Memoirs of the Faculty of Engineering, Okayama University, Vol.27, No. 2, pp.19-40, March 1993

Complex Dynamics and Search in A Cycle-Memory Neural Network

Shigetoshi NARA and Hiroo TOTSUJI

(Received January 29 , 1993)

SYNOPSIS

Numerical simulations of a single layer recurrent neural network model in which the synaptic connection matrix is formed by summing cyclic products of succesive patterns show that complex dynamics can occur with the reduction of a connectivity parameter which is the number of connection between neurons. The structure in these dynamics is discussed from the viewpoint of realizing complex function using complex dynamics.

Department of Electrical and Electronic Engineering

1 Introduction

Neural network models have been extensively investigated from the viewpoint of understanding mechanisms for parallel distributed information processing (see for example the collection of pioneering papers in [1] and [2]). Although some neural networks are known to show complex dynamics such as chaos [3], [4], only a small proportion of the works so far have been devoted to functional aspects of complex dynamics. In order to advance the argument that chaos could play significant functional roles in adaptive, self-organizing and evolving systems [5] $\sim$ [8], it is important to consider chaos dynamics in neural networks in a functional context. In this paper we consider chaotic dynamics in a neural network in the context of a memory search task. Our motivation for this is two-fold ; (1) as a way of characterizing dynamics, and (2) to get insights into the usefulness of a chaotic network for processing tasks.

We consider a discrete time synchronous recurrent network. The memory matrix is an asymmetric matix chosen so dynamics tend to converge to a cycle of memory patterns [10]. In this network, an important parameter is the connectivity of neurons. Mori *et al.* [10] showed that when fully connected the robustness of the memory cycles improves with cycle length. This is due to a type of dynamical "self-annealing" which reduces the effect of spurious attractors. If connectivity is reduced it is expected that cycles will become unstable and that complicated dynamical behaviour may appear. Such complicated dynamical behaviour bifurcating out of stable memory cycles with the change of the connectivity parameter may be useful if the structure in the dynamics is related in some meaningful way to the stored memory.

Parisi [12] proposed that temporal instability due to strong asymmetry could be useful in the learning process, with the metastability of memory states serving to distinguish them from other, "chaotic" states. It is known from information processing tasks such as optimization and learning, that stochastic dynamics can be useful in the right context [13] $\sim$ [17]. It is natural to speculate that the onset of deterministic chaotic dynamics may make it possible for a processing network system to intrinsically generate all the stochasticity needed, easily and efficiently. We focus on the possibility of inducing in the network with a simple adaptive change of a parameter, the appearance of useful chaos - useful in the sense of a search task.

Such search access could be useful in stochastic pattern recognition, adaptation and learning. It is known for example that stochasticity is useful in the learning stages in neural networks. We take the viewpoint that even if the network connections have been determined (either by learning process, or programming) in such a way that direct, non-stochastic access of memory states is possible for a specific sort of input code, if at some later stage the memory states need to be accessed using a completely new type of input code it may be necessary to do a type of search access, or sampling among multiple basins. In general, search access is necessary for accessing information in cases where effective direct relationships between input and attractor basins have not been established ; i.e. in cases where decoding or testing is possible, but coding or direct selection is not possible. In this context, one of the author (S. N.) and Peter Davis made numerical investigations and showed that the above mentioned function of search acess is actually realizable using complex dynamics introduced in a certain recurrent neural network model [11].

Using chaos for search has been mentioned elsewhere (for example, Tsuda *et al.* [7]). We might expect that (a) intrinsic complex dynamics from simple parameter adjustment means we don't need a separate complex sequence generation mechanism, and (b) non-uniform randomness of the complex dynamics, i.e. the balance between structure and randomness, may allow the possibility of more efficient search. We introduce a concrete method for using chaos for search. We emphasize the need for (1) an adaptive dynamical mechanism for selection of favoured states from among the chaotically generated states, and (2) a match between the internal dynamical structure and the externally specified search task.

2 Neural Network Model Indicating Associative Function and Complex Dynamics

First, let us start form a brief description of our model. There are N neurons in states represented by $S_i = \pm 1 (i = 1 \sim N)$. Each neuron is updated by the discrete time rule, $\mathbf{S}(t+1) = F(\mathbf{S}(t))$,

$$S_i(t+1) = \mathrm{sgn}(\sum_{j=1}^{j=N} W_{ij} S_j(t)), \tag{1}$$

where sgn(x) is the step threshold function taking value $+1(-1)$ for $x > 0 (x < 0)$. W_{ij} are the synaptic connection coefficients. In the synaptic connection matrix, $\mathbf{T}$, a number of cycles of random patterns ξ are embedded by taking

$$\mathsf{W} = \sum_{\mu=1}^{L} \sum_{\lambda=1}^{M} \vec{\xi}_{\mu}^{\lambda+1} \otimes \vec{\xi}_{\mu}^{\lambda}, \tag{2}$$

where $\otimes$ means to take direct product. L, M are the number of cycles and the period of each cycle, respectively, so that $\vec{\xi}_{\mu}^{M+1} = \vec{\xi}_{\mu}^{1}$. Fig.1 helps the understanding of our model architecture and the dynamical aspects of firing pattern updating.

$K = LM$ is the total number of stored patterns. In this mode, for $K/N \ll 1$, there are coexisting cycle attractors, each with a corresponding basin of attraction. If the output is sampled at the cycle period M, each of the memory patterns will look like a fixed point attractor. This model has the same "many-to-one" (i.e. mapping of a basin onto an attractor) dynamical structure as the autocorrelation type of memory model, and similarly can act as a classifier or error correcting memory. In network memories, the appearance of complex phase space structure, such as spurious attractor basins, can detract from the function of memory retrieval. It has been shown that the noise reduction performance in this limit cycle memory can be better than that of autocorrelation type model due to a dynamic self-averaging effect which reduces trapping in spurious attractors [10].

Other studies have used nonsymmetric networks from a different point of view, addressing the interesting issue of the possible roles and properties of pattern transitions, cycles and more complicated dynamical behaviour in neural networks. Shinomoto [18] used an assumption that each synapse must be either excitatory or inhibitory to form an asymmetric memory matrix for which the system either converges to one of the memorized patterns or to a homogeneous "don't know" state for too noisy inputs. Nonsymmetric matrices, in which a symmetric contribution causes convergence toward a stationary pattern and an asymmetric part causes transitions between patterns, have been used in models of temporal pattern sequence generation. There are a number of models which either use nonsymmetric synaptic interactions with temporal features such as dynamic synamptic strength [19], or time delayed transmission [20], [21], [22], or use thermal noise to induce transitions [23]. (See other models [24], [25], [26]).

Some previous works on recurrent asymmetric neural networks have investigated various aspects of complicated dynamical behavior. Some studies have considered the effect of a percentage of asymmetric connections on autocorrelation memory retrieval performance [27], [28]. It has been found that there is a critical threshold for stability of autocorrelation memory with respect to asymmetric dilution. Adding a weak random asymmetry to a symmetric network can have an effect similar to increasing the level of noise [29].

Sompolinsky *et al.* studied a continuous-time network with random W_{ij}. Using a self-consistent mean-field theory, exact in the limit of infinite N, they predicted a transition from a stationary phase to a chaotic phase occurring at a critical value of the gain parameter g contained in the threshold function which they used $\tanh gx$. For finite N, they commented that simulations show the appearance of limit cycles of increasing complexity in the intermediate parameter regime between the stationary phase and chaos.

One could consider several primary parameters of our system which would affect the stability of the cycles ; for instances, threshold levels, or time delayed effect in eqn.(1) or synaptic connectivity, or cycle strength coefficients in eqn.(2). In the present paper, we consider a synaptic connectivity parameter and investigate the effect of its reduction. The existence of neurons which inhibit the transmission of signal pulses in a neural network and thus reduce neuron connectivity should play an important role in the stability of neurodynamics. Although such dynamics have not yet been observed in a biological system, an observation of synapse-on-synapse [31], [32] suggests the existence of such dynamics.

For reduced connectivity the time development of neuron states is calculated as

$$S_i(t+1) = \mathrm{sgn}\Big(\sum_{j=\{i\pm R/2\}} W_{ij} S_j(t)\Big), \tag{3}$$

where $\{i \pm R/2\}$ means that the j-summation shoud be taken over a certain number R of synaptic connections, $R \leq N$. R is called the connectivity parameter.

If the ratio of number of stored patterns to the total number of neurons is small enough, the fully connected neural network has great stability of memory retrieval because of the strong collective "molecular" field acting on each neuron near stored memory states and the consequent existence of large basins of attraction in the high dimensional phase space of neuron states. A slight reduction of connectivity has no effect on the stability of attractors. However, when the connectivity is greatly reduced, the collective field becomes so weak that the memory patterns cease to be stable. In the auto-correlation case ($M = 1$) the pattern tends to fall into a spurious fixed sate. On the other hand, in numerical trials on a network with embedded cycles, when $R \ll N$ the trajectory of firing patterns wandered through the phase space. Fig. 2 shows the long time behaviour of the overlap $m \equiv \vec{S}(t) \bullet \vec{\xi}/N$ with a memory pattern $\vec{\xi}$ when the connectivity is reduced from full connectivity for a case in which the initial pattern is the memory pattern $\vec{\xi}$. One can see that itinerant trajectories can occur for small values of connectivity parameter and that these trajectories wander in a region of the phase space which typically has small overlap with the initial pattern. Short cycle spurious attractors seem to become more common again at very small connectivity, for example at $R = 2$. We haven't exhaustively sutudied the behaviour for all initial conditions and parameters. In this letter we concentrate on typical properties of wandering dynamics.

The dynamics of this system consists of itinerant orbits among 2^{400} patterns which are the vertices of hyper-cube in 400-dimensions. In order to describe an itinerant trajectory we can use a symboic dynamics method - partition the space up into disjoint cells and record the sequence of transitions among cells. The question is, what is an appropriate partition? For now, let us consider the partition defined by the basins of attraction of memory patterns at full connectivity. We call this the " memory basin partition". Letting $i = \lambda + (\mu - 1)M$, we label the i-th cell in the partition as C^{m_i} ($i = 0,\ 1,\ 2,\ \ldots,\ K = LM$), and for $i = 1, 2, \ldots, K$, define it by

$$C^{m_i} \equiv \{\vec{S} : \text{as } k \to \infty, F^{kM}(\vec{S}) \to \vec{\xi}^{\lambda}_{\mu} \text{ or the inverse pattern, denoted } \xi^{\lambda\,\dagger}_{\mu}\}, \tag{4}$$

where F is the one step mapping of network state $\vec{S}$ at full connectivity, eq.(2). To complete the partition we define cell C^{m_0} as the complement of the space, which includes the basins of so called "spurious attractors".

$$C^{m_0} \equiv \{\vec{S} \in C^{m_j}, \forall j \;\; 1 \leq j \leq K\} \quad \text{for} \quad i = 0 \tag{5}$$

Note the memory cells so defined are (1) disjoint (2) typically complex in shape, and (3) typically have macroscopic size when the memory patterns are a relatively small number of random patterns ($K/N \ll 1$). Clearly at reduced connectivity, these cells may no longer correspond to memory basins. However, we adopt the memory basin partition because our ultimate interest is to use the dynamics at small connectivity in conjunction with dynamics at full connectivity. For convenience we shall also make use of cycle cells, where a cycle cell is the union of the memory cells containing memory patterns in the same cycle.

How do the dynamics for reduced connectivity look in the memory partition? Let us present typical examples for $N = 400, L = 10, M = 3, K = 30$ and $R = 6$. Table 1 shows the initial stages of two neighbor orbits whose starting patterns differ by only one bit. The orbits are represented as sequences of the integer labels, i, assigned to the memory basins. The sequences are very different, indicating a sensitive dependence on initial condition. In each case, the sequence repeatedly visits all memory cells, but the visiting sequence is not periodic (or at least has very long period). We refer to this complex wandering phenomenon, with sensitive dependence on initial states, as chaos.

Fig. 3 shows the visiting ratio for cycle cells for the first case in the table. Comparing the visiting ratios over 1000 time steps (Fig. 3a) and 2000 time steps(Fig. 3b), it can be seen that the resulting distribution is roughly identical, which indicates the distribution has converged. Fig. 3c shows the completely different distribution obtained for the second orbit with initial pattern differing by only one bit. Different distributions are obtained for different starting patterns, showing the co-existence of different types of asymptotic orbits.

Fig. 4 shows the distribution on the memory partition that would be expected if the dynamics at reduced connectivity were completely random in the space (ie. a mapping which independently inverts N bits S_j each with probability 1/2). This is just a projection of a homogeneous measure onto the memory basins at full connectivity and shows the relative size of these basins.

Clearly, there is considerable structure in the chaotic orbits. With respect to basin distribution, each of the chaos examples examined differed from the random case. The degree of similarity with the random case was found to depend on the connectivity parameter.

3 Memory Search as A Complex Information Processing Function

One can now consider these dynamics from the point of view of sampling the phase space. Let us consider using the chaotic dynamics to do a search task. Since the itinerant dynamics, at small connectivity, visit states which correspond, at large connectivity, to all the basins of recorded memory patterns, all memory patterns can be generated (all-be-it in a random sequence) by just modulating the connectivity. If we have a test criterion for the memory patterns generated, we can use this method to search for memory patterns satisfying the criterion.

The test criterion we shall use is whether or not a pattern has a certain "feature". By way of example we think of the N neuron elements as pixels in a p by (N/p) two dimensional image, and define the feature as a measure of the similarity of two vertical stripes separated by q pixels. In this case the feature function is

$$f(\vec{S}) = \cos\theta_{p,q} = \frac{p}{N}\sum_{i=1}^{[N/p]} S_{ip}S_{ip+q} \tag{6}$$

Here $[x]$ means the largest integer not greater than x. It is easy to understand that the definition of eq.(6) gives an inner product of two stripe vetors as shown in Fig.5, for instance, by taking $p = 20$ and $q = 10$.

A feature partition can be specified by specifying the set of feature values which can be taken by $f(\vec{S})$,

$$f_i = \frac{2}{p}(i-1) - 1 \quad \text{for} \quad i = 0,\ 1,\ 2,\ ...,\ J \tag{7}$$

where $J = p + 1$. (eq. $f = -1,\ -.9,\ -.8,\ ...,\ .8,\ .9,\ 1$ for $p = 20$ case). The i-th cell in a feature partition C^{f_i} $(i = 1,\ 2,\ 3,\ ...,\ Q)$ is defined by

$$C^{f_i} = \{\ \vec{S}\ :\ f(\vec{S}) = f_i\ \} \text{ for } i = 1,\ 2,\ 3,\ ...,\ Q \tag{8}$$

This second partition is unrelated to the dynamics at full connectivity. For a random sequence of patterns, the distribution on the feature partition is just a binomial distribution centered on feature value zero. In the case of chaotic dynamics at reduced connectivity, it was found that chaos at intermediate connectivities typically resembles that for the random generation of patterns. For other values of connectivity, the distribution can be significantly different, for example, displaced and narrower.

As an example of search, we consider "feature-first" search in which feature value is tested first and memory membership tested second. Such feature-first search is particularly meaningful in a memory system when the number of memory patterns is very high. We have investigated the search procedure in which patterns are generated at reduced connectivity, $R = R_b$ $(0 < R_b < N)$, until one is found with a specified feature value, and then full connectivity, $R = N$, is restored for a number, z, of steps to test if that pattern is in a basin of cycle containing a memory pattern with the target feature value.

$$R(t+1) = N - (N - R_b) \prod_{k=t-z}^{t} E(k) \tag{9}$$

The error variable takes on one of two values, $E = 1$ when the pattern has the target feature value and $E = 0$ otherwise. This is a type of "two-level adaptive bifurcation" [33], [34]. The effect of the product on the right-hand side of Eqn.9 is that connectivity R is set at full connectivity, $R = N$, so long as any of the patterns generated in the last z steps had the target feature. Taking $z > M$ results in trapping in a cycle if one of the patterns in the cycles has the target feature. We include a random reset of pattern if search is not successful before a certain time limit. A corresponding random search is defined as a search in which the reduced connectivity $R = R_b$ steps are replaced by completely random choices of pattern.

Fig. 6 shows an example of average search times as a function of R_b for a particular target feature value in a case where $N = 400, M = 6$ and $L = 5$. The search times are typically smaller for intermediate values of connectivity, than for extremes of large or small connectivity. Moreover, the shortest search times are comparable if not smaller than the value for random search, which is indicated by the dotted line. The measured search time for random search case is 52. There are 19 values of small R for which the chaos search is better than the random search. $R = 46$ is best.

Let p_1 be the measure of the dynamical sequence at reduced connectivity $R = R_b$ on the target feature ($E = 0$) partition cells, and p_2 the proportion of this measure in the target memory cell. For search where the probability p of success is the same each trial, the average search time τ is given by $\tau = (1-p)/p$. In the random case, the one-step success probability p is just $p = p_2 \cdot p_1$. Using the distributions on the partitions to obtain the values of p_1 and p_2, the expected average search time for random search was found to be about 50, which is consistent with the average value of 52 obtained in the search trials.

Let us assume that in the chaos case, as in the random case the one-step success probability p is estimated as $p = p_2 \cdot p_1$. Then we can say the following. On the one hand, search at small connectivity, such as $R = 10$, is slower than random search because both p_1 and p_2 are smaller than in the random case. On the other hand, search at intermediate values of connectivity, such as $R = 46$, compares favourably with the random search because, for typical orbits, p_1 is comparable with that in the random case, and p_2 is slightly larger than that in the random case. In other words, the suitability of the intermediate connectivity values for the search task is due to the fact that on the one hand there is sufficient randomness with respect to the arbitrarily specified feature partition so p_1 is comparable with random generation, while on the other hand there is remnant dynamical structure associated with the memory partition, giving high p_2.

This consideration demonstrates in a concrete way how the effectiveness of the search depends on the match of the chaotic dynamics with the search task. This highlights an issue which is central to the question of the possible role of chaos in adaptive systems [33], [34]. It is clear that given a specific problem, in general there can be some chaos which will be better than random search, while there is other chaos which will not be. This is similar to the problem of the efficiency of genetic algorithms in stochastic search tasks [35]. However, the above example shows that when there is bifurcation to chaos from stable attractors in a neural network, there can be a regime, easily accessed by moving a single parameter, where the chaos typically has a balance of randomness and meaningful structure which at least makes it comparable with random search for search based on arbitrarily specified features. How such matching can be improved by learning algorithms remains to be investigated.

4 Concluding Remarks

In this paper we considered whether in the cycle-memory neural network model there appear chaotic dynamics which enable us to realize a complex information function. The important queations are "Does chaos appear?" and if so "Are the chaotic dynamics appropriate for an application to a complex function realization?"

As shown in this paper, the complex dynamics can be easily induced in a neural network model in which a certain number of limit cycles are embedded with the use of a connection matrix formed by summing direct products of successive pattterns in cyclic sequences. Chaos-like wandering or dynamics are induced by reducing the synaptic connectivity. The structure in the chaos dynamics suggest such dynamics could be potentially useful for search in pattern space. We have shown it is possible to do search among all memories with just modulation of just a single parameter. It was found that there were chaotic dynamics which appeared random in some partitions but had remnants of the programmed or learnt structure in other partitions, and we showed that this determined the match of the internal dynamics with the search task and thus the effectiveness of search with chaos. The challenge remains to find ways (for example learning algorithms), in which a system can adapt the internal dynamical structure, for example via a learning mechanism, to improve the match between internal structure and search tasks.

References

[1] Neurocomputing, edited by J. A. Anderson and E. Rosenfeld, The MIT Press

[2] Neurocomputing 2, edited by J. A. Anderson, A. P. Pellionisz and E. R. Rosenfeld, The MIT Press (1991)

[3] K. Aihara, T. Takabe and M. Toyoda : Phys. Lett. vol. 144 (1990) 333

[4] T. Fukai and M. Shiino : Phys. Rev. Lett. vol. 64 (1990) 1465

[5] C. A. Skarda and W. J. Freeman : Brain and Behavioral Science, vol. 10 (1987) 161 ; Y. Yao and W. J. Freeman : Neural Networks vol. 3 (1990) 153

[6] P. Davis : Jpn. Jour. Appl. Phys., vol. 29 (1990) L1238

[7] I. Tsuda, E. Koerner and H. Shimizu : Prog. Theor. Phys. vol. 78 (1987) 51

[8] N. Packard : "Dynamic Patterns in Complex Systems" p.293, ed. J. A. Kelso, A. J. Mandell and M. F. Shlesinger, World Scientific 1988

[9] K. Kaneko : Physica D, vol.41, 137 (1990)

[10] Y. Mori, P. Davis and S. Nara : Journal of Physics A, vol. 22 (1989) L525

[11] S. Nara and P. Davis : Prog. Theor. Phys. vol. 88 (1992) 845

[12] G. Parisi : J. Phys. A vol. 19 (1986) L675

[13] N. Metropolis, A. W. Rosenbluth, A. H. Teller and E. Teller : J. Chem. Phys. vol. 6 (1953) 1087

[14] S. Kirkpatrick, C. D. Gellat and M. P. Vecci : Science vol. 220 (1983) 671

[15] D. H. Ackley, G. E. Hinton and T. J. Sejnowski : Cognitive Science, vol. 9 (1985) 147

[16] A. G. Barto : *Vision, Brain and Cooperative Computation* p. 665, ed. M. A. Arbib and A. R. Hanson, MIT Press Cambridge 1987

[17] M. Lewenstein and A. Nowak : Phys. Rev. Lett. vol. 62 (1989) 225

[18] S. Shinomoto : Biol. Cybern. vol. 57 (1987) 197

[19] P. Peretto and J. J. Niez : "Disordered Systems and Biological Organization", ed. E. Bienenstock, F. Fogelman-Soulié and G. Weisbuch, Springer (1986)

[20] D. Kleinfeld : Proc. Natl. Acad. Sci. (USA) vol. 83 (1986) 9469

[21] H. Sompolinsky and I. Kanter : Phys. Rev. Lett., vol. 57 (1986) 2861

[22] H. Gutfreund and M. Mézard : Phys. Rev. Lett. vol. 61 (1988) 235

[23] J. Buhmann and K. Schulten : Europhys. Lett. vol. 4 (1987) 1205

[24] L. Personnaz, I. Guyon and G. Dreyfus : Phys. Rev. A vol. 34 (1986) 4217

[25] S. Dehaene, J. P. Changeux and J. P. Nadal : Proc. Nat. Acad. Sci. (USA) vol. 84 (1987) 2727

[26] I. Guyon, L. Personnaz, J. P. Nadal and G. Dreyfus : Phys. Rev. A vol. 38 (1988) 6365 ; L. Personnaz, I. Guyon and G. Dreyfus : Ph

[27] J. J. Hopfield : Proc. Nat. Acad. Sci. (USA) vol.79 (1982) 2554

[28] D. J. Amit : Proc. of Int. Symp. on the Physics of Structure Formation, T"ubingen, Berlin, Berlin 1986

[29] M. V. Feigelmann and L. B. Ioffe : Int. J. Mod. Phys. vol. B1 (1987) 51

[30] H. Sompolinsky, A. Crisanti and H. J. Sommers: Phys. Rev. Lett. vol. 61 (1988) 259

[31] B. Alberts, D. Bray, J. Lewis, M. Raff, K. Robert and J. D. Waton : *Molecular biology of the cell*, Garland Publishing Co. New York 1983

[32] E. R. Kandel : "Principles of Neural Science", second edition, ed. E. R. Kandel and J. H. Schwartz, Elsevier 1985

[33] P. Davis and S. Nara : Tec. Dig. of Int. Conf. on Fuzzy Logic and Neural Networks, Iizuka, (1990)

[34] P. Davis and S. Nara : Proceedings of The First Symposium on Nonlinear Theory and Its Applications, p. 97 (1990)

[35] D. B. Fogel and J. W. Atmar : Biological Cybernetics vol. 63 (1990) 111 ys. Rev. A vol. 34 (1986) 4217

(a) initial condition 1		(b) initial condition 2	
time step t	memory basin number	time step t	memory basin number
295	2	300	8†
302	9†	321	9†
305	20†	326	9†
306	9†	328	17†
307	4†	336	8†
317	29	338	9†
319	2	339	28
327	4†	340	8†
339	4†	342	9†
342	13†	343	4†
343	2	351	4†
344	29	358	8†
345	8†	360	8†
347	13†	365	29
357	14	368	8†
363	4†	379	2
374	12†	383	8
378	9†	384	8†
379	2	388	8†
392	29	395	9†
395	26†	415	2
397	29	429	9†
399	4†	432	8†
411	24†	434	30
414	13†	436	8†
415	2	438	12†

Table 1: Symbolic dynamic representation of orbits for two initial patterns differing by just one bit when the range is $R = 6$ in a case with $N = 400$, $M = 3$, $L = 10$. Location in pattern space is indicated by symbol i if $S \in C^{m_i}$. The steps in the spurious cell, $i = 31$, are not shown here. The symbol † means that the firing pattern is an exact inversion of a memory pattern. Initial state is memory pattern $i = 1$.

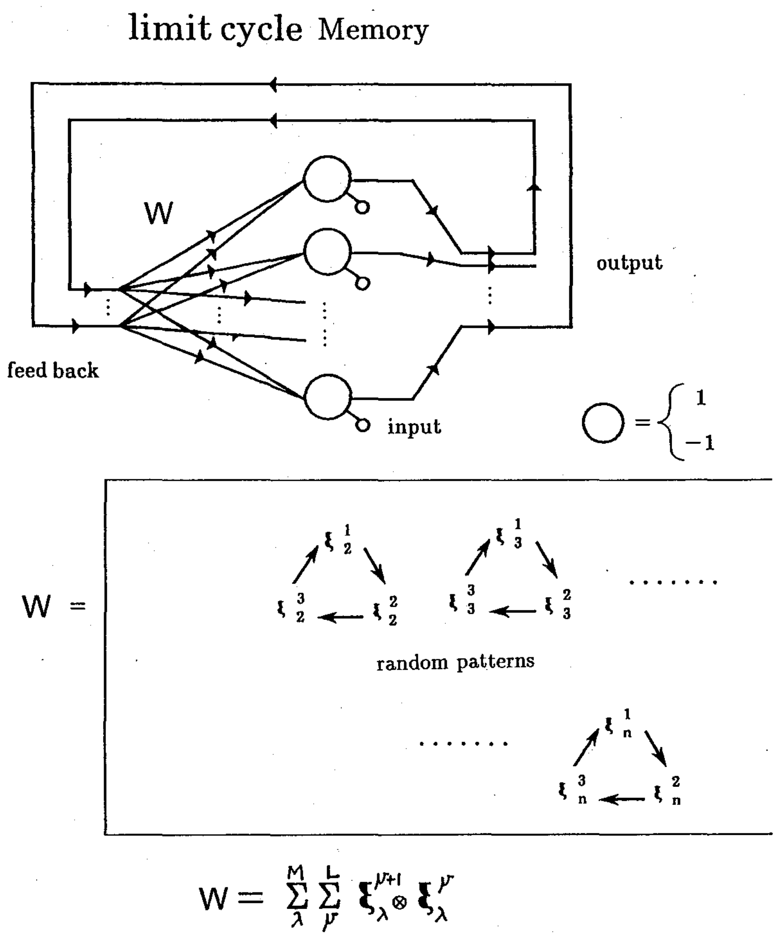


Figure 1: Schematic representation of our model architechture and the embedded pattern as limit cycles

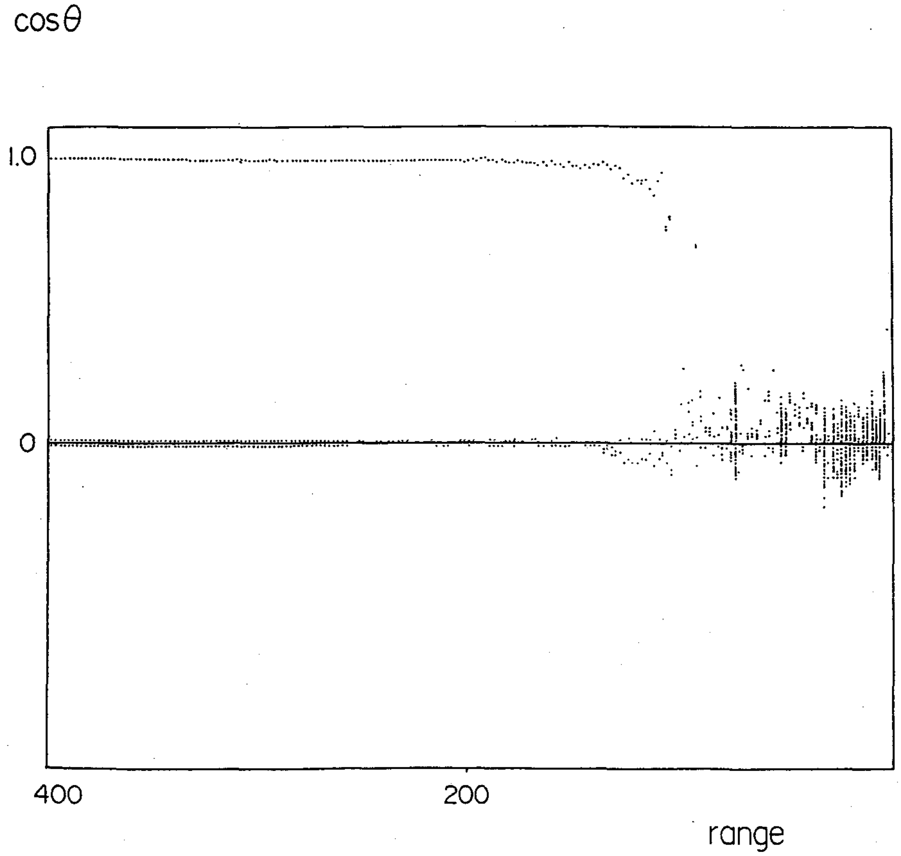


Figure 2: Itinerant orbits for reduced connectivities. Orbit is expressed in terms of directional cosine with initial memory pattern, i.e. $\cos\theta = \vec{S}(t) \bullet \xi / N$.

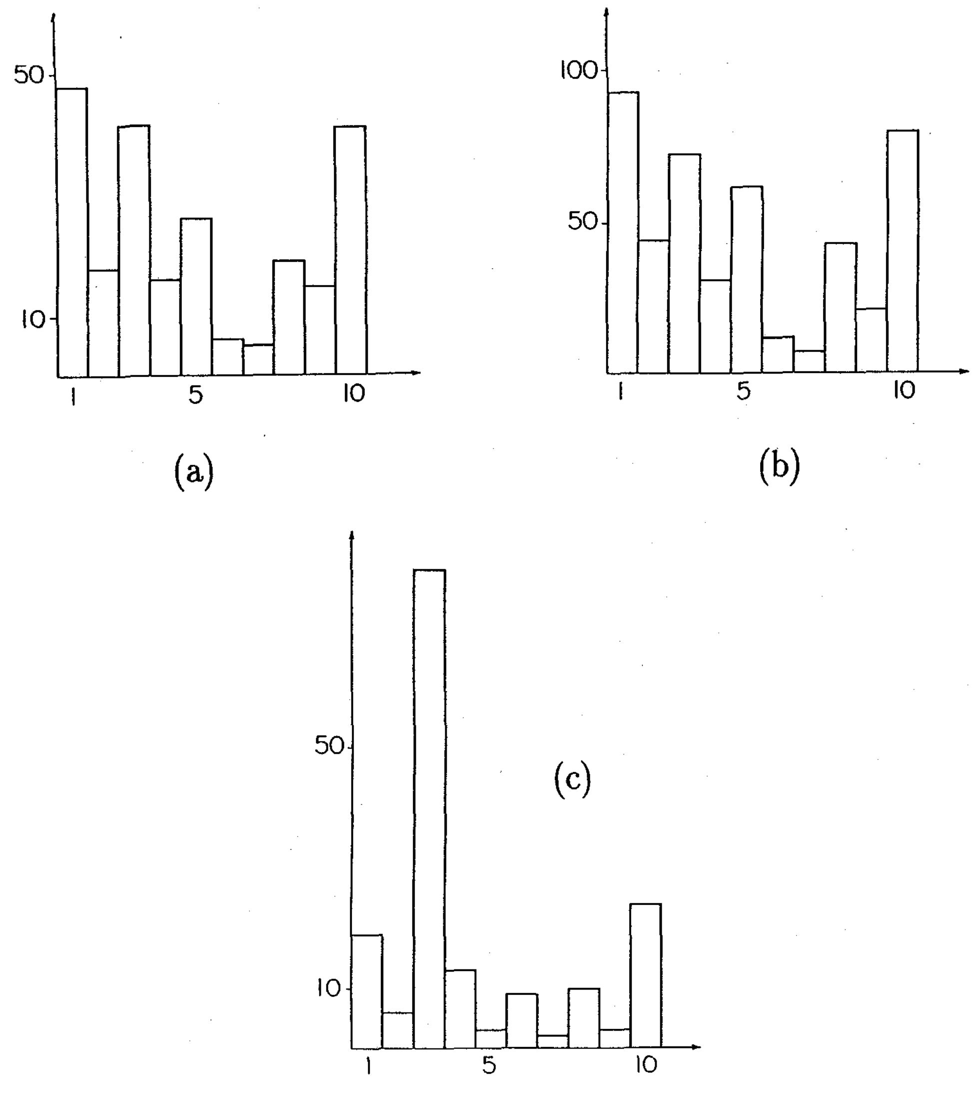


Figure 3: Memory partition distribution for orbits at connectivity $R = 6$. (a) 1000 steps of orbit (a) in the table. (b) 2000 steps of orbit (a) in the table. (c) 1000 steps of orbit (b) in the table.

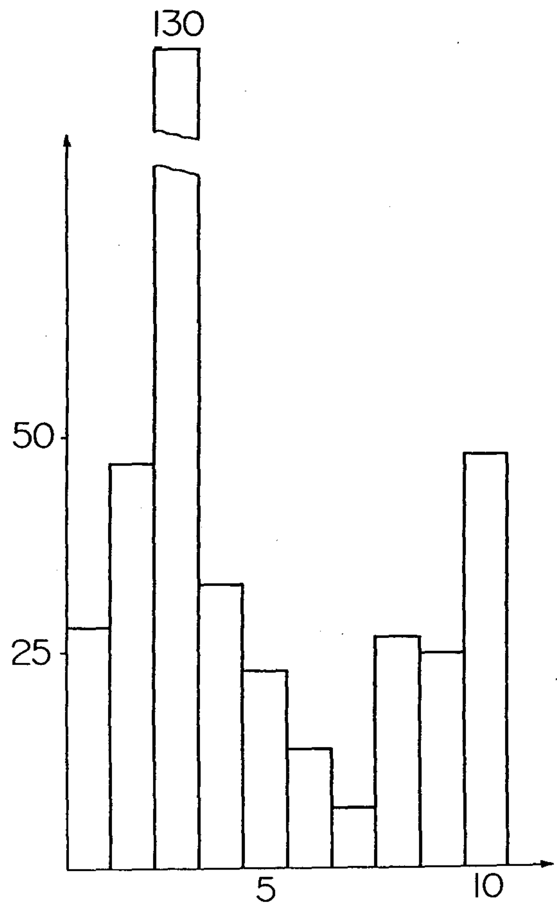
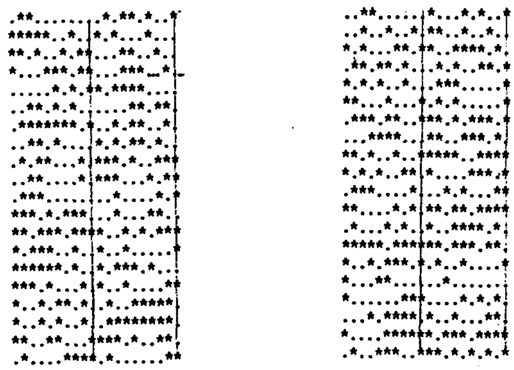


Figure 4: Memory partition distribution for random generation of patterns.



No.11 No.19

Figure 5: Twomemory patterns, the 11-th and 19-th, which have the feature value -0.5 where the feature is defind as an inner product of two tripe vectors shown in the figure as solid line, a center and a edge stipe.

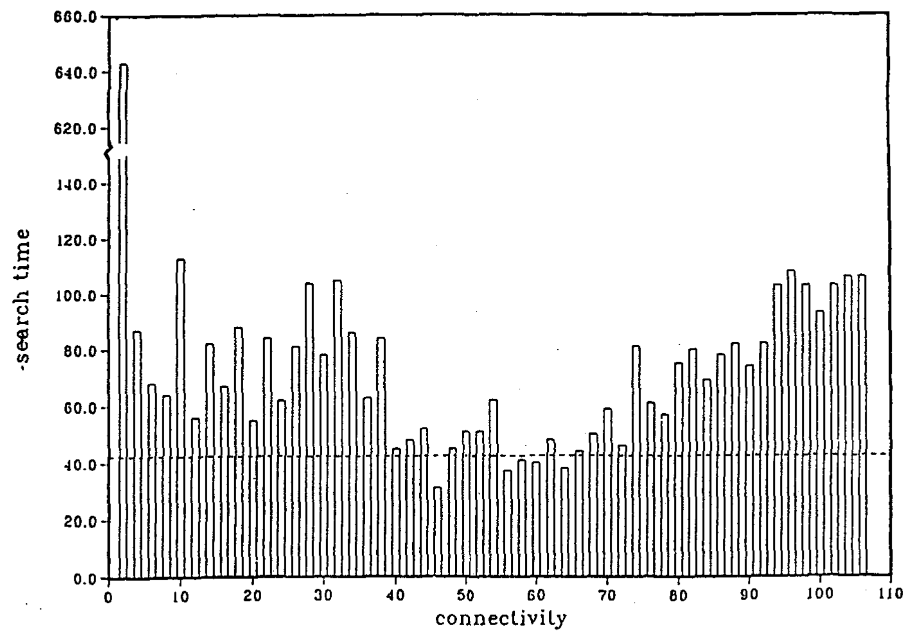


Figure 6: Dependence of search time on connectivity parameter R_b. $N = 400$, $M = 6$, $L = 5$. Feature target value is -0.5 ± 0.1. Memory pattern numbers 11 and 19 have the target feature value. Average over 300 initial conditions at each parameter value. The cutoff number for steps at full connectivity before random reset occurs is 120. Changing the cutoff number doesn't change the qualitative conclusion. The dashed line shows the search time for the random search case.