Architectural Exploration for On-chip, Online Learning in Spiking Neural Networks

Subhrajit Roy, Sougata Kr. Kar and Arindam Basu
OUTLINE

• Motivation for on-chip machine-learners and online learning
• Motivation for active dendrites and structural plasticity
• Neurons with active dendrites
• Network Re-Wiring (NRW) Learning rule for structural plasticity
• Architectural Explorations
• Online NRW learning rule
• Performance and Conclusion
OUTLINE

• Motivation for on-chip machine-learners and online learning
• Motivation for active dendrites and structural plasticity
• Neurons with active dendrites
• Network Re-Wiring (NRW) Learning rule for structural plasticity
• Architectural Explorations
• Online NRW learning rule
• Performance and Conclusion
Motivation for on-chip machine-learners

**PROBLEMS**

- Data rate / channel ~ 200 Kbps
- 1000 channels → 200 Mbps
- Huge power dissipation

**UNSUSTAINABLE**

**SOLUTIONS**

- On chip neural processing unit
- Requirement for low-power hardware implementations of supervised classifiers
Motivation for online learning

- Storage space reduction by storing ‘mini-batches’
- Accommodates time constant limitation for calculating ‘true gradient’
- Robust to changes of the input data distribution
OUTLINE

• Motivation for on-chip machine-learners and online learning
• Motivation for active dendrites and structural plasticity
• Neurons with active dendrites
• Network Re-Wiring (NRW) Learning rule for structural plasticity
• Architectural Explorations
• Online NRW learning rule
• Performance and Conclusion
Motivation for active dendrites and structural plasticity

Neuromorphic sensors – Retina-inspired Imager (Lichtsteiner et al., 2008)

Spike inputs from biological neurons

Applications in face recognition, handwriting recognition

Applications in brain-machine interfaces (BMI)

Goals/Challenges

Design classifiers:

1) Reduce number of synapses without hampering accuracy
2) Learning using synapses with low resolution weights
Motivation for active dendrites and structural plasticity

Neuromorphic sensors – Retina-inspired Imager (Lichtsteiner et al., 2008)

Spike inputs from biological neurons

Applications in face recognition, handwriting recognition

Applications in brain-machine interfaces (BMI)

Solutions

1) Reduce number of synapses without hampering accuracy
   - Exploit dendritic properties
2) Learning using synapses with low resolution weights
   - Exploit structural plasticity

OUTLINE

• Motivation for on-chip machine-learners and online learning
• Motivation for active dendrites and structural plasticity
• Neurons with active dendrites
• Network Re-Wiring (NRW) Learning rule for structural plasticity
• Architectural Explorations
• Online NRW learning rule
• Performance and Conclusion
Neurons with Active Dendrites

Linear Cell w Binary Synapses

Unable to recognize the different combination of inputs
Neurons with Active Dendrites

Motivation

Source: Polsky et al., 2004

Source: Poirazi et al., 2001

#synapses per branch : \( k \ (k << d) \)

- Currently considering only lumped dendritic nonlinearity

Source: Polsky et al., 2004
Neurons with Active Dendrites

Linear Cell w Binary Synapses

Non linear cell w Binary Synapses

Unable to recognize the different combination of inputs

Capable of recognizing the different combination of inputs
OUTLINE

• Motivation for on-chip machine-learners and online learning
• Motivation for active dendrites and structural plasticity
• Neurons with active dendrites
• Network Re-Wiring (NRW) Learning rule for structural plasticity
• Architectural Explorations
• Online NRW learning rule
• Performance and Conclusion
Network Re-Wiring Learning Rule

Employ two neuronal cells for binary classification

For positive cell: \( c_{ij} = \langle (t - y)x_jx_{ij} \rangle \)  
For negative cell: \( c_{ij} = -\langle (t - y)x_jx_{ij} \rangle \)

- Search for the worst performing synapse i.e. lowest \( c_{ij} \) synapse in the set \( n_T \)
- Choose a random set \( n_R \) of the input lines → Replace by its best
OUTLINE

• Motivation for on-chip machine-learners and online learning
• Motivation for active dendrites and structural plasticity
• Neurons with active dendrites
• Network Re-Wiring (NRW) Learning rule for structural plasticity
• Architectural Explorations
• Online NRW learning rule
• Performance and Conclusion
Architectural Explorations

Arch. I

Arch. II

Arch. III

Arch. IV
Comparison of architectures

$N_c$: Number of $c_{ij}$ calculators per cell

Task considered: Classifying spatio-temporal Spike trains at the output of reservoir of Liquid State Machine

Table 1: Comparison of DER architectures*

<table>
<thead>
<tr>
<th>Arch.</th>
<th>$N_c$</th>
<th>MAE</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>$k \times m + nR \times m$ (119)</td>
<td>0.0912</td>
<td>Least $MAE = MAE_{min}$, Highest $N_c = N_c^{max}$</td>
</tr>
<tr>
<td>II</td>
<td>$m + nR \times m$ (56)</td>
<td>0.0946</td>
<td>$MAE \approx MAE_{min}$, $N_c &lt; N_c^{max}$</td>
</tr>
<tr>
<td>III</td>
<td>$m + nR$ (14)</td>
<td>0.1088</td>
<td>$MAE = 1.1930MAE_{min}$, $N_c \ll N_c^{max}$</td>
</tr>
<tr>
<td>IV</td>
<td>$k + nR$ (17)</td>
<td>0.1094</td>
<td>$MAE = 1.1996MAE_{min}$, $N_c \ll N_c^{max}$</td>
</tr>
</tbody>
</table>

*Only 1.76% MAE increase by using 8.5 times less $c_{ij}$ calculators

$k=10$, $m=7$ and $nR=7$
OUTLINE

• Motivation for on-chip machine-learners and online learning
• Motivation for active dendrites and structural plasticity
• Neurons with active dendrites
• Network Re-Wiring (NRW) Learning rule for structural plasticity
• Architectural Explorations
• Online NRW learning rule
• Performance and Conclusion
Online NRW learning rule

Timeline of batch NRW rule:

Timeline:

- $0 \rightarrow T_p$
- $2T_p \rightarrow (n-1)T_p$
- $nT_p \rightarrow PT_p$
- $(P+1)T_p \rightarrow (P+2)T_p$
- $2PT_p$

Morphological connection change takes place

Connections switched back if MAE does not decrease

$c_{ij}$ averaged over entire batch

Calculation of MAE

$T_p$ : Pattern duration and $P = \#$ patterns

LIMITATIONS

- Let $T_p = 0.5 \text{ secs}$ and $P = 100 \rightarrow PT_p = 50 \text{ secs}$
- Analog averaging circuits operate in the order of milliseconds

NOT SUITABLE $\rightarrow$ Look for Online Learning
Timeline of online NRW rule:

- **Morphological connection change takes place**
- **Connections switched back if MAE does not decrease**
- **Morphological connection change takes place**
- **Connections switched back if MAE does not decrease**
- **Completion of presentation of the entire batch**

**Timeline:**

- \( T_p \): Pattern duration
- \( t_{sp} \): Averaging time
- \( P_{conn} \): # patterns after which connection change takes place
- \( P_{MAE} \): # patterns required for MAE calculation
OUTLINE

• Motivation for on-chip machine-learners and online learning
• Motivation for active dendrites and structural plasticity
• Neurons with active dendrites
• Network Re-Wiring (NRW) Learning rule for structural plasticity
• Architectural Explorations
• Online NRW learning rule
• Performance and Conclusion
Performance of online NRW rule

\[ P_{MAE} = \frac{P}{4}, P_{conn} \text{ varied} \]

\[ P_{conn} = \frac{P}{4}, P_{MAE} \text{ varied} \]

When \( P_{MAE} = P \) and \( P_{conn} = P \)  MAE = 0.1075 → Comparable performance by using 4200 times lesser time constant averaging circuit
Conclusion

- Various architectures for hardware implementation of DER and NRW
- Devised an optimized architecture maintaining similar performance.
  - Uses 8.5 times less resources while hurting performance by 1.76%
- Proposed an online version of the NRW learning rule
  - Uses an averaging circuit having 4200 times lesser time constant while providing similar performance.
THANK YOU

QUESTIONS?
$c_{ij} = \langle x_{ij} x_j (t - y) \rangle$

**If $t = 1$**

- $x_{ij}$
- $x_j$

MUL → LPF → LTA

Out = 0 if $t = y$

If $t = 1$ LTA replaced by WTA
Experiments and Results

- **Task I**: Spike Train Classification Problem

![Diagram](image)

- Possible spike train segments
- Liquid Output for Class 1
- Liquid Output for Class 2
- Testing input spike train: Jittered version of Class 1 input
Liquid State Machine

Huge recurrent interconnection of spiking neurons

Readout can be:
• Single Perceptron
• Parallel layer of perceptrons
• Multilayer perceptrons

Synapses with random weights

No learning

Learning of only these synapses

Inputs: Spike Trains or Analog current

Task-specific training of readout

Properties of LSM

- **Why the term Liquid?** Short term memory effect leading to Temporal Integration.

**Parallel Processing:**
- Multiple readouts with the same liquid
- Each trained to perform different tasks on same inputs

**Liquid Advantages:**
1. Encodes Spike Times
2. Low dimension to high dimension \(\rightarrow\) Increases separability
3. Recurrence \(\rightarrow\) Memory effect
4. General \(\rightarrow\) Multiple features extracted
LSM w Dendritically Enhanced Readout (LSM-DER)

- The **proposed algorithm LSM-DER** constitutes the liquid of LSM followed by a two neuronal cell architecture.
- The parallel perceptron stage of LSM-PPR has been replaced by a two neuronal cell architecture.

**LSM-PPR** : $Lx_n$ synapses  
**LSM-DER** : $L$ synapses
Network Re-Wiring (NRW) Learning Rule

- \( t \) = Teacher signal, \( y \) = LSM-DER output
- Applying Gradient Descent algorithm:

\[
\Delta w_{ij} = - \frac{\partial e^2}{\partial w_{ij}} = 2 < (t - y) \frac{\partial y}{\partial w_{ij}} > \\
= 2 < (t - y) \frac{\partial g \left( f_1^{DER}(x) - f_2^{DER}(x) \right)}{\partial w_{ij}} > \\
= 2 < (t - y) \frac{\partial g \left( \sum_{j=1}^{m} b \left( \sum_{i=1}^{k} w_{ij} \right) \right)}{\partial w_{ij}} > \\
\]

For positive cell: \( \Delta w_{ij} = 2 < (t - y) g' b_j x_{ij} > \)

For negative cell: \( \Delta w_{ij} = -2 < (t - y) g' b_j x_{ij} > \)

Binary synapses in our case so \( \Delta w_{ij} \) can be considered as a fitness parameter / correlation calculator \( c_{ij} \)

For positive cell: \( c_{ij} = < (t - y) x_{ij} x_{ij} > \)

For negative cell: \( c_{ij} = -< (t - y) x_{ij} x_{ij} > \)

\( g' \) dropped for ease in h/w implementation

\( b() \) is a saturating squared non linearity