Energy Efficient and Scalable Neuromemristive Computing Substrates

Dhireesha Kudithipudi†, Cory Merkel‡, James Mnatzaganian†, Nicholas Soures†, Qutaiba Saleh†

†NanoComputing Research Lab
Rochester Institute of Technology

‡Information Directorate
Air Force Research Laboratory

DISTRIBUTION STATEMENT A. Approved for public release; distribution unlimited; 27 June 2016.
Brain-inspired adaptive computing platforms based on nanoscale resistive memory (memristors)

Memristor characteristics facilitate efficient computation and learning

Improve the efficiency (over conventional computers) of natural processing tasks

DISTRIBUTION STATEMENT A. Approved for public release; distribution unlimited; 27 June 2016.
Why the Brain?

- Incredibly versatile
  - Can learn anything!
- Energy efficient
  - \(~10^{16}\) ops/sec @ a few watts!
- Robust/Resilient
  - Functions with noise!
  - Unreliable and damaged components!

---

[1] Scientific American
[3] http://sites.psu.edu/cigerber02141993/2014/04/14/are-two-halves-better-the-one-whole/

DISTRIBUTION STATEMENT A. Approved for public release; distribution unlimited; 27 June 2016.
## Brain vs. Conventional Computing

<table>
<thead>
<tr>
<th></th>
<th>Brain</th>
<th>Computer</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Signals</strong></td>
<td>Mixed signal</td>
<td>Digital</td>
</tr>
<tr>
<td><strong>Precision</strong></td>
<td>Low</td>
<td>High</td>
</tr>
<tr>
<td><strong>Parallelism</strong></td>
<td>Very high</td>
<td>Low</td>
</tr>
<tr>
<td><strong>Information Density</strong></td>
<td>High</td>
<td>Low</td>
</tr>
<tr>
<td><strong>Processing-Memory</strong></td>
<td>Closely-coupled</td>
<td>Separated</td>
</tr>
<tr>
<td><strong>Mutability</strong></td>
<td>Plastic</td>
<td>Constant</td>
</tr>
</tbody>
</table>

Brain-like computing is better for massively parallel applications with noisy data and relaxed precision requirements
Neuromemristive Systems

Concepts

High-Level Features

Low level Features

Low level features

High-Level Features

Low level Features

Low level features

Tiger Cub

Face

Ears, Fur, Stripes

Neuromemristive Systems

DISTRIBUTION STATEMENT A. Approved for public release; distribution unlimited; 27 June 2016.
Memristors for Plasticity

\[ i_m = G_m(\gamma) v_m \]

2-terminal device with state-dependent Ohm’s Law

- Compatibility with CMOS
- Memristor characteristics facilitate efficient computation and learning

DISTRIBUTION STATEMENT A. Approved for public release; distribution unlimited; 27 June 2016.
Reconfigurable Synapses

\[ s_{ij} = \left( 2 \frac{G_{m1j}}{G_{m1j} + G_{m2j}} - 1 \right) x_j = w_{ij} x_j \]

Inhibitory and Excitatory Synapses
Reconfigurable Neurons

Non-Monotonic Neuron

Energy-Delay Product

DISTRIBUTION STATEMENT A. Approved for public release; distribution unlimited; 27 June 2016.
On-Chip Training

- Variation is exploited in the training process
Random Weight Synapses

- Exploit random mismatch in current mirrors
- Control distribution with sizing

DISTRIBUTION STATEMENT A. Approved for public release; distribution unlimited; 27 June 2016.
On-Chip Training

• Least-mean squares (LMS) training algorithm:

\[ \Delta w_{ij}^{(p)} = \alpha u_j^{(p)} \left( y_i^{(p)} - \hat{y}_i^{(p)} \right) \]

Expected output

• Converted to stochastic LMS (SLMS) using proposed method:

\[ \Delta w_{ii}^{(p)} = \alpha \text{sgn} \left( y_i^{(p)} - \hat{y}_i^{(p)} \right) X_j^{(p)} \left| Y_i^{(p)} - \hat{Y}_i^{(p)} \right| \]

Sensors, real-time data

\[ u(t) \rightarrow h_W \rightarrow \hat{y}(t) \]
Unsupervised Clustering

Manhattan Distance Metric

Distance Calculation

Memristor Crossbar Synapses

Boost Update

Weight Update

Manhattan Distance Metric

Memristor Crossbar

Inputs

Distance Calculation WTA

Inputs

MATLAB

Proposed

Epoch

DISTRIBUTION STATEMENT A. Approved for public release; distribution unlimited; 27 June 2016.
Hierarchical Temporal Memory

Inspiration/Motivation

- Inspired by the neocortex
- Highly parallelizable
- Suitable for hardware design

Critical Aspects

- Spatiotemporal data
- Online, unsupervised learning
- Classification & prediction
- Distinct learning components
- Customizable architecture

Applications/Results

Given: (A) 11111−−−−−−−−−−−−− Predicted: (?) −−−−−−−−−−−−−

A. Approved for public release; distribution unlimited; 27 June 2016.
Mathematical Formalization of the Spatial Pooler

\[ \hat{\alpha} = \{ 0, 1, 1, 3, 2, 2, 1, 3, 5, 4, 6, 2 \} \]

\[ \hat{\alpha} \equiv \begin{cases} \hat{\alpha}_i b_i & \hat{\alpha}_i \geq \rho_d, \quad \forall i \quad \hat{\alpha}_i \equiv X_i \cdot Y_i \\ 0 & \text{otherwise} \end{cases} \]

Overlap

\[ \hat{c} \equiv I(\hat{\alpha}_i \geq \hat{\gamma}_i) \quad \forall i \]

inhibition

\[ \hat{\gamma} \equiv \max(\text{kmax}(H_i \odot \hat{\alpha}, \rho_c), 1) \quad \forall i \]

Learning

\[ \delta \Phi \equiv \hat{c}^T \odot (\phi_+ X - (\phi_- \neg X)) \]

\[ \Phi \equiv \text{clip}(\Phi \oplus \delta \Phi, 0, 1) \]
Reconfigurable HTM Architecture

Storage processor units may leverage PCIe SSD technology.
Reconfigurable HTM Architecture
Generalizable Intelligence Engine

DISTRIBUTION STATEMENT A. Approved for public release; distribution unlimited; 27 June 2016.
Reconfigurable Reservoir Architecture

DISTRIBUTION STATEMENT A. Approved for public release; distribution unlimited; 27 June 2016.
Reconfigurable Reservoir Architecture

User Authentication based on Gait Patterns

Rebooting’16
Smart Grid Load Forecasting

Actual Electric Load Data from Grid

US Mid-Atlantic Region Energy dissipation in Joules/Hour

Stochastic Learning System

Global Trainer

1 XOR gate
12 Comparators
3 MUX gates
3 AND gates
3 Counters

3 Synaptic Trainers

System Inputs

\[ i_1 = L(t) - L(t-1) \]
\[ i_2 = L(t-1) - L(t-2) \]

(L(t) : Load value at hour ‘t’)

Pre-processed Input data

Bias i1 i2

3 Synapses:
60 CBRAM devices
(30 inhibitory)
(30 excitatory)

Post-synaptic signal

Output Neuron

Forecasted Load value for hour ‘t’
Smart Grid Load Forecasting

![Graph showing load forecasting comparison]

- **Actual Load Data**
- **Predicted—Ideal Synapse**
- **Predicted—CBRAM (Before Training)**
- **Predicted—CBRAM (After Training)**

**Graph Details:**
- **Y-axis:** Load [MW]
- **X-axis:** Time [hours]
- **Scatter plots** illustrating load data and predictions over time.
Summary

▪ Reconfigurability is integral to the nature of computation
  – Precomputation is occurring in communication channels
  – No standardized metrics/benchmarks to evaluate
  – Designing technology agnostic vs. technology aware systems

▪ Looking forward
  – one shot learning ....
Team & Collaborators

DISTRIBUTION STATEMENT A. Approved for public release; distribution unlimited; 27 June 2016.