Towards Energy-Efficient Computing Hardware Based on Memristive Nanodevices

COMPUTING HARDWARE IS ONE of the crucial drivers of artificial intelligence (AI) that impacts our daily lives. However, despite the significant improvements made in recent decades, the energy consumption of computing hardware that powers AI, especially deep neural networks, remains considerably higher than that of human brains. Hardware innovations based on emerging nanodevices like memristors offer potential solutions to energy-efficient computing systems. This review discusses the challenges associated with developing energy-efficient computing hardware based on memristive nanodevices and summarizes recent progress in memristive devices, crossbar

Digital Object Identifier 10.1109/MNANO.2023.3297106

YI HUANG, VIGNESH RAVICHANDRAN, WUYU ZHAO, AND QIANGFEI XIA

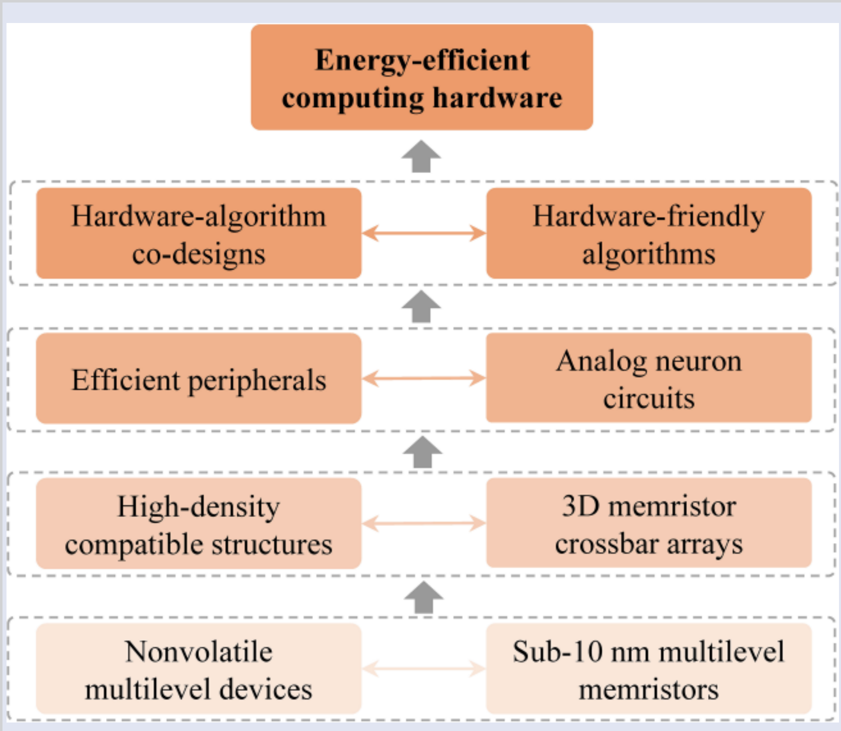

FIGURE 1 The requirements (left) for energy-efficient computing hardware and our corresponding research (right) at different levels, from device study to algorithm development.

arrays, systems, and algorithms, aiming at addressing these issues from a bottom-up approach. Potential research directions are proposed to further improve future computing hardware's energy efficiency.

INTRODUCTION

The energy efficiency of computing systems is becoming increasingly important with the development of technologies such as artificial intelligence (AI) [1], the Internet of Things (IoT) [2], and autonomous robotic agents [3]. Meanwhile, the energy consumption of training AI models has sharply risen due to the tremendous increase in their parameters, complexity, and training data size [4]. On the other hand, with the increasing demand for edge devices to perform complex tasks, low power consumption has become a critical requirement for the hardware that deploys data-intensive computing to facilitate AI algorithms [2]. However, traditional computing systems are facing a significant challenge, the von Neumann bottleneck. This issue arises because of constant data transfer between memory and processing unit, which limits the speed and energy efficiency. Another related challenge is the performance mismatch between the memory and processing units, referred to as the memory wall issue. These challenges hinder the further improvement of computing systems under the von Neumann architecture. To solve these problems, in-memory computing hardware, which allows computing to be performed at the same location where data is stored, has been explored from device to system levels and proved promising as the next-generation computing scheme [5], [6], [7].

At the device level, memristive devices have emerged as leading candidates for in-memory computing because of their unique characteristics [8], [9], [10], [11]. These devices can store information as conductance values because their internal states can be modified by the voltages/currents applied to them. In addition, memristive devices require no energy to hold their resistance, making them ideal as basic components of in-memory computing [12]. At the structural level, the small area and fast switching speed of memristive devices enable them to be organized as dense crossbar arrays or stacked as three-dimensional arrays to implement vector-matrix multiplications (VMMs), the most common operations in deep neural networks, in highly parallel fashion [13]. At the system level, memristive devices and crossbar arrays empower analog in-memory computing to avoid data movement and digital-analog conversions from analog sensors and peripherals. Furthermore, they are used as neural circuits in brain-inspired computing, which seeks to develop computational systems inspired by the human brain's structure, function, and learning mechanisms [14], [15], [16]. As a result, the broad spectrum of innovations stemming from memristive devices can potentially improve the speed, area, and energy efficiency of future computing systems and revolutionize high-performance computing. Despite the potential benefits, there are still challenges from the device to the system levels to fully utilize memristive nanodevices in in-memory computing for energy-efficient hardware systems.

In this review, we discuss the requirements for memristive hardware at different levels to achieve efficient in-memory computing. We also present our recent efforts in addressing some of these essential requirements, as illustrated in Figure 1. Furthermore, we propose potential research directions that can further enhance the energy efficiency of hardware systems with the continued advancements of memristive technology.

REQUIREMENTS FOR EFFICIENT IN-MEMORY COMPUTING

MEMRISTIVE DEVICES

As building blocks for in-memory computing hardware, a long list of properties of memristive devices must be considered, including the number of distinguishable conductance states, retention, endurance, device dimension, switching speed, and switching energy [17]. The number of distinguishable conductance states in a single memristor is important to achieve high storage density and computing precision, as it represents the bits of data that can be stored in one device. Stable retention and high endurance of each device are also crucial for long-term data storage and consistent computing results when switching memristive devices to multilevel states, given that

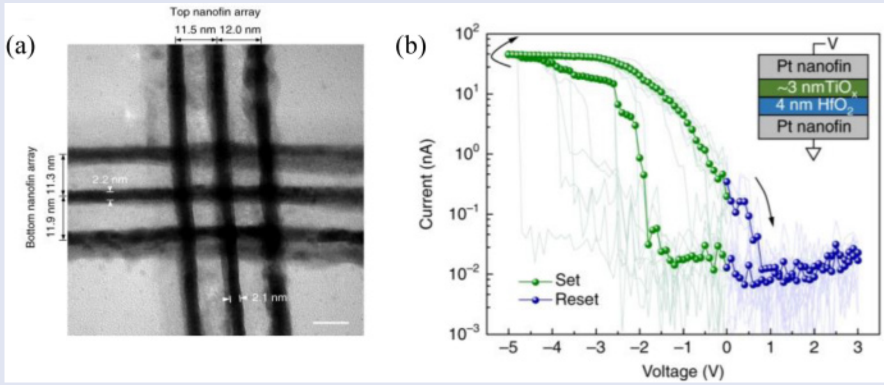

FIGURE 2 (a) Transmission electron microscopy (TEM) of the 3 × 3 memristor crossbar array with 2 × 2 nm^2 device area and sub-12-nm pitch. Scale bar: 10 nm. (b) The I–V curve for the 2-nm memristor in the array. Inset shows the materials stack and measurement voltage polarity of the memristors [23].

memristive devices are used as nonvolatile memory for in-memory computing. Properties related to hardware performance also need research attention, in addition to the requirements for in-memory computing. The nanoscale device dimension is important for integrating memristive devices into crossbar arrays and reducing total chip area. Fast switching speed and low switching energy to achieve different levels are preferred to minimize latency and power consumption and improve the overall throughput of the computing system.

MEMRISTOR CROSSBAR ARRAYS

Addressing sneak path current is the primary challenge when integrating memristive devices into crossbar arrays for parallel VMMs. Sneak path current occurs when the current flows through unintended memristors in the array, leading to inaccurate reading and programming of memristors in most passive crossbar arrays. Therefore, the ability to suppress the sneak path current is the basic requirement for designing memristor crossbar arrays. Also, high device density without sacrificing performance is critical for achieving high throughput in in-memory computing. Moreover, the flexibility of memristor arrays to adapt to various computing schemas for different applications needs to be considered. Beyond the array structure, compatibility with complementary metal–oxide–semiconductor (CMOS) technology is essential since most peripherals facilitating memristor crossbar arrays are designed based on CMOS circuits.

PERIPHERALS

The rapid progress in memristive devices and crossbar arrays has significantly improved on-chip computing performance regarding throughput and energy efficiency [18]. But off-chip peripherals responsible for more than 90% of the area, latency, and energy consumption of the whole computing system [19], have become a primary obstacle to developing efficient in-memory computing systems. These peripherals mainly perform analog-digital conversions and critical functions other than VMMs in digital computing, such as data movement and high-precision calculations. To address this challenge, a fully analog hardware implementation is necessary for peripherals used in memristive hardware. This eliminates the need for power-hungry digital-to-analog converters (DACs) and analog-to-digital converters (ADCs). Furthermore, developing peripherals with low latency and high energy efficiency for signal transmission and data movement is crucial to keep pace with the high data throughput caused by advancements in memristive devices and arrays and enhance the overall performance of in-memory computing hardware.

HARDWARE-ALGORITHM CO-DESIGNS

Compared to traditional digital computers, memristive in-memory computing hardware differs in many aspects, including computing with natural physical laws, analog information encoding, signal transmission, etc. [20], [21], [22] These differences require hardware-algorithm co-designs to develop energy-efficient computing systems based on memristive devices. On the hardware side, because of the highly parallel computing schema, high-throughput interfaces, efficient data movement approaches, and reconfigurable architectures are key requirements to consider when designing circuits and systems for different algorithms and applications. On the software side, algorithms initially designed for high-precision digital computing must be adjusted or redeveloped to fit the relatively low-precision computing in the analog domain without sacrificing performance. Also, algorithms that can effectively tolerate or even leverage the non-idealities of memristive devices should be designed for in-memory computing hardware rather than attempting to eliminate the non-idealities from devices. Hence, hardware-algorithm co-designs complementing each other are critical to developing future energy-efficient computing systems that enable various machine learning and brain-inspired algorithms.

SUB-10 NANOMETER AND MULTILEVEL MEMRISTOR DEVICES

At the device level, two breakthroughs including small device dimensions and 2048 conductance levels are discussed. These devices are measured with remarkable single-device performance,

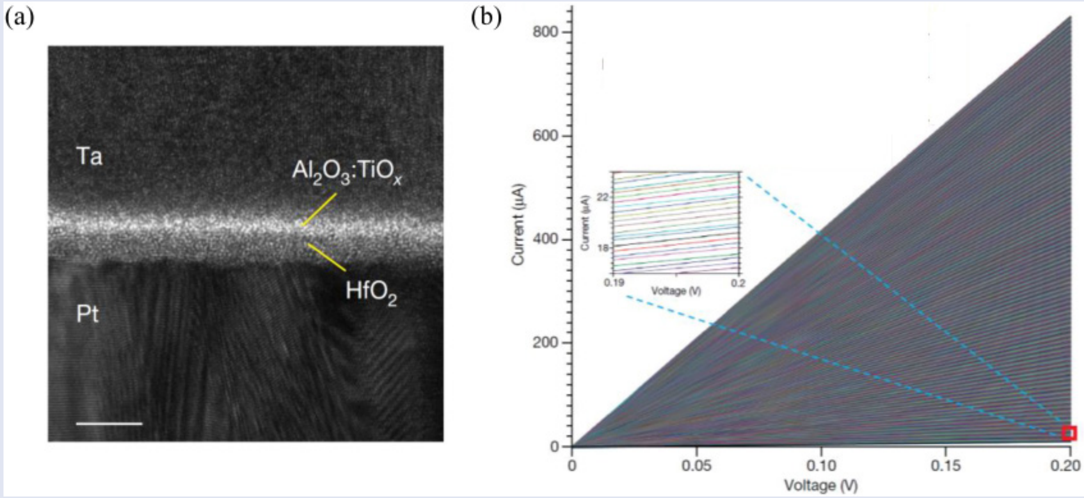

FIGURE 3 (a) TEM of a Ta/Al_2O_3/HfO_2/ Pt memristor device. Scale bar: 5 nm. (b) 2048 conductance levels were measured by off-chip driving circuitry. Inset, zoom-in part of the measured conductance levels [25].

integrated into crossbar arrays, and experimentally proved for information storage and data processing.

Achieving sub-10 nanometer(nm) memristive devices that can be integrated into high-density crossbar arrays is a crucial step toward reducing chip area. However, the fabrication of highly ordered conductive nanoelectrodes remains a significant challenge. In a recent study [23], metal nanostructures with ultrahigh height-to-width aspect ratio were first proposed and demonstrated as electrodes of memristor crossbar arrays. This approach enabled the fabrication of memristor arrays with a 2-nm feature size and a 6-nm half-pitch. A 3×3 crossbar array using the proposed nanostructure and a Pt/TiO_x/HfO_2/ Pt stack was fabricated (Figure 2). The resulting 2-nm memristors demonstrated a high dynamic range, with an average ON/OFF ratio of 454, while exhibiting bipolar nonvolatile switching behavior. In addition, the switching current of the 2-nm memristor was 46 nA, leading to low programming power consumption. This 2-nm memristor and the 3×3 crossbar array achieved extreme scalability and high energy efficiency, providing promising potential for future advancements in memristive devices and crossbar arrays [24].

In addition to area scalability, one crucial factor for memristive devices used in computing systems is the number of conductance levels that can be achieved in a single device. More conductance levels result in high computing precision and benefit the chip area and power consumption, since more bits of data can be stored in a single device. While theoretically, a memristor device is analog and can be tuned to an infinite number of conductance levels, in practice, fluctuations at each conductance level impose a constraint on the number of distinguishable levels that can be attained within a particular range of conductance values. Recent research revealed that the regular switching operation, whether SET or RESET, inevitably results in incomplete conduction channels in the form of either islands or blurred edges adjacent to the primary conduction channel. These secondary channels are less stable than the main conduction channel [25]. Therefore, a denoise process was developed for programming the Ta/Al_2O_3/ HfO_2/ Pt memristor devices. By applying small voltage pulses with optimized amplitude and width, the denoising process substantially reduced the fluctuation and tuned the memristor device to 2048 conductance levels. The device stack and measured 2048 conductance levels are shown in Figure 3, where the conductance levels were read by sweeping D.C. voltages from 0 to 0.2 V, with target conductance ranging from 50 μS to 4144 μS, and a 2-μS between adjacent levels. The memristor devices were also integrated with CMOS circuits in 256 × 256 one-transistor-one-memristor (1T1R) crossbar arrays in a commercial chip foundry, proving the potential of memristors in future computing systems with CMOS-based peripherals.

Developing small and multilevel devices with satisfactory retention, endurance, and switching speed provides a solid foundation for integrating memristive devices in high-density crossbar arrays and paves the way for further exploration of memristive hardware in in-memory computing systems.

THREE DIMENSIONAL MEMRISTOR CROSSBAR ARRAYS

Based on the progress made in memristive devices, high-density crossbar arrays were constructed by extending the two-dimensional (2D) to three-dimensional (3D) to increase the computing throughput and packing density. Structural designs using the extra dimension were also explored to mitigate the sneak path current issue and enhance the flexibility of crossbar arrays for computing.

In an early demonstration [26], 3D crossbar arrays were fabricated with self-rectifying memristors based on silicon, enabling compatibility with the CMOS foundry process. Figure 4 depicts the device stack of the self-rectifying memris-

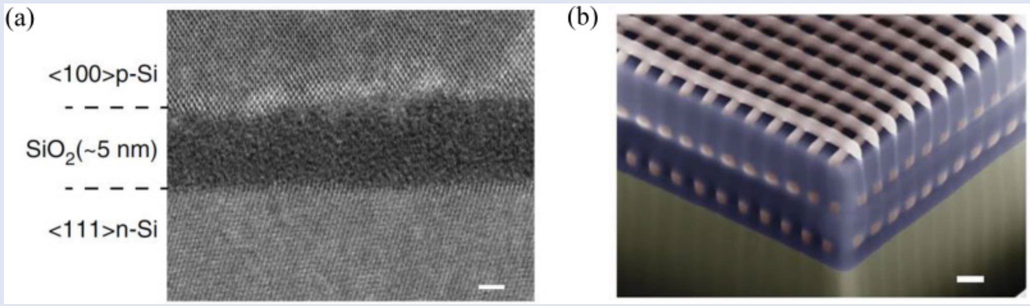

FIGURE 4 (a) TEM of a $Si/SiO_2/Si$ memristor. Scale bar: 2 nm. (b) The 3D stacked crossbar array with the all-silicon-based devices and isolation between different layers [26]. Scale bar: 200 nm.

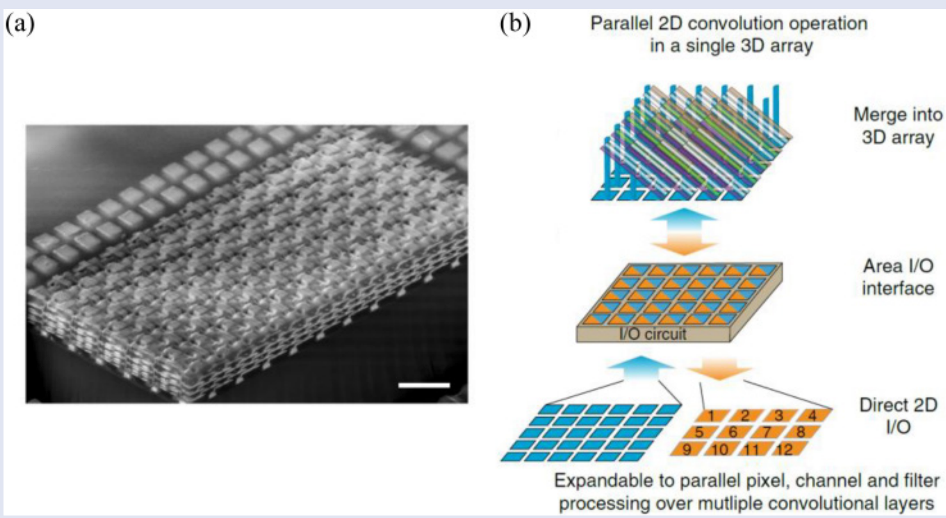

FIGURE 5 (a) Scanning electron micrograph (SEM) of the 3D memristive circuit with eight layers of monolithically integrated memristors. Scale bar: 2 μm. (b) The diagram of the implementation of parallel 2D convolution operations in all the row banks of the 3D array [27].

tors and the corresponding 3D crossbar arrays. The top and bottom electrodes of the $Si/SiO_2/Si$ memristor devices were made of p-type and n-type doped single crystalline silicon, which were transferred from fluid-supported membranes. The switching layer of the device was a thin layer of silicon oxide produced chemically, and its switching mechanism was verified experimentally as the formation and rupture of a sub-5 nm silicon-rich conduction channel within the oxide layer. The self-rectifying unipolar memristors, possessing the intrinsic diode effect, effectively suppressed intra- and inter-layer sneak path currents, as all possible sneak paths in the crossbar array involved at least one reversely biased cell. This presented the feasibility of electrical operations of 3D memristor arrays without external selectors. The 3D crossbar arrays with silicon-based memristors demonstrated that the 3D stack used a simplified fabrication process and can be integrated with CMOS-based peripheral circuits for further development as memory or computing components.

Other than the direct extension from 2D to 3D crossbar arrays, interconnect designs between layers using the additional dimension in 3D arrays were also explored, allowing for more flexibility to accommodate different in-memory computing schemas. As shown in Figure 5(a), a 3D circuit composed of eight layers of monolithically integrated memristive devices was fabricated to use the 3D structure to directly map and implement complex neural networks in hardware [27]. In conventional 2D crossbar arrays, 2D image pixel matrices from many neural network applications needed to be unrolled to 1D vectors to fit the input rows of the crossbar array. This costs extra power consumption and limits the throughput of in-memory computing. In the purposely designed 3D array, memristors in each row bank were used as the weights of convolutional kernels in a convolutional neural network (CNN). As shown in Figure 5(b), the row banks constructed the 3D array for a 2D convolution operation without unrolling the 2D inputs. This design takes 2D inputs for convolutions and enables bilateral 2D data communications between input/output (IO) circuits and periph-

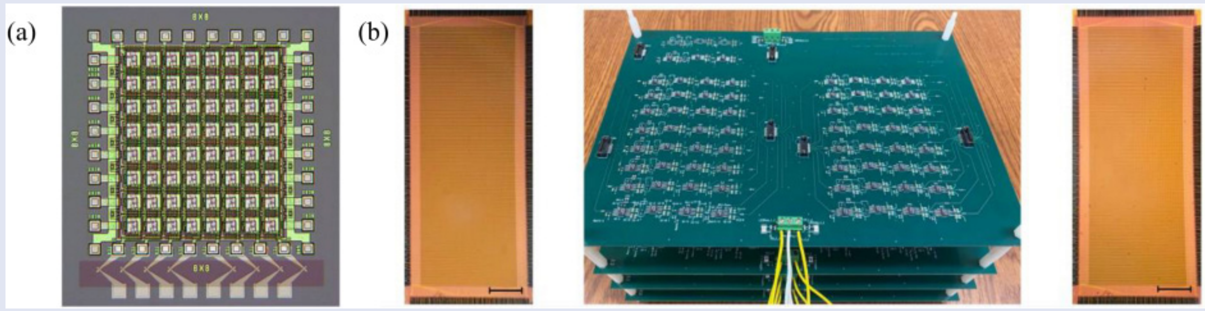

FIGURE 6 Analog neuron circuits with memristive synapses. (a) Optical micrograph of a fully memristive neural network with 8×8 1T1R cells as synapses connecting to eight diffusive memristors as neurons [35]. (b) The fully analog two-layer neural network consists of two 1T1R crossbar arrays (left and right) and fully analog hardware neuron circuits made of off-the-shelf-electronics (middle) [36].

eral circuits underneath, significantly enhancing the throughput of in-memory computing. Moreover, it offers highly scalable and independent operations in row banks, allowing them to be programmed flexibly for different output pixels, filters, or kernels from different convolutional layers, highly improving the flexibility of 3D crossbar arrays.

The 3D memristive circuits pioneer the way for high-density crossbar arrays and achieve remarkable computing performance. The preliminary exploration of the additional dimension also opens up new possibilities for various computing scenarios. 3D memristor crossbar arrays with compatible peripherals can serve as powerful computing cores for different algorithms and applications.

ANALOG NEURON CIRCUITS

Despite significant improvements in the energy efficiency of computing cores for VMMs through advances in memristive devices and crossbar arrays, the overall energy efficiency of computing hardware is still constrained by peripherals that facilitate memristive computing cores [28], [29]. This is because other operations in neural networks, such as activation functions, backward propagation, and derivative calculations, are still processed in digital processors, which necessitates the use of ADCs and DACs to convert analog and digital signals back and forth, resulting in additional latency and power consumption [30], [31]. In addition to device engineering to reduce analog-digital conversions[32], [33], [34], two analog neuron circuits were proposed to connect neighboring layers in neural networks implemented in memristor crossbar arrays to eliminate the need for analog-digital conversions.

One of the proposed neuron circuits involves the utilization of diffusive memristors that exhibit dynamics similar to those found in the ion channels of biological neurons [35]. The diffusive memristor device, composed of silver nanoparticles within a dielectric film, can imitate the neuron function that exhibits stochastic leaky integrate-and-fire dynamics and has an integration time that can be adjusted by either silver migration alone or its interaction with circuit capacitance. These neuron devices implemented rectified linear units (ReLU) activations and were further connected with nonvolatile memristors to construct a fully memristive artificial neural network, as illustrated in Figure 6(a). The network consisted of an 8×8 1T1R array with synapse drift memristors and eight neuron circuits with diffusive memristors fabricated on the same chip. Through this integrated network, we experimented with demonstrating unsupervised synaptic weight updating and pattern classification. The realization of electronic neuronal functionality makes it possible to process the analog outputs from the crossbar arrays directly. It opens the door to developing fully analog neural networks based on memristive devices.

In addition to memristive neurons, analog ReLU circuits implemented with off-the-shelf components were also proposed as neurons for multilayer neural networks using memristor crossbar arrays, eliminating unnecessary analog-digital conversions, communication, and processing between layers of neural networks [36]. Each ReLU circuit was composed of a half-wave current rectifier, a voltage follower, and an inverting amplifier, all built with operational amplifiers. The current rectifier generated a rectified output voltage directly from the input current, which was the output of the previous layer of the neural network. The voltage follower was a unity gain buffer to isolate the first stage. The inverting amplifier was responsible for producing the necessary positive output voltage required for the ReLU activation. Additionally, it adjusted the output voltage to a range of 0 to 0.2 V through scaling for the inputs of the next layer. Figure 6(b) depicts a two-layer hardware neural network utilizing the proposed analog ReLU circuits. The fully analog network consisted of two 1T1R crossbar arrays serving as weight matrices for the two layers and 64 ReLU circuits working as activation neurons. With the Modified National Institute of Standards and Technology (MNIST) dataset, the full hardware network achieved recognition accuracy of 93.63% in the classification task. The experimental demonstration of the fully analog ReLU circuits proves the analog signal transmissions between layers of neural networks without analog-digital conversions. It delivers higher computing throughput and energy efficiency of multilayer neural networks.

Although these neuron circuits only demonstrated the functionality of small

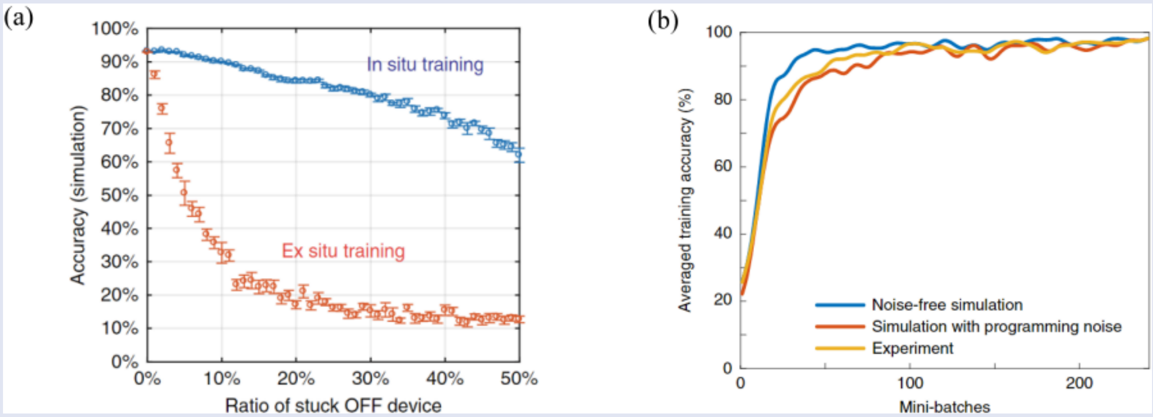

FIGURE 7 Tolerance of memristor non-idealities using in-situ training. (a) The impact of non-responsive devices on the inference accuracy of MNIST dataset with in-situ and ex-situ training approaches [43]. (b) The smoothed accuracy using in-situ training and weight sharing for the convolutional-LSTM network, the experimental curve, the simulation with programming noise, and the simulation with ideal programming [44].

neural networks in analog hardware, they verified the potential of fully analog neural networks and their contributions to the computing throughput and energy efficiency of in-memory computing systems. The promising results demonstrate the possibility of implementing other critical functions of neural networks in the analog domain, further enhancing the energy efficiency of future computing hardware. Additionally, they highlight the importance of system-level designs for computing hardware based on memristive devices.

HARDWARE AND ALGORITHM CO-DESIGN

Memristive hardware innovations were primarily used to accelerate VMMs in traditional neural networks [37], [38], [39]. But merely implementing traditional algorithms designed for high-precision digital computing is difficult to achieve optimal performance due to the non-idealities of memristive devices. The intrinsic dynamics and inevitable conductance drift of these devices can negatively affect computing precision. Therefore, hardware and algorithm co-designs are required to optimize algorithms for memristive hardware considering the unique characteristics of memristors and build circuits and architectures specifically designed to suit novel algorithms and applications. Insightful research has been conducted to address energy-hungry problems in neural networks by leveraging the non-ideal properties and intrinsic noise of memristors [40], [41], [42]. Based on hardware progress discussed in previous sections, hardware-friendly methods that incorporate the non-idealities of memristive devices in the training of neural network algorithms were also proposed.

A self-adaptive in-situ learning algorithm designed for memristor crossbar arrays in multilayer neural networks was developed [43]. Because the memristors in the crossbar arrays can be tuned gradually by controlling the voltages applied to the top electrode of memristors and gates of transistors, which control the compliance current across the devices, linear and symmetric conductance tuning can be realized with minimal cycle-to-cycle and device-to-device variations. Based on this device programming schema, the gradients calculated from the outputs of the memristor crossbar arrays were directly converted to voltage values and applied to memristors to change their conductance representing the weights of neural networks. This process of gradually tuning the synaptic weights of hardware neural networks was called in-situ training. For comparison, ex-situ training, a training diagram programming the conductance to weight values trained by software, was also performed for a two-layer neural network based on memristor crossbar arrays for MNIST dataset classification. The accuracy degradation with the increase of non-responsive devices, which are stuck in a low-conductance state and considered defect devices, is illustrated in Figure 7(a). The comparison showed that the in-situ training process can compensate for non-idealities in the hardware, resulting in significantly greater defect tolerance than using ex-situ training weights in neural networks.

Given the promising results of the in-situ method in hardware training, it was combined with spatiotemporal weight sharing and applied to recurrent convolutional neural networks using memristor crossbar arrays [44]. Because of the structure of crossbar arrays, the weights of multiple kernels in one convolutional layer can be unrolled and programmed to multiple columns of the memristor arrays. This way, multiple kernels in one convolutional layer can share the input data simultaneously and perform the operations in a single cycle. Taking this one step further, the same weights of the long short-term memory (LSTM) network were also mapped to memristor conductance in different columns to be shared across all time steps in LSTM. This weight sharing in both convolutional layers and all time steps of the LSTM network was called spatiotemporal weight sharing and effectively reused the weights programmed to memristor crossbar arrays. Memristive convolutional-LSTM utilizing the spatiotemporal weight sharing and in-situ training achieved comparable accuracies with ideal simulations according to the comparison between the experimental accuracies and the simulations (with or without noise) shown in Figure 7(b). The weight sharing and in-situ training led to reduced trainable parameters and high tolerance to noise caused by non-idealities of memristive devices.

The in-situ training algorithm and the spatiotemporal weight sharing in convolutional-LSTM are pioneering works designed to incorporate device non-idealities during the training process and fully take advantage of the hardware structure. These hardware-algorithm co-designs achieve robust training and parallelism, resulting in higher accuracy and computing throughput. Furthermore, these works demonstrate the necessity of hardware-algorithm co-designs in developing energy-efficient computing hardware systems.

SUMMARY AND PERSPECTIVE

In summary, research at different levels to facilitate the development of energy-efficient computing hardware using memristive nanodevices has been presented. The sub-10 nanometer passive array and 1T1R array with memristors achieving 2048 conductance levels offered solid device solutions to high-density crossbar arrays with stable retention and high endurance. The innovative 3D structural designs, including the CMOS-compatible arrays and flexible arrays for complex networks, improved the packing density and computing throughput, providing energy-efficient computing cores for analog in-memory computing. The fully analog neuron circuits were proposed and experimentally verified in hardware multilayer neural networks, attempting to eliminate analog-digital conversions to boost the overall performance of in-memory computing systems. Hardware-algorithm co-designs were also preliminarily implemented to utilize the intrinsic non-idealities of memristive devices in network training and share weights mapped to the conductance of memristors in crossbar arrays. These research advancements made it feasible to achieve an estimated chip performance of 118 tera operations per second per Watt (TOPS/W) [44]. In the future, further improvements can be expected through the implementation of system-level innovations.

While the comprehensive research presented paves the path to exploring efficient in-memory computing based on memristive devices, challenges remain between these proof-of-concept implementations and computing hardware systems with energy efficiency like human brains. As such, we propose several potential directions for further research to bridge the gap. For memristive devices, investigating memristors with thousands of conductance levels for computing is worth considering, as they have only been proven to function as memory in crossbar arrays [25]. Fully utilizing the thousands of conductance levels in crossbar arrays to improve computing precision will further enhance the accuracy of neural networks, making them competitive with digital computers. Based on multilevel devices, creative structure and interconnect innovations are encouraged to explore the additional dimension in 3D crossbar arrays. The 3D circuit proposed in [27] was only designed to avoid unrolling for 2D inputs and perform 2D convolutions in parallel. Exploring other designs can create more possibilities for building 3D crossbar arrays supporting highly parallel matrix operations. For peripheral circuits, although the designed analog neuron circuits are preliminary implementations and only parts of the computing systems, realizing additional functions in the analog domain is a promising path to follow to avoid extra analog-digital conversions. Along with the hardware progress, more attention should be drawn to hardware-algorithm co-designs, as interdisciplinary research in this direction is still nascent but has already shown promise. In conclusion, while improving the performance of memristive devices, computing cores, and essential peripherals remains crucial for the energy efficiency of in-memory computing, system-level research integrating circuits, interfaces, architectures, and algorithm designs for in-memory computing based on memristive nanodevices is the top priority for the development of energy-efficient computing hardware systems.

ACKNOWLEDGMENT

This work was supported in part by NSF under Grants ECCS-1253073, ECCS-2023752, and CCF-2133475, in part by AFOSR under Grants FA9550-12-1-0038 and FA9550-19-1-0213, in part by AFRL under Grants FA8750-15-2-0044 and FA8750-18-2-0122, and in part by DARPA under Grant N66001-11-1-4143.

ABOUT THE AUTHORS

Yi Huang (yihuang@umass.edu) is with the Department of Electrical and Computer Engineering University of Massachusetts, Amherst, MA, 01003, USA.

Vignesh Ravichandran (ravichan@umass.edu) is with the Department of Electrical and Computer Engineering University of Massachusetts, Amherst, MA, 01003, USA.

Wuyu Zhao (wuyuzhao@umass.edu) is with the Department of Electrical and Computer Engineering University of Massachusetts, Amherst, MA, 01003, USA.

Qiangfei Xia (corresponding author: (qxia@umass.edu)) is with the Department of Electrical and Computer Engineering University of Massachusetts, Amherst, MA, 01003, USA.

REFERENCES

[1] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” *Nature*, vol. 521, no. 7553, pp. 436–444, May 2015, doi: 10.1038/nature14539.

[2] S. Kumar, P. Tiwari, and M. Zymbler, “Internet of Things is a revolutionary approach for future technology enhancement: A review,” *J. Big Data*, vol. 6, no. 1, pp. 1–21, Dec. 2019, doi: 10.1186/s40537-019-0268-2.

[3] A. Mehonic and A. J. Kenyon, “Brain-inspired computing needs a master plan,” *Nature*, vol. 604, no. 7905, pp. 255–260, Apr. 2022, doi: 10.1038/s41586-021-04362-w.

[4] E. Strubell, A. Ganesh, and A. McCallum, “Energy and policy considerations for deep learning in NLP,” in *Proc. 57th Annu. Meeting Assoc. Comput. Linguistics*, 2019, pp. 3645–3650.

[5] A. Sebastian, M. L. Gallo, R. Khaddam-Aljameh, and E. Eleftheriou, “Memory devices and applications for in-memory computing,” *Nature Nanotechnol.*, vol. 15, no. 7, pp. 529–544, Jul. 2020, doi: 10.1038/s41565-020-0655-z.

[6] S. Jung et al., “A crossbar array of magnetoresistive memory devices for in-memory computing,” *Nature*, vol. 601, no. 7892, pp. 211–216, Jan. 2022, doi: 10.1038/s41586-021-04196-6.

[7] S. Yin et al., “Monolithically integrated RRAM- and CMOS-based in-memory computing optimizations for efficient deep learning,” *IEEE Micro*, vol. 39, no. 6, pp. 54–63, Nov./Dec. 2019, doi: 10.1109/MM.2019.2943047.

[8] S. Kumar, X. Wang, J. P. Strachan, Y. Yang, and W. D. Lu, “Dynamical memristors for higher-complexity neuromorphic computing,”

Nature Rev. Mater., vol. 7, no. 7, pp. 575–591, Jul. 2022, doi: 10.1038/s41578-022-00434-z.

[9] S. Goswami et al., "Decision trees within a molecular memristor," *Nature*, vol. 597, no. 7874, pp. 51–56, Sep. 2021, doi: 10.1038/s41586-021-03748-0.

[10] H. Kim, M. R. Mahmoodi, H. Nili, and D. B. Strukov, "4K-memristor analog-grade passive crossbar circuit," *Nature Commun.*, vol. 12, no. 1, Dec. 2021, Art. no. 5198, doi: 10.1038/s41467-021-25455-0.

[11] F. M. Bayat, M. Prezioso, B. Chakrabarti, H. Nili, I. Kataeva, and D. Strukov, "Implementation of multilayer perceptron network with highly uniform passive memristive crossbar circuits," *Nature Commun.*, vol. 9, no. 1, 2018, Art. no. 2331, doi: 10.1038/s41467-018-04482-4.

[12] J. M. Hung et al., "A four-megabit compute-in-memory macro with eight-bit precision based on CMOS and resistive random-access memory for AI edge devices," *Nature Electron.*, vol. 4, no. 12, pp. 921–930, Dec. 2021, doi: 10.1038/s41928-021-00676-9.

[13] Q. Xia and J. J. Yang, "Memristive crossbar arrays for brain-inspired computing," *Nature Mater.*, vol. 18, no. 4. pp. 309–323, Apr. 2019, doi: 10.1038/s41563-019-0291-x.

[14] W.-H. Chen et al., "CMOS-integrated memristive non-volatile computing-in-memory for AI edge processors," *Nature Electron.*, vol. 2, no. 9, pp. 420–428, Sep. 2019, doi: 10.1038/s41928-019-0288-0.

[15] Y. Zhong et al., "A memristor-based analogue reservoir computing system for real-time and power-efficient signal processing," *Nature Electron.*, vol. 5, no. 10, pp. 672–681, Oct. 2022, doi: 10.1038/s41928-022-00838-3.

[16] H. Ning et al., "An in-memory computing architecture based on a duplex two-dimensional material structure for in situ machine learning," *Nature Nanotechnol.*, vol. 18, no. 5, pp. 493–500, Mar. 2023, doi: 10.1038/s41565-023-01343-0.

[17] Z. Wang et al., "Resistive switching materials for information processing," *Nature Rev. Mater.*, vol. 5, no. 3. pp. 173–195, Mar. 2020, doi: 10.1038/s41578-019-0159-3.

[18] W. Wan et al., "A compute-in-memory chip based on resistive random-access memory," *Nature*, vol. 608, no. 7923, pp.504–512,Aug.2022,doi:10.1038/s41586-022-04992-8.

[19] P. Yao et al., "Fully hardware-implemented memristor convolutional neural network," *Nature*, vol. 577, no. 7792, pp. 641–646, 2020, doi: 10.1038/s41586-020-1942-4.

[20] H. Tan et al., "Tactile sensory coding and learning with bio-inspired optoelectronic spiking afferent nerves," *Nature Commun.*, vol. 11, no. 1, Dec. 2020, Art. no. 1369, doi: 10.1038/s41467-020-15105-2.

[21] Z. Zhang et al., "In-sensor reservoir computing system for latent fingerprint recognition with deep ultraviolet photo-synapses and memristor array," *Nature Commun.*, vol. 13, no. 1, Dec. 2022, Art. no. 6590, doi: 10.1038/s41467-022-34230-8.

[22] C. Choi et al., "Reconfigurable heterogeneous integration using stackable chips with embedded artificial intelligence," *Nature Electron.*, vol. 5, no. 6, pp. 386–393, Jun. 2022, doi: 10.1038/s41928-022-00778-y.

[23] S. Pi et al., "Memristor crossbar arrays with 6-nm half-pitch and 2-nm critical dimension," *Nature Nanotechnol.*, vol. 14, no. 1, pp. 35–39, Jan. 2019, doi: 10.1038/s41565-018-0302-0.

[24] "Testing memory downsizing limits," *Nature Nanotechnol.*, vol. 14, no. 1, pp. 1–1, Jan. 2019, doi: 10.1038/s41565-018-0355-0.

[25] M. Rao et al., "Thousands of conductance levels in memristors integrated on CMOS," *Nature*, vol. 615, no. 7954, pp. 823–829, Mar. 2023, doi: 10.1038/s41586-023-05759-5.

[26] C. Li et al., "Three-dimensional crossbar arrays of self-rectifying Si/SiO2/Si memristors," *Nature Commun.*, vol. 8, no. 15666, 2017, Art. no. 15666, doi: 10.1038/ncomms15666.

[27] P. Lin et al., "Three-dimensional memristor circuits as complex neural networks," *Nature Electron.*, vol. 3, no. 4, pp. 225–232, Apr. 2020, doi: 10.1038/s41928-020-0397-9.

[28] F. Cai et al., "A fully integrated reprogrammable memristor–CMOS system for efficient multiply–Accumulate operations," *Nature Electron.*, vol. 2, no. 7, pp. 290–299, Jul. 2019, doi: 10.1038/s41928-019-0270-x.

[29] J. Oh et al., "Preventing vanishing gradient problem of hardware neuromorphic system by implementing imidazole-based memristive ReLU activation neuron," *Adv. Mater.*, vol. 35, 2023, Art. no. 2300023, doi: 10.1002/adma.202300023.

[30] Y. Huang, F. Kiani, F. Ye, and Q. Xia, "From memristive devices to neuromorphic systems," *Appl. Phys. Lett.*, vol. 122, no. 11, Mar. 2023, Art. no. 110501, doi: 10.1063/5.0133044.

[31] W. Cao, X. He, A. Chakrabarti, and X. Zhang, "NeuADC: Neural network-inspired synthesizable analog-to-digital conversion," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 39, no. 9, pp. 1841–1854, Sep. 2020, doi: 10.1109/TCAD.2019.2925391.

[32] S. Kumar, R. S. Williams, and Z. Wang, "Third-order nanocircuit elements for neuromorphic engineering," *Nature*, vol. 585, no. 7826, pp. 518–523, Sep. 2020, doi: 10.1038/s41586-020-2735-5.

[33] S. Dutta et al., "An ising Hamiltonian solver based on coupled stochastic phase-transition nano-oscillators," *Nature Electron.*, vol. 4, no. 7, pp. 502–512, Jul. 2021, doi: 10.1038/s41928-021-00616-7.

[34] J. Moon et al., "Temporal data classification and forecasting using a memristor-based reservoir computing system," *Nature Electron.*, vol. 2, no. 10, pp. 480–487, 2019, doi: 10.1038/s41928-019-0313-3.

[35] Z. Wang et al., "Fully memristive neural networks for pattern classification with unsupervised learning," *Nature Electron.*, vol. 1, no. 2, pp. 137–145, Feb. 2018, doi: 10.1038/s41928-018-0023-2.

[36] F. Kiani, J. Yin, Z. Wang, J. J. Yang, and Q. Xia, "A fully hardware-based memristive multilayer neural network," *Sci. Adv.*, vol. 7, no. 48, Nov. 2021, Art. no. 4801, doi: 10.1126/sciadv.abj4801.

[37] Z. Wang et al., "Reinforcement learning with analogue memristor arrays," *Nature Electron.*, vol. 2, no. 3, pp. 115–124, 2019, doi: 10.1038/s41928-019-0221-6.

[38] B. Gao et al., "Memristor-based analogue computing for brain-inspired sound localization with in situ training," *Nature Electron.*, vol. 13, no. 1, Dec. 2022, Art. no. 2026, doi: 10.1038/s41467-022-29712-8.

[39] G. Milano et al., "In materia reservoir computing with a fully memristive architecture based on self-organizing nanowire networks," *Nature Mater.*, vol. 21, no. 2, pp. 195–202, Feb. 2022, doi: 10.1038/s41563-021-01099-9.

[40] F. Cai et al., "Power-efficient combinatorial optimization using intrinsic noise in memristor Hopfield neural networks," *Nature Electron.*, vol. 3, no. 7, pp. 409–418, Jul. 2020, doi: 10.1038/s41928-020-0436-6.

[41] K. S. Woo, J. Kim, J. Han, W. Kim, Y. H. Jang, and C. S. Hwang, "Probabilistic computing using Cu0.1Te0.9/HfO2/Pt diffusive memristors," *Nature Commun.*, vol. 13, no. 1, Sep. 2022, Art. no. 5762, doi: 10.1038/s41467-022-33455-x.

[42] T. Dalgaty, N. Castellani, C. Turck, K.-E. Harabi, D. Querlioz, and E. Vianello, "In situ learning using intrinsic memristor variability via Markov chain Monte Carlo sampling," *Nature Electron.*, vol. 4, no. 2, pp. 151–161, Jan. 2021, doi: 10.1038/s41928-020-00523-3.

[43] C. Li et al., "Efficient and self-adaptive in-situ learning in multilayer memristor neural networks," *Nature Commun.*, vol. 9, no. 1, pp. 7–14, 2018, doi: 10.1038/s41467-018-04484-2.

[44] Z. Wang et al., "In situ training of feed-forward and recurrent convolutional memristor networks," *Nature Mach. Intell.*, vol. 1, pp. 434–442, 2019, doi: 10.1038/s42256-019-0089-1.