

Rapid learning and robust recall of long sequences in modular associator networks

Michael Lawrence, Thomas Trappenberg*, Alan Fine

Dalhousie University, Halifax, Canada

Available online 10 January 2006

Abstract

Biologically inspired neural networks which perform temporal sequence learning and generation are frequently based on hetero-associative memories. Recent work by Jensen and Lisman has suggested that a model which connects an auto-associator module to a hetero-associator module can perform this function. We modify this architecture in a simplified model which in contrast uses a pair of connected auto-associative networks with hetero-associatively trained synapses in one of the paths between them. We simulate both models, finding that accurate and robust recall of learned sequences can easily be performed with the modified model introduced here, strongly outperforming the previous architecture.

© 2005 Elsevier B.V. All rights reserved.

Keywords: Sequence memory; Hetero-associations

1. Introduction

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. In general, it is possible that $\xi^{\mu_1} = \xi^{\mu_2}$ for $\mu_1 \neq \mu_2$, in which case knowing the current pattern $\xi^{\mu_1} = \xi^{\mu_2}$ is not sufficient to determine if the next pattern is ξ^{μ_1+1} or ξ^{μ_2+1} during recall. For a pattern ξ^μ , the length of the preceding subsequence which uniquely determines ξ^μ is called the *degree* of ξ^μ [18]. The degree of a sequence is the maximum degree of its components, and a sequence is called *simple* if it has degree 1, otherwise it is known as *complex*. For example, the sequence A, B, C, D is a simple sequence, and A, B, C, D, B, C, A is a complex sequence of degree 3.

In contrast to a large body of work on engineering applications of sequence recognition and generation, we are specifically interested in neural architectures that can easily learn and recall temporal sequences based on associative networks and Hebbian learning. Networks that are able to learn and recall sequences rely on some form of hetero-associations between different patterns in a sequence in addition to standard auto-associative

point-attractor networks. Such networks have long been proposed and studied by Grossberg [3,4]. Hopfield [6] noted the difficulty in recalling sequences of more than four items, and subsequent research has focused on additional mechanisms to stabilize hetero-associative sequential memories [16,8,13,5].

In this paper we discuss a modular neural architecture that is able to learn and robustly recall long temporal sequences. The architecture consists of two coupled recurrent networks with auto- and hetero-associative connections. Similar networks have been studied by Jensen and Lisman [7] to model memory functions in a cortico-hippocampal network, and later to model the hippocampus in more detail [10]. The networks studied by Jensen and Lisman are based on the coupling of an auto-associative network with a hetero-associative network, whereas we consider here a network consisting of two auto-associative networks that are coupled through hetero-associative connections.

A related approach to sequence learning and recognition is discussed in the literature under the name of synfire chains [1,2,9]. In contrast to the approaches that grew out of the auto-association literature, this theory is based on pure hetero-associative connections in (recurrent) diverging/converging chains [1]. However, a bidirectional

*Corresponding author. Fax: +1 902 492 1517.
E-mail address: tt@cs.dal.ca (T. Trappenberg).

modular model very similar to the model studied here has recently been studied by Sommer and Wennerkers [12]. This model couples two auto-associative modules with hetero-associative intra-modular connection, whereas we found it beneficial to consider only one intra-modular pathway to be hetero-associative. While synfire chains are often discussed with neocortical processing [2], it has also been implicated with Hippocampal functions [19].

While we compare the performance of sequence learning and recall of an architecture used by Jensen and Lisman with some modified versions, it is important to note that the models by Jensen and Lisman, as well as related hippocampal models by others [17], consider sequences of items that are time compressed through a phase precession mechanism or a related mechanism based on multiplexing with gamma oscillations. Such sequences are limited to a relatively small number of items (in the order of 4–7) or are limited in their timescale to events occurring on the order of seconds. Such memories of short duration sequences are not considered in this study. Instead, we consider here the storage and recall of long sequences with hetero-associations that can bind memories of arbitrary time scales.

The rest of this paper is organized as follows: Section 2 reviews associative memory models of sequence memory based on auto- and hetero-associations. Section 3 describes the modular architecture and the two different models studied in this paper. In Section 4, the models are compared experimentally with simulations to solidify the intuition behind their operation. Conclusions and suggestions for future research direction are given in Section 5.

2. Associative sequence memory

We consider in this paper associative memories of leaky-integrator nodes governed by the dynamics

$$\tau \frac{du_i(t)}{dt} = -u_i + \sum_j w_{ij} r_j(t) \quad (1)$$

on a time scale of τ , where $u_i(t)$ is the internal state of node i at time t . r_i is the corresponding firing rate related to u_i through a gain function g ,

$$r_i(t) = g(u_i(t)). \quad (2)$$

The weights between node j and node i are trained with Hebbian learning rules. In such rules, the weights are incremented proportional to the activity of the framing nodes

$$\Delta w_{ij} \propto (r_i - \langle r_i \rangle)(r_j - \langle r_j \rangle), \quad (3)$$

where $\langle \cdot \rangle$ indicates mean node activity. In the following, we simplify the notation by assuming rate representations with zero mean and learning procedures which clamp the network state to the training patterns. We distinguish between (symmetric) auto-associative weights

$$w_{ij}^A = \frac{1}{N} \sum_{\mu} \xi_i^{\mu} \xi_j^{\mu} \quad (4)$$

and (asymmetric) hetero-associative weights

$$w_{ij}^H = \frac{1}{N} \sum_{\mu} \xi_i^{\mu+1} \xi_j^{\mu}, \quad (5)$$

where N is the number of nodes in the network, and ξ_i^{μ} is the i th component of the μ th pattern of a sequence of p patterns ($\mu = 1, \dots, p$). These rules are consistent with Eq. (3) for zero initial weights and an appropriate choice of learning rates and training repetitions.

Another form of learning sequence relations, which generalizes the basic form of Eq. (5), is Hebbian trace learning. In this approach a temporal average (trace) of node activity,

$$\bar{r}_j(t) = \int_0^{\infty} G(x) r_j(t-x) dx, \quad (6)$$

is used in the learning rule, for example

$$\Delta w_{ij}^H \propto \bar{r}_i r_j. \quad (7)$$

G is a kernel function which specifies how the firing history is to be sampled, and has the property that $\int_0^{\infty} G(x) dx = 1$. For example, the delta function, which is 0 for all x except for an infinite peak of area 1 at $x = k$, where k is the presentation time of the previous pattern, leads to the training rule of Eq. (5). This could be implemented in the brain by slow synapses. In contrast, fast synapses would lead to an auto-associative rule similar to Eq. (4) (see [7] for a discussion of the physiology underlying these mechanisms).

Both auto-associative and hetero-associative connections are necessary to achieve robust sequence memory. This conjecture is based on the argument that the auto-associative weights are important for pattern completion that enables the recall of items with partial cues in noisy environments, whereas the hetero-associations drive the state of the system from one pattern to the next. For example, Hopfield [6] considered a model

$$\tau \frac{du_i(t)}{dt} = -u_i + \sum_j w_{ij}^A r_j(t) + \lambda \sum_j w_{ij}^H r_j(t). \quad (8)$$

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 the initiation state $\mathbf{r}(0)$. As λ gets larger the network tends to attract 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 4 [6,11].

To solve this problem it is sufficient to allow some delay between the time a new state is entered and when a transition from that state to the next is induced. This can

be achieved using an update rule

$$\tau \frac{du_i(t)}{dt} = -u_i + \sum_j w_{ij}^A r_j(t) + \lambda \sum_j w_{ij}^H \bar{r}_j(t), \quad (9)$$

where the quantity $\bar{r}_j(t)$ is again a convolution of the firing history as noted in Eq. (6), although this time the trace is used during recall and not during learning as in Eq. (7). Kleinfeld [8] uses a delta function $\delta(x-k)$, representing slow hetero-associative synapses with a delay of k . Tank and Hopfield [16] use an exponential kernel function similar to an alpha function with a peak at $x=k$ for some $k>0$ causing the firing rate history at time $t-k$ to be strongest, representing a combination of short-term memory effects and delayed synapses.

Another related approach is to replace λ in Eq. (1) with $\lambda(t)$, allowing the strength of the hetero-associative synapses to vary explicitly over time. In this way strong hetero-association can be used to initiate a transition to the next pattern in the sequence, followed by strong auto-association for developing a clear representation of that pattern. Additional functionality over the kernel function approach is that it allows adjustment of the duration for which each pattern in the sequence is present. This is the approach used by Wang in [18], who alternates between phases of hetero-association and auto-association in the connections between a pair of recurrent networks in order to accurately recall a simple sequence of pairs of patterns. In the brain, a time-dependent λ may represent the effect of modulatory synapses from another population of neurons which acts as a controller during the recall process. However, the activity of this population must be carefully considered and modelled if the goal is to explain sequence recall in animals—one cannot simply choose any $\lambda(t)$ which works without considering how the dynamics might be achieved in the brain.

Hetero-associative continuous attractor networks have been studied by Stringer et al. to solve path-integration [15] and in conjunction with learning motor primitives and motor control [14]. During learning, these models associate the movement of the activity packet in the continuous attractor network over movement-indicator cells corresponding to the lambda values in the formulations above. This is an example where the hetero-associative weights are modulated by the firing of other neuronal groups. Paper [14] gives an example of using this strategy in a modular network and also demonstrates that complex sequences can be handled in this approach.

3. The models

We study a modular architecture of two connected recurrent networks as shown in Fig. 1. The modules are labelled A and B, respectively. The activities of nodes in modules A and B are governed by leaky-integrator

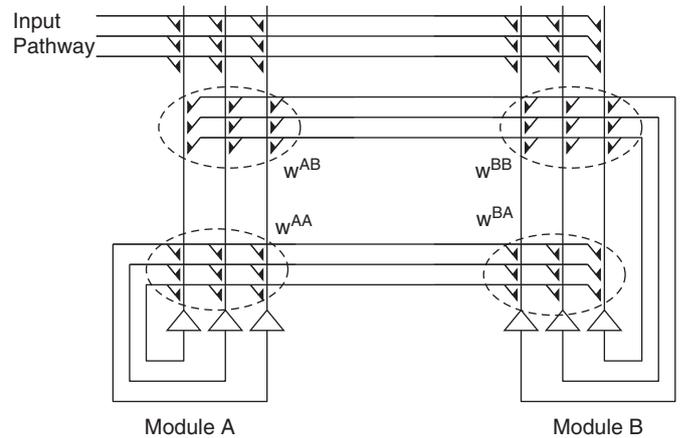


Fig. 1. The proposed network, a pair of connected cortical modules A and B. w^{AA} , w^{BB} , w^{AB} and w^{BA} represent the synaptic strengths in the recurrent connections in modules A and B, and the inter-module connections from B to A and from A to B, respectively. Lisman's description [10] is similarly pictured with w^{BB} trained hetero-associatively (Eq. (5)), and all other weights set auto-associatively (Eq. (4)). In our model w^{AB} is trained hetero-associatively (Eq. (5)), and all other weights auto-associatively (Eq. (4)).

dynamics

$$\tau \frac{dh_i^A(t)}{dt} = -h_i^A(t) + \lambda^{AA} \sum_{j=1}^N w_{ij}^{AA} r_j^A(t) + \lambda^{AB} \sum_{j=1}^N w_{ij}^{AB} r_j^B(t), \quad (10)$$

$$\tau \frac{dh_i^B(t)}{dt} = -h_i^B(t) + \lambda^{BB} \sum_{j=1}^N w_{ij}^{BB} r_j^B(t) + \lambda^{BA} \sum_{j=1}^N w_{ij}^{BA} r_j^A(t), \quad (11)$$

$$r_i^A(t) = \tanh(h_i^A(t)), \quad (12)$$

$$r_i^B(t) = \tanh(h_i^B(t)). \quad (13)$$

Within this architecture we consider two principal models with different placements of auto- and hetero-associative weight matrices.

Model 1: All weights are auto-associative except the weights in module B which are hetero-associative. This model corresponds to an auto-associative coupling of an auto-associative memory with a hetero-associative memory.

Model 2: All weights are auto-associative except the weights from module B to module A which are hetero-associative. Thus, model 2 couples two auto-associative memories with hetero-associations in one direction and auto-associations in the other direction.

Model 1 is reminiscent of the cortico-hippocampal architecture of Jensen and Lisman [7] and the hippocampal model of Lisman [10]. For example, in [10] module A

corresponds to a recurrent network formed by dentate neurons contacting mossy cells that feed back to dentate neurons, while module B would correspond to a CA3 network. Lisman states that the process of sequence recall would work as follows. Upon external stimulation to the first pattern ξ^1 in the sequence, the recurrent hetero-associative connections in the CA3 would tend to make it converge towards a noisy version $\xi^{2'}$ of the next pattern ξ^2 in the sequence. The connections from the CA3 would tend to excite the pattern $\xi^{2'}$ in the dentate, which would then use auto-association to reduce the noise and produce ξ^2 . The pattern ξ^2 would then be transmitted back to the CA3, which would again tend to move towards the next pattern in the sequence using its hetero-associative connections.

The intuition of model 2 during recall is that each module works somewhat independently, cleaning up its own representation of the current pattern in the sequence ξ^μ . When module B has a “clean enough” representation of ξ^μ , it can begin to push module A 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:

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

The temporal behaviour is a consequence of the growth of these forces and their necessary alternation, in that, when A’s strength is at a maximum B should be in a transition and hence its strength is minimized, and vice versa for when B’s strength is at a maximum. For example in case 1 above, the strength of B pushing A toward $\xi^{\mu+1}$ is related to how clearly ξ^μ is represented in B, i.e. how close $\mathbf{r}^B(t)$, the firing rates of nodes in module B at time t , are to ξ^μ . As module B draws closer to ξ^μ , eventually its strength in moving A toward $\xi^{\mu+1}$ becomes large enough to push A into the basin of attraction for pattern $\xi^{\mu+1}$. At this point A has a small strength in affecting B as it is far away from, but moving toward $\xi^{\mu+1}$. As A draws closer to this attractor its strength in pushing B toward $\xi^{\mu+1}$ increases until this occurs, at which point B’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 B are sufficient to affect the course of module A when B 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, B has more effect on

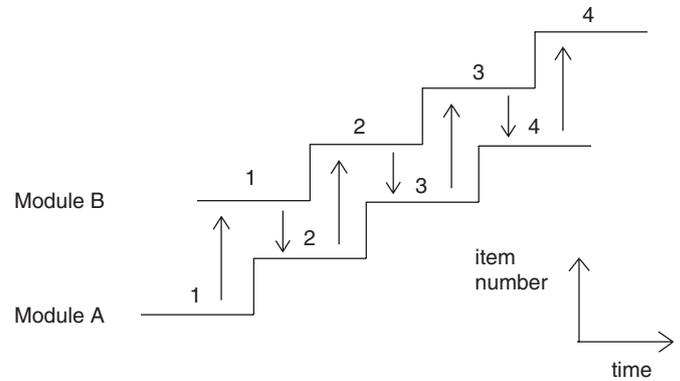


Fig. 2. Learning of intra-modular connections in model 2. A sequence of four patterns is presented to both modules. The sequence of module B is delayed with respect to module A. Learning of connections from A to B and B to A are interleaved, resulting in auto-associative connections from A to B (long arrows) and hetero-associations from B to A (short arrows).

A (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. Similarly, weaker inter-module strengths will result in slower transitions between patterns, or none at all when set too weak.

Although the placement of different connection types is easy to implement in simulations, it is important to note that their biological realization and interpretation is somewhat different in models 1 and 2. In model 1 we assume that during learning the same pattern is presented simultaneously to both modules and that the time course of plasticity is sufficiently fast in most pathways to result in auto-associative learning. Only the plasticity of intra-modular connections in module B need to be based on ion channels with receptors that have a slow activation kinetics (like NMDAR in CA3, see discussions in [7]) to enable hetero-associative learning in the same module. This is different in model 2 where hetero-associative learning between modules does not require receptors that have a slow activation kinetics. For example, it is possible to present the time series of patterns with a fixed offset in the different modules as shown in Fig. 2 which only requires a fixed delay in the external pathways to the different modules. With a further assumption that intra-modular plasticity is modulated to specific time windows as illustrated by the arrows in Fig. 2 we arrive naturally at model 2.

4. Experimental results

All models were tested with a basic sequence recall experiment in which we trained the network with a sequence of p patterns and tried to adjust the relative strength of the different pathways (λ s) to enable correct recall of the sequence. We also verified the stability of sequence recall by using a noisy initial pattern and noisy transmission, and explored the limits of the sequence length when stable solutions were found. Each pattern ξ^μ is

a string of binary numbers representing node activities, i.e. $\xi^\mu \in \{-1, 1\}^N$ where N is the number of nodes. The patterns are uniformly distributed so that the mean node activity is zero, allowing us to use the simplified learning rules of Eqs. (4 and 5). In our hetero-associative training we associate ξ^1 from ξ^p , so that the sequence is cyclic. Sequence recall is illustrated in the following by plotting the inner product between the network states and the training patterns.

4.1. Model 1

With model 1 we were not able to find appropriate parameters to get consistent results. An example of sequence recall using model 1 showing some transitions is shown in Fig. 3. After extensive experimentation, this is an example of the best progress towards a correctly recalled sequence that we were able to achieve. Modules of $N = 1000$ nodes were used, and the model was trained on short a sequence of only six patterns. Experiments on longer sequences failed similarly. Both modules were initialized to the first pattern without noise. Initializing the modules with different patterns increased the difficulty in finding satisfactory λ s. Decreasing the number of nodes and patterns adds some stability to the sequences, but this may be an artefact of small networks. We found that typically module B either attracts quickly to some stable fixed point which does not represent any of the stored patterns, or as λ^{BB} is increased, experiences very rapid oscillations. module A meanwhile usually attracts to a fixed

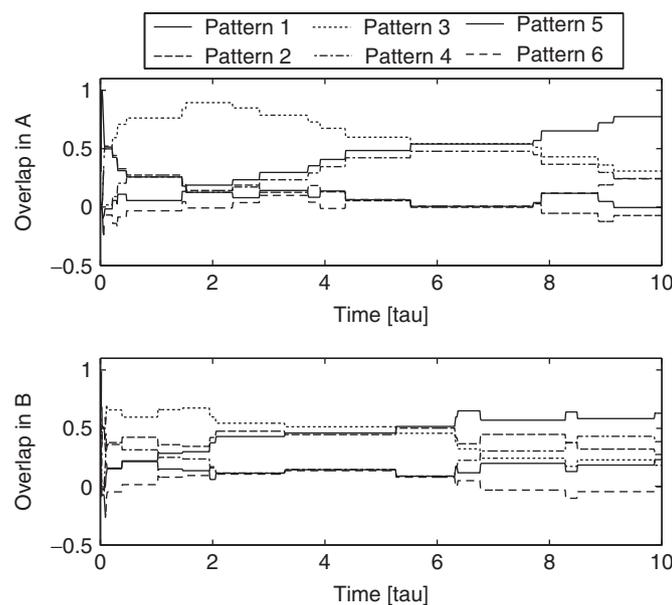


Fig. 3. Result of recall of a learned sequence of six random patterns using model 1 with $N = 1000$ nodes. The parameters used were $\lambda^{AA} = 1$, $\lambda^{BB} = 2.2$, $\lambda^{BA} = 2$, $\lambda^{AB} = 4$. The simulation was performed in the same way as that of Fig. 4. Both modules were initialized with the first pattern without noise. The result shows the pattern oscillations of module B due to its hetero-associative synapses (bottom), and the oscillations of module A (top) due to (selective) interaction from B.

point, or will make transitions if B is oscillating and λ^{AB} is large enough. However, A does not ever converge to a pattern in this case, and its transitions do not recall the patterns in order.

The problem with tuning the λ s is as follows: increasing λ^{AB} will give a greater chance that A can be brought into the basin of attraction of a particular pattern in order to clean up its representation of this pattern, but as B continues to oscillate this brings a greater chance that A will be disrupted from its process of cleaning up this pattern. Thus the network does not quite behave according to the intuitive notion of stages of hetero-association followed by cleaning of the signal, since the activities of B are not stable enough to affect A in a predictable manner.

These results reflect the known difficulties of previous simulations with single module networks [6]. These networks have separate hetero- and auto-associative connections but constrain the response to input from these connections by having only one class of neurons. The results here show that removing this constraint does not improve the recognition ability of the network. Also, while model 1 reflects the principal architecture of the models used by Jensen and Lisman [7,10], it is important to note that the implementations in their work include synaptic dynamics that can stabilize sequence recall in such networks as discussed in Section 3.

4.2. Model 2

Fig. 4 shows the result of recall of a sequence of 20 random patterns using model 2 with 1000 nodes in each module. It is easy to find strength parameters which lead to reliable sequence recall. The values for the example shown in Fig. 4 are $\lambda^{BB} = \lambda^{AA} = \lambda^{BA} = 1$, $\lambda^{AB} = 2$. The modules were initialized with a noisy version of pattern ξ^1 in module A and a random pattern module B. The overlap between the network states and the stored patterns, $\mathbf{r}^A(t)' \xi^\mu / N$ and $\mathbf{r}^B(t)' \xi^\mu / N$ was measured, but is only shown for module A (top figure) since the overlap of module B is very similar, being only a slightly shifted version of the plot of module A. It can be seen from Fig. 4(A) that each pattern is perfectly recalled and is stable for some short period. Since it is difficult to distinguish between the 20 different patterns in Fig. 4(A) even if different line styles were used, the index μ of the pattern for which $\mathbf{r}^A(t)' \xi^\mu / N$ is largest is plotted for each time in Fig. 4(B), where it can be seen that the patterns are indeed recalled in the correct order. It was also possible to get a correct and stable sequence recall with 50 patterns in a network with 2000 nodes in each module (Fig. 4(C)).

Since synaptic transmission is not perfect, and nodes may fire with less input than normal, or may not fire when given enough input, we also conducted experiments where the transmission of signals is noisy. This was done by negating a certain percentage of the r_i values in Eq. (10) at each time step during recall. We found that reliable recall of 20 random patterns in a network of $N = 1000$ nodes can

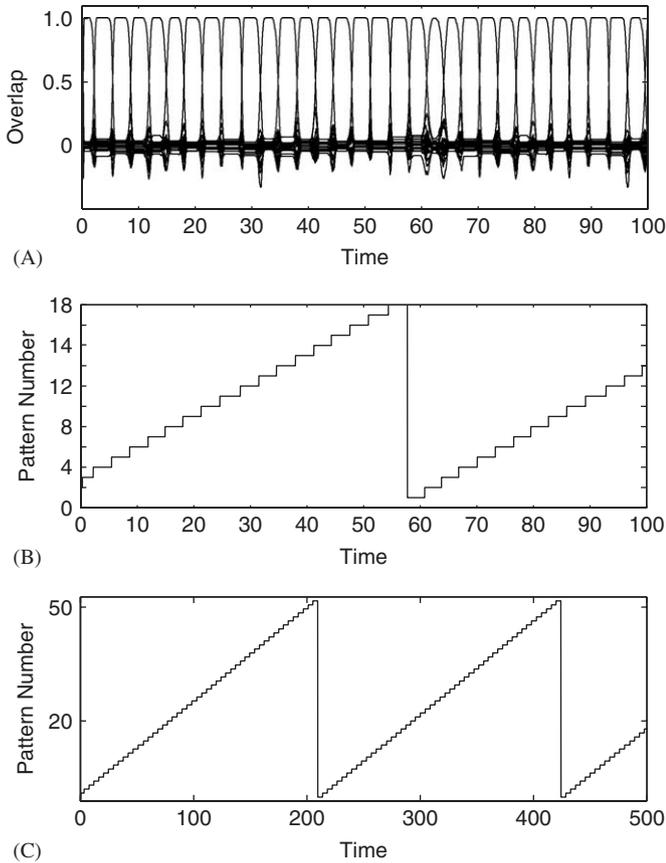


Fig. 4. (A + B) Result of recall of a learned sequence of 20 random patterns using model 1 with $N = 1000$ nodes. The inter-module synapses from A to B were given a weight of $\lambda^{BA} = 1$, the same as the intra-module (recurrent) synapses, and the inter-module synapses from B to A were given a weight of $\lambda^{AB} = 2$ times larger. Sequence recall was initiated by setting the firing rates of nodes in module A to the first pattern with 30% noise, and the firing rates of nodes in module B to random values. (A) The overlap between the firing rates $r^A(t)$ of the nodes in module A and each of the stored patterns at each time. (B) The pattern which has the highest overlap with the firing rates of module A at each time. (C) Plot similar to (B) for a network of 2000 nodes in each module that is trained on a sequence of 50 patterns.

be performed with up to 30% noise. With 40% noise perfect recall sometimes occurs, but is often disrupted. The network still performs sequence recall to a degree, but does not recall any pattern perfectly and often jumps between different segments of the sequence.

Fig. 5 shows how the speed of recall can be increased by increasing the intra-module synaptic strengths λ^{AB} and λ^{BA} . The top figure shows the recall of a sequence of 10 patterns with the same parameters as in Fig. 4, while in the bottom figure λ^{BA} is increased from 1 to 1.6, and λ^{AB} is increased from 2 to 2.1. Increasing λ^{AB} any further causes A to be preemptively disrupted from its auto-associative recall of pattern ξ^μ for recall of pattern $\xi^{\mu+1}$. The patterns are still recalled in order, but each pattern is not fully recalled. Increasing all λ s has little effect on the sequence recall, it is the difference in strength between the inter- and intra-module connections which is the major factor in recall speed.

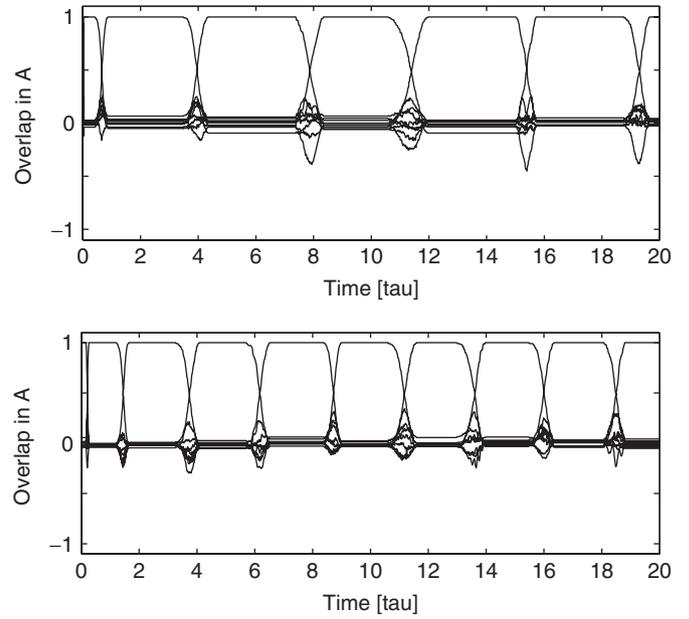


Fig. 5. Overlap of the firing rates in module A with each of pattern during recall of a length ten sequence. $N = 1000$ nodes were used in both cases, with $\lambda^{BB} = \lambda^{AA} = 1$ and (top) $\lambda^{AB} = 2$, $\lambda^{BA} = 1$ and (bottom) $\lambda^{AB} = 2.1$, $\lambda^{BA} = 1.6$. The period for which each pattern is stable is decreased with higher inter-module synaptic strengths.

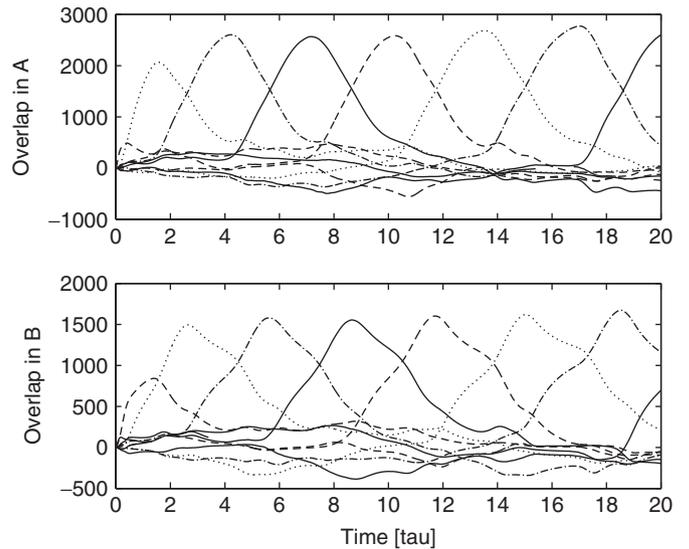


Fig. 6. Overlap of the net activation $\mathbf{h}(t)$ of nodes in each module with the stored patterns in a sequence of length 10 during recall. Here the dynamics of the interaction between the modules A and B can clearly be seen. As the activation overlap rises for a particular pattern in B (bottom), A begins moving toward the next pattern in the sequence (top) due to the B→A hetero-associations. As the activation overlap rises for a particular pattern in A, B follows shortly behind from the A→B auto-associations. Note that because there are only four available line styles, some patterns (not adjacent in the sequence) share the same line style.

Fig. 6 shows the overlap of the net activation ($\mathbf{h}^A(t)$ and $\mathbf{h}^B(t)$) of nodes in both modules with the patterns in a sequence of length 10 during recall. Here the interaction between the two modules can clearly be seen. For example

at time $\tau = 2$, B is attracting towards the pattern indicated by the dotted line, causing A to move away from this pattern and attract to the next pattern in the sequence, indicated by the dot-dashed line. As this happens B follows closely behind, causing A to once again transition to the next pattern in the sequence.

5. Conclusions and future work

Multi-modular approaches to sequence generation have the property that the timing of transitions between items in a sequence is a product of the dynamics of the interaction between different modules. This is in contrast to approaches involving a single recurrent network, which require an explicit temporal function which must be chosen ahead of time. Recall speed in a single network can be changed by adjusting the parameters of the chosen temporal functions, which implies an adjustability of the synaptic activation dynamics. The speed of recall in the modular approach studied here can be altered by modulating the overall strength between the modules.

The model presented in this paper that has hetero-associative connections between the auto-associative modules is able to learn and generate simple sequences of considerable length easily and reliably, even in the presence of noise. This is in contrast to an architecture that connects a hetero-associative module with an auto-associative module. Our architecture permits storage of simple sequences with arbitrary and varying time scales; it does not require combining a trace of previous activity in a recurrent network with new activity, as the pattern in different networks can be timed differently and learning occurs between these patterns.

In this paper we only consider simple sequences, but we explored some extension of the models for complex sequences. For example, in one extension we implemented a trace rule as in Eq. (6) using an exponentially weighted moving average, i.e. our hetero-association in Eq. (5) becomes

$$w_{ij}^{AB} = \frac{1}{N} \sum_{\mu} \xi_i^{\mu+1} \bar{\xi}_j^{\mu},$$

with $\bar{\xi}_j^{\mu}$ being the exponentially weighted moving average of all previous activity of node j

$$\bar{\xi}_j^{\mu} = (1 - \eta) \xi_j^{\mu} + \eta \bar{\xi}_j^{\mu-1}. \quad (14)$$

During simulation we then compute a trace of previous node activity

$$\bar{r}_j^B(t) = (1 - \gamma) r_j^B(t) + \gamma \bar{r}_j^B(t - dt)$$

as activity of module B in Eq. (10), where dt is the time step of the simulation. However, we were not able to find values for γ and η to recall sequences when different patterns in the sequence were more than 70% similar.

Another approach that we explored was to store with each pattern some information which amounts to the

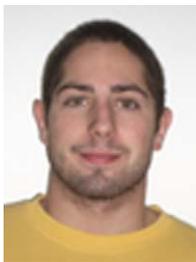
“context” of that pattern. For example if the firing rates of a small group of N_P nodes represented the actual pattern, and the other $N_C = N - N_P$ represented the context, complex sequence recall should be possible if the different contexts of a repeated pattern are unique. However, in experiments with $N = 1000$ nodes we were only able to recall complex sequence in this approach for considerable large context such as $N_C = 800$ nodes. Thus, reliable storage and recall of complex sequences is still a major challenge. Another direction for further research is the learning of sequences whose items do not occur at regular intervals.

The types of sequences this paper is concerned with are of considerable length (larger than 7) and have variable time frames. This is in contrast to the sequence memory discussed by Jensen and Lisman and others in conjunction with hippocampal functions. Sequence learning has also been discussed in conjunction with the basal ganglia in the framework of reward and temporal difference learning [20]. It is likely that several mechanisms and different roles of temporal sequences processing are utilized in the brain, and it is necessary to distinguish these different forms of sequence processing and their mechanistic realization in more detail.

References

- [1] M. Abeles, *Corticonics: Neural Circuits of the Cerebral Cortex*, Cambridge University Press, Cambridge, UK, 1991.
- [2] E. Bienenstock, A model of neocortex, *Network: Comput. Neural Syst.* 6 (1995) 179–224.
- [3] S. Grossberg, Some physiological and biochemical consequences of psychological postulates, *Proc. Nat. Acad. Sci. USA* 60 (3) (1968) 758–765.
- [4] S. Grossberg, Some networks that can learn, remember and reproduce any number of complicated space–time patterns, *J. Math. Mech.* 19 (1) (1969) 53–91.
- [5] H. Gutfreund, M. Mezard, Processing of temporal sequences in neural networks, *Phys. Rev. Lett.* 61 (1988) 235–238.
- [6] J.J. Hopfield, Neural networks and physical systems with emergent collective computational abilities, *Proc. Nat. Acad. Sci.* 79 (1982) 2554–2558.
- [7] O. Jensen, J.E. Lisman, Theta/gamma networks with slow NMDA channels learn sequences and encode episodic memory: role of NMDA channels, *Learning and Memory* 3 (1996) 264–278.
- [8] D. Kleinfeld, Sequential state generation by model neural networks, *Proc. Nat. Acad. Sci.* 83 (1986) 9469–9473.
- [9] W. Levy, A computational approach to hippocampal function, in: *Computational Models of Learning in Simple Neural Systems*, Academic Press, New York, 1989, pp. 243–305.
- [10] J. Lisman, Relating hippocampal circuitry to function: recall of memory sequences by reciprocal dentate—CA3 interactions, *Neuron* 22 (2) (1999) 233–242.
- [11] H. Nishimori, T. Nakamura, M. Shiino, Retrieval of spatio-temporal sequence in asynchronous neural network, *Phys. Rev. A* 41 (1990) 3346–3354.
- [12] F.T. Sommer, T. Wennekers, Synfire chains with conductance-based neurons: internal timing and coordination with timed input, *Neurocomputing* 65–66 (2005) 449–454.
- [13] H. Sompolinsky, I. Kanter, Temporal association in asymmetric neural networks, *Phys. Rev. Lett.* 57 (1986) 2861–2864.

- [14] S.M. Stringer, E.T. Rolls, T.P. Trappenberg, I.E.T. de Araujo, Self-organizing continuous attractor networks and motor function, *Neural Networks* 16 (2) (2003) 161–182.
- [15] S.M. Stringer, T.P. Trappenberg, E.T. Rolls, I.E.T. de Araujo, Self-organising continuous attractor networks and path integration: one-dimensional models of head direction cells, *Network: Comput. Neural Syst.* 13 (2002) 217–242.
- [16] D.W. Tank, J.J. Hopfield, Neural computation by concentrating information in time, *Proc. Nat. Acad. Sci.* 84 (1987) 1896–1900.
- [17] H. Wagatsuma, Y. Yamaguchi, Cognitive map formation through sequence encoding by theta phase precession, *Neural Comput.* 16 (12) (2004) 2665–2697.
- [18] L. Wang, Heteroassociations of spatio-temporal sequences with the bidirectional associative memory, *IEEE Trans. Neural Networks* 11 (6) (2000) 1503–1505.
- [19] A.B.H. William, B. Levy, X. Wu, Interpreting hippocampal function as recoding and forecasting, *Neural Networks* 18 (2005) 1242–1264.
- [20] F. Wörgötter, B. Porr, Temporal sequence learning, prediction, and control: a review of different models and their relation to biological mechanisms, *Neural Comput.* 17 (2005) 245–319.



Michael Lawrence received a BCSc degree from Dalhousie University in Halifax, Canada in 2004. He is currently an NSERC Canada Graduate Scholar working towards a MSc degree from Dalhousie University. His research interests include approximation algorithms and combinatorial optimization, data warehousing, on-line analytical processing and distributed computing.



Thomas Trappenberg studied physics at RWTH Aachen University and held research positions at KFA/HLRZ Jülich, the RIKEN Brain Science Institute and Oxford University. He is currently associate professor in the Faculty of Computer Science at Dalhousie University. His principal research interests are computational neuroscience and applications of machine learning methods for data classification and signal analysis. He is the author of the textbook 'Fundamentals of Computational Neuroscience' published by Oxford University Press.



Alan Fine studied philosophy and biology at Harvard University, and obtained a veterinary MD and Ph.D. in physiology at the University of Pennsylvania. He is Team Leader in the Division of Neurophysiology at the National Institute for Medical Research in London, and Professor of Physiology and Biophysics at Dalhousie University. His research deals with mechanisms of synaptic transmission and learning.