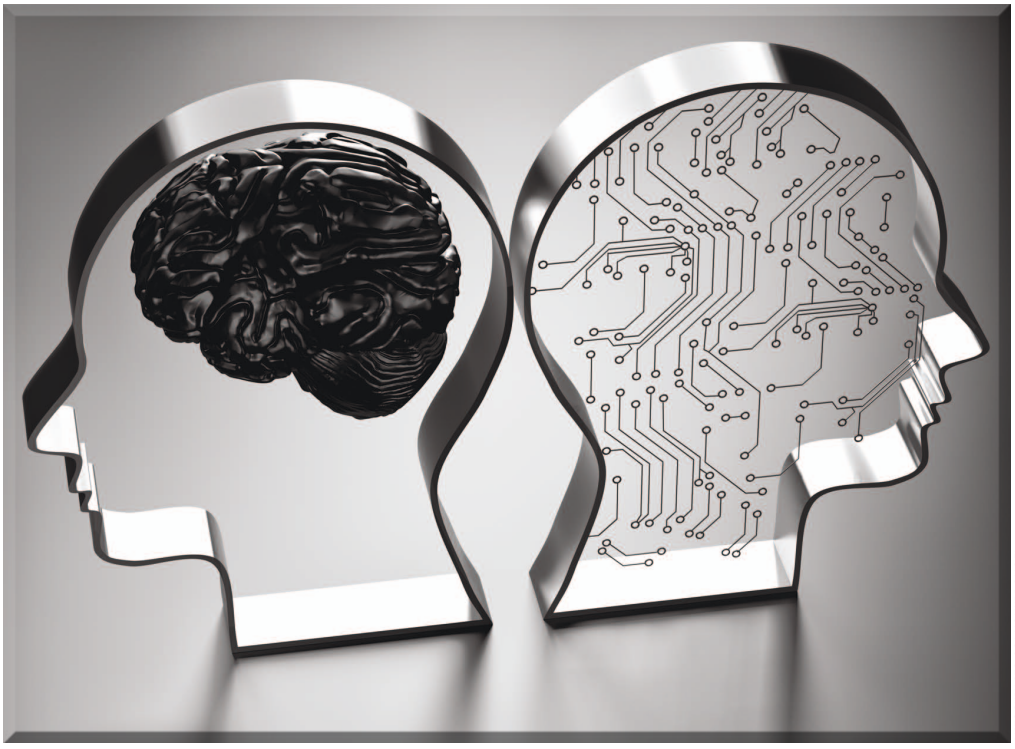
Building Brain-Inspired Computing Systems



©ISTOCKPHOTO.COM/ONURDONGEL

Examining the role of nanoscale devices.

BRAIN-INSPIRED COMPUTING IS attracting considerable attention because of its potential to solve a wide variety of data-intensive problems that are difficult for even state-of-the-art supercomputers to tackle. The ability of the human brain to process visual and audio inputs in real time and make complex logical decisions by consuming a mere 20 W makes it the most power-efficient computational engine known to man. While state-of-the-art digital complimentary metal–oxide–semiconductor (CMOS) technology permits the realization of individual devices and circuits that mimic the dynamics of neurons and synapses in the brain, emulating the immense parallelism and event-driven computational architecture in systems with comparable complexity and power budget as the brain, and in real time, remains a formidable challenge.

In the past decade, machine learning algorithms inspired by the brain's capability to learn and adapt based on the information it receives have made significant strides in achieving superhuman performance for several benchmark pattern recognition and analysis tasks [1]. These algorithms have caused a paradigm

S.R. NANDAKUMAR, SHRUTI R. KULKARNI, ANAKHA V. BABU, AND BIPIN RAJENDRAN

Digital Object Identifier 10.1109/MNANO.2018.2845078
Date of publication: 16 July 2018

Machine-learning algorithms range from simple linear regression models to multilayered deep neural networks.

shift from the static stored program algorithmic approach to a more data-driven adaptive model development approach to make decisions or predictions. Based on the underlying statistical relationships of the observed data, these models adapt to make more accurate predictions.

Machine-learning algorithms range from simple linear regression models to multilayered deep neural networks (DNNs). DNNs are a class of artificial neural networks (ANNs) that have achieved considerable success in recent years due to the development of efficient training algorithms, improved computational capabilities, and access to vast troves of training data. Such DNNs mimic the high-level organizational architecture of the brain because the processing units (neurons) are stacked in layers, with adjacent layers interconnected via adjustable weights (synapses). Each neuron receives a weighted sum of outputs from a subset of neurons in the previous layer and creates an output based on a nonlinear transformation. The weights of the network are trained to perform specific tasks based on the input data in a supervised or unsupervised manner.

With unsupervised learning, the data fed to the network has no labels and is used to extract general features from the data. In supervised learning, the network is trained with a labeled set of training data and the mismatch between network response and the label is used to determine a weight update that will minimize the error. Stochastic gradient descent (SGD)-based back-propagation algorithms [2] are commonly used for supervised training of multilayer (deep) neural network architectures. The multilayer structure combined with the nonlinear processing of neurons enables DNNs to tackle complex classification problems. Typical artificial neurons use differentiable nonlinearities for the ease of back-propagation-based weight update determination.

However, the nonlinear dynamics of neurons in the human brain are more complex. In a simplified picture, each neuron integrates the current it receives via the receptors on its dendrites, causing its membrane potential to rise above the resting potential. When the potential exceeds a threshold, an action potential, or spike, is issued, which propagates along the axon of the neuron. The axons are connected to the downstream neurons via synaptic junctions; the spikes will then induce currents proportional to the synaptic strength in the postsynaptic neurons. Each neuron in the human neocortex receives input spikes from approximately 10^4 other neurons, with each neuron spiking at a sparse rate between 0.1 and 100 Hz [3], [4]. This parallelism and sparse activity combined with the temporal integration property is believed to make the brain a power-efficient and error-tolerant decision maker. Artificial spiking neural networks (SNNs) attempt to mimic the previously mentioned features of the brain such as spike-based data encoding, event-triggered processing, and temporal processing of data to realize energy-efficient learning networks [5].

A key requirement of brain-inspired neural networks is the ability to process several streams of data and its features in parallel. Studies indicate that there is a direct correlation between the computational capabilities of these networks and their size (depth), and the amount of data used to train them [6], [7]. As a result, neural network training is computationally intensive and consumes huge amounts of time and energy. Furthermore, because of the large number of network parameters and size of the training data, network training using conventional Silicon microprocessors involves constant shuttling of data between the physically separated processor and its memory units, making the von Neumann bottleneck a significant limitation in achieving good performance. Also, the temporal processing of parallel data streams in SNNs makes simulating them in the conventional computer architecture very time consuming.

Platforms based on field-programmable gate arrays, embedded processors, and graphical processing units (GPUs) have been employed for the simulation of large SNNs and DNNs [8]. However, they are often power hungry, less scalable, and limited by the high data transfer rates, making them highly inefficient compared to the human brain. However, recent progress in nanoscale materials and devices has opened up possibilities for developing compact memory device arrays that are amenable to data storage, modification, and in-memory computation, buoying the hope for a single-chip or system-level solution that implements large neural networks approaching the efficiency of the brain.

In this article, we describe some key modeling aspects of SNNs and review the various physical aspects of the nanoscale devices that could be exploited to develop energy-efficient parallel architectures for implementing these networks. We also discuss key advances toward realizing such brain-inspired devices and the challenges in the path to full-system demonstrations.

SNNs

Neural network models can be classified into three generations, as illustrated in Figure 1. These networks mimic the multilayered architecture of the human brain with its high-fan-out connectivity, though the behavior of the neurons differs significantly in the three generations. In the first generation perceptron, the output of a neuron is binary (0, 1) and is obtained by a simple thresholding of the weighted synaptic input. In the second generation models extensively used in deep learning today (commonly referred to as ANNs), the output of a neuron can be a real number, obtained as a weighted

synaptic input and transformed using a nonlinear function such as the tanh or the sigmoid function. These network models are highly efficient for processing stored data or snapshots of events. However, for processing temporal real-time data, the human brain offers an efficient signal-encoding paradigm in which information is encoded in the time of binary spike events. Essentially, each neuron can be thought of as a leaky integrator of the input current, and the integrated signal is used to determine the time of spike [9].

While the behavior of real neurons is mediated by complex ion channel dynamics, we will now describe the essential mechanisms of spike initiation and how these are used to inspire the development of simplified neuron models capturing some essential signal encoding characteristics. We will also discuss the plasticity behavior and associated models for synapses, as it is crucial to understanding the learning mechanisms necessary for creating SNNs capable of performing useful cognitive tasks.

NEURON MODELS

The first complete, biologically plausible model of the spiking neuron was developed by Hodgkin and Huxley, and incorporates the detailed dynamics of the membrane potential and the Na, K, and leak ion channels in a set of four coupled differential equations [10]. However, this model is not suitable or necessary for engineering applications, and several simplified models have been proposed based on model-order-reduction strategies. The second-order model proposed by Izhikevich [11] and the adaptive exponential integrate-and-fire (IF) model proposed by Brette and Gerstner [12] are sufficiently rich to capture most of the spiking dynamics observed in biological neurons.

The most computationally simple spiking-neuron model is that of the leaky integrate-and-fire (LIF) model [13]. The LIF model represents the potential of a neuron as the voltage across a capacitor connected in parallel with a leaky conductance path and is charged by incoming input currents. The membrane potential $V(t)$ evolves according to the differential equation

$$C\frac{dV(t)}{dt} = -g_L(V(t) - E_L) + I_{\text{syn}}(t). \quad (1)$$

When $V(t)$ exceeds a threshold V_T, a spike is issued and transmitted to the downstream synapses; the membrane potential is reset to its resting value E_L after the spike. C and g_L model the membrane's capacitance and leak conductance, respectively. Biological neurons enter a refractory period immediately after a spike is issued, during which another spike cannot be issued. This can be implemented by holding the membrane potential at $V(t) = E_L$ for a short refractory period, t_{ref}, after the issue of a spike. Note that the LIF model is a special case of the more general Spike Response Model commonly used in neuroscience literature [14]. IF neuron models, which neglect the leak term, are also used in different SNN demonstrations, where they operate by directly integrating the incoming spikes [15], [16].

SYNAPSE MODELS

While neurons issue spikes that are the tokens of information processing in the brain, it is the conductivity of synaptic junctions and its modulation that determines the communication pathways in the brain. Synapses are junctions between the axon of a transmitter neuron and the dendritic terminals of the receptor neurons. These junctions regulate the flow of signals between the neurons through the issue of neurotransmitters [17]. The released neurotransmitters bind to the postsynaptic neuron, allowing ionic current to flow into the downstream neuron and it is this feature that is essentially modeled in artificial neural models.

In the first two generations of neural network models, synaptic strength is modeled as a real number (positive or negative) and is adjusted based on various learning rules to optimize a cost function. In artificial SNNs, the synapse is typically modeled as a filter, which converts incoming spikes to postsynaptic current waveforms, and is scaled by a real-valued synaptic strength. The filter kernel of the synapse, $\alpha(t)$, is typically modeled using a single or double-decaying exponential function or a low-pass filter response [18], [19]. The spikes arriving at a synapse having a strength (weight) w will generate a postsynaptic current $[I_{\text{syn}}(t)]$ in its downstream neuron, given by the expressions

$$c(t) = \sum_i \delta(t - t^i) * \alpha(t) \quad (2)$$

and

$$I_{\text{syn}}(t) = w \times c(t), \quad (3)$$

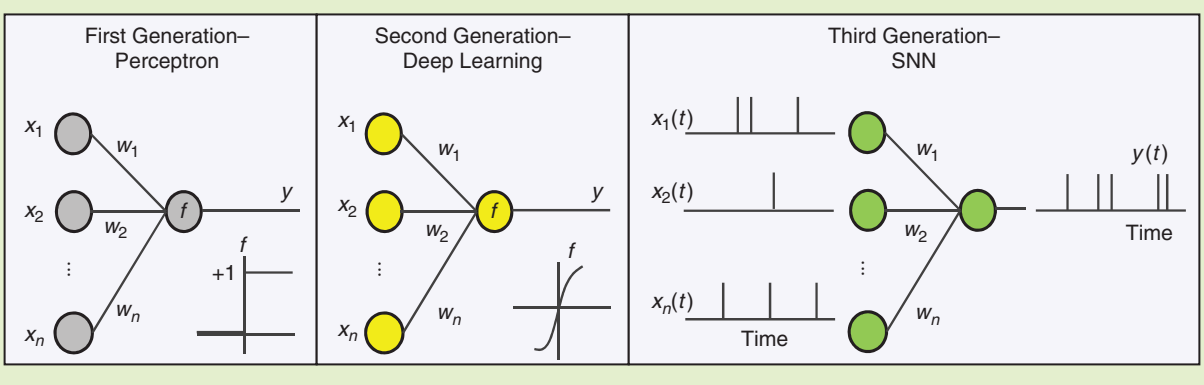


FIGURE 1 Three generations of neural network models.

where t^i denotes the time of issue of the i^{th} incoming spike and $*$ is the convolution operator. Note that there is a strong nonlinearity between the times of spikes issued by the LIF neuron and the times of spikes arriving on its incoming synapses, due to the reset after each spike. In (2) and (3), the synaptic current does not depend on the membrane potential of the postsynaptic neuron, although this is an approximation, as it is indeed a function of the difference between the reversal potential and the membrane potential of the postsynaptic neurons in biological networks.

Biological synapses and axons have a delay associated with them for transporting spikes to the downstream neurons [20]. Several efforts on developing learning algorithms have also made use of these delays as adjustable parameters in addition to the synaptic weights [21], [22]. It has also been shown that the presence of synaptic delays in SNNs increases their information capacity [23], [24]. Various neuromorphic chips emulating SNNs also implement axonal and synaptic delays as programmable features of the network [25], [26].

GENERALIZED LINEAR MODELS

While the previously described models are useful engineering abstractions for emulating network behavior, they fail to capture the statistical characteristics of spike trains obtained from intra/extracellular physiological readings. Considering the fact that neurons exhibit stochastic variability, probabilistic models are exhaustively used in neuroscience literature [28]. In an effort to capture the statistical dynamics of biological neurons, generalized linear models (GLMs) based on a linear-nonlinear Poisson model have been proposed [29]. In GLMs, linear functions of the spike stimulus (input) and generated spike history are nonlinearly transformed to determine the spike response of the neuron, as shown in Figure 2(a). GLMs have been successful in mimicking single, as well as multispiking, neuronal readings from different regions of the brain [27], [29], and [30] [Figure 2(b)–(e)]. Moreover, these models may allow for the development of mathematically tractable forms of learning rules for SNNs [31].

SYNAPTIC PLASTICITY

Neurobiological studies have shown that the strength of the synapses undergoes changes depending on the activity patterns of its upstream and downstream neurons [17]. Depending on the nature of the excitation, some synaptic modifications last only for a few seconds or minutes (short-term plasticity), whereas some changes persist for much longer durations (long-term plasticity) [9]. One of the most prominent adaptation rules was given by D. O. Hebb, who postulated that the strength of the synaptic connection between two neurons is proportional to their correlated spiking rates or activities [32].

However, a drawback of this rule is that there is no mechanism to bound the weights under the conditions of persistent firing. The Spike-timing-dependent plasticity (STDP) rule can address this issue [33] because the weights get updated according to the precise timings of spikes from the pre-(t_{pre}) and post-(t_{post}) synaptic neurons in a specific learning window. There are several studies showing that such timing-dependent plasticity rules could be used in spiking networks for supervised and unsupervised learning tasks [18], [34]–[36].

Inspired by biologically observed plasticity behaviors that involve the effect of

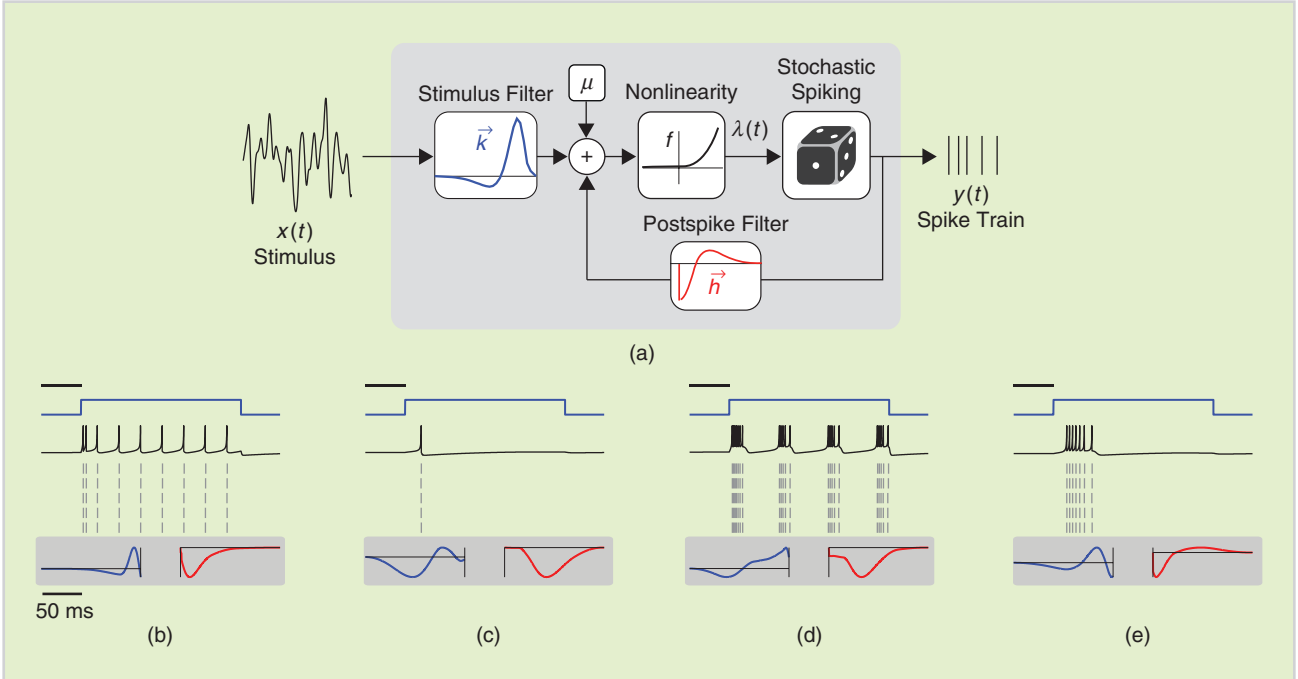


FIGURE 2 (a) A generalized linear model in which a neuron's spiking rate, $y(t)$, is nonlinearly determined by a linear function of input stimulus and spike history. By adjusting the shape of the stimulus and feedback kernels, a wide variety of neuronal behaviors can be generated, such as (b) tonic spiking, (c) phasic spiking, (d) tonic bursting, and (e) phasic bursting [27].

neuro-modulators in addition to pre- and postsynaptic traces on synaptic strength adaptation, other learning rules have been proposed [37]. For instance, the SuperSpike supervised learning rule [38] incorporates the error, postsynaptic neuron membrane potential, and presynaptic spike trace for calculating weight updates. Since standard backpropagation uses the same weights for both forward and backward pass, which is not biologically plausible, a new learning scheme called *feedback alignment* has been proposed: one set of synaptic weights is used for forward pass and a different, randomly chosen set is used for backward error propagation [39]. This rule has been applied to train SNNs in an online manner, although further improvements are necessary to improve network performance [40].

In addition to these biologically inspired learning schemes, there have been numerous efforts to derive learning rules for SNNs analytically [41]–[43]. Efficient methods have also been proposed to convert deep networks trained using backpropagation to their equivalent spiking versions [15], [16], [19], [44]. SNNs obtained using these approaches have shown state-of-the-art inference accuracies for the benchmark ImageNet classification problem [44]. Highlighting the benefits of SNNs in terms of energy efficiency, a near-two-times reduction in the number of operations has been reported compared to deep ANNs for benchmark problems based on the MNIST (Modified National Institute of Standards and Technology) and CIFAR-10 (Canadian Institute For Advanced Research) databases.

Even though significant strides have been made in developing learning algorithms for SNNs, further work is required to demonstrate that deep spiking networks can efficiently use the temporal dimension for information encoding and learning and to quantify their performance metrics for large benchmark problems.

SIGNAL ENCODING

Analogous to the brain efficiently sampling real-world information using our sense organs, real-time data must be encoded into spikes for the SNNs for further processing. A straightforward approach might be to use a rate-coding scheme in which real numbers are scaled and translated into the rate of arrival of spikes, which can be fed to SNNs. However, rate codes are inefficient and slow since the neurons must effectively wait for a certain duration to estimate the firing rate and make decisions. Hence, several schemes have been proposed in which information is encoded using the precise spike timings, inspired by the brain [45], [46].

Latency codes encode information in the time to first spike after a reference signal. In its most efficient form, only the first spike is relevant and the spiking neuron could be shut off by inhibition until the onset of the next stimulus. Phase codes are a variant of this rule in which the reference signal is a periodic oscillation and the phase of the spike with respect to the oscillation encodes the information. Such background oscillations have been observed in hippocampus, visual cortex, and other brain areas [47].

Multiplexed codes with multiple coding schemes could also be used to encode complementary information in different time scales. For example, short time-scale phase information may be multiplexed with long-duration spike rates. A recent work suggests using a variant of STDP known as *fatiguing STDP* to learn in the presence of multiplexed codes such as timing and rate [48]. Moreover, the noise in spike codes may be reduced by using homogeneous populations of neurons to represent the same information (population coding).

Inspired by these encoding mechanisms in the brain, hardware sensors have also been used for event-based representations. The dynamic vision sensor camera encodes only pixel-level changes from motion, instead of sending entire frames at a fixed rate [49]. Similarly, the silicon cochlea chip generates activity patterns in different frequency ranges in an address event representation format from stereo audio signals [50].

> GLMs have been successful in mimicking single, as well as multispiking, neuronal readings from different regions of the brain.

SPECIAL-PURPOSE HARDWARE

The high-fanout architecture in the brain (and also in ANNs) enables multiple streams of data that encode different spatial and temporal entities to be integrated in parallel to make decisions. However, modern computers are designed for sequential processing based on the von Neumann architecture. While central processing units (CPUs) and GPUs can be used to simulate this parallelism by sequential processing of information and storing the intermediate results in memory, this is highly inefficient for simulating large networks, which has prompted the search for better architectural implementations for emulating brain-inspired networks efficiently.

There are two energy-intensive operations in neural network emulation: 1) parallel signal propagation, which is weighted according to synaptic strength and summed based on network connectivity, and 2) event-driven updates of synaptic weights across multiple layers of the network. Various neuromorphic chips have been demonstrated over the past five years that achieve these operations by trying to address the von Neumann bottleneck [25], [26], [51], [52]. The architecture in most of these chips is based on a tiled array of crossbars, where small blocks of synaptic memory arrays (using SRAM cells) are tiled in a two-dimensional array, such that networks for a wide variety of applications can be mapped onto them. Figure 3 illustrates the tiled-array concept used in the million-neuron TrueNorth chip from International Business Machines Corporation (IBM) and

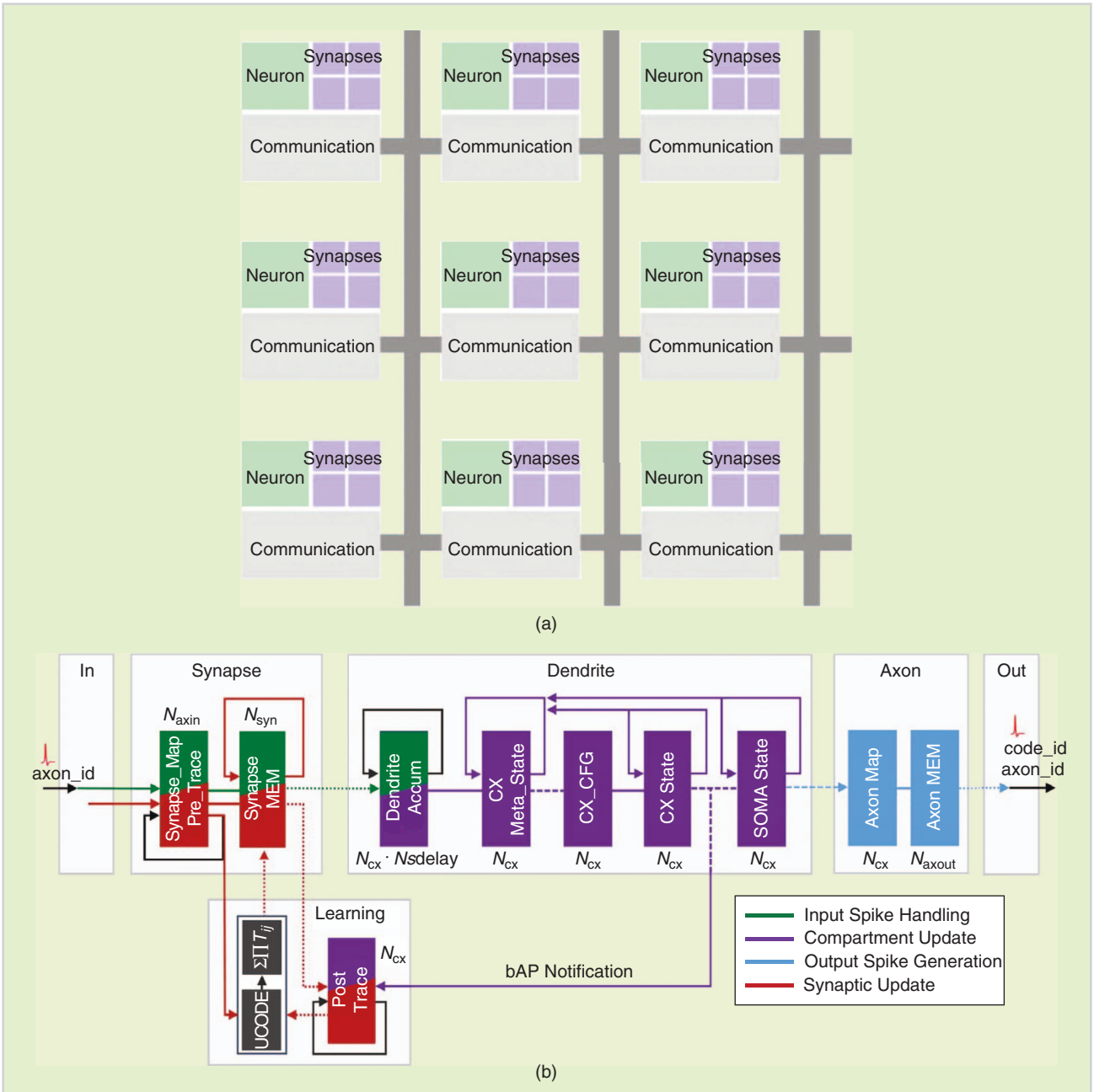


FIGURE 3 (a) The architecture of the TrueNorth chip indicating the close integration of memory and processor, along with an event-driven communication network [25] and (b) the micro architecture of a Loihi Neuromorphic core showing the functionalities of different units [26].

the high-level architecture of the digital CMOS Loihi learning chip developed by Intel using a 14-nm CMOS process for realizing SNNs.

While these CMOS-based designs illustrate the potential and feasibility of using these special-purpose chips for implementing a wide variety of cognitive tasks [53], high-level design studies suggest that significant improvements in efficiency are possible if nanoscale devices could be engineered specifically for emulating the function of neurons and synapses [54]. Nanoscale crosspoint arrays, with neuronal devices at the periphery and resistive memory devices as synapses, have been used to implement ANNs (nonspiking) for pattern classification problems [55], [56]. These networks perform matrix multiplication of neuronal inputs (V_j denoting the output of neuron j in the input layer) with the synaptic weights (G_{ij} denotes the conductivity of the synapse between neuron j in the input layer to neuron i in the output layer) utilizing Kirchhoff's law of current addition according to the relation

$$I_i = \sum_j G_{ij} V_j. \tag{4}$$

The use of crossbars reduces the multiplication complexity from $O(N^2)$ to

$O(1)$, where N is the number of neurons in a layer (Figure 4).

The tiled crossbar-array architecture is ideally suited to implement large spiking networks because the computation within the core can be performed in the analog domain and the only signal to be transmitted between cores are binary spike events. Neurons in a core can connect to synapses in other cores by storing the target axonal addresses in a lookup table and utilizing an on-chip routing network. The routing network could be asynchronous or driven by a high-speed clock (compared to the emulation dynamics of the neurons and synapses), ensuring that all spikes are routed to its destinations faithfully accounting for any synaptic delays [57], [58].

Neurons in a core can connect to synapses in other cores by storing the target axonal addresses in a lookup table and utilizing an on-chip routing network.

MEMRISTIVE DEVICES

There have been extensive efforts directed toward engineering nanoscale devices supporting programmable, nonvolatile resistance states for solid-state memory applications. Some of these devices also exhibit memristive history-dependent current versus voltage (I-V) characteristics [64], making them ideal candidates for representing the IF dynamics of neurons as well as the plastic synaptic state in neuromorphic circuits. Note that the key signature of memristance is a pinched hysteresis in the I-V response of the device [65]. Next, we discuss some of the emerging nanoscale device technologies that exhibit such desirable characteristics.

PHASE-CHANGE MEMORY

Phase-change memory (PCM) is one of the most mature nonvolatile memory technologies today and is based on chalcogenide alloys such as GeTe and $Ge_2Sb_2Te_5$, [66], [67]. The reversible electrical-resistance switching based on phase transition in these materials was discovered by Ovshinsky in 1968 [68]. If large currents (with densities exceeding 10^6 A/cm^2) are passed through polycrystalline-thin films of the material (typically <100-nm thick) sandwiched between inert metal electrodes sufficient to raise the temperature above the melting point (>600 °C), and if the input excitation is subsequently removed quickly (within a few nanoseconds), the molten region can be quenched into an amorphous volume [Figure 5(a)]. Since the resistivity of the amorphous phase of the material is much higher compared to the crystalline phase, the device is effectively switched to a high-resistance state by this electrical pulse. In the high-resistance state, if the applied voltage is such that the electric field across the amorphous volume exceeds a critical field, the device exhibits a negative differential resistance transition accompanied by a rapid increase in the current through the device. With appropriately chosen programming pulses that raise the film temperature above the crystallization temperature (but below the melting point), the amorphous region can be annealed back to its polycrystalline phase, and the low-resistance state of the device can be restored.

PCM devices exhibit excellent endurance (>10^{12} programming cycles) and retention (>10 years at 85 °C) characteristics [69], [70]. The switching speed of the device lies in the range of a few tens to hundreds of nanoseconds. Furthermore, the crystallization of the amorphous volume could be implemented in an incremental manner by using partial-crystallization pulses, enabling the device conductance to be gradually increased to higher levels. However, the melt-quench process is less gradual, making it difficult to reduce the conductance levels gradually. As a result, a single PCM cell could be used to mimic gradual potentiation observed in biological synapses.

If two PCM devices are used in a differential configuration (i.e., $G_{\text{eff}} = G^+ - G^-$), then both gradual potentiation and depression can be achieved, by incrementally increasing one of the G^+ or G^- devices with a periodic reinitialization of the conductance of saturated devices [71]. There are many studies showing gradual conductance evolution and STDP behavior in PCM devices [72] [73] and using them for supervised and unsupervised training of ANNs [74] and SNNs [75].

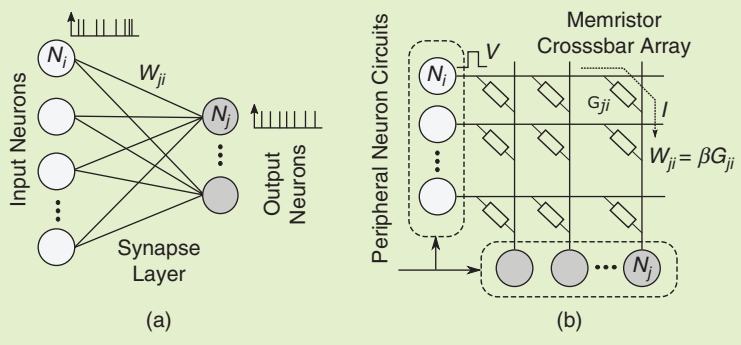


FIGURE 4 (a) Neural networks are brain-inspired computational models designed for parallel processing and decision making and (b) crossbar arrays with memristive devices at the junctions can efficiently implement this parallel connectivity, synaptic communication, and plasticity in hardware.

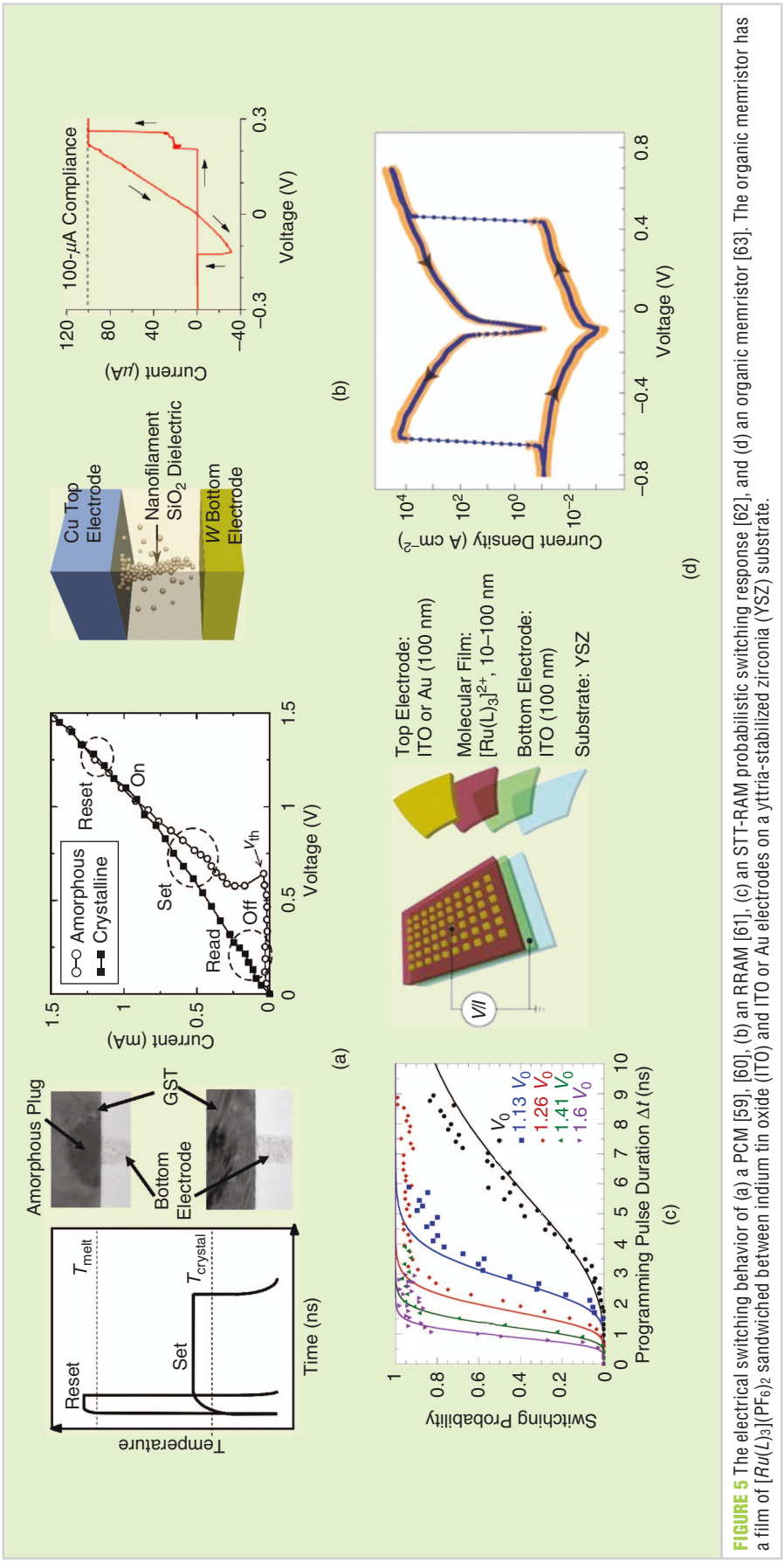


FIGURE 5 The electrical switching behavior of (a) a PCM [59], [60], (b) an RRAM [61], (c) an STT-RAM probabilistic switching response [62], and (d) an organic memristor [63]. The organic memristor has a film of $[Ru(L)_3](PF_6)_2$ sandwiched between indium tin oxide (ITO) and ITO or Au electrodes on a yttria-stabilized zirconia (YSZ) substrate.

RESISTIVE RANDOM-ACCESS MEMORY

Resistive random-access memory (RRAM) devices exhibit conductance modulation based on electric field-driven rearrangement of mobile-charged species in a dielectric material sandwiched between two metal electrodes [76]. The electrochemical process mediating the conduction modulation can be anion induced or cation induced. Anion-type RRAMs are characterized by low-resistance conductance pathways formed by the migration of oxygen vacancies. This low-resistance state can be reversed by applying an electric field in the opposite direction causing the recombination of oxygen ions with the vacancies and switching the device back to a high-resistance state. Anion-type RRAMs often require an inert electrode, which are oxygen-ion active or can act as an oxygen-ion reservoir during resistance switching. Dielectric thin films such as TiO_x, HfO_x, SiO_x, TaO_x, AlO_x, and WO_x have demonstrated this kind of oxygen-vacancy-mediated resistive switching.

Cation-type RRAMs are often characterized by a metallic filament connecting the top and the bottom metal electrodes following a redox reaction; they are also referred to as *conductance bridge RAM* (*CBRAM*) *devices* [Figure 5(b)] [77]. These devices require an active top electrode, e.g., Ag (silver) and Cu (copper), whose ions are mobile in the dielectric under an applied field. During electrical programming, the metal ions will oxidize, migrate into the dielectric, and will get reduced at the other electrode, forming a filamentary path. A reversal of the applied field will result in ionic motion in the opposite direction, breaking the filament and switching the device back to a high-resistance state. CBRAMs have a high on-off ratio with lower operating voltages, compared to that of the oxygen-vacancy RRAMs.

The low-resistance conductance paths formed in the dielectrics are nanoscale filaments, which result in the observation of quantized-conductance

states [61], [76], and [78]. RRAMs are extensively researched for their gradual conductance change and as synaptic devices [79]. The material combination, device geometry, interface effects, doping, annealing, and other fabrication techniques could be engineered to attain gradual resistance transitions in these devices [80], [81]. For example, W/Al/$Pr_{0.7}Ca_{0.3}MnO_3$ (PCMO)/Pt-based RRAM show a gradual conductance change due to the oxidation and reduction of AlO_x at the Al/PCMO interface [82], and this dielectric-based device has been used for STDP demonstrations using biomimetic programming waveforms [83]. In a recent work, the filamentary pathway was confined to engineered dislocations in a SiGe epitaxial layer, resulting in gradual conductance changes in the device and improvements in retention, reliability, and endurance [84].

MAGNETIC RAM

Magnetic RAMs store information in the relative orientation of the magnetization of two ferromagnetic plates separated by a thin insulating material resulting in a magnetic tunnel junction (MTJ) [85]. One of the plates is of fixed magnetic orientation, while the other is a free layer, whose magnetic orientation can be altered by an external field. The plates could be in parallel or antiparallel orientation at equilibrium, resulting in a high or low conductance state respectively for the junction. The magnetization of the layer is retained in the absence of an applied voltage, allowing stable binary data storage in the device.

A variant of the MRAM is the spin-transfer torque (STT) RAM, with lower power consumption and more scalability. When directed to the free layer, a spin-polarized current, which is created by passing it through the fixed magnetic layer, results in spin-angular momentum exchange because of the interaction between the spins of local magnetization of the layer and that of the free electrons. The free-layer magnetic orientation can be switched to a parallel or antiparallel state depending on the direction of the current [86], [87]. While STT-RAMs predominantly show binary states, there has also been an increased effort in making domain wall (DW)-based devices to store multiple states [88].

Furthermore, by either adjusting the programming-current amplitude or the pulsewidth below the critical conditions for switching, the probability of switching can be tuned [Figure 5(c)] [62], [89], [90]. This probabilistic switching behavior could be used to realize a gradual conductance change or STDP in a synapse composed of multiple devices configured in a parallel configuration [91].

FERROELECTRIC RAM

Ferroelectric RAMs use a thin layer of ferroelectric material sandwiched between two metal electrodes. The ferroelectric polarization state of the material is switched between two stable states for conventional solid-state memory applications [92]. Multiple regions of different polarization vectors called *ferroelectric domains* may be present in a ferroelectric sample [93]. Recently it has been demonstrated that the resistance of $BaTiO_3$(2 nm)/$La_{0.67}Sr_{0.33}MnO_3$(30-nm)-based ferroelectric tunnel junctions can be tuned based on the relative fraction of the ferroelectric domains that points toward one electrode or the other [94]. It is possible to alter the domain population by the application of electrical pulses to the electrodes, thereby tuning the electrical resistivity. This concept has been used to mimic synaptic plasticity in supertetragonal $BiFeO_3$ tunnel barriers using electrical programming waveforms [95].

ORGANIC MEMORIES

Memristors based on organic compounds are attractive because of the possibility of inexpensive solution-processing-based fabrication and chemical tunability of their properties. These devices have an organic thin film that is sandwiched between electrodes. Because of the complex nature of the compounds involved, the physics behind the switching mechanism is often unclear. Structural changes, redox reaction, and field-driven polarization have been proposed to explain the switching transitions in these materials [63], [96], [97]. However, except for a recent demonstration [Figure 5(d)] [63], these devices generally suffer from low endurance and stability.

In a study based on organic terpyridyl-iron polymer-based memristor [96] gradual conductance changes, short-term potentiation and long-term potentiation have been demonstrated, taking advantage of the drift of the programmed states. Although these devices require high switching voltages (~3 V) and long (millisecond) switching times, such explorations demonstrate the feasibility of realizing the complex dynamics of synapses and neurons in potentially inexpensive hardware platforms.

NANOSCALE SPIKING NEURONS

While there has been extensive research to mimic the complex dynamics of neurons using subthreshold CMOS circuits [101], nanoscale device-based approaches offer the potential for further reductions in area and power, with significant enhancements to the scalability of neuromorphic designs. Recently, there have been a few single-device designs and demonstrations to mimic the behavior of leaky IF neurons.

INSULATOR-METAL TRANSITION NEURONS

Two-terminal devices based on transition-metal oxides such as VO_2 and NbO_2, exhibit insulator-metal transition (IMT) mediated by thermally or electrically triggered phase transitions in nanosecond time scales [102]. This phase transition is volatile, and as the triggering source (voltage/temperature) falls below a threshold, the device switches back to its initial state. This behavior could be used to design oscillatory circuits [Figure 6(a)] and have been proposed to mimic the neuronal spiking behavior [98], [103]. The leaky-integration behavior could be incorporated by using an R—C low-pass filter at the gate of the access transistor connected to the IMT device. As the integrated gate voltage exceeds a threshold, the device switches between a high-resistance insulating phase to a low-resistance metallic phase in an oscillatory manner until the gate voltage subsides below a certain threshold. IMT-based neurons have been used as stochastic sampling machines to improve the generalization accuracy in MNIST handwritten image classification problems and project a 30-times power reduction compared to a 22-nm CMOS-ASIC implementation [104].

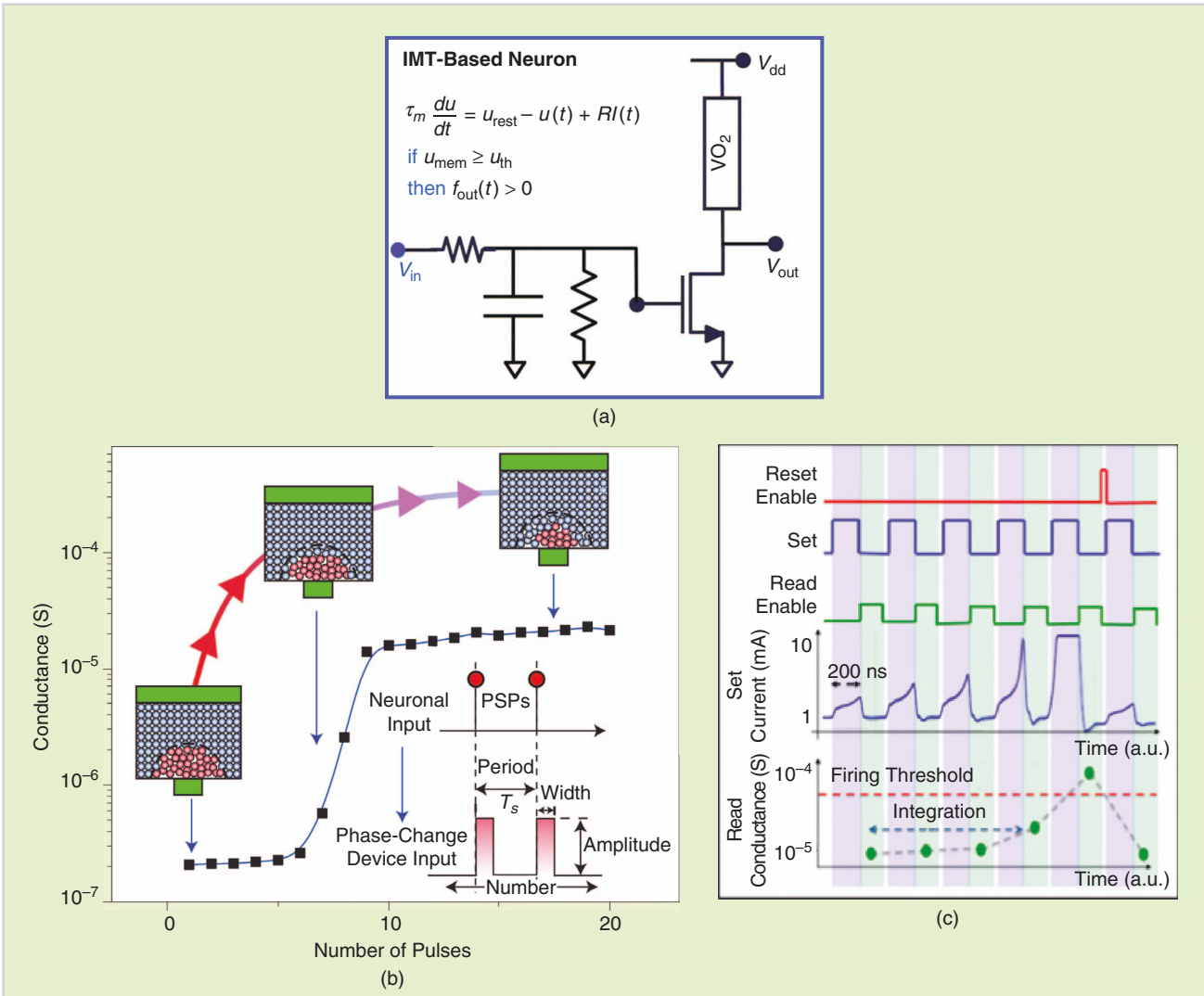


FIGURE 6 (a) An IMT-based neuron circuit [98]; (b) a PCM-based LIF neuron in which the amorphous volume encodes the membrane potential, which evolves depending on the incoming postsynaptic potentials (PSPs) [99]; (c) the characteristics of PCMO-based RRAM used to emulate neuron integrate-and-fire behavior [100]. a.u.: arbitrary unit.

PCM NEURONS

Similarly, PCM-based stochastic neurons have been proposed in which the phase configuration of the chalcogenide film is used to represent the neuron membrane potential [Figure 6(b)] [99]. The input current integrated by an LIF neuron is supplied as short crystallizing pulses, which gradually reduces the amorphous volume inside the device and can represent the integration behavior of the membrane potential. Once the device conductance analogous to the membrane potential crosses a threshold, the device is reset by a RESET programming pulse. Separate devices could be used for excitatory and inhibitory input accumulation; therefore, a single neuron may be composed of more than one PCM device. Mechanisms external to the device dynamics are necessary to incorporate the neuron leak. Since the crystallization process in PCM devices is stochastic, the overall IF dynamics of the neuron is also stochastic.

RRAM NEURONS

Based on the previously described integration and reset principle, RRAMs have also been shown to mimic the features of a spiking neuron when operated in the low-current regime [105]. Recently, PCMO-based RRAM devices have been demonstrated as (IF) neurons in SNNs for solving a pattern classification problem [Figure 6(c)] [100]. These devices have also been able to mimic the spike frequency adaptation characteristics of biological neurons.

SPINTRONIC NEURONS

Various forms of neuronal behaviors ranging from simple-step (nonspiking) neurons to stochastic-spiking neurons have been demonstrated in spin-based devices [88]. Studies have shown that domain wall (DW)-based devices have an input current-integrating feature through the motion of the DW making them ideal for mimicking an IF-spiking neuron. More than a 1,000-fold reduction in energy

consumption for MNIST and CIFAR-10 image classification compared to 45-nm CMOS-based designs have been projected using a hybrid device-circuit-architecture co-simulation framework.

MTJ-based spintronic oscillators have been used to mimic biological neurons and interneuron communication through magnetic-field coupling [106]. They have also been experimentally demonstrated for spoken digit-recognition tasks with accuracies close to state-of-the-art neural networks.

The key to realizing plastic synapses using memristive devices to achieve online learning is to convert the weight updates requested by the training algorithm into reliable conductance changes in the device using suitable programming waveforms.

NANOSCALE SYNAPSES

Synapses and their plasticity are key to memory, learning, and adaptation in neural networks. From an information storage perspective, biological synapses in the hippocampus are estimated to be capable of storing 26 distinguishable synaptic states, corresponding to 4.7 bits [108]. The excitatory postsynaptic current measurement in [33] suggests that synaptic conductivity can support a dynamic range (on-off ratio) of at least 50. In addition to STDP, short-term plasticity observed in synapses also seem to have computational roles in biological networks [109]. While the true computational advantages of STDP and similar biological learning mechanisms are still unclear, there are attempts to relate STDP-like rules to SGD-based supervised learning algorithms [110]–[112]. In this section, we review how some of these synaptic properties can be efficiently implemented by the various nanoscale devices discussed previously.

Because of their ability to retain a programmed state and modulate their conductivity in an activity- or history-dependent manner, memristive devices are ideally suited to represent plastic synapses in hardware implementations. It is desirable that the device exhibit symmetric and gradual conductance changes with an appropriate choice of programming pulses so that they naturally accumulate the conductance changes dictated by local spike events.

Most experimental memristors whose conductance can be programmed to analog states are modulated by atomic/ionic rearrangement, which is stochastic and prone to read noise. Examples of memristive devices that exhibit gradual conductance change include PCM and RRAMs based on PCMO, HfO_x, TiO_x, and so on. Irrespective of the device geometry, material systems, and the switching mechanisms, none of these experiments have demonstrated more than 4–5 bits per device. Furthermore, the conductance change achievable with simple programming pulses is state dependent and asymmetric. However, such stochasticity may not be too detrimental for implementing online learning systems; in fact, similar stochastic characteristics are measured in biological synapses as well and may very well play a key role in fuzzy information processing in the brain. Both artificial and spiking neural networks may also leverage this stochasticity to perform useful computations, as the final decisions of the network are dependent on the relative magnitude and overall distribution of a large number of synapses, rather than the absolute value of any single device.

On the other hand, memristors based on spin orientation and filamentary switching can only be reliably programmed to two states. These binary devices also exhibit probabilistic switching behaviors around their switching threshold and could be exploited to realize gradual conductance change and STDP behavior in a multimemristor configuration [91]. External random number generators and pulse amplitude modulation have been suggested to control the binary switching probability of CBRAM and have been used in feature extraction networks [113], [114]. In another recent study, two binary MTJ devices whose switching probabilities were externally controlled were used to implement a synapse, with one of the devices implementing short-term plasticity while the other implemented long-term plasticity [115].

The key to realizing plastic synapses using memristive devices to achieve online learning is to convert the weight updates requested by the training algorithm into reliable conductance changes in the device using suitable programming waveforms (Figure 7). In STDP-based training, spike-timing intervals must be transformed into amplitudes and polarity of the effective programming pulses.

One way to achieve this is to use two waveforms of carefully designed shapes to be applied from the two terminals of the device so that the conductance is altered only when they overlap in time. The programming waveform could be engineered to realize arbitrary forms of STDP characteristics [116], [117]. This approach has been used for PCM- [73], OxRAM- [118], CBRAM- [79], and PCMO- [83] based devices. The knowledge of underlying device physics and operating characteristics is essential to design effective programming waveforms that achieve the desired plasticity behavior in an energy-efficient manner.

Another approach used to achieve successful training with memristive device arrays is to program the devices only if the desired change is comparable to the update granularity of the device [119]. The smaller changes requested by the training algorithm could be accumulated in the digital memory until it is sufficiently large to be reliably programmed to the device.

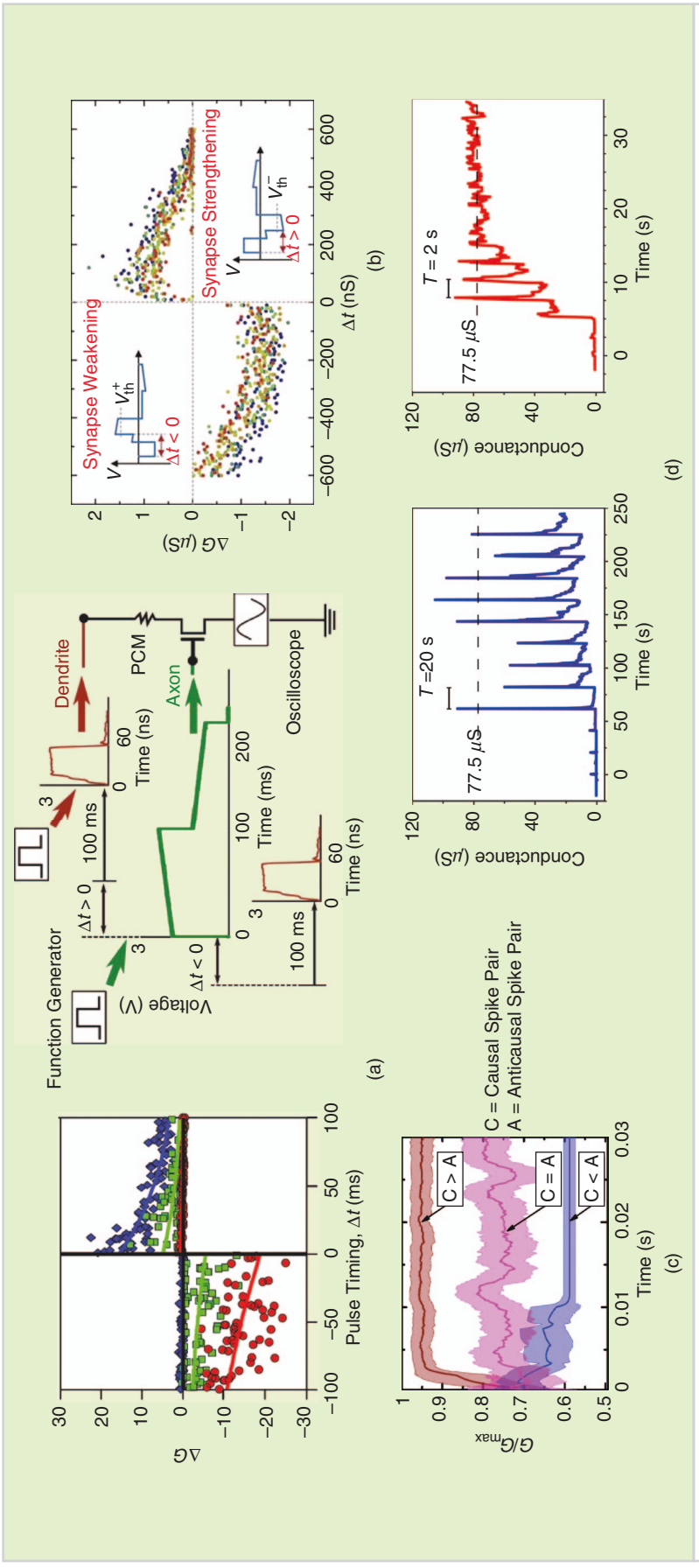


FIGURE 7 (a) STDP behavior in PCM devices caused by programming waveform engineering [73], (b) STDP observed in FeRAM [95], (c) simulations showing gradual conductance changes by using a combination of multiple binary switching devices per synapse [91], and (d) short- and long-term plasticity in the atomic switch [107].

NETWORK IMPLEMENTATION

Neural networks are currently simulated on von Neumann machines with training acceleration offered by the parallel processing cores of GPUs. There are two main motives behind the quest for building dedicated neuromorphic hardware: 1) acceleration of machine-learning algorithms for large real-world applications and 2) in-the-field learning for energy- and memory-constrained embedded applications. While power-hungry and bulky GPU clusters are clearly not suitable for the latter application, they are nonoptimal even for the former application considering the training times for large networks and their associated power budgets. Therefore, dedicated hardware capable of emulating neural network operations in parallel in an accelerated and energy-efficient manner could transform both enterprise computing and intelligent embedded Internet of Things (IoT) platforms.

For hardware applications, SNNs have certain advantages compared to their second-generation counterparts, especially for processing real-time data. Spike-based information encoding enables sparse representations of real-world data, and the communication of network tokens (spikes) through on-chip routing networks is clearly much more efficient than transporting real-valued activation values. However, the temporal dynamics of spiking neurons are more complex than ANNs and call for more dedicated parallel processors.

The SpiNNaker project, which is attempting to create a parallel network of 1 million ARM (advanced reduced-instruction set computer machine) cores capable of simulating a billion neurons, is one approach to achieve this using existing technologies [122]. IBM's TrueNorth chip for SNNs is a parallel computing platform with 1 million spiking neurons and 256 million nonplastic-SRAM synapses fabricated in 28-nm CMOS [25]. By using event-based spike communication and optimized low-leakage transistor technology for fabrication, the platform has demonstrated approximately 10^5-times improvement in energy per event in computational efficiency compared to conventional

CPUs for network emulation. However, this system does not support on-chip learning, since multibit synapses cannot be incorporated even in state-of-the-art technology nodes for such large networks within a reasonable silicon area. In contrast, Intel's Loihi chip fabricated in 14-nm CMOS supports several on-chip learning rules with up to 9 bits for each synaptic weight but can only emulate networks with up to 130,000 neurons and 130 million synapses [26].

Herein lies the promise and potential of hardware solutions based on nanoscale devices. Using these nanoscale devices, the synaptic cell area can be made as small as $4F^2$ [123] and an IF neuronal-device area can be reduced to $225F^2$ [124], resulting in integration densities exceeding 10^8 neurons/cm^2 and 10^{10} synapses/cm^2 in 10-nm nodes (F = 29 nm). Nonvolatile-memory crossbar-memory arrays can perform the large vector-matrix multiplications in $O(1)$ complexity. This enables the parallel-signal propagation in real time in the analog domain. Additionally, the programmability of these nonvolatile memory devices makes on-chip learning feasible. It is possible to envision large, multilayered SNNs using tiled crossbar arrays in which the synaptic communication and adaptation is done in analog mode, and communication between cores is implemented using digital peripheral circuitry.

Numerous studies have demonstrated small SNNs on special hardware using memristive devices as synaptic storage elements and also as spiking neuronal units for various classification problems. The initial demonstration of PCM as a crossbar-compatible synaptic device for SNN has been reported in several works using device-conductance-response models [71], [72], [125]. On-chip STDP learning was demonstrated recently in a 90-nm neuromorphic chip with 256 × 256 PCM cells configured as analog synapses, as illustrated in Figure 8(a) [120].

RRAM devices have also been used to mimic the biological mechanisms of precise timing-based synaptic-weight updates in a network of spiking neurons [126]. Compared to PCM-based arrays, optimized RRAMs [as shown in the inset of Figure 8(b)] are projected to show energy-efficiency improvements by a factor of 100–1,000 [121].

Based on a hybrid device-circuit-architecture co-simulation framework, it has been projected that an all-spin SNN neuromorphic system as an inference engine [Figure 8(c)] can have more than 1,000-times energy efficiency and more than a 100-times speedup compared to a 45-nm CMOS baseline for multilayer SNN architectures for classification on datasets such as MNIST, CIFAR-10, and SVHNs (street-view house numbers) [127].

FUTURE OUTLOOK

The spike-based architecture of the human brain enables the efficient encoding of real-time data, and parallel- and event-driven communication between neurons in a high fan-out network. These features also make SNNs attractive candidates for hardware implementation of cognitive computing applications.

However, in terms of computational capability demonstrations, SNNs lag behind their second-generation counterparts today. This could be attributed to the following factors: the powerful SGD-based back-propagation algorithms are not directly applicable to spiking neurons due to their nondifferentiable dynamics; and secondly, the inherent nature of SNNs to process data as time-series events and the temporal integration of LIF neurons make simulating these networks in conventional computational systems highly time consuming, thus preventing the implementation and testing of large network architectures and algorithms. Hence, developments in the domain of parallel computational architectures including dedicated hardware implementations that could accelerate SNN simulations may also advance their learning algorithms.

The computational capabilities of the SNNs are a relatively less-explored domain, though recent results are highly promising. For instance, starting from the initially chaotic networks of spiking neurons, SNNs can be trained to implement a wide variety of complex cognitive tasks such as reproducing the singing behavior of songbirds and encoding and replaying a movie scene [128]. Using the temporal domain for information encoding also endows a higher representational power for spiking networks; a single binary threshold neuron with k inputs can store $2k$ bits of information [129], while a spiking neuron can store up to $3k$ bits of information [130].

However, the true potential of SNNs for ubiquitous IoT and other embedded and enterprise applications can be realized only if dedicated parallel and energy-efficient hardware solutions can be developed. Though significant progress has been made on many fronts, several challenges remain before large-scale, multiarray, crossbar-based nanodevice platforms for SNNs become a reality. The following are a few of these hardware-related challenges.

1) Novel devices for supporting massively parallel and adaptive networks: While there is a significant body of work on building nanoscale memristive devices for mimicking STDP-like plasticity or supporting gradual weight updates, further work is required to mimic other key functionalities such as current integration at the periphery, stochastic neuronal spiking with automatic reset and refractory period after spike, synaptic delays, and structural plasticity that enables new connections to be made (or deleted) between neurons based on activity. Today, these functions are implemented using large CMOS circuits and simply optimizing the crossbar memory for synapses will make these peripheral circuits the efficiency limiting factor. Compact, nanoscale single-device solutions at the energy scales of biology for these functions will be necessary to build truly integrated and interconnected networks of neurons and synapses.
2) Improving the learning capacity of synapses: Conductance change granularity is a key factor in determining the trainability of hardware neural networks. The process of training is often a search over an error space created in hyperdimensions of synaptic weights by taking small increments along the

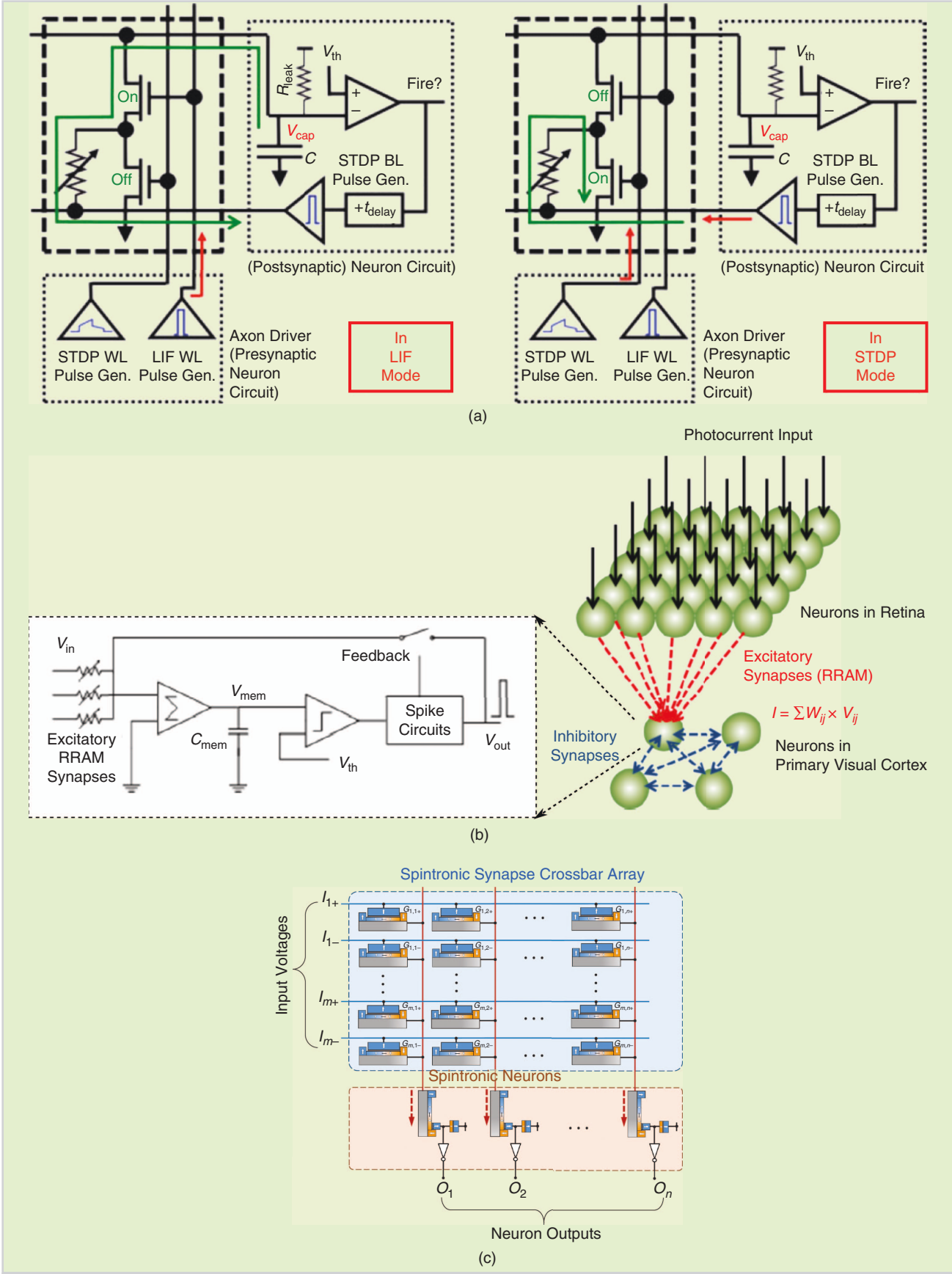


FIGURE 8 (a) The schematic of a PCM synapse-based circuit design supporting on-chip learning [120], (b) RRAM as synapses [121], and (c) all-spintronic SNN architecture [88]. All of these studies use some form of the STDP learning rule for weight adaptation. Gen: generator; WL: word line; BL: bit line.

direction of the gradient. Depending on the complexity of the error surface, the step size needs to be small enough to reach the optima, and the device must support symmetric updates in both directions of conductance changes based on simple programming pulses. If devices with tunable delays can be designed, this may add a new dimension for improving the learning capacity of spiking networks [23], [24]. While mixed precision architecture in which a high-precision CMOS gradient accumulator compensates for the reduced-device precision may be used to address this issue, a nanoscale device-level solution is highly desirable.

3) Improving system-level reliability: All memristive devices that rely on atomic or ionic rearrangement exhibit intradevice and interdevice variability in terms of resistance levels, programming voltages, and limited programming endurance and retention times. While materials and device optimization as well as better fabrication technologies could address some of these issues to a certain extent, algorithmic- and architectural-level innovations that can mitigate these limitations that are inherent at the nanoscale will be crucial to guaranteeing system-level performance and reliability.

With sufficient investments in interdisciplinary research, we are optimistic that these challenges can be met and the long-awaited dream of reverse engineering the brain to build intelligent machines that can be ubiquitously deployed in the field may well be realized in the near future.

ACKNOWLEDGMENTS

This work was supported in part by the CAMPUSENSE project grant from CISCO Systems Inc., the Semiconductor Research Corporation, and the National Science Foundation grant 1710009. S. R. Nandakumar gratefully acknowledges IBM Research Zurich for hosting him as a research intern at the time this article was written.

ABOUT THE AUTHORS

S.R. Nandakumar (ns599@njit.edu) is with the New Jersey Institute of Technology, Newark.

Shruti R. Kulkarni (srk68@njit.edu) is with the New Jersey Institute of Technology, Newark.

Anakha V. Babu (av442@njit.edu) is with the New Jersey Institute of Technology, Newark.

Bipin Rajendran (bipin@njit.edu) is with the New Jersey Institute of Technology, Newark.

REFERENCES

[1] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436–444, May 2015.

[2] P. Werbos, "Beyond regression: New tools for prediction and analysis in the behavioral sciences," Ph.D. dissertation, Harvard Univ., Cambridge, MA, 1974.

[3] S. Shoham, D. H. O'Connor, and R. Segev, "How silent is the brain: Is there a 'dark matter' problem in neuroscience?" *J. Comparative Physiol. A*, vol. 192, pp. 777–784, Aug. 2006.

[4] B. Wang, W. Ke, J. Guang, G. Chen, L. Yin, S. Deng, Q. He, Y. Liu, T. He, R. Zheng, Y. Jiang, X. Zhang, T. Li, G. Luan, H. Lu, D. Haidong, M. Zhang, X. Zhang, and Y. Shu, "Firing frequency maxima of fast-spiking neurons in human, monkey, and mouse neocortex," *Frontiers Cellular Neurosci.*, vol. 10, p. 239, Oct. 2016.

[5] W. Maas, "Networks of spiking neurons: The third generation of neural network models," *Neural Netw.*, vol. 10, no. 9, pp. 1659–1671, Dec. 1997.

[6] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vision*, vol. 115, pp. 211–252, Dec. 2015.

[7] C. Sun, A. Shrivastava, S. Singh, and A. Gupta, "Revisiting unreasonable effectiveness of data in deep learning era," in *Proc. IEEE Int. Conf. Computer Vision*, 2017, pp. 843–852.

[8] S. R. Kulkarni, A. V. Babu, and B. Rajendran, "Spiking neural networks—Algorithms, hardware implementations and applications," in *Proc. 60th Int. IEEE Midwest Symp. Circuits and Systems (MWSCAS)*, Aug. 2017, pp. 426–431.

[9] P. Dayan and L. Abbott, *Theoretical Neuroscience: Computational and Mathematical Modeling of Neural Systems.* Cambridge, MA: MIT Press, 2001.

[10] A. L. Hodgkin and A. F. Huxley, "A quantitative description of membrane current and its application to conduction and excitation in nerve," *J. Physiol.*, vol. 117, pp. 500–544, Aug. 1952.

[11] E. M. Izhikevich and G. M. Edelman, "Large-scale model of mammalian thalamocortical systems," *Proc. Nat. Acad. Sci.*, vol. 105, no. 9, pp. 3593–3598, 2008.

[12] R. Brette and W. Gerstner, "Adaptive exponential integrate-and-fire model as an effective description of neuronal activity," *J. Neurophysiol.*, vol. 94, no. 5, pp. 3637–3642, 2005.

[13] L. F. Abbott, "Lapicque's introduction of the integrate-and-fire model neuron (1907)," *Brain Res. Bulletin*, vol. 50, pp. 303–304, Nov.-Dec. 1999.

[14] W. Gerstner, R. Ritz, and J. L. van Hemmen, "Why spikes? Hebbian learning and retrieval of time-resolved excitation patterns," *Biol. Cybern.*, vol. 69, pp. 503–515, Sept. 1993.

[15] B. Rueckauer, I.-A. Lungu, Y. Hu, and M. Pfeiffer. (2016). Theory and tools for the conversion of analog to spiking convolutional neural networks. arXiv. [Online]. Available: https://arxiv.org/abs/1612.04052

[16] P. U. Diehl, D. Neil, J. Binas, M. Cook, S.-C. Liu, and M. Pfeiffer, "Fast-classifying, high-accuracy spiking deep networks through weight and threshold balancing," in *Proc. IEEE Int. Joint Conf. Neural Networks (IJCNN)*, 2015, pp. 1–8.

[17] W. Gerstner and W. M. Kistler, *Spiking Neuron Models: Single Neurons, Populations, Plasticity.* Cambridge, U.K.: Cambridge Univ. Press, 2002.

[18] F. Ponulak and A. Kasiński, "Supervised learning in spiking neural networks with ReSuMe: Sequence learning, classification, and spike shifting," *Neural Comput.*, vol. 22, pp. 467–510, Feb. 2010.

[19] E. Hunsberger and C. Eliasmith. (2016). Training spiking deep networks for neuromorphic hardware. arXiv. [Online]. Available: https://arxiv.org/abs/1611.05141

[20] R. Miledi, "The measurement of synaptic delay, and the time course of acetylcholine release at the neuromuscular junction," *Proc. Roy. Soc. Lond. B*, vol. 161, no. 985, pp. 483–495, 1965.

[21] A. Taherkhani, A. Belatreche, Y. Li, and L. P. Maguire, "DL-ReSuMe: A delay learning-based remote supervised method for spiking neurons," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, pp. 3137–3149, Dec. 2015.

[22] B. Schrauwen and J. Van Campenhout, "Improving spikeprop: Enhancements to an error-backpropagation rule for spiking neural networks," in *Proc. 15th ProRISC Workshop*, 2004.

[23] E. M. Izhikevich, "Polychronization: Computation with spikes," *Neural Comput.*, vol. 18, no. 2, pp. 245–282, 2006.

[24] W. Maass and A. M. Zador, "Dynamic stochastic synapses as computational units," *Neural Comput.*, vol. 11, no. 4, pp. 903–917, 1999.

[25] P. A. Merolla, J. V. Arthur, R. Alvarez-Icaza, A. S. Cassidy, J. Sawada, F. Akopyan, B. L. Jackson, N. Imam, C. Guo, Y. Nakamura, B. Brezzo, I. Vo, S. K. Esser, R. Appuswamy, B. Taba, A. Amir, M. D. Flickner, W. P. Risk, R. Manohar, and D. S. Modha, "A million spiking-neuron integrated circuit with a scalable communication network and interface," *Science*, vol. 345, no. 6197, pp. 668–673, 2014.

[26] M. Davies, N. Srinivasa, T. H. Lin, G. Chinya, Y. Cao, S. H. Choday, G. Dimou, P. Joshi, N. Imam, S. Jain, Y. Liao, C. K. Lin, A. Lines, R. Liu, D. Mathaikutty, S. McCoy, A. Paul, J. Tse, G. Venkataramanan, Y. H. Weng, A. Wild, Y. Yang, and H. Wang, "Loihi: A neuromorphic manycore processor with on-chip learning," *IEEE Micro*, vol. 38, pp. 82–99, Jan. 2018.

[27] A. I. Weber and J. W. Pillow, "Capturing the dynamical repertoire of single neurons with generalized linear models," *Neural Comput.*, vol. 29, no. 12, pp. 3260–3289, 2017.

[28] J. W. Pillow, L. Paninski, V. J. Uzzell, E. P. Simoncelli, and E. J. Chichilnisky, "Prediction and decoding of retinal ganglion cell responses with a probabilistic spiking model," *J. Neurosci.*, vol. 25, no. 47, pp. 11,003–11,013, 2005.

[29] J. W. Pillow, J. Shlens, L. Paninski, A. Sher, A. M. Litke, E. J. Chichilnisky, and E. P. Simoncelli, "Spatio-temporal correlations and visual signaling in a complete neuronal population," *Nature*, vol. 454, pp. 995–999, Aug. 2008.

[30] B. Babadi, A. Casti, Y. Xiao, E. Kaplan, and L. Paninski, "A generalized linear model of the impact of direct and indirect inputs to the lateral geniculate nucleus," *J. Vision*, vol. 10, pp. 22–22, Aug. 2010.

[31] A. Bagheri, O. Simeone, and B. Rajendran. (2017, Oct.). Training probabilistic spiking neural networks with first-to-spike decoding. arXiv. [Online]. Available: https://arxiv.org/abs/1710.10704

[32] D. Hebb, *Organization of Behavior.* New York: Wiley, 1949.

[33] G. Q. Bi and M.-M. Poo, "Synaptic modifications in cultured hippocampal neurons: Dependence

on spike timing, synaptic strength, and postsynaptic cell type," *J. Neurosci.*, vol. 18, pp. 10,464–10,472, Dec. 1998.

[34] P. U. Diehl and M. Cook, "Unsupervised learning of digit recognition using spike-timing-dependent plasticity," *Frontiers Comput. Neurosci.*, vol. 9, no. 99, 2015. doi: 10.3389/fncom.2015.00099.

[35] S. R. Kheradpisheh, M. Ganjtabesh, S. J. Thorpe, and T. Masquelier, "STDP-based spiking deep convolutional neural networks for object recognition," *Neural Netw.*, vol. 99, pp. 56–67, Mar. 2017.

[36] A. Tavanaei and A. S. Maida, "Multi-layer unsupervised learning in a spiking convolutional neural network," in *Proc. IEEE Int. Joint Conf. Neural Networks (IJCNN)*, May 2017, pp. 2023–2030.

[37] N. Frémaux and W. Gerstner, "Neuromodulated spike-timing-dependent plasticity, and theory of three-factor learning rules," *Frontiers Neural Circuits*, vol. 9, p. 85, Jan. 2016.

[38] F. Zenke and S. Ganguli. (2017). Superspike: Supervised learning in multi-layer spiking neural networks. arXiv. [Online]. Available: https://arxiv.org/abs/1705.11146

[39] T. P. Lillicrap, D. Cownden, D. B. Tweed, and C. J. Akerman, "Random synaptic feedback weights support error backpropagation for deep learning," *Nature Commun.*, vol. 7, Nov. 2016. doi: 10.1038/ncomms13276.

[40] E. Hunsberger, "Spiking deep neural networks: Engineered and biological approaches to object recognition," Ph.D dissertation, Depart. Syst. Design Eng., University of Waterloo, Ontario, Canada, 2018.

[41] N. Anwani and B. Rajendran, "NormAD—Normalized approximate descent based supervised learning rule for spiking neurons," in *Proc. IEEE Int. Joint Conf. Neural Networks (IJCNN)*, July 2015, pp. 1–8.

[42] J. H. Lee, T. Delbruck, and M. Pfeiffer, "Training deep spiking neural networks using backpropagation," *Frontiers Neurosci.*, vol. 10, p. 508, Nov. 2016.

[43] P. O'Connor and M. Welling. (2016). Deep spiking networks. arXiv. [Online]. Available: https://arxiv.org/abs/1602.08323

[44] B. Rueckauer, I.-A. Lungu, Y. Hu, M. Pfeiffer, and S.-C. Liu, "Conversion of continuous-valued deep networks to efficient event-driven networks for image classification," *Frontiers Neurosci.*, vol. 11, p. 682, Dec. 2017.

[45] C. P. Billimoria, R. A. DiCaprio, J. T. Birmingham, L. F. Abbott, and E. Marder, "Neuromodulation of spike-timing precision in sensory neurons," *J. Neurosci.*, vol. 26, pp. 5910–5919, May 2006.

[46] S. J. Ryan, D. E. Ehrlich, A. M. Jasnow, S. Daftary, T. E. Madsen, and D. G. Rainnie, "Spike-timing precision and neuronal synchrony are enhanced by an interaction between synaptic inhibition and membrane oscillations in the amygdala," *PLoS ONE*, vol. 7, Apr. 2012. doi: 10.1371/journal.pone.0035320.

[47] S. Panzeri, N. Brunel, N. K. Logothetis, and C. Kayser, "Sensory neural codes using multiplexed temporal scales," *Trends Neurosci.*, vol. 33, pp. 111–120, Mar. 2010.

[48] T. Moraitis, A. Sebastian, I. Boybat, M. L. Gallo, T. Tuma, and E. Eleftheriou, "Fatiguing STDP: Learning from spike-timing codes in the presence of rate codes," in *Proc. IEEE Int. Joint Conf. Neural Networks (IJCNN)*, 2017, pp. 1823–1830.

[49] C. Brandli, R. Berner, Y. Minhao, L. Shih-Chii, and T. Delbruck, "A 240 × 180 130 dB 3 μs latency global shutter spatiotemporal vision sensor," *IEEE J. Solid-State Circuits*, vol. 49, pp. 2333–2341, Oct. 2014.

[50] S. C. Liu, A. Van Schaik, B. A. Minch, and T. Delbruck, "Asynchronous binaural spatial audition sensor with 2 × 64 × 4 Channel output," *IEEE Trans. Biomed. Circuits Syst.*, vol. 8, no. 4, pp. 453–464, 2014.

[51] J. Gehlhaar, "Neuromorphic processing: A new frontier in scaling computer architecture," in *Proc. 19th Int. Conf. Architectural Support for Programming Languages and Operating Systems (ASPLOS)*, 2014, pp. 317–318.

[52] N. Qiao, H. Mostafa, F. Corradi, M. Osswald, F. Stefanini, D. Sumislawska, and G. Indiveri, "A reconfigurable on-line learning spiking neuromorphic processor comprising 256 neurons and 128k synapses," *Frontiers Neurosci.*, vol. 9, pp. 141, Apr. 2015.

[53] S. K. Esser, P. A. Merolla, J. V. Arthur, A. S. Cassidy, R. Appuswamy, A. Andreopoulos, D. J. Berg, J. L. McKinstry, T. Melano, D. R. Barch, C. di P. Nolfo, A. Datta, B. Amir, M. Taba, D. Flickner, and D. S. Modha, "Convolutional networks for fast, energy-efficient neuromorphic computing," *Proc. Nat. Acad. Sci.*, vol. 113, no 41, pp. 11,441–11,446, 2016.

[54] B. Rajendran, Y. Liu, J. Seo, K. Gopalakrishnan, L. Chang, D. Friedman, and M. Ritter, "Specifications of nanoscale devices & circuits for neuromorphic computational systems," *IEEE Trans. Electron Devices*, vol. 60, no. 1, pp. 246–253, 2013.

[55] G. Tayfun and Y. Vlasov. (2016). Acceleration of deep neural network training with resistive cross-point devices. arVix. [Online]. Available: https://arxiv.org/abs/1603.07341

[56] T. Gokmen, M. Onen, and W. Haensch. (2017). Training deep convolutional neural networks with resistive cross-point devices. arXiv. [Online]. Available: https://arxiv.org/abs/1705.08014

[57] S. Ambrogio, N. Ciocchini, M. Laudato, V. Milo, A. Pirovano, P. Fantini, and D. Ielmini, "Unsupervised learning by spike timing dependent plasticity in phase change memory (PCM) synapses," *Frontiers Neurosci.*, vol. 10, Mar. 2016. doi: 10.3389/fnins.2016.00056.

[58] S. B. Eryilmaz, S. Joshi, E. Neftci, W. Wan, G. Cauwenberghs, and H.-S. P. Wong, "Neuromorphic architectures with electronic synapses," in *Proc. 17th Int. Symp. Quality Electronic Design (ISQED)*, 2016, pp. 118–123.

[59] M. J. Breitwisch, "Phase change memory," in *Proc. 2008 Int. Interconnect Technology Conf.*, 2008, pp. 219–221.

[60] A. Pirovano, A. L. Lacaita, F. Pellizzer, S. A. Kostylev, A. Benvenuti, and R. Bez, "Low-field amorphous state resistance and threshold voltage drift in chalcogenide materials," *IEEE Trans. Electron Devices*, vol. 51, no. 5, pp. 714–719, 2004.

[61] S. R. Nandakumar, M. Minvielle, S. Nagar, C. Dubourdieu, and B. Rajendran, "A 250 mV Cu/SiO_2/W memristor with half-integer quantum conductance states," *Nano Lett.*, vol. 16, pp. 1602–1608, Mar. 2016.

[62] A. F. Vincent, J. Larroque, N. Locatelli, N. B. Romdhane, O. Bichler, C. Gamrat, W. S. Zhao, J. O. Klein, S. Galdin-Retailleau, and D. Querlioz, "Spin-transfer torque magnetic memory as a stochastic memristive synapse for neuromorphic systems," *IEEE Trans. Biomed. Circuits Syst.*, vol. 9, no. 2, pp. 166–174, 2015.

[63] S. Goswami, A. J. Matula, S. P. Rath, S. Hedström, S. Saha, M. Annamalai, D. Sengupta, A. Patra, S. Ghosh, H. Jani, S. Sarkar, M. R. Motapothula, C. A. Nijhuis, J. Martin, S. Goswami, V. S. Batista, and T. Venkatesan, "Robust resistive memory devices using solution-processable metal-coordinated azo aromatics," *Nature Mater.*, vol. 16, pp. 1216–1224, Oct. 2017.

[64] L. Chua, "Memristor—The missing circuit element," *IEEE Trans. Circuit Theory*, vol. 18, no. 5, pp. 507–519, 1971.

[65] L. Chua, "Resistance switching memories are memristors," *Appl. Phys. A, Mater. Sci. Process.*, vol. 102, no. 4, pp. 765–783, 2011.

[66] H. S. P. Wong, S. Raoux, S. Kim, J. Liang, J. P. Reifenberg, B. Rajendran, M. Asheghi, and K. E. Goodson, "Phase change memory," *Proc. IEEE*, vol. 98, pp. 2201–2227, Dec. 2010.

[67] G. W. Burr, M. J. Brightsky, A. Sebastian, H. Y. Cheng, J. Y. Wu, S. Kim, N. E. Sosa, N. Papandreou, H. L. Lung, H. Pozidis, E. Eleftheriou, and C. H. Lam, "Recent progress in phase-change memory technology," *IEEE J. Emerg. Select. Topics Circuits Syst.*, vol. 6, pp. 146–162, June 2016.

[68] S. R. Ovshinsky, "Reversible electrical switching phenomena in disordered structures," *Phys. Rev. Lett*, vol. 21, pp. 1450–1453, Nov. 1968.

[69] S. Lai and T. Lowrey, "Oum—A 180 nm nonvolatile memory cell element technology for stand alone and embedded applications," in *Proc. IEEE Int. Electron Devices Meeting (IEDM), Technical Dig*, 2001, pp. 36.5.1–36.5.4.

[70] A. Pirovano, A. Redaelli, F. Pellizzer, F. Ottogalli, M. Tosi, D. Ielmini, A. Lacaita, and R. Bez, "Reliability study of phase-change nonvolatile memories," *IEEE Trans. Device Mater. Rel.*, vol. 4, pp. 422–427, Sept. 2004.

[71] M. Suri, O. Bichler, D. Querlioz, O. Cueto, L. Perniola, V. Sousa, D. Vuillaume, C. Gamrat, and B. DeSalvo, "Phase change memory as synapse for ultra-dense neuromorphic systems: Application to complex visual pattern extraction," in *Proc. IEEE Int. Electron Devices Meeting (IEDM)*, 2011, pp. 4.4.1–4.4.4.

[72] D. Kuzum, R. G. D. Jeyasingh, B. Lee, and H.-S. P. Wong, "Nanoelectronic programmable synapses based on phase change materials for brain-inspired computing," *Nano Lett.*, vol. 12, no. 5, pp. 2179–2186, 2012.

[73] B. L. Jackson, B. Rajendran, G. S. Corrado, M. Breitwisch, G. W. Burr, R. Cheek, K. Gopalakrishnan, S. Raoux, C. T. Rettner, A. Padilla, A. G. Schrott, R. S. Shenoy, B. N. Kurdi, C. H. Lam, and D. S. Modha, "Nano-scale electronic synapses using phase change devices," *ACM J. Emerg. Technol. Comp. Syst.*, vol. 9, no. 2, p. 12, 2013.

[74] G. Burr, R. Shelby, C. di Nolfo, J. Jang, R. Shenoy, P. Narayanan, K. Virwani, E. Giacometti, B. Kurdi, and H. Hwang, "Experimental demonstration and tolerancing of a large-scale neural network (165,000 synapses), using phase-change memory as the synaptic weight element," in *Proc. IEEE Int. Electron Devices Meeting (IEDM)*, 2014, pp. 29.5.1–29.5.4.

[75] S. R. Nandakumar, I. Boybat, M. L. Gallo, A. Sebastian, B. Rajendran, and E. Eleftheriou, "Supervised learning in spiking neural networks with MLC PCM synapses," in *Proc. 75th Annu. Device Research Conf. (DRC)*, 2017, pp. 1–2.

[76] R. Waser and M. Aono, "Nanoionics-based resistive switching memories," *Nature Mater.*, vol. 6, pp. 833–840, Nov. 2007.

[77] M. Kund, G. Beitel, C.-U. Pinnow, T. Rohr, J. Schumann, R. Symanczyk, K. Ufert, and G. Muller, "Conductive bridging RAM (CBRAM): An emerging non-volatile memory technology scalable to sub 20 nm," in *Proc. IEEE Int. Electron Devices Meeting (IEDM)*, 2005, pp. 754–757.

[78] X. Zhu, W. Su, Y. Liu, B. Hu, L. Pan, W. Lu, J. Zhang, and R. W. Li, "Observation of conductance quantization in oxide-based resistive switching memory," *Adv. Mater.*, vol. 24, no. 29, pp. 3941–3946, 2012.

[79] S. R. Nandakumar and B. Rajendran, "Synaptic plasticity in a memristive device below 500 mv," *ECS Trans.*, vol. 77, no. 2, pp. 31–37, 2017.

[80] I. Valov, R. Waser, J. R. Jameson, and M. N. Kozicki, "Electrochemical metallization memories—fundamentals, applications, prospects," *Nanotechnology*, vol. 22, p. 254003, July 2011.

[81] J. J. Yang, D. B. Strukov, and D. R. Stewart, "Memristive devices for computing," *Nature Nanotechnol.*, vol. 8, no. 1, pp. 13–24, 2013.

[82] S. Park, J. Noh, M.-L. Choo, A. M. Sheri, M. Chang, Y.-B. Kim, C. J. Kim, M. Jeon, B.-G. Lee, B. H. Lee, and H. Hwang, "Nanoscale RRAM-based synaptic electronics: Toward a neuromorphic computing device," *Nanotechnology*, vol. 24, p. 384009, Sept. 2013.

[83] N. Panwar, D. Kumar, N. Upadhyay, P. Arya, U. Ganguly, and B. Rajendran, "Memristive synaptic

plasticity in PCMO RRAM by bio-mimetic programming," in *Proc. 72nd Annu. Device Research Conf. (DRC)*, 2014, pp. 135–136.

[84] S. Choi, S. H. Tan, Z. Li, Y. Kim, C. Choi, P.-Y. Chen, H. Yeon, S. Yu, and J. Kim, "SiGe epitaxial memory for neuromorphic computing with reproducible high performance based on engineered dislocations," *Nature Mater.*, vol. 17, pp. 1–6, Jan. 2018.

[85] N. Locatelli, V. Cros, and J. Grollier, "Spin-torque building blocks," *Nature Mater.*, vol. 13, no. 1, pp. 11–20, 2014.

[86] D. C. Ralph and M. D. Stiles, "Spin transfer torques," *J. Magn. Magn. Mater.*, vol. 320, no. 7, pp. 1190–1216, 2008.

[87] T. Kawahara, K. Ito, R. Takemura, and H. Ohno, "Spin-transfer torque RAM technology: Review and prospect," *Microelectron. Rel.*, vol. 52, no. 4, pp. 613–627, 2012.

[88] A. Sengupta and K. Roy, "A vision for all-spin neural networks: A device to system perspective," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 63, no. 12, pp. 2267–2277, 2016.

[89] A. F. Vincent, J. Larroque, N. Locatelli, N. B. Romdhane, O. Bichler, C. Gamrat, W. S. Zhao, J.-O. Klein, S. Galdin-Retailleau, and D. Querlioz, "Spin-transfer torque magnetic memory as a stochastic memristive synapse for neuromorphic systems," *IEEE Trans. Biomed. Circuits Syst.*, vol. 9, no. 2, pp. 166–174, 2015.

[90] U. Roy, T. Pramanik, L. F. Register, and S. K. Banerjee, "Write error rate of spin-transfer-torque random access memory including micromagnetic effects using rare event enhancement," *IEEE Trans. Magn.*, vol. 52, no. 10, pp. 1–6, 2016.

[91] A. Singha, B. Muralidharan, and B. Rajendran, "Analog memristive time dependent learning using discrete nanoscale RRAM devices," in *Proc. 2014 Int. Joint Conf. Neural Networks (IJCNN)*, pp. 2248–2255.

[92] V. Garcia and M. Bibes, "Ferroelectric tunnel junctions for information storage and processing," *Nature Commun.*, vol. 5, pp. 1–12, 2014.

[93] J. Guyonnet, *Ferroelectric Domain Walls*. Switzerland: Springer, 2014.

[94] A. Chanthbouala, V. Garcia, R. O. Cherifi, K. Bouzehouane, S. Fusil, X. Moya, S. Xavier, H. Yamada, C. Deranlot, N. D. Mathur, M. Bibes, A. Barthélémy, and J. Grollier, "A ferroelectric memristor," *Nature Mater.*, vol. 11, pp. 860–864, Oct. 2012.

[95] S. Boyn, J. Grollier, G. Lecerf, B. Xu, N. Locatelli, S. Fusil, S. Girod, C. Carrétéro, K. Garcia, S. Xavier, J. Tomas, L. Bellaiche, M. Bibes, A. Barthélémy, S. Saïghi, and V. Garcia, "Learning through ferroelectric domain dynamics in solid-state synapses," *Nature Commun.*, vol. 8, pp. 1–7, Apr. 2017.

[96] X. Yang, C. Wang, J. Shang, C. Zhang, H. Tan, X. Yi, L. Pan, W. Zhang, F. Fan, Y. Liu, Y. Chen, G. Liu, and R.-W. Li, "An organic terpyridyl-iron polymer based memristor for synaptic plasticity and learning behavior simulation," *RSC Advances*, vol. 6, no. 30, pp. 25,179–25,184, 2016.

[97] T. Berzina, A. Smerieri, M. Bernab, A. Pucci, G. Ruggeri, V. Erokhin, and M. P. Fontana, "Optimization of an organic memristor as an adaptive memory element," *J. Appl. Phys.*, vol. 105, no. 12, May 2009.

[98] M. Jerry, W. Y. Tsai, B. Xie, X. Li, V. Narayanan, A. Raychowdhury, and S. Datta, "Phase transition oxide neuron for spiking neural networks," in *Proc. 74th Annu. Device Research Conf. (DRC)*, 2016, pp. 1–2.

[99] T. Tuma, A. Pantazi, M. Le Gallo, A. Sebastian, and E. Eleftheriou, "Stochastic phase-change neurons," *Nature Nanotechnol.*, vol. 11, no. 8, pp. 693–699, 2016.

[100] S. Lashkare, S. Chouhan, T. Chavan, A. Bhat, P. Kumbhare, and U. Ganguly, "PCMO RRAM for integrate-and-fire neuron in spiking neural networks," *IEEE Electron Device Lett.*, vol. 39, no. 4, pp. 484–487, Apr. 2018.

[101] G. Indiveri, B. Linares-Barranco, T. J. Hamilton, A. van Schaik, R. Etienne-Cummings, T. Delbruck, S.-C. Liu, P. Dudek, P. Hafliger, S. Renaud, J. Schemmel, G. Cauwenberghs, J. Arthur, K. Hynna, F. Folowosele, S. Saighi, T. Serrano-Gotarredona, J. Wijekoon, Y. Wang, and K. Boahen, "Neuromorphic silicon neuron circuits," *Frontiers Neurosci.*, vol. 5, May 2011. doi: 10.3389/fnins.2011.00073.

[102] M. Son, J. Lee, J. Park, J. Shin, G. Choi, S. Jung, W. Lee, S. Kim, S. Park, and H. Hwang, "Excellent selector characteristics of nanoscale VO_2 for high-density bipolar ReRAM applications," *IEEE Electron Device Lett.*, vol. 32, no. 11, pp. 1579–1581, 2011.

[103] K. Moon, E. Cha, D. Lee, J. Jang, J. Park, and H. Hwang, "ReRAM-based analog synapse and IMT neuron device for neuromorphic system," in *Proc. Int. Symp. VLSI Technology, Systems and Applications (VLSI-TSA)*, 2016, pp. 1–2.

[104] M. Jerry, A. Parihar, B. Grisafe, A. Raychowdhury, and S. Datta, "Ultra-low power probabilistic IMT neurons for stochastic sampling machines," in *Proc. Int. Symp. VLSI Technology, Systems and Applications (VLSI-TSA)*, June 2017, T186–T187.

[105] J. Woo, D. Lee, Y. Koo, and H. Hwang, "Dual functionality of threshold and multilevel resistive switching characteristics in nanoscale hfo2-based RRAM devices for artificial neuron and synapse elements," *Microelectron. Eng.*, vol. 182, pp. 42–45, Oct. 2017.

[106] J. Torrejon, M. Riou, F. A. Araujo, S. Tsunegi, G. Khalsa, D. Querlioz, P. Bortolotti, V. Cros, K. Yakushiji, A. Fukushima, H. Kubota, S. Yuasa, M. D. Stiles, and J. Grollier, "Neuromorphic computing with nanoscale spintronic oscillators," *Nature*, vol. 547, no. 7664, p. 428, 2017.

[107] T. Ohno, T. Hasegawa, T. Tsuruoka, K. Terabe, J. Gimzewski, and M. Aono, "Short-term plasticity and long-term potentiation mimicked in single inorganic synapses," *Nature Mater.*, vol. 10, pp. 591–595, June 2011.

[108] T. M. Bartol, C. Bromer, J. Kinney, M. A. Chirillo, J. N. Bourne, K. M. Harris, and T. J. Sejnowski, "Nanoconnectomic upper bound on the variability of synaptic plasticity," *eLife*, vol. 4, Nov. 2015. doi: 10.7554/eLife.10778.

[109] Z. Rotman, P.-Y. Deng, and V. A. Klyachko, "Short-term plasticity optimizes synaptic information transmission," *J. Neurosci.*, vol. 31, no. 41, pp. 14,800–14,809, 2011.

[110] Y. Dan and M.-M. Poo, "Spike timing-dependent plasticity: From synapse to perception," *Physiological Rev.*, vol. 86, pp. 1033–1048, July 2006.

[111] Y. Bengio, D.-H. Lee, J. Bornschein, T. Mesnard, and Z. Lin. (2016, Aug.). Towards biologically plausible deep learning. arXiv. [Online]. Available: http://arxiv.org/abs/1502.04156

[112] Y. Bengio, T. Mesnard, A. Fisher, S. Zhang, and Y. Wu. (2015, Sept.). An objective function STDP. arXiv. [Online]. Available: http://arxiv.org/abs/1509.05936

[113] M. Suri, D. Querlioz, O. Bichler, G. Palma, E. Vianello, D. Vuillaume, C. Gamrat, and B. DeSalvo, "Bio-inspired stochastic computing using binary CBRAM synapses," *IEEE Trans. Electron Devices*, vol. 60, pp. 2402–2409, July 2013.

[114] S. Yu, B. Gao, Z. Fang, H. Yu, J. Kang, and H.-S. P. Wong, "Stochastic learning in oxide binary synaptic device for neuromorphic computing," *Frontiers Neurosci.*, vol. 7, p. 186, Oct. 2013.

[115] G. Srinivasan, A. Sengupta, and K. Roy, "Magnetic tunnel junction based long-term short-term stochastic synapse for a spiking neural network with on-chip STDP learning," *Scientific Rep.*, vol. 6, pp. 1–13, June 2016.

[116] S. Saïghi, C. G. Mayr, T. Serrano-Gotarredona, H. Schmidt, G. Lecerf, J. Tomas, J. Grollier, S. Boyn, A. F. Vincent, D. Querlioz, S. La Barbera, F. Alibart, D. Vuillaume, O. Bichler, C. Gamrat, and B. Linares-Barranco, "Plasticity in memristive devices for spiking neural networks," *Frontiers Neurosci.*, vol. 9, pp. 1–16, Mar. 2015.

[117] N. Panwar, B. Rajendran, and U. Ganguly, "Arbitrary spike time dependent plasticity (STDP) in memristor by analog waveform engineering," *IEEE Electron Device Lett.*, vol. 38, pp. 740–743, June 2017.

[118] Y.-F. Wang, Y.-C. Lin, I.-T. Wang, T.-P. Lin, and T.-H. Hou, "Characterization and modeling of nonfilamentary Ta/TaOx/TiO2/Ti analog synaptic device," *Scientific Rep.*, vol. 5, p. 10,150, Sept. 2015.

[119] S. R. Nandakumar, M. Le Gallo, I. Boybat, B. Rajendran, A. Sebastian, and E. Eleftheriou. (2017, Dec.). Mixed-precision training of deep neural networks using computational memory. arXiv. [Online]. Available: https://arxiv.org/abs/1712.01192

[120] S. Kim, M. Ishii, S. Lewis, T. Perri, M. BrightSky, W. Kim, R. Jordan, G. Burr, N. Sosa, A. Ray, J.-P. Han, C. Miller, K. Hosokawa, and C. Lam, "NVM neuromorphic core with 64k-cell (256-by-256) phase change memory synaptic array with on-chip neuron circuits for continuous in-situ learning," in *Proc. IEEE Int. Electron Devices Meeting (IEDM)*, 2015, p. 17.

[121] S. Yu, B. Gao, Z. Fang, H. Yu, J. Kang, and H. S. P. Wong, "A neuromorphic visual system using RRAM synaptic devices with sub-pj energy and tolerance to variability: Experimental characterization and large-scale modeling," in *Proc. Int. Electron Devices Meeting (IEDM)*, 2012, pp. 10.4.1–10.4.4.

[122] S. B. Furber, F. Galluppi, S. Temple, and L. A. Plana, "The SpiNNaker project," *Proc. IEEE*, vol. 102, pp. 652–665, May 2014.

[123] B. Chen, X. Wang, B. Gao, Z. Fang, J. Kang, L. Liu, X. Liu, G-q Lo, and D-l Kwong, "Highly compact (4F2) and well behaved nano-pillar transistor controlled resistive switching cell for neuromorphic system application," *Scientific Rep.*, vol. 4, p. 6863, May 2015.

[124] V. Ostwal, R. Meshram, B. Rajendran, and U. Ganguly, "An ultra-compact and low power neuron based on SOI platform," in *Proc. Int. Symp. VLSI Technology, Systems and Application (VLSI-TSA)*, 2015, pp. 1–2.

[125] D. Garbin, M. Suri, O. Bichler, D. Querlioz, C. Gamrat, and B. DeSalvo, "Probabilistic neuromorphic system using binary phase-change memory (pcm) synapses: Detailed power consumption analysis," in *Proc. 13th IEEE Int. Conf. Nanotechnology (IEEE-NANO)*, 2013, pp. 91–94.

[126] D. Garbin, O. Bichler, E. Vianello, Q. Rafhay, C. Gamrat, L. Perniola, G. Ghibaudo, and B. DeSalvo, "Variability-tolerant convolutional neural network for pattern recognition applications based on oxram synapses," in *Proc. IEEE Int. Electron Devices Meeting (IEDM)*, 2014, pp. 28.4.1–28.4.4.

[127] A. Sengupta, A. Ankit, and K. Roy, "Performance analysis and benchmarking of all-spin spiking neural networks (special session paper)," in *Proc. Int. Joint Conf. Neural Networks (IJCNN)*, 2017, pp. 4557–4563.

[128] W. Nicola and C. Clopath, "Supervised learning in spiking neural networks with FORCE training," *Nature Commun.*, vol. 8, no. 1, pp. 1–15, 2017.

[129] D. MacKay, *Information Theory, Inference and Learning Algorithms*. Cambridge, U.K.: Cambridge Univ. Press, 2003.

[130] R. Gutig and H. Sompolinsky, "The tempotron: A neuron that learns spike timing-based decisions," *Nature Neurosci.*, vol. 9, pp. 420–428, Mar. 2006.

N