Intrinsic Bounds for Computing Precision in Memristor-Based Vector-by-Matrix Multipliers

M. R. Mahmoodi*, A. F. Vincent*#, H. Nili, and D. B. Strukov

***Abstract*— Analog computing with crossbars of memristors is a promising approach to build compact energy-efficient vector-by-matrix multiplier (VMM), a key block in many data-intensive algorithms. However, device non-linearity, process variations, interconnect parasitics, noise, and memory state drift limit the computing precision of such systems. In this paper, we investigate the impact of such non-idealities in analog current-mode memristive VMMs through simulations and experiments on the most prospective passive crossbars. We show that there is an optimal tuning voltage to minimize the computation error. Furthermore, error balancing and bootstrapping are introduced as two techniques for improving the precision. It is also shown that when size of $N\times N$ crossbar is scaled up, the optimum interconnect wire conductance should increase quadratically with N to preserve the computing precision when using naive error balancing approach, and that the differential scheme is imperative for temperature insensitive operation and also to reduce the IR-drop effect.**

***Index Terms*— ReRAM, Analog Computing, Computing Precision, Vector-by-Matrix Multiplier, Artificial Neural Network**

I. INTRODUCTION

The emergence of new promising nonvolatile memory technologies [1] has renewed interest in analog and mixed-signal computing, especially for VMM, which is broadly used in data-intensive algorithms. This work is focused on hardware implementations based on resistive switching devices (also known as ReRAM or memristors). By coupling experimental work on crossbars of metal-oxide memristors (Fig. 1a) with circuit-level simulations, this paper investigates the impact of device and circuit imperfections (such as device nonlinearity and variations, line resistance), and crossbar topology on the computing precision of analog VMM circuits and provides insights into their possible improvements. To make our analytical study more general, we target two applications of VMM circuits: an analog mutli-layer perceptron implementation and 2D convolution for edge detection filtering. Furthermore, due to the so-far limited write endurance of memristors, we assume infrequent conductance tuning, which makes the write operation and memory selector issues of a less concern.

This paper was submitted for review on October 8, 2019.
All authors are with the Department of Electrical and Computer Engineering, University of California at Santa Barbara, Santa Barbara, CA, USA, 93106. E-mail: mrmahmoodi@ ucsb.edu. *These authors contributed equally. #Present address: Laboratoire de l'Intégration du Matériau au Système, Bordeaux, France.

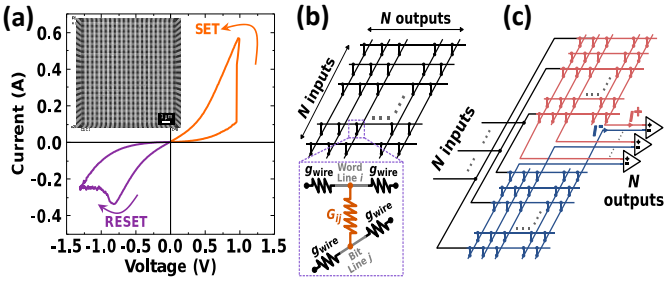

Fig. 1. (a) Experimental *I-V* curve of a TiO_2 memristor. Inset shows scanning electron microscopy image of a 20×20 passive crossbar circuit. (b) Single-quadrant and (c) differential $N\times N$ crossbar circuit with memristive crosspoint devices.

Several works in the literature addressed similar topics, but with a focus on devices with selectors [2-4], write operation [5], or programming error due to variations [6]. Nonlinear input mapping is proposed in [7] to compensate device nonlinearity. However, variations and interconnect parasitics are neglected in that study. Only linear devices and the impact of IR drop are considered in [8]. Passive crossbar circuits are explored in [9] using a simplified model focusing only on nonlinearity and IR drop. Similarly, Ref. [23] proposes a conversion algorithm to minimize the error due to device non-idealities and IR drop. Finally, some works explored the impact of non-idealities in the context of specific applications, such as for neuromorphic computing [10, 11], which is typically much more resilient to precision errors, or digital memory [12].

II. MODELING METHODOLOGY

We model $N\times N$ crossbar circuits with passively integrated TiO_{2-x} memristors (Fig. 1), based on the technology developed by our group [13, 15, 16, 19]. Our general focus is on circuit and device parameters, e.g. values of N, wire conductance g_{wire}, and memristor currents, specific to our technology. However, the considered TiO_{2-x} devices have rather typical *I-V* characteristics of many metal-oxide memristors and since we also extend our analysis to larger g_{wire} values,

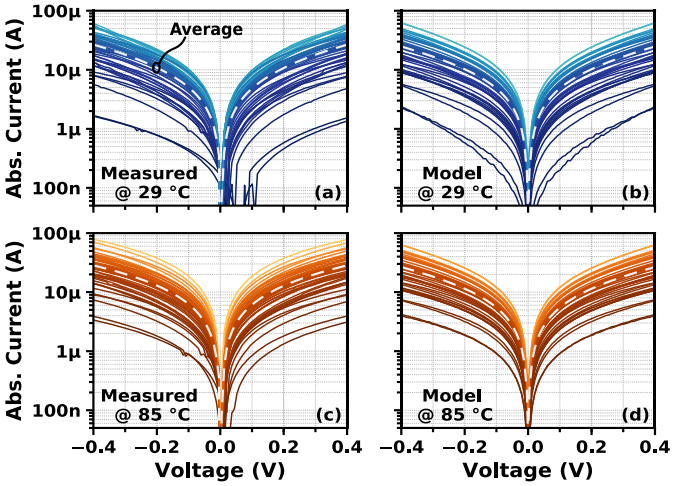

Fig. 2. Example of (a, c) experimentally measured, and (b, d) fitted models of static *I-V* characteristics for 50 devices, studied at two ambient temperatures. The developed model accurately predicts device currents within range of non-disturbing voltages, based on the single current measurement at 0.1 V.

Table I. Notations (top) and related equations (bottom).

N	Linear dimensions of crossbar array
B	Number of connections used for bootstrapping per crossbar line
$U_{\max}$	Largest (non-disturbing) voltage applied to the crossbar circuit inputs during operation
$kU_{\max}$	Voltage at which device conductance is tuned
g_{wire}	Crossbar line (full-pitch- long segment) conductance (Fig. 1b)
G_{ij}	Crosspoint device conductance (Fig. 1b)
$G_{\min}, G_{\max}$	Minimum, maximum value of G_{ij}
$I^{(n)}$	n-th element of the output current vector
$I^{(n)}_{\text{ideal}}$	Ideal value for $I^{(n)}$, i.e. for linear devices and $g_{\text{wire}}\to\infty$
$\lvert I\rvert_{\max}$	Maximum absolute current value
ε	Relative (computing) current error – see below
$\lvert\varepsilon\rvert_{99.9\%}$	99.9% percentile of $\lvert\varepsilon\rvert$, also referred as worst-case error
W_{ij}	Weight value, assumed in [0, 1] and [-1, 1] ranges for single-quadrant and differential topologies, respectively
Single-quadrant	$G_{ij} = G_{\min} + W_{ij}(G_{\max} - G_{\min})$
	$\varepsilon = (I^{(n)} - I^{(n)}_{\text{ideal}}) \,/\, \lvert I\rvert_{\max}$
Differential	$G_{ij} = G^{+}_{ij} - G^{-}_{ij}$ $G^{\pm}_{ij} = G_{\min} + (1 \pm W_{ij}) \times (G_{\max} - G_{\min})/2$
	$\varepsilon = (I^{(n)} - I^{(n)}_{\text{ideal}}) \,/\, (2\,\lvert I\rvert_{\max})$

representative of more advanced fabrication processes (e.g., with larger aspect ratio crossbar electrodes made of higher conductance metals, compared to those used in our circuits) we believe that the results of this paper are quite general.

Simulation results are obtained from SPICE using an in-house phenomenological compact model of TiO_2 memristors [14,25]. As opposed to previous work, this compact model is developed based on an extensive statistical study and captures the device imperfections needed for a comprehensive analysis of memristive circuits and systems. The studied non-idealities include nonlinearities in static *I-V* characteristics, device-to-device variations, noise, and temperature dependency (e.g., see Fig. 2).

We consider conventional memristor-based VMM topology in which data are encoded in voltage amplitude of signals. Compared to time-based encoding, the voltage-mode approach is fully compatible with the most prospective passive ("0T1R") technology and is potentially more promising, in terms of throughput and energy efficiency, particularly in medium precision regimes – see, e.g., our previous works on system level analysis [21-22].

We study both single-quadrant (Fig. 1b) and differential (Fig. 1c) topologies and take into account parasitic wire resistance of a memristive crossbar circuits. For each crossbar size N, we randomly generated 512 input voltage vectors, with vector elements uniformly distributed in the range [0, $U_{\max}$] - see Table I for the definition of all parameters used in this study. In addition, 512 crossbar circuits are randomly generated, each with unique device-to-device (d2d) variations and crosspoint conductances (at 0.1 V), uniformly distributed in the range [$G_{\min}$, $G_{\max}$]. The crosspoint conductances were obtained indirectly by first generating dimensionless weights and then converting them according to the VMM topology (Table I). The VMM errors were then calculated for all combinations of input vectors and crossbar circuits, with a total of 256k configurations for each N. Most of the results are reported in terms $|\varepsilon|_{99.9\%}$, which is 99.9% percentile of output current errors $|\varepsilon|$ for each studied crossbar size N, where $|\varepsilon|$ is the absolute difference between ideal and the actual output current, normalized to its maximum value.

III. Computing Precision Analysis

A. Parasitics, device nonlinearity, and process variations

In our first study, we consider g_{wire} = 0.4 S, which corresponds to the measured line conductance in 20×20 crossbar circuits [15,16] (but smaller compared to other recent works [2]). Fig. 3 shows histograms of the simulated output currents for each output of a 16×16 VMM circuit, for different topologies, biasing strategies, and temperatures. The single-quadrant architecture severely suffers from the voltage drop on interconnect parasitics. Biasing the word lines from both sides can help to mitigate the shift and the spread of the errors with respect to the output index (Fig. 3a, b) at the temperature (25 °C) used during programming. (A more general and powerful biasing approach is further discussed in Section IV.) Nevertheless, it does not reduce the sensitivity to the

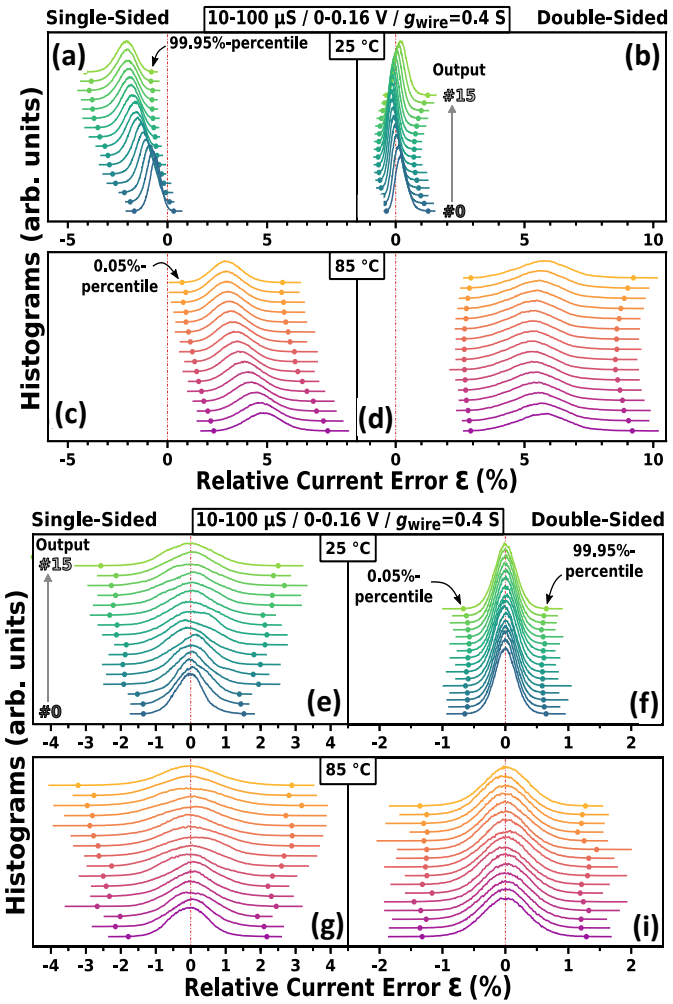

Fig. 3. Simulated results for output current error ε for (a-d) single-quadrant and (e-i) differential topologies implementing 16×16 VMM. Output #0 (#0 and #15) is the closest to where the input voltages are applied to for single-quadrant (differential) topology. Panels (a, c, e, g) are for single-sided voltage biasing, while (b, d, f, i) are for double-sided ones. Top and bottom panels are for 25 ˚C and 85 ˚C, respectively. For all cases, G_{min} = 10 µS, G_{max} = 100 µS, and U_{max} = 0.16 V.

temperature (Fig. 3c, d). The differential scheme (Fig. 3e-i), however, centers and narrows down the distribution of output errors around zero. This is because the effective conductance of the device is (almost) a monotonic function of temperature. Hence, the temperature related terms cancel out in the differential topology. Combined with two-sided voltage biasing, this topology is particularly appealing to keep the error as low as possible. Despite conservative choice of g_{wire}, $|\varepsilon|_{99.9\%}$ is still below 1.5% and 0.75%, for 85 ˚C and 25 ˚C, respectively (Fig. 3f, i).

For a more practical case of larger crossbar circuits, the finite g_{wire} can cause significant voltage drops, reducing the effective voltage drop on the crosspoint device (Fig. 4a). At smaller g_{wire}, the $|\varepsilon|_{99.9\%}$ is roughly inversely proportional to the square of g_{wire}, which is consistent with

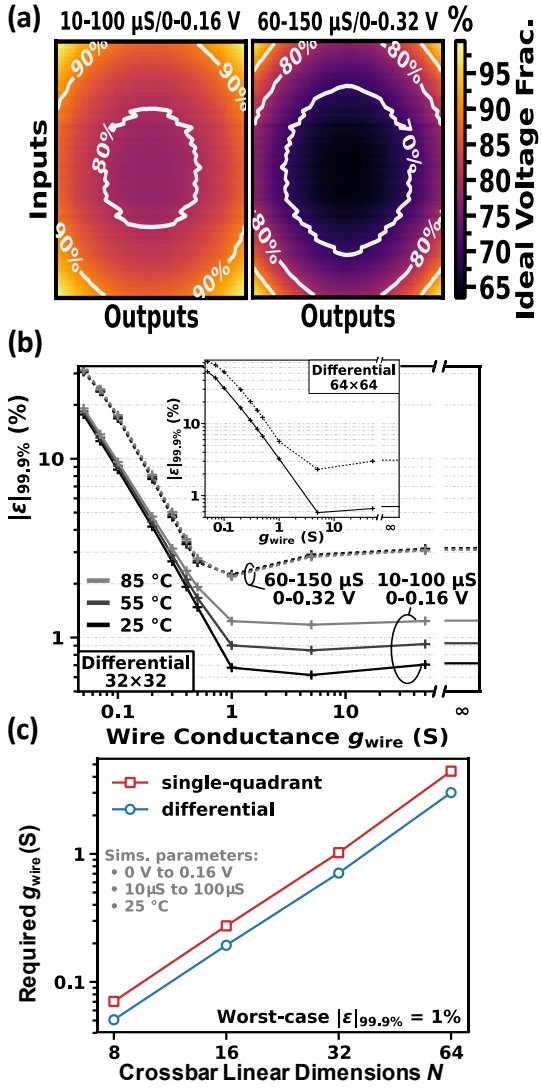

Fig. 4. (a) Heatmap of normalized (by U_{max}) average voltages across crosspoint devices in a 64×64 crossbar circuit with g_{wire} = 0.4 S and double-sided voltage biasing, simulated at room temperature (25 ˚C), for two different values of U_{max} and ranges of device conductances. (b) Simulated worst-case error as a function of g_{wire} for differential 32×32 circuit. Inset shows the results for 64×64 differential crossbar circuit simulated at room temperature. (c) Extrapolated wire conductance g_{wire}, which is required to ensure the 1% worst-case error.

the worst-case error due to the IR drops on the crossbar lines. At larger g_{wire}, $|\varepsilon|_{99.9\%}$ is leveling, independent of the voltage and the conductance ranges (Fig. 4b). In fact, $|\varepsilon|_{99.9\%}$ slightly increases before plateauing, due to excessive currents injected by nonlinear devices at higher biases, which compensates the current deficiency created by IR drops on the crossbar lines. In addition, intrinsic device characteristics, e.g., the higher temperature sensitivity in lower conductance ranges also have a significant impact on the error.

Furthermore, the error plateau is lower for smaller N (Fig. 4b inset). Assuming g_{wire} > 1 S similar to [2], Fig. 4b

results predict $|\varepsilon|_{99.9\%} < 1\%$ at 25 ˚C, even for large, >1K-cell crossbar circuit. Unsurprisingly, g_{wire} should be increased quadratically with N to ensure error below 1% (Fig. 4c). Also, the differential topology is naturally more immune to IR-drop because its impact on the currents through both devices is compensated by a differential pair.

For relatively small crossbar circuits with smaller voltage drops, device nonlinearity becomes the main precision-limiting factor, while for larger crossbar circuits, the dominant factor is interconnect parasitics (Fig. 5). On the other hand, device-to-device variations do not have much impact on the error, at least for the considered applications. This is because write-verify tuning algorithm allows to accurately program the device conductance at read voltage despite variations in *I-V* characteristics. Furthermore, the impact of device-to-device variations reduce even further for large crossbars due to averaging.

B. Noise, state drift, and peripheral circuit considerations

In a current-mode VMM circuit, the total noise in the differential output currents (e.g., $I^{\pm}$ in Fig. 1c) is equal to the sum of the noises of the corresponding crosspoint memristors, which share the same output crossbar line. Current fluctuations through different devices are independent and the signal-to-noise ratio (SNR) of the output current, assuming maximum current output range, is proportional to $\sqrt{N}$. For high-speed operation, e.g. >200 MHz, the noise spectrum is predominately white, and neglecting the input-referred current noise of the peripherals (which is often well below that of crossbar [17]), SNR > 35 dB for 16×16 crossbar circuit, which is equivalent to >5 bits of precision, >50 dB (> 8 bit) for N = 64, and increases further with N.

For most applications, it is imperative that crosspoint devices retain their conductance state over a long period of time. Otherwise, frequent retuning would be required, and/or the loss of accuracy due to the memory state drift should be considered. Accelerated retention measurements at 85 °C over 15h for in-house 325 crossbar-integrated TiO_2 devices showed < 0.8% average change in conductance, with < 2.7% standard deviation (Fig. 6). Such conductance drift can be safely neglected, by performing infrequent retuning, e.g., every 6 months.

We should also note that the design of precise peripheral circuits is rather straightforward. For example, a buffered transimpedance amplifier can be utilized to supply the current for the crossbar array, while maintaining the impedance matching conditions on the crossbar lines. Such amplifiers can be designed with near ideal transfer characteristics, and hence won't degrade

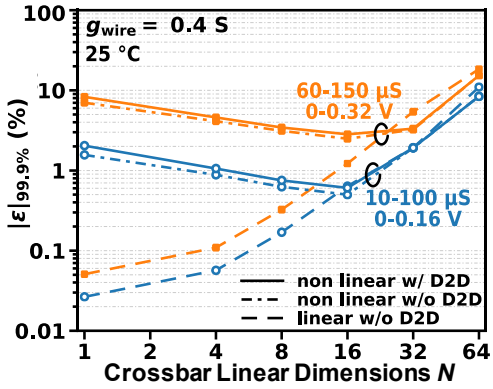

Fig. 5. Impact of *I-V* static nonlinearity and d2d variations on the worst-case error, for differential topology.

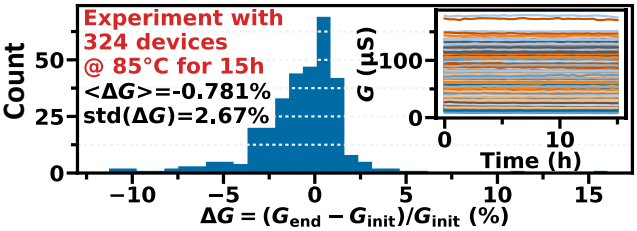

Fig. 6. Experimental results for memory state retention, showing change in device conductance, measured at 0.1 V, after baking for 15 h at 85˚C.

computing accuracy. Its area and energy overhead, especially for crossbar circuits with high conductance crosspoint devices, could be a concern. However, the peripheral overhead can be greatly reduced by taking advantage of tunability of analog-grade memory cells, e.g. to compensate offsets due to process variations in the sensing and driving circuitry, as suggested in [18].

IV. Techniques for Improving Precision

The computing error due to nonlinearity in the static *I-V* characteristics can be reduced by optimizing procedure of mapping weights to the memory states of crosspoint devices. In case of ideal, linear devices, the slope of *I-V* curve, i.e. the memory state, is typically assessed by measuring device current at the highest operating voltage (U_{max}), when using write-verify tuning algorithm [13]. For the devices with nonlinear *I-V* curves, such approach leads to negligible error at U_{max} voltage input, but the error might be significant at voltages below U_{max} due to smaller effective conductance. To reduce such error, the crosspoint devices can be tuned at smaller voltages kU_{max}, where $0 < k \leq 1$, i.e. by setting device's effective conductance at $I(kU_{max})/(kU_{max})$ to the desired value at the tuning algorithm. In this case, the error would be the smallest at kU_{max} and is more balanced between the larger and smaller ranges of the input voltages (Fig. 7a inset).

For the in-house devices, G = 100 µS, and the distribution of the inputs assumed in the modeling

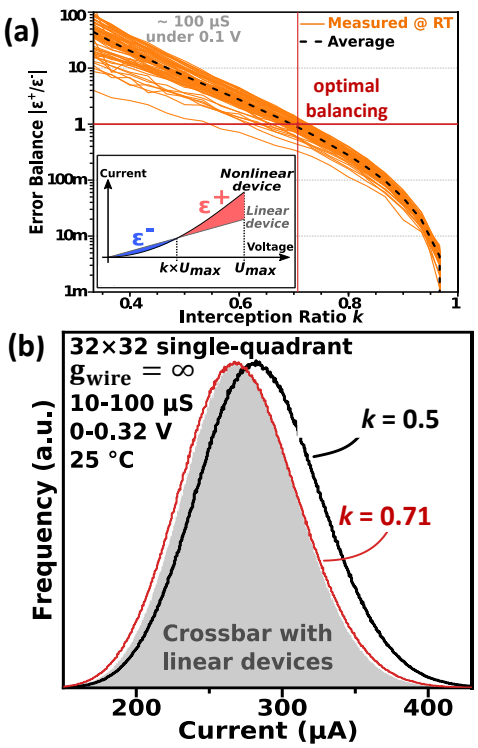

Fig. 7. (a) Ratio between positive and negative error integrals, calculated from the measured *I-V* characteristics of TiO_2 devices which were tuned to have $I(kU_{max})/(kU_{max})$ = 100 µS at 27°C, as a function of used k value. Inset shows schematically definition of positive (red) and negative (blues) current error integrals for single devices. (b) Simulated output currents in a 32×32 single-quadrant architecture with nonlinear crosspoint devices, when tuned at k = 0.5 (black line) and k = 0.71 (red line), and ideal linear devices (grey-filled area).

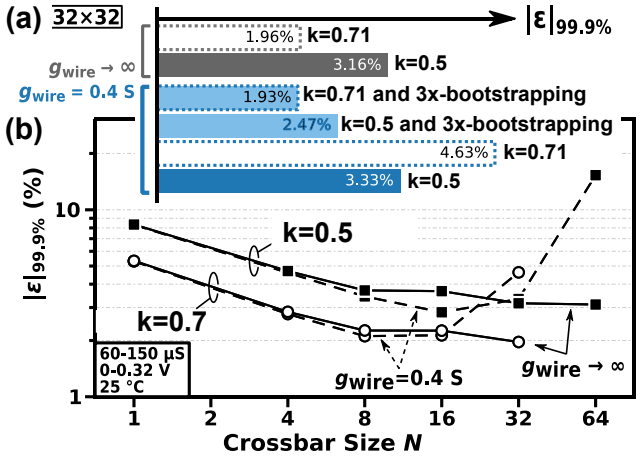

Fig. 8. Worst-case error for (a) 32×32 crossbar circuit and (b) as a function of crossbar linear size, when using various techniques for improving precision and two assumptions of wire conductance. Wire resistance of the spare lines is neglected in the analysis of the bootstrapped circuits.

(Section II), the computing error is minimized by using $k \approx \text{sqrt}(2)/2 \approx 0.71$ (Fig. 7a). Fig. 7b shows a comparison of VMM output currents from crossbar circuits with ideal (linear) devices, and those with nonlinear devices for two cases of k. As expected from Fig. 7a results, the currents are higher for k = 0.5 due to larger error integral at higher input voltages. The distribution of currents for the case of nonlinear devices is almost perfectly matching the ideal one when using optimal k. This is because individual current errors of single devices (i.e., those in computed product terms) are canceling each other out when device currents are added up on the crossbar lines.

The error balancing technique works well when voltage drops on the crossbar lines are not significant. However, the error can be even larger otherwise compared to suboptimal balancing approach, e.g., for 32×32 circuit with g_{wire} = 0.4 S (Fig. 8). This is because IR drops across crossbar lines compensate higher currents for the suboptimal balancing (i.e., right shift of the currents in the histogram in Fig. 7b). One solution to deal with large IR drops is to compute optimal values of k based on the particular device location in the crossbar, e.g. by combining the balancing technique with the one described in Ref. 23.

An orthogonal solution is to employ a bootstrapping technique, e.g., similar to the one used in NOR flash memory circuits. In a bootstrapped design, all crossbar lines are backed up with spare lines, which, e.g., can be routed in the lower metal layers for back-end-integrated crossbar circuits. Each spare line is connected to the original crossbar line in $B > 1$ locations (denoted as "B×-bootstrapping"), which are equally distributed along the length of the line. For example, B = 2 implies that the original and spare lines are connected at the edges of the crossbar, i.e., corresponds to the already mentioned double-sided architecture. For 3×- bootstrapping, there are three connections - one in the middle and two at ends of the line, etc.

Bootstrapping technique significantly improves the computing precision (Fig. 8a), while comes at the typically acceptable cost of utilizing additional metal layers below and/or above crossbar array. For passive memristor technology, bootstrapping also requires increasing crossbar dimensions from N to N+B-2 to accommodate connections inside the crossbar array, though such overhead is minor for the most practical cases $N >> B$.

V. Application Demonstrations

The proposed techniques for improving precision are further verified by modeling two representative applications of mixed-signal current-mode VMM circuits. The first studied application is an edge detection with 5×5 Laplacian of Gaussian filter, in which convolution of an image with a specific filter is computed to extract high frequency information or image edges (Fig. 9). The image

convolution operation was modeled assuming differential architecture with 25 inputs and 1 output for the specific image (Fig. 9a) using a hybrid approach. In particular, 50 devices in a 20×20 crossbar circuit were tuned to the desired values corresponding to the kernel weights (Fig. 9b), at the voltages specific to the used k, and their static I-V characteristics collected. The data were then fitted using approach discussed in [25] and used for simulating dot-product currents. 8 different implementations, with different k, B, and g_{wire} are studied (Fig. 9e inset), including ideal case scenario, i.e. with $g_{wire} = \infty$ and linear I-V characteristics. Fig. 9c shows an example of filtered image assuming scenario D, i.e., using measured I-V characteristics, $k = 0.5$, $B = 2$, and $g_{wire} = 0.4$ S.

The results show that due to smaller parasitics, the crosspoint device nonlinearity is a major source of computing error, see, e.g., scenario A vs. B (Fig. 9d, e). This is why balancing technique is the most useful for this application. Indeed, among the considered nonlinear device scenarios B, D, F, G, H, the error is smaller for scenarios F, G, H. On the other hand, bootstrapping does not help and can actually increase error (e.g. E cf. B, and F cf. H). This is due to already mentioned compensation of IR drops across crossbar. Even $k = 0.71$ is apparently not optimal (and hence H has smaller error than F) for this particular application because of different distributions of conductances and inputs as compared those used in Figs. 7 and 8.

The second studied application is neuromoprhic inference of MNIST benchmark images using 784-64-10 multilayer perceptron classifier with rectified linear activation (Fig. 10). The first layer is modeled by assuming that 24 64×64 and 2 17×64 crossbar circuits are connected in two 785×64 virtual crossbars to realize differential architecture, while the second layer is modeled with two 65×10 crossbar circuits. (The additional inputs is due to the bias.) The other hyperparameters and ex-situ training approach (with 60k / 20k training / test images) are similar to [20].

The inference is simulated using memristor compact model which accounts for d2d variations in I-V characteristics [25], and also assuming that input voltages for physical crossbar circuits are applied individually (i.e. that $N \leq 64$). The computing error in the first MLP layer (error in the output currents), and the corresponding classification errors are shown in Fig. 10c and d, respectively, for several scenarios (Fig. 10e). The results show that, unlike for previously studied application, the impact of IR drops on the performance is more severe compared to device nonlinearity (test 2 cf. tests 1 and 3). This is due to smaller devices' conductances (i.e. large number of small weights as shown in Fig. 10b) as well as

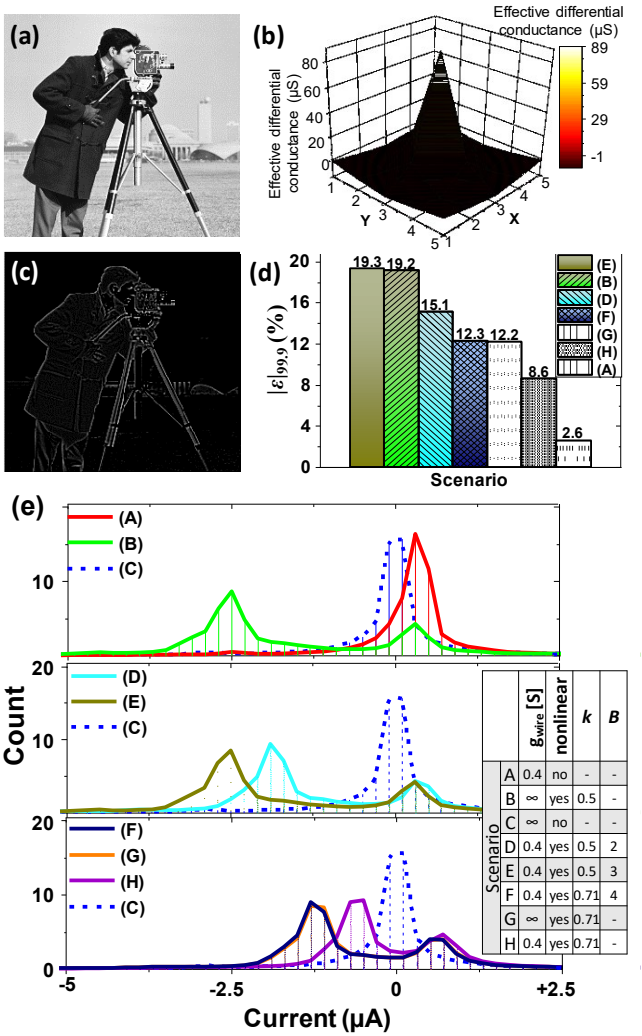

Fig. 9. Modeling of edge detection algorithm using 5×5 Laplacian of Gaussian filter assuming differential implementation based on two 25×1 memristive crossbar circuits and taking into account device's I-V nonlinearity and d2d variations: (a) Original image, (b) effective conductance of a differential pair used to implement a 5×5 filter. X and Y are filter dimensions. (c) simulation results for the computed image assuming 2×bootstrapping and $g_{wire} = 0.4$ S. (d) The worst-case error and (e) output current histograms for several considered scenarios. The details for each studied scenario are provided in the inset of panel e. $T = 25$°C, $U_{max} = 0.16$ V.

larger crossbar circuits. Both VMM error and the classification accuracy improves by increasing the crossbar line conductance (tests 4, 6, 7, 8, 9) and/or number of bootstrapping connections (tests 6, 10, 11). Similar to previous application, a small non-zero wire resistance could be beneficial for compensating current overshoot (test 3 cf. test 9). The results also show that the error is the largest for the single sided architecture (test 4) for which only half of the crossbar circuits were employed in modeling, while a combination of more optimal balancing and aggressive boostrapping leads to the classification performance of 2.09%. This number is close to the best-case 2%, obtained by simulating the same

MLP network in a software using high precision arithmetics – see, e.g. test 13 cf. test 1.

VI. Summary

We have developed a framework for circuit-level simulations of memristive crossbar circuits and utilized comprehensive device models as well as experimentally measured data for metal-oxide memristors to investigate the impact of various imperfections on the computing precision of analog memristor-based VMM circuits. Using statistical numerical simulations, we quantified the impact of interconnect parasitics and analyzed different topologies on the precision under range of temperatures. Finally, error balancing and bootstrapping techniques were proposed to mitigate device and circuit imperfections, which are further verified by modeling two representative applications.

VII. Acknowledgements

This work was supported via a Semiconductor Research Corporation (SRC) funded JUMP CRISP center, and by NSF/SRC E2CDA grant 1740352.

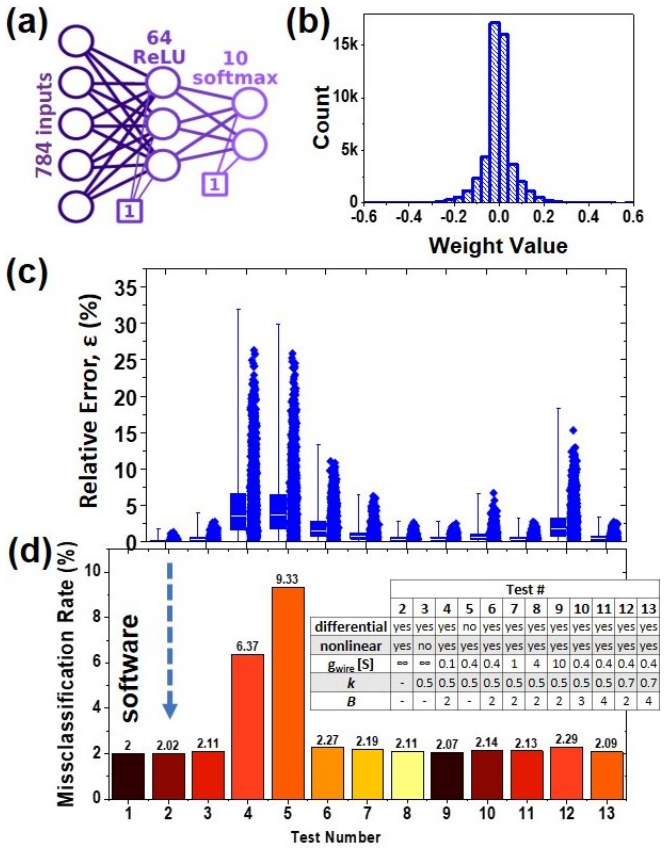

Fig. 10. Modeling of image classification inference with multilayer perceptron network: (a) Studied network. (b) Histogram of VMM weights for classification of MNIST benchmark images, obtained using ex-situ training method. (c) Simulated error in the output currents of the 1st layer VMM circuits and (d) corresponding misclassification errors for several studied scenarios (tests). The details for each studied scenario are provided in the inset of panel d.

References

[1] H.S.P. Wong and S. Salahuddin, "Memory leads the way to better computing", *Nat. Nanotechnol.*, vol. 10, pp. 191-194, 2015.

[2] C. Li *et al*., "Analogue signal and image processing with large memristor crossbars", *Nat. Electron.*, vol. 1, pp. 52–59, 2018.

[3] G. W. Burr *et al*., "Experimental demonstration and tolerancing of a large-scale neural network using phase-change memory as the synaptic weight element", *IEEE Trans. Electron Devices*, vol. 62, pp. 3498–3507, 2015.

[4] H. Li *et al*., "Resistive RAM-centric computing: Design and modeling methodology", *IEEE Trans. Circuits Syst. I*, vol. 64, pp. 2263–2273, 2017.

[5] S. Agarwal *et. al.*, "Compensating for parasitic voltage drops in resistive memory arrays", in: *Proc. IMW'17*, Monterey, CA, May 2017, pp. 1-4.

[6] P. Sheridan *et al.*, "Sparse coding with memristor networks," *Nat. Nanotechnol.*, vol. 12, pp.784–789, 2017.

[7] T. Kim *et. al.* "Input voltage mapping optimized for resistive memory-based deep neural network hardware", *IEEE Electron Device Lett.,* vol. 38, pp.1228-1231, 2017.

[8] Y. Jeong, M. Zidan, and W. Lu, "Parasitic effect analysis in memristor-array-based neuromorphic systems", *IEEE Trans. Nanotechnol.,* vol. 17 pp.184-193, 2018

[9] L. Xia *et al*., "Technological exploration of RRAM crossbar array for matrix-vector multiplication," *J. Comp. Sci. Technol*., vol. 31, pp. 3-19, 2016.

[10] B. Yan *et al*., "Understanding the trade-offs of device, circuit and application in ReRAM-based neuromorphic computing systems," in: *Proc. IEDM'17*, San Francisco, CA, Dec. 2017, pp. 11.4.1 – 11.4.4.

[11] F. Merrikh Bayat *et al.*, "Memristor-based perceptron classifier: Increasing complexity and coping with imperfect hardware," in: *Proc. ICCAD'17*, Irvine, CA, Nov. 2017, pp. 549–554.

[12] A. Chen, "A comprehensive crossbar array model with solutions for line resistance and nonlinear device characteristics", *IEEE Trans. Electron Devices*, vol. 60, pp. 1318–1326, 2013.

[13] F. Alibart *et al*., "High precision tuning of state for memristive devices by adaptable variation-tolerant algorithm," *Nanotechnology*, vol. 23, art. 075201, 2012.

[14] F. Merrikh Bayat, B. Hoskins, and D.B. Strukov, "Phenomenological modeling of memristive devices", *Appl. Phys. A*, vol. 118 (3), pp. 770-786, 2015.

[15] M. Prezioso *et al*., "Modeling and implementation of firing-rate neuromorphic-network classifiers with bilayer Pt/Al_2O_3/TiO_{2-x}/Pt memristors", in: *Proc. IEDM'15*, Washington, DC, Dec. 2015, pp. 17.4.1 – 17.4.4.

[16] H. Kim, H. Nili, M. Mahmoodi, and D. Strukov, "4K-memristor analog-grade passive crossbar circuit", ArXiv: 1906.12045, 2019.

[17] M.R. Mahmoodi and D.B. Strukov, "An ultra-low energy internally analog, externally digital vector-matrix multiplier based on NOR flash memory technology," in: *Proc. DAC'18*, San Francisco, CA, June 2018, art. 22.

[18] M.R. Mahmoodi and D.B. Strukov, "Breaking POp/J barrier with analog multiplier circuits based on nonvolatile memories", in: *Proc. ISLPED'18*, Bellevue, WA, July 2018, art. 39.

[19] F. Merrikh-Bayat *et al*., "Implementation of multilayer perceptron network with highly uniform passive memristive crossbar circuits", *Nat. Commun.,* vol. 9, art. 2331, 2018.

[20] X. Guo *et al*., "Fast, energy-efficient, robust, and reproducible mixed-signal neuromorphic classifier based on embedded NOR flash memory technology", in: *Proc. IEDM'17*, San Francisco, CA, Dec. 2017, pp. 6.5.1-6.5.4.

[21] M. Bavandpour, S. Sahay, M.R. Mahmoodi, and D.B. Strukov, "Mixed-signal neuromorphic processors: Quo vadis?", in: *Proc. IEEE S3S'19*, San Jose, CA, Dec. 2019, pp. 1-2.

[22] M. Bavandpour *et al.,* "Mixed-signal neuromorphic inference accelerators: Recent results and future prospects", in: *Proc. IEDM'17*, San Francisco, CA, Dec. 2018, pp. 20.4.1-20.4.4.

[23] M. Hu *et al*., "Dot-product engine for neuromorphic computing: Programming 1T1M crossbar to accelerate matrix-vector multiplication", in: *Proc. DAC'16*, San Jose, CA, June 2016, pp. 1-6.

[24] M. Hu *et al.,* "Double bias memristive dot product engine for vector processing", U.S. Patent No. 10,109,348, 23 Oct. 2018.

[25] H. Nili *et al.,* "Comprehensive compact model of integrated metal-oxide memristors", *IEEE Trans. Nanotechnology*, 2020 (in print).