*International Journal of Theoretical Physics, Vol. 37, No. 1, 1998*

# Quantum Computation: From the Sequential Approach to Simulated Annealing

**G. Castagnoli,[1] A. Ekert,[2] and C. Macchiavello[2,3,4]**

*Received July 4, 1997*

This is a brief outline of some essential features of quantum computation. We describe sequential quantum data processing and speculate on new modes of quantum computation such as quantum annealing.

It has been known for several years that the theory of quantum computers—i.e., machines that rely on characteristically quantum phenomena to perform computations (Deutsch, 1985)—is substantially different from the classical theory of computation, which is essentially the theory of the universal Turing machine. In particular, quantum computers can perform certain computational tasks, such as factorization (Shor, 1994), using quantum mechanical algorithms (Deutsch and Jozsa, 1992; Bernstein and Vazirani, 1993) which have no classical analogues and can be overwhelmingly more efficient than any known classical algorithm.

The theory of quantum computation has been extensively developed during the last few years and several review papers have covered the recent progress in the field (Ekert and Jozsa, 1996; Lloyd, 1995). Here we provide a brief outline of the essential features of quantum computation, but, in order to avoid repetitions and overlaps with these reviews, we take a slightly unorthodox approach and present the computation as a type of pattern formation and recognition process. This view allows us to stress that it may be worthwhile going beyond a unitary sequential evolution and consider other

[1] Elsag Bailey, a Finmeccanica Company, 16154 Genova, Italy.
[2] Clarendon Laboratory, University of Oxford, Oxford OX1 3PU, U.K.
[3] ISI Foundation, Villa Gualino, 10133 Torino, Italy.
[4] Correspondence should be addressed to Chiara Macchiavello, Clarendon Laboratory, Oxford OX1 3PU, U.K.; e-mail:chiara@mildred.physics.ox.ac.uk.



0020-7748/98/0100-046 3$15.00/0 

modes of quantum computations such as, for example, quantum annealing (Castagnoli and Rasetti, 1993).

Intuitively, a quantum computer is any physical quantum system whose dynamical evolution takes it from one of a set of input states to one of a set of output states. The states are labeled in some canonical way so that the labels provide the input value and the result of the computation. If a quantum computing machine is composed of a quantum register **X** and some auxiliary quantum components, then quantum computation can be viewed as a process which consists in preparing an input state $\rho_{in}$ of the register and evolving it to an output state $\rho_{out}$,

$$\rho_{out} = \sum_i A_i \rho_{in} A_i^\dagger \tag{1}$$

where $\{A_i\}$ is any set of linear operators which form a decomposition of unity

$$\sum_i A_i = 1 \tag{2}$$

and defines a completely positive map $\rho_{in} \to \rho_{out}$. The output state usually contains a pattern encoded in the matrix elements of $\rho_{out}$ which can be revealed by an appropriate measurement performed on the register. The computation or the pattern formation process can be performed in a number of different ways. The most common one is based on a sequential unitary computation with additional registers for storing intermediate results.

Evolution of register **X** described by equation (1) is typically presented as a unitary evolution of **X** together with an auxiliary register **Y.** The two registers **X** and **Y** are composed of respectively $m$ and $n$ qubits, i.e., two-state quantum systems. Any binary string of length $m$, $x \in B^m$ is represented by a vector $|x\rangle$ from a $2^m$-dimensional Hilbert space $\mathcal{H}_X$ associated with register **X.** States corresponding to different strings are orthogonal, $\langle x|x'\rangle = \delta_{xx'}$. Evaluation of any Boolean function $f$: $\mathrm{B}^m \to B^n$ is then determined by an appropriate unitary evolution of the two registers,

$$|x\rangle|0\rangle \overset{U_f}{\to} |x\rangle|f(x)\rangle \tag{3}$$

where $f(x) \in B^n$. The power of quantum computation then comes from our ability to prepare a superposition of all input values $x$ as a single state and by running the computation $U_f$ *only once* we can compute *all* of the $2^m$ values $f(0), \ldots, f(2^m - 1)$,

$$\left(\frac{1}{2^{m/2}} \sum_{x=0}^{2^m-1} |x\rangle\right)|0\rangle \overset{U_f}{\to} \frac{1}{2^{m/2}} \sum_{x=0}^{2^m-1} |x\rangle|f(x)\rangle \tag{4}$$

The snag is that no quantum measurement can extract all of the $2^m$ values $f(0)$, $f(1)$, . . ., $f(2^m - 1)$ from $|f\rangle$. However, there are measurements that provide information about joint properties of all values $f(x)$, such as, for example, periodicity. For if we ignore and trace over the register **Y**, the evolution $U_f$ can be written, following equation (1), as

$$\rho_{\text{in}} = \frac{1}{2^m} \sum_{x,x'} |x\rangle\langle x'| \rightarrow \rho_{\text{out}} = \frac{1}{2^m} \sum_{x,x':f(x)=f(x')} |x\rangle\langle x'| \tag{5}$$

Any periodicity in $f(x)$ will be reflected in a periodic pattern of matrix elements of $\rho_{\text{out}}$. As an example let us consider Simon's algorithm (Simon, 1994). We are given a quantum 'black box' which computes a function $f$: $\mathrm{B}^m \rightarrow B^m$ which is guaranteed to be a 2-to-1 function with periodicity $r$: $f(x) = f(x')$ iff $x' = x \oplus r$ for all $x$, $x' \in B^m$. The problem is to find the periodicity $r$.

We start with the preparation of register $X$ in a equally weighted superposition of all values $x$; this can be achieved by applying transformation

$$T = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \tag{6}$$

to each qubit, initially in state $|0\rangle$. The corresponding transformation on the global state of the register is given by the so-called Hadamard transform

$$H_m|x\rangle = \frac{1}{2^{m/2}} \sum_{y \in B^m} (-1)^{x \cdot y}|y\rangle \tag{7}$$

Then we apply $U_f$, which in our case is defined by

$$U_f: \quad |x\rangle|y\rangle \rightarrow |x\rangle|y \oplus f(x)\rangle \tag{8}$$

Register $X$ alone evolves as

$$\frac{1}{2^m} \sum_{x,x'} |x\rangle\langle x'| \rightarrow \frac{1}{2^{m+1}} \sum_{x} |x\rangle\langle x| + |x\rangle\langle x \oplus r| \tag{9}$$

Finally we measure qubit by qubit an observable specified by eigenvectors $1/\sqrt{2}(|0\rangle \pm |1\rangle)$, i.e., the following global projector in $\mathcal{H}_X$:

$$P_z = \frac{1}{2^m} \sum_{x,x'} (-1)^{z \cdot (x+x')}|x\rangle\langle x'| \tag{10}$$

where $z \in B^m$ and represents the result of the measurement. The state of the system after the measurement, corresponding to result $z$, is given by

$$P_z \rho_{\text{out}} P_z^{\dagger} \tag{11}$$

and has the explicit form

$$\frac{1}{2^{m+1}}[1 + (-1)^{z\cdot r}] \sum_{y,y'} (-1)^{z\cdot(y+y')}|y\rangle\langle y'| \tag{12}$$

As we can see from equation (12), values $z'$ such that $z' \cdot r = 1$ never correspond to outcomes of the measurement, while values corresponding to $z \cdot r = 0$ are obtained with equal probability. The periodicity $r$ can therefore be obtained by repeating the procedure for a sufficient number of times and then solving the system of linear equations $z_i \cdot r = 0$, where $z_i$ are the results of the measurements. In this way the periodicity $r$ can be found with a number of iterations polynomial in $m$, in contrast to the classical case, where the value of $f$ must be computed for each of the $2^m$ input states separately and therefore an exponential number of computations are needed.

Notice that the information about the pattern of the function contained in the density matrix, i.e., the value of the periodicity $r$, has been revealed by a suitable measurement on the $X$ register. Such a measurement can be also performed by first applying the Hadamard transform (7) to the register qubits followed by a simple projector measurement $|z\rangle\langle z|$ in the computational basis. Notice that the application of the Hadamard transform to a basis projector $|z\rangle\langle z|$ leads to projector $P_z$. The last step of the algorithm can then be viewed either as a nonunitary transformation on register **X**, given by the projector measurement $P_z$, or as a sequence of the Hadamard transform (7) followed by the projector measurement $|z\rangle\langle z|$.

We want to point out that apart from the preparation of the initial state of the register **X**, all the steps in Simon's algorithm, including the actual "computation step" given by equation (9), can be realized as a sequence of nonunitary gates acting on register **X.** The action of such gates can be thought of as the result of the interaction of the state of the register with an external system, ignoring what happens to the external system. The external system must be chosen in such a way that its coherence time is larger than the interaction time of the computational step (9). This is required in order to preserve the quantum coherence in the register (it is the quantum coherence which is necessary for the exponential speedup of the quantum algorithm with respect to its classical counterpart).

We notice that the structure of Simon's algorithm is very similar to the more popular Shor algorithm for factoring large integers into primes. Actually, Shor's algorithm also can be viewed as a pattern recognition problem, where the relevant information about the structure of the pattern is contained in the period of a function defined on the group of integers modulo $q$, $Z_q$, and given by

$$g(x) = a^x \bmod N \tag{13}$$

where $q$ is of the order of the square of the number to be factorized $N$, and $a$ is an integer coprime with $N$ and chosen at random. The information about period is extracted in a similar fashion as in the Simon algorithm: the state of register $X$ is first prepared in an equally weighted quantum superposition of all basis states in $Z_q$ and then interacts with a second system $Y$ in order to compute function $g(x)$. In this way the characteristic pattern of $g$ is reflected in the density matrix $\rho_x$ and is revealed in the same way as in Simon's algorithm, replacing the Hadamard transform (7) with the discrete Fourier transform

$$FT|x\rangle = \frac{1}{\sqrt{q}} \sum_{y=0}^{q-1} \exp(2\pi i yx/q)|y\rangle \tag{14}$$

The sequential quantum computation, which we have outlined above, has been the most popular approach in this field for some time. However, we believe that other physical methods of performing computations (after all, any physical process can be viewed as a kind of computation) should be also given a serious consideration. Let us mention briefly one alternative approach known as quantum annealing.

Classical simulated annealing (Kirkpatrick *et al.*, 1983) deals with complex optimization problems, which usually boil down to minimizing certain 'cost functions.' There are many computational problems in which we want to find the minimum of a function in the presence of requirements that push toward different directions and it is difficult to establish which could be the best compromise. Mathematically we could say that we have $N$ Boolean variables $x_i$ and the cost function, which can be of the type

$$H(x) = \sum_{ijk} J_{ijk} \cdot x_i \cdot x_j \cdot x_k \tag{15}$$

with a given set of the $J_{ijk}$ coefficients. It is a hard mathematical problem to find the set of $x_i$ which for a given instance of the $J$'s minimizes $H$. However, if we can find an appropriate physical system, such as, for example, spin glasses, so that the cost function represents the Hamiltonian of the system, we may solve the problem with a help of statistical mechanics. We start with a random, thermal, distribution of $x_i$ and then we cool the system, driving it to its ground state, i.e., the global minimum of $H$, and then we read the corresponding $x_i$.

Consider, for example, a Boolean network which represents a Boolean function $f: B^m \to B^n$ with input nodes $U$ and output nodes $V$ and an assignment of logical values $u$ and $v$ such that $f(u) = v$. Finding $(u, v)$ for a particular value of $u$ can, of course, be realized by simply running a sequential computation on the network with the input nodes set to $u$ and subsequent reading of the output $v$. Another approach is to design the network so that each combination

$(u,v)$ of the Boolean states of a set of nodes belonging to the network elements (gates or wires) corresponds to some energy $E_{u,v}$. Combinations satisfying the relation $f(u) = v$ belong to a degenerate ground energy level, others stand on a discretely higher (by some $\Delta E$) level. Consequently, the overall network ground state corresponds to assignments of the node Boolean variables which satisfy all gates and wires, and are thus network solutions. There may be a number of local energy minima. Therefore the network should be put in interaction with a heat bath so that the system gradually loses its energy and is driven to the ground state without being trapped into these local minima. We emphasize that the simulated annealing approach gives the possibility of considering Boolean networks with closed-loop structures that are not implementable in a simple way in a time-sequential model. By introducing wires which connect the output to the input, leading to a situation where both the input and the output are in a functional dependence on each other, we can give the network additional complexity.

In the quantum annealing computation proposed by Castagnoli and Rasetti (1993), combinations $(x, y)$ with energies $E_{x,y}$ become eigenstates and eigenvalues of the gate Hamiltonian. In the ground state, the gate can dwell in any superposition of the eigenstates (i.e., tensor products of the gate qubit eigenstates) which satisfy it. For a suitable temperature $T$ of the heat bath, such that $0 < kT << \Delta E$, the gate will relax onto ground following (asymptotically) the exponential law $p = 1 - e^{-\rho t}$, with $\rho > 0$, where $p$ is the probability of finding the gate in the ground state by time $t$. For example, the logical operation NOT can be implemented by driving an appropriate two-node gate to its ground state which is in a superposition of pairs $(x,y)$ which satisfy $y$ = NOT $x$, i.e., in a superposition of $|01\rangle$ and $|10\rangle$:

External constraints which determine the energy minima can be imposed by an appropriate tuning of the intergate and network–environment interactions or by some symmetry properties due to quantum statistics (Castagnoli, 1998).

Quantum annealing requires a coupling with the environment; in this case dissipation and decoherence may be employed to perform useful computations. Let us mention in passing that a sequential quantum computation is only possible when one can minimize the unwelcome effects of dissipation. As it is not clear at present which technology, if any, will support quantum computation in the future, it seems appropriate to keep the repertoire of potential approaches to quantum computation as broad as possible and certainly include quantum annealing.

In summary, we have briefly outlined some basic ideas behind the theory of quantum computation, including quantum annealing as an interesting alternative for quantum architecture. Recently both experimental and theoretical research in quantum computation is accelerating worldwide. New technol-

ogies for realizing quantum computers are being proposed, and new types of quantum computation with various advantages over classical computation are continually being discovered and analyzed and we believe some of them will bear technological fruit. From a fundamental standpoint, however, it does not matter how useful quantum computation turns out to be, nor does it matter whether we build the first quantum computer tomorrow, next year, or centuries from now. The quantum theory of computation must in any case be an integral part of the world view of anyone who seeks a fundamental understanding of the quantum theory and the processing of information.

## ACKNOWLEDGMENTS

This work has been supported in part by Elsag Bailey, a Finmeccanica Company, and by the European TMR Research Network ERB 4061PL95-1412.

## REFERENCES

Bernstein, E., and Vazirani, U. (1993). In *Proceedings 25th ACM Symposium on Theory of Computation* , p. 11.

Castagnoli, G. (1998). *International Journal of Theoretical Physics*, this issue.

Castagnoli, G., and Rasetti, M. (1993). *International Journal of Theoretical Physics,* **32**, 2335.

Deutsch, D. (1985). *Proceedings of the Royal Society of London A,* **400**, 97.

Deutsch, D. (1989). *Proceedings of the Royal Society of London A,* **425**, 7.

Deutsch, D., and Jozsa, R. (1992). *Proceedings of the Royal Society of London A,* **439**, 553.

Ekert, A., and Jozsa, R. (1996). *Reviews of Modern Physics,* **68**, 733.

Kirkpatrick, S., Gelatt, C. D., Jr., and Vecchi, M. P. (1983). *Science,* **220**, 671.

Lloyd, S. (1995). *Scientific American,* **273**, 44.

Shor, P. W. (1994). In *Proceedings 35th Annual Symposium on the Foundations of Computer Science*, IEEE Computer Society, Los Alamitos, California, p. 124.

Simon, D. S. (1994). In *Proceedings of the 35th Annual Symposium on the Foundations of Computer Science*, S. Goldwasser, ed., IEEE Computer Society Press, Los Alamitos, California, p. 116.