3D Parallel ReRAM Computation-in-Memory for Hyperdimensional Computing

Fuyuki Kihara†, Nonmember and Chihiro Matsui†, Nonmember and Ken Takeuchi†, Member

SUMMARY In this work, we propose a 1T1R ReRAM CiM architecture for Hyperdimensional Computing (HDC). The number of Source Lines and Bit Lines is reduced by introducing memory cells that are connected in series, which is especially advantageous when using a 3D implementation. The results of CiM operations contain errors, but HDC is robust against them, so that even if the XNOR operation has an error of 25%, the inference accuracy remains above 90%.

key words: Computation-in-Memory, ReRAM, Chain-cell Memory, Hyperdimensional Computing

1. Introduction

Hyperdimensional Computing (HDC) is an emerging neuromorphic computing paradigm [1-5]. HDC has similar characteristics to neurons such as hyper-dimensionality, lower bit width, randomness, and robustness. The most important feature of HDC is that all data are represented in Hypervector (HV). Typically, a HV consists of 1 bit × 10,000 dimensions. Almost all operations are bitwise operations, therefore HDC is suitable for parallel processing and Computation-in-Memory (CiM) [5-11]. CiM is one of the many parallel processing methods. CiM performs simple calculations (e.g., multiply-accumulate (MAC) operation) simultaneously with the memory readout. For large-scale algorithms such as Machine Learning, the cost of data transfer is a major barrier, known as the von Neumann bottleneck, and CiM is seen as a promising way to solve this problem. In addition, using ReRAM, one of the emerging Non-Volatile Memories (NVMs), grants some advantages such as energy efficiency [9-13]. In this work, we propose a ReRAM CiM architecture suitable for N-gram Encoder in HDC.

2. HDC and Language Classification

Language Classification task classifies texts consisting of 26 letters of the alphabet and space into 21 European languages [4]. As shown in Fig. 1(a), in the training phase, Prototype HVs (PHV) are made from texts written in the respective language by the Encode module and stored in Associative Memory (AM). In the inference phase, a Query HV (QHV) is compared with PHVs. If the distance is below the critical level, the language corresponding to the PHV is declared the winning language with a winner-elector (AE).

![Fig. 1 Summary of Language Classification process.](image_url)

(a) Language Classification with Hyperdimensional Computing

- **Training**
  - Text (Language_1)
  - Text (Language_2)
  - Text (Language_3)
  - PHV (Language_1)
  - PHV (Language_2)
  - PHV (Language_3)

- **Inference**
  - Similarity
  - Inference result

(b) Encode module

- **Input text**
  - Item Memory
  - N-gram encoder
  - N-gram
  - Majority rule

---

† The authors are with Department of Electrical Engineering and Information Systems, Graduate School of Engineering, The University of Tokyo, Bunkyo-ku, 113-8656 Japan.
a) E-mail: kihara@co-design.t.u-tokyo.ac.jp

Copyright © 20XX The Institute of Electronics, Information and Communication Engineers
is made from the text written in a European language by the Encode module and taken similarity with each PHV. Fig. 1(b) shows the structure of an Encoding module. Item Memory (IM) stores Letter HVs (LHV) that are independently assigned to each letter. All N-grams in the input text are converted into respective N-gram HV. LHV of former letters are permutated by a permutation matrix (ρ) to keep series information. N-gram HV is obtained by taking the bitwise XNOR of all rows and permutated LHV, and the Text HV is obtained by taking the bitwise major rule operation of all the all N-gram HV.

3. Proposed ReRAM CiM

Fig. 2(a) shows the proposed ReRAM CiM array suitable for HDC. It can take the sum and XNOR by accumulating resistances of ReRAM devices when read out. The SUM operation is the number of ReRAM devices in the High Resistance State (HRS), and XNOR is calculated as HRS=1, Low Resistance State (LRS)=0. As shown in Fig. 2(b), this block is arranged in a row and HVs are mapped to implement an N-gram encoder. Since the results of the SUM and XNOR operations have the relationship shown in Fig. 2(c), the XNOR result can be calculated with this CiM array.

Fig. 3(a) Switching mechanism of ReRAM [13]. (b) Conductance and Resistance distribution of ReRAM device (without FET).

Fig. 4 Operations of ReRAM CiM array. (a) Reset. (b) Set. (c) Read (One cell). (d) Read (CiM).

Fig. 5 Equivalent circuit of 3D integration.
This architecture reduces the number of Source Lines (SL) and Bit Lines (BL) [14-15]. Then, each of these Lines can be manufactured thicker to suppress the resistance. SL and BL parasitic resistances cause IR drop and make device characteristics worse. In addition, cell size is expected to be smaller. According to [14], the chain cell structure reduces the area per cell from $8\times 2$ to $4\times 2$. We assume that a similar effect might be expected with ReRAM. Furthermore, in 3D integration, the construction of SL and BL on BEOL metallic layers as shown in Fig. 5 provides significant benefit [9,15].

Implementation of the $N$-gram encoder by the ReRAM CiM is shown in Fig. 6. ReRAM CiM stores all raw and permuted LHV s as IM and outputs $N$-gram HV at the same time as read out. $N$ WLs corresponding to each letter of an $N$-gram are activated, and bitwise XNOR of the LHV s is calculated, then $N$-gram HVs are obtained [4].

\begin{figure}[h]
\centering
\includegraphics[width=0.8\textwidth]{fig6.png}
\caption{Proposed $N$-gram encoder. 4-gram encoder mapped to ReRAM CiM array with 108 WLs. Operation example when Input text is “I am”.}
\end{figure}

4. Evaluation

(i) Language Classification with Error

Fig. 7 shows the evaluation result of Language Classification described in Section 2 with bit inversion error in XNOR operation of $N$-gram encoder in inference phase. For each language, there is one training text of about 100k characters and 1000 testing texts. 4-gram encoder with bit inversion error is used to generate PHV from the training text and QHV from the testing text. The inference accuracy remains higher than 0.9 for up to a bit error rate as high as 0.25.

(ii) Proposed ReRAM CiM

\begin{figure}[h]
\centering
\includegraphics[width=0.8\textwidth]{fig7.png}
\caption{Inference accuracy of error injection into $N$-gram encoder.}
\end{figure}

\begin{figure}[h]
\centering
\includegraphics[width=0.8\textwidth]{fig8.png}
\caption{Simplified circuit for simulation.}
\end{figure}

Fig. 8 shows the simplified circuit that consists of fixed ohmic resisters used by the simulations. Resistors of resistance $R_{ON}$ or $R_{OFF}$ are replacement for the ON and OFF states of the FETs. Resistance of ReRAM devices, $R_{LRS}$ and $R_{HRS}$, are chosen randomly from measured resistance of ReRAM (Fig. 3(b)).

Fig. 9 shows simulated resistances of 8-bit ReRAM CiM. Overlaps are minimized when $R_{OFF}$ equals $10 \times R_{LRS}$. The results of a 108-bit ReRAM CiM are shown in Fig. 10. To evaluate the behavior as an $N$-gram encoder ($N = 4$), in this simulation, the number of selected WLs is restricted to 4. A one shift in the result of the SUM operation means that the effect might be expected with ReRAM.

In this case, overlaps are minimized when $R_{OFF}$ equals $1 \times R_{LRS}$. As shown in Fig. 10(b)(c), by setting the thresholds of ADC (red line) appropriately, the read error and the error rate of the XNOR operation is inverted. In this case, overlaps are maximized when $R_{OFF}$ equals $1 \times R_{LRS}$ or $10 \times R_{LRS}$. These overlaps get worse because of change in the distribution of $R_{HRS}$, with increase in $R_{OFF}$. When $R_{OFF}$ is not quite larger than $R_{HRS}$, the distribution of $R_{OFF}/R_{HRS}$ is suppressed. The change in $R_{OFF}/R_{HRS}$ when $R_{HRS}$ changes by $\Delta R_{HRS}$ can be calculated as follows:

$$R_{OFF}/(R_{HRS} + \Delta R_{HRS}) - R_{OFF}/R_{HRS} = \frac{R_{OFF}^2 \Delta R_{HRS}}{(R_{OFF} + R_{HRS} + \Delta R_{HRS})(R_{OFF} + R_{HRS})}$$

By adjusting $R_{OFF}$, the error rate can be reduced.
Fig. 9 Simulated combined resistance of CiM operation of 8-bit ReRAM array (for general purpose). $R_{OFF} = (a)0.1 \times R_{LRS}$ (100) \times R_{LRS}$.

Fig. 10 Simulated combined resistance of CiM operation of 108-bit (27 letters × 4-bit) ReRAM array (for N-bit encoder). 4 WLs are selected. $R_{OFF} = (a)0.1 \times R_{LRS}$ (b)1 \times R_{LRS}$ (d)100 \times R_{LRS}$. In (b) and (c), examples of thresholds (red line) where the read error is below 25% are included.

5. Conclusion

This work proposes the ReRAM CiM architecture suitable for HDC. The size of memory cell block and IR drop in BL and SL can be reduced by introducing cell with FET and ReRAM connected in parallel. This CiM method includes errors in the calculation results, but Language Classification using HDC can maintain 90% inference accuracy even when the XNOR operation contains 25% errors. This result is not greater than the loss of inference accuracy due to other errors [4,5]. Therefore, the reduction in inference accuracy due to the implementation of the proposed CiM into HDC can be tolerated. Reducing the $R_{OFF}$ variation is important to reduce errors in the computation and a future challenge [16].

References


