Journal of Statistical Physics, Vol. 29, No. 3, 1982

Quantum Mechanical Hamiltonian Models of Turing Machines

Paul Benioff[1]

Received October 5, 1981; revised June 9, 1982

Quantum mechanical Hamiltonian models, which represent an aribtrary but finite number of steps of any Turing machine computation, are constructed here on a finite lattice of spin-1/2 systems. Different regions of the lattice correspond to different components of the Turing machine (plus recording system). Successive states of any machine computation are represented in the model by spin configuration states. Both time-independent and time-dependent Hamiltonian models are constructed here. The time-independent models do not dissipate energy or degrade the system state as they evolve. They operate close to the quantum limit in that the total system energy uncertainty/computation speed is close to the limit given by the time–energy uncertainty relation. However, the model evolution is time global and the Hamiltonian is more complex. The time-dependent models do not degrade the system state. Also they are time local and the Hamiltonian is less complex.

KEY WORDS: Schrödinger equation description of Turing machines; nondissipative models of computers; quantum spin lattices.

1. INTRODUCTION

In recent years there has been an upsurge of interest in the physical limitations of the computation process. In particular the energy cost of computation or information transfer and whether or not there must be energy dissipation are the subjects of much discussion.[1–10] Some years ago it was felt[3,7] that there must be dissipation associated with the computation process because the process is irreversible. However, in 1973, Bennett[2] constructed reversible models of the computation process and

[1] Division of Environmental Impact Studies, Argonne National Laboratory, Argonne, Illinois 60439.

0022-4715/82/1100-0515$03.00/0

discussed thermodynamically reversible models of computation. Recent papers on the subject[1,10] which assume that energy is dissipated in the computation process have been criticized by Deutsch.[5]

In recent work Landauer[11] has stressed the importance of determining if dissipationless models of the computation process exist. Fredkin and Toffoli[12] have constructed a classical mechanical billiard ball model of the computation process which dissipates no energy. In other work[13-15] quantum mechanical Hamiltonian models of Turing machines and of abstract discrete processes were constructed. These models used successive scatterings to drive the model. Two of these models were dissipative in the sense that as the overall system state evolved the amplitude of undesirable components in the (pure) state increased with time.[14,15] Another model[13] was not dissipative. This was a consequence of the assumption that the kinetic energy (of the scatterer) is a linear function of the momentum.

In this work, quantum mechanical Hamiltonian models of Turing machines will be constructed which avoid the use of successive scatterings. The models will be constructed entirely on a lattice of spin-1/2 systems in which some configurations of spin projections along a fixed axis represent descriptions of the Turing machines at the completion of computation steps. Changes in the system can be represented by spin–flip operators acting on appropriate lattice sites. Since no systems move in the models constructed here, the sources of dissipation which were present in the other models, [14,15] such as wave packet spreading, etc., are absent here.

In the next section a brief review of Turing machines is given. It is followed by the construction of the representation of the complete description of any Turing machine plus record system state as a configuration on the lattice of spin-1/2 systems. The corresponding model configuration states, projection operators, and elementary configuration change operators are given in Section 3. These states and operators are used in Sections 4 and 5 to construct respective time-dependent and time-independent Hamiltonian models for the first J steps of any Turing machine computation.

Sections 6 and 7 discuss characteristics of the models constructed here, and properties and restrictions on measurements made on the model systems. It is seen that the time-independent models dissipate no energy and do not degrade the model state as they evolve. They also operate close to the quantum limit in that the total system energy uncertainty/computation speed $\leqslant 2\pi\hbar$. However, the Hamiltonians are complex. They are also time global which make the carrying out of measurements to determine system parameters quite difficult. Such measurements also necessarily introduce energy dissipation and perturb the system state. [A model is time global if, as the model system state evolves from a state representing stage n to a state representing stage $n + 1$, it passes at intermediate time through

states which are linear combinations of states representing all stages. It is time local if for all intermediate times the state is a linear combination of the stage n and stage $n+1$ states only.]

The time-dependent Hamiltonian models also do not degrade the system state. The Hamiltoians are less complex and the model evolution is time local. As a result measurements to determine if the computation has halted are neither so difficult to carry out as they are for the time-independent models nor do they perturb the system state or introduce energy dissipation. However, an external agent is required to turn on and off the successive step Hamiltonians. Section 8 compares some aspects of the models constructed here with those constructed elsewhere.[13–15]

2. TURING MACHINES

2.1. Preliminaries

Since Turing machines have been described elsewhere[2,16] the description given here will be brief. Turing machines consist of three parts, an internal machine $\mathfrak{L}$, a computation tape $\mathfrak{T}$, and a computation head **h**. The states of $\mathfrak{L}$ will be represented here by the numbers 0, 1, . . . in N. $\mathfrak{T}$ is an infinite array of cells where each cell can assume any one of a finite number of states in S, the tape symbol alphabet. A special element b of S denotes the blank. The expressions on $\mathfrak{T}$ are given by any symbol sequence $\gamma: Z \to S$ where Z is the set of integers and $\gamma(j) = b$ except for at most a finite number of j values. $(S)_b^Z$ denotes the set of all such γ. **h** scans one cell at a time with its states given by the cell labels j in Z.

The basic operations of the machines are represented by quintuples of the form $l(s,s'\alpha)l'$ which states that $\mathfrak{L}$ in state l and the symbol s in the cell of $\mathfrak{T}$ scanned by **h** are changed to state l' and symbol s' and **h** is either shifted one cell to the right ($\alpha = +1$), or to the left ($\alpha = -1$) or stays where it is ($\alpha = 0$). Each Turing machine corresponds to a finite set Q of quintuples no two of which begin with the same two symbols. If at the end of a step, $\mathfrak{L}$ is in state l and s is the tape symbol scanned by **h**, the next step of a machine Q is given by the quintuple in Q of the form $l(s,- -)$-. If no such quintuple is in Q, the machine halts.

Each machine Q defines a function $\tau_Q: N \times S \to N \times S \times \{-1, 0, +1\}$, where for each (ls) if there is a quintuple, $l(s,s'\alpha)l'$ in Q beginning with l and s then

$$\tau_Q(ls) = (l's'\alpha) \tag{1}$$

If no quintuple in Q begins with l and s then $\tau_Q(lS) = (ls0)$.

From τ_Q one can define a machine transfer function, T_Q, as a map $T_Q: ID \to ID$ where $ID = N \times (S)_b^Z \times Z$ is the set of all instantaneous descriptions of the machine. T_Q is defined from τ_Q by

$$T_Q(l\gamma j) = (l'\gamma' j') \tag{2}$$

where $\tau_Q(l, \gamma(j)) = (l'\gamma'(j)\alpha), j' = j + \alpha$, and $\gamma'(k) = \gamma(k)$ for all $k \neq j$. The steps of Q correspond to iterations of T_Q and the process halts at a fixpoint of T_Q.

It is convenient to restrict Turing machines to those which carry out computations in a standard form. That is, at the outset $\mathfrak{L}$ is in state "1", **h** is scanning cell "0" of $\mathfrak{T}$, and the initial expression $\gamma_i(j) = b$ if $j < 0$ and no two nonblank symbols on γ_i are separated by a blank. Also the states of $\mathfrak{L}$ are arranged in the quintuples of Q in a standard ordering. That is, after n steps of any computation by any standard Turing machine, the state of $\mathfrak{L}$ lies in the first N_n numbers in N where

$$N_n = \sum_{j=0}^{n} m^j. \tag{3}$$

Here m is the number of symbols in S. The standard form of the final state is similar to the initial state except that $\mathfrak{L}$ is in a different designated state.

In what follows, numbers will be represented on the lattice model as binary strings of spin up (+) and spin down (−). The representation of all positive numbers $\leqslant n$ requires binary strings of length $l_2(n)$, where

$$l_2(n)\begin{cases} = [\ln_2(n)] + 1 & \text{if} \quad \ln_2(m) - [\ln_2(m)] > 0 \\ = \ln_2(n) & \text{if} \quad \ln_2(m) - [\ln_2(m)] = 0 \end{cases} \tag{4}$$

$[r]$ denotes the largest integer contained in r. Blank cells of $\mathfrak{T}$ will be modelled as strings of (−) spins. In representations of binary numbers on the lattice, spin up (+) corresponds to 1 and spin down (−) to 0. Thus, any blank cell of $\mathfrak{T}$ corresponds to the number 0 recorded in the cell.

From now on, we shall consider systems which model the first J steps only of any standard Turing machine computation. This restriction is done purely in the interests of mathematical simplicity to avoid dealing with the quantum mechanics of infinite-dimensional systems. A consequence of this restriction is that the states of $\mathfrak{L}$ then lie in the set $\{1, 2, \ldots, N_j\}$.

In general, the transfer function T_Q, for Turing machines is many–one. To construct a Hamiltonian model of a discrete process, it is necessary that the step function (or transfer function) for the process be one–one. This is done here by addition of a record system $\mathfrak{R}$ and a recording head **j**. For each Turing machine step, three types of operations are considered: a recording operation, a compute operation, and a shift operation. In the recording operation, the state of $\mathfrak{L}$, the contents of the $\mathfrak{T}$ cell scanned by **h**,

and the position of **h** are recorded in the (blank) cell of $\mathfrak{R}$ scanned by **j**. In the compute operation, the state of $\mathfrak{L}$, the contents of the $\mathfrak{T}$ cell scanned by **h**, and the position of **h** are changed corresponding to the quintuple of Q whose first two symbols are recorded in the $\mathfrak{R}$ cell scanned by **j**. The third type of operation shifts **j** to a fresh record cell. These three types of operations will be modeled on the lattice either as three types of steps repeated over and over in the order given (Section 4) or as one step, which combines the operations, repeated over and over (Section 5).

2.2. Spin Lattice Model

The overall system model is constructed on a two-dimensional lattice of spin-1/2 systems. Each component system will be modeled as a sublattice of spin systems. Besides sublattices for the internal machine $\mathfrak{L}$, the computation tape $\mathfrak{T}$, and the computation head **h**, there are sublattices for the recording head **j** and the record system $\mathfrak{R}$. Figure 1 shows the sublattices. A more detailed description follows.

A spin-1/2 lattice model of $\mathfrak{L}$ valid for J steps of any standard Turing machine calculation requires a lattice region $R_{\mathfrak{L}}$ of at least $l_2(N_J)$ sites. For convenience $R_{\mathfrak{L}}$ is here taken to extend for $J+1$ sites in the x direction, from position 0 to J, and M sites in the y direction, from position 0 to $M-1$. Here and in what follows $M = l_2(m)$, the length of binary strings needed to represent the symbols, is S. Note that since $N_J \leqslant m^{J+1}$ by Eq. (2), $MJ + J \geqslant l_2(N_J)$.

Each state l of $\mathfrak{L}$ which is reachable in $\leqslant J$ steps, when considered as a number in $\{1, \ldots N_J\}$, has an inverted binary representation as a finite string of zeros and ones. That is $2 = 01$, $3 = 11$, $4 = 001$, etc. The inversion is done so that one can extend the representation as a 0, 1 sequence on $\{1 \ldots M \cdot (J+1)\}$ by adding zeros to the right without changing the value. In what follows l will denote either the number or its extended inverted binary representation. It will be clear from context which is intended.

Let Θ be a fixed map which well-orders the sites of $R_{\mathfrak{L}}$. An example is $\Theta(j,k) = jM + k$ for the y coordinate $k = 0, 1 \ldots M-1$ and x coordinate $j = 0, \ldots, J$. Θ is a bijection from $R_{\mathfrak{L}}$ to $\{0, \ldots [(J+1) \cdot M)] - 1\}$. Then each state l of $\mathfrak{L}$ corresponds to a spin configuration F_l on $R_{\mathfrak{L}}$ given by $F_l(i,j) = l(\Theta(i,j))$ for each site (i,j) in $R_{\mathfrak{L}}$. This example corresponds to laying out the inverted binary representation of the state l as follows: the first M zeros and ones along the line of M spins in the y direction at $x = 0$, the next M zeros and ones along the M spins at $x = 1, \ldots,$ and the last M zeros and ones along the M spins at $x = J$.

The computation tape $\mathfrak{T}$ is modeled by a rectangular region $R_{\mathfrak{T}}$ of length $2J+1$, from $-J$ to J, in the x direction, and of length M, from M

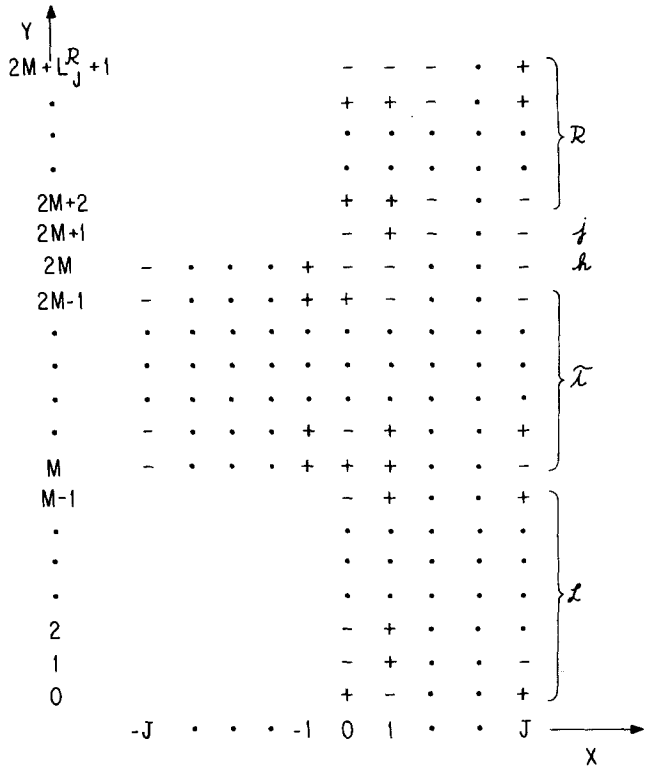

Fig. 1. A representation of the lattice model of the overall computing system. The X and Y components of the positions of the lattice sites are given by the numbers from $-J$ to J and from 0 to $2M + L_J^R + 1$, respectively. The lattice regions for the $\mathfrak{L}$, **j**, and $\mathfrak{R}$ component systems extend from 0 to J in the X direction and for the $\mathfrak{T}$ and **h** components the regions extend from $-J$ to J. The extent and positions of the $\mathfrak{L}$, $\mathfrak{T}$, and $\mathfrak{R}$ regions in the Y direction is given by the curly brackets. The sites for the heads **h** and **j** occupy one row each at Y positions $2M$ and $2M + 1$, respectively. + denotes spin up and − denotes spin down. The dots indicate that the regions are filled with one spin-1/2 system at each site.

to $2M - 1$, in the y direction, Fig. 1. For each j where $-J \leqslant j \leqslant J$, the sublattice of $R_{\mathfrak{T}}$ of sites at x position j and extending from M to $2M - 1$ in the y direction corresponds to the jth cell of the tape. The length of the tape is dictated by the fact that in a standard computation, the head starts in the center and in J steps can move at most J steps to the left or right.

Assume a given representation of S to the set of $+, -$ strings of length M. Then the spin configuration in the sublattice of $R_{\mathfrak{T}}$ described above corresponds to the contents of the jth cell of $\mathfrak{T}$. This can be extended in an obvious way so that each tape expression γ corresponds to a configuration on $R_{\mathfrak{T}}$. In what follows, depending on context, γ will denote either a tape

expression or the corresponding spin configuration on $R_{\mathfrak{T}}$. It will also be assumed that under the given representation, a blank cell corresponds to a string of $-$ signs or all spins down in the corresponding sublattice of $R_{\mathfrak{T}}$.

The computation head **h** is modeled as a line of spins at y position $2M$ extending from $-J$ to J in the x direction. All spins in the line are down except one whose x position denotes the position of **h** and the $\mathfrak{T}$ cell scanned. For example, the configuration $-, -, \ldots, -, +-, \ldots, -$, where the spin up $(+)$ system is at position j represents **h** at position j.

The recording head **j** is modeled in the same way as **h** except that the sublattice of spins occupies y position $2M + 1$ and extends from 0 to J in the x direction.

The model of the record tape system $\mathfrak{R}$ is somewhat more complex because a triple of numbers can be recorded in each cell, or the cell can be blank. One records in each cell the state of $\mathfrak{L}$, the $\mathfrak{T}$ cell symbol scanned by **h**, and the position of **h**. For the first J steps of any calculation and for any standard Turing machine, the state of $\mathfrak{L}$ will lie in N_J, the contents of the $\mathfrak{T}$ cell scanned by **h** will lie in S, and the position of **h** will lie in $\{-J, J\}$.

Assume a one–one map of $(N_J \times S \times \{-J, J\}) \cup \{b\}$ into the set of all binary sequences of length $L_J^{\mathfrak{R}}$. The extra b allows for the fact that the record cell can be blank. Since the map must be onto or into, one must have $L_J^{\mathfrak{R}} \geqslant l_2\,[(N_J \cdot m \cdot (2J + 1) + 1]$; the equality is taken here. m is the number of elements in S. A standard example of such a map is the function Φ defined by

$$\Phi(lsj) = 2_J(K(K(l, s), u(j)) + 1) \tag{5}$$

and $\Phi(b) = 2_J(0)$. Here u maps $\{-J, J\}$ to $\{0, \ldots; 2J\}$ according to $u(j) = 2j + 1$ if $j > 0$ and $u(j) = -2j$ if $j \leqslant 0$. K is the pairing function[17] defined by $K(m, n) = \frac{1}{2}(m^2 + 2mn + n^2 + 3m + n)$. The symbol s on the right-hand side denotes the value of s in $\{1, m\}$ under a fixed bijection of S to $\{1, m\}$. $2_J(n)$ gives the usual binary representation of n extended by zeros to the left so that $2_J(n)$ has length L_J^R for all $n \leqslant [N_J \cdot m \cdot (2J + 1)] + 1$.

The $\mathfrak{R}$ lattice region $R_{\mathfrak{R}}$ extends from 0 to J in the x direction and from $2M + 2$ to $2M + 1 + L_J^R$ in the y direction, Fig. 1. The contents of the kth record cell is modeled by the spins in the sublattice of $R_{\mathfrak{R}}$ at x position k and which extends from $2M + 2$ to $2M + 1 + L_J^R$ in the y direction. The representation $2_J(j)$ of j in each model cell is organized so that at y position $2M + 2 + n$, a $+$ spin corresponds to 1.2^n and a $-$ spin to 0.2^n. This corresponds to the usual binary representation $j = \sum_{n=0}^{L^R_J} [2_J(j)](n) \cdot 2^n$.

Let ϕ be a map from $\{0, J\}$ to $(N_J \times S \times \{-J, J\}) \cup \{b\}$. Then ϕ gives the contents of the record system with $\phi(k)$ the contents of the kth record cell. One can use the map ϕ to construct from each record expres-

sion ϕ, a spin configuration G_ϕ on $R_{\mathcal{R}}$ where for each lattice position $(k, 2M+2+j)$

$$G_\phi(k, 2M+2+j) = [\Phi(\phi(k))](j) \tag{6}$$

For simplicity and in order to have one fixed lattice model to represent all machines, the size of the lattice is larger than is necessary. For example, both $R_{\mathcal{R}}$ and $R_{\mathcal{L}}$ can be greatly reduced if the lattice model is to apply to one machine only.

It is helpful to give a concrete realization of the foregoing model representation. To this end, let $S = \{b, s_1, s_2\}$, where b, s_1, s_2 correspond to the numbers 0, 1, 2. Then $m = 3$ and $l_2(m) = 2$. Let $J = 5$. Then Fig. 2 gives the state of the lattice resulting after the operations triple, record–compute–j-shift have occurred twice where the quintuples used for the two compute

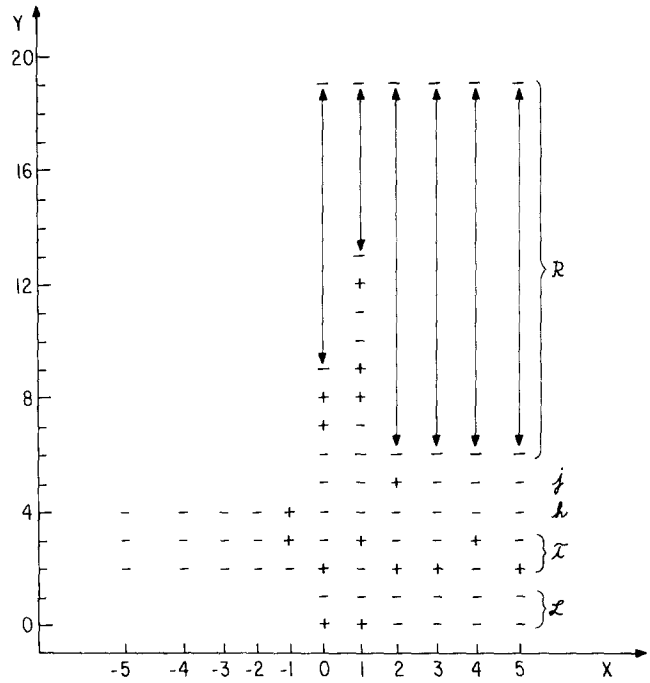

Fig. 2. The lattice model spin configuration for the example given in the text. The Y and X lattice site position coordinates are given by the ordinate and abscissa scales, respectively. The two-headed arrows denote sites occupied with spin down (−) systems. As in the previous figure, the regions associated with each component system are denoted by the curly brackets and/or the script letters.

operations are $1(b, 1-1)3$ and $3(b, 20)5$. The configuration on $R_{\mathfrak{L}}$ corresponds to the inverse binary representation of 5, or $1010\ldots0$, as $\mathfrak{L}$ is in state 5. The expression on $\mathfrak{T}$ is, from left to right, $bbbbs_2s_1s_2s_1s_1s_2s_1$. In particular, s_2 must be at x position -1, and s_1 must be at position 0 with the remaining symbols ($s_2s_1s_1s_2s_1$ which are arbitrary) given as part of the standard input. **h** is at position -1 as given by the quintuples and **j** is at position 2 scanning a blank record cell. All record cells to the right of **j** are also blank.

The length of the $\mathfrak{R}$ sublattice in the y direction is given by $L_J^R = l_2(N_J \cdot m(2J+1)+1)$. Since $N_J = 364$ [Eq. (2)], $L_J^R = 14$. According to the quintuples, the first and second record cells have $(1,0,0)$ and $(3,0,-1)$ recorded in them. This means that $\Phi(1,0,0) = 2_J(6) = \ldots 110$ and $\Phi(3,0,-1) = 2_J(76) = \ldots 1001100$ are recorded in cells 0 and 1 of $\mathfrak{R}$. This is shown in Fig. 2.

3. MODEL STATES AND OPERATORS

3.1. Model States

Let $\psi_+(i,j)$ and $\psi_-(i,j)$ denote the respective spin up and spin down states for the spin-1/2 system at lattice site (i,j). ψ_+ and ψ_- are given by the respective column vectors $\binom{1}{0}$ and $\binom{0}{1}$ in the representation under consideration.

Let f by any configuration defined over a subregion R of the lattice. Then the configuration state Ψ_f is given by

$$\Psi_f = \bigotimes_{(i,j)\in R} \psi_{f(i,j)}(i,j) \tag{7}$$

This generic definition can be used to give the definitions of the configuration states needed here. $\Psi_l^{\mathfrak{L}}$, the state which corresponds to $\mathfrak{L}$ being in state l, is defined by Eq. (7) with $R = R_{\mathfrak{L}}$ (Fig. 1). Here l denotes either the state of $\mathfrak{L}$ or the corresponding configuration of spins over $R_{\mathfrak{L}}$. It will be clear from context which is meant.

$\Psi_\gamma^{\mathfrak{T}}$, $\Psi_j^{\mathbf{h}}$, $\Psi_k^{\mathfrak{L}}$, and $\Psi_\phi^{\mathfrak{R}}$ are defined similarly. Note that

$$\Psi_\gamma^{\mathfrak{T}} = \bigotimes_{j=-J}^{J} \Psi_{\gamma(j)}^{\mathfrak{T}_j} \tag{8}$$

where $\Psi_{\gamma(j)}^{\mathfrak{T}_j}$ is defined over the region $R_{\mathfrak{T}_j}$ of the lattice which corresponds to cell j of $\mathfrak{T}$ and is the configuration state coresponding to symbol $\gamma(j)$ in cell j of $\mathfrak{T}$. A similar definition holds for $\Psi_\phi^{\mathfrak{R}}$ which corresponds to expression ϕ in the cells of $\mathfrak{R}$. Equation (6) is used to give the configuration

corresponding to ϕ. The states $\Psi_j^{\mathbf{h}}$ and $\Psi_k^{\mathbf{j}}$ which correspond to the computation head **h** at position j and the record head at position k are defined by Eq. (7) over R_h and $R_{\mathbf{j}}$ (Fig. 1). For **h** at j, $f(i,2M) = -$ if $i \neq j$ and $f(j,2M) = +$. For **j** at position k $f(i,2M+1) = -$ if $i \neq k$ and $f(k,2M+1) = +$.

The overall lattice state for the spin configuration which describes $\mathfrak{L}$ in state l, $\mathfrak{T}$ with expression γ, $\mathfrak{R}$ with expression ϕ, and the heads **h** and **j** in positions j and k is given by

$$\Psi_{l\gamma jk\phi} = \Psi_l^{\mathfrak{L}} \otimes \Psi_\gamma^{\mathfrak{T}} \otimes \Psi_j^{\mathbf{h}} \otimes \Psi_k^{\mathbf{j}} \otimes \Psi_\phi^{\mathfrak{R}} \tag{9}$$

Such states are a subset of all possible lattice configuration states given by $\Psi_f = \otimes(m,n) \in R_J \Psi_{f(m,n)}(m,n)$, where R_J is the whole lattice region shown in Fig. 1 and f is an arbitrary configuration on R_J.

3.2. Model Projection Operators

The model projection operators needed here can also be obtained from a basic definition. As before let f be a configuration defined on a region R of the lattice. Then the projection operator for finding the systems in region R in the configuration state Ψ_f is defined by

$$P_f = \bigotimes_{(i,j)\in R} P_{f(i,j)}(i,j) \tag{10}$$

where the projection operator for finding the spin-1/2 system at site (i,j) with spin up(+) or down(−) is given by

$$P_\pm(i,j) = \frac{1 \pm \sigma_3(i,j)}{2} \tag{11}$$

Here $\sigma_3(i,j)$ is the Pauli spin matrix $\left(\begin{smallmatrix} 1 & 0 \\ 0 & -1 \end{smallmatrix}\right)$ for the spin of site (i,j). From the definition one has that for any configuration g defined over R

$$P_f \Psi_g = \Psi_f \delta_{f,g}$$

where $\delta_{f,g} = 1$ if $f = g$ and 0, if $f \neq g$.

By means of the above one can define $P_l^{\mathfrak{L}}$, over the region $R_{\mathfrak{L}}$, $P_\gamma^{\mathfrak{T}}$ over $R_{\mathfrak{T}}$, $P_j^{\mathbf{h}}$ over $R_{\mathbf{h}}$, $P_k^{\mathbf{j}}$ over $R_{\mathbf{j}}$ and $P_\phi^{\mathfrak{R}}$ over $R_{\mathfrak{R}}$. Note that the projection operator for finding expression γ on $\mathfrak{T}$ can be written as

$$P_\gamma^{\mathfrak{T}} = \bigotimes_{j=-J}^{\mathfrak{T}} P_{\gamma(j)j}^{\mathfrak{T}} \tag{12}$$

where $P_{\gamma(j)j}^{\mathfrak{T}}$ is the projection operator for finding symbol $\gamma(j)$ in cell j of γ. A similar decomposition holds for $P_\phi^{\mathfrak{R}}$ the projection operator for finding expression ϕ in the cells of $\mathfrak{R}$. Note that, for example $P_l^{\mathfrak{L}}$ is the projection operator for finding $\mathfrak{L}$ in configuration state $\Psi_l^{\mathfrak{L}}$ whereas one speaks of $P_l^{\mathfrak{L}}$

as the projection operator for finding $\mathfrak{L}$ in state l. This confusion between configuration states and system states which they represent will be continued as it will always be clear from context which is meant. For each complete machine description $l\gamma jk\phi$, the projection operator for finding the system in state $\Psi_{l\gamma jk\phi}$, Eq. (9), is given by

$$P_{l\gamma jk\phi} = P_l^{\mathfrak{L}} \otimes P_\gamma^{\mathfrak{I}} \otimes P_j^{\mathfrak{h}} \otimes P_k^{\mathfrak{j}} \otimes P_\phi^{\mathfrak{R}} \tag{13}$$

3.3. Model Configuration Change Operators

Let f and g be two configurations defined over the same region R of the lattice. Define σ_{fg} by

$$\sigma_{fg} = \bigotimes_{(i,j)\in D_{fg}} \sigma_1(i,j) \tag{14}$$

Here $D_{fg} = \{(i,j)|f(i,j) \neq g(i,j)\}$ is the set of all lattice sites at which f differs from g and $\sigma_1(i,j)$ is the spin–flip operator for the system at site (i,j). σ_1 is the Pauli matrix $\left(\begin{smallmatrix}0&1\\1&0\end{smallmatrix}\right)$ which exchanges ψ_+ and ψ_-. σ_{fg} is the operator which exchanges Ψ_f and Ψ_g. That is $\sigma_{fg}\Psi_f = \Psi_g$ and $\sigma_{fg}\Psi_g = \Psi_f$. In fact since $\sigma_1^2 = 1$, one has $\sigma_{fg}^2 = 1$.

Note that $\sigma_{fg}\Psi_h \neq \Psi_h$ for all other configurations whose domain of definition has a nonempty overlap with D_{fg}. If $f = g$ then D_{fg} is empty and $\sigma_{fg} = 1$.

It is convenient to generalize the above somewhat and consider the unitary operators U_{fg} defined by

$$U_{fg} = e^{i\beta(f,g)}\sigma_{fg} \tag{15}$$

where $\exp[i\beta(f,g)]$ is a phase factor which can depend on f and g. U_{fg} is unitary, and is self-adjoint if and only if $\beta(f,g) = 0 \pmod{2\pi}$.

As is well known, the exact form of $\beta(f,g)$ depends on the form of the interaction used to generate the exchange operation. Here the interaction Hamiltonian H_{fg} will be taken to have the form

$$H_{fg} = \frac{\pi\hbar}{2\Delta}\sigma_{fg} \tag{16}$$

where Δ is an arbitrary time interval. Then with $U_{fg}(t)$ defined by

$$U_{fg}(t) = e^{-itH_{fg}/\hbar} \tag{17}$$

one has

$$U_{fg}(t) = \cos\left(\frac{\pi t}{2\Delta}\right) - i\sigma_{fg}\sin\left(\frac{\pi t}{2\Delta}\right) \tag{18}$$

In this case

$$U_{fg}(\Delta) = U_{fg} = -i\sigma_{fg} \tag{19}$$

where $\beta(f, g) = 3\pi/2$ independent of f and g. Many other choices of H_{fg} are possible but these will not be examined here.

4. TIME-DEPENDENT HAMILTONIAN MODELS

4.1. Record, Compute, and Shift Steps

In this section models of the computation process will be constructed in which each step in the process is replaced by three steps: record, then compute, then shift. The reason this is done is that it becomes possible to arrange things so that the systems whose configuration states determine which configuration change operations are to be used are different from the systems on which the configuration changes are carried out. Speaking crudely, one requires that in each step the systems examined are different from the systems whose states are changed.

The reason this requirement is imposed is that it results in a relatively simple Hamiltonian description for each step of the process. This is a consequence of the fact that the projection operators which function as system examining operators commute with the system configuration change operators. This would not be the case if the projection operators referred to the same systems as the configuration change operators and would result in a more complex Hamiltonian.

The function of the record step in the forward compute phase is to record the state of $\mathfrak{L}$, the contents of the $\mathfrak{T}$ cell scanned by **h**, and the position of **h** into the blank record cell scanned by **j**. The compute step carries out on the system $\mathfrak{L} + \mathfrak{T} + \mathbf{h}$, the operation defined by τ_Q, Eq. (1), whose arguments are given by the values of l and s in the record cell scanned by **j**. The recorded position of **h** is used to choose the position at which the **h** head shift, if any, will occur. The third type shifts the recording head **j** to a fresh record cell.

The operator V_1 for the record operation is given by

$$V_1 = \sum_{l=1}^{N_J} \sum_{s \in S} \sum_{j=-J}^{J} \sum_{k=0}^{J} P_l^{\mathfrak{L}} \otimes P_{sj}^{\mathfrak{T}} \otimes P_j^{\mathbf{h}} \otimes P_k^{\mathbf{j}} \otimes U_{(lsj)b}^{\mathfrak{R}k} + 1 - P_1 \tag{20}$$

where

$$P_1 = \sum_{l=1}^{N_J} \sum_{s \in S} \sum_{j=-J}^{J} \sum_{k=0}^{J} P_l^{\mathfrak{L}} \otimes P_{sj}^{\mathfrak{T}} \otimes P_j^{\mathbf{h}} \otimes P_k^{\mathbf{j}} \otimes 1^{\mathfrak{R}}$$

The operator $U^{\mathfrak{R}k}_{(lsj)b}$ is given by Eq. (19) with f and g the respective configurations (lsj) and b in cell k of $\mathfrak{R}$. The $1 - P_1$ term takes care of the fact that there are spin configurations on the overall lattice model which do not correspond to any desired state as defined by Eq. (9). An example is a configuration with more than one spin up in the **h** or **j** sublattice.

It is clear that V_1 satisfies the requirement since $\mathfrak{L}$, $\mathfrak{T}$, **h**, and **j** are the systems examined and $\mathfrak{R}$ is the system whose configuration is changed. V_1 functions as follows: if the record cell scanned by **j** is blank then V_1 records into the cell the state of $\mathfrak{L}$, the symbol in the cell $\mathfrak{T}$ which is scanned by **h**, and the position of **h**. Conversely, if the record cell scanned by **j** already contains a correct record of the state of $\mathfrak{L}$, the symbol of $\mathfrak{T}$ scanned by **h** and the position of **h**, then V_1 erases the record cell. If the record in the cell scanned by **j** does not correspond to the state of $\mathfrak{L}$, the symbol of $\mathfrak{T}$ scanned by **h** and the position of **h**, then V_1 makes other changes on the record cell scanned by **j**. However, these are of no consequence here.

Mathematically the above is expressed by

$$V_1 \Psi_{l\gamma jk\phi} = -i\Psi_{l\gamma jk\phi'} \tag{21}$$

where $\phi'(k) = (l, \gamma(j), j)$ if $\phi(k) = b$ and $\phi'(k) = b$ if $\phi(k) = (l, \gamma(j), j)$. Note that $\phi(h) = \phi'(h)$ for all $h \neq k$. V_1 also makes changes on $\phi(k)$ if $\phi(k)$ has other values. However, these are not of concern here.

The operator for the compute operation is given by

$$V_2 = \sum_{l=1}^{N_J} \sum_{s \in S} \sum_{j=-J}^{J} \sum_{k=0}^{J} U^{\mathfrak{L}\mathfrak{T}\mathbf{h}j}_{(ls),\ (ls'\alpha)} \otimes P^{\mathbf{j}}_k \otimes P^{\mathfrak{R}}_{(lsj)k} + 1 - P_2 \tag{22}$$

where

$$P_2 = \sum_{l=1}^{N_J} \sum_{s \in S} \sum_{j=-J}^{J} \sum_{k=0}^{J} 1^{\mathfrak{L}\mathfrak{T}\mathbf{h}} \otimes P^{\mathbf{j}}_k \otimes P^{\mathfrak{R}}_{(lsj)k}$$

and the $1 - P_2$ term serves the same purpose as the $1 - P_1$ term in Eq. (20). l', s', and α are defined by $\tau_Q(ls) = (l's'\alpha)$, Eq. (1).

By Eq. (19), one has

$$U^{\mathfrak{L}\mathfrak{T}\mathbf{h}j}_{(ls),\ (l's'\alpha)} = -i\sigma^{\mathfrak{L}}_{ll'} \otimes \sigma^{\mathfrak{T}j}_{ss'} \otimes \sigma^{\mathbf{h}j}_{\alpha} \tag{23}$$

For $\sigma^{\mathfrak{L}}_{ll'}$ f and g are the configurations for l and l' on $R_{\mathfrak{L}}$ (Fig. 1). For $\sigma^{\mathfrak{T}j}_{ss'}$ f and g are the configurations for s and s' on $R_{\mathfrak{T}j}$, the region of the lattice for cell j of $\mathfrak{T}$, and for $\sigma^{\mathbf{h}j}_{\alpha}$ f and g are the configurations for j and $j + \alpha$ on region $R_{\mathbf{h}}$. [Note that positions $J + 1$ and $-(J + 1)$ become $-J$ and J, respectively.] If $\alpha = 0$, then $\sigma^{\mathbf{h}j}_{\alpha}$ is the identity operator.

V_2 also satisfies the requirement noted before since $\mathfrak{R}$ and **j** are the examined systems and $\mathfrak{L}$, $\mathfrak{T}$, and **h** are the systems whose configurations are

changed. V_2 functions as follows: if the cell of $\mathfrak{R}$ scanned by **j** contains some record lsj which correctly represents the state of $\mathfrak{L}$, the contents of the cell of $\mathfrak{T}$ scanned by **h**, and the position of **h**, then V_2 carries out a computation step on $\mathfrak{L}$, $\mathfrak{T}$, and **h** in that it changes l, s, and j to l', s', and $j+\alpha$ where $\tau_Q(l,s)=(l's'\alpha)$. If the cell of $\mathfrak{R}$ scanned by **j** contains lsj but the state l' of $\mathfrak{L}$, the contents s' of the $\mathfrak{T}$ cell scanned by **h** and the position j' of **h** are related to lsj by $\tau_Q(ls)=(l's'\alpha)$ with $j'=j+\alpha$, then V_2 undoes a computation step by changing l' to l, s' to s and j' to j. If the contents of the record cell scanned by **j** and the state of $\mathfrak{L}$ the symbol in the cell of $\mathfrak{T}$ scanned by **h** and the position of **h** are not related as described above, then V_2 also changes the states of $\mathfrak{L}$, $\mathfrak{T}$ and **h**. However, these changes are of no concern here.

Mathematically, this is expressed by

$$V_2\Psi_{l_1\gamma_1 j_1 k\phi}=-i\Psi_{l'\gamma' j' k\phi} \tag{24}$$

The first change is that for which for some (l,s,j) $\phi(k)=(l,s,j)$ and $l_1=l$, $\gamma_1(j)=s$ and $j_1=j$. In this case, which corresponds to a compute step (l',γ',j') satisfy $\tau_Q(l,s)=(l',\gamma'(j),\alpha)$ with $j'=j+\alpha$ and $\gamma'(h)=\gamma_1(h)$ for all $h\neq j$. The other change is that for which $\phi(k)=(lsj)$ and $\tau_Q(ls)=(l_1s_1\alpha)$ where $j_1=j+\alpha$ and $\gamma_1(j_1)=s_1$. In this case which corresponds to an inverse compute step $l'=l$, $\gamma'(j_1)=s$, $j'=j$, and $\gamma'(k)=\gamma_1(k)$ for all $k\neq j_1$. V_2 also makes other changes.

The operator for the shift step is defined by

$$V_3=\sum_{k=0}^{J}1^{\mathfrak{L}\mathfrak{T}\mathbf{h}}\otimes U_{+1}^{\mathbf{j}k}\otimes P_k^{\mathfrak{R}}+1-P_3 \tag{25}$$

where $P_3=\sum_{k=0}^{J}1^{\mathfrak{L}\mathfrak{T}\mathbf{h}\mathbf{j}}\otimes P_k^{\mathfrak{R}}$. $P_k^{\mathfrak{R}}$ is the projection for all model states of the record sublattice such that cell k of $\mathfrak{R}$ is the last (in the direction of increasing k) nonblank cell. It is defined by $P_k^{\mathfrak{R}}=\sum_{\phi}^{k}P_\phi^{\mathfrak{R}}$ where the superscript k on the ϕ sum denotes the restriction to those ϕ such that $\phi(k)\neq b$ and $\phi(j)=b$ for all j such that $k<j\leqslant J$.

For V_3, one has

$$V_3\Psi_{l\gamma jk\phi}=-i\Psi_{l\gamma jk'\phi} \tag{26}$$

where $k'=k+1$ if k is the last filled cell in ϕ and $k'=k-1$ if $k-1$ is the last filled cell in ϕ. [Note that $k+1=0$ if $k=J$ and $k-1=J$ if $k=0$.]

The above can be used to show that the unitary operators V_1, then V_2, then V_3 applied over and over in the order given to an appropriate initial state $\Psi_{1\gamma}00\underline{b}$ generate the desired steps of the process. Here 1 denotes the (standard) initial state of L and $\underline{b}$ means that all cells of $\mathfrak{R}$ are blank. In particular, if $m\leqslant J$ then $(V_3V_2V_1)^m\Psi_{1\gamma 00\underline{b}}$ is the model state corresponding to the completion of m computation steps. Details on the model state for $m>J$ will be given in Section 6.1.

4.2. Time-Dependent Hamiltonians

The goal is to construct for each Turing machine, a corresponding model lattice Hamiltonian H such that the Schrödinger evolution of selected configuration states models the first J steps in the Turing machine computation. To this end, let Δ be a convenient time interval and define H by

$$H = \begin{cases} H_1 & \text{if } 3m\Delta \leqslant t < (3m+1)\Delta \\ H_2 & \text{if } (3m+1)\Delta \leqslant t < (3m+2)\Delta \\ H_3 & \text{if } (3m+2)\Delta \leqslant t < (3m+3)\Delta \end{cases} \tag{27}$$

for each integer m. H_1, H_2, and H_3 are step interaction Hamiltonians which satisfy ($\hbar = 1$)

$$V_j = e^{-i\Delta H_j} \tag{28}$$

for $j = 1, 2, 3$.

It is clear that H, as defined, is time dependent as it requires an external agency to switch on and off the appropriate interactions. However, the time evolution under H does proceed as desired. To see this, let $t = (3m + h)\Delta$ where $h = 0$, 1, or 2. Then if the whole lattice system is in state Ψ at time 0, the state $\Psi(t)$ at time t is given by

$$\Psi(t) = e^{-iHt}\Psi = e^{-iH_{h+1}\Delta} \ldots e^{-iH_1\Delta}\left(e^{-iH_3\Delta}e^{-iH_2\Delta}e^{-iH_1\Delta}\right)^m \Psi \tag{29}$$

which is the desired result.

One sees from the above that for times $t = (3m + h)\Delta$ where $m \leqslant J$, $\exp[-iHt]$ applied to the state $\Psi_{1\gamma 00\underline{b}}$ generates a model state describing the Turing machine plus recording system after $3m + h$ steps. If $m > J$, the state $\Psi(t)$ is given by Eq. (29), but the model no longer describes the record–compute–shift evolution of the Turing machine plus record system.

The step interaction Hamiltonian for the recording step is taken to be

$$H_1 = \sum_{l=1}^{N_J} \sum_{s \in S} \sum_{j=-J}^{J} \sum_{k=0}^{J} P_l^{\mathfrak{L}} \otimes P_{sj}^{\mathfrak{T}} \otimes P_j^{\mathfrak{h}} \otimes P_k^{\mathfrak{J}} \otimes H_{(lsj)b}^{\mathfrak{R}k} \tag{30}$$

where $H_{(lsj)b}^{\mathfrak{R}k}$ is given by Eq. (16) where the two configurations f and g denote those for (lsj) and b in cell k of the record system. For H_2, one has

$$H_2 = \sum_{l=1}^{N_J} \sum_{s \in S} \sum_{j=-J}^{J} \sum_{k=0}^{J} H_{(ls),(l's'\alpha)}^{\mathfrak{L}\mathfrak{T}\mathfrak{h}j} \otimes P_k^{\mathfrak{J}} \otimes P_{(lsj)k}^{\mathfrak{R}} \tag{31}$$

From Eq. (16) one has

$$H_{(ls),(l's'\alpha)}^{\mathfrak{L}\mathfrak{T}\mathfrak{h}j} = \frac{\pi\hbar}{2\Delta} \sigma_{ll'}^{\mathfrak{L}} \otimes \sigma_{ss'}^{\mathfrak{T}j} \otimes \sigma_{\alpha}^{\mathfrak{h}j} \tag{32}$$

where the configurations f and g denote respectively those for l and l' on the system $\mathfrak{L}$, for s and s' in the jth cell of $\mathfrak{T}$, and positions j and $j+\alpha$ for $\mathbf{h}$.

For each value of l and s, l', s' and α are given by τ_Q, Eq. (1). If $\tau_Q(ls) = (ls0)$, then $H^{l\mathfrak{T}\mathbf{h}j}_{(ls),\,(l's'\alpha)} = \pi\hbar/2\Delta$. This corresponds to no change in the configurations of $\mathfrak{L}$, $\mathfrak{T}$, and $\mathbf{h}$.

For H_3, one has

$$H_3 = \sum_{k=0}^{J} H^{\mathbf{j}k}_{+1} \otimes P^{\mathfrak{R}}_k \tag{33}$$

where $H^{\mathbf{j}k}_{+1}$ is given by Eq. (16) with configurations f and g referring to positions k and $k+1$ (mod $J+1$) of the head $\mathbf{j}$.

The above choices for H_1, H_2, and H_3 satisfy Eqs. (20), (22), and (25) when they are substituted into Eq. (28). This follows from the pairwise orthogonality of the terms in the l, s, j, and k sums.

At times, $t = (3m+h)\Delta + \delta$ which are not integral multiples of Δ, $\exp[-iHt]$ can be written as $\exp[-iH_{h+1}\delta]\exp[-iH(3m+h)\Delta]$ where $h = 0$, 1, or 2. The action of $\exp[-iH(3m+h)\Delta]$ is given by Eq. (29). $\exp[-iH_1\delta]$ is given by Eq. (20) with $U^{\mathfrak{R}k}_{(lsj)b}$ replaced by $\exp[-iH^{\mathfrak{R}k}_{(lsj)b}\delta]$. This in turn is given by Eqs. (17) and (18) where f and g refer to configurations (lsj) and b in cell k of $\mathfrak{R}$.

For $h = 1$ $\exp[-iH_2\delta]$ is given by Eq. (22) with $U^{\mathfrak{L}\mathfrak{T}\mathbf{h}j}_{(ls),(l's'\alpha)}$ replaced by

$$\exp\left[-iH^{\mathfrak{L}\mathfrak{T}\mathbf{h}j}_{(ls),\,(l's'\alpha)}\delta\right] = \cos\left(\frac{\pi\delta}{2\Delta}\right) - i\left(\sigma^{\mathfrak{L}}_{ll'} \otimes \sigma^{\mathfrak{T}j}_{ss'} \otimes \sigma^{\mathbf{h}j}_{\alpha}\right)\sin\left(\frac{\pi\delta}{2\Delta}\right) \tag{34}$$

Here Eq. (18) has been used. For $h = 3$, $\exp[-iH_3\delta]$ is given by Eq. (25) with $U^{\mathbf{j}k}_{+1}$ replaced by $\exp[-iH^{\mathbf{j}k}_{+1}\delta]$. This can also be put into the above form by use of Eq. (18).

The Hamiltonian given above is fairly simple, which makes it straightforward to understand how it works. However, it has the disadvantage that it is time dependent. This requires the use of some external agency to change the overall system Hamiltonian according to the prescription given by Eq. (27).

It is more desirable to construct a model of the process in which the Hamiltonian is time independent. Such a model has the advantange that the evolution is truly isolated and does not require the intervention of an external agency.

5. TIME-INDEPENDENT HAMILTONIANS

The method used here to construct a model with a time-independent Hamiltonian is based on the following observation. In the construction

given, the origin of the time dependence is in the turning on of the successive step-type Hamiltonians so that the operations V_1, V_2, and V_3 could be carried out in succession, each in an interval Δ. However, there is no reason why the model evolution cannot be speeded up so that in a time interval Δ, the operation V corresponding to all three steps, record, then compute, then shift, is carried out. That is, it is desired to construct a unitary operator V such that for each machine description $(l\gamma jk\phi)$

$$V\Psi_{l\gamma jk\phi} = \Psi_{l'\gamma' j' k' \phi'} \tag{35}$$

where $\Psi_{l\gamma jk\phi}$ and $\Psi_{l'\gamma' j' k' \phi'}$ are related by

$$V_3 V_2 V_1 \Psi_{l\gamma jk\phi} = i\Psi_{l'\gamma' j' k' \phi'} \tag{36}$$

The relationship between the primed quantities and unprimed quantities and the overall phase factor in Eq. (36) can be constructed from Eqs. (21), (24), and (26). Details will not be given here and are left to the reader. Note though that if $l\gamma jk\phi$ describes a (standard) Turing machine at the end of n steps of a standard computation ($n < J$), then $(l'\gamma' j' k' \phi')$ describes the machine at the end of $n + 1$ steps. Recall that there are many configurations $(l\gamma jk\phi)$ which do not correspond to any state of a standard Turing machine computation. Also by the construction given in Section 2 there are configurations f which do not represent any $(l\gamma jk\phi)$. An example is a configuration with more than one spin up in the lattice region associated with the record head **j**. V may or may not be the identity on these configuration states; which it is depends on the configuration.

Consider some state $\Psi_{1\gamma 00\underline{b}}$ which represents the initial state of a standard computation (Section 3). Let N_γ be some number defined by $(V)^{N_\gamma}\Psi_{1\gamma 00\underline{b}} = \Psi_{1\gamma 00\underline{b}}$. Such a number exists because V is unitary and the set of configurations on the lattice is finite. Define the orbit of V at $\Psi_{1\gamma 00\underline{b}}$ by the set of states $\{V^n \Psi_{1\gamma 00\underline{b}} | n = 0, 1, \ldots, N_\gamma - 1\}$.

For each γ, there is an orbit of length N_γ. Since V is unitary no two orbits have any states in common. Note that $N_\gamma > J$ for each γ. As a result, up to J iterations of V on $\Psi_{1\gamma 00\underline{b}}$ correspond to the carrying out of up to J standard Turing machine computation steps. Continued iteration on V up to $N_\gamma - 1$ times destroys the representation as the resulting states do not correspond to stages in the computation. However, the effect of the continued iteration is to erase the record and undo the computation, since the initial state is recovered.

Besides the above orbits, there are many other nontrivial ones. Any lattice configuration f such that Ψ_f is not in the orbits already generated can be used to generate a new orbit. By this process, one can exhaust the configuration states on the lattice and find all the orbits of V.

The above is easily translated into standard Hilbert space language.

The lattice Hilbert space $\mathcal{H}$, spanned by all the configuration states, can be decomposed into a set of closed subspaces which are invariant and irreducible under V and are in one–one correspondence with the orbits. Each subspace is spanned by the states in the corresponding orbit. In particular, for each standard tape expression γ, there is such a subspace $\mathcal{H}_\gamma$ spanned by $\{V^n\Psi_{1\gamma 00\underline{b}} | n = 0, 1 \ldots N_\gamma - 1\}$.

Let $\mathcal{H}_1, \mathcal{H}_2, \ldots, \mathcal{H}_N$ be a listing of all the V invariant, irreducible subspaces with $P_1, \ldots, P_N$, the corresponding projection operators on the subspaces. Then V can be decomposed as

$$V = \sum_{j=1}^{N} V_j P_j$$

where $V_j\mathcal{H}_j = \mathcal{H}_j = P_j\mathcal{H}$ and $[V_j, P_j] = 0$. Here, one is interested in the subspaces H_γ with their corresponding projection operators P_γ and the restrictions V_γ of V to the $\mathcal{H}_\gamma$. To this end, another operator W is defined by

$$W = \sum_{\gamma} V_\gamma P_\gamma + \left(1 - \sum_{\gamma} P_\gamma\right) \tag{37}$$

where the sum is over all possible initial tape expressions for a standard computation. W is a unitary operator which is identical with V on the spaces $\mathcal{H}_\gamma$ and the identity elsewhere.

The goal is to construct a Hamiltonian H so that

$$W = e^{-i\Delta H} \tag{38}$$

To this end, it is convenient to construct for each γ the eigenvalues and eigenvectors of V_γ. (The method given is general for finite-dimensional spaces and applies to any V_j in the decomposition of V.)

Fix γ and consider the configuration states $\Psi_0^\gamma, \Psi_1^\gamma, \ldots \Psi_{N_\gamma - 1}^\gamma$ in $\mathcal{H}_\gamma$ defined by $\Psi_n^\gamma = V^n\Psi_{1\gamma 00\underline{b}}$ for $n = 0, 1, \ldots, N_\gamma - 1$. For $n \leqslant J$ Ψ_n is the state representing the Turing machine after n steps of the computation. The above gives

$$V_\gamma \Psi_n^\gamma = \Psi_{n+1}^\gamma \tag{39}$$

with the understanding that if $n = N_\gamma - 1$, $n + 1 = 0$.

The above shows that V_γ is the bilateral shift operator on $\mathcal{H}_\gamma$. Since $\mathcal{H}_\gamma$ is finite dimensional V_γ is pure discrete. The eigenvalues and eigenvectors of such operators are known. The eigenvalues of V_γ are the N_γth roots of unity $\alpha_0, \alpha_1, \ldots, \alpha_{N_\gamma - 1}$ given by

$$\alpha_l = \exp\left[-\frac{2\pi i l}{N_\gamma}\right] \tag{40}$$

The eigenvectors $\Phi_0^\gamma, \ldots, \Phi_{N_\gamma - 1}^\gamma$ are given by

$$\Phi_l^\gamma = \frac{1}{\sqrt{N_\gamma}} \sum_{j=0}^{N_\gamma - 1} (\alpha_l)^{-j+1} \Psi_j^\gamma \tag{41}$$

One clearly has $V_\gamma \Phi_l^\gamma = \alpha_l \Phi_l^\gamma$. From the above it is seen that the Hamiltonian H must satisfy Eqs. (37) and (38). To this end one requires that H satisfy

$$H = \sum_\gamma H_\gamma \tag{42}$$

where for each standard tape expression γ, H_γ operates only in $\mathcal{H}_\gamma$ and is zero elsewhere, and $V_\gamma = \exp(-iH_\gamma \Delta)$. Then for all times t, $W(t) = \exp[-iHt]$ satisfies

$$W(t) = \sum_\gamma e^{-iH_\gamma t} P_\gamma + 1\left(1 - \sum_\gamma P_\gamma\right) \tag{43}$$

since $H_\gamma H_{\gamma'} = 0$ for all $\gamma \neq \gamma'$.

There are many choices of the Hamiltonian H_γ which satisfy $\exp[-i\Delta H_\gamma] = V_\gamma$. Here H_γ will be taken to be given by ($\hbar = 1$)

$$H_\gamma = \sum_{l=0}^{N_\gamma - 1} \frac{2\pi}{\Delta} \frac{l}{N_\gamma} Q_l^\gamma \tag{44}$$

where Q_l^γ is the projection operator for the eigenvector Φ_l^γ. Note that the choice $H_\gamma = \sum_{l=0}^{N_\gamma - 1} (2\pi/\Delta)(l/N_\gamma + n_l^\gamma) Q_l^\gamma$ where n_l^γ is an arbitrary l-dependent integer also satisfies Eqs. (37) and (38). Equation (44) is the simplest of these possibilities in that $n_l^\gamma = 0$ for all l and γ.

Let $V_\gamma(t) = \exp(-iH_\gamma t)$ be the time shift operator for the Hamiltonian H_γ as defined by Eq. (44). Then one has

$$V_\gamma(t) = \sum_{l=0}^{N_\gamma - 1} \exp \frac{-2\pi i l t}{N_\gamma \Delta} Q_l^\gamma \tag{45}$$

The time shift operator $V_\gamma(t)$ and the Hamiltonian H_γ are given in Eqs. (44) and (45) in terms of their eigenprojectors Q_l^γ. Since these are difficult to directly visualize it is worthwhile to express $V_\gamma(t)$ and H_γ directly in terms of spin configuration exchange operators and projection operators on spin configuration states. This can be done by use of Eqs. (40) and (41) to express Q_l^γ as

$$Q_l^\gamma = \frac{1}{N_\gamma} \sum_{\substack{j=0 \\ k=0}}^{N_\gamma - 1} \exp\left[-\frac{2\pi i l}{N_\gamma}(j-k)\right] \Psi_j^\gamma)(\Psi_k^\gamma \tag{46}$$

Now one has

$$\Psi_j^\gamma)(\Psi_k^\gamma = \sigma_{jk} P_k^\gamma \tag{47}$$

where σ_{jk}, which is the operator which exchanges configurations j and k, is given by Eq. (14). Note that configurations j and k which correspond to the overall system state at the respective jth and kth step, are defined over the whole lattice region in Fig. 1. P_k^γ is the projection operator on the configuration state Ψ_k^γ.

Use of Eqs. (46) and (47) in Eqs. (44) gives an alternate expression for the Hamiltonian

$$H_\gamma = \sum_{j,k=0}^{N_\alpha - 1} d_{jk} \sigma_{jk}^\gamma P_k^\gamma \tag{48}$$

where the coefficient d_{jk} is given by

$$d_{jk} = \sum_{l=0}^{N_\gamma - 1} \frac{2\pi l}{\Delta N_\gamma^2} \exp \frac{2\pi i l(j-k)}{N_\gamma} \tag{49}$$

The time shift operator, Eq. (45) becomes

$$V_\gamma(t) = \sum_{j,k=0}^{N_\gamma - 1} b_{jk}(t) \sigma_{jk}^\gamma P_k^\gamma \tag{50}$$

where the coefficient $b_{jk}(t)$ is given by

$$b_{jk}(t) = \frac{1}{N_\gamma} \sum_{l=0}^{N_\gamma - 1} \exp - \frac{2\pi i l}{N_\gamma} \left(\frac{t}{\Delta} + k - j \right) \tag{51}$$

It is clear that the system Hamiltonian given by Eq. (42) and Eqs. (44) or (48) has the desired properties. It is time independent. Also the evolution of $\Psi(t) = \exp(-iHt)\Psi_{1\gamma 00\underline{b}}$ is such that at time $t = n\Delta$ $\Psi(n\Delta) = \Psi_n^\gamma$, which is the configuration state corresponding to the nth step of the process. (Recall that $\Psi_{1\gamma 00\underline{b}} = \Psi_0^\gamma$.) This follows from the fact that, by Eq. (51) $b_{jk}(n\Delta) = 0$ unless $n \bmod N_\gamma + k - j = 0$ or N_γ. For both of these cases $b_{jk}(n\Delta) = 1$. For $t = N_\gamma \Delta$, the recurrence cycle time for the lattice system in any state in $\mathcal{H}_\gamma$ one has $W(N_\gamma \Delta)\Psi_n^\gamma = \Psi_n^\gamma$ for each $n < N_\gamma$. Thus if one starts at $t = 0$ with the initial state $\Psi_{1\gamma 00\underline{b}}$ one arrives at time $N_\gamma \Delta$ back at the initial state.

The effect of $W(t)$ on configuration states at times which are not integral multiples of Δ can be easily obtained from Eq. (50). In particular, let $t = n\Delta + \delta$ where $0 \leqslant \delta \leqslant 1$. Then $W(t)$ [or equivalently $V_\gamma(t)$] acting on $\Psi_{1\gamma 00\underline{b}}$ gives

$$\Psi(n\Delta + \delta) = \sum_{m=0}^{N_\gamma - 1} b_{m-n}(\delta) \Psi_m^\gamma \tag{52}$$

Here use was made of the fact that from Eq. (51) one sees that $b_{m0}(n\Delta + \delta) = b_{mn}(\delta) = b_{m-n}(\delta)$.

6. CHARACTERISTICS OF THE MODELS

6.1. Representation of Computations

It is of value to discuss some aspects of the operation of the time-independent and time-dependent models in more detail. The two models are constructed so that three steps of the time-dependent model correspond to one step of the time-independent model. In particular, the state Ψ_n^γ which is arrived at in a time interval of $n\Delta$ in the time-independent model starting from Ψ_0^γ at time 0, is arrived at in the time-dependent model after an interval of $3n\Delta$. Furthermore this holds for all n.

Both models describe the first J steps of any Turing machine computation. If a computation halts in m steps where m is less than J, both models repeatedly record in cells $m+1, m+2, \ldots, J$ of the record, the same final triple of the state of $\mathfrak{L}$, the contents of the cell of $\mathfrak{T}$ scanned by **h**, and the position of **h** as is recorded in cell m. The state of $\mathfrak{L}$, $\mathfrak{T}$, and **h** is stationary at all times $n\Delta$ where $m \leqslant n \leqslant J$ (time independent) or $3m \leqslant n \leqslant 3J$ (time dependent) as it must be for a completed computation.

For times $n\Delta$ with $J < n < N_\gamma$ for the time-independent model the representation is destroyed as the model configuration states do not correspond in general to any stages in the computation. However, it is clear that the record is erased and the computation is undone since one arrives at time $N_\gamma\Delta$ at the initial state. This phase will be referred to in this work as the reversal phase. If desired, this reversal phase can be made to occur as the exact inverse of the forward compute phase as is done in the models constructed by Bennett.[2] The corresponding quantum mechanical models, which are more complex than the ones discussed here, are described elsewhere.[15] This discussion also applies to the time-dependent model except that one has $3J < n < 3N_\gamma$.

6.2. Time Locality

The behavior of the models at times which are not integral multiples of Δ is also of some interest. To put this in perspective consider any system which evolves from state n at time $n\Delta$ to state $n+1$ at time $(n+1)\Delta$. In general, at times $t = n\Delta + \delta$ with $0 \leqslant \delta \leqslant \Delta$, one would expect to find the system in the state n with a finite time-dependent probability $P_n(\delta)$ and in the state $n+1$ with a probability $P_{n+1}(\delta)$. Furthermore one would expect

that as δ increased from 0 to Δ $P_n(\Delta)$ decreased from 1 to 0 and $P_{n+1}(\delta)$ increased from 0 to 1 with $P_n(\delta) + P_{n+1}(\delta) = 1$ for all t where $n\Delta \leqslant t \leqslant (n+1)\Delta$. As a result one would not expect to find the system in a state which it had passed through several steps in the past or which it would arrive at several steps in the future.

This can be taken over directly in quantum mechanics. In particular a system will be called *time local* if for each n and for all times $t = n\Delta + \delta$ with $0 \leqslant \delta \leqslant 1$, the overall system state $\Psi(t)$ is a linear combination of the states Ψ_n and Ψ_{n+1} only, that is

$$\Psi(n\Delta + \delta) = \alpha_n(\delta)\Psi_n + \beta_n(\delta)\Psi_{n+1} \tag{53}$$

Here Ψ_n and Ψ_{n+1} are orthonormal states which the system is in at time $n\Delta$ and $(n+1)\Delta$. $\alpha_n(\delta)$ and $\beta_n(\delta)$ are complex-valued coefficients such that $|\alpha_n(\delta)|^2 + |\beta_n(\delta)|^2 = 1$ and $\alpha_n(0) = \beta_n(\Delta) = 1$ and $\alpha_n(\Delta) = \beta_n(0) = 0$. In particular, $\Psi(t)$ contains no components Ψ_m for $m < n$ (which correspond to stages already reached in the past) or for $m > n + 1$ (which correspond to stages to be reached in the future).

It is also possible that the system evolution is such that for one or more values of n the system is not time local. That is, besides Ψ_n and Ψ_{n+1} appearing in the linear superposition, states Ψ_m may appear with $m \neq n$, $n+1$ with coefficients $\gamma_m(\delta) \neq 0$. How much of each component is present depends on the magnitude $|\gamma_m(\delta)|$.

If a system and its associated Hamiltonian are such that for each n and for at least some times $t = n\Delta + \delta$ with $0 < \delta < \Delta$ the state $\Psi(t)$ is a linear superposition over all possible states Ψ_m with nonzero coefficients $\gamma_{nm}(\delta)$ then the system shall be said to be *time global*. The reason for the nomenclature is that as the system evolves from state Ψ_n at time t_n to state Ψ_{n+1} the overall system state $\Psi(t)$ contains components which correspond to states reached at all stages in the past as well as those to be reached at all stages in the future.

These ideas can be applied to the models constructed here. One sees from Eq. (53) that the time-independent Hamiltonian models constructed here are all time global. In particular, the coefficients $b_{m-n}(\delta)$ can all be shown by continuity and differentiability arguments to be nonzero for most δ between 0 and Δ (by most δ is meant all δ except possibly for isolated points). This means that as the Turing machine model evolves from state Ψ_n^γ at time $n\Delta$ to state Ψ_{n+1}^γ at time $(n+1)\Delta$ the overall system is a linear superposition over states representing all stages in the first J steps of the computation as well as all states occurring in the remaining $N_\gamma - J$ stages. Speaking somewhat loosely one may say that the system state $\Psi(t)$ starts at time $n\Delta$ in a single configuration state Ψ_n^γ, then expands as t increases into a linear superposition over all past and future configurations states in the

orbit and then collapses at $t = (n + 1)\Delta$ back to a single configuration state Ψ^{γ}_{n+1}. (The meaning of collapse here is quite different from the concept of wave function collapse which appears in measurement theory.)

The time-dependent Hamiltonian models constructed here differ from the time-independent models in that they are time local. One sees from Section 4.2 that for each n and for all times $n\Delta + \delta$ with $0 \leqslant \delta \leqslant \Delta$, Eq. (53) is satisfied with $\alpha_n(\delta) = \cos(\pi\delta/2\Delta)$ and $\beta_n(\delta) = -i\sin(\pi\delta/2\Delta)$. Note that $\alpha_n(\delta)$ and $\beta_n(\delta)$ are independent of n. $\Psi_n = \Psi(n\Delta)$ and $\Psi_{n+1} = \Psi((n + 1)\Delta)$ are given by Eq. (29) where m and h satisfy $n = 3m + h$ with $h = 0$, 1, or 2. Thus for all time $n\Delta + \delta$ with $0 \leqslant \delta \leqslant \Delta$, the only states contributing to the overall system state $\Psi^{\gamma}(n\Delta + \delta)$ are the configuration states Ψ^{γ}_n and Ψ^{γ}_{n+1} which correspond to the just completed stage and the next stage of the model computation process.

The time locality of the time-dependent models considered here has another important consequence. Consider the evolution of a complex system spread out over a region of space. One intuitively expects that as the evolution proceeds, changes will occur in the states of the subsystems in one region with the states of systems in other regions remaining stationary. Then the changes will transfer to some other subsystems in another region and occur in the new region for some time with the states of the subsystems in the first region remaining stationary. When the transferrals occur and to which subsystems in which regions they occur, or whether or not the whole system changes at some point, depends on the details of the process.

The Turing machine computation process fits this description very well. For example, initially the kth record cell is blank and it remains blank while changes are occurring in other parts of the system. The state of the kth record cell as a subsystem changes only when a triple is recorded in it. Thereafter it remains stationary until it is erased many steps later on. Similar periods of change interspersed with periods of stationarity also exist for other parts of the system.

The point to note here is that because of the property of time locality possessed by the time-dependent Hamiltonian models constructed here, states of the various model subsystems have the above property. For example, the state of the model subsystem corresponding to the kth record cell being blank remains stationary for all times t from 0 to $3k\Delta$. For times between $3k\Delta$ and $(3k + 1)\Delta$ the state of the record cell changes from $\Psi^{\mathcal{R}_k}_b$ to $\Psi^{\mathcal{R}_k}_{lsj}$ for some appropriate lsj. The state remains stationary until $t = 3J\Delta$. For $t > 3J\Delta$ but $< 3N_{\gamma}\Delta$ the stationarity of the state depends on the details of the reversal phase. There is some time $q\Delta$ at which the kth record cell state returns to $\Psi^{\mathcal{R}_k}_b$ and remains there.

The situation is quite different for the time-independent Hamiltonian models constructed here because they are time global. In particular one

sees from Eq. (52) that for each n as the time increases from $n\Delta$ to $(n+1)\Delta$ the state of every model subsystem changes. If the configuration of the subsystem at step n is the same as at step $n+1$ then the model subsystem state is the same at times $n\Delta$ and $(n+1)\Delta$. However, the model subsystem state is different at intermediate times.[2]

The magnitude of the changes in the state of any subsystem as t increases from $n\Delta$ to $(n+1)\Delta$ depends mainly on the values of the coefficients $b_{m-n}(\delta)$ for all values of m for which the configuration of the subsystem in Ψ^{γ}_{m} is different from that in Ψ^{γ}_{n}. An important parameter is the time distance or number of steps from n to the step at which the subsystem configuration is changed. For example, the configuration of the kth record cell is changed during step number $k+1$ from b to an appropriate *lsj*. The next changes occur in the steps occurring after the Jth when the complex reversal is occurring. At some step number q the kth record cell is converted back to b and remains there at the conclusion of each of the remaining $N_{\gamma}-q$ steps.

The state of the kth record cell system at time $n\Delta+\delta$, as given by the density operator $\rho^{\Re_k}(n\Delta+\delta)$, fits the above description. In particular

$$\rho^{\Re_k}(n\Delta+\delta)=\left(\sum_{m=0}^{k}+\sum_{m=q}^{N_{\gamma}-1}\right)\left[|b_{m-n}(\delta)|^2\right]P_b^{\Re_k}+\sum_{m=k+1}^{J}|b_{m-n}(\delta)|^2P_{lsj}^{\Re_k}$$
$$+\sum_{m',\,m=J+1}^{q-1}b^{*}_{m'-n}(\delta)b_{m-n}(\delta)\mathrm{Tr}'\left[\Psi^{\gamma}_{m}\right)\left(\Psi^{\gamma}_{m'}\right] \tag{54}$$

Here *lsj* denotes the appropriate triple stored in the kth record cell and $P_b^{\Re_k}$ and $P_{lsj}^{\Re_k}$ are the respective projection operators on the states $\Psi_b^{\Re_k}$ and $\Psi_{lsj}^{\Re_k}$. P^{γ}_{m} is the projection operator on the overall system state Ψ^{γ}_{m}. The prime on the trace means that it is taken over all spin-1/2 systems except those comprising the kth record cell.

[2] The state of any lattice model subsystem X at time $n\Delta+\delta$ is given in general by a density operator $\rho^{X}(n\Delta+\delta)$, where by Eq. (52)

$$\rho^{X}(n\Delta+\delta)+\sum_{j}\sum_{m,\,m'}^{j}b_{m-m}(\delta)b^{*}_{m'-n}(\delta)\mathrm{Tr}_{-X}[\Psi^{\gamma}_{m})(\Psi^{\gamma}_{m'}]$$

The subscript $-X$ means that the trace is taken over all spin-1/2 systems not in X. The j sum is over all disjoint subsets of the N_{γ} configurations defined such that all configurations within each subset are identical outside of X, and any two configurations with each one from different subsets are different outside of X. The m, m' sum is over all pairs of configurations within the jth subset. If a subset j contains just one configuration h then the m, m' sum contribution to the density operator becomes $|b_{h-n}(\delta)|^2P^{X}_{h\,|\,X}$, where $P^{X}_{h\,|\,X}$ is the projection operator for the configuration state $\mathrm{J}^{X}_{h\,|\,X}$ on X and $h\,|\,X$ is the restriction of h to X.

From the above equation and the properties of the $b_{m-n}(\delta)$ coefficients one sees that if $\delta = 0$, $\rho^{\mathcal{R}_k}(n\Delta) = P_b^{\mathcal{R}_k}$ if $0 \leqslant n \leqslant k$ or $q \leqslant n \leqslant N_\gamma$ and $\rho^{\mathcal{R}_k}(n\Delta) = P_{lsj}^{\mathcal{R}_k}$ if $k < n \leqslant J$. For these times (integral multiples of Δ) the above quantum states are pure and correspond to $\Psi_b^{\mathcal{R}_k}$ and $\Psi_{lsj}^{\mathcal{R}_k}$. For times $n\Delta + \delta$ with $0 < \delta < \Delta$ $\rho^{\mathcal{R}_k}(n\Delta + \delta)$ is a mixture of the states $P_b^{\mathcal{R}_k}$ and $P_{lsj}^{\mathcal{R}_k}$ as well as other states which may be contributed by the right-hand term of Eq. (54). If $0 \leqslant n \leqslant k$ or $q \leqslant n < N_\gamma$, the dominant component in the mixture is $P_b^{\mathcal{R}_k}$. How much of the other components are present depends on δ, and on the distance $m - n$ for all $k + 1 < m < q - 1$, or how far away in steps changes in the record cell are from step n. A similar discussion holds also if $k + 1 \leqslant n \leqslant J$.

6.3. Complexity of the Hamiltonians

The two types of Hamiltonians considered here differ in complexity in an important way. In particular, the time-independent Hamiltonian models constructed here are more complex than the time-dependent Hamiltonian models. One reason is that because the three types of steps—record, compute, and shift—all occur in one step in the time-independent models, the lattice regions over which configuration changes must be implemented in each step are larger in the time-independent models than in the time-dependent models.

Another reason is that the construction of the time-independent Hamiltonians requires that one know in advance the details of all computations that can be done with the Turing machine in question. This can be seen by examination of Eqs. (39), (42), and (44) or (48). For instance, to construct the Hamiltonian given by Eq. (42), one must know for each standard initial expression γ of length $\leqslant J$ on the computation tape all instantaneous descriptions and the order in which they occur in the first J steps in the computation process. This is clearly necessary in order to construct the operators σ_{jk}^{γ} and P_k^{γ} appearing in Eq. (48). This requirement is also equivalent to solving the halting problem for the first J steps of any computation.

This is an undesirable state of affairs because it makes any such model useless from a practical standpoint. Clearly one wants to use any such model to generate new information by carrying out calculations and not just to repeat old information which one has had to obtain previously in order to construct the model. In particular the Hamiltonian must be sufficiently "simple" so that its construction does not require a complete solution of the problems for which the model would be used.

The time-dependent Hamiltonian models constructed here are satisfac-

tory in this respect. The construction of the record and shift Hamiltonians H_1 and H_3, Eqs. (30) and (33) does not require any knowledge of Turing machine computations. The construction of the compute Hamiltonian H_2, Eq. (31), for a given Turing machine requires that one know the function τ_Q, Eq. (1), or equivalently have available all quintuples in the set Q.

This type of knowledge is what one needs to construct any digital computer and the input programs. It is thus consistent with the practical use of such a model to make calculations. Construction of such a Hamiltonian does not require knowledge of all possible computation orbits of the model.

In this work, two types of models have been constructed. In one, the evolution is time local and the Hamiltonians are less complex but are time dependent. In the other type the evolution is time global and the Hamiltonians are more complex but are time independent.

The question arises whether these results can be generalized. In particular, must all time global Hamiltonian models of the computation process be sufficiently complex that the construction of the Hamiltonian requires prior knowledge of all computation orbits? It is speculated that the answer is "yes." An equally important question is "must all time-independent Hamiltonian models be time global?"

To answer this one notes that a theorem can be proven which says that any Hamiltonian model for which $V(\Delta) = \exp[-i\Delta H]$ takes states Ψ_n^γ corresponding to the completed nth computation step into states Ψ_{n+1}^γ corresponding to the completed $n+1$st step in a time interval Δ for all $n = 0, 1 \ldots N_\gamma - 1$ and all standard initial tape expressions γ (i.e., a bilateral shift) as in Eqs. (38)–(43), must be time global. However, it does not follow from this theorem that all time-independent Hamiltonian models of the computation process are time global. For example, the models constructed elsewhere[13] are time independent and time local. The reason these models escape the restrictions of the theorem results from the fact that they can be regarded as appropriate limits of sequences of models similar to those constructed here. Although the theorem holds for each model in the sequences it does not hold for the limit models. It is hoped to give further details elsewhere.

6.4. Energy Dissipation

Another property of the time-independent Hamiltonian models constructed here is that they dissipate no energy as they evolve. That is, not only is the total energy a constant of the motion as it must be for any Hamiltonian evolution but there is no state degradation. In particular at each time $n\Delta$ for $n = 0, 1, \ldots, N_\gamma - 1$ the overall system state corresponds

to a single spin configuration on the lattice. It is not a linear combination $\sum_{m=0}^{N_\gamma-1} c_m(n\Delta)\Psi_m^\gamma$ of lattice spin configuration states where the coefficients $c_m(n\Delta)$ are such that $c_m(n\Delta) \neq 0$ for values of $m \neq n$ and the amplitude $|c_n(n\Delta)|$ for finding Ψ_n^γ in $\Psi^\gamma(n\Delta)$ decreases as n increases. This would be the case if the overall system state were degrading as it evolved.

For the time-dependent models energy dissipation may well occur in the external agency which turns the successive step Hamiltonians on and off. Energy changes occur within the model only when a step in which the Hamiltonian, although active, makes no changes in the model state follows or is followed by a step in which the active Hamiltonian changes the model system state. This occurs because by Eq. (18) if $\Psi((3m + h)\Delta) = \Psi((3m + h + 1)\Delta)$ then $(\Psi(t), H_{h+1}\Psi(t)) = \pi\hbar/2\Delta$ for all $t = (3m + h)\Delta + \delta$ with $0 < \delta < \Delta$. However, if $\Psi((3m + h)\Delta) \neq \Psi((m + h + 1)\Delta)$ then $(\Psi(t), H_{h+1}\Psi(t)) = 0$. An example of this occurs if a computation halts in $L < J$ steps, then for all times t with $3L\Delta < t < 3J\Delta$, H_1 and H_3 when active change the model system state but H_2, when active makes no changes.

Another aspect of the time-independent Hamiltonian models constructed here is that they operate essentially at the quantum limit. That is the system energy uncertainty δE divided by the computation speed $1/\Delta$ is close to the limit given by the time energy uncertainty principle.[18] To see this, one notes that the spread in energy eigenvalues, given by $(2\pi\hbar/\Delta)(1 - 1/N_\gamma) \approx 2\pi\hbar/\Delta$ is an upper limit for δE. The uncertainty principle gives a lower limit as $\delta E \gtrsim \hbar/\Delta$.

Also for the models constructed here limits on the computation speed which arise from the presence of energy dissipation[1] are not applicable. One can in principle at least increase the computation speed by increasing the average system energy $< H\gamma >$ without introducing state degradation and energy dissipation. One sees this from Eqs. (48) and (49), which give for any lattice configuration f

$$< H_\gamma > = (\Psi_f^\gamma, H_\gamma \Psi_f^\gamma) = d_{ff} = \frac{\pi\hbar}{\Delta}\left(1 - \frac{1}{N_\gamma}\right) \approx \frac{\pi\hbar}{\Delta} \tag{55}$$

This shows that the energy is independent of f and is given by $\pi\hbar/\Delta$ if N_γ is large. The result follows from the fact that Δ is the time it takes the model to carry out one computation step (and is also roughly the lifetime of the system in each of the states Ψ_n^γ), so $1/\Delta$, the computation speed, $\simeq \langle H_\gamma \rangle / \pi\hbar$.

These results support Deutsch[5] and Landauer[19] in their criticism of Bekenstein[1] in that the computation speed can be made arbitrarily high by a sufficient increase of the average energy of the Turing machine model system. For example, if the average energy is 1 eV (electron volt), then the computation speed is 4×10^{14} steps/sec. If the energy is 1 erg, then the

computation speed is about 3×10^{26} steps/sec, a value well above Bekenstein's[1] limit of 10^{15} steps/sec.

7. MEASUREMENTS

In order to use any of the models constructed here for actual calculations, one must be able to make measurements of the parameters which are relevant to the Turing machine model representation. For example, one may want to carry out a complete measurement to determine the l, λ, j, k, ϕ values. Or one may want to determine if the computation has halted, and if so, what the final tape expression is. Since important aspects of these measurements are different for the time-independent and time-dependent Hamiltonian models constructed here, those models will be considered separately.

7.1. The Time-Independent Models

Consider first a complete system measurement of the Turing machine plus record system parameters in a time-independent Hamiltonian model as constructed here. The possible outcomes of such a measurement include all values of l, λ, j, k, ϕ which describe the first J steps of the calculation as well as all other spin configurations reached in returning to the initial state.

Let such a measurement take place over a time interval of width δ centered on $n\Delta$ for some $n < J$ and let $l\lambda jn\phi$ denote the state of the Turing machine plus record system after n steps. That is $\Psi^{\gamma}_{l\lambda jn\phi} = \Psi^{\gamma}(n\Delta)$. Then one can show that if $\delta \ll \Delta$ then one obtains with high probability, $1 - c\delta^2/\Delta^2$, the outcome $l\lambda jn\phi$. (c is a constant of order unity.) The probability of obtaining any other outcome is $c\delta^2/\Delta^2$.

Furthermore, any such measurement necessarily perturbs the system state. The reason is that the model system is in an exact eigenstate of any of the observables of interest here [i.e., in the form given by Eq. (9)] only at the instants $t = n\Delta$. At other times the state $\Psi^{\gamma}(t)$ is a linear superposition over all N^{γ} configuration states in the computation orbit and is not an eigenstate of any of the observables of interest here.

In particular one can show[3] that at the end of the measurement

[3] Let $\{\phi_f \mid f = 0, 1 \ldots N_{\gamma} - 1\}$ and ϕ_b denote orthonormal states of a measurement apparatus where ϕ_b denotes the initial apparatus state. Let the measurement interaction Hamiltonian be given by $H' = (\pi\hbar/2\delta)[\sum_f (P_f \otimes \sigma_f) - 1]$, where P_f is the projection operator for finding the model system in state Ψ^{γ}_f and σ_f exchanges the states ϕ_f and ϕ_b and is the identity elsewhere. The measurement is described by the Hamiltonian H', which is turned on at time $n\Delta - \delta/2$ and off at $n\Delta + \delta/2$, plus the Hamiltonian H_{γ} [Eq. (48)] which is active at all times. The model plus apparatus system state at time $n\Delta + \delta/2$ is given by $\sum_f \theta^{\gamma}_f \otimes \phi_f + \theta^{\gamma}_b \otimes \phi_b$, where θ^{γ}_f and θ^{γ}_b have the properties described in the text.

interaction at time $n\Delta + \delta/2$ the model system state which corresponds to the outcome $l\lambda jn\phi$ is given by $\theta^{\gamma}_{l\lambda jn\phi}$, where $|\theta^{\gamma}_{l\lambda jn\phi}|^2 = 1 - c\delta^2/\Delta^2$ and

$$\theta^{\gamma}_{l\lambda jn\phi} = \Phi + \Psi^{\gamma}(n\Delta + \delta/2)(\Psi^{\gamma}(n\Delta + \delta/2), \theta^{\gamma}_{l\lambda jn\phi})$$

Φ is an (unnormalized) state which is orthogonal to $\Psi^{\gamma}(n\Delta + \delta/2)$ and whose leading term, in powers of δ/Δ, is of order δ^2/Δ^2. Since $\Psi^{\gamma}(n\Delta + \delta/2)$ is the state of the model system in the absence of any measurement, Φ represents the perturbation by the measurement associated with outcome $l\lambda jn\phi$.

The model system state θ^{γ}_{f} associated with any other outcome f is such that $|\theta^{\gamma}_{f}|^2 < c\delta^2/\Delta^2$. It represents a large perturbation by the measurement on the model system state and cannot be considered to represent any stage in the Turing machine computation.

The above shows that if a complete system measurement is to yield the correct outcome with probability close to unity and not perturb the system state appreciably its duration δ must be smaller than the time interval Δ between computation steps. This can be quite difficult especially if Δ is short.

Another consequence of the fact that any complete measurement of finite duration δ necessarily perturbs the system state is that such a measurement also necessarily introduces energy dissipation into the system. This can be seen by the fact that the state of the model system at the end of the measurement is no longer a pure state but is a mixed state with finite entropy. (The state is obtained as a trace, over the apparatus degrees of freedom, of the model system and apparatus state at the conclusion of the measurement.)[2] The energy dissipation, which can be defined as the initial system energy minus the energy associated with the desired final state $\theta^{\gamma}_{l\lambda jn\phi}$, is given by $\pi\hbar/\Delta - (\theta^{\gamma}_{l\lambda jn\phi}, H_{\gamma}\theta^{\gamma}_{l\lambda jn\phi})$, which equals $(K\delta^2/\Delta^2)(\pi\hbar/\Delta)$ where K is a constant of order unity.

One sees then that to avoid appreciable perturbation and energy dissipation by a complete measurement, the duration δ of the measurement must be short with $\delta \ll \Delta$. However, as was seen this may be difficult to achieve. Clearly it would be desirable not to have to restrict measurements to such short durations.

Unlike the complete system measurements one can determine whether or not the computation has halted by examination of a subsystem only. It is sufficient to examine two adjacent record cells. If the contents of the kth and $k + 1$st cell examined at any time after $(k + 2)\Delta$ but before $J\Delta$ are the same, the computation halted by the kth step. If the contents are different, the computation did not halt by the kth step.

One can apply the analysis just discussed to such halting measurements of the kth and $k + 1$st record cell subsystems. The results show that

such measurements also necessarily perturb the system and introduce energy dissipation even if they are restricted to intervals $\delta < \Delta$. However, as a practical matter the amount of change in the state of the kth and $k + 1$st record cells may be sufficiently small so that the measurement can be extended over several Δ intervals without much perturbation and dissipation. The actual amount can be determined from an equation very similar to Eq. (54) for n such that $J > n > k$ and depends on n, k, and system parameters.

7.2. Time-Dependent Models

Complete system measurements on the time-dependent Hamiltonian models constructed here also perturb the model state and introduce energy dissipation even if the duration is less than Δ. However, halting measurements which examine the kth and $k + 1$st record cells at times t where $3(k + 1)\Delta < t < 3J\Delta$ do not perturb the system state or introduce energy dissipation. The reason is that because these models are time local the states of subsystems are strictly stationary during all times for which the corresponding steps are not changing the configurations of the subsystem in question. This can be seen from Eq. (53), which shows that the configuration state of any subsystem which is the same in Ψ_n^γ as in Ψ_{n+1}^γ remains stationary over all times between $n\Delta$ and $(n + 1)\Delta$. This means that measurements such as the halting measurements need not be restricted to durations less than Δ and can extend over many intervals Δ.

8. DISCUSSION

It is good to briefly review the advantages and disadvantages of the models constructed here. The time-independent Hamiltonian models have the advantanges that they are time independent—no external agent is required. Also they do not dissipate energy or degrade the system state, and they operate at the quantum limit in that the system energy uncertainty divided by the computation speed $< 2\pi\hbar$. They have the disadvantage that they are very sensitive to external influences. Also the Hamiltonian is quite complex in that its construction requires prior knowledge of all J step computation orbits of the Turing machine under consideration. Finally the evolution is time global. This has the result that all measurements—even those confined to subsystems—perturb the model system state and introduce energy dissipation. This occurs even if the duration of the measurements is less than the computation step time—a requirement which must be met for any system measurement.

The time-dependent models have the advantage that there is no state degradation. Also, the Hamiltonians are less complex. In particular for each Turing machine, construction of the Hamiltonian requires knowledge of the function τ_Q, Eq. (1). One need not know the J step computation orbits. Finally the evolution is time local. This has the consequence that measurements, restricted to appropriate subsystems, such as halting measurements, need not perturb the system or introduce energy dissipation. Also they can extend over several computation step intervals. The models have the disadvantage that they are very sensitive to external influences. Also an external agent is required to turn on and off the record, compute, and shift step Hamiltonians.

In the models constructed elsewhere,[13-15] the external agent which carried out this turning on and off is provided by successive scatterings of a moving system from a sequence of fixed scatters[14,15] or a sequence of moving systems scattering in succession off a fixed scatterer.[13] Model parameters were fixed so that one system scattering from one fixed scatterer carried out one step in the model computation. In these models the Hamiltonians were time independent and, under the approximation used in one of the papers,[13] were time local and did not dissipate energy or degrade the system state. The models discussed in other papers[14,15] showed state degradation and energy dissipation even if no measurements are carried out. The models are also time local to the extent that various approximation made can be considered to be exact. Furthermore, the Hamiltonians in these models can be less complex in that the Turing machine model Hamiltonians used are essentially those used for the time-dependent models constructed here. Finally one notes the successive scattering models have the advantage that the representation is not destroyed after J computation steps have occurred. In particular, the state of the Turing machine plus record system becomes permanently stationary after J computation steps so that measurements can take as long as is desired.

In conclusion, one notes that the mathematical existence of nondissipative quantum mechanical Hamiltonian models of a finite number of steps of any Turing machine computation has been shown here. However, it remains an open question whether such models can actually be constructed in the laboratory. For example, if one accepts as true the statement that at most a countable infinity of Hamiltonian models can actually be constructed in the laboratory, then most Hamiltonians which exist mathematically are not physically constructible.

On the other hand, the existence of such Hamiltonians means that one must be cautious about assuming that the computation process must dissipate energy and cannot be carried out by models operating close to the

quantum limit. The results of this paper show that if one wants to argue that such models of the computation process cannot be physically constructed in principle, the argument cannot be based on the nonexistence of Hamiltonian models. Some other principle must be invoked.

REFERENCES

1. Jacob D. Bekenstein, *Phys. Rev. Lett.* **46**:623 (1981).
2. Charles H. Bennett, *IBM J. Res. Dev.* **17**:525 (1973).
3. Hans J. Bremerman, "Part I. Limitations on Data Processing Arising From Quantum Theory," in *Self-Organizing Systems,* M. C. Yovits, G. T. Jacobi, and G. D. Goldstein, eds. (Spartan Books, Washington, D.C., 1967).
4. Hans J. Bremerman, "Energy and Entropy Costs in Information Transfer and Computing," Preprint to appear in Proceedings of Conference on Physics of Computation, MIT, May 6–8, 1981.
5. David Deutsch, *Phys. Rev. Lett.* **48**:286 (1982).
6. Robert W. Keyes, *Proc. IEEE* **69**:267 (1981); Rolf Landauer, "Fundamental Physical Limitations of the Computational Process," Preprint, 1981; *Ber. Bunsenges. Phys. Chem.* **80**:1048 (1976).
7. Rolf Landauer, *IBM J. Res. Dev.* **5**:183 (1961); Rolf Landauer and James W. F. Woo, *J. Appl. Phys.* 42:2301 (1971); Robert W. Keyes and Rolf Landauer, *IBM J. Res. Dev.* **14**:152 (1970).
8. K. K. Likharev, "Classical and Quantum Limitations on Energy Consumption in Computation," to appear in *Proceedings of Conference on Physics of Computation*, MIT, May 6–8, 1981.
9. L. B. Levitin, "Physical Limitations of Rate, Depth, and Minimum Energy in Information Processing," Preprint, Bielefeld University.
10. D. Mundici, *Nuovo Cimento* **61B**:297 (1981).
11. Rolf Landauer, Preprint, to appear in Proceedings of Conference on Physics of Computation, MIT, May 6–8, 1981 *Int. J. Theor. Phys.* **21**:283 (1982).
12. Edward Fredkin and Tommaso Toffoli, "Conservative Logic," Tech. Memo., MIT/LCS/TM-197, April 29, 1981.
13. Paul Benioff, *J. Stat. Phys.* **22**:563 (1980).
14. Paul Benioff, *J. Math. Phys.* **22**:495 (1981).
15. Paul Benioff, "Quantum Mechanical Hamiltonian Models of Discrete Processes that Erase Their Own Histories: Application to Turing Machines," in Proceedings of Conference on Physics of Computation MIT, May 6–8, 1981.
16. Martin Davis, *Computability and Unsolvability* (McGraw Hill, New York, 1958).
17. Hartley Rogers, Jr., *Theory of Recursive Functions and Effective Computability* (McGraw Hill, New York, 1965), pp. 64, 65.
18. A. Messiah, *Quantum Mechanics,* Vol. I (John Wiley and Sons, New York, 1958), Chap. IV, Sec. 10.
19. Rolf Landauer, Informal Notes, 1981.