Spiking Neural Networks (SNNs) with Improved Inherent Recurrence Dynamics for Sequential Learning

Wachirawit Ponghiran and Kaushik Roy, Purdue University, wponghir@purdue.edu

Motivations

- SNNs have an inherent recurrence/internal states like RNNs
  → RNNs alternative for low-power sequential learning applications

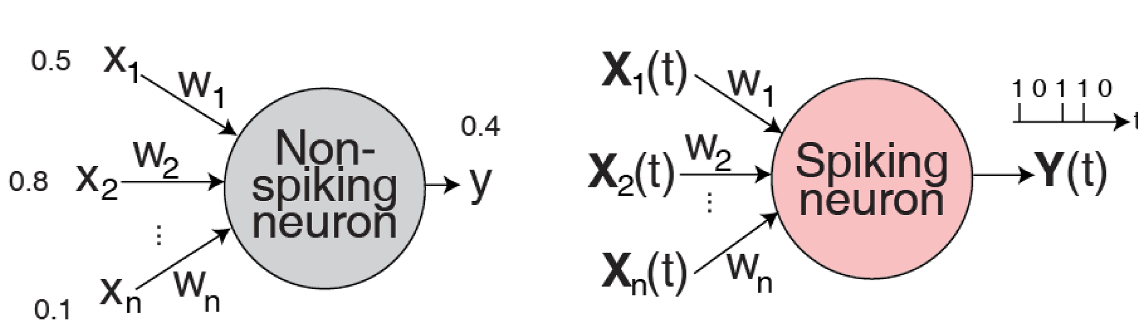
Representative works do not demonstrate **the usefulness of the inherent recurrence**
(Diehl et al. 2015; Rueckauer et al. 2017; Sengupta et al. 2019)

- Cramer et al. (2020) has successfully trained SNN for classifying digits from spoken words

However, apart from this, SNNs have not been applied to sequential learning applications due to **the difficulty in training**

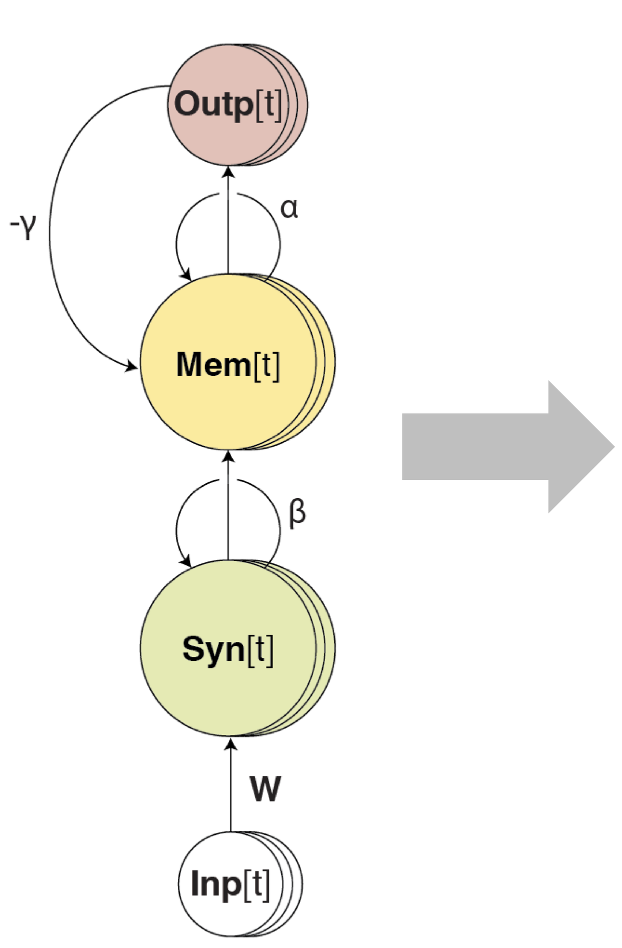
SNNs Dynamics and Vanishing Gradient Problem

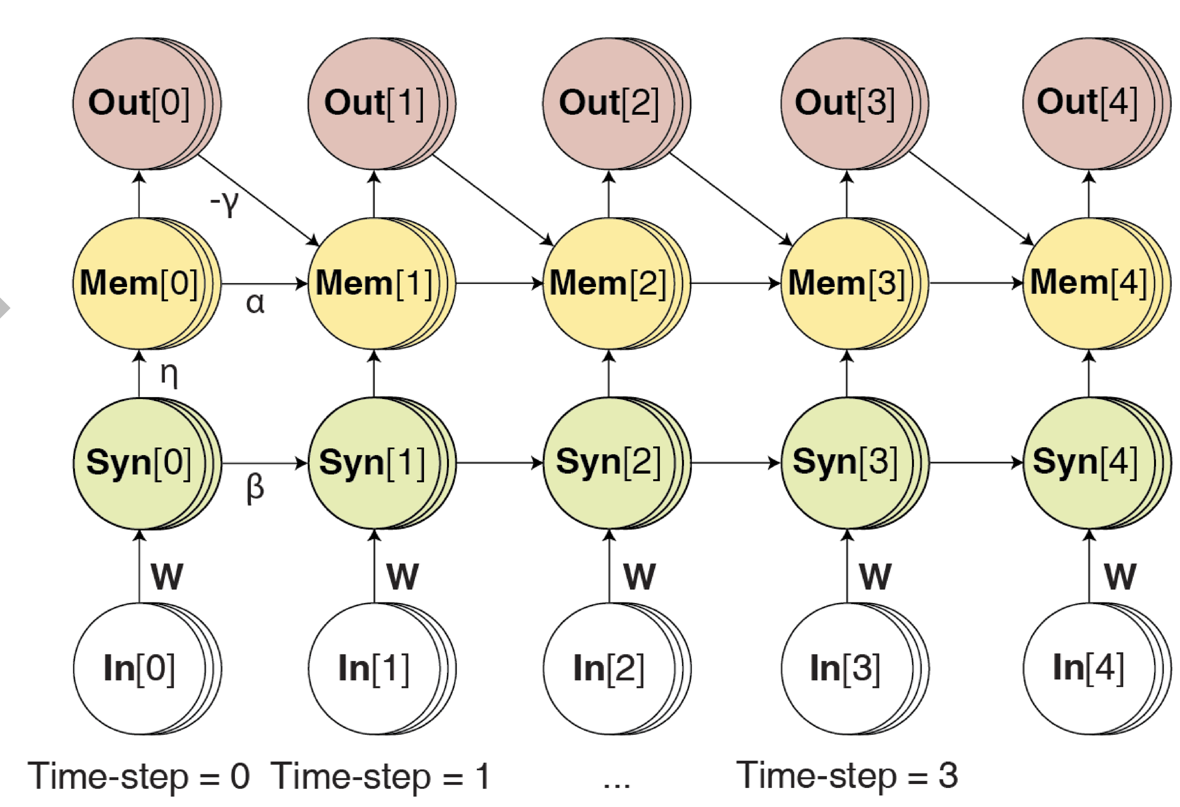
- Spiking neurons communicate asynchronously with {0, 1} to mimic spiking activity → Potential energy/power saving on an event-driven hardware



- Neurons have two internal states: synaptic current ***Syn*** and membrane potential ***Mem***
- 1s are generated whenever ***Mem*** exceeds a threshold ***γ***

Unrolling the computation graph reveals **the vanishing gradient problem**





Modifications To Improve The SNNs Dynamics

- Selectively update the ***Syn*** to avoid forceful decay of its value

Vanilla SNNs

$$\mathbf{Syn}[n] = \beta\ \mathbf{Syn}[n-1] + \mathbf{W}\ \mathbf{Inp}[t]$$

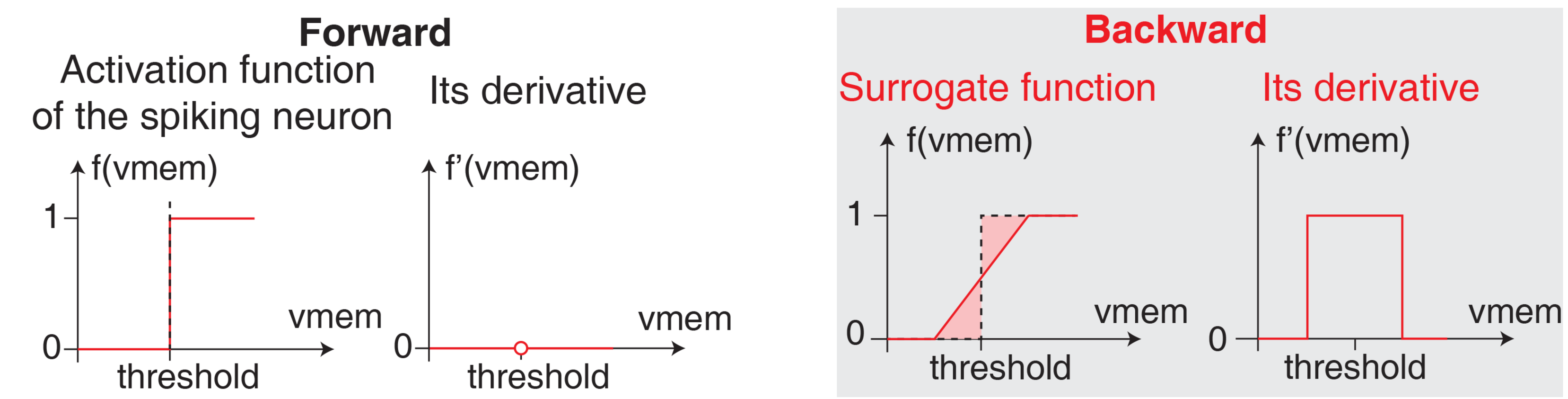
Proposed SNNs with the improved dynamics

$$\mathbf{Forget}[n] = \sigma(\mathbf{W}_{fi}\ \mathbf{Inp}[n] + \mathbf{W}_{fo}\ \mathbf{Outp}[n-1])$$
$$\boldsymbol{Cand}[n] = \mathrm{ReLU}(\mathbf{W}_{ci}\ \mathbf{Inp}[n] + \mathbf{W}_{co}\ \mathbf{Outp}[n-1])$$
$$\mathbf{Syn}[n] = \mathbf{Forget}[n] \cdot \mathbf{Syn}[n-1] + (1 - \mathbf{Forget}[n]) \cdot \mathbf{Cand}[n]$$

- ***Forget*** and ***Cand*** indicate the amount to be removed and added to the synaptic currents ***Syn***

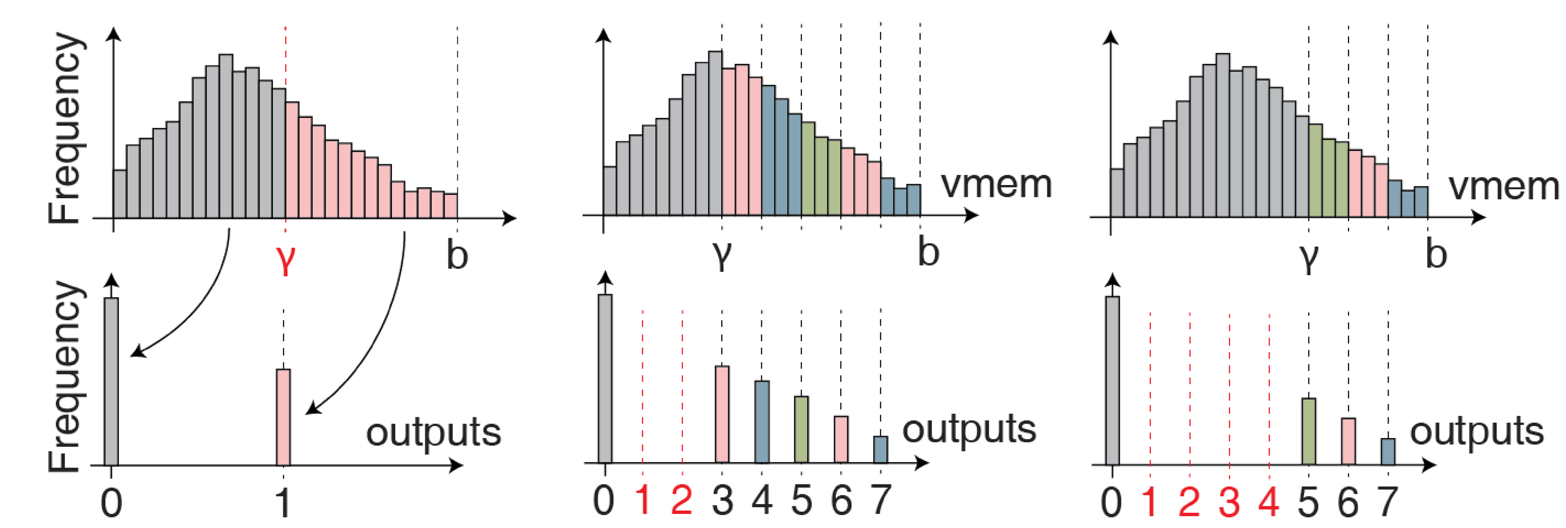
Gradient Mismatch and The Proposed Training

- Use of surrogate gradient leads to noisy gradient updates
  → Affect the model's learning performance



- Mitigate the problem by increasing output precision and making neurons output multi-level values

Generalize spiking output generation to multi-level output cases



Experimental Results

- Comparison between **the proposed SNNs**, **vanilla SNNs**, LSTMs, GRUs on TIMIT and LibriSpeech speech recognition tasks

Results on TIMIT dataset

Architecture	Prediction accuracy (%)	# of zero outputs (%)	Avg ops/inf (normalized)
LSTMs	82.68	<0.01	1
GRUs	82.26	<0.01	0.81
Vanilla SNNs	70.66	59.90	0.13
Proposed SNNs	81.28	84.29	0.08

Results on LibriSpeech dataset

Architecture	Prediction accuracy (%)	# of zero outputs (%)	Avg ops/inf (normalized)
LSTMs	89.95	<0.01	1
GRUs	89.77	<0.01	0.83
Vanilla SNNs	78.39	59.96	0.16
Proposed SNNs	88.25	86.33	0.08

- The proposed SNNs provide **2x reduction in the number of trainable parameters** over LSTMs while achieving comparable speech recognition accuracies
- The proposed SNNs lead to **>10x reduction in the number of MultOps** over GRUs due to their sparse communications

Acknowledgement

This work is supported in part by C-BRIC, one of six centers in JUMP, a Semiconductor Research Corporation (SRC) program sponsored by DARPA, IARPA Micro4AI program, DARPA AIE, NSF, and DoD

C-BRIC Center for Brain Inspired Computing
SRC® Semiconductor Research Corporation
IARPA
DARPA
NSF
U.S. DEPARTMENT OF ENERGY