Memristive devices are essential for artificial neural networks (ANNs) due to their similarity to biological synapses and neurons in structure, dynamics, and electrical behaviors. By building a crossbar array, memristive devices can be used to conduct in-memory computing efficiently. Herein, approaches to realize memristive neural networks (memNNs) from the device level to the system level are introduced with state-of-art experimental demonstrations. First, algorithm fundamentals for networks and device fundamentals for synapses and neurons are briefly given to provide guidance for developing ANNs based on memristive devices; second, recent advances in memristive synapses are discussed on the device level, including the optimization of device, the emulation of biological functions and the array integration; third, artificial neurons based on complement metal-oxide-semiconductor (CMOS) transistors and memristive devices are described; then, systemic demonstrations and latest developments of memNNs are elaborated; finally, summary and perspective on memristive devices and memNNs are presented.

1. Introduction

Thanks to the advance in transistor scaling, computing capabilities of central processing units (CPUs) and graphics processing units (GPUs) have been increased greatly, resulting in the fast development of artificial neural networks (ANNs). ANNs predict the correlation between a set of input and output parameters by emulating the principles of biological neural networks. Nowadays, ANNs have been applied to a broad spectrum, such as computer vision, natural language processing, autonomous driving, and decision making. Although ANNs are advantageous in some specific tasks, such as playing the game of Go, the human brain outperforms ANNs in cognitive and classification tasks with fast speed and high power efficiency. The advantages of the human brain lie in the special architectures with high massive parallelism, spiking neurons, and enormous number of basic computing units of neurons ($\approx 10^{13}$) and synapses ($\approx 10^{15}$), whereas computers show high speed of basic operation and precision. The massively parallel processing compensates the slow speed of neurons and synapses, whereas spiking neurons compute more powerful than other types of neurons. The parallel processing occurs based on variable synaptic weights, which is the basis of memory. Moreover, the same information is processed by many neurons and transmitted to a downstream neuron, which enhances the precision as well. Therefore, the human brain shows a salient feature, i.e., in-memory computing, suitable for high-performance applications with a large amount of data. In contrast, conventional digital computing systems face the Von Neumann bottleneck, because in the Von Neumann architecture, computing units and memory are separated, hindering the enhancement of ANNs. Thus, several novel architectures are promoted, such as brain-inspired computing (processing information based on spiking neurons, also called as spiking neural networks (SNNs)) and in-memory computing on chip, to overcome the Von Neumann bottleneck. To implement the new architectures, artificial synapses and neurons are necessary.

The massive parallelism of the brain takes advantage of a large number of neurons and synapses; however, the emulation of the functions of biological synapses and neurons requires tens of complement metal-oxide-semiconductor (CMOS) transistors. Therefore, it is urgent to realize synapses and neurons with a simplified method for the development of ANNs. As synapses adjust their weights step-by-step repeatedly, synaptic devices need to exhibit analog switching behavior. However, biological neurons show the analog membrane potential and generate the all-or-nothing rapidly, so both analog and digital devices are acceptable for the emulation of neurons.

Many emerging devices have been proposed to develop ANNs based on hardware, e.g., memristive devices, phase change memories (PCMs), magnetic random-access memories (MRAMs), and ferroelectric field-effect transistors (FeFETs). Each technology has its advantages and disadvantages. PCM and MRAM show more mature development, but their analog switching performances require improvement. Although FeFETs show better analog switching performance, more structure complexity is unavoidable. Due to simple structure, high programming...
speed (<1 ns), efficient energy consumption (<1 pJ), great scalability (2 nm) and versatile performances (analog/digital and volatile/nonvolatile switching), memristive devices, the resistance of which can be modulated by electrical stimuli, are promising for ANNs. Memristive devices have been realized in various materials, such as oxides, polymers, 2D materials and perovskite materials. In addition, the vector–matrix multiplication (VMM) operation, the most data-intensive computation in ANNs, can be conducted in a memristive crossbar array. For instance, as shown in Figure 1a, a process that an input vector \([X_1, X_2, X_3, X_4]\) is multiplied with a synaptic matrix \(W\) can be demonstrated with a memristive array when the time complexity equals to \(O(1)\). The device conductance \(G\) is considered as the synaptic weight \(W\), while the input \(X\) is represented by a voltage pulse \(V\), and the output current \(I\) indicates the output \(Y\), then \(I_i = \sum G_{ij} V_j\) according to Ohm and Kirchhoff laws.

Two approaches to implement in-memory computing ANNs are proposed based on memristive devices. One approach is to perform ANNs in hardware, e.g., feed-forward ANNs and recurrent neural networks (RNNs), to accelerate the VMM operations based on memristive crossbar arrays, when the node conductance of the memristive array represents the synaptic weight. In this approach, the massive operations of VMM are conducted parallely with fast speed and ultrahigh energy efficiency. Another approach is to realize SNNs to faithfully mimic the biological brain in structure and function. The information is encoded with spike-timing, and synapses adjust weights according to biological learning rules, such as the spike timing-dependent plasticity (STDP) learning rule. More details about these two approaches are presented in Section 2.

Very recently, several review articles were published in terms of device reliability, memristive arrays, in-memory computing and neuroinspired computing chips. Different from these published works, this article concentrates on the construction of memristive neural networks (memNNs), including the realization of memristive synapses/neurons and their connectivity for specific neural networks. The article is organized as
The SRDP learning rule depicts the synaptic weights modiﬁcation according to the synaptic changes in presynaptic spikes, whereas the LTD occurs when the postsynaptic spikes precede the presynaptic spikes; long-term depression (LTD) occurs when the postsynaptic spikes follow the presynaptic spikes; long-term potentiation (LTP) occurs when the postsynaptic spikes follow the presynaptic spikes; long-term depression (LTD) occurs when the postsynaptic spikes precede the presynaptic spikes. The SRDP learning rule depicts the synaptic weights modiﬁed by the timing or rating (Figure 1d). As SNNs compute only when events occur, they are more energy efﬁcient than digital computing. The artiﬁcial neurons and synapses in this case are more similar to those in biology. Noteworthy, the architectures for nonspiking ANNs, e.g., CNN and RNN, can also be utilized in SNNs for more complex tasks.

As shown in Figure 1d, a neuron in SNNs receives spikes from presynapses and the membrane potential is changed. When the membrane potential is beyond the threshold, the neuron generates a spike to postsynapses. Several models were proposed to illustrate the properties of neurons, e.g., the Hodgkin–Huxley (HH) model and the leaky integrate-and-ﬁre (LIF) model. The HH model describes the state of ion channels in the neuron membrane and is closest to the bioneuron dynamics whereas only the basic functions are included in the LIF model to simplify the computing process. Synapses in SNNs require multilevel performances and to obey the learning rule in a biological brain, such as STDP.

As the neuron function based on spike trains are nondifferentiable, to train SNNs directly by the error backpropagation algorithm is diﬃcult, although there were some attempts in software. However, as referring to biological systems, the training process in SNNs based on memristive devices can be realized based on STDP or spike rating-dependent plasticity (SRDP) according to encoding schemes. The STDP learning rule illustrates the Hebbian learning, which states the relationship between the timing of neurons and the updating of the weights. Long-term potentiation (LTP) occurs when the postsynaptic spikes follow the presynaptic spikes; long-term depression (LTD) occurs when the postsynaptic spikes precede the presynaptic spikes. The SRDP learning rule depicts the synaptic weights modiﬁed by the timing or rating. The high frequency of presynaptic spikes leads to the LTP and the low frequency induces the LTD. However, the learning rules of STDP and SRDP are both localized spatially and temporally, therefore,
it is not necessary to train an entire network at once; SNN can be trained layer-by-layer to get a relatively high accuracy.[47]

2.2. Device Fundamentals

Memristive devices, analog and digital type, are used to realize artificial synapse and neurons. The conductance of an analog device is modulated gradually by electrical stimuli, whereas a digital one shows abrupt binary resistance states. Analog-type memristive devices are more likely to play the role of biological synapses, because the multiple resistance states can represent synaptic weights; whereas digital memristive devices with volatility are compatible to the generation of all-or-nothing signals of the neurons in SNNs.

Based on the switching mechanism, the analog switching can be classified into three categories: filament width modulation, interfacial modulation, and channel composition modulation. The conductance of a filamentary device can be modulated through tuning the size (width) of the filament, as shown in the top panel of Figure 2a. Ions (such as oxygen vacancies [\(V_{O}^{\bullet}\)]) migrate in an oxide under an electric field, inducing the increment or decrement in the width of the filament, and the conventional current–voltage (I–V) relation is shown in the bottom panel of Figure 2a. Usually, the conductance of the filament approximately equals to that of the device, as found in Ta\(_2\)O\(_5\) and HfO\(_2\).[48–50] Furthermore, the filament correlates with the type of the interfacial barrier, and the area ratio of different types of barriers determines the device conductance. For instance, the oxygen-vacancy channel in Pd/WO\(_3\)/W forms a high conductance channel induced by the tunneling current, whereas the rest region maintains the low conductance state due to the Schottky barrier at the oxide/electrode interface. The area ratio of the low-conductance interfacial barrier can be decreased by widening the filament, thus increasing the device conductance.[51,52]

In contrast, interfacial modulating devices are switched based on the tunable interfacial barrier height induced by migrating ions. As shown in the top panel of Figure 2b, ions accumulate at the electrode/oxide interface and decrease the barrier height, increasing the device conductance. The interfacial modulating device usually shows analog switching performances, as shown in the bottom panel of Figure 2b, but these devices suffer from poor retention because of the ionic self-diffusion.[53–55] Also, due to the existence of the barrier, the devices show the nonlinear current–voltage relation, which can be used to suppresses the sneak-path current in a passive array.[56] However, the nonlinearity limits the application of the amplitude encoding, because the device conductance is different under various applied voltages.

The channel composition modulation provides another way to tune the resistance of memristive devices, which shows better performances in some aspects.[10] As shown in the top panel of Figure 2c, the conductive channel consisted of two types of migrating ions, then the conductance is determined by the ratio of two types of ions. The conductance can be modulated precisely by variable compliance currents, and the typical I–V relation of a Ta/HfO\(_2\)/Pd device is shown in the bottom panel of Figure 2c.[57] The switching is attributed to the continuous channel composition modulation of Ta and O, whereas the Ta-rich and O-deficient channel shows a high conductance. Due to the large diffusion barrier of Ta\(^{2+}\) in HfO\(_2\), the device exhibited an excellent retention. The shuttling ionic specie should be carefully designed, because the diffusion barrier in the matrix material directly determines the retention property, thus influences the performance of the memNN greatly. In three-terminal transistor-like

---

**Figure 2.** Typical behavioral properties of memristive devices based on various mechanisms. a) Width modulating with high operating speed and abrupt SET process. Reproduced with permission.[115] Copyright 2017, American Chemical Society. b) Interfacial modulation with analog switching and poor retention performances. Reproduced with permission.[134] Copyright 2017, Wiley-VCH. c) Channel composition modulation with high endurance and operating current. Adapted under the terms of the CC-BY Creative Commons Attribution 4.0 International License.[157] Copyright 2016, The Authors, published by Springer Nature. d) Diffusive memristor with volatile properties. Reproduced with permission.[160] Copyright 2016, Springer Nature.
devices, ions with small radii, like protons (H⁺) and lithium ions (Li⁺), are mostly adopted. By controlling the concentration of ions in the channel material through the gate voltage, the resistance of the channel can be tuned accordingly. Thanks to the intrinsic READ–WRITE decoupling nature, these devices can maintain a certain resistance level quite steadily. This kind of device can provide about 500 resistance states and show nearly linear weight updating behavior.

Digital-type memristive devices with volatility, also called as threshold switching (TS) devices, can be attributed to a special type of the width modulation, where the width in an insulating device equals to 0. TS devices are featured with a resistance change by orders of magnitude at a certain voltage point during voltage sweeping. The great change can be attributed to the formation and dissolution of a metallic conduction channel (shown in the top panel of Figure 2d), such as Ag or Cu atoms in silicon oxides, or the thermal-induced Mott phase transition in niobium oxides.[59] The bottom panel of Figure 2d shows the pulse response curve of the TS device of Ag/SiO₂/N₂:Ag/Pd.[60] When a switching pulse is applied, the device requires a finite delay time to switch on and has a finite relaxation time before it recovers back after the switching pulse is removed, yielding a superior 1–V nonlinearity and unique temporal conductance evolution dynamics.[60] These intriguing features pave ways to build novel blocks of electronic circuits and systems, such as access devices in crossbar arrays,[61] artificial neurons with integrate-and-fire functions,[62] and true random number generators.[63]

3. Artificial Synapses

A synapse is a special junction connecting a presynaptic neuron and a postsynaptic one, which has a certain weight. The weight determines the translating efficiency between two neurons, and is modulated according to neuronal activities. Memristive devices with tunable conductance are great candidates for the emulation of synapses due to their similarities in structures and functions; the top electrode and the bottom electrode are connected to the presynaptic neuron and the postsynaptic neuron, respectively, and the device conductance is used to represent the synaptic weight. The output current of a memristive device is decided by the Ohm’s law $I_{\text{output}} = \sigma V_{\text{input}}$, and the conductance $\sigma$ is increased/decreased by SET/RESET operation. However, nonideal effects of memristive synapses, such as the nonlinear and asymmetrical updating, poor retention performance, cycle-to-cycle and device-to-device variations, limit the performance of neural networks. In addition, to realize the brain-inspired computing in SNNs, more biological functions of synapses are needed. At the array-level, sneak-path currents among devices also affect the precision of a neural network.

3.1. Nonideality of Artificial Synapses

As a single memristive device can emulate a synapse, the nonideality of the device, e.g., abrupt SET operation, nonlinear, and asymmetric updating or poor retention, limits the synaptic performances in memNNs. The abrupt SET operation complicates the training process; nonlinear and asymmetric updating decreases the precision in the programming operation, resulting in a low accuracy; whereas poor retention limits the long-time inference process. Therefore, great efforts have been made to overcome the device nonideality.

As for width modulating devices, the analog modulating usually happens in the RESET operation, whereas the abrupt SET operation (shown in Figure 3a) harms the training process of an memNN.[64–66] The abrupt SET process is found to result from the positive feedback of the electric field.[67] and this effect is mitigated at higher temperatures, because multiple weak conductive filaments, instead of a strong one, are formed at a higher temperature. Therefore, a thermal enhancement layer (TEL) TaOₓ was capped on a HfAlOₓ switching layer (Figure 3b) to increase the local temperature under an electrical field. As shown in Figure 3c, the abrupt SET process was then translated into a gradual one.[49,68] The positive feedback was also suppressed by a barrier layer to slow down the ion migration. Woo et al. inserted an Al layer between the bottom electrode and the HfO₂ layer to induce AlOₓ as a barrier layer. The AlOₓ layer limited the migration of oxygen vacancies and prevented the total dissolution of the filament, which resulted in the analog switching in the SET process.[65]

The analog switching in both SET and RESET processes is also not perfect when the conductance modulation is nonlinear or asymmetrical under identical spikes, as shown in Figure 3d. The nonlinearity and asymmetry seriously limit the classification accuracy of neural networks.[69–71] To insert a layer limiting ion diffusion or a TEL is beneficial to get more linear modulation.[72] Wang et al. proposed to insert a diffusion-limiting SiO₂ layer between the top electrode and the TaOₓ layer (Figure 3e) to tune the dynamics of the filament growth and dissolution, which resulted in the optimization of memristive updating performances.[70] As shown in Figure 3f, the increment or decrement in conductance could be modulated linearly, however, as a trade-off the on/off ratio was decreased a little.

When the width of a conductive filament is close to several atoms, the device shows the quantized conductance characteristics. The conductance has a significant linear correlation with the number of atoms in the finest part of the conductive filament.[73] By precise modulation with a scanning probe microscope, at least 16 half-integer multiples of quantized conductance states were achieved.[74] With careful operation, up to 32 consecutive quantized conductance states were implemented, which makes the device a good candidate for analog synapses.[74]

As for interfacial modulating devices, the partially volatile performance (Figure 3g) can be used to emulate the biological synaptic functions, which is explained in Section 3.2, but the volatility has the negative impacts of the inference on ANNs.[75] Therefore, it is critical to find a way to modulate the volatile speed for the neuromorphic computing. Considering that the volatility originates from the ion diffusion, great efforts have been made to control the migration rate of ions. Waser et al. found that SrOₓ was formed at the Pt/SrTiO₃ interface with low ion mobility, inducing a good retention. Xiong et al. fabricated SrTiO₃-based devices with various electrode compositions of Pt and Al (Figure 3h). The existence of Al-generated Al₂O₃ in the top electrode decreases the migration speed of oxygen vacancies. Correspondingly, the volatility was inhibited when increasing the Al/Pt ratio in the bottom electrode (Figure 3i), and the device of Al/SrTiO₃/Nb:Si/SrTiO₃ showed the nonvolatile performance.[77]
Compared with the aforementioned scenarios, the composition modulating is a more practical approach to tune the analog conductance in the performance of retention, linear current–voltage relation, and linear and symmetric updating during training. Jiang et al. proposed a Ta/HfO$_2$/Pt memristor, whose conduction channel consisted of migrating Ta ions and oxygen vacancies, as shown in Figure 3j.[57] By connecting in series, a transistor with tunable gate voltage, the device was operated under various compliance currents, and showed the advantages of fast switching speed, high endurance, and reliable retention, and the conductance could be updated linearly and symmetrically (Figure 3k).[57] Yoon et al. used Ru as mobile species, instead of Ta cations.[78] The switching based on Ru showed a low switching current, fast switching speed, and long retention.
Composition modulations were also observed in three-terminal memristive devices, where the channel conductance was modulated by varying the concentration of doping ions by the gate voltage. For instance, protons (H+) or lithium ions (Li+) were doped in MoO3 to generate the high conductance composition, enhancing the updating performances of linearity and symmetry. [79,80] Similarly, Li+ insertion/desertion in the cathode materials of lithium-ion batteries, such as LiCoO2, induced the modulation of the conductance, showing the multilevel states and the linear and symmetrical updating. [81] Due to the low self-discharge rate of batteries, a battery-like device was proposed to enhance the nonvolatility. As shown in Figure 3l, the gate of poly(3,4-ethylenedioxythiophene):polystyrene sulfonate (PEDOT: PSS) was the cathode and the channel of PEDOT:PSS/PEI was the anode. [82] Under an electric field, protons can transport between the cathode and the anode via the Nafion electrolyte due to a redox reaction, which modulates the conductance of the channel (anode) precisely (Figure 3m). Thanks to the electronic insulating property of the electrolyte, the redox reaction was suppressed without an external electric field, which enhanced the retention performance. To further improve the updating performances, the same material, Li2CoO2, was used in the cathode and the anode to decrease the intrinsic potential. [83] The diffusive memristor in series was able to suppress the volatility induced by the diffusion in the battery. [84]

However, the nonideality of the devices can be neglected in a specific situation. Digital-type memristive devices have only two states of “0” and “1,” which is a large nonlinearity. Nonetheless, it is still possible to use digital-type devices in ANNs, because the synaptic weights in trained neural networks usually have two states. Compared with analog memristive devices, digital ones are advantageous in retention, integration density, uniformity, and compatibility with digital computers. Therefore, the feature of binary states helps to realize a neural network with binary activations and synaptic weights. Compared with analog-type neural networks, digital-type neural networks are simpler, can work more efficiently, while with limited decrease in the network performances. [84–86]

3.2. Updating in Artificial Synapses

Biological synaptic learning rules, such as paired-pulse facilitation (PPF), STDP, SRDP, are essential in SNNs. However, it is difficult to implement these biosinspired functions with high speed and energy efficiency by digital computing. Therefore, many works focus on the emulations of biorealistic functions using the internal ionic dynamics of memristive devices.

Synaptic plasticity reflects a synapse’s response to the pre- and postsynaptic activities, including short-term and long-term plasticities. Different from the LTP, the conductance modulation can only last for a short time (from several microseconds to several minutes) in the short-term plasticity (STP). For instance, the conductance potentiated by several spikes relaxes to the initial state in several seconds (Figure 4a). As a typical STP for the temporal information processing, PPF was implemented in many volatile memristive devices. [75,66,79,87–90] In biology, when paired pulses are applied to a synapse, the second pulse induces a higher response current than the first pulse, and a larger interval between two pulses leads to a smaller enhancement. Similar PPF was observed in volatile memristive devices, as shown in Figure 4b. The peak current induced by the second pulse is larger than the one by the first pulse, and the PPF index, defined as \( A_2/A_1 \), decreases with the increment in the time interval. [92] Stimulating pulses can also induce the short-term depression, which is called pair-pulse depression (PPD). As shown in Figure 4c, the PPD and the transition between PPF and PPD were realized in the Ag/SiO2,Cu/Ag/Pd device, where the dynamics of Ag+ in oxide was similar to that of Ca2+ in biological systems. [40]

The long-term plasticity in biology depends on the timing or rating of connected neurons, that is STDP or SRDP. In most previous works, STDP was realized by overlapping two well-designed spikes applied to two terminals of a memristive device, to translate a specific timing interval into an amplitude of a spike. [50,91] Figure 4d illustrates a method to design the spikes: when the prespike precedes the postspike, the summed spike shows a large positive amplitude, inducing the LTP; when the prespike is after the postspike, LTD is resulted. The overlapping method was used for the associating learning in biology, e.g., Pavlovian conditioning. [92]

A nonoverlapping method was also developed, in which different conductance changes were realized with the internal ionic dynamics of a second-order memristor or with a diffusive memristor in series. Second-order memristors of Ta2O5 had two state-variables: the size of the conducting channel determined the conductance and the temperature modulated the migrating speed of ions. [93] Therefore, an extra heating pulse followed the WRITE pulse could control the second-state variable, as shown in Figure 4e. The temperature was first increased by a heating pulse, and then it decreased over time. As a consequence, the ion-migrating speed at the time of receiving the second pulse was related to the time interval \( \Delta t \) between the two pulses: a smaller \( \Delta t \) corresponded to a higher ion-migrating speed and resulted in more conductance change in the second-order memristor. The demonstrated STDP learning rule is shown in Figure 4e, illustrating the features of the Hebbian learning.

Similar to the second-order memristor with a time-dependent variable of temperature, a diffusive memristor also shows a time-dependent conductance. In the circuit shown in Figure 4f, the voltage over the synaptic memristor is related to the conductance of the diffusive memristor, decided by the time interval \( \Delta t \) between pre- and postspikes. A smaller \( \Delta t \) corresponds to a higher voltage over the synaptic memristor, and results in more conductance change. Figure 4f shows the designed pulse applied to the synapse: the resistance of the diffusive memristor is switched to the ON state under the first pulse, then the resistance gradually increases over time. [60] As the second applied pulse is identical, the voltage over the synaptic memristor depends on the resistance of the diffusive memristor, that is, depends on the time interval \( \Delta t \). Realized STDP rules are shown in Figure 4g, similar to the rule in biology.

Higher-order synaptic plasticity, such as triplet-STDP and metaplasticity, have also been realized with memristive devices. Triplet STDP, according to the neuroscience, shows the interaction between spike pairs to describe the STDP induced by natural complex spike patterns encountered in vivo. As the first-occurring spike suppresses the efficiencies of later spikes in short-term dynamics, the first spike plays a dominating role in the synaptic
Triplet STDP was realized by second-order memristive devices with forgetting effect, e.g., Pt/SrTiO$_3$/Nb-STO and Pt/WO$_3$/Pt. For an instance shown in Figure 4h, the first spike pair of the quadrant IV has a post-pre order and induces LTD, whereas a pre-post order of the second spike pair induces LTP. The net effect of these two spike pairs shows LTD of a large area, clearly indicating the first-spiking-dominating rule.

Metaplasticity, a phenomenon that the change of weights is modulated by an episode of stimulus applied before the learning activity, was also realized in several memristive devices. For instance, as shown in Figure 4i, although the first pulse of 1.7 V × 10 μs does not update the synaptic weight of a Pt/WO$_3$/Pt artificial synapse, the following potentiation process is enhanced significantly by the second pulse.

The SRDP learning rule was also realized in memristive devices with internal dynamics. There is a threshold frequency between LTP and LTD, as shown in Section 2. The threshold is experience-dependent, according to the Bienenstock–Cooper–Munro (BCM) rule. Du et al. demonstrated the BCM learning rule in a second-order memristor of Pt/SrTiO$_3$/W with the spontaneous decay of the low resistance state (LRS) current. The BCM learning rule was also realized in a phosphorus–silicate–glass-based three-terminal device by Ren et al. For a wider range of the threshold modulation, controllable ion dynamics was realized in Pt(Al)/SrTiO$_3$/Nb-SrTiO$_3$ devices with the tunable composition of the top electrode. As shown in Figure 4j, LTD (negative weight change) occurs at low frequencies, whereas LTP (positive weight change) occurs at high frequencies. In addition, the threshold is tuned by the frequency of the priming pulse, the higher frequency of the priming pulse increases the threshold.

3.3. Crossbar Array

The implementation of a hardware-based neural network should address many critical issues, such as the sneak path current,
device-to-device variation, and compatibility with the CMOS process.

For the applications of memristive devices in crossbar arrays, the sneak path current from neighboring cells occurs during WRITE or READ operations, seriously limiting the device operation and large-scale integration.\(^{100,101}\) Many efforts have been made to address the issue, e.g., adding a selector in series, developing a self-rectifying device or a device with nonlinear \(I-V\) relation. To add a transistor is the most practical choice, forming a one-transistor–one-resistor (1T1R) configuration.\(^{68,102}\) The addition of a transistor also benefits the modulation of synaptic weights with the controllable compliance current by a gate voltage. However, the large size of the three-terminal structure of the transistor compromises the scalability of the crossbar structure.\(^{103}\)

Alternatively, one can add a two-terminal selector to form the one-selector–one-resistor (1S1R) configuration. Two-terminal selectors are proposed based on various structures, including Schottky diode,\(^{104}\) multilayer oxide/nitride junctions,\(^{105}\) metal–insulator transition (MIT),\(^{106}\) and TS.\(^{60,61,101,107–110}\) The Schottky diode maintains the OFF state in the cutoff region, therefore, it is only used for unipolar memristive devices.\(^{111}\) Devices based on multilayer oxide/nitride junctions show the gradually varying \(I-V\) curve with high nonlinearity, mitigating the sneak path current.\(^{112}\) The nonlinearity is also realized with the abruptly varied \(I-V\) curve based on MIT or TS.\(^{61,110}\) In the operations of READ or WRITE for the 1S1R configuration, the operating voltage \(V_{R/W}\) is applied to one terminal of the selected cell when the other terminal is grounded, other footprints on the same array are under \(1/2V_{R/W}\). A self-rectifying device integrates a Schottky diode and a memristive device in a single device.\(^{104,111}\) Similarly, memristive devices with highly nonlinear \(I-V\) relation can be used to fabricate a passive crossbar array to avoid the sneak-path current.\(^{56}\)

Device-to-device variation is unavoidable, especially for filament-type devices due to the uncontrollable filament dynamics.\(^{114}\) Therefore, it is critical to control the growth of the filament in a memristive device. The filament is usually formed in the electroforming process, whereas the growth of the filament in electrical field is extremely stochastic, and the “over-forming” causes a high variation.\(^{115}\) Kim et al. realized the electroforming-free device of Pt/Ta/HfO\(_2\)/RuO\(_2\)/Pt by thinning the HfO\(_2\) film down to 3.0 nm, which increased the repeatability and uniformity of the switching performance.\(^{116}\) Choi et al. demonstrated a single-crystalline SiGe epitaxial device with minimal cycle-to-cycle/device-to-device variations.\(^{117}\) They utilized defect-selective etching before electroforming to widen the dislocations providing preferential diffusion paths. As a result, the filaments were restricted to the dislocation pipes, which modulated the filament dynamics precisely. Wang et al. introduced migrating protons into α-MoO\(_3\) through annealing in the H\(_2\)/Ar atmosphere, which also avoided the destructive electroforming process.\(^{118}\) Therefore, the switching behavior was enhanced with high yield and minimal spatial/temporal variations.

Memristive devices have to be integrated with CMOS devices to be implemented in a network, thus a memristive array should be compatible with the CMOS technology. The deposition of the function layer should be compatible with the CMOS process. Until now, large-scale memristive arrays are all deposited by sputtering or atomic layer deposition (ALD), and the foundry compatible materials, e.g., SiO\(_2\), HfO\(_2\), Al\(_2\)O\(_3\), TiO\(_2\), and WO\(_3\), are used.\(^{119–121}\) A widely used electrode material, such as Pt, is not compatible with the CMOS process due to the difficulty of etching. Thus, Ding et al. proposed a TiN electrode to address the issue.\(^{122}\)

4. Artificial Neurons

In contrast to the artificial synapse realized by a single memristive device, artificial neurons are more complex in structure and functions, thus the external circuit is essential. As shown in Section 2.1, ANN accelerators and SNNs use different neurons. A neuron in an ANN accelerator is an activation function in mathematics. However, biological neuronal functions, such as the LIF function, are indispensable in SNNs. The implementation of artificial neurons mostly depends on the CMOS technology, in which many transistors are needed to emulate one neuron. Recently, memristive devices are also used to mimic neuronal functions to realize SNNs efficiently.

4.1. Artificial Neurons Based on CMOS Devices

Artificial neurons in ANN accelerators read currents summarized from the previous layer via synapses and generate stimulating pulses with specific parameters to the next layer. Therefore, an artificial neuron generally includes an analog-to-digital converter (ADC) to read currents, a digital-to-analog converter (DAC) to generate pulses, and a microcontroller to process the information. The functions of reading and generating are realized by the semiconductor parameter analyzer (SPA) with high precision.

Artificial neurons in SNNs are more complex in biological functions; these functions can be realized with conventional CMOS devices as well. Sheridan et al. built a neuron with an operational amplifier (OPA) and a capacitor in parallel.\(^{96}\) Prezioso et al. demonstrated the LIF neuron with two OPAs, one comparator, and an arbitrary waveform generator (AWG).\(^{123}\) The first OPA amplified signals from the synapses and the second one emulated the membrane potential, and the comparator was used for the threshold and the AWG generated the spikes. Jiang et al. proposed a novel threshold-controlled neuron to realize the LIF functions.\(^{124}\) In the operation cycle of the threshold-controlled neuron, the local membrane potential was constant, and the threshold was increased. In the system, only one time-dependent threshold was required, and the neuron with the lowest potential was considered as a winner. Therefore, the utilization of the threshold-controlled neuron decreased the number of time-dependent parameters realized by capacitors. Considering the capacitor area on a chip, the system area was significantly reduced.

4.2. Artificial Neurons Based on Memristive Devices

The circuits of artificial neurons can be simplified with memristive devices, and more biological functions are realized based on the internal dynamics of memristive devices. On the structure
level, a biological neuron consists of ionic channels and a lipid bilayer, as shown in the top panel of Figure 5a; on the function level, the membrane potential of a neuron increases as the neuron receives spikes, and the neuron fires an action potential once the membrane potential is above the threshold, as shown in the bottom panel of Figure 5a.

There are mainly three approaches to realize an artificial neuron. First, a memristive device mimics an ionic channel in the neuronal membrane of the neuron, and the conductance of the memristive device is the that of the ionic channel, whereas capacitors in parallel represent the lipid bilayer. Second, the conductance of a memristive device emulates the membrane potential of the neuron. Third, a memristive device is used for firing the threshold of the neuron, whereas the increasing potential of a capacitor in parallel emulates the integrating function.

A TS memristive device shows a high similarity to a biological ion channel. Based on the equivalent circuit of the HH model, an artificial neuron was realized with two Pt/NbO$_2$/Pt TS devices and two capacitors in parallel (Figure 5b). With the input of current pulses, the voltage over either capacitor was increased gradually; once the voltage was above the threshold of the device, two memristors successively switched to the high conductance state, emulating the successive opening of the ion channels. The successive switching of the two devices generated a neuronal spike (Figure 5c). To increase the frequency of the spikes, carbon nanotubes were added into a VO$_2$-based TS device.

As the local stimulus-induced change in the membrane potential (local graded potential, LGP) is an analog value, the LGP can also be represented by the conductance of a memristive device. For instance, a nonvolatile stochastic phase change device.
used to implement a stochastic neuron with the integrate-and-fire function. As shown in Figure 5d, the electrical stimuli increase the LGP (conductance of the device) until it is over the firing threshold, and then the device is initiated by a RESET operation.[9] Necessary number of stimuli decreases with higher or wider pulses.

When volatile devices are used, the LIF functions can all be realized. Huang et al. fabricated volatile memristive devices of W/WOx/PEDOT:PSS/Pt based on the proton migration.[127] Protons were generated in the PEDOT:PSS bottom electrode under an electric field and moved fast through the WOx layer due to the high proton conductivity. The reaction between protons and WOx induced conductive H2WO3, resulting in an analog switching. A circuit with two W/WOx/PEDOT:PSS/Pt devices can emulate a biological neuron, one device is for the LIF function and the other one for generating HH spikes. As shown in Figure 5e, increased amplitudes or frequencies of the pulses result in the LGP exceeding the threshold, successfully emulating the spatial and temporal integration.

In the neuron consisting of a TS device and a capacitor in parallel (Figure 5f), the LIF is represented by the voltage over the capacitor.[62,128–130] As shown in Figure 5g, charges are accumulated in the capacitor when electric pulses are applied, and leak out in the RESET time; once the voltage over the capacitor is above the threshold of the memristive device, the device switches to the ON-state, resulting in a current pulse due to the discharge of the capacitor. In addition, when the memristive device is stochastic, a stochastic neuron can be realized based on this structure.[131]

Simplified artificial neurons were also proposed: a single device emulated the LIF functions.[132] In a three-terminal structured device, there exists an intrinsic capacitor between the gate electrode and the channel, and the capacitor is used to integrate electric signals. For instance, a dual-gated transistor based on MoS2 mimicked the integrate-and-fire functions, whereas the top gate controlled the threshold and the signals from the bottom gate were integrated to determine the spike firing.[133] The spikes applied on the top gate pushed Li+ ions in the electrolyte toward the channel to decrease the threshold, and the stimulations applied on the bottom gate opened the channel. Consequently, an action potential was propagated.

5. Memristive Neural Networks

To perform artificial intelligence applications, a system has to be implemented by connecting a large number of synapses and neurons on the basis of a specific algorithm. In this section, we first introduce how to demonstrate memNNs based on memristive arrays and artificial neurons, then review the latest progress in ex situ training ANN accelerators, in situ training ANN accelerators, and SNNs.

5.1. Demonstration of memNNs

To conduct tasks in a memNN, several steps should be included: 1) to connect artificial synapses and neurons; 2) to encode information; 3) to train the neural network with the training set and to test the network with the testing set.

5.1.1. Connecting Artificial Synapses and Neurons

CMOS circuit-based memNNs are generally realized with discrete devices on a customer-designed printed circuit board (PCB), with which a memristive array is connected via a probe card or a ceramic chip carrier (Figure 6a). As for an ANN consisted of two types of memristive devices, such as the synaptic ones and the neuronal ones, these devices can be connected directly on a chip. As shown in Figure 6b, eight diffusive memristors (marked with a blue dashed rectangle) were fabricated to emulate neurons, and connected with an 8 × 8 synaptic array (marked with a red dashed rectangle).[62]

As shown in Section 2, there are several algorithms in ANNs and the biggest difference among these algorithms is the configurations of neurons and synapses. To increase the flexibility of the ANN for various algorithms, a memristive array with a large size can be divided into several subarrays. For instance, as shown in Figure 6c, the convolutional operation (fifteen 3 × 3 kernels and four 2 × 2 kernels) and the FC classifier (64 × 10) were all implemented to demonstrate a CNN in a 128 × 64 memristive array.[114] By constructing different subarrays, many ANNs can be realized in the memristive array, demonstrating the flexibility of memNNs.[102,134–137] Moreover, multiplexers (MUXs) are added to choose the connectivity structures between synapses and neurons. For instance, a neuron plays different roles in various modes. In the feed-forward mode, the neuron in a row generates the READ pulses, which needs a DAC; in the back-propagation mode, the neuron receives the stimulus and gets the current value, which needs an amplifier and an ADC; in the programming mode, the neuron needs another DAC or to be grounded, and these modes can be switched with MUXs.[119,121]

Fully integrated chips with memristive array and CMOS circuit further improve the performances. Cai et al. fabricated a complete, integrated hybrid memristor-CMOS system on top of the CMOS circuit (Figure 6d).[121] They integrated a full set of ADCs, DACs, digital bus, memory, and a processor on a chip, to increase the processing speed and the energy efficiency as much as possible. The hybrid system had an enhanced efficiency for the matrix operations based on the crossbar array and the flexibility in implementing various algorithms provided by the CMOS system.

5.1.2. Encoding Information

The information from the real world or the dataset cannot be directly used as the input of memristive arrays. Instead, the information should be first translated into pulses with specific parameters, e.g., amplitude, width, number, timing, and frequency. For instance, the widely used Modified National Institute of Standards and Technology (MNIST) dataset (Figure 6e) includes 8-bit grayscale figures with 28 pixels × 28 pixels, and the grayscale of every pixel is represented by a number from 0 to 255; the encoding schemes for the grayscale are shown in Figure 6f. Correspondingly, output parameters are varied with different encoding schemes: currents of the neurons in the output layer are read in an amplitude encoding ANNs, the accumulated charge for each neuron is...
calculated in width or number encoding ANNs, and the output signals are encoded by timing and frequency in SNNs. The information can be encoded spatiotemporally based on the short-term memristor dynamics. 

In a $4 \times C_2^4$ bitmap shown in the left panel of Figure 6g, each row is represented by an input neuron, whereas each pixel in the row is encoded temporally (right panel of Figure 6g).

Among these parameters, the amplitude, width, and number of the pulses are used in ANN accelerators, whereas the timing and frequency of spikes are used in SNNs. In ANN accelerators,
which parameter is chosen for encoding is dependent on the performances of the memristive array and the training rule. If the devices in the memristive array show the $I$–$V$ linearity, the amplitude of the pulses is chosen, or the width and the number of the pulses are used, thus the power consumption and chip area of the peripheral circuit can be reduced. In SNNs, the training rule also depends on the encoding parameter: the STDP rule is used in a timing-encoding network, the SRDP rule fits for a frequency-encoding network.

### 5.1.3. Training and Testing Neural Networks

The training process is realized by updating synaptic weight according to a specific algorithm, and the flowchart for the training and testing is shown in Figure 6h. There are mainly two algorithms to get the synaptic weights for ANN accelerators and SNNs. In an ANN accelerator, the encoded information in the training set is processed by a memristive array to get a feed-forward result, the differences between which and target results are considered as errors. For instance, ten output neurons are required to classify the MNIST patterns, thus the certain parameter of ten neurons, $O_{\text{processed}} = [n_1 \ n_2 \ \cdots \ n_{10}]$, can be extracted after processing. The neuron with the biggest parameter indicates the classifying result, for example, “2” is classified by the network if $n_2$ is the biggest one among $n_1$~$n_{10}$. In contrast, the target results for the pattern is $\hat{O} = [0 \ 0 \ n_{\text{target}} \ 0 \ \cdots \ 0]$ if “3” is the right result. Then the error is calculated by $\delta = \hat{O} - O_{\text{processed}}$. The errors are then back propagated to every neuron, whereas the change of the synaptic weight is determined by the error and the input to a layer. The training process in the memristive array requires high requirements of endurance and precision: in the ex situ training, the target synaptic weights are calculated by software or simulation, and then the weights are mapped onto the memristive array. In the in situ training, the training process is directly realized on the memristive array. Each step in the training adapts to hardware imperfections, which suppresses the negative impact resulted from the device-to-device variation and the precision issue.

In SNNs, the back-propagated training method is difficult to be used because the neuronal function is not differentiable. Instead, the weight change is determined by two connected neurons according to the STDP or SRDP learning rule. The encoded information is imported to the synaptic array from preneurons, whereas the timing or frequency output is obtained in the postneuron. Then the change of each synaptic weight can be calculated by the timing/frequency in the input and output layers. With several training epochs, the error for the training set can be reduced to a small value, indicating that the training process is finished. Afterward, the information in a testing set is inputted to the SNN, and then the testing results will be obtained.

### 5.2. Ex situ Training in ANN Accelerators

Representative examples of the memNNs based on memristive crossbar arrays are shown in Table 1. In the ex situ training, the massive training process is carried out by software, and the calculated synaptic weights are mapped onto a memristive array. Therefore, the array used in an ex situ training ANN accelerator is requested to be nonvolatile and show high updating precision. Several tasks, e.g., sparse coding, pattern classification, and image recovery, were successfully implemented in ex situ training feed-forward ANNs and RNNs. In contrast, mapping weights onto the array consumes much time, therefore, efforts were made to improve the efficiency of the mapping process.

As for ex situ training feed-forward ANNs, Sheridan et al. demonstrated the sparse coding in a $32 \times 32$ WO$_3$ memristive array with lateral neuron inhibition. The inhibition prevented multiple neurons from representing the same pattern and was beneficial to the optimized output. They trained different dictionary sets in software, and stored all results in the same array. Based on the learned dictionary, they conducted the natural image processing. Hu et al. mapped the weights onto a $128 \times 64$ array of 1T1R cells for classifying the handwritten digits in the MNIST dataset. Each input neuron represented a pixel of the image in the dataset, and each output neuron stood for a number from 0 to 9. The input data were encoded by pulse amplitudes and the highest current response of the output neuron indicated the input number. Due to the high VMM accuracy equivalent of 6 bits, a recognition accuracy of 89.9% was achieved.

As for ex situ training RNNs, Mahmoodi et al. realized versatile stochastic dot product circuits based on a $20 \times 20$ passive stochastic memristive array. The stochastic dot-product operation could be used in the simulating annealing algorithms, which updated the neuron activations in a network with fixed weights to find the minimum energy (the lower energy meant a better optimized state), thus they demonstrated stochastic HNNs with a $64 \times 64$ array for the annealing algorithms. With the help of the simulating annealing algorithms, the HNN successfully solved combinatorial optimization problems, i.e., weighted maximum-clique problems, weighted vertex cover problem, independent set problem, and graph partitioning optimization problem. For another case of the ex situ training ANNs, please refer to the study by Zhou et al.

The high-precision mapping process is usually done cell-by-cell, which is very time consuming. To accelerate the mapping process, a parallel programming scheme with high precision was demonstrated by Chen et al. They programed conductance row-by-row with incremental gate voltage in IT1R arrays, when the SET voltage remained always unchanged. They achieved a recognition accuracy with $<0.3\%$ accuracy loss. By fully integrating the array with peripheral circuits on the same chip, the energy efficiency or computing speed was further enhanced.

### 5.3. In Situ Training in ANN Accelerators

To pave the way for the applications of memNNs, more researches have been focused on the in situ training. Compared with the ex situ training ANN accelerators, the in situ training ones train the network on chip, avoiding the mapping process, but high endurance performances are required for the synaptic devices due to the frequent SET/RESET operation during training process. In the in situ training ANN accelerators, feed-forward ANNs were demonstrated from FC neural networks to CNNs, showing improved ability for pattern classification.
Table 1. Memristive arrays and applications.

| Array image | Array/Cross-point size | Device Features | Peripheral circuitry | Encoding scheme | Connectivity of ANN | Application demonstration |
|-------------|------------------------|----------------|----------------------|-----------------|---------------------|--------------------------|
|             | 32 x 32                | Pd/WO/ /W       | Passive              | Pulse width     | Single layer        | Sparse coding            |
|             | 500 x 500 nm²          |                | Chip Carrier + off-chip MCU with ADC/DAC |                  |                     |                          |
|             | 128 x 64               | Ta/HfO/Pt or Pd | 1T1R                 | Pulse amplitude | SLP                 | MNIST classification     |
|             | 4 x 4 µm²              |                | Probe card + off-chip MCU with ADC/DAC |                  |                     |                          |
|             | 20 x 20                | Pt/Ti/TiO/Al₂O/Pt | Passive            | Binary          | HNN                 | Stochastic and chaotic simulated annealing, exponential annealing |
|             | 200 x 200 nm²          |                | Chip carrier + Switch matrix + SPA |                  |                     |                          |
|             | 128 x 64               | Ta/HfO/Pt or Pd | 1T1R                 | Pulse amplitude | MLP                 | MNIST classification     |
|             | 4 x 4 µm²              |                | Probe card + off-chip MCU with ADC/DAC |                  |                     |                          |
|             | 128 x 16               | TiN/Ta₂O/HfO/TiN | 1T1R               | Pulse width     | CNN                 | MNIST classification with CNN |
|             | 500 x 500 nm²          |                | Chip carrier + on-chip MCU with ADC, MUX + off-chip MCU with DAC |                  |                     |                          |
|             | 128 x 64               | Ta/HfO/Pt or Pd | 1T1R                 | Pulse amplitude | LSTM RNN            | Identifying human by gait / MNIST-sequence video classification |
|             | 4 x 4 µm²              |                | Probe card + off-chip MCU with ADC/DAC |                  |                     |                          |
|             | 54 x 108               | Au/Pd/WO/W/Au   | Passive              | Pulse width     | Single/multiple layer | Perceptron network, sparse coding and principal component analysis |
|             | 500 x 500 nm²          |                | On-chip MCU with ADC, DAC and MUX |                  |                     |                          |
|             | 10 x 10 µm²            | Synapse:       | 1T1R                 | Pulse amplitude | /                   | Pattern classification   |
|             | Synapse:               |                | Probe card + off-chip MCU with ADC/DAC |                  |                     |                          |
|             | 8 x 8                  | Ta/HfO/Pd      | 1T1R                 | Pulse amplitude | /                   | Pattern classification   |
|             | 4 x 4 µm²              | Neuron:        |                       |                  | /                   | Pattern classification   |
|             | Neurons:               | Pt/Ag/SiO₂/Ag/Pt |                   |                  | /                   | Pattern classification   |
|             | 8 x 1 x 1              | NA             |                      |                  | /                   | Pattern classification   |

Reproduced with permission,[56,62,119–121,137] Copyright 2017, 2018, 2020, 2018, 2020, 2019, respectively, Springer Nature.
The demonstration of RNNs promoted the classification of a sequence of patterns. Except for the classification application, the task-system controlling, feature extraction, dimensionality reduction, and data clustering were also realized with ANNs based on memristive arrays, which were also considered as the unsupervised learning because samples were not labeled.

FC neural networks were demonstrated in hardware to realize the classification applications, and most of researchers focused on the MLP.[120] In an active 1T1R memristive array, multiple layers were demonstrated on a single $128 \times 64$ array.[136] The researchers achieved 91.71% accuracy on the complete 10,000-image testing set of MNIST, and the network showed the self-adapting tolerance to hardware imperfections. Nevertheless, the increment in the complexity of the neural network needed more ADC and DAC, consuming lots of power and space, thus a new circuit architecture, the binary hidden neuron layer, was promoted.[148] In this approach, the original outputs of the neurons in the hidden layer were compared with 0 during inference, thus the outputs were quantized into 0 and 1, whereas the degradation of the accuracy was only 15%.

As shown in Section 2, CNN is very successful in image recognition, and the convolutional computing can be demonstrated with memristive crossbar arrays.[149–151] Dong et al. presented a specific circuit for the CNN with binary or multilevel memristive devices.[152] One kernel was represented by two rows of memristive devices and eight output currents were pooled and activated simultaneously for one value, which was regarded as the input of the FC layer. The accuracies for classifying MNIST handwritten digits with the binary devices and the multilevel devices were 97% and 98%, respectively. Yao et al. fabricated a processing element (PE) unit on a chip which integrated several functional blocks (i.e. input/output registers, MUXs, ADCs, shift, and add control units) and a $128 \times 16$ memristive array with high yield, performance, and uniformity.[119] The PE unit could be divided into subarrays for several small convolutional layers, two PE units could also be used cooperatively for a large matrix of a convolutional layer or an FC layer. With the help of a customized PCB, a five-layer CNN were fully implemented with eight PE units, six units of which were for the convolutional layer and the rest for the shared FC layer. The PCB was consisted of dynamic random-access memory (DRAM) block, advanced RISC machine (ARM) core, configured circuits, and voltage generator to demonstrate digital computing, such as accumulator, activation function, pooling function, and calculating updates. Handwritten digits in the MNIST dataset were classified by the CNN, in which the data were encoded by the pulse number. They also proposed an effective hybrid training method to suppress the imperfections and avoid highly complex operation. The weights of the convolutional layers were transferred from the ex situ training results and kept unchanged, whereas the weights of the FC layer were updated after the weight transfer.

Compared with feed-forward ANNs, RNNs own the ability to analyze temporal sequential data. LSTM networks are a type of RNNs using LSTM units, in which the functions of remembering and forgetting are realized by recurrently connecting the nodes in the same layer and doing VMM operation. Therefore, based on a $128 \times 64$ memristive array, Li et al. built a multilayer RNN, which included an LSTM layer and an FC layer, for identifying individual persons by the gait.[117] With the combination of convolutional functions, RNNs with the LSTM units showed enhanced performance for featuring spatial and temporal input simultaneously, such as classifying the MNIST-sequence videos with high accuracy.[114] These works integrated different configurations of the neural networks on the same chip, which was advantageous to minimize transferred data, reduce inference latency, and enhance power consumption efficiency.

As for the unsupervised learning, Cai et al. fabricated a fully integrated memristor–CMOS system with a $54 \times 108$ passive memristive array of $\text{WO}_x$, a full set of analog/digital interface blocks and a digital processor.[121] The integration of all elements on one chip not only demonstrated the sparse coding algorithm and the principle component analysis (PCA, for reducing the data dimension), but also enhanced the speed and the power efficiency. For other cases of the in situ training ANNs, please refer to the previous studies.[68,135,153–156]

5.4. SNNs

SNNs are advantageous in power efficiency, because the neurons of SNNs generate spikes only when the membranes are over the threshold. Although the implementation of SNNs still faces large challenges, e.g., lack of mature training algorithms and requirement for complex external circuits, several emerging works based on memristive arrays have been proposed. On the algorithm level, the STDP/SRDP learning rule in biology is a bit complex for the hardware realization. Therefore, the STDP learning rules are simplified, where the LTP only occurs when the prespikes precede the postspikes within a certain time interval (5 ms), whereas the LTD occurs at the other time intervals.[157,158] The function of the lateral inhibition is added in the output neurons, only one neuron can be activated in one pattern (or called winner-take-all [WTA]). The lateral inhibition is widely deployed in SNNs, which mitigates the locality of STDP/SRDP and make the self-adapting network energy efficient.[62,124,159]

Consequently, new types of neurons also benefited the implementation of SNNs. Jiang et al. demonstrated the pattern recognition in a TiN/HfO$_x$/AlO$_x$/Pt array with threshold-controlled neurons.[124] In their system, the training and the classification process were separated, and the recognition accuracy achieved $\approx90\%$ with 10% noise. Wang et al. integrated LIF neurons based on diffusive memristors with an $8 \times 8$ memristive array on the same chip, building a fully memNN.[62] The memNN performed the STDP learning rule with the unsupervised weight updating, and achieved the pattern classification. The STDP learning rule was also improved for adapting the synapses. To work with large conductance variations in artificial synapses, a network with dilute encoding spike events were proposed by Guo et al., which achieved an accuracy of 75% for MNIST handwritten digits.[159]

For the other cases of the SNNs, please refer to the previous studies.[123,157,160–162]

6. Conclusion

Memristive devices are essential for realizing ANNs with high speed and energy efficiency, due to their versatile performances for emulating synaptic and neuronal functions, simple crossbar structure for the VMM operation, and CMOS compatibility for
mass production. According to the connectivity structure and
the encoding scheme, ANNs are classified into feed-forward
ANNs, RNNs, and SNNs, which require devices with versatile
performances. Great efforts have been made to enhance the anal-
og properties of filamentary and interfacial devices for artificial
synapses, which paves the way for the array integration. Due to
the specific internal dynamics of memristive devices, synaptic
and neuronal functions in biology are realized in a simplified
method. Moreover, experimentally implemented memNNs
have been applied for various datasets (e.g., MNIST), showing
an energy efficiency that is orders of magnitude higher than
the ANNs based on conventional CMOS technology.\[119]\ With
the widespread application of ANNs in the image processing
and the decision making, a specially designed block for ANNs
has been adopted. The high efficiency based on memristive arrays
enables the edge computation in mobile instruments and enhances
the computing ability of data center in the era of big data.

However, memNNs are just in the early developmental stage,
and more complicated tasks need a large size of arrays. Many chal-
enges still need to be tackled to promote the development of
ANNs based on memristive devices. On the device level, although
synaptic devices with long retention and analog switching can be
fabricated, linear and symmetrical weight updating, efficient
power consumption should still be further pursued, e.g., by reduc-
ing the READ/SET/RESET current. Moreover, some optimization
methods in ANNs can be realized by means of the specific perform-
cance of memristive devices, e.g., the function of dropout neurons
during training helps to prevent the overfitting.\[160\] In addition,
switching endurance, cycle-to-cycle/device-to-device variations and
the compatibility with the CMOS process require more attention.

On the system level, the peripheral circuit still consumes large
area and high energy, because a memNN is a mixed-signal
system requiring digital–analog converting blocks. Considering
that information is encoded in the analog mode in the real world,
sensory neurons are proposed recently, which will greatly reduce
the complexity to process analog signals from sensors. Some sen-
sory neurons based on memristive devices, such as light-
responding devices, were proposed to combine the functions of
sensing and processing, which is more similar to a biological
system.\[164,165\] The performance and power consumption of a neu-
ral network can be further optimized by emulating some biologi-
cally neuronal functions (e.g., dendritic computing functions) with
memristive devices.\[166\] In addition, new architectures, such as
the Bayesian neural network\[167\] and the generative adversarial net-
work\[168\] should also be demonstrated for more useful functions
by memristive arrays. In realizing SNNs, memristive devices are
advantageous over CMOS devices, however, it is difficult to train a
SNN for a complex task with the STDP or SRDP learning rules,
therefore, a mature learning algorithm is urgently awaited in
SNNs by learning from biological systems. In algorithm, other
solutions to train SNNs are to use back-propagation algorithms
based on spikes or with an ANN–SNN converter, which can also
be referred by memNNs.\[160–171\]

**Acknowledgements**

This work was supported by the National Key Research and Development
Program of China (grant no. 2018YFE0203802) and the HUAWEI Project
(YBN2019055139).

**Conflict of Interest**

The authors declare no conflict of interest.

**Keywords**

artificial neural networks, in situ/ex situ training, memristive devices,
neuron, synapses

---

**References**

1. N. E. Karkalos, A. P. Markopoulos, *Computational Methods and
Production Engineering*, Woodhead Publishing, Duxford 2017.
2. D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. van den
Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam,
M. Lanctot, S. Dieleman, D. Grewe, J. Nham, N. Kalchbrenner,
I. Sutskever, T. Lillicrap, M. Leach, K. Kavukcuoglu, T. Graepel,
D. Hassabis, *Nature* 2016, 529, 484.
3. D. Silver, J. Schrittwieser, K. Simonyan, I. Antonoglou, A. Huang,
A. Guez, T. Hubert, L. Baker, M. Lai, A. Bolton, Y. Chen, T. Lillicrap,
F. Hui, L. Sifre, G. van den Driessche, T. Graepel, D. Hassabis,
*Nature* 2017, 550, 354.
4. J. Von Neumann, P. M. Churchland, P. S. Churchland, *The Computer
and the Brain*, Yale University Press, New Haven, CT 2000.
5. W. Maass, *Neural Netw.* 1997, 10, 1659.
6. M. A. Zidan, J. P. Strachan, W. D. Lu, *Nat. Electron.* 2018, 1, 22.
7. R. Yang, H. M. Huang, X. Guo, *Adv. Electron. Mater.* 2019, 5, 1900287.
8. W. Zhang, B. Gao, J. Tang, X. Li, W. Wu, H. Qian, H. Wu, *Phys. Status
Solidi RRL* 2019, 13, 1900204.
9. T. Tuma, A. Pantazi, M. Le Gallo, A. Sebastian, E. Eleftheriou,
*Nat. Nanotechnol.* 2016, 11, 693.
10. A. D. Patil, H. Hua, S. Gonugondla, M. Kang, N. R. Shanbhag, in 2019
*IEEE Int. Symp. on Circuits and Systems (ISCAS)*, IEEE, Piscataway, NJ 2019, p. 1.
11. S. Dutta, C. Schafer, J. Gomez, K. Ni, S. Joshi, S. Datta, *Front. Neurosci.*
2020, 14, 634.
12. B. J. Choi, A. C. Torrezan, J. P. Strachan, P. G. Kotula, A. J. Lohn,
M. J. Marinella, Z. Li, R. S. Williams, J. J. Yang, *Adv. Funct. Mater.*
2016, 26, 5290.
13. J. Zhou, F. Cai, Q. Wang, B. Chen, S. Gaba, W. D. Lu, *IEEE Electron
Device Lett.* 2016, 37, 404.
14. S. Pi, C. Li, H. Jiang, W. Xia, H. Xin, J. J. Yang, Q. Xia, *Nat.
Nanotechnol.* 2019, 14, 35.
15. L. Jiang, F.-C. Lv, R. Yang, D.-C. Hu, X. Guo, *J. Materiomics* 2019, 5,
296.
16. T. Shi, J.-F. Wu, Y. Liu, R. Yang, X. Guo, *Adv. Electron. Mater.* 2017, 3,
1700046.
17. X.-B. Yin, R. Yang, K.-H. Xue, Z.-H. Tan, X.-D. Zhang, X.-S. Miao,
X. Guo, *Phys. Chem. Chem. Phys.* 2016, 18, 31796.
18. A. Serb, A. Khiat, T. Prodromakis, *IEEE Trans. Electron Devices* 2015,
62, 3685.
19. B. Zhang, F. Fan, W. Xue, G. Liu, Y. Fu, X. Zhuang, X.-H. Xu, J. Gu,
R.-W. Li, Y. Chen, *Nat. Commun.* 2019, 10, 736.
20. S. Goswami, S. P. Rath, D. Thompson, S. Hedström, M. Annamalai,
R. Pramanick, B. R. Ilic, S. Sarkar, S. Hooda, C. A. Nijhuis, J. Martin,
R. S. Williams, S. Goswami, T. Venkatesan, *Nat. Nanotechnol.* 2020,
15, 380.
21. H.-K. He, R. Yang, W. Zhou, H.-M. Huang, J. Xiong, L. Gan, T.-Y. Zhai,
X. Guo, *Small* 2018, 14, 1800079.
He-Ming Huang is currently working toward a Ph.D. degree in School of Material Science and Engineering, Huazhong University of Science and Technology (HUST). His research topic is memristive devices and their neuromorphic applications.
Zhe Wang is currently working toward a Ph.D. degree in School of Material Science and Engineering, Huazhong University of Science and Technology (HUST). His research topic is memristive devices and their neuromorphic applications.

Xin Guo is a distinguished professor at School of Materials Science and Engineering, Huazhong University of Science and Technology (HUST), Wuhan, China. Before joining HUST in 2012, he worked at the Research Center Juelich, Germany, from 2002 to 2012, and at the Max Planck Institute for Solid State Research, Stuttgart, Germany, from 1998 to 2002. In 1992, he received his Ph.D. from HUST. His current research focuses are: neuromorphic devices and systems, gas sensors and artificial olfactory, solid-state batteries, defects in complex oxides, 3D printing of functional devices.