A Quick Overview of Memristor Research

Andrey Kurenkov
LSM Lab, EPFL

Leon Chua`s Predicted Memristor

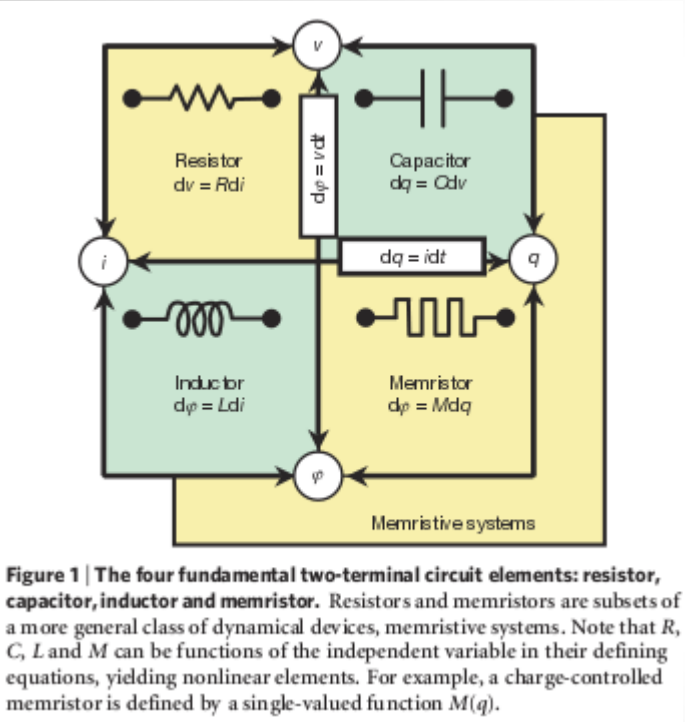


Figure 1 | The four fundamental two-terminal circuit elements: resistor, capacitor, inductor and memristor. Resistors and memristors are subsets of a more general class of dynamical devices, memristive systems. Note that R, C, L and M can be functions of the independent variable in their defining equations, yielding nonlinear elements. For example, a charge-controlled memristor is defined by a single-valued function $M(q)$.

Symmetry argument, from “The Missing Memristor Found”

In 1971`s “Memristor: The Missing Circuit Element,” Leon Chua made a symmetry argument for there needing to be a fourth fundamental circuit element that relates flux to charge. This is the last relationship between voltage, current, charge, and flux that has not yet been realized by the resistor, cap, and inductor. The resistance of such a device depended on past current through it, so it was dubbed the “memristor.”

In 1976`s “Memrisive Devices and Systems” he defined a broader class of memristive systems special for all having a hysteresis loop that had a zero crossing (f(0)=0).

a special case of a much more general class of dynamical systems--henceforth called *memristive systems*--defined by[1]

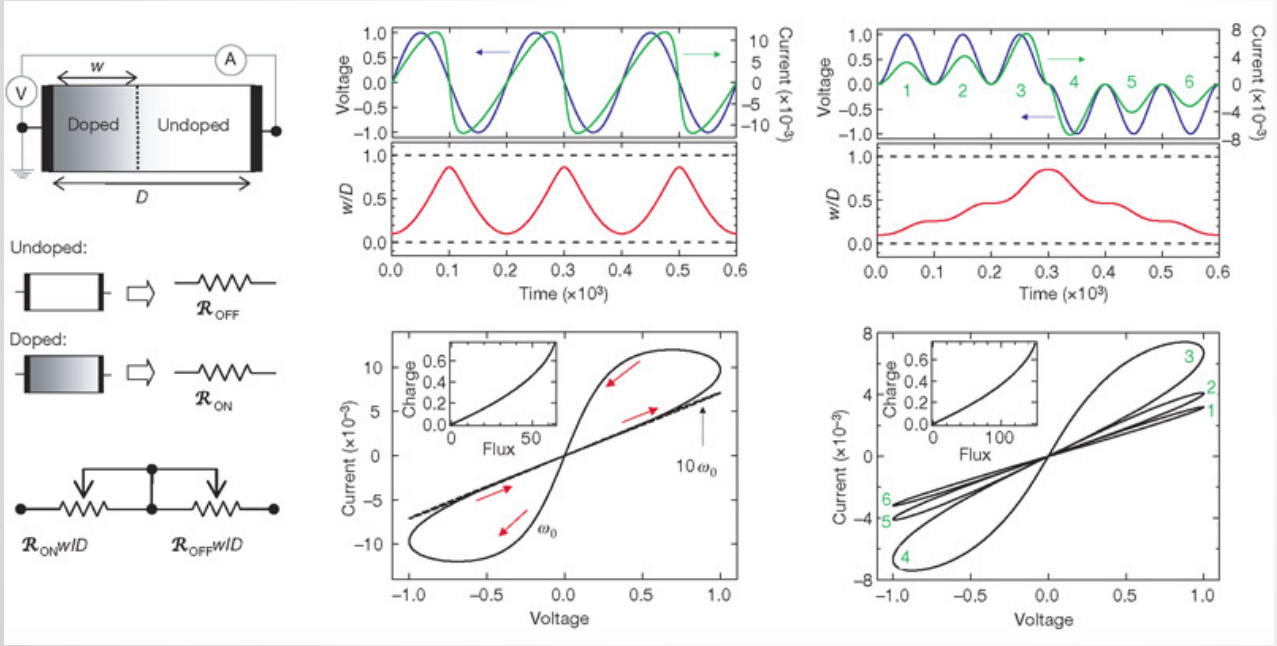
$$\dot{x} = f(x, u, t)$$
$$y = g(x, u, t)u \qquad (1)$$

where u and y denote the *input* and *output* of the system and x denotes the *state* of the system. The function $f: \mathbb{R}^n \times \mathbb{R} \times$

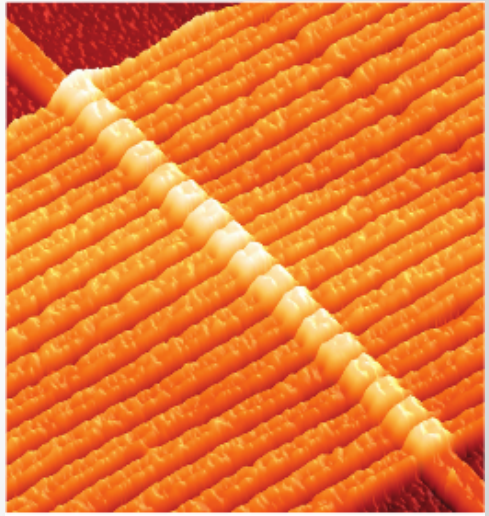
From “Memristive devices and systems”

Memristor Concept Revived in 2008

A 2008 letter titled “The Missing Memristor Found” from HP labs suggested a range of nano devices were actually modeled by the memristor concept.



Model and pinched hysteresis loop, from “Missing Memristor Found”



CROSSBAR ARCHITECTURE: A memristor's structure, shown here in a scanning tunneling microscope image, will enable dense, stable computer memories. IMAGE: R. STANLEY WILLIAMS/HP LABS

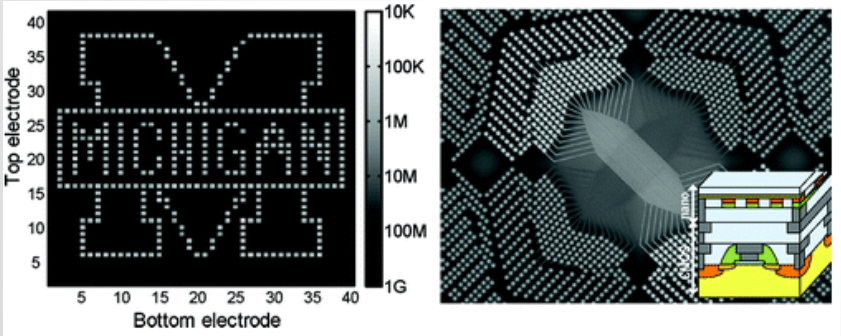
Crossbar, from “How We Found The Missing Memristor”

Flurry of Research Since 2008

As of recently, 2215 citations of “The Missing Memristor Found.” Notable applications include:

- ReRam (resistive RAM) Memory to replace Flash (near commercialization)
- Logic/Computation within memory
- Reconfigurable Circuits (improved FPGAs, PLAs, etc)
- Neuromorphic circuits

Much research on best way to model memristors, ranging from the simple original HP model to much more complex models. Wide range of physical realizations and behaviors.



2012, “A Functional Hybrid Memristor Crossbar-Array/CMOS System for Data Storage and Neuromorphic Applications”

For pedagogical reasons, any 2-terminal black box (figure 1(a)) with two electrical terminals is called a *memristor* if it obeys the following state-*dependent* Ohm’s law.

Current-controlled state-dependent Ohm’s Law:

$$v = R(\mathbf{x})\, i \qquad (1a)$$

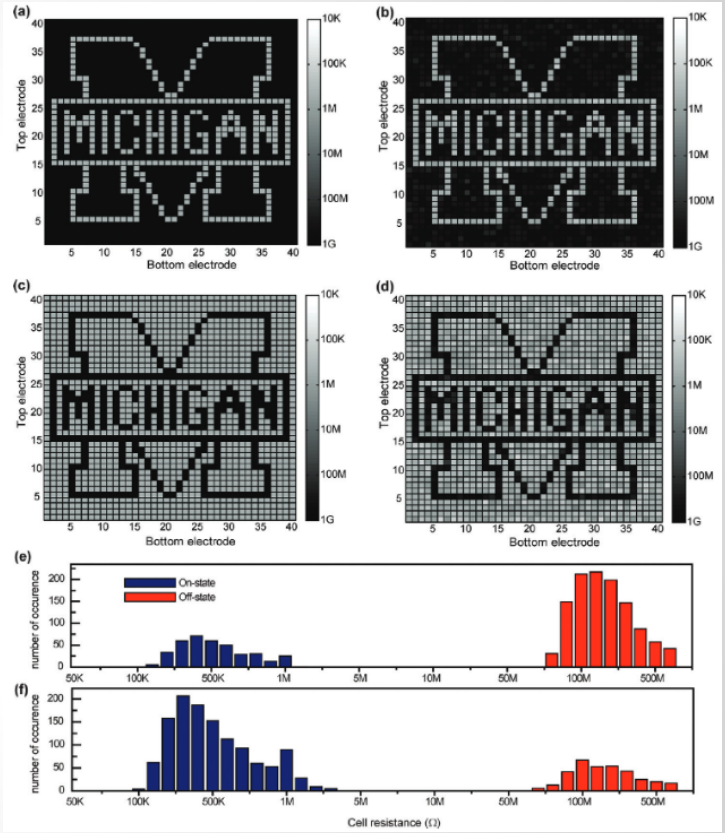
$$\text{State equations: } \frac{d\mathbf{x}}{dt} = f(\mathbf{x}, i) \qquad (1b)$$

if the current i is the input, or

Latest definition by Leon Chua, “Memristor, Hodgkin–Huxley, and Edge of Chaos“

Memristor for ReRam

Essentially use resistance rather than charge for storage in memory. Crossbar allows for very dense and power efficient storage, though read times only make it competitive with FLASH.



2012, “A Functional Hybrid Memristor Crossbar-Array/CMOS
System for Data Storage and Neuromorphic Applications”

Memristor for ReRam (cont.)

A much studied problem for crossbars is “sneak path” currents. Solutions include additional components, two opposite polarity memristor as “Complementary Resistive Switches “ and using devices that are insulating at negative bias. Non binary memory storage has been demonstrated using the last approach.

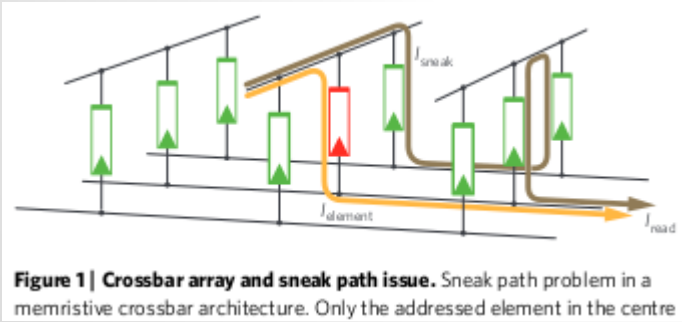
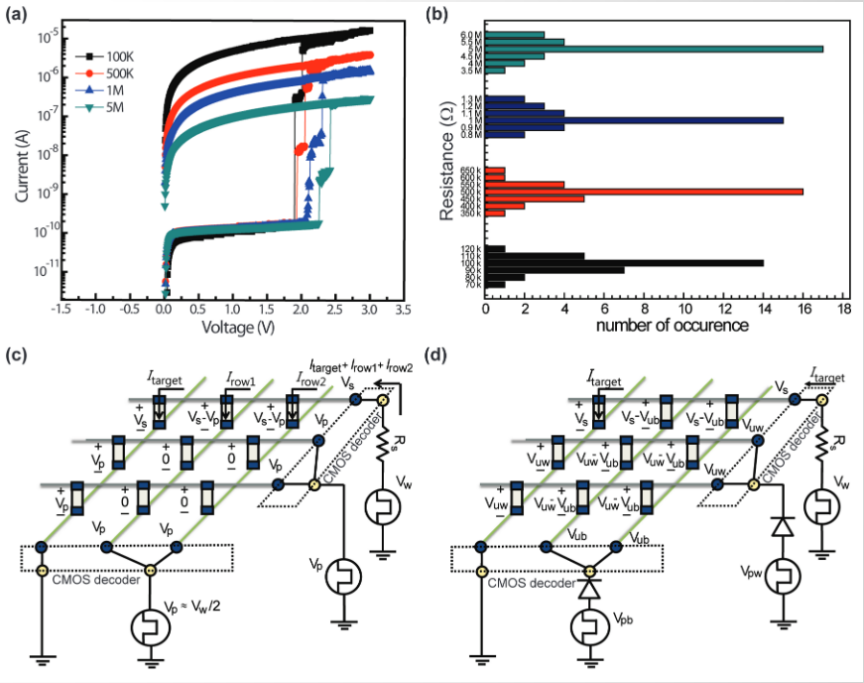


Figure 1 | Crossbar array and sneak path issue. Sneak path problem in a memristive crossbar architecture. Only the addressed element in the centre

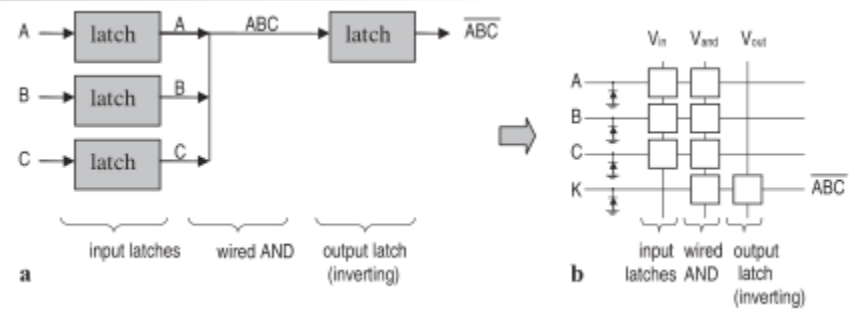
2010, “Complementary Resistive Switches for passive nanocrossbar memories”



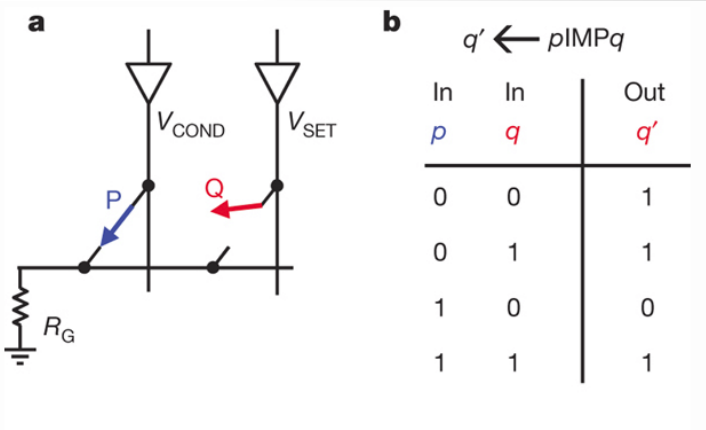
2012, “A Functional Hybrid Memristor Crossbar-Array/CMOS
System for Data Storage and Neuromorphic Applications”

Logic Within Memory

Research into logic in nano crossbars as early as 2005, led by HP`s Gregory S. Snider. Limited by need for closed junctions and device variability.



2005, “Computing with hysteretic resistor crossbars”

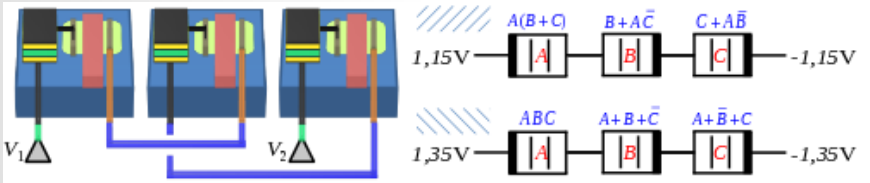


2010, “‘Memristive’ switches enable ‘stateful’ logic operations via material implication”

Logic Within Memory (cont.)

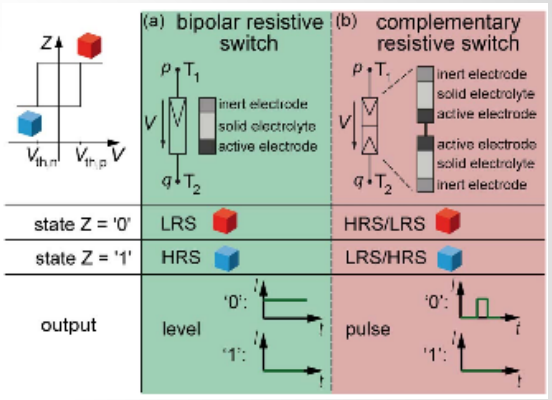
Many more suggestions, all showing universal boolean logic. No large scale systems /architectures proposed for computing.

Several papers on analog or digital arithmetic.

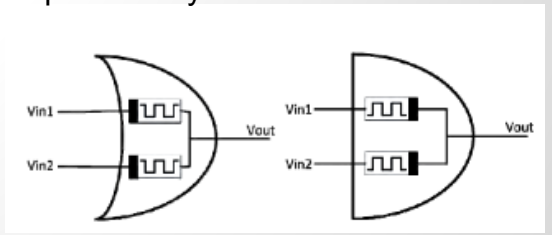


Registry		1	A	B
1st operation level	operation	$A+B$	$A+\bar{B}$	$\bar{A}+B$
	W/R	W(1)	REFRESH	REFRESH
2nd operation level	operation	$(A+B)(\bar{A}+B)$	1	1
	W/R	REFRESH	Nothing	Nothing

2012, “Dynamic Computing Random Access Memory“

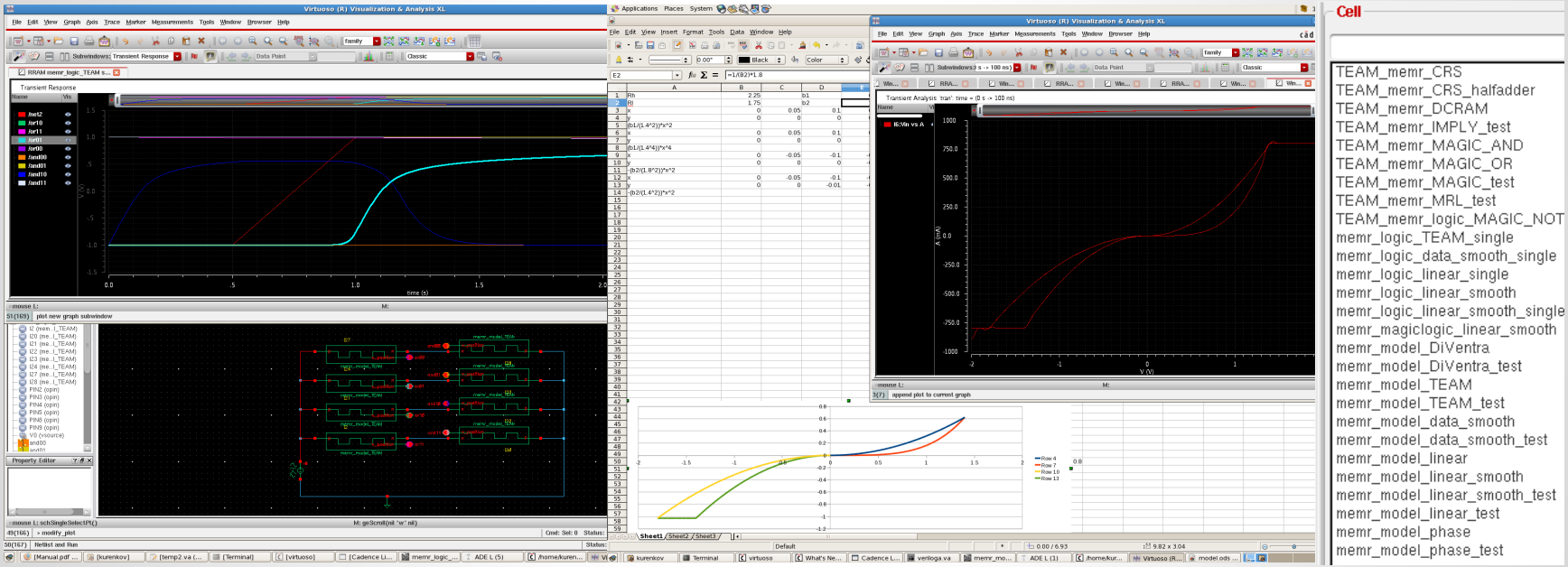


2011, “Crossbar Logic Using Bipolar and Complementary Resistive Switches”



2012, “MRL – Memristor Ratioed Logic”

VerilogA/Schematic Work With Logic



Simulation with Cadence was done to verify and evaluate different schemes for performing logic. Operation of logic is verified, though working logic is not nearly enough for general computation and scaling up from the basic boolean gates is hugely non-trivial.

Universal Memcomputing Machines

Turing Machines are a theoretic model of the Von Neuman Architecure, UMMs proposed by Fabio L. Traversa and Massimiliano Di Ventra as theoretical model for computing in memory.

Each memory element also computes.

Can solve NP-complete problems in O(n) time.

$$UMM = (M, \Delta, \mathcal{P}, S, \Sigma, p_0, s_0, F),$$

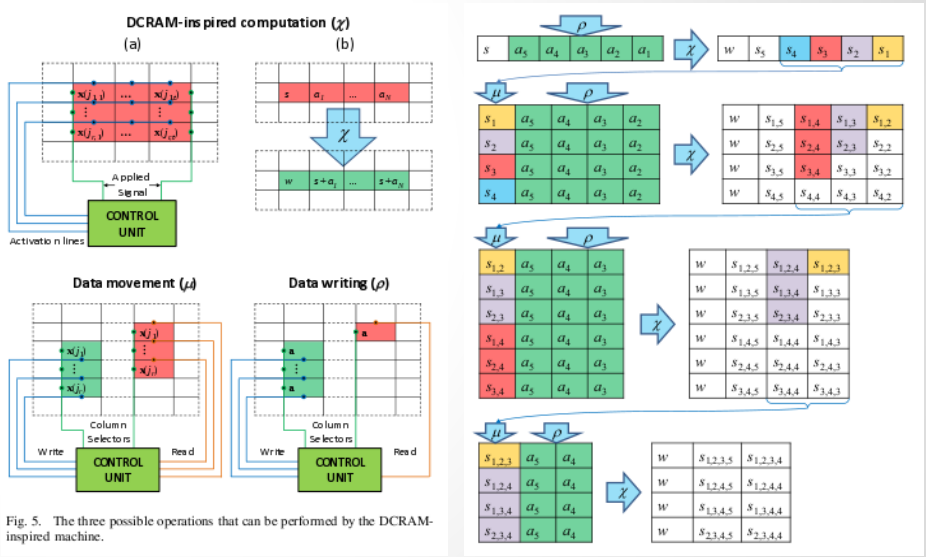


Fig. 5. The three possible operations that can be performed by the DCRAM-inspired machine.

Theoretical solution to subset sum in O(n), 2013, “Universal Memcomputing Machines“

Memprocessor for Memcomputing

Completely different model from Von Neuman, but not actually radical.

Doing computation with storage elements does not inherently change computing, if the computation is still localized and digital.

A control unit with many memprocessors is very similar to distributed computing/systolic arrays. This is MIMD (multiple instruction, multiple data) parallelism, and it is clear why it allows for O (n) solutions to NP-complete problems.

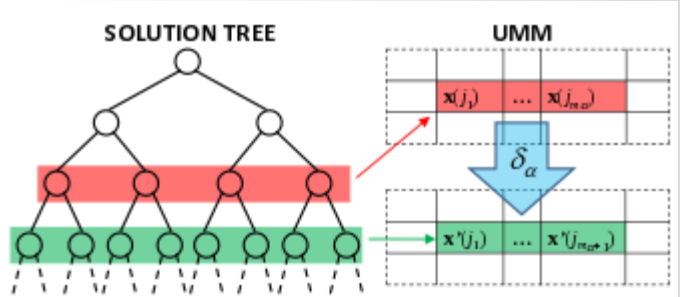


Fig. 2. Left panel: a typical solution tree for an NP–complete problem. Right panel: the sketch of the UMM transition function to solve it.

Solving NP-Complete problem with parallelism, 2013, “Universal Memcomputing Machines”

Tangent: Criticism of Memristors

Some criticisms of both memristor theory and some of the excitement over memristors.

“"The conclusion is: ***Put the computation near the data.”***

Is this anything new? Generations of systems designers and engineers have given their best to achieve this ideal.

…

But again, the fundamental difference of computation (operator) and data (operands) is kept alive and is determining the rest of the game.”

-Why Memristors Wont Change Anything

-The Mythical Memristor

Computing in Neural Nets

On the other hand, analog, learning-based, truly-distributed computation/storage in memory is fundamentally different from UTM computing.

Memristors are perfect for synapses due to high density, continuous value, and inherent STDP.

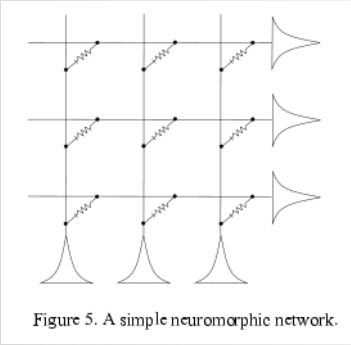
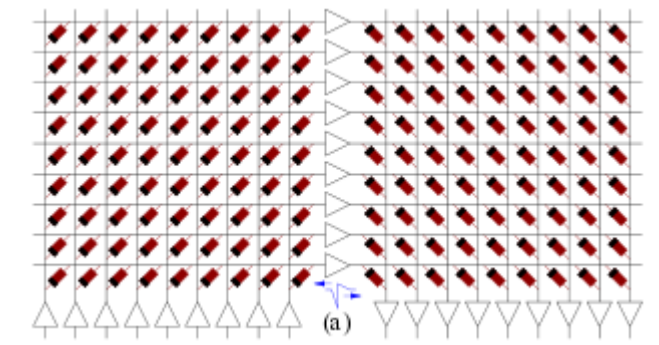


Figure 5. A simple neuromorphic network.

2008,“Spike-Timing Dependent Learning in Memristive Nanodevices”



2009, “Exploiting memristance in adaptive asynchronous spiking neuromorphic nanotechnology systems“

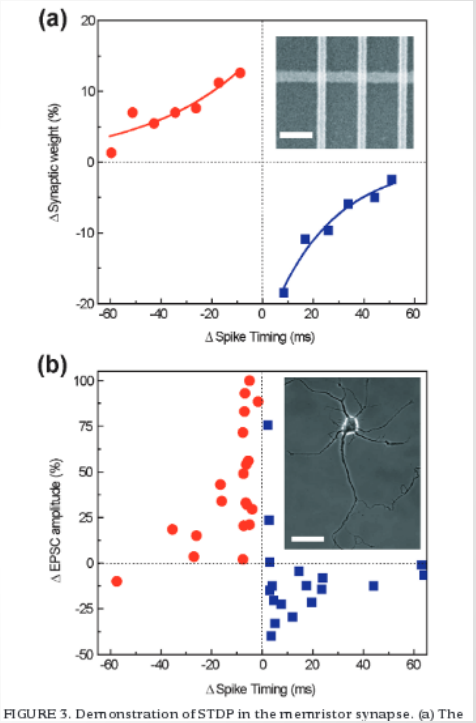
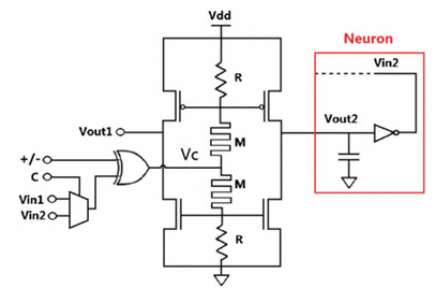


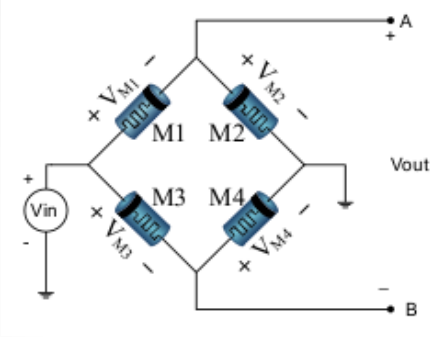
FIGURE 3. Demonstration of STDP in the memristor synapse. (a) The

2010, “Nanoscale Memristor Device as Synapse in Neuromorphic Systems“ (500+ citations)

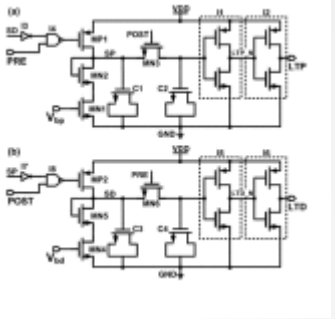
Many Synapse Designs



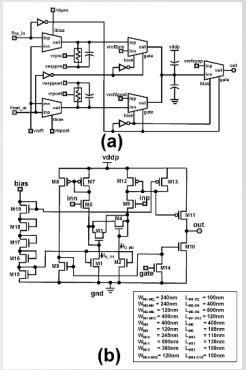
2013, “Reconfigurable Neuromorphic Computing System with Memristor-Based Synapse Design”



2012, “Memristor Bridge Synapses”



2013, “Design of an electronic synapse with spike time dependent plasticity based on resistive memory device”



2012, “Energy-Efficient Neuron, Synapse and STDP Integrated Circuits”

Can be memristors in crossbar, but many are more complicated due to actual complexity of synapses and problems in fabrication. The spike-time dependent plasticity (STDP) unsupervised learning mechanism is the primary focus due to biological link and ease of implementation.

Even More Complex: Neuron Design

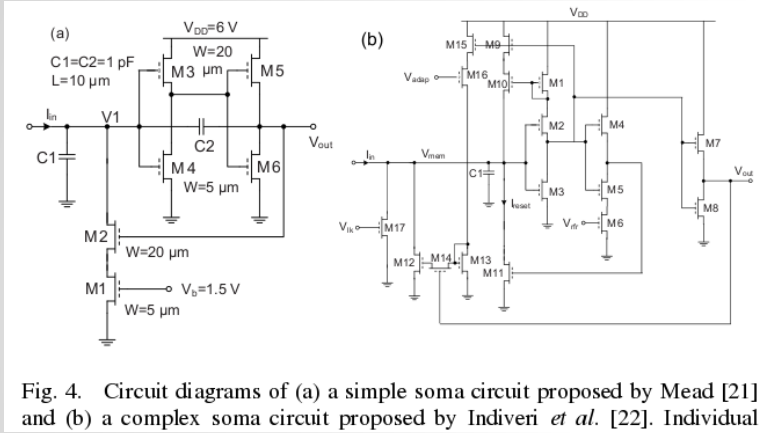
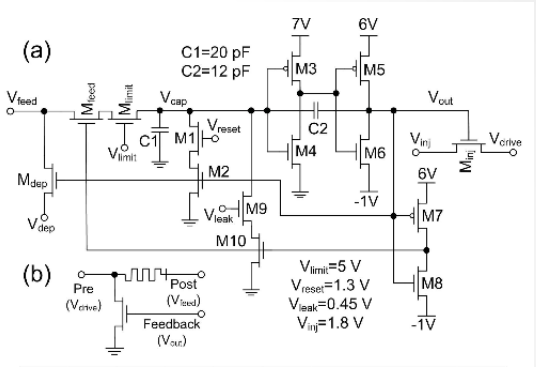
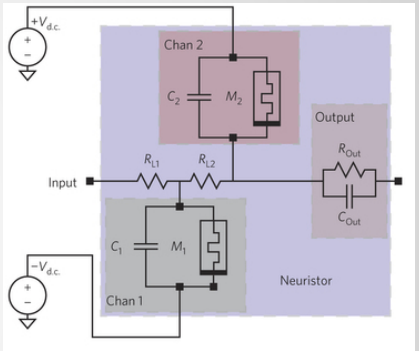


Fig. 4. Circuit diagrams of (a) a simple soma circuit proposed by Mead [21] and (b) a complex soma circuit proposed by Indiveri *et al.* [22]. Individual

Older designs, 2012, “Low-Temperature Fabrication of Spiking Soma Circuits Using Nanocrystalline-Silicon TFTs“



2012, “Neural Learning Circuits Utilizing Nano-Crystalline Silicon Transistors and Memristors”



2012, “A scalable neuristor built with Mott memristors“

Much larger than synapses, but there also need to be far fewer. Human brain has 10^14 synapses and 10^10 neurons, so neurons can be in the micrometer range if synapses are in the nanometer range. Address Event Representation (AER) is commonly used to implement large amount of connections between neurons. However, issues with power and overall scalability are not yet solved.

Simulation Results

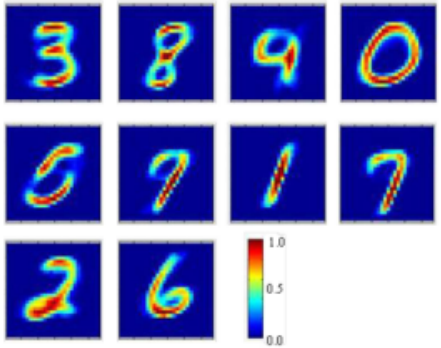


Fig. 3. Weights (conductances) learned in a simulation with 10 output neurons. Red is maximum weight, blue is minimum weight.

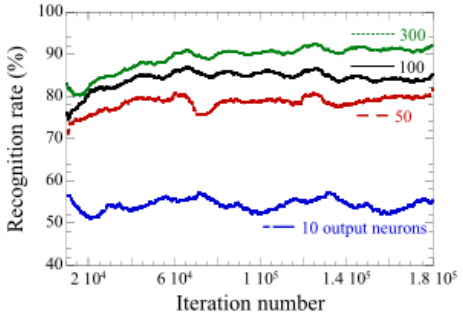


Fig. 4. Recognition rate during learning for simulations with different numbers of output neurons (from bottom to top: 10, 50, 100, 300). The recognition rates are running averages on 10,000 iterations.

2011, “Simulation of a memristor-based spiking neural network immune to device variations”

Small and large crossbars tested in simulation and reveal promising results. DARPA SyNAPSE project leading with the largest architectures, though other scalable systems are proposed.

Fig. 14. Sample of training images each of size 24 × 18 pixels. (a) Positive training images of rear of car. (b) Negative training images.

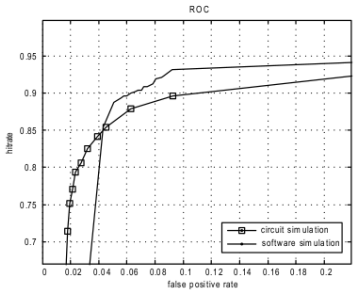


Fig. 15. ROCs curve of the initial software-trained network (software simulaion) and the proposed hardware-trained network (circuit simulation).

2012, “Memristor Bridge Synapse-Based Neural Network and Its Learning“ (multi-layer, trained on a computer)

Possibility of supervised learning

Most research focused on STDP, but newer papers are suggesting more complete designs for multi-layer neural networks. Multi-layer STDP has also been demonstrated for learning more complicated functions such as movement comprehension.

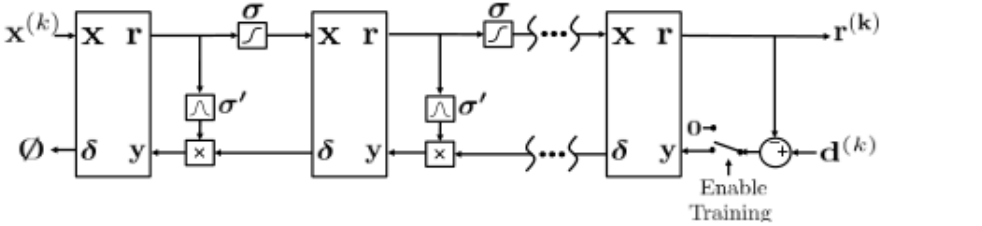


Fig. 4 Implementing the Backpropagation for a MNN with MSE error, using a modified version of the original circuit. The function boxes denote the operation of either $\sigma(\cdot)$ or $\sigma'(\cdot)$ on the input, and the $\times$ box denotes a component-wise product. For detailed circuit schematics, see [31].

2014, “Memristor-based multilayer neural networks with online gradient descent training“

Prospective Work

Currently, working to simply enable simulation of neural net learning with existing synapse and neuron designs.

In the longer term, a novel topic needs to be focused on. A possible approach to this is to research how more recent Machine Learning concepts can be incorporated into existing ideas, as none of the papers exploring neuromorphic architectures do this. Another possibility is to research novel ways to incorporate neural nets into conventional computers, such as learned branch prediction. Finally, my work could be more directly connected to past publications of this lab.