# Neural Schemas: A Mechanism for Autonomous Action Selection and Dynamic Motivation

LEE MCCAULEY
Department of Computer Science
The University of Memphis
Dunn Hall Room 373, Memphis, TN  38152
USA

*Abstract:* - Neural schema mechanism is a new autonomous agent control structure that makes use of both neural network and symbolic constructs to learn sensory motor correlations and abstract concepts through its own experience. The mechanism can also learn which intermediate states or goals should be achieved or avoided based on its primitive drives. In addition, a psychological theory of consciousness is modeled that allows the system to come up with creative action sequences to achieve goals even under situations of incomplete knowledge. The result is an architecture for robust action selection that learns not only how to achieve primitive drives, but also learns appropriate sub-goals that are in service of those drives.

*Key-Words:* - Schema, learning, motivation, consciousness, neural, autonomous, control

## 1 Introduction

As soon as a child is born they are given the daunting task of creating a reality. They must form the knowledge and goal structures that will dictate how they interact with the world around them – and why. From relatively few initial clues each child must learn everything that they need to survive and prosper. They need to learn things like how their muscles work, how to communicate with their parents and others, and how to conform to the norms of their society. Some of this learning is just a matter of noting the common sensory results to actions performed under a given situation, simple muscle control, for example. Almost all learning beyond these first simple steps, however, requires that the child be able to judge not only what an action does, but also whether that result is a desirable one or not. Luckily for humanity, evolution has provided us with a set of innate sensory inputs that are pre-wired in our brains to give us pleasure, pain, happiness, sadness or any number of other feelings. Even so, there is more to what a child must learn than just which actions directly result in activation of one or more of the innate feelings; they must also learn which environmental states that, in and of themselves, do not trigger an innate response, nonetheless represent an increased likelihood of encountering a state that does trigger an innate positive or negative response at some point in the future.

To take the first steps in accomplishing this in an artificial system, a sufficiently general mechanism called schema mechanism created by Gary Drescher was extended to more closely approximate some human cognitive phenomenon [3, 4, 5]. The new extension is called neural schema mechanism due to its resemblance to connectionist architectures. The result is a mechanism that uses its own experience to learn about its world by creating new nodes and links that embody new knowledge about its environment and its own abilities to effect that environment. In addition, the mechanism includes attention and “consciousness” modules that allow it to more efficiently utilize resources and discover novel and original solutions to daunting problems.

All told, the neural schema mechanism takes steps toward the implementation of a comprehensive mechanism of mind that includes conceptual and sensory-motor learning, “consciousness,” and a blending of logic with desires via a dynamic motivational system.

## 2 Neural Schemas

In general, one can think of the mechanism as a network of nodes connected via links. Each node has its own activation, which it can spread to other nodes through the links. The links perform a function on the activation sent through them from their input node and deliver the resulting activation to their output node. With the exception of some rules that dictate which links are allowed to transmit activation, this part of the mechanism works exactly like any other connectionist network. Abilities are added to the nodes and links of the network that allow them to keep track of the necessary statistical data for the schema mechanism. In addition, nodes have the ability to check the statistical information in order to determine if a new node needs to be created and have the ability to create the new network elements.

## 2.1 Item Nodes

Every agent must have some way of determining the current state of its environment. In neural schema mechanism, environmental states are expressed through the activation of item nodes. Primitive item nodes, those that are built in at the time of creation of the agent, are attached directly to some sensory apparatus. Their activation can be set by the sensors to any value between –1 and 1, and is converted into a discrete state via a threshold function. If the node's activation is above the threshold, then the item node is considered to be *on*. If the node's activation is below some negative threshold then the item node is considered to be *off*. Otherwise the state is said to be *unknown*.

Item nodes can also be learned. These synthetic item nodes are created when the mechanism discovers a state of the world that is not expressed by any of the existing items. The method of this discovery is based on the notion of object permanence; in other words, that things in the world tend to stay put for some period of time. For example, a person standing in one room of their house might look behind them and see a blue vase. Under most circumstances, looking backwards does not result in seeing a blue vase, but if the person has just turned around and seen the vase, then a repeat of the action is likely to result in the same relatively unusual vision. Neural schema mechanism uses this fact to create a synthetic item node that will, over time, approximate the conditions under which the unusual occurrence can be relied on. In this way, the mechanism can create concrete representations of abstract concepts.

## 2.2 Action Nodes

Once the agent has perceived its environment it must have some way of acting upon that environment. This function is performed by action nodes. As with item nodes, there are both innate and learned action nodes. Primitive action nodes connect a neural schema agent to its actuators, while learned action nodes generally represent higher-level actions. One such higher-level action might be "ride-a-bike" as opposed to the numerous individual muscle movements necessary to actually ride the bike.

Even though the action nodes carry out the agent's actions, an action node by itself cannot directly be executed. Instead it must be activated through the execution of a schema node for which it is the designated action. Each schema node can have one and only one action node associated with it.

## 2.3 Schema Nodes

Explicit knowledge of the agent's environment is expressed through schema nodes. Each schema node embodies the knowledge that a given action will have a specific result if it is executed under certain conditions. The result and context of a schema node are each made up of a set of item nodes to which it is connected via result and context links respectively.

The links contain information relating to the relevance of the linked-to item with regard to the schema node's action. For instance, a result link states that the item node linked to is more likely to turn *on* (or *off*) when the schema node's action is executed than when the action is not executed. Note that this does not imply that the result is likely to occur, only that it is more likely to occur when the action is taken than when it is not. In a similar fashion, context links state that the results specified by the schema node are more likely to follow the action when the linked-to item node is *on* (or *off*).

Learning new schema nodes is the primary form of knowledge acquisition in the mechanism. The initial state of a neural schema network contains only primitive item, goal, and action nodes along with what are called "bare" schema nodes. A bare schema node is one in which there is only a link to an action node. Since action nodes cannot directly be executed, this is necessary to allow the system to perform any actions. It also gives the system the foundations upon which future learning can take place. When a schema node is chosen for activation (randomly at first) the change of activations in the item nodes will tend to cause them, along with the just previously activated schema node, to be attended to by the attention mechanism. This, in turn, causes generic links to be created between the schema node and item nodes. These generic links will, from that point forward, maintain statistical information needed to determine the node's relevance to that schema node's action. For instance, if a potential result link (one that has a schema node as the input node and an item node for output) notes that its item node turns *on* more often than it turns *off*, then it will have a high positive relevance. Note that this statistic is not affected when the state of the item node does not transition. When the relevance of a potential result link is significantly positive or negative then a new schema node is created that is a duplicate of the existing one except that it has a result link noting the relevance of the particular item node to which it connects. Once a schema node has one or more result links, potential context links (those that have an item node as input and a schema node for output) note if its item node's state is a determining factor to the success of the schema node. For this, the link must maintain statistics on the success of the schema node when the item node's state was *on* as opposed to *off*. When a relevant item node is noticed through a potential context link, a new schema node is created with that item node connected to it via an appropriate context link. In this way the agent's knowledge of its environment is constructed through its own experience.

Schema nodes also keep track of their own reliability. In other words, given that the schema node's context is satisfied and the action is executed, what is the probability that the entire result set will obtain? The reliability of a schema node is a factor in determining how much resistance flows through it and the likelihood that it will be chosen for activation.

The creation of new schema nodes eventually leads to the chaining of nodes where one schema node's results correspond to the context set of another schema node which, itself, feeds into a third schema node and so on until a goal state is reached. But with multiple paths to a goal or even to multiple goals, the system must have some way to judge which is the best path.

### 2.4 Goal Nodes

The motivations of an agent in neural schema mechanism take the form of goal nodes. A primitive goal node is analogous to basic drives; its purpose being to influence the mechanism in such a way as to bring about its goal state, however that might be accomplished. Whether the goal node is primitive or learned, the method for bringing about its goal state is the same, namely the spreading of resistance through the network.

Resistance is spread like activation except that it flows backwards through the network. Resistance attempts to measure the desirability of an item node or the usefulness/importance of activating a given schema node with respect to achieving the agent's current goals. There are two primary ways that resistance changes the behavior of the agent. First, the resistance on a link modifies the weight on that link so as to increase or decrease the amount of activation that gets delivered to the output node. Secondly, the resistance on a particular schema node is used as a determining factor in selecting which schema node to activate on any given cycle.

A primitive goal node in the new mechanism will have an intrinsic primitive resistance value provided at design time. Even though these nodes propagate resistance constantly, the amount sent out is determined by the degree to which the goal state represented by the node is being recognized.

New non-primitive goal nodes are created whenever a new schema is created whose result set is novel. These learned goal nodes, therefore, represent the desire to attain that state. The question, of course, is how to determine whether this arbitrarily abstract state should be generally sought or avoided. The mechanism accomplishes this through the concept of delegated resistance. Learned goal nodes keep track of the difference between the highest resistance value of any applicable schema nodes when the goal node's state is *on* and when it is *off*. The delegated resistance of the goal node is, therefore, a function of that difference and will, over time, acquire appropriate values for the goal state.

## 3 Action Selection

The ability for the neural schema mechanism to pursue multiple conflicting or complimentary goals is accomplished via the propagation of resistance through the network. Even though we have borrowed the electricity metaphor for resistance the values are reversed to decrease confusion. We would say that a goal that is desired would propagate a positive resistance and one for which avoidance is preferred would propagate a negative resistance.

There are limits as to which links are permitted to transmit resistance. Item nodes can only send resistance through result links to schema nodes and schema nodes are only allowed to send resistance through context links to item nodes. This process maintains the path to the instigating goal node with stronger resistance values occurring closer to that goal. Resistance does not actually change the activation of a node; instead it affects the amount of activation that passes through a link.

For goal nodes in service of composite (learned) actions, resistance is only propagated for a short period of time (based on the expected duration of the action) unless the action is repeatedly selected. Primitive goals will continuously spread resistance based on the current state of the system. This allows a primitive goal, such as hunger, to be pursued to varying degrees at all times while non-primitive goals will be pursued only when they are appropriate. When the goal node's need is high, the agent may pursue it single-mindedly; however, once such a goal has been satisfied, it would be reduced to a level that has little to no effect on the behavior of the system.

While resistance begins at the item nodes pointed to by active goal nodes, activation has its source in item nodes that represent the current state of the environment. Activation flows in an opposite manner to resistance. An item node can pass activation through its context links to schema nodes, which can pass activation through their result links to item nodes. Activation can be transmitted from any node whose activation level exceeds a threshold or drops below a negative threshold. The total amount of activation of a network is limited to half of the total number of nodes. If the total activation of the network is below this maximum, then only a decay rate is used to reduce the activation of a given node. However, if the total activation goes above the maximum, then the activation is normalized so that each node maintains its correct percentage of the whole but the total activation of the network remains equal to the maximum. It may also be possible to allow this value to change during a run. For instance, lowering the value from 0.5 to 0.25 might promote more abstract thinking by forcing the system to highly activate fewer, hopefully more general, nodes.

Even though activation plays an important role in action selection, it is not the only element to consider. In addition to activation, a schema node's applicability is taken into account. Under normal circumstances, for a schema node to activate the node must be applicable and have the highest activation of all applicable schema nodes. The use of resistance increases the likelihood that the schema node selected for activation lies on some path to a goal state. In addition, the fact that the resistance values become stronger the closer a node is to the goal state, means that the schema node chosen for activation is more likely to be one which is fewer steps away from achieving the goal. This method gives the neural schema mechanism the ability to be opportunistic while pursuing multiple goals.

## 4 "Consciousness" and attention

It was stated in the previous section that, under normal circumstances, a schema node must have the highest activation of all applicable schema nodes to be chosen for activation. The other way that a schema node could become active is through a response to a "conscious" broadcast. The term "conscious" is used here only to denote that the mechanism implements a portion of a psychological theory of consciousness. It is neither our intent to suggest nor our belief that the mechanism displays true consciousness. That being said, it is our hope that, within the limited scope of the "conscious" mechanism implemented, one might be able to form hypotheses regarding human consciousness or be able to discover discrepancies that could be remedied in future versions of the system.

The particular theory of consciousness that has been partially modeled in the neural schema mechanism is Bernard Baars' global workspace model [1, 2]. This theory puts forward the claim that human cognition is implemented by a multitude of relatively small, generally unconscious, special purpose processes. Within this multi-agent system, groups of such processes, called coalitions, compete for entry into a global workspace. The workspace serves to broadcast the coalition's message to all the unconscious processors, in hopes of recruiting other processors to help solve the current impasse or to take note of the current novel situation. The number of processors that can be in the workspace at any one time is limited but not set at a specific number.

Even though in humans consciousness plays an important role in solving problematic situations, it is easy to recognize from one's own experience that consciousness is not only used in these circumstances. To the contrary, humans are usually conscious at any given moment without, necessarily, being in the midst of solving some unresolved issue. This is where attention comes into play. In the theatre metaphor presented by Baars, the spotlight of attention is the method by which coalitions get into consciousness[1, 2]. However, the theory does not dictate what criteria should go into deciding which coalitions are attended to, nor does it dictate that all coalitions that get the spotlight of attention shined on them must be a novel situation or problem, only that novelties have an increased probability of entry.

Attention in the neural schema mechanism has a similar purpose to the one created for the original schema mechanism by Foner and Maes; namely, to reduce the computational requirements of the system [6]. Neural schema mechanism, however, works by using a combination of the change in a node's activation over time, the magnitude of resistance flowing through the node, and the node's association to other nodes recently in "consciousness." Each node in the network has a degree of "consciousness." In this way it can be thought of as similar to activation. All node types, with the exception of action nodes, are candidates for inclusion into "consciousness." The first two elements that make up the calculation for the degree of "consciousness" in any given node are determined without respect to the node's relationship to any other node. First, the slope is calculated for every node in the network. Slope is defined here as the absolute value of the average change in activation over the last five cycles, plus the absolute value of the resistance at the node multiplied by a resistance factor. The resistance factor is used to balance attention between highly active nodes and nodes deemed important in obtaining or avoiding goals. The resulting slope value is proportional to the rate of activation change in the previous five cycles modified by the resistance level. This portion of the equation gives a measure of the node's novelty (rate of change of activation) and usefulness in accomplishing current goals (resistance). This value is combined with the amount of "consciousness" that was sent to the node through its links. The passing of consciousness is meant to provide the agent with some continuity of thought from one cycle to the next. Under most circumstances this give the agent something akin to a train-of-thought event though highly novel or unexpected occurrences could force the agent to jump to a new awareness unrelated to the previous contents of "consciousness." At this point the "consciousness" values for all of the nodes in the system are normalized in order to maintain a constant value for the total amount of "consciousness" in the system.

The branching factor of the network is largely determined by the creation of links between attended nodes. Once a set of nodes is selected for "consciousness" by the attention mechanism, two things happen. The first task is to update the links between all of the nodes in the spotlight. A node cannot have a link to itself, and action and item type nodes cannot link to other action and item type nodes. If two nodes, say Node-A and Node-B, had no previous connections between them, then two new links are created, one with Node-A as input and Node-B as output, and one connecting the two in the opposite direction. Previous research has shown that an attention mechanism that operates in a similar manner with regards to this first task, reduces the branching factor of the mechanism without significant lose of learning capabilities [6].

The second thing that happens to "conscious" nodes is that a broadcast is sent out to all schema nodes in the network. The broadcast consists of a set of item nodes. If an item node is in the spotlight, then it is, itself, one of the item nodes broadcast. If, on the other hand, attention is broadcasting information for a schema node or goal node, then that node's context set is broadcast. For goal nodes, we would say that their goal set is broadcast. In total, the broadcast will contain all of the item nodes represented by the coalition of nodes in the spotlight. Schema nodes respond to the broadcast if any of the items transmitted correspond to any or all of its result set. The schema node responding to a "conscious" broadcast sets an internal flag that denotes that it is responding and increases its activation proportional to the percentage of its result set represented in the broadcast. Responding to a broadcast does not automatically guarantee selection for activation, but it does increase the likelihood.

The flag denoting a node's response to the broadcast is an overriding factor to the applicability rule for action selection. It is in this way that an agent can discover new paths to a solution or goal state that did not previously exist or were not active highly enough to become chosen for activation. Selecting schema nodes whose result sets match the desired goal under these special circumstances even when the context is different or unknown allows the agent the possibility of quickly learning the correct context or alternate context for achieving the goal. For example, if the schema node chosen for activation by a response to the broadcast has no context at that moment then its selection serves to decrease the time necessary to discover the correct context due to its increased likelihood of activation.

The general formula for attention serves a number of different purposes. First off, it increases the likelihood that unexpected occurrences will become "conscious". The other element in the formula for attention is the resistance of a node. As shown previously, resistance is the measure of how much a given node is in service of the agent's currently desired goal context (or contexts). For this reason, the addition of resistance to the formula increases the degree to which the agent's current goals are represented in "consciousness."

The use of "consciousness" does not always change the behavior of the system. To the contrary, for most situations the mechanism would have chosen the same set of schema nodes in the same order due to the other factors already laid out. At times such as these, one could say that the agent is "consciously" aware of what it is doing or experiencing, but that "consciousness" plays no discernable role in the agent's behavior. However, under circumstances where a path is not clear or where two (or more) separate chains of schema nodes must converge on a goal state, "consciousness" provides a solution.

## 4 Conclusion

Neural schema mechanism is an attempt at creating an autonomous agent control structure such that an agent using it would perform well even when placed in an environment for which it has little or no prior knowledge. From an initial state consisting only of a set of actions, sensory inputs, and primitive goals/drives a neural schema agent can discover what its actions do and how to achieve its goals. It learns about its environment through its own experience and makes value judgments based on its own internal drives. There are still several avenues of experimentation and extension that would be of benefit. An episodic memory, for instance, might be of use. This could be accomplished by allowing a subset of the item nodes to trigger recall from a memory store of past similar states and be read into a separate set of item nodes. An addition to this might even limit the recall triggers to those item nodes currently in "consciousness."

There are two drawbacks to the neural schema mechanism in its current state. First, the learning rate of the system with regard to the rate of new knowledge created is slower than had been hoped and may well be slower than other methods although direct comparison is difficult and has not been attempted as of yet. The second drawback is the computational complexity of the system. The network within a run does not grow exponentially fast due to the utilization of several methods that suppress redundant or obviously fruitless paths and due to the use of trash collecting techniques that remove unnecessary nodes after they have been deemed erroneous or very rarely used. Even so, the system does display a small but noticeable slowdown when the network begins to get large. The mechanism has not yet been run to its breaking point but the question of scalability is a serious issue. If a relatively small environment can tax the system computationally, then it is unclear if larger environments could be handled in real-time. There is always the possibility that new hardware will render this issue moot, but optimizations should, nonetheless, be explored.

Despite these issues neural schema mechanism has shown promise from both an artificial intelligence and cognitive science perspective. In essence, a neural schema agent, like a human child, creates its own reality. It takes agent autonomy one step further, allowing an agent to not only decide on the path to its goal, but also to create new goals. In addition to being inspired by human psychology, neural schemas demonstrate clear advantages to the use of those cognitive phenomenons. The way in which consciousness, in particular, has been modeled allows the mechanism to display behavioral characteristics similar to what the psychological theory describes as benefits of consciousness [1, 2]. The mechanism has shown the ability to choose appropriate or reasonable actions more often than randomness would dictate in situations where complete knowledge of the world in that state had not yet been reached. In addition, the neural schema mechanism's motivational system has shown that it has the ability to not only pursue multiple goals, but to do so in such a way as to reduce its own risk.

*References:*

[1]. Baars, B., *A Cognitive Theory of Consciousness*, Cambridge, MA: Cambridge University Press, 1988.

[2]. Baars, B., *In the Theater of Consciousness*, Oxford, UK: Oxford University Press, 1997.

[3]. Drescher, G., *The Schema Mechanism: A Conception of Constructivist Intelligence*, MIT M.S. thesis, 1985.

[4]. Drescher, G., A mechanism for early Piagetian learning, *In Proceedings of the Sixth National Conference on Artificial Intelligence*. AAAI-1987.

[5]. Drescher, G., *Made Up Minds: A Constructivist Approach to Artificial Intelligence*, Cambridge, MA: MIT Press, 1991.

[6]. Foner, L. and Maes, P., Paying Attention to What's Important: Using Focus of Attention to Improve Unsupervised Learning, *In Proceedings of the Third International Conference on Simulation of Adaptive Behaviour '94*. Cambridge, MA: MIT Press, 1994.