Learning Accuracy Analysis of Memristor-based Nonlinear Computing Module on Long Short-term Memory

Hongyu An
The Bradley Department of Electrical and Computer Engineering, Virginia Tech, Blacksburg, 24060, USA
hongyu51@vt.edu

Mohammad Shah Al-Mamun
The Bradley Department of Electrical and Computer Engineering, Virginia Tech, Blacksburg, 24060, USA
samamun@vt.edu

Marius K. Orlowski
The Bradley Department of Electrical and Computer Engineering, Virginia Tech, Blacksburg, 24060, USA
marius@vt.edu

Yang Yi
The Bradley Department of Electrical and Computer Engineering, Virginia Tech, Blacksburg, 24060, USA
cindy_yangyi@vt.edu

ABSTRACT

To accelerate the training efficiency of neural network-based machine learning, a memristor-based nonlinear computing module is designed and analyzed. Nonlinear computing operation is widely needed in neuromorphic computing and deep learning. The proposed nonlinear computing module can potentially realize a monotonic nonlinear function by successively placing memristors in a series combing with a simple amplifier. The proposed module is evaluated and optimized through the Long Short-term Memory with the digit number recognition application. The proposed nonlinear computing module can reduce the chip area from microscale to nanoscale, and potentially enhance the computing efficiency to $O(1)$ while guaranteeing accuracy. Furthermore, the impact of the resistance variation of memristor switching on the training accuracy is simulated and analyzed using Long Short-term Memory as a benchmark.

KEYWORDS

LSTM, Memristor, Nonlinear activation functions, learning accuracy, memristor resistance variation

ICONS '18, July 23–26, 2018, Knoxville, TN, USA

ACM ISBN 978-1-4503-6544-4/18/07 $15.00
https://doi.org/10.1145/3229884.3229889

This work was supported by the National Science Foundation under Grant CCF-1750450

1. INTRODUCTION

Long Short-term Memory [1] (LSTM) is one of the most important Recurrent Neural Networks (RNNs) architecture which has various applications on neural languages processing [2], machine translation [3], and speech recognition [4]. In LSTM, the gradient vanishing problem is effectively avoided by replacing the simple hidden layer unit of RNNs with a complex pre-designed memory cell with the "Gates". These "Gates" are implemented mathematically with extra nonlinear functions and weight matrices. Through updating the weights of the matrixes during the training process, LSTM has the capability to control the data flow within the layers, consequently to avoid the vanishing issue.

However, the complex memory cell structure and extra control gates inevitably increase the computing workloads when updating the weight matrix values, performing the nonlinear activation function operations, and calculating the matrix products in the training processes [1]. This increase of computing complexity would further escalate the computing hardware efficiency requirements in deep learning and neuromorphic computing.

Several optimization methods have been proposed to enhance the training efficiency [5]. However, these algorithm-based optimization methods cannot fundamentally prevent the drastic increase of the computational complexity due to the inherent limits of data processing strategy of the computing hardware platform. Traditionally, the von Neumann based computing platform encodes the data in a binary scheme, which is tailored intentionally for Boolean and arithmetic calculations with a sequential data processing and transportation strategy for a nonlinear calculation. In general, the computational complexity of a nonlinear function would be $O(n^2)$ through numerical methods [6], where n is the input data size. In order to enhance the machine learning training efficiency, innovations of computing architectures and their hardware implementations are required.

In this paper, we address these limitations of the computation bottleneck by proposing and designing a novel memristor-based nonlinear activation function hardware. Through the physical implementation of the nonlinear computing, the nonlinear activation functions can be calculated potentially with $O(1)$ computational complexity due to the elimination of numerical methods which require large operation cycles [6-8]. Compare to existing hardware nonlinear activation implementations [9, 10], the proposed design utilizes emerging device memristor that can reduce the chip area to nanoscale and lead to a lower power consumption [11]. Furthermore, by careful designing the set voltage, high resistance state (HRS), and low resistance state (LRS) of the memristors in the nonlinear module, the proposed memristor-based nonlinear module potentially can implement a monotonic nonlinear function.

The proposed nonlinear computing module is evaluated with an application of digit number recognition based on LSTM. The simulation results indicate that the proposed nonlinear computing module with no significant accuracy compromise can complete the training task. Furthermore, the resistance variation influence of memristors in the modules on training accuracy is investigated and analyzed.

2. THE SWITCHING BEHAVIOR OF THE CASCADED MEMRISTORS SWITCHING

The nonlinear activation function is realized through operations on cascaded memristors [12]. The memristor is generally fabricated in a metal-insulator-metal (MIM) configuration with the metals on top and bottom serving as terminals. If the applied stimulus signal on these terminals excesses a specific threshold (set voltage), the resistance of the memristor can be switched from high resistance state (HRS) to low resistance state (LRS). This switching behavior is due to the formation of the conductive filaments (CFs) in the insulator material between two terminals.

Not like other conventional memristor configurations with only one memristor like 1T1R or 2T1R, the proposed nonlinear computing module needs to perform switching behavior in a cascaded memristor configuration. To experimentally verify the switching behavior in this cascaded configuration, we conducted an I-V characteristics test of a metal-insulator-metal (MIM) device structure arranged in a crossbar array. In this crossbar structure, Copper (Cu) is used as a top metal electrode, oxygen-deficient tantalum oxide (TaOx) as solid electrolyte and Rhodium (Rh) as an inert bottom electrode with a thin Chromium (Cr) layer as a glue layer between Rhodium and the subjacent oxidized silicon wafer. The device was fabricated by the Micro and Nanofabrication Laboratory of The Bradley Department of Electrical and Computer Engineering at Virginia Tech [13].

The device has been characterized by monitoring the forming voltage (Vform) when conductive filaments (CFs) are being formed initially, the reset voltage, Vreset, the set voltage, Vset, and the resistance of the filament Ron when CFs are formed. The applied set conditions are: Icc=50 μA and the voltage ramp rate rr= 2.0V/s, reset conditions are: Icc=0.1 A, with voltage ramp rate rr= 0.2 V/s. During measurement of filament resistance, the test conditions are Icc=5 μA with voltage ramp rate rr= 10 V/s and this test condition ensures that the filament is not ruptured/modified by the reading conditions[14].

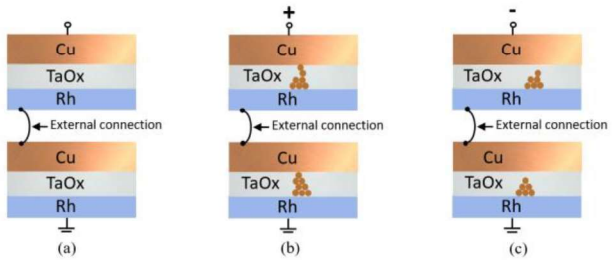

Figure 1: The switching states of the cascaded memristor configuration (two memristors in series): (a) two cascaded memristors without any bias voltage at pristine state; (b) two cascaded memristors at LRS state (set process) (c) two cascaded memristors at HRS state (reset process)

Two devices are connected in series through external flying probe as shown in Figure 1. The diagram of the experimental setup is illustrated in Figure 2. The device combination shows switching behavior with successful set and reset operations as depicted in Figure 3 and Figure 4, respectively. The measurement results and test data are summarized in Table 1.

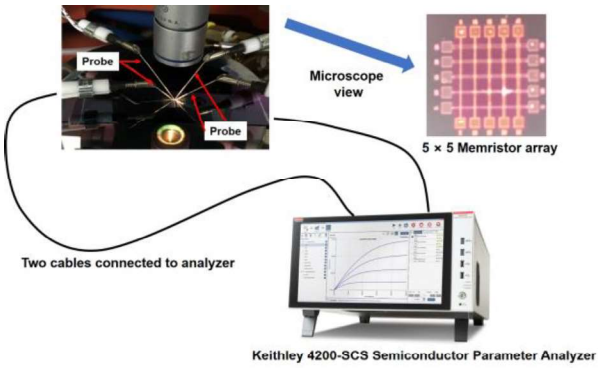

Figure 2: The experimental setup. Two probes are used to apply appropriate bias connecting to semiconductor parameter analyzer, while other two probes provide an external connection for cascaded configuration

Table 1: Measurement results of two serially connected devices

Vform	4 V
Vset	2.85 V
Ron(Top device)	1E+04 Ω
Ron(Overall)	1E+04 Ω
Vreset	-3 V

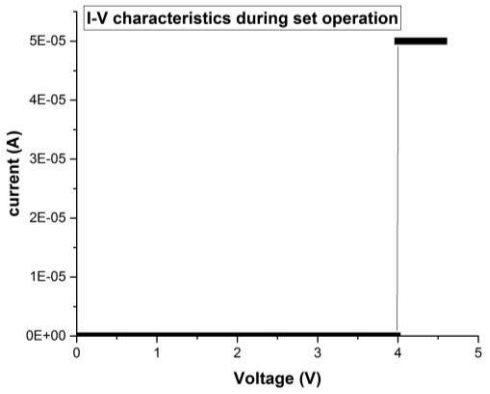

Figure 3: I-V characteristics during set operation (two devices connected serially through external probes)

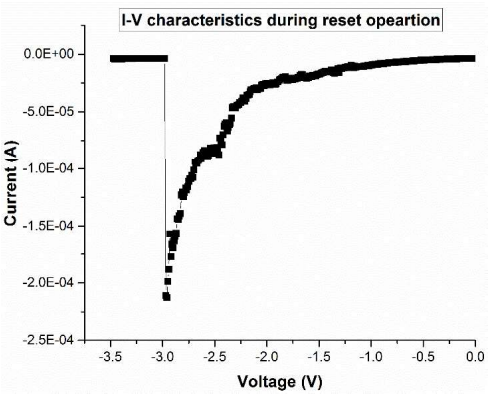

Figure 4: I-V characteristics during a reset operation (two devices connected serially through external probes)

When a positive bias is applied (as shown in figure 1b), the entire applied voltage is distributed across the devices. The capacitive division principle determines the partial voltage drops across the two individual devices. When the voltage is ramped up, more and more Cu atoms of the top device are oxidized into Cu+ ions (Cu→Cu++e), drift towards inert bottom electrode under the influence of high applied electric field. Eventually, Cu+ ions are deposited on the surface of Rh electrode as Cu atoms. Over time Cu atoms stack on top of each other grows vertically and eventually connects the top and bottom electrode and form a conducting filament (CF). Now, the device changes from high resistance state (HRS) to low resistance state (LRS) and the device is in so-called on-state (LRS). As soon as one of the devices moves to LRS, the entire stimulus voltage is applied across the remaining devices which are still on HRS. This relatively high electric field causes the conducting filament formation relatively faster for the subsequent devices illustrated in Figure 1(b). During the experiments of our serially connected devices, this process happened so quickly that we could not distinguish them as separate instances. Now if we ramp up the reversed voltage as connected in Figure 1(c). At the critical voltage Vreset, the current through the devices reaches a critical current (Ireset), the filaments of one or all the devices are ruptured and the device switches from LRS to HRS due to Joules heating. Then the device is restored to an off-state (HRS). In our experiments, the CFs in two memristors were ruptured that has been confirmed through monitoring the resistance of two memristors individually. As more and more devices are connected serially, higher and higher voltage needs to be applied to them since they are distributed across all the devices connected in series.

3. MEMRISTOR-BASED NONLINEAR FUNCTION MODULE DESIGN

The various types of memristors can be formed with different geometries, material combinations, and set/reset voltages [15, 16]. The value of the set voltage highly depends upon the memristor's thickness and material combinations [17, 18]. The low resistance state and large resistance state distribution of state-of-art fabricated memristors are summarized in Figure 5 [19]. The memristor index represents the fabricated memristors. The red left spot indicates the low resistance, meanwhile, the blue right spot represents the high resistance of the memristor. Figure 5 indicates that the values of LRS and HRS of the memristor are the controllable design parameters. It is worthy to note that the resistance variations of the LRS and HRS are mainly dependent by the various materials and different memristor sizes. Integrating multiple materials in memristor fabrication has been demonstrated in several groups[20-22].

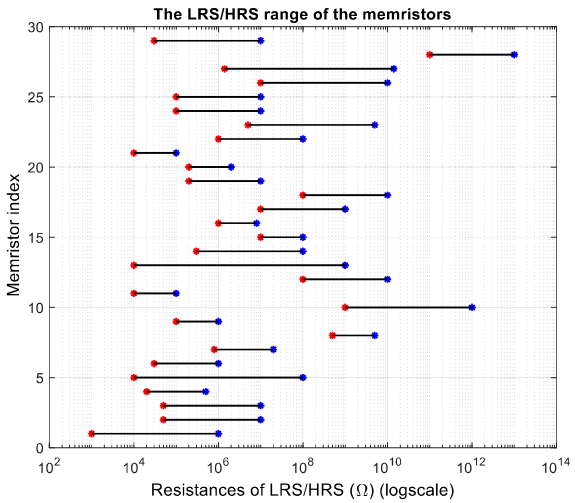

Figure 5: The memristor low resistance state and large resistance state distribution

With the condition of the controllable set voltage and LRS/HRS, we proposed a memristor-based nonlinear module by placing the memristors with various set voltages in cascade as illustrated in Figure 6. In this nonlinear module, the total resistance can be modified corresponding to the applied current stimulus since each memristor in the series can be breaking down (soft breaking down) at different levels of the current stimulus.

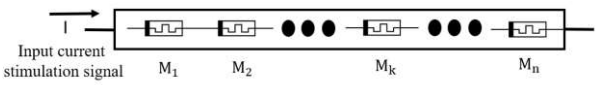

Figure 6: The memristor-based nonlinear module

In the initial state, the total resistance of the module R_{inital} is a sum of the total resistance of the memristors at their high resistance state (HRS) given as the following equation:

$$R_{inital} = R_H^{(1)} + R_H^{(2)} + \cdots + R_H^{(k)} + \cdots + R_H^{(n)} \quad (1)$$

where $R_H^{(k)}$ is the high resistance state value of the k-th memristor in the sequence; $R_L^{(k)}$ is the low resistance state value of the k-th memristor in the sequence; n is the total number of the memristors; and the k is the memristor index, which is also corresponding to the set voltage. This means the set voltage of k-th memristor is always lower than the (k+1)-th memristor.

With the increment of the applied current stimulus signal, the memristors would reach their set voltages sequentially. Correspondently, the memristors with lower set voltages sequentially switch from HRS to LRS. The equation representing the total resistance of the module during this breaking process is:

$$R_k = R_L^{(1)} + \cdots + R_L^{(k)} + R_H^{(k+1)} \cdots + R_H^{(n)} \quad (2)$$

By precise designing the resistance difference between HRS and LRS for each memristor ($R_H^{(k)} - R_L^{(k)} = \Delta R^{(k)}$), we can manipulate the total resistance decreasing rate of the nonlinear module. Since the memristors configuration is in series, the current in each memristor is unique, which is used as the input stimulus signal. Thus, the total resistance of the memristor-based current nonlinear module is summarized using the equations:

$$\begin{cases} R = (n-k)R_{\mathrm{H}} + kR_{\mathrm{L}} \\ I = k \times \Delta I_{th} \end{cases} \quad (3)$$

$$R = \left(n - \frac{I}{\Delta I_{th}}\right) R_{\mathrm{H}} + \frac{I}{\Delta I_{th}} R_{\mathrm{L}} \quad (4)$$

Where

R is the total resistance of nonlinear module;

R_H is the high resistance value of each memristor;

R_L is the low resistance value of each memristor;

n is the total number of memristors in the module;

k is the step index whose value is from 0 to n;

ΔI_{th} is an interval of threshold current values between two consecutive memristor ($I_{th_k} - I_{th_{k-1}} = \Delta I_{th}$), where I_{th_k} is the threshold value of kth memristor.

Figure 7 shows the simulation resistance switching behavior of the proposed module with different decrease rate. The number of the memristors in series is ten. In Figure 7, the breaking step is defined as how many memristors are in their LRS. When all the memristors switch to their LRS, the total resistance would be stay constant in the sum of their low state resistances.

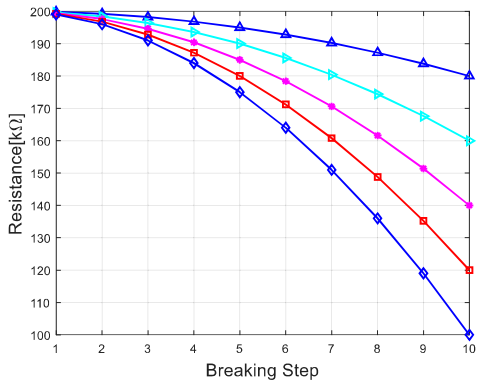

Figure 7: Nonlinear resistance switching behavior with various decreasing rate (Ten memristors).

In a digital computing system, a nonlinear function is generally calculated through a piecewise linear approximation method [23], in which the input domain of a nonlinear function is partitioned into several small segments. In each segment, the output is calculated through a linear approximation. The accuracy of the piecewise linear approximation highly depends upon the segmentation resolution. Inspired by this piecewise linear approximation, the slope of the linear relation of each segment of the input domain can be simulated with a resistance value. If we correlate the segments of the input domain with the breaking step values in Figure 7, we can enable the resistance of the memristor-based nonlinear module continually changing corresponding to the stimulus at each partitioned input domain.

By adding this specified designed memristor-based nonlinear module in an amplifier, we designed a nonlinear function computing module as shown in Figure 8.

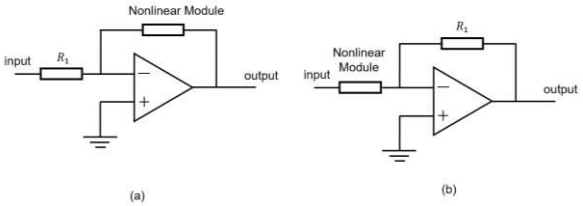

Figure 8: Nonlinear activation function modules

The memristor-based cascaded nonlinear module (Figure 6) is placed in two different locations; consequentially are implemented as two types of input-output nonlinear relationships as illustrated in Figure 9. Note: the amplifier in the design is ideal for the input-output relationship of $V_o = -(R_f/R_1)R_1$.

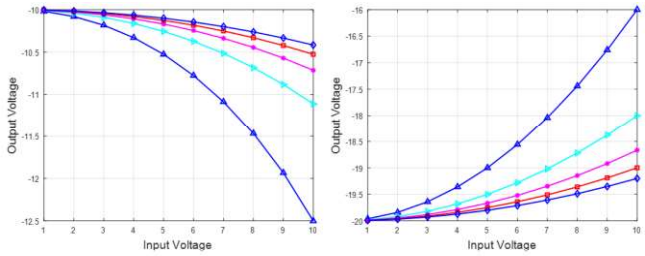

Figure 9: Nonlinear output-input relationships of the proposed nonlinear activation function module

At the different input interval, the slopes of the approximation linear function change correspondingly. By calculating the value of slopes at each input segment interval, we can obtain the resistance requirements at that specific input interval. Thereby the low resistance and high resistance state memristor can be further determined. The inflection points are corresponding to the set voltages of each memristor. By carefully designing the HRS, LRS, and set voltages of the memristors in the sequence shown in Figure 6, the proposed nonlinear activation function modules can potentially implement a monotonic nonlinear function.

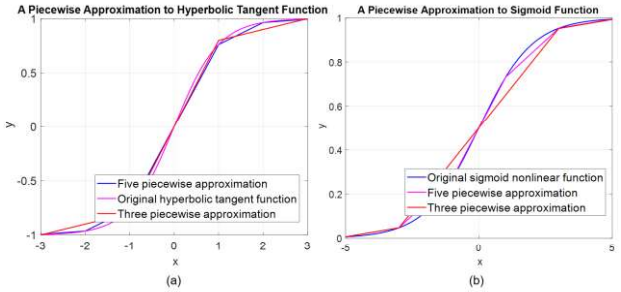

Figure 10: (a) A piecewise approximation to hyperbolic tangent function with various segments; (b)A piecewise approximation to sigmoid function with various piecewise approximation;

In this paper, we implemented the hyperbolic tangent and sigmoid nonlinear functions using the proposed memristor-based nonlinear activation module by applying this design methodology. Figure 10 illustrates the piecewise approximation approaches of two popular nonlinear functions: Hyperbolic tangent and Sigmoid function.

4. SIMULATIONS AND RESISTANCE VARIATION IMPACT ANALYSIS

In this paper, we applied the proposed nonlinear computing module to Long-short term memory (LSTM) for the purpose of evaluation and optimization as a benchmark due to its high computational resource demand [1]. LSTM is a particular recurrent neural network (RNN) by replacing conventional hidden layers with a pre-defined memory cell with data flow controlling gates as shown in Figure 11.

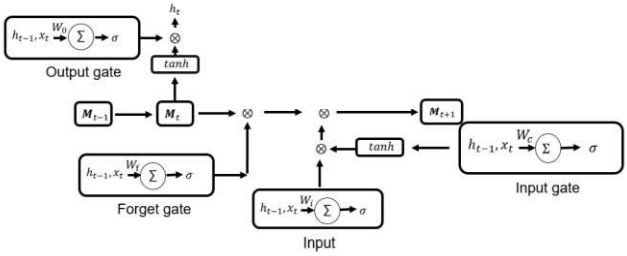

Figure 11: Replacing the hidden layers with memory blocks and mathematical gates

These data flow control gates are implemented by mathematical nonlinear functions. The typical updating equations of LSTM are listed in equation set (5).

$$\begin{cases} f_t = \sigma\left(W_f x_t + U_f h_{t-1} + b_f\right) \\ i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i) \\ o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o) \\ C_t = f_t \circ C_{t-1} + i_t \circ \tilde{C}_t \\ h_t = o_t \circ tanh(C_t) \\ \tilde{C}_t = tanh(W_c x_t + U_c h_{t-1} + b_c) \end{cases} \quad (5)$$

where:

x_t : input vector to the memory cell at time step t;

W_i, W_f , W_c, W_o, U_i, U_f, U_c, U_o, V_o: weight matrix;

b_i, b_f , b_c, b_o are bias vectors;

h_t : is the value of the memory cell at time step t;

i_t and $\tilde{C}_t$ are values of the input gate and the candidate state of the memory cell at time step t;

C_t : cell state at time step t;

o_t and h_t are values of the output gate and the value of the memory cell at time t;

f_t : forget gate vector;

In order to evaluate the proposed nonlinear activation function module, we replaced the embedded nonlinear activation functions in TensorFlow with the proposed nonlinear activation of sigmoid and hyperbolic tangent functions (Figure 10). The classic digit number recognition application was performed with MNIST (Modified National Institute of Standards and Technology) dataset. The back-forward propagation method was used as the training algorithm for LSTM.

The training accuracies at different time steps are shown in Figure 12. As illustrated in Figure 12, the training accuracies with five and three piecewise segments enhance with the time steps in an approximatively same trend. These trend match behaviors are observed in both sigmoid and hyperbolic tangent activation function piecewise approximations. The simulation results indicate that the linear piecewise approximation method of computing nonlinear function would not influence the training efficiency and accuracy. As shown in Figure 12, the proposed memristor-based nonlinear computing module can potentially be used in training process with no accuracy and efficiency compromise.

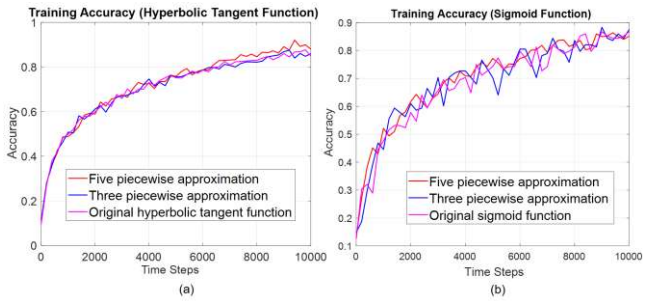

Figure 12: (a) Training accuracy on digit number recognition using hyperbolic tangent activation function with different piecewise approximations; (b) Training accuracy on digit number recognition using sigmoid activation function with different piecewise approximations

As shown in the simulation results, the slopes of the piecewise linear function at each interval are determined and constant. However, in real operation situation, the switching behavior of memristor is not able to be determined. The resistances of high resistance state and low resistance state in each memristor switching cycle shows some levels of variations statistically [11, 24-26]. These variations would influence the slope of the piecewise approximation of the activation function at each nonlinear calculation cycle, and can eventually influence the training accuracy.

In order to investigate the effect of this resistance switching variation on training accuracy, we compared the training accuracy trends when different standard deviation values are given to the LRS/HRS of the memristors at each nonlinear computing cycle. The simulation results are illustrated in Figure 13. The training accuracies with different standard deviations share the same increase trend within initial 1000 time steps. After that time period, the training accuracies begin to increase with different accuracy trends. Higher variations significantly reduce the training accuracy and the accuracy degrades increasingly with the time steps, and hence decrease the training efficiency.

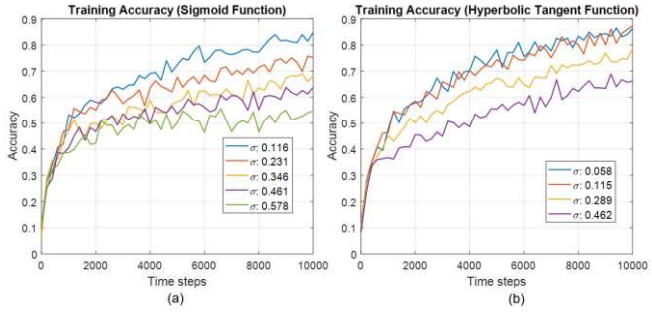

Figure 13: (a) Training accuracies of five piecewise sigmoid function approximation with different variations of memristor resistance; (b) Training accuracies of three piecewise hyperbolic tangent function approximation with different variations of memristor resistance

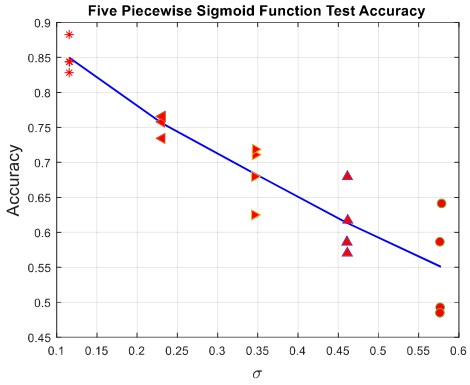

Figure 14: Test accuracy for five piecewise sigmoid function approximation with different variations of switching memristor resistance switching

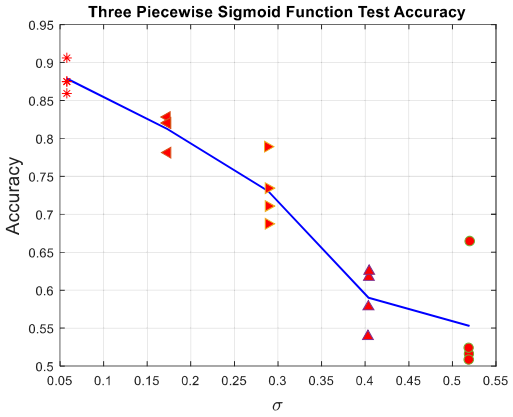

Figure 15: Test accuracy for three piecewise sigmoid function approximation with different variations of switching memristor resistance

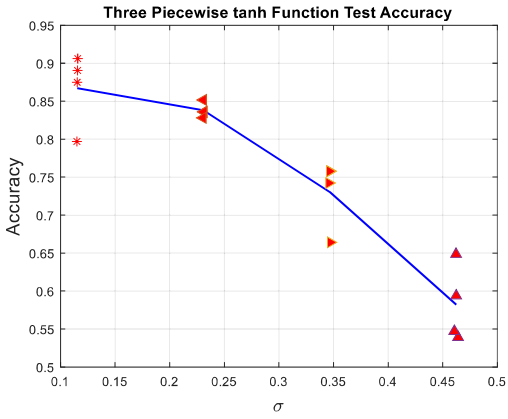

Figure 16: Test accuracy for three piecewise hyperbolic tangent function approximation function approximation with different variations of switching memristor resistance

The low training efficiency also significantly jeopardizes the quality of the final trained neural network. Figure 14 to Figure 16 show the test accuracies vs the different standard deviations of the memristor resistance. With the higher standard deviations of the memristor resistance, the test accuracies also decline accordingly.

5. CONCLUSIONS

In this paper, we proposed, designed and evaluated a nanoscale memristor-based nonlinear computing module. The nonlinear computational operation is intensively and widely performed in deep learning and neuromorphic computing. We demonstrated that comparing the algorithm-based optimization and piecewise approximation methodology, the conducting this computational expensive operation through hardware implementations rather than proves to be much more efficient. The proposed nonlinear computing module is evaluated with the digit number recognition application through LSTM. The simulation results indicate that the training accuracy would not be degraded by using the proposed computing module without considering the resistance variation of the memristor. However, the training accuracy would be impacted by the large resistance variation of memristor on switching operation. The simulation results indicate that the testing accuracies decrease almost linearly with the increase of resistance variation of the memristor. Therefore, limiting the resistance variation of memristor in an acceptable range would be a design task for using the proposed nonlinear activation function module.

6. ACKNOWLEDGMENT

We wish to present a special thank you to Dr. Zhen Zhou. This work would not have been possible without her constructive and valuable suggestions. The opportunity to work with Dr. Zhou was an honor; we immensely appreciate her selfless sharing of knowledge and experience.

REFERENCES

[1] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation,* vol. 9, pp. 1735-1780, 1997.

[2] S. Wang and J. Jiang, "Learning natural language inference with LSTM," *arXiv preprint arXiv:1512.08849,* 2015.

[3] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *arXiv preprint arXiv:1409.0473,* 2014.

[4] A. Graves and N. Jaitly, "Towards end-to-end speech recognition with recurrent neural networks," in *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, 2014, pp. 1764-1772.

[5] A. Laudani, G. M. Lozito, F. R. Fulginei, and A. Salvini, "On training efficiency and computational costs of a feed forward neural network: a review," *Computational intelligence and neuroscience,* vol. 2015, p. 83, 2015.

[6] P. Arbenz, "Numerical Methods for Computational Science and Engineering," *impulse,* vol. 1, p. h3, 2013.

[7] J. Solomon, *Numerical algorithms: methods for computer vision, machine learning, and graphics*: CRC Press, 2015.

[8] M. Hu and J. P. Strachan, "Accelerating Discrete Fourier Transforms with Dot-product Engine," *2016 Ieee International Conference on Rebooting Computing (Icrc),* 2016.

[9] A. H. Namin, K. Leboeuf, R. Muscedere, H. Wu, and M. Ahmadi, "Efficient hardware implementation of the hyperbolic tangent sigmoid function," in *Circuits and Systems, 2009. ISCAS 2009. IEEE International Symposium on*, 2009, pp. 2117-2120.

[10] B. Zamanlooy and M. Mirhassani, "Efficient VLSI implementation of neural networks with hyperbolic tangent activation function," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems,* vol. 22, pp. 39-48, 2014.

[11] A. Adamatzky and L. Chua, *Memristor Networks*: Springer Science & Business Media, 2013.

[12] T. Liu, Y. Kang, M. Verma, and M. Orlowski, "Novel highly nonlinear memristive circuit elements for neural networks," in *Neural Networks (IJCNN), The 2012 International Joint Conference on*, 2012, pp. 1-8.

[13] M. Al-Mamun, S. W. King, and M. K. Orlowski, "Impact of the Heat Conductivity of the Inert Electrode on Reram Memory Cell Performance and Endurance," in *Meeting Abstracts*, 2018, pp. 1476-1476.

[14] M. Al-Mamun and M. Orlowski, "Instability of High Resistance Conductive Filaments in RRAM Cells During the Read Operation," *MRS Advances*.

[15] V. Keshmiri, "A Study of the Memristor Models and Applications," 2014.

[16] H.-S. P. Wong, H.-Y. Lee, S. Yu, Y.-S. Chen, Y. Wu, P.-S. Chen*, et al.*, "Metal–oxide RRAM," *Proceedings of the IEEE,* vol. 100, pp. 1951-1970, 2012.

[17] S. Kvatinsky, E. G. Friedman, A. Kolodny, and U. C. Weiser, "TEAM: Threshold adaptive memristor model," *IEEE Transactions on Circuits and Systems I: Regular Papers,* vol. 60, pp. 211-221, 2013.

[18] K. M. Kim, D. S. Jeong, and C. S. Hwang, "Nanofilamentary resistive switching in binary oxide system; a review on the present status and outlook," *Nanotechnology,* vol. 22, p. 254002, 2011.

[19] H.-S. P. Wong, C. Ahn, J. Cao, H.-Y. Chen, S. W. Fong, Z. Jiang*, et al.* (2017). *Stanford Memory Trends*. Available: https://nano.stanford.edu/stanford-memory-trends

[20] J. Lee, J. Shin, D. Lee, W. Lee, S. Jung, M. Jo*, et al.*, "Diode-less nano-scale ZrO x/HfO x RRAM device with excellent switching uniformity and reliability for high-density cross-point memory applications," in *Electron Devices Meeting (IEDM), 2010 IEEE International*, 2010, pp. 19.5. 1-19.5. 4.

[21] J. Sohn, S. Lee, Z. Jiang, H.-Y. Chen, and H.-S. P. Wong, "Atomically thin graphene plane electrode for 3D RRAM," in *Electron Devices Meeting (IEDM), 2014 IEEE International*, 2014, pp. 5.3. 1-5.3. 4.

[22] Q. Luo, X. Xu, H. Liu, H. Lv, T. Gong, S. Long*, et al.*, "Demonstration of 3D vertical RRAM with ultra low-leakage, high-selectivity and self-compliance memory cells," in *Electron Devices Meeting (IEDM), 2015 IEEE International*, 2015, pp. 10.2. 1-10.2. 4.

[23] H. Amin, K. M. Curtis, and B. R. Hayes-Gill, "Piecewise linear approximation applied to nonlinear function of a neural network," *IEE Proceedings-Circuits, Devices and Systems,* vol. 144, pp. 313-317, 1997.

[24] Y. Gao, D. C. Ranasinghe, S. F. Al-Sarawi, O. Kavehei, and D. Abbott, "Memristive crypto primitive for building highly secure physical unclonable functions," *Scientific reports,* vol. 5, 2015.

[25] H. An, Z. Zhou, and Y. Yi, "Opportunities and challenges on nanoscale 3D neuromorphic computing system," in *Electromagnetic Compatibility & Signal/Power Integrity (EMCSI), 2017 IEEE International Symposium on*, 2017, pp. 416-421.

[26] H. An, M. A. Ehsan, Z. Zhou, F. Shen, and Y. Yi, "Monolithic 3D neuromorphic computing system with hybrid CMOS and memristor-based synapses and neurons," *Integration, the VLSI Journal,* 2017.