

A Cell Assembly Model of Sequential Memory

Hina Ghalib, Christian Huyck

Abstract—Perception, prediction and generation of sequences is a fundamental aspect of human behavior and depends on the ability to detect serial order. This paper presents a plausible model of sequential memory at the neurological level based on the theory of cell assemblies. The basic idea is that sequences in the brain are represented by cell assemblies. Each item of the sequence and the sequential association between the items are represented by cell assemblies. Simulation results show that the model is capable of recognizing and discriminating multiple sequences stored in memory. The cell assemblies that represent the sequential association between two items are activated if these items occur in the input in the correct order. These sequence detecting cell assemblies form the basis of this model. A simulation presenting 100 stored sequences and 100 not stored recognizes perfectly 90% of the time with no false positives.

INTRODUCTION

THE ability to process sequential information is important for intelligent behavior produced by natural and artificial systems. For many years researchers have been studying various models to explain sequential learning and memory, but the neural dynamics are still not well understood. This paper presents a neurally plausible mechanism for sequential memory. It addresses the question of how to store a sequence of discrete elements and recall them in correct temporal order.

The concept of cell assemblies provides a reasonable account of the neural representation of basic concepts in the human brain [1]. However, the organization of neural structures that result in higher cognitive processes such as sequence recognition is not known yet. In this paper, a fatiguing leaky integrate-and-fire neuron model is used as the basis. Cell assemblies that emerge from these simulated neurons form a neurally plausible mechanism for storing and retrieving temporal sequences. Two different types of cell assemblies are simulated: base cell assemblies, that represent the item of the sequence, and sequence cell assemblies that represent different sequences.

BACKGROUND

A great deal of research has been directed towards the problem of understanding psychological phenomena from a neural perspective. The cell assembly hypothesis is central

to much of this research. Specifically there has been a lot of interest in understanding sequences and the neural basis of sequences. We will first review the cell assembly hypothesis and then propose how dynamics of neuronal assemblies can account for complex sequential behavior.

Cell Assembly

Hebb proposed that synapses are modified by an activity dependant mechanism [1]. The synaptic modification rule proposed by Hebb facilitates the storage of information in human memory. He suggested that neurons do not individually represent memories but a large group of simultaneously active neurons account for any concept or idea in the human brain. Central to his theory is the idea of cell assemblies, which are formed as a result of the proposed learning process. Learning is achieved by increasing the synaptic strength between any two neurons in the network if the activation of the two neurons is correlated and decreasing it if they are activated in an uncorrelated manner. Therefore when an external input is presented to the network, the neurons that are simultaneously active acquire connections with each other. Repeated presentations of the external input increase the strength of the connections between these neurons. The increase in the synaptic strength allows this group of neurons to function together. This group of strongly connected neurons is called a cell assembly. Formation of these cell assemblies accounts for long term memory. Whenever a large subset of the cell assembly is activated by external input, all the neurons of the cell assembly are activated due to recurrent excitatory connections. The strong recurrent excitatory connections enable activation to reverberate among the neurons of the group even after the external input is removed. Such persistent neural activity in a cell assembly, even after the external input has ceased, accounts for short term memory. Hebb's theory and associated learning rule have inspired several computational neural network models (eg. [2-16]).

The cell assembly theory successfully explains how the brain learns neural representations of different concepts and how they are stored in the human memory. These concepts enable us to recognize different objects, names and events. This paper is an attempt to extend this theory to explain more complex human behavior than perception. Sequence learning is one such aspect of behavior and learning serial order association between two items of a sequence is crucial for executing a wide range of cognitive tasks.

Sequence Learning

For many decades, researchers have investigated the organization of sequences in memory. One of the earliest

Manuscript received January 31, 2007. This work was supported in part by EPSRC grant EP/DO59720

Hina Ghalib, is a student at the Middlesex University, London, NW4 4BE, UK. (e-mail: h.ghalib@mdx.ac.uk).

Christian Huyck is with the Computing Science Department of Middlesex University, London, NW4 4BE, UK. (phone: +44 (0)20 8411 5412 ; fax: +44 (0)20 8411 6943 ; e-mail: c.huyck@mdx.ac.uk).

accounts of serial order memory was given by Ebbinghaus [17]. He believed that serially organized behavior can be represented as a chain of associations. According to his view, sequences were learned by associating each item or memory pattern of the sequence with the item that follows it in a sequence. So a sequence ABCD is simply stored as a set of associations A-B, B-C, C-D. In this paradigm, each item of the sequence when recalled remains active for some time after which it facilitates the activation of the the next item in the sequence to which it is associated.

Hebb suggests cell assemblies become organized into more complex systems. As repeated activation of ideas result in the formation of cell assemblies, ideas that are frequently activated contiguously become associated by establishing direct synaptic connections between their cell assemblies. This leads to sequential activation of cell assemblies and results in the organization of inter-related cell assemblies into a phase sequence. Stimulation of a phase sequence represents successive steps of serial computation at the neural level. A higher level organization of phase sequences in a series constitutes a phase cycle.

A number of neural network models of serial order were based on the principle of forming associations between successive items of a sequence [18-21]. These models based on the idea of associative chaining were not capable of handling sequences containing repeated items. Consider a sequence R-E-S-E-A-R-C-H, where there are two associative links arising from E. Associative chaining models provide no mechanism for choosing between multiple associative links. Therefore sequences with recurring patterns cannot be learned by direct association between items. In the cell assembly model each cell assembly may be associated with a number of other cell assemblies, but simple and direct association between cell assemblies, like other associative chaining models, fail to account for memory of complex sequences. Lashley [22] rejected simple pair wise associative chaining as a plausible mechanism to account for complex serial behavior.

The problem of sequence learning has been addressed using various techniques, such as hidden Markov models, dynamic programming, connectionist systems and evolutionary computation. In order to deal with sequential information, the system must be designed such that several input patterns together in their given order influence the output of the system. The system therefore needs some form of memory to hold past inputs. Many recent simulations have used connectionist systems to model how the brain might process sequential information. These connectionist models either use recurrent connections or some other non recurrent mechanism such as time-delay or windows to capture temporal features of the input. All these mechanism try to incorporate some kind of contextual information in a connectionist structure. The Time Delay Neural Network is an example of a feed-forward network where hidden neurons and output neurons are replicated across time to process

temporal information [23]. Finite Impulse Response (FIR) is another feed-forward network which models each synapse as a linear FIR filter to provide dynamic interconnectivity [24]. Networks that use feedback connections to store representations of recent input events are broadly termed as recurrent networks. A large number of connectionist models capable of processing sequential information are based on a feedback architecture [25-29]. Temporal back-propagation is a successful learning mechanism with these recurrent networks but it lacks biological plausibility as it is a supervised mechanism. Another class of popular connectionist system that is used for sequence processing are based on the Hopfield model of associative memory [30-34]. All these models consider a temporal sequence as an association of contiguous items, and the network learns these associations in a supervised manner.

A number of unsupervised connectionist models have also been presented. Some of these models based on self organizing maps [35], such as Temporal Kohonen Map [36-37] and Recurrent Self-Organizing Map [38], process temporal information by incorporating leaky integrator memory to preserve the temporal context of the input.

A number of computational models can successfully replay the sequence of items just presented. These models only account for short-term memory of sequences but none of them addresses the question of how these sequences are learned in an autonomous fashion and consolidated into long-term memory [39-41].

This paper presents a model that overcomes the problems faced by associative chaining models and their variants. The model presented does not store serial order relationship in the connections between the cell assemblies of the associated elements. This serial order relationship is extracted and translated into a more stable representation that is itself a cell assembly. In this paper, this third cell assembly is called a sequence cell assembly and it represents context information when processing sequences. The sequence cell assembly stores serial order associations in long term memory. Each sequence cell assembly receives connections from a cell assembly representing the context and the last element of the sequence. It is sensitive to the order in which associated items are presented. The model is then used for the problem of sequence recognition.

SIMULATION MODEL

This section presents the model that is simulated. Simulations described in this paper are based on fatiguing leaky integrate-and-fire neurons with dynamic threshold as discussed below. Neurons are organized into two types of networks, a base network and four sequence networks. The connection scheme within a network differs from the connection scheme between networks.

The Neuron Model

A biologically plausible neuron model is used [42]. The

system presented is a sparsely connected network of fatiguing leaky integrate-and-fire neurons. The network is composed of neurons, 80% of which are excitatory and 20% inhibitory as in the mammalian cortex [43]. The neurons in the network are either excitatory or inhibitory; that is, all synapses leaving a neuron are either excitatory, or they are all inhibitory. This is consistent with Dale's principle [44]. Connections are uni-directional as in the brain. Neurons do not have self connections.

Each neuron receives activations from all neurons that are connected to it. The activation received over time is integrated. The neuron leaks some activation at each step that it does not fire, modeling the leakage of current through the cell membrane of biological neurons [45]. Activation is lost at a constant rate which is called the decay rate. Activation of a neuron i at time step t is represented by the following equation:

$$a_i(t) = a_i(t-1) \times \tau + \sum w_{ij} \times o_j(t-1) \quad (1)$$

Where $a_i(t)$ and $a_i(t-1)$ is the activation of neuron i at time step t and $t-1$ respectively, w_{ij} is the strength of the connection between neuron i and j , $o_j(t)$ is the output of a neuron j at time t , τ is the decay rate and $\sum w_{ij} \times o_j(t-1)$ represents the sum of all the inputs received by a neuron i at time step t . The output of a neuron is defined by a binary Heaviside step function.

$$o_j(t) = \begin{cases} 1 & \text{when } a_i(t) \geq \theta_i(t) \\ 0 & \text{when } a_i(t) < \theta_i(t) \end{cases} \quad (2)$$

Where $\theta_i(t)$ is the threshold of neuron i at time step t . The neuron fires if activation integrated over time exceeds the threshold. As the neuron fires, it sends a spike to other connected neurons. After the spike, the neuron loses all its activation. On the other hand, if the integrated activation is below the threshold the neuron remains quiescent. Up to this point, the neural model is consistent with a great deal of work with leaky integrate-and-fire neural models (e.g. [46]). The model incorporates the idea of dynamic threshold to account for neuronal fatigue. An integrate-and-fire neural model with dynamic threshold is rarely used despite its extensive biological evidence [45]. The dynamic threshold is a useful mechanism that helps the network move from one stable state to the next stable state. Threshold of each neuron changes at each time step depending on its fatigue.

$$\theta_i(t) = \begin{cases} \theta_i(t-1) + f_c & \text{when } a_i(t) \geq \theta_i(t) \\ \theta_i(t-1) - f_r & \text{when } a_i(t) < \theta_i(t) \end{cases} \quad (3)$$

Where f_c is the fatigue constant and f_r is the fatigue recovery constant. f_c is the constant value with which the

threshold increases at each time step t when the neuron fires. Increase in threshold increases the amount of activation required for a neuron to fire. This makes a neuron that has fired frequently in the recent past more difficult to fire. Fatigue therefore acts as a self inhibition mechanism that shuts off cell assemblies that have been active for a long time allowing another to activate. A neuron's threshold decreases with a constant value of f_r at each step that it does not fire. A lower threshold value for neurons that remain quiescent for long makes these neurons more sensitive to the input received.

TABLE I
NETWORK PARAMETERS

Symbol	Parameter	Base and Sequence Net
θ	Threshold	4.5
τ	Decay Rate	0.9
f_c	Fatigue	0.25
f_r	Fatigue Recovery	0.35
η	Learning Rate	0.07

Parameters for base and sequence networks of the model

Table 1 shows the parameters used for the networks in the simulation. These values were determined by a process of manual parameter exploration.

The Network Architecture

The system presented in this paper is a multi-network architecture, consisting of 5 networks connected to each other to process sequences of up to length 5. The network is composed of 1 base network and 4 sequence networks; Figure 1 shows the coarse topology of the system. The neurons within each network are sparsely connected via a distance based connectivity scheme. Since distance is important, the network organization follows a toroidal topology to avoid border effects. The number of connections leaving each neuron is uniform. Each network contains N neurons and network size for simulations described in this paper is $N = 250$ for each network. An $N \times N$ matrix $C = [c_{ij}]$ represents connectivity within each network.. There exists a connection between neuron i and j of same network only if $c_{ij} = 1$. Inhibitory neurons are randomly connected. The connectivity rule for excitatory neurons is given by.

$$c_{ij} = \begin{cases} 1 & \text{if } r < (d * \nu) \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

Where r is a random number, d is the city block distance between two neurons and ν is the connection probability. This connectivity rule suggests that connections between the neurons is a function of the distance between them and is also influenced by the chosen connection probability of the network. Simulations described in this paper set $d = 4$ and

$\nu = 0.8$. Therefore each neuron is connected to other neurons in the network within a local radius of 4.

Apart from being locally connected, each neuron in the network has one long distance axon with several synapses. One quarter of the connections go down this axon, and in this area connections are also distance biased. So, neurons send connections to their area and one other area in the network. This is motivated by biological neural topology [45].

Each network is broken into several orthogonal cell assemblies. This is done by a process of unsupervised training (see sections III C and D).

The base network is connected to the level one sequence network with fixed forward connections as shown in Fig1.

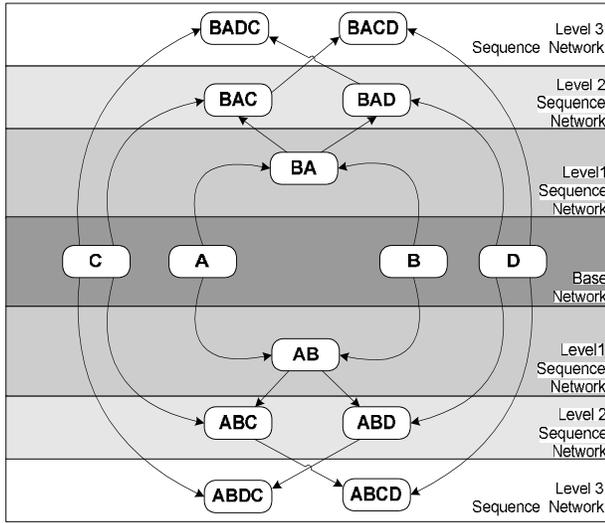


Fig.1. One Base Network and four sequence networks are connected to process sequential information.

Each sequence network is connected to its higher level sequence network. The synaptic strength of these connections is pre-calculated, there is no training.

The sequence networks store the serial order association between the basic patterns. Each serial order association in a sequence network is represented by a cell assembly composed of 50 neurons. The A and B cell assembly represent items A and B in the base network, and the AB and BA cell assemblies in the first sequence network represent their occurrence in different temporal order. Individual items are stored in such a way that the pattern of short term memory activity, neural firing, encodes both the items that have occurred and the temporal order in which they have occurred. In the cognitive literature, such a mechanism is often said to store both item and order information [20].

The model presented here employs a weight based technique to encode and maintain serial order information in the network. Connections between each network have pre-calculated weights to detect the temporal order of activations

received from the connected network. Sequence cell assemblies have two types of input, one from the first element of the sequence that it represents and another from the second element. So a level 1 sequence CA has two types of connections from the base network, and a higher level sequence CA has the first type from the prior level, and the second from the base. The weights calculated for simulations described in this paper are 0.41 from first element and 0.18 from second element of the sequence. There are 100 connections from each cell assembly of the base network to each cell assembly in the sequence network.

For example, neurons in the base network representing A are connected with the neurons of the AB cell assembly in the level 1 sequence network with a fixed weight of 0.41, and neurons in B are connected with the neurons of the AB cell assembly in the level one sequence network with a fixed weight of 0.18. Experiments show that the activation received by AB from A increases the activation level of neurons in the assembly but is not enough to fire them.

Learning

The base network is trained with five basic patterns and each sequence network is trained with five sequential patterns. Neurons in the appropriate Cell Assembly were simultaneously fired via external activation. Each pattern is composed of 50 neurons, and the patterns do not share any neurons (are orthogonal).

The network learns these patterns using a correlatory Hebbian learning rule [47]. Following this learning mechanism, the weights of the synaptic connections are modified according to the equation:

$$\Delta^+ w_{ij} = (1 - w_{ij}) * \eta \quad (5)$$

$$\Delta^- w_{ij} = w_{ij} * -\eta \quad (6)$$

Where w_{ij} is the weight of the synapse from neuron i to neuron j and η is the learning rate. At each computing cycle the synaptic weight changes if the pre synaptic neuron fires. The weight increases, the first Hebbian equation, if the post synaptic neuron also fires. On the other hand, weights are reduced if the post synaptic neuron does not fire, the second anti-Hebbian equation. This combination of Hebbian and anti-Hebbian learning forces the synaptic weights from neuron i to neuron j, w_{ij} , towards the likelihood that neuron j fires when neuron i fires [47].

Simulation Experiment

Simulations were run to test the storage of items and their sequences of length two through five. The simulations consisted of the following steps.

First, the base network was trained with five basic patterns and each sequence network with five sequence patterns using

the learning mechanism discussed in section III C. After the learning phase, the sequences to be stored were encoded into the network as discussed in section III B.

In order to test the storage of sequential information, sequences of basic items were presented to the base network; some presentations had stored sequences, and some had sequences that were not stored. Each pattern consisting of 50 neurons was presented for 50 cycles. So, if the ABCDE sequence was presented, A would be presented for 50 cycles, then B for 50, and so forth until E ceased presentation at cycle 250.

Each pattern when presented to the base network activated the cell assembly in the base network representing the input. Once activated, the cell assembly reverberated after the external input had ceased. Each cell assembly has strong inhibitory and weak excitatory connections to other cell assemblies in the same network. Therefore if one cell assembly is activated it inhibits the neurons belonging to other cell assemblies in the network. The neurons belonging to the reverberating cell assembly fatigue due to continuous firing. As the next input in the sequence is presented to the base network, fatigue and inhibition cause the previously active cell assembly to deactivate.

After the patterns were presented in a particular sequence, activity was measured in sequence cell assemblies with any firing showing that the presented sequence was stored. Consider an experiment to test the storage of a simple sequence ABC. When patterns were presented in this sequence to the base network, the A cell assembly is ignited at first. The A cell assembly then sends activation to the sequence cell assemblies to which it is connected. The activation thus sent is not enough to ignite any sequence cell assembly but primes them, that is, increases their activation level. As the next pattern in the sequence, B, is presented it shuts down the A cell assembly in the base network and ignites the already primed AB cell assembly in the first level sequence network. Activation of AB indicates A is followed by B in the input sequence. The AB cell assembly sends activation to other cell assemblies in the second level sequence network to which it is connected and primes them. Presentation of C ignites the C cell assembly in the base network and also the ABC cell assembly in the second level sequence network which was already primed due to activation received from the AB cell assembly. Results show that the ABC cell assembly is only ignited when patterns are presented in this particular sequence.

RESULTS

Experiments were carried out using simple sequences of length 2 to 5. Table II shows a summary of results obtained from the experiment.

The system stored five patterns of length two through five. The five patterns were ABCDE, BACDE, ECDAB, EBEAC, and DCABE. The shorter length sequences were

the prefixes of the length five sequences (e.g. the length two patterns were AB, BA, EC, EB, and DC). The system was presented with each instance and five instances that were not stored: CABDE, CBADE, BCDAE, ACBED, and EACBD. These were selected arbitrarily but no adjacent letters were permitted (e.g. AABCD would not work). Letters can occur more than once in a sequence, e.g. EBEAC. This was run on 20 different networks. Table II shows the results. The simulated model recognizes the stored sequences effectively, but recognition error increases as the length of the sequence increases. Sequences that are not stored are not retrieved, so there are no false positives.

Presentation of ECDAB, for example, leaves activity in the level 1 AB sequence cell assembly. This is not reflected in Table II. Similarly, the sequence CABDE, which is not stored, leaves activation in AB, as AB is stored.

TABLE II
SIMULATION RESULTS

	Length 2	Length 3	Length 4	Length 5
Stored	94	91	89	88

Results of retrieval of sequences stored

CONCLUSION

A model of sequential memory based on the hypothesis of cell assemblies has been proposed. The model presented introduces the novel idea of sequence cell assemblies which account for neural representation of serial order information in long-term memory. The sequential information is stored in the inter-network connections. However, these connections were calculated and a more complete account would investigate how these weights are adapted by repeated presentations of a sequence. Future work will include forming long term sequential memory from presentations of sequences. This may benefit from work on sequential rehearsal work (e.g. [40]).

Simulations have shown networks that store five sequences of up to length five. Further studies will be carried out to evaluate the scalability of the proposed model, but the model requires only a constant number of sequence cell assemblies for each stored sequence. The idea of chunking as a memory mechanism proposed by Miller [48] can provide a useful starting point for developing a network which does not grow in size too quickly with longer sequences. It is also important to test the validity of this model with evidence from brain studies. Exploring these aspects is critical for understanding how neural circuits are used to produce cognitive function.

REFERENCES

- [1] D. Hebb, *The Organization of Behavior*. New York: John Wiley & Sons, 1949.

- [2] M. Abeles, *Corticonics: Neural Circuits of the Cerebral Cortex*. Cambridge, UK: Cambridge University Press, 1991.
- [3] D.J. Amit, *Modeling brain function*. Cambridge, (UK): Cambridge University Press, 1989.
- [4] V. Braitenberg, "Cell assemblies in the cerebral cortex," in *Theoretical approaches to complex systems Lecture notes in biomathematics*, vol. 21 R. Heim and G. Palm, Eds. Berlin: Springer Verlag, 1978, pp. 171-188.
- [5] E. Fransén, A. Lansner and H. Liljenström "A model of cortical associative memory based on Hebbian cell assemblies" in *Connectionism in a Broad Perspective: Selected papers from the Swedish Conference on Connectionism*, Eds. L. F. Niklasson and M. B. Bodén, pp. 165-172, 1994.
- [6] G. L. Gerstein, P. Bedenbaugh and A. M. H. J. Aertsen, "Neuronal Assemblies," *IEEE Transactions on Biomedical Engineering*, Vol.36 No 1, pp. 4-14, 1989.
- [7] C. Huyck, "Cell Assemblies and Neural Network Theory: From Correlators to Cell Assemblies," Middlesex University Technical Report ISSN 1462-0871 CS-02-02, 2002.
- [8] S. Kaplan, M. Sonntag and E. Chown, "Tracing recurrent activity in cognitive elements (TRACE): A model of temporal dynamics in a cell assembly," *Connection Science*, vol. 3, pp. 179-206. 1991.
- [9] R. Miller, "Neural assemblies and laminar interactions in the cerebral cortex," *Biol. Cybern.* 75:253-261, 1996.
- [10] P. M. Milner, "Neural Representations: Some Old Problems Revisited," *J. Cog. Neurosci.* 8:69-77, 1996.
- [11] G. Palm, *Neural Assemblies. An Alternative Approach to Artificial Intelligence Studies of Brain Function*. New York: Springer-Verlag, 1982.
- [12] F. Pulvermuller, "Words in the Brain's Language," *Behavioral and Brain Sciences*, 22, 253-336, 1999.
- [13] Y. Sakurai, "How do cell assemblies encode information in the brain?" *Neuroscience and Biobehavioral Reviews*, Vol. 23, No 6, pp. 785-796, 1999.
- [14] F. T. Sommer, "On cell assemblies in a cortical column," *Neurocomputing Paper* (32-33) pp.517 - 522, 2000.
- [15] M. V. Tsodyks, "Associative memory in neural networks with the Hebbian learning rule," *Modern Physics Letters*, vol. 3, pp. 555-560, 1989.
- [16] W. A. Wickelgren, "Webs, cell assemblies, and chunking in neural nets. Concepts in Neuroscience," *Psychology Dep. Columbia Univ* 3, 1-53, 1992.
- [17] H. Ebbinghaus, *Memory: A contribution to Experimental Psychology*. H. Ruger and C. Bussenius, Trans. New York: Dover Publications, 1964.
- [18] Gardner-Medwin, "The Recall of Events through the Learning of Associations between their Parts" in *Proceedings of the Royal Society of London. Series B, Biological Sciences*, Vol. 194, No. 1116, 1976, pp. 375-402 1997.
- [19] D. Kleinfeld and H. Sompolinsky, "Associative neural network model for the generation of temporal patterns. Theory and application to central pattern generators." *Biophys J.*, vol. 54, pp. 1039-1051, 1988.
- [20] S. Lewandowski, and B. Murdock, "Memory for serial order," *Psychological Review*, vol. 96, pp. 25-57, 1989.
- [21] W. A. Wickelgren, "Context sensitive coding, associative memory, and serial order in (speech) behavior," *Psychological Review*, 76, 1-15 1969.
- [22] S. Lashley, "The problem of serial order in behavior," in *Cerebral Mechanisms in Behavior*, Ed. L.A. Jeffress, pp. 112-146 New York: Wiley, 1951.
- [23] K. J. Lang, A. H. Waibel G. E. Hinton, "A time delay neural-network architecture for isolated word recognition," *Neural Networks*, 3(1):23-44, 1990.
- [24] E. Wan, "Temporal backpropagation for FIR neural networks," *International Joint Conference on Neural Networks*, San Diego, Vol. 1 pp. 575-580, 1990.
- [25] J. Elman, "Finding structure in time," *Cogn. Sci* vol. 14, Univ. California, pp. 179--211, 1990.
- [26] M. Jordan, "Attractor dynamics and parallelism in a connectionist sequential machine," *Proc. of the Eighth Ann. Conf. of the Cognitive Science Society*, pp.531-546, 1986.
- [27] B. A. Pearlmutter, "Gradient calculation for dynamic recurrent neural networks: a survey," *IEEE Transactions on Neural Networks*, 6(5):1212-1228, 1995.
- [28] D. E. Rumelhart, G. E. Hinton and R. J. Williams *Learning internal representations by error propagation Parallel distributed processing: explorations in the microstructure of cognition*, Eds. D. E. Rumelhart, J. L. McClelland and the PDP Research Group, pp. 318-362, Cambridge MA : MIT Press, 1986.
- [29] R. J. Williams and D. Zipser, "learning algorithm for continually running fully recurrent neural networks," *Neural Computation*, vol. 1, pp. 270-280, 1989.
- [30] J. Hopfield, "Neural networks and physical systems with emergent collective computational properties," *Proceedings of the National Academy of Sciences of the USA*, vol. 79, pp. 2554 - 2588, 1982.
- [31] I. Guyon, L. Personnaz, J. P. Nadal, and G. Dreyfus, "Storage and retrieval of complex sequences in neural networks," *Phys. Rev. A*, vol.38: pp. 6365-6372, 1988.
- [32] B. Kosko, "Bidirectional Associative Memories," *IEEE Transactions on Systems, Man, and Cybernetics Univ. California*. 18, pp. 49-60, 1988.
- [33] D. W. Tank and J. J. Hopfield, "Neural computation by concentrating information in time," *Proc. Natl. Acad. Sci. USA*, vol. 84, pp. 1896-1900, 1987.
- [34] H. Sompolinsky, I. Kanter, "Temporal association in asymmetric neural networks," *Phys. Rev. Letters*, 1986.
- [35] T. Kohonen, *Self Organizing Maps*. New York : Springer-Verlag, 1997.
- [36] J. -C. Chappelier and A. Grumbach, "A Kohonen Map for Temporal Sequences" in *Proceedings of the Eighth 8th International Conference on Neural Networks and their Applications*, 1996, pp.104- 110.
- [37] Chappell G. J., Taylor J. G. "The temporal Kohonen Map" in *Neural Networks*, Vol.6, Oxford, UK:Elsevier Science Ltd, 1993, pp. 441-445.
- [38] M. Varsta, J. del R. Millán and J. Heikkonen, "A Recurrent Self Organizing Map for Temporal Sequence Processing," in *ICANN'97*, LNCS vol. 1327, pp. 421-426, 1997.
- [39] G. Bradski, G. Carpenter and S. Grossberg, "STORE Working Memory Networks for Storage and Recall of Arbitrary Temporal Sequences," *Biological Cybernetics*, 71, 469-480, 1994.
- [40] G. Deco and E. Rolls, "Sequential Memory: A Putative Neural and Synaptic Dynamical Mechanism," *J. Cog. Neuroscience*, 17, 294-307, 2005.
- [41] M. Botvinick and D. C. Plaut, "Short-term memory for serial order: A recurrent neural network model," *Psychological Review*, 113, 201-233, 2006.
- [42] R. B. Stein, "A theoretical analysis of neuronal variability," *Biophys J*, vol. 5, pp. 173-194, 1965.
- [43] V. Braitenberg, "Some arguments for a theory of cell assemblies in the cerebral cortex," In *Neural Connections, Mental Computation*, Nadel, Cooper, Culicover, and Harnish Eds. Cambridge MA: MIT Press, 1989.
- [44] J. Eccles, "Chemical transmission and dale's principle." *Prog. Brain Research*, vol. 68, pp. 3-13, 1986.
- [45] P. Churchland and T. Sejnowski, *The Computational Brain*. Cambridge MA: MIT Press, 1992.
- [46] W. Maas and C. Bishop, *Pulsed Neural Networks*. Cambridge MA: MIT Press, 2001.
- [47] C. Huyck, "Overlapping cell assemblies from correlators," *Neurocomputing Middlesex University Technical Report* 56: 435-439 2004.
- [48] G. A. Miller, "The Magical Number Seven, Plus or Minus Two: Some Limits on our Capacity for Processing Information," *Psych. Review*, 63, 81-97 1956.