SAND2022-15624C

Sandia National Laboratories

Varied Delays in Spiking Neural Networks Support Learning with Conjunctive Temporal Features

Presented by

Felix Wang, Corinne Teeter

11/9/2022

SNUFA 2022

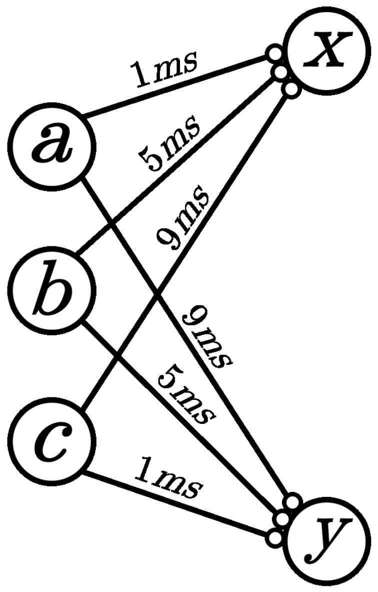
U.S. DEPARTMENT OF ENERGY

NNSA National Nuclear Security Administration

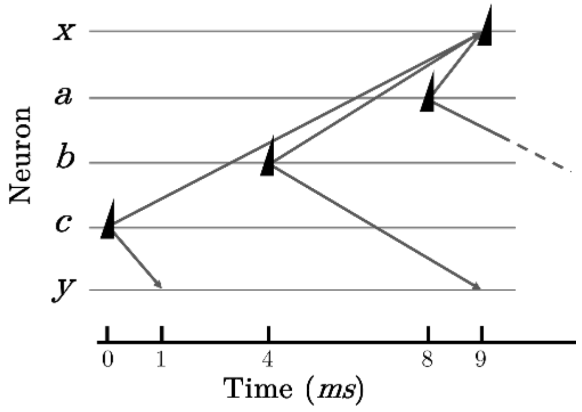
Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

The importance of spike timing

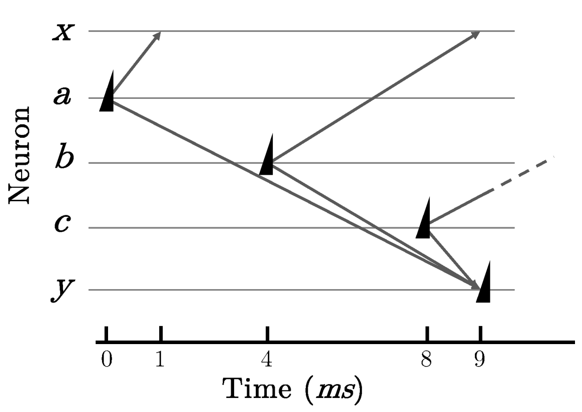
Structure in network connectivity drives spike-timing patterns



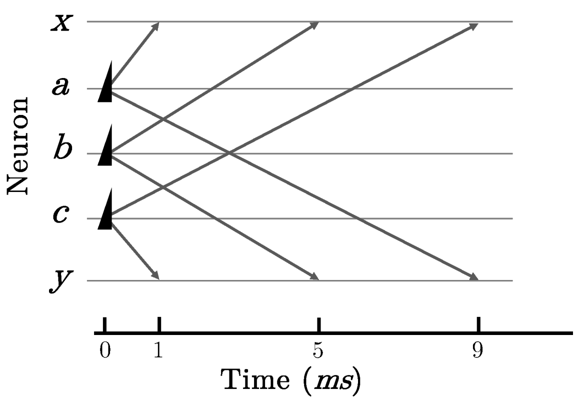
Simple network with timing-dependent connectivity



Optimal activation of neuron x



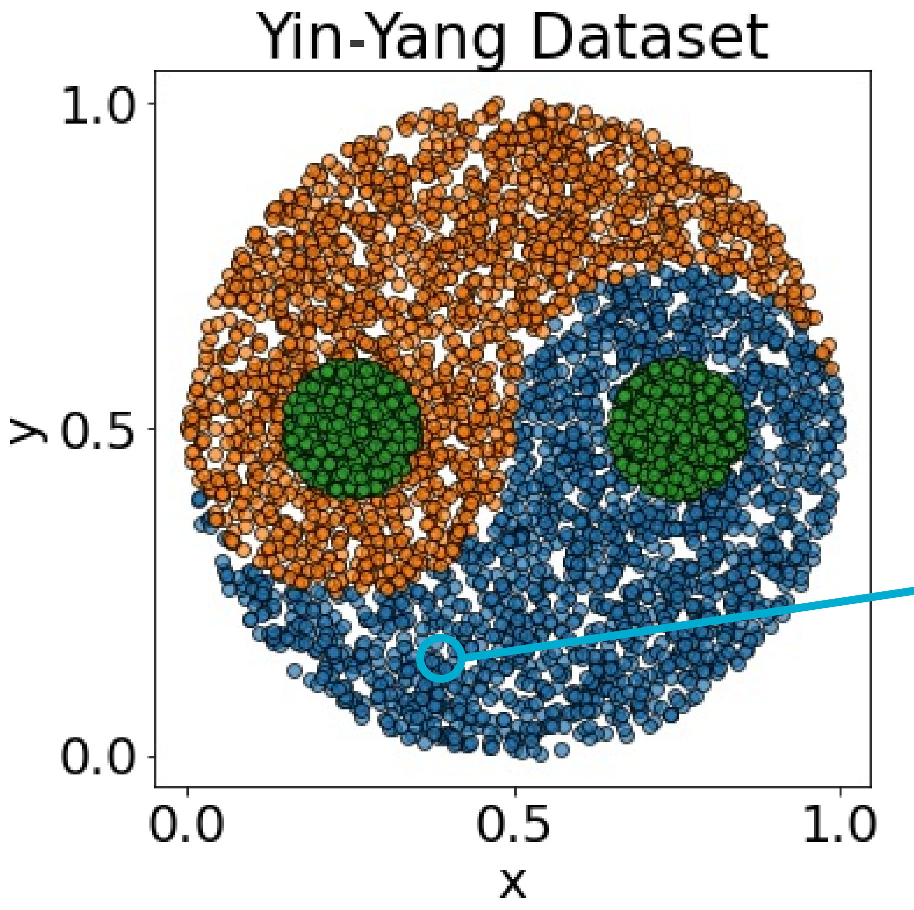
Optimal activation of neuron y



Suboptimal activation of neurons x, y

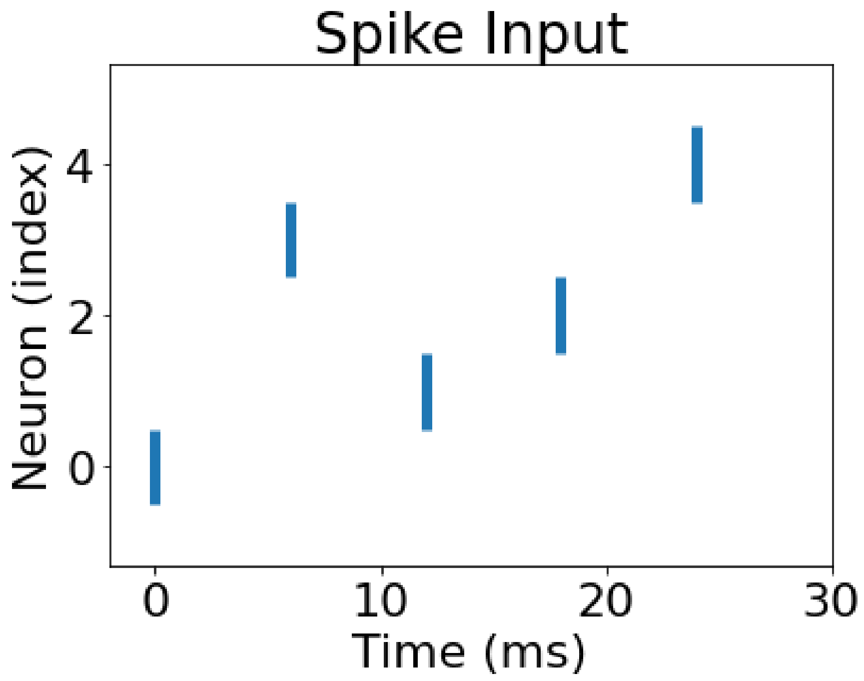
Yin-Yang Dataset

Spatio-temporally encoded, 3-class dataset developed for research on biologically plausible error backpropagation



Spike Encoding:

- Bias Neuron : 0
- x-coordinate : $x\, t_{max}\,, (1-x)\, t_{max}$
- y-coordinate : $y\, t_{max}\,, (1-y)\, t_{max}$



EventProp for Training SNNs

Derives an exact gradient across event-based neuron models using the adjoint method and partial-derivative jumps

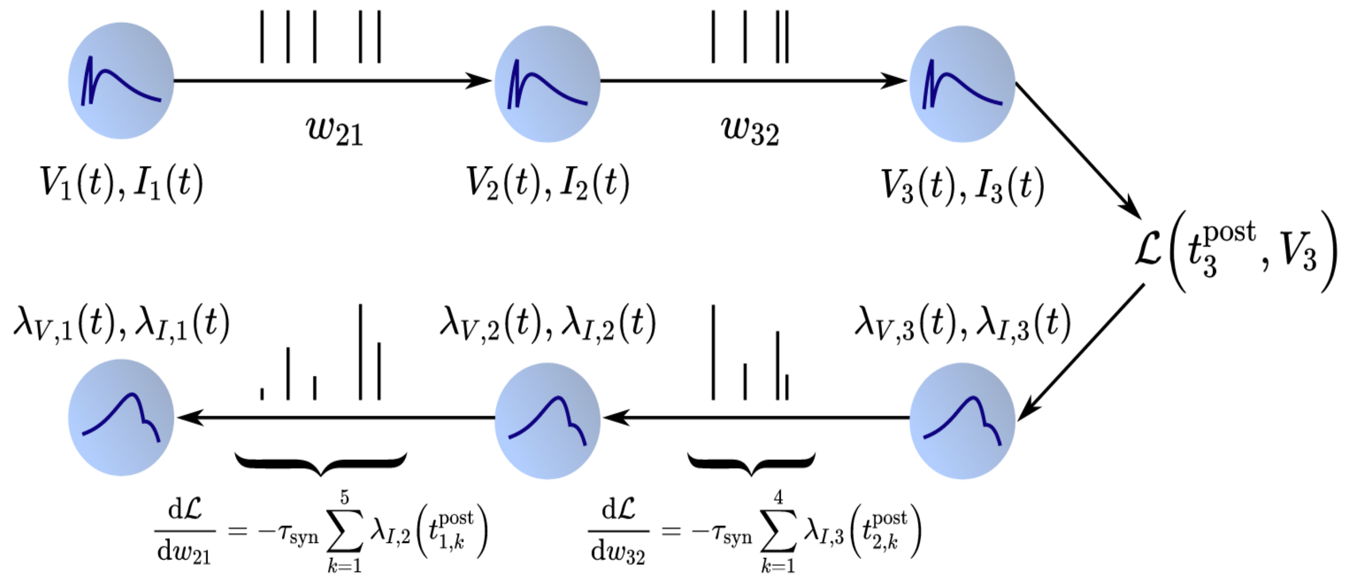
For spike-based loss (e.g. time-to-spike), this is computed as:

$$\tau_{mem}\lambda_V' = -\lambda_V - \frac{\partial l_V}{\partial V}$$
$$\tau_{syn}\lambda_I' = -\lambda_I + \lambda_V$$

Transitions (at spikes t_k^{post}):

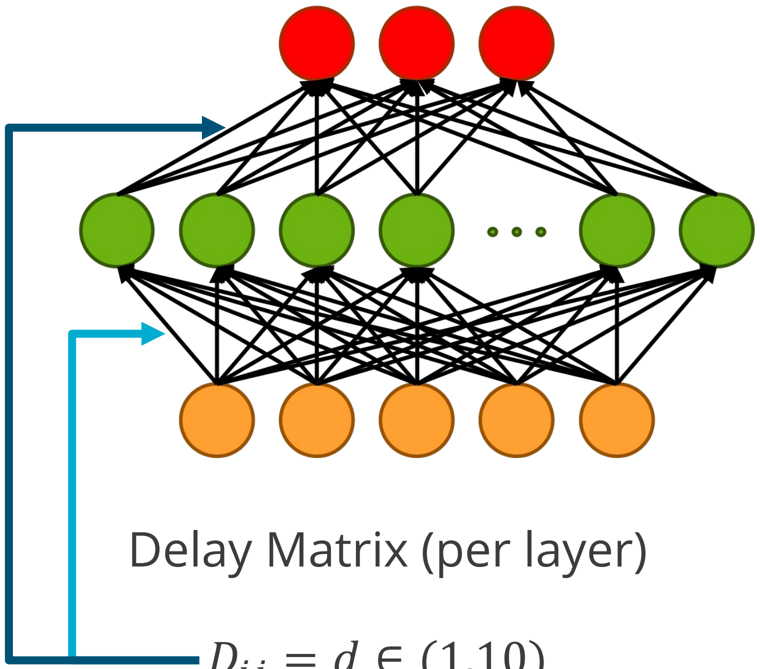
$$(\lambda_V^-)_{n(k)} = (\lambda_V^+)_{n(k)} + \frac{1}{\tau_{mem}\left(\dot{V}^-\right)_{n(k)}}\left[v(\lambda_V^+)_{n(k)} + \left(W^T(\lambda_V^+ - \lambda_I)\right)_{n(k)} + \frac{\partial l_p}{\partial t_k^{post}}\right]$$

EventProp: Backpropagation for Leaky Integrate-and-Fire Neurons



Simple Feedforward Network with Delays

We implement a delay matrix for indexing during the forward and backward passes



Delay Matrix (per layer)

$D_{ij} = d \in (1,10)$

As contrasted with typical DL-based SNN formulations which have uniform, single-time-step delays between layers

Forward and backward temporal indices

$$T_{f,ij} = \min(0, t - D_{ij})$$

$$T_{b,ij} = \max(T\_\max, t + D_{ij})$$

Updates to the forward time-evolution:

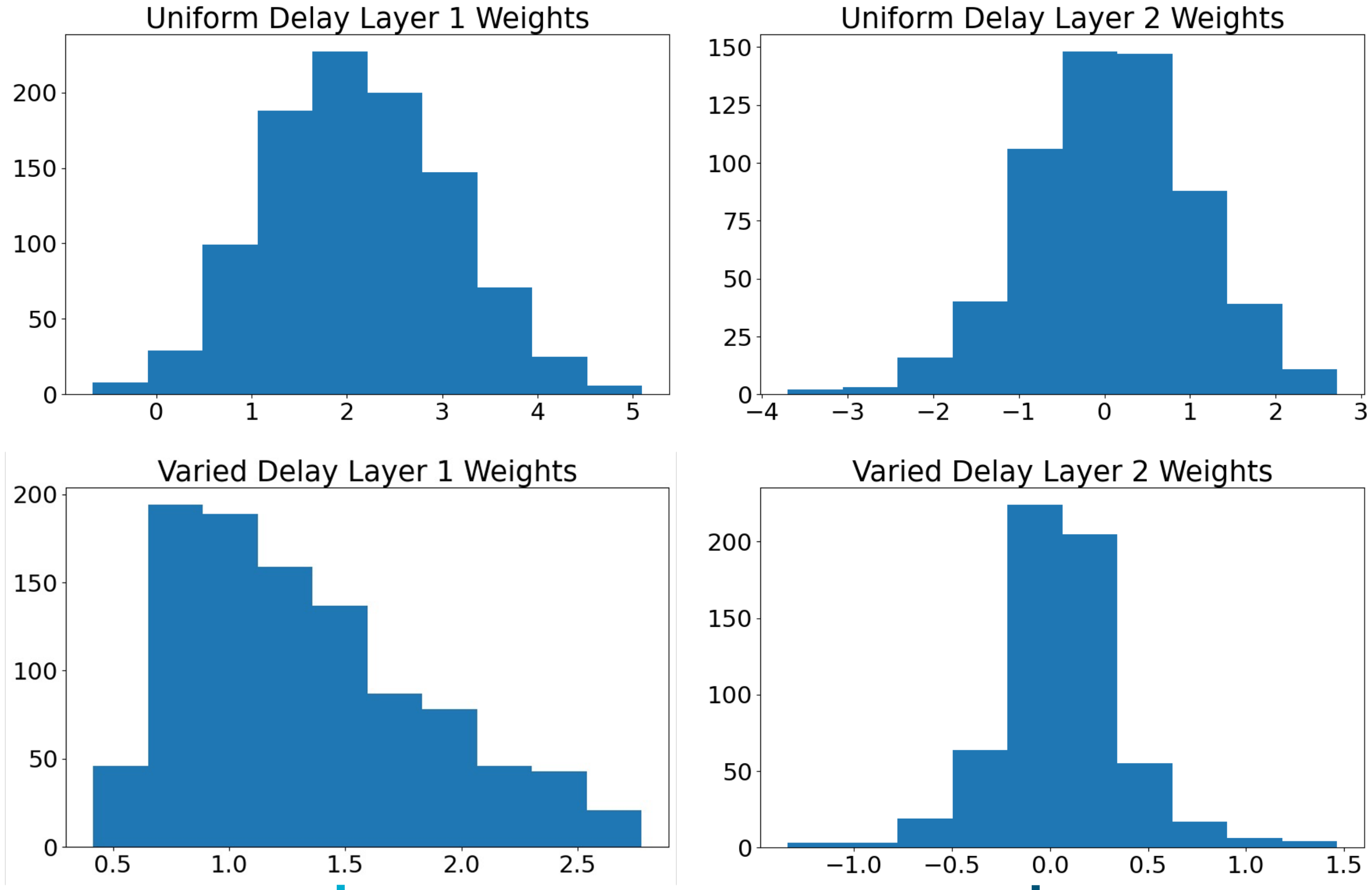
$$I_j^+ = I_j^- + W_j e_{n,T_{f,j}}$$

Updates to the backward gradient jump:

$$\left(\lambda_{V\,i}^{-}\right)_{n(k)} = \cdots \left(W_i^T \left(\lambda_{V,T_{b,i}}^{+} - \lambda_{I,T_{b,i}}\right)\right)_{n(k)} \cdots$$

SNN Training Results

Layer weights are distributed qualitatively differently

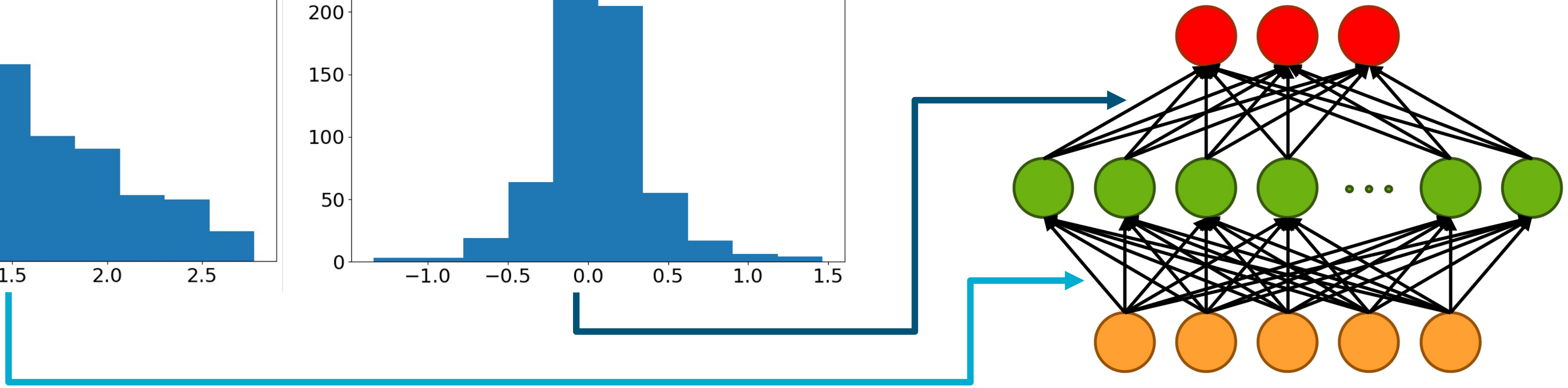


We sampled the Yin-Yang dataset:

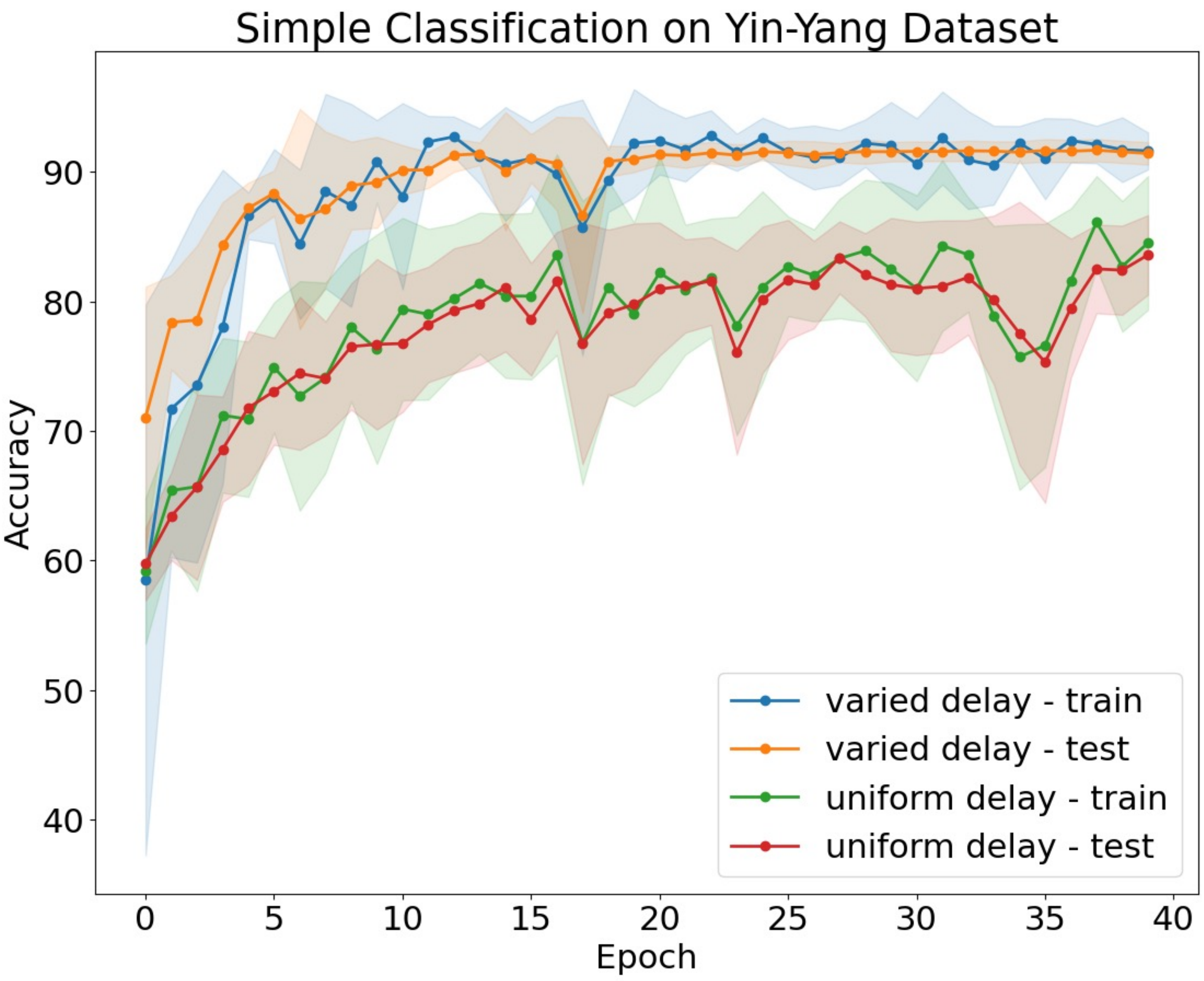
- 5000 times for training
- 1000 times for testing

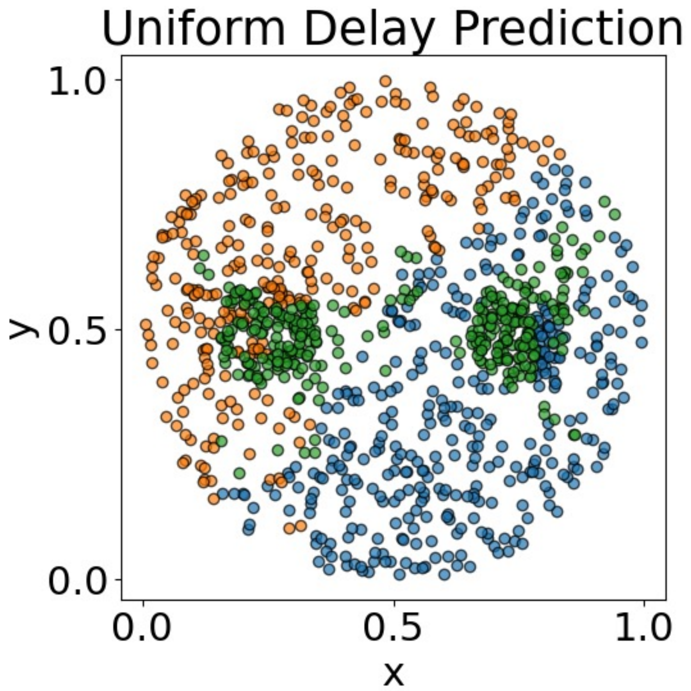
We used stochastic gradient descent (SGD) optimization with a simple learning rate scheduler

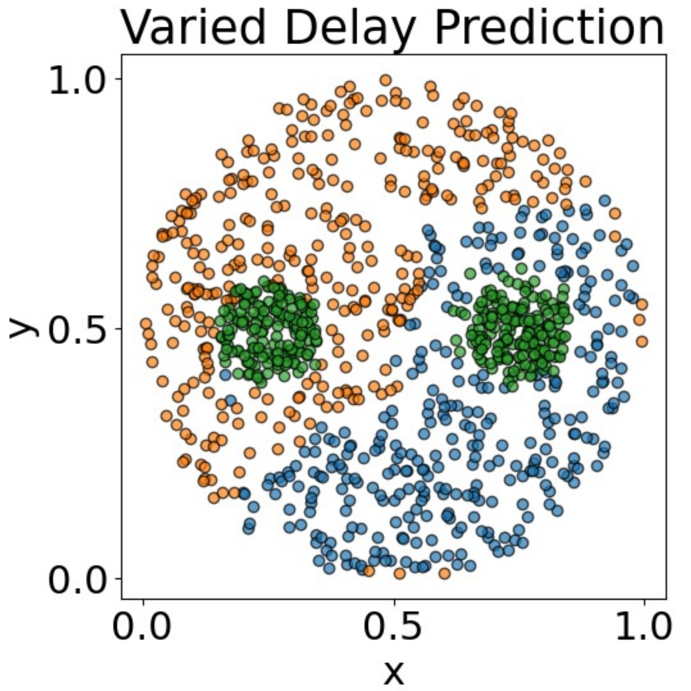
We trained for 40 epochs with shuffled mini-baches of 100



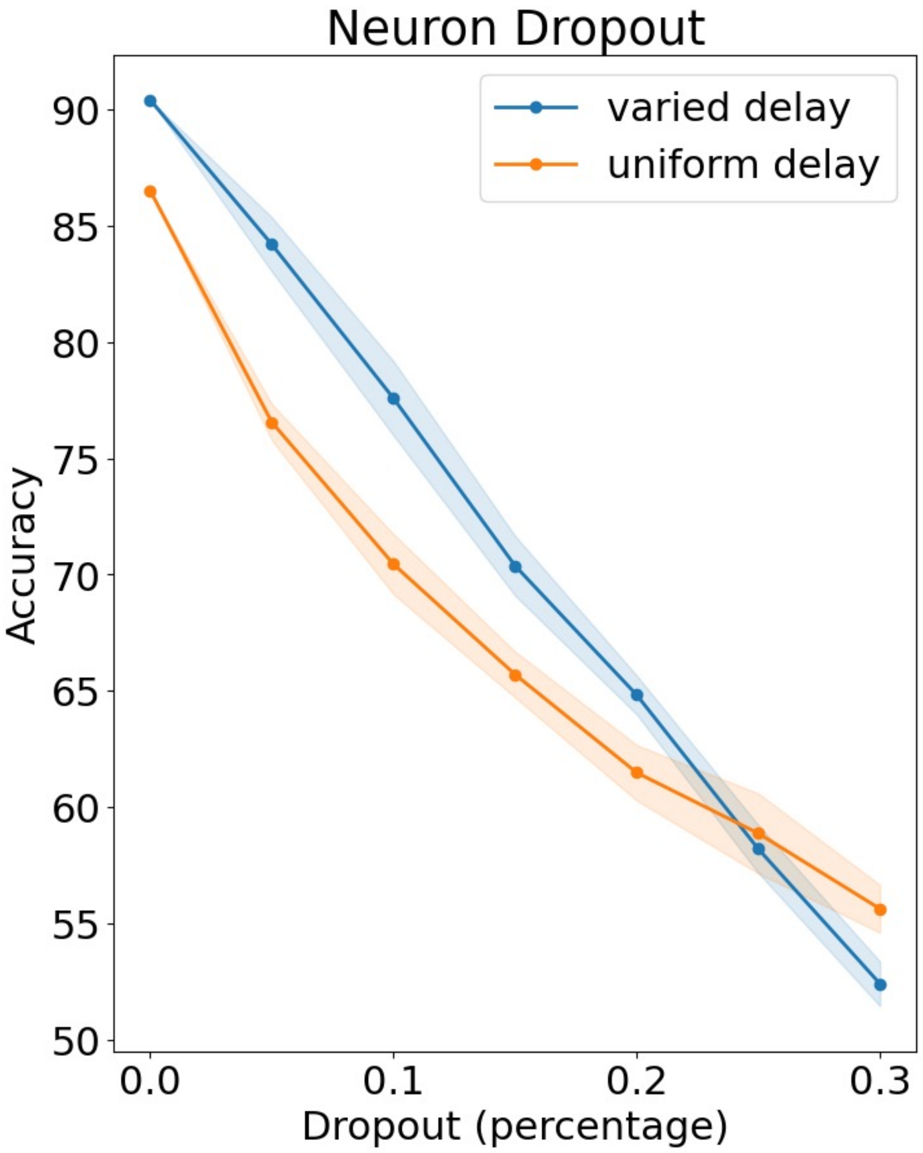
SNN Classification Results



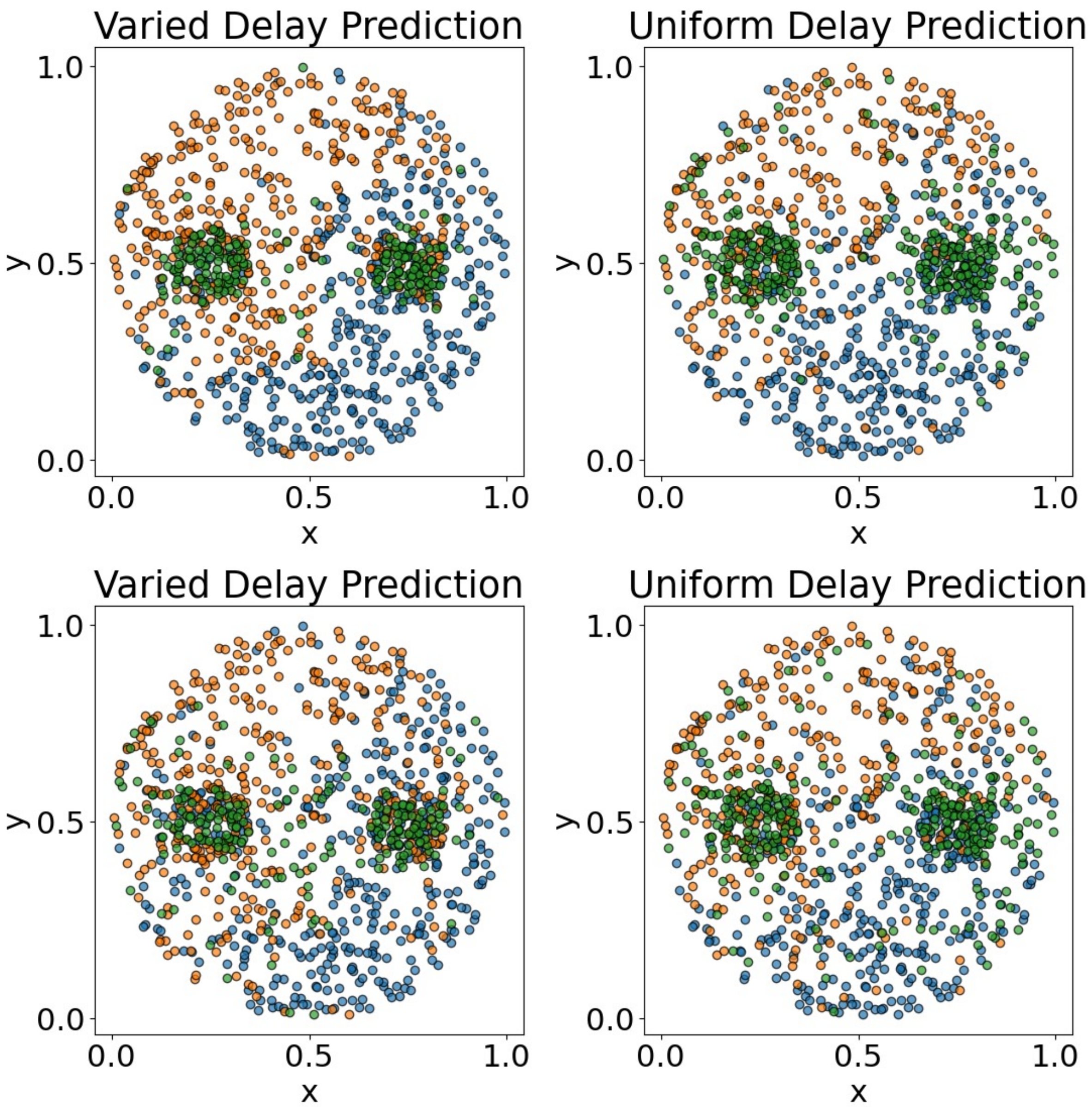




Spike Dropout in Inference



Classification plots (0.15 top, 0.25 bottom)



Summary and Takeaways

- Varied connection delays are a notable architectural difference between more biological SNNs (e.g. reservoirs) compared to their artificial counterparts (e.g. layers)
  - Learning rules such as STDP may be taking advantage of varied delays at a network level
  - Computationally, varied delays result in differentially evaluated upstream spiking activity
- We extend ANN/DL-based methods (EventProp) to compute gradients through varied delays by introducing a delay matrix and forward/backward temporal indices
  - We perform experiments on a simple temporal classification task (Yin-Yang dataset)
  - We found that delays result in improvements in training (e.g. faster learning, robustness)
- Future work will be required in exploring the hyperparameter and architecture space
  - What determines the relationship between connection delays and the input space?
  - Can we start to merge learning rules? (e.g. STDP + STP + BackProp)
  - Are there additional mechanisms to support training? (e.g. background spiking activity)
- Current tools are typically for either biological SNNs or DL/ANNs, but not both
  - Implementation of connection delays is suboptimal with tools like PyTorch

References

1. Wunderlich, T. C. and Pehle, C. “Event-based backpropagation can compute exact gradients for spiking neural networks”, *Scientific Reports,* vol. 11, pp. 12829, (2021)
2. Kriener, L. Göltz, J. and Petrovici, M. A. “The Yin-Yang dataset”, *arXiv CoRR,* abs/2102.08211, (2021)
3. Izhikevich, E. “Polychronization: computation with spikes”, *Neural Computation*, vol. 18, pp. 245-282, (2006)