Nonlinear Dendritic Coincidence Detection for Supervised Learning

Fabian Schubert

April 5, 2021

1 Introduction

In recent years, a growing body of research has addressed the functional implications of the distinct physiology and anatomy of cortical pyramidal neurons. In particular, on the theoretical side, we saw a paradigm shift from treating neurons as point-like electrical structures towards embracing the entire dendritic structure. This was mostly due to the fact that experimental work uncovered dynamical properties of these cells that simply could not be accounted for by point models.

An important finding was that the apical dendritic tree of cortical pyramidal neurons can act as a separate nonlinear synaptic integration zone. Under certain conditions, a dendritic Ca^{2+} spike can be elicited that propagates towards the soma, causing rapid, bursting spiking activity. One of the cases in which dendritic spiking can occur was termed 'backpropagation-activated Ca^{2+} spike firing' ('BAC firing'): A single somatic spike can backpropagate towards the apical spike initiation zone, in turn significantly facilitating the initiation of a dendritic spike. This reciprocal coupling is believed to act as a form of coincidence detection: If apical and basal synaptic input co-occurs, the neuron can respond with a rapid burst of spiking activity. The firing rate of these temporal bursts exceeds the firing rate that is maximally achievable under basal synaptic input alone, therefore representing a form of temporal coincidence detection between apical and basal input.

Naturally, these mechanisms also affect plasticity and thus learning within the cortex. While the interplay between basal and apical stimulation and its effect on synaptic efficacies is subject to ongoing research, there is some evidence that BAC-firing tends to shift plasticity towards long-term potentiation (LTP). Thus, coincidence between basal and apical input appears to also gate synaptic plasticity.

In a supervised learning scheme, where the top down input arriving at the apical compartment acts as the teaching signal, the most straight-forward learning rule for the basal synaptic weights would be derived from an appropriate loss function, such as a mean square error, based on the difference between basal and apical input, i.e. $I_\mathrm{p} - I_\mathrm{d}$. Theoretical work has investigated possible learning mechanisms that could utilize an intracellular error signal (Urbanczik and Senn, 2014; Schiess et al., 2016; Guerguiev et al., 2017). However, a clear experimental evidence for a physical quantity encoding such an error is—to our knowledge—yet to be found. On the other hand, Hebbian-type plasticity is extensively documented in experiments. Therefore, our work was based on the question whether the non-linear interactions between basal and apical synaptic input could, when combined with a Hebbian plasticity rule, allow a neuron to learn to reproduce an apical teaching signal in its proximal input.

In our work, we combined a phenomenological model that generates the output firing rate as a function of two streams of synaptic input (subsuming basal and apical input) with Hebbian as well as BCM-like plasticity rules on basal synapses. We hypothesized that this combination of neural activation and plasticity rules would lead to an increased correlation between basal and apical input.

Furthermore, this temporal alignment could potentially facilitate apical inputs to act as top-down teaching signals, without the need for an explicit error-driven learning rule. Thus, we also tested our model in a simple linear supervised classification task and compared it to the performance of a simple point neuron equipped with similar plasticity rules.

2 Model

2.1 Neuron Model

The neuron model used throughout this study is a discrete-time rate encoding model that uses two separate input variables, subsuming the total synaptic input current injected arriving at the basal (proximal) and apical (distal) dendritic structure of a pyramidal neuron, respectively. The model is a slightly simplified version of a phenomenological model proposed by Shai et al. (2015). Denoting the input currents by I_p (proximal) and I_d (distal), the model is written as

$$\begin{aligned} y\left(t\right) = {} & \sigma\left(I_\mathrm{p}(t) - \theta_{p0}\right)\left[1 - \sigma\left(I_\mathrm{d}(t) - \theta_d\right)\right] \\ & + \alpha\sigma\left(I_\mathrm{d}(t) - \theta_d\right)\sigma\left(I_\mathrm{p}(t) - \theta_{p1}\right) \end{aligned} \tag{1}$$

$$\sigma(x) \equiv \frac{1}{1+\exp(-4x)}\,. \tag{2}$$

Here, θ_{p0}, θ_{p1} and θ_d are threshold variables with respect to proximal and distal input. Overall, this equation describes two distinct regions of neural activation in the $(I_\mathrm{p}, I_\mathrm{d})$-space which differ in their maximal firing rates, which are set to 1 and α, where $0 < \alpha < 1$. A plot of (1) is shown in Fig. 1.

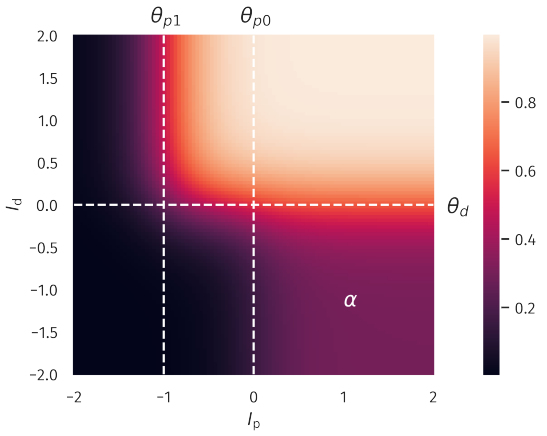

Figure 1: **Two-compartment rate model.** Firing rate as a function of proximal and distal input I_p and I_d, see (1). The thresholds θ_{p0}, θ_{p1} and θ_d define two regions of neural activity with a a maximal firing rate of 1 and α.

In all our numerical experiments, we compared this model with a simple point neuron model that was given by

$$y(t) = \sigma\left(I_\mathrm{p}(t) + I_\mathrm{d}(t) - \theta\right) \ . \tag{3}$$

In all our numerical experiments, the apical input I_d was generated "as is", meaning, it was not dynamically calculated as a superposition of multiple presynaptic inputs, but given by

$$I_\mathrm{d}(t) = n_d(t)x_d(t) - b_d(t) \ , \tag{4}$$

where $n_d(t)$ is a scaling factor, $x_d(t)$ a pre-generated discrete time sequence and $b_d(t)$ a bias. Note that n_d and b_d are time dependent since they were subject to adaptation processes that are described in the next section.

Similarly, $I_\mathrm{p}(t)$ was given by

$$I_\mathrm{p}(t) = n_p(t)\sum_{i=1}^{N} x_{p,i}(t)w_i(t) - b_p(t) \ , \tag{5}$$

where N is the number of presynaptic afferents, $x_{p,i}(t)$ the corresponding sequences and $w_i(t)$ the synaptic efficacies. As for $I_\mathrm{d}(t)$, $n_p(t)$ and $b_p(t)$ was used as a time dependent scaling and bias.

2.2 Plasticity

We implemented a Hebbian plasticity rule for the basal synaptic weights given by the following update equation:

$$w_i(t+1) = w_i(t) + \mu_w\left(x_{p,i}(t) - \tilde{x}_{p,i}(t)\right)\left(y(t) - \tilde{y}\right) \tag{6}$$
$$\tilde{x}_{p,i}(t+1) = (1-\mu_{av})\tilde{x}_{p,i}(t) + \mu_{av}x_{p,i}(t+1) \tag{7}$$
$$\tilde{y}(t+1) = (1-\mu_{av})\tilde{y}(t) + \mu_{av}y(t+1) \tag{8}$$

Additionally, we used a synaptic normalization constraint

$$w_i(t) \to \frac{w_i(t)}{\|\mathbf{w}(t)\|} \tag{9}$$

θ_{p0}	0	V_d^t	0.25
θ_{p1}	-1	μ_b	10^{-3}
θ_d	0	μ_n	10^{-4}
α	0.3	μ_{av}	$5\cdot10^{-3}$
μ_w	$5\cdot10^{-5}$	I_p^t	0
V_p^t	0.25	I_d^t	0

Table 1: Model parameters

in each time step, where $\|\mathbf{w}(t)\|$ denotes the Euclidean norm of the synaptic weight vector.

For comparative reasons, the point neuron model was equipped with the same plasticity rule for the proximal weights as (6).

Additionally, the scaling and bias variables were changing dynamically according to the following homeostatic plasticity rules:

$$b_p(t+1) = b_p(t) + \mu_b\left[I_\mathrm{p}(t) - I_\mathrm{p}^t\right] \tag{10}$$
$$b_d(t+1) = b_d(t) + \mu_b\left[I_\mathrm{d}(t) - I_\mathrm{d}^t\right] \tag{11}$$
$$n_p(t+1) = n_p(t) + \mu_n\left[V_p^t - \left(I_\mathrm{p}(t) - \tilde{I}_p(t)\right)^2\right] \tag{12}$$
$$n_d(t+1) = n_d(t) + \mu_n\left[V_d^t - \left(I_\mathrm{d}(t) - \tilde{I}_d(t)\right)^2\right] \tag{13}$$
$$\tilde{I}_p(t+1) = (1-\mu_{av})\tilde{I}_p(t) + \mu_{av}I_\mathrm{p}(t+1) \tag{14}$$
$$\tilde{I}_d(t+1) = (1-\mu_{av})\tilde{I}_d(t) + \mu_{av}I_\mathrm{d}(t+1) \tag{15}$$

Here, I_p^t, I_d^t, V_p^t and V_p^t define targets for the temporal means and variances of I_p and I_d. The dynamic variables $\tilde{I}_p$ and $\tilde{I}_d$ are simply low-pass filtered running averages of I_p and I_d.

A list of all parameter values is given in Table 1.

3 Results

3.1 Increased Alignment between Basal and Apical Inputs

As a first test, we wanted to quantify the neuron's ability to align its basal input to the apical teaching signal. To do so, we defined the pearson correlation coefficient $\rho[I_\mathrm{p}, I_\mathrm{d}]$ between the basal and apical input current as our measure of interest. We determined this temporal correlation coefficient after simulating all plasticity mechanisms under stationary random input sequences with certain statistical properties that shall be explained in the following. As a starting point, we chose all $x_{p,i}(t)$ to be randomly drawn from a uniform distribution, where $x_{p,i}(t) \in (0,1)$. For $I_\mathrm{d}(t)$ to be fully 'reconstructable' by the basal input, $x_d(t)$ had to be some linear combination of all $x_{p,i}(t)$. Therefore, we chose $x_d(t) = \sum_{i=1}^{N} a_i x_{p,i}(t)$, where $\mathbf{a}$ is a random vector with unit length. Since we used a Hebbian learning scheme, we expected that the direction and magnitude

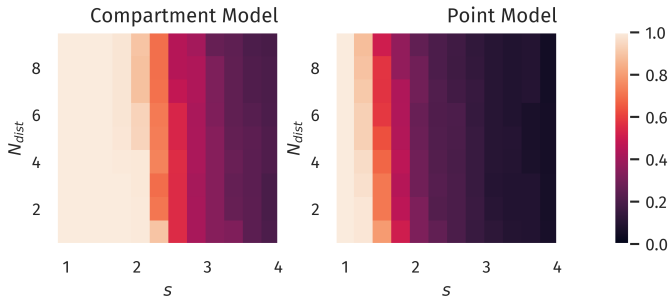

Figure 2: **Alignment between Basal and Apical Input.** Color encodes the Pearson correlation $\rho[I_p, I_d]$ for different number of orthogonal distraction directions N_{dist} and the corresponding scaling faction s.

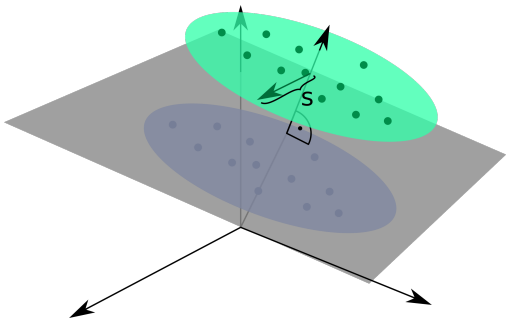

Figure 3: **Input Space for a Simple Linear Classification Task.** Two clusters of presynaptic basal activity were generated from multivariate Gaussian distributions. Here, s denotes the standard deviation orthogonal to the normal vector of the classification hyperplane.

of the principal components of the basal input would also significantly affect the outcome of the experiment: A large variance in the basal input orthogonal to the 'reconstruction vector' $\mathbf{a}$ should act as a distraction for the plasticity and reduce the resulting alignment between I_p and I_d. Therefore, we applied a transformation to the input sequences $x_{p,i}(t)$ that were parameterized by two quantities, a scaling factor s and the dimension N_{dist} of a subspace of the basal input space. A set of N_{dist} orthonormal basis vectors was randomly generated, which were also orthogonal to $\mathbf{a}$. Withing this N_{dist}-dimensional subspace, the input sequences $x_{p,i}(t)$ were then rescaled by the factor s.

After both $x_{p,i}(t)$ and $x_d(t)$ were generated, a simulation was run using all previously described plasticity mechanisms until the dynamic variables reached a stationary state. After this learning phase, another set of input sequences was generated using the previously described protocol and $\rho[I_\mathrm{p}, I_\mathrm{d}]$ was calculated. Note that plasticity was turned off in this phase. This entire procedure allowed us to calculate $\rho[I_\mathrm{p}, I_\mathrm{d}]$ as a function of the distraction parameters s and N_{dist}. The results for both neuron models is shown in Fig. 2. Here, the total number of basal inputs was $N = 10$. One can observe a decorrelation transition for both models. However, the compartment model supports a significantly stronger distraction in terms of the scaling factor s as compared to the point model. This was a first confirmation of our hypothesis that nonlinear interactions between basal and apical input could improve learning guided by top-down signals.

3.2 Supervised Learning in a Linear Classification Task

Next, we investigated if the observed differences would also improve the performance in an actual supervised learning task. For this purpose, we constructed presynaptic basal input as illustrated in Fig. 3. Written in vector form, each sample from the basal input was generated from the following expression:

$$\mathbf{x}_p(t) = \mathbf{b} + \mathbf{a}\left(c(t) + \sigma_a \zeta_a(t)\right) + s \cdot \sum_{i=1}^{N_{dist}} \zeta_{dist,i}(t)\mathbf{v}_{dist,i} \ . \tag{16}$$

Here, $\mathbf{b}$ is a random vector drawn uniformly from $(0,1)^N$, $\mathbf{a}$ is random unit vector as introduced in Section 3.1, $c(t)$ is a binary variable drawn from $\{-0.5, 0.5\}$ with equal probability and $\zeta_a(t)$ and the $\zeta_{dist,i}(t)$ are independent random Gaussian variables with zero mean and unit variance. Hence, σ_a simply denotes the standard deviation of each Gaussian cluster along the direction of the normal vector $\mathbf{a}$ and was set to $\sigma_a = 0.25$. Finally, the set of $\mathbf{v}_{dist,i}$ forms a randomly generated orthogonal basis of N_{dist} unit vectors which—just as in Section 3.1—are also orthogonal to $\mathbf{a}$. The free parameter s parameterized the standard deviation along this subspace orthogonal to $\mathbf{a}$. As indicated by the time dependence, the Gaussian and binary random variables were drawn for each time step. The vectors $\mathbf{b}$, $\mathbf{a}$, and $\mathbf{v}_{dist,i}$ were generated once before the beginning of a simulation run.

For the classification task, we created two output neurons receiving the same basal presynaptic input, while the top-down input encoded the correct linear classification in a one-hot scheme, that is

$$x_{d,0}(t) = 1 - \Theta\left(\left(\mathbf{x}_p(t) - \mathbf{b}\right)^T \mathbf{a}\right) \tag{17}$$

$$x_{d,1}(t) = \Theta\left(\left(\mathbf{x}_p(t) - \mathbf{b}\right)^T \mathbf{a}\right) \ , \tag{18}$$

where $\Theta(x)$ is the Heaviside step function.

As in the previous experiment, we ran a full simulation until all dynamic variables reached a stationary state. After this, a test run without plasticity and with the apical input turned off was used to evaluate the classification performance. For each sample, the index of the neuron with the highest activity was used as the predicted class. Accuracy was then calculated as the fraction of correctly classified samples.

The resulting accuracy as a function of N_{dist} and s is shown in Fig. 4. Albeit differences were small, the com-

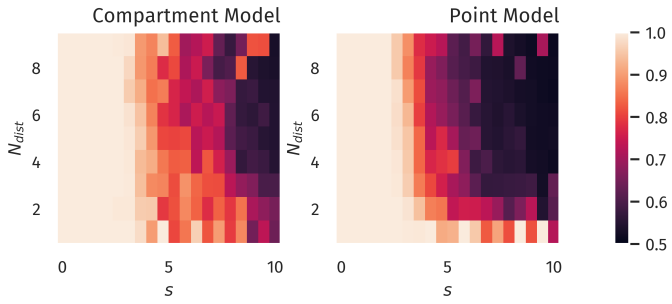

Figure 4: **Binary Classification Accuracy.** Fraction of correctly classified patterns as illustrated in Fig. 3, see Section 3.2.¸

partment model did show a better overall accuracy for the tested parameter range.

3.3 Non-Hebbian Learning Rules

Instead of Hebbian learning, we also considered a BCM-like learning rule for the basal weights (Bienenstock et al., 1982; Intrator and Cooper, 1992). The form of the BCM-rule we consider here reads

$$\Delta w_i \propto y\,(y - \theta_M)\,x_i - \epsilon w_i\ , \tag{19}$$

where θ_M is a threshold defining a transition from LTP to LTD and ϵ is an optional decay term on the weights. In the variant introduced by Law and Cooper (1994), the sliding threshold is simply the temporal average of the squared neural activity, $\theta_M = \langle y^2 \rangle$. In practice, this would be calculated as a running average, thereby preventing the weights from growing indefinitely.

However, for our compartment model, we chose to explicitly set the threshold to be the mean value between the high- and low-activity regime in our compartment model, i.e. $\theta_M = (1 + \alpha)/2$. By doing so, LTP was preferably induced if both basal and apical input was present at the same time. Furthermore, instead of the weight decay term, we chose to keep the weight normalization as introduced in (9). Obviously, for the point model, the reasoning behind our choice of θ_M did not apply. Still, to provide some level of comparability, we also ran simulations with a point model where the sliding threshold was calculated as a running average of y^2. Furthermore, we did not use weight normalization in this case, but chose to use a small weight decay term with $\epsilon = 0.1$. The results are shown in Fig. 5 (classification task) and Fig. 5 (Basal-Apical alignment). While the accuracy of the classification for the point model was at most comparable to Hebbian learning, the BCM-like rule for the compartment model significantly increased the accuracy for the tested parameter range (compare Fig. 4).

Likewise, the compartment model also significantly benefited from the BCM rule in terms of basal-apical alignment, while only marginal improvements could be observed for the point model (compare Fig. 6 with Fig. 2).

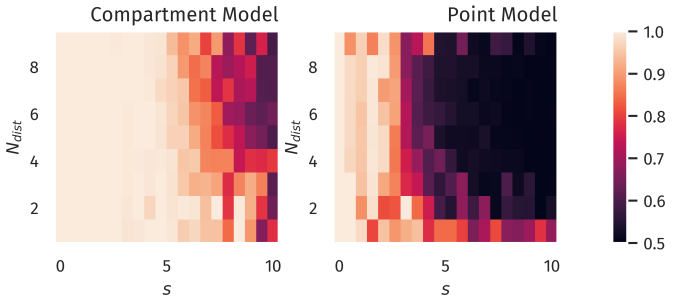

Figure 5: **Binary Classification Accuracy, BCM Rule.** Fraction of correctly classified patterns as illustrated in Fig. 3, after training with a BCM-like learning rule.¸

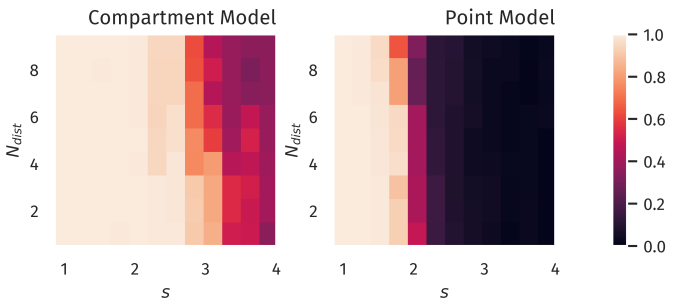

Figure 6: **Alignment between Basal and Apical Input, BCM Rule.** Color encodes the Pearson correlation $\rho[I_p, I_d]$ for different number of orthogonal distraction directions N_{dist} and the corresponding scaling faction s after training with a BCM-like rule. Compare to Fig. 2.

4 Discussion

To be continued

References

Bienenstock, E. L., Cooper, L. N., and Munro, P. W. (1982). Theory for the development of neuron selectivity: Orientation specificity and binocular interaction in visual cortex. *Journal of Neuroscience*, 2(1):32–48.

Guerguiev, J., Lillicrap, T. P., and Richards, B. A. (2017). Towards deep learning with segregated dendrites. *eLife*, 6.

Intrator, N. and Cooper, L. N. (1992). Objective function formulation of the BCM theory of visual cortical plasticity: Statistical connections, stability conditions. *Neural Networks*, 5(1):3–17.

Law, C. C. and Cooper, L. N. (1994). Formation of receptive fields in realistic visual environments according to the Bienenstock, Cooper, and Munro (BCM) theory. *Proceedings of the National Academy of Sciences of the United States of America*, 91(16):7797–7801.

Schiess, M., Urbanczik, R., and Senn, W. (2016). Somato-dendritic Synaptic Plasticity and Error-backpropagation in Active Dendrites. *PLoS Computational Biology*, 12(2):1004638.

Shai, A. S., Anastassiou, C. A., Larkum, M. E., and Koch, C. (2015). Physiology of Layer 5 Pyramidal Neurons in Mouse Primary Visual Cortex: Coincidence Detection through Bursting. *PLOS Computational Biology*, 11(3).

Urbanczik, R. and Senn, W. (2014). Learning by the Dendritic Prediction of Somatic Spiking. *Neuron*, 81(3):521–528.