Adaptive **Behavior**

Structure and Dynamics of Random Recurrent Neural Networks

Hugues Berry[1], Mathias Quoy[2]
[1]*INRIA Futurs – 4 rue J. Monod 91893 Orsay, France, hugues.berry@inria.fr*
[2]*ETIS UMR8051 – IUT de Cergy-Pontoise – ENSEA, 6, av. du ponceau, 95014 Cergy-Pontoise, France, mathias.quoy@ensea.fr*

Contrary to Hopfield-like networks, random recurrent neural networks (RRNN), where the couplings are random, exhibit complex dynamics (limit cycles, chaos). It is possible to store information in these networks through Hebbian learning. Eventually, learning "destroys" the dynamics and leads to a fixed point attractor. We investigate here the structural changes occurring in the network through learning. We show that a simple Hebbian learning rule organizes synaptic weight redistribution on the network from an initial homogeneous and random distribution to a heterogeneous one, where strong synaptic weights preferentially assemble in triangles. Hence learning organizes the network of the large synaptic weights as a "small-world" one.

Keywords associative memory · complex networks · Hebbian learning · chaotic neural networks

1 Introduction

Using random recurrent neural networks is part of the dynamical systems approach to the simulation of some cognitive functions. The start of our work is the understanding of Freeman's experiments on the olfactory bulb, where the spontaneous neuron dynamics are chaotic and the dynamics reduces to a simpler attractor (a limit cycle) when a known odor is recognized (Freeman, 1987; Skarda & Freeman, 1987; Yao & Freeman, 1990). We have been able to replicate these findings using random recurrent neural networks (RRNN) with a classical Hebbian learning rule (Daucé, Quoy, Cessac, Doyon, & Samuelides, 1998). Following Amari (1972) we developed a mean-field theory that allowed us to study the dynamics of the system theoretically and understand the influence of its various parameters (Cessac, Doyon, Quoy, & Samuelides, 1994). Using a modified Hebbian learning rule, we have shown that this kind of network is able to store and retrieve complex spatial (static) and temporal (sequences) patterns (Daucé, Quoy, & Doyon, 2002).

Parallel to this approach, recent studies have focused on the topological structure of large networks using complex network approaches. They have proven successful in understanding the global properties of several complex systems originating from highly disparate fields, from biological to social and technological domains. Hence the same (or similar) reasoning can be applied to understand cell metabolism (Jeong, Tombor, Albert, Oltvai, & Barabasi, 2000), the citation of scientific articles (Newman, 2001), software architecture (Valverde, Cancho, & Solé, 2002), the Internet (Barabasi & Albert, 1999) or electronic circuits (Cancho, Janssen, & Solé,

Correspondence to: Hugues Berry NRIA Futurs – 4 rue J. Monod 91893 Orsay, France, hugues.berry@inria.fr
Tel.: + 33 1 30 73 62 92; *Fax.*: + 33 1 30 73 66 27.

(2006), Vol 14(2): 00–00.
[1059–7123(200612) 14:2; 00–00; 065742]

2001). The most common statistical structures are the so-called *small-world* and *scale-free* networks. Small-world networks are sparse complex networks with both small average shortest path and a large degree of clustering, while scale-free networks are defined by a connectivity probability distribution that decreases as a power law (see section 5 for more formal definitions). At a much coarser grain, complex networks methods have recently been applied to networks of cortical areas (Eguiluz, Chialvo, Cecchi, & Apkarian, 2005; Sporns, Chialvo, Kaiser, & Hilgetag, 2004) i.e. not networks of neurons but networks of neuron *areas*, with the prospect of understanding the network functions.

We propose here to combine a dynamical system approach with a complex network analysis, in order to understand the dynamical and structural changes occurring through learning in our model. So, in the following, we first present our model (Section 2), then show some typical dynamical behaviors (Section 3), before introducing the learning rule (Section 4) and studying the structure of the network before and after learning (Section 5). Finally we conclude in Section 6.

2 Model

A random recurrent neural network is a set of N fully connected neurons. The connection weights w_{ij} are initially randomly drawn according to a Gaussian law $\mathcal{N}(0, J^2/N)$, where J is the standard deviation. The state of the neurons in the network is $\mathbf{x}(t) = \{x_i(t)\}$, $i = 1 \ldots N$ where each x_i is a real number $\in [0–1]$ that is proportional to the firing frequency of neuron i. The state dynamics is given by the following set of discrete-time recurrent equations:

$$\forall t \geq 0, \qquad \forall i = 1 \ldots N,$$

$$x_i(t+1) = f\left(\sum_{j=1}^{N} w_{ij} x_j(t) + I_i - \theta_i\right) \tag{1}$$

where f is a sigmoidal function with slope g in 0. The thresholds θ_i are randomly chosen according to a Gaussian law $\mathcal{N}(\bar{\theta}, \sigma_\theta^2)$. $\mathbf{I} = \{I_i\}$ is an N dimensional constant input vector with mean 0 and variance 0.1. Time varying sequences have been studied in Daucé et al. (2002). Hence the parameters of the system are: g, J, $\bar{\theta}$, σ_θ and $\mathbf{I}$. Without loss of generality, in this study we have set J to 1, and $\bar{\theta}$ and σ_θ to 0.

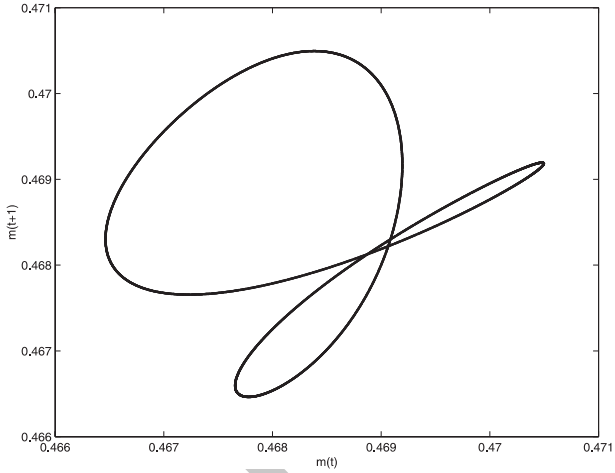

Figure 1 Limit cycle ($g = 6$, $N = 100$) represented in phase space ($m_{net}(t+1)$ versus $m_{net}(t)$). See text for explanations on m_{net}.

3 Dynamics

The network dynamics have been thoroughly studied in our previous work (Cessac et al., 1994; Daucé et al., 1998; Daucé et al., 2002). We report here on the main results of these studies. Depending on the parameter g (J being set to 1), various dynamical behaviors may arise. First, when g is low, the system exhibits a unique fixed point. When g increases, bifurcations occur leading successively to a limit cycle (see Figure 1), a torus (see Figure 2), frequency locking and finally to chaos

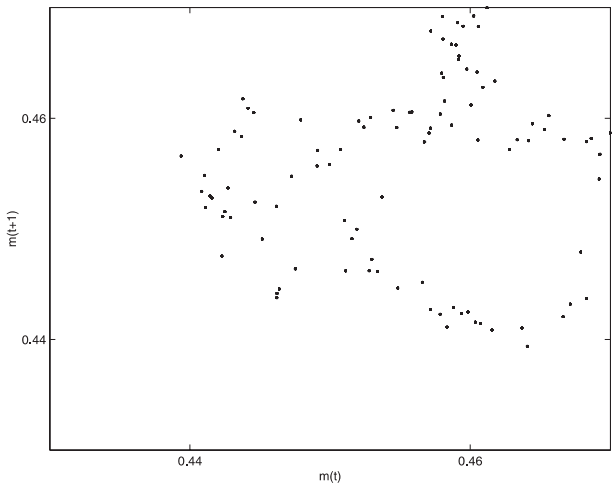

Figure 2 T2 Torus ($g = 6.85$, $N = 100$). See Figure 1 for explanations.

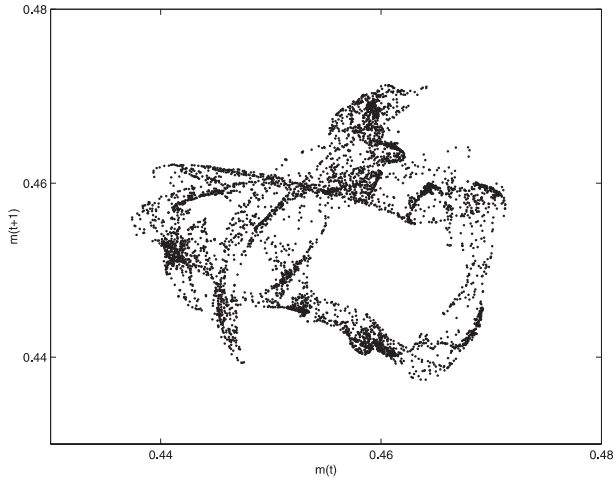

Figure 3 Chaos ($g = 7$, $N = 100$). See Figure 1 for explanations.

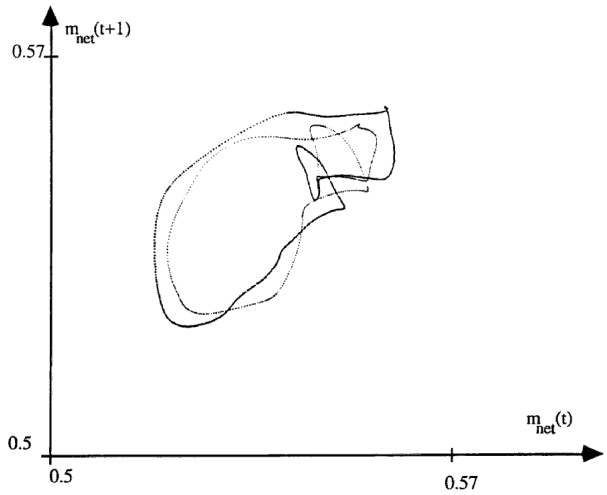

Figure 4 The attractor linked with the same input $\mathbf{I}^1$ after learning of $\mathbf{I}^1$ alone (light line), and after learning of $\mathbf{I}^1$ followed by learning of a different input $\mathbf{I}^2$ (darker line). The attractor is slightly different because the weights have changed through learning. However, the same group of neurons are active (Daucé et al., 1998).

(see Figure 3). This corresponds to a quasi-periodicity route to chaos (Doyon, Cessac, Quoy, & Samuelides, 1993). Observation of the network activity is performed through the mean activity of all neurons:

$$m_{net}(t) = \frac{1}{N}\sum_{i=1}^{N} x_i(t) \qquad (2)$$

Note that not all neurons behave the same way. Some neurons are either damped near $x_i = 0$, or saturated near $x_i = 1$. Some others really oscillate in the [0–1] range. In the following, the saturated and oscillating neurons will be referred to as the *active neurons or active population*. This set of active neurons is precisely the part of the network that sustains the dynamics. They are sequentially activated, so that there is a small cluster of neurons activated at each time step. This set of neurons is specific to the applied input pattern **I** (Daucé et al., 1998).

4 Learning Rule

In this study, learning is performed using the following Hebbian rule:

$$\begin{aligned} &\forall t \geq 1, \\ &\text{if} \quad x_j(t) \geq 0.5 \quad \text{then} \\ &w_{ij}(t+1) = w_{ij}(t) + a.x_i(t+1).x_j(t) \end{aligned} \qquad (3)$$

In addition a weight may not change its sign (it remains positive or negative). It is easily seen from this learning rule that only weights between active neurons will change. More complicated learning rules may be used as well (Daucé et al., 1998), but Equation (3) is the simplest and generic enough for the study we want to perform.

Learning an input modifies the weights, and therefore the dynamics of the system. Hence, during learning, we observe an inverse quasi-periodicity route to chaos, i.e. from chaos to a limit cycle to a fixed point. If learning is stopped when the dynamics settles on the limit cycle, this cycle happens to be specific of the learned input. Hence, the dynamics in the absence of input is chaotic, and presenting the input reduces the dynamics on the corresponding limit cycle. This behavior is similar to the experimental results observed by Freeman in the olfactory bulb (Freeman, 1987).

Each input vector leads to a different attractor because an input is similar to a threshold, and thus may be seen as a *parameter* of the system. Upon learning of another input, the weights are again modified. Thus the attractor corresponding to the first input is modified. However, it stays similar in terms of frequency, rotation number or center (see Figure 4). More importantly, the group of activated neurons remains almost unchanged

(Daucé et al., 1998). Hence, applying a learning rule relating these active neurons to a decision neuron on another layer will lead to the activation of the same decision neuron. We used this property to design the motor command of a mobile robot (Daucé et al., 2002).

Before learning, applying an input leads the system to a chaotic dynamics. Only learning reduces the dynamics onto a much simpler attractor. After learning and removing of the learned input, the dynamics of the system is different from the initial one (because the weights have changed) but chaotic again. Learning a different input leads the system to a different state. So the state reached after learning is input-specific. When learning different inputs successively, each input is linked with a different attractor. An input which is not learned does not change the dynamics in the same way as learned inputs (Daucé et al., 1998).

Note that unlike Hopfield networks, RRNNs possess a unique attractor at a given time step, so that for each set of parameters, any initial condition converges on the *unique* attractor. Theoretical and numerical studies reported in Cessac et al. (1994) confirmed that in the thermodynamic limit, the attractor is unique and showed that in this case, only fixed point or chaotic attractors are possible. For finite size networks, the number of attractors present at a given time usually remains unique but the types of attractors accessible to the dynamics increases to also include limit cycles and tori. Furthermore, in finite size systems, there exists a narrow parameter range (theoretically proven and numerically evidenced) where one fixed point and one strange attractor can coexist. We have also conducted experiments in which the weights are progressively symmetrized, and found that other attractors may appear (in the perfectly symmetrical case, RRNNs are Hopfield-like neural networks). Because inputs are parameters, there are different attractors for different inputs. But in our case, the input is given by the applied I_i's, *not* by the initial condition $\mathbf{x}(0)$ (which is randomly chosen). Thus our system is different from many chaotic neural networks for which chaos stems from the exploration dynamics of several attractors belonging to the *same* dynamical system (e.g. chaotic itinerancy; Tsuda, 2001).

5 Learning and Structure

The mean value and variance of the weights are changing because the weights are updated according to the learning rule (which remains unchanged). So after learning, the weights are not following the initial Gaussian distribution anymore (see below) and therefore, we have not yet been able to theoretically describe why the dynamics is reducing when learning takes place. However, simulations and statistics on the weight matrix give us some insights. We have studied the structure of the network before and after learning with complex networks tools (Barabasi & Oltvai, 2004). The results are reported in the following. Our aim here is to quantify how learning influences the way large synaptic weights distribute on the neural network. Towards this aim, we first proceed to the thresholding of the weight matrix. Let $\mathcal{W} = \{w_{ij}\}$ be the weight matrix and $\mathcal{A} = \{a_{ij}\}$ its corresponding (possibly thresholded) adjency matrix, i.e.

$$a_{ij} = \Theta(w_{ij} - \epsilon) \tag{4}$$

where ϵ is the threshold and $\Theta(\ldots)$ the Heavyside step function:

$$\Theta(x) = \begin{cases} 1 & \text{if } x \geq 0 \\ 0 & \text{if } x < 0 \end{cases} \tag{5}$$

Note that we are interested here in the strength of the connection between two neurons, regardless of their inhibitory/excitatory nature. Thus we restrain our analysis to the weight intensity, retaining only their absolute values, i.e. $w_{ij} \leftarrow |w_{ij}|$.

Because we are dealing with undirected graphs, incoming and outgoing connections are distinguished. Thus, we should ideally study each statistical indicator in triplicate: one concerning outgoing links only, one for incoming links only, and one dealing with the total links (incoming + outgoing). In this case, dealing with multiple statistical indicators rapidly becomes difficult to handle. We thus chose a tradeoff solution, considering the graph as undirected most of the time, while taking directionality into account in the clustering index.

We define the global weight $w_{i,j}$ as the maximal value of the incoming and outgoing weights:

$$w_{i,j} = \max(w_{ij}, w_{ji}) \tag{6}$$

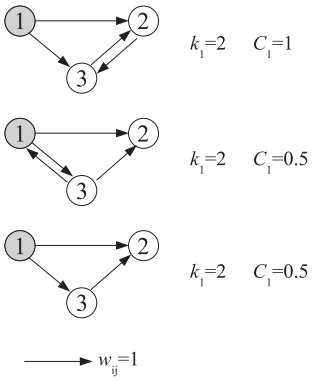

Figure 5 Illustration of the clustering index for three network examples. The value of the degree k_i and clustering index C_i for neuron number 1 are given in each case.

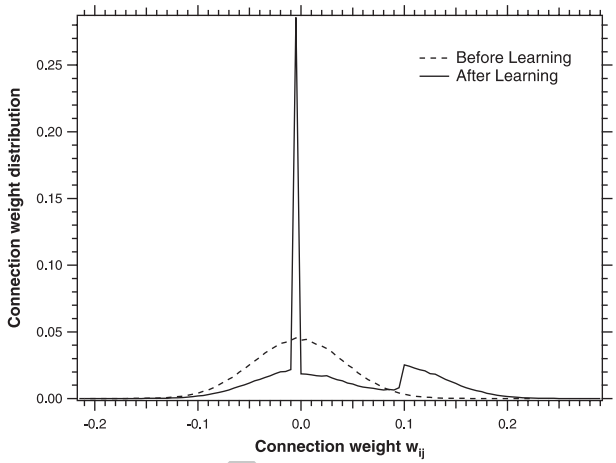

Figure 6 Weight distribution before and after learning.

Likewise, we note

$$a_{i,j} = \max(a_{ij}, a_{ji}) \tag{7}$$

We then define the degree (connectivity) of node i as:

$$k_i = \sum_{j=1}^{N} a_{i,j} \tag{8}$$

k_i thus denotes the number of i's neighbors (a neighbor is a neuron with at least one connection leading into or out of i). The average clustering coefficient C expresses the probability that two nodes connected to a third one are also connected together (degree of cliquishness). The clustering index C_i of node number i is:

$$C_i = \frac{1}{k_i(k_i-1)} \sum_{j,h} a_{i,j} a_{i,h} (a_{jh} + a_{hj}) \tag{9}$$

With this definition at hand, the clustering index takes link directionality (reciprocal links) into account, but only when reciprocal links are between i's neighbors (see Figure 5). Reciprocal links linking i to its neighbors are thus not explicitly accounted for in C_i. This information can be found, for instance, in the average density of reciprocal links.

We then classically define the clustering index C, as the average value of C_i over the network

$$C = N^{-1} \sum_{i=1}^{N} C_i \tag{10}$$

Let $d(i, j)$ be the shortest path (in number of links) between neuron i and j, then the mean shortest path (*MSP*) is its average over the $N(N-1)$ nonidentical neuron pairs

$$MSP = 1/(N^2 - N) \sum_{i,j} d(i,j) \tag{11}$$

The probability distribution of the weights before and after learning is displayed in Figure 6. Initially, the distribution follows the Gaussian law described above (Section 3). After learning, some of the weights still display their initial Gaussian distribution (unlearned weights). However, the most prominent event is the emergence of two new modes in the weight distribution, with a first group of weights centered around 0, and another group, centered around 0.12. Weights around 0.12 are those corresponding to learned connections linking together active neurons (see Section 4). Weights around 0 correspond to initially negative weights that are bounded by the condition that a weight may not change its sign. However, the peak near 0.12 was not particularly expected because the positive weights distribution could have been almost uniform.

Properly setting the threshold ϵ, we obtain the network made of the largest synaptic connections only. Hence, thresholding enables us to gradually isolate the active weight network from the inactive part. Before learning, weights are randomly (homogeneously) distributed over the network whatever the applied threshold, so that the average MSP (Figure 7), as well as the average clustering index (Figure 8), remain identical to their values in a comparable random network.

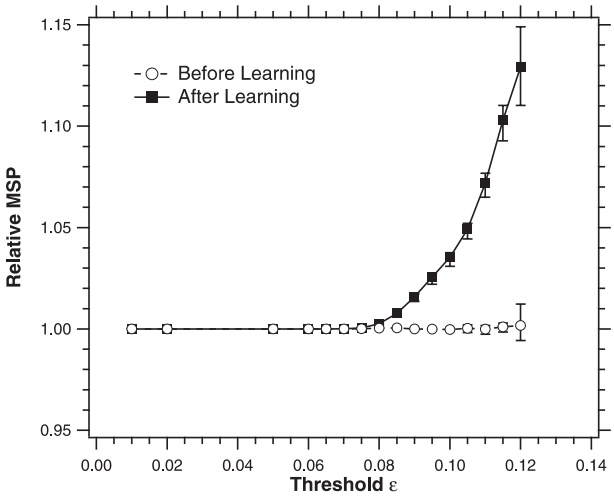

Figure 7 Mean shortest path versus threshold ϵ. Values are normalized by those observed on a random network with same number of neurons and connections. Presented are averages over three different initial conditions. The bars indicate corresponding minimum and maximum values.

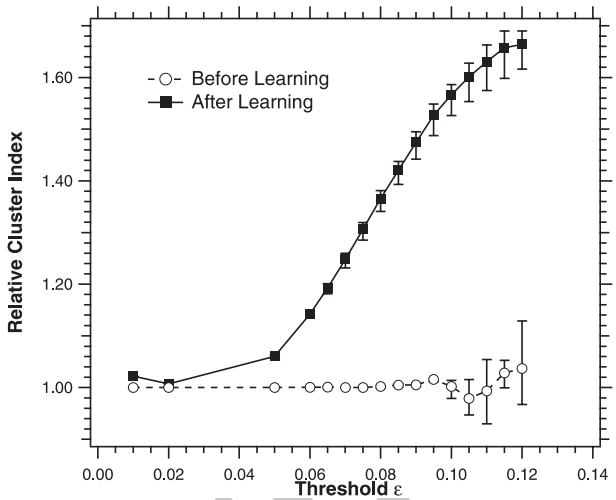

Figure 8 Average clustering index versus threshold ϵ. Values are normalized by those observed on a random network with same number of neurons and connections. Presented are averages over three different initial conditions. The bars indicate corresponding minimum and maximum values.

After learning, increasing thresholds (thus gradually isolating the active neuron network) reveal a slight increase (less than 15%) in the MSP (see Figure 7). This indicates that the average degree of separation between two neurons of the active population increases only slightly. However, the average clustering index increases up to 65% (as compared with purely random networks, see Figure 8). This denotes that after learning, the distribution of the large synaptic weights over the network is no longer random. Before learning, the larger-weight synapses are randomly distributed across the network. In particular, the probability that two neurons i and j are connected through a large synaptic weight w_{ij} does not depend on the overall connectivity pattern of these two neurons. The large increase in the clustering index after learning indicates that in this case, the probability that two neurons i and j are connected through a large-weight synapse w_{ij} is much more likely if i and j have a third common neighbor k to which they are connected via two large-weight synapses (w_{ik} and w_{jk}). Hence the probability of finding triangular connectivity circuits between neurons i, j and k in which all the implied synaptic weights are strong, is increased by learning. In other words, after learning, the distribution of the large weights over the network is no longer random because strong synaptic weights have preferentially assembled into triangles. However, this distribution of the large weight synapses still guarantees that the number of large synaptic weights separating any two neurons in the network remains very low.

To further demonstrate the reality of this organized weight distribution, we proceeded to the following weight shuffling simulations. After Hebbian learning, each weight w_{ij} of the weight matrix $\mathcal{W}$ was exchanged, with probability p, with another randomly chosen weight w_{kl}. Figures 9 and 10 show the MSP and clustering index, respectively, for various shuffling probabilities p. Clearly, as p increases, the corresponding weight matrices progressively lose their organized properties and display characteristics that tend toward those of random networks. Hence, random weight shuffling destroys the organized weight structure and reverses the structural effects of the Hebbian learning rule. Further measurements, such as the clustering index dependence on connectivity or connectivity correlations (Vázquez, Pastor-Satorras, & Vespignani, 2002) have also been computed and will be reported elsewhere.

The two previous properties (high clustering and short distances) are the signature of so-called "small-world" networks. In the broadest sense, the "small-world" phenomenon only relates to sparse networks that display a low mean shortest path (or more pre-

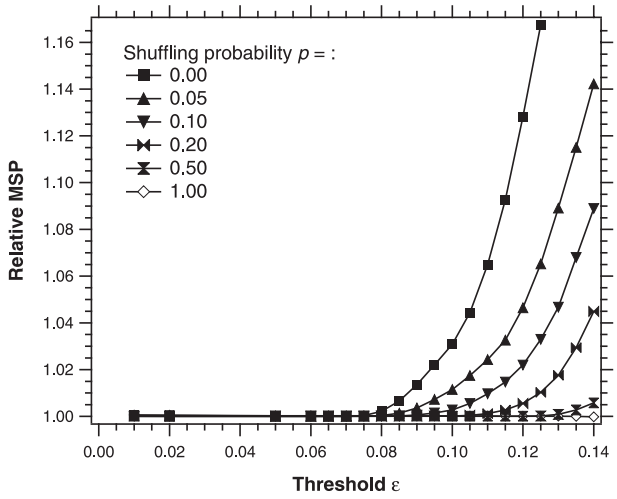

Figure 9 Mean shortest path versus threshold ϵ for different values of the shuffling probability p. After learning, each weight w_{ij} of the weight matrix was exchanged, with probability p, with a randomly chosen other weight w_{kl}. Values are normalized by those observed on a random network with same number of neurons and connections.

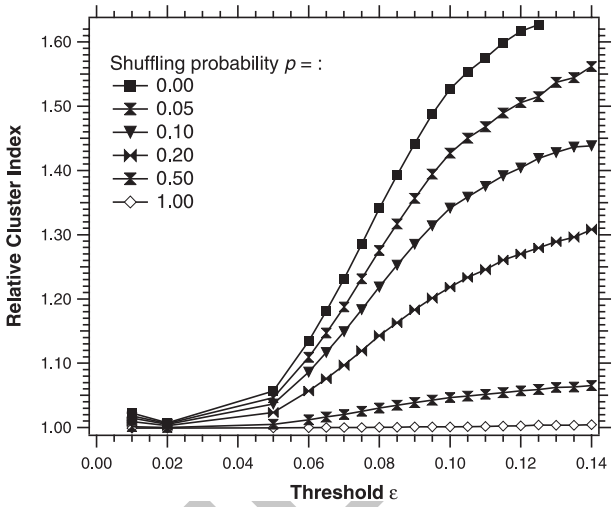

Figure 10 Average clustering index versus threshold ϵ for different values of the shuffling probability p. After learning, each weight w_{ij} of the weight matrix was exchanged with probability p with a randomly chosen other weight w_{kl}. Values are normalized by those observed on a random network with same number of neurons and connections.

cisely one that scales as the logarithm of the network size). However, in recent literature, the term "small-world network" is usually employed in the sense of Watts & Strogatz's (1998) model, where "small-world" networks are networks with both a low mean shortest path (i.e. displaying the "small-world" phenomenon) *and* a clustering index that is much higher than in comparable purely random networks. Actually, this definition encompasses non-trivial networks in which both the local (high clustering index) and global (short mean shortest path) properties are optimal. In this sense, our present results show that after learning, the network built by the strongest synaptic weights is such a "small-world" network.

Interestingly, "small-world" networks are also encountered in many real-world networks such as those formed by cell metabolism (Jeong et al., 2000), the citation of scientific articles (Newman, 2001), software architecture (Valverde et al., 2002), the Internet (Barabasi & Albert, 1999) or electronic circuits (Cancho et al., 2001). Furthermore, several studies have shown that this organization might be common in real biological neural networks. Indeed, it has been evidenced for the complete nervous system of the worm *Caenorhabditis elegans* (Watts & Strogatz, 1998) and mammal cortical area networks (Sporns & Zwi, 2004; Sporns et al., 2004). Hence our results yield the formulation of the hypothesis that small-world organizations could spontaneously emerge as a result of generic forms of Hebbian learning. Of course, this hypothesis will need to be tested in future work.

6 Conclusion

We show that beginning with a randomly connected neural network, and running a Hebbian learning rule produces the following behaviors:

- the dynamics reduces from chaos to a limit cycle and finally to a fixed point;
- the set of "active" neurons is reinforced, and is specific to the applied input pattern;
- the structure composed by the large synaptic weights organizes as a "small-world" network.

This picture leads us to formulate the hypothesis that Hebbian learning in neural networks could shape the network structure to produce small-world networks. Interestingly, this structure itself reflects the small group of active neurons sustaining the initially chaotic dynamics. This also questions the relevance of small-

world architectures for storing and processing information.

However, the effects observed on the network structure are obtained after hundreds of learning steps, whereas only a few dozen of them are enough to reduce the dynamics and recognize a learned pattern (Daucé et al., 1998). This means firstly that the structure observed after learning may only be the result of long-term learning shaping the global structure of the network. Secondly, this also means that besides the dynamics of the network, we have to take its "dynamical topology" (as opposed to the structural topology) into account. This dynamical topology is a functional one that relates neurons responding to the same input. Indeed the response of a neuron may be highly dependent on the input frequency and may not directly depend on the existence of a direct link between the two neurons (Cessac & Sepulchre, 2004). Thus, one further step will be the investigation of the influence of learning on the ability to transmit a signal from one neuron to another.

The study of the network structure presented here has only been performed after learning one input. It is now necessary to investigate this structure after learning several inputs. We expect to see the same kind of small-world structure.

Our system is a discrete time one. We can expand it into two directions. The first one concerns dealing with "spikes" rather than firing rates. We have already shown that the network dynamics in this case are qualitatively identical the properties exposed here (Daucé, 2004). The other direction deals with introducing continuous time dynamics instead of discrete time ones. This is a very important issue for the following reasons. Firstly, though synchronizing processes occur, the natural brain does not compute with fixed discrete time steps. Some results concerning chaos in the same kind of dynamical systems have already been published (Sompolinsky, Crisanti, & Sommers, 1988). We also performed stability simulations in the continuous time version of our equations. We obtained results that are similar to those reported here in the discrete time case, but for another range of parameters (Quoy, 2004). Thus, bifurcations and chaos are also observed. However, it remains unclear whether we will keep the same learning and retrieval properties. Indeed, our learning scheme mainly relies on the correlation between active neurons at two successive time steps. In the discrete time system, these active neurons are changing quite rapidly because at each time step the dynamics goes from one part of the attractor to another one. In the continuous time system, we will lose this property. Therefore, learning may have to be adapted to this different behavior. This remains to be investigated.

Acknowledgments

This work was supported by a French ACI program (neurosciences intégratives et computationnelles) on the dynamics of biologically plausible neural networks in collaboration with M. Samuelides (SupAéro, Toulouse), G. Beslon (INSA, Lyon), and E. Daucé (Perception et mouvement, Marseille). The authors also acknowledge support from the French National Research Agency (ANR, project JC05_63935).

References

Amari, S. (1972). Characteristics of random nets of analog neuron-like elements. *IEEE Transactions on Systems Man and Cybernetics*, *2*(3).

Barabasi, A., & Albert, R. (1999). Emergence of scaling in random networks. *Science*, *286*, 509–512.

Barabasi, A., & Oltvai, Z. (2004). Network biology: understanding the cell's functional organization. *Nature Reviews, Genetics*, *5*, 101–113.

Cancho, R. F., Janssen, C., & Solé, R. (2001). The topology of technology graphs: the small-world patterns in electronic circuits. *Physical Review E*, *64*, 32767.

Cessac, B., Doyon, B., Quoy, M., & Samuelides, M. (1994). Mean-field equations, bifurcation map and route to chaos in discrete time neural networks. *Physica D*, *74*, 24–44.

Cessac, B., & Sepulchre, J. (2004). Stable resonances and signal propagation in a chaotic network of coupled units. *Physical Review E*, *70*, 056111.

Daucé, E. (2004). Short term memory in recurrent networks of spiking neurons. *Natural Computing*, *3*, 135–157.

Daucé, E., Quoy, M., Cessac, B., Doyon, B., & Samuelides, M. (1998). Self-organization and pattern-induced reduction of dynamics in recurrent networks. *Neural Networks*, *11*, 521–533.

Daucé, E., Quoy, M., & Doyon, B. (2002). Resonant spatio-temporal learning in large random neural networks. *Biological Cybernetics*, *87*, 185–198.

Doyon, B., Cessac, B., Quoy, M., & Samuelides, M. (1993). Chaos in neural networks with random connectivity. *International Journal of Bifurcation and Chaos*, *3*(2), 279–291.

Eguiluz, V., Chialvo, D., Cecchi, G., & Apkarian, A. (2005). Scale-free brain functional networks. *Physical Review Letters*, *94*, 018102.

Freeman, W. (1987). Simulation of chaotic eeg pattern with a dynamic model of the olfactory system. *Biological Cybernetics*, *56*, 139–150.

Jeong, H., Tombor, B., Albert, R., Oltvai, Z., & Barabasi, A. (2000). The large-scale organization of metabolic networks. *Nature*, *407*, 651–654.

Newman, M. (2001). The structure of scientific collaboration networks. *Proceedings of the National Academy of Sciences USA*, *98*, 404–109.

Quoy, M. (2004). Bifurcation scheme of continuous time random recurrent neural networks. Internal report.

Skarda, C., & Freeman, W. (1987). How brains make chaos in order to make sense of the world. *Behavioral and Brain Sciences*, *10*, 161–195.

Sompolinsky, H., Crisanti, A., & Sommers, H. (1988). Chaos in random neural networks. *Physical Review Letters*, *61*, 259–262.

Sporns, O., Chialvo, D., Kaiser, M., & Hilgetag, C. (2004). Organization, development and function of complex brain networks. *Trends in Cognitive Sciences*, *8*(9), 418–425.

Sporns, O., & Zwi, J. (2004). The small world of the cerebral cortex. *Neuroinformatics*, *2*, 145–162.

Tsuda, I. (2001). Towards an interpretation of dynamic neural activity in terms of chaotic dynamical systems. *Behavioral and Brain Sciences*, *24*, 793–847.

Valverde, S., Cancho, R. F., & Solé, R. (2002). Scale-free networks from optimal design. *Europhysics Letters*, *60*, 512–517.

Vázquez, A., Pastor-Satorras, R., & Vespignani, A. (2002). Large-scale topological and dynamical properties of the internet. *Physical Review E*, *65*, 066130.

Watts, D., & Strogatz, S. (1998). Collective dynamics of "small-world" networks. *Nature*, *393*, 440–442.

Yao, Y., & Freeman, W. (1990). Model of biological pattern recognition with spatially chaotic dynamics. *Neural Networks*, *3*, 153–170.

About the authors

Hugues Berry received his Ph.D. in Biochemistry and Biophysics from the Technology University of Compiègne (France) in 1999. In 2000 he joined the ERRMECe laboratory of the University of Cergy-Pontoise (France) as an associate professor. Since 2004, he has been working in secondment with the Alchemy group at INRIA, Orsay, France. His research interests include modeling of complex biological systems, learning and topology in biological neural networks and biologically-inspired approaches for computer architectures. *Mailing address*: INRIA Futurs – 4 rue J. Monod 91893 Orsay, France
Email: hugues.berry@inria.fr

Mathias Quoy received his Ph.D. in Computer Science from the Ecole Nationale Supérieure de l'Aéronautique et de l'Espace (ENSAE) in Toulouse (France) in 1994. In 1995 he joined the ETIS laboratory of the University of Cergy-Pontoise (UCP, France) as an associate professor. Since 2004, he has been a Professor at the technological Institute at UCP. His research interests include the study of neural networks as dynamical systems and their use in biologically inspired control and planning architecture for autonomous mobile robots. *Mailing address*: ETIS 6, av. du Ponceau 95014 Cergy-Pontoise, France
Email: quoy@ensea.fr