Modeling and compensation of IR drop in crosspoint accelerators of neural networks

N. Lepri, *Student Member, IEEE*, M. Baldo, *Student Member, IEEE*, P. Mannocci, *Student Member, IEEE*, A. Glukhov, *Student Member, IEEE*, V. Milo, *Member, IEEE*, and D. Ielmini, *Fellow, IEEE*

***Abstract*— In-memory computing has the potential of accelerating data-intensive computing tasks, such as inference and training of neural networks, by the matrix vector multiplication (MVM) in the crosspoint array. As the array size increases, however, a significant problem arises due to the parasitic IR drop at row and column lines. Understanding and compensating IR drop is essential to enable high-density accelerators of MVM. This work presents an analytical model for the IR drop and a numerical algorithm capable of accelerating the nodal analysis of crosspoint arrays up to 5 orders of magnitude with respect to SPICE analysis. The numerical algorithm is used to study the impact of array size, data density, device/wire resistance and non-linearity on the IR drop and the related loss of accuracy in IMC. We also derive two simple compensation schemes and an architectural solution for the mitigation of IR drop that are able to increase the accuracy of neural networks from 59% up to 96.6%, compared to an ideal accuracy of 97%.**

***Index Terms*— Hardware accelerator, In-memory computing (IMC), Neural network, Nonvolatile memory, Resistive switching memory.**

I. Introduction

In-memory computing (IMC) is attracting an increasing interest as it overcomes the memory bottleneck of conventional von Neumann architectures [1], [2]. The most significant advantage of IMC is the parallel matrix-vector multiplication (MVM) which can be executed in a crosspoint array in the analogue domain [3]. As shown in Fig. 1a, MVM can be executed in just one step by applying a voltage vector at the row terminals and collecting the currents at column terminals. Based on Ohm's and Kirchhoff's law, the resulting current is $I = GV$, where G is the conductance matrix, I is the current vector and V the voltage vector. It has been reported that crosspoint MVM can improve the throughput of neural networks by a factor $30,000$, thanks to the parallel MVM concept [4]. A major concern of crosspoint MVM is the relatively large current, due to the parallel activation of several devices at the same time. The large current results in a relatively large parasitic voltage drop across the array rows and columns, referred to as IR drop (Fig. 1b) [5]. IR drop was previously taken into account by on-line training in hardware [6] or by including the effects of IR drop in off-line software training via circuit simulations [7]–[9]. More general compensation schemes have been proposed via calibration of scaling factors [10], [11], calibration of the synaptic weights [12], [13] or mapping and partitioning algorithms [3], [13], [14]. Other mitigation techniques involve device optimization by conductance limitation [4], [15].

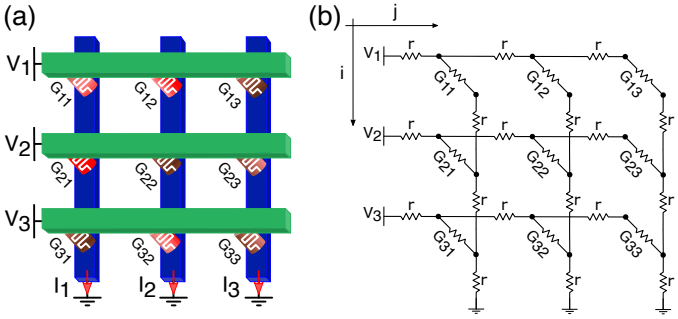


Fig. 1. Illustration of the MVM operation in a crosspoint array with 1R configuration (a) and its lumped-parameter model including wire resistances causing IR drop (b).

This work proposes new tools and understandings for modeling, simulation and compensation of IR drop in crosspoint arrays. In particular, the paper is organized as follows: by adopting a lumped-parameter model along the same lines as the one proposed in [16], Sec. II describes the simulation framework for predicting the IR drop in crosspoint arrays and proposes a continuous analytical model for simplified scenarios. Sec. III presents the numerical algorithm needed to accelerate the nodal analysis, Monte Carlo simulations are presented in Sec. IV to study the impact of IR drop as a function of array size, pattern of stored data, wire resistance, device conductance and linearity. Then, two compensation schemes and an architectural solution for IR drop are proposed in Sec. V. Finally, in Sec. VI the proposed techniques are validated during neural network inference, demonstrating an increase of the accuracy from 59% up to 96.6%, close to the ideal accuracy of 97%. This work can contribute to the optimization and design of in-memory neural networks with high accuracy and energy efficiency.

This work was supported by the Ministero degli Affari Esteri e della Cooperazione Internazionale (grant number PGR01011).
N. Lepri, M. Baldo, P. Mannocci, A. Glukhov and D. Ielmini are with the Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano and IU.NET, 20133 Milan, Italy (e-mail: nicola.lepri@polimi.it).
V. Milo was with the Dipartimento di Elettronica, Informazione e Bioingegneria (DEIB), Politecnico di Milano and IU.NET, 20133 Milan, Italy (e-mail: valerio.milo@polimi.it). Now he is with Applied Materials Italia Srl, 42124 Reggio Emilia, Italy (e-mail: valerio_milo@amat.com).

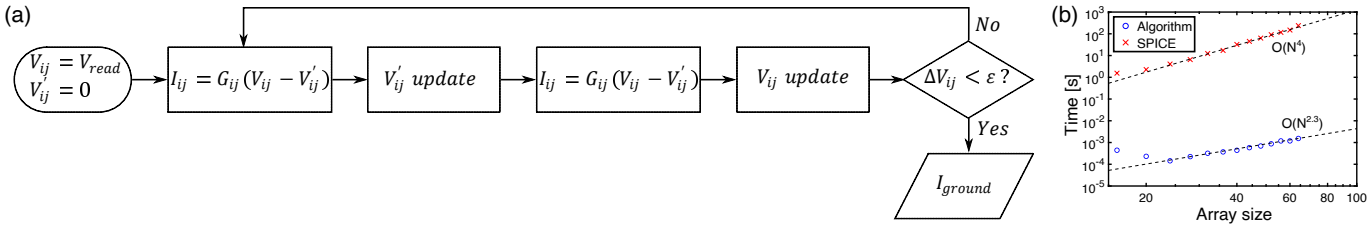


Fig. 2. Description of the numerical algorithm for nodal analysis of the crosspoint array. (a) Flowchart of the algorithm and (b) measured computing times of the proposed algorithm with respect to traditional SPICE analysis as a function of the array size.

II. Analytical Model of IR Drop

We will adopt a traditional lumped-parameter model for the circuital simulations, exemplified in Fig. 1b by a one-resistor (1R) crosspoint array of resistive memories. The crosspoint array is assumed to have a square shape with size $N \times N$. In the crosspoint circuit, each memory device can be a resistive switching memory (RRAM), a phase change memory (PCM) or any other resistive device with reconfigurable conductance G. The row and column wires are assumed to be connected at the top electrode (TE) and bottom electrode (BE) of each device, respectively, and have a finite non-zero resistance r between each node, usually in the range of few Ω for sub-100 nm CMOS technologies [17]. By applying Kirchhoff's laws, the TE potential $V_{i,j}$ at the generic node at row i and column j is given by:

$$V_{i,j} = V_{i,j-1} - r \cdot \sum_{k=j}^{N} G_{i,k}(V_{i,k} - V'_{i,k}), \qquad (1)$$

where $V'_{i,j}$ and $G_{i,j}$ are the BE potential and the conductance of the device at row i and column j. The conductance $G_{i,j}$ might be an analogue value or it might display binary values, namely G_{LRS} for the low resistance state (LRS) and G_{HRS} for the high resistance state (HRS). In Eq. (1), the voltage drop across the cell-cell resistance r includes the IR drop due to the current of cell (i, j) and all cells with higher index j on the same row. Similarly, the BE potential $V'_{i,k}$ at the generic node at column i and row k is given by:

$$V'_{i,j} = V'_{i-1,j} - r \cdot \sum_{k=1}^{i-1} G_{k,j}(V_{k,j} - V'_{k,j}), \qquad (2)$$

where the voltage drop across r now includes the current of all the cells with lower index i on the same column of device (i, j). Boundary conditions are $V(i, 0) = V_i$, where V_i is the applied voltage at row i, and $V'(N, j) = 0$, as all column terminals are grounded, or connected to the virtual ground of an analogue amplifier to sense the current I. By combining Eqs. (1) and (2) with their boundary conditions, potentials V and V' can be obtained via numerical solution of the set of $2N$ equations. The same analytical description can be extended to other types of crosspoint arrays, such as one-selector/one-resistor (1S1R) or the one-transistor/one-resistor (1T1R) structure by including the voltage drop across the select device or with a suitable approximation, *e.g.* a linear model.

In the particular cases of uniform G and V, it is straightforward to extend Eqs. (1) and (2) to the continuous domain. In this domain, the crosspoint array is described as a continuous 2D surface with square shape of size L, where r and G are normalized to the unit length of the structure, resulting in units Ωm^{-1} and Sm^{-1} for r and G respectively. In particular, in a 'worst-case scenario' with all cells set to the maximum conductance value G Eqs. (1) and (2) become:

$$\frac{\partial V(x,y)}{\partial x} = -rG \int_x^L (V(x',y) - V'(x',y))dx' \qquad (3)$$

$$\frac{\partial V'(x,y)}{\partial y} = -rG \int_0^y (V(x,y') - V'(x,y'))dy', \qquad (4)$$

which are defined in the domain $0 < x < L$ and $0 < y < L$. Boundary conditions read:

$$V(0,y) = V_{read} \qquad V'(x,L) = 0 \qquad (5)$$

where V_{read} is the boundary potential for $x = 0$.

The proposed formulation can be useful to evaluate the profiles of V and V' in presence of a uniform G and V, or when they can be expressed by analytical function of position x and y. Also, the continuous model can benefit from well-optimized, fast finite-element method (FEM) solvers of partial differential equations (PDEs).

III. Numerical Algorithm for IR Drop Evaluation

To assess the impact of IR drop in applications such as the inference of a neural network, it is mandatory to develop a tool capable of rapidly computing the nodal analysis of the array by Eqs. (1) and (2), since software based on SPICE would result in an unfeasible computational time. For this purpose, we developed an iterative numerical algorithm, whose flowchart is reported in Fig. 2a. The procedure is organized as follows:

1) The current of each cell is at first ideally computed as $I_{i,j} = \mathbf{V}_i \cdot G_{i,j}$, where $\mathbf{V}$ is the applied voltage vector and $G_{i,j}$ is the cell conductance.
2) For each column j, moving from the underlying virtual ground we obtain a first estimation of the BE voltage $V'_{i,j}$ by summing the computed current contributions of each cell and the corresponding voltage drops on the parasitic resistances along the column.
3) At this point, each cell current can be better estimated as $I_{i,j} = (\mathbf{V}_i - V'_{i,j}) \cdot G_{i,j}$.
4) For each row i, starting from the applied voltage $\mathbf{V}_i$ we obtain an estimation of the TE voltage $V_{i,j}$ by summing the computed current contributions and the corresponding voltage drops along the row.

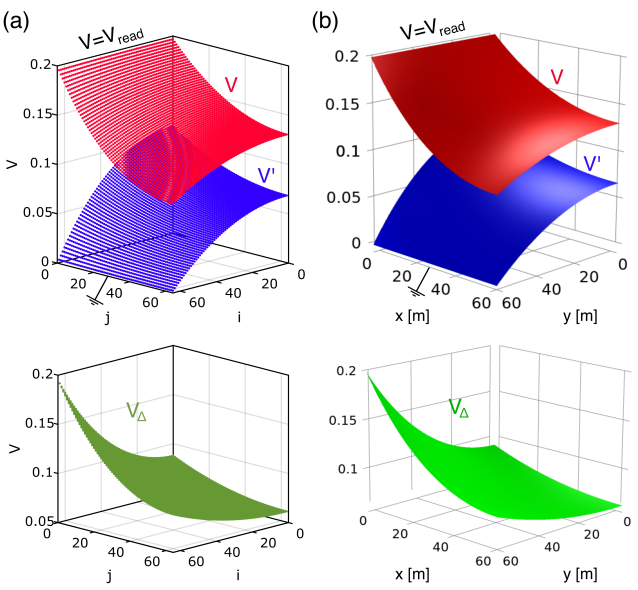


Fig. 3. Calculated $\boldsymbol{V}$, $\boldsymbol{V'}$ and $\boldsymbol{V_\Delta}$ for the discrete problem (a) and the continuous domain (b) in the worst-case scenario.

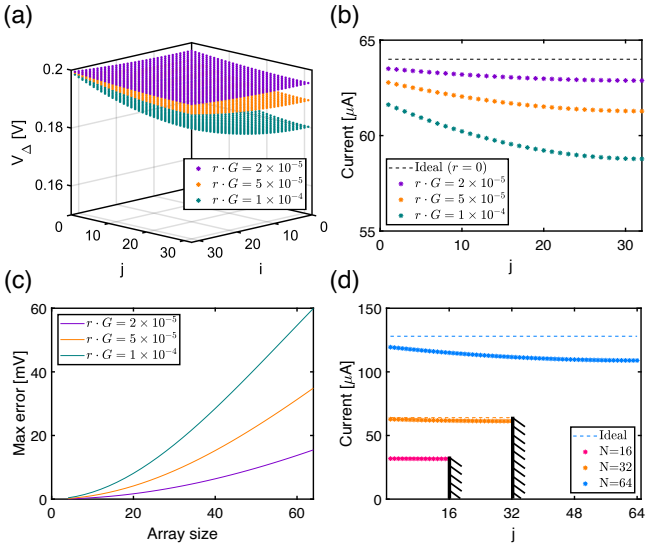


Fig. 4. Calculated $\boldsymbol{V_\Delta}$ (a) and column current (b) for increasing $\boldsymbol{rG}$ ($\boldsymbol{N = 32}$). Calculated maximum error $\boldsymbol{V_{read} - V_\Delta}$ (c) and column current (d) for increasing array size $\boldsymbol{N}$.

5) Using the obtained estimations, each cell current is now computed as $I_{i,j} = (V_{i,j} - V'_{i,j}) \cdot G_{i,j}$.
6) Along the same lines of point 2), by calculating the IR drop along columns with the last current estimations we compute a new guess for V' and we average it with the previous one. This allows to refine the estimation of the cell currents and hence of V, computing IR drop through rows and averaging the guess with the previous one. The cycle is repeated until the difference between two subsequent iterations becomes lower than a chosen threshold.

The proposed algorithm can compute the nodal analysis of a crosspoint array with negligible error in few iterations, outperforming SPICE analysis of several orders of magnitude in terms of computational time. Fig. 2b shows the computing time of our numerical model and SPICE, evaluated with an Intel i7 exa-core CPU with clock frequency of 2.6 GHz and 32 GB of RAM. This massive improvement is mainly due to the decoupling of the two dimensions, since at each step calculation regards only one of the two axis. In particular, the simulation results in Fig. 2b indicate a computational complexity of $O(N^{2.3})$, compared to $O(N^4)$ for the SPICE analysis. In a practical case of a 64x64 random MVM the algorithm is five orders of magnitude faster than SPICE.

Depending on the framework and on the application, the algorithm can be further refined, e.g. by using a weighted average unbalanced toward the last guess to speed up the computation or adding pre-conditioning corrective matrices in point 1) and 3), that can be extracted from approximated solutions of Eqs. (3)-(4) for G and V set at the average conductance $\langle G \rangle$ and average input voltage $\langle V \rangle$.

The convergence domain of the algorithm is sufficiently large for our application. In case of extremely severe IR drop or extremely large array, it is possible to broaden it further by unbalancing the aforementioned weighted average toward the previous iteration in order to reduce the oscillation of the consequent guesses, in view of a minor increase of the computational time. Finally, it is worth to notice that the structure of the proposed algorithm is easily extendable to arrays with non-linear cells or to more complex architectures.

IV. Simulation Results

By solving Eqs. (1)-(2), one can extract the potentials V, V' and the voltage drop across the generic device $V_\Delta = V - V'$ in either the discrete or the continuous domain for a 'worst-case scenario', where a uniform voltage is applied to the rows and all cells are programmed in the LRS, assuming a product $rG = 5 \times 10^{-4}$. This might correspond, for instance, to a cell-cell wire resistance $r = 10\ \Omega$ and a device conductance $G = 50\ \mu S$, which is consistent with the LRS of a typical HfO_2-based RRAM device [18]. The wire resistance can be obtained as $r = \rho \times 2FA^{-1} = 2\rho F^{-1}$, where ρ is the metal resistivity, F is the feature size controlling the cell-cell pitch $2F$ and the wire cross section $A = F^2$. Assuming a tungsten wire with $\rho = 10\ \mu\Omega cm$ [19] and $F = 20$ nm, we obtain $r = 10\ \Omega$. Fig. 3 shows the calculated V, V' (top) and V_Δ (bottom) in the discrete (a) and continuous (b) domain, respectively. An array size $N = 64$ and a nominal voltage $V_{read} = 0.2$ V were assumed, except were noted.

Fig. 3 shows an excellent agreement between discrete and continuous simulation results, indicating a decay of V and V' along the row and column direction, as a result of the IR drop. Due to the particular 'worst-case' configuration, V_Δ shows a symmetrical shape with monotonic decay for increasing j and decreasing i. In particular, V_Δ approaches the ideal value V_{read} only in correspondence of $j = 1$ and $i = N$, for which there is only one cell between row and column terminals. For all other cells, the actual voltage drop is significantly smaller than V_{read}. Since the MVM concept relies on the applied voltage V_{read} to drop across each cell in the array, a significant error can thus arise due to the IR drop.

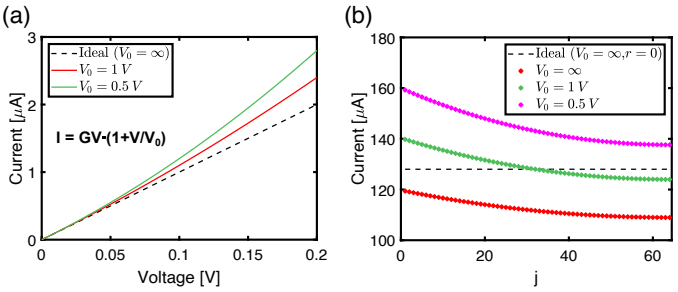


Fig. 5. Impact of non-linearity of the I-V curve of the device. (a) I-V characteristic of non-linear cells for various parameter $\boldsymbol{V_0}$, with $\boldsymbol{G = 10\ \mu S}$. (b) Calculated column current in a worst-case scenario for increasing non-linearity of memory cells.

A. Impact of rG product, array size and non-linearity

Eqs. (1)-(4) indicate a major dependence of V and V' on the product rG. The roles of r and G are comparable, since an increase of G causes a linear increase in the current, while an increase of r causes a linear increase of voltage drop $r \times I$. To study the impact of $r \times G$, Fig. 4a shows the calculated V_Δ for a 32×32 array at increasing $rG = 2 \times 10^{-5}$, 5×10^{-5} and 1×10^{-4}. As rG increases, the difference between V_Δ and the ideal value $V_{read} = 0.2\ V$ increases, thus causing an increasing error in the resulting column current, as shown in Fig. 4b. Fig. 4c shows the maximum IR drop in V_Δ as a function of the array size N for increasing rG. The error increases non-linearly with N, because of the combination of the increase of wire current and the increase of wire resistance. Fig. 4d shows the calculated column current for increasing array size, which further highlights the strong impact of the array size N on MVM errors. These results indicate that the memory devices need to be optimized for IMC, *e.g.*, by minimizing G via suitable device engineering such as a metal oxide channel in electrochemical random access memory (ECRAM) [20]. Also, a tiled architecture of IMC systems is recommendable to minimize N [21].

Depending on the electrical transport mechanism in the memory device, the I-V characteristic can be non-linear in the adopted operating range. Non-ohmic effects may origin from trap-assisted tunneling [22] or Poole-frenkel conduction [23], particularly at relatively low conductance. Fig. 5 considers the case of memory cells with characteristic curve $I = GV \cdot (1 + V/V_0)$, where V_0 is a characteristic voltage to account for the non-linear behavior. Fig. 5b shows the calculated column current for various values of V_0, and assuming a worst-case scenario with uniform LRS states and applied voltage. As V_0 decreases, the overall current increases due to non-linearity, while the relative swing of the current along the columns increases due to the increased IR drop.

B. Impact of data density

Both input signals, represented by the applied V, and the synaptic parameters, stored as device G, can have a lower density compared to the worst-case scenario considered so far. To address the impact of data density, Fig. 6 shows representative data patterns in a binary domain (a), calculated V and V' (b), and column current I (c) for a data density

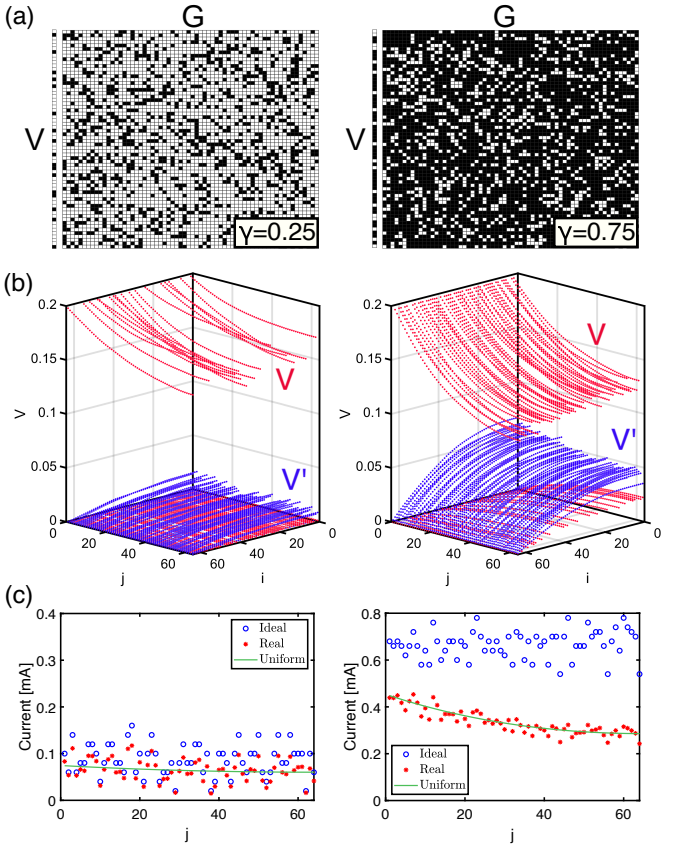


Fig. 6. Data pattern (a) and calculated $\boldsymbol{V}$ and $\boldsymbol{V'}$ (b) for data density of $\mathbf{0.25}$ and $\mathbf{0.75}$ in binary domain. The ideal current ($\boldsymbol{r = 0}$), the real current for the specific pattern of $\boldsymbol{G}$ and $\boldsymbol{V}$ with density $\boldsymbol{\gamma}$, and the current calculated for uniform $\boldsymbol{G = \gamma_G G_{LRS}}$ and $\boldsymbol{V = \gamma_V V_{read}}$ are shown in (c).

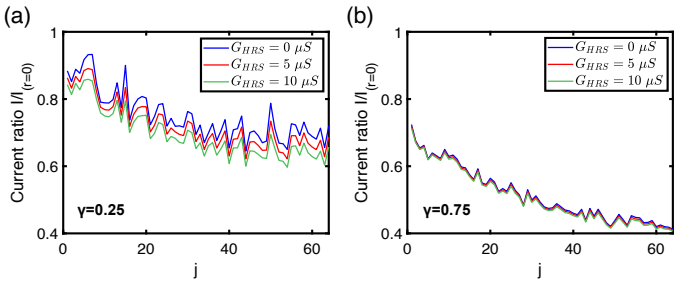


Fig. 7. Calculated column current divided by the ideal current for $\boldsymbol{r = 0}$, as a function of the column position for increasing $\boldsymbol{G_{HRS}}$ and assuming $\boldsymbol{\gamma = 0.25}$ (a) and $\boldsymbol{\gamma = 0.75}$ (b).

$\gamma = 25\%$ (left) and 75% (right). The data density γ was defined as the fraction of selected rows with $V = V_{read}$ and the fraction of cells in the LRS, with all other cells having an ideal $G_{HRS} = 0$. The IR drop increases with γ, resulting in larger errors of the resulting currents. Fig. 7 shows the ratio between the calculated currents with and without IR drop $(r = 0)$ for the case of non-zero G_{HRS}. While the impact of IR drop increases with data density, the contribution of a non-zero G_{HRS} is significant only for low-density patterns.

Interestingly, the current for a random pattern of G and V is well described by the current of a uniform pattern $G = \gamma G_{LRS}$ and uniform applied voltage $V = \gamma V_{read}$. In other words, the currents of a random, sparse MVM with density γ (red dots

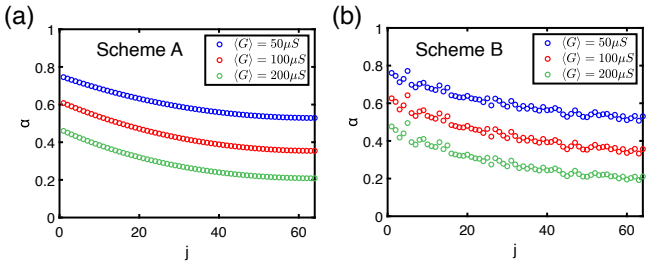


Fig. 8. Compensation vectors $\boldsymbol{\alpha} = \boldsymbol{I/I_{ideal}}$ for increasing $\langle \boldsymbol{G} \rangle$ in multilevel domain. Scheme A assumes a uniform pattern of $\boldsymbol{G}$ and $\boldsymbol{V}$ (a), while Scheme B considers the specific sparse pattern of $\boldsymbol{G}$ and a uniform pattern of $\boldsymbol{V}$ (b).

in Fig. 6c) are approximately the same as those of a uniform MVM with the parameters V and G reduced by a multiplicative factor γ (lines in Fig. 6c). The analysis can be easily extended also to the multilevel domain by defining γ as the average conductance $\langle G \rangle$ normalized with respect to the conductance programming range.

V. Compensation of IR drop in crosspoint arrays

A. Compensation by algorithm

Given the difference between the ideal current I_{ideal} (with no IR drop) and the actual current I in Fig. 6c, one can correct the measured currents operating a division by a compensation vector $\alpha = I/I_{ideal}$. An accurate correction would require that the exact patterns of V and G are known and that the nodal analysis is carried out for each operation, which might be practically unfeasible. However, a first-order compensation is still possible based on the values of γ, as supported by the agreement between current collected from real and uniform patterns in Fig. 6c, valid in both binary and multilevel domain. Based on this concept, two possible compensation schemes are proposed in this work, which differ for the vector α. In *Scheme A*, the vector $\alpha = I/I_{ideal}$ is calculated assuming uniform $G = \gamma_G G_{LRS} = \langle G \rangle$ and $V = \gamma_V V_{read} = \langle V \rangle$, while in *Scheme B* α is obtained assuming uniform $V = \gamma_V V_{read} = \langle V \rangle$ and the specific pattern G. To illustrate these compensation vectors, Fig. 8 shows the calculated $\alpha = I/I_{ideal}$ for increasing $\langle G \rangle$ respectively in *Scheme A* and *Scheme B* scenarios.

The two schemes share the same simplifying hypothesis, i.e. the assumption of a constant input voltage vector $V = \overline{V}$. Indeed, the impact of IR drop in a generic configuration having average input voltage $\langle V \rangle = \overline{V}$ and average conductance $\langle G \rangle = \overline{G}$ can be approximated and then successfully corrected with an α vector computed as the current error ratio of a configuration with fixed input values $V = \overline{V}$ and $G = \overline{G}$, or even more accurately considering the specific conductance pattern G. The chosen assumption is also essential since it makes α independent of $\overline{V}$ and thus unique for every possible input V applied to the same conductance matrix G. Hence, in all the contexts where the matrix of weights G is fixed or has a fixed $\langle G \rangle$ it is sufficient to compute α once and the scheme will result effective for every input vector V. This is the case of the inference phase of a neural network, which is a natural application of crosspoint arrays because of the massive MVM

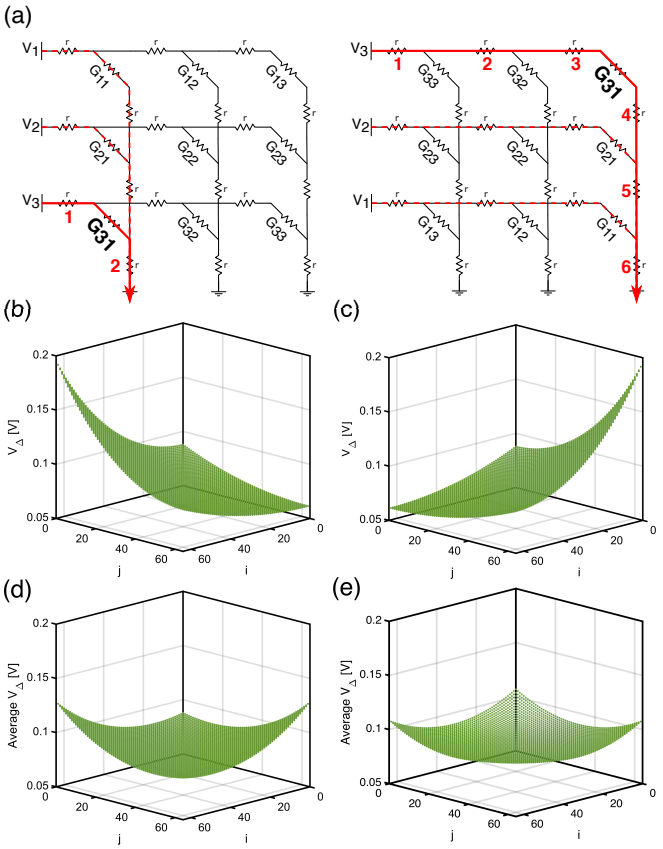


Fig. 9. Original and mirrored structure (a) for performing MVM with crosspoint array, with corresponding $\boldsymbol{V_\Delta}$ curves (b)-(c). In the mirrored structure, memory cells are located specularly with respect to the center and the applied voltage vector is inverted. In this way currents collected at the grounds contain the same informational content of the original structure, while the voltage loss due to IR drop affects more the cells that are less penalised in the original structure. Collecting current outcomes from both architectures, the error results more equally distributed, as suggested by the curve of average $\boldsymbol{V_\Delta}$ (d). With the proper mappings, the concept can be further extended: (e) shows the average $\boldsymbol{V_\Delta}$ in the case of a configuration with four arrays (original, flipped left to right, flipped up to down, mirrored with respect to the centre).

performed. Once trained, a network does not only have a fixed $\langle G \rangle$, but the whole G matrix is fixed and known, which can be used to apply compensation *Scheme B*.

Although compensation techniques including scaling factors have been proposed in literature, the presented schemes differ from prior art in several aspects. Indeed, Schemes A and B are computed once, with low-computational cost and offline, thus without needing for calibration during training [10] or empirical calibration [11]. Also, the schemes can be directly applied during neural networks inference via analogue or digital multiplication of the arrays output current, thus without needing for conductance calibration [12], [13] and using a reduced number of corrective factors.

B. Compensation by architecture

In addition to the algorithm-level schemes, we propose an architectural-level solution capable of strongly mitigating IR drop induced error. The proposed improvement is depicted in Fig. 9. The solution consists in the addition of a second crosspoint array with the conductance matrix flipped left to

right and up to down, thus mirrored with respect to the centre, and with a flipped input voltage vector (Fig. 9a). The second array features the same informational content of the original array, while every memory cell experiences a conductive path with a number of parasitic resistances complementary to the one experienced in the original structure. By averaging the corresponding output currents from the two arrays, one obtains a more equally distributed IR drop error (Fig. 9d). Although this mitigation technique is based on the first-order approximation of same IR drop across each parasitic resistance, its effectiveness is still significant when considering the complete nodal analysis. Indeed, despite the significant voltage drop, the balancing of IR drop results in an operation closer to the ideal MVM, except for a scale factor (Fig. 9d). Since the balance is recovered at the cost of a more uniform voltage loss, the proposed architecture is particularly effective when combined with some form of normalization. This is especially suitable for crosspoint accelerators of neural networks, considering the large diffusion of normalization techniques in Deep Learning. The same concept can be straightforwardly extended to a larger number of arrays, introducing another degree of freedom during design. For instance, one can adopt a 4-array architecture combining the two arrays in Fig. 9a with one array with G flipped left to right and one with both G and V flipped up to down. Fig. 9e shows the resulting average V_Δ for the 4-array configuration, clearly evidencing the improved uniformity of IR drop. Finally, note that the proposed architecture can also mitigate computational errors induced by device conductance variability, thanks to inherent redundancy of the replicated structure [24].

VI. Application in Neural Network Inference

We studied the validity of the proposed compensation techniques by considering a 3-layer fully-connected (FC) neural network for recognizing handwritten digits of the MNIST dataset. The neural network was designed to read 28×28 MNIST digits in gray scale, corresponding to 784 input neurons, followed by 64 hidden neurons and 10 output neurons. We assumed a sigmoidal activation function for each neuron and we trained the weights by backpropagation, using mean square error as loss function. The software accuracy with floating-point (FP) precision of the synaptic weights W was 97%. The resulting synaptic weights were mapped in 7 crosspoint arrays (6 for the first layer, one for the second layer), each having a size of 128×128, for MVM acceleration. Each weight W was mapped in a pair of memory elements with conductance G_+ and G_-, allowing for a differential synapse with weight $W = G_+ - G_-$. Considering a minimum conductance G_0, positive weights W can be mapped with $G_+ = W + G_0$ and $G_- = G_0$, while negative weights W can be mapped with $G_- = -W + G_0$ and $G_+ = G_0$. Ideal FP precision and zero variability were assumed for G_+ and G_-. Fig. 10 shows a sketch of the MVM implementation of the two synaptic layers (a), where G_0 was set equal to 5 μS and the maximum programmed conductance is $G_{MAX} = 300$ μS.

The testing accuracy of the neural network in Fig. 10 was evaluated by computing MVM in each individual tile, by including the effect of the IR drop through the numerical algorithm proposed in Sec.III. In the case of $r = 4$ Ω the resulting inference accuracy was 59%, thus significantly affected by the IR drop during MVM.

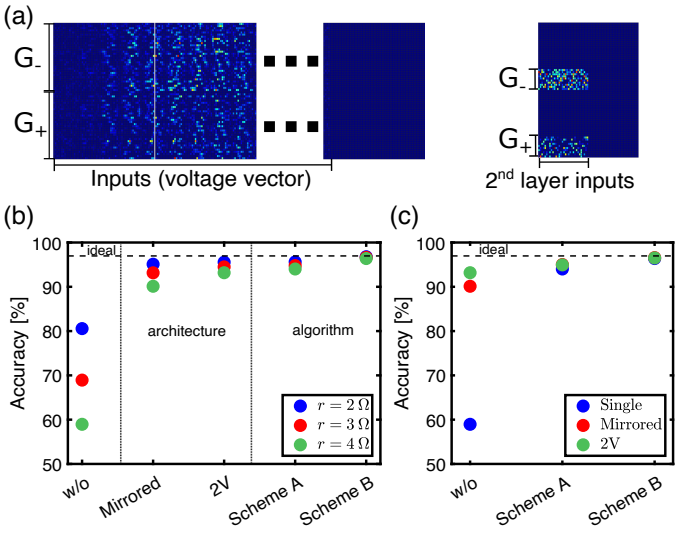


Fig. 10. Sketch of the first and second synaptic layers (a) implemented in $\mathbf{128 \times 128}$ crosspoint arrays. Plot (b) shows test accuracy with and without the compensation techniques applied separately, for various $\boldsymbol{r}$, while plot (c) presents test accuracy when combining compensation by algorithm and by architecture, for $\boldsymbol{r = 4\ \Omega}$ (mirrored architecture refers to the double array configuration shown in Fig. 9a, while 2V refers to doubling the input voltage vector [13], [25]). Accuracy loss due to IR drop is almost fully recovered.

To assess the effectiveness of the proposed solutions, the error was first compensated by applying separately every compensation technique. As a benchmark, we also considered an architectural compensation technique introduced by [25] and adopted in [13], where the voltage input vector was applied at both sides of the crosspoint array by a second vector of voltage generators. This technique, referred to as *2V* in Fig. 10, results in a significant reduction of the IR drop. As summarized in Fig. 10b, an accuracy up to 90.1% could be achieved by architectural compensation with mirrored mapping for $r = 4$ Ω, while an accuracy of 96.4% can be obtained by applying compensation scheme B. As shown in Fig. 10c, the combination of the algorithm-level schemes and the mirrored architecture results in an increase of the accuracy to 96.6%, which is close to the ideal software accuracy.

Compensation schemes A and B can be implemented as analogue or digital post-processing of the sensed current, by dividing it by the compensation vector α. This can be stored in digital format as a look-up table, depending on the known pattern density γ of the weights in the crosspoint array. In particular, IR drop compensation is straightforward for neural network accelerators since the synaptic weight is generally known after training. Scheme A is more general as it is independent of the specific pattern of G, however it is less accurate than scheme B. Again, note that since Eqs. (1)-(2) are linear in V and V', a change in input/activation voltage vector only leads to a linear change in the MVM current, with no impact on the compensation vector α, that can thus be computed only once and offline.

VII. Conclusion

In this work, IR drop in crosspoint arrays is discussed and studied with respect to parameters such as device conductance and linearity, wire resistance, array size and data density. An analytical description of IR drop in the continuous domain is derived for simplified scenarios. A new numerical algorithm for nodal analysis is presented, enabling an improvement of computing time by 5 orders of magnitude with respect to the conventional SPICE approach. Finally, two compensation schemes at the algorithm level and one architectural solution are presented and validated for a 2-layer neural network. In the chosen framework, the proposed compensation techniques can almost fully recover the ideal software accuracy.

References

[1] M. A. Zidan, J. P. Strachan, and W. D. Lu, "The future of electronics based on memristive systems," *Nature electronics*, vol. 1, no. 1, pp. 22–29, Jan 2018, doi:10.1038/s41928-017-0006-8.

[2] D. Ielmini and H.-S. P. Wong, "In-memory computing with resistive switching devices," *Nature Electronics*, vol. 1, no. 6, pp. 333–343, Jun 2018, doi:10.1038/s41928-018-0092-2.

[3] M. Hu, J. P. Strachan, Z. Li, E. M. Grafals, N. Davila, C. Graves, S. Lam, N. Ge, J. J. Yang, and R. S. Williams, "Dot-product engine for neuromorphic computing: Programming 1T1M crossbar to accelerate matrix-vector multiplication," in *2016 53nd ACM/EDAC/IEEE Design Automation Conference (DAC)*. IEEE, Jun 2016, pp. 1–6, doi:10.1145/2897937.2898010.

[4] T. Gokmen and Y. Vlasov, "Acceleration of deep neural network training with resistive cross-point devices: Design considerations," *Frontiers in neuroscience*, vol. 10, no. 333, pp. 1–13, Jul 2016, doi:10.3389/fnins.2016.00333.

[5] D. Ielmini and G. Pedretti, "Device and circuit architectures for In-Memory Computing," *Advanced Intelligent Systems*, vol. 2, no. 2000040, pp. 1–19, Jul 2020, doi:10.1002/aisy.202000040.

[6] M. Rao, Z. Wang, C. Li, H. Jiang, R. Midya, P. Lin, D. Belkin, W. Song, S. Asapu, Q. Xia, and J. J. Yang, "Learning with resistive switching neural networks," in *2019 IEEE International Electron Devices Meeting (IEDM)*. IEEE, Dec 2019, pp. 35.4.1–35.4.4, doi:10.1109/IEDM19573.2019.8993465.

[7] Y. Liao, B. Gao, P. Yao, W. Zhang, J. Tang, H. Wu, and H. Qian, "Diagonal matrix regression layer: Training neural networks on resistive crossbars with interconnect resistance effect," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 40, no. 8, pp. 1662–1671, Aug 2021, doi:10.1109/TCAD.2020.3021309.

[8] I. Chakraborty, D. Roy, and K. Roy, "Technology aware training in memristive neuromorphic systems for nonideal synaptic crossbars," *IEEE Transactions on Emerging Topics in Computational Intelligence*, vol. 2, no. 5, pp. 335–344, Sep 2018, doi:10.1109/TETCI.2018.2829919.

[9] M. E. Fouda, S. Lee, J. Lee, G. H. Kim, F. Kurdahi, and A. M. Eltawi, "IR-QNN framework: An IR drop-aware offline training of quantized crossbar arrays," *IEEE Access*, vol. 8, no. 228392, pp. 1–17, Dec 2020, doi:10.1109/ACCESS.2020.3044652.

[10] S. Jain and A. Raghunathan, "Cxdnn: Hardware-software compensation methods for deep neural networks on resistive crossbar systems," *ACM Transactions on Embedded Computing Systems (TECS)*, vol. 18, no. 6, pp. 1–23, Nov 2019, doi:10.1145/3362035.

[11] Y. Jeong, M. A. Zidan, and W. D. Lu, "Parasitic effect analysis in memristor-array-based neuromorphic systems," *IEEE Transactions on Nanotechnology*, vol. 17, no. 1, pp. 184–193, Dec 2018, doi:10.1109/TNANO.2017.2784364.

[12] Y. K. Lee, J. W. Jeon, E.-S. Park, C. Yoo, W. Kim, M. Ha, and C. S. Hwang, "Matrix mapping on crossbar memory arrays with resistive interconnects and its use in in-memory compression of biosignals," *Micromachines*, vol. 10, no. 5, p. 306, May 2019, doi:10.3390/mi10050306.

[13] 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," *Journal of Low Power Electronics and Applications*, vol. 11, no. 1, p. 9, Feb 2021, doi:10.3390/jlpea11010009.

[14] F. Zhang and M. Hu, "Mitigate parasitic resistance in resistive crossbar-based convolutional neural networks," *ACM Journal on Emerging Technologies in Computing Systems (JETC)*, vol. 16, no. 3, pp. 1–20, May 2020, doi:10.1145/3371277.

[15] S. Cosemans, B. Verhoef, J. Doevenspeck, I. Papistas, F. Catthoor, P. Debacker, A. Mallik, and D. Verkest, "Towards 10000TOPS/W DNN inference with analog in-memory computing–a circuit blueprint, device options and requirements," in *2019 IEEE International Electron Devices Meeting (IEDM)*. IEEE, Dec 2019, pp. 22.2.1–22.2.4, doi:10.1109/IEDM19573.2019.8993599.

[16] A. Chen, "A comprehensive crossbar array model with solutions for line resistance and nonlinear device characteristics," *IEEE Transactions on Electron Devices*, vol. 60, no. 4, pp. 1318–1326, Mar 2013, doi:10.1109/TED.2013.2246791.

[17] *International Technology Roadmap for Semiconductors (ITRS)*, [Online]. Available: http://www.itrs2.net/2013-itrs.html.

[18] V. Milo, C. Zambelli, P. Olivo, E. Pérez, M. K. Mahadevaiah, O. G. Ossorio, C. Wenger, and D. Ielmini, "Multilevel HfO2-based RRAM devices for low-power neuromorphic networks," *APL Materials*, vol. 7, no. 081120, pp. 1–10, Aug 2019, doi:10.1063/1.5108650.

[19] P. Zheng and D. Gall, "The anisotropic size effect of the electrical resistivity of metal thin films: Tungsten," *Journal of Applied Physics*, vol. 122, no. 135301, pp. 1–12, Oct 2017, doi:10.1063/1.5004118.

[20] J. Tang, D. Bishop, S. Kim, M. Copel, T. Gokmen, T. Todorov, S. Shin, K.-T. Lee, P. Solomon, K. Chan, S. Shin, K. Lee, P. Solomon, K. Chan, W. Haensch, and J. Rozen, "ECRAM as scalable synaptic cell for high-speed, low-power neuromorphic computing," in *2018 IEEE International Electron Devices Meeting (IEDM)*. IEEE, Dec 2018, pp. 13.1.1–13.1.4, doi:10.1109/IEDM.2018.8614551.

[21] Q. Wang, X. Wang, S. H. Lee, F.-H. Meng, and W. D. Lu, "A deep neural network accelerator based on tiled RRAM architecture," in *2019 IEEE International Electron Devices Meeting (IEDM)*. IEEE, Dec 2019, pp. 14.4.1–14.4.4, doi:10.1109/IEDM19573.2019.8993641.

[22] G. Bersuker, D. Gilmer, D. Veksler, P. Kirsch, L. Vandelli, A. Padovani, L. Larcher, K. McKenna, A. Shluger, V. Iglesias, M. Porti, and M. Nafria, "Metal oxide resistive memory switching mechanism based on conductive filament properties," *Journal of Applied Physics*, vol. 110, no. 12, p. 124518, Dec 2011, doi:10.1063/1.3671565.

[23] D. Ielmini, F. Nardi, and C. Cagli, "Physical models of size-dependent nanofilament formation and rupture in NiO resistive switching memories," *Nanotechnology*, vol. 22, no. 25, p. 254022, May 2011, doi:10.1088/0957-4484/22/25/254022.

[24] G. Pedretti, P. Mannocci, C. Li, Z. Sun, J. P. Strachan, and D. Ielmini, "Redundancy and analog slicing for precise in-memory machine learning—part I: Programming techniques," *IEEE Transactions on Electron Devices*, vol. 68, no. 9, pp. 4373–4378, Sep 2021, doi:10.1109/TED.2021.3095433.

[25] J. Mustafa and R. Waser, "A novel reference scheme for reading passive resistive crossbar memories," *IEEE transactions on nanotechnology*, vol. 5, no. 6, pp. 687–691, Nov 2006, doi:10.1109/TNANO.2006.885016.