An Energy-Efficient Gate-Controlled RRAM-Based Analog Ising Machine

Congcong Hong[1,2], Yongxiang Li[1,2], Yubiao Luo[1,2], Pushen Zuo[1,2], Zheng Miao[1,2], and Zhong Sun[1,2,3]
[1]*Institute for Artificial Intelligence, Peking University, Beijing, China*
[2]*School of Integrated Circuits, Peking University, Beijing, China*
[3]*Beijing Advanced Innovation Center for Integrated Circuits, Beijing, China*
Email: zhong.sun@pku.edu.cn

***Abstract*—This paper presents an energy-efficient gate-controlled RRAM-based analog Ising machine that operates fully in the continuous-time domain. By employing gate-controlled feedback through the word line, the proposed circuit achieves rapid Ising problem solving via matrix-vector multiplication and spin-state update entirely in the analog domain. A 1T1R array with in-house Pt/Ta/HfO_2/Pt RRAM devices was fabricated using a CMOS process. Experiments on 8-vertex Max-Cut problems show that the proposed circuit achieves an average approximation ratio above 99%. Scalable simulations on 64-vertex problems yield a time-to-solution of 11.3 µs, along with a 4.87× enhancement in array power efficiency and a 4.44× improvement in energy-to-solution relative to the prior state-of-the-art work, thanks to the partial turn-off of array cells. These results demonstrate that the proposed design serves as a high-performance and energy-efficient hardware platform for combinatorial optimization.**

***Index Term*—analog computing, Ising machine, Hopfield neural network, 1T1R RRAM, memristor.**

I. Introduction

Combinatorial optimization problems (e.g., Max-Cut, VLSI design) are NP-hard, with state space growing exponentially with problem size [1]. To improve solving efficiency, physics-inspired models such as Hopfield neural network (HNN) and Ising machine have been proposed [2]. However, conventional approaches suffer from high computational overhead due to intensive matrix operations and data movement. Memristor-based computing-in-memory (CIM) architectures accelerate Ising solvers by performing matrix-vector multiplication (MVM) directly in the crossbar [3-5]. Yet, hybrid analog-digital designs rely on DACs/ADCs and clock synchronization, limiting them to discrete-time operation and incurring significant conversion overhead.

Fully analog, continuous-time Ising machines eliminate clock dependency by embedding spin updates in analog circuitry [6-8]. Conventional RRAM-based computing schemes (1R or 1T1R) use a feedback loop between bit line (BL) and source line (SL) (Fig. 1a), requiring orthogonal BL and SL and sustaining static power due to always-on word lines (WLs) [9-11]. By leveraging the transistor's gate rather than treating it as a mere selector, the 1T1R device allows the active computing ports to be flexibly selected. The design that reconfigures the feedback path to WL-BL or WL-SL as the active port pair, only requires orthogonality between the active ports—a condition satisfied by both crossbar arrays and parallel arrays (i.e., memory-structure arrays, where BL and SL are parallel but WL is orthogonal to both).

Building upon this insight, we propose an energy-efficient gate-controlled RRAM-based analog Ising machine that implements the HNN update rule (Fig. 1b). The circuit is implemented in a 1T1R crossbar array with a closed-loop feedback that leverages WL-BL as active computing ports. It is also compatible with parallel array where either WL-BL or WL-SL can serve as the active port pair. This configuration, together with operational amplifiers (OPAs) and comparators, forms a fully analog continuous-time Ising machine that achieves fast solving speed by performing MVM and spin-state updates entirely within the analog domain. Furthermore, since the WL directly encodes the binary state {0,1} of each spin, a low level on the WL (corresponding to state 0) turns off the access transistor, eliminating the current path through the corresponding cell. This mechanism applies during both transient evolution and steady-state operation, keeping a substantial portion of array cells in the off state and thereby significantly reducing array power consumption.

II. Circuit Implementation

A. Solving Principle of Hopfield Neural Network

The dynamic process of HNN aims to minimize its defined energy function. For a binary state vector $\boldsymbol{x} = (x_1, x_2, \cdots, x_n)^T$, the energy function is typically defined as:

$$E = -\sum_{i,j} w_{ij} x_i x_j + \sum_i h_i x_i = -\frac{1}{2}\boldsymbol{x}^T \boldsymbol{W}\boldsymbol{x} + \boldsymbol{h}^T \boldsymbol{x} \quad (1)$$

where $\boldsymbol{W} = (w_{ij}) \in R^{n\times n}$ is a symmetric weight matrix and $\boldsymbol{h} = (h_i) \in R^n$ is a bias vector. The network follows an update rule: for any node *i*, if $\sum_j w_{ij} x_j \geq h_i$, then set $x_i = 1$; otherwise, set $x_i = 0$. It can be shown that each state update guarantees that the energy E does not increase [2], thereby enabling the network to eventually converge to a local minimum of the energy function. This principle forms the foundation for using the HNN to solve combinatorial optimization problems.

B. Fully Analog Circuit Implementation

By exploiting the formal equivalence between the Ising model and the HNN, we implement an analog Ising machine through a gate-controlled RRAM-based circuit, as shown in Fig. 1b. The core of the circuit is an *n*×*n* 1T1R array, where the RRAM conductance matrix is denoted as $\boldsymbol{G}$. It is supplemented by three peripheral circuits: *n* transimpedance amplifiers (TIAs) and *n* inverting amplifiers (with a gain of -1)—both implemented using OPAs—along with *n* voltage comparators.

For each *i* (*i* = 1, 2, …, *n*), the source line SL_i is tied to a fixed voltage V_0. The bit line BL_i connects to the input of the *i*-th TIA, which has a transimpedance gain of R. Its output feeds into the input of the *i*-th inverting amplifier. The output of the inverting amplifier then drives the non-inverting input of the *i*-th voltage comparator, while the inverting input of the

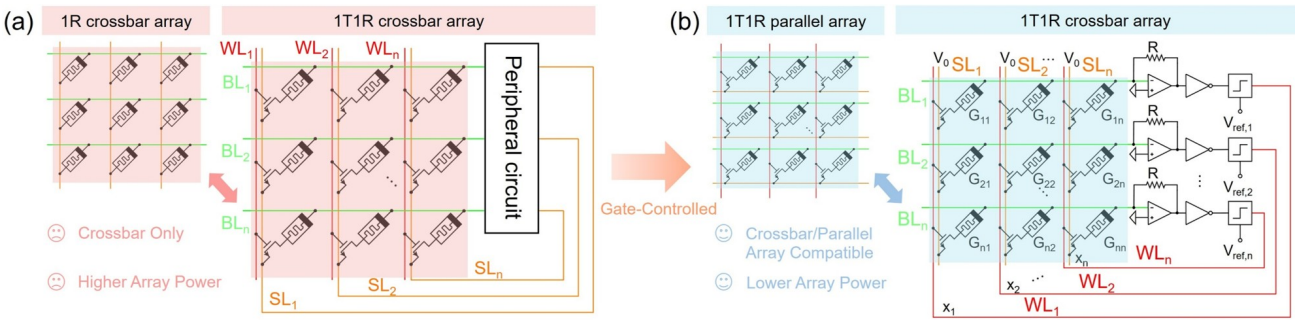

Fig. 1. Comparison of conventional and proposed RRAM-based Ising machines. (a) BL-SL closed-loop feedback in 1R or 1T1R crossbar arrays. (b) Proposed WL-BL closed-loop feedback using OPAs and comparators in 1T1R crossbar array; the architecture is also compatible with 1T1R parallel array using either WL-BL or WL-SL feedback.

comparator is set to a fixed reference voltage $V_{ref,i}$. The comparator output directly drives WL_i, and the corresponding spin state x_i is obtained by reading the voltage on WL_i.

The circuit operates as follows. The voltage comparators are supplied by a single rail, V_{DD}. Consequently, the i-th comparator output, i.e., WL_i voltage, is binary: $V_{WL_i} \in \{0, V_{DD}\}$. The mapping from this voltage to the spin state is defined as: $x_i = 1$ when $V_{WL_i} = V_{DD}$, and $x_i = 0$ when $V_{WL_i} = 0$.

The core network computation occurs at the input of the TIAs. According to Kirchhoff's Current Law, Ohm's Law, and the OPA's virtual short and virtual open principles, the input current I_i flowing into the i-th TIA is the sum of the currents from all RRAM branches connected to BL_i, i.e., $I_i = \sum_{j=1}^{n} I_{ij}$. The current I_{ij} in the j-th row branch is determined by the RRAM conductance G_{ij}, the SL_j voltage V_0, and the WL_j voltage V_{WL_j}: when $V_{WL_j} = V_{DD}$ (corresponding to $x_j = 1$), the corresponding transistor is turned on, giving $I_{ij} = G_{ij}V_0$; when $V_{WL_j} = 0$ (corresponding to $x_j = 0$), the transistor is off, and $I_{ij} = 0$. Therefore, the total current can be uniformly expressed as $I_i = V_0 \sum_j G_{ij} x_j$. This current is converted to a voltage by the TIA, producing the output $V_{TIA,i} = -RI_i = -RV_0 \sum_j G_{ij} x_j$. After passing through the inverting amplifier, the voltage presented to the non-inverting input of the comparator becomes $V_{comp+,i} = RV_0 \sum_j G_{ij} x_j$. The comparator compares $V_{comp+,i}$ with the reference voltage $V_{ref,i}$ and updates its output accordingly:

$$V_{WL_i} = \begin{cases} V_{DD} & (x_i = 1), \ if\ RV_0 \sum_j G_{ij} x_j \geq V_{ref,i} \\ 0 & (x_i = 0), \quad otherwise \end{cases} \quad (2)$$

A comparison with the update rule of the HNN reveals that the two are identical in mathematical form. By establishing the following mapping:

$$G_{ij} = w_{ij} G_0 \, , V_{ref_i} = h_i R G_0 V_0 \quad (3)$$

where G_0 is the unit conductance. The dynamic evolution of the proposed circuit becomes equivalent to minimizing the energy function $E = -\frac{1}{2}\boldsymbol{x}^T \boldsymbol{W}\boldsymbol{x} + \boldsymbol{h}^T \boldsymbol{x}$, thereby enabling the solution of combinatorial optimization problems.

III. Experimental Demonstration

A. Fabrication and Characterization of the 1T1R Array

A 1T1R array was fabricated to validate the proposed circuit architecture. Fig. 2a presents the micrograph of the fabricated array. The transistor array was implemented using a commercial CMOS process, while the RRAM devices were prepared in-house with a Pt/Ta/HfO_2/Pt stack (Fig. 2b).

Fig. 3a shows the set/reset I-V curves of the cell under eight different gate voltages, with each condition tested over 20 cycles. The non-volatile characteristic of the fabricated RRAM was verified through retention measurements. As shown in Fig. 3b, eight distinct conductance states were programmed and monitored for over 10^5 seconds at room temperature. Fig. 3c presents the endurance characteristics of the RRAM device, demonstrating reliable programming after 10^5 cycles. Fig. 3d plots the BL-to-SL equivalent conductance of the 1T1R cell versus gate voltage V_g for various programmed RRAM conductance G_{RRAM}. Below the transistor threshold, the conductance is 4.5 nS. As V_g exceeds threshold, the conductance increases and saturates, remaining essentially unchanged beyond V_g = 3V, indicating that a WL high voltage of 3V or higher is sufficient for proper operation. The BL-to-SL equivalent conductance can be approximated as G_{RRAM}, since the transistor's ON-state conductance is much larger.

B. Circuit Functional Verification

Fig. 4a shows the PCB-based experimental setup for the proposed Ising machine. The SL bias voltages and reference voltage vector $\boldsymbol{V}_{ref}$ are provided by DACs, while the WL voltages are read out via ADCs. It is worth noting that the binary nature of WL outputs allows the use of simple 1-bit ADCs, which would significantly lower power and area consumption. An MCU mounted on the backside of the PCB coordinates all data and control signals during the experiment. Fig. 4b shows the conductance matrix $\boldsymbol{G}$ and $\boldsymbol{V}_{ref}$ programmed for a representative 8×8 problem instance in the functional verification experiment. In this experiment, the SL-BL voltage difference was set to V_0 = 0.1 V, with a reference ground of V_{ground} = 0.6 V. The OPAs and voltage comparators were powered from a single 5 V supply, with a transimpedance gain R = 2 kΩ. The conductance matrix $\boldsymbol{G}$ was mapped from $\boldsymbol{W}$ within the range of 10 μS to 150 μS, with a unit conductance of G_0 = 100 μS. The mapping error was controlled below 5% through write-and-verify operations.

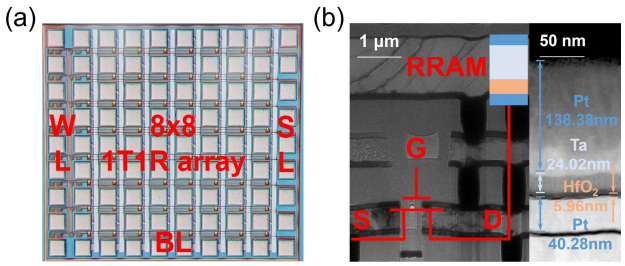

Fig. 2. (a) Micrograph of the fabricated array. (b) TEM image of a transistor (left) and RRAM (right).

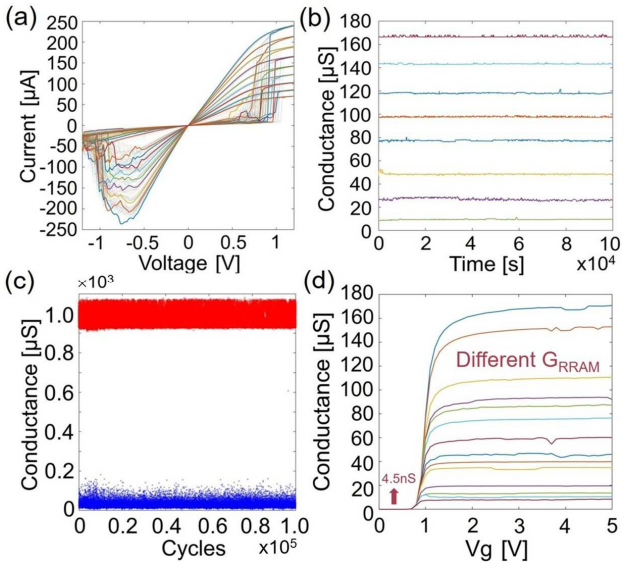

Fig. 3. Experimental characterization of the 1T1R cell. (a) I-V characteristics of the 1T1R cell under various gate voltages. (b) Retention performance of eight conductance states. (c) Endurance characteristics of the RRAM device. (d) BL-to-SL equivalent conductance versus V_g and G_{RRAM}.

Fig. 4c shows the transient evolution of the output voltages $\boldsymbol{V}_{WL} = (V_{WL_1}, V_{WL_2}, \cdots, V_{WL_8})$ recorded by an oscilloscope during circuit operation. The voltages stabilize within 5 μs after the circuit is triggered, with the corresponding energy converging to the minimum (Fig. 4d), validating the feasibility and effectiveness of the proposed circuit.

C. *Application to the Max-Cut Problem*

The Max-Cut problem is a classic NP-hard combinatorial optimization problem with important applications (Fig. 5a). Here, we use it as an example to demonstrate the effectiveness of the proposed method. Fig. 5b presents the experimental results of the proposed circuit on 150 random 8-vertex Max-Cut instances. The circuit attains an average approximation ratio exceeding 99%, demonstrating the strong optimization capability of the proposed Ising machine.

IV. Simulation Result

To comprehensively evaluate the circuit's performance in solving the Max-Cut problem, we conducted simulations on Simulation Program with Integrated Circuits Emphasis (SPICE) across different problem scales. In simulation, we adopted OPAs with a higher GBW of 600 MHz [12] and comparator models with a shorter delay of 1.5 ns [13],

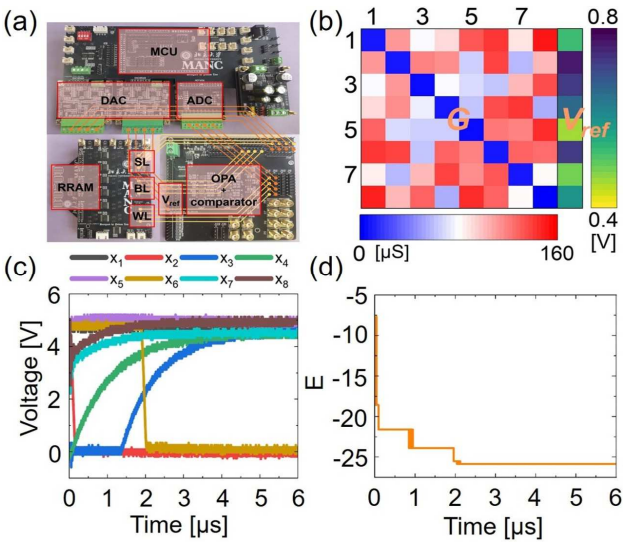

Fig. 4. (a) PCB-based experimental setup for the proposed Ising machine. (b) Heatmap of the conductance matrix $\boldsymbol{G}$ and the reference voltage vector $\boldsymbol{V}_{ref}$. (c) Oscilloscope recording of the transient evolution of eight WL output voltages. (d) Corresponding energy evolution.

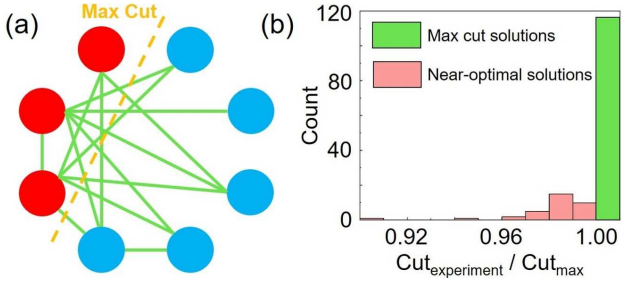

Fig 5. (a) Example 8-vertex Max-Cut problem. (b) Histogram of the normalized cut value for 150 random 8-vertex Max-Cut instances. A value of 1 indicates the solution reached the global maximum cut.

enabling faster voltage state switching than in the physical experiment. Fig. 6a illustrates an example 64-vertex Max-Cut problem used in the simulation. Fig. 6b shows the transient simulation waveform for the 64-vertex problem, where the WL voltage converges within 60 ns, and the corresponding cut value exhibits an overall increasing trend before stabilizing at the maximum. The occasional decreases observed during the evolution result from the synchronous updates of the WL voltage states, an inherent mechanism that enhances the probability of escaping local optima.

For each problem size from 8 to 64 vertices, 20 random Max-Cut instances were generated. For each instance, 100 Monte Carlo simulations were performed under random initial WL voltages, each covering 200 ns of circuit evolution. To increase the probability of escaping local optima, a simulated annealing (SA) mechanism was implemented by injecting mutually independent noise currents into each BL: $I_{i,noise} = Me^{-t/\tau}w_i(t)$, where $M = 0.3A$ is the amplitude, $\tau = 50\,ns$ is the decay time constant, and $w_i(t)$ are mutually independent uniformly distributed white noise sources. Fig. 7a shows the success probability of reaching the global optimum with and without SA. Fig. 7b presents the corresponding time-to-solution (TTS) at 99% success probability, which reaches 11.3 μs for the 64-vertex problem.

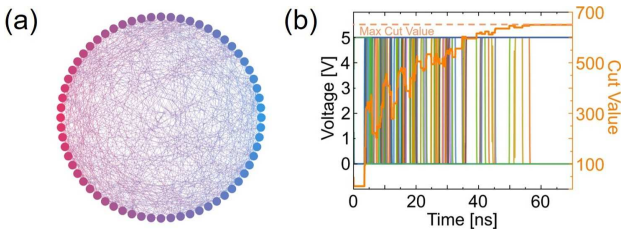

Fig. 6. (a) Illustration of an example 64-vertex Max-Cut problem. (b) Transient waveform of WL voltage for the 64-vertex Max-Cut problem and the evolution of the corresponding cut value.

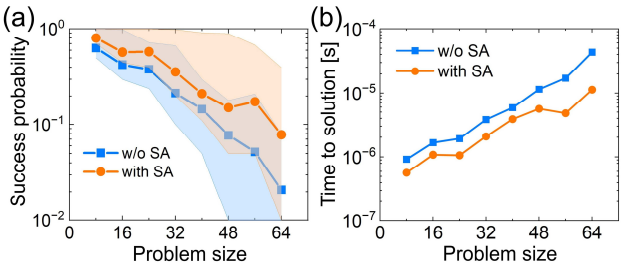

Fig. 7. Simulation results. (a) Success probability versus problem size with and without SA. (b) Corresponding time-to-solution at 99% success probability.

TABLE I presents benchmark results comparing this work with other state-of-the-art solvers. For a fair comparison among RRAM-based implementations, the SL-BL voltage difference was uniformly set to 0.1 V. In the proposed design, the WL-BL or WL-SL feedback architecture keeps approximately half of the WLs low (state 0) on average, turning off the corresponding transistors and reducing array power by half relative to conventional schemes where all cells remain active. As a result, at comparable solving time, the proposed circuit achieves a 4.87× improvement in array power efficiency and a 4.44× improvement in energy-to-solution (ETS) over prior work.

V. Conclusion

In this paper, we proposed a gate-controlled RRAM-based analog Ising machine that solves Ising problems through fast continuous-time analog computing. Experimental results on 8-vertex Max-Cut problems demonstrated near-optimal solutions with an average approximation ratio exceeding 99%, while large-scale simulations on 64-vertex problems achieved a TTS of 11.3 µs. Compared to prior state-of-the-art works, the proposed design offers a 4.87× improvement in array power efficiency and a 4.44× improvement in ETS. These results confirm that gate-controlled analog feedback is a promising approach for energy-efficient combinatorial optimization hardware.

Acknowledgment

This research was funded by the National Natural Science Foundation of China under Grant No. 62572011, the Beijing Natural Science Foundation under Grant No. 4252016, and the 111 Project under Grant No. B18001.

The authors declare that ChatGPT is used exclusively for language polishing to enhance the readability of the manuscript. It is not considered or used as a source of ideas, technical content, or data.

TABLE I. Performance comparison with state-of-the-art solvers

	ESSERC 2025 [8]	Nat. Commun. 2025 [4]	IEDM 2024 [6]	This work
Basic computing element	Ring oscillator	1R / 1T1R	1R / 1T1R	1T1R
Feedback port pair	N/A	BL-SL	BL-SL	WL-BL / WL-SL
Compute paradigm	Continuous-time	Discrete-time	Continuous-time	Continuous-time
Representation of spins	Analog (Voltage Phase)	Digital (Register)	Analog (Voltage Level)	Analog (Voltage Level)
Representation of couplings	Discrete (Coupler Number)	Analog (RRAM Conductance)	Analog (RRAM Conductance)	Analog (RRAM Conductance)
Parallel Array Compatible	N/A	No	No	Yes
Spins	45	42	100	64
Time to solution	3s	553µs	10.3µs	11.3µs
Array power	10.8mW	3.7mW	10mW	2.05mW
Energy to solution (Array)	32.4mJ	2.1µJ	103nJ	23.2nJ

References

[1] C. H. Papadimitriou and K. Steiglitz, *Combinatorial Optimization: Algorithms and Complexity*. Mineola, NY, USA: Dover Publications, 1998.

[2] J. J. Hopfield, "Neural networks and physical systems with emergent collective computational abilities," *Proc. Nat. Acad. Sci.,* vol. 79, no. 8, pp. 2554–2558, Apr. 1982.

[3] W. Wan et al., "A compute-in-memory chip based on resistive random-access memory," *Nature*, vol. 608, no. 7923, pp. 504–512, Aug. 2022.

[4] Y. He et al., "A hardware demonstration of a universal programmable RRAM-based probabilistic computer for molecular docking," *Nat. Commun.*, vol. 17, p. 702, Dec. 10, 2025.

[5] M. Jiang, K. Shan, X. Sheng, C. Graves, J. P. Strachan, and C. Li, "An efficient synchronous-updating memristor-based Ising solver for combinatorial optimization," in *International Electron Devices Meeting (IEDM)*, Dec. 2022, p. 22.2.1-22.2.4.

[6] K. Shan et al., "One-Step Combinatorial Optimization Solver with Fully Integrated Analog Memristors and Annealing Module," in *IEEE International Electron Devices Meeting (IEDM)*, Dec. 2024, pp. 1–4.

[7] R. Wu and P. Liu, "Enhancing Efficiency in Continuous-Time Ising Machine Through Direct Current Comparison," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 72, no. 11, pp. 1790–1794, Nov. 2025.

[8] C. Li et al., "A coupled oscillator based Ising chip with a fully-connected annealing core, a Hamiltonian engine, and a RISC-V gradient search for solving 2,500+ spin combinatorial optimization problems," in *Proc. IEEE Eur. Solid-State Electron. Res. Conf. (ESSERC)*, Sep. 2025, pp. 297–300.

[9] P. Zuo et al., "Precise and scalable analogue matrix equation solving using resistive random-access memory chips," *Nat. Electron.*, vol. 8, pp. 1222–1233, Oct. 13, 2025.

[10] S. Wang and Z. Sun, "Dual in-memory computing of matrix-vector multiplication for accelerating neural networks," *Device*, vol. 2, no. 12, Dec. 2024.

[11] S. Wang et al., "In-memory analog computing for non-negative matrix factorization," *Nat. Commun.*, vol. 17, p. 1881, Jan. 19, 2026.

[12] Analog Devices. " ADA4899-1 Unity-Gain Stable, Ultralow Distortion, 1 nV/√Hz Voltage Noise, High Speed Op Amp," Datasheet Rev. C, May 2016. [Online]. Available: https://www.analog.com/media/en/technical-documentation/data-sheets/ADA4899-1.pdf. Accessed: Dec. 28, 2025.

[13] Texas Instruments, "TLV360x 2.5-ns, High-Speed Comparators With Rail-to-Rail Input," Datasheet Rev. E, Mar. 2023. [Online]. Available: https://www.ti.com.cn/cn/lit/ds/symlink/tlv3601.pdf. Accessed: Oct. 26, 2025.