

Symbolic Reasoning in Spiking Neurons: A Model of the Cortex/Basal Ganglia/Thalamus Loop

Terrence C. Stewart (tcstewar@uwaterloo.ca)

Xuan Choo (fchoo@uwaterloo.ca)

Chris Eliasmith (celiasmith@uwaterloo.ca)

Centre for Theoretical Neuroscience, University of Waterloo
Waterloo, ON, N2L 3G1

Abstract

We present a model of symbol manipulation implemented using spiking neurons and closely tied to the anatomy of the cortex, basal ganglia, and thalamus. The model is a general-purpose neural controller which plays a role analogous to a production system. Information stored in cortex is used by the basal ganglia as the basis for selecting between a set of inferences. When an inference rule is selected, it commands the thalamus to modify and transmit information between areas of the cortex. The system supports special-case and general-purpose inferences, including the ability to remember complex statements and answer questions about them. The resulting model suggests modifications to the standard structure of production system rules, and offers a neurological explanation for the 50 millisecond cognitive cycle time.

Keywords: decision making; neural production system; neural engineering; cognitive architectures

Introduction

The primary goal of our ongoing research is the creation of a biologically realistic neural cognitive architecture. Such an architecture would provide an explicit and quantitative connection between cognitive science and neuroscience. Bridging these fields leads to benefits in both directions; aspects of a cognitive theory can predict and be constrained by neurological details, and the neurological details can in turn identify important modifications to cognitive theory.

In this paper, we present a model of sequential symbolic reasoning implemented using 373,000 simulated spiking neurons. The connectivity of these neurons, their neural parameters, and their associated neurotransmitters are fixed based on neurological evidence from the basal ganglia, thalamus, and cortex. By adjusting the synaptic connections of neurons at the inputs and outputs of the basal ganglia, we can define the inferences that the system will follow. Since these rules can be adjusted for a wide variety of IF-THEN symbol manipulation tasks, we believe that our model is the first biologically realistic general-purpose neural controller that can play a role analogous to a production system.

The model involves the basal ganglia, the thalamus, and various cortical areas. The cortex holds a variety of information about the current situation, such as visual input and the contents of working memory. The basal ganglia performs action selection, taking information from the cortex to determine which of the rules is most appropriate to use in the current situation. This choice is sent to the thalamus, which acts as a routing system, implementing the effects of those rules by transferring information between

cortical areas. As the information stored in the cortical areas changes, different actions will be selected in turn, allowing for controlled and organized sequences of actions.

To present this model, we first provide a brief description of the Neural Engineering Framework (NEF; Eliasmith and Anderson, 2003), a general method for organizing realistic spiking neuron models so as to represent and transform information. This is used to derive the optimal synaptic connections (under neurological constraints) for creating our model. Next, we introduce Vector Symbolic Architectures (VSAs; Gayler, 2003), a method for efficiently encoding symbolic structures as high-dimensional fixed-length vectors. This is used to encode structured information in the cortex and to represent the IF-THEN rules themselves.

Given these tools, we then define the three anatomical components of our model (cortex, basal ganglia, and thalamus). This includes specifying the neurological parameters of the neurons involved, such as the neurotransmitters used. This is important for providing accurate timing predictions from our model, since various neurotransmitters have varying characteristic time constants.

We demonstrate our model performing three separate tasks: repeating the alphabet, repeating the alphabet starting from a particular letter, and answering questions using working memory. For each of these tasks we use exactly the same neural model; the only differences are the sensory inputs to the system.

Finally, we provide two conclusions that connect cognitive theory and neuroscience. First, we show that particular types of IF-THEN rules are more efficient to implement in spiking neurons, leading to a possible modification of standard production system-based theories. Second, we show that the time needed to select an action is determined primarily by the re-uptake rate of the neurotransmitter GABA in the basal ganglia, thus providing a neurological explanation for the 50-millisecond cognitive cycle time commonly found in behavioural results.

Neural Engineering Framework

To build a complex neural model, we need a method for determining how neurons can represent and transform information. We use the Neural Engineering Framework (NEF; Eliasmith and Anderson, 2003), which generalizes established findings on how sensory and motor neurons represent multidimensional information. This allows us to treat a group of neurons as representing a single vector of arbitrary length. By adjusting the connectivity between

groups of neurons, we can indicate how these representations should be changed over time.

The basic assumption of the NEF is that within a neural group, each neuron has a preferred value \mathbf{e} (for *encoding*) to which it responds most strongly (i.e. fires most quickly). As the difference between the actual value and the preferred value increases, this firing rate will decrease. If the value to be represented by the neurons is \mathbf{x} , this behaviour can be captured in terms of the amount of ionic current J flowing into the neuron given by Equation 1. Adjusting the neuron gain α , the background input current J_{bias} , and the preferred direction vector \mathbf{e} allows us to capture a wide range of known neural tuning curves.

$$J = \alpha \mathbf{e} \cdot \mathbf{x} + J_{bias} \quad (1)$$

In the simplest case, 100 neurons could represent a 100 dimensional vector \mathbf{x} by having each \mathbf{e} be a different unit vector in each of the 100 dimensions. This would provide a completely local representation of each value in the vector. More realistically, 100 neurons could represent one or two dimensions by having \mathbf{e} values chosen randomly (i.e. uniformly distributed around the unit hypersphere in that many dimensions). This approach has been observed in numerous areas of visual and motor cortex (e.g. Georgopoulos et al., 1986). The advantage of having more neurons than there are dimensions is that the amount of representational error can be controlled. Neurons are highly stochastic devices, but we have previously shown that the overall error is inversely proportional to the number of neurons per dimension (Eliasmith & Anderson, 2003).

Using Equation 1 to set the amount of input current to a particular neuron to represent a particular value, we can use existing models of neuron behaviour to determine the resulting spike times. There are an extremely wide variety of suitable neuron models, from Hodgkin-Huxley-type models up to extremely detailed compartmental models. For this model, we use a standard Leaky Integrate-and-Fire model, where input current causes voltage inside the neuron to gradually build up until it reaches a threshold, at which point it fires, producing a spike. Thus, given a particular vector, we can determine the resulting sequence of spikes.

We can also perform the opposite operation: given a sequence of spikes we can estimate the original vector. As shown elsewhere (Eliasmith & Anderson, 2003), this can be done by deriving the decoding vectors \mathbf{d} as per Equation 2, where a_i is the average firing rate for neuron i with a given vector \mathbf{x} , and the integration is over all values of \mathbf{x} .

$$\mathbf{d} = \Gamma^{-1} \mathbf{Y} \quad \Gamma_{ij} = \int a_i a_j d\mathbf{x} \quad \mathbf{Y}_j = \int a_j \mathbf{x} d\mathbf{x} \quad (2)$$

The resulting vectors \mathbf{d} can be used to determine an estimate of the represented value using Equation 3, where $h(t)$ is the current produced in a post-synaptic neuron by the pre-synaptic neuron firing at time $t=0$, and $t_{i,n}$ is the time that the i^{th} neuron fired for the n^{th} time.

$$\hat{\mathbf{x}}(t) = \sum_{i,n} \delta(t - t_{i,n}) * h_i(t) \mathbf{d}_i = \sum_{i,n} h(t - t_{i,n}) \mathbf{d}_i \quad (3)$$

This is an estimate that varies over time based on the individual spikes. Importantly, it is the *optimal* estimate

when under the constraint that the estimate must be built by linearly adding the effects of the post-synaptic currents caused by each spike. This is the constraint for other neurons receiving these spikes, so Equation 3 gives the optimal reconstruction of the vector by another neuron.

As a consequence of this, the decoding vectors \mathbf{d} provide an extremely important tool that is at the heart of the Neural Engineering Framework. We can use \mathbf{d} and \mathbf{e} to derive *optimal synaptic connection weights* to perform particular mathematical manipulations on the encoded information. If one group of neurons represents \mathbf{x} and we want another group to represent some particular linear transformation of this value (i.e. $\mathbf{y} = \mathbf{M}\mathbf{x}$), then we simply set the synaptic connection weights \mathbf{w} as per Equation 4.

$$\mathbf{w}_{ij} = \alpha_j \mathbf{e}_j \mathbf{M} \mathbf{d}_i \quad (4)$$

For nonlinear functions, we can modify Equation 2 to produce decoding vectors $\mathbf{d}^{f(\mathbf{x})}$ that optimally approximate any nonlinear function $f(\mathbf{x})$, as shown in Equation 5.

$$\mathbf{d}^{f(\mathbf{x})} = \Gamma^{-1} \mathbf{Y} \quad \Gamma_{ij} = \int a_i a_j d\mathbf{x} \quad \mathbf{Y}_j = \int a_j f(\mathbf{x}) d\mathbf{x} \quad (5)$$

This approach allows us to create complex neural models where we directly derive the necessary synaptic connection weights, rather than relying on a particular learning rule.

Vector Symbolic Architectures

While the NEF provides a method for representing vectors, in order to implement a cognitive model we need to represent complex symbol-like structures. That is, while we might be able to say that one particular vector represents the concept of a square, another vector represents a triangle, and another represents a particular colour, this does not address the question of how we can represent “a blue circle and a red square”.

A general approach to this problem is to use a Vector Symbolic Architecture (VSA; Gayler, 2003). There are three core ideas for all VSAs. First, each symbol is represented by a particular high-dimensional vector. For our purposes, we randomly choose these vectors, but they could also be selected based on semantic and sensory knowledge. Second, two vectors can be combined by superposition (+) to produce a new vector that is *similar* to both of the original vectors. Third, two vectors can be combined by binding (\otimes) to produce a new vector that is *dissimilar* to both of the original vectors.

This binding operation can be reversed by binding with the inverse of a vector ($\hat{\cdot}$), such that $\mathbf{A} \otimes \mathbf{B} \otimes \hat{\mathbf{B}} \approx \mathbf{A}$. These operations are similar to standard addition and multiplication in terms of being associative, commutative, and distributive.

For our model, we chose a particular VSA known as Holographic Reduced Representations (HRRs; Plate, 2003). For this, superposition is performed by vector addition and the binding operation is circular convolution. These operations can be efficiently implemented in spiking neurons using synaptic connections calculated using the NEF (Eliasmith, 2005) and Equations 4 and 5, above.

With such a system we can represent symbol trees by combining superposition and binding. For example, we can find a vector to represent “a blue circle and a red square” by performing the following calculation:

$$\text{blue} \otimes \text{circle} + \text{red} \otimes \text{square}$$

The result is a single vector of the same dimensionality as the vectors for the basic symbols (**blue**, **red**, **square**, etc.). This one vector can be interpreted as a representation of the entire structure because it is possible to extract the original components. For example, to determine which object is red, we take the whole vector and bind it with the inverse of **red**.

$$\begin{aligned} & (\text{blue} \otimes \text{circle} + \text{red} \otimes \text{square}) \otimes \text{red}^* \\ &= \text{blue} \otimes \text{circle} \otimes \text{red}^* + \text{red} \otimes \text{square} \otimes \text{red}^* \\ &\approx \text{blue} \otimes \text{circle} \otimes \text{red}^* + \text{square} \end{aligned}$$

The result is a vector that is similar to **square**, but is not exactly the same since it has an additional term superposed on it. Due to the properties of the binding operation, however, **blue** \otimes **circle** \otimes **red**^{*} will be a vector that is highly dissimilar to all of the original symbols, and can be treated as randomly distributed noise. We have previously shown how spiking neuron models can remove this noise (Stewart, Tang, & Eliasmith, 2009).

The Model

Basal Ganglia

The basal ganglia is generally believed by both neuroscientists (e.g. Redgrave et al., 1999) and cognitive scientists (e.g. Anderson et al., 2004) to be responsible for action selection. That is, given a wide variety of possible options as to what to do next, a single one must be chosen. This can be thought of as a winner-take-all mechanism: each option will have a numerical value indicating how relevant (or how beneficial) each action is in the current context, and the best of these should be chosen. Although winner-take-all mechanisms are common in neural models, there are few that adhere to the biological constraints of the basal ganglia, and none we are aware of that use realistic spiking neurons.

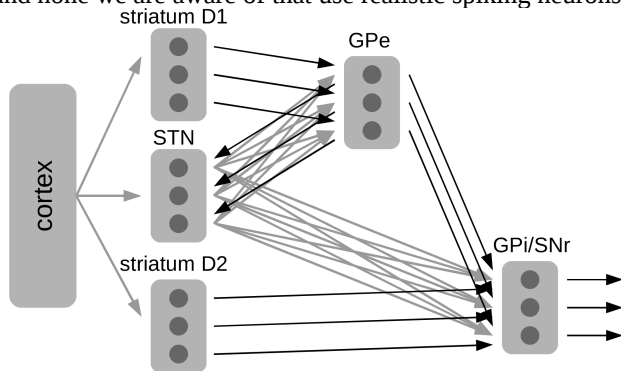


Figure 1: Basal ganglia model with three possible actions. Light lines are excitatory connections. Dark lines are inhibitory (based on Gurney et al., 2001, Figure 5).

While we have previously investigated simple mutual inhibition approaches for winner-take-all (Stewart & Eliasmith, 2009), for our current model we adapt work by Gurney, Prescott, and Redgrave (2001). As shown in Figure 1, the D1 cells in the striatum inhibit corresponding cells in the globus pallidus internal (GPi) and substantia nigra reticulata (SNr), while the subthalamic nucleus (STN) sends a broad excitatory signal to the GPi/SNr and globus pallidus external (GPe). The GPe and the D2 cells in the striatum act as a control signal on the excitation from the STN, adjusting it so that the correct amount of excitation is provided to select a single action. Each of these connections is well-documented anatomically, and the model's behaviour matches neurological results in rats and monkeys both with and without particular lesions (Gurney et al., 2001).

However, the Gurney et al. model uses idealized piecewise-linear non-spiking neurons that respond instantly without any random variation to changes in their inputs. We thus adapt their model, replacing individual idealized neurons with groups of realistic leaky-integrate-and-fire (LIF) spiking neurons. For our neurons, the membrane time constant (τ_{RC} ; controlling the amount of current leaking out of the neuron) was fixed at 20ms, and the α and J_{bias} values were randomly chosen constrained by the reported response properties given by Gurney et al., including background firing rates of 60-80Hz and maximum firing rates of 400Hz. All synaptic connections were derived using Equation 4. We use 20 neurons to replace one ideal neuron (circle in Figure 1), so 100 neurons are needed per possible action.

The behaviour of this model is shown in Figure 2. The inputs to the model (top) are the desirability of three different actions. The firing response of the output of the basal ganglia (bottom) is shown as these inputs change over time. As in the actual basal ganglia, the output is inhibitory, so an action is selected by *turning off* the appropriate output neurons, stopping them from performing their inhibition. It should be noted that this output lags behind the input due to the time constants of the post-synaptic current caused by different neurotransmitters. In this case, the excitatory connections use glutamate with AMPA receptors (2ms; Spruston et al., 1995), and the inhibitory connections use GABA (10ms; Gupta et al., 2000).

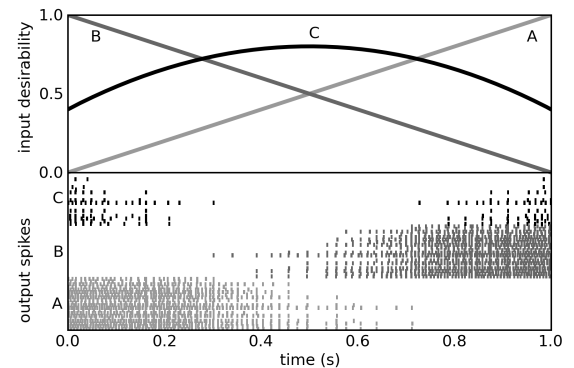


Figure 2: Inputs and outputs (GPi) of our basal ganglia model. The largest valued input consistently causes the corresponding output neurons to stop spiking.

Cortex

For the tasks under consideration in this paper, we need a visual area (for representing the current visual scene), a motor area (for producing outputs from the model), and a working memory (for storing a statement and questions to be answered). Each of these is implemented as 10,000 spiking neurons, storing a 250 dimensional VSA vector as per the NEF. We present stimuli to our model by injecting current into the visual area (**V** in Figure 3) using Equation 1. We can examine the contents of any area of the cortex by decoding the activation (Equation 3) and measuring the similarity (dot product) between the resulting vector and an ideal calculated vector. The closer this value is to 1.0, the more accurate the representation.

To perform general purpose tasks (such as question answering), our model contains two working memory areas: **A** and **B**. In order to maintain information over time, these areas contain connections back to themselves as per Equation 4 where **M** is the identity matrix. This forms the basis of an integrator model of memory, which has previously been used to model somatosensory working memory (Singh & Eliasmith, 2006). Areas **A** and **B** are also connected to two other neural groups **C** and **D** such that $\mathbf{C} = \mathbf{A} \otimes \mathbf{B}$ and $\mathbf{D} = \mathbf{A} \otimes \mathbf{B}'$. These connection weights are defined using Equation 5, where $f(\mathbf{x})$ is the circular convolution (see Eliasmith, 2005 for details). This allows the system to compute the VSA operations that are needed to perform symbol manipulation.

Thalamus

The only mechanism in our model for modifying the contents of the working memory areas and the motor areas is the thalamus. If the thalamic areas are all zero then no information is transferred between cortical areas. If the thalamic area corresponding to working memory **A** is set to some value (via the basal ganglia), then this value will be sent to cortical area **A**, using synaptic connections from Equation 4 with **M** as the identity matrix. Crucially for information transfer, if the thalamic area controlling the connection between **V** and **A** is set to **X**, then the value $\mathbf{V} \otimes \mathbf{X}$ will be sent to **A**.

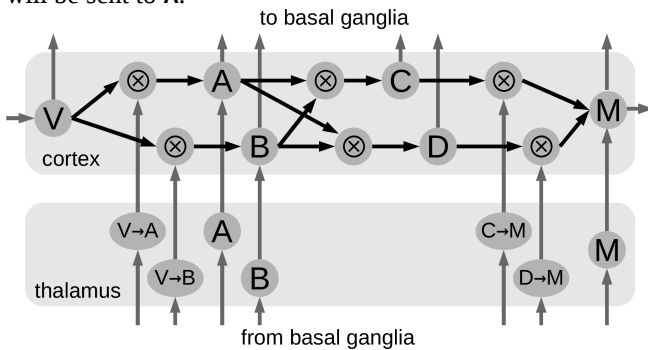


Figure 3: Thalamus and cortex model. Circles are 10,000 neurons representing 250 dimensional vectors (**V**=vision; **M**=motor; **A**,**B**,**C**,**D**=working memory). \otimes are 40,000 neurons computing the binding operation.

Modelled Tasks

Fixed Sequences of Actions

The simplest task to perform with this model is sequentially going through a list of items, such the alphabet. We implement this by defining 25 rules of the following form:

IF working memory contains **letter+A**

THEN set working memory to **letter+B**

We create the IF portion of a rule by setting the synaptic connections between the working memory area of cortex and the striatum and sub-thalamic nucleus. Each component of the basal ganglia has a group of neurons corresponding to each rule (the dark circles in Figure 1). We set the input synaptic weights using Equation 4, where **M** is the vector corresponding to the IF portion of the rule (**letter+A**).

To implement the THEN portion of the rule, we set the synaptic connections at the output of the basal ganglia. In this case, we create a group of neurons that connect to the thalamic neurons that feed to working memory. We again use Equation 4 to set these weights, with **M** set to be the vector corresponding to **letter+B**. We then connect the group of neurons in the GPi that correspond to this rule to these new neurons. Because GPi is inhibitory, this connection will cause the new neurons to not fire at all, except in the case that the action selection system in the basal ganglia chooses this particular action. In that case, the inhibition will be turned off (as those GPi neurons will stop spiking), allowing **letter+B** to be sent to working memory. This in turn will cause the next rule to be selected, and so on. It should be noted that our model does not yet include the phonological loop, so any timing influence it may have on producing this sequence is not taken into account.

To test the model, we initialize it by forcing current into the working memory neurons as per Equation 1 such that they will represent **letter+A**. After this, all subsequent activity is due to the interconnections between neurons. Figure 4 shows the model correctly following the alphabet sequence. From the spiking pattern we see that the correct action for each condition is successfully chosen by turning off the appropriate inhibitory neurons in the GPi.

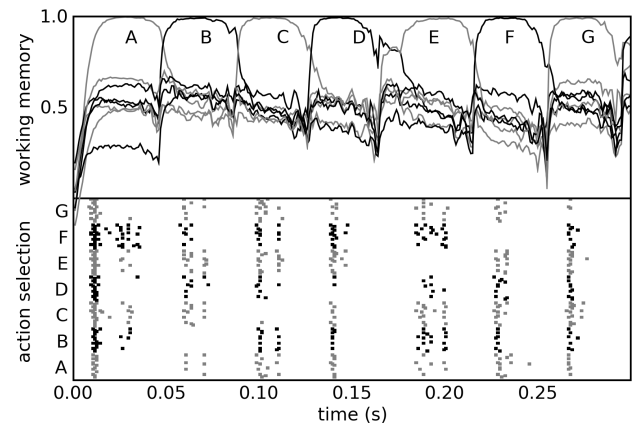


Figure 4: Contents of working memory (top) and spiking output from GPi indicating the action to perform (bottom).

Variables and Generic Rules

The previous section demonstrated that our model is capable of implementing rules where a specific pattern is sent to a specific part of cortex. While these sorts of rules may account for some kinds of highly specialized behaviour, most symbolic cognitive architectures assume that it is possible to have *general-purpose* rules. That is, these rules can contain variables, such as the following, where **?X** represents an unknown variable:

IF visual cortex contains **letter+?X**

THEN set working memory to **letter+?X**

The presence of this sort of rule in addition to the ones in the previous section would allow the model to start going through the alphabet starting from any letter. We would simply present the particular letter we wanted it to start from to the visual cortex (**letter+F**) and it would copy this value to working memory and continue from there.

While the above method is the standard approach for expressing this sort of rule, in order to implement it in our model, we need to slightly reformulate it as the following:

IF visual cortex contains **letter+?X**

THEN copy visual cortex to working memory

This rule has exactly the same effect as the first one. To implement it, we use the same approach as in the previous section. The synaptic connection weights for the inputs to the basal ganglia are set using Equation 4 with **M** as the vector for **letter**. For the output, instead of connecting to the parts of the thalamus which send information directly to cortical areas, we connect to the neural group which gates connections between these cortical areas. If we set this to the identity vector **I**, then working memory will now contain **V⊗I=V**. This has the effect of routing information between cortical areas.

The result of this model when **letter+F** is placed in the visual cortex is shown in Figure 5. The model correctly starts repeating the alphabet from F. Changing the visual stimulus to some other letter will start from there, demonstrating that the rule can apply to multiple situations.

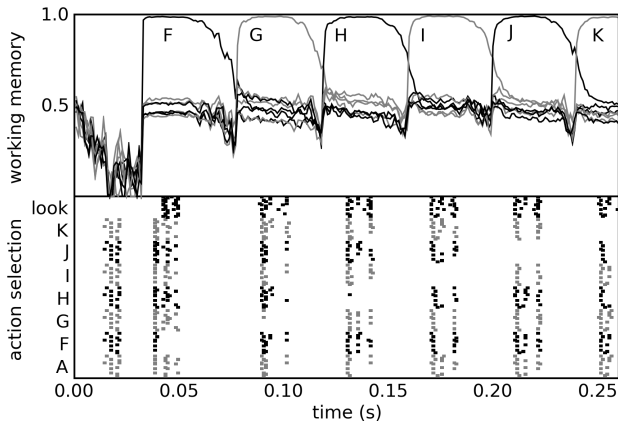


Figure 5: Contents of working memory (top) and spiking output from GPi indicating the action to perform (bottom). The *look* action takes information from visual cortex (in this case, **letter+F**) and routes it to working memory.

Question Answering

For the final task, we consider question answering. We perform this by first presenting the model with a symbolic statement such as the following:

statement + blue⊗circle + red⊗square

This would indicate a blue triangle and a red square are all in the visual field. The statement is presented to visual cortex for 50ms, and it will use the following rule to move it into working memory, as in the previous section:

IF visual cortex contains **statement+?X**

THEN copy visual cortex to working memory

After the statement is shown for 50ms, we stop stimulating visual cortex for another 50ms. This means that the system must successfully keep the statement in working memory over this time. After this time, we present a question to the visual cortex, such as the following:

question + red

A separate rule is defined for dealing with this situation:

IF visual cortex contains **question+?X**

THEN copy visual cortex to working memory B and also copy from working memory D to motor cortex

This rule copies the question to a separate area of working memory (B). As described previously (see Figure 3), this area allows a vector to be combined with the current contents of working memory. Furthermore, this rule also copies information from a third area of working memory (D) to the motor cortex. Since area D is connected to A and B so as to store the result of convolving area A (the statement) with the inverse of area B (the question), it should contain the answer to the question.

The results of this model answering two different questions from the same remembered statement are given in Figure 6. These two generic rules can answer any question provided in this format. Previous work on the capabilities of neural implementations of VSAs (Stewart, Tang, & Eliasmith, 2009) indicates that this system will scale well to 8 or more terms in a statement, out of a total vocabulary of 100,000 possible terms.

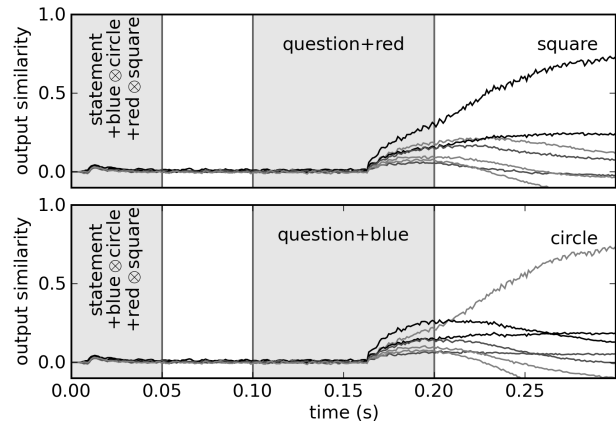


Figure 6: Answering two different questions starting from the same statement. The similarity between the contents of motor cortex and 7 possible answers is shown. The correct answer is chosen in both cases.

Implications

The model presented here helps to bridge the gap between cognitive science and neuroscience. It allows us to transform symbolic rules (the basis of much of cognitive theory) into specifications for the synaptic connectivity between neurons in cortex, basal ganglia, and thalamus. The resulting models give detailed predictions about the timing of events and the spiking behaviour of the neurons involved. With such models, we can also predict performance accuracy and the effects of various types of neurological damage.

The model also addresses a long-standing concern in cognitive science as to how neurons can possibly support the rich cognitive capabilities that seem clearly based on symbols and symbol manipulation. Specifically, we suggest that a VSA approach to representing symbols can be implemented in spiking neurons, and that these representations can be manipulated in a controlled and generic manner. We are aware of no other neural model with this flexibility, scalability, and connection to the underlying neurophysiology.

Rule types

Bridging cognitive science and neuroscience provides more than a mere neural implementation of cognitive theory. For our model, it has also suggested possible modifications to cognitive theory. When implementing the rules, we changed them from including explicit variables into commands to transform and copy the information currently represented in various parts of visual and working memory. If our future applications of this model continue to find this approach to rule definition sufficient for a wide variety of cognitive tasks, then we would argue this may be a more suitable framework for expressing cognitive rules than the standard variable-binding approach.

Timing

Our model is also highly constrained by known neurological data; the characteristics of the neurons involved and their connectivity are based on empirical results. As such, we can predict results that were previously derived purely by parameter fitting. For example, in most production system models of cognition (Soar, GOMS, EPIC, ACT-R, etc.), a certain amount of time is needed to select and apply an action. Based on empirical evidence, this is normally fixed to be 50 milliseconds (e.g. Anderson et al., 1995).

As can be seen in Figure 4 and Figure 5, our model requires just under 50 milliseconds to select and apply an action. While the median time needed is 44 milliseconds, the mean time for our current model is 48 milliseconds, due to the model occasionally repeating a step. These times are not affected by the size of our model, but can be changed by adjusting the time constant for the inhibitory neurotransmitter GABA in the basal ganglia. We currently use a value of 10ms (Gupta et al., 2000), and are seeking more detailed results from this area of the basal ganglia.

Conclusion

We presented a large-scale (373,000 spiking neuron) model capable of exhibiting rule-like behaviours such as question answering. By representing the conditions for applying inference rules as VSA vectors, and by representing the effects of those rules as vector transformations between cortical areas, we have shown a generic method for controlling neurally realistic cognitive systems.

Our ongoing work explores the broader capabilities of this model, including scaling up the number of rules (only 100 neurons need to be added per rule), and exploring the accuracy of the question answering as the vocabulary size increases. Other neural areas can also be added, including full vision and motor systems, as well as long-term memory.

References

- Anderson, J. R., Bothell, D., Byrne, M. D., Douglass, S., Lebiere, C., & Qin, Y. (2004). An integrated theory of the mind. *Psychological Review* 111(4), 1036-1060.
- Eliasmith, C. (2005). Cognition with neurons: A large-scale, biologically realistic model of the Wason task. 27th Annual Meeting of the Cognitive Science Society.
- Eliasmith, C. & Anderson, C. (2003). *Neural Engineering: Computation, representation, and dynamics in neurobiological systems*. Cambridge: MIT Press.
- Gayler, R. (2003). Vector symbolic architectures answer Jackendoff's challenges for cognitive neuroscience. *International Conference on Cognitive Science*.
- Georgopoulos, A.P., Schwartz, A., & Kettner, R.E. (1986). Neuronal population coding of movement direction. *Science*, 233, 1416-1419.
- Gupta, A., Wang, Y., & Markram, H. (2000). Organizing Principles for a Diversity of GABAergic Interneurons and Synapses in the Neocortex. *Science* 287(5451), 273-278.
- Gurney, K., Prescott, T., & Redgrave, P. (2001). A computational model of action selection in the basal ganglia. *Biological Cybernetics* 84, 401-423.
- Plate, T. (2003). *Holographic reduced representations*. Stanford, CA: CSLI Publication.
- Redgrave, P., Prescott, T., & Gurney, K. (1999). The basal ganglia: a vertebrate solution to the selection problem? *Neuroscience* 86, 353-387.
- Singh, R. & Eliasmith, C. (2006). Higher-dimensional neurons explain the tuning and dynamics of working memory cells. *J. Neuroscience* 26, 3667-2678.
- Spruston, N., Jonas, P., & Sakmann, B. (1995). Dendritic glutamate receptor channel in rat hippocampal CA3 and CA1 pyramidal neurons. *J. Physiology* 482, 325-352.
- Stewart, T. C., & C. Eliasmith (in press). Compositionality and biologically plausible models. *Oxford Handbook of Compositionality*.
- Stewart, T., & Eliasmith, C. (2009). Spiking neurons and central executive control: The origin of the 50-millisecond cognitive cycle. *ICCM 2009*, Manchester.
- Stewart, T., Tang, Y., & Eliasmith, C. (2009). A biologically realistic cleanup memory: Autoassociation in spiking neurons. *ICCM 2009*, Manchester, UK.