Technological Exploration of RRAM Crossbar Array for Matrix-Vector Multiplication

Peng Gu[1], Boxun Li[1], Tianqi Tang[1], Shimeng Yu[2], Yu Cao[2], Yu Wang[1], Huazhong Yang[1]
[1]Dept. of E.E., Tsinghua National Laboratory for Information Science and Technology (TNList), Tsinghua University, Beijing, China
[2]School of Electrical, Computer and Energy Engineering, Arizona State University, Arizona, USA
e-mail: yu-wang@mail.tsinghua.edu.cn

***Abstract*—The matrix-vector multiplication is the key operation for many computationally intensive algorithms. In recent years, the emerging metal oxide resistive switching random access memory (RRAM) device and RRAM crossbar array have demonstrated a promising hardware realization of the analog matrix-vector multiplication with ultra-high energy efficiency. In this paper, we analyze the impact of nonlinear voltage-current relationship of RRAM devices and the interconnect resistance as well as other crossbar array parameters on the circuit performance and present a design guide. On top of that, we propose a technological exploration flow for device parameter configuration to overcome the impact of nonideal factors and achieve a better trade-off among performance, energy and reliability for each specific application. The simulation results of a support vector machine (SVM) and MNIST pattern recognition dataset show that the RRAM crossbar array-based SVM is robust to the input signal fluctuation but sensitive to the tunneling gap deviation. A further resistance resolution test presents that a 4-bit RRAM device is able to realize a recognition accuracy of $\sim 90\%$, indicating the physical feasibility of RRAM crossbar array-based SVM. In addition, the proposed technological exploration flow is able to achieve 10.98% improvement of recognition accuracy on the MNIST dataset and 26.4% energy savings compared with previous work.**

I. Introduction

The matrix-vector multiplication is of significant importance in many applications [1], [2]. Recently, the emerging metal oxide resistive switching random access memory (RRAM) device and RRAM crossbar array have demonstrated an efficient hardware implementation of the matrix-vector multiplication [3], [4], [5]. Many studies have explored the potential of computing with RRAM crossbar array. For example, a low power approximate computing system, which is based on the RRAM crossbar implementation of matrix multiplication and neural network, has demonstrated power efficiency of ≥ 400 GFLOPS/W [6].

Although many works have adequately demonstrated the benefits of RRAM crossbar-based computing systems, many important nonideal factors are neglected. Most of the previous works are based on a simplified circuit model [5] [7] [8] and use a linear resistor to represent an RRAM device, which may lead to inaccurate conclusions [9]. Moreover, some nonideal factors, such as the nonlinear voltage-current relationship of RRAM devices, the interconnect resistance, and the resistance state deviation, may significantly influence the performance of RRAM crossbar array-based computing systems. For instance, the interconnect resistance between two adjacent RRAM devices is 2.97Ω for 22nm technology node [10]. The resistance of a wire in a 100×100 crossbar would be as large as $\sim 300\Omega$. Since the lowest resistance state of an RRAM device is only $\sim 500\Omega$, such a large interconnection resistance may have a significant impact on the voltage distribution [11]. In conclusion, a detailed and comprehensive analysis of the impact of these nonideal factors is still lacking.

The contributions of this paper include:

1) We analyze the impact of various nonideal factors on the performance of RRAM crossbar array. We demonstrate that the RC delay of the array could be ignored ($\sim 10ps$ for a 100×100 crossbar according to our simulation). We also propose that the nonlinearity of RRAM devices and interconnect resistance will have a major influence on the computation accuracy of output voltage. Moreover, we present that the minimum resistance state of RRAM devices has little impact on computation accuracy while increasing load resistance will significantly improve computation accuracy.
2) We propose a technological exploration flow of RRAM crossbar array to mitigate the impact of nonideal factors and realize a better trade-off among performance, energy, and reliability for each specific application. The proposed flow includes the technology node and load resistance configuration, the algorithm of mapping matrix parameters to RRAM resistance states, and an iterative solution to achieve a better trade-off between power and performance.
3) Finally, we use the MNIST dataset and a linear SVM classifier as a case study to test the performance of the proposed technology exploration flow. The simulation results demonstrate that the exploration flow significantly contributes to configuring the RRAM and crossbar array parameters and achieving 10.98% improvement of recognition accuracy and 26.4% power reduction compared with previous work [7].

II. Preliminaries

A. RRAM Characteristics and Device Model

The RRAM device is a passive two-port element based on metal oxide materials like TiO_x [12], WO_x [13], and HfO_x [14] with variable resistance. In this paper, we use the HfO_x based RRAM for study because it is one of the most mature RRAM materials explored [15].

Fig. 1(a) demonstrates a 2D filament model of the HfO_x based RRAM [11]. Its conductance is exponentially dependent on the tunneling gap distance (d). When a large voltage is applied on the electrodes, the tunneling gap distance d will change due to the electric filed and temperature-enhanced oxygen ion migration, and the resistivity of RRAM device will switch between the highest resistance state R_{OFF} and the lowest resistance state R_{ON}. Theoretically, an

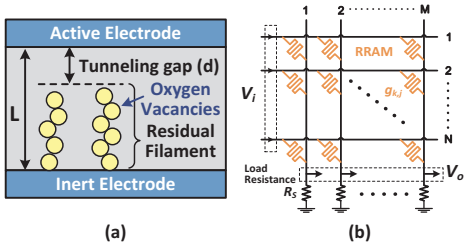

Fig. 1. (a). Physical model of the HfO_x based RRAM. (b). Structure of the RRAM Crossbar Array.

This work was supported by 973 project 2013CB329000, National Science and Technology Major Project (2011ZX03003-003-01, 2013ZX03003013-003) and National Natural Science Foundation of China (No.61373026, 61261160501, 61271269), The Importation and Development of High-Caliber Talents Project of Beijing Municipal Institutions, and Tsinghua University Initiative Scientific Research Program.

978-1-4799-7792-5/15/$31.00 ©2015 IEEE

RRAM device can achieve any resistance in the range between R_{ON} and R_{OFF}. This work focuses on the choice of the resistivity of RRAM devices and other device parameters. How to tune the RRAM device to the specific resistance state will not be discussed in the paper.

For the HfO_x based RRAM device, the nonlinear I-V relationship can be empirically expressed as follows [11]:

$$I = I_0 \cdot \exp(-\frac{d}{d_0}) \cdot \sinh(\frac{V}{V_0}) \tag{1}$$

where d is the average tunneling gap distance. I_0 ($\sim 1mA$), d_0 ($\sim 0.25nm$) and V_0 ($\sim 0.25V$) are fitting parameters through experiments.

In order to analyze the device and circuit interaction issues for the RRAM crossbar array based computation, we use HSPICE to simulate the circuit performance based on a recent Verilog-A model described in [11].

B. RRAM Crossbar Array

The RRAM crossbar array is able to perform the analog matrix-vector multiplication efficiently. Fig. 1(b) illustrates the structure of the RRAM crossbar array. The relationship between the input voltage vector ($\vec{V_i}$) and output voltage vector ($\vec{V_o}$) can be expressed as follows [5]:

$$\begin{bmatrix} V_{o,1} \\ \vdots \\ V_{o,M} \end{bmatrix} = \begin{bmatrix} c_{1,1} & \cdots & c_{1,N} \\ \vdots & \ddots & \vdots \\ c_{M,1} & \cdots & c_{M,N} \end{bmatrix} \begin{bmatrix} V_{i,1} \\ \vdots \\ V_{i,N} \end{bmatrix} \tag{2}$$

Supposing that k (k = 1,2,..,N) and j (j = 1,2,..,M) are the index numbers of input and output voltages, the matrix parameter $c_{k,j}$ can be represented by the conductivity of the RRAM device ($g_{k,j}$) and the load resistor (g_s) as:

$$c_{k,j} = \frac{g_{k,j}}{g_s + \sum_{l=1}^{N} g_{k,l}} \tag{3}$$

Since both g_s and $g_{k,j}$ can only be positive, two RRAM crossbar arrays are required to represent a matrix with both positive and negative parameters. The input voltage vectors of the positive RRAM crossbar array and the negative RRAM crossbar array should be $\vec{V_i}$ and $-\vec{V_i}$, respectively. The relationship between the input and output voltage vectors can be expressed as:

$$\begin{aligned} \vec{V_o} &= C^+ \cdot \vec{V_i} + C^- \cdot -\vec{V_i} \\ &= (C^+ - C^-) \cdot \vec{V_i} \\ &= C \cdot \vec{V_i} \end{aligned} \tag{4}$$

where C^+ and C^- are the matrices represented by the positive and negative RRAM crossbar arrays as described in Eq. (2) & (3).

III. Design Challenge Discussion

In this section, the nonlinear I-V relationship of RRAM devices and the interconnect resistance are studied. Especially, the sneak path problem [16] when RRAM crossbar array is used as a memory bank will not be a major problem when it is used for computation. To further explain, the sneak path problem occurs only in memory applications when one word line and one bit line are selected for each write or read operation and the unselected lines will have negative impact on the accuracy of output signals. In matrix-vector multiplication applications, all the lines will be selected and the sneak path problem will be eliminated.

As the goal of this paper is to explore design methodologies for efficient computing systems based on RRAM crossbar array, the computation error rate in different cases should be one of the major considerations. The computation error rate of output voltage can be defined as:

$$\epsilon = \max |\frac{V_{actual} - V_{theoretical}}{V_{theoretical}}| \times 100\% \tag{5}$$

where $V_{theoretical}$ is calculated by Eq. (2). Other performance, such as the operating speed of the crossbar array, is also analyzed in this section.

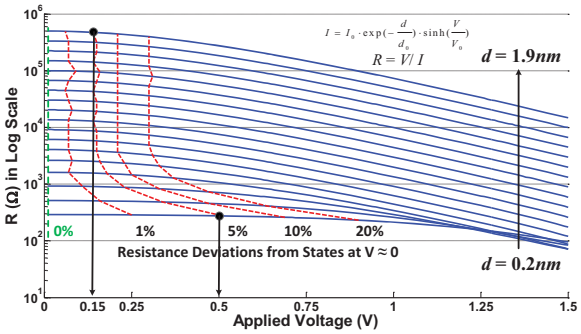

Fig. 2. The RRAM resistance states under different tunneling gap distance (d) and different applied voltage (V). The two vertical lines intersect the tilted dotted line with 2 points, representing the same voltage deviation (5%) from approximate linear resistance state at different (d) with distinct applied voltage. Both the tunneling gap distance d and the applied voltage (V) should be limited to realize an approximate linear resistance state at $V \approx 0$ for a better computation result.

A. Impact of Nonlinear Characteristics of RRAM Devices

As shown in Eq. (1), the I-V relationship of RRAM devices is nonlinear. However, the resistance states of RRAM devices should be constant to represent a specific matrix stably when the RRAM devices are used to realize the matrix-vector multiplication. Therefore, to confine the resistance deviations of RRAM devices, the range of the voltage applied on the RRAM devices should be limited. According to Eq. (1), the linearity of RRAM devices is mainly determined by the term $\sinh(\frac{V}{V_0})$. The RRAM device comes to an ideal linear resistance state when $V \approx 0$:

$$\sinh(\frac{V}{V_0}) \sim \frac{V}{V_0} \tag{6}$$

Fig. 2 illustrates the resistance states of an RRAM device under different tunneling gap distance (d) and different applied voltages (V). The tilted dotted line tracks the maximum voltage that could be applied on an RRAM device under a specific maximum deviation from the approximate linear resistance state at $V \approx 0$. For example, a voltage of $0.5V$ will cause a 5% resistance deviation for $d = 0.2nm$. Considering the same (5%) resistance deviation, the voltage is limited to the range of $\sim 0.15V$ for $d = 1.9nm$. These results demonstrate that the RRAM resistance states vary with the applied voltage and both d and V have influence on the stability of the RRAM resistance states. Since Ohmic current dominates in the low resistance state while tunneling current dominates in the high resistance state, a smaller RRAM resistance state with a smaller tunneling gap distance d will result in a more linear I-V relationship under different voltages. **Therefore, in order to realize a more linear I-V relationship of RRAM devices, both the RRAM resistance state (the tunneling gap distance d) and the applied voltage (V) should be confined.**

B. Impact of Interconnections

As the technology node continues to scale down, the parasitic parameters induced by interconnects in crossbar structure can exert negative influence on the performance of the circuit. In this paper, two major impacts are studied: the RC delay and the interconnect resistance.

RC delay may have a negative impact on the operating speed of RRAM crossbar array-based computation [17]. However, the RC

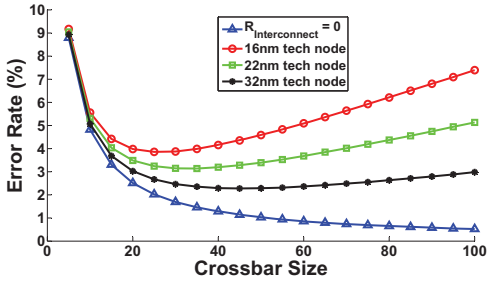

Fig. 3. Worst case computation error rates (ϵ) of RRAM crossbar arrays with different crossbar array sizes ($N \times N$) and different technology node. The RRAM resistance states are calculated at $V \approx 0$.

delay for RRAM crossbar array is trivial ($\sim 10ps$ according to our simulation results) when the wire length between two adjacent junctions is around tens of nanometers for a 100×100 RRAM crossbar array. **Therefore, the RC delay is not a major consideration of the RRAM crossbar array-based computing system design. The design should focus on the performance of peripheral circuits which may significantly impact the operating speed.**

In order to analyze the impact of interconnect resistance on output voltage computation accuracy, a SPICE simulation in the worst case scenario is conducted as a corner case to guarantee the computation accuracy in normal cases. A worst case scenario is defined that all the input voltages of the RRAM crossbar array are of the same amplitude and the worst result can be reflected by the output port which is farthest away from the input ports, while all the RRAM devices are in the lowest resistance states R_{ON}. The load resistance (R_S) is set to $5k\Omega$ and the lowest resistance states of RRAM devices (R_{ON}) is set to $1k\Omega$. The amplitude of input voltages is set to $0.9V$. The crossbar size is varied from 5×5 to 100×100 and the computation error rate is tested as defined in Eq. (2) under different technology nodes. The interconnect resistance between two adjacent junctions is 4.53Ω, 2.97Ω, and 1.55Ω, respectively, for a $4F^2$ RRAM crossbar structure under $16nm$, $22nm$, and $32nm$ technology node according to the International Technology Roadmap for Semiconductors 2013 [10]. An ideal case without any interconnect resistance is also simulated as a comparison.

The results are demonstrated in Fig. 3. When the interconnect resistance is neglected, the computation error rate decreases with the rise of crossbar size $N \times N$. To be specific, the equivalent resistance of the N shunt RRAM devices in a column will drop while the load resistance in that column remains the same. The decreased voltage applied on the RRAM devices will result in better linearity, making the crossbar array represent the matrix more accurately as described in Eq. (3). Therefore, the computation accuracy increases with the crossbar size. However, when the interconnect resistance is taken into consideration, the computation error rate will decrease at the beginning and finally increase due to the voltage drop on the interconnect resistance. Therefore, under the interaction of the nonlinearity of RRAM devices and interconnect resistance, there will be an optimal crossbar size $N \times N$ for each technology node in the worst case scenario, and the optimal crossbar size will shift slightly as the technology node scales down. This result implies that the nonlinearity of RRAM devices and interconnect resistance should be considered together to realize a better implementation of the matrix-vector multiplication operations.

IV. Technological Exploration Flow of RRAM Crossbar Array

In order to overcome the impact of nonideal factors, we describe the proposed technological exploration flow of RRAM crossbar array and achieve better trade-off among performance, energy and reliability. Fig. 4 demonstrates the overview of the proposed flow.

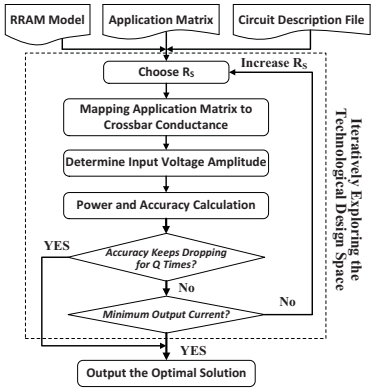

Fig. 4. Proposed Technological Exploration Flow of RRAM Crossbar Array. The flow includes the technology node and load resistance configuration, the algorithm of matrix mapping to crossbar array and a consideration on the trade-off between power and performance.

The flow consists of four stages: (1). Determine the crossbar size and technology node according to characteristics of the application; (2). Choose a proper initial R_S to reduce the impact of interconnect resistance; (3). Map the application matrix C to the RRAM conductance matrix G robustly; and (4). Iteratively explore the technological design space and optimize the performance, energy and reliability of the system.

A. The First Stage: Determine the Technology Node

Given an application, the crossbar array size is constrained by the characteristics of the application. As the interconnect resistance has negative impact on the computation accuracy of RRAM crossbar array, the technology node should be scaled up to support applications that require a large crossbar array or a high computation accuracy. The scaling down of technology node will shrink the area of RRAM crossbar array. There may exist a trade-off between the area and computation accuracy. After the setup of crossbar size and technology node, device level parameters can be further configured as discussed in the next stage.

B. The Second Stage: Choice of R_s

The value of R_S needs to be determined along with R_{ON} since they influence the voltage applied on RRAM devices together. Theoretically, when R_S increases or R_{ON} decreases, the voltage applied on the RRAM devices will decline. As discussed in Section III-A, a smaller applied voltage will result in better linearity of RRAM devices and better computation accuracy. In order to study the impact of R_S and R_{ON} on output voltage computation accuracy, a simulation is conducted in the worst case scenario as defined in III-B. In the experiment setup, we vary R_{ON} from 500Ω to $5k\Omega$ and R_S from $1k\Omega$ to $11k\Omega$ with a 50×50 crossbar size and under $22nm$ technology node.

The value of R_S needs to be determined considering R_{ON} since they influence the linearity of RRAM devices together. Theoretically, when R_S increases or R_{ON} decreases, the voltage applied on the RRAM devices will decline. As discussed in Section III-A, a smaller applied voltage will result in a better linearity of RRAM devices and better computation accuracy. However, a smaller R_{ON} can also lead to a more serious impact of the interconnect resistance. The impact of R_{ON} on the computation accuracy is hard to predict. In order to better study the impact of R_S and R_{ON} in the worst case scenario as defined in III-B, where all the RRAM devices are set to R_{ON}, a simulation is conducted. The crossbar size is set to 50×50 and the amplitude of input voltages (which are the same) are set to $0.9V$ ($\sim 0.1V$ will be applied on the RRAM devices). The technology node is set to $22nm$. We vary R_{ON} from 500Ω to $5k\Omega$ and vary R_S from $1k\Omega$ to $11k\Omega$.

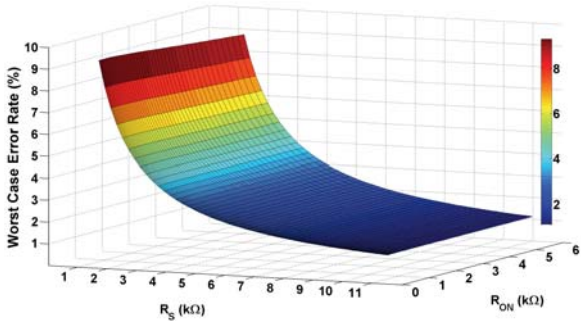

Fig. 5. Computation Error Rates of RRAM crossbar array with different R_S and R_{ON}. The simulation results demonstrate that the computation error rate decreases exponentially with R_S, while R_{ON} has little impact on the computation accuracy. Therefore, the technological exploration flow of RRAM crossbar array for matrix-vector multiplication should focus on the choice of R_S.

The simulation results are illustrated in Fig. 5. It demonstrates that the computation error rate decreases exponentially with the rise of R_S. Compared with R_S, the computation accuracy improves $< 1\%$ when R_{ON} is varied from 500Ω to $5k\Omega$ under the same R_S. This result indicates that R_{ON} has little impact on the computation accuracy. Therefore, the choice of R_{ON} can be neglected for convenience, and the technological exploration flow should focus on the choice of R_S. To be specific, the simulation results illustrated in Fig. 5 can serve as a look-up table and the technological exploration flow will first choose a proper initial R_S to satisfy the worst case and reduce the impact of interconnect resistance. In addition, since the application performance is also influenced by the practical resistance distribution of RRAM devices, a larger R_S cannot guarantee a better computation accuracy. A smaller initial R_S can be used and the optimal choice of R_S can be achieved by iteratively exploring the technological design space in the next stages of the technological exploration flow.

C. The Third Stage: Map Matrix Parameters to RRAM Device Conductivities Robustly

The conductance states of RRAM devices in the crossbar array must be configured properly to realize the multiplied matrix C. However, as shown in Eq. (3), $c_{k,j}$ not only relies on the conductivity of the corresponding RRAM device $g_{k,j}$, but also depends on all the RRAM device conductance states in the same jth column in the crossbar array. In order to realize a one-to-one mapping between matrix C and the conductance matrix of the RRAM crossbar array, some previous work proposed a few simple and fast approximations to the mapping problem like [7]:

$$g_{k,j} = c'_{k,j} \cdot (g_{ON} - g_{OFF}) + g_{OFF} \tag{7}$$

When:

$$g_s \gg (g_{ON} - g_{OFF}) \cdot \sum_{l=1}^{N} c'_{k,l} \tag{8}$$

Eq. (3) can be approximated to:

$$c_{k,j} \approx c'_{k,j} \cdot \frac{g_{ON}}{g_s} = c'_{k,j} \cdot g_{ON} \cdot R_s \tag{9}$$

where $c_{k,j}$ is the matrix parameter of a specific application. g_{ON} and g_{OFF} are the maximum and minimum conductance states of the RRAM devices in the crossbar array.

The above equation demonstrates a linear one-to-one mapping between matrix C and the RRAM conductance matrix G when g_s is determined. However, the precondition of the approximation may be difficult to be satisfied and may decrease computation accuracy. For example, $R_{ON} \approx 1k\Omega$ for a physical RRAM device [11], [18] and $\sum_{l=1}^{N} c'_{k,l} \approx 5$ for a 256×256 RRAM crossbar array [7]. And according to Eq. (8), R_s should be $\sim 100\Omega$. However, as described in Section IV-B, such a small R_s will lead to a large computation error because of the interconnect resistance and the nonlinearity of RRAM devices.

In this work, we propose a numerical iteration algorithm to map the matrix C to the conductance matrix G without any approximation, which improves the computation accuracy of RRAM crossbar array.

Eq. (3) can be expressed as:

$$g_{k,j} - c_{k,j} \cdot \sum_{l=1}^{N} g_{k,l} = g_s \cdot c_{k,j} \tag{10}$$

All of the RRAM devices in the kth column in the crossbar array can form a system of linear equations of N variables as together:

$$\begin{bmatrix} 1-c_{k,1} & -c_{k,1} & \cdots & -c_{k,1} \\ -c_{k,2} & 1-c_{k,2} & \cdots & -c_{k,2} \\ \vdots & \vdots & \ddots & \vdots \\ -c_{k,N} & -c_{k,N} & \cdots & 1-c_{k,N} \end{bmatrix} \begin{bmatrix} g_{k,1} \\ g_{k,2} \\ \vdots \\ g_{k,N} \end{bmatrix} = \begin{bmatrix} g_s \cdot c_{k,1} \\ g_s \cdot c_{k,2} \\ \vdots \\ g_s \cdot c_{k,N} \end{bmatrix} \tag{11}$$

The accurate conductance states of RRAM devices ($g_{k,j}$) can be achieved by solving the above equations when the matrix parameters $c_{k,j}$ are provided. However, several constraints must be considered to guarantee the solved conductance states can be realized by physical RRAM devices. The first constraint results from the range of conductance states that can be realized by physical RRAM devices. Supposing the minimum and maximum conductance states of RRAM devices in a crossbar array are g_{OFF} and g_{ON}, respectively. The parameters $c_{k,j}$ must be of the following range to enable all the solved $g_{k,j}$ are within the range between g_{OFF} and g_{ON}.

$$\chi_{min} \le c_{k,j} \le \chi_{max} \tag{12}$$

$$\chi_{min} = \frac{g_{OFF}}{g_s + g_{OFF} + (N-1)g_{ON}} \tag{13}$$

$$\chi_{max} = \frac{g_{ON}}{g_s + g_{ON} + (N-1)g_{OFF}} \tag{14}$$

where χ_{max} and χ_{min} are the maximum and minimum matrix parameters that can be represented by a physical RRAM crossbar array.

Moreover, as described in Eq. (4), two crossbar are required to represent a matrix with both positive and negative parameters. In order to satisfy the condition described in Eq. (12), Eq. (4) should be revised to:

$$\hat{C} = \hat{C}^+ - \hat{C}^- = \alpha[(C^+ + \Delta) - (C^- + \Delta)] \tag{15}$$

where:

$$c^+_{k,j} = \begin{cases} c_{k,j} & c_{k,j} > 0 \\ 0 & c_{k,j} \le 0 \end{cases} \tag{16}$$

$$c^-_{k,j} = \begin{cases} -c_{k,j} & c_{k,j} < 0 \\ 0 & c_{k,j} \ge 0 \end{cases} \tag{17}$$

α and Δ are parameters to map $\hat{C}^+$ and $\hat{C}^-$ to the range described in Eq. (12). The choice of α and Δ can be achieved by exhausted search. In order to reduce the search space, a restriction of α and Δ is required. We set $c_{max} = \max(|c_{k,j}|)$. According to Eq. (12)-(15), the constraints of α and Δ can be expressed as:

$$\frac{\chi_{min}}{\alpha} \le \Delta \le \frac{\chi_{max}}{\alpha} - c_{max} \tag{18}$$

$$\alpha \le \frac{\chi_{max} - \chi_{min}}{c_{max}} \tag{19}$$

Finally, Algorithm 1 demonstrates the steps of mapping the matrix C to the conductance matrix G^+ and G^-. $Line\ 1 \sim 4$ in the algorithm are used to set up parameter constraints. $Line\ 7 \sim 8$ & $10 \sim 11$ calculate candidate G^+ and G^-. $Line\ 9$ & 12 check the feasibility of candidate solutions.

Algorithm 1: Robust Parameter Mapping Algorithm

Input: C, g_{ON}, g_{OFF}, g_s, $SearchStep$
Output: G^+, G^-

Calculate C^+ and C^- according to Eq. (16);
Calculate χ_{min} and χ_{max} according to Eq. (13)-(14);
Calculate α_{max} according to Eq. (19);
Calculate Δ_{max} and Δ_{min} according to Eq. (18);
for $\alpha = \alpha_{max} : -SearchStep : 0$ **do**
  for $\Delta = \Delta_{min} : SearchStep : \Delta_{max}$ **do**
    $\hat{C}^+ \leftarrow \alpha(C^+ + \Delta)$;
    Calculate G^+ by solving the equation set in Eq. (11)
    if $g_{OFF} \leq G^+ \leq g_{ON}$ **then**
      $\hat{C}^- \leftarrow \alpha(C^- + \Delta)$;
      Calculate G^- by solving the equation set in Eq. (11)
      if $g_{OFF} \leq G^- \leq g_{ON}$ **then**
        return G^+ *and* G^-
      end
    end
  end
end
return *'Bad Parameters'*

D. The Forth Stage: Trade-off among Performance, Energy and Reliability

The proposed technological exploration flow iteratively tests the performance of different parameters and tracks the optimal point. To be specific, the technological exploration flow will first choose a proper initial R_S as discussed in Section IV-B. The selected R_S should be a small one to guarantee the amplitude of output current. Afterwards, the technological exploration flow will calculate the corresponding G^+ and G^- according to the selected R_S and the proposed robust mapping algorithm. The calculated parameters will be used for simulating the detailed performance of the RRAM crossbar array-based computing systems. As a larger R_S may lead to a better computation accuracy, less energy consumption but smaller amplitude of output current, the technological exploration flow will keep increasing R_S gradually to track the change of the system performance, energy and reliability. The exploration of the design space will stop once the output current becomes too small. In addition, the rise of R_S can only guarantee that the computation accuracy increases in the worst case. The input pattern and RRAM resistance state distribution may lead to a worse computation accuracy for a larger R_S. Therefore, the exploration of the design space will also stop when the computation accuracy begins to decrease continuously for a period of time. Finally, by comparing all the tracked solutions, the technological exploration flow is able to provide a solution with better trade-off among performance, energy, and reliability.

V. Experimental Results

In this section, we use the support vector machine (SVM) as a case study to demonstrate the performance of the proposed technological exploration flow.

Support Vector Machine (SVM) is one of the most crucial machine learning algorithms [19] with considerable matrix-vector multiplication workload. Supposing the data can be represented as x, SVM focuses on learning the hyperplane $\vec{w}$ with max-margin to distinguish x and other data. The decision of the class of x is determined by the sign of calculating $\vec{w}^T x + b = \vec{w'}^T \hat{x}$, where $\hat{x} = [1; x]$ and $\vec{w'} = [b; w]$. Since many hyperplanes $\vec{w}$ can form a matrix W together, the major operation of a SVM is the matrix-vector multiplication. Therefore, we use the RRAM crossbar array and the proposed technology exploration technological exploration flow to implement a SVM and test its performance.

A. Experiment Setup

In our experiment, the MNIST dataset is used to test the performance of RRAM-based SVM. MNIST is a widely used dataset with more than 60,000 handwritten digits for optical character recognition. In our experiment, we choose 20,000 examples of handwritten digits of '0' $\sim$ '9' to train the SVM. We extract a 49-dimension feature through principal component analysis [20] from the original 28×28 images. In other words, the dimension of input data $\hat{x}$ is 50 when one dimension for the offset b is considered. As there are ten classes of handwritten digits in the MNIST dataset, we train 10 different SVMs to distinguish only one digit from others. The recognition accuracy of SVM trained on CPU is 94%. And the size of the combined matrix W of 10 SVMs is 50×10. We realize this matrix with a 50×50 RRAM crossbar array. All the other 40 output ports are regraded as virtual nodes whose states will not be considered. The unused RRAM devices in the crossbar array are set to the highest resistance states to reduce the extra energy consumption and negative impact. 5,000 other examples in the MNIST dataset are used to test the performance of RRAM-based SVM. The maximum amplitude of input voltage is set to $1V$ to achieve better linearity of RRAM devices. Most of the input voltages applied on the RRAM devices are around tens to hundreds of millivolt. A current comparator is used to select the port with the highest output current and provide the recognition results. The simulation results are provided in Table I. Some comparisons are made between the proposed technological exploration flow and the method based on [7] under different technology nodes.

B. Performance of Matrix Mapping Algorithm

We first compare the proposed matrix mapping algorithm with the one proposed in [7] under the same technology node. The experiment results demonstrate that both algorithms work well when R_S is very small ($R_S = 100\Omega$). However, as discussed in Section IV-B, such a small R_S will lead to bad computation accuracy because of the interconnect resistance. Only $\sim 80\%$ recognition accuracy is achieved in this situation. As for the cases with a larger R_S of $3k\Omega$, the recognition accuracy of the proposed technological exploration flow significantly increases to $> 90\%$, while a dramatic decrease from 90% to 9% is observed for the previous method. These results demonstrate the approximation used in the previous work doesn't work well for a larger R_S. And the proposed method is robust since there's no approximation used in the mapping algorithm.

C. Impact of R_S and Interconnects

We also increase R_S to $10k\Omega$ to test the impact of R_S on the RRAM-based SVM performance. We first fix the technology node to test the impact of R_S. Compared with the cases when $R_S = 3k\Omega$, the recognition accuracy doesn't increase but drops from 93% to 86%. The reason lies in that a different R_S will lead to different RRAM conductance matrix. The RRAM conductance matrix at R_S may be affected more seriously by the variation of RRAM resistance states and the interconnect resistance. Such results verify the discussion in Section IV-D that a larger R_S is not necessary to lead to a better computation accuracy in practical machine learning tasks instead of the worst case. Then, we vary the technology node of interconnection from $16nm$ to $32nm$ fixing R_S. The results demonstrate that a lower interconnect resistance is beneficial to the recognition accuracy for RRAM-based SVM.

D. Robustness of RRAM Crossbar Array

The above results demonstrate that the RRAM-based SVM works well under ideal conditions. However, several nonideal factors may influence the RRAM-based SVM performance. In this section, we discuss the impact of device variation, signal fluctuation and resistance resolution to test the robustness of the RRAM-based SVM.

TABLE I
EXPERIMENT RESULTS OF RRAM-BASED SVM WITH DIFFERNET PARAMETERS

Map Algm	Tech Node	$R_S(\Omega)$	Signal Fluctuation (%)	Device Variation (%)	Accuracy (%)	Improve (%)	Power (mW)	Savings (%)
[7]	$22nm$	100	0	0	82	-	1.96	-
	$22nm$	3k			9	-89.02	1.93	2.02
	Ideal	1			90	9.76	3.00	-52.73
Proposed	$22nm$	100	0	0	83	1.22	4.07	-106.94
	$22nm$	3k			93	13.41	2.02	-3.04
	$16nm$	3k			90	9.76	1.97	-0.40
	$16nm$	10k			83	1.22	1.40	28.99
	$22nm$	10k			86	4.88	1.42	27.64
	$32nm$	10k			91	**10.98**	1.45	**26.40**
	$22nm$	3k	0	5	90	9.76	2.11	-7.26
	$22nm$	3k	0	10	74	-9.76	2.13	-8.36
	$22nm$	3k	0	20	53	-35.37	2.62	-33.26
	$22nm$	3k	5	0	92	12.20	2.03	-3.33
	$22nm$	3k	10	0	90	9.76	2.11	-7.59
	$22nm$	3k	20	0	87	6.10	2.07	-5.51

1) Impact of Device Variation: The device variation represents the deviation of resistance or conductance state caused by the fluctuation of tunneling gap distance (d). Just as mentioned in Eq. (1), the tunneling gap distance (d) has an exponential relationship with the RRAM resistance state. Therefore, the device variation may have drastically impact on the RRAM-based computing system performance. We test the performance of RRAM-based SVM with different maximum deviation of 5%, 10%, and 20%, respectively. The simulation results verify the above hypothesis and the recognition accuracy significantly drops from 90% to only 53%. The RRAM-based SVM is very sensitive to the variation of tunneling gap distance.

2) Impact of Signal Fluctuations: The electrical noise from the input ports will lead to input signal fluctuation. Here we simulate the performance of RRAM-based SVM under different fluctuations of input signals. The results show that the proposed RRAM-based SVM is robust to the signal fluctuations. For example, a 10% variation of the input signal only reduces the recognition accuracy from 92% to 90%. These results demonstrate that the RRAM-based SVM is able to work in the environments with larger signal fluctuations

3) Impact of Resistance Resolution: Bit-level represents the number of bits that can be represented by an RRAM device when building memory architecture. For example, a 3-bit RRAM device will have $2^3 = 8$ levels of resistance states to represent different binary values. We test the performance of SVM based on different bit-levels. The simulation results are illustrated in Fig. 6. The simulation results show that the bit-level requirement for SVM on MNIST pattern recognition task is not strict. A 4-bit RRAM device is able to realize a recognition accuracy of more than 90%. Such results demonstrate that the RRAM Crossbar based SVM will be quite easy to be physically realized.

VI. CONCLUSIONS

In this paper, we study the impact of a wide range of parameters and propose a technology exploration flow to configure these parameters to achieve a better trade-off among performance, energy and reliability for RRAM crossbar array-based computing system design. We first analyze the impact of the nonlinear I-V relationship of RRAM devices, the interconnects, and other device parameters on the performance of RRAM crossbar array. In order to overcome the impact of these nonideal factors, we propose a technological exploration flow of RRAM crossbar array-based computation, including the technology node and load resistance configuration, the algorithm of matrix mapping to crossbar array with considerations on the trade-off between power and performance. We use the MNIST dataset and a linear SVM classifier as a case study to test the performance of the proposed framework. The simulation results show 10.98% improvement of recognition accuracy and 26.4% power reduction compared with previous work [7].

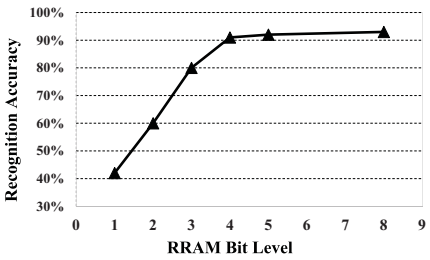

Fig. 6. Recognition Accuracy under Different RRAM Bit Levels.

REFERENCES

[1] J.-W. Jang, S. B. Choi, and V. K. Prasanna, "Energy-and time-efficient matrix multiplication on fpgas," *TVLSI*, 2005.
[2] S. Williams, L. Oliker, R. Vuduc, J. Shalf, K. Yelick, and J. Demmel, "Optimization of sparse matrix–vector multiplication on emerging multicore platforms," *Parallel Computing*, 2009.
[3] C. Xu *et al.*, "Design implications of memristor-based rram cross-point structures," in *DATE*, 2011.
[4] Y. Wang, B. Li, R. Luo, Y. Chen, N. Xu, and H. Yang, "Energy efficient neural networks for big data analytics," in *DATE*, 2014.
[5] M. Hu *et al.*, "Hardware realization of bsb recall function using memristor crossbar arrays," in *DAC*, 2012.
[6] B. Li *et al.*, "Memristor-based approximated computation," in *ISLPED*, 2013.
[7] H. Li *et al.*, "Memristor crossbar-based neuromorphic computing system: A case study," *Neural Networks and Learning Systems, IEEE Transactions on*, 2014.
[8] B. Li *et al.*, "Training itself: Mixed-signal training acceleration for memristor-based neural network." in *ASP-DAC*, 2014.
[9] Y. Deng *et al.*, "Rram crossbar array with cell selection device: A device and circuit interaction study," *TED*, 2013.
[10] ITRS, "International technology roadmap for semiconductors," 2013.
[11] S. Yu *et al.*, "A low energy oxide-based electronic synaptic device for neuromorphic visual systems with tolerance to device variation," *Advanced Materials*, 2013.
[12] K. Seo *et al.*, "Analog memory and spike-timing-dependent plasticity characteristics of a nanoscale titanium oxide bilayer resistive switching device," *Nanotechnology*, 2011.
[13] T. Chang, S.-H. Jo, and W. Lu, "Short-term memory to long-term memory transition in a nanoscale memristor," *ACS nano*, 2011.
[14] Z. Fang *et al.*, "Multilayer-based forming-free rram devices with excellent uniformity," *IEEE Electron Device Letters*, 2011.
[15] H.-S. Wong *et al.*, "Metal–oxide rram," *Proceedings of the IEEE*, 2012.
[16] S. Kannan *et al.*, "Sneak-path testing of crossbar-based nonvolatile random access memories," *TNANO*, 2013.
[17] S.-C. Wong, G.-Y. Lee, and D.-J. Ma, "Modeling of interconnect capacitance, delay, and crosstalk in vlsi," *Semiconductor Manufacturing, IEEE Transactions on*, 2000.
[18] A. Kawahara *et al.*, "Filament scaling forming technique and level-verify-write scheme with endurance over 107 cycles in reram," in *ISSCC*, 2013.
[19] L. Wang, *Support Vector Machines: theory and applications*. Springer, 2005.
[20] C. M. Bishop *et al.*, *Pattern recognition and machine learning*. Springer, 2006.