Learning Environmental Contexts in a Goal-Seeking Neural Network

Thomas E. Portegys
School of Information Technology
Illinois State University
Campus Box 5150, Normal, Illinois 61790, USA
portegys@ilstu.edu

Abstract

An important function of many organisms is the ability to learn contextual information in order to increase the probability of achieving goals. For example, a cat may watch a particular mouse hole where she has experienced success in catching mice in preference to other similar holes. Or a person will improve his chances of getting back into his house by taking his keys with him. In this paper, predisposing conditions that affect future outcomes are referred to as environmental contexts. These conditional probabilities are learned by a goal-seeking neural network. Environmental contexts of varying complexities are generated that contain conditional state-transition probabilities such that the probability of some transitions is affected by the completion of others. The neural network is capable of expressing responses that allow it to navigate the environment in order to reach a goal. The goal-seeking effectiveness of the neural network in a variety of environmental complexities is measured.

Key Words:

Connectionism, context learning, goal-seeking, neural networks.

1 INTRODUCTION

Context learning is an important function for many organisms, especially for humans. Behavior that is socially accepted at a sporting event will not be welcome in a classroom setting. The odds of obtaining a cookie from my Grandmother's cookie jar may far exceed those of finding one in my cookie jar, despite that we have identical jars. These are examples of the significance of context: making certain arrangements in one's environmental state prior to attempting an act can have a great deal to do with the outcome. Many animals are capable of learning context from their environment, as well as being taught by a conditioning process known as behavior shaping (Carpenter 1974).

There are many references to general context learning in the literature (Bonzon, P. 1997; Schank and Childers, 1984; Turner, 1998). Various specialized approaches also exist. For example, context learning has been described as hierarchical sequence learning by Sun and Giles (2001). Researchers have also proposed context models of brain and behavior such as Howard and Kahana's Temporal Context Model (TCM) of the recency and continguity memory effects (2002), and Hasselmo and McClellands' model of the hippocampus' role in memory formation (1999). In the robotics field, Brooks and Maes trained a robot operated by a hierarchy of control

contexts to walk by using environmental feedback (1990). For non-symbolic learning, mathematical methods have been developed to optimize reinforcement produced by an environmenal context function (Sabes and Jordan, 1996).

The subject of context learning narrows considerably as an application of artificial neural networks. Perhaps some of the most relevant work is in the field of grammar learning (Bodén and Wiles, 2002; Steijvers and Grunwald, 1996) and text classification (Wermter, Arevian, and Panchev, 1999) using recurrent and cascading neural networks. In the grammar learning studies, neural networks are trained to recognize sequences of inputs produced by a grammar, and are later tested on their predictive performance given incomplete sequences. The neural network plays a passive recognition role in these experiments. In our study, the aim is to allow the neural network to take an active part in the learning process by producing responses that affect state-transition probabilities. The predisposing conditions that affect future outcomes are referred to as environmental contexts. Learning these contexts allows the neural network to navigate its environment to reach a goal state. The goal-seeking effectiveness of the neural network in a variety of environmental complexities is measured.

The purpose of this project is to develop and test a learning method suitable for a goal-seeking neural network called Mona. Although a connectionist architecture, Mona is more of a state-based planning system that a conventional pattern classifying neural network. Planners (e.g. Benson and Nilsson, 1995) are typically symbolic, not connectionistic systems, necessitating a novel learning solution for Mona. Mona has modeled complex behavior on a number of tasks, including foraging and cooperative nest-building (Portegys, 1999 & 2001). See www.itk.ilstu.edu/faculty/portegys/programs/NestViewer/NestViewer.html for an exhibit of the nest-building task. Mona features an integrated motivation mechanism designed to produce responses that yield need-reducing outcomes. It is currently being taught to learn probabilistically generated mazes. This study represents a milestone in a continuing program of development for Mona.

1.1 A Review of Mona

Mona is a model based on the rationale that brains are goal-seeking neural networks. It has a simple interface with the environment, shown in Figure 1. All knowledge of the state of the environment is absorbed through “senses”. Responses are expressed to the environment with the goal of eliciting sensory inputs which are internally associated with the reduction of needs.

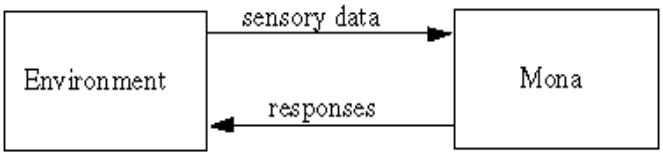

Figure 1 – Mona/Environment Interface

Events can be drawn from sensors, responses, or the states of component neurons, calling for three types of neurons. Neurons attuned to sensors are receptors, those associated with responses are motors, and those mediating other neurons are mediators. Mediators can be structured in hierarchies representing environmental contexts. A mediator neuron controls the transmission of need through and the enablement of its component neurons.

To elucidate by example, consider this task: Mona must get into her home from somewhere out in the world, a locked door barring the way inside, thus necessitating the use of a key to unlock the door. She needs to know several things, such as how to get to the door, how to unlock the door, and how to enter her home through the unlocked door. Mona must produce a sequence of responses to proceed from an initial keyless condition in the world to her home.

Figure 2 depicts the portion of Mona's neural network which manages the entering of home through an unlocked door. Let the house-shaped objects be receptor neurons, such as the one marked "Door"; the inverted houses be motor neurons, such as "Move"; and the diamonds be mediator neurons, such as "Enter home". The numbers in parentheses indicate need levels, which will be discussed later; suffice it to say for now that the "Home" receptor has been associated with the reduction of a need, and is thus a goal for Mona. The numbered arrows proceeding from a mediator indicate a sequence of neurons mediated by it. In this case, "Enter home" mediates a sequence of events associated with the receptor "Door", the motor "Move", and the receptor "Home". This mediator thus governs the process of entering home by moving through a door. The type of mediation exerted by "Enter home" is an enabling one, meaning that it allows firing events to propagate enabling influences.

Initially the door is locked, thus the "Enter home" mediator is disabled, meaning that it cannot function until preconditions establish an enabling context for it. This is represented by the dotted outline of the mediator. In order to enable "Enter home", another mediator must come into play: "Enable enter home". This mediator will enable the "Enter home" neuron when the "Unlock door" neuron fires.

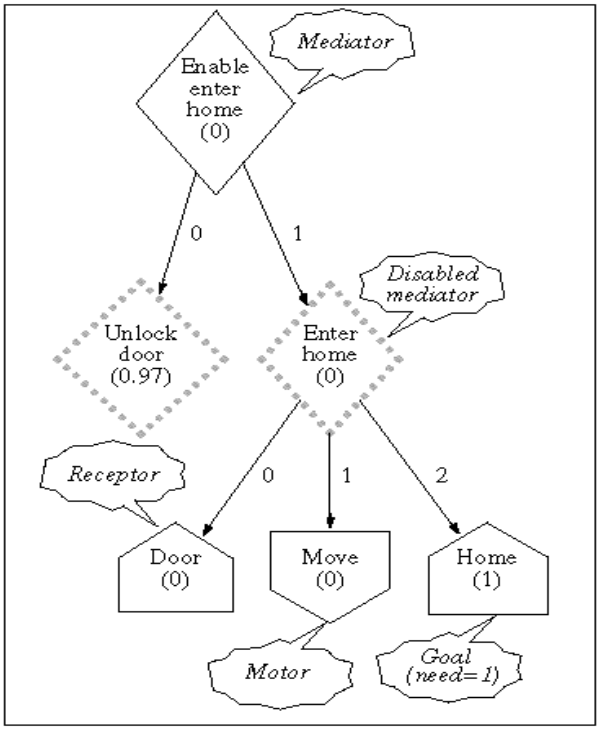

Figure 2 – Enable enter home/Enter home Mediators

However, the "Unlock door" neuron is also in a disabled state, requiring "Get key", shown in Figure 3, to fire as a precondition: the door cannot be unlocked without the key.

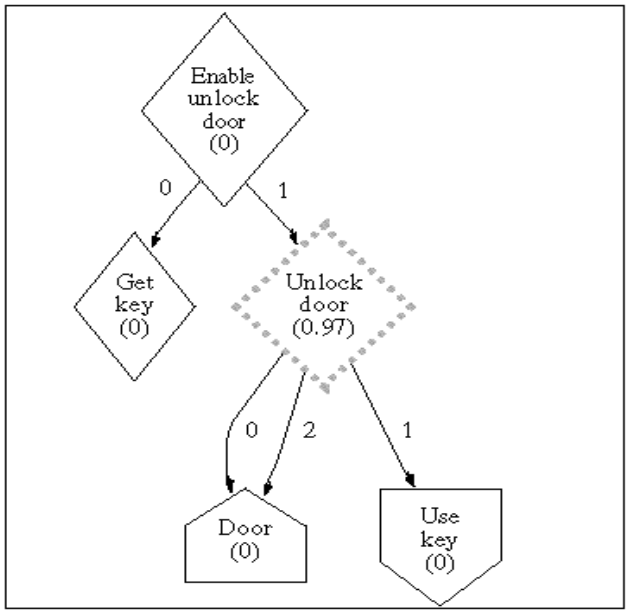

Figure 3 – Enable unlock door/Enable door

The final two pieces are supplied in Figure 4: how to get a key ("Get key"), and how to get to the door from the world ("Go to door").

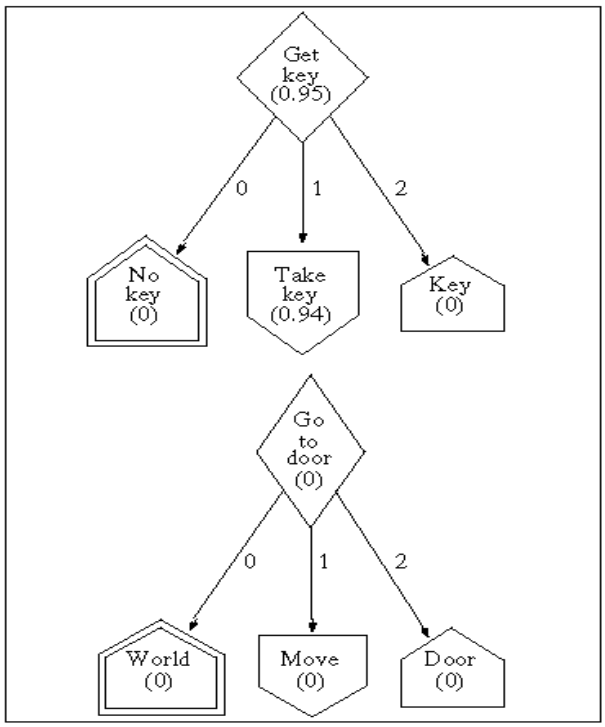

Figure 4 – Get key/Go to door

Since these diagrams show the initial state of network, the "World" and "No key" receptors are firing, denoted by the double outlines on their graphical symbols.

Recognizing environmental context is only part of the task addressed by Mona; needs emanating from goals require a control mechanism to transform them into appropriate responses. The networks that follow contrain mediators to manage two neurons: a "cause" and an "effect". For example, Figure 5 shows a situation in which Mediator2 must become enabled in order to achieve the goal. Need in this case will flow from the Receptor goal into Motor1, Mediator1's cause, firing its associated response. This results in Mediator2 becoming enabled, as shown in Figure 6. Need then flows into Motor2, firing its response, resulting in the subsequent firing of the goal Receptor.

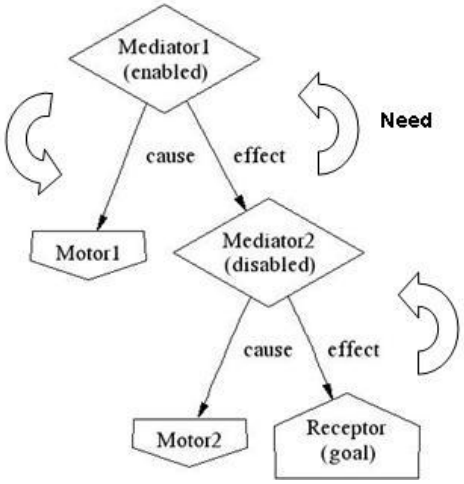

Figure 5 – Enabling Mediator2

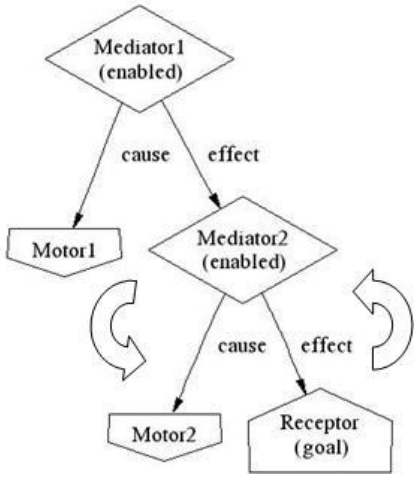

Figure 6 – Mediator2 enabled

2 DESCRIPTION

2.1 Environment

The purpose of the environment for this project is to allow the proposed learning techinque to be verified. An environment is randomly generated given a complexity parameter defining the number of contexts it contains. A context defines the probability of producing an effect given a cause. Causes and effects are recursively defined as either stimuli to the learner or other contexts:

<context> ::= <cause>< probability><effect>
<cause> :: = <context> | <stimulus>
<effect> ::= <context> | <stimulus>

A network of contexts is generated as follows: an initial goal stimulus is created and directed to produce the desired number of contexts surrounding it. This number is referred to as a branching value. A stimulus may create a context by (1) changing into a context, spawning a pair of cause/effect stimuli in the process, and/or (2) creating one or more parent contexts for which the current node is the effect. The branching value of each created node is determined by randomly dividing the current node's branching value less the branching that it has generated itself. As

each context is generated, a random probability ranging from 1.0 to -1.0 is assigned to it. A negative probability is an artificial quantity used in the accumulation process described below. There is one restriction to force the neural network to learn individual cause and effect probabilities: a cause stimulus can only branch by changing into a context.

Figure 7 shows a simple network of three contexts, denoted by the diamond shapes. A unique number and associated probability is also shown for each. Stimulus 0 is the goal. This network was generated by first branching Context 4 with a cause that subsequently changed into Context 1. Context 4 also branched as the effect for Context 6.

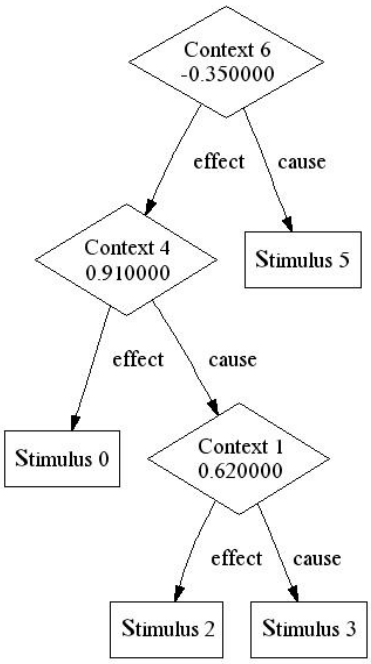

Figure 7 – Context Network

A stimulus can be in a "ready" or "fire" state, depending on whether it has been presented to the learner (fire) or not (ready). A context has three states: "ready", "set", and "fire". Initially ready, a context transitions to the set state when its cause fires. When its effect subsequently fires, the context fires. Probabilities accumulate as follows. When a context enters the set state, its probability propagates, accumulating on its effect node. If its effect is also a context in the set state, the accumulation continues. Thus in Figure 7, firing Stimulus 5 would have the effect of moving Context 6 into the set state, which in turn would cause its -.35 probability to accumulate at Context 4, making its probability .56. In order to set Context 4, Context 1 must fire, which turn requires that Stimulus 3 and 2 must fire. The rationale behind the accumulating probabilities is to provide a means for contexts to affect the outcome of responses by the learner. A negative or zero probability results in no chance of a successful achievement of an effect, while a one or greater probability results in a certain chance.

It is possible to generate environments that are complex and have improbable goal states. Figure 8 is an example of an environment with eight contexts that contains an unlikely path to Stimulus 0. The path is: Context 2 (.19), Context 9 (to increase Context 1 to .095), and Context 14 (.465). The combined probability is .005.

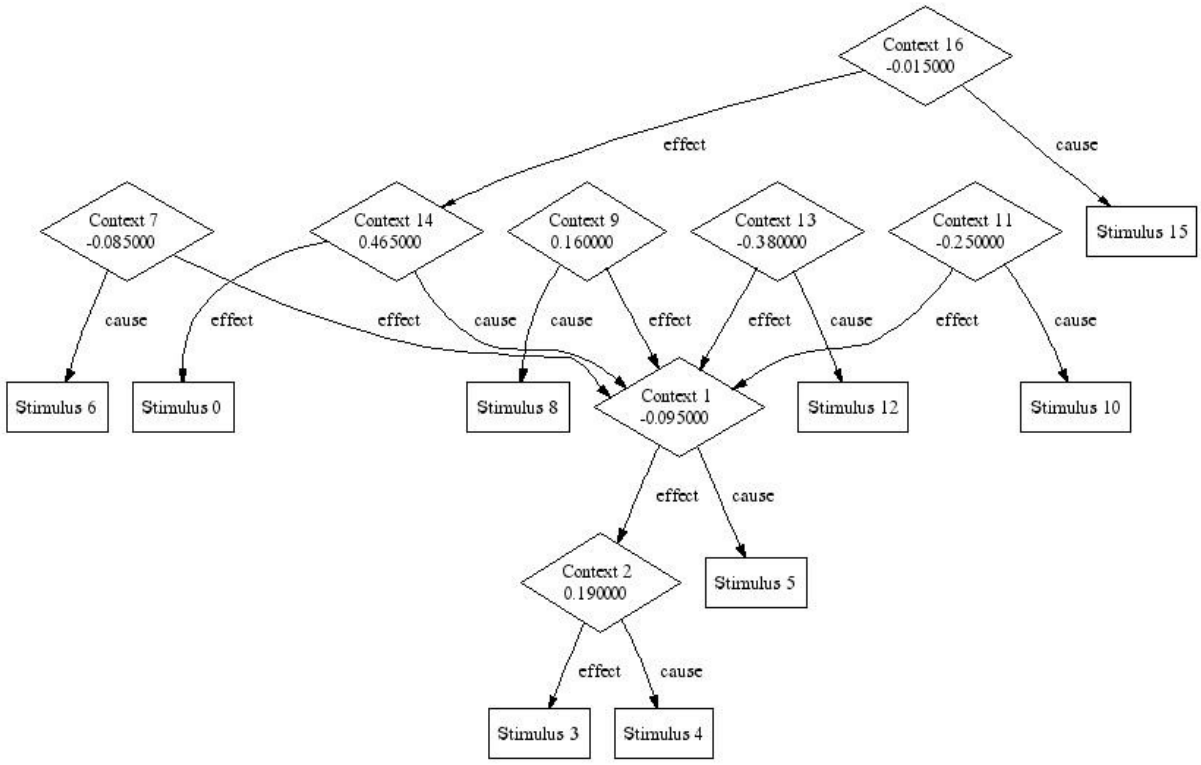

Figure 8 – An Improbable Network

2.2 Neural Network

After the context environment is generated, a corresponding neural network is manually created that mirrors the cause and effect relationships in the environment. Mediator initial, or "base", enablements are set to a minimum value of .0001. The objective of the learner in this study is to learn the probabilities of environmental contexts; generating new mediators from learned cause and effect is a topic currrently under investigation.

Figure 9 shows the neural network corresponding to the environment depicted in Figure 7 after learning for 100 sense-response cycles has taken place. The three types of neurons in Mona, shown in separate partitions, are: (1) Receptors that sense environmental information, (2) Motors that express responses to the environment, and (3) Mediators that connect other neurons (including other mediators), in a cause and effect manner.

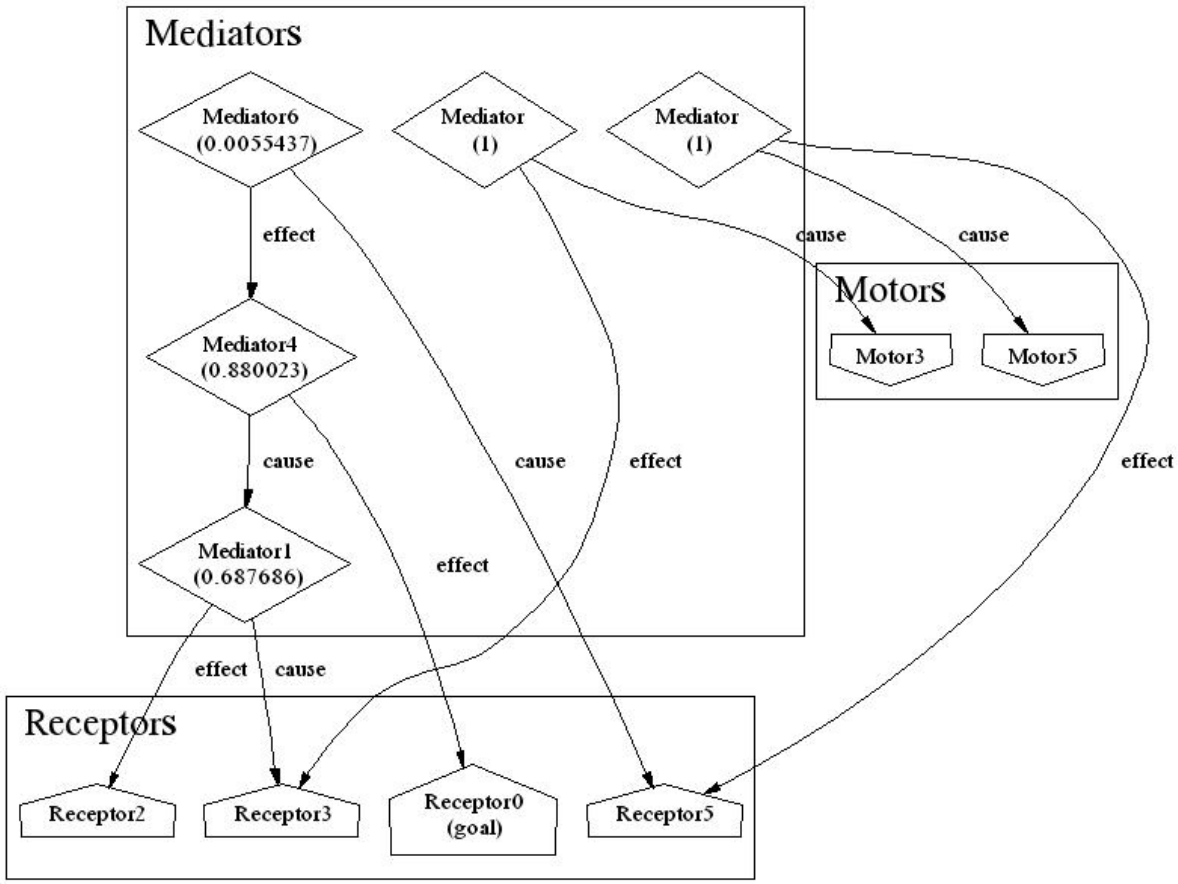

Figure 9 – Neural Network

The environment and neural network form a complementary system. The environment outputs a "current" stimulus that can be sensed by a receptor neuron. The neural network then produces a response that is used by the environment to probabilistically produce another stimulus. The initial stimulus is a null value not sensed by any receptor. When Stimulus 0 is sensed, the goal has been achieved. Motor neurons output responses that deterministically produce cause (but not effect) stimuli in the environment, thus allowing the learner to "navigate". After a cause stimulus is reached, the learner may attempt to produce its associated effect by issuing a "wait" response, not associated with a motor neuron. At that time, the environment determines, according to accumulated probabilities, whether to make the associated effect stimulus current. For example, in the environment and neural network shown above, the optimal response sequence is: produce Stimulus 3 via Motor 3, wait for Stimulus 2 with .62 probability, then having fired Context 1, wait again the the goal with .91 probability. Issuing a response from Motor 5 would be counterproductive, since it lowers the probability of achieving the goal through Context 4. The numbers shown in the Mediator symbols in the neural network are enablements that correspond to learned probabilities: Mediator 4 = .88, and Mediator 1 = .68. Mediator 6 has been learned to be ineffective in goal-seeking.

As previously mentioned, responses are motivated in Mona via a propagation scheme originating from goal states and accumulating in motor neurons. A response is probabilistically selected based on the weighted motivation values resident in motor neurons. In this study,

Receptor 0 is always associated with a positive goal value. During motivation propagation, an highly enabled mediator, that is, one with a high probability of success, shunts motivation to its cause neuron; the rationale being that the mediator's cause-effect transition will succeed if its cause neuron can be influenced to fire. A mediator with a low enablement shunts motivation to higher mediators for which it is an effect, thus influencing them to produce responses that enable the mediator. The enablement accumulation is somewhat analogous to the probability accumulation in the environment. When the cause of an enablement mediator fires, motivation is shunted to influence its effect to occur.

Learning is based on a wager/payoff scheme. When its cause fires, a mediator expresses a "stake" in its effect firing by issuing a wager on that neuron. The magnitude of the wager is proportional to the firing strength of the cause. Motor and receptor neurons fire at a constant strength of 1.0. A mediator's firing strength is a product of its effect's firing strength and size of the wager on the effect. Wagers from higher level mediators iteratively propagate to effect neurons, carrying an enabling influence. If the effect neuron fires, the mediator is rewarded in proportion to the size of its wager. Conversely, should the neuron not fire within a prescribed time, the mediator is punished proportionately. Reward and punishment take the form of an increase and decrease to the base enablement of the mediator, respectively. Base enablement is roughly defined to be the conditional probability that the mediator can fire its effect given that its cause fires. The base enablement update begins by computing a weighted wager:

$$\text{weighted-wager}_{\text{mediator}} = (\text{wager}_{\text{mediator}})^2 / (\text{base-enablement} * \sum \text{wagers}_{\text{all-mediators}}) \quad (1)$$

This term represents the payoff weight should the wager succeed. For example, a maximum weighted-wager of 1.0 represents the entire base enablement of a mediator as the sole wager on an outcome. Conversely, the payoff weight is less if other wagers are present to contribute to the outcome or if a mediator wagers only a fraction of its base enablement. The updated enablement is the historical sum of the successful weighted-wagers, including the current weighted-wager, divided by the total sum:

$$\text{base-enablement} = \sum \text{weighted-wager}_{\text{successful}} / \sum \text{weighted-wagers} \quad (2)$$

Initially the learner is likely to attempt to reach the goal stimulus from an immediate cause. Doing this repeatedly forces it to learn the "true" base enablement values of these mediators. In a probabilistic manner, the learner wagers from higher level mediators. In some networks, cause neurons are themselves mediators. To fire them, the motivation mechanism essentially turns their effects into secondary goals. Over sense-response iterations, base enablement values are learned and goal-seeking becomes more efficient.

3 RESULTS

Figure 10 shows success rate plotted as number of contexts increases. A trial consists of iterations of sense-response cycles. A successful trial is defined as reaching the goal state within an amount of time that permits the learner to produce every elementary cause stimulus. Thus the time is proportional to the number of contexts. For this experiment, the context probabilities ranged from -1.0 to +1.0 inclusive. Three lines are graphed: base, learning, and random success rates. The base rate is intended to give a notion of a non-learning task; the base enablement

values of the mediator neurons are initially set to the actual probabilities of the corresponding contexts. A negative probability corresponds to a zero base enablement. Although these quantities are not identical in meaning, it serves as a suitable performance baseline. The learning success rate is the experimental result. The random rate serves as a lower baseline and is the result of producting random valid responses. The value plotted for each context level is the average of 100 test trial samples, each sample taken after 1000 learning trials.

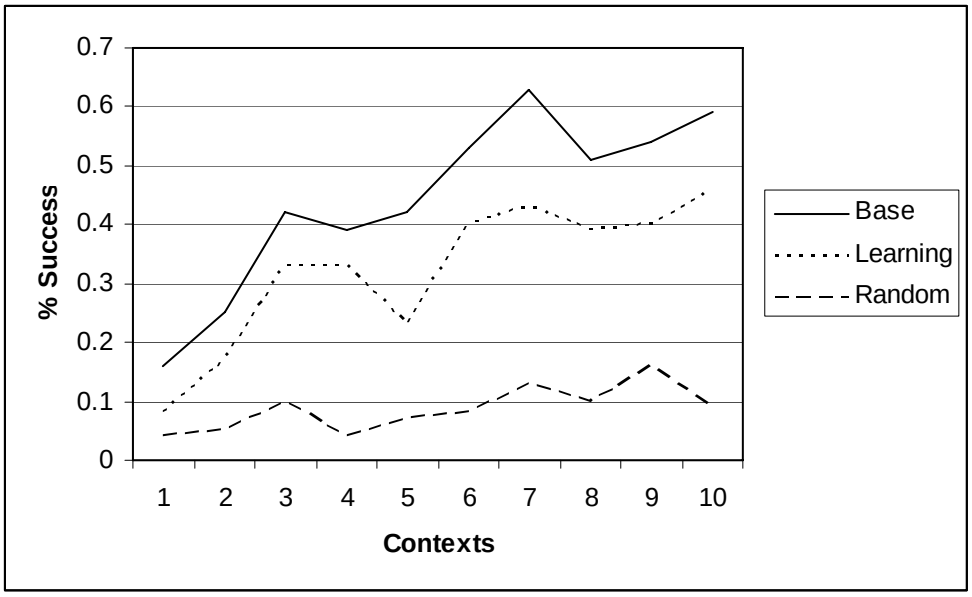

Figure 10 – Success rate with increasing contexts

The data indicates that the learner performs better as the environmental complexity, in the form of the number of contexts, increases. The reason for this is that a greater number of contexts often affords more pathways for success. For example, in an environment of a single context, due to the possibility of negative probabilities, there is a 50% chance of an unreachable goal stimulus. With more contexts, more pathways are possible to reach the goal.

Figure 11 is intended to display the effectiveness of learning. Here, whether successful or not, the average accumulated goal-reaching probability is plotted over an increasing number of contexts. Consistent with the findings presented in Figure 10, as contexts increase, opportunities to accumulate positive probabilities increase.

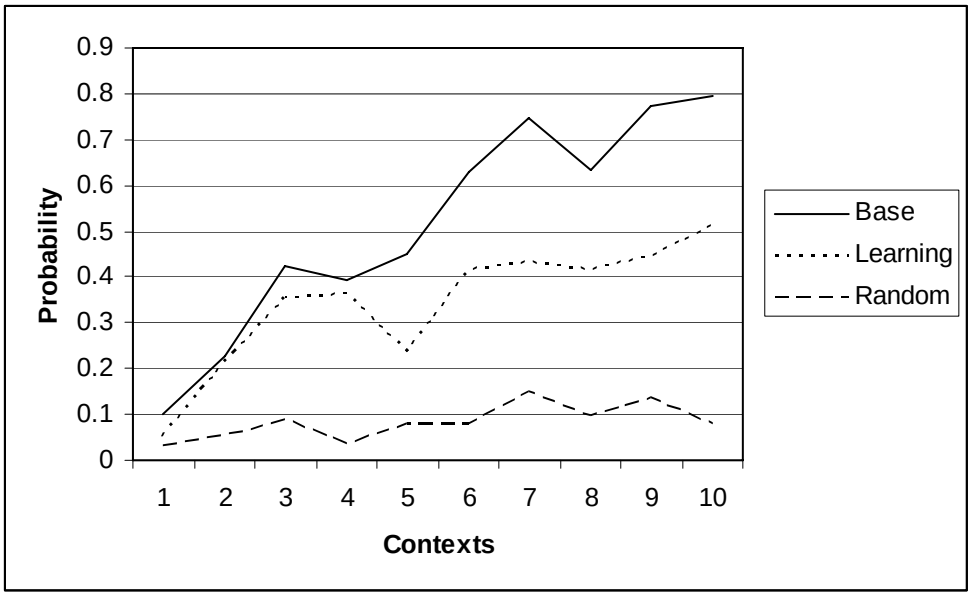

Figure 11 – Success probability with increasing contexts

Figure 12 shows the rate of learning as a function of increasing trials for environments having 5 and 10 contexts. For this experiment, in order to accentuate the influence of conflicting contexts on learning, all contexts directly "funnel" into the goal stimulus, i.e., all contexts have the goal stimulus as their effect. Furthermore, only one of the contexts contributes a +1.0 probability; the remainder contribute -1.0. Thus the task is to learn which is the positive context. Each plotted value is the result of averaging 100 samples. The simplified task of discriminating among 5 contexts is reflected in the more rapid learning curve. At about 300 trials, however, the learner is also able to learn the 10 context task successfully.

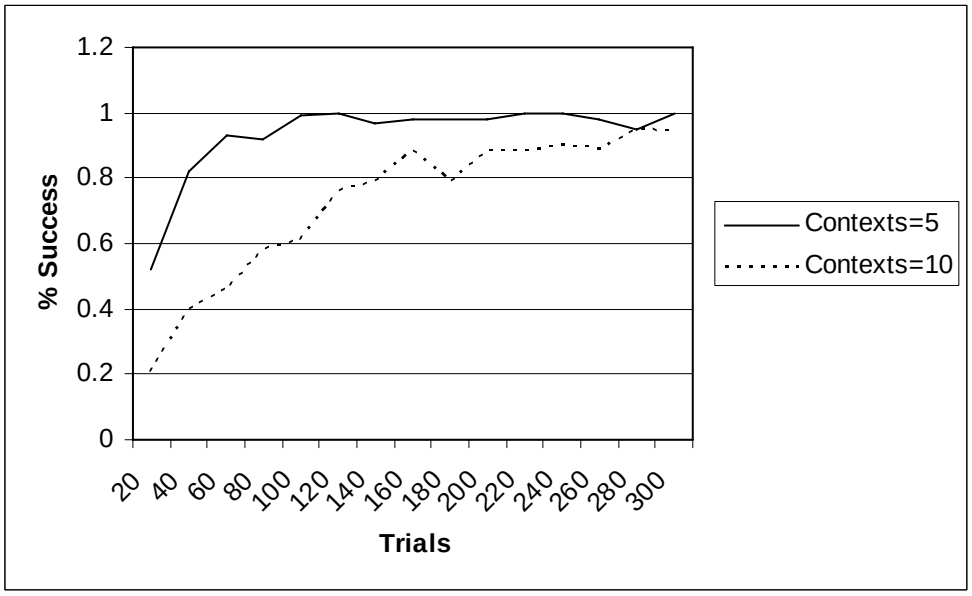

Figure 12 – Success rate with increasing trials

4 DISCUSSION

One way of looking at the method presented here is as an extension of reinforcement learning (Kaelbling, Littman, and Moore, 1996) to context-related problems. The purpose of reinforcement learning is to learn paths through a state space to goal states. Actions causing state transitions are scored with a utility value according to how well they contribute to goal-seeking. In this sense the utility of a mediator corresponds to its enablement. In prototypical reinforcement learning models, exemplified by Q-Learning (Watkins, 1989) and Temporal Difference Learning (Sutton, 1988), the state space is a flat Markovian space, necessitating the embedding of context information into state labels, which in turn can result in a proliferation of states. The use of hierarchies is a powerful means of avoiding this proliferation: they provide modularity and reusability. For example a state transition S0 → S1 may exist within context C0 and C1 wherein the two contexts affect the transition probability differently. In a flat space, S0/C0 → S1/C0 and S0/C1 → S1/C1 are needed to express this. Moreover, context hierarchies allow the dynamic linking of cause and effect chains that are not explicity encoded. For example, suppose context C0 has S0 → S1, and in C1 has S1 → Goal. The two contexts can be linked through the shared state S1 to create a goal path. If S1 were encoded in a flat space as S1/C0 and S1/C1 the linkage information would be lost.

The question arises as to how Mona differs from more conventional, e.g. feedforward, artificial neural networks. A recurrent network could plausibly be trained to recognize input patterns representing the existence of contexts and to associate these with response sequences. The most important distinction is that Mona is a goal-seeker, more like a planner than a pattern classifier. For a goal-seeker, many state path variations may suffice to achieve success. Pattern classifiers, such as feedforward neural networks, can be used to recognize environmental states. In sum, goal-seeking and pattern classification are complementary techniques.

5 CONCLUSION AND FUTURE WORK

The described technique allows Mona to successfully learn environmental contexts in the abstracted sense defined here. The expectation is that this will carry-over to more general learning situations. This work is viewed as a progress step, supplying a key piece of infrastructure necessary for more elaborate learning tasks. The ability to create new mediator neurons by hypothesizing cause and effect relationships, including those involving logical conjunctions among multiple causes, is being designed as part of a maze-learning task. As an example of a conjunction, a door might not open unless a code is entered and a key is used. In this case, a logical *and* relationship exists between the two causal events. The maze-learning task also involves the ability to learn inhibiting influences, which is essential to many tasks.

The C++ source code and other reference materials are available at:
www.itk.ilstu.edu/faculty/portegys/research/.

7 REFERENCES

Benson, S. and Nilsson, N. 1993. Reacting, Planning and Learning in an Autonomous Agent, *Machine Intelligence 14*, Edited by K.Furukawa, D. Michie, and S. Muggleton. Oxford: Clarendon Press, pp. 29-64.

Bodén, M. and Wiles, J. 2002. On learning context free and context sensitive languages, *IEEE Transactions on Neural Networks*. **13(2)**, 491-493.

Bonzon, P. 1997. A Reflexive Proof System for Reasoning in Contexts, in *Proceedings 14th National Conference on Artificial Intelligence* (AAAI 97), Providence, RI, 1997 Carpenter, F. 1974. *The Skinner Primer: Behind Freedom and Dignity*. New York: The Free Press, a Division of Macmillan Publishing Company, Inc.

Howard, M. and Kahana, M. 2002. A Distributed Representation of Temporal Context, *Journal of Mathematical Psychology*, **46**, 269-299.

Hasselmo, M. and McClelland, J. 1999. Neural models of memory, *Current Opinion in Neurobiology*, **9**, 184–188.

Kaelbling, L., Littman, M., and Moore, A. 1996. Reinforcement Learning: A Survey. *Journal of Artificial Intelligence Research*, **4**, 237-285

Maes, P. and Brooks, R. (990. Learning to Coordinate Behaviors, *AAAI-90*, Boston, MA. pp. 796-802.

Portegys, T. 1999. A Connectionist Model of Motivation, *IJCNN'99 Proceedings.*

Portegys, T. 2001. Goal-Seeking Behavior in a Connectionist Model, *Artificial Intelligence Review*, **16 (3)**, 225-253.

Sabes, P. and Jordan, M. 1996. Reinforcement Learning by Probability Matching, In *Advances in Neural Information Processing Systems*, **8**, 1080-1086

Schank, R. C., Childers, P. G. 1984. *The Cognitive Computer; On Language, Learning, and Artificial Intelligence*. Addison-Wesley Publishing Company, Inc.

Steijvers, M., and Grunwald, P. 1996. A Recurrent Network that performs a Context-Sensitive Prediction Task**,** In *Proceedings of the 18th Annual Conference of the Cognitive Science Society*. Erlbaum.

Sun, R. and Giles, C.L. 2001. Sequence Learning: From Recognition and Prediction to Sequential Decision Making, *IEEE Intelligent Systems*, **16(4)**, 67-70.

Sutton, R. 1988. Learning to predict by the method of temporal differences. *Machine Learning*, **3(1)**, 9-44.

Turner, R. 1998. Context-Mediated Behavior for Intelligent Agents, *International Journal of Human-Computer Studies* special issue on "Using Context in Applications", **48(3)**, 307-330.

Watkins,C. 1989. Learning from Delayed Rewards, Thesis, University of Cambidge,England.

Wermter, S., Arevian, G., and Panchev, C. (1999). Recurrent neural network learning for text routing. In *Proceedings of the International Conference on Artificial Neural Networks*, pp. 898-903, Edinburgh, UK.