1

Deterministic event-based simulation of universal quantum computation

K. Michielsen[1], H. De Raedt[1], and K. De Raedt[2]

[1]Applied Physics - Computational Physics, Materials Science Centre, University of Groningen, Nijenborgh 4, NL-9747 AG, Groningen, The Netherlands
[2]Department of Computer Science, University of Groningen, Blauwborgje 3, NL-9747 AC, Groningen, The Netherlands

Summary. We demonstrate that locally connected networks of classical processing units that have primitive learning capabilities can be used to perform a deterministic, event-based simulation of universal quantum computation. The new simulation method is applied to implement Shor 's factoring algorithm.

1.1 Introduction

The basic ideas of quantum computation were formulated more than twenty years ago [1,2]. Ten years ago DiVincenzo has proven that the CNOT gate and single-qubit operations constitute a set of gates that can be used to construct a universal quantum computer [3]. This statement is equivalent to the one that a digital classical computer can be constructed by means of NAND gates only. Conventional computers can simulate the physical behavior of quantum computer hardware by solving the (time-dependent) Schrödinger equation [4, 5]. However, the idea of building a quantum computer has lead to many advances in nanotechnology, making it possible to control individual ions, atoms, photons and the like and these single events cannot be described by quantum theory. In this paper we demonstrate how a quantum computer can be built from locally-connected networks of processing units with a primitive learning capability that deterministically generate events with frequencies that agree with the corresponding quantum mechanical probabilities.

1.2 Stochastic learning machines (SLMs)

In [6] we explained in detail how a DLM works. Here we describe its stochastic variant. The term stochastic refers to the method that is used to select the output channel that will carry the outgoing message. Only changes are made to the learning algorithm of the output DLM, DLMo, described in [6]. It can be proven that in the stationary regime $x_0^2+x_1^2$ and $x_2^2+x_3^2$, where x_0, x_1, x_2 and x_3 are the four elements of the internal vector $\mathbf{x}$ of a DLM, correspond to the probabilities of quantum theory. Instead of sending out messages in a deterministic way as described in [6], we choose a random number $0<r<1$ and send out a zero event if $x_1^2+x_2^2 \le r$ and a one event otherwise. Although the learning process of this processor is still deterministic, in the stationary regime the output events are randomly distributed over the two possibilities. Replacing DLMs by SLMs in a DLM-network changes the order in which messages are being processed but leaves the content of the messages intact.

1.3 Universal quantum computation: Concepts

Qubit. – The state $|\Phi\rangle$ of a qubit can be written as $|\Phi\rangle = a_0|0\rangle + a_1|1\rangle$, where a_0 and a_1 are complex numbers so that $|a_0|^2+|a_1|^2=1$. In general, the state $|\Phi\rangle$ of the qubit can also be written as $|\Phi\rangle = \sqrt{p_0}e^{i\psi_0}|0\rangle + \sqrt{p_1}e^{i\psi_1}|1\rangle$, where p_0 and $p_1 = 1-p_0$ denote the probability to find the qubit in state $|0\rangle$ and $|1\rangle$, respectively, and ψ_0 and ψ_1 denote phase factors.

In principle, any physical object that carries a $S=1/2$ degree-of-freedom can serve as a physical realization of a qubit [7]. The state $|\phi\rangle$ of a qubit can therefore also be written as a linear combination of the spin-up and spin-down states [8] $|\Phi\rangle = a_0|\uparrow\rangle + a_1|\downarrow\rangle$, where $|\uparrow\rangle = |0\rangle = (10)^T$ and $|\downarrow\rangle = |1\rangle = (01)^T$ [7]. The three components of the spin-1/2 operator $\mathbf{S}=(S^x,S^y,S^z)$ are defined (in units such that $\hbar=1$) by [8]

$$S^x = \frac{1}{2}\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad S^y = \frac{1}{2}\begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \quad S^z = \frac{1}{2}\begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}, \tag{1.1}$$

and have been chosen such that $|\uparrow\rangle$ and $|\downarrow\rangle$ are eigenstates of S^z with eigenvalues +1/2 and -1/2, respectively. The expectation values of the three components of the qubits are defined as $\langle Q^\alpha\rangle = 1/2 - \langle S^\alpha\rangle$, $\alpha = x,y,z$, where $\langle A\rangle = \langle\Phi|A|\Phi\rangle/\langle\Phi|\Phi\rangle$. A qubit is in the state $|0\rangle$ or $|1\rangle$ if $\langle Q^z\rangle = 0$ or $\langle Q^z\rangle = 1$, respectively.

Single qubit operations. – The most general operation on a single qubit can be expressed as a rotation of the operator $\mathbf{S}$ about a vector $\mathbf{v}$

$$e^{i\mathbf{v}\cdot\mathbf{S}} = \mathbb{1}\cos\frac{v}{2} + \frac{2i\mathbf{v}\cdot\mathbf{S}}{v}\sin\frac{v}{2}, \tag{1.2}$$

where $\mathbb{1}$ denotes the unit matrix and v is the length of the vector $\mathbf{v}$. A special case of Eq.(1.2) is the Hadamard operation defined by [7]

$$H \equiv e^{-i\pi/2}e^{i\pi(S^x+S^z)/2} = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}. \tag{1.3}$$

CNOT operation. – By definition the CNOT gate flips the target qubit if the control qubit is in the state $|1\rangle$. If we take the first qubit, that is the

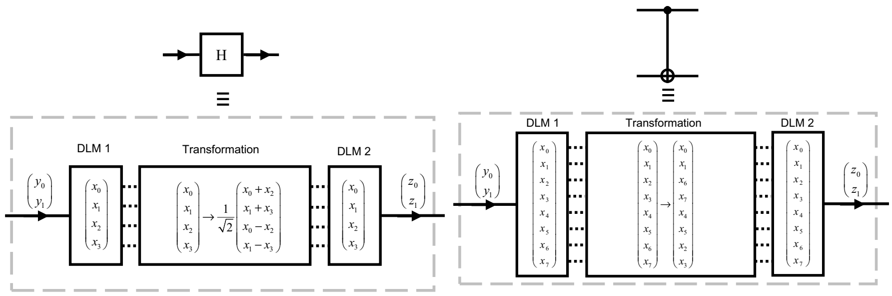

Fig. 1.1. Diagram of DLM networks that perform a deterministic event-based simulation of a Hadamard gate (left) and a CNOT gate (right). The arrows on the solid lines represent the input and output events. Dashed lines indicate the flow of data within the DLM-based processor.

least significant bit in the binary notation of an integer, as the control bit the operation of the CNOT gate on a two qubit state $|\phi\rangle = a_0|00\rangle + a_1|01\rangle + a_2|10\rangle + a_3|11\rangle$ results in $|\phi'\rangle = a_0|00\rangle + a_3|01\rangle + a_2|10\rangle + a_1|11\rangle$. In other words, the probability amplitudes of the states $|01\rangle$ and $|11\rangle$ interchange.

1.4 Hadamard and CNOT gate: DLM-based processors

The diagram of a network that performs a deterministic event-based simulation of a Hadamard gate is shown in Fig. 1.1 (left). The DLM-based Hadamard gate consists of three units. Unit one, called DLM1, "learns" about the occurrence of 0 and 1 events, as described in [6]. The 0 event, corresponding to a qubit in state $|0\rangle$, carries a message $\mathbf{y_0} = (y_0, y_1) = (\cos\psi_0, \sin\psi_0)$ and the 1 event, corresponding to a qubit in state $|1\rangle$, carries a message $\mathbf{y_1} = (y_2, y_3) = (\cos\psi_1, \sin\psi_1)$. Unit two transforms the output of DLM1 by performing an Hadamard operation and feeds this data in the third unit, called DLM2. DLM2 "learns" about the transformed data and responds to the input event by sending out either a 0 or 1 event, as described in [6].

In the quantum system a Hadamard gate transforms the qubit state $|\Phi\rangle$ into $|\Phi'\rangle = [(a_0 + a_1)|0\rangle + (a_0 - a_1)|1\rangle]/\sqrt{2}$. In the corresponding classical system the ys are thus transformed as $(y_0, y_1) \rightarrow (y_0 + y_2, y_1 + y_3)/\sqrt{2}$ and $(y_2, y_3) \rightarrow (y_0 - y_2, y_1 - y_3)/\sqrt{2}$. This is the transformation performed by the second unit in the DLM-based Hadamard processor.

The schematic diagram of the DLM-network that performs the CNOT operation on an event-by-event (particle-by-particle) basis is shown in Fig. 1.1 (right). Conceptually the structure of the network is the same as in the case of a single qubit operation. We now have four instead of two different types of input events, corresponding to the quantum states $|00\rangle$, $|01\rangle$, $|10\rangle$ and $|11\rangle$. Each event carries a message consisting of two real numbers $\mathbf{y} = (\cos\phi_i, \cos\phi_i)$ for

$i = 0, \ldots, 3$, corresponding to the phase of the quantum mechanical probability amplitudes $a_0/|a_0|$, $a_1/|a_1|$, $a_2/|a_2$ and $a_3/|a_3|$, respectively. The internal unit vector of the input and output DLMs have length eight and there are sixteen candidate update rules. The transformation in the transformation unit is simple: all it has to do is swap the two pairs of elements (x_2, x_3) and (x_6, x_7). This corresponds to the interchange of the probability amplitudes of the states $|01\rangle$ and $|11\rangle$ in the quantum system.

Summarizing, by making use of the DLM-networks for the Hadamard and CNOT gate we can build a deterministic event-based universal quantum computer.

1.5 Factoring $N = 15$ using Shor's algorithm

As an example of a quantum algorithm running on the deterministic event-based quantum computer we consider the problem of factoring $N = 15$ on a seven-qubit quantum computer using Shor's algorithm [7,9]. Shor's algorithm is based on the fact that the factors p and q of an integer $N = pq$ can be deduced from the period M of the function $f(j) = a^j modN$ for $j = 0, \ldots, 2^n - 1$ where $N \leq 2^n$. Here $a < N$ is a random number that has no common factors with N. Once M has been determined at least one factor of N can be found by computing the greatest common divisor of N and $a^{M/2} \pm 1$. The quantum network for $N = 15$, $a = 11$ can be found in [10]. The quantum network contains Hadamard and CNOT gates and a network to perform the Fourier transform, containing Hadamard gates and controlled phase shifts. In this particular case the period M of the function $f(j)$ can be determined from the expectation values of the first three qubits, that are the qubits involved in the Fourier transform. According to quantum theory we expect to find $Q_1 = \langle Q_1^z \rangle = 0$, $Q_2 = \langle Q_2^z \rangle = 0.5$ and $Q_3 = \langle Q_3^z \rangle = 0.5$.

We transform the quantum circuit in a DLM or SLM circuit by replacing the quantum gates by DLM or SLM-based gates. The DLM or SLM network to perform a controlled phase shift is constructed by mimicking the procedure for constructing the CNOT gate. We count the number of 1 events in the three output channels of the Fourier transform and divide these numbers by the total number of events analyzed to obtain numerical estimates for the qubits Q_1, Q_2 and Q_3. In Fig. 1.2 we present simulation results for the DLM (left) and SLM (right) implementation of the circuit. After processing a few events the results of quantum theory are reproduced with high accuracy.

1.6 Discussion

We have demonstrated that networks of locally connected processing units with primitive learning capabilities are capable of simulating universal quantum computers, not through solving the Schrödinger equation, but by generating event by event. Since it is known that the time evolution of the wave

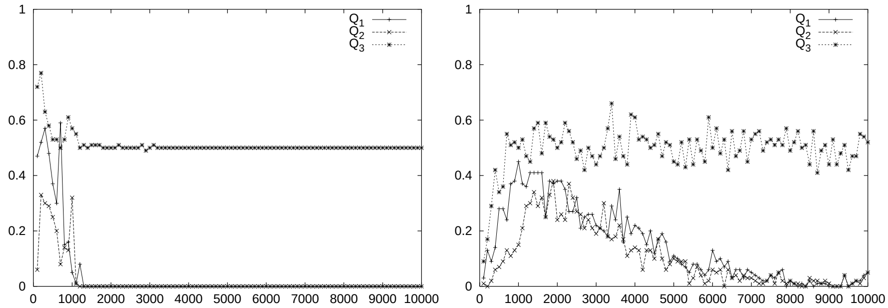

Fig. 1.2. Event-based simulation of Shor's quantum algorithm for factoring $N = 15$, using the value $a = 11$. Each data point represents the average of 100 output events. The parameter that controls the learning process is $\alpha = 0.999$. Left: Results for DLM-network. Right: Results for SLM-network.

function of a quantum system can be simulated on a quantum computer [7], it should thus be possible to simulate real-time quantum dynamics through a deterministic event-based simulation. In conclusion, the work presented in this paper suggests that there exist deterministic, particle-like processes that generate the probability distributions of quantum theory.

References

1. P. Benioff, J. Stat. Phys. **22**, 563 (1980).
2. R.P. Feynman, Int. J. Theor. Phys. **21**, 467 (1982).
3. D.P. DiVincenzo, Phys. Rev. A **51**, 1015 (1995).
4. H. De Raedt, A.H. Hams, K. Michielsen, and K. De Raedt, Comp. Phys. Comm. **132**, 1 (2000).
5. H. De Raedt and K. Michielsen, in Handbook of Computational and Theoretical Nanotechnology, Edited by M. Rieth and W. Schommers, American Scientific Publisher, Los Angeles (2005), in press
6. H. De Raedt, K. De Raedt, K. Michielsen, *Event-by-event simulation of quantum phenomena*, in Computer Simulation Studies in Condensed-Matter Physics XVIII, eds. D.P. Landau et al., Springer Proceedings in Physics, Volume ?? (Springer, Berlin), ?? (200?)
7. M. A. Nielsen, I. L. Chuang, *Quantum computation and quantum information* (Cambridge University Press, Cambridge, New York, 2000).
8. G. Baym, *Lectures on Quantum Mechanics* (W.A. Benjamin, Reading MA, 1974).
9. P.W. Shor, SIAM Review **41**, 303 (1999).
10. U. M. K. Vandersypen, M. Steffen, G. Breyta, C. S. Yannoni, M. H. Sherwood, and I. L. Chuang, Nature **414**, 883 (2001).