RL-Guided Thermal-Aware Quantization for Efficient and Robust ReRAM CIM Systems

Lihua An, Jiayi Li and Pingqiang Zhou
School of Information Science and Technology, ShanghaiTech University, Shanghai, China
Email: {anlh2022, lijy22023, zhoupq}@shanghaitech.edu.cn

2026 31st Asia and South Pacific Design Automation Conference (ASP-DAC) | 979-8-3315-9123-6/26/$31.00 ©2026 IEEE | DOI: 10.1109/ASP-DAC66049.2026.11420481

***Abstract*—Resistive RAM (ReRAM)-based Computing-in-Memory (CIM) systems present significant advantages in energy efficiency and computational throughput for neural network acceleration. However, their performance is highly constrained by thermal-induced conductance drift, especially under aggressive quantization strategies. This work presents a reinforcement learning (RL)-guided thermal-aware layer-wise quantization framework optimized for ReRAM-based CIM systems. The proposed method encourages sparse and low-magnitude weight representations, while adaptively exploring layer-wise bit-width configurations guided by direct hardware evaluation feedback and thermal-aware reward. Experiments on CIFAR-10 and ImageNet benchmarks with ResNet and VGG show that the proposed method achieves up to 10.2% peak temperature reduction, and an average top-1 accuracy improvements of 46.37% over fixed 8-bit baselines. Compared to prior layer-wise quantization methods without thermal considerations, our approach improves accuracy by 6.36%–15.06% on average.**

***Index Terms*—Neural Network, Computing in Memory, Thermal Effect, Reinforcement Learning, Quantization, ReRAM**

I. Introduction

Resistive RAM (ReRAM)-based computing-in-memory (CIM) architectures have emerged as a promising paradigm for energy-efficient neural network acceleration, particularly in power- and area-constrained edge devices [1]. By tightly integrating computation and storage within dense crossbar arrays, these architectures effectively eliminate the von Neumann bottleneck and enable highly parallel matrix-vector multiplications (MVMs) through current-sum manner. However, frequent analog computations are sensitive to non-idealities like Stuck-At-Fault (SAF) [2], conductance variations [3], especially thermal effects [4], posing critical challenges to reliability and accuracy.

ReRAM devices exhibit significant temperature sensitivity, particularly under low-bit resolutions (1-4 bits) [5]. Fig. 3(a) [4] shows the asymmetry of conductance variations: the conductance of low resistance state (LRS) drops rapidly as temperature increases especially after 330K, while high resistance state (HRS) increases slowly. Such thermal-induced conductance drift accumulates along array columns and can significantly distort the current output, causing more than 90% inference accuracy degradation at high temperature [6]. Existing thermal mitigation approaches primarily focus on hardware and software solutions. Hardware solutions [5], [7] typically aim to compensate for column-wise current distortion. At the algorithm level, most approaches focus on adjusting weight distributions to reduce localized hotspots or protect

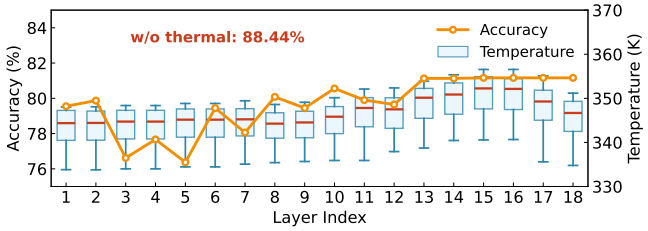
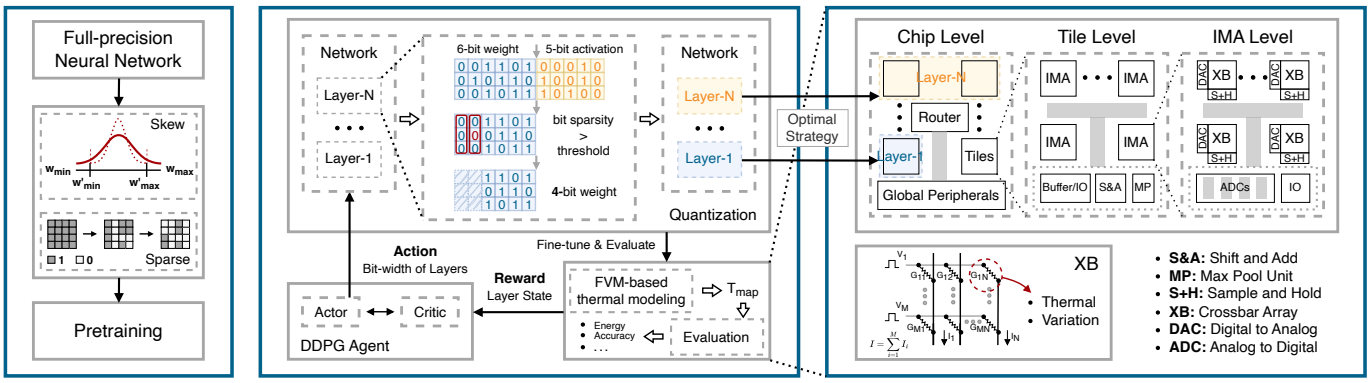


Fig. 1. The box plot shows layer-wise temperature distribution of an 8-bit ResNet18 model, while the orange line indicates the accuracy when thermal effects are applied to a single layer at a time, with all other layers remaining ideal. This illustrates the impact of each layer's thermal behavior on overall inference accuracy. The accuracy without any thermal effects is 88.44%.

thermally vulnerable weights, like re-mapping schemes [8], [9] and thermal-aware weight decomposition [5]. While these techniques are effective, they fundamentally operate under the constraint of a fixed number of weights, merely using additional specific hardware design or redistributing weights to avoid thermal hotspots. In contrast, structural compression methods such as pruning and quantization offer a more direct and scalable solution by reducing the total number of active weights [10], thereby lowering power density. However, pruning [11], [12] can significantly change the weight distribution, thus adversely affecting inference accuracy [13].

Quantization, on the other hand, offers a more flexible solution to address thermal and energy efficiency challenges. By converting the high-precision weights into low-precision formats, quantization reduces DAC/ADC power consumption and ReRAM write energy, effectively reducing localized power density and temperature without removing weights. However, uniform assignment across all layers is suboptimal, since different layers exhibit varying sensitivity to quantization and thermal drift, as shown in Fig. 1. Prior works [10], [15]–[19] have explored quantization schemes for area and energy efficiency. While fine-grained methods (e.g., block-wise [10] or group-wise [16]) offer higher flexibility, they typically introduce significant control or computation overhead. Layer-wise quantization provides a better trade-off between hardware feasibility and adaptability. Representative examples include HAQ [19], which uses reinforcement learning (RL) to optimize bit-widths, and BSQ [18] uses sparsity-driven training. However, these methods primarily focus on area and energy without considering thermal effects or device-level variations. To this end, these insights motivate a thermal-aware layer-wise



(A) Pretraining **(B) Thermal-aware Quantization Search** **(C) ReRAM-based CIM System**

Fig. 2. Overview of the proposed framework. (A) Pretraining stage guides sparse and low-magnitude weight distributions. (B) The DDPG agent explores the layer-wise bit-widths under thermal and accuracy constraints. (C) ISAAC-based [14] ReRAM CIM systems for evaluation.

quantization framework.

In this paper, *we propose the first RL-guided, thermal-aware layer-wise quantization framework with a sparse and low-magnitude weight distribution*. Unlike prior methods, our approach leverages reinforcement learning to automatically explore optimal bit-width configurations guided by evaluation feedback. By directly integrating thermal feedback into the search process, the proposed method achieves both thermal robustness and high efficiency on ReRAM-based CIM systems. The key contributions of this work are summarized as follows:

- We introduce a thermal-resilience regularization technique that promotes sparse and low-magnitude weight regularizations, improving the system's thermal stability.
- We propose a Deep Deterministic Policy Gradient (DDPG)-based RL algorithm to autonomously explore layer-wise bit-width configurations under both thermal and accuracy constraints. To enhance thermal robustness, we integrate thermal evaluation feedback into the reward function and design a layer-specific bit-width search space based on thermal sensitivity. Additionally, a post-search quantization refinement step is introduced at the end of each search cycle to further reduce bit-widths.
- Experimental results show that our method improves top-1 accuracy by 46.37% on average, reduces peak temperature by up to 10.2%, and achieves up to $\sim 3.60\times$ improvement in energy efficiency compared to fixed 8-bit quantization. Compared to prior layer-wise quantization methods without thermal considerations, our approach improves accuracy by 6.36%–15.06% on average.

II. Background

A. ReRAM-based CIM Architecture

Fig. 2(C) illustrates the ISAAC architecture [14], a representative ReRAM-based CIM systems design. This architecture consists of multiple tiles connected via routers. Each tile contains in-situ multiply-accumulate (IMA) units based on ReRAM crossbar arrays, serving as basic computation units for MVM operations. Neural network layers are mapped onto these tiles. Within each crossbar, ReRAM devices connect to bitlines and wordlines, where input features are converted to voltages via DACs, and weights are programmed to the conductance G of ReRAM. According to Kirchhoff's law, MVM operations are performed in parallel: $I = \sum_{i=1}^{M} V_i \times G_{i,j}$, where M and N denote the number of rows and columns, G is the conductance matrix, V is the input voltage vector, and I is the output current, which is subsequently converted to the digital domain by ADCs.

B. Quantization and Bit-Level Mapping

Quantization is a widely adopted technique in hardware accelerators to reduce energy consumption. It maps high-precision values into lower bit-width representations while preserving accuracy. Bit-width determines the hardware-level storage capacity, while precision effective numerical fidelity achieved through quantization schemes and weight distribution optimization. In this work, each weight w is quantized into n-bit as:

$$w_q = round(w/s) \times s \tag{1}$$

where $s = 1/(2^n - 1)$ is the scaling factor.

In ReRAM-based CIM systems, quantized weights are further decomposed into binary representations and mapped onto multiple ReRAM devices within the crossbar arrays. For 1-bit ReRAM devices, each bit corresponds to either a high conductance state (LRS, denoted as "1") or a low conductance state (HRS, denoted as "0"). For example, representing an 8-bit weight requires 8 separate devices. As a result, the bit-level structure and overall bit-width of a quantized weight directly determines the number of activated "1" states in the array. Since "1" states exhibit greater thermal sensitivity (Fig. 3(a)), minimizing their occurrence can reduce thermal stress and improve reliability, which promoting bit-level sparsity. Moreover, employing low bit-width quantization reduces the number of devices and columns required per weight, leading to lower DAC/ADC power consumption and ReRAM write energy, thereby contributing to reduced power density and temperature.

III. Proposed Methodology

A. Overview

Our proposed framework consists of three stages, as illustrated in Fig. 2. **(A) Pretraining Stage:** We apply two regularization techniques during trainig to promote sparse and low-magnitude weight distributions. This step reduces the number of thermally sensitive "1" conductance states and improves thermal robustness before quantization process. **(B) Thermal-aware Quantization Search:** A DDPG agent is used to explore layer-wise quantization configurations under both thermal and accuracy constraints. The reward function incorporates system-level evaluation feedback such as temperature, energy and accuracy, and we design a layer-specific bit-width search space based on thermal sensitivity, which guides the search process toward thermally efficient bit-width assignments. Additionally, a post-search quantization refinement step at the end of each search cycle is applied to further reduce bit-widths. **(C) System-level Evaluation:** The quantized model is mapped onto a ReRAM-based CIM simulator to evaluate temperature, energy and inference accuracy. These results are fed back to the RL agent for further optimization.

B. Thermal-Aware Pretraining

Thermal reliability has become a key challenge in ReRAM-based CIM systems due to the temperature sensitivity of analog conductance states. Fig. 3(a) reveals that the "1" conductance state exhibits greater sensitivity to thermal variation compared to the "0" state. Since the thermal variation has selective impacts on the different conductance states, the weight distribution will largely affect the thermal robustness of the model, which motivates the following observations:

Observation 1: Sparse weights lead to fewer programmed "1" states, reducing the number of thermally sensitive devices and improving overall robustness (Fig. 3(b)).

Observation 2: Errors in higher bit positions (e.g., MSBs) cause more severe quantization distortion under thermal drift than those in lower bit positions. Assigning "1" states to less significant bits improves tolerance to thermal variation, as shown in Fig. 3(c).

Therefore, thermal robustness depends not only on power reduction but also on weight distribution and bit-width assignment. To this end, we introduce a thermal-aware pretraining stage that shapes the weights toward sparsity and low magnitude, minimizing the number and impact of thermally sensitive "1" states before quantization. This is achieved by incorporating two regularization terms during training, which guide the model toward a more thermally robust representation.

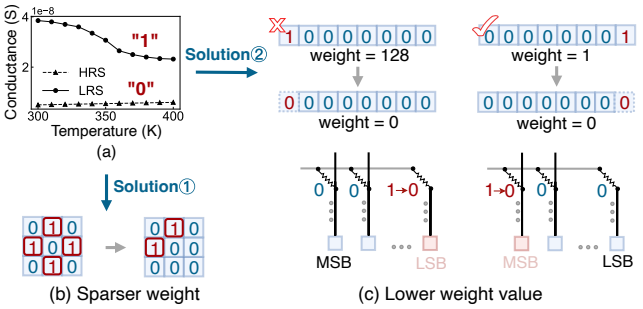


Fig. 3. (a) Temperature-dependent conductance variation in ReRAM devices [4], where HRS and LRS denote high and low resistance state. (b) Sparsity improves robustness. (c) Low-magnitude weights are less vulnerable to thermal drift.

Bit-Level Sparsity: In this work, we define sparsity as the proportion of zero-valued bits in the binary representation of quantized weights across all layers. Firstly, we convert floating-point weights to binary representation. For a n-bit quantized weight w_q, the corresponding floating-point weight w is defined as:

$$w \equiv \text{sign}(w) \odot s w_q \equiv \text{sign}(w) \odot \frac{s}{2^n - 1} \sum_{b=0}^{n-1} w^{(b)} 2^b \quad (2)$$

where $w^{(b)}$ denotes the b^{th} bit in the binary representation, $\odot$ denotes the element-wise Hadamard product, and s is the scaling factor.

Inspired by [18], we encourage sparsity by applying a layer-wise group Lasso penalty to minimize bit-level activity:

$$\mathcal{L}_{GL} = \sum_{l=1}^{L} \left(\frac{\#Para(w^l) \times \#Bit(w^l)}{\#Para(w^{(1:L)})} \sum_{b=0}^{n-1} \left\| w^{(l,b)} \right\|_2 \right) \quad (3)$$

where $w^{(l,b)}$ denotes the binary representations of the b^{th} bit weights of layer l, L is the number of layers, $\#Para(w^l)$ and $\#Bit(w^l)$ respectively denote the parameter number and bit-width of layer l.

Low-Magnitude Weight: To reduce high-conductance states, we apply skewed L2 regularization [20] that penalizes values outside a reference range δ:

$$\mathcal{L}_{skewed} = \sum_{|w^l|<|\delta|} \left\| w^l - \delta \right\|_2 \quad (4)$$

Weights outside this reference range δ are strongly penalized, encouraging the model to constrain values within the desired range. The weight distribution after training with above regularization is shown in the dash line of Fig. 2(A), which has a new weight range $[w'_{min}, w'_{max}]$.

The overall training loss function for each layer is:

$$\mathcal{L} = \mathcal{L}_{CE} + \alpha \mathcal{L}_{GL} + \beta \mathcal{L}_{skewed} \quad (5)$$

where $\mathcal{L}_{CE}$ is the original cross entropy loss function, α and β are the regularization strength for the group Lasso and skewed regularization, respectively. The training process starts with converting a floating-point model to a relatively high initial bit-width (e.g., 8-bit).

C. Thermal-Aware Quantization Search

To ensure thermal robustness in ReRAM-based CIM systems, it is critical to assign bit-widths in a layer-wise manner that reflects each layer's thermal sensitivity. Our goal is to dynamically balance inference accuracy, thermal stability, and energy efficiency. To this end, we propose a thermal-aware quantization framework that incorporates feedback from system-level thermal evaluation into the search process.

Unlike prior quantization methods that only optimize for energy or area, our approach integrates thermal variation into the decision-making process. As illustrated in Fig. 2(B), the procedure includes: (1) the DDPG agent makes actions to determine bit-width configurations based on current state and feedback from prior evaluations; (2) quantized model is fine-tuned and evaluated using the ReRAM-based CIM simulators to obtain temperature, accuracy, and energy metrics; and (3) the reward is updated to guide the next step.

1) State and Action Space: At each step, the agent observes a state space containing the current layer index l, layer type, dimensions (e.g., input/output channels or hidden units), and the previous action a_{l-1}. All features are normalized to [0, 1] for consistency.

For action space, we use a continuous space to represent the quantization bit-widths of each layer [19]. At the l^{th} time step, the action $a_l \in [0, 1]$ is mapped to the bit-width b_l:

$$b_l = round(b_{min} + 0.5 + a_l \times (b_{max} - b_{min} + 1)) \quad (6)$$

where b_{min} and b_{max} are the minimum and maximum bit-widths, typically ranging from 2-bit to 8-bit.

In this work, we implement a layer-specific bit-width assignment search space based on the sensitivity of each layer to accelerate the RL search process. As shown in Fig. 1, thermal sensitivity is layer-dependent, although deeper layers exhibit higher temperatures, they demonstrate greater robustness. And following insights from [18], deeper layers tend to have lower quantization bit-widths. Therefore, we define bit-width ranges as follows: [2-bit, 8-bit] for early convolutional layers to preserve feature extraction capability, [2-bit, 4-bit] for later convolutional layers due to higher redundancy, and [6-bit, 8-bit] for FC layers to maintain accuracy. This adaptive strategy improves search efficiency of the search process while preserving overall model performance.

2) Reward Function: The objective of quantization is to find the optimal bit-widths for each layer l to minimize the accuracy loss while keeping the sparsity obtained from the pretraining stage. The reward function is defined as:

$$\mathcal{R} = \lambda \times (acc_{quant} - acc_{origin} + \frac{1}{cost}) \quad (7)$$

where acc_{origin} is the top-1 accuracy of the full precision model, acc_{quant} is the accuracy of the quantized model which is obtained from the hardware evaluation, λ is set to 0.1 in our experiments, and the cost function is defined as:

$$cost = \sum_{l=1}^{L} \left(\sum_{b=0}^{n-1} w^{(l,b)} + \Delta T_l \right) \quad (8)$$

where ΔT_l is the changes of temperature of layer l. Our reward function incorporates thermal feedback through finite volumn method (FVM)-based thermal modeling [21], [22], making it the first solution to directly optimize thermal-accuracy trade-offs during RL search.

3) Post-Search Quantization Refinement: Once the agent selects the action a_l, each layer's weights and activations are quantized using Equation (1). For simplicity, activation quantization was not applied in the ImageNet experiments.

To further reduce bit-widths, we apply a additional post-search quantization refinement at the end of each exploration epoch, leveraging the sparsity of the model. As shown in Fig. 2(B), this process analyzes the binary representation of weights. Taking a 6-bit weight as an example, each bit is examined from the most significant bit (MSB) to the least significant bit (LSB). If a certain bit for all weight in a layer is zero, or the number of non-zero elements is less than a threshold (5% in our experiments, which provides a good trade-off between preserving accuracy and reducing redundant bits), that bit is considered redundant and discarded. This process continues until the first non-zero bit is encountered, representing the most significant bit that cannot be safely discarded. This bit-level refinement further reduces quantization bit-widths, thereby lowering the number of ReRAM devices required for weight storage.

D. System-Level Evaluation

The optimized quantization policies are deployed within a ReRAM-based CIM simulator that integrates a FVM-based thermal model to evaluate thermal profiles, energy efficiency and inference accuracy. The FVM thermal module [21], [22] estimates the temperature distribution T_{map} across the crossbars based on power density. These thermal profiles, along with system-level metrics such as energy consumption and inference accuracy, are used to compute the reward, which is then propagated back to update the RL agent. This feedback loop enables the agent to optimize for both accuracy and power efficiency, while minimizing the local thermal hotspots.

IV. Results

A. Experimental Setup

We used ISAAC [14] as the ReRAM-based CIM architecture and develop a simulator based on MNSIM2.0 [23], integrated with finite volumn method (FVM)-based thermal modeling [21], [22] to evaluate energy, thermal, and accuracy performance. The accelerator consists of 256 tiles, each with 8×8 processing elements (PEs), where each PE contains a 128×128 ReRAM crossbar (1-bit). Device parameters follow a 40nm technology node, with HRS/LRS conductance set to 5/38.5 nS, and 8-bit/1-bit ADC/DAC resolutions.

We evaluate on VGG16, ResNet18, and ResNet34 with the CIFAR10 [24] and ImageNet [25] datasets. Comparisons are made with state-of-the-art quantization schemes, including HAQ [19], BSQ [18], and a 8-bit uniform quantization baseline. The DDPG agent comprises actor and critic networks sharing the same architecture: two 400-unit fully connected layers, followed by a merged path with additional layers of size 300 and 1. The regularization coefficients are set to $\alpha = 1 \times 10^{-5}$ and $\beta = 0.0015$, and use the standard deviation as the reference weight σ with training for 500 epochs.

TABLE I

RESULTS OF PRETRAINING STAGE

Model	Method	Acc. (%)	(Δ%)
ResNet18 on CIFAR10	Baseline	31.09	0.00
	Sparse	36.80	+5.71
	Skewed	46.64	+15.55
	Sparse + Skewed	51.80	+20.71

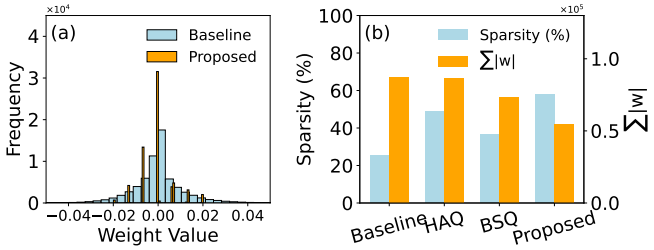


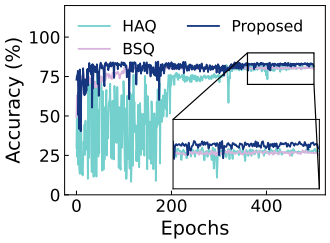
Fig. 4. (a) Weight distribution comparison between the pretrained and baseline models on ResNet18 with CIFAR10 dataset. (b) Comparison of the weight sparsity and sum of absolute value on ResNet18 with CIFAR10 dataset.

B. Sparsity and Skewed Weight Distribution

To evaluate the effectiveness of our pretraining strategy, we conduct an ablation study on ResNet18 using CIFAR10 dataset. As summarized in Table I, both group Lasso and skewed regularizations improve model robustness, while their combination yields the highest accuracy improvement of +20.71%. Fig. 4(a) shows that the proposed regularization leads to a sparser weight distribution compared to the baseline, with weights concentrated in a narrower range. We defined sparsity as the proportion of zero-valued bits in the binary representation of quantized weights across all layers. Fig. 4(b) presents our method achieving both higher sparsity and lower overall weight magnitude, which satisfies our requirements for thermal-aware quantization search stage.

C. Convergence Analysis

Fig. 5 shows the searching process of the quantization bit-widths on ResNet18 with CIFAR10 dataset. Our proposed method converges within approximately 100 epochs, significantly faster than HAQ [19], which requires over 200. In additon, we summarize the top-1 accuracy comparison across different methods, including results with and without pretraining. The proposed method with pretraining achieves the highest performance, exceeding HAQ [19] and BSQ [18] by 2.60% and 3.30%.



HAQ [19]		81.90%
BSQ [18]		81.20%
Proposed	**w/o pretrain**	82.80%
	w/ pretrain	**84.50%**

(b)

Fig. 5. (a) Convergence analysis on ResNet18 with CIFAR10 dataset. (b) Accuracy comparison on ResNet18 with CIFAR10 dataset.

D. Comparisons

As shown in Fig. 6, the proposed method achieves the highest top-1 accuracy on both CIFAR10 and ImageNet datasets. Specifically, the ImageNet top-1 accuracy for ResNet18 and ResNet34 are 57.32% and 67.00%, representing an average improvement of 46.37% over fixed 8-bit baseline configurations. Compared to HAQ [19] and BSQ [18], our approach improves accuracy by 6.36%–15.06% on average.

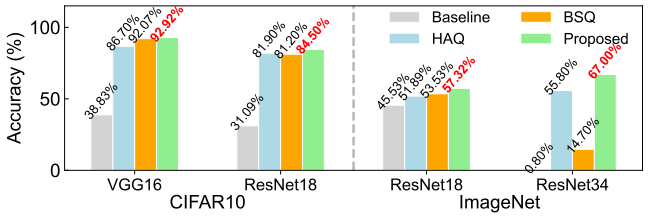


Fig. 6. Comparison of the top-1 accuracy for different methods on CIFAR10 and ImageNet datasets.

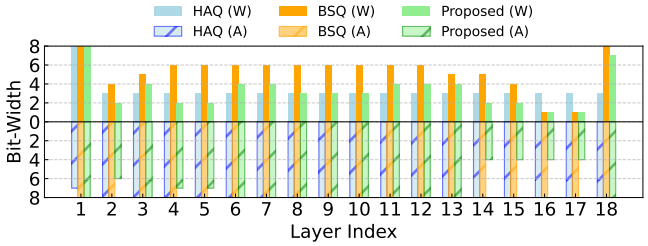


Fig. 7. Quantization bit-width distribution of weight (W) (upper) and activation (A) (lower) across layers for different methods on ResNet18 with CIFAR10.

Fig. 7 presents the bit-width distributions for different methods, the upper plane shows the weights and lower is the activations. BSQ [18] tends to assign lower bit-widths to deeper layers, while HAQ [19] adopts a more uniform assignment. In contrast, our method enables more flexible and adaptive bit-width assignments based on the thermal sensitivity of each layer. Table II summarizes the performance comparison across different benchmarks, including the number of crossbars, energy consumption, and energy efficiency. Our proposed method achieves better performance in most cases, while maintaining competitive accuracy. For eample, on CIFAR10, our method delivers the best energy efficiency, with improvements of ∼3.48× and ∼3.60× over the baseline. The larger improvement on CIFAR10 is due to lower input resolution and smaller network models, where thermal-induced errors accumulate less across layers, enabling our quantization method to better preserve accuracy.

Fig. 8(a-d) present all the 500 candidate solutions in terms of average quantization bit-width versus their corresponding

TABLE II

COMPARISON OF PERFORMANCE METRICS FOR DIFFERENT METHODS

Dataset	DNN	#Crossbar				Energy (mJ)				Energy Efficiency (GOPs/W)			
		Baseline	HAQ [19]	BSQ [18]	Proposed	Baseline	HAQ [19]	BSQ [18]	Proposed	Baseline	HAQ [19]	BSQ [18]	Proposed
CIFAR10	VGG16	7592	3458	2403	**2167**	20.14	10.93	7.33	**5.79**	31.10	57.34	85.50	**108.13**
	ResNet18	7032	2662	3283	**2072**	9.84	2.81	5.05	**2.74**	31.25	109.53	60.93	**112.30**
ImageNet	ResNet18	7256	3531	4103	**3173**	9.87	**1.13**	1.37	1.16	31.26	**66.49**	54.64	64.87
	ResNet34	11232	5127	**4233**	6066	4.98	1.91	**1.54**	2.34	30.25	78.83	**97.55**	64.32

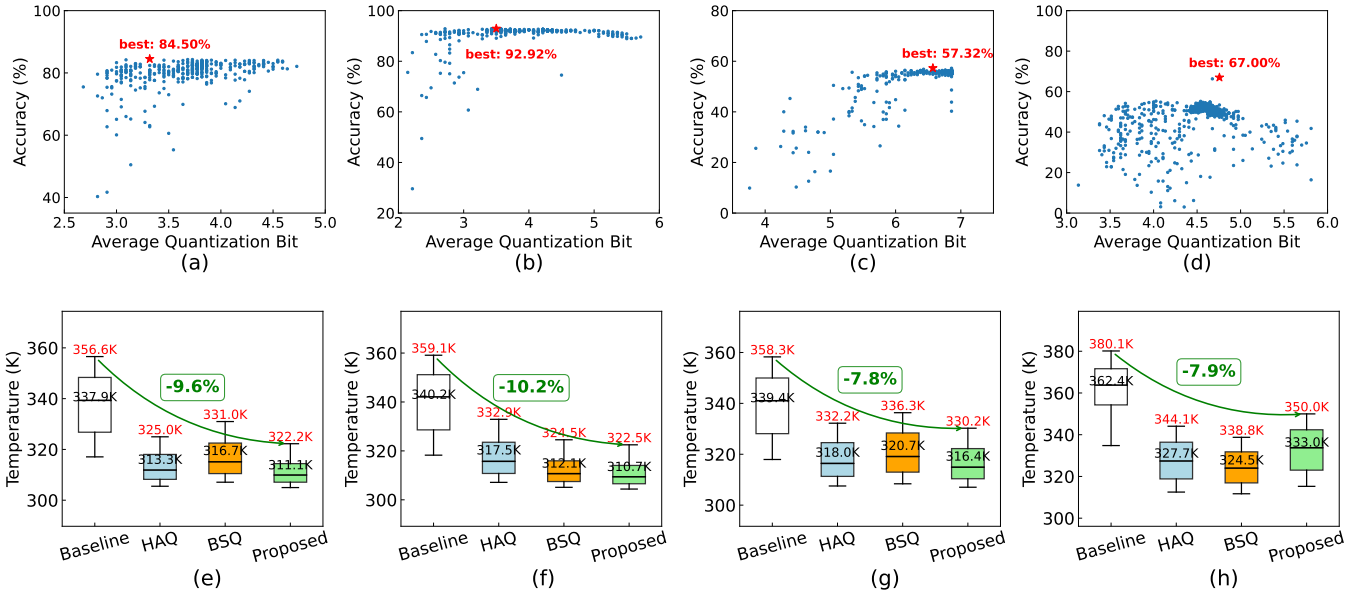


Fig. 8. Quantization solutons during the 500 search epochs on different benchmarks in (a) ResNet18 with CIFAR10, (b) VGG16 with CIFAR10, (c) ResNet18 with ImageNet, and (d) ResNet34 with ImageNet. Comparison of temperature distribution for (e) ResNet18 with CIFAR10, (f) VGG16 with CIFAR10, (g) ResNet18 with ImageNet, and (h) ResNet34 with ImageNet.

accuracy on different benchmarks. The clustering of points in a compact region indicates that the search process converges effectively during exploration. Fig. 8(e-h) show the temperature distribution of the whole model with different methods. The box represents the range of 25%∼75% of the total temperature, and the line in the box indicates the median value. The proposed method not only lowers the peak temperature but also reduces temperature variance. Compared to the baseline, the peak temperature is reduced by 7.8%-10.2%.

Accuracy remains the foremost priority in neural network deployment. Although BSQ [18] achieves lower energy consumption and temperatures for ResNet34 through aggressive bit-width reduction, it incurs severe accuracy degradation (14.70% vs. 52.30%, as shown in Fig. 6). This further demonstrates that these methods primarily optimize for energy efficiency and compression ratio without considering thermal effects. In contrast, our method partially sacrifices energy efficiency, as ensuring robustness often requires assigning higher bit-widths to thermally sensitive layers. This trade-off is acceptable since preserving inference accuracy under thermal stress is more crucial, and our method still provides a balanced improvement across accuracy, energy, and thermal stability.

V. CONCLUSION

In this work, we present a novel RL-guided thermal-aware layer-wise quantization framework for ReRAM-based CIM systems. By jointly optimizing weight sparsity and layer-wise bit-width configurations using evaluation feedback, our approach mitigates thermal degradation while preserving accuracy and efficiency. Compared to existing methods, our framework achieves superior trade-offs in temperature, energy, and inference accuracy. This framework provides a promising foundation for future research on thermally adaptive quantization in ReRAM-based CIM systems.

REFERENCES

[1] L. Song, X. Qian, H. Li, and Y. Chen, "PipeLayer: A Pipelined ReRAM-Based Accelerator for Deep Learning," in *IEEE International Symposium on High Performance Computer Architecture (HPCA)*, 2017, pp. 541–552.

[2] Z. He, J. Lin, R. Ewetz, J.-S. Yuan, and D. Fan, "Noise Injection Adaption: End-to-End ReRAM Crossbar Non-Ideal Effect Adaption for Neural Network Mapping," in *ACM/IEEE Design Automation Conference (DAC)*, 2019, pp. 1–6.

[3] B. Liu, H. Li, Y. Chen, X. Li, Q. Wu, and T. Huang, "Vortex: Variation-aware training for memristor X-bar," in *ACM/EDAC/IEEE Design Automation Conference (DAC)*, 2015, pp. 1–6.

[4] C. Walczyk, D. Walczyk, T. Schroeder, T. Bertaud, M. Sowinska, M. Lukosius, M. Fraschke, D. Wolansky, B. Tillack, E. Miranda *et al.*, "Impact of Temperature on the Resistive Switching Behavior of Embedded HfO_2-Based RRAM Devices," *IEEE Transactions on Electron Devices*, vol. 58, no. 9, pp. 3124–3131, 2011.

[5] H. Shin, M. Kang, and L.-S. Kim, "A Thermal-aware Optimization Framework for ReRAM-based Deep Neural Network Acceleration," in *IEEE/ACM International Conference On Computer Aided Design (ICCAD)*, 2020, pp. 1–9.

[6] X. Liu, M. Zhou, T. S. Rosing, and J. Zhao, "HR3AM: A Heat Resilient Design for RRAM-based Neuromorphic Computing," in *IEEE/ACM International Symposium on Low Power Electronics and Design (ISLPED)*, 2019, pp. 1–6.

[7] Y. Ling, Z. Wang, Z. Yu, S. Bao, Y. Yang, L. Bao, Y. Sun, Y. Cai, and R. Huang, "Temperature-Dependent Accuracy Analysis and Resistance Temperature Correction in RRAM-Based In-Memory Computing," *IEEE Transactions on Electron Devices*, vol. 71, no. 1, pp. 294–300, 2024.

[8] M. V. Beigi and G. Memik, "Thermal-aware Optimizations of ReRAM-based Neuromorphic Computing Systems," in *ACM/ESDA/IEEE Design Automation Conference (DAC)*, 2018, pp. 1–6.

[9] C. Zhang, Y. Ma, and P. Zhou, "Thermal-Aware Layout Optimization and Mapping Methods for Resistive Neuromorphic Engines," in *Asia and South Pacific Design Automation Conference (ASP-DAC)*, 2022, pp. 50–55.

[10] X. Wu, E. Hanson, N. Wang, Q. Zheng, X. Yang, H. Yang, S. Li, F. Cheng, P. P. Pande, J. R. Doppa *et al.*, "Block-Wise Mixed-Precision Quantization: Enabling High Efficiency for Practical ReRAM-Based DNN Accelerators," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 43, no. 12, pp. 4558–4571, 2024.

[11] P.-Y. Chen, F.-Y. Gu, Y.-H. Huang, and I.-C. Lin, "WRAP: Weight RemApping and Processing in RRAM-based Neural Network Accelerators Considering Thermal Effect," in *Design, Automation & Test in Europe Conference & Exhibition (DATE)*, 2022, pp. 1245–1250.

[12] J. Meng, I. Yeo, W. Shim, L. Yang, D. Fan, S. Yu, and J.-S. Seo, "Sparse and Robust RRAM-based Efficient In-memory Computing for DNN Inference," in *IEEE International Reliability Physics Symposium (IRPS)*, 2022, pp. 1–6.

[13] S. Qu, B. Li, S. Zhao, L. Zhang, and Y. Wang, "A Coordinated Model Pruning and Mapping Framework for RRAM-Based DNN Accelerators," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 42, no. 7, pp. 2364–2376, 2023.

[14] A. Shafiee, A. Nag, N. Muralimanohar, R. Balasubramonian, J. P. Strachan, M. Hu, R. S. Williams, and V. Srikumar, "ISAAC: A Convolutional Neural Network Accelerator with In-Situ Analog Arithmetic in Crossbars," in *ACM/IEEE Annual International Symposium on Computer Architecture (ISCA)*, 2016, pp. 14–26.

[15] Z. Zhu, H. Sun, Y. Lin, G. Dai, L. Xia, S. Han, Y. Wang, and H. Yang, "A Configurable Multi-Precision CNN Computing Framework Based on Single Bit RRAM," in *ACM/IEEE Design Automation Conference (DAC)*, 2019, pp. 1–6.

[16] B. Li, S. Qu, and Y. Wang, "An Automated Quantization Framework for High-Utilization RRAM-Based PIM," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 41, no. 3, pp. 583–596, 2022.

[17] S. Huang, A. Ankit, P. Silveira, R. Antunes, S. R. Chalamalasetti, I. El Hajj, D. E. Kim, G. Aguiar, P. Bruel, S. Serebryakov *et al.*, "Mixed Precision Quantization for ReRAM-based DNN Inference Accelerators," in *Asia and South Pacific Design Automation Conference (ASP-DAC)*, 2021, pp. 372–377.

[18] H. Yang, L. Duan, Y. Chen, and H. Li, "BSQ: Exploring Bit-Level Sparsity for Mixed-Precision Neural Network Quantization," in *International Conference on Learning Representations (ICLR)*, 2021.

[19] K. Wang, Z. Liu, Y. Lin, J. Lin, and S. Han, "HAQ: Hardware-Aware Automated Quantization With Mixed Precision," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[20] S. Zhang, G. L. Zhang, B. Li, H. H. Li, and U. Schlichtmann, "Aging-aware Lifetime Enhancement for Memristor-based Neuromorphic Computing," in *Design, Automation & Test in Europe Conference & Exhibition (DATE)*, 2019, pp. 1751–1756.

[21] A. Kaul, Y. Luo, X. Peng, M. Manley, Y.-C. Luo, S. Yu, and M. S. Bakir, "3-D Heterogeneous Integration of RRAM-Based Compute-In-Memory: Impact of Integration Parameters on Inference Accuracy," *IEEE Transactions on Electron Devices*, vol. 70, no. 2, pp. 485–492, 2023.

[22] Y. Zhang, Y. Zhang, and M. S. Bakir, "Thermal Design and Constraints for Heterogeneous Integrated Chip Stacks and Isolation Technology Using Air Gap and Thermal Bridge," *IEEE Transactions on Components, Packaging and Manufacturing Technology*, vol. 4, no. 12, pp. 1914–1924, 2014.

[23] Z. Zhu, H. Sun, T. Xie, Y. Zhu, G. Dai, L. Xia, D. Niu, X. Chen, X. S. Hu, Y. Cao *et al.*, "MNSIM 2.0: A Behavior-Level Modeling Tool for Processing-In-Memory Architectures," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 42, no. 11, pp. 4112–4125, 2023.

[24] A. Krizhevsky, "Learning Multiple Layers of Features from Tiny Images," 2009. [Online]. Available: https://api.semanticscholar.org/CorpusID:18268744

[25] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009, pp. 248–255.