What can astrocytes compute?

Erik J. Peterson*[1]

[1]Department of Psychology, Carnegie Mellon University

Abstract

A foundational result in neural computation was proving the firing-rate model of neurons defines a universal function approximator. In this paper I prove it is possible for astrocytes to compute anything neurons can, by extending this original proof to a model of calcium waves in astrocytes. I confirm this in a series of computer simulations. The major limit for astrocytes, it turns out, is not their ability to learn solutions but the computational complexity of their network. I suggest some initial experiments that might be used to confirm these predictions.

We've known for sometime astrocytes generate elaborate calcium waves, and that somehow these waves are important for learning [40, 16]. But two recent experiments offer an intriguing new possibility for computation in these cells. Mu et al (2019) showed direct signal integration by astrocytes leading to motor control in zebrafish [34]. Slezack et al (2019) showed direct integration of visual information, and behavioral state, in mice [41]. While it is was well known that astrocytes generate calcium waves [29, 19, 11], their role was thought to be limited to tuning neurons as part of the tripartite synapse [5, 45, 36]. These new results from [34] and [41], along with older speculations [37], however suggest a direct role for astrocytes in computation.

Inspired by these experimental results the aim of this paper is to use theoretical analysis to "rule in" a large range of computations as mathematically possible for astrocytes, reaching well beyond what we have evidence for experimentally.

I consider three questions.

1. Can the fundamental universal approximator proofs for neurons be extended to astrocytes?

2. If astrocytes can approximate universally, why do neurons exist?

3. Can simulations of astrocytes learn as well as a simulations of neurons?

Results

The results have three parts answering the three questions from the introduction. First is a theoretical study of function approximation in astrocytes. Second, I analyze the computational complexity of astrocyte networks. Third is some computer simulations.

Neural approximation. Let's first review biological neural networks reduced to firing-rate models of point cells. From there we can define a universal approximation.

The kind of two-layer neural network we are interested in is shown in Figure 1a and has the mathematical form,

$$y = \sum_{i=1}^{r} c_i^r \sigma \left(\sum_{j=1}^{n} w_{ij}^n x_j^n + b_i^n \right) \tag{1}$$

Where the input $\mathbf{x^n} = \{x_1^n, \ldots, x_n^n\}$ is a vector of size n, each $x_i \in \mathbf{R}$, and the output y is a single real value $y \in \mathbf{R}$. Here $w_{ij} \in \mathbf{R}$ is the weight between ith element of the input and the jth unit in layer n, $b_i^n \in \mathbf{R}$ is the threshold of the ith unit, and $c_i^r \in \mathbf{R}$ is the weight between the ith unit in layer r (which is also the output y). I use σ to denote some nonlinear function, restricted as described below.

Note the use of subscripts to denote indexing under summation and superscripts to denote

*erik.exists@gmail.com

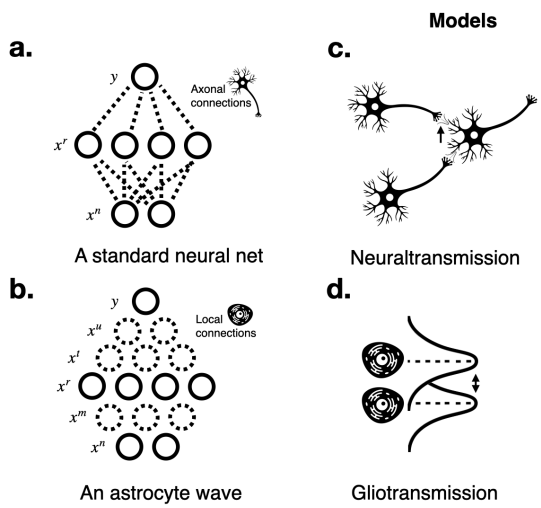

Figure 1: Models. **a**-**b** Diagram of communication in firing rate neural network (wij*a*), and an astrocyte wave with $k = 2$ neighbors (*b*). **c**-**d** Modes of transmission. Neuronal transmission (*a*) generally occurs in a well defined synapse, which among other things tightly controls the diffusion of the transmitter. Gliotransmission is far less confined, and can diffuse some distance leading to a potential for cross-talk between cells, a.k.a. leak.

the layer each term belongs to. In a two layer neural network these superscripts seem redundant. But as we move to study many-layered astrocyte wave networks they are a convenient way to prevent proliferation of excess terms.

It is known that two-layer neural networks [2] are universal function approximators. All approximator proofs follow the same form. They remake the question of how good an approximation is into a question about the "density" between two sets of numbers [38]. Density formalized as requiring every element in M is within a neighborhood $\epsilon > 0$ of an element in C. If this criterion holds, then M is said to be dense in C. Thiss kind of density is an abstract way to measure near equivalence in sets, and as a result approximation between functions.

Our target function I denote f, and set up as a continuous real function $f \in C(\mathbb{R}^N)$. It is this function we wish to approximate. To do that we'll be using an approximator F, given in this case by Equation 1.

To ask density questions about these we'll need sets, not function. To that end, it is common to focus on some compact subset K of $\mathbb{R}^n$. The set for f is then $\{f(x) : x \in K\}$. For the approximator F things are slightly complex, because we need a parameterized set. To get that I follow [38] and define a set building scheme, M_r (below), otherwise known as the spanning set.

$$M_r(\sigma) = \Big\{ \sum_{i=1}^{r} c_i^r \sigma \left(\sum_{j=1}^{n} w_{ij}^n x_j^n + b_i^r \right) \Big| \; w_{ij}^n, b_i^r, c_i^r, \in \mathbb{R} \Big\} \tag{2}$$

Armed with a way to build sets for F and f, on the "ball" K, we can then ask the following question. For which σ is it true that,

$$\max_{x \in K} |F(x) - f(x)| < \epsilon? \tag{3}$$

Answering this question positively means proving universal approximation. For neural networks of the kind above this has been studied for several kinds of σ, but also for may other networks beyond two-layers [12, 26, 2, 38, 44, 31, 33, 44, 23, 32, 28]. In this work, I borrow a classic theorem for the two-layer network as

given by [38]. It is modified to include a continuous nonzero derivative on σ.

Theorem 0.1 (Neural approximation) *Let σ by any nonlinear function that is not a polynomial function and has a continuous nonzero derivative at some point. Then finite sums of the form in Equation 1 are dense in $C(\mathbb{R}^n)$ on K. In other words, given any $f(x) \in C(\mathbb{R}^n)$ and $\epsilon > 0$ there is function $F(x)$ for which, $|F(x) - f(x)| < \epsilon$, for all $x \in K$.*

Proving approximation only means F can "do the job" approximating f. It does not ensure an efficient method to cause F to approximate f exists or is known. Another way to explain this limit is to say that density proofs are not constructive proofs. I take up construction later on.

Astrocyte equivalence. The model of astrocytes I'll consider is shown in Figure 1b. Compare it to a standard two-layer neural network, in Figure 1a. I assume,

- Astrocyte communication is only between k nearest neighbors in the "forward" direction. (Forward is defined as the flow from input to output).
- Astrocytes can be modeled as point cells on an ordered rectangular grid.
- That gliotransmission is the dominant mode of communication.
- (If instead gap junctions are dominant this would improve the specificity of communications and so improve practical approximation performance. See the Astrocyte complexity section below.)
- The lack of synapses means transmitters are released from the cell wall uniformly, and "leaks" across to neighborhoods. (We show this leak in Figure 1d as a Gaussian; Contrast this leak to the "exact" synapses in a neural network as shown in Figure 1c.)
- This leak does not extend past k neighbors.

The limited connectivity between astrocytes in the model means writing down the general equation for astrocytes is more complex than for neurons. I show an example of a $r = 4, k = 2$ astrocytes network in Figure 1b and mathematically below. This example matches the max width $r = 4$ of the two layer neural network depicted in Figure 1a.

$$y = \sum_{i=1}^{u} c_i^u \sigma\Big(\sum_{j=1}^{t} w_j^t \rho_{\mathbf{x}}(x_j^t) + b_i^u \Big) \tag{4}$$

$$x^t = \begin{bmatrix} \sum_{i=1}^{r-2} c_i^r \sigma\Big(\sum_{j=1}^{m-1} w_{ij}^m \rho_{\mathbf{x}}(x_j^m) + b_i^r \Big) \\ \sum_{i=2}^{r-1} c_i^r \sigma\Big(\sum_{j=1}^{m} w_{ij}^m \rho_{\mathbf{x}}(x_j^m) + b_i^r \Big) \\ \sum_{i=3}^{r} c_i^r \sigma\Big(\sum_{j=2}^{m} w_{ij}^m \rho_{\mathbf{x}}(x_j^m) + b_i^r \Big) \end{bmatrix} \tag{5}$$

$$x^m = \begin{bmatrix} \sum_{j=1}^{n-1} w_j^n \rho_{\mathbf{x}}(x_j^n) + b_i^m \\ \sum_{j=1}^{n} w_j^n \rho_{\mathbf{x}}(x_j^n) + b_i^m \\ \sum_{j=2}^{n} w_j^n \rho_{\mathbf{x}}(x_j^n) + b_i^m \end{bmatrix} \tag{6}$$

$$x^n = \mathbf{x} \tag{7}$$

I assume an astrocyte network should take the general form shown in Equation 4, generalized for any choice of finite n and $r > n + 1$ [28], where r denotes the maximum width of the widest layer. The function $\rho_{\mathbf{x}}$ stands in for a generic convolution operation take on each element in some x based in part on the vector $\mathbf{x}$, where refers to some layer input. I other words, this is how I model transmitter diffusion, or "leak". Having introduced the physical idea of leak for now though I will neglect it, by setting $\rho_{\mathbf{x}}$ to 1, and treating astrocytes as if they formed direct connections. Justifications for, and tests of, these assumptions are found in the simulations below. With these we can now prove any astrocyte network G can be made equivalent to any two-layer neural network F, which is itself a universal approximator. Without loss of generality I write the spanning set for astrocytes $P(\sigma)$, as a series of sum and compositions based on Equations 4, generalizable as necessary to other n and r.

$$\begin{aligned} P_r(\sigma) = \Big\{ & y(x^t(x^m(x))) \\ & \Big| \; w_j^t, b_i^u, c_i^u, w_{ij}^m, b_i^r, c_i^r, w_j^n, b_i^m, w_j^n \in \mathbb{R} \Big\} \end{aligned} \tag{8}$$

Theorem 0.2 (Astrocyte equivalence) *Let σ by any nonlinear function that is not a polynomial function and has a continuous nonzero derivative at some point. Then generalized finite sums of the form in Equation 4 can be made equivalent with the form in Equation 1 there if $F(x)$ is dense then $G(x)$ is also dense in $C(\mathbb{R}^N)$ on K. In other words, given any $g \in P_r(\sigma)$ there is a neural network $F \in M(\sigma)$ for which $|G(x) - F(x)| = 0$, for all $x \in K$. And*

therefore given any $f(x) \in C(\mathbb{R}^N)$ and $\epsilon > 0$ there is function $G(x)$ for which, $|G(x) - f(x)| < \epsilon$ for all $x \in K$.

The proof for this is trivial and proceeds in two parts. We first recognize the inner sum in Equation 1, that is $\sum_{j=1}^{n} w_{ij}x_j + b_i$, is an simply an affine function, which can perfectly reconstructed by linearizing the corresponding region in G. That is, layers n through r are linearized allowing for an exact superposition solution.

Each layer and set of connection in the "outer" sum, $\sum_{i=1}^{r} c_i\sigma_s(.)$ is composed of small networks mathematically equivalent to neural networks with $n = 3$, and is therefore composed of compositions of nonlinear terms that are already proven to be universal approximators. In other words, the outer sum is necessarily stepwise dense in C by reusing the argument made for neurons.

This proof is not intended to be a guide to constructing astrocyte networks. Nor is the linearizing step necessary for construction of working waves, as I show in the next section. This theoretical approach is instead a means to an end to establish density.

Astrocyte complexity Before giving mathematical details the big picture for the complexity of astrocyte waves is easy to state. To span distances astrocytes must pass messages between cells. Each cell is in a sense a new set of parameters, and so complexity of the whole model increases, compared to neurons.

If we assume neurons axons can span any distance $|m - n|$, then for a neuronal network we can move from m to n in $l = 1$ layers, and a total of $m + n$ cells.

The lack of axons on astrocytes means however there is a necessary link between the size or width of a feed forward network and it's depth. If each cell rests on the corners of a regular grid, going from from input size m to output size n, where $m \neq n$, it will take $l \geq m + \frac{|m-n|}{2} - 2$ layer steps, if both m and n are even or odd. It will take $l \geq m + \frac{|m-n+1|}{2} - 2$ steps if there is m is even and n odd, or vice versa. If the widths are the same, so $m = n$, the layer "penalty" is $l = m - 2$. If l is the difference in layer numbers between astrocytes and neurons is l, then the cell number penalty is the sum over l, with the generic term o_i standing for the width of each layer. That is, $\sum_{i=1}^{l} o_i$.

In computer science terms an $\mathbb{O}(l)$, or linear penalty, for a computation is still viable, even efficient, computation. I argue this viability is not necessarily so in biology. For example, imagine neurons had not been "invented" and circuits were limited to wave/step computations and we continue to use grid models to make the calculations convenient. If the wave transmission in a single astrocyte takes t_a seconds, the the total transmission time is $T_a = t_a * l$ for astrocytes. It would be t_a for neurons. If we modestly assume a very small neural circuit to compare to, say $m = 2$ and $n = 12$ so $l = 4$, then T_a has grown four fold. Consider what this means for motor output, as an example. If the best motor reaction time of an animal with neurons is 10 ms, it would grow to 40 ms with astrocytes. This is a substantial difference in reaction time for say a prey escaping an agile predator. Now consider what a linear time penalty would imply for mammalian brains with their trillions of connections [1, 35].

In sum, the analysis in this paper implies the neuron's innovation—with the introduction of axons and synapses—-was not the ability to learn. It was the ability to compute as efficiently, in terms of width, depth, and cell number. But this is not the whole story, perhaps. Our view has been looking at what it would take to make astrocytes mimic neurons, exactly. In some ways this provides insight into the base question of "why neurons"?. But, there is no reason for evolution to hold this perspective, exactly.

Astrocytes can be expected to perform more efficiently than neurons when there is some "sympathy" between them and the learning problem at hand [50]. For example, in [34] et al astrocytes act to integrate incoming signals. If this integration was done instead using neurons, it would require several cells, or circuits, with exacting properties [48, 8, 15]. Meanwhile slow calcium dynamics, combined with connectionist waves, generate natural integration [42, 14, 34, 41]. That is despite the analysis above, we have meaningful evidence that astrocytes can be more efficient in terms of cell

number, for some computations which match their properties like integration. The question I pose is, given astrocytes can act as universal approximators, what other functions might they fulfil efficiently which have been so far missed in our experiments?

Simulations. Astrocytes learn (nearly) as well as neurons in three test simulations. Both astrocyte and neural models were trained by stochastic gradient descent, using identical procedures and parameters. The learning tasks were one classic nonlinear learning test (the XOR problem) and two classic visual recognition datasets (MINST digits and fashion). These tasks are shown in Figure 2.

To design astrocytes waves I intermingled the three basic courses a feed-forward wave can take–spreading themselves out, collapsing or gathering themselves in, or simply sliding forward with no change in width. The final model architectures are shown in *Table 1*, and *2* (Methods).

The change in size from input image to output class was too large, given astrocytes complexity. And the various tricks for fixing vanishing gradients available to the machine learning practitioner do not seem biologically sound. To overcome the complexity limits of astrocytes, and the vanishing gradients which follow from it, all vision tasks had two stages. First the high dimensional images were projected to a low dimensional space. This was done with neural negotiation. One approach mimicked the fixed random sparse connections sometimes reported in cortex [6, 39]. The other approach trained a variational autoencoder [30] during astrocyte training, in an online way. This was termed co-learning, and it mimicked a learning process where astrocytes adjust online to ongoing changes/learning in their (presumptively) neuronal input.

On the XOR task astrocytes and neurons showed perfect accuracy (Figure 3**a**). In both MINST tasks astrocyte performance was reduced between 0.05 and 0.08 (3**b-e**). Variance was also substantially higher in astrocytes and they were substantially slower to train (see inset panels in 3). The difference in speed of learning was as predicted. See *Complexity*

Connections in this working model were lim-

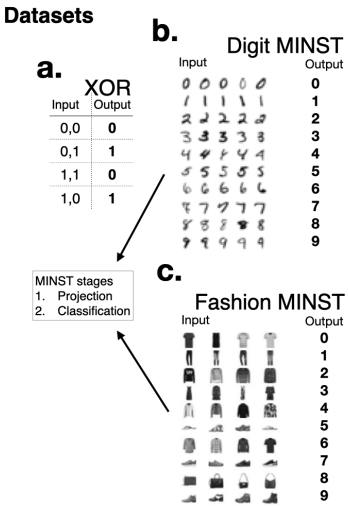

Figure 2: **a**-**b**. Datasets. We considered three decision tasks. **a.** XOR task. **b.** MINST digit recognition **c.** MINST fashion recognition. Learning in MINST sets had two distinct stages (grey box). First the raw pixel input data had its dimensionality reduced to 1d vector ($z = 20$). This representation was used as the input to the learned classifier networks, both neurons and astrocytes.

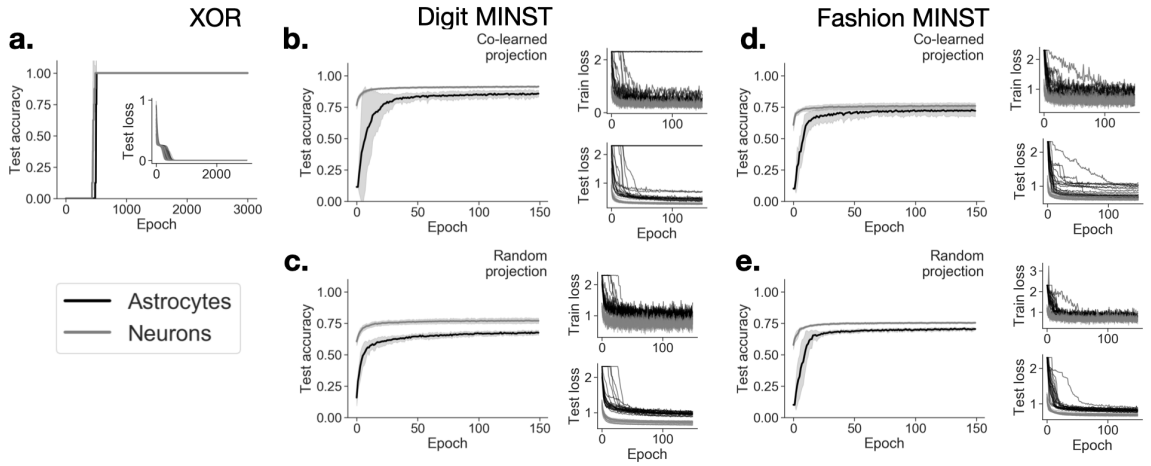

Figure 3: Learning performance. **a**. XOR task classification accuracy as a function of training epoch, on the test set (which is the training set for this task). Inset plot is individual loss curves for all 20 experiments. **b-c**. MINST digit classification accuracy for co-learning and random projection schemes. To the right of each are the individual train/text loss curves. **d-e**. MINST fashion classification accuracy for co-learning and random projection schemes. To the right of each are the individual train/text loss curves.

ited to each cell's $k = 3$ nearest downstream neighbors. However in this model I was free to explore transmitter leak between greater than k neighbors, modeled by a 1d Gaussian convolution of each layers output. These simulations also let me explore biological factors like additive noise, and communication failures (dropout) between neighbors (*Methods*). See, Figure 4.

Performance is robust to < 0.3 standard deviations of leak, followed by sharp decline. In contrast, injected (additive) transmission noise, and signal loss, had much smoother declines in classification. The biological significance of these patterns is unclear. Overall these biological perturbations showed what seems to be reasonable robustness leak, noise, and signal loss. There is however a significant caveat. It is difficult to know how plausible the magnitude of these perturbations were compared to what real astrocytes, biological systems, endure. Detailed data does at present seem to be available.

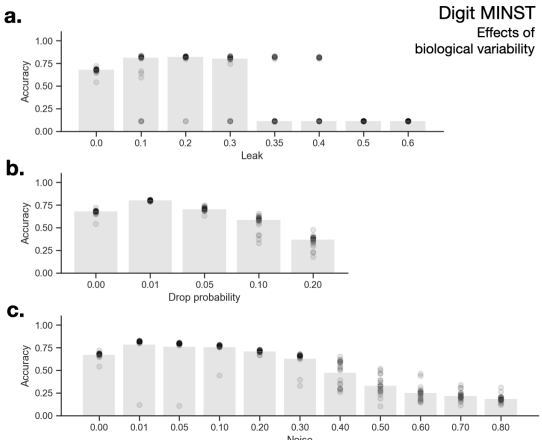

Figure 4: Biological realism. Leak (diffusion), communication failures (message dropping), and noise (message corruption) and their effects on astrocyte learning performance. **a.** Final test accuracy after convolution with a spatial Gaussian kernel (0-0.6 standard deviations). **b.** Final test accuracy with probabilistic connection loss between neighbors (0.0-0.20). **c.** Final test accuracy with connection noise (0.0-0.8 standard deviations). Points represent 20 individual experiments. Grey bars represent median accuracy.

Discussion

Limtations. I have studied a highly simplified model of feed-forward computations, with near-

est neighbor connections between cells. I presume this is a minimal but adequate account of calcium waves. That said, this simplified model neglects:

- Lateral and recurrent calcium dynamics [20, 40, 49].
- Gap junctions and their associated plasticity [16].
- Neuronal-astrocyte interactions during wave-computation [21, 40].
- Calcium microdomains in individual cells.
- The complex, sometimes gated, calcium-gliotransmitter relationship [4].
- Astrocytes are a diverse cell type, with diverse shapes roles and electrical properties [27]

I think of these details as adding more degrees of freedom, and so dimensions, to astrocytes computational potential [46, 9, 47]. It is often the case in other modeling studies increasing detail has at worse has no effect, and often leads to improvement [24, 25, 18].

Predictions. To detect if astrocytes carry on their own computations the first challenge is isolating these signals. [41] has shown this is possible by direct observation. A reasonable default is to say astrocyte activity is a delayed slow filtered reflection of neuronal drive. Under this null any pattern in astrocyte waves is an epiphenomenon of neural activity. If it was then shown that wave patterns are not just reflections, this would be some evidence for astrocyte's wave computations.

For example, in multi-area, brain-wide recordings the high dimensional dynamics of neurons is well captured by low dimensional dynamical systems [10, 51]. [43] developed convergent cross mapping, a way to compare if two dynamical systems are equivalent using only noisy measurements. This was adapted to work in latent spaces by [7] If astrocyte waves are simple an epiphenomenon this procedure should identify an equivalence between a learned low dimensional model neuronal firing rates, and measurements of astrocytes of calcium dynamics. Taken over several cortical areas, experiments in cross mapping would provide good initial evidence for astrocyte computation and independence.

Another open problem is glia biology is understanding why cortical astrocytes are so spread out. The results for leak and complexity suggests an answer. Astrocytes are spread out to minimize leak, or crosstalk, between themselves. In other words, are they spread out to play a role as waveguides [4, 22]?

Methods

All learning networks were implemented in python using the pytorch framework. Code and data are available at `https://github.com/CoAxLab/glia_playing_atari`. Simulations were run on a 4 Nvidia GeForce GTX 1080 Ti cluster.

We trained all models on the XOR problem, and on two computer vision tasks MINST digits and MINST fashion. MINST datasets were sourced from torchvision. Images were randomly assigned to training and reporting sets, and were used as provided. XOR data was its truth table implemented in tensors. Note the XOR problem has no meaningful way to split between training and testing

Neural networks were multilayered perceptrons, with all to all connections. These networks played two roles. The first used a variational autoencoder [30] to reduce input dimensionality into something astrocyte waves can practically learn from. More on this below. The second kind implemented the classification networks, which were the point of comparison for the astrocyte waves. The autoencoder design is discussed in the next section. The hyperparameters for the three classifier networks are shown in Table 1.

The input image size for both MINST dataset is N by N pixels, which gives flattened vectors of size K. The output size for both is 10, the number of classes to be learned. As astrocytes in simulation needed l steps to make a dimensionality change of size K. This led to vanishing gradients [17]. To solve this two kinds of dimensionality reduction were considered. One inspired by standard machine learning practice. The use of a variational autoencoder. The other was inspired by the neural circuits, which are often modelled as sparse random projections [13, 3, 6].

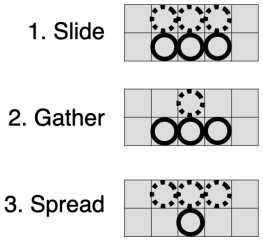

Figure 5: Astrocyte tensor operations – three prototypes for astrocyte steps. Solid circles represent the current cell (the input for that step). Dashed circles represent the projected output (for that step). For compactness in the Tables which describe the networks, I use s to denote spread steps, g to denote gather, and d for slide.

To implement astrocyte communication we defined three prototype "steps" for wave propagation. Our use of the term steps is synonymous with "layers" in typical neural networks. We distinguish them because our astrocyte waves may need several steps to accomplish the same output as the equivalent single layer of neurons. If m is the number of cells in the input and m the number of cells in the output, we define three step types where $m > n$, $m = n$, and where $n < m$. s 1. *Slide* (d) steps have input size n and output size $m = n$. 2. *Gather* (g) steps have input size n and output size $\max(n-2, 1)$. 3. *Spread* (s) steps have input size n and output size $n+2$. For illustrated examples, see Figure 5.

Each step ensures only local, (i, j, k) nearest neighbor interactions. These were implemented as pytorch tensors of full rank n and m but whose sum operations for each jth cell were locally indexed during both forward computation and backpropagation. This meant working around pytorch's limits on "in place" operations. Each wave step could contain stochastic dropout, or (Gaussian) noise injection or (Gaussian) spatial convolution.

References

[1] Frederico A.C. Azevedo, Ludmila R.B. Carvalho, Lea T. Grinberg, José Marcelo Farfel, Renata E.L. Ferretti, Renata E.P. Leite, Wilson Jacob Filho, Roberto Lent, and Suzana Herculano-Houzel. Equal numbers of neuronal and nonneuronal cells make the human brain an isometrically scaled-up primate brain. *J. Comp. Neurol.*, 513(5):532–541, April 2009.

[2] A.R. Barron. Universal approximation bounds for superpositions of a sigmoidal function. *IEEE Trans. Inform. Theory*, 39(3):930–945, May 1993.

[3] Alison L. Barth and James F.A. Poulet. Experimental evidence for sparse firing in the neocortex. *Trends in Neurosciences*, 35(6):345–355, June 2012.

[4] Narges Bazargani and David Attwell. Astrocyte calcium signaling: The third wave. *Nat Neurosci*, 19(2):182–189, February 2016.

[5] Alba Bellot-Saez, Orsolya Kékesi, John W. Morley, and Yossi Buskila. Astrocytic modulation of neuronal excitability through K + spatial buffering. *Neuroscience & Biobehavioral Reviews*, 77:87–97, June 2017.

[6] Michael Beyeler, Emily Rounds, Kristofor Carlson, Nikil Dutt, and Jeffrey L. Krichmar. Sparse coding and dimensionality reduction in cortex. *bioRxiv*, June 2017.

[7] Edward De Brouwer. Latent Convergent Cross Mapping. *34th Conference on Neural Information Processing Systems*, pages 1–22, 2020.

[8] Javier A. Caballero, Mark D. Humphries, and Kevin N. Gurney. A probabilistic, distributed, recursive mechanism for decision-making in the brain. *PLOS Computational Biology*, 14(4):e1006033, April 2018.

[9] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling. *arXiv:1412.3555 [cs]*, December 2014.

[10] Mark M Churchland, Byron M Yu, Maneesh Sahani, and Krishna V Shenoy. Techniques for extracting single-trial activity patterns from large-scale neural recordings. *Current Opinion in Neurobiology*, 17(5):609–618, October 2007.

[11] A. Cornell-Bell, S. Finkbeiner, M. Cooper, and S. Smith. Glutamate induces calcium waves in cultured astrocytes: Long-range glial signaling. *Science*, 247(4941):470–473, January 1990.

[12] G Cybenkot. Approximation by superpositions of a sigmoidal function. *Math. Control Signals System*, 2:12, 1989.

[13] Peter Dayan and L F Abbott. *Theoretical Neuroscience*. MIT press, 2005.

[14] Tara Deemyad, Joel Lüthi, and Nelson Spruston. Astrocytes integrate and drive action potential firing in inhibitory subnetworks. *Nat Commun*, 9(1):4336, December 2018.

[15] Kyle Dunovan and Timothy Verstynen. Believer-Skeptic Meets Actor-Critic: Rethinking the Role of Basal Ganglia Pathways during Decision-Making and Reinforcement Learning. *Front. Neurosci.*, 10, March 2016.

[16] Steven Finkbeiner. Calcium waves in astrocytes-filling in the gaps. *Neuron*, 8(6):1101–1108, June 1992.

[17] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016.

[18] Jordan Guerguiev, Timothy P Lillicrap, and Blake A Richards. Towards deep learning with segregated dendrites. *eLife*, 6, December 2017.

[19] Peter B. Guthrie, Joshua Knappenberger, Menahem Segal, Michael V. L. Bennett, Andrew C. Charles, and S. B. Kater. ATP Released from Astrocytes Mediates Glial Calcium Waves. *J. Neurosci.*, 19(2):520–528, January 1999.

[20] Hajime Hirase, Lifen Qian, Peter Barthó, and György Buzsáki. Calcium Dynamics of Cortical Astrocytic Networks In Vivo. *PLoS Biol*, 2(4):e96, April 2004.

[21] Thomas Höfer, Laurent Venance, and Christian Giaume. Control and Plasticity of Intercellular Calcium Waves in Astrocytes: A Modeling Approach. *J. Neurosci.*, 22(12):4850–4859, June 2002.

[22] Tyler W. Hughes, Ian A. D. Williamson, Momchil Minkov, and Shanhui Fan. Wave Physics as an Analog Recurrent Neural Network. *arXiv:1904.12831 [physics]*, April 2019.

[23] Jesse Johnson. Deep, Skinny Neural Networks are not Universal Approximators. *arXiv:1810.00393 [cs, stat]*, September 2018.

[24] Ilenna Simone Jones and Konrad Paul Kording. Can Single Neurons Solve MNIST? The Computational Power of Biological Dendritic Trees. *arXiv:2009.01269 [q-bio]*, September 2020.

[25] Ilenna Simone Jones and Konrad Paul Kording. Do biological constraints impair dendritic computation? *arXiv:2103.03274 [q-bio]*, March 2021.

[26] L.K. Jones. Constructive approximations for neural networks by sigmoidal functions. *Proc. IEEE*, 78(10):1586–1589, Oct./1990.

[27] Baljit S. Khakh and Benjamin Deneen. The Emerging Nature of Astrocyte Diversity. *Annu. Rev. Neurosci.*, 42(1):187–207, July 2019.

[28] Patrick Kidger and Terry Lyons. Universal Approximation with Deep Narrow Networks. *arXiv:1905.08539 [cs, math, stat]*, June 2020.

[29] Warren T. Kim, Marc G. Rioult, and Ann H. Cornell-Bell. Glutamate-induced calcium signaling in astrocytes. *Glia*, 11(2):173–184, June 1994.

[30] Diederik P. Kingma and Max Welling. Auto-Encoding Variational Bayes. *arXiv:1312.6114 [cs, stat]*, December 2013.

[31] Hongzhou Lin and Stefanie Jegelka. ResNet with one-neuron hidden layers is a Universal Approximator. *32nd Conference on Neural Information Processing Systems*, pages 1–10, 2018.

[32] Ehsan Lotfi and M.-R. Akbarzadeh-T. A Novel Single Neuron Perceptron with Universal Approximation and XOR Computation Properties. *Computational Intelligence and Neuroscience*, 2014:1–6, 2014.

[33] Zhou Lu, Hongming Pu, Feicheng Wang, Zhiqiang Hu, and Liwei Wang. The Expressive Power of Neural Networks: A View from the Width. *NIPS*, page 9, 2017.

[34] Yu Mu, Davis V. Bennett, Mikail Rubinov, Sujatha Narayan, Chao-Tsung Yang, Masashi Tanimoto, Brett D. Mensh, Loren L. Looger, and Misha B. Ahrens. Glia Accumulate Evidence that Actions Are Futile and Suppress Unsuccessful Behavior. *Cell*, 178(1):27–43.e19, June 2019.

[35] Seweryn Olkowicz, Martin Kocourek, Radek K. Lučan, Michal Porteš, W. Tecumseh Fitch, Suzana Herculano-Houzel, and Pavel Němec. Birds have primate-like numbers of neurons in the forebrain. *Proc Natl Acad Sci USA*, 113(26):7255–7260, June 2016.

[36] Lucia Pasti, Andrea Volterra, Tullio Pozzan, and Giorgio Carmignoto. Intracellular Calcium Oscillations in Astrocytes: A Highly Plastic, Bidirectional Form of Communication between Neurons and Astrocytes *In Situ*. *J. Neurosci.*, 17(20):7817–7830, October 1997.

[37] Alfredo Pereira and Fábio Augusto Furlan. Astrocytes and human cognition: Modeling information integration and modulation of neuronal activity. *Progress in Neurobiology*, 92(3):405–420, November 2010.

[38] Allan Pinkus. Approximation theory of the MLP model in neural networks. *Acta Numerica*, 8:143–195, January 1999.

[39] Cindy Poo and Jeffry S. Isaacson. Odor Representations in Olfactory Cortex: "Sparse" Coding, Global Inhibition, and Oscillations. *Neuron*, 62(6):850–861, June 2009.

[40] Eliana Scemes and Christian Giaume. Astrocyte calcium waves: What they are and what they do. *Glia*, 54(7):716–725, November 2006.

[41] Michal Slezak, Steffen Kandler, Paul P. Van Veldhoven, Chris Van den Haute, Vincent Bonin, and Matthew G. Holt. Distinct Mechanisms for Visual and Motor-Related Astrocyte Responses in Mouse Visual Cortex. *Current Biology*, page S0960982219309583, September 2019.

[42] Hiraku Suda, Hiroaki Mano, Masatsugu Toyota, Kenji Fukushima, Tetsuro Mimura, Izuo Tsutsui, Rainer Hedrich, Yosuke Tamada, and Mitsuyasu Hasebe. Calcium dynamics during trap closure visualized in transgenic Venus flytrap. *Nat. Plants*, 6(10):1219–1224, October 2020.

[43] G. Sugihara, R. May, H. Ye, C.-h. Hsieh, E. Deyle, M. Fogarty, and S. Munch. Detecting Causality in Complex Ecosystems. *Science*, 338(6106):496–500, October 2012.

[44] Ilya Sutskever and Geoffrey E. Hinton. Deep, Narrow Sigmoid Belief Networks Are Universal Approximators. *Neural Computation*, 20(11):2629–2636, November 2008.

[45] Akinobu Suzuki, Sarah A. Stern, Ozlem Bozdagi, George W. Huntley, Ruth H. Walker, Pierre J. Magistretti, and Cristina M. Alberini. Astrocyte-Neuron Lactate Transport Is Required for Long-Term Memory Formation. *Cell*, 144(5):810–823, March 2011.

[46] V Vapnik and A Chervonenkis. On the Uniform Convergence of Relative Frequencies of Events to Their Probabilities. *Theory of Probability & Its Applications*, 16(2):17, 1971.

[47] Joel Veness, Tor Lattimore, Avishkar Bhoopchand, Agnieszka Grabska-Barwinska, Christopher Mattern, and Peter Toth. Online Learning with Gated Linear Networks. page 40.

[48] Xiao-Jing Wang. Probabilistic Decision Making by Slow Reverberation in Cortical Circuits. *Neuron*, 36(5):955–968, December 2002.

[49] Tamily A. Weissman, Patricio A. Riquelme, Lidija Ivic, Alexander C. Flint, and Arnold R. Kriegstein. Calcium Waves Propagate through Radial Glial Cells and Modulate Proliferation in the Developing Neocortex. *Neuron*, 43(5):647–661, September 2004.

[50] D.H. Wolpert and W.G. Macready. No free lunch theorems for optimization. *IEEE Trans. Evol. Computat.*, 1(1):67–82, April 1997.

[51] Byron M. Yu, John P. Cunningham, Gopal Santhanam, Stephen I. Ryu, Krishna V. Shenoy, and Maneesh Sahani. Gaussian-Process Factor Analysis for Low-Dimensional Single-Trial Analysis of Neural Population Activity. *Journal of Neurophysiology*, 102(1):614–635, July 2009.

Table 1: Astrocyte model and training hyper-parameters. Note: *s* denotes spread steps, *g* denotes gather, and *d* denotes slide. The learning rate for the XOR network was 0.001. The two MINST models have a learning rate of 0.004, and a 128 image batch size.

Network	Layers	Input size	Output size	Layer steps
XOR	13	2	1	(s,d,s,d,s,d,g,s,g,s,g,g,g)
Digits	10	20	10	(g,s,g,s,g,s,g,s,g,s,g)
Fashion	10	20	10	(g,s,g,s,g,s,g,s,g,s,g)

Table 2: Network models and training hyper-parameters. The learning rate for the XOR network was 0.001. The two MINST models have a learning rate of 0.004, and a 128 image batch size.

Network	Layers	Input size	Output size	Hidden sizes
XOR	1	2	1	1
Digits	1	20	10	20
Fashion	1	20	10	20