Proc. Natl. Acad. Sci. USA
Vol. 88, pp. 10983–10987, December 1991
Chemistry

Chemical implementation of neural networks and Turing machines

ALLEN HJELMFELT†, EDWARD D. WEINBERGER†, AND JOHN ROSS‡

†Max-Planck-Institut für Biophysikalische Chemie, D-3400 Göttingen, Federal Republic of Germany; and ‡Department of Chemistry, Stanford University, Stanford, CA 94305

Contributed by John Ross, September 6, 1991

ABSTRACT **We propose a reversible reaction mechanism with a single stationary state in which certain concentrations assume either high or low values dependent on the concentration of a catalyst. The properties of this mechanism are those of a McCulloch–Pitts neuron. We suggest a mechanism of interneuronal connections in which the stationary state of a chemical neuron is determined by the state of other neurons in a homogeneous chemical system and is thus a "hardware" chemical implementation of neural networks. Specific connections are determined for the construction of logic gates: AND, NOR, etc. Neural networks may be constructed in which the flow of time is continuous and computations are achieved by the attainment of a stationary state of the entire chemical reaction system, or in which the flow of time is discretized by an oscillatory reaction. In another article, we will give a chemical implementation of finite state machines and stack memories, with which in principle the construction of a universal Turing machine is possible.**

Computations may be supported by many different systems (1, 2), including physical systems like the digital computer, Fredkin logic gates (3), billiard-ball collisions (4), enzymes operating on a polymer chain (1, 5), and more abstract systems like cellular automata (6–8), partial differential equations that simulate cellular automata (9), generalized shifts (4), and neural networks (10–13). Some of these systems can be computationally universal and thus are formally equivalent with a universal Turing machine (10, 14). We may inquire about whether computationally universal devices may be constructed solely from chemical reaction mechanisms in a homogeneous medium. All living entities process information to varying degrees, and this can occur only by chemical means. It is for this reason alone that the subject is of interest. In this article, we discuss the construction of chemical networks where coupled reaction mechanisms implement "programmed" computations as the concentrations evolve in time. It has already been noted that bistable chemical systems are in many ways analogous to a flip-flop circuit, by coupling bistable reactions it is possible to build universal automata (15, 16), and that various chemical mechanisms share a formal relationship with electronic devices (17, 18). We address the construction of computational devices from the viewpoint of neural networks. We propose a chemical reaction network, which is a "hardware" implementation of a neural network, and hence the network can in principle be as powerful as a universal Turing machine (10).

Neural networks are a versatile basis for computation (19). Any finite state machine, and hence the finite state part of a universal Turing machine, can be simulated by a neural network (10, 20). Neural networks also form the basis of many collective computational systems such as feedforward networks or Hopfield's network (11–13). A chemical neural network may serve as the "hardware" for any of the approaches to computation.

The publication costs of this article were defrayed in part by page charge payment. This article must therefore be hereby marked "*advertisement*" in accordance with 18 U.S.C. §1734 solely to indicate this fact.

Neural networks are composed of simple mutually interacting elements. These elements are typically varieties of McCulloch–Pitts neurons (21, 22), which have one of two possible states, either firing or quiescent. The neuron is quiescent for low values of an input parameter and firing for high values of an input parameter. Our chemical neuron consists of coupled steps in a chemical reaction mechanism and two of the chemical species have effectively one of two steady-state concentrations; one of these exists for low values of a catalyst concentration and the other exists for high values of a catalyst concentration. The state of a neuron j is communicated to other neurons i by connections that are either excitatory (increases the input parameter of i if neuron j fires) or inhibitory (decreases the input parameter of i if neuron i is firing). In our chemical network, we treat the species that describes the state of neuron j as an essential activator (excitatory connection) or inhibitor (inhibitory connection) of the catalyst that determines the state of neuron i. One copy of the basic mechanism of the neuron exists for each chemical neuron in the network. Each neuron is chemically distinct, but for convenience we assume that the reactions that constitute each neuron are mechanistically similar. We allow the neurons to communicate with each other only by the activation and inhibition reactions with the catalyst that determines the state of a neuron.

We present hardware for computation. Since the hardware is solely chemical, the evolution of the network is described by mass-action kinetics. We begin with the kinetic equations with properties similar to that of a single McCulloch–Pitts neuron and then proceed to the construction of interneuronal connections. With this in hand, we detail the construction of logic gates: AND, OR, j AND NOT k, and NOR gates, as well as a chemical mechanism for synchronization in temporally discretized networks.

We continue with a chemical implementation of a binary adding machine and stack memories. We discuss compartmentalization of individual neurons either by membranes or as in separated electrochemical cells instead of homogeneous reaction mechanisms for all the neurons, as in this article. The thermodynamics of the computational process is that of chemical thermodynamics, and we connect computational processes like the minimization of Hopfield's energy function to chemical kinetics and thermodynamics.

A Chemical Neural Network

A Single Chemical Neuron. As a basis for a reaction mechanism with neuron-like properties, we use a cyclic enzyme system similar to one studied by Okamoto *et al.* (23, 24, 30) in metabolic regulation and shown by them to have properties of a McCulloch–Pitts neuron. Okamoto *et al.* (23, 24) used irreversible reactions, and we modify the system so all the steps are reversible; this allows us to define affinities for each mechanistic step and thus to examine the thermodynamics of the system. We have also modified one step to be a catalytically mediated reaction rather than a reactant flux. The chemical mechanism that constitutes the ith chemical neuron is

$$I^*_{1i} + C_i \rightleftharpoons X_{1i} + C_i \qquad J_{1i} = k_1 C_i - k_{-1} C_i X_{1i}$$

$$X_{1i} + B_i \rightleftharpoons X^*_{2i} + A_i \qquad J_{2i} = k_2 X_{1i} B_i - k_{-2} A_i$$

$$X_{3i} + A_i \rightleftharpoons X^*_{4i} + B_i \qquad J_{3i} = k_3 X_{3i} A_i - k_{-3} B_i$$

$$X_{3i} \rightleftharpoons I^*_{2i} \qquad J_{4i} = k_4 X_{3i} - k_{-4}, \qquad [1]$$

where the concentrations of the species marked by the * are held at a constant value either by buffering or by flows and have been absorbed into the rate coefficients. The system is open with respect to I_{1i}, X_{2i}, X_{4i}, and I_{2i} such that their concentrations are held constant; however, the remaining species are not permitted to cross the boundary of the system. C_i is the input parameter for the chemical neuron i. The species X_{1i}, X_{3i}, A_i, and B_i evolve in time to a unique stationary state dictated by C_i according to the differential equations

$$\frac{dX_{1i}}{dt} = J_{1i} - J_{2i} \qquad [2]$$

$$\frac{dX_{3i}}{dt} = -J_{3i} - J_{4i} \qquad [3]$$

$$\frac{dA_i}{dt} = -\frac{dB_i}{dt} = J_{2i} - J_{3i}. \qquad [4]$$

The concentration of the species A_i determines the state of the neuron. The species A_i and B_i are linked by a conservation constraint ($A_i + B_i = A_o$); the chemical neuron is said to be firing when the concentration of A_i is large (B_i is small) and quiescent when A_i is small (B_i is large). The concentrations of the species A_i and B_i in the steady state are plotted in Fig. 1 for the rate constants given in Table 1. For $C_i < 0.90$ mmol/liter, the concentration of $A_i < 2 \times 10^{-4}$ mmol/liter, and for $C_i > 1.10$ mmol/liter, the concentration of $A_i > 0.999$ mmol/liter. The rate coefficients k_2 and k_3 determine the steepness of the jump near $C_i = 1$ mmol/liter and therefore act as the gain (12) of this chemical neuron. If $k_2 \neq k_3$ then the gain curve as a function of C_i is not symmetric around $C_i = 1$ mmol/liter. The values of k_2 and k_3 used by Okamoto and used here correspond to a symmetric high gain case; however, a few trial calculations were performed where this is not the case and the results do not differ significantly.

Okamoto *et al.* (23–25) use the reaction mechanism on which our model of a chemical neuron is based for a simplified cyclic enzyme system, where A_i and B_i are cofactors of

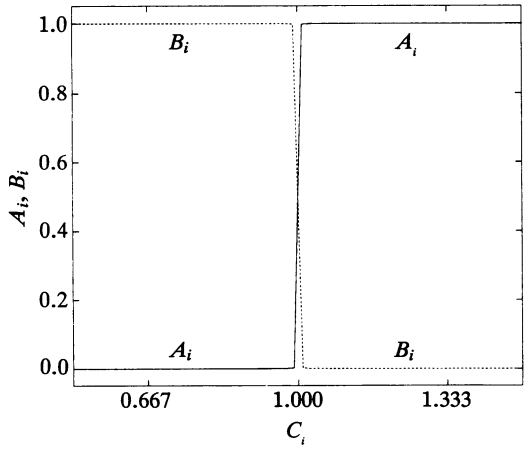

FIG. 1. Plot of the stationary state concentration of A_i and B_i as a function of C_i given by Eqs. 1–4 and the constants in Table 1.

Table 1. Constants used in the calculations

Parameter	Value
k_1	100 s^{-1}
k_{-1}	1 s^{-1} (mmol/liter)$^{-1}$
k_2	5×10^4 s^{-1}·(mmol/liter)$^{-1}$
k_{-2}	1 s^{-1}
k_3	5×10^4 s^{-1}·(mmol/liter)$^{-1}$
k_{-3}	1 s^{-1}
k_4	1 s^{-1}·(mmol/liter)$^{-1}$
k_{-4}	100 s^{-1}·(mmol/liter)
A_o	1 (mmol/liter)

Rate constants chosen for the calculations correspond to those used by Okamoto *et al.* (23), with additional constants chosen arbitrarily to make the reactions reversible.

enzymes that mediate reactions J_2 and J_3. For example, A_i may be NAD^+ and B_i may be NADH. Although this particular model (Eqs. 1–4) is a simplification, mechanisms that show this Heaviside response are not uncommon in metabolic regulation. One example is a mechanism proposed for bacterial sugar transport (26), and another is a mechanism proposed for a pyruvate dehydrogenase reaction (27). See Okamoto *et al.* (23–26) and references therein for further examples and discussion. We also note that steps J_2 and J_3 need not be enzymatic; they may be redox reactions in which A_i and B_i are oxidized and reduced forms of a species, perhaps a metal ion.

Construction of Interneuronal Connections. For each neuron in the network, there is one copy of mechanism 1; the neurons are mechanistically similar but chemically distinct. The effect of the state of the other neurons $j, k, \ldots$ on neuron i is contained in C_i. The species A_i and B_i may affect the concentration of the catalyst C_j of other neurons by activation and inhibition reactions. As one of many possible couplings, we choose an enzyme mechanism in which the concentration of one of the two species (A_j or B_j) acts as an essential activator of an enzyme E_{ij} to form C_{ij}. We stipulate one such enzyme–effector pair corresponding to each connection from neurons j to neuron i, and the sum of the active forms of each enzyme–effector pair gives C_i in Eq. 1,

$$C_i = \sum_j C_{ij}. \qquad [5]$$

We may also construct a system in which the state species of more than one neuron act as activators of the same enzyme species E_i to form C_i. We simplify the description of the connections by assuming that the binding of the activators with the enzyme is a fast process compared with the relaxation time of a single neuron to its stationary state, and hence the enzyme–activator reaction is assumed to be at equilibrium. We also make the typical assumption that the concentration of the enzyme is very small compared to the maximum concentration of the effector (A_i or $B_i = 1$). Suppose that the state of neuron j is to excite neuron i; then we require that if A_j is low, C_{ij} is also low, and if A_j is high, C_{ij} is also high. Therefore, if the species A_j is an essential activator of E_{ij} to form C_{ij}, then we have the desired properties.

$$E_{ij} + A_j \rightleftharpoons C_{ij} \qquad C_{ij} = \frac{E^o_{ij}}{1 + \dfrac{1}{KA_j}}. \qquad [6]$$

$E^o_{ij} = C_{ij} + E_{ij}$ is the total concentration of the enzyme in all its forms, and K is the equilibrium constant for reaction 6. C_{ij} contributes to C_i through Eq. 5. By adjusting the constants E^o_{ij} and K, we can give this mechanism the desired properties for a specific excitatory connection.

Suppose that the state of neuron j is to inhibit neuron i; then we require that if A_j is low, C_{ij} is high, and if A_j is high, C_{ij} is low. Therefore, if the species B_j is an essential activator of E_{ij} to make C_{ij}, we have the desired properties

$$E_{ij} + B_j \rightleftharpoons C_{ij} \qquad C_{ij} = \frac{E_{ij}^o}{1 + \dfrac{1}{K(A_o - A_j)}}. \qquad [7]$$

One such reaction of type **6** or **7** occurs for each connection between neurons.

Construction of Logic Gates. We can make many different types of combinations of connections between neurons. In the following, we describe various possibilities for a neuron i whose state is affected by neurons j and k. Neuron i can be viewed as performing logical operations on the state of neurons j and k, and we describe various types of logic gates.

In Fig. 2, we show schematically two reaction mechanisms constituting neurons i and j and the influence of neurons j, k, and l on neuron i. The state of neuron j determines the concentration of C_{ij}, and we suppose that the firing of neuron i is inhibited by the firing of neuron j (Eq. **7**). The states of neurons k and l (data not shown) likewise determine the concentrations C_{ik} and C_{il}, and the sum of C_{ij}, C_{ik}, and C_{il} is C_i, the parameter that determines the state of neuron i. The state of neuron i in turn determines the concentration of C_{ki}, and the firing of neuron k is inhibited by the firing of neuron i.

AND Gate. For neuron i to perform an AND operation on the output of neurons j and k we allow one generic activation reaction (Eq. **6**) to occur with each of the species A_j and A_k. The total concentration of the catalyst ($C_i = C_{ij} + C_{ik}$) is

$$C_i = \frac{1}{1 + \dfrac{1}{2A_j}} + \frac{1}{1 + \dfrac{1}{2A_k}} = \begin{cases} 4/3 & A_j = A_k = 1 \\ 2/3 & A_j = 1 \quad A_k = 0 \\ 2/3 & A_j = 0 \quad A_k = 1 \\ 0 & A_j = A_k = 0 \end{cases}, \qquad [8]$$

where $K = 2$ and $E_{ij}^o = 1$ in Eq. **6**. The two terms in this equation represent the independent effects of A_j and A_k. Only when both A_j and A_k are large (those neurons are firing) is $C_i > 1$, which causes neuron i to fire (see Fig. 1), and neuron i thus acts as an AND gate pertaining to the inputs from neurons j and k.

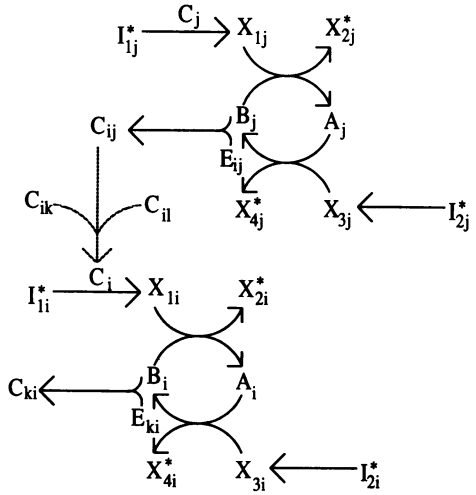

FIG. 2. Schematic of two reaction mechanisms constituting neurons i and j, and the influence of neurons j, k, and l on i. All reactions are reversible, and the concentration of the species marked by * is held constant. The firing of neuron j inhibits the firing of neuron i, and neurons k and l (data not shown) also influence the state of neuron i. The firing of neuron i inhibits the firing of neuron k.

OR Gate. An OR gate is the same as an AND gate, except we raise the value of each numerator, so if only one of the two neurons j or k is firing then neuron i will also fire. We increase the value of E_{ij}^o in Eq. **6** to $E_{ij}^o = 2$ and find

$$C_i = \frac{2}{1 + \dfrac{1}{2A_j}} + \frac{2}{1 + \dfrac{1}{2A_k}} = \begin{cases} 8/3 & A_j = A_k = 1 \\ 4/3 & A_j = 1 \quad A_k = 0 \\ 4/3 & A_j = 0 \quad A_k = 1 \\ 0 & A_j = A_k = 0 \end{cases}. \qquad [9]$$

Thus, neuron i fires if either neuron j or k fires, and therefore acts as an OR gate.

A_j AND NOT A_k Gate. The construction of neurons whose state may become quiescent in response to firing of other neurons requires inhibitory connections. One excitatory connection and one inhibitory connection can make an A_j AND NOT A_k response in neuron i. Thus, the total concentration of the catalyst C_i is

$$C_i = \frac{1}{1 + \dfrac{1}{2A_j}} + \frac{1}{1 + \dfrac{1}{2(1 - A_k)}} = \begin{cases} 4/3 & A_j = 1 \quad A_k = 0 \\ 2/3 & A_j = A_k = 1 \\ 2/3 & A_j = A_k = 0 \\ 0 & A_j = 0 \quad A_k = 1 \end{cases}. \qquad [10]$$

A_j is the excitatory input and A_k is the inhibitory input. As long as $A_k = 1$ the neuron will not fire regardless of the concentration of A_j. However, if $A_k = 0$ the concentration of A_j determines whether the neuron will fire.

NOR Gate. With a NOR gate the presence of any firing input from neurons j and k prevents neuron i from firing. Thus, we need two inhibitory connections of the type given by Eq. **7**. The concentration of the catalyst is

$$C_i = \frac{1}{1 + \dfrac{1}{2(1 - A_j)}} + \frac{1}{1 + \dfrac{1}{2(1 - A_k)}} = \begin{cases} 4/3 & A_j = A_k = 0 \\ 2/3 & A_j = 1 \quad A_k = 0 \\ 2/3 & A_j = 0 \quad A_k = 1 \\ 0 & A_j = A_k = 1 \end{cases}. \qquad [11]$$

When any input, j or k, is firing the firing of neuron i is suppressed.

Specific Inhibition. It is possible to construct connections where the connection enzyme (C_{ij}) in Eq. **5** is inhibited or activated by more than one species. That is, A_j and A_k interact with the same enzyme E_i to form the catalyst C_i. This, for example, allows specific inhibition of one connection instead of the nonspecific inhibition given by Eq. **7**. For a two-input neuron, this is another way of constructing an A_j AND NOT A_k gate. We write

$$E_i + A_j \rightleftharpoons C_i \qquad [12]$$

$$E_i + A_k \rightleftharpoons (E_i A_k), \qquad [13]$$

where A_j is the excitatory input, A_k is the inhibitory input, C_i is the active form of the enzyme, and $(E_i A_k)$ is inactive. The concentration of the catalyst is given by

$$C_i = \frac{E_i^o}{1 + \dfrac{1}{K_A A_j} + \dfrac{K_I A_k}{K_A A_j}}, \qquad [14]$$

where $E_i^o = E_i + C_i + (E_iA_k)$, K_A is the equilibrium constant of the activation reaction (Eq. **12**), and K_I is the equilibrium constant of the inhibition reaction **13**. With the proper choice of K_A and K_I, A_k can inhibit the activation of neuron i by neuron j. Specific inhibition can effectively be produced by using connections of the types given by Eqs. **6** and **7**; however, with a complex set of connections, the mechanism of inhibition and activation (Eqs. **12** and **13**) of the same enzyme may be more transparent.

Synchronization. In many types of neural networks the flow of time is continuous, and issues regarding timing of updating of neurons are unimportant. Such networks include some Hopfield networks and many feedforward networks. However, in other networks the flow of time is discretized, such as in the neural nets described by Minsky (10) for the basis of a Turing machine. The state of all of the neurons and thus their outputs are updated synchronously, depending on the inputs to the neurons during the prior time step. In this section, we describe a clock mechanism that is only used in clocked networks. This discreteness of time and synchronization of state changes can be implemented chemically by the use of an autonomously oscillating catalyst, ε. We assume that ε oscillates in a nonsinusoidal manner as is common in many chemical oscillators (28). The concentration of ε is assumed to be very small except during an interval that is short compared to the oscillator period and to the relaxation time of a chemical neuron (Eqs. **1–4**). The catalyst ε interacts with the species A_j (or B_j) of each neuron j:

$$A_j \overset{\varepsilon}{\rightleftharpoons} A_j' \qquad B_j \overset{\varepsilon}{\rightleftharpoons} B_j'. \qquad [15]$$

The equilibrium point concentration of A_j' is taken to be much smaller than that of A_j, so these reactions do not significantly alter the concentration of A_j or B_j. The catalyst ε is present at a sufficiently large concentration only during a short interval, and only then are reactions **15** allowed to proceed. We now take A_j' or B_j' to be the effector of the enzyme E_{ij} in Eq. **6** or **7**: $E_{ij} + A_j' \rightleftharpoons C_{ij}$, etc. The pulse of ε equilibrates reactions **15** and **6** or **7** based on the concentration of A_j at that instant; however, the time interval of the pulse is too short for the states of the neurons to change. Following the pulse, the neurons relax to their new stationary states, and the values of A_j' and C_{ij} do not change until the next pulse of catalyst. The short pulses of ε may also be imposed on the system externally such as by a pulse of light or gaseous material.

Input–Output. In our chemical neural networks, data are encoded as a set of concentrations of A_is. In some neural networks, initial conditions are specified and the network evolves in time to the solution. In our chemical neural network, the concentrations of the A_is are set at the initial time and then allowed to decay in time to the solution, which is represented as a set of stationary A_i concentrations of the network.

In other neural networks, data are encoded as states of neurons that are specified in time. In these chemical neural networks, various concentrations must be modulated in time. This may be effected, for instance, by allowing the flow of C_i into and out of the neuron and by controlling the concentration of C_i in the feed stream or by generating and removing C_i in an electrochemical or photochemical reaction whose rate is controlled by the external world. The output is represented by the concentration of A_i of particular neurons that can be monitored.

Discussion

In our implementation of neural networks, each neuron consists of eight species. The species I_1, X_2, X_4, and I_2 may be common to all neurons, but the species X_1, A, B, and X_3 must be chemically distinct from those in other neurons. Thus, a minimum of $4 + 4N$ distinct species are required for N neurons. Furthermore, each connection (Eq. **6** or **7**) requires one additional distinct species. Thus, the chemical network as a whole requires $4 + 4N + M$ distinct species where M is the number of connections. The need for large numbers of species presents a practical problem; however, if we assume that the species are polymer chains like proteins, then this large variety of specificity and activity can be imagined. Also, if we were to consider a spatially distributed system (compartmentalized as in biological neural networks), then the variety of species required can easily be reduced.

We have presented a versatile basis for computational hardware based on relating chemical reaction dynamics to neural networks. In particular, it is the stationary state properties of reactions **1–4** that form the basis of the model. The reaction kinetics cause each neuron i to relax toward the stationary state dictated by C_i given in Fig. 1. However, the species C_i may vary with time due to changes of the state of other neurons (the concentration of A_j) or due to modulation in time by the experimenter.

We have constructed computational devices from our chemical neurons: logic gates in this article, and binary adding machines, stack memories, and Hopfield networks in articles to follow. A universal Turing machine can be constructed from a clocked neural network of finite size augmented by two stacks of infinite size (29). In contrast to so-called "computationally reducible" systems, whose behavior can be predicted by a more computationally efficient means than by direct simulation (2), the ability of a chemical network to simulate a Turing machine guarantees in general no such short-cut exists. The existence of a general short-cut would represent the solution to Turing's insoluble Halting problem (10). The lack of predictability of computationally universal systems is stronger than deterministic chaos (4). The motion of chaotic systems is at least confined to an attractor that may be identified beforehand, but the arbitrarily and unpredictably long transients of computationally irreducible systems preclude the a priori identification of attractors, if any. Consideration of computational irreducibility may apply to biological networks.

A.H. and E.D.W. acknowledge support from the Max Planck Society and thank Dr. Manfred Eigen for providing a stimulating research environment, and we thank him for his interest in this work. This work was supported in part by the National Science Foundation.

1. Bennett, C. (1982) *Int. J. Theor. Phys.* **21,** 905–940.
2. Wolfram, S. (1985) *Phys. Rev. Lett.* **54,** 735–738.
3. Fredkin, E. & Toffoli, T. (1982) *Int. J. Theor. Phys.* **21,** 219–253.
4. Moore, C. (1990) *Phys. Rev. Lett.* **20,** 2354–2357.
5. Bennett, C. & Landauer, R. (1985) *Sci. Am.* **253** (7), 38–46.
6. Wolfram, S. (1983) *Rev. Mod. Phys.* **55,** 601–644.
7. Margolus, N. (1984) *Physica* **10D,** 81–94.
8. Wolfram, S. (1984) *Physica* **10D,** 1–35.
9. Omohundro, S. (1984) *Physica* **10D,** 128–134.
10. Minsky, M. (1967) *Computation: Finite and Infinite Machines* (Prentice-Hall, Englewood Cliffs, NJ).
11. Hopfield, J. (1982) *Proc. Natl. Acad. Sci. USA* **79,** 2554–2558.
12. Hopfield, J. (1984) *Proc. Natl. Acad. Sci. USA* **81,** 3088–3092.
13. Hopfield, J. & Tank, D. (1986) *Science* **233,** 625–633.
14. Turing, A. (1936) *Proc. Lond. Math. Soc. Second Ser.* **42,** 230–265.
15. Rössler, O. (1974) in *Lecture Notes in Biomathematics 4*, eds. Conrad, M., Güttinger, W. & Dal Cin, M. (Springer, Berlin), pp. 399–418.
16. Rössler, O. (1974) in *Lecture Notes in Biomathematics 4*, eds. Conrad, M., Güttinger, W. & Dal Cin, M. (Springer, Berlin), pp. 546–582.

17. Seelig, F. & Rössler, O. (1972) *Z. Naturforsch. B* **27**, 1441–1444.
18. Rössler, P. (1972) *J. Theor. Biol.* **36**, 413–417.
19. Rumelhart, D. & McClelland, J. (1986) *Parallel Distributed Processing* (Massachusetts Institute of Technology Press, Cambridge, MA), Vols. 1 and 2.
20. Siegelman, H. & Sontag, E. D. (1991) *Report SYNCON-91-08* (Dept. of Mathematics, Rutgers Univ., New Brunswick, NJ).
21. McCulloch, W. S. & Pitts, W. (1943) *Bull. Math. Biophys.* **5**, 115–133.
22. von Neumann, J. (1956) in *Automata Studies*, eds. Shannon, C. & McCarthy, J. (Princeton Univ. Press, Princeton), pp. 43–98.
23. Okamoto, M., Sakai, T. & Hayashi, K. (1987) *BioSystems* **21**, 1–11.
24. Okamoto, M. & Hayashi, K. (1983) *J. Theor. Biol.* **104**, 591–598.
25. Okamoto, M., Katsurayama, A., Tsukiji, M., Aso, Y. & Hayashi, K. (1980) *J. Theor. Biol.* **83**, 1–16.
26. Okamoto, M. & Hayashi, K. (1985) *J. Theor. Biol.* **113**, 785–790.
27. Stadtman, E. & Chock, P. (1977) *Proc. Natl. Acad. Sci. USA* **74**, 2761–2765.
28. Field, R. & Burger, M. (1985) *Oscillations and Traveling Waves in Chemical Systems* (Wiley, New York).
29. Hopcroft, J. & Ullman, J. (1969) *Formal Languages and Their Relation to Automata* (Addison-Wesley, Reading, MA).
30. Okamoto, M., Sakai, T. & Hayashi, K. (1988) *Biol. Cybern.* **58**, 295.