Towards deep neural networks with spintronic nano-oscillators as neurons

Julie Grollier[1]

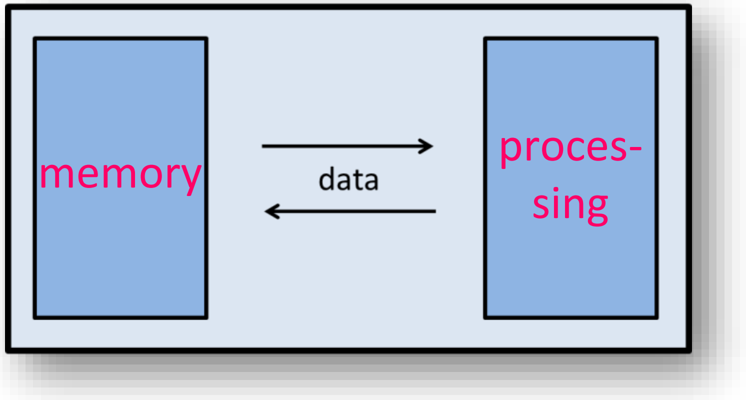
Nathan Leroux[1], Danijela Marković[1], Jérémie Laydevant[1], Dedalo Sanz Hernandez[1], Philippe Talatchian[1], Miguel Romera[1], Mathieu Riou[1], Jacob Torrejon[1], Flavio Abreu Araujo[1], Paolo Bortolotti[1], Juan Trastoy[1], Teodora Petrisor [1], Vincent Cros[1], Guru Khalsa[2], Mark Stiles[2], Sumito Tsunegi[3], Kay Yakushiji[3], Akio Fukushima[3], Hitoshi Kubota[3], Shinji Yuasa[3], Ricardo Ferreira[4], Alex Jenkins[4], Leandro Martins[4], Damir Vodenicarevic[5], Tifenn Hirtzlin[5], Maxence Ernoult[5], Nicolas Locatelli[5], Alice Mizrahi[1], Damien Querlioz[5]

[1]CNRS/Thales, France [2]NIST, USA [3]AIST, Japan [4]INL,Portugal [5]C2N, France

Training neural networks on computers is extremely power inefficient

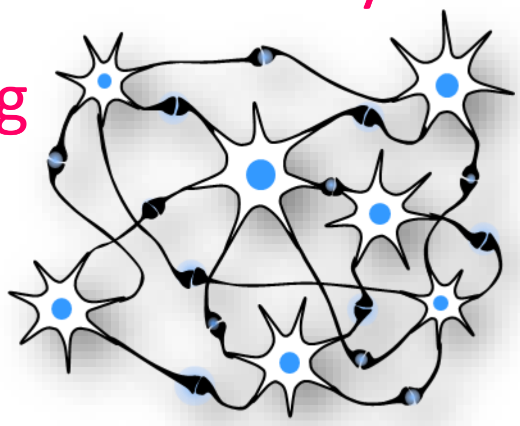
Digital computer:

CPUs, GPUs, TPUs, FPGAs



1000 kW.h to train a
Natural Language Processor

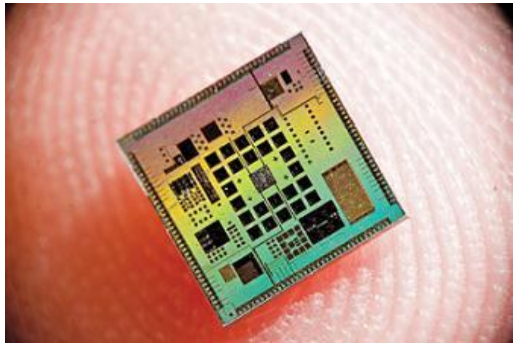
Brain : 20 W

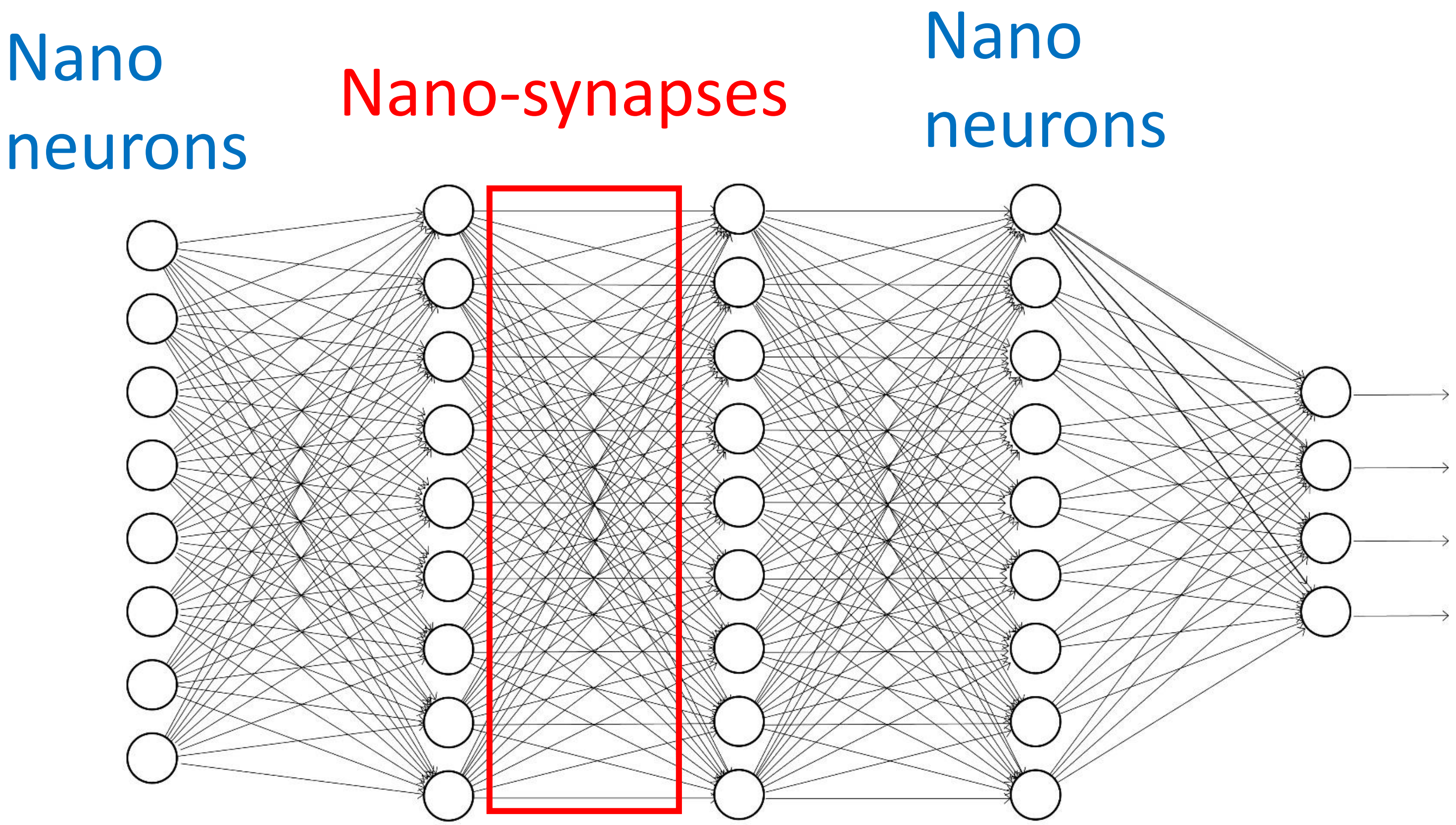


6 years of brain operation

D. Marković et al, "Physics for neuromorphic computing", arXiv:2003.04711

Orders of magnitude in energy can be saved by assembling physical synapses and neurons in neuromorphic chips



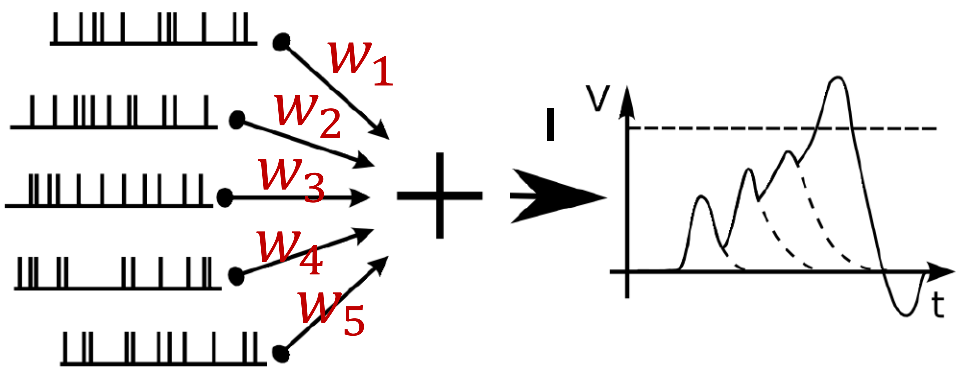


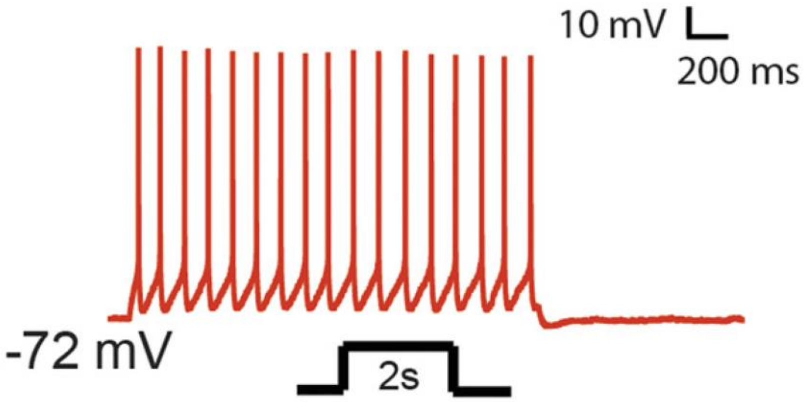
Hundred millions of neurons and synapses in a 1 cm^2 chip

→ Each device smaller than 1 μm^2

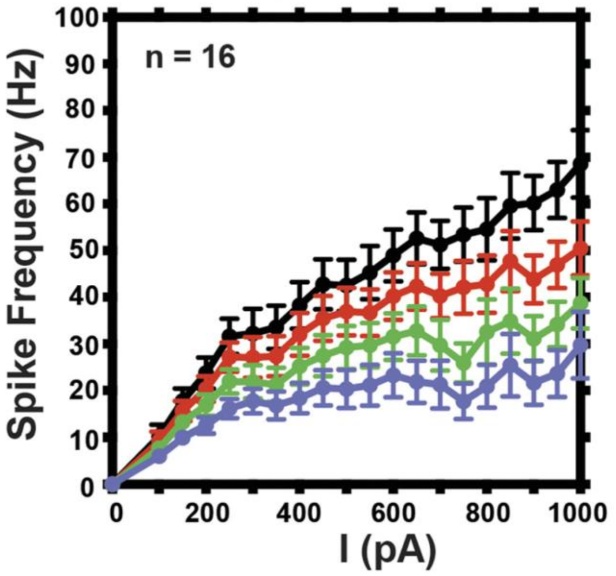
Neurons are non-linear and synapses are valves with memory

- **Brain**



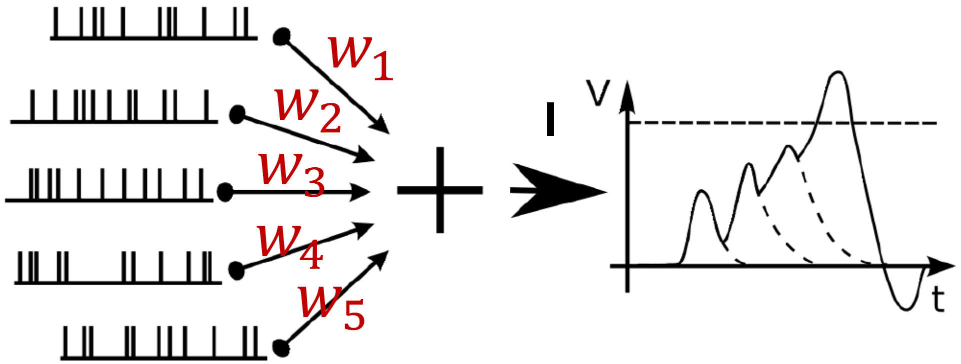


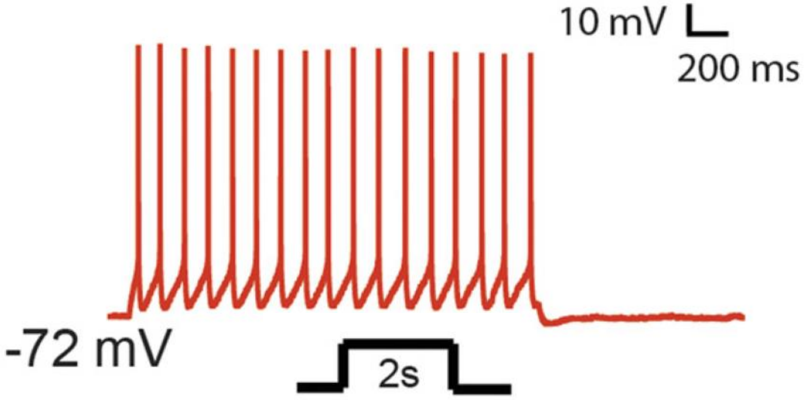
D. Guan et al, J Neurophysiol. 113, 2014 (2015)

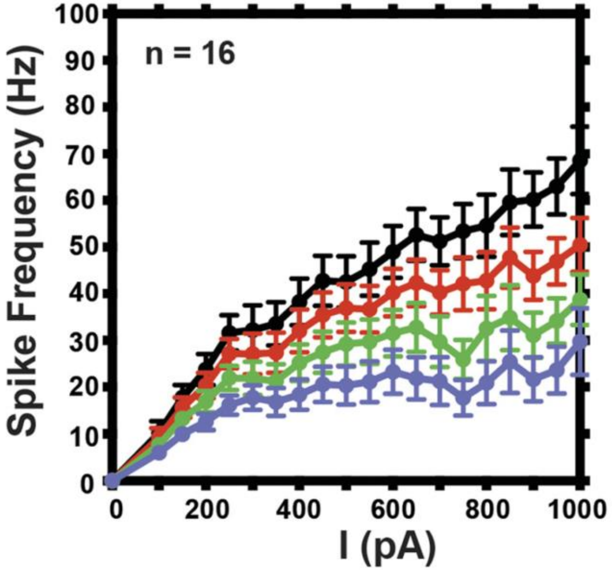


Neurons are non-linear and synapses are valves with memory

- **Brain**



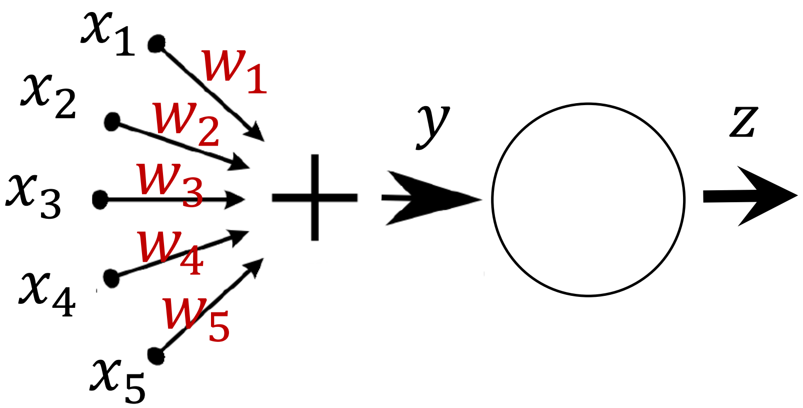




D. Guan et al, J Neurophysiol. 113, 2014 (2015)

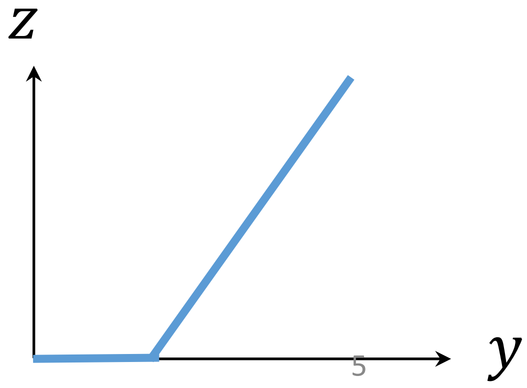
- **Most neural networks today**

Real values

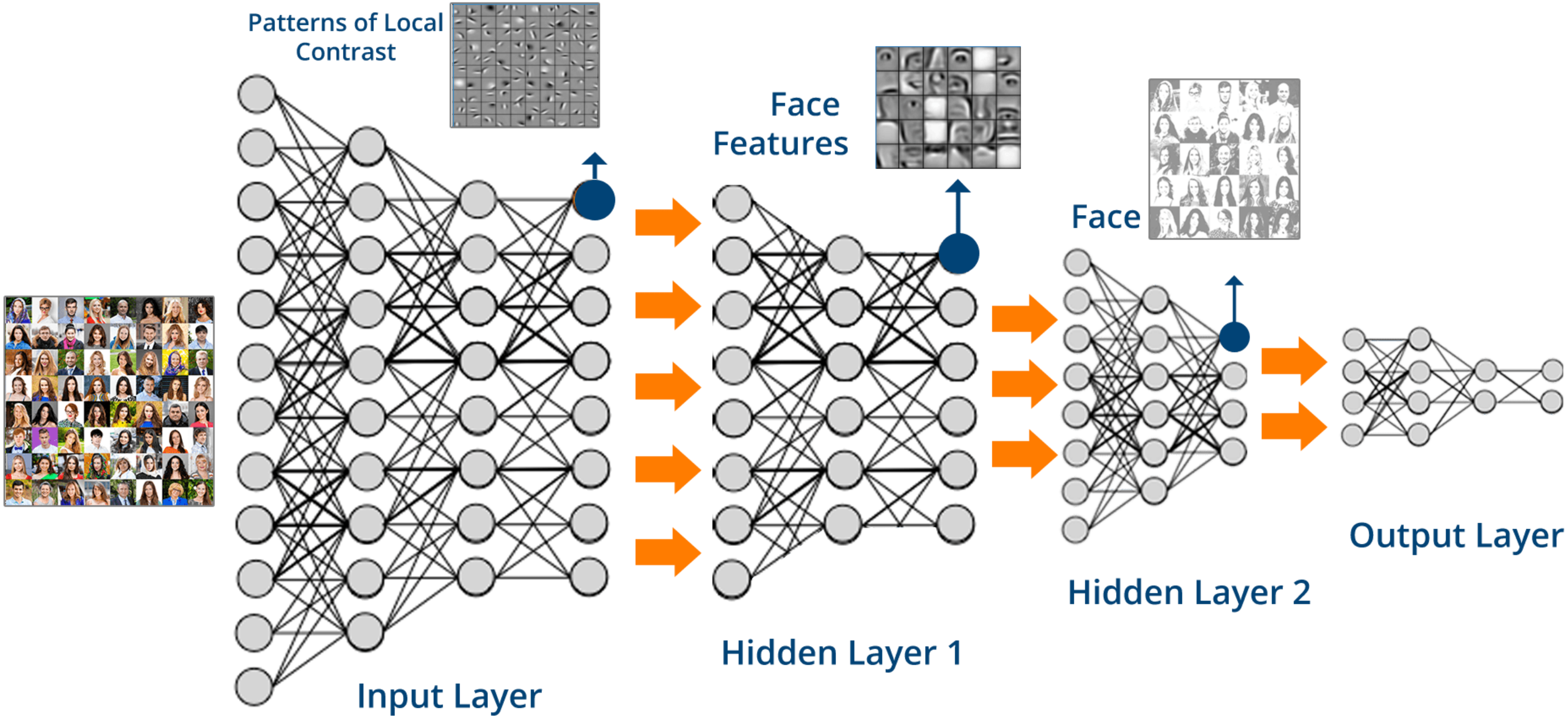


$$y = \sum w_i x_i$$

is called a Multiply and Accumulate (**MAC**) operation



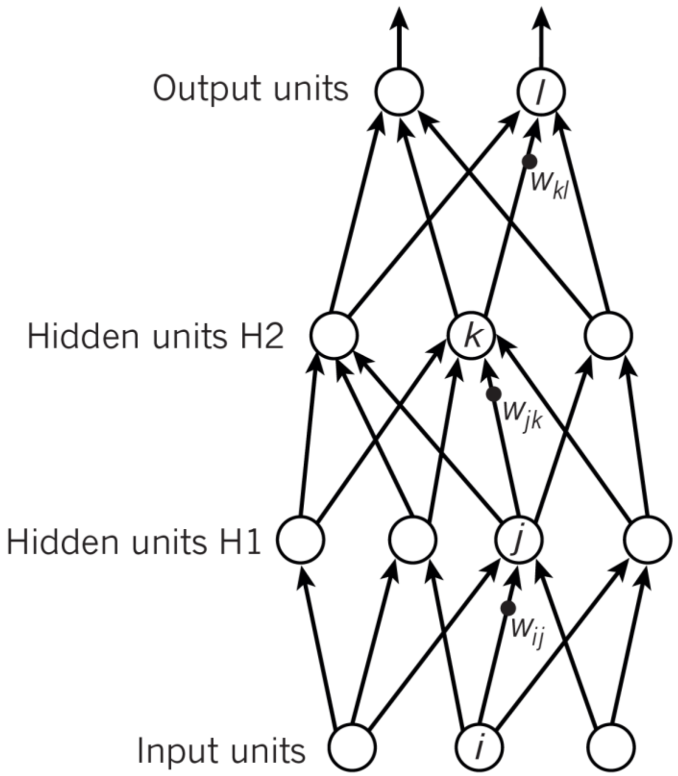
State-of-the-art neural networks are deep: they extract features layer by layer



High performance neural networks are trained through backpropagation of errors

$$\Delta w = -\alpha \frac{\partial E}{\partial w}$$

Forward pass: inference



$$y_l = f(z_l)$$

$$z_l = \sum_{k \,\varepsilon\, \mathrm{H2}} w_{kl}\, y_k$$

$$y_k = f(z_k)$$

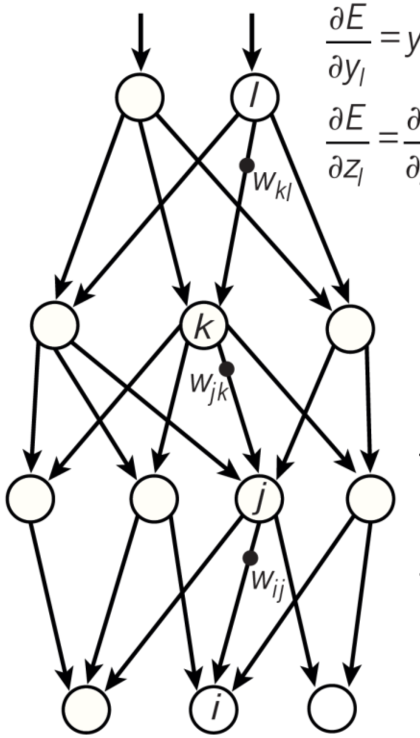
$$z_k = \sum_{j \,\varepsilon\, \mathrm{H1}} w_{jk}\, y_j$$

$$y_j = f(z_j)$$

$$z_j = \sum_{i \,\varepsilon\, \mathrm{Input}} w_{ij}\, x_i$$

Backward pass

Compare outputs with correct answer to get error derivatives



$$\frac{\partial E}{\partial y_l} = y_l - t_l$$

$$\frac{\partial E}{\partial z_l} = \frac{\partial E}{\partial y_l}\frac{\partial y_l}{\partial z_l}$$

$$\frac{\partial E}{\partial y_k} = \sum_{l \,\varepsilon\, \mathrm{out}} w_{kl}\frac{\partial E}{\partial z_l}$$

$$\frac{\partial E}{\partial z_k} = \frac{\partial E}{\partial y_k}\frac{\partial y_k}{\partial z_k}$$

$$\frac{\partial E}{\partial y_j} = \sum_{k \,\varepsilon\, \mathrm{H2}} w_{jk}\frac{\partial E}{\partial z_k}$$

$$\frac{\partial E}{\partial z_j} = \frac{\partial E}{\partial y_j}\frac{\partial y_j}{\partial z_j}$$

Yann Lecun, Yoshua Bengio and Geoffrey Hinton, *Nature* 521, 436 (2015)

The challenges of neuromorphic computing today are:

- To fabricate optimized synapses and neurons
- To build deep networks (successive MACs)
- To implement backpropagation

Filamentary switching

Phase-change

Organics

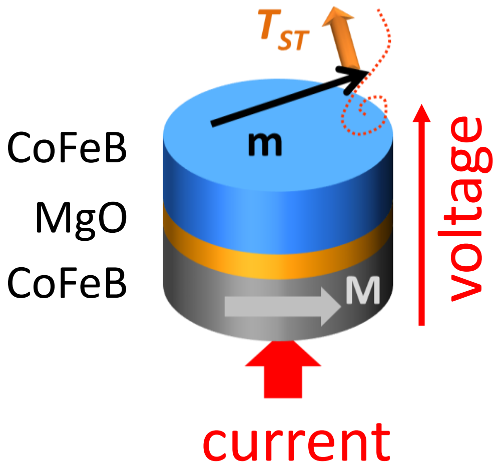
Optics

Ferroelectrics

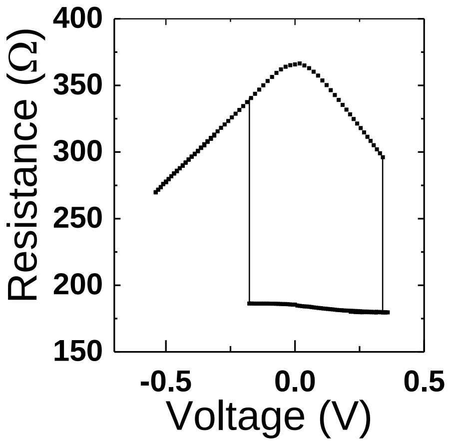
Spintronics neural nets

Spintronics is a toobox for neuromorphic computing

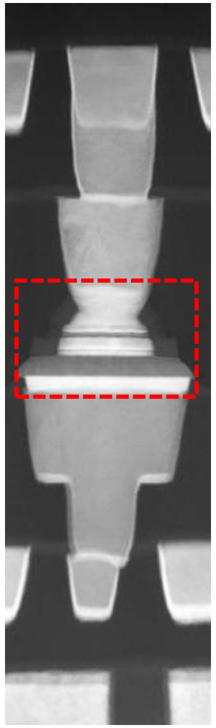
Magnetic tunnel junction



Binary memory

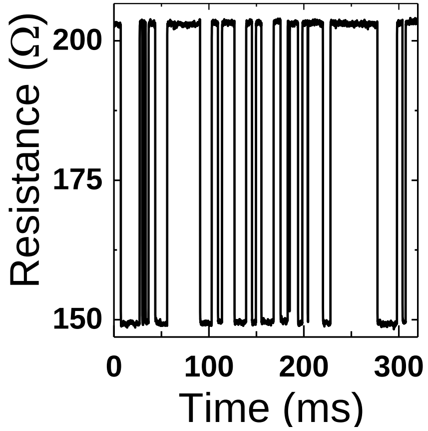


28nm Logic



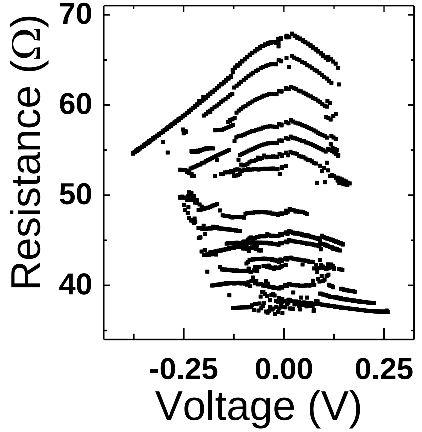
Samsung IEDM 2018

Alternative materials



Stochastic oscillator

Magnetic textures



Memristor

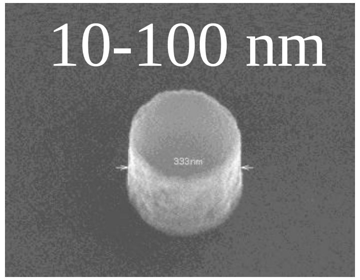
→ non-volatile memory and multifunctionality

J. Grollier et al, “Neuromorphic Spintronics”, Nature Electronics (2020)

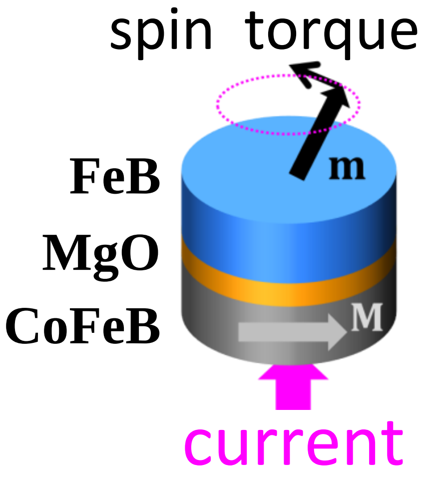
Spin-torque nano-oscillators can be used as radio-frequency neurons

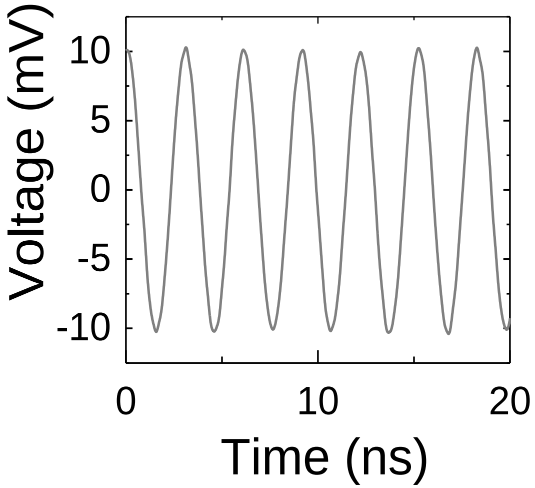
Nanoscale, fast (GHz), non-linear and easily measurable

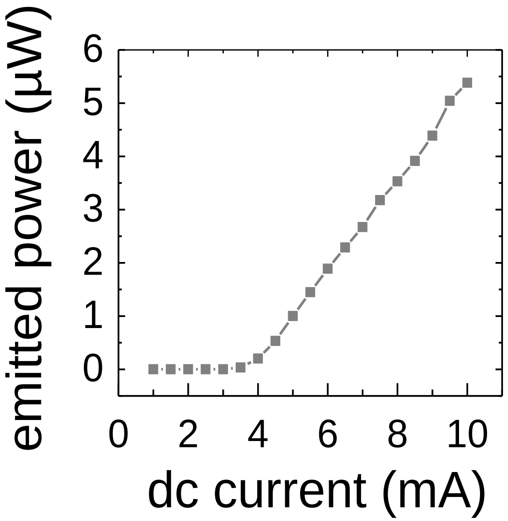
magnetic tunnel junction



compatible with CMOS



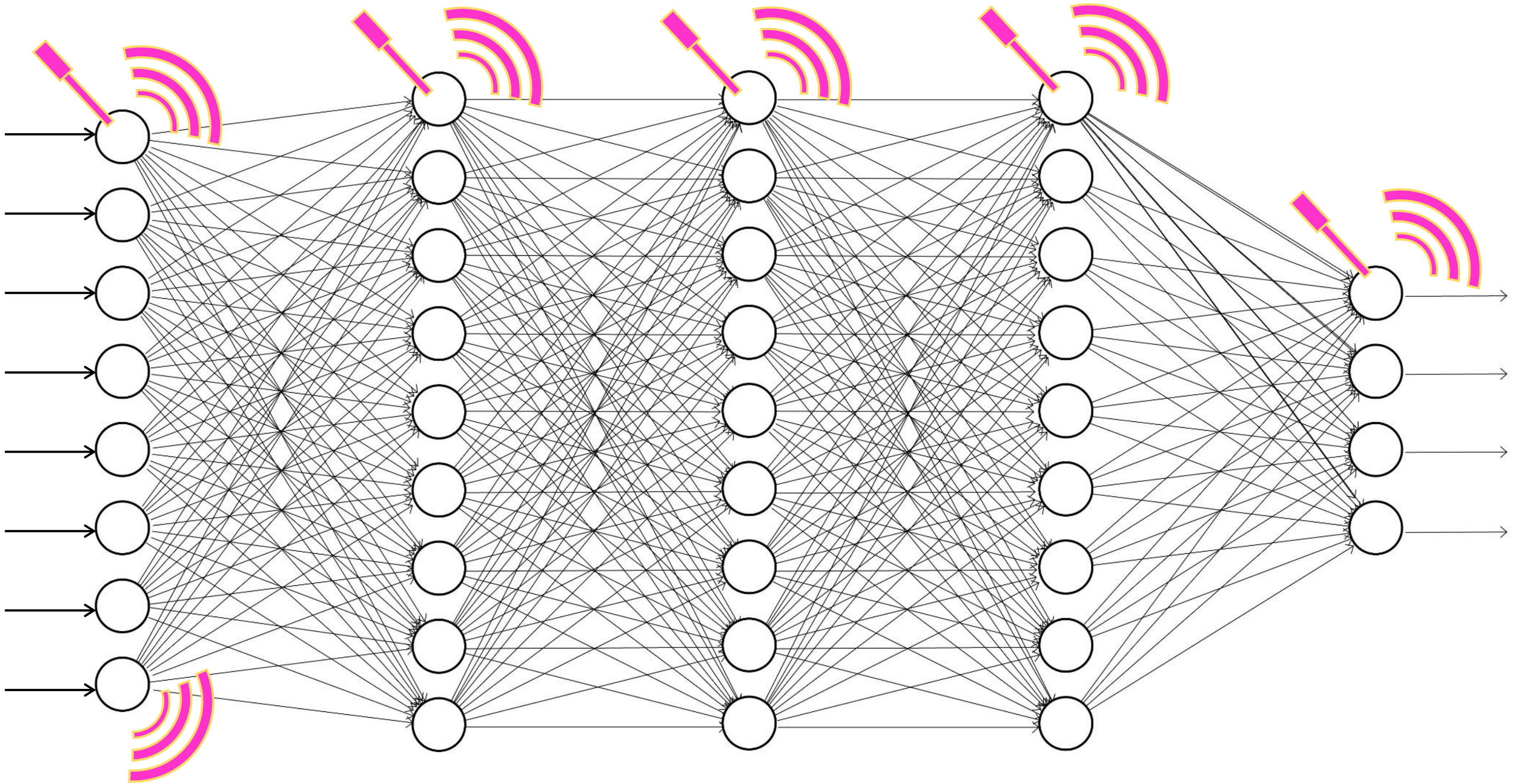




Same structure as magnetic memories

J. Grollier et al, “Neuromorphic Spintronics”, Nature Electronics (2020)

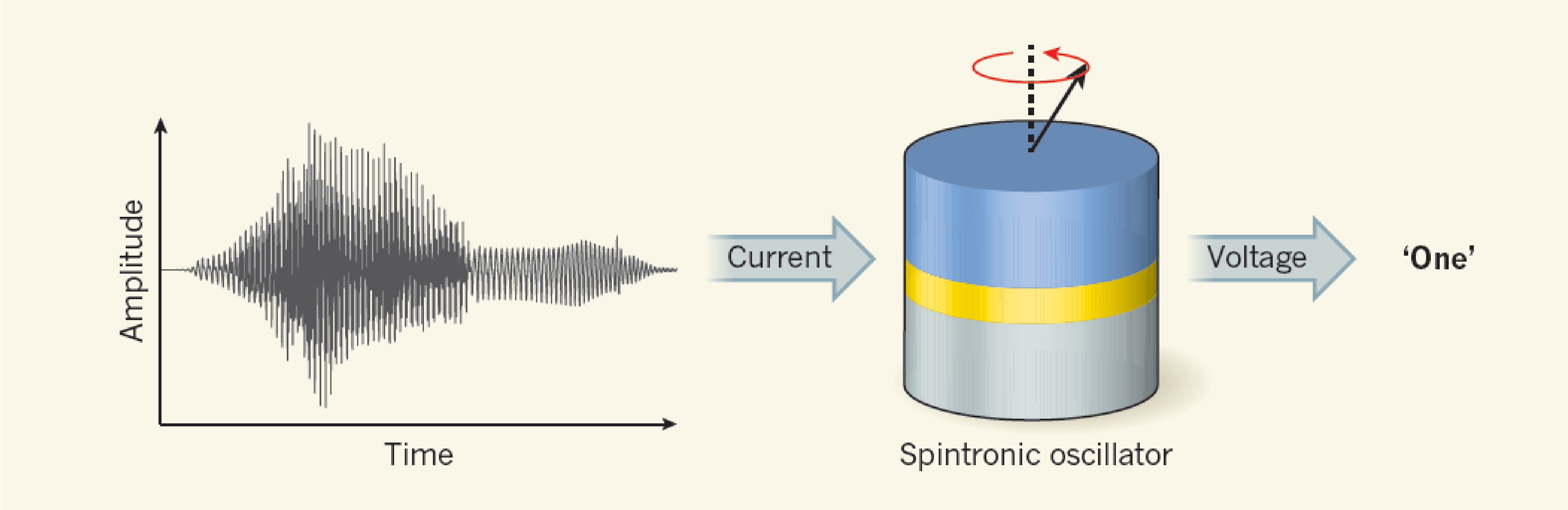
Deep learning through RF communications?



Step 1: Single oscillator

Due to its rich dynamics the nano-oscillator recognizes spoken digits with a success rate > 99.6%

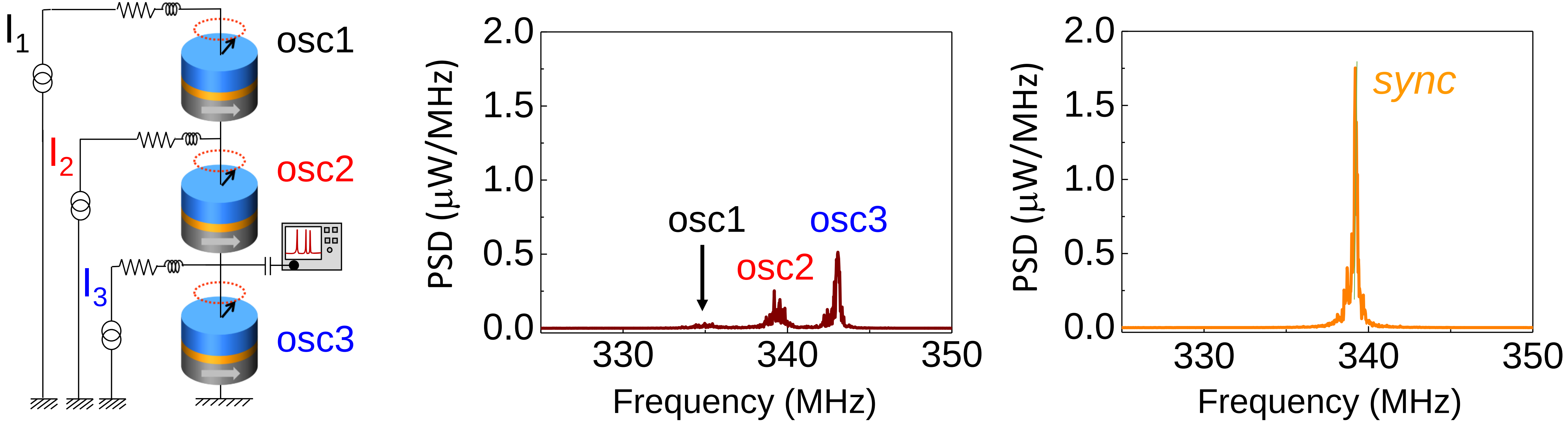
TI-46 database, 5 female speakers, cochlear pre-processing



J. Torrejon, M. Riou, F. Abreu Araujo et al, Nature 547, 428 (2017)

Step 2: single layer neural network

Three nano-oscillators recognize patterns by synchronizing to each other



M. Romera, P. Talatchian et al, arXiv:2001.08044

Miguel Romera

Marie Curie Fellow at
Complutense, Madrid

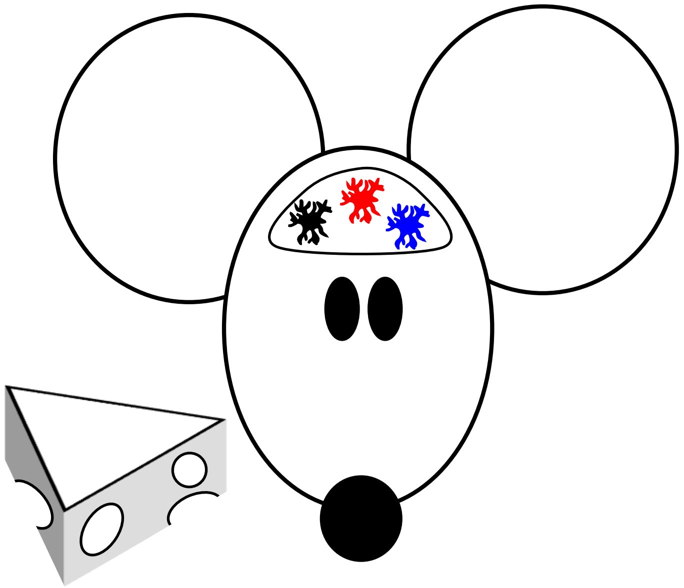
Philippe Talatchian

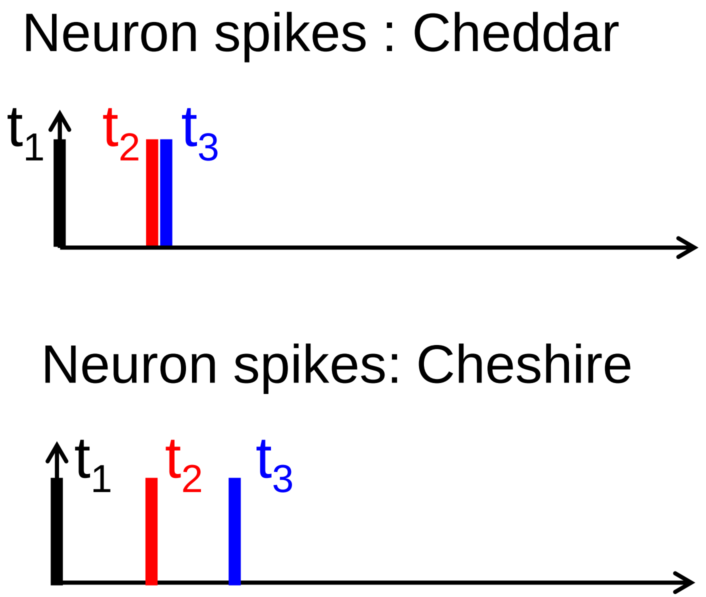
Post-doctoral researcher
at NIST, Gaithersburg

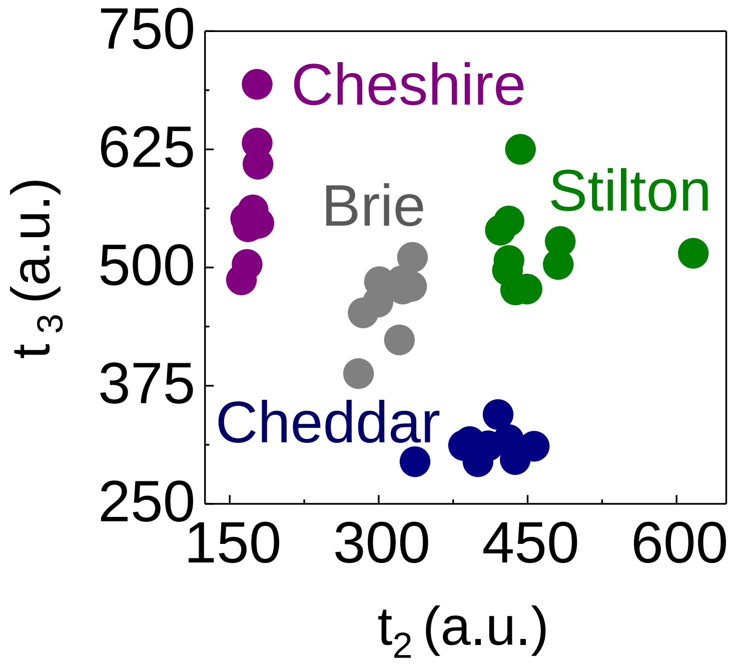
M. Romera, P. Talatchian et al, arXiv:2001.08044

Task for the oscillators: binding temporal events

Neurons 1, 2, 3

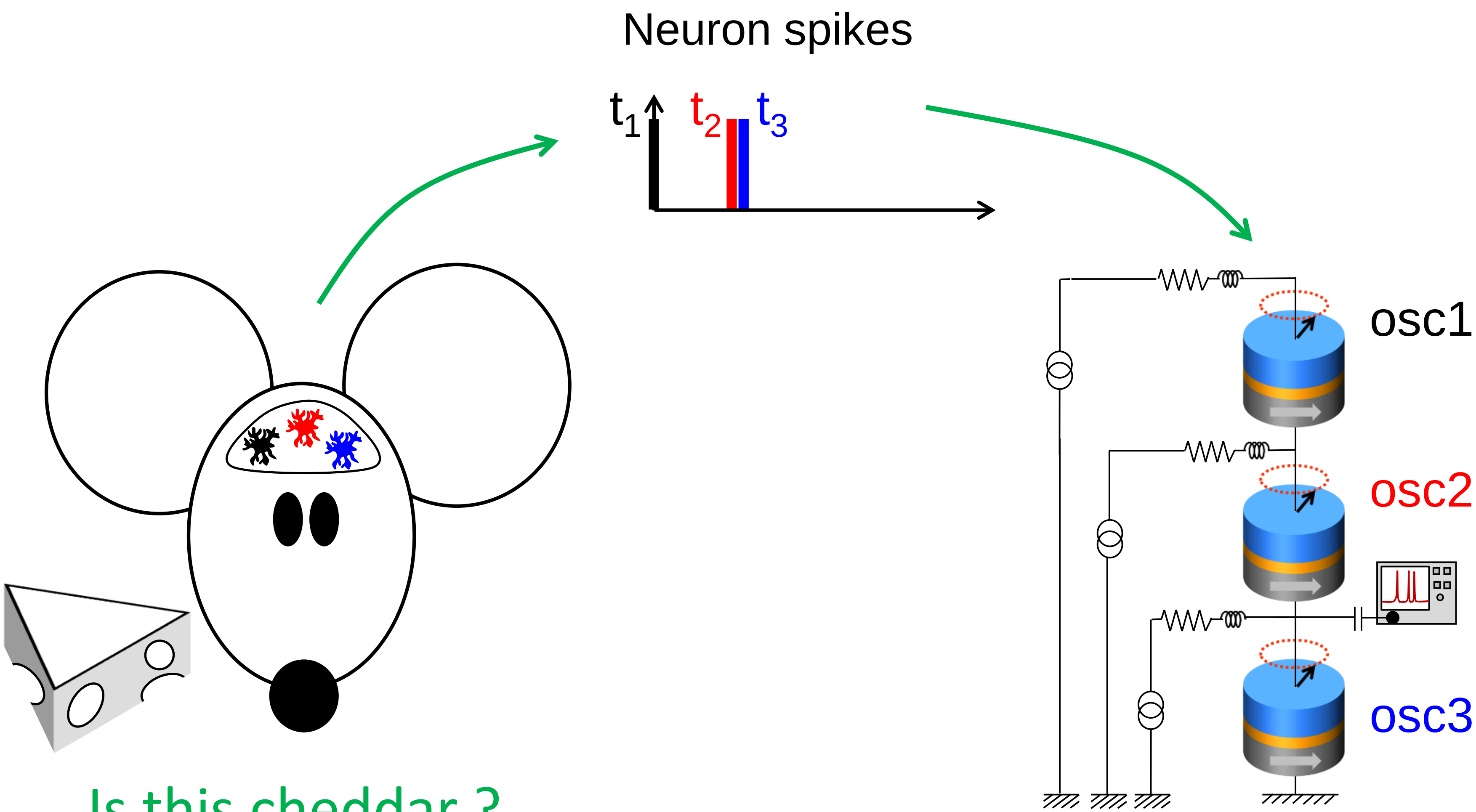






J. J. Hopfield et al, PNAS 98, 1282 (2001)

Each oscillator deals with one spike

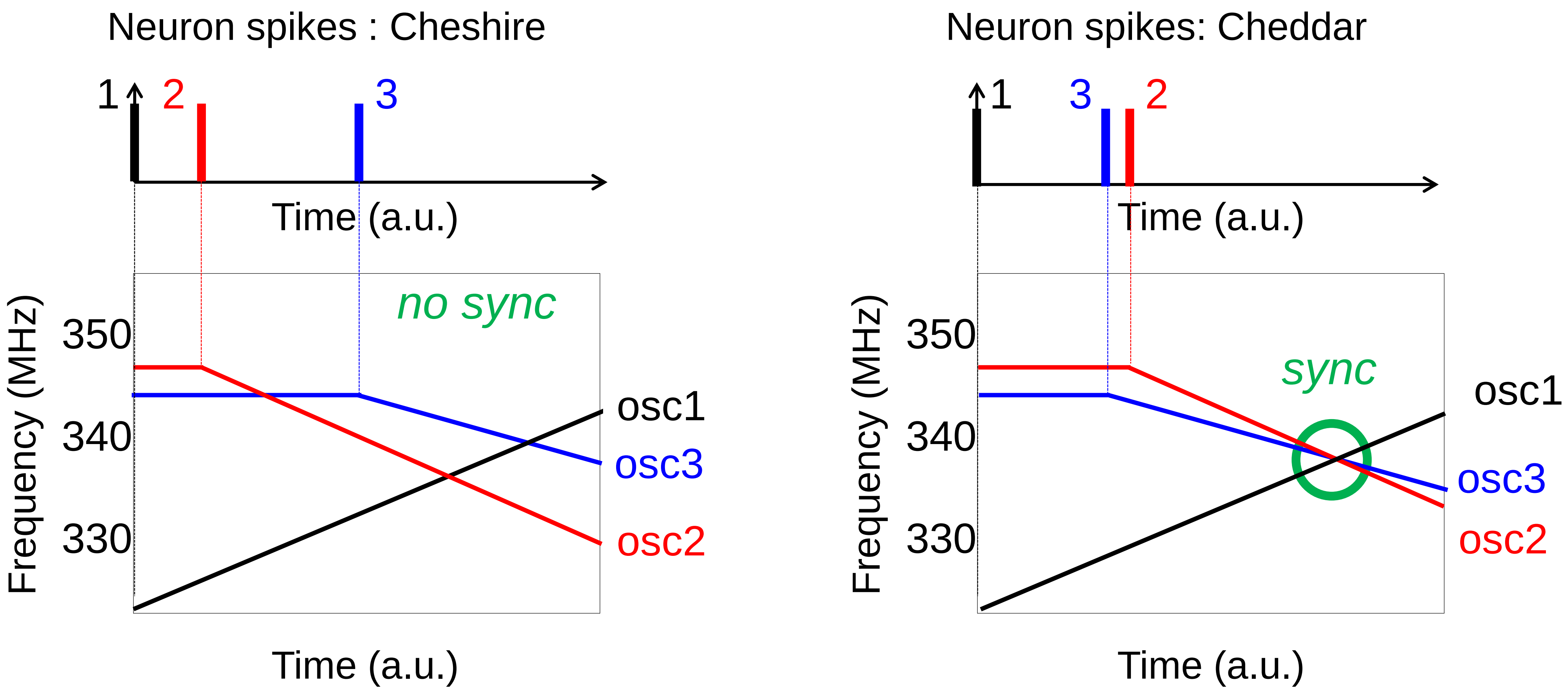


Is this cheddar ?

If the oscillators synchronize, yes, this is cheddar

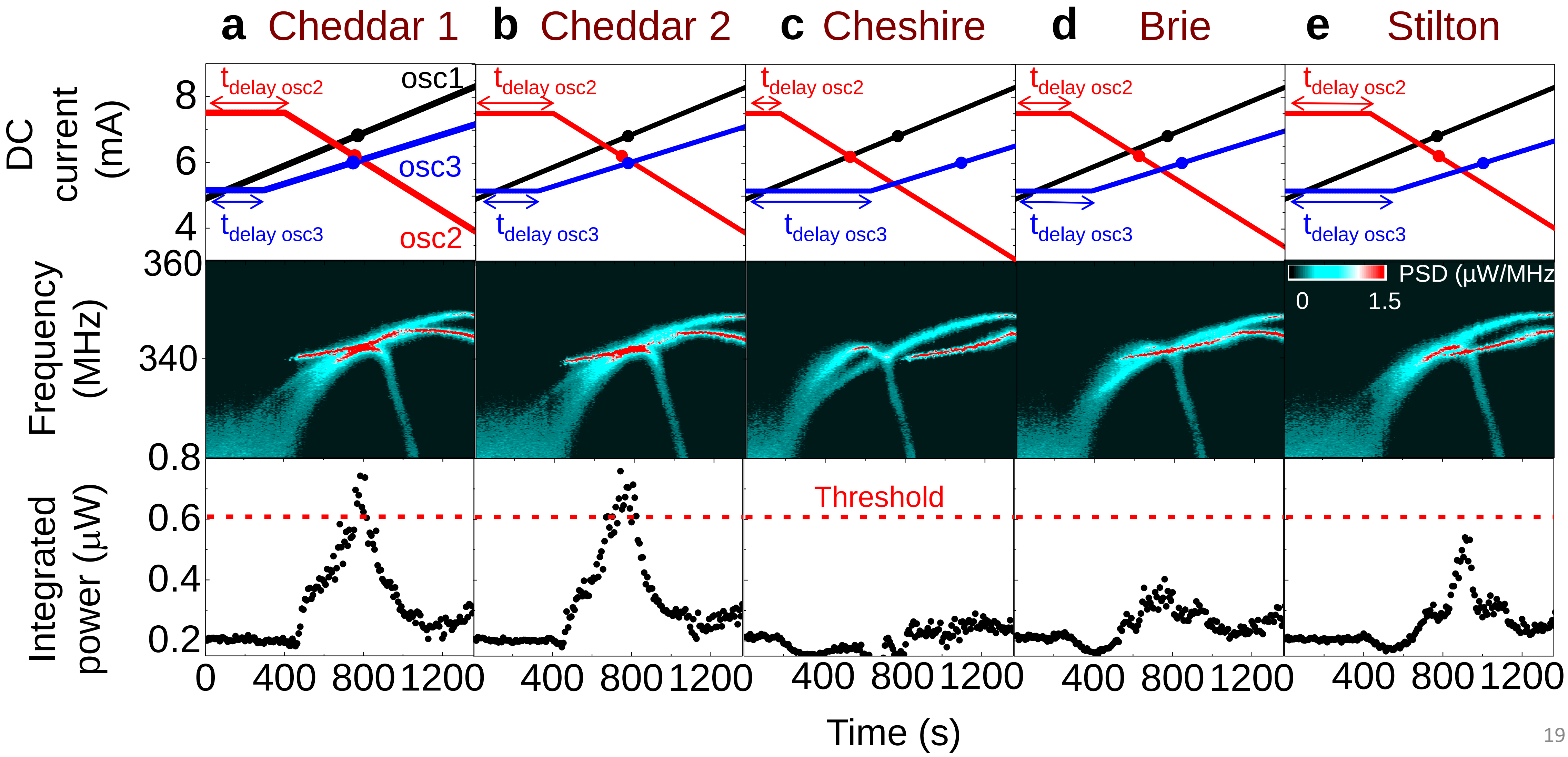
J. J. Hopfield et al, PNAS 98, 1282 (2001)

A ramp is generated when a neuron spikes, mutual synchronization occurs if ramps converge

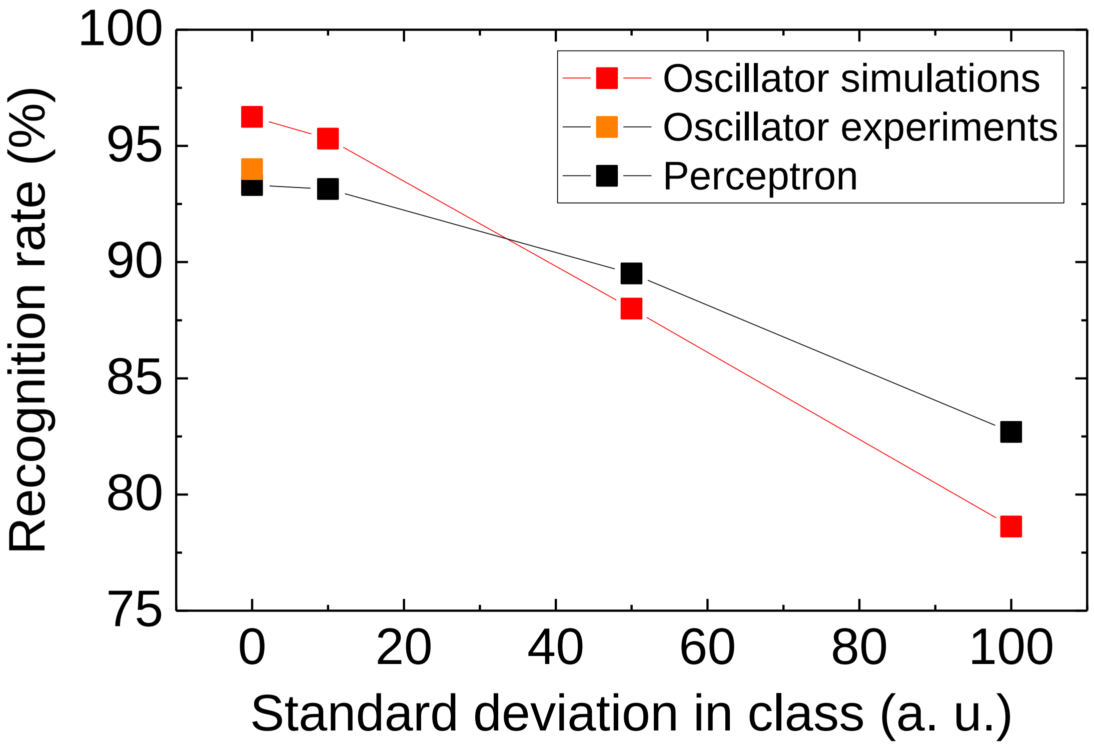


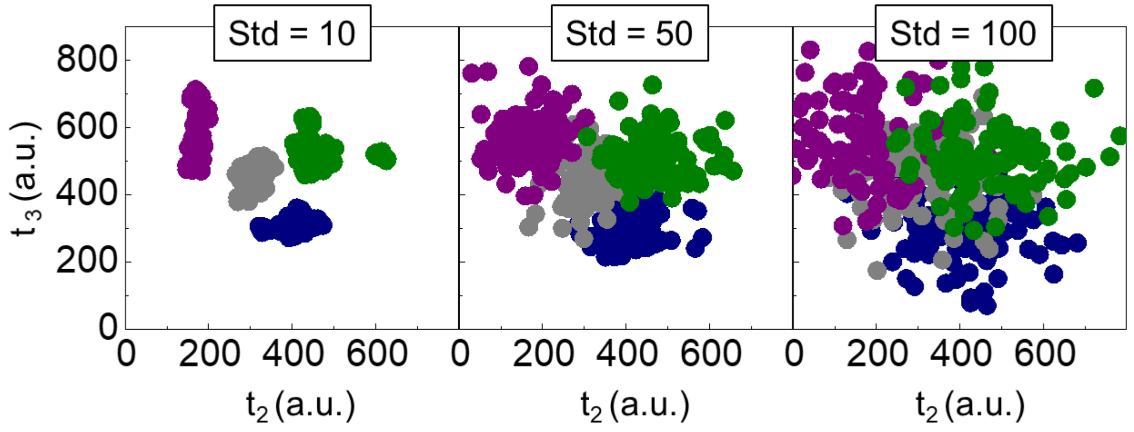
J. J. Hopfield et al, PNAS 98, 1282 (2001)

We train three spintronic oscillators to recognize 'Cheddar'



The recognition rate is close to the perceptron despite the simple learning rule

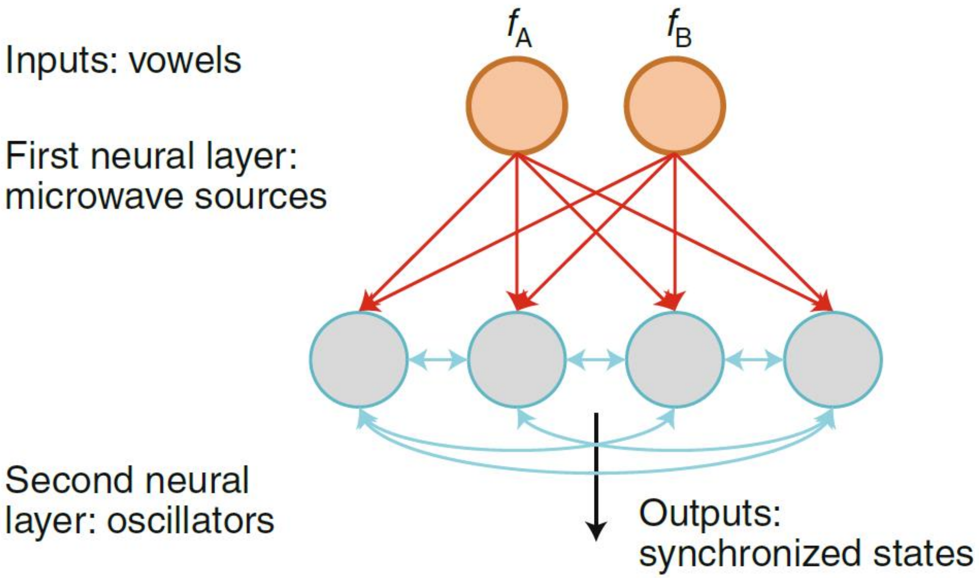


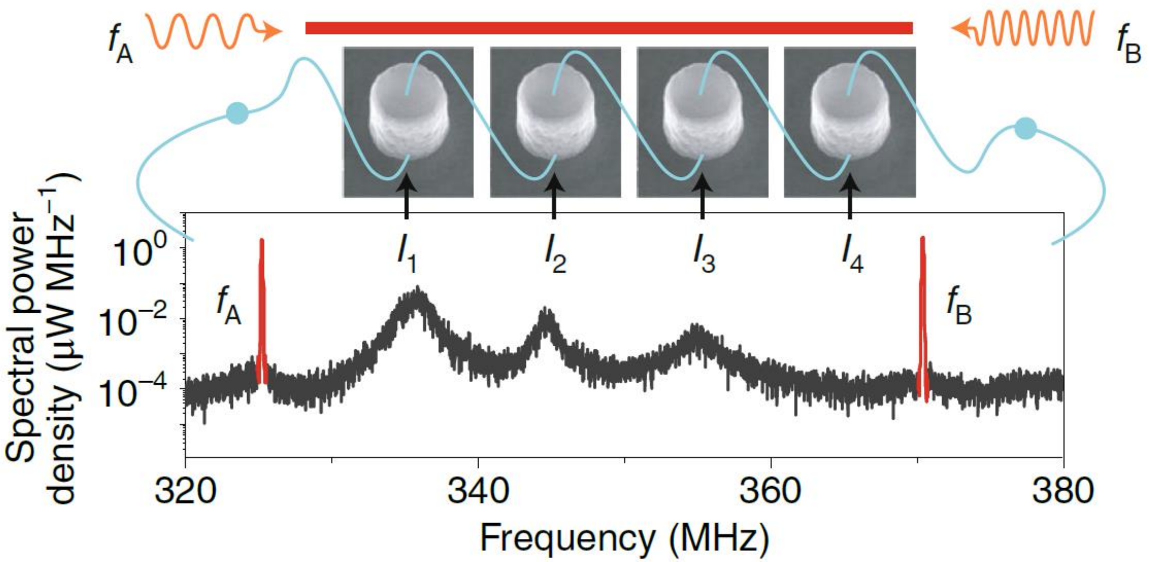


M. Romera, P. Talatchian et al, arXiv:2001.08044

Single layer neural network through the mutual synchronization of spin-torque nano-oscillators

Step 3: RF communication between the two layers of a spin-torque oscillator neural network

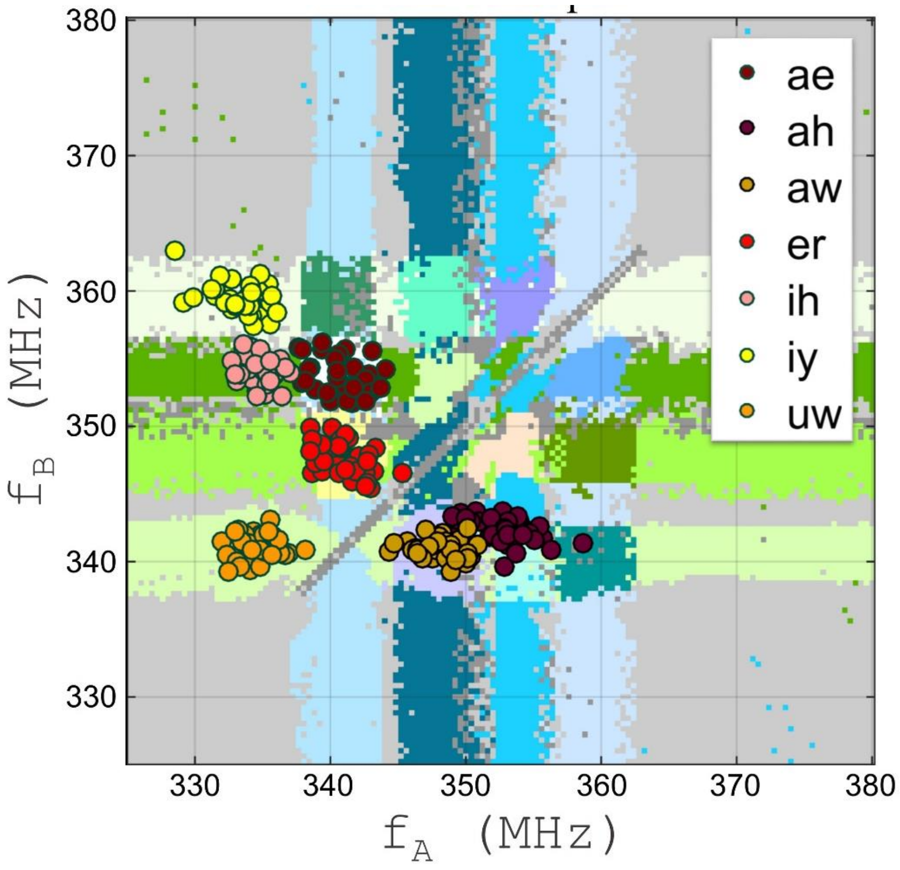




M. Romera, P. Talatchian et al, Nature 563, 230 (2018)

85% success rate

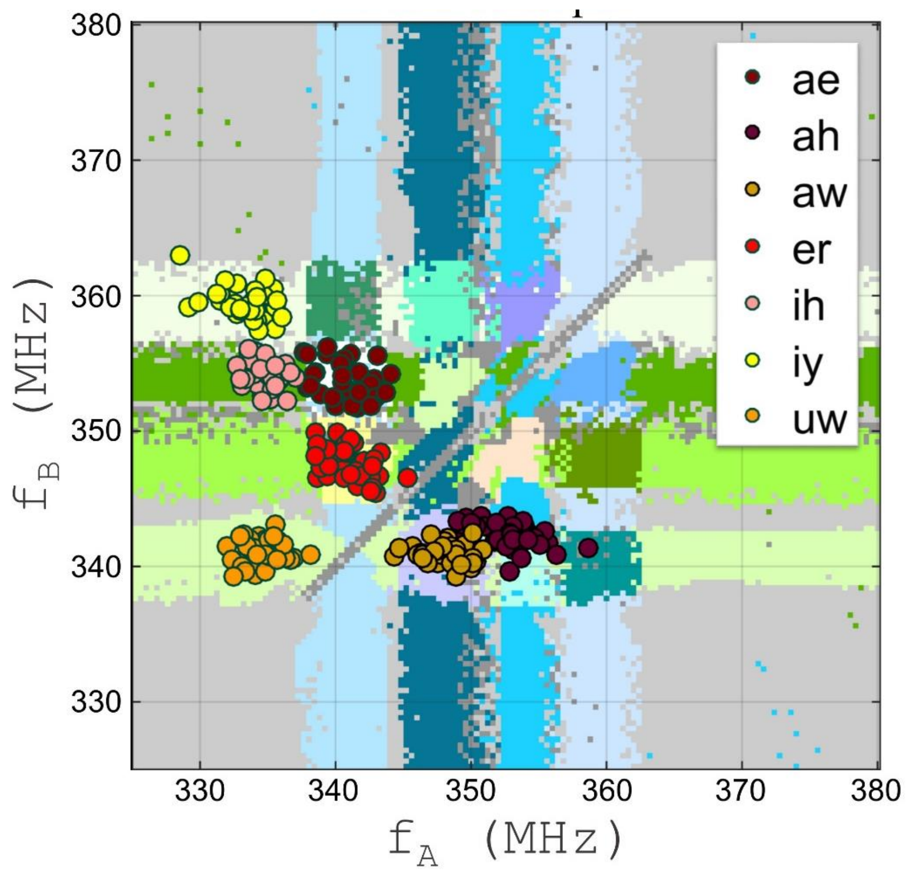
Fixed synapses



M. Romera, P. Talatchian et al, Nature 563, 230 (2018)

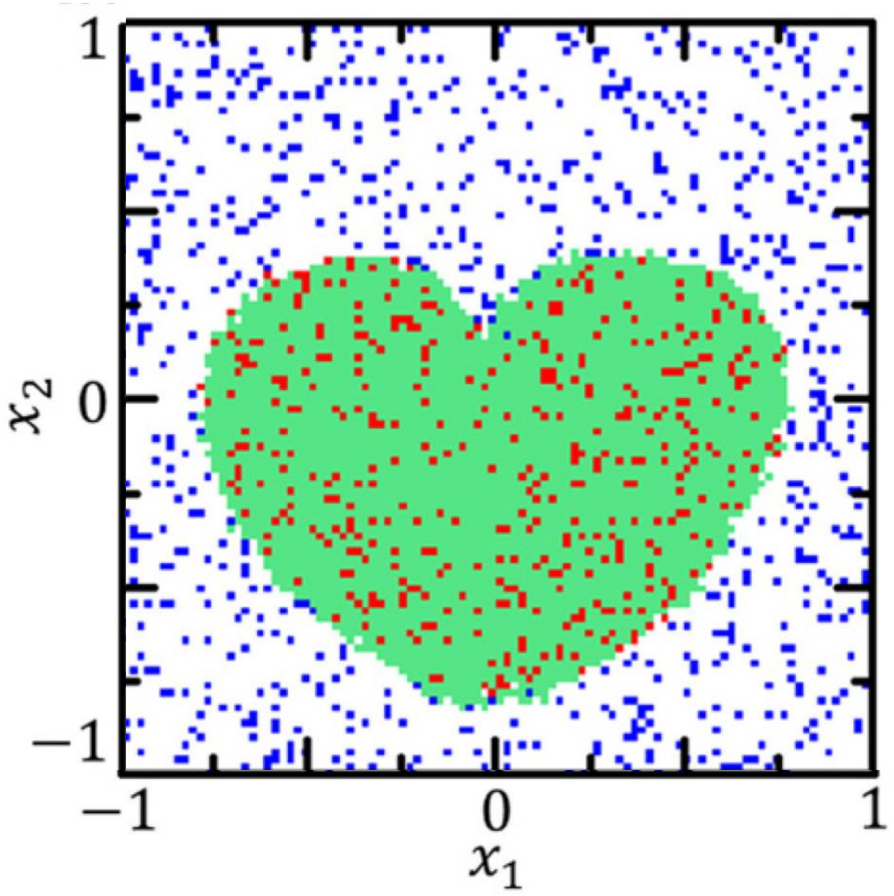
Plastic synapses are necessary to separate data with complicated boundaries

Fixed synapses



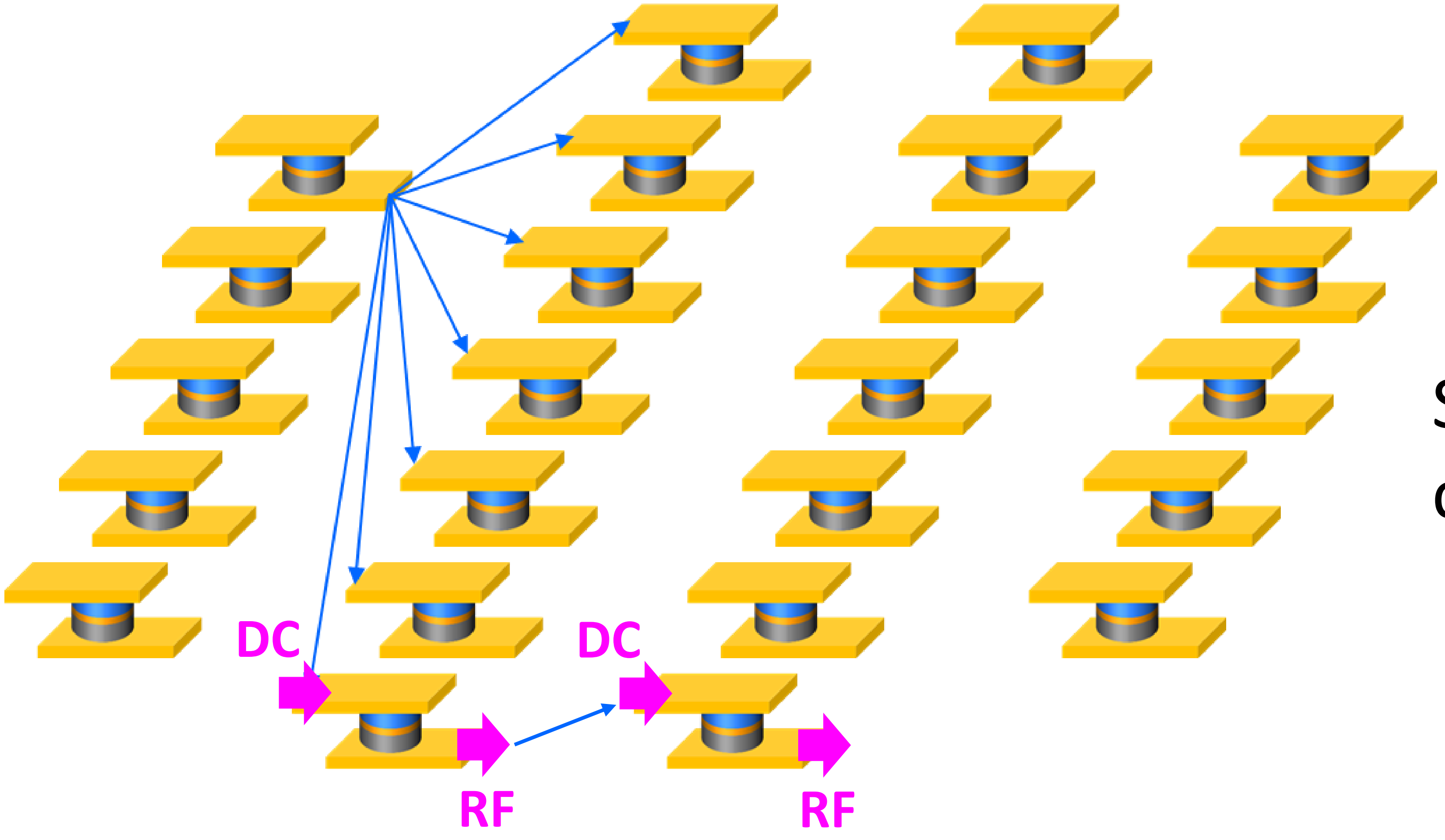
M. Romera, P. Talatchian et al, Nature 563, 230 (2018)

Plastic synapses



D. Vodenicarevic et al, JAP 124, 152117 (2018)

We need trainable synapses and MAC operations to build deep networks



Synapses that convert RF to DC ?

Nathan Leroux

Ph.D Student,
CNRS

Danijela Marković

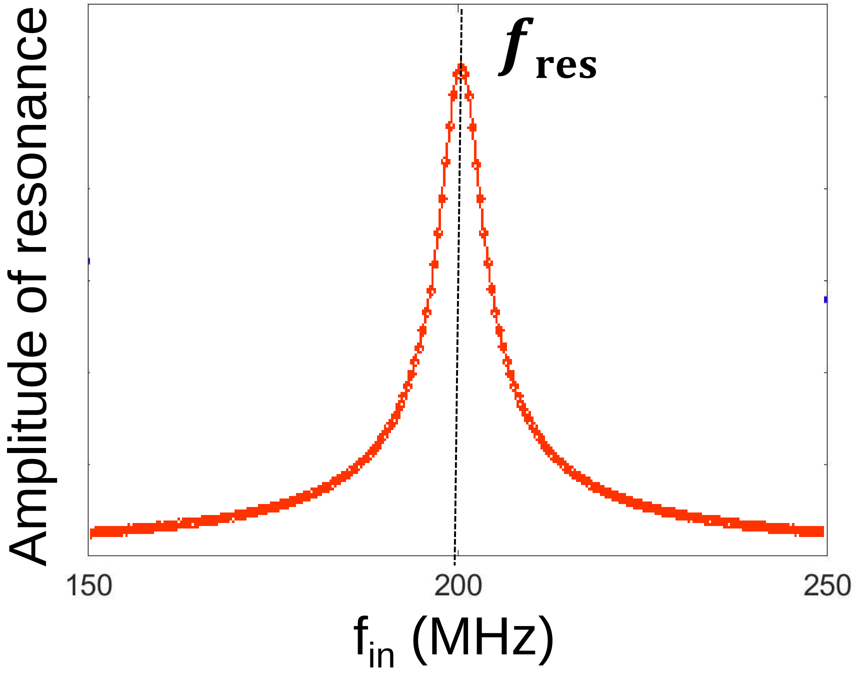
Post-doctoral
researcher, CNRS

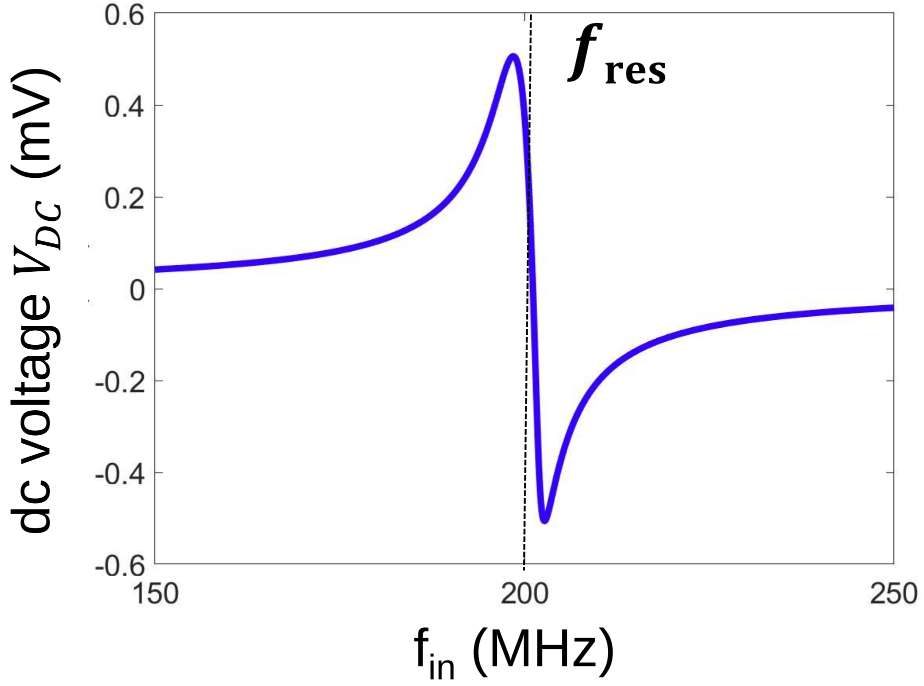
Alice Mizrahi

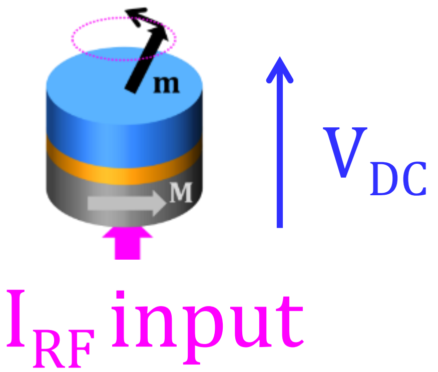
Researcher,
Thales Company

Nathan Leroux, Danijela Marković, Alice Mizrahi et al, in preparation

The spin-diode effect converts RF to DC in magnetic tunnel junctions



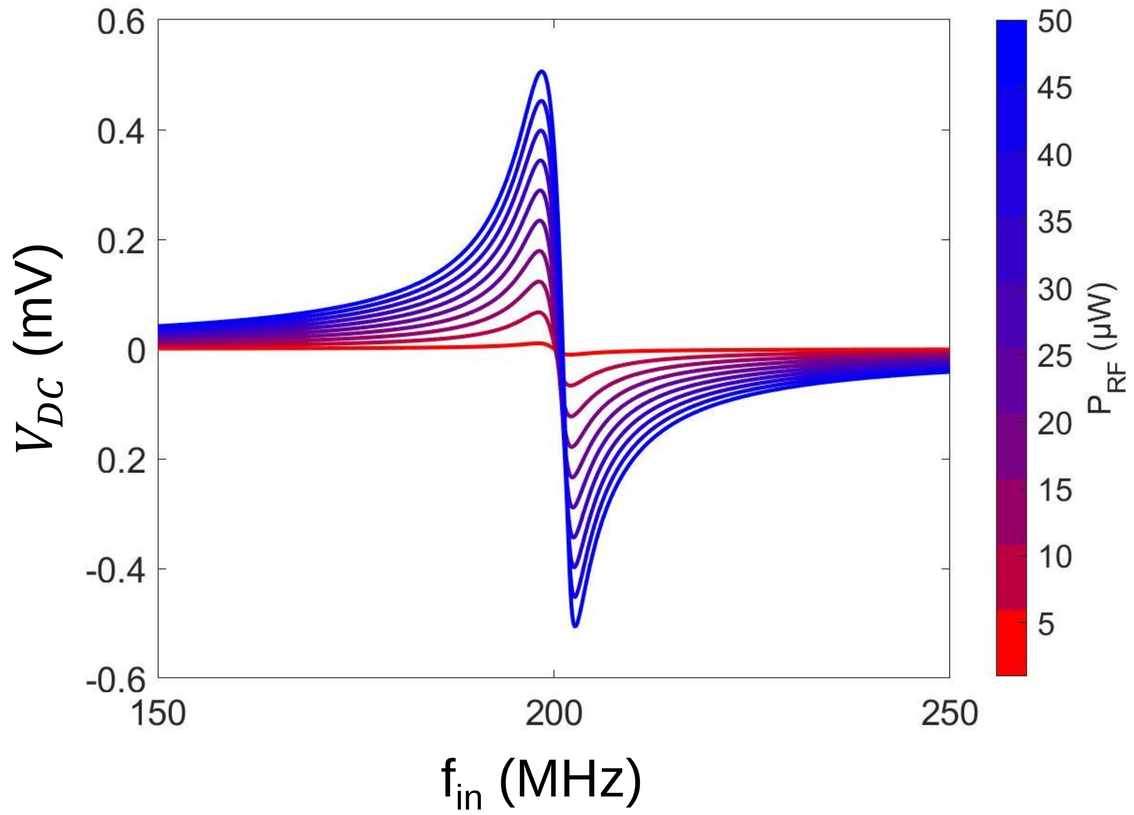


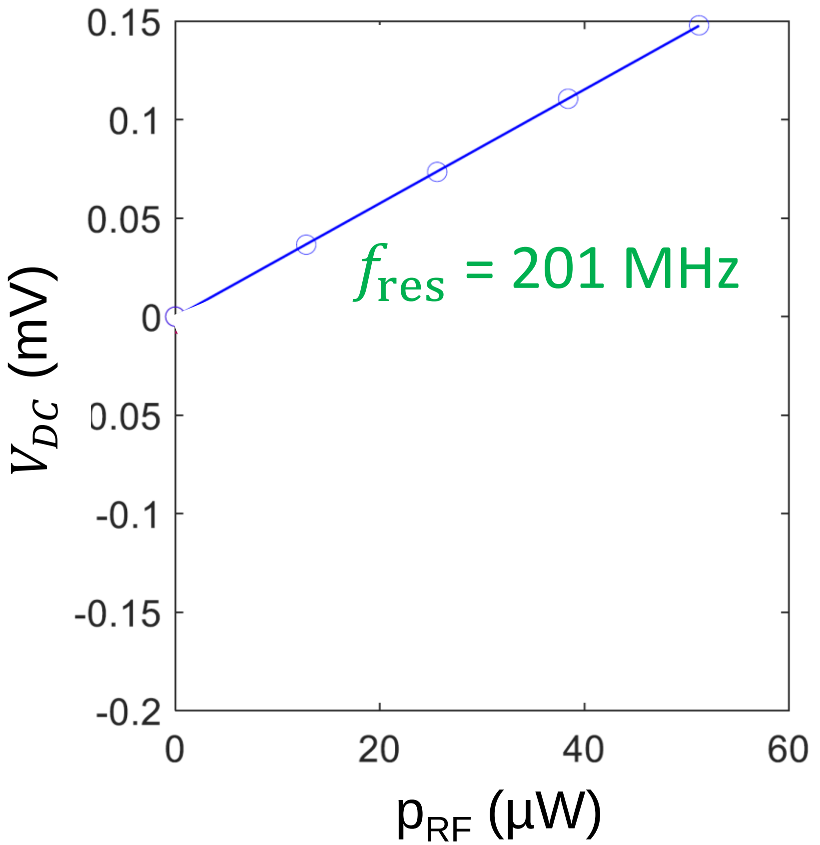


$$I(t) \times R(t) \rightarrow V_{DC}$$

Tulapurkar et al, Nature 2005

The spin-diode effect acts like a synapse

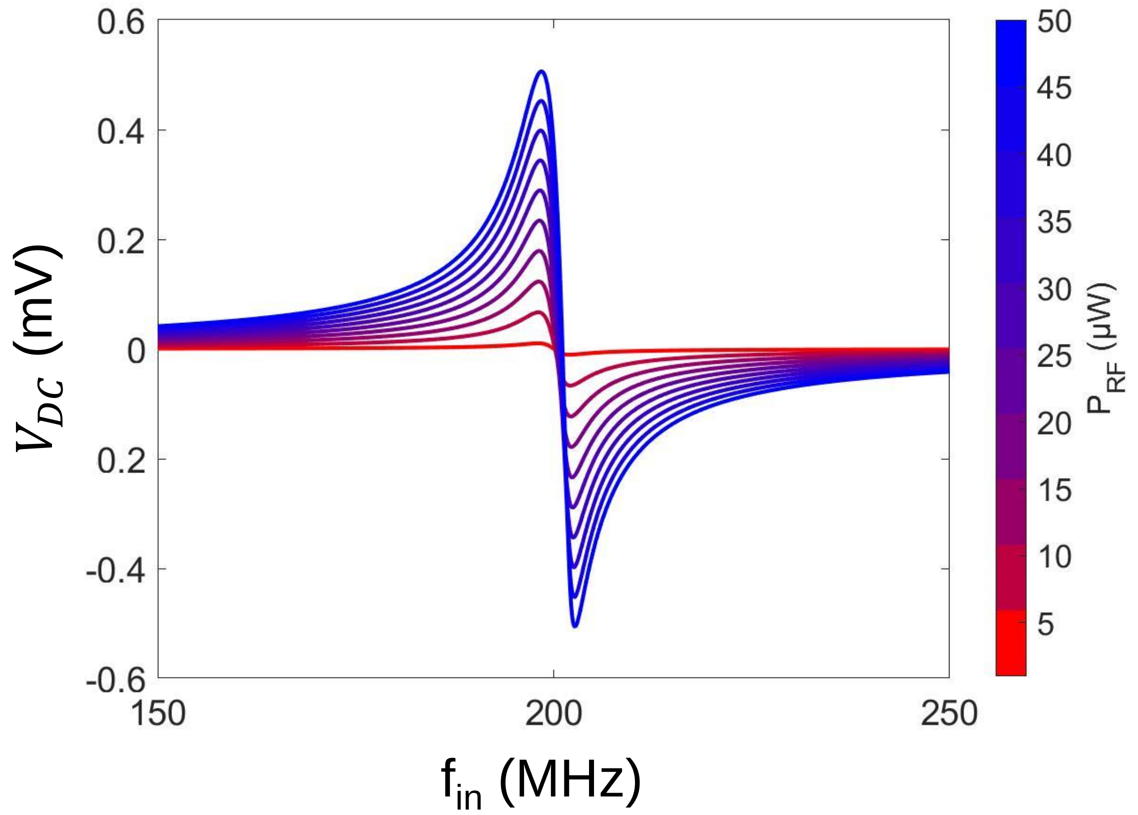


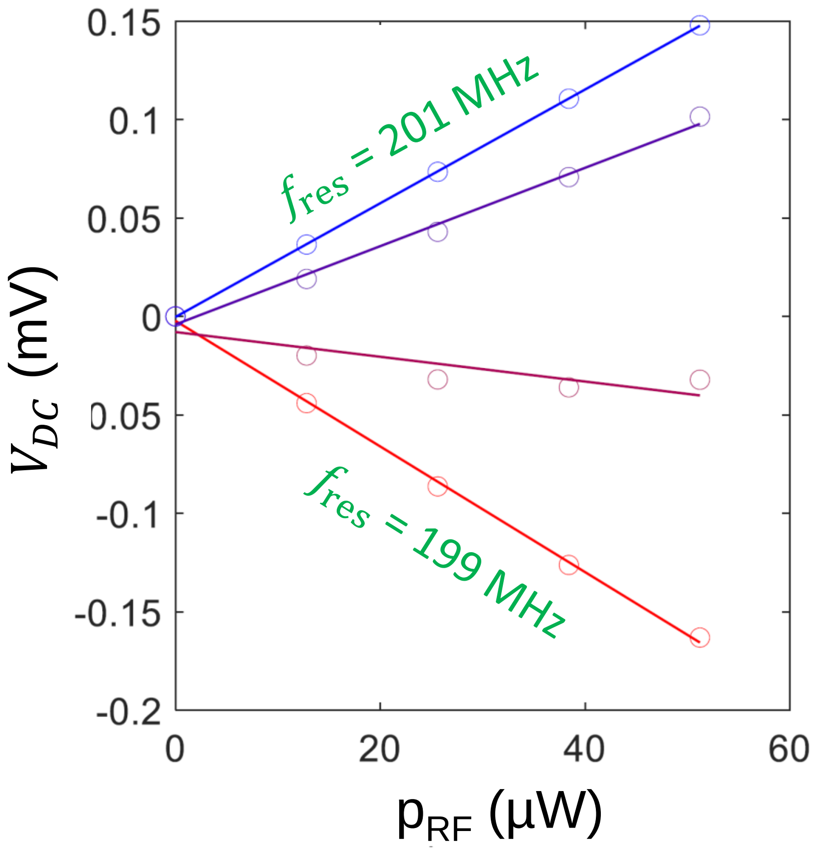


Synaptic multiplication : Output = W x Input

$$V \propto (f_{in} - f_{res}) \times P_{RF}$$

The spin-diode effect acts like a synapse



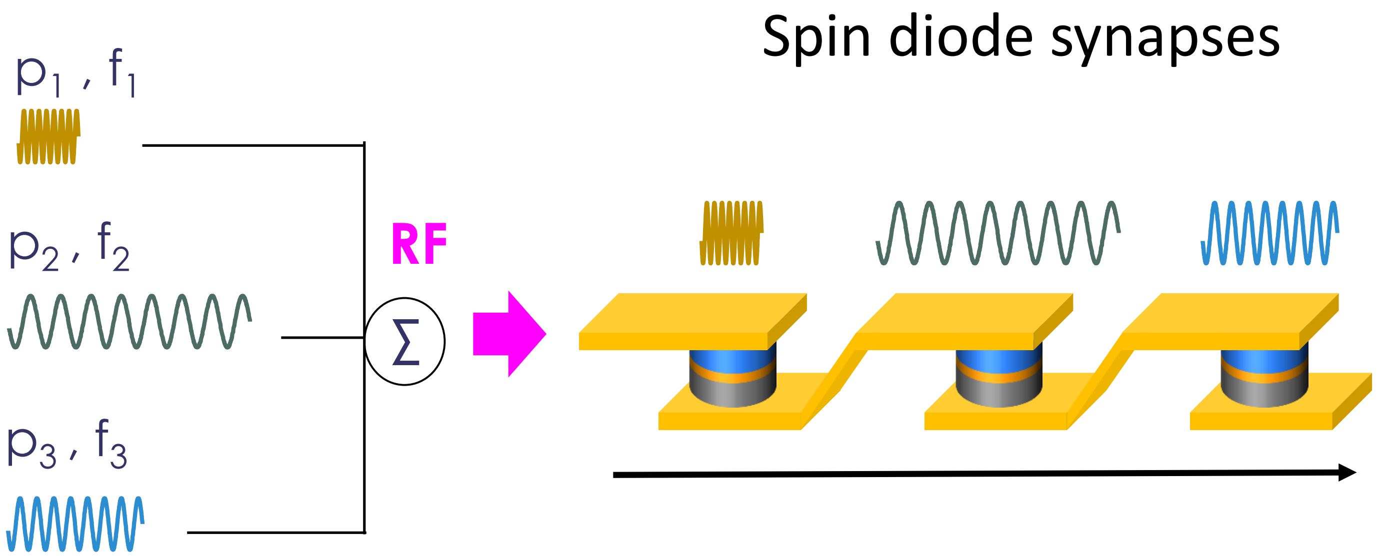


Synaptic multiplication : Output = W x Input

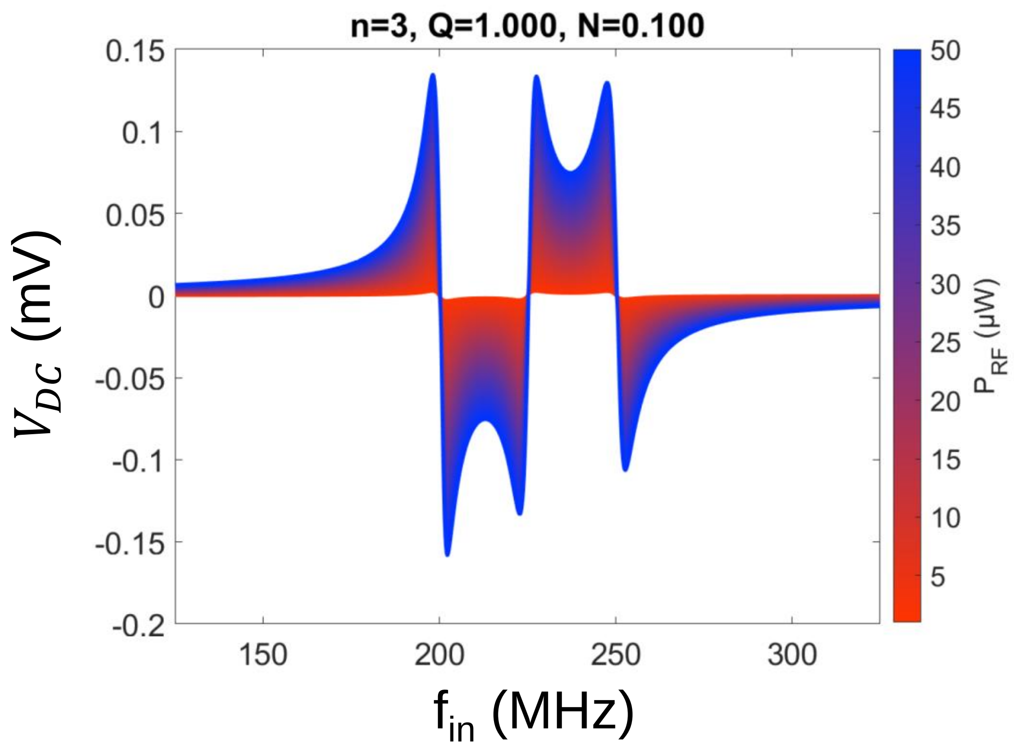
$$V \propto (f_{in} - f_{res}) \times P_{RF}$$

We can use for synapses the same junctions as neurons!

The MAC operation (Multiply and Accumulate) can be achieved through frequency multiplexing

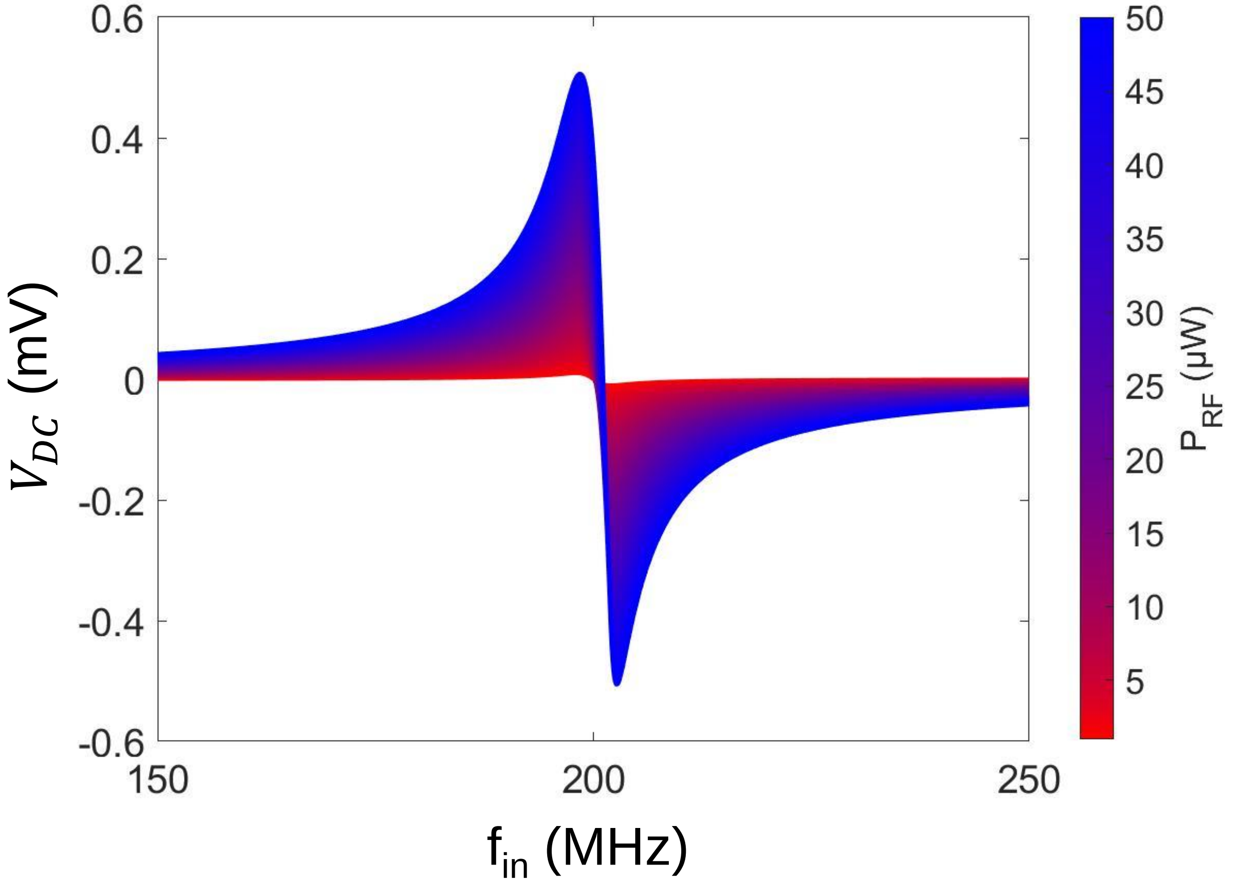


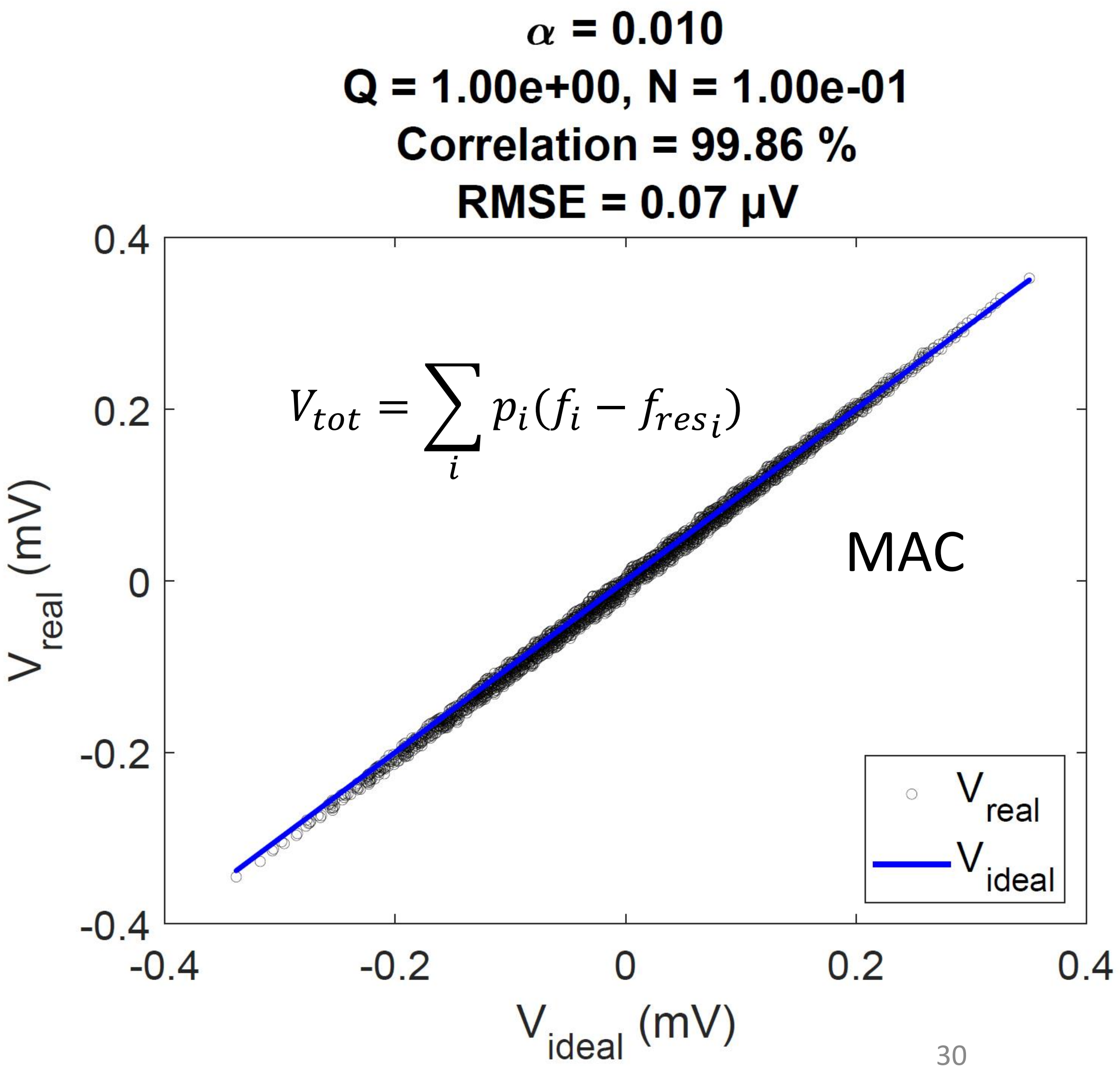
$$V_{tot} = \sum_i V_{DC\,i} = \sum_i p_i(f_i - f_{res_i})$$



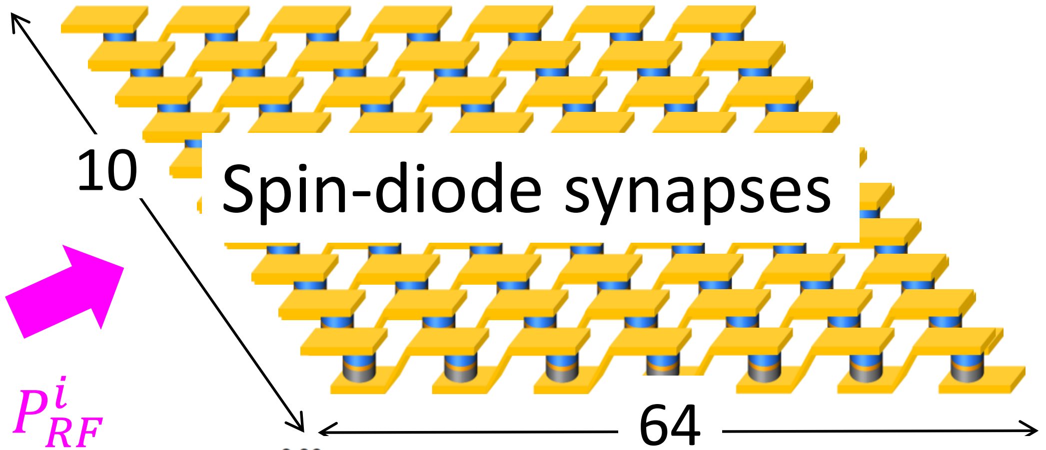
Simulations show that the MAC operation can be achieved even with non-linearities

$$V \propto \left(f_{\text{in}} - f_{\text{res}}(P_{RF})\right) \times P_{RF}$$

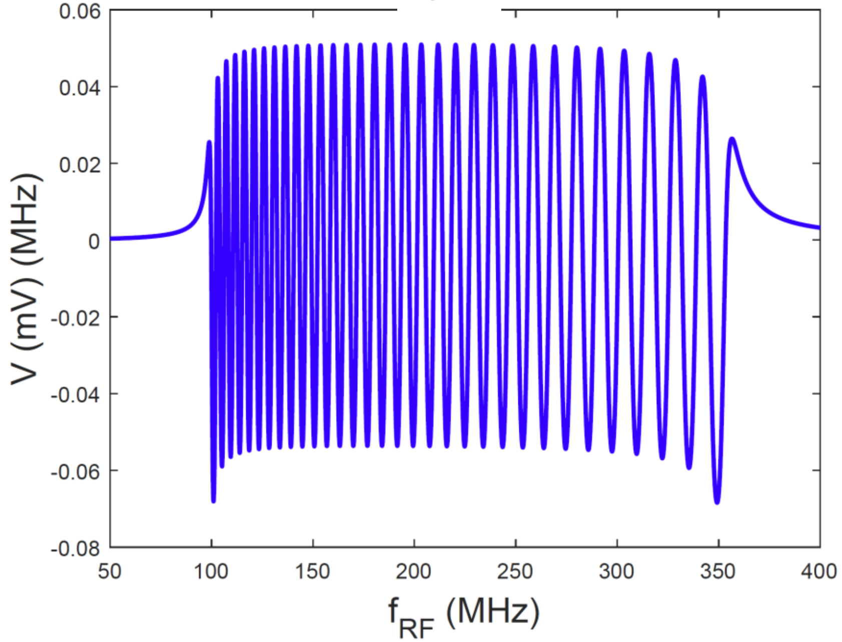




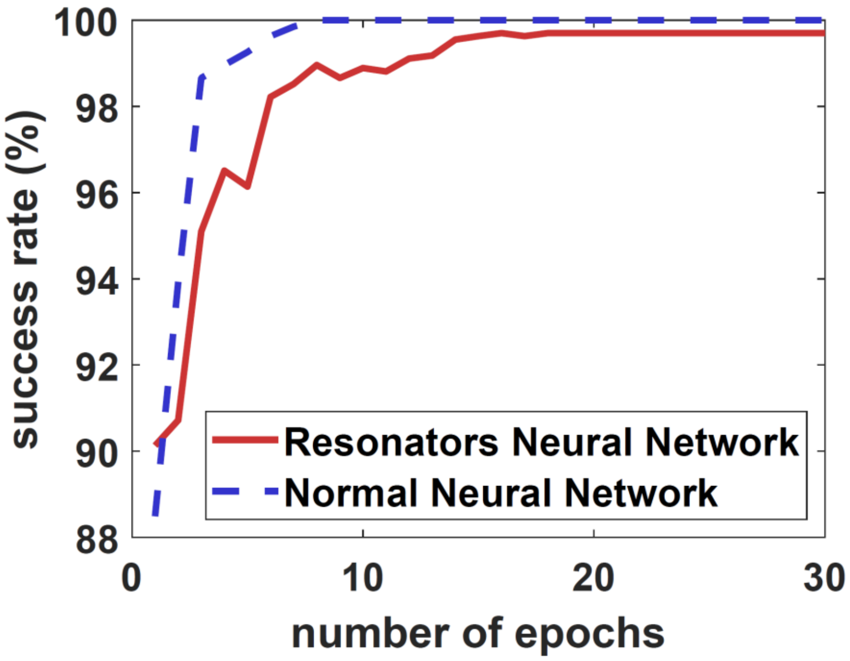
A simulated perceptron with realistic device parameters recognizes digits through backpropagation!



Pixels converted to P_{RF}^{i}

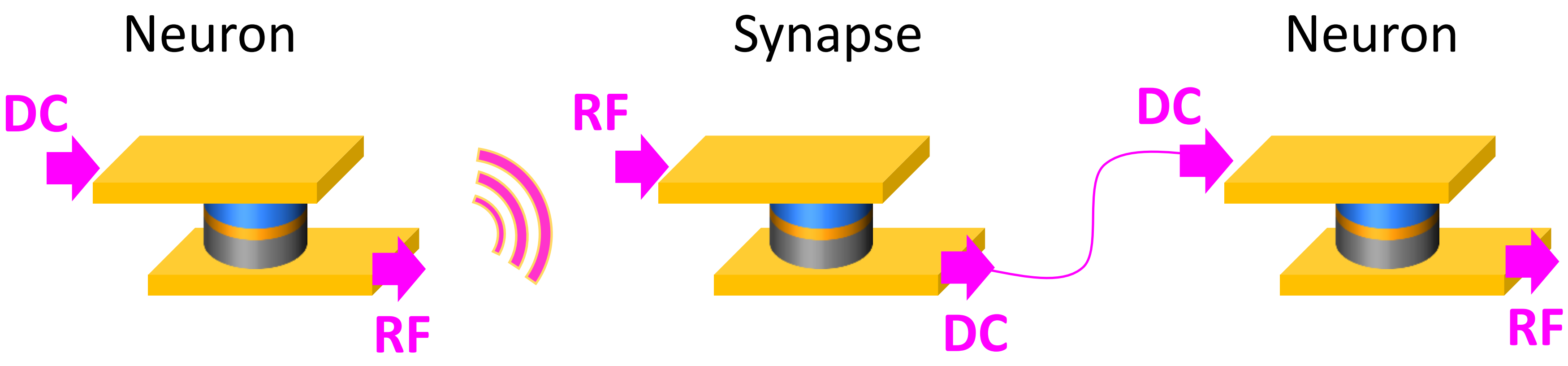


Including non-linearities

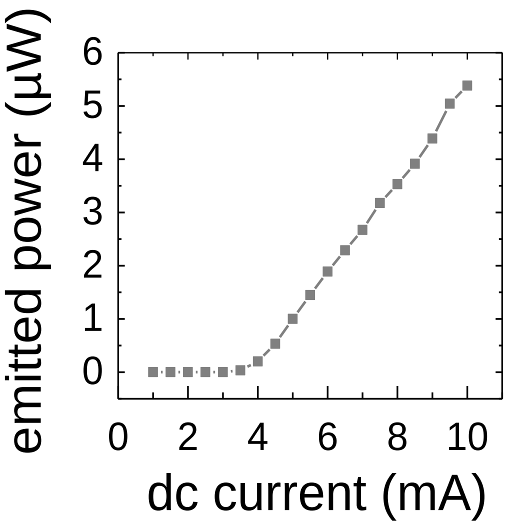


Nathan Leroux, Danijela Marković, Alice Mizrahi et al, in preparation

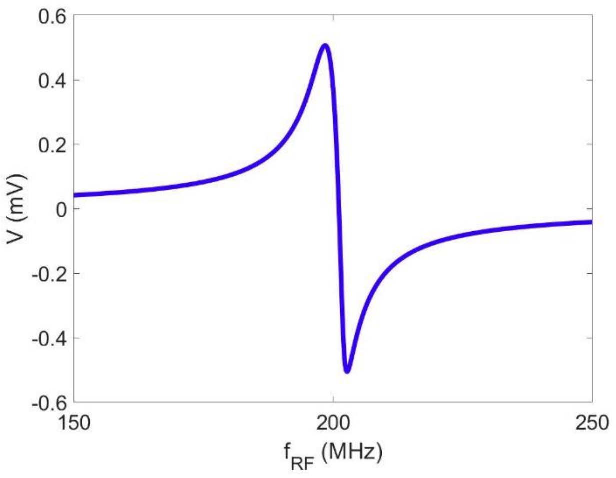
We can cascade junctions as neurons and synapses



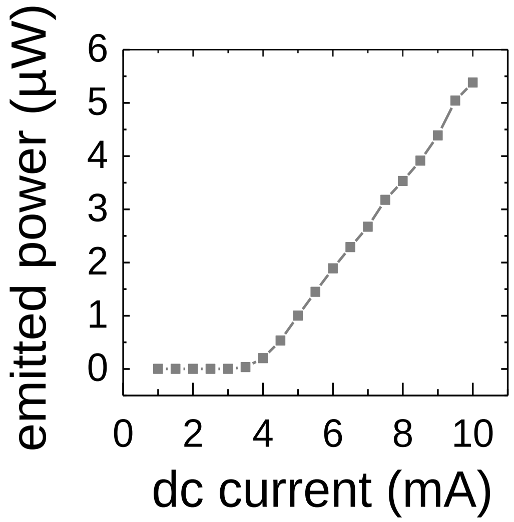
Spin-torque nano-oscillator



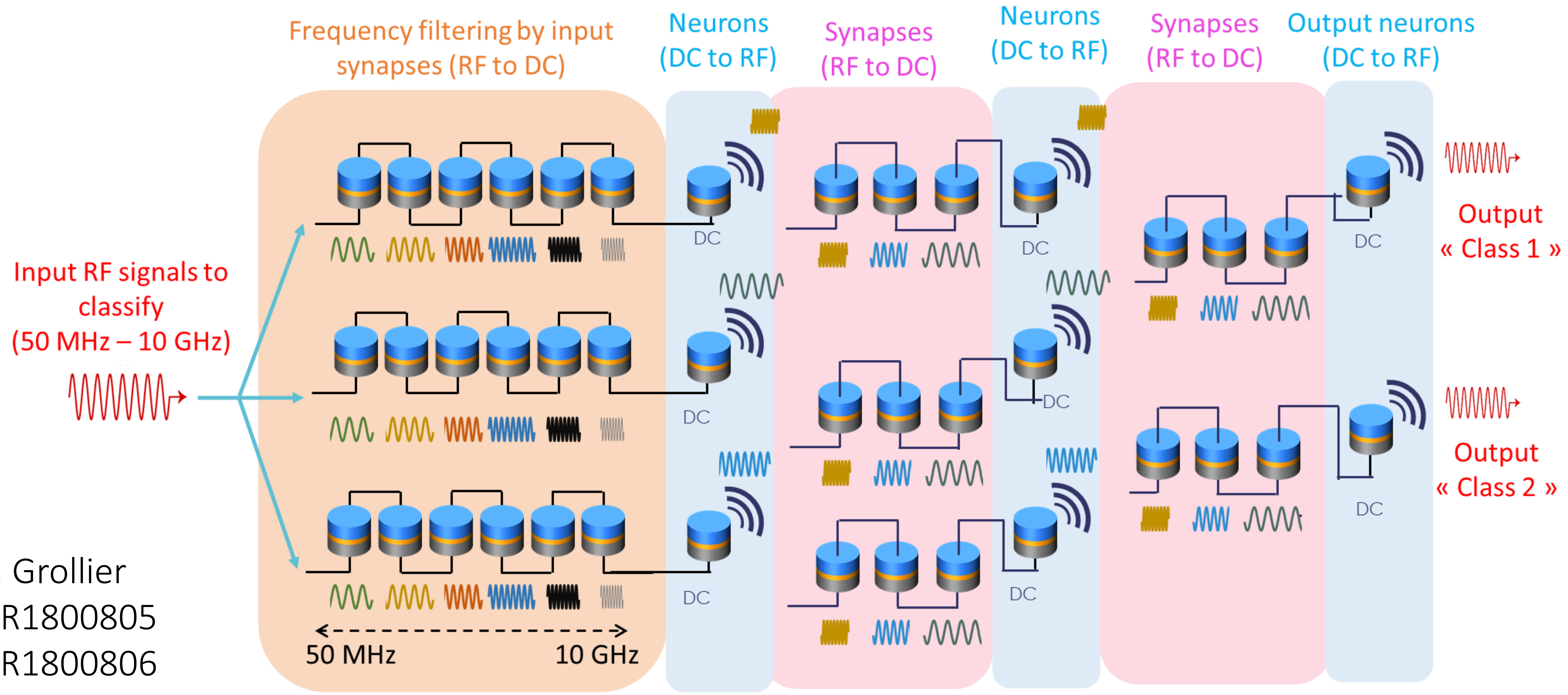
Spin diode



Spin-torque nano-oscillator



RF deep networks can be built with ultra high density



J. Grollier
FR1800805
FR1800806

In progress: simulations and experiments

The challenges of neuromorphic computing today are:

- To fabricate optimized synapses and neurons
- To build deep networks (successive MACs)
- To implement backpropagation

We have developed solutions to these challenges for spintronic neural networks :

… Can’t wait to be fully back in the lab ;)