

# Eligibility propagation in multi-layer recurrent spiking neural networks.

## Presentation 10: Evaluation meeting 2

Werner van der Veen  
(w.k.van.der.veen.2@student.rug.nl)

March 2, 2021

# Introduction – context

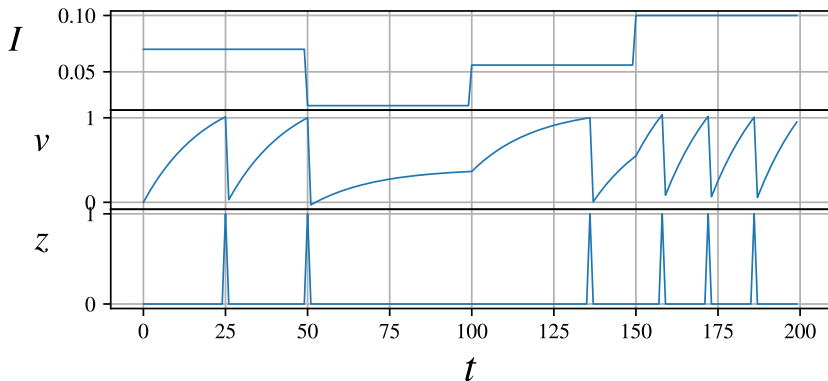
- Biological inspiration to AI
- ANNs and backpropagation
- Deep learning

# Introduction – relevance

- Deep learning is energy intensive
- Fundamental differences between DL and biological learning
  - Continuous vs. binary communication
  - Backpropagation/BPTT vs. learning signals

# Introduction – SNNs

- Binary spikes  $\rightarrow$  no backpropagation
- Good performance, but not competitive with DL



# Introduction – neuromorphic computing

- Physical embedding of neural network in an analog medium
- Colocalized memory and computation
  - no backpropagation
- SNNs as ideal biologically inspired NC paradigm for efficient computation
- Analog SNNs are extremely fast and energy efficient
- Problem: learning rule must be *local* and *online*
  - revalue biological learning

# Introduction – biological learning

- Hebbian learning
  - runaway excitation
- STDP
  - how to teach?
- Learning signals: three-factor Hebbian learning, R-STDP
- Biological learning signals: neurotransmitters
  - credit assignment
- Eligibility traces

# Introduction – eligibility propagation

- Mathematical approximation to BPTT
- Local learning signals and eligibility traces
- Applicable to any SNN topology and multiple neuron models
- Also applicable to neuromorphic VLSIs
- Competitive with LSTMs on phone classification

# Introduction – eligibility propagation

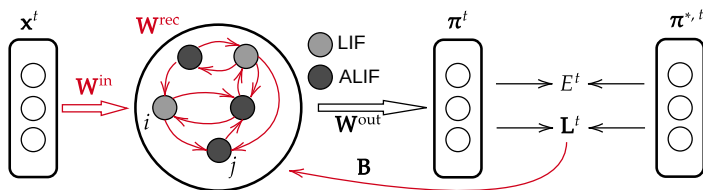
- E-prop: only ALIF so far
- New research: STDP-LIF and Izhikevich neuron display STDP



# Introduction – this research

- Reproduce results of original e-prop paper  
→ explicitly instead of automatic differentiation
- Extend STDP-LIF to STDP-ALIF
- Experimentally verify the STDP-ALIF and Izhikevich neuron on phone classification task
- Examine effects of e-prop in multi-layer recurrent SNN

# Methods – technical framework (e-prop)

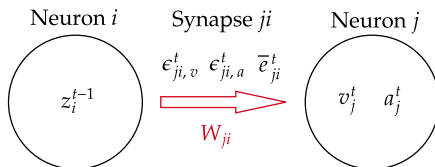


E-prop model  $\mathcal{M} = \langle M, f \rangle$

$$\mathbf{h}_j^t = M(\mathbf{h}_j^{t-1}, \mathbf{z}^{t-1}, \mathbf{x}^t, \mathbf{W}_j) \quad (1)$$

$$\mathbf{z}_j^t = f(\mathbf{h}_j^t) \quad (2)$$

# Methods – the ALIF neuron model



# Methods – the ALIF neuron model

$$z_j^t = H(v_j^t - v_{\text{th}} - \beta a_j^t) \quad (3)$$

$$v_j^{t+1} = \alpha v_j^t + \sum_{i \neq j} W_{ji}^{\text{rec}} z_i^t + \sum_i W_{ji}^{\text{in}} x_i^{t+1} - z_j^t v_{\text{th}} \quad (4)$$

$$a_j^{t+1} = \rho a_j^t + z_j^t \quad (5)$$

# Methods – e-prop

$$\frac{dE}{dW_{ji}} = \sum_t L_j^t e_{ji}^t$$

$$\frac{dE}{dW_{ji}} = \sum_t \frac{dE}{dz_j^t} \frac{\partial z_j^t}{\partial \mathbf{h}_j^t} \underbrace{\sum_{t \geq t'} \underbrace{\frac{\partial \mathbf{h}_j^t}{\partial \mathbf{h}_j^{t-1}} \cdots \frac{\partial \mathbf{h}_j^{t+1}}{\partial \mathbf{h}_j^{t'}}}_{\epsilon_{ji}^t} \cdot \frac{\partial \mathbf{h}_j^{t'}}{\partial W_{ji}}}_{e_{ji}^t}$$

# Methods – e-prop for ALIF

$$\frac{dE}{dW_{ji}} = \sum_t L_j^t e_{ji}^t$$

$$\mathbf{h}_j^t = \begin{pmatrix} v_j^t \\ a_j^t \end{pmatrix} \quad (6)$$

$$\psi_j^t = \gamma \max \left( 0, 1 - \left| \frac{v_j^t - v_{\text{th}} - \beta a_j^t}{v_{\text{th}}} \right| \right) \quad (7)$$

$$\begin{pmatrix} \epsilon_{ji,v}^{t+1} \\ \epsilon_{ji,a}^{t+1} \end{pmatrix} = \begin{pmatrix} \alpha \cdot \epsilon_{ji,v}^t + z_i^{t-1} \\ \psi_j^t \epsilon_{ji,v}^t + (\rho - \psi_j^t \beta) \epsilon_{ji,a}^t \end{pmatrix} \quad (8)$$

$$e_{ji}^t = \psi_j^t (\epsilon_{ji,v}^t - \beta \epsilon_{ji,a}^t) \quad (9)$$

# Methods – e-prop weight update

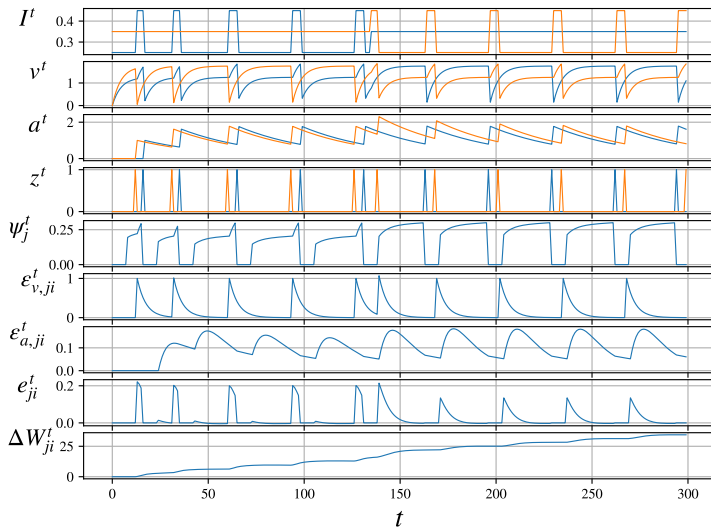
$$\Delta W_{ji} = -\eta \underbrace{\sum_t \sum_k B_{jk} (\pi_k^t - \pi_k^{*,t})}_{=L_j^t} \underbrace{\sum_{t' \leq t} \kappa^{t'-t} e_{ji}^{t'}}_{\stackrel{\text{def}}{=} \bar{e}_{ji}^t} \quad (10)$$

where  $\pi_k^t = \sigma_k(y_1^t, \dots, y_K^t)$ .

$$\Delta W_{kj}^{\text{out}} = -\eta \sum_t (\pi_k^t - \pi_k^{*,t}) \sum_{t' \leq t} \kappa^{t'-t} z_j^t \quad (11)$$

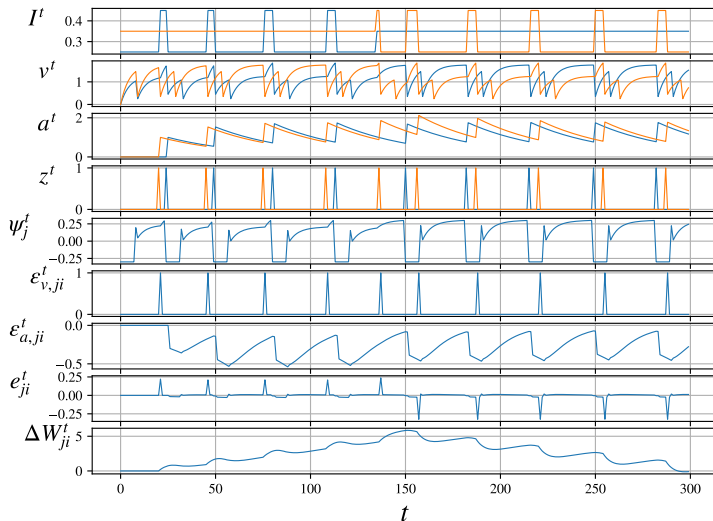
$$\Delta b_k = -\eta \sum_t (\pi_k^t - \pi_k^{*,t}) . \quad (12)$$

# Methods – Visualizing the ALIF neuron

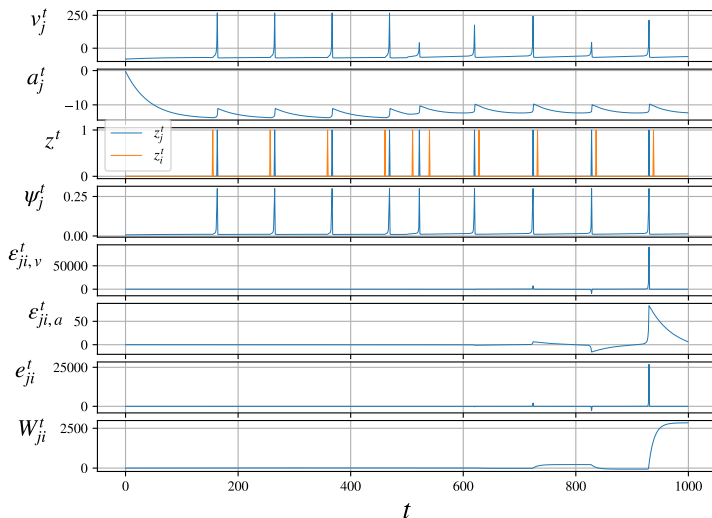




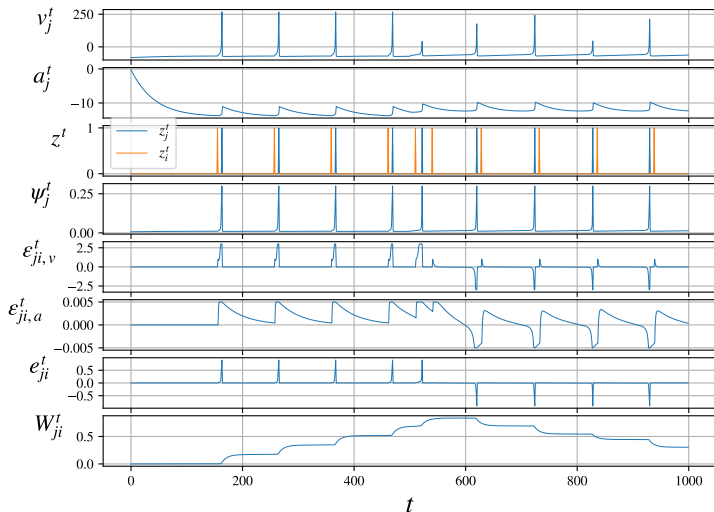
# Methods – STDP-ALIF



# Methods – Izhikevich



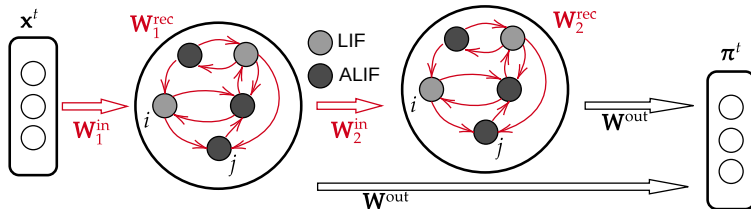
# Methods – Izhikevich fixed



# Methods – Multilayer

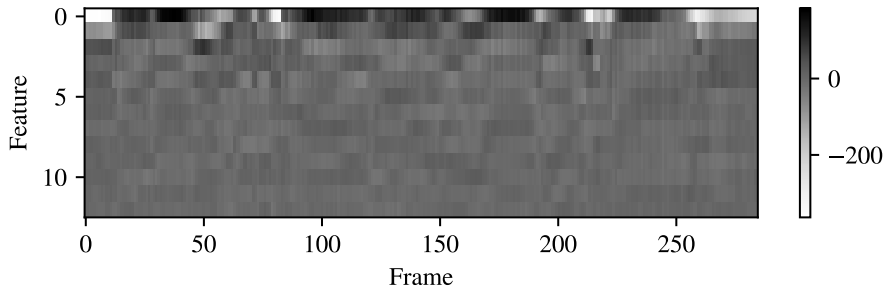
$$\mathbf{h}_{rj}^t = \begin{cases} M(\mathbf{h}_{rj}^{t-1}, \mathbf{z}_r^{t-1}, \mathbf{x}^t, \mathbf{W}_{rj}) & \text{if } r = 1 \\ M(\mathbf{h}_{rj}^{t-1}, \mathbf{z}_r^{t-1}, \mathbf{z}_{r-1}^t, \mathbf{W}_{rj}) & \text{otherwise,} \end{cases} \quad (13)$$

where  $r \in [1 \dots R]$ .



# Methods – Task

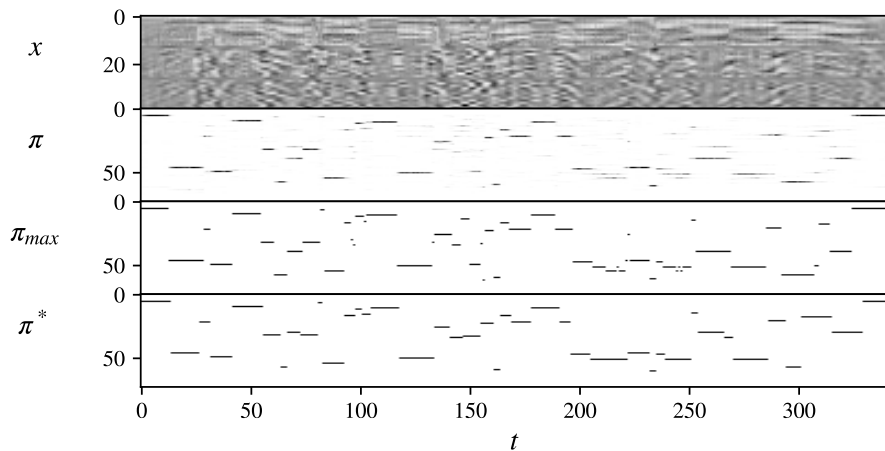
- Classify frame-wise phones from speech signals.
- There are 61 different class labels, and approximately 4000 speech sentences containing around 200-300 frames each.
- Speech is preprocessed into MFCCs (2–13) with their 1<sup>st</sup> and 2<sup>nd</sup> deltas, standardized per channel



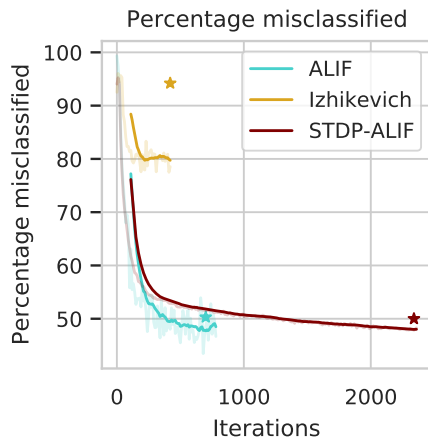
# Methods – Other settings

- Adam optimizer for e-prop
- Firing rate regularization
- L2 regularization

# Results – Example



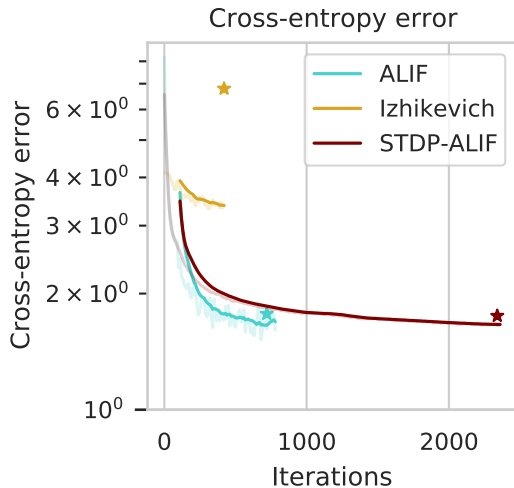
# Results – Accuracy per neuron type



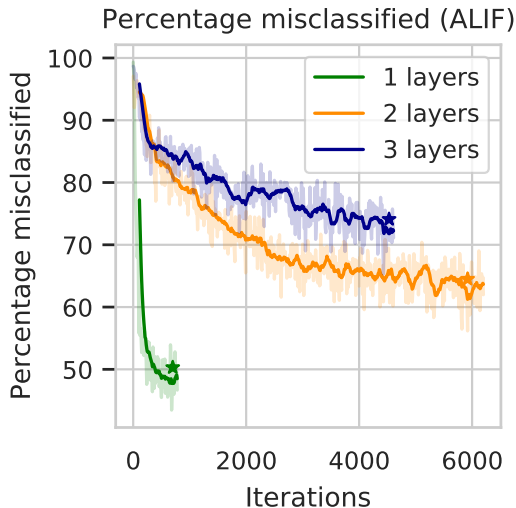
Test scores: ALIF = 50.3%, STDP-ALIF = 50%, Izhikevich = 94.2%



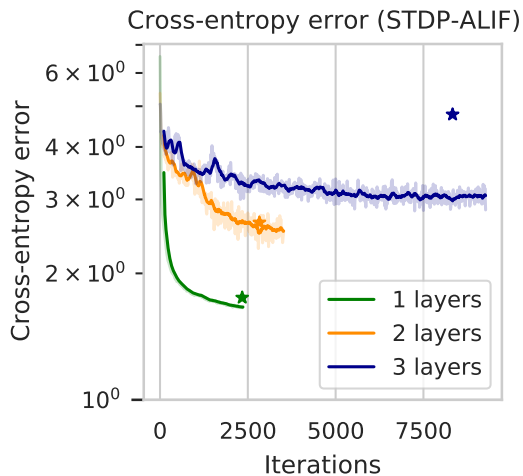
## Results – Cross-entropy per neuron type



## Results – Multi-layer effects



# Results – Multi-layer effects



# Discussion

- Izhikevich neuron unsuitable
- More layers = less efficient
- ALIF and STDP-ALIF contested
  - STDP inclusion slower but stabler and more accurate gradient descent

# Discussion – future research

- Smoothen output
- Distributed parameters
- Custom connectivity graphs
- Dynamic pruning and growing
- Synaptic delay

# Conclusion

- Good trade-off between running cost and performance
- Interesting neurocomputational model
- More bioplausibility → easier in NC but not always better performance
- More research is needed on applying e-prop in other types of learning tasks, and many ideas exist that may improve performance