees with the set of input-output pairs from the specification examples. The result of the QCM's learning process is thus a circuit that describes a *complete mapping* that agrees with the set of input-output pairs from the specification examples (Miller et al., 2006).
Thus, let f be a three-qubit *incompletely specified* reversible function defined in Table 1.

Target	0	1
Control: ab		
00	000	001
01	X	X
11	100	101
10	X	X

Table 1. An incompletely specified UCM table of the QCM.

The function can be completed as a *reversible* map since all output *care* values $\{000, 001, 100, 101\}$ are different. The table thus represents the set of learning examples, also called the *problem specification*. Then an *arbitrary* unitary transformation U satisfies all the *specified* transitions $U|000\rangle \to |000\rangle$, $U|001\rangle \to |001\rangle$, $U|110\rangle \to |100\rangle$ and $U|111\rangle \to |101\rangle$. Together, these transformations are a *valid solution* to the *learning problem* specified in Table 1. Using simplicity criteria, the circuit is reduced and its *UCM* unitary matrix is simplified as well (Miller et al., 2006).

3.2.1 The difference between classical and quantum learning

In the classical learning method the QCMs learn a deterministic function. In our probabilistic quantum learning, the QCM learns a unitary mapping to a quantum state that is quantum-deterministic only before the measurement. However the observer never knows to which classical state this deterministic state will collapse as the result of the measurement. Thus, when we design the network, we set certain constraints for the QCM's behaviour but we can only probabilistically predict how the QCM will behave within the constraints (Curtis and Meyer, 2004).

The QCM's UCM unitary control map represents the learned function using the quantum probabilistic learning. The *don't cares* are denoted by U_0 and U_1, where the subindex denotes the desired output value. The quantum probabilistic learning has similar results to standard probabilistic learning with the difference that the probabilities are calculated from quantum states, which are complex vectors.

Assume a single output function defined by the QCM's UCM table specifying the *desired* outputs as would result by observing values *0* and *1* on the QCM's quantum output in some special cases of the state of the environment. The UCM table contains *don't cares* and *cares*, and assuming there is a method to synthesize the cares, the problem that remains to be solved is the manner in which it is possible to specify the values of *don't cares*. The unitary operators used in the quantum probabilistic control based QCM network model are:

$$\left\{[NOT],[U],\left[U^{\dagger}\right],[CNOT],[controlled-U],\left[controlled-U^{\dagger}\right]\right\}. \tag{25}$$

How the *don't cares* are filled with respect to the QCM's learning method should be specified. For this, let $S_{out}=\{0,1,U_0,U_1\}$ be the set of all possible symbolic outputs of the given single output function. The $U_0=U|0\rangle$ and $U_1=U|1\rangle$ symbols represent quantum states, vectors, or complex numbers that correspond to measuring or observing the QCM's system in state $M(U_0)$ and $M(U_1)$, where M is the quantum measurement operator (Curtis and Meyer, 2004).

The QCM's expected input-output mapping can be changed by replacing the *controlled-U* operators by controlled-Hadamard gates. The well defined input values *0* and *1* remain the same; however the *don't care input* specifications can be mapped to four different symbols: $0,1,U_0,U_1$. The *X don't cares* will be replaced with $0,1,U_0$ or U_1. The well defined inputs remain the same, while the *don't cares* are mapped to one of the four possible values $X\rightarrow 0,1,U_0,U_1$. The probabilities of the output states depend directly on the *logic of the QCM* and thus on the quantum gates selected by the learning algorithm (Miller et al., 2006). We use the reversible quantum gates, CNOT, Controlled-U, and Controlled-$U^{\dagger}$ gates. In our quantum probabilistic based control system, if the controlled quantum bit is an eigenvalue of the unitary transformation, the behaviour is deterministic. Otherwise the behaviour is probabilistic, according to the rules of quantum measurement (Curtis and Meyer, 2004).

3.2.2 Deterministic solution to the QCM's learning

In that case, if the QCM has a global three-qubit state in superposition, for example for state $|110\rangle$, the transition can be expressed as follows:

$$|110\rangle \xrightarrow{U} \frac{1-i}{2}|101\rangle+\frac{1+i}{2}|111\rangle. \tag{26}$$

At the end of the transition, the first and last qubits are unaffected by the measurement process, but if we assume deterministic learning, the required quantum circuit also satisfies the incomplete function for a single qubit (Miller et al., 2006). The more detailed view of the implementation allows us to make the next analysis based on symbols $U, U^{\dagger}$, and their algebra

$$UU = NOT,\ U^{\dagger}U^{\dagger} = NOT,\ UU^{\dagger} = U^{\dagger}U = I. \tag{27}$$

The *deterministic* quantum circuit is shown in Fig. 14.

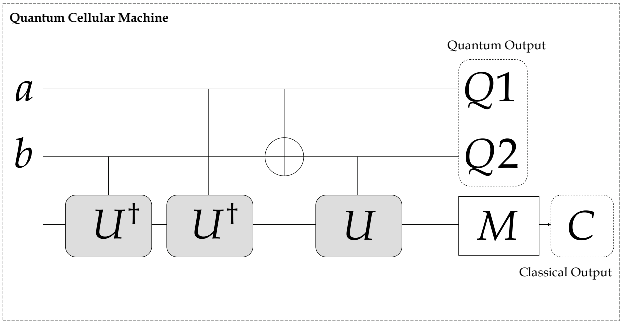

Fig. 14. A deterministic circuit to realize the learning of the QCM with incomplete function specification. The *don't cares* will be selected deterministically.

In Table 2, the example of analysis of the QCM's target signal *C* in the circuit from Fig. 14 is shown.

Target	0	1
Control: ab		
00	I	I
01	$UU^{\dagger}$	$UU^{\dagger}$
11	$U^{\dagger}U^{\dagger}$	$U^{\dagger}U^{\dagger}$
10	$U^{\dagger}U$	$U^{\dagger}U$

Table 2**.** Analysis of the QCM's target qubit in the circuit. The mapping of the UCM table for the incompletely defined specifications is deterministic.

Let us assume that the QCM gets an $|110\rangle$ input from the environment. The output of its quantum circuit can be derived as follows:

1. Since the second qubit is the control of the first $U^{\dagger}$ transformation, the QCM applies the *controlled-* $U^{\dagger}$ transformation on the third qubit.

$$\xrightarrow{controlled-U^{\dagger}} |11\rangle \frac{1-i}{\sqrt{2}}(|0\rangle + |1\rangle). \tag{28}$$

2. Since the first qubit is the control of the second $U^{\dagger}$ transformation, the QCM applies the *controlled-* $U^{\dagger}$ transformation on the third qubit.

$$\xrightarrow{controlled-U^{\dagger}} |110\rangle. \tag{29}$$

3. The QCM applies the *controlled-NOT* transformation:

$$\xrightarrow{CNOT} |101\rangle. \tag{30}$$

On the other hand, if the input is $|111\rangle$, then the output of the circuit is $|100\rangle$. As can be concluded from the results, the QCM generates the inverse of the target as the output only if *ab = 11*; otherwise it makes an identity transformation.

As we have seen, the QCM's quantum circuit realizes a deterministic output for input $|110\rangle$; on the other hand – as we will see here – it becomes quantum probabilistic for input $|100\rangle$. However, to achieve the quantum probabilistic behaviour of the QCM, we have to make a small modification on the circuit, as follows (Curtis and Meyer, 2004):

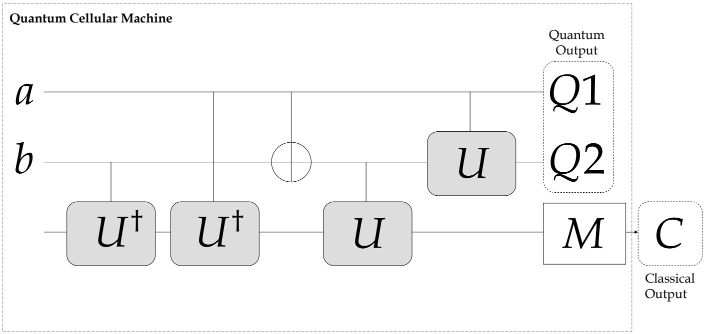

Fig. 15. The quantum probabilistic circuit.

Compared to the previous quantum circuit, we have added the *controlled-U* transformation to the circuit. As a result, after a measurement on the third qubit, the state of the second qubit becomes quantum probabilistic (Miller et al., 2006).

The new quantum circuit can generate *deterministic* output; however it also has quantum *probabilistic behaviour*. To see it, we analyze the QCM's transition for the input state $|100\rangle$ as follows.

1. Since the first qubit is the control of the $U^\dagger$ transformation, the QCM applies the *controlled-$U^\dagger$* transformation on the third qubit.

$$\xrightarrow{controlled-U^\dagger} |10\rangle \frac{1-i}{\sqrt{2}}\left(|0\rangle + |1\rangle\right). \tag{31}$$

2. The QCM applies the *controlled-NOT* transformation:

$$\xrightarrow{CNOT} |11\rangle \frac{1-i}{\sqrt{2}}\left(|0\rangle + |1\rangle\right). \tag{32}$$

3. Then, the QCM applies the *controlled-U* transformation on the third qubit, since the CNOT gate has changed the value of its control qubit b:

$$\xrightarrow{controlled-U} |110\rangle. \tag{33}$$

Finally, the QCM applies the gate's *controlled-U* on the second qubit:

$$\xrightarrow{controlled-U} |1\rangle \frac{1+i}{\sqrt{2}}\left(|0\rangle + |1\rangle\right)|0\rangle$$
$$\rightarrow \frac{1+i}{\sqrt{2}}\left(|100\rangle + |110\rangle\right). \tag{34}$$

The original *three-to-one incompletely specified function* is mapped to a *reversible* quantum *three-by-three* circuit. The map of the deterministic circuit with incomplete function specification will have *probabilistic behaviour* from the multiple qubit measurement. The output value of the second qubit is not determined after the measurement made on the third qubit. This output can be used to control other parts of the network or to realize truly random behaviour in the network.

Target	0	1
Control: ab		
00	$\|000\rangle$	$\|001\rangle$
01	$\|010\rangle$	$\|011\rangle$
11	$\|1\rangle\left[\frac{1+i}{\sqrt{2}}(\|0\rangle+\|1\rangle)\right]\|1\rangle$	$\|1\rangle\left[\frac{1+i}{\sqrt{2}}(\|0\rangle+\|1\rangle)\right]\|0\rangle$
10	$\|1\rangle\left[\frac{1+i}{\sqrt{2}}(\|0\rangle+\|1\rangle)\right]\|0\rangle$	$\|1\rangle\left[\frac{1+i}{\sqrt{2}}(\|0\rangle+\|1\rangle)\right]\|1\rangle$

Table 3. The map of the deterministic circuit with incomplete function specification and quantum probabilistic behaviour.

The quantum control based QCM forms a quantum system exploiting the *superposition* of the state of the QCM with the state of the network environment. In the presented quantum probabilistic QCM network the focus is not on how the QCM is implemented with respect to its environment, but rather on the *strategies* for learning the QCM network behaviours based on quantum circuit structures (Miller et al., 2006).

3.3 Conclusion and future work

The QCM's internal self model modifies the classical commands and the logical behaviour of the QCM on different levels. The QCM's internal self model is generated on the lowest level; the expressed logical behaviour is based on quantum measurements. The proposed quantum model demonstrates the quantum mechanical based model's superiority to classical cellular automata based self-organizing networks. We constructed a real biologically inspired non-deterministic network model, based on quantum mechanics, allowing the QCM to use quantum level communication on a higher logical level.

As future work we would like to extend our novel quantum mechanical based control mechanism to other biologically inspired self-organizing structures and to publish simulation results on the convergence rate of our quantum probabilistic learning method.

4. Quantum learning algorithm based controller QCM

In this part we define the extended version of the QCM – called the *controller QCM* – and show that it can be used for solving hard problems, such as network controlling, routing, and network organizing, using very efficient quantum searching and quantum learning algorithms. We define quantum algorithms for controlling the truly quantum probabilistic network structure, and we demonstrate the performance of the quantum-learning based QCM over the classical network controlling solutions.

4.1 Introduction

In this part, we define the extended version of QCM as the main controller element. This QCM is able to interact with the other network components – using both quantum and classical information – and can dynamically reorganize the activities to serve the dynamic needs in an adaptive and goal-oriented way. Moreover, the QCM has a deep impact on the self-organizing capability of the network.

The main component of the quantum probabilistic self-organizing network structure is the *controller QCM*, which offers and uses services that adapt without any human interaction to the changes of environment of the network, and creates plans and a knowledge map. The knowledge map represents a snapshot of the current network state. The primary task of the controller QCM is to use and provide services. It monitors the internal and external environment of the self-organizing network's structure, adapts itself to the changing conditions, and knows its capabilities and how to adapt. The model consists of two parts. The common part contains the same functionalities as a common QCM. The specific part contains advanced functions, such as the implementation of the quantum-learning algorithm. The controller QCM uses quantum computing for the development of network controlling, routing, path finding, and other problems relating to the effectiveness of self-organizing. The elements of the quantum probabilistic self-organizing networks are other, simpler QCMs, without integrated advanced quantum searching and learning methods. In the QCM model, quantum mechanics and quantum computing act on the QCM's quantum registers and it modifies its output. The quantum based communication appears in two major forms in our quantum network model, since both the self-organizing processes in the network are truly random and the controlling and routing tasks are also solved quantum mechanically. We present the quantum mechanical based learning and controlling algorithm of the QCM component model, which is the core of the quantum probabilistic self-organizing framework.

The actual *part* of this chapter is organized as follows. First, we introduce the basic properties of the controller QCM. In the section, we describe the properties of the extended version of the QCM module. At the end of this part, we conclude with the results and benefits of our quantum mechanical based model.

4.2 Related work

The quantum searching algorithm was presented by Grover et al. for quantum database searching (Imre and Balazs, 2005), (Nielsen and Chuang, 2000). The effectiveness of quantum searching is based on a fundamental property of quantum information processing, the quantum parallelism. The problem of quantum searching was developed to solve the problem of identification of an item in an unsorted database with N elements. This kind of sorting problem generally requires $N/2$ steps in classical systems, and hence the complexity of the problem is $\mathcal{O}(N)$ classically. With the help of quantum information processing the searching can be solved with complexity $\mathcal{O}\left(\sqrt{N}\right)$; hence, with the quantum based approach a quadratic increase can be achieved in the speed of the searching process (Imre and Balazs, 2005).

In this section we present an extended version of the QCM shown in the previous part of this chapter, using quantum searching and quantum learning algorithms to speed up the steps of self-organizing and other hard problems such as routing in self-organizing systems or path selection, and so on. One of the most important properties of quantum searching methods is that it can be implemented by elementary quantum circuit elements, and hence the implementation of the algorithm can be realized easily in practice. We implement a more advanced version of the quantum algorithm, compared to the original searching algorithm. We call it the quantum learning algorithm, and it is able to use the fundamental properties of quantum searching and can combine it with service and environment demands. The extended version of the QCM has the capability to control and sense the network and can find solutions for network-related problems using the quantum-searching based quantum learning method.

4.3 Problem discussion and motivation

The controller QCM reads the classical command from the self-organizing network, builds a map from the current network structure, and applies a unitary transform to modify the current state of the network. After the transformation the QCM measures the state of its final quantum register that affects the output. The actual state of the network model is stored in the QCM's initial quantum register. The result of the advanced quantum learning algorithm and the results of the quantum iteration processes are stored in the QCM's final quantum register, which then realizes a direct contact with the network. The structure of the network controller QCM has a different structure from the one presented in Section 2, since it is *equipped with advanced quantum-learning algorithms*. From an engineering point of view, the controller QCM is an extended version of the traditional QCM, equipped with advanced quantum learning methods. The traditional QCM is not implemented with the quantum-searching algorithm, hence its complexity is much higher than the controller QCMs.

In Fig. 16 we illustrated the decision mechanism of the controller QCM module. The input command is modified by the state of the internal quantum state, and the decisions are based on the result of the quantum learning algorithm. As can be concluded, the controller QCM can be viewed as an extended QCM with extended functions – such as quantum learning and quantum probabilistic decisions. The other parts of the self-organizing network structure can be realized by simpler QCMs, which are responsible only for simpler tasks such as self-organizing and so on.

The QCM uses the logical function to determine its output. The logical output is processed from the value of the QCM's final quantum register, which stores the result of the quantum learning algorithm. The result of the QCM's quantum probabilistic decision acts mainly on he network command language by rewriting the network structure.

In Fig. 17 we illustrate the connection between controller-QCMs and ordinary QCMs. The controller-QCMs have extended capabilities such as advanced quantum searching and learning methods. The ordinary QCMs communicate with each other and the controller-QCMs to realize the self-organizing network structure.

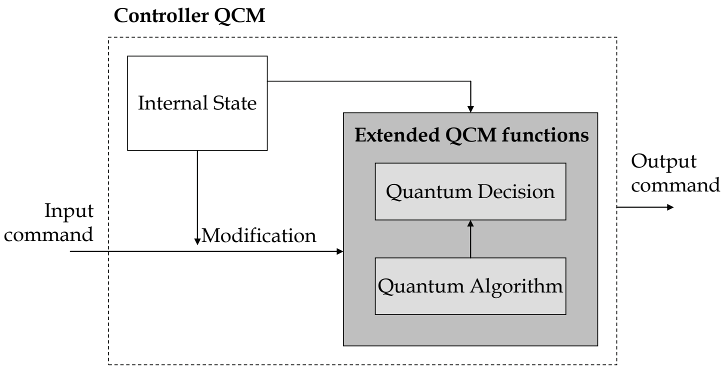

Fig. 16. The decision and output determination of the extended QCM.

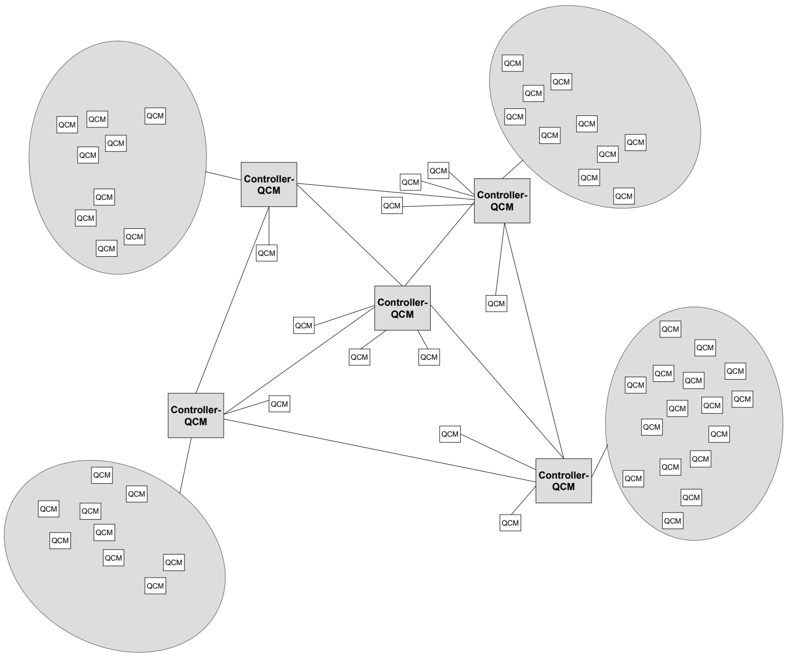

Fig. 17. The controlled self-organizing quantum probabilistic network structure.

To realize the controlling of the network, we implement a very efficient $\mathcal{O}\left(\sqrt{N}\right)$ quantum searching algorithm to control the self-organizing processes of the network and to find the best solution to the input command (Arrighi et al., 2007).

4.3.1 Extended QCM functions

The controller QCM senses and processes information from its external network environment. In this model, the external network is a quantum probabilistic self-organizing system. The QCM can also process quantum and classical information and accomplish deterministic and quantum probabilistic tasks. The information unit is a quantum bit, which can lie in the coherent superposition state of logical states zero and one, and thus it can simultaneously store zero and one. Using quantum bits, we can effectively speed up the solutions of the classical problems and even solve some hard problems that a classical computer cannot solve (Feynman, 1982). The key aspect behind the optimal decisions of QCM is the design of a high-efficiency searching algorithm. In our model, we use the ability of quantum parallel processing to design a corresponding quantum learning control algorithm, which can effectively reduce the complexity of solving problems and speed up information processing (Nielsen and Chuang, 2000).

4.3.2 Parallel processing of QCM module

As we mentioned before, our QCM is basically a complex quantum system, its state is also represented by the quantum state, and thus we encode all information according to quantum bits. The state of the QCM's internal $|\psi\rangle$ quantum state can be written into a superposition state as $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$, where α and β are complex coefficients and $|\alpha|^2 + |\beta|^2 = 1$. The states $|0\rangle$ and $|1\rangle$ are two orthogonal states; the eigenstates of $|\psi\rangle$ correspond to logic states zero and one. The $|\alpha|^2$ represents the occurrence probability of $|0\rangle$ when the quantum state is measured, and $|\beta|^2$ is the probability of obtaining result $|1\rangle$. The value of a classical bit is either zero or one; however a quantum state can be prepared in the coherent superposition state of zero and one. A quantum bit can simultaneously store zero and one, which is one of the main differences between quantum and classical information processing. If the QCM applies a unitary transformation U to a superposition state, the transformation will act on all eigenstates of the superposition state $|\psi\rangle$, and the output will be a new superposition state obtained by superposing the results of eigenstates. If the QCM processes function $f(x)$, the transformation U can simultaneously work out many different results for a certain input x.

The ability of strong parallel processing is a very important advantage of our QCM's module over traditional QCM. Let us consider an n-qubit quantum state which lies in a superposition state:

$$|\psi\rangle = \sum_{x=00\ldots0}^{11\ldots1} c_x |x\rangle, \tag{35}$$

where $\sum_{x=00\ldots0}^{11\ldots1} |c_x|^2 = 1$, and c_x are the complex coefficients. The state $|x\rangle$ has 2^n values; it contains all integers from 1 to 2^n. Since U is linear, processing function $f(x)$ can be expressed as follows:

$$U \sum_{x=00\ldots0}^{11\ldots1} c_x |x,0\rangle = \sum_{x=00\ldots0}^{11\ldots1} c_x U |x,0\rangle = \sum_{x=00\ldots0}^{11\ldots1} c_x U |x, f(x)\rangle, \tag{36}$$

where $|x,0\rangle$ represents the input joint state and $|x, f(x)\rangle$ is the output joint state. The controller QCM uses the superposition principle of quantum states, and hence with an n-qubit quantum register it can simultaneously process 2^n states.

4.4 Quantum circuits of quantum searching

In our quantum searching algorithm, the controller QCM updates the probability amplitudes of the quantum register according to a given reward value derived from the network environment. The quantum searching process can be implemented using the Hadamard-transformations and conditional phase shift operations. Through the Hadamard-gate, a quantum bit in the state $|0\rangle$ or $|1\rangle$ is transformed into a superposition state of two states as

$$H|0\rangle = \frac{1}{\sqrt{2}}\left(|0\rangle + |1\rangle\right) \text{ or } H|1\rangle = \frac{1}{\sqrt{2}}\left(|0\rangle - |1\rangle\right), \tag{37}$$

that is, the magnitude of the amplitude in each state is $\frac{1}{\sqrt{2}}$, but the phase of the amplitude in the state $|1\rangle$ is inverted. Let us consider a quantum system described by n quantum bits which has 2^n possible states. To prepare an equally weighted superposition state, the controller QCM performs the transformation H on each qubit independently. The state transition matrix representing this operation will be of the dimensions $2^n \times 2^n$ and can be implemented by n Hadamard-gates. The process can be represented as:

$$H^{\otimes n}\left|\overbrace{00\ldots0}^{n}\right\rangle = \frac{1}{\sqrt{2^n}}\sum_{a=00\ldots0}^{\overbrace{11\ldots1}^{n}}|x\rangle. \tag{38}$$

According to the above method, we can accomplish the initialization of state and action. To realize quantum searching, the controller QCM has to make a Hadamard-transformation on the initial quantum register. In the searching process, it selects the solutions from the quantum register using probability amplitude amplification, and finally applies a Hadamard-transformation to obtain the answer sought for the input problem.

In Fig. 18 we illustrated the general scheme of the controller QCM's searching process. The classical network input is stored in its internal quantum register, and the Hadamard-transformations and the searching process are defined only in quantum space.

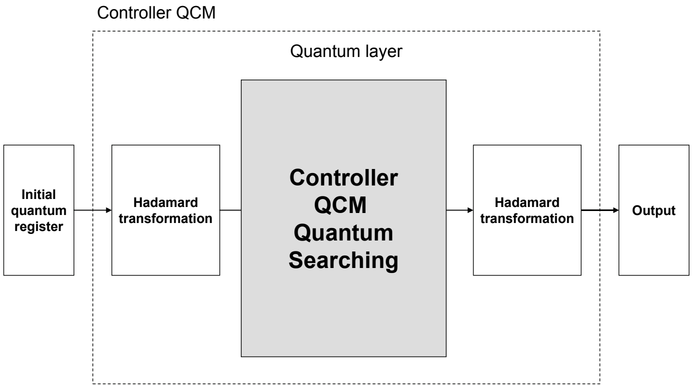

Fig. 18. The general model of the controller QCM's quantum searching module

The conditional phase-shift operation is an important element to carry out the quantum searching iteration. The phase-shift transformation ϕ for a two-state system can be expressed as

$$\phi = \begin{pmatrix} e^{i\phi_1} & 0 \\ 0 & e^{i\phi_2} \end{pmatrix}, \tag{39}$$

where $i=\sqrt{-1}$ and ϕ_1, ϕ_2 are arbitrary real numbers. The conditional phase-shift operation does not change the probability of each state, since the square of the absolute value of the amplitude in each state is the same. To update probability amplitudes we reinforce the selected decision corresponding to a larger reward value through repeating the quantum searching process T times. We initialize the action

$$f(s) = \left|x_s^{(n)}\right\rangle = \frac{1}{\sqrt{2^n}} \sum_{a=00\ldots0}^{\overbrace{11\ldots1}^{n}} |x\rangle. \tag{40}$$

Then we construct a reflection transform

$$U_x = 2\left|x_s^{(n)}\right\rangle\left\langle x_s^{(n)}\right| - I, \tag{41}$$

which preserves $\left|x_s^{(n)}\right\rangle$, but flips the sign of any vector orthogonal to $\left|x_s^{(n)}\right\rangle$.

Geometrically, if U_a acts on an arbitrary vector, it preserves the component along $\left|x_s^{(n)}\right\rangle$ and flips the component in the hyperplane orthogonal to $\left|x_s^{(n)}\right\rangle$, and thus it can be viewed as an operation of inversion about the mean value of the amplitude (Nielsen and Chuang, 2000). The controller QCM exchanges $\left|x_s^{(n)}\right\rangle$ with the k-th computational basis state $\left|x_k\right\rangle$, and constructs another reflection transformation $U_{x_k} = I - 2\left|x_k\right\rangle\left\langle x_k\right|$. Thus we can form a unitary transformation

$$U_I = U_s U_k = U_x U_{x_k}. \tag{42}$$

It repeatedly applies the transformation U_I on $\left|x_s^{(n)}\right\rangle$, and thus it can enhance the probability amplitude of the k-th path in the quantum probabilistic self-organizing network environment, denoted by $\left|x_k\right\rangle$, while suppressing the amplitude of all other actions.

In Fig. 19 we illustrated the realization of reflection and rotation transformations. The initial state is denoted by $|\psi\rangle$, and the reflected state along the L-axis is denoted by $|\psi'\rangle$. The angle of the rotation process is denoted by θ.

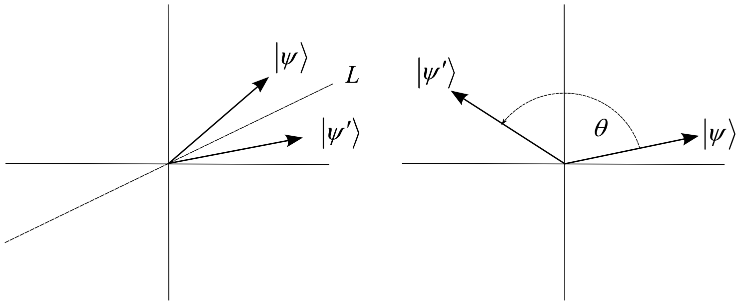

Fig. 19. The geometrical representation of reflection and rotation.

This process can be viewed as a rotation in a two-dimensional space, and the initial decision $f(s)$ can be expressed as (Nielsen and Chuang, 2000):

$$f(s) = \left|x_s^{(n)}\right\rangle = \frac{1}{\sqrt{2}}\left|x_k\right\rangle + \sqrt{\frac{2^n - 1}{2^n}}\left|\varphi\right\rangle, \tag{43}$$

where $\left|\varphi\right\rangle = \sqrt{\frac{1}{2^n - 1}} \sum_{a \neq a_k} \left|x\right\rangle$. We define the rotation angle θ satisfying $\sin\theta = \frac{1}{\sqrt{2^n}}$, and thus

$$f(s) = \left|x_s^{(n)}\right\rangle = \sin\theta\left|x_k\right\rangle + \cos\theta\left|\varphi\right\rangle. \tag{44}$$

The controller QCM applies the quantum searching iteration U_I T times on $\left|x_s^{(n)}\right\rangle$, and thus

$$U_I^T\left|x_s^{(n)}\right\rangle = \sin\left((2T+1)\theta\right)\left|x_k\right\rangle + \cos\left((2T+1)\theta\right)\left|\varphi\right\rangle. \tag{45}$$

By repeating the quantum iteration operator, the controller QCM can reinforce the probability amplitude of the corresponding decision according to the feedback value (Gyongyosi et al., 2009). The QCM's searching algorithm only applies the quantum iteration operator to reinforce the possible paths in the network environment.

4.4.1 General description of searching problem

In classical computation, an unstructured searching problem of *searching space N*, the classical algorithm complexity is $\mathcal{O}(N)$. In our network it is an important task to search for a suitable decision from quantum probabilistic self-organizing network space, based on the current state of the QCM.

If the complexity of the state or *decision space* is $\mathcal{O}(N)$, the problem complexity in a traditional QCM is $\mathcal{O}(N^2)$; since it is not equipped with the quantum searching algorithm. The extended QCM can reduce the complexity to $\mathcal{O}(N\sqrt{N})$ by using a quantum searching algorithm. It means, that for the *searching space N*, the QCM has to apply only $\sqrt{N}$ steps to find the solution, instead of the classical N steps. Thus, if the QCM has N possible actions, where $2^n \geq N \geq 2^{n-1}$, we can prepare an equally weighted superposition quantum state

$$|s\rangle = \frac{1}{\sqrt{2^n}} \sum_{i=1}^{2^n} |i\rangle. \tag{46}$$

This quantum state can be accomplished by applying the Hadamard-transformation to each quantum bit of the n-qubit state $|00\ldots00\rangle$. Then, we construct a reflection transform

$$U_s = 2|s\rangle\langle s| - I. \tag{47}$$

If the QCM applies U_s on an arbitrary vector, it preserves the component along $|s\rangle$ and flips the component orthogonal to $|s\rangle$, and thus if we apply U_s to $|\psi_0\rangle$ we get

$$\begin{aligned} U_s|\psi_0\rangle = 2|s\rangle\langle s|\psi_0\rangle - |\psi_0\rangle = 2|s\rangle\sqrt{2^n}\frac{1}{2^n}\sum_{i=1}^{2^n} x_i - |\psi_0\rangle = \\ 2\frac{1}{\sqrt{2^n}}\sum_{i=1}^{2^n}|i\rangle\sqrt{2^n}\frac{1}{2^n}\sum_{i=1}^{2^n}x_i - \sum_{i=1}^{2^n}x_i|i\rangle = \sum_{i=1}^{2^n}\left(2\frac{1}{2^n}\sum_{i=1}^{2^n}x_i - x_i\right)|i\rangle. \end{aligned} \tag{48}$$

Then, we use another reflection transform

$$U_k = -2|k\rangle\langle k| + I, \tag{49}$$

where $|k\rangle$ is the k-th eigenstate, and by applying U_k to state $|\psi_0\rangle$, we obtain

$$U_k|\psi_0\rangle = -2|k\rangle\langle k|\psi_0\rangle + |\psi_0\rangle = -2|k\rangle x_k + |\psi_0\rangle = \sum_{i=1, i\neq k}^{2^n} x_i|i\rangle - x_k|k\rangle. \tag{50}$$

The transformation U_k only changes the amplitude's sign of $|k\rangle$ in the superposition quantum state, and thus we can form a U_I unitary quantum iteration transformation:

$$U_I = U_s U_k. \tag{51}$$

If the QCM repeatedly applies the iteration transformation U_I on $|\psi_0\rangle$, it enhances the probability amplitude of $|k\rangle$, while suppressing the amplitude of all other states $|i \neq k\rangle$. By applying the iteration transformation enough times, the QCM can make state $|\psi_0\rangle$ collapse into state $|k\rangle$ with a probability of almost 1. In the description of the iteration process, we define an angle θ, which satisfies the equation $\sin^2\theta = \frac{1}{2^n}$. After applying the iteration U_I to $|\psi_0\rangle$ j times, the probability amplitude of state $|k\rangle$ becomes

$$x_k^j = \sin\left((2j+1)\theta\right). \tag{52}$$

The QCM's rotation process is illustrated in Fig. 20. In every iteration step, the QCM rotates state $|\psi\rangle$ into $G|\psi\rangle$.

4.4.2 Changing the probability amplitude through the Iterations

The "good" and "bad" answers to the input problem are denoted by basis states $|A\rangle$ and $|B\rangle$, respectively. In every iteration step, the QCM tries to get closer to the good answer, denoted by state $|A\rangle$.

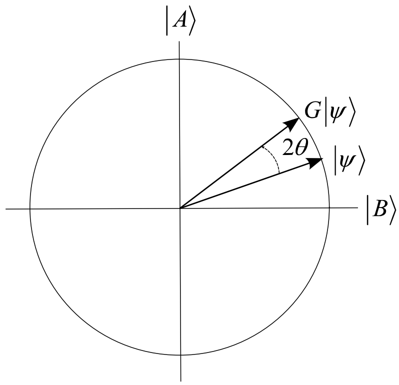

Fig. 20. In every iteration step, the QCM rotates the initial state by a given phase.

If the QCM applies the iteration U_I to $|\psi_0\rangle$ $j = \frac{\pi - 2\theta}{4\theta}$ times, then $(2j+1)\theta \approx \frac{\pi}{2}$, and thus

$$x_k^j = \sin\left(\frac{\pi}{2}\right) = 1. \tag{53}$$

However, the QCM must perform an integer number of iterations (Nielsen and Chuang, 2000), and thus we have to calculate with some probability of failure, which is no more than $1/2^N$, if the QCM performs the iteration U_I $\text{int}\left[\frac{\pi}{4\theta}\right]$ times.

In Fig. 21 we illustrated the searching process as a series of different rotations. The angle between two lines L_1 and L_2 is θ. Rotation of state $|\psi\rangle$ by angle 2θ can be realized by two reflections. We reflect $|\psi\rangle$ first about L_1 and then about L_2, and we can conclude that state $|\psi\rangle$ is rotated by angle 2θ.

The QCM uses quantum searching to make the optimal decision, and the theoretical results show that QCM can reduce the complexity of $\mathcal{O}(N^2)$ in traditional QCM to

$$\mathcal{O}(N\sqrt{N}) \tag{54}$$

using the quantum searching algorithm. If N is large the QCM can find the optimal decision with a high probability of $1 - \mathcal{O}\left(\frac{1}{N}\right)$.

Using the quantum iteration, the QCM can find the needed result with a probability of almost 1 in $\sqrt{N}$ steps. As we can conclude, the QCM dramatically reduces the complexity of the searching process, and it can make a suitable decision based on the current state of the QCM.

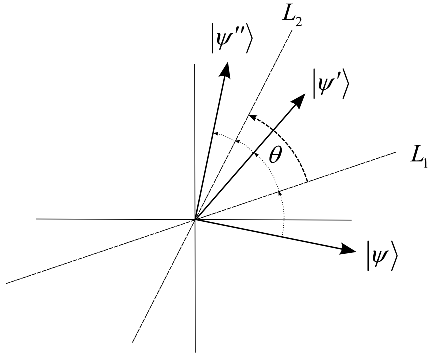

Fig. 21. At the end of the iteration process, the QCM rotates the initial state close to the right basis state.

4.5 Optimal QCM decisions

In the searching problem, the QCM wants to find a logical path from a given network node *A* to network node *B* in the network environment, while no available data source tells it how to achieve the goal. The QCM must accumulate experience by itself and become more intelligent through learning from its self-organizing network environment. At a certain iteration step, the QCM observes the state of the environment S_t, inside and outside the QCM.

The QCM makes a decision d_t in which the QCM chooses a path in the network environment, and afterwards the QCM receives feedback g_{t+1} which reflects how good that selected path is. The goal of the QCM decision is to realize a mapping from states to decisions, and to make a connection from logical state *A* to logical state *B* in the quantum probabilistic self-organizing network environment, with a minimum cost. The QCM makes the decisions based on the policy

$$\pi : S \times \cup_{i \in S} D_{(i)} \rightarrow [0,1], \tag{55}$$

so that the expected (*E*) sum of the discounted feedback of each state will be maximized (Nielsen and Chuang, 2000):

$$\begin{aligned} V_{(s)}^{\pi} &= E\left\{ g_{t+1} + \gamma g_{t+2} + \gamma^2 g_{t+3} + \ldots \middle| s_t = s, \pi \right\} = E\left[g_{t+1} + \gamma V_{(s_{t+1})}^{\pi} \middle| s_t = s, \pi \right] \\ &= \sum_{d \in D_s} \pi(s,d) \left[g_s^d + \gamma \sum_{s'} p_{ss'}^d V_{(s')}^{\pi} \right], \end{aligned} \tag{56}$$

where $\gamma \in [0,1]$ is the discount factor and $\pi(s,d)$ is the probability of selecting a given path in the self-organizing structure. The output of the path selection is based on decision d

according to state s. Under policy π, the probability of state transition is $p_{ss'}^{d} = \Pr\{s_{t+1} = s' | s_t = s, d_t = d\}$ and the expected one-step feedback is $g_s^d = E\{g_{t+1} | s_t = s, d_t = d\}$. The QCM's quantum searching algorithm could help to find an optimal logical path from network node A to node B using the effects of quantum mechanics. The output of the controller QCM is propagated back to the classical network layer as we have illustrated in Fig. 22.

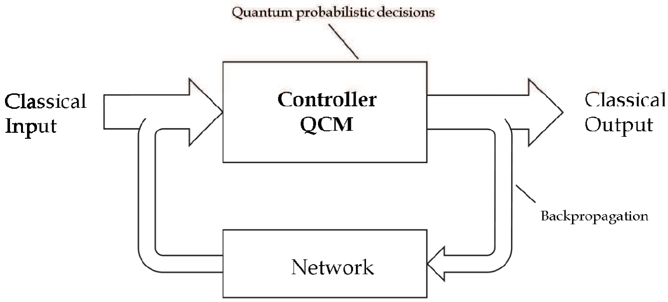

Fig. 22. The QCM communicates with the classical network layer.

The QCM can be represented as a dynamical quantum system which is controlled by the classical network layer (Gyongyosi et al., 2009).

4.5.1 Finding the optimal solution

The output of the QCM is based on a state-decision pair $\{State(t), Decision(t)\}$. The QCM's quantum decision process uses a scalar value, named feedback, to reflect how good that selected path is.

We propose a novel quantum learning method inspired by quantum superposition and quantum parallelism. Let N_S and N_d be the number of states and decisions of the QCM, and let m, n be numbers which are characterized by the following equations:

$$N_S \leq 2^m \leq 2N_S,\ N_d \leq 2^n \leq 2N_d. \quad (57)$$

The QCM uses m and n quantum bits to represent state set $S = \{s\}$ and decision set $D = \{d\}$. The learning procedure was inspired by the superposition principle of quantum states and quantum parallel computation. The occurrence probability of the eigenvalue is denoted by the probability amplitude of the quantum state, which is updated according to feedback. To realize the searching process, the QCM first initializes state $\left|s^{(m)}\right\rangle = \sum_{s=00\ldots0}^{\overbrace{11\ldots1}^{m}} c_S |s\rangle$, thus mapping from states to decisions as $f(s) = \pi : S \to D$, where $f(s) = \left|d_s^{(n)}\right\rangle = \sum_{a=00\ldots0}^{\overbrace{11\ldots1}^{n}} c_d |d\rangle$, and c_s, c_d are the probability amplitudes of state $|s\rangle$ and decision $|d\rangle$.

In the learning process, the QCM initializes the state and decision to the equal superposition state $\left|s^{(m)}\right\rangle$ as follows:

$$\left|s^{(m)}\right\rangle = \frac{1}{\sqrt{2^m}} \sum_{s=00\ldots0}^{\overbrace{11\ldots1}^{m}} |s\rangle, \tag{58}$$

with map function

$$f(s) = \left|d_s^{(n)}\right\rangle = \frac{1}{\sqrt{2^n}} \sum_{s=00\ldots0}^{\overbrace{11\ldots1}^{n}} |d\rangle. \tag{59}$$

After the initialization phase, the QCM repeats for all states

$$\left|s^{(m)}\right\rangle = \sum_{s=00\ldots0}^{11\ldots1} c_S |s\rangle = \frac{1}{\sqrt{2^m}} \sum_{s=00\ldots0}^{\overbrace{11\ldots1}^{m}} |s\rangle \tag{60}$$

the following algorithm:

Algorithm

1. Observe $f(s) = \left|d_s^{(n)}\right\rangle$ and get decision $|d\rangle$;
2. Take decision $|d\rangle$ and observe next state $\left|s'^{(m)}\right\rangle$ and reward g from the network
   2.1. The QCM updates state value $V(s)$:
   $V(s) \leftarrow V(s) + \alpha\left(g + \gamma V(s') - V(s)\right)$.
3. Update probability amplitudes by repeating the quantum searching iteration U_I T times:
   $U_I^T \left|d_s^{(n)}\right\rangle = \left[U_x U_{x_k}\right]^T \left|d_s^{(n)}\right\rangle$, and $c_x \leftarrow e^{\lambda(r+V(s'))} c_x$.

The initial register of the QCM is illustrated in Fig. 23. In the initial phase, the QCM's quantum register contains

$$|\psi\rangle = H|0\rangle = \frac{1}{\sqrt{N}} \sum_{x=0}^{N-1} |x\rangle, \tag{61}$$

where every state has the same probability amplitude $\frac{1}{\sqrt{N}}$.

In Fig. 24 we illustrate the QCM's quantum register after the iteration steps of the searching process. The state which represents the answer to the question of the QCM has higher

probability while the "bad" states have lower probability amplitudes than the average probability amplitude in the initial quantum register.
The QCM's quantum searching algorithm is inspired by the superposition principle of quantum states and the power of parallel quantum computations.

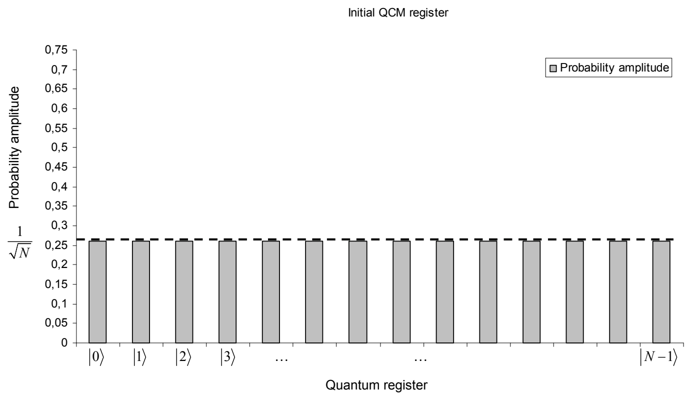

Fig. 23. The probability amplitudes of the QCM's initial quantum register.

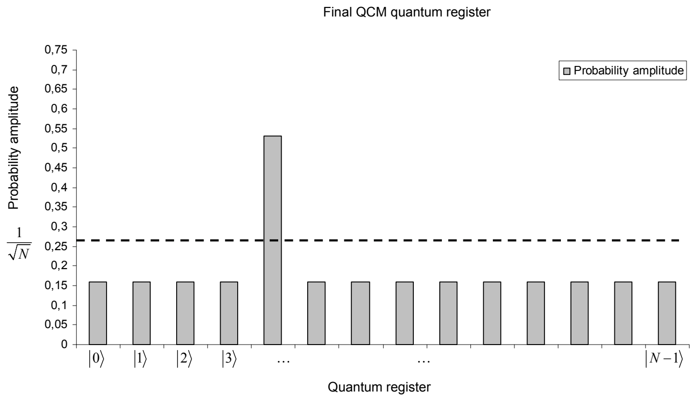

Fig. 24. The probability amplitudes of the QCM's states after the quantum searching process.

4.6 Numerical analysis

Our numerical analysis is based on the shortest path problem, in which the QCM has to find the shortest path in the network environment from communication element A to network element B. The problem can be stated as a searching problem in an unsorted database.
In the numerical analysis we compare the performance of traditional QCM and QCM. The QCM has to find the shortest path in the network from a given network element to a given network component, determined by the classical network command. The QCM puts the initial input network command into a quantum register, and it applies the quantum searching. The searching process is based on the classical input command and the QCM's internal state. The internal state describes the actual network state, and the quantum

searching algorithm seeks the best solution which describes the shortest path between the network elements.
In Fig. 25 we illustrate the inputs of the QCM's searching module.

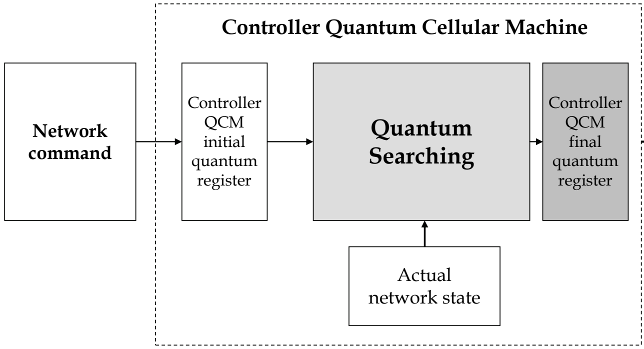

Fig. 25. The realization of the QCM's quantum searching process.

In the numerical analysis, we have assumed that the *shortest path* between nodes *A* and *B* defined in the input command contains *20 network nodes*.
In Fig. 26 we illustrate the number of required steps as function iterations. As we can conclude, the traditional QCM converges after 2500 steps, while the QCM finds the optimal path after 20 steps.

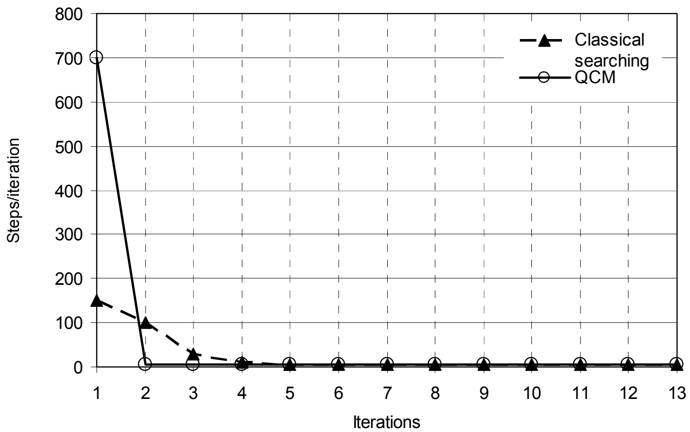

Fig. 26. The number of iterations required to find the optimal solution with traditional and extended QCM.

The QCM converges very fast to the optimal solution; however in the initial phase of the searching process, the number of steps per iteration is significantly higher. We can conclude that in the initial phase the QCM has higher uncertainty than the traditional QCM. The QCM finds the solution exponentially faster, and can find the solution much faster than the classical implementation. From the simulation results, we can conclude that the QCM's

quantum searching method needs only a few iteration steps to find the optimal way in the network environment.
The numerical results for the number of steps per iteration with traditional QCM are illustrated in Fig. 27.

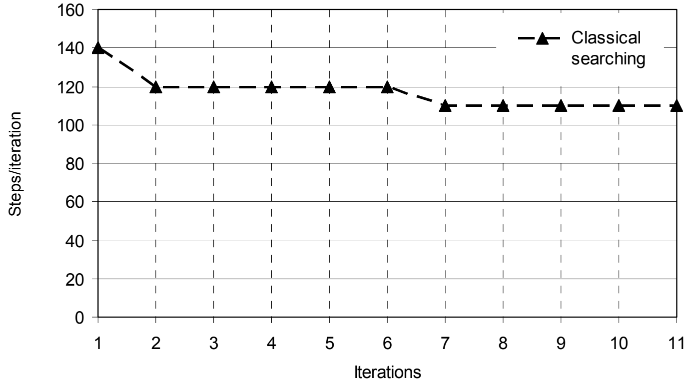

Fig. 27. The required number of classical searches after 50 iterations is still much higher than the optimal one.

We conclude that the quantum searching based QCM converges much faster than the classical one and the speed of the iteration process increments dramatically.

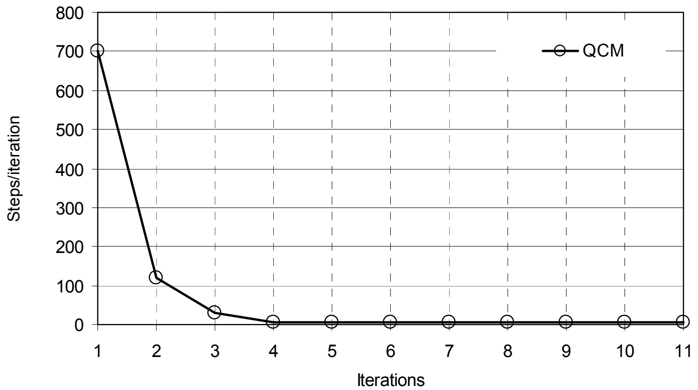

Fig. 28. The QCM finds the solution after 20 iterations, while the classical solution converges only after 2500 iterations.

From our numerical results we can conclude that the QCM module can help to improve he performance of truly random networks such as the described quantum probabilistic self-organizing networks and the overall performance of quantum probabilistic self-organizing communication systems and future Internet services.

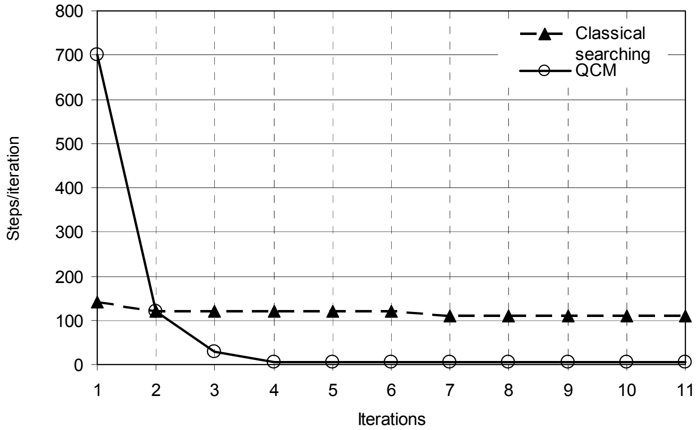

Fig. 29. Comparison of the performance of classical searching and QCM.

The QCM module can be used to dramatically speed up network controlling and it can be integrated efficiently into classical architectures.

4.7 Conclusion and future work

The QCM's internal self model modifies the classical commands and the logical behaviour of the QCM on different levels. The QCM's internal self model is generated on the lowest level, and the expressed logical behaviour is based on quantum measurements.

In this part we constructed a truly random non-deterministic network model based on quantum mechanics, allowing the QCM to use quantum level communication and quantum searching. The speed of QCM decision mechanisms can be increased using powerful parallel computing and fast searching ability. In our model, we integrate quantum searching to find the best solution to the given problem, encoded in the input network command of the QCM. In the numerical analysis we showed that the quantum communication layer could improve the performance of classical systems. The performance of the communication elements of quantum probabilistic self-organizing networks could be increased dramatically by quantum searching. The QCM's searching algorithm is based on the superposition principle of quantum states, whose behaviour cannot be described classically.

5. Conclusions

Quantum computing offers fundamentally new solutions in the field of computer science. The classical biologically inspired self-organizing systems have increasing complexity and these constructions do not seem to be suitable for handling the service demands of the near future. The cell-organized, quantum mechanics based cellular automata models have many advantages over classical models and circuits. As we have seen, for a quantum cellular machine, every cell is a finite-dimensional quantum system with unitary transformations, and there is a difference between the axiomatic structure of classical and quantum versions of cellular automata. To see clearly the advantages of quantum information processing based solutions, we have discussed the parallel address mechanisms of quantum cellular machines.

In Section 2, we have given a brief overview of quantum mechanics, such as the definition of a quantum bit, the postulates of quantum mechanics, and the basics of quantum algorithms. In Sections 3 and 4, we have exhibited an application of a quantum cellular automata model to a quantum probabilistically controlled, self-organizing biological network structure. The network control mechanisms of this self-organizing structure are performed by an extended version of a quantum cellular machine, with integrated quantum searching processes and built-in quantum learning algorithms. In the proposed searching process, a QCM selects the solutions from the quantum register, using probability amplitude amplification, and finally applies a Hadamard-transformation, to get the searched-for answer to the input problem. As we have presented, the performance of the communication of self-organizing biological networks could be increased dramatically by quantum searching. The QCM's searching algorithm is based on the superposition principle of quantum states, which behavior cannot be described classically.

As future work we would like to extend our quantum learning algorithm to other network elements, and we would like to integrate our method into truly random self-organizing networks.

6. References

Arrighi, P.; Nesme, V. & Werner, R. (2007). N-dimensional quantum cellular automata. 0711.3975.

Benioff, P. (1980). The computer as a physical system: A microscopic quantum mechanical hamiltonian model of computers as represented by Turing machines. *Journal of Statistical Physics*, 22:563–591.

Curtis, D. & Meyer, D. (2004). Towards quantum template matching, pp. 134–141.

Dam, W. (1996). Quantum cellular automata. Master's thesis, University of Nijmegen.

Feynman, R. (1982). Simulating physics with computers. *International Journal of Theoretical Physics*, 21:467–488.

Gardner, M. (1970). "Mathematical Games: The fantastic combinations of John Conway's new solitaire game "Life"". *Scientific American* 223: 120–123

Grössing, G. & Zeilinger, A. (1988). Quantum cellular automata. *Complex Syst.*, 2:197–208.

Gyongyosi, L.; Bacsardi, L. & Imre, S. (2009). Novel Approach for Quantum Mechanical Based Autonomic Communication, In *FUTURE COMPUTING 2009 Proceedings*, The First International Conference on Future Computational Technologies and Applications, pp. 586-590., Athen, Greece.

Imre, S. & Balazs, F. (2005). *Quantum Computing and Communications – An Engineering Approach*, Published by John Wiley and Sons Ltd.

Margolus, N. (1991). Parallel quantum computation. In: W. H. Zurek, editor, *Complexity, Entropy, and the Physics of Information, Santa Fe Institute Series*, pages 273–288. Addison Wesley, Redwood City, CA.

Meyer, D. A. (1996). From quantum cellular automata to quantum lattice gases. *Journal of Statistical Physics*, 85:551–574.

Miller, D. M.; Maslov, D. & Dueck, G. W. (2006). Synthesis of quantum multiple-valued circuits, *Journal of Multiple-Valued Logic and Soft Computing*, vol. 12, no. 5-6, pp. 431–450.

Neumann, J. (1966). *Theory of Self-Reproducing Automata*. University of Illinois Press, Champaign, IL, USA.

Nielsen, M. A. & Chuang, I. L. (2000). *Quantum Computation and Quantum Information*, (Cambridge University Press.

Perez-Delgado, C. A. & Cheung, D. (2005). Models of quantum cellular automata.

Richter & Werner. (1996). Ergodicity of quantum cellular automata. *Journal of Statistical Physics*, 82:963–998.

Toffoli, T. & Margolus, N. H. (1990). Invertible cellular automata: A review. *Physica D: Nonlinear Phenomena*, 45:229–253.

Toth, G. & Lent, C. S. (2001). Quantum computing with quantum-dot cellular automata. *Physical Review A*, 63:052315.

Vollbrecht, K. G. H. & Cirac (2008), J. I. Quantum simulators, continuous-time automata, and translationally invariant systems. *Phys. Rev. Lett.*, 100:010501, 2008.

Watrous, J. (1995). On one-dimensional quantum cellular automata. In: *Proceedings of the 36th Annual Symposium on Foundations of Computer Science*, pages 528–537.

Wootters, W. K. & Zurek, W. H. (1982). A single quantum cannot be cloned. *Nature*, 299:802–803.

Cellular Automata - Innovative Modelling for Science and Engineering

Edited by Dr. Alejandro Salcido

ISBN 978-953-307-172-5
Hard cover, 426 pages
Publisher InTech
Published online 11, April, 2011
Published in print edition April, 2011

Modelling and simulation are disciplines of major importance for science and engineering. There is no science without models, and simulation has nowadays become a very useful tool, sometimes unavoidable, for development of both science and engineering. The main attractive feature of cellular automata is that, in spite of their conceptual simplicity which allows an easiness of implementation for computer simulation, as a detailed and complete mathematical analysis in principle, they are able to exhibit a wide variety of amazingly complex behaviour. This feature of cellular automata has attracted the researchers' attention from a wide variety of divergent fields of the exact disciplines of science and engineering, but also of the social sciences, and sometimes beyond. The collective complex behaviour of numerous systems, which emerge from the interaction of a multitude of simple individuals, is being conveniently modelled and simulated with cellular automata for very different purposes. In this book, a number of innovative applications of cellular automata models in the fields of Quantum Computing, Materials Science, Cryptography and Coding, and Robotics and Image Processing are presented.

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821