

We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

6,500

Open access books available

176,000

International authors and editors

190M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com



Enabling Neuromorphic Computing for Artificial Intelligence with Hardware-Software Co-Design

Bojing Li, Duo Zhong, Xiang Chen and Chenchen Liu

Abstract

In the last decade, neuromorphic computing was rebirthed with the emergence of novel nano-devices and hardware-software co-design approaches. With the fast advancement in algorithms for today's artificial intelligence (AI) applications, deep neural networks (DNNs) have become the mainstream technology. It has been a new research trend to enable neuromorphic designs for DNNs computing with high computing efficiency in speed and energy. In this chapter, we will summarize the recent advances in neuromorphic computing hardware and system designs with non-volatile resistive access memory (ReRAM) devices. More specifically, we will discuss the ReRAM-based neuromorphic computing hardware and system implementations, hardware-software co-design approaches for quantized and sparse DNNs, and architecture designs.

Keywords: neuromorphic computing, ReRAM, deep neural network, processing-in-memory, hardware-software co-design

1. Introduction

The explosive growth of data and the increasing scale and complexity of deep learning models have made traditional von Neumann architectures inefficient in terms of speed and energy for AI processing. As a result, novel computer architecture and hardware are gaining increasing attention due to their potential to break the “memory wall” constraints in traditional von Neumann architectures. Neuromorphic computing, which is inspired by the functioning of the human brain and now refers to the hardware implementation of artificial neural networks, has been extensively explored and has demonstrated great potential to revolutionize AI processing. By leveraging new nano-devices such as resistive memory (ReRAM), significant progress has been made in this area. Artificial synapses [1–12] and neurons [13–19], and corresponding neuromorphic circuits and hardware systems have been successfully investigated [11, 20–29].

The two-terminal ReRAM device forms a high-density crossbar structure, enabling highly efficient vector-matrix computation naturally [30]. It is worth noting that challenges still exist in applying the ReRAM crossbar to neural network implementation due to factors such as inherent sneak-path leakage, signal noise, and limited conductance states, which can degrade computing accuracy. To address these challenges, various hardware-software co-design methodologies have been proposed [31–41]. For example, researchers explored specialized circuits and algorithms to tolerate the sneak-path current and guarantee the computing accuracy [11, 25–29]. Hardware-adaptive neural network pruning was also been investigated to promote computational efficiency with reduced hardware design and computing energy costs. In addition, several works have focused on improving the computing reliability and security in ReRAM-based neuromorphic computing systems.

The ReRAM, along with its neuromorphic designs, has inspired the development of ReRAM-based in-memory processing accelerators. These accelerators, including PRIME [42], ISAAC [43], and PipeLayer [44], were designed to accelerate the training and inference of convolution neural networks (CNNs). Based on these fundamental explorations, researchers have designed accelerators for various neural networks and applications, such as ReRAM-based accelerators for GANs [45], RNNs [46, 47]. These accelerators demonstrated improved speed and energy efficiency compared to traditional computing platforms such as CPUs and GPUs. With the continuous advancement of ReRAM technology, the ReRAM-based neuromorphic engines are being applied in broader domains [48–61].

In this chapter, we will summarize recent research in ReRAM-based neuromorphic computing. The aforementioned research areas, including hardware implementation, hardware-software co-design, and accelerator architecture, will be covered.

2. ReRAM-based neural network implementation

The emerging ReRAM device has shown superior performance in neuromorphic computing. The ReRAM device is promising to enable highly parallel, ultra-low-power computing in memory for AI applications owing to its structural simplicity, low power consumption, and ease of integration. Hardware implementations of artificial neurons and synapses play an important role in neuromorphic computing, attracting considerable attention over the past few decades. The fundamental function of artificial neurons is to emulate potential accumulation processes and spike generation functions. Meanwhile, artificial synapses are designed to implement various synaptic plasticities and learning weight signals. These plasticities are crucial for the learning function in neuromorphic computing. Based on the artificial neurons and synapses, neuromorphic hardware systems are built for the whole neural network functions. In this section, we will provide an overview of the hardware implementation of ReRAM-based neuromorphic synapses, neurons, and hardware systems.

2.1 ReRAM-based artificial synapses

The performance of synaptic devices directly impacts the learning accuracy and efficiency of a neuromorphic computing system. ReRAM devices are widely utilized to implement synaptic plasticities, such as signal weighting, short-term potentiation/depression (STP/STD), long-term potentiation/depression (LTP/LTD), and spiking-time dependent plasticity (STDP).

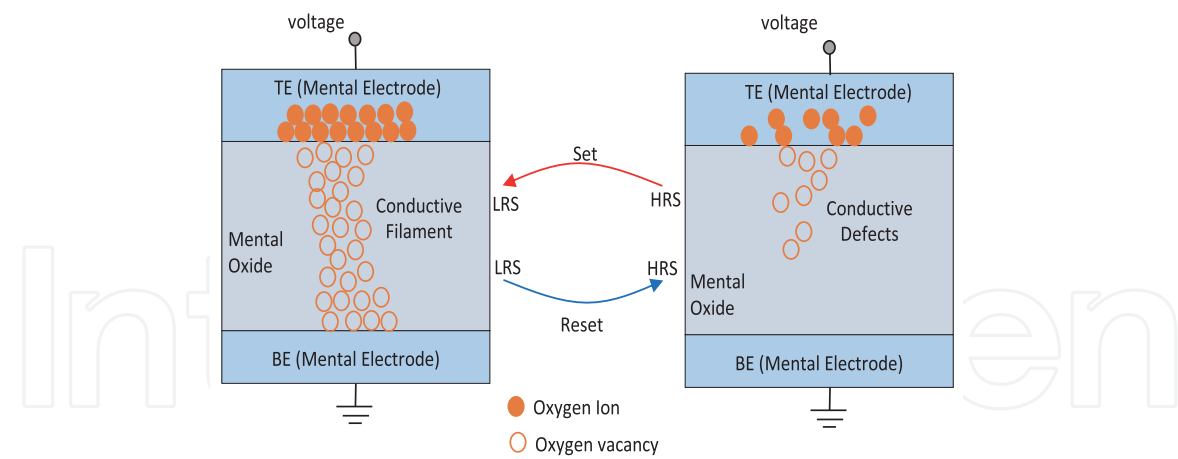


Figure 1.
The typical structure of ReRAMs devices and their switching states [62].

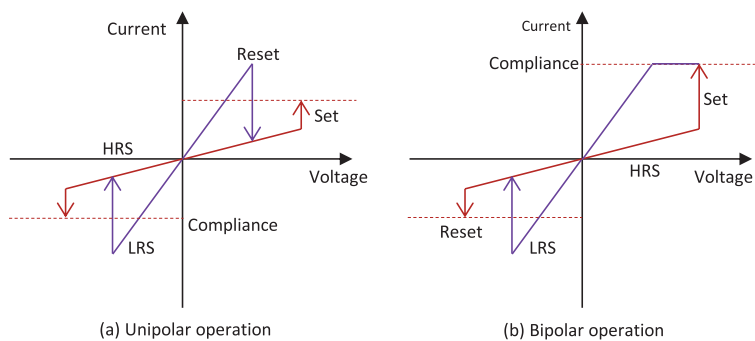


Figure 2.
I-V curves for ReRAM devices on unipolar and bipolar operations [62].

The typical ReRAM device is in a metal/insulator/metal (MIM) structure [62] as is depicted in **Figure 1**, where the device consists of the bottom electrode (BE), top electrode (TE), and oxide layers. As illustrated in **Figure 2**, ReRAM devices can be switched between a high resistance state (HRS) and a low resistance state (LRS) with the unipolar or bipolar operation. This nonlinear feature resembles biological synapses, whose weight changes in response to stimuli passing from pre- to post-synaptic neurons.

In 2010, Lu proposed the first resistive synapse and demonstrated its function of STDP [1]. To achieve the synaptic function of STDP, well-designed shapes of neuronal pulses were also required. Therefore, Yu et al. proposed a metal oxide ReRAM-based synapse and an energy-efficient signal scheme for synapse programming, which involved tuning the pulse amplitude in each time slot [2]. As a further study, Ohno et al. proposed an inorganic synapse using Ag_2S [3], showing that a single proposed synapse device exhibits both time-dependent STP and LTP features of a biological synapse by adjusting the repetition rate of input stimuli without the special design needs of neuronal pulses. Later, Li et al. demonstrated a chalcogenide resistive electronic synapse with an $\text{Ag}/\text{AgInSbTe}/\text{Ag}$ structure to implement STDP function [5], in which the synaptic weights were modified with the cooperation of pre- and post-synaptic spikes and the growth of the weights was more stable by utilizing synaptic saturation mechanism. To eliminate the resistant fluctuations issue [6], Gao et al. proposed an oxide-based 3D vertical structure synapse and simulated a single synapse by using the mean value of a group of resistive switching devices.

Challenges in applying the ReRAM devices in neuromorphic computing were further explored and novel ReRAM-based synapses were proposed. Woo et al. analyzed the TiN/HfO₂/Ti/TiN ReRAM-based synapse and observed deteriorated accuracy drop of the neuromorphic computing under abrupt SET switching operation [8]. Based on this observation, they proposed an optimized ReRAM-based synapse with an Al electrode on top to improve the accuracy of the ReRAM-based neuromorphic computing. To address the issue of abrupt SET switching, Wu et al. found that increasing the temperature led to the transition from abrupt switching to analog switching [9]. Therefore, they proposed a HfO_x/TEL ReRAM-based synapse, which incorporated a thermal enhanced layer (TEL) to confine heat for realizing analog switching. Kim et al. proposed a Ni/SiN_x/AlO_y/TiN ReRAM-based synapse, where the AlO_y layer reduces current overshoot, resulting in a smooth reset switching transition to enhance analog switching performance [10]. Sun et al. proposed an XNOR-ReRAM synapse that enables equivalent XNOR and bit-counting operations to be carried out in parallel in binary neural networks (BNN) [11]. Roy et al. improved the reliability of HfO₂ ReRAM devices through processes of Al-doping, ozone treatment, and post-deposition annealing [12].

2.2 ReRAM-based artificial neurons

Researchers have started to investigate ReRAM-based neuron designs with low design and computation costs for a few years. By leveraging the threshold transition characteristic of ReRAM devices, the dynamic process of pulse generation and resetting of neurons can be accurately represented. In addition, the ReRAM devices hold unique nonlinear features and flexible structures to achieve the rich kinetic properties of neurons. Furthermore, the nanoscale properties of ReRAM devices enable neurons with high power efficiency with the capability of achieving neuron functions with one or a few devices.

The ReRAM-based analog integrate-and-fire (I&F) neurons have been extensively explored. In 2016, Mehonic et al. implemented the Hodgkin-Huxley (H-H) neural model and leaky integrate-and-fire (LIF) neural model using silicon oxide ReRAM devices [13]. As is depicted in **Figure 3**, the charging and discharging process of neurons in the LIF model was successfully implemented. Compared to traditional hybrid analog/digital CMOS silicon neurons, the proposed ReRAM neurons

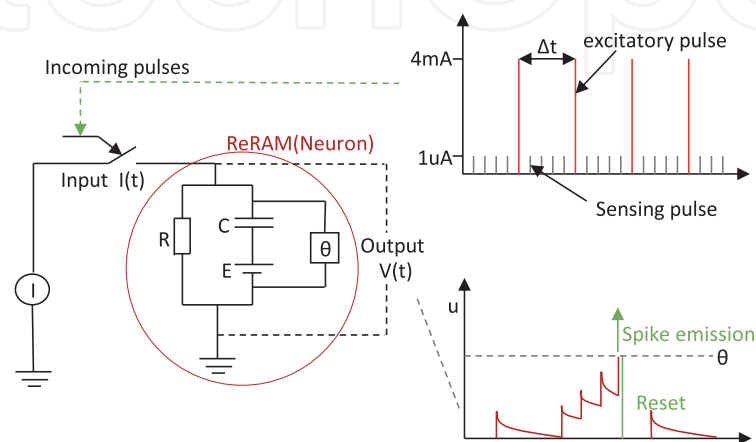


Figure 3. The hardware implementation and equivalent circuits of LIF neural model with silicon oxide ReRAMs [13].

significantly simplified the hardware design. Similarly, Kwon et al. proposed a ReRAM-based analog integrate-and-fire (I&F) neuron circuit without capacitors, which greatly reduces power consumption, delay, and neuron size [15]. Zhang et al. implemented a new I&F neuron based on an Ag/SiO₂/Au threshold switching ReRAM device [16]. The proposed neuron can achieve four critical neuron functions, including all-or-nothing spiking, threshold-driven spiking, a refractory period, and a strength-modulated frequency response. As the first attempt, Lashkare et al. proposed a Pr_{0.7}Ca_{0.3}MnO₃ (PCMO) ReRAM device to implement an integrate-and-fire (I&F) neuron with all the neuronal functionalities—integration, reset, threshold comparison, spike, etc., by only utilizing a single device without any external control circuits [17].

In addition, the ReRAM-based neuron implementations have been extended to a large scale. For example, spiking neural networks on a large scale were achieved based on PCMO ReRAM-based neurons [17]. Lin et al. implemented neurons with a one-transistor-one-ReRAM (1T1R) structure device and integrated the neurons within the synaptic crossbar to build more dense and high-throughput process element [18].

2.3 Neuromorphic system implementation

The CMOS-based neuromorphic chips have several limitations that hinder their further development and applications [20–23]. Firstly, on-chip memory based on SRAM is area inefficient and can hardly store the weights of a large-scale neural network. In addition, the power consumption of off-chip storage using DRAM is more than 100 times higher than that of on-chip memory. The ReRAM device, on the other hand, offers advantages such as low programming voltage, fast switching speed, high integration density, and excellent performance scalability, which make it a promising candidate for neuromorphic computing system implementation.

In the last 10 years, ReRAM-based neuromorphic computing systems for artificial neural network implementation have been extensively studied. In 2012, Hu et al. proposed a neuromorphic hardware system using ReRAM crossbar arrays to achieve recall functions of Brain-State-in-a-Box (BSB) network [24]. In the hardware implementation, the resistive crossbar naturally performs the intensive vector-matrix multiplication, which is the basic computation in the neural network model in parallel. The signal strength was represented by the voltage amplitude and analog neurons with circuitry (e.g., amplifier, analog-to-digital converter) were utilized. Two years later, Hu et al. further implemented the training functions of the BSB model with ReRAM crossbar arrays and alleviated the noise issues found in the previous works, and impacted the computing accuracy [25]. They introduced an iterative scheme for training that uses the sign of inputs and magnitude differences between outputs and inputs, which significantly reduced the circuit design complexity. The analog designs of the neuromorphic systems are vulnerable to signal variations and introduce extremely heavy hardware design and area costs due to the large analog circuit components. With the target of solving the above challenges, in 2015, Liu et al. proposed spiking neuromorphic systems, where input signals were represented by spikes and the need for large-scale analog circuits was significantly reduced [26]. In their works, spiking neuromorphic systems with 1T1R resistive crossbar arrays were implemented for feedforward and Hopfield networks on digital image recognition with IBM 130 nm technology. The 1T1R structure is utilized to control the impact of the sneak path and thus guarantee computing accuracy and a novel Integrate-and-Fire Circuit (IFC) with high-speed and low-power consumption was developed.

With the continuous growing of artificial neural networks in scale and complexity, there is an urgent need to improve computing efficiency in speed and energy. Neuromorphic systems for large-scale and complex neural networks were developed accordingly. For example, Yakopcic et al. implemented the first deep convolutional neural network (CNN) on ReRAM-based crossbars with all data represented in 16 values without classification accuracy degradation [27]. The proposed neuromorphic hardware also achieved parallel computing of the convolution operations. Wen et al. implemented a long short-term memory (MLSTM) network using ReRAM cross-bars [29]. The neural network training process is in extremely high demand of computing resources, while its speed and energy efficiency are highly constrained. Therefore, researchers also explored neuromorphic computing systems that can support neural network training besides inference. For example, Yao et al. implemented parallel online training for gray-scale face classification using an integrated 1024-cell array [28].

3. Neuromorphic hardware-software Co-design

3.1 Algorithm-driven neuromorphic hardware optimizations

The accuracy of data representation in neural network models has a significant impact on computing accuracy. It is worth noting that ReRAM devices face challenges in providing high-precision computing due to manufacturing and hardware design costs although analog conductance states can be ideally achieved. Therefore, ReRAM devices with the limited physical precision pose a severe challenge to neuromorphic computing accuracy.

Researchers have attempted to alleviate the accuracy loss caused by the limited precision through hardware-software co-optimization. For example, Wang et al. proposed a new quantization regularization according to the computing characteristics of ReRAM devices and leverage different levels of regularization for different network layers [31]. They also minimized the impact of quantization by dynamical bias tuning under the fixed weights. Their quantization method achieved minimal accuracy loss under the limited resolution of synaptic weights. Yang et al. investigated a novel approach that processed quantization and training concurrently by optimizing the calculation of continuous weights and quantized weights in stochastic gradient descent [32]. Researchers found that quantizing partial sums was an effective approach to performing high-precision calculations in the ReRAM-based neuromorphic computing system. A team from the University of Illinois at Urbana-Champaign developed a comprehensive quantization approach that considered inputs, weights, and partial sums [33]. They developed a deep reinforcement learning-based search method that can automatically discover the best-mixed configuration to identify the optimal precision configurations of these three types of data. **Table 1** presents a comparison of the accuracy achieved by these quantization methods.

DNN models are becoming increasingly complex and involve tremendous parameters. It was noted that sparsity exists in the neural networks, which indicates that a significant number of parameters are redundant and can be pruned without causing accuracy loss [34, 35]. In the ReRAM-based neuromorphic designs, pruned neural networks can ideally reduce hardware costs and improve computation speed and energy efficiency. Therefore, identifying and eliminating redundancies is crucial to enhance computational efficiency while maintaining accuracy. SNrram [63]

	Dataset	Model	Base-quantization	Ideal Acc	Base Acc	Proposed quantization
Wang et al. [31]	MNIST	MLP	1-level	98.39%	95.97%	98.00%
	MNIST	LeNET	1-level	99.15%	90.77%	98.96%
	CIFAR-10	CNN	1-level	82.12%	17.80%	76.59%
Yang et al. [32]	MNIST	MLP	2-level	97.51%	85.98%	94.53%
	MNIST	LeNET1	2-level	99.00%	69.24%	98.75%
	CIFAR-10	LeNET2	2-level	76.99%	26.20%	73.04%
Huang et al. [33]	—	LeNET	mix-level	97.27%	—	96.09%

Table 1.
The accuracy comparison of quantization methods.

	Software design	Hardware design	Performance
Snrram [63]	Normalizing sparsity in weight and activation into the same format and eliminate redundancy	IRU [*] units store sparse indexes and corresponding peripheral circuit design	Pruned connection(%): 62.7, 82.7, 96.2 in VGG, MLP, Lenet Saved resource(%): 53.2, 81.3, 75 in VGG, MLP, Lenet Speedup: 1.53x, 3.41x in VGG, Lenet
Recom [64]	<ul style="list-style-type: none">• SWOF[†] positively finds and eliminates sparse redundancy in weights and activations• IPMC[‡] to improve pipeline computing efficiency	Column fetching unit for SWOF [†] and corresponding peripheral circuit design	Energy saving: 3.7x, 3.07x, 1.59x in Lenet, Alexnet, Caffenet Speedup: 4.81x, 4.40x, 2.25x in Lenet, Alexnet, Caffenet
Pim-prune [65]	Dividing into blocks and use structural pruning to avoid dislocation problem when focusing on row sparsity and column sparsity simultaneously	<ul style="list-style-type: none">• Modules to store sparsity table for both row and column• Explore optimal energy-efficiency hardware design	Pruned accuracy(%)(lost): 93.84 (−0.30), 93.23(−0.47), 68.72(−1.04) in Resnet, VGG, Resnet Compression rate: 24.85x, 26.85x, 3.56x in Resnet, VGG, Resnet
Sme [66]	<ul style="list-style-type: none">• Quantization and encoding scheme to increase bit-level sparsity• Bit-slicing scheme to accumulate the 0-bits to the same crossbars and decouple crossbar structure• Squeeze-out pruning method to eliminate redundancy	<ul style="list-style-type: none">• Buffer connection to support squeeze-out algorithm and peripheral circuits design to support activation results splicing• Low overhead and orthogonal hardware design that can combine with other research	Pruned accuracy(%)(lost): 93.6 (+0.1), 94.19(+0.05), 76.03(−0.1), 71.57(−0.31) in VGG, Resnet, Resnet, Mobilenet Register reduction(%): 77.80, 96.80 compared to Pim-Prune, SRE Energy efficiency: 2.3x in Resnet Area efficiency: 6.1x in Resnet

^{*}IRU represents indexing register unit. [†]SWOF represents structurally compressed weight oriented fetching. [‡]IPMC represents in-layer pipeline for memory and computation.

Table 2.
The comparison of pruning methods.

introduced a sparsity transfer algorithm to standardize sparsity in weights and activation and an indexing register unit was designed to store sparsity indexes and parse the data. Later, researchers proposed ReCom, another hardware and software co-design accelerator for high-sparsity networks based on ReRAM [64]. ReCom utilized a group lasso algorithm to standardize the shape of filters and accomplish pruning by compressing sparse weights through the regularization of each layer. Other methods besides regularization for ReRAM-based network pruning have been proven to be effective [65, 66]. **Table 2** illustrates a comparison of these pruning methods.

3.2 Hardware-driven hardware-software co-optimization

In ReRAM-based neuromorphic systems, process variation [67, 68], circuit noise [69, 70], retention issues, and endurance issues [71–73] greatly impact its practical applications in real world [74].

Bit failure in resistive devices is a common fault in high-density crossbar arrays. The ReRAM device is unable to switch its conductivity in response to the writing voltage when the failure occurs, which causes fixed resistive devices that are expressed as constant weights in neural network computation. These fixed resistive devices may destroy the overall neural network accuracy. To solve this challenge, Liu et al. first proposed a novel hardware and software co-design approach to improve computational accuracy in neuromorphic designs with high defects [75]. The proposed approach had two strategies: network retraining (software) and redundant resistive devices (hardware). The network retraining consisted of a standard weight-tuning process and a retraining method that prevented the weights at defective resistive devices from being updated. The redundant resistive device strategy deployed additional columns for highly significant weights with defects. Similarly, researchers at Northeastern University proposed a hierarchical progressive pruning method to improve the fault tolerance of ReRAM computing under stuck-off defects and a corresponding differential mapping scheme to support their method for both stuck-on and stuck-off defects [76].

A series of research works have utilized network retraining techniques to minimize the accuracy loss introduced by the imperfect hardware [77–79]. Nevertheless, as the location and quantity of defective resistive devices can differ for each ReRAM-based chip, it is necessary to retrain the network for every instance, leading to an enormous computational burden. To address this challenge, researchers at Purdue University proposed CxDNN, a solution that combines hardware-software compensation techniques for DNNs [80]. CxDNN consists of three optimization steps: a quantization and conversion algorithm, a re-training method, and hardware compensation. The quantization and conversion algorithm extracts a fixed-point neural network from open-access weights based on the accuracy of inputs, weights, and ADC/DAC components. The re-training process mitigates accuracy loss resulting from the nonlinear representation of components such as ADC/DAC, and leverages the available weights to accelerate calculation. Finally, the hardware compensation mechanism adjusts the compensation factor of each column in crossbar arrays based on relative and absolute errors to reduce accuracy loss caused by hardware limitations.

In addition to the aforementioned permanent defects, ReRAM-based systems can also experience instabilities during processing, including noise [81], drifting [82], and programming errors [83]. To mitigate these issues during computation, researchers proposed FTNNA, a ReRAM-customized advanced error-correcting output code (ECCO) scheme [84]. FTNNA applied collaborative logistic classifiers to replace the

classic softmax function and adjusts the weights of these classifiers through transfer learning. Furthermore, they designed a variable-length decode-free coding scheme to reduce neural competition [84]. This approach resulted in significant accuracy improvements without the need for any hardware-specific calibration. Researchers at Tsinghua University proposed a re-configurable redundancy scheme to rescue accuracy degradation caused by stocked resistive devices [85]. Many other studies are dedicated to addressing various aspects of ReRAM technology, such as stuck failures [86, 87], temperature [88], IR-drop [89, 90], and other factors, to enhance the error tolerance of ReRAM-based neuromorphic computing.

4. ReRAM-based in-memory computing architectures

In today's DNN-based AI, the concept of neuromorphic computing has been expanded to non-von Neumann architecture computing paradigms that integrate processing and memory on a single chip. The successful hardware implementation of synapses and neurons with ReRAM devices has enabled the prosperous development of ReRAM-based in-memory accelerators for various DNN applications, including convolutional neural networks (CNNs), graph neural networks (GNNs), etc. These accelerators have demonstrated significant computing speedup and energy efficiency in AI applications such as image processing and language processing. In this chapter, we discuss early exploration of ReRAM-based accelerators and computing architectures for various DNNs and applications.

4.1 ReRAM-based in-memory computing accelerators

With the successful exploration of ReRAM-based neuromorphic hardware, researchers also started to explore architecture-level innovations by developing novel ReRAM-based in-memory processing accelerators.

In 2016, researchers proposed a ReRAM-based neural network accelerator for DNN applications—PRIME [42] with novel in-memory processing architecture. As is shown in **Figure 4**, unlike previous in-memory processing architectures that integrate additional processing units to memory, PRIME has full-function (FF) subarrays that

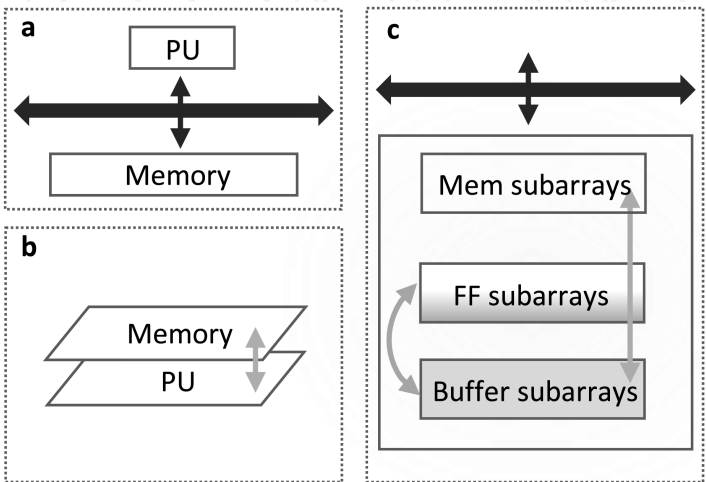


Figure 4.
(a) Traditional shared memory-based processor-coprocessor architecture, (b) processing in-memory approach using 3D integration technologies, (c) PRIME design [42].

can act as a memory unit for data storage or perform matrix multiplication in DNN computation. The FF subarrays mainly consist of the components: (1) decoders and drivers that provide analog inputs to ReRAM crossbar arrays by controlling voltage through amplifiers, latches, etc.; (2) multiplexers that are used to support subtraction and the sigmoid activation function; (3) sense amplifiers to sense computing results from the ReRAM devices and support ReLU functions with the additional hardware units. PRIME maximum reuses the periphery circuits to enable the switch of ReRAM between storage and computing with reduced design costs. Moreover, PRIME proposed high-precision NN acceleration based on its architecture and provided software/hardware interfaces for developers.

In the same year, researchers at the University of Utah introduced ISAAC, a comprehensive ReRAM-based accelerator, aimed at offering high-speed and energy-efficient computations of convolutional neural networks (CNNs) [43]. The ISAAC is a multi-stage architecture, which includes tiles, IMAs, and ReRAM crossbar arrays, as illustrated in **Figure 5**. The fundamental component used for deploying a neural network layer is IMAs, where several IMAs share eDRAM activation components and other peripheral circuits. ISAAC presented an inter-layer pipeline coupled with a buffer optimization strategy. ISAAC processed all layers in a CNN concurrently and redeployed layers to maintain the pipeline’s continuity, and hence improved overall throughput and computational efficiency were achieved. The subsequent layer started processing as soon as the output feature maps of the current layer meet the size requirement of the convolution kernels of the next layer instead of waiting for the final results. This strategy significantly reduced the capacity requirement and on-chip space occupation of eDRAM. In addition, ISAAC developed an encoding scheme to reduce the ADC resolution needs to 1-bit, which can significantly reduce hardware costs and improve computing efficiency. The ISAAC achieved significant improvement in computing throughput, energy, and computational density compared to previous designs.

Although previous architectures demonstrated high computing efficiency, limitations still exist. For example, they were designed only for inference and did not

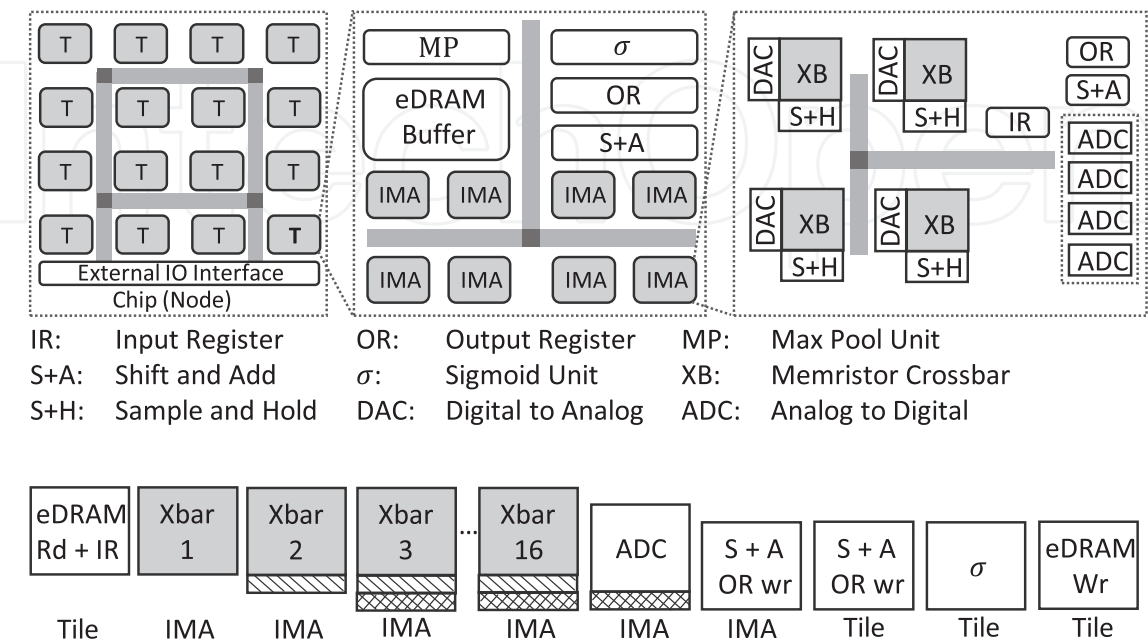


Figure 5.
Architecture and pipeline of ISAAC [43].

	CE* (GOPS/s/mm ²)	PE† (GPOS/s/W)	Large data mapping	Data precision	Support Train/Inf
PRIME [42]	1230	2100	Crossbar (Analog)	6-bit (In.) & 8-bit (Wt.)	Inf
ISAAC [43]	479.0	380.7	Crossbar (Analog)	16-bit Fixed	Inf
PipeLayer [44]	1485	142.9	Bitline-wise Keral/X-bar	16-bit	Train & Inf
*CE represents computation efficiency. †PE represents power efficiency.					

Table 3.
The comparison of ReRAM-based in-memory computing accelerators.

support the training process. The computing pipelines were deep and may introduce pipeline bubbles. Therefore, in 2017, a novel ReRAM-based in-memory computing architecture PipeLayer was proposed aiming to solve this challenge [44]. The proposed architecture is based on PRIME and ISAAC and can support the training process. In addition, PipeLayer has optimized inter-layer and intra-layer pipelines to achieve high throughput in both training and inference. The same as PRIME, the ReRAM subarrays were utilized for both storage and computing with the capability of functionality shifting. During the training process, the error is obtained after comparing the labels and loss function in the last forward cycle. In the error backpropagation process, weights and errors, which were also calculated in the ReRAM morphable subarrays and saved in storage subarrays, were updated by the layer errors and partial derivatives and passed to the previous layer. It is noted that the depth of logical cycles was independent of the number of NN layers, preventing potential issues arising from deep pipelines, such as stall, bubble, and latency. In addition, PipeLayer introduced a parameter to repeatedly deploy operators, building a trade-space-for-time intra-layer parallel pipeline. This intra-layer pipeline, together with the inter-layer pipeline, supports the high throughput of PipeLayer. The comparison of these ReRAM-based in-memory computing accelerators is shown in **Table 3**.

4.2 ReRAM-based accelerators for various DNNs and applications

After the great success of the ReRAM-based neuromorphic accelerators, customized ReRAM accelerators tailored for specific types of neural networks and applications are also emerging. For example, generative adversarial networks (GANs) [91] require frequent data transmission between the generative and adversarial networks, which makes them highly demanding in terms of memory, computational resources, and data transfer. ReGAN, a ReRAM-based accelerator for GANS, has been designed to leverage the advantages of processing in-memory and ReRAM crossbar arrays, as well as optimized parallel computation and data dependency, to achieve significantly high performance and energy efficiency [45]. To expedite the execution of recurrent neural networks (RNNs), ReRAM accelerators tailored to various RNN configurations were proposed [46].

In graph processing tasks, high data transfer costs have led researchers to focus on the optimization of memory access [92]. ReRAM architecture’s inherent *in situ* computing can minimize these overheads. GraphR was the first to introduce a ReRAM-based architecture for graph computation [51]. The feasibility of ReRAM for graph processing was analyzed, and sparse matrix-vector multiplications were applied to

data blocks of compressed representation. This approach designed ReRAM-based graph engine to accelerate the network with parallel computation, while a drawback of additional useless multiplications with zero due to sparsity still exists. Subsequently, Spara presented a novel vertex mapping strategy to address this challenge [52]. There are also ReRAM-based architectures for graph processing focus on sparsity [53, 54], three-dimensional architecture [55, 56], regularization, redundant computation [57], etc.

Transformers, one of the most advanced models in current natural language processing (NLP), present several challenges to the ReRAM-based neuromorphic computing [93]. For example, one layer's input includes the results of the previous layer due to the self-attention mechanism, generating data dependencies that result in bubbles in the traditional interlayer pipeline. The calculation of scaled dot-product attention significantly differs from traditional multiply-accumulate operations in CNNs. ReTransformer [58] and ReBERT [59] are dedicated to addressing these challenges. ReTransformer mathematically decomposed matrix multiplication into two steps, reducing the pipeline bubbles by performing the first initialization step and the second calculation step sequentially in two separate cycles. ReBERT concentrated on the attention mechanism and proposed a window self-attention mechanism with a corresponding window size search algorithm to support the ReRAM crossbar arrays and achieved significant speedup and energy performance. There are also other ReRAM-based architectures for NLP, including sparsity attention [60], and BERT deployment [61], etc.

5. Conclusions

In this chapter, we gave an overview of the implementation of ReRAM-based neuromorphic computing engines in the last several years. The overview covers hardware designs of synapses and neurons, neural network implementation, hardware and software co-design, and novel architectural designs. The ReRAM-based synapses and neurons give a rebirth of neuromorphic computing through its ultra-small scale, inherent synaptic plasticity property, and capability to enable high-parallelism neural network operations. Accordingly, neuromorphic computing systems for different neural networks were implemented with significantly high computing efficiency in speed and energy compared to the traditional computing platform. Hardware-software co-design is a widely utilized approach to overcome the ReRAM-based hardware limitations such as limited resistive states, stuck-at-fault, signal fluctuation. Novel in-memory computing architectures are also extensively explored based on the ReRAM-based neuromorphic engine and various techniques are explored to overcome the challenges in neural network computing and optimize computing efficiency. With the prosperous advancement of ReRAM-based neuromorphic computing, specialized accelerators for various neural networks and applications are also under extensive study.

Acknowledgements

This work is supported by NSF CNS-1939380. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of grant agencies or their contractors.

IntechOpen

Author details

Bojing Li^{1†}, Duo Zhong^{1†}, Xiang Chen² and Chenchen Liu^{1*}


1 University of Maryland Baltimore County, Baltimore, USA

2 George Mason University, Fairfax, USA

*Address all correspondence to: ccliu@umbc.edu

† These authors contributed equally.

IntechOpen

© 2023 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/3.0>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited. 

References

- [1] Jo SH, Chang T, Ebong I, Bhadviya BB, Mazumder P, Wei L. Nanoscale memristor device as synapse in neuromorphic systems. *Nano Letters*. 2010;**10**(4):1297-1301
- [2] Shimeng Y, Yi W, Jeyasingh R, Kuzum D, Philip H-S, Wong. An electronic synapse device based on metal oxide resistive switching memory for neuromorphic computation. *IEEE Transactions on Electron Devices*. 2011; **58**(8):2729-2737
- [3] Ohno T, Hasegawa T, Tsuruoka T, Terabe K, Gimzewski JK, Aono M. Short-term plasticity and long-term potentiation mimicked in single inorganic synapses. *Nature Materials*. 2011;**10**(8):591-595
- [4] Shimeng Y, Gao B, Fang Z, Hongyu Y, Kang J, Philip H-S, et al. A low energy oxide-based electronic synaptic device for neuromorphic visual systems with tolerance to device variation. *Advanced Materials*. 2013; **25**(12):1774-1779
- [5] Li Y, Zhong Y, Zhang J, Lei X, Wang Q, Sun H, et al. Activity-dependent synaptic plasticity of a chalcogenide electronic synapse for neuromorphic systems. *Scientific Reports*. 2014;**4**(1):4906
- [6] Gao B, Bi Y, Chen H-Y, Liu R, Huang P, Chen B, et al. Ultra-low-energy three-dimensional oxide-based electronic synapses for implementation of robust high-accuracy neuromorphic computation systems. *ACS Nano*. 2014; **8**(7):6998-7004
- [7] Kim S, Chao D, Sheridan P, Ma W, Choi SH, Lu WD. Experimental demonstration of a second-order memristor and its ability to biorealistically implement synaptic plasticity. *Nano Letters*. 2015;**15**(3): 2203-2211
- [8] Woo J, Moon K, Song J, Lee S, Kwak M, Park J, et al. Improved synaptic behavior under identical pulses using AlO_x/HfO₂ bilayer rram array for neuromorphic systems. *IEEE Electron Device Letters*. 2016;**37**(8):994-997
- [9] Wei W, Huaqiang W, Gao B, Deng N, Shimeng Y, Qian H. Improving analog switching in HfO_x-based resistive memory with a thermal enhanced layer. *IEEE Electron Device Letters*. 2017; **38**(8):1019-1022
- [10] Kim S, Kim H, Hwang S, Kim M-H, Chang Y-F, Park B-G. Analog synaptic behavior of a silicon nitride memristor. *ACS Applied Materials & Interfaces*. 2017;**9**(46):40420-40427
- [11] Sun X, Yin S, Peng X, Liu R, Seo J-s, Yu S. Xnor-rram: A scalable and parallel resistive synaptic architecture for binary neural networks. In: 2018 Design, Automation & Test in Europe Conference & Exhibition (DATE). Dresden, Germany: IEEE; 2018. pp. 1423-1428
- [12] Roy S, Niu G, Wang Q, Wang Y, Zhang Y, Heping W, et al. Toward a reliable synaptic simulation using Al-doped HFO₂ RRAM. *ACS Applied Materials & Interfaces*. 2020;**12**(9): 10648-10656
- [13] Mehonic A, Kenyon AJ. Emulating the electrical activity of the neuron using a silicon oxide rram cell. *Frontiers in Neuroscience*. 2016;**10**:57
- [14] Babacan Y, Kaçar F, Gürkan K. A spiking and bursting neuron circuit based on memristor. *Neurocomputing*. 2016;**203**:86-91

- [15] Kwon M-W, Kim S, Kim M-H, Park J, Kim H, Hwang S, et al. Integrate-and-fire (I&F) neuron circuit using resistive-switching random access memory (rram). *Journal of Nanoscience and Nanotechnology*. 2017;**17**(5): 3038-3041
- [16] Zhang X, Wang W, Liu Q, Zhao X, Wei J, Cao R, et al. An artificial neuron based on a threshold switching memristor. *IEEE Electron Device Letters*. 2017;**39**(2):308-311
- [17] Sandip Lashkare S, Chouhan TC, Bhat A, Kumbhare P, Ganguly UJIEDL. Pcmo rram for integrate-and-fire neuron in spiking neural networks. *IEEE Electron Device Letters*. 2018;**39**(4): 484-487
- [18] Lin J, Yuan J-S. A scalable and reconfigurable in-memory architecture for ternary deep spiking neural network with ReRAM based neurons. *Neurocomputing*. 2020;**375**:102-112
- [19] Suhas Kumar R, Williams S, Wang Z. Third-order nanocircuit elements for neuromorphic engineering. *Nature*. 2020;**585**(7826):518-523
- [20] Markram H. The blue brain project. *Nature Reviews Neuroscience*. 2006; **7**(2):153-160
- [21] Merolla PA, Arthur JV, Alvarez-Icaza R, Cassidy AS, Sawada J, Akopyan F, et al. A million spiking-neuron integrated circuit with a scalable communication network and interface. *Science*. 2014;**345**(6197):668-673
- [22] Benjamin BV, Gao P, McQuinn E, Choudhary S, Chandrasekaran AR, Bussat J-M, et al. Neurogrid: A mixed-analog-digital multichip system for large-scale neural simulations. *Proceedings of the IEEE*. 2014;**102**(5): 699-716
- [23] Davies M, Srinivasa N, Lin T-H, Chinya G, Cao Y, Choday SH, et al. Loihi: A neuromorphic many core processor with on-chip learning. *IEEE Micro*. 2018;**38**(1):82-99
- [24] Hu M, Li H, Wu Q, Rose GS. Hardware realization of BSB recall function using Memristor crossbar arrays. In: *Proceedings of the 49th Annual Design Automation Conference*. New York, USA: ACM; 2012. pp. 498-503
- [25] Miao H, Li H, Chen Y, Qing W, Rose GS, Linderman RW. Memristor crossbar-based neuromorphic computing system: A case study. *IEEE Transactions on Neural Networks and Learning Systems*. 2014;**25**(10):1864-1878
- [26] Liu C, Yan B, Yang C, Song L, Zheng L, Liu B, et al. A spiking neuromorphic design with resistive crossbar. In: *Proceedings of the 52nd Annual Design Automation Conference*. New York, USA: ACM; 2015. pp. 1-6
- [27] Yakopcic C, Alom MZ, Taha TM. Memristor crossbar deep network implementation based on a convolutional neural network. In: *2016 International Joint Conference on Neural Networks (IJCNN)*. Vancouver, BC, Canada: IEEE; 2016. pp. 963-970
- [28] Yao P, Huaqiang W, Gao B, Eryilmaz SB, Huang X, Zhang W, et al. Face classification using electronic synapses. *Nature Communications*. 2017;**8**(1):15199
- [29] Wen S, Wei H, Yang Y, Guo Z, Zeng Z, Huang T, et al. Memristive LSTM network for sentiment analysis. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*. 2019;**51**(3): 1794-1804
- [30] Joshua Yang J, Strukov DB, Stewart DR. Memristive devices for

computing. *Nature Nanotechnology*. 2013;8(1):13-24

[31] Wang Y, Wen W, Song L, Li HH. Classification accuracy improvement for neuromorphic computing systems with one-level precision synapses. In: 2017 22nd Asia and South Pacific Design Automation Conference (ASP-DAC). Chiba, Japan: IEEE; 2017. pp. 776-781

[32] Yang Q, Li H, Qing W. A quantized training method to enhance accuracy of ReRAM-based neuromorphic systems. In: 2018 IEEE International Symposium on Circuits and Systems (ISCAS). Florence, Italy: IEEE; 2018. pp. 1-5

[33] Huang S, Ankit A, Silveira P, Antunes R, Chalamalasetti SR, El Hajj I, et al. Mixed precision quantization for ReRAM-based dnn inference accelerators. In: 2021 26th Asia and South Pacific Design Automation Conference (ASP-DAC). New York, USA: ACM; 2021. pp. 372-377

[34] Changpinyo S, Sandler M, Zhmoginov A. The power of sparsity in convolutional neural networks. *arXiv preprint arXiv:1702.06257*. 2017

[35] Han S, Mao H, Dally WJ. Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding. *arXiv preprint arXiv:1510.00149*. 2015

[36] Wang Y, Jin S, Li T. A low cost weight obfuscation scheme for security enhancement of ReRAM based neural network accelerators. In: 2021 26th Asia and South Pacific Design Automation Conference (ASP-DAC). Tokyo, Japan: IEEE; 2021. pp. 499-504

[37] Zhang J, Wang C, Cai Y, Zhu Z, Kline D, Yang H, et al. Wesco: Weight-encoded reliability and security co-design for in-memory computing

systems. In: 2022 IEEE Computer Society Annual Symposium on VLSI (ISVLSI). Nicosia, Cyprus: IEEE; 2022. pp. 296-301

[38] Cai Y, Chen X, Lu T, Yu W, Yang H. Enabling secure in-memory neural network computing by sparse fast gradient encryption. In: 2019 IEEE/ACM International Conference on Computer-Aided Design (ICCAD). Westminster, CO, USA: IEEE; 2019. pp. 1-8

[39] Zou M, Zhou J, Cui X, Wang W, Kvatinsky S. Enhancing security of memristor computing system through secure weight mapping. In: 2022 IEEE Computer Society Annual Symposium on VLSI (ISVLSI). Nicosia, Cyprus: IEEE; 2022. pp. 182-187

[40] Zou M, Du N, Kvatinsky S. Review of security techniques for memristor computing systems. *arXiv preprint arXiv:2212.09347*. 2022

[41] Yang C, Liu B, Li H, Chen Y, Barnell M, Qing W, et al. Thwarting replication attack against memristor-based neuromorphic computing system. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*. 2020;39(10):2192-2205

[42] Chi P, Li S, Xu C, Zhang T, Zhao J, Liu Y, et al. Prime: A novel processing-in-memory architecture for neural network computation in ReRAM-based main memory. In: 2016 ACM/IEEE 43rd Annual International Symposium on Computer Architecture (ISCA). New York, NY, USA: ACM; 2016. pp. 27-39

[43] Shafiee A, Nag A, Muralimanohar N, Balasubramonian R, Strachan JP, Hu M, et al. Isaac: A convolutional neural network accelerator with in-situ analog arithmetic in crossbars. In: 2016 ACM/IEEE 43rd Annual International Symposium on Computer Architecture

(ISCA). New York, NY, USA: ACM; 2016. pp. 14-26

[44] Song L, Qian X, Li H, Chen Y. Pipelayer: A pipelined ReRAM-based accelerator for deep learning. In: 2017 IEEE International Symposium on High Performance Computer Architecture (HPCA). 2017. pp. 541-552

[45] Chen F, Song L, Chen Y. Regan: A pipelined ReRAM-based accelerator for generative adversarial networks. In: 2018 23rd Asia and South Pacific Design Automation Conference (ASP-DAC). Jeju, Korea (South): IEEE; 2018. pp. 178-183

[46] Long Y, Na T, Mukhopadhyay S. ReRAM-based processing-in-memory architecture for recurrent neural network acceleration. IEEE Transactions on Very Large Scale Integration (VLSI) Systems. 2018; **26**(12):2781-2794

[47] Hochreiter S, Schmidhuber J. Long short-term memory. Neural Computation. 1997;**9**(8):1735-1780

[48] Liu X, Zhou M, Ausavarungnirun R, Eilert S, Akel A, Rosing T, et al. FPRA: A fine-grained parallel rram architecture. In: 2021 IEEE/ACM International Symposium on Low Power Electronics and Design (ISLPED). Boston, MA, USA: IEEE; 2021. pp. 1-6

[49] Yuan G, Behnam P, Li Z, Shafiee A, Lin S, Ma X, et al. Forms: Fine-grained polarized ReRAM-based in-situ computation for mixed-signal DNN accelerator. In: 2021 ACM/IEEE 48th Annual International Symposium on Computer Architecture (ISCA). 2021. pp. 265-278

[50] Jin H, Liu C, Liu H, Luo R, Jiahong X, Mao F, et al. REHY: A

ReRAM-based digital/analog hybrid PIM architecture for accelerating CNN training. IEEE Transactions on Parallel and Distributed Systems. 2022;**33**(11): 2872-2884

[51] Song L, Zhuo Y, Qian X, Li H, Chen Y. Graphr: Accelerating graph processing using ReRAM. In: 2018 IEEE International Symposium on High Performance Computer Architecture (HPCA). 2018. pp. 531-543

[52] Zheng L, Zhao J, Yu H, Wang Q, Zeng Z, Xue J, et al. Spara: An energy-efficient ReRAM-based accelerator for sparse graph analytics applications. In: 2020 IEEE International Parallel and Distributed Processing Symposium (IPDPS). 2020. pp. 696-707

[53] Yu H, Zheng L, Liao X, Jin H, Yao P, Gui C. Ragra: Leveraging monolithic 3d ReRAM for massively-parallel graph processing. In: 2019 Design, Automation Test in Europe Conference Exhibition (DATE). Florence, Italy: IEEE; 2019. pp. 1273-1276

[54] Yang T, Li D, Ma F, Song Z, Zhao Y, Zhang J, et al. PASGCN: An ReRAM-based PIM design for GCN with adaptively sparsified graphs. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems. 2022; **42**(1):150-163

[55] Arka AI, Doppa JR, Pande PP, Joardar BK, Chakrabarty K. Regraphx: NOC-enabled 3d heterogeneous ReRAM architecture for training graph neural networks. In: 2021 Design, Automation Test in Europe Conference Exhibition (DATE). Grenoble, France: IEEE; 2021. pp. 1667-1672

[56] Choudhury D, Barik R, Rajam AS, Kalyanaraman A, Pande PP. Software/hardware co-design of 3d NOC-based GPU architectures for accelerated graph

computations. *ACM Transactions on Design Automation of Electronic Systems (TODAES)*. New York, USA: ACM; 2022;27(6):1-22

[57] Chen C, Li K, Li Y, Zou X. REGNN: A redundancy-eliminated graph neural networks accelerator. In: 2022 IEEE International Symposium on High-Performance Computer Architecture (HPCA). 2022. pp. 429-443

[58] Yang X, Yan B, Li H, Chen Y. Retransformer: ReRAM-based processing-in-memory architecture for transformer acceleration. In: 2020 IEEE/ACM International Conference on Computer Aided Design (ICCAD). 2020. pp. 1-9

[59] Kang M, Shin H, Kim L-S. A framework for accelerating transformer-based language model on ReRAM-based architecture. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*. 2022;41(9): 3026-3039

[60] Li H, Jin H, Zheng L, Yu H, Liao X, Chen D, et al. CPSAA: Accelerating sparse attention using crossbar-based processing-in-memory architecture. *arXiv preprint arXiv:2210.06696*. 2022

[61] Kang M, Shin H, Shin J, Kim L-S. A framework for area-efficient multi-task Bert execution on ReRAM-based accelerators. In: 2021 IEEE/ACM International Conference on Computer Aided Design (ICCAD). Munich, Germany: IEEE; 2021. pp. 1-9

[62] Akinaga H, Shima H. Resistive random access memory (ReRAM) based on metal oxides. *Proceedings of the IEEE*. 2010;98(12):2237-2251

[63] Wang P, Yu J, Hong C, Lyu Y, Wang D, Xie Y. SNRRAM: An efficient sparse neural network computation

architecture based on resistive random-access memory. In: 2018 55th ACM/ESDA/IEEE Design Automation Conference (DAC). 2018. pp. 1-6

[64] Ji H, Song L, Jiang L, Li H, Chen Y. Recom: An efficient resistive accelerator for compressed deep neural networks. In: 2018 Design, Automation Test in Europe Conference Exhibition (DATE). 2018. pp. 237-240

[65] Chu C, Wang Y, Zhao Y, Ma X, Ye S, Hong Y, et al. Pim-prune: Fine-grain dcnn pruning for crossbar-based process-in-memory architecture. In: 2020 57th ACM/IEEE Design Automation Conference (DAC). San Francisco, CA, USA: IEEE; 2020. pp. 1-6

[66] Liu F, Zhao W, He Z, Wang Z, Zhao Y, Yang T, et al. SME: ReRAM-based sparse-multiplication-engine to squeeze-out bit sparsity of neural network. In: 2021 IEEE 39th International Conference on Computer Design (ICCD). Storrs, CT, USA: IEEE; 2021. pp. 417-424

[67] Chen A, Lin M-R. Variability of resistive switching memories and its impact on crossbar array performance. In: 2011 International Reliability Physics Symposium. 2011. pp. MY.7.1-MY.7.4

[68] Dongale TD, Patil KP, Mullani SB, More KV, Delekar SD, Patil PS, et al. Investigation of process parameter variation in the memristor based resistive random access memory (RRAM): Effect of device size variations. *Materials Science in Semiconductor Processing*. 2015;35:174-180

[69] Ambrogio S, Simone Balatti A, Cubeta AC, Ramaswamy N, Ielmini D. Understanding switching variability and random telegraph noise in resistive ram. In: 2013 IEEE International Electron Devices Meeting. Washington, DC, USA: IEEE; 2013. pp. 31-35

- [70] Choi S, Yang Y, Wei L. Random telegraph noise and resistance switching analysis of oxide based resistive memory. *Nanoscale*. 2014;**6**(1):400-404
- [71] Beckmann K, Holt J, Manem H, Van Nostrand J, Cady NC. Nanoscale hafnium oxide RRAM devices exhibit pulse dependent behavior and multi-level resistance capability. *MRS Advances*. 2016;**1**(49):3355-3360
- [72] Chen YY, Goux L, Clima S, Govoreanu B, Degraeve R, Kar GS, et al. Endurance/retention trade-off on HfO₂/Metal cap 1t1r bipolar rram. *IEEE Transactions on Electron Devices*. 2013; **60**(3):1114-1121
- [73] Wong H-SP, Lee H-Y, Yu S, Chen Y-S, Wu Y, Chen P-S, et al. Metal-oxide rram. *Proceedings of the IEEE*. 2012; **100**(6):1951-1970
- [74] Chen Y, Xie Y, Song L, Chen F, Tang T. A survey of accelerator architectures for deep neural networks. *Engineering*. 2020;**6**(3):264-274
- [75] Liu C, Hu M, Strachan JP, Li H. Rescuing memristor-based neuromorphic design with high defects. In: 2017 54th ACM/EDAC/IEEE Design Automation Conference (DAC). New York, USA: ACM; 2017. pp. 1-6
- [76] Yuan G, Liao Z, Ma X, Cai Y, Kong Z, Shen X, et al. Improving DNN fault tolerance using weight pruning and differential crossbar mapping for ReRAM-based edge AI. In: 2021 22nd International Symposium on Quality Electronic Design (ISQED). 2021. pp. 135-141
- [77] Chakraborty I, Roy D, Roy K. Technology aware training in memristive neuromorphic systems for nonideal synaptic crossbars. *IEEE Transactions on Emerging Topics in Computational Intelligence*. 2018;**2**(5): 335-344
- [78] Chen L, Li J, Chen Y, Deng Q, Shen J, Liang X, et al. Accelerator- friendly neural-network training: Learning variations and defects in RRAM crossbar. In: Design, Automation & Test in Europe Conference & Exhibition (DATE), 2017. Lausanne, Switzerland: IEEE; 2017. pp. 19-24
- [79] Chen P-Y, Lin B, Wang I-T, Hou T-H, Ye J, Vrudhula S, et al. Mitigating effects of non-ideal synaptic device characteristics for on-chip learning. In: 2015 IEEE/ACM International Conference on Computer-Aided Design (ICCAD). IEEE; 2015. pp. 194-199
- [80] Jain S, Raghunathan A. CXDNN: Hardware-software compensation methods for deep neural networks on resistive crossbar systems. *ACM Transactions on Embedded Computing Systems (TECS)*. New York, USA: ACM; 2019;**18**(6):1-23
- [81] Yide D, Jing L, Fang H, Chen H, Cai Y, Wang R, et al. Exploring the impact of random telegraph noise-induced accuracy loss on resistive ram-based deep neural network. *IEEE Transactions on Electron Devices*. 2020; **67**(8):3335-3340
- [82] Chang T, Jo S-H, Wei L. Short-term memory to long-term memory transition in a nanoscale memristor. *ACS Nano*. 2011;**5**(9):7669-7676
- [83] Liu T, Wen W, Jiang L, Wang Y, Yang C, Quan G. A fault-tolerant neural network architecture. In: 2019 56th ACM/IEEE Design Automation Conference (DAC). 2019. pp. 1-6
- [84] Liu T, Liu Z, Lin F, Jin Y, Quan G, Wen W. Mt-spike: A multilayer time-

- based spiking neuromorphic architecture with temporal error backpropagation. In: 2017 IEEE/ACM International Conference on Computer-Aided Design (ICCAD). IEEE; 2017. pp. 450-457
- [85] Xia L, Huangfu W, Tang T, Yin X, Chakrabarty K, Yuan Xie Y, et al. Stuck-at fault tolerance in rram computing systems. *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*. 2018;**8**(1):102-115
- [86] Zhang B, Uysal N, Fan D, Ewetz R. Handling stuck-at-fault defects using matrix transformation for robust inference of DNNs. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*. 2020; **39**(10):2448-2460
- [87] Yeo I, Chu M, Gi S-G, Hwang H, Lee B-G. Stuck-at-fault tolerant schemes for memristor crossbar array-based neural networks. *IEEE Transactions on Electron Devices*. 2019;**66**(7): 2937-2945
- [88] Beigi M, V, Memik G. Thermal-aware optimizations of ReRAM-based neuromorphic computing systems. In: 2018 55th ACM/ESDA/IEEE Design Automation Conference (DAC). 2018. pp. 1-6
- [89] Liang J, Philip Wong H-S. Cross-point memory array without cell selectors—Device characteristics and data storage pattern dependencies. *IEEE Transactions on Electron Devices*. 2010; **57**(10):2531-2538
- [90] Huang C, Nuo X, Qiu K, Zhu Y, Ma D, Fang L. Efficient and optimized methods for alleviating the impacts of ir-drop and fault in RRAM based neural computing systems. *IEEE Journal of the Electron Devices Society*. 2021;**9**:645-652
- [91] Goodfellow I, Pouget-Abadie J, Mirza M, Bing X, Warde-Farley D, Ozair S, et al. Generative adversarial networks. *Communications of the ACM*. 2020;**63**(11):139-144
- [92] Ham T, J, Lisa W, Sundaram N, Satish N, Martonosi M. Graphicionado: A high-performance and energy-efficient accelerator for graph analytics. In: 2016 49th Annual IEEE/ACM International Symposium on Microarchitecture (MICRO). Taiwan, China: IEEE; 2016. pp. 1-13
- [93] Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, et al. Attention is all you need. In: 2017 31st Conference on Neural Information Processing Systems (NIPS). Long Beach, CA, USA: Neural Information Processing Systems Foundation; 2017. pp. 5998-6008