Burst-dependent synaptic plasticity can coordinate learning in hierarchical circuits

Alexandre Payeur[1•] Jordan Guerguiev[2,3•] Friedemann Zenke[4] Blake A. Richards[5,6,7,8*•] and Richard Naud[1,9,*•]

1 uOttawa Brain and Mind Institute, Centre for Neural Dynamics, Department of Cellular and Molecular Medicine, University of Ottawa, Ottawa, ON, Canada
2 Department of Biological Sciences, University of Toronto Scarborough, Toronto, ON, Canada
3 Department of Cell and Systems Biology, University of Toronto, Toronto, ON, Canada
4 Friedrich Miescher Institute for Biomedical Research, Basel, Switzerland
5 Mila, Montréal, QC, Canada
6 Department of Neurology and Neurosurgery, McGill University, Montréal, QC, Canada
7 School of Computer Science, McGill University, Montréal, QC, Canada
8 Learning in Machines and Brains Program, Canadian Institute for Advanced Research, Toronto, ON, Canada
9 Department of Physics, University of Ottawa, Ottawa, ON, Canada

∗ Corresponding authors, email: rnaud@uottawa.ca, blake.richards@mcgill.ca
• Equal contributions

Abstract

Synaptic plasticity is believed to be a key physiological mechanism for learning. It is well-established that it depends on pre and postsynaptic activity. However, models that rely solely on pre and postsynaptic activity for synaptic changes have, to date, not been able to account for learning complex tasks that demand hierarchical networks. Here, we show that if synaptic plasticity is regulated by high-frequency bursts of spikes, then neurons higher in the hierarchy can coordinate the plasticity of lower-level connections. Using simulations and mathematical analyses, we demonstrate that, when paired with short-term synaptic dynamics, regenerative activity in the apical dendrites, and synaptic plasticity in feedback pathways, a burst-dependent learning rule can solve challenging tasks that require deep network architectures. Our results demonstrate that well-known properties of dendrites, synapses, and synaptic plasticity are sufficient to enable sophisticated learning in hierarchical circuits.

Introduction

The current canonical model of synaptic plasticity in the cortex is based on the co-occurrence of activity on the two sides of the synapse, pre and postsynaptic [44, 79]. The occurrence of either long-term depression (LTD) or long-term potentiation (LTP) is controlled by specific features of pre and postsynaptic activity [5, 21, 28, 34, 49, 58, 61, 88, 89] and a more global state of neuromodulation [3, 31, 32, 36, 48, 57, 65, 79, 86, 90]. However, local learning rules by themselves do not provide a guarantee that behavioral metrics will improve. With neuromodulation driven by an external reward/punishment mechanism, this guarantee is achievable. But, such learning is very slow in tasks that require large or deep networks because a global signal provides very limited information to neurons deep in the hierarchy [9, 59, 94]. Thus, an outstanding question is (Fig. 1): how can neurons higher in the hierarchy signal to other neurons — sometimes

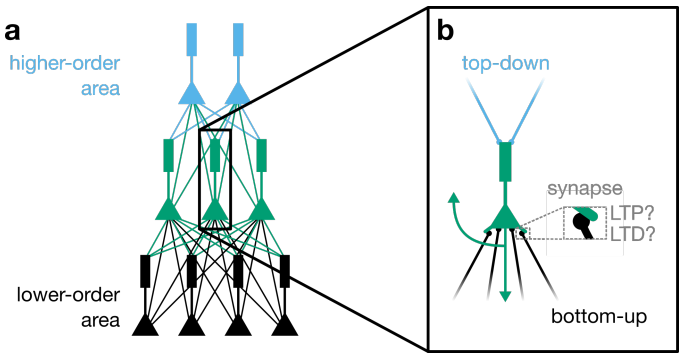

Figure 1. The credit assignment problem for hierarchical networks. (**a**) Illustration of a hierarchical neural network with feedforward and feedback connections. (**b**) For orchestration of learning in this network, the representations in higher-level neurons should steer the plasticity of connections at a lower level.

multiple synapses lower — whether to engage in LTP or LTD in order to improve behavior [79]? This question is sometimes referred to as the "credit assignment problem", essentially, how can we assign credit for any errors or successes to neurons that are multiple synapses away from the output [77]?

As established in non-hierarchical systems, such as the electrosensory lateral line lobe of the electric fish [8, 14, 66] or the cerebellum [15], feedback connections on dendrites are well-poised to orchestrate learning [78]. But for credit assignment in hierarchical networks, these connections should obey four constraints: 1) Feedback must *steer* the sign and magnitude of plasticity. 2) Feedback signals from higher-order areas should be *multiplexed* with feedforward signals from lower-order areas so that credit information can percolate down the hierarchy with minimal disruption to sensory information. 3) There should be some degree of *alignment* between feedback connections and feedforward connections. 4) Integration of credit-carrying feedback signals should be close to *linear* and avoid saturation (i.e., feedback signals should be linear with respect to any credit information). Experimental and theoretical work have addressed steering [16, 34], multiplexing [1, 45, 51, 69], alignment [2, 17, 18, 59] or linearity [67] in isolation, often by learning in an offline fashion [2, 4, 40, 47, 59, 83, 84, 101], without learning rules based on spikes [4, 53, 80, 82–84] or without learning to solve tasks that necessitate hierarchical processing. Thus, it remains unclear whether a single set of cellular and subcellular mechanisms address all four requirements for orchestrating learning in cortical hierarchies efficiently.

Here, we address the credit assignment problem with a spike-based learning rule that models how high-frequency bursts determine the sign of synaptic plasticity [33, 49, 58, 70]. Extending the work of Körding and König (2001) [53], we show that combining properties of apical dendrites [55] with this burst-dependent learning rule allows feedback to steer plasticity. We further show that feedback information can be multiplexed across multiple levels of a hierarchy when feedforward and feedback connections have distinct short-term plasticity (STP) [38, 87]. Using spiking simulations, we demonstrate that these mechanisms can be used to coordinate learning across a hierarchical circuit. A coarse-grained equivalent of these dynamical properties will, on average, lead to learning that approximates loss-function gradients that are used in machine learning for training deep neural networks. We further show that this biological approximation to loss-function gradients is improved by a burst-dependent learning rule performing the alignment of feedback weights with feedforward weights, as well as recurrent connections that linearize credit signals. Finally, we show that networks trained with these mechanisms can learn to classify complex image patterns with high accuracy. Altogether, our work highlights that well-known properties of dendritic excitability, synaptic transmission, and synaptic plasticity are sufficient to solve the credit assignment problem in hierarchical networks.

Results

A burst-dependent rule enables top-down steering of plasticity

Experimental work has demonstrated that the sign of plasticity can be determined by patterns of pre and postsynaptic activity. The most common formulation of this is spike-timing-dependent plasticity (STDP), wherein the timing of pre and postsynaptic spikes is what determines whether LTP or LTD occurs [10, 35, 61]. However, there is also evidence suggesting that in many circuits, particularly mature ones [64], the principal determinant of plasticity is the level of postsynaptic depolarization, with large depolarization leading to LTP and small depolarization leading to LTD [5, 21, 28, 89]. Importantly, one of the easiest ways to induce large magnitude depolarization in dendrites is via backpropagation of high-frequency bursts of action potentials [50]. Thus, evidence suggests that the sign of synaptic plasticity is largely dependent on postsynaptic bursting [33, 49, 58, 89].

To explore how high-frequency bursting could control learning in biological neural networks, we formulated a burst-dependent plasticity rule as an abstraction of the experimental data. We consider a burst to be any occurrence of at least two spikes with a short (e.g. under 20 ms) interspike interval [7]. Following Ref. [69], we further define an event as either an isolated single spike or a burst. Thus, for a given neuron's output, there is an event train (similar to a spike train, except that events can be either bursts or single spikes) and a burst train, which comprises a subset of the events (see Methods). The learning rule states that the change over time of a synaptic weight between postsynaptic neuron i and presynaptic neuron j, dw_{ij}/dt, results from a combination of an eligibility trace of presynaptic activity, $\widetilde{E}_j$, and the potentiating (or depressing) effect of the burst train B_i (or event train E_i) of the postsynaptic cell (Fig. 2a):

$$\frac{dw_{ij}}{dt} = \eta[B_i(t) - \overline{P}_i(t)E_i(t)]\widetilde{E}_j(t). \tag{1}$$

The variable $\overline{P}_i$ controls the relative strength of burst-triggered potentiation and event-triggered depression. To ensure a finite growth of synaptic weights, we set this to a moving average of the proportion of events that are bursts in postsynaptic neuron i, with a slow ($\sim 1 - 10$ s) time scale (see Methods).

The plasticity rule stipulates that when a presynaptic input is paired with a postsynaptic burst LTP is induced, and otherwise, LTD results (Fig. 2a) [33, 49, 58, 64, 70]. Using this rule, we simulated a series of synaptic plasticity experiments from the experimental and computational literature. First, we examined a frequency-dependent STDP protocol [89]. We found that when the spike pairing frequency is low, LTD is produced, and when the pairing frequency is high, LTP is produced (Fig. 2b). This matches previous reports on frequency-dependent STDP and shows that a burst-dependent synaptic plasticity rule can explain this data. Then, we explored the behavior of our rule when the pre and postsynaptic neuron fire independently according to Poisson statistics [74] (Fig. 2c). Experimental results have established that in such a situation the postsynaptic firing rate should determine the sign of plasticity [89]. As in similar learning rules [74], we found that a burst-dependent plasticity rule produces exactly this behavior (Fig. 2c). Notably, the Bienenstock-Cooper Munro (BCM) model [11] has posited that the switching point between LTD and LTP depends on a nonlinear moving average of past activity. In the present case, the adaptive threshold is $\overline{P}_i$, a nonlinear function of the past postsynaptic activity. These results demonstrate that a burst-dependent plasticity rule is capable of uniting a series of known experimental and theoretical results.

The burst-dependent rule suggests that feedback-mediated steering of plasticity could be achieved if there were a mechanism for top-down control of the likelihood of a postsynaptic burst. To illustrate this, in Fig. 2d we simulated another protocol wherein events were generated with Poisson statistics, and each event could become a burst with probability p (x axis in Fig. 2d). Manipulating this burst probability against the initial burst probability estimate ($\overline{P}_i(0) = 20\%$) controlled the occurrence of LTP and LTD, while changing the pre and postsynaptic event rates simply modified the rate of change of the weight (but not the transition point between LTP and LTD). This shows that one way for neurons to control the sign of plasticity to ensure effective learning may be to regulate the probability of high-frequency bursts.

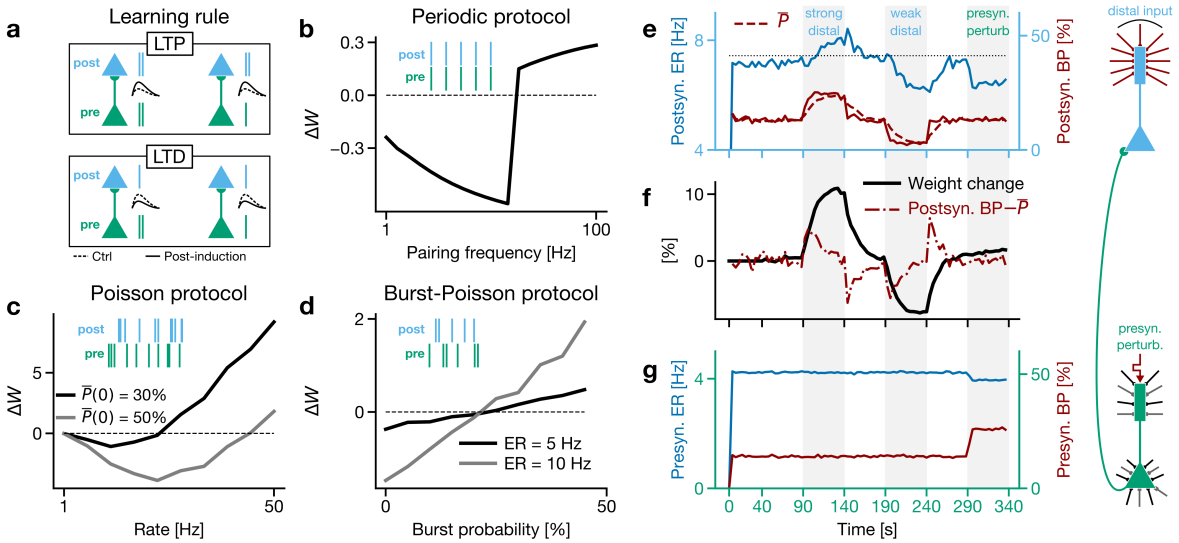

Figure 2. Burst-dependent plasticity rule. (**a**) Schematics of the learning rule. The occurrence of a postsynaptic burst leads to potentiation (top) whereas an isolated postsynaptic spike leads to depression of the synapse (bottom). (**b-d**) Net weight change for different pairing protocols. (**b**) The periodic protocol consisted of 15 sequences of 5 pairings, separated by a 10 s interval. We used pairings with $t_{\mathrm{post}} = t_{\mathrm{pre}}$. (**c**) For the Poisson protocol, the pre and postsynaptic activities were Poisson spike trains with equal rates. The protocol was repeated with different initial time-average burst probabilities ($\overline{P}$). (**d**) For the burst-Poisson protocol, pre and postsynaptic Poisson events were generated at a fixed rate (ER). For each event, a burst was produced with a probability that varied from 0 to 50%. (**e-g**) Impact of distal inputs on burst probability and feedforward synaptic weights for constant presynaptic event rate. Positive distal input (90–140 s) increases burst probability (**e**) and strengthens feedforward synapses (**f**). Negative distal input (190–240 s) decreases burst probability and weakens synapses. A dendritic input to the *presynaptic* neuron (290–340 s) increases its burst probability and mildly affects its event rate (**g**), but does not significantly change the weights (**f**). (**e**) Event rate (ER; blue), burst probability (BP; solid red curve) and estimated BP (dashed red curve) for the postsynaptic population. The black dotted line indicates the prestimulation ER and serves as a reference for the variations of the ER with plasticity. (**f**) Weight change relative to the initial average value of the weights. (**g**) Same as panel **e**, but for the presynaptic population. For the schematic on the right-hand side, black and grey axonal terminals onto the presynaptic (green) population represent Poisson input noise; such noise is absent for the postsynaptic (light blue) population for this simulation.

Interestingly, evidence suggests that in cortical pyramidal neurons of sensory cortices the probability of generating high-frequency bursts is controlled by inputs to the distal apical dendrites and their activation of voltage-gated calcium channels (VGCCs) [26, 30, 55, 56, 98]. Anatomical and functional data has shown that these inputs often come from higher-order cortical or thalamic regions [19, 29].

We wondered whether combining a burst-dependent plasticity rule with regenerative activity in apical dendrites could permit top-down signals to act as a "teaching signal", instructing the sign of plasticity in a neuron. To explore this, we ran simulations of pyramidal neuron models with simplified VGCC kinetics in the apical dendrites (see Methods). We found that by manipulating the distal inputs to the apical dendrites we could control the number of events and bursts in the neurons independently (Figs. 2e,g). Importantly, the inputs to the apical dendrites in the postsynaptic neurons were what regulated the number of bursts, and this also controlled changes in the synaptic weights, through the burst-dependent learning rule. When the relative proportion of bursts increased, the synaptic weights potentiated on average, and when the relative proportion of burst decreased, the synaptic weights depressed (Fig. 2f). Modifying the proportion of bursts in the presynaptic neurons had little effect on the weights (see the rightmost gray shaded area in Fig. 2e-g). The sign of plasticity was independent of the number of events, though the magnitude was not. Therefore, while the number of events contributed to the determination of the magnitude of changes, the top-down inputs to the apical dendrites controlled the sign of plasticity. In this way, the top-down inputs acted as a "teaching signal" that determined whether LTP or LTD would occur. These results show that a burst-dependent learning rule paired with the control of bursting provided by apical dendrites enables a form of top-down steering of synaptic plasticity in an online, local, and spike-based manner.

Dendrite-dependent bursting combined with short-term plasticity supports multiplexing of feedforward and feedback signals

The question that naturally arises from our finding that top-down inputs can steer synaptic plasticity via a burst-dependent rule is whether feedback can steer plasticity without affecting the communication of bottom-up signals. Using numerical simulations, we previously have demonstrated that in an ensemble of pyramidal neurons the inputs to the perisomatic and distal apical dendritic regions can be distinctly encoded using the event rate computed across the ensemble of cells and the percentage of events in the ensemble that are bursts (the "burst probability"), respectively [69]. When communicated by synapses with either short-term facilitation (STF) or short-term depression (STD), this form of "ensemble multiplexing" may allow top-down and bottom-up signals to be simultaneously transmitted through a hierarchy of pyramidal neurons.

To explore this possibility, we conducted simulations of two reciprocally connected ensembles of pyramidal neurons along with interneurons providing feedforward inhibition. One ensemble received currents in the perisomatic region and projected to the perisomatic region of the other ensemble (Fig. 3a, green ensemble). The other ensemble (Fig. 3a, light blue) received currents in the distal apical compartment and projected to the distal apical compartment of the first ensemble. As such, we considered the first ensemble to be "lower" (receiving and communicating bottom-up signals), and the other to be "higher" (receiving and communicating top-down signals) in the hierarchy. Furthermore, we made one key assumption in these simulations. We assumed that the synapses in the perisomatic regions were short-term depressing, whereas those in distal apical dendrites were short-term facilitating. Additionally, we assumed that the inhibitory interneurons targeting the perisomatic region possessed STD synapses, and the inhibitory interneurons targeting the distal apical dendrites possessed STF synapses. These properties are congruent with what is known about parvalbumin-positive and somatostatin-positive interneurons [63, 76, 93], which target the perisomatic and apical dendritic regions, respectively.

In these simulations, we observed that currents injected into the lower ensemble's perisomatic compartments were reflected in the event rate of those neurons (Fig. 3c3), whereas the currents injected into the distal apical dendrites of the higher ensemble were reflected in the burst probability of those neurons (Fig. 3b2). Importantly, though, we also observed that these signals were simultaneously propagated up and down. Specifically, the input to the lower ensemble's perisomatic compartments

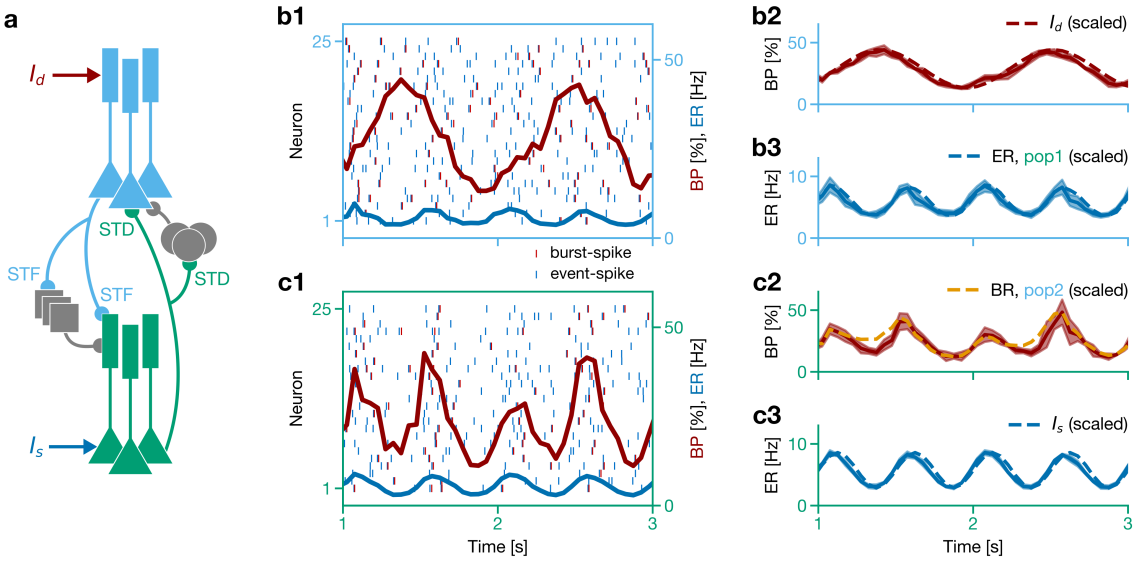

Figure 3. Dendrite-dependent bursting combined with short-term plasticity supports the simultaneous propagation of bottom-up and top-down signals. (**a**) Schematic of the network. Lower-level pyramidal neurons (green) received a somatic current I_s and projected with STD synapses to the somatic compartments of both a higher-level pyramidal neuron population (light blue) and to a population providing disynaptic inhibition (grey discs). The dendritic compartments of the light blue population received a current I_d. The light blue neurons innervated with STF synapses both the dendritic compartments of the green pyramidal neurons and a population providing disynaptic inhibition (grey squares). Results referring to the light blue and green population appear in panels **b1-b3** and **c1-c3**, respectively. (**b1, c1**) Raster plots of 25 out of the 4000 neurons per pyramidal population. Blue ticks show the start of an event, being either a burst or an isolated spike. Red ticks are the second spike in a burst; the remaining spikes in a burst are not shown. The population event rate (ER, blue line) and the burst probability (BP, red line) are superposed. (**b2-b3**), (**c2-c3**) Population averages ± 2 standard deviations over five realizations of the Poisson noise applied to all neurons in the network. In each panel, the encoded input signal (dashed line) has been linearly scaled. For instance, in panel **c2** the BP of the green population encodes the BR of the light blue population. The bin size used in the population averages was 50 ms.

was also encoded by the higher ensemble's event rate (Fig. 3b3). As well, the burst rate of the higher ensemble was encoded by the lower ensemble's burst probability (Fig. 3c2). In this way, the lower ensemble had access to information about the signal transmitted to the higher ensemble's distal apical dendrites, as well as the higher ensemble's event rate.

These simulations demonstrate that if bottom-up connections to perisomatic regions and perisomatic inhibition rely on STD synapses, while top-down connections to apical dendrites and distal dendritic inhibition utilize STF synapses, then ensembles of pyramidal neurons are capable of simultaneously processing both a top-down signal and a bottom-up signal. We conclude that a top-down signal to apical dendrites can 1) control the sign of plasticity locally (steering; Fig. 2) and 2) be communicated to lower ensembles without affecting the flow of bottom-up information (multiplexing; Fig. 3).

Combining a burst-dependent plasticity rule with short-term plasticity and apical dendrites can solve the credit assignment problem

To test whether STP, dendrite-dependent bursting and a burst-dependent learning rule can act simultaneously in a hierarchy, we built a simulation of ensembles of pyramidal neurons arranged in three layers, with two ensembles of cells at the input, one ensemble of cells at the output, and two ensembles of cells in the middle (the "hidden" layer; Fig. 4a). The distal dendrites of the top ensemble received "teaching"

signals indicating desired or undesired outputs. No other teaching signal was provided to the network. As such, the hidden layer ensembles were informed of the suitability of the output only via the signals they received from the output ensemble's bursts. Currents injected into the somatic compartments of the input layer populations controlled their activity levels in accordance with the learning task to be discussed below. Compared to Figs. 2-3, for this simulation we made a few modifications to synaptic transmission and pyramidal neuron dynamics to streamline the burst-event multiplexing and decoding (see Methods). The most important addition, however, is that we modified the learning rule in Eq. 1 by multiplying the right-hand side by an additional global term, $M(t)$, that gates plasticity. This term abstracts a number of possible sources of control of plasticity, like local inhibition [67], burst sizes [99] or transient neuromodulation [73,86,100]. Importantly, $M(t)$ in our model gates plasticity without changing its sign, contrary to some models on the role of neuromodulation in plasticity [36]. Its role was to make sure that plasticity elicited by the abrupt onset and offset of each training example does not overcome the plasticity elicited by the teaching signal, i.e. it was used to ensure a supervised training regime. We accomplished this by setting $M = 0$ when no teaching signal was present at the output layer and $M = 1$ under supervision. In this way, we ensured that the teaching signal was the primary driver of plasticity.

We trained our 3-layer network on the exclusive or (XOR) task, wherein the network must respond with a high output if only one input pool is active, and low output if neither or both input pools are active (Fig. 4). We chose XOR as a canonical example of a task that requires a nonlinear hierarchy with appropriate credit assignment for successful learning. Before learning, the network was initialized such that the output pool treated any input combination as roughly equivalent (Fig. 4c, dashed line). To compute XOR, the output pool would have to learn to reduce its response to simultaneously active inputs and increase its response to a single active input.

We set up the network configuration (Fig. 4a) to address a twofold question:

(1) Would an error signal applied to the top-layer neurons' dendrites be propagated downward adequately? (2) Would the burst-dependent learning rule combine top-down signals with bottom-up information to make the hidden-layer neurons better feature detectors for solving XOR?

Importantly, if the answer to these two questions were true, we would expect that the two hidden ensembles would learn different features if they receive different feedback from the output. To test this, we provided hidden pool 1 with positive feedback from the output, and hidden pool 2 with negative feedback (Fig. 4a, light blue symbols). With this configuration, adequate error propagation to the two hidden pools would make their responses diverge with learning, and the output pool would learn to take advantage of this change. Indeed, the XOR task was solved after training (Fig. 4c, solid line).

To understand how this solution was aided by appropriate credit assignment, we examined the information about the top-down teaching signals in each layer. According to the learning rule, plasticity can be steered by controlling the instantaneous propensity to burst with respect to a moving average of the burst probability (see term $B_i - \overline{P}_i E_i$ in Eq. 1 and Fig. 2e-f). In the output pool, the error signal applied to the apical dendrites induced a temporary decrease in the burst probability when the input pools were both active or both inactive, and a temporary increase when only one input pool was active (Fig. 4d). These changes in the output burst probability modified the output burst rate, which was propagated to the hidden pools. As mentioned above, the hidden pools received top-down signals with different signs (Fig. 4e1-2, orange lines), and indeed their respective burst probabilities were altered in opposite directions (Fig. 4e1-2, red lines). Due to these distinct top-down signals and the adaptive threshold $\overline{P}_i$, the hidden pools' response diverged during learning (Fig. 4f1-2). For instance, hidden pool 1 reduced its responses to both inputs being active, while hidden pool 2 increased its response. These changes were due to the top-down control of the plasticity of synapses between the input and hidden pools. We verified that solving this task depends on the plasticity of connections from input to hidden units, but only weakly on the size of the ensembles (Fig. S3). These results demonstrate that the propagation of errors using bursts and the burst-dependent learning rule can combine to achieve hierarchical credit assignment in ensembles of pyramidal neurons.

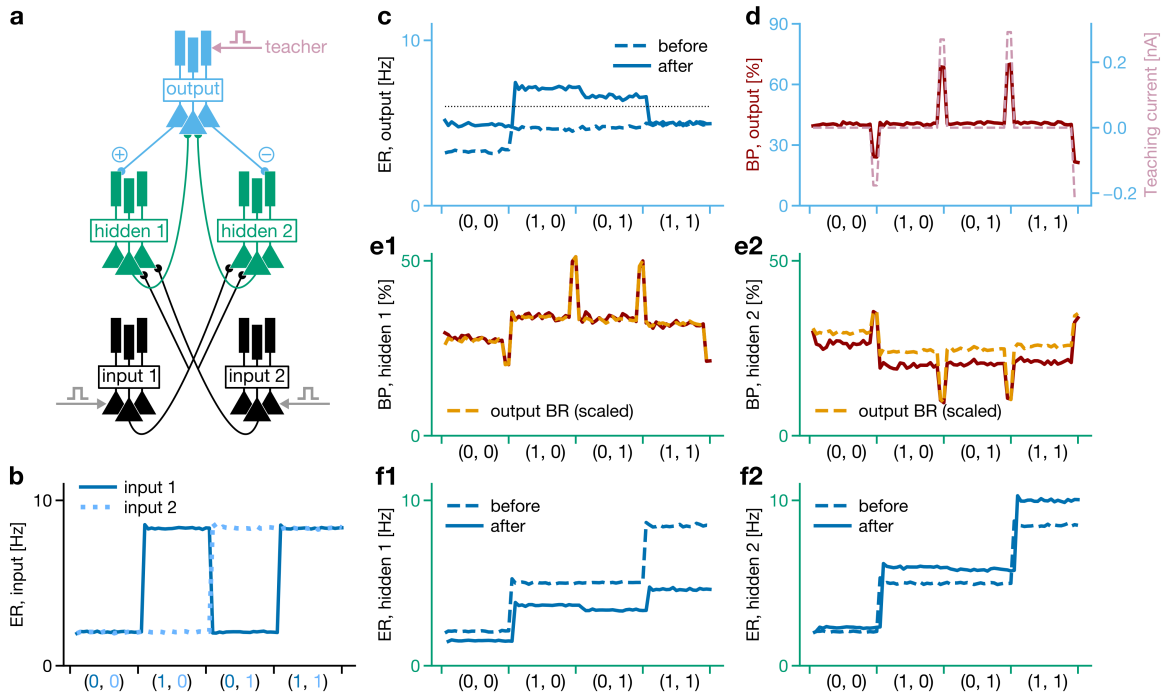

Figure 4. Burst-dependent plasticity can solve the credit assignment problem for the XOR task. (**a**) Each neuron population contained 2000 pyramidal neurons. Feedforward connections transmitted events, while feedback connections transmitted bursts. The teacher (pink arrow) was applied by injecting a hyperpolarizing current into the output ensemble's dendrites if their event rate was high in the presence of inputs that are either both active or both inactive. A depolarizing current was injected into the output ensemble's dendrites if their event rate was low when only one of the inputs was active. The activity of the input populations was controlled by somatic current injections (grey arrows). The $\oplus$ and $\ominus$ symbols represent the initialization of the feedback synaptic weights as mainly excitatory or inhibitory. (**b**) Input layer event rates (ERs) for the four input conditions presented sequentially in time. The duration of each example was 20 s. (**c**) Output ER before and after learning. The output ensemble acquired strong firing (event rate above the dotted line) at the input conditions associated with "true" in XOR. (**d**) During learning, the dendritic input (dashed pink) applied to the output ensemble's neurons controlled their burst probability in the last two seconds of the input condition. (**e1-e2**) During learning, the burst rate (BR) at the output layer is encoded into the BP of the hidden layer to propagate the error. For the hidden-2 population, this inherited credit is inverted with respect to that in the hidden-1 population. (**f1-f2**) After (full line) vs. before (dashed line) learning for the hidden layer. ER decreased in hidden-1 but increased in hidden-2. The bin size used in the population averages was 1 s.

Burst-dependent plasticity promotes linearity and alignment of feedback

Having demonstrated that a burst-dependent learning rule in pyramidal neurons enables online, local, spike-based solutions to the credit assignment problem, we were interested in understanding the potential relationship between this algorithm and the gradient-descent-based algorithms used for credit assignment in machine learning. To do this, we wanted to derive the average behavior of the burst-dependent learning rule at the coarse-grained, ensemble-level, and determine whether it provided an estimate of a loss-function gradient. More precisely, in the spirit of mean-field theory and linear-nonlinear rate models [27, 75, 85], we developed a model where each unit represents an ensemble of pyramidal neurons, with event rates, burst probabilities, and burst rates as described above (Fig. S8). Specifically, for an ensemble of pyramidal neurons, we defined $e(t)$ and $b(t)$ as ensemble averages of the event and burst trains, respectively. Correspondingly, $p(t) = b(t)/e(t)$ refers to the ensemble-level burst probability. We then defined the connection weight between an ensemble of presynaptic neurons and an ensemble of postsynaptic neurons, $W_{\text{post,pre}}$, as the effective impact of the presynaptic ensemble on the postsynaptic ensemble, taking into consideration potential polysynaptic interactions. Note that this means that the ensemble-level weight, $W_{\text{post,pre}}$, can be either positive or negative, as it reflects the cumulative impact of both excitatory and inhibitory synapses (see Supplemental Materials).

Our goal was then to derive the ensemble-level weight updates from the burst-dependent plasticity rule (Eq. 1). We assumed that any given pair of neurons were only weakly correlated on average, a reasonable assumption if the synaptic weights in the circuit are small [25]. Moreover, decorrelation between neurons is observed when animals are attending to a task [25], which suggests that this is a reasonable assumption for active processing states. We further assumed that the neuron-specific moving average burst probability ($\overline{P}_i$) is independent of the instantaneous occurrence of events. Using these assumptions, it can be shown (see Supplemental Materials) that the effective weight averaged across both pre and postsynaptic ensembles obeys:

$$\frac{dW_{\text{post,pre}}}{dt} = \eta M(t)[p_{\text{post}}(t) - \overline{p}_{\text{post}}(t)]e_{\text{post}}(t)e_{\text{pre}}(t) \tag{2}$$

where the learning rate η is different from that appearing in Eq. 1, and $\overline{p}_{post}(t)$ is a ratio of moving averages for the postsynaptic burst rate and event rate. This learning rule can be shown to implement an approximation of gradient descent for hierarchical circuits, like the backpropagation-of-error algorithm [81]. Specifically, if we assume that the burst probabilities remain in a linear regime (linearity), that the feedback synapses are symmetric to the feedforward synapses (alignment), and that error signals are received in the dendrites of the top-level ensembles, then $-[p_{\text{post}}(t) - \overline{p}_{\text{post}}(t)]e_{\text{post}}(t)$ is equivalent to the error signal sent backwards in backpropagation (see Supplemental Materials). For the sake of computational efficiency, when simulating this ensemble-level learning, we utilized simplifications to the temporal dynamics (i.e. we implemented a discrete-time version of the rule), though the fundamental computations being implemented were identical to the continuous-time equation above (see Methods and Supplemental Materials).

The assumptions of feedback linearity and alignment can be supported by the presence of additional learning mechanisms. First, we examined learning mechanisms to keep the burst probabilities in a linear regime. Inspired by a previous study that used distal apical inhibition to control error signals [82], and consistent with our previous results, we used recurrent excitatory and inhibitory inputs to control the apical compartments' potential (Fig. 5a). These dendrite-targeting inputs propagated bursts from neural ensembles at the same processing stage in the hierarchy. We found that a simple homeostatic learning rule (see Methods) could learn to keep burst probabilities in a linear regime, thus improving gradient estimates (Fig. 5b). Second, we explored potential mechanisms for learning weight symmetry. Symmetry between feedforward and feedback weights is an implicit assumption of many learning algorithms that approximate loss-function gradients. However, it has been shown that it is possible to learn weight symmetry [17]. In one classic algorithm [52], weight symmetry is obtained if feedforward and feedback weights are updated with the same error signals, plus some weight decay [2]. In our model, this form of feedback weight update was implemented locally because the error signal used to update the feedforward weights in discrete time is the deviation of the burst rates from the moving average baseline, and this, we

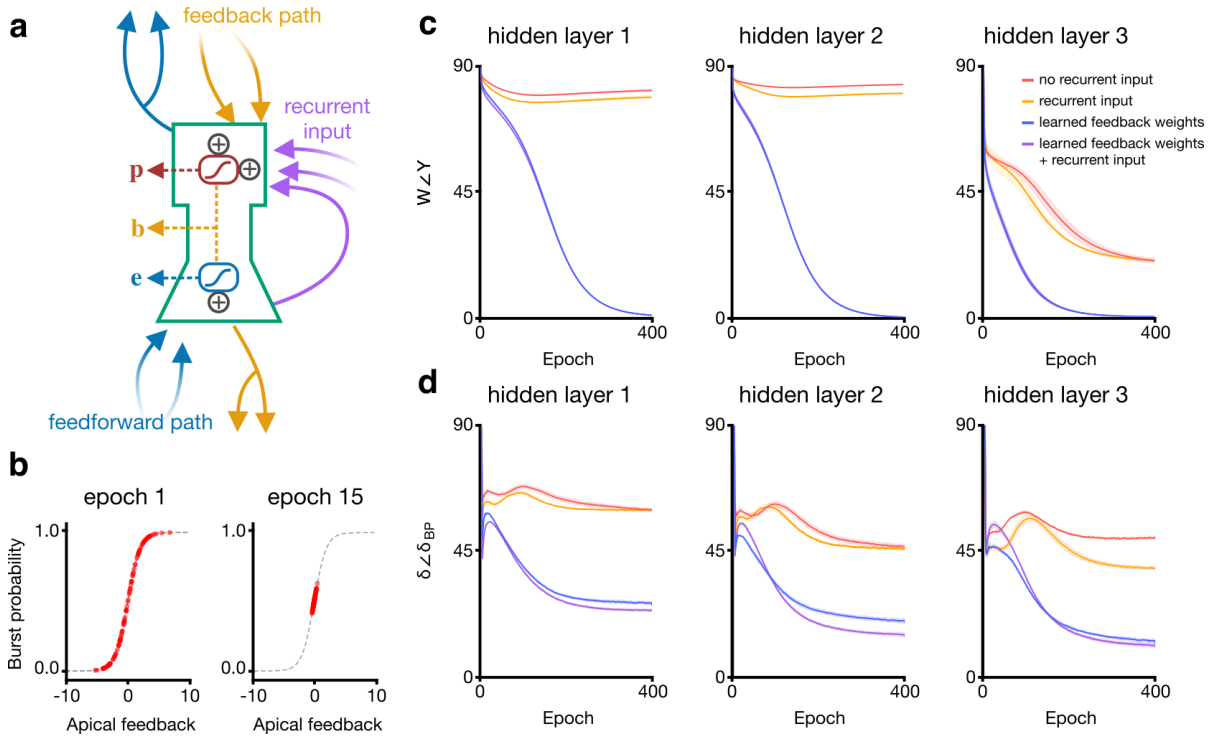

Figure 5. Burst-dependent plasticity of recurrent and feedback connections promotes gradient-based learning by linearizing and aligning feedback. (**a**) Diagram of a hidden-layer unit in the rate model. Each unit (green outline) in the network represents an ensemble of pyramidal neurons. Recurrent inputs (purple arrows) from all ensembles in a layer provide homeostatic control of the dendritic potential. (**b**) Throughout learning, recurrent weights were updated in order to push the burst probabilities towards the linear regime. (**c**) Alignment of feedback weights Y and feedforward weights W for three layers in a three-hidden-layer network trained on MNIST. Each hidden layer contained 500 units. Homeostatic recurrent inputs slightly reduce the angle between the two sets of weights, denoted $W\angle Y$, while learning on the feedback weights dramatically improves weight alignment. Each datapoint is the angle between feedforward and feedback weights at the start of a training epoch. (**d**) Angle between our weight updates (δ) and those prescribed by the backpropagation algorithm (δ_{BP}), for three layers in a three-hidden-layer network trained on MNIST. Recurrent inputs slightly improve the approximation to backpropagation, whereas learning on the feedback weights leads to a much closer correspondence. Each datapoint is the average angle between weight updates during a training epoch. In **c** and **d**, results are displayed as mean $\pm$ std over $n = 5$ trials.

propose, is also determining the updates to the feedback weights (see Methods). When we implemented this form of learning on the ensemble-level feedback weights we observed rapid weight alignment (Fig. 5c) and convergence to a loss-function gradient (Fig. 5d). Altogether, these results demonstrate that the burst-dependent learning rule, averaged across ensembles of pyramidal neurons, and paired with biologically plausible learning rules for recurrent inputs and feedback connections, can provide a good estimate of loss-function gradients in hierarchical networks.

Ensemble-level burst-dependent plasticity in deep networks can support good performance on standard machine learning benchmarks

We wanted to determine whether the ensemble-level learning rule could perform well on difficult tasks from machine learning that previous biologically plausible learning algorithms have been unable to solve. Specifically, we built a deep neural network comprised of pyramidal ensemble units that formed a series of convolutional layers followed by fully-connected layers (Fig. 6a). We then trained these networks

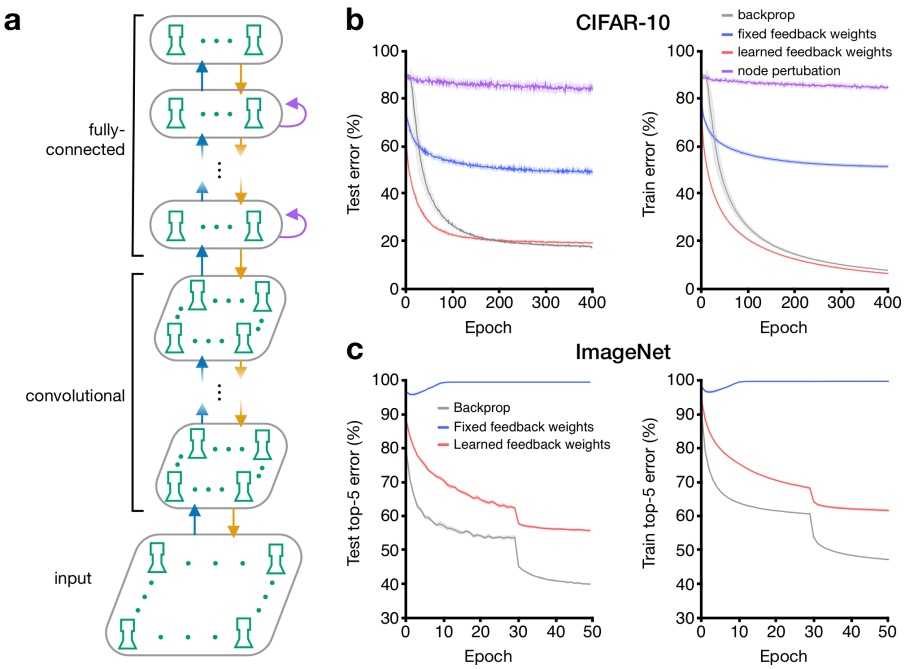

Figure 6. Ensemble-level burst-dependent plasticity supports learning in deep networks. (**a**) The deep networks consisted of an input layer, a series of convolutional layers, and a series of fully-connected layers. Layers were connected with sets of feedforward weights (blue arrows) and feedback weights (orange arrows). Fully-connected hidden layer contained recurrent connections (purple arrows). (**b**) Our learning rule, combined with learning of the feedback weights, was able to reach the performance of the backpropagation algorithm (backprop) on the CIFAR-10 classification task. (**c**) A network trained using our learning rule was able to learn to classify images in the ImageNet dataset when feedback weights were also updated. In **b** and **c**, results are displayed as mean $\pm$ std over $n = 5$ trials.

on two challenging image categorization datasets that previous biologically plausible algorithms have struggled with: CIFAR-10 and ImageNet [6].

The training throughout the network used our burst-dependent plasticity rule and recurrent inputs for linearization. For the CIFAR-10 dataset, we observed a classification test error rate of 20.1 % after 400 epochs (where an epoch is a pass through all training images), similar to the error rate achieved with full gradient descent in a standard artificial neural network (Fig. 6b). Training the feedback weights was critical for enabling this level of performance on CIFAR-10, as fixed feedback weights led to much worse performance, in-line with previous results [6]. Furthermore, rich unit-specific feedback signals were critical. A network trained using a global reward signal, while theoretically guaranteed to follow gradient descent on average [94,95], was unable to achieve good performance on CIFAR-10 in a reasonable amount of time (Fig. 6b, node perturbation). For the ImageNet dataset we observed a classification error rate of 56.1 % on the top 5 predicted image classes with our algorithm, which is much better than the error rate achieved when keeping the feedback weights fixed, and much closer to that of full gradient descent (Fig. 6c). We also trained a network on the MNIST dataset, and achieved a similar performance of 1.1% error on the test set with all three algorithms (Fig. S7). Therefore, these results demonstrate that the ensemble-level burst-dependent learning rule, coupled with additional mechanisms to promote multiplexing, linearity and alignment, can solve difficult tasks.

Discussion

In this paper, we explored the possibility that high-frequency bursts in pyramidal neurons can provide an instructive signal for synaptic plasticity that can coordinate learning across hierarchical circuits. We have shown that a combination of a burst-dependent plasticity rule with STP and regenerative dendritic activity turns feedback connections into a teacher (Fig. 2), which by multiplexing (Fig. 3) can coordinate plasticity across multiple synaptic jumps (Fig. 4). We then showed that, with some additional burst-dependent learning at recurrent and feedback synapses, these mechanisms provide an approximation of a loss-function gradient for supervised learning (Fig. 5) and perform well on challenging image classification tasks (Fig. 6). Together, these results show that a local, spike-based learning rule that utilizes high-frequency bursts as an instructive signal can enable sophisticated credit assignment in hierarchical circuits.

It is important to note that there are a number of limitations to our model. First, our ensemble-level models utilized many "ensemble units" that incorporated the activity of many pyramidal neurons, which could potentially require disproportionately-sized networks. However, the functional impact of using many neurons in an ensemble is to provide a means for averaging the burst probabilities. Theoretically, this averaging could be done over time, rather than over neurons. If so, there is no reason that the algorithm could not work with single-neuron ensembles, though it would require a much longer time to achieve good estimates of the gradients. To some extent, this is simply the typical issue faced by any model of rate-based coding: if rates are used to communicate information then spatial or temporal averaging is required for high-fidelity communication. Furthermore, we suspect that allowing population coding could reduce the number of neurons required for a reliable representation [13].

Next, by focusing on learning, we ignored other ongoing cognitive processes. For instance, the close link between attention and credit assignment implies that the same mechanisms may serve both attention and learning purposes [60,80]. Although some experimental data points to a role of bursting in attention [91,96], further work is required to establish if burst coding can give rise to attention-like capabilities in neural networks.

The presence of the gating term, $M(t)$, may be seen as an additional limitation in the model, since it is left in an abstract form and not directly motivated by biology. This term was introduced in order to ensure that learning was driven by the teaching signal and not by changes in the stimuli. Of course, if the goal is not supervised learning, but unsupervised learning, then this term may be unnecessary. Indeed, one may view this as a prediction, i.e. that learning to match a target should involve an additional gating mechanism that is not required for unsupervised learning. We believe that this may be a fruitful

avenue for future research.

Another set of limitations derive from how we moved from detailed cellular-level simulations to abstract neural network models that were capable of solving complex tasks. For example, in moving to the abstract models, we gradually made a number of simplifying assumptions, including clear separation between bursts and single spikes, simplified STP, simplified bursting mechanisms, and ensemble-level units that represented spiking activity across multiple neurons with a single value. We highlight these limitations because it is important to keep them in mind when considering the potential for the cellular-level plasticity rule to implement sophisticated credit assignment. Ideally, we would have the computational resources to fully simulate many thousands of ensembles of pyramidal neurons and interneurons with complex synaptic dynamics and bursting in order to see if the cellular-level burst-dependent rule could also solve complicated tasks. However, these questions will have to be resolved by large-scale projects that can simulate millions of biophysically realistic neurons with complicated internal dynamics [42, 62]. Finally, our model did not include any sophisticated neuromodulatory systems. Yet, we know neuromodulation can regulate synaptic plasticity [86]. Future work could investigate how burst-dependent plasticity and neuromodulation could interact to improve learning further.

Our model makes a number of falsifiable experimental predictions that could be examined experimentally. First, the model predicts that there should be a polarization of STP along the sensory hierarchy, with bottom-up synaptic projections being largely STD and top-down synaptic projections being largely STF. There are reports of such differences in thalamocortical projections [38, 87], which suggests that an important missing component of our model is the inclusion of thalamic circuitry. There are also reports of polarization of STP along the basal dendrosomatic axis [39], and our model would predict that this polarization should extend to apical dendrites. Second, because our model proposes that burst firing carries information about errors, there should be a relationship between burst firing and progress in learning. Specifically, our model predicts that the *variance* in burst probabilities across a population should be correlated with the errors made during learning (Fig. S6). Experimental evidence in other systems supports this view [26, 45]. Finally, our model predicts that inhibition in the distal apical dendrites serves, in part, to homeostatically regulate burst probabilities to promote learning. Thus, a fairly simple prediction from the model is that manipulations of distal dendrite targeting interneurons, such as somatostatin positive interneurons, should lead to unusual levels of bursting in cortical circuits and disrupt learning. Some recent experimental evidence supports this prediction [20, 26].

Linking low-level and high-level computational models of learning is one of the major challenges in computational neuroscience. Our focus on supervised learning of static inputs was motivated by recent progress in this area. However, machine learning researchers have also been making rapid progress in unsupervised learning on temporal sequences in recent years [24, 41]. We suspect that many of the same mechanisms we explored here, e.g. burst-dependent plasticity, could also be adapted for unsupervised learning of temporal sequences in hierarchical circuits. Ultimately, by showing that a top-down orchestration of learning is a natural result of a small set of experimentally observed phenomena, our work opens the door to future approaches that utilize the unique physiology of cortical microcircuits to implement powerful learning algorithms using time-varying signals.

Methods

Spiking model

Spiking simulations were performed using the Auryn simulator [102], except for the pairing protocols of Fig. 2b-d, which used Python. The code is available at `https://github.com/apayeur/spikingburstprop`.

Event and burst detection

An *event* was said to occur either at the time of an isolated spike or at the time of the first spike in a burst. A *burst* was defined as any occurrence of at least two spikes with an interspike interval (ISI) less than the threshold $b_{\text{th}} = 16$ ms [7, 69]. Any additional spike with ISI $< b_{\text{th}}$ belonged to the same burst. A neuron i kept track of its time-averaged burst probability $\overline{P}_i$ by using exponential moving averages of its event train E_i and burst train B_i:

$$\overline{E}_i(t) = \frac{1}{\tau_{\text{avg}}} \int_0^\infty E_i(t-\tau) e^{-\tau/\tau_{\text{avg}}} d\tau \tag{3}$$

$$\overline{B}_i(t) = \frac{1}{\tau_{\text{avg}}} \int_0^\infty B_i(t-\tau) e^{-\tau/\tau_{\text{avg}}} d\tau \tag{4}$$

$$\overline{P}_i(t) = \frac{\overline{B}_i(t)}{\overline{E}_i(t)}, \tag{5}$$

where τ_{avg} is a slow time constant ($\sim$ 1-10 s). Also, $E_i(t) = \sum_{\text{event}} \delta(t - t_{i,\text{event}})$ and $B_i(t) = \sum_{\text{burst}} \delta(t - t_{i,\text{burst}})$, where $t_{i,\text{event}}$ and $t_{i,\text{burst}}$ indicate the timing of an event and of the second spike in a burst, respectively.

Plasticity rule

Weights were updated upon the detection of a postsynaptic event or burst according to

$$\frac{dw_{ij}}{dt} = \eta M \left\{ \left[(1 + H_i) B_i + (-\overline{P}_i + H_i) E_i \right] \widetilde{E}_j + G_i E_j \right\} \tag{6}$$

where $\widetilde{E}_j(t) = \int_0^\infty E_j(t-\tau) e^{-\tau/\tau_{\text{pre}}} d\tau$ is a presynaptic trace with time constant τ_{pre}. Here, τ_{pre} is typically much smaller than τ_{avg}, with $\tau_{\text{pre}} \sim 10$ ms, but it could possibly be made larger to accommodate plasticity rules with slower dynamics [12]. The prefactor M gates plasticity during training: in the XOR task (Fig. 4), $M = 1$ when the teaching signal is present and 0 otherwise. In Fig. 2, $M = 1$ throughout.

Homeostatic terms help to restrict the activity of neurons to an appropriate range. The homeostatic functions H_i and G_i were defined as

$$H_i(t) = -\left(1 - \frac{e_{\max}}{\overline{E}_i(t)}\right) \Theta(\overline{E}_i(t) - e_{\max}) \tag{7}$$

$$G_i(t) = \left[e_{\min} - \overline{E}_i(t) \right] \Theta(e_{\min} - \overline{E}_i(t)), \tag{8}$$

where $e_{\min}$ (resp. $e_{\max}$) is a minimum (resp. maximum) event rate, and $\Theta(\cdot)$ denotes the Heaviside step function. When the neuron-specific running average of the event rate, $\overline{E}_i(t)$, lies within these limits, $H_i = G_i = 0$, and we recover the learning rule of Eq. 1. In most simulations, network parameters were chosen in such a way that the homeostatic plasticity had little to no effect. Typically, we used $e_{\min} = 2$ Hz and $e_{\max} = 10$ Hz.

Pairing protocols

For all pairing protocols of Fig. 2b-d, we had $\tau_{\text{pre}} = 50$ ms, $\tau_{\text{avg}} = 15$ s, $\eta = 0.1$, and we set the homeostatic terms to zero.

- *Periodic protocol.* Five consecutive pairings were separated by a quiescent period of 10 s, 15 times. We used pairings with $\Delta t = 0$. For each pairing frequency the starting value for the estimated burst probability was $\overline{P}_i(t=0) = 0.15$ and $\overline{E}_i(t=0) = 5$ Hz.

- *Poisson protocol.* Both the pre and postsynaptic neurons fired spikes at a Poisson rate r with no refractory period. For each r, the induction lasted 60 s and we averaged over 10 independent realizations. We used $\overline{E}_i(t=0) = 5$ Hz.

- *Burst-Poisson protocol.* Both the pre and postsynaptic neurons produced events at a Poisson rate r, including a refractory period $\tau_{\text{ref}}^{(E)} > b_{\text{th}}$. For each event, a burst was generated with probability p and an intraburst ISI was sampled from $\text{Unif}(\tau_{\text{ref}}^{(B)}, \tau_{\text{ref}}^{(B)} + t_{\max})$, with $\tau_{\text{ref}}^{(B)} + t_{\max} < b_{\text{th}}$. For the simulations in Fig. 2d, we used $\tau_{\text{ref}}^{(E)} = 20$ ms, $\tau_{\text{ref}}^{(B)} = 4$ ms and $t_{\max} = 10$ ms. We set $\overline{P}_i(t=0) = 0.2$ and the event rate of the pre and postsynaptic neurons were set to $r = 5$ Hz and $r = 10$ Hz, with corresponding values for the initial postsynaptic event rate estimates. For each r, the induction lasted 100 s and we averaged over 20 independent realizations.

Neuron models

- *Pyramidal neurons* The somatic compartment obeyed

$$\begin{aligned} C_s \dot{V}_s &= -(C_s/\tau_s)(V_s - E_L) + g_s f(V_d) + I_s - w_s \\ \tau_{w_s} \dot{w}_s &= -w_s + b\tau_{w_s} S(t) \end{aligned} \tag{9}$$

  where V_s is the somatic membrane potential, w_s is an adaptation variable, I_s is the total current applied to the soma (includes noise and other synaptic inputs) and $S(t)$ is the spike train of the neuron. The function $f(V_d)$ in the equation for V_s takes into account the coupling with the dendritic compartment, with $f(V_d) = 1/\{1 + \exp[-(V_d - E_d)/D_d]\}$ and parameters $E_d = -38$ mV and $D_d = 6$ mV. A spike occurred whenever V_s crossed a moving threshold from below. The latter jumped up by 2 mV right after a spike and relaxed towards -50 mV with a time constant of 27 ms. Other somatic parameters were: $\tau_s = 16$ ms, $C_s = 370$ pF, $E_L = -70$ mV, $\tau_{w_s} = 100$ ms, $b = 200$ pA, and $g_s = 1300$ pA. The reset voltage after a spike was $V_r = -70$ mV.

  The dendritic compartment obeyed

$$\begin{aligned} C_d \dot{V}_d &= -(C_d/\tau_d)(V_d - E_L) + g_d f(V_d) + c_d (K * S)(t) + I_d - w_d \\ \tau_{w_d} \dot{w}_d &= -w_d + a_w (V_d - E_L). \end{aligned} \tag{10}$$

  The function $f(V_d)$ is the same as above and is responsible for the regenerative dendritic activity. The term $c_d(K * S)(t)$ represents the backpropagating action potential, with the kernel K modeled as a box filter of amplitude one and duration 2 ms, delayed by 0.5 ms with respect to the somatic spike. Other dendritic parameters were: $\tau_d = 7$ ms, $C_d = 170$ pF, $E_L = -70$ mV, $\tau_{w_d} = 30$ ms, $a = 13$ nS, and $g_d = 1200$ pA.

  This model and its parameters are described in more detail and compared with experimental data in Ref. [69].

- *Dendrite-targeting inhibition.* We modeled somatostatin-positive interneurons [46, 71, 92] using the adaptive exponential integrate-and-fire (AdEx) model [68]:

$$\begin{aligned} C\dot{V} &= -g_L(V - E_L) + g_L \Delta_T e^{\frac{V - V_T}{\Delta_T}} + I - w \\ \tau_w \dot{w} &= a(V - E_L) + b\tau_w S(t) - w \end{aligned}$$

  where I is the total current applied to the neuron. A spike occurred whenever V crossed $V_{\text{cut}} = 24$ mV and was followed by a refractory period τ_{ref}. Parameter values were $C = 100$ pF, $g_L = 5$ nS, $E_L = -70$ mV, $V_T = -62$ mV, $\Delta_T = 4$ mV, $\tau_w = 500$ ms, $a = 0.5$ nS, $b = 10$ pA, $V_r = -65$ mV and $\tau_{\text{ref}} = 2$ ms. In Fig. 3, these model neurons (grey squares in Fig. 3a) were receiving top-down excitation from higher-level pyramidal cells.

- *Perisomatic inhibition* We modeled parvalbumin-positive neurons [72] using the AdEx model with parameters chosen to reproduce qualitatively their typical fast-spiking phenotype. Parameter values were $C = 100$ pF, $g_L = 10$ nS, $E_L = -70$ mV, $V_T = -48$ mV, $\Delta_T = 2$ mV, $V_r = -55$ mV, $\tau_{\text{ref}} = 1$ ms and $a = b = 0$. In Fig. 3, these model neurons (grey discs in Fig. 3a) were receiving bottom-up excitation from the lower-level pyramidal cells.

Connectivity

In general, connections between distinct neural ensembles were sparse ($\sim 5\%$ connection probability). Pyramidal neurons within an ensemble had no recurrent connections between their somatic compartments. Within a pyramidal ensemble, burst-probability linearization was enacted by sparse STF inhibitory synapses onto the dendritic compartments (Fig. S1). These STF connections were not illustrated in Fig. 3a for clarity.

Synapses

All synapses were conductance-based. The excitatory (resp. inhibitory) reversal potential was $E = 0$ mV (resp. $E = -80$ mV) and the exponential decay time constant was 5 ms (resp. 10 ms). There were no NMDA components to excitatory synapses. For a given connection between two ensembles, existing synapses had their strengths all initialized to the same value.

Noise

Each neuron (for single-compartment neurons) and each compartment (for two-compartment neurons) received its own (private) noise in the form of a high-frequency excitatory Poisson input combined to an inhibitory Poisson input. The only exception was the noise applied to the neural populations in Fig. 2e-g, where we used sparse connections from a pool of excitatory and inhibitory Poisson neurons. Noise served to decorrelate neurons within a population and to imitate *in vivo* conditions.

Short-term plasticity

STP was modeled following the extended Markram-Tsodyks model [63]. Using the notation of Ref. [22], the parameters for STF were $D = 100$ ms, $F = 100$ ms, $U = 0.02$ and $f = 0.1$. For STD, the parameters were $D = 20$ ms, $F = 1$ s, $U = 0.9$ and $f = 0.1$. These sets of parameters were chosen following [69] to help decode bursts (using STF) and events (using STD).

Spiking XOR gate

A XOR gate maps binary inputs $(0, 0)$ and $(1, 1)$ onto 0 and inputs $(1, 0)$ and $(0, 1)$ onto 1. In the context of our spiking network, input 0 corresponded to a low event rate (~ 2 Hz) and input 1 to a higher event rate (~ 10 Hz). These were obtained by applying a hyperpolarizing (resp. depolarizing) current for 0 (resp. 1) to the corresponding input-layer population. Importantly, compared to the spiking simulations described above, our implementation of the spiking XOR gate used three simplifications to reduce the dimension of the parameter search space. First, events and bursts were propagated directly instead of relying on STP (see Fig. S2). Second, disynaptic inhibition was replaced by direct inhibition coming from the pyramidal cells. Third, we used a simplified pyramidal neuron model. Below, we describe this model, as well as the initialization of the network, the error generation and the learning protocol for the XOR gate.

- *Simplified pyramidal neuron model.* The effect of dendritic regenerative activity on the somatic compartment (controlled by g_s in Eqs. 9-10) was replaced by a conditional burst probability: whenever a somatic event occurred, a burst was produced with probability $f(V_d)$. This function is the same as that appearing in Eqs. 9-10, but with $E_d = -57$ mV. This model permitted a cleaner burst-detection process and burst-ensemble multiplexing.

- *Initialization of the network.* The feedforward synaptic strengths were initialized so that the event rates of all pyramidal ensembles in the network belonged to $[e_{\min}, e_{\max}]$ for all inputs. Excitatory synaptic strengths from the input layer to the hidden layer were all equal, and likewise for the inhibitory synapses. For the hidden-to-output feedforward connections, the ratio of the excitatory synaptic strengths was 1.4:1.05 in favor of hidden 1. This ratio for inhibition was 5:0.3 in favor of hidden 2. The feedback strengths from the output population to the hidden populations—the only existing ones—were initialized so that one of the hidden population (hidden pool 2 in Fig. 4) would receive inhibition on average, and the other excitation. A constant depolarizing current was applied to the hidden pool 2's dendritic compartments to compensate for this stronger inhibition.

- *Error generation.* At the output layer, we specified a maximum and a minimum event rate, $e_{\max}$ and $e_{\min}$ (the same as in the learning rule of Eq. 6). The following linearly transformed $\overline{E}_i$

$$\overline{E}_i' = \frac{\overline{E}_i - e_{\min}}{e_{\max} - e_{\min}}$$

  was then used in conjunction with a cross entropy loss function to compute the error for each neuron of the output population. As a result, a dendritic current, $I_i^{(d)}$, was injected into every neuron so that its burst probability would increase or decrease according to the running average of its event rate and the desired output:

$$\text{if desired output} = 0 \Rightarrow I_i^{(d)} = -c/(e_{\max} - \overline{E}_i)$$
$$\text{if desired output} = 1 \Rightarrow I_i^{(d)} = c/(\overline{E}_i - e_{\min}).$$

  where $c \sim 1$ nA $\cdot$ Hz. For instance, if the desired output was 0 and $\overline{E}_i$ was large, then the injected current was strongly hyperpolarizing. The injected current was set to zero when $\overline{E}_i$ was to within 1 Hz of the desired value.

- *Learning protocol.* A simulation proceeded as follows. With the plasticity off, there was first a relaxation interval of duration $3\tau_{\text{avg}}$, with no input applied to the network (for Fig. 4, $\tau_{\text{avg}} = 5$ s). Then, the four different input pairs were applied consecutively to give the "before learning" response in Fig. 4d. Afterward, the four input/output pairs were applied consecutively, for 20 s each (typically in the same order); namely one epoch. For each input/output pair, first, the input alone was applied to the input populations with the plasticity off. We let the network reach its steady state for that input for the first 90% of the duration of an example. During this prediction interval, the moving average of the burst probability would converge towards the actual burst probability of the population for that given input. During the last 10% of the example duration, the plasticity was activated for all feedforward excitatory synapses and the teacher was applied. For computational efficiency, the error was computed once, at the very end of the prediction interval. The total number of epochs required to reach decent performance depended on the initialization of the network and the learning rate; for Fig. 4, we used 500 epochs. At the end of learning, the plasticity was switched off for good and the "after learning" response was computed.

Deep network model for categorical learning

We now describe the deep network model that was used to learn the classification tasks reported in Figs. 5-6. The code is available at `https://github.com/jordan-g/Burstprop`. The model can be seen as a limiting case of a time-dependent rate model, which itself can be heuristically derived from the spiking network model under simplifying assumptions (see Supplemental Materials).

For the fully-connected layers in the network, we defined the "somatic potentials" of units in layer l as:

$$\mathbf{v}_l = \mathbf{W}_l \mathbf{e}_{l-1},$$

where $\mathbf{W}_l$ is the weight connecting layer $l-1$ to layer l. Note that in this formulation we include a bias term as a column of $\mathbf{W}_l$. The event rate of layer l was given by

$$\mathbf{e}_l = f_l\left(\mathbf{v}_l\right),$$

where f_l is the activation function for layer l. In models trained on MNIST and CIFAR-10, the activation function was a sigmoid. In the model trained on ImageNet, a ReLU activation was used for hidden layers and a softmax activation was used at the output layer.

During the feedforward pass, the burst probability at the output layer ($l = L$) was set to a constant, $p_L^{(0)}$ (in these experiments, this was set to 0.2). Our previous research [69] has shown that the dendritic transfer function is a sigmoidal function of its input (see also Fig. S4). Therefore, the hidden-layer burst probabilities, $\mathbf{p}_l$, for $l < L$, were computed using a sigmoidal function of a local "dendritic potential" $\mathbf{u}_l$ as

$$\mathbf{p}_l = \sigma(\beta \mathbf{u}_l + \alpha),$$

where α and β are constants controlling the dendritic transfer function. In our experiments, we set $\beta = 1$ and $\alpha = 0$. Figure S4 illustrates various mechanisms affecting these parameters. The dendritic potentials were given by

$$\mathbf{u}_l = h(\mathbf{e}_l) \odot \left(\mathbf{Y}_l \mathbf{b}_{l+1}\right), \tag{11}$$

where $\odot$ is the elementwise product. The vector-valued function $h(\mathbf{e}_l) \equiv f'(\mathbf{v}_l) \odot \mathbf{e}_l^{-1}$ depends on the chosen activation function; of course, some caution is required when ReLU and softmax activations are used (see Supplemental Materials). The burst rate is given by

$$\mathbf{b}_{l+1} = \mathbf{p}_{l+1} \odot \mathbf{e}_{l+1}. \tag{12}$$

Finally, $\mathbf{Y}_l$ is the feedback weight matrix. For the feedback alignment algorithm, $\mathbf{Y}_l$ is a random matrix and is fixed throughout learning [59]. In the standard backpropagation algorithm, the feedforward and feedback weight matrices are symmetric so that $\mathbf{Y}_l = \mathbf{W}_{l+1}^T$, where ^{T} denotes the transpose. Below, we also describe how to learn the feedback weights to make them symmetric with the feedforward weights using the Kolen-Pollack algorithm [2].

With the teacher present, the output-layer burst probabilities were set to a squashed version of $p_L^{(0)} - h(\mathbf{e}_l) \odot \nabla_{\mathbf{e}_L} \mathcal{L}$, where $\mathcal{L}$ is the loss function (a mean squared error loss for Figs. 5-6). The squashing function was to make sure that $p_{L,i} \in [0, 1]$. The Supplemental Materials provide a few examples of squashing functions. The burst probabilities of the hidden layers were then computed as above. Finally, the weights were updated according to

$$\Delta \mathbf{W}_l = \eta_l \big((\mathbf{p}_l - \overline{\mathbf{p}}_l) \odot \mathbf{e}_l\big) \mathbf{e}_{l-1}^T - \lambda \mathbf{W}_l, \tag{13}$$

where $\mathbf{p}_l$ and $\overline{\mathbf{p}}_l$ denote the burst probabilities with and without teacher, respectively, η_l is the learning rate hyperparameter for units in layer l, and λ is a weight decay hyperparameter. Note that, for this model, $\overline{\mathbf{e}}_l$ lags $\mathbf{e}_l$ by a single computational step (see Supplemental Materials). Therefore, when the teacher appears, $\overline{\mathbf{e}}_l = \mathbf{e}_l$ and we can write

$$(\mathbf{p}_l - \overline{\mathbf{p}}_l) \odot \mathbf{e}_l = \mathbf{b}_l - \overline{\mathbf{b}}_l.$$

This means that, in this model, the error is directly represented by the deviation of the burst rate with respect to a reference.

In the case of convolutional layers, the event rates of ensembles in layer l were given by

$$\mathbf{e}_l = f_l(\mathbf{W}_l * \mathbf{e}_{l-1}),$$

where $*$ represents convolution. Similarly, the dendritic potentials in layer l were given by $\mathbf{u}_l = \mathbf{Y}_l * \mathbf{b}_{l+1}$ while burst probabilities were calculated as in the fully-connected layers. Finally, the weights of convolutional layers were updated as

$$\Delta \mathbf{W}_l = \eta_l \psi\big(\mathbf{b}_l - \overline{\mathbf{b}}_l, \mathbf{e}_{l-1}\big) - \lambda \mathbf{W}_l, \tag{14}$$

where ψ combines the burst deviations and event rates to compute an approximation of the gradient with respect to the convolutional weights $\mathbf{W}_l$.

Learning the recurrent weights

In certain experiments, we introduced recurrent inputs into the hidden layers that served to keep burst probabilities in the linear regime of the sigmoid function. At layer l, we set the reference dendritic potentials to

$$\overline{\mathbf{u}}_l = h(\mathbf{e}_l) \odot \mathbf{Y}_l \overline{\mathbf{b}}_{l+1} - \mathbf{Z}_l \overline{\mathbf{b}}_l, \tag{15}$$

where $\mathbf{Z}_l$ is the recurrent weight matrix and the burst rates used here, in bold sans-serif, are calculated as the burst rate *without* any recurrent inputs and *without* the teaching signal:

$$\overline{\mathbf{b}}_l = \sigma\left(\beta h(\mathbf{e}_l) \odot \mathbf{Y}_l \overline{\mathbf{b}}_{l+1} + \alpha\right) \odot \mathbf{e}_l. \tag{16}$$

Otherwise, the dendritic potentials and burst rates must be solved self-consistently, slowing down computations. Recurrent weights are then updated in order to minimize $\overline{\mathbf{u}}_l$:

$$\Delta \mathbf{Z}_l = -\eta_r \overline{\mathbf{u}}_l \overline{\mathbf{b}}_l^T, \tag{17}$$

where η_r is the learning rate. Note that, with these recurrent inputs, the updates of matrix $\mathbf{W}_l$ are the same as before, but now with

$$\mathbf{p}_l = \sigma[\beta\big(h(\mathbf{e}_l) \odot \mathbf{Y}_l \mathbf{b}_{l+1} - \mathbf{Z}_l \overline{\mathbf{b}}_l\big) + \alpha].$$

.

Learning the feedback weights

Kolen and Pollack [52] found that if the feedforward and feedback weights are updated such that

$$\Delta \mathbf{W}_l = \eta \mathbf{A} - \lambda \mathbf{W}_l$$
$$\Delta \mathbf{Y}_l = \eta \mathbf{A} - \lambda \mathbf{Y}_l,$$

where $\mathbf{A}$ is any matrix with the same shape as $\mathbf{W}_l$ and $\mathbf{Y}_l$, then $\mathbf{Y}_l$ and $\mathbf{W}_l$ will converge. This means that if the feedback weights are updated in the same direction as the feedforward weights and weight decay is applied to both sets of weights, they will eventually become symmetric. Thus, we implemented the following learning rule for the feedback weights between layer $l+1$ and layer l:

$$\Delta \mathbf{Y}_l = \eta_l \left(\mathbf{b}_{l+1} - \overline{\mathbf{b}}_{l+1}\right) \mathbf{e}_l^T - \lambda \mathbf{Y}_l, \tag{18}$$

where λ is a weight decay hyperparameter. In convolutional layers, we used the following weight update:

$$\Delta \mathbf{Y}_l = \eta_l \psi\big(\mathbf{b}_{l+1} - \overline{\mathbf{b}}_{l+1}, \mathbf{e}_l\big) - \lambda \mathbf{Y}_l. \tag{19}$$

Training the model with CIFAR-10 and ImageNet

The network architectures described in Table S2 of the Supplemental Materials were trained on standard image classification datasets, CIFAR-10 [54] and ImageNet [23]. The CIFAR-10 dataset consists of 60,000 32×32 px training images belonging to 10 classes, while the ImageNet dataset consists of 1.2 million images (resized to 224×224 px) split among 1000 classes.

Each unit in these networks represents an ensemble of pyramidal neurons and has an event rate, burst probability, and burst rate. For each training example, the input image is presented and a forward pass is done, where event rates $\mathbf{e}_l$ throughout the network are computed sequentially, followed by a feedback pass where burst probabilities $\overline{\mathbf{p}}_l$ and burst rates $\overline{\mathbf{b}}_l$ are computed. Then, the teaching signal is shown at the output layer, and new burst probabilities $\mathbf{p}_l$ and burst rates $\mathbf{b}_l$ are computed backward through

the network. Weights are then updated using our weight update rules. Networks were trained using stochastic gradient descent (SGD) with mini-batches, momentum and weight decay. ReLU layers were initialized from a normal distribution using Kaiming initialization [43], whereas Xavier initialization was used in sigmoid layers [37]. Hyperparameter optimization was done on all networks using validation data (see Supplemental Materials for details).

Training the model using node perturbation

Node perturbation is a technique that approximates gradient descent by randomly perturbing the activations of units in the network, and updating weights according to the change in the loss function [94, 95]. In the model trained using node perturbation, at each step, first the input is propagated through the network as usual, after which the global loss, L, is recorded. Then, the same input is propagated again through the network but the activations of units in a single layer l are randomly perturbed:

$$\mathbf{e}_l = f_l(\mathbf{W}_l * \mathbf{e}_{l-1} + \boldsymbol{\xi}_l), \tag{20}$$

where the elements of $\boldsymbol{\xi}_l$ are chosen from a normal distribution with mean 0 and standard deviation σ. The new loss, L_{NP}, is recorded. The weights in layer l are then updated using the following weight update rule:

$$\Delta \mathbf{W}_l = \eta_l \big((L_{\mathrm{NP}} - L) \boldsymbol{\xi}_l / \sigma^2 \big) \mathbf{e}_{l-1}^T. \tag{21}$$

The layer to be perturbed, l, is changed with each mini-batch by iterating bottom-up through all of the layers in the network.

Acknowledgments

We thank Adam Santoro and Leonard Maler for comments on this manuscript. We also thank Markus Hilscher and Maximiliano José Nigro for sharing data about SOM neurons. This work was supported by two NSERC Discovery Grants, 06872 (RN) and 04947 (BAR), a CIHR Project Grant (RN383647 - 418955), and a Fellowship from the CIFAR Learning in Machines and Brains Program (BAR).

Author contributions

All authors contributed to the burst-dependent learning rule. AP performed the spiking simulations. JG designed the recurrent plasticity rule and performed the numerical experiments on CIFAR-10 and ImageNet. BAR and RN wrote the manuscript, with contributions from JG and AP. BAR and RN co-supervised the project.

References

[1] T. Akam and D. M. Kullmann. Oscillatory multiplexing of population codes for selective communication in the mammalian brain. *Nat. Rev. Neurosci.*, 15(2):111, 2014.

[2] M. Akrout, C. Wilson, P. C. Humphreys, T. Lillicrap, and D. Tweed. Using weight mirrors to improve feedback alignment. *arXiv preprint arXiv:1904.05391*, 2019.

[3] Y. Aljadeff, J. D'Amour, R. E. Field, R. C. Froemke, and C. Clopath. Cortical credit assignment by hebbian, neuromodulatory and inhibitory plasticity. *arxiv*, 2019.

[4] Y. Amit. Deep learning with asymmetric connections and hebbian updates. *Frontiers in computational neuroscience*, 13:18, 2019.

[5] A. Artola, S. Bröcher, and W. Singer. Different voltage dependent thresholds for inducing long-term depression and long-term potentiation in slices of rat visual cortex. *Nature*, 347:69–72, 1990.

[6] S. Bartunov, A. Santoro, B. Richards, L. Marris, G. E. Hinton, and T. Lillicrap. Assessing the scalability of biologically-motivated deep learning algorithms and architectures. In *Advances in Neural Information Processing Systems*, pages 9368–9378, 2018.

[7] J. Bastian and J. Nguyenkim. Dendritic modulation of burst-like firing in sensory neurons. *J. Neurophys.*, 85(1):10–22, 2001.

[8] C. C. Bell, A. Caputi, K. Grant, and J. Serrier. Storage of a sensory pattern by anti-hebbian synaptic plasticity in an electric fish. *Proceedings of the National Academy of Sciences*, 90(10):4650–4654, 1993.

[9] G. Bellec, F. Scherr, A. Subramoney, E. Hajek, D. Salaj, R. Legenstein, and W. Maass. A solution to the learning dilemma for recurrent networks of spiking neurons. *bioRxiv*, page 738385, 2019.

[10] G. Bi and M. Poo. Synaptic Modifications in Cultured Hippocampal Neurons: Dependence on Spike Timing, Synaptic Strength, and Postsynaptic Cell Type. *Journal of Neuroscience*, 18(24):10464, 1998.

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

[12] K. C. Bittner, A. D. Milstein, C. Grienberger, S. Romani, and J. C. Magee. Behavioral time scale synaptic plasticity underlies ca1 place fields. *Science*, 357(6355):1033–1036, 2017.

[13] M. Boerlin, C. K. Machens, and S. Denève. Predictive coding of dynamical variables in balanced spiking networks. *PLoS computational biology*, 9(11):e1003258, 2013.

[14] K. Bol, G. Marsat, E. Harvey-Girard, A. Longtin, and L. Maler. Frequency-tuned cerebellar channels and burst-induced ltd lead to the cancellation of redundant sensory inputs. *Journal of Neuroscience*, 31(30):11028–11038, 2011.

[15] G. Bouvier, D. Higgins, M. Spolidoro, D. Carrel, B. Mathieu, C. Léna, S. Dieudonné, B. Barbour, N. Brunel, and M. Casado. Burst-dependent bidirectional plasticity in the cerebellum is driven by presynaptic nmda receptors. *Cell Reports*, 15(1):104–116, 2016.

[16] F. Brandalise and U. Gerber. Mossy fiber-evoked subthreshold responses induce timing-dependent plasticity at hippocampal ca3 recurrent synapses. *Proceedings of the National Academy of Sciences*, 111(11):4303–4308, 2014.

[17] K. S. Burbank. Mirrored stdp implements autoencoder learning in a network of spiking neurons. *PLoS computational biology*, 11(12):e1004566, 2015.

[18] K. S. Burbank and G. Kreiman. Depression-biased reverse plasticity rule is required for stable learning at top-down connections. *PLoS computational biology*, 8(3):e1002393, 2012.

[19] L. Cauller. Layer i of primary sensory neocortex: where top-down converges upon bottom-up. *Behavioural brain research*, 71(1):163–170, 1995.

[20] S. X. Chen, A. N. Kim, A. J. Peters, and T. Komiyama. Subtype-specific plasticity of inhibitory circuits in motor cortex during motor learning. *Nature neuroscience*, 18(8):1109–1115, 2015.

[21] C. Clopath, L. Büsing, E. Vasilaki, and W. Gerstner. Connectivity reflects coding: a model of voltage-based stdp with homeostasis. *Nature neuroscience*, 13(3):344–52, Mar 2010.

[22] R. P. Costa, P. J. Sjöström, and M. C. Van Rossum. Probabilistic inference of short-term synaptic plasticity in neocortical microcircuits. *Front. Comput. Neurosci.*, 7, 2013.

[23] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*, 2009.

[24] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[25] B. Doiron, A. Litwin-Kumar, R. Rosenbaum, G. K. Ocker, and K. Josić. The mechanics of state-dependent neural correlations. *Nature neuroscience*, 19(3):383, 2016.

[26] G. Doron, J. N. Shin, N. Takahashi, C. Bocklisch, S. Skenderi, M. Drueke, L. de Mont, M. Toumazo, M. von Heimendahl, M. Brecht, et al. Perirhinal input to neocortical layer 1 controls learning. *bioRxiv*, page 713883, 2019.

[27] O. D. Faugeras, J. D. Touboul, and B. Cessac. A constructive mean-field analysis of multi population neural networks with random synaptic weights and stochastic inputs. *Frontiers in computational neuroscience*, 3:1, 2009.

[28] D. Feldman. Timing-based ltp and ltd and vertical inputs to layer ii/iii pyramidal cells in rat barrel cortex. *Neuron*, 27:45–56, 2000.

[29] D. Felleman and D. van Essen. Distributed hierarchical processing in the primate cerebral cortex. *Cereb. Cortex*, 1:1–47, 1991.

[30] L. N. Fletcher and S. R. Williams. Neocortical topology governs the dendritic integrative capacity of layer 5 pyramidal neurons. *Neuron*, 101(1):76–90, 2019.

[31] N. Frémaux, H. Sprekeler, and W. Gerstner. Functional requirements for reward-modulated spike-timing-dependent plasticity. *The Journal of Neuroscience*, 30(40):13326–13337, 2010.

[32] J. Friedrich and M. Lengyel. Goal-directed decision making with spiking neurons. *Journal of Neuroscience*, 36(5):1529–1546, 2016.

[33] R. C. Froemke, I. A. Tsay, M. Raad, J. D. Long, and Y. Dan. Contribution of individual spikes in burst-induced long-term synaptic modification. *Journal of Neurophysiology*, 95(3):1620–1629, 2006.

[34] F. Gambino, S. Pagès, V. Kehayas, D. Baptista, R. Tatti, A. Carleton, and A. Holtmaat. Sensory-evoked ltp driven by dendritic plateau potentials in vivo. *Nature*, 515(7525):116, 2014.

[35] W. Gerstner, R. Kempter, J. van Hemmen, and H. Wagner. A neuronal learning rule for sub-millisecond temporal coding. *Nature*, 383(6595):76–78, 1996.

[36] W. Gerstner, M. Lehmann, V. Liakoni, D. Corneil, and J. Brea. Eligibility traces and plasticity on behavioral time scales: experimental support of neohebbian three-factor learning rules. *Frontiers in neural circuits*, 12, 2018.

[37] X. Glorot and Y. Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 249–256, 2010.

[38] B. Granseth, E. Ahlstrand, and S. Lindström. Paired pulse facilitation of corticogeniculate epscs in the dorsal lateral geniculate nucleus of the rat investigated in vitro. *The Journal of physiology*, 544(2):477–486, 2002.

[39] F. W. Grillo, G. Neves, A. Walker, G. Vizcay-Barrena, R. A. Fleck, T. Branco, and J. Burrone. A distance-dependent distribution of presynaptic boutons tunes frequency-dependent dendritic integration. *Neuron*, 99(2):275–282, 2018.

[40] J. Guerguiev, T. P. Lillicrap, and B. A. Richards. Towards deep learning with segregated dendrites. *eLife*, 6:e22901, 2017.

[41] T. Han, W. Xie, and A. Zisserman. Video representation learning by dense predictive coding. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 0–0, 2019.

[42] M. Hawrylycz, C. Anastassiou, A. Arkhipov, J. Berg, M. Buice, N. Cain, N. W. Gouwens, S. Gratiy, R. Iyer, J. H. Lee, et al. Inferring cortical function in the mouse visual system through large-scale systems neuroscience. *Proceedings of the National Academy of Sciences*, 113(27):7337–7344, 2016.

[43] K. He, X. Zhang, S. Ren, and J. Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*, pages 1026–1034, 2015.

[44] D. O. Hebb. *The Organization of Behavior*. Wiley, New York, 1949.

[45] D. J. Herzfeld, Y. Kojima, R. Soetedjo, and R. Shadmehr. Encoding of action by the purkinje cells of the cerebellum. *Nature*, 526(7573):439, 2015.

[46] M. M. Hilscher, R. N. Leão, S. J. Edwards, K. E. Leão, and K. Kullander. Chrna2-martinotti cells synchronize layer 5 type a pyramidal cells via rebound excitation. *PLOS Biology*, 15(2):1–26, 02 2017.

[47] D. Huh and T. J. Sejnowski. Gradient descent for spiking neural networks. In *Advances in Neural Information Processing Systems*, pages 1433–1443, 2018.

[48] E. Izhikevich. Solving the distal reward problem through linkage of STDP and dopamine signaling. *Cerebral Cortex*, 17:2443–2452, 2007.

[49] B. Kampa, J. Letzkus, and G. Stuart. Requirement of dendritic calcium spikes for induction of spike-timing-dependent synaptic plasticity. *J Physiol*, Jan 2006.

[50] B. M. Kampa and G. J. Stuart. Calcium spikes in basal dendrites of layer 5 pyramidal neurons during action potential bursts. *J Neurosci*, 26(28):7424–32, Jul 2006.

[51] C. Kayser, M. Montemurro, N. Logothetis, and S. Panzeri. Spike-phase coding boosts and stabilizes information carried by spatial and temporal spike patterns. *Neuron*, 61(4):597–608, 2009.

[52] J. F. Kolen and J. B. Pollack. Backpropagation without weight transport. In *Proceedings of 1994 IEEE International Conference on Neural Networks (ICNN'94)*, volume 3, pages 1375–1380. IEEE, 1994.

[53] K. P. Körding and P. König. Supervised and unsupervised learning with two sites of synaptic integration. *J Comput. Neurosci.*, 11(3):207–215, 2001.

[54] A. Krizhevsky, V. Nair, and G. Hinton. Cifar-10 (canadian institute for advanced research). Technical report, University of Toronto, 2009.

[55] M. Larkum, J. Zhu, and B. Sakmann. A new cellular mechanism for coupling inputs arriving at different cortical layers. *Nature*, 398:338–341, 1999.

[56] M. E. Larkum and J. J. Zhu. Signaling of layer 1 and whisker-evoked ca2+ and na+ action potentials in distal and terminal dendrites of rat neocortical pyramidal neurons in vitro and in vivo. *Journal of neuroscience*, 22(16):6991–7005, 2002.

[57] R. Legenstein, D. Pecevski, and W. Maass. A learning theory for reward-modulated spike-timing-dependent plasticity with application to biofeedback. *PLOS Comput. Biol.*, 4:e1000180, 2008.

[58] J. J. Letzkus, B. M. Kampa, and G. J. Stuart. Learning rules for spike timing-dependent plasticity depend on dendritic synapse location. *J. Neurosci.*, 26(41):10420–10429, Oct. 2006.

[59] T. P. Lillicrap, D. Cownden, D. B. Tweed, and C. J. Akerman. Random synaptic feedback weights support error backpropagation for deep learning. *Nature Commun.*, 7, 2016.

[60] G. W. Lindsay and K. D. Miller. How biological attention mechanisms improve task performance in a large-scale visual system model. *eLife*, 7:e38105, 2018.

[61] H. Markram, J. Lübke, M. Frotscher, and B. Sakmann. Regulation of synaptic efficacy by coincidence of postsynaptic APs and EPSPs. *Science*, 275:213–215, 1997.

[62] H. Markram, E. Muller, S. Ramaswamy, M. W. Reimann, M. Abdellah, C. A. Sanchez, A. Ailamaki, L. Alonso-Nanclares, N. Antille, S. Arsever, et al. Reconstruction and simulation of neocortical microcircuitry. *Cell*, 163(2):456–492, 2015.

[63] H. Markram, Y. Wang, and M. Tsodyks. Differential signaling via the same axon of neocortical pyramidal neurons. *Proc. Natl. Acad. Sci. USA*, 95(9):5323–5328, 1998.

[64] R. M. Meredith, A. M. Floyer-Lea, and O. Paulsen. Maturation of long-term potentiation induction rules in rodent hippocampus: role of gabaergic inhibition. *J Neurosci*, 23(35):11142–11146, Dec 2003.

[65] T. Miconi. Biologically plausible learning in recurrent neural networks reproduces neural dynamics observed during cognitive tasks. *Elife*, 6:e20899, 2017.

[66] S. Z. Muller, A. Zadina, L. Abbott, and N. Sawtell. Continual learning in a multi-layer network of an electric fish. *Cell*, 2019.

[67] M. Murayama, E. Pérez-Garci, T. Nevian, T. Bock, W. Senn, and M. E. Larkum. Dendritic encoding of sensory stimuli controlled by deep cortical interneurons. *Nature*, 457(7233):1137–1141, 2009.

[68] R. Naud, N. Marcille, C. Clopath, and W. Gerstner. Firing patterns in the adaptive exponential integrate-and-fire model. *Biol Cybern*, 99:335–347, 2008.

[69] R. Naud and H. Sprekeler. Sparse bursts optimize information transmission in a multiplexed neural code. *Proceedings of the National Academy of Sciences*, 2018.

[70] T. Nevian and B. Sakmann. Spine ca2+ signaling in spike-timing-dependent plasticity. *J. Neurosci.*, 26(43):11001–11013, 2006.

[71] M. J. Nigro, Y. Hashikawa-Yamasaki, and B. Rudy. Diversity and connectivity of layer 5 somatostatin-expressing interneurons in the mouse barrel cortex. *Journal of Neuroscience*, 38(7):1622–1633, 2018.

[72] A. M. Packer and R. Yuste. Dense, unspecific connectivity of neocortical parvalbumin-positive interneurons: a canonical microcircuit for inhibition? *Journal of Neuroscience*, 31(37):13260–13271, 2011.

[73] V. Parikh, R. Kozak, V. Martinez, and M. Sarter. Prefrontal acetylcholine release controls cue detection on multiple timescales. *Neuron*, 56(1):141–154, 2007.

[74] J. Pfister and W. Gerstner. Triplets of spikes in a model of spike timing-dependent plasticity. *Journal of Neuroscience*, Jan 2006.

[75] A. Renart, N. Brunel, and X.-J. Wang. Mean-field theory of irregularly spiking neuronal populations and working memory in recurrent cortical networks. *Computational neuroscience: A comprehensive approach*, pages 431–490, 2004.

[76] A. Reyes, R. Lujan, A. Rozov, N. Burnashev, P. Somogyi, and B. Sakmann. Target-cell-specific facilitation and depression in neocortical circuits. *Nat. Neurosci.*, 1(4):279–285, 1998.

[77] B. A. Richards, T. Lillicrap, P. Beaudoin, Y. Bengio, R. Bogacz, A. Christensen, C. Clopath, A. De Berker, S. Ganguli, C. Gillon, A. Hafner, Danijar Kepecs, N. Kriegeskorte, P. Latham, G. Lindsay, K. Miller, R. Naud, C. Pack, P. Poirazi, R. P. Costa, P. Roelfsema, J. Sacramento, A. Saxe, A. Schapiro, W. Senn, G. Wayne, D. Yamins, F. Zenke, D. Zylberberg, Joel Therien, and K. Kording. A deep learning framework for systems neuroscience. *Nat Neurosci*, 2019.

[78] B. A. Richards and T. P. Lillicrap. Dendritic solutions to the credit assignment problem. *Current opinion in neurobiology*, 54:28–36, 2019.

[79] P. R. Roelfsema and A. Holtmaat. Control of synaptic plasticity in deep cortical networks. *Nature Reviews Neuroscience*, 19(3):166, 2018.

[80] P. R. Roelfsema and A. v. Ooyen. Attention-gated reinforcement learning of internal representations for classification. *Neural computation*, 17(10):2176–2214, 2005.

[81] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Learning representations by back-popagating errors. *Nature*, 323:533–536, 1986.

[82] J. Sacramento, R. P. Costa, Y. Bengio, and W. Senn. Dendritic cortical microcircuits approximate the backpropagation algorithm. In *Advances in Neural Information Processing Systems*, pages 8721–8732, 2018.

[83] A. Samadi, T. P. Lillicrap, and D. B. Tweed. Deep learning with dynamic spiking neurons and fixed feedback weights. *Neural computation*, 29(3):578–602, 2017.

[84] B. Scellier and Y. Bengio. Towards a biologically plausible backprop. *arXiv preprint arXiv:1602.05179*, 2016.

[85] T. Schwalger, M. Deger, and W. Gerstner. Towards a theory of cortical columns: From spiking neurons to interacting neural populations of finite size. *PLoS computational biology*, 13(4):e1005507, 2017.

[86] G. H. Seol, J. Ziburkus, S. Huang, L. Song, I. T. Kim, K. Takamiya, R. L. Huganir, H.-K. Lee, and A. Kirkwood. Neuromodulators control the polarity of spike-timing-dependent synaptic plasticity. *Neuron*, 55(6):919–929, 2007.

[87] S. M. Sherman. Thalamocortical interactions. *Current opinion in neurobiology*, 22(4):575–579, 2012.

[88] P. J. Sjöström and M. Häusser. A cooperative switch determines the sign of synaptic plasticity in distal dendrites of neocortical pyramidal neurons. *Neuron*, 51(2):227–238, 2006.

[89] P. J. Sjöström, G. G. Turrigiano, and S. B. Nelson. Rate, timing, and cooperativity jointly determine cortical synaptic plasticity. *Neuron*, 32(6):1149–1164, 2001.

[90] H. F. Song, G. R. Yang, and X.-J. Wang. Reward-based training of recurrent neural networks for cognitive and value-based tasks. *Elife*, 6:e21492, 2017.

[91] N. Takahashi, T. G. Oertner, P. Hegemann, and M. E. Larkum. Active cortical dendrites modulate perception. *Science*, 354(6319):1587–1590, 2016.

[92] R. Tremblay, S. Lee, and B. Rudy. Gabaergic interneurons in the neocortex: from cellular properties to circuits. *Neuron*, 91(2):260–292, 2016.

[93] Y. Wang, A. Gupta, M. Toledo-Rodriguez, C. Wu, and H. Markram. Anatomical, physiological, molecular and circuit properties of nest basket cells in the developing somatosensory cortex. *Cerebral Cortex*, 12:395–410, 2002.

[94] J. Werfel, X. Xie, and H. S. Seung. Learning curves for stochastic gradient descent in linear feedforward networks. In *Advances in neural information processing systems*, pages 1197–1204, 2004.

[95] R. J. Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3-4):229–256, 1992.

[96] T. Womelsdorf, S. Ardid, S. Everling, and T. A. Valiante. Burst firing synchronizes prefrontal and anterior cingulate cortex during attentional control. *Current Biology*, 24(22):2613–2621, 2014.

[97] X. Xie and H. S. Seung. Equivalence of backpropagation and contrastive hebbian learning in a layered network. *Neural Comput.*, 15(2):441–454, 2003.

[98] N.-l. Xu, M. T. Harnett, S. R. Williams, D. Huber, D. H. O?Connor, K. Svoboda, and J. C. Magee. Nonlinear dendritic integration of sensory and motor input during an active sensing task. *Nature*, 492(7428):247–251, 2012.

[99] Y. Yang and S. G. Lisberger. Purkinje-cell plasticity and cerebellar motor learning are graded by complex-spike duration. *Nature*, 510(7506):529, 2014.

[100] A. V. Zaitsev and R. Anwyl. Inhibition of the slow afterhyperpolarization restores the classical spike timing-dependent plasticity rule obeyed in layer 2/3 pyramidal cells of the prefrontal cortex. *Journal of neurophysiology*, 107(1):205–215, 2011.

[101] F. Zenke and S. Ganguli. Superspike: Supervised learning in multilayer spiking neural networks. *Neural computation*, 30(6):1514–1541, 2018.

[102] F. Zenke and W. Gerstner. Limits to high-speed simulations of spiking neural networks using general-purpose computers. *Frontiers in neuroinformatics*, 8:76, 2014.