

A multi-modular associator network for simple temporal sequence learning and generation

Michael Lawrence¹ and Thomas Trappenberg¹ and Alan Fine²

1- Dalhousie University - Faculty of Computer Science
6050 University Avenue, Halifax, NS, B3H 1W5 - Canada

2- Dalhousie University - Department of Physiology and Biophysics
5850 College Street, Halifax, NS B3H 1X5 - Canada

Abstract. Temporal sequence generation readily occurs in nature. For example performing a series of motor movements or recalling a sequence of episodic memories. Proposed networks which perform temporal sequence generation are often in the form of a modification to an auto-associative memory by using hetero-associative or time-varying synaptic strengths, requiring some pre-chosen temporal functions. Intra-modular synapses are trained auto-associatively with a Hebb rule, while a set of inter-module synapses are hetero-associative. Our model is compared to one by Lisman, which uses hetero-associative recurrent synapses in one of the modules, and auto-associative synapses between modules.

1 Introduction

Understanding brain mechanisms for storing and recalling sequences is an important unsolved problem in neuroscience. From a computational sequence perspective, a sequence S of length p is a list $\xi^1, \xi^2, \dots, \xi^p$ of patterns, each pattern representing a memory in the sequence characterized by the firing rate of each node in a recurrent (associative) network (i.e. $\xi^\mu \in \{1, -1\}^N$ where $\xi_i^\mu = 1$ indicates node i is firing at a maximal rate and $\xi_i^\mu = -1$ means node i is not firing at all). Here we only discuss *simple* sequences in which the presence of one pattern predicts the next pattern in the sequence uniquely.

Most approaches to storing a sequence of patterns in memory are based on modification to associative networks. Hopfield [1] suggested a modification to the Hebbian learning rule using hetero-associatively trained weights

$$w_{i,j}^H = \frac{1}{N} \sum_{\mu} \xi_i^{\mu+1} \xi_j^{\mu} \quad (1)$$

combined with the usual auto-associative weights

$$w_{i,j}^A = \frac{1}{N} \sum_{\mu} \xi_i^{\mu} \xi_j^{\mu} \quad (2)$$

in a node governed by leaky integrator dynamics on a time scale of τ

$$\tau \frac{\delta h_i(t)}{\delta t} = -h_i(t) + \sum_j (w_{i,j}^A + \lambda w_{i,j}^H) S_j(t) \quad (3)$$

$$S_i(t) = \tanh(h_i(t)). \quad (4)$$

A recurrent network using Equation 3 should be able to move its state from one pattern to the next in the sequence. However, in simulations the result is that, when λ is too small, the network makes no transitions between patterns at all, usually attracting to the pattern closest to $\mathbf{S}(0)$. As λ gets larger the network tends to transition to a later pattern in the sequence before it has come to fully represent the current pattern in the sequence, causing it to overlap a number of consecutive patterns and consequently lose the sequence entirely. Accurate sequence generation using this approach has only been successfully demonstrated for sequences of length four [1, 2]. More successful approaches depend on some explicit timing mechanism, for example delayed synapses [3, 4, 5], time-dependent synaptic strengths [6, 7] or the interaction from time-decaying short term memory of previous patterns in the sequence [8, 9, 4, 3].

Various multi-modular approaches to sequence processing [5, 6, 10] have been previously proposed, each with different capabilities in terms of the types of sequences they can recall and recognize. Here we propose a pair of connected recurrent associative networks X and Y to generate simple temporal sequences of patterns. Each module uses auto-associative Hebbian learning in its recurrent synapses, with hetero-associatively trained synapses (which associate a given pattern with the pattern following it in the sequence) from Y to X . The model is compared to one with major features proposed by Lisman [10], which in contrast uses hetero-association in one of the sets of recurrent synapses and auto-association elsewhere. By using hetero-associative synapses between, rather than within, modules, we avoid difficulties when a module moves toward the next pattern in the sequence while still having only a partial representation of the current pattern, causing poor retrieval and loss of sequence information. Rather, the inter-module hetero-associations cause one module to push the other toward the next pattern in the sequence, and when it is sufficiently close, to use auto-association to clean its representation of that pattern.

2 The Model

The model we propose uses hetero-associative synapses between modules, with the architecture shown in Figure 1, with w^{XY} , the synaptic strengths of connections from Y to X , set according to Equation 1 and all other weights set according to Equation 2. The dynamics of this model are given by

$$\tau \frac{\delta^X h_i(t)}{\delta t} = -h_i^X(t) + \lambda^{XX} \sum_{j \in X} w_{i,j}^{XX} S_j^X(t) + \lambda^{XY} \sum_{j \in Y} w_{i,j}^{XY} S_j^Y(t) \quad (5)$$

(and similarly for Y). The λ s are independent parameters used to adjust the relative strengths of the inter and intra-module connections.

Each module can be considered to work somewhat independently, cleaning up its own representation of the current pattern in the sequence ξ^μ . When module Y has a ‘‘clean enough’’ representation of ξ^μ , it begins to push module X toward the next pattern in the sequence, $\xi^{\mu+1}$, due to hetero-associations. In this way a natural timing can be achieved from the competition between the two modules:

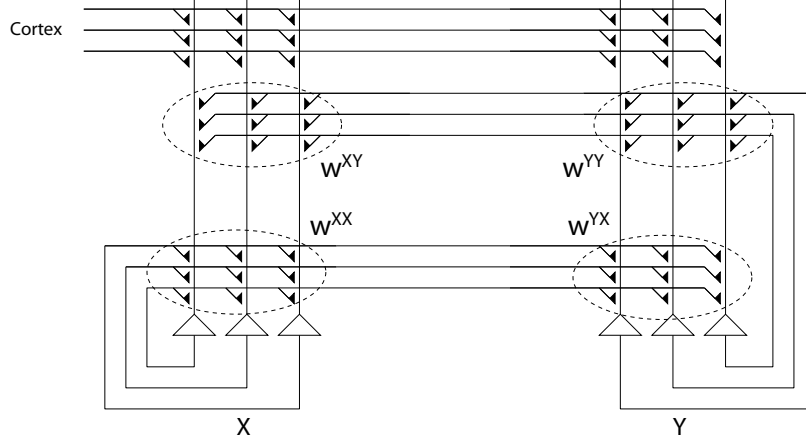


Fig. 1: The proposed network, a pair of connected modules X and Y . w^{XX} , w^{YY} , w^{YX} and w^{XY} represent the synaptic strengths in the recurrent connections in modules X and Y , and the inter-module connections from X to Y and from Y to X respectively. The network in Lisman's model [10] is similar, with w^{YY} trained hetero-associatively (Equation 1), and all other weights set auto-associatively (Equation 2). In our model w^{XY} is trained hetero-associatively (Equation 1), and all other weights auto-associatively (Equation 2).

1. When they both represent the same pattern, ξ^μ , there is competition between the recurrent synapses of module X , which want to continue attracting toward ξ^μ , and between the inter-module synapses from Y , which want to push X toward $\xi^{\mu+1}$.
2. When X represents $\xi^{\mu+1}$ and Y ξ^μ , there is a competition between the recurrent synapses of module Y , which want to continue attracting toward ξ^μ , and the inter-module synapses from X to Y , which want to push Y toward $\xi^{\mu+1}$.

The temporal behaviour is a consequence of the growth of these forces and their necessary alternation, in that, when Y 's strength is maximal X should be in transition and hence its strength is minimal, and vice-versa. For example in case 1 above, the strength of Y pushing X toward $\xi^{\mu+1}$ is related to how clearly ξ^μ is represented in Y , i.e. how close $\mathbf{S}^Y(t)$, the firing rates of nodes in module Y at time t , are to ξ^μ . As module Y draws closer to ξ^μ , eventually its strength in moving X toward $\xi^{\mu+1}$ becomes large enough to push X into the basin of attraction for pattern $\xi^{\mu+1}$. At this point X has a small strength in affecting Y as it is far away from, but moving toward, $\xi^{\mu+1}$. As X draws closer to this attractor its strength in pushing Y toward $\xi^{\mu+1}$ increases until this occurs, at which point Y 's strength becomes minimized during its transition. To achieve these dynamics the inter-module synapses must be weighted higher than the recurrent synapses, so that the forces from module Y are sufficient to affect the course of module X when Y is sufficiently close to one of the stored patterns and vice-versa. The time for which each pattern is stable can hence be adjusted with the strength of the inter-module

connections. When they are made stronger, Y has more effect on X (and vice versa), moving it from a stable pattern more quickly and resulting in a smaller amount of time in which a pattern is represented.

In order to understand how, in the brain, synaptic strengths could evolve as described, we consider a sequence of events as it occurs. Pattern ξ^1 , representing the first event in the sequence, will be excited in module X by the cortical input, and through excitatory connections from X to Y , the nodes of Y will in turn be excited to pattern ξ^1 . After this, the event which is encoded as ξ^2 takes place, causing X to be stimulated to this pattern by the cortex. Meanwhile module Y has sustained firing in the pattern ξ^1 (a phenomenon which has been observed in the hippocampus), hence excitation of ξ^1 in module Y has been followed by excitation of ξ^2 in module X , and Hebbian learning causes the hetero-association of ξ^2 from ξ^1 in the $Y \rightarrow X$ synapses. By contrast, auto-association is learned in the $X \rightarrow Y$ synapses, since excitation of ξ^1 in X results in the firing of ξ^1 in Y . Auto-association is also learned in the recurrent connections of both modules, as usual for rate models in recurrent networks.

3 Experiments

The network in Figure 1 was implemented in two cases: where w^{XY} is hetero-associative and the other weights are auto-associative (our model), and where w^{YY} is hetero-associative and the other weights are auto-associative (Lisman's model). We initialized the networks with noisy versions of the first pattern in module X and a random pattern in module Y and measured the overlap between the network states and the stored patterns, $\mathbf{S}'(t)\xi^\mu/N$. Figure 2 shows the result of a sequence recall of six random patterns using our model with $N = 1000$ nodes for each module. It is easy to find strength parameters that lead to good sequence recall. The values for the example shown in Figure 2 are $\lambda^{XX} = \lambda^{YY} = 1$, $\lambda^{YX} = 1.2$ and $\lambda^{XY} = 1.8$. Each pattern is stable for some period with large overlaps to a stored pattern.

The same experiment was attempted with Lisman's model, but we were not able to find appropriate parameters to get consistent results. We found that typically the nodes of module Y either attract quickly to some stable fixed point which does not represent any of the stored patterns, or, as λ^{YY} is increased, experiences very rapid oscillations without settling to any particular state. Module X meanwhile usually attracts to a fixed point, or will make state transitions if Y is oscillating and λ^{XY} is large enough. An example that shows some form of transitional dynamic in simplified Lisman model is shown in Figure 3. The problem with tuning the λ s is as follows: increasing λ^{YX} increases the chance that X can be brought into the basin of attraction of a particular pattern in order to clean up its representation of this pattern, but as Y continues to oscillate, this increases the chance that X will be disrupted from cleaning up its current pattern. Both modules were initialized to the first pattern without noise. Initializing the modules with different patterns in Lisman's model increased the difficulty in finding satisfactory λ s. Decreasing the number of nodes and pattern adds some stability to the sequences, but this may be an artifact of small networks.

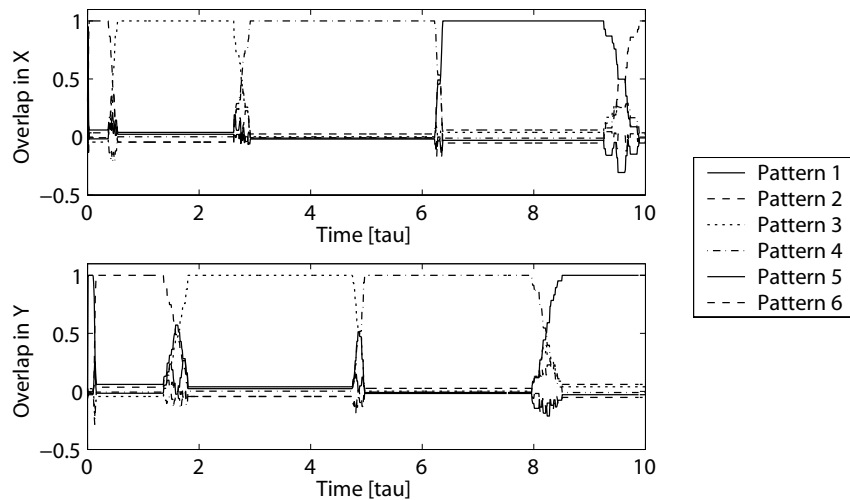


Fig. 2: Result of recall of a learned sequence of six random patterns using the proposed model with $N = 1000$ nodes in each module. The inter-module synapses from X to Y were given a weight of $\lambda^{YX} = 1.2$ times larger than the intra-module (recurrent) synapses, and the inter-module synapses from Y to X were given a weight of $\lambda^{XY} = 1.8$ times larger. Sequence recall was initiated by setting the firing rates of nodes in module X to the first pattern with 30% noise, and the firing rates of nodes in module Y to a pattern with randomly chosen binary rates. **(top)** The overlap between the firing rates $\mathbf{S}^X(t)$ of the nodes in module X and each of the stored patterns. **(bottom)** The overlap between the firing rates $\mathbf{S}^Y(t)$ of the nodes in module Y and each of the stored patterns.

4 Conclusions and Future Work

Multi-modular approaches to sequence generation have an advantage over approaches involving a single recurrent network in that timing can be achieved solely from the dynamics of the interaction between the modules. Effective approaches involving a single module rely on an explicit temporal function which must be chosen ahead of time. The model studied in this paper, with hetero-associative connections between the modules, is able to learn and generate simple sequences easily and reliably, in contrast to an architecture that connects a hetero-associative module with an auto-associative network. Our architecture permits storage of sequences with arbitrary and varying time scales; it does not require combining a trace of previous activity in a recurrent net with new activity, as the pattern in different networks can be timed differently and learning occurs between these pattern. We are currently investigating further the possible involvement of such modular network sequence storage to hippocampal function.

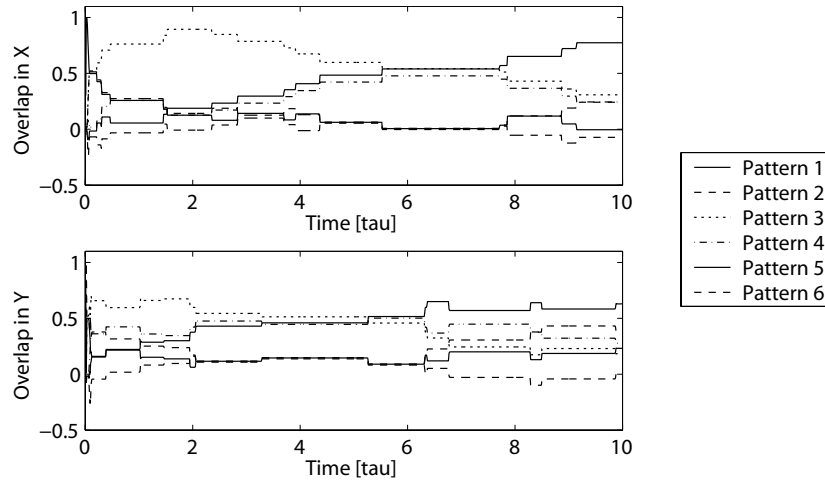


Fig. 3: Result of recall of a learned sequence of four random patterns using the modified Lisman model with $N = 1000$ nodes in each module. The parameters used were $\lambda^{XX} = 1$, $\lambda^{YY} = 2.5$, $\lambda^{YX} = 2$, $\lambda^{XY} = 4$. The simulation was performed in a similar way as that of Figure 2. Both modules were initialized with the first pattern without noise. The result shows the pattern oscillations of module Y due to its hetero-associative synapses (**bottom**), and the oscillations of module X (**top**) due to (selective) interaction from Y .

References

- [1] J. J. Hopfield. Neural Networks and Physical Systems with Emergent Collective Computational Abilities. *Proceedings of the National Academy of Science*, 79:2554–2558, April 1982.
- [2] H. Nishimori, T. Nakamura, and M. Shiino. Retrieval of spatio-temporal sequence in asynchronous neural network. *Physical Review A*, 41:3346–3354, March 1990.
- [3] D. W. Tank and J. J. Hopfield. Neural Computation by Concentrating Information in Time. *Proceedings of the National Academy of Science*, 84:1896–1900, April 1987.
- [4] D. Kleinfeld. Sequential State Generation by Model Neural Networks. *Proceedings of the National Academy of Science*, 83:9469–9473, December 1986.
- [5] D. Wang and M.A. Arbib. Complex temporal sequence learning based on short-term memory. *Proceedings of the IEEE*, 78(9):1536–1543, September 1990.
- [6] Lipo Wang. Heteroassociations of spatio-temporal sequences with the bidirectional associative memory. *IEEE Transactions on Neural Networks*, 11(6):1503–1505, 2000.
- [7] H. Gutfreund and M. Mezard. Processing of temporal sequences in neural networks. *Physical Review Letters*, 61:235–238, July 1988.
- [8] U. Riedel, R. Kühn, and J. L. van Hemmen. Temporal sequences and chaos in neural nets. *Physical Review A*, 38:1105–1108, July 1988.
- [9] H. Sompolinsky and I. Kanter. Temporal association in asymmetric neural networks. *Physical Review Letters*, 57:2861–2864, December 1986.
- [10] J.E. Lisman. Relating hippocampal circuitry to function: Recall of memory sequences by reciprocal dentate-ca3 interactions. *Neuron*, 22(2):233–242, 1999.