

An interpretative recurrent neural network to improve pattern storing capabilities - dynamical considerations

Colin Molter[†], Utku Salihoglu[†] and Hugues Bersini[†]

[†]Laboratory of artificial intelligence IRIDIA cp 194/6
Université Libre de Bruxelles - 50 av F.Roosevelt, 1050 Brussels, Belgium.
Email: {cmolter-usalihog}@iridia.ulb.ac.be, bersini@ulb.ac.be

Abstract—Seminal observations performed by Skarda and Freeman [1] on the olfactory bulb of rabbits during cognitive tasks have suggested to locate the basal state of behavior in the network's spatio-temporal dynamics. Following these neurophysiological observations, a new learning task for recurrent neural networks has been proposed by the authors in recent papers [2], [3]. This task consists in storing information in spatio-temporal dynamical attractors of these artificial networks. Two innovative learning algorithms are discussed and compared here, based on dynamical considerations. Firstly an iterative supervised Hebbian learning algorithm were the all of the information is fully specified. Secondly, an iterative unsupervised Hebbian learning algorithm were the network has to categorize external stimuli, by building itself its own internal representations.

1. Introduction

Since Hopfield and Grossberg precursor works ([4], [5]), the privileged regime to code information in artificial nets has been fixed point attractors. However, many neurophysiological reports ([6], [7], [8], [9]) tend to indicate that brain dynamics is much more "dynamical" than fixed point and more faithfully characterized by cyclic and weakly chaotic regimes instead. Based on these observations, the authors have shown in previous works [10] how synaptic matrix generated randomly allows to exploit cyclic attractors for information encoding. An information is made of a pair of data: the stimulus and the corresponding limit cycle attractor. Promising results were obtained showing both a high encoding potential and a relation between this potential and the chaotic dynamics' presence in the network: high potential networks have stronger presence of chaotic dynamics.

To encode specific information, two different learning tasks have been proposed in previous papers ([2], [3]). To follow neurophysiological observations, the learning of the synaptic matrix has been chosen to be based on local Hebbian mechanisms. Comparisons of these two algorithms, based on dynamical observations, are performed in this paper.

2. Description of the learning tasks and of the model

2.1. The model

The network is fully connected. Each neuron's activation is a function of other neurons' impact and of an external stimulus. The neurons activation f is continuous and is updated synchronously by discrete time step. The activation value of a neuron x_i at a discrete time step n is:

$$\begin{aligned} x_i(n+1) &= f(g \text{net}_i(n)) \\ \text{net}_i(n) &= \sum_{j=1}^N w_{ij}x_j(n) + w_{is}t_i \end{aligned} \quad (1)$$

where N is the number of neurons (set to 25 in this paper for legibility), g is the slope parameter, w_{ij} and w_{is} are connections weight and t_i is the external stimulus for the neuron i . The saturating activation function f is taken continuous (here tanh) to ease the study of the networks' dynamical properties. To be able to compare binary patterns with internal states of the continuous network, a filter layer is added based on the sign function.

2.2. The learning tasks

Two different learning tasks are proposed. Both of them consist in the storing of external stimuli in limit cycle attractors of the network's internal dynamics. In the first one, each data stored in the network is fully specified a priori.

In the second one, the information is not fully specified a priori: only the external stimuli are known before learning. It means that the limit cycle attractor $\zeta^{\mu,i}$ associated with an external stimulus χ^μ is identified through the learning procedure. This learning procedure can be viewed as the ability of the system to create its own categories. Before learning, the data set is defined by \mathcal{D}_{bl} (bl standing for "before learning"):

$$\mathcal{D}_{bl} = \{\mathcal{D}_{bl}^1, \dots, \mathcal{D}_{bl}^q\} \quad \mathcal{D}_{bl}^\mu = \chi^\mu \quad \mu = 1, \dots, q \quad (2)$$

After learning, the data set becomes:

$$\mathcal{D}_{al} = \{\mathcal{D}_{al}^1, \dots, \mathcal{D}_{al}^q\} \quad \mathcal{D}_{al}^\mu = (\chi^\mu, (s^{\mu,1}, \dots, s^{\mu,l_\mu})) \quad (3)$$

l_μ is the size of the cycle associated to the stimulus χ^μ .

3. Implementation of iterative Hebbian algorithms

3.1. The supervised Hebbian learning algorithm

This innovative algorithm has been described by the authors in previous papers (see [2]). It is based on classical iterative Hebbian algorithms described in [11]. The principle can be described as follows: at each learning iteration, the stability of every nominal pattern ξ^μ , is tested. Whenever one pattern has not reached stability yet, the responsible neuron i sees its connectivity reinforced by adding a Hebbian term to all the synaptic connections impinging on it:

$$\begin{aligned} w_{ij} &\mapsto w_{ij} + \varepsilon_s S_i^{\mu,v+1} S_j^{\mu,v} \\ w_{is} &\mapsto w_{is} + \varepsilon_b S_i^{\mu,v+1} \chi_i^\mu \end{aligned} \quad (4)$$

where ε_s and ε_b respectively define the learning rate and the stimulus learning rate.

This algorithm has been improved by adding explicit noise during the learning phase: in order to not only store the patterns, but also to ensure a sufficient enough content-addressability, we had to “excavate” the basins of attraction. To compare the network’s continuous internal states with bit-patterns, a filter layer quantizing the internal states is added. It enables to perform symbolic investigations on the dynamical attractors.

3.2. The unsupervised Hebbian learning algorithm

This innovative algorithm has been described by the authors in a previous paper [3]. The main difference with the supervised Hebbian algorithm lies in the nature of the information learned. In the supervised version, each information to learn is given a priori and is fully specified. In the unsupervised version, only the external stimuli are given a priori, the information is a consequence of the learning. In other words, the information is “generated” through the learning procedure assigning a “meaning” to each external stimulus: the learning procedure enforces a mapping between each stimulus of the data set and a limit cycle attractor of the network’s inner dynamic, whatever it is.

Inputs of this algorithm are a data set \mathcal{D}_{bl} to learn (see Equation 2) and a range $[\min_{c_s}, \max_{c_s}]$ which defines the bounds of the accepted periods of the limit cycle attractors coding the information. This algorithm can be broken down in two phases which are constantly iterated until convergence:

1. proposal of an attractor code for each stimulus χ^μ :
the network is stimulated by χ^μ , consequently trapping it in an attractor **output** $^\mu$. To constrain the network as little as possible, the meaning assigned to the stimulus χ^μ is obtained by associating it with a new version of the attractor **output** $^\mu$, called **cycle** $^\mu$, respecting the periodic bounds $[\min_{c_s}, \max_{c_s}]$ and being “original”¹;

¹original means that each pattern composing the limit cycle attractor must be different from all patterns of the data set

2. learning the information:

once all new attractors **cycle** $^\mu$ have been proposed, there will be tentatively learned by relying on a supervised procedure. However, only a limited number of iterations of the supervised algorithm is performed in order to avoid constraining the network too much.

It has to be noted that this learning mechanism implicitly supplies the network with an important robustness to noise. First of all, the coding attractors are the ones naturally proposed by the network. Secondly, they need to have large and stable basins of attraction in order to resist the process of trials, errors and adaptations.

4. Encoding capacity

The number of information that can be learned in an acceptable amount of time in RNNs’ limit cycle attractors through the supervised and the unsupervised algorithms have already been discussed in previous works ([2] and [3]). The main results are briefly summarized below.

It has been demonstrated how the unsupervised learning outperforms its supervised counterpart: the storing capacity can be enhanced by a factor larger than six. By studying and comparing the robustness to the noise injected in the external stimulus and in the initial states, it has been shown that again, the unsupervised learning algorithm considerably improves robustness.

5. Dynamical considerations

Tests performed here aim at analyzing the so-called background or spontaneous dynamics obtained when the network is feeded with other external stimuli than the learned ones. Quantitative analyses have been performed using two kinds of measures: the mean Lyapunov exponent² and the probability to have chaotic dynamics. Both measures come from statistics on a huge number (here 1000) of learned networks. For each learned network, dynamics obtained from 1000 randomly chosen external stimuli and initial states have been tested.

Figure 1 compares the chaotic dynamics’ presence in 25-neurons networks learned with different data set of period-4 cycles by plotting the mean value of the first Lyapunov exponent and the probability to have chaotic dynamics. This figure demonstrates how networks learned through the supervised algorithm are becoming more and more chaotic while the learning task is strengthened. In the end, the probability of falling in a chaotic dynamics is equal to one, with huge mean Lyapunov exponents. In the unsupervised case, when networks are asked to categorize the stimuli using their own representations, even after learning a huge

²The computation of the first Lyapunov exponent is done empirically while the computation of the Lyapunov spectrum is performed through Gram-Schmidt re-orthogonalization of the evolved system’s Jacobian matrix (which is estimated at each time step from the system’s equations). Both methods have been initially proposed by Wolf [12].

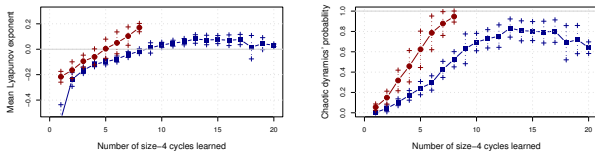
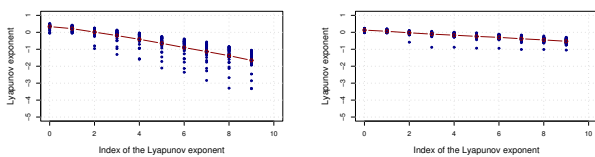


Figure 1: Impact of the learning algorithm on the complexity of the global dynamics. The mean Lyapunov exponent and the probability to have chaotic dynamics of learned networks are plotted in function of the learning sets' size. Circles are associated to supervised learned networks while squares are associated to unsupervised learned networks. $N=25$.

amount of data, networks do not become fully chaotic, and chaotic dynamics obtained have smaller Lyapunov exponents.

Figure 2 compares Lyapunov spectra obtained from chaotic dynamics occurring in learned networks. Lyapunov spectra obtained in supervised learned networks are characteristic of deep chaos or hyper-chaos. In hyper-chaos [13] the presence of more than one positive Lyapunov exponent is expected and these exponents are expected to be huge. By contrast, Lyapunov spectra obtained in unsupervised learned networks are the characteristic of a frustrated chaos. In this type of chaos, the dynamics is attracted to learned memories—which is indicated by negative Lyapunov exponents—while in the same time it escapes from them—which is indicated by the presence of at least one positive Lyapunov exponent. However, this positive Lyapunov exponent must be only slightly positive in order not to erase completely the system's past history and thus to keep traces of the learned memories. This regime of frustration is increased by some modes of neutral stability indicated by the presence of many exponents whose values are close to zero [13].

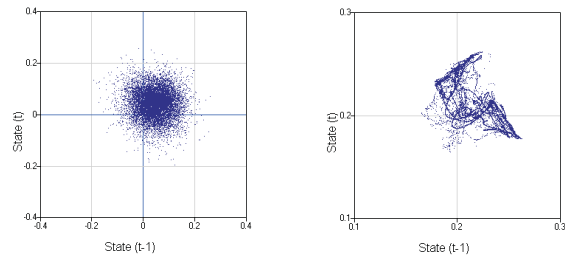


(a) Sup learning-12 size-2 cycles (b) Unsup learning-50 size-2 cycles

Figure 2: These figures show the ten first Lyapunov exponents. Each time, 10 different learning sets have been tested. For each obtained network, 100 chaotic dynamics have been studied. $N=25$.

Further evidences are obtained from return maps analysis. In Figure 3, it appears that networks having learned in a supervised way too many information show presence of very uninformative deep chaos similar to white noise. In other words, having too many competing limit cycle at-

tractors leads to unstructured dynamics. The unsupervised Hebbian learning instead preserves more structure in the chaotic dynamics and leads to a form of chaos called frustrated chaos [14]. It is a dynamical regime which appears in a network when the global structure is such that local connectivity patterns responsible for stable and meaningful oscillatory behaviors are intertwined, leading to mutually competing attractors and unpredictable itinerancy among brief appearance of these attractors. In this paper, this chaos appear after Hebbian learning the cycles associated to stimuli.



(a) Sup learning-6 size 4 cycles (b) Unsup learning-16 size 4 cycles

Figure 3: Return map comparisons between an hyper-chaos and a frustrated chaos It has been obtained after . $N=25$

Figure 4 compares a deep chaos and a frustrated chaos by plotting the probability of presence of the nearby limit cycle attractors in chaotic dynamics. In both figures, 25-neurons networks have learned 4 data in limit cycles attractors of size 5. The use of the supervised Hebbian algorithm (left figure) hardly constrains the network and, as a consequence, chaotic dynamics appear very uninformative: by shifting the external stimulus from one attractor to another one, the chaos in-between has lost any information concerning these two limit cycle attractors. In contrast, when learning using the unsupervised algorithm (right figure), when driving the dynamics by shifting the external stimulus from one attractor to another one, the chaos encountered on the road appears much more structured: strong presence of the nearby limit cycles is easy to observe.

6. Conclusion and future works

This paper aims at comparing two ways to code the information in the network's cyclic dynamics by relying on innovative iterative Hebbian algorithms. In the first way, stimuli are encoded in a priori specified cyclic attractors of the network's dynamics. In the second way, the network create by itself its own internal representations to be associated with the stimuli: the semantics of the information is left unprescribed until the learning occurs. Based on the encoding capacities, the superiority of the unsupervised way to encode the stimuli is clearly demonstrated. Based on dynamical analyses of learned networks when feeded by unlearned external stimuli, iterative Hebbian learning can

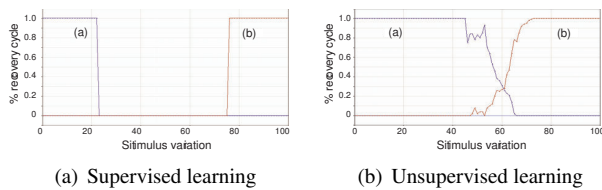


Figure 4: Probability of presence of specific limit cycle attractors when the external stimulus is slowly shifted between two external stimuli previously learned (region (a) and (b)). The network's dynamics goes from the limit cycle attractor associated to the former stimulus to the limit cycle attractor associated to the latter stimulus. In between chaos shows up, which turns out to be "frustrated" in the unsupervised case. $N=25$.

be seen as an alternative road to chaos: the more information the network has to store the more chaotic it spontaneously tends to behave. However, when relying on supervised Hebbian learning, the background chaos spreads widely and adopts a very unstructured shape similar to white noise. In contrast, unsupervised learning, by being more "respectful" of the network intrinsic dynamics, maintains much more structure in the obtained chaos. It is still possible to observe in the chaotic regime the traces of the learned attractors. This complex but still very informative regime has been called the "frustrated chaos".

In current works, to stay in line with neurophysiological observations where a robust correlation is observed between behavioral states and transient periods of synchronization of oscillating neuronal discharges in the frequency range of gamma oscillations [8], [15], [9], [16], we propose to retrieve the information stored in the cycles of the network's internal dynamics by relying on synchronization occurring between neuronal groups. We have already demonstrated that the synchronization picture associated with a frustrated regime keeps traces of the neuronal groups characterizing the two nearby limit cycle attractors. Because the static synchronization picture compresses some information of the system's past history, we propose to use it in future works as the basal information in more complex neuronal architectures.

References

- [1] W.J. Freeman. Simulation of chaotic eeg patterns with a dynamic model of the olfactory system. *Biological Cybernetics*, 56:139–150, 1987.
- [2] C. Molter, U. Salihoglu, and H. Bersini. Learning cycles brings chaos in continuous hopfield networks. *Proceedings of the IJCNN*, 2005.
- [3] C. Molter, U. Salihoglu, and H. Bersini. Introduction of an hebbian unsupervised learning algorithm to boost the encoding capacity of hopfield networks. *Proceedings of the IJCNN*, 2005.
- [4] J.J. Hopfield. Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences USA*, 79:2554–2558, April 1982.
- [5] S. Grossberg. *Neural Networks and Natural Intelligence*. MIT Press, Cambridge, 1992.
- [6] J. Nicolis and Tsuda. Chaotic dynamics of information processing. the 'magic number seven plus minus two' revisited. *Bulletin of Mathematical Biology*, 47:34365, 1985.
- [7] A. Babloyantz and Loureno. Computation with chaos: A paradigm for cortical activity. *Proceedings of National Academy of Sciences*, 91:9027–9031, 1994.
- [8] C.A. Skarda and W. Freeman. How brains make chaos in order to make sense of the world. *Behavioral and Brain Sciences*, 10:161–195, 1987.
- [9] E. Rodriguez, N. George, J.P. Lachaux, B. Renault, J. Martinerie, B. Reunault, and F.J. Varela. Perception's shadow: long-distance synchronization of human brain activity. *Nature*, 397:430–433, 1999.
- [10] C. Molter and H. Bersini. How chaos in small hopfield networks makes sense of the world. *Proceedings of the IJCNN conference*, 2003.
- [11] B.M. Forrest and D.J. Wallace. *Models of Neural Networks*, volume 1, chapter Storage Capacity and Learning in Ising-Spin Neural Networks, pages 129–156. Springer, 2nd edition, 1995.
- [12] A. Wolf, J.B. Swift, H.Swinney, and J.A. Vastano. Determining lyapunov exponents from a time series. *Physica*, (D16):285–317, 1984.
- [13] K. Kaneko and I. Tsuda. Chaotic itinerancy. *Chaos: Focus Issue on Chaotic Itinerary*, 13(3):926–936, 2003.
- [14] H. Bersini and P. Sener. The connections between the frustrated chaos and the intermittency chaos in small hopfield networks. *Neural Networks*, 15:1197–1204, 2002.
- [15] C.M. Gray, P. Kning, A.K. Engel, and W. Singer. Oscillatory responses in cat visual cortex exhibit intercolumnar synchronization which reflects global stimulus properties. *Nature*, 338:334–337, 1989.
- [16] F. Varela. Resonant cell assemblies: a new approach to cognitive function and neuronal synchrony. *Biol.Res.*, 28:81–95, 1995.