



HAL
open science

Neuronal mechanisms for sequence learning in behavioral modeling

Nicolas P. Rougier, Hervé Frezza-Buet, Frédéric Alexandre

► **To cite this version:**

Nicolas P. Rougier, Hervé Frezza-Buet, Frédéric Alexandre. Neuronal mechanisms for sequence learning in behavioral modeling. Sixteenth International Joint Conference on Artificial Intelligence, Workshop: Neural, Symbolic, and Reinforcement Methods for Sequence Learning, 1999, Stockholm, Sweden, 6 p. inria-00098756

HAL Id: inria-00098756

<https://inria.hal.science/inria-00098756v1>

Submitted on 26 Sep 2006

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Neuronal mechanisms for sequence learning in behavioral modeling

Nicolas Rougier and Hervé Frezza-Buet and Frédéric Alexandre
LORIA, BP 239, 54506 Vandoeuvre-les-Nancy Cedex, FRANCE

Abstract

Sequence learning and management have been reported as central features in behavioral studies. Indeed, tasks like route memorization or planning are temporal behaviors and can be hardly modeled without taking the temporal aspect into consideration. On the contrary, classical information processing tools are not very efficient at such temporal processing. In this paper, we shortly review these tools and show that another approach, close to neurobiological modeling, can yield a variety of neuronal mechanisms for several aspects of sequence learning and management.

1 Introduction

The study of living beings behavior brings up many questions concerning the ability of surviving in an hostile environment. Given that each environment possesses its own laws, any living being which is able to use all or part of these laws is able to predict some part of its future, and is then able to survive. Major part of these living beings are able to handle these laws thanks to experience and observation. Smart behavior is then the ability to construct, memorize and use some crucial temporal sequences, i.e. the ability to know in advance the consequences of a specific action, the ability to predict its own future. In this framework, we may ask which tools are we offered by computer sciences for endowing a model with sequence learning abilities. If traditionnal artificial intelligence methods are now able to deal with temporal sequence learning, it remains a very little known territory of the neural networks research field. Nonetheless, results from fine studies of behavior by neurobiologists and neuropsychologists allow the design of new computational models which can handle various kind of temporal sequences. The purpose of this paper is to introduce three mechanisms directly inspired by biology which will handle sequences at three different levels : a temporal level, a spatial level and a meta-level.

2 Current computational tools for modeling behavior: the share of neural networks

Many algorithms endowing an artificial system with the capability to behave in an external environment can be found in the literature [Maes *et al.*, 1996]. Non connectionist approaches will be first discussed here, in order to compare their properties to the ones provided by neural techniques.

2.1 Non connectionist approaches

Even if smart behavior of natural agents appears to be provided by an assembly of interconnected neural cells, non connectionist computational models are able to take fundamental features of behavior into account. Directly dealing with properties that have emerged from nervous cells of biological beings, these models can be viewed as *global approaches* to behavior generation.

High level planning capabilities required for clever behaviors have been modeled within the artificial intelligence framework. Behavior then consists in reaching goals or performing a task by successively splitting goals into subgoals or tasks into subtasks. The computation of subgoals from goals is provided by theorem prover-like architectures, dealing with explicit symbolic representations (see [Donnart and Meyer, 1994] for illustration). Nevertheless, even if the high level sequential management of goals and subgoals provided by such techniques is a central ability required by control architectures, the difficulty to ground symbols in a numerical perceptive input stream makes such algorithms difficult to apply on real robots.

Like artificial intelligence techniques, Markov decision process (MDP) based methods address behavior in a global way. We just mention here the Q-Learning algorithm [Watkins, 1989] that models Skinnerian operant conditioning. Within this framework, the behavior consists in selecting an action in order to make the agent execute a transition from the current environment state to another one, so that the reward received all along the sequence of state transitions is maximal. The method is grounded on a convergence theorem [Jaakkola *et al.*, 1994] provided by the MDP mathematics. Nevertheless,

applying such methods to real engineering problems requires a very large state space, and therefore unrealistic computational resources.

Endly, we would like here to mention operational research techniques, such as genetic algorithms that have been applied to design the control of behavior of low level agent of an assembly. These approaches are convenient to design elementary behaviors but require a long time to converge when applied to more “cognitive” behaviors.

2.2 Are current connectionist tools more suitable for behavior generation ?

Neural networks have explicitly been inspired from biological observations of the nervous cells. Applying such techniques consists in using local computations to make the targeted property of the network emerge. As opposed to the methods described in section 2.1, using neural tools for behavior generation is a *local approach*. Our purpose in this section is to overview classical neural networks and evaluate their relevance for behavior modeling, and then mention some less known but more specific artificial neuronal networks.

Most classical artificial neural networks are the famous multilayer perceptrons with their backpropagation learning algorithm [Rumelhart and McClelland, 1986]. Perceptrons are suitable for supervised classification within numerical high dimensional space, or statistical analysis related to pattern matching. We claim that the main question concerning the use of perceptrons for autonomous behavior is not the much debated biological plausibility of back-propagation but the relevance of supervised classification for autonomous behavior. From that point of view, unsupervised classification, provided by techniques like Self Organizing Maps [Kohonen, 1988] or Neural Gas [Fritzke, 1995] appears to be more suitable. Accordingly, classification with neural networks will provide sub-symbolic representations from numerical inputs.

Like classification, association is a well known capability of neural networks. Many adaptations of the Hebbian rule have been proposed, like the Widrow-Hoff rule. The latter is analog to the Rescorla-Wagner rule [Sutton and Barto, 1981] used to model human contingency judgment [Allan, 1993]. These association capabilities are the basis of regularity extraction from observations of the external world.

Every neural techniques mentioned above have been successfully applied to industrial problems, and mathematical theorems are available for most of them. Nevertheless, considering the biological framework they come from and our behavior modeling purpose, these methods suffer from a severe lack: they do not take the concept of time into account. Actually, behaving implies learning, perceiving, managing information from a continuous perceptive stream, and organizing them temporally. Consequently, the models described further will focused on these time features.

Although classical neural networks are slanted toward to engineering applications, few others aim at modeling

biological observations. Most of these observations are data from Pavlovian conditioning experiments, and the design of neurons is grounded on the following concepts. An unconditioned stimulus (UCS) is a stimulus that triggers a unconditioned response (UCR) before any conditioning occurs. Roughly, conditioning consist in presenting several times a conditioned stimulus (CS) *before* UCS, and then observe the conditioned response (CR) when CS alone is presented. CS is associated with UCS when CR looks like UCR. The association performed during conditioning experiments is the basis for causality and contingency evaluation, this evaluation depending on the association strength between two time-separated perceptions (the distinction between causality and contingency is discussed in section 4). The biological role of such temporal association between CS and UCS is to use CS as a predictor of UCS.

In [Sutton and Barto, 1981], a formal neuron is presented, whose inputs are several stimulus, considered as possible CSs, and a UCS. The UCS input is sufficient to make the neuron fire (high output), and the neuron associates (by increasing an appropriate weight) the earliest CS with UCS so that this CS fires the neuron before the occurrence of UCS. As non temporal associations are based on Hebbian-like mechanisms that correlate signals, temporal associations are provided by the same mechanisms but correlating *traces* of signals instead of signals themselves. A trace of a signal is a function of the signal that “slowly” decreases after the signal extinguishes, then keeping trace of the occurrence of the signal. The kind of temporal association performed by such mechanisms strongly depends on the shape of the trace. For example, in [Grossberg and Schmajuk, 1989], the trace function allows to select the CS that did occur during a given time interval before UCS, in [Frezza-Buet and Alexandre, 1999], the shape of the trace allows to select the closest predictor in time, etc. Correlating trace signals is a convenient way to detect correlations between time separated signals, and building sequences of correlated events with such neurons endows neural networks with the capability to deal with temporal events, causal relationships, as required by behavioral problems.

Clearly, even if most of neural techniques have been inspired from biology, very few of them deal with time relationships. Nevertheless, few neural models, closer to biology, provide robust and convenient tools for grounding temporal computation in numerical perceptions, but application to control architectures has still to be done on a large scale.

3 Temporal observations for temporal sequence learning

Several studies of the human cortex lead to a better understanding of mechanisms involved in cognitive tasks completion. Particularly, some of them can be understood at the unit level. Cortex is made of several elementary units : the cortical columns which are assemblies of neurons and have been extensively studied in [Burnod,

1989]. From a functional point of view (c.f. fig. 1), the cortical column can be seen as a cellular automaton which can be activated at three different levels of activity:

- Level E0 : neutral state
- Level E1 : calling state
- Level E2 : excited state

In a simple model of the posterior cortex [Burnod, 1989], each cellular automaton can be linked to either an elementary perception (e.g. red color perception in a particular retinal region) or an elementary action (e.g. muscle activation). The rules of transition are quite simple :

- When the automaton reaches the level of activity E1, this activity is spread to all neighbours, making them reach level E1.
- When the corresponding elementary perception occurs (context shift for example) and the automaton is currently at level E1, then the automaton reaches level E2.
- When the context allows the corresponding elementary action to be performed and the automaton is currently at level E1, the action is performed and the automaton reaches level E2.

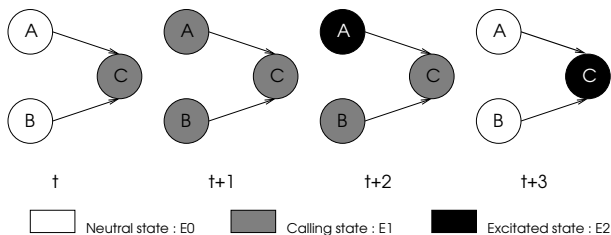


Figure 1: Functional view of columnar activations : At time t , the perceptive unit C is calling to be excited. At time $t + 1$, call has reached units A and B . At time $t + 2$, the context allows action A to be performed and the unit A becomes excited. At time $t + 3$, the context has been changed because of the previous action, and then unit C becomes excited. Here, learning occurs, and unit C will little by little preferentially call unit A .

An automaton will be able to learn as soon as it is excited at level E2 from level E1. Then, it will be memorized which automaton was excited (at level E2) just before it was (from activity traces), and the power of call toward this automaton will be reinforced. If an automaton is often excited before another, it may be deduced that the action performed or the perception received by the former is necessary for the excitation of the latter.

Since these columns are linked to elementary perceptions or actions, the various learned sequences are quite elementary and can be reused in different contexts, depending on the situation. The immediate consequence is that it allows the differentiation between delayed contingency and causality up to a certain point. If a sequence

is always learned within the same context and used in this very context, then nothing will differentiate delayed contingency and causality. Now, if there is a possibility to use a previous learned sequence in a different context then there is also a possibility to check whether causality was involved or not. The call mechanism associated to the local encoding of action and perception will indeed allow to test such hypothesis. For example, if it is observed that cocks sing just before the rise sun at 6 a.m., the following rule can be inferred : cocks singing make the sun to rise. If this rule is tested within the same context (i.e. at 6 a.m.), then differentiation between delayed contingency and causality is not possible. But, if the same rule can be tested within a different context (at 1 a.m. for example), then differentiation can occur. Making cocks sing (calling) at 1 a.m. (context shift) will not make the sun rise (excitation). Of course, if a rule remains true throughout all available contexts, the rule will be consequently considered as true.

The very structure of the column does not allow context encoding, but offers quite powerful generalization tools. Nevertheless, [Frezza-Buet and Alexandre, 1998] showed that when manipulating several sequences, there is a crucial need to take the existence of different possible contexts into account. Since the call propagation mechanism is not supervised, there is consequently no way of knowing in advance what will be performed.

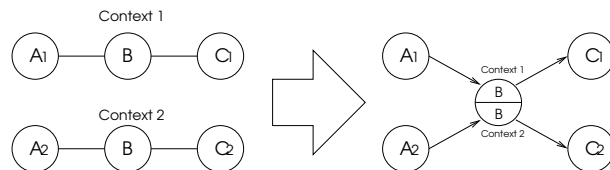


Figure 2: In this example, there exist two sequences using the same intermediate state B . The first state A_1 will shift the current context to $context_1$ while the second state A_2 will shift the current context to $context_2$. So, the action linked to B will lead to the state C_1 or C_2 depending on the context. The division of the column representing state B will help identifying the desired context.

For example, let us suppose there exist two sequences $[A_1 - B - C_1]$ and $[A_2 - B - C_2]$ within a model and state C_1 is called to be reached. Call will then be propagated up to state A_1 and A_2 . If both actions of state A_1 and state A_2 are possible within the current context, then, there is no way of knowing which of these two actions will be performed. While the action of state A_1 will lead to the right context for B to be performed ($context_1$), the other will lead to another context ($context_2$) where the performing of B action will lead to state C_2 . The state C_1 was called, state C_2 is reached. One possible solution offered by biology is to endow the column with a split capability (c.f. fig. 2) in order to be able to efficiently route calls towards their right destination.

Hence, biological facts concerning posterior cortex give serious hints for the design of distributed architectures successfully dealing with the time component of causality

relationships of the external world.

4 Temporal observations for spatial sequence learning

Until now, we were only concerned at direct encoding of what was observed in the environment in order to be able to predict consequences of an action. Moreover, with the will to encode world regularities, we were able to bypass cortical context consideration, leading this way to encode what is called procedural knowledge. But, among the large number of temporal sequences an animal is able to observe in its environment, some of them require a different kind of encoding. It is especially the case when dealing with cognitive maps construction.

Autonomous navigation is among the most crucial abilities required for mammals to survive in their environment and one way to complete this task is to construct an internal cognitive map of the environment, thus requiring the spatialization of temporal observations. When being in a locus A and going to a locus B, the path leading from A to B has to be learned somehow and in this framework, this spatial information has to be considered in order to survive, despite the fact that the path is perceived within a temporal sequence (being in locus A then walking ahead then being in locus B). Here, we are no more interested in procedural knowledge (which could have been for example when walking ahead, objects which were far from me become nearer) but in episodic knowledge.

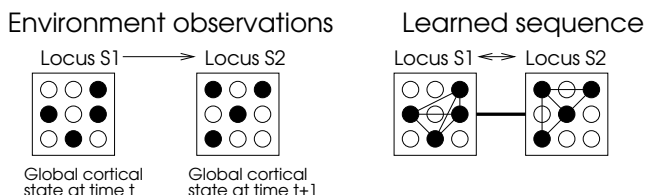


Figure 3: Spatial sequence construction : One of the temporal sequences observed is : when perceiving the global cortical state S_1 and performing the action A_1 at time t , the global cortical state S_2 is observed at time $t+1$. The corresponding internal representation is : state S_1 is linked to state S_2 . The time dimension is not explicitly encoded.

We want to memorize that locus $L1$ which is perceived as a global cortical state (i.e. excitation distribution over the cortex) S_1 is linked to locus $L2$ which is also perceived as a cortical state S_2 . The process of learning will then be quite different from the previous one because the time dimension has to be greatly reduced during the encoding process and will indeed become implicit. Linking a locus A to a locus B is a way to say that there is a possibility when being in locus A to go in locus B, and when being in locus B there is a possibility to go in locus A. Consequently, it implicitly encodes that there is no way to be simultaneously in loci A and B.

Moreover, the sequence learning is performed at the entire cortical level because of the distributed representation of locus within the model. But, since we are

now interested in explicit representation of global internal state and context, there is consequently a need for a working memory in order to actively recall previously learned states and link them to the current ones. Traces of units activities are no more sufficient for linking different perceived states. We then need to be able to instantiate a previously learned global internal state (i.e. exciting corresponding cortex units). One of the solutions offered by biology is to use a specific structure for storing and retrieving these previously learned states, hence allowing sort of context instantiation. This is one of the possible role attributed to the hippocampus in the human brain [Hirsh, 1974; McClelland *et al.*, 1994; McNaughton and Nadel, 1990; Squire *et al.*, 1984]. This working memory is able to recall a previously learned locus (corresponding to a cortical multimodal representation of a locus) while perceiving another locus, allowing this way a reciprocal linking between the two representations. Moreover, the fact of explicitly taking into account the cortical context allows fine sequence learning such as :

- $A \rightarrow C$
- $B \rightarrow C$
- $\neg(A \wedge B \rightarrow C)$

If the rule $A \rightarrow C$ is learned locally, nothing will prevent model from inferring : $A \wedge B \rightarrow C$ whereas the explicit context encoding will encode $A \wedge B$ as a new item and manipulate it regardless of A and B. Explicit learning indeed allows the differentiation of “ambiguous” sequences when procedural learning is unable.

5 Control sequences: action oriented representations

Neural mechanisms for sequence processing presented above endow living beings with high representation capabilities, concerning causal rules of the world and its topography. Nevertheless, in the case of superior living beings, behaving is much more than exploiting causality relationships in an reactive “Pavlovian” way. Actually, many animals are sensitive to reward: they are able to organize action in order to avoid pain and maximize pleasure. Concerning superior mammals, and especially human beings, the prefrontal cortex (PFC) seems to be the neural substrate of the management of action, when action is goal directed [Fuster, 1996]. Therefore, current studies and models concerning neurons in the PFC can give hints for the design of an artificial neural action manager.

Although functions of the prefrontal cortex are numerous, we only insist in this paper on the *provisional memory* and its implications on temporal management of action [Fuster, 1996]. Many neurons in the PFC are able to sustain their activity for a relatively long time. This sustained activity is supposed to provisionally store a current goal when subgoals are activated. For illustration (c.f. fig. 4), let us suppose that the plan consists in performing a sequence of two tasks $A - B$. Let task

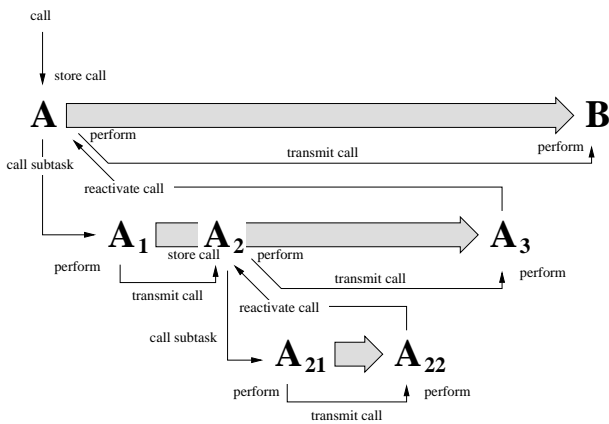


Figure 4: Stacking mechanism of frontal units. See section 5 for comments

A require the sequence of three subtasks $A_1 - A_2 - A_3$, and A_2 the sequence of subsubtasks $A_{21} - A_{22}$. Let us then suppose that a set of frontal units corresponds to each of the A, B, A_i, A_{ij} . Initially, the first unit A is called but it cannot be performed. This call is then stored in A as a *sustained* activity, and A calls A_1 unit. When A_1 is performed (for example the call in A_1 has been transmitted to posterior associative cortex that actually performed the subtask A_1), A_1 calls A_2 . As A_2 requires the subsubtask A_{21} , the call in A_2 is stored in unit A_2 as it was for A , and A_2 transmits call to A_{21} . Two calls are stored respectively with sustained activity of A and A_2 . These activities enable the PFC to *stack* the current goal when subgoals are required. When A_{21} is performed, A_{22} is called and then performed. The achievement of A_{22} reactivates the call stored in A_2 and A_2 is performed. The sustained activity in A_2 then stops, subtask A_2 is *popped* from the goal stack. The achievement of A_2 make A_2 transmit the call to A_3 . When A_3 is performed, the call stored in A is reactivated, A is performed, and B is called and performed. This simplified example (see [Burnod, 1989]) gives the basis for neural hierarchical sequence management, with stacking properties, in order to deal with plans. Such sustained activities are provisional memory, coding the planning context. This model has been used to model prefrontal neuron activities in the monkey's cortex during go-no go tasks [Guigon *et al.*, 1995].

The learning mechanisms that enable to connect PFC neurons so that they provide the correct plan hierarchy are not straightforward, but recent biological models [Guigon *et al.*, 1995] give hints on the way reward can control this learning.

6 Discussion

Neurobiology and neuropsychology provide many hints concerning the central role of time in living being behavior and we aimed at showing a possible bridge linking these two domains to computer sciences. Indeed, these two domains give precious hints concerning the design

of control architecture dealing with sequences of events, and we illustrated the computational power of three mechanisms, respectively inspired from posterior cortex, hippocampus and frontal cortex studies. These mechanisms allow the construction and exploitation of causal sequences observed in the environment, in order to build topographic cognitive maps and to control sequences of successive goals. Nevertheless, these three aspects of sequence processing are far from being completely understood because, in the framework of autonomous agent design, they have to be completely specified in a robust and flexible way to allow their integration within a complete system including such features as distributed representation, learning transfer or generalization. Moreover, the whole background of motivational aspects are not really considered when as psychology emphasizes the crucial role of motivation in mammals behavior, especially concerning the paradigm of action selection. To sum up, sequence processing mechanisms clearly need to be better studied and integrated in autonomous behavior systems, but they are not the ultimate step toward cognition understanding. Consequently, the studies must not be driven alone but within the global and complex framework of biology modeling.

Nevertheless, in spite of these limitations, the strategy of first observing biological mechanisms to translate them in computational models seems fruitful. Foundations of connectionism lie on such a strategy. The main point is that we are now able to integrate mechanisms coming from a unified and consistent system : the brain. Without considering higher level feature such as emotions, consciousness or soul, we know in advance that, at the unit and map level, the system is consistent and then, if the translation is properly done, we will be able to design consistent models. Such a model is being under construction for endowing an animat with autonomous behavior.

References

- [Allan, 1993] Lorraine G. Allan. Humain contingency judgments : Rule based or associative. *Psychological Bulletin*, 114(3):435–448, 1993.
- [Burnod, 1989] Y. Burnod. *An adaptive neural network the cerebral cortex*. Masson, 1989.
- [Donnart and Meyer, 1994] Jean-Yves Donnart and Jean-Arcady Meyer. A hierarchical classifier system implementing a motivationnaly autonomous animat. In *Proceedings of the 3rd Int. Conf. on Simulation of Adaptive Behavior*, pages 144–153. The MIT Press/Bradford Books, 1994.
- [Frezza-Buet and Alexandre, 1998] Hervé Frezza-Buet and Frédéric Alexandre. Selection of action with a cortically-inspired model. In *Seventh European Workshop on Learning Robots*, pages 13–21, 1998.
- [Frezza-Buet and Alexandre, 1999] Hervé Frezza-Buet and Frédéric Alexandre. Specialization within cortical models: An application to causality learning. 7th

- European Symposium on Artificial Neural Networks, 1999. To appear.
- [Fritzke, 1995] B. Fritzke. A growing neural gas network learns topologies. In G. Tesauero, D. S. Touretzky, and T. K. Leen, editors, *Advances in Neural Information Processing Systems 7*, pages 625–632. MIT Press, Cambridge MA, 1995.
- [Fuster, 1996] J.M. Fuster. Frontal lobe and the cognitive foundation of behavioral action. In A.R. Damasio, H. Damasio, and Y. Christen, editors, *Neurobiology of Decision-Making*. Springer, 1996.
- [Grossberg and Schmajuk, 1989] Stephen Grossberg and Nestor A. Schmajuk. Neural dynamics of adaptive timing and temporal discrimination during associative learning. *Neural Network*, 2:79–102, 1989.
- [Guigon *et al.*, 1995] E. Guigon, B. Dorizzi, Y. Burnod, and W. Schultz. Neural correlates of learning in the prefrontal cortex of the monkey: A predictive model. *Cerebral Cortex*, 5(2):135–147, 1995.
- [Hirsh, 1974] R. Hirsh. The hippocampus and contextual retrieval of information from memory: A theory. *Behavioral Biology*, 12:421–444, 1974.
- [Jaakkola *et al.*, 1994] Tommi Jaakkola, Michael I. Jordan, and Satinder P. Singh. On the convergence of stochastic iterative dynamic programming algorithms. *Neural Network*, 6:1185–1201, 1994.
- [Kohonen, 1988] Teuvo Kohonen. *Self-Organization and Associative Memory*. Springer-Verlag, 1988.
- [Maes *et al.*, 1996] Pattie Maes, Maja J. Mataric, Jean-Arcady Meyer, Jordan Pollack, and Stewart W. Wilson, editors. *From animals to animat 4 : Proceedings of the Fourth International Conference on Simulation of Adaptive Behavior*. The MIT Press, 1996.
- [McClelland *et al.*, 1994] J. L. McClelland, B. L. McNaughton, and R. C. O’Reilly. Why there are complementary learning systems in the hippocampus and neocortex: Insights from the successes and failures of connectionists models of learning and memory. Technical report, Carnegie Mellon University and The University of Arizona, March 1994.
- [McNaughton and Nadel, 1990] B. L. McNaughton and L. Nadel. *Neuroscience and connectionist theory*, chapter 1 : Hebb-Marr networks and the neurobiological representation of action in space, pages 1–63. M. Gluck and Rumelheart, Hillsdale, erlbaum edition, 1990.
- [Rumelhart and McClelland, 1986] D. E. Rumelhart and J. L. McClelland. *Parallel Distributed Processing: Exploration in the Microstructure of Cognition*. Cambridge, MA: MIT Press, 1986.
- [Squire *et al.*, 1984] L.R. Squire, N. J. Cohen, and L. Nadel. *Memory consolidation*, chapter The medial temporal region and memory consolidation : A new hypothesis, pages 185–210. H. Weingartner and E. Parker, erlbaum edition, 1984.
- [Sutton and Barto, 1981] Richard S. Sutton and Andrew G. Barto. Toward a modern theory of adaptive network : Expectation and prediction. *Psychological Review*, 88(2):135–170, 1981.
- [Watkins, 1989] C. J. C. H. Watkins. *Learning from delayed rewards*. PhD thesis, University of Cambridge, 1989.