SCIENCE CHINA
Information Sciences

• RESEARCH PAPER •

February 2019, Vol. 62 022401:1–022401:11
https://doi.org/10.1007/s11432-018-9555-8

Efficient evaluation model including interconnect resistance effect for large scale RRAM crossbar array matrix computing

Runze HAN[1], Peng HUANG[1], Yudi ZHAO[1], Xiaole CUI[2*],
Xiaoyan LIU[1] & Jinfeng KANG[1*]

[1]*Institute of Microelectronics, Peking University, Beijing* 100871*, China;*
[2]*Key Lab of Integrated Microsystems, Peking University Shenzhen Graduate School, Shenzhen* 518055*, China*

Received 24 March 2018/Revised 9 July 2018/Accepted 23 August 2018/Published online 28 December 2018

Abstract Crossbar architecture has been considered as an efficient means to execute a matrix-vector multiplication computation. An efficient evaluation model for this computation including the interconnect resistance effect on the high density resistive random access memmory (RRAM) crossbar array is proposed in this paper. The proposed model considers the interconnect resistance impacts on the columns and rows separately. The simulation results indicate that the computing speed of the proposed model can be boosted by over three orders of magnitude with the computation deviation of 7.7% in comparison with the precise comprehensive model in the 64 kb crossbar array fabricated at the 14 nm technology node. Based on the proposed evaluation model, the impacts of the parameters including nonlinearity and load resistance, on the computation are discussed along with solutions to improve the computational performance.

Keywords crossbar array, evaluation model, interconnect resistance, matrix-vector multiplication, RRAM

Citation Han R Z, Huang P, Zhao Y D, et al. Efficient evaluation model including interconnect resistance effect for large scale RRAM crossbar array matrix computing. Sci China Inf Sci, 2019, 62(2): 022401, https://doi.org/10.1007/s11432-018-9555-8

1 Introduction

Owing to their advantages of simple structure and ultra-high density integration, resistive random access memory (RRAM) crossbar arrays have been widely investigated, not only for emerging non-volatile memories [1–3], but also for logic computing [4–6] and neuromorphic computing applications [7–12]. In neuromorphic computing applications, many operations can be expressed in the form of matrix-vector multiplication and can be executed via an analogue approach in the RRAM crossbar array [8]. When applying an RRAM crossbar array to matrix-vector computations, the input data are converted into voltage pulses applied to word lines, while the output voltages distributed at different bit lines represent the different weighted sum results. Therefore, the computation process can be performed by adopting a parallel approach, which greatly improves the computation speed [13, 14]. Because the resistance state of the RRAM cell represents the matrix weight in the matrix-vector multiplication, and neuromorphic computing is a data-driven application [15], the cell resistances need to be changed frequently during the computation process to obtain the expected result [11, 12]. Therefore, a fast and efficient evaluation of

* Corresponding author (email: cuixl@pkusz.edu.cn, kangjf@pku.edu.cn)

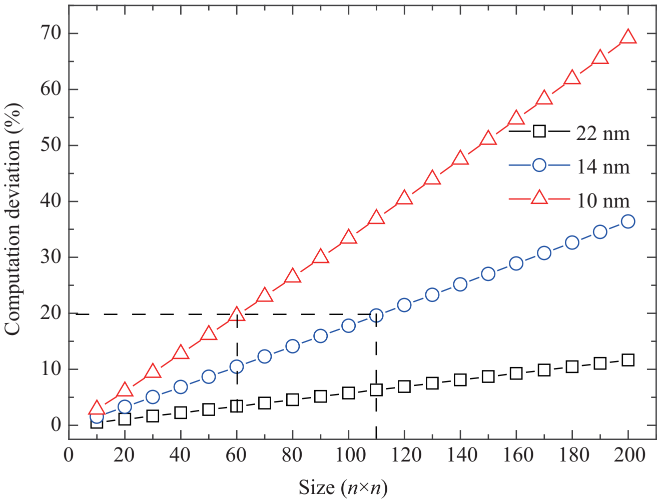

Figure 1 (Color online) Computation deviation of the connection matrix method vs. array size ($n \times n$) at different tech nodes.

the connection between the input voltage and the output voltage pulses is required. A simple and fast connection matrix method without consideration of the interconnect wiring resistance ($R_{\rm wire}$) has been proposed for such applications [11]. However, when the array size increases, the impact of $R_{\rm wire}$ also increases and renders the computation deviation unacceptable.

According to ITRS 2015 [15], the $R_{\rm wire}$ between two adjacent junctions in the crossbar array is 3.56, 10.88, and 20.15 Ω at the 22, 14, and 10 nm technology node, respectively. Figure 1 shows the computation deviation of the connection matrix method at different technology nodes from 10 to 22 nm. The computation deviation rate is defined as follows:

$$\varepsilon = \left| \frac{V_{\rm actual} - V_{\rm theoretical}}{V_{\rm actual}} \right| \times 100\%, \tag{1}$$

where $V_{\rm actual}$ is the actual output voltage and $V_{\rm theoretical}$ is calculated by the specific evaluation model (representing the connection matrix method). The computation deviation of the connection matrix method remains low at the 22 nm technology node with a low $R_{\rm wire}$. However, the deviation exceeds 20% when the crossbar array size is larger than 60 × 60 and 110 × 110 at the 10 nm and 14 nm technology nodes, respectively. A precise comprehensive model that considers the impact of $R_{\rm wire}$ on the characteristics of the RRAM crossbar array has been proposed [16]. With regard to the comprehensive model, a computing matrix with a size of $mn \times mn$ is needed to compute a crossbar array with a size of $m \times n$, and the computing matrix needs to be reversed. Therefore, the comprehensive model requires a great amount of computing resources and time.

In this paper, an evaluation model that includes the wiring resistance effect is proposed to reduce the required amount of computational resources and time. The interconnect resistance effects on the column and row, respectively, were considered. The evaluation model exhibited great improvement in terms of speed in comparison with the comprehensive model, and the computational deviation was maintained at a low level. This paper is organized in three parts. In Section 2, the proposed evaluation model is described from the viewpoint of row and column, respectively. Additionally, considerations and approximations to improve the computing speed while maintaining high accuracy are discussed. In Section 3, the performance of the evaluation model in crossbar arrays with different cell resistance distributions is discussed. In Section 4, the effects of critical parameters on the computation are discussed and solutions to improve the computational performance are suggested with regard to the proposed evaluation model.

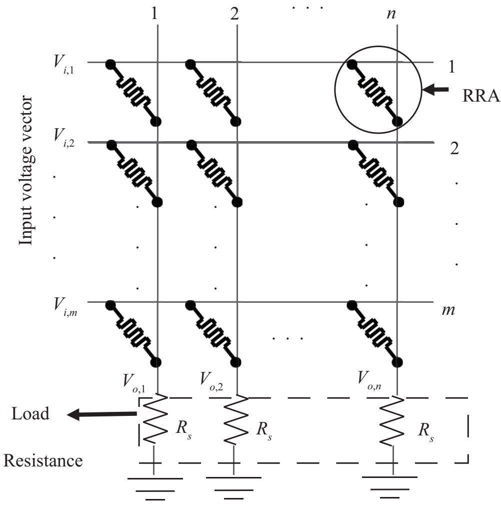

Figure 2 Structure of $m \times n$ RRAM crossbar array. The input voltage vector is applied to the left sides of the rows, and the bottom sides of the columns are connected to the ground through load resistances.

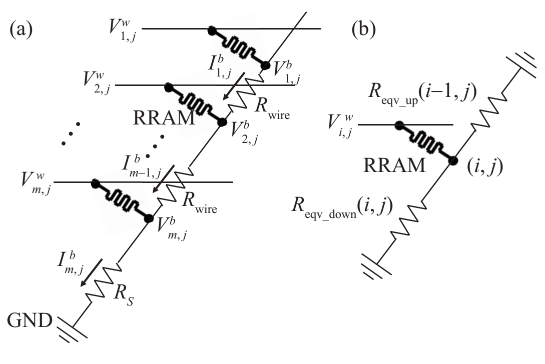

Figure 3 (a) Equivalent circuit diagram of one individual column (bit line); (b) $R_{\text{eqv\_up}}$ and $R_{\text{eqv\_down}}$ are used to represent the equivalent resistance from the (i, j) position to the upper side and down side of the bit line.

2 Model description

The structure of an RRAM-based crossbar array with m rows and n columns is shown in Figure 2. In the crossbar structure, the RRAM device is at each cross-point junction of the word line and the bit line. The input voltage vector $\{V_{i,1}, V_{i,2}, \ldots, V_{i,m}\}$ is applied to the word lines, and the columns are connected to the ground through load resistance (R_s). The current flows in the RRAM device are the product of the applied voltage and internal resistance. Currents flow through devices sharing the same column in the RRAM crossbar array and accumulate at the end of each column. Therefore, the amplitudes of the output voltages fall on the load resistances connected to different columns and represent different weighted-sum results. During the computation process, the right and top sides of the RRAM crossbar array float.

The impact of the interconnect resistance on the computation result can be categorized into three types: impact on rows, impact on columns, and interaction between the impact of R_{wire} on the columns and rows. In this paper, the impact of R_{wire} on the voltage distribution along both the columns and rows in the crossbar array is considered in terms of the tradeoff between computing speed and accuracy. In the proposed evaluation model, the interaction between the impact of R_{wire} on the columns and its impact of R_{wire} on the rows is not considered.

2.1 Impact on columns

The equivalent circuit diagram of one individual column j is shown in Figure 3(a). The voltages at the (i, j) position of the word and bit lines are denoted by $V^w_{i,j}$ and $V^b_{i,j}$, respectively. In the proposed model, the I-V characteristic of the RRAM is considered to be linear; therefore, the circuit is also linear.

According to the superposition principle, the equivalent output voltage responding to multiple stimuli is the sum of the responses caused by each applied stimulus, respectively, which means that all other unapplied voltage sources are connected to the ground. When $V^w_{i,j}$ is applied individually, the output voltage at column j is denoted by $V^i_{o,j}$. The equivalent resistance from the (i, j) position at the bit line to the ground is denoted by $R_{\text{eqv}}(i, j)$, and can be represented by the parallel resistance of $R_{\text{eqv\_down}}(i, j)$ and $R_{\text{eqv\_up}}(i-1, j)$. The equivalent resistance from the (i, j) position to the bottom position at the bit line is denoted by $R_{\text{eqv\_down}}(i, j)$, while the equivalent resistance from the top position to the (i, j) position at the bit line is denoted by $R_{\text{eqv\_up}}(i-1, j)$, as shown in Figure 3(b). Moreover, $R_s(j)$ is the resistance of the load resistance R_s at column j, and $R_{i,j}$ is the resistance of RRAM at the (i, j) position.

The equations describing these relationships are expressed as follows:

$$R_{\text{eqv\_up}}(i,j)=\begin{cases}R_{\text{wire}}+R_{1,j}, & i=1,\\ R_{\text{eqv\_up}}(i-1,j)//R_{i,j}+R_{\text{wire}}, & i\neq 1,\end{cases}\tag{2}$$

$$R_{\text{eqv\_down}}(i,j)=\begin{cases}R_{\text{wire}}+R_{\text{eqv\_down}}(i+1,j)//R_{i+1,j}, & i\neq m,\\ R_s(j), & i=m,\end{cases}\tag{3}$$

$$R_{\text{eqv}}(i,j)=\begin{cases}R_{\text{eqv\_down}}(i,j), & i=1,\\ R_{\text{eqv\_up}}(i-1,j)//R_{\text{eqv\_down}}(i,j), & i\neq 1.\end{cases}\tag{4}$$

The ratio of the output voltage to $V^b_{i,j}$ is denoted by $f_{i,j}$, and $f_{i,j}$ is expressed as follows:

$$f_{i,j}=\begin{cases}1, & i=m,\\ f_{i+1,j}\cdot\dfrac{R_{\text{eqv\_down}}(i+1,j)//R_{i+1,j}}{R_{\text{eqv\_down}}(i+1,j)//R_{i+1,j}+R_{\text{wire}}}, & i\neq m.\end{cases}\tag{5}$$

The output voltage $V^i_{o,j}$ responding to the stimulus at the i-th word line is as follows:

$$V^i_{o,j}{=}V^w_{i,j}\cdot\frac{R_{\text{eqv}}(i,j)}{R_{\text{eqv}}(i,j)+R_{\text{wire}}}\cdot f_{i,j}.\tag{6}$$

According to the superposition principle, $V_{o,j}$ (the output voltage at column j) is expressed by the sum of $V^i_{o,j}$ in the same column, as follows:

$$V_{o,j}=\sum_{i=1}^{m}V^i_{o,j}.\tag{7}$$

2.2 Impact on rows

The equivalent circuit diagram of row k is shown in Figure 4, and $I_{k,j}$ $(1\leqslant j\leqslant n)$ represents the current flows through the RRAM at the (k,j) position. The simplified circuit diagram of row k is shown in Figure 4(b). The equivalent resistance from the node in the (k,j) position at the bit line to the ground can be expressed as follows:

$$\mathrm{RR}_{k,j}=R_{k,j}+(m-k)\cdot R_{\text{wire}}(k,j)+R'_s(k,j).\tag{8}$$

The accumulated resistance of the interconnect resistance from position (k,j) at the bit line to the ground is $(m-k)\cdot R_{\text{wire}}(k,j)$, and $R'_s(k,j)$ is the equivalent resistance of load resistance at column j sensed by node (k,j), and is expressed as follows:

$$R'_s(k,j)=\frac{\sum_{s=1}^{m}1/R_{s,j}}{1/R_{k,j}}\cdot R_s(j).\tag{9}$$

The voltage distribution along the row is expressed by the following equations:

$$\mathrm{RRR}_{k,j}=\begin{cases}\mathrm{RR}_{k,j}+R_{\text{wire}}, & j=n,\\ \dfrac{\mathrm{RR}_{k,j}\cdot\mathrm{RRR}_{k,j+1}}{\mathrm{RR}_{k,j}+\mathrm{RRR}_{k,j+1}}+R_{\text{wire}}, & j\neq n,\end{cases}\tag{10}$$

$$V^w_{k,j}=\frac{\mathrm{RRR}_{k,j}-R_{\text{wire}}}{\mathrm{RRR}(k,j)}\cdot V^w_{k,j-1},\quad j\neq 1.\tag{11}$$

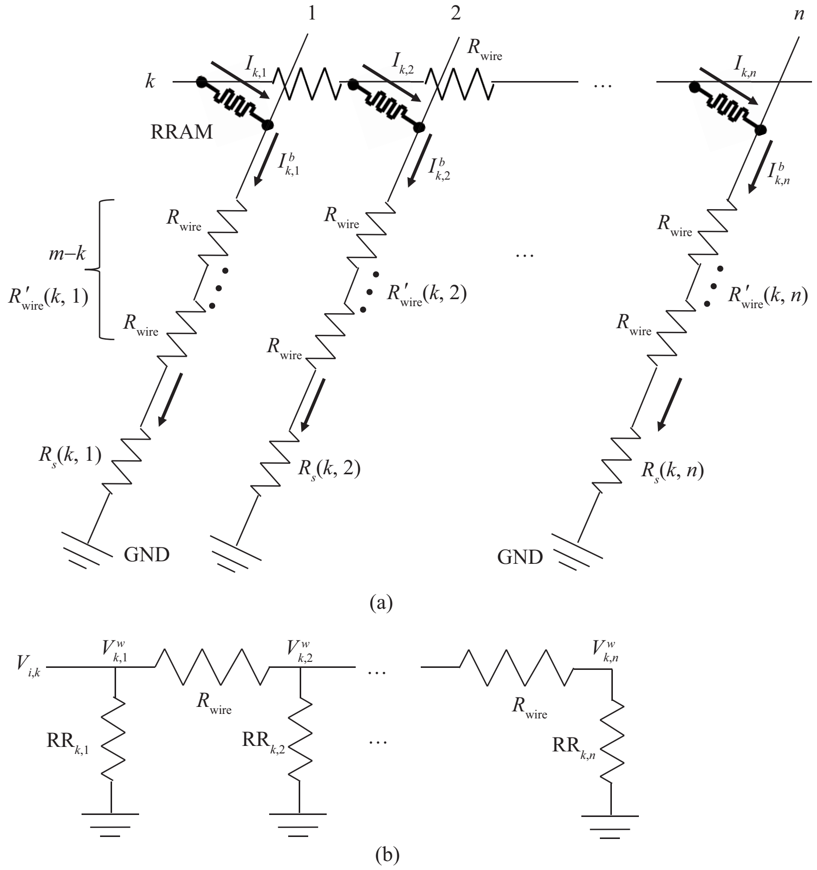

Figure 4 (a) Equivalent circuit diagram of row (word line) k; (b) simplified circuit diagram of row k.

2.3 Without interconnect resistance impact

If the wire resistance is disregarded, i.e., $R_{\rm wrie} = 0$, then the proposed model can be simplified to the computation matrix expressed in Eqs. (12) and (13). This computation matrix is equal to the connection matrix method proposed in [11]:

$$\begin{pmatrix} V_{o,1} \\ V_{o,2} \\ \vdots \\ V_{o,n} \end{pmatrix} = \begin{pmatrix} C_{1,1} & C_{1,2} & \cdots & C_{1,m} \\ C_{2,1} & C_{2,2} & \cdots & C_{2,m} \\ \vdots & \vdots & & \vdots \\ C_{n,1} & C_{n,2} & \cdots & C_{n,m} \end{pmatrix} \cdot \begin{pmatrix} V_{i,1} \\ V_{i,2} \\ \vdots \\ V_{i,n} \end{pmatrix}, \tag{12}$$

$$C_{i,j} = \frac{g_{i,j}}{g_s + \sum_{s=1}^{m} g_{i,s}}. \tag{13}$$

3 Model performance

To evaluate the efficiency of the proposed evaluation model, the output voltages at the farthest and nearest port were calculated using our model. The input voltages had the same amplitude. The condition of the cell resistance distributions in the crossbar array being random and uniform, were respectively considered to verify the feasibility of the proposed model. The parameters used in the simulation are listed in Table 1.

The voltage distributions at the word lines with the uniform and random distributed cell resistance of the 50 × 50 RRAM crossbar array are shown in Figure 5. The voltages at the word lines decreased gradually from the top left corner to the bottom right corner in the crossbar array. This was caused by the degradation of the applied voltage pulses along the lines under the impact of $R_{\rm wire}$. As shown in

Table 1 Simulation parameters used for model validation

Parameter	Value
Interconnect resistance ($R_{\rm wire}$)	10.88 Ω
Low resistance state ($R_{\rm on}$)	10 kΩ
Load resistance (R_s)	5 kΩ
Input voltage amplitude	1 V
Resistance ratio ($R_{\rm on}/R_{\rm off}$)	1000

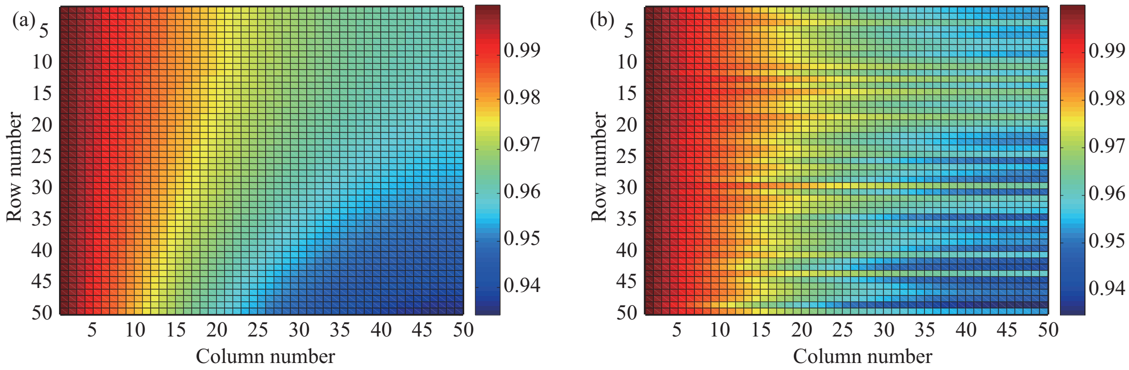

Figure 5 (Color online) Voltage distribution at word lines with (a) uniform and (b) random cell resistance distribution.

Figure 5(a), when the cell resistance distribution was uniform, the voltage at the word lines decreased gradually from the top left corner to the bottom right corner. When the cell resistance distribution was random, the voltage also decreased gradually from the top left to the bottom right corner. However, the voltage at certain rows lying at the top sides was higher in comparison with the uniform distribution, as shown in Figure 5(b), owing to the resistance states being very high at certain rows. Consequently, the voltage level at the word lines of the top side increased.

The proposed model was compared with the connection matrix method [11] and the comprehensive model [16], from the viewpoint of accuracy and speed. Figure 6(a) shows the output voltage (V_o) at the nearest/farthest corner of the uniform cell resistance distribution, as calculated by the three methods. The nearest/farthest output voltage represents the best/worst case where $R_{\rm wire}$ had the smallest/largest influence. The output voltage at the nearest corner obtained using the evaluation model was identical to the comprehensive model. When array size increased, the output voltage V_o increased because the parallel resistance of the RRAM devices decreased. Therefore, the voltage falling on the load resistance decreased. As the array size continued to increase, the impact of $R_{\rm wire}$ and the parallel resistance of RRAM cancelled out. Therefore, the output voltage at the nearest output became stable. The evaluation model and the comprehensive model exhibited this trend sufficiently, whereas the connection matrix method exhibited an increasing output voltage. Additionally, the evaluation model did not exhibit any deviation at the nearest output, while the deviation rate of the connection matrix method increased as the array size increased. The output voltage at the farthest output exhibited the same trend as the nearest V_o, when the array size started to increase. However, as the array size continued to increase, the impact of $R_{\rm wire}$ became dominant, and the voltages applied to the farthest column started to decrease. Therefore, as the array size increased, V_o decreased further. Moreover, the deviation rate of the evaluation model increased as the array size increased. At the farthest output, the deviation rates were 7.7%, 15.7% and 23.5% with an array size of 64 k, 256 k and 1 M, separately.

Figure 6(b) shows the output voltage (V_o) at the nearest/farthest corner with the random cell resistance distribution calculated by the three methods. The output voltage at the farthest output exhibited the same trend with the uniform cell resistance distribution as the array size increased. The deviation rates at the output voltage at the nearest output corner of the evaluation model were still zero. The deviation rates of the farthest output corner were 4.5%, 13.4%, and 21.6% with an array size of 64 k, 256 k, and 1 M, respectively. This demonstrates that the proposed evaluation model is also applicable to a random

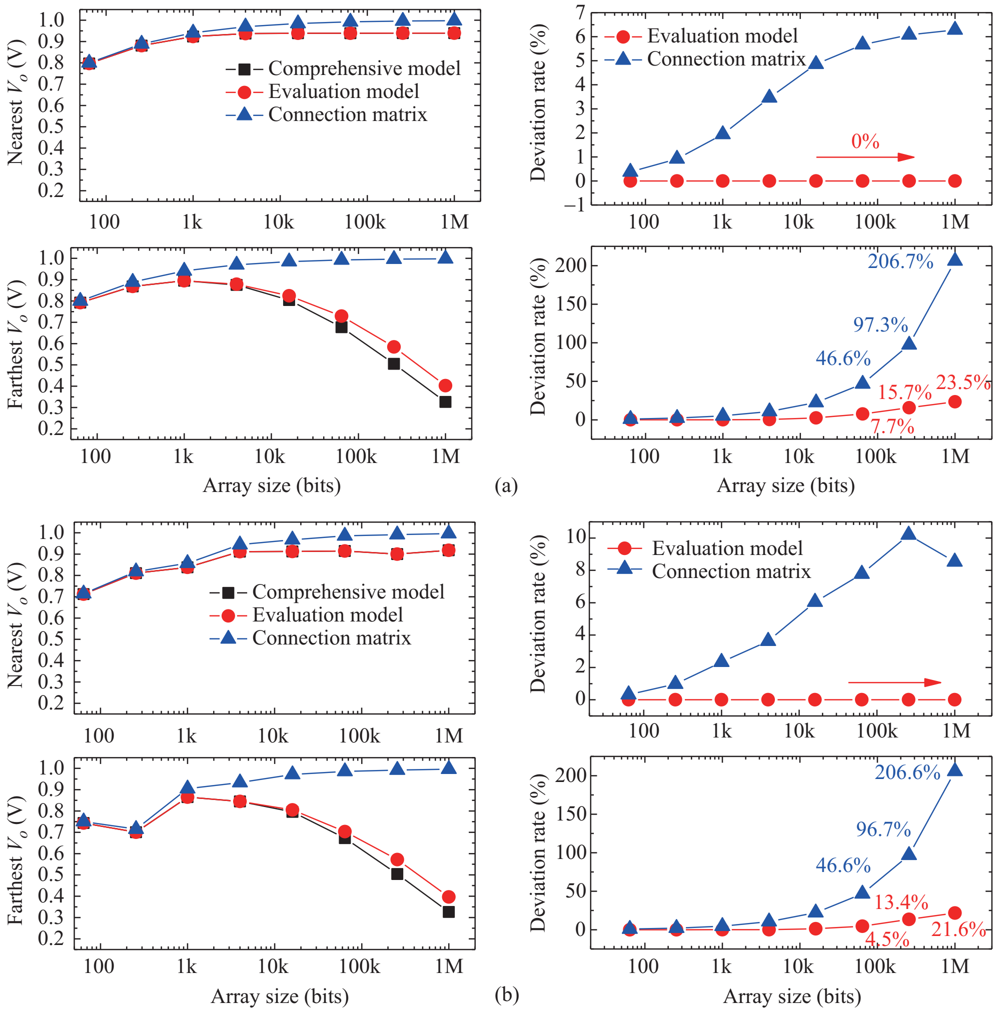

Figure 6 (Color online) Voltage and computation deviation rate at the nearest/farthest output with the (a) uniform and (b) random cell resistance distribution of the three methods vs. the crossbar array size at the 14 nm tech node.

stored data pattern.

Figure 7 shows the comparison of the computing speed using the connection matrix method, the proposed evaluation model, and the precise comprehensive model. When the array size increased from 24 × 24 to 96 × 96, the computation time of the comprehensive model increased dramatically by three orders of magnitude from 0.1 to 41.85 s, while that of the proposed model and the connection matrix method increased by one order of magnitude. The reason for the exponential increase in the comprehensive model's computation time is that, for a 96 × 96 crossbar array, a computation matrix with a size of 9216 × 9216 is required. Moreover, the inverse matrix of such a large scale matrix must be computed by the comprehensive model. Therefore, a great amount of time is consumed. In comparison, to compute the output result of a RRAM crossbar array with a size of 96 × 96, less than 0.1 s is required when using the proposed model, because the impact of $R_{\rm wire}$ on the columns and rows, respectively, is considered in the proposed model. Therefore, only sum and multiply operations are required instead of the time-consuming matrix inverse operation. The results were obtained by these three methods using the same computer with a 2.67 GHz Intel Xeon CPU and 1333 MHz DDR3 DRAM with 47 GB of total memory capacity.

4 Model applications

When using the RRAM crossbar array in the memory application, typically, only one word line and one bit line are selected. The unselected lines will cause sneak current and crosstalk issues. Therefore, the

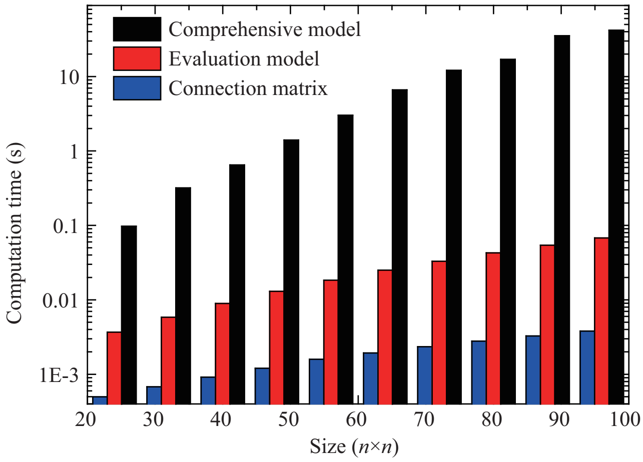

Figure 7 (Color online) Computation time of three methods with different crossbar array sizes ($n \times n$).

system's performances will deteriorated [17]. However, for a computing application, most lines will be selected. Thus, the sneak path problem is not a major concern [12].

4.1 Nonlinearity impact on the computing application

In the proposed model, the RRAM device is treated as a linear element. However, in practical situations, the *I*-*V* characteristics of RRAM devices are nonlinear and considered to be useful for reading out of the memory array [17]. With regard to the computing applications of the RRAM crossbar, the nonlinear *I*-*V* characteristic increases the uncertainty of the computation results. The impact of the nonlinearity was investigated on the basis of the proposed model. In this study, the expressions of RRAM nonlinearity at a low resistance state (LRS) and high resistance state (HRS) were set identical to those reported in [18], and are represented as follows.

For LRS:

$$I = g_{\mathrm{on}} \cdot V, \tag{14}$$

For HRS:

$$I = g_{\mathrm{off}} \cdot \sinh(\alpha \cdot V). \tag{15}$$

In the above equations, g_{on} and g_{off} represent the conductance of the RRAM device in the LRS and HRS, respectively. If the RRAM device is in the LRS, owing to the metallic conduction, the *I*-*V* curves will be linear in most cases. If the RRAM is in the HRS, the conductance will be dominated by a hopping current [19], which indicates a nonlinear characteristic. The α parameter represents the nonlinear characteristic of the RRAM *I*-*V* curves. The *R*-*V* curves of the RRAM with different α are shown in Figure 8.

When α increases, the decrease of resistance in the HRS becomes steeper as the applied voltage increases. Therefore, the resistance window decreases, which makes it harder to distinguish the HRS and LRS. In the HRS, the resistance change with the change of applied voltage introduces uncertainty to the computation. The change of the farthest output voltage with an array size with different nonlinearities is shown in Figure 9. The input voltage and resistance states of the cells in the crossbar array are set to 1 V and HRS, respectively, to maximize the influence of nonlinearity. As the nonlinearity increases, the resistance state of the RRAM device decreases. Therefore, the more V_o increases, the larger becomes the deviation of the estimated farthest V_o with the actual output voltage. When the array size increases, the parallel resistance of the RRAM cells decreases. Therefore, the voltage falling on the RRAM cells decreases. As shown in Figure 8, when the applied voltage decreases, the resistance change also decreases. Therefore, the deviation of the estimated farthest V_o with actual output voltage also decreases. Decreasing the applied voltage can decrease the influence of nonlinearity, and the analysis is the same when the array size increases.

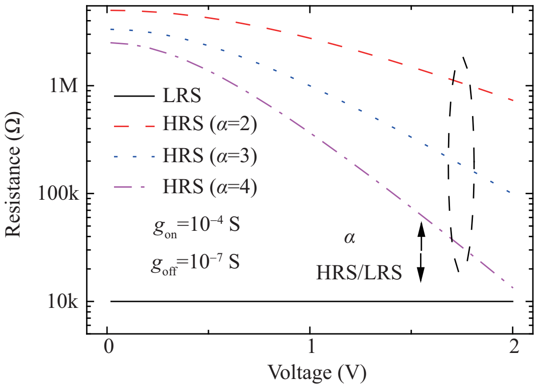

Figure 8 (Color online) Resistance-voltage curves with a different nonlinear factor α; $g_{\rm on}$ is set to 10^{-4} while $g_{\rm off}$ is set to 10^{-7}.

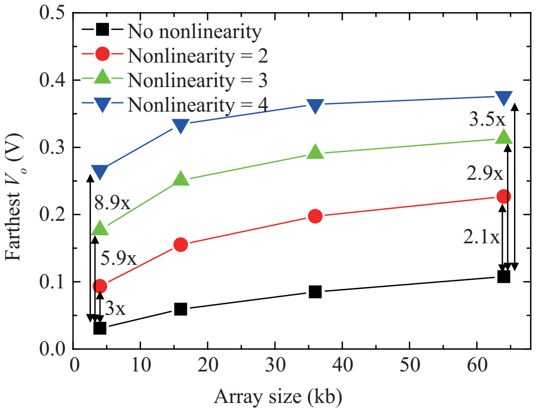

Figure 9 (Color online) Voltage at the farthest output vs. array size with different nonlinearities.

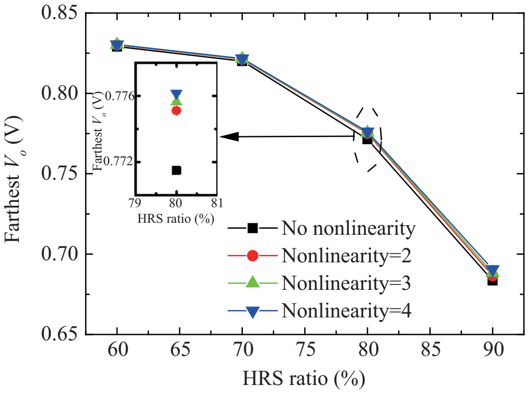

Figure 10 (Color online) Voltage at the farthest output vs. HRS ratio with different nonlinearities. The array size is set to 64 × 64.

The above analysis regarding the impact of nonlinearity is an extreme case. In most cases, the distribution of cell resistances is random. Figure 10 shows the relationship of the farthest V_o and the different HRS ratios obtained with different nonlinearities. The HRS ratio is the ratio of the RRAM cells in the crossbar array in the HRS. The cell resistances in the LRS were set to 10 kΩ. Contrary to the case of the crossbar array with the uniform distributed resistance states, nonlinearity had little impact on the distribution of random resistance states. From the inset figure, we can see that, when the HRS ratio was 80%, the farthest V_o increased the most with the increase in nonlinearity, but only with a limited value. The reason is that, in a random case, the dominant influence of the output voltage is that exerted by the cells in the LRS. Because the resistance of cell in the LRS is linear, the output voltage can be simulated well by using the proposed model. Figure 10 shows that the deviation increased when the HRS ratio increased. However, the deviation was still negligible in the random case.

4.2 Load resistance choice

A simple way to convert the output current into voltage is to use the load resistance R_s that connects the bit line to the ground. In this section, we discuss the influence of R_s on the output voltage.

The effects of load resistance and crossbar array size on the output voltage at the farthest output were analyzed by the proposed model. The simulation results obtained using the evaluation model are presented in Figure 11. As the load resistance R_s increased, owing to the division of voltage, more voltage fell on R_s. Therefore, the voltage at the farthest output increased. When the array size first increases, the parallel resistance equivalent to the cell resistance shared common bit line decreases. Owing to the

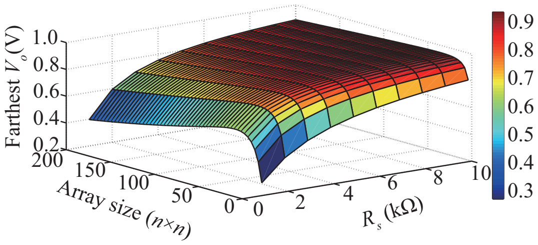

Figure 11 (Color online) Voltage at the farthest output vs. load resistance and crossbar array size. Cell resistance = 10 kΩ, $R_{\rm wire}$ = 10.88 Ω.

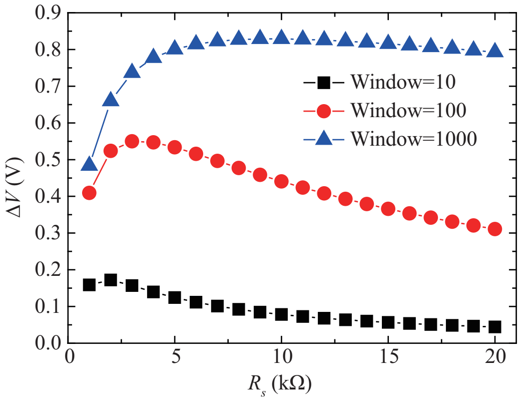

Figure 12 (Color online) Voltage difference vs. load resistance at different resistance windows. $R_{\rm on}$ = 10 kΩ, $R_{\rm wire}$ = 10.88 Ω, the crossbar array size is 100 × 100.

division of voltage, more voltage fell on the load resistance. Therefore, the output voltage increased. However, when the array size continued to increase, the equivalent resistance of the serially-connected $R_{\rm wire}$ increased. Therefore, the impact of the interconnect resistance became dominant. With a larger equivalent resistance from the voltage sources to R_s, the voltage at the farthest output decreased.

The different cell resistance states stored in the RRAM crossbar array represent different information such as different convolution kernels [13]. This resistance difference must be sensed to output the correct convolution result. The direct reflection of the difference of the cell resistance states in one column is the difference of the output voltages. A larger output voltage difference makes the computed results more distinguishable. The voltage difference (ΔV) shown in Figure 12 is defined as the output voltage difference at the farthest output between the conditions under which the cell resistances are in the LRS and HRS. The relationship between ΔV and R_s with different resistance windows is shown in Figure 12. With a fixed load resistance, the output voltage difference at the farthest port became larger as the resistance window increased. The output voltage difference first increased as the R_s continued to increase, and ΔV started to decrease. With a lower/higher R_s, the output voltage was maintained at a low/high level, which rendered the voltage difference indistinct. Therefore, a proper R_s should be chosen to maximize the output voltage difference. When the resistance window increased with a fixed $R_{\rm on}$, the equivalent resistance of the cells also increased. Therefore, a larger R_s is needed to maximize the output voltage difference.

Apart from choosing an appropriate R_s value, a transimpedance amplifier (TIA) can also be used to amplify the output voltage difference [20, 21]. This method can also be simulated by using the proposed method, with the only difference being that the R_s value is substituted with a wiring resistance between the bottom node at the bit line and the TIA, and an amplifier coefficient is multiplied when calculating the output voltage.

5 Conclusion

An efficient evaluation model is proposed for large scale RRAM-based crossbar array matrix computing. The impact of the interconnect resistance on the output voltage in cases with different cell resistance distributions was analyzed using the proposed model. In comparison with the precise comprehensive model, the computing speed of the crossbar array with a size of 96 × 96 improved by three orders of magnitude. The computational deviation of the proposed model was 7.7%, when computing the worst case of a 64 k crossbar array, in comparison with the computational deviation of 48.8% of the connection matrix. The effects of nonlinearity on the RRAM cell and the effects of load resistance on the computation applications were investigated using the proposed model. Lower applied voltages on the

RRAM cells helped to reduce the impact of nonlinearity. These solutions included lower applied voltages on the word line and increased load resistance. Decreasing the nonlinearity of the RRAM cell in the HRS also helped to reduce the impact of nonlinearity. With regard to the sensing margin of the computation, a larger resistance window and an optimal load resistance are required.

Acknowledgements This work was supported by National Natural Science Foundation of China (Grant Nos. 61334007, 61421005), and Shenzhen Science and Technology Innovation Committee (Grant No. JCYJ2017041215-0411676).

References

1 Wong H S P, Lee H Y, Yu S, et al. Metal-oxide RRAM. Proc IEEE, 2012, 100: 1951–1970

2 Hudec B, Hsu C W, Wang I T, et al. 3D resistive RAM cell design for high-density storage class memory-a review. Sci China Inf Sci, 2016, 59: 061403

3 Waser R, Dittmann R, Staikov G, et al. Redox-based resistive switching memories: nanoionic mechanisms, prospects, and challenges. Adv Mater, 2009, 21: 2632–2663

4 Borghetti J, Snider G S, Kuekes P J, et al. 'Memristive' switches enable 'stateful' logic operations via material implication. Nature, 2010, 464: 873–876

5 Yang J J, Strukov D B, Stewart D R. Memristive devices for computing. Nat Nanotech, 2013, 8: 13–24

6 Huang P, Kang J F, Zhao Y D, et al. Reconfigurable nonvolatile logic operations in resistance switching crossbar array for large-scale circuits. Adv Mater, 2016, 28: 9758–9764

7 Hu M, Li H, Chen Y R, et al. Memristor crossbar-based neuromorphic computing system: a case study. IEEE Trans Neural Netw Learn Syst, 2014, 25: 1864–1878

8 Upadhyay N K, Joshi S, Yang J J. Synaptic electronics and neuromorphic computing. Sci China Inf Sci, 2016, 59: 061404

9 Cao J D, Li R X. Fixed-time synchronization of delayed memristor-based recurrent neural networks. Sci China Inf Sci, 2017, 60: 032201

10 Yu S M, Gao B, Fang Z, et al. A low energy oxide-based electronic synaptic device for neuromorphic visual systems with tolerance to device variation. Adv Mater, 2013, 25: 1774–1779

11 Hu M, Li H, Wu Q, et al. Hardware realization of BSB recall function using memristor crossbar arrays. In: Proceedings of the 49th Annual Design Automation Conference, San Francisco, 2012. 498–503

12 Gu P, Li B X, Tang T Q, et al. Technological exploration of RRAM crossbar array for matrix-vector multiplication. In: Proceedings of the 19th Asia and South Pacific Design Automation Conference (ASP-DAC), Chiba, 2015. 106–111

13 Gao L G, Chen P Y, Yu S M. Demonstration of convolution kernel operation on resistive cross-point array. IEEE Electron Device Lett, 2016, 37: 870–873

14 Li H T, Gao B, Chen Z, et al. A learnable parallel processing architecture towards unity of memory and computing. Sci Rep, 2015, 5: 013330

15 Semiconductor Industry Association. International Technology Roadmap for Semiconductors. 2015. https://www.semiconductors.org/main/2015_international_technology_roadmap_for_semiconductors_itrs/

16 Chen A. A comprehensive crossbar array model with solutions for line resistance and nonlinear device characteristics. IEEE Trans Electron Device, 2013, 60: 1318–1326

17 Vontobel P O, Robinett W, Kuekes P J, et al. Writing to and reading from a nano-scale crossbar memory based on memristors. Nanotechnology, 2009, 20: 425204

18 Deng Y X, Huang P, Chen B, et al. RRAM crossbar array with cell selection device: a device and circuit interaction study. IEEE Trans Electron Device, 2013, 60: 719–726

19 Huang P, Liu X Y, Chen B, et al. A physics-based compact model of metal-oxide-based RRAM DC and AC operations. IEEE Trans Electron Device, 2013, 60: 4090–4097

20 Sheridan P M, Cai F X, Du C, et al. Sparse coding with memristor networks. Nat Nanotech, 2017, 12: 784–789

21 Li C, Hu M, Li Y N, et al. Analogue signal and image processing with large memristor crossbars. Nat Electron, 2018, 1: 52–59