MesoNet: Optimizing Spiking Neural Networks with Dynamic Saddle Distributions

Krishna Bhatt | San Jose, California (Grade 10)

Artificial neural network (ANN) clustering requires high wattage, limiting use in energy-critical applications. Spiking neural networks (SNNs) offer a low-power alternative; however, the prevalent spike-timing-dependent-plasticity (STDP) algorithm scales poorly, and can reduce accuracy by 40% compared to ANNs. My project addresses these challenges through an investigation of SNN learning dynamics, before developing an algorithm for improved generalization.

After approximating the LIF-kernel via quadratic form, I found STDP follows a stability-switch bifurcation regulated by learning rate. Furthermore, I hypothesized an optimal learning regime at the edge of chaos. Leveraging this, I developed two algorithms–variable plasticity and triangulated attribution– to form dynamic saddle distributions, guiding weights within optimal learning conditions.

MesoNet achieves 56.6% accuracy on CIFAR-10, outperforming prior SNNs by 18.57%. MesoNet lies within 5.05% points of state-of-the-art ANNs while consuming half (47.23%) the energy. This architecture has vast potential for deep-space exploration, medical implants, and remote sensing.

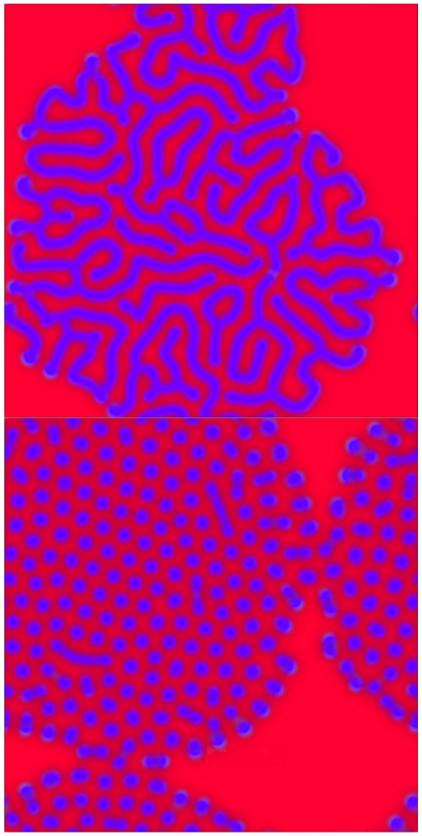


Figure 1. Emergent self-organization with MesoNet

INTRODUCTION

Artificial Neural Network (ANN) **clustering algorithms** require hundreds of watts, considerably limiting **energy critical applications**. A promising solution, SNNs require mere *milliwatts* on **neuromorphic chips.**

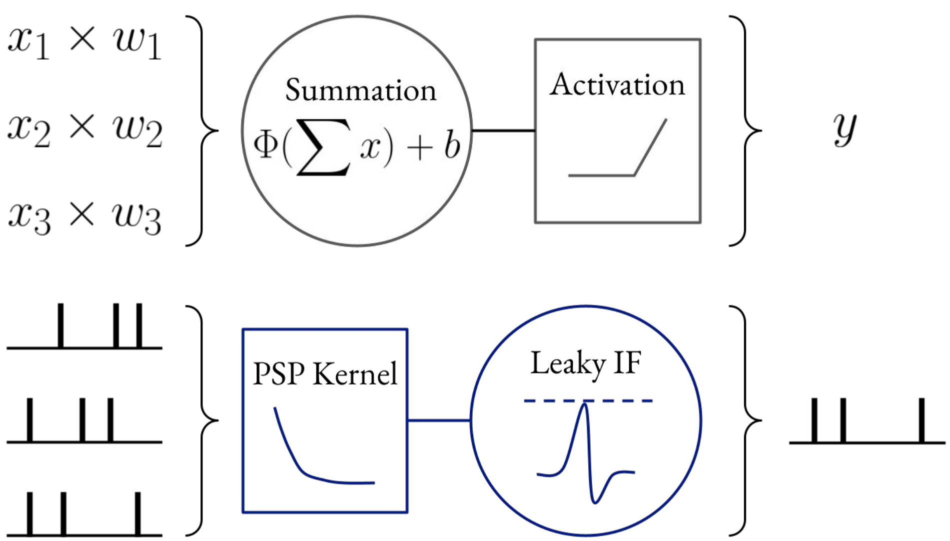


Figure 1. A design comparison of Artificial Neural Networks (top) vs. SNNs (bottom).

Spiking Neural Networks (SNNs)

Spiking Neural Networks (SNNs) closely mimic biological neurons by utilizing discrete events called “spikes” to represent information.

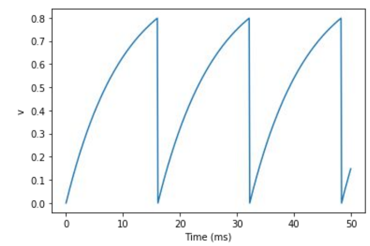


Figure 2. Running voltage plot of a Leaky-Integrate-and-Fire neuron

Figure 3. The Loihi neuromorphic chip. (Image Credit: intel.com)

Much like a capacitor, neurons accumulate input and spike at a threshold. Information is encoded within the spike timings, forming spike trains. Sparse representation makes SNNs highly energy efficient.

STDP (spike timing dependent plasticity) is the prevalent unsupervised learning rule for SNNs, setting synaptic weights based on **spiking correlation**.

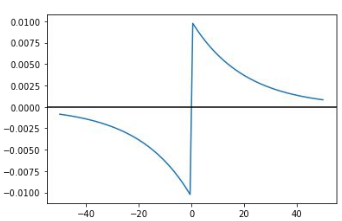


Figure 4. The STDP Function

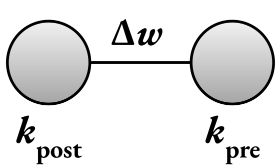


Figure 5. 1:1 LIF Connection

Definition: STDP Learning Rule

Given a synaptic weight $w \in \mathbb{R}$ and spike timing difference $\Delta t = t_{post} - t_{pre}$, the weight update is given as:

$$\Delta w = \begin{cases} A_+ e^{-\frac{\Delta t}{\tau_m}}, & \text{if } \Delta t > 0 \\ A_- e^{\frac{\Delta t}{\tau_m}}, & \text{if } \Delta t < 0 \end{cases}$$

where A_+, A_-, and τ_m are learning parameters. This rule strengthens causal relationships and weakens anti-causal ones, enabling bio-plausible local learning.

INTRODUCTION

Compared to the extensive study of ANNs, little research has explored **internal learning behavior** in SNNs.

Zhang, et al. published a review of the bifurcation dynamics regarding the LIF Neuron; however, their methods focused upon setting hyperparameters, rather than the learning of weights.

Prior Research (Work by Others)

1907	1952	1993	2017	2021
Louis Lapicque describes the first LIF neuron model	Hodgkin and Huxley discover the voltage dynamics of a squid axon	Henry Markram invents STDP based on Hebbian learning	Kheradpisheh develops the convolution SNN	Zhang develops the bifurcation neuron model

Figure 6. Timeline of the most significant contributions towards the field of SNNs.

Research Gap & Approach

Challenges

- SNNs face 'catastrophic forgetting' with large inputs.
- STDP rarely enables deeper layers to generalize on lower level features.
- No viable real-world use case beyond toy datasets (MNIST).

My Approach

- Uncovered nonlinear learning dynamics in SNN systems.
- Investigated information loss and chaos in deep layers.
- Developed a novel **plasticity based** learning algorithm and mathematically proved its improved capability

STDP, the prevalent learning rule, struggles to scale from toy datasets to practical tasks, **reducing accuracy by up to 40%** compared to ANNs (Deng, et al.)

Engineering Goal: Develop an algorithm that builds upon current literature in STDP to improve SNN accuracy, speed, and resource efficiency for larger tasks (eg. CIFAR-10).

Design Constraints: Testing completed on GTX 1080-Ti GPU (Brian2 simulator) rather than on lab-access neuromorphic chips.

Project Origin: While researching neuroimaging last year, I came across bio-plausible neuron simulations. I was fascinated by its possible computational applications and individually began work on this year's project in July 2024, after a literature review.

Continuation: None

PROBLEM FRAMEWORK

Project Development Timeline

STAGE 1	STAGE 2	STAGE 3	STAGE 4
Jul. '24 → Sep. '24 Literature Review and research	**Sep. '24 → Jan. '25** Brainstorming, Failed Attempts, Experimentation	**Jan. '25 → Feb. '25** Composing complete theorem justifications	**Feb. '25 → Mar. '25** Implementation and evaluation

Mathematical Framework

To deconstruct multilayer SNN learning, I took the lens of **dynamical systems theory**, which explains how systems behave over time. This is similar to modeling all the water particles in an ocean interact with each other to form currents.

I modeled multilayer behaviors from the **basic 1:1 synapse case**, focusing on a 3-layer network of LIF Neurons.

The Leaky Integrate and Fire Neuron Model

$$\tau_m \frac{\mathrm{d}V}{\mathrm{d}t} = -(V(t) - V_{\mathrm{rest}}) + R(I(t))$$

where V is voltage, t is timestep, R is conductance, and I is input current.

Definition: The 3-Layered Network

Consider an L-layer feed-forward spiking neural network with width N with synaptic weight matrices $\mathbf{W}^l \in \mathbb{R}^{N \times N}$, membrane constants $\boldsymbol{\tau}_{\mathbf{m}}{}^{l}$, post-synaptic potential $\epsilon\ (t)$, and input synaptic current $\boldsymbol{I}\ (t) = (I_1\ (t), \ldots, I_M\ (t))^{\mathrm{T}}$ denotes the M-dimensional input signals.

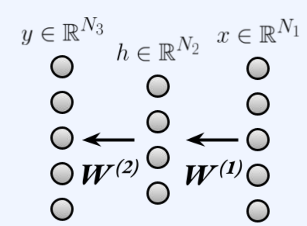


Figure 7. The basic three layer network for preliminary study

Some Key Concepts Used in my Project:

- Multivariable: local linearity, eigenvalues, Jacobian, saddles, etc.
- Dynamical Systems: stability analysis, attractors, bifurcations, lyapunov exponents (chaos)

NOTE: This presentation only covers theorem outcomes; full justifications are in the project journal (Supplemental info).

FINDINGS: EXPLORING LEARNING DYNAMICS

STDP learning forms a feedback loop (Δw→Δt→Δw). To model the base case deterministically, I developed a novel spike approximation technique.

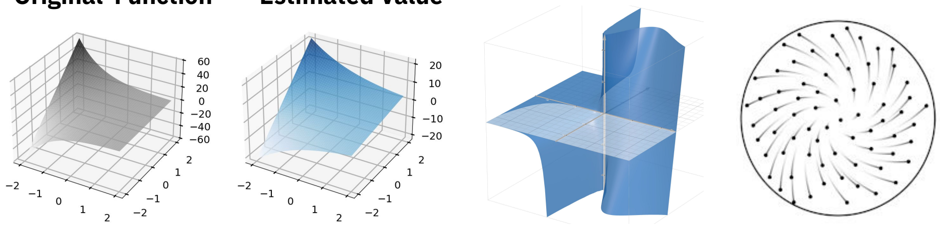
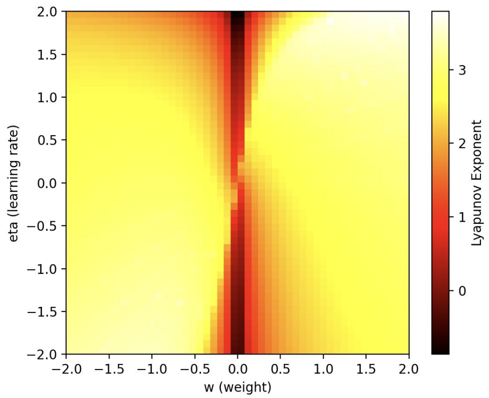


Figure 8-10. Q-Spike Approximation, w^* point distribution across parameter options, Map of a dissipative system (ie. STDP, Image Credit: wikipedia.com)



Lifting from the basic (1:1) to multilayer case, mapping Lyapunov exponents revealed strange attractors as well as stability at the locus of parameter options.

Figure 11. A Lyapunov exponent heatmap of a multilayer STDP network, quantifying divergence and demonstrating the "edge of chaos"

Hypothesis: Guiding Principle

The bifurcation parameter η defines whether the system converges, **learns efficiently at the edge of chaos**, or diverges. Optimal learning occurs in the narrow regime where **Lyapunov exponents are close to 0**.

Theorem 1: Quadratic Spike Approximation.

For any given LIF neuron with a membrane voltage V and postsynaptic potential ϵ, if given an input parameter λ, the membrane voltage can be generalized as:

$$f(x) = w_j I e^{-\frac{x}{\tau_m}}$$

where the spike timing difference is given by $\Delta t(w)$ with a Taylor expansions truncated at the N-th ordered term such that the N satisfies:

$$t' - t = x = -\tau_m \omega\left(-\frac{\theta}{w_j k \tau_m}\right) \qquad \min_{N \in \mathbb{Z}} (N+1)! \geq 8e\tau_m^2$$

Theorem 2: Bifurcational Classification.

A multilayer STDP-LIF network is an entirely dissipative system composed of unstable critical points w^*. Solving for local linearity in the neighborhood of points w^*,

$$w^* = \left(\theta \cdot \exp\left(\frac{k}{\tau_m}\right)\right)/k^2 \qquad f'(w^*) = \eta\left(\frac{n}{\tau_m w^* k\left(1 - \frac{k}{\tau_m}\right)}\right)$$

where in the triangular-block Jacobian, each represents the eigenvalues, thus making the system contain a stability-switch bifurcation (for η) within codimension-1. A candidate Lyapunov energy function modeling kinetic and potential interactions is provided as:

$$E(w, \eta) = \frac{1}{2}[\Delta t(w)]^2 + \frac{\lambda}{2}(f'(w))^2$$

FINDINGS: ENHANCING SNN LEARNING

My hypothesis was founded on the idea that the network can **switch between chaos and stability** in an optimal manner to accelerate learning. After months of trial and error, my solution came through **two breakthroughs**.

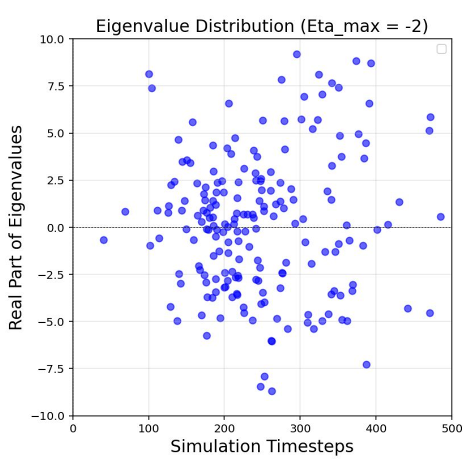
Breakthrough #1: Variable Plasticity

To ensure that the network can **selectively retain important information**, I designed a bio-plausible plasticity mechanism where learning rate decays naturally over time, increasing with frequent weight updates.

$$\tau_d \frac{\mathrm{d}\eta}{\mathrm{d}t} = -(\eta - \eta_{\mathrm{min}}) + \frac{1}{(\eta_{\mathrm{max}} - w(\Delta t))}$$

Coupled with **synaptic flipping** mechanism where synapses can switch between excitatory and inhibitory at $\eta = 0$ and $w = 0$.

Setting the η_{max} and η_{min} parameters allow me to **directly control the eigenvalues of the network**, setting the network mode. This was originally given by:



$$\lambda = \alpha = (\eta n k \exp(-\frac{k}{\tau_m}))/(\tau_m \theta(\frac{k}{\tau_m} - 1))$$

This creates a dynamic eigenvalue distribution across the network, allowing each synapse to convert its fixed points into saddles.

Figure 13. Randomly sampled synaptic eigenvalues from a three neuron network over 500 timesteps, provided gaussian noise input.

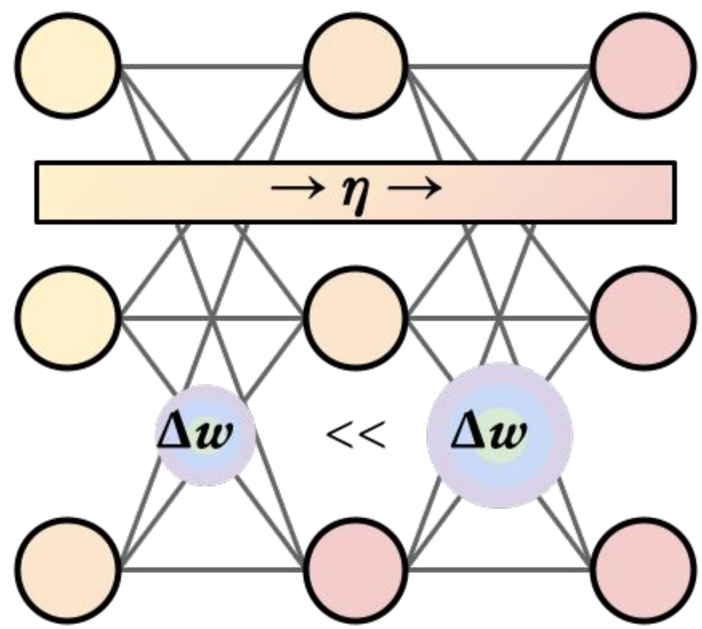


Figure 12. Plasticity is modulated by frequency of weight updates.

Algorithm Definitions

For variable plasticity, η represents the STDP learning rate, η_{min} and η_{max} are the minimum and maximum rate, respectively, and τ_d is the decay constant. For triangulated attribution, R represents the relevance score, p is the target synapse, i and j are intermediary neurons, and ε is an update constant.

FINDINGS: ENHANCING SNN LEARNING

Variable plasticity, however, created a **spiralling effect** as weight updates would lead to further updates through the medium of dynamic plasticity.

Breakthrough #2: Triangulated Attribution

Variable plasticity along with attractor/repeller dynamics leads to **runaway behavior and slow convergence**. I created an external weight attribution algorithm for the network to **accurately propagate a semi-global error signal.**

$$R_p^{i,j} = \eta_p \exp(-\frac{|w_{pi} - w_{pj}|}{\kappa})$$

$$\Delta w_{p^*} = \epsilon w(\Delta t) \qquad p^* = \underset{p \in \rho}{\operatorname{argmax}}\, R_p^{i,j}$$

This mechanism limits the spiralling effect of variable plasticity by **restricting the domain** for possible eigenvalues and network states.

Intuitively, this occurs as the weights settle down and "remove excess kinks" through triangulation, favoring minimal disruption.

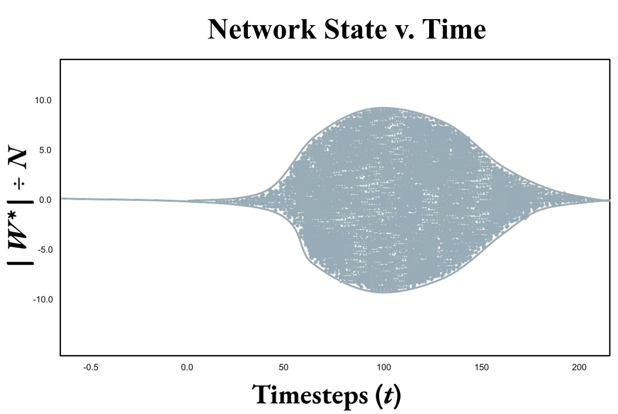


Figure 15. Constrained bifurcation plot of a 2 layer (3 neuron) simple MesoNet network provided varying gaussian noise input.

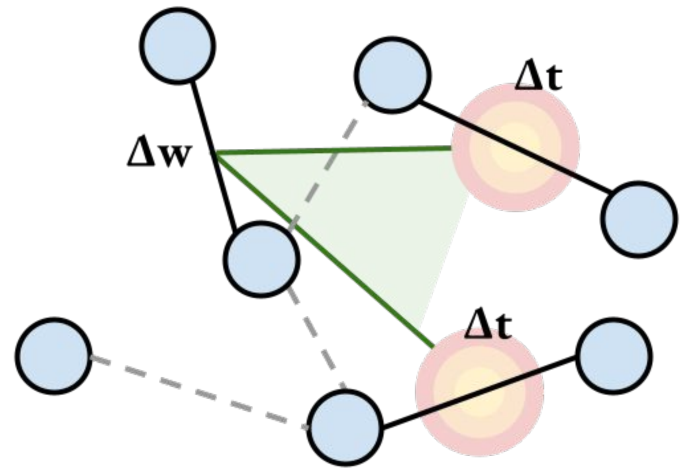


Figure 14. Various synapses triangulate a target synapse with high influence to spiking activity.

Inspiration for Variable Plasticity: *The longer your ideas are unopposed, the more stubborn you become of them.*

Inspiration for Triangulated Attribution: *Identifying what you are least confident about when learning a new skill.*

Iterative Development

Attempted to use SVD to Decouple Spike Timing Relationships	Tried decomposing global signals with Fourier Transform for attribution	Overlooked certain redundancies in regards to plasticity updates.	Faced compute limits while running heavy workloads on personal PC.

FINDINGS: ENHANCING SNN LEARNING

My Central Theorem

Considering a fully connected multilayer LIF-STDP network incorporating variable plasticity and triangulated attribution, the fixed points and the Jacobian of the locally linearized dynamics, respectively would be given by

$$\eta^* = \eta_{\min} + \frac{1}{\eta_{\max} - w^*} \qquad J_{\text{local}} = \begin{pmatrix} -\frac{\eta^*}{\tau_m}A + \epsilon g_w & 0 \\ -\frac{1}{\tau_d(\eta_{\max} - w^*)^2} & -\frac{1}{\tau_d} \end{pmatrix}$$

For large networks, the Jacobian spectrum follows a free convolution of semicircular laws centered at λ_1, given below, with a stable eigenvalue at λ_2, thus allowing the network to freely alter modes from a saddle-type bifurcation and a stable manifold.

$$\lambda_1 = -\frac{\eta^*}{\tau_m} \cdot \frac{\omega(-\frac{\theta}{wk\tau_m})}{\tau_m w^2 k(1 + \omega(-\frac{\theta}{wk\tau_m}))} + \epsilon g_w$$

The instability is dominated by $1/w^2$ which shrinks as w approaches w^*, developing stability.

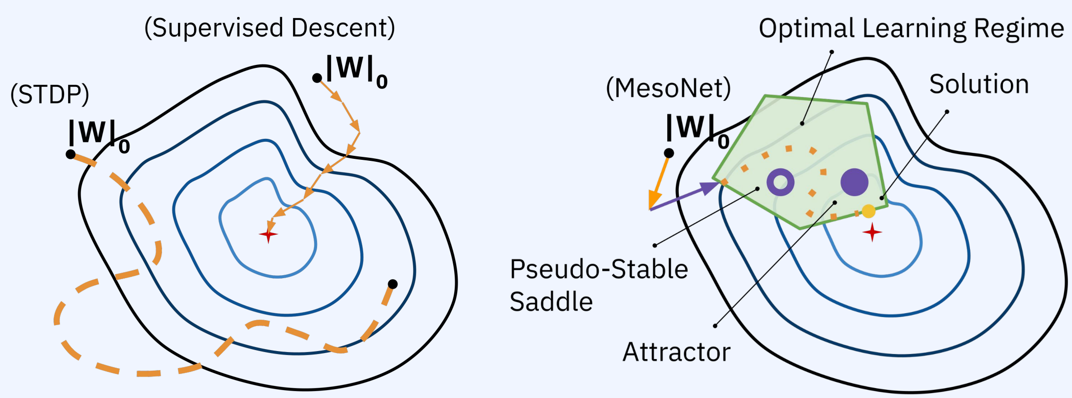


Figure 16. Saddle points and attractors are dynamically created and annihilated to "lure" weights in the optimal direction without awareness of the gradient itself.

Analyzing the Jacobian spectrum across a circular initial distribution, we can prove that saddles **condense to an optimal region**, near the desired solution.

This allows the network to **scan across the entire weight space and local minimas** before determining the optimal fixed point.

Emergent Properties

<u>Synaptic Interactions:</u> I generated a smoothed Turing map of inhibitory/excitatory interactions, revealing that MesoNet forms **cortical-like neuron organization** unlike standard STDP.

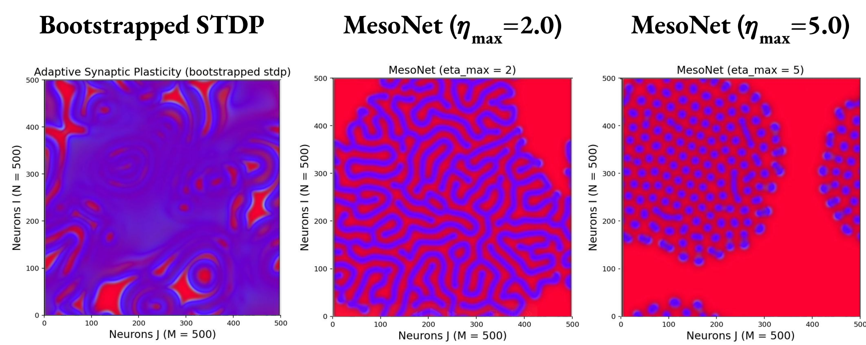


Figure 17. Turing map of inhibitory (red) /excitatory (blue) interactions (STDP on the left, MesoNet being the right two)

Demonstrates the fascinating ability for variable plasticity to **self-organize into clusters of related function**, similar to the brain.

FINDINGS: PERFORMANCE VALIDATION

Implementation

Split-&-Merge modules divide input images into multiple pathways for processing by different convolutional filters. Within them, I designed **neural cores to develop diverse representations**, exploiting MesoNet's self-organization.

(1) Improved multi-scale feature extraction
(2) Enhances network parallelization

Simulated with...

Image Credit from respective logos.

Triangulated Approximation was optimized through use of **Q-Spike Approximation** (Theorem 1) and **Dijkstra's Algorithm**.

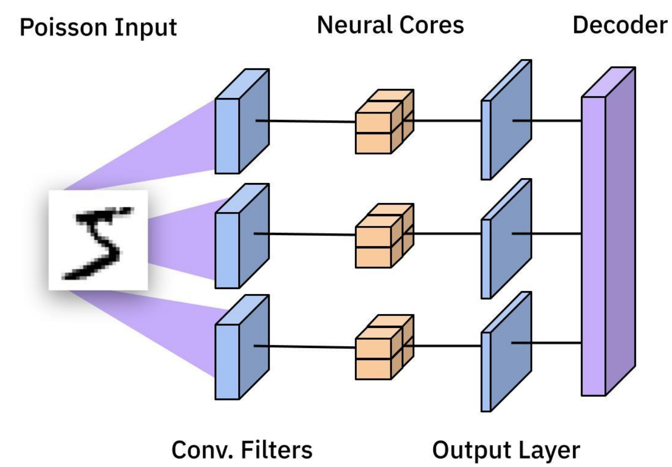


Figure 18. A simplified diagram of the MesoNet architecture.

Testing Procedure

To accurately validate MesoNet's performance, I conducted a rigorous suite of tests, evaluating **accuracy, speed, and energy efficiency** across various tasks.

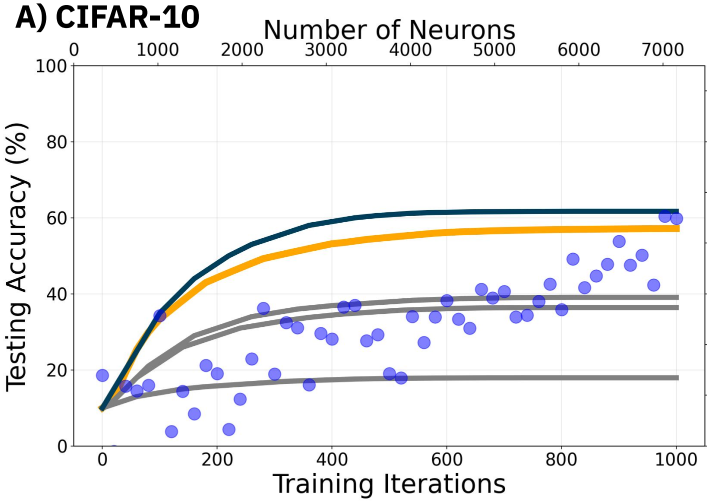
I compared MesoNet to three of the most popular SNN models: CSNN (SLAYER), Adaptive Synaptic Plasticity, Basic. Additionally, I compared MesoNet to the current state-of-the-art ANN clustering model (IIC).

Testing Hyperparameters

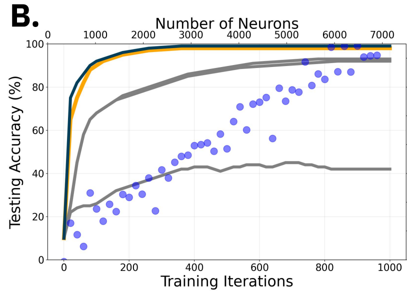
Parameter	MNIST	CIFAR10	CIFAR100	F-MNIST	E-MNIST
Encoding Length	300	350	450	300	300
Learning Rate	0.01	0.005	0.001	0.01	0.01
Membrane Time Constant	2 ms	2 ms	2 ms	2 ms	2 ms
Synapse Time Constant	1 ms	1 ms	1 ms	1 ms	1 ms

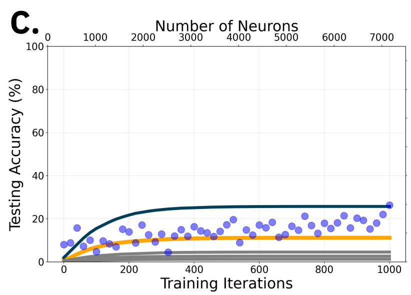
Figure 19. To ensure testing validity, all hyperparameter options were standardized for testing, and were all ran within the split-and-merge architecture.

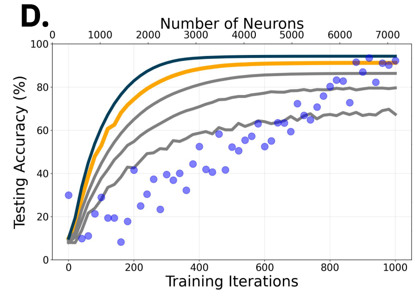
FINDINGS: PERFORMANCE VALIDATION

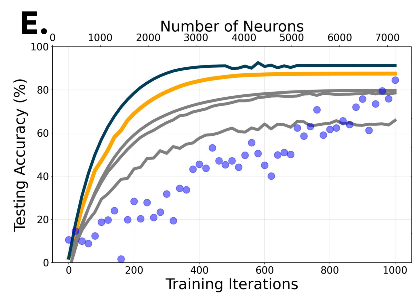


Figures A-E. Accuracy Curve across datasets (Scatter plot demonstrating minimum required neurons)







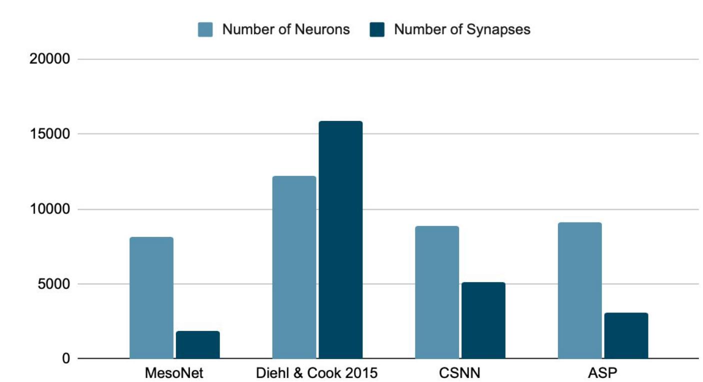


Robustness was tested by removing synapses randomly until a 5% reduction in accuracy was observed.

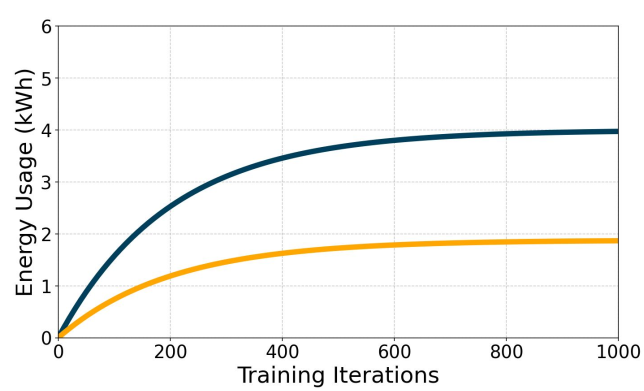
SOTA Contenders: Convolution SNN (SLAYER), Adaptive Synaptic Plasticity, Simple, **Invariant Information Clustering (IIC)**, **MesoNet**

- Interestingly, the minimum required neurons to achieve accuracy (testing for robustness), **is linearly correlated with accuracy, yet largely random**, indicating a level of synaptic independence in its activity, encouraging future research into why this occurs.
- MesoNet specifically demonstrated **improved performance on larger tasks** (ie. CIFAR-10, F-MNIST), with trivial differences for smaller ones (ie. MNIST).

G. Min. Number of Components (MNIST)



F. Energy v. Train Iterations (MNIST)



MesoNet surpasses CSNN, achieving a **56.6% accuracy on CIFAR-10**, 18.57% points above the previous SOTA model.

Figures F-G. MesoNet required **37% of CSNN's synapses.** And operated with **47.23% the energy of IIC**, demonstrating phenomenal resource efficiency.

CONCLUSIONS: DISCUSSION & APPLICATIONS

5.1%
within range of the SOTA ANN model

53%
less energy than the SOTA ANN (IIC)

Enhanced SNN performance on practical tasks

- Significant improvement in multilayer SNN for deeper tasks (i.e. CIFAR-10)
- Comparable convergence time to SOTA ANNs (ie. ICC)

Novel understanding of SNN dynamical systems

- Supported the existence of an optimal learning regime
- Defined full bifurcation dynamics underlying STDP SNN learning

Image Credit: wikipedia.org

Medical Implants. MesoNet can be utilized in implants like pacemakers to assist patients for *over 15 years*.

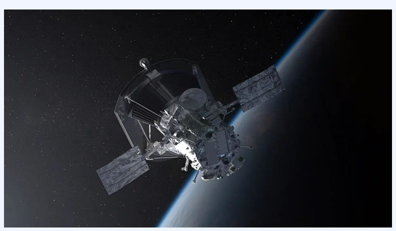


Image Credit: nasa.gov

Deep Space Missions. MesoNet can reduce energy consumption on the Parker Solar Probe by *around 23%*, massively extending its operating lifespan.

Image Credit: thenextsummit.com

Remote Sensing. MesoNet can enhance the energy efficiency of battery-powered wildfire sensing cameras by nearly *40%*.

Limitations

- Evaluation was restricted to computer vision datasets
- Study only considers Poisson Rate Encoding techniques

Future Directions

- Stricter restrictions regarding the “optimal regime” (dynamical isometry?)
- Exploration into efficient geometries for self-organizing structures.

SCOPE OF WORK

New Work by Author

- **Conception**– I had thought of addressing STDP learning individually.
- **Execution**– I was inspired to take on this project from the lens of Dynamical Systems theory by (Zhang. et. al.) who utilized similar methods for their analysis of LIF neurons. I formed the idea of Quadratic Spike Approximation and applied it to bifurcation analysis. I adapted insights from Pennington, et al. describing DNN learning to describe an optimal regime of different characteristics for SNNs. I developed variable plasticity and triangulated attribution and proved its efficiency, forming all argumentations individually. I implemented the algorithms and conducted testing.
- **Presentation**– I have created the proofs, videos, and presentation for this project. I would like to thank Ms. Julia Wendel for providing feedback on my project and display for the regional fair.

Professional, Institutional, and Academic Resources and Support

I researched my project individually. I had access to GTX 1080 Ti graphics card for testing my system as well as access to certain papers from the San Jose State University Public Library, where I conducted some of my literature review.

REFERENCES / SUPPLEMENTAL INFORMATION

Author's Note

All theorems, images, and figures were created by Krishna Bhatt, unless otherwise noted.

Key References

Zhang, S.-Q., Zhao-Yu Zhange, & Zhou, Z.-H. (2021). Bifurcation spiking neural network. The Journal of Machine Learning Research, 22(1), 11459–11479. https://doi.org/10.5555/3546258.3546511

Deng, Lei, et al. "Rethinking the Performance Comparison Between SNNS and ANNS." Neural Networks, vol. 121, Sept. 2019, pp. 294–307. https://doi.org/10.1016/j.neunet.2019.09.005.

Meng, Mingyuan, et al. "High-parallelism Inception-like Spiking Neural Networks for Unsupervised Feature Learning." Neurocomputing, vol. 441, Mar. 2021, pp. 92–104. https://doi.org/10.1016/j.neucom.2021.02.027.

Pennington, Jeffrey, et al. "Resurrecting the Sigmoid in Deep Learning Through Dynamical Isometry: Theory and Practice." In Proceedings of the 31st International Conference on Neural Information Processing Systems (NIPS'17), 2017, pp. 4788–98. ACM Digital Library, https://doi.org/10.5555/3295222.3295232.

Saxe, Andrew, et al. "Exact Solutions to the Nonlinear Dynamics of Learning in Deep Linear Neural Networks." In Proceedings of the 2nd International Conference on Learning Representations (ICLR), 2014, https://doi.org/10.48550/arXiv.1312.6120.

S. H. Strogatz. Nonlinear Dynamics and Chaos. CRC Press, 2018.

Supplemental Information

Video Overview: https://youtu.be/4yPmUsaXKY0, **Turing Patterns**: https://youtu.be/vj3dSUOi5Mo, **Project Journal**: https://docs.google.com/document/d/1e2v2yKZbb1bnuixkM7dCZfcH4pTlicknByTlDitCO1c/view, **Codebase**: https://github.com/blizzard-labs/NMCtests