High-Accuracy, High-Performance In-Memory Computing With High-Resistance Spin-Orbit Torque (SOT) Magnetic Memory

F. Carletti, *Graduate Student Member, IEEE*, M. Y. Song, E. Ambrosi, C. Y. Hu, C. F. Hsu, P. Mannocci, *Member, IEEE*, G. L. Chen, I. J. Wang, K. M. Chen, Y. C. Hsin, M. Farronato, *Member, IEEE*, X. Y. Bao, and D. Ielmini, *Fellow, IEEE*

***Abstract*— In-memory computing (IMC) has emerged as a promising solution for artificial intelligence (AI) accelerators due to the reduced data movement and improved parallelism in the crosspoint array. However, IMC faces several limitations such as the device variations affecting the computing accuracy, the area- and energy-consuming peripheral circuitry, and the time-consuming high-voltage program-verify operations of the nonvolatile memory (NVM). In addition, the relatively large summation currents cause IR drop within the array, leading to further loss of accuracy. To overcome these issues, this work presents 1-bit quantized networks based on spin-orbit-torque magnetoresistive random access-memory (SOT-MRAM) with high resistance–area (RA) product. We provide a detailed statistical characterization of SOT-MRAM arrays and develop quantization-aware training of various neural networks. Our results indicate that SOT-MRAM enables: 1) high inference accuracy, thanks to excellent uniformity; 2) negligible input-dependent IR drop nonlinearity, thanks to high resistance; and 3) high-speed, low-power reconfiguration, thanks to fast device programming. These results support high-RA SOT-MRAM for digital-like and reconfigurable IMC accelerators of edge AI.**

***Index Terms*— Artificial intelligence (AI), binary neural network (BNN), deep neural network (DNN), in-memory computing (IMC), spin-orbit-torque magnetic random access-memory (SOT-MRAM), ternary neural network (TNN).**

Received 8 September 2025; revised 3 November 2025 and 27 November 2025; accepted 1 December 2025. Date of publication 10 December 2025; date of current version 6 January 2026. This work was supported by European Research Council (ERC) through European Union's Horizon Europe Research and Innovation Program under Grant 101054098. The review of this article was arranged by Editor S. Alam. *(Corresponding author: F. Carletti.)*

F. Carletti, P. Mannocci, M. Farronato, and D. Ielmini are with the Dipartimento di Elettronica, Informazione e Bioingegneria (DEIB), Politecnico di Milano and IU.NET, 20133 Milan, Italy (e-mail: fabio.carletti@polimi.it; daniele.ielmini@polimi.it).

M. Y. Song, E. Ambrosi, C. Y. Hu, and C. F. Hsu are with Taiwan Semiconductor Manufacturing Company, Corporate Research, Hsinchu 300096, Taiwan.

G. L. Chen, I. J. Wang, K. M. Chen, and Y. C. Hsin are with the Industrial Technology Research Institute, Hsinchu 31040, Taiwan.

X. Y. Bao is with Taiwan Semiconductor Manufacturing Company, Corporate Research, San Jose, CA 95134 USA.

Digital Object Identifier 10.1109/TED.2025.3640599

I. Introduction

ARTIFICIAL intelligence (AI) is rapidly growing across all sectors, including health, industry, information, and society. Recent advances in large language models (LLMs) are revolutionizing the AI landscape, although the computational cost of training and inference tasks raises a concern about the energy sustainability of AI [1]. Two approaches have been proposed to improve the energy efficiency of AI, namely: 1) quantization of deep neural networks (DNNs) and 2) in-memory computing (IMC). Quantized DNNs can largely reduce model size and speed up inference while maintaining high performance. Weight precisions of just one or two bits have shown promising results [2], [3], [4], [5], while low-precision activation enables circuit simplification and high-speed operation [6], [7], [8]. However, even if recent studies have increasingly focused on this topic, DNN quantization typically results in degraded accuracy due to lower precision of weights and activations and more complex training [9], [10], [11], [12], [13], [14].

IMC can process data within the memory via massively parallelized matrix-vector multiplication (MVM), thus improving energy efficiency by orders of magnitude [15]. However, the current summation during inference can lead to large IR drop in the memory array impacting the computing accuracy and limiting the maximum size of the computational array [16]. To reduce IR drop, nonvolatile memories (NVMs) with a relatively high resistance of the low-resistive states (LRSs) should be developed. In addition, fast weight reconfiguration has become an essential asset for transformer-based LLMs, where matrix multiplication between activations is a key task in selective attention mechanism [17]. Fast reconfiguration requires high-speed reprogramming of the memory array, thus ruling out time-intensive program-verify algorithms of multilevel memory devices.

Here, we report an in-depth simulation-based analysis of IMC-based AI accelerators based on spin-orbit torque (SOT) magnetic random access memory (MRAM) displaying: 1) high resistance–area (RA) product enabling low IR drop

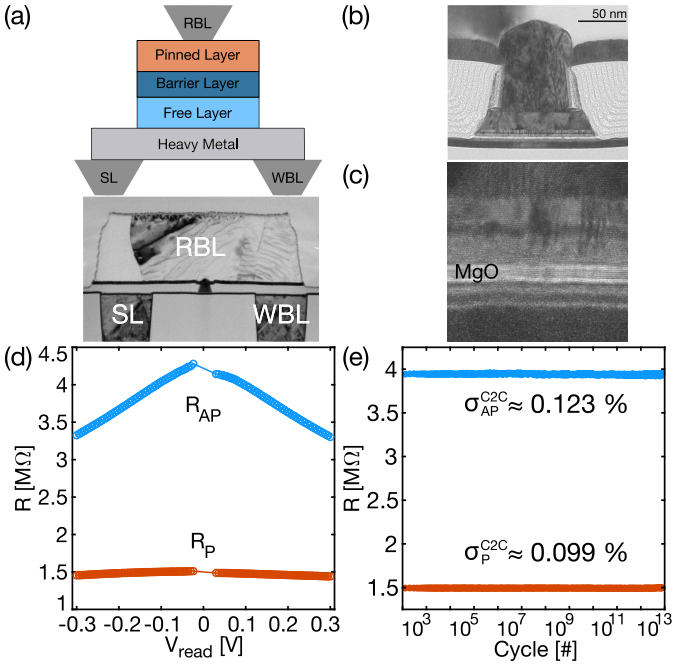


Fig. 1. (a) SOT-MRAM cell TEM with its simplified schematic. (b) Cell TEM cross section. (c) MTJ TEM cross section right above the SOT heavy metal. Relatively thick MgO enables *M*Ω range resistance. (d) Resistance as a function of the reading voltage. P state shows very high linearity opposed to slight nonlinearity of the AP state. (e) Low C2C variability is demonstrated up to 10^{13} write–read pulses.

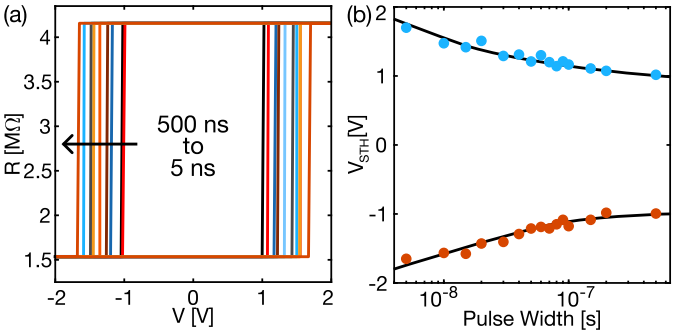


Fig. 2. Array R-V minor loops demonstrating repeatable and sharp switching at different voltage sweep times down to 5 ns. (b) Voltage needed to achieve a 50% switching probability at a given pulsewidth.

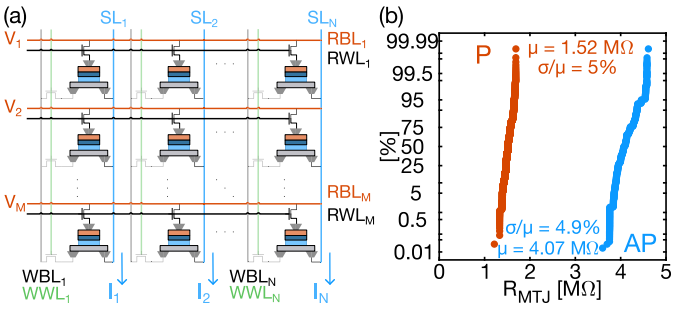


Fig. 3. (a) Sketch of array biasing during vector–matrix multiplication. For each SOT-MRAM only the read-access transistor is turned on. The write-access transistors are off. (b) Die-to-die distributions arising from 64 different dies. Ultralow variability with $\sigma/\mu \approx 5\,\%$ is obtained.

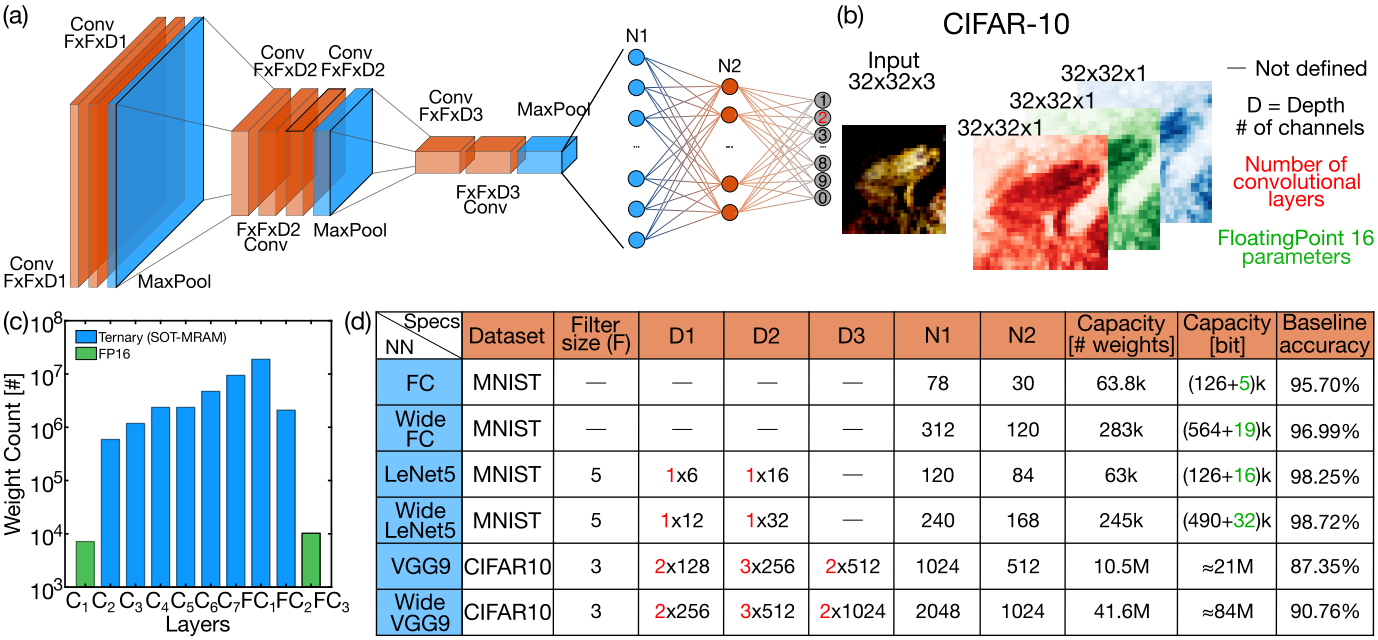
conductance distortion during MVM; 2) low device-to-device (D2D) variability enabling high accuracy in binary/ternary neural network (TNN) accelerators; and 3) fast reconfiguration (<5 ns) with negligible cycle-to-cycle (C2C) variability; and 4) high endurance ($\geq 10^{13}$ cycles) enabling the multiplication of activation matrices in transformer-based LLMs. These results highlight high-RA SOT-MRAM as an ideal memory technology for quantized IMC, offering high-precision computation, negligible hardware-induced performance drop, low energy consumption, and fast reconfiguration for next-generation AI accelerators.

II. Device and Array Characterization

Fig. 1(a) shows a schematic illustration of the SOT-MRAM device and the transmission electron microscopy (TEM) of a memory cell. The device consists of a magnetic-tunnel-junction (MTJ), serving as the read element, on top of a heavy-metal write line, serving as the write element for the electrical control of the MTJ parallel/antiparallel polarization [18], [19]. The three-terminal structure allows for separating the read and program paths; however, it raises the need for three separate metal lines, namely, the read bitline (RBL), the write bitline (WBL), and the source line (SL). The RBL and the WBL are used in the read and write phases, respectively, while the SL is accessed in both the phases. Fig. 1(b) shows a close-up cross section of an SOT-MRAM device, evidencing the MTJ region and the heavy-metal line. Fig. 1(c) shows the TEM cross section of the MTJ where the pinned layer, reference layer, and the MgO tunnel layer are clearly visible. A detailed description of the process flow can be found in [18]. Fig. 1(d) shows the resistance R of the MTJ measured across the RBL and the SL as a function of the readout voltage V_{read}. The LRS corresponds to parallel magnetization in the MTJ with an approximately constant resistance R_P, indicating ohmic behavior. On the other hand, the high-resistive state (HRS) corresponds to antiparallel magnetization in the MTJ, showing a decrease in resistance R_{AP} with V_{read}. Fig. 1(e) shows the measured R_P and R_{AP} as a function of the number of cycles during repeated set/reset operation. Resistance values are highly stable with negligible standard deviation σ of the C2C variation, in the range of about 0.1%. Data also indicate a high programming endurance above 10^{13}, thus supporting the ability to withstand write-intensive tasks, which are typical of transformer models [17]. Note that all the measurements have been performed at room temperature and ambient pressure.

Fig. 2(a) shows the measured R as a function of the voltage applied between WBL and SL for decreasing write times. The results demonstrates the possibility of low-voltage programming even at relatively short time, in the range of 5 ns. Fig. 2(b) shows the switching voltage V_{STH} marking the transition from LRS to HRS and vice versa as a function of the applied pulsewidth. The switching voltage was defined as the voltage causing a write-error rate (WER) of 50%. Increasing the voltage allows for shorter programming times and/or lower WER [18].

Fig. 3(a) shows a sketch of a memory array of SOT-MRAM devices, highlighting the applied voltages during a matrix–vector multiplication. The cell structure includes two transistors, serving as selecting transistors for the read path



Specs / NN	Dataset	Filter size (F)	D1	D2	D3	N1	N2	Capacity [# weights]	Capacity [bit]	Baseline accuracy
FC	MNIST	—	—	—	—	78	30	63.8k	(126+5)k	95.70%
Wide FC	MNIST	—	—	—	—	312	120	283k	(564+19)k	96.99%
LeNet5	MNIST	5	1x6	1x16	—	120	84	63k	(126+16)k	98.25%
Wide LeNet5	MNIST	5	1x12	1x32	—	240	168	245k	(490+32)k	98.72%
VGG9	CIFAR10	3	2x128	3x256	2x512	1024	512	10.5M	≈21M	87.35%
Wide VGG9	CIFAR10	3	2x256	3x512	2x1024	2048	1024	41.6M	≈84M	90.76%

Fig. 4. Neural network sketch and parameters. (a) CIFAR-10 dataset, comprising 60k RGB images belonging to ten different classes. (b) General architecture typically used for image recognition tasks. All the networks can be obtained by adding or removing layers and connections between them. (c) Capacity of each layer constituting wide VGG9 and weight precision. (d) Hyperparameters defining the networks used throughout this work.

and the write path, respectively. The gate of the read-select transistor is connected to the read word-line (RWL), while the gate of the write-select transistor is connected to the write word-line (WWL). The read currents of each device in the same column are accumulated at the SLs, thanks to Kirchhoff's current law, while an input vector $\mathbf{V}$ is applied to the RBLs. During MVM, the write transistors are all in the OFF-state by keeping the WWL voltage below the threshold voltage. Fig. 3(b) shows the cumulative distribution of the measured MTJ resistance for 64 dies on the same wafer, indicating a median tunneling magnetoresistance (TMR), defined as $(R_{AP} - R_P)/R_P$, of about 170%, with relatively low D2D variation of about 5%, thus supporting the ability to execute MVM with relatively high precision in the array.

III. Neural Networks

Previous studies have explored methods to enhance weight precision using multilevel cells (MLCs) based on innovative SOT-MRAM structures [20]. However, given the binary distribution in Fig. 3(b), our SOT-MRAM devices are most suitable for implementing binary neural networks (BNNs) [5], [6] and TNNs [4], [8]. For instance, TNN weights can be obtained with a differential weight scheme where the individual current is obtained as the subtraction of the currents at a positive device with conductance G_+ and a negative device with conductance G_-. The differential weight $W = G_+ - G_-$ thus allows for: 1) incorporation of both positive and negative weights in hardware and 2) cancellation of the relatively large OFF-state current of the HRS due to the relatively low TMR in Fig. 3(b) [21]. Three weights + 1, 0, and −1, can thus be obtained as 1) $W_{+1} = G_P - G_{AP}$, $W_0 = G_{AP} - G_{AP} = 0$, and $W_{-1} = G_{AP} - G_P = -W_{+1}$, respectively.

Another major burden for IMC is the relatively large overhead of area and power consumption mainly due to peripheral circuits, such as the analog-to-digital converters (ADCs), the analog programming circuits, and the digital circuits for activation, max-pooling, and batch normalization. To minimize the area and power overhead, we adopt binary activation, where the ADC is replaced by a simple sense amplifier or a comparator. The resulting neural network can thus be viewed as a binary-activation/ternary-weight neural network (BATWNN). Also note that digital circuits for batch normalization can be largely simplified in the case of binary activations [22], [23].

To investigate the IMC performance of SOT-MRAM, three different DNN architectures were considered, namely: 1) a fully connected (FC) network; 2) the convolutional network LeNet5; and 3) the convolutional network VGG9. In addition, two different datasets where considered, namely: 1) MNIST and 2) CIFAR-10 which is a reference dataset for computer vision.

Fig. 4(a) shows the general architecture that underlies all the considered DNN models in our work, where the model generally includes convolutional layers and FC layers. In the convolutional layers, F represents the size of the filter, while $D1$, $D2$, and $D3$ represent the number of channels for layers 1, 2, and 3, respectively. In the FC layers, generally serving as the classification module, $N1$ and $N2$ represent the number of neurons for each layer. Note that D critically impacts the network dependence on quantization and nonideality [24], [25], thus all the networks were studied for two values of D, where the networks with the largest D are referred to as the *wide* network. Fig. 4(b) shows a reference image for CIFAR-10, including low-resolution images belonging to ten different classes.

Fig. 4(c) shows the number of weights per layer for wide VGG9, the largest among the networks considered in this work, with a total of 41.6M parameters distributed among

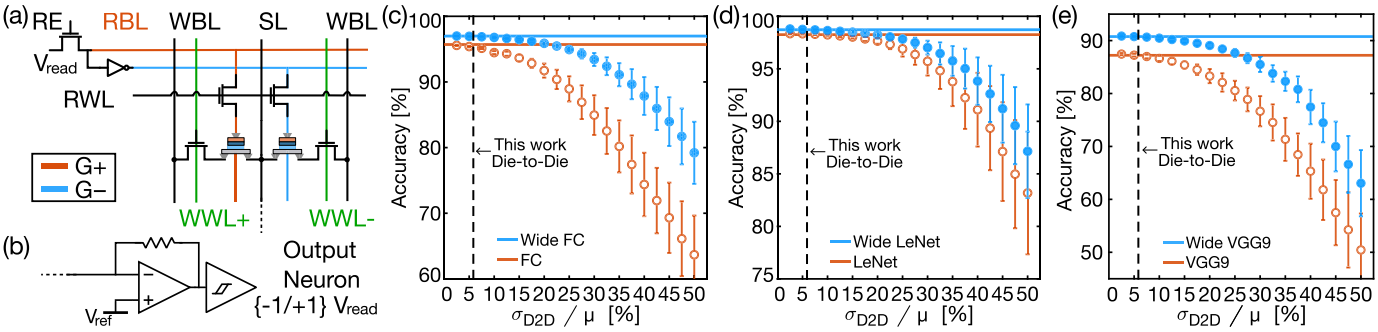


Fig. 5. D2D variability simulations. (a) Sketch showing how two different SOT-MRAMs are combined to obtain the differential cell. (b) Circuital sketch of the binary activation. (c) D2D variability simulations results for the FC networks. (d) Same as (c) for LeNet5 (e) Simulations for VGG9. It is possible to see how these binary–ternary networks reject extremely well the D2D variability and how the die-to-die variability of our process has no impact on the attainable accuracy.

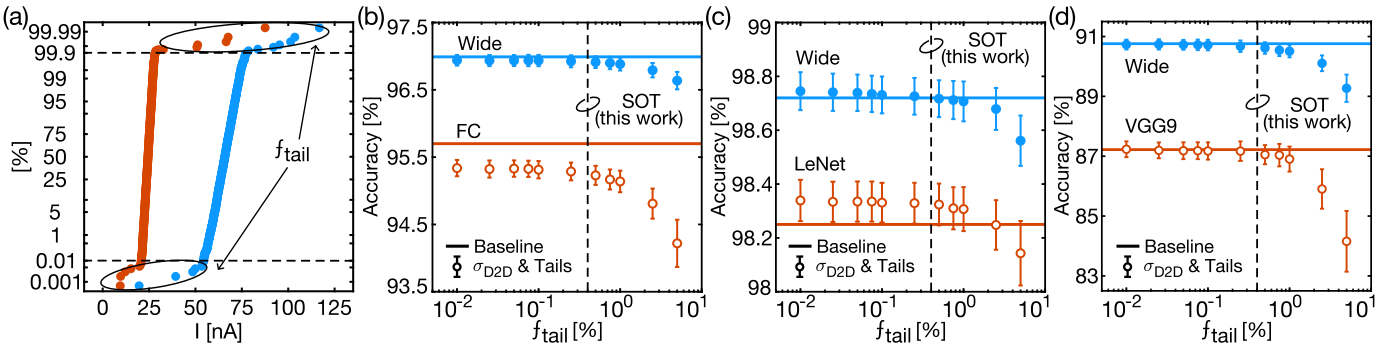


Fig. 6. (a) Simulated distribution for two 64 × 64 arrays considering high tails (f_{tail}^{H}) and low tails (f_{tail}^{L}) in addition to intrinsic Gaussian distribution. (b), (c), and (d) Simulation results for different percentage values assuming $f_{\text{tail}}^{H} = f_{\text{tail}}^{L} = f_{\text{tail}}$. Negligible impact on average network accuracy is demonstrated in all cases of interest of our process [18].

seven convolutional layers and three FC layers. The figure shows that although the first and last layers are generally computed with floating-point 16-bit (FP16) precision in conventional BNNs/TNNs, the overhead in terms of memory area is negligible. Fig. 4(d) summarizes the hyperparameters of the networks, their capacity, and their software baseline accuracy obtained by training the network. State-of-the-art (SOTA) accuracies for BNN/TNN are obtained. As a figure of merit (FOM) for the hardware accuracy, we considered the accuracy drop with respect to the software baseline, *i.e.*, the overall detrimental effect on the model performance induced by the hardware nonideality. This FOM is most important when investigating memory technologies for hardware neural accelerators, as it describes the precision of a given memory technology including variations and IR drop effects.

IV. Inference Simulations and Results

Fig. 5(a) shows the circuit schematic of the ternary weight implementation, consisting of two SOT-MRAM cells connected in a differential pair. Fig. 5(b) shows the schematic for the binary activation function, consisting of a simple sign(x) function realized with a simple transimpedance amplifier and comparator. This concept reduces design complexity and minimizes the area and power consumption of peripherals.

Fig. 5(c)–(e) shows the calculated accuracy for the FC, LeNet5 and VGG9, as a function of the D2D variability of the devices, for both the normal network and the wide version with twice the number of channels. More than 1000 TensorFlow simulation runs were performed to ensure a large statistical significance. A Gaussian distributions was assumed for both the conductance states G_P and G_{AP}, with relative variability σ, leading to a ternary weight variability given by

$$\sigma_{\pm 1} = \sqrt{\left(\sigma_{\text{LRS}} * \mu_{I_{\text{LRS}}}\right)^2 + \left(\sigma_{\text{HRS}} * \mu_{I_{\text{HRS}}}\right)^2} \tag{1}$$

for ternary weights W_{+1} and W_{-1}, and

$$\sigma_{0} = \sqrt{\left(\sigma_{\text{HRS}} * \mu_{I_{\text{HRS}}}\right)^2 + \left(\sigma_{\text{HRS}} * \mu_{I_{\text{HRS}}}\right)^2} \tag{2}$$

for the ternary weight W_0. It is worth mentioning that the relative variability of SOT-MRAM devices, namely, $\sigma/\mu = 5\%$, causes no drop of accuracy in any of the considered networks. Note that wide networks are more robust to device variability, thanks to the larger number of channels, which, however, leads to a larger number of parameters, hence larger circuit area and energy consumption. This accuracy–area–energy tradeoff is critical in defining the performance of application-specific edge AI accelerators based on IMC.

A. Impact of Process Tails

To account for possible process nonidealities, we considered statistical tails as shown in the calculated distributions in Fig. 6(a) with probability f_{tail}^{H} and f_{tail}^{L} for the high and low tails, respectively. Fig. 6(b)–(d) shows the accuracy for FC, LeNet5, and VGG9, respectively, as a function of the tail probability, assuming $f_{\text{tail}}^{H} = f_{\text{tail}}^{L} = f_{\text{tail}}$. Tail currents ranging from 0 to 250 nA, more than three times the nominal current of the P state, have been included. These relatively higher conductances carry enough current to completely dominate the output of a 3 × 3 convolution filter and substantially impact the results of 5 × 5 filters. We did not include the

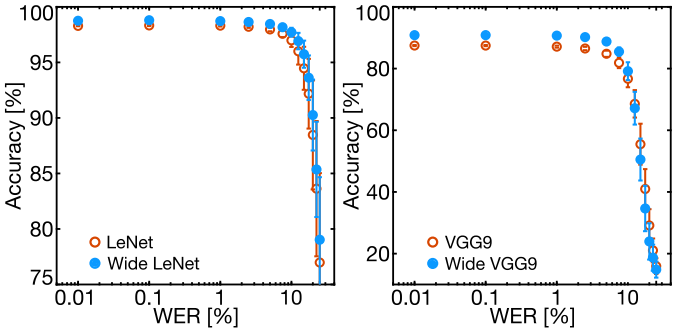


Fig. 7. Hardware accelerator open-loop programming considering different WER percentages. Inference accuracy is not affected for WER as high as 2%. Width and model overfitting affect network response at high WER (≥ 5%) percentages.

case of short-circuited MTJs which may occur due to etch residues bridging the MTJs electrodes [26]; however, in an industry-level grade process, such defects appear only at the parts-per-million level and would be excluded through appropriate mapping and redundancy strategies [27], [28]. The simulation results indicate a good robustness against tails due to process-induced defects with negligible impact on accuracy for $f < 2\%$.

B. Impact of Programming Errors

To assess how stochastic programming in SOT-MRAM switching affects hardware reliability, we performed TensorFlow simulations of inference accuracy as a function of the WER. The model accounts for all possible weight perturbations arising from both first-order and second-order bit flips. A first-order flip occurs when only one device within a differential cell experiences a write error, whereas a second-order flip occurs when both the devices in the differential pair are written incorrectly.

Fig. 7 shows the simulated accuracy for LeNet5 and VGG9. The simulation results show that a WER smaller than 2% is perfectly rejected. By increasing the WER above 5-10%, the accuracy sharply decreases for both the models. These results highlight the potential for quickly reconfigurable accelerators using fast and low-energy program schemes, without a time-consuming closed-loop program-verify methodology. Similar results were obtained for the FC network (not shown).

C. Impact of the IR Drop

To quantify the impact of IR drop, we developed an analytical Python model consistent with the formulations in [29] and [30]. Before each inference experiment, we introduced our 5% D2D conductance variations, mapped the resulting weights onto differential arrays, and then computed the IR-drop induced degradation of the effective conductances. Classification was performed in TensorFlow using these degraded weights. This entire procedure was repeated 1000 times to fully capture the statistical impact of D2D variability, which however turned out to be of negligible importance to assess the IR-drop effect. Fig. 8 shows the calculated accuracy drop

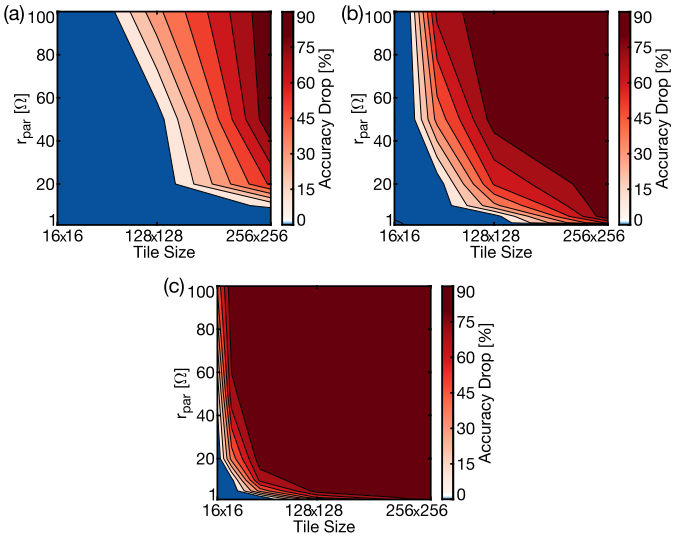


Fig. 8. Simulated IR-induced degradation of accuracy as a function of tile size r_{par} and device conductance. (a) High RA with G_P = 660 nS. (b) Medium RA with G_P = 6.6 μS. (c) Low RA with G_P = 66 μS. A TMR = 170% was assumed in all cases.

as a function of the tile size and the parasitic resistance r_{par} between two neighbor cells along the SL for wide LeNet. Three different RA products are considered, namely, high RA Fig. 8(a), medium RA Fig. 8(b), and low RA Fig. 8(c). Thanks to the high resistance in the MΩ range, high accuracy can be achieved in relatively large arrays with relatively high parasitic resistance of 10 Ω. On the other hand, medium and low-RA arrays are more heavily affected by IR drop, due to the higher summation current along the SL.

V. Memory Benchmarking

We now compare the performance achieved in the studied architectures among various NVM technologies such as resistive-random access-memory (RRAM) [31], phase-change-memory (PCM) [32] 3-D cross-point (3DXP) devices [33], and spin-transfer-torque (STT) MRAM [34].

Fig. 9 shows the average drop of accuracy with respect to software baseline due to the hardware implementation according to reported measured D2D and C2C variations. Our SOT-MRAM device features the smallest accuracy drop among the considered NVM technologies.

In some cases, the accuracy drop is negative for the SOT-MRAM devices, namely, the hardware implementation shows a higher accuracy compared with the software baseline. This could be due to a better approximation of the continuous minima in the optimization space, which parameters are bound to be quantized by software when optimizing with $\{-1, 0, 1\}^{N_{weights}}$ as hyperparameter search space.

We then compare the performance of the SOTMRAM technology with other NVMs in the context of IMC. Fig. 10 shows the average cell current σ_I and its standard deviation σ_I. SOT-MRAM devices display: 1) a relatively low read current which allows to minimize IR-drop effects and thus to increase the array tile size; 2) extremely low D2D

eNVM σ_I/μ / NN	PCM 10%	3DXP 15%	RRAM 12%	STT 15%	SOT (this work) 5%
FC	0.58	2.05	1.4	2.05	0.31
Wide FC	0.22	0.54	0.36	0.54	0.04
LeNet5	0.07	0.27	0.15	0.27	-0.11
Wide LeNet5	0.09	0.26	0.15	0.26	-0.03
VGG9	0.59	0.85	1.14	0.85	-0.02
Wide VGG9	0.12	1.83	0.59	1.83	0.01

Fig. 9. Additional test error (Δ) due to hardware implementation assuming the σ_{D2D} reported below each device name. Results coming from Fig. 5(c). This work shows the lowest Δ in all cases even considering the die-to-die distributions. Smaller Δ should be obtained by moving to intradie data.

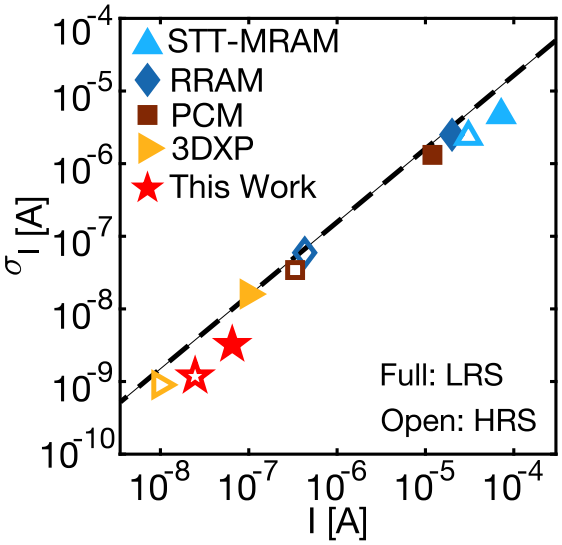


Fig. 10. Standard deviation as a function of average current for high RA SOT-MRAM and other emerging NVMs. Our devices achieve superior variability with respect to technologies reported in literature.

eNVM Param	PCM	3DXP	RRAM	STT	SOT (this work)
μ_{LRS}/μ_{HRS}	100	10	>100	2.65	2.7
μ_{LRS} [A]	10 μA	100 nA	20 μA	28 μA	65 nA
σ_I/μ	10%	15%	12%	7%	<5%
Write verify	Yes	Yes	Yes	No	No
t_{write} [ns]	>100	>100	N.A.	10	<10
E_{write} [pJ]	10	N.A.	$>10^3$	>1	0.65
E_{read} [fJ]	36	N.A.	40	>100	0.66
IR drop	Yes	No	Yes	Yes	No
Endurance	10^7	$>10^7$	10^7	10^{11}	$>10^{13}$
Number of bits	3	1	3	1	1
Cell size (F^2)	Low (6-10)	Lowest (4/n)	Lowest (4/n)	Medium (6-20)	High (24-40)

Fig. 11. In depth eNVMs comparison. SOT-MRAM has the lower hand on information density, especially with respect to vertical technologies, n being the number of layers stacked. On the other hand, our devices show more important features for computing, like the lowest current, lowest variability, write time, and write/read energy. Moreover $M\Omega$ resistances allow to completely neglect IR-drop for arrays up to 256×256 for both the training and inference phases.

Ref. Param	[20]	[35]	[36]	(This work)
RA	Medium	Low	Low	High
σ_I/μ	12%	10%	11%	<5%
TMR	100%	115%	171%	170%
Write verify	Yes	No	Yes	No
IR drop	High	Highest	High	Negligible
Number of bits	3	3	1	1
SOT track size	High	Medium	Low	Low
Process complexity	Medium	High	Low	Low

Fig. 12. Comparison between different SOT-MRAM basic cells. Even when compared with multipillar or multistack approaches that can partially compensate the low bit-density of SOT-MRAM, our cell shows the best properties for computing applications.

variations, namely, about 5% compared with about 15% displayed by other NVM technologies; and 3) a relatively low ON/OFF-ratio of SOT-MRAM which, however, can be compensated by the differential weight scheme. The small variation in SOT-MRAM devices enables high accuracy in IMC-based AI accelerators. Fig. 11 summarizes the main performance of NVM technologies, compared with SOT-MRAM. While area performance is the worst among other technologies, in our opinion this is not so important for computing applications where typically readout electronics and scratchpad memory (SPM) dominate the area requirements. On the other hand, high-RA SOT-MRAM displays unique properties for quantized IMC such as: 1) extremely fast and low-energy hardware reconfiguration with no verify, thanks to negligible C2C variation and network WER resilience; 2) high endurance for write-intensive tasks such as attention layers; 3) high resistance, hence low read-energy and negligible IR drop; and finally 4) low D2D variations for precise software emulation.

The lack of multilevel operation and high HRS leakage are fully mitigated by the ternary weight approach by the differential weight mapping. These characteristics position high-RA SOT-MRAM as the optimal eNVM device for hardware accelerators in quantized architectures.

Finally, Fig. 12 shows comparison of our SOT-MRAM basic cell with other SOT implementations, two of which

developed to address the absence of multilevel operation [20], [35], [36]. Although multipillar and multistack designs boost the effective bits per cell, their increased process complexity, larger area requirements, susceptibility to variations along with lower TMR, and more intricate write mechanisms render these approaches impractical for reconfigurable applications and precise computing. In the end, our SOT-MRAM technology emerges as the ideal candidate for IMC accelerators of quantized DNNs and LLMs.

VI. Conclusion

This work provides an in-depth study of in-memory BATWNN for image recognition based on SOT-MRAM with high RA product. Variation-aware simulations show that SOT-MRAM is the best candidate for hardware accelerators in terms of software fidelity. The low WER of SOT-MRAM at programming times below 10 ns, combined with the high WER rejection of these networks, enables the fast reconfiguration of relatively large models with relatively low energy consumption. In addition, the relatively high resistance of high-RA MTJs enables negligible IR drop at a relatively large array size. These results support the high-RA SOT-MRAM technology for highly accurate, reconfigurable, and energy-efficient IMC hardware accelerators of edge AI.

References

[1] K. Crawford, "World view," *Nature*, vol. 626, p. 693, Jan. 2024.

[2] H. Wang et al., "BitNet: Scaling 1-bit transformers for large language models," 2023, *arXiv:2310.11453*.

[3] S. Ma et al., "The era of 1-bit LLMs: All large language models are in 1.58 bits," 2024, *arXiv:2402.17764*.

[4] J. Sundaram and R. Iyer, "LLaVaOLMoBitnet1B: Ternary LLM goes multimodal!," 2024, *arXiv:2408.13402*.

[5] F. Li, B. Liu, X. Wang, B. Zhang, and J. Yan, "Ternary weight networks," 2016, *arXiv:1605.04711*.

[6] M. Rastegari, V. Ordonez, J. Redmon, and A. Farhadi, "XNOR-Net: ImageNet classification using binary convolutional neural networks," in *Proc. Eur. Conf. Comput. Vis.*Cham, Switzerland: Springer, 2016, pp. 525–542.

[7] I. Hubara, M. Courbariaux, D. Soudry, R. El-Yaniv, and Y. Bengio, "Binarized neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 29, 2016, pp. 4107–4115.

[8] H. Alemdar, V. Leroy, A. Prost-Boucle, and F. Pétrot, "Ternary neural networks for resource-efficient AI applications," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, May 2017, pp. 2547–2554.

[9] Z. Liu et al., "ParetoQ: Improving scaling laws in extremely low-bit LLM quantization," 2025, *arXiv:2502.02631*.

[10] T. Chen et al., "TernaryLLM: Ternarized large language model," 2024, *arXiv:2406.07177*.

[11] A. Kaushal, "Trilm vs floatLM: Ternary LLMs are more performant than quantized FP16 LLMs," in *Proc. ICML Workshop Found. Models Wild*, Jul. 2024, pp. 1–12.

[12] M. Dehghankar, M. Erfanian, and A. Asudeh, "An efficient matrix multiplication algorithm for accelerating inference in binary and ternary neural networks," 2024, *arXiv:2411.06360*.

[13] Y. Shang, Z. Yuan, Q. Wu, and Z. Dong, "PB-LLM: Partially binarized large language models," 2023, *arXiv:2310.00034*.

[14] J. Chen et al., "LoTA-QAF: Lossless ternary adaptation for quantization-aware fine-tuning," 2025, *arXiv:2505.18724*.

[15] D. Ielmini and H.-S.-P. Wong, "In-memory computing with resistive switching devices," *Nature Electron.*, vol. 1, no. 6, pp. 333–343, Jun. 2018.

[16] 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.

[17] A. Vaswani et al., "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2025, pp. 5998–6008.

[18] M. Y. Song et al., "High speed (1ns) and low voltage (1.5 V) demonstration of 8Kb SOT-MRAM array," in *Proc. IEEE Symp. VLSI Technol. Circuits (VLSI Technol. Circuits)*, Jun. 2022, pp. 377–378.

[19] X. Han, C. Wan, and G. Yu, "Materials, physics, and devices of spin–orbit torque effect," *Appl. Phys. Lett.*, vol. 118, May 2021, Art. no. 180401.

[20] J. Doevenspeck et al., "Multi-pillar SOT-MRAM for accurate analog in-memory DNN inference," in *Proc. Symp. VLSI Technol.*, May 2021, pp. 1–2.

[21] M. Y. Song et al., "High RA dual-MTJ SOT-MRAM devices for high speed (10ns) compute-in-memory applications," in *IEDM Tech. Dig.*, Dec. 2023, pp. 1–4.

[22] 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.

[23] S. Hwang, W. Lee, J. W. Park, and D. Suh, "First realization of batch normalization in flash-based binary neural networks using a single voltage shifter," *IEEE Trans. Nanotechnol.*, vol. 23, pp. 677–683, 2024.

[24] A. Mishra, E. Nurvitadhi, J. J. Cook, and D. Marr, "WRPN: Wide reduced-precision networks," 2017, *arXiv:1709.01134*.

[25] J. Doevenspeck et al., "Noise tolerant ternary weight deep neural networks for analog in-memory inference," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, vol. 15, Mar. 2021, pp. 1–8.

[26] M. Ji et al., "Study on the effect of re-deposition induced by ion beam etching on MTJ performances," *AIP Adv.*, vol. 9, Aug. 2019, Art. no. 085317.

[27] Q. Xu et al., "Fault tolerance in memristive crossbar-based neuromorphic computing systems," *Integration*, vol. 70, pp. 70–79, Jan. 2020.

[28] O. Yousuf et al., "Layer ensemble averaging for fault tolerance in memristive neural networks," *Nature Commun.*, vol. 16, no. 1, p. 1250, Feb. 2025.

[29] F. Corinto, "Modelling memristive devices via ideal memristor and nonlinear resistors," in *Proc. 19th Int. Conf. Synth., Modeling, Anal. Simulation Methods Appl. Circuit Design (SMACD)*, Jul. 2023, pp. 1–4.

[30] G. Zoppo et al., "A mathematical formulation of the wire resistance problem in memristor crossbars," in *Proc. IEEE 22nd Int. Conf. Nanotechnol. (NANO)*, Jul. 2022, pp. 461–464.

[31] D. Bridarolli et al., "High-density multilevel 3D vertical resistive switching memory (VRRAM) for massively parallel in-memory computing," in *IEDM Tech. Dig.*, Dec. 2024, pp. 1–4.

[32] L. Pistolesi et al., "Differential phase change memory (PCM) cell for drift-compensated in-memory computing," *IEEE Trans. Electron Devices*, vol. 71, no. 12, pp. 7447–7453, Dec. 2024.

[33] F. Carletti et al., "Low-energy, high-accuracy convolutional network inference in 3D crosspoint (3DXP) arrays," in *Proc. IEEE Eur. Solid-State Electron. Res. Conf. (ESSERC)*, Sep. 2024, pp. 412–415.

[34] T.-N. Pham, Q.-K. Trinh, I.-J. Chang, and M. Alioto, "STT-BNN: A novel STT-MRAM in-memory computing macro for binary neural networks," *IEEE J. Emerg. Sel. Topics Circuits Syst.*, vol. 12, no. 2, pp. 569–579, Jun. 2022.

[35] V. Nehra, S. Prajapati, T. N. Kumar, and B. K. Kaushik, "High-performance computing-in-memory architecture using STT-/SOT-based series triple-level cell MRAM," *IEEE Trans. Magn.*, vol. 57, no. 8, pp. 1–12, Aug. 2021.

[36] Z. He, S. Angizi, F. Parveen, and D. Fan, "High performance and energy-efficient in-memory computing architecture based on SOT-MRAM," in *Proc. IEEE/ACM Int. Symp. Nanosc. Archit. (NANOARCH)*, Jul. 2017, pp. 97–102.