Can Quantum Computers Have Simple Hamiltonians?

Michael Biafore
Laboratory for Computer Science
Massachusetts Institute of Technology
Cambridge, MA 01239
biafore@mit.edu

Abstract

Recently, Shor has shown that quantum computers, computers which can operate simultaneously on a quantum superposition of inputs, permit efficient (i.e. polynomial-time) solutions of problems for which no efficient classical-mechanical solution is known. This has led to renewed interest in the question of whether or not quantum computers can be physically realized. One kind of quantum computer, quantum cellular automata, can be described by relatively simple Hamiltonians that resemble the Hamiltonians of spin systems. In this paper, we report a quantum cellular automaton which, though not itself computation-universal, forms an essential part of any quantum cellular automaton which is synchronized using Feynman's technique. This quantum cellular automaton has as its Hamiltonian the one-dimensional XY Hamiltonian, which is exactly solvable. Furthermore, there is experimental evidence from low-temperature measurements of the heat capacity and electric susceptibility that the Hamiltonian of the quantum cellular automaton is realized in nature by the rare-earth compound praseodymium ethyl sulfate near 1K.

1 Introduction

This article describes what we believe is the first quantum cellular automaton whose Hamiltonian is simple enough to be identified with a real physical system[1]. This particular cellular automaton is not computation-universal, and therefore not as powerful as a general-purpose computer. But it can be used to implement an essential and seemingly difficult to realize part of certain kinds of quantum computers. Since the quantum mechanisms needed to realize the remaining components of the quantum computer have a similar character, the existence of quantum cellular automata in nature suggests that quantum computers may be easier to discover—or engineer—than one might intuitively expect.

Quantum computers have recently attracted renewed interest in large part because Shor[15] has succeeded in showing that quantum computers admit efficient (i.e. polynomial-time) algorithms for certain problems, notably the Factoring Problem that forms the basis of the RSA cryptographic method, for which no efficient algorithm is known to exist on ordinary computers.

Benioff[1, 2] and Deutsch[6] were the first to initiate a line of investigation that begins with a particular model of computation, the Turing machine, and seeks to construct a quantum Hamiltonian such that the operation of the Turing machine corresponds to the evolution of a quantum spin system.

The search for quantum cellular automata in nature can be viewed as a continuation of this approach to quantum computation.

The factoring algorithm described by Shor[15] is designed to run on a quantum Turing machine[6]. A Turing machine consists of a one-dimensional tape composed of cells and a moving "head". Each cell has a few internal states which represents a symbol drawn from a finite alphabet. Most often, the alphabet is the set of binary values $\{0, 1\}$. The head possesses internal states of its own, which it changes in response to the state of the cell it is currently scanning. In addition, the head is free to move one cell to the left or right at each time step. This classical-mechanical mechanism was generalized by Benioff and Deutsch to obtain a quantum Turing machine.

Quantum Turing machines. In a quantum Turing machine, the binary state of each cell is replaced by a quantum-mechanical spin-$\frac{1}{2}$ variable in which the state $|\uparrow\rangle$ represents binary value 1, and the state $|\downarrow\rangle$

[1]Adapted from Ch. 5, *A computational role for the one-dimensional XY model*, Ph.d. thesis, MIT Physics (1993)

represents binary value 0. Consequently, the quantum Turing machine may be in a superposition of many states of the corresponding classical Turing machine. The operation of the Turing machine—the movement of the head and the evolution of the states of the cells and head—is described by a time-dependent Hamiltonian $\widehat{H}(t)$ via the corresponding unitary Schrödinger evolution operator. It is assumed both that the cells can be initialized prior to the computation and that their values can be determined when the computation is finished by applying whatever physical operations are used, respectively, to prepare the quantum states $\{|\uparrow\rangle, |\downarrow\rangle\}$ in a definite eigenstate and to measure the corresponding quantum-mechanical observable.

Since the model of quantum computation presented here is a Hamiltonian model, it is necessarily hampered by a number of impediments if the actual Hamiltonian differs from the desired Hamiltonian. Landauer has long recognized the serious handicap that accumulation of errors poses to truly reversible computation[10]. In addition, to obtain the unique benefits of quantum computation, a coherent superposition of states of the computer must be maintained. As recently shown by Unruh[16], environmentally-induced quantum decoherence forces an almost complete isolation of the quantum computer from its environment at temperature T if the computation lasts for a time $t > \frac{\hbar}{k_B T}$.

In the Benioff approach to quantum computation, the time-dependence of the Hamiltonian is one of the physically unrealistic features one strives to eliminate in order to arrive at models of quantum computation that more closely resemble real systems. Feynman[7] discovered a general technique for turning time-dependent models of quantum computers into time-independent models. Feynman's technique eliminates the time-dependence of the Hamiltonian by building the quantum computers on top of another quantum system that serves as a "synchronization backbone". This synchronization subsystem has the difficult task of ensuring that—even in the absence of an explicit time dependence—all the computations going on in various parts of the quantum computer occur at the proper point in time. Without the synchronization subsystem, the quantum computer would quickly make too many errors to be of any use. Feynman applied this technique to Benioff's quantum Turing machine[7]. As shown by Margolus[12, 13], this technique can also be extended to quantum cellular automata in two or more dimensions. (Feynman's quantum Turing machine can be regarded as a one-dimensional quantum cellular automaton.)

Quantum cellular automata provide a natural starting point for investigating quantum computation because, unlike other models of computation, they share some fundamental characteristics with real physical systems. In particular, they evolve according to a single rule applied everywhere, just as a homogeneous physical system evolves according to a single dynamical law.

In this article, we show that for a certain one-dimensional cellular automaton, a cellular automaton which is identical to the one considered by Feynman[7], and closely related to the one considered by Margolus[13]—the Hamiltonian for the Feynman synchronization backbone is identical to $\widehat{H}_{1\mathrm{D}XY}$, the Hamiltonian of the one-dimensional XY model. As I have discussed elsewhere[4], experimental data[8, 9] strongly suggest that this Hamiltonian accurately describes at least one system—praseodymium ethyl sulfate near 1K—that actually occurs in nature.

In this article, I focus on a second remarkable aspect of the quantum cellular automaton described by $\widehat{H}_{1\mathrm{D}XY}$. Because the one-dimensional XY model is exactly solvable, it is possible to predict such things as the computation-rate of a quantum cellular automaton synchronized by the Feynman technique.

2 The synchronization subsystem

The method used here is to apply the Feynman synchronization technique to a one-dimensional lattice-gas cellular automaton. In contrast to ordinary cellular automata, which require the state of each cell to depend on its own former state and that of at least two neighboring cells—a total of three cells, in the lattice-gas cellular automaton considered here, neighboring cells interact in pairs. Restricting ourselves to this class of cellular automata will eventually allow us to map them onto a quantum-mechanical system involving two-body interactions.

Despite its simplicity, this class of "two-body" cellular automata is known to be capable of computing anything a Turing machine can compute[5].

There is, however, one rule that must be strictly enforced. After two cells A and B have interacted, thereby changing their internal states, as in Figure 1, both the event where cell A interacts with its left neighbor and the event where cell B interacts with its right neighbor must occur before cells A and B are permitted to interact again.

It makes no difference in which order the two events occur, so long as they both occur before cells A and

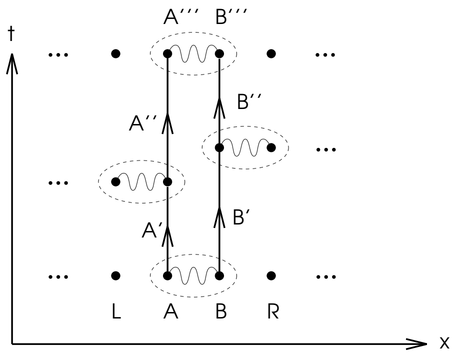

Figure 1: Either the two-body interaction between B' and R, or the two-body interaction between L and A' may occur first (as shown). But both must occur before the cells originally in states A and B are permitted to interact again.

B are permitted to affect one another's state again. If they are not prevented from interacting until both have interacted with their other neighbor, any computation the cellular automaton was performing is irremediably corrupted. Preventing these undesirable interactions is the role of the synchronization subsystem.

The key problem in constructing a quantum cellular automaton, therefore, is to find a quantum mechanical system that can enforce this constraint.

It is not hard to imagine a classical mechanical system that enforces the required synchronization of cell updates. One synchronization subsystem that can easily seen to prevent errors requires only two states at each site of the synchronization backbone. When the states of the synchronization subsystem for two neighboring cells are in the configuration (01), the computational states of those cells are permitted to evolve. Concurrently—and as part of an indivisible transaction—the state of the synchronization subsystem undergoes the transformation $(01) \rightarrow (10)$. For any other configuration of states of the synchronization system, no evolution of either the computational or synchronization subsystems occurs.

When this synchronization backbone is translated into a quantum spin system, the resulting Hamiltonian is

$$\widehat{H}_{\rm sync} = \sum_{n=0}^{M-1} \sigma_n^+ \sigma_{n+1}^- + \sigma_n^- \sigma_{n+1}^+. \tag{1}$$

We recognize the effective Hamiltonian $\widehat{H}_{\rm sync}$ as an extremely anisotropic Heisenberg antiferromagnet

$$\widehat{H} = J_\perp \sum_{n=0}^{M-1} (\sigma_n^x \sigma_{n+1}^x + \sigma_n^y \sigma_{n+1}^y) + J_z \sum_{n=0}^{M-1} \sigma_n^z \sigma_{n+1}^z \tag{2}$$

with exchange couplings $J_\perp = 1$ and $J_z = 0$. This model is sometimes called the one-dimensional XY model[11].

The computation-rate operator. From the identification $\widehat{H} \equiv \widehat{H}_{\rm sync} = \widehat{H}_{1DXY}$, we know the energy eigenstates and eigenvalues from the results of Lieb, Shultz and Mattis[14]; however, in this paper we are principally interested in the spectrum of a different operator, an operator that characterizes the computation rate of the quantum cellular automaton.

To address this issue, we must explicitly define an operator $\widehat{\Gamma}$ that captures some reasonable notion of the net rate at which the parallel computation of the quantum cellular automaton is moving forward. Then we need to show that the computation-rate operator $\widehat{\Gamma}$ has an eigenstate $|\gamma\rangle$ with a positive eigenvalue. Finally, we need to show that as the system evolves the rate of computational progress does not decrease to zero. To do this, it suffices to show that $|\gamma\rangle$ is also an eigenvector of $\widehat{H}$, since the eigenvalue $\gamma > 0$ is then a constant of the motion.

To help motivate the definition of $\widehat{\Gamma}$, we define an auxiliary operator $\widehat{N}_c$. On the basis states—states in which each cell has been updated a definite number of times—$\widehat{N}_c$ returns the sum over all cells of the number of forward minus the number of backward steps that have taken place relative to some arbitrarily fixed state. We can think of the computational-rate operator as the time-derivative of this operator

$$\widehat{\Gamma} \equiv \frac{d}{dt}\widehat{N}_c = \frac{1}{i\hbar}[\widehat{H}, \widehat{N}_c]. \tag{3}$$

So $\widehat{\Gamma}$ characterizes the net rate at which the computation runs in the forward direction.

Evaluating the commutator, we obtain

$$\widehat{\Gamma}(M) = \frac{1}{i} \sum_{n=0}^{M-1} \sigma_n^+ \sigma_{n+1}^- - \sigma_n^- \sigma_{n+1}^+. \tag{4}$$

where M is the number of spins in a one-dimensional cellular automaton with periodic boundary conditions.

This computation-rate operator can be diagonalized by first tranforming to Jordan-Wigner operators and the Fourier transforming. The resulting single-particle (i.e., single Jordan-Wigner fermion) spectra of the Hamiltonian and computation-rate operator $\widehat{\Gamma}$

are shown in Fig. 2. Since the Feynman-Margolus synchronization scheme described above requires each quantum state of the cellular automaton to have equal numbers of spin-up and spin-down sites, them number of Jordan-Wigner fermions is required to be $M/2$ and the ground state consists of Jordan-Wigner fermions occupying the shaded states of Fig. 2.

3 Relation between energy and computation rate

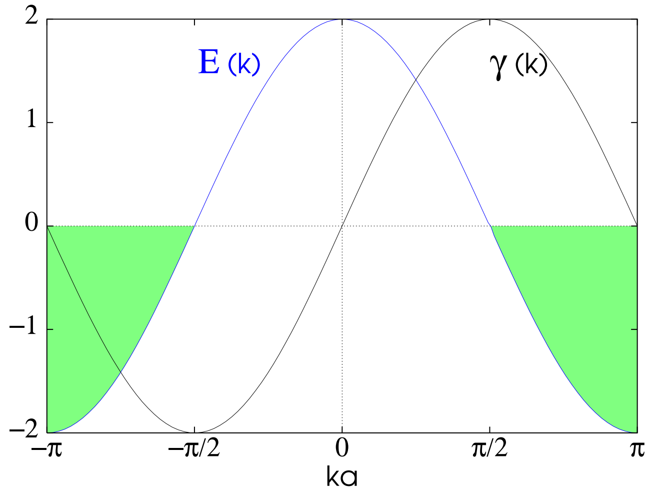

Figure 2: Single-particle spectra of $\widehat{H}(M)$ and computation-rate operator $\widehat{\Gamma}(M)$ in the limit $M \to \infty$.

When the system is in its ground state, the computation rate γ vanishes because the energy spectrum is symmetric and the computation-rate spectrum antisymmetric about $k = 0$. In fact, we can see from Fig. 3 that the computation rate γ vanishes whenever the system is in thermal equilibrium at any finite T. This follows from the fact that the computation-rate spectrum $\gamma(k)$ is symmetric about the Fermi surface at $k = \pm\pi/2$, so that the finite-temperature Fermi distribution function (bold line) cannot change $\gamma(M)$ from its $T = 0$ value, which is $\gamma(M) = 0$.

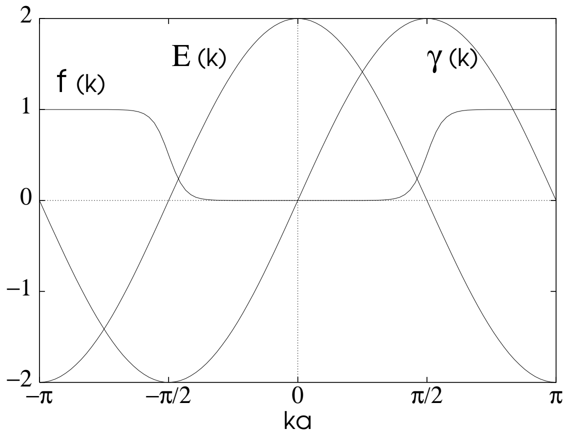

Figure 3: Temperature dependence of the computation rate. The Fermi distribution function $f(k)$ (solid bold line) at the finite temperature $\beta = 10$ illustrates why $\langle\gamma\rangle = 0$ at all finite temperatures.

Although it is not possible to compute faster than diffusively when the system is in thermal equilibrium[3], the quantum cellular automaton can attain a substantial fraction of its maximum possible computation rate with surprisingly little excitation energy.

To show this, we must first identify the state with the largest computation rate among those of a given energy. Alternatively, we can achieve the same end by identifying the state with minimum energy among those with a given computation rate. States of many different total energies can have the same parallel computation-rate. For a given $\gamma_0 \leq \gamma_{\max} = 2M/\pi$, we would like to find the minimum energy at which γ_0 can be achieved. From the spectra of $\widehat{H}$ and $\widehat{\Gamma}$ (Fig. 2), we see that to obtain a given γ_0 at a minimum energy cost we should shift the ground state configuration of the Fermi sea to the left until γ has increased from its ground state value of zero to γ_0. This follows from the fact that the spectra of $\widehat{H}$ and $\widehat{\Gamma}$ are shifted by $\pi/2$ along the k-axis, and therefore the left shift adds the k states with the largest available eigenvalues of $\widehat{\Gamma}$ and smallest positive energy (near $k = \pi/2$), while simultaneously vacating the states of least negative energy and most negative computation-rate (near $k = -\pi/2$).

For example, assume $M \pmod 4 \neq 0$, so that

$e^{ikMa} = 1$, and therefore $k_n a = 2\pi n/M$. The single-particle eigenvalues in this case are

$$\begin{array}{ll} 2\sin 2\frac{\pi}{M}n & (\text{for } \widehat{\Gamma}), \\ 2\cos 2\frac{\pi}{M}n. & (\text{for } \widehat{H}) \ . \end{array} \tag{5}$$

When M is large, the full $M/2$-particle eigenvalues of $\widehat{\Gamma}$ and $\widehat{H}$ can be approximated by integrals over occupied single-particle states. If $\delta(ka) = \epsilon$ denotes the amount by which the Fermi sea has shifted toward the left, then

$$\gamma(\epsilon) = 2\int_0^{\epsilon} 2\cos ka \frac{M}{2\pi} d(ka) = \frac{2M}{\pi}\sin\epsilon \tag{6}$$

and

$$E(\epsilon) = 2\int_0^{\epsilon} 2\sin ka \frac{M}{2\pi} d(ka) = \frac{4M}{\pi}\sin^2\frac{\epsilon}{2}. \tag{7}$$

From these, we can express the maximum computation rate obtainable at energy E,

$$\gamma(E) = \sqrt{E(\frac{4M}{\pi} - E)}. \tag{8}$$

Figure 4 shows the maximum computation rate per site that can be attained for a given energy density (i.e., energy per site). The parallel computation rate is normalized to the maximum attainable at any energy. The maximum parallel computation rate $\gamma_{\max}(M)$ occurs when the energy density is equal to half the maximum energy density the system can possess. Note that near the ground-state, a small excitation energy can produce a relatively large increase in the computation rate. An excitation energy $\Delta E = 0.1 \times E_{\max}$ one can yield a parallel computation rate $\gamma = 0.6 \times \gamma_{\max}$. In contrast, as the maximum computation rate is approached, relatively large increments to the energy of the system are unable to produce much additional computation rate. Beyond an excitation energy equal to half the maximum energy content of the system, additional excitation energy actually reduces the computation rate.

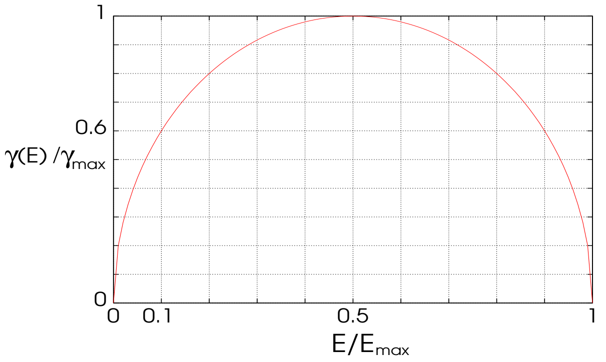

Figure 4: Maximum computation rate per site as a function of excitation energy per site.

4 Conclusion

We have shown that the Feynman synchronization backbone for two-body cellular automata has as its quantum Hamiltonian $\widehat{H}_{1DXY}$. This Hamiltonian is known to accurately describe praseodymium ethyl sulfate at low temperatures[8] and [9]. Furthermore, the energy eigenstates and eigenvalues of this Hamiltonian can be solved for exactly and theoretical predictions obtained for how the computation-rate of the quantum cellular automaton depends on the energy.

Since the synchronization backbone is the only long-ranged part of the full Hamiltonian for a two-body quantum cellular automaton, these results suggests that quantum computation may not require unphysical, highly-contrived Hamiltonians.

In particular, the only part of the Hamiltonian that may turn out to require physically-unrealistic interactions is restricted to the interactions of pairs of cells of the quantum cellular automaton. For the type of cellular automaton considered here, it is known that in order to make a two-body cellular automaton computation-universal[5], no more than six spin-$\frac{1}{2}$ variables per cell are required.

Consequently, the difficulties that remain to be overcome in order to arrive at a physically-realistic, if highly idealized, model of a quantum computer are evidently comparable to the difficulty in finding a physically-realistic Hamiltonian to describe how six spin-$\frac{1}{2}$ quantities might couple to the XY backbone and to one another in such a way as to produce the dynamics of a computation-universal two-body cellular automaton.

References

[1] Benioff, Paul A., "The Computer as a Physical System: A Microscopic Quantum Mechanical Hamiltonian Model of Computers as Represented by Turing Machines", *Journal of Statistical Mechanics* **22**, 563–591 (1980).

[2] Benioff, Paul A., "Quantum Mechanical Hamiltonian Models of Computers", *Annals of the New York Academy of Sciences* **480**, 475–486 (1986).

[3] Bennett, Charles H., "Thermodynamics of Computation", *International Journal of Theoretical Physics* **21**, 905–940 (1982).

[4] Biafore, Michael, *A Computational Role for the One-dimensional XY Model*, PhD thesis, MIT (Physics) (1993).

[5] Biafore, Michael, "Universal computation in few-body automata", *Complex Systems* **7**, 221–239 (1993).

[6] Deutsch, David, "Quantum theory, the Church-Turing principle and the universal quantum computer", *Proceedings of the Royal Academy of London A* **400**, 97–117 (1985).

[7] Feynman, Richard P., "Quantum Mechanical Computers", *Foundations of Physics* **16**, 507 (1986).

[8] Folinsbee, J. T., Harrison, J. P., McColl, D. B., and Taylor, D. R., "Specific heat, electric susceptibility and thermal conductivity of praseodymium ethyl sulphate: A one-dimensional electric dipolar XY system", *Journal of Physics C* **10**, 743–759 (1977).

[9] Harrison, J. P., Hessler, Jan P., and Taylor, D. R., "One- and three-dimensional antiferromagnetic ordering in $PrCl_3$", *Physical Review B* **14**, 2979–2982 (1976).

[10] Landauer, R., "Is Quantum Mechanics Useful?", *Proceedings of the Royal Society London* **A** (1994).

[11] Lieb, E. H., Shultz, T., and Mattis, D. C., "Two Soluble Models of an Antiferromagnetic Chain", *Annals of Physics (N.Y)* **16**, 407–466 (1961).

[12] Margolus, Norman, "Quantum Computation", *Annals of the New York Academy of Sciences* **480**, 487–497 (1986).

[13] Margolus, Norman, "Parallel Quantum Computation", In *Complexity, Entropy and the Physics of Information* (Zurek, W., ed.), 301–318 (1990).

[14] Mattis, D. C., *Theory of Magnetism*, Vol. 1, Springer, Berlin (1985).

[15] Shor, P., "Algorithms for Quantum Computation: Discrete Log and Factoring", *Proceedings of the 26th Annual ACM Symposium on the Theory of Computing* **35**, 1–14 (1994).

[16] Unruh, W. G., "Maintaining coherence in quantum computers", *Physical Review Letters* (1994).