LA-UR-92- 92-3339

TITLE: "Review of Quantum Computation"

LA-UR--92-3339

DE93 003735

AUTHOR(S): Seth Lloyd, T-13 Complex Systems Group

SUBMITTED TO: To appear in the Proceedings of the International Symposium on Quantum Physics in the Universe, held at Waseda University, Tokyo, Japan, August 19-23rd, 1992.

Los Alamos

Los Alamos National Laboratory
Los Alamos, New Mexico 87545

FORM NO. 836 R4
ST. NO. 2629 5/91

REVIEW OF QUANTUM COMPUTATION

Seth Lloyd

Complex Systems Group (T-13)
and *Center for Nonlinear Studies*
Los Alamos National Laboratory
Los Alamos, New Mexico 87545

COMPUTERS AND PHYSICS

Digital computers are machines that can be programmed to perform logical and arithmetical operations. Contemporary digital computers are 'universal,' in the sense that a program that runs on one computer can, if properly compiled, run on any other computer that has access to enough memory space and time. Any one universal computer can simulate the operation of any other; and the set of tasks that any such machine can perform is common to all universal machines.

Many classical systems, ranging from cellular automata to hard-sphere gases, have been shown to be capable of universal computation (Wolfram 1985; Toffoli 1982, 1977; Minsky 1967; Omohundro 1984; Moore 1990, 1991). Since Bennett's discovery that computation can be carried out in a non-dissipative fashion, a number of Hamiltonian quantum-mechanical systems have been proposed whose time-evolutions over discrete intervals are equivalent to those of specific universal computers (Bennett 1973,1982; Benioff 1980, 1982, 1986; Feynman 1982,1985,1986; Deutsch 1985, 1989).

The first quantum-mechanical treatment of computers was given by Benioff, who exhibited a Hamiltonian system with a basis whose members corresponded to the logical states of a Turing machine, and whose unitary [illegible]n transformed those basis states at integer times into the states corresponding [illegible] logical successors (Benioff 1980, 1982). In order to make the Hamiltonian local, in the sense that its structure depended only on the part of the computation being performed at that time, Benioff found it

necessary to make the the Hamiltonian time–dependent. Feynman discovered a way to make the computational Hamiltonian both local and time–independent by incorporating the direction of computation in the initial condition. In Feynman's quantum computer, the program is a carefully prepared wave packet that propagates through different computational states; at any time, the computation is in a superposition of a number of computational states (Feynman 1985, 1986). Deutsch presented a quantum computer that exploits the possibility of existing in a superposition of computational states to perform tasks that a classical computer cannot, such as generating purely random numbers, and carrying out superpositions of computations as a method of parallel processing (Deutsch 1985). Further examinations of the properties of quantum computers, including their behavior in the presence of noise, were carried out by Zurek and Peres (Zurek 1984; Peres 1985).

In this paper, we show that such computers, by virtue of their common function, possess a common form for their quantum dynamics.

TURING MACHINES AND QUANTUM MECHANICS

The original model for a digital computer is the Turing machine (Turing 1936). Turing's intent was to model in rudimentary fashion the way in which a mathematician does math, by writing equations and symbols on sheets of paper, and by examining and altering previously written equations and symbols. In its simplest form, a Turing machine consists of a 'head,' that can be in one of a finite number of states, and that can read and write on a 'tape,' that is divided up into squares each one of which contains one of a finite number of symbols. The head is initially located at some square of the tape, and at the next time step, as a function of its internal state and the symbol on the square that it is reading, it writes a new symbol on that square, changes its internal state, and either stays where it is or moves one square to the left or right. Turing showed that, despite their simplicity, such machines could perform a wide variety of mathematical and logical tasks. In addition, he exhibited Turing machines that were universal in the sense that they could be programmed to simulate the action of any other Turing machine. Such universal machines need have no more than a small number of tape symbols and head states.

At any step in a computation, the state of a Turing machine can be characterized by a word of finite length in the tape alphabet specifying the contents of the nonblank squares on the tape, an integer giving the position of the head on the tape, and a finite,

discrete label identifying the internal state of the head. The computational states of a universal Turing machine or digital computer make up a discrete, countably infinite set. The dynamics of the machine are deterministic on that set, and are discrete in time.

If one labels the set of integers that determine the computational state of a universal digital computer at a particular time as b, then the computer's state at the next time step is a function of b: $b' = \Upsilon(b)$. The quantum computers discussed by Benioff have computational states that are pure states, $|b\rangle$, corresponding to the computational states b. These quantum computers each possess a unitary time evolution U over discrete time intervals Δt, such that $U|b\rangle = e^{i\phi(b)}|\Upsilon(b)\rangle$, where $e^{i\phi(b)}$ is an arbitrary phase, and Υ specifies the computational dynamics of some universal computer.

Deutsch's computer is capable in addition of performing a rotation of the spin-like variable that corresponds to a single bit by an angle that is an irrational fraction of 2π. This single additional operation allows the computer to perform tasks beyond those of which a classical digital computer is capable, including constructing random numbers from measurements on quantum-mechanical superpositions, and constructing arbitrary unitary transformations on finite-dimensional Hilbert spaces. The above description applies to those states of Deutsch's computer that behave like conventional computational states.

To be realizable by the sort of pure state quantum computer described above, the computational dynamics Υ must be a one-to-one function, since for a unitary operator U, $U|b\rangle = U|b'\rangle$ implies $|b\rangle = |b'\rangle$: that is, each computational state b must have not only a unique image, $\Upsilon(b)$, but a unique preimage, $\Upsilon^{-1}(b)$. Bennett has shown that there exist universal computers with a one-to-one computational dynamics. Although most logical operations, such as addition, or taking the AND of two bits, have outputs that are not one-to-one functions of their inputs, these operations may be embedded in one-to-one functions, and extra registers may be supplied to record the information that is not needed for the logical operation of the computer, but that must be recorded in order to make that logical operation one-to-one. As noted by Landauer, a computer whose computational dynamics are many-to-one, such as current digital computers, must be dissipative (Landauer 1961).

2. DIAGONAL REPRESENTATION OF COMPUTATIONAL EVOLUTION OPERATORS

Our immediate goal is to characterize the form of the unitary evolution operator

U for a universal pure-state quantum computer. Since a universal computer has a countably infinite number of computational states, the Hilbert space spanned by the $\{|b\rangle\}$ is infinite dimensional. There is no guarantee, then, that U is diagonalizable. Our first result is to show that U is in fact diagonalizable, and to exhibit the form of its diagonal representation. Any system whose discrete time unitary evolution operator has the same diagonal representation as that for some universal quantum computer, itself has a basis with respect to which its behavior is that of a universal computer.

If a computer with a one-to-one dynamics Υ starts computing in a particular state, then it either returns to that state, or never returns. Similarly, if one applies the operator U repeatedly to a computational state $|b\rangle$, two things may happen. First, one may find that $U^m|b\rangle = e^{i\phi}|b\rangle$, for some integer m and phase ϕ. Second, one may find that $|b\rangle$ never returns to itself for any number of iterations of U. The span of the set $\{U^n|b\rangle\}$ is an invariant subspace $\mathcal{H}_b$ under U: $U\mathcal{H}_b = \mathcal{H}_b$. In the first case above, this invariant subspace has dimension m; in the second, it is infinite dimensional. To diagonalize U, we need only diagonalize it on the set of invariant subspaces generated by all such computational states $|b\rangle$.

Consider first the case for which $U^m|b\rangle = e^{i\phi}|b\rangle$. We have $U^m|\psi\rangle = e^{i\phi}|\psi\rangle$ for any $|\psi\rangle \in \mathcal{H}_b$. The eigenvalues of U on this invariant subspace are all m-th roots of $e^{i\phi}$; there are m of these roots, the k-th of which is $\alpha_k = e^{i(2\pi k+\phi)/m}$. The eigenvector corresponding to the eigenvalue α_k is $|\alpha_k\rangle = 1/\sqrt{m}\sum_{j=0}^{m-1}\alpha_k^{-j}U^j|b\rangle$. The decomposition of the computational states within the invariant subspace $\mathcal{H}_b$ in terms of eigenstates of U is given by $U^j|b\rangle = 1/\sqrt{m}\sum_{k=0}^{m-1}\alpha_k^j|\alpha_k\rangle$. The case, $\phi = 0$, is well known (Schwinger 1970; Benioff 1980, 1982,1986; Peres 1985) In this finite dimensional invariant subspace, the m eigenvalues are distributed at equal intervals around the unit circle in the complex plane, and the computational eigenstates are uniform superpositions of the eigenvectors.

Now consider the case in which $\mathcal{H}_b$ is infinite dimensional, spanned by the set $\{U^m|b\rangle\}$, where m can be any integer. By analogue with the finite dimensional case, we can define eigenstates $|\alpha\rangle = \sum_{m=-\infty}^{\infty} e^{-im\alpha}U^m|b\rangle$. These states are eigenvectors of U with eigenvalue $e^{i\alpha}$, but they are not normalizable within the Hilbert space $\mathcal{H}_b$. This situation is common in quantum mechanics: for example, in the case of the particle in a box, the position eigenstates lie outside of the Hilbert space spanned by the energy eigenstates, and require delta function normalization. The non normalizability of the eigenstates of U is not a hindrance in decomposing the computational states in terms of these eigenstates: we have $U^m|b\rangle = 1/\sqrt{2\pi}\int_0^{2\pi} e^{im\alpha}|\alpha\rangle$, and the states $|\alpha\rangle$ are normalized so that $\langle\alpha|\alpha'\rangle = \delta(\alpha - \alpha')$.

The eigenstates of U are discrete Fourier transforms of the computational states, $U^m|b\rangle$, and the computational states are continuous Fourier transforms of the eigenstates, the analogue of plane waves for the particle in the box.

The unitary operator U that induces the computational dynamics is diagonalizable, and has both a discrete spectrum that corresponds to computations that cycle after a finite number of steps, and a continuous spectrum corresponding to computations that never repeat. A universal computer must be capable of computations that never repeat, (for example, counting from 1 to ∞), but need not necessarily be capable of engaging in computations that cycle (for example, a universal computer could possess a clock register that continues to count upward after computation halts). For quantum universal computers, the discrete part of the spectrum is optional, but the continuous part is mandatory. In addition, each universal computer is capable of a countable infinity of non-overlapping, non-cyclic computations; the continuous part of the spectrum therefore has a countable infinity of orthogonal sectors each with eigenvalues $e^{i\alpha} : \alpha \in [0, 2\pi]$.

The above decomposition applies not only to quantum computers such as those of Benioff, that perform the same logical tasks as classical computers, but also to quantum computers of the sort proposed by Deutsch, that are able to rotate the spin-like variables corresponding to single bits by an irrational amount, and that can thereby perform actions, such as true random number generation, that are forbidden to classical computers. That is, for such computers there exists some subset of computational states, $\{|c\rangle\}$, such that $U|c\rangle = \gamma_1|b_1\rangle + \gamma_2|b_2\rangle$, where $|b_1\rangle$ and $|b_2\rangle$ are computational states that differ from eachother only by the value of a single bit, $|\gamma_1|^2 + |\gamma_2|^2 = 1$, and $sin^{-1}(|\gamma_1|)/2\pi$ is irrational. In the case of a computer in which all cycles are infinite, so that $\langle c|U^m|c\rangle = 0$ for all m, $|c\rangle$, the decomposition given above for the states in an infinite cycle $U^m|c\rangle$ still holds. $|b_1\rangle$ and $|b_2\rangle$ can always be selected to be uniform decompositions of eigenvectors, subject to the constraint that they add up to $U|c\rangle$ as above. Deutsch's computers can evolve into superpositions of binarily labelled states, but the binary labels of the states of Deutsch's computer (corresponding, say, to configurations of a set of spins along predetermined axes) no longer correspond to the integer labels of the countably infinite set of computational states. The diagonal form of the time evolution operator remains the same as for a pure state quantum computer that realizes conventional one-to-one digital computations.

Feynman's computer has a unitary time evolution of a form slightly different from that just given. Feynman was interested in creating a computer with a time independent

Hamiltonian, whose interactions were local. For the computers described above, the unitary evolution is time-independent, and the discrete-time interactions are local, only involving a few bits at each computational step. The treatment given above says nothing, however, about what happens in between the discrete instants in time at which the computer is in computational states. In fact, as pointed out by Benioff, at in-between times such a computer can be in a superposition of all logical states in a computational cycle. Feynman's solution to this problem was to define a Hamiltonian, $H = U + U^\dagger$, corresponding to a unitary time evolution operator over an interval t of $e^{iHt} = 1 + iHt - H^2t^2/2 - \ldots$. Such a computer, prepared initially in a computational state of U, will evolve at time t into a superposition of all computational states in the same cycle. As noted above, if the initial state of the system is prepared in the proper wave packet, the action of e^{iHt} on that state, corresponding to a superposition of actions of U and $U^\dagger$ operating on the state different number of times, tends to propagate the computation forward. To perform a computation on such a computer, one starts the computer out in a wave packet centered at the beginning of the computation, then waits a suitable amount of time, and makes a measurement on the Halt bit of the computer alone. If the result of this measurement is that the Halt bit is one, i.e., the computer has halted, then one makes a measurement on the output register to extract the result of the computation. Since U is local and time-independent, so is H, and the eigenvectors of U and H are the same. For Feynman's computer, the spectrum of H can be derived by first finding the spectrum of the corresponding U, as above, and then taking its projection onto the real axis.

3. CONCLUSION

We have reviewed the quantum computers proposed by Benioff, Deutsch, and Feynman, and shown that under the proper circumstances, their time evolution operators all possess the same diagonal form.

REFERENCES

Benioff, P. (1980) *J. Stat. Phys.* **22** 563–591; (1982) *Phys. Rev. Lett.* **48**, 1581–1585; (1982) *J. Stat. Phys.* **20**, 515–546); (1986) *Ann. N.Y. Acad. Sci.* **480**, 475–486.

Bennett, C.H. (1973) *IBM J. Res. Develop.* **17**, 525–532; (1982) *Int. J. Theor. Phys.* **21**, 905–940.

Deutsch, D. (1985) *Proc. Roy. Soc. Lond.* **A 400**, 97–117; (1989) *Proc. Roy. Soc. Lond.* **A 425**, 73–90.

Feynman, R.P. (1985) *Opt. News* **11**, 11-20; (1986) *Found. Phys.* **16**, 507-531; (1982) *Int. J. Theor. Phys.* **21**, 467-488.

Landauer, R. (1961) *IBM J. Res. Dev.* **5**, 183-191.

Margolus, N. (1986) *Ann. N.Y. Acad. Sci.* **480**, 487-497.

Minsky, M. (1967) *Computation: Finite and Infinite Machines*, Prentice Hall, Englewood Cliffs.

Moore, C. (1990) *Phys. Rev. Lett.* **64**, 2354-2357; (1991) *Nonlinearity* **4**, 199-230.

Omohundro, S. (1984) *Physica* **10D**, 128-134.

Peres, A. (1985) *Phys. Rev.* **A 32**, 3266-3276.

Schwinger, J. (1970) *Quantum Kinematics and Dynamics*, W.A. Benjamin, New York.

Toffoli, T. (1977) *J. Comp. Syst. Sci* **15**, 213-231; Fredkin, E., Toffoli, T. (1982) *Int. J. Theor. Phys.* **21**, 219-253.

Turing, A.M. (1936-1937) *Proc. Lond. Math. Soc.*, series 2, **42**, 230-265.

Wolfram, S. (1985) *Phys. Rev. Lett.* **54**, 735-738.

Zurek, W.H. (1984) *Phys. Rev. Lett.* **53**, 391-394.