Connecting Biological Detail with Neural Computation: Application to the Cerebellar Granule-Golgi Microcircuit

Andreas Stöckel (astoecke@uwaterloo.ca)[1]
Terrence C. Stewart (terrence.stewart@nrc-cnrc.gc.ca)[2]
Chris Eliasmith (celiasmith@uwaterloo.ca)[1]
[1]Centre for Theoretical Neuroscience, University of Waterloo
[2]National Research Council of Canada, University of Waterloo Collaboration Centre
200 University Avenue West, Waterloo, ON N2L 3G1 Canada

Abstract

Neurophysiology and neuroanatomy limit the set of possible computations that can be performed in a brain circuit. Although detailed data on individual brain microcircuits is available in the literature, cognitive modellers seldom take these constraints into account. One reason for this is the intrinsic complexity of accounting for mechanisms when describing function. In this paper, we present multiple extensions to the Neural Engineering Framework that simplify the integration of low-level constraints such as Dale's principle and spatially constrained connectivity into high-level, functional models. We apply these techniques to a recent model of temporal representation in the Granule-Golgi microcircuit in the cerebellum, extending it towards higher degrees of biological plausibility. We perform a series of experiments to analyze the impact of these changes on a functional level. The results demonstrate that our chosen functional description can indeed be mapped onto the target microcircuit under biological constraints. Further, we gain insights into why these parameters are as observed by examining the effects of parameter changes. While the circuit discussed here only describes a small section of the brain, we hope that this work inspires similar attempts of bridging low-level biological detail and high-level function. To encourage the adoption of our methods, we published the software developed for building our model as an open-source library.

Keywords: biologically plausible spiking neural networks; Dale's principle; Neural Engineering Framework; delay network; cerebellum; NengoBio

Introduction

Human cognition is ultimately grounded in neurophysiological processes. As suggested by Marr's "levels of analysis" (Marr & Poggio, 1976), cognitive scientists tend to implement models of cognition at algorithmic and computational levels, without explicitly taking limitations of the underlying neural substrate into account (Eliasmith & Kolbeck, 2015).

Depending on the hypothesis that is being explored, ignoring biological detail can be reasonable. Yet, a closer look at biology may help in two complementary ways. First, we can *validate* hypotheses about cognition by determining whether a particular algorithm can be implemented using the constraints of the biological neural network in question. Second, we can *generate* new hypotheses by asking what class of algorithms a particular neural network could support.

We believe that cognitive modelling must ultimately embrace a combination of these two approaches to narrow down the vast space of possible cognitive science theories and to direct research attention within that space. However, a central roadblock to the adaptation of such methods is the availability of modelling tools that make it possible to specify detailed biological constraints (e.g., neural response curves, spike rates, synaptic time constants, connectivity patters) while still being abstract enough to facilitate the specification of high-level cognitive function.

One approach designed to help bridge this gap is the Neural Engineering Framework (NEF; Eliasmith & Anderson, 2003), in conjunction with the related Semantic Pointer Architecture (SPA; Eliasmith, 2013). Up until recently however, it has been unclear how to incorporate certain biological constraints that are often described in the neuroscience literature into NEF networks. For example, and despite initial progress in this direction (Parisien, Anderson, & Eliasmith, 2008), accounting for Dale's principle with purely excitatory and inhibitory neuron populations, as well as incorporating spatial connectivity constraints, has been relatively challenging with the existing NEF-based software tool, Nengo (Bekolay et al., 2014). Furthermore, certain aspects of the NEF, such as the neural bias currents J_{bias} have a somewhat unclear relationship to biology.

In this paper, we describe recent advances in modeling techniques that partially alleviate the shortcomings of the NEF listed above. We then apply these methods to extend a previous biologically detailed model of the Granule-Golgi circuit in the cerebellum (Stöckel, Stewart, & Eliasmith, 2020). In this way, we validate that—at least for the set of constraints considered in our experiments—the Granule-Golgi circuit is indeed well-suited to implementing a specific algorithm for encoding temporal information using basis functions. Furthermore, we generate possible hypotheses as to why various biological parameters (such as the sparse connectivity patterns and the time constants of the neurotransmitters) are as observed. We have released the open-source tool we developed to encode these constraints as an add-on to Nengo called NengoBio.[1]

The remainder of this paper is structured as follows. We first review the high-level function we hypothesise could be implemented in the Granule-Golgi circuit, as well as the particular neurophysiological constraints of this network. We then discuss five neural network implementations with an increasing amount of biological detail, along with the corresponding extensions to the NEF. Finally, we perform a series of experiments that explore the impact of individual parameters on the performance of the increasingly realistic system.

[1] See `https://github.com/astoeckel/nengo-bio`.

Background

In order to explore the consequences of adding biological details to a neural system, we need to choose the desired computation that the neural system should ideally perform. In machine learning, the simplest artificial neural networks are purely feed-forward, i.e., they possess no backwards-directed or *recurrent* connections. It is well-known that such neural networks are universal function approximators (Hornik, Stinchcombe, & White, 1989). That is, given enough neurons, any function $f(x) = y$ can be implemented as a neural network by simply having a single hidden layer of neurons that receives x as an input (via a set of input weights) and produces y as an output (via a set of readout weights).

Neurobiological systems differ from the artificial neural networks mentioned above in two key aspects. First, they are intrinsically dynamical systems, i.e., input and output are functions over time. Second, they often include recurrent connections. As has been shown by Eliasmith and Anderson (2003), adding recurrent connections along with their associated synaptic dynamics allows for the creation of neural networks that approximate any differential equation of the form $\frac{\mathrm{d}\mathbf{m}}{\mathrm{d}t} = f(\mathbf{m}, u, t)$. Again, with a sufficient number of neurons and the corresponding connectivity, such differential equations can be approximated to any desired degree of accuracy. Our goal in this paper is to explore how well such computations can be performed in the presence of other biological constraints.

The Delay Network

As a benchmark for evaluating this performance, we have chosen the following linear differential equation.

$$\begin{aligned}
&\theta\dot{\mathbf{m}} = \mathbf{A}\mathbf{m} + \mathbf{B}u\,, \quad \mathbf{A} \in \mathbb{R}^{q\times q}, \mathbf{B} \in \mathbb{R}^{q}, \mathbf{m} \in \mathbb{R}^{q} \\
&(\mathbf{A})_{ij} = \begin{cases} (2i+1)(-1) & i < j\,, \\ (2i+1)(-1)^{i-j+1} & i \geq j\,, \end{cases} \\
&(\mathbf{B})_i = (2i+1)(-1)^i\,.
\end{aligned} \tag{1}$$

This equation is derived by taking the Padé approximate of the continuous-time delay $F(s) = e^{-\theta s}$. As such, this differential equation stores the past history of its inputs the state variable $\mathbf{m}$ (Voelker & Eliasmith, 2018). That is, given $\mathbf{m}$ at any particular point in time t, it is possible to recover an approximate value of u at a previous point in time $t - \theta'$ for $0 \leq \theta' \leq \theta$:

$$\hat{u}(t - \theta') = \sum_{\ell=0}^{q-1} m_\ell d_\ell(\theta')\,, \qquad \text{where } d_\ell = \tilde{P}_\ell\left(\frac{\theta'}{\theta}\right)\,, \tag{2}$$

where $\tilde{P}_\ell$ is the shifted Legendre polynomial of degree ℓ.

Another way to think of this system is that it encodes the past history of its inputs using a set of *temporal basis functions*. The particular temporal basis functions that are used are the Legendre polynomials, because they have been shown to be optimal for encoding such temporal memory (Voelker & Eliasmith, 2018). Of course, some information is lost in this process, and this is controlled by the dimensionality of the state variable $\mathbf{m}$, which is a q-dimensional vector. As q increases, more details (i.e., higher frequencies) about the past are kept in $\mathbf{m}$. The neural implementation of this computation is called a "Delay Network" (DN), and is also the core part of a novel machine learning algorithm known as the Legendre Memory Unit, which has been shown to outperform LSTMs on several benchmark tasks (Voelker, Kajić, & Eliasmith, 2019).

However, if we use biologically constrained neural networks to approximate this operation, then the actual computation performed by the system, and hence the quality of the time window encoded in $\mathbf{m}$, may be different. We can thus use the ideal computation expressed in eq. (1) as a benchmark.

In the following, we build various approximations of the DN using different biological constraints, systematically provide them with inputs, and evaluate their performance in terms of how well the input history can be recovered. In all cases, we compute the optimal recurrent and readout connection weights independently for each target population to approximate eq. (1) given the biological constraints. This avoids the need for a stochastic training process, such as backpropagation.

Neural Computation in the Cerebellum

The biological system of particular interest in this paper is the cerebellum. Not only is it well-studied and highly regular in its structure, but there are also reasons to believe that it does compute something akin to the operation performed by the Delay Network. Behaviourally, the cerebellum is known to be vital for some delay conditioning tasks, such as eye-blink conditioning (McCormick et al., 1981). In eye-blink conditioning, a sensory cue (e.g., an audio tone) is given before a puff of air into the eye. After some time, animals consistently learn to blink the correct amount of time after the sensory cue, such that the eye is closed when the puff actually happens.

There is no current consensus on how exactly the cerebellum learns these delays. One theory is that the responses observed in tasks such as eye-blink conditioning rely on intrinsic properties of the Purkinje cells (Lusk, Petter, MacDonald, & Meck, 2016). Another theory—and this is what we assume in this paper—is that the Granule-Golgi microcircuit is responsible for computing a temporal basis function representation (cf. Dean, Porrill, Ekerot, & Jörntell, 2010; Rössert, Dean, & Porrill, 2015), from which arbitrary delays can be linearly decoded. This is similar in principle to the Legendre basis functions used in the Delay Network. Indeed, Stöckel et al., 2020 show an initial implementation of eye-blink conditioning using this approach, but with less biological detail than is explored in this paper. We believe the findings in this paper provide another strong argument that the biology of the Granule-Golgi circuit is very well suited for implementing some kind of temporal basis function generation.

The Granule-Golgi microcircuit

The cerebellum contains about 50 billion granule cells, making them the most common type of neuron in the entire human brain. They are tiny cells that receive input from pre-cerebellar nuclei (PCN) via "mossy fibres" and project via "parallel fibres" onto the Purkinje cells. Granule cells also have interneurons interspersed amongst them, known as Golgi cells, forming an inhibitory feedback loop with the granule cells. That is, granule cells excite Golgi cells, and Golgi cells inhibit granule cells (Ito, 2010). Notably, each granule cell on average receives input from only four mossy fibres, as well as one or two Golgi cells (Chadderton, Margrie, & Häusser, 2004). These numbers are known in the literature as the *convergence* of a projection. Furthermore, the connectivity between Golgi and granule cells is spatially constrained, i.e., Golgi cells only connect to granule cells in their vicinity. The ratio of granule to Golgi cells is about 430:1 (D'Angelo et al., 2013).

Levels of Biological Detail

To demonstrate our approach of adding biological detail, we present five models of the Granule-Golgi microcircuit of increasing complexity. While the first model is merely an abstract implementation of eq. (1), the final model respects spatial sparsity, convergence, tuning curves, and neurotransmitter constraints. All models are depicted in Fig. 2. In all cases, the scalar input u is received from one hundred spiking Leaky Integrate-and-Fire (LIF) neurons with randomly chosen tuning curves representing the PCN (see Model B for details).

Model A: "Direct" Implementation For this model, we directly solve the differential equation in eq. (1) by integration. That is, we have a single layer of "neurons" that are pure integrators (i.e., no non-linearity). The matrix **A** describes the recurrent connection weights, and **B** the input connection weights. This model does not distinguish between the granule and Golgi cells, and does not include details such as individual neurons or spikes. Instead, it focuses on the high-level theory of what the system is computing.

Model B: Single Population We now replace the integrators with a single layer of 200 spiking Leaky Integrate-and-Fire (LIF) neurons. These neurons form a distributed representation of **m** using a population code. Each neuron i is parametrized by a randomly chosen preferred stimulus vector $\mathbf{e}_i$ (for *encoder*), gain α_i and bias current J_i^{bias}, resulting in a desired response (i.e., tuning curve) for each neuron:

$$a_i(\mathbf{m}) = G[J_i(\mathbf{m})] = G[\alpha_i(\mathbf{e}_i \cdot \mathbf{m}) + J_i^{\mathrm{bias}}], \qquad (3)$$

where G is neural response curve of the LIF neuron model. The parameters α and J_{bias} are randomly chosen from a distribution that ensures a maximum firing rate of 50 Hz to 100 Hz, consistent with biological recordings of granule cells (Chadderton et al., 2004). We then use least-squares to solve for optimal input and recurrent connection weights that result in these desired tuning curves while implementing the equivalent calculation as in Model A. Importantly, when solving for the recurrent connection weights, we also take into account the synaptic filter, which we model here as a decaying exponential (i.e., a low-pass). This is the standard process in the NEF (Eliasmith & Anderson, 2003).

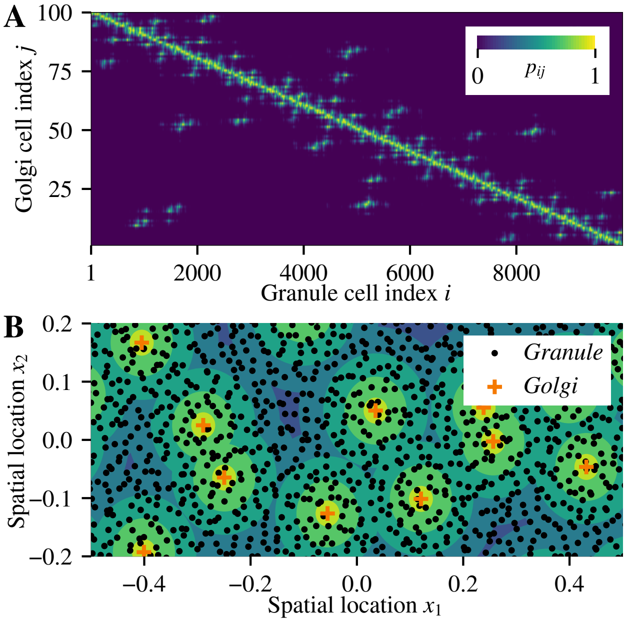

Figure 1: Spatial connectivity constraints. **(A)** Normalised connection probabilities p_{ij} for $\sigma = 0.25$. **(B)** Spatial organisation of the Golgi and granule cells. The background depicts the cumulative density of the Golgi to granule connection probability for a virtual Granule cell at each location (same colors as in **A**).

Model C: Inter-neurons As a next step, we separate the single layer of neurons into two separate populations corresponding to the Golgi and granule cells, reflecting the actual biology of the cerebellum (see above). This introduces two separate synaptic filters which need to be taken into account when solving for the connection weights that best approximate eq. (1). Furthermore, to at least approximate the fact that there are far fewer Golgi cells than granule cells, we use 20 Golgi cells and 200 granule cells.

Model D: Inhibition and Excitation So far, we have not accounted for Dale's principle, i.e., Golgi cells being purely inhibitory, and granule cells being purely excitatory. We handle this by switching to the non-negative least-squares problem described in Stöckel & Eliasmith, 2019. For each post-neuron i we minimize

$$\min_{\mathbf{w}_i^+, \mathbf{w}_i^-} \sum_{k=1}^{N} \left(\mathbf{w}_i^+ \cdot \mathbf{a}_k^+ - \mathbf{w}_i^- \cdot \mathbf{a}_k^- - J_i(\mathbf{m}_k)\right)^2 \text{ w.r.t. } \mathbf{w}_i^+, \mathbf{w}_i^- \geq 0,$$

where $\mathbf{a}_k^+$, $\mathbf{a}_k^-$ are the excitatory and inhibitory pre-activities for sample k, $\mathbf{w}^+$, $\mathbf{w}^-$ are the connection weights for excitatory and inhibitory pre-neurons, and $J_i(\mathbf{m}_k)$ is the current required to represent the desired value $\mathbf{m}_k$ as defined in eq. (3).

Model E: Sparse connectivity and activity For this model, we add in realistic constraints on how connected the neurons are. The previous models used all-to-all connections, whereas for this model, we only allow a subset of those connections to be non-zero. This applies to both the input to the Golgi-Granule system and for the recurrent connections within the granule cells. In particular, we account for the granule cell convergence numbers by randomly selecting five PCN and five Golgi cells as pre-neurons—this number is slightly larger than the number reported above, since, as we discuss below, the number of pre-neurons places a strict upper limit on the connectivity (see below for more details). Given this extremely sparse connectivity, we increase the number of neurons in the simulation to 10 000 granule and one hundred Golgi cells, which is closer to the ratio observed in nature.

To account for spatially imposed connectivity constraints, we assign a location $\mathbf{x}$ in $[-1,1]^2$ to each neuron. The probability p_{ij} of a post-neuron i to receive a connection from a pre-neuron j is proportional to $\exp\left(-\|\mathbf{x}_i - \mathbf{x}_j\|^2/\sigma^2\right)$ (Fig. 1).

Finally, the input representation in the PCN cells was made more sparse by adjusting the tuning curves of the input neurons. Neural recordings indicate that there is very little input activity when no stimulus is present (Chadderton et al., 2004), while, per default, the randomly-chosen NEF tuning curves result in many neurons being active when representing the value $u = 0$.

Experiments and Results

To evaluate the effects of these biological details, we systematically generate two different types of input $u(t)$, feed those into the network and record the resulting activity. In particular, we present results with a periodic pulse input of varying pulse width t_{on} and band-limited white noise of varying bandwidth B. These are meant to depict typical sorts of inputs that may arise in experimental situations (pulses) and more real-world situations (band-limited white noise). We then determine how accurately the past history of u over the window θ can be recovered from the resulting network activity via optimal linear readout weights. We use $\theta = 0.4\,\mathrm{s}$ in all simulations; each individual experiment simulates the network for $T = 10\,\mathrm{s}$. The error is measured as the RMSE of the reconstruction divided by the RMS power of the input signal itself (normalized RMSE, or NRMSE).

The overall results for all five models are shown in Fig. 3. This shows the average reconstruction error for varying inputs (horizontal axis) and for varying time delays (vertical axis) over ten trials. An example run of Model E (the model with the most biological detail) is shown in Fig. 4. The different decoded output lines (bottom) are all based on the neural activity (middle), but with different readout weights. These approximate the input value u with various time delays over the entire time window, from the immediate input right now ($\theta'/\theta = 0$) to θ seconds ago ($\theta'/\theta = 1$).

As can be seen in Fig. 3, the network successfully functions as a method for encoding the temporal pattern of input data over the desired window of time θ. Adding more biological detail decreases the accuracy with which the system approximates eq. (1), but most of the information is still encoded. The input pulses (Fig. 4A) show that the reconstruction is a smoother version of the input; the system is not good at representing sudden changes, and this is the main source of noise in the reconstruction. This is as expected from using smooth Legendre polynomials as temporal basis functions.

Furthermore, we note that there is a peak in accuracy when decoding data from $\theta' = 70\,\mathrm{ms}$ in the past ($\theta'/\theta = 0.175$), and this peak is more pronounced as more biological detail is added. This corresponds to the neurotransmitter time constant $\tau = 70\,\mathrm{ms}$ we use for all connections, and which is based on a first-order low-pass fit to the Granule-Golgi dynamics reported in Dieudonné (1998). Importantly, we can use the model to determine what the accuracy would be like if we changed this value. This is shown in Fig. 5A. Interestingly, the performance of the system is best in the range between 50 ms to 60 ms, which is relatively close to what we observe in nature. Both smaller and larger values lead to an increase in error.

As discussed above, a striking feature of the cerebellar microcircuitry is the low granule cell convergence. One possible hypothesis is that these numbers are a trade-off between minimizing connectivity and the overall performance of the resulting system. In our model, we can test this hypothesis by systematically varying the number of pre-neurons the solver has access to. Results are shown in Figs. 5B and C. The performance of the system does improve with larger limits, yet plateaus at still relatively small convergences. More importantly, as mentioned above, the specified desired convergence solely controls the number of *potential* pre-neurons. Since the neural weight solver may still set a weight to zero, these convergence numbers are strict upper limits. Measuring the actual convergence in the PCN to granule connections (Fig. 5D), we see a peak at one to three PCN neurons connecting to each granule cell. This peak is almost independent of the desired convergence number and close to observed averages.

Discussion

We successfully mapped a high-level, mathematical function onto a brain microcircuit while incorporating biological constraints. This process was simplified by the ability of our modeling tool to automatically account for Dale's principle, spatial constraints, as well as convergence numbers.

Our results show that the Granule-Golgi circuit could in principle implement a temporal basis function representation, which is in agreement with existing hypotheses about cerebellar function. Measurements from our model could be used to generate hypotheses about the kind of electrophysiological data we would expect to find, if this function was indeed realised in the brain. Having access to low-level biological parameters *in silico* furthermore facilitates the exploration of physiological changes that are difficult to achieve experimentally *in vivo*. As discussed above with respect the to synaptic time constants and convergence numbers, this allows us to investigate why certain parameters are as observed.

A key difference of our approach to existing models of the Granule-Golgi circuit (such as Rössert et al., 2015), is that

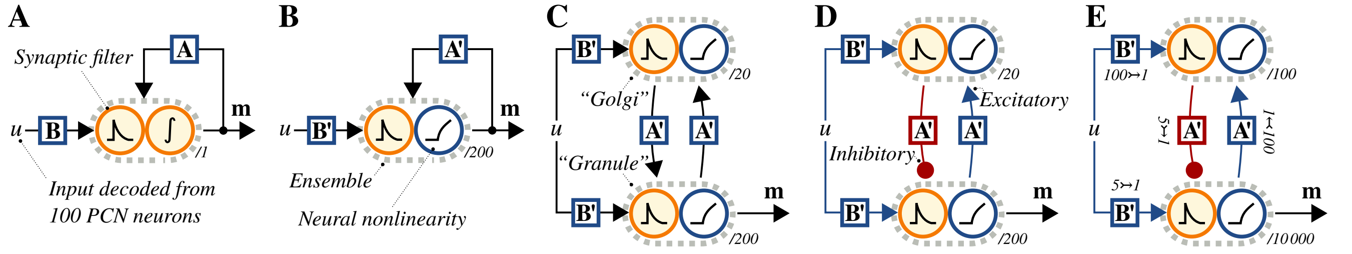

Figure 2: Network types used in our experiments. **(A)** "Direct" implementation with an optimal integrator. **(B)** Using the synaptic filter of a single population of spiking neurons for temporal integration. **(C)** Inter-neuron population in the recurrent path. **(D)** Same as **C**, but accounting for Dale's principle. **(E)** Same as **D**, but with more detailed biological constraints (see text).

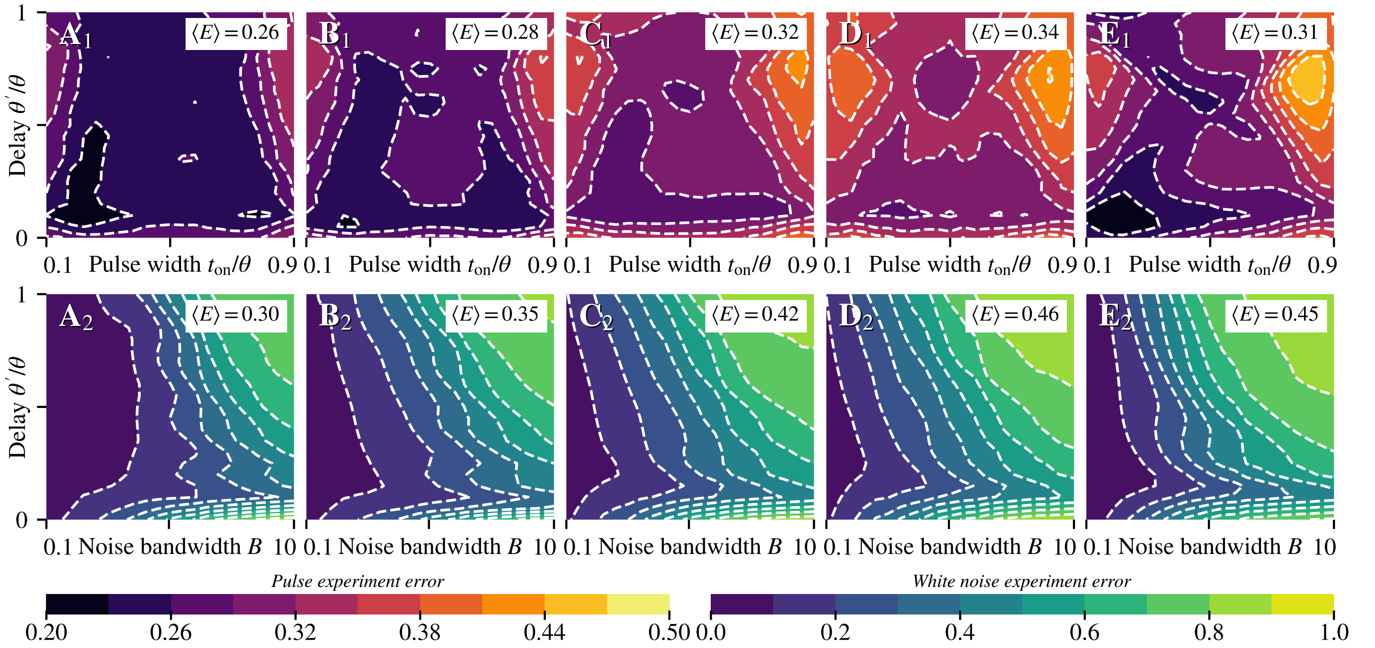

Figure 3: Delayed signal reconstruction error for different types of input signals, delays, and network types. All error values are expressed as RMSE divided by the average RMS of the input signal over ten trials. Columns correspond to the network types in Fig. 2 above. *Top row:* Reconstruction error for rectangle pulse signals of varying width. *Bottom row:* Reconstruction error for a band-limited white noise input signal with varying band-limit.

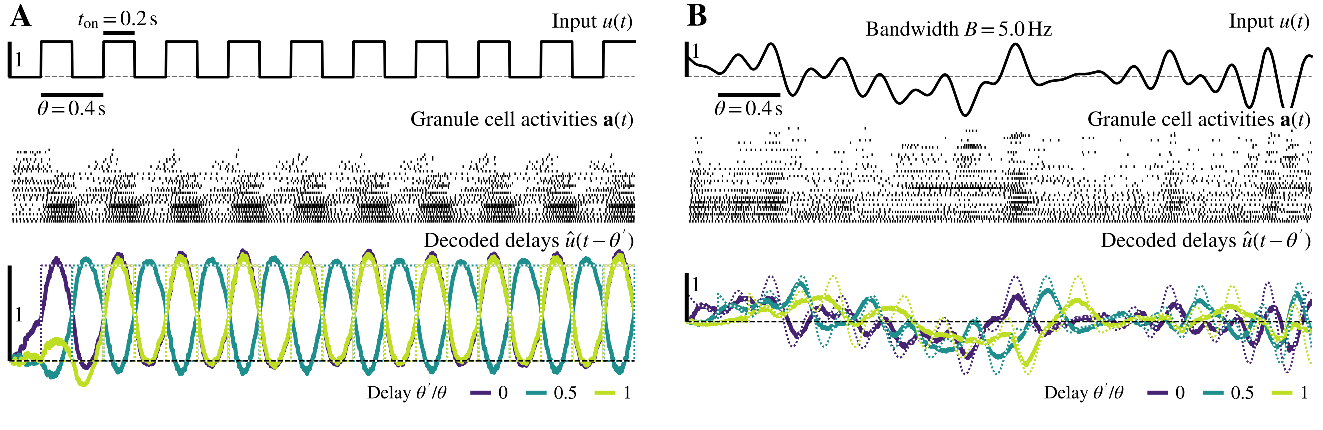

Figure 4: Examples showing the delayed input signals decoded form the granule layer in the detailed model (Fig. 2E). *Top row:* Input signal (rectangle pulses in **A**, white noise in **B**). *Middle row:* Spike raster for 40 randomly selected granule cells. *Bottom row:* Delays decoded from one thousand randomly selected granule cells. Dotted lines correspond to an optimal delay.

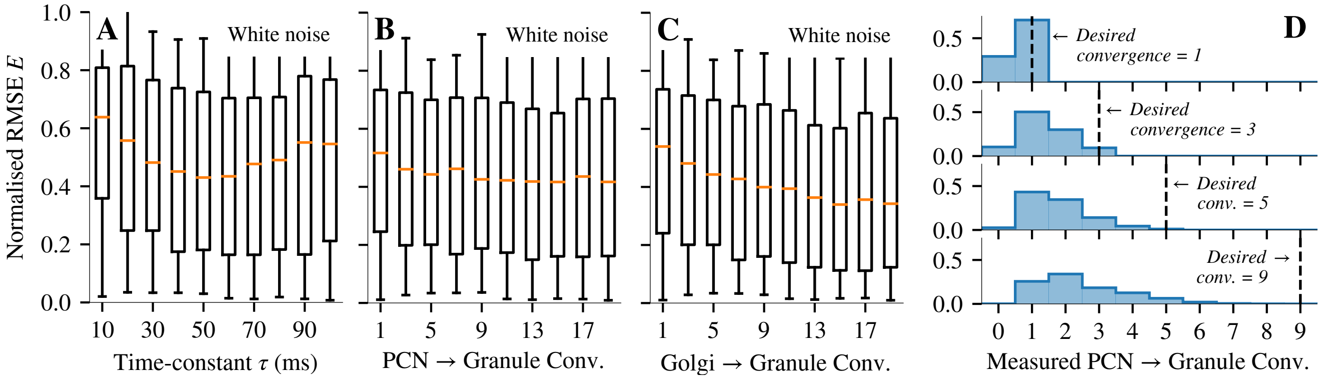

Figure 5: Parameter exploration. **(A-C)** Effects of varying parameters on the delay reconstruction error. The box plots show the median, lower and upper quartile of all the data depicted as a contour plot in Fig. 3; whiskers are the min/max. The total number of data-points in each bar plot is $n = 441$. **(D)** Histograms showing the frequency of measured PCN to granule convergences.

our modeling techniques are more general with respect to the high-level function that is being mapped onto the underlying circuit. Instead of relying on random connectivity, we directly specify the high-level function we would like the system to perform. We encourage cognitive modellers to view this particular model as an example; the techniques we present here are in principle compatible with all NEF models, including models of cognitive phenomena using the Semantic Pointer Architecture (SPA; Eliasmith, 2013).

We hope that this research facilitates grounding cognitive theories in biological mechanisms beyond what was already possible with the NEF and Nengo. Future work will focus on incorporating additional biological detail into the model (such as separate biological time constants for all synapses), as well as applying our techniques to more complex models.

References

Bekolay, T., Bergstra, J., Hunsberger, E., DeWolf, T., Stewart, T. C., Rasmussen, D., … Eliasmith, C. (2014). Nengo: A python tool for building large-scale functional brain models. *Frontiers in Neuroinformatics*, *7*(48).

Chadderton, P., Margrie, T. W., & Häusser, M. (2004). Integration of quanta in cerebellar granule cells during sensory processing. *Nature*, *428*(6985), 856–860.

Dean, P., Porrill, J., Ekerot, C.-F., & Jörntell, H. (2010). The cerebellar microcircuit as an adaptive filter: Experimental and computational evidence. *Nat. Rev. Neurosci.*, *11*(1).

Dieudonné, S. (1998). Submillisecond kinetics and low efficacy of parallel fibre-Golgi cell synaptic currents in the rat cerebellum. *The Journal of Physiology*, *510*(3).

D‘Angelo, E., Solinas, S., Mapelli, J., Gandolfi, D., Mapelli, L., & Prestori, F. (2013). The cerebellar Golgi cell and spatiotemporal organization of granular layer activity. *Frontiers in Neural Circuits*, *7*, 93.

Eliasmith, C. (2013). *How to build a brain: A neural architecture for biological cognition*. Oxford University Press.

Eliasmith, C., & Anderson, C. H. (2003). *Neural engineering: Computation, representation, and dynamics in neurobiological systems*. MIT Press.

Eliasmith, C., & Kolbeck, C. (2015). Marr's Attacks: On Reductionism and Vagueness. *TopiCS*, 1–13.

Hornik, K., Stinchcombe, M., & White, H. (1989). Multilayer feedforward networks are universal approximators. *Neural Networks*, *2*(5), 359–366.

Ito, M. (2010). Cerebellar Cortex. In G. Shepherd & S. Grillner (Eds.), *Handbook of Brain Microcircuits* (1st ed., pp. 293–300). Oxford University Press.

Lusk, N. A., Petter, E. A., MacDonald, C. J., & Meck, W. H. (2016). Cerebellar, hippocampal, and striatal time cells. *Current Opinion in Behavioral Sciences*, *8*.

Marr, D., & Poggio, T. (1976). From Understanding Computation to Understanding Neural Circuitry. *MIT AI Memo*(357).

McCormick, D. A., Lavond, D. G., Clark, G. A., Kettner, R. E., Rising, C. E., & Thompson, R. F. (1981). The engram found? Role of the cerebellum in classical conditioning of nictitating membrane and eyelid responses. *Bulletin of the Psychonomic Society*, *18*(3).

Parisien, C., Anderson, C. H., & Eliasmith, C. (2008). Solving the problem of negative synaptic weights in cortical models. *Neural Computation*, *20*, 1473–1494.

Rössert, C., Dean, P., & Porrill, J. (2015). At the edge of chaos: How cerebellar granular layer network dynamics can provide the basis for temporal filters. *PLOS Computational Biology*, *11*(10).

Stöckel, A., Stewart, T. C., & Eliasmith, C. (2020). A biologically plausible spiking neural model of eyeblink conditioning in the cerebellum. In *Proceedings of the 42th Annual Conference of the Cognitive Science Society.* Cognitive Science Society.

Stöckel, A., & Eliasmith, C. (2019). Passive nonlinear dendritic interactions as a general computational resource in functional spiking neural networks. *arXiv*.

Voelker, A. R., & Eliasmith, C. (2018). Improving Spiking Dynamical Networks: Accurate Delays, Higher-Order Synapses, and Time Cells. *Neural Computation*, *30*(3).

Voelker, A. R., Kajić, I., & Eliasmith, C. (2019). Legendre memory units: Continuous-time representation in recurrent neural networks. In *Advances in NeurIPS.*