

The Ghost in the Machine
Basins of Attraction
of Random Boolean Networks

Andrew Wuensche

CSRP 281

1993

UNIVERSITY OF



SUSSEX
AT BRIGHTON

Cognitive Science
Research Papers

ISSN 1350-3162

To be published in Artificial Life III Proceedings, Santa Fe Institute Studies in the
Sciences of Complexity, Proceedings vol.XVI.

The Ghost in the Machine

Basins of Attraction of Random Boolean Networks

Andrew Wuensche

Santa Fe Institute
and The University of Sussex
School of Cognitive and Computing Sciences
wuensch@santafe.edu
andywu@cogs.susx.ac.uk

contact address:
48 Esmond Road, London W4 1JQ
tel 081 995 8893 fax 081 742 2178
100020.2727@compuserve.com

Abstract

This paper examines the basins of attraction of random Boolean networks, a very general class of discrete dynamical systems, in which cellular automata (CA) form a special subclass. A reverse algorithm is presented which directly computes the set of pre-images (if any) of a network's state. Computation is many orders of magnitude faster than exhaustive testing, making the detailed structure of random network basins of attraction readily accessible for the first time. They are portrayed as diagrams that connect up the network's global states according to their transitions. Typically, the topology is branching trees rooted on attractor cycles.

The homogeneous connectivity and rules of CA are necessary for the emergence of coherent space-time structures such as gliders, the basis of CA models of artificial life. On the other hand random Boolean networks have a vastly greater parameter/basin field configuration space capable of emergent categorisation.

I argue that the basin of attraction field constitutes the network's memory; but not simply because separate attractors categorise state space - in addition, within each basin, sub-categories of state space are categorised along transient trees far from equilibrium, creating a complex hierarchy of content addressable memory. This may answer a basic difficulty in explaining memory by attractors in biological networks where transient lengths are probably astronomical.

I describe a single step learning algorithm for re-assigning pre-images in random Boolean networks. This allows the sculpting of their basin of attraction fields to approach any desired configuration. The process of learning and its side effects are made visible. In the context of many semi-autonomous weakly coupled networks, the basin field/network relationship may provide a fruitful metaphor for the mind/brain.

Introduction

Recent work in unravelling the global dynamics of discrete dynamical systems such as cellular automata³¹ and, more generally, of random Boolean networks^{32,33,34}, allow their basins of attraction to be explicitly portrayed. These are diagrams that connect up the network's global states according to their transitions - typically, the topology is branching trees rooted on attractor cycles. The diagrams are efficiently constructed with a reverse algorithm that directly computes a state's set of pre-images (if any).

Following Hopfield⁸, I argue that *attractors* constitute the networks "content addressable" memory; but not simply because separate regions of state space flow to energy minima - in addition, states space is categorised hierarchically along transient trees far from equilibrium.

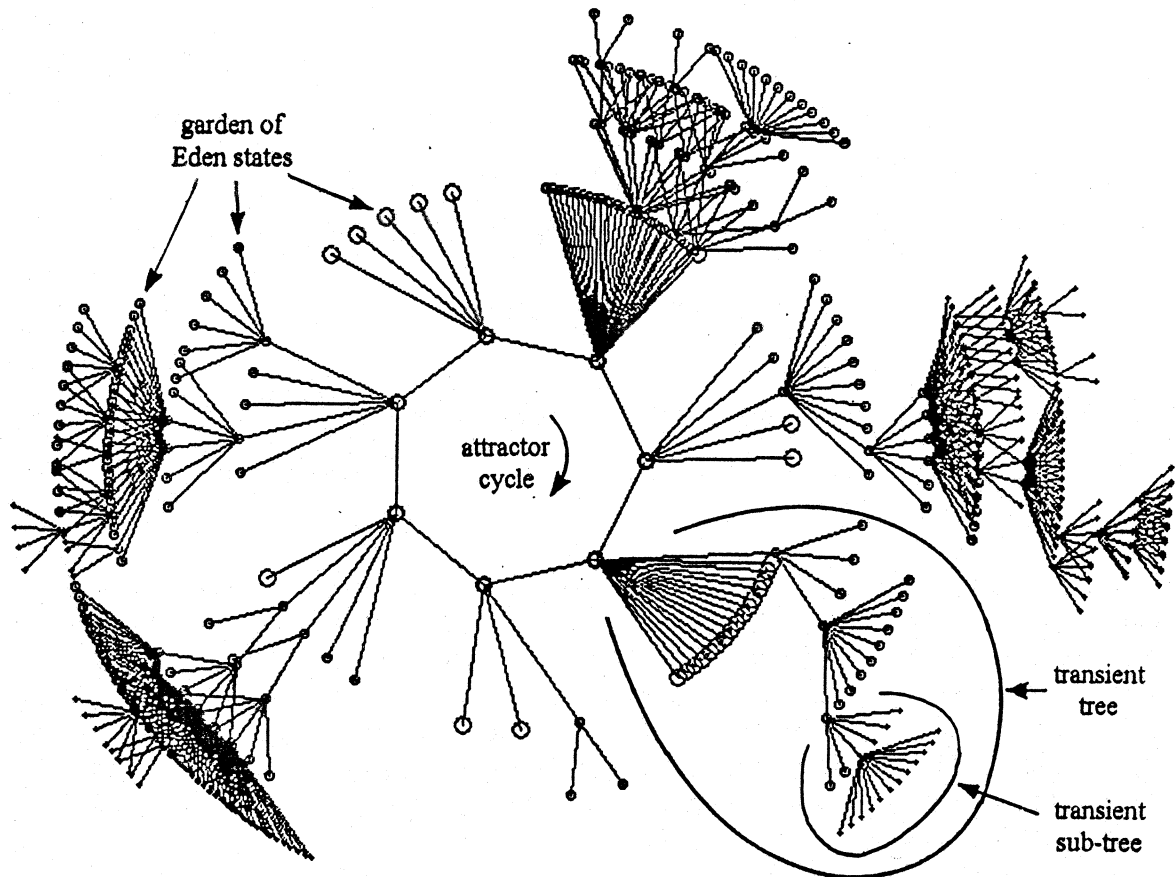


Figure 1

Above. A basin of attraction of a random Boolean network ($N=13$, $K=3$). The basin links 604 states, of which 523 are garden of Eden states. The attractor has period 7. The direction of time is inwards from garden of Eden states to the attractor, then clock-wise. The basin is one of 15, and is indicated in the basin of attraction field in figure 2.

Right. The random Boolean network wiring/rule parameters. Wiring and rules were assigned at random, except that the neighbourhood $000 \rightarrow 0$.

cell	wiring	rule,	table
1	3,12,6	86,	01010110
2	7,11,4	4,	00000100
3	3,3,1	196,	11000100
4	11,3,9	52,	00110100
5	8,7,5	234,	11101010
6	1,8,1	100,	01100100
7	12,4,13	6,	00000110
8	8,6,8	100	01100100
9	9,2,6	6,	00000110
10	5,1,1	94,	01011110
11	2,7,1	74,	01001010
12	7,8,4	214,	11010110
13	1,4,7	188,	10111100

In continuous deterministic dynamical systems, all possible time series make up the vector field which is represented by the system's phase portrait, an idea introduced by Poincaré. This is the field of flow imposed on phase space by the system's dynamical rule. A set of attractors, be they fixed point, limit cycles or chaotic, attract various regions of phase space in the *basin of attraction field*. Analogous concepts apply to discrete deterministic dynamical systems, such as cellular automata (CA) and the more general case, random Boolean networks, which are noise free and update synchronously. An important difference, however, is that transients can merge onto one successor state far from equilibrium in these discrete systems, whereas in continuous systems they cannot.

Neither does Hopfield's model⁸ support deterministically merging transients because his updating method is randomly asynchronous, and thus non-deterministic. It is open to debate whether synchronous or asynchronous updating in a local network is more or less biologically plausible. However, synchronous random networks have greater potential as content-addressable memory systems because not only attractors categorise state space. Sub-categories of state space are also categorised by a reliable time-series of unique states along each transient tree far from

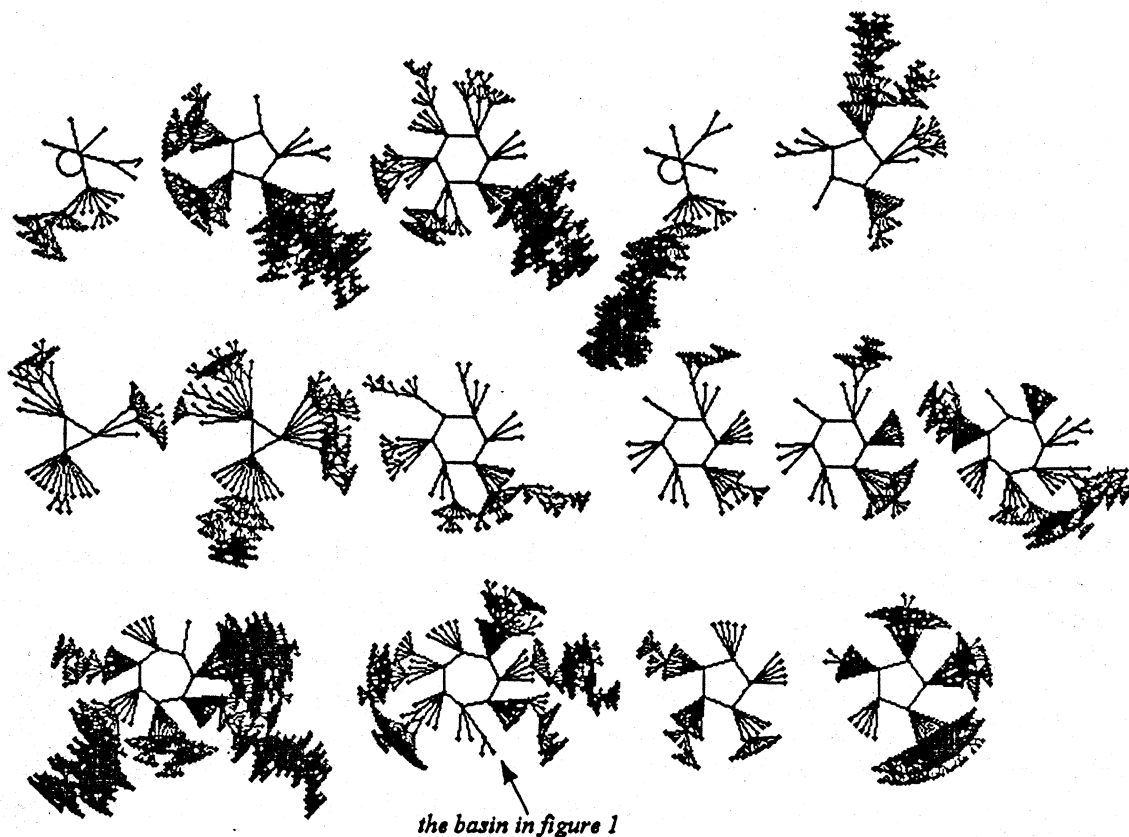


Figure 2.

The basin of attraction field a random Boolean network ($N=13$, $K=3$). The $2^{13}=8192$ states in state space are organised into 15 basins, with attractor period ranging from 1 to 7. The number of states in each basin is: 68, 984, 784, 1300, 264, 76, 316, 120, 64, 120, 256, 2724, 604, 84, 428. Figure 1 shows the arrowed basin in more detail, and the network's wiring/rule scheme.

equilibrium, creating what is effectively a complex hierarchy of content addressable memory.

The range of topologies of basins of attraction, and the potential for emergent complex categorisation of network states, suggests that the basin of attraction field, a mathematical object in space-time, is the network's cognitive substrate - *the ghost in the machine*³². A basic difficulty in explaining memory by attractors in biological networks has been the probably astronomical transient lengths needed to reach an attractor in large networks, whereas reaction times in biology are extremely fast. The answer may lie in the notion of memory far from equilibrium along merging transients³⁴.

CA (of whatever dimension) may be regarded as a special random network sub-class with homogeneous connectivity and rules. Evidence is presented that this *local* architecture is necessary for the emergence of coherent space-time structures such as gliders, the basis of CA models of artificial life. Random network architecture breaks these two basic premises, the *wiring/rule* scheme may be arbitrary and different at each cell, though divergence from CA architecture is a question of degree. An arbitrary wiring/rule scheme implies a vastly greater parameter space, and thus basin field configuration space than for CA. Perhaps any basin of attraction field configuration is possible. The process of adaptation and learning modifies the network's parameters, its wiring/rule scheme or size/connectivity, resulting in a modified basin of attraction field. The stability of the field under small perturbations to parameters is noteworthy.

This paper describes CA and random network architecture, and contrasts their dynamics in terms of space-time patterns and basins of attraction. The reverse algorithm for computing pre-images is explained. I suggest that random networks may provide a component for a biological model; in the context of many semi-autonomous weakly coupled networks, the basin field/network relationship may provide a fruitful metaphor for the mind/brain.

I describe learning algorithms that automatically re-assign pre-images in a single step. New attractors can be created and transient trees and sub-trees transplanted, *sculpting* the basin of attraction field to approach any desired configuration. The effects and side effects of learning become immediately apparent by re-drawing the modified basin of attraction field, or some fragment of it. Such *visible learning* may lead to useful applications as well as helping to clarify the process of memory and learning in a variety of artificial neural network architectures.

Basins of attraction

CA and random networks are both examples of discrete deterministic dynamical systems made up from many simple components acting in parallel. The dynamics is driven by the iteration of a constant global updating procedure (the *transition function*) resulting in a succession of global states, the system's *trajectory*. Given a noise free, deterministic transition function within an *autonomous* system (cut off from outside influence), any global state imposed on the network will seed a determined trajectory (though it may be unpredictable). In fact the system may be regarded as *semi-autonomous*, in the sense that a global initial state must be imposed or perturbed from outside to set the system going along a new trajectory. The system also needs a channel to communicate its internal state to the outside.

A trajectory is one particular path within a *basin of attraction*, familiar from continuous dynamical systems. In a finite network of size N and value range V there are V^N global states. Any path must inevitably encounter a repeat. When this occurs the system has entered and is locked into a state cycle (the *attractor*). Many trajectories typically exist leading to the same attractor. The set of all such trajectories, including the attractor itself, make up a basin of attraction. This is composed of *merging* trajectories linked according to their dynamical relationships, and will typically have a topology of branching transient trees rooted on the attractor cycle (though this may be a stable point - an attractor cycle with a period of 1).

Separate basins of attraction typically exist within state space. A transition function will, in a sense, crystallise state space into a set of basins, the *basin of attraction field*, a mathematical object in space-time which constitutes the dynamical flow imposed on state space. If represented as a graph the field is an explicit portrait of the network's entire repertoire of behaviour. It includes all possible trajectories.

Basins of attraction are portrayed as computer diagrams in the same graphic format as presented in "*The Global Dynamics of Cellular Automata*"³¹. Various other names are sometimes used, for example, flow graphs, state transition graphs, networks of attraction. Global states are represented by nodes, or by the state's binary or decimal expression at the node position. Nodes are linked by directed arcs. Each node will have zero or more incoming arcs from nodes at the previous time-step (*pre-images*), but because the system is deterministic, exactly one outgoing arc (one *out degree*). Nodes with no pre-images have no incoming arcs, and represent so called *garden of Eden* states. The number of incoming arcs is referred to as the *degree of pre-imaging* (or *in degree*).

Figure 1 shows a typical basin of attraction of a random Boolean network (it is part of the basin of attraction field shown in figure 2). Figure 4 shows the basin of attraction field of a CA

where many symmetries are evident, a major difference between the topologies of the two systems.

In the graphic convention, the length of transition arcs decreases with distance away from the attractor, and the diameter of the attractor cycle asymptotically approaches an upper limit with increasing period. The forward direction of transitions is inward from garden of Eden states to the attractor, which is the only closed loop in the basin, and then clockwise around the attractor cycle.

Typically, the vast majority of nodes in a basin of attraction lie on transient trees outside the attractor cycle, and the vast majority of these states are garden of Eden states. A transient tree is the set of all paths from garden of Eden states leading to a particular state on the attractor cycle. A transient sub-tree is the set of all paths from garden of Eden states leading to a state within a transient tree, as indicated in figure 1.

Computing transient trees or sub-trees, and basins of attraction, poses the problem of finding the complete set of pre-images of any global state. The trivial solution, exhaustive testing of the entire state space, rapidly becomes intractable in terms of computer time as the network's size increases beyond modest values. A *reverse algorithm* for 1-D CA, that directly computes the pre-images of a global state, with an average computational performance many orders of magnitude faster than exhaustive testing, was recently introduced³¹. Section 4 sets out a *general* direct reverse algorithm³² for random Boolean networks (which includes CA of arbitrary dimension), and which may be generalised for random networks with a greater value range.

Cellular Automata

A CA is sometimes described as a discretised artificial universe with its own local physics¹⁴. Space is a lattice of cells with a particular geometry; each cell contains a variable from a limited range (often just 0 or 1). All cells update synchronously as time advances in discrete steps. The updating rule is the same for all cells, and depends only on local relations, usually a closed symmetrical neighbourhood.

Conversely, one could say that the homogeneous neighbourhood template defines a given space, and if the CA is finite, implies periodic boundary conditions (i.e. a circle of cells for 1-D, a toroidal surface for 2-D). Finite 1-D CA architecture is illustrated in figure 3 where cells are arranged in a circle. Time steps are shown in sequence from the top down.

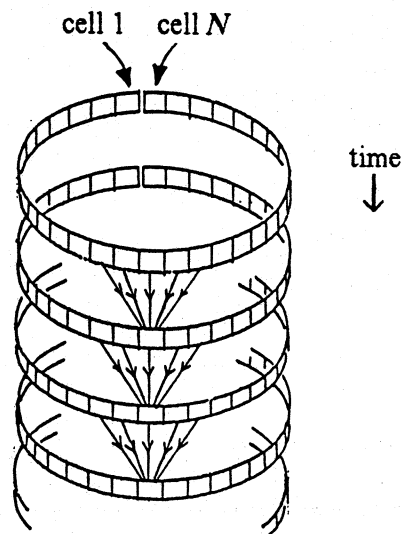


Figure 3.

1-D finite CA architecture, $K=5$, each cell has the same wiring template and rule. Boundary conditions are periodic by definition. The network is synchronously updated in discrete time-steps.

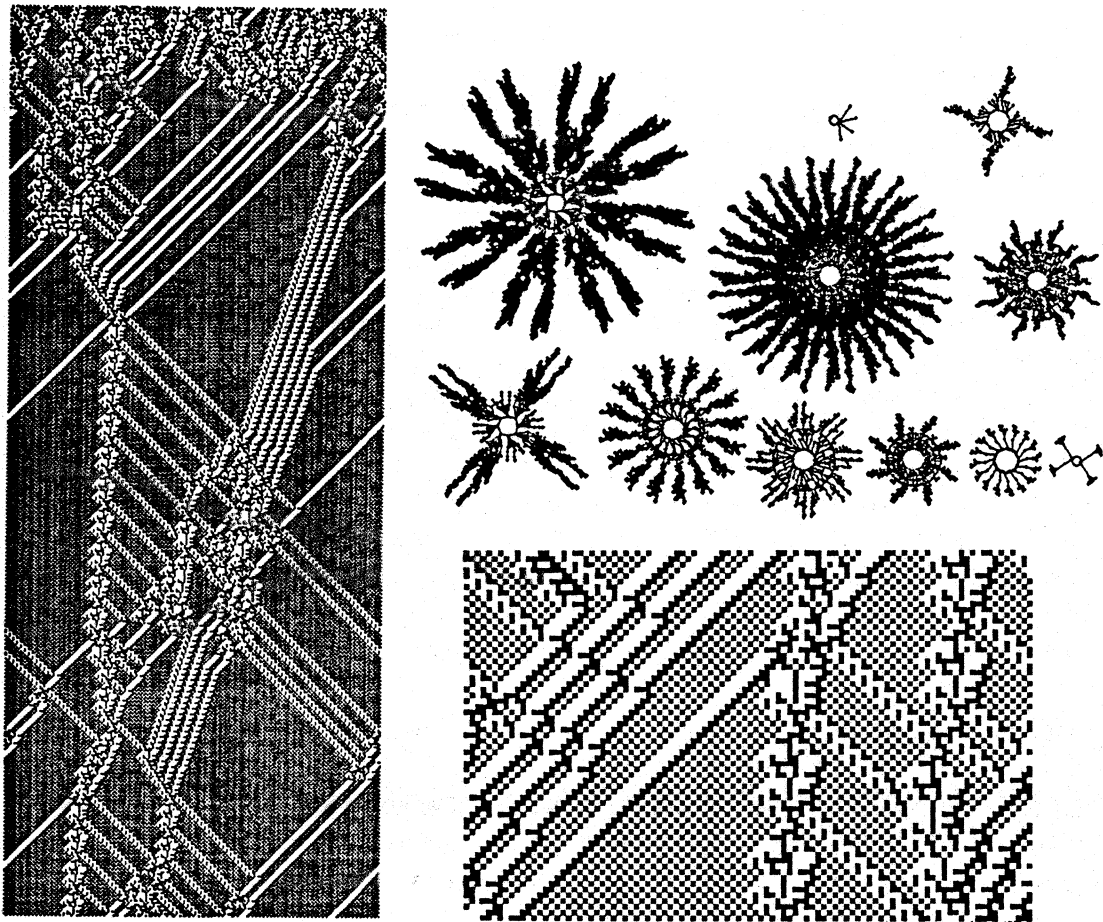


Figure 4.

An example of 1-D, $K=5$ complex CA dynamics, the rule number is 906663673.

Left. *the space-time pattern from a random initial state, $N=200$, 480 time-steps; gliders (and a glider-gun) emerge against a checkerboard background after an initial sorting out phase.*

Top right. *The basin of attraction field for the same rule, $N=16$. The 2^{16} states in state space are organised into 17 basins of attraction, only the 11 non-equivalent basins are shown. The number of states in each basin, and the number of each type, is as follows (from left to right), 30928(1), 6(1), 220(4), 23808(1), 1136(2), 1972(2), 2064(1), 568(1), 448(2), 144(1), 26(1). Note the basin symmetry due to equivalent transient trees.*

Bottom right. *A detail of typical glider interactions.*

The neighbourhood template (the *wiring scheme*) extends to the previous time step only; there are no connections to the more distant past.

Consider a periodic 1-D lattice with N cells and an odd number of connections, K . $[K/2]$ is the integer neighbourhood radius. The time evolution of the i -th cell is given by

$$C_i^{(t+1)} = f\left(C_{i-[K/2]}^{(t)}, \dots, C_{i-1}^{(t)}, C_i^{(t)}, C_{i+1}^{(t)}, \dots, C_{i+[K/2]}^{(t)}\right)$$

to satisfy periodic boundary conditions, for $x < 1$, $C_x = C_{N+x}$; for $x > N$, $C_x = C_{x-N}$

If the value range $= V$, there are V^K permutations of values in a neighbourhood of size K . The most general expression of the function f is a look-up table (the *rule table*) with V^K entries, giving

V^K possible rules. There are also sub-categories of rules that can be expressed as simple algorithms, Boolean derivatives²², totalistic rules or threshold functions. The number of effectively different rules is reduced by symmetries in the rule table^{24,31}. By convention²⁹ the rule table is arranged in descending order of the values of neighbourhoods, and the resulting bit string converts to the decimal rule number. For example, the rule table for rule 30 ($V=2, K=3$) is,

rule-table	111	110	101	100	011	010	001	000	neighbourhoods
	0	0	0	1	1	1	1	0	outputs (0 or 1)

Some CA support coherent space-time structures. Periodic self sustaining configurations, known variously as particles, solitary waves and gliders, may emerge and propagate across the lattice, as in Conway's 2-D *game of life*³. For simplicity I will call all such configurations *gliders* though their velocity may vary between zero and the system's 'speed of light', equal to the neighbourhood radius per time-step.

The emergence of gliders characterises complex rules, which are said to occur at a *phase transition* in rule space balanced between simple and chaotic behaviour¹⁴, the so called *edge of chaos*. From an initial random state, a limited number of glider types emerge after an initial sorting out phase, and continue to interact for an extended period of time. Collisions between gliders may produce new gliders that collide in their turn. So called *glider guns* may eject other gliders at periodic intervals. These complex interactions can encode logical operations supporting universal computation^{3,30,14}. Von Neuman's original self reproducing automaton¹⁹ and subsequent examples of glider reproduction have lead to the notion of artificial life^{13,15}.

Examples of such *edge of chaos* dynamics can be found in the simplest CA, for example the much studied $K=3$, 1-D, rule 110¹⁷. An example of a complex $K=5$, 1-D CA rule (from amongst many others³⁵) is shown in figure 4. 1-D CA dynamics is conveniently represented as a space-time pattern diagram. The cylinder in figure 3 is split between cells 1 and N and flattened out. A time-series of global states is represented as in figure 4, with space across (rows of black and white squares) and time running from the top down.

Previous work on the structure and topology of CA basins of attraction³¹ have shown that there are a number of general constraints to CA dynamics, which are reflected in their basin of attraction fields. They do not necessarily apply to random networks. The constraints relate to various symmetries and hierarchies within state space and rule space, summarised below.

1. Rotation symmetry (the number of repeating segments in the bit pattern) is conserved. In a transient, rotation symmetry can not decrease over time; in an attractor cycle, rotation symmetry must remain constant. In symmetric rules the same principles apply to bilateral symmetry.
2. Rotation equivalent states (that differ only by any rotation of the circular lattice) are embedded in equivalent behaviour. This results in basins of attraction with identical topology, but rotated states. Symmetries within basins occur if a sequence of rotation equivalent states repeat in the attractor cycle; transient trees with identical topology, but rotated states, will be rooted on the repeats.
3. Rule-space can be divided into symmetry categories by transformations within rule tables.
4. Equivalence classes and rule cluster relationships exist in rule space due to transformations between rule tables.

A parameter, Z , was developed in³¹ to capture correlations between a CA's rule table and its behaviour/basin field topology. The Z parameter is measured directly from the rule table and relates to Langton's λ parameter^{13,14} and the equivalent concept of *internal homogeneity* introduced earlier by Walker²³. Whereas λ simply counts the fraction of 1s in a binary rule table, Z takes into account the allocation of rule table values to sub-categories of related neighbourhoods. Behaviour predictions on the basis of Z avoid the exceptions characteristic of λ . The Z parameter predicts various interrelated aspects of basin field topology such as the degree of pre-imaging (the convergence of state space) and the density of garden of Eden states in state space. These in turn relate to attractor cycle and transient length, and the number and size of separate basins. The table below shows varying aspects of behaviour of increasingly large CA arrays as the Z parameter changes from 0 to 1 (Wolfram's behaviour classes^{29,30} are shown in brackets).

	<u>simple (class 1&2)</u>	<i>phase transition</i>	<u>chaotic (class 3)</u>
garden of Eden density	converges to 1	→ <u>complex (class 4)</u>	→ converges to 0
transient, attractor length	very short	balanced	very long
Z parameter (0 - 1)	≈ 0-0.6	moderately long	≈ 0.8 - 1
		≈ 0.6 - 0.8	

The variation of garden of Eden density against the Z parameter for 1-D CA is examined in³⁵. All $K=3$ rules, and $K=5$ totalistic rules are included, and a sample of complex $K=5$ rules. A strong inverse correlation was found. Complex rules are rare, but are most likely to occur in a distinct but broad band of Z . Related correlations may exist for random Boolean networks.

Random Boolean Networks

The idea of randomly connected multi-function networks as dynamical systems with a corresponding field containing all lines of behaviour can be traced back to Ross Ashby, in his book *Design for a Brain*². Random Boolean networks (sometimes called Kauffman's model), have been investigated for a considerable time by Stuart Kauffman in theoretical biology and complex systems^{9,10,11}, specifically to model gene regulation underlying embryonic development. Others have studied variations of random Boolean networks in the context of memory in the immune network, for instance²⁸, and in complex systems in general^{23-26,16}. The studies have built statistical data on network dynamics from many separate *forward* simulations. The focus of interest has often been to gain an insight into the topology of basin of attraction fields in relation to a range of possible system parameters.

Random Boolean network architectures are in many ways similar to weightless (or logical) neural networks¹, where standard memories (RAMs) hold each cell's look-up table. Classical neural network architectures use weighted connection and threshold functions. A random Boolean network may be regarded as a discrete generalisation of a sparsely connected classical neural network. Connections with higher weights may simply be replaced by multiple couplings, and the threshold function applied. However, a threshold function is a tiny sub-class of the V^K possible CA rules.

Random Boolean networks may be viewed as generalised (*disordered*) CA³², breaking two basic premises of CA architecture by allowing arbitrary wiring and/or rules at each cell. The effect on behaviour of deviating from either or both of these premises by degrees will be discussed below (see figures 8,9,10). Not surprisingly, coherent space-time patterns and emergent complex

structures such as *gliders*, characteristic of CA, are progressively degraded. A relatively small number of basins with low period attractors typically emerge.

Random Boolean networks have a vastly greater parameter space, and thus behaviour space, than CA. The various symmetries and hierarchies that constrain CA dynamics previously described need no longer apply. Consequently it might be conjectured that any arbitrary basin of attraction field configuration is possible given the right set of parameters. There is no limit to the speed of propagation across the network. The notion of space and the 'speed of light' lose significance as the homogeneous wiring/rule scheme of CA is progressively scrambled, though this can occur by degrees.

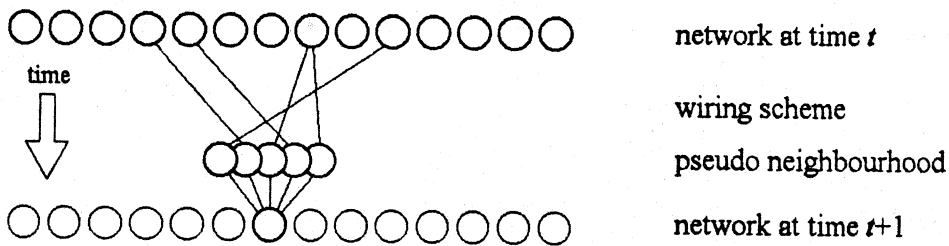


Figure 5. Random network architecture. Each cell in the network synchronously updates its value according to the values in a pseudo neighbourhood, set by single wire couplings to arbitrarily located cells at the previous time-step. Each cell may have a different wiring/rule scheme. The system is iterated.

A random Boolean network implies a value range of 2 (0 or 1), but in principle the arguments in this paper could equally apply to a network where cells have more than two values. As in CA, the global state of a network of N cells is the pattern resulting from values assigned to each cell, from a finite range of values V (usually $V=2$). Each cell synchronously updates its value in discrete time steps. The value of a cell C_i at $t+1$ depends on its particular CA rule, f_i , applied to a notional or pseudo neighbourhood, size K . Values in the neighbourhood are set according to single wire couplings to arbitrarily located cells in the network at time t . The system is iterated. The system's parameters are set by specifying the pseudo neighbourhood wiring and CA rule for each of the N cells. Each cell may have a different wiring/rule scheme (but not necessarily). Once set, the network's wiring/rule scheme is fixed over time.

The i -th cell C_i has its neighbourhood wiring connections chosen as $w_{i1}, w_{i2}, \dots, w_{iK}$. Connections are assigned to any of the N cells in the network, including C_i itself. Duplicate connections are allowed, giving N^K possible alternative wiring options. The i -th cell's rule f_i is chosen from V^{V^K} alternatives in rule-space. The time evolution of the i -th cell is given by

$$C_i^{(t+1)} = f_i \left(C_{w_{i1}}^{(t)}, C_{w_{i2}}^{(t)}, \dots, C_{w_{iK}}^{(t)} \right)$$

The number of alternative wiring/rule schemes that can be assigned to a given network turns out to be vast even for small networks, and is given by

$$S = (N^K)^N \times (V^{V^K})^N \quad \text{for example, a network where } V=2, N=16, K=5, \quad S = 2^{832}$$

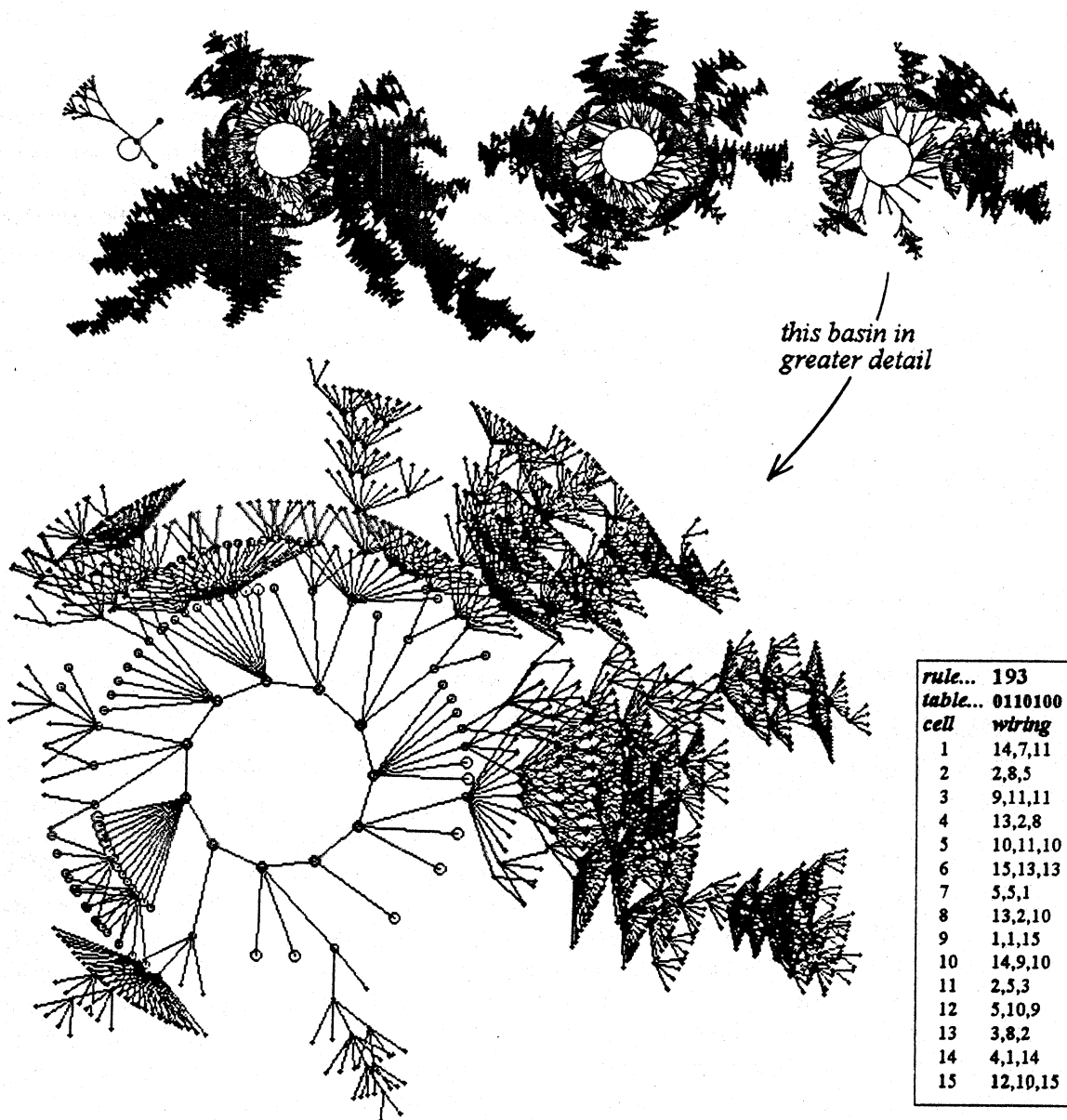


Figure 6.

Top Left. The basin of attraction field of a randomly wired, single rule network (non-local CA). $N=15$, $K=3$, rule 193. The $2^{15}=32768$ states in state space are organised into 4 basins of attraction. The total number of states in each basin is as follows (with attractor period in brackets): 24 (1), 26926 (22), 3498 (17), 2320 (11). The field has 27057 garden of Eden states, garden of Eden density = 0.823.

Bottom left. The last basin shown in greater detail (garden of E density = 0.823).

Bottom right. The pseudo neighbourhood wiring scheme.

Intermediate architecture

A spectrum of intermediate architectures is possible between CA and fully random networks as indicated in figure 7. The two main categories are a homogeneous rule but random wiring, or homogeneous wiring but a random rule mix. Within each category there are many possible variations.

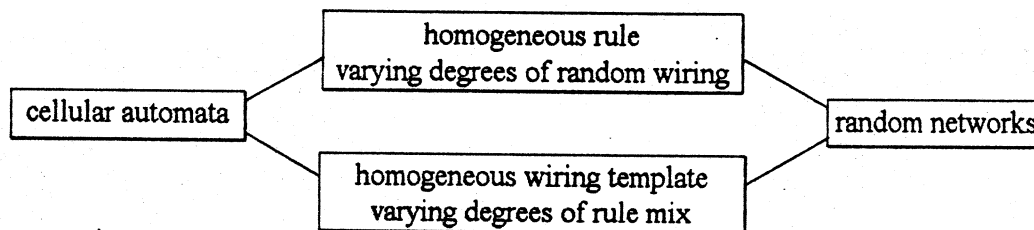


Figure 7. Intermediate architectures between CA and fully random networks.

Homogeneous rule - Random Wiring

A variety of constraints can be imposed on random wiring, for instance in Hopfield's model⁸ only symmetric wiring is allowed. Connections to a cell may be constrained to be distinct. Duplicate connections would correspond to unequal weights on single input lines¹⁰.

An example of the space-time patterns of a network with a homogeneous rule but fully scrambled wiring is given in the lower half of figure 8a. Space-time patterns appear random until the onset of periodic behaviour at the attractor. This is not surprising because totally random wiring destroys the continuity of space - geometrically the network becomes N -dimensional. However the pattern density (the density of 1s) tends to settle and fluctuate at a characteristic mean-field level, though bi-stability is also possible as in figure 9a. The mean-field pattern density probably relates to the rule's internal homogeneity or λ parameter.

Fluctuations are sometimes large with extended periods. The trace of the pattern density (as in figures 8b) is somewhat reminiscent of the EEG (electro-encephalogram) measure of the mean excitatory state of a patch of neurons in the brain.

Networks with partially and totally random wiring (non-local CA) have been studied by Li¹⁶. He found that complex rules may result in a form of edge of chaos dynamics in these systems, by the emergence of cooperations among a cluster of components, which he calls *coherent clusters*. He also found evidence that the magnitude of density fluctuations becomes smaller for larger systems.

Walker²³⁻²⁶ has studied basins of attraction by statistical methods for a $K=3$ neighbourhood where only the two outer wires were randomised (each cell is connected to itself). The system retains some notion of space. A spectrum of constraints on random wiring could be imposed in a similar way to retain a degree of spacial information in a network of given dimension and geometry. A varying proportion of wires from the neighbourhood could be set at random. Random wiring could be confined to a local periodic zone of a certain radius; this seems to produce sub-attractors as in figure 9b. Connecting each cell to itself could be mandatory, or excluded. Kauffman's random Boolean network models¹¹ generally do not constrain random wiring, except that the number of connections per cell, K , is constant, and often set at $K=2$.

To illustrate how coherent space-time structures are degraded by randomising *local CA* wiring, figure 8a shows the space-time pattern of a complex CA rule, $K=5$, $N=150$. Gliders emerge from a random seed. After approximately 240 time-steps, the wiring is totally randomised (but not the rule). All coherent space-time structure appears to be destroyed. Figure 8b shows the pattern density against time; irregular periodic fluctuations are apparent.

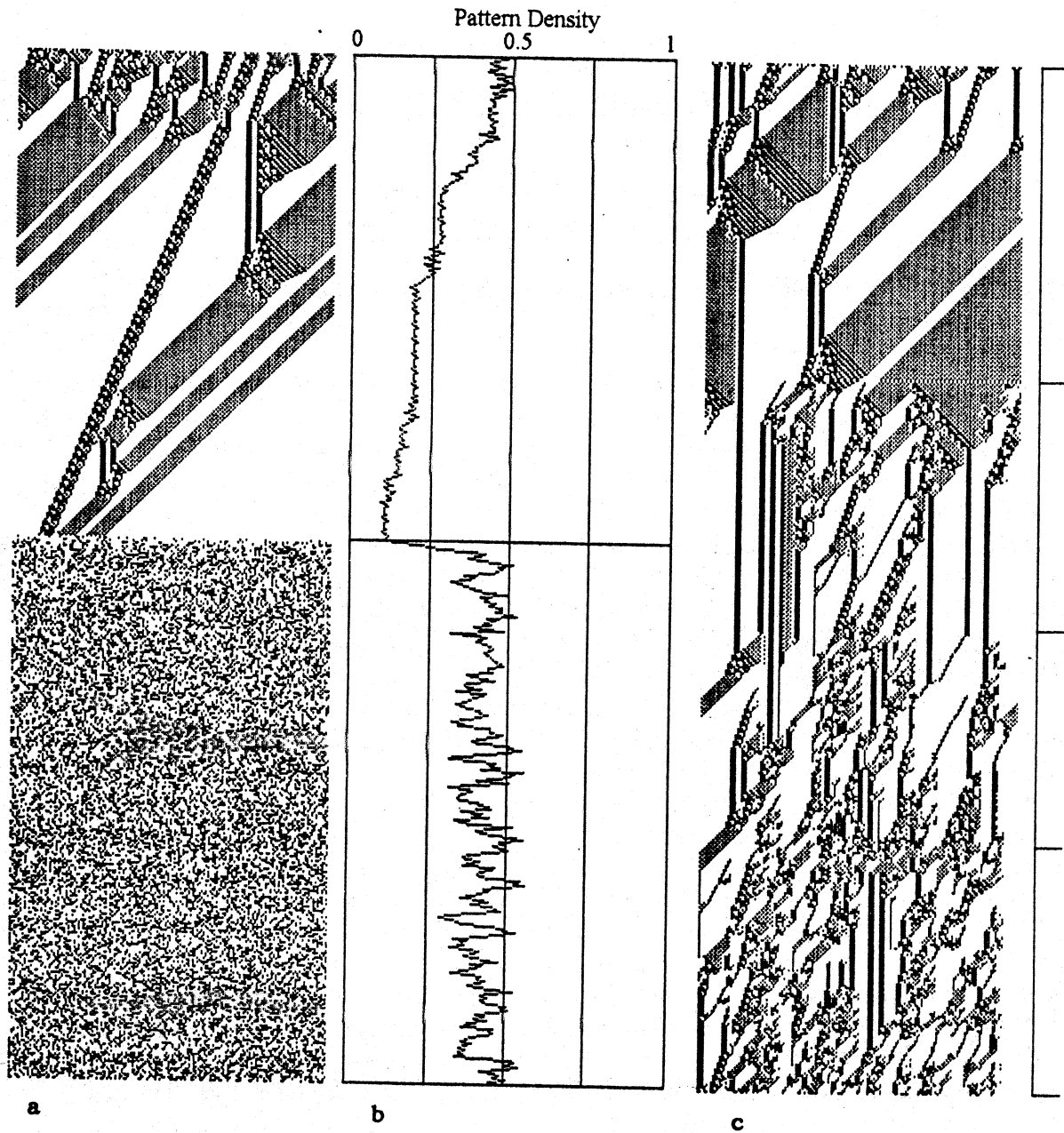


Figure 8. Single rule, local and random wiring. (non-local CA), $N=150$, $K=5$, rule 3162662612
 a) The space-time pattern of a complex rule with local wiring from a random initial state. After about 240 time steps, the wiring scheme has been totally randomised.
 b) Shows the pattern density (density of 1s) in successive global states analogous to an EEG.
 c) The space-time pattern of the same rule from another random initial state. At the times indicated, 2% of the wires (15 out of 750) are cumulatively randomised.

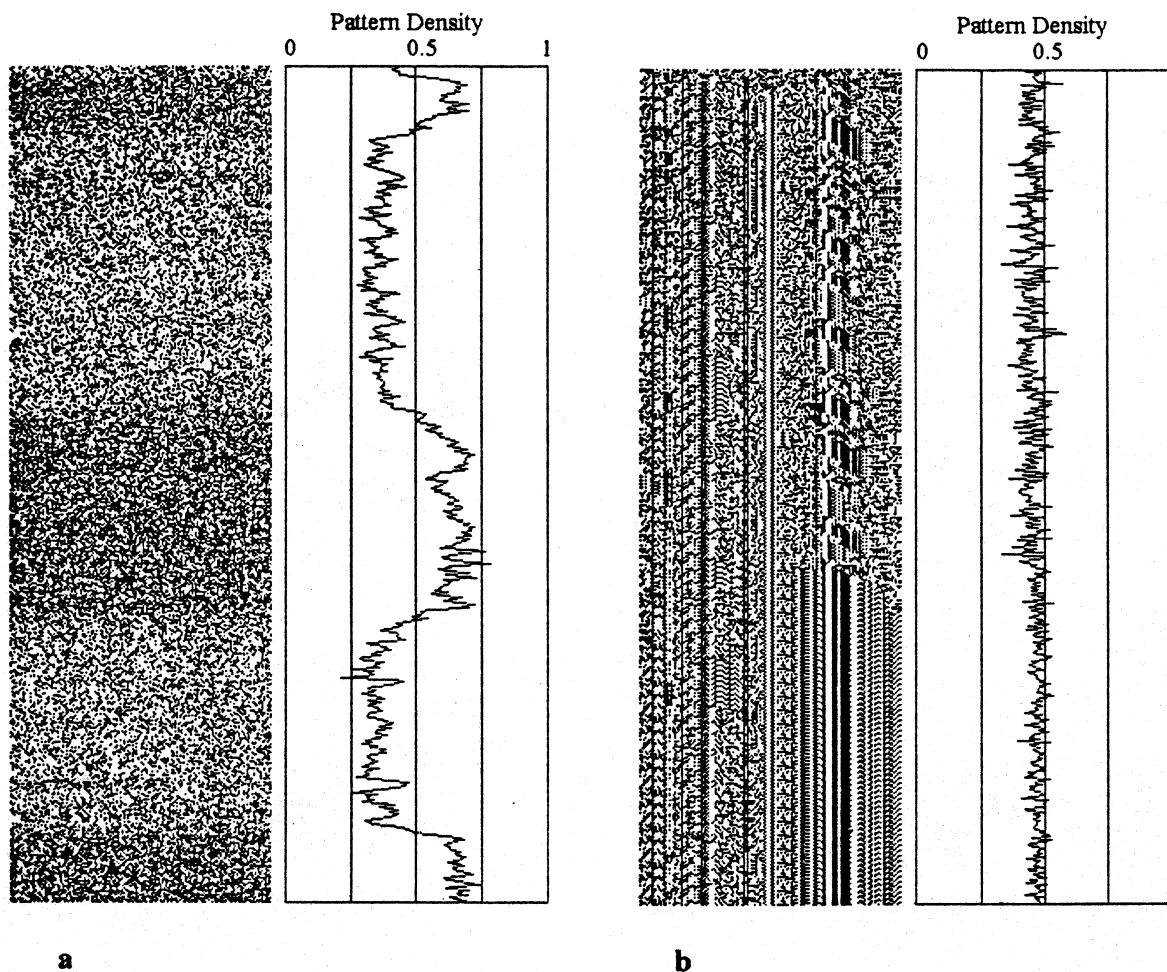


Figure 9. Examples of the space time patterns and pattern density plots in non-local CA (single rule/random wiring), $N=150$.

- a) $K=5$ rule 2129193089 with random wiring. This is a threshold rule set at 0.5 (majority rule) but with the end bits flipped, ie $111 \rightarrow 0$ and $000 \rightarrow 1$. Note the bi-stable pattern density.
- b) $K=3$ rule 193 with random wiring constrained to a 10 cell wide local periodic zone. Sub-attractors emerge.

Randomising the wiring of the CA in stages will progressively transform a structured space-time pattern to a seemingly random pattern. Figure 8c shows the space-time pattern of the same complex CA from another initial random state. At the intervals indicated, 2% of all available wires (randomly selected) are cumulatively randomised (15 wires out of 750). Coherent structure is progressively eroded. Eventually, the space-time pattern will look like the lower half of figure 8a.

Homogeneous Wiring - Random rule mix

A network's rule mix may be assigned at random, or the choice may be restricted to any combination of sub-categories of rules from rule-space; for instance rules with a particular setting of the Z parameter, only *additive* rules²⁹, only threshold functions, or only *canalizing* functions¹⁰.

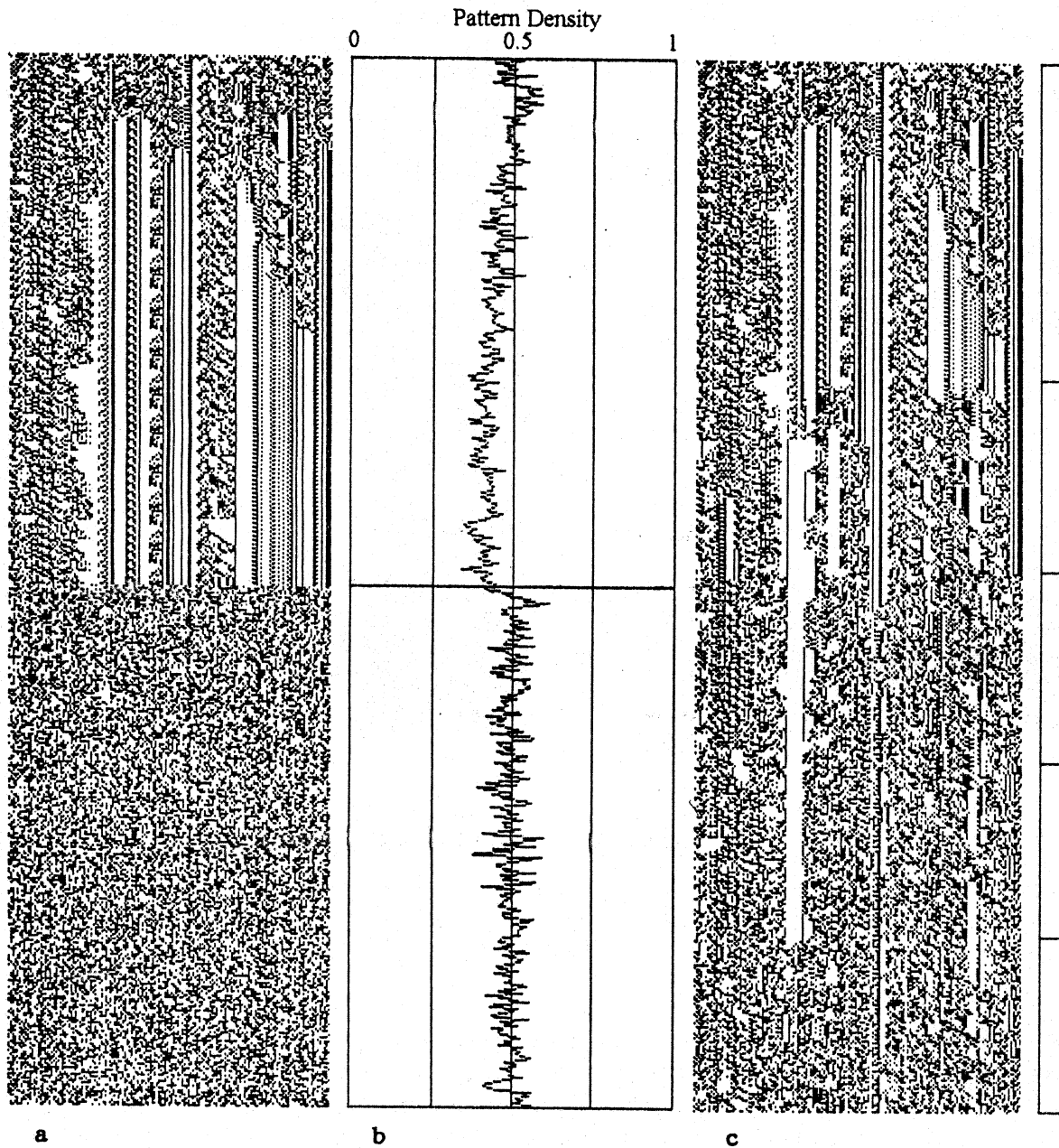


Figure 10. $K=5$ mixed rule network with local and random wiring, $N=150$

- a) The space-time pattern of with local wiring from a random initial state; sub-attractors (vertical features) emerge. After about 240 time steps, the wiring scheme is totally randomised. The space-time pattern becomes chaotic but some vertical features persist.
- b) Shows the pattern density (density of 1s) in successive global states analogous to an EEG.
- c) The space-time pattern of the same rule from another random initial state. At the times indicated, 4% of the wires (30 out of 750) are cumulatively randomised.

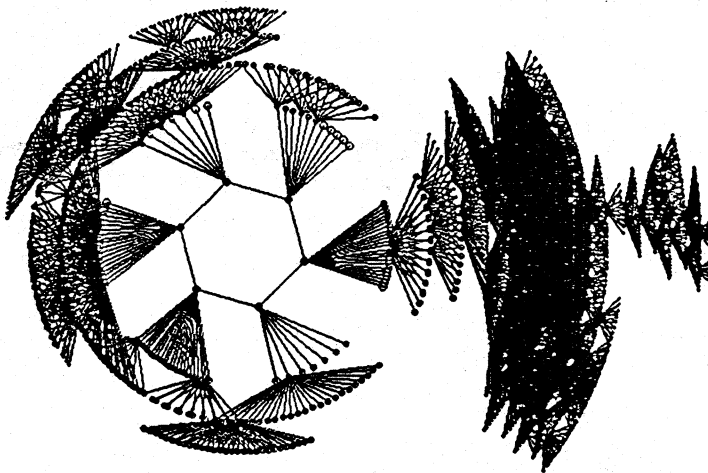
In homogeneous wiring/mixed rule networks, periodic structures confined within vertical bands will rapidly emerge within the CA space (frozen islands and isolated islands of variable elements¹⁰). A random rule mix tends to compartmentalise space, depending on the degree of rule heterogeneity; the vertical features are local sub-attractors. In random networks (with both random wiring and mixed rules) local vertical features rapidly emerge for $K=2$ rules due to frozen islands, but become less dominant as K increases. At $K=5$, space-time patterns appear chaotic, but with some residual vertical features (see figure 10a). The mean-field pattern density probably relates to the *mean* internal homogeneity or λ parameter of all the rules in the network, corrected to allow for the proportion of output couplings from each cell.

Figure 10a shows the space-time pattern of a homogeneous wiring/mixed rule network. Frozen islands and local sub-attractors emerge from a random initial state. After approximately 240 time steps, the wiring has been totally randomised (but not the rule scheme), resulting in the loss of all coherent structure, though some vertical features are evident. Figure 10b shows the pattern density. Figure 10c shows the space-time pattern of the same homogeneous wiring/mixed rule network from the same initial random state. At the intervals indicated, 4% of all available wires (randomly selected) are cumulatively randomised (30 wires out of 750). Frozen island structure is progressively eroded. Eventually, the space-time pattern will look like the lower half of figure 10a.

Comparative Global Dynamics, CA \leftrightarrow random networks

Although there is much work to be done given the enormous behaviour space open to investigation, a few general observations can be made on the basin of attraction field topology of random Boolean networks, and intermediate architectures, on the basis of many computer runs to reconstruct basins of attraction to date.

The various symmetries and hierarchies that dominate CA basin field topology³¹, as in figure 4, are absent in random Boolean networks; though some symmetries are still evident in small randomly wired networks with a single homogeneous rule, or a limited rule mix. This lack of constraint on the basin of attraction field topology, and their enormous parameter space, suggest that random networks (or a network of networks) are an ideal vehicle for emergent *categorisation*, and might be described as having the potential for *brain-like* behaviour.



cell	wiring	rule,-table
1	8,13,10	51, 00110011
2	4,4,3	55, 00110111
3	13,10,6	11, 00001011
4	8,12,8	166, 10100110
5	3,3,3	77, 01001101
6	1,9,9	182, 10110110
7	13,11,2	122, 01111010
8	6,2,11	188, 10111100
9	4,2,2	207, 11001111
10	9,12,6	112, 01110000
11	2,9,6	6, 00000110
12	11,13,9	63, 00111111
13	6,10,6	56, 00111000

Figure 11. The basin of attraction field of a random Boolean network with only one attractor, period 6. $N=13$, $K=3$. The $2^{13}=8192$ states in state space are all linked into one basin of attraction. The network's wiring/rule scheme is set out in the table.

On the other hand, CA are discrete approximations of physical systems¹⁵, characterised by *local* connections. Just as the physical world has the potential for the spontaneous emergence of life, under certain *edge of chaos* conditions CA seem to support the spontaneous emergence of analogous complex dynamical phenomena such as gliders. In turn, a key property of life is the emergence of complex networks with *non-local* connections; biological systems are replete with non-local connections, from brains to economics.

The basin structure of random Boolean networks is extremely varied, but for parameters set at random, the number of basins and their attractor periods is generally very small and increases only slowly with system size, whereas in some categories of CA the growth may be exponential. Examples of just one attractor with a short period taking up the entire state space are not uncommon, as the example in figure 11 for a $N=13$, $K=3$ network.

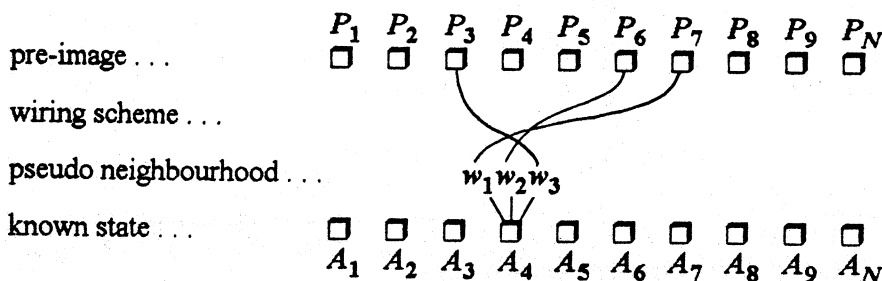
The surprisingly small number of attractors with small periods ties in with Kauffman's studies¹¹ for large random Boolean networks (for N up to 10,000). He reported that for $K=2$ networks, attractor cycles and the number of alternative attractors increased as \sqrt{N} , and at an increasing rate with greater K .

A systematic investigations of the topology and structure of basins of attraction for random Boolean networks and various intermediate architectures, using the tools now available, has yet to be done.

Computing Pre-Images in random Boolean networks

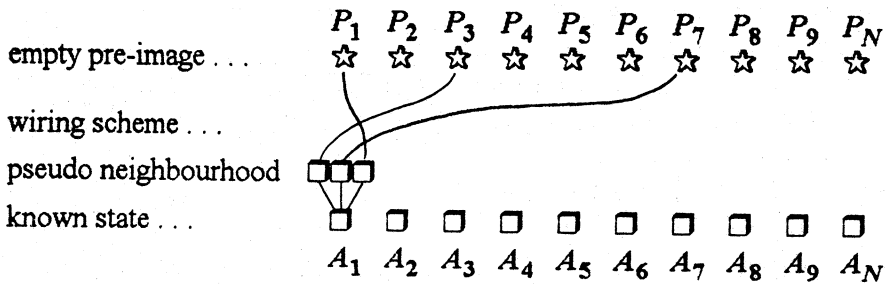
The construction of a single basin of attraction poses the problem of finding the complete set of pre-images of a given network state. The trivial solution is to exhaustively test the entire state space. Every state that is linked together in the basin would require exhaustive testing. This obviously becomes intractable in terms of computer time as the network's size increases beyond modest limits. To overcome this problem, methods have been invented for computing pre-images directly, without exhaustive testing. The network's dynamics can, in effect, be run *backwards* in time; backward trajectories will, as a rule, diverge. A *reverse algorithm* that directly computes pre-images for one-dimensional CA was presented in "*The Global Dynamics of Cellular Automata*"³¹. I set out below a *general* direct reverse algorithm for random Boolean networks³².

Consider a random Boolean network with N cells, with a pseudo neighbourhood size K . The algorithm will be demonstrated for $N=10$, $K=3$. An equivalent algorithm applies for larger N and K . For convenience, the system is represented as a 1-D array, $A_1, A_2, \dots, A_i, \dots, A_N$. Each cell, A_i (value 0 or 1), has a pre-set wiring/rule scheme (possibly selected at random). The wiring scheme is given by, $A_i(w_1, w_2, w_3)$ where w_1 is a number between 1 and N signifying the position of the wire connections from first branch of the pseudo neighbourhood, and so on. The rule scheme is given by, $A_i(T_7, T_6, \dots, T_0)$, the $K=3$ rule table. In the example below the wiring scheme for A_4 is $w_1=7$, $w_2=6$ and $w_3=3$.



To derive the pre-images of an arbitrary global state, consider a candidate pre-image as an *empty* array, consisting of *empty* cells. The cells are empty because their values are unknown, and unallocated as either 0 or 1. Empty cells are denoted by the wild card symbol \star , known cells (with values established as 0 or 1) are denoted by the symbol \square .

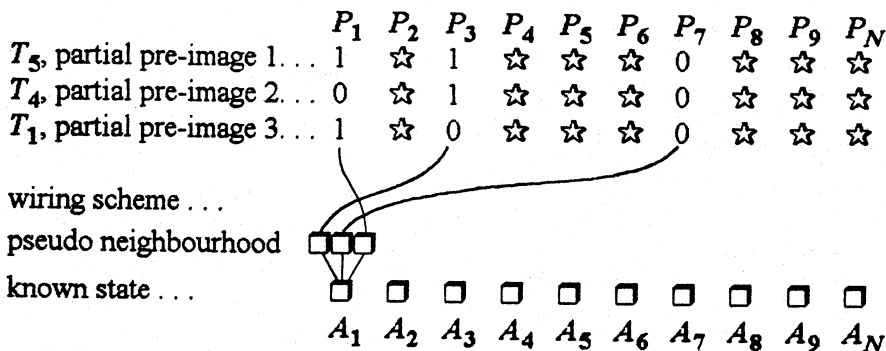
Consider a known network state, A_1, A_2, \dots, A_N and the empty pre-image state P_1, P_2, \dots, P_N .



Starting with the first cell of the known state, A_1 , the valid pseudo neighbourhood values, consistent with the value of A_1 are assigned to separate copies of the empty pre-image according to the wiring scheme $A_1, (w_1, w_2, w_3)$. As there will be a mix of 0s and 1s in the rule table, only some of the 8 possible pseudo neighbourhoods will be valid. If, say, 3 are valid, 3 *partial pre-images* (with some cells *known*, and some *empty*) will be generated. For example, given the $K=3$ rule 50 at A_1 , with a rule table as follows,

rule-table...	111	110	101	100	011	010	001	000	...neighbourhoods
	0	0	1	1	0	0	1	0	...outputs (0 or 1)
	T_7	T_6	T_5	T_4	T_3	T_2	T_1	T_0	

If $A_1=1$, then only 3 outputs match A_1 , T_5 , T_4 and T_1 , corresponding to the neighbourhoods 101, 100 and 001. These valid neighbourhoods are allocated to 3 empty arrays according to the wiring scheme, say $A_1(3,7,1)$. Each of the 3 arrays now have some of their cells allocated as 0s or 1s, and are termed *partial pre-images*, as illustrated below.



The procedure continues with the next cell of the known state, A_2 (though the order may be arbitrary). Say that the value of A_2 has 5 (out of 8) valid pseudo neighbourhoods in its rule table, $A_2(T_7, T_6, \dots, T_0)$. The pseudo-neighbourhoods are allocated to 5 copies of each of the partial pre-images that were generated at A_1 , according to the wiring scheme $A_2(w_1, w_2, w_3)$.

If the allocation of a value to a given cell conflicts with the value already assigned to that cell, then the partial pre-image is rejected. Otherwise, the partial pre-image is added to the partial pre-image stack. The allocation will be valid if it is made to an empty cell, or to a known cell with an equal value. Valid allocation increases the size of the partial pre-image stack, conflicts reduce the size of the stack.

This procedure is repeated in turn for the remaining cells, $\dots A_3, A_4, \dots, A_N$. At each successive cell, more partial pre-images may be added to the stack, but also rejected. The size of the stack will typically vary according to a Gaussian distribution. If the stack size is reduced to zero at any stage then the known state A_1, A_2, \dots, A_N has no pre-images; it is a *garden of Eden* state. In general, the vast majority of states in state space turn out to be garden-of-Eden states.

Note that the order in which cells in A_1, A_2, \dots, A_N are taken is entirely arbitrary. For the most efficient computation, that minimises the growth of the partial pre-image stack, the order should correspond to the greatest overlap of wiring schemes.

When the procedure is complete, the final stack may still have empty cells, signifying that these cells are not sampled by any wiring couplings. Final stack arrays with empty cells are duplicated so that all possible configurations at empty cell positions are represented. The resulting pre-image stack is the complete set of pre-images of the given state, without duplication. An equivalent, but extended, procedure is used for $K=5$ rules. In this case the wiring/rule scheme is specified for each cell in the array A_1, A_2, \dots, A_N as follows,

the wiring scheme $\dots A_i(w_1, w_2, w_3, w_4, w_5)$ the rule scheme $\dots A_i(T_{31}, T_{30}, \dots, T_0)$

The general reverse algorithm for computing pre-images works for fully random networks or any degree of intermediate architecture, which of course includes CA of any dimension - the wiring scheme is simply set accordingly. In principle, the algorithm will work for any size of pseudo neighbourhood, K , and any value range, V . Provided that $K < N$, the algorithm is many orders of magnitude faster than the exhaustive testing of state space, the only previous method available.

This information is used to construct the pre-image *fan*, from the given state to its set of pre-images (if any). The pre-image fan for each pre-image is then computed, and so on, until only garden of Eden states remain. In this way transient trees (or sub-trees) may be constructed. Basins of attraction (or the complete basin of attraction field) are constructed by first running the network forward to reveal the attractor cycle, then computing each transient tree in turn. The method, and the graphic conventions, are explained more fully in ³¹.

Memory, far from equilibrium.

Memory far from equilibrium along merging transients may answer a basic difficulty in explaining memory by attractors in biological neural networks. A view of the brain as a complex dynamical system made up of many inter-linked specialised neural networks is perhaps the most powerful paradigm currently available. Specialised neural networks may consist of further sub-categories of semi-autonomous networks, and so on, which contribute to re-setting or perturbing each other's global state. A biological neural sub-network is nevertheless likely to be extremely large; as a dynamical system the time required to reach an attractor from some arbitrary global state will probably be astronomical. This may be demonstrated with a simple $K=5$ random Boolean network with a few hundred cells. Even when an attractor is reached, it may well turn out to be a long cycle or a quasi-infinite chaotic attractor. The notion of memory simply as attractors seems to be inadequate to account for the extremely fast reaction times in biology.

A discrete dynamical system with synchronous updating categorises its state space reliably along transient trees, far from equilibrium, as well as at the attractors. A network that has evolved or learnt a particular global dynamics may be able to reach memory categories in a few steps, possibly just one. Moreover, the complex transient tree topology in the basin of attraction field, makes for a much richer substrate for memory than attractors alone, allowing hierarchies of memory sub-categories.

Deterministic transient tree topology, as described in this paper, requires synchronous updating in the network, though *asynchronous* updating might conceivably produce analogous phenomena provided that the updating was deterministic, and not random as in Hopfield's model⁸. Deterministic asynchronous updating could be implemented by adding an extra parameter to a random network model; the *length* of each wire coupling as proposed by Harvey *et al*⁷, possibly in discrete intervals. A standard speed of signal transmission applied to a network with variable discrete length wires is equivalent to a network with only single length wires, but couplings to network states *before* the previous time step, for example a cell at time t might have connections to cells at $t-2$, $t-3$,... as well as to $t-1$.

A difficulty with this model is that a given network state at a particular instant may have multiple successors, and would be likely to occur many times in an enlarged state space. Alternatively, the states in the system's state space would need to consist of multiple time-steps, not just one. Network architecture with 'higher order in time'³⁰ or 'historical time reference'³¹ may be worth pursuing but is beyond the scope of this paper.

There is evidence that the firing of nearby biological neurons is strictly related in time. Phase locking of spike discharges between neighbouring cells has been observed, extending up to 7mm across the cortex^{4,21,27}. Synchronous firing may be mediated by interneurons, which lack axons²⁰, or mechanisms relying on close physical proximity between neurons (their dendrites, cell bodies and axons). Gap junction effects (physical connections between neurons made by large macro molecules), and ephaptic interactions (the local electrical field) serve to synchronise local neuronal activity¹⁸.

A biological model

A random Boolean network may serve as a model of a patch of semi-autonomous biological neurons whose activity is synchronised. A cell's wiring scheme models that sub-set of neurons connected to a given neuron. Applying the CA rule to a cell's pseudo neighbourhood models the non-linear computation that a neuron is said to apply to these inputs to decide whether or not to fire at the next time-step. This is far more complex than a threshold function²⁰. The biological computation may be a function of the topology of the dendritic tree, the microcircuitry of synaptic placements and intrinsic membrane properties. Networks *within* cells based on the cytoskeleton of microtubules and associated protein polymers may be involved, suggested by Hameroff *et al*^{5,6} as the neuron's 'internal nervous system'. There appears to be no shortage of biological mechanisms that could perform the role of a CA's rule-table.

The cells in a random Boolean network are arranged in an orderly array for convenience only, but their location may be arbitrary. One presumes that the actual location of neurons in the brain has been optimised through evolution to achieve high density by minimising the average length of connections that occupy space and consume resources.

The network has been re-drawn as a brain-like model in figure 13; a semi-autonomous population of 27 idealised neurons distributed in 3 dimensions. Each neuron (figure 12) receives a post-synaptic excitatory (1) or inhibitory (0) signal from up to 5 neurons in the population

Figure 12
A cell of a random Boolean network represented as an idealised neuron with an overriding pre-synaptic contact from outside the system, and a projection axon to other networks.

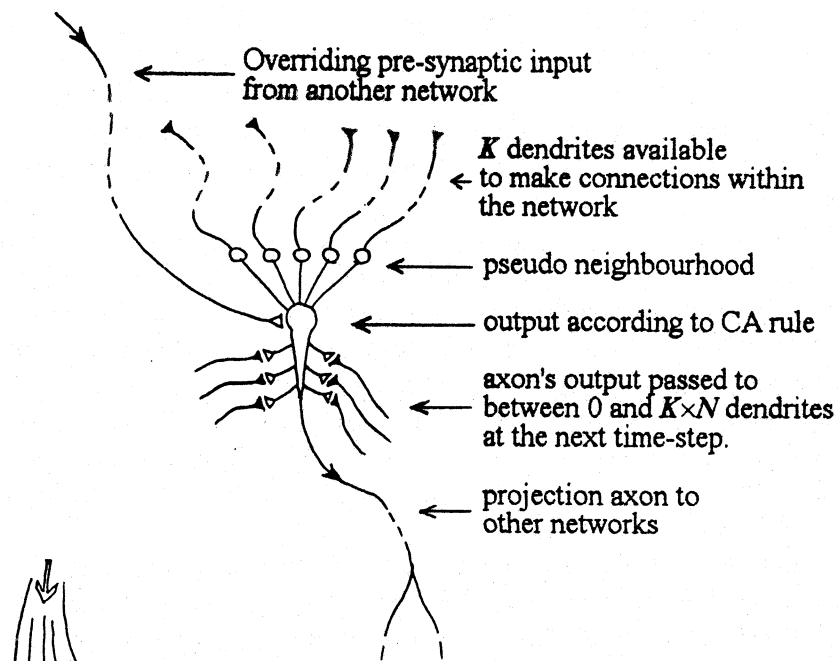
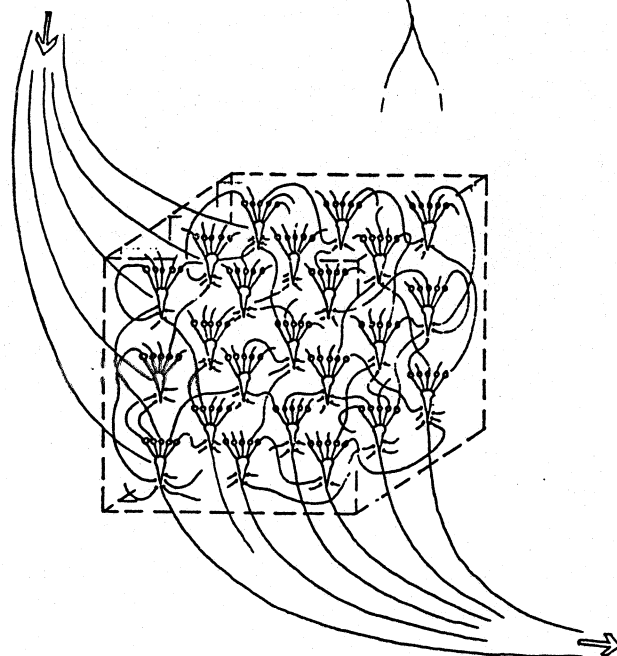


Figure 13
A random Boolean network ($N=27$, $K=5$) represented as a semi-autonomous population of idealised neurons. Neurons may have input and output links to other networks.



(possibly including itself) via its 5 dendrites ($K=5$), and computes a response signal to its axon according to its particular 32 bit rule-table applied to its pseudo-neighbourhood. The updating of axonic response is synchronous, and the process is iterated in discrete time steps.

The network's basin of attraction field is implicit in its wiring rule/scheme. In a patch of biological neurons it would be implicit in the wetware. Recognition (categorisation) of a global state is automatic and direct according to the particular deterministic trajectory (within the transient tree of the basin of attraction) that will inevitably follow. Learning new behaviour implies amending the basin of attraction field by adjusting the wiring/rule scheme, analogous to some physical change to the wetware's neural architecture and synaptic function. How the appropriate change might occur in biological networks to achieve a desired novel behaviour is unknown. Hameroff *et al*^{5,6} suggest that cytoskeletal functions may provide retrograde signalling (analogous to back propagation in artificial neural networks) which may reconfigure intra-neuronal architecture.

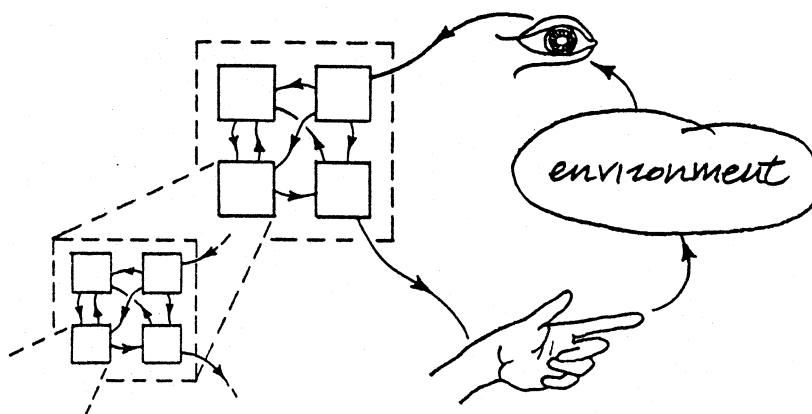


Fig 14. A network of weakly coupled semi-autonomous random Boolean networks linked to the environment as a biological model. Each network may consist of a nested hierarchy of networks.

The system is *semi*-autonomous because it must be capable of receiving input from outside to reset its global state, and also to communicate its current internal state to the outside via feed forward channels. Thus each neuron in the model has overriding postsynaptic inputs from outside the population, and a projection axon to targets outside the population.

The model may be elaborated by weakly inter-connecting a number of random Boolean networks (or perhaps 3-state networks), so that the output of a particular network constituted the overriding input of another. Communication between networks may be *asynchronous*, and at a slower frequency than a particular network's internal synchronous clock. Such an assembly of networks will have implicit in its particular pattern of connections at any instant, a vastly more complex but intangible web of interacting basin of attraction fields - *the ghost in the machine* ?

Learning Algorithms

Whether or not such a model is biologically plausible, it may be useful in its own right as an artificial neural network where learning (and its side effects) is visualised as alterations to the detailed structure of the basin of attraction field. In networks too large to allow basins, or even fragments of basins, to be computed, the principles still apply. Separate basins in the basin of attraction field, and each node onto which dynamical flow converges, categorise state space. All the network's states other than *garden of Eden* states are content-addressable memories. Any external input will automatically initiate a dynamical flow along a unique chain of states. Each successive state categorises states in its transient sub-tree, far from equilibrium, forming a complex hierarchy of categorisation culminating at the attractor. The set of attractors and their branching trees constitute the network's collective memory.

This section sets out learning algorithms that enable a random Boolean network to learn new transitions from experience (and also to forget). Suppose we want to make the state P_1 the pre-image of state A . Any mismatches between cell values of the actual successor state B_1 (of the aspiring pre-image, P_1) and state A can be corrected in one step by either of two methods, adjusting the network's wiring or rule scheme. The two methods have very different consequences.

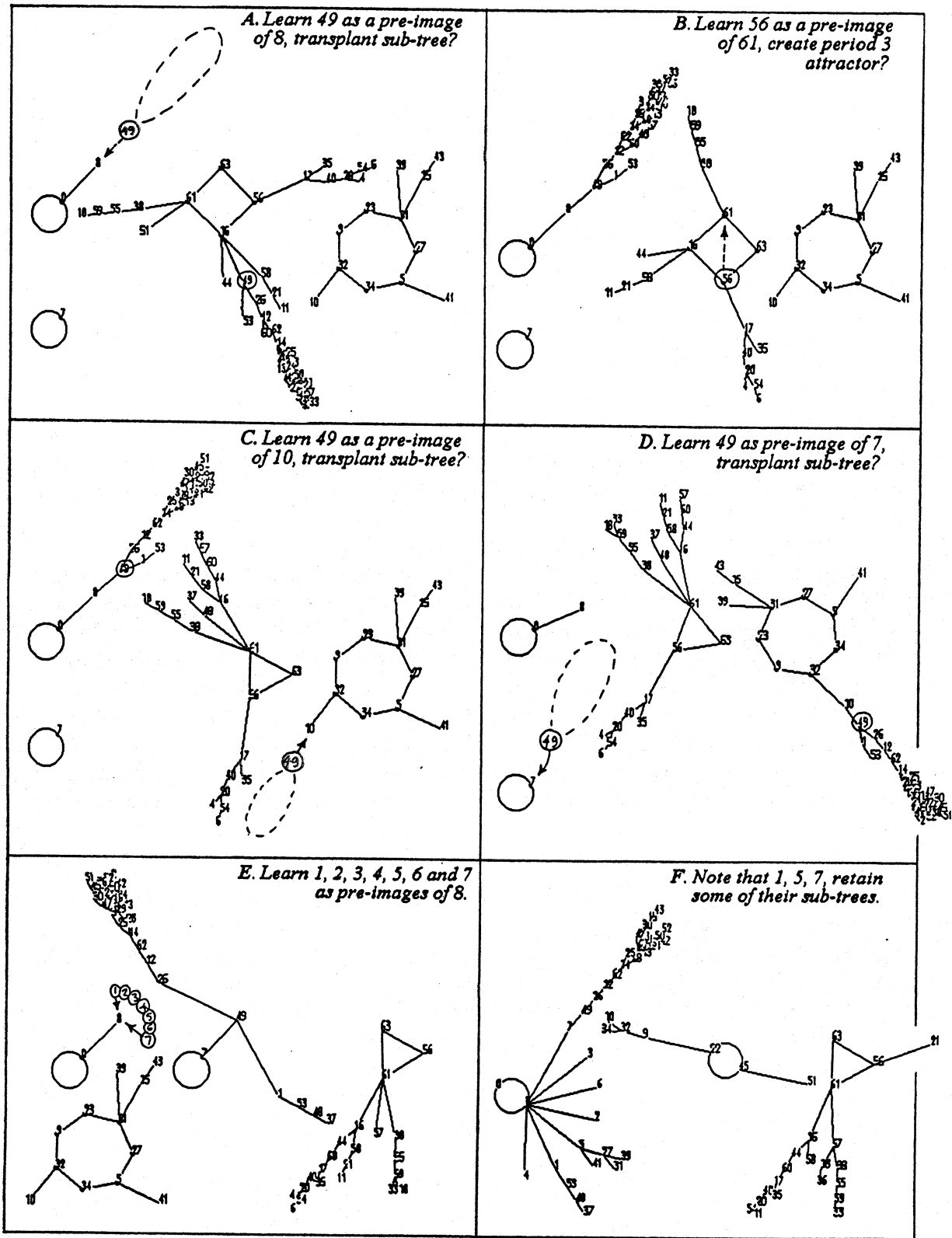


Figure 15. A sequence of learning steps, A to F, in a 6 cell network. Note the stability of basin structure as well as the side effects of learning at each step. Bit patterns are represented by decimal numbers.

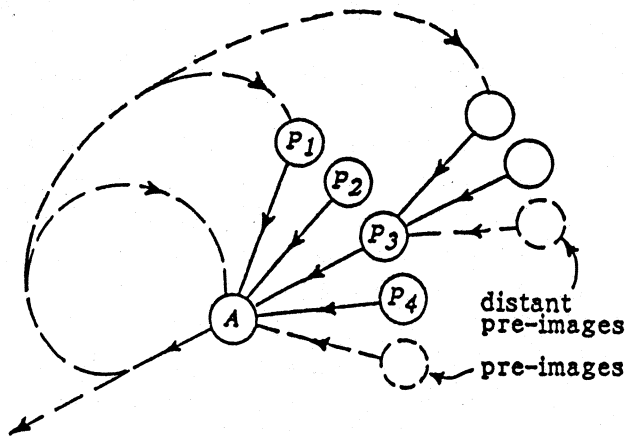
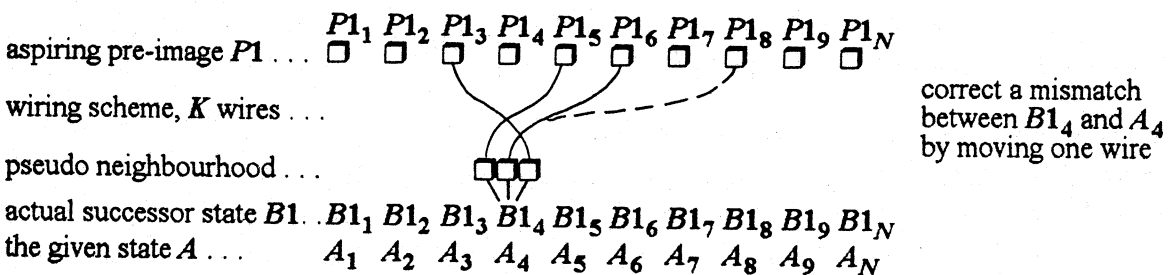


Figure 16. States P_1, P_2, P_3, \dots etc may be learnt as pre-images of the state A . Distant pre-images of A may also be learnt, for instance as pre-images of P_3 . Learning A as a pre-image of itself creates a point attractor. Learning A as a distant pre-image of itself creates a cyclic attractor. If A is learnt as the pre-image of some other state in the basin of attraction field, the states flowing into A , its transient sub-tree, may be fully or partially transplanted along with A .

Before learning starts, a wiring/rule scheme must already be in place. If the relevant transitions in the basin of attraction field are already close to the desired behaviour, the side effects of learning will be minimised. The wiring/rule scheme may be selected from an atlas of basin of attraction fields, such as in³¹. It may be pre-evolved from a population of wiring/rule schemes using a genetic algorithm, or in the worst case assumed at random.

Learning by re-wiring

Consider the state $P1$, $(P1_1, P1_2, \dots, P1_N)$, that the network is to learn as a pre-image of the given state A , (A_1, A_2, \dots, A_N) . When $P1$ (the aspiring pre-image) is evolved forward by one time-step according to its current K neighbour wiring/rule scheme its actual successor state is $B1$, $(B1_1, B1_2, \dots, B1_N)$. If the states $P1$ and A were selected at random, then $B1$ will probably have mismatches with A in about $N/2$ locations.



Suppose that $B1_4 \neq A_4$. One wire, or more if necessary) of the cell's wiring couplings is moved to a new position. We will limit this analysis to single wire moves only. Any move resulting in a pseudo neighbourhood with opposite output (according to that cell's look up table) will correct the mismatch. This is a stochastic method as there are likely to be many alternative successful options.

Assuming that $P1$ has a roughly equal proportion 0s and 1s, there will be $\approx N/2$ alternative positions where a wire move will pick up a changed value. Given K wires there will be $\approx K \times N/2$

alternative re-wiring options that will change the pseudo neighbourhood. Assuming the rule at cell 4 has $\lambda \approx 0.5$, (the proportion of 1s in the rule table), $\approx 1/2$ of the changed pseudo neighbourhoods will change the value of $B1_4$ so that $B1_4 = A_4$. The number of valid re-wiring options to correct each mismatch is therefore $\approx K \times N/4$. The choice for re-wiring may initially be selected at random from among the valid options to correct each mismatch making $P1$ the pre-image of A .

Suppose that another aspiring pre-image, $P2$ (with an actual successor $B2$) is to be learnt as a pre-image of A by re-wiring, but without forgetting $P1$. If $B2_4 \neq A_4$ there will be $\approx K \times N/4$ valid re-wiring options to correct the mismatch. However $\approx 1/4$ of the options would change the value of $B1_4$, forgetting $P1$ ($\approx 1/2$ the wire moves would change the pseudo neighbourhood of $B1_4$, $\approx 1/2$ of which will have outputs different to $B1_4$).

If several pre-images are to be learnt in succession by single wire re-wiring, without forgetting any previously learnt pre-images, the space of valid re-wiring options for a particular cell will be reduced by $\approx 1/4$ for each mismatch correction. The shrinking of the valid re-wiring space is sensitive to the initial re-wiring choices, and the order of learning. Some re-wiring choices will be fitter than others, affecting the capacity of the network to learn that particular set of pre-images.

If the pre-images are close to each other in terms of Hamming distance, there will be fewer mismatches to correct thus greater learning capacity. Close pre-images are likely to be learnt by default; the network is able to generalise from examples.

What is the probability of *forgetting* pre-images on the same fan, or transitions elsewhere in the basin of attraction field, as a result of learning by re-wiring?

Consider an arbitrary transition $X \rightarrow Y$ elsewhere in the basin. The probability of *forgetting* the transition by moving one wire is given by,

$$F_1 = 1/4$$

$$\text{by moving two wires, } F_2 = F_1 + F_1(1 - F_1)$$

$$\text{by moving } n \text{ wires, } F_n = \sum_{i=0}^{n-1} \left(1 - \frac{1}{4}\right)^i \times \frac{1}{4}$$

The capacity of the network to learn more states as pre-images of a given state will thus depend on a number of factors: the original wiring/rule scheme, the similarity of the new pre-images, the order of learning, and the choice of re-wiring options. However, the network may have additional capacity to learn *distant* pre-images, further upstream in the transient tree as in figure 16. Note that if the network learns the given state itself as its own pre-image, this will result in a point attractor. If the state is learnt as a distant pre-image, this will result in a cyclic attractor with a period equal to the distance.

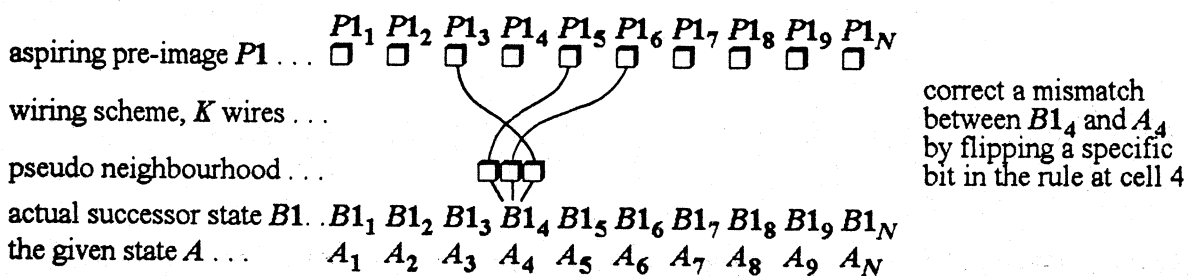
Learning by mutating the rule scheme

To correct a mismatch between a cell in A and the corresponding cell in the successor of an aspiring pre-image, P_1 , by mutating the rule scheme, one specific bit in that cell's rule table is flipped (changed from $0 \rightarrow 1$ or $1 \rightarrow 0$). There is only one option, certain to succeed. Adding another aspiring pre-image, by the same method cannot cause P_1 to be forgotten. This is because any mismatch between a particular cell in the successor state B_2 (of the *aspiring* pre-image, P_2) and A can not relate to the same rule table entry that was 'looked up' to determine P_1 's successor. Otherwise there would have been no mismatch. Any change to correct the mismatch must be to a different rule table entry; P_1 's successor cannot be affected. It turns out that there is no limit to the number of pre-images of a given state that can be learnt by this method, and no risk of forgetting previously learnt pre-images of the state, but of course there may be side effects

elsewhere in the basin of attraction field.

As an extreme example, all states in state space can be made the pre-images of any arbitrary state A . The two trivial rules with rule tables consisting of only 0s or only 1s are allocated to cells in A according to whether a cell equals 0 or 1. The result will be a basin of attraction field consisting of a single point attractor. All other states in state space will be *garden-of-Eden* pre-images of the point attractor.

Consider the state $P1$, $(P1_1, P1_2, \dots, P1_N)$, that the network is to *learn* as a pre-image of the given state A , (A_1, A_2, \dots, A_N) . When $P1$ (the aspiring pre-image) is evolved forward by one time-step according to its current K neighbour wiring/rule scheme its actual successor state is $B1$, $(B1_1, B1_2, \dots, B1_N)$.



Suppose that $B1_4 \neq A_4$. To correct the mismatch, the CA rule at cell 4 is mutated by flipping the bit in its rule table corresponding to the pseudo neighbourhood at $B1_4$. There is only one option. As the rule table has 2^K bits, flipping one bit represents a change of $1/2^K$ in the rule table, i.e. for 3-neighbour wiring $1/8$, for 5-neighbour wiring $1/32$. This is the probability of *forgetting* another transition elsewhere in the basin of attraction field, outside the immediate pre-image fan where the probability of forgetting = 0. For larger K the probability of a rule table bit flip resulting in forgetting some arbitrary transition, $X \rightarrow Y$ becomes smaller at an exponential rate as K increases. It was shown in³¹ that a 1 bit mutation (to each cell, i.e. flipping a total of N bits) in a 5-neighbour CA resulted generally in a small change in the basin of attraction field.

Consider an arbitrary transition $X \rightarrow Y$ elsewhere in the basin. The probability of *forgetting* the transition by one bit-flip is given by,

$$F_1 = 1/2^K$$

by two bit-flips,

$$F_2 = F_1 + F_1(1 - F_1)$$

by n bit-flips,

$$F_n = \sum_{t=0}^{n-1} \left(1 - \frac{1}{2^K}\right)^t \times \frac{1}{2^K}$$

If a set of pre-images to be learnt are close to each other in terms of Hamming distance, there will be fewer side effects elsewhere in the basin of attraction field. Again, close pre-images are likely to be learnt by default; and the network is able to generalise from examples.

Sculpting the basin of attraction field

Re-wiring has a much a greater effect on basin structure than mutating the rule scheme, but in either case the stability of basin structure is noteworthy. Using these methods, point attractors, cyclic attractors and transient sub-trees can be created. Transient sub-trees are sometimes transplanted along with the repositioned state (see figure 15), indicating how learnt behaviour can be re-applied in a new context. Generalisation is present, because bit patterns in the same pre-image fan are likely to be close in Hamming distance to each other, and so may be learnt by

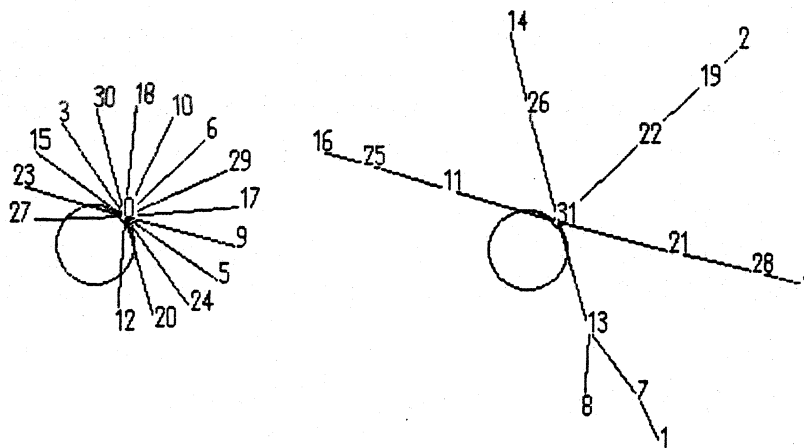


Figure 17. The basin of attraction field of a 5 cell parity sorting net-work. The first basin has only even parity states, the second has only odd parity states. Both basins have point attractors. Bit patterns are represented by decimal numbers.

default from examples. Forgetting involves *pruning* pre-images and transient sub-trees, and is achieved by the inverse of the method for learning. Since it is sufficient to create just one mismatch in order to forget, the side effects are minimal as compared with learning.

Figure 15 shows an arbitrary example (with no particular aim) of *visible* learning (and side effects) in a 6 cell network with regular wiring but randomly allocated rules. Figure 17 shows the basin of attraction field of a 5 cell network that has been taught to segregate 5-bit strings with odd and even parity into two separate basins.

Combining wiring and rule scheme adjustments, may result in a powerful method of cumulative learning (supervised and unsupervised) in random Boolean networks. In future work I anticipate applying the algorithms to simple recognition tasks, to use genetic algorithms to evolve improved start parameters of networks prior to learning, to extend the learning methods to include changes in system size and connectivity, and to investigate assemblies of weakly coupled semi-autonomous networks.

Conclusions

Whereas CA, embedded as they are in a continuous space with homogeneous laws, provide models of processes in physics, including the emergence of self organisation and simple life-like phenomena, random networks may be appropriate as models for the *non-local* interactions between living sub-systems typical of biology. Non-local interactions transcend inanimate physics and are perhaps a key property distinguishing living from non-living matter. Complex non-local networks with emergent properties occur hierarchically at all scales in biology, from networks of genes regulating one another within a cell¹¹, to networks regulating entire ecosystems, not to mention human society and technology.

Understanding the dynamics of abstract random networks may provide insights into biological networks in general, where concepts analogous to memory may be ubiquitous, take for instance memory in the immune system.

Non-local interactions, shaped by evolution and learning, allow the emergence of

unconstrained *categorisation* in a system's repertoire of dynamical behaviour - its content addressable memory. The basin of attraction diagrams of random Boolean networks capture the network's memory. The diagrams demonstrate that a complex hierarchy of categorisation exists within transient trees, far from equilibrium, providing a vastly richer substrate for memory than attractors alone. In the context of many semi-autonomous weakly coupled networks, the basin field/network relationship may provide a fruitful metaphor for the mind/brain.

Acknowledgements

I am grateful to Chris Langton, Stuart Kauffman, Steen Rasmussen, Mats Nordhal, Brossl Hasslacher, Crayton Walker, Stuart Hameroff, Joshua Smith, Tim Thomas, Ron Bartlett, Pedro de Oliveira, Inman Harvey, Harry Barrow, and Phil Husbands (especially for his expressions for the probability of forgetting), and many other friends and colleagues at the Santa Fe Institute, the University of Sussex and elsewhere, for discussions and comments.

References

1. Alexander, I., W. Thomas and P. Bowden, (1984) *WISARD, a radical new step forward in image recognition*, Sensor Review, 120-4.
2. Ashby, W.R., (1960) *Design for a Brain*, Chapman & Hall.
3. Conway, J.H., (1982) *What is Life?* in *Winning ways for your mathematical plays*, Berlekamp, E., J.H. Conway and R. Guy, Vol. 2, chap. 25, Academic Press, New York.
4. Grey C.M., W.P. König, A.K. Engel, W. Singer, (1989) *Oscillatory responses in cat visual cortex exhibit inter-columnar synchronization which reflects global stimulus properties*, Nature 388, 334-337.
5. Hameroff, S., S. Rasmussen and B. Mansson, (1989) *Molecular Automata in Microtubules: Basic Computational Logic of the Living State*. in *Artificial Life*, C. Langton (ed) Santa Fe Institute Studies in the Sciences of Complexity, Proceedings Vol. VI, Addison-Wesley 521-553.
6. Hameroff, S.R., J.E. Dayhoff, R. Lahoz-Beltra, S. Rasmussen, E.M. Insinna and D. Koruga (1993) *Nanoneurology and the Cytoskeleton: Quantum Signaling and Protein Conformational Dynamics as Cognitive Substrate*, in Behavioral Neurodynamics, K. Pribram and H. Szu, eds., Pergamon Press
7. Harvey, I., P. Husbands, D. Cliff, (1993) *Issues in Evolutionary Robotics*, in *From Animals to Animals 2*, Proceedings of the 2nd International Conf. on Simulation of Adaptive Behaviour, ed. Mayer, Roitblat and Wilson, MIT Press, 364-373.
8. Hopfield, J.J., (1982) *Neural networks and physical systems with emergent collective computational abilities*, Proceedings of the National Academy of Sciences 79:2554-2558.
9. Kauffman, S.A., (1969) *Metabolic stability and epigenesis in randomly constructed genetic nets*, Journal of Theoretical Biology, 22, 437-467.
10. Kauffman, S.A., (1984) *Emergent properties in random complex systems*, Physica D, vol 10D, 146-156.
11. Kauffman, S.A., (1991) *Requirements of evolvability in complex systems; Orderly dynamics and frozen components*, in Complexity, Entropy and the physics of information, ed. W.H. Zurek. Santa Fe Institute Studies in the Sciences of Complexity, Proceedings Vol. VIII, Addison-Wesley, 151-192.
12. Koestler, A., (1967) *The Ghost in the Machine*, Pelican Books
13. Langton, C.G., (1986) *Studying Artificial Life with Cellular Automata*, Physica D 22, 120-149.
14. Langton, C.G., (1990). *Computation at the edge of chaos: Phase transitions and emergent computation*, Physica D 12-37
15. Langton, C.G., (1991) *Life at the Edge of Chaos*, in *Artificial Life II* eds C. Langton et al, Santa Fe Institute Studies in the Sciences of Complexity, Proceedings Vol. X, Addison-Wesley
16. Li, W., (1991) *Phenomenology of Non-Local Cellular Automata*, Santa Fe Institute Working Paper 91-01-001.

17. Li, W., and M.G. Nordahl, (1992) *Transient Behaviour of Cellular Automata Rule 110*, Santa Fe Institute Working Paper 92-03-016.
18. McCormick, D.A., (1990) *Membrane properties and neurotransmitter actions*, in *The Synaptic Organization of the Brain*, Shepherd, G.M., ed 3rd edition, Oxford University Press, 32-66.
19. von Neuman, J., (1966) *Theory of Self Reproducing Automata*, edited and completed by A.W. Burks, University of Illinois Press
20. Shepherd, G.M., ed, (1990) *The Synaptic Organization of the Brain*, 3rd edition, Oxford University Press.
21. Singer, W., (1991) *Response synchronization of cortical neurons; An epiphenomenon or a solution to the binding problem?* IBRO News 19, 6-7.
22. Vichniac, G.Y. (1990) *Boolean derivatives on Cellular Automata*, Physica D 45, 63-74.
23. Walker, C.C., and W.R. Ashby, (1966) *On the temporal characteristics of behavior in certain complex systems*, Kybernetik 3, 100-108
24. Walker, C.C., (1971) *Behavior of a class of complex systems: the effect of system size on properties of terminal cycle*, Cybernetics, 55-67
25. Walker, C.C., (1987) *Stability of equilibrial states and limit cycles in sparsely connected, structurally complex Boolean nets*, Complex Systems 1, 1063-1086
26. Walker, C.C., (1990) *Attractor dominance patterns in random-structure cellular automata*, to be published in Physica D.
27. Wasserman, G.S., (1992) *Isomorphism, Task Dependence, and the Multiple Meaning Theory of Neural Coding*, Biological Signals.
28. Weisbuch, G., R. DeBoer and A.S. Perelson, (1990) *Localized memories in idiotypic networks*. J. Theoret. Biol. 146, 483-499.
29. Wolfram, S. (1983) *Statistical Mechanics of cellular automata*, Review of Modern Physics, vol 55, no 3, 601-64.
30. Wolfram, S., (1984) *Universality and complexity in cellular automata*, Physica 10D, 1-35.
31. Wuensche, A., and M.J. Lesser. (1992) *"The Global Dynamics of Cellular Automata; An Atlas of Basin of Attraction Fields of One-Dimensional Cellular Automata"*, (diskette included), Santa Fe Institute Studies in the Sciences of Complexity, Reference Vol. I, Addison-Wesley.
32. Wuensche, A. (1992) *The Ghost in the Machine; Basin of Attraction Fields of Disordered Cellular Automata Networks*, Santa Fe Institute Working Paper 92-04-017.
33. Wuensche, A. (1992) *Basins of Attraction in Disordered Networks*, in *Artificial Neural Networks*, 2, ed. I. Alexander and J. Taylor, Elsevier.
34. Wuensche, A. (1993) *Memory, Far from Equilibrium; Basins of Attraction of Random Boolean Networks*, Proceedings, European Conference on Artificial Life '93, Centre for non linear phenomena and complex systems, Université Libre de Bruxelles.
35. Wuensche, A. (1993) *Complexity in one-dimensional cellular automata; Gliders, basins of attraction and the Z parameter*, Cognitive Science Research Paper, University of Sussex.

All rights reserved:

All rights to the algorithms described in this paper are the property of the author.