

Towards an on-line neural conditioning model for mobile robots

Erol Şahin

Starlab Research Laboratories
Latour De Freins, Rue Engelandstraat 555, 1180 Brussels, Belgium
erol@starlab.net
<http://www.starlab.org/>

Abstract. This paper presents a neural conditioning model for on-line learning of behaviors on mobile robots. The model is based on Grossberg's neural model of conditioning as recently implemented by Chang and Gaudiano. It attempts to tackle some of the limitations of the original model by (1) using a temporal difference of the reinforcement to drive learning, (2) adding eligibility trace mechanisms to dissociate behavior generation from learning, (3) automatically categorizing sensor readings and (4) bootstrapping the learning process through the use of unconditioned responses. Preliminary results of the model that learn simple behaviors on a mobile robot simulator are presented.

1 Introduction

Mobile robots provide a challenging testbed for the development of neural models of learning. Neural models that are designed to learn using pre-recorded data sets or in unrealistic simulations, usually fail on mobile robots. Most of these models are plagued with implicit assumptions that are revealed when they are faced with the constraints of the real world. On-line learning is a fundamental constraint for learning algorithms on mobile robots. Once implemented on a robot, the model has to process a continuous stream of noisy data both for learning and testing in real time.

Animal learning research have unveiled two basic forms of learning that allows animals to recognize the informative cues in their environment and take actions that increase their survival chance: Classical and operant conditioning. The classical conditioning can be illustrated by the following experiment: A hungry dog presented with food salivates. However, it does not salivate when a bell rings. Then the bell is rung prior to the presentation of food during several learning trials. After this, ringing of the bell alone yields salivation. Hence, classical conditioning enables the animal to recognize relevant stimuli in the environment. Here, food is called the *unconditioned stimulus* (UCS), salivation is called the *unconditioned response* (UCR), and the bell is called the *conditioned stimulus*. In operant conditioning, an animal learns to suppress behaviors leading to punishment, and exhibit behaviors leading to rewards.

2 Grossberg's neural conditioning model

In 1971, Grossberg [2] designed a detailed neural model to account for classical and operand conditioning in animals. By going through a thought experiment, Grossberg first set out the constraints, which he called psychological postulates, to be satisfied and then designed a minimal neural model to satisfy these. This model was then refined and studied in detail in later studies [3][4].

Recently Chang and Gaudiano [1] successfully implemented Grossberg's neural conditioning model to learn to generate avoidance/approach behaviors on two different robot platforms. This work follows up their work. Now we will first point out and discuss the limitations of their model and then propose a new model to tackle some of them. Note that some of these limitations are specific to Chang and Gaudiano's implementation whereas others are rooted in Grossberg's model.

- The model does not have a predictive nature. At a first glance, their results suggest that the model has a predictive nature. In the obstacle avoidance experiment, the model seemed to predict oncoming collisions with obstacles and avoid them by suppressing harmful behaviors in advance. However the model has no mechanism of making predictions since it could only learn the sensory cues that occur at the time of the collision *not* prior to the collision. This pseudo-predictive ability relies on the implicit assumption that the activation of sensory cues will get larger as the robot gets closer to a collision.

Note that, this limitation is specific to this implementation. Grossberg [3] had argued that the short term memory mechanism for the sensory cues would be able to keep the activity that occurred prior to the collision for learning. However, the mechanism that was proposed by Grossberg relies heavily on the dynamics of short term memory, and was omitted by Chang and Gaudiano.

- The model did not have a mechanism to create sensory cues (i.e. *CS*'s) from raw sensor readings. Although raw sensor readings had sufficed for their experiments, the model needs to be extended to create sensory cues. Discovery of sensory cues should be done with feedback from the conditioning system.
- The model did not use a bootstrapping mechanism for learning. Motor commands were generated at random. Unconditioned response mechanism could not only not only naturally replace the random activation mechanism, but can also have the model learn on-line by ensuring that unconditioned responses would get take the robot out of bad states.

The following section describes the proposed model which tackles some of the problems mentioned above.

3 An on-line neural model of conditioning

In this section we outline the neural model that we propose to tackle the limitations of the original model.

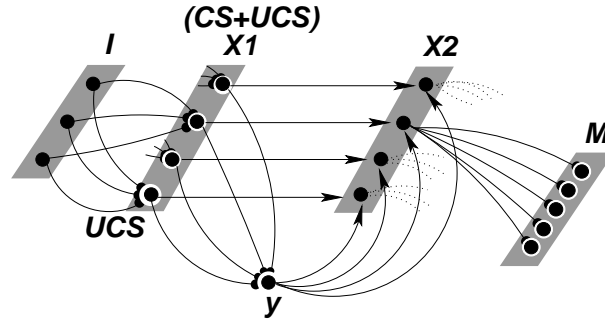


Fig. 1. Schematic description of the model.

The signals that causes activation and learning needs to be dissociated. Although this problem is not explicitly addressed in Grossberg's model, it is a well-known one and has been addressed by other conditioning models [6, 5]. A different signalling mechanism, often called *eligibility traces*, needs to be used for enabling learning. Such a mechanism can sustain prior information about the activity of a neuron, without interfering with ongoing behavior generation. The cellular mechanisms that may implement such a mechanism in biological neurons through long lasting pre-synaptic changes are discussed in [6]

3.1 Behavior Generation

Figure 1 illustrates the structure of the neural model. The leftmost layer, I , indicates the sensory neurons. This layer is connected to the X_1 layer through weighted connections, Z_I . The neurons at the X_1 layer are divided into two groups: CS and UCS . The neurons in the CS group have adaptive weights and learn the prototypes of sensory patterns creating *learned sensory cues*. The UCS neurons represent the *innate* collision detectors. The activation of the neurons at this layer are given by

$$x_{1j}(t) = f_{CS/UCS}(\sum_i I_i(t)z_{X_j}(t)). \quad (1)$$

Here f_{CS} is a sigmoid function that bounds the activation of the CS neurons between 0 and 1. f_{UCS} is a hard threshold function that turns on when the activity of the corresponding input rises above the threshold signalling a collision.

The winner of X_1 layer, J , is then determined by $\max(\{x_{1j}\}, v)$, where v is the vigilance parameter that sets the minimum level of neuron activity to win. Such a winner take all mechanism creates a mutually exclusive generation of behaviors.

The reinforcement neuron (named as the drive neuron by Grossberg), Y represents the reinforcement signal. The neuron is activated by the winner of the X_1 layer weighted by Z_y as

$$Y(t) = x_{1J}(t)z_Y(t) \quad (2)$$

If there are no winners at X_1 then Y is set to 0. Y energizes the learning through the rest of the model but does not affect the behavior generation.

At layer X_2 are the inputs from the X_1 layer and the Y converge. However this conversion is for learning only. During behavior generation, the winner take all activity of X_1 is merely copied into X_2 .

Finally, the layer at the far right, M , represent the motor layer that control the robot. A one dimensional layer of neurons represent different angular velocities for the robot. The most active node generates a turn that it corresponds to. For instance, activating the leftmost node would generate a full left turn, whereas activation of the center node would move the robot straight ahead. A positive Gaussian shaped activation is placed at the center of this of the layer drives the robot straight ahead in the absence of any obstacles. The activation of neurons at this layer are as

$$m_l(t) = \exp -(l - n)^2 / \sigma^2 + \sum_k x_{2k}(t) z_{ml}(t). \quad (3)$$

Here the first exponential term represents the Gaussian placed at the center of the layer (i.e. $n = 0$). The position of this Gaussian can be set to drive the robot to certain target. The second term is the summed inhibition caused through X_2 activation. When an obstacle is detected by the UCS or the CS neurons, the corresponding neuron at X_2 will become active and suppress the angular velocity nodes that may cause a collision.

3.2 Learning

Learning occurs when the change in the reinforcement Y is above a certain threshold. The reinforcement node releases an eligibility signal

$$Y^e(t) = Y(t) - Y(t - 1) \quad (4)$$

that enables learning throughout the model. The idea of using the time derivative of the reinforcement signal is central to whole class of conditioning models, for a review see [7])

The eligibility signal generated by the reinforcement node causes learning in three places. First, it is used for automatic categorization of sensory inputs into sensory cues at X_1 . Initially all the CS nodes are *uncommitted* with all their incoming weights set to zero. If the eligibility signal received from Y is larger than a threshold (0.5), the eligibility of the X_1 neurons, $x_j^e(t) = x_j(t - 1)$. are checked. If the the activation of the maximally eligible neuron (a CS node) is below the vigilance, v , then the next uncommitted neuron is employed. Incoming weights of the neuron are set to the normalized copy of the $I^e(t) = I(t - 1)$. Here the eligibility mechanism is used for storing prior activations of the neurons, and learning them forms the basis for the predictive nature of the proposed mode. The normalization of weights during learning ensures that the closest CS node be maximally activated when a similar input comes. If the activation of the maximally eligible neuron is above the vigilance, no learning takes place.

Second, the weight of the maximally eligible X_1 neuron to Y is updated as

$$z_J(t) = z_J(t-1) + \gamma_1(\lambda Y - z_J(t-1)) \quad (5)$$

where γ_1 is the learning rate and $\lambda = 0.9$ is the discount rate of reinforcement in time.

Finally, the weights from X_2 to M are updated as follows to suppress the maximally active motor neuron:

$$z_{mkl}(t) = z_{mkl}(t-1) - \gamma_2 x_2^e(t) f_m(L) \quad (6)$$

where γ_2 is the learning rate, $f_m()$ is the suppression function that linearly suppresses the nodes from the L , the index of the maximally eligible node, to the center node.

4 Experimental Results

The model described above is implemented on the Webots mobile robot simulator (Cyberbotics SA, Switzerland) to control a simulated Khepera robot (K-team SA, Switzerland). The robot, shown in Figure 2-(a) is a miniature differential-drive robot with eight infrared proximity sensors. Only the six frontal sensors are used for this experiment. The two rear-facing infrareds are ignored.

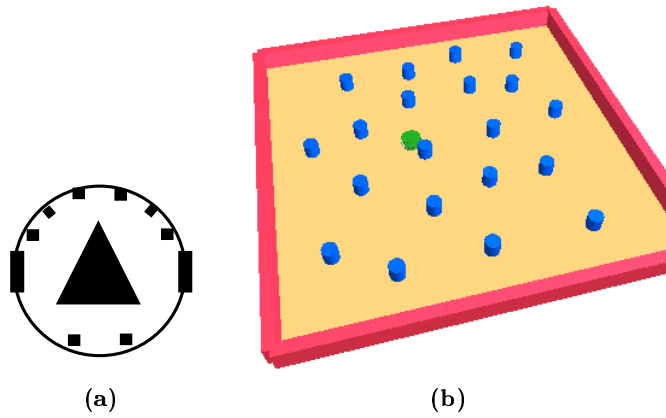


Fig. 2. (a) Top view of the Khepera, showing the placement of the infrared proximity sensors. (b) The environment of the simulated Khepera to learn obstacle avoidance.

Figure 2-(b) shows the environment of the simulated Khepera robot. The robot is left freely in this environment and no forms of supervision is done.

A gaussian activation centered at the motor map causes the robot to move in straight line unless an obstacle gets in its way. In this experiment six *UCS* neurons are used: each one detecting the collision on one of the sensors. The connection weights from the sensor layer are shown as bar graphs on the left side of Figure 3. The first three of the *UCS* nodes causes a full left turn as their *UCR*, whereas the remaining three causes a full right turn. The *UCS* – *UCR* connections are the only innate connections in the model. Yet, the innate behaviors they produce are sufficient for bootstrapping the learning process. The weights of the first six sensory cues that were learned during the experiment are shown on the right side of Figure 3. It can be seen that they tend to be separate well in the sensory space and correspond to the sensory patterns that are prior to collisions.

Figure 4 plots the distance of the robot to the closest object with respect to time during learning. It can easily be seen that at the beginning, the robot tend to get closer to the objects, avoiding them through its innate *UCR* behaviors. As the learning process creates more predictive sensory cues and learns to activate the right avoidance behaviors, the robot is able to predict oncoming collisions and avoid the obstacles without getting too close.

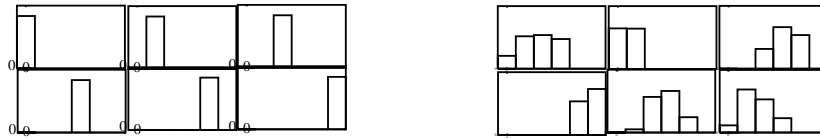


Fig. 3. The connections weights from *I* layer to the first twelve nodes of X_1 layer is shown. On the left, six weight vectors show the innate connection weights to the six *UCS* nodes. On the right, first six weight vectors of the *CS* nodes are shown.

5 Conclusions

We believe that neural models of learning have to run on-line on mobile robots. This requires that not only the core learning problem, but also seemingly peripheral problems like the sensory categorization, bootstrapping needs to be addressed. Grossberg's neural conditioning model creates a nice framework to study this. Although reinforcement learning, which was initiated through animal learning studies, provides a better computational analysis, we believe that a neural implementation of the same ideas can provide more insight to the problem.

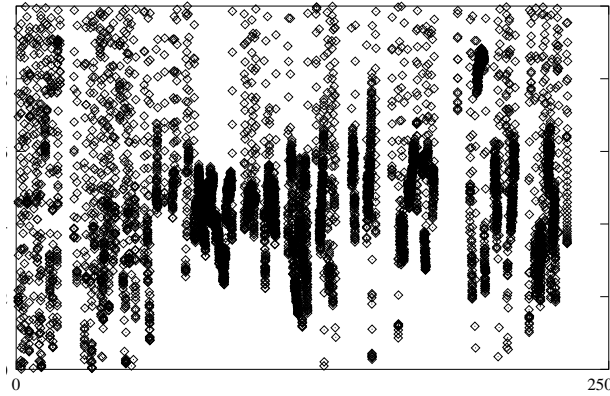


Fig. 4. The distance to the closest object as a function of time throughout learning.

References

1. Chang, C., Gaudiano, P.: Application of biological learning theories to mobile robot avoidance and approach behaviors. *Journal of Complex Systems*, **1**(1) (1998), 79–114.
2. Grossberg, S.: On the dynamics of operant conditioning. *Journal of Theoretical Biology*, **33**, (1971) 225–255.
3. Grossberg, S., Levine, D.: Neural dynamics of attentionally modulated Pavlovian conditioning: blocking, interstimulus interval, and secondary reinforcement. *Applied Optics*, **26**, (1987) 5015–5030.
4. Grossberg, S., Schmajuk, N. A.: A neural network architecture for attentionally-modulated Pavlovian conditioning: Conditioned reinforcement, inhibition and opponent processing. *Psychobiology*, **15**, (1987) 195–240.
5. Klopff, A. H.: *The Hedonistic Neuron: A theory of memory, learning and intelligence*. Hemisphere, Washington, D. C., (1982).
6. Sutton, R. S., Barto, A. G.: Toward a modern theory of adaptive networks: Expectation and prediction *Psychological Review*, **88**, (1981) 135–170.
7. Sutton, R. S., Barto, A. G.: A temporal-difference model of classical conditioning In *Proceedings of the Ninth Conference of the Cognitive Science Society Hillsdale, NJ*. Lawrence Erlbaum Associates, (1987).