

# Learning Complex Population-Coded Sequences

Kiran V. Byadarhaly, Mithun Perdoor, Suresh Vasa, Emmanuel Fernandez, and  
Ali A. Minai \*

Department of Electrical & Computer Engineering, University of Cincinnati,  
Cincinnati, OH 45221-0030  
[bkiranv@yahoo.com](mailto:bkiranv@yahoo.com), [perdoomc@email.uc.edu](mailto:perdoomc@email.uc.edu), [vasash@email.uc.edu](mailto:vasash@email.uc.edu),  
[emmanuel@eecs.uc.edu](mailto:emmanuel@eecs.uc.edu), [Ali.Minai@uc.edu](mailto:Ali.Minai@uc.edu)

**Abstract.** The sequential structure of complex actions is apparently learned at an abstract “cognitive” level in several regions of the frontal cortex, independent of the control of the immediate effectors by the motor system. At this level, actions are represented in terms of kinematic parameters – especially direction of end effector movement – and encoded using population codes. Muscle force signals are generated from this representation by downstream systems in the motor cortex and the spinal cord.

In this paper, we consider the problem of learning population-coded kinematic sequences in an abstract neural network model of the medial frontal cortex. For concreteness, the sequences are represented as line drawings in a two-dimensional workspace. Learning such sequences presents several challenges because of the internal complexity of the individual sequences and extensive overlap between sequences. We show that, by using a simple module-selection mechanism, our model is capable of learning multiple sequences with complex structure and very high cross-sequence similarity.

**Key words:** Sequence learning, population coding, motor system

## 1 Introduction

The ability to act in complex and purposive ways is a central attribute of higher animals, and is essential to the understanding of intelligence and cognition. Over the last several decades, neuroscientists and cognitive scientists have elucidated many of the mechanisms underlying motor control. Experiments with primates have shown that complex voluntary movements are encoded at multiple levels in the cortex, brainstem and the spinal cord [1–15]. However, the overall process by which such movements are generated remains a subject of significant debate [16–20].

Research with simple animals such as lampreys and salamanders [21, 22] has suggested the existence of *motor programs* – pre-configured patterns of movements that can be triggered as a whole by selection through the basal ganglia

---

\* Corresponding author

and the brainstem. This view has often been applied to the higher vertebrates as well, and offers a plausible explanation for simple movements such as walking, swallowing, reaching, etc. However, much more complex movements – such as writing, speaking, playing on a keyboard, pitching a baseball – can become integrated into a precisely repeatable, yet flexible motor program, and recalled as a *single entity* when needed. This “chunking” [23] is thought to arise through a combination of pattern recognition and sequence learning, and is the focus of this paper.

## 2 Background and Motivation

All complex movements can be considered sequences of simpler movements, and there is considerable experimental evidence showing that neural activity explicitly encodes the sequential structure of movement at several levels [1, 2, 4, 19]. However, in many cases – e.g., writing, speaking, playing on a keyboard, etc. – what might be learned initially as a sequence of simple movements eventually becomes a single complex movement. For example, a child may learn to write letters by connecting dots, or as sequences of pencil strokes, but eventually, each of these sequences becomes stored as a *single object* – still sequential in structure, but triggered by a single cognitive command, e.g., ”write A”. Something similar must happen as the child learns to write whole words without spelling them out, or speaking them as continuous phoneme sequences. Sequential tasks are also used extensively as an experimental paradigm by researchers studying the motor system [3, 4, 9–12, 14, 19]. Based on these considerations, sequence learning has been considered a fundamental issue in motor control, and several models have been developed for it [24–29]. However, the such sequence learning still presents several challenges.

Though partially challenged by some recent data [14, 19], the standard view of motor control [6] postulates two distinct system levels. The higher level system, comprising the prefrontal cortex (PFC), the premotor cortex (PM), the supplementary motor area (SMA) and the pre-SMA, encodes kinematic information such as direction and velocity as well as information about task context [3, 4, 9, 8, 10–12, 19]. In particular, the direction of end effector movement is represented using *population codes* [1, 2] defined over neurons tuned to particular *preferred directions* (see next section for details). In contrast, the lower level system comprising the motor cortex (M1), the spinal cord (SC) and intermediate systems represents posture-dependent muscle force signals needed to execute the actions [16, 17, 20]. Thus, with reference to sequential actions, the overall system implements a convenient division of labor, where motor sequences such as writing letters or words, drawing shapes, playing key sequences, etc., are learned as *action programs* at an abstract, posture- and effector-independent kinematic level in the PFC/PM/SMA system, and are then “translated” into sequences of motor commands by M1 [3], possibly guided by the basal ganglia and cerebellum [23, 30, 31]. Of course, this does not preclude the possibility that sequence recognition and learning may also occur in M1 [5, 14, 19], and the model we study here

could well apply to population-coded representations in M1 as well. However, we consider purely kinematic action programs, encoded as sequences of direction population codes.

There are several difficulties inherent in learning population-coded action sequences, including the following: 1) Successive elements of an individual sequence may be very similar or even identical (e.g., representing several moves in the same direction); 2) Within an individual sequence, the same element may be followed by different successors at different points, requiring disambiguation [32]; 3) Different sequences may overlap strongly, e.g., the sequences for writing an O and a C may share most of their structure; 4) The directional information encoded in the population code may be noisy over different learning trials.

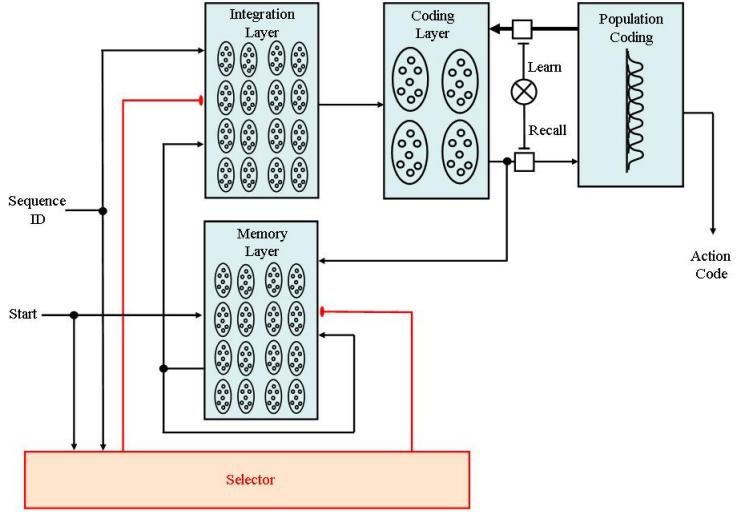
These features mean that a simple heteroassociative scheme linking successive elements is not feasible. Approaches have been suggested to address some of these problems for non-population coded [33, 34, 32, 27, 28, 35], and population-coded [24–26] data.

In this paper, we present an abstract but biologically motivated model of how population-coded action sequences may be learned in the brain. This model is based on data from experimental studies on sequence encoding and learning in many areas of the frontal cortex. Some neurons in the prefrontal cortex fire preferentially at the beginning of action sequences [36], while others remain active for the duration of the sequence [37]. These signals go to both the SMA/PM region, and to the basal ganglia and cerebellum, where they are hypothesized to select [23, 36, 31] and pace [29, 31] action sequences, respectively. Neurons with sequence-specific activity, i.e., sequence identifiers, have been shown in pre-SMA, while neurons encoding element order in action sequences have been found in both PFC and SMA [3, 38, 9, 7, 10, 36]. Our model incorporates these observations to build upon an earlier sequence learning model [27]. In particular, our system can learn highly similar sequences by sequence-specific selection of neural groups within the network. This is an example of switched modularity, which we have proposed as a general mechanism for information processing in complex biological networks [39–42]

### 3 Model Description

A complex drawing, e.g., a letter or shape, is modeled as a sequence of *elementary actions* starting from a canonical initial point. Each elementary action results in the drawing of a short segment in a specific direction on a 2-dimensional workspace. This is similar to tasks used in behavioral experiments with primates [3, 4, 9–12, 14, 19]. Thus, for example, an L-shape starting at the top is a sequence of downward segments followed by a sequence of leftward ones.

The system learns action sequences from noisy exemplars, where each elementary action – henceforth called *action* – is represented as a population code over a set of directionally tuned neurons. We assume that this representation arises in cortical regions receiving sensory input, and is available during learning.



**Fig. 1.** The sequence learning module.

The structure of the sequence learning system is shown in Figure 1. For learning, the system receives three inputs: 1) A *start* input, marking the beginning of a new sequence (corresponding to the PFC signal found in [36]); 2) An *action* input, which is the population code of the current elementary action (corresponding to representations found in PFC [10] and SMA [3, 38, 9, 7]); and 3) A *sequence ID* input, which represents the identity of the action sequence (corresponding to sequence-specific activity in the pre-SMA [38, 9, 7]).

The system operates in two modes: 1) A *learning mode*, in which it receives sequences of action inputs and learns them in association with the given sequence ID; and 2) A *recall mode*, in which it receives no action input and recalls the sequence associated with the given sequence ID. The recalled sequence is decoded into the original population code by the population coding system. This and other parts of the system are described below in greater detail.

### 3.1 Population Coding

Each action in the system is encoded by its direction using a population code [1]. This is done by a layer of  $N_P$  directionally-tuned neurons. The response of a neuron,  $k$ , in this layer is given by a cosine function [2, 4]:  $z_k = \cos(\theta_d - \theta_k^*)$ , where  $\theta_d$  is the currently coded direction (represented as an angle in a fixed reference frame), and  $\theta_k^*$  is the preferred direction of neuron  $k$  (note that  $z_k$  can be rectified by adding an offset, but we do not use that here for simplicity.) The preferred directions for the neurons in the layer are chosen to cover  $0^\circ$  to  $360^\circ$  uniformly, so the vector,  $z^P = [z_1 \dots z_{N_P}]$  is a population coded representation of the direction

$\theta_d$  — and thus of the corresponding action. The direction can be recovered from it by using a simple decoding function [4]:  $\hat{\theta} = \text{phase} \left[ \frac{1}{N_P} \sum_{k=1}^{N_P} z_k \exp(i\theta_k^*) \right]$ .

The direction input to the population coder comes from the sensory areas (not simulated) during learning and from the sequence learning system during recall.

### 3.2 Sequence Learning System

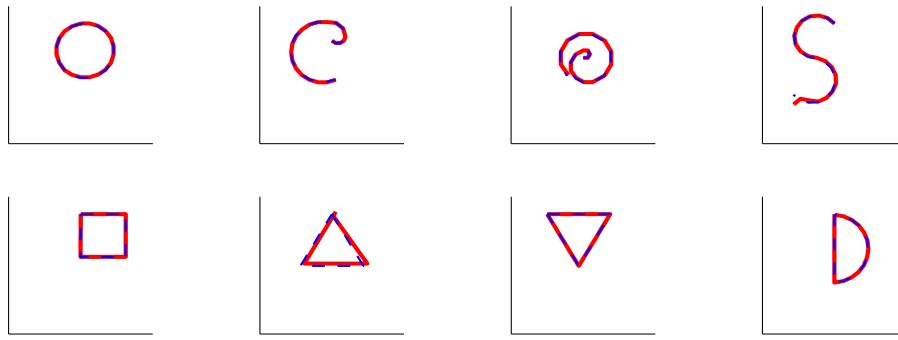
The sequence learning system learns abstract representations of action sequences from noisy exemplars. As shown in Figure 1, the core of the system comprises three layers, each implemented as a set of winner-take-all (WTA) clusters. These clusters may be considered an abstraction of cortical hypercolumns [43]. Activity in this core region – corresponding roughly to parts of the PFC and SMA – is selectively biased in a sequence-specific way by the *selector* system, which plays the role hypothesized for the basal ganglia [23, 37, 30, 36, 31].

The *coding layer*,  $C$ , has  $N_C$  neurons organized into  $M_C$  clusters of  $n_C = N_C/M_C$  neurons each. Activity within each cluster is WTA, so only the most strongly activated neuron in the cluster can fire. Non-selected clusters have no activity. The *memory layer*,  $R$ , is a recurrently connected layer of  $N_R$  neurons, organized into  $M_R$  clusters of  $m_r$  neurons each. Only  $m_R < M_R$  clusters are selectively activated at a time by the signal from the selector system, and activity in each selected cluster is WTA. The *integration layer*,  $I$ , has  $N_I$  neurons clustered into  $M_I$  WTA clusters of  $n_I$  neurons each. Of these,  $m_I < M_I$  clusters are selectively activated at a time by the selector system. The selector system,  $B$ , is a heteroassociative network that maps each input into a particular sparse pattern of  $N_B = M_R + M_I$  binary outputs, each of which gates a cluster in the memory or integration layers. This architecture is a simplified version of that found in the BG, whose neurons target very specific patches of the cortex [23, 30].

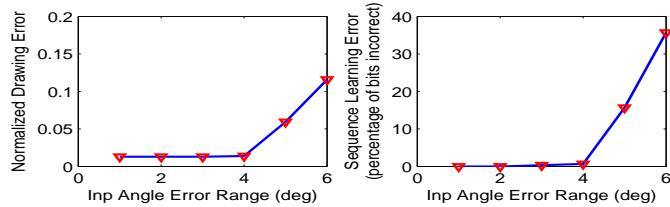
During learning, the coding layer receives a very strong action input,  $X(t)$ , from the population coder, generating a sparse representation,  $Z^C(t)$ , of the current observed action. This is projected to the memory layer via fixed random weights. The memory layer also receives input from itself through fixed random weights, so that its current state,  $Z^R(t)$ , comes to represent a compressed code for the activity of the coding layer over the last several steps. The integration layer receives the current sequence ID input as well as  $Z^R(t)$  through fixed random weights, and integrates them into a sequence-specific representation,  $Z^I(t)$ , of the sequence up to that point. This representation is then associated with the next coding layer state,  $Z^C(t+1)$ , through modifiable weights from Layer I to Layer C using a delta learning rule [44]. In both the  $R$  and  $I$  layers, the selector system selects sequence-specific groups of clusters, thus ensuring that representations for differently identified sequences remain distinct even if the sequences are similar. The *start* signal is input to the memory layer, where it triggers a heteroassociatively stored *initial representation*, thus ensuring that the integration layer always receives the same input at the start of a sequence. The *start* signal also resets the selector system.

During recall, the flow between the population coder and the coding layer is reversed, and the system is given only the sequence ID and *start* inputs. This then triggers the recall of the previously learned sequence as the coding and memory layers recover their states during learning. The activity of Layer  $C$  is decoded by the population coding system into the population code for the corresponding actions. This decoding is based on training of the weights from Layer  $C$  to the population coder using a delta rule [44]. This learning can occur concurrently with the sequence learning or separately via ideomotor babbling.

#### 4 Simulations and Results

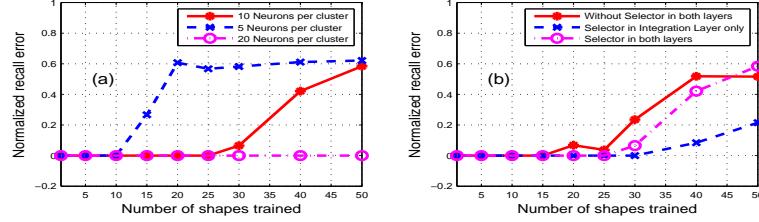


**Fig. 2.** Recall of eight learned figures. The dashed (blue) lines show the ideal figures, the solid (red) lines the actual figures generated after learning. The input has a small amount of angular noise



**Fig. 3.** Effect of angular input error in input on system performance: (a) Mean normalized angular drawing error/segment; (b) Percentage of incorrect bits in sequence recall.

An implementation of the system with  $M_C = 30$ ,  $M_R = 60$ ,  $M_I = 60$  and  $n_C = n_R = n_I = n$  was simulated using a canonical repertoire of 8 shapes,

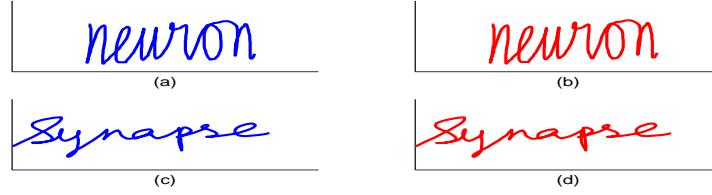


**Fig. 4.** Graph (a): Effect of varying cluster sizes on the capacity for learning near-identical shapes. Graph (b): Comparison of the system without cluster selection and those with cluster selection in Layer  $I$  and both layers  $I$  and  $R$ , while learning near-identical shapes.

chosen so that several of them have strong similarities, while others have runs of repeated actions. Each action code generated a line segment of fixed length in the direction specified by the code. Only connected shapes were used, though disconnected shapes can be included easily by giving the simulated pen a degree of freedom perpendicular to the workspace. The model currently does not include any inertial terms so the direction changes instantaneously with each population code. The ID input was encoded by 50-bit binary vectors, with 5 randomly chosen bits set to 1 for each figure.

Figure 2 demonstrates the shapes that the system with  $n = 10$  (i.e.,  $N_C = 300$ ,  $N_R = 600$  and  $N_I = 600$ ) draws for each of the training sequences after learning. Each figure was presented 20 times during learning, with random angular noise of  $\pm 0.5^\circ$  for each line segment. Figure 3 shows performance when an angular error uniformly distributed in a range around  $0^\circ$  is added to each action segment during training. It demonstrates that the system is quite robust to such noise, mainly because of the use of WTA clusters in the sequence learner.

The main issue addressed in this paper is the system's ability to learn highly similar sequences, and the role of the selector system in doing so. This was studied by training the system on shapes with 20 segments each, such that the first 19 segments in each shape were almost identical (in a circular arrangement) and the only significant difference arose in the 20th segment. The shapes had distinct IDs. To learn these shapes, the system had to distinguish the internal representations of the sequences sufficiently. Figure 4(a) shows the results for storing up to 50 near-identical shapes in networks with  $n = 5$ , 10 and 20. The results demonstrate that increasing the size of clusters increases the capacity for such learning, but even a small network can store several nearly identical sequences. Using the same training set (near-identical shapes), Figure 4(b) shows the comparative performance of a network without cluster selection, one with cluster selection only in layer  $I$ , and one with selection in both layers  $I$  and  $R$ . It is clear that ID-specific cluster selection provides significant increase in capacity for learning similar sequences. Interestingly, the best performance is obtained when selection is applied only to layer  $I$ , presumably because disambiguation is important mainly for training the  $I$ -to- $C$  connections.



**Fig. 5.** Learning a complex sequence in a network with  $n = 20$ . Graph (a, c): Training text; Graph (b, d): Recalled text.

Finally, Figure 6 shows the results when the same network is trained to store two very long and complex sequences representing two different words. As shown in the figure, the sequences are recalled nearly perfectly after training.

## 5 Conclusion

In this paper, we have presented a model for learning population-coded kinematic sequences in a simple but biologically motivated neural system. In particular, our results show that, with the inclusion of a basal ganglia-like selector, the system is able to learn a large number of almost identical sequences, and many different types of shapes, including very complex ones. More detailed studies of the system, including the role of velocity coding, inertial factors, internal noise and downstream motor control will be presented in future papers.

### Acknowledgement:

This research was partly supported by a grant from the University of Cincinnati College of Engineering to the *Intelligent Autonomous Systems* research group.

## References

1. Georgopoulos, A., Kalaska, J., Caminiti, R., Massey, J.: On the relations between the direction of two-dimensional arm movements and cell discharge in primate motor cortex. *Journal of Neuroscience* **2** (1982) 1527–1537
2. Schwarz, A., Kettner, R., Georgopoulos, A.: Primate motor cortex and free arm movement to visual targets in 3-d space. i. relations between single cell discharge and direction of movement. *Journal of Neuroscience* **8** (1988) 2913–2927
3. Mushiake, H., Inase, M., Tanji, J.: Neuronal activity in the primate premotor, supplementary and precentral motor cortex during visually guided and internally determined sequential movements. *Journal of Neurophysiology* **66** (1991) 705–718
4. Schwartz, A.: Motor cortical activity during drawing movements: single unit activity during sinusoid tracing. *Journal of Neurophysiology* **68** (1992) 528–541
5. Kakei, S., Hoffman, D., Strick, P.: Muscle and movement representations in the primary motor cortex. *Science* **285** (1999) 2136–2139

6. Seitz, R., Stephan, K., Binkofski, F.: Control of action as mediated by the human frontal lobe. *Experimental Brain Research* **133** (2000) 71–80
7. Shima, K., Tanji, J.: Neuronal activity in the supplementary and presupplementary motor areas for temporal organization of multiple movements. *Journal of Neurophysiology* **84** (2000) 2148–2160
8. Kakei, S., Hoffman, D., Strick, P.: Direction of action is represented in the ventral premotor cortex. *Nature Neuroscience* **4** (2001) 1020–1025
9. Nakamura, K., Sakai, K., Hikosaka, O.: Neuronal activity in medial frontal cortex during learning of sequential procedures. *Journal of Neurophysiology* **80** (1998) 2671–2687
10. Averbeck, B., Chafee, M., Crowe, D., Georgopoulos, A.: Parallel processing of serial movements in prefrontal cortex. *Proceedings of the National Academy of Sciences USA* **99** (2002) 13172–13177
11. Averbeck, B., Chafee, M., Crowe, D., Georgopoulos, A.: Neural activity in prefrontal cortex during copying geometrical shapes. i. single cells encode shape, sequence and metric parameters. *Experimental Brain Research* **150** (2003) 127–141
12. Averbeck, B., Crowe, D., Chafee, M., Georgopoulos, A.: Neural activity in prefrontal cortex during copying geometrical shapes. ii. decoding shape segments from neuronal ensembles. *Experimental Brain Research* **150** (2003) 142–153
13. d'Avella, A., Saltiel, P., Bizzi, E.: Combinations of muscle synergies in the construction of natural motor behavior. *Nature Neuroscience* **6** (2003) 300–308
14. Lu, X., Ashe, J.: Anticipatory activity in primary motor cortex codes memorized movement sequences. *Neuron* **45** (2005) 967–973
15. Graziano, M.: The organization of behavioral repertoire in motor cortex. *Annual Review of Neuroscience* **29** (2006) 105–134
16. Ajemian, R., Bullock, D., Grossberg, S.: Kinematic coordinates in which motor cortical cells encode movement direction. *Journal of Neurophysiology* **84** (2000) 2191–2203
17. Sergio, L., Kalaska, J.: Systematic changes in motor cortex cell activity with arm posture during directional isometric force generation. *Journal of Neurophysiology* **89** (2003) 212–228
18. Cisek, P.: Cortical mechanisms of action selection: the affordance competition hypothesis. *Phil. Trans. R. Soc. B* **362** (2007) 1585–1599
19. Matsuzaka, Y., Picard, N., Strick, P.: Skill representation in the primary motor cortex after long-term practice. *Journal of Neurophysiology* **97** (2007) 1819–1832
20. Ajemian, R., Green, A., Bullock, D., Sergio, L., Kalaska, J., Grossberg, S.: Assessing the function of motor cortex: single-neuron models of how neural response is modulated by limb biomechanics. *Neuron* **58** (2008) 414–428
21. Grillner, S.: Biological pattern generation: the cellular and computational logic of networks in motion. *Neuron* **52** (2006) 751–766
22. Ijspeert, A., Crespi, A., Ryczko, D., Cabelguen, J.M.: From swimming to walking with a salamander robot driven by a spinal cord model. *Science* **315** (2007) 1416–1420
23. Graybiel, A.: Building action repertoires: memory and learning functions of the basal ganglia. *Current Opinion in Neurobiology* **5** (1995) 733–741
24. Lukashin, A., Georgopoulos, A.: A neural network for coding of trajectories by time series of neuronal population vectors. *Neural Computation* **6** (1994) 19–28
25. Lukashin, A., Wilcox, G., Georgopoulos, A.: Overlapping neural networks for multiple motor engrams. *Proceedings of the National Academy of Sciences, USA* **91** (1994) 8651–8654

26. Lukashin, A., Amirikian, B., Mozhaev, V., Wilcox, G., Georgopoulos, A.: Modeling motor cortical operations by an attractor network of stochastic neurons. *Biological Cybernetics* **74** (1996) 255–261
27. Ans, B., Coiton, Y., Gilhodes, J.P., Velay, J.L.: A neural network model for temporal sequence learning and motor programming. *Neural Networks* **7** (1994) 1461–1476
28. Bullock, D., Cisek, P., Grossberg, S.: Cortical networks for control of voluntary arm movements under variable force conditions. *Cerebral Cortex* **8** (1998) 48–62
29. Grossberg, S., Paine, R.: A neural model of cortico-cerebellar interactions during attentive imitation and predictive learning of sequential handwriting movements. *Neural Networks* **13** (2000) 999–1046
30. Doya, K.: What are the computations of the cerebellum, the basal ganglia and the cerebral cortex? *Neural Networks* **12** (1999) 961–974
31. Houk, J.: Agents of the mind. *Biological Cybernetics* **92** (2005) 427–437
32. Minai, A., Barrows, G., Levy, W.: Disambiguation of pattern sequences with recurrent networks. In: Proc. WCNN, San Diego. Volume IV. (1994) 176–180
33. Sun, R., Giles, C.: Sequence Learning: Paradigms, Algorithms, and Applications. Springer (2001)
34. Reiss, M., Taylor, J.: Storing temporal sequences. *Neural Networks* **4** (1991) 773–787
35. Doboli, S., Minai, A.: Using latent attractors to discern temporal order. In: Proceedings of IJCNN, Budapest, Hungary (July 2004)
36. Fujii, N., Graybiel, A.: Representation of action sequence boundaries by macaque prefrontal cortical neurons. *Science* **301** (2003) 1246–1249
37. Jog, M., Kubota, Y., Connolly, C., Hillgaart, V., Graybiel, A.: Building neural representations of habits. *Science* **286** (1999) 1745–1749
38. Matsuzaka, Y., Aizawa, H., Tanji, J.: A motor area rostral to the supplementary motor area (presupplementary motor area) in the monkey: neuronal activity during a learned motor task. *Journal of Neurophysiology* **68** (1992) 653–662
39. Ghanem, A., Minai, A.: A modular gene regulatory network model of ontogenesis. In: Proc. Int. Conf. on Complex Sys., Boston, MA (2007)
40. Doboli, S., Minai, A., Brown, V.: Adaptive dynamic modularity in a connectionist model of context-dependent idea generation. In: Proceedings of the International Conference on Neural Networks, Orlando, Florida. (2007) 2183–2188
41. Minai, A., Iyer, L., Padur, D., Doboli, S.: A dynamic connectionist model of idea generation. In: Proc. IJCNN 2009. (2009)
42. Perumal, S., Minai, A.: Stable-yet-switchable (sys) attractor networks. In: Proc. IJCNN 2009. (2009)
43. Mountcastle, V.: The columnar organization of the neocortex. *Brain* **120** (1997) 701–722
44. Widrow, B., Hoff, M.: Adaptive switching circuits. In: 1960 IRE WESCON Convention Record, Part 4. (1960) 96–104