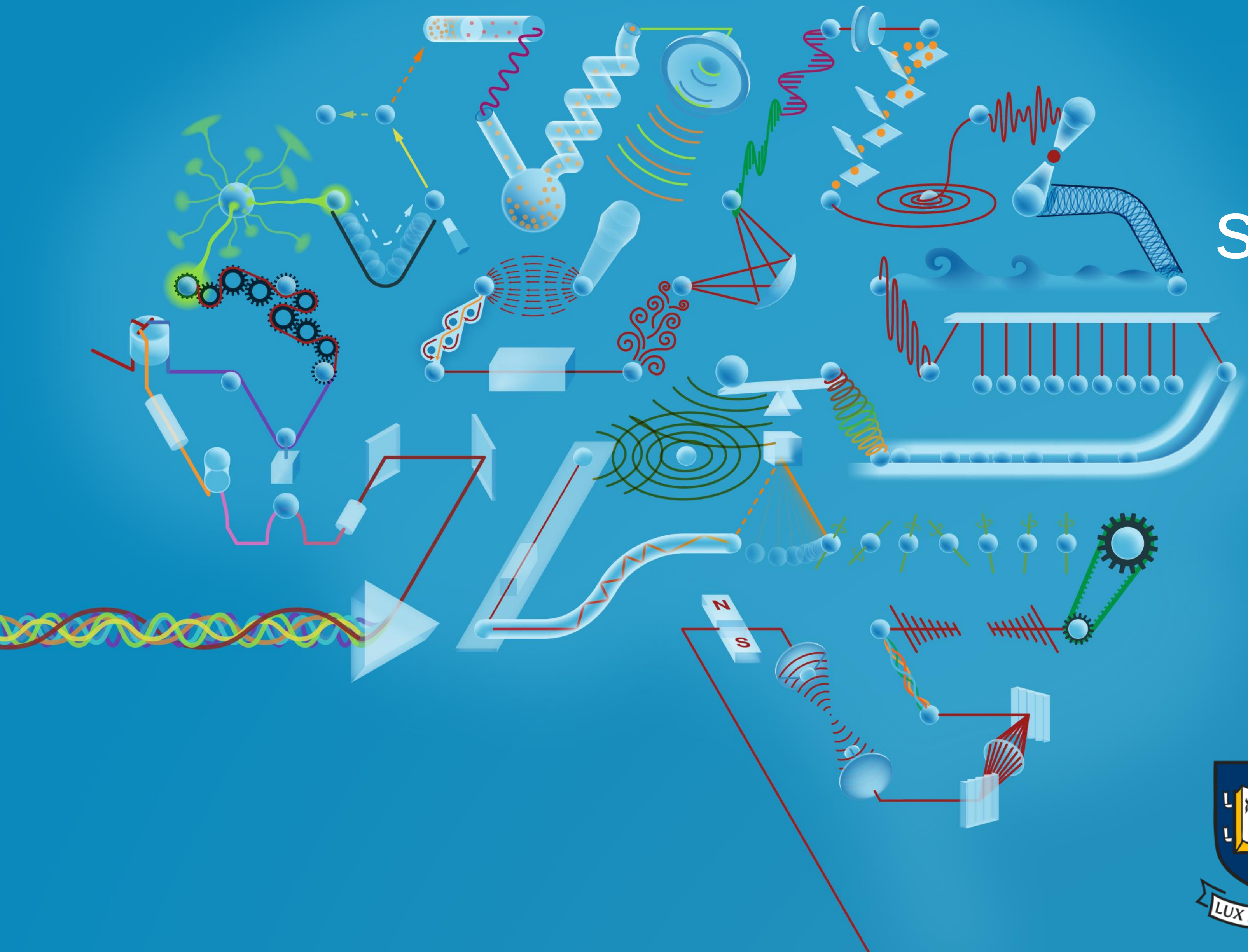
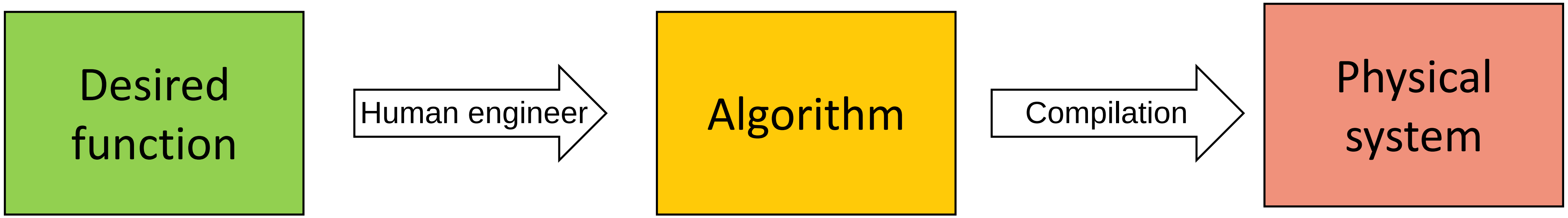


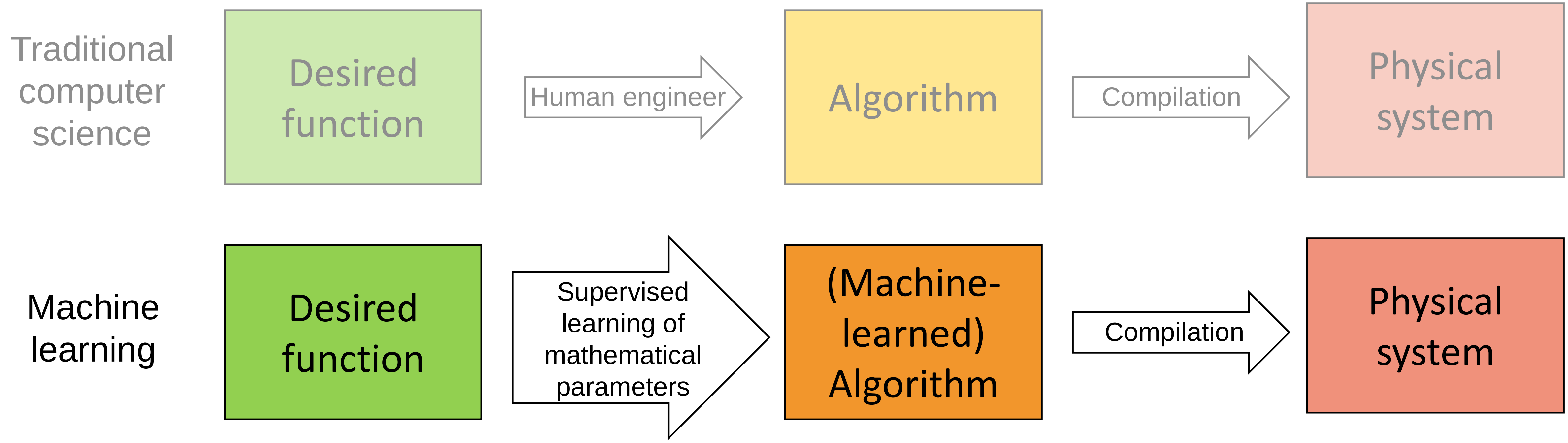
Deep physical neural networks: training physical systems like neural networks

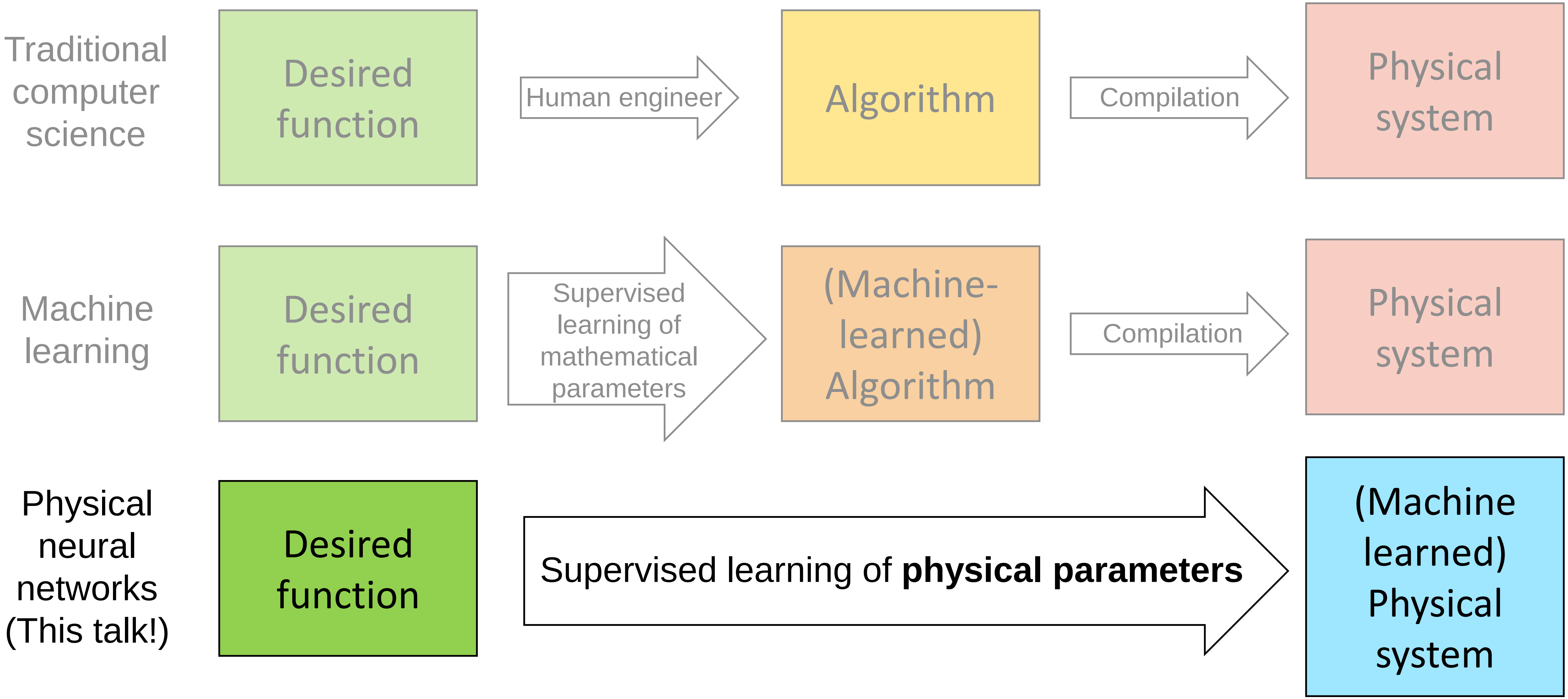
Computing with Physical Systems Aspen Conference (Jan. 2024)

Logan G. Wright

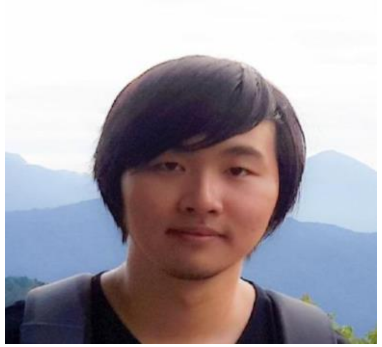
Traditional computer science




Traditional computer science
Desired function
Human engineer
Algorithm
Compilation
Physical system
Machine learning
Desired function
Supervised learning of mathematical parameters
(Machine-learned) Algorithm
Compilation
Physical system


Traditional computer science
Desired function
Human engineer
Algorithm
Compilation
Physical system
Machine learning
Desired function
Supervised learning of mathematical parameters
(Machine-learned) Algorithm
Compilation
Physical system
Physical neural networks (This talk!)
Desired function
Supervised learning of **physical parameters**
(Machine learned) Physical system

Acknowledgments



Tatsuhiro Onodera (co-lead)

Martin Stein

Tianyu Wang

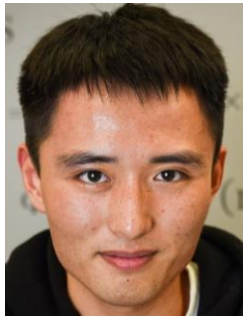
Darren Schachter

Zoey Hu

Peter McMahon (PI)

Maxwell Anderson

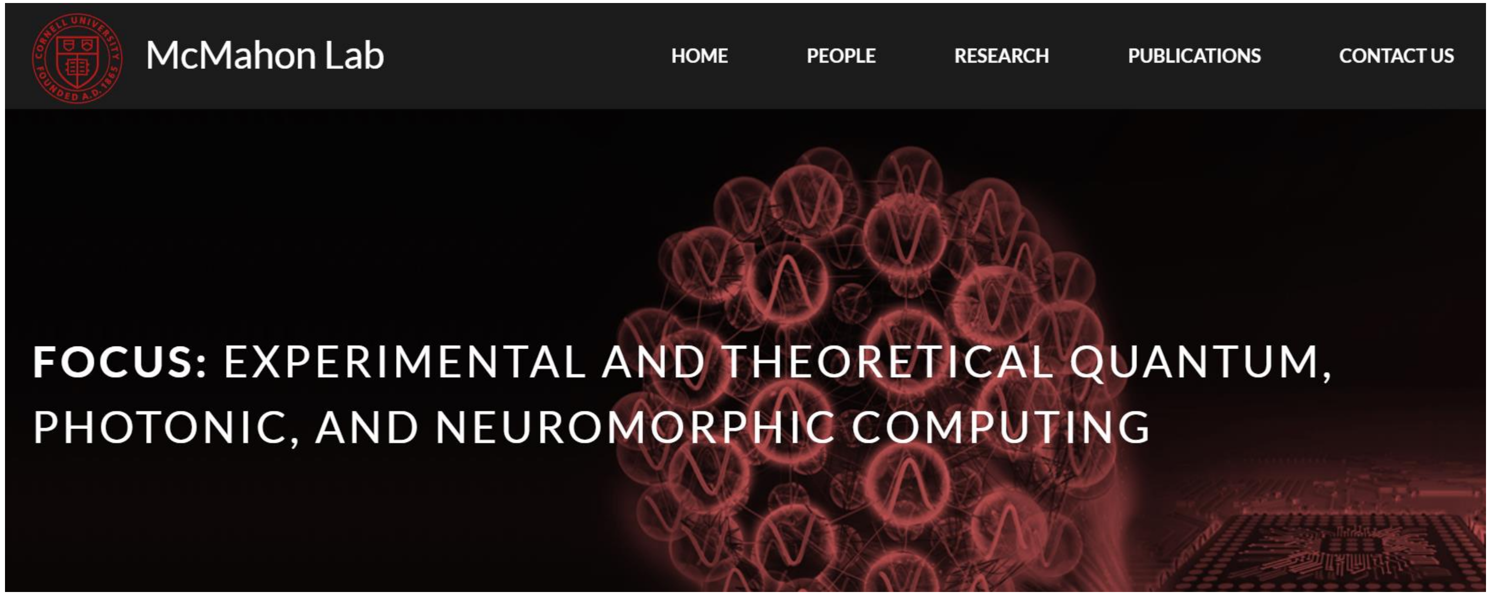
Mandar Sohoni

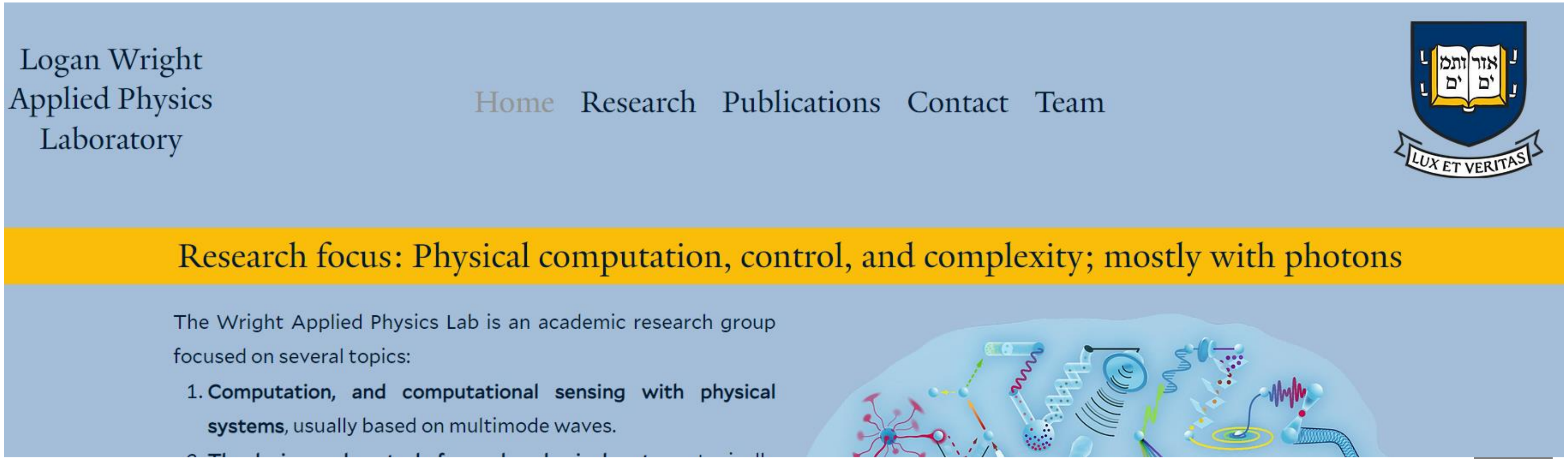


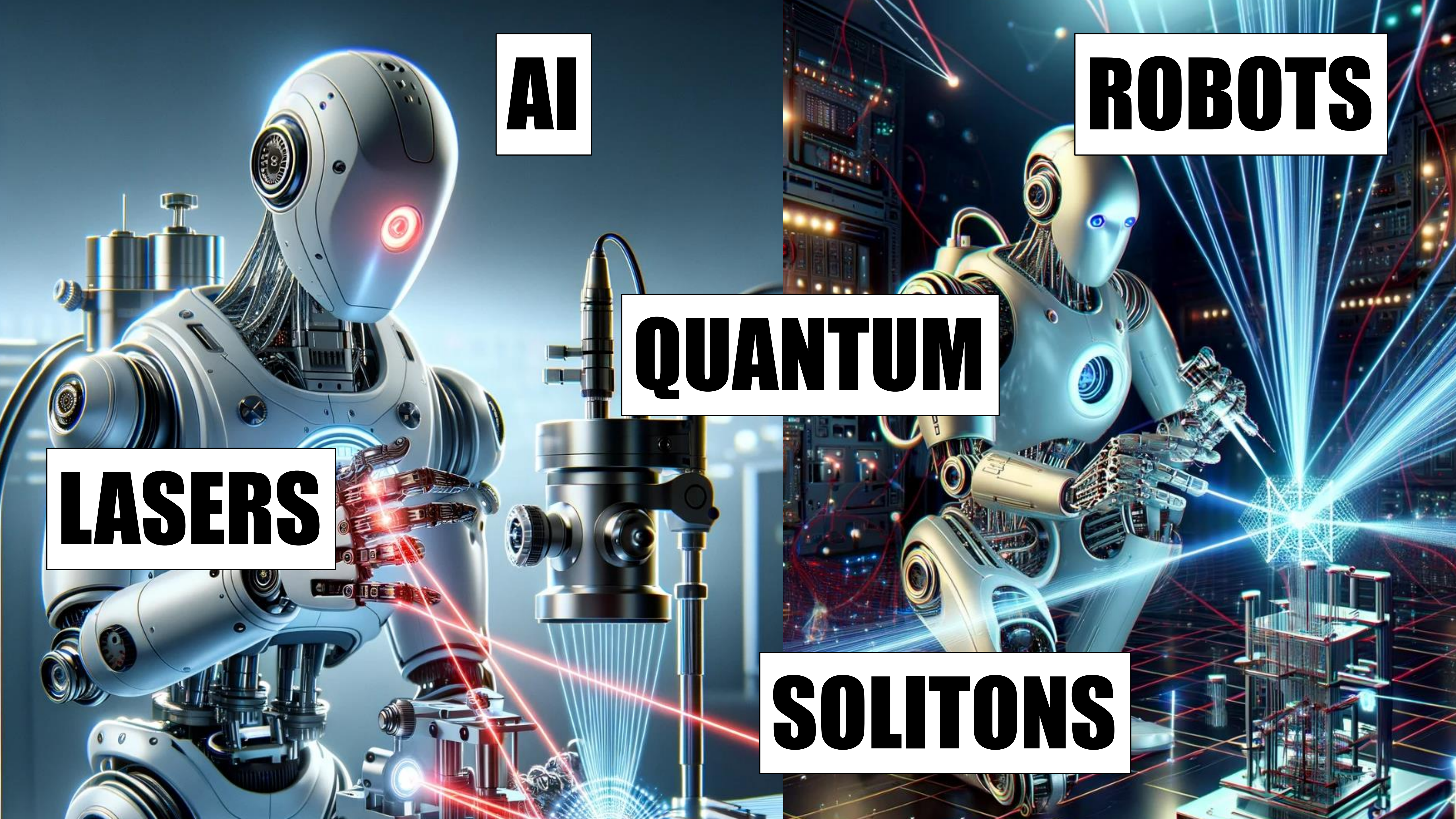
Shiyuan Ma

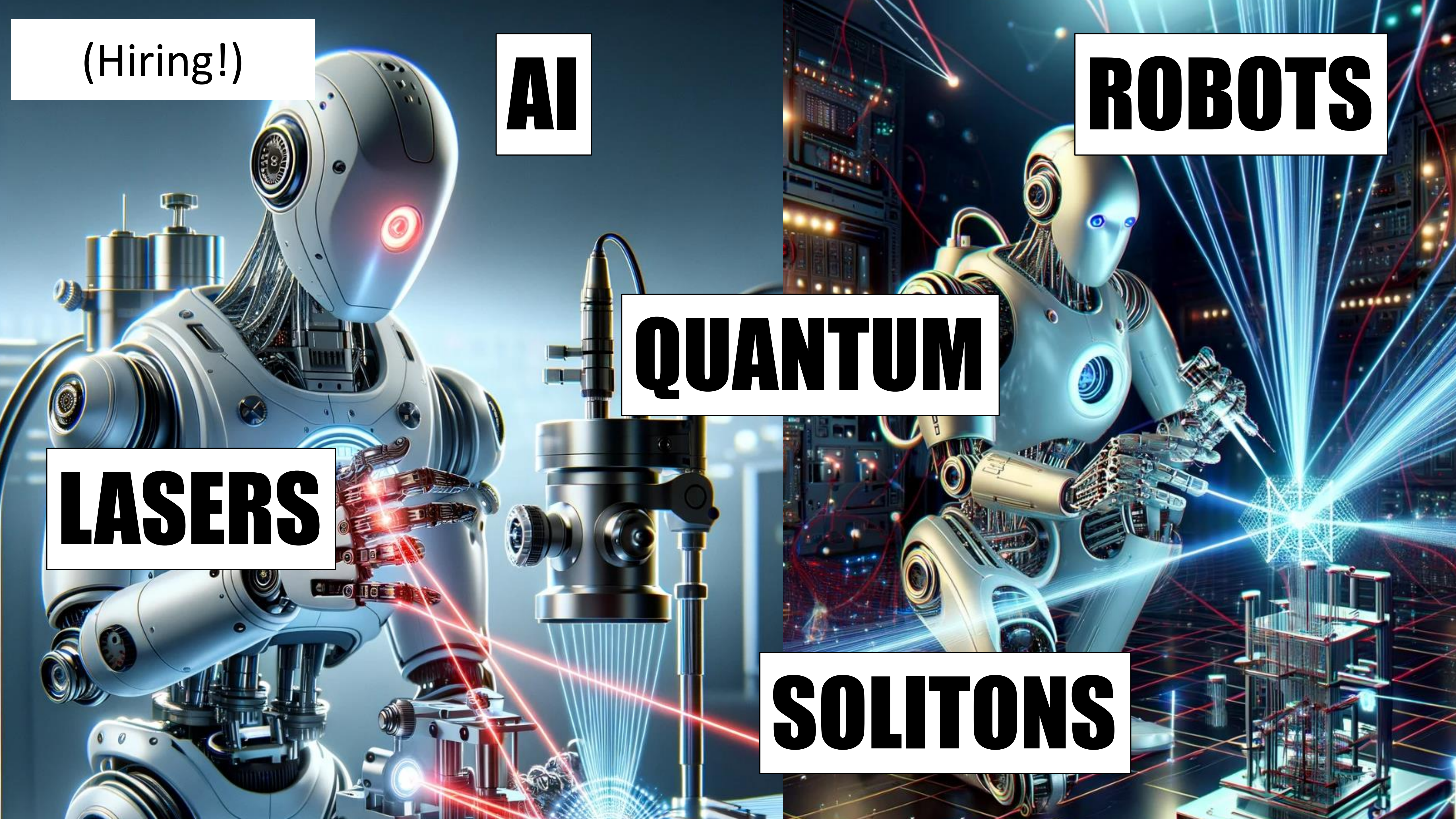
Jeremie Laydevant

New lab at Cornell → New lab at Yale






AI
ROBOTS
QUANTUM
LASERS
SOLITONS


(Hiring!)
AI
ROBOTS
QUANTUM
LASERS
SOLITONS

(Mathematical) neural networks

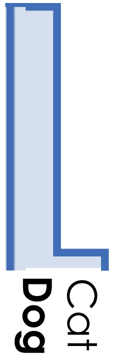
Deep learning: “just” high-dimensional curve-fitting*

$\vec{x}$

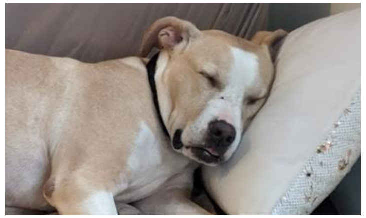
Input

$\vec{\theta}_1$ $\vec{\theta}_2$ $\vec{\theta}_3$

$\vec{y}$



Output

$\vec{y} = \mathrm{f}($  $, \vec{\theta}_1, \vec{\theta}_2, \vec{\theta}_3)$

*This intro blatantly copied from S. Dillavou

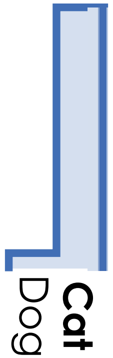
Deep learning: “just” high-dimensional curve-fitting*

$\vec{x}$

Input

$\vec{\theta}_1$ $\vec{\theta}_2$ $\vec{\theta}_3$

$\vec{y}$

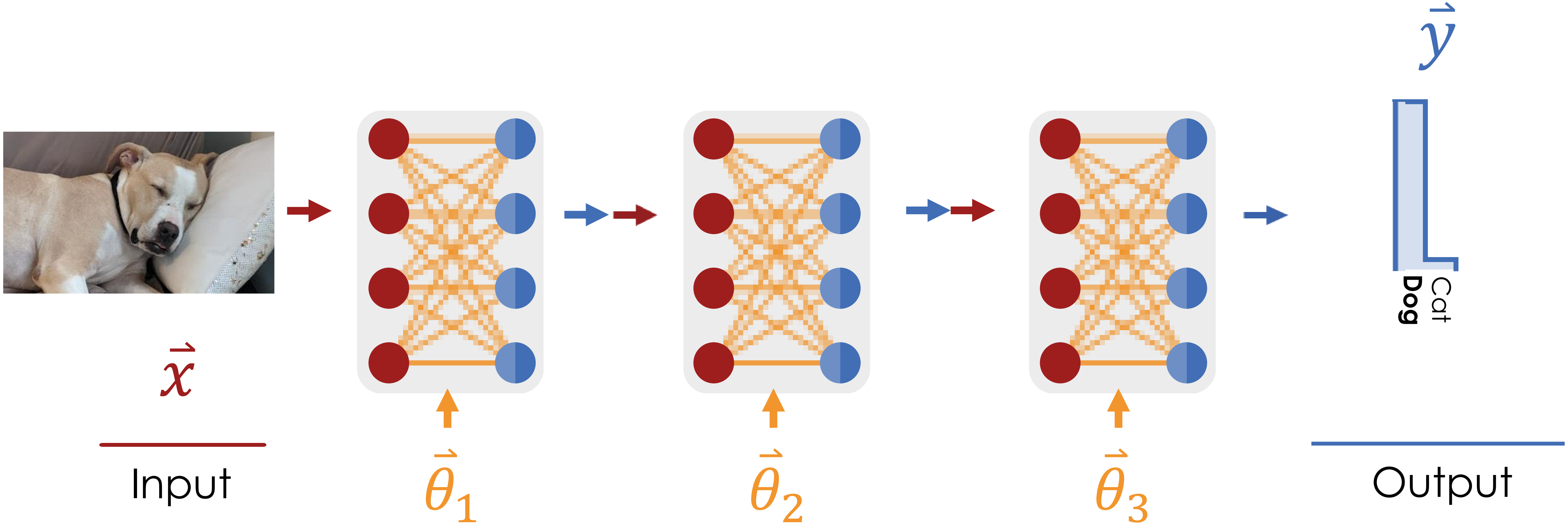


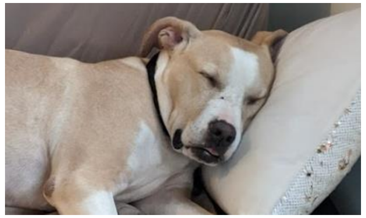
Output

$\vec{y} = \mathrm{f}($  $, \vec{\theta}_1, \vec{\theta}_2, \vec{\theta}_3)$

*This intro blatantly copied from S. Dillavou

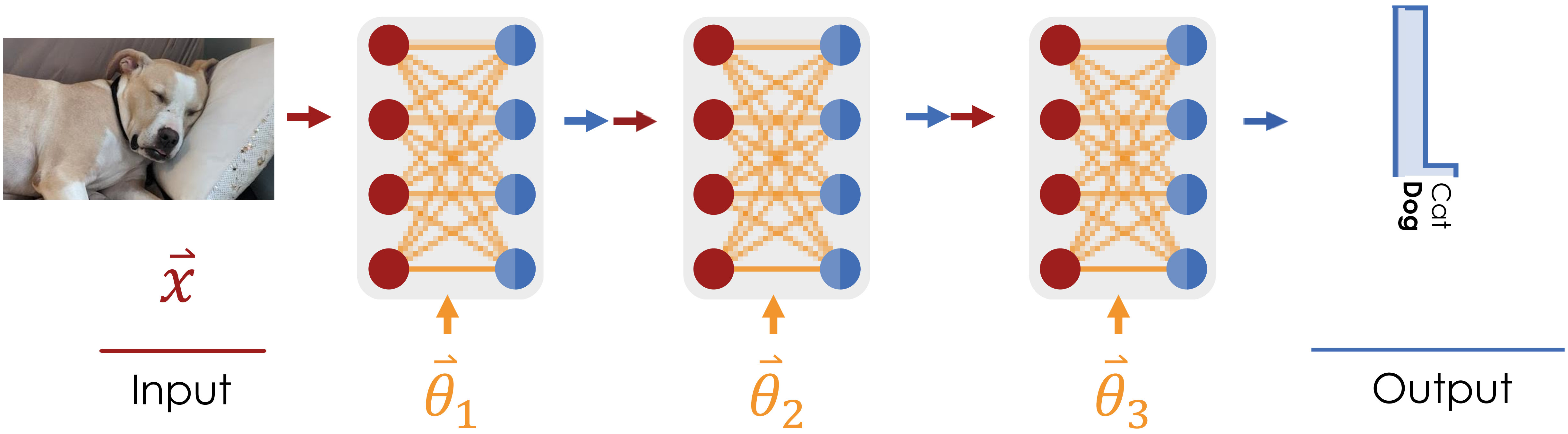
Deep learning: the 'deep' means multi-layer neural networks

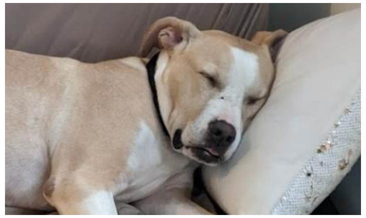
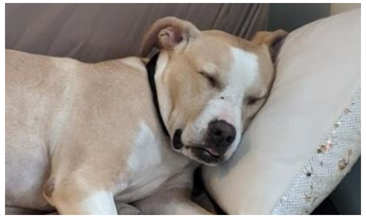


$\vec{y} = \mathrm{f}($  $, \vec{\theta}_1, \vec{\theta}_2, \vec{\theta}_3)$

Deep learning: the ‘deep’ means multi-layer neural networks

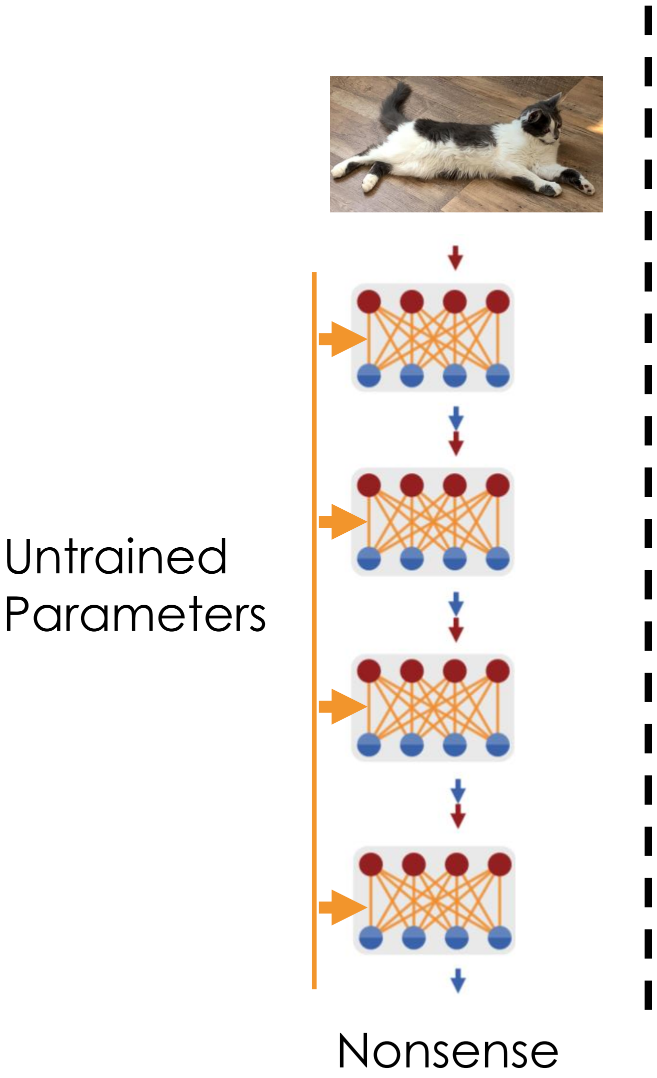
Deep neural networks learn **hierarchical** computations



$\vec{y} = \mathrm{f}($  $, \vec{\theta}_1, \vec{\theta}_2, \vec{\theta}_3) = \mathrm{f}(\mathrm{f}(\mathrm{f}($  $, \vec{\theta}_1), \vec{\theta}_2), \vec{\theta}_3)$

Deep neural networks: training versus inference

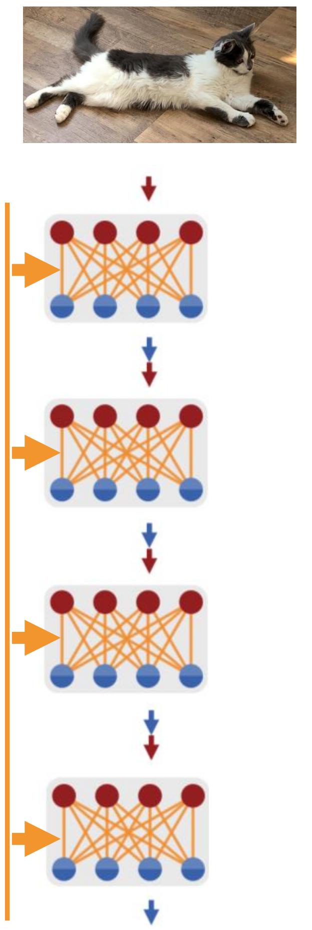
Untrained



Deep neural networks: training versus inference

Untrained

Untrained Parameters



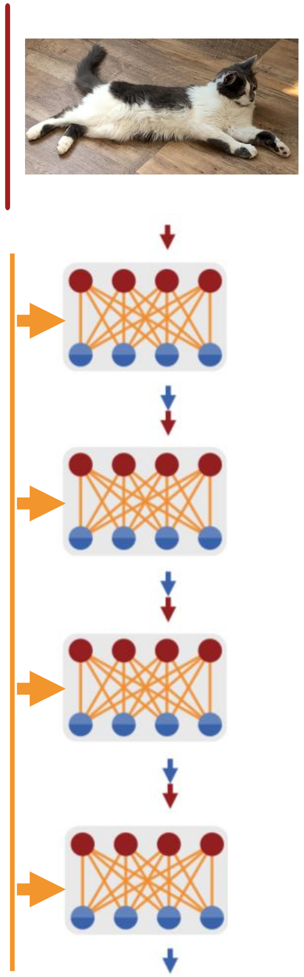
Nonsense

Training

Training input data

Parameters are **changed**

$$\vec{\theta} \rightarrow \vec{\theta} + d\vec{\theta}$$

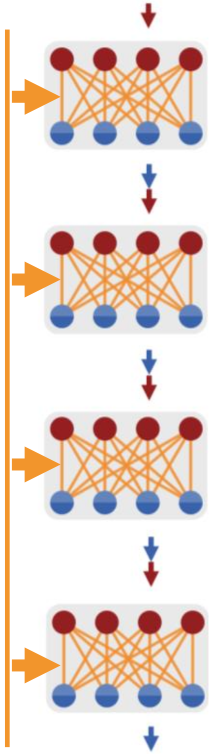


“Cat”

Deep neural networks: training versus inference

Untrained

Untrained Parameters



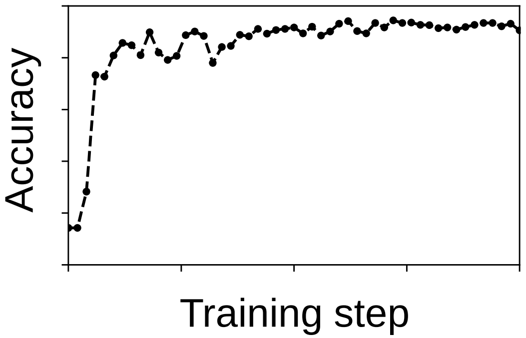
Nonsense

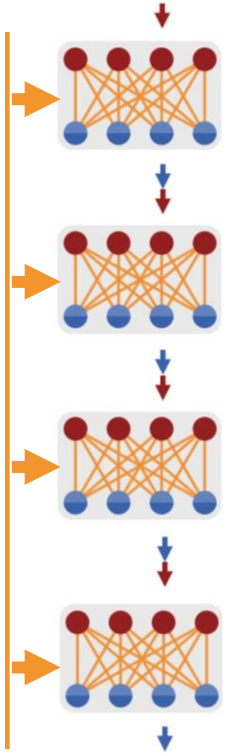
Training

Training input data

Parameters are **changed**

$$\vec{\theta} \rightarrow \vec{\theta} + d\vec{\theta}$$

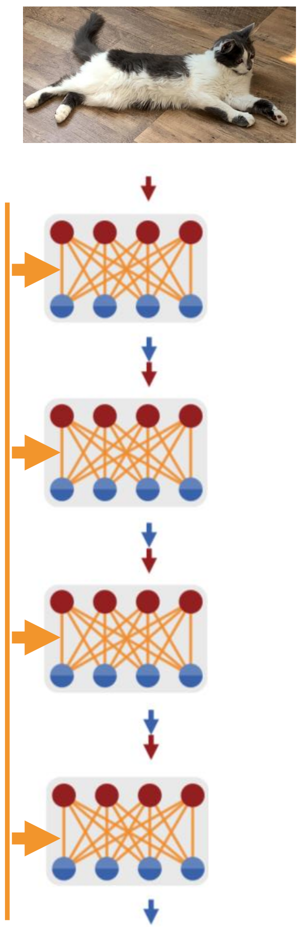




“Cat”

Deep neural networks: training versus inference

Untrained



Untrained Parameters

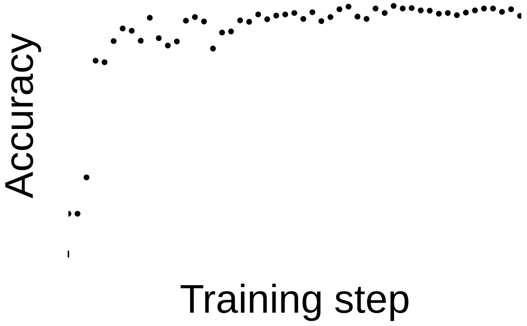
Nonsense

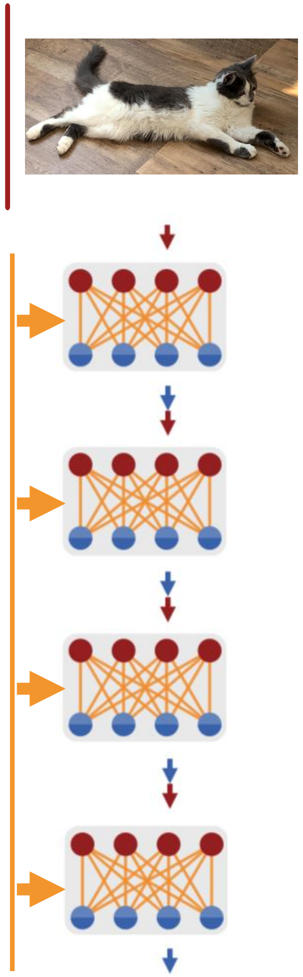
Training

Training input data

Parameters are **changed**

$$\vec{\theta} \rightarrow \vec{\theta} + d\vec{\theta}$$



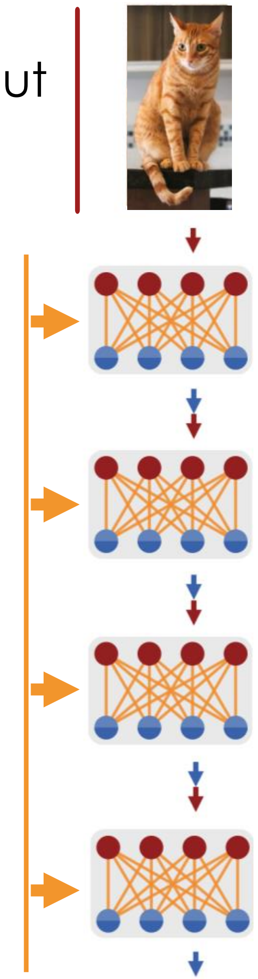


"Cat"

Inference

Unseen new input data

Parameters are **fixed**

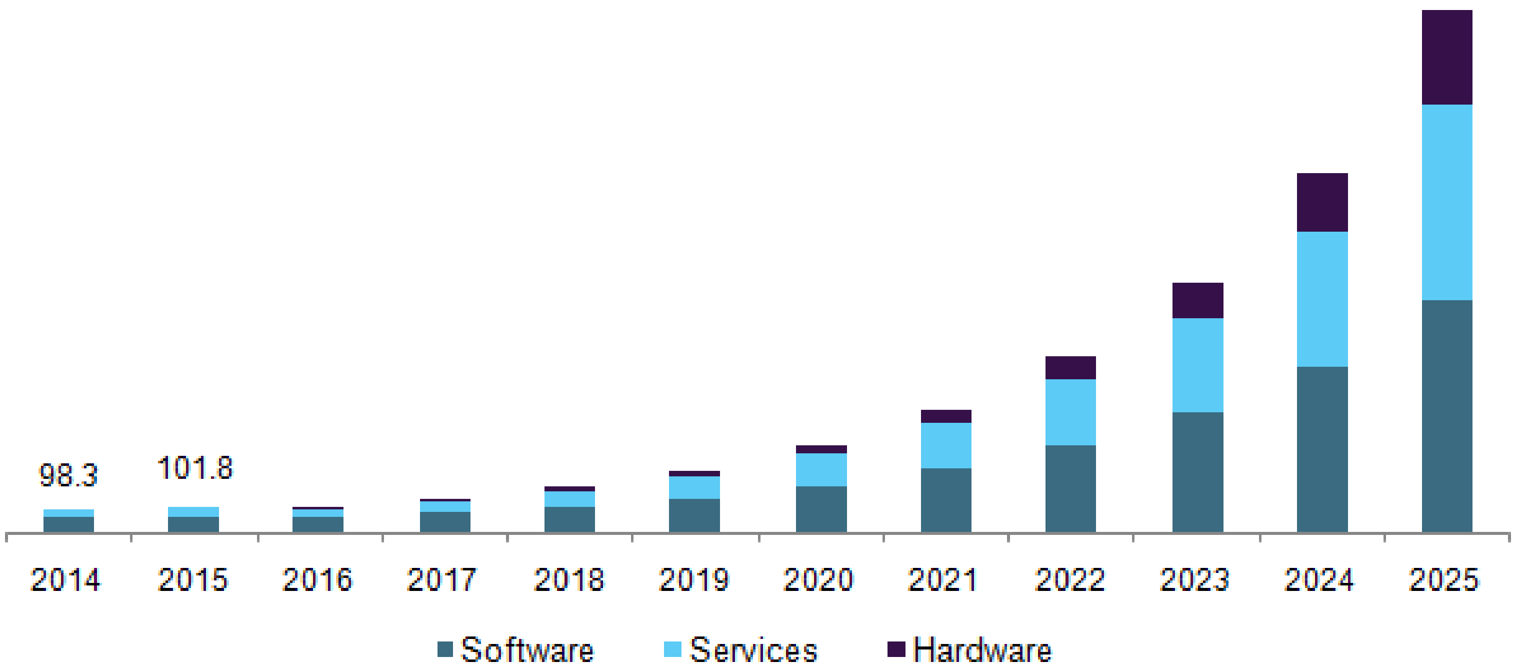


"Cat"

Deep learning is growing rapidly

Exponential growth of:

Market size



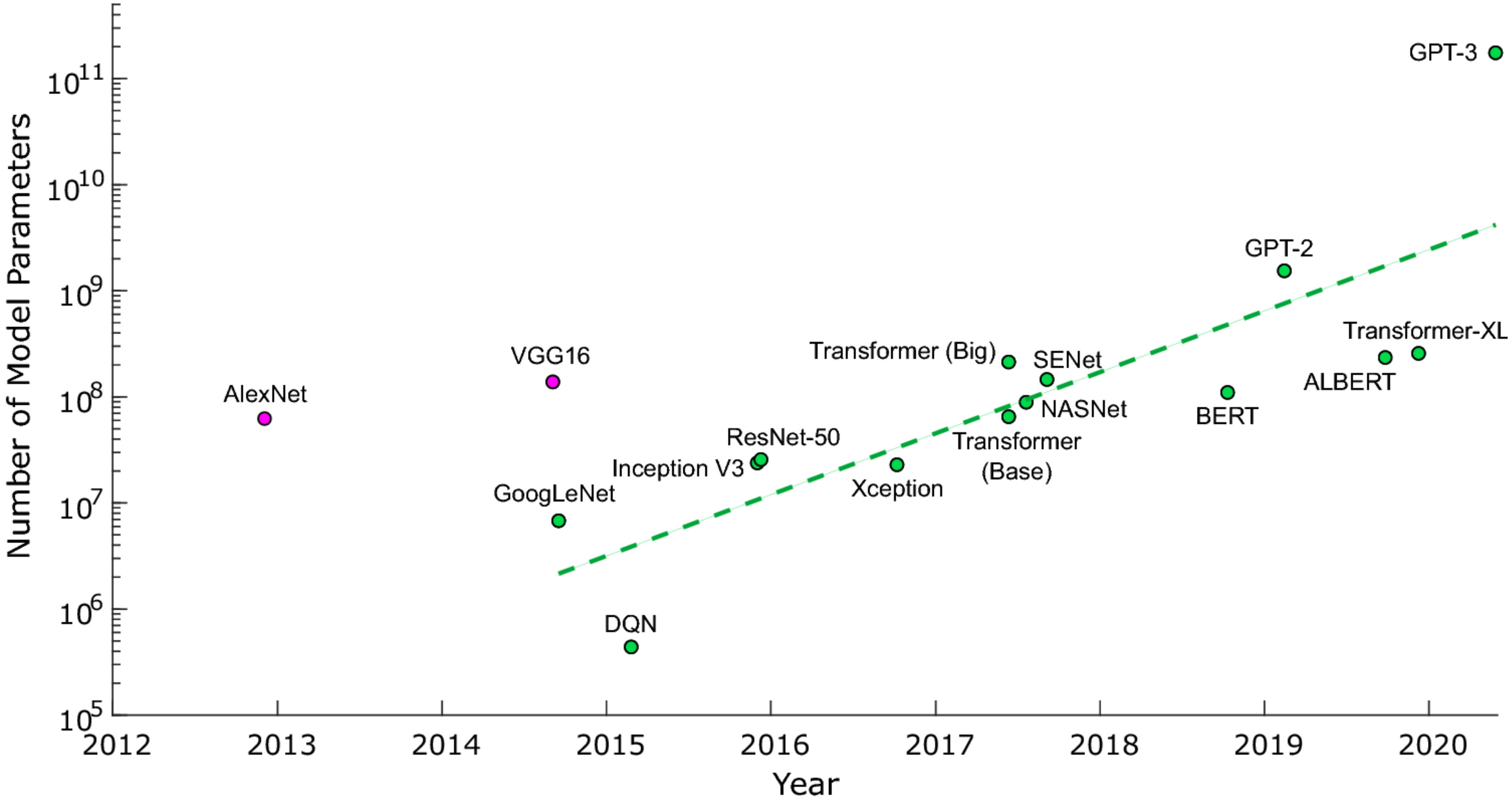
https://www.grandviewresearch.com/industry-analysis/deep-learning-market

Deep learning is growing rapidly

Exponential growth of:

Market size

Parameters



Bernstein, L., et al. "Freely scalable and reconfigurable optical hardware for deep learning." *Scientific Reports* (2021)

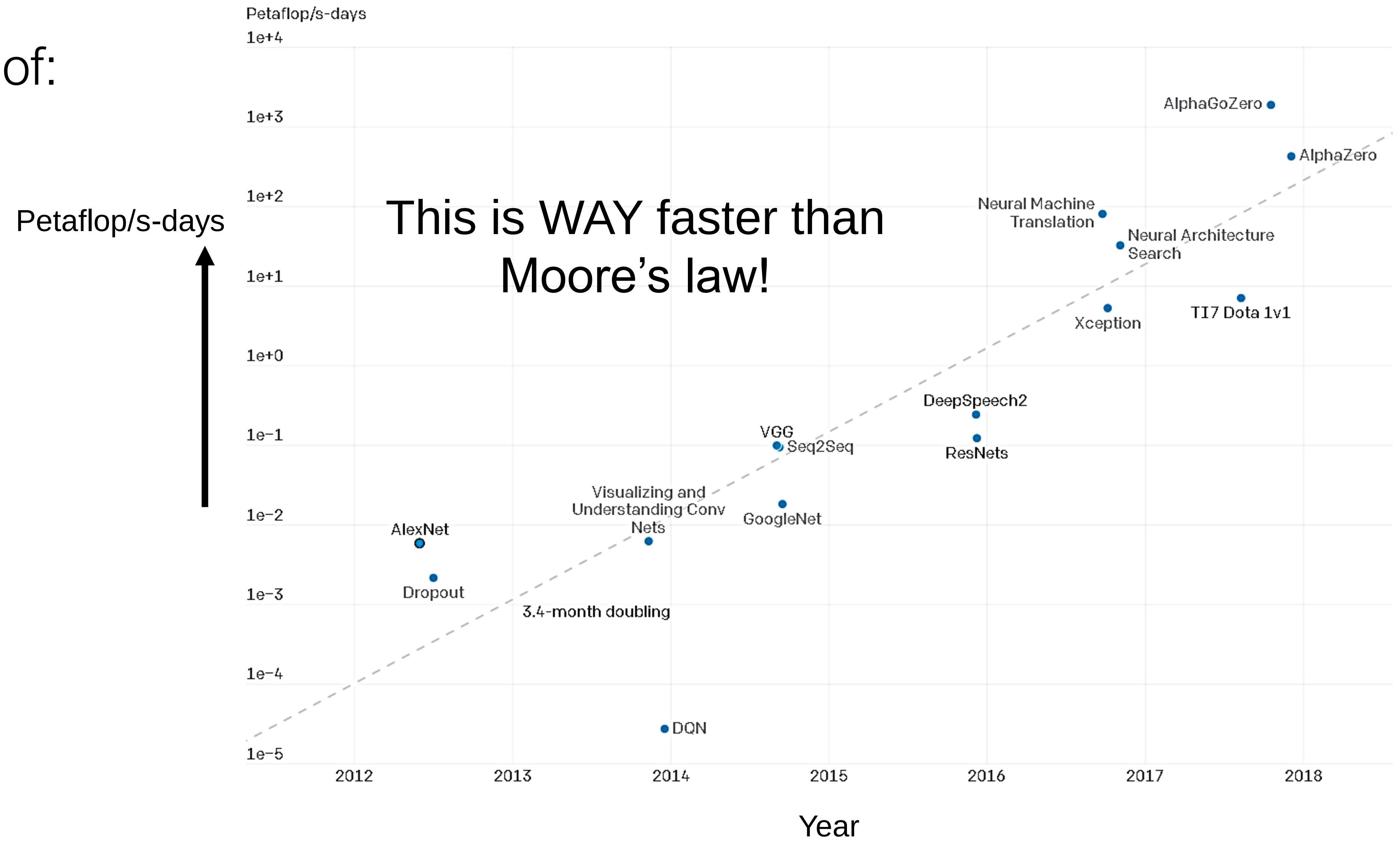
Deep learning is growing rapidly

Exponential growth of:

Market size

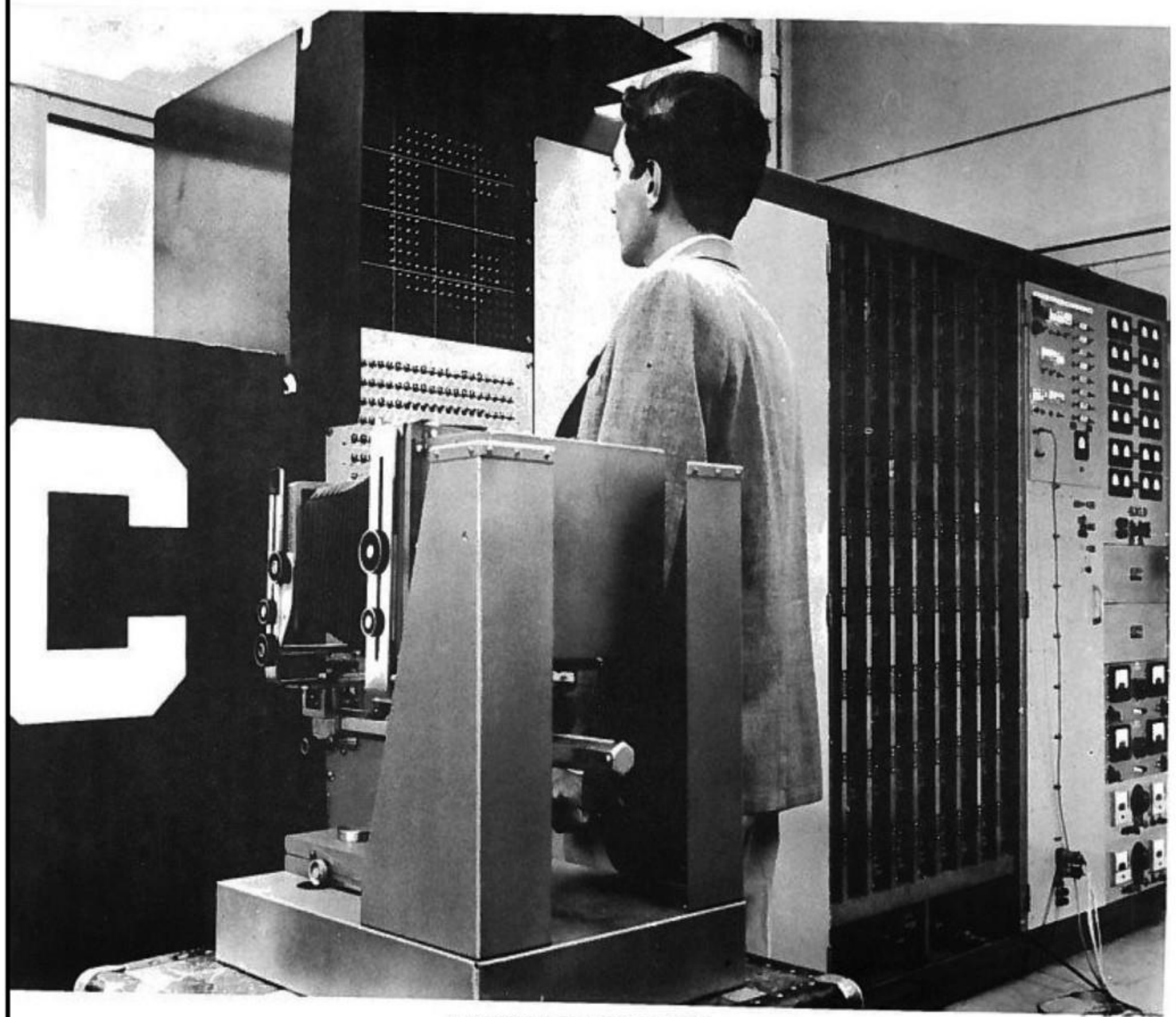
Parameters

Compute



https://openai.com/blog/ai-and-compute/

Good news: Neural networks are ideal for analog hardware



THE MARK I PERCEPTRON

NEW NAVY DEVICE LEARNS BY DOING

Psychologist Shows Embryo of Computer Designed to Read and Grow Wiser

WASHINGTON, July 7 (UPI)—The Navy revealed the embryo of an electronic computer today that it expects will be able to walk, talk, see, write, reproduce itself and be conscious of its existence.

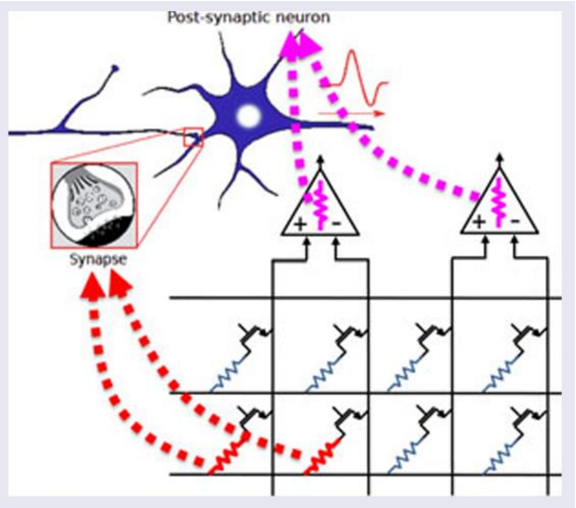
The embryo—the Weather Bureau's $2,000,000 "704" computer—learned to differentiate between right and left after fifty attempts in the Navy's demonstration for newsmen.

The service said it would use this principle to build the first of its Perceptron thinking machines that will be able to read and write. It is expected to be finished in about a year at a cost of $100,000.

Dr. Frank Rosenblatt, designer of the Perceptron, conducted the demonstration. He said the machine would be the first device to think as the human brain. As do human beings, Perceptron will make mistakes at first, but will grow wiser as it gains experience, he said.

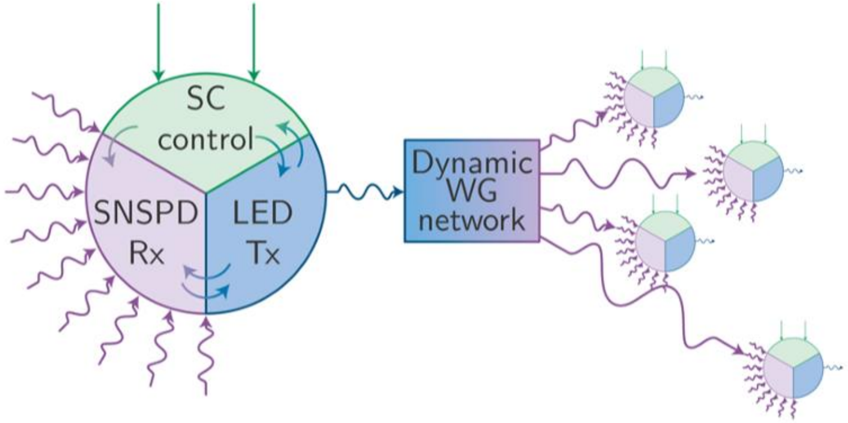
Dr. Rosenblatt, a research psychologist at the Cornell Aeronautical Laboratory, Buffalo, said Perceptrons might be fired to the planets as mechanical space explorers.

Neural network hardware uses analog physics to more energy-efficiently perform neural network calculations

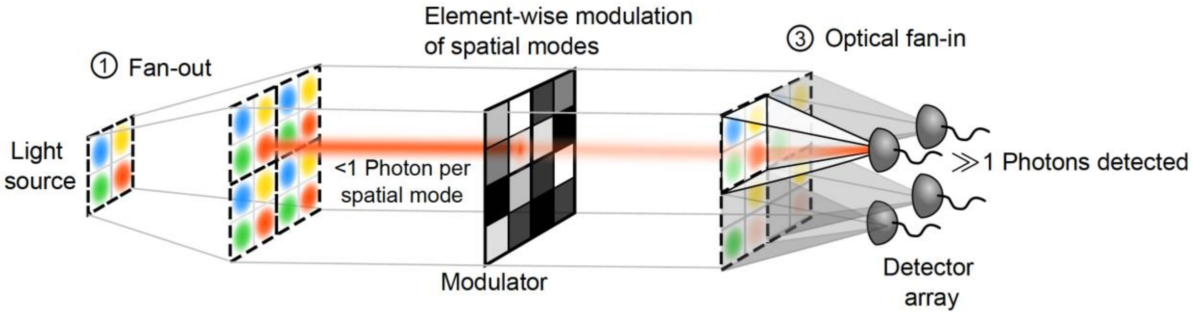


G.W. Burr et al. *Advances in Physics: X* (2017)

Lightmatter Mars chip from Hot Chips 32

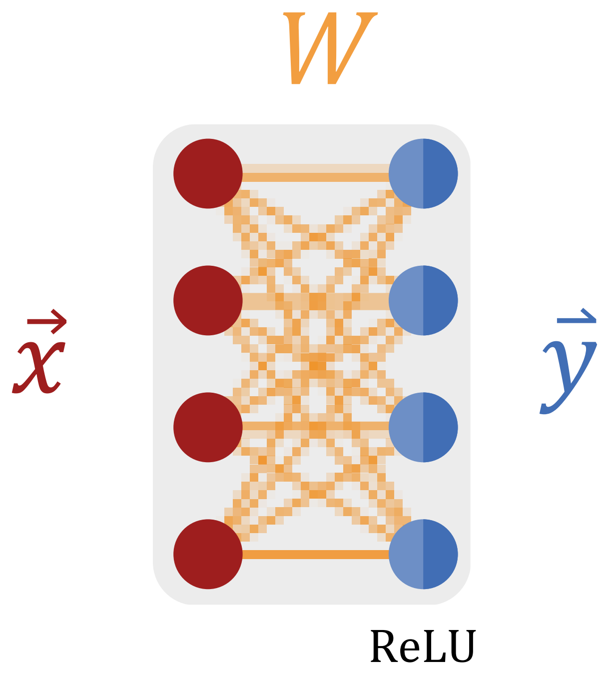


J.M Shainline et al. *Phys. Rev. Applied* (2017)



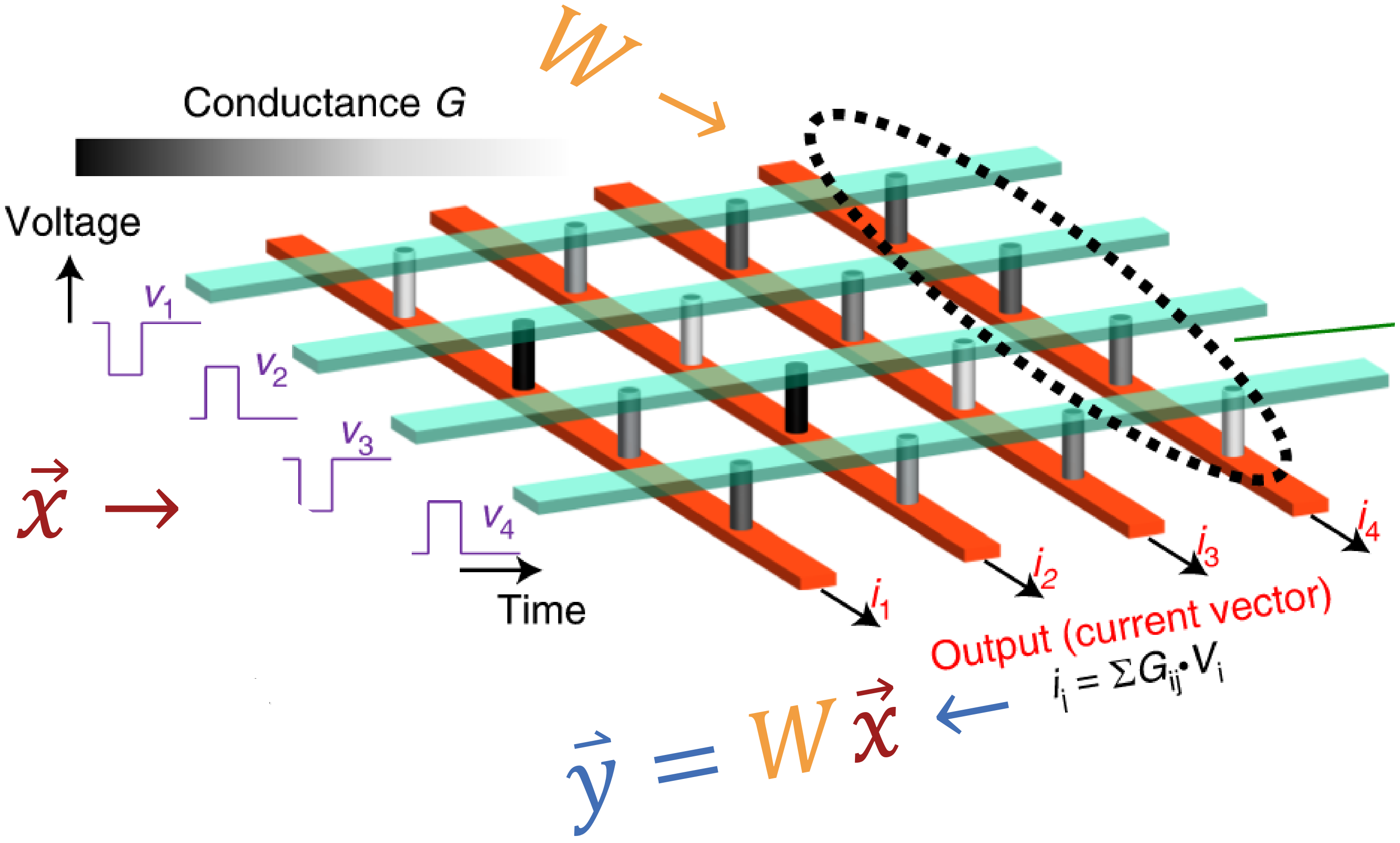
T. Wang, S.-Y. Ma, LGW et al. *Nature Comm* (2022)

These hardware usually rely on math-physics isomorphism



$$\vec{y} = \mathrm{ReLU}(W\vec{x})$$

where $W\vec{x}$ is a matrix-vector product



Cai et al., *Nature Electronics* (2020)

But achieving rigorous isomorphism involves trade-offs

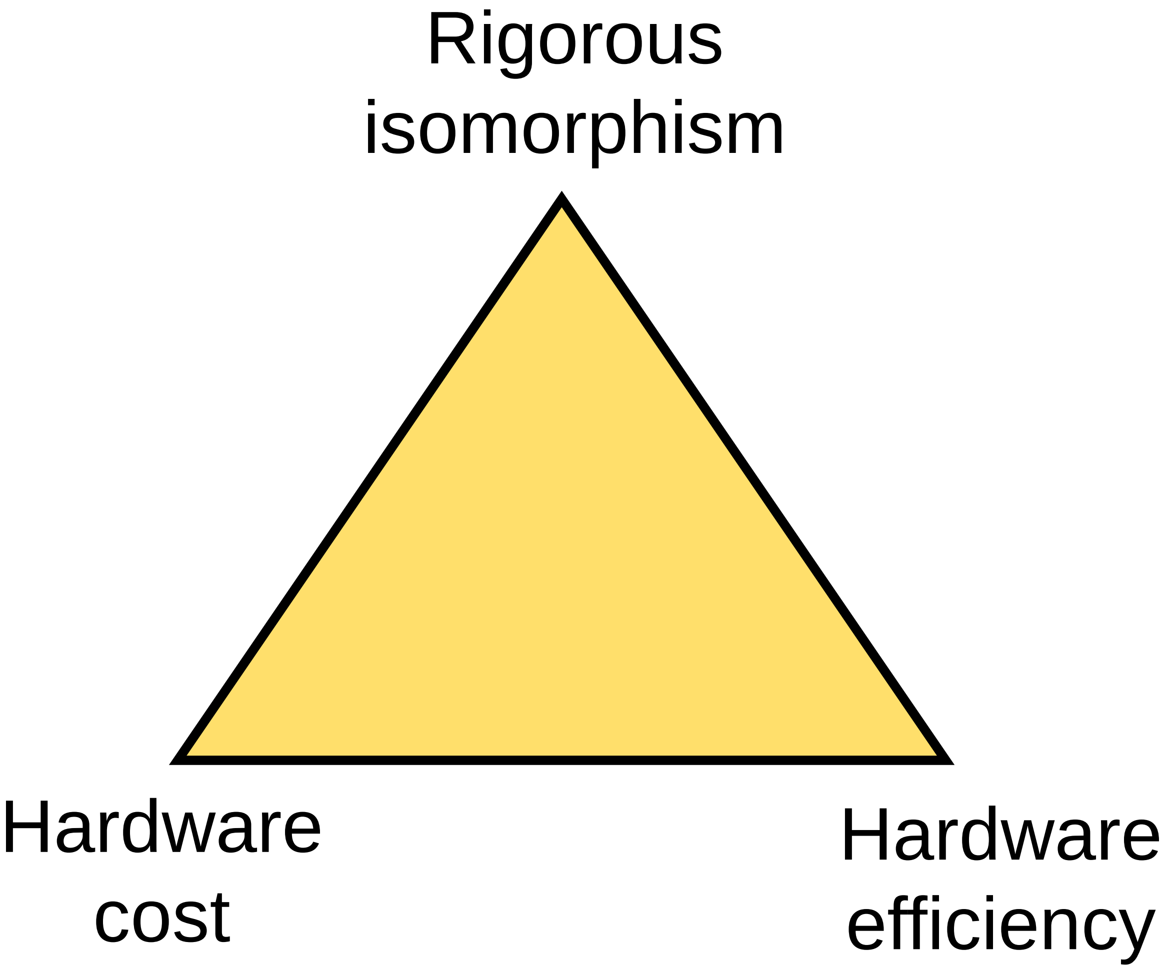
Calibration

Safe parameter regimes

Fabrication tolerances

Explicit error correction

→ High barrier for novel hardware



But achieving rigorous isomorphism involves trade-offs

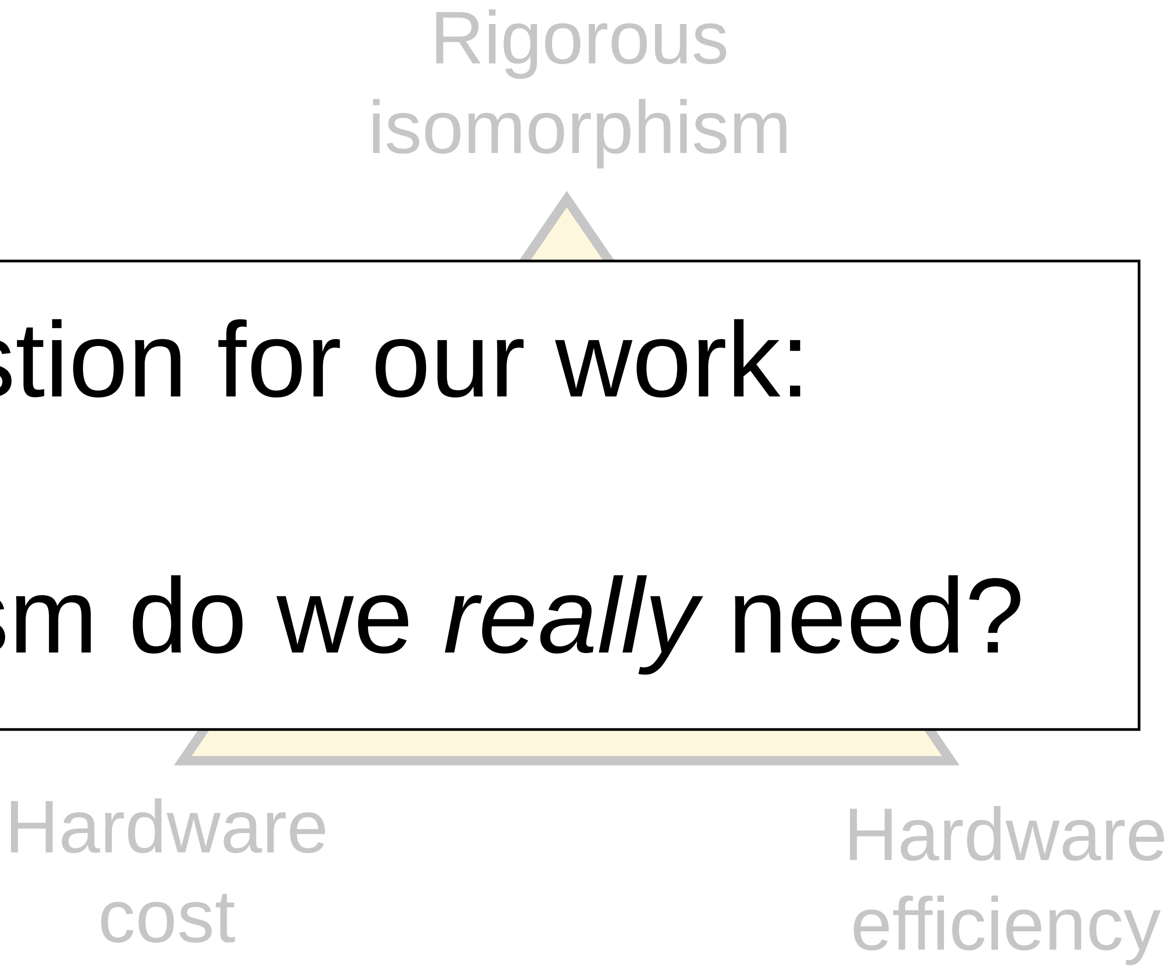
Calibration

Safe parameter regimes

Fa

Ex

→ High barrier for novel hardware

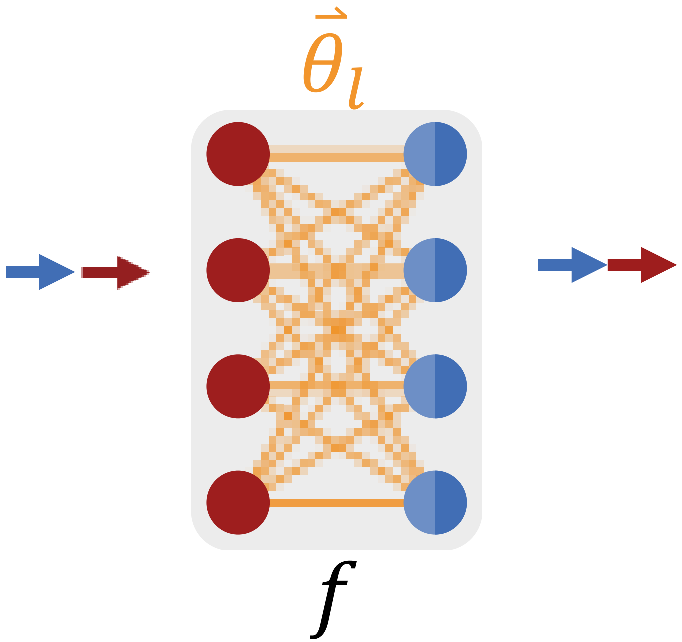


A motivating question for our work:

How much isomorphism do we *really* need?

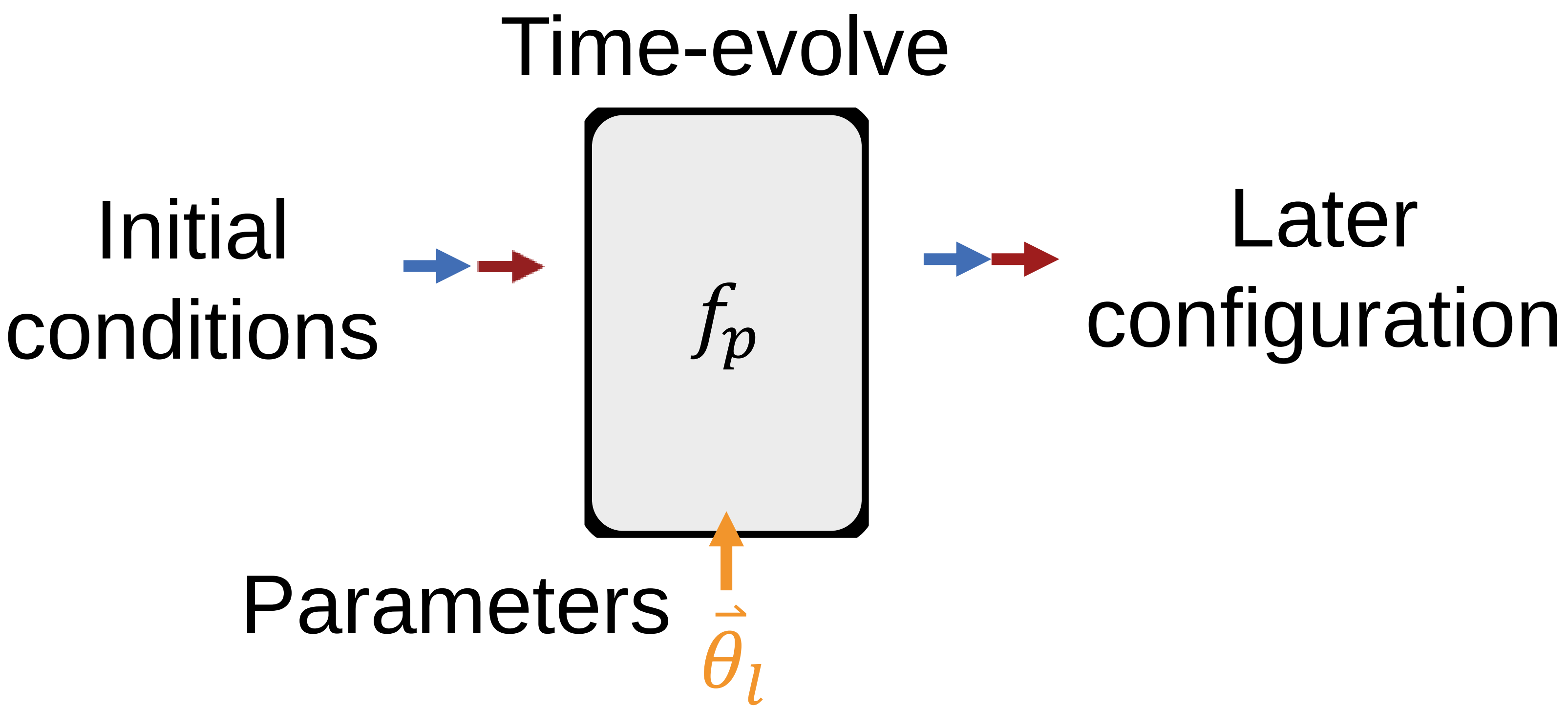
(Physical) neural networks

Deep neural network layers are controlled mathematical transformations



$$\vec{y}[l+1] = f(\vec{y}[l], \vec{\theta}_l)$$

Programmable physical systems give us controllable *physical* transformations



$$\vec{y}[l+1] = f_p(\vec{y}[l], \vec{\theta}_l)$$

Example: dynamics of coupled oscillators

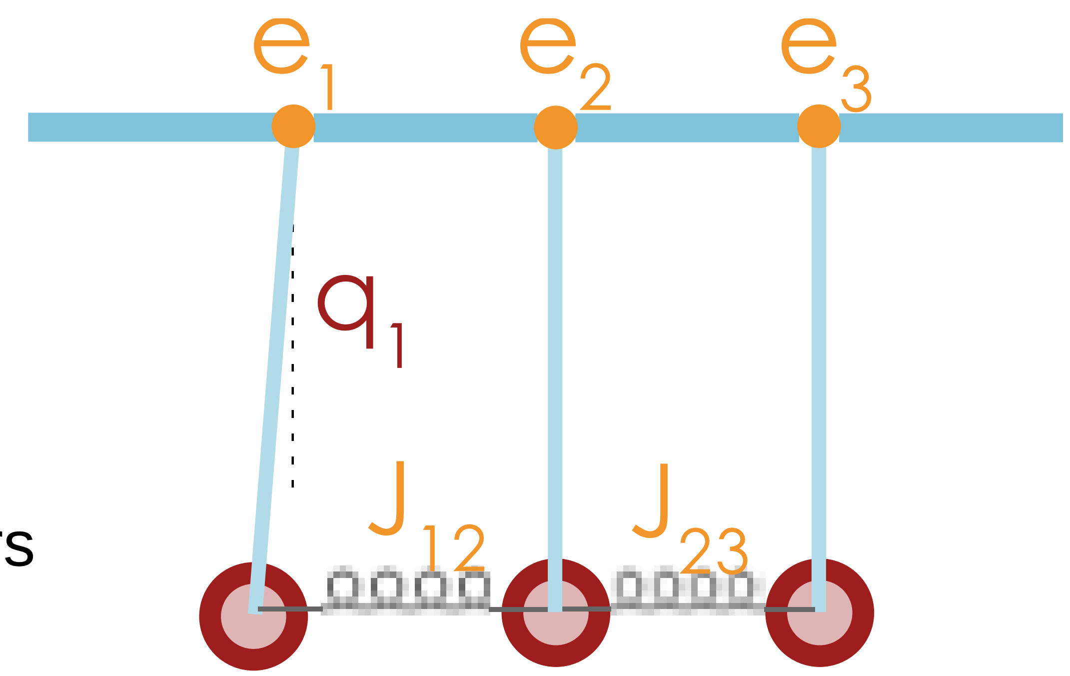
$$\frac{\mathrm{d}^2 q_i}{\mathrm{d}t^2} = -\sin q_i + \sum_{j=1}^{N} \boxed{J_{ij}} (\sin q_j - \sin q_i) + \boxed{e_i}$$

Input data = initial ($t = 0$) angles

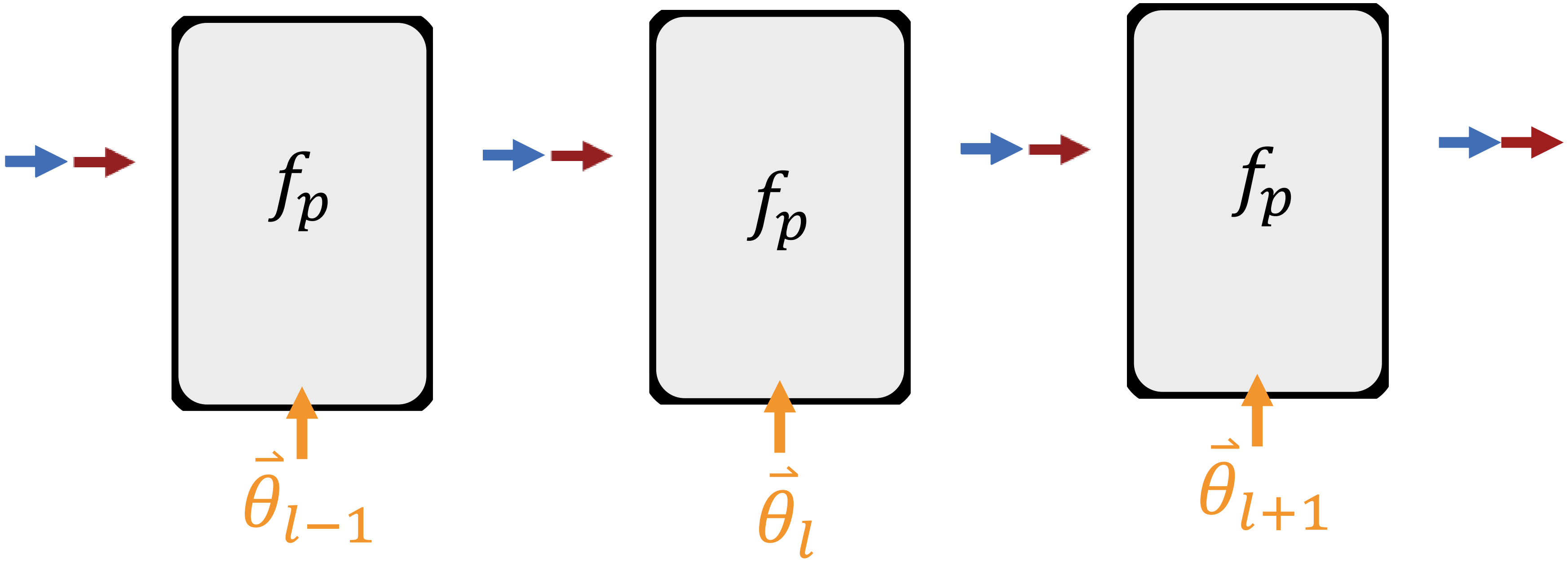
Parameters = coupling between oscillators (spring stiffness)

drive (fixed torque at joint)

Output = Later ($t = T$) angles of the oscillators



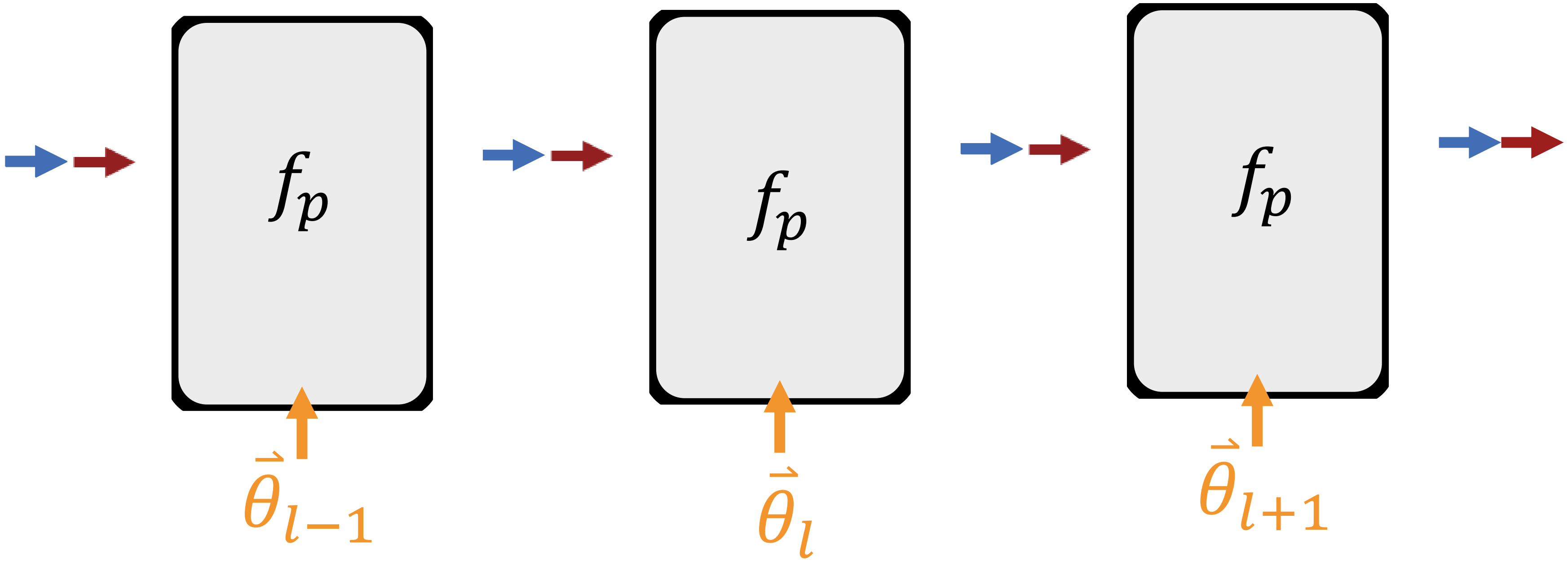
Deep physical neural networks



Physical neural network:

Network of controllable physical transformations

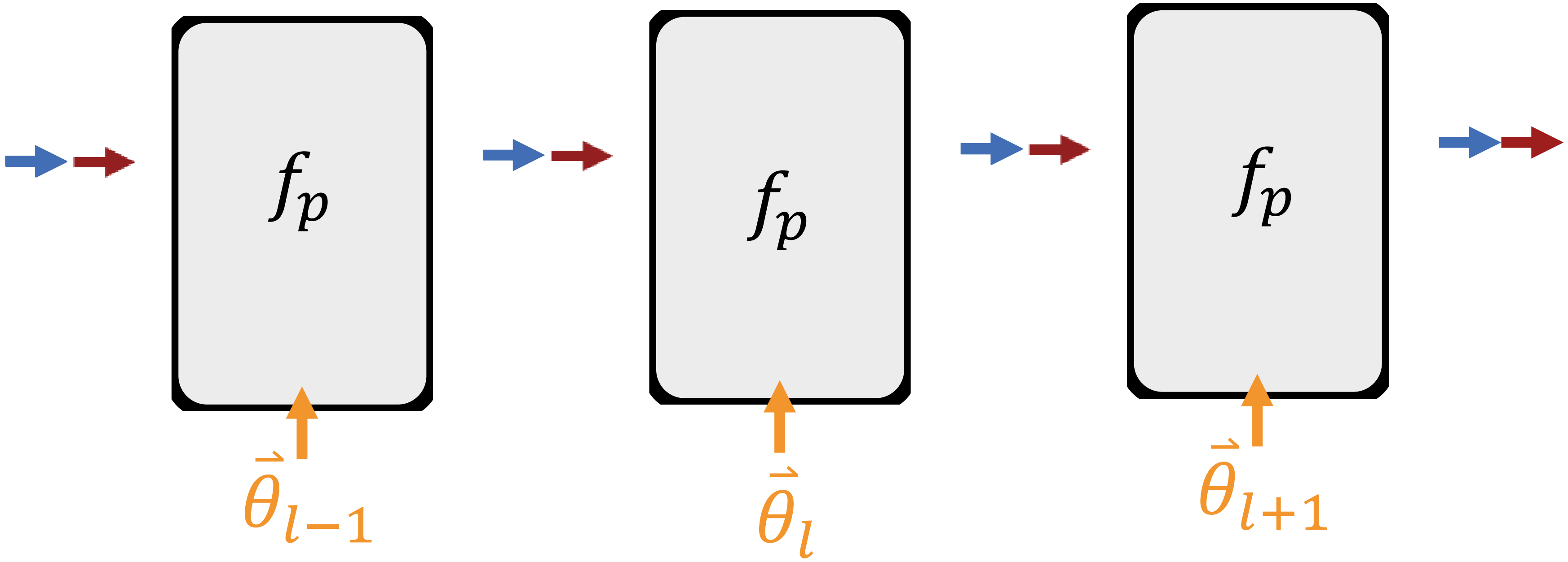
Deep physical neural networks



Physical neural network:

Network of controllable physical transformations, trained to perform **physical functions**

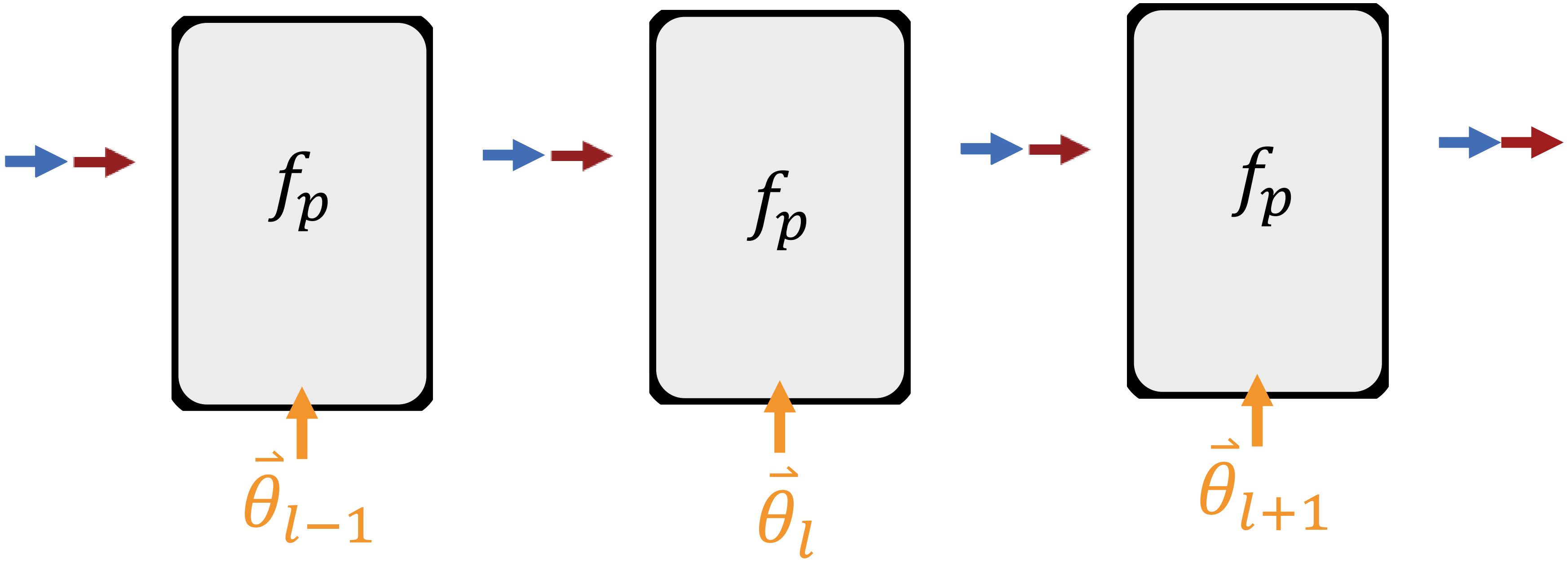
Deep physical neural networks



Physical neural network:

Network of controllable physical transformations, trained to perform **physical functions**, similar to how (artificial) neural networks are trained to perform *mathematical functions*

Deep physical neural networks



Physical neural network:

Network of controllable physical transformations, trained to perform **physical functions**, similar to how (artificial) neural networks are trained to perform *mathematical functions*

→This is a "flexible" analogy, ***not*** a strict 1:1 emulation of any specific artificial neural network's math!

Why on earth should this work?

Why on earth should this work?

DNNs model real-world physics well because they have similar structure

> Nonlinear, hierarchical, high-dimensional, noisy, analog, local, sparse,…

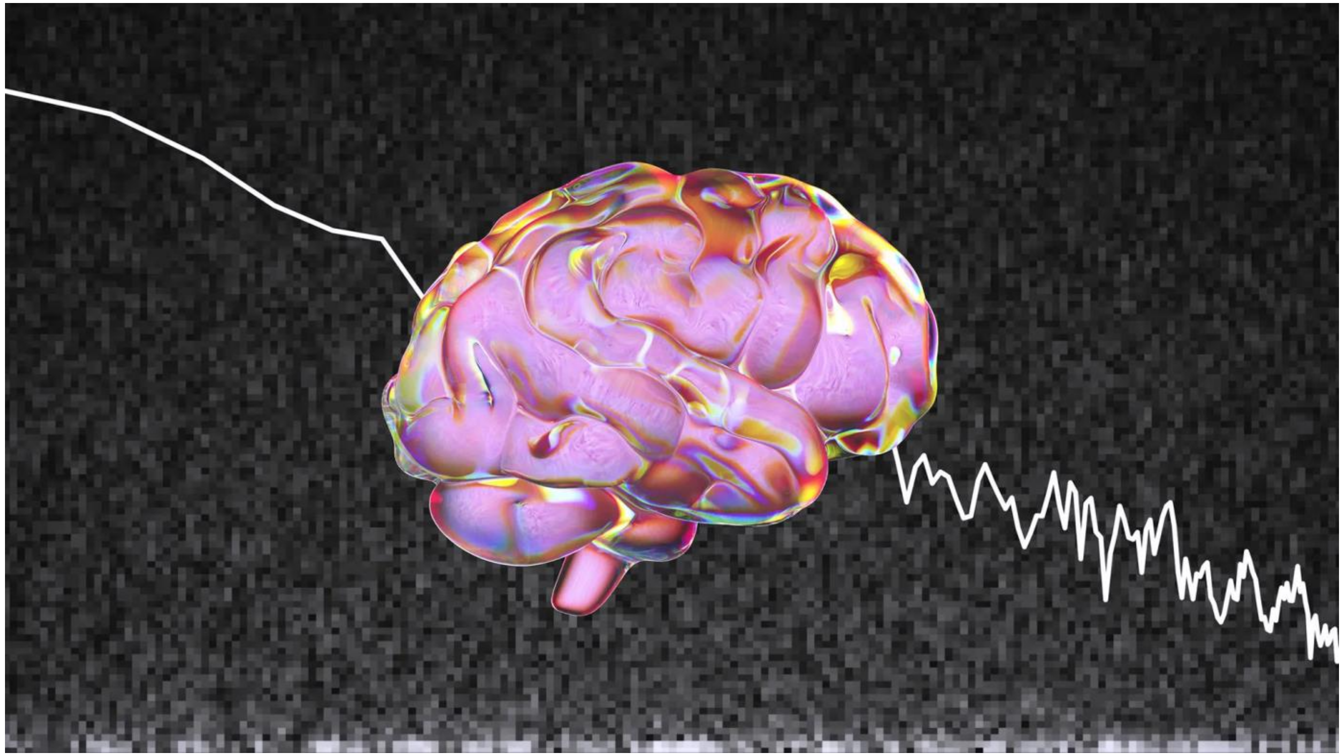


Illustration: Olena Shmahalo/Quanta Magazine; Thomas Donoghue

H.W. Lin et al., *Journal of Statistical Physics* (2017)

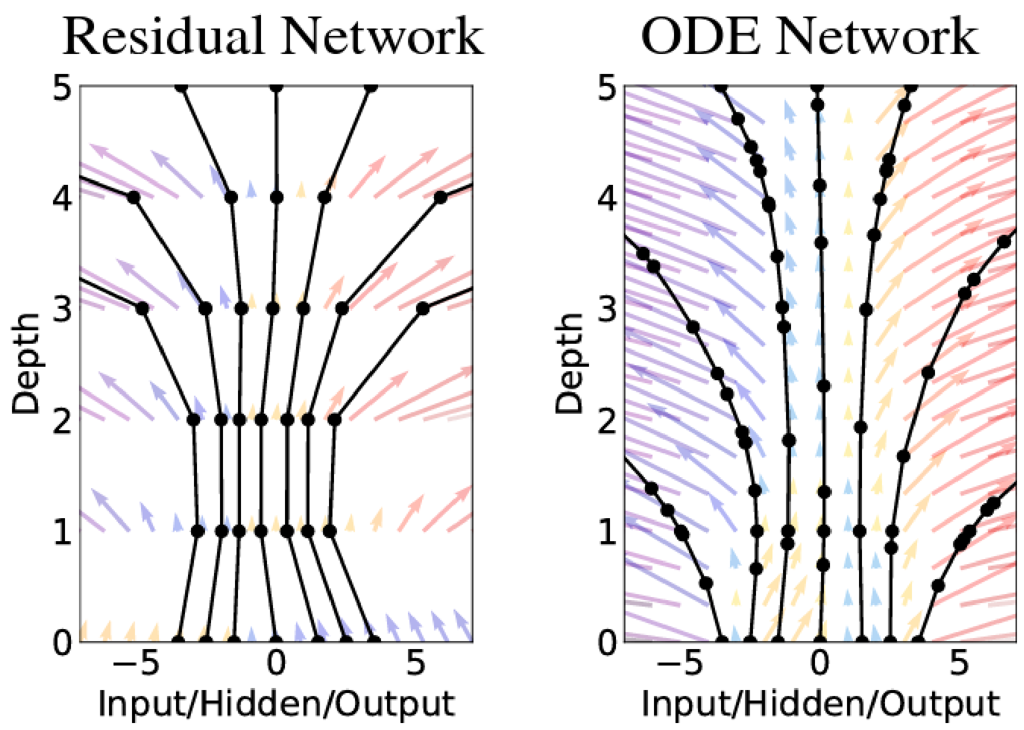
Why on earth should this work?

DNNs model real-world physics well because they have similar structure

Nonlinear, hierarchical, high-dimensional, noisy, analog, local, sparse,…

Neural ordinary differential equations

$$\mathbf{h}_{t+1} = \mathbf{h}_t + f(\mathbf{h}_t, \theta_t) \longrightarrow \frac{d\mathbf{h}(t)}{dt} = f(\mathbf{h}(t), t, \theta)$$



R.T.Q. Chen et al., *NeurIPS* (2018)

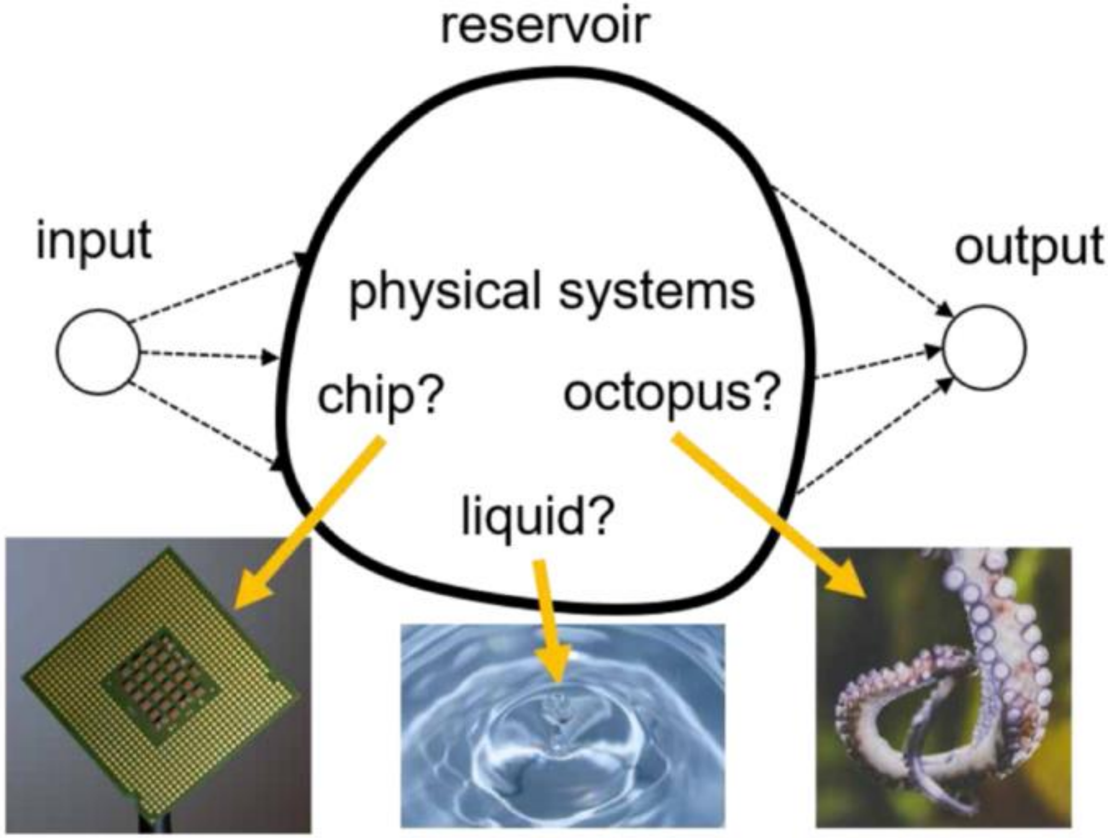
Why on earth should this work?

DNNs model real-world physics well because they have similar structure

Nonlinear, hierarchical, high-dimensional, noisy, analog, local, sparse,…

Neural ordinary differential equations

Physical reservoir computing



K. Nakajima, *Japanese Journal of Applied Physics* (2020)

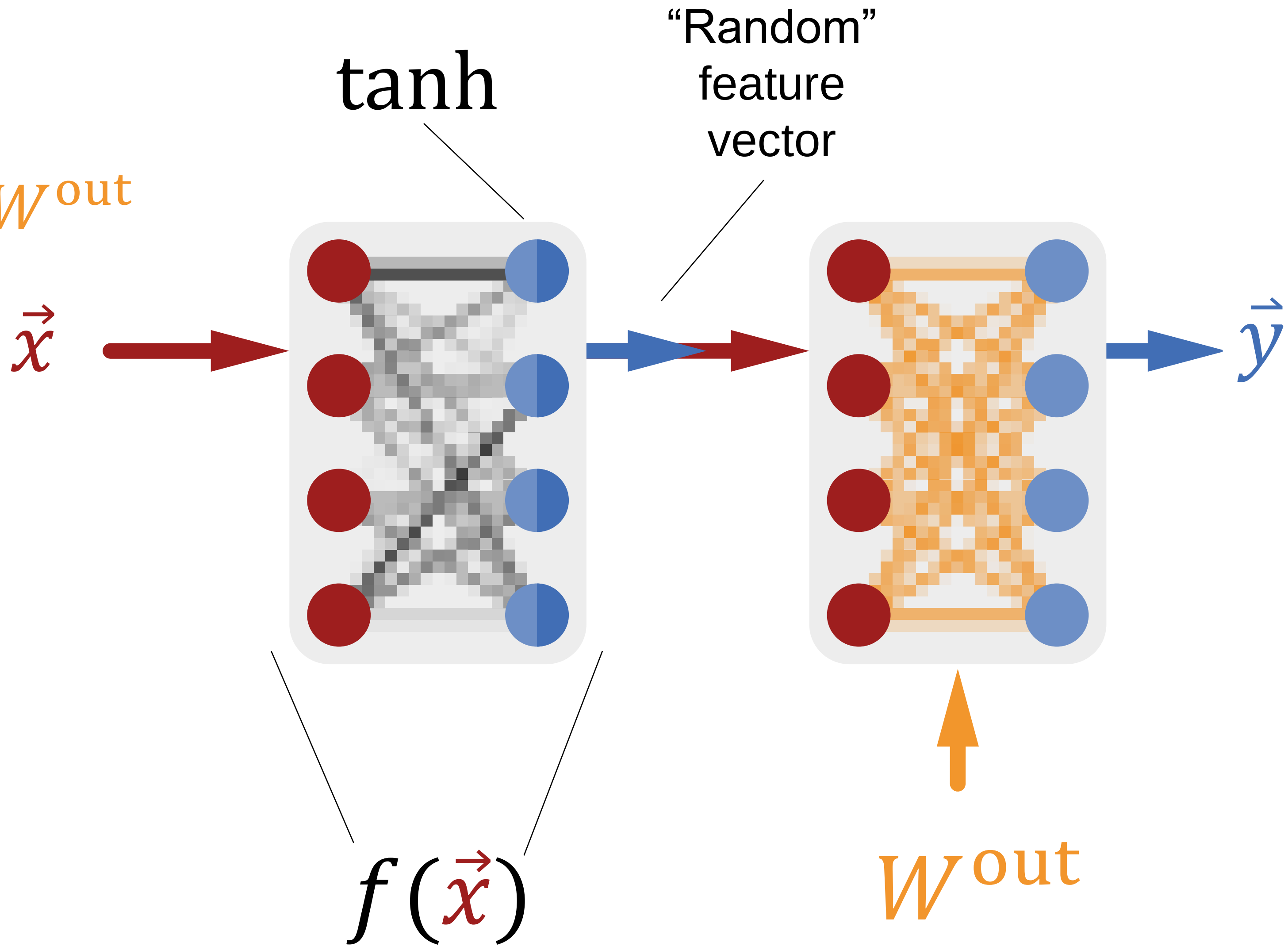
Random features – echo state, liquid state, "extreme learning"

Untrained random neural network

Trained linear digital output layer, W^{out}

Fast + stable training

$$\vec{y} = W^{\text{out}} f(\vec{x})$$



M. Lukoševičius. "A practical guide to applying echo state networks." *Neural networks: Tricks of the trade*. (2012).

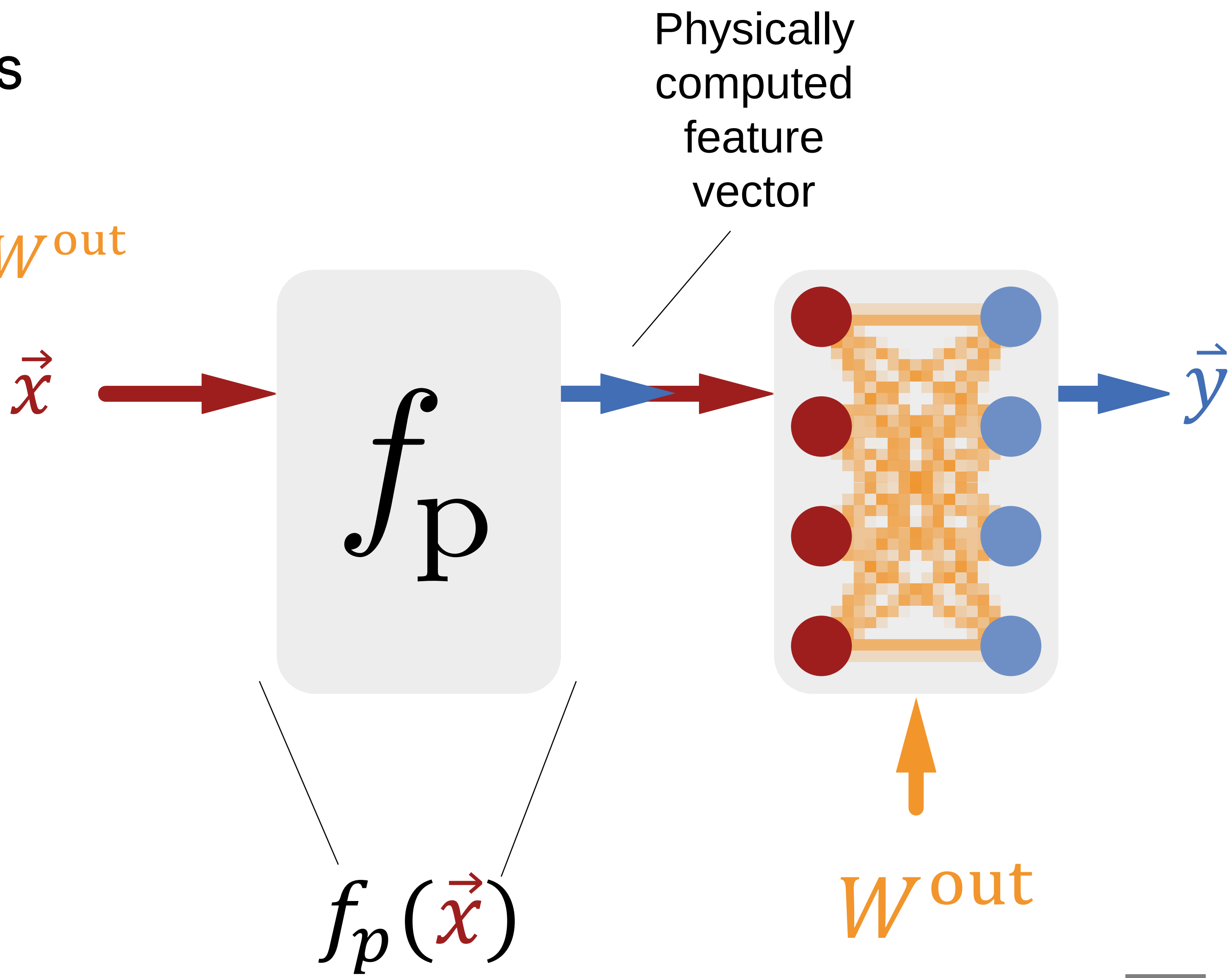
Physical reservoir computing

Untrained physical transformations

Trained linear digital output layer, W^{out}

Fast + stable training

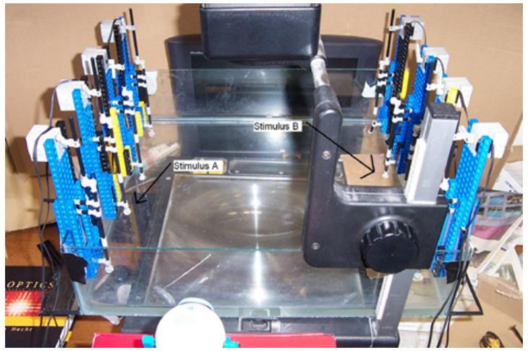
$$\vec{y} = W^{\mathrm{out}} f_{\mathrm{p}}(\vec{x})$$



K. Nakajima, *Japanese Journal of Applied Physics* (2020)

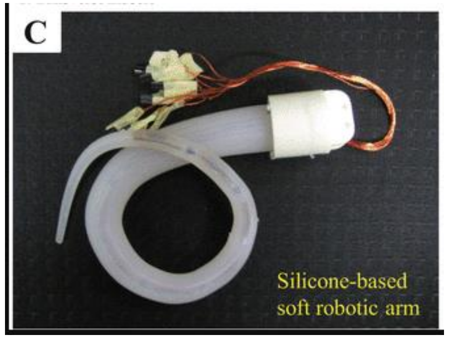
A marvelous range of things provide USEFUL physical features!

A bucket of water



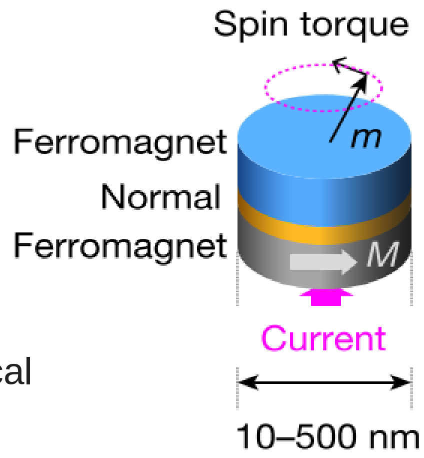
Fernando and Sojakka. "Pattern recognition in a bucket." *European Conference on Artificial Life* (2003).

Octopus arms



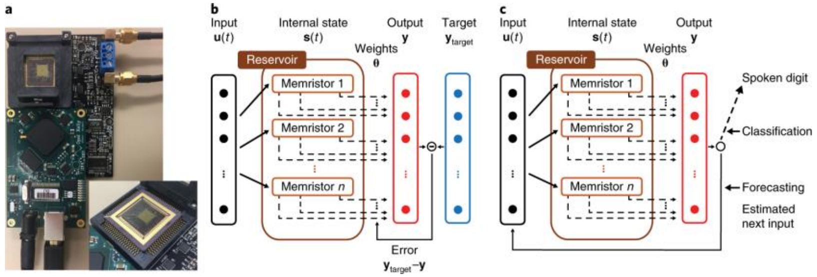
Nakajima, "Muscular-hydrostat computers: Physical reservoir computing for octopus-inspired soft robots." *Brain Evolution by Design* (2017)

Nano-oscillators (spintronic)



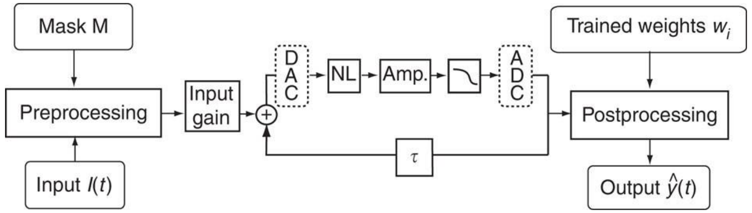
Torrejon et al. "Neuromorphic computing with nanoscale spintronic oscillators." *Nature* (2017)

Nonlinear analog electronics (memristors)



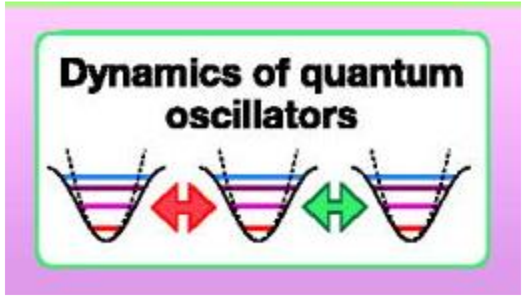
Moon, et al. "Temporal data classification and forecasting using a memristor-based reservoir computing system." *Nature Electronics* (2019)

Optoelectronic loops and networks



Appeltant et al. "Information processing using a single dynamical node as complex system." *Nature Communications* (2011)

Quantum nonlinear oscillators



Marković & Grollier, Appl. Phys. Lett (2020)

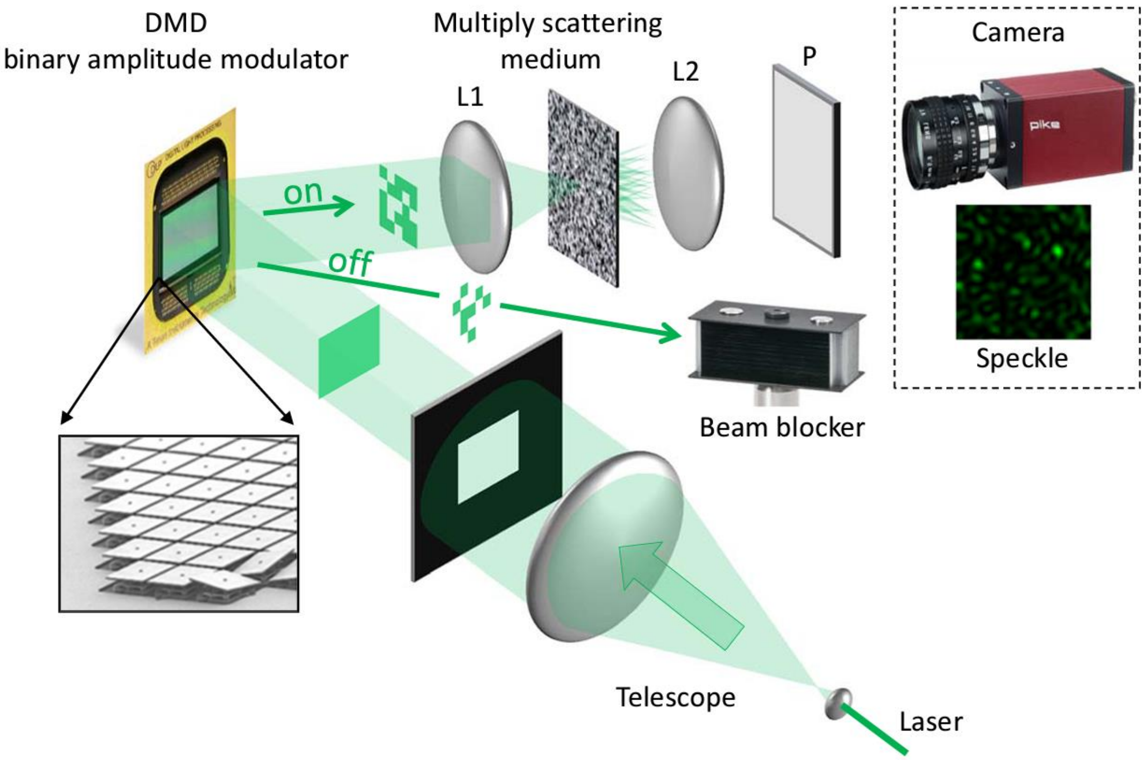
And many more…

Many such features are computed physically with VASTLY more energy-efficiency than is possible with digital electronics

Just one example:

Random matrix-vector features at: ~100 analog Peta-operations/s
~10 aJ/op

10^6 more efficient than GPU*

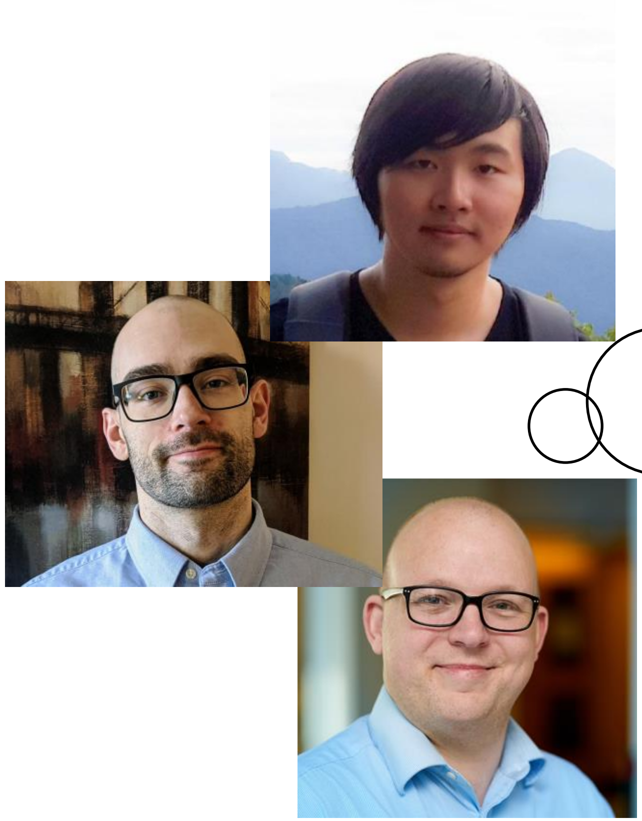


A. Saade et al. International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2016

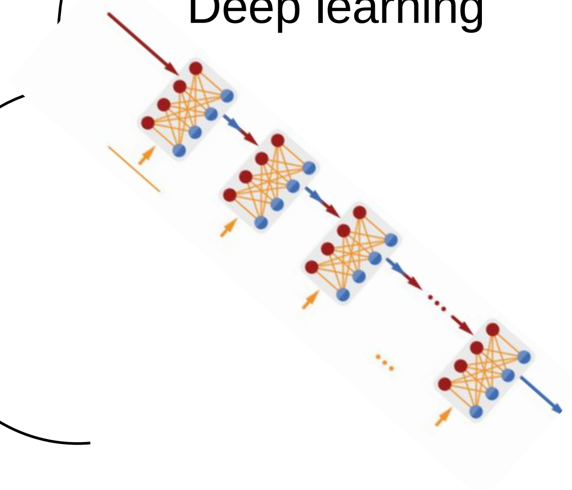
See poster
by Fei Xia, ENS

*for random matrix-vector operations, see supplementary section 3 of arXiv:2104.13386

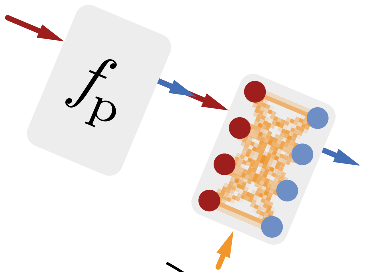
Circa 2018-2020…



Deep learning

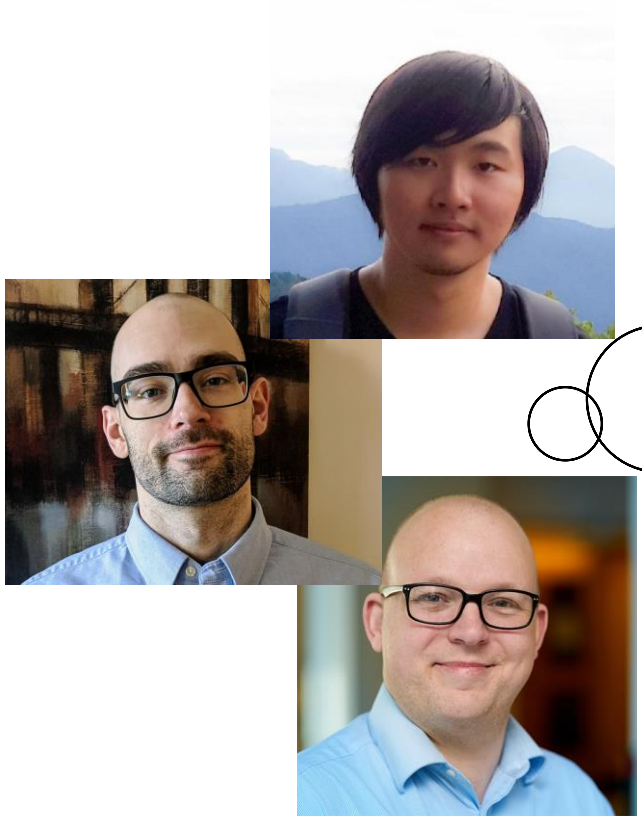


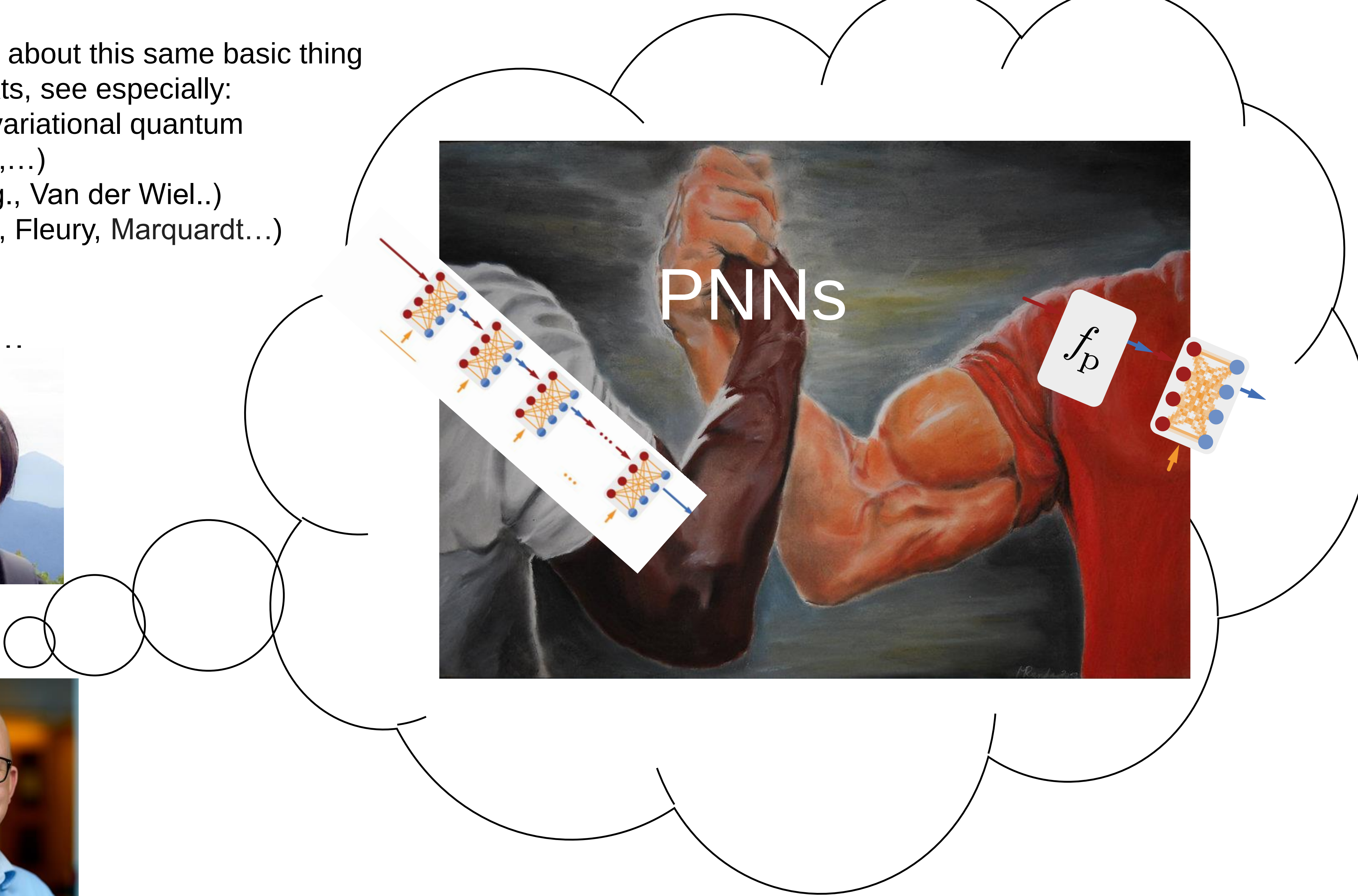
Physical reservoir computing



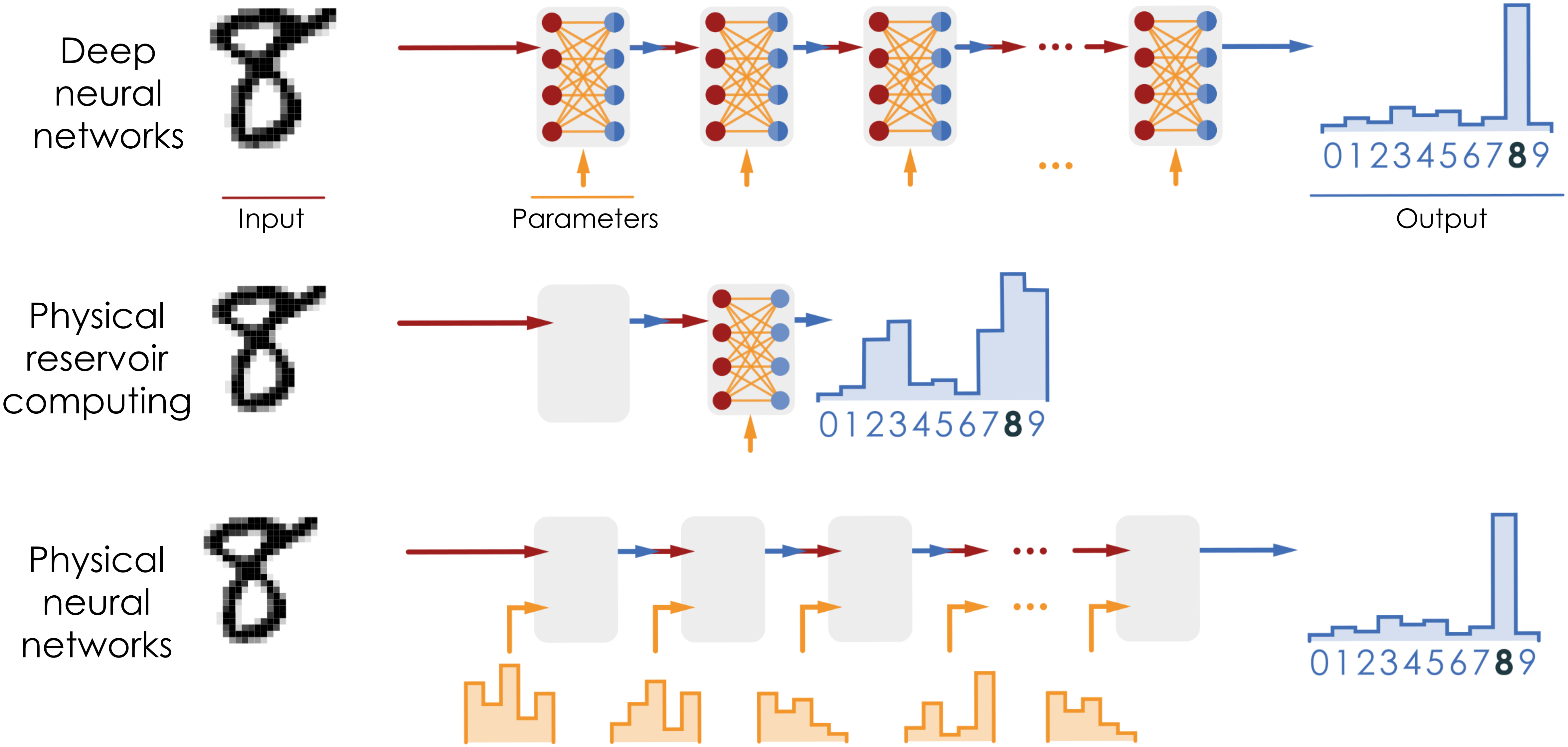
**Many others were thinking about this same basic thing too, albeit in different contexts, see especially:
- quantum circuit learning / variational quantum algorithms (e.g., Fujii, Coles,…)
- “In materio computing” (e.g., Van der Wiel..)
- wave computing (e.g., Fan, Fleury, Marquardt…)

Circa 2018-2020…





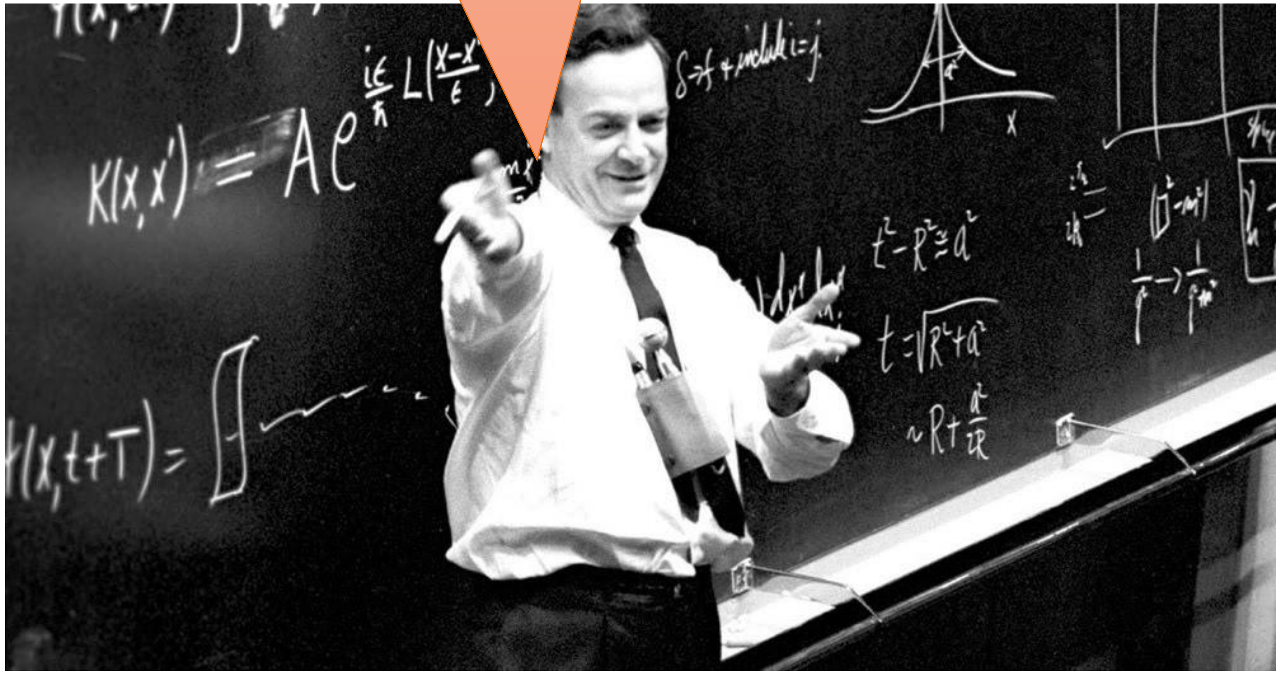
Physical neural networks combine the key ingredients of deep learning with the physics-first opportunism of reservoir computing



What would be the pay-off if it works?

There is plenty of room at the bottom!
(for hardware innovation)

- Automated physics-first computing!
- *(Potentially!) HUGE* speed up + energy-efficiency boosts for DNN-like calculations
- Learn complex *physical* functions (e.g., “smart” sensing, micromachines)



An example physical neural network

Classifying images with coupled nonlinear oscillators

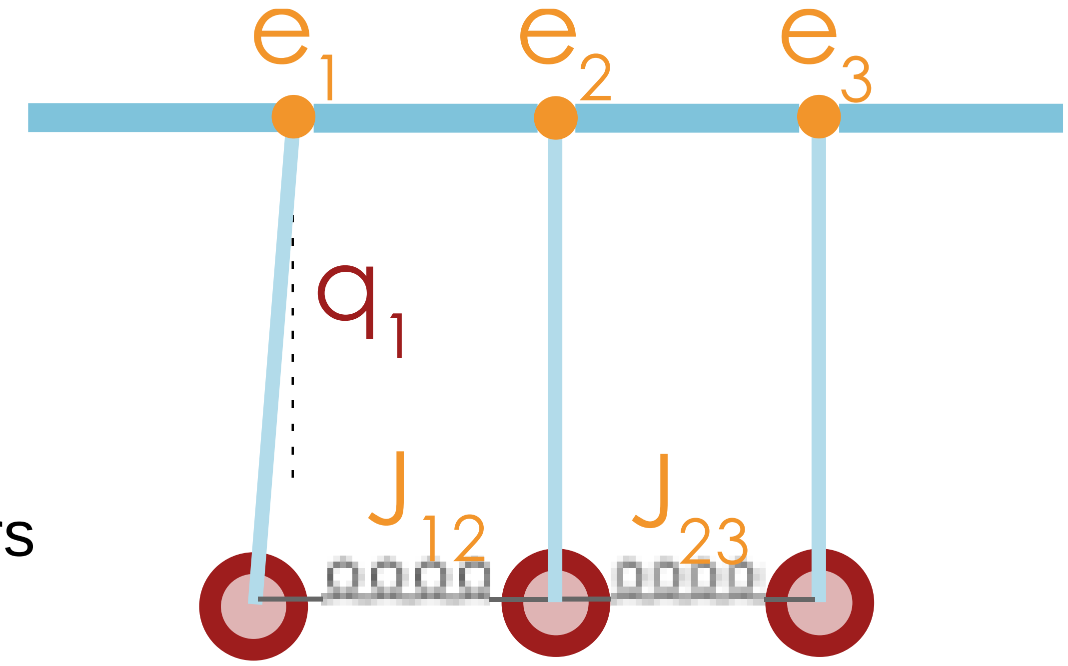
$$\frac{\mathrm{d}^2 q_i}{\mathrm{d}t^2} = -\sin q_i + \sum_{j=1}^{N} \boxed{J_{ij}} (\sin q_j - \sin q_i) + \boxed{e_i}$$

Input data = initial ($t = 0$) angles

Parameters = coupling between oscillators (spring stiffness)

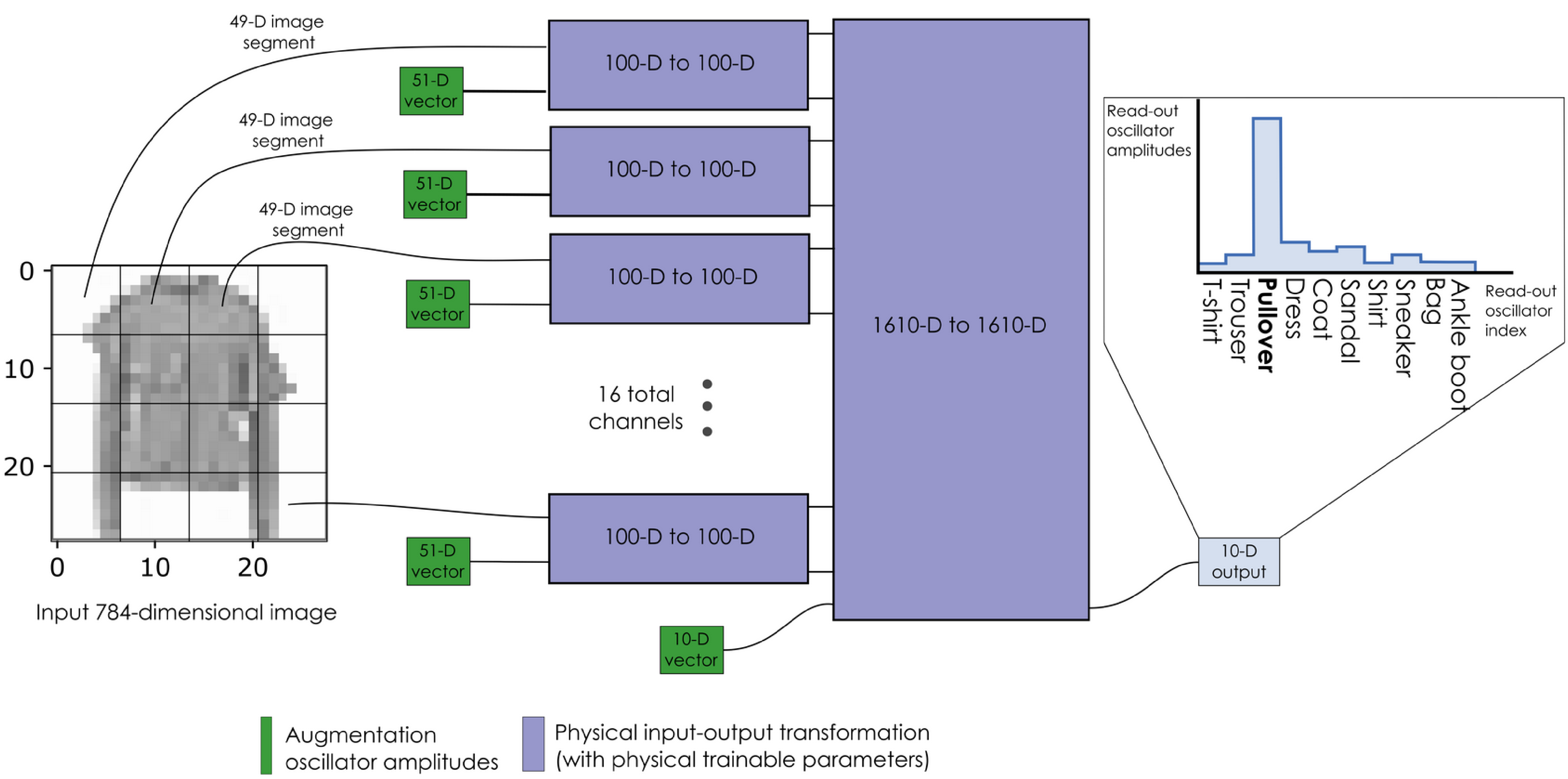
drive (fixed torque at joint)

Output = Later ($t = T$) angles of the oscillators

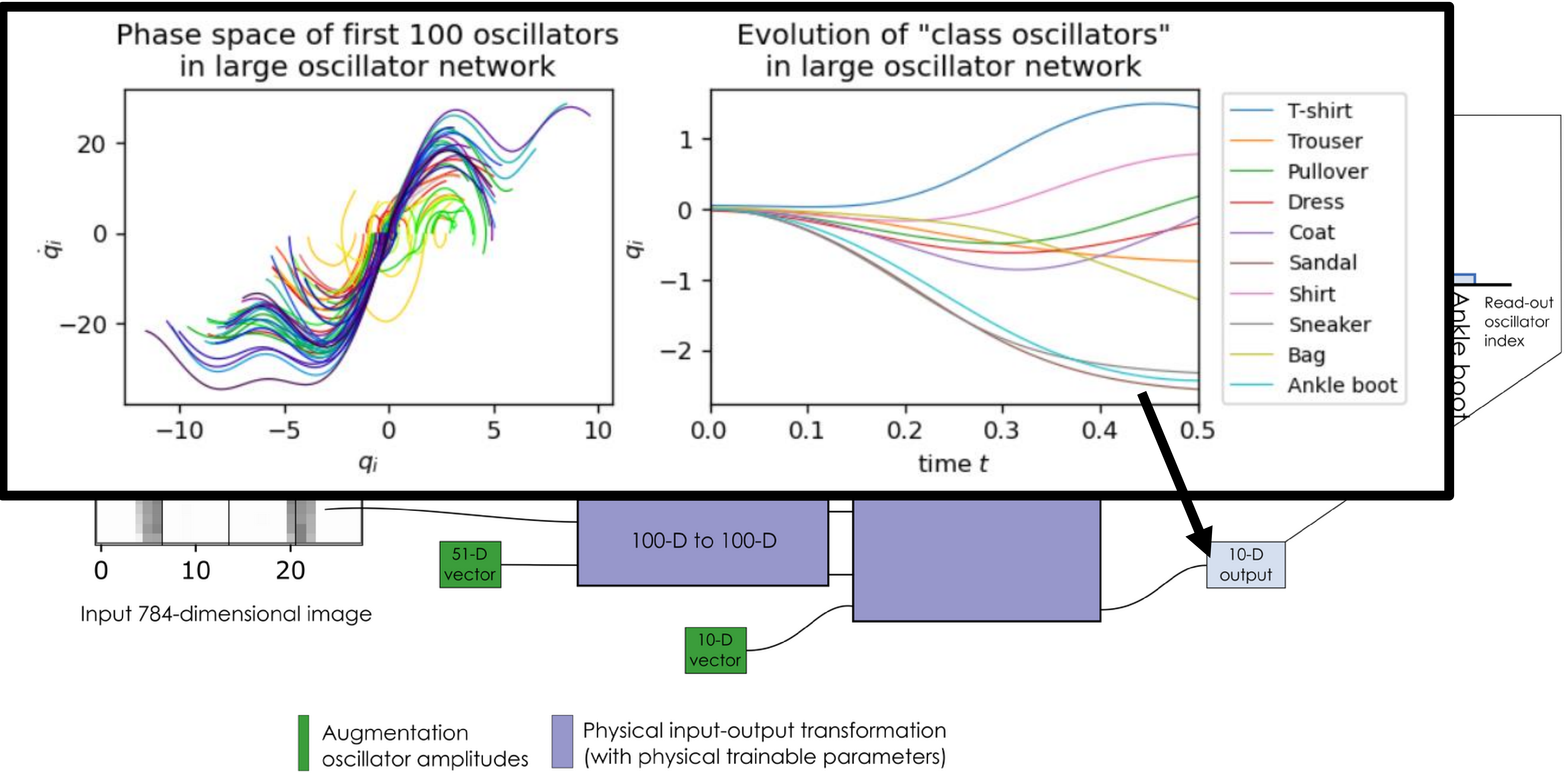


https://github.com/mcmahon-lab/Physics-Aware-Training

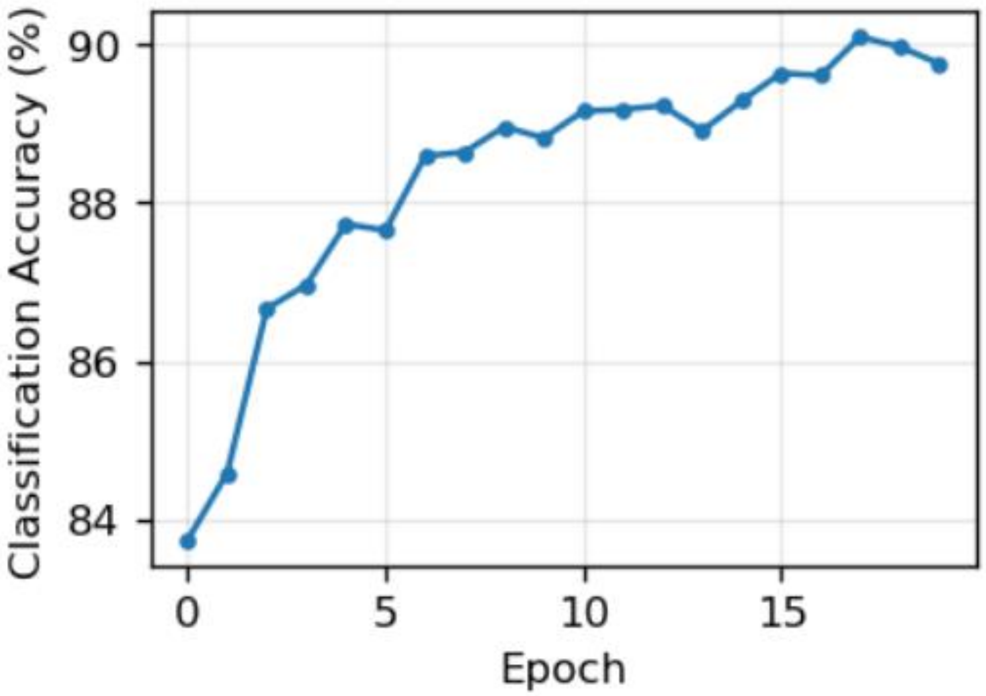
Physical neural network architecture

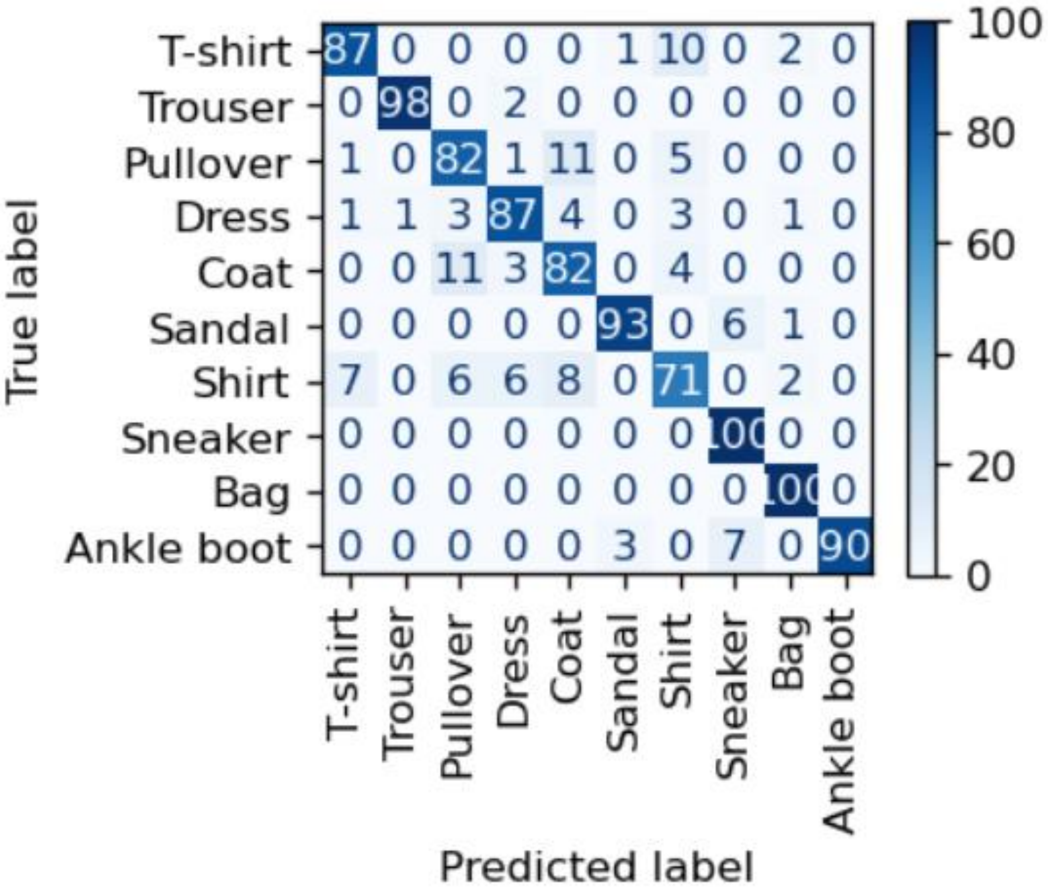


Physical neural network architecture



Classifying fashion images with an oscillator-PNN

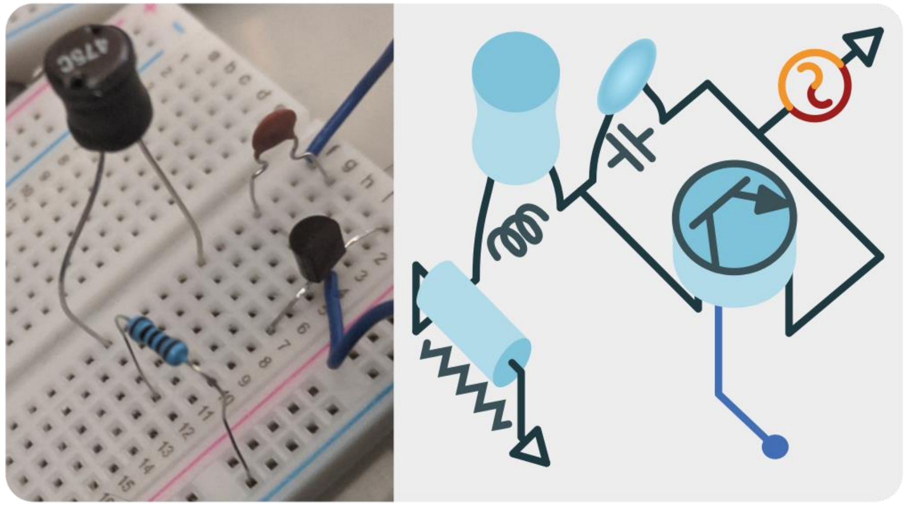




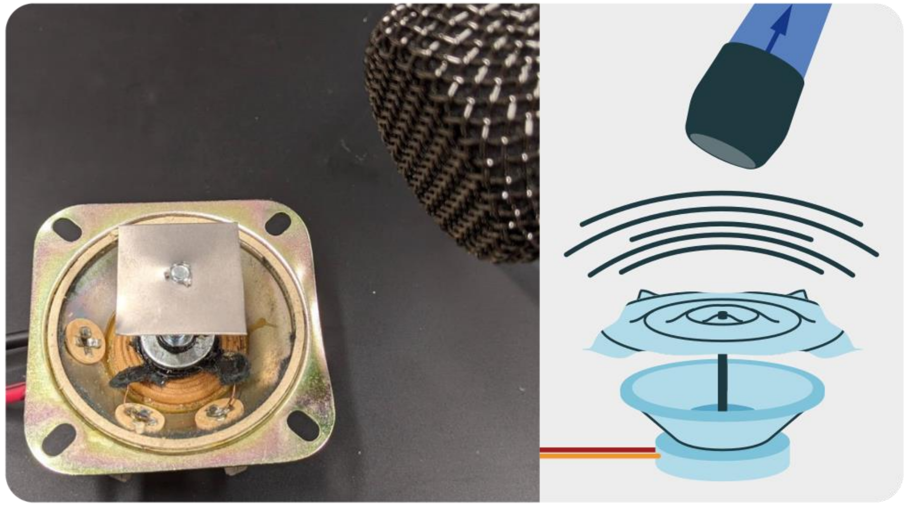
Can *everything* be a neural network?

Yes! (but not always a good one)

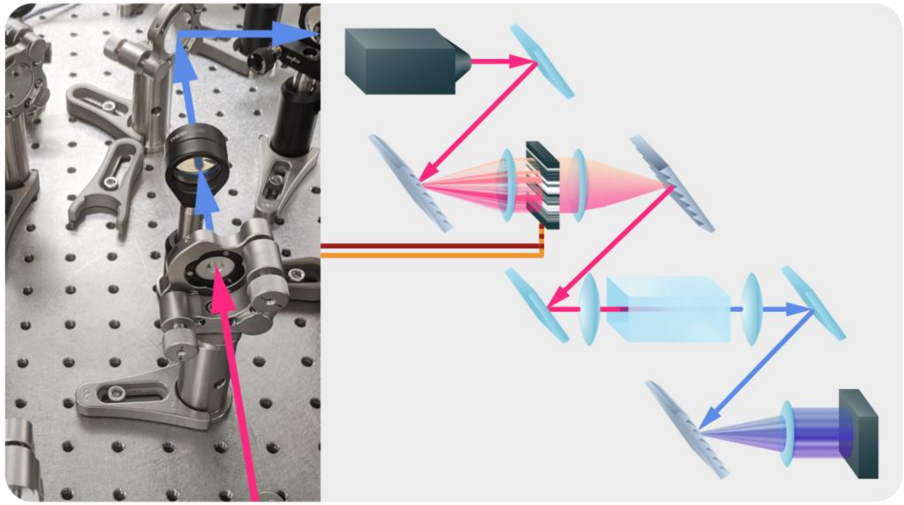
Electronics



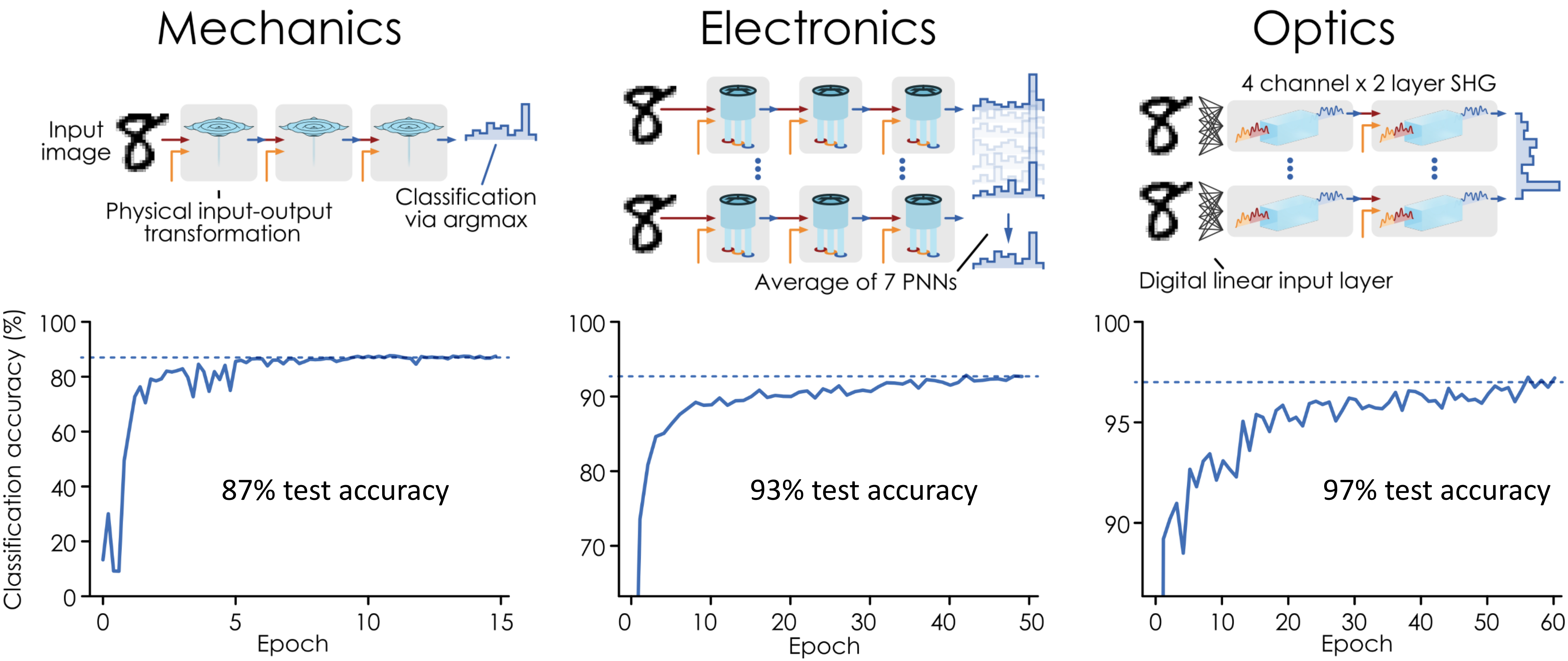
Mechanics



Optics



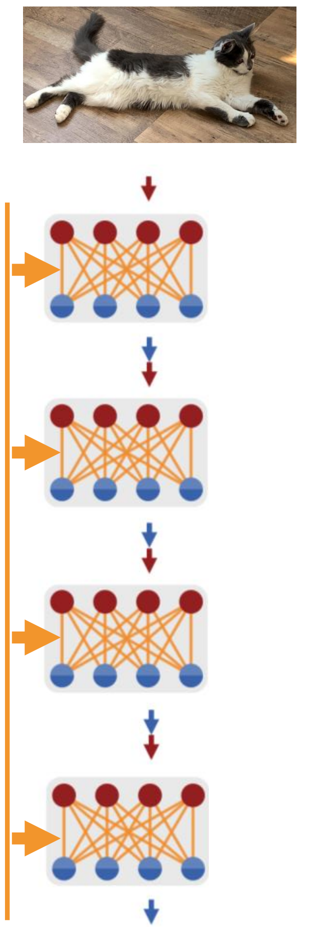
Diverse PNNs for handwritten digit image classification



Wright*, Onodera*, Stein et al., Nature (2022)

Deep neural networks: training versus inference

Untrained



Untrained Parameters

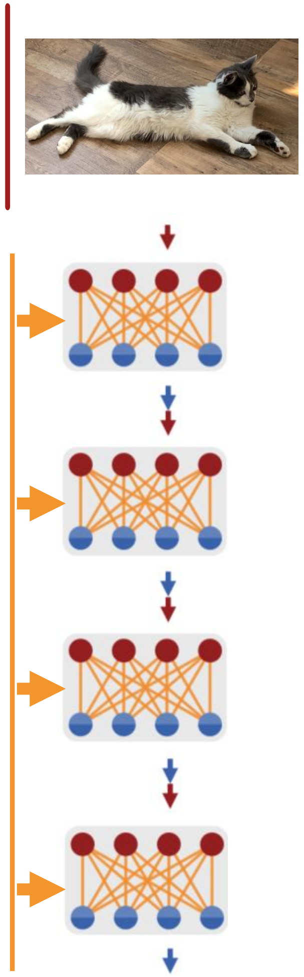
Nonsense

Training

Training input data

Parameters are **changed**

$$\vec{\theta} \rightarrow \vec{\theta} + d\vec{\theta}$$

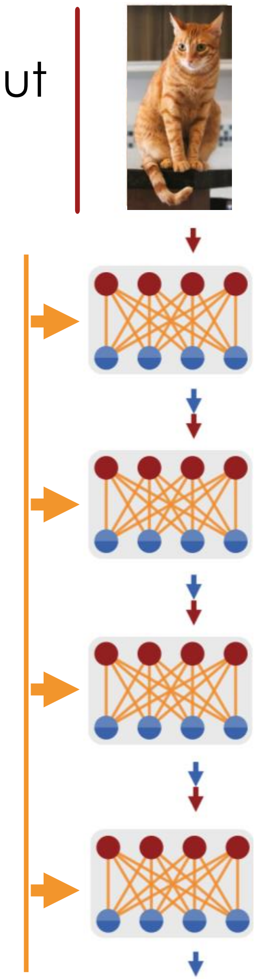


“Cat”

Inference

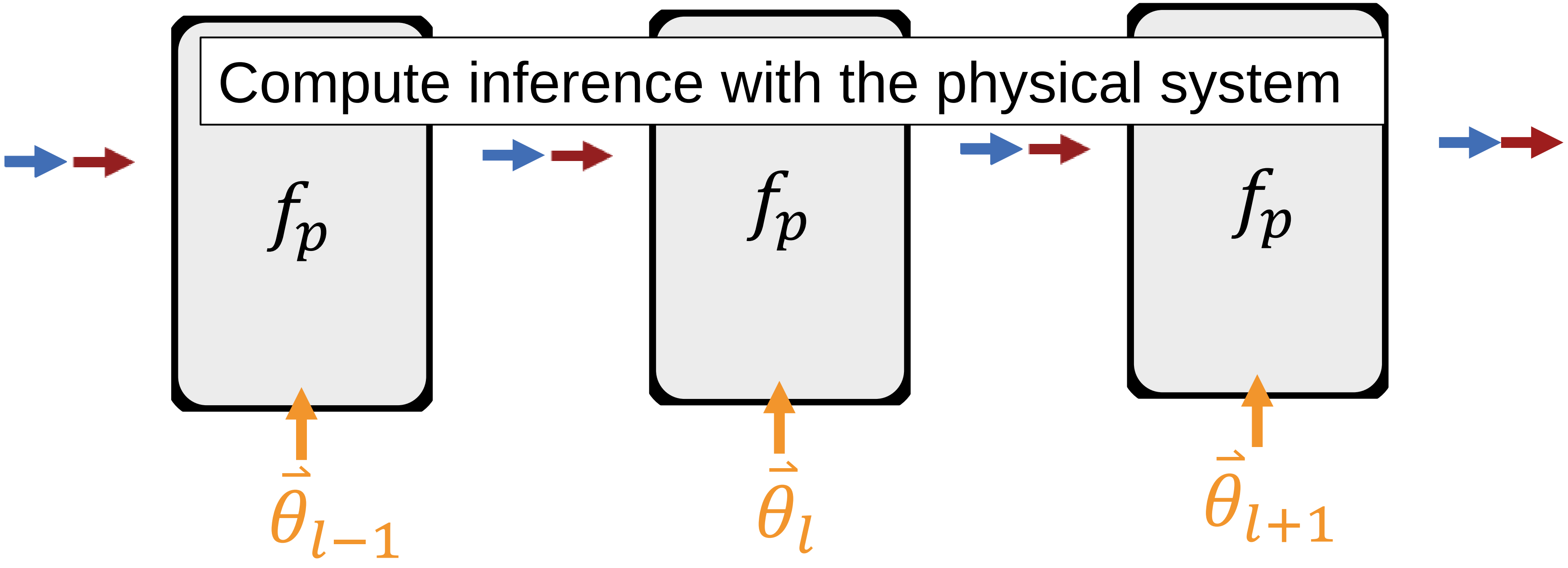
Unseen new input data

Parameters are **fixed**



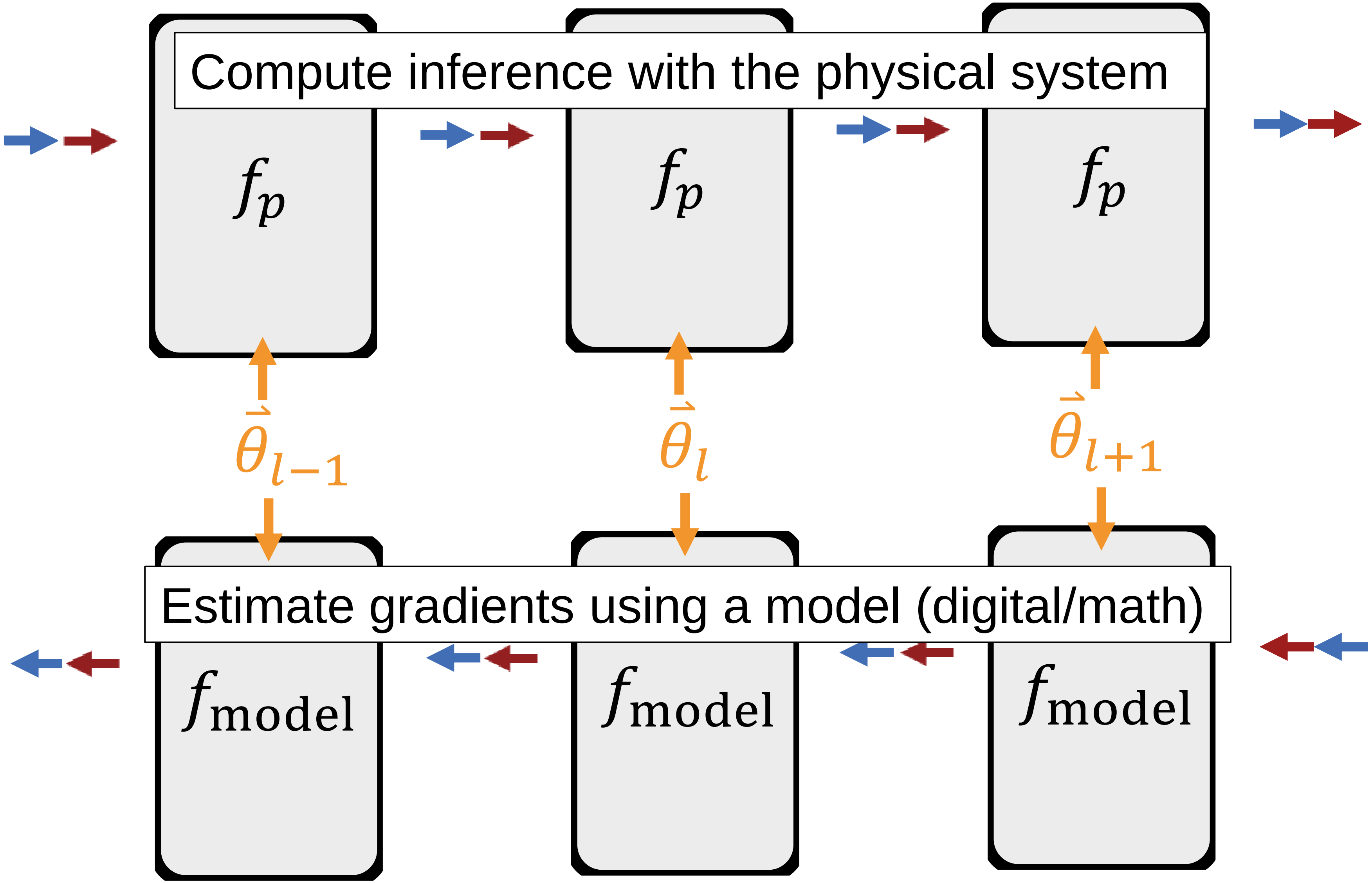
“Cat”

Physics-aware training: Backpropagation through f_p



Wright*, Onodera* et al., Nature (2022)

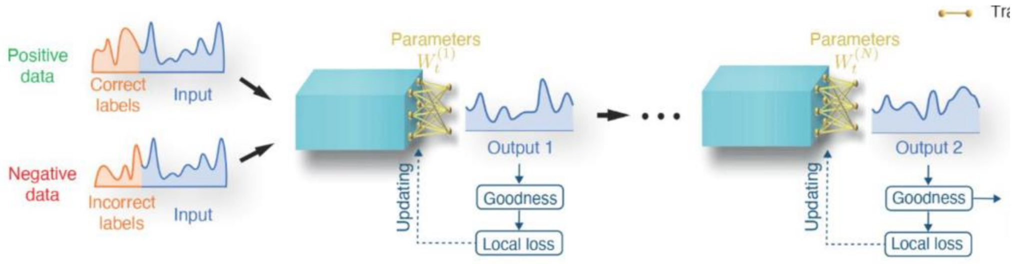
Physics-aware training: Backpropagation through f_p



Wright*, Onodera* et al., Nature (2022)

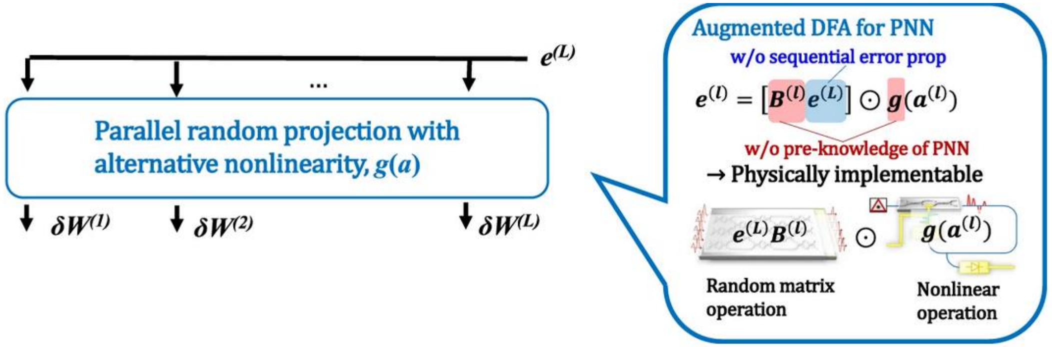
Training PNNs *beyond* physics-aware training

Forward-forward (layer-by-layer)



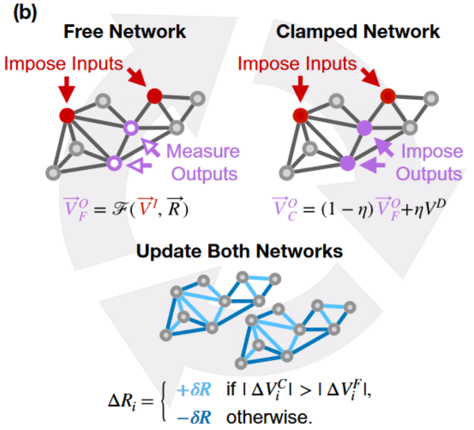
Momeni et al., Science (2023)
Hinton, NeurIPS (2023)

Direct feedback alignment



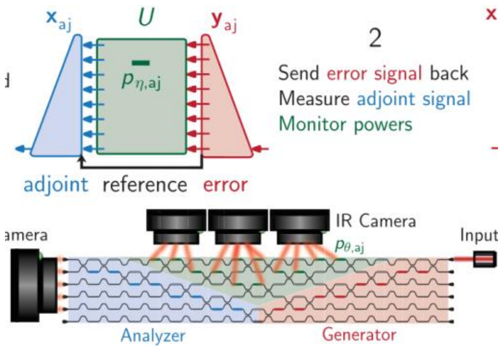
Nakajima et al., Nat. Comm (2022)
Lillicrap et al., Nat. Comm (2016)

Equilibrium propagation / coupled learning



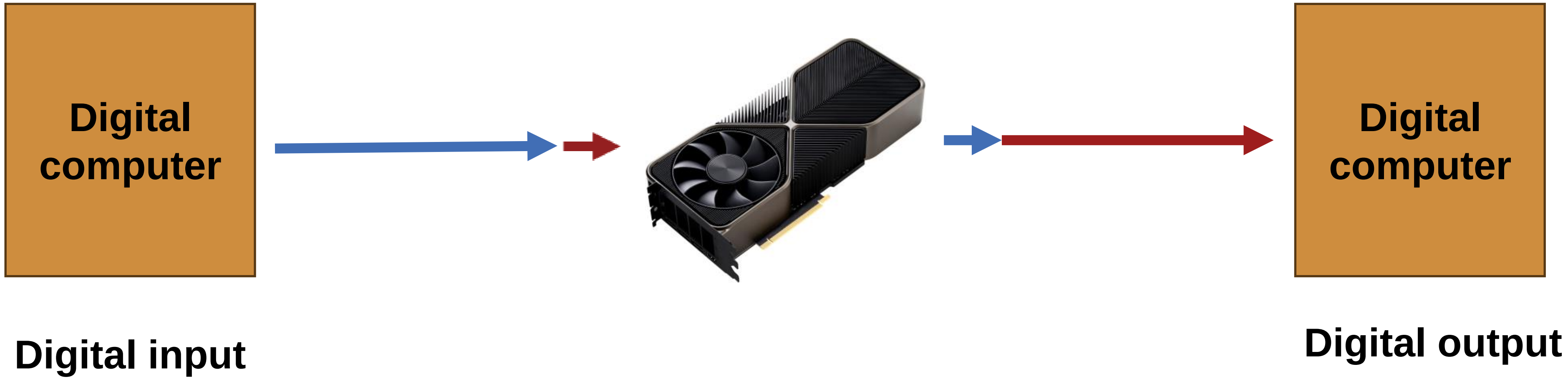
Scellier & Bengio, Frontiers in Comp. Neuro (2017)
Dillavou, Stern, Liu & Durian, Phys Rev Applied (2022)
Laydevant, Ernoult, Querlioz & Grollier, CVF (2021)

“Physical adjoint”

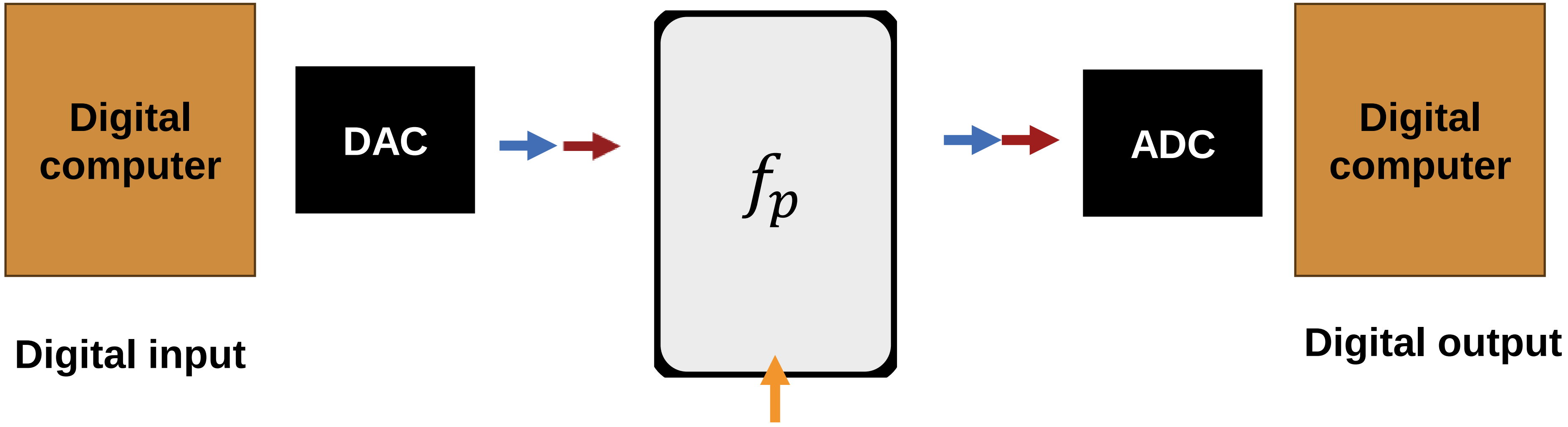


Pai et al., Science (2023)
Lopez-Pastor and Marquardt, PRX (2023)

What can we do with PNNs?



“Deep learning accelerator”



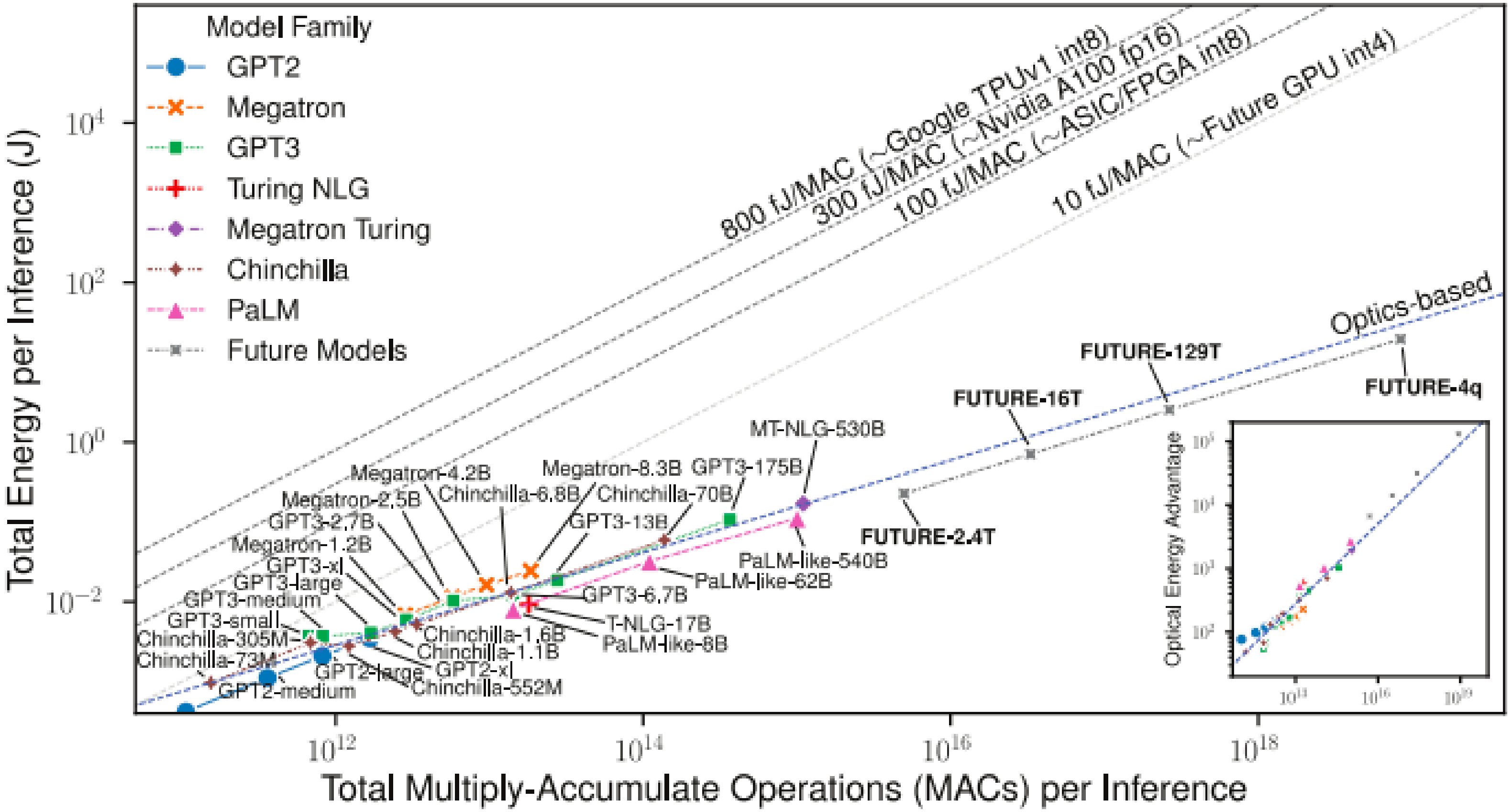
"Deep learning accelerator"

PNNs for deep learning acceleration

Peter McMahon

Tianyu Wang

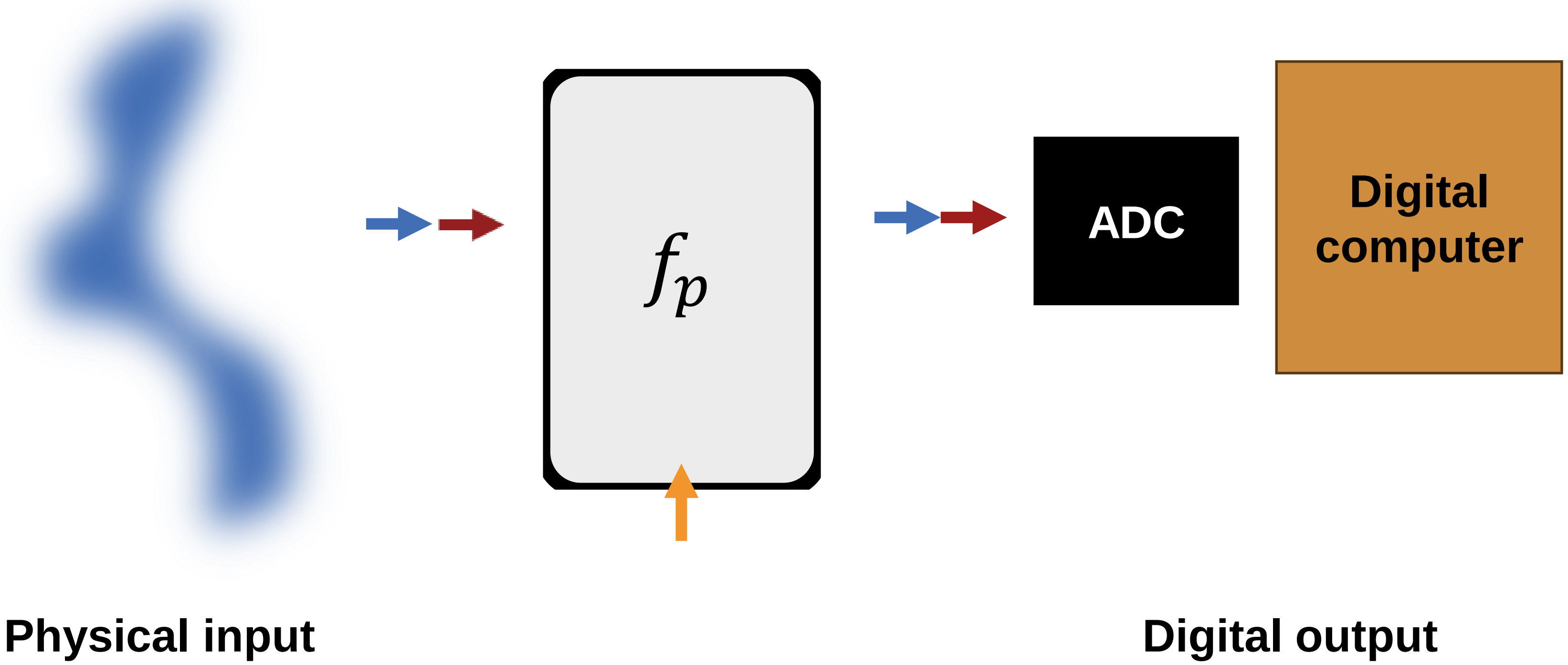
Maxwell Anderson



TLDR:

Optics has **fundamental scaling advantage** – prospect for 100,000x efficiency gain for future Transformer models!

M.G. Anderson, S.-Y Ma, T. Wang, L.G. Wright, and P.L. McMahon. Optical Transformers. arXiv:2302.10360.



“Smart sensor”

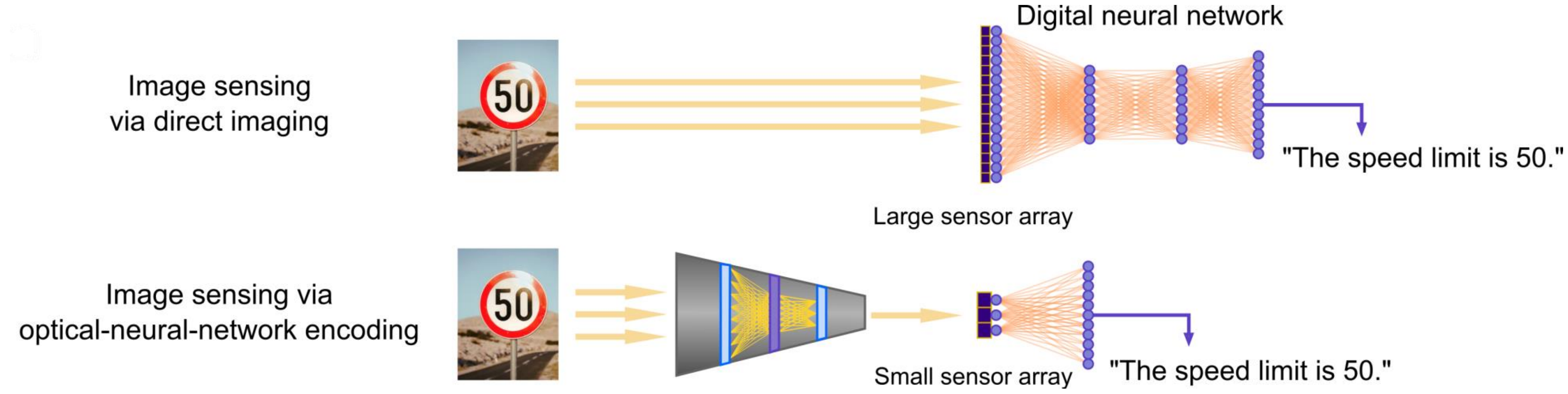
PNNs for smart sensing

See Mandar and Tianyu’s poster(s)!

Peter McMahon

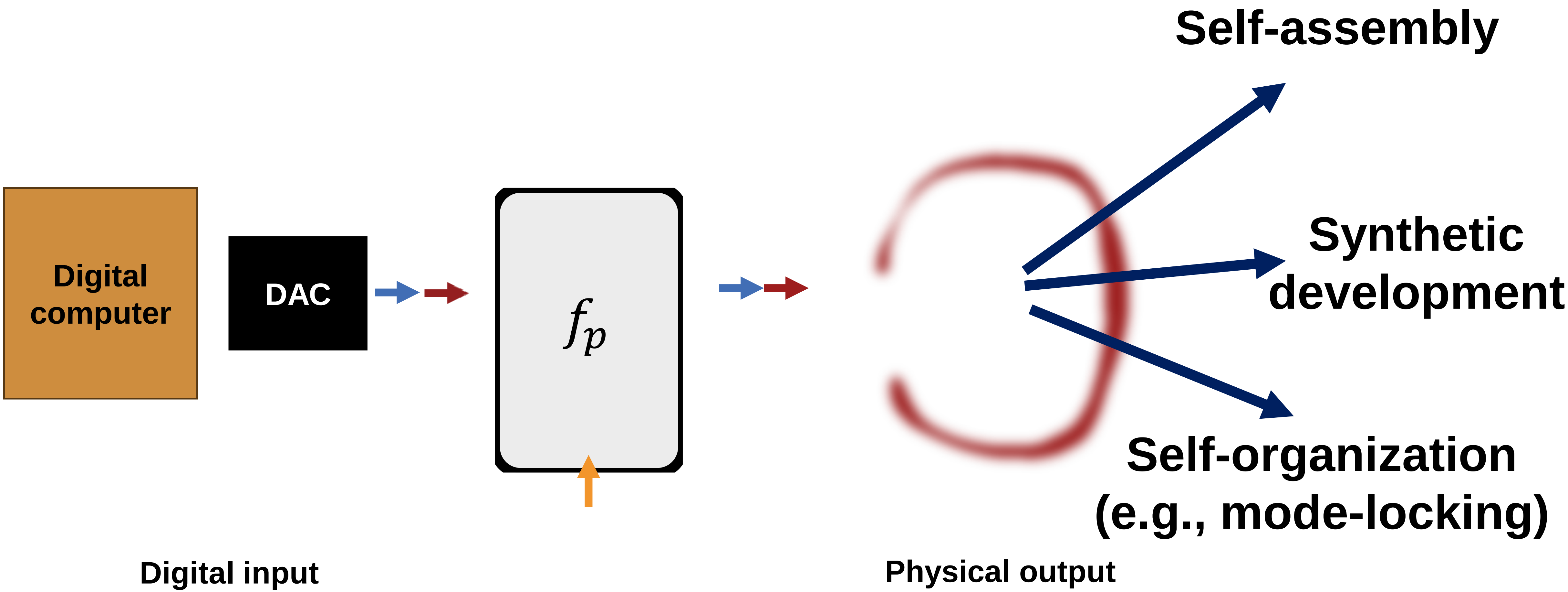
Tianyu Wang

Mandar Sohoni

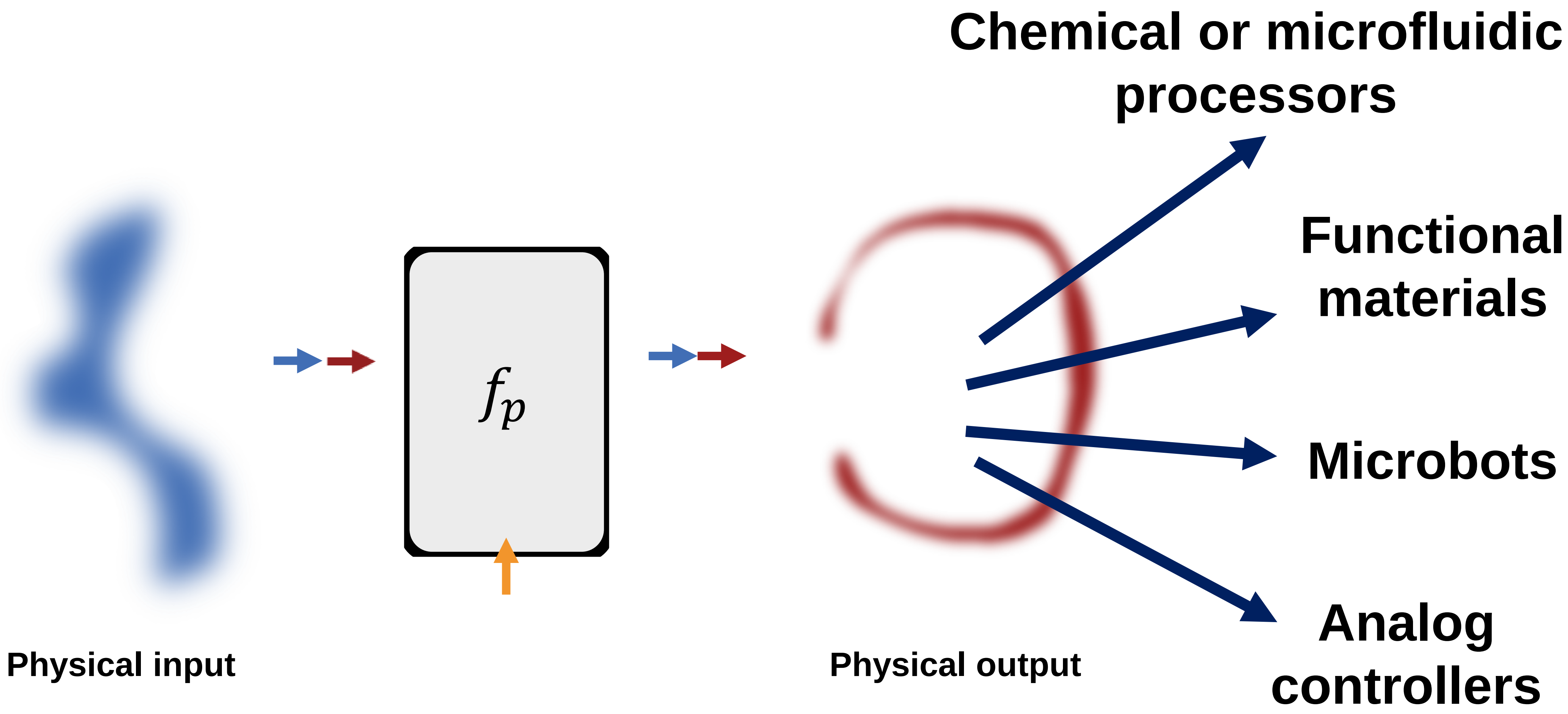


TLDR:
Optical neural network pre-processing allows faster, more efficient machine vision

T. Wang*, M. M. Sohoni*, L. G. Wright, …, P. L. McMahon. “Image sensing with multilayer, nonlinear optical neural networks” *Nature Photonics (2023)*



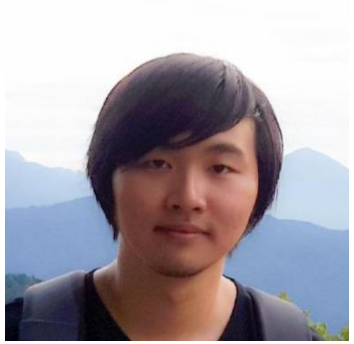
“Physical neural network generator”



"Physical neural network machine"

PNNs for learning photonic devices

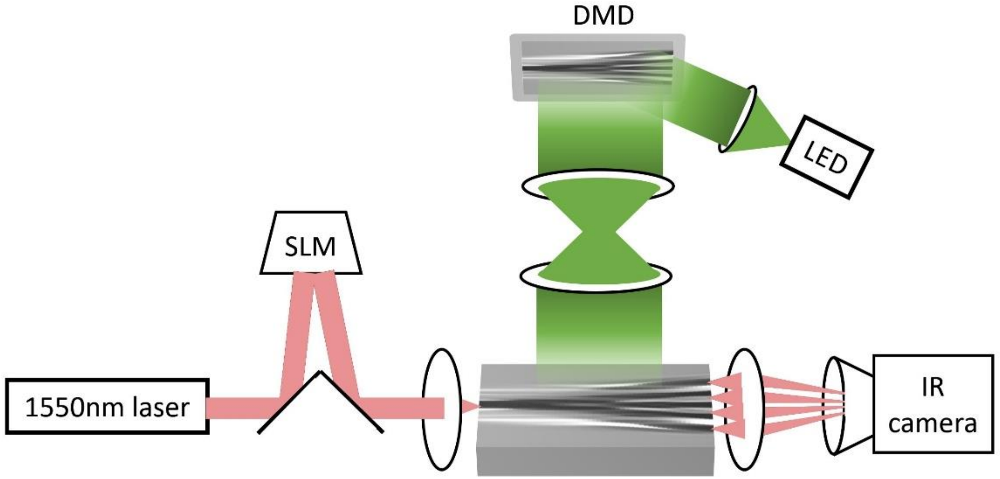
Peter McMahon

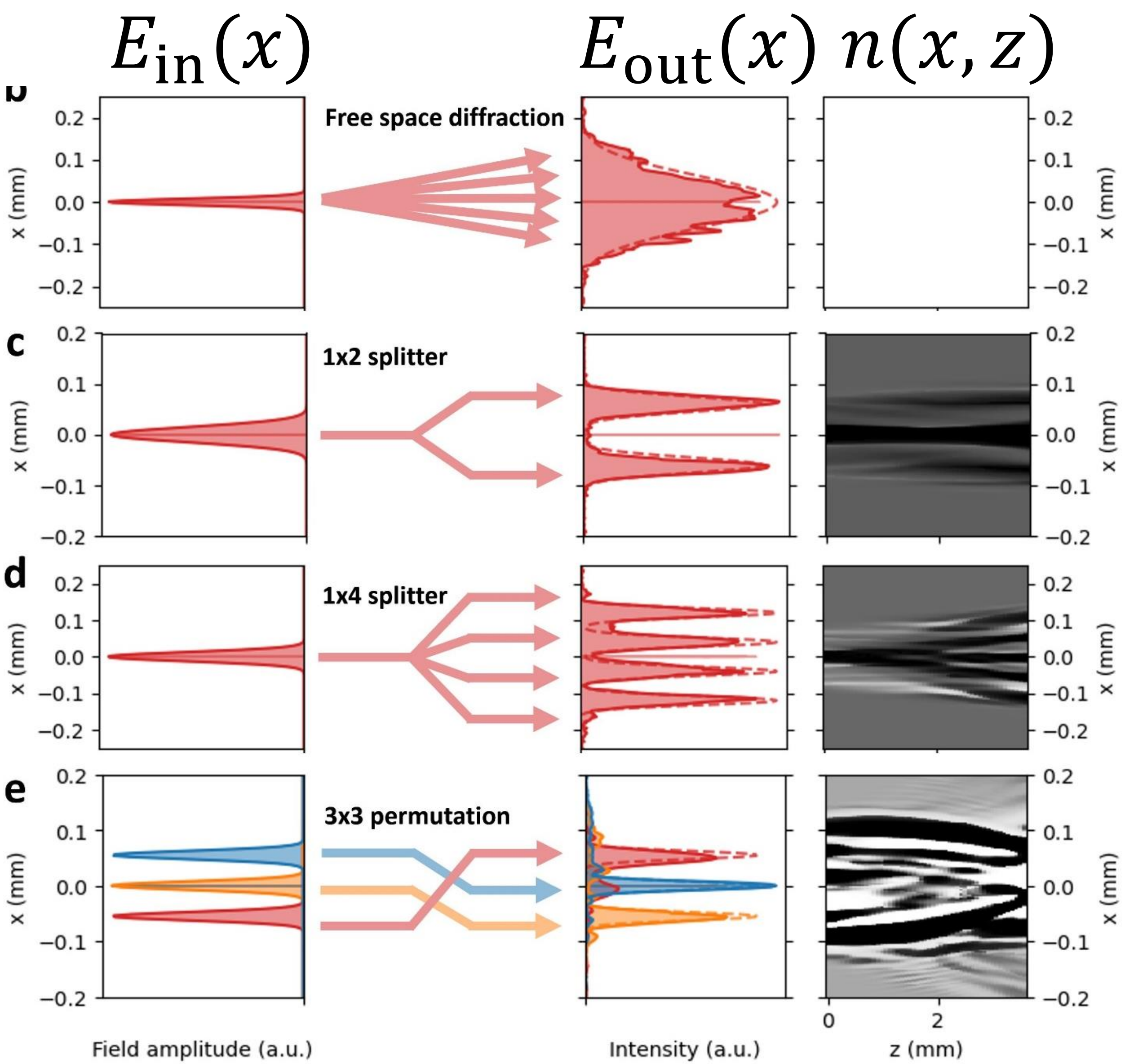


Tatsuhiro Onodera

Martin Stein

Recall Hiro's talk yesterday,
See his poster!





T. Onodera*, M.M. Stein* et al., arXiv soon….

Some parting thoughts

Software is *already* “physics-informed”

In this paper we also make the converse claim; **that the state of computer architecture has been a strong influence on our models of thought**.

R. A. Brooks in “Intelligence without reason” (1991)

S. Hooker "The hardware lottery." *Communications of the ACM* (2021)

Software is *already* “physics-informed”, but not quite *purposefully*

In this paper we also make the converse claim; **that the state of computer architecture has been a strong influence on our models of thought**.

R. A. Brooks in “Intelligence without reason” (1991)

S. Hooker "The hardware lottery." *Communications of the ACM* (2021)

Hardware physics constrains communal optimization of algorithms

$$\text{Transformers} \approx \max_{\{\text{algorithms}\}} (\text{"AI goodness"})$$

Jeremie Laydevant*

*Channeling decades of ideas in neuromorphic computing and theoretical neuroscience…

Hardware physics constrains communal optimization of algorithms

$$\text{Transformers} \approx \max_{\{\text{algorithms}\}} (\text{"AI goodness"})$$

$$\text{subject to: GPU}$$

Jeremie Laydevant*

*Channeling decades of ideas in neuromorphic computing and theoretical neuroscience…

Towards purposeful physics-constrained software-hardware

$$??? \approx \max_{\{\text{algorithms}\}} (\text{"AI goodness"})$$

$$\text{subject to: physics alone}$$

Laydevant*, Wright*, Wang & McMahon "The hardware is the software", Neuron (2023)

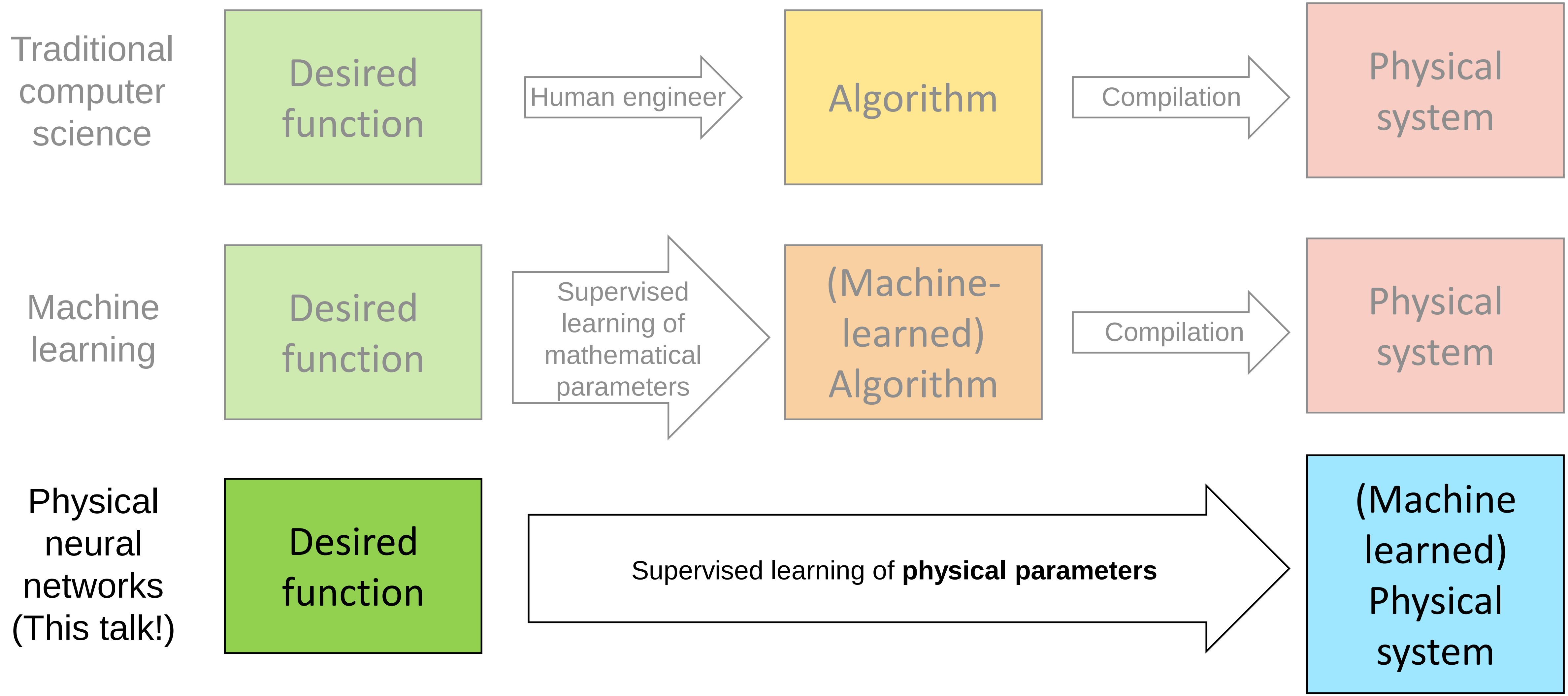
Towards purposeful physics-constrained software-hardware

$$??? \approx \max_{\{\text{algorithms}\}} (\text{"AI goodness"})$$

$$\text{subject to: physics alone}$$

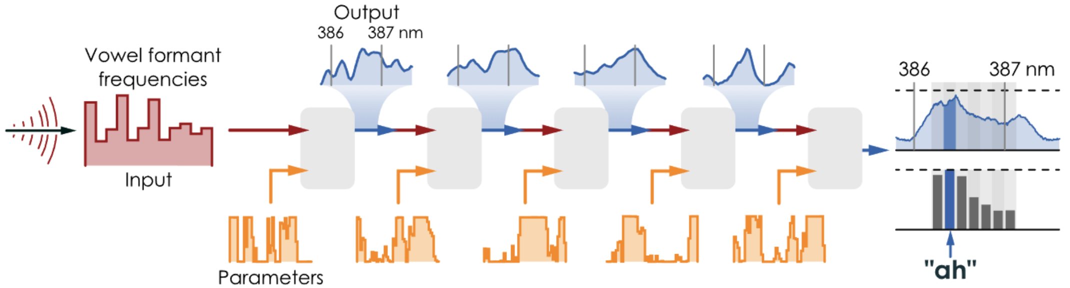
(Also features laser-based aliens…lighfeforms)

Laydevant*, Wright*, Wang & McMahon “The hardware is the software”, Neuron (2023)


Traditional computer science
Desired function
Human engineer
Algorithm
Compilation
Physical system
Machine learning
Desired function
Supervised learning of mathematical parameters
(Machine-learned) Algorithm
Compilation
Physical system
Physical neural networks (This talk!)
Desired function
Supervised learning of **physical parameters**
(Machine learned) Physical system

Contributions

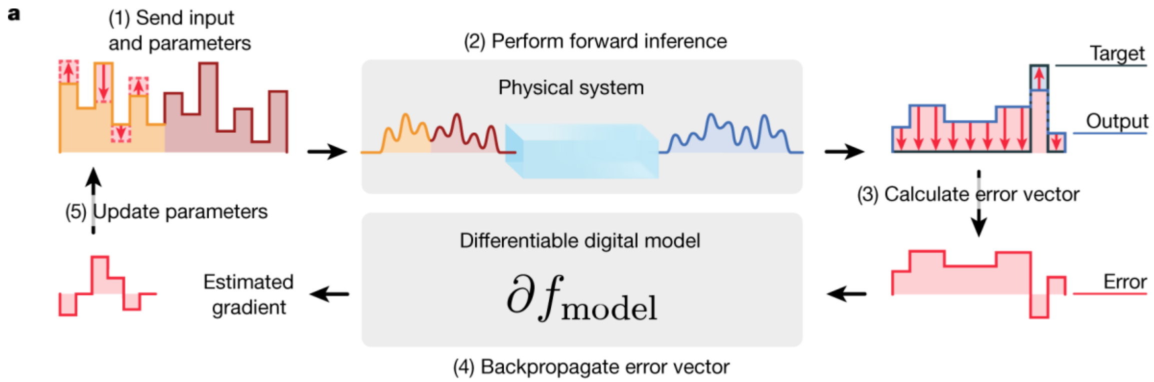
(Deep) physical neural networks



First demonstrations of PNNs: DNN-like calculations with networks of trained physical data transformations.

Potential for:
→ *Many* orders-of-magnitude better speed/efficiency
→ Learning approach to physical functionalities

Physics-aware training



First demonstrations of backprop to train arbitrary physical systems *in situ*

→ Scales to high-dimensional parameter spaces
→ Trained PNN models inherently mitigate device imperfections, simulation-reality gap, and noise.

L.G. Wright*, T. Onodera*, M.M. Stein, T. Wang, D.T. Schachter, Z. Hu, P.L. McMahon,
Deep physical neural networks trained with backpropagation, *Nature* **601**, 549-555 (2022)

The hardware IS the software

- In the brain, information processing is **emergent** from physical substrate

Jeremie Laydevant

Tianyu Wang

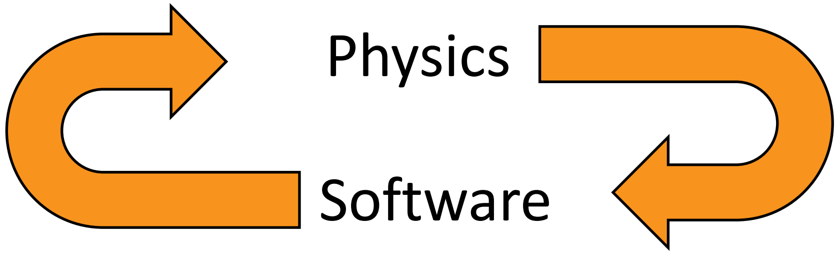
Peter McMahon

Laydevant*, Wright*, Wang & McMahon “The hardware is the software”, Neuron (2023)

The hardware IS the software

- In the brain, information processing is **emergent** from physical substrate

- Computers we develop should be the same! “Physics-first”



Jeremie Laydevant

Tianyu Wang

Peter McMahon

Laydevant*, Wright*, Wang & McMahon “The hardware is the software”, Neuron (2023)

The hardware IS the software

- In the brain, information processing is **emergent** from physical substrate
- Computers we develop should be the same! "Physics-first"
- BUT: hardware physics != physics of biology (on Earth)

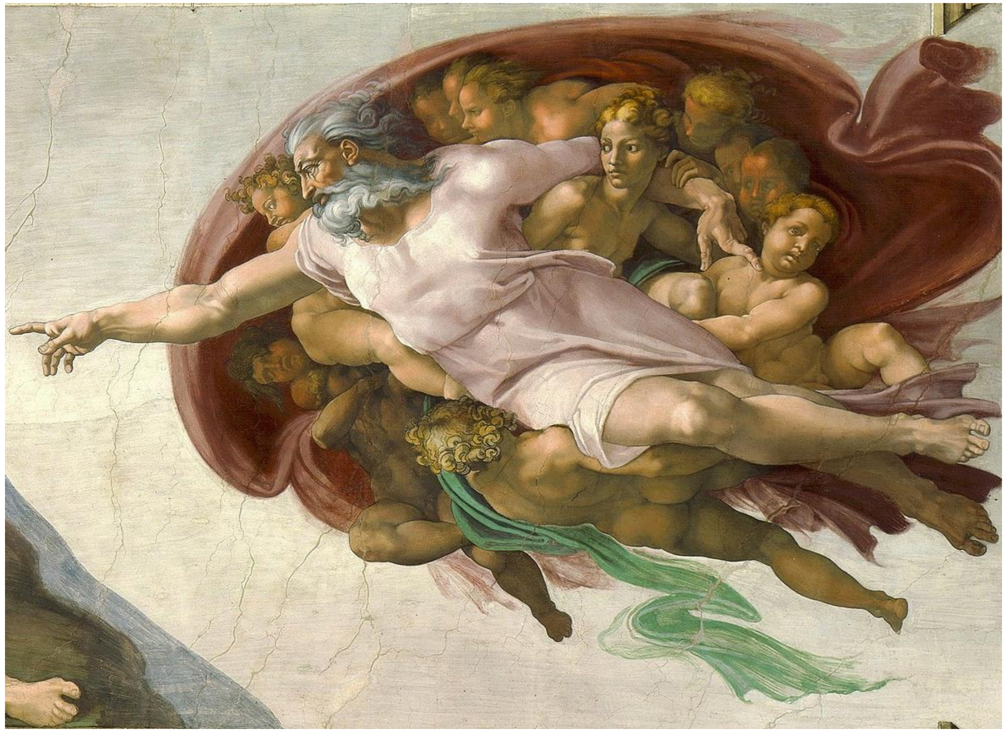
Jeremie Laydevant

Tianyu Wang

Peter McMahon

Laydevant*, Wright*, Wang & McMahon "The hardware is the software", Neuron (2023)

A new(ish) set of questions for neuromorphic computing



Jeremie Laydevant

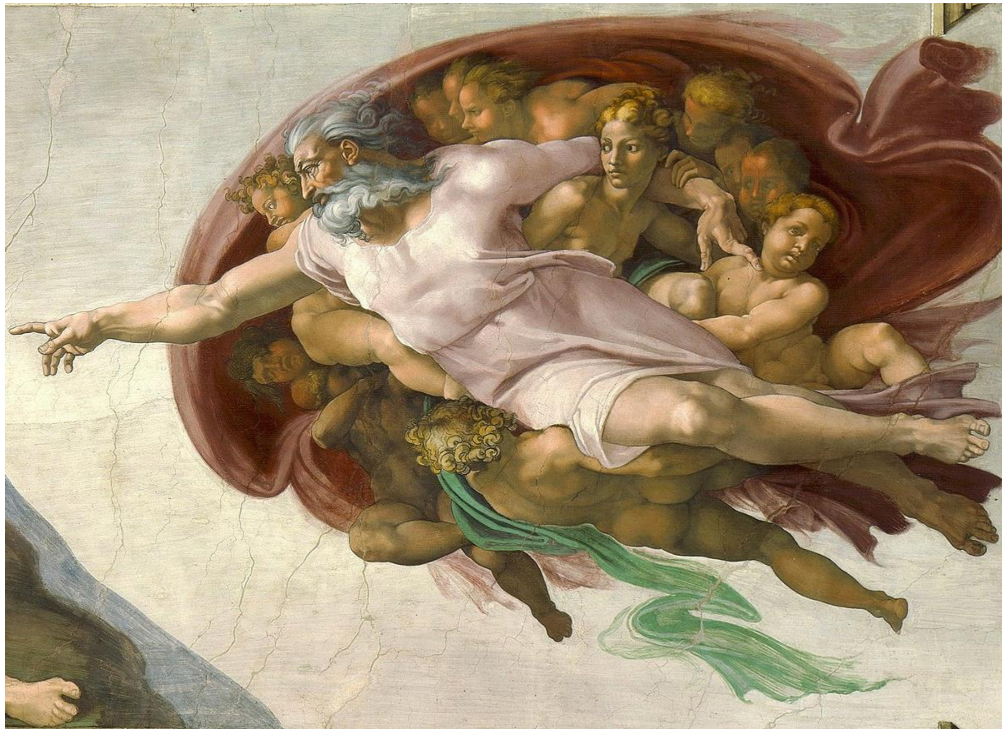
Tianyu Wang

Peter McMahon

- *Alien neuromorphics*: What would the brains and bodies of alien intelligences look like if their biology had early on incorporated "alien" elements like laser radiation, semiconductor electronics, etc.?

Laydevant*, Wright*, Wang & McMahon "The hardware is the software", Neuron (2023)

A new(ish) set of questions for neuromorphic computing



Jeremie Laydevant

Tianyu Wang

Peter McMahon

- *Alien neuromorphics*: What would the brains and bodies of alien intelligences look like if their biology had early on incorporated “alien” elements like laser radiation, semiconductor electronics, etc.?
- *Universal neuromorphics*: What are the “universal principles of intelligence” – physical features we’d expect of **all** intelligent physical systems?

Laydevant*, Wright*, Wang & McMahon “The hardware is the software”, Neuron (2023)