A Low-bit And Data-conversion-free Memristive Spiking Computing Network

Guanchao Qiao and Shaogang Hu*

School of Electronic Science and Engineering, University of Electronic Science and Technology of China, No. 4, Section 2, Jianshe North Road, Chengdu, China
*Email: sghu@uestc.edu.cn

Abstract. Memristor crossbar holds promising potential in massively parallel data processing, as computing tasks beyond vector-matrix multiplication typically requires additional complex data conversion modules. In this work, we report a memristive spiking computing network (MSCN) without data conversion. Two memristor crossbars are used to perform vector-matrix multiplication, while spiking neurons are used to integrate the signals from each column of the memristor crossbars. Input and output signals are binary spikes (spiking or non-spiking); thus data conversions between the analog and digital domains are avoided. The MSCN is suitable for accelerating artificial neural network (ANN). A multi-layer perception (MLP) consisting of four layers is simulated based on the analog MSCN, and recognition accuracy of 98.55% under the MNIST test set is achieved. However, in the practical memristor, it is challenging to adjust conductance analogously and even challenging to implement multi-bit conductance states. Therefore, the memristor conductance states are further reduced to 3 bits and 1 bit. Encouragingly, recognition accuracies just slightly decrease to 98.36% and 97.96% for 3-bit and 1-bit conductance states, respectively, which is cost-effective considering the hardware reliability improvement.

1. Introduction

For the physical separation of information processing and storage, conventional computing architectures suffer from the “Von-Neumann bottleneck” [1], which brings challenges including the “memory wall” and the “heat wall.” Recently, the computing in-memory (CiM) concept [2-4], which intends to integrate computing and memory together at the device level, has attracted much attention for its great potential to break the “Von-Neumann bottleneck.” Traditional memory devices such as DRAM [5], SRAM [6-8], and floating gate memory [9, 10], as well as emerging nonvolatile memory devices such as STT-RAM [11, 12] and memristor are reconsidered to be used as CiM cells. Among various memory devices, memristor has been widely studied for its simple structure, high integration density, low working voltage, fast speed, excellent scalability [13-15]. Moreover, a single memristor can realize memory and computing simultaneously, making it be one of the most promising candidates for future CiM application [16-18].

Memristor crossbar structure integrates the CiM function of memristor with the parallelism of the crossbar, and it is suitable for vector-matrix multiplication (or dot-product) [18-21]. Normally, information from the memristor crossbar needs to be further processed. Memristor crossbars process information in the analog domain, while other modules for further information processing usually work in the digital domain, making it inevitable to use analog-to-digital converter (ADC) and digital-to-analog (DAC) to convert data between the digital and analog domains [22-27]. Furthermore, to
realize global parallel information processing, each column (or row) requires a DAC at the input port and an ADC at the output port, making the CiM structure complex, expensive, high-power consumption, and difficult to scale up.

In this work, we propose a DAC/ADC-less memristive spiking computing network (MSCN) using memristor crossbars for vector-matrix multiplication and compact spiking neurons for the following information integration. A multi-layer perception (MLP) simulated based on an analog MSCN (MSCN with analog conductance state) achieves recognition accuracy of 98.55% under the MNIST test set. Furthermore, even the memristor reduces to 3- and 1-bit conductance states, the recognition accuracies still remain at 98.36% and 97.96%, respectively.

2. Proposed Architecture
For memory application, information is usually stored as resistance or conductance (G) state of a memristor. For example, for digital memory application, logic “0” is represented by a high-resistance state (HRS), while logic “1” is represented by a low-resistance state (LRS). Recently, it has been reported that the conductance of memristors can be adjusted analogously, which can be used for analog memory [28]. Meanwhile, the current-voltage (I-V) characteristic of memristor obeys the Ohm’s law, making it suitable for implementing multiplication \(i = v \times g\) if its input is in a voltage form \(v\) and output is in a current form \(i\) [26]. And thus, memristor holds the capabilities of information memory and processing simultaneously. Besides, the memristor crossbar further integrates the massive parallelism of the crossbar and can implement vector-matrix multiplication in parallel with high bandwidth.

Figure 1. (a) The architecture of MSCN. (b) The structure of the spiking neuron circuit.
Figure 1(a) shows the diagram of the proposed MSCN. Since the conductance of memristor cannot be negative in physics, two memristor crossbars are required to store a matrix consisting of positive and negative elements [21]. One memristor crossbar (MC+) consisting of \(m\) columns and \(n\) rows is used to process the nonnegative elements of the matrix, and the other one (MC-) of the same size is used to handle the negative elements of the matrix. Information from the MC+ and MC- need to be integrated by integration units. If using artificial neurons with nonlinear activation function (i.e., sigmoid activation function) as the integration units, the inputs to the memristor crossbars and the outputs of the artificial neurons are analog signals. Since the other information processing modules are always work in the digital domain, it is inevitable to avoid column-level ADC and DAC for data conversion, which makes the system complex and is hardware expensive. In the proposed MSCN, information is conveyed and processed in the form of binary spike similar to the brain, and bio-plausible spiking neurons are used as the fundamental information processing units to integrate signals from each column of the MC+ and MC-.

In the subtractor, there are two operational amplifiers whose outputs are given by:

\[
v_{o1} = -r_{f1} \times \frac{v_{in}}{r_1}
\]

\[
v_{o2} = -r_{f2} \times \left( \frac{v_{in} + v_{o1}}{r_2} \right) = -r_{f2} \times v_{in} \times \left( \frac{1}{r_2} - \frac{r_{f1}}{r_3} \times \frac{1}{r_1} \right)
\]

where \(v_{o1}\) and \(v_{o2}\) are the output of the first and second operational amplifiers, respectively. \(1/r_1\) and \(1/r_2\) can be regarded as the equivalent conductance seeing from a particular column of MC- (\(g_- = 1/r_1\)) and the corresponding column of MC+ (\(g_+ = 1/r_2\)), respectively. For the sake of simplicity, it is assumed that \(r_1 = r_3\):

\[
v_{o2} = -r_{f2} \times v_{in} \times (g_+ - g_-)
\]

The activity of the integrator circuit is governed by:

\[
v_{o2} = -c_1 \times \frac{dv_o}{dt}
\]

where \(v_o\) is the voltage across the capacitor \(c_1\). Unfolding Eq. (4) with the Euler’s rule (regarding \(dt\) as one time step for the convenience of the simulation) and substituting Eq. (2), we have:

\[
v_o(t + 1) = v_o(t) - \frac{v_{o2}}{r_4 \times c_1} = v_o(t) + r_{f2} \times v_{in} \times (g_+ - g_-)
\]

Assuming \(r_2 = r_4\), we have:

\[
v_o(t + 1) = v_o(t) + \frac{v_{in} \times (g_+ - g_-)}{c_1}
\]

Finally, the comparator determines the activation state of the output:

\[
v_{out} = \begin{cases} 1 & \text{if } v_o \geq v_{th} \\ 0 & \text{if } v_o < v_{th} \end{cases}
\]

where \(v_{th}\) is the threshold voltage of the spiking neuron. When \(v_o\) reaches \(v_{th}\), \(v_{out}\) will be at a high voltage level, and the switched-transistor \((s)\) will be turned on to discharge rapidly; and when \(v_o\) is discharged to below \(v_{th}\), \(v_{out}\) will decrease to a low voltage level. And thus, a spike is formed at the output port of the spiking neuron.

3. Results and Discussions
To validate the feasibility of the proposed MSCN, an MLP based on the MSCN was simulated with Matlab. The MLP consists of an input layer, two hidden layers, and an output layer, and the number of neurons in each layer is 784, 1000, 1000 and 10, respectively. The MLP was trained with the backpropagation (BP) algorithm under the MNIST training set. To map the weights to the memristor crossbar in the follow-up processes, two tricks were taken during the training process. Firstly, all neurons used rectified linear unit (ReLU) as their activation functions for two reasons: the output of ReLU can be approximated as the firing rate of an IF neuron [29], and the ReLU allows the weights to be scaled by a factor to meet the conductance features of fabricated memristors. Secondly, the biases

\[\text{ReLU}(x) = \begin{cases} x & \text{if } x > 0 \\ 0 & \text{otherwise} \end{cases}\]

are backpropagated to the memristors in the subtractor.
of all neurons were set to zero. In this way, the performance of the MLP is mainly determined by the ratio of weights to thresholds, which effectively reduces the difficulties in parameter adjustment. During training, the ten neurons in the output layer were labeled as 0 to 9 in turn.

After training, classification on the MNIST test set was carried out in the MSCN. The weights of each layer of MLP were directly mapped to two memristor crossbars, and the spiking neuron model deduced from Eqs. (1-7) were used, as shown in Figure 2. It is worth noting that the inputs ($V_{in}^m$) to the MSCN were rate-coded as Poisson spike trains whose frequencies were proportional to the inputs (real values) fed to the MLP. The output of each spiking neuron was collected, and the number of spikes of a spiking neuron fired during the time window was used to determine the classification result. The most active spiking neuron fired the most spikes, and its label was regarded as the recognition result.

Figure 2. Schematic of the implementation of handwritten-digit recognition on MSCN.

![Figure 2](image)

Figure 3. (a) Recognition accuracy as a function of the time step. (b) Recognition accuracy as a function of noise ratio.

Considering the actual memristor implementation, original weights are scaled up before mapping to the realistic memristor’s conductance. Specifically, for the memristor crossbar MC+, the weights those larger than $10^{-4}$ are multiplied with $10^{-4}$, and other weights are set to $10^{-7}$. On the contrary, for the memristor crossbar MC-, if a weight is lower than $-10^{-4}$, it multiplies with $+10^{-4}$; otherwise, it is set to $10^{-7}$. Therefore, the conductance of memristors in MSCN ranges from $10^{-7}$ S to $10^{+4}$ S that is practically achievable. The trained MLP achieves recognition accuracy of 98.54% under the MNIST test set. The analog MSCN (with analog conductance state) shows little degrade after mapping, and
the recognition accuracy remains at 98.55% for conductance ratio larger or equal to 300, as shown in Figure 3(a).

Memristor’s conductance may vary in a neuromorphic hardware system due to the variation of the process, voltage, temperature, and so on. Therefore, it is meaningful to evaluate the performance of MSCN when added noise. Figure 3(b) gives the recognition accuracies after adding Gaussian noise with different standard deviations (corresponding to different levels of noise) to the synaptic weights and memristor conductances of MLP, analog MSCN, 3-bit MSCN, and 1-bit MSCN, respectively. The accuracies of MLP and MSCN remain higher than 90% when the noise ratio is up to 45%, and the impact of 1-bit MSCN due to noise is much smaller than the other three. This means MSCN demonstrates great robustness of conductance variation.

Furthermore, an image with 2062×3093 pixels is processed with MSCN. The image is convolved with a 3×3 sharpness filter \(
\begin{bmatrix}
-2 & -2 & -2 \\
-2 & 18 & -2 \\
-2 & -2 & -2 \\
\end{bmatrix}
\) and an edge-detection filter \(
\begin{bmatrix}
-1/2 & -1/2 & -1/2 \\
-1/2 & 4 & -1/2 \\
-1/2 & -1/2 & -1/2 \\
\end{bmatrix}
\), as shown in Figure 4. MSCN can successfully realize different convolution operations proves its potential to map the popular convolutional neural networks and implement more difficult tasks, which is our ongoing work.

**Figure 4.** Image processing with MLP and MSCN.
4. Conclusion
In summary, an MSCN without data conversion is reported in this work. Two memristor crossbars are used to implement vector-matrix multiplication, and spiking neurons are used to integrate information from each column of the memristor crossbars. Little degrade in recognition accuracy on the MNIST test set is observed in the analog MSCN. Although recognition accuracies 3-bit and 1-bit MSCNs slightly decrease to 98.36% and 97.96%, respectively, the small degrade is acceptable in most cases especially considering the reliability and achievability brought by the reduction of conductance states. This work provides a reliable and achievable method to implement memristor-based CiM, which may be useful for building ultra-high computing density networks beyond the Von-Neumann architecture.

References
[1] Backus J 1978 Can programming be liberated from the von Neumann style? A functional style and its algebra of programs Commun. ACM 21 613-41
[2] Gokhale M, Holmes B and Iobst K 1995 Processing in memory: the Terasys massively parallel PIM array Computer 28 23-31
[3] Elliott D G, Stumm M, Snellgrove W M, Cojocaru C and Mckenzie R 1999 Computational RAM: implementing processors in memory IEEE Design & Test of Computers 16 32-41
[4] Li B, Yan B and Li H 2019 An Overview of In-memory Processing with Emerging Non-volatile Memory for Data-intensive Applications. In: Proceedings of the 2019 on Great Lakes Symposium on VLSI, (Tysons Corner, VA, USA: ACM) pp 381-6
[5] Seshadri V, Lee D, Mullins T, Hassan H, Boroumand A, Kim J, Kozuch M A, Mutlu O, Gibbons P B and Mowry T C 2017 Ambit: in-memory accelerator for bulk bitwise operations using commodity DRAM technology. In: Proceedings of the 50th Annual IEEE/ACM International Symposium on Microarchitecture (Cambridge, Massachusetts: ACM) pp 273-87
[6] Khwa W, Chen J, Li J, Si X, Yang E, Sun X, Liu R, Chen P, Li Q, Yu S and Chang M 2018 A 65nm 4Kb algorithm-dependent computing-in-memory SRAM unit-macro with 2.3ns and 55.8TOPS/W fully parallel product-sum operation for binary DNN edge processors. In: 2018 IEEE International Solid-State Circuits Conference -(ISSCC), (San Francisco, CA, USA: IEEE) pp 496-8
[7] Agrawal A, Jaiswal A, Lee C and Roy K 2018 X-SRAM: Enabling In-Memory Boolean Computations in CMOS Static Random Access Memories IEEE Trans. Circuits Syst. Regul. Pap. 65 4219-32
[8] Wang J, Wang X, Eckert C, Subramanian A, Das R, Blaauw D and Sylvester D 2019 14.2 A Compute SRAM with Bit-Serial Integer/Floating-Point Operations for Programmable In-Memory Vector Acceleration. In: 2019 IEEE International Solid- State Circuits Conference -(ISSCC), (San Francisco, CA, USA: IEEE) pp 224-6
[9] Merrikh-Bayat F, Guo X, Klachko M, Prezioso M, Likharev K K and Strukov D B 2018 High-Performance Mixed-Signal Neurocomputing With Nanoscale Floating-Gate Memory Cell Arrays IEEE Transactions on Neural Networks and Learning Systems 29 4782-90
[10] Gonugondla S K, Kang M, Kim Y, Helm M, Eilert S and Shanbhag N 2018 Energy-Efficient Deep In-memory Architecture for NAND Flash Memories. In: 2018 IEEE International Symposium on Circuits and Systems (ISCAS), (Florence, Italy: IEEE) pp 1-5
[11] Kang W, Wang H, Wang Z, Zhang Y and Zhao W 2017 In-Memory Processing Paradigm for Bitwise Logic Operations in STT–MRAM IEEE Trans. Magn. 53 1-4
[12] Jain S, Ranjan A, Roy K and Raghunathan A 2018 Computing in Memory With Spin-Transfer Torque Magnetic RAM IEEE Trans. Very Large Scale Integr. VLSI Syst. 26 470-83
[13] Waser R, Dittmann R, Staikov G and Szot K 2009 Redox-based resistive switching memories - nanoionic mechanisms, prospects, and challenges Adv. Mater. 21 2632-63
[14] Yang J J S, Strukov D B and Stewart D R 2013 Memristive devices for computing Nat. Nanotechnol. 8 13-24
[15] Yu S and Chen P 2016 Emerging Memory Technologies: Recent Trends and Prospects IEEE Solid-State Circuits Mag. 8 43-56

[16] Haron A, Yu J, Nane R, Taouil M, Hamdioui S and Bertels K 2016 Parallel matrix multiplication on memristor-based computation-in-memory architecture. In: 2016 International Conference on High Performance Computing & Simulation (HPCS), (Innsbruck, Austria: IEEE) pp 759-66

[17] Chen W, Lin W, Lai L, Li S, Hsu C, Lin H, Lee H, Su J, Xie Y, Sheu S and Chang M 2017 A 16Mb dual-mode ReRAM macro with sub-14ns computing-in-memory and memory functions enabled by self-write termination scheme. In: 2017 IEEE International Electron Devices Meeting (IEDM), (San Francisco, CA, USA: IEEE) pp 28.2.1-2.4

[18] Le Gallo M, Sebastian A, Mathis R, Manica M, Giefers H, Tuma T, Bekas C, Curioni A and Eleftheriou E 2018 Mixed-precision in-memory computing Nature Electronics 1 246-53

[19] Chakrabarti B, Lastras-Montaño M A, Adam G, Prezioso M, Hoskins B, Payvand M, Madhavan A, Ghofrani A, Theogarajan L, Cheng K T and Strukov D B 2017 A multiply-add engine with monolithically integrated 3D memristor crossbar/CMOS hybrid circuit Scientific Reports 7 42429

[20] Chen W, Li K, Lin W, Hsu K, Li P, Yang C, Xue C, Yang E, Chen Y, Chang Y, Hsu T, King Y, Lin C, Liu R, Hsieh C, Tang K and Chang M 2018 A 65nm 1Mb nonvolatile computing-in-memory ReRAM macro with sub-16ns multiply-and-accumulate for binary DNN AI edge processors. In: 2018 IEEE International Solid-State Circuits Conference - (ISSCC), (San Francisco, CA, USA: IEEE) pp 494-6

[21] Li C, Hu M, Li Y, Jiang H, Ge N, Montgomery E, Zhang J, Song W, Dávila N, Graves C E, Li Z, Strachan J P, Lin P, Wang Z, Barnell M, Wu Q, Williams R S, Yang J and Xia Q 2018 Analogue signal and image processing with large memristor crossbars Nature Electronics 1 52-9

[22] Haensch W 2018 Analog Computing for Deep Learning: Algorithms, Materials & Architectures. In: 2018 IEEE International Electron Devices Meeting (IEDM), (San Francisco, CA, USA: IEEE) pp 3.2.1-3.2.4

[23] Merkel C, Hasan R, Soures N, Kudithipudi D, Taha T, Agarwal S and Marinella M 2016 Neuromemristive Systems: Boosting Efficiency through Brain-Inspired Computing Computer 49 56-64

[24] Roth R M 2019 Fault-Tolerant Dot-Product Engines IEEE Trans. Inf. Theory 65 2046-57

[25] Ankit A, Haji I E, Chalamalasetti S R, Ndu G, Foltin M, Williams R S, Faraboschi P, Hwu W-m W, Strachan J P, Roy K and Milojicic D S 2019 PUMA: A Programmable Ultra-efficient Memristor-based Accelerator for Machine Learning Inference. In: Proceedings of the Twenty-Fourth International Conference on Architectural Support for Programming Languages and Operating Systems, (Providence, RI, USA: ACM) pp 715-31

[26] Hu M, Strachan J P, Li Z and Grafals E M 2016 Dot-product engine for neuromorphic computing: Programming 1T1M crossbar to accelerate matrix-vector multiplication. In: 53nd ACM/EDAC/IEEE Design Automation Conference (DAC), (Austin, TX, USA: IEEE) pp 1-6

[27] Yu H, Yu H, Yu H and Yu H 2017 Distributed In-Memory Computing on Binary RRAM Crossbar Acm Journal on Emerging Technologies in Computing Systems 13 36

[28] Seo K, Kim I, Jung S, Jo M, Park S, Park J, Shin J, Biju K P, Kong J, Lee K, Lee B and Hwang H 2011 Analog memory and spike-timing-dependent plasticity characteristics of a nanoscale titanium oxide bilayer resistive switching device Nanotechnology 22 254023

[29] Diehl P U, Neil D, Binas J, Cook M, Liu S and Pfeiffer M 2015 Fast-classifying, high-accuracy spiking deep networks through weight and threshold balancing. In: 2015 International Joint Conference on Neural Networks (IJCNN), (Killarney, Ireland: IEEE) pp 1-8