Design Methodology towards High-Precision SRAM based Computation-in-Memory for AI Edge Devices

Tianzhu Xiong, Yongliang Zhou, Yuyao Kong, Bo Wang, An Guo, Yufei Wang, Chen Xue, Haiming Hsu, Xin Si*, Jun Yang

Southeast University

the National ASIC System Engineering Center

Nanjing, China (*email: xinsi@seu.edu.cn)

2021 18th International SoC Design Conference (ISOCC) | 978-1-6654-0174-6/21/$31.00 ©2021 IEEE | DOI: 10.1109/ISOCC53507.2021.9613913

Abstract—**Artificial Intelligence (AI) processors commonly use deep-learning-based neural networks (DNN), which mainly include convolution layers and fully connected (FC) layers. Both the convolution and FC layers require highly parallel multiply-and-accumulate (MAC) operations and generate a great deal of intermediate data. Under the von Neumann computing architecture, data transfer between processor and memory imposes high energy consumption and long latency, which significantly deteriorates the system's performance and efficiency. Computation-in-memory (CIM) is a promising candidate to improve the energy efficiency of multiply-and accumulate (MAC) operations of artificial intelligence (AI) chips. However, there are always constraints between high precision and high energy efficiency in CIM. This paper reviews the precision requirements of popular DNN models, and outlines the tradeoff between the precision and the energy efficiency in SRAM-CIM designs.**

Keywords; Artificial intelligence (AI), computation-in-memory (CIM), neural network, static random access memory.

I. INTRODUCTION

Artificial Intelligence (AI) algorithms enable computers to handle various cognitive tasks. Machine learning (ML) algorithm based deep neural network (DNN) processors have been successfully applied in image classification, speech recognition, natural language processing, robot and other AI fields. However, these applications require inference under strict restrictions on latency, energy consumption, and area overhead [1]. Therefore, it is an important issue to implement ML algorithm on resource constrained edge devices. This is because the bandwidth of data transmission between digital processor and memory is limited by the memory wall, which is caused by the growing speed gap between processor and memory.

Computation-in-Memory (CIM) is a paradigm that can achieve the parallel multiply-accumulate operations within memory circuits. It can break through the limitation of memory wall. In addition, SRAM-CIM is used to encode multi-bit features/weights in a customized CIM macro, which is then designed to perform some or all of the multiply-accumulate operations needed for AI or ML inference during the computation mode. These approaches typically involve multiple bit- or word-lines for highly parallel multiply-accumulate operations. For this reason, SRAM-CIM can process and store data at the same locations, which makes it become one of the most desirable choices for energy-efficient DNN accelerators and edge AI devices.

In the rest of this paper, we summarize up-to-date CIM-implemented precision scalability for energy-efficient AI Edge devices, and investigate the key challenges and prospects in high-precision SRAM-CIM designs.

II. PRECISION REQUIREMENTS IN CIM BASED MULTIPLY-ACCUMULATE UNITS FOR NEURAL-NETWORK PROCESSING

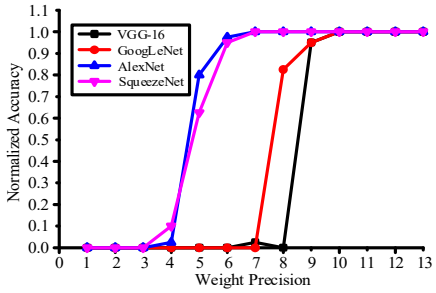

Fig. 1 Normalized inference accuracy versus various weight precisions with the employment of some typical neural networks [2]

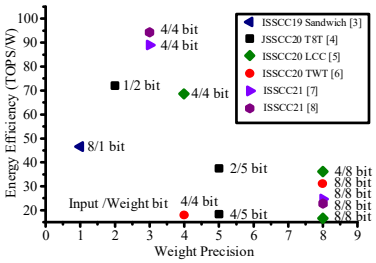

Fig. 2 Energy efficiency of recent silicon-verified SRAM-CIM macros versus the quantized weight bit-precision.

Fig. 1 shows the relationship between the accuracy and the weight precision required for inference in some well-known DNN models, such as AlexNet, GoogLeNet, SqueezeNet, and VGGNet. Different DNN models show diversity on the optimal weight precision for high inference accuracy. For example, at least 10-bits is required in GooglLeNet in order to maintain a considerable recognition accuracy. And there is a trend that most DNN models require higher bit-precisions to achieve better accuracies. However, previous CIM-based accelerators face significant challenges and tradeoffs to meet the high-accuracy requirement of complex neural networks, such as limited signal margin, larger area overhead, and poor configurability. Therefore, existing CIM-based DNN

accelerators should be further explored to support applications that require high bit-precision.

Fig.2 shows the tradeoffs associated between the energy efficiency and the weight precision in state-of-the-art SRAM-CIM designs. The increase of weight precision decreases energy efficiency because multiple memory cells are required to represent a multibit weight. Based on the same SRAM-CIM structure, the increase of input precision leads to the extension of execution time, and further decreases the throughput and energy efficiency. Recently, with the consideration of the minimum bit-precision required for popular DNN models, most SRAM-CIM designs are explored to support high-precision MAC operations, such as 8bit MAC operations.

III. HIGH-PRECISION CIM DESIGN BASED ON SRAM

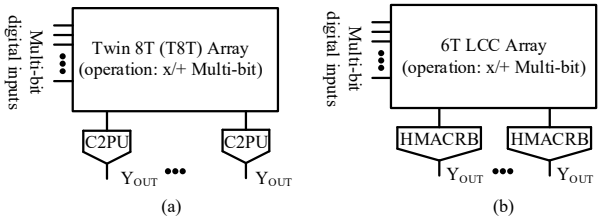

Fig. 3 High-level of previous SRAM-CIM structure (a) Twin-8T (T8T) SRAM-CIM for CNN [4]. (b) Local-Computing-Cell (LCC) SRAM-CIM [5].

The designs of high-precision SRAM-CIM face major challenges and tradeoffs in (a) suffered write disturb issue when SRAM-CIM works at compute mode; (b) limited signal and sensing margins with the increase of number of MAC operations; (c) significant area overhead caused for signed weight storage; and (d) large area cost and power consumption of high-precision readouts. To explore the design methodology towards high-precision SRAM-CIM, several precision-scalable CIM designs for MAC operations have been proposed and silicon-verified.

In conventional 6T-SRAM based SRAM-CIM designs, when multiple word-lines are activated simultaneously to implement MAC operation, the bit-line voltage can be pulled down to lower than the write margin of single bit-cell. Therefore, it may disturb the stored weight data in each bit-cell, which can also be called as write disturb issue. To overcome this issue, the use of more complicated SRAM bit-cell can help to support compute mode without the influence of conventional memory mode, such as twin-8T, as shown in Fig. 3 (a).

With the increase of data bit-precision, the signal margin between two adjacent MAC values is limited. This can lead to large readout accuracy loss when the signal margin is smaller than the input offset of multibit readout circuits. In order to enlarge the signal margin, multibit data mapping schemes can be employed to ensure sufficient signal margin at analog domain, such as weight-bitwise MAC mapping method, as shown in Fig. 3 (b), in which a high-precision MAC operation is split into multiple analog domain low-precision weight-bitwise MAC operations and a digital weight place-value combination operation.

In typical DNN models, the MAC values from the previous convolution layer undergo a rectified linear unit (ReLU) activation before being fed into the following convolution layer. Thus, the inputs are all positive values, whereas the weights are either positive or negative weights. To implement multibit weights based CIM designs, two methods can be adopted to handle positive and negative weights: (a) separated positive-negative weight placement, and (b) two's complement mapping scheme.

To reduce large area cost and power consumption caused by the high-precision readouts, software-hardware co-design methodology can be employed. For example, algorithm-adaptive low-MAC aware readout circuit is developed with the observation of the MAC distribution [9].

IV. CONCLUSION

This paper explores the challenges of state-of-the-art SRAM-CIM designs in the high-accuracy required DNN models. And we also outlined the trends in the developments of various SRAM-CIM designs. In pursuit of high-energy efficiency, earlier SRAM-CIM designs mainly focused on low precision MAC operations, but with significant accuracy loss when applied to complex datasets. To further support high-precision MAC operations, we outlined and summarized some design methods towards high-precision SRAM-CIM designs.

ACKNOWLEDGEMENT

This work was supported by the Fundamental Research Funds for the Central Universities of China under Grant 2242021k30031.

REFERENCES

[1] M.-F. Chang, Nonvolatile Circuits for Memory, Logic, and Artificial Intelligence, IEEE International Solid State Circuits Conference (ISSCC) tutorial (2018).

[2] J. -H. Kim et al., Z-PIM: A Sparsity-Aware Processing-in-Memory Architecture With Fully Variable Weight Bit-Precision for Energy-Efficient Deep Neural Networks, in IEEE Journal of Solid-State Circuits, vol. 56, no. 4, pp. 1093-1104, April 2021.

[3] J. Yang et al., Sandwich-RAM: An Energy-Efficient In-Memory BWN Architecture with Pulse-Width Modulation, IEEE International Solid-State Circuits Conference, pp. 394-396, (2019).

[4] X. Si et al., A Twin-8T SRAM Computation-in-Memory Unit-Macro for Multibit CNN-Based AI Edge Processors, in IEEE Journal of Solid-State Circuits, pp. 189-202, Jan. 2020.

[5] X. Si et al., A 28nm 64Kb 6T SRAM Computing-in-Memory Macro with 8b MAC Operation for AI Edge Chips, IEEE International Solid-State Circuits Conference, pp. 246-248 (2020).

[6] J. Su et al., A 28nm 64Kb Inference-Training Two-Way Transpose Multibit 6T SRAM Compute-in-Memory Macro for AI Edge Chips, IEEE International Solid- State Circuits Conference, pp. 240-242 (2020)..

[7] Y. -D. Chih et al., An 89TOPS/W and 16.3TOPS/mm2 All-Digital SRAM-Based Full-Precision Compute-In Memory Macro in 22nm for Machine-Learning Edge Applications, IEEE International Solid-State Circuits Conference, pp. 252-254, (2021).

[8] J. -W. Su et al., A 28nm 384kb 6T-SRAM Computation-in-Memory Macro with 8b Precision for AI Edge Chips, IEEE International Solid-State Circuits Conference, pp. 250-252, (2021).

[9] X. Si et al., A Local Computing Cell and 6T SRAM based Computing-in-Memory Macro with 8b MAC Operation for Edge AI Chips, IEEE Journal of Solid-State Circuits, (2021).