Journal of Biological Systems, Vol. 2, No. 2 (1994) 193–228

NEURAL NETWORKS TO COMPUTE MOLECULAR DYNAMICS

LARRY S. LIEBOVITCH[1], NIKITA D. ARNOLD[2] and LEV Y. SELECTOR[3]
[1] *Center for Complex Systems, Florida Atlantic University, Boca Raton, FL 33431, USA*
[2] *Angewandte Physik, J. Kepler University, A-4040 Linz, Austria*
[3] *Dept. of Ophthalmology, Columbia University, New York, NY 10032, USA*

Received 22 November 1993

ABSTRACT

Large molecules such as proteins have many of the properties of neural networks. Hence, neural networks may serve as a natural and thus efficient method to compute the time dependent changes of the structure in large molecules. We describe how to encode the spatial conformation and energy structure of a molecule in a neural network. The dynamics of the molecule can then be computed from the dynamics of the corresponding neural network. As a detailed example, we formulated a Hopfield network to compute the molecular dynamics of a small molecule, cyclohexane. We used this network to determine the distribution of times spent in the twist and chair conformational states as the cyclohexane thermally switches between these two states.

Keywords : Neural network, molecular dynamics, kinetics, protein motions, cyclohexane.

1. Introduction

In different scientific fields a new type of model has been useful in understanding how global properties of a complex system arise from the interplay of many local interactions. In psychology these models are called neural networks [1,19]. In computer science they are called parallel, distributed processing [34,35]. In physics they are called spin glasses [2,9]. A neural network, for example, consists of nodes and connections between them. At each time step the value of a node is updated. The new value depends on the values of the other nodes connected to it and the strength of their connections.

Large molecules, such as proteins, also consist of many interacting pieces. An ion channel protein, for example, has approximately a 1000 amino acid residues that interact by atomic bonds and electrostatic forces. Proteins have many of the characteristics of neural networks such as: frustration, energy landscapes with many local minima and ultrametricity [11,12,25,33,41,47,48]. In a protein, sidechains, regions and subunits can have multiple orientations in space that conflict with each other. There is no one structure that can satisfy all these conflicting constraints to uniquely minimize the energy. This property is called "frustration." The large number of different possible structures corresponds to an energy landscape with

many local minima. In changing from one conformational state to another, the protein must temporarily increase in energy to cross the hills in the energy landscape separating those states. That is, the protein structure needs to unfold slightly before it can refold in a different way. This property is called "ultrametricity."

Neural networks have been used as computational devices to predict the secondary structure of a protein (α-helix, β-sheet, etc.) from the primary sequence of amino acid residues [5,39]. Because of the common features between neural networks and proteins described above, neural networks have also been used as physical models of the structure and thermodynamical properties of proteins [11,12,41,48]. In this article we present a new application of these common features. We show how to construct neural networks with an energy structure similar to a given molecule, so that the dynamics of a molecule can be computed from the dynamics of the corresponding neural network. This approach leads to a new way of thinking about molecular dynamics. It may also lead to the development of faster algorithms to compute molecular motions.

Protein motions are important in how proteins function as structural units, how they catalyze chemical reactions and how they bind ligands [12,25]. Protein motions are now calculated by evaluating the force on each atom, updating its position and then repeating this procedure many times [33]. However, these time steps must be very small in order to accurately compute the new positions of the atoms. The 100 000 time steps needed to compute the motions of myoglobin over 10^{-10} s required 6 hours of supercomputer time [25]. Thus, this presently used approach is inadequate to study important motions within proteins that extend from nanoseconds to minutes.

Transforming a problem into a different but mathematically equivalent form can sometimes lead to a much more efficient computational algorithm. For example, recent box counting algorithms have reduced the computational time for fractal dimensions by three orders of magnitude [4,24,30]. Because neural networks and proteins share common features, the neural network is a natural representation, and thus may be a much more efficient method for the computation of molecular dynamics. Moreover, neural networks may achieve additional gains in computational speed because they are simple and intrinsically parallel structures that can take good advantage of computers with parallel architectures and custom integrated circuits.

In this article we explore the use of neural networks to compute molecular dynamics. First, we outline the properties of neural networks. Then we describe different ways that neural networks can be formulated to compute molecular dynamics. We then present a detailed formulation and analysis of the properties of one type of network, the Hopfield network, and show how it can be used to compute molecular dynamics. As an illustrative example, we used a Hopfield network to compute the dynamics of cyclohexane switching between twist and chair conformational states.

onformational state to another, the
ross the hills in the energy landscape
ucture needs to unfold slightly before
is called "ultrametricity."
itational devices to predict the sec-
t, etc.) from the primary sequence
ommon features between neural net-
works have also been used as physical
properties of proteins [11,12,41,48].
ese common features. We show how
ructure similar to a given molecule,
mputed from the dynamics of the
ads to a new way of thinking about
evelopment of faster algorithms to

teins function as structural units,
they bind ligands [12,25]. Protein
force on each atom, updating its
y times [33]. However, these time
compute the new positions of the
ute the motions of myoglobin over
e [25]. Thus, this presently used
tions within proteins that extend

t mathematically equivalent form
nputational algorithm. For exam-
the computational time for fractal
0]. Because neural networks and
work is a natural representation,
for the computation of molecular
eve additional gains in computa-
ically parallel structures that can
chitectures and custom integrated

tworks to compute molecular dy-
ral networks. Then we describe
ulated to compute molecular dy-
and analysis of the properties of
w how it can be used to compute
e used a Hopfield network to com-
en twist and chair conformational

2. Formulation of Neural Networks to Compute Molecular Dynamics

2.1. *Neural Networks*

As shown in Fig. 1, a neural network consists of nodes connected to other nodes [2,9,13,35]. Each node has a value associated with it. These values may be discrete (such as -1 or $+1$) or continuous. The nodes can be connected together in different ways. Each node may be connected to all the other nodes, or to only its nearest neighbors, or they may be organized into a layered structure, so that only nodes between adjacent layers are connected.

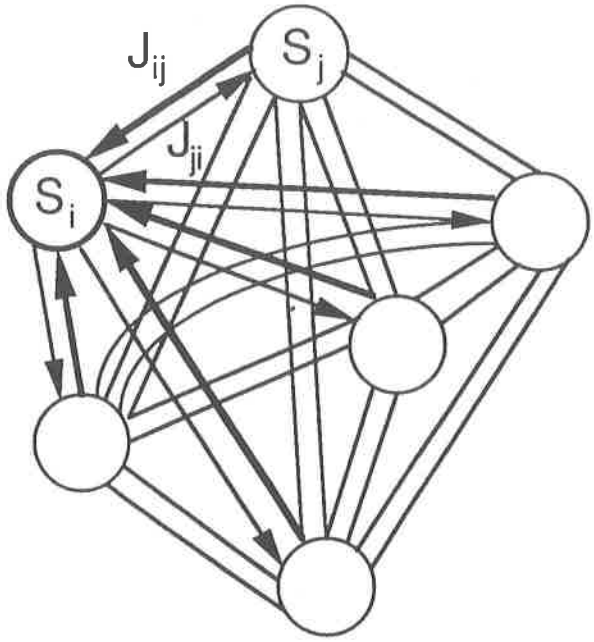


Fig. 1. Schematic representation of a neural network. The network consists of nodes (circles) and connections (arrows) between them. Each node has a value associated with it. The value of ith node is equal to S_i. At each time step, the value of a node is updated. The new value depends on the values of other nodes, such as S_j, and the strength J_{ij} of the connections between them.

Between any pair of nodes, the connection strengths can be equal in both directions (symmetric) or unequal (asymmetric). The connection strengths can be assigned directly, or determined by a learning procedure where the strengths are adjusted so that the network produces the required output for a given input.

The values of the nodes can be updated continuously or at discrete time steps. In the case of discrete updating, the new value of a node (S_i) at each time step is a function of the values of the other nodes (S_j) and the strength of the connections (J_{ij}) between them. This function may directly determine the new value of the node or determine the probability that the node has a certain value.

In order to use a neural network to compute molecular dynamics we must: (1) encode the spatial structure of the molecule in the values of the nodes, (2) determine the topology and the strengths of the connections between the nodes so that the network has the same energy structure as the molecule, and (3) update the values of the nodes in a way that includes both the influence of thermal fluctuations due to the interaction with environment and the forces within the molecule that are modeled by the connection strengths. The output of the computation is the value of the nodes, and thus the spatial structure of the molecule, as a function of time.

We now consider these issues of determining the encoding, the connection strengths, and the updating method.

2.2. *Encoding the Spatial Structure*

The structure of the molecule needs to be represented in the values of the nodes of the neural network. This can be done in different ways. The method chosen will be most efficient when the encoding is based on the form of the molecule. It should also be chosen so that it simplifies the computation of the energy of the network. In a protein, for example, the values of the nodes can encode the spatial positions of atoms, amino acid residues, or subunits of the molecule. In each case, the values of the nodes can represent either the spatial coordinates, the angles between the units (such as the ϕ, ψ angles between amino acid residues), the coefficients of a series expansion of the spatial positions based on polynomial splines [42,43], or Fourier components [49], or a number identifying the spatial location corresponding to a conformational state (such as -1 if an ion channel protein is closed and $+1$ if it is open).

It is important that the change of a value of one node should correspond to a small change in the spatial structure of the molecule. For example, a poor encoding would be one where the coordinates of the positions of each atom are represented by the binary number formed from a set of nodes, each of which has the value 0 or 1. The problem with this encoding is that a change in the value of only one node in a position corresponding to a high power of 2 will produce a large change in the number denoting the spatial position of the atom in the molecule.

2.3. *Topology and Connection Strengths*

We studied three types of neural networks to compute molecular dynamics: activation networks, layered networks and Hopfield networks.

In the activation network all the nodes are connected to each other. The values of the nodes (S_i) are typically continuous and the connection strengths are typically asymmetric ($J_{ij} \neq J_{ji}$). These networks were developed to describe the behavior of the brain in analyzing sensory input [7]. In that case, the values of the nodes correspond to the strengths of characteristic features [34]. These networks are also useful for classifying data into clusters with different characteristics [26]. In

computing molecular dynamics, the connection strengths are determined from the influence of one node on another. For example, if the nodes are atoms, the strengths can be made proportional to the force of one atom on another. If the nodes are amino acid residues, the strengths are proportional to the interactions between pairs of residues that can be described by a two dimensional connectivity matrix [6,44]. In some preliminary studies we used an activation network to compute the switching of cyclohexane between its twist and chair conformational states. Each node corresponded to one of the six carbon atoms. The value of each node was 1 if the carbon atom was closer to its position in the twist configuration and -1 if the carbon atom was closer to its position in the chair configuration. The connection strengths were determined from the change in energy computed by the program Insight II (Biosym Technologies Inc.) when the position of one atom at a time was perturbed.

In the layered network with three layers, a layer of input nodes connects to a middle layer of nodes, which is in turn connected to an output layer of nodes [13,35]. The middle layer is called a hidden layer because it is not connected directly to the input or output signals. The values of all the nodes are typically continuous. The connection strengths are typically asymmetric. The values of the output nodes are a nonlinear function of the values of the input nodes. These networks were developed because networks without a hidden layer could not compute some simple functions (such as the output of the exclusive "or" function from two input values). The hidden layer usually has fewer nodes than the input layer. Thus, the hidden layer extracts the characteristics of the input and presents them as a smaller number of input values to the output layer. These networks are popular in applications because the connection strengths do not have to be computed explicitly. They can be determined by a learning procedure (such as back propagation) where the connection strengths are iteratively adjusted to provide the best match of output values compared to the desired output values for a training set of input values. In computing molecular dynamics, the connection strengths can be determined by a learning procedure. For example, an energy computation program (such as Insight II) may be first used to compute the energy of the molecule in different conformational states and perturbations of the positions of the atoms from those states. The connection strengths can then be determined by using a training procedure to find the connection strengths that provide the closest output energies for the structural information held fixed at the input nodes. It is not yet clear how large a training set is needed to adequately represent a protein. The representation formed in the hidden layer provides an approximate representation of the energy of the molecule as a function of its spatial structure. Once the connection strengths have been determined, the values at the input nodes that represent the structure of the molecule are no longer held at their fixed input values, but are free to be determined by the rest of the network. These values will evolve as a function of the values of the other nodes and an additional stochastic component, applied to the network, in order to model thermal fluctuations.

In the Hopfield network all the nodes are connected to each other. The values of the nodes (S_i) are typically discrete (such as -1 and $+1$) and the connection strengths are symmetric ($J_{ij} = J_{ji}$). The stable conformational states of the molecule, and perhaps some additional states, are represented as memories. The connection strengths are directly computed from these memories. These networks were formulated using concepts from thermodynamics and dynamical systems [2,9,21,22,23]. Thus, their energies and dynamics most closely resemble those in physical systems such as molecules. Hence, we felt these networks were the most promising type to use to compute molecular dynamics. The properties of these networks and their application to computing molecular dynamics are described in detail below.

2.4. *Updating and Dynamics*

There are different methods to update the values of the nodes. In synchronous updating all the values of the nodes are changed simultaneously to their new values. In asynchronous updating the value of one node at a time is changed. The node chosen for updating can be determined randomly or by a preset order. The dynamics of some networks depends on the updating method used [2]. In computing molecular dynamics, the dynamics of the network must not depend sensitively on the details of the updating method used. We found that our simulations of small activation networks were most robust when the value of one randomly chosen node was updated at each time step.

The new value of each node is a function of the values of the other nodes and their connection strengths. This function consists of two components. A deterministic component represents the deterministic forces in the molecule. A random component represents the stochastic forces due to thermal energy. The dynamics of the neural network determined by the updating method must correspond to the physical dynamics of the molecule. For example, it can be shown that the updating method (described below), based on Glauber dynamics, does correctly lead to a fraction of time spent in each of two states that is equal to $\exp(-\Delta E/kT)$ where ΔE is the energy difference between the two states, k is the Boltzmann constant and T is the absolute temperature. For the Hopfield network the "energies" of the network correspond to the physical energies. Thus, the dynamics of these networks can be directly related to the dynamics of the molecule. For the activation and layered networks it is less clear how to relate the "energies" of the network to the physical energies.

3. Hopfield Network

3.1. *General Formulation*

In this network all the nodes are connected to each other [21–23]. In the original model the values of the nodes had two discrete values. This was later extended to continuous values. Here we will use binary nodes with the discrete values of $S_i = -1$

or $S_i = +1$. Hopfield showed that if the connection strengths were symmetric ($J_{ij} = J_{ji}$) the network had two important properties. First, a physically meaningful energy function could be defined. Second, the values of the nodes of the network evolve in a way that continually lowers the energy function, until the values of the nodes reach the values corresponding to a "memory" encoded in the connection strengths. The memories correspond to local minima of the energy function. The network evolves, like a dynamical system dominated by friction rather than inertia, travelling downhill in an energy landscape until it reaches the memory at the energy minimum. The network is called associative or content addressable because the final values of the nodes share common characteristics with the initial values of the nodes. The original idea was that all of the energy minima would correspond to the memories encoded in the network. However, additional shallow minima arise from the interactions between the memories. In the original deterministic updating method the network sometimes reached and remained in these spurious memories. This can be avoided by using stochastic dynamics which makes it possible to go uphill in energy, with small probability, and thus escape from the shallow local minima of the spurious states.

The energy E has the form:

$$E = -\frac{1}{2}\sum_{i,j=1}^{N} J_{ij} S_i S_j \,, \tag{3.1}$$

where N is the number of nodes in the system, S_i, $S_j = \pm 1$ are the values of nodes i and j, and J_{ij} is the connection strength between them. The state of the network at a given time is given by the values of all the nodes, which we denote by the N-dimensional vector

$$\mathbf{S} = (S_1, \ldots, S_N) \tag{3.2}$$

which has components $S_i = \pm 1$. Equation (3.1) is the form of the energy for a system with N binary nodes where there are long range interactions that are bilinear in S_i.

Each memory corresponds to a set of values of the nodes. Thus, each memory can also be represented as a vector

$$\xi = (\xi_1, \ldots, \xi_N) \tag{3.3}$$

with N components, each of which $\xi_i = \pm 1$. The values of the components of these memories are determined *a priori*. They depend on how the problem to be solved is encoded into the network. If there are p different memories, then the entire set of memories can be represented by the vectors

$$\xi^\mu = (\xi_1^\mu, \ldots, \xi_N^\mu), \qquad \mu = 1 \cdots p \,. \tag{3.4}$$

The connection strengths J_{ij} are determined from the memories, namely

$$J_{ij} = \frac{1}{N}\sum_{\mu=1}^{p} \xi_i^\mu \xi_j^\mu \,. \tag{3.5}$$

A set of very useful quantities are called the overlaps, which are given by

$$m_\mu = \frac{1}{N} \sum_{i=1}^{N} \xi_i^\mu S_i \,. \tag{3.6}$$

Geometrically m_μ is the cosine between vectors ξ^μ and $\mathbf{S}$ in N-dimensional space. These overlaps provide a quantitative measure of the discrepancy between a given state $\mathbf{S}$ of the network and the state of the network that corresponds to a given memory ξ^μ. An overlap may have one of the $(N+1)$ values: -1, $(-1+2/N)$, $(-1+4/N), \ldots, (-1+2(N-1)/N)$, 1.

The overlap m_μ describes how close the state of the network is to a given memory. If the state coincides with the memory, or its mirror image, then the overlap m_μ has its maximum absolute value of 1. On the other hand, if the state of the network is far from a given memory, then the overlap m_μ is close to 0. Two different memories are "orthogonal" when there is zero mutual overlap between them. Thus, memories ξ^μ, $\xi^{\mu'}$ are orthogonal if $m_{\mu\mu'} = 0$.

Using Eqs. (3.5) and (3.6), the energy in Eq. (3.1) can be rewritten as

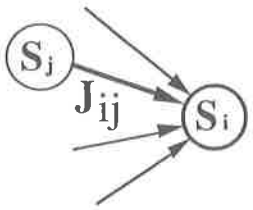
$$E = -\frac{N}{2} \sum_{\mu=1}^{p} m_\mu^2 \,. \tag{3.7}$$

If the overlaps with all the memories are small, then the state $\mathbf{S}$ has high energy. If the overlap with one or more memories is large, then the state $\mathbf{S}$ has low energy.

The dynamics of the network is determined by the updating method. We will update one randomly chosen node i at each time step. If the value of the node changes, then the state $\mathbf{S}$ of the network will change. The updating method must ensure that the state $\mathbf{S}$ of the network will reach the local minima in the energy function Eq. (3.7) corresponding to one of the p memories in Eq. (3.4). It must also satisfy the thermodynamic requirement that the probability that the network is in state $\mathbf{S}$ is proportional to $\exp(-E(\mathbf{S})/T)$, where $E(\mathbf{S})$ is the energy in state $\mathbf{S}$, and T is the temperature expressed in energy units. An updating method that meets these requirements is illustrated in Fig. 2 and described in detail in Appendix 1. In brief, this updating method consists of first computing the change in the energy of the network ΔE that would result if the value of node i were changed from S_i to $-S_i$, which is given by

$$\Delta E = 2S_i \sum_{j=1, j\neq 1}^{N} J_{ij} S_j \,. \tag{3.8}$$

This change in energy ΔE determines the probability $P(\Delta E)$ that the value of the node should be changed. For the Glauber dynamics (described in Appendix 1) this



$$\Delta E_{S_i \to -S_i} = 2\, S_i \sum_{j=1,\, j\neq i}^{N} J_{ij} S_j$$

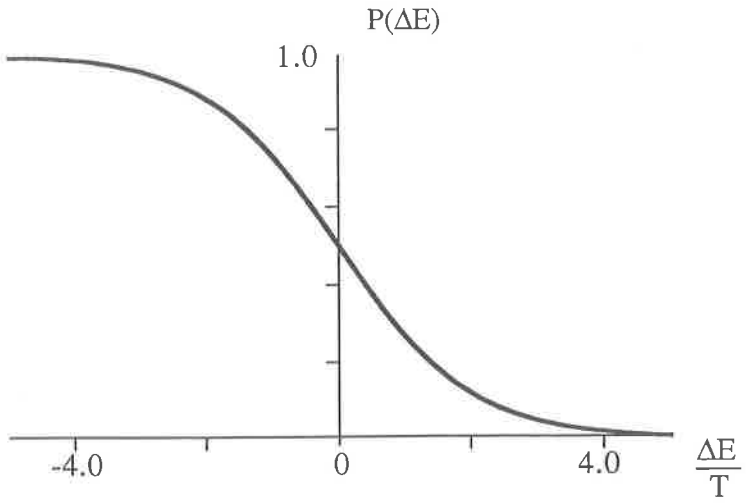


Fig. 2. Schematic representation of the Glauber updating method of the Hopfield network. The values of each node are either -1 or $+1$. At each time step a node i is chosen at random. The new value of this node depends on the values S_j of the other nodes and their connection strengths J_{ij}. First, the quantity ΔE is determined. Then $P(\Delta E)$ is determined from Eq. (3.9). T is the absolute temperature in energy units. Then a random number R is chosen in the range $0 < R < 1$. If $P(\Delta E) < R$, then the value of node i is multiplied by -1, otherwise it remains the same.

probability is given by

$$P(\Delta E) = \frac{1}{2}\left(1 - \tanh\left(\frac{\Delta E}{2T}\right)\right). \tag{3.9}$$

At each step in the computation a node is chosen at random. Then ΔE and $P(\Delta E)$ are determined. A random number $0 < R < 1$ is then chosen from a uniform

distribution. If $P(\Delta E) > R$, then the value of the node remains at S_i, and if $P(\Delta E) \leq R$, then the value of the node is changed to $-S_i$.

3.2. *Properties of Hopfield Network Relevant to Molecular Dynamics*

First, we show that the energy function of the Hopfield network has stable states corresponding to the memories, which are separated by many shallow minima corresponding to the spurious memories. This is analogous to the energy function of large molecules, such as proteins, which have local minima corresponding to stable conformational shapes separated by a range of many energy barriers with many shallow minima [11,12,25,33,47]. Consider the change in energy of the network if the value of node i were to change from S_i to S_i'. Since $S_i = \pm 1$, the change is equivalent to inverting the value, namely

$$S_i' = -S_i \,. \tag{3.10}$$

The new overlap m_μ' will then be

$$m_\mu' = m_\mu - \frac{2\xi_i^\mu S_i}{N} \tag{3.11}$$

and the change in energy will be

$$\Delta E = E' - E = -\frac{N}{2}\sum_{\mu=1}^{p}(m_\mu'^2 - m_\mu^2) = -\frac{N}{2}\sum_{\mu=1}^{p}(m_\mu' - m_\mu)(m_\mu' + m_\mu)$$

$$= \sum_{\mu=1}^{p}\xi_i^\mu S_i\left(2m_\mu - \frac{2\xi_i^\mu S_i}{N}\right) = 2\left(S_i\sum_{\mu=1}^{p} m_\mu \xi_i^\mu - \frac{p}{N}\right). \tag{3.12}$$

Since $N \gg P$, the p/N term in Eq. (3.11) is small. Thus,

$$\Delta E > 0 \quad \text{if } S_i \sum m_\mu \xi_i^\mu > 0 \text{ and}$$

$$\Delta E < 0 \quad \text{if } S_i \sum m_\mu \xi_i^\mu < 0 \,. \tag{3.13}$$

If the state $\mathbf{S}$ coincides with one or more of the memories (or their mirror images) $\mathbf{S} = \pm\xi^\mu$, and all memories are orthogonal (mutual overlaps between memories are zero), then all overlaps except one are equal to zero, and thus $\Delta E = 2|m_\mu| = 2$. Since a change in the value of any node only increases the energy, these states are located at the lowest point of a local energy minima, and are thus stable. However, there also exist other states $\mathbf{S}$, far from the memories, which can also satisfy $\Delta E > 0$ for all i in Eq. (3.13). These states are also stable. Thus, far from any one particular memory, the energy landscape has a rough surface with many shallow minima. In order to go from one state to another the energy of the network has to rise over

e of the node remains at S_i, and if
anged to $-S_i$.

levant to Molecular Dynamics

ne Hopfield network has stable states
arated by many shallow minima cor-
analogous to the energy function of
local minima corresponding to stable
of many energy barriers with many
e change in energy of the network if
o S_i'. Since $S_i = \pm 1$, the change is

(3.10)

$$\frac{\xi_i^\mu S_i}{N} \tag{3.11}$$

$$-\frac{N}{2}\sum_{\mu=1}^{p}(m'_\mu - m_\mu)(m'_\mu + m_\mu)$$

$$\sum_{\mu=1}^{p} m_\mu \xi_i^\mu - \frac{p}{N}\Big). \tag{3.12}$$

nall. Thus,

$\xi_i^\mu > 0$ and

$$\xi_i^\mu < 0\,. \tag{3.13}$$

e memories (or their mirror images)
tual overlaps between memories are
o zero, and thus $\Delta E = 2|m_\mu| = 2$.
ncreases the energy, these states are
nima, and are thus stable. However,
ories, which can also satisfy $\Delta E > 0$
e. Thus, far from any one particular
face with many shallow minima. In
rgy of the network has to rise over

many intermediate barriers to reach a globally stable state. This property is called ultrametricity.

The dynamics of the Hopfield network depends on two components: a deterministic function of the values of the nodes and their connection strengths, and a stochastic function. This is analogous to the deterministic forces between the atoms and the stochastic forces due to thermal energy in a molecule. The switching of the network from one state to another can be analyzed by a Markov processes where a transition probability matrix describes the probability per time step of a jump from state $\mathbf{S}$ to $\mathbf{S}'$. The details of this dynamic analysis are described in Appendix 1.

The dynamics of the Hopfield network has different qualitative properties, called phases, depending on the number of nodes, the number of stored memories, and the temperature. These phases are analogous to the phases of proteins such as the folded conformation, random coil, molten globule, or unfolded conformation [41].

The dynamics of the network depends on the number of nodes. When the number of nodes N is finite, the state of the network evolves so that after long times the probability that the network is in state $\mathbf{S}$ is proportional to $\exp(-E(\mathbf{S})/T)$, where $E(\mathbf{S})$ is the energy in state $\mathbf{S}$. This is described in detail in Appendix 1. As the number of nodes N is increased, the time for the network to visit all the states also increases. This is not only because the number of states is larger, but also because the energy barriers between the memories increases with increasing N. The height of the energy barriers between the memories is approximately proportionally to the number of nodes N. This can be inferred from Eq. (3.7). When the state of the network is far from all the memories, all the overlaps m_μ are approximately 0, and the energy of the network has its maximal value approximately equal to 0. When the state of the network is at one memory, the overlap of that memory $m_\mu = 1$, and if the other memories are orthogonal, their overlaps $m_{\mu'} = 0$, and the energy of the network is near its minimal value approximately equal to $-N/2$. Thus, the height of the energy barriers between the memories is approximately equal to $-N/2$.

The dynamics of the network depends dramatically on the temperature. We consider the case where the network has a very large number of nodes N. At high temperatures, the network will have enough energy to cross over the energy barriers into all the states corresponding to all the memories. Thus, as the state of the network evolves in time, it will eventually pass through the states of all the memories. This behavior is called ergodic. However, at low temperatures, the network will not have enough energy to cross over the energy barriers into all the states during a long time, and hence its behavior is nonergodic. Thus, there is a critical temperature at which the dynamics of the network passes through a phase transition between ergodic and nonergodic dynamics. At even lower temperatures, there are additional phase transitions as the state of the network is increasingly trapped into smaller fractions of all the possible states. At very low temperatures the state of the network may only be able to reach the nearest local minima among the spurious states that arise from the interactions of the memories.

The probability that the state of the network is within the domain of a given memory and has a given energy between E and $E + \Delta E$ is proportional to the product of the Boltzmann factor $\exp(-E(\mathbf{S})/T)$ and the number of states of the network within the domain of this memory and within this energy interval. This is analogous to the fact that the probability to find a molecule in a given conformation with a given free energy depends both on the energy of that conformation and its entropy. The Boltzmann factor decreases with increasing energy. The number of states increases with increasing energy. This is because as the energy increases there are more nodes that have values that are not equal to the value of the memory and thus there is a larger number of such combinations that have the same energy (Appendix 3). At low temperature, the energy dependence of the Boltzmann factor dominates, and the probability that the network has a given energy decreases with increasing energy. At high temperature, the energy dependence of the Boltzmann factor is less strong, the factor due to the number of states dominates, and the probability increases with increasing energy.

If the energy landscape has separate, deep, well defined minima then we can identify several different conformational states. This is analogous to proteins with well defined, stable structures and the distribution of times spent in each state is the sum of exponential terms. If the energy landscape has many, shallow minima, then we cannot identify unique structures. This is analogous to proteins with many conformational substates and the distribution of times spent in each state is nonexponential.

3.3. *Computing Molecular Dynamics Using a Hopfield Network*

Our goal is to formulate a network with the dynamics corresponding to the dynamics of a molecule switching between different conformational states. First, we must encode the spatial structure of the molecule into the network. Second, the energy landscape of the network must be constructed so that it resembles that of the molecule. We will do this by encoding the stable states of the molecule as the memories ξ^{μ}.

First, we encode the spatial structure of the molecule in the values of the nodes of the network. Structurally similar conformations of the molecule should correspond to states of the network that are close to each other, and dissimilar conformations of the molecule should correspond to states of the network that are far from each other. Here we describe one such encoding that is particularly useful for a molecule consisting of a linear string of connected units, for example, a string of amino acid residues in a protein. The backbone of the molecule will be a curve in 3-dimensional space

$$\mathbf{X}(s) = (X_1(s),\, X_2(s),\, X_3(s))\,. \tag{3.14}$$

The distance from the beginning of the chain is given by the variable s, where $0 \leq s \leq L$, and L is the length of the chain. The origin of the coordinate system is assumed to coincide with the center of mass of the molecule.

We now define a function $F(s')$ on the interval $0 \leq s' \leq 3L$, to represent the structure of the molecule:

$$\begin{aligned} F(s') &= X_1(s') & 0 < s' < L \\ F(s') &= X_2(s' - L) & L < s' < 2L \\ F(s') &= X_3(s' - 2L) & 2L < s' < 3L \end{aligned} \tag{3.15}$$

We distribute the nodes of our network uniformly along the interval [0,3L] and assign a value to each node of -1 or $+1$ depending on the sign of the function $F(s')$ at this point, namely

$$S_i = \text{sign}(F(s')) . \tag{3.16}$$

The number of nodes of the neural network may be much smaller than the number of amino residues in the protein if the position of the backbone varies smoothly in space. If the dynamics and biologically important features depend more on the overall shape of the structure, rather than its fine details, than we can smooth $F(s')$ by expanding it into a Fourier series and retain only the leading terms. This may significantly reduce the complexity of the description. However, some proteins may have important properties determined by the fine details of the local changes in conformation, and in these cases the higher order Fourier harmonics must be retained.

Now we define the representation of the energy. We begin by determining the memories that represent the stable conformational states of the molecule. We express the energy in terms of the overlaps between the current state **S** of the network, and these memories ξ^μ. That is, we consider that a particular state of the network consists of a combination of the memories. The physical interpretation of this representation is that a particular structure of the molecule consists of a combination of its stable conformational states. The values of overlaps describe the proximity of the current state to all the different memories, that is, the relative contribution of those stable conformational states. In this encoding the overlaps and thus the energy changes by only a small amount when the value of one node is updated. This important feature corresponds to the fact that over small time intervals, the structure of the protein changes by only a small amount.

Some features of the expression for the energy in Eq. (3.7) now need to be considered. First, the basis functions ξ^μ, $\mu = 1 \cdots p$ are not necessarily orthogonal or uncorrelated. Second, we will assign a specific weight factor α_μ to each memory to encode the relative stability of its corresponding conformational state. Third, the energy defined by Eq. (3.7) depends on the number of nodes N. This is not a useful feature since the number of nodes N is arbitrary and depends on the method used to encode the spatial structure of the molecule. We remove the dependence on N by dividing Eq. (3.7) by N. This also ensures that the energy remains finite in the limit of large N. Thus, using Eqs. (3.1), (3.5) and (3.6), our new expression

for the energy of a particular state **S** is defined as

$$E = -\frac{1}{2N^2} \sum_{\mu=1}^{p} \sum_{i,j=1}^{N} \alpha_\mu \xi_i^\mu \xi_j^\mu S_i S_j = -\frac{1}{2} \sum_{\mu=1}^{p} \alpha_\mu m_\mu^2 , \tag{3.17}$$

where α_μ are the weights of the memories, and m_μ are the overlaps between μth memory ξ^μ and the current state **S**.

The values of weights α_μ are determined from the constraints that the energy in μth memory is equal to E_μ, which is the known value of the depth of the energy minima corresponding to stable conformational state μ. This leads to the set of linear equations for the determination of α_μ:

$$E_\mu = -\frac{1}{2} \sum_{\mu'=1}^{p} \alpha_{\mu'} m_{\mu\mu'}^2 \tag{3.18}$$

where $m_{\mu\mu'}$ is the mutual overlaps between memories ξ^μ, $\xi^{\mu'}$. For orthogonal memories, where $m_{\mu\mu'} = \delta_{\mu\mu'}$ (Kronecker delta symbol), then

$$\alpha_\mu = -2E_\mu . \tag{3.19}$$

Using the values of α_μ from Eq. (3.17) and the memories ξ^μ we can now determine the values of the connection strengths:

$$J_{ij} = \frac{1}{N^2} \sum_{\mu=1}^{p} \alpha_\mu \xi_i^\mu \xi_j^\mu . \tag{3.20}$$

Up to now we have constructed the expression for the energy so that it includes only the information about the stable conformational states. However, we can expand the definition of the energy function so that it also includes information about metastable or unstable conformational states. We can do this by adding additional positive terms to the right hand side of Eq. (3.17) of the form

$$\cdots + \frac{1}{2} \sum_{\nu} \beta_\nu m_\nu^2 . \tag{3.21}$$

We then use a procedure analogous to Eq. (3.18) to determine the values of α_μ and β_μ.

The simple bilinear form of the energy that we have used determines only the location and depth of the energy minima. An advantage of this form is its simplicity. A disadvantage is that it is not complex enough to represent the shape of the energy minima. However, the shape of the energy minima also depend on other parameters of the model. Thus, we can include additional information about the energy minima by properly adjusting those parameters. For example, the effective width of the potential wells near the memories depends on the number of nodes of the network.

By adjusting the number of nodes N, as described in Appendix 3, we adjust the width of the potential wells to generate energy landscapes with narrow minima that are well separated, or broad minima that intersect.

The neural network approach may be much more computationally efficient than the standard molecular dynamics computation. In the standard molecular dynamics computation the time step needs to be kept small enough to keep the changes in the potentials small during the time step. In the neural network approach only the changes in the generalized objects (the overlaps) need to be kept small for each updating step. Thus, the physical time corresponding to a time step in the neural network computation may be considerably longer than the physical time of the time step in the standard molecular dynamics computation. Moreover, the updating in the standard molecular dynamics requires changing the position of all the atoms and, in some schemes, thermalization of their velocities due to the interaction with the solvent. The additional thermalization step is not necessary in the network approach, because thermalization is included implicitly in the dynamics itself. The energy structure of the neural network is a simplified description of the energy structure of the molecule. The accuracy of the dynamics computed depends on the accuracy of this energy approximation.

4. Hopfield Network Computation of Molecular Dynamics of Cyclohexane

We illustrate the general ideas above by formulating a Hopfield network to compute the dynamics of the small, organic molecule cyclohexane. Applying the neural network approach to this simple molecule allows us to trace all steps of the method and to elucidate the nature of problems that may arise.

4.1. *Encoding the Spatial Structure and Energy*

Cyclohexane is a 6-membered carbon ring that has two stable conformational structures, called the twist and the chair, which are illustrated in Fig. 3. There are

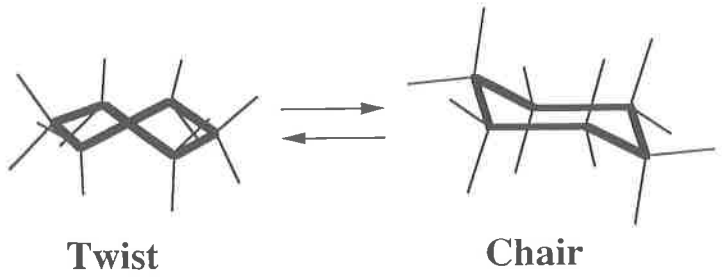


Fig. 3. The small carbon ring molecule cyclohexane can exist in different conformational states. The most stable forms are the twist and chair. The energy from thermal fluctuations causes the molecule to spontaneously switch between these conformational states.

additional unstable conformational structures between the twists that are called boats. Thermal fluctuations provide enough energy for the molecule to spontaneous interconvert between these structures. Because of the many symmetries in this molecule, its energy function can be represented by a 2-dimension surface in a 3-dimensional space [38], which is shown in Fig. 4. This energy surface has 2 global minima, 6 local minima and 6 saddle points. The 2 chair structures correspond to the deep, global minima at the poles. The 6 twist structures correspond to the shallow, local minima distributed uniformly along the equator. The 6 boat structures correspond to the saddle points between the twists. The coordinate axes of Fig. 4 are the coefficients of the Fourier harmonics for the out of plane displacements of the carbon atoms shown in Fig. 5. The z coordinate correspond to the coefficient of the $\cos(6\pi s/L)$ harmonic, while the x and the y coordinates correspond to the coefficients of the $\cos(4\pi s/L)$ and $\sin(4\pi s/L)$ harmonics respectively.

To encode the spatial structure of the molecule in the network we use the coordinate $0 \leq s \leq L$ of the distance along the ring. Since the molecule is cyclic, the positions $s = 0$ and $s = L$ coincide. Thus, the nodes of the network are distributed along the coordinate s. The structures of the chair, twist and boat can be described

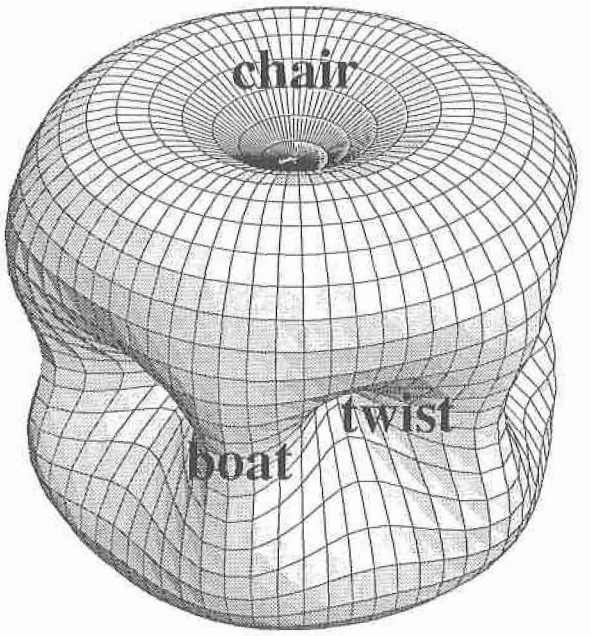
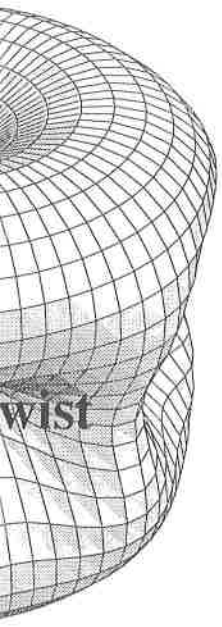


Fig. 4. The energy function of cyclohexane can be presented by a two dimension surface in a three-dimensional space. The surface shown is a schematic representation of the energies calculated by Pickett and Strauss [38]. The deep minima at the poles of the energy surface correspond to the two chair conformations. The shallow local minima along the equator correspond to the six twist conformations. The saddle points along the equator correspond to the six boat conformations.

between the twists that are called
energy for the molecule to sponta-
Because of the many symmetries in
sented by a 2-dimension surface in a
. 4. This energy surface has 2 global
The 2 chair structures correspond to
ist structures correspond to the shal-
the equator. The 6 boat structures
wists. The coordinate axes of Fig. 4
or the out of plane displacements of
rdinate correspond to the coefficient
the y coordinates correspond to the
harmonics respectively.
cule in the network we use the coor-
ng. Since the molecule is cyclic, the
nodes of the network are distributed
nair, twist and boat can be described



resented by a two dimension surface in a
ic representation of the energies calculated
les of the energy surface correspond to the
ng the equator correspond to the six twist
respond to the six boat conformations.

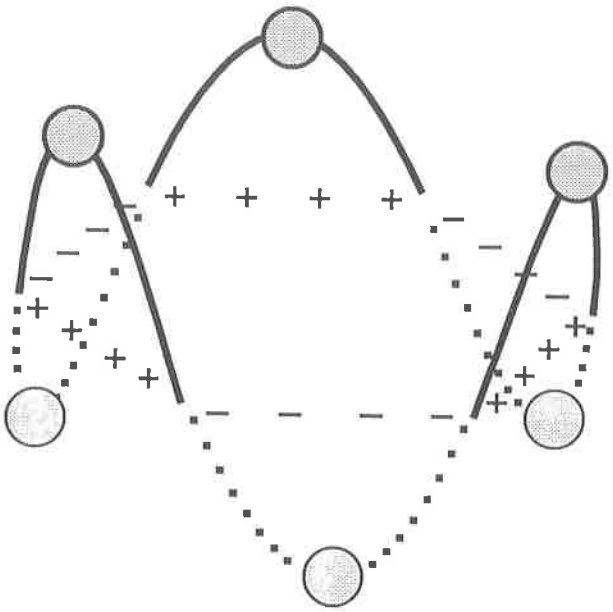


Fig. 5. The method used to encode the spatial structure of cyclohexane in the values of the nodes of the Hopfield network. The nodes of the network correspond to spatial positions around the carbon ring. If the atoms or the bonds connecting them are below the plane, the value of the corresponding node is -1. If the atoms or the bonds connecting them are above the plane, the value of the corresponding node is $+1$. The values of -1 and $+1$ are represented here as $-$ and $+$.

by the displacements of the carbon atoms in the direction perpendicular to the plane of the ring [38] as shown in Fig. 5. Since these displacements take place in only one direction, we have the simplification that $F(s')$ depends only on the one coordinate X_1. These small out of plane displacements of the chairs, twists and boats on the interval $[0, L]$ are given by:

$$\begin{aligned}
&\pm\cos\left(3\cdot\frac{2\pi s}{L}\right) && \text{for chairs}\\
&\pm\sin\left(2\cdot\frac{2\pi s}{L}\right)\ \text{and}\ \pm\sin\left(2\left(\frac{2\pi s}{L}\pm\frac{\pi}{3}\right)\right) && \text{for twists}\\
&\pm\cos\left(2\cdot\frac{2\pi s}{L}\right)\ \text{and}\ \pm\cos\left(2\left(\frac{2\pi s}{L}\pm\frac{\pi}{3}\right)\right) && \text{for boats}
\end{aligned} \tag{4.1}$$

The value of the ith node $S_i = -1$ if the displacement at that position s is negative, and $S_i = +1$ if the displacement is positive.

We will use the two types of stable structures (chair and twists) as the memories to be encoded into the network. The connection strengths are determined from these memories using Eq. (3.5). The boat structures do not correspond to minima in the energy function, and thus we will not use them as memories. (However, we

will see below that the existence of the boat structures arises from the interaction of the chair and twist memories.) It is only necessary to use the chair and twist memories with the "+" sign in Eq. (4.1), since those with the "−" sign appear automatically, because the expression for energy is invariant under the transformation $\mathbf{S} \rightarrow -\mathbf{S}$. Therefore, if ξ is a memory, then its mirror image $-\xi$ has the same energy. Thus, we use the memories corresponding to 1 chair and 3 twists. The additional mirror symmetric structures are automatically included in the energy structure of the network.

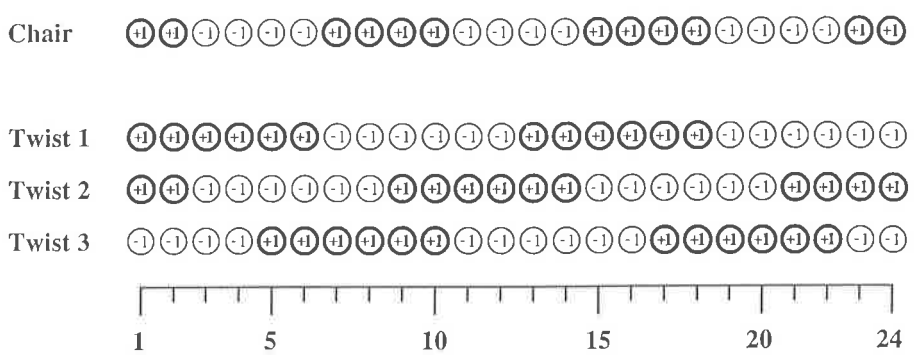


Fig. 6. Memories used to represent the structure of cyclohexane in the Hopfield network. The values are given for the $N = 24$ nodes of the network for the one memory that corresponds to the chair conformation, and for the three memories that correspond to the twist conformations.

The chair and twist memories are orthogonal and the chair and boat memories are orthogonal and thus their overlaps are equal to 0. However, the twists among themselves are not orthogonal, and the twists and boats are not orthogonal. These overlaps are given by:

$$m_{\text{twist-twist'}} = -\frac{1}{3}, \qquad m_{\text{twist-boat}} = 0, \ \pm\frac{2}{3} \tag{4.2}$$

4.2. *Topology and Connection Strengths*

Due to the fact that twist memories are not mutually orthogonal, we have to use the general procedure of Eq. (3.18) to calculate the weights α_μ of the memories. Using Eqs. (4.2), (3.18) and (3.19), we find that

$$\alpha_{\text{chair}} = -2E_{\text{chair}}, \qquad \alpha_{\text{twist}} = -\frac{18}{11}E_{\text{twist}}\,. \tag{4.3}$$

If the highest energy barrier of the transition state is defined to be zero, then $E_{\text{chair}} = -14.5$ kcal/mol and $E_{\text{twist}} = -6.5$ kcal/mol [38].

Then, from Eqs. (3.18), (4.2) and (4.3) we conclude that

$$E_{\text{boat}} = \frac{8}{11}E_{\text{twist}}\,. \tag{4.4}$$

Note that we did not explicitly put into the network any information about the existence or energy of the boat structure. The boat structure arises from the interaction of the chair and twist memories.

4.3. *Updating and Dynamics*

One randomly chosen node was updated at each time step of the computation. Metropolis or Glauber dynamics were used (see Fig. 2 and Appendix 1).

The time in the numerical simulation is an integer number, denoting the total number of elapsed time steps. We have to determine the relationship between the time step of the neural network computation and the physical time in seconds. This is very hard to do because this relationship depends on the updating method and on the parameters of the computation such as the number of nodes N, the number of memories p and the temperature T. This type of problem can sometimes be overcome by an appropriate rescaling of the equations converting them to dimensionless form. We were not able to accomplish such a transformation at this time.

Despite these difficulties we describe one method that we explored in order to relate the time steps in the neural network updating method to a physical time. The temporal behavior of the neural network may be approximated as a diffusion process in the space of the overlaps m_μ. In Appendix 4 we show that for Glauber dynamics the diffusion coefficient D for this process is

$$D \approx \frac{1}{\tau N^2}\,, \tag{4.5}$$

where τ is the time step between two sequential updatings and N is the number of nodes. In Appendix 5 we show that the changes in the protein structure may also be considered a diffusion process driven by the input of energy from collisions of the molecules in the surrounding solution. In a normalized dimensionless space this diffusion coefficient D is given by

$$D \approx \frac{T^{1/2}}{4\sqrt{3}\, l^2 \sigma n \mu^{1/2}}\,, \tag{4.6}$$

where l is a characteristic length, σ is the cross section of the interactions between the protein and the solvent molecules, n is the particle density of the solvent, and μ is the mass of the solvent molecules. Equating D in Eqs. (4.5) and (4.6) we derive an estimate for τ:

$$\tau \approx \frac{1}{DN^2} \approx 4\sqrt{3}\frac{l^2 \sigma n \mu^{1/2}}{N^2 T^{1/2}} \tag{4.7}$$

For cyclohexane in solvent CS_2 at the absolute temperature 200 K, $l = 3 \cdot 10^{-10}$ m, $\sigma = 3 \cdot 10^{-18}$ m^2, $n = 10^{28}$ m^{-3}, $\mu = 1.26 \cdot 10^{-25}$ kg. (Note that the temperature in Eq. (4.7) is in energy units, and needs to be divided by the Boltzmann constant to be expressed in °K.) From Eq. (4.6) we find that $D = 7.9 \cdot 10^9\,\mathrm{s}^{-1}$. For our network with $N = 24$ nodes we find from Eq. (4.7) that

$$\tau \approx 2.2 \cdot 10^{-13}\ \mathrm{s}\,. \tag{4.8}$$

4.4. *Numerical Details and Computer Program*

The network had $N = 24$ nodes. At each time step one node was chosen at random and its value updated. The value of the node was multiplied by -1, and then the new values of the overlaps and the energy were computed. The probability p that this new value should be accepted was determined by Metropolis or Glauber dynamics (Appendix 1). A random number $0 < R < 1$ was chosen from a uniform distribution. If $p < R$, then the new value of the node was retained, otherwise it reverted back to its original value.

To analyze the time evolution of the network, we determined the structure that best corresponds to a given state of the network by computing the absolute value of the overlaps with respect to the 1 chair and 3 twist memories. The memory with the largest absolute value of the overlap then defined the structural state of the network. For example, if the overlap with one of the twists was the largest overlap, then the state was denoted as being that "twist" structure, although at many times it did not coincide exactly with the memory trace of that twist. Thus, the dwell time in the "twist" structure includes wandering between many slightly different twist-like states. The same may be said about the chair "state". This definition of state is the same as the operational definition of conformational state experimentally measured by NMR, X-ray diffraction, light absorption, fluorescence and other techniques, where a measured conformational state includes the interconversions between very similar conformational substates.

To perform the computation we used dimensionless variables of energy and temperature. In cyclohexane, the lowest value of the energy, which corresponds to a chair structure, is 14.5 kcal/mol below the highest value of the energy, which corresponds to an unstable transition state [38]. The dimensionless energy function was equal to the energy divided by 14.5 kcal/mol. Thus, the dimensionless energy had a maximum value of 0 and a minimum value of -1. The dimensionless temperature was equal to the absolute temperature in °K divided by $[(14.5\ \text{kcal/mol})/R]$, where R is the gas constant. Thus, the value of the dimensionless temperature $T = 1$, corresponds to a temperature of 7300 °K.

The computer program was written in standard ANSI C-language. This allowed us to use the sophisticated user-friendly interface of Think C 5.0 on a Macintosh IIfx to debug the program, before transporting it to faster computers like the Silicon Graphics IRIS workstation and Convex supercomputer. The computational results were stored in ASCII files, transferred to the Macintosh IIfx through FTP (File Transfer Protocol), and analyzed and graphed using Igor Software (WaveMetrics Inc.).

Typical computational times on the IRIS to complete 10 000 switches between the twist and chair conformations ranged from minutes at high temperatures, to hours at low temperatures. The computational time depended most strongly on temperature and less strongly on the other parameters, such as the number of nodes. The computational time increased exponentially with decreasing temperature. We

studied some ways that the computation at low temperature might be speeded up considerably by eliminating many of the iterations during which the state of the network does not change.

Some aspects of the coding were important in computational efficiency. We used integer or long integer representations instead of float or double precision where possible. We also precalculated tables of all possible values for certain variables, such as the overlaps. This allowed us to avoid recalculations at every time step. The change of an overlap during one time step is equivalent to a shift by one position in the array of possible overlap values. The dwell time distributions of the twist and chair times were computed from combined histograms of different bin width so that bins are narrow for short times and wide for long times. The method is very efficient and accurate at providing the dwell time distribution over the largest range of dwell times (Liebovitch *et al.*, in preparation).

4.5. *Results*

First we describe some features of the energy function computed by the neural network. The structure and energies of the chair and twist conformations were used in formulating the network. The structure and energy of the boat conformation was then computed by the network. We found that the network predicted that the energy of the boat conformation is equal to −4.8 kcal/mol, which is similar to the value −6.0 kcal/mol given by Pickett and Strauss [38]. We emphasize that we did not explicitly put into the network any information about the boat conformation. Nonetheless, the network contains the structure of the boat conformation and it has approximately the correct energy value. The boat conformation arises out of the interactions of the input chair and twist memories. A new property, especially a global property, that arises from such local interactions is called an emergent property. The boat conformation is an emergent property of the network. In some sense, the few memories chosen to formulate the network have captured the natural form of the molecule. Thus, the network reproduces additional properties of the molecule that may appear distinct to us but are actually emergent properties of the stable conformational states of the molecule.

The energy of the network evolves in time. We computed the number of times the network has each energy value. The functional form of this distribution changes with temperature. This distribution provides a useful test of our computer program. As described in Appendix 2, this distribution can be determined analytically for a network with one memory. As shown in Fig. 7, the distribution computed numerically from the network simulation closely matches the analytical relationship. This correspondence increases our confidence in the computer code.

In this distribution we can observe the qualitative change of the behavior when passing through a certain critical temperature. This is a manifestation of a phase transition in a finite system. The quantitative description of this phenomena is given in Appendix 2. The time spent in a conformational state depends on both the depth

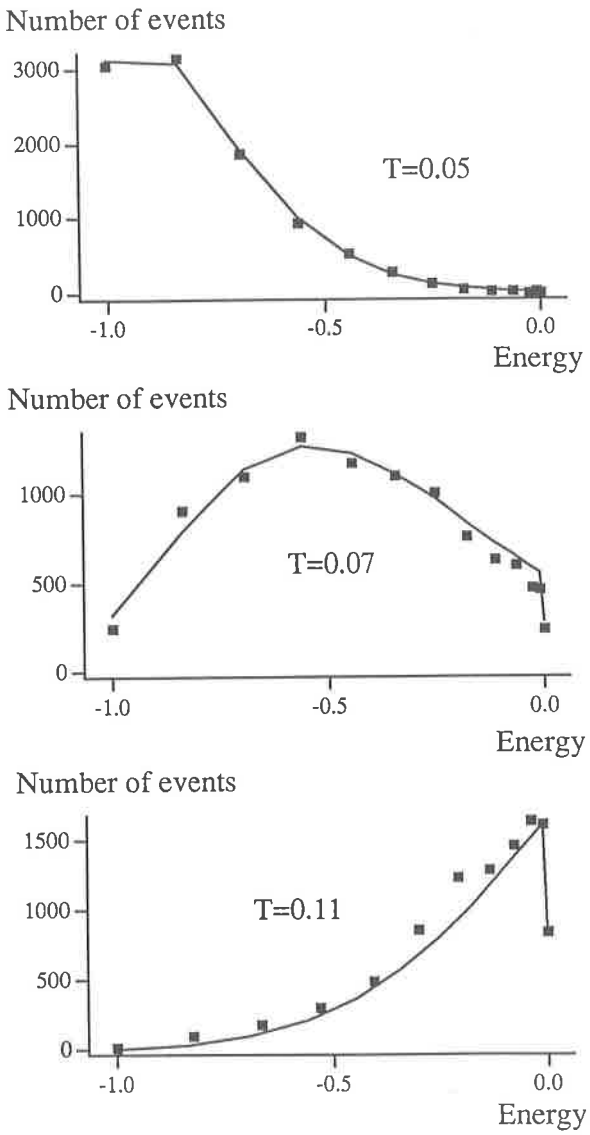


Fig. 7. At a given temperature, the distribution of the number of times that a Hopfield network with one memory has each value of energy is plotted versus energy. The network used had $N = 24$ nodes. The functional form of this distribution changes with temperature. At low temperatures it is monotonically decreasing. At intermediate temperatures it has a local maximum. At high temperatures it is monotonically increasing. The three distributions shown are at $T = 0.05$, 0.07, 0.11 in dimensionless energy units which correspond to temperatures of 365 °K, 511 °K, 803 °K. This distribution is a useful test of the computer code because it can also be determined analytically. The values computed numerically from the neural network simulation are shown as squares. The values computed from Eq. (A1.5) are shown as lines.

of the energy well and the density of states within the well. At low temperatures the molecule spends most of its time in the conformational state corresponding to the deepest potential well. However, at sufficiently high temperatures, the molecule spends most of its time in the conformational state with the largest density of substates. Thus, at higher temperatures, the structures corresponding to shallow, broad wells are more important than structures corresponding to narrow, deep wells.

When the depth of potential wells are kept constant, at a given number of memories, the relative number of states belonging to an energy minima decreases with increasing N (Appendix 3). This makes it less probable for a system to be trapped in any one valley. Thus, the critical temperature needed to ensure that the system visits all valleys decreases. The critical temperature scales in inverse proportion to the number of nodes N. We confirmed this expectation by simulations of networks with different numbers of nodes N.

The time evolution of the energy, conformational state, and overlaps of the cyclohexane network simulations are shown in Fig. 8. The conformational state is defined according to which memory has a maximum overlap. Note that this does not distinguish between the three distinct twist conformations, so that there are

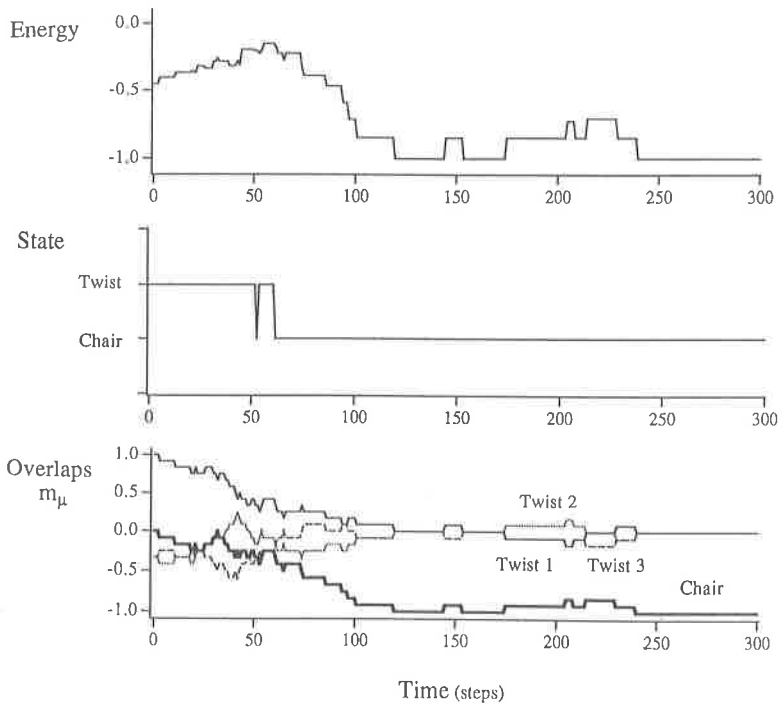
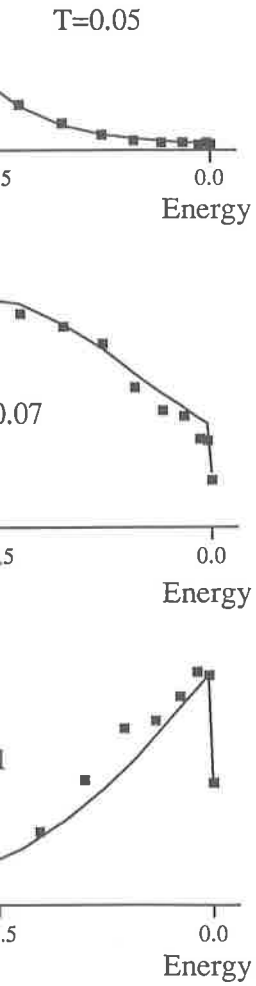


Fig. 8. A small sample of the time evolution of the properties of the Hopfield network representation of cyclohexane. The energy, state and overlaps of the network are shown as a function of time.



the number of times that a Hopfield network
versus energy. The network used had $N = 24$
nges with temperature. At low temperatures
nperatures it has a local maximum. At high
three distributions shown are at $T = 0.05$,
respond to temperatures of 365 °K, 511 °K,
puter code because it can also be determined
the neural network simulation are shown as
shown as lines.

transitions, where the molecule jumps from one twist to another twist without first passing through the chair conformation. Such a transition can be seen near $t = 40$. At the low temperature used to compute Fig. 8, the molecule spends most of its time in the chair configuration and only small part of it in the higher energy twist and the highest energy transition states.

The distribution of dwell times spent in the twist and chair conformations are illustrated in Fig. 9 for 219 °K (top) and 292 °K (bottom). These temperatures correspond to $T = 0.03$ and T=0.04 in dimensionless temperature units. The dwell time distribution of the twist conformation is approximately a power law or stretched exponential. The dwell time distribution of the chair has two regions: a steep decay at short times (that is possibly a power law) and a single exponential at longer times. The dwell time distributions of cyclohexane have not yet been measured over a large enough range of time scales to compare to these results. The qualitative dependence of dynamics on temperature is as expected. The average time spent in the more stable chair increases significantly at lower temperatures. It is not yet clear how to determine the physical time that corresponds to each time

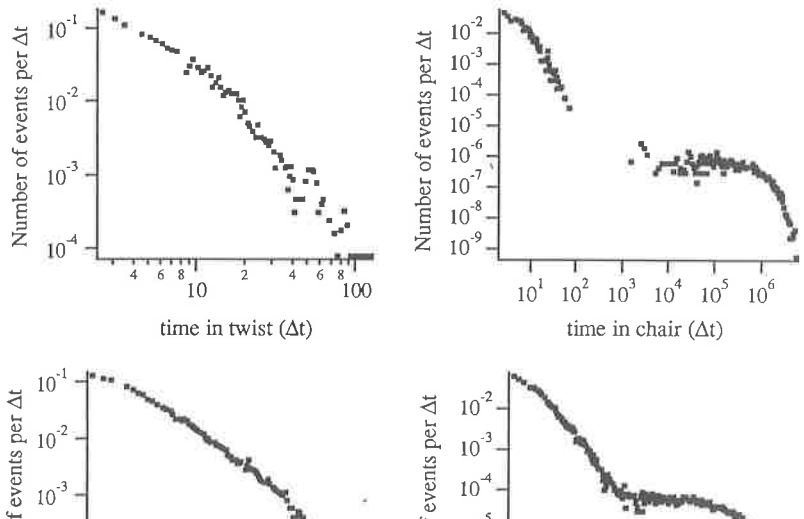


Fig. 9. Distribution of the dwell times spent in the twist and chair conformations computed from the Hopfield network representation of cyclohexane. These histograms are based on the computation of the durations of 10 000 twist and chair conformations. The distributions are shown for temperatures $T = 0.03$ (219 °K) at the top, and $T = 0.04$ (292 °K) at the bottom.

step of the neural network computation. Using the method described above that equates the thermal component of the neural network with the energy flux imparted by collisions of solvent molecules, we find that the time scale for the chair to twist transition predicted by the neural network is approximately 10^4 faster than that found from the NMR measurements [3]. It is not clear whether this means the neural network computation is in serious error or merely that we do not yet know how to scale the time steps appropriately to physical time.

5. Discussion

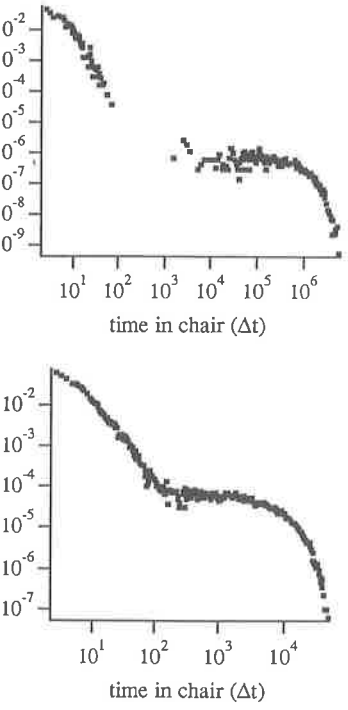
Our motivation for studying protein dynamics is not the "protein folding problem," that is, to predict the spatial structure and folding pathways from the primary sequence of amino acid residues. We are interested in how a protein switches from one stable conformational state to another. In particular, we want to understand how the motions inside a cell membrane ion channel protein causes it to switch between conformational states that are open or closed to the passage of ions. Using the patch clamp technique we can measure the sequence of times that an individual channel molecule spends in each state [31,32]. We want to understand the information about protein structure and dynamics that is conveyed to us by this data. This was the motivation for computing the dwell time distributions of cyclohexane. Interestingly, the forms of the dwell time distributions shown in Fig. 9 are the forms most commonly seen in the patch clamp data. These forms include a power law or a stretched exponential form over all time scales, or power law behavior at short times and single exponential behavior at long times [14,28,29,31,32,45].

Neural networks may prove useful in thinking about protein dynamics. The neural network representation suggests that we may have placed too much emphasis on the structure of the most stable conformational states. Forces between nearby atoms are stronger than forces between distant atoms. Hence, highly ordered structures form in small regions of the network. Each of these small regions may have different and conflicting local structures. Thus, while many nodes may have values corresponding to one stable conformational state, there will be other nodes that have values corresponding to other stable conformational states. In principle, we could enumerate all possible combinations of the values of all the nodes, and call them "states". However, this it not a useful way to think about what is happening. A more useful interpretation is to think of a protein as being approximately in one state (corresponding to one memory), although some of its parts may be in other conflicting states (corresponding to other memories).

An ion channel is open when most of the nodes of the corresponding network have values corresponding to the open conformational state. Thermal fluctuations change some nodes into the closed state. Local interactions between nodes then form small regions that are locally in the closed configuration. These small closed regions conflict with their surrounding locally open regions. The structure of the channel protein is always bulging out into the wrong states in local regions. As time goes

ne twist to another twist without first
a transition can be seen near $t = 40$.
g. 8, the molecule spends most of its
l part of it in the higher energy twist

he twist and chair conformations are
2 °K (bottom). These temperatures
nensionless temperature units. The
tion is approximately a power law or
ution of the chair has two regions: a
power law) and a single exponential
ns of cyclohexane have not yet been
cales to compare to these results. The
erature is as expected. The average
ignificantly at lower temperatures. It
l time that corresponds to each time



e twist and chair conformations computed
xane. These histograms are based on the
hair conformations. The distributions are
and $T = 0.04$ (292 °K) at the bottom.

by, more and more local regions will change into the closed conformational state. The nonlinear local interactions then latches up the channel protein structure into its new conformational state.

Neural networks may also prove useful in computing protein dynamics. Protein motions are now calculated by evaluating the force on each atom, updating its position, and then repeating this procedure many times [25,33]. This method is inefficient because the nature of the protein is not intrinsically contained in the method. The method is constantly fighting the protein, keeping the steps small so that the forces do not change much over the distances that the atoms move in each time step. However, the properties of a neural network (such as its energy structure) is similar to that of a protein. Thus, the neural network forms a natural encoding of the protein. Encoding a few essential features of the protein as the memories of neural network may thus reproduce many additional, emergent properties of the protein. We showed above that the memories of the twist and chair conformational states of cyclohexane were sufficient to generate the existence of the boat conformation and even approximately reproduce the value of its energy.

The evolution of the neural network is limited by the roughness of the energy landscape. Motions within local regions on this landscape can be accurately accomplished by the neural network in one time step, while the $F = ma$ molecular dynamics might take a very large number of small steps to integrate the positions of the atoms over physical space. The neural network encodes only some of the information about the protein. This coarseness means that the accuracy of the computed dynamics is limited. It also means that if the neural network contains the essential features of the protein, then the dynamics of these features can be computed very efficiently. The cyclohexane calculation was an important first step that was useful in illustrating and resolving some of the issues involved in this new method. However, in order to fully test the efficiency of this new method for large molecules, it will be necessary to use it to compute more complex systems, such as the gating of ionic channels or protein folding, and compare those results to that obtained from other methods.

The purpose of this paper is to present a beginning to formulating neural networks with the properties of given molecules so that the dynamics of the molecule can be computed from the dynamics of the corresponding neural network. We briefly described methods of encoding the spatial structure of the molecule into the network, appropriate types of neural networks and methods of updating. We described in detail the properties of the Hopfield network and its use to compute the twist to chair transitions of cyclohexane. We have not presented a finished method. That would be too ambitious an undertaking for one paper. Rather, we have presented a review of the possibilities and illustrated them with a specific example in order to clarify the questions that need to be answered and to explicitly state the problems that need to be solved. The best method of encoding the spatial structure, the best type of neural network to use and the best updating scheme to use remain to be determined. Perhaps the most difficult questions are: (1) How much information

about the molecule needs to be encoded and how much will arise as an emergent property of the network, and (2) What is the relationship between the time step of the neural network computation and physical time in seconds?

6. Conclusions

Neural networks are a new type of model which are useful in studying systems with many interacting pieces. Proteins have many of the properties of neural networks. Thus, neural networks may serve as a useful paradigm in thinking about molecular structure and may lead to an efficient method of computing molecular dynamics.

7. Acknowledgements

We are pleased to acknowledge the computational assistance and facilities provided by Dr John M. Hubbard, Director of the Molecular Modeling Facility for Molecular Biology at Columbia University and NSF Grant DIR-8720229.

This work was done during the tenure of an Established Investigatorship of the American Heart Association awarded to L. S. L. and was also supported in part by NIH grant EY6234 and Research to Prevent Blindness, Inc.

References

[1] Aarts E. and Korst J., *Simulated Annealing and Boltzmann Machines* (John Wiley and Sons, New York, 1989).

[2] Amit D. J., *Modeling Brain Function* (Cambridge Univ. Press, New York, 1989).

[3] Anet F. A. L. and Bourn A. J. R., Nuclear magnetic resonance studies of ring inversion in cyclohexane - d_{11}. *J. of Am. Chem. Soc.* **89** (1967) pp. 760–768.

[4] Block A., von Bloh W. and Schellnhuber H. J., Efficient box-counting determination of generalized fractal dimensions. *Phys. Rev.* **A42** (1990) pp. 1869–1874.

[5] Bohr H., Bohr J., Brunak S., Cotterill R. M. J., Lautrup B., Norskov L., Olsen O. and Petersen S., Protein secondary structure and homology by neural network – the α-helices in rhodopsin. *FEBS Lett.* **241** (1988) pp. 223–228.

[6] Bohr H. and Wolynes P. G., Initial events of protein folding from an information-processing viewpoint. *Phys. Rev.* **A46** (1992) pp. 5243–5248.

[7] Carpenter G. A. and Grossberg S., ART 2: Self-organization of stable category recognition codes for analog input patterns. *Appl. Opt.* **26**, 23 (1987) pp. 4919–4930.

[8] De Bruyn N. G., *Asymptotic Methods in Analysis*, 2nd ed (North Holland, Amsterdam, 1961).

[9] Domany E., van Hemmen J. L. and Schulten K. (eds.), *Models of Neural Networks* (Springer-Verlag, New York, 1991).

[10] Eliel E. L., Allinger N. L., Angyal S. J. and Morrison G. A., *Conformational Analysis* (Interscience, New York, 1967).

[11] Fraunfelder H., Proteins and glasses. In *Structure and Dynamics of Nucleic Acids, Proteins, and Membranes*, Clementi E. and Chin S., eds. (Plenum Press, New York, 1986) pp. 169–177.

[12] Fraunfelder H., Silgar S. G. and Wolynes P. G., The energy landscapes and motions of proteins. *Science* **254** (1991) pp. 1598–1603.

[13] Freeman J. A. and Skapura D. M., *Neural Networks: Algorithms, Applications, and Programming Techniques* (Addison-Wesley, New York, 1991).

[14] French A. S. and Stockbridge L. L., Fractal and Markov behavior in ion channel kinetics. *Can. J. Physiol. Pharmacol.* **66** (1988) pp. 967–970.

[15] Friedrichs M. S., Goldstein R. A. and Wolynes P. G., Generalized protein tertiary structure recognition using associative memory Hamiltonians. *J. Mol. Biol.* **222** (1991) pp. 1013–1034.

[16] Gantmacher F. R., *The Theory of Matrices* **2** (Chelsea Publishing Company, New York, 1959).

[17] Glauber R. J., Time dependent statistics of the Ising model. *J. Math. Phys.* **4** (1963) pp. 294–307.

[18] Hänggi P., Talkner P. and Borkovec M., Reaction rate theory: Fifty years after Kramers. *Rev. of Modern Phys.* **62** (1990) pp. 251–341.

[19] Hecht-Neilsen R., *Neurocomputing* (Addison-Wesley, New York, 1990).

[20] Hille B., *Ionic Channels of Excitable Membranes* (Sinauer Assoc., Sunderland MA, 1984).

[21] Hopfield J. J., Neural networks and physical systems with emergent collective computational abilities. *Proc. Natl. Acad. Sci. USA* **79** (1982) pp. 2554–2558.

[22] Hopfield J. J., Neurons with graded responses have collective computational properties like those of two-state neurons. *Proc. Natl. Acad. Sci. USA* **81** (1984) pp. 3088–3092.

[23] Hopfield J. J. and Tank D. W., Computing with neural circuits: a model. *Science* **233** (1986) pp. 625–633.

[24] Hou X. J., Gilmore R., Mindlin G. B. and Solari H. G., An efficient algorithm for fast O(N*ln(N)) box counting. *Phys. Lett.* **A151** (1990) pp. 43–46.

[25] Karplus M. and Petsko G. A., Molecular dynamics simulation in biology. *Nature* **347** (1990) pp. 631–639.

[26] Kohonen T., *Self-organization and Associative Memory* (Springer-Verlag, New York, 1984).

[27] Landau L. D. and Lifshitz E. M., *Statistical Physics. Course of Theoretical Physics* **5** (Pergamon Press, Oxford, New York, 1984).

[28] Liebovitch L. S., Testing fractal and Markov models of ion channel kinetics. *Biophys. J.* **55** (1989) pp. 373–377.

[29] Liebovitch L. S., Interpretation of protein structure and dynamics from the statistics of the open and closed times measured in a single ion channel protein. *J. Stat. Phys.* **70** (1993) pp. 329–337.

[30] Liebovitch L. S. and Toth T. I., A fast algorithm to determine fractal dimensions by box counting. *Phys. Lett.* **A141** (1989) pp. 386–390.

[31] Liebovitch L. S., Fischbarg J. and Koniarek J. P., Ion channel kinetics: a model based on fractal scaling rather than multistate Markov processes. *Math. Biosci.* **84** (1987) pp. 37–68.

[32] Liebovitch L. S. and Sullivan J. M., Fractal analysis of a voltage-dependent potassium channel from cultured mouse hippocampal neurons. *Biophys. J.* **52** (1987) pp. 979–988.

[33] McCammon J. A. and Harvery S. C., *Dynamics of Proteins and Nucleic Acids* (Cambridge Univ. Press, New York, 1987).

[34] McClelland J. L. and Rumelhart D. E., *Parallel Distributed Processing* **1–2** (MIT Press, Cambridge, MA, 1986).

[35] McClelland J. L. and Rumelhart D. E., *Explorations in Parallel Distributed Processing* (MIT Press, Cambridge, MA, 1989).

[36] Metropolis N., Rosenbluth A., Rosenbluth M., Teller A. and Teller E., Equation of state calculations by fast computing machines. *J. Chem. Phys.* **21** (1953) pp. 1087–1092.

[37] Mislow K., *An Introduction to Stereochemistry* (W. A. Benjamin, New York, 1965).
[38] Pickett H. M. and Strauss H. L., Conformational structure, energy, and inversion rates of cyclohexane and some related oxanes. *J. Am. Chem. Soc.* **92**, 25 (1970) pp. 7281–7290.
[39] Qian N. and Sejnowski T. J., Predicting the secondary structure of globular proteins using neural network models. *J. Mol. Biol.* **202** (1988) pp. 865–884.
[40] Rytov S. M., *Introduction into Statistical Radiophysics* (Nauka, Moscow, 1976).
[41] Sasai M. and Wolynes P. G., Molecular theory of associative memory Hamiltonian models of protein folding. *Phys. Rev. Lett.* **65** (1990) pp. 2740–2743.
[42] Schlick T. and Olson W. K., Trefoil knotting revealed by molecular dynamics simulations of supercoiled DNA. *Science* **257**, 5073 (1992) pp. 1110–1115.
[43] Schlick T. and Olson W. K., Supercoiled DNA energetic and dynamics by computer simulation. *J. Mol. Biol.* **223**, 4 (1992) pp. 1089–1119.
[44] Shakhnovich E. I., Farztdinov G., Gutin A. M. and Karplus M., Protein folding bottlenecks: a lattice Monte Carlo simulation. *Phys. Rev. Lett.* **67** (1991) 1665–1668.
[45] Stockbridge L. L. and French A. S., Characterization of a calcium-activated potassium channel in human fibroblasts. *Can. J. Physiol. Pharmacol.* **67** (1989) pp. 1300–1307.
[46] Van Kampen N. G., *Stochastic Processes in Physics and Chemistry* (North Holland, Amsterdam, 1981).
[47] Welch R. R., *The Fluctuating Enzyme* (John Wiley & Sons, New York, 1986).
[48] Wolynes P. G., Search and recognition: spin glass engineering as an approach to protein structure prediction. In *Biologically Inspired Physics*, L. Pelditi, ed. (Plenum Press, New York, 1991) pp. 15–37.
[49] Zhang P. Olson W. K. and Tobias I., Accelerated record keeping Fourier series Monte Carlo simulations of an isotropic elastic rod model of DNA. *Comput. Polymer Sci.* **1** (1991) pp. 3–17.

APPENDIX 1

Dynamics of the Evolution of Probabilistic Networks

The evolution of a probabilistic network can be described by its time dependent probability density function $p(\mathbf{S})$, where $\mathbf{S}$ is the state of the network at time t. The changing value of this function in time is determined by an equation, called the master equation (see, for example, [46]). For the discrete time steps $1, \ldots, n$, $n+1, \ldots$ where $p^n(\mathbf{S})$ is the probability density function at the nth time step, the master equation has the form

$$p^{n+1}(\mathbf{S}) - p^n(\mathbf{S}) = \sum_{\mathbf{S}' \neq \mathbf{S}} W(\mathbf{S}|\mathbf{S}')p^n(\mathbf{S}') - W(\mathbf{S}'|\mathbf{S})p^n(\mathbf{S}), \qquad \text{(A1.1)}$$

where $W(\mathbf{S}|\mathbf{S}')$ is the probability of a jump from the state $\mathbf{S}'$ to $\mathbf{S}$ during one time step.

If a system is finite and not degenerate in some sense [16], the probability density function at long times approaches a unique final equilibrium distribution. In physical systems this is distribution is the Gibbs-Boltzmann distribution, which has the form

$$p = \text{const} \cdot \exp\left(-\frac{E(\mathbf{S})}{T}\right), \qquad \text{(A1.2)}$$

where T is temperature measured in energy units. In order to reach this distribution, and in order to meet some other physical requirements, the Markov transition matrix W should satisfy the condition of detailed balance condition with the final distribution (A1.2), [46] namely:

$$W(\mathbf{S}|\mathbf{S}')p(\mathbf{S}') = W(\mathbf{S}'|\mathbf{S})p(\mathbf{S}) \tag{A1.3}$$

Substituting (A1.2) into (A1.3) we obtain

$$W(\mathbf{S}|\mathbf{S}') = W(\mathbf{S}'|\mathbf{S}) \exp\left(-\frac{\Delta E}{T}\right), \tag{A1.4}$$

where ΔE is the change of energy for the process $\mathbf{S} \rightarrow \mathbf{S}'$.

In the stochastic dynamics of one particular system the energy is not necessarily a decreasing function of time because of thermal fluctuations. We now use the probability distribution $p(\mathbf{S})$ to describe the evolution of an ensemble of these systems, rather than of one particular system. To characterize the properties of an ensemble of systems, we use the free energy F which is given by the relation:

$$F = \langle E - TS \rangle = \sum_{\mathbf{S}} p(\mathbf{S})[E(\mathbf{S}) + T \ln(p(\mathbf{S}))] \tag{A1.5}$$

Here $\langle S \rangle$ is the dimensionless entropy of a distribution.

The relations (A1.1) and (A1.4) imply that the free energy is a nonincreasing function of time.

An infinite number of possible updating methods exist that satisfy (A1.4). The choice between them is dictated by practical considerations. The most common updating method for computer simulations is asynchronous updating where only one node at each time step is considered for updating. This updating method avoids problems, such as closed cycles of length 2, that are common in synchronous updating methods where all the nodes are updated at the same time [2,9].

In the asynchronous updating method, where the value of only one node i is considered for updating at each time step, the new state of the network $\mathbf{S}'$ can differ from its previous state only in the value of that one node, and thus all the other $W(\mathbf{S}'|\mathbf{S})$ are equal to zero. The node i to be updated can be chosen in a fixed sequence, randomly, or randomly from the nodes which have not been updated in a current updating cycle of the length N. For our purpose the random updating seems to be the most appropriate because it resembles the physical events where random collisions of the solvent molecules into the protein add energy at a local point that changes the conformation of the protein.

Different functional forms of $W(\mathbf{S}'|\mathbf{S})$ are possible. The two most commonly used are the Metropolis [36] form:

$$W(\mathbf{S}'|\mathbf{S}) = \begin{cases} 1, & \text{for } \Delta E < 0 \\ \exp\left(-\dfrac{\Delta E}{T}\right), & \text{for } \Delta E > 0 \end{cases} \tag{A1.6}$$

nits. In order to reach this distribu-
requirements, the Markov transition
iled balance condition with the final

$$\mathbf{S}'|\mathbf{S})p(\mathbf{S}) \tag{A1.3}$$

$$\left(-\frac{\Delta E}{T}\right), \tag{A1.4}$$

ess $\mathbf{S} \to \mathbf{S}'$.
system the energy is not necessarily
l fluctuations. We now use the prob-
on of an ensemble of these systems,
terize the properties of an ensemble
iven by the relation:

$$(\mathbf{S}) + T \ln(p(\mathbf{S}))] \tag{A1.5}$$

ibution.
the free energy is a nonincreasing

hods exist that satisfy (A1.4). The
considerations. The most common
asynchronous updating where only
updating. This updating method
2, that are common in synchronous
ted at the same time [2,9].
re the value of only one node i is
new state of the network $\mathbf{S}'$ can
of that one node, and thus all the
e updated can be chosen in a fixed
s which have not been updated in
our purpose the random updating
sembles the physical events where
the protein add energy at a local
ein.
ossible. The two most commonly

$$\text{for } \Delta E < 0$$
$$\text{for } \Delta E > 0 \tag{A1.6}$$

and the Glauber [17] form:

$$W(\mathbf{S}'|\mathbf{S}) = \frac{1}{2}\left(1 - \tanh\left(\frac{\Delta E}{2T}\right)\right) \tag{A1.7}$$

More precisely, (A1.7) gives the element of the W matrix multiplied by N, since their is probability $(1/N)$ to pick an arbitrary node for updating. It can be shown that both (A1.6) and (A1.7) satisfy the requirement (A1.4).

The Metropolis form (A1.6) is a fast algorithm that is favored in many in numerical simulations. On the other hand, the Glauber form (A1.7) is favored in theoretical analysis because of the useful property of the tanh function that $\tanh(Sx) = S \cdot \tanh(x)$ for $S = 0, \pm 1$. Note also, that the Glauber form in thermal equilibrium (A1.2) is equivalent to assigning the value 1 to a node with a probability

$$p = \left(1 + \exp\left(\frac{\Delta E}{T}\right)\right)^{-1}, \tag{A1.8}$$

where $\Delta E = E(S_i = 1) - E(S_i = -1)$. This probability is independent of the previous value of the node. The updating method (A1.8) is the equilibrium version of Glauber dynamics. It is used in the Boltzmann machine network [1,34].

APPENDIX 2

Dependence of the Form of the Energy Distribution on Temperature

We consider here a neural network with one memory. A more general description is given in Refs. [2,9].

If the state $\mathbf{S}$ and the memory ξ have N^+ nodes with the same values and N^- nodes with opposite values, then from (3.6):

$$N^+ + N^- = N, \quad m = 2\frac{N^+}{N} - 1, \quad N^+ = \frac{N}{2}(1+m), \quad N^- = \frac{N}{2}(1-m) \tag{A2.1}$$

The energy, calculated from the overlap m of the one memory is

$$E = -\frac{1}{2}\alpha m^2 . \tag{A2.2}$$

The probability for a system to have energy E is proportional to the product of 2 components: (1) the Boltzmann probability to occupy a state with this energy and (2) the number of possible realizations of the state with this energy. According to (A2.1, A2.2) the energy depends only on m, which can be expressed using N^+. The number of possible realizations with the given energy E may be defined as the number of states with given value N^+, that is

$$C_N^{N^+} = \frac{N!}{N^+!(N-N^+)!}, \tag{A2.3}$$

where $C_N^{N^+}$ is the number of combinations of N things taken N^+ at a time. For the probability for the state of the network to have the energy E, we find that

$$p(E(m(N^+))) = \text{const} \cdot C_N^{N^+} \exp\left(-\frac{E}{T}\right). \tag{A2.4}$$

Figure 7 shows the excellent agreement between the analytical form (A2.4) and the numerical results from the computer simulation of the network.

APPENDIX 3

Geometrical Properties of an N-dimensional Cube

The values of the components of the state **S** of the network can be represented by the coordinates of the vertices of an N-dimensional cube $[-1, 1]^N$. Each vertex corresponds to one state, and the total number of states equals 2^N. Each memory is responsible for one term in the energy (A2.2) and for a corresponding potential well on the energy surface. We define a state to be in the domain of a memory if the absolute value of its overlap m_μ with this μth memory is greater than a certain value m_0:

$$|m_\mu| > m_0 \tag{A3.1}$$

The number of states of the network, where there are N^+ matches between the the values of the nodes and the values of their corresponding memory ($S_i = \xi_i$) is given by (A2.3). Therefore, using (A2.1) we find that the probability $p(m_0)$ that the condition (A3.1) is satisfied is given by

$$p(m_0) = 2^{-N} \sum_{N^+ > N_0^+} C_N^{N^+}, \qquad N_0^+ = \frac{N}{2}(1 + m_0). \tag{A3.2}$$

Thus, (A3.2) shows that $p(m_0)$ depends on N. In the limit of large N, the probability $p(m_0)$ decreases approximately exponentially with increasing N.

Note that while the relative number of states in the vicinity ($|m_\mu| > m_0$) of the μth memory trace decreases with increasing N, the absolute number of states in the domain of this memory increases. The angle φ between two adjacent states decreases according to the relationship:

$$\varphi = \arccos\left(1 - \frac{2}{N}\right) \approx \frac{2}{N^{1/2}} \tag{A3.3}$$

The number of nodes N also determines the variety of possible directions to leave a potential valley. In terms of proteins, that may mean that the number of states near a stable conformational state may therefore depend on the size of a protein and its number of important degrees of freedom.

APPENDIX 4

Diffusion Coefficient of the Probability Density Function in the Space of the Overlaps

We now show, that for large numbers of nodes N, the evolution of the state of the network can be considered as a diffusion process in the space of overlaps. We will

derive the expression for the diffusion coefficient for the Glauber dynamics case. For simplicity we will analyze the case where the network has only one memory. We will use a memory of the form $\xi = (1, 1, \ldots, 1)$. A single memory of another form, yields exactly the same result, because we can rotate the N-dimensional space to transform that second memory into the form of the first memory. A more detailed description for several memories is given in Ref. [9].

First we note, that with number of nodes N, the overlap m with a single memory pattern can have only a discrete number of $(N+1)$ values given by

$$m = -1,\ -1+\varepsilon, \ldots, 1. \qquad \varepsilon = \frac{2}{N}. \tag{A4.1}$$

Note, that changing the value of one node changes the value of m by $+\varepsilon$.

We now consider the probability ρ_m that the overlap of the network has the value m. The Markov equation for the evolution of this probability density function is given by

$$\rho_m^{n+1} = (\rho^n p_+)_{m-\varepsilon} + (\rho^n p_0)_m + (\rho^n p_-)_{m+\varepsilon}\,. \tag{A4.2}$$

Here p_+, p_0 and p_- are respectively the probabilities to increase, keep and decrease the overlap by ε per one time step, and n stands for the total number of elapsed time steps.

We now evaluate p_+, p_0 and p_- as functions of the overlap m. If one particular node S_i is chosen, then in Glauber dynamics (A1.7, 3.12) the probability to keep the value of the node unchanged is given by

$$\frac{1}{2}\left(1-\tanh\left(\frac{\Delta E}{2T}\right)\right) = \frac{1}{2}\left(1-\tanh\left(\frac{\alpha m S_i}{NT}\right)\right) = \frac{1}{2}\left(1-S_i\,\tanh\left(\frac{\alpha m}{NT}\right)\right). \tag{A4.3}$$

The increase in m takes place when we change the value of the node, which was in a state $S_i = -1$. The probability that this is the node that we will choose for updating is equal to N^-/N. Thus, using (A2.1) we obtain:

$$\begin{aligned} p_+ &= \frac{N^-}{N}\frac{1}{2}\left(1+\tanh\left(\frac{\alpha m}{NT}\right)\right) = \frac{(1-m)}{4}\left(1+\tanh\left(\frac{\alpha m}{NT}\right)\right) \\ p_- &= \frac{N^+}{N}\frac{1}{2}\left(1-\tanh\left(\frac{\alpha m}{NT}\right)\right) = \frac{(1+m)}{4}\left(1-\tanh\left(\frac{\alpha m}{NT}\right)\right) \\ p_0 &= 1-p_+-p_- \end{aligned} \tag{A4.4}$$

Substituting this into (A4.2), we find that

$$(\rho^{n+1}-\rho^n)_m = (\rho^n p_+)_{m-\varepsilon} - (\rho^n p_+)_m + (\rho^n p_-)_{m+\varepsilon} - (\rho^n p_-)_m\,.$$

On the left hand side we have the time difference of a value, and on the right hand side there are two "space" differences. We now make a Taylor expansion of both

sides. The left hand side we expand in terms of the powers of the time step τ. The right hand side we expand in terms of the powers of the "space" step ε:

$$\sum_{k=1}^{\infty} \frac{\tau^k}{k!} \frac{\partial^k \rho}{\partial t^k} = \sum_{k=1}^{\infty} \frac{\varepsilon^k}{k!} \frac{\partial^k}{\partial m^k} [\rho(p_- + (-1)^k p_+)]$$

Assuming that ε is small and neglecting high order terms:

$$\frac{\partial \rho}{\partial t} = -\frac{\partial}{\partial m}\left(\frac{\varepsilon}{2\tau}\left(\tanh\left(\frac{\alpha m}{NT}\right) - m\right)\rho\right) + \frac{\varepsilon^2}{4\tau}\frac{\partial}{\partial m}\left(\left(1 - \tanh\left(\frac{\alpha m}{NT}\right)\right)\frac{\partial \rho}{\partial m}\right) \tag{A4.5}$$

This expression has the form of the Fokker-Planck equation [46]. Note that the coefficient under the derivative in the second term at the right hand side is always positive and is greater than $(1 - |m|)$ for m in $[-1, 1]$. Therefore, the diffusion coefficient is always positive and of order of magnitude

$$D \approx \frac{\varepsilon^2}{4\tau}$$

or, recalling (A4.1):

$$D = \frac{1}{\tau N^2}, \qquad \tau = \frac{1}{DN^2}. \tag{A4.6}$$

APPENDIX 5

Diffusion Coefficient of the Evolution of the Physical Structure

We can relate the time step in the neural network simulation to the physical time by equating the diffusion coefficient of the changing state of the neural network in the previous section with the diffusion coefficient based a physical description of the thermal fluctuations in the molecule. Thus, in this section we seek an estimate of the diffusion coefficient of the changes in structure due to the thermal energy supplied to the molecule by collisions with the solvent molecules. For simplicity, and to give us an insight into the method itself, we use the highly simplified case of one variable.

The equation of motion for one degree of freedom of a thermally fluctuating molecule can be written in the Langevin form [46]:

$$M\ddot{x} + \gamma\dot{x} + \frac{\partial U}{\partial x} = f_L(t) \tag{A5.1}$$

Here M is the effective mass of a given degree of freedom, γ is the attenuation coefficient, U is the potential function, x is the space coordinate, $f_L(t)$ is the random Langevin force. The dots above the variable x denote time derivatives.

When the surrounding media which causes thermal fluctuations is dense enough, then the first inertial term in (A5.1) is much smaller than the second dissipative

term. Thus, we neglect the first inertial term, and consider the regime dominated by spatial diffusion [18]. We also replace the space variable x by the dimensionless variable m:

$$m \approx \frac{x}{l}\,, \tag{A5.2}$$

where l is a length characterizing the distance between different stable conformations. The variable m is similar to the dimensionless overlap in the network model.

The Langevin equation (A5.1) can now be rewritten as:

$$\dot{m} = -\gamma^{-1}\frac{\partial U}{\partial m} + l\gamma^{-1} f_L(t) \tag{A5.3}$$

If $f_L(t)$ is δ-correlated stochastic noise with intensity F,

$$\langle f_L(t) f_L(t')\rangle = F\delta(t-t')\,, \tag{A5.4}$$

then the expression (A5.3) is equivalent to the Fokker-Planck equation [46] for the probability density function $p(m)$:

$$\dot{p} = \gamma^{-1}\frac{\partial}{\partial m}\left(p\frac{\partial U}{\partial m} + \frac{Fl^2}{2\gamma}\frac{\partial p}{\partial m}\right) \tag{A5.5}$$

The stationary solution of this equation is

$$p = \text{const}\cdot\exp\left(-\frac{2\gamma}{Fl^2}U\right), \tag{A5.6}$$

which should be equal to the Boltzmann distribution $p = \text{const}\cdot\exp(-U(x)/T)$. Equating these two expressions we obtain

$$\gamma = \frac{l^2 F}{2T}\,. \tag{A5.7}$$

This relation between γ and F shows how the friction arises from thermal collisions. It is connected with the Einstein relation for diffusion and mobility, and it is also a consequence of fluctuation-dissipative theorem. Substituting (A5.7) into (A5.5),

$$\dot{p} = \frac{2T^2}{Fl^2}\frac{\partial}{\partial m}\left(T^{-1}p\frac{\partial U}{\partial m} + \frac{\partial p}{\partial m}\right). \tag{A5.8}$$

From this diffusion equation we define the diffusion coefficient D,

$$D = \frac{2T^2}{Fl^2}\,. \tag{A5.9}$$

In order to estimate the value of D we must know the value F of the intensity of the random force. The Langevin force (A5.4) is the result of frequent random collisions between our molecule and the molecules of the solvent. The average absolute value

of the force experienced by the fluctuating molecule due to one collision is equal to the change of momentum of a molecule of the solvent:

$$|a| \approx \frac{2\mu v_x}{\vartheta}, \tag{A5.10}$$

where μ is the mass of the solvent molecule, v_x is the component of thermal velocity for a μ-molecule in the direction of collision and ϑ is the average duration of each collisions. The average value $\langle a \rangle = 0$. The total random force is the sum of all the forces from the random collisions. For an impulse of this kind [40] the correlation function is given by

$$\psi(t) = n_1 \langle a^2 \rangle \vartheta \cdot \exp\left(-\frac{|t|}{\vartheta}\right), \tag{A5.11}$$

where n_1 is the frequency of collisions. For short correlation times ϑ, we approximate the exponential function by a δ function of t with the same integral. The coefficient F of this δ-function is given by

$$F = 8 n_1 \mu^2 v_x^2 . \tag{A5.12}$$

Using the equality (see, for example [27]) $\mu v_x^2/2 = T/2$, $n_1 \approx v n \sigma \approx (3T/\mu)^{1/2} n \sigma$, where v is the thermal velocity, n is particle density of the solvent, and σ is cross section of interaction between our molecule and molecules of the solvent, T is the temperature measured in energy units, we obtain from (A5.9), (A5.12):

$$D \approx \frac{T^{1/2}}{4\sqrt{3}\, l^2 \sigma n \mu^{1/2}} \tag{A5.13}$$