Learning in Neural Networks

Wulfram Gerstner

EPFL, Lausanne, Switzerland

from brain-style computing to neuromorphic computing

Objectives for today:

- review three-factor rules/brain-style computing
- local learning rules for hardware
- neuromorphic chips
- alternative computing hardware
- reducing energy consumption

Recent Development (IBM and INTEL):
Chip companies invest in neuromorphic

Potential reduction of energy consumption with SNN and three-factor rules.

Background reading:

University of Zurich/ETHZ Institute of Neuroinformatics.
Rubino∗, Cartiglia∗, Payvand, Indiveri, Neuromorphic analog circuits for robust on-chip always-on learning in spiking neural networks (2023)

https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=10168620

University of Heidelberg: Schmitt et al. Neuromorphic Hardware in the loop (2017),
https://ieeexplore.ieee.org/abstract/document/7966125

IBM research lab
Bert Offrein et al., 2020, Prospects for photonic implementations of neuromorphic devices and systems, *IEEE Xplore,*
https://ieeexplore.ieee.org/abstract/document/9371915

LOIHI Chip (intel)
https://en.wikichip.org/wiki/intel/loihi
https://download.intel.com/newsroom/2021/new-technologies/neuromorphic-computing-loihi-2-brief.pdf

EPFL
A.Momemi, B. Rahmani, M. Mallejac, Ph del Hougne and Romain Fleury, Backpropagation-free training of deep physical neural networks, *Science* (2025) 383:1297–1303

https://www.science.org/doi/10.1126/science.adi8474

Our computers work well, where is the problem?
Machine learning gets better, where is the problem?

Energy consumption!

- Deep Learning/LLM burns huge amounts of energy
- Big companies buy (or build) power plants
- Inference (usage after training) = 90 percent of energy
- training GPT3 costs 500 000kg Co2 equivalent emission

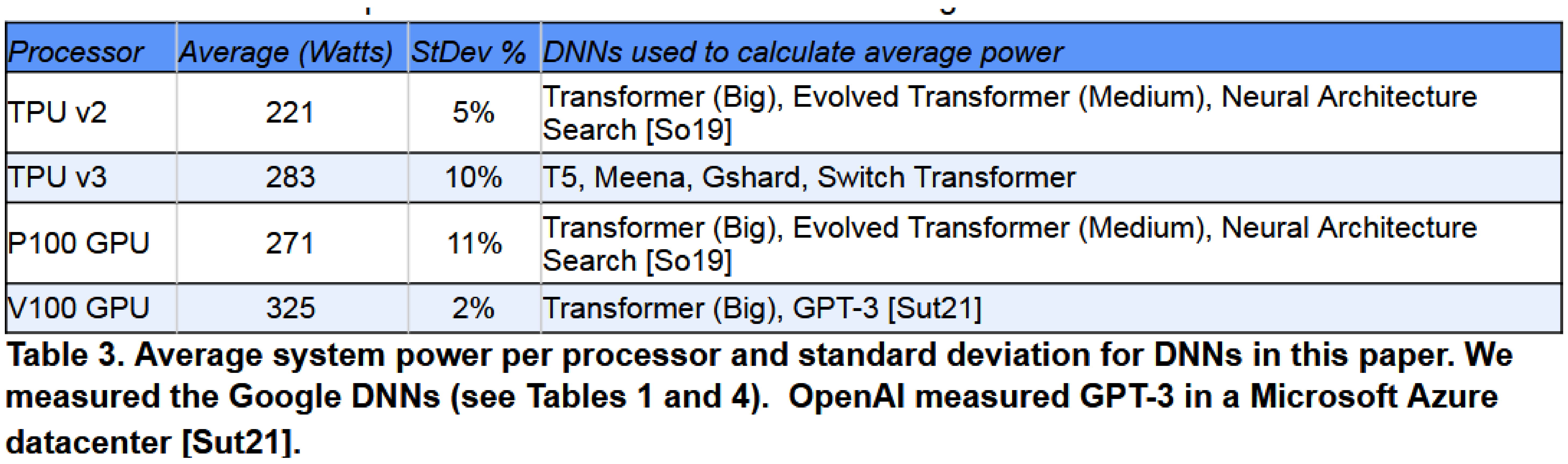
BUT:

Brain uses 24 – 30 Watt   (5 modern light bulbs)

Carbon Emissions and Large Neural Network Training
D. Patterson, et al. (2021) https://arxiv.org/abs/2104.10350

(previous slide)

Carbon Emissions and Large Neural Network Training (2021)
David Patterson, Joseph Gonzalez, Quoc Le, Chen Liang, Lluis-Miquel Munguia, Daniel Rothchild, David So, Maud Texier, Jeff Dean
https://arxiv.org/abs/2104.10350



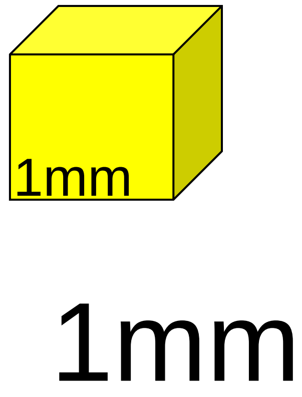
Processor	*Average (Watts)*	*StDev %*	*DNNs used to calculate average power*
TPU v2	221	5%	Transformer (Big), Evolved Transformer (Medium), Neural Architecture Search [So19]
TPU v3	283	10%	T5, Meena, Gshard, Switch Transformer
P100 GPU	271	11%	Transformer (Big), Evolved Transformer (Medium), Neural Architecture Search [So19]
V100 GPU	325	2%	Transformer (Big), GPT-3 [Sut21]

Table 3. Average system power per processor and standard deviation for DNNs in this paper. We measured the Google DNNs (see Tables 1 and 4). OpenAI measured GPT-3 in a Microsoft Azure datacenter [Sut21].

Review: Neurons and Synapses form a big network

brain

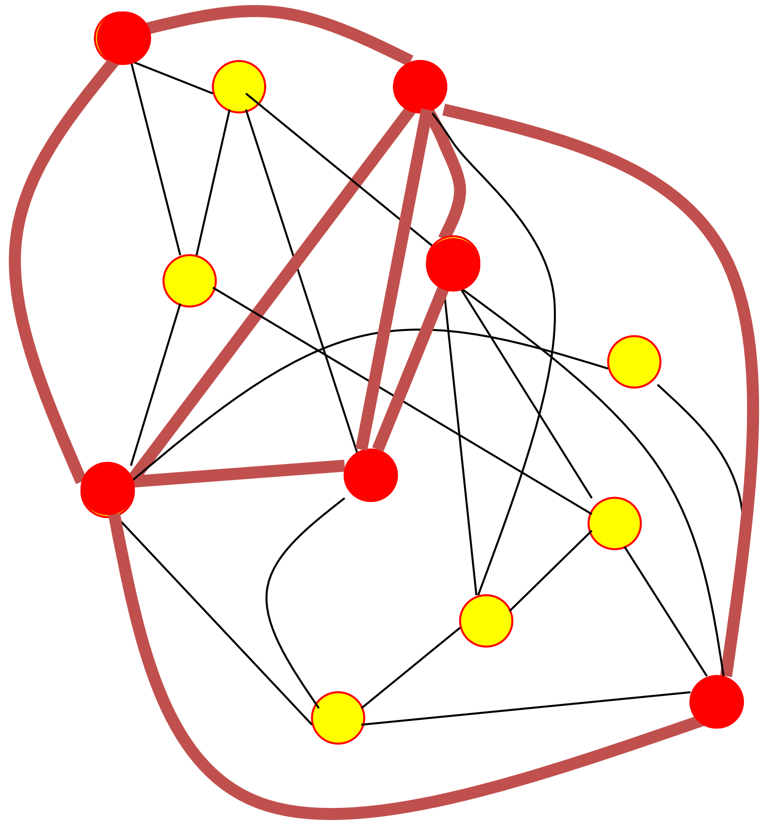
non-von-Neumann computing &hardware ‘brain-style computing’



10 000 neurons
3 km of wire

10 billions neurons

10 000 connexions/neurons



Distributed Architecture

memory in the connections

No separation of processing and memory

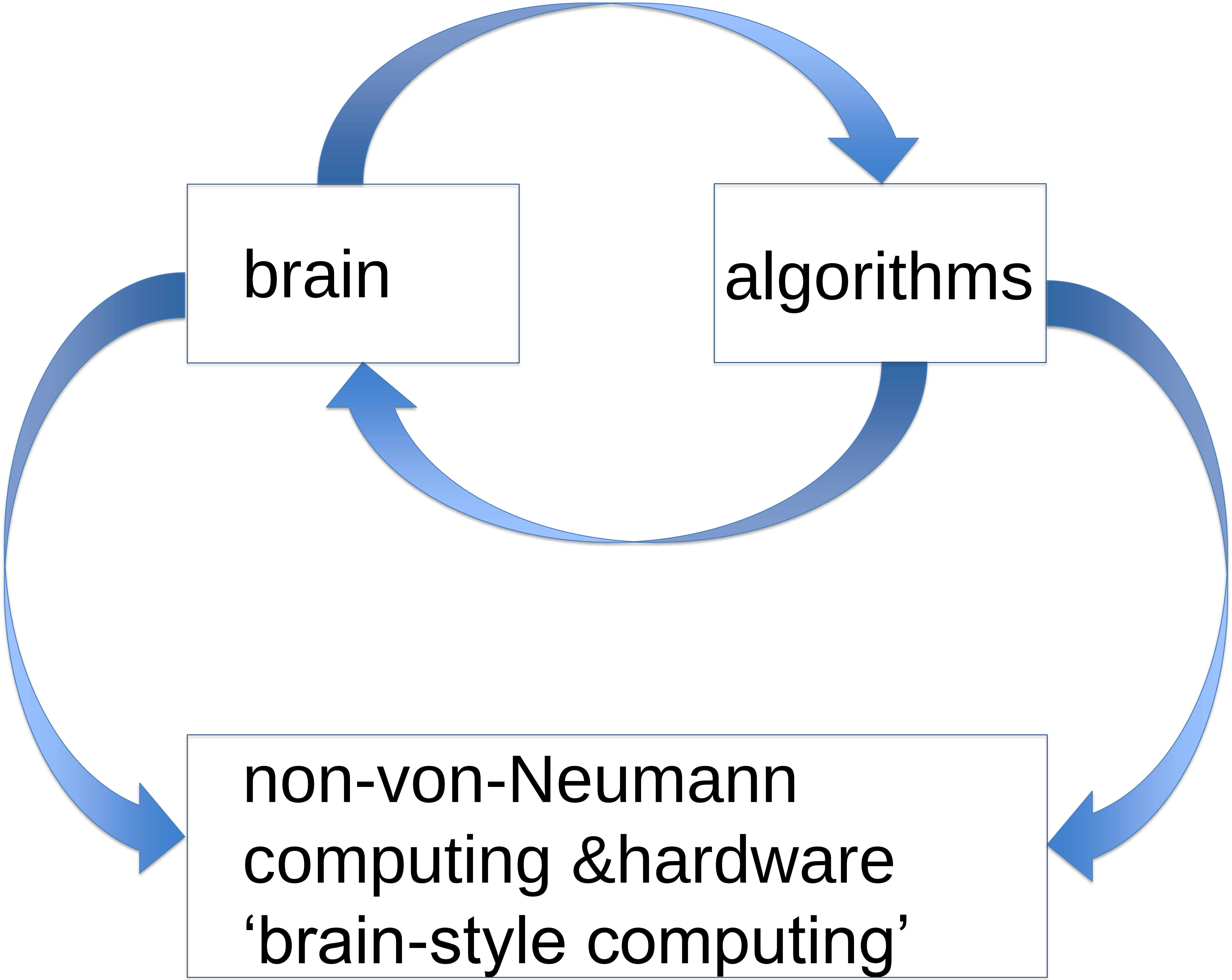
Previous slide. Review from previous lectures.

In the first lecture it was mentioned that the brain is radically different from the classical von-Neumann architecture that lead to our standard compute devises.

Particularly important differences are that the brain-style computing architecture is completely distributed, without centralized clock, no centralized controller and no separation of computing and memory.

We take in the following the learning rules of RL as a starting point of what this means and mention at the end novel hardware.

Review: Learning Rules of Reinforcement Learning



brain

algorithms

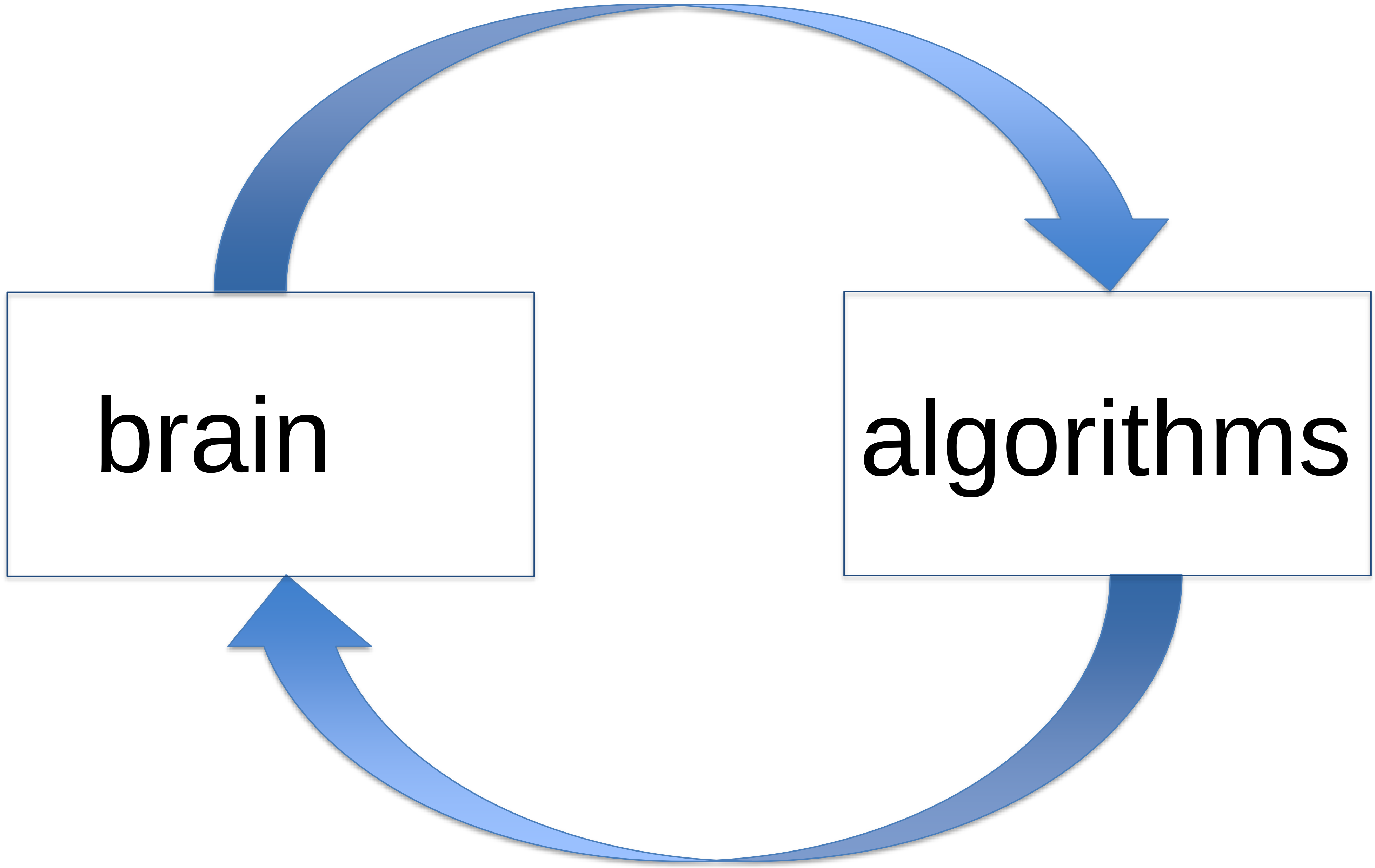
non-von-Neumann
computing &hardware
‘brain-style computing’

Previous slide. Review from previous lectures.

RL has two roots: optimization and Markov Decision Problems and Brain sciences.

We take in the following the learning rules of RL to see how they lead to alternative computing paradigms.

Review: Three-factor Learning Rules of Reinforcement Learning



Update of all eligibility traces

$$z_{lk} \leftarrow z_{lk}\lambda_z$$

$$z_{lk} \leftarrow z_{lk} + \frac{d}{dw_{lk}}\ln[\pi(a|s, wl_k)]$$

Change of **all** weights

$$\boxed{\Delta w_{lk} = \eta \ r_t \ z_{lk}}$$

$$\boxed{\Delta w_{lk} = \eta \ \delta_t \ z_{lk}}$$

The learning rule of the (advantage) actor-critic or REINFORCE with eligibility traces are both compatible with three-factor rules

Updates proportional to the reward r or TD error δ_t

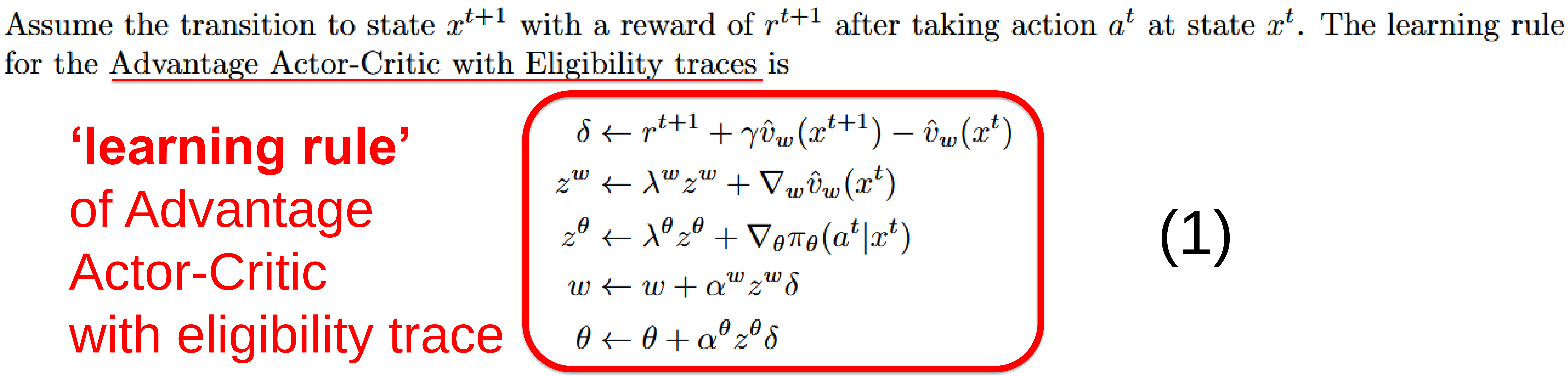
Previous slides.
Review of algorithm with actor-critic architecture with eligibility traces.

The Advantage actor critic has parameter updates proportional to the TD error.

Reinforce/policy gradient has updates proportional to the momentary reward.

Apart from this difference, the overall structure of the two algorithms is very similar.

Review: Learning Rules of Reinforcement Learning



Assume the transition to state x^{t+1} with a reward of r^{t+1} after taking action a^t at state x^t. The learning rule for the Advantage Actor-Critic with Eligibility traces is

'learning rule' of Advantage Actor-Critic with eligibility trace

$$\begin{aligned}
\delta &\leftarrow r^{t+1} + \gamma \hat{v}_w(x^{t+1}) - \hat{v}_w(x^t) \\
z^w &\leftarrow \lambda^w z^w + \nabla_w \hat{v}_w(x^t) \\
z^\theta &\leftarrow \lambda^\theta z^\theta + \nabla_\theta \pi_\theta(a^t|x^t) \\
w &\leftarrow w + \alpha^w z^w \delta \\
\theta &\leftarrow \theta + \alpha^\theta z^\theta \delta
\end{aligned} \qquad (1)$$

→ Learning rules of other ONLINE RL policy gradient models are special cases of (1).

→ We take (1) as a starting point to discuss the relation with the brain and with hardware

Can such a **learning rule** be implemented in the brain?

Can such a **learning rule** be implemented in hardware?

Previous slide. Review from previous lecture.

In the following we take the Advantage Actor Critic as our Reference Model.

As we have seen earlier, other Algorithms in the Family of Policy Gradients can be identified as special cases.

Last week we have seen how such a learning rule (update algorithm) be implemented in the brain.

In this lecture we ask: how could an implementation of the actor-critic look like in the brain? And in hardware?

Review: CLAPP Loss

Illing, et al. NeurIPS 2021

$$L_{CLAPP}^{t,l} = \max(0, 1 - y^t \cdot u_t^{t+\delta t,l})$$

$u_t^{t+\delta t,l}$ = similarity:

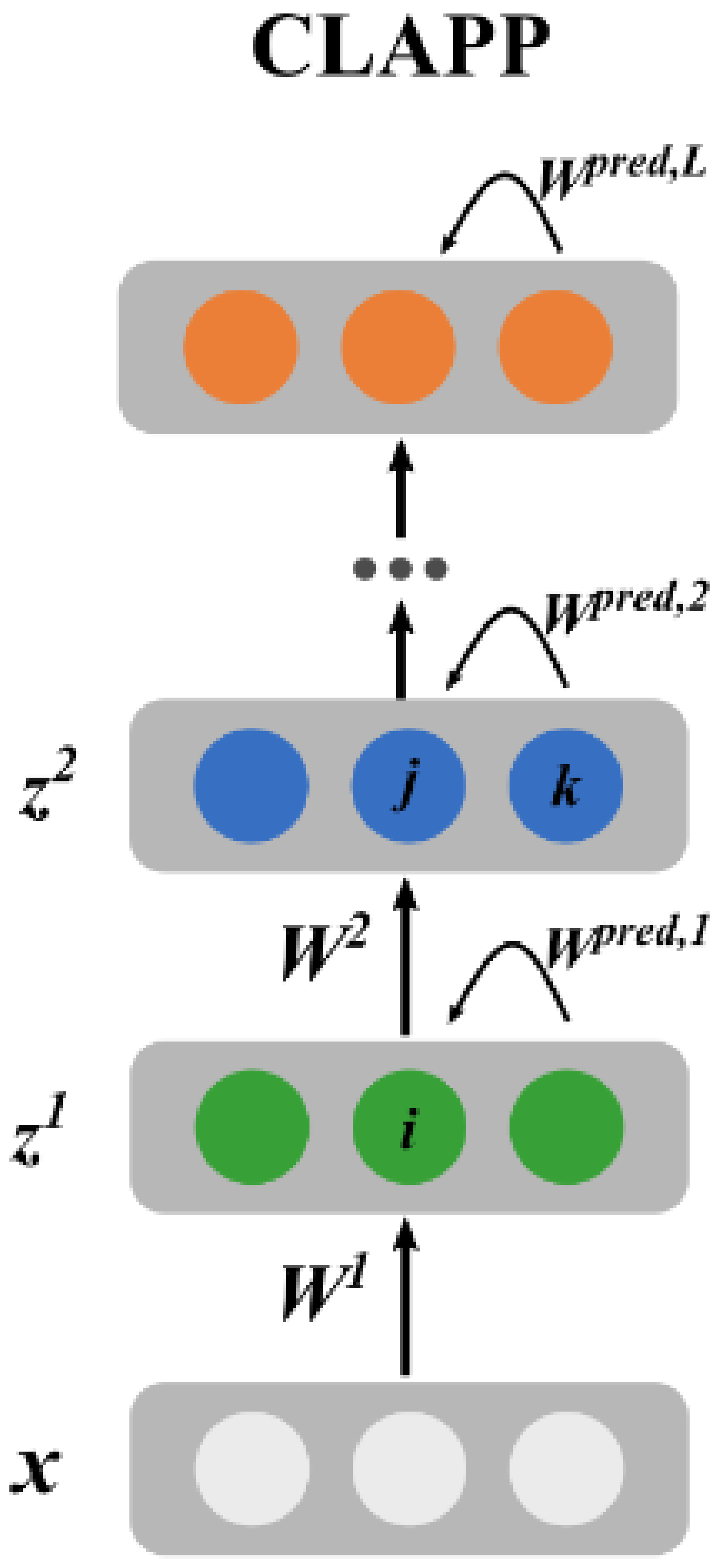
$$u_t^{t+\delta t,l} = \mathbf{z}^{t+\delta t,l}\underbrace{W^{pred,l}\mathbf{c}^{t,l}}$$

feedforward vs lateral prediction

y^t = sameness signal/contrastive signal

$$y^t = \begin{array}{c} 1 \;\; if\ same\ sample \\ -1 \; if\ negative\ sample \end{array}$$

→ Biologically plausible local learning rule



Can such a **learning rule** be implemented in hardware?

Previous slide.
The small index l is the layer index. The last layer is l=L.

Let us look at the loss in the last layer. It is a hinge-loss (picture on the next slide): either zero or linear in u.

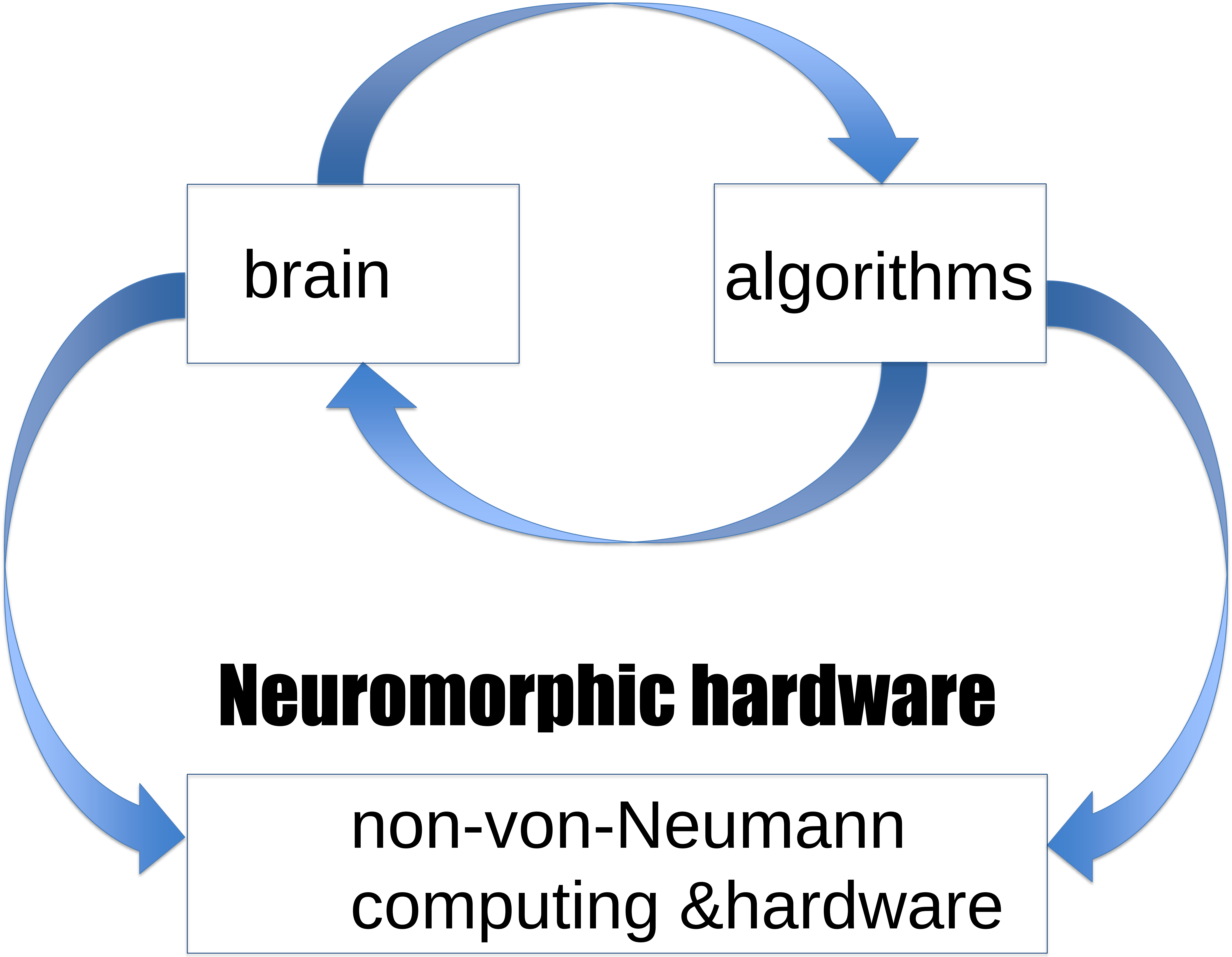
The variable u is a measure of the similarity between the activity state vector **z** in layer l and the lateral prediction from OTHER neurons **c = z** in the same layer.

If variable y tells whether the prediction comes from the SAME object (y=1) or a different object (y=-1).

The boldface **z** refers to all neuron in a layer. For an interpretation it is easier to look at individual neurons such as neuron i in layer l. Its activity depends ONLY on the feedforward pathway

$$z_i^{t+\delta t,l} = g(\sum_j w_{ij}^l \, z_j^{t,l-1})$$

Local Learning Rules, Spiking Neurons,



brain

algorithms

Neuromorphic hardware

non-von-Neumann
computing &hardware

Previous slide:
The lecture last week covered the relation between learning rules used by the brain and those implemented in modern reinforcement learning algorithms.
This lecture will make the link to recent developments in modern neuromorphic computing architectures that are completely different than the class model of von-Neumann computing architectures.

One aspect is that these hardware approaches explore potential advantages of Spiking Neural Networks.

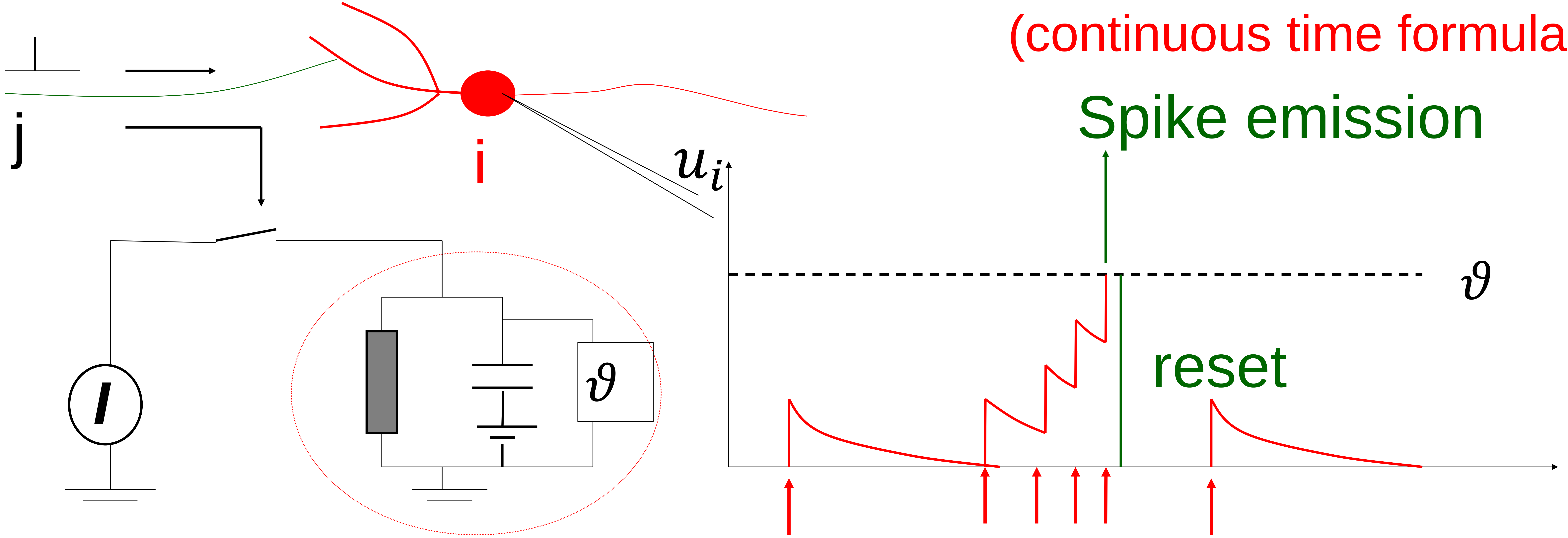
Another aspect is that they rely on local learning rules, in particular three-factor rules.

A third aspect is that they could potentially reduce energy consumption.

This lecture today provides an outlook onto current developments for specialized, bio-inspired chips that will eventually use much less energy than conventional chips. The category of chips is often called neuromorphic chips since they take inspiration from biological principles in neuroscience.
In particular, they use communication with spiking neurons and local learning rules.

Review: Spiking Neurons – Leaky Integrate-and-Fire Model

(continuous time formulation)



$\mathrm{I_i(t)} = short\ pulses = \sum_j w_{ij}\ \delta(t - t_j^{pre})$

$$\tau_m \frac{d}{dt} u_i = -u_i(t) + I_i(t)$$ linear

if $u_i(t) = \vartheta$ note spike+reset to $u(t) = 0$ threshold

Previous slide:

The Leaky integrate-and-fire model written in **continuous time** involves a LINEAR differential equation that can be interpreted as an electrical RC circuit charged by a current *I(t)*. Such a circuit is straightforward to implement in ANALOG hardware!
The current *I(t)* consists here of short electrical pulses that present spike arrivals. The $\delta(t - t_j^{pre})$ denotes the Dirac delta function for each presynaptic spike arrival at times t_j^{pre} and w_{ij} are the weights. We can interpret w_{ij} as the charge delivered by the current pulse at time t_j^{pre}. Again, this is easy to implement in analog hardware.

The linear equation is combined with a NONLINEAR FIRE-and-RESET process. If the variable u ('membrane potential of the neuron') reaches the threshold theta, then u is reset to zero. Easy to implement.

Side Note: An electrical RC circuit consists of a capacitance C and a resistor R and has a time constant $\tau = RC$. Therefore after each short current pulse, the voltage (membrane potential) decays exponentially back to zero with time constant $\tau = RC$.

The time constant of modern hardware is 10'000 times shorter than that of biology! Three years of human learning time translates into 2 hours of training neuromorphic hardware.

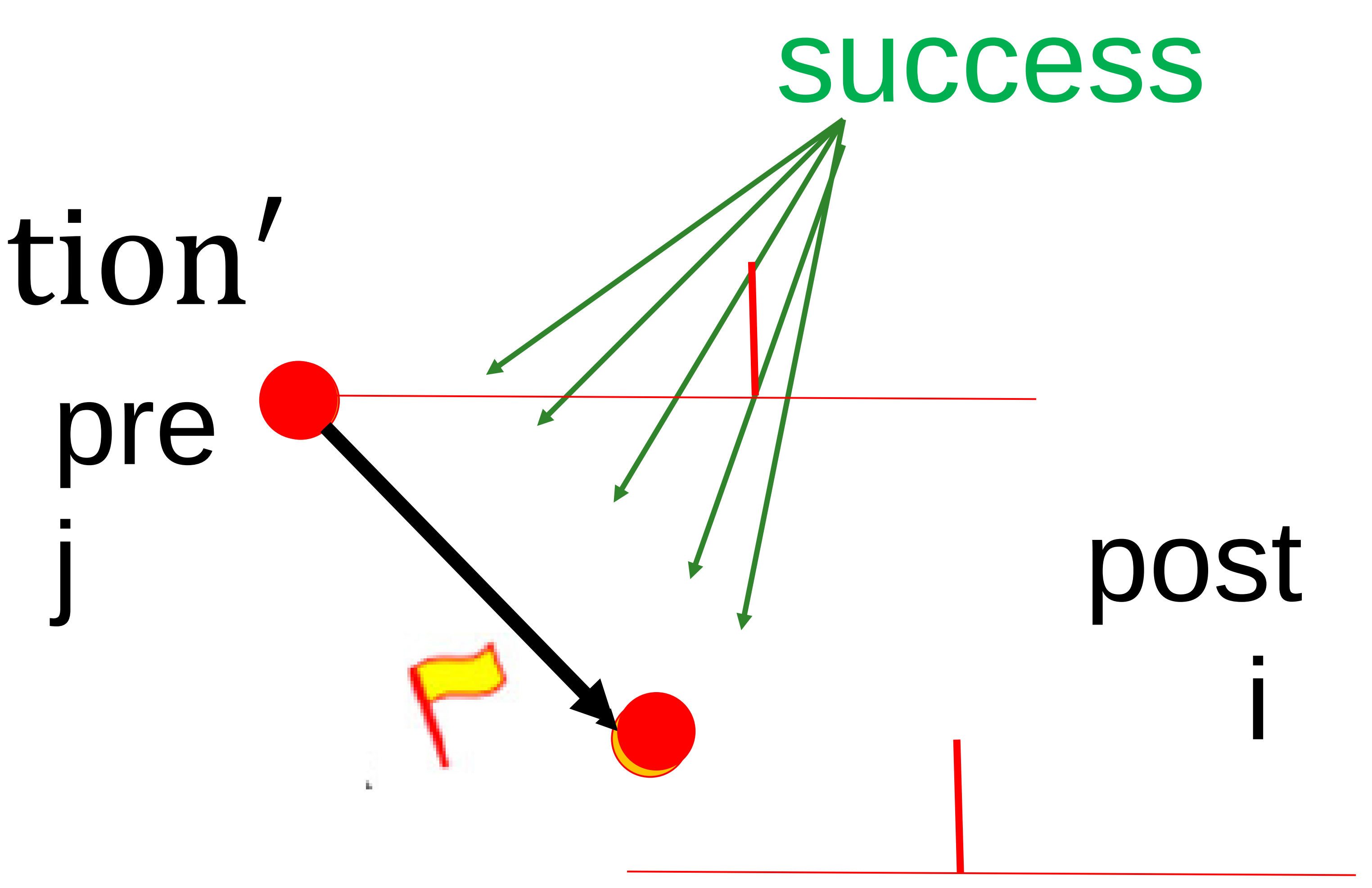
Review: Coincidence detection rule of STDP

$$\Delta w_{ij} \propto F(pre, post, SUCCESS)$$

Success signal:
TD error

Eligibility trace:

$$\Delta z_{ij} > 0 \ \ if \ \ '\mathrm{STDP-condition}'$$



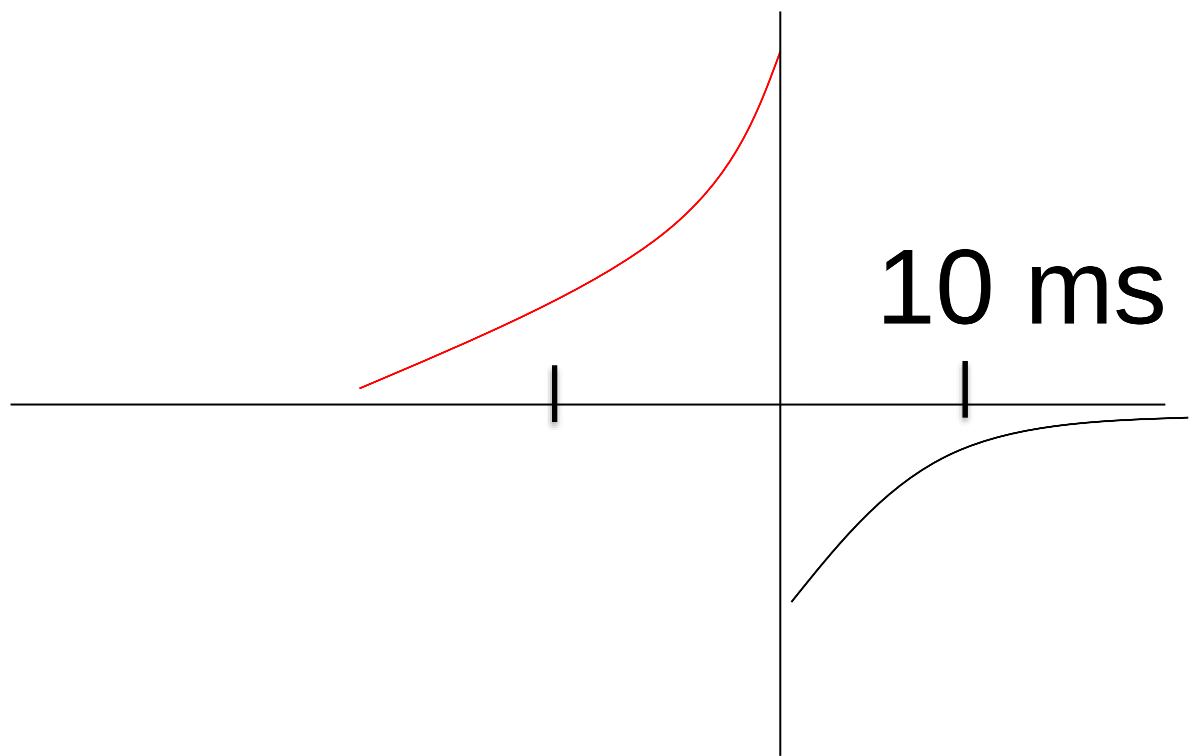
Weight

$$\Delta w_{ij} = z_{ij}\, S$$

Success signal

Hebb rule/eligibility trace

STDP condition



Xie and Seung 2003, Izhikevich, 2007; Florian, 2007; Legenstein et al., 2008, Fremaux et al. 2010, 2013

Previous slide:

A specific biologically plausible three-factor rule with eligibility traces would be the following:

- Spike-Timing-Dependent Plasticity (STDP) picks up coincidences between pre and postsynaptic spikes on a time scale of 10 milliseconds. STDP is hence a spike-based version of Hebbian learning.

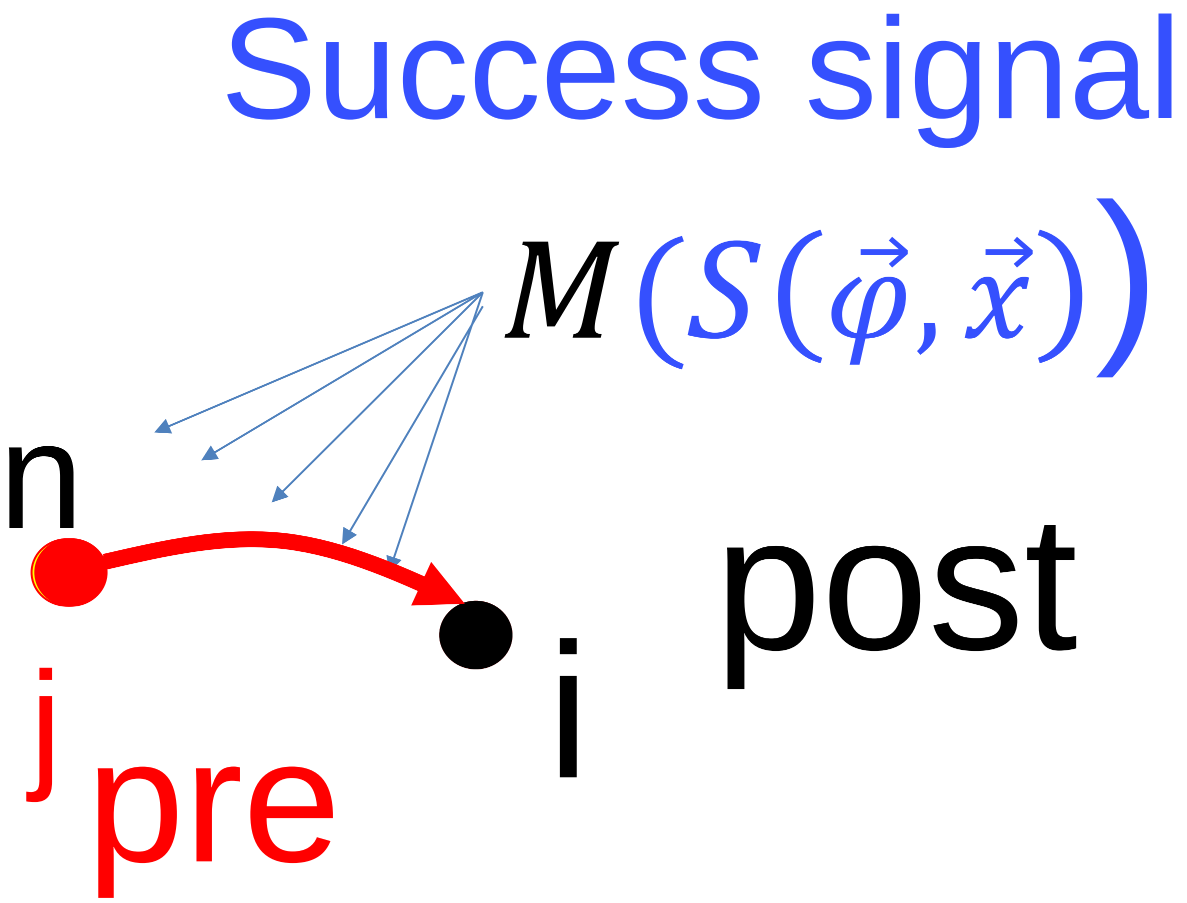
- If furthermore the success signal arrives within one second, then the weight is updated.

Review: Three-factor rules with eligibility trace

Three-factor rule defines a framework

x_j = activity-trace left by of presynaptic neuron

φ_i = activity-trace left by of postsynaptic neuron



Step 1: co-activation sets eligibility trace

$$\Delta z_{ij} = \eta \; f(\varphi_i)\, g(x_j)$$

Step 2: eligibility trace decays over time

$$z_{ij} \leftarrow \lambda_z z_{ij}$$

Step 3: eligibility trace translated into weight change

$$\Delta w_{ij} = \eta\, M\big(S(\vec{\varphi},\vec{x})\big) z_{ij}$$

Previous slide:

There are many different Hebbian rules or STDP rules. Similarly, there is not a single three-factor rule. Rather three-factor rules are a framework formulated as follows:

- The trace left by presynaptic activity contributes some nonlinear factor $g(x_j)$

- The trace left by postsynaptic activity contributes some nonlinear factor $f(\varphi_i)$

- The eligibility trace e_{ij} is changed proportional to the two factors f times g

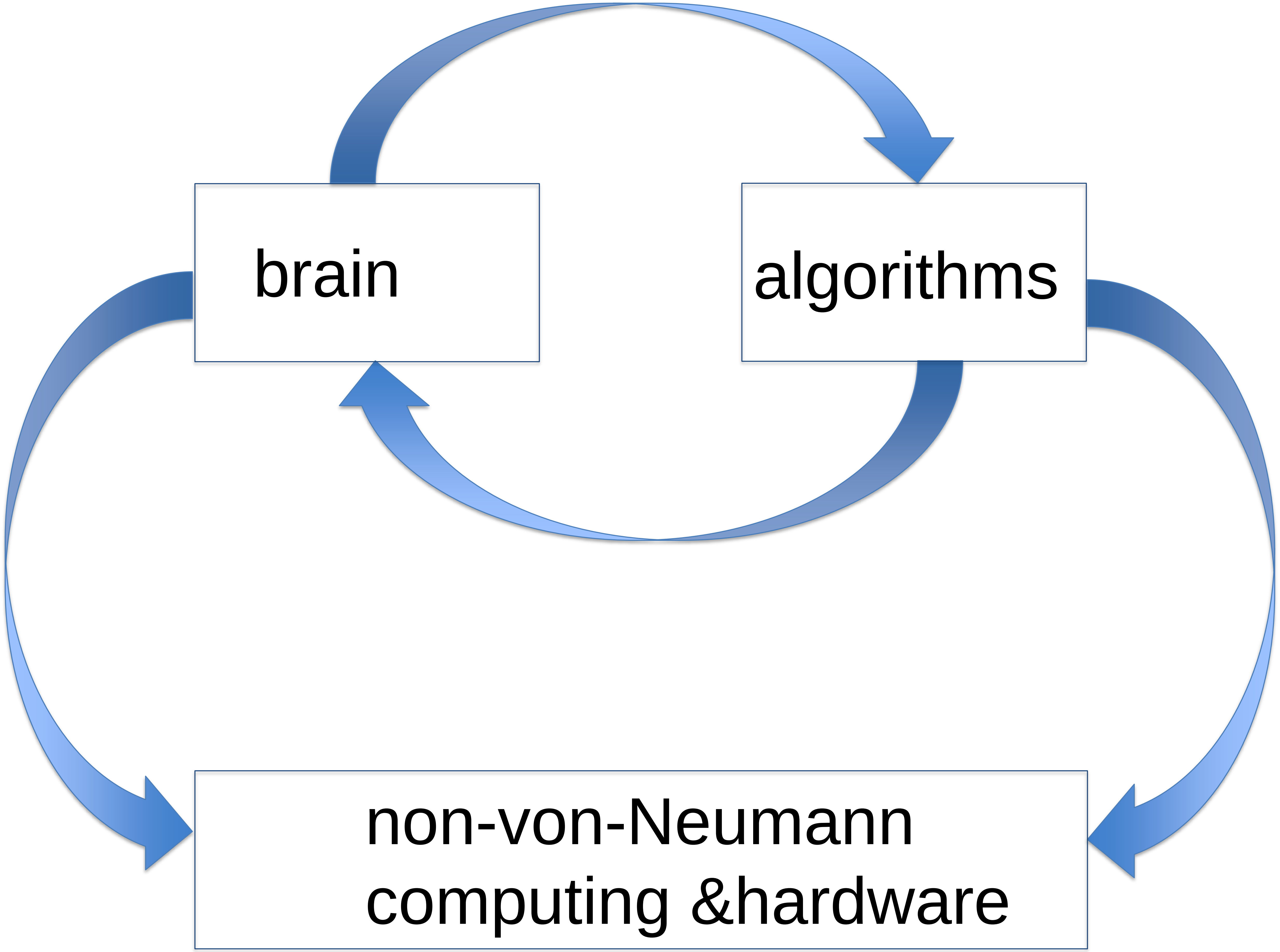
λ_z

- The eligibility trace decays by a factor λ_z corresponding to a time scale of about one second

- Weights are updates proportional to eligibility trace e_{ij} times M with a modulator M that is a nonlinear function of the success S. The modulator is the ‘third factor’ in the update rule.

- The modulator M adjusts not only the learning speed but also the direction of change. In other words, the sign of the update (increase/decrease) depends on the sign of M.

Three- factor Learning Rules and Spiking Neurons



Exploit: spikes are ‘rare’ events.
(most of a time a neuron does not a emit a spike)

Summary:

- Neurons communicate by short pulses
- EPSPs decay over 10ms
- Pulses last 1ms
- Pulses are rare events
- A pulse timing pre-before-post (within 20ms) sets an eligibility trace
- The eligibility trace decay over 1s
- Dopamine, a global neuromodulator, sends a TD signal

Previous slide:

After this introduction to spiking neurons, and review of three-factor rules, we make a small detour to an application that you have seen already at several occasions.

And then we are prepared to look at the first hardware implementation.

Learning in Neural Networks :

from brain-style computing to neuromorphic computing

Wulfram Gerstner

EPFL, Lausanne, Switzerland

1. Review: brain-style computing
2. **Spikey Chip/BrainScales** (Univ. Heidelberg)

(previous slide)

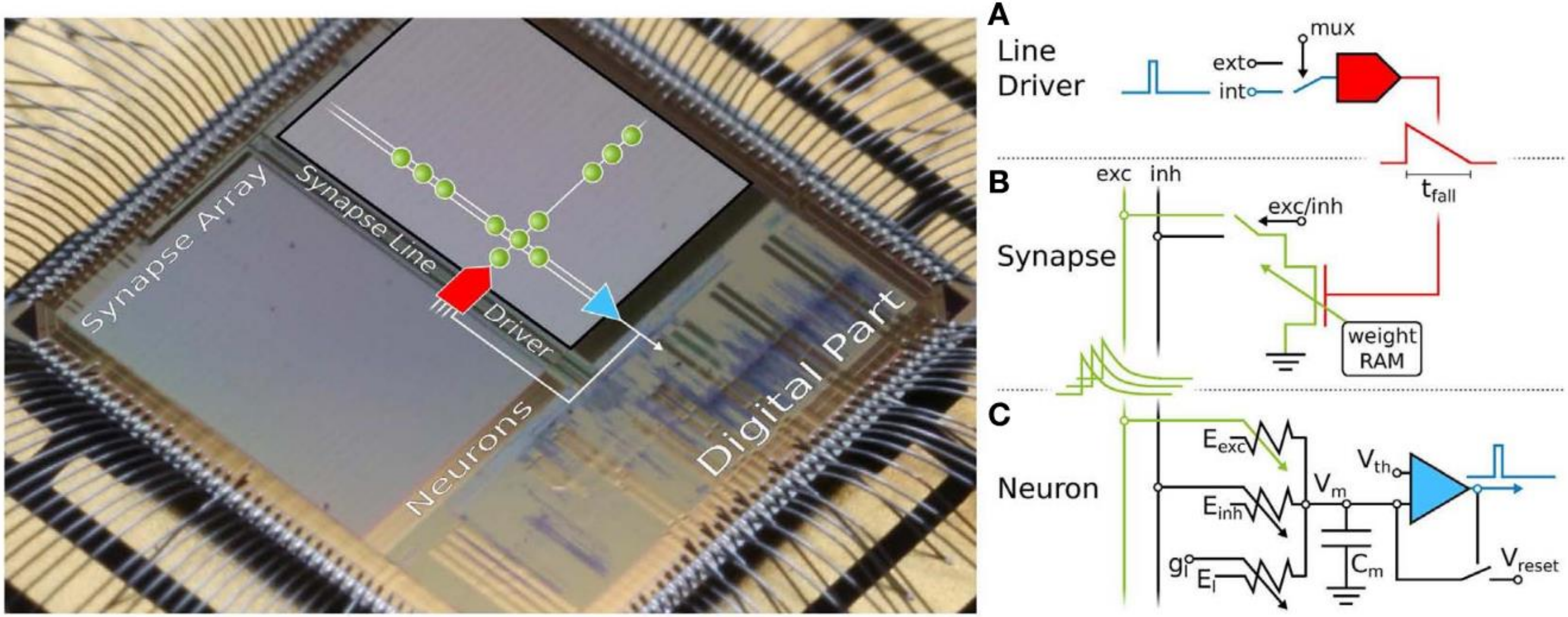
Neuromorphic hardware is a hot topic.
Many (but not all) neuromorphic chips use spiking neurons.

Gläscher et al. 2010

University of Heidelberg: Brainscales/Spikey

Integrate-and-fire neurons on chip. STDP on chip.

Analog hardware (physical realization).



Pfeil et al., Front. Neurosci. 2013
https://doi.org/10.3389/fnins.2013.00011

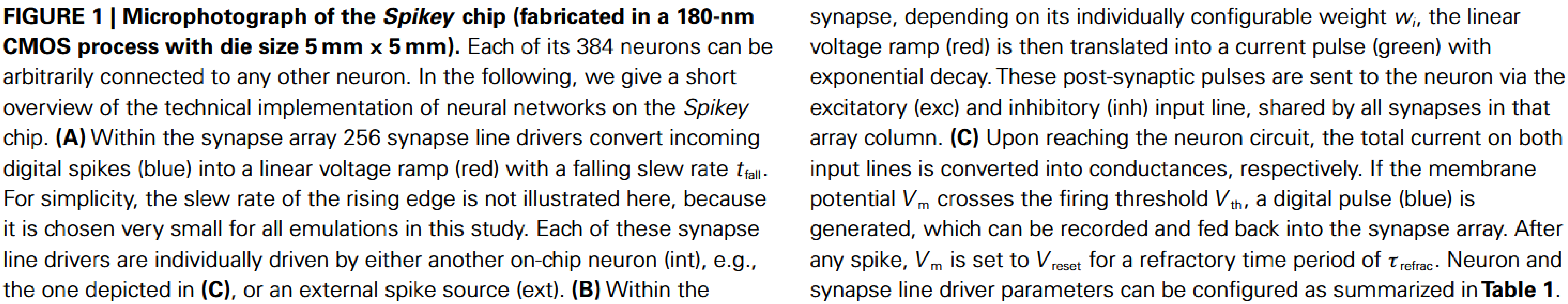
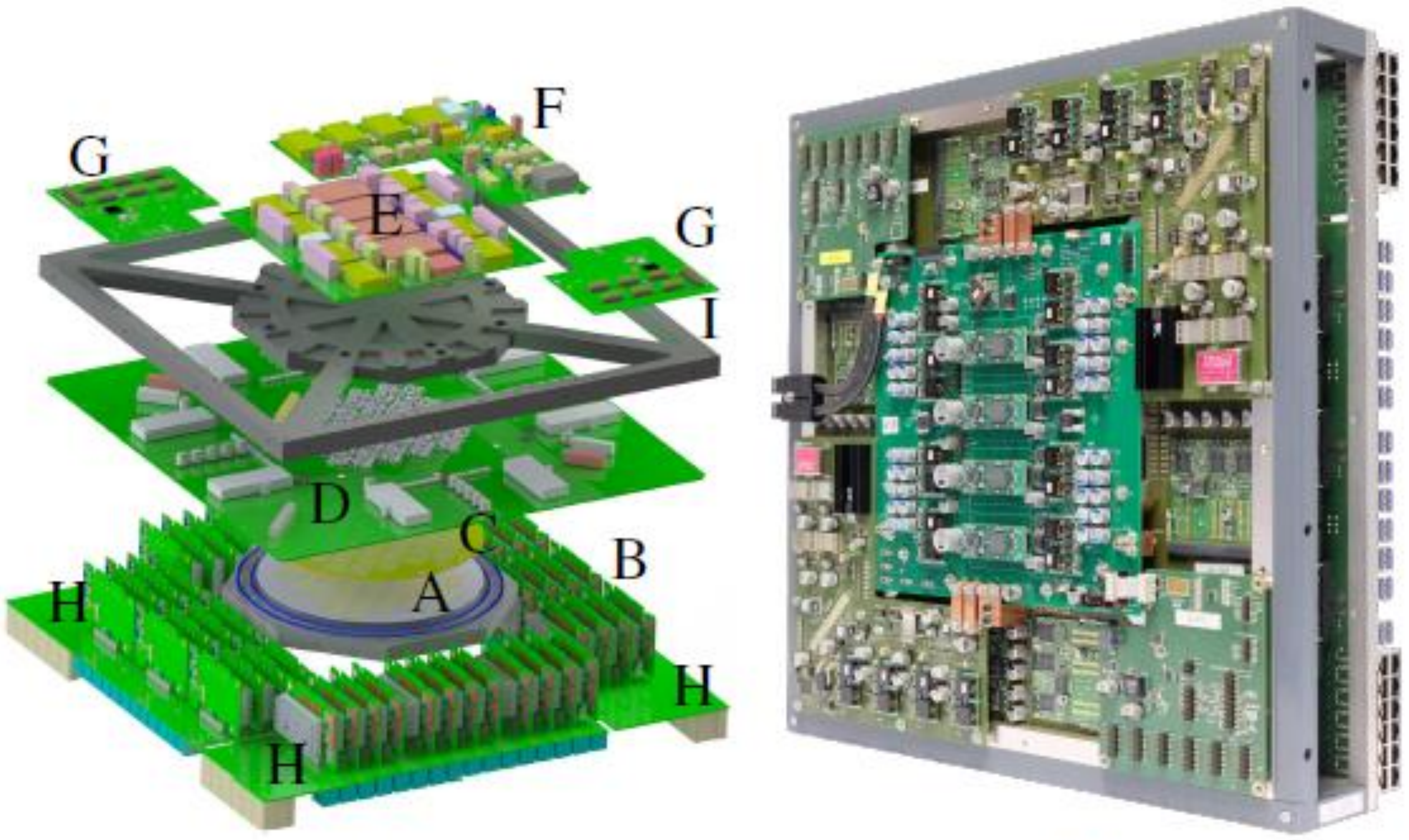


FIGURE 1 | Microphotograph of the *Spikey* chip (fabricated in a 180-nm CMOS process with die size 5 mm x 5 mm). Each of its 384 neurons can be arbitrarily connected to any other neuron. In the following, we give a short overview of the technical implementation of neural networks on the *Spikey* chip. **(A)** Within the synapse array 256 synapse line drivers convert incoming digital spikes (blue) into a linear voltage ramp (red) with a falling slew rate t_{fall}. For simplicity, the slew rate of the rising edge is not illustrated here, because it is chosen very small for all emulations in this study. Each of these synapse line drivers are individually driven by either another on-chip neuron (int), e.g., the one depicted in **(C)**, or an external spike source (ext). **(B)** Within the synapse, depending on its individually configurable weight w_i, the linear voltage ramp (red) is then translated into a current pulse (green) with exponential decay. These post-synaptic pulses are sent to the neuron via the excitatory (exc) and inhibitory (inh) input line, shared by all synapses in that array column. **(C)** Upon reaching the neuron circuit, the total current on both input lines is converted into conductances, respectively. If the membrane potential V_{m} crosses the firing threshold V_{th}, a digital pulse (blue) is generated, which can be recorded and fed back into the synapse array. After any spike, V_{m} is set to V_{reset} for a refractory time period of τ_{refrac}. Neuron and synapse line driver parameters can be configured as summarized in **Table 1**.

Pfeil et al., Front. Neurosci. 2013

published: 18 February 2013
doi: 10.3389/fnins.2013.00011

University of Heidelberg: Brainscales/Spikey



(a)

(b)

Waverscale integration:

-512 neurons/chip

-384 chips/waver

-analog neurons (RC circuit)

-digital communication

-energy: 0.1 nJ per syn. event

-learning with hardware in the loop

Schmitt et al.
2017 International Joint Conference on Neural Networks (IJCNN), DOI: 10.1109/IJCNN.2017.7966125

(previous slide)

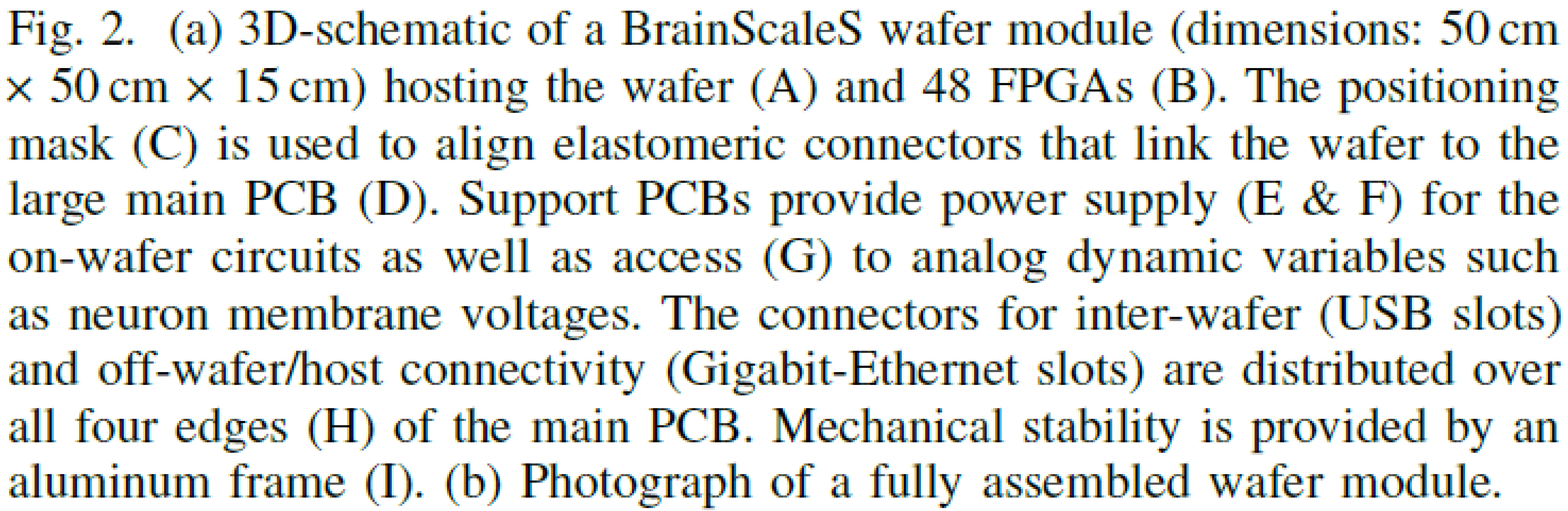


Fig. 2. (a) 3D-schematic of a BrainScaleS wafer module (dimensions: 50 cm × 50 cm × 15 cm) hosting the wafer (A) and 48 FPGAs (B). The positioning mask (C) is used to align elastomeric connectors that link the wafer to the large main PCB (D). Support PCBs provide power supply (E & F) for the on-wafer circuits as well as access (G) to analog dynamic variables such as neuron membrane voltages. The connectors for inter-wafer (USB slots) and off-wafer/host connectivity (Gigabit-Ethernet slots) are distributed over all four edges (H) of the main PCB. Mechanical stability is provided by an aluminum frame (I). (b) Photograph of a fully assembled wafer module.

Learning in Neural Networks :

from brain-style computing to neuromorphic computing

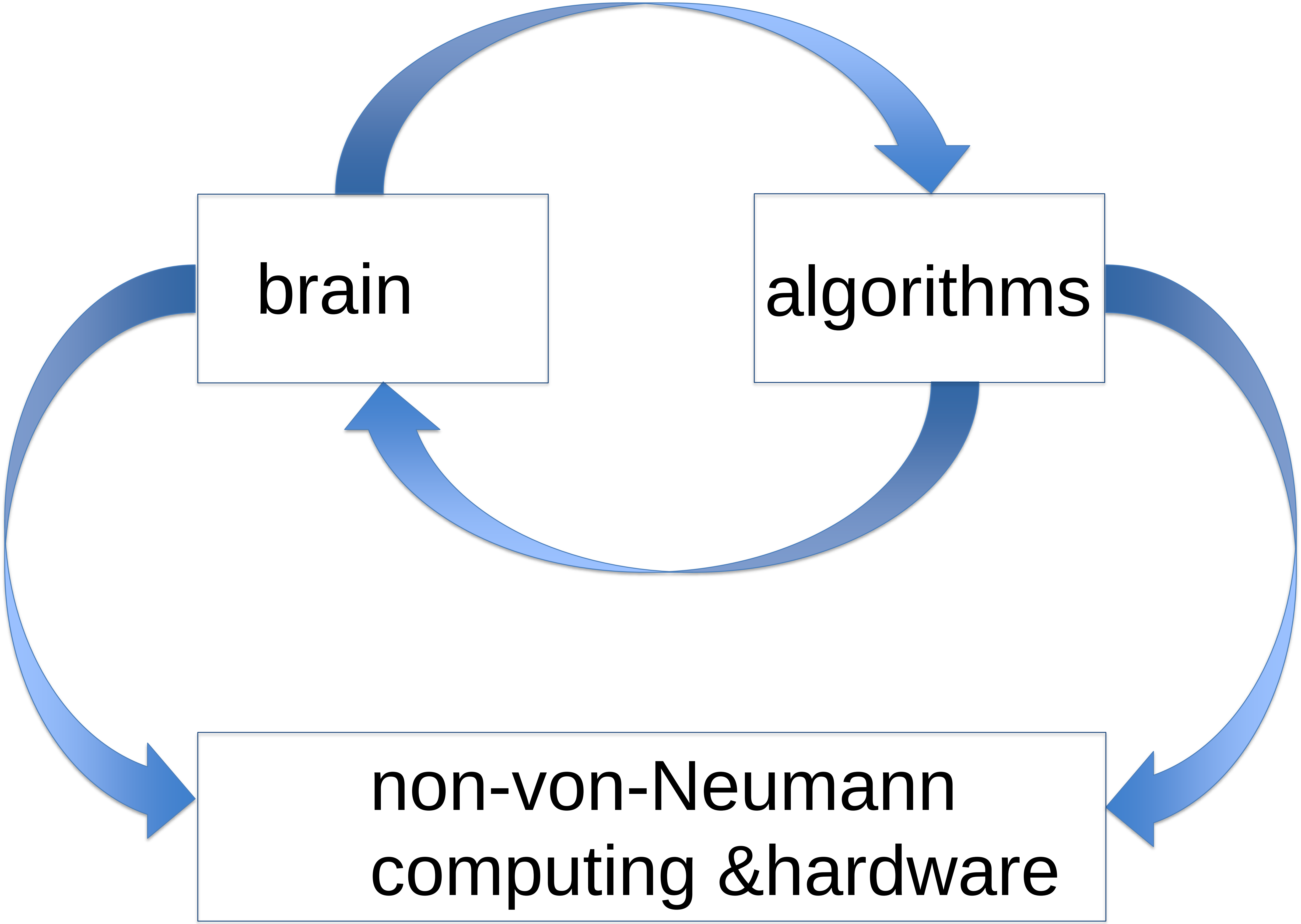
Wulfram Gerstner

EPFL, Lausanne, Switzerland

1. Detour: Spiking Neural Networks (SNN)
2. Spikey Chip (Univ. Heidelberg)
3. **Loihi Chip (INTEL)**

Three-factor Learning Rules

Spiking neurons (event-based signal transmission)



brain

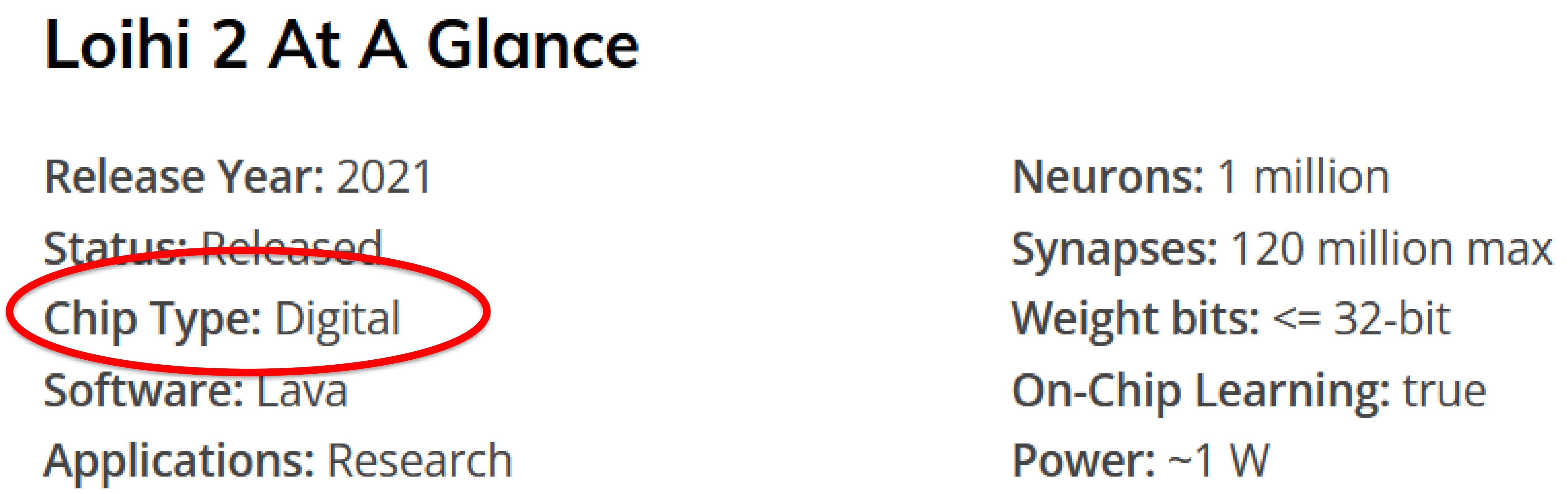
algorithms

non-von-Neumann computing &hardware

INTEL:

Loihi (announced 2017, appeared 2018)

Loihi2 (announced fall 2021): 1 chip = 1Mio neurons



Loihi 2 At A Glance

Release Year: 2021

Status: Released

Chip Type: Digital

Software: Lava

Applications: Research

Neurons: 1 million

Synapses: 120 million max

Weight bits: <= 32-bit

On-Chip Learning: true

Power: ~1 W

Previous slide:

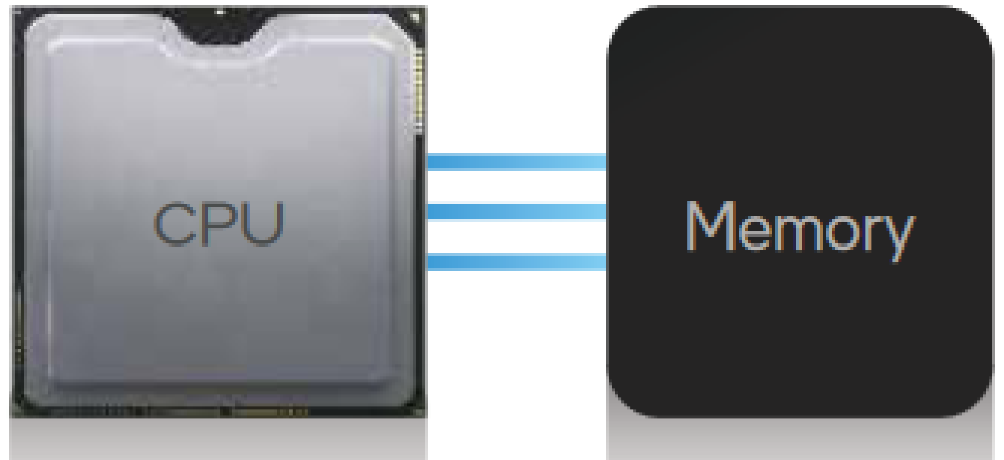
The Loihi chip of Intel that appeared as a research support chip in 2017/2018 is interesting because it gives a direct implementation of three-factor rules where the third factor is a GLOBAL signal

The modern chip Loihi2 moved away from this concept. There is still a third factor but it is now LOCAL per neuron. This also enables to implement BackProp algorithms

INTEL, Loihi research chip

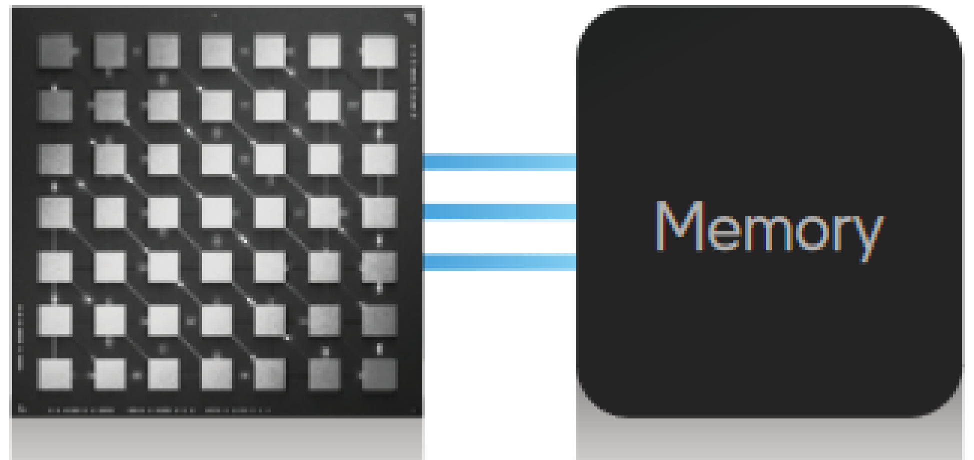
Computing Architectures

Conventional Computing



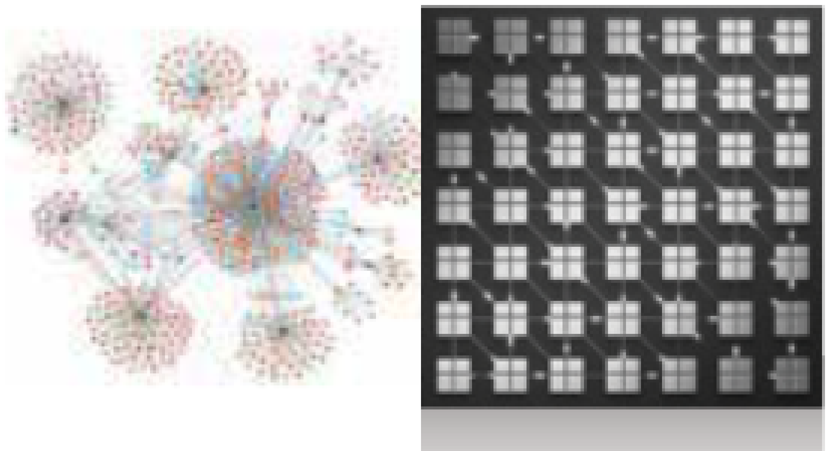
- Programming by Encoding Algorithms
- Synchronous Clocking
- Sequential Threads of Control

Parallel Computing

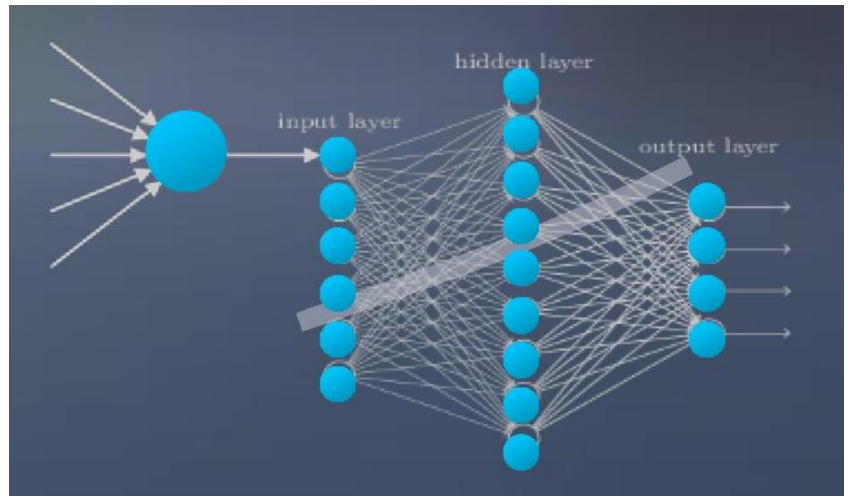


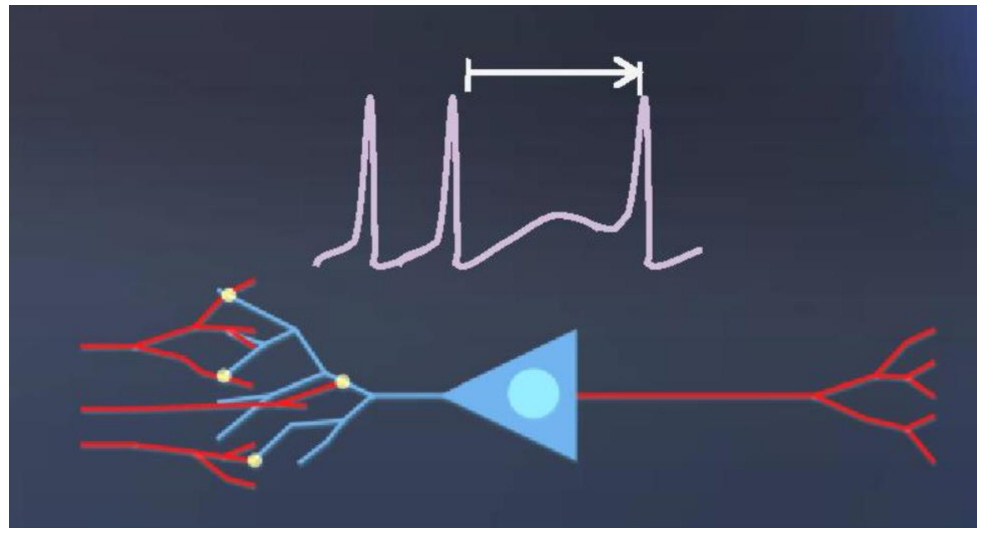
- Offline Training Using Labeled Datasets
- Synchronous Clocking
- Parallel Dense Compute

Neuromorphic Computing



- Learn On-the-Fly Through Neuron Firing Rules
- Asynchronous Event-Based Spikes
- Parallel Sparse Compute





https://download.intel.com/newsroom/2021/new-technologies/neuromorphic-computing-loihi-2-brief.pdf

Previous slide:

This slide from INTEL emphasize the differences in the computing architecture of LOIHI (first chip).

LEFT: classical von Neumann computing with separation of CPU and memory. Compute operations are mapped to logical operations performed in discrete time.

MIDDLE: Parallel computing and GPU architectures. The separation of computing and memory remains, and operations are still performed in discrete time. The only difference is that certain operations (such as convolutions) or updates of layer-wise dynamics in ANNs can be performed in parallel.

RIGHT: Neuromorphic computing architectures. Neurons compute with spikes which leads to nonlinear compute operations and signal transmission at rare moments in time defined by the moments of threshold-crossing. In between neurons are updated in ‘subthreshold’ mode with simple linear operations (leaky integration). Ideally, computing is asynchronous and in continuous time (even though this specific INTEL hardware implementation is still ‘digital’).

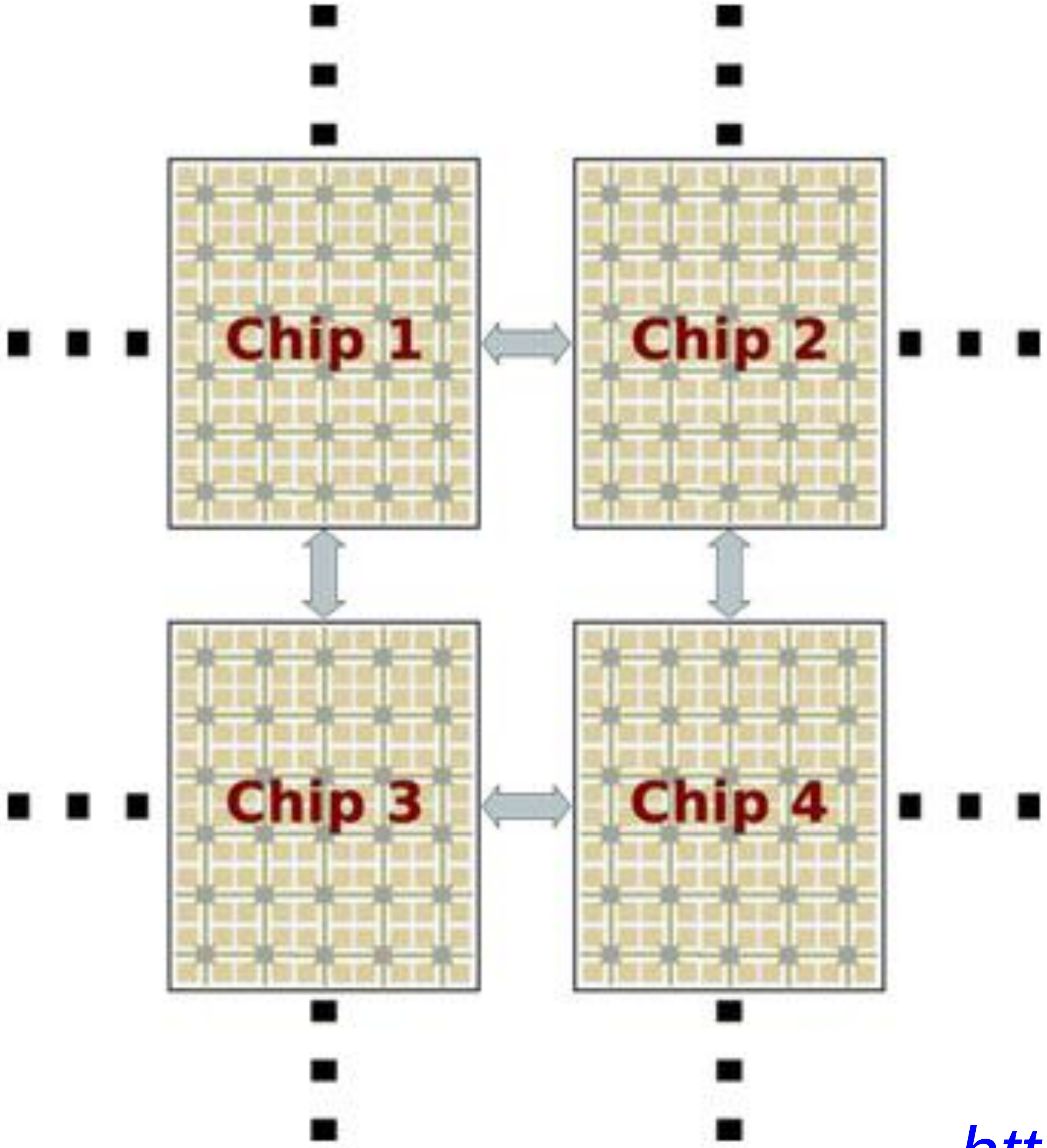
More recently the first generation of Loihi has been replaced by Loihi2 with more general functionalities.

Two related arguments:

- energy consumption:

  Loihi < 1 W (GPU > 300W)

- asynchronous computing/event-based messaging



1 chip = mesh of 128 neuromorphic cores

Spiking neural network (SNN)

1 core = 1024 simple spiking neurons:
leaky integrate-and-fire

On-chip integrated learning rule

https://en.wikichip.org/wiki/intel/loihi

Previous slide:

Why would one want to change the computing architecture?

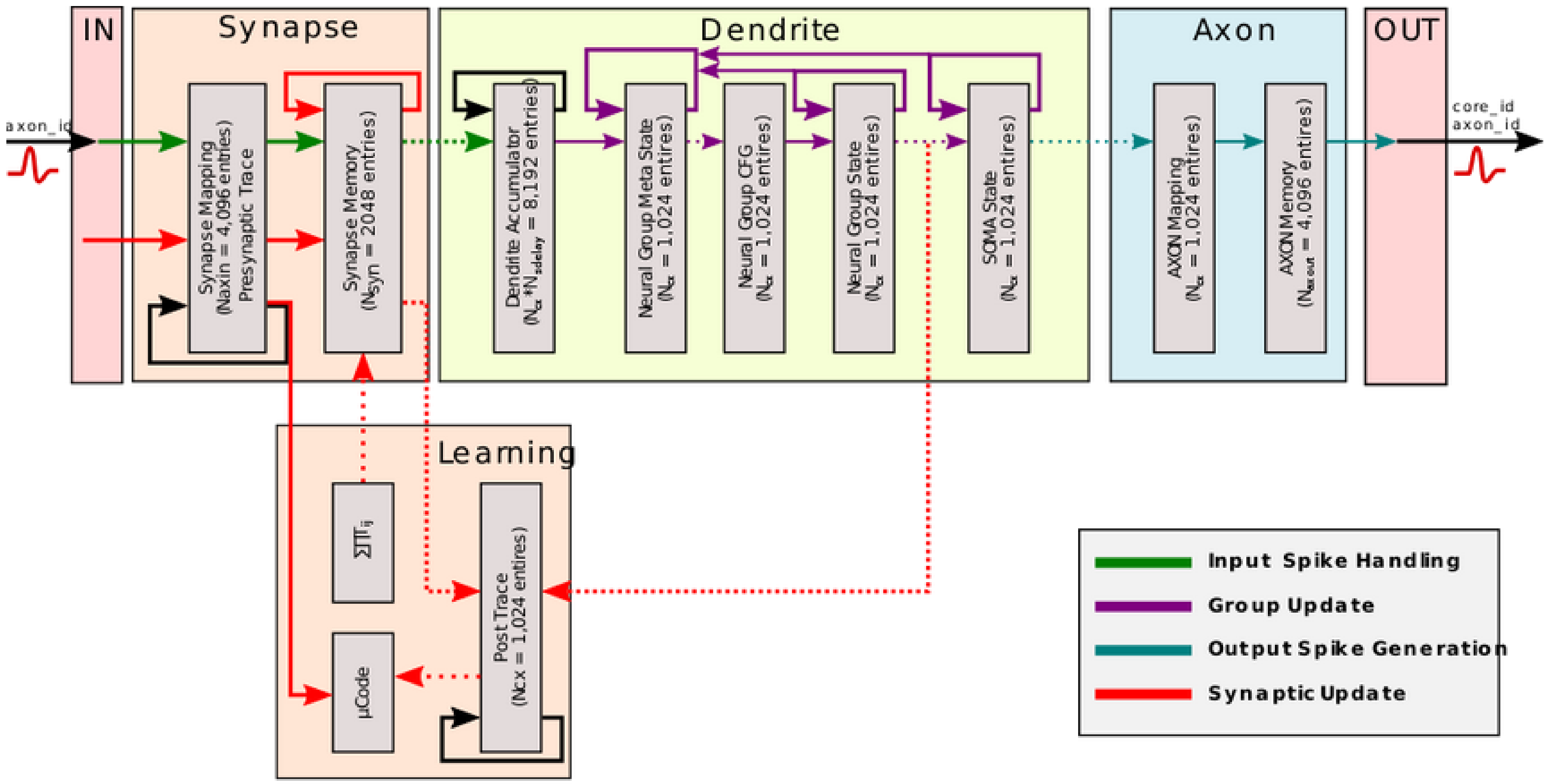
Essentially because asynchronous, event-based computing could lead to enormous reductions in energy consumptions, since the expensive nonlinear processing steps and transmission steps are sparse in time: they are rare compared to the elementary time step in a discrete-time implementation.

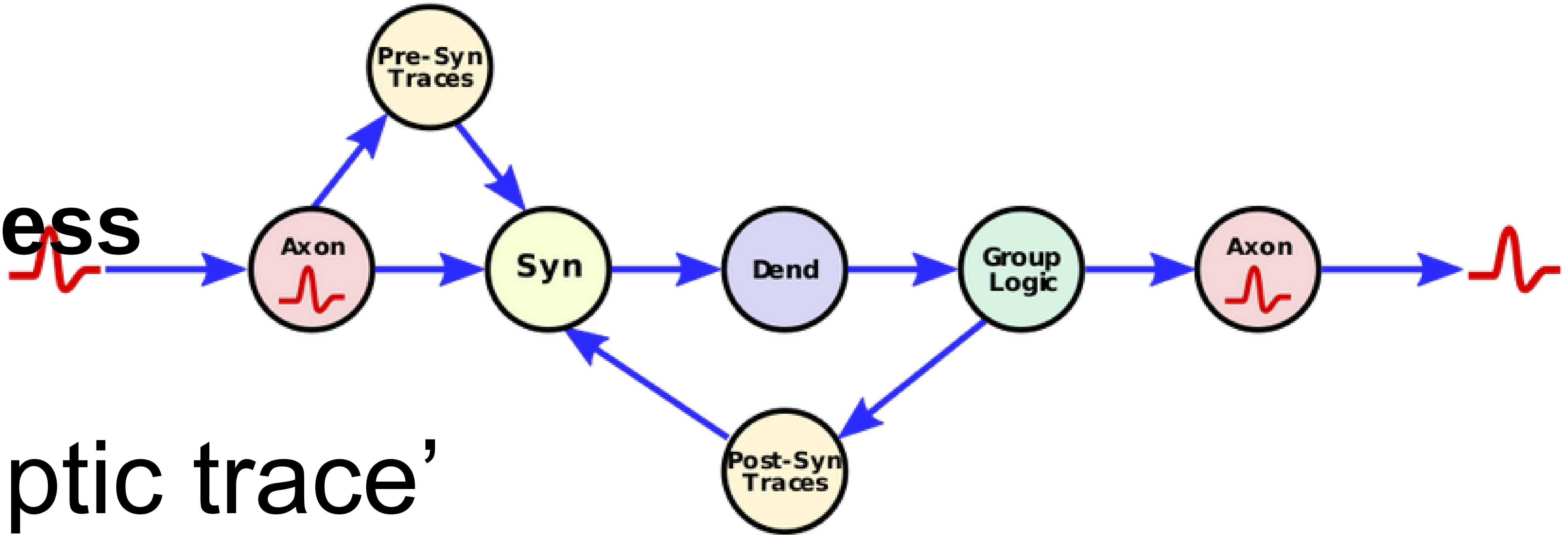
1 chip contains 128 cores, each one able to simulate about 1000 simple leaky integrate-and-fire neurons.

Loihi: (first chip, 2018)

- 128 neuron cores per chip
- Up to 128’000 neurons per chip
- 2 billion transistors
- Fully digital chip

- Standard integrate-fire neuron model
- **Three-factor learning rule trace(pre) trace(post) success**


IN
Synapse
Dendrite
Axon
OUT
axon_id
Synapse Mapping (Naxin = 4,096 entries) Presynaptic Trace
Synapse Memory (Nsyn = 2048 entries)
Dendrite Accumulator (N_{cx}*N_{sdelay} = 8,192 entries)
Neural Group Meta State (N_{cx} = 1,024 entires)
Neural Goup CFG (N_{cx} = 1,024 entires)
Neural Group State (N_{cx} = 1,024 entires)
SOMA State (N_{cx} = 1,024 entires)
AXON Mapping (N_{cx} = 1,024 entires)
AXON Memory (N_{axout} = 4,096 entires)
core_id
axon_id
Learning
ΣΠT_{ij}
μCode
Post Trace (Ncx = 1,024 entires)
Input Spike Handling
Group Update
Output Spike Generation
Synaptic Update



‘each spike leaves a synaptic trace’

→ STDP coincidence (time scale 10ms in biology)

Previous slide:

Importantly, the framework of the learning rule that is possible on the Loihi chip is exactly that of three-factor rules explained above.

Each presynaptic spike leaves a trace (synaptic trace/NOT eligibility trace). The combination with the trace left by a postsynaptic spike gives the coincidence signal. Further combination with a success signal defines the weight update.

Learning rules

→Loihi (2017): Three-factor learning rules

presynaptic factor, postsynaptic factor, **global success**

→ **single-layer RL algorithms**

→ Loihi2 (2022): Detailed three-factor learning rules

presynaptic factor, postsynaptic factor, **neuron-specific feedback**

→ **approximate BackProp in Multi-Layer RL**

Previous slide:

In the new version, they generalized the learning rule so that it can now also implement an approximate version of BackProp.

Introducing Loihi 2

Programmable Neurons

Neuron models described by microcode instructions

Generalized Spikes

Spikes carry integer magnitudes for greater workload precision

Enhanced Learning

Support for powerful new "three factor" learning rules from neuroscience

10x Faster

2-10x faster circuits[2] and design optimizations speed up workloads by up to 10x[3]

8x More Neurons

Up to 1 million neurons per chip with up to 80x better synaptic utilization, in 1.9x smaller die

Better Scaling and Integration

3D scaling with 4x more bandwidth per link[4], >10x compression[5] with standard interfaces

Fabricated with Intel 4 process

(pre-production)

[2] Based on silicon characterization of Loihi 1 and a combination of silicon and pre-silicon simulation estimates for Loihi 2.

[3] Based on simulation modeling of a 9-layer Sigma-Delta Neural Network implementation of the PilotNet DNN inference workload compared to a rate-coded SNN implementation on Loihi 1.

[4] Based on pre-silicon circuit simulations.

[5] Based on a 7-chip Locally Competitive Algorithm workload analysis.

See backup for analysis details.
Results may vary.

Previous slide:

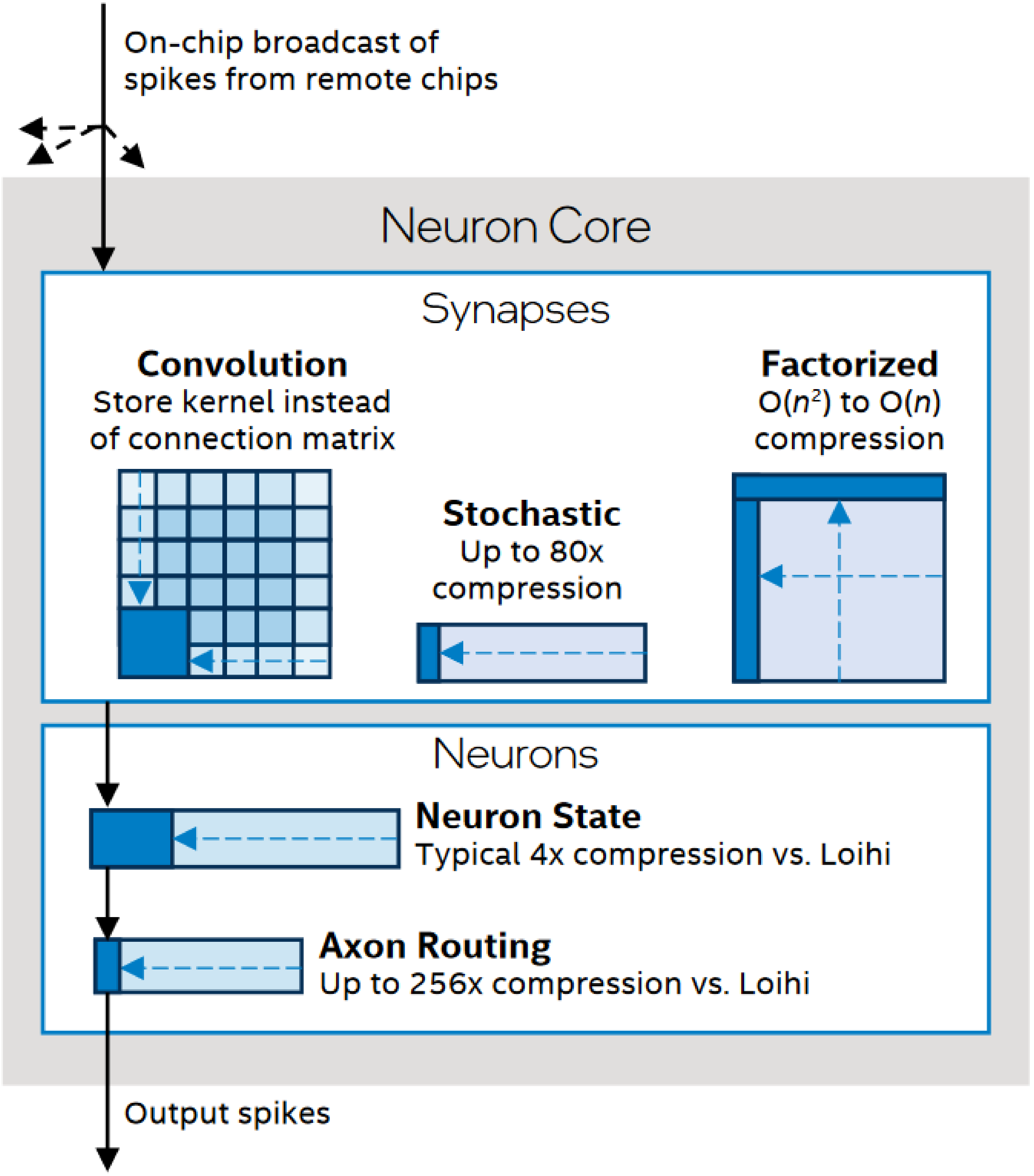
Official INTEL slide.

Loihi2 (2022):

- 128 neuron cores per chip
- Up to 1 Mio neurons per chip
- 2 billion transistors

- programmable neuron model
- programmable learning rule f(pre),g(post),3rd(neuron_i)

- spike broadcast at destination chip
- convolutional networks
- outer-product weight matrix
- Linked to C/Phython programming interface


On-chip broadcast of spikes from remote chips
Neuron Core
Synapses
Convolution
Store kernel instead of connection matrix
Stochastic
Up to 80x compression
Factorized
O(n^2) to O(n) compression
Neurons
Neuron State
Typical 4x compression vs. Loihi
Axon Routing
Up to 256x compression vs. Loihi
Output spikes

Previous slide:

Apart from spike broadcast (as opposed to targeted delivery lines), the chip also implements features such as weight matrices compatible with convolutional neural networks and outer-product weight matrices (factorial, see conv-net lecture).

Importantly, the learning rule framework now enables the user to switch from a GLOBAL third factor to a use-defined programmable NEURON-specific third factor.

The 80-percent question again:

[ ] In this hardware part, at least 60 percent of the material was new to me.

[ ] for this hardware part, I have the feeling that I
understood at least 80 percent of the material
(at the rough level at which it was presented)

Artificial Neural Networks and RL :

from brain-style computing to neuromorphic computing

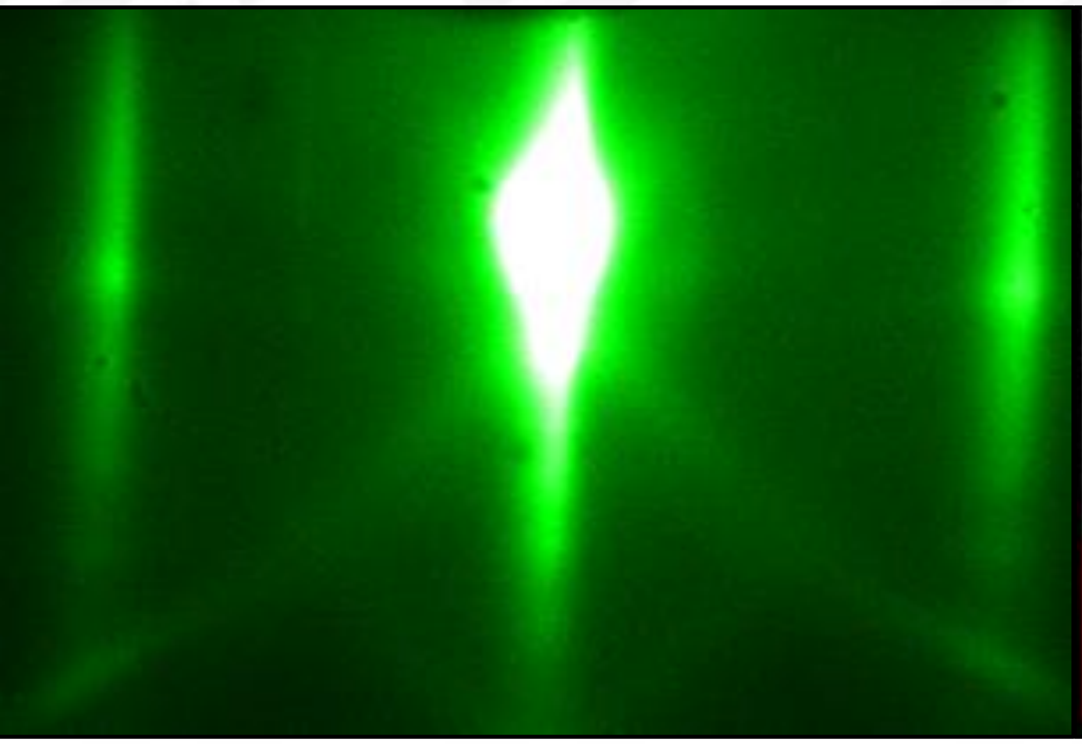
Wulfram Gerstner

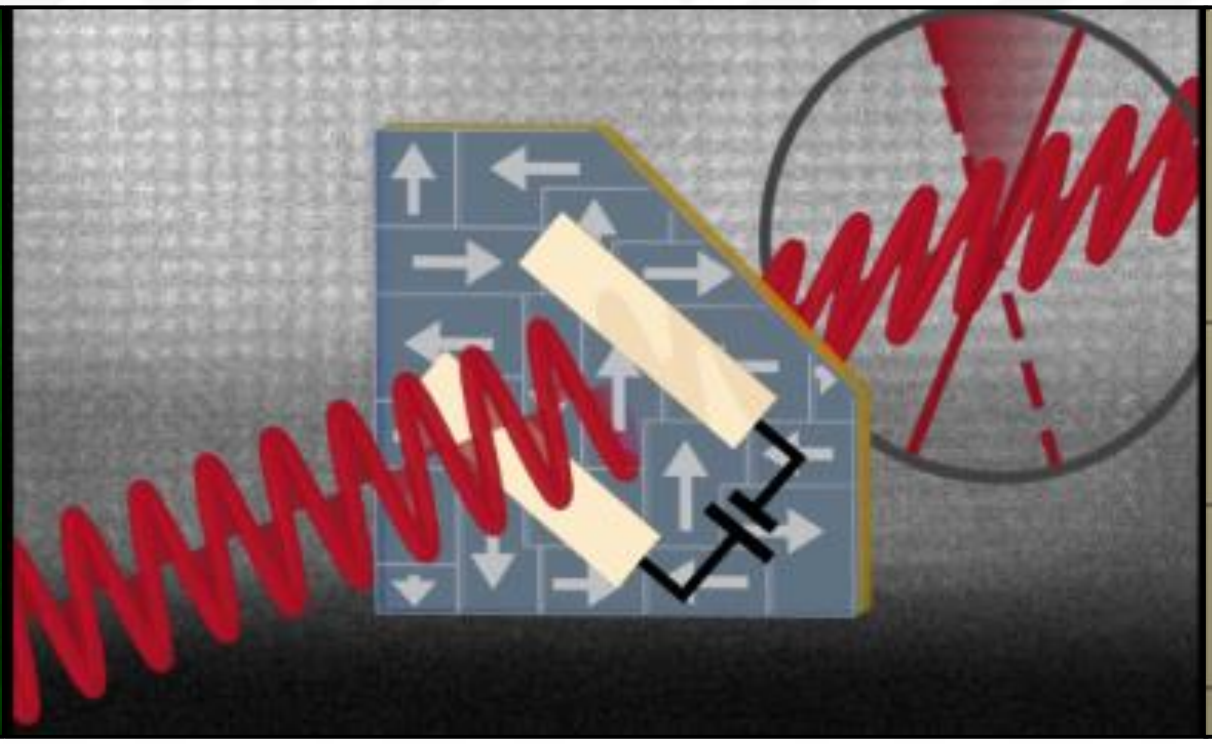
EPFL, Lausanne, Switzerland

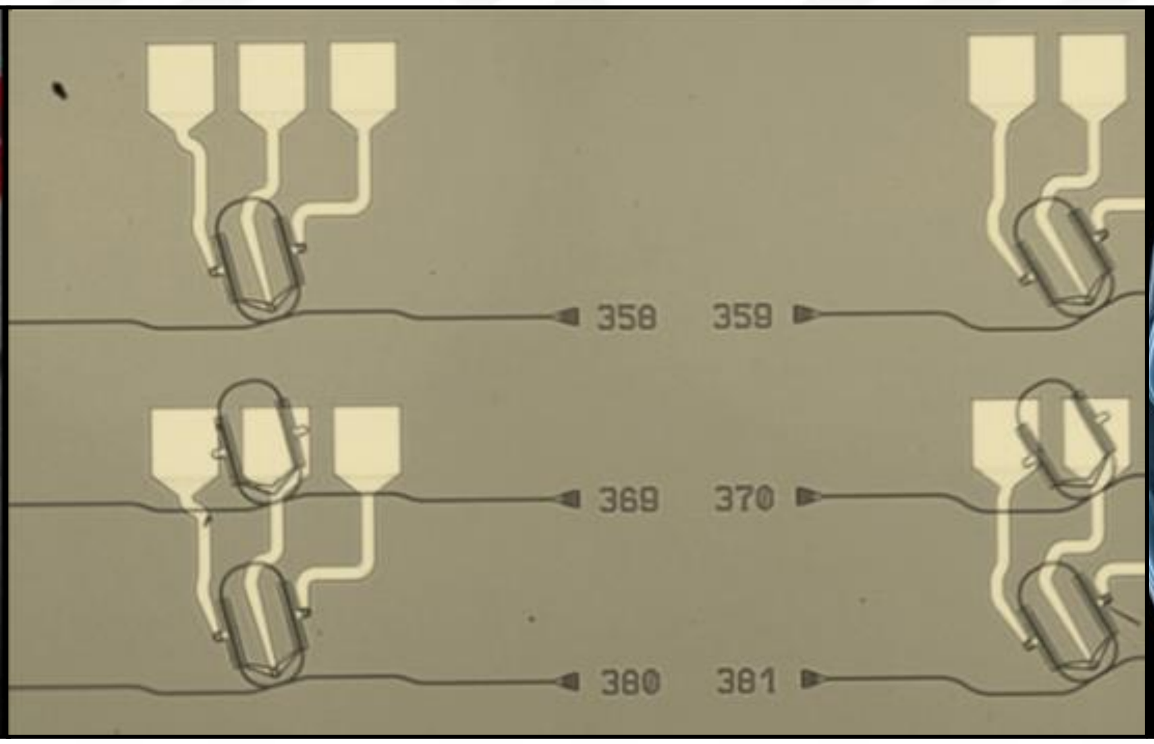
1. Detour: Spiking Neural Networks (SNN)
2. Example: Navigation in a Maze (Model Study)
3. Loihi Chip (INTEL)
4. **Memristor technology (IBM)**

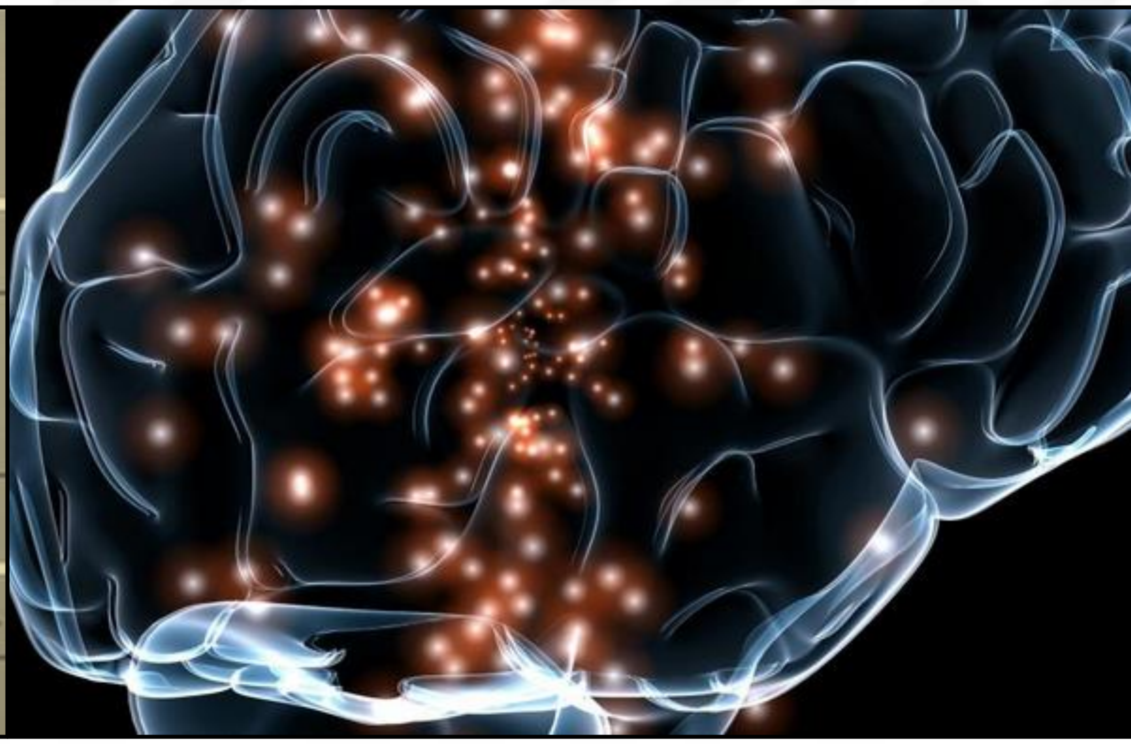
Analog synaptic signal processing for neural network inference and training

Bert Jan Offrein









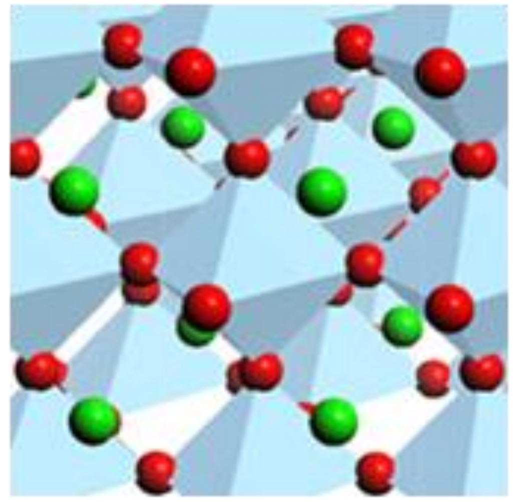
Neuromorphic Devices and Systems Group, IBM Research Europe - Zurich

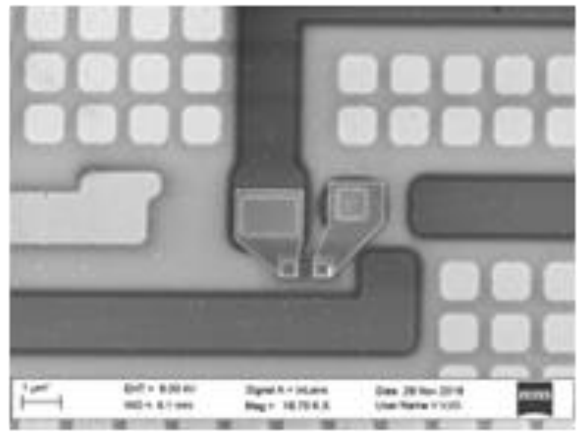
Reading

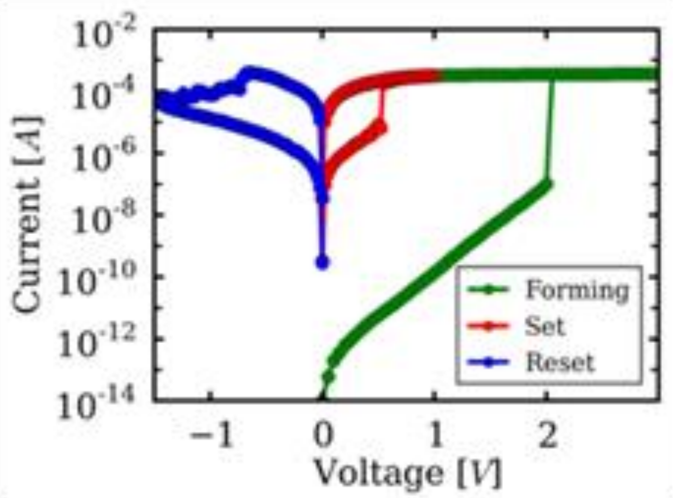
Bert Offrein et al., 2020, Prospects for photonic implementations of neuromorphic devices and systems, *IEEE Xplore, https://ieeexplore.ieee.org/abstract/document/9371915*

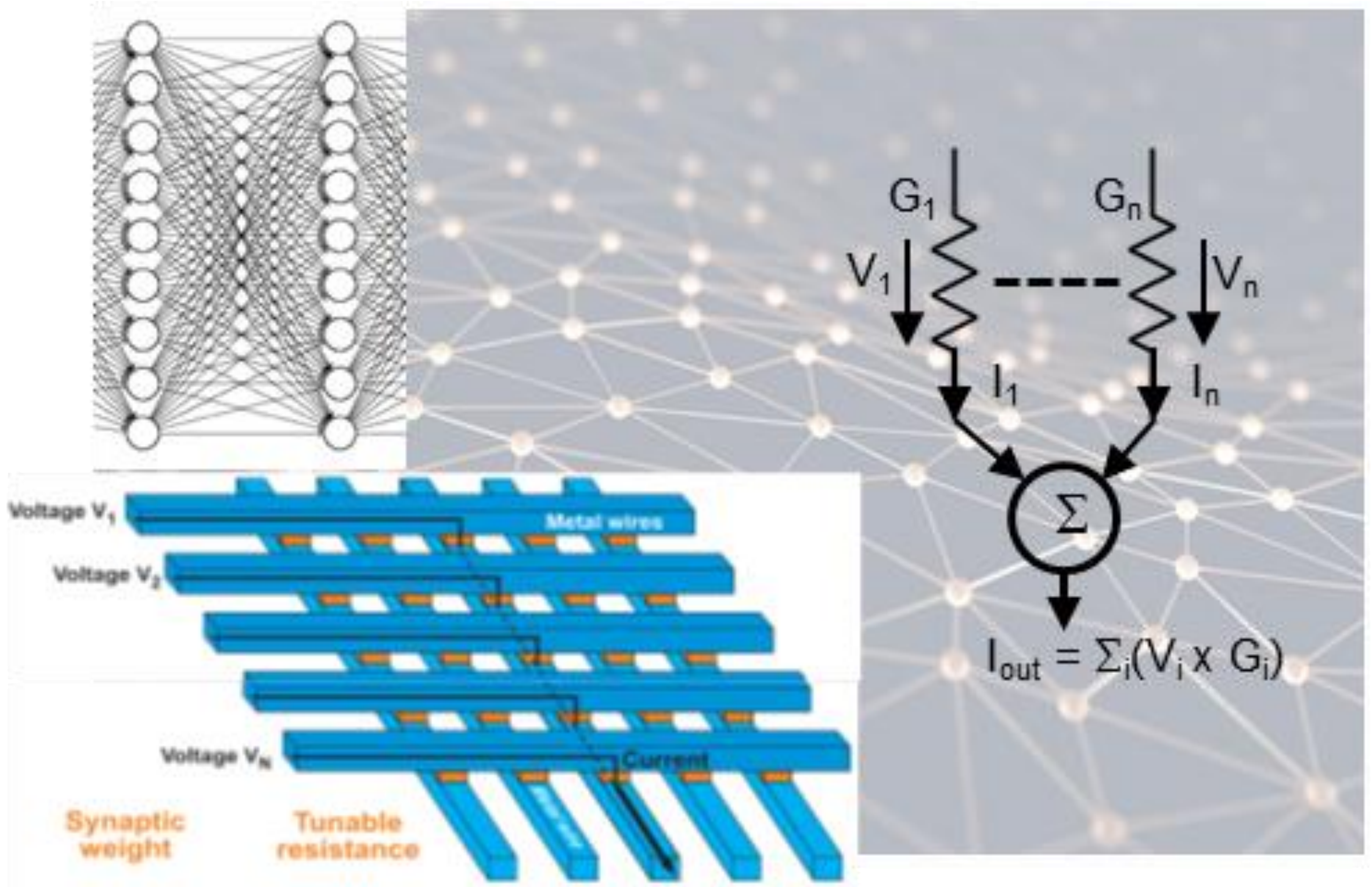
The slides are adapted from a presentation of Bert Offrein who leads a group of neuromorphic computing at IBM research in Zurich-Ruschlikon.

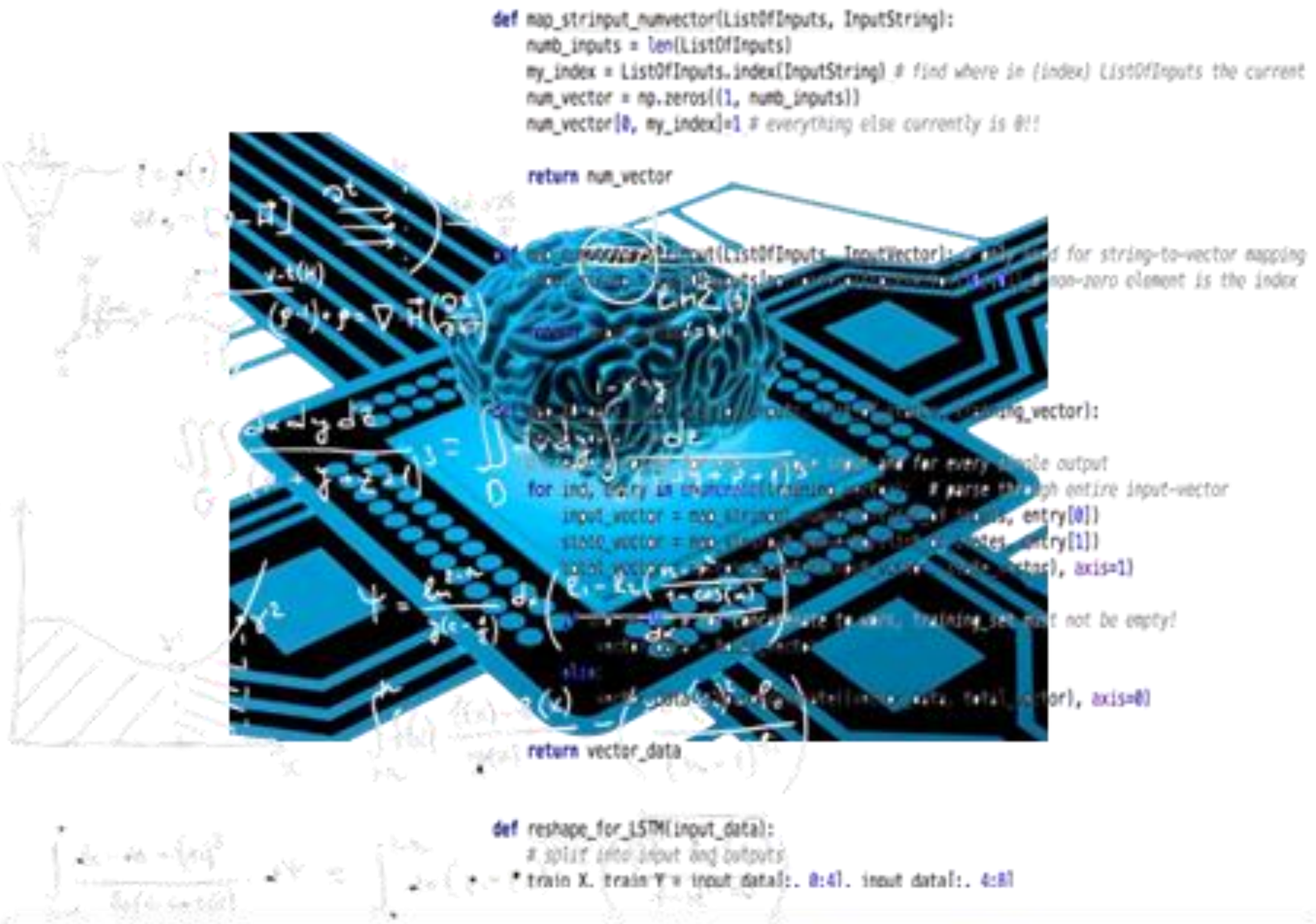
Accelerating Neuromorphic Workloads – Innovation required at all levels











```
def map_strinput_numvector(ListOfInputs, InputString):
    numb_inputs = len(ListOfInputs)
    my_index = ListOfInputs.index(InputString) # find where in (index) ListOfInputs the current
    num_vector = np.zeros((1, numb_inputs))
    num_vector[0, my_index]=1 # everything else currently is 0!!

    return num_vector
```

```
    return vector_data


def reshape_for_LSTM(input_data):
    # split into input and outputs
    train X, train Y = input data[:, 0:4], input data[:, 4:8]
```

New Materials and Devices

Non *von Neumann* Architecture

Hardware – Algorithm Interplay

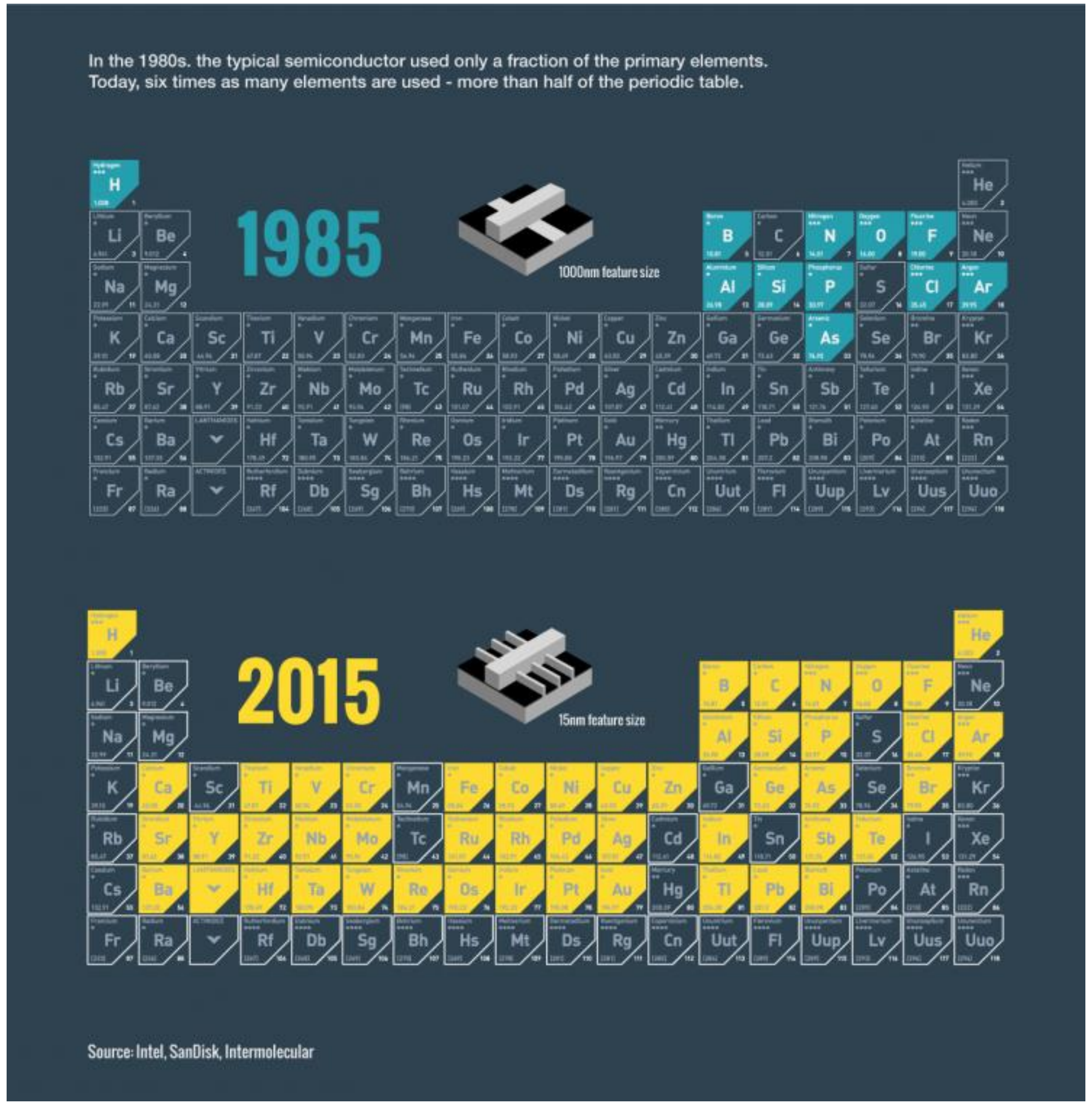
IBM slide

Previous slide:

The project of IBM research focuses mostly on Matrix multiplication (middle) and update of the matrix elements as a result of a learning rule (‘algorithm’, right).

Three pillars for Si technology

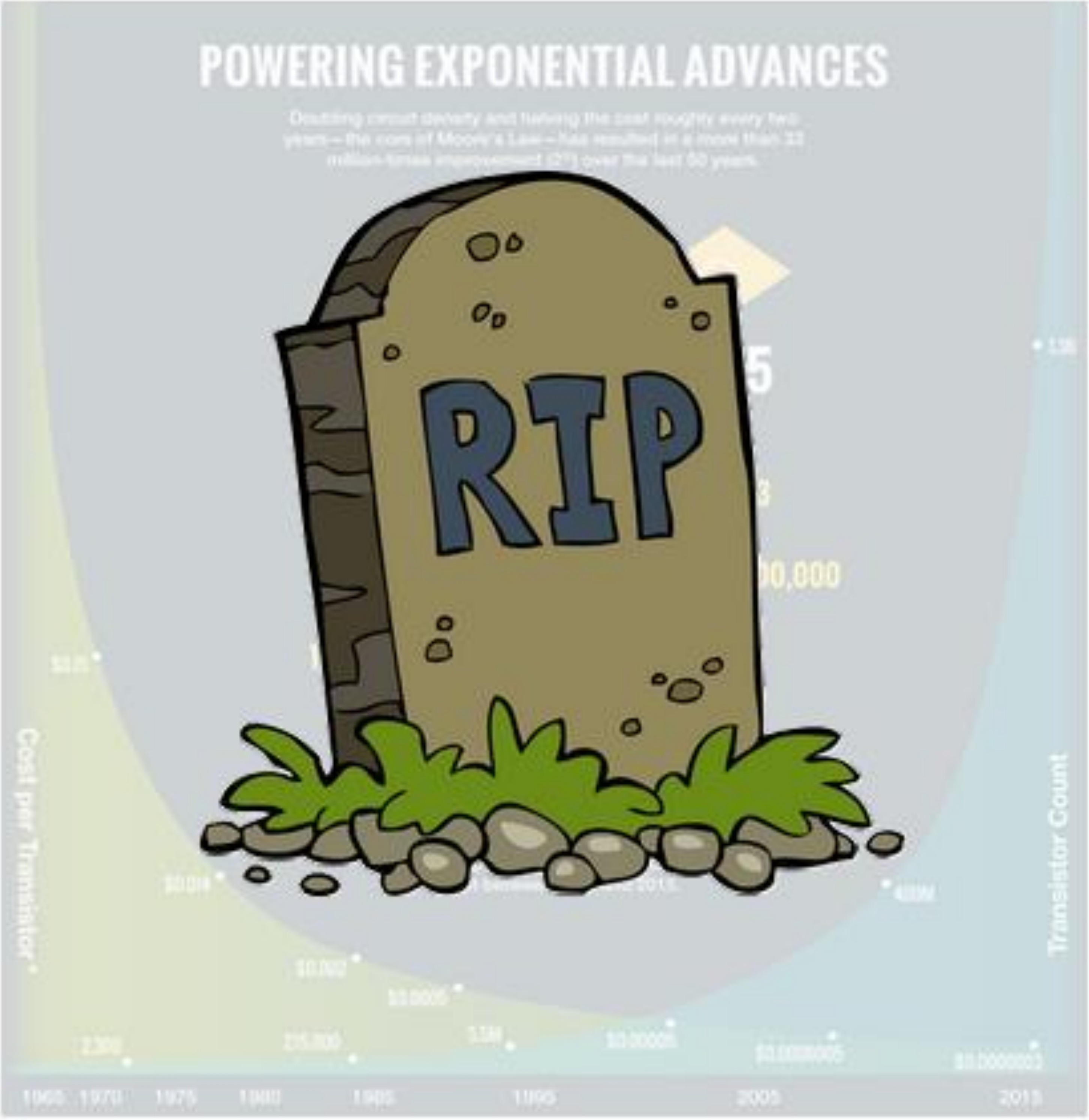
New combinations of Materials



In the 1980s. the typical semiconductor used only a fraction of the primary elements. Today, six times as many elements are used - more than half of the periodic table.

Source: Intel, SanDisk, Intermolecular

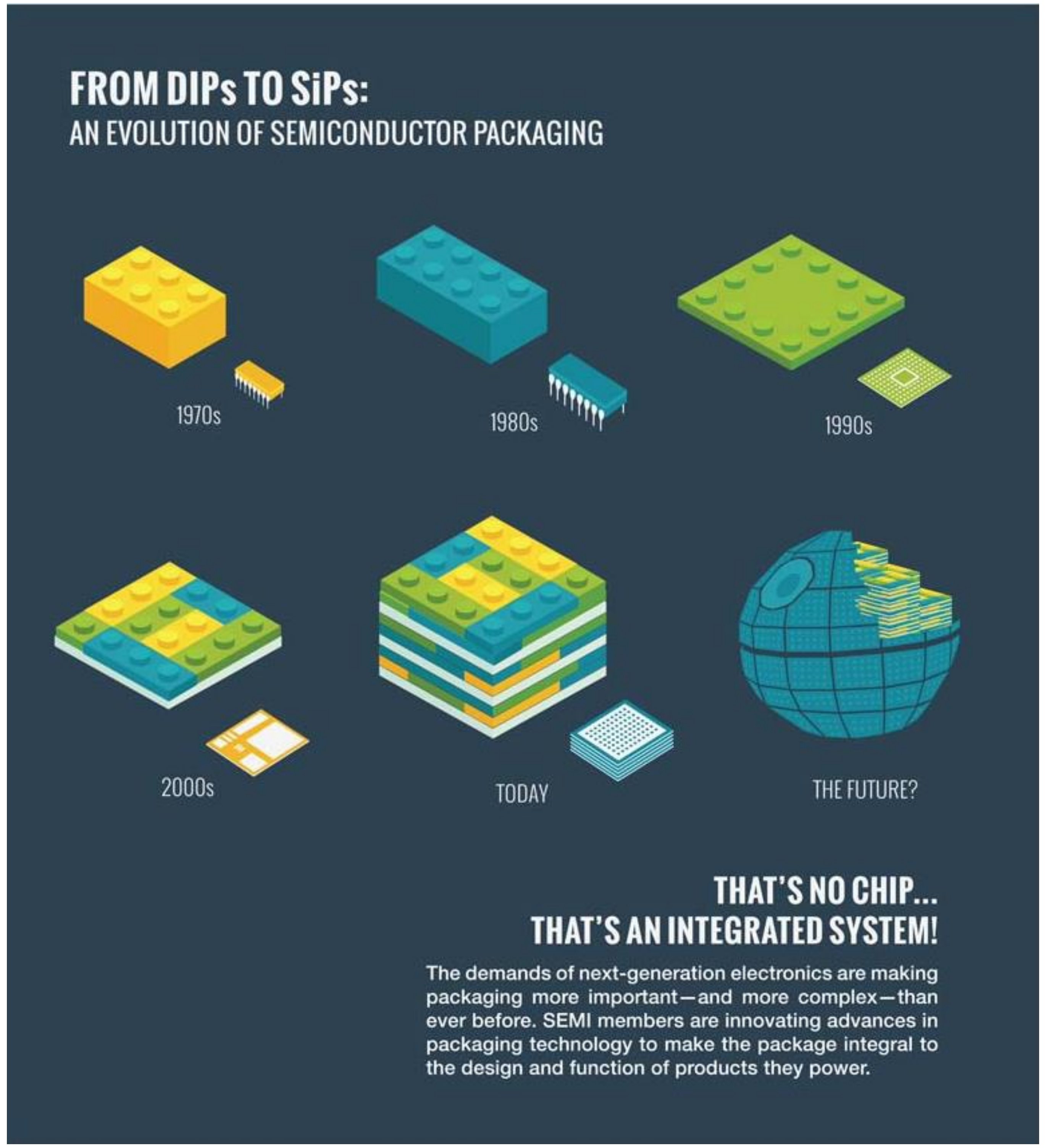
Scaling



POWERING EXPONENTIAL ADVANCES

Doubling circuit density and halving the cost roughly every two years—the core of Moore's Law—has resulted in a more than 33 million-times improvement (2^{25}) over the last 50 years.

Packaging in 3 dimensions



FROM DIPs TO SiPs:
AN EVOLUTION OF SEMICONDUCTOR PACKAGING

THAT'S NO CHIP...
THAT'S AN INTEGRATED SYSTEM!

The demands of next-generation electronics are making packaging more important—and more complex—than ever before. SEMI members are innovating advances in packaging technology to make the package integral to the design and function of products they power.

www.semi.org

The traditional scaling law (‘Moore’s law’) is dead! IBM slide

Previous slide:

In the IBM research the focus is more on new materials that enable faster and energy-efficient matrix multiplication as well as weight-update rules.

The three drivers of the changes are:

Left: New materials combine many more elements than older ones.

Middle: Moore’s law, the traditional scaling law of hardware performance increase, has come to its end.

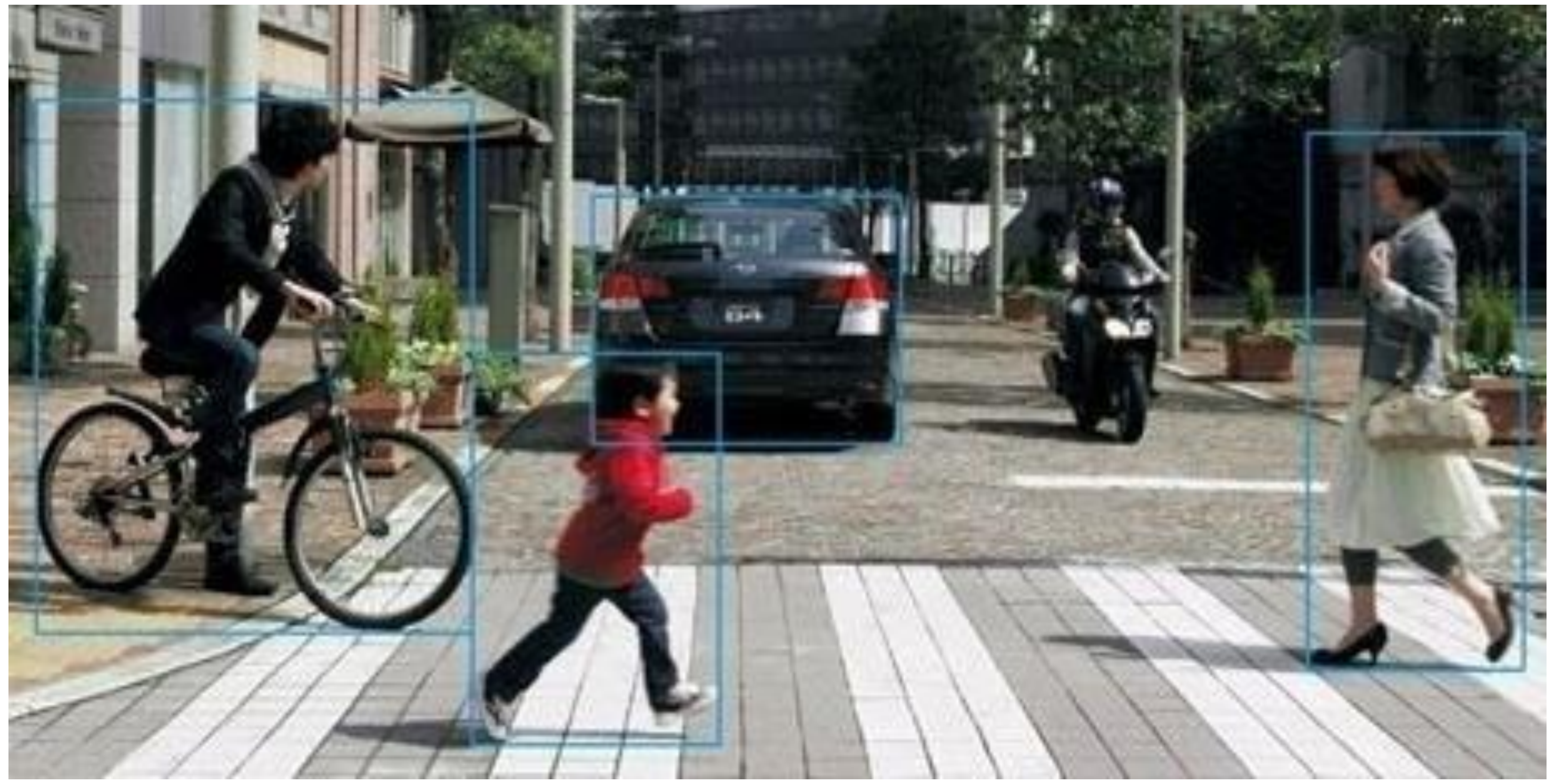
Right: Packaging has to go from 2-dim to 3-dim arrangements.

Experiment: “Human Brain vs. Computer”

Task 1: Mathematics

$$\sqrt{2} = ?$$

Task 2: Image recognition



Traditional silicon scaling ended
New types of problems gain interest

Explore new functionalities, More than Moore
Explore new computing paradigms

- approximate computing
- large parallel data streams

Previous slide:

This shows a simple theoretical experiment, where we want to compare the performance of the human brain with a computer based on two different tasks.

In task 1, both candidates have to calculate the square root of 2 as fast as possible.

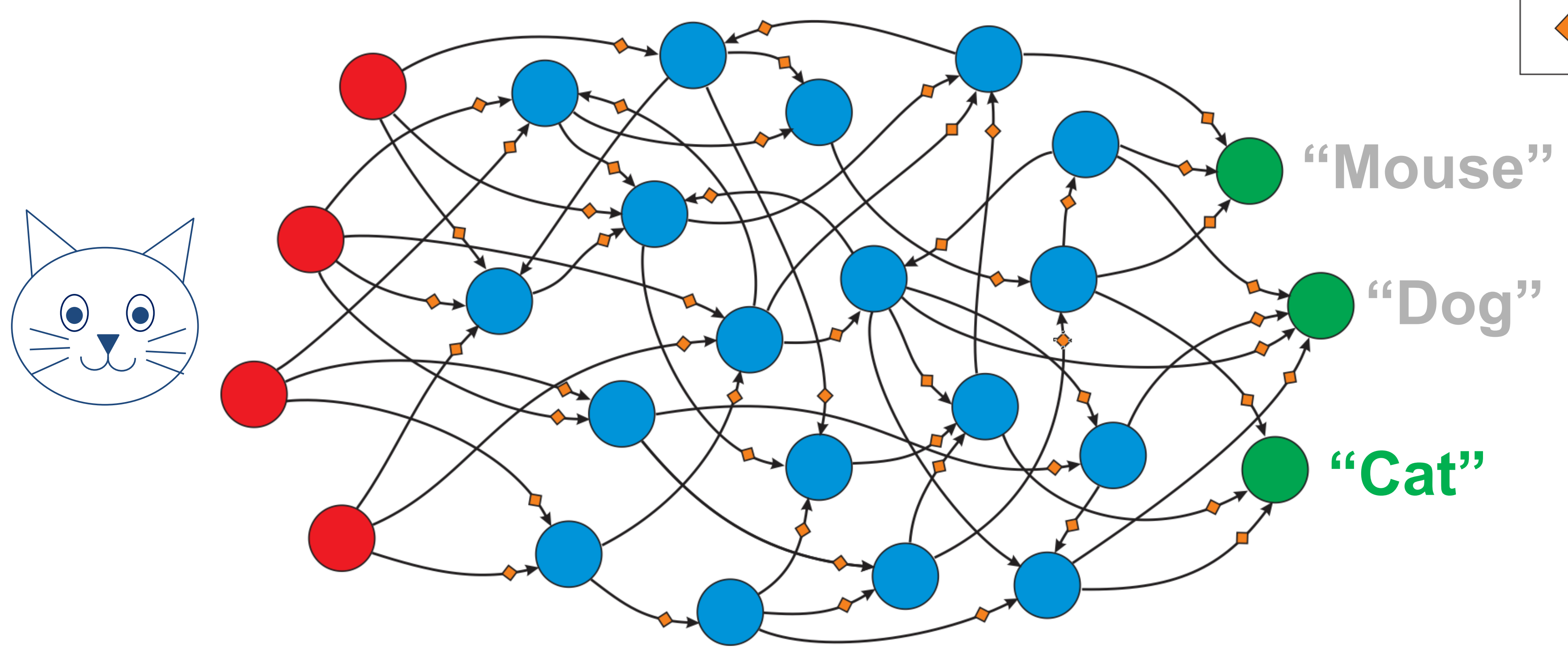
In task 2 both candidates have to interpret a scene.

The point is that that task requirements in task 2 are very different!

For example, a single noisy pixel (or noisy compute process) is less relevant. Handling of large data streams is more important.

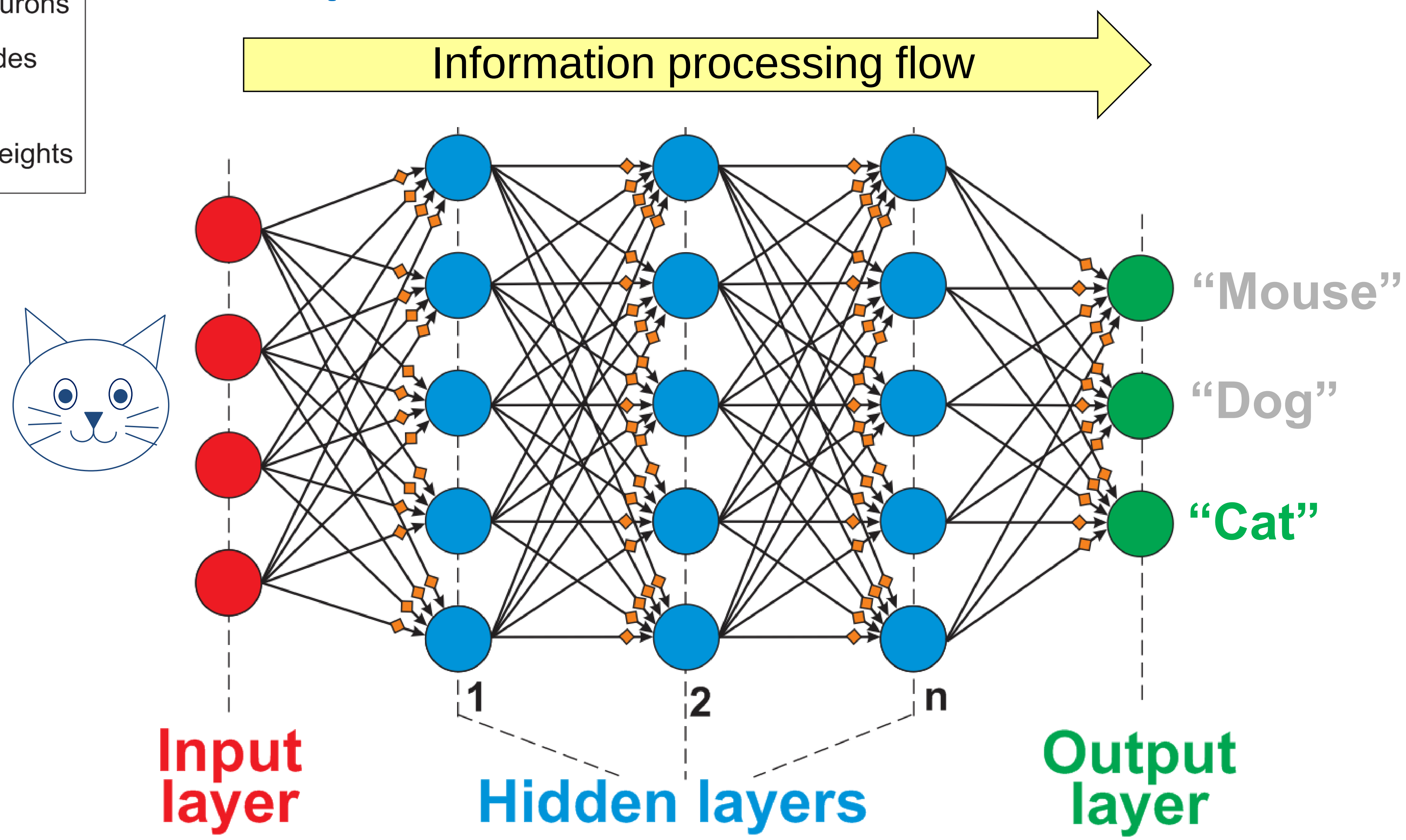
Review Brain inspired computing:

Brain-like Neural network:



- 🟥 Input neurons
- 🟦 Hidden neurons
- 🟩 Output nodes
- ◆ Synaptic weights

Deep Artificial Neural Network:



Simplify

- Omni-directional signal flow
- Asynchronous pulse signals
- Information encoded in signal timing/Spiking Neural Networks

➔ Difficult to implement efficiently on standard computer hardware

- Feed-forward sequential processing
- Information encoded in signal amplitude
- Neuron activation: Accumulate + Threshold
- **Training: Backpropagation Algorithm**

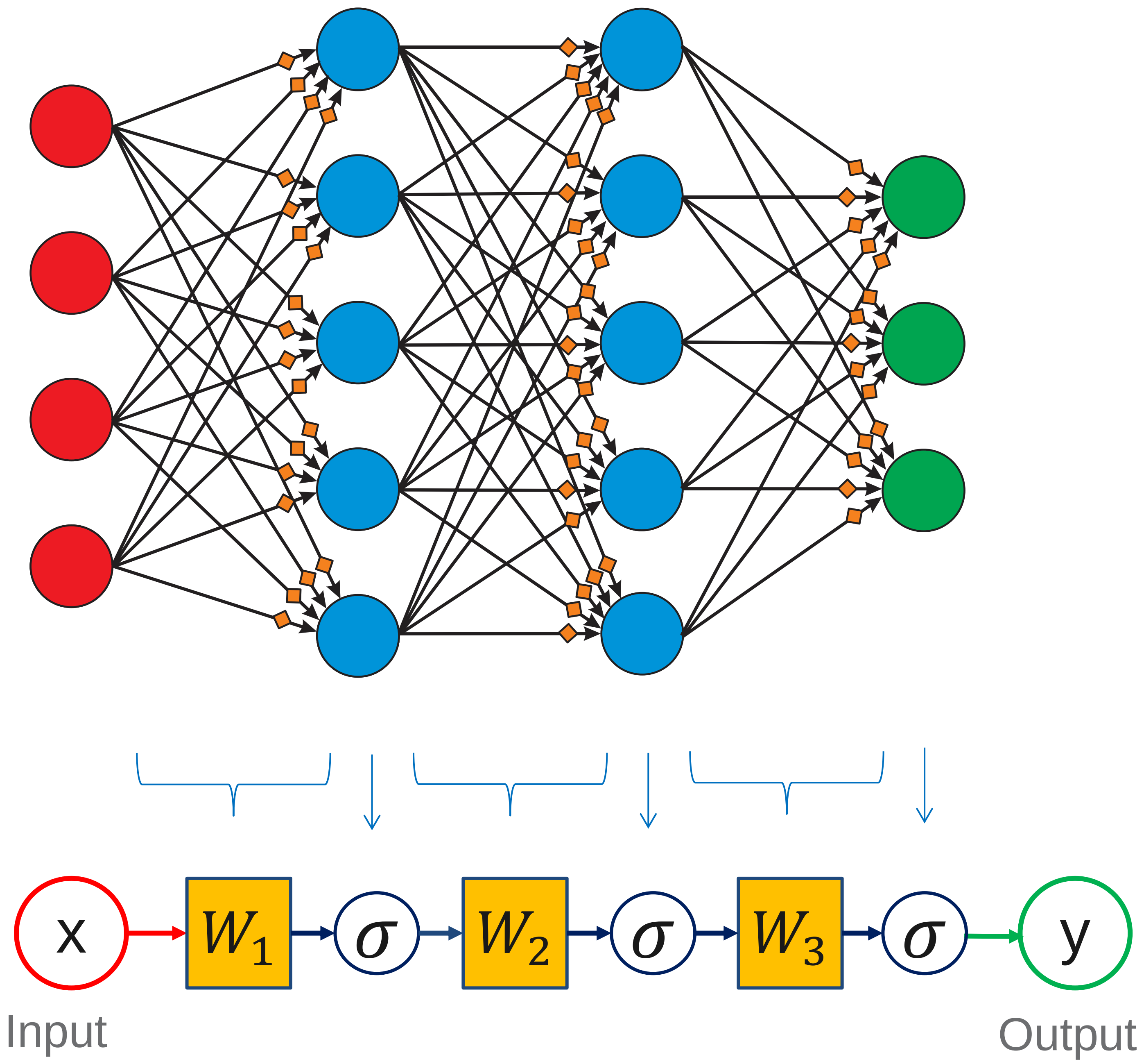
IBM slide

Previous slide:

Standard comparison of a few differences Brain vs ANN. Not shown in class.

Review: Training with Backpropagation algorithm

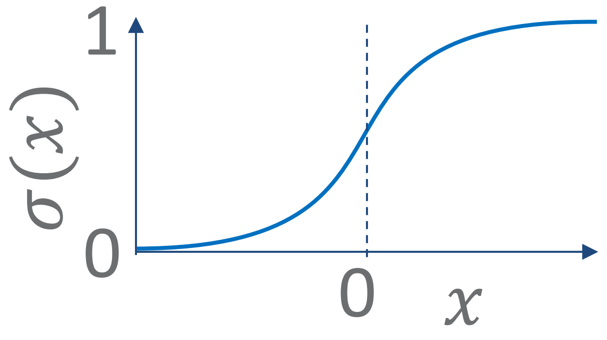
Neural net as chain of vector operations:



⟶ : Signal vector

W_n : Synaptic weight matrix

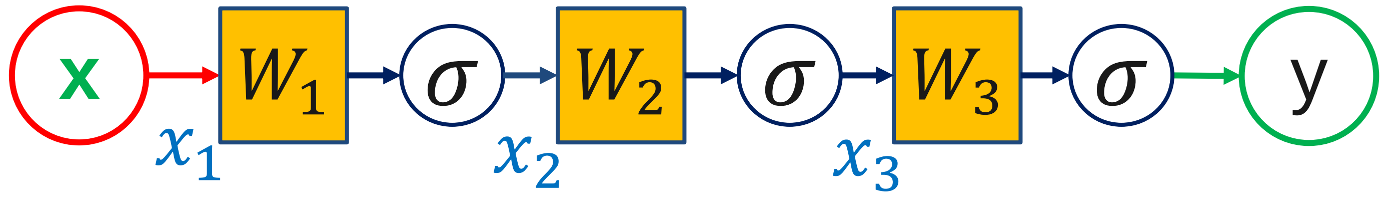
σ : Per-element neural activation function (sigmoid)



Backpropagation algorithm:

For many training cases **x** with target response **t**:

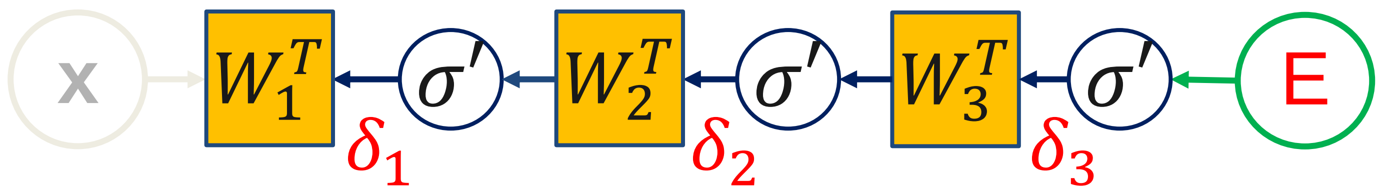
1. Forward Propagate:



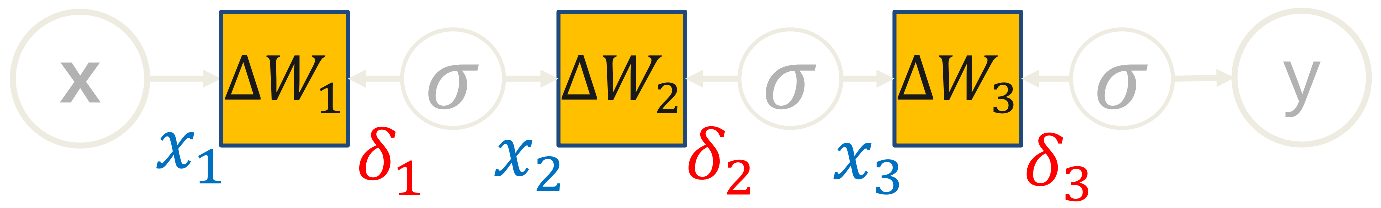
2. Determine output error:

$$\boxed{(y-t)^2} \rightarrow \text{E}$$

3. Backward Propagate: Determine neuron input influence δ on error **E**:



4. Adjust the active weights, proportional to their influence on the error: $\Delta W = -\eta\, \boldsymbol{x} \otimes \boldsymbol{\delta}$



IBM slide

Previous slide: not shown in class

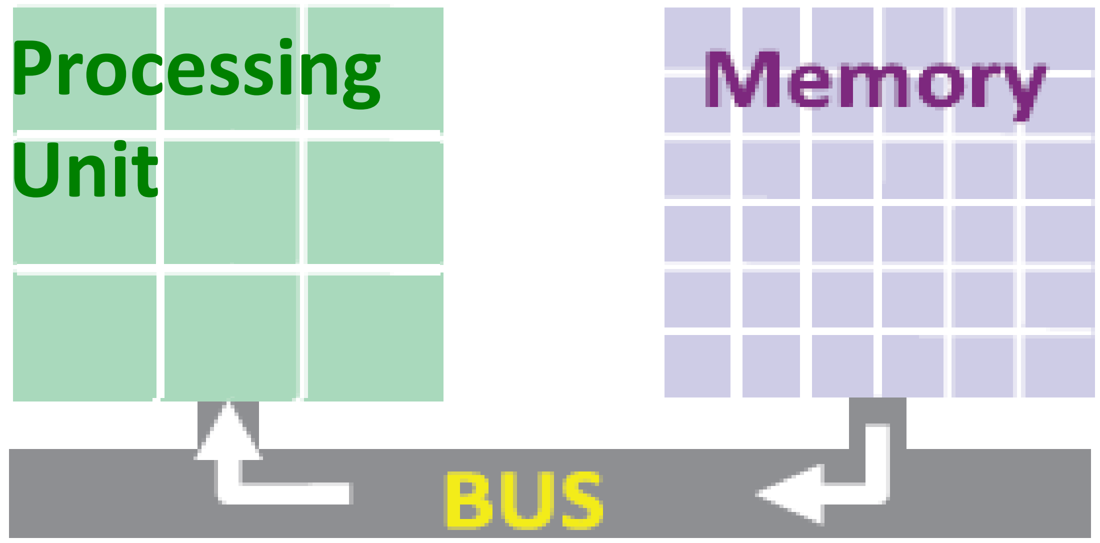
Backpropagation involves

- multiple Matrix multiplications (weight matrix per layer)
- Update of the matrix elements (learning rule)

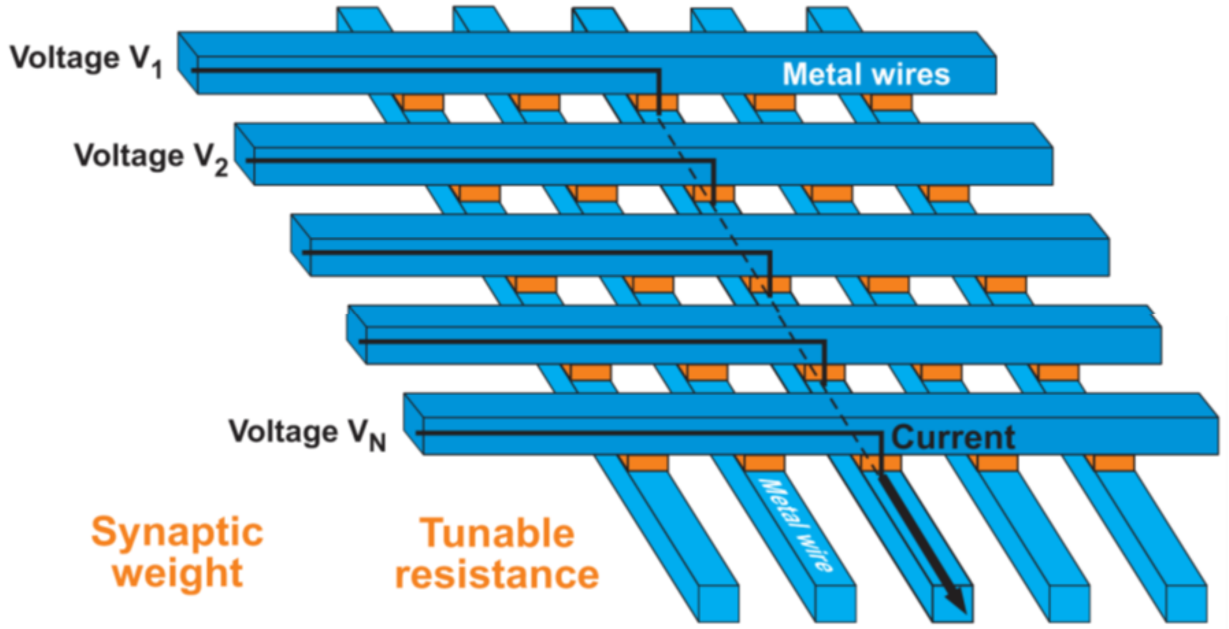
Analog signal processing for scalability

- **Limiting factors of von Neumann architecture**
  - Memory access
  - Sequential operations
  - Digital signal processing

- **Overcome by**
  - In-memory computing
  - Parallel operations
  - Analog signal processing



Compute effort ~O(#Neurons^2)



Compute effort ~O(N)

Electrical (and optical solutions) are viable candidates

IBM slide

Previous slide:

For these kind of matrix operations we should exploit new computing concepts.
The traditional von-Neumann paradigm is limited by signal flow and bad scaling as a the number of neurons per layer increases.

Training of Artificial Neural Networks: many matrix multiplications

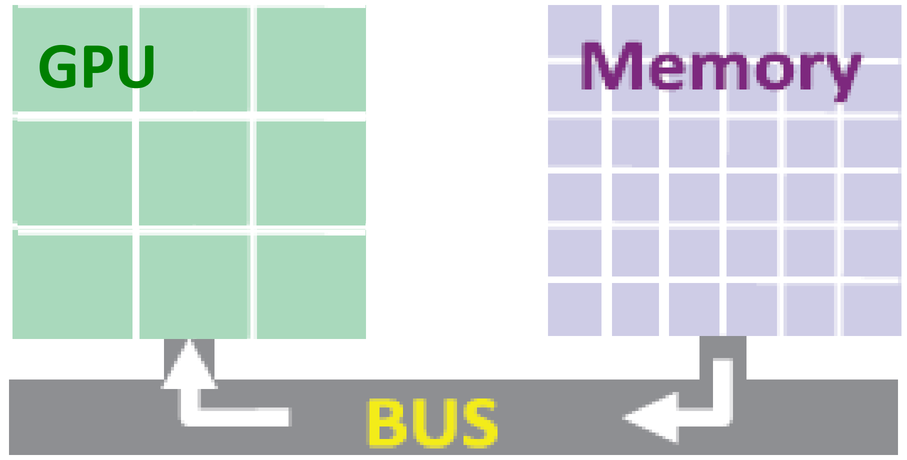
Training by Backpropagation Method:

- **Processing dominated by many large matrix operations**
  - Forward Propagation: $W_{1,2..}$
  - Backward Propagation: $W^T_{1,2..}$
  - Weight Update: $\Delta W_{1,2..}$

Scale ∝ N^2

Neurons/layer

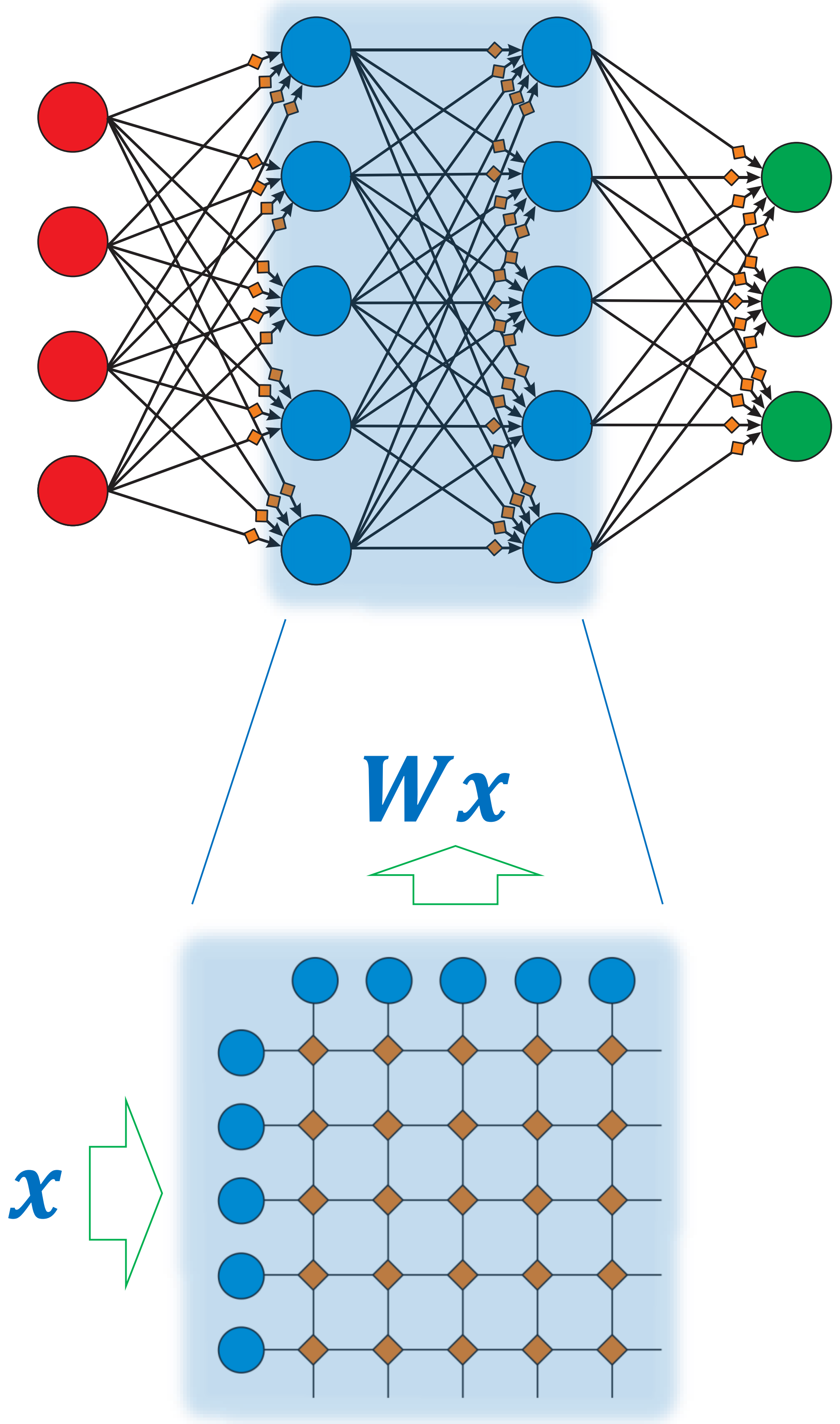
- **Ineffficient on standard Von Neumann systems**:
  - (Mostly) Serial processing
  - Low computation to IO ratio → Memory bottleneck



For fast and efficient neural network data processing:

- Fully parallel processing
- Tight integration of processing and memory
- Analog signal processing

Crossbar arrays
- Electrical
- Optical



📖 G. W. Burr et al., “Tech. Dig. - Int. Electron Devices Meet. IEDM, vol. 2016–Febru, no. 408, p. 4.4.1-4.4.4, 2016.
📖 T. Gokmen and Y. Vlasov, Front. Neurosci., vol. 10, no. JUL, pp. 1–13, 2016.

IBM slide

Previous slide:
Top:
In the week on BackPropagation we already discussed the scaling:

The algorithm scale proportional to the number of weights.
Assume that we have many layers and N neurons per layer. Then the scaling is O(N^2).

This is true for each of the three steps: forward pass, backward pass, weight update.

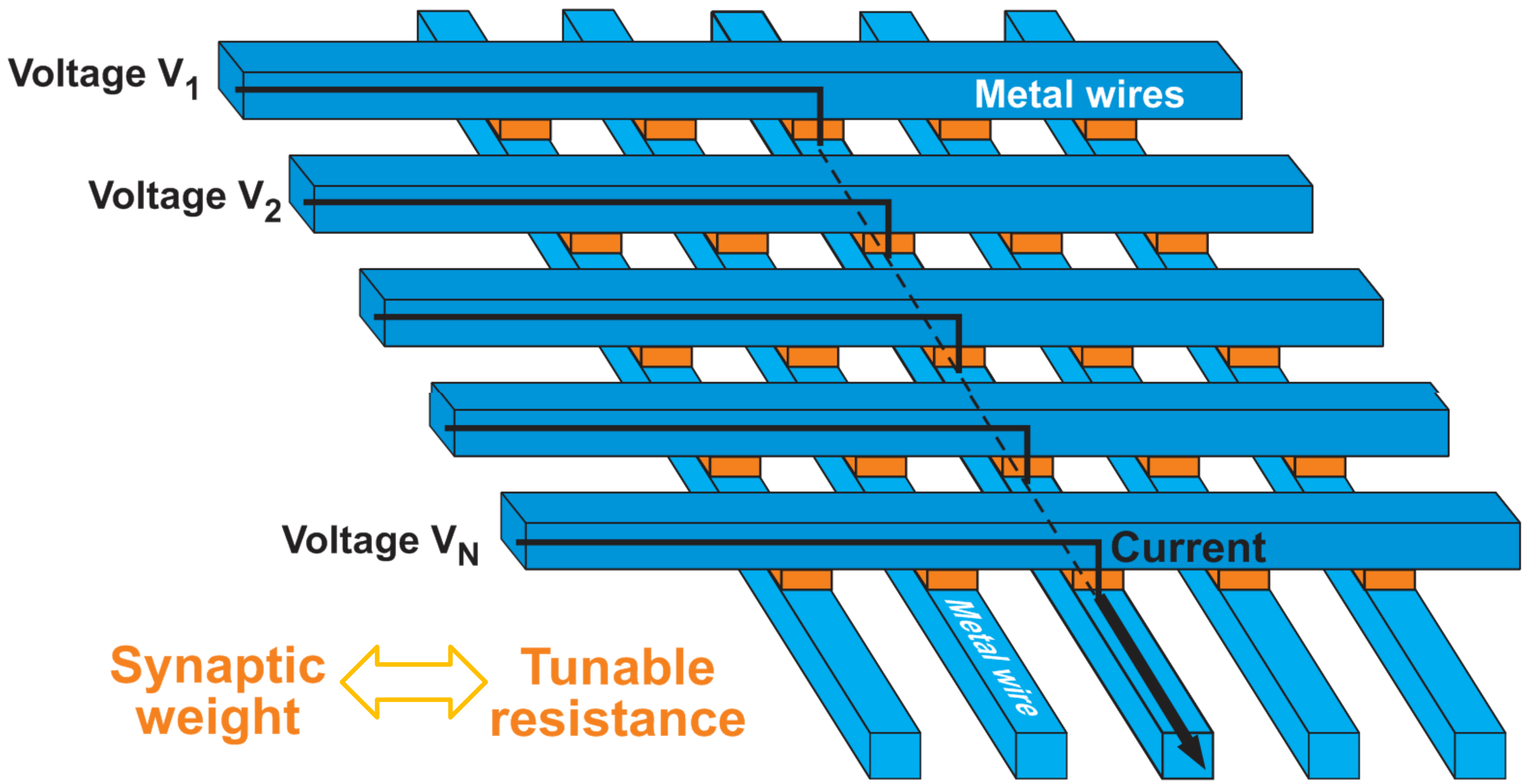
Bottom:
With analog implementation of the matrix multiplication we should be able to achieve a better scaling:

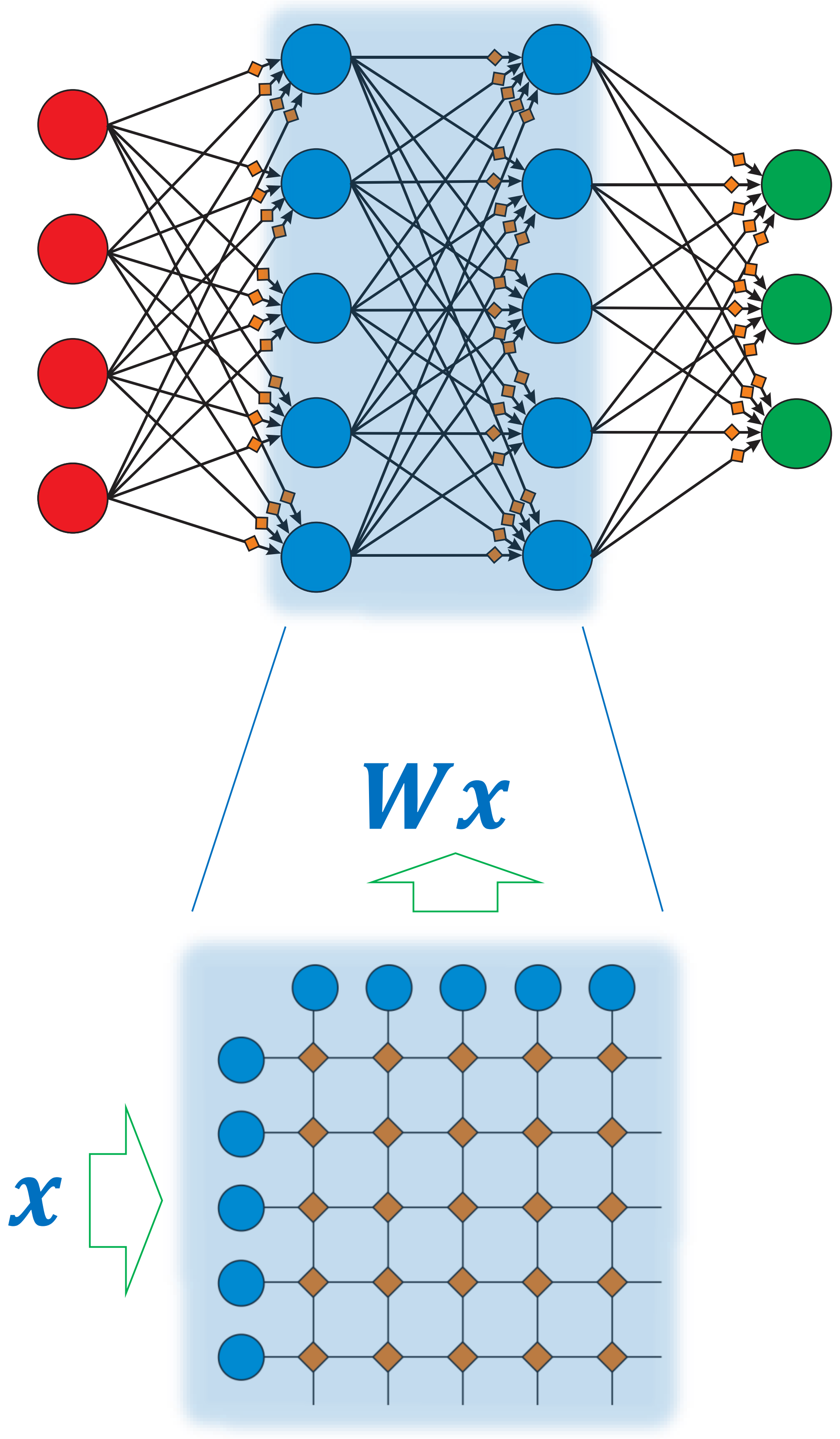
Forward pass: O(1)
Backward pass: O(1)
Weight update: O(N^2) ????

Efficient training of Deep Artificial Neural Networks:

Matrix multiplication = Ohms law: $V = R\,I \;\rightarrow I = V/R$

Electrical crossbar array:





Input signal $x_j = V_j$ voltage of neuron j

Weight $w_{ij} = 1/R_{ij}$ resistor at crossing

Output $I_i = \sum_j \frac{V_j}{R_{ij}} = \sum_j w_{ij}\, x_j$ current into neuron i

Images: IBM

Previous slide:
Each blue bar is a perfect conductor. The red crossing points are tunable resistors that play the role of synaptic weights.

From Ohm’s law follows that the current from neuron j to neuron i is $I_{ij} = V_j/R_{ij}$.

Kirchhoff’s law (conservation of current) gives the final summation equation.

Tunable weights via Memristive Devices

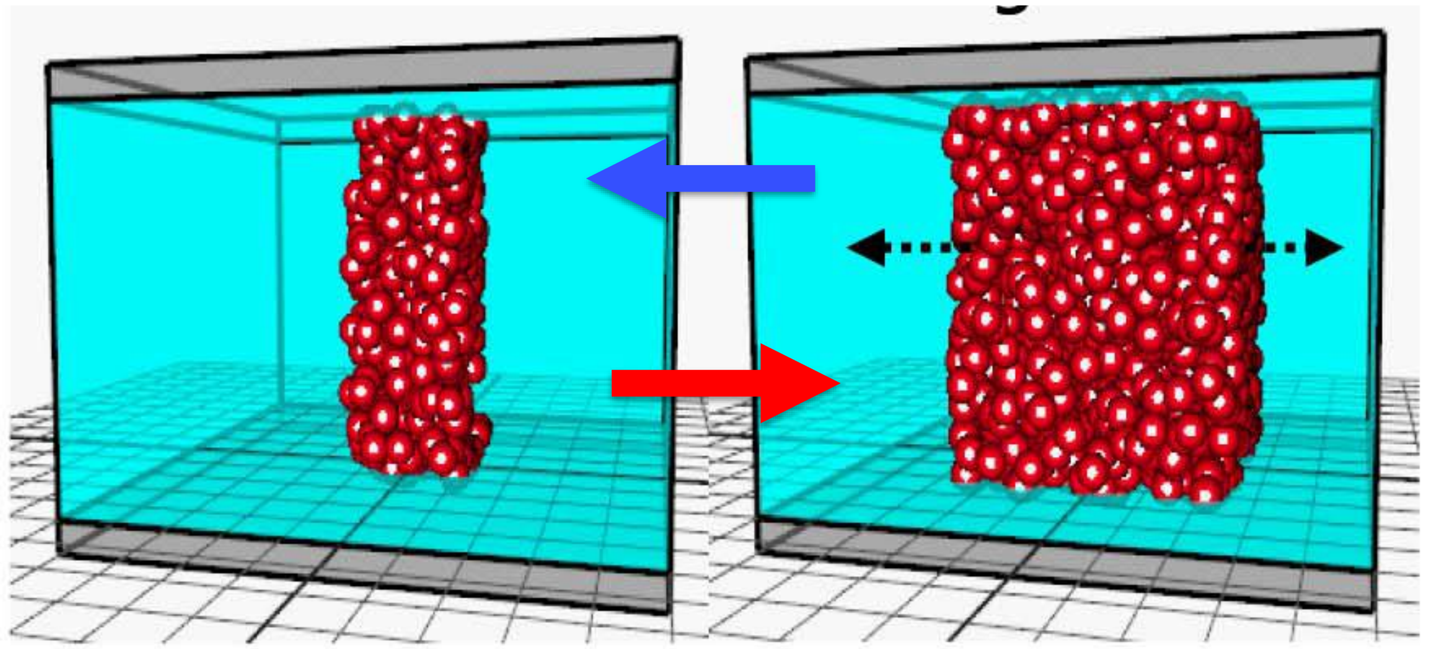
‘memory of resistance’ = ‘memristor’

Understanding the mechanism

large neg. pulse

IBM MO_3+HfO_2

Continuous & symmetric change of R



Woo et al. IEEE Electr. Dev. Lett. 38, 9 (2017)

large pulse> pos. threshold

- Resistance depends on molecular configuration
- Resistance increase or decreases with voltage pulses **above threshold values**
- Resistance keeps memory

Images: IBM

Previous slide:

Memristive material studied by IBM.

The basic function arises from the following principle.

The material in light blue is an electrical insulator (dielectric material). However, with a first strong voltage pulse one can create an initial breakdown in the material. This leads to a short-cut illustrated by a thin red column of molecules in a conducting state (lower left). Now the material is now longer insulating, but has a finite resistance.

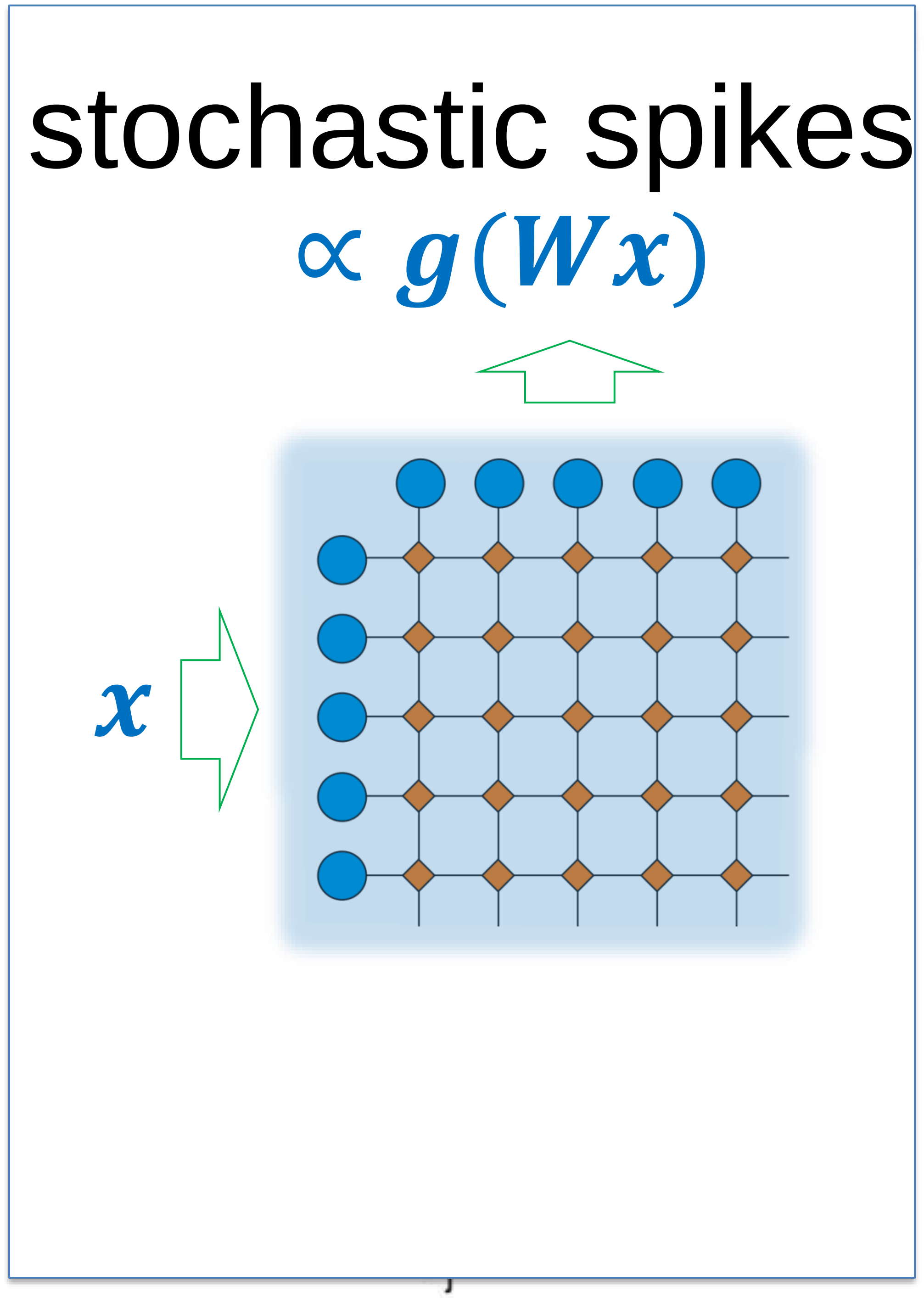
With an additional medium-sized **positive voltage pulse** (red), the column of conducting molecules can be made thicker so that the resistance decreases (lower right).

With a later medium-sized **negative voltage pulse** (blue), one can return to the initial configuration (lower left).

Weak currents and weak voltage pulses have no effect. Hence the material keeps its configuration and resistance for a long time. It has a **‘Memory of Resistance’ → Memristor.**

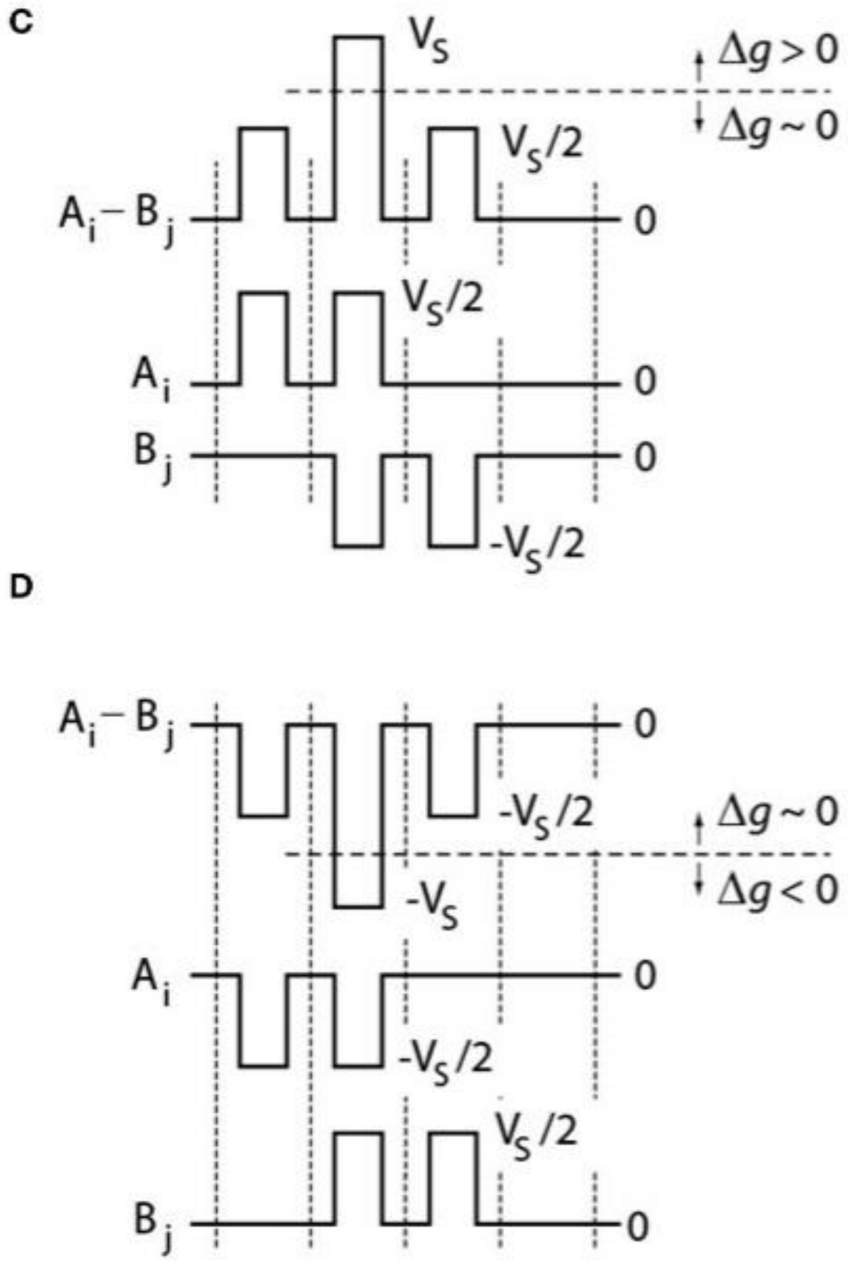
Efficient training of Deep Artificial Neural Networks: spiking network

Weight update order 1 !

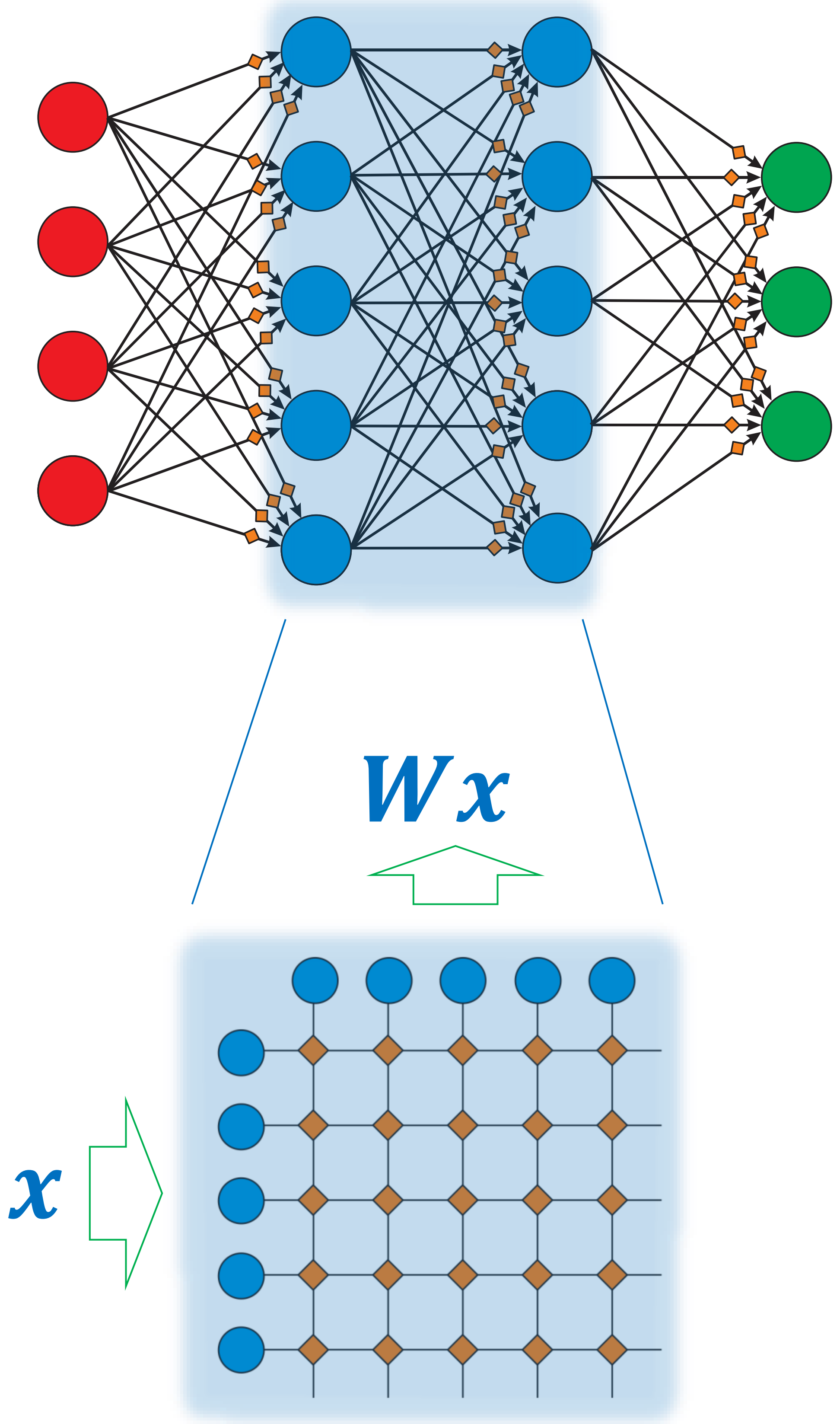


stochastic spikes

$\propto g(\boldsymbol{Wx})$



- Local Hebbian Learning rule
- Spike coding (SNN)



- Fully parallel processing

- Pulse coding

- Stochastic Poisson Process

Crossbar arrays

- Electrical

Gokman and Vlasov, Acceleration of Deep Neural Networks with Resistive Cross-Point Devices Frontiers, 2016

Previous slide:

We now image the following coding principle (not yet implemented in hardware, but proposed some years ago).

Each presynaptic neurons sends voltage pulses (‘spikes’ of finite width) at random moments in time (Poisson process).

Each postsynaptic neuron sends voltage pulses (‘spikes’ of finite width) at random moments in time (an independent Poisson process).

The amplitude of the single pulse is such that it does not reach the switching amplitude of the memristive material. But if two pulses coincide, then it reaches the threshold and increases the weight (decreases the resistance).

Thus we have a proposition to implement a local (two-factor) Hebbian learning rule in hardware. And, unexpectedly, we need spike coding for this implementation scheme!

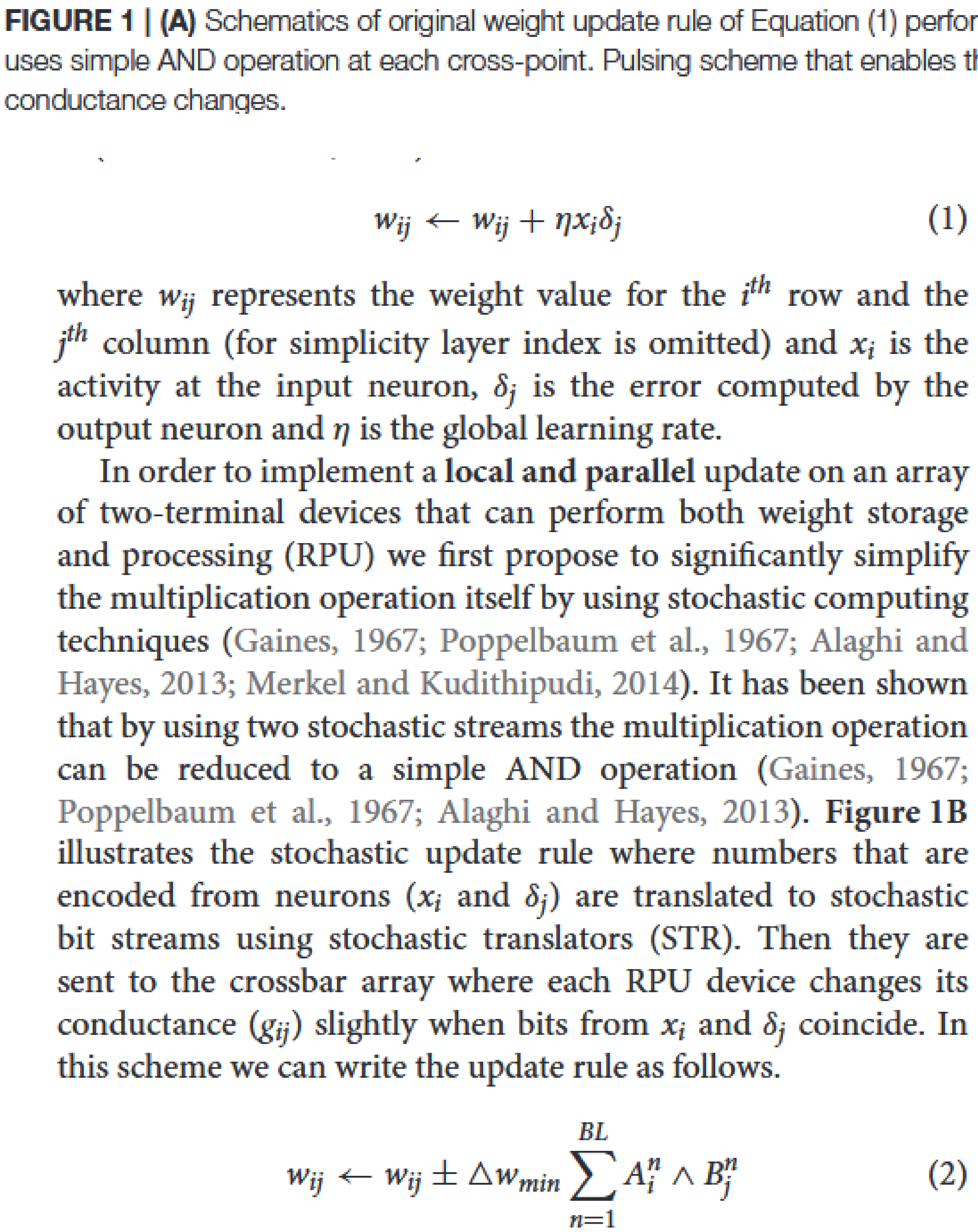


FIGURE 1 | (A) Schematics of original weight update rule of Equation (1) performed at each cross-point. **(B)** Schematics of stochastic update rule of Equation (2) that uses simple AND operation at each cross-point. Pulsing scheme that enables the implementation of stochastic updates rule by RPU devices for **(C)** up and **(D)** down conductance changes.

$$w_{ij} \leftarrow w_{ij} + \eta x_i \delta_j \tag{1}$$

where w_{ij} represents the weight value for the i^{th} row and the j^{th} column (for simplicity layer index is omitted) and x_i is the activity at the input neuron, δ_j is the error computed by the output neuron and η is the global learning rate.

In order to implement a **local and parallel** update on an array of two-terminal devices that can perform both weight storage and processing (RPU) we first propose to significantly simplify the multiplication operation itself by using stochastic computing techniques (Gaines, 1967; Poppelbaum et al., 1967; Alaghi and Hayes, 2013; Merkel and Kudithipudi, 2014). It has been shown that by using two stochastic streams the multiplication operation can be reduced to a simple AND operation (Gaines, 1967; Poppelbaum et al., 1967; Alaghi and Hayes, 2013). **Figure 1B** illustrates the stochastic update rule where numbers that are encoded from neurons (x_i and δ_j) are translated to stochastic bit streams using stochastic translators (STR). Then they are sent to the crossbar array where each RPU device changes its conductance (g_{ij}) slightly when bits from x_i and δ_j coincide. In this scheme we can write the update rule as follows.

$$w_{ij} \leftarrow w_{ij} \pm \Delta w_{min} \sum_{n=1}^{BL} A_i^n \wedge B_j^n \tag{2}$$

Gokman and Vlasov, Acceleration of Deep Neural
Networks with Resistive Cross-Point Devices
Frontiers in Neuroscience, 2016

Previous slide:

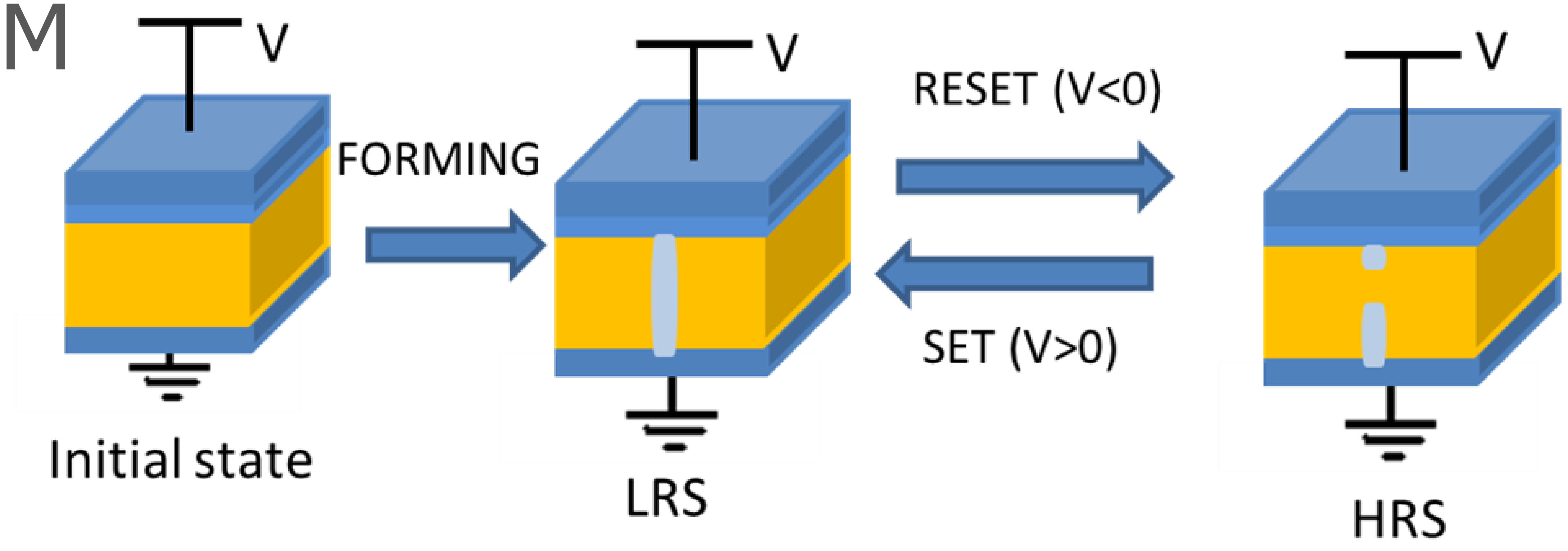
This is a copy of the relevant section of the original publication

The device challenge

- Create breakdown
  ('mild shortcut)
- Make size of breakdown tunable

$w=1/R$

- RRAM



Courtesy E. Vianello

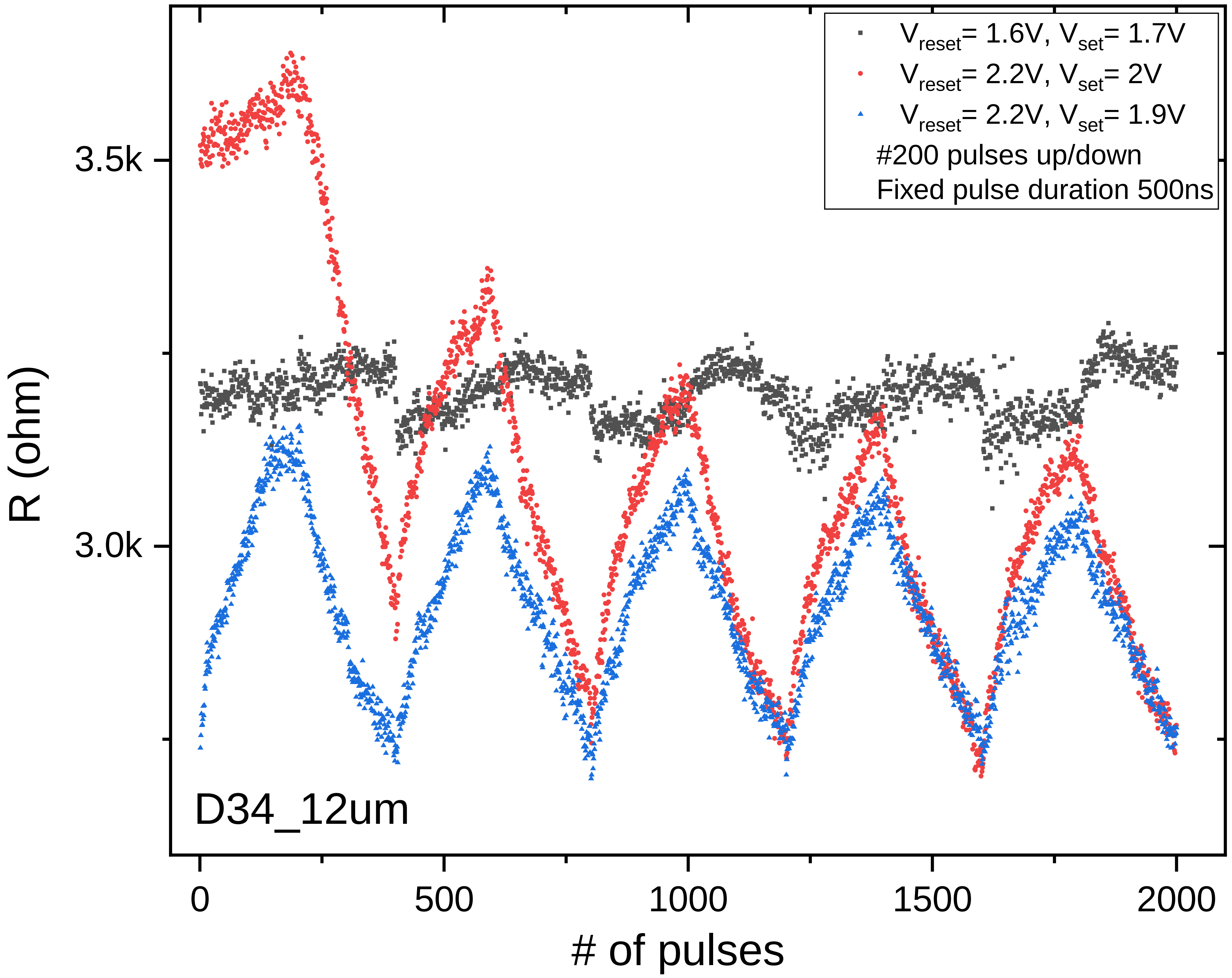
Images: IBM

Previous slide:

Here The material in yellow is an electrical insulator (dielectric material). However, with a first strong voltage pulse one can create an initial breakdown (blue channel) in the material.

The question now is the following: Can we SMOOTHLY TUNE
sith several additional medium-sized **positive voltage pulse** (red), or **negative voltage pulse** (blue), one can return to the initial configuration (lower left).

Changes induced by 200 pulses up (and down) → change the weights of ANN by appropriate pulses



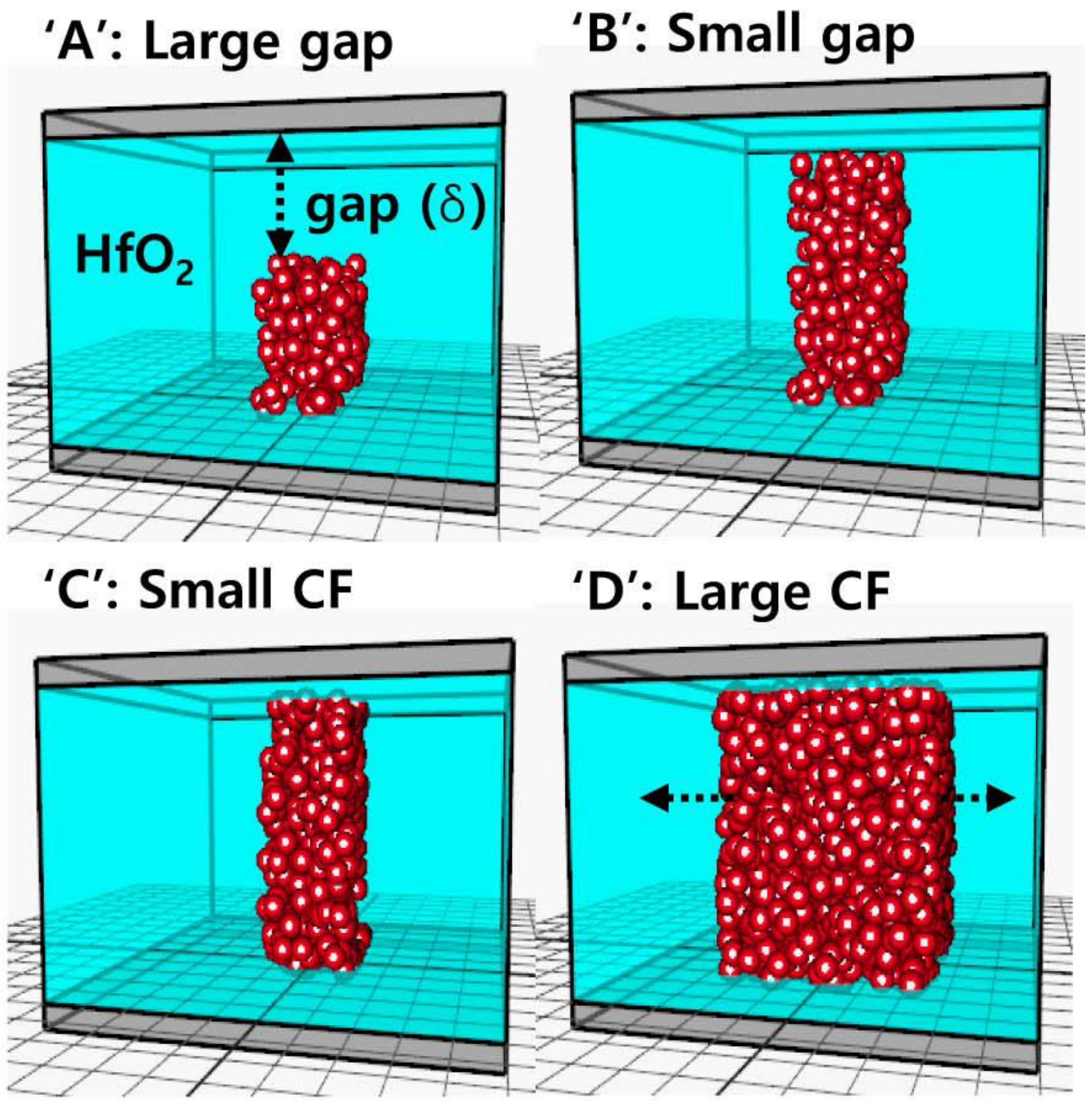
Understanding the mechanism

HfO_2 baseline

Abrupt switching

IBM MO_3+HfO_2

Continuous & symmetric change of G



Woo et al. IEEE Electr. Dev. Lett. 38, 9 (2017)

Experimental demonstration of symmetric and continuous change of G

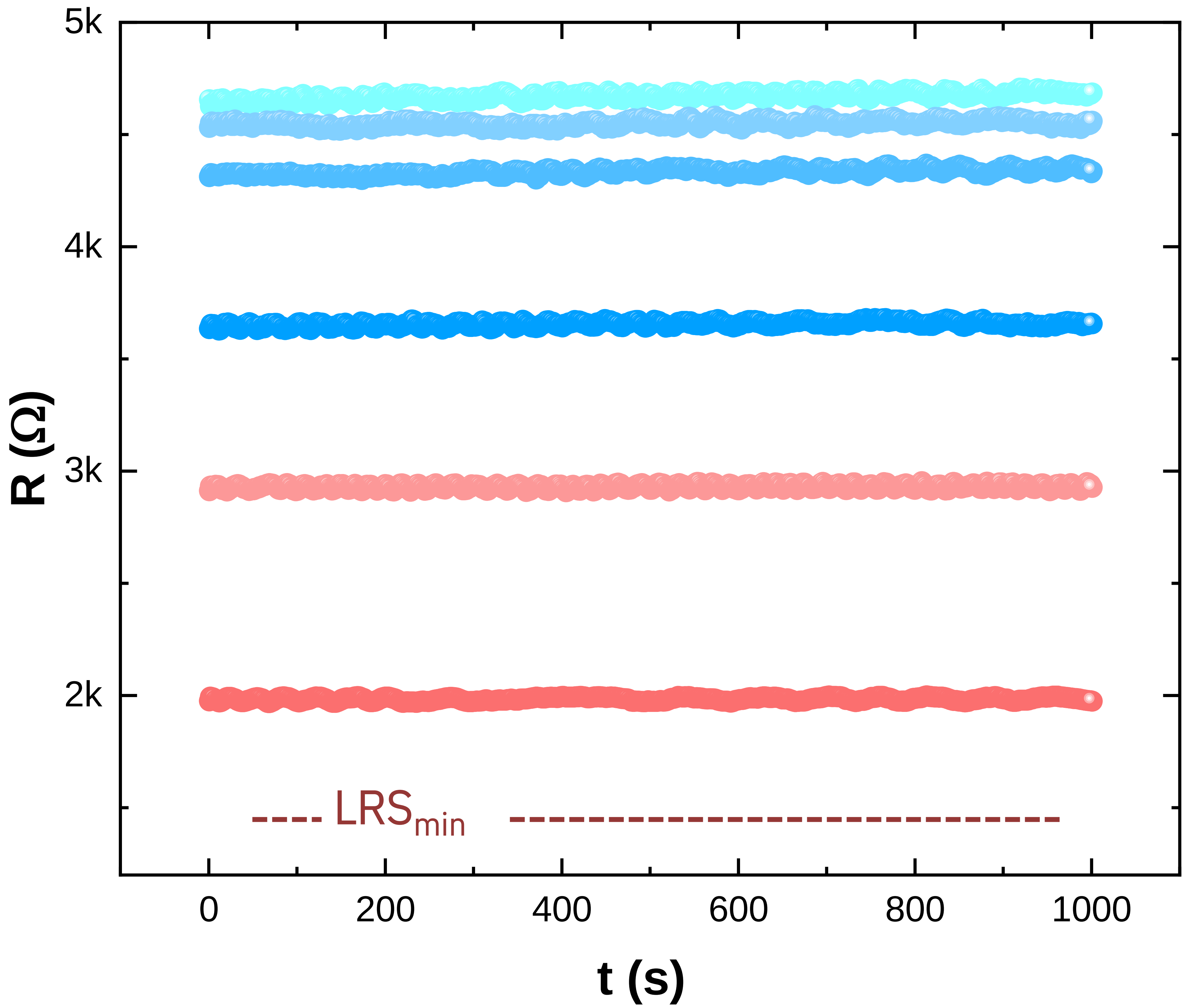
Images: IBM

Previous slide:

Experimental test with the material at the bottom shows that smooth tuning is possible (blue dots). Horizontal axis shows the number of pulses applied. After about 200 pulses the sign is switched so that the resistance goes down again.

Retention of the Resistance Values over time (intermediate values)

1. Negative sweep to put it in Low-Resistance State
2. Read at 0.2V constantly for 1000s
3. Negative + positive cycle to put it in increasing High-Resistance State (HRS)
4. Read at 0.2V constantly for 1000s for each HRS



Intermediate states show no drift up to 1000 s

→We can change values of resistance

→New resistance is reliable over time

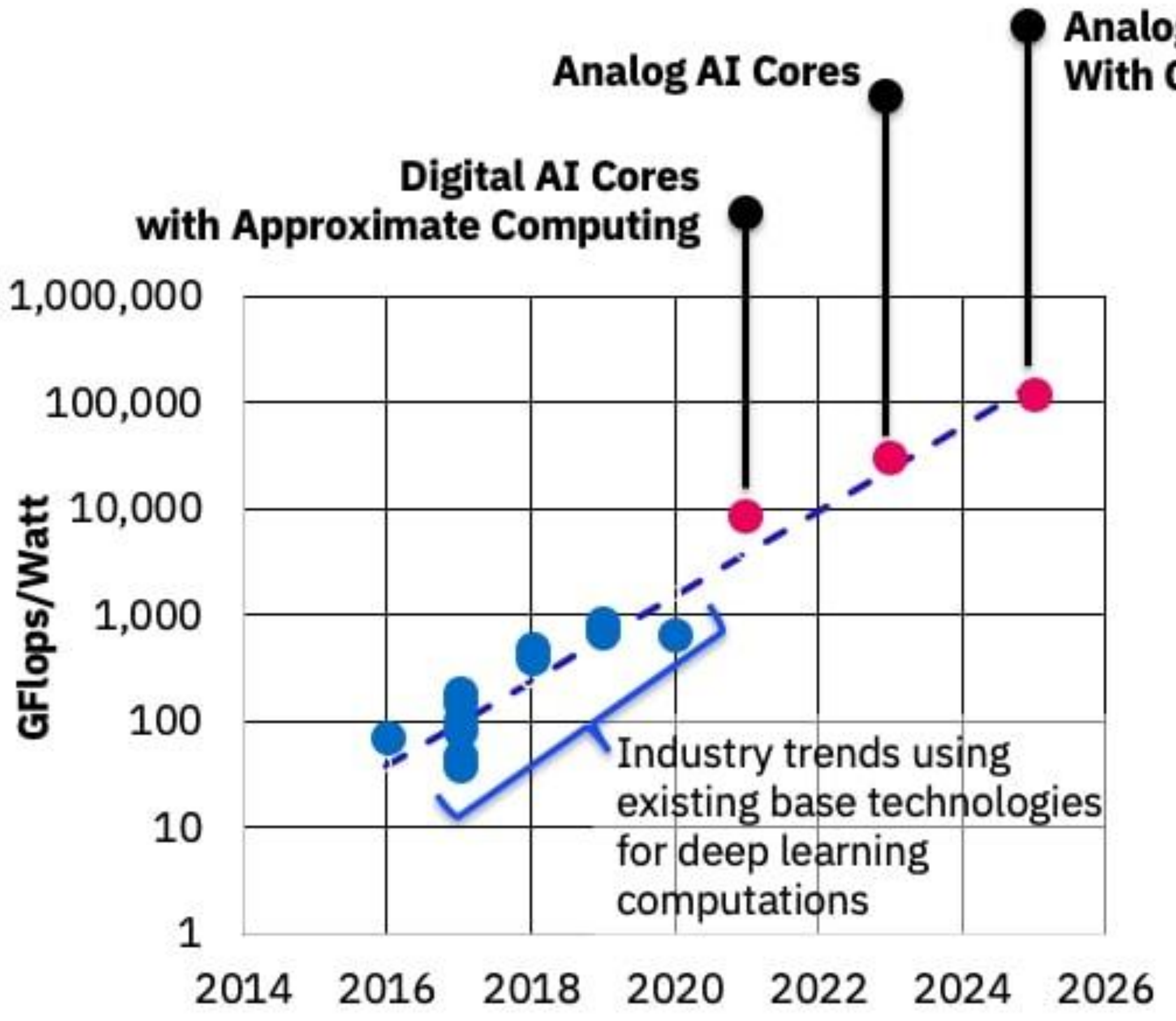
→We can change again

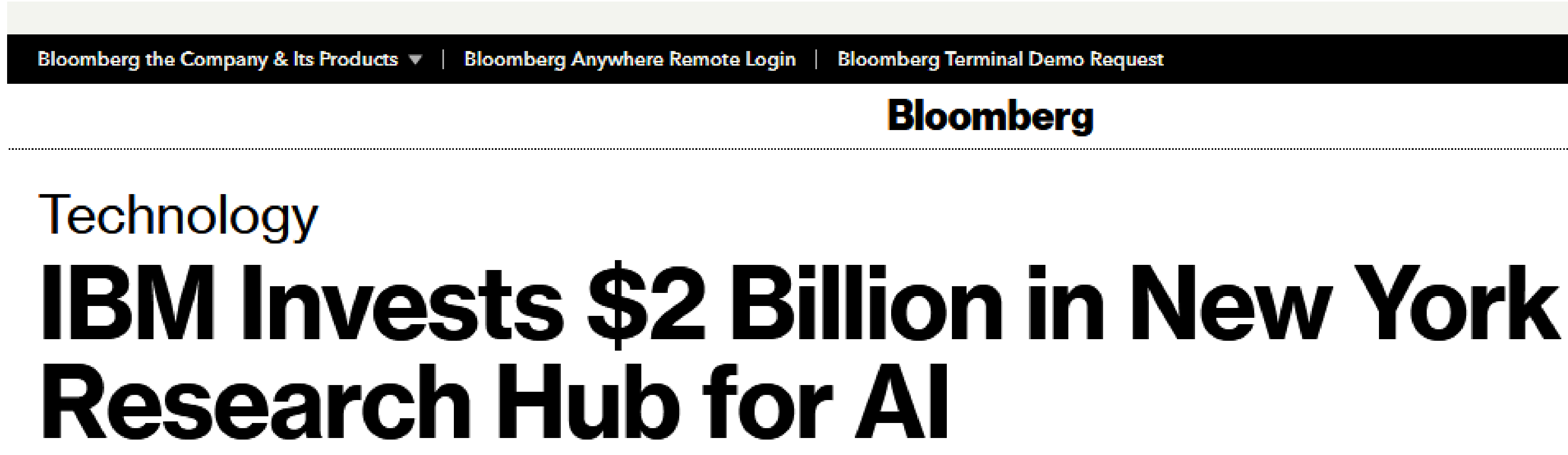
→'Online Learning'

Previous slide:

Moreover, after tuning the resistance remains constant

AI Technology roadmap

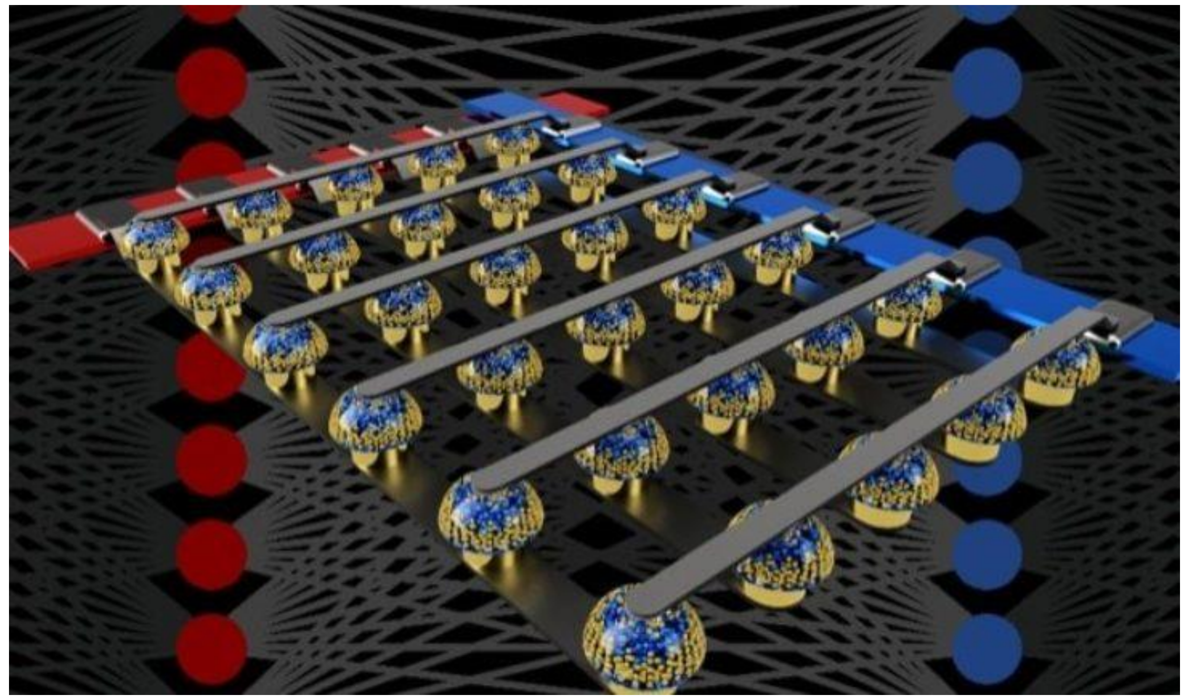



Bloomberg the Company & Its Products ▾ | Bloomberg Anywhere Remote Login | Bloomberg Terminal Demo Request

Bloomberg

Technology

IBM Invests $2 Billion in New York Research Hub for AI



- Analog AI Cores
  - For the synaptic processing function
  - Apply memristive devices: Ohms law & Kirchhoff’s law
  - Parallel forward inference & backward and weight update

compute performance efficiency

New memristive devices required

Analog synaptic processing

	Inference	Training
Resistance	1-100 MΩ	1-100 MΩ
# Levels	100	1000
Weight set / update	To desired level	Symmetric

Previous slide: not shown in class

The road map shows several aspects:

- Traditional scaling has increased not only the compute power, but also reduced the Watt per Flop.
- Traditional scaling expected to come to an end (or may continue, red dots)
- Staying digital, but allowing for approximate computing might give an extra jump in performance
- Going analog would yield a further jump.

The 80-percent question again:

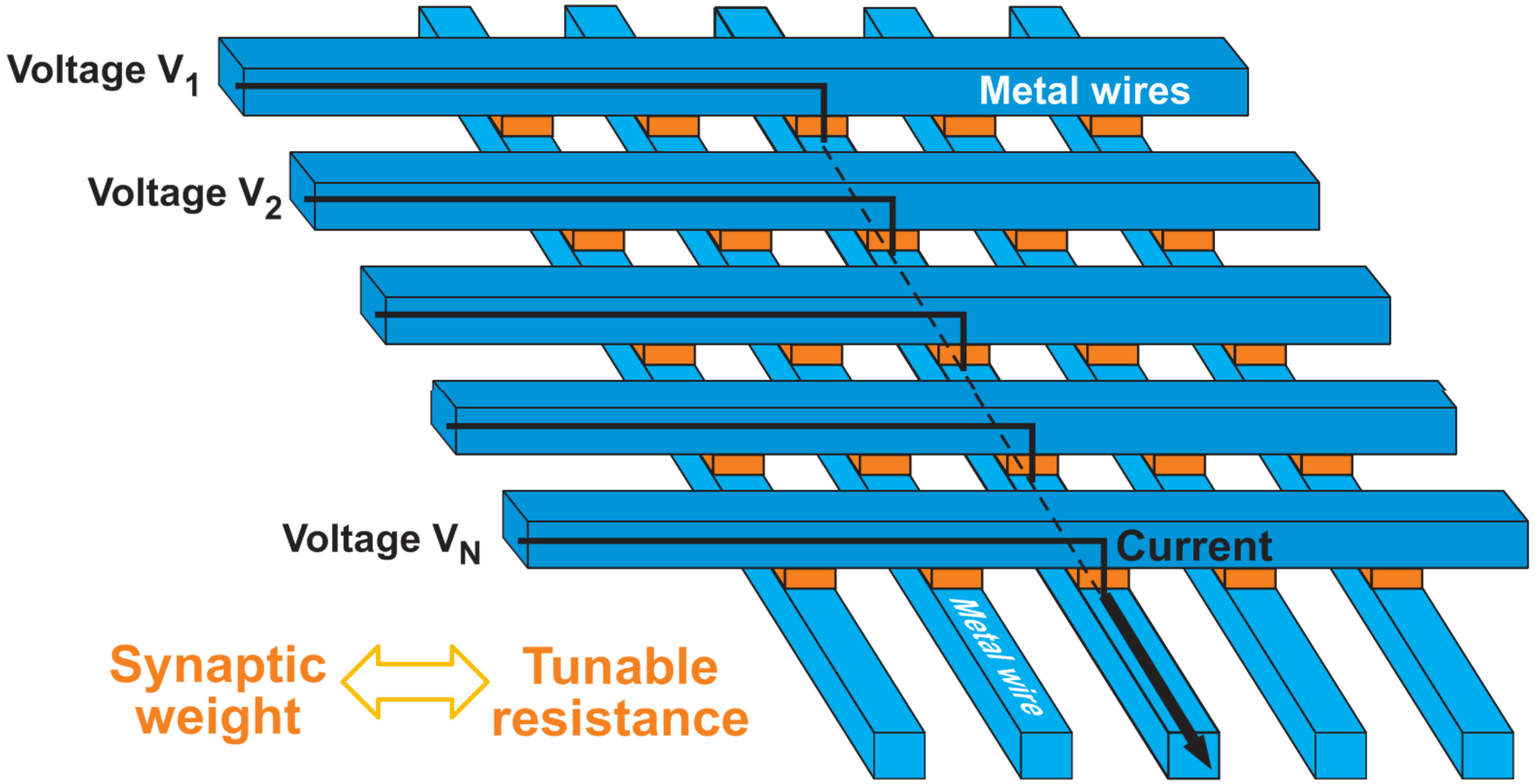
[ ] In this hardware part on memristors, at least 60 percent of the material was new to me.

[ ] for this hardware part, up to here, I have the feeling that I
understood at least 80 percent of the material

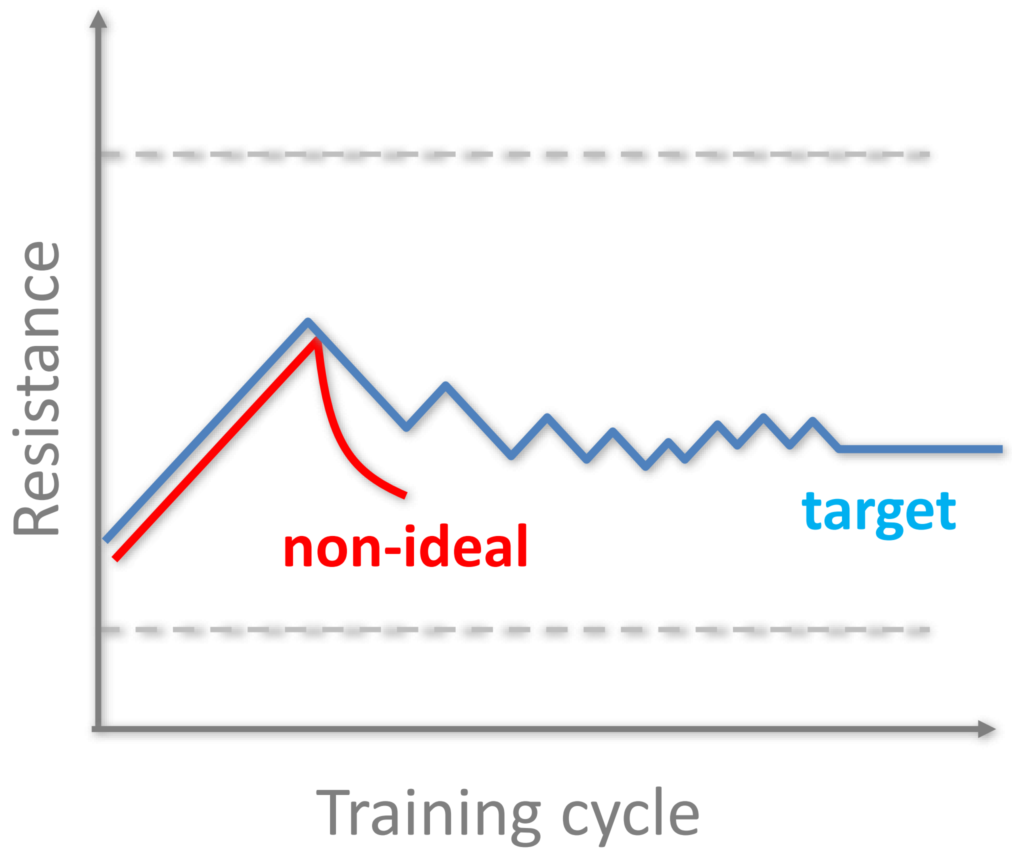
Previous slide:

Analog crossbar arrays: Update for BackProp

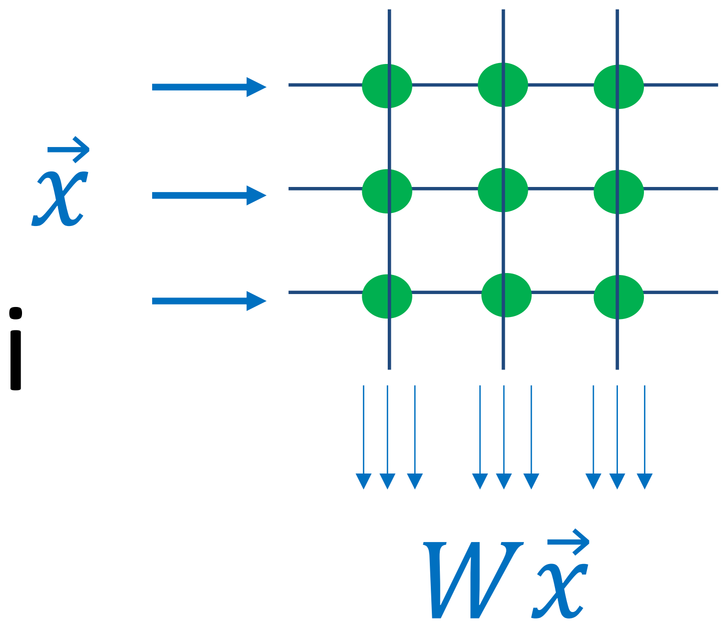
Electrical crossbar array:



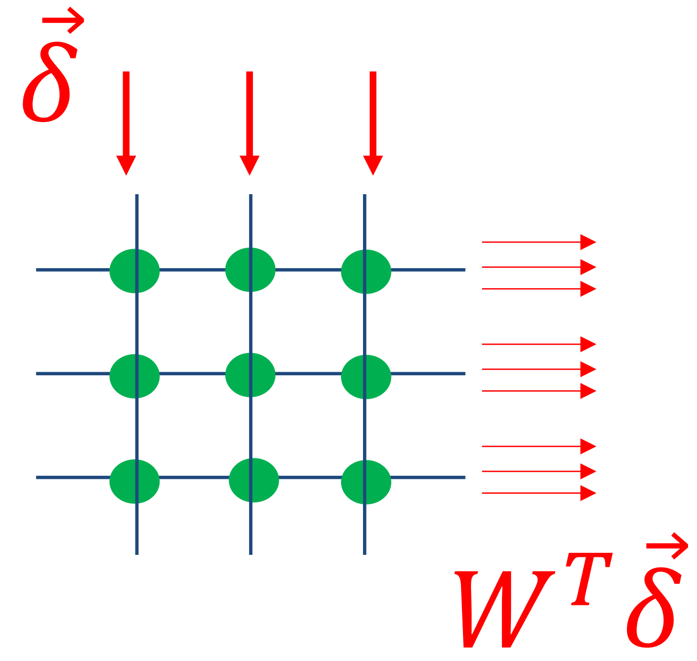
- **Weight update:** proportional to signals on row and column
  - Symmetric increase and decrease of weight
  - >1bit analog levels required
- **Physical challenge**: Identify material systems that meet these requirements



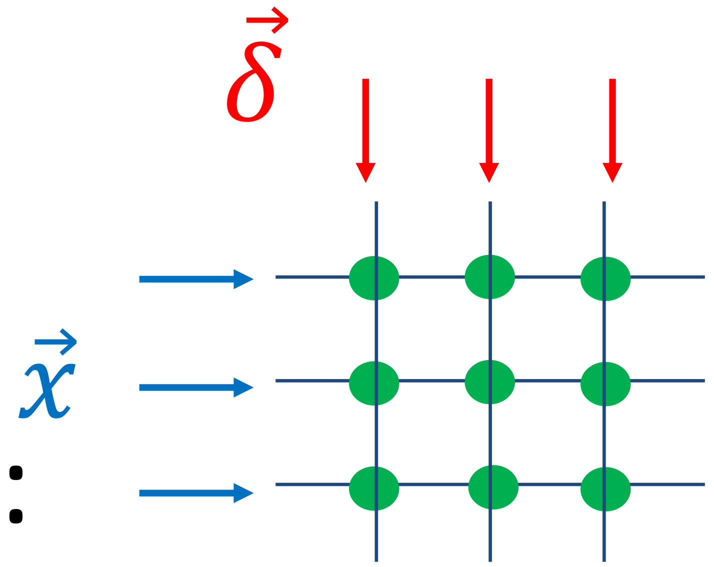
Forward propagation:



Backward propagation:



Weight update:



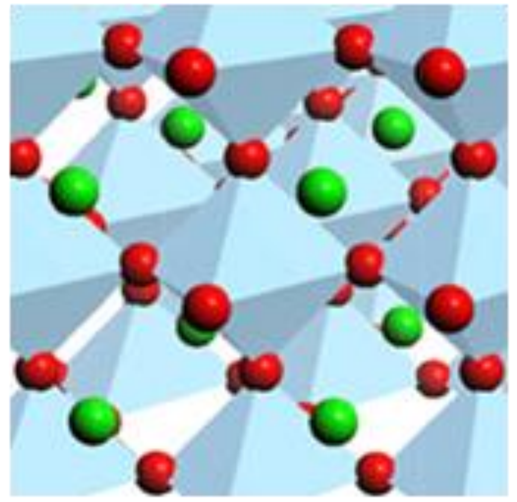
$$\Delta w_{ij} = -\eta \, x_i \, \delta_j$$

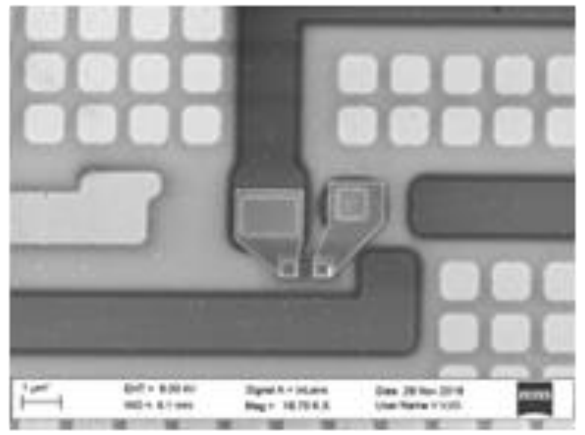
Previous slide: not shown in class.

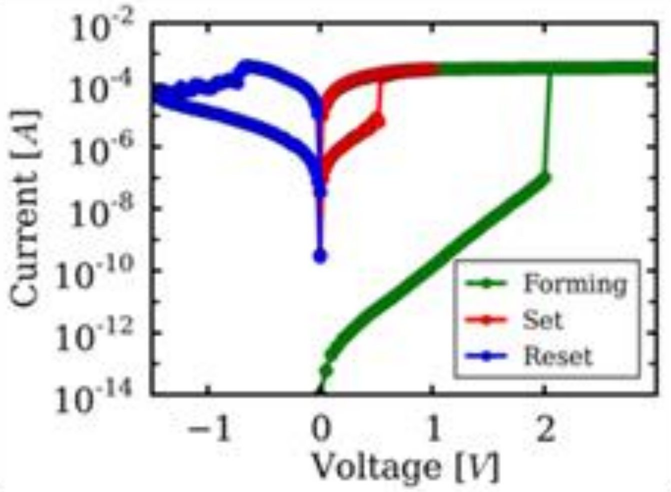
To have an impact, local Hebbian rules are not enough.

But one can also extend these ideas to local implementations of BackPROP.

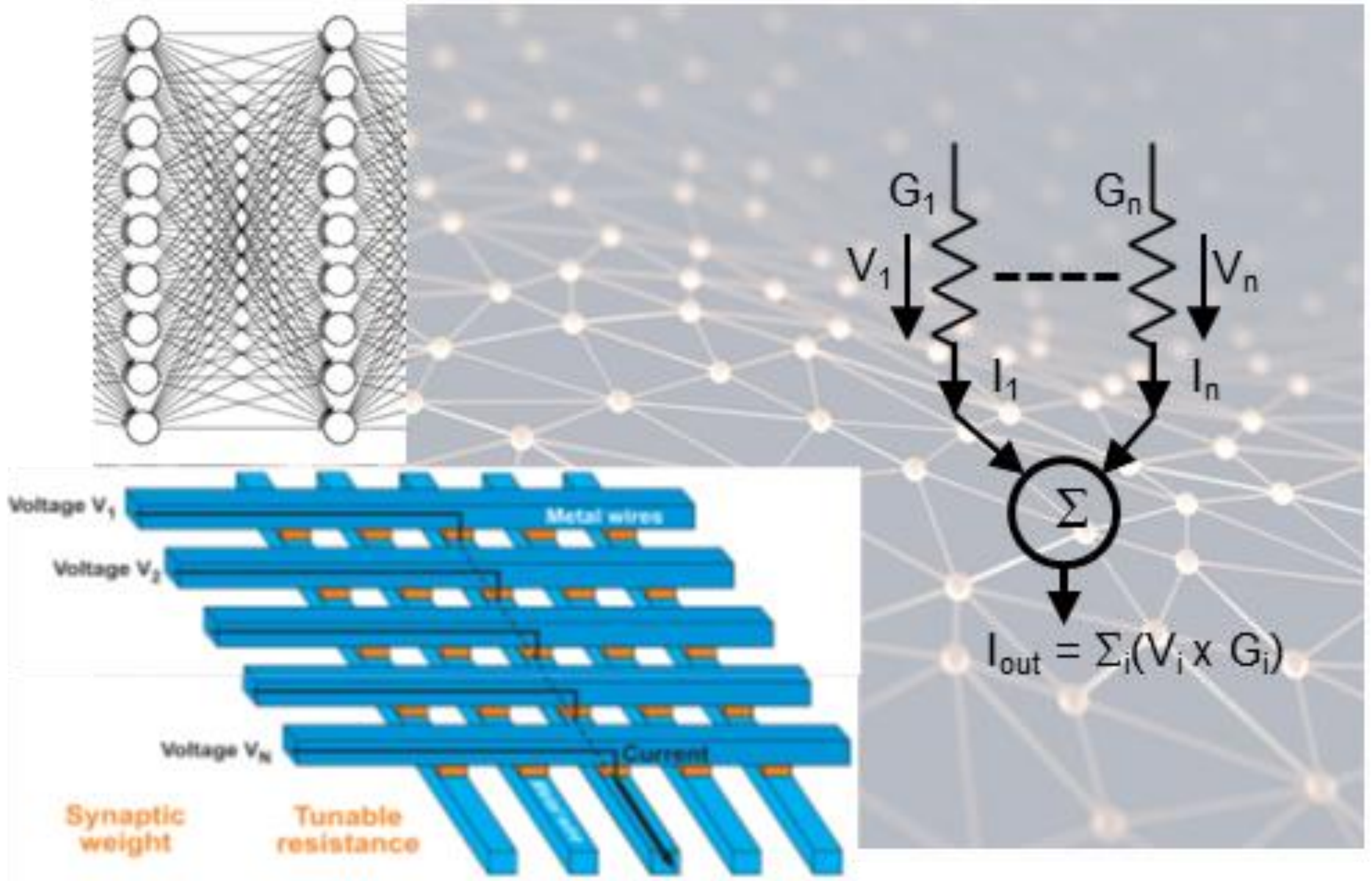
Neuromorphic – local learning rules in hardware



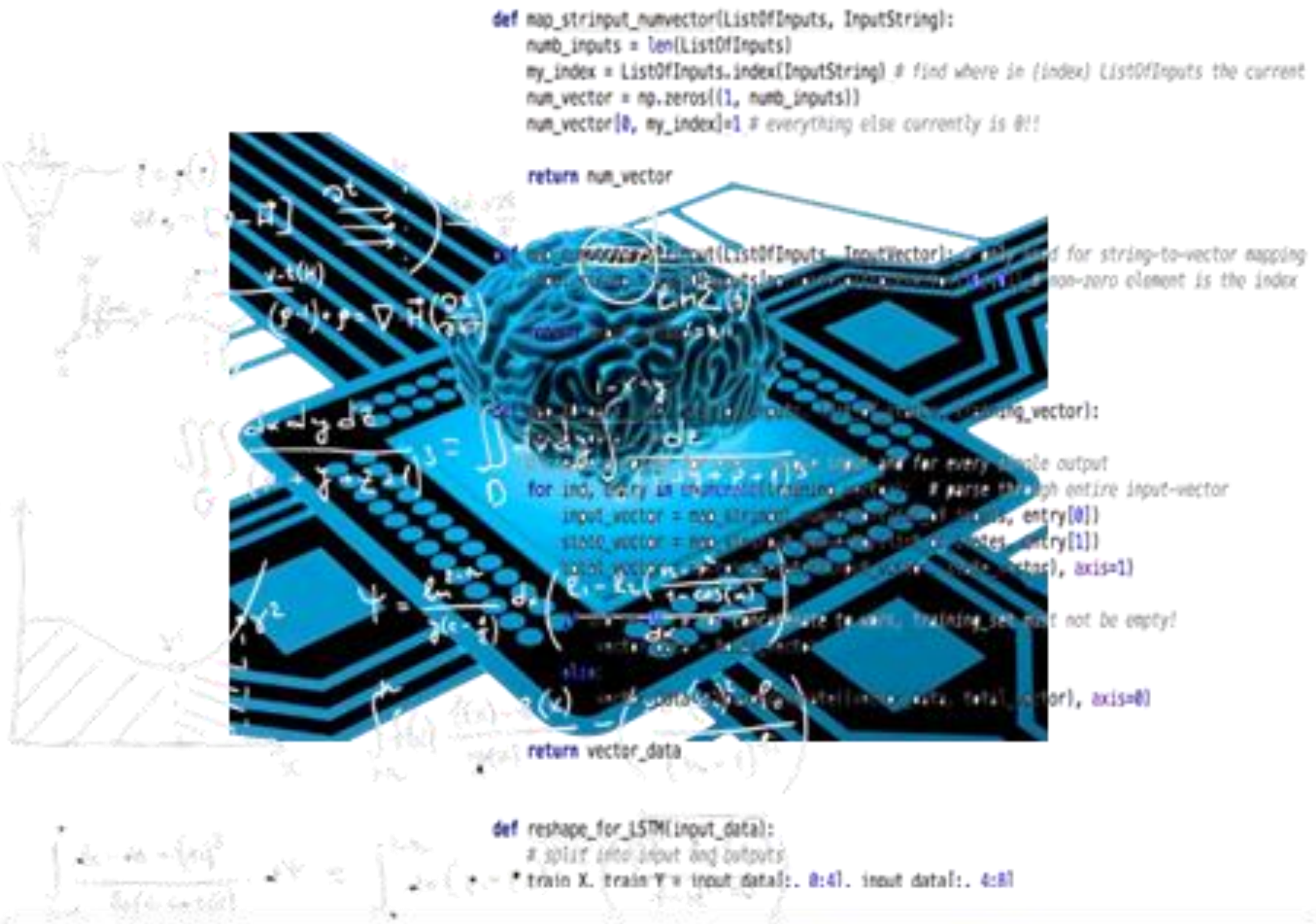




New Materials and Devices



Non *von Neumann* Architecture



```
def map_strinput_numvector(ListOfInputs, InputString):
    numb_inputs = len(ListOfInputs)
    my_index = ListOfInputs.index(InputString) # find where in (index) ListOfInputs the current
    num_vector = np.zeros((1, numb_inputs))
    num_vector[0, my_index]=1 # everything else currently is 0!!

    return num_vector

[illegible](ListOfInputs, InputVector): [illegible] for string-to-vector mapping
    [illegible] non-zero element is the index

[illegible]ing_vector):
    [illegible] for every [illegible]le output
    for ind, [illegible] # parse th[illegible]gh entire input-vector
        input_vector = [illegible], entry[0])
        [illegible] entry[1])
        [illegible]), axis=1)
    [illegible] not be empty!
    [illegible]
    else:
    [illegible]or), axis=0)

    return vector_data

def reshape_for_LSTM(input_data):
    # split into input and outputs
    train X, train Y = input data[:, 0:4], input data[:, 4:8]
```

Hardware – Algorithm Interplay

→ Extension from two-factor to three-factor rules possible O(1)!

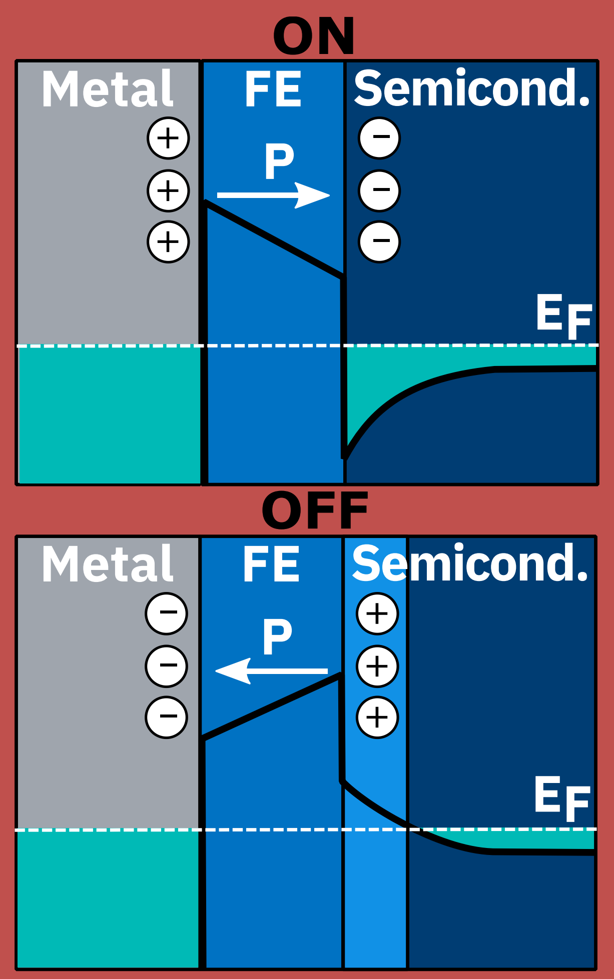
→ Extension to (approximative) Backprop possible O(N)!

Literature of ferroelectrics in AI hardware

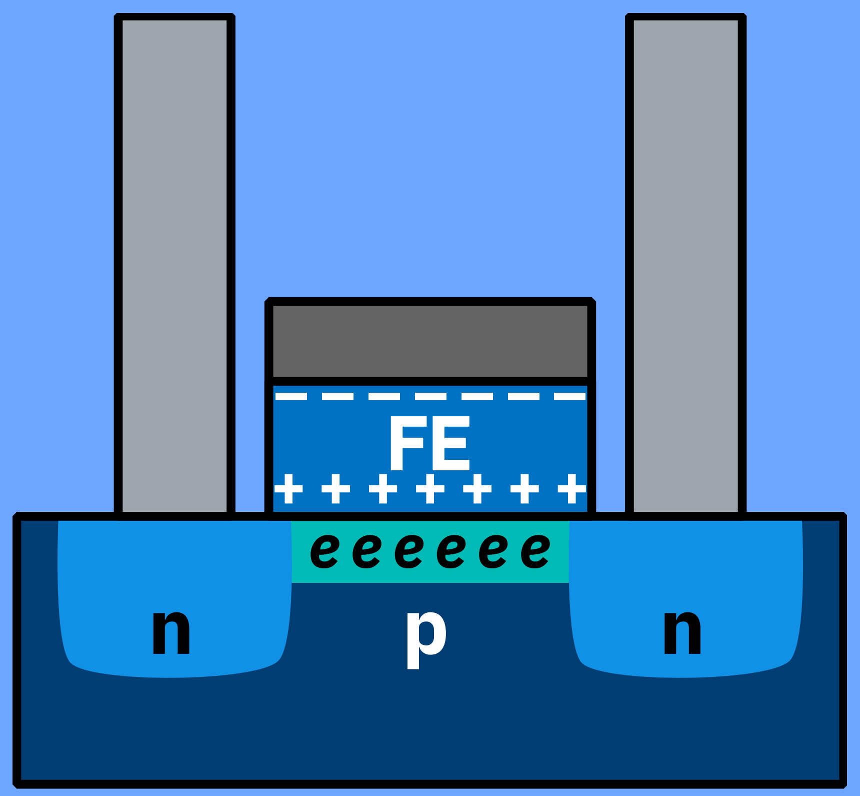
- 2011: discovery of a ferroelectric phase in HfO_2.
- 2017: FeFET integrated in a 28nm HKMG technology (Mulaosmanovic *et al.*, VLSI 2017)
- 2018: IBM: crystallization of $HfZrO_4$ in the FE phase **below 400°C** (O’Connor *et* a*l*., APL Mater. 6, 121103 (2018))
- 2020: this work: first demonstration of a BEOL, CMOS FeFET

Ferroelectric

- FTJ



- FeFET



Summary

- Silicon technology remains the basis for computing devices
  - Leverage existing processes, infrastructure and know-how
  - Continuous extending of materials and function

- New computing paradigms – Neuromorphic computing - provides a path to handle unstructured data
  - Analog signal processing in crossbar arrays
  - Parallel processing of key algorithms in neural networks
  - Electrical and optical implementations

→ Extension to three-factor rule possible!

Previous slide:

Learning in Neural Networks :

from brain-style computing to neuromorphic computing

Wulfram Gerstner

EPFL, Lausanne, Switzerland

1. Review: brain-style computing
2. **Spikey Chip** (Univ. Heidelberg)
3. Loihi Chip (INTEL)]
4. Memristor chips (IBM)
5. Even more **non-standard computing** (EPFL)

   → use physics for rapid and cheap nonlinear 'computing'

A. Momemi, B. Rahmani, M. Mallejac, Philipp del Hougne and Romain Fleury, Backpropagation-free training of deep physical neural networks, *Science* (2025) 383:1297–1303

https://www.science.org/doi/10.1126/science.adi8474

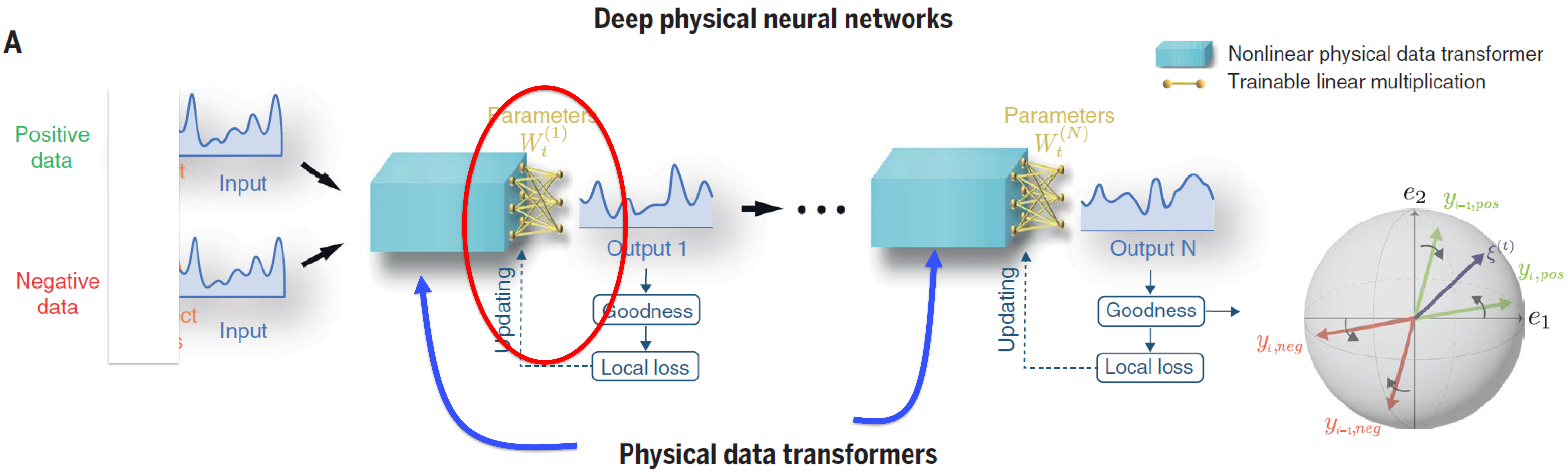
Previous slide:

Romain Fleury is Professor in STI, Laboratory of Wave Engineering

A. Momemi, B. Rahmani, M. Mallejac, Philipp del Hougne and Romain Fleury, Backpropagation-free training of deep physical neural networks, *Science* (2025) 383:1297–1303

https://www.science.org/doi/10.1126/science.adi8474

Local learning rules for physical hardware



Like in CLAPP: -contrastive learning (positive and neg. samples)

-layer-wise loss function

-linear weight matrix is optimized by learning

Hardware: **nonlinearities implemented by physics**

Previous slide and next slide:

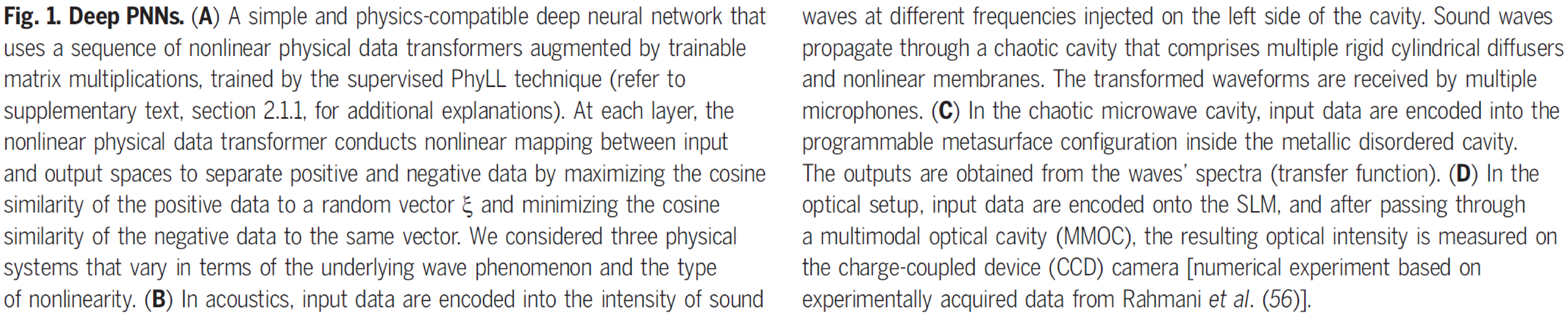


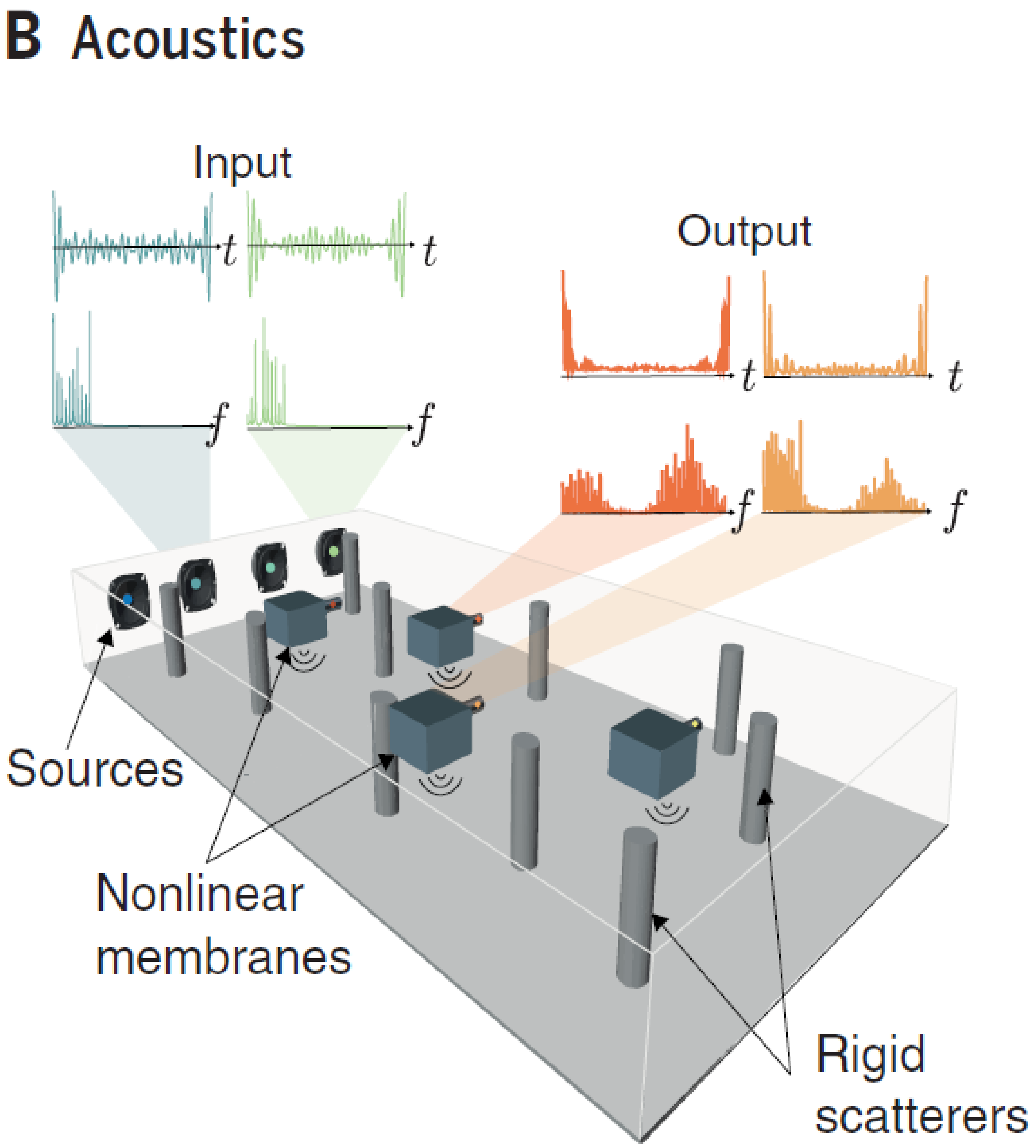
Fig. 1. Deep PNNs. (**A**) A simple and physics-compatible deep neural network that uses a sequence of nonlinear physical data transformers augmented by trainable matrix multiplications, trained by the supervised PhyLL technique (refer to supplementary text, section 2.1.1, for additional explanations). At each layer, the nonlinear physical data transformer conducts nonlinear mapping between input and output spaces to separate positive and negative data by maximizing the cosine similarity of the positive data to a random vector ξ and minimizing the cosine similarity of the negative data to the same vector. We considered three physical systems that vary in terms of the underlying wave phenomenon and the type of nonlinearity. (**B**) In acoustics, input data are encoded into the intensity of sound waves at different frequencies injected on the left side of the cavity. Sound waves propagate through a chaotic cavity that comprises multiple rigid cylindrical diffusers and nonlinear membranes. The transformed waveforms are received by multiple microphones. (**C**) In the chaotic microwave cavity, input data are encoded into the programmable metasurface configuration inside the metallic disordered cavity. The outputs are obtained from the waves' spectra (transfer function). (**D**) In the optical setup, input data are encoded onto the SLM, and after passing through a multimodal optical cavity (MMOC), the resulting optical intensity is measured on the charge-coupled device (CCD) camera [numerical experiment based on experimentally acquired data from Rahmani *et al*. (*56*)].

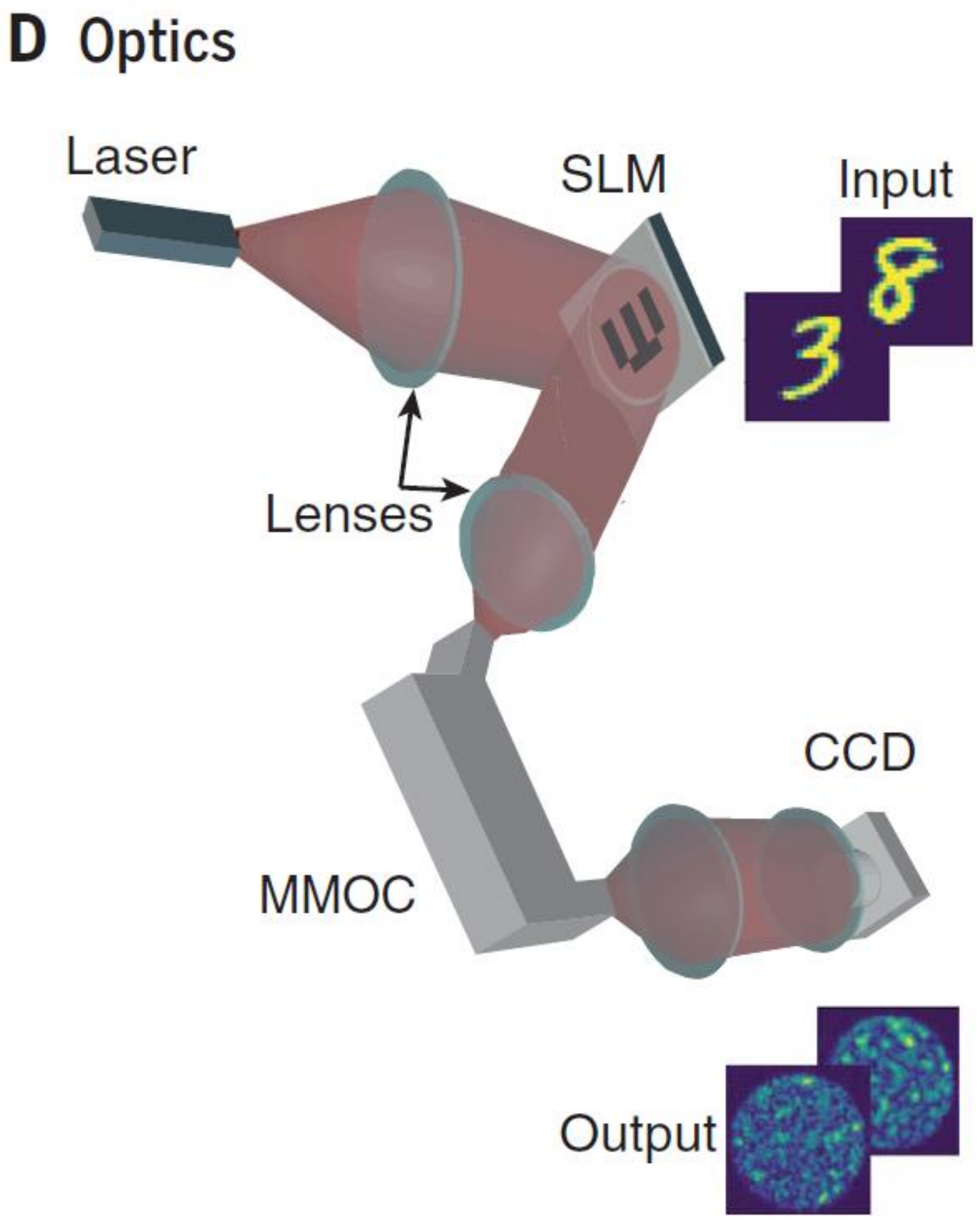
- Input is high-dimensional, but shown here plotted here organized in 1d.
  → discretize the horizontal axis, and then you can read-off the input values
- The nonlinear transform is fixed (physical transformation), and only the weights are learned

A. Momemi, B. Rahmani, M. Mallejac, Philipp del Hougne and Romain Fleury, Backpropagation-free training of deep physical neural networks, *Science* (2025) 383:1297–1303

https://www.science.org/doi/10.1126/science.adi8474

Physical hardware for nonlinear data transforms





MMOC:
Multi-modal optical cavity

CCD-camera
Charge-Coupled Device

SLM:
Spatial light modulator

A.Momemi, B. Rahmani, M. Mallejac, Ph del Hougne and Romain Fleury, Backpropagation-free training of deep physical neural networks, *Science* (2025) 383:1297–1303

https://www.science.org/doi/10.1126/science.adi8474

Previous slide:

Wikipedia: A **spatial light modulator** (**SLM**) is a device that can control the intensity, phase, or polarization of light in a spatially varying manner. A simple example is an overhead projector transparency. Usually when the term SLM is used, it means that the transparency can be controlled by a computer.
https://en.wikipedia.org/wiki/Spatial_light_modulator

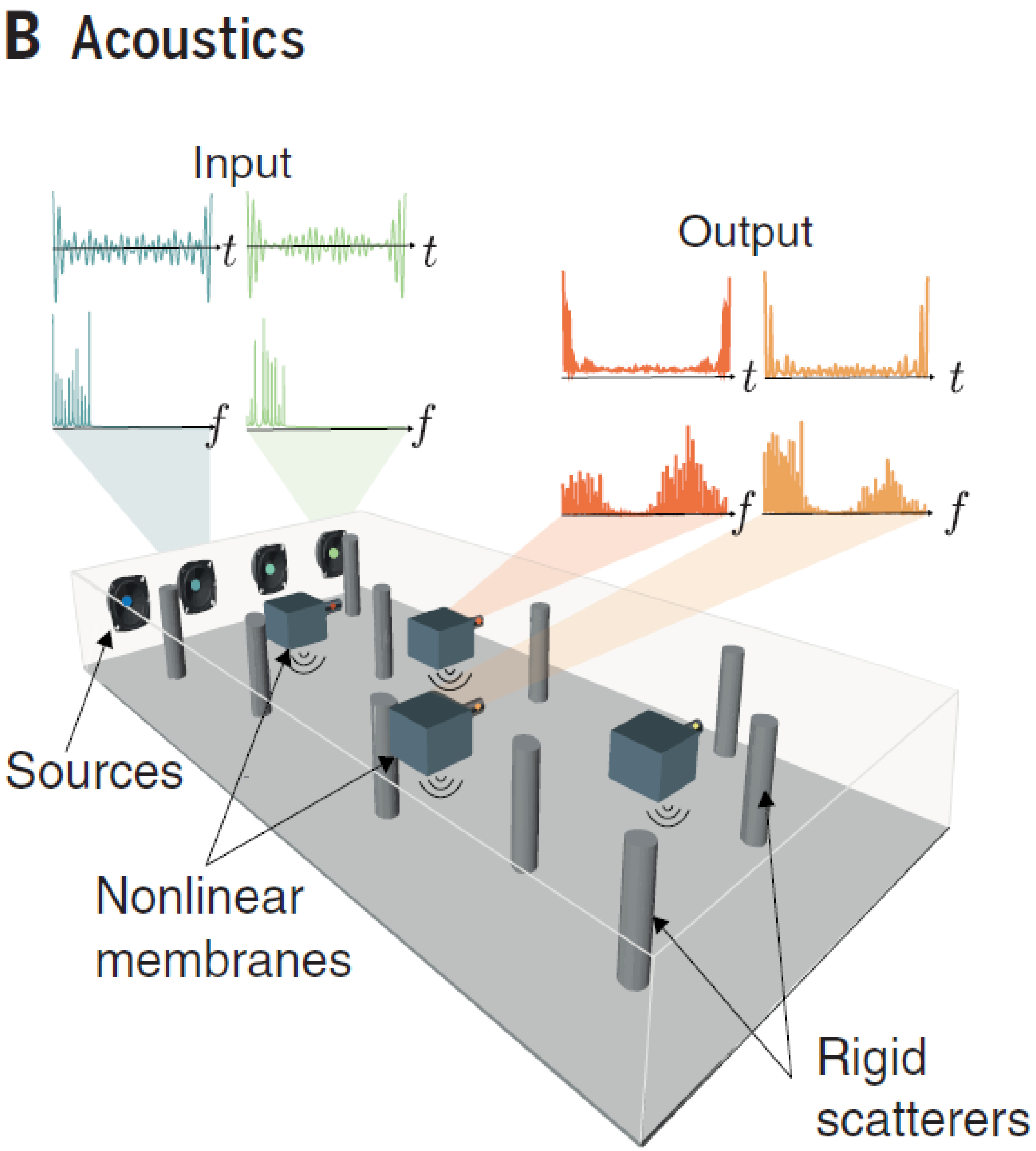
In the paper of Momeni, the SLM encoded information in the **phase** (in contrast to the Wikipedia example of an overhead projector, where only the intensity is modulated).

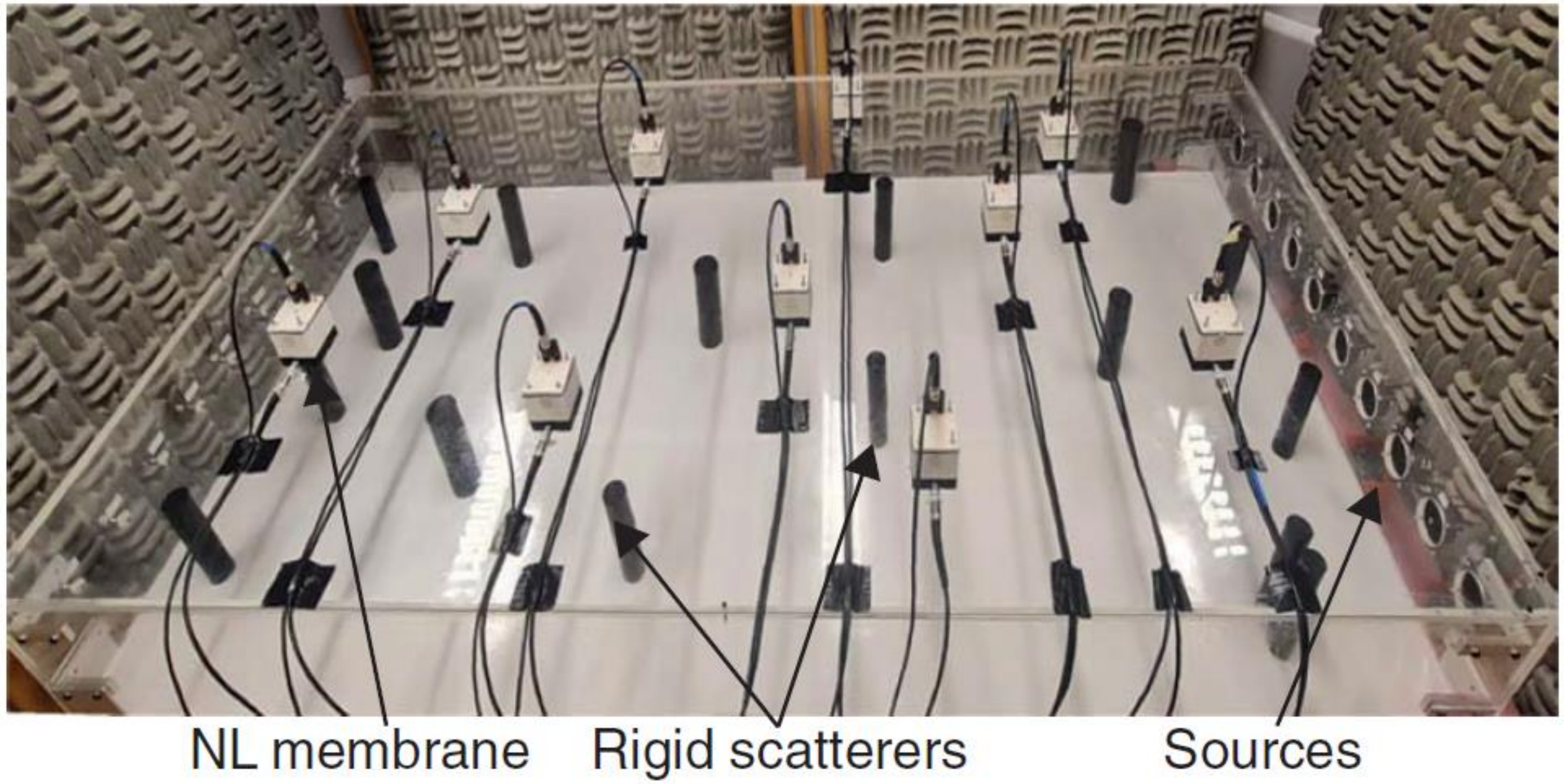
“The physical optical system performed a complex spatial transformation. Although this transformation was linear in the complex domain, the process became nonlinear as a result of the data being encoded onto the phase (SLM) and the subsequent measurement of the intensity squared on the camera.»

A.Momemi, B. Rahmani, M. Mallejac, Ph del Hougne and Romain Fleury, Backpropagation-free training of deep physical neural networks, *Science* (2025) 383:1297–1303

https://www.science.org/doi/10.1126/science.adi8474

Physical hardware



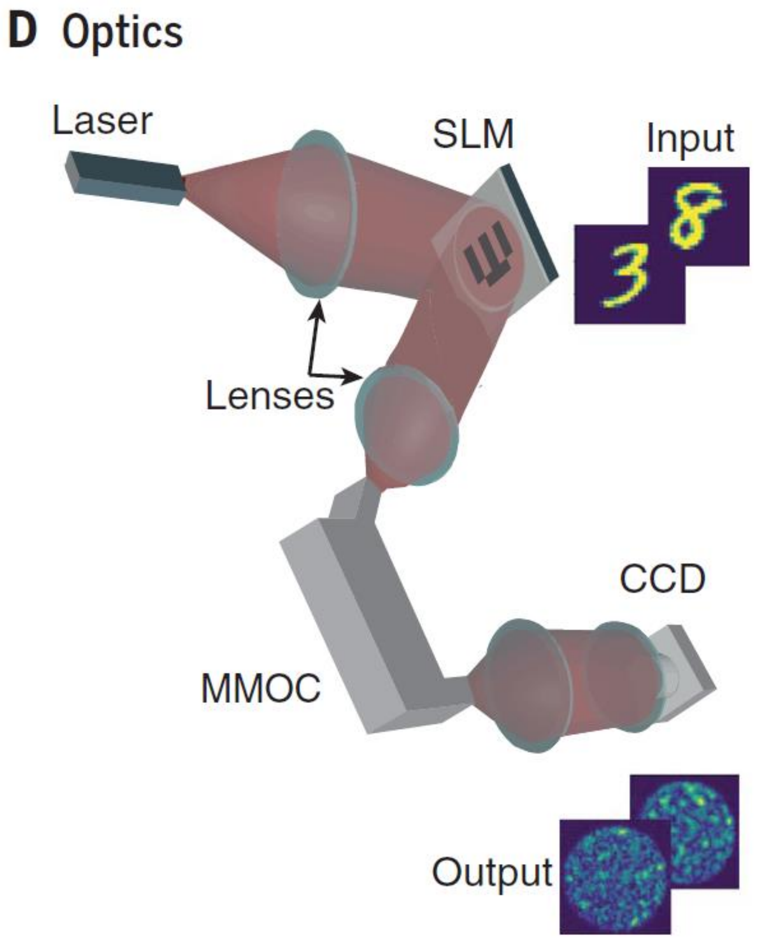


Previous slide:
On the right: actual lab realization of the acoustic nonlinear physical system

A.Momemi, B. Rahmani, M. Mallejac, Ph del Hougne and Romain Fleury, Backpropagation-free training of deep physical neural networks, *Science* (2025) 383:1297–1303

https://www.science.org/doi/10.1126/science.adi8474

Local learning rules for physical hardware



MMOC:

Multi-modal optical cavity

CCD-camera

Charge-Coupled Device

SLM:

Spatial light modulator

Optical system:

«At inference–time (usage of trained network) energy–efficiency advantage of ~8000 × compared with that of digital-electronic processors for large-scale future transformer models»

A.Momemi, B. Rahmani, M. Mallejac, Ph del Hougne and Romain Fleury, Backpropagation-free training of deep physical neural networks, *Science* (2025) 383:1297–1303

https://www.science.org/doi/10.1126/science.adi8474

(previous slide)

Alternatives to standard chips are a hot topic. The reason is again energy consumption.

The 80-percent question again:

[ ] In this hardware part on physical neural nets, at least 60 percent of the material was new to me.

[ ] for this hardware part, up to here, I have the feeling that I
understood at least 80 percent of the material
(at the level at which it was presented)

Previous slide:

Learning in Neural Networks :

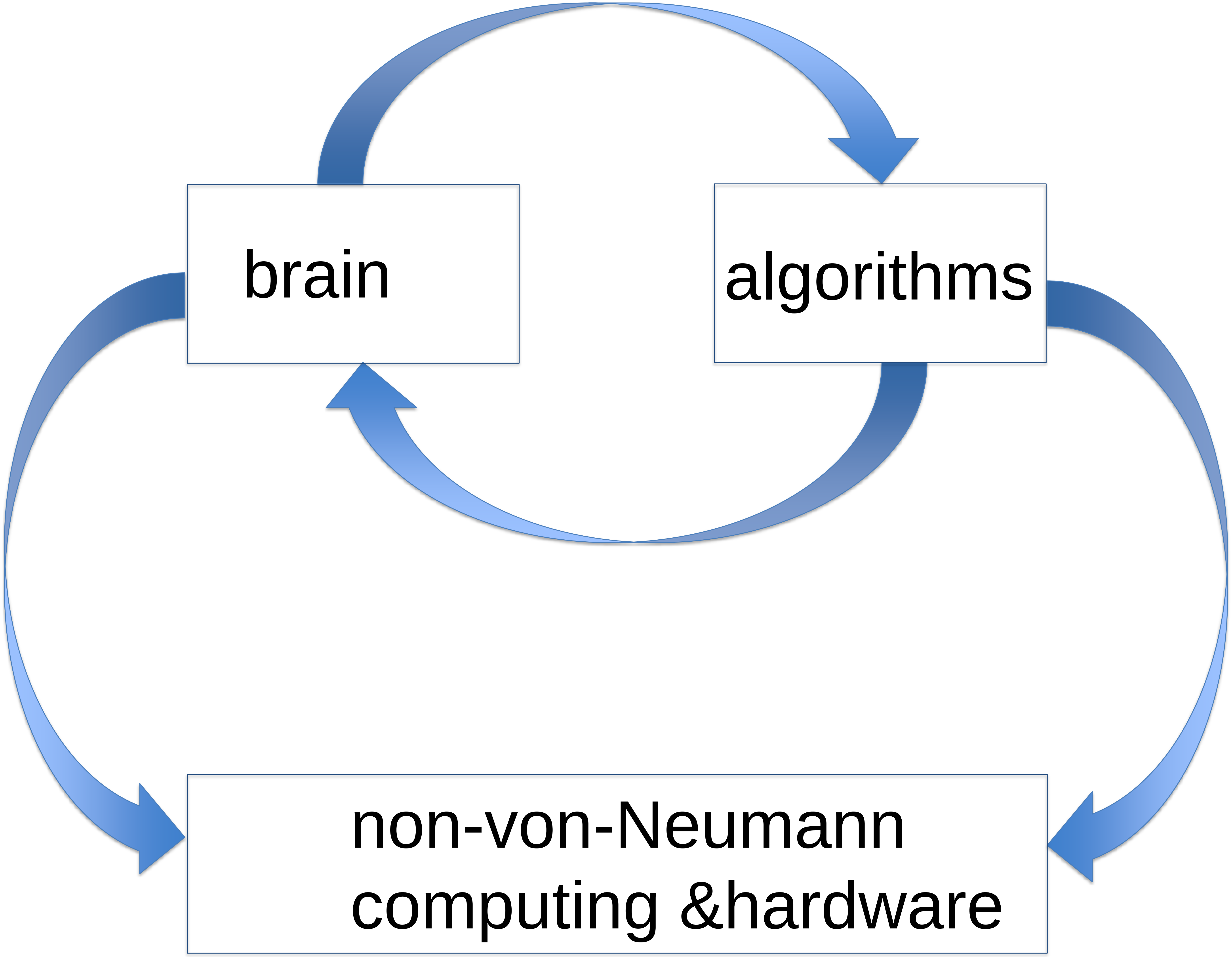
from brain-style computing to neuromorphic computing

Wulfram Gerstner

EPFL, Lausanne, Switzerland

1. Review: brain-style computing
2. Spikey Chip (Univ. Heidelberg)
3. Loihi Chip (INTEL)]
4. Memristor chips (IBM)
5. Even more non-standard computing (EPFL)
6. **The problem of Energy consumption**

Learning Rules



brain

algorithms

non-von-Neumann computing &hardware

Energy consumption of the brain

- Sedentary humans eat and use 2500 kCal per day
- Translate to Joule → 10 000 kJ
- Brain facts: 20 percent of energy consumption of human at rest goes into the brain
- Most of it goes into synaptic signaling (spike transmission)
- Brain uses 24 – 30 Watt (5 modern light bulbs)

The power consumption of the brain is relatively low!

→ **10h of hard thinking = 0.3kWh**

https://www.brainfacts.org/Brain-Anatomy-and-Function/Anatomy/2019/How-Much-Energy-Does-the-Brain-Use-020119
https://biology.stackexchange.com/questions/16316/what-is-the-energy-consumption-of-the-brain

Previous slide.
Claim the power consumption of the brain (30W) is relatively low.

Low compared to what?
- Compare with GPU
= Compare with household power consumption.

Energy consumption of one GPU

- 300 W (RX 6800/6900 XT)
- 350 W (RTX 3080/3090)

→ 10h of training an ANN on 1 GPU = 3.5 kWh

1 day of training an ANN on 1 GPU = 8000Wh = 8 kWh

4 months GPU usage → 1000 kWh

12 months GPU usage → 3000 kWh

https://www.tomshardware.com/features/graphics-card-power-consumption-tested

Previous slide:

A day has 24 hours. So we multiply the power (350W) with the number of hours.

4 months have 120 days. Again a simple multiplication

The question then is: are 3000kWh per year a lot?

We need to compare with ‘normal’ energy consumption.

Electrical household energy consumption

Typical Swiss **electricity use in household** (fridge, TV, light)
→ **about 1000 kWh per year and person.**

- 2 persons sharing apartment = 2200kWh per year
- 4 persons sharing house = 4000kWh per year

Heating/warm water with heat pump

- 4 persons sharing house: 6000kWh per year

→ **1500 kWh per year and person**

https://pubdb.bfe.admin.ch/de/publication/download/10559

Previous slide:

Comparison

- Brain 30W → 260 kWh per year and person
- Living in Switzerland → 2500 kWh per year and person
- GPU → 3000 kWh per year and GPU

Problem!!!!

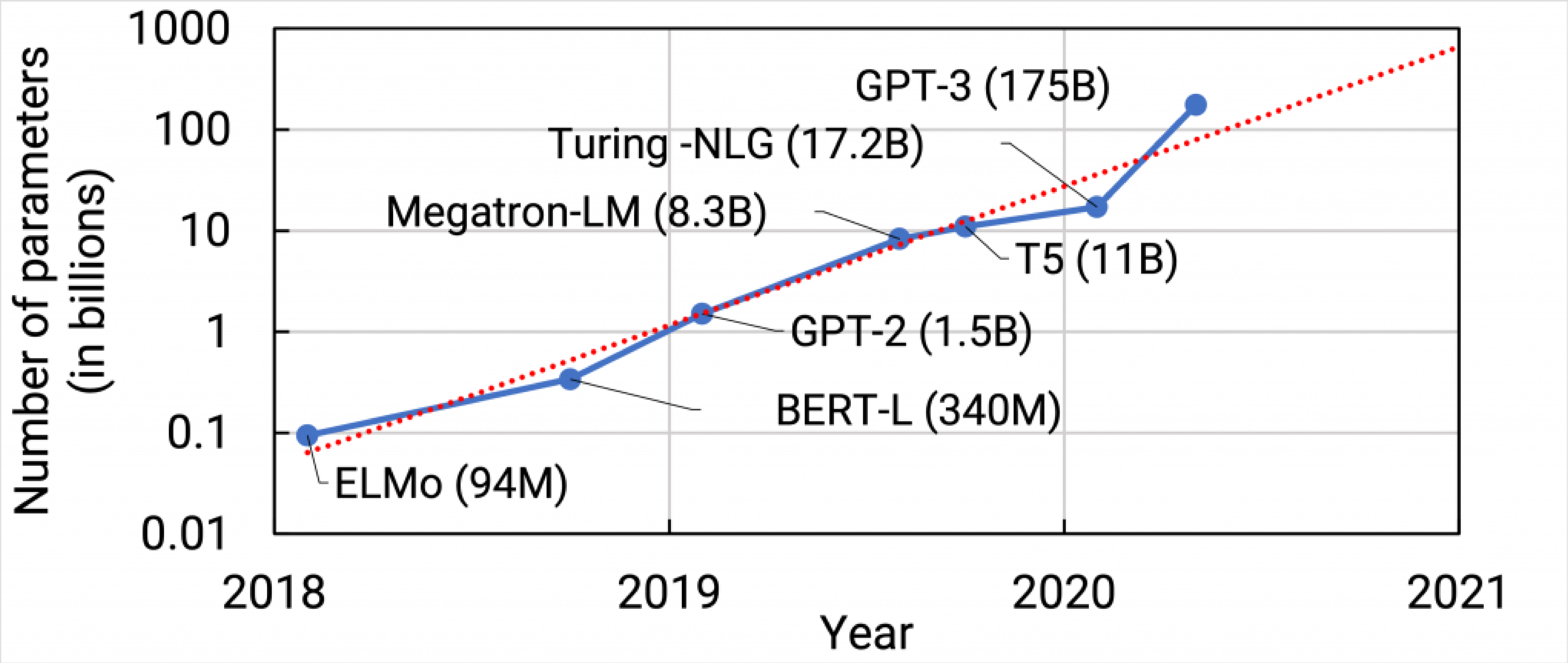
Solution? – use your machine carefully!

- think about better computer architecture!

Previous slide:

The problem is that if you your model optimization uses on average a single GPU over the year, you use more energy on your GPU than you use by living in a normal rental apartment.

The neural network *size* explosion

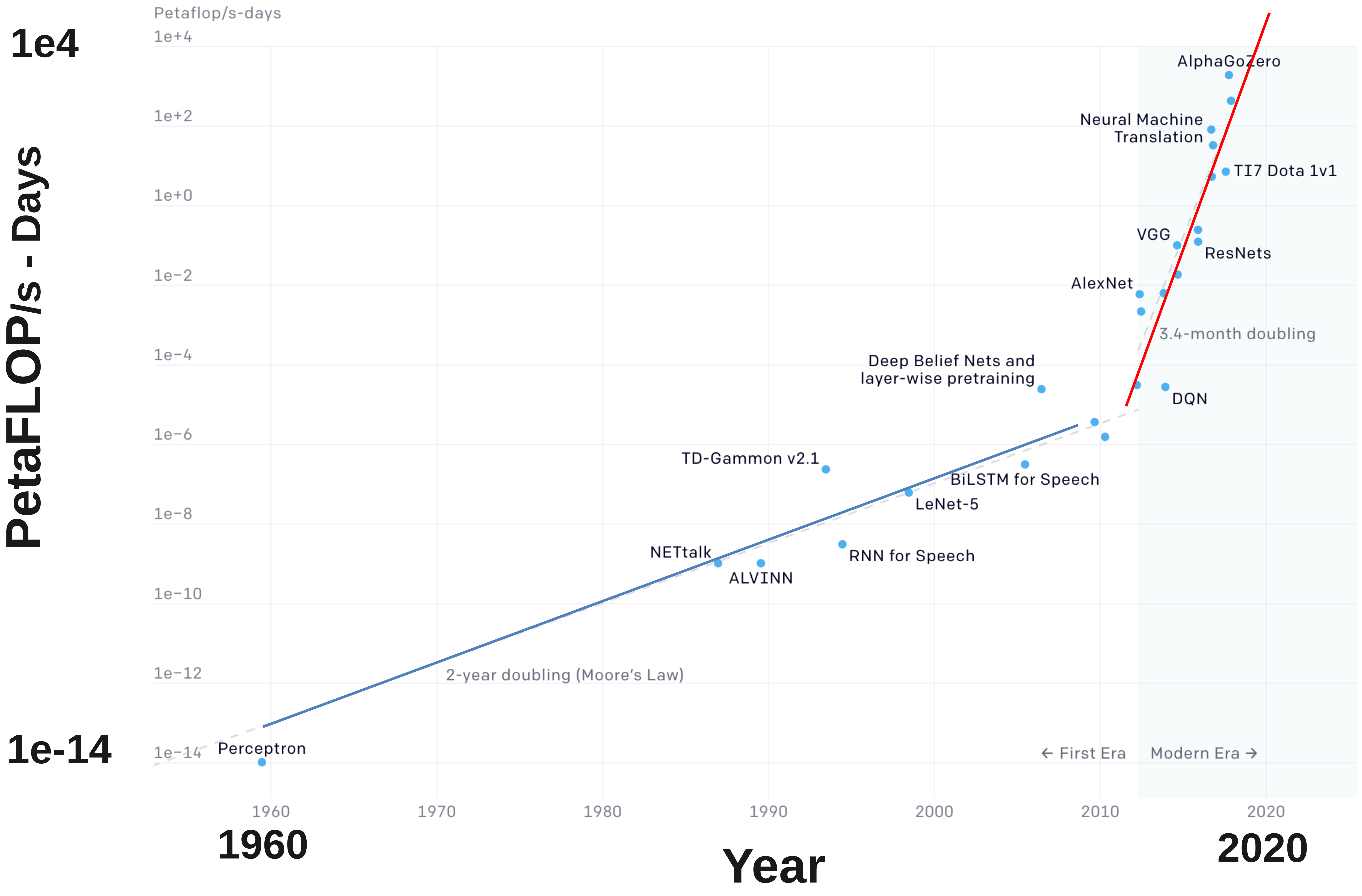


- Source: NVIDIA

Previous slide:

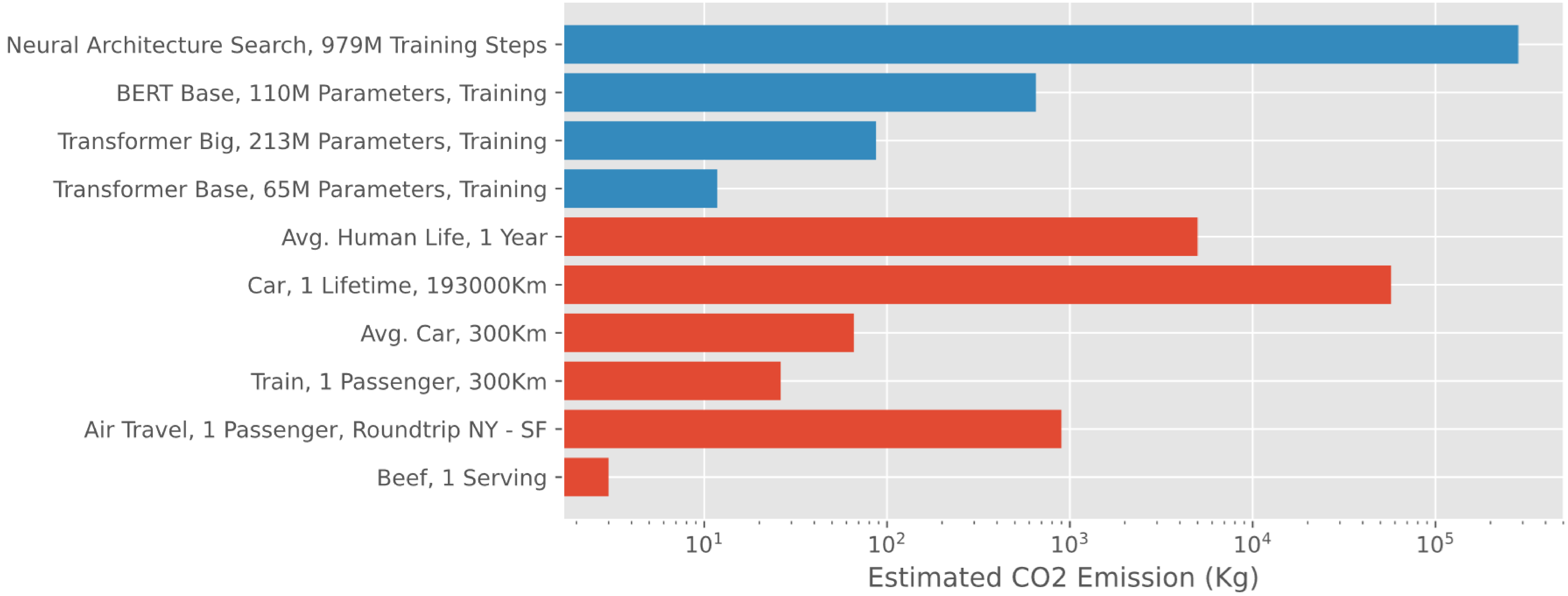
The *compute* explosion

Source:
Openai.com



Previous slide:

The *power and carbon emission* explosion



- Source: E. Strubell et al., arXiv:1906.02243

oil/gasoline: 1liter = 10kWh = 2.5 kg CO_2

1000 km by plane emits as much CO_2 as 1000 km by car.

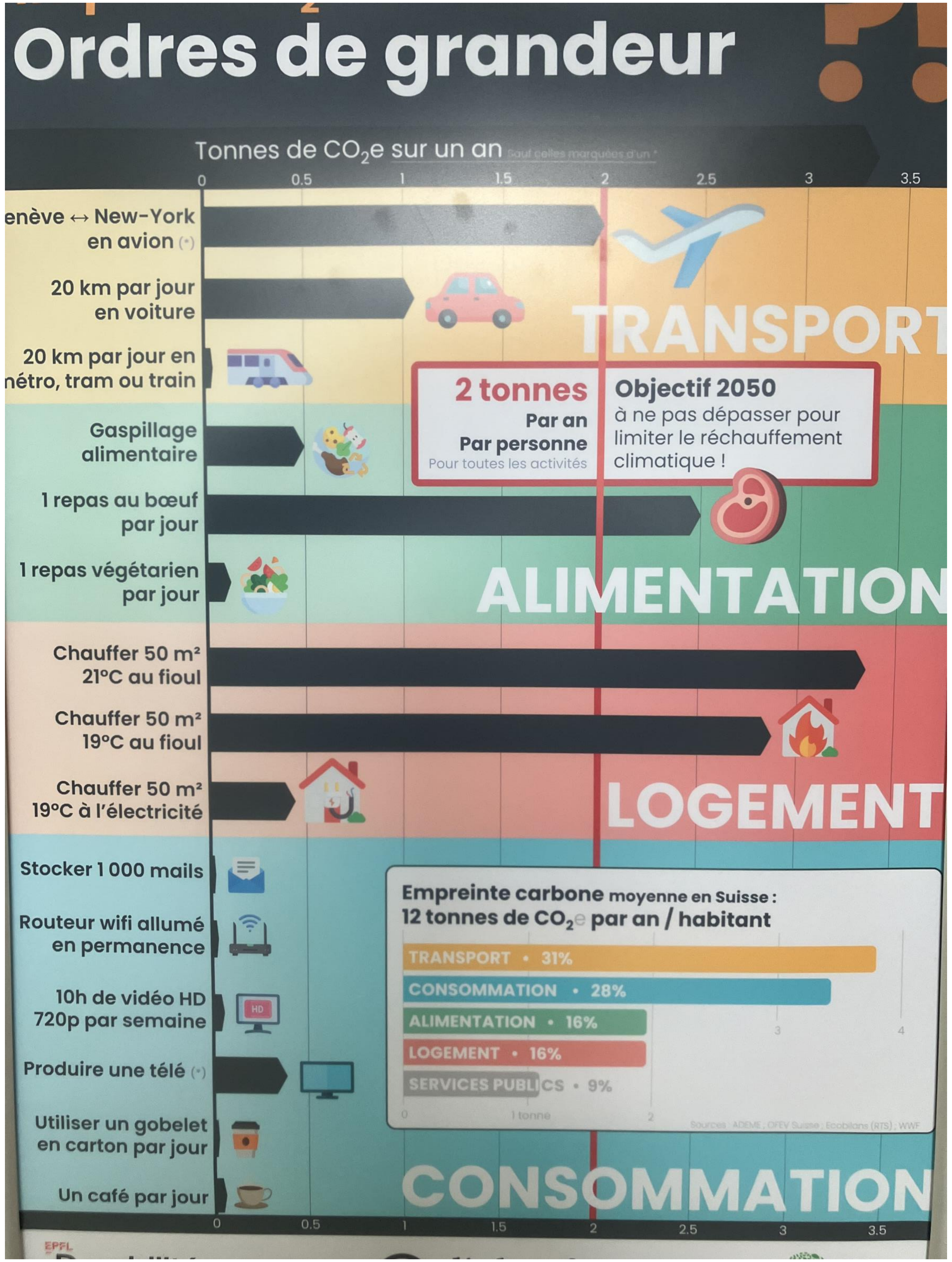
New York → Geneva = 6200km

Flight Geneva ←→New York

1 beef meal per day

Heating 50m2: 21°vs
(fossil fuels) 19°
heat pump

Fabrication TV screen
1 paper cup per day



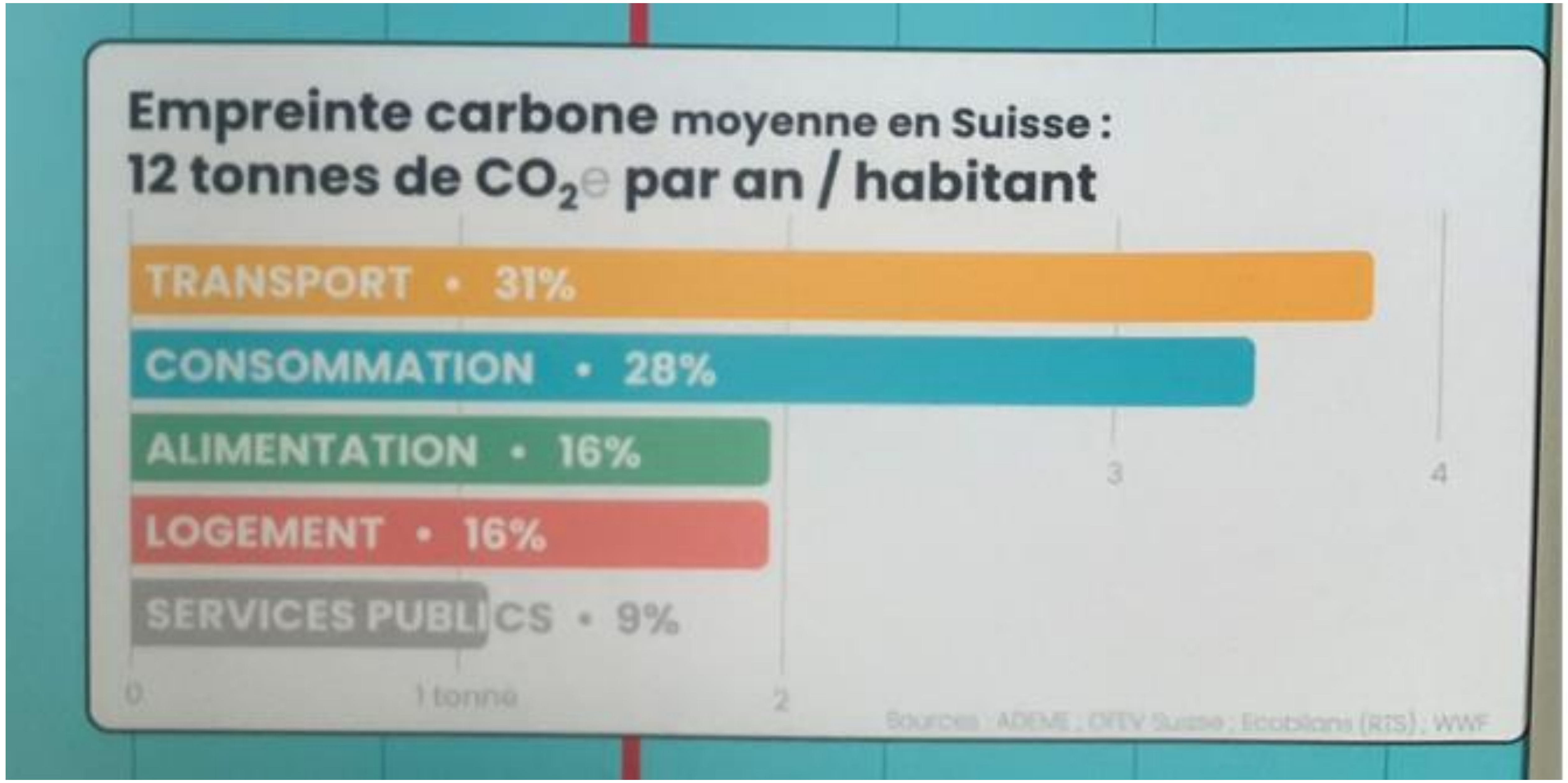
2 tonnes Par an Par personne Pour toutes les activités

Objectif 2050 à ne pas dépasser pour limiter le réchauffement climatique !

Empreinte carbone moyenne en Suisse :
12 tonnes de CO_2e par an / habitant

Sources : ADEME ; OFEV Suisse ; Ecobilans (RTS) ; WWF

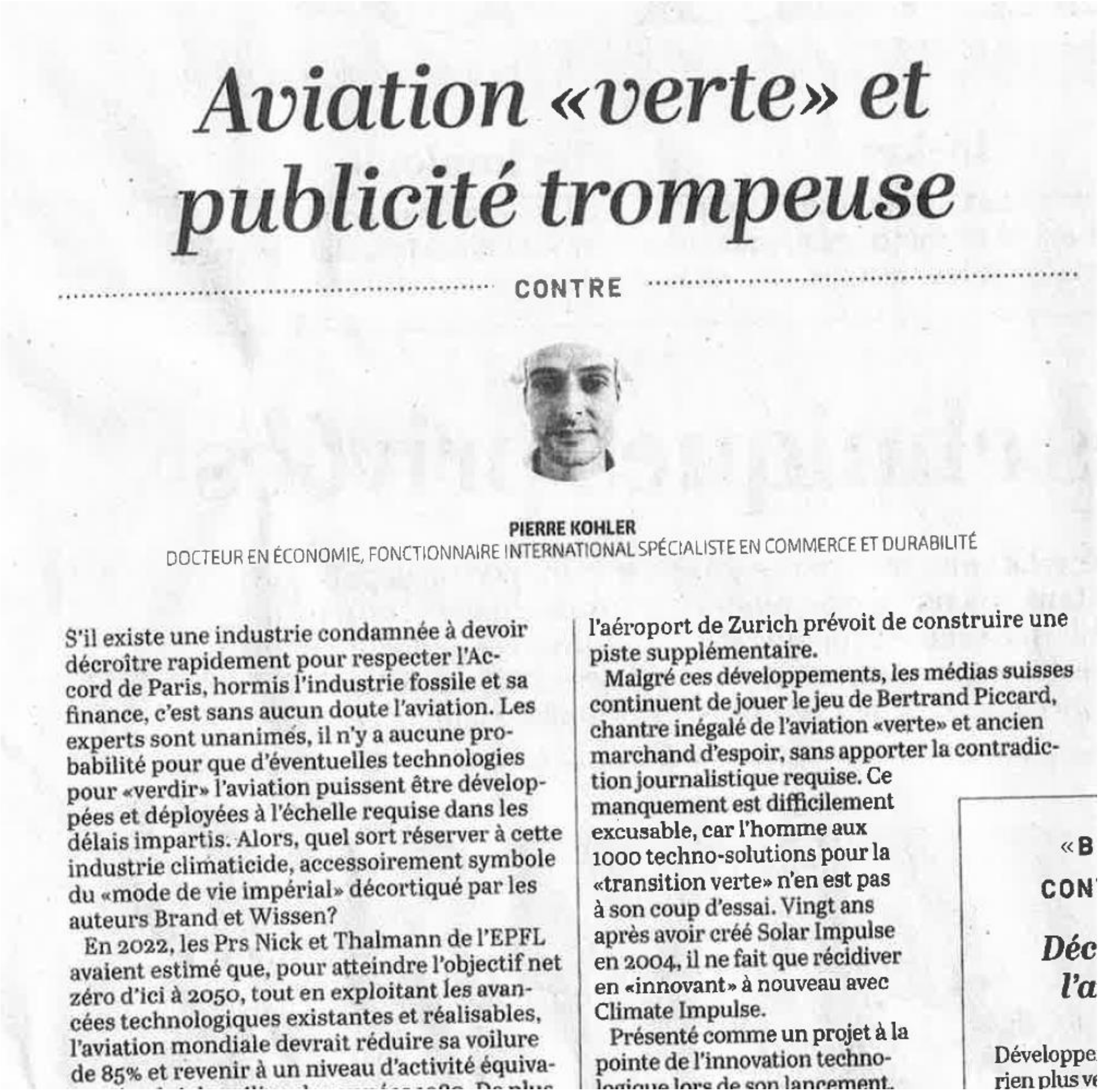
Where can YOU contribute?



Empreinte carbone moyenne en Suisse :
12 tonnes de CO_2e par an / habitant

Sources : ADEME ; OFEV Suisse ; Ecobilans (RTS) ; WWF

Le Temps, April 30, 2024, Bien au Contraire: Decarboniser l’aviation



Aviation «verte» et publicité trompeuse

CONTRE

PIERRE KOHLER
DOCTEUR EN ÉCONOMIE, FONCTIONNAIRE INTERNATIONAL SPÉCIALISTE EN COMMERCE ET DURABILITÉ

S’il existe une industrie condamnée à devoir décroître rapidement pour respecter l’Accord de Paris, hormis l’industrie fossile et sa finance, c’est sans aucun doute l’aviation. Les experts sont unanimes, il n’y a aucune probabilité pour que d’éventuelles technologies pour «verdir» l’aviation puissent être développées et déployées à l’échelle requise dans les délais impartis. Alors, quel sort réserver à cette industrie climaticide, accessoirement symbole du «mode de vie impérial» décortiqué par les auteurs Brand et Wissen?

En 2022, les Prs Nick et Thalmann de l’EPFL avaient estimé que, pour atteindre l’objectif net zéro d’ici à 2050, tout en exploitant les avancées technologiques existantes et réalisables, l’aviation mondiale devrait réduire sa voilure de 85% et revenir à un niveau d’activité équiva-

l’aéroport de Zurich prévoit de construire une piste supplémentaire.

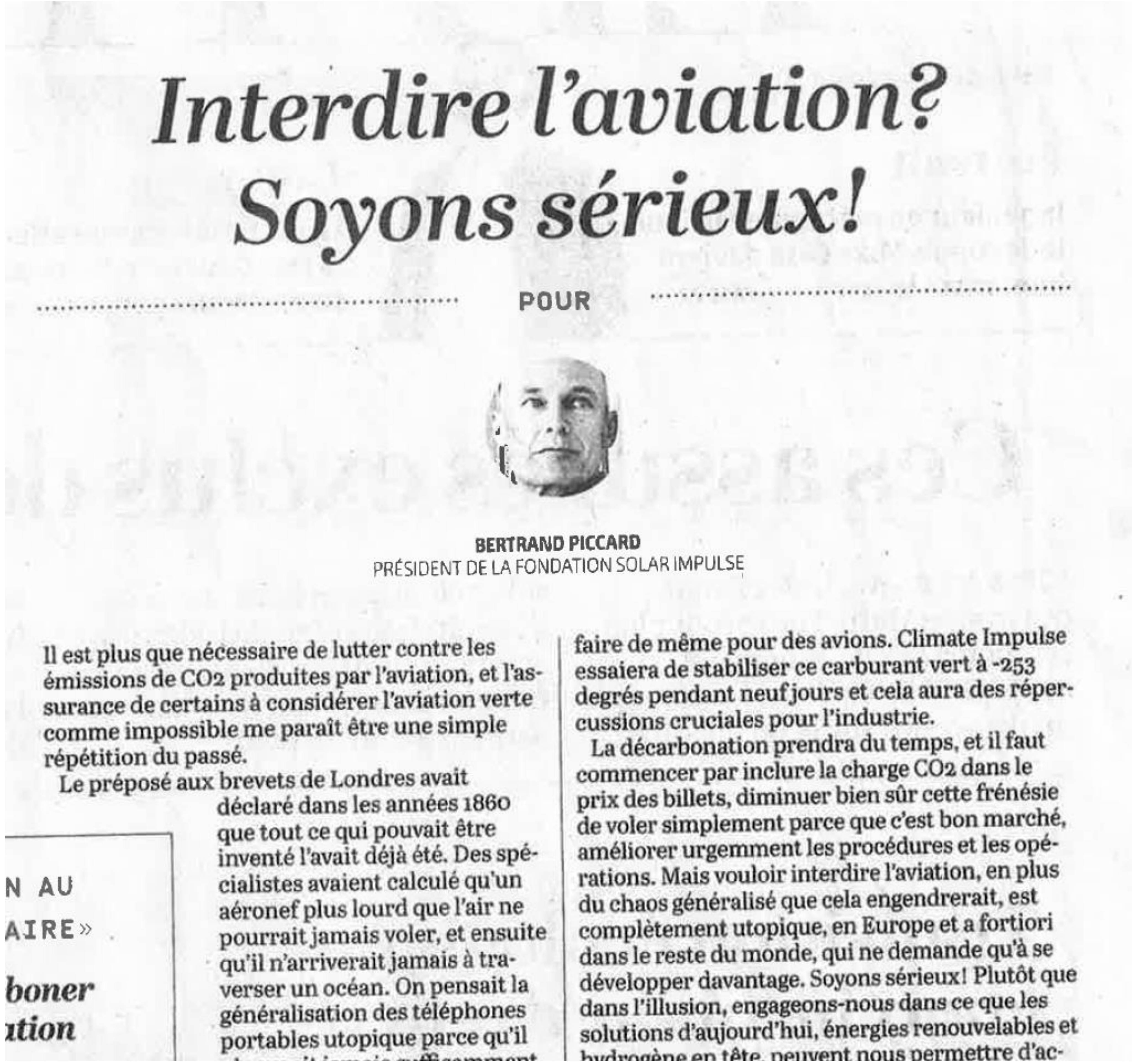
Malgré ces développements, les médias suisses continuent de jouer le jeu de Bertrand Piccard, chantre inégalé de l’aviation «verte» et ancien marchand d’espoir, sans apporter la contradiction journalistique requise. Ce manquement est difficilement excusable, car l’homme aux 1000 techno-solutions pour la «transition verte» n’en est pas à son coup d’essai. Vingt ans après avoir créé Solar Impulse en 2004, il ne fait que récidiver en «innovant» à nouveau avec Climate Impulse.

Présenté comme un projet à la pointe de l’innovation technologique lors de son lancement,

«B
CONT

Déc
l’a

Développe
rien plus ve



Interdire l’aviation? Soyons sérieux!

POUR

BERTRAND PICCARD
PRÉSIDENT DE LA FONDATION SOLAR IMPULSE

N AU
AIRE»

boner
ation

Il est plus que nécessaire de lutter contre les émissions de CO2 produites par l’aviation, et l’assurance de certains à considérer l’aviation verte comme impossible me paraît être une simple répétition du passé.

Le préposé aux brevets de Londres avait déclaré dans les années 1860 que tout ce qui pouvait être inventé l’avait déjà été. Des spécialistes avaient calculé qu’un aéronef plus lourd que l’air ne pourrait jamais voler, et ensuite qu’il n’arriverait jamais à traverser un océan. On pensait la généralisation des téléphones portables utopique parce qu’il

faire de même pour des avions. Climate Impulse essaiera de stabiliser ce carburant vert à -253 degrés pendant neuf jours et cela aura des répercussions cruciales pour l’industrie.

La décarbonation prendra du temps, et il faut commencer par inclure la charge CO2 dans le prix des billets, diminuer bien sûr cette frénésie de voler simplement parce que c’est bon marché, améliorer urgemment les procédures et les opérations. Mais vouloir interdire l’aviation, en plus du chaos généralisé que cela engendrerait, est complètement utopique, en Europe et a fortiori dans le reste du monde, qui ne demande qu’à se développer davantage. Soyons sérieux! Plutôt que dans l’illusion, engageons-nous dans ce que les solutions d’aujourd’hui, énergies renouvelables et hydrogène en tête, peuvent nous permettre d’ac-

Energy consumption problem (for computing) will further increase over time!

Solution? – use your machine carefully!
- think more, simulate less!
- invent better computer architecture!

Global warming is a reality!

→ Some regions no longer inhabitable/agriculture
→ Big migration waves/relocation

Solution? – tax on CO2
- reliable and predictable increase from 10cent to 100 dollars over 25 years.
- few countries start, others will follow

Thanks!

The END

… for today. There will be 1 more session.

Paper Presentation During the Exam (12 min, bring your computer).

Selection of papers:

Rules: Each student has to present a **different** paper during the exam. Here is the algo in order to achieve a fair and transparent assignment of papers to students:

Step 1: From the pool of papers, you freely choose and reserve 2 available papers for you for the duration of 5 days. While you have reserved the papers, no other student can reserve the same paper. Paper status marked by tag: ‘reserved by NAME on DATE’

Step 2: After at most 5 days the paper status must change by picking one of the two XOR options:

(i) EITHER You give back both papers to the pool and GO TO Step 1.

(ii) OR you decide to present one of your two reserved papers in the exam. This paper changes status from ‘reserved’ to ‘chosen for presentation by NAME’. It is not accessible to other students. At the same time you give back the other paper to the pool of available papers (i.e., you remove the tag ‘reserved by’) so that other students can choose it and you GO TO Step 3.

Step 3. Prepare for the exam and END.

Papers fall into three different categories (Experimental,Theory,Hardware), and you can pick whatever you prefer. The categories are only meant as a first guide for your selection.

your EPFL e-mail will soon be migrated to Exchange Online on 20.05.2025.

This migration requires to modernize our e-mail clients

What to do before the migration

- Make sure you are using a compatible email client:
  - Microsoft Outlook
  - Apple Mail
  - Thunderbird
  - Gmail
- Check that your email client is correctly configured by following the documentation: EPFL Messagerie and Listes

What to do on the day of migration and after

- Restart or reconfigure your email client if necessary (EPFL Messagerie and Listes, How To videos)

In the event of a problem after reconfiguring your e-mail client

- Consult the Knowledge Base
- Contact Service Desk IT Support
- Temporarily use https://outlook.office.com
- Please check the ‘Junk mail’ folder for important messages