Memristive Dynamical Spiking Neural Networks with Spatiotemporal Heterogeneity

Xinming Shi[1*], Peng Zhou[2], Connlaoth McTaggart[1], Xin Yao[3]
[1]School of Electronics, Electrical Engineering and Computer Science, Queen's University Belfast, UK
[2]Department of Electrical Engineering, City University of Hong Kong, Hong Kong SAR
[3] School of Data Science, Lingnan University, Hong Kong SAR
Emails: x.shi@qub.ac.uk, pengzhou@cityu.edu.hk, cmctaggart05@qub.ac.uk, xinyao@ln.edu.hk

***Abstract*—We propose a fully memristive spiking neural network (MSNN) that incorporates spatiotemporal heterogeneity to improve temporal representation and fault tolerance. In our proposed work, each neuron possesses a distinct time constant (spatial heterogeneity) that evolves over time in response to input stimuli (temporal heterogeneity), enabling diverse, adaptive, and temporally rich responses. Both synaptic and neuronal behaviors are modeled using SPICE-level analog memristors, and the network is trained end-to-end using backpropagation through time (BPTT) in a differentiable framework. This approach eliminates the need for digital interfacing circuits such as ADCs or explicit comparators, supporting compact and efficient hardware deployment. Evaluations on the MNIST and DVS128 Gesture datasets show competitive accuracy and significantly improved robustness to hardware faults, such as stuck-at errors in RRAM cells. These results demonstrate the effectiveness of spatiotemporally heterogeneous MSNNs for scalable, reliable neuromorphic computing.**

***Index Terms*—Neuromorphic computing, memristor, spiking neural network, heterogeneous neurons, brain-inspired computing.**

I. Introduction

SPIKING Neural Networks (SNNs) impose neuroscience-inspired constraints to modern deep learning algorithms, and have accordingly demonstrated significant improvements in energy efficiency. By transitioning from full-precision activations to temporally encoded data representations, neuromorphic hardware benefits from significant energy and latency reductions [1]–[3].

Training SNNs has historically relied on surrogate gradient methods to overcome the inherent non-differentiability of discrete spikes [4]. However, the emergence of memristive devices—resistive elements capable of stateful, history-dependent analog computation—opens new avenues for building fully differentiable SNNs by leveraging intrinsic device dynamics [5]. These dynamics generate analog voltage spikes that are inherently differentiable and eliminate the need for surrogate gradient approximations.

At the neuronal level, recent studies in neuroscience have revealed that biological neurons exhibit spatiotemporal heterogeneity in their dynamics. For instance, the dendritic processing times and memory time constants (τ) can vary significantly across different neurons and evolve over time even for the same neuron depending on the input context and past activity [6], [7]. Inspired by these observations, we propose to model both spatial heterogeneity (neurons with individually unique, fixed τ values) and temporal heterogeneity (neurons adapting τ over time) in memristive integrate-and-fire (MIF) neurons. This allows each neuron to develop a context-aware memory trace, more closely mimicking biological systems.

In this work, we incorporate such spatiotemporal heterogeneity into the dynamic equations of MIF neurons. We implement this within a fully memristive SNN, trained via backpropagation through time (BPTT) [8]–[10] applied directly to SPICE-based analog models of neurons and synapses. Our experimental results on MNIST and DVS128 Gesture datasets demonstrate that this added heterogeneity yields not only higher accuracy but also enhanced fault tolerance. In particular, networks trained with heterogeneous MIF dynamics exhibit improved robustness to hardware-level faults such as stuck-at-0/1 failures in memristor arrays, consistent with earlier findings that introducing heterogeneity or negative correlation can enhance fault tolerance in evolved circuits [11].

Our main contributions are as follows:

- We propose a biologically inspired spatiotemporally heterogeneous MIF neuron model where both time-varying and neuron-specific memory time constants are integrated into the dynamics.
- We demonstrate how to apply BPTT to fully memristive SNNs with heterogeneous neurons, without the need for surrogate gradients.
- We show that our framework achieves superior accuracy on static and neuromorphic benchmarks (MNIST, DVS128), outperforming baseline homogeneous MSNNs.
- We empirically validate that introducing spatiotemporal heterogeneity significantly enhances the fault tolerance of MSNNs, increasing robustness to common RRAM non-idealities such as stuck-at faults.

II. Background

A. Fully Memristive Spiking Neural Networks

Most fully memristive SNNs use memristors to implement synapses, and in some cases neurons, but often resort to simplified models or surrogate-based training due to the

*Corresponding author: x.shi@qub.ac.uk

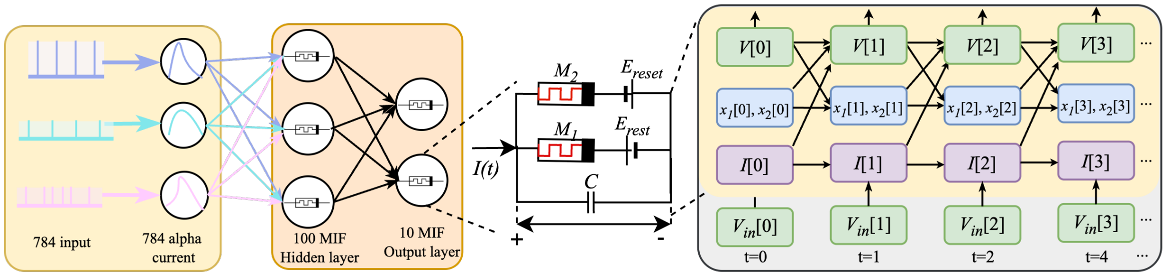

Fig. 1. A fully memristive spiking neural network (MSNN) architecture and its computational graph based on memristive dynamics.

challenges of differentiating spiking behaviors. Some prior works employ unsupervised STDP for learning [12], while others pretrain networks in the digital domain before mapping them to memristive hardware, losing physical realism and fine-grained control across device dynamics [13], [14].

In contrast, our approach explicitly models MIF neurons that capture capacitive and resistive dynamics. These neurons are trained using backpropagation through time (BPTT) applied directly to differentiable analog models. This enables the network to adaptively shape voltage trajectories across time, offering gradient-level access to internal device states.

B. Differentiable Memristive Learning Frameworks

To support training in a fully analog, SPICE-level framework, recent efforts have integrated device models into software simulators such as MemTorch [15], NeuroSim [16], and NeuroPack [17]. However, these typically assume fixed device behaviors during inference or separate learning phases from dynamics. We instead treat the MIF neuron as a dynamic system embedded within the computational graph, where action potentials emerge from voltage evolution governed by memristive states, and gradients are computed accordingly.

C. Heterogeneous Dynamics in Neuromorphic Systems

Biological neurons exhibit remarkable diversity in both spatial and temporal domains. Experimental studies have shown that different neurons, even within the same population, possess distinct membrane time constants, ion channel dynamics, and firing thresholds, contributing to functional robustness and specialization [18]. Moreover, temporal heterogeneity—where a single neuron's dynamics vary depending on input history—enables flexible adaptation to spatiotemporal patterns.

Although recent advances in neuromorphic computing have begun to explore structural diversity—such as synaptic sparsity and nonuniform connectivity patterns—in hardware implementations [19], heterogeneity at the level of neuronal dynamics remains largely unaddressed. This limitation is particularly evident in fully memristive architectures, where both synaptic and neuronal behaviors are typically statically defined by circuit parameters. In contrast, the computational intelligence community has long recognized the potential of heterogeneity. Liu and Yao [20] demonstrated that artificial neural networks comprising diverse computational nodes, evolved via evolutionary algorithms, can exhibit enhanced learning capacity and functional diversity. These insights suggest that embedding dynamic heterogeneity at the circuit level may offer new avenues for improving both the robustness and expressiveness of memristive neuromorphic systems.

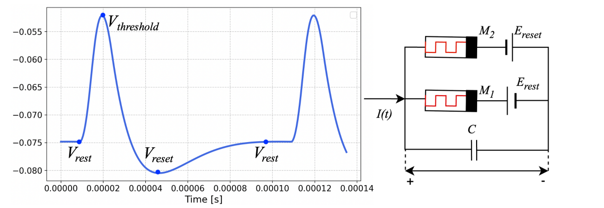

Fig. 2. MIF neuron model comprising two memristors and a capacitor. It reproduces a full spiking cycle—voltage integrates to threshold, triggers reset, and returns to rest—mimicking action potential dynamics.

III. Methods

A. Memristive Integrate-and-Fire Neuron and Fully Memristive Architecture

The neuron model employed in our Memristive Spiking Neural Network (MSNN) is the MIF neuron, illustrated in Fig. 2. Upon injection of a positive current i, the membrane potential v^t begins to rise from its resting potential E_{rest}. Once the potential reaches a critical level, the electric field across memristor M_2 becomes sufficient to switch it on, shorting the output to the reset voltage E_{reset}. Subsequently, the membrane potential discharges and returns to equilibrium at E_{rest}, forming an analog voltage spike as shown in Fig. 2. Our prior experimental validation of this mechanism using Knowm memristors is detailed in [21]:

$$\begin{cases} i = Gv_M = G\dfrac{1}{1+GR}v = G_1 v, \\ \dot{x}_{\text{MR}} = \dfrac{1}{\tau}\left(1+e^{-\beta\left(\frac{1}{1+RG}v - V_{\text{ON}}\right)}\right)(1-x_{\text{MR}}) \\ \qquad - \dfrac{1}{\tau}\left(1-\left(1+e^{-\beta\left(\frac{1}{1+RG}v+V_{\text{OFF}}\right)}\right)^{-1}\right)x_{\text{MR}}, \end{cases} \tag{1}$$

where G and G_1 denote the memductance of the Knowm memristor and its series configuration with a current-limiting resistor, respectively. The internal state $x_{\text{MR}} \in [0,1]$ represents the switching ratio of the device and evolves based on the applied voltage v_{M}. The current through the memristor is i_{M},

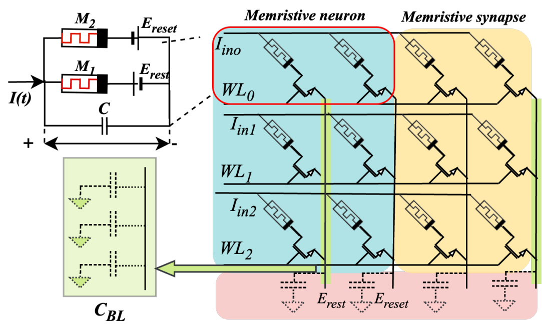

Fig. 3. Schematic of a fully memristive neural network. C_{BL} represents the parasitic bit-line capacitance.

and $\beta = 1/V_T$ is the inverse thermal voltage. Behavior is characterized by parameters R_{ON}, R_{OFF}, V_{ON}, V_{OFF}, and τ.

The underlying memristor behavior follows the generalized metastable switch (MSS) model [22], which captures a broad class of device dynamics. In this framework, a memristor consists of a population of MSS elements that stochastically switch between two internal states based on their voltage and thermal conditions. The normalized internal state $x \in [0,1]$ reflects the average state of all MSS elements in the device.

The evolution of the MIF neuron is governed by a system of coupled differential equations. The membrane voltage v is modulated by the conductances G_1 and G_2 of devices M_1 and M_2, as well as the membrane capacitance C and voltages E_{rest} and E_{reset}. The conductances themselves are nonlinear functions of the internal states x_1 and x_2 and the corresponding on/off resistances R_{on} and R_{off}. The time evolution of x_1 and x_2 is governed by their respective time constants τ_1 and τ_2, thermal voltage V_{th}, and volatility parameter k_v.

To implement this architecture in hardware, we adopt a fully memristive crossbar layout (Fig. 3), where synaptic weights are realized as conductance values within a matrix $\mathbf{G}$, and neuron outputs form a voltage vector $\mathbf{V}$. The resulting current vector $\mathbf{I}$ is computed as Eq. 2 and is used to drive the inputs of downstream MIF neurons. Each current I_n is a weighted sum of voltages from the previous layer (Eq. 3).

$$\mathbf{I} = \mathbf{G} \times \mathbf{V}, \tag{2}$$

$$I_n = \sum_{i=1}^{m} v_i \cdot w_{ni}. \tag{3}$$

The crossbar structure shown in Fig. 3 includes 5 MIF neurons and a 3×2 memristive crossbar. It receives five input currents I_{in0} to I_{in4} and produces three output currents. E_{rest} and E_{reset} are static voltage sources within the MIF circuit. Bit-line parasitic capacitance C_{BL}, arising from interconnects or transistor parasitics, serves as the effective membrane capacitance.

B. Discretized Integration of MIF Neuron Dynamics

To incorporate the memristive integrate-and-fire (MIF) model into a differentiable framework, we discretize its continuous-time dynamics using the forward Euler method. Compared to higher-order schemes such as backward Euler or Runge-Kutta (RK4), forward Euler offers a simple, explicit update rule that can be efficiently unrolled through time for backpropagation-through-time (BPTT) training. The membrane potential is updated as:

$$v^{t+1} = \frac{I^t - G_1^t(v^t - E_{\text{rest}}) - G_2^t(v^t - E_{\text{reset}})}{C} + v^t, \tag{4}$$

where I^t is the synaptic input current, and G_1^t, G_2^t represent voltage-controlled conductances governing leakage and reset behaviors.

The internal dynamics of the memristive components are captured by gating variables x_1 and x_2, which evolve nonlinearly with membrane voltage. For example, x_1 is updated according to:

$$x_1^{t+1} = \frac{1}{\tau_1}\left(\frac{1-x_1^t}{1+\exp\left(\frac{v_{\text{on1}}-(v^t-E_{\text{rest}})}{V_{\text{th}}\cdot k_{\text{v}}}\right)} - \frac{x_1^t}{1+\exp\left(\frac{(v^t-E_{\text{rest}})-v_{\text{off1}}}{V_{\text{th}}\cdot k_{\text{v}}}\right)}\right)+x_1^t \tag{5}$$

To replace idealized Dirac delta spike inputs with physically plausible signals, we adopt alpha-shaped synaptic currents modulated by spike trains. These are governed by:

$$\tau_{\text{syn}}\frac{\mathrm{d}I}{\mathrm{d}t} = a - I, \quad \tau_{\text{syn}}\frac{\mathrm{d}a}{\mathrm{d}t} = -a + W_i\sum_n \delta(t - t_i^n), \tag{6}$$

where a is an internal current state, and W_i is the synaptic weight. This formulation is compatible with analog circuits such as memristors and captures graded responses commonly observed in sensory neurons [23].

All variables—including v, x_1, x_2, and τ—are embedded in a differentiable computational graph. The result is a forward-mode simulation framework that preserves the physical behavior of the MIF neuron while enabling efficient, gradient-based training in neuromorphic learning systems.

C. Spatiotemporal Heterogeneity in MSNN

To enhance fault tolerance and biological fidelity, we incorporate both spatial and temporal heterogeneity into the MIF neuron model.

Spatial heterogeneity, characterized by neuron-specific time constants τ, is introduced at initialization to emulate the diversity observed in biological neural systems [21], [22].

Temporal heterogeneity allows τ to evolve during simulation based on internal history, enabling neurons to adapt their integration dynamics over time. This is modeled through dynamic updates of τ_1 and τ_2 that depend on recent voltage history, introducing memory-dependent plasticity into the system.

Specifically, the time constant τ dynamically evolves with the membrane potential v as:

$$\tau = \tau_{\min} + (\tau_{\max} - \tau_{\min}) \cdot \sigma\left(\frac{v}{\zeta}\right), \tag{7}$$

where $\sigma(\cdot)$ denotes the sigmoid function. The parameter ζ regulates the steepness of the sigmoid transition, shaping how sensitively τ responds to changes in membrane potential v. Voltage-dependent τ modulates the neuron's internal state update as follows:

$$x_1^{t+1} = \frac{1}{\tau_{\min} + (\tau_{\max} - \tau_{\min}) \cdot \sigma\left(\frac{v^t}{10}\right)} \left(\frac{1 - x_1^t}{1 + \exp\left[\frac{v_{\mathrm{on1}} - (v^t - E_{\mathrm{rest}})}{V_{\mathrm{th}} \cdot k_{\mathrm{v}}}\right]} - \frac{x_1^t}{1 + \exp\left[\frac{(v^t - E_{\mathrm{rest}}) - v_{\mathrm{off1}}}{V_{\mathrm{th}} \cdot k_{\mathrm{v}}}\right]} \right) + x_1^t. \tag{8}$$

Such heterogeneity is differentiable and integrated into the computational graph, enabling learning via gradient descent. This structured variability boosts the model's robustness against hardware non-idealities such as stuck-at faults.

D. Backpropagation Through Time in MSNNs

With the MIF neuron dynamics discretized into explicit recursive forms, our MSNN can be unrolled through time as a directed acyclic graph, enabling gradient-based optimization via BPTT.

Unlike neural ODE frameworks that compute loss only at the final state, we calculate the loss at every time step using the membrane potential v of the output neurons. This temporal loss accumulation is crucial in spiking systems, where information is embedded in the precise timing of neuronal activity. BPTT is thus a more effective training method for capturing the dynamic behavior of the network.

In our framework, the trainable parameters w represent synaptic weights, and the entire spiking mechanism arises from the intrinsic dynamics of the MIF neuron model. Spike generation is governed by physical threshold-switching behavior of memristive elements, rather than relying on artificial hard-thresholding. This allows the full MIF process—including charge integration, gating transitions, and firing—to be captured directly within the computational graph.

Importantly, all dynamic variables, including v, x_1, x_2, and τ, are continuously differentiable and governed by physical equations. This eliminates the need for surrogate gradient approximations, enabling biologically faithful and hardware-compatible training.

The conductance states G_1 and G_2 depend on the internal variables x_1 and x_2, which evolve according to nonlinear functions of membrane voltage. Their dynamics are controlled by parameters such as the state time constants τ_1, τ_2, thermal voltage V_{th}, and a volatility coefficient $k_{\mathrm{v}} \in [0, 1]$ that modulates state retention. As $k_{\mathrm{v}} \to 0$, the system becomes increasingly nonvolatile, consistent with high-retention resistive memory behavior.

Through this training strategy, our MSNN leverages physical memristor dynamics for end-to-end learning—combining gradient-based optimization with device-level realism and compatibility with analog RRAM hardware.

TABLE I
MIF CIRCUIT PARAMETERS

Parameter	**Value**	**Parameter**	**Value**
v_{off_1}, v_{off_2}	5 mV	v_{on_1}, v_{on_2}	110 mV
R_{off_1}, R_{off_2}	0.1 MΩ	R_{on_1}, R_{on_2}	1 kΩ
τ_1, τ_2	1 ms	τ_{syn}	0.64 ms
E_{rest}	0 mV	E_{reset}	50 mV
V_{th}	25 mV	k_{v}	0.6
C	100 pF		

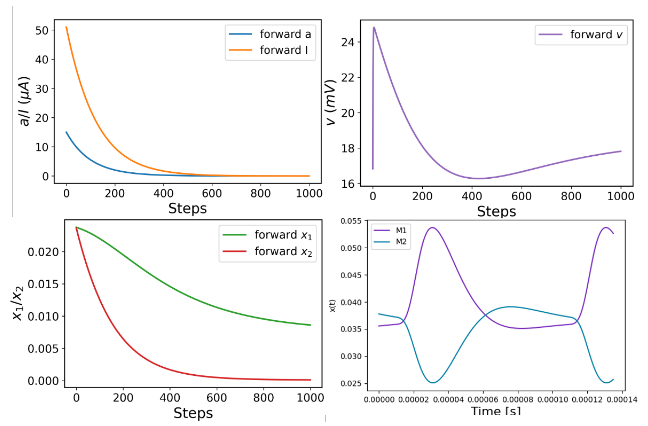

Fig. 4. Simulation results of MIF neuron solved using forward Euler numerical integration.

IV. Experiments

A. MIF Single Neuron Simulation

We conducted a single-neuron simulation in Python over 1,000 steps with a time resolution of 0.01 ms (total duration 10 ms). As shown in Fig. 4, the simulation tracks the evolution of the alpha-shaped current a, integrated current I, and membrane potential v, all of which exhibit biologically plausible decay dynamics. The internal state variables x_1 and x_2 demonstrate a smooth temporal evolution, with x_2 decaying faster than x_1, highlighting distinct decay time constants. Furthermore, the dynamic state trajectories of two memristors (M_1 and M_2) driven by this neuron model are shown to evolve continuously, exhibiting oscillatory yet bounded behavior. These results confirm that the forward Euler method provides a numerically stable and accurate approximation of the MIF neuron and its coupled memristive dynamics.

B. Network Architecture

To evaluate the applicability of MIF neurons within a deep learning framework, we implemented a compact 3-layer fully connected MSNN comprising 100 hidden MIF neurons (Fig. 1). Each simulation spans 1,000 discrete steps over a 10 ms duration.

The input layer receives either 784 features (static datasets) or 2,048 (downsampled neuromorphic data). An alpha current generator, scaled by input intensity and modulated by memristive synaptic weights, drives the hidden MIF neurons. Their outputs are passed through another set of memristive synapses

TABLE II
TEST ACCURACIES FOR THE MNIST AND DVS128 GESTURE DATASETS.

Dataset	**Batch**	**LR**	**Best**	**Avg.** ($n = 5$)	σ	ζ
MNIST	128	1e-4	**93.59**	**93.48**	0.07	10
DVS-128	16	1e-4	**84.52**	**83.83**	0.95	10

[1] LR: learning rate η. [2] σ: standard deviation. [3] ζ: steepness of the sigmoid transition.

to the output MIF layer. To account for loading effects, the input current to each MIF layer is attenuated using a predefined scaling factor.

C. Dataset and Training Setup

To enable fair comparison with SOTA methods, we evaluate the MSNN on MNIST [24] (static 28×28 images) and DVS128 Gesture [25] (event-driven sequences of 11 gestures). MNIST inputs are encoded as alpha-shaped current pulses injected every 100 steps, while DVS128 data are downsampled to $32 \times 32 \times 2$ (ON/OFF channels) and integrated over 100 steps for training and 360 for testing. The MIF model is implemented in PyTorch 1.10.1 (Python 3.8) using forward Euler integration and native autodiff. Networks are trained for 100 epochs using Adam [26], minimizing the negative log-likelihood of output membrane potentials v^t, with gradients propagated through SPICE-level memristive dynamics.

D. Results

1) Accuracy Comparison: The final test accuracy is averaged over 5 trials with up to 100 training epochs and early stopping. The results with standard deviation are shown in Table II. On MNIST, our fully memristive SNN achieves 93.48% accuracy, surpassing prior MSNN baselines without any digital peripherals. Though slightly below full-precision non-memristive models, it leverages only intrinsic device dynamics. On DVS128 Gesture, the MSNN achieves 83.83%, marking the first demonstration of effective training on neuromorphic event-based data using native threshold-switching for spike generation.

Table III summarizes comparisons with prior memristive SNNs. Wang *et al.* [14] employed analog neurons and TDMA reuse, but lacked scalability. Wijesinghe *et al.* [13] used ANN-to-SNN conversion, which restricts learning dynamics and performs poorly on neuromorphic tasks. Duan *et al.* [27] adopted surrogate gradients for hard-threshold LIF neurons, achieving 83.2% but without physical alignment. Zhou *et al.* [28] applied BPTT to analog neurons with fixed τ, reaching 93.2% on MNIST.

In contrast, our MSNN integrates spatiotemporal heterogeneity—each neuron has a unique and input-adaptive time constant—enabling richer temporal encoding and diverse neural responses. This enhances the model's representational capacity, leading to higher accuracy (93.6%) without the need for ADC/DACs, and improved robustness under hardware non-idealities.

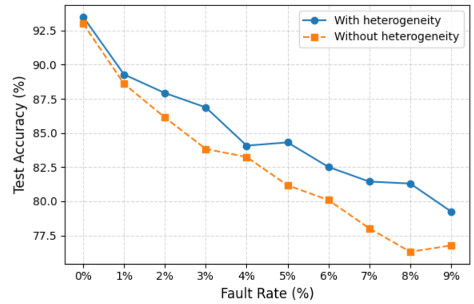

Fig. 5. Test accuracy of MSNN with and without heterogeneity.

2) Fault Tolerance: Fig. 5 presents the test accuracy of MSNNs under varying synaptic fault rates, where each synapse has a fixed probability of being randomly stuck-at-0 or stuck-at-1. The model incorporating spatiotemporal heterogeneity consistently maintains higher accuracy across all fault rates, particularly under moderate to severe fault conditions.

As fault rates rise, the homogeneous model deteriorates rapidly, while the heterogeneous model shows a slower, more resilient decline—exceeding a 2% accuracy gap at 9% fault rate. This robustness arises from the diverse dynamics and functional redundancy introduced by spatiotemporal heterogeneity, which help buffer against synaptic faults.

These results highlight heterogeneity as a key enabler of fault-tolerant memristive SNNs, paving the way for resilient neuromorphic computing on unreliable hardware.

3) Circuit Performance: Our fully memristive SNN (MSNN) removes ADCs and activation serialization, improving efficiency. A 784-100-10 MSNN uses 110 MIF neurons and 79.4k synapses, implemented via 0.16M RRAM devices across 10 tiles (128×128). At 65 nm, this yields a total area of 2.77×10^{-2} mm^2 [29], while an 8-bit ANN with 4 SAR ADCs per tile (3×10^{-3} mm^2 each) occupies $5.33\times$ more area. Assuming 2% neuron activity, $R_{\text{avg}} = 50.5\,\text{k}\Omega$, $v_{\text{avg}} = 57.5\,\text{mV}$, the power is:

$$P_{\text{ave}} = \frac{v_{\text{ave}}^2}{R_{\text{ave}}} \times N_{\text{cells}} \times 2\% = 21.45\ \mu\text{W}, \tag{9}$$

totaling 0.21 mW for 10 tiles—$38.3\times$ lower than 8.21 mW consumed by 40 ADCs. Latency is also reduced: conventional ADC-based designs require 7.44 ms (6.4 µs input + 0.4 µs output $\times$ 1,000 steps + 0.64 ms alpha delay), whereas MSNNs operate purely via analog dynamics with $\sim$100 ns intrinsic RC delays, eliminating conversion overhead.

V. CONCLUSION

We presented a fully memristive spiking neural network (MSNN) framework with spatiotemporal heterogeneity, with direct backpropagation on analog SPICE-level models. By leveraging the intrinsic dynamics of memristive devices, our approach generates naturally differentiable voltage spikes and eliminates the need for digital interfacing circuits or surrogate gradients. The introduction of temporal and spatial heterogeneity enriches neuronal dynamics, enhancing temporal

TABLE III
COMPARISON OF FULLY MEMRISTIVE SNN IMPLEMENTATIONS.

Method	Architecture	Neuron Elements	Refractory	Neuron Type	Heterogeneity	Accuracy	Extra Logic
[14]	1Mem-C-M-C-M-1024F-10F	4(5)+33T	✗	Mixed-signal	✗	97.1%	High
[13]	784-C-M-C-M-10F	5+2T	✗	Digital	✗	∼96%	High
[27]	784-100F-10F	3	✗	Analog	✗	83.2%	Low
[28]	784-100F-10F	3(5)	✓	Analog	✗	93.2%	Low
Ours	784-100F-10F	3(5)	✓	Analog	✓	**93.6%**	Low

T: Transistor; Mem: Memristor; C: Conv layer; M: Max-pooling; F: Fully connected.

encoding and robustness. This is consistent with an earlier observation in a different context [11]. Evaluations on MNIST and DVS128 Gesture datasets show that spatiotemporally heterogeneous MSNNs achieve higher accuracy and robustness, especially under hardware nonidealities such as RRAM stuck-at faults. These results underscore their potential for efficient and reliable neuromorphic computing. Future work will extend MIF neurons throughout the network, incorporate memristive variations, and scale up to more complex real-world tasks.

REFERENCES

[1] X. Wang, X. Lin, and X. Dang, “Supervised learning in spiking neural networks: A review of algorithms and evaluations,” *Neural Networks*, vol. 125, pp. 258–280, 2020.

[2] X. Shi, L. L. Minku, and X. Yao, “Evolving memristive reservoir,” *IEEE Trans. Neural Netw. Learn. Syst.*, pp. 1–15, 2023.

[3] X. Shi, Z. Zeng, L. Yang, and Y. Huang, “Memristor-based circuit design for neuron with homeostatic plasticity,” *IEEE Transactions on Emerging Topics in Computational Intelligence*, vol. 2, no. 5, pp. 359–370, 2018.

[4] E. O. Neftci, H. Mostafa, and F. Zenke, “Surrogate gradient learning in spiking neural networks: Bringing the power of gradient-based optimization to spiking neural networks,” *IEEE Signal Processing Magazine*, vol. 36, no. 6, pp. 51–63, 2019.

[5] L. O. Chua, “Memristor-the missing circuit element,” *IEEE Transactions on Circuit Theory*, vol. 18, no. 5, pp. 507–519, 1971.

[6] F. Ponulak and A. Kasinski, “Introduction to spiking neural networks: Information processing, learning and applications,” *Acta neurobiologiae experimentalis*, vol. 71, no. 4, pp. 409–433, 2011.

[7] X. Shi, L. L. Minku, and X. Yao, “Adaptive memory-enhanced time delay reservoir and its memristive implementation,” *IEEE Transactions on Computers*, vol. 71, no. 11, pp. 2766–2777, 2022.

[8] M. C. Mozer, “A focused backpropagation algorithm for temporal pattern recognition,” in *Backpropagation*. Psychology Press, 2013, pp. 137–169.

[9] A. J. Robinson and F. Fallside, *The utility driven dynamic error propagation network*. University of Cambridge Department of Engineering Cambridge, 1987, vol. 11.

[10] P. J. Werbos, “Generalization of backpropagation with application to a recurrent gas market model,” *Neural networks*, vol. 1, no. 4, pp. 339–356, 1988.

[11] T. Schnier and X. Yao, “Using negative correlation to evolve fault-tolerant circuits,” in *International Conference on Evolvable Systems*. Springer, 2003, pp. 35–46.

[12] Z. Wang, S. Joshi, S. Savel’ev, W. Song, R. Midya, Y. Li, M. Rao, P. Yan, S. Asapu, Y. Zhuo *et al.*, “Fully memristive neural networks for pattern classification with unsupervised learning,” *Nature Electronics*, vol. 1, no. 2, pp. 137–145, 2018.

[13] P. Wijesinghe, A. Ankit, A. Sengupta, and K. Roy, “An all-memristor deep spiking neural computing system: A step toward realizing the low-power stochastic brain,” *IEEE Transactions on Emerging Topics in Computational Intelligence*, vol. 2, no. 5, pp. 345–358, 2018.

[14] J. Wang, S. Hu, X. Zhan, Q. Yu, Z. Liu, T. P. Chen, Y. Yin, S. Hosaka, and Y. Liu, “Handwritten-digit recognition by hybrid convolutional neural network based on hfo2 memristive spiking-neuron,” *Scientific Reports*, vol. 8, no. 1, pp. 1–7, 2018.

[15] C. Lammie, W. Xiang, B. Linares-Barranco, and M. R. Azghadi, “Memtorch: An open-source simulation framework for memristive deep learning systems,” *Neurocomputing*, vol. 485, pp. 124–133, 2022.

[16] P.-Y. Chen, X. Peng, and S. Yu, “Neurosim: A circuit-level macro model for benchmarking neuro-inspired architectures in online learning,” *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 37, no. 12, pp. 3067–3080, 2018.

[17] J. Huang, S. Stathopoulos, A. Serb, and T. Prodromakis, “Neuropack: An algorithm-level python-based simulator for memristor-empowered neuro-inspired computing,” *arXiv preprint arXiv:2201.03339*, 2022.

[18] F. P. Chabrol, A. Arenz, M. T. Wiechert, T. W. Margrie, and D. A. DiGregorio, “Synaptic diversity enables temporal coding of coincident multisensory inputs in single neurons,” *Nature neuroscience*, vol. 18, no. 5, pp. 718–727, 2015.

[19] H. Zheng, Z. Zheng, R. Hu, B. Xiao, Y. Wu, F. Yu, X. Liu, G. Li, and L. Deng, “Temporal dendritic heterogeneity incorporated with spiking neural networks for learning multi-timescale dynamics,” *Nature Communications*, vol. 15, no. 1, p. 277, 2024.

[20] Y. Liu and X. Yao, “Evolutionary design of artificial neural networks with different nodes,” in *Proceedings of IEEE international conference on evolutionary computation*. IEEE, 1996, pp. 670–675.

[21] S. M. Kang, D. Choi, J. K. Eshraghian, P. Zhou, J. Kim, B.-S. Kong, X. Zhu, A. S. Demirkol, A. Ascoli, R. Tetzlaff, and L. O. Chua, “How to build a memristive integrate-and-fire model for spiking neuronal signal generation,” *IEEE Transactions on Circuits and Systems I: Regular Papers*, 2021.

[22] T. W. Molter and M. A. Nugent, “The generalized metastable switch memristor model,” in *CNNA 2016; 15th International Workshop on Cellular Nanoscale Networks and Their Applications*. VDE, 2016, pp. 1–2.

[23] J. K. Eshraghian, S. Baek, J.-H. Kim, N. Iannella, K. Cho, Y. S. Goo, H. H. Iu, S.-M. Kang, and K. Eshraghian, “Formulation and implementation of nonlinear integral equations to model neural dynamics within the vertebrate retina,” *International Journal of Neural Systems*, vol. 28, no. 07, p. 1850004, 2018.

[24] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition,” *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[25] A. Amir, B. Taba, D. Berg, T. Melano, J. McKinstry, C. Di Nolfo, T. Nayak, A. Andreopoulos, G. Garreau, M. Mendoza *et al.*, “A low power, fully event-based gesture recognition system,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 7243–7252.

[26] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *arXiv preprint arXiv:1412.6980*, 2014.

[27] Q. Duan, Z. Jing, X. Zou, Y. Wang, K. Yang, T. Zhang, S. Wu, R. Huang, and Y. Yang, “Spiking neurons with spatiotemporal dynamics and gain modulation for monolithically integrated memristive neural networks,” *Nature Communications*, vol. 11, no. 1, pp. 1–13, 2020.

[28] P. Zhou, D.-U. Choi, W. D. Lu, S.-M. Kang, and J. K. Eshraghian, “Gradient-based neuromorphic learning on dynamical rram arrays,” *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, vol. 12, no. 4, pp. 888–897, 2022.

[29] Q. Wang, X. Wang, S. H. Lee, F.-H. Meng, and W. D. Lu, “A deep neural network accelerator based on tiled rram architecture,” in *2019 IEEE International Electron Devices Meeting (IEDM)*. IEEE, 2019, pp. 14–4.