

Thermodynamic Neural Network

Todd Hylton 

Department of Electrical and Computer Engineering, University of California, San Diego, CA 92093, USA; thylton@ucsd.edu

Received: 15 January 2020; Accepted: 20 February 2020; Published: 25 February 2020



Abstract: A thermodynamically motivated neural network model is described that self-organizes to transport charge associated with internal and external potentials while in contact with a thermal reservoir. The model integrates techniques for rapid, large-scale, reversible, conservative equilibration of node states and slow, small-scale, irreversible, dissipative adaptation of the edge states as a means to create multiscale order. All interactions in the network are local and the network structures can be generic and recurrent. Isolated networks show multiscale dynamics, and externally driven networks evolve to efficiently connect external positive and negative potentials. The model integrates concepts of conservation, potentiation, fluctuation, dissipation, adaptation, equilibration and causation to illustrate the thermodynamic evolution of organization in open systems. A key conclusion of the work is that the transport and dissipation of conserved physical quantities drives the self-organization of open thermodynamic systems.

Keywords: self-organization; open thermodynamic systems; neural networks; dissipative adaptation; causal learning; multiscale complex systems; thermodynamic evolution

1. Introduction

Applying concepts from thermodynamics and statistical physics to neural network models has a relatively long history, much of it centered on models of interacting spins on a lattice Ising model. The dynamics of these systems near critical points [1,2] have been studied to derive, for example, magnetic response functions using mean field approximations. The statistical properties of randomly disordered Ising models—spin glasses—are also well studied and understood in limiting cases including infinite range interactions [3]. These ideas were extended to networks capable of storing memories as attracting states recalled using only a portion of the initial memory—Hopfield Networks [4]. The statistical mechanics of these networks have been developed in detail, particularly regarding their capacity to store and recall memories [5–8]. Further development of these ideas to include “hidden” spin states not directly determined by a set of training vectors and thermodynamically inspired techniques for training these networks was captured in a class of models called Boltzmann Machines [9,10]. Statistical physics also found application in non-Ising neural network models including layered networks [11], complex network model spaces [12,13], and directed Markovian neuronal networks [14]. The motivations and techniques employed in this work borrow heavily from this history and the work on Ising models in particular.

As compared to these earlier works, the Thermodynamic Neural Network (TNN) model presented here is distinguished by its electric circuit inspiration and the incorporation of physical concepts, such as charge conservation, potential diffusion, reaction kinetics and dissipation-driven adaptation. In particular, the transport of a conserved quantity did not appear as a primary concern in these earlier works. Although learning is the primary objective of the model, the motivation is not the statistical generalization of a training set, the replication of a function, or the storage of a memory; rather, learning is viewed as an adaptation to improve equilibration with external potentials and a thermal reservoir.

The larger inspiration for this work is the long standing hypothesis that the evolution of the natural world, including life, is driven by thermodynamics and constrained by the laws of physics [15,16]. In particular, the model addresses the case of external boundary potentials that vary slowly compared to the characteristic equilibration time of the network and the resulting evolution of the network to minimize internal entropy production [17], while creating entropy at the boundaries via the transport of conserved quantities [18]. These are the essential physics that drive the self-organization of the network. A particular contribution of this work is the recognition that the selective internal dissipation of conserved quantities is the means by which the system “learns” without “forgetting” earlier configurations that were effective under other boundary potentials. Initially, this work was inspired by experiments on collections of metallic balls in oil that self-assemble to create electrical connections when subject to external potentials [19]. Although the primary context for the work is physics and thermodynamics, we were also inspired by ideas from complex systems, chemistry, neurobiology, computation and cognitive science.

2. Results

The model comprises a collection of nodes connected by symmetrically weighted edges in contact with a thermal reservoir and driven by a collection of external biasing nodes through which external potentials may be applied. A network of urban streets, intersections and traffic signals can serve as a useful (but imperfect) analogy to help motivate and understand the model. Just as the goal of the traffic system is to maximize the flow of vehicles, subject to constraints in the traffic network, the goal of TNN is to maximize the flow of charge, subject to constraints embedded in the model (Section 2.2.1). Just as traffic signals control the flow of conflicting traffic by alternately routing vehicles with a series of decisions (“red” and “green” lights), the nodes in the TNN control the flow of conflicting charge by alternately routing it with a series of state decisions (Section 2.2.3). In each case, certain physical quantities (vehicles and charge) must be conserved, which requires that while one conserved group is transported, conflicting groups must be allowed to accumulate so that they can be transported by later decisions (Section 2.2.2). At a signal intersection, for example, vehicles travelling north—south may flow unimpeded, while vehicles travelling east—west accumulate until the signal changes. Constraints in the TNN create a similar situation at the nodes. Just as the streets in the traffic system connect the signals and have a capacity to carry vehicles depending on their width, the edges in the TNN connect the nodes and have a capacity to carry charge depending on their weight. Just as the traffic network needs to be responsive to changes in its external inputs (e.g., rush hour inflow or outflow to the suburbs), the TNN needs to be responsive to changes in charge injected into it from external nodes (Section 2.2.6). Just as the efficient operation of the overall traffic network depends upon the coordinated decision making of the signals in the short term, and in the adaption of the streets in the long term, the efficient operation of the TNN depends upon the coordinated decision making of the nodes in the short term and the adaptation of the edge weights in the long term (Section 2.2.5). With this analogy in mind, we now proceed to detail the TNN. Section 2.1 presents a qualitative description of the concepts that underpin the model, which in Section 2.2 are described mathematically. Computer simulations implementing the model are described in Section 2.3.

2.1. Model Concepts

Referring to Figure 1, we compare the TNN model with the well known Ising model. The Ising model is a network of nodes in which the node states e_i , typically conceived as “spins”, interact via symmetric weights w_{ij} . Low-energy states are those that align the node state e_j with the net total of the weighted interactions with its connected nodes, which can be viewed as accumulating all of the interactions $\sum_i e_i w_{ij}$ into a single compartment. The TNN model is also a network of nodes in which the node states e_i , conceived as electrical potentials, interact via symmetric weights w_{ij} , but nodes interact via the exchange of charge and low-energy node states are those that effectively transport charge among the node’s inputs while independently conserving both positive and negative charges.

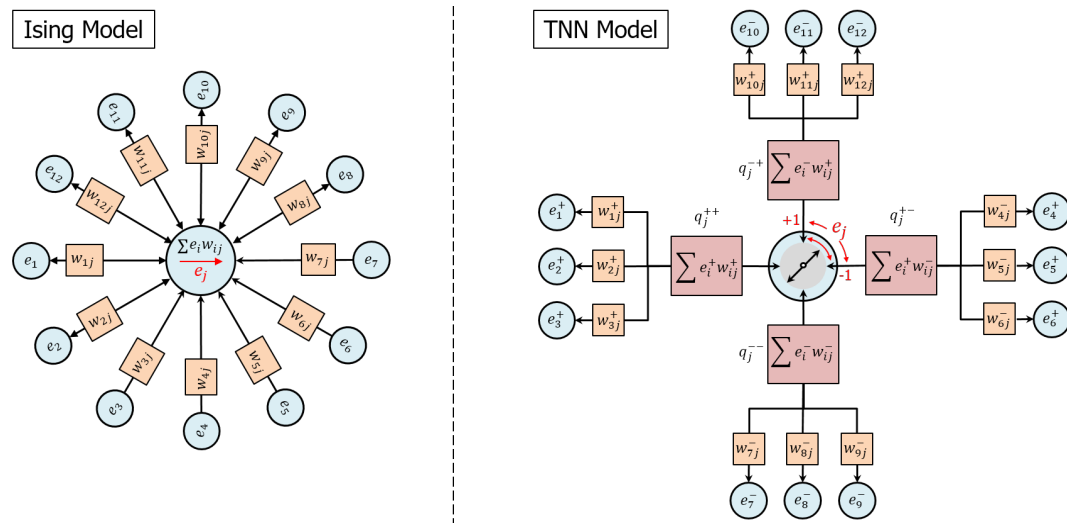


Figure 1. (left) The Ising model is a network of nodes in which the node states (“spins”) e_i interact via symmetric weights w_{ij} . Low-energy states are those that align the node state e_j with the net total of the weighted interactions. (right) The TNN model is also a network of nodes in which the node states (“potentials”) e_i interact via symmetric weights w_{ij} , but nodes interact via the exchange of charge and low-energy node states are those that effectively transport charge among the node’s inputs while independently conserving positive and negative charge. Node state selection is a competition to connect two different pairs of compartments having opposite charge polarity and the same node input state polarity, illustrated as rotating switch.

Referring now to Figure 1(right), the following list describes the concepts whereby the TNN model communicates and updates network node states:

- A node j is characterized by a state e_j representing a potential. In general, the model supports any number of values of the node state on the interval $e_j \in [-1, 1]$. For example, a binary node might assume a state $e_j \in \{-1, 1\}$.
- An edge connecting nodes i and j is characterized by a real, symmetric weight, w_{ij} , describing a capacity to transport charge.
- Node i may apply a potential e_i to an edge weight w_{ij} and generate an edge charge $q_{ij} = e_i w_{ij}$ that becomes the input to node j . Similarly, node j may apply a potential e_j to the same edge weight and generate an edge charge $q_{ji} = e_j w_{ij}$ that becomes the input to node i . In order to clarify the relationship between potentials and edges, we may sometimes designate potentials with two subscripts as $e_{ij} = e_i$ and $e_{ji} = e_j$.
- Positive and negative charges are independently conserved (i.e., they never sum to cancel each other) and communicate potential along which charges of opposite polarity should flow. By this means, externally applied potentials are able to diffuse through the network and connect to complementary potentials.
- For a given node j , charge conservation requires the aggregation of input node charges into four compartments, $q_j^{\pm\pm} = \sum_i e_i^{\pm} w_{ij}^{\pm}$, distinguished by the signs of potentials e_i^{\pm} and weights w_{ij}^{\pm} that create the charge. Depending on the inputs, anywhere from one to four compartments may be populated with charge at the time of the node state decision.
- Node state selection optimizes the transfer of charge between pairs of competing compartments using Boltzmann statistics, illustrated as a “switch” in Figure 1.
- If two nodes are connected by an edge, then configurations in which the nodes have opposite potentials will be favored. Hence, if the nodes are arranged on a regular grid and connected locally, then domains of “anti-ferromagnetic” order typically emerge.

Referring now to Figure 2(left), node state selection results in a “setting of the switch” that connects a pair of *selected* compartments and relieves accumulated charge by transporting complementary charges through the compartments as output to their connected nodes. In this way, positive and negative charges are conserved and complementary conduction pathways are created through the network. The following list describes these ideas in more detail:

- The node state decision selects and connects two compartments and transfers complementary charge between them, while leaving unselected compartments disconnected. If the compartments are populated such that no complementary pairs exist, then the state selection becomes uniformly random. *Residual charge* is that portion of the charge remaining on the selected compartments after charge transfer, which is, in general, unavoidable, owing to the thermal fluctuations in the network.
- When a node is near equilibrium, residual charge in the selected compartments is dissipated to the thermal reservoir via updates to the associated edge weights according to a Boltzmann distribution. In general, these updates improve the ability of the node to transport charge among the selected compartments if similar conditions are encountered in the future.
- Edge weights associated with unselected compartments are not updated and the charge accumulated in the unselected compartments is retained. In this way, the node retains memory of “contrary” inputs (charges) and correlations (weights) that may be important to future decisions, which is intended to address the long standing “forgetting problem” in artificial neural networks.

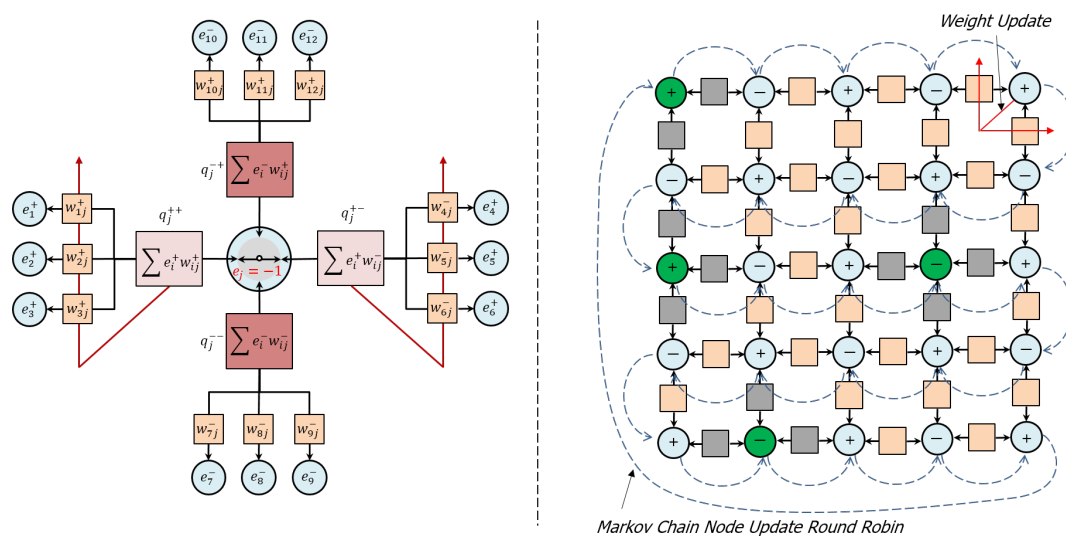


Figure 2. (left) Node state selection results in a “setting of the switch” that connects a pair of *selected* compartments (light pink) and relieves accumulated charge while *unselected* compartments (dark pink) retain their accumulated charge. (right) A network of internal, adapting nodes (light blue) and external, biased nodes (green). A round robin Markov chain method (dashed blue arrows) continuously updates the node states, while edge states update only when a node is near equilibrium (red arrows) and executes an irreversible update (see Section 2.2.5). During these irreversible updates, edge weights associated with the selected compartments adapt through the *dissipation* of residual compartment charge (red arrows, left and right), while edge weights associated with unselected compartments are not updated. In order to transport complementary charge, node states favor connections to nodes with opposite polarities (nearest neighbor network shown).

Referring now to Figure 2(right), A network of internal, adapting nodes and external, biased nodes that inject charge into the network is updated via a round robin Markov chain method that continuously updates node states, while edge weights update only when a node is near equilibrium. The following list describes these ideas in more detail:

- Node states are updated in a continuous round-robin Markov chain, which guarantees that at the time of updating any particular node all the other nodes in the network have already updated and thereby preserves temporal consistency within the network interactions.
- Updates may be either reversible or irreversible, depending on the node's state of equilibration at the time of the update. A node can determine its state of equilibration by examining the fluctuations in its energy over time. If those fluctuations are small compared to the temperature, then the node may be deemed to be equilibrated and vice-versa.
- A reversible update, which happens when the node is non-equilibrated, corresponds to the node temporarily updating its compartments with new input charges and communicating its state to its connected nodes without updating edge weights. The purpose of the reversible update is to generate fluctuations in the network that explore its configuration space to search for an equilibrium without destroying its previously acquired structure.
- An irreversible update, which happens when the node is equilibrated, corresponds to the node permanently updating its compartments with new input charges, updating the edge weights as described above, and communicating its state to its connected nodes. The purpose of the irreversible update is to adapt the network to make it more effective at transporting charge in the future.
- This method of updating the network creates a continuous cycle of fluctuation, equilibration, dissipation and adaptation that connects and refines features at large spatial/short temporal scale (i.e., the collection of network node states), intermediate spatial/intermediate temporal scale (i.e., compartment charges) and small spatial/long temporal scale (i.e., edge weights) as the network evolves. In this way, the network can rapidly equilibrate to large-scale changes in its environment through the continuous refinement of its multiscale, internal organization.
- A range of network topologies are possible, including multi-dimensional grid networks with near-neighbor connectivity, probabilistically connected, gridded networks with a metric that determines the probability of connection, and random networks. In general, there is no imposition of hierarchy or "layers" upon the network, as is common in most neural network models, but these kinds of networks can also be supported. Because connected nodes are driven to orient anti-ferromagnetically, most network configurations are inherently "frustrated", in that the nodes cannot find a way to satisfy this orientation with all their connected nodes. For a special class of networks that are partitioned into two groups (bi-partitioned networks), in which nodes of one partition can connect only to nodes in the opposite partition, this frustration can be avoided. Nearest neighbor grid networks are inherently bi-partitioned and are also attractive to study because they are easy to visualize.

As will be discussed in Section 2.3, isolated networks exposed only to a thermal bath can spontaneously order in ways reminiscent of solids, liquids and gases, while networks with externally biased nodes can self-organize in order to efficiently transport charge through the network.

2.2. Model Details

2.2.1. Network Model

The energy of network H_N is the sum of the node energies

$$H_N(e, w, q) = \sum_{j=1}^n H_j(e_j, w_j, q_j), \quad (1)$$

where

- e refers to the set of node potentials $\{e_1, e_2 \dots e_n\}$,
- w refers to the set of edge weights $\{w_1, w_2 \dots w_n\}$,

- w_j refers to the set of weights $\{w_{1j}, w_{2j}, \dots, w_{m,j}\}$ associated with node j ,
- q refers to the set of edge charges $\{q_1, q_2, \dots, q_n\}$,
- q_j refers to the set of edge charges $\{q_{1j}, q_{2j}, \dots, q_{m,j}\}$ associated with node j ,
- n refers to the number of nodes in the network, and
- n_j refers to the number of edges associated with node j .

The network is assumed to be in contact with a thermal reservoir of inverse temperature β , and the probability of a network state is assumed to follow Boltzmann statistics.

$$P_N(e, w, q) = \frac{\exp(-\beta H_N(e, w, q))}{\sum_{ewq} \exp(-\beta H_N(e, w, q))} \quad (2)$$

In the following subsections, we develop a method to evolve the network toward the equilibrium of Equation (2) (even while the network is exposed to time varying external potentials) via a combination of reversible and irreversible updates, as previously described. Reversible updates do not modify the edge states w and q and, thereby, sample the space of fluctuations in e without modifying Equation (2). Conversely, irreversible updates do modify the edge states w and q and, thereby, also modify Equation (2).

2.2.2. Compartment Model

The charge conservation, complementary conduction pathways and state selection concepts previously described require the segregation of edge charges and weights according to the polarities of the potentials and weights that generate the charge. We therefore segregate the edge charges as

$$q_{ij}^{\pm\pm} = e_i^{\pm} w_{ij}^{\pm\pm}, \quad (3)$$

where the superscripts refer to the polarity of the potential and weights associated with edge ij . Thus, for example, e_i^+ refers to a positive input potential from node i , q_{ij}^{+-} refers to an edge charge associated with a positive input potential from node i and a negative weight connecting nodes ij , and w_{ij}^{-+} refers to an edge weight associated with a negative input potential from node i and a positive weight connecting nodes ij . In the convention used here, where there are two polarity superscripts; the first refers to the sign of the input potential and the second to the sign of the edge weight. Edge charges are allowed to accumulate over multiple time steps, implying that a single edge may have more than one polarity-segregated charge and weight at a given time. The total charge that may accumulate on an edge is limited by the maximum node potentials $e_i^{\pm} = \pm 1$, which in Equation (3) may be realized as $e_i^+ = \min\{\sum_t e_i(t), 1\}$ and $e_i^- = \max\{-\sum_t e_i(t), -1\}$.

We further aggregate input charge, potential and weights into compartments by summing over the edges with common polarities of their input voltages and weights, thereby defining the following compartment input charges

$$q_j^{\pm\pm} = \sum_i q_{ij}^{\pm\pm} = \sum_{i \in i^{\pm\pm}} q_{ij},$$

where $i^{\pm\pm}$ designates sets of edge indices with \pm input potentials and \pm edge weights, respectively. Correspondingly, aggregating the output edge charges $p_{ij} = e_j w_{ij}$ from node j after state selection e_j , yields compartment output charges as

$$p_j^{\pm\pm} = e_j w_j^{\pm\pm}$$

where

$$w_j^{\pm\pm} = \sum_i w_{ij}^{\pm\pm} = \sum_{i \in i^{\pm\pm}} w_{ij}.$$

2.2.3. Node Model

The choice of the node energy equation is driven by four considerations

- minimizing residual charge on the node
- maximizing charge transport through the node
- avoiding attractors to node states $e_j = 0$
- employing “kinetic” factors to sharpen state decisions and direct residual charge dissipation and accumulation processes

After much experimentation, the following expression for the node energy was selected

$$H_j(e_j) = (f_j^+)^2[(q_j^{++} - p_j^{+-})^2 - (q_j^{++} + p_j^{+-})^2 + (q_j^{+-} - p_j^{++})^2 - (q_j^{+-} + p_j^{++})^2] \\ + (f_j^-)^2[(q_j^{--} - p_j^{-+})^2 - (q_j^{--} + p_j^{-+})^2 + (q_j^{-+} - p_j^{--})^2 - (q_j^{-+} + p_j^{--})^2] \quad (4)$$

The first, third, fifth and seventh terms are associated with minimizing residual charge on the node. The second, fourth, sixth and eighth terms are associated with maximizing charge transport. The avoidance of attractors at $e_j = 0$ is addressed by the pairs of positive and negative terms that cancel terms in $(q_j^{\pm\pm})^2$ and $(p_j^{\pm\pm})^2$. $f_j^\pm = f_j^\pm(e_j)$ are the kinetic factors, which we describe later. Expanding and collecting terms yields

$$\frac{H_j(e_j)}{4} = -(f_j^+)^2(q_j^{++}p_j^{+-} + q_j^{+-}p_j^{++}) - (f_j^-)^2(q_j^{--}p_j^{-+} + q_j^{-+}p_j^{--}) \quad (5)$$

The terms in Equation (5) are suggestive complementary pairs of “forces” $q_j^{\pm\pm}$ and “fluxes” $p_j^{\pm\mp}$, which are familiar from thermodynamics. We rewrite Equation (5) as

$$\frac{H_j(e_j)}{4} = -(f_j^+)^2(q_j^{++}w_j^{+-} + q_j^{+-}w_j^{++})e_j - (f_j^-)^2(q_j^{--}w_j^{-+} + q_j^{-+}w_j^{--})e_j \quad (6)$$

and recognize that each term in the node energy promotes the transfer of charge among compartments with opposite charge polarity, and the node state selection is a competition between two pairs of compartments distinguished by the sign of their input potential. The positive potential/positive weight/positive charge compartment, q_j^{++} , and the positive potential/negative weight/negative charge component, q_j^{+-} , form a pair that favors states $e_j < 0$, while the negative potential/negative weight/positive charge compartment, q_j^{--} , and the negative potential/positive weight/negative charge component, q_j^{-+} , form a pair that favors states $e_j > 0$. We note each term in Equation (6) is the product of three factors: two factors that are determined by the compartment inputs, $q_j^{\pm\pm}$ and $w_j^{\pm\pm}$, and one that is determined by the node state, e_j . This enables the TNN to connect complementary potentials and update weights without reliance on carefully engineered network architectures and post hoc error assignment techniques like back-propagation, as will be explained below.

The kinetic factors of Equation (6) focus the updates to state variables in one of the compartment pairs. The physical inspiration for the kinetic factors is that the state selection process, while “exciting” the reaction of the selected compartments, also “inhibits” the reaction of the unselected compartments. A more colloquial interpretation is that the node can “address only one thing at a time” while “saving other things for later”. We chose the name “kinetic” to connote the idea that this factor decides which compartments of charge should and should not “move” through the node. We experimented with different kinetic factors, but found the following rectifying function to be effective

$$f_j^\pm(e_j) = h(\mp e_j), \quad (7)$$

where h is the Heaviside step function. The energy landscape of the node using the kinetic factor of Equation (7) is shown in Figure 3, illustrating how the kinetic factors sharpen the state selection.

As shown in the next section, kinetic factors also focus weight and charge updates on selected compartments, while protecting weight and charge associated with the unselected compartments.

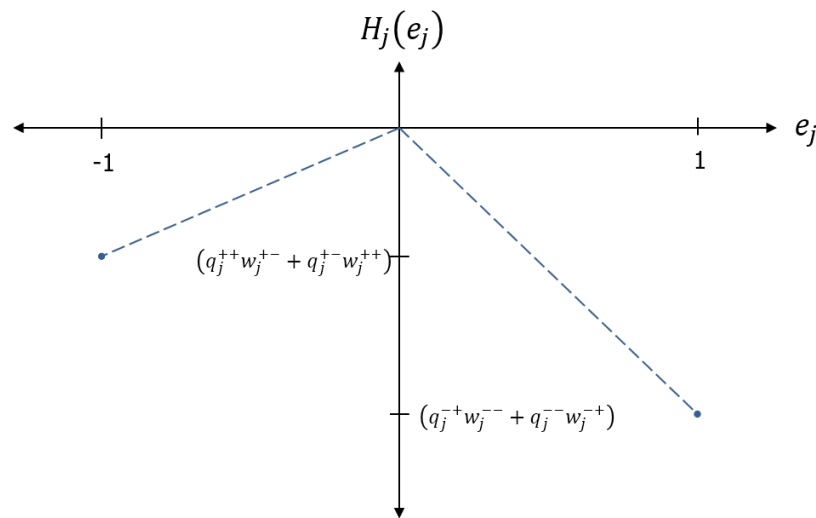


Figure 3. The energy landscape of a node using Heaviside kinetic factors, which promote state selection near $e_j = \pm 1$. The energy is piecewise linear in the state, which is a convenient choice that allows a simple spacing of node energies by a characteristic energy scale (temperature).

2.2.4. Edge Model

For nodes executing an irreversible update, residual charge on the selected compartments is used to adapt the weights, such that the node is more effective when encountering similar inputs in the future. Selecting the residual charge terms from Equation (4), we write the residual node energy after state selection as

$$\begin{aligned} H_j(q_j, w_j | e_j) &= (f_j^+)^2 [(q_j^{++} - e_j w_j^{+-})^2 + (q_j^{+-} - e_j w_j^{++})^2] \\ &\quad + (f_j^-)^2 [(q_j^{--} - e_j w_j^{--})^2 + (q_j^{--} - e_j w_j^{--})^2] \\ &= (-\delta_j^{++})^2 + (-\delta_j^{+-})^2 + (-\delta_j^{--})^2 + (-\delta_j^{--})^2 \end{aligned} \quad (8)$$

where

$$-\delta_j^{\pm\pm} = f_j^{\pm} (q_j^{\pm\pm} - e_j w_j^{\pm\mp}) \quad (9)$$

are the residual compartment charges that we wish to minimize through weight updates $w_{ij} \rightarrow w_{ij} + \Delta w_{ij}$. We capture this objective by rewriting Equation (8) as

$$\begin{aligned} H_j(\Delta w_j) &= \left(\sum_{i^{++}} e_i^+ \Delta w_{ij}^{++} - \sum_{i^{+-}} e_j \Delta w_{ij}^{+-} - \delta_j^{++} \right)^2 + \left(\sum_{i^{+-}} e_i^+ \Delta w_{ij}^{+-} - \sum_{i^{++}} e_j \Delta w_{ij}^{++} - \delta_j^{+-} \right)^2 \\ &\quad + \left(\sum_{i^{--}} e_i^- \Delta w_{ij}^{--} - \sum_{i^{--}} e_j \Delta w_{ij}^{--} - \delta_j^{--} \right)^2 + \left(\sum_{i^{--}} e_i^- \Delta w_{ij}^{--} - \sum_{i^{--}} e_j \Delta w_{ij}^{--} - \delta_j^{--} \right)^2 \end{aligned} \quad (10)$$

As will be explained below, we elect to distribute the residual charges to the edges and rewrite Equation (10) as

$$\begin{aligned}
H_j(\Delta w_j) = & \left(\sum_{i^{++}} (e_i^+ \Delta w_{ij}^{++} - \eta_{ij}^{++}) - \sum_{i^{+-}} (e_j \Delta w_{ij}^{+-} - \mu_{ij}^{+-}) \right)^2 \\
& + \left(\sum_{i^{+-}} (e_i^+ \Delta w_{ij}^{+-} - \eta_{ij}^{+-}) - \sum_{i^{++}} (e_j \Delta w_{ij}^{++} - \mu_{ij}^{++}) \right)^2 \\
& + \left(\sum_{i^{--}} (e_i^- \Delta w_{ij}^{--} - \eta_{ij}^{--}) - \sum_{i^{-+}} (e_j \Delta w_{ij}^{-+} - \mu_{ij}^{-+}) \right)^2 \\
& + \left(\sum_{i^{-+}} (e_i^- \Delta w_{ij}^{-+} - \eta_{ij}^{-+}) - \sum_{i^{--}} (e_j \Delta w_{ij}^{--} - \mu_{ij}^{--}) \right)^2
\end{aligned} \tag{11}$$

where

$$\begin{aligned}
\eta_{ij}^{\pm\pm} &= \frac{|e_i|}{\Delta_j^{\pm\pm}} \cdot \delta_j^{\pm\pm} \\
\mu_{ij}^{\pm\mp} &= \frac{|e_j|}{\Delta_j^{\pm\mp}} \cdot \delta_j^{\pm\mp} \\
\Delta_j^{\pm\pm} &= \sum_{i^{\pm\pm}} |e_i| + \sum_{i^{\pm\mp}} |e_j|
\end{aligned} \tag{12}$$

The terms $\Delta_j^{\pm\pm}$ represent the aggregated compartment input potentials and their paired output compartment potentials. The residual charge distributions to the edges, $\eta_{ij}^{\pm\pm}$ and $\mu_{ij}^{\pm\mp}$, are apportioned by their respective input potential's contribution to the $\Delta_j^{\pm\pm}$. In other words, the residual charge remaining on the selected compartments is distributed as “error” to their edge weights in proportion to the magnitude of their corresponding input potentials. This distribution of charge to the edge weights is the only way that the network can dispose of charge—the only way that it can “violate” charge conservation. Hence, we refer to this process as the “dissipation” of residual charge to the edge weights, which are adapted by it, as we describe next.

In the implementation described below, the weight updates are sampled independently, such that the time average of the cross terms in Equation (11) vanish. Hence, we rewrite Equation (11) as

$$\begin{aligned}
H_j(\Delta w_j) \approx & \sum_{i^{++}} (e_i^+ \Delta w_{ij}^{++} - \eta_{ij}^{++})^2 + \sum_{i^{+-}} (e_j \Delta w_{ij}^{+-} - \mu_{ij}^{+-})^2 \\
& + \sum_{i^{+-}} (e_i^+ \Delta w_{ij}^{+-} - \eta_{ij}^{+-})^2 + \sum_{i^{++}} (e_j \Delta w_{ij}^{++} - \mu_{ij}^{++})^2 \\
& + \sum_{i^{--}} (e_i^- \Delta w_{ij}^{--} - \eta_{ij}^{--})^2 + \sum_{i^{-+}} (e_j \Delta w_{ij}^{-+} - \mu_{ij}^{-+})^2 \\
& + \sum_{i^{-+}} (e_i^- \Delta w_{ij}^{-+} - \eta_{ij}^{-+})^2 + \sum_{i^{--}} (e_j \Delta w_{ij}^{--} - \mu_{ij}^{--})^2
\end{aligned} \tag{13}$$

Collecting terms, completing squares, dropping terms that are independent of Δw_{ij} , and substituting the definitions from Equation (12) yields

$$\begin{aligned}
 H_j(\Delta w_j) \approx & \sum_{i^{++}} (e_i^2 + e_j^2) \left[\Delta w_{ij}^{++} - \frac{1}{e_i^2 + e_j^2} \left(\frac{e_i}{|e_i|} \frac{\delta_j^{++}}{\Delta_j^{++}} e_i^2 - \frac{e_j}{|e_j|} \frac{\delta_j^{+-}}{\Delta_j^{+-}} e_j^2 \right) \right]^2 \\
 & + \sum_{i^{+-}} (e_i^2 + e_j^2) \left[\Delta w_{ij}^{+-} - \frac{1}{e_i^2 + e_j^2} \left(\frac{e_i}{|e_i|} \frac{\delta_j^{+-}}{\Delta_j^{+-}} e_i^2 - \frac{e_j}{|e_j|} \frac{\delta_j^{++}}{\Delta_j^{++}} e_j^2 \right) \right]^2 \\
 & + \sum_{i^{--}} (e_i^2 + e_j^2) \left[\Delta w_{ij}^{--} - \frac{1}{e_i^2 + e_j^2} \left(\frac{e_i}{|e_i|} \frac{\delta_j^{--}}{\Delta_j^{--}} e_i^2 - \frac{e_j}{|e_j|} \frac{\delta_j^{-+}}{\Delta_j^{-+}} e_j^2 \right) \right]^2 \\
 & + \sum_{i^{-+}} (e_i^2 + e_j^2) \left[\Delta w_{ij}^{-+} - \frac{1}{e_i^2 + e_j^2} \left(\frac{e_i}{|e_i|} \frac{\delta_j^{-+}}{\Delta_j^{-+}} e_i^2 - \frac{e_j}{|e_j|} \frac{\delta_j^{--}}{\Delta_j^{--}} e_j^2 \right) \right]^2
 \end{aligned} \tag{14}$$

The form of Equation (14) is the motivation for the choice of distributing residual charge among the weights, as in Equation (12). The sum of quadratic terms, when exponentiated according to Boltzmann statistics, results in independent Gaussian distributions with simple offsets, which makes it possible to sample weight updates Δw_{ij} independently. As described in Section 2.2.5, weight updates are executed on the edges of the selected compartments whenever the node makes an irreversible update by sampling the distribution and updating the edge weight as $w_{ij} \leftarrow w_{ij} + \Delta w_{ij}$. Importantly, these weight updates include a stochastic component associated with the sampling process, as well as a correction associated with the offset. Because each weight is connected to two nodes, it receives an update as specified in Equation (14) from each node. In our implementation, these two updates are separate sampling events, each providing $\frac{1}{2}$ of the total update (i.e., the Gaussian offset is divided by two) in Equation (14). Also, though not detailed here, to the lowest order these weight updates do not affect the charge transport terms in Equation (4) that were ignored in Equation (8).

The updating of weights according the technique just described is convenient because the updates can be made independently, which greatly simplifies the computational model. It does, however, introduce weight growth as an undesirable artifact. If we consider the collection of weights w_j associated with node j as a vector, then the effect of this independent weight updating is to, on average, increase the magnitude, $|w_j|$, which can be seen from

$$\langle (w_j + \Delta w_j)^2 \rangle = w_j^2 + \langle \Delta w_j^2 \rangle = w_j^2 + \frac{n_j}{2\beta} > w_j^2,$$

where n_j is the number of edges connected to node j that are being updated and β is the inverse temperature. One solution to this problem that still allows the weight updates to be performed independently is to reduce the size of the weight update to account for this weight growth. In our implementation, we reduce the size of each weight on every update by a factor

$$w_j \rightarrow w_j \left(\frac{1}{1 + n_j/(2\beta w_j^2)} \right)^{1/2} \tag{15}$$

Note that this correction becomes small as w_j^2 becomes large, but has a large effect at small weight sizes.

Because the residual charge terms in Equation (9) are proportional to the kinetic terms f_j^\pm , in the most general case only a fraction of the residual charge will be dissipated as weight updates, in which case charge conservation requires that the undissipated fraction be retained as edge charge (which would influence future state decisions and weight updates). Hence, as the edge weight updates are sampled according to Equation (14), in general the edge charges are updated as

$$q_{ij}^{\pm\pm} \rightarrow (1 - f_j^\pm) q_{ij}^{\pm\pm}. \tag{16}$$

In the case of the Heaviside kinetic factors considered here, this simplifies to

$$q_{ij}^{\pm\pm} \rightarrow h(\pm e_j) q_{ij}^{\pm\pm},$$

which dissipates all the residual charge in the selected compartments to update their weights (while retaining all of the residual charge in the unselected compartments).

2.2.5. Network Evolution Model

The task of the network simulation is to evolve the network toward global low-energy states in the presence of time-varying inputs and thermal fluctuations, which we implement as a Markov chain, in which each node state is sampled in a continuous round robin for the duration of the simulation. Each cycle through the nodes is considered to be a “step” in the simulation (displayed as a frame in the videos of Section 2.3), although there is no discontinuity in the round robin update between steps. When the node j is reached in the round robin, its conditional state is sampled according to a Boltzmann distribution as

$$P_j(e_j | \mathbf{e} \neq e_j, \mathbf{w}, \mathbf{q}) = \frac{\exp(-\beta H_j(e_j))}{Z_j} \quad (17)$$

$$Z_j = \sum_{e_j} \exp(-\beta H_j(e_j))$$

After sampling a node state e_j according to Equation (17), the node makes a choice to update its connections in one of two ways:

Reversible update—If the node’s energy is fluctuating between simulation steps by an amount larger than its temperature, then it has not achieved equilibrium with the network. The node temporarily updates its compartment input charges, selects a state, and communicates its state to its connected nodes without updating its edge weights. This update drives fluctuations in the network that feedback to the node as the surrounding network evolves to find a common, low-energy configuration of node states. Different network states are sampled in this process, but their energy distribution is unchanged, which makes this node update process “reversible”. We refer to these updates as “global” or “large-scale”, because their primary purpose is to drive the larger network to find a low energy configuration of node states.

Irreversible update—If the node’s energy is fluctuating between simulation steps by an amount smaller than its temperature, then it has achieved equilibrium with the network. The node permanently updates its input compartment charges, selects a state, communicates its state to its connected nodes, and dissipates residual charge in its selected compartments as updates to its edge weights (Equation (14)). This update permanently modifies the energy distribution of the network states, which makes it “irreversible”. We refer to these updates as “local” or “small-scale”, because their primary purpose is to improve the efficiency of each node by updating its edge weights.

The intuition behind this update strategy is that when fluctuations are high, the node should communicate with its connections reversibly, until the surrounding network can settle into an equilibrium state. After this settling, the node can “commit” to updates that will improve its performance in the future. At the scale of the network, the idea is that a combination of rapid, global, reversible relaxation of the network node states combined with the slower, local, irreversible relaxation of the edge states will connect large and small scale dynamics while using only local interactions to make the computations. A familiar analogy is a meeting of people in which information is communicated until a common (low-fluctuation) understanding is achieved, at which time individual commitments can be made to improve upon the current situation.

Additionally, the Markov chain round robin implementation of these reversible and irreversible updates (Figure 2) ensures that the network dynamics obey causality, by (1) guaranteeing that the entire network is updated prior to any node update, and by (2) learning causal structure (as may be imposed by externally biased nodes) by updating edge states only near equilibrium. Hence, the method addresses the long-standing issue of separating causation and correlation in statistics.

The underlying hypothesis employed here is that thermodynamic evolution always proceeds in the direction of (local) equilibrium and that causal structure in the external potentials becomes embodied within the resulting organization.

We note that network evolution in the TNN is continuous and online without separate passes for “learning” and “inference” found in most artificial neural network models.

2.2.6. External Bias Model

Network models may also include external bias nodes that are sources and sinks of charge for the larger network. In the results of Section 2.3.2, these bias nodes have predetermined node states (e.g., a temporal pattern of $e_j = \pm 1$) and large, fixed output weights. A single bias node is able to polarize proximally connected network nodes (i.e., create some order in the network in its vicinity) and, thereby, diffuse potential into the network. When paired with another bias node of opposite polarity, these polarized regions can evolve to form a conducting bridge that connects them or a domain wall that separates them.

2.2.7. Other Network Effects

For most network topologies, the nodes are “frustrated” because there is no way to achieve perfect antiferromagnetic order. The result of this frustration is that networks segment themselves into “domains” in order to minimize energy and effectively transport charge. Edges within domains adapt weights to remove residual charges and improve transport efficiency (these edges belong to selected compartments), while edges separating domains sustain charges and weights (these edges belong to unselected compartments). Hence, the domain walls maintain both intermediate-term memory (as compartment charges) and long-term memory (as edge weights) of previous network configurations, even as the network adapts to its current inputs. These memories enable the network to rapidly adapt to inputs similar to those it has previously encountered. The creation and destruction of domains in these frustrated networks is perhaps analogous the creation of “virtual networks” among collections of excitatory and inhibitory neurons in biological systems [20]. In bi-partitioned networks, however, the geometric frustration just described can be largely avoided. Nearest neighbor grid networks are inherently bi-partitioned because the nodes fall naturally into two groups on a “checkboard” pattern. These networks are particularly attractive to study because they are easy to visualize. As externally biased nodes are introduced into bi-partitioned networks, however, geometric frustration can once again emerge, as the node biases may conflict depending on their placement in the partitions and their relative polarity. These effects are elaborated in Section 2.3.2 below.

Different node and edge temperatures allow the exploration of temperature-dependent ordering dynamics in unbiased networks. Lower (higher) node temperature relative to edge temperature results in more (less) ordered networks. Ordering in the unbiased networks is also very sensitive to the connectivity of the nodes—more connectivity typically yields greater order. Externally biased nodes introduce other energy scales into the network that compete with thermal fluctuations to determine network dynamics and organization.

2.3. Model Simulation

The following images are snapshots of the evolution of the network captured as images of the node states at the end of one complete cycle of node updates using the techniques described in Section 2.2. These images are frames from videos that can be viewed by following the hyperlinks in the image description. In the images that follow, each node is one square, and its state is indicated on a grayscale with *black* = -1 and *white* = $+1$. The examples below focus on two-dimensional (bi-partitioned) networks with nearest neighbor (NN) connectivity and periodic boundary conditions (the left/right and top/bottom edges are connected) because these networks allow easy visualization of the network organization. The ideas presented below also apply to higher dimensional networks and networks with more complex connectivity, such as randomly connected networks. In every case,

the network is initialized with random node and edge state values and allowed to evolve according to the methodology of the previous section.

2.3.1. Isolated Networks

Figures 4 and 5 are sample results from unbiased networks with with 4 nearest neighbor connectivity interacting with a thermal bath, illustrating the propensity of the network to organize. At the node and edge temperatures $T_{node}/T_{edge} = 1$ selected for these simulations, ordering is local and transient. Figure 5 inverts the display polarity in one of the network partitions in order to show the ordering as “ferromagnetic” instead of “anti-ferromagnetic” (this convention is adopted in all figures except Figure 4 for bi-partitioned networks).

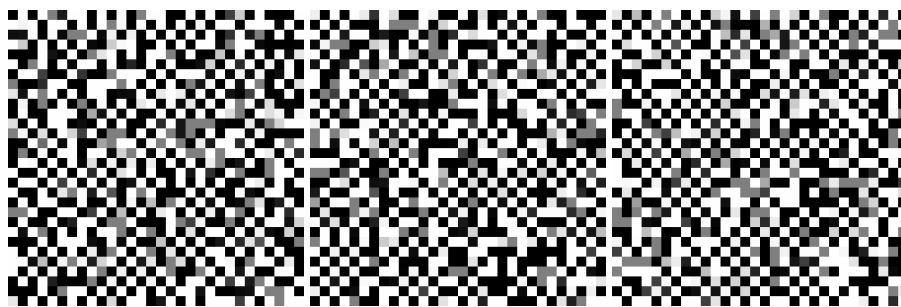


Figure 4. Snapshots from the evolution of a two-dimensional, bi-partitioned network of 900 nodes with each node connected to its 4 nearest neighbors in the opposite partition with $T_{node}/T_{edge} = 1$. Networks are dynamic and noisy owing to contact with the thermal bath. The propensity for the nodes to organize anti-symmetrically is evident in the checkboard appearance of the various regions of the images. Video S1 and at <https://youtu.be/3WFO41aV9lg>.

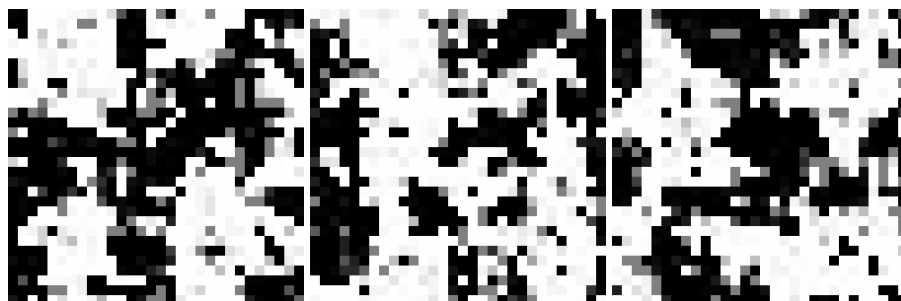


Figure 5. A different simulation of the same network as in Figure 4. Because antisymmetric order is challenging to visualize, in these images the order is displayed by reversing the sign of the node state in one of the partitions (i.e., on every other square on the checkboard). This change is applied only to the image display—the underlying order is still antiferromagnetic. When displaying images this way, domains appear as preferentially “white” or “black”. Video S2 and at https://youtu.be/8_dvWLFr4mA.

Figures 6–8 shows the evolution of larger bi-partitioned networks with 16 nearest neighbor connections per node at 3 different node-to-edge temperatures chosen to illustrate the dynamics of the network with different levels of thermal excitation. All simulations start from a highly disordered state (not shown in figures) and evolve to display complex, multiscale dynamics. In general, as the node-to-edge temperature decreases, ordering extends over larger spatial and temporal scales. Figure 9 shows plots of selected network statistics capturing the ordering of the network of Figure 7.

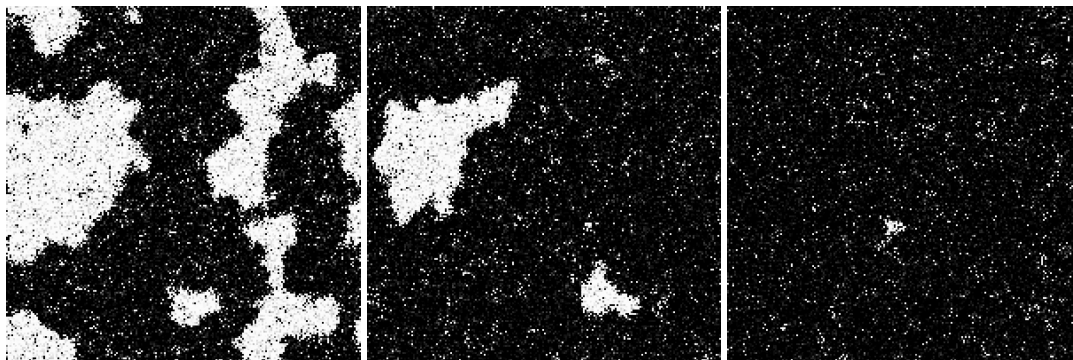


Figure 6. Three frames from the evolution (early to late from left to right) of a two dimensional, bi-partitioned network of 40,000 nodes with each node connected to its 16 nearest neighbors in the opposite partition at $T_{node}/T_{edge} = 80$. The network evolves to a large single domain with local excitations. Video S3 and at <https://youtu.be/1Dj59EI93KE>.

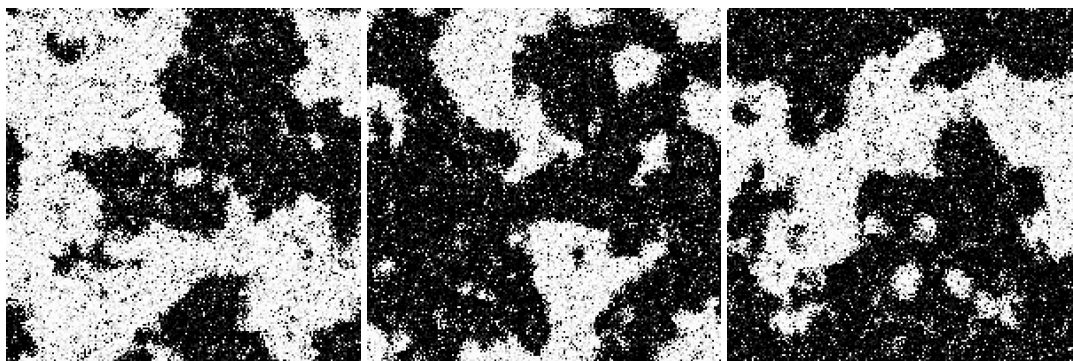


Figure 7. Three frames in the evolution of the same network as in Figure 6 at $T_{node}/T_{edge} = 100$. The network evolves complex multiscale dynamics with domain nucleation and growth. Video S4 and at <https://youtu.be/Ca9XEGyBytg>.

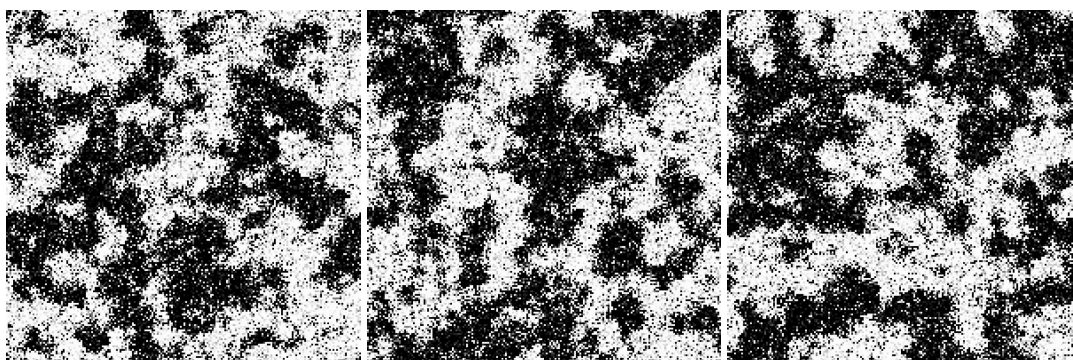


Figure 8. Three frames in the evolution of the same network as in Figure 6 at $T_{node}/T_{edge} = 120$. The network is dominated by faster, smaller scale dynamics as compared to Figure 7. Video S5 and at <https://youtu.be/UUA08xwcMAQ>.

Figure 10 shows the evolution of a bi-partitioned network of 40,000 nodes that are randomly connected to 16 other nodes in the opposite partition. For a narrow band of temperature around $T_{node}/T_{edge} = 120$, the network state rapidly coheres across the entire scale of the network and periodic oscillatory dynamics emerge. The period of oscillation depends on temperature (not shown).

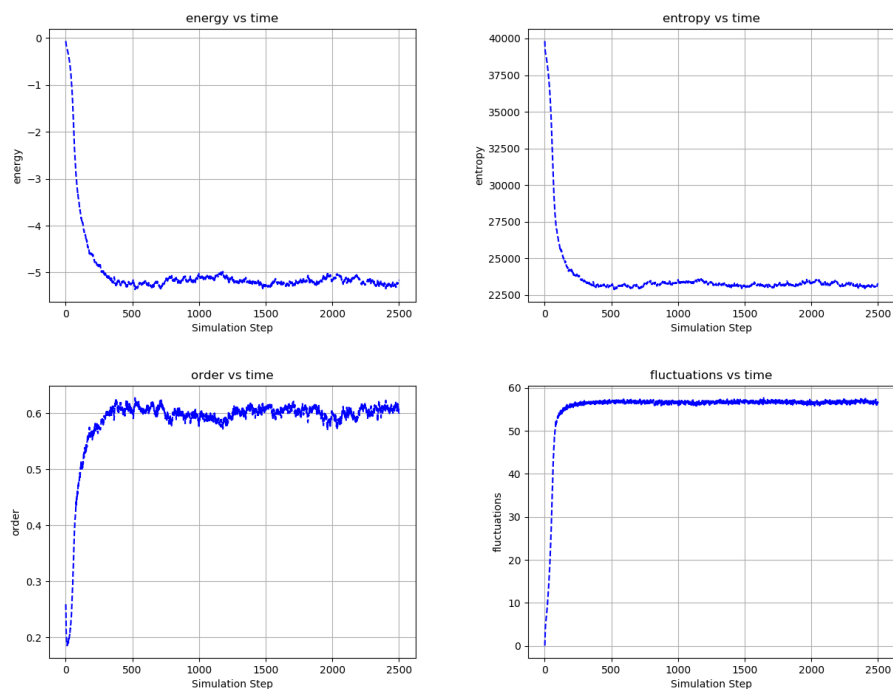


Figure 9. Temporal evolution of selected network averages for the simulation of Figure 7—node energy, node entropy, order, and fluctuations vs simulation time. Energy is the node energy of Equation (5) averaged over all the nodes. Entropy is the sum of the nodes entropies derived from Equation (17) normalized to a maximum value of 40,000. Order is the average over all edges of (the negative of) the product of the edge’s connected node states. Fluctuations are the percentage of time that nodes choose a reversible update averaged over all the nodes. Ordering in the network is consistent with decreased node energy and entropy and increased order and reversible node updates.

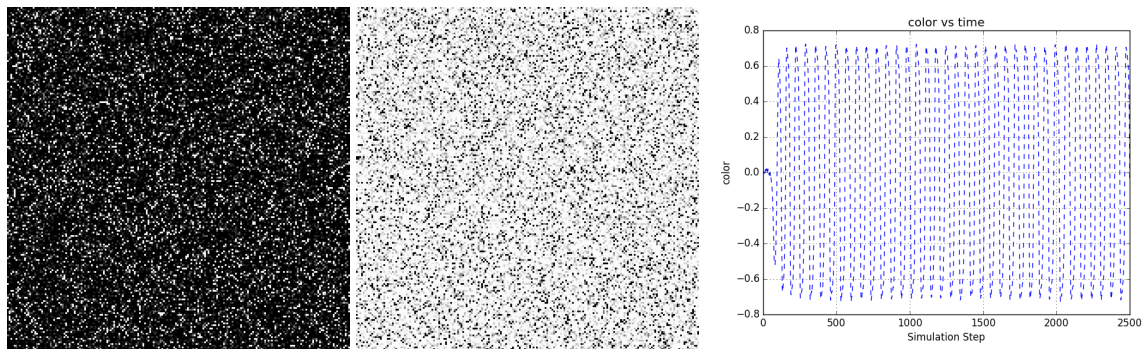


Figure 10. A bi-partitioned network of 40,000 nodes randomly connected to 16 nodes in the opposite partition at $T_{node}/T_{edge} = 120$ uniformly oscillates periodically between mostly dark and light states. (left and center) Two frames from the evolution (<https://youtu.be/1YA5xauI5Y0>) of the network showing dark and light states. (right) Color shows the average state value of the network nodes and illustrates the regularity of the oscillations. Video S6 and at <https://youtu.be/1YA5xauI5Y0>.

2.3.2. Externally Biased Networks

Figures 11–16 are sample results from the simulation of networks biased with external potentials while interacting with a thermal bath. The panels in Figures 11–13 and 15 are frames from videos of the network evolution of various bipartite, nearest neighbor networks with periodic boundary conditions. Once again, these examples were chosen because their evolution is easy to visualize.

Recursive edges were employed in these models to avoid local minima in the network nodes as the bias nodes change polarity.

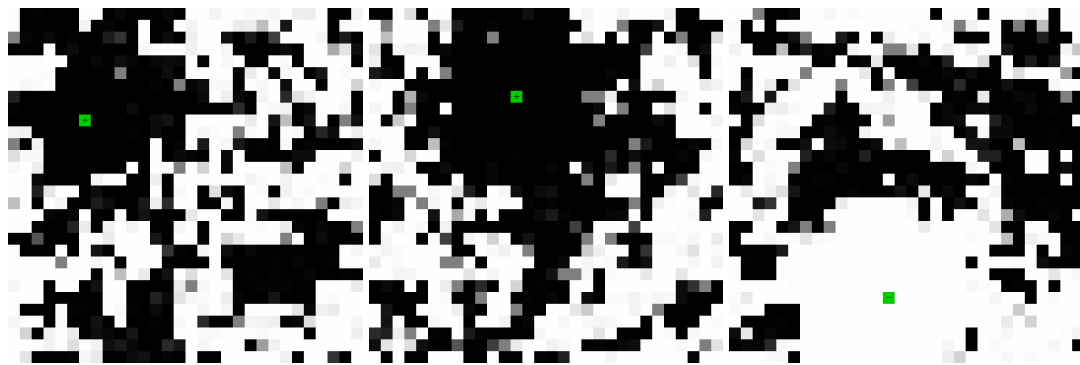


Figure 11. Snapshots from the evolution of a bi-partitioned network of 900 nodes with 4 nearest neighbor connections and a single recurrent connection per node at $T_{node}/T_{edge} = 1$ with a single externally biased node polarizing the region in its vicinity. The sequence of images is from three different simulations with increasing bias strength (increasing size of fixed edge weights connecting the bias node to its neighbors) from left to right in the images. Larger bias creates a larger region of polarization. Domain polarization changes polarity as the biasing node changes sign. Videos S7–S9 and at <https://youtu.be/8kiLYNyOMZ8>, https://youtu.be/HD_kjCEqrYA, <https://youtu.be/pZy6S5Huph4>.

Figures 11 and 12 illustrate the polarization of the network by externally biased nodes and the interaction of externally biased nodes within a network. Figure 11 illustrates the ability of an externally biased node to polarize nodes in its vicinity and communicate with the larger network. Figure 12 illustrates four basic interaction types for two nodes in the network depending on their polarity and their partition placement. In general, the network is able to connect nodes by building strong weights over the long term and to separate nodes by building domain walls in the short term, which is the foundation on which the network can efficiently and rapidly organize itself to transport charge among dynamic external inputs.

Figure 13 illustrates the same concepts as Figures 11 and 12 in a much larger network with 10 pairs of externally biased nodes with complementary partitions and complementary potentials that change polarity with different periods. As it evolves, the network becomes increasingly efficient at segmenting into domains that connect and separate the various bias nodes. In the video, the propagation of potential through the network is easily visualized as the movement of domain walls in response to changes in the external node polarities. Selected statistics of the network in Figure 13 are shown in Figure 14, generally showing improvement in the network's performance with time.

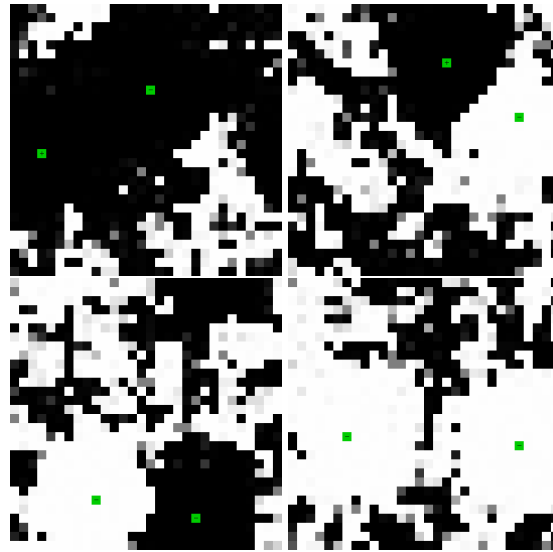


Figure 12. In the same network as in Figure 11, two biased nodes interact through the network. If the nodes are of opposite polarity and opposite partition (**top left**), the network evolves a connection. If the nodes are of opposite polarity and the same partition (**top right**) or of the same polarity and opposite partition (**bottom left**), the network evolves a domain wall. If the nodes are of the same polarity and same partition (**bottom right**), the nodes jointly polarize their surrounding region, but do not grow strong weights between them. Videos S10–S13 and at <https://youtu.be/nj-juPln5b0>, <https://youtu.be/JdBHyPSUFL0>, <https://youtu.be/2N8zqGX0swM>, <https://youtu.be/tVmCSKUA8dQ>.

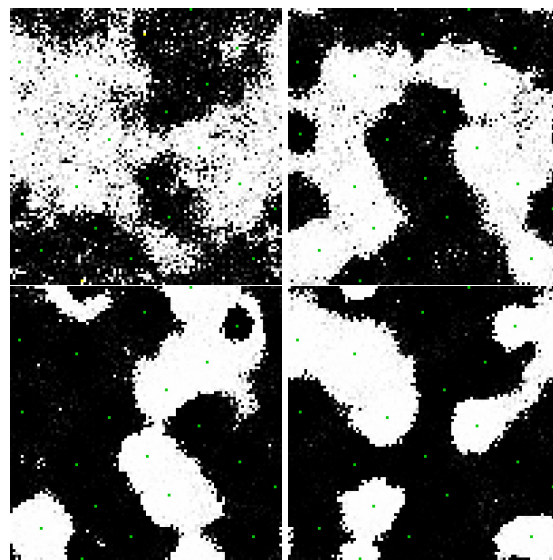


Figure 13. Frames from the evolution of a 10,000-node, bi-partitioned, 16 nearest neighbor network with 2 recurrent connections per node and 10 pairs of bias nodes at $T_{node}/T_{edge} = 100$. Each bias pair is composed of complementary nodes (opposite polarity and opposite partition) that change periodically in time, each of the 10 pairs with different periods. These four images show different configurations of the network as it adapts to changes in its inputs from early to late in the network evolution (left to right and top to bottom). As the edge weights grow, the domains become sharper and better connected. As the input nodes change polarity, the network rapidly adapts by creating, destroying and moving domain walls. In general, the network is challenged to connect and separate nodes into black and white domains according to their polarity and partition (see Figure 12). Video S14 and at <https://youtu.be/xy-eivZ2vJg>.

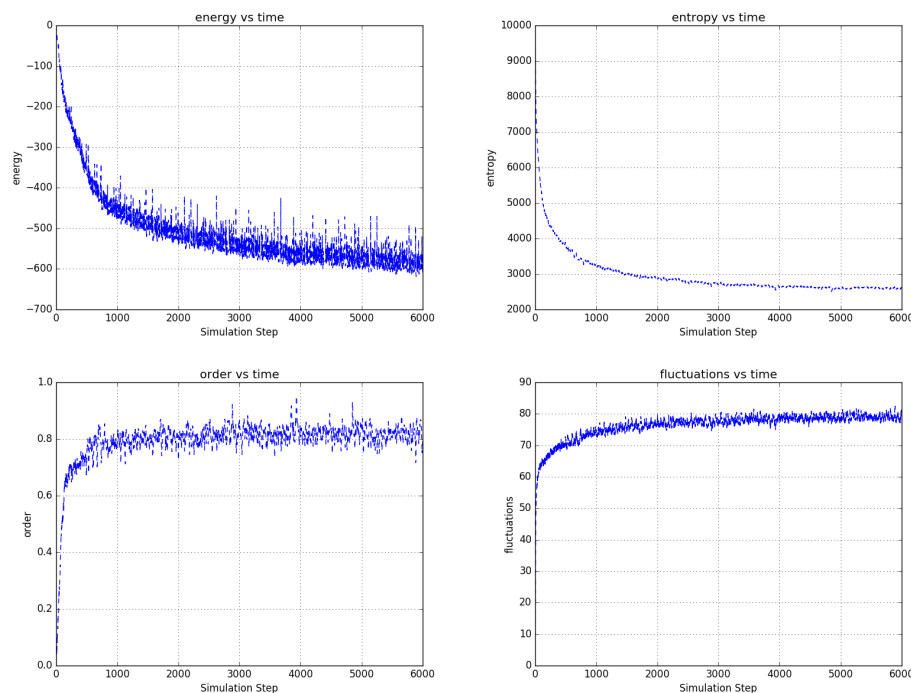


Figure 14. Temporal evolution of selected network averages for the simulation of Figure 13—node energy, node entropy, order, and fluctuations vs simulation time. Energy is the node energy of Equation (6) averaged over all the nodes. Entropy is the sum of the node entropies derived from Equation (17) normalized to a maximum value of 10,000. Order is the average over all edges of (the negative of) the product of the edge’s connected node states. Fluctuations are the percentage of time that nodes choose a reversible update averaged over all the nodes. Even as the bias nodes change polarities, the large-scale behavior of the network is well behaved. As compared to the unbiased network example of Figures 7 and 9, this network shows lower energy and entropy and a higher degree of order and fluctuation, which is consistent with the application of external bias to the network.

Figure 15 replicates the network parameters of Figure 13 to explore the effect of repeatedly adding and removing external bias potential in a series of “waking” and “sleeping” phases. As can be seen in the associated video, the sleeping phase retains some of the modular structure (i.e., the coherent regions surrounding each bias node that are similarly polarized) learned during the waking phase. In this simulation, the external bias weights are small relative to those of the network of Figure 13, as we found this important in creating complex dynamics in the sleeping phase.

Figure 16 shows selected statistics of a single-partition randomly connected network otherwise having many features in common with the bi-partitioned nearest neighbor networks of Figure 13. By its structure, this network is highly frustrated in its ability to connect nodes and create underlying antiferromagnetic order. Nonetheless, the statistics of its evolution are qualitatively similar to those of its less frustrated counterpart of Figure 14. The video of the network evolution shows evidence of ordering but is much less obvious than in the bi-partitioned, nearest neighbor network examples presented above.

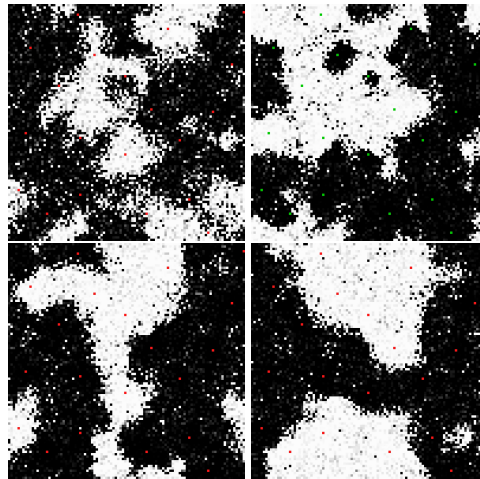


Figure 15. Frames from the evolution of a network like that of Figure 13 at $T_{node}/T_{edge} = 120$ and using smaller fixed weights to bias the network. The simulation periodically applies and removes the bias potentials: bias nodes are green when on and red when off. **(upper left)** The network is not yet exposed to any bias inputs and the dynamics are like those of Figure 8. **(upper right)** Bias has been applied and the network evolves to connect bias nodes as in Figure 13. **(lower left & right)** With the bias removed, the network dynamics continue but are clearly influenced by the modular structure that emerged during the bias phase, seemingly resembling both the unbiased and biased stages of its evolution. Edge weight magnitudes are largely preserved during the unbiased phases of evolution even as they are continuously updated (not shown). Video S15 and at <https://youtu.be/ctmWAu09qTE>.

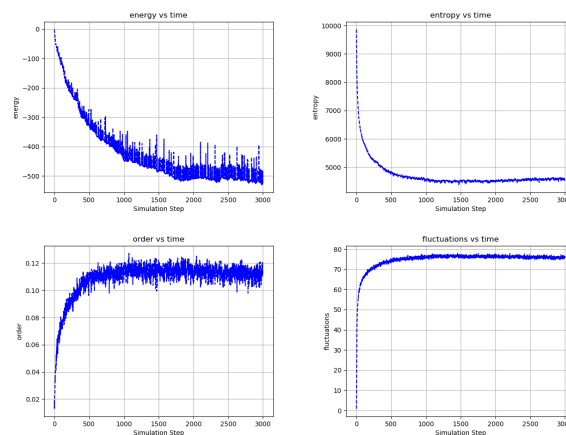


Figure 16. Temporal evolution of selected network averages of a 10,000-node, single partition randomly connected network with 16 connections per node plus 2 recurrent connections per node and 10 pairs of bias nodes at $T_{edge}/T_{node} = 100$. Each pair is composed of opposite polarity nodes that change periodically in time, each of the 10 pairs with different periods. Unlike the networks of Figures 13 and 15, which are bi-partitioned and locally connected, this network is inherently frustrated and must “carve out” connections amid a vast web of competing interactions. **(top left)** Energy is the node energy of Equation (6) averaged over all the nodes. **(top right)** Entropy is the sum of the nodes entropies from Equation (17) normalized to a maximum value of 10,000. **(bottom left)** Order is the average over all edges of (the negative of) the product of the edge’s connected node states. **(bottom right)** Fluctuations are the percentage of time that nodes choose a reversible update averaged over all the nodes. Although inherently frustrated by its connectivity, as evidenced by its relatively low order, the statistics of the network evolution are similar to those of the nearest-neighbor, bi-partitioned network of Figure 14, indicating that the network can effectively evolve connectivity among the bias nodes. Video S16 and at <https://youtu.be/PEFAkAcMdVk>.

3. Discussion

3.1. Model Features

The following paragraphs summarize the main features of the TNN in terms of the physical concepts that motivated its development and the limitations of other artificial neural network paradigms:

Conserved complementary quantities (positive and negative charge) interact but do not cancel. The accumulation of charges within the network represents internal potentials (or energetic costs) that can be reduced by transporting these charges through the network and connecting them to external sources and sinks of charge. These ideas are consistent with relations among forces and fluxes in near-equilibrium thermodynamic systems [21].

Fluctuations are inherent in the model formulation. As compared to most artificial networks, which introduce noise in an ad hoc way, the fluctuations in the TNN are governed by the same relaxation dynamics responsible for charge transport and weight adaptation. In addition, the round robin Markov Chain node updates generate fluctuations with correlations that extend through the network. The fluctuations in the TNN are not “noise”; rather, they are thermodynamically consistent, multiscale variations in the network organization.

Dissipation of residual charge is coupled to fluctuations. Mismatch of input charges on a node creates residual charge that must be dissipated as the network equilibrates. The edge weight updates (14) strive to eliminate this mismatch, but thermal noise in the updates means that the match is always imperfect—even in networks driven by constant external inputs evolved to a low energy steady state. Hence, dissipation in the network is inextricably linked to fluctuations and charge transport “resistance” is emergent and unavoidable. Fluctuation-dissipation effects are well known from equilibrium statistical physics [22] and recently extended to non-equilibrium systems [23,24].

Adaptation is coupled to the dissipation of conserved quantities. While updating weights to eliminate errors in an objective function is the foundation of most neural network models, in the TNN model it also has a physical interpretation. Namely, the dissipation of conserved quantities within an open physical system (e.g., the residual node charges for the TNN), when coupled to the system features responsible for their creation (e.g., the edge weights), can adapt the system to reduce dissipation under similar future conditions. If the environment in which this system is embedded has certain stable features in the potentials presented to the system, then through its interaction with that environment, the system may come to represent and predict those features and to thereby minimize internal dissipation [25]. The intuition here is that the dissipation of a conserved physical quantity requires a physical structure to transport it out of the system. In the TNN model, the supposition is that the edge weight, which creates the charge imbalance, mediates the transport of the residual charge to the reservoir, and in the process is adapted by it. As an example from everyday life, consider a housing construction site in which certain raw materials (the conserved quantities) are cut as the house is built and residual scraps of material are produced that cannot be used. Those scraps, which must be transported away from the construction site (the dissipation), can be used to inform the acquisition of materials in future (the adaptation) and to improve the efficiency of the construction process up to the point that variances in materials and construction permit (the fluctuation). This process can become highly predictable and efficient if the same house is constructed many times, material suppliers are reliable, and labor is consistent (a stable environment).

Edge states adapt with respect to the current state of their connected nodes without destroying state information associated with other node states. The kinetic factors in the model adapt edge states selectively, depending on the node state (Equations (6), (9) and (14)). We presume that this technique allows edge state updates associated with somewhat different configurations of the collective node states to reinforce to the degree that they are similar, but also not cancel to the degree that they are different, thereby enforcing commonalities (generalization) while preserving differences (specialization). Referring again to the example of constructing a house, building a wall and building

a floor may share similar tools and fasteners, but use different types of lumber. In adapting for these tasks, we would like to generalize the adaptation for tools and fasteners while specializing the adaptation for the different types of lumber.

Rapid, global relaxation of the nodes states and slower, local adaptation of the edge states results in the evolution of a multiscale, complex system. Successful adaptation requires that the network achieves equilibrium as quickly as its inputs change and that it refines its connectivity with time to improve this ability to equilibrate in the future, which is implemented through the reversible and irreversible node update decisions, respectively. In the house building analogy, a reversible update might involve the distribution of the materials for the day's work, while an irreversible update might involve the many individual activities using those materials to construct the house. Building the house efficiently requires both that the materials are well distributed and that future distributions are refined according to the waste produced in the construction process. A variety of natural, networked systems and models of such systems involve the idea of adaptation at different scales. The comparison of various systems related to physics, materials, ecology, biology and cognition indicate that "dual phase evolution" may explain common observations of modularity, network statistics and criticality [26] in these diverse domains. We speculate that this idea may also be responsible for the abilities of brains to rapidly orient to new environments and to rapidly learn from unfamiliar experiences and place them in their correct context (so called "one shot" learning).

The TNN employs concepts from equilibrium statistical physics to evolve the dynamic organization of a model system that may be in or out of equilibrium. Both the reversible and irreversible updates to the TNN (Equations (17) and (14)) employ a Boltzmann distribution to sample node and edge states. Furthermore, these distributions are not employed to compute statistics, but to drive the dynamics of network evolution. While the overall validity of such an approach as a model of natural phenomena is not clear, all state decisions in the model are local to the nodes and, as such, local equilibria may be the correct context for updating a distributed, interacting network like the TNN.

Large scale stochastic dynamics can emerge through local interactions. The videos referenced in Figures 7, 13 and 15, for example, clearly show large scale stochastic dynamics. The node-to-edge temperature ratio, bias strength, and network connectivity also play crucial roles in the large scale dynamics of the networks. Although not emphasized in the work presented here, sharp transitions from order to disorder, such as temperature changes and phase transitions, are also possible.

Networks evolve causal dynamics related to the spatial and temporal structure of potentials in their environment. As external potentials change in time, the network responds with corresponding changes in its spatial and temporal structure, which can be characterized as the potentials *causing* change within the network via the transport of charge through it. The round robin Markov chain technique ensures temporal consistency because every node update is preceded by an update of all the other nodes in the network. The irreversible updates, occurring when the nodes are in local equilibrium, ensure that the organization learned by the network is that which is consistent with the second law of thermodynamics and the causative "arrow of time".

The TNN avoids many computational challenges found in other neural network models. Nodes can be connected as networks of any type without creating dynamic instabilities. Node and edge updates are continuous and online without forward and backward passes; there is no separation of "learning" and "inference". There are few ad hoc meta-parameters: for example, there are no learning rates. There are no gradients that need to be computed or communicated across layers.

The TNN unifies concepts of conservation, potentiation, fluctuation, dissipation, adaptation, equilibration and causation under a common physical model to illustrate a multiscale, self-organizing, complex, adaptive system. The model self-organizes with and without external inputs. Externally applied potentials propagate through the network by polarizing connected nodes. Self-organization is strongly modulated by network effects, the relative temperatures of the nodes and edges, and the strength of the external applied potentials.

3.2. Limitations and Speculation on Future Opportunities

The most challenging part of the implementation of the model is the search of the network space to find a representative, low-energy state in the Markov chain round robin. As is typical, the search for a global optimum is frustrated by local minima and there is no all-purpose algorithm to address this problem [27]. In the implementation described here, this is typically recognized as a domain that fails to change state as the external potentials transition (which is partially mitigated by the recursive edges in the examples of Section 2.3.2). In the videos associated with Section 2.3.2, these failures are sometimes visible directly in the networks and are also explicitly illustrated by the yellow color of the bias nodes when any of their connected nodes have the wrong polarity. There is little doubt that the methods used here might be improved to address this challenge, but more comprehensive searches also face a combinatorial explosion of potential evaluations.

It is interesting to consider the source of this challenge in the context of the thermodynamic concepts that motivated the TNN. Every computing model is composed of a sequence of variable assignments. The ability to make these assignments requires that the variables of the model be independent at the time of assignment. For example, if we wish to perform the assignment $a \leftarrow b \cdot c$ then b and c must exist and be independent of a at the time of the assignment. In the model implementation described here, this limitation is recognized in the constant node-at-a-time round-robin search for a low-energy configuration of the node states, as in Equation (17) and in the approximations leading to the edge weight updates Equations (13)–(15). More generally, the challenge of creating the TNN can be seen as taking the relatively simple statement of Equation (2) and the concepts of Section 2.1 and translating them into a sequence of variable assignments that effectively addresses the challenges of capturing the interdependencies of the state variables. The computational techniques described above are both critical to the implementation of the TNN model as well as the illustration of its greatest difficulty.

While we can claim some success in our efforts to address the challenge just described and suppose even that there might be useful implementation of the TNN, there are certain ironies implicit in simulating complex thermodynamic systems on deterministic computing hardware. We must, for example, calculate probability distributions and generate pseudo-random numbers to sample fluctuations using computing hardware that, at great expense, is engineered, manufactured and operated to prevent fluctuations. We must, for example, at great expense, search for a representative sample of an equilibrium distribution, while every natural system does this at essentially zero cost. So, perhaps the most promising future implementations of models such as the one presented here would involve hardware in which the device electronics inherently perform the thermodynamic relaxation that drives the evolution of the network. For example, nodes might be constructed of multistate devices that are marginally stable at their operating temperature and that can be biased to favor transition to a particular state by the charge received from their inputs. Also, edges might be constructed of semi-stable, hysteretic resistive components (“memristors” or “memcapacitors”) that change impedance depending on the history of the current passing through them [28]. Such systems would have orders of magnitude higher energy efficiency, scalability and perhaps offer much more complex functionality than the computational model described here. These future systems, in combination with conventional computing elements, might create the foundations for a “thermodynamic computer” [29] that can both evolve “from below” according to the basic thermodynamics of its components and be constrained “from above” by human-specified code. In such systems, “thermodynamic evolution” might be an omnipresent capacity driving their self-organization toward a high-level, human-specified goal.

4. Materials and Methods

These results are the product of simulation on a laptop computer. The implementation stresses flexibility in the exploration of ideas and has not been refined for efficiency of execution or speed. There is a very high degree of parallelism in the model that very likely could be effectively executed in

modern architecture, such as multi-core CPUs, GPUs, and emerging asynchronous and neuromorphic computing systems [30] (DeBole et al., 2019). The code is available at <https://github.com/toddhylton/Thermodynamic-Neural-Network---Public>.

5. Conclusions

We have described a neural network model comprising a collection of nodes and edges that organize according to basic principles of physics and thermodynamics. Charge conservation laws and the hypothesis that nodes should evolve to transport charge effectively results in networks of nodes that organize to maximize charge transport efficiency. Node and edge state updates derive from the relaxation of the network according to Boltzmann statistics. Node states relax globally and reversibly in concert with edge states that relax locally and irreversibly, resulting in a multiscale self-organizing, complex system with dynamics that are sensitive to network structure and temperature. Externally applied potentials diffuse into the network, establishing strong connections to complementary potentials and creating domain walls to separate competing potentials. The model integrates ideas of conservation, potentiation, fluctuation, dissipation, adaptation, equilibration and causation to illustrate the thermodynamic evolution of organization.

Supplementary Materials: The following are available online at <http://doi.org/10.5281/zenodo.3686045>, Video S1: Figure 4 network evolution, Video S2: Figure 5 network evolution, Video S3: Figure 6 network evolution, Video S4: Figure 7 network evolution, Video S5: Figure 8 network evolution, Video S6: Figure 10 network evolution, Video S7: Figure 11 network evolution w/ small bias weights, Video S8: Figure 11 network evolution w/ medium bias weights, Video S9: Figure 11 network evolution w/ large bias weights, Video S10: Figure 12 network evolution w/ opposite polarity & opposite partition bias nodes, Video S11: Figure 12 network evolution w/ opposite polarity & same partition bias nodes, Video S12: Figure 12 network evolution w/ same polarity & opposite partition bias nodes, Video S13: Figure 12 network evolution w/ same polarity & same partition bias nodes, Video S14: Figure 13 network evolution, Video S15: Figure 15 network evolution.

Funding: This research received no external funding.

Acknowledgments: The inspiration for this work started roughly 12 years ago, shortly before I began working at DARPA. Since that time many people have influenced my thinking in the domains of thermodynamics, learning, neural networks, complex systems, computation, cognitive science, neuroscience and related domains that were key to the development of the TNN. I would like to acknowledge these people in particular for their insights and support—Alex Nugent, Yan Yufik, Yaneer Bar Yam, Robert Fry, Ben Mann, Dharmendra Modha, Narayan Srinivasa, Stan Williams, Jennifer Klamo, Filip Piekiewicz, Patryk Laurent, Csaba Petre, Micah Richert, Dimitri Fisher, Tom Conte, Michael Hazoglou, Natesh Ganesh, and Ken Kreutz-Delgado.

Conflicts of Interest: The author declares no conflict of interest.

References

1. Glauber, R.J. Time-dependent statistics of the Ising model. *J. Math. Phys.* **1963**, *4*, 294–307. [CrossRef]
2. Suzuki, M.; Kubo, R. Dynamics of the Ising model near the critical point. I. *J. Phys. Soc. Jpn.* **1968**, *24*, 51–60. [CrossRef]
3. Kirkpatrick, S.; Sherrington, D. Infinite-ranged models of spin-glasses. *Phys. Rev. B* **1978**, *17*, 4384. [CrossRef]
4. Hopfield, J.J. Neural networks and physical systems with emergent collective computational abilities. *Proc. Natl. Acad. Sci. USA* **1982**, *79*, 2554–2558. [CrossRef]
5. Amit, D.J.; Gutfreund, H.; Sompolinsky, H. Spin-glass models of neural networks. *Phys. Rev. A* **1985**, *32*, 1007. [CrossRef]
6. Bruce, A.; Gardner, E.; Wallace, D. Dynamics and statistical mechanics of the Hopfield model. *J. Phys. Math. Gen.* **1987**, *20*, 2909. [CrossRef]
7. Sompolinsky, H. Statistical mechanics of neural networks. *Phys. Today* **1988**, *41*, 70–80. [CrossRef]
8. Gutfreund, H. From statistical mechanics to neural networks and back. *Phys. A Stat. Mech. Appl.* **1990**, *163*, 373–385. [CrossRef]
9. Ackley, D.H.; Hinton, G.E.; Sejnowski, T.J. A learning algorithm for Boltzmann machines. *Cogn. Sci.* **1985**, *9*, 147–169. [CrossRef]
10. Hinton, G.E.; Sejnowski, T.J. Learning and relearning in Boltzmann machines. In *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*; MIT Press: Cambridge, MA, USA, 1986; pp. 282–317.

11. Levin, E.; Tishby, N.; Solla, S.A. A statistical approach to learning and generalization in layered neural networks. *Proc. IEEE* **1990**, *78*, 1568–1574. [\[CrossRef\]](#)
12. Albert, R.; Barabási, A.L. Statistical mechanics of complex networks. *Rev. Mod. Phys.* **2002**, *74*, 47. [\[CrossRef\]](#)
13. Watkin, T.L.; Rau, A.; Biehl, M. The statistical mechanics of learning a rule. *Rev. Mod. Phys.* **1993**, *65*, 499. [\[CrossRef\]](#)
14. Clark, J.W. Statistical mechanics of neural networks. *Phys. Rep.* **1988**, *158*, 91–157. [\[CrossRef\]](#)
15. Schrödinger, E. *The Physical Aspect of the Living Cell*; Cambridge University press: Cambridge, UK, 1944.
16. Schneider, E.D.; Kay, J.J. Life as a manifestation of the second law of thermodynamics. *Math. Comput. Model.* **1994**, *19*, 25–48. [\[CrossRef\]](#)
17. Glansdorff, P.; Prigogine, I. On a general evolution criterion in macroscopic physics. *Physica* **1964**, *30*, 351–374. [\[CrossRef\]](#)
18. Schlögl, F. On the statistical foundation of the thermodynamic evolution criterion of Glansdorff and Prigogine. *Ann. Phys.* **1967**, *45*, 155–163. [\[CrossRef\]](#)
19. Jun, J.K.; Hübner, A.H. Formation and structure of ramified charge transportation networks in an. *Proc. Natl. Acad. Sci. USA* **2005**, *102*, 536–540. [\[CrossRef\]](#)
20. Yufik, Y.M. How the mind works: An exercise in pragmatism. In Proceedings of the 2002 International Joint Conference on Neural Networks, Honolulu, HI, USA, 12–17 May 2002; Volume 3, pp. 2265–2269.
21. Onsager, L. Reciprocal relations in irreversible processes. I. *Phys. Rev.* **1931**, *37*, 405. [\[CrossRef\]](#)
22. Callen, H.B.; Welton, T.A. Irreversibility and generalized noise. *Phys. Rev.* **1951**, *83*, 34. [\[CrossRef\]](#)
23. Jarzynski, C. Nonequilibrium equality for free energy differences. *Phys. Rev. Lett.* **1997**, *78*, 2690. [\[CrossRef\]](#)
24. Crooks, G.E. Entropy production fluctuation theorem and the nonequilibrium work relation for free energy differences. *Phys. Rev. E* **1999**, *60*, 2721. [\[CrossRef\]](#) [\[PubMed\]](#)
25. Still, S.; Sivak, D.A.; Bell, A.J.; Crooks, G.E. Thermodynamics of prediction. *Phys. Rev. Lett.* **2012**, *109*, 120604. [\[CrossRef\]](#) [\[PubMed\]](#)
26. Paperin, G.; Green, D.G.; Sadedin, S. Dual-phase evolution in complex adaptive systems. *J. R. Soc. Interface* **2010**, *8*, 609–629. [\[CrossRef\]](#) [\[PubMed\]](#)
27. Wolpert, D.H.; Macready, W.G. No free lunch theorems for optimization. *IEEE Trans. Evol. Comput.* **1997**, *1*, 67–82. [\[CrossRef\]](#)
28. Wang, Z.; Joshi, S.; Savel'ev, S.E.; Jiang, H.; Midya, R.; Lin, P.; Hu, M.; Ge, N.; Strachan, J.P.; Li, Z.; et al. Memristors with diffusive dynamics as synaptic emulators for neuromorphic computing. *Nat. Mater.* **2017**, *16*, 101. [\[CrossRef\]](#)
29. Hylton, T.; Conte, T.; DeBenedictis, E.; Ganesh, N.; Still, S.; Strachan, J.P.; Williams, R.S.; Alemi, A.; Altenberg, L.; Crooks, G.; et al. *Thermodynamic Computing: Report Based on a CCC Workshop Held on 3–5 January 2019*; Technical Report; Computing Community Consortium: Washington, DC, USA, 2019.
30. DeBole, M.V.; Taba, B.; Amir, A.; Akopyan, F.; Andreopoulos, A.; Risk, W.P.; Kusnitz, J.; Otero, C.O.; Nayak, T.K.; Appuswamy, R.; et al. TrueNorth: Accelerating From Zero to 64 Million Neurons in 10 Years. *Computer* **2019**, *52*, 20–29. [\[CrossRef\]](#)

