DEVICE MODELING FOR NEUROMORPHIC APPLICATIONS

L.E. Calvet, C. Bennett, D. Querlioz, Center for Nanoscience and Nanotechnology, Université Paris-Sud, France

T. Krauss, U. Schwalke, TU Darmstadt, Germany

A. Kloes and M. Schwarz NanoP, Technische Hochschule Mittelhessen, Germany
Outline

• Introduction
• Architecture simulations
• Device Modeling
• Combining device modeling and architectures
• Conclusions
The Brain vs. The Cluster

Brain
~ 20 W
~ 10^{-3} \text{ m}^3

Cluster
> 10^5 \text{ W}
> 1 \text{ m}^3
The biological advantage

### BRAIN

<table>
<thead>
<tr>
<th>Microelectronics</th>
<th>Biology</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ultrafast basic devices (&lt;100ps)</td>
<td>Slow basic devices (~ms)</td>
</tr>
<tr>
<td>Deterministic basic devices</td>
<td>Noisy or stochastic basic devices</td>
</tr>
<tr>
<td>64 bits real numbers</td>
<td>Imprecise real numbers</td>
</tr>
<tr>
<td>Operations performed sequentially</td>
<td>Ultra-massive parallelism</td>
</tr>
</tbody>
</table>
Neural Computing

- Signals travel electro-chemically
- Neurons compute and communicate
- Synapses transfer and store information

Ion Channels are the current source

~5nm
Neural-inspired Computing

Neurons: **Decide** whether to transmit information

Synapses: Memory element that can **learn**
Electronic idealization: Neuron

\[ \Sigma + \]

Winner Takes All

\( \text{MAX} \)

NEURON 1

NEURON 2
Types of synaptic devices

Filamentary devices
Barbara et al ACS Nano 2015

Ferroelectric Synapses
Chanthbouala et al Nat Mat 2012

Atomic Switches
Ohno et al Nat Mat 2011

Phase Change Synapses
Tuma et al Nat Nano 2016

Spin torque Synapses
Vincent et al IEEE Trans Biomed Cir Sys 2015

Floating Gate transistors
Mead et al IEEE Trans 1996
Learning Rules

Synaptic Learning: 
\[ \delta \text{ rule} \]

Do nothing

CHANGE BY \( \delta \)
II. Architectures
The cross bar array for neural computing

- CMOS Neurons input data
- Synapses store the weights
- Kirchoff’s law provides summation
- CMOS winner takes all neuron circuit calculates the output

\[ \text{ANSWER} = \max \left( \sum w_i x_i + b \right) \]
• **SUPERVISED:** Teacher provides a data set $X$ with known targets $T$

• **TRAINING:** Compute $(\sum w_i x_i + b)_j$ for each sample $x_i$ and each category $c_j$

- Compute the gradient descent of the square error to get the delta rule:

\[ \delta = \eta (w \cdot x - t) \]

- ‘adaptive linear neuron’ (adaline rule)

SINGLE LAYER NETWORKS KNOWN AS ‘PERCEPTRONS’
No propagation multilayer perceptron

\[
\begin{align*}
\beta \text{-matrix: min/max values randomly assigned and fixed throughout learning} \\
\text{ANSWER} = \max (\sum w_i h_i)
\end{align*}
\]

Use learning rule: \( \delta = \eta (w_j \cdot h - t_j) \)

- Inspired by ‘Extreme Learning Machine’ Algorithm
- Avoids backpropagation circuitry at the expense of having a larger number of hidden layers
Iris Classification

- 150 samples from 3 species of Iris
- 4 Features: Length and width of the sepals and petals (in cm)

Classification results using software and ideal learning:
- Perceptron: 97%
Iris Classification

Classification results using software and ideal learning:

• Perceptron: 97%
• Back-prop: 98% at 4 hidden nodes
Iris Classification

\[ \eta = 0.001 \]

Classification results using software and ideal learning:

- Perceptron: 97%
- Back-prop: 98% at 4 hidden layers
- Online ELM: 92.4% average
Transfering To Hardware

Software learning rule:
\[ \delta = \eta (w_j \cdot h - t_j) \]
\[ \eta = 0.02 \]

AVERAGE: 86.8%

Hardware learning rule:
\[ \delta = \Delta G \cdot \text{sign}(w_j \cdot h - t_j) \]
levels = 200

AVERAGE: 83.0%
Transfering To Hardware

Software learning rule:
\[ \delta = \eta (w_j \cdot h - t_j) \]
\[ \eta = 0.02 \]

Hardware learning rule:
\[ \delta = \Delta G \cdot \text{sign}(w_j \cdot h - t_j) \]
levels =200

AVERAGE: 86.8%

AVERAGE: 83.0%
III. Device Modeling
SONOS Transistor as a synapse

- SONOS allows superior number of read/write cycles than conventional floating gate MOSFETs
- Engineer structure to optimize the timing
Modeling using TCAD Synopsys

- Charge injected into the gate via Fowler Nordheim tunneling
- Find device characteristics and device operational range
- Determine conductance range of synapses, compact model of device behavior

**DEVICE PARAMETERS:**
- Tunnel oxide: 18 Å
- Si₃N₄ layer: 80 Å
- Top oxide: 40 Å
- \( L_g \): 130 nm
Device Characteristics

- To generate two curves:
  - 300 pulses +6 V (+ charge)
  - 300 pulses -3.75 V (- charge)
- Choose $V_g$ range
- Find: $G_{\text{max}}$, $G_{\text{min}}$, # levels
At each pulse ± 6V, an Id vs Vg is taken. This graph plots the fraction change of the evolution of Vg =0.75 V
IV. Combining Device and Architecture Simulations
Overview of methodology

Device Simulations to obtain $G_{\text{max}}$, $G_{\text{min}}$, $\Delta G$,

Architecture simulations with weights from device simulations

Compact model

Architecture simulations to obtain pulse sequences

Simulated weights

Pulse sequences

Device Simulations to obtain weight matrices

Dresden Sept. 3, 2017
Learning with a SONOS transistor (linear)

Learning:
• Each pulse changes by $\Delta V_t$.
• $\Delta G = \Delta V_t$.

Inputs applied to $V_{ds}$

Use Linear Regime

$I_{d} = \mu C_{ox} w/L (V_{gs} - V_t - V_{dS}/2) V_{d}$

Take weights $= V_t$

Linear

High power consumption
Learning with a SONOS transistor (exponential)

Inputs applied to $V_g$

Subthreshold current:
$$I_{\downarrow d} = I_{\downarrow o} \exp[\alpha(V_{\downarrow g} - V_{\downarrow t})]$$

Take $w = V_t$

Learning:
- Each pulse changes by $\Delta V_t$.
- $\Delta G = \exp[-\alpha \Delta V_{\downarrow t}]$

Exponential weights
Low power consumption
First Results

Device Simulations to obtain $G_{\text{max}}$, $G_{\text{min}}$, $\Delta G$,

Architecture simulations to obtain pulse sequences

Architecture simulations with weights from device simulations

Device Simulations to obtain weight matrices
Conclusions

• Device modeling of synapses in neuromorphic architectures can optimize classification

• Floating gate devices based on a layer of traps/defects are ideal for such simulations because wide variations in the trapping properties of different materials
Thank you for your attention!