Neuromorphic computing

Daniele Ielmini

Dipartimento di Elettronica, Informazione e Bioingegneria
Politecnico di Milano
daniele.ielmini@polimi.it
Schedule of the course

Feb. 14  Moore’s law, novel CMOS materials/devices
Feb. 16  Memristive technology
Feb. 17  Memristive computing
Feb. 21  Spintronic computing
Feb. 23  Quantum computing
Mar.  3   Neuromorphic computing
Mar.  9   Interactive seminars

Lecture hours: 9am-1pm
Where: room Alpha/Beta at DEIB, building 24, Via Golgi 40
Online information:  http://home.deib.polimi.it/ielmini/psc.htm
Or contact me  daniele.ielmini@polimi.it
After the end of Moore’s law
3 grand challenges of future systems

How to store all these data?

How to efficiently compute with large amounts of data?

How to extract information from unstructured data?

Are CMOS Si-based devices adequate to meet these challenges?
Software intelligence

Hardware intelligence

Robot navigation

Traffic monitoring/vehicle detection

Drone navigation

Facial recognition

Epileptic seizure prediction


Post-Si computing – 6
Outline

• Neural networks
  ▪ Supervised learning
  ▪ Unsupervised learning
• Neuromorphic circuits
• Memristive synapses
• Summary
Outline

• Neural networks
  ▪ Supervised learning
  ▪ Unsupervised learning
• Neuromorphic circuits
• Memristive synapses
• Summary
Neuromorphic architectures

- Density in the human cortex:
  - Cells = $10^7$ cm$^{-2}$
  - Synapses = $10^{11}$ cm$^{-2}$ (10$^4$ average connectivity)

<table>
<thead>
<tr>
<th>In vivo</th>
<th>In silico</th>
</tr>
</thead>
<tbody>
<tr>
<td>Neuron</td>
<td>CMOS</td>
</tr>
<tr>
<td>Axons/dendrites</td>
<td>Interconnect</td>
</tr>
<tr>
<td>Synapses</td>
<td>RRAM/memristors</td>
</tr>
</tbody>
</table>
Neural networks and neuromorphic circuits

- **Neural networks**: inspired generically by neural system with neuron cells + synaptic connections, e.g., cellular neural networks, accelerated artificial networks, etc.

- **Neuromorphic networks**:
  - Compact, real-time, energy-efficient computing to emulate the style of the brain
  - Same timing as the brain allows for brain-machine interfaces or robots interacting with real world and humans
  - Allows to explore the computational properties of the neural system they emulate and thus gain a better understanding of its operational principles
  - ‘Neuromorphic’ first coined by Carver Mead
The neuron

- **Soma** = cell body
- **Dendrites** = input wires for incoming spikes
- **Axon** = output wire to send the neuron spike
- Each axon terminates with a synapse connecting to other dendrites
Neuron output can be described mathematically by: $a_i = g\left(\sum_{j=0}^{m} W_{ji} a_j\right)$

- $a_j =$ input coming from presynaptic neurons
- $W_{ji} =$ synapse weights
- $g = \text{threshold function, e.g., Heavyside step, or sigmoid}$
Learning networks: the perceptron

- Introduced by Minsky in the late 1950
- Modeled by convergence theorem by Rosenblatt, 1962
- Perceptron can be trained to learn and classify patterns
Perceptron as a universal logic gate

- **AND**
  - \( W = 1 \)
  - \( t = 1.5 \)

- **OR**
  - \( W = 1 \)
  - \( t = 0.5 \)

- **NOT**
  - \( W = -1 \)
  - \( t = -0.5 \)

- **I_1 and I_2**
  - \( I_1 \) and \( I_2 \)
  - \( I_1 \) or \( I_2 \)
  - \( I_1 \) xor \( I_2 \)
• $x_1, x_2 = \text{generic inputs}$
• Line = separation between set of classified inputs and set of unclassified inputs
• For a one-layer perceptron to work, the inputs must be linearly separable for convergence
Supervised learning

• Output is already known (not bio-inspired)
• Synapse weights are externally updated until the perceptron yields the correct answer
• Backpropagation algorithm:
  - Let $y$ be the correct output, and $f(x)$ the current output function of the perceptron ($x = \text{input vector}$)
  - Let $\Delta$ be the error, namely $\Delta = y - f(x)$
  - Weight update according to $W_j' = W_j + \alpha x_j \Delta$
Outline

• Neural networks
  ▪ Supervised learning
  ▪ Unsupervised learning
• Neuromorphic circuits
• Memristive synapses
• Summary
Unsupervised learning

- Hebb’s rule = neurons that fire together, wire together
- Hebb’s rule awards synapses where there is a temporal coincidence between pre- and post-synaptic activity
- Rate-based learning rules:
  - Malsburg (1970): synaptic potentiation is proportional to the product of pre- and post-synaptic activities
  - Bienenstock-Cooper-Munro (BCM, 1992):
    - Plasticity: \( \frac{dW_i}{dt} = x_i \phi(y, \theta_m) \)
    - \( W_i \) = weight, \( x_i \) = pre-synaptic rate, \( \phi \) = non-linear function of post-synaptic rate \( y \), \( \theta_m \) = threshold which moves according to negative feedback
    - Metaplasticity: \( \theta_m = \langle y^{1+\mu} \rangle \)
    - \( \langle \rangle \) is the sliding temporal average
Rate-based and time-based learning rules

A.

**Assumptions**

**BCM theory**

- Synaptic Change vs. Postsynaptic rate

B.

**Pair-based kernel theory**

- Synaptic change vs. \( \Delta t \) and Postsynaptic rate

Consequences:

- Receptive fields plasticity
- Phenomenological rule

- Rate based model

References:

- Bienenstock et al 1982
- Cooper et al (book) 2004
- Kempter et al 1999
Spike-timing dependent plasticity (STDP)

$\Delta t = t_{\text{POST}} - t_{\text{PRE}}$

$\Delta t > 0$ Potentiation

$\Delta t < 0$ Depression

PRE

Pre-synaptic neuron

synapse

POST

Post-synaptic neuron
Evidence in biological systems


- Pairing of isolated pre/post spikes leads to LTD only
- Pairing with post-synaptic burst leads to LTD and LTP

Triplet learning rule

C. Triplet based kernel theory

- Triplet LTP + pair LTD

3 point kernel

LTP magnitude vs. Δt1 and Δt2

Pfister, Gerstner 2006
Gjorgjieva et al 2011

Synaptic Change
Postsynaptic rate

D. CaDP (single coincidence detector)

Biophysical assumptions

Shouval et al 2002

Induction protocol (STDP)

Normalized weight (%)

Biophysical rule

Post-Si computing – 6
Daniele Ielmini
• Neural networks
  ▪ Supervised learning
  ▪ Unsupervised learning
• Neuromorphic circuits
• Memristive synapses
• Summary
Analog CMOS VLSI circuits

- Non-von Neumann computation:
  - Massively parallel architecture
  - In-memory computing

- Time represents itself → asynchronous, real-time operation

- ms-scale of computation to interact with the environment

- Solution = log-domain subthreshold circuits
  - Low power consumption
  - Transport = diffusion, similar to biological ionic channels
  - Current can be adjusted to achieve response in the ms-timescale
Hodgkin-Huxley (HH) neuron model

- $I_{\text{leak}}$ leads to an increase of $V_m$, which opens the Na channel
- Large Na current causes $V_m$ to rise steeply to $E_{Na}$ (pull up)
- $V_m$ eventually opens the K channel causing $V_m$ to reduce to $E_K$ (pull down)
- $V_m$ goes back to sub-threshold regime $\rightarrow$ refractory period with $V_m = E_R$
HH neuron circuit

Voltage spike shape and refractory time

\[ V_m (V) \]

\[ V_n (V) \]

\[ \text{Time (10}^{-3} \text{ Sec)} \]

• Neural networks
  - Supervised learning
  - Unsupervised learning
• Neuromorphic circuits
• Memristive synapses
• Summary
CMOS neuron and memristive synapses

• RRAM has been identified as a potential synapse

• For synaptic behavior, RRAM must display 3 fundamental properties:
  1. Controllable R increase/decrease
  2. Stochastic switching
  3. Scaling
Memristors devices and physics

- **RRAM – resistive switching memory**
- **PCM – phase change memory**
- **STTRAM – spin torque transfer magnetic memory**

![Graph showing current vs. voltage for memristor devices.](image)

- **Set 10 kΩ**
- **Reset 10 MΩ**

- **HfO₂**
- **TiN**
- **Top Electrode**
- **a-GST**
- **c-GST**
- **Filament**
- **Bottom Electrode**
- **Heater**
- **Insulator**
- **Vdd**
- **ON**
- **0**

Post-Si computing – 6
Daniele Ielmini
The left side of the U is the basis of a fundamental change in computing that while it can easily do conventional Von Neumann binary operations, it is also possible in a single spot to do higher mathematical and cognitive computing. That is, the objective of building intelligence, such as learning capability and adaptability, into the computer is possible. In fact, the activity represented by the left side of the U very much resembles the neurosynaptic activity of the brain. Keep in mind that the left and right side describe memory in a single spot and cooperatively work together to perform tasks heretofore not possible in semiconductors. An amazing situation for a single device.
Rate-dependent potentiation

![Diagram of synaptic connections and conductance over time for two different time periods (T = 20 s and T = 2 s).](image)
Sensory, short-term and long-term memory

a. Sensory memory (SM), short-term memory (STM) and long term memory (LTM)

b. Rehearsal needs to be carried out at high rate to achieve LTM

c. The more rehearsals, the more retention

Supervised training

- Output is already known (not bio-inspired)
- Synapse weights are externally updated until the perceptron yields the correct answer
- Backpropagation algorithm:
  - Let \( y \) be the correct output, and \( f(x) \) the current output function of the perceptron (\( x = \) input vector)
  - Let \( \Delta \) be the error, namely \( \Delta = y - f(x) \)
  - Weight update according to \( W'_j = W_j + \alpha x_j \Delta \)
Backpropagation algorithm

1) Forward propagation

\[ X_j^B = f(\sum x_i^A w_{ij}) \]

\[ \Delta w_{ij} = \eta \cdot x_i \cdot \delta_j \]

2) Compare against correct answer

\[ y_1 - g_1 = 0 \]

\[ y_2 - g_2 = 1 \]

\[ y_k - g_k = \text{NN results} \]

\[ y_{10} - g_{10} = 9 \]

3) Backpropagation

G. W. Burr, et al., IEDM Tech. Dig. (2014)
Implementation with PCM synapses

G. W. Burr, et al., IEDM Tech. Dig. (2014)
Linear and symmetric synapses

- Gradual potentiation and depression
- Linearity ($\alpha = 1$)
- Symmetry

Impact of update characteristics

Identical pulses

Non-identical pulses

Linear & symmetric

Non-linear & symmetric

Non-linear & symmetric

Non-ideal memristors

- Linearity and symmetry are required, but difficult to realize in practical memristors
- Variability and noise further degrade performance

S. Yu, RRAM workshop (2015)
### Other synapse specs

<table>
<thead>
<tr>
<th>Metrics</th>
<th>Desired Targets</th>
</tr>
</thead>
<tbody>
<tr>
<td>Device Dimension</td>
<td>&lt;10 nm</td>
</tr>
<tr>
<td>Programming Voltage</td>
<td>~1 V</td>
</tr>
<tr>
<td>Energy Consumption</td>
<td>&lt;10 fJ/spike</td>
</tr>
<tr>
<td>Multilevel States</td>
<td>&gt;100</td>
</tr>
<tr>
<td>Dynamic Range (on/off ratio)</td>
<td>&gt;100</td>
</tr>
<tr>
<td>Endurance</td>
<td>&gt;1E9 (for online)</td>
</tr>
<tr>
<td>Retention</td>
<td>&gt;years (for offline)</td>
</tr>
</tbody>
</table>

S. Yu, RRAM workshop (2015)
Unsupervised learning

• Hebb’s rule (neurons that fire together, wire together) awards synapses where there is a temporal coincidence between pre- and post-synaptic activity

• Rate-based learning rules:
  - Malsburg (1970): synaptic potentiation is proportional to the product of pre- and post-synaptic activities
  - Bienenstock-Cooper-Munro (BCM, 1992):
    - Plasticity: \( \frac{dW_i}{dt} = x_i \phi(y, \theta_m) \)
      - \( W_i \) = weight, \( x_i \) = pre-synaptic rate, \( \phi \) = non-linear function of post-synaptic rate \( y \), \( \theta_m \) = threshold which moves according to negative feedback (\( \theta_m = \langle y^{1+\mu} \rangle \) where \( \langle \cdot \rangle \) is a sliding temporal average)

• Time-based learning rules: spike timing dependent plasticity (STDP)
Spike timing dependent plasticity (STDP)

G.-Q. Bi and M.-M. Poo, J. Neuroscience 18, 1998


STDP achieved in memristors thanks to external time-voltage conversion
- Synapse connecting PRE and POST
- Train of pulses with reducing pulse width
- 5 timeslots for COMM, LTP+, LTP-, LTD+, LTD-

G. S. Snider, IEEE/ACM NANOARCH 85, 2008
Long-term potentiation (LTP)

- Only 3 time slots for (0) communication, (1) LTD and (2) LTP
- Train of pulses with reducing width → STDP

G. S. Snider, IEEE/ACM NANOARCH 85, 2008
G. S. Snider, IEEE/ACM NANOARCH 85, 2008

- LTP or LTD are achieved depending on positive/negative delay between COM spikes
- Time modulation is not effective for continuous variation of LTP and LTD
Phase-change memristor

HfO$_x$-AlO$_x$ memristor

- Voltage, not time, is modulated for better STDP efficiency

Double-spike STDP

Performance and limits

- Voltage-driven is superior to time-driven STDP (exponential vs. linear)\[ \frac{d\phi}{dt} = Ae^{\frac{E_A(V)}{kT(V)}} \]

- However, resistance control is most efficient through:
  - Voltage control during reset or LTD
  - Current control during set or LTP

- Sub-threshold pulses may affect STDP \( \rightarrow \) LTP/LTD pulses must be gated

- Due to the large area for neuron circuit, there is sufficient area for more complicated synapse
Controllable switching and variability

- I-controlled set
- V-controlled reset


Cycle-to-cycle variation of set/reset characteristics
Hybrid CMOS/RRAM synapse

1T1R synapse

V_G \quad V_{TE} \quad V_{TE}

2T1R synapse

V_{TE} \quad V_{CG} \quad V_{FG}

\[ \Delta G/G = \begin{cases} -1 \text{ V} & \quad \text{if } V_{TE0} = -2 \text{ V} \\ \end{cases} \]


2-transistor/1-resistor (2T1R) synapse

- PRE neuron
- POST-neuron
- Communication gate
- Fire gate
- Integrate
- Fire

Symbols:
- $V_{TE}$
- $V_{CG}$
- $V_{FG}$
- $V_{int}$
Communication

PRE neuron

\[ V_{CG} \]

Communication gate

Fire gate

\[ V_{TE} \]

\[ V_{FG} \]

POST neuron

\[ C \]

\[ V_{int} \]

Integrate

Fire

\[ V_{TE} \]

\[ V_{CG} \]

\[ V_{TE,max} \]

\[ V_{TE,min} \]

\[ V_{int} \]

150 ms

1 ms
Long-term potentiation (LTP)

**Diagram:**
- **PRE neuron** connected to **POST neuron**
  - **V_{TE}** and **V_{CG}**
  - **Fire gate**
  - **Communication gate**
  - **BE**
  - **Integrate** and **Fire** block

**Graph:**
- **V_{TE}** vs. **t**
  - **V_{TE, max}** and **V_{TE, min}**
  - **Δt > 0**
  - **1 ms** and **150 ms**

**Equations:**
- \( V_{TE,max} \)
- \( V_{TE, min} \)
- \( V_{TFG} \)
- \( V_{int} \)
- \( V_{TE} \)
- \( V_{CG} \)
Long-term depression (LTD)

- **PRE neuron**
- **POST neuron**

**Diagram Elements**
- **V_{TE}**
- **V_{CG}**
- **V_{FG}**
- **Fire gate**
- **Communication gate**
- **BE**
- **Integrate Fire**
- **C**

**Graph**
- **V_{TE}**
- **V_{TE,max}**
- **V_{TE,min}**
- **1 ms**
- **Δt < 0**

**Text**
- Long-term depression (LTD)
- Post-Si computing
- PRE neuron
- POST neuron
- V_{TE} (Voltage)
- V_{CG} (Voltage)
- V_{FG} (Voltage)
- Fire gate
- Communication gate
- BE
- Integrate Fire
- C

**Equations**
- V_{TE,max} = 150 ms
- V_{TE,min} = 1 ms
- Δt < 0
STDP characteristics

\[ R/R_0 \]

\[ \Delta t [\text{ms}] \]


Random STDP

Measured

Calculated

$V_{TE}$

$V_{FG}$

$\Delta t > 0$

$\Delta t < 0$

Initial $R$

$10^5$

$10^4$

$10^3$

$10^2$

$10^1$

$10^0$

$10^{-1}$

$10^{-2}$

$10^{-3}$

$10^{-4}$

$10^{-5}$

$10^{-6}$

$\Delta t$ [ms]

$R_0/R$

$10^2$

$10^1$

$10^0$

$10^{-1}$

$10^{-2}$

$10^{-3}$

$10^{-4}$

$10^{-5}$

$10^{-6}$

$\Delta t$ [ms]
2-layer network for pattern learning

Pattern

First layer
N neurons
(retina)

Synapses
First layer
N neurons
(retina)

Pattern

Second layer
M neurons

Schematic layout

PRE

POST

Pattern

First layer
N neurons
(retina)

Second layer
M neurons
Submit ‘X’, learn ‘X’

‘X’ synapses
Other synapses
Learning, then forgetting

Pattern ‘X’

Pattern ‘O’

‘X’ synapses
‘O’ synapses
Other synapses
64 PRE x 4 POST network

Post-Si computing – 6
2T1R synapse by IBM

- Similar design but integrated 2T1R structure
- LIF (leaky integrate & fire) → communication gate
- STDP → fire gate
- PCM instead of a RRAM, advantage = unipolar operation, no need for negative voltages
- Write currents can be optimized at the expense of large R, noise and variability
LIF and STDP modes

- LIF mode = $C$ is discharged in correspondence of PRE spikes
- STDP mode = as $V_{\text{cap}} < V_{\text{th}}$, STDP WL is activated, causing weight update depending on delay with respect to STDP BL from PRE

Chip and STDP characteristics

(a) 2T-1R PCM synaptic memory array

(b) PCM cell operation mode

<table>
<thead>
<tr>
<th>Neuron circuit (NC)</th>
<th>LIF</th>
<th>STDP</th>
<th>Refractory</th>
</tr>
</thead>
<tbody>
<tr>
<td>Axon driver (AD)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>LIF</td>
<td>Discharge</td>
<td>Idle*</td>
<td>Idle</td>
</tr>
<tr>
<td>STDP</td>
<td>Idle</td>
<td>Weight update</td>
<td>Idle</td>
</tr>
<tr>
<td>Idle</td>
<td>Idle</td>
<td>Idle</td>
<td>Idle</td>
</tr>
</tbody>
</table>

![Chip image](image_url)
One-transistor/one-resistor (1T1R) synapse

Synapse $V_{TE}$ Costante
PRE-spike
PRE $V_G$ BE
POST
Integrate
Fire $V_{th}$
Post-spike computing – 6 Daniele Ielmini
Depression ($\Delta t < 0$)

Potentiation (Δt > 0)

STDP characteristics

Learning of handwritten digits (MNIST database) was simulated with a 2-layer network (N = 28x28)
On-line learning and update

Multiple learning neurons

Pattern

First layer
N neurons

Second layer
M neurons

PRE

1

2

3

N

POST

1

2

N

M
Learning multiple digits

[Image of input and multiple synapses showing different patterns for each synapse]
‘What I cannot create, I do not understand’
Stochastic spiking

- Pattern
- Background

Pattern

Epoch

Channel

0 1 2 3 4 5 6 7 8

0 1 2 3 4 5 6 7 8

Pattern
Stochastic spiking

- Noise
- Pattern

Graph:
- X-axis: Epoch
- Y-axis: Channel
- Data points: Noise and Pattern
Stochastic spiking
Stochastic spiking

Pattern

Noise

Channel vs. Epoch

Pattern

Noise
Stochastic spiking

Pattern

Noise

Post-Si computing – 6 Daniele Ielmini
Stochastic spiking

![Graph showing noise and pattern over epochs and channels.](image)
Stochastic spiking

Noise

Pattern

Channel

Epoch
Pattern learning from HRS synapses

Input

Weights

\( V_{\text{TH}} \)
Pattern learning from LRS synapses

Input

Weights

\[ V_{\text{TH}} \]
On-line learning of 3 sequential patterns

Input

Calculated Weights

Measured Weights

\[ V_{\text{int}} \text{ [V]} \]

\[ 0 \quad 0.5 \quad 1 \quad 1.5 \]

\[ 0 \quad 200 \quad 400 \quad 600 \quad 800 \quad 1000 \]

\[ 1/R \text{ [\Omega^{-1}]} \]

\[ 10^{-6} \quad 10^{-4} \]

\[ 0 \quad 200 \quad 400 \quad 600 \quad 800 \quad 1000 \]

Epoch
Gray-scale learning

Input  Calculated Weights  Measured Synapses

\[ V_{\text{int}} [\text{V}] \]

\[ \frac{1}{R} [\Omega^{-1}] \]

Epoch
Multiple-pattern learning and tracking
Spike-rate dependent plasticity

- LTP (LTD) for high (low) spike rate

V. Milo, et al., IEDM 2016
Conclusions

• Although neural networks have been studied for decades, only recently there has been sufficient momentum thanks to mature technology, solid understanding and emerging devices for nonvolatile synapses

• Memristive synapses enable nanoelectronic neuromorphic circuits with extremely high number of connections, e.g., crossbar architecture

• Larger hybrid synapses allow for low power communication and STDP, SRDP, fine tuning of synapse weights, and high parallelism