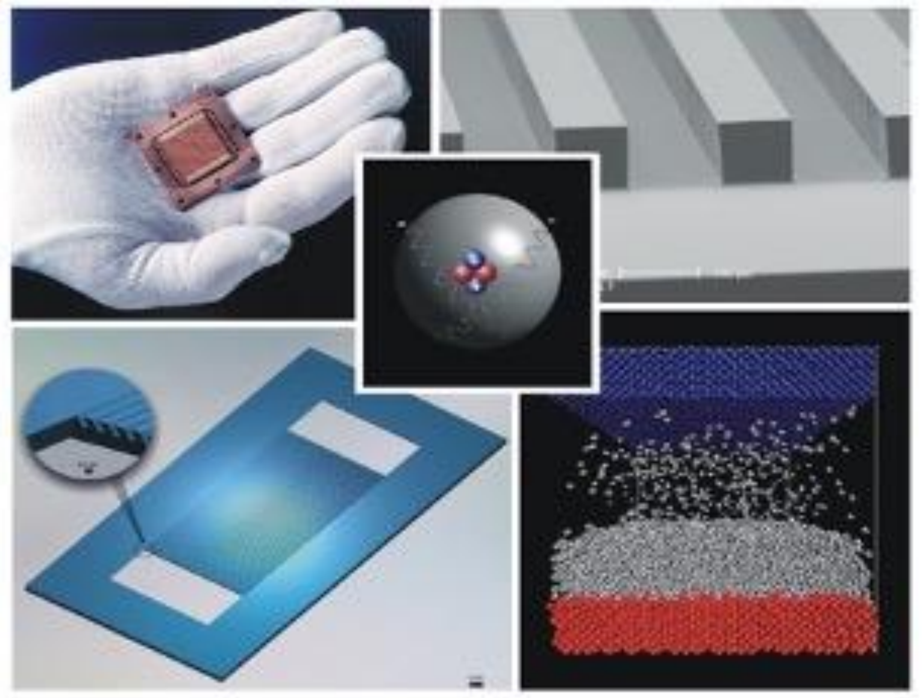
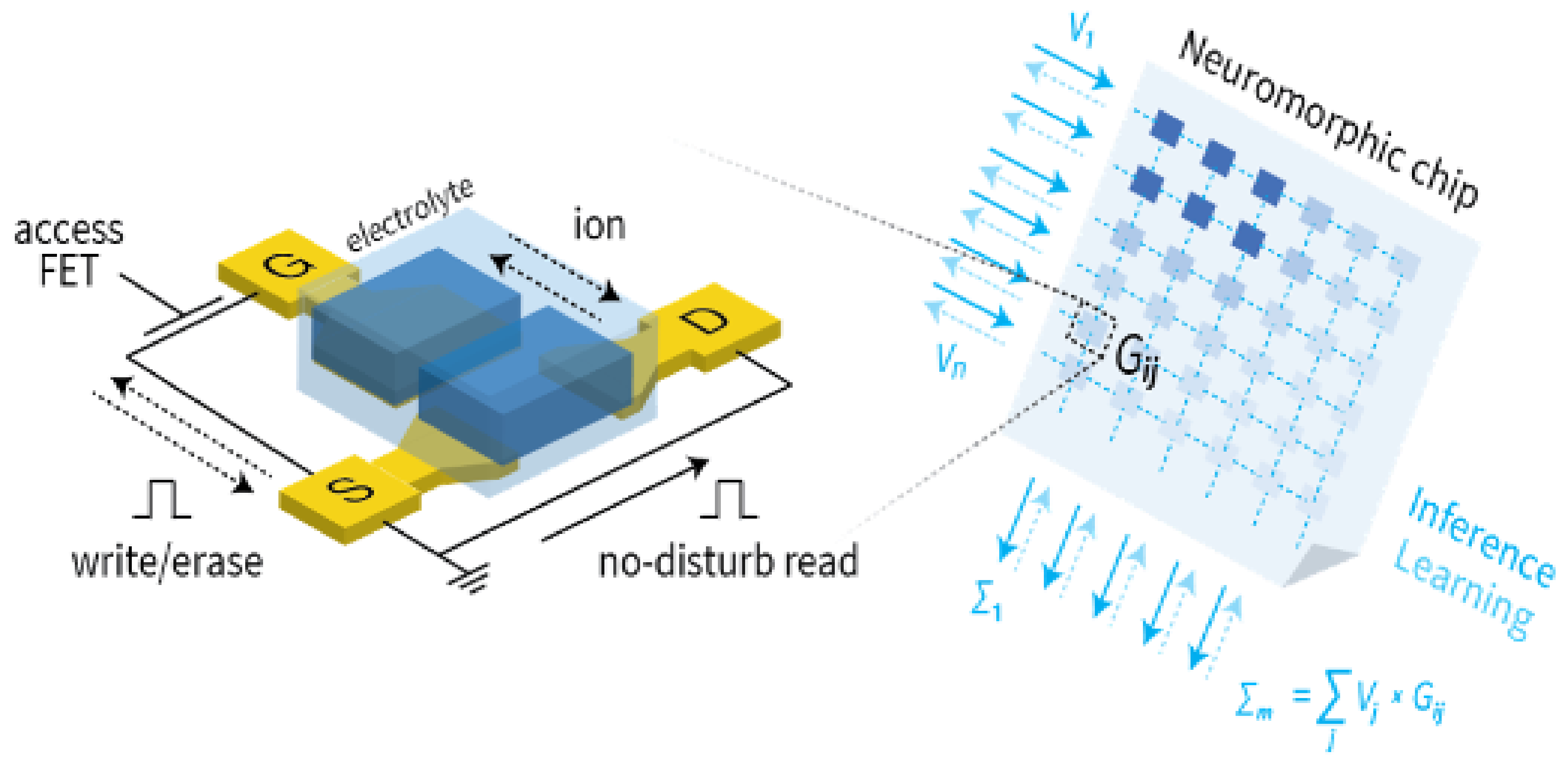
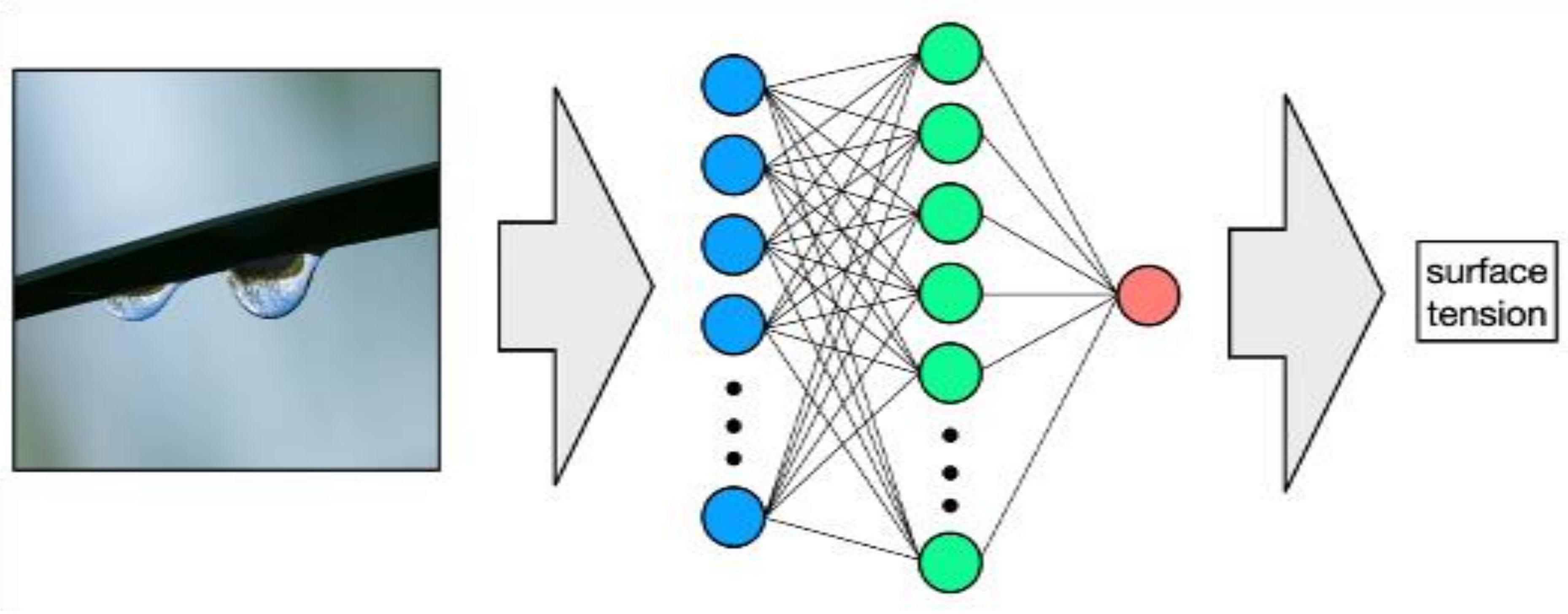
Machine learning for Multi-physics Modelling and Design (4AI000)

Kick-off meeting 24 April 2025







Course information

Lecturers:

Silvia Gaastra-Nedea (SvG)
Energy Technology

Yoeri van de Burgt (YvB)
Microsystems

Stein Stoter (SS)
Power & Flow

Ondřej Rokoš (OR)
Mechanics of Materials

Course information

Applying ML techniques to understand physics, predict properties and/or generate models/devices/materials

- *Intermolecular interactions, heat transfer and macroscopic properties*
  SvG: generate ML based model

- *Micro-manufacturing techniques*
  YvB: Neuromorphic Computing: Hardware for efficient A.I.

- *Machine learning for scale interaction*
  SS: learning closure relations high-fidelity simulation data.

- *Inverse design of magnetoactive mechanical metamaterials*
  OR: generative design, first-principles computational modelling

Course information

Course description

- To apply knowledge on data processing and AI algorithms to multi-physics modelling and design problems in the context of mechanical Engineering systems and applications
- Students will experience in a CBL format combining multidisciplinary and AI knowledge to solve engineering problems, develop machine learning alg, advanced data processing alg.
- Projects are connected to current research of the lecturers, offering high relevance of the domain-knowledge

Course information

Planning

- April 24 (today): Introduce topics (slides and project descriptions on canvas)
- Before April 27 : choose preferred assignments (on canvas)
- Tuesday, April 29th: 13.30h Q&A with Challenge Supervisor
- Mid-term presentation: May 15
- Final Presentation: June 17 at 13:30 am

- Final oral examination: to be planned individually with the lecturers

Working format : groups of X students (X is project dependent)

Course information

Mid-term presentation:

- week 4 (May 15): 10 min presentation per group (format of the presentation available on canvas soon).

- Students can ask feedback and receive questions from each other and from the lecturers. Students should not only provide feedback to each other but should also understand the challenges in the other projects.

Course information

Final Examination:

- week 9 (June 17): Presentation + Short paper + Oral exam
  - Short paper: max 6 pages containing (research questions, approach, results, analysis/reflection process, etc.) Format of the paper on canvas soon.
- Oral examination (15-20 min)
  - Reflection on your work
  - Questions on the short paper
  - Questions on the other projects regarding the topics, goals, challenges and AI solutions

Course information

- Final grade:
    - 50% short paper
    - 50% final oral examination
- All component of the final mark should be > 5.5!
- Re-examination : to be planned in the summer in agreement with the lecturer
    - improved report
    - oral examination

Course information

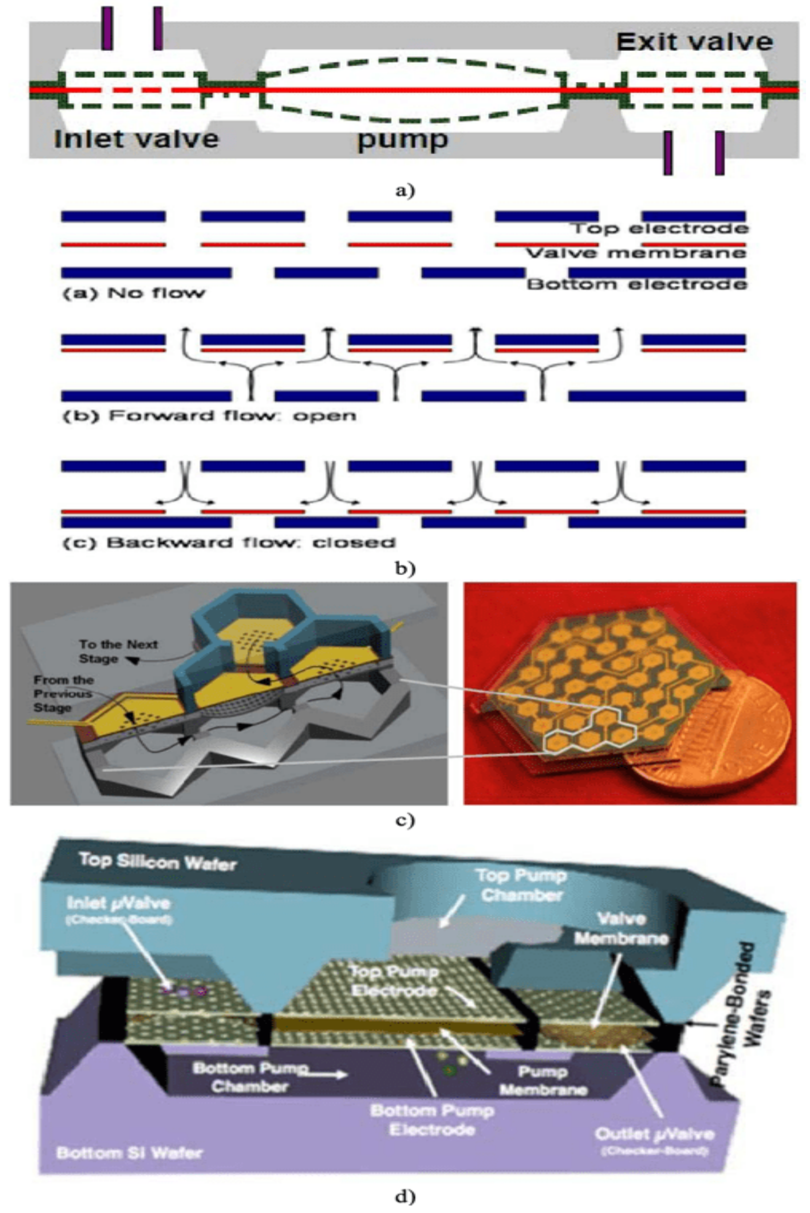
This is a **challenge-based** course:
you are the responsible and the driving force!

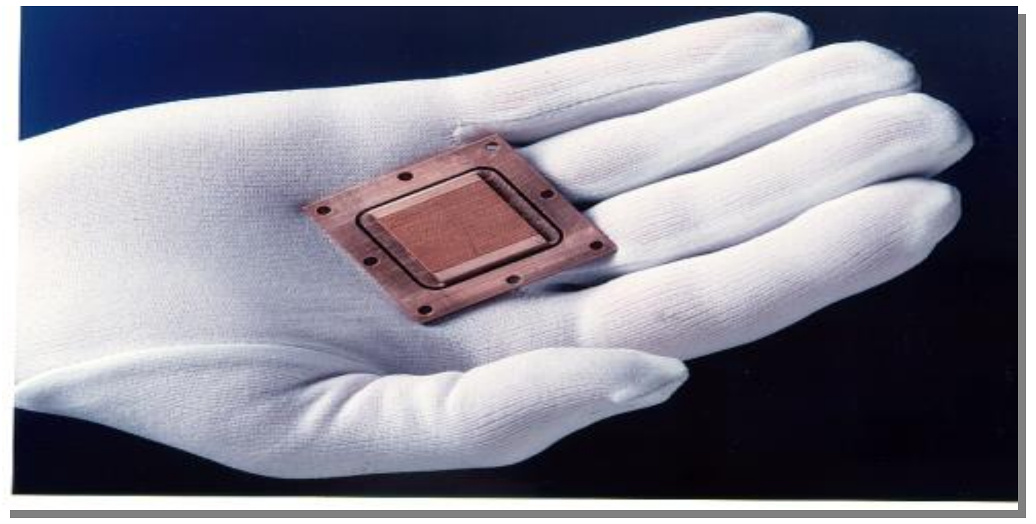
Challenge 1

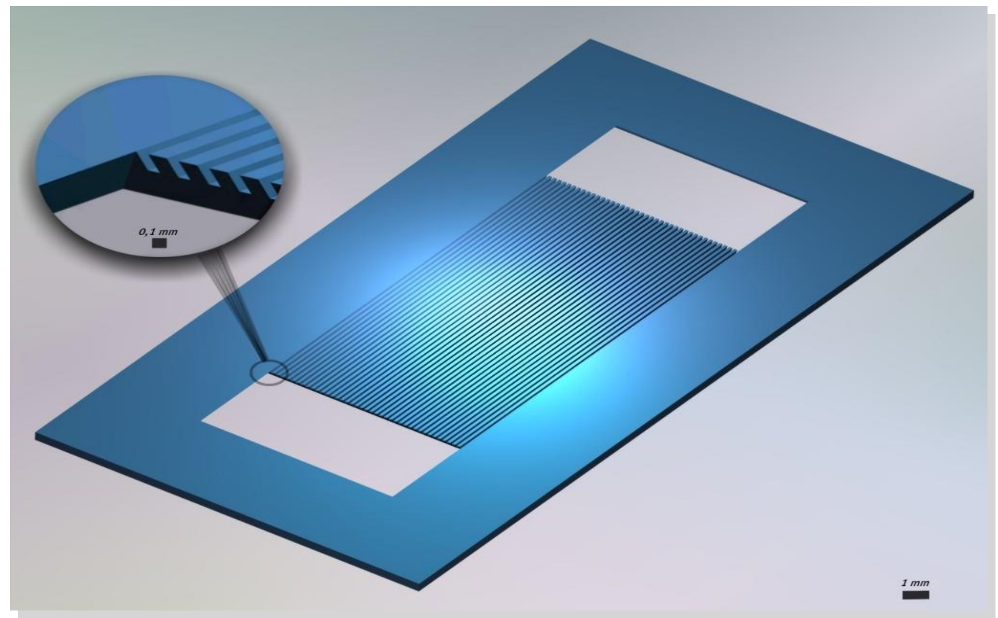
Intermolecular interactions, heat transfer and macroscopic properties

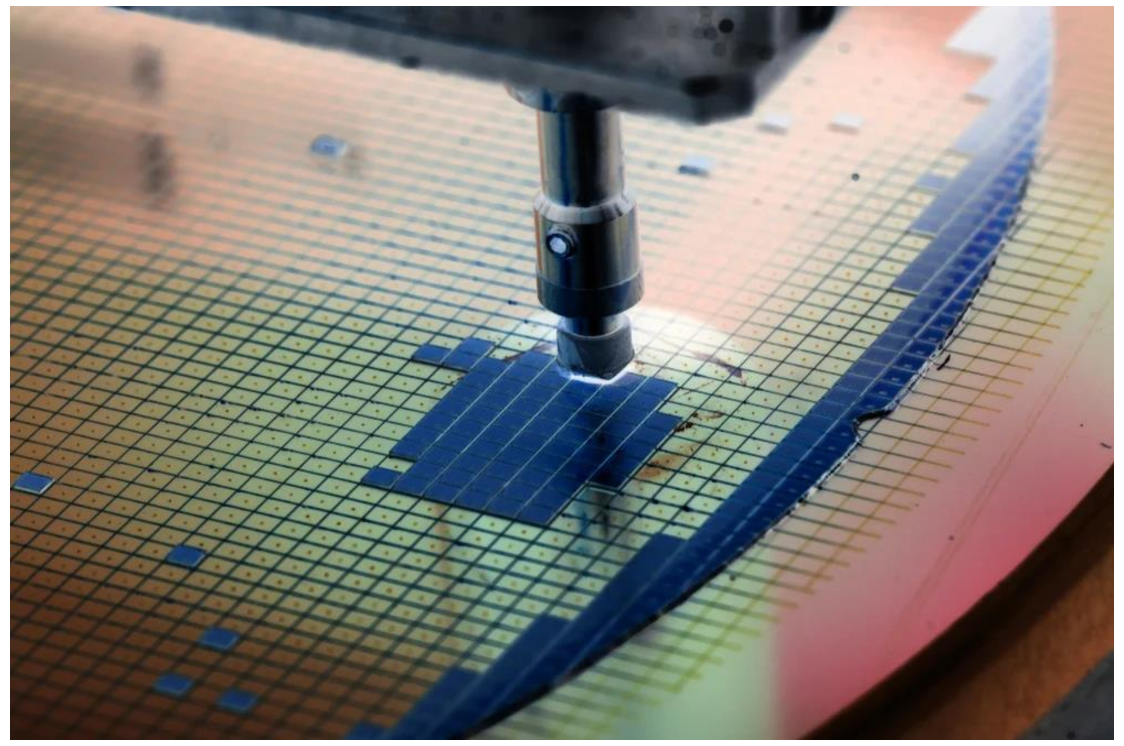
Machine learning based models for rarefied gas flow: Physics of interfaces.

Application fields:



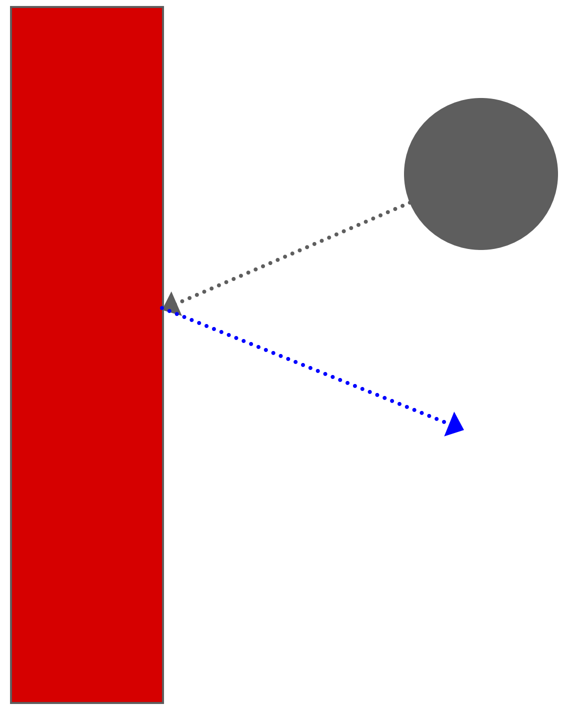






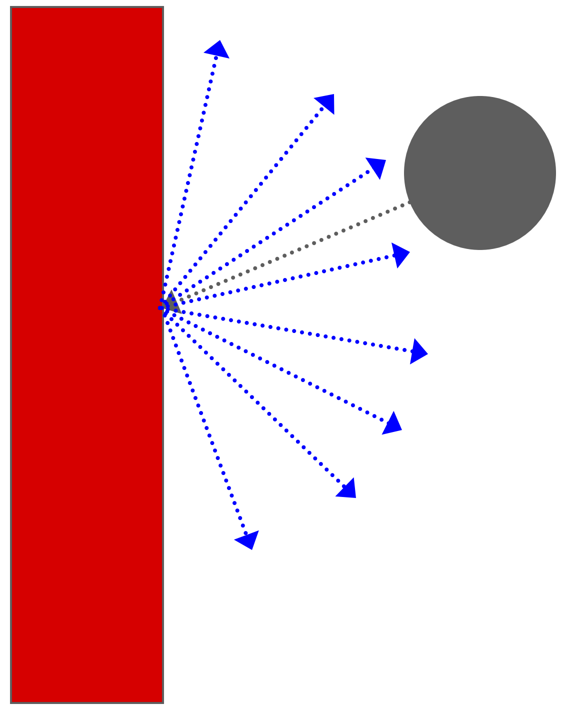
Modelling fluid-wall interactions

Specular wall / reflecting wall



No energy exchange with wall

Thermal wall / diffusive wall



Energy exchange from wall to particle

Diffusive-specular wall

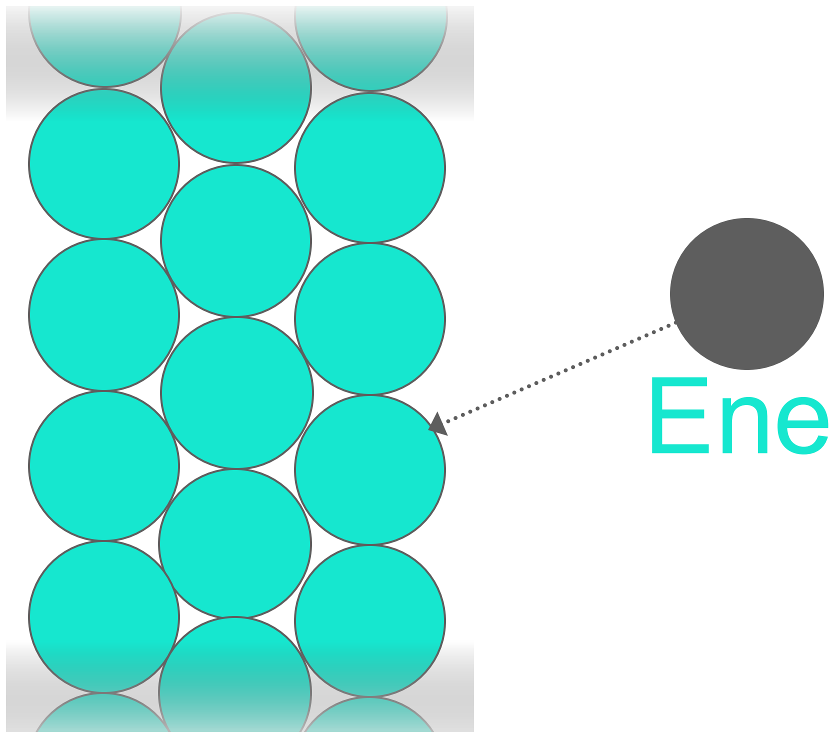
=

α Diffusive wall

+

(1-α) Specular wall

Explicit MD wall



Energy exchange between
wall and particle

Challenge

- Accommodation Coefficient (α_K) **is a measure to quantify the momentum or energy exchange** at the gas-solid interface

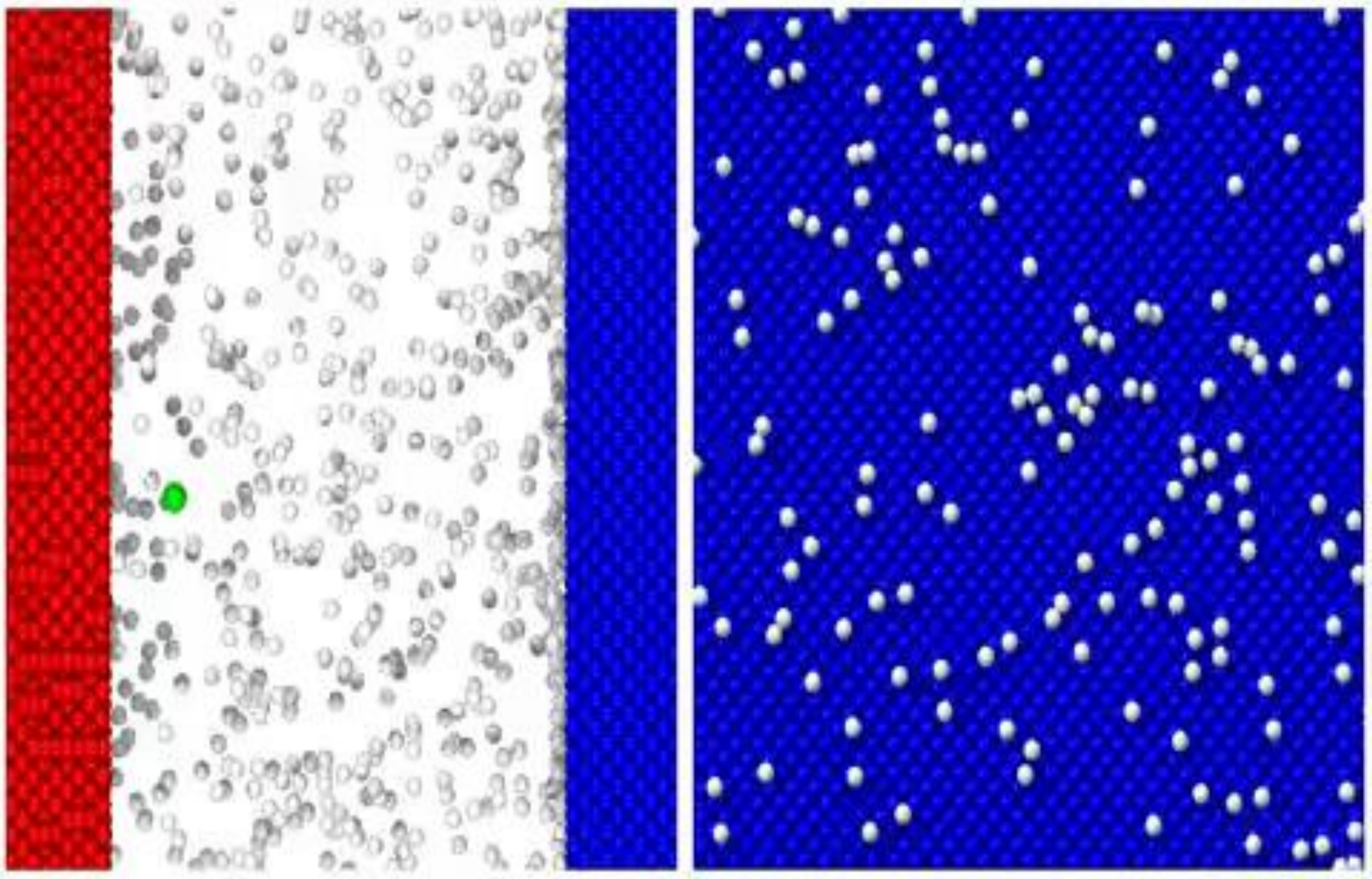
  - **Experimental determination** of α_K is a **very challenging task**

↓

- α_K depends on many factors:

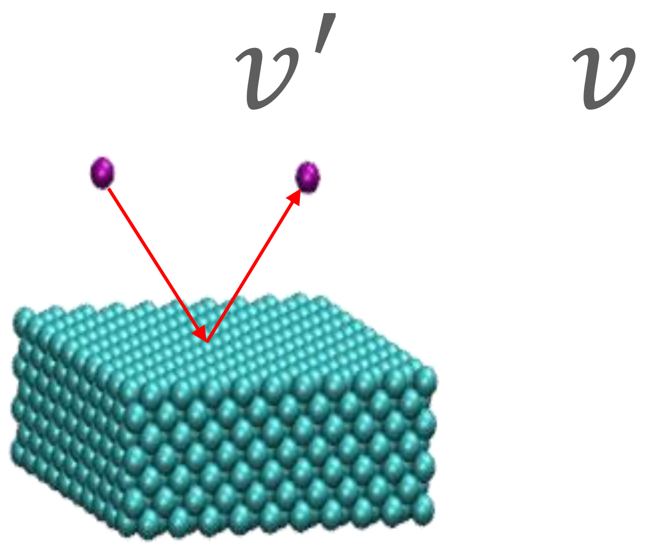
i. Temperature (gas & surface)
ii. Surface condition (cleanness & roughness)
iii. Gas-solid mass ratio
iv. Elastic module of the solid

MD scattering for computing thermal accommodation coefficients



$$\alpha_1 = \frac{T_{in} - T_{out}}{T_{in} - T_S}$$

Goal: Deriving a gas-wall interaction model using Machine Learning



Common Approach

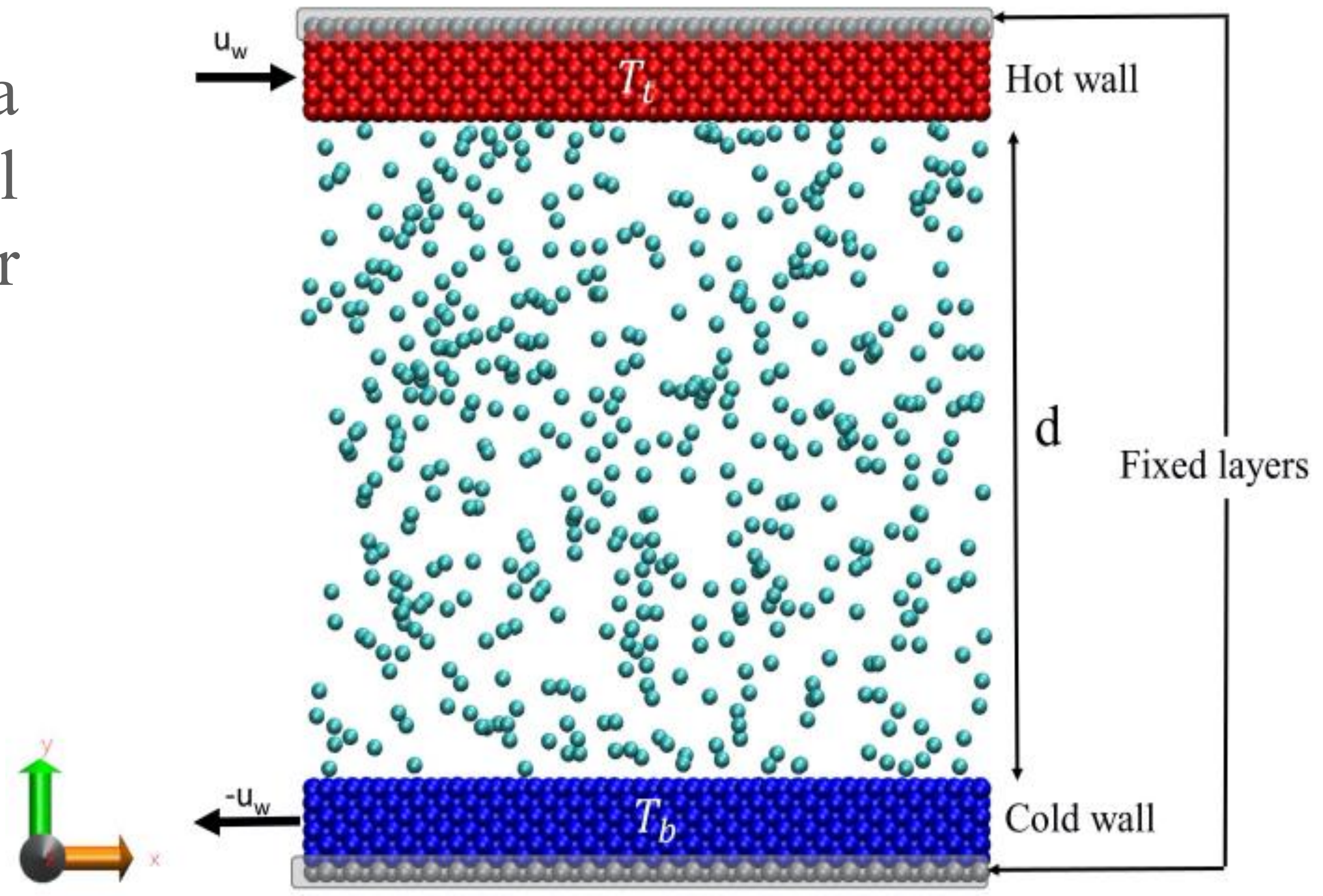
Gathering collision data → Computing different ACs → Scattering kernels

New Approach

Gathering collision data → Scattering kernels

Assignment

Develop Machine Learning (ML) models to construct a statistical gas-surface scattering model based on the collisional data obtained from Molecular Dynamics (MD) simulations for hydrogen-nickel interactions in a micro/nano-system.



Goal: To study the isothermal and non-isothermal Couette flow of a diatomic gas (Hydrogen) confined between two parallel infinite Nickel walls.

Structure:

- Part A: ML for isothermal/non-isothermal H2-Ni interactions
- Part B :ML model for isothermal-non-isothermal H2-Ni with a Couette flow included
- Part C: ML model for predicting the atomic angular velocity distributions and translational/rotational energy accommodation coefficients.
- Part D: Compare ML models on the ACC predictions.

Q&A: Next: Thursday, May 1, 10.30h. -> scripts, data, examples and documentation

Challenge 2

Neuromorphic Computing: Hardware for efficient A.I.

Neural networks and deep learning

Are ubiquitous in daily life…

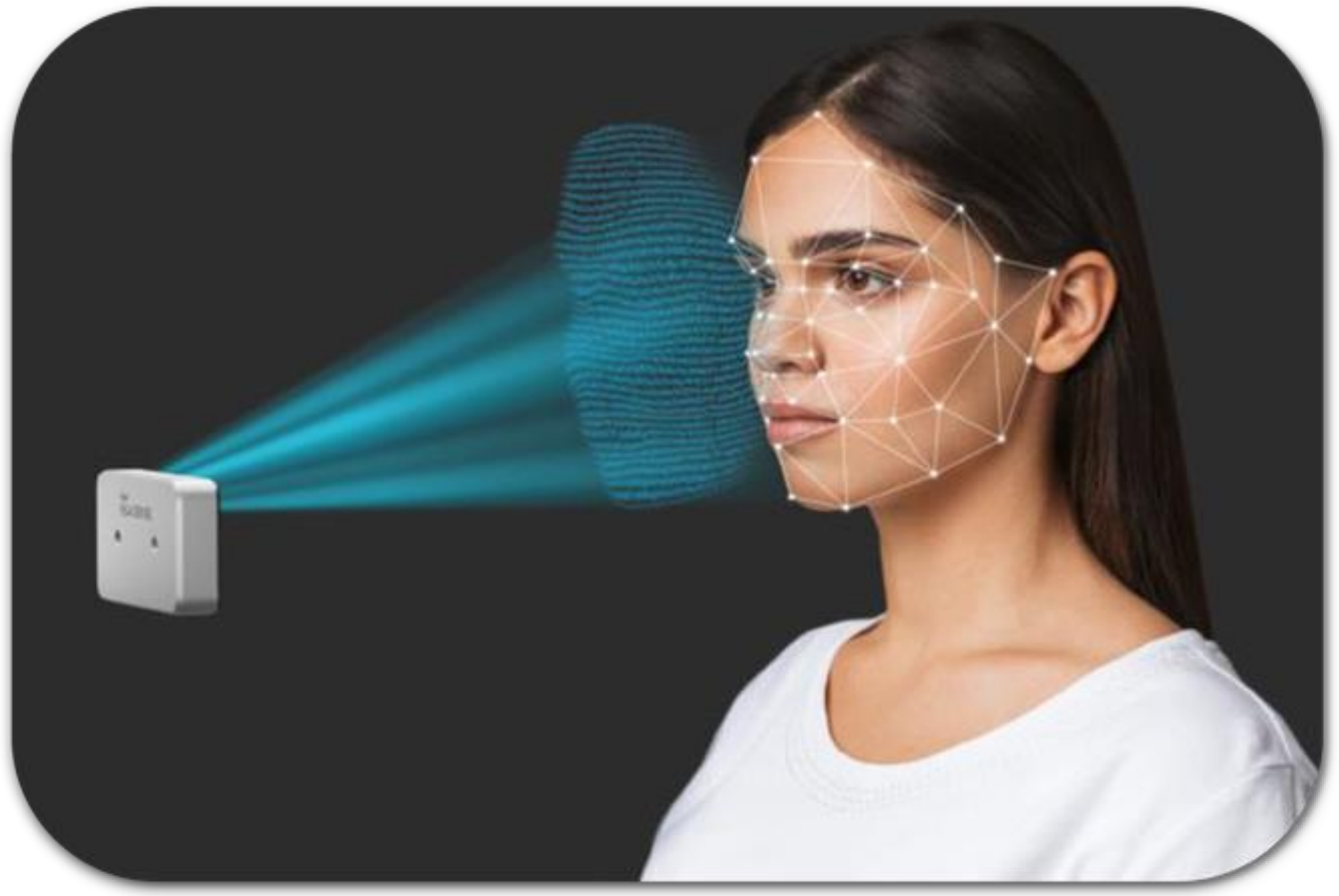
Medical Diagnosis

Generative AI | OpenAI

Drug discovery

Point of care

Autonomous driving | Waymo



Facial recognition

Local fast processing and learning

Neural networks and deep learning

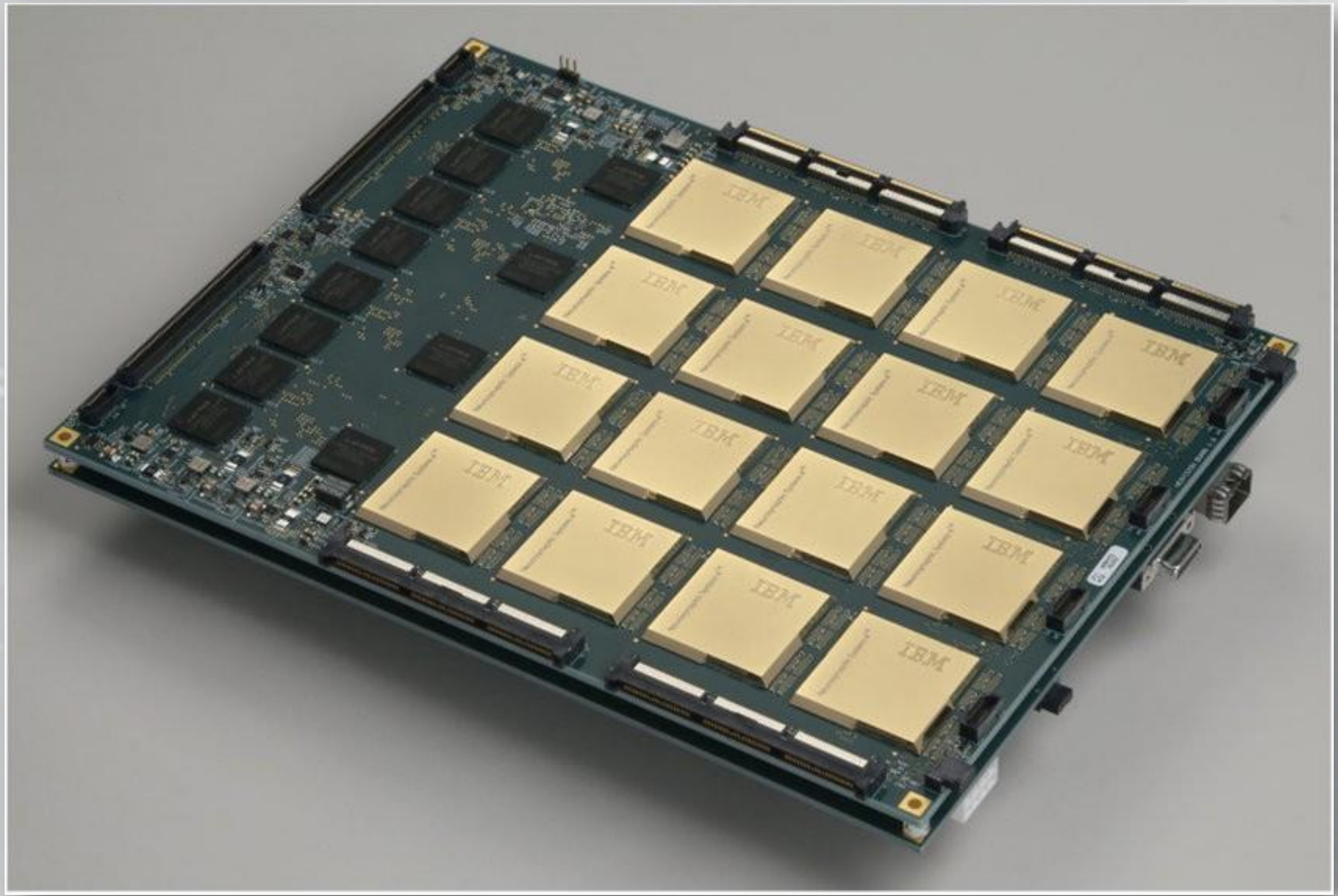
..but require **huge computation** resources and energy

VS

While our **brain** can do most of these tasks very efficiently

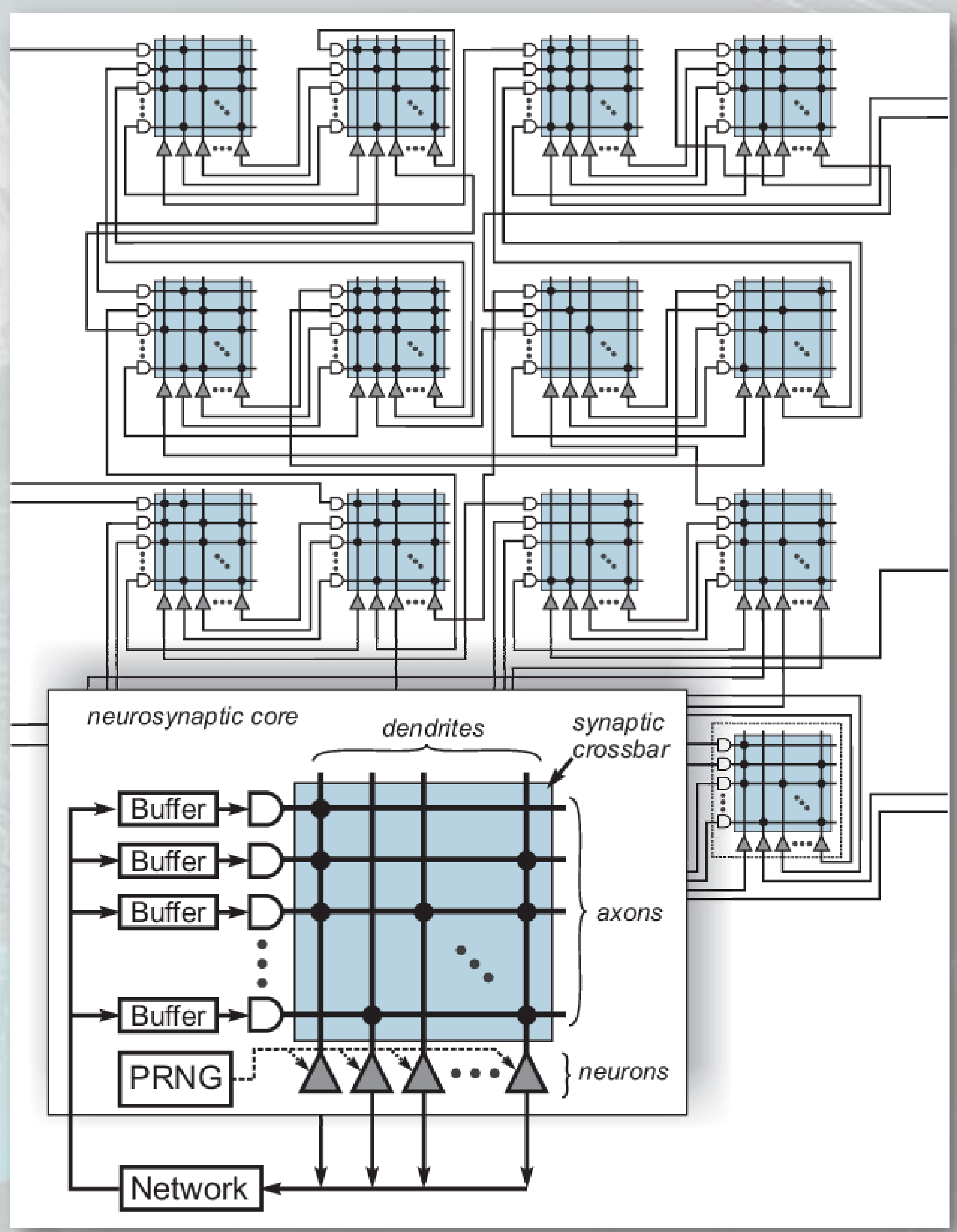
Neuromorphic Computing

Is inspired by the brain to efficiently run A.I. tasks in dedicated hardware



IBM TrueNorth

$$y_j^l = \sum_{i=1}^{n} x_i^{l-1} w_{ij}$$



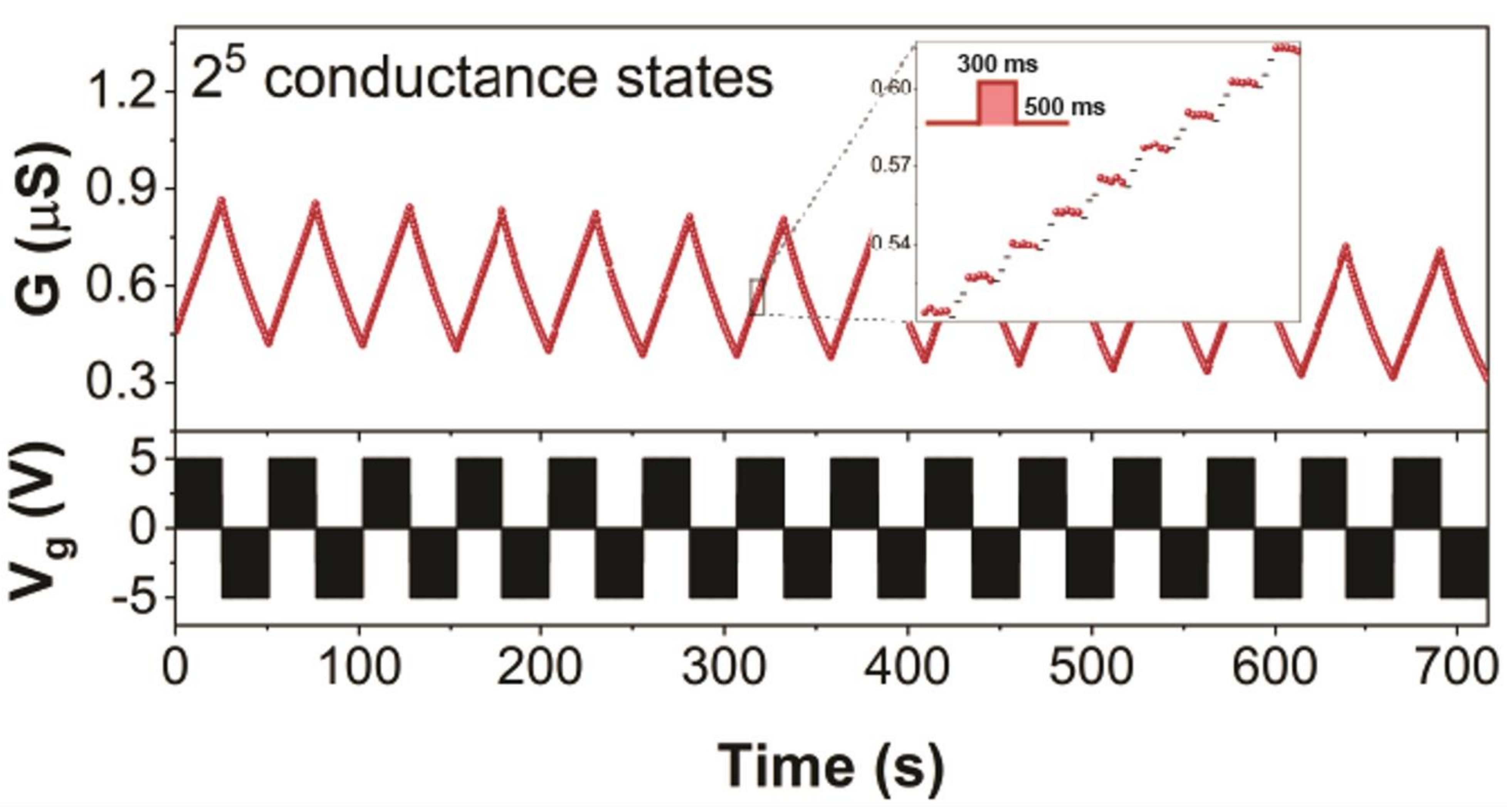
Neuromorphic Computing

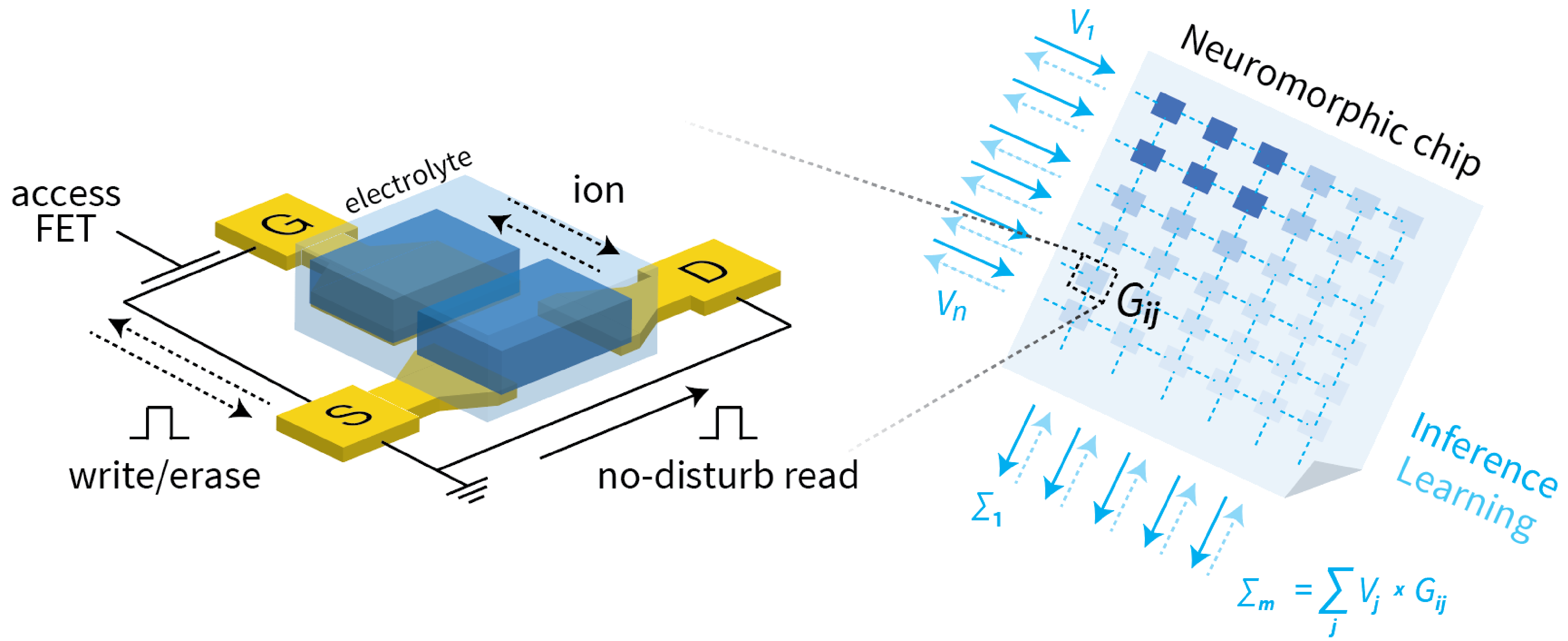
What are the limitations and challenges when designing **hardware** for A.I applications?

General idea:

1. Fabricate and characterise a single synaptic device
2. Using these (measured) properties to train models to perform a benchmark problem

So how would a hardware-based neural network behave with your devices



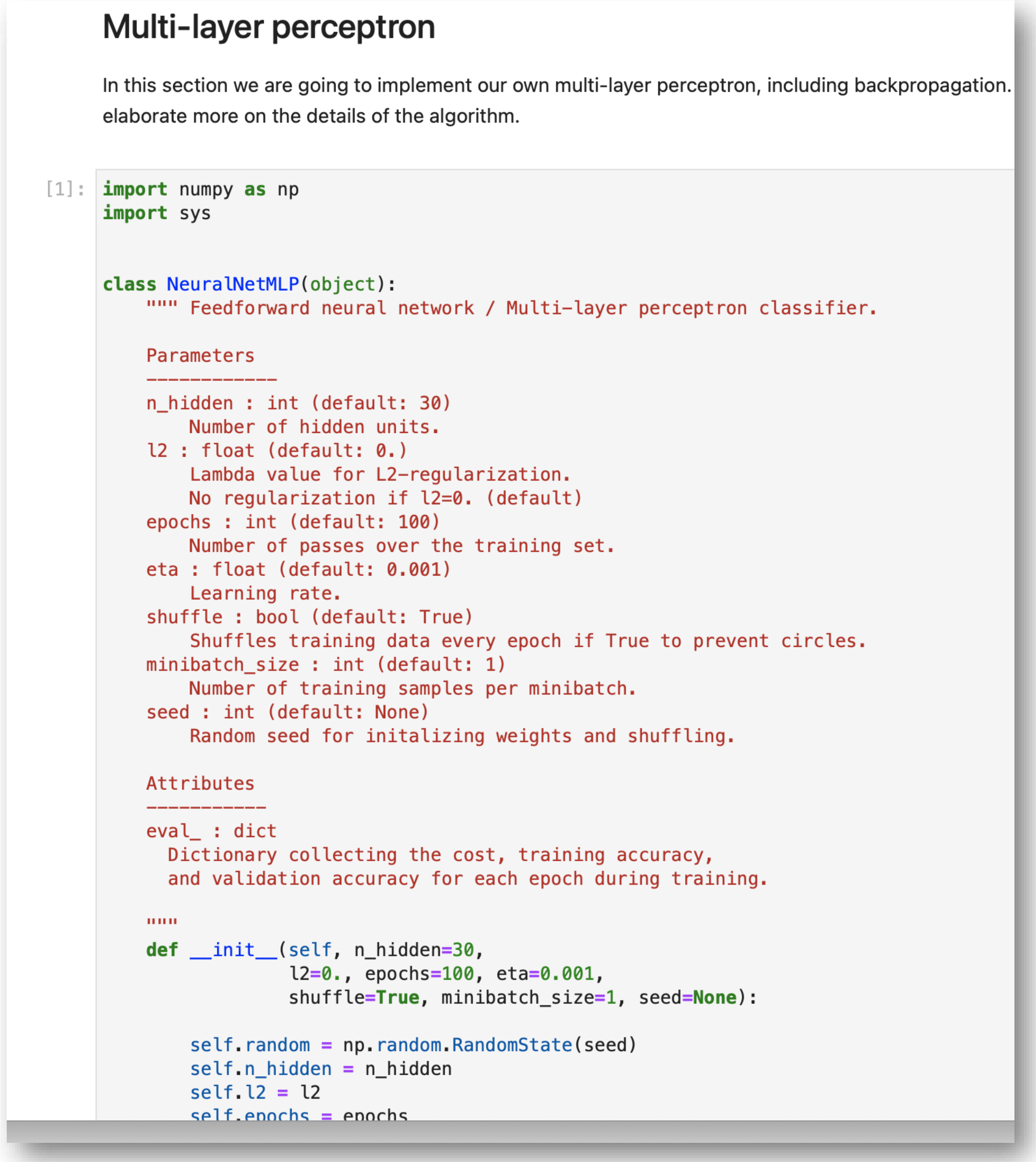


Devices and simulations

You will **fabricate and characterise** your own neuromorphic devices in our lab

You are given a basic **script** to train models which you can adapt to include your device characteristics

Microfab lab @ TU/e



Multi-layer perceptron

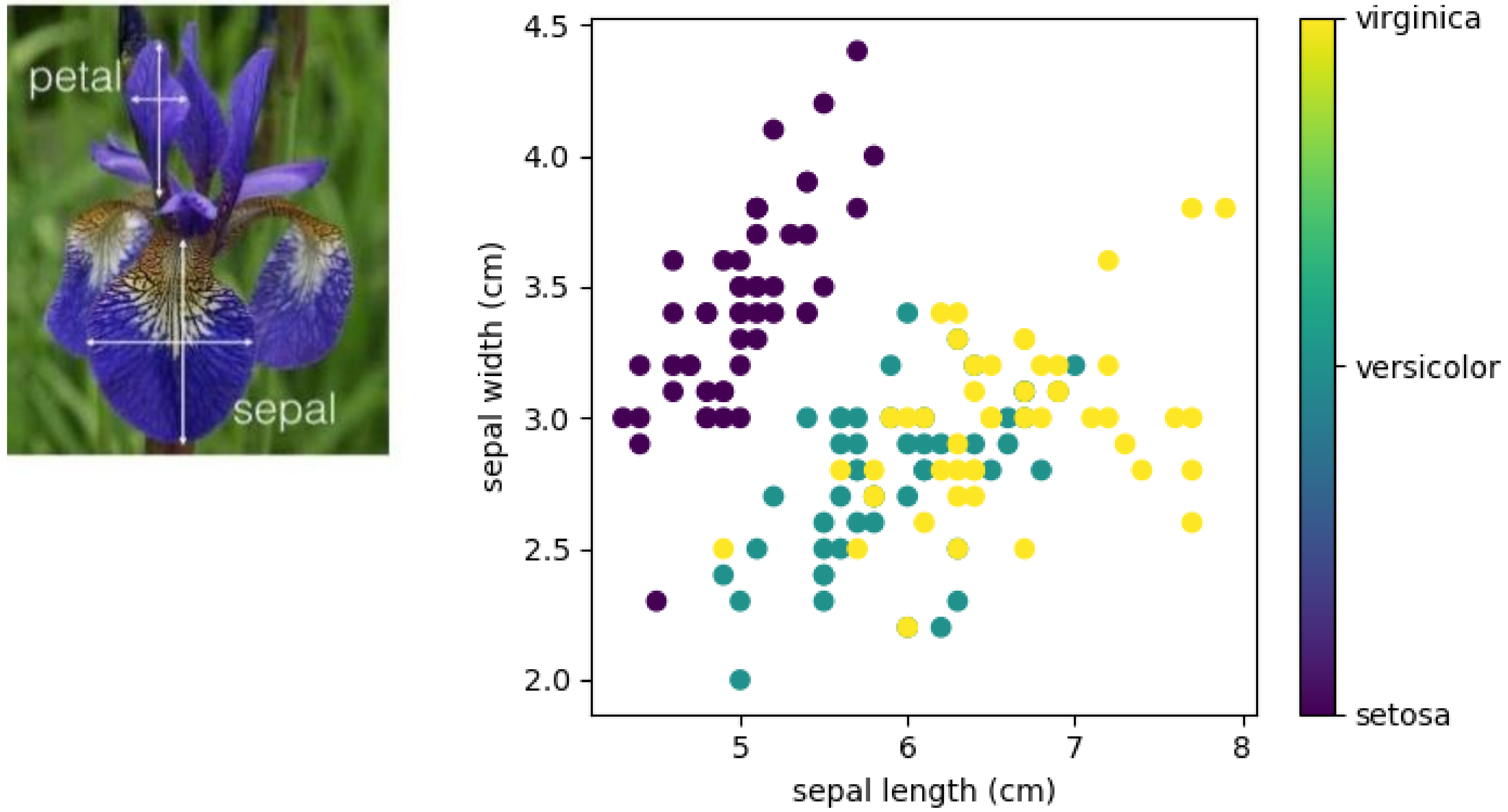
In this section we are going to implement our own multi-layer perceptron, including backpropagation.
elaborate more on the details of the algorithm.

[1]:
```
import numpy as np
import sys


class NeuralNetMLP(object):
    """ Feedforward neural network / Multi-layer perceptron classifier.

    Parameters
    ------------
    n_hidden : int (default: 30)
        Number of hidden units.
    l2 : float (default: 0.)
        Lambda value for L2-regularization.
        No regularization if l2=0. (default)
    epochs : int (default: 100)
        Number of passes over the training set.
    eta : float (default: 0.001)
        Learning rate.
    shuffle : bool (default: True)
        Shuffles training data every epoch if True to prevent circles.
    minibatch_size : int (default: 1)
        Number of training samples per minibatch.
    seed : int (default: None)
        Random seed for initalizing weights and shuffling.

    Attributes
    -----------
    eval_ : dict
      Dictionary collecting the cost, training accuracy,
      and validation accuracy for each epoch during training.

    """
    def __init__(self, n_hidden=30,
                 l2=0., epochs=100, eta=0.001,
                 shuffle=True, minibatch_size=1, seed=None):

        self.random = np.random.RandomState(seed)
        self.n_hidden = n_hidden
        self.l2 = l2
        self.epochs = epochs
```

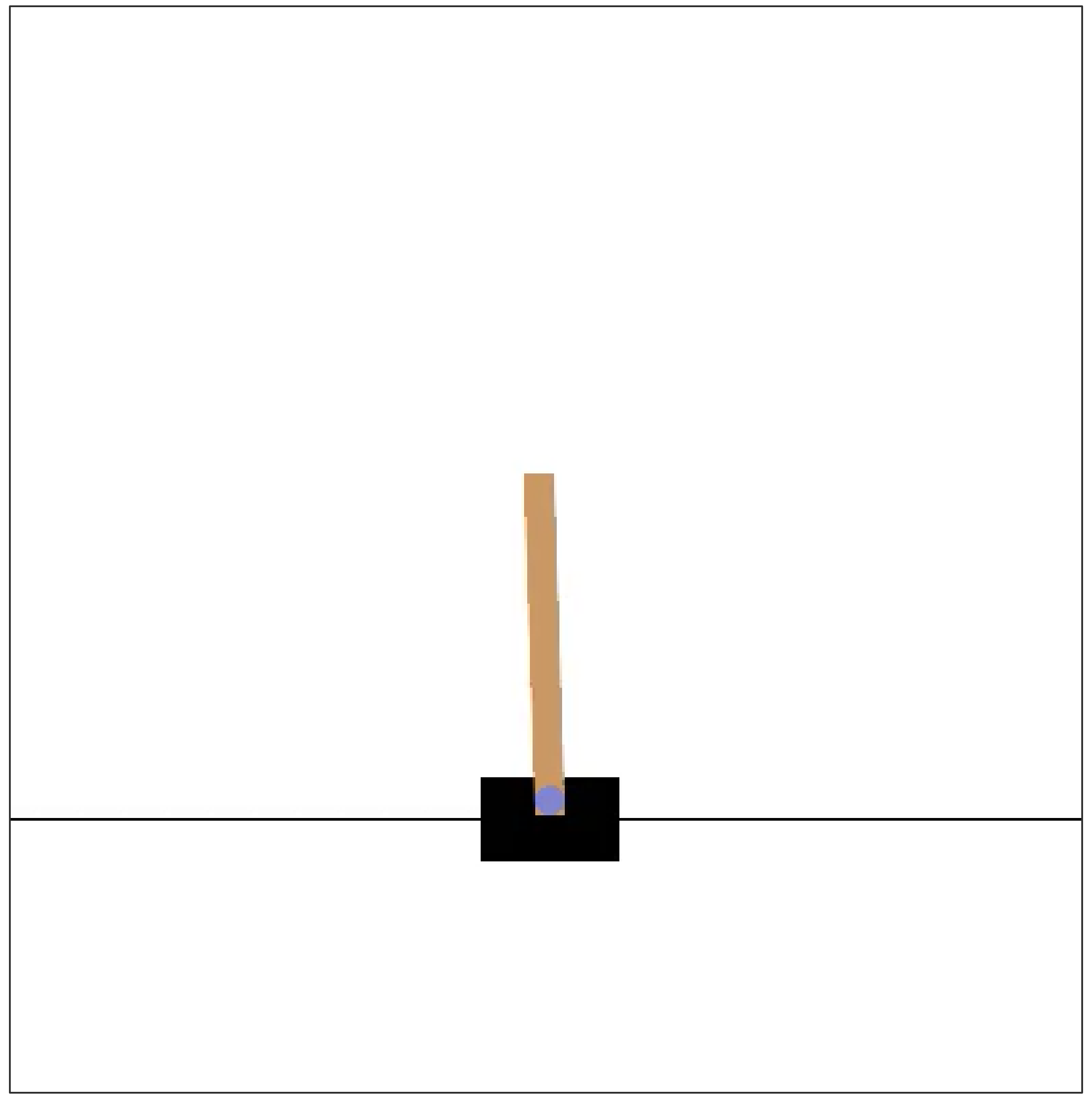
Neuromorphic Computing

Challenge based project. The goal is to evaluate the performance of your hardware-based neural network circuit:

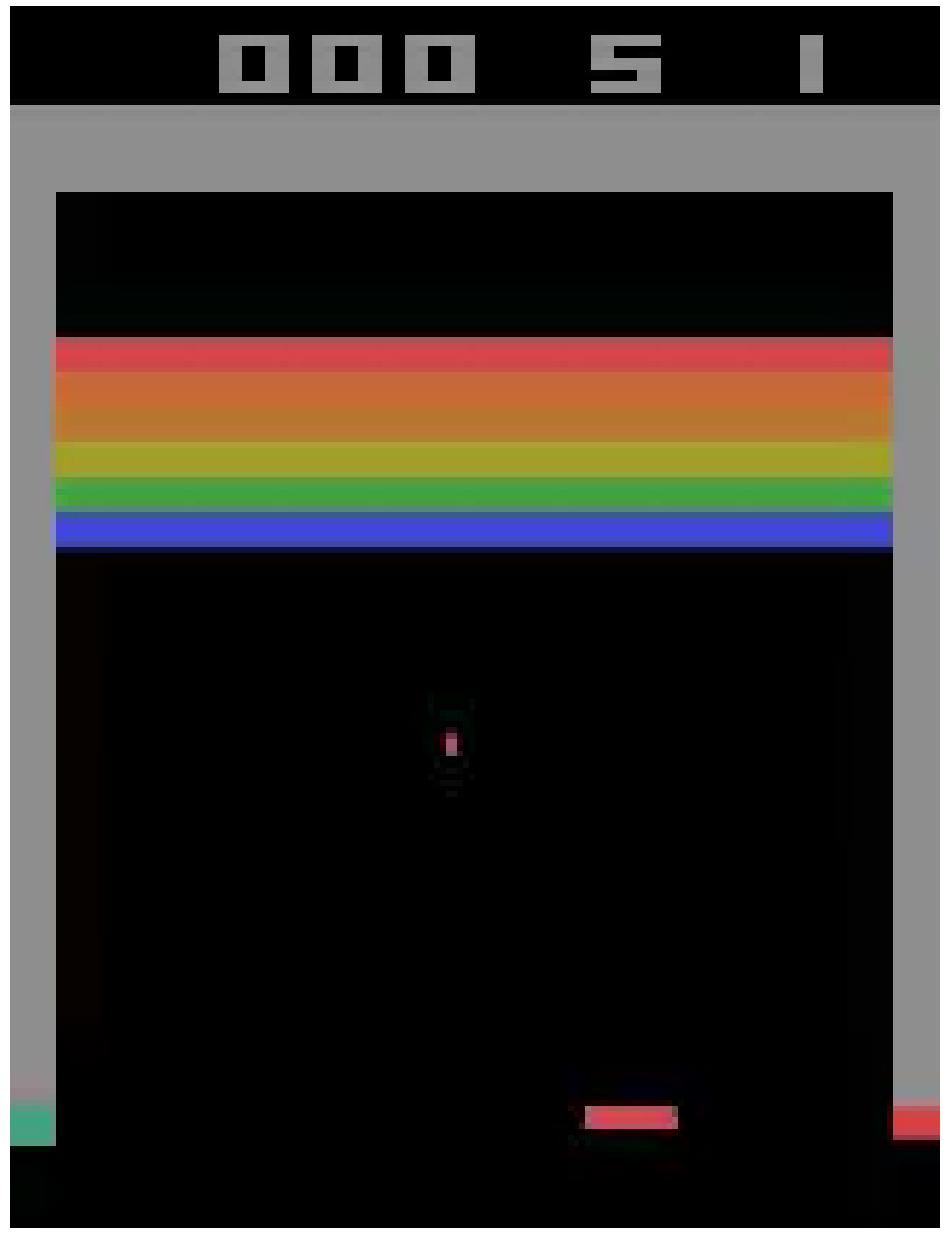
- *How does the retention time effect accuracy?*
- *How does the linearity affect speed of convergence / stable movement?*
- *What about noise? Clipping of the weights?*



Iris Dataset – Scipy Lecture Notes



CartPole - v1

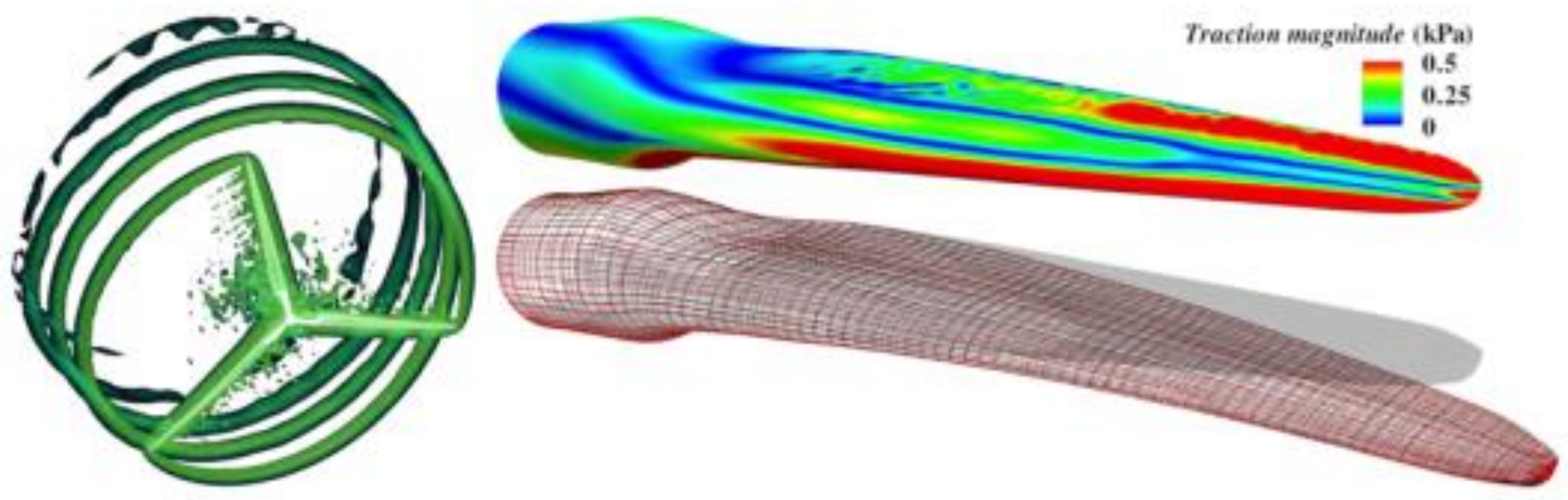


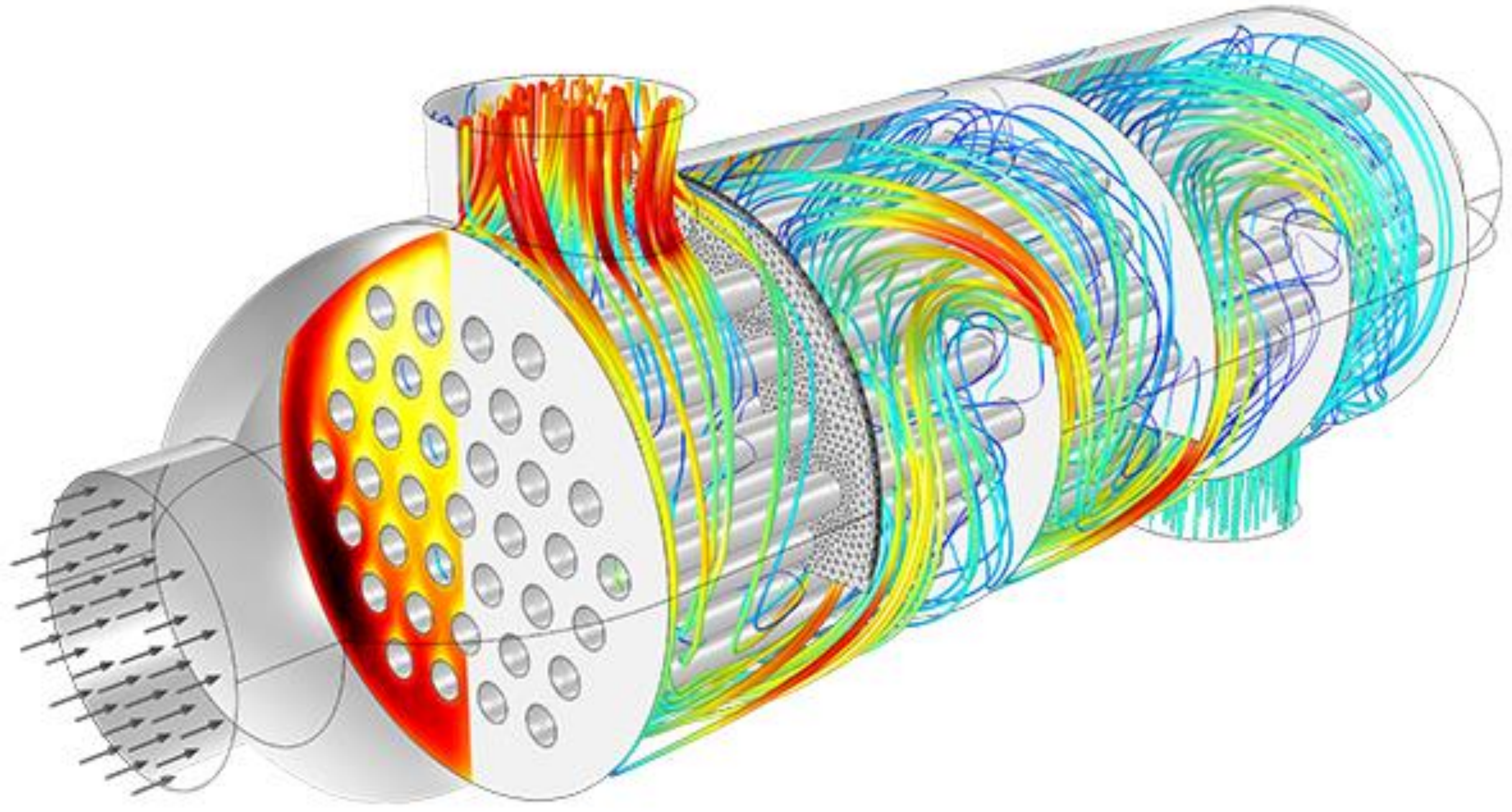
Breakout – v4

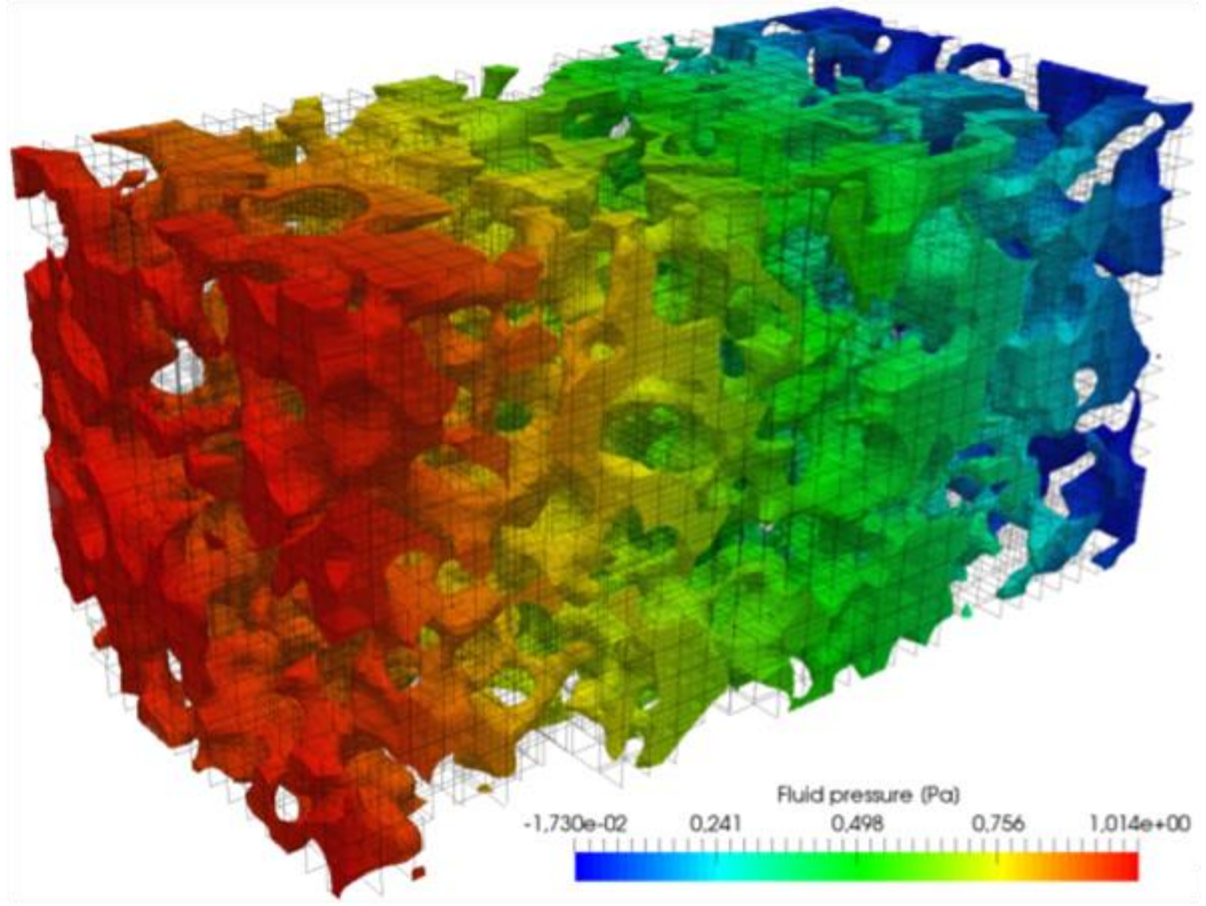
Challenge 3

Machine learning for scale interaction

Neural networks vs FEM/FVM/FDM







We love our simulation tools!

Physical model

- Physical parameters
- Geometric parameters
- Boundary conditions
- Initial conditions

Simulation tool → Solution

Satisfies physical laws
Expensive?
Input vs reality?

Machine learning → Solution

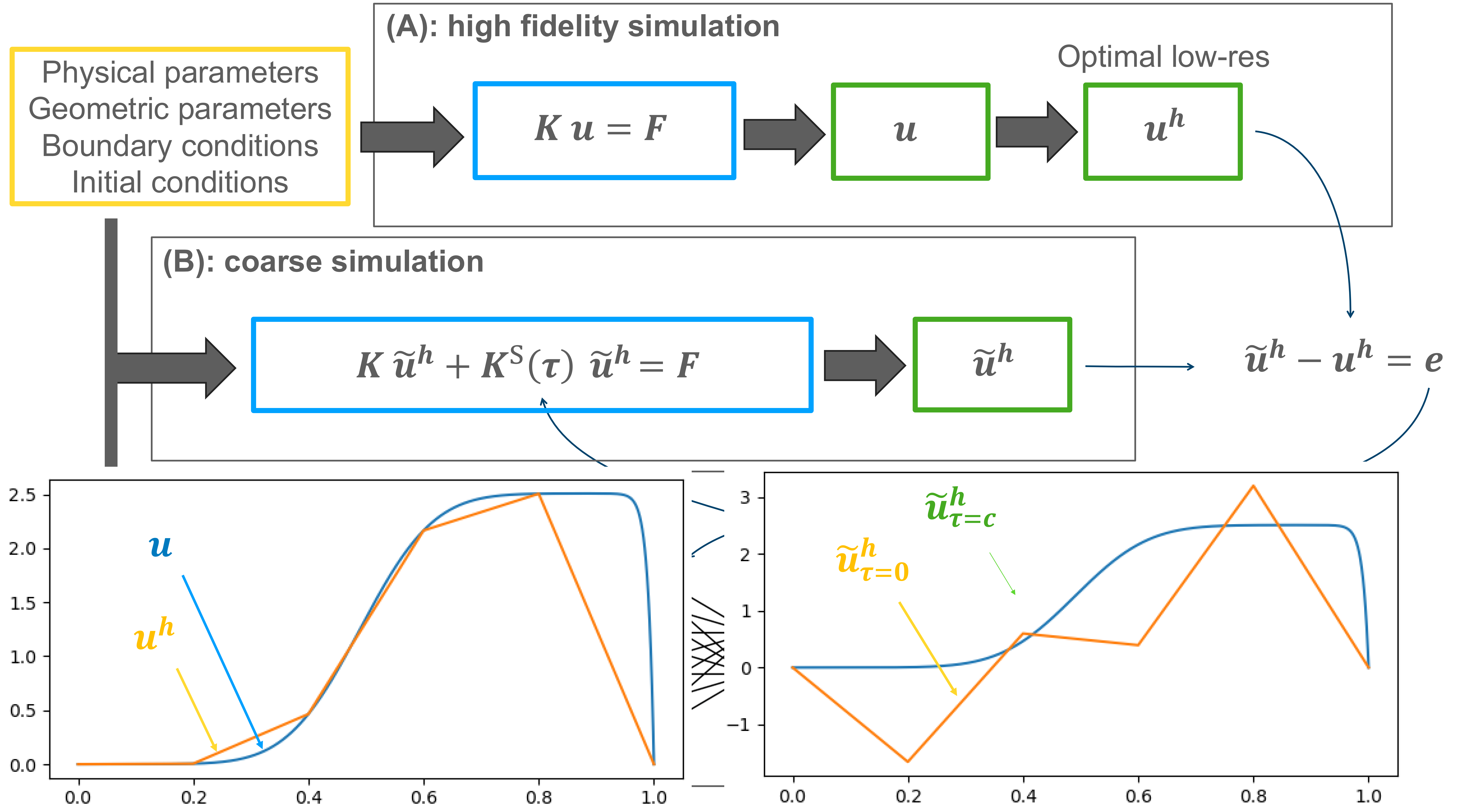
Satisfies physical laws
Cheap
Trained on reality

Does not satisfy physical laws
Cheap
Trained on reality

Background: turbulent flow

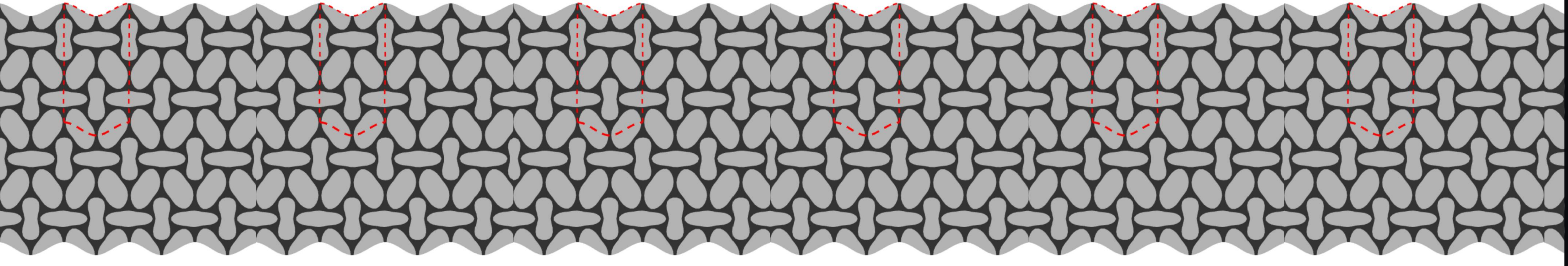
Vorticity magnitude – *Zhu, Stoter, et al. 2020*

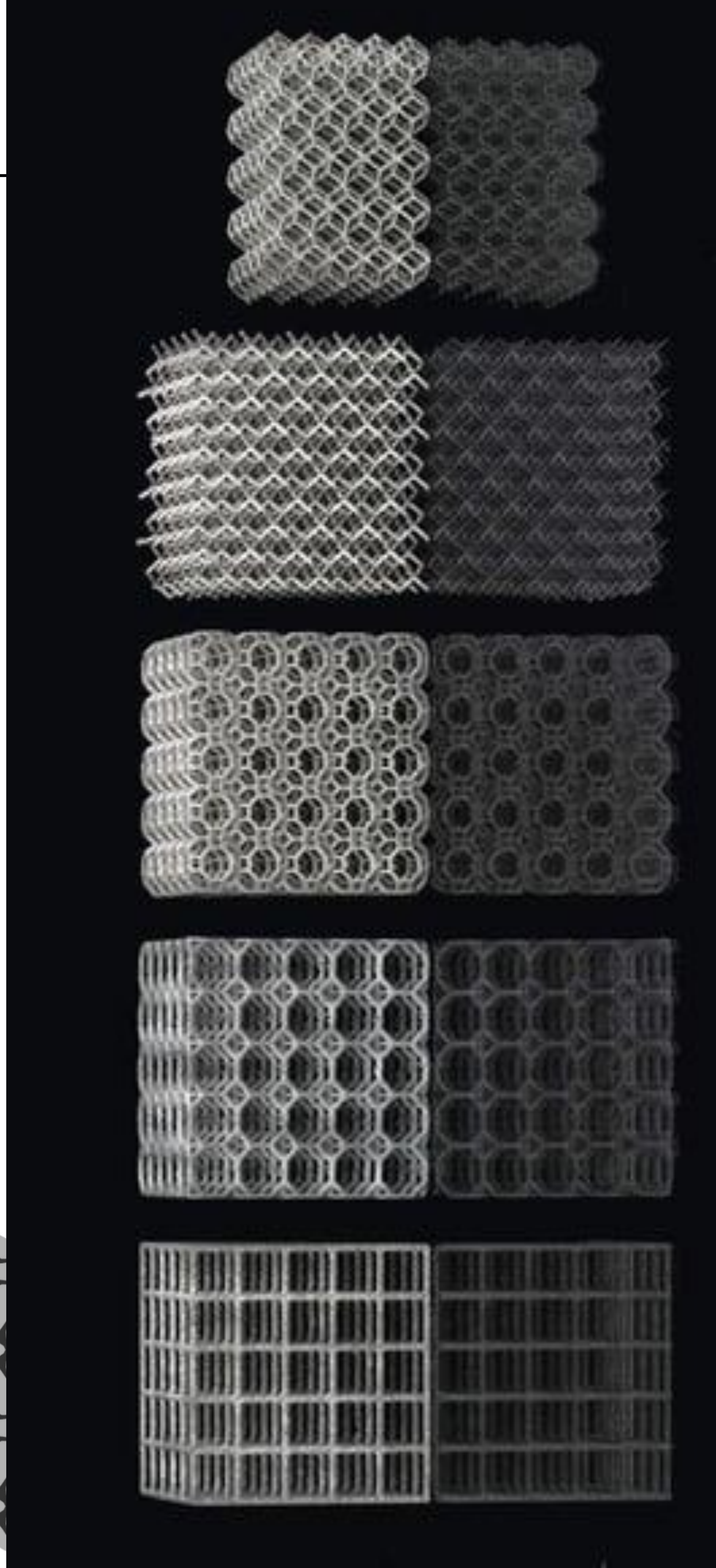
Neural networks for 'modeling' scale interaction in flows



Challenge 4

Inverse Design of Magnetoactive Mechanical Metamaterials



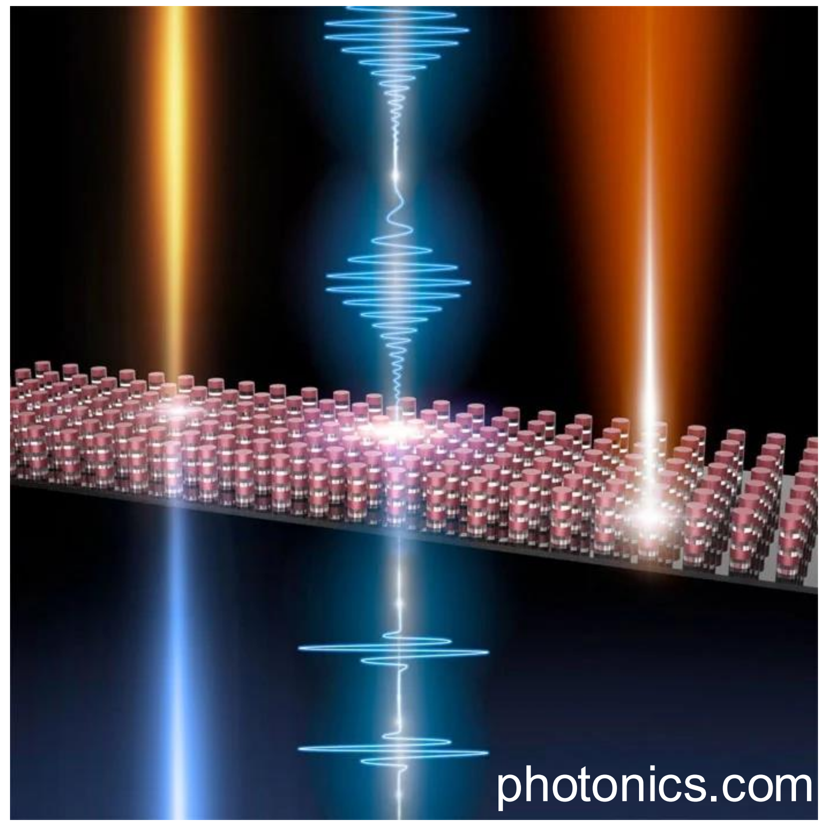


Materials. 2019; 12(4):635

/metamaterials – definition

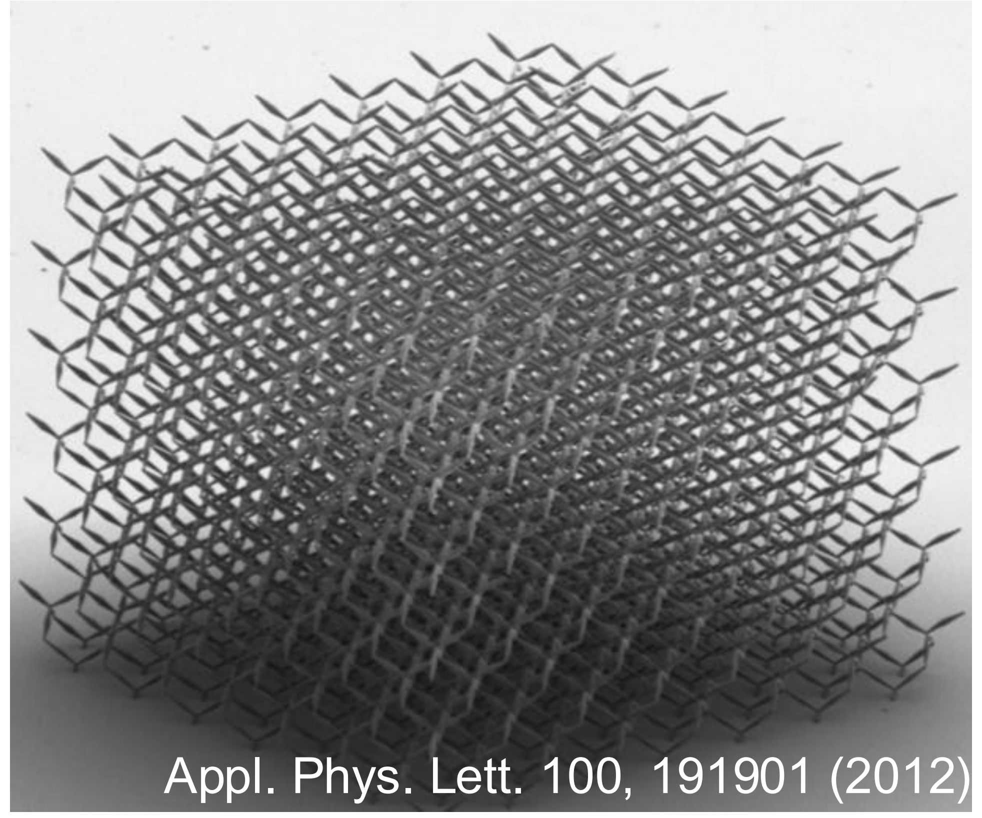
"Metamaterials are materials with an inner structure, which can be tailored and engineered to yield properties that are not found in naturally occuring materials."

Electromagnetic/optical

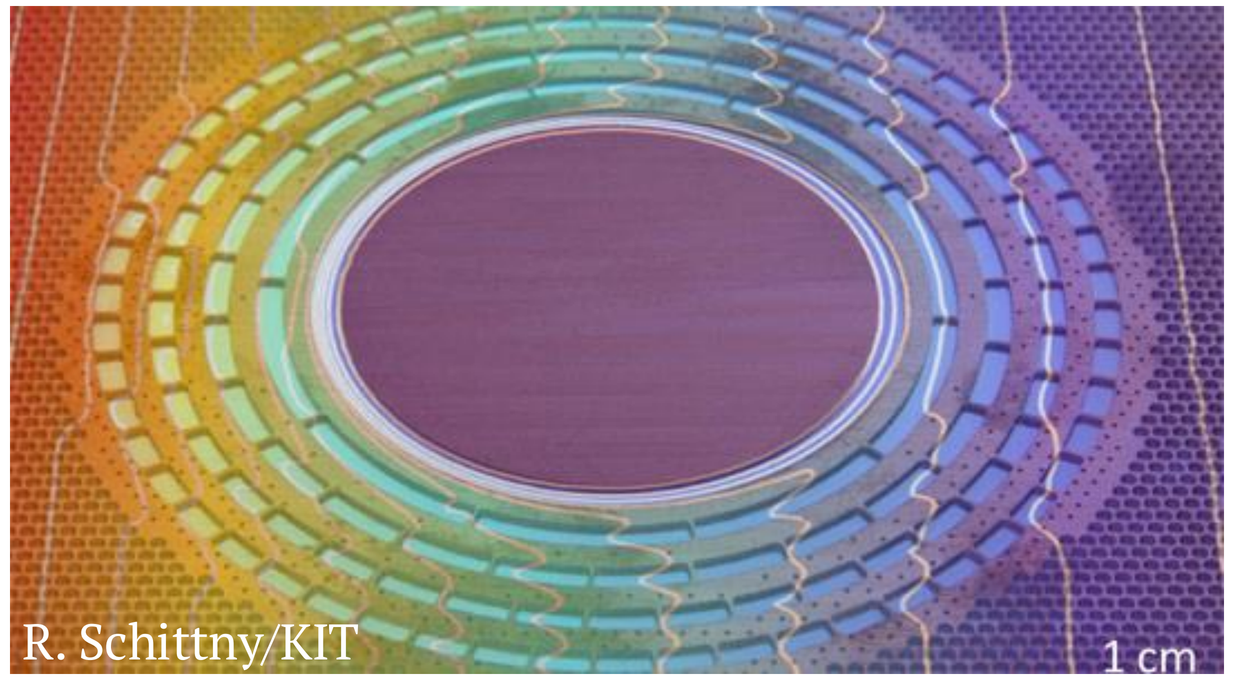


Acoustic

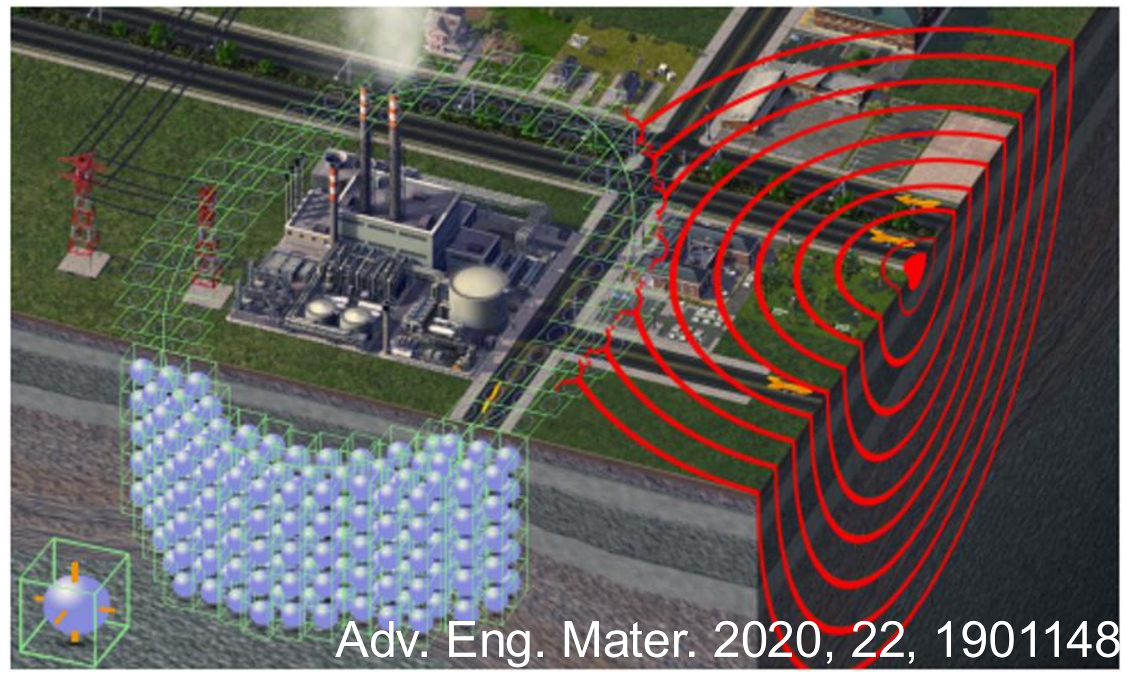
Mechanical



Thermal



Seismic



/metamaterials – applications

Special/smart properties

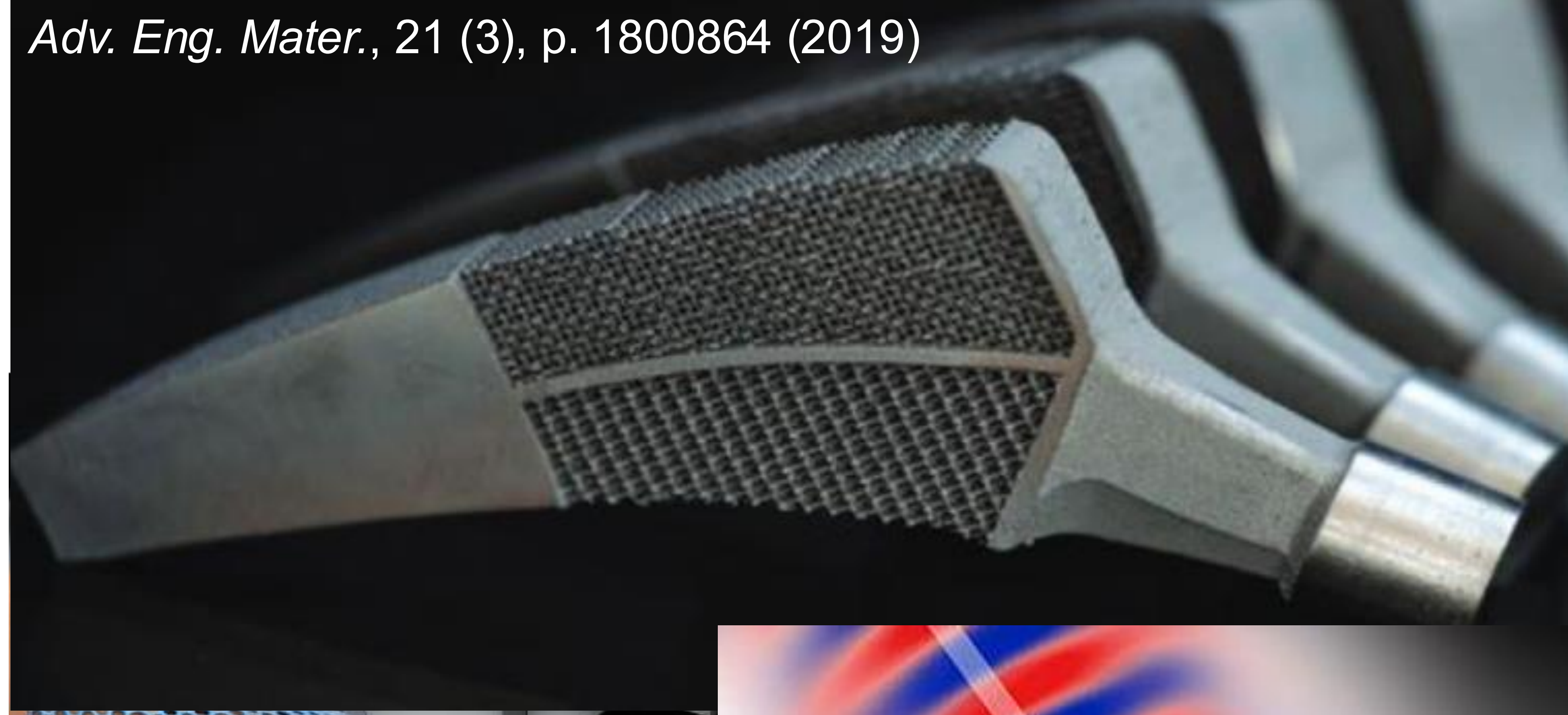
- Auxetics (negative Poisson's ratio $v < 0$)
- Negative compressibility ($K < 0$)
- Pentamode metamaterials (vanishing G/K)
- Electromagnetic/acoustic bandgaps
- ...

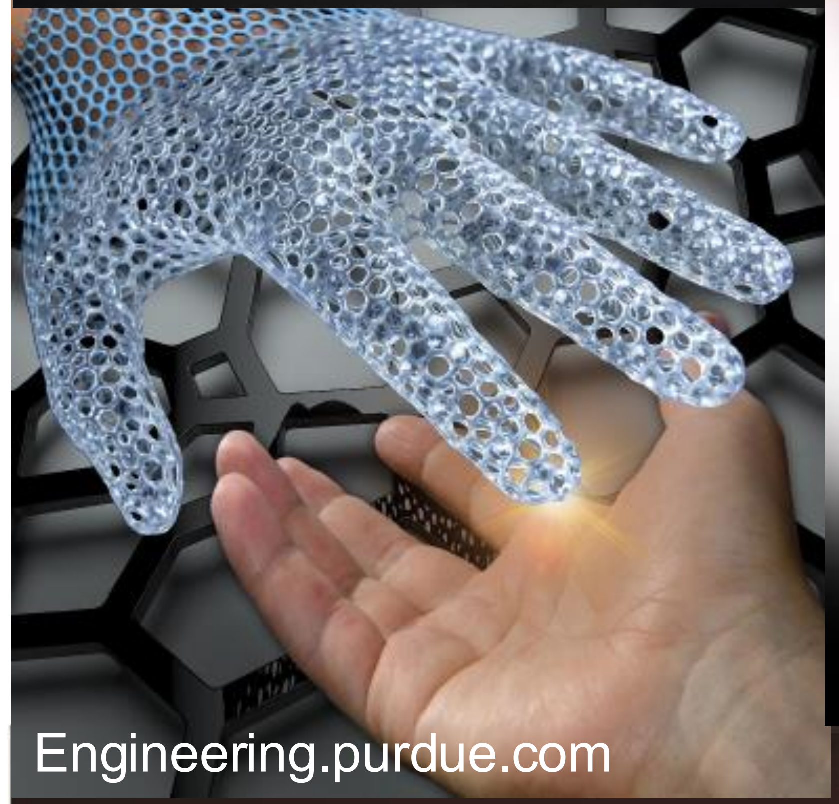
Why?

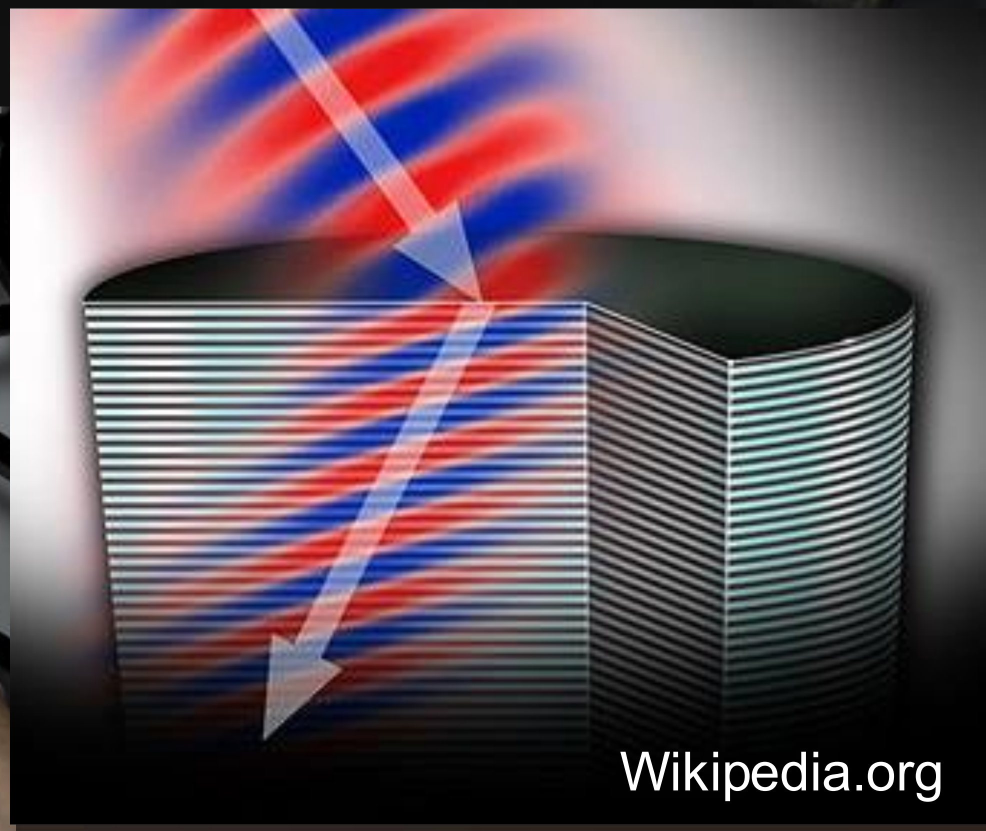
- Increasing performance demands of applications
- On-demand (tunable) properties
- Applications: energy harvesting & storage, switchable stiffness, shock protection, shape-morphing...

What is offered?

- Large design space (geometry, multiphysics interactions, ...) → optimization
- Contradicting requirements (ultra ligth & ultra stiff)

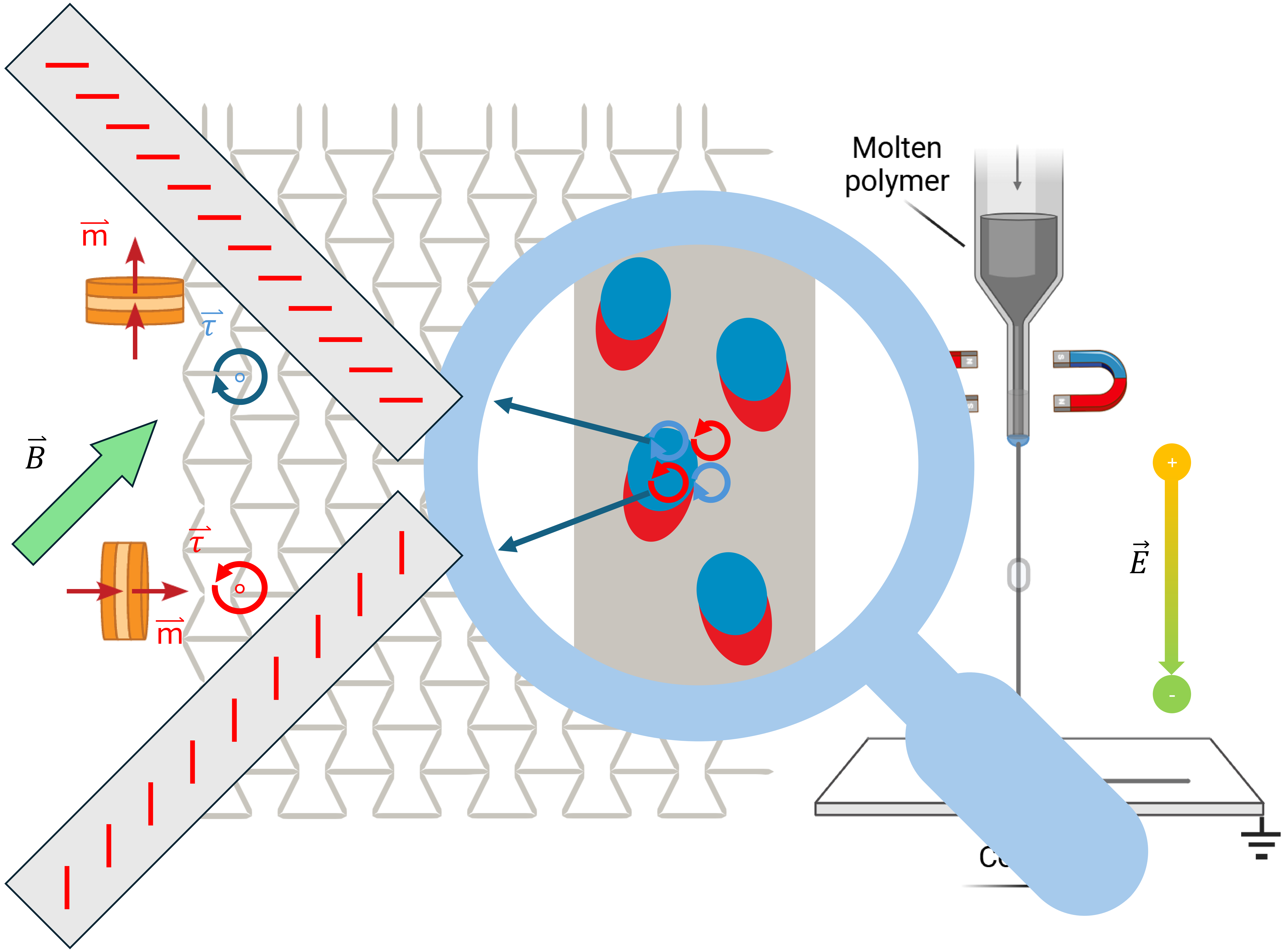

Adv. Eng. Mater., 21 (3), p. 1800864 (2019)

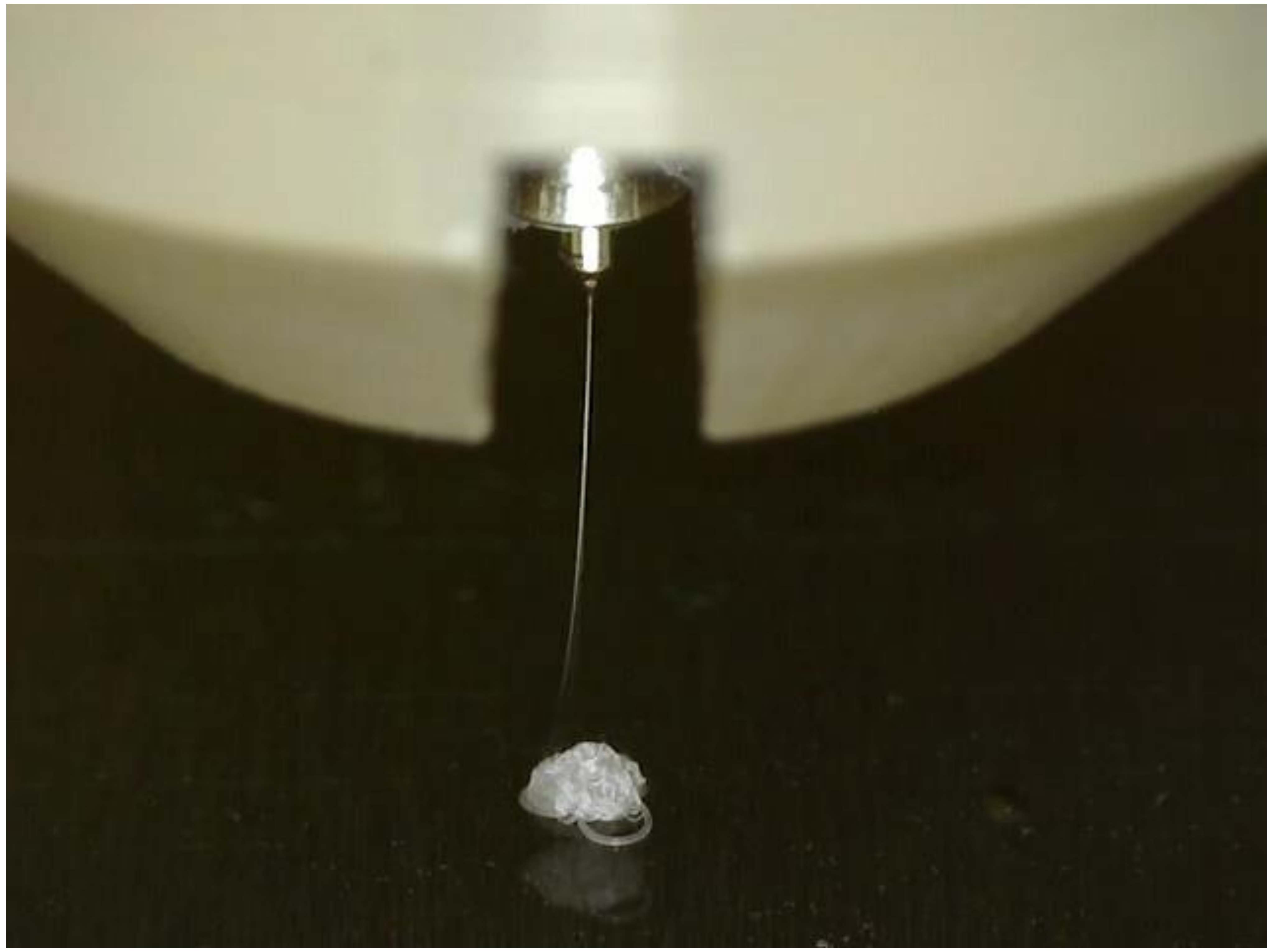

Engineering.purdue.com


Wikipedia.org

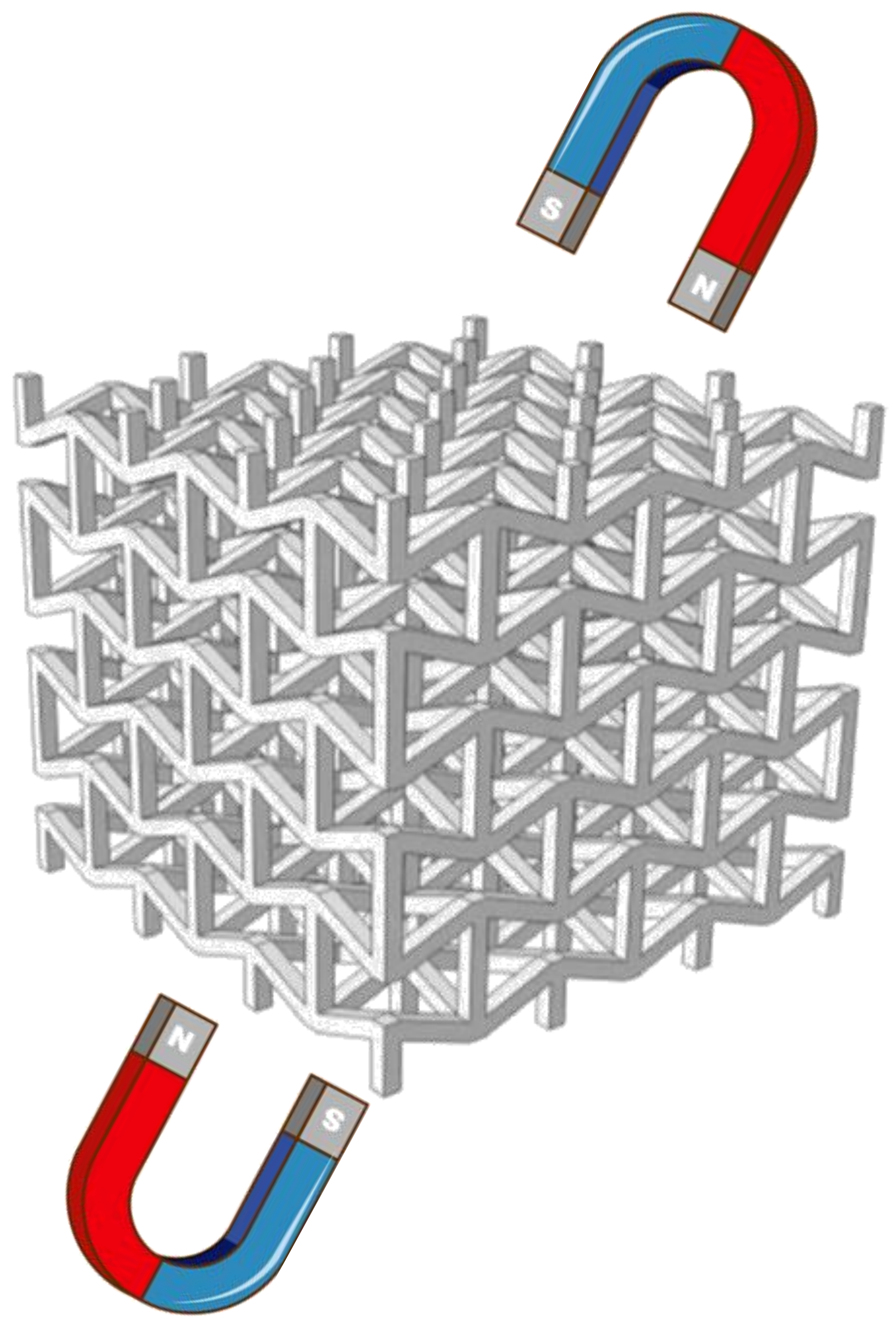

Science (80), vol. 334, no. 6058, pp. 962–965, Nov. 2011

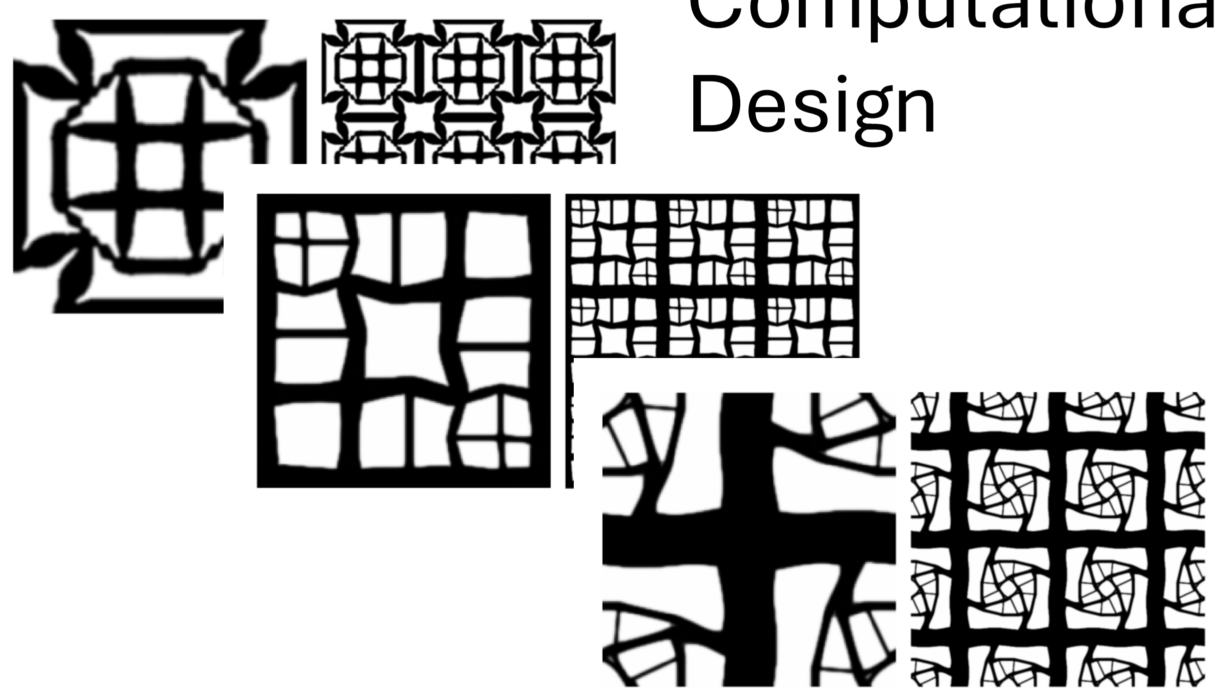
/magnetoactive metamaterials



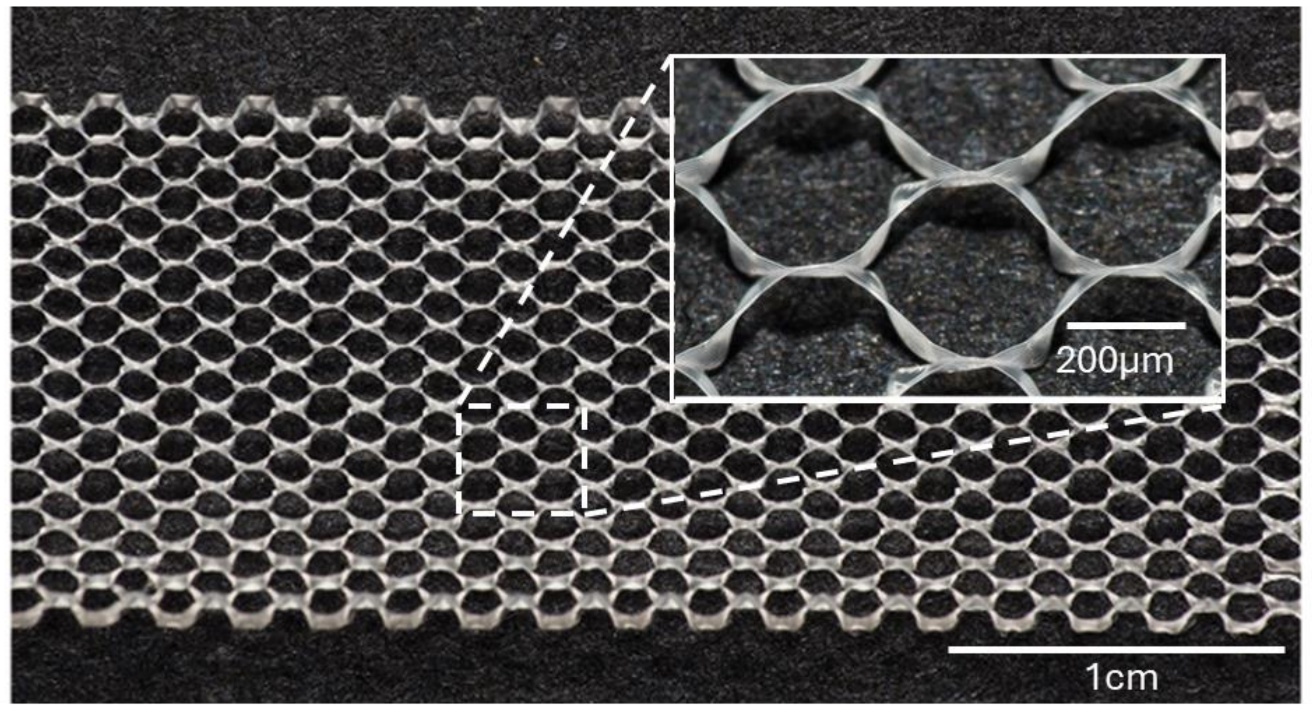


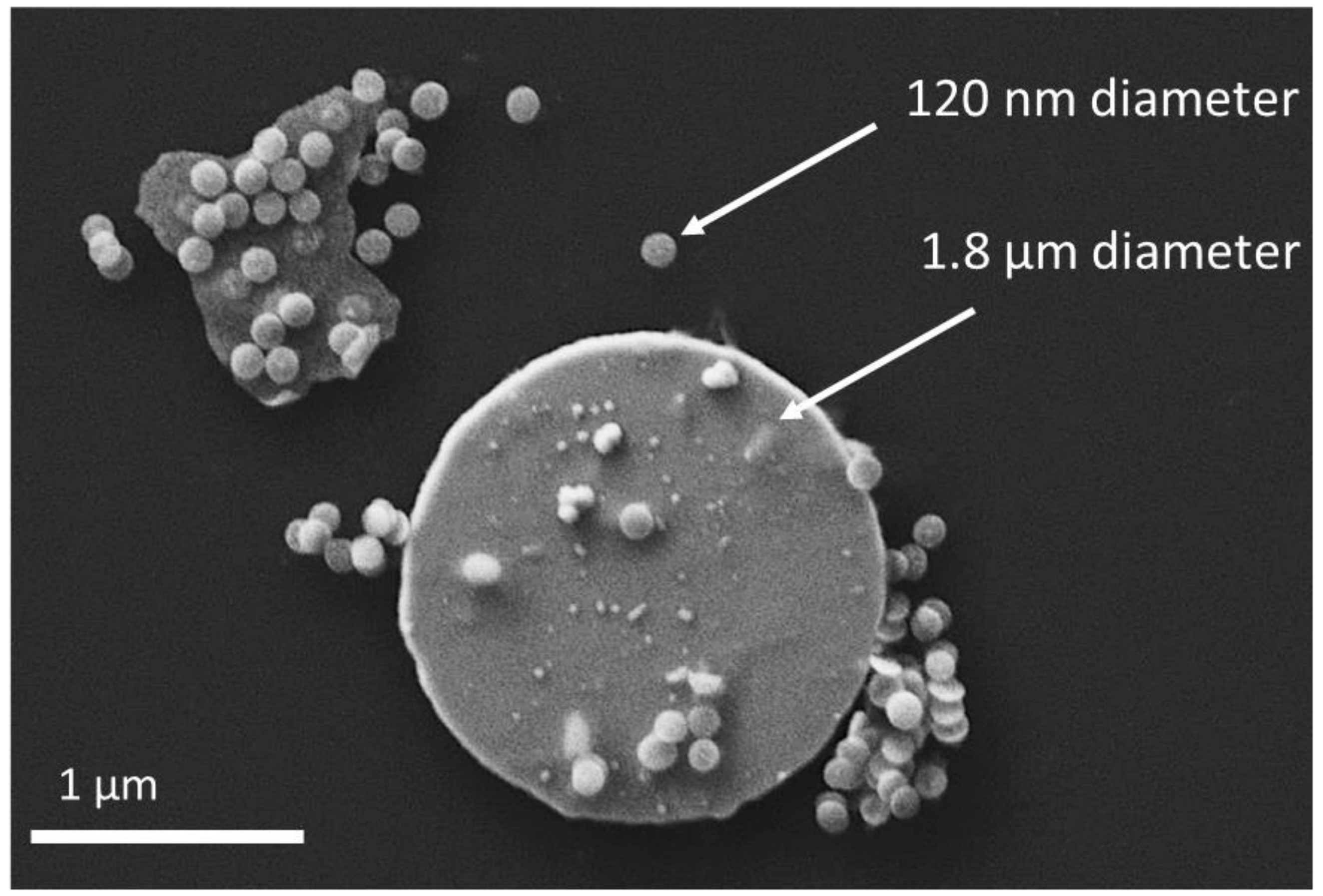
/design space of magnetoactive metamaterials





Computational Design





Magnetoactive metamaterial, programmable stiffness, magnetic field actuation $\vec{B}(\theta)$
~1 cm³

Geometrical design, orientation of platelets
~100s of μm

Magnetic platelets introducing torques
~50 nm – 10s of μm

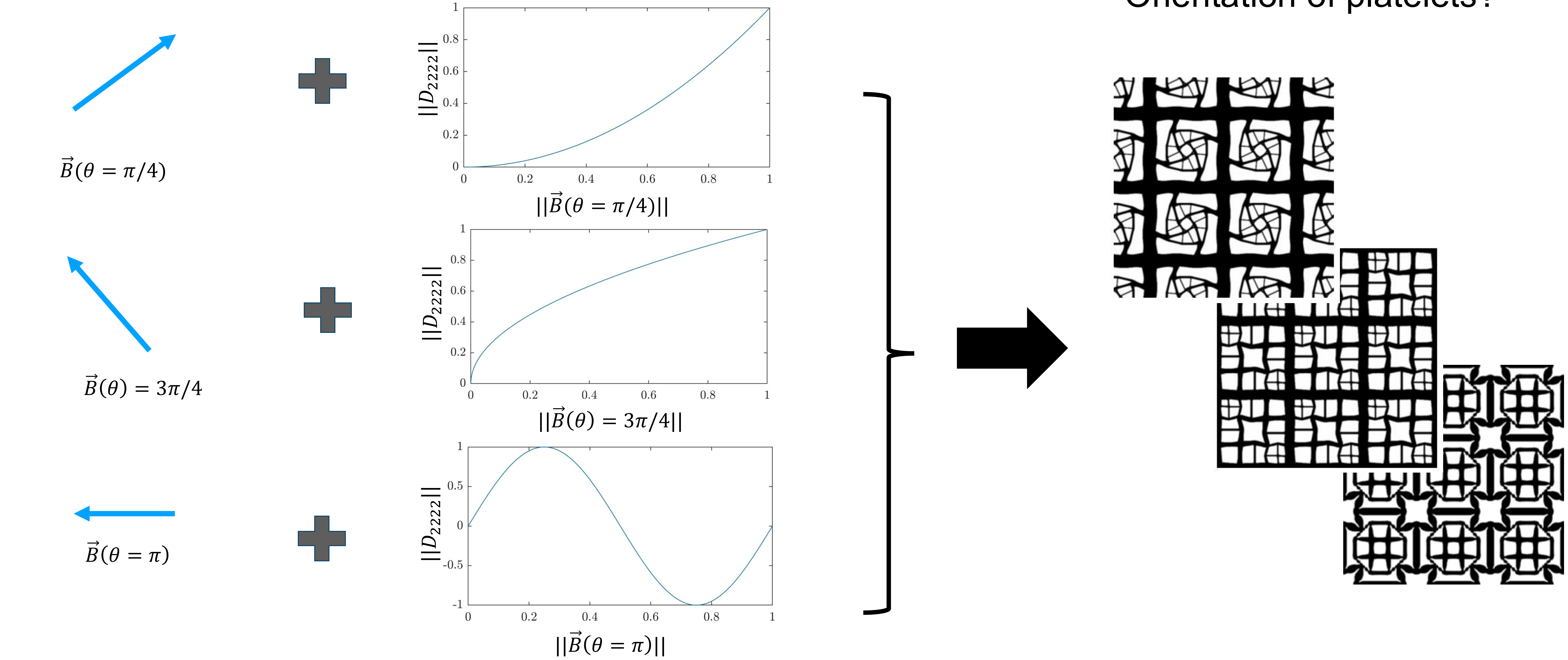
Length scale

/challenge: inverse design of magnetoactive mechanical metamaterials

Magnetic field $\vec{B}(\theta)$

Target effective stiffness D_{2222}

Internal structure?
Orientation of platelets?



/generative design through denoising