Analysis of the Impact of Wire Resistance on Nano-scale Memristor Crossbar Array Implementing Perceptron Neural Network

Son Ngoc Truong
Faculty of Electrical and Electronics Engineering, Ho Chi Minh City University of Technology and Education, Vietnam
sontn@hcmute.edu.vn

Abstract. In this work, the impact of wire resistance in pure memristor crossbar array is mathematically analysed and verified by the circuit simulation. The memristor crossbar without CMOS device is utilized for application of character image recognition, in which wire resistance is presented. Memristor crossbar circuit is analysed separately with respect to wire resistance on vertical line and wire resistance on horizontal line. The result shows that wire resistance on vertical line can be eliminated because they can be self-compensated. The simulation result agrees with the analysis. The variation of output voltage caused by wire resistance less affect the recognition rate of memristor circuit. On the other hand, when wire resistance on horizontal line is assumed to be 2.5Ω. The output voltages are varied remarkably. Such variation of output voltage degrades the recognition rate of memristor crossbar circuit. The interesting phenomenon is also investigated. The column that is close to the first column has less variation of output voltage, whereas the one that is far from the first column has much variation of output voltage. The result can be used in improving the memristor crossbar architecture, which can tolerate the impact of wire resistance. For example, if we want to increase the size of crossbar, we should increase the number of rows, rather than the number of columns.

1. Introduction
Neuromorphic computing system which is inspired from human brain has been considered as an alternative solution to overcome the limitation of von Neumann computer architecture [1]. Neuromorphic refers to a hardware realization of artificial neural network mimicking the functionality of biological brain. For implementing neuromorphic systems, various research activities that are based on CPUs, GPUs, FPGAs, analog circuits, memory circuits, etc. have been carried out both in academia and industry [2]-[9]. Since memristor was experimentally demonstrated in 2008, it has become a potential nano-scale device for the design of neuromorphic computing system [10], [11]. Moreover, memristor can be formed as crossbar array and stacked layer by layer for high-density architecture [12]. More important, the charge-flux relationship of memristor is very close to the behavior of neural synapse [13]. Modeling neural synapse using memristor offers many advantages in term of power consumption and area occupation.

There are several ways to design neural synapse using memristor. The popular structures are the combination of memristor and CMOS devices [14]-[16]. Though such structures exhibit the excellent implementation of neural synapse, they seem to be limited since CMOS device can not be scaled down
more and more, and CMOS is power-hungry. Alternatively, pure memristor crossbar array without CMOS device is more efficient in term of the integration and power consumption [17]-[19]. Memristor crossbar array circuits gain more and more advantage in implementing neuromorphic computing system. Miao Hu et al proposed a crossbar of synaptic array that is composed of plus and minus crossbar array representing plus- and minus-polarity connection matrix [19] for implementation of neural network. The pure memristor crossbar array is very effective, compared to using memristor-CMOS hybrid circuits. To reduce the area, as well as the power consumption more, S. N. Truong proposed a new memristor crossbar array architecture which is composed of a single memristor array and a constant-term circuit. The proposed architecture shows that the power consumption is reduced by 48% and the area is reduced by 50% too [20]. Since then, pure memristor has demonstrated to be effective in applications such as speech recognition and image recognition [21], [22].

In pure memristor array, some amount of voltage drop can be caused by interconnect resistance, also known as wire resistance along row and column lines [23]-[24]. Such resistance degrades the performance of circuit seriously as the size of array is increased more and more [24]. In all of the previous work, the memristor is model using Verilog A, and the crossbar circuit is simulated using Cadence Spectre Simulation [19], [20]. For such approach, the impact of wire resistance is ignored, although it becomes serious problem in practical.

In this work, we analyse the impact of wire resistance to the performance of pure memristor crossbar array, which implements the perceptron neural network. The impact of wire resistance is mathematically analysed and then verified by the circuit simulation. The result emerges that the impact of wire resistance on vertical line can be compensated itself. Wire resistance on horizontal line degrades the performance of memristor crossbar remarkably. However, it is interesting that the variation caused by line resistance on horizontal line is proportional to the length of metal line. The analysis is more useful in improving the reliability of pure memristor crossbar array.

2. Analysis of the impact of wire resistance on crossbar-based synaptic array circuit

The improved memristor crossbar array for low power consumption and small area is shown in Figure 1. Here only one crossbar array is employed, associated with a constant term circuit for implementing negative and positive neural synapse [20].

In Figure 1, \( g_{j,k} \) represents the memristor’s conductance at the crossing point between the \( j^{th} \) row and \( k^{th} \) column. \( V_{\text{IN},j} \) is the input voltage applied to the \( j^{th} \) row. \( V_{\text{C},k} \) is the column-line voltage on the \( k^{th} \) column. In Figure 1, \( V_{\text{C},F} \) enters \( G_F \) that is inverting OP amp with the negative feedback resistor, \( R_F1 \) to generate the reference voltage, \( V_F \). \( V_F \) is propagated to all column lines from \( V_{\text{C},1} \) to \( V_{\text{C},n} \) through \( R_F2 \), as indicated in Figure 1. By applying Kirchhoff current law to the column line, \( V_{\text{C},F} \), we can calculate \( V_F \) with Eq. (1).

Figure 1. The improved memristor-based crossbar architecture with single memristor array and a constant-term circuit for implementing the negative and positive neural synapse [20].
Similarly, we obtain the column voltage as follows.

\[
V_{\theta,k} = - \left[ \sum_{j=1}^{n} \left( R_0 \cdot g_{j,k} \cdot V_{IN,j} \right) + \frac{R_0}{R_{F2}} V_F \right]
\]  

(2)

If we choose \(R_{F1}=R_{F2}\) and substitute \(V_F\) by Eq. (1) the following equation (3) can be obtained [20].

\[
V_{\theta,k} = - \left[ \sum_{j=1}^{n} \left( R_0 \cdot g_{j,k} - \frac{R_0}{R_B} V_{IN,j} \right) \right]
\]  

(3)

If \(- \left( R_0 \cdot g_{j,k} - \frac{R_0}{R_B} \right)\) is defined as synaptic weight of the \(j^{th}\) row and \(k^{th}\) column, \(w_{j,k}\), we can rewrite Eq. (3) with Eq. (4).

\[
V_{\theta,k} = \sum_{j=1}^{n} w_{j,k} V_{IN,j}
\]

where \(w_{j,k} = R_0 \left( \frac{1}{R_B} - g_{j,k} \right) = R_0 \left( \frac{1}{R_B} - \frac{1}{M_{j,k}} \right) \)  

(4)

In Eq. 4, the output voltage is a summation of all inputs, which are modified by their corresponding weights. Equation 4 figures out that each column has the functionality like a perception neuron, which has the output is a summation of all inputs multiplying with the corresponding synaptic weights, as shown in Figure 2(a) [25]. Figure 2(b) demonstrates that each column in memristor crossbar can implement a perceptron neuron, since the output of each column is consistent with the output of each perceptron neuron.

**Figure 2.** A perceptron neuron modeled by a sum of inputs with corresponding weights. (b) a perceptron neural modeled by one column in memristor crossbar.

In Figure 2(b), \(M_{j,k}\) is programmable memristance value of the crossing point between the \(j^{th}\) row and \(k^{th}\) column. Resistor, \(R_B\) is a constant and \(R_B\) is chosen in a range of from low resistance state (LRS) to high resistance state (HRS). Then \(w_{j,k}\) is programmed to be greater or less than \(R_B\) to generate negative or positive synaptic weight.
The output of each perceptron neuron is 0 or 1 decided by the threshold function [25]. The threshold function can be implemented by adding the comparator, C_k, as shown in Figure 1. By doing this, the neuron’s output of k-th column, OUT_k, can be activated or not. V_{REF} is the reference voltage for the comparators from C_1 to C_n. When V_{O,k} is larger than V_{REF}, OUT_k becomes 1. If V_{O,k} is smaller than V_{REF}, OUT_k is decided to be 0, as mathematically shown in Eq. (5).

\[
\text{OUT}_k = \begin{cases} 
1, & \text{if } V_{O,k} \geq V_{REF} \\
0, & \text{if } V_{O,k} < V_{REF} 
\end{cases} 
\]

In memristor crossbar array, the voltage drops along column and row lines is inevitable [23], [24]. It becomes more serious as the size of array is increased more and more. Voltage drop is caused by the line resistance between the interconnections. The line resistance is modeled by the small-value resistor lying on the vertical lines and horizontal lines, as shown in Figure 3.

In this work, we analyse the impact of wire resistance by mathematically calculating the amount of variation of output voltage as wire resistance is taken into account. Here we define the variation of output voltage as a difference of voltage when wire resistance is assumed to be 0 Ω and that one when wire resistance is assumed to be 2.5 Ω.

In Figure 3, wire resistance along horizontal and vertical lines are represented by small-value resistor, r connecting between two crossing points. (a) Wire resistances on horizontal lines are neglected. (b) Wire resistances on vertical lines are neglected.

We analyse the circuit separately with respects to wire resistance on vertical lines and wire resistance on horizontal lines, as shown in Figure 3(a) and (b), respectively. In Figure 3(a), we define V_{b1}, V_{b2} are the amount of voltage of node b_1 and b_2, which are on the first column. In common case, V_{b_j} is the amount of voltage of node b_j on the first column. Similarly, V_{k_j} is the amount of voltage of node k_j which is on the j-th column. Applying Kirchhoff current laws for all nodes in Figure 2(a), V_F and V_{O,k} can be estimated as follows:

\[
\frac{-V_F}{R_{F1}} = \frac{V_{b_{k,u}} - V_{b_{a}}}{R_{g}} + \ldots + \frac{V_{b_{k,j}} - V_{b_{j}}}{R_{g}} + \ldots + \frac{V_{b_{k,1}} - V_{b_{1}}}{R_{g}} 
\]

\[
V_F = -R_{F1} \left( \sum_{j=1}^{k} \frac{V_{b_{k,j}}}{R_{g}} - \sum_{j=1}^{k} \frac{V_{b_{j}}}{R_{g}} \right) 
\]

\[
\frac{-V_{O,k}}{R_0} = \left( V_{b_{k,u}} - V_{b_{a}} \right) g_{a,k} + \ldots + \left( V_{b_{k,j}} - V_{b_{j}} \right) g_{j,k} + \ldots + \left( V_{b_{k,1}} - V_{b_{1}} \right) g_{1,k} + \frac{V_F}{R_{F2}} 
\]

\[
V_{O,k} = -R_0 \left( \sum_{j=1}^{k} V_{b_{k,j}} g_{j,k} - \sum_{j=1}^{k} V_{b_{j}} g_{j,k} + \frac{V_F}{R_{F2}} \right) 
\]

If we assume that R_{F1}=R_{F2}, Eq. 7 can be simplified as follows:
By comparing Eq. (8) and Eq. (4), we can derive the variation of voltage, $\Delta V$ caused by wire resistance on vertical lines as follow.

$$
\Delta V = -\sum_{j=1}^{m} R_0 \frac{V_{ij}}{M_{j,k}} + \sum_{j=1}^{m} R_0 \frac{V_{ij}}{R_g}
$$

(9)

$M_{j,k}$ is memristance which is programmed according to the synaptic weight using Eq. 4. For example, if the synaptic weight, $w_{j,k}$ is negative, $M_{j,k}$ is less than $R_B$, and if $w_{j,k}$ is positive, $M_{j,k}$ is greater than $R_B$. Because the synaptic weight, $w_{j,k}$ obtained from the training process can have either negative or positive values, the variation of voltage, $\Delta V$ in Eq. (9) can be very small due to the compensation of the negative term of $-\sum_{j=1}^{m} R_0 \frac{V_{ij}}{M_{j,k}}$ and the positive term of $\sum_{j=1}^{m} R_0 \frac{V_{ij}}{R_g}$.

Under this assumption, the variation of voltage, $\Delta V$ caused by wire resistance on vertical line can be compensated itself, and less affect to the output voltage.

Next, wire resistance on vertical lines is ignored whereas that one on horizontal lines is considered as shown in Figure 3(b). The input voltage for each column becomes different since the voltage is dropped on the small-value resistors which they pass through. In Figure 3(b), $V_{1(k)}$ is the amount of voltage on the resistor, which is on the first row and between the $k$th and $(k-1)$th column, the amount of voltage applied to first row of the $k$th column is significantly degraded and can be calculated as following equation.

$$
V_{0,k} = \sum_{i=1}^{n} V_{i,k-1} - \sum_{i=1}^{n} V_{i,k} + \sum_{i=1}^{n} V_{i,1} + \sum_{i=1}^{n} V_{i,k}
$$

(10)

Here $V_{IN,1(k)}$ is the amount of voltage applied to the first row of the $k$th column. Then the output voltage of $k$th column, $V_{O,k}$ becomes

$$
V_{O,k} = \left[ \sum_{j=1}^{m} \left( R_0 \cdot g_{j,k} \cdot V_{IN,j(k)} - \frac{R_0}{R_g} V_{IN,j} \right) \right]
$$

(11)

By comparing Eq. (11) and Eq. (3), we obtain the variation of voltage, $\Delta V_k$, caused by wire resistance on horizontal lines

$$
\Delta V_k = \sum_{j=1}^{m} R_0 g_{j,k} V_{IN,j} - \sum_{j=1}^{m} R_0 g_{j,k} V_{IN,j(k)}
$$

(12)

Here, $V_{IN,j(k)}$ can be calculated using Eq. (10), then we obtain $\Delta V_k$, as shown in Eq. (13)

$$
\Delta V_k = \sum_{j=1}^{m} \left( R_0 g_{j,k} \sum_{i=1}^{k} V_{j,i} \right)
$$

(13)

In Eq. (13), the term of $\sum_{i=1}^{k} V_{IN,j(i)}$ is the total voltage that drops on $k$ resistors from the first column to the $k$th column of the $j$th row. Eq. (13) emerges that the amount of voltage applied to the $k$th column is degraded proportionally to distance from that column to the first column. Since the applied voltage is decreased, the output voltage obtained from the output of inverting OP amp can be increased. From this mathematical analysis, we can claim that the column that is close to the first column has less variation of voltage, and the column that is far from the first column has much variation of voltage. More interesting, two adjacent columns can have nearly same variation of voltage caused by wire resistance. The analysis result is very useful for design of memristor crossbar circuit, which can tolerate the effect of wire resistance.

3. Simulation and result
The SPECTRE circuit simulation is performed to verify the mathematical analysis of the impact of wire resistance. The memristor crossbar array circuit in Figure 1 is used for the application of
character image recognition. Figure 4 shows 8×8 character images used in this simulation. Each character image is composed of 8×8 pixels. To recognize 26 character images, the memristor crossbar is composed of 27 columns. The first column connects to all inputs through \( R_B \) to generate the negative voltage as mentioned in the previous section. And 26 columns are corresponding to 26 perceptron neurons for recognizing 26 character images from ‘A’ to ‘Z’. The 64 rows correspond to 64 input voltages obtained from 64 pixels of character image to be recognized.

![Figure 4. The 8×8 pixels images of characters used to test the memristor array circuit](image)

In this simulation, we train 26 neurons to recognize 26 character images using chip-in-the-loop algorithm [20]. In testing phase, we apply the character image ‘Z’ to the crossbar and observer the output voltage of all columns when wire resistance on vertical line is 0 Ω and 2.5 Ω, respectively, wire resistance of horizontal line is ignored. The simulation result is shown in Figure 5. In Figure 5(a), when wire resistance is 0 Ω, the output voltage of the last column is close to 1V with respect to the input character image of ‘Z’, others are close to 0V, represented by square symbol in Figure 5(a). When wire resistance is 2.5 Ω, the change of output voltage can be observed, as represented by round symbol in Figure 5(a). However, the variation of voltage is not too large. If the threshold voltage is chosen to be 0.5 V, the crossbar can recognize the character ‘Z’ well. This exhibits that wire resistance on vertical line less affect to the performance of memristor crossbar circuit. The simulation result is consistent with the mathematical analysis in Eq. (9).

![Figure 5. The output voltage of 26 columns with respect to the input character image ‘Z’.](image)

Figure 5(b) shows the simulation result of memristor crossbar circuit for recognition of 26 character images, in which wire resistance on vertical line are ignored and wire resistance on
horizontal line is assumed to be 0 Ω and 2.5 Ω, respectively. In Figure 5(b), we apply the character image of ‘Z’ to the input and obtain the column voltages. As the result, when wire resistance is assumed to be 2.5Ω, the output voltages of columns tend to increase, as represented by round symbol. Since the voltage drops on wire resistors, the actual amount of voltage applied to the last column is decreased remarkably. The output voltage of the last column is increased, as shown in Figure 5(b). It also shows that the column that is close to the first column has less variation of voltage, compared to that one is far from the first column, as illustrated in Figure 5(b). The simulation result is consistent with the analysis mathematically reasoned in Eq. (13). In Figure 5(b), if the threshold is kept to be 0.5 V, the columns of 23rd, 24th, and 25th are activated when the input is ‘Z’, meaning the crossbar recognizes the input character images incorrectly. Wire resistance on horizontal line strongly affects the performance of crossbar array circuit that implements the perceptron neural network for character image recognition.

In summary, in the memristor crossbar array implementing the perceptron neural network, wire resistance on vertical line can be eliminated by compensating itself. By contrast, wire resistance on horizontal line strongly affects the performance of the memristor crossbar circuit. From the analysis and the simulation, if we want to increase the size of crossbar, we should increase the number of rows, rather than the number of columns.

4. Conclusion
In this work, the impact of wire resistance in pure memristor crossbar array is mathematically analyzed and verified by the circuit simulation. The crossbar is designed to implement the perceptron neural network for character image recognition. Wire resistance on vertical lines can be compensated itself, and less affect to the performance of the crossbar array. Wire resistance on horizontal lines degrades the recognition rate of the crossbar array significantly. It is interesting that the column that is close to the first column has less variation of output voltage, compared to the one is far from the first column. The circuit simulation results agree with the mathematical analysis. This analysis is useful. The result can be used to design the memristor circuits which can tolerate the effect of wire resistance. For example, if we want to increase the size of crossbar, we should increase the number of rows, rather than the number of columns.

5. References
[1] Mead C 1990 Proceedings of the IEEE 78 1629
[2] Morns I P and Dlay S S 1999 IEEE Trans. Neural Netw. 10 951
[3] Lont J B and Guggenbühl 1992 W IEEE Trans. Neural Netw. 3
[4] Mirsa J and Saha I 2010 Neurocomputing 74 239
[5] Mohamad K, Mahmud M F O, Adnan F H, and Abdullah W F H IEEE Symposium on Humanities, Science and Engineering Research 133
[6] Himavathi S, Anitha D, and Muthuramalingam A 2007 IEEE Trans. Neural Netw. 18 880
[7] Al-Kazzaz S A 2008 International Conference on Information and Communication Technologies 1
[8] Granado J M, Vega M A, Pérez R, Sánchez J M, and Gómez J A 2006 IEEE International Conference on Electronics, Circuit and System 934
[9] Shima T, Kimura T, Kamatani Y, Itakura T, Fujita Y, and Iida T 1992 IEEE J. Solid-State Circuits 27 1868
[10] Chua L O 1971 IEEE Trans. Circuit Theory 18 507
[11] Strukov D B, Sinder G S, Stewart D R, and Williams R S 2008 Nature 453 80
[12] Kügeler C, Meier M, Rosezin R, Gilles S, and Waser R 2009 Solid State Electron 53 1287
[13] Jo S H, Chang T, Ebong I, Bhadiya B. B, Mazumder P, and Lu W 2010 Nano Letters 10 1297
[14] Wang H, Li H, and Pino R E 2012 International Joint Conference on Neural Networks., pp. 1
[15] Kim H, Sad P, Yang C, Roska T, and Chua L O 2012 IEEE Trans. Circuit and System 59 148
[16] Adhikari S P, Yang C, Kim H, and Chua L O 2012 Trans. Neural Network and Learning System 23, 1426
[17] Chen Y C, Li H, Zhang W, Pino R E 2012 IEEE Trans. Nanotechnology 11
[18] Liang J and Wong H S P 2010 *IEEE Trans. Electron Device* **57** 2531
[19] Hu M, Li H, Wu Q, Rose G S, and Chen Y 2012 *International Joint Conference on Neural Networks* 1
[20] Truong S N and Min K S 2014 *Journal of Semiconductor Technology and Science* **14** 356
[21] Truong S N, Ham S J, and Min K S 2014 *Nanoscale Research Letters* **9** 1
[22] Truong S N and Min K S 2015 *IEEE Transactions on Nanotechnology* **14** 1104
[23] Linn E, Rosezin R, Kügeler C, and Waser R 2010 *Nature Materials* **9** 403
[24] Shin S H, Byeon S D, Song J S, Truong S N, Mo H S, Kim D J, and Min K S 2015 *Journal of Semiconductor Technology and Science* **15**
[25] Mitchell T M 1997 (McGraw-Hill Science)