Exploring the Precision Limitation for RRAM-Based Analog Approximate Computing

Boxun Li, Peng Gu, Yu Wang, and Huazhong Yang
Tsinghua University

Editor's notes:
RRAM-based approximate computing systems are significantly more energy efficient than many digital approximate computing systems, but their accuracy is less easily controlled and quantified. This paper analyzes the precision limitation for such systems and highlights the importance of RRAM device resolution in low-resistance states.
—Qiang Xu, The Chinese University of Hong Kong

Approximate computing is an emerging design paradigm that is able to achieve better energy efficiency by trading off the quality (e.g., accuracy) and effort (e.g., energy) of computation [2]. Approximate computing takes advantage of the characteristic that many modern applications, such as machine learning and signal processing, are able to produce results with acceptable quality even if many calculations are executed imprecisely [3]. The tolerance of imprecise computation can be leveraged to acquire substantial performance gains and has inspired a wide range of architectural innovations [4].

Recent work in approximate computing mainly has focused on the design of basic elements, such as approximate adders and logics [5], [6]. These techniques have adequately demonstrated the benefit of approximate computing, but the fixed functionality and low-level design limit further improvement of performance. Besides, these techniques are all based on the complementary metal–oxide–semiconductor (CMOS) technology, despite the circumstance where device technology innovations have offered a great opportunity for radically different forms of architecture design [7].

In recent years, the innovation of resistive-switching random access memory (RRAM) devices has demonstrated a promising solution to the implementation of approximate computing. The RRAM device and the crossbar structure provide an innovative alternative to the von Neumann architecture by changing the architecture to combine computation and memory together, and naturally realizing the matrix–vector multiplication. Based on these characteristics, the RRAM devices are able to implement artificial neural networks (ANNs) and realize a RRAM-based approximate computing (RAC) framework which has been demonstrated hundreds of times of power efficiency gains compared with the central processing unit (CPU) [8].

Compared with many digital approximate computing systems, whose accuracy will be influenced

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/MDAT.2015.2487218
Date of publication: 05 October 2015; date of current version: 18 January 2016.

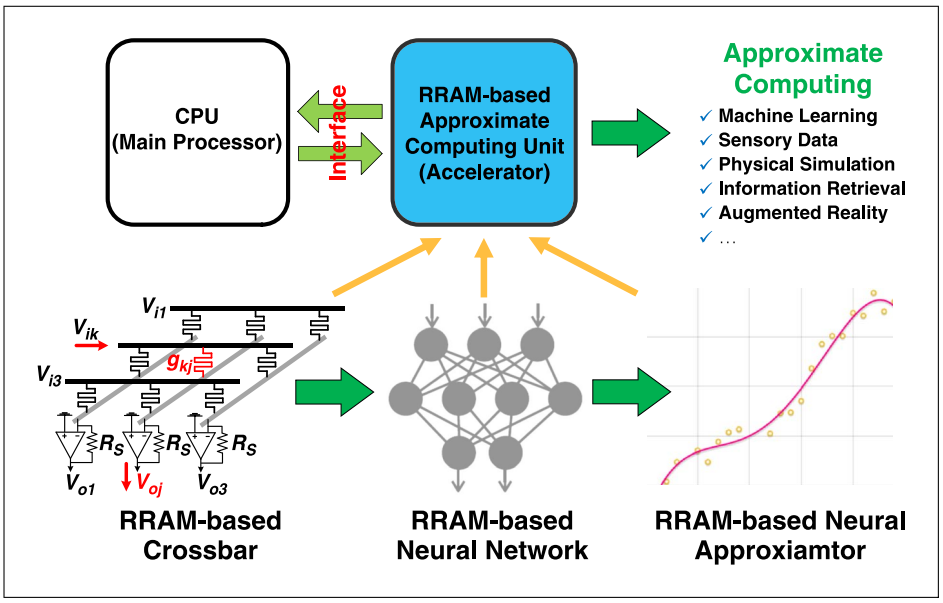

Figure 1. Overview of RRAM-based approximate computing (RAC).

by the algorithm or the functionality of the system, the precision of RAC is also impacted by two other factors: 1) the bit level of RRAM devices; and 2) the resolution of the interface between digital systems and analog RAC units.

On the one hand, the functionality of RAC directly depends on the resistance states of RRAM devices. The ideal RAC requires continuous variable resistance states. However, the RRAM resistance states are usually quantized to discrete levels to reduce the complexity of write circuit design and realize a quick RRAM state tuning. The deviation of RRAM resistance caused by bit-level quantization will significantly impact the precision of RAC [9].

On the other hand, the interfaces between digital and analog units are always key considerations of RAC. The RRAM crossbar realizes matrix operations in analog. Analog-to-digital and digital-to-analog converters (AD/DAs) are usually required in this mixed-signal system to bridge the digital part and RRAM-based analog signal processing units. The resolution of AD/DA will directly limit the precision of RAC. Moreover, compared with the high density and efficiency of RRAM units, AD/DAs not only take up most of the chip area, but also consume much more power than RRAM devices and other analog peripheral circuits. The overhead of data conversions also significantly hinder the potential efficiency gains of RAC [1].

In this work, we first investigate the distribution and impact of RRAM bit level and demonstrate the effectiveness of different RRAM resistance quantizing methods. We then analyze the impact of AD/DA interface on the precision of RAC. Moreover, a joint analysis of the RRAM bit level and AD/DA resolution is presented. Finally, recent work [1] introduced a more flexible configuration of RAC interface resolution by MErging the Interface (MEI) into the RRAM crossbar itself. The impact of RRAM bit level and interface resolution on MEI is also discussed in this work.

Preliminaries

RRAM device characteristics and crossbar structure

A RRAM device is a passive two-port element with variable resistance states. The key structure is a resistive switching layer, which can move when a large voltage is applied on the device [10].

The RRAM devices can be used to build the crossbar structure as shown in Figure 1. The relationship between the input voltage "vector" ($\vec{V}_i$) and output voltage "vector" ($\vec{V}_o$) can be expressed as follows [11]:

$$V_{o,j} = \sum_k c_{k,j} \cdot V_{i,k} \tag{1}$$

where k ($k = 1, 2, \ldots, N$) and j ($j = 1, 2, \ldots, M$) are the indexes of input and output ports of the crossbar. The parameter $c_{k,j}$ can be represented by the conductivity of the RRAM device ($g_{k,j}$) and the load resistor (R_s) as

$$c_{k,j} = g_{k,j} \cdot R_S. \tag{2}$$

Therefore, the RRAM crossbar array is able to perform analog matrix–vector multiplication and the parameters of the matrix depend on the RRAM resistance states.[1]

[1]Since both R_s and $g_{k,j}$ can only be positive, two crossbar arrays are usually required to represent positive and negative weights by the subtraction of paired RRAM devices.

RRAM-based approximate computing (RAC)

With the RRAM crossbar structure, RAC can be implemented by realizing analog ANNs, where the RRAM-based ANN works as a universal approximator [8].

An ANN processes the data by executing the following operations layer by layer:

$$\vec{y}_j = f(W_{ij} \cdot \vec{x}_i + \vec{b}_i) \tag{3}$$

where $\vec{x}_i$ and $\vec{y}_j$ represent the data in the ith and $j$thlayers of the network, respectively. W_{ij} is the weight matrix between layer i and layer j. $f(x)$ is a nonlinear activation function, e.g., the sigmoid function $f(x) = 1/(1 + e^{-x})$.

The basic building blocks of an ANN are the matrix–vector multiplication and the nonlinear activation function, which can be implemented with RRAM crossbar and analog peripheral circuits, respectively [8].

An artificial neural networks is an "end-to-end" model, which is able to learn the relationship between the input and output data automatically. Specifically, an ANN can be configured as an efficient model to fit complex numerical functions and realize the RAC as shown in Figure 1. It has been demonstrated that hundreds of times of power efficiency gains can be achieved by executing analog approximate computing compared with the CPU [8].

Exploring the precision limitation of RAC

Impact of RRAM resistance resolution

Generally speaking, the resistance range of RRAM devices is usually quantized to discrete levels for RRAM state tuning and reducing the complexity of write circuit design. However, the ideal RAC requires continuous variable resistance states. The resistance resolution will significantly influence the precision of RAC.

The weights of a neural approximator depend on the resistance states of RRAM devices in the crossbar structure directly. As described in (2), the RAC parameters need to be mapped to the appropriate states of RRAM devices in the crossbar. Improperly converting the network weights to the RRAM conductance states may result in many problems: the converted results exceed the actual range of the RRAM device, the dynamic range of converted results is so small that the RRAM state may easily saturate [8], [12].

In order to minimize risk of improper parameter conversion and tolerate process variation, Li et al. [8] analyzed the relationship between the RRAM resistance state and neural approximator parameter [W_{ij} in (3)] and proposed a mapping method as follows:

$$\begin{cases} \frac{1}{R_{\text{pos}}} = g_{\text{pos}} = g_{\text{mid}} + \frac{w}{2R_S} \\ \frac{1}{R_{\text{neg}}} = g_{\text{neg}} = g_{\text{mid}} - \frac{w}{2R_S} \end{cases} \tag{4}$$

where g_{pos} and g_{neg} represent the mapped RRAM conductance states in the positive and negative crossbar arrays, respectively. g_{mid} is the median of RRAM conductance states which takes process variation into consideration and avoids improper converting [8].

In neural approximator, parameters (w) are usually small and around zero. According to (4), most mapped RRAM resistance states (R_{pos} and R_{neg}) will fall into a small range around $1/g_{\text{mid}}$. Figure 2 demonstrates a case study of the distribution of the neural network parameters (W_{ij}) and the mapped RRAM resistance states of the positive crossbar (R_{pos}). The parameters are taken from the first layer of the neural approximators for six benchmarks [4]. The RRAM resistance range is set to 500 Ω–10 kΩ [10]. The load resistance R_S is set to 2 kΩ [9]. The result illustrates that the mapped RRAM resistance states fall into a narrow range around ~1 kΩ.

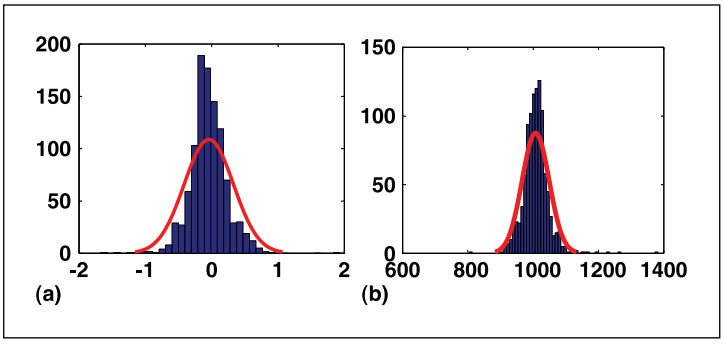

Figure 2. Distribution of the parameter range (W_{ij}) in the first layer of neural approximator for six benchmarks [4], and the corresponding RRAM resistance states in the positive crossbar array mapped by (4). (a) Approximator parameters (W_{ij}). (b) Mapped RRAM resistance (Ω).

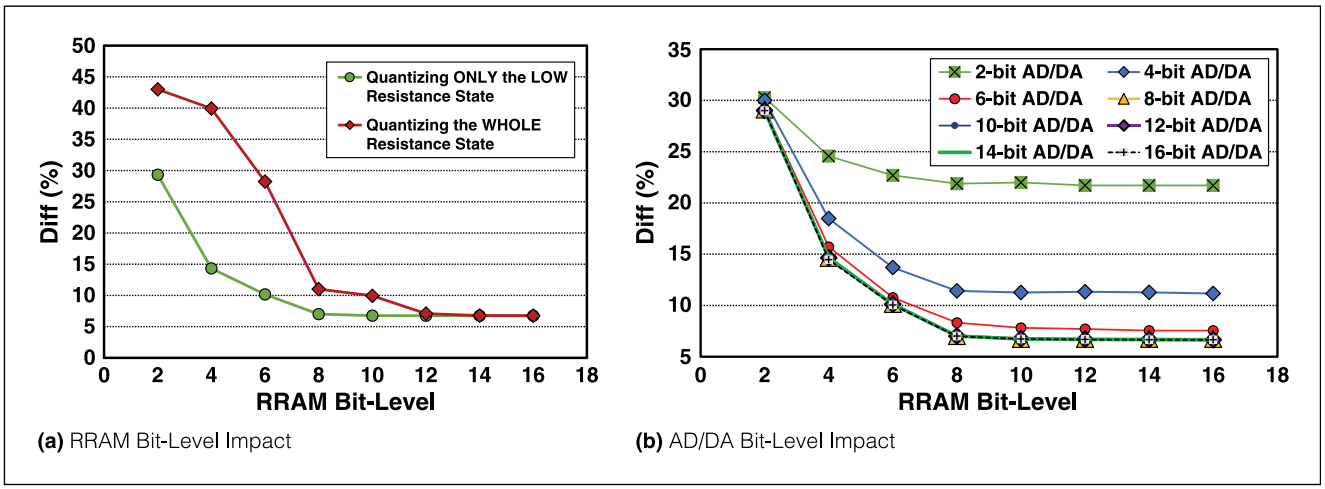

Figure 3. Impact of RRAM bit level and AD/DA resolution on the precision of a $2 \times 8 \times 2$ RAC for robotics. (a) The bit level of AD/DA is set to 8 b when quantizing the RRAM resistance state. (b) RAC precision under different AD/DA bit level and RRAM bit level.

The RRAM is usually equipped with a very large range of resistance state to enable multilevel storage and enlarge the interval between two neighbor levels. However, the neural approximator parameter w depends on the inversion of the RRAM resistance. Although the RRAM devices are known to have a large range of resistance states, most high-resistance states will contribute little to the magnitude of the inversion. As a result, RAC will make use of only a small part of RRAM resistance range in low resistance, and the high-resistance state is seldom reached. When the RRAM device is used for approximate computing, most quantization effort should be paid to carefully tuning the low-resistance state, instead of the whole resistance range.

Specifically, we evaluate the impact of two quantization methods, i.e., only quantizing the low-resistance states and quantizing the whole resistance states uniformly, on the RAC precision. We use a $2 \times 8 \times 2$ RAC[2] for robotics with 8-b AD/DA as a case study [4]. The experiment setup is the same as Figure 2. The RRAM resistance range is set to 500 Ω–10 kΩ as reported in [10]. The medium resistance is set to 1 kΩ as the load resistance R_S is set to 2 kΩ as [9]. We choose the maximum narrow resistance as the low-resistance states to be quantized, i.e., 500–1500 Ω, which evenly distributes around the medium resistance. Finally, the parameters that exceed the mapping range of (4), i.e., $[-2R_s g_{\text{mid}}, 2R_s g_{\text{mid}}] = [-4, 4]$, will be truncated in case of overflow. The results are illustrated in Figure 3a. Conclusions can be drawn as follows.

- As the mapping relationship depends on the inversion of the RRAM resistance, the low-resistance states will contribute significantly to the mapped neural approximator parameters w. Compared with the method which only enables high resolution in low-resistance states (500–1500 Ω), the error rate increases drastically after quantizing uniformly the whole resistance range (500 Ω–10 kΩ), especially when the bit level is low. This result is expected because most mapped RRAM states will be quantized to the same low-resistance state in this method, which will lead to a large deviation of approximator parameters.
- The result also demonstrates that the method which uniformly quantizes the whole resistance range requires around four more bits to achieve a comparable resolution of low-resistance states and realize a similar precision. This phenomenon can be predicted as the ratio of the resistance range to be quantized by two methods is $((10\ \text{k}\Omega – 500\ \Omega)/(1500 – 500\ \Omega)) \approx 3.3$ b.
- Finally, the results show that the bit-level requirement for RAC system is kind of "strict." A 6-b RRAM device is expected to achieve an acceptable

[2]An $I \times H \times O$ RAC represents that the RAC consists of a three-layer ANN with I nodes in the input layer, H nodes in the hidden layer, and O nodes in the output layer.

precision. However, the precision no longer improves after the RRAM device reaches a certain bit level. And thus we do not need to go beyond those bit levels (e.g., 8 b).

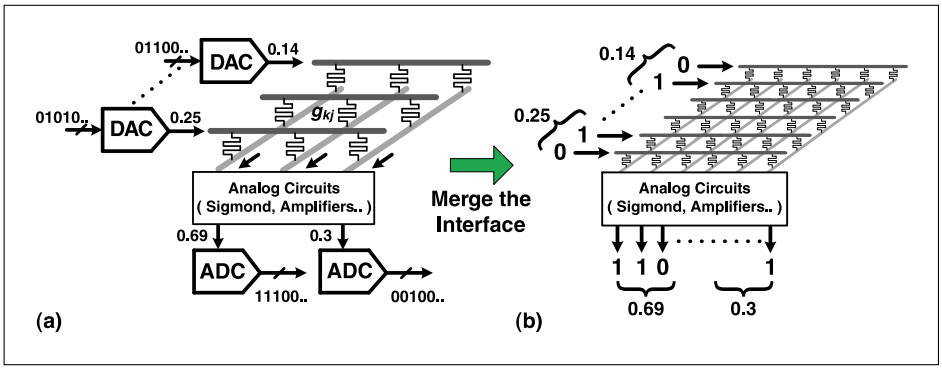

Figure 4. (a) Original RAC with AD/DAs. (b) Basic idea of MEI [1].

Impact of the interface

The RRAM system executes approximate computing in analog. Interfaces, e.g., AD/DAs, are usually required in this mixed-signal system to bridge the digital part and the analog RAC unit. The resolution of AD/DA will also limit the precision of RAC. We use the same benchmarks in the Impact of RRAM resistance resolution section to demonstrate the impact of AD/DA resolution. The result is shown in Figure 3b. A similar trend can be found in the results, where the error rate decreases with the rise of bit level of AD/DA. And the precision of RAC begins to saturate again after we raise the AD/DA bit level to 8-b resolution.

Joint analysis

We further analyze the impact of interface resolution and RRAM bit level jointly. We evaluate the precision of RAC under different AD/DA bit level and RRAM bit level. The results are illustrated in Figure 3b. Compared with the interface resolution, the RRAM bit level demonstrates to have more impact on the RAC precision.

Specifically, for the case study of a $2 \times 8 \times 2$ RAC for robotics, the inflection, where the precision starts to increase slowly, is 6 and 8 b for AD/DA and RRAM devices, respectively. The bit-level requirement of RRAM devices for RAC demonstrates to be higher than the AD/DA interface. Moreover, the precision begins to stall when the bit level of AD/DA interface becomes 8 b. As the RAC calculates the relationship between the input and output signals approximately, increasing the resolution of AD/DA interface will have little effect on the RAC precision, especially when the AD/DA resolution gets better than the precision of RAC. In our study, a 6–8-b AD/DA is usually sufficient to perform a wide range of approximate computing tasks [4]. However, we can still improve the precision, though very slowly, by increasing the RRAM bit level from 8 to 12 b. The parameters of RRAM devices directly determine the functionality of neural approximator. A better resolution or a better bit level of RRAM devices can usually lead to a better precision of RAC.

Exploring the precision limitation of MEI

Merging the interface (MEI)

In the original RAC, the AD/DA not only limits the precision of RAC, but also takes up most of the chip area and power consumption. For example, for the $2 \times 8 \times 2$ RAC benchmark used in previous sections, RRAM devices only account for ~1% proportion of the whole system. In contrast, AD/DAs contribute to a significant portion (>85%) of the area and power consumption [1]. Consequently, the potential efficiency gains of RAC are significantly hindered by the interface overhead.

In order to reduce the interface overhead, Li et al. [1] introduced the MEI technique as demonstrated in Figure 4 The basic idea is inspired by the "end-to-end" characteristic of ANN. MEI tries to make the neural approximator directly learn the relationship between the binary 0/1 arrays representing the input and output digital signals, instead of approximating the function between the analog value converted by AD/DA. For example, for a traditional $I \times H \times O$ RAC equipped with B-bit AD/DAs, MEI will directly set $I \times B$ and $O \times B$ ports in the input layer and the output layer of MEI, respectively. Digital 0/1 signals will be set to MEI in parallel, and MEI will directly calculate the corresponding digital output arrays. Therefore, MEI is able to directly connect to digital systems without requiring AD/DAs. Simulation results demonstrate that MEI significantly saves 54.63%–86.14% area

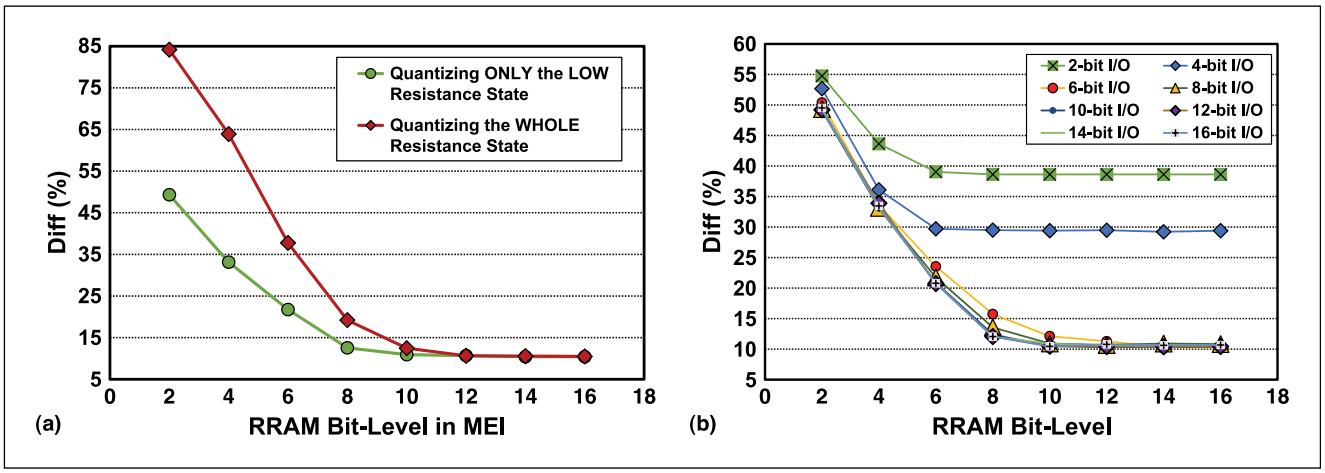

Figure 5. Impact of RRAM bit level and AD/DA resolution on the precision of a $(2 \cdot B) \times 32 \times (2 \cdot B)$ RAC with MEI for robotics. (a) The input/output bit level of MEI is set to 8 b when quantizing the RRAM resistance state. (b) MEI precision under different input/output bit level and RRAM bit level.

and reduces 61.82%–86.8% power consumption under quality guarantees in a set of six diverse benchmarks.

Impact of the resolution of RRAM and input signals

We also evaluate the impact of RRAM bit level and input/output resolution on the precision of MEI. The $2 \times 8 \times 2$ RAC for robotics is used again with the same experimental setup. We scale up the RAC with MEI to $(2 \cdot B) \times 32 \times (2 \cdot B)$as in [1], where B represents the number of input/output ports in MEI (i.e., the resolution of the interface).

We first set the input and output resolution of MEI to 8 b when quantizing RRAM resistance. The results are illustrated in Figure 5a. Compared with the original architecture with AD/DAs, MEI also requires to pay more attention to the low-resistance states when quantizing the RRAM devices. ~8 b is expected to achieve an acceptable precision. However, MEI performs worse than the original RAC with AD/DA when the bit level is low. This phenomenon may come from the scale of MEI. As MEI expands a B-bit analog port to an array of B 0-1 signal ports, B^2 RRAM devices are demanded in MEI compared with the original RAC with AD/DA. And thus the larger number of RRAM devices will magnify the fluctuation, which will lead to larger sensitivity of RRAM bit level for MEI when the resistance resolution is low. However, after the RRAM bit level reaches a certain value, e.g., 8 b, MEI is able to demonstrate better robustness to RRAM process variation as it is easier for the large number of RRAM devices to compensate and cooperate with each other under small fluctuation [1].

We then evaluate the impact of input/output port number to the precision of MEI. Figure 5b illustrates the simulation results. The precision also begins to stall after the number of input/output ports reaches 10 b. And compared with the original RAC, where the precision begins to saturate after the bit level of AD/DA reaches 8 b, MEI trends to reap more benefits by exposing and increasing the resolution of input/output ports.

Finally, Figure 5b demonstrates a joint evaluation of the precision of MEI between the RRAM bit level and input/output number. Compared with the number of input/output ports (bit level of interface), the RRAM bit level has more impact on the precision of MEI. For the case study of a $(2 \cdot B) \times 32 \times (2 \cdot B)$, the precision begins to saturate after the input/output number and RRAM bit level come to 10 and 14 b, respectively. At the same time, as the cost of increasing the interface resolution of MEI, where we only need to increase the number of input/output ports and train a better neural approximator, is much lower than improving the resolution of AD/DA, MEI makes it possible to let the designer focus on optimizing the resistance resolution of RRAM devices, instead of trading off between the AD/DA choice and RRAM bit level.

RAC PROVIDES A promising solution to boost performance and power efficiency. In this work, we

demonstrate that both the interface resolution and RRAM bit level are important to the precision of RAC. Moreover, experimental results illustrate that the precision no longer improves after the RRAM device or the AD/DA resolution reaches a certain bit level, and thus we do not need to go beyond those bit levels. Finally, we also compare the efficiency of different quantizing methods for RRAM resistance states, and demonstrate that we should pay more attention to the resolution of RRAM devices in low-resistance states, instead of the whole resistance states. ■

Acknowledgment

This work was supported by the 973 Project 2013CB329000; the National Natural Science Foundation of China under Grant 61373026; the Brain Inspired Computing Research, Tsinghua University under Grant 20141080934; the Tsinghua University Initiative Scientific Research Program; the Importation and Development of High-Caliber Talents Project of Beijing Municipal Institutions; and Huawei Technologies.

■ References

[1] B. Li, L. Xia, P. Gu, Y. Wang, and H. Yang, "Merging the interface: Power, area and accuracy co-optimization for RRAM crossbar-based mixed-signal computing system," in *Proc. 52nd Annu. Design Autom. Conf.*, 2015, pp. 13:1–13:6.

[2] Q. Zhang, F. Yuan, R. Ye, and Q. Xu, "Approxit: An approximate computing framework for iterative methods," in *Proc. 51st ACM/EDAC/IEEE Design Autom. Conf.*, 2014, DOI: 10.1109/DAC.2014.6881424.

[3] R. Ye et al., "On reconfiguration-oriented approximate adder design and its application," in *Proc. Int. Conf. Comput.-Aided Design*, 2013, pp. 48–54.

[4] H. Esmaeilzadeh, A. Sampson, L. Ceze, and D. Burger, "Neural acceleration for general-purpose approximate programs," in *Proc. 45th Annu. IEEE/ACM Int. Symp. Microarchitect.*, 2012, pp. 449–460.

[5] V. Gupta, D. Mohapatra, A. Raghunathan, and K. Roy, "Low-power digital signal processing using approximate adders," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 32, no. 1, pp. 124–137, Jan. 2013.

[6] S. Venkataramani, A. Sabne, V. Kozhikkottu, K. Roy, and A. Raghunathan, "Salsa: Systematic logic synthesis of approximate circuits," in *Proc. 49th Annu. Design Autom. Conf.*, 2012, pp. 796–801.

[7] V. Narayanan et al., "Video analytics using beyond CMOS devices," in *Proc. Conf. Design Autom. Test Eur.*, 2014, pp. 344:1–344:5.

[8] B. Li et al., "RRAM-based analog approximate computing," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 34, no. 12, pp. 1905–1917, Dec. 2015.

[9] P. Gu et al., "Technological exploration of RRAM crossbar array for matrix-vector multiplication," in *Proc. 20th Asia South Pacific Design Autom. Conf.*, 2015, pp. 106–111.

[10] S. Yu et al., "A low energy oxide-based electronic synaptic device for neuromorphic visual systems with tolerance to device variation," *Adv. Mater.*, vol. 25, no. 12, pp. 1774–1779, 2013.

[11] M. Hu, H. Li, Q. Wu, and G. S. Rose, "Hardware realization of BSB recall function using memristor crossbar arrays," in *Proc. Design Autom. Conf.*, 2012, pp. 498–503.

[12] B. Li, Y. Wang, Y. Chen, H. H. Li, and H. Yang, "Ice: Inline calibration for memristor crossbar-based computing engine," in *Proc. Conf. Design Autom. Test Eur.*, 2014, pp. 184:1–184:4.

Boxun Li is currently working toward an MS at the Department of Electronic Engineering, Tsinghua University, Beijing, China. His research mainly focuses on energy-efficient hardware computing system design, and parallel computing based on GPU. Li has a BS in electronic engineering from Tsinghua University (2013). He is a Student Member of the IEEE.

Peng Gu is currently working toward a PhD at the Department of Electrical and Computer Engineering, University of California Santa Barbara, Santa Barbara, CA, USA. His research interests include low-power system design, hardware acceleration, and computing with emerging devices. Gu has a BS in electronic engineering from Tsinghua University, Beijing, China (2015). He is a Student Member of the IEEE.

Yu Wang is an Associate Professor at the Department of Electronic Engineering, Tsinghua University, Beijing, China. His research focuses on power/reliability aware system design methodologies, parallel circuit analysis, and application-specific heterogeneous hardware computing, especially brain-related topics. Wang has a PhD (honors) from

Tsinghua University (2007). He is a Senior Member of the IEEE.

Huazhong Yang is a Specially Appointed Professor of the Cheung Kong Scholars Program at the Department of Electronic Engineering, Tsinghua University, Beijing, China. His research interests include wireless sensor networks, data converters, parallel circuit simulation, nonvolatile processors, and energy-harvesting circuits. Yang has a PhD from Tsinghua University (1998). He is a Senior Member of the IEEE.

■ Direct questions and comments about this article to Yu Wang, Department of Electrical Engineering, Tsinghua National Laboratory for Information Science and Technology (TNList), Centre for Brain Inspired Computing Research (CBICR), Tsinghua University, Beijing, China; yu-wang@tsinghua.edu.cn.