Received 8 March 2023; accepted 3 April 2023. Date of publication 17 April 2023; date of current version 3 November 2023.
The review of this article was arranged by Editor K. Ota.

Digital Object Identifier 10.1109/JEDS.2023.3265875

In-Memory Computing for Machine Learning and Deep Learning

N. LEPRI (Graduate Student Member, IEEE), A. GLUKHOV (Graduate Student Member, IEEE), L. CATTANEO (Graduate Student Member, IEEE), M. FARRONATO (Graduate Student Member, IEEE), P. MANNOCCI (Graduate Student Member, IEEE), AND D. IELMINI (Fellow, IEEE)

Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano and IU.NET, 20133 Milan, Italy

CORRESPONDING AUTHOR: D. IELMINI (e-mail: daniele.ielmini@polimi.it)

This work was supported by the EU's Horizon Europe Research and Innovation Programme under Grant 101070679.

ABSTRACT In-memory computing (IMC) aims at executing numerical operations via physical processes, such as current summation and charge collection, thus accelerating common computing tasks including the matrix-vector multiplication. While extremely promising for memory-intensive processing such as machine learning and deep learning, the IMC design and realization must face significant challenges due to device and circuit nonidealities. This work provides an overview of the research trends and options for IMC-based implementations of deep learning accelerators with emerging memory technologies. The device technologies, the computing primitives, and the digital/analog/mixed design approaches are presented. Finally, the major device issues and metrics for IMC are discussed and benchmarked.

INDEX TERMS In-memory computing, deep learning, deep neural network, emerging memory technologies, matrix-vector multiplication.

I. INTRODUCTION

Today, artificial intelligence and its enabling technology, the deep neural networks (DNN), have become largely popular in various applications such as image recognition, autonomous vehicles, speech recognition, and natural language processing. In the last five years, a state-of-the-art deep neural network model increased the number of its parameters by about 4 orders of magnitude, leading to a significant increase in computational and memory requirements for both the training and the inference operations [1], [2], [3], [4], [5], [6]. Traditional computing systems (Fig. 1a) typically store massive information on a memory unit that is physically connected to the computational unit by a data bus. The continuous data movement between the processing and the memory units represents the main bottleneck due to the limited bandwidth, long latency, sequential data processing, and high energy consumption [7], [8].

To minimize the latency and energy overhead of conventional von Neumann computers, in-memory computing (IMC) aims at performing the computation in close proximity to the memory or even in situ within the memory itself [9], [10]. The range of operations that can be executed within memory devices includes stateful logic [11], [12], pulse integration [13], [14], associative memory [15], [16], and stochastic computing [17]. The most popular and enabling IMC operation is, however, matrix-vector multiplication (MVM) via Ohm's and Kirchhoff's law in a memory array [18], [19]. IMC has been thus largely targeted for hardware accelerators of DNN, where MVM is by far the most intensive workload. The ability to execute MVM in a single operation by activating all rows and all columns in parallel represents a key benefit of IMC that is unrivaled by other technologies. Despite the simplicity of the MVM concept and the potential advantages of IMC, the design options and the interaction between circuit operation and device nonidealities still represent a key open challenge.

This work provides an overview of IMC for DNN acceleration from the perspectives of device technology, circuit design, device-circuit interaction, and its impact on computing accuracy. Section II illustrates the emerging nonvolatile memory technologies that are currently considered for IMC. Section III presents an overview of various IMC circuit

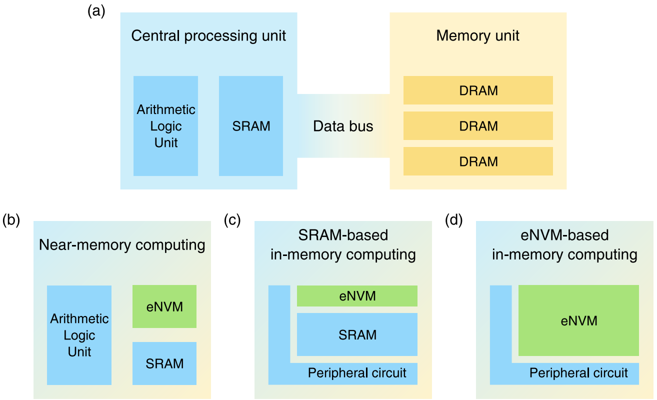


FIGURE 1. Several examples of CPU - memory integration. (a) Von Neumann architecture, in which CPU and memory are separated and connected through a high-bandwidth bus, (b) Near-memory computing, which features the embedding of a nonvolatile memory on the same silicon as the CPU, for increased bandwidth and reduced data transfers, (c) SRAM-based in-memory computing, in which the computation is performed directly in the SRAM memory array, and (d) eNVM-based in-memory computing, which features the integration of a high-density memory allowing both parameter storage and calculation.

topologies for performing matrix-vector multiplication and their possible applications. Among these applications, the most promising one is the IMC acceleration of DNN inference, discussed in Section IV. Hence, Section V illustrates the most critical device nonidealities affecting the accuracy of IMC circuits. Section VI provides an overview of the open challenges for the research field, while Section VII concludes the work.

II. COMPUTATIONAL MEMORY TECHNOLOGIES

The main benefit of IMC is the improved energy efficiency thanks to the reduction or suppression of data movement. A first option to mitigate data movement is to bring the main memory core directly on the chip via high-density embedded DRAM [20] or embedded nonvolatile memory (NVM). This approach, called near-memory computing and depicted in Fig. 1b, allows the storage of even megabytes of model parameters, such as synaptic weights and activations, in close proximity to the processing unit. A second option [21], [22], [23] is true IMC where computation is executed directly within the SRAM array as shown in Fig. 1c. A key limitation of this option is the volatile nature of SRAM and the relatively low density compared to DRAM and emerging NVM. In fact, each SRAM cell consists of at least 6 transistors and the bit value remains stored only until the power supply is switched off. To overcome these limitations, the third option embraces emerging NVM devices for both nonvolatile storage of computational parameters and in situ MVM acceleration (Fig. 1d).

Here, we will focus on emerging NVM technologies that are suitable for the IMC concept of Fig. 1d. In general, these devices have three major advantages, namely (i) nonvolatile storage which allows for the persistence of synaptic weights even when the supply is disconnected, (ii) integration in the back-end of line (BEOL), which allows compatibility of the NVM process irrespective of the details of the front-end technology and (iii) high density compared to SRAM. The major NVM technologies for IMC applications are sketched in Fig. 2.

The resistive-switching random access memory (RRAM in Fig. 2a) consists of a metal-insulator-metal (MIM) stack, where the insulator serves as active switching material [24]. The memory operation relies on the activation and deactivation of a conductive filament across the switching layer [25]. RRAM generally displays binary states, referred to as low resistance state (LRS) and high resistance state (HRS) [26]. However, RRAM can also display multilevel operation [27] where the conductance can be tuned in the analog domain [28]. RRAM devices can be easily integrated into crosspoint arrays [25] and scaled down to 22nm CMOS technology [29].

The phase change memory (PCM in Fig. 2b) relies on the ability to electrically change the crystalline/amorphous phase of an active chalcogenide material, where the resistance correspondingly changes by at least two orders of magnitude [68]. The most typical material is $Ge_2Sb_2Te_5$ (GST) [69], although Ge-rich alloys are adopted for high-temperature retention in embedded solutions [70]. The phase change is induced by Joule heating via the application of voltage pulses. If the local temperature exceeds the melting temperature, the resulting phase is amorphous, corresponding to a HRS. If instead, the local temperature is below the melting temperature for sufficient time, the structure stabilizes to crystalline, corresponding to LRS [71]. Thanks to the relatively mature technology, these devices have been extensively used for IMC demonstrators [72].

The ferroelectric random access memory (FeRAM in Fig. 2c) consists of a metal-ferroelectric-metal (MFM) structure, where the ferroelectric layer exhibits a permanent and switchable electrical polarization [73]. FeRAM has received renewed interest after the discovery of ferroelectric hafnium oxides HfO_2 with orthorhombic structure [74]. A key issue with FeRAM is its destructive readout operation, due to reading being performed above the coercive field. This limitation is overcome by ferroelectric tunnel junction (FTJ), where different polarization states seem to show different resistances even at low voltages [75].

The spin-transfer torque magnetic random access memory (STT-MRAM in Fig. 2d) consists of a MIM stack where the top and bottom metals are ferromagnetic (FM) metals, such as Fe, Co, Ni, and their alloys. The MIM displays a magneto-tunnel junction (MTJ) effect, where different orientations of the magnetic polarization in the two FM layers, namely a parallel (P) or antiparallel (AP) state, result in a LRS or HRS, respectively [76]. STT-MRAMs feature fast switching and good cycling endurance [77], despite suffering from a relatively small resistance window and difficult multilevel operation, which limits the use of STT-MRAM to binarized neural networks.

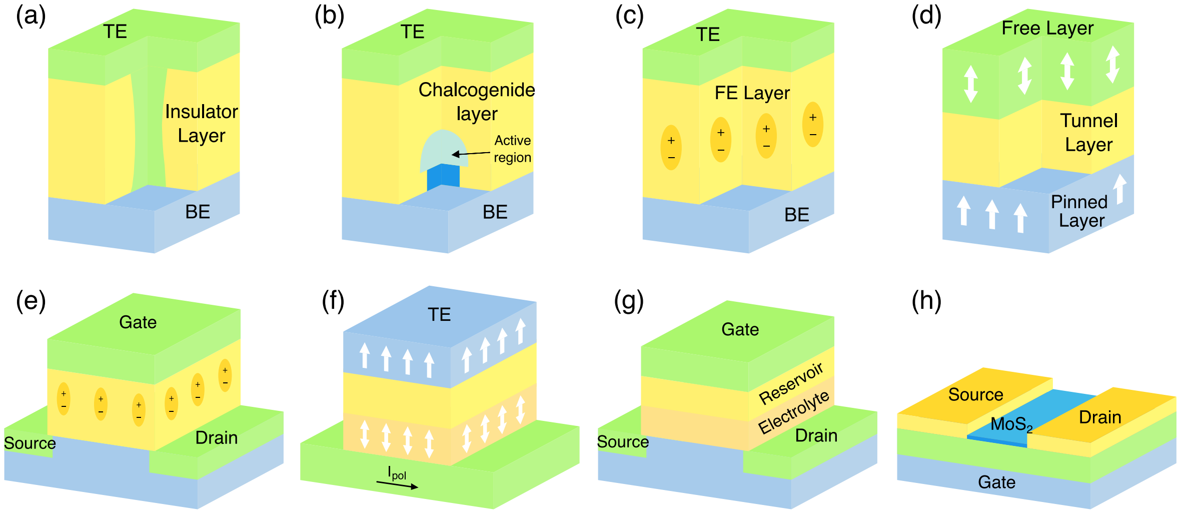


FIGURE 2. **Graphic representation of the main emerging memory devices. (a) Resistive random access memory (RRAM). (b) Phase change memory (PCM). (c) Ferroelectric random access memory (FeRAM). (d) Spin-transfer torque magnetic random access memory (STT-MRAM). (e) Ferroelectric field-effect transistor (FeFET). (f) Spin-orbit torque magnetic random access memory (SOT-MRAM). (g) Electrochemical random access memory (ECRAM). (h) Memtransistor device.**

Devices in Fig. 2a-d have a two-terminal structure, which makes them suitable for high-density crosspoint architectures [10]. In many cases, two-terminal devices are connected to an access transistor resulting in a one-transistor/one-resistor (1T1R) structure with improved control of the device current during programming and readout. Alternatively, three-terminal devices have been proposed. The ferroelectric field-effect transistor (FeFET in Fig. 2e) consists of a field-effect transistor in which the gate stack contains a ferroelectric layer [78]. The ferroelectric polarization is reflected by the threshold voltage V_T of the device, resulting in a memory effect similar to floating gate devices. FeFET arrays with ferroelectric HfO_2 have been recently demonstrated [35], [79].

The spin-orbit torque magnetic random access memory (SOT-MRAM in Fig. 2f) consists of a magnetic tunnel junction (MTJ) structure deposited on top of a line of heavy metal, such as Pt or W [80]. The MTJ is programmed in a P/AP state by a current flowing across the heavy-metal line via spin-orbit coupling. The cell is read by sensing the MTJ resistance, as in the STT-MRAM. The three-terminal structure allows the separation of the programming and the reading paths, improving the cycling endurance and the write speed [81].

The electrochemical random access memory (ECRAM in Fig. 2g) consists of a transistor device where the conductivity of the channel is modified in a nonvolatile way and can be reversed by injecting ionized dopants across an electrolyte layer [82]. ECRAM generally shows high endurance and extremely low-power consumption thanks to the low mobility channel, for instance, WO_3 [83]. ECRAM also exhibits a controllable, linear weight update that is suitable for training accelerators [82], [84].

The memtransistor (Fig. 2h) consists of a transistor device with a 2D semiconductor material for the channel layer [85], [86], [87]. The memory behavior can be obtained by migration of dislocations in polycrystalline MoS_2 [88], lateral migration of Ag across the source/drain electrodes [85], or charge-trapping [89]. In some cases, MoS_2 memtransistors display gradual weight-update characteristics that are useful for reservoir computing [89] and training accelerators [90].

A. COMPARISON OF NVM TECHNOLOGIES

In order to summarize and provide some quantitative information, Table 1 shows a comparison between the main emerging memories and the charge-based CMOS memories [91]. Fig. 3a shows a correlation plot of speed, evaluated as the inverse of the read time, and density, evaluated as the inverse of the cell area. Data from the literature are compared to the typical ranges for CMOS-based conventional memory technologies, such as SRAM, DRAM, and NAND Flash. The performance/cost of emerging NVM is usually intermediate between CMOS memories, where speed approaches DRAM whereas density is still generally between SRAM and DRAM.

Fig. 3b shows the array size as a function of the technology node for various NVM demonstrators. The capacity spans the whole range from embedded memory (1-100 MB) to standalone memory (1-100 GB). Note that smaller technology nodes do not necessarily lead to higher array capacity, which is due to the different maturity levels of the technologies. Fig. 3c shows the memory capacity of some NVM demonstrators as a function of the year, highlighting the continuous development of various memory technologies.

III. IN-MEMORY MATRIX-VECTOR MULTIPLICATION

Most IMC implementations aim at accelerating matrix-vector multiplication (MVM), which is by far the most essential computing primitive in deep learning and machine learning [92]. Fig. 4 shows a sketch of the MVM concept

TABLE 1. Indicative performances and characteristics of different semiconductor memory technologies.

	NOR flash	NAND flash	DRAM	SRAM	RRAM	PCM	FeRAM	FeFET	STT-MRAM	SOT-MRAM
Cell Size (planar)	10 F^2	4-5 F^2	6-8 F^2	150-500 F^2	4-100 F^2	4-50 F^2	6-50 F^2	6-50 F^2	4-130 F^2	12-100 F^2
Write time	10-100 µs	0.1-1 ms	<10 ns	0.1-1 ns	5-50 ns	∼ 50 ns	∼ 30 ns	∼ 10 ns	5-50 ns	<10 ns
Read time	∼ 50 ns	10-100 µs	1-10 ns	0.1-1 ns	<10 ns	<10 ns	<30 ns	∼ 10 ns	5-50 ns	1-10 ns
Write energy	∼ 200 pJ/bit	∼ 10 fJ/bit	>10 fJ/bit	∼ 1 fJ/bit	1-100 pJ/bit	30-300 pJ/bit	10-100 fJ/bit	>1 fJ/bit	10-30 pJ/bit	>1 pJ/bit
Endurance	$\sim 10^5$	$\sim 10^4$	$>10^{16}$	$>10^{16}$	10^5-10^8	10^6-10^9	$\sim 10^{10}$	10^5-10^9	$\sim 10^{15}$	$\sim 10^{12}$
Data Retention	Nonvolatile	Nonvolatile	Volatile, dynamic with t_{RET} >100 ms	Volatile	Nonvolatile with weak drift	Nonvolatile with drift	Nonvolatile	Nonvolatile	Nonvolatile and volatile	Nonvolatile

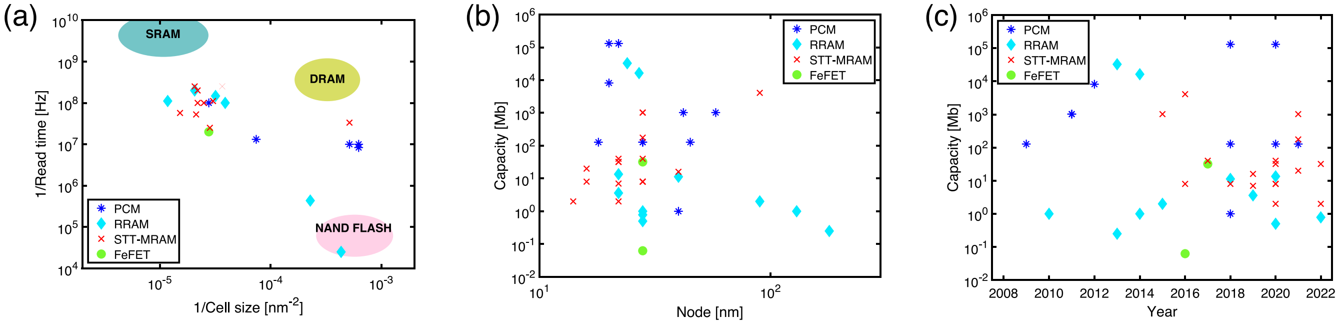


FIGURE 3. Performances and characteristics of various emerging memory demonstrators. (a) Memory speed (expressed as the inverse of the read time) as a function of the device miniaturization (expressed as the inverse of the cell size) [29], [30], [31], [32], [33], [34], [35], [36], [37], [38], [39], [40], [41], [42], [43], [44], [45], [46], [47], [48], [49], [50]. (b) Memory capacity as a function of the technology node [29], [30], [31], [32], [33], [34], [35], [36], [37], [38], [39], [40], [41], [42], [43], [44], [45], [46], [47], [48], [49], [50], [51], [52], [53], [54], [55], [56], [57], [58], [59], [60], [61], [62], [63], [64], [65], [66], [67]. (c) Memory array capacity during years [29], [30], [31], [32], [33], [34], [35], [36], [37], [38], [39], [40], [41], [42], [43], [44], [45], [46], [47], [48], [49], [50], [51], [52], [53], [54], [55], [56], [57], [58], [59], [60], [61], [62], [63], [64], [65], [66], [67].

implemented in a crosspoint memory array. The applied voltage signals generate currents across each resistive element that are given by the voltage-conductance multiplication according to Ohm's law. Currents reaching a grounded common row are the summation of individual cell currents according to Kirchhoff's current law (KCL). The output current I_i at the ith row is thus given by:

$$I_i = \sum_{j}^{N} G_{i,j} \cdot V_j, \tag{1}$$

where $G_{i,j}$ is the conductance of the memory device at a certain position i, j, V_j the voltage applied at the jth column and N is the number of columns and rows [10], [93].

MVM can thus be carried out by physical laws, in situ, without modifying or moving the stored parameters [10]. Most importantly, thanks to the inherent parallelism of the array, the MVM computation is virtually performed in one step independently of the size of the matrix, thus achieving an outstanding time complexity of $O(1)$. Note that the memory array is typically compatible with the BEOL process, allowing for 3D stacking and a memory density scalable down to $4F^2/N$, where N is the number of stacked layers and F is the feature size of the lithographic process.

Depending on the required specifications and the memory devices, various IMC implementations of MVM accelerators are possible. Fig. 5a shows the resistive crosspoint array, similar to Fig. 4, where device conductances can be programmed in the binary [94], [95] or multilevel domain [96], [97]. Steady-state currents collected at the grounded rows are generally acquired by a readout chain consisting of a transimpedance amplifier (TIA) and an analog-to-digital converter (ADC) [98]. A major limitation of this architecture is the programming operation, where voltage/currents might be difficult to control [99]. In particular, when applying various programming schemes [100], [101], a certain number of half-selected cells experience a non-negligible leakage current.

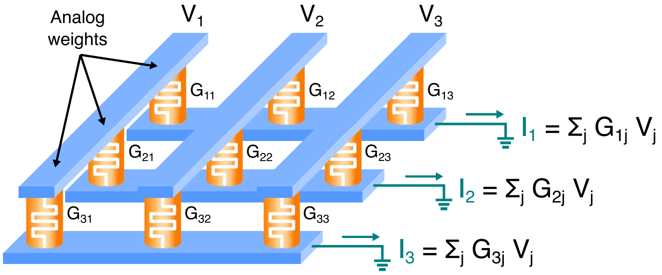


FIGURE 4. Crosspoint memory array based on resistive memories can perform matrix-vector multiplication directly in situ, by means of Ohm's law and Kirchhoff's current law. By applying a voltage vector at the columns, the analog conductive elements produce a current that is collected at the rows, conveniently biased at 0 V. The resulting output current vector is the multiplication of the conductance matrix G with the voltage vector V.

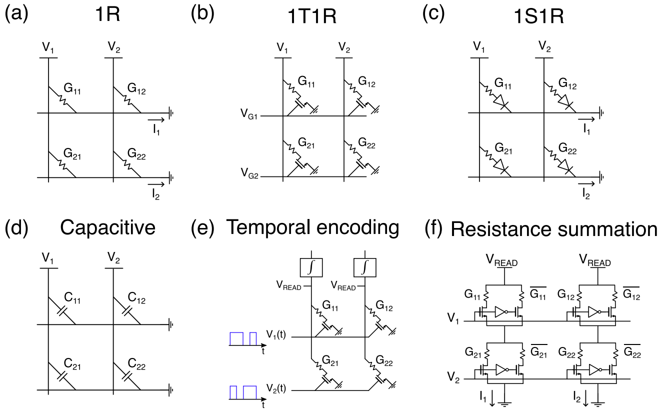


FIGURE 5. Various implementations of IMC crosspoint accelerators of MVM. (a) Resistive crosspoint array (1R). (b) Array with one-transistor/one-resistor (1T1R) configuration. The transistor prevents sneak path currents during the programming phase and allows finer current control. (c) Array with one-selector/one-resistor (1S1R) configuration. The highly non-linear selector prevents sneak path currents, maintaining the cell footprint. (d) Capacitive crosspoint array, composed of memory elements whose small-signal capacitance can be programmed. (e) Temporal encoding of the input vector through gate voltage pulses whose widths represent the input signals. Integration is required to collect the transient currents. (f) MVM through resistance summation. An XNOR-Multiply is performed by the 2T2R cell, which activates the path corresponding to the multiplication result. The series of the resistive paths inherently performs the accumulation.

To address these programming issues, an access device is normally added in series to the resistive element. Fig. 5b shows the 1T1R configuration, which allows finer control of the program/read current, at the cost of a larger cell footprint and of an additional line for the transistor gate terminal [102], [103]. Fig. 5c shows the one-selector/one-resistor (1S1R) configuration [104], [105]. A selector is a non-linear element capable of suppressing the leakage, also called sneak path, currents of half-selected cells during the programming phase, while maintaining a small cell footprint and a compact two-terminal configuration [106], [107].

Fig. 5d illustrates a crosspoint array based on capacitive memory elements, whose small-signal capacitance can be programmed. In this configuration, MVM computation is typically carried out in two distinct phases. First, the capacitors are pre-charged by applying a voltage proportional to the input vector. Then, the capacitors are discharged by switches placed at the end of columns and rows, while the accumulated charges are collected by analog integrators [108]. In this case, multiplication is carried out by the characteristic law of the capacitance, namely $Q_{i,j} = C_{i,j} \cdot V_j$, where $C_{i,j}$ serves as the weight and V_j is the applied input/activation.

The input signals can generally be encoded either in the voltage amplitude, through amplitude encoding, or in the pulse width, through temporal encoding. This approach, shown in Fig. 5e, is typically implemented in 1T1R arrays, where the memory elements are subject to a fixed voltage V_{READ} while the input signals are applied to the transistor gates. By integrating the transient currents on a capacitance or through the adoption of analog integrators, the resulting voltage output will be proportional to the MVM result [72].

Kirchhoff's voltage law (KVL) can be used instead of KCL for accumulation [109]. This is shown in Fig. 5f, where the adoption of a 2T2R cell configuration enables a binary XNOR multiplication between the input voltage and the conductance. The multiplication activates only one of the two paths, showing a LRS or an HRS depending on the result of the multiplication. By sensing the series resistance summation at each column, it is possible to collect the results of the MVM.

A. APPLICATIONS OF IMC MVM ACCELERATORS

Since MVM is ubiquitous in a variety of algorithms and workloads, IMC circuits to accelerate MVM have thus been demonstrated in several data-intensive computing tasks, as schematically depicted in Fig. 6.

Applications include image processing and image compression (Fig. 6a) via the discrete cosine transform (DCT). Here, image processing/compression can be achieved by applying the concept of MVM between a fixed DCT matrix and the pixel intensity input vector, preserving only frequencies within a desired frequency band based on the compression ratio [19], [111].

In closed-loop IMC (CL-IMC), the MVM array core is connected in the feedback loop of an array of operational amplifiers (OAs), as shown in Fig. 6b [112]. This class of circuits allows the acceleration of a broad range of linear algebra operations, such as matrix inversion [113], eigenvector extraction [114], linear regression [115], and ridge regression [116] with a significant reduction in time complexity.

Combinatorial optimization (Fig. 6c) relies on the intrinsic noise of the memory elements and the peripheral circuit as an on-chip source of entropy to carry out a physical simulated annealing to escape from local minima during the iterative search [117]. In these applications, MVM accelerators are typically used in recurrent architectures to map restricted Boltzmann machines (RBM) [13], [118], [119] or Hopfield neural networks [120], [121], [122]. Similarly, Bayesian neural networks (Fig. 6d) rely on the intrinsic variations of programmed conductance to model the probability distributions of a Bayesian network [123].

The most popular application for MVM remains DNN inference (Fig. 6e) and training (Fig. 6f). A key difference between these applications is that synaptic weights are obtained from ex situ software-based training in the case of inference accelerators, while they are trained in situ via iterative gradient descent algorithms in the case of DNN training accelerators. Typically, a training accelerator is capable of performing inference via forward propagation, while featuring also an in situ weight-update scheme generally via vector-vector outer product within the crosspoint array [124]. Weight update requires linearity and symmetry of the conductance update under the application of a sequence of identical pulses, in line with the backpropagation algorithm. The best candidate materials to yield a linear

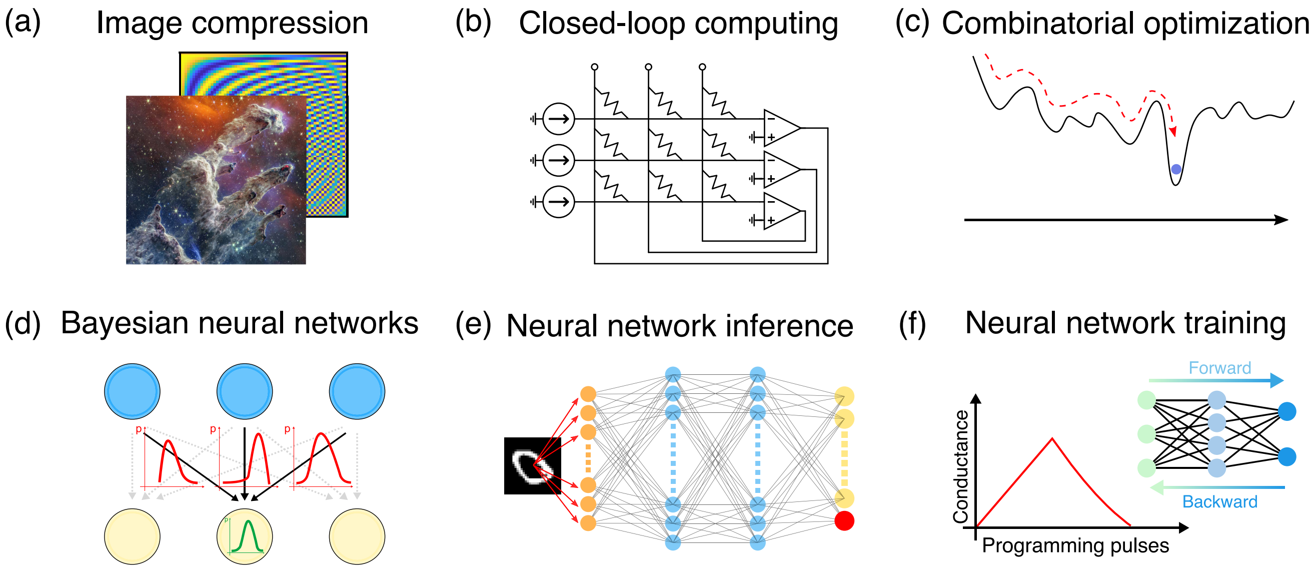

(a) Image compression
(b) Closed-loop computing
(c) Combinatorial optimization
(d) Bayesian neural networks
(e) Neural network inference
(f) Neural network training
Forward
Backward
Conductance
Programming pulses

FIGURE 6. Example of applications that benefit from IMC matrix-vector multiplication. Depending on the frequency update requirements and the noise sensitivity of the application, each hardware solution should combine memory devices with specific physical properties with adequate peripheral circuits. For instance, applications that rely on one-time programming of weight values after an ex situ software-based training (e.g., DNN inference, CL-IMC, and DCT) can trade off the need for accurate tuning algorithms with less stringent requirements on the cycling endurance of the device itself. On the other hand, applications that demand frequent and continuous updates of the conductance matrix (e.g., DNN training) require efficient gradual programming and endurance capabilities of the adopted memory device. Image "Pillars of Creation" from James Webb Space Telescope gallery [110].

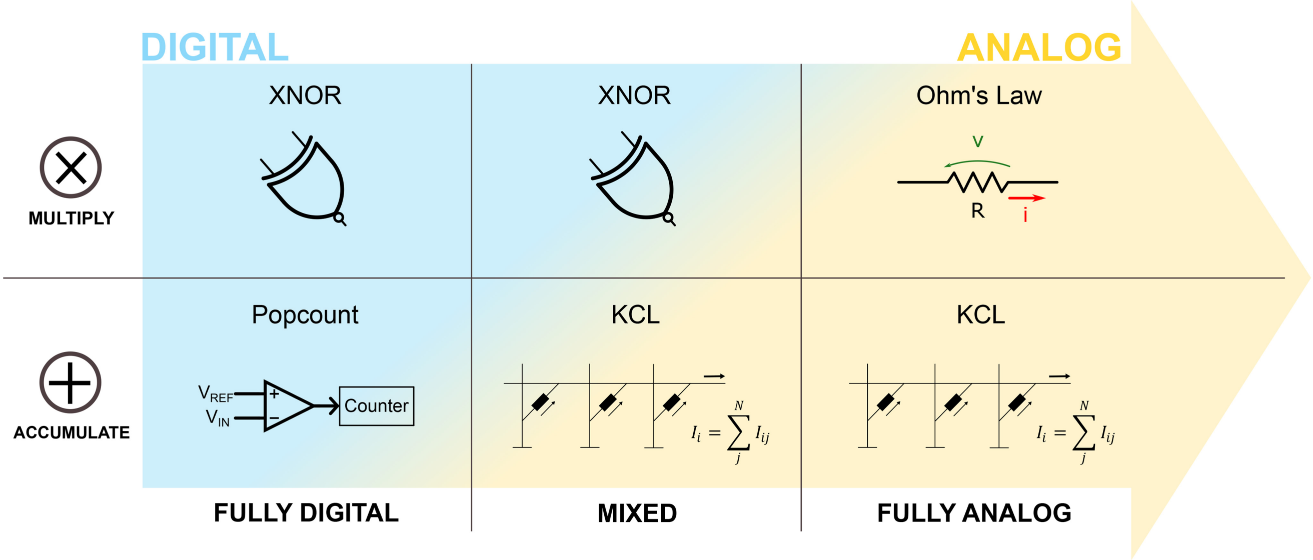


FIGURE 7. DNN inference workload mainly consists of MVM, which is basically a Multiply-and-Accumulate operation. Crosspoint accelerators of DNN inference can be classified depending on the way these two operations are performed. A fully digital approach relies on memory logic gates implementing an XNOR-Multiply and on a counter for the accumulation. A mixed digital-analog approach requires an analog accumulation via Kirchhoff's current law (KCL). A fully analog approach relies on resistive elements, that allow the encoding of multilevel weights and activations. Going from digital to analog, the parallelism and the information density of the accelerator increase, at the expenses of more severe parasitic effects and more complex peripheral circuits. Further explorations of the fully analog approach are needed to unleash the potential of IMC for DNN inference acceleration.

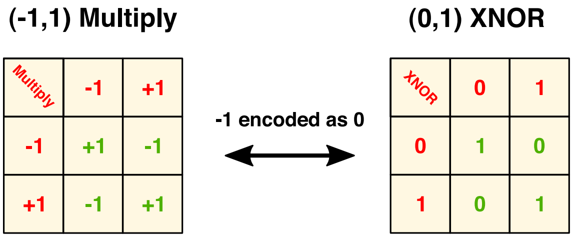
weight update are ECRAM devices [125] and MoS_2-based charge-trap memory [89], [90].

IV. IN-MEMORY ACCELERATION OF DNN INFERENCE

The computational workload of a DNN mostly consists of MVM with variable input vectors and stationary weight matrices, which can be directly accelerated by a memory array. Depending on multiply and accumulate operations being performed by analog or digital operations, three different options can be identified for MVM accelerators, as depicted in Fig. 7.

A. FULLY DIGITAL CIRCUITS

The fully digital approach relies on memory logic gates to perform the multiplication, and on counters to perform the sequential accumulation. To encode the binary alphabet of a



Multiply	-1	+1
-1	+1	-1
+1	-1	+1

XNOR	0	1
0	1	0
1	0	1

FIGURE 8. Binary DNN usually adopt a (−1, 1) alphabet for activations and weights. A multiplication in the (−1, 1) alphabet can be implemented in the classical (0, 1) binary domain through an XNOR logic gate, encoding −1 as 0.

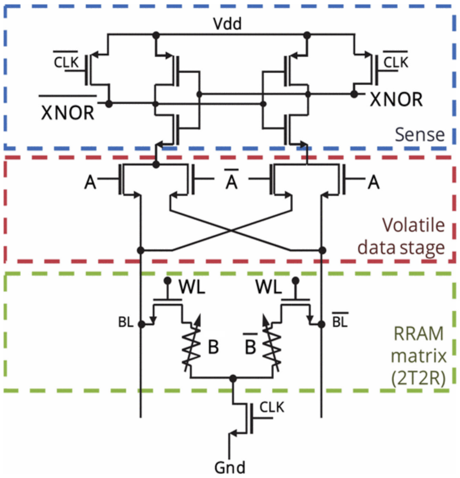


FIGURE 9. Error-resilient implementation of an XNOR in a fully digital accelerator, based on a differential 2T2R RRAM cell. The activation signal *A* enables the connection of the cell to the sense amplifier in a straight or crossed path, allowing the sense amplifier to perform the comparison between the two resistive states. Adapted with permission from [128].

binary neural network (BNN) [126], where activations and weights can be -1 or 1, the logic gates usually implement an XNOR operation, that allows mapping a $(-1, 1)$ multiplication in the classical $(0, 1)$ binary domain [127], as schematically shown in Fig. 8.

Digital accelerators have been proposed with various nonvolatile emerging memories, such as RRAM [128], [129], [130], STT-MRAM [131], and FeFET [132]. The memory logic gate is generally based on a single 1T1R or differential 2T2R cell.

Fig. 9 shows a building block based on differential 2T2R, also displaying the XNOR gate and the sense amplifier (from bottom to top). The binary weight is stored as a resistive pair (HRS, LRS) or (LRS, HRS) in the 2T2R cell. For instance, to map a weight equal to 1, the memory element corresponding to B is programmed to LRS while its complementary $\overline{B}$ is programmed to HRS. The activation (input) signal A and its complementary $\overline{A}$ connect the 2T2R cell to the sense amplifier in a straight path, for input $\mathrm{A} = 1$, or crossed path, for input $\mathrm{A} = 0$. When the clock signal closes the conductive path to ground, the cross-coupled latch of the sense amplifier compares the resistive states of the memory elements and raises the voltage in one of the two output nodes, while decreasing the other one. For instance, assuming $A = 1$ and $B = 1$, the $XNOR$ node potential increases while $\overline{XNOR}$ decreases. The XNOR output is then digitally counted by a popcount operation. Thanks to the binary comparison of the two device resistances in the 2T2R structure, the memory cell is resilient to drift, noise, device variability, and temperature variations [128], [130].

SRAM-based digital accelerators have also been demonstrated with various memory cells, ranging from six-transistor (6T) cells to twelve-transistor (12T) cells [21], [22], [23], [133]. While providing only volatile storage of weights, SRAMs provide the advantage of a fully-CMOS integration which can be manufactured even for extremely scaled technology nodes, such as 5nm [133].

In general, the fully digital approach is exceptionally robust to various nonidealities, such as device variability, drift, noise, or IR drop, and it can have higher reconfigurability [134], [135], [136]. However, because of the accumulation through counting, the parallelism of the computation is limited to just one row at a time, thus limiting the available throughput.

B. MIXED DIGITAL-ANALOG CIRCUITS

In a mixed digital-analog circuit for DNN acceleration, accumulation is performed in the analog domain by KCL, thus avoiding the sequential counting of the pulses, while multiplication remains implemented in the digital domain by an XNOR gate.

The mixed approach has been demonstrated either with emerging NVM, such as RRAM [137], [138], [139], [140], [141] or FeFET [142], [143], or with various SRAM cells, generally from 6T to 12T [23], [144], [145], [146], [147]. As for the fully digital approach, the XNOR gate can adopt the 1T1R cell [137], [140] with various differential techniques with NVM [139], [142], [148] or SRAM, whose output result is typically stored in a capacitor as a binary charge quantity [145], [149], [150].

Fig. 10 shows the computing core of a mixed digital-analog accelerator based on SRAM. XNOR is implemented in an eight-transistors/one capacitance (8T1C) cell, where the weight a and its negated ab are stored in the SRAM, while activations x and its negated xb are applied at the PMOS transistors connected to the cell capacitance [145]. Assuming $a = 1$ and $x = 0$, the complementary node ab is shorted to the capacitance, setting the output voltage to $0\ V$. Accumulation is then performed through charge sharing of all cell capacitors to the shared bitline [145], [151], [152]. Alternatively, charge accumulation has been proposed by charge redistribution on weighted capacitances [150], [153], [154].

When the multiplication results are produced in the form of steady state currents instead of charge, it is sufficient to collect them through a common node, exploiting KCL, and acquire the output current sums through a readout circuit [137], [138], [139], [140]. Depending on the BNN,

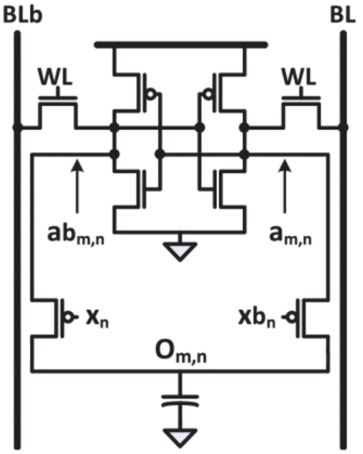


FIGURE 10. Implementation of an XNOR with an 8T1C SRAM cell in a mixed digital-analog accelerator. Analog accumulation is performed by charge sharing on a shared bitline. Reprinted with permission from [145].

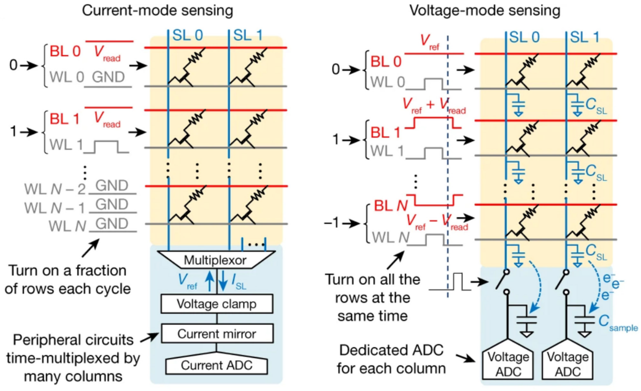


FIGURE 11. Possible implementations of the computing core in a fully analog approach based on 1T1R configuration, relying on current and charge accumulation, respectively. Reprinted with permission from [155].

the resulting current sum can also be directly compared to a reference current by means of a sense amplifier [138], thus operating a threshold-type activation function. When adopting differential NVM cells, another proposed method to perform accumulation consists in implementing a voltage divider composed of pull-up or pull-down resistances according to XNOR results [141], [147], [148], and then acquiring the common node voltages, which are proportional to the result of the MVM.

Overall, a mixed digital-analog approach takes advantage of the inherent parallelism of IMC, virtually reaching a computational complexity of $O(1)$. However, the analog accumulation requires a more complex peripheral circuitry, often involving a bulky and energy-hungry readout chain, and is more sensitive to parasitic effects, such as IR drop and noise. Furthermore, when the multiplication relies on a single NVM, without conductance comparisons or error-resilient circuits, also device variability and drift can affect the computation.

C. FULLY ANALOG CIRCUITS

Fully analog circuits perform accumulation by KCL and multiplication by resistive memory elements via Ohm's law. The adoption of resistive memory elements limits possible implementations to NVM technologies only, since SRAM cells cannot provide ohmic behavior or work with analog voltages. NVM-based analog accelerators have been implemented with RRAM [94], [156], [157], [158], PCM [72], [159], [160], [161], STT-MRAM [103], [162] and FeFET devices [163].

Thanks to the multilevel operation, resistive memories are suitable for implementing non-binary weights in the same circuit footprint, thus enabling a higher area efficiency, defined as the number of performed operations per area unit. Indeed, memory elements can be programmed in binary [103], [163] or multilevel mode [102], [160]. Alternatively, multilevel weights are obtained through bit-slicing techniques [156], [157], [162], differential implementations, or more complex cell structures, allowing several conductive levels to be obtained [159]. Also, a hybrid binary-multilevel accelerator has been proposed to achieve the best trade-off between accuracy and area efficiency [161]. Alongside the increase in the number of conductive levels, memory cells can contain a variable number of elements, for instance, 1T1R cell [95], [102], [164], differential 2T2R cell [158], or higher-complexity cells such as 8T4R [159]. In addition to multilevel weights, analog accelerators typically feature multilevel or analog activation signals that can be modulated through amplitude [102], [159] or temporal encoding [72], [156], [165].

Fig. 11 shows two possible implementations of fully analog circuits, that rely on either current or charge accumulation. Current-mode sensing requires applying a clamped voltage to the source lines, thus generating current contributions in each 1T1R cell that are collected and converted to a voltage by the current ADC. On the other hand, voltage-mode sensing consists of two separate phases. First, the multiplication results are stored in the source line capacitances, then they are accumulated into a sample capacitance by charge sharing. The voltage across the sample capacitance is finally collected by the ADC [155].

Fully analog accelerators can harness the full potential of IMC, thanks to the massive parallelism and the extremely high information density of multilevel weights and activations. On the other hand, accurate readout and conversion circuits are essential to fully benefit from these features, resulting in a significant overhead of area, power, and cost. Furthermore, analog computing is critically affected by parasitic effects at device and circuit levels.

V. MEMORY NONIDEALITY AND METRIC

Memory devices and circuits rely on physical, materials-based storage concepts that are never ideal. Fig. 12 summarizes the main nonideality features, namely IR drop (a), conductance variation (b), drift (c), and fluctuations (d).

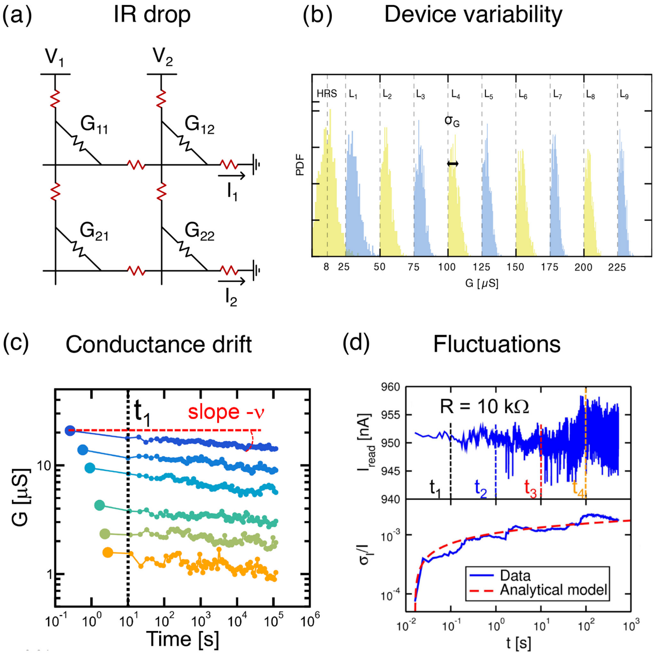


FIGURE 12. Examples of memory nonidealities. (a) Parasitic resistance along wire connections responsible for the IR drop, (b) programming variability in multilevel programming, (c) conductance drift that affects the cells, and (d) Gaussian noise that is measured during the readout of a RRAM cell. Reprinted with permission from [166], [167], [168].

IR drop refers to the current-induced voltage drop across parasitic wire resistances along the rows and the columns of the memory arrays (Fig. 12a). Wire resistance is non-negligible in very scaled arrays because of the small section of the metal lines. Furthermore, analog accumulation in parallel IMC requires several cells to be read at the same time, thus increasing the wire current, hence the IR drop. IR drop causes a modification of the effective cell voltage compared to the externally applied signal, thus resulting in a current error that is proportional to the average device conductance, to the wire resistance, and to the square of the array size [99], [169]. In practice, the error induced by IR drop is the main limitation to array size up-scaling, thus preventing reaching the ideal computational complexity of $O(1)$. Generally, IR drop is reduced by adopting low conductance devices, differential cells [158], or small computing-tile architectures [169]. More elaborated techniques have been proposed at architectural level [170], [171], algorithmic level [169], [172], [173], and training level [174].

Multilevel operation allows the improvement of area efficiency [28], [175], [176]. However, NVMs have limited precision in programming the conductance, for instance, due to size variations of the conductive filament in RRAM or crystalline grain size in PCM. The limited precision arises as a device-to-device (D2D) variability or a cycle-to-cycle (C2C) variability within the same device [177], [178]. D2D variability is shown in Fig. 12b, reporting a multilevel RRAM device with a non-negligible spread of the conductive states. Differently from the digital domain, where binary levels can be discriminated despite a possible spread, computing in the analog domain can be critically affected even by a small variation.

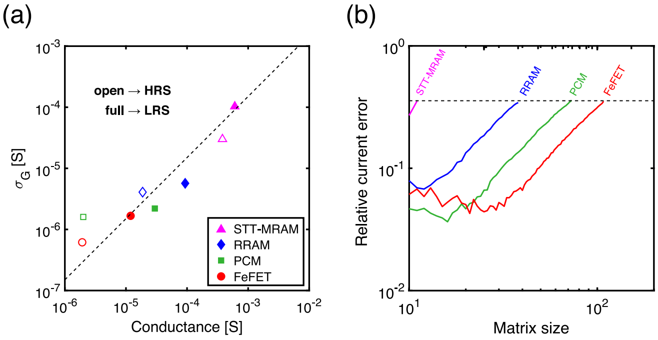


FIGURE 13. (a) Plot of the correlation between the conductance value G, and its standard deviation, for a certain technology. (b) The simulated relative current error of the MVM product as a function of the matrix size. Device parameters were extracted from [79], [166], [184], [185].

Drift is generally observed in PCM, where the structural relaxation of the amorphous phase causes an increase in resistance with time [179]. Drift can also affect the polycrystalline phase in multilevel PCM devices, as a result of residual amorphous regions [167], [180]. Fig. 12c shows the temporal decay of conductance of multilevel analog states, described by their slope ν on the bilogarithmic plot. Drift is also observed in other devices, such as RRAM and FeFET, although the physical mechanism is different from PCM. Drift can be mitigated by adopting reference PCM cells [140], [181], [182], [183] or differential 2T2R structures [128].

Finally, various sources of noise and fluctuations may affect NVM devices. For instance, Fig. 12d shows the 1/f current noise of RRAM devices, causing an increasing relative spread of the measured current [168]. In addition to 1/f, thermal and random telegraph noise (RTN) can contribute to time-dependent variations of the weights, thus affecting the accuracy of the analog MVM. Noise might be mitigated by adopting analog integration of the readout current, although at the cost of reduced speed of computation.

To properly benchmark various NVM technologies for use in mixed or fully analog DNN accelerators, it is important to set a common metric. To this purpose, Fig. 13a shows a correlation plot between the average conductance value G and the standard deviation σ_G. Data were obtained for various NVM devices, including FeFET [79], PCM [185], RRAM [166], and STT-MRAM [184]. The conductance G should be minimized to reduce readout currents, hence energy consumption and IR drop effects. Similarly, σ_G should be minimized to improve the computing accuracy in analog/mixed circuits. The observed trend in the figure is that σ_G and G approximately correlate with a formula $\sigma_G/G \approx 0.15$, irrespective of the NVM technology and the programmed state. Fig. 13b illustrates the relative current error for an MVM operation in the presence of variations and IR drop as a function of the array size for the NVM devices in Fig. 13a. For relatively small array sizes, the error decreases as a result of variability averaging among NVM devices. As the array size increases, IR drop causes the error to steeply increase. The optimum size of the array, which is identified in correspondence with the minimum error, is dictated by σ_G and G, which control variability and IR drop, respectively.

VI. OUTLOOK

IMC circuits are dense, fast, energy-efficient, and scalable. Several solutions and applications have already been identified and explored for both machine learning and deep learning. However, various technological and design challenges have also been identified. Further development and industrialization of IMC require addressing these challenges in two major directions.

The first direction concerns the study of device technology and materials. IMC paradigm would greatly benefit from the adoption of precise, stable, and low-current memory devices that could be easily integrated in the BEOL of extremely scaled lithographic processes, while also being programmable in multiple conductive levels. Investigation of materials and device physics can enlighten the phenomena underlying nonidealities such as fluctuations and drift, with the aim of developing new memory devices which are immune from parasitic effects. Besides device developments, the engineering of the memory cell configuration, such as 1S1R or 1T1R structure, could drastically reduce the operating current, with strong advantages in terms of lower energy consumption, lower IR drop, and higher area efficiency of the IMC system. In summary, developments at the device level would boost IMC performance in terms of increased information density, throughput, area, and energy efficiency.

The second direction to be explored is the study of computing architectures and their interplay with the workload. To maximize the system performance, computing parallelism should be maximized to prevent multiplexing of the readout chain. This approach is usually challenging since peripheral circuits consume the largest portion of energy and area budget. However, these limitations could be relaxed by an accurate co-design of the hardware and the neural network. On the one hand, IMC circuits must be designed specifically for an application, thus avoiding unnecessary features or excessive precision, for instance by reducing ADC quantization or implementing simplified activation functions. On the other hand, given a target application, the neural network can be customized to adopt the features that are suitable for IMC acceleration, such as low-level quantization or hardware-aware training procedures. Finally, an electronic design automation (EDA) toolchain is needed in order to bridge the gap between the end-user and the hardware system, ranging from application-specific, high-level-of-abstraction design tools [186], to dedicated compilers [187], [188], [189] performing low-level core optimization in real-world implementations, similarly to existing CPU- and GPU-based computing system.

VII. CONCLUSION

This work provides an overview of memory devices and circuit topologies for IMC-based acceleration of machine learning and deep learning. Among various applications, a particular focus is given to the IMC acceleration of DNN inference, for which various approaches are presented and discussed, considering circuit overheads and parasitic effects affecting the final accuracy. IMC is a potentially-disruptive paradigm shift, either in terms of architectural change or raw computing performances. Further research on memory device engineering and understanding as well as on the hardware-network synergy could eventually unleash the full potential of IMC.

REFERENCES

[1] A. Gholami, Z. Yao, S. Kim, M. W. Mahoney, and K. Keutzer, *AI and Memory Wall*, RiseLab Medium Post, Berkeley, CA, USA, 2021.

[2] J. Kaplan et al. "Scaling Laws for Neural Language Models." Jan. 2020. [Online]. Available: http://arxiv.org/abs/2001.08361

[3] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. Alemi. "Inception-V4, Inception-ResNet and the Impact of Residual Connections on Learning." 2016. [Online]. Available: https://arxiv.org/abs/1602.07261

[4] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. "BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding." 2018. [Online]. Available: https://arxiv.org/abs/1810.04805

[5] M. Shoeybi, M. Patwary, R. Puri, P. LeGresley, J. Casper, and B. Catanzaro. "Megatron-LM: Training Multi-Billion Parameter Language Models Using Model Parallelism." 2019. [Online]. Available: https://arxiv.org/abs/1909.08053

[6] T. B. Brown et al. "Language Models Are Few-Shot Learners." 2020. [Online]. Available: https://arxiv.org/abs/2005.14165

[7] M. Horowitz, "1.1 computing's energy problem (and what we can do about it)," in *IEEE Int. Solid-State Circuits Conf. Dig. Tech. Papers (ISSCC)*, 2014, pp. 10–14.

[8] T. M. Conte, E. Track, and E. DeBenedictis, "Rebooting computing: New strategies for technology scaling," *Computer*, vol. 48, no. 12, pp. 10–13, 2015.

[9] M. A. Zidan, J. P. Strachan, and W. D. Lu, "The future of electronics based on memristive systems," *Nat. Electron.*, vol. 1, no. 1, pp. 22–29, Jan. 2018. [Online]. Available: https://www.nature.com/articles/s41928-017-0006-8

[10] D. Ielmini and H.-S. P. Wong, "In-memory computing with resistive switching devices," *Nat. Electron.*, vol. 1, no. 6, pp. 333–343, Jun. 2018. [Online]. Available: https://www.nature.com/articles/s41928-018-0092-2

[11] J. Borghetti et al., "A hybrid nanomemristor/transistor logic circuit capable of self-programming," *Proc. Nat. Acad. Sci. USA*, vol. 106, no. 6, pp. 1699–1703, Feb. 2009. [Online]. Available: https://www.pnas.org/doi/abs/10.1073/pnas.0806642106

[12] M. Cassinerio, N. Ciocchini, and D. Ielmini, "Logic computation in phase change materials by threshold and memory switching," *Adv. Mater.*, vol. 25, no. 41, pp. 5975–5980, 2013. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/adma.201301940

[13] C. D. Wright, P. Hosseini, and J. A. V. Diosdado, "Beyond von-Neumann computing with nanoscale phase-change memory devices," *Adv. Funct. Mater.*, vol. 23, no. 18, pp. 2248–2254, 2013.

[14] T. Tuma, A. Pantazi, M. Le Gallo, A. Sebastian, and E. Eleftheriou, "Stochastic phase-change neurons," *Nat. Nanotechnol.*, vol. 11, no. 8, pp. 693–699, Aug. 2016. [Online]. Available: https://www.nature.com/articles/nnano.2016.70

[15] L. Zheng, S. Shin, S. Lloyd, M. Gokhale, K. Kim, and S.-M. Kang, "RRAM-based TCAMs for pattern search," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, May 2016, pp. 1382–1385.

[16] C. Li et al., "Analog content-addressable memories with memristors," *Nat. Commun.*, vol. 11, no. 1, p. 1638, Apr. 2020. [Online]. Available: https://www.nature.com/articles/s41467-020-15254-4

[17] S. Gaba, P. Knag, Z. Zhang, and W. Lu, "Memristive devices for stochastic computing," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, Jun. 2014, pp. 2592–2595.

[18] S. N. Truong and K.-S. Min, "New memristor-based crossbar array architecture with 50-% area reduction and 48-% power saving for matrix-vector multiplication of analog neuromorphic computing," *J. Semicond. Technol. Sci.*, vol. 14, no. 3, pp. 356–363, 2014. [Online]. Available: https://koreascience.kr/article/JAKO201420249945718.page

[19] C. Li et al., "Analogue signal and image processing with large memristor crossbars," *Nat. Electron.*, vol. 1, no. 1, pp. 52–59, 2018.

[20] D. Keitel-Schulz and N. Wehn, "Embedded DRAM development: Technology, physical design, and application issues," *IEEE Des. Test Comput.*, vol. 18, no. 3, pp. 7–15, May/Jun. 2001.

[21] B. Yan et al., "A 1.041-Mb/mm^2 27.38-TOPS/W signed-INT8 dynamic-logic-based ADC-less SRAM compute-in-memory macro in 28nm with reconfigurable bitwise operation for AI and embedded applications," in *Proc. IEEE Int. Solid- State Circuits Conf. (ISSCC)*, vol. 65, Feb. 2022, pp. 188–190.

[22] Y.-D. Chih et al., "16.4 an 89TOPS/W and 16.3TOPS/mm^2 all-digital SRAM-based full-precision compute-in memory macro in 22nm for machine-learning edge applications," in *Proc. IEEE Int. Solid- State Circuits Conf. (ISSCC)*, vol. 64, Feb. 2021, pp. 252–254.

[23] A. Agrawal et al., "Xcel-RAM: Accelerating binary neural networks in high-throughput SRAM compute arrays," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 66, no. 8, pp. 3064–3076, Aug. 2019.

[24] R. Waser and M. Aono, "Nanoionics-based resistive switching memories," in *Nanoscience and Technology*. London, U.K.: Macmillan, Aug. 2009, pp. 158–165. [Online]. Available: http://www.worldscientific.com/doi/abs/10.1142/9789814287005_0016

[25] D. Ielmini, "Resistive switching memories based on metal oxides: Mechanisms, reliability and scaling," *Semicond. Sci. Technol.*, vol. 31, no. 6, Jun. 2016, Art. no. 063002. [Online]. Available: https://iopscience.iop.org/article/10.1088/0268-1242/31/6/063002

[26] H.-S. P. Wong et al., "Metal–oxide RRAM," *Proc. IEEE*, vol. 100, no. 6, pp. 1951–1970, Jun. 2012. [Online]. Available: http://ieeexplore.ieee.org/document/6193402/

[27] S. Balatti, S. Larentis, D. C. Gilmer, and D. Ielmini, "Multiple memory states in resistive switching devices through controlled size and orientation of the conductive filament," *Adv. Mater.*, vol. 25, no. 10, pp. 1474–1478, Mar. 2013. [Online]. Available: https://onlinelibrary.wiley.com/doi/10.1002/adma.201204097

[28] S. Yu, Y. Wu, and H.-S. P. Wong, "Investigating the switching dynamics and multilevel capability of bipolar metal oxide resistive switching memory," *Appl. Phys. Lett.*, vol. 98, no. 10, 2011, Art. no. 103514. [Online]. Available: https://doi.org/10.1063/1.3564883

[29] C.-C. Chou et al., "A 22nm 96KX144 RRAM macro with a self-tracking reference and a low ripple charge pump to achieve a configurable read window and a wide operating voltage range," in *Proc. IEEE Symp. VLSI Circuits*, Jun. 2020, pp. 1–2.

[30] H. Chung et al., "A 58nm 1.8V 1Gb PRAM with 6.4MB/s program BW," in *Proc. IEEE Int. Solid-State Circuits Conf.*, Feb. 2011, pp. 500–502.

[31] Y. Choi et al., "A 20nm 1.8V 8Gb PRAM with 40MB/s program bandwidth," in *Proc. IEEE Int. Solid-State Circuits Conf.*, Feb. 2012, pp. 46–48.

[32] T.-Y. Liu et al., "A 130.7 mm^2 2-layer 32Gb ReRAM memory device in 24nm technology," in *IEEE Int. Solid-State Circuits Conf. Dig. Tech. Papers*, Feb. 2013, pp. 210–211.

[33] M.-F. Chang et al., "19.4 embedded 1Mb ReRAM in 28nm CMOS with 0.27-to-1V read using swing-sample-and-couple sense amplifier and self-boost-write-termination scheme," in *IEEE Int. Solid-State Circuits Conf. Dig. Tech. Papers (ISSCC)*, Feb. 2014, pp. 332–333.

[34] J. Zahurak et al., "Process integration of a 27nm, 16Gb cu ReRAM," in *Proc. IEEE Int. Electron Devices Meeting*, Dec. 2014, pp. 6.2.1–6.2.4.

[35] S. Dünkel et al., "A FeFET based super-low-power ultra-fast embedded NVM technology for 22nm FDSOI and beyond," in *Proc. IEEE Int. Electron Devices Meeting (IEDM)*, Dec. 2017, pp. 19.7.1–19.7.4.

[36] Y. J. Song et al., "Demonstration of highly manufacturable STT-MRAM embedded in 28nm logic," in *Proc. IEEE Int. Electron Devices Meeting (IEDM)*, Dec. 2018, pp. 18.2.1–18.2.4.

[37] C.-C. Chou et al., "An N40 256K×44 embedded RRAM macro with SL-precharge SA and low-voltage current limiter to improve read and write performance," in *Proc. IEEE Int. Solid-State Circuits Conf. (ISSCC)*, Feb. 2018, pp. 478–480.

[38] T. Kim et al., "High-performance, cost-effective 2z nm two-deck cross-point memory integrated by self-align scheme for 128 Gb SCM," in *Proc. IEEE Int. Electron Devices Meeting (IEDM)*, Dec. 2018, pp. 37.1.1–37.1.4.

[39] F. Arnaud et al., "Truly innovative 28nm FDSOI technology for automotive micro-controller applications embedding 16MB phase change memory," in *Proc. IEEE Int. Electron Devices Meeting (IEDM)*, Dec. 2018, pp. 18.4.1–18.4.4.

[40] Y.-C. Shih et al., "Logic process compatible 40-nm 16-Mb, embedded perpendicular-MRAM with hybrid-resistance reference, sub-&micro a sensing resolution, and 17.5-nS read access time," *IEEE J. Solid-State Circuits*, vol. 54, no. 4, pp. 1029–1038, Apr. 2019.

[41] L. Wei et al., "13.3 a 7Mb STT-MRAM in 22FFL FinFET technology with 4ns read sensing time at 0.9V using write-verify-write scheme and offset-cancellation sensing technique," in *Proc. IEEE Int. Solid-State Circuits Conf. (ISSCC)*, Feb. 2019, pp. 214–216.

[42] P. Jain et al., "13.2 a 3.6Mb 10.1Mb/mm^2 embedded non-volatile ReRAM macro in 22nm FinFET technology with adaptive forming/set/reset schemes yielding down to 0.5V with sensing time of 5ns at 0.7V," in *Proc. IEEE Int. Solid-State Circuits Conf. (ISSCC)*, Feb. 2019, pp. 212–214.

[43] Y.-D. Chih et al., "13.3 a 22nm 32Mb embedded STT-MRAM with 10ns read speed, 1M cycle write endurance, 10 years retention at $150^{\circ}C$ and high immunity to magnetic field interference," in *Proc. IEEE Int. Solid-State Circuits Conf. (ISSCC)*, Feb. 2020, pp. 222–224.

[44] Y.-C. Shih et al., "A reflow-capable, embedded 8Mb STT-MRAM macro with 9nS read access time in 16nm FinFET logic CMOS process," in *Proc. IEEE Int. Electron Devices Meeting (IEDM)*, Dec. 2020, pp. 11.4.1–11.4.4.

[45] D. Edelstein et al., "A 14 nm embedded STT-MRAM CMOS technology," in *Proc. IEEE Int. Electron Devices Meeting (IEDM)*, Dec. 2020, pp. 11.5.1–11.5.4.

[46] V. B. Naik et al., "JEDEC-qualified highly reliable 22nm FD-SOI embedded MRAM for low-power industrial-grade, and extended performance towards automotive-grade-1 applications," in *Proc. IEEE Int. Electron Devices Meeting (IEDM)*, Dec. 2020, pp. 11.3.1–11.3.4.

[47] A. Fazio, "Advanced technology and systems of cross point memory," in *Proc. IEEE Int. Electron Devices Meeting (IEDM)*, Dec. 2020, pp. 24.1.1–24.1.4.

[48] J. J. Sun et al., "Commercialization of 1Gb Standalone spin-transfer torque MRAM," in *Proc. IEEE Int. Memory Workshop (IMW)*, May 2021, pp. 1–4.

[49] T. Shimoi et al., "A 22nm 32Mb embedded STT-MRAM macro achieving 5.9ns random read access and 5.8MB/s write throughput at up to Tj of 150°C," in *Proc. IEEE Symp. VLSI Technol. Circuits (VLSI Technol. Circuits)*, Jun. 2022, pp. 134–135.

[50] S. M. Seo et al., "First demonstration of full integration and characterization of 4F^2 1S1M cells with 45 nm of pitch and 20 nm of MTJ size," in *Proc. Int. Electron Devices Meeting (IEDM)*, Dec. 2022, pp. 10.1.1–10.1.4.

[51] G. Servalli, "A 45nm generation phase change memory technology," in *Proc. IEEE Int. Electron Devices Meeting (IEDM)*, Dec. 2009, pp. 1–4.

[52] C. Gopalan et al., "Demonstration of conductive bridging random access memory (CBRAM) in logic CMOS process," in *Proc. IEEE Int. Memory Workshop*, May 2010, pp. 1–4.

[53] S. H. Lee et al., "Highly productive PCRAM technology platform and full chip operation: Based on 4F^2 (84nm pitch) cell scheme for 1 Gb and beyond," in *Proc. Int. Electron Devices Meeting*, Dec. 2011, pp. 3.3.1–3.3.4.

[54] A. Kawahara et al., "Filament scaling forming technique and level-verify-write scheme with endurance over 107 cycles in ReRAM," in *IEEE Int. Solid-State Circuits Conf. Dig. Tech. Papers*, Feb. 2013, pp. 220–221.

[55] M. Ueki et al., "Low-power embedded ReRAM technology for IoT applications," in *Proc. Symp. VLSI Technol. (VLSI Technol.)*, Jun. 2015, pp. T108–T109.

[56] C. Park et al., "Systematic optimization of 1 Gbit perpendicular magnetic tunnel junction arrays for 28 nm embedded STT-MRAM and beyond," in *Proc. IEEE Int. Electron Devices Meeting (IEDM)*, Dec. 2015, pp. 26.2.1–26.2.4.

[57] S.-W. Chung et al., "4Gbit density STT-MRAM using perpendicular MTJ realized with compact cell structure," in *Proc. IEEE Int. Electron Devices Meeting (IEDM)*, Dec. 2016, pp. 27.1.1–27.1.4.

[58] Y. J. Song et al., "Highly functional and reliable 8Mb STT-MRAM embedded in 28nm logic," in *Proc. IEEE Int. Electron Devices Meeting (IEDM)*, Dec. 2016, pp. 27.2.1–27.2.4.

[59] D. Shum et al., "CMOS-embedded STT-MRAM arrays in 2x nm nodes for GP-MCU applications," in *Proc. Symp. VLSI Technol.*, Jun. 2017, pp. T208–T209.

[60] J. Y. Wu et al., "A 40nm low-power logic compatible phase change memory technology," in *Proc. IEEE Int. Electron Devices Meeting (IEDM)*, Dec. 2018, pp. 27.6.1–27.6.4.

[61] F. Arnaud et al., "High density embedded PCM cell in 28nm FDSOI technology for automotive micro-controller applications," in *Proc. IEEE Int. Electron Devices Meeting (IEDM)*, Dec. 2020, pp. 24.2.1–24.2.4.

[62] C.-F. Yang et al., "Industrially applicable read disturb model and performance on mega-bit 28nm embedded RRAM," in *Proc. IEEE Symp. VLSI Technol.*, Jun. 2020, pp. 1–2.

[63] S. H. Han et al., "28-nm 0.08 mm^2/Mb embedded MRAM for frame buffer memory," in *Proc. IEEE Int. Electron Devices Meeting (IEDM)*, Dec. 2020, pp. 11.2.1–11.2.4.

[64] D. Min et al., "18nm FDSOI technology platform embedding PCM & innovative continuous-active construct enhancing performance for leading-edge MCU applications," in *Proc. IEEE Int. Electron Devices Meeting (IEDM)*, Dec. 2021, pp. 13.1.1–13.1.4.

[65] K. Lee et al., "28nm CIS-compatible embedded STT-MRAM for frame buffer memory," in *Proc. IEEE Int. Electron Devices Meeting (IEDM)*, Dec. 2021, pp. 2.1.1–2.1.4.

[66] T. Ito et al., "A 20Mb embedded STT-MRAM array achieving 72% write energy reduction with self-termination write schemes in 16nm FinFET logic process," in *Proc. IEEE Int. Electron Devices Meeting (IEDM)*, Dec. 2021, pp. 2.2.1–2.2.4.

[67] C. Peters, F. Adler, K. Hofmann, and J. Otterstedt, "Reliability of 28nm embedded RRAM for consumer and industrial products," in *Proc. IEEE Int. Memory Workshop (IMW)*, May 2022, pp. 1–3.

[68] S. Raoux, W. Wełnic, and D. Ielmini, "Phase change materials and their application to nonvolatile memories," *Chem. Rev.*, vol. 110, no. 1, pp. 240–267, Jan. 2010. [Online]. Available: https://pubs.acs.org/doi/10.1021/cr900040x

[69] M. Wuttig and N. Yamada, "Phase-change materials for rewriteable data storage," *Nat. Mater.*, vol. 6, no. 11, pp. 824–832, Nov. 2007. [Online]. Available: https://www.nature.com/articles/nmat2009

[70] P. Zuliani et al., "Overcoming temperature limitations in phase change memories with optimized Ge$_x$Sb$_y$Te$_z$," *IEEE Trans. Electron Devices*, vol. 60, no. 12, pp. 4020–4026, Dec. 2013.

[71] D. Ielmini, A. Lacaita, A. Pirovano, F. Pellizzer, and R. Bez, "Analysis of phase distribution in phase-change nonvolatile memories," *IEEE Electron Device Lett.*, vol. 25, no. 7, pp. 507–509, Jul. 2004. [Online]. Available: http://ieeexplore.ieee.org/document/1308435/

[72] P. Narayanan et al., "Fully on-chip MAC at 14nm enabled by accurate row-wise programming of PCM-based weights and parallel vector-transport in duration-format," in *Proc. Symp. VLSI Technol.*, Jun. 2021, pp. 1–2.

[73] T. Mikolajick et al., "FeRAM technology for high density applications," *Microelectron. Rel.*, vol. 41, no. 7, pp. 947–950, Jul. 2001. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S002627140100049X

[74] T. S. Böscke, J. Müller, D. Bräuhaus, U. Schröder, and U. Böttger, "Ferroelectricity in hafnium oxide thin films," *Appl. Phys. Lett.*, vol. 99, no. 10, Sep. 2011, Art. no. 102903. [Online]. Available: http://aip.scitation.org/doi/10.1063/1.3634052

[75] S. Majumdar, "Back' end CMOS compatible and flexible ferroelectric memories for neuromorphic computing and adaptive sensing," *Adv. Intell. Syst.*, vol. 4, no. 4, Apr. 2022, Art. no. 2100175. [Online]. Available: https://onlinelibrary.wiley.com/doi/10.1002/aisy.202100175

[76] C. Chappert, A. Fert, and F. N. Van Dau, "The emergence of spin electronics in data storage," *Nat. Mater.*, vol. 6, no. 11, pp. 813–823, Nov. 2007. [Online]. Available: https://www.nature.com/articles/nmat2024

[77] R. Carboni et al., "Modeling of breakdown-limited endurance in spin-transfer torque magnetic memory under pulsed cycling regime," *IEEE Trans. Electron Devices*, vol. 65, no. 6, pp. 2470–2478, Jun. 2018. [Online]. Available: https://ieeexplore.ieee.org/document/8338113/

[78] A. I. Khan, A. Keshavarzi, and S. Datta, "The future of ferroelectric field-effect transistor technology," *Nat. Electron.*, vol. 3, no. 10, pp. 588–597, Oct. 2020. [Online]. Available: https://www.nature.com/articles/s41928-020-00492-7

[79] M. Trentzsch et al., "A 28nm HKMG super low power embedded NVM technology based on ferroelectric FETs," in *Proc. IEEE Int. Electron Devices Meeting (IEDM)*, Dec. 2016, pp. 11.5.1–11.5.4.

[80] I. M. Miron et al., "Perpendicular switching of a single ferromagnetic layer induced by in-plane current injection," *Nature*, vol. 476, no. 7359, pp. 189–193, Aug. 2011. [Online]. Available: http://www.nature.com/articles/nature10309

[81] K. Garello et al., "Ultrafast magnetization switching by spin-orbit torques," *Appl. Phys. Lett.*, vol. 105, no. 21, Nov. 2014, Art. no. 212402. [Online]. Available: https://aip.scitation.org/doi/full/10.1063/1.4902443

[82] J. Tang et al., "ECRAM as scalable synaptic cell for high-speed, low-power neuromorphic computing," in *Proc. IEEE Int. Electron Devices Meeting (IEDM)*, Dec. 2018, pp. 13.1.1–13.1.4. [Online]. Available: https://ieeexplore.ieee.org/document/8614551/

[83] J. Lee, R. D. Nikam, D. Kim, and H. Hwang, "Highly scalable (30 nm) and ultra-low-energy (∼5fJ/pulse) vertical sensing ECRAM with ideal synaptic characteristics using ion-permeable Graphene electrodes," in *Proc. Int. Electron Devices Meeting (IEDM)*, Dec. 2022, pp. 2.2.1–2.2.4. [Online]. Available: https://ieeexplore.ieee.org/document/10019326/

[84] S. Kim et al., "Metal-oxide based, CMOS-compatible ECRAM for deep learning accelerator," in *Proc. IEEE Int. Electron Devices Meeting (IEDM)*, Dec. 2019, pp. 35.7.1–35.7.4. [Online]. Available: https://ieeexplore.ieee.org/document/8993463/

[85] M. Farronato, M. Melegari, S. Ricci, S. Hashemkhani, A. Bricalli, and D. Ielmini, "Memtransistor devices based on MoS$_2$ multilayers with volatile switching due to AG cation migration," *Adv. Electron. Mater.*, vol. 8, no. 8, Jan. 2022, Art. no. 2101161. [Online]. Available: https://onlinelibrary.wiley.com/doi/10.1002/aelm.202101161

[86] H. Lee et al., "Dual-gated MoS$_2$ memtransistor crossbar array," *Adv. Funct. Mater.*, vol. 30, no. 45, Nov. 2020, Art. no. 2003683. [Online]. Available: https://onlinelibrary.wiley.com/doi/10.1002/adfm.202003683

[87] R. A. John et al., "Ultralow power dual-gated subthreshold oxide neuristors: An enabler for higher order neuronal temporal correlations," *ACS Nano*, vol. 12, no. 11, pp. 11263–11273, Nov. 2018. [Online]. Available: https://pubs.acs.org/doi/10.1021/acsnano.8b05903

[88] V. K. Sangwan et al., "Multi-terminal memtransistors from polycrystalline monolayer molybdenum disulfide," *Nature*, vol. 554, no. 7693, pp. 500–504, Feb. 2018. [Online]. Available: https://www.nature.com/articles/nature25747

[89] M. Farronato, P. Mannocci, M. Melegari, S. Ricci, C. M. Compagnoni, and D. Ielmini, "Reservoir computing with charge-trap memory based on a MoS$_2$ channel for neuromorphic engineering," *Adv. Mater.*, Oct. 2022, Art. no. 2205381.

[90] M. Farronato, M. Melegari, S. Ricci, S. Hashemkani, C. M. Compagnoni, and D. Ielmini, "Low-current, highly linear synaptic memory device based on MoS^2 transistors for online training and inference," in *Proc. IEEE 4th Int. Conf. Artif. Intell. Circuits Syst. (AICAS)*, 2022, pp. 1–4.

[91] D. Ielmini and S. Ambrogio, "Emerging neuromorphic devices," *Nanotechnology*, vol. 31, no. 9, Feb. 2020, Art. no. 092001. [Online]. Available: https://iopscience.iop.org/article/10.1088/1361-6528/ab554b

[92] S. Shukla et al., "A scalable multi-TeraOPS core for AI training and inference," *IEEE Solid-State Circuits Lett.*, vol. 1, no. 12, pp. 217–220, Dec. 2018.

[93] A. Chen, "A comprehensive crossbar array model with solutions for line resistance and nonlinear device characteristics," *IEEE Trans. Electron Devices*, vol. 60, no. 4, pp. 1318–1326, Apr. 2013.

[94] W.-H. Chen et al., "A 65nm 1Mb nonvolatile computing-in-memory ReRAM macro with sub-16ns multiply-and-accumulate for binary DNN AI edge processors," in *Proc. IEEE Int. Solid-State Circuits Conf. (ISSCC)*, Feb. 2018, pp. 494–496.

[95] S. D. Spetalnick et al., "A 40nm 64kb 26.56TOPS/W 2.37Mb/mm^2 RRAM binary/compute-in-memory macro with 4.23x improvement in density and >75% use of sensing dynamic range," in *Proc. IEEE Int. Solid-State Circuits Conf. (ISSCC)*, vol. 65, Feb. 2022, pp. 1–3.

[96] T.-H. Kim, J. Lee, S. Kim, J. Park, B.-G. Park, and H. Kim, "3-bit multilevel operation with accurate programming scheme in TiO_x/Al_2O_3 memristor crossbar array for quantized neuromorphic system," *Nanotechnology*, vol. 32, no. 29, Apr. 2021, Art. no. 295201. doi: 10.1088/1361-6528/abf0cc.

[97] V. Milo et al., "Multilevel HfO_2-based RRAM devices for low-power neuromorphic networks," *APL Mater.*, vol. 7, no. 8, Aug. 2019, Art. no. 081120. [Online]. Available: https://aip.scitation.org/doi/full/10.1063/1.5108650

[98] I. Yeo, M. Chu, S.-G. Gi, H. Hwang, and B.-G. Lee, "Stuck-at-fault tolerant schemes for memristor crossbar array-based neural networks," *IEEE Trans. Electron Devices*, vol. 66, no. 7, pp. 2937–2945, Jul. 2019.

[99] D. Ielmini and G. Pedretti, "Device and circuit architectures for in-memory computing," *Adv. Intell. Syst.*, vol. 2, no. 7, 2020, Art. no. 2000040. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/aisy.202000040

[100] Y.-C. Chen et al., "An access-transistor-free (0T/1R) non-volatile resistance random access memory (RRAM) using a novel threshold switching, self-rectifying chalcogenide device," in *Proc. IEEE Int. Electron Devices Meeting*, Dec. 2003, pp. 37.4.1–37.4.4.

[101] D. Ielmini and Y. Zhang, "Physics-based analytical model of chalcogenide-based memories for array simulation," in *Proc. Int. Electron Devices Meeting*, Dec. 2006, pp. 1–4.

[102] M. Hu et al., "Memristor-based analog computation and neural network classification with a dot product engine," *Adv. Mater.*, vol. 30, no. 9, 2018, Art. no. 1705914. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/adma.201705914

[103] H. Cai et al., "Proposal of analog in-memory computing with magnified tunnel magnetoresistance ratio and universal STT-MRAM cell," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 69, no. 4, pp. 1519–1531, Apr. 2022.

[104] J. M. Lopez et al., "1S1R optimization for high-frequency inference on binarized spiking neural networks," *Adv. Electron. Mater.*, vol. 8, no. 8, 2022, Art. no. 2200323. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/aelm.202200323

[105] J. M. Lopez et al., "1S1R sub-threshold operation in crossbar arrays for low power BNN inference computing," in *Proc. IEEE Int. Memory Workshop (IMW)*, May 2022, pp. 1–4.

[106] G. W. Burr et al., "Access devices for 3D crosspoint memory," *J. Vacuum Sci. Technol. B*, vol. 32, no. 4, Jul. 2014, Art. no. 040802. [Online]. Available: https://avs.scitation.org/doi/full/10.1116/1.4889999

[107] S. A. Chekol, J. Song, J. Park, J. Yoo, S. Lim, and H. Hwang, "Chapter 5—Selector devices for emerging memories," in *Memristive Devices for Brain-Inspired Computing* (Woodhead Publishing Series in Electronic and Optical Materials), S. Spiga, A. Sebastian, D. Querlioz, and B. Rajendran, Eds. London, U.K.: Woodhead, Jan. 2020, pp. 135–164. [Online]. Available: https://www.sciencedirect.com/science/article/pii/B9780081027820000058

[108] Y.-C. Luo, A. Lu, J. Hur, S. Li, and S. Yu, "Design of non-volatile capacitive crossbar array for in-memory computing," in *Proc. IEEE Int. Memory Workshop (IMW)*, May 2021, pp. 1–4.

[109] 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. [Online]. Available: https://www.nature.com/articles/s41586-021-04196-6

[110] "Pillars of Creation (NIRCam and MIRI Composite Image)." Accessed: Mar. 7, 2023. [Online]. Available: https://webbtelescope.org/contents/media/images

[111] S. N. Truong, S. Shin, S.-D. Byeon, J. Song, and K.-S. Min, "New twin crossbar architecture of binary memristors for low-power image recognition with discrete cosine transform," *IEEE Trans. Nanotechnol.*, vol. 14, no. 6, pp. 1104–1111, Nov. 2015.

[112] Z. Sun, G. Pedretti, E. Ambrosi, A. Bricalli, W. Wang, and D. Ielmini, "Solving matrix equations in one step with cross-point resistive arrays," *Proc. Nat. Acad. Sci. USA*, vol. 116, no. 10, pp. 4123–4128, 2019.

[113] P. Mannocci, G. Pedretti, E. Giannone, E. Melacarne, Z. Sun, and D. Ielmini, "A universal, analog, in-memory computing primitive for linear algebra using memristors," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 68, no. 12, pp. 4889–4899, Dec. 2021.

[114] Z. Sun, E. Ambrosi, G. Pedretti, A. Bricalli, and D. Ielmini, "In-memory PageRank accelerator with a cross-point array of resistive memories," *IEEE Trans. Electron Devices*, vol. 67, no. 4, pp. 1466–1470, Apr. 2020.

[115] Z. Sun, G. Pedretti, A. Bricalli, and D. Ielmini, "One-step regression and classification with cross-point resistive memory arrays," *Sci. Adv.*, vol. 6, no. 5, 2020, Art. no. eaay2378.

[116] P. Mannocci, E. Melacarne, and D. Ielmini, "An analogue in-memory ridge regression circuit with application to massive MIMO acceleration," *IEEE J. Emerg. Sel. Topics Circuits Syst.*, vol. 12, no. 4, pp. 952–962, Dec. 2022.

[117] M. Mahmoodi et al., "An analog neuro-optimizer with adaptable annealing based on 64 × 64 0T1R crossbar circuit," in *Proc. IEEE Int. Electron Devices Meeting (IEDM)*, 2019, pp. 14–7.

[118] M. N. Bojnordi and E. Ipek, "Memristive Boltzmann machine: A hardware accelerator for combinatorial optimization and deep learning," in *Proc. IEEE Int. Symp. High Perform. Comput. Archit. (HPCA)*, 2016, pp. 1–13.

[119] Y. Kiat, Y. Vortman, and N. Sapir, "Feather moult and bird appearance are correlated with global warming over the last 200 years," *Nat. Commun.*, vol. 10, no. 1, p. 2540, 2019.

[120] J. J. Hopfield, "Neural networks and physical systems with emergent collective computational abilities," *Proc. Nat. Acad. Sci. USA*, vol. 79, no. 8, pp. 2554–2558, 1982.

[121] G. Pedretti et al., "A spiking recurrent neural network with phase-change memory neurons and synapses for the accelerated solution of constraint satisfaction problems," *IEEE J. Explor. Solid-State Computat. Devices Circuits*, vol. 6, no. 1, pp. 89–97, Jun. 2020.

[122] F. Cai et al., "Power-efficient combinatorial optimization using intrinsic noise in memristor Hopfield neural networks," *Nat. Electron.*, vol. 3, no. 7, pp. 409–418, 2020.

[123] T. Dalgaty, E. Esmanhotto, N. Castellani, D. Querlioz, and E. Vianello, "*Ex situ* transfer of Bayesian neural networks to resistive memory-based inference hardware," *Adv. Intell. Syst.*, vol. 3, no. 8, 2021, Art. no. 2000103.

[124] S. Agarwal et al., "Resistive memory device requirements for a neural algorithm accelerator," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, 2016, pp. 929–938.

[125] X. Xu et al., "40× retention improvement by eliminating resistance relaxation with high temperature forming in 28 nm RRAM chip," in *Proc. IEEE Int. Electron Devices Meeting (IEDM)*, 2018, pp. 20.1.1–20.1.4.

[126] M. Courbariaux, I. Hubara, D. Soudry, R. El-Yaniv, and Y. Bengio. "Binarized Neural Networks: Training Deep Neural Networks With Weights and Activations Constrained to $+1$ or -1." Mar. 2016. [Online]. Available: http://arxiv.org/abs/1602.02830

[127] T. Simons and D.-J. Lee, "A review of binarized neural networks," *Electronics*, vol. 8, no. 6, p. 661, Jun. 2019. [Online]. Available: https://www.mdpi.com/2079-9292/8/6/661

[128] M. Bocquet et al., "In-memory and error-immune differential RRAM implementation of binarized deep neural networks," in *Proc. IEEE Int. Electron Devices Meeting (IEDM)*, Dec. 2018, pp. 20.6.1–20.6.4.

[129] H. Kim, Y. Kim, and J.-J. Kim, "In-memory batch-normalization for resistive memory based binary neural network hardware," in *Proc. 24th Asia South Pac. Design Autom. Conf. (ASPDAC)*, 2019, pp. 645–650. [Online]. Available: https://doi.org/10.1145/3287624.3287718

[130] E. Giacomin, T. Greenberg-Toledo, S. Kvatinsky, and P.-E. Gaillardon, "A robust digital RRAM-based convolutional block for low-power image processing and learning applications," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 66, no. 2, pp. 643–654, Feb. 2019.

[131] S. Angizi, Z. He, A. Awad, and D. Fan, "MRIMA: An MRAM-based in-memory accelerator," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 39, no. 5, pp. 1123–1136, May 2020.

[132] Y. Long et al., “A ferroelectric FET-based processing-in-memory architecture for DNN acceleration,” *IEEE J. Explor. Solid-State Comput. Devices Circuits*, vol. 5, no. 2, pp. 113–122, Dec. 2019.

[133] H. Fujiwara et al., “A 5-nm 254-TOPS/W 221-TOPS/mm^2 fully-digital computing-in-memory macro supporting wide-range dynamic-voltage-frequency scaling and simultaneous MAC and write operations,” in *Proc. IEEE Int. Solid-State Circuits Conf. (ISSCC)*, vol. 65, Feb. 2022, pp. 1–3.

[134] F. Tu et al., “A 28nm 29.2TFLOPS/W BF16 and 36.5TOPS/W INT8 reconfigurable digital CIM processor with unified FP/INT pipeline and bitwise in-memory booth multiplication for cloud deep learning acceleration,” in *Proc. IEEE Int. Solid-State Circuits Conf. (ISSCC)*, vol. 65, Feb. 2022, pp. 1–3.

[135] H. Kim, Q. Chen, T. Yoo, T. T.-H. Kim, and B. Kim, “A 1-16b precision reconfigurable digital in-memory computing macro featuring column-MAC architecture and bit-serial computation,” in *Proc. IEEE 45th Eur. Solid-State Circuits Conf. (ESSCIRC)*, Sep. 2019, pp. 345–348.

[136] C.-F. Lee et al., “A 12nm 121-TOPS/W 41.6-TOPS/mm^2 all digital full precision SRAM-based compute-in-memory with configurable bit-width for AI edge applications,” in *Proc. IEEE Symp. VLSI Technol. Circuits (VLSI Technol. Circuits)*, Jun. 2022, pp. 24–25.

[137] H. Oh, H. Kim, N. Kang, Y. Kim, J. Park, and J.-J. Kim, “Single RRAM cell-based in-memory accelerator architecture for binary neural networks,” in *Proc. IEEE 3rd Int. Conf. Artif. Intell. Circuits Syst. (AICAS)*, Jun. 2021, pp. 1–4.

[138] X. Sun, S. Yin, X. Peng, R. Liu, J.-S. Seo, and S. Yu, “XNOR-RRAM: A scalable and parallel resistive synaptic architecture for binary neural networks,” in *Proc. Design Autom. Test Europe Conf. Exhibit. (DATE)*, Mar. 2018, pp. 1423–1428.

[139] S. Yin, X. Sun, S. Yu, and J.-S. Seo, “High-throughput in-memory computing for binary deep neural networks with monolithically integrated RRAM and 90-nm CMOS,” *IEEE Trans. Electron Devices*, vol. 67, no. 10, pp. 4185–4192, Oct. 2020.

[140] Y.-F. Qin, R. Kuang, X.-D. Huang, Y. Li, J. Chen, and X.-S. Miao, “Design of high robustness BNN inference accelerator based on binary memristors,” *IEEE Trans. Electron Devices*, vol. 67, no. 8, pp. 3435–3441, Aug. 2020.

[141] A. P. Chowdhury, P. Kulkarni, and M. N. Bojnordi, “MB-CNN: Memristive binary convolutional neural networks for embedded mobile devices,” *J. Low Power Electron. Appl.*, vol. 8, no. 4, p. 38, Dec. 2018. [Online]. Available: https://www.mdpi.com/2079-9268/8/4/38

[142] D. Saito et al., “Analog in-memory computing in FeFET-based 1T1R array for edge AI applications,” in *Proc. Symp. VLSI Circuits*, Jun. 2021, pp. 1–2.

[143] C. Matsui, K. Toprasertpong, S. Takagi, and K. Takeuchi, “Energy-efficient reliable HZO FeFET computation-in-memory with local multiply & global accumulate array for source-follower & charge-sharing voltage sensing,” in *Proc. Symp. VLSI Circuits*, Jun. 2021, pp. 1–2.

[144] J.-W. Su et al., “16.3 a 28nm 384kb 6T-SRAM computation-in-memory macro with 8b precision for AI edge chips,” in *Proc. IEEE Int. Solid-State Circuits Conf. (ISSCC)*, vol. 64, Feb. 2021, pp. 250–252.

[145] H. Jia, H. Valavi, Y. Tang, J. Zhang, and N. Verma, “A programmable heterogeneous microprocessor based on bit-scalable in-memory computing,” *IEEE J. Solid-State Circuits*, vol. 55, no. 9, pp. 2609–2621, Sep. 2020.

[146] Q. Dong et al., “15.3 a 351TOPS/W and 372.4GOPS compute-in-memory SRAM macro in 7nm FinFET CMOS for machine-learning applications,” in *Proc. IEEE Int. Solid-State Circuits Conf. (ISSCC)*, Feb. 2020, pp. 242–244.

[147] S. Yin, Z. Jiang, J.-S. Seo, and M. Seok, “XNOR-SRAM: In-memory computing SRAM macro for binary/ternary deep neural networks,” *IEEE J. Solid-State Circuits*, vol. 55, no. 6, pp. 1733–1743, Jun. 2020. [Online]. Available: https://ieeexplore.ieee.org/document/8959407/

[148] P.-F. Chiu, W. H. Choi, W. Ma, M. Qin, and M. Lueker-Boden, “A binarized neural network accelerator with differential crosspoint memristor array for energy-efficient MAC operations,” in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, May 2019, pp. 1–5.

[149] Z. Jiang, S. Yin, J.-S. Seo, and M. Seok, “C3SRAM: In-memory-computing SRAM macro based on capacitive-coupling computing,” *IEEE Solid-State Circuits Lett.*, vol. 2, no. 9, pp. 131–134, Sep. 2019.

[150] H. Wang et al., “A 32.2 TOPS/W SRAM compute-in-memory macro employing a linear 8-bit C-2C ladder for charge domain computation in 22nm for edge inference,” in *Proc. IEEE Symp. VLSI Technol. Circuits (VLSI Technol. Circuits)*, Jun. 2022, pp. 36–37.

[151] H. Jia et al., “15.1 a programmable neural-network inference accelerator based on scalable in-memory computing,” in *Proc. IEEE Int. Solid-State Circuits Conf. (ISSCC)*, vol. 64, Feb. 2021, pp. 236–238.

[152] H. Valavi, P. J. Ramadge, E. Nestler, and N. Verma, “A mixed-signal binarized convolutional-neural-network accelerator integrating dense weight storage and multiplication for reduced data movement,” in *Proc. IEEE Symp. VLSI Circuits*, Jun. 2018, pp. 141–142.

[153] D. Bankman, L. Yang, B. Moons, M. Verhelst, and B. Murmann, “An always-on 3.8 &micro J/86% CIFAR-10 mixed-signal binary CNN processor with all memory on chip in 28-nm CMOS,” *IEEE J. Solid-State Circuits*, vol. 54, no. 1, pp. 158–172, Jan. 2019.

[154] M. E. Sinangil et al., “A 7-nm compute-in-memory SRAM macro supporting multi-bit input, weight and output and achieving 351 TOPS/W and 372.4 GOPS,” *IEEE J. Solid-State Circuits*, vol. 56, no. 1, pp. 188–198, Jan. 2021.

[155] 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. [Online]. Available: https://www.nature.com/articles/s41586-022-04992-8

[156] H. Jiang, W. Li, S. Huang, and S. Yu, “A 40nm analog-input ADC-free compute-in-memory RRAM macro with pulse-width modulation between sub-arrays,” in *Proc. IEEE Symp. VLSI Technol. Circuits (VLSI Technol. Circuits)*, Jun. 2022, pp. 266–267.

[157] C.-X. Xue et al., “15.4 a 22nm 2Mb ReRAM compute-in-memory macro with 121-28TOPS/W for Multibit MAC computing for tiny AI edge devices,” in *Proc. IEEE Int. Solid-State Circuits Conf. (ISSCC)*, Feb. 2020, pp. 244–246.

[158] Q. Liu et al., “33.2 a fully integrated analog ReRAM based 78.4TOPS/W compute-in-memory chip with fully parallel MAC computing,” in *Proc. IEEE Int. Solid-State Circuits Conf. (ISSCC)*, Feb. 2020, pp. 500–502.

[159] R. Khaddam-Aljameh et al., “HERMES-core—A 1.59-TOPS/mm^2 PCM on 14-nm CMOS in-memory compute core using 300-ps/LSB linearized CCO-based ADCs,” *IEEE J. Solid-State Circuits*, vol. 57, no. 4, pp. 1027–1038, Apr. 2022.

[160] V. Joshi et al., “Accurate deep neural network inference using computational phase-change memory,” *Nat. Commun.*, vol. 11, no. 1, p. 2473, May 2020. [Online]. Available: https://www.nature.com/articles/s41467-020-16108-9

[161] W.-S. Khwa et al., “A 40-nm, 2M-cell, 8b-precision, hybrid SLC-MLC PCM computing-in-memory macro with 20.5–65.0TOPS/W for tiny-Al edge devices,” in *Proc. IEEE Int. Solid-State Circuits Conf. (ISSCC)*, vol. 65, Feb. 2022, pp. 1–3.

[162] P. Deaville, B. Zhang, and N. Verma, “A 22nm 128-kb MRAM row/column-parallel in-memory computing macro with memory-resistance boosting and multi-column ADC readout,” in *Proc. IEEE Symp. VLSI Technol. Circuits (VLSI Technol. Circuits)*, Jun. 2022, pp. 268–269.

[163] T. Soliman et al., “Ultra-low power flexible precision FeFET based analog in-memory computing,” in *Proc. IEEE Int. Electron Devices Meeting (IEDM)*, Dec. 2020, pp. 29.2.1–29.2.4.

[164] C.-X. Xue et al., “16.1 a 22nm 4Mb 8b-precision ReRAM computing-in-memory macro with 11.91 to 195.7TOPS/W for tiny AI edge devices,” in *Proc. IEEE Int. Solid-State Circuits Conf. (ISSCC)*, vol. 64, Feb. 2021, pp. 245–247.

[165] J.-M. Hung et al., “An 8-Mb DC-current-free binary-to-8b precision ReRAM nonvolatile computing-in-memory macro using time-space-readout with 1286.4-21.6TOPS/W for edge-AI devices,” in *Proc. IEEE Int. Solid-State Circuits Conf. (ISSCC)*, vol. 65, Feb. 2022, pp. 1–3.

[166] A. Glukhov et al., “Statistical model of program/verify algorithms in resistive-switching memories for in-memory neural network accelerators,” in *Proc. IEEE Int. Rel. Phys. Symp. (IRPS)*, Mar. 2022, pp. 1–7.

[167] S. Ambrogio et al., "Reducing the impact of phase-change memory conductance drift on the inference of large-scale hardware neural networks," in *Proc. IEEE Int. Electron Devices Meeting (IEDM)*, Dec. 2019, pp. 6.1.1–6.1.4.

[168] S. Ambrogio, S. Balatti, V. McCaffrey, D. C. Wang, and D. Ielmini, "Noise-induced resistance broadening in resistive switching memory—Part I: Intrinsic cell behavior," *IEEE Trans. Electron Devices*, vol. 62, no. 11, pp. 3805–3811, Nov. 2015.

[169] N. Lepri, M. Baldo, P. Mannocci, A. Glukhov, V. Milo, and D. Ielmini, "Modeling and compensation of IR drop in crosspoint accelerators of neural networks," *IEEE Trans. Electron Devices*, vol. 69, no. 3, pp. 1575–1581, Mar. 2022.

[170] N. Lepri, A. Glukhov, and D. Ielmini, "Mitigating read-program variation and IR drop by circuit architecture in RRAM-based neural network accelerators," in *Proc. IEEE Int. Rel. Phys. Symp. (IRPS)*, Mar. 2022, pp. 1–6.

[171] F. L. Aguirre, N. M. Gomez, S. M. Pazos, F. Palumbo, J. Suñé, and E. Miranda, "Minimization of the line resistance impact on memdiode-based simulations of multilayer perceptron arrays applied to pattern recognition," *J. Low Power Electron. Appl.*, vol. 11, no. 1, p. 9, Mar. 2021. [Online]. Available: https://www.mdpi.com/2079-9268/11/1/9

[172] C. Mackin et al., "Optimised weight programming for analogue memory-based deep neural networks," *Nat. Commun.*, vol. 13, no. 1, p. 3765, Jun. 2022. [Online]. Available: https://www.nature.com/articles/s41467-022-31405-1

[173] F. Zhang and M. Hu, "Mitigate parasitic resistance in resistive crossbar-based convolutional neural networks," *ACM J. Emerg. Technol. Comput. Syst.*, vol. 16, no. 3, pp. 1–25, 2020. [Online]. Available: https://doi.org/10.1145/3371277

[174] D. Joksas et al., "Nonideality-aware training for accurate and robust low-power memristive neural networks," *Adv. Sci.*, vol. 9, no. 17, 2022, Art. no. 2105784. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/advs.202105784

[175] V. Milo et al., "Accurate program/verify schemes of resistive switching memory (RRAM) for in-memory neural network circuits," *IEEE Trans. Electron Devices*, vol. 68, no. 8, pp. 3832–3837, Aug. 2021.

[176] A. Athmanathan, M. Stanisavljevic, N. Papandreou, H. Pozidis, and E. Eleftheriou, "Multilevel-cell phase-change memory: A viable technology," *IEEE J. Emerg. Sel. Topics Circuits Syst.*, vol. 6, no. 1, pp. 87–100, Mar. 2016.

[177] S. Ambrogio, S. Balatti, A. Cubeta, A. Calderoni, N. Ramaswamy, and D. Ielmini, "Statistical fluctuations in HfOx resistive-switching memory: Part I—Set/reset variability," *IEEE Trans. Electron Devices*, vol. 61, no. 8, pp. 2912–2919, Aug. 2014.

[178] E. Pérez et al., "Analysis of the statistics of device-to-device and cycle-to-cycle variability in TiN/Ti/Al:HfO^2/TiN RRAMs," *Microelectron. Eng.*, vol. 214, pp. 104–109, Jun. 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167931719301303

[179] D. Ielmini, S. Lavizzari, D. Sharma, and A. L. Lacaita, "Physical interpretation, modeling and impact on phase change memory (PCM) reliability of resistance drift due to chalcogenide structural relaxation," in *Proc. IEEE Int. Electron Devices Meeting*, 2007, pp. 939–942.

[180] N. Ciocchini, E. Palumbo, M. Borghi, P. Zuliani, R. Annunziata, and D. Ielmini, "Modeling resistance instabilities of set and reset states in phase change memory with ge-rich GeSbTe," *IEEE Trans. Electron Devices*, vol. 61, no. 6, pp. 2136–2144, Jun. 2014.

[181] Y.-H. Lin et al., "Performance impacts of analog ReRAM non-ideality on neuromorphic computing," *IEEE Trans. Electron Devices*, vol. 66, no. 3, pp. 1289–1295, Mar. 2019.

[182] I. Muñoz-Martín, S. Bianchi, O. Melnic, A. G. Bonfanti, and D. Ielmini, "A drift-resilient hardware implementation of neural accelerators based on phase change memory devices," *IEEE Trans. Electron Devices*, vol. 68, no. 12, pp. 6076–6081, Dec. 2021.

[183] M. Bertuletti, I. Munoz-Martín, S. Bianchi, A. G. Bonfanti, and D. Ielmini, "A multilayer neural accelerator with binary activations based on phase-change memory," *IEEE Trans. Electron Devices*, vol. 70, no. 3, pp. 986–992, Mar. 2023.

[184] C.-C. Chang et al., "NV-BNN: An accurate deep convolutional neural network based on binary STT-MRAM for adaptive AI edge," in *Proc. 56th Annu. Design Autom. Conf. (DAC)*, 2019, pp. 1–6. [Online]. Available: https://doi.org/10.1145/3316781.3317872

[185] M. Le Gallo, A. Sebastian, G. Cherubini, H. Giefers, and E. Eleftheriou, "Compressed sensing with approximate message passing using in-memory computing," *IEEE Trans. Electron Devices*, vol. 65, no. 10, pp. 4304–4312, Oct. 2018.

[186] P.-Y. Chen, X. Peng, and S. Yu, "NeuroSim+: An integrated device-to-algorithm framework for benchmarking synaptic devices and array architectures," in *Proc. IEEE Int. Electron Devices Meeting (IEDM)*, Dec. 2017, pp. 6.1.1–6.1.4. [Online]. Available: http://ieeexplore.ieee.org/document/8268337/

[187] S. Achour, R. Sarpeshkar, and M. C. Rinard, "Configuration synthesis for programmable analog devices with Arco," *ACM SIGPLAN Notices*, vol. 51, no. 6, pp. 177–193, Aug. 2016. [Online]. Available: https://dl.acm.org/doi/10.1145/2980983.2908116

[188] S. Achour and M. Rinard, "Noise-aware dynamical system compilation for analog devices with Legno," in *Proc. 25th Int. Conf. Archit. Support Program. Lang. Oper. Syst.*, Mar. 2020, pp. 149–166. [Online]. Available: https://dl.acm.org/doi/10.1145/3373376.3378449

[189] S. Misailovic, M. Carbin, S. Achour, Z. Qi, and M. C. Rinard, "Chisel: Reliability- and accuracy-aware optimization of approximate computational kernels," *ACM SIGPLAN Notices*, vol. 49, no. 10, pp. 309–328, Dec. 2014. [Online]. Available: https://dl.acm.org/doi/10.1145/2714064.2660231