In-Memory Computing with Imperfect or Unreliable Memory Devices

Damien Querlioz, Tifenn Hirtzlin, Jacques-Olivier Klein, Clément Turck, Kamel-Eddine Harabi  
Université Paris-Saclay, CNRS, C2N, Palaiseau

Marc Bocquet, Jean-Michel Portal  
IM2NP, Aix-Marseille Univ, CNRS

Elisa Vianello, Thomas Dalgaty, Niccolo Castellani, Etienne Nowak,  
CEA, LETI, Grenoble
Context: the AI Energy Problem

- AI of self-driving car prototypes ~ 2,000 W
- Human brain ~20W
- Training the GPT-3 language model required 190,000 kWh
- *1,000 years of brain operation!*

Due to this energy problem, most AI is done on the cloud.
This Energy Cost Largely Originates from the Separation of Computation and Memory

- In a modern computer

<table>
<thead>
<tr>
<th>Operation</th>
<th>Energy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Addition of data (fixed point)</td>
<td>1x</td>
</tr>
<tr>
<td>Access data (onchip cache)</td>
<td>60x</td>
</tr>
<tr>
<td>Access data (offchip RAM)</td>
<td>3500x</td>
</tr>
</tbody>
</table>

Pedram et al., IEEE D&T 2016

Absent in the brain!

In-memory computing approach of the brain could dramatically reduce the energy consumption of AI
The Solution: Resistive Memory (Memristor) and Other Emerging Memories

• Fast, non-volatile memory that can be embedded at the core of CMOS
• Memory state is the electrical resistance of the device
• Many variations (memristive, phase change / PCM, magnetoresistive / MRAM)
• In industry test production (Samsung, TSMC, Intel, ST Microelectronics...)

Can be ideal technology for merging logic and memory
The Challenge: Device Imperfection!

- Nanometer-scale physics is very noisy and prone to variations
- Emerging memories well described by the tools of **statistics**

**Cumulative Distribution Function (Number of Sigmas)**

- *HfO$_2$ RRAM* programmed at high voltage
- *HfO$_2$ RRAM* programmed at low voltage
Dealing with Device Imperfections

• Intel RRAM (memristor) 22 nm process

• Current products with emerging memories:
  multiple error correcting codes (ECC)
  – Intel RRAM and MRAM: ECC corrects up to 3 errors (TEC) per 128b

Error detection and correction is costly in terms of delay, energy, area
The brain also functions with noisy nanodevices and does **not** use formal ECC.

This talk: embrace the statistical nature of emerging memories for neuromorphic computing.
Memory-Centric Artificial Intelligence

• Circuits that work even if the devices make many errors – *Binarized Neural Networks*

• Exploiting the statistical nature of memristors – *Markov Chain Monte Carlo learning*
Binarized Neural network: A “Super Low Precision” Neural Networks

Hubara, Courbariaux et al. NIPS 2016
Yoshua Bengio’s group

Can approach state of the art performance on vision tasks!
Binarized Neural network: Very Simple Logical Operations

Multiplication → Accumulation → Non-linear function

\[ W_{ij} \cdot a_j \]
\[ \sum_j W_{ij} \cdot a_j \]
\[ a_i = f \left( \sum_j W_{ij} \cdot a_j \right) \]

XNOR → Bitcount → Sign

Binarized Neural Networks inference is particularly adapted for in-Memory / near-Memory Computing
Training Is Not Binarized

- During training, synapses are associated with a *hidden* real weight
- The binary weight is the sign of the real weight

Binarized Neural Networks are truly attractive for *inference* hardware
How to Deal With Bit Errors?

Classical Approach: Error Correction Code

→ A problem with in-memory computing

ECC decoding would use most of the area and energy consumption!
Memory Array for ECC-less In-Memory Binarized Neural Networks

130nm CMOS + HfO$_2$ RRAM

Reducing the Error Rate without ECC: Two RRAM Devices as One Binary Synapse

Devices programmed in a complementary fashion

Circuit to differentiate resistance state

Pre-Charge Sense Amplifier (PCSA)

Doubles the number of devices

Logic in Memory Reading Circuit

- No ECC offers opportunity for in memory operation
- XNOR operation directly in sense amplifier circuit

\[ \text{XNOR} \rightarrow \text{Bitcount} \rightarrow \text{Sign} \]

\[ a_{\text{prev}0} \cup \ldots \cup W^{b}_{i0} \rightarrow \text{sign} \sum_{a_{\text{post}i}} \ldots \]

\[ a_{\text{prev}j} \cup \ldots \cup W^{b}_{ij} \rightarrow \text{sign} \sum_{a_{\text{post}i}} \ldots \]

\[ a_{\text{prev}n} \cup \ldots \cup W^{b}_{in} \rightarrow \text{sign} \sum_{a_{\text{post}i}} \ldots \]

Weisheng Zhao, … Klein, Querlioz …., *IEEE TCAS I:* 2014.
IEEE Guillemin-Cauer Best Paper Award

2T2R Architecture to Limit Bit Errors without ECC Decoding

- **Experimental bit error rate**

- **Error Correction Code SECDED**

Bocquet et al., IEEE IEDM, p. 20.6.1, 2018; Hirtzlin et al., Front. Neurosci. 13, p. 1383, 2020
Do All Errors Really Need to Be Corrected?

Validation Accuracy (%)

Weight Bit Error Rate

- MNIST
- CIFAR-10
- ImageNet (Top-5)
- ImageNet (Top-1)

Bocquet et al, IEEE IEDM, 2018; Hirtzlin et al, IEEE AICAS 2019
## Benefits of Embracing Errors

<table>
<thead>
<tr>
<th></th>
<th>Typical</th>
<th>Optimized Endurance</th>
<th>Optimized Programming Energy</th>
</tr>
</thead>
<tbody>
<tr>
<td>SET Compliance</td>
<td>200 µA</td>
<td>200 µA</td>
<td>200 µA</td>
</tr>
<tr>
<td>RESET Voltage</td>
<td>2.5 V</td>
<td>1.5 V</td>
<td>2 V</td>
</tr>
<tr>
<td>Programming time</td>
<td>1 µs</td>
<td>1 µs</td>
<td>0.1 µs</td>
</tr>
<tr>
<td>2T2R BER</td>
<td>(&lt;10^{-7})</td>
<td>(&lt;10^{-4})</td>
<td>(&lt;10^{-5})</td>
</tr>
<tr>
<td>Programming Energy (SET/RESET)</td>
<td>(~ 300 \text{ pJ}~)</td>
<td>(~ 300 \text{ pJ}~)</td>
<td>(~ 25 \text{ pJ}~)</td>
</tr>
<tr>
<td>Cyclability</td>
<td>(&gt;10^8)</td>
<td>(&gt;10^{10})</td>
<td>(&gt;10^8)</td>
</tr>
</tbody>
</table>

The sum is achieved with few bits integer digital circuits
  – *We do not use Kirchhoff laws*
Energy Benefits of BNN

Inference Energy for Recognizing a MNIST Digit:
- CPU / GPU \( \sim \) mJ

- In-memory ASIC:

![Graph showing validation accuracy versus inference energy](image)

(only arithmetic operations, 8 bits)
(full system)

Analysis with Cadence encounter, and DK of a 28 nm commercial technology

Hirtzlin et al., Front. Neurosci. 13, p. 1383, 2020
Possible Application of In-Memory BNN: Edge Analysis of Medical Signals

- Could allow detection of strokes, epileptic seizures, heart attacks or BCI
- Without relying on the cloud (better privacy, security, reliability)

However, AI algorithms have high energy cost

B. Penkovsky et al., DATE, p. 690, 2020
Application to Biomedical Signal Analysis

- We tried two tasks
  - ECG electrodes misplacement
  - EEG motor movement/imagery
- 1-D convolutional neural network

BNN implementation saves 76% memory wrt. 8-bit precision neural network for ECG (58% for EEG), at equivalent accuracy

B. Penkovsky et al., DATE, p. 690, 2020
• Circuits that work even if the devices make many errors – *Binarized Neural Networks*

• Exploiting the statistical nature of memristors – *Markov Chain Monte Carlo learning*
What Happens If We Implement Backpropagation with Memristors

New challenge: learning with memristors

Idea: memristor conductance represents synaptic weights (real)

Backpropagation requires very fine tuning of the weight: difficult to achieve due to memristor imperfections!
Learning by Embracing the Statistical Nature of Memristors

Learning with Metropolis-Hastings Markov Chain Monte Carlo (MCMC)

The jumps $p(g_{n+1}|g_n)$ can be performed easily using the statistical behavior of memristors!

Memristors are ideal for MCMC-based learning!
Memristor-Based MCMC in Practice

Initialise model $g$

Sample $p(g_p | g)$

Compare $\frac{p(g_p) p(t | g_p, V)}{p(g) p(t | g, V)}$

ACCEPT/REJECT

Learning takes place inside of the memory!

Dalgaty et al, Nature Electronics 4, p. 151, 2021
Supervised Learning with Memristor-Based MCMC

Computer-in-the-loop experiment with an array of 16,384 memristors


Dalgaty et al, Nature Electronics 4, p. 151, 2021
Supervised Learning with Memristor-Based MCMC

The experimental system was able to detect malignant tissue with 98% accuracy

Dalgaty et al, Nature Electronics 4, p. 151, 2021
MCMC Learning Leads to a collection of Models: It Provides a *Bayesian* Model

Bayesian model parameters / activations are probability distributions describing uncertainty

Particularly adapted for the “small data” world, which has a lot of uncertainty

Why might output uncertainty be useful?

Deterministic model

‘You have input a 3’

Bayesian model

‘The input looks most like a 3… but I am very uncertain about that’
MCMC Learning Is Highly Energy-Efficient

Preliminary mixed-signal design results (full model training)

Intel Xeon processor (7nm) implementation of MCMC sampling required 600mJ

<table>
<thead>
<tr>
<th></th>
<th>Step 1 (Model evaluation)</th>
<th>Step 2 (Model acceptance/rejection)</th>
<th>Step 3 (RRAM programming)</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of repetitions</td>
<td>$500 \times 10 \times 512$</td>
<td>$10 \times 512$</td>
<td>$10 \times 512$</td>
<td></td>
</tr>
<tr>
<td>Total energy (130nm)</td>
<td>$5.8\mu J$</td>
<td>$120nJ$</td>
<td>$1.1\mu J$</td>
<td>$6.9\mu J$</td>
</tr>
<tr>
<td>Total energy (28nm)</td>
<td>$2.5\mu J$</td>
<td>$34nJ$</td>
<td>$1.1\mu J$</td>
<td>$3.6\mu J$</td>
</tr>
</tbody>
</table>

Dalgaty et al, Nature Electronics 4, p. 151, 2021
RRAM devices are prone to aging.

MCMC still works even if devices have been programmed millions of times, and would be unusable as conventional memory.

Dalgaty et al, Nature Electronics 4, p. 151, 2021
Other Approaches that Exploit Memory Devices Imperfections

- **TRNGs and PUFs**

- **Bayesian Neural Network Inference**

- **Stochastic Computing-Based Optimization**
  - Borders et al., Nature 573, 390 (2019)
Conclusions

• Binarized Neural Networks excellent candidates for In-Memory Computing with emerging memories

• Embracing bit errors has important benefits in terms of programming energy, cell area and reliability

• The statistical nature of the devices can even be exploited for learning Bayesian models
Thank You for Your Attention!

I am hiring postdocs