Artificial Intelligence for Simulation

an R&D collaboration program

Atos BDS R&D - AI4Sim Group

Gaël Goret – Group Leader
Cédric Bourrasset – Product Manager

Alexis Giorkallos
Léo Nicoletti
Mathis Peyron
Rémi Druilhe
– Dev Team

CRIANN Scientific Day 2020

Trusted partner for your Digital Journey

Thursday 19th November 2020

Atos

Artificial Intelligence for Simulation

an R&D collaboration program

Summary

0. HPC and data-driven approach
1. An application to flight physics
2. The use of space correlation
3. A collection of NN models for scientific ML
4. Integration with the SU2 open-source solver
5. Learning from PDEs: solving equations with PINN
6. The Ai4Sim Library

Atos AI in HPC, AI for HPC and AI enhanced HPC

HPC/AI Converged Infrastructure

BullSequana XH2000

Codex AI Suite

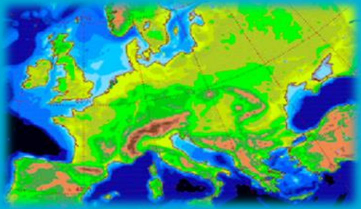
- ✓ Reduce TCO for HPC and AI
- ✓ Data science platform for HPC & Large-Scale AI

AI-driven optimization tool

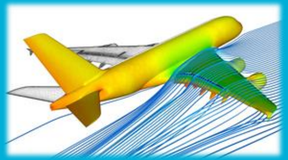
- Resource Management
- Job Placement
- Communication
- Data Movement

- ✓ Boost performance with AI
- ✓ Learn from your production

Integrating ML in Science application



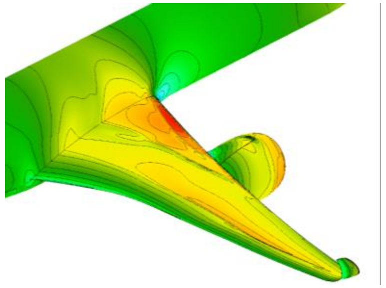
Research & Innovation

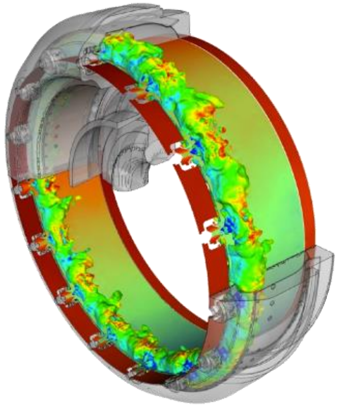


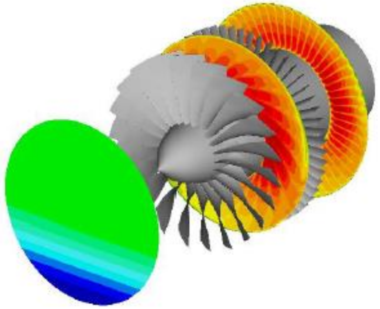
- ✓ Improve simulation with data-driven approach
- ✓ Atos data science teams

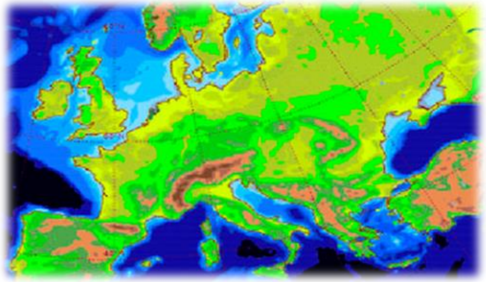
AI4Simulation: an R&D collaboration program

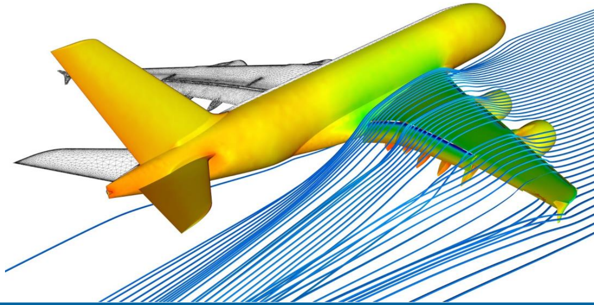
The Ai4Sim collaboration program aims at **co-designing and co-developing simulation solutions** with industrial and academic partners to demonstrate how **artificial intelligence** can make physical modeling **more accurate and efficient** and how they enable addressing new simulation challenges.









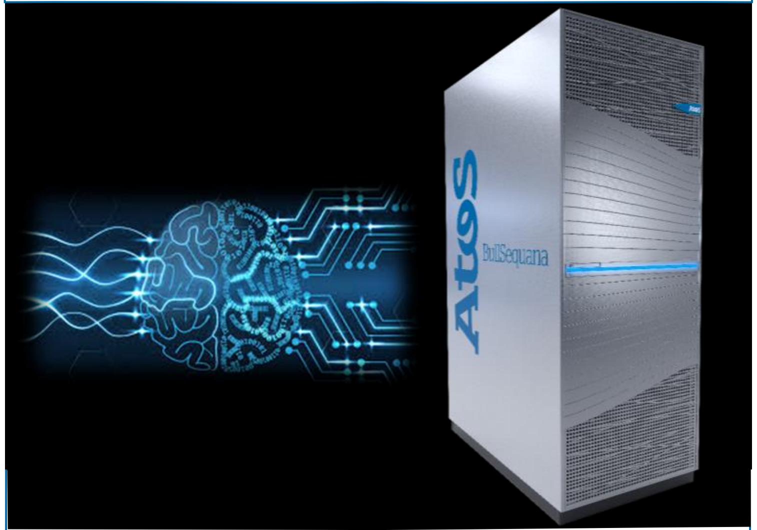


Ai4Simulation: 3 main development tracks

Model Architecture

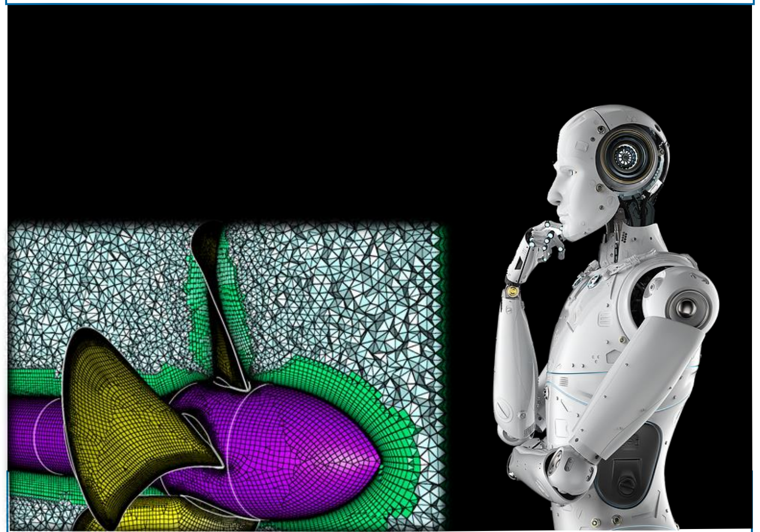
- ✓ Exploring DL technics to surrogate Physical Models (MLP, CNN)
- ✓ Unstructured grids, Mesh Free approach, Physics-informed NN models (PINN, HNN), etc.

Coupling AI & Simulation



- ✓ Advanced data coupling between ML inference & numerical solvers
- ✓ AI/HPC workflow orchestration for continuous improvement.

Online/Meta Learning Strategy



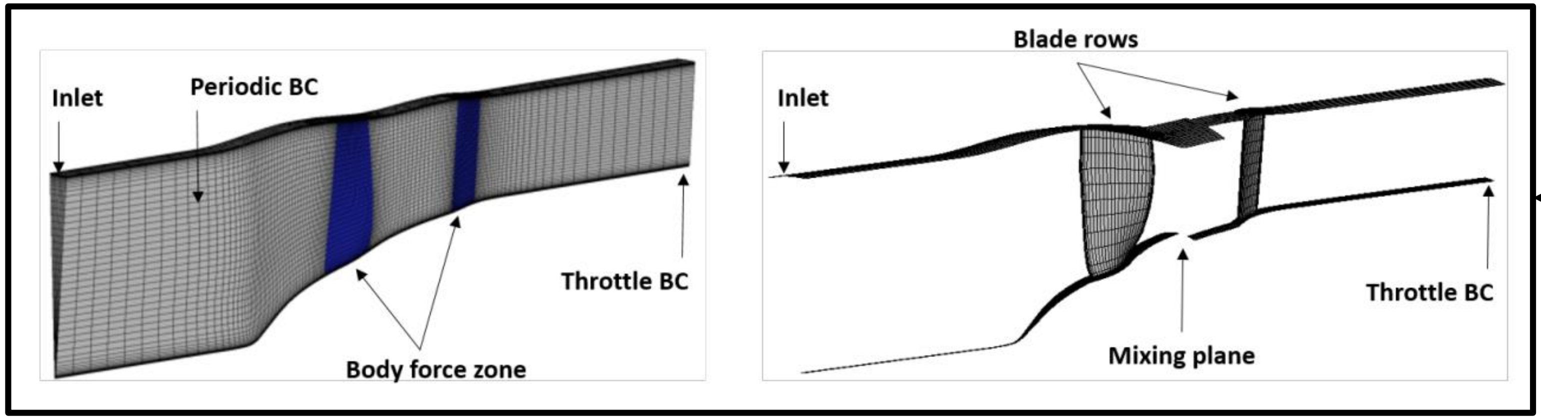
- ✓ Hyperparameters & topologies optimization (AutoML)
- ✓ Automatic data refinement for surrogate modeling & simulation efficiency

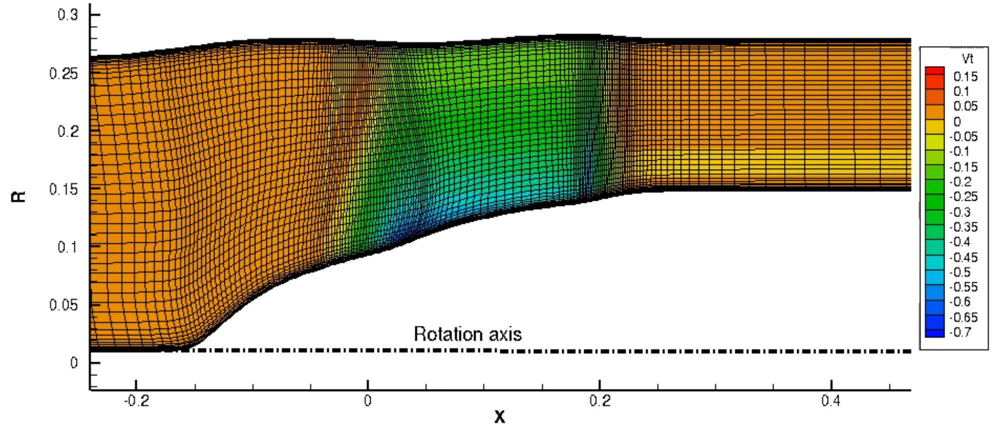
1 An application to flight physics

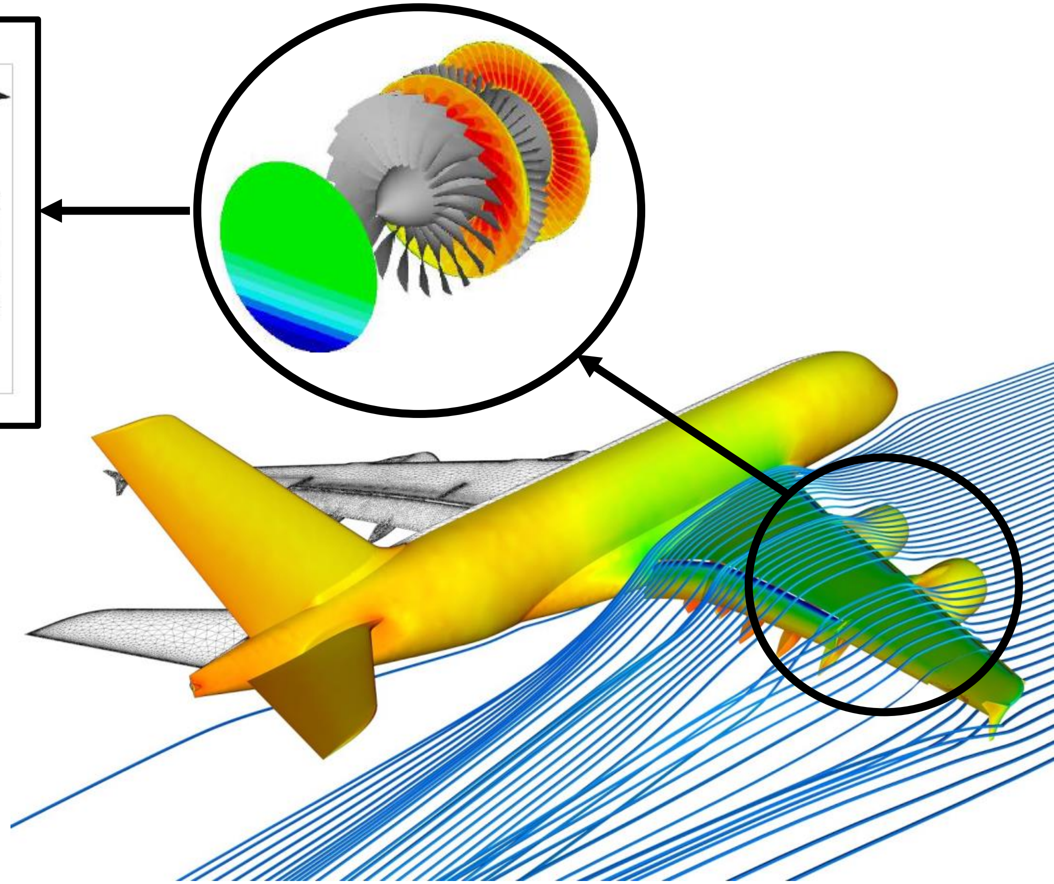
An application to flight physics

Body forces modeling using Machine Learning

- First implementation described in [Lopez de Vega et al, Global Power and Propulsion Society, 2018]()
  - Validate the CFD+AI approach but needed improvements to reach full potential

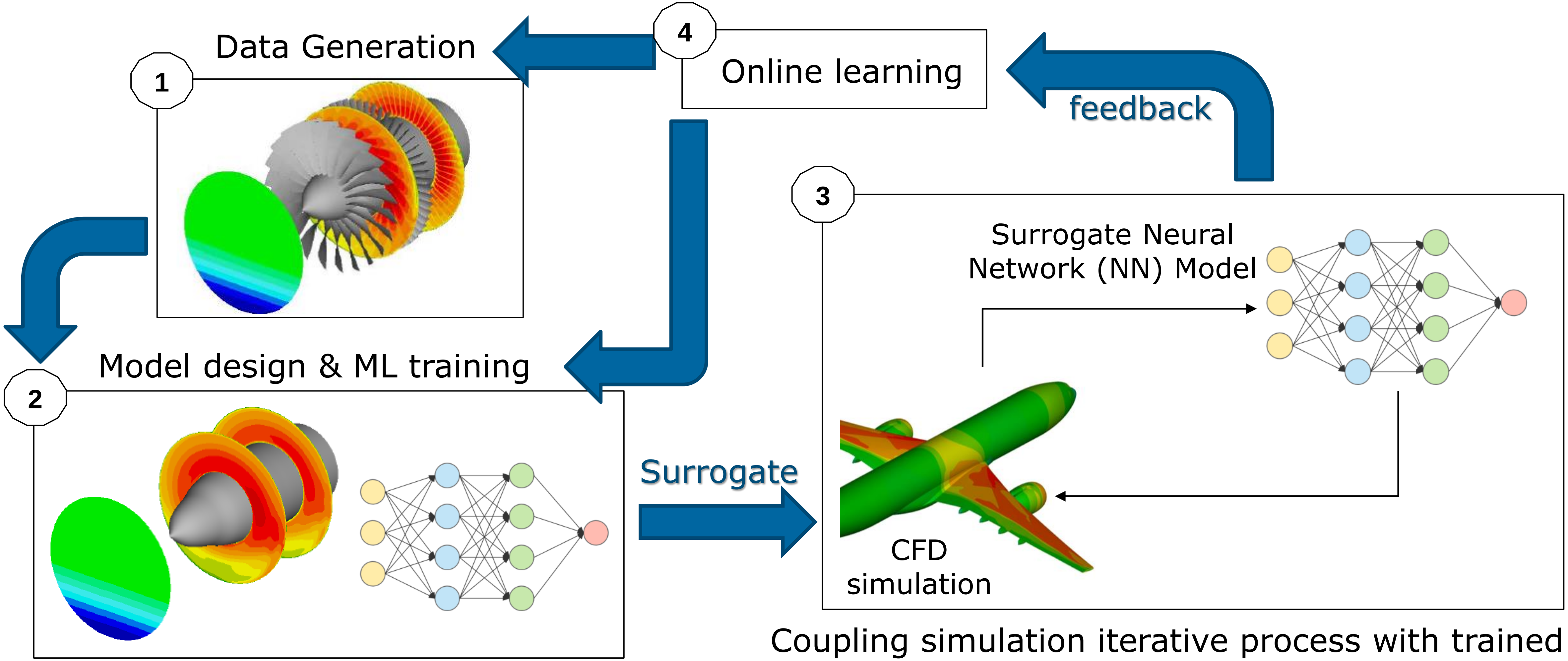






Deep learning approach to surrogate Physical Models

AI-Augmented Workflows

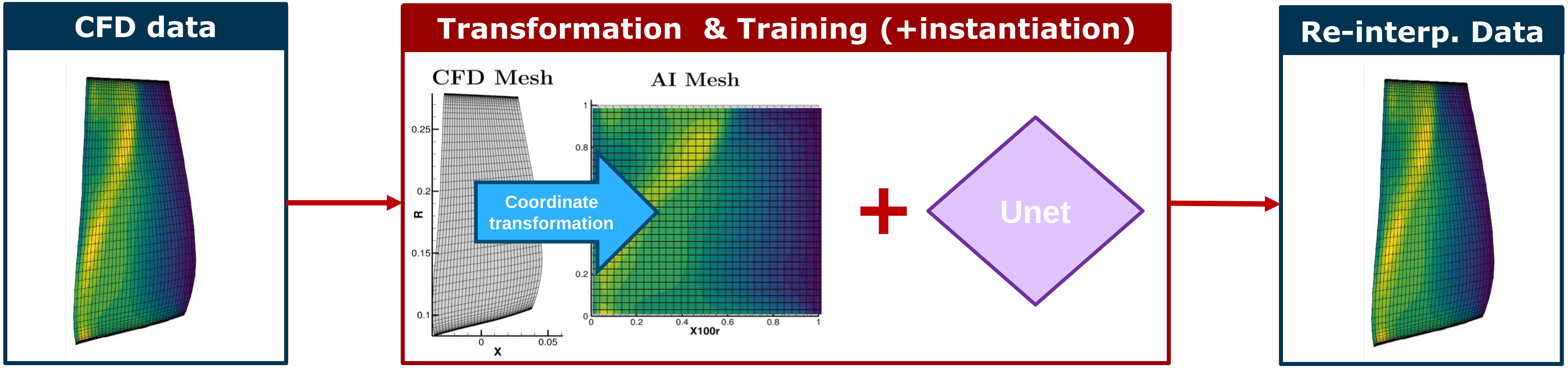


Coupling simulation iterative process with trained NN

2 The use of space correlation

Use of space correlation
a ConvNets approach with Unet

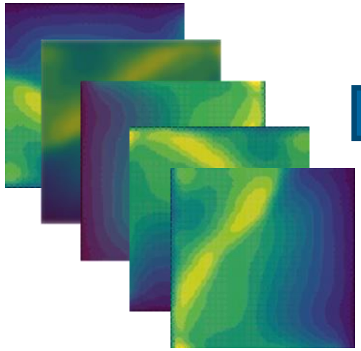
- **Goal:** make use of spatial correlations in AI-based body force models
- **Unet** architectures adapted for performing a **regression** task
- Process chain involves data **interpolation**, training (/instantiation) & re-interpolation

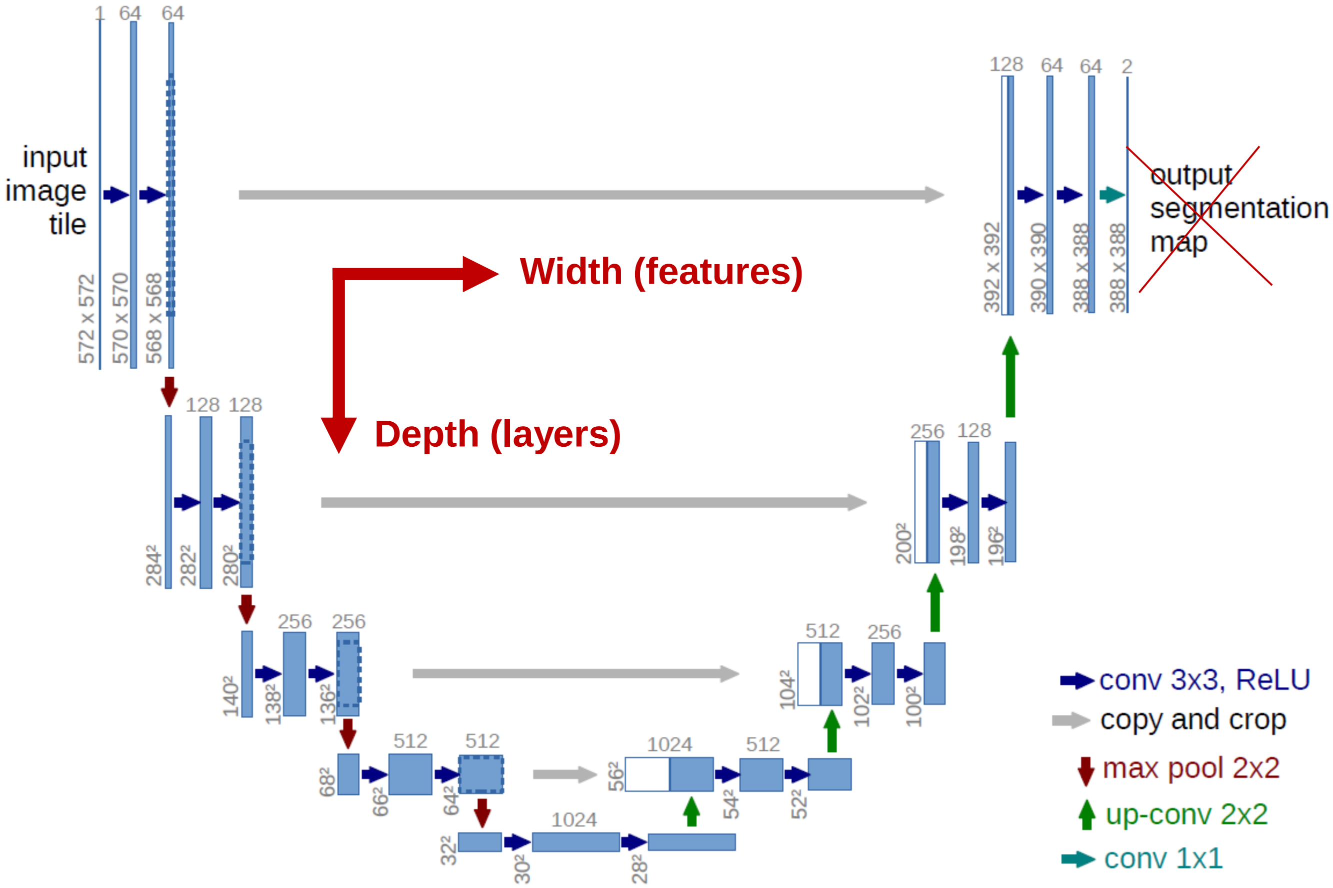


Use of space correlation
a ConvNets approach with Unet

▶ **Unet** architecture has been adapted for **regression tasks**

CFD data



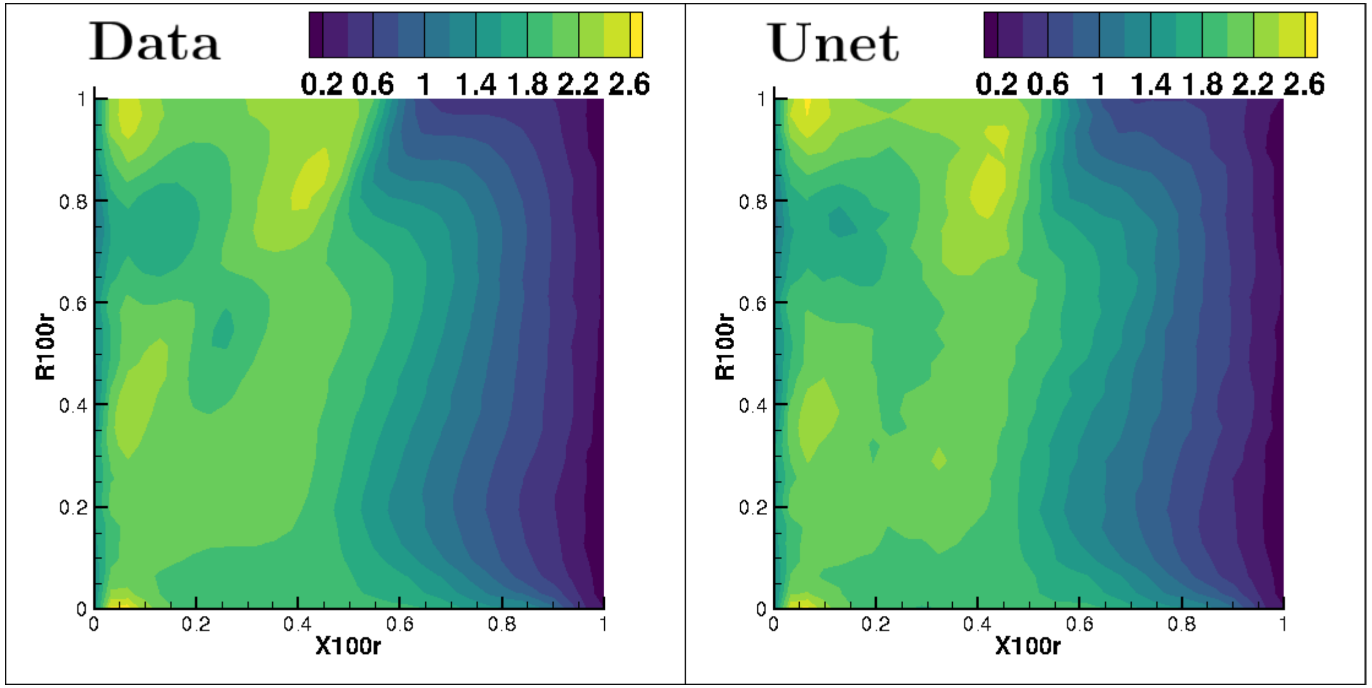


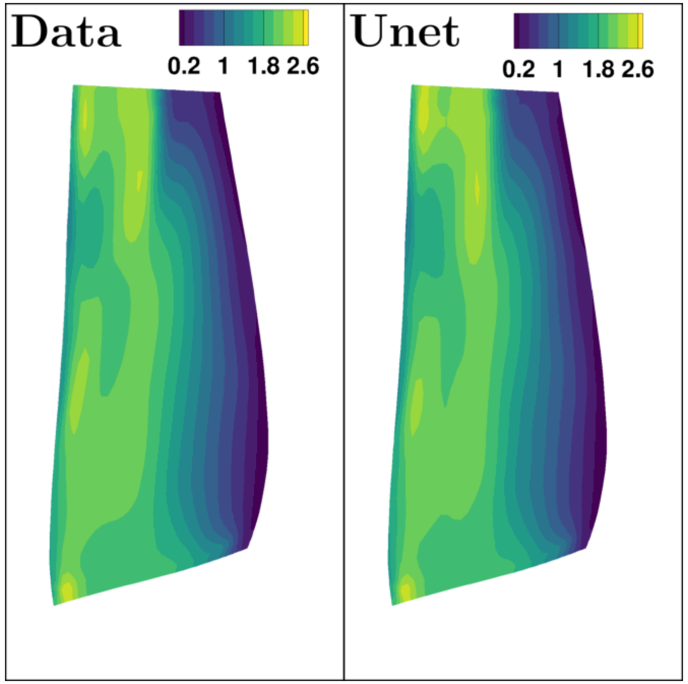
L2 norm:

$$\sum_{im}\sum_{i,j}[f(i,j)-\hat{f}(i,j)]^2$$

Use of space correlation
a ConvNets approach with Unet

- **Application to fn modeling in rotor blade**
- **Topology:** 3x3 filters, 3 layers, [64, 128, 256] features, **R^2 > 0.99**



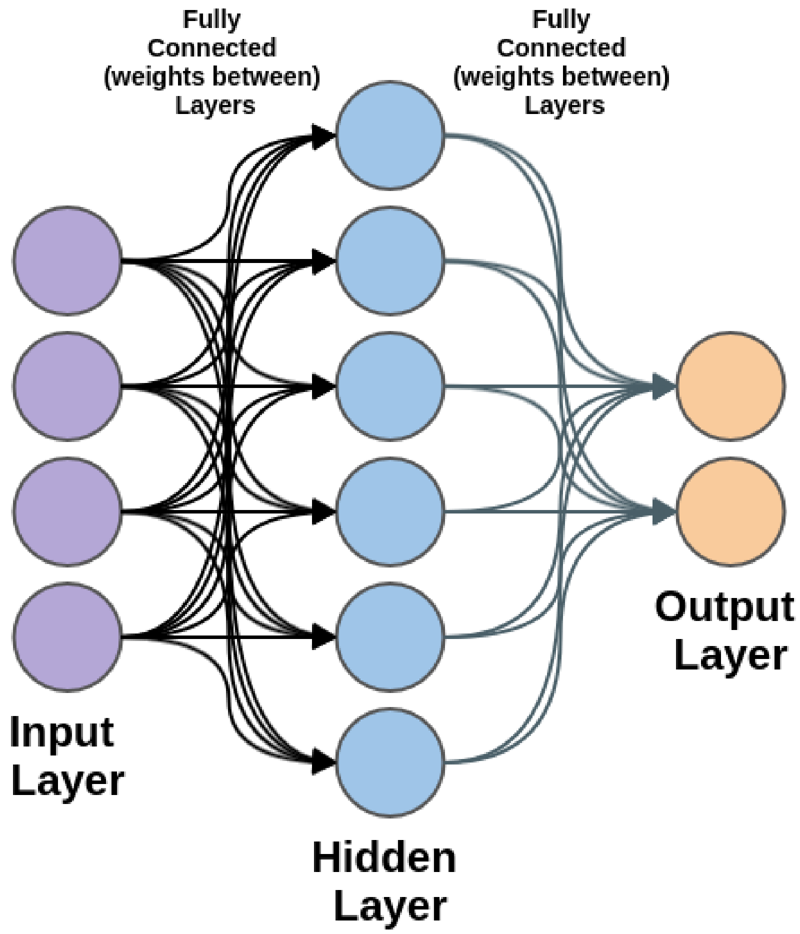


3

A collection of NN models for scientific ML

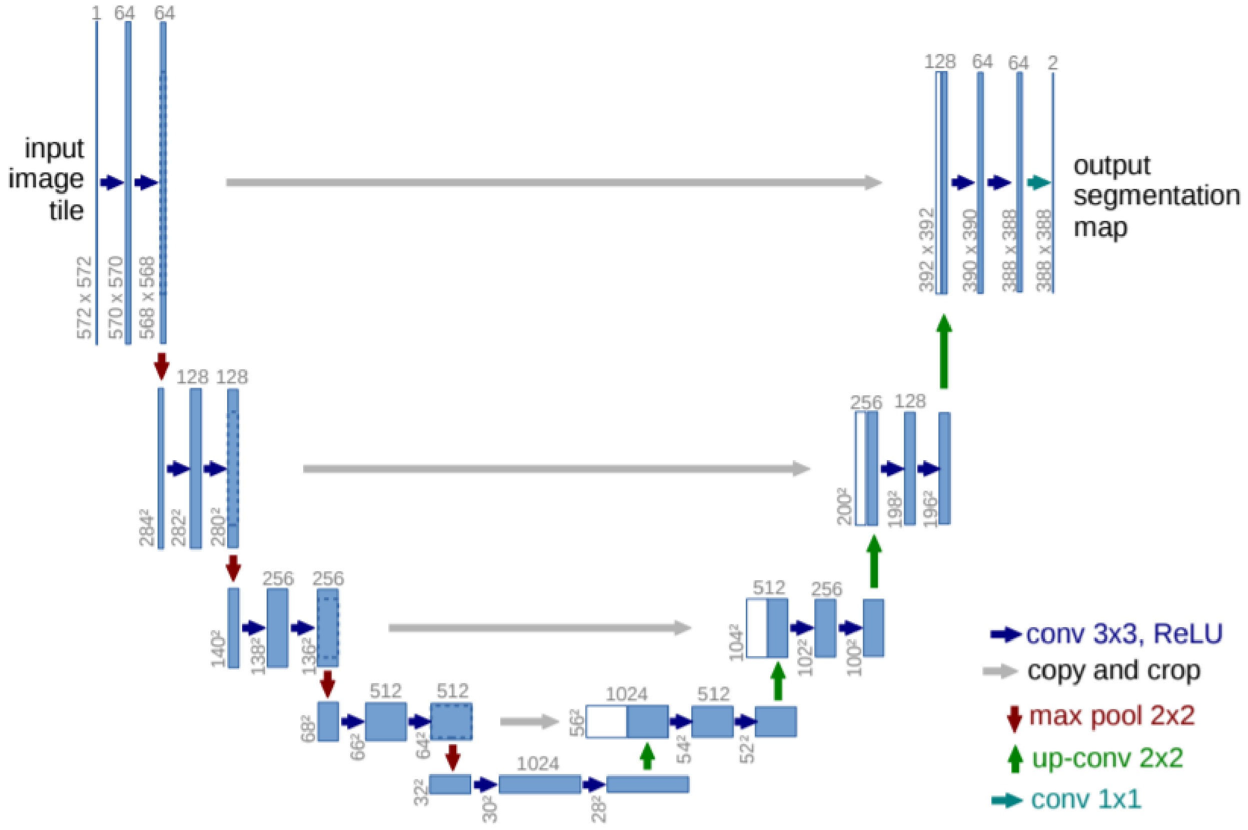
Multilayer Perceptron (MLP)
Baseline for deep learning methods

- No use of spatial structures / correlations
- Deployed in multiple Ai4Sim use-cases:
  - Machine learning body forces (flight physics)
  - 2D CFD simulations (e.g. vorticity-streamfunction)



U-Net
Convolutional contracting networks

- Network adapted for spatial data (2D, 3D) with convolution kernels.
- Contracting to a *latent space* for essential information encoding, *eg.* physical dynamics…
- Successfully applied to:
  - Biomedical image segmentation (Ronneberger et al.)
  - Machine learning body forces (flight physics)
  - Turbulent sub-grid scale combustion (Lapeyre et al.)



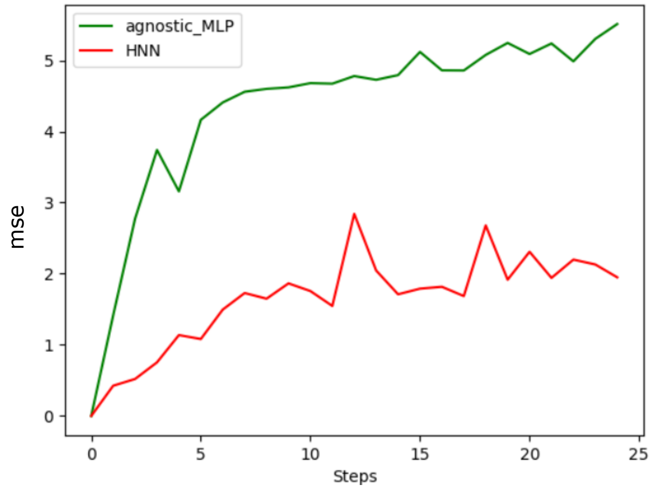
O. Ronneberger, P. Fischer, T. Brox. *U-Net: Convolutional Networks for Biomedical Image Segmentation*. May 2015.

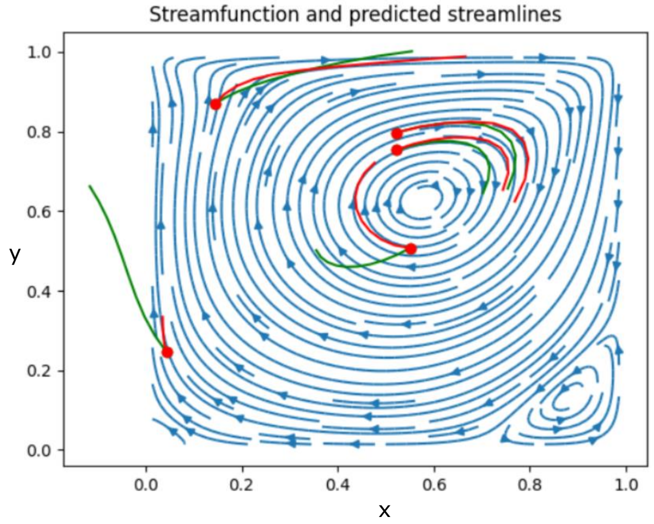
Hamiltonian Neural Network (HNN)
Physics-informed deep learning

- Networks informed by physics, learning an Hamiltonian $\begin{cases} \dot{q} & = +\frac{dH}{dp} \\ \dot{p} & = -\frac{dH}{dq} \end{cases}$
- *Soft constraint*: physical constraint hardcoded in the loss function

$$\underset{\theta}{\operatorname{argmin}} \left\| \frac{d\mathbf{q}}{dt} - \frac{\partial \mathcal{H}_\theta}{\partial \mathbf{p}} \right\|^2 + \left\| \frac{d\mathbf{p}}{dt} + \frac{\partial \mathcal{H}_\theta}{\partial \mathbf{q}} \right\|^2$$

- Successfully applied to:
  - Learning the streamfunction (Ai4Sim R&D) from 2D CFD Dynamic Numeric Simulation

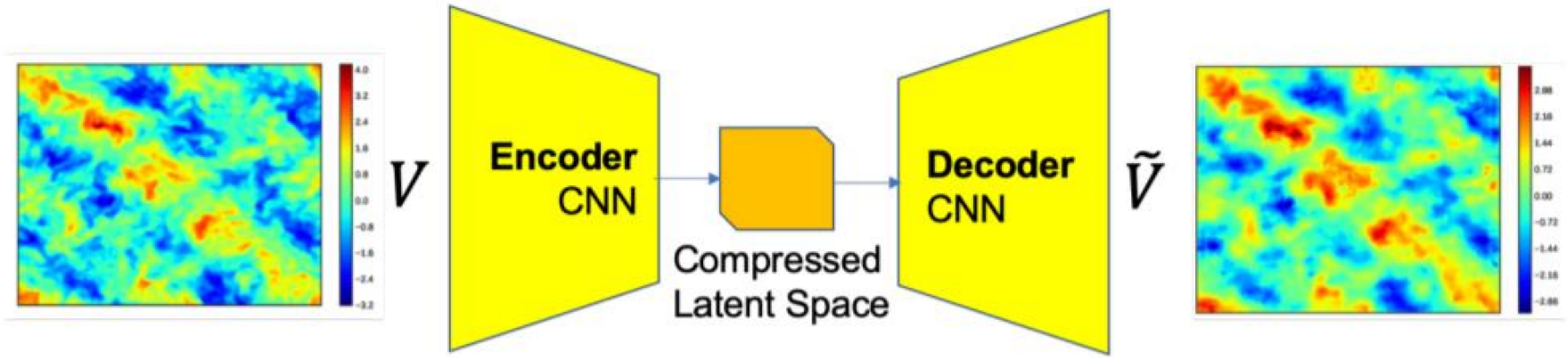




S. Greydanus, M. Dzamba, J. Yosinski. *Hamiltonian Neural Networks.* Sep 2019.

Convolutional Auto-Encoders (CAE)
Physical constraints in a latent space

- Encoding in a latent space
  - Soft constraints: physical constraints in the latent space (*e.g.* VAE, CAE-HNN)
  - Hard constraints: physical constraints as topology (*e.g.* Conv kernel as numerical stencils)



- Currently being explored/benchmarked

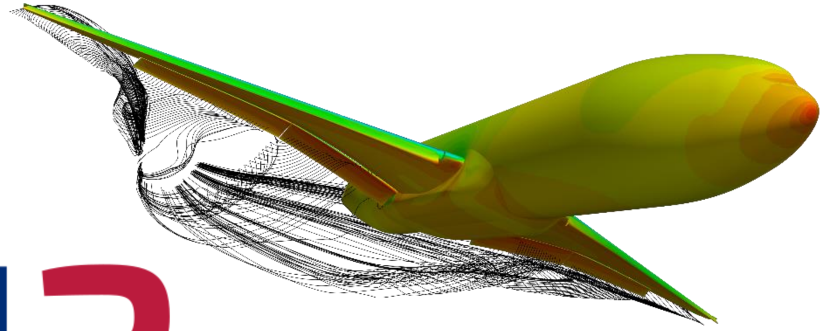
T. Mohan, N. Lubbers, D. Livescu, M. Chertkov. *Embedding hard physical constraints in neural network coarse-graining of 3D turbulence.* Feb 2020.

4

Integration with the SU2 open-source solver

Integration with SU2 open-source solver
Overview

SU2 is an open source Solver for Computational Fluid Dynamics (CFD) with a modular architecture

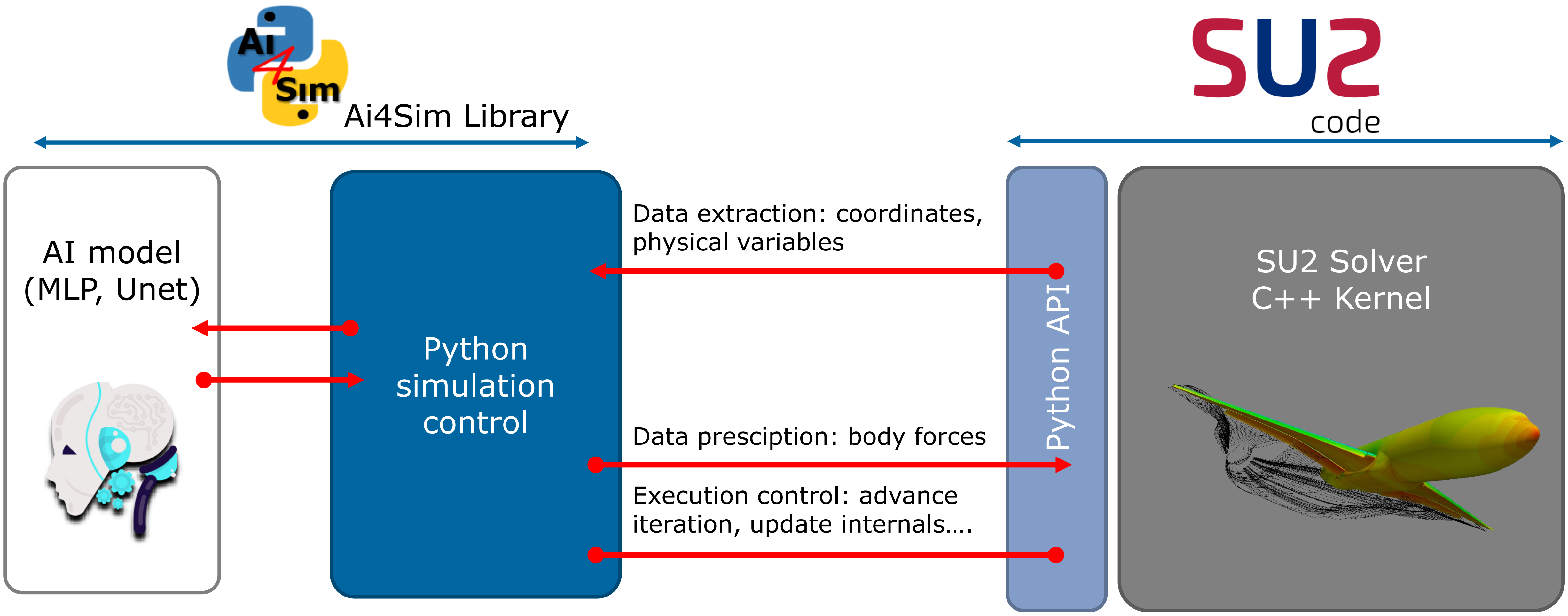


SU2 code

Data Generation:
- Mixing plane simulation chain.

Interfacing external ML model:
- Body force integration

Integration with SU2 open-source solver
Control & Coupling



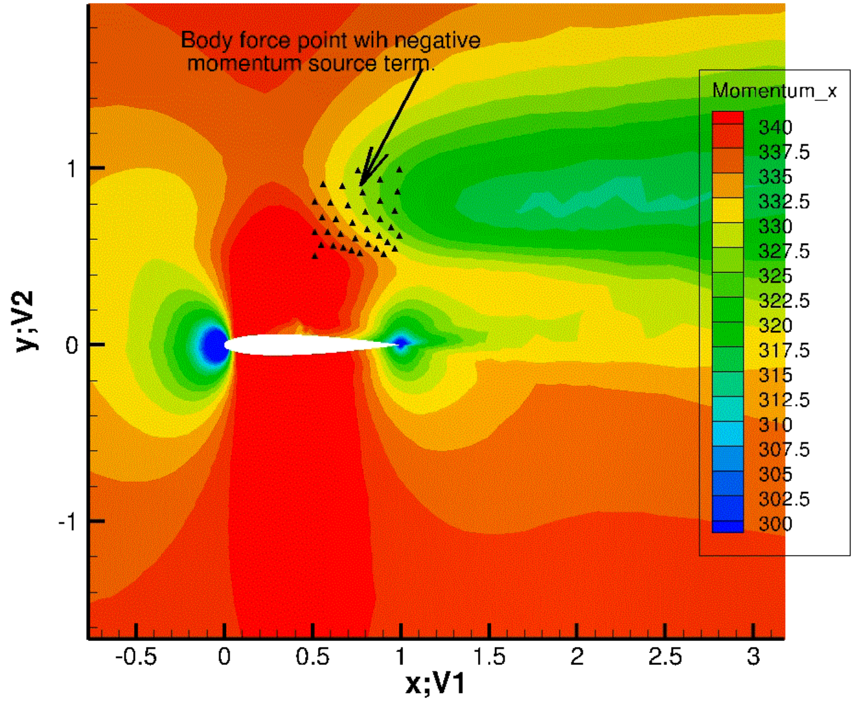
Atos

Integration with SU2 open-source solver
Basic Validation

```
FluidSolver.ResetConvergence()
while Iter < 1000:
  FluidSolver.Preprocess(Iter)
  #set the body forces
  for iNode in range(FluidSolver.GetTotalNumberNodes()):
      xCoord = FluidSolver.GetNodeCoordX(iNode)
      yCoord = FluidSolver.GetNodeCoordY(iNode)
      if xBFMin < xCoord and xCoord < xBFMax and xBFMin < yCoord and yCoord < xBFMax:
          #Collect the flow variables
          consVar = FluidSolver.GetSolution(iNode, iVar)

          #Some processing to implement here


          #Apply body forces in the CFD solver
          FluidSolver.SetVariable_BodyForce(iNode, iVar, 100.1)
  FluidSolver.Run()
  FluidSolver.Update()
  Iter += 1
```



5

Learning from PDEs: solving equations with PINN

Learning from PDEs: solving equations with PINN

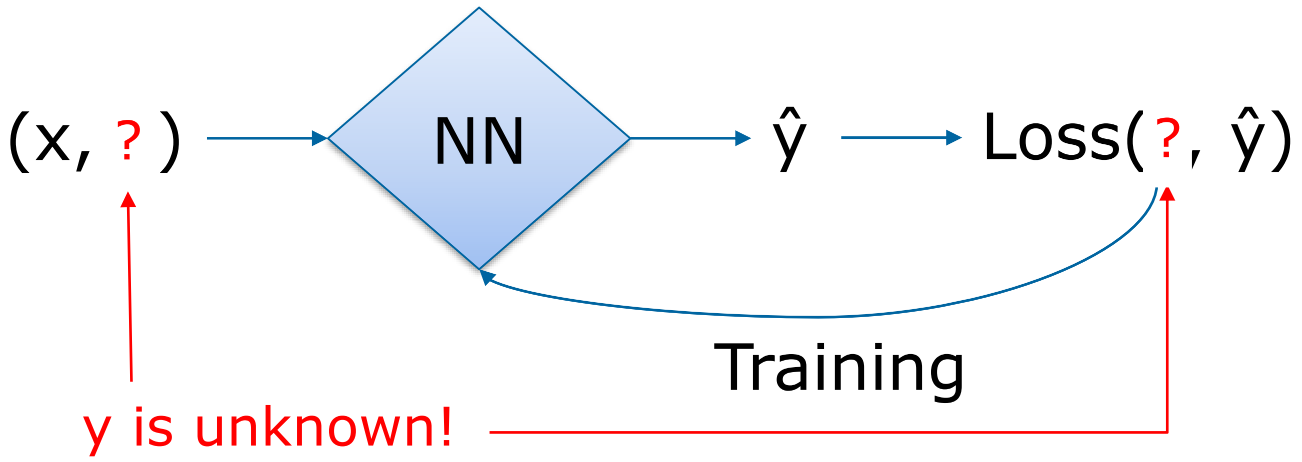
Pros and cons

Classical Methods	AI-Augmented
✓ Good results on mesh	✓ Once trained, fast accurate computations
✗ Resolution limited to mesh topology	✓ Mesh-free resolution
✗ Slow computations	✗ Difficult to tune manually

Atos

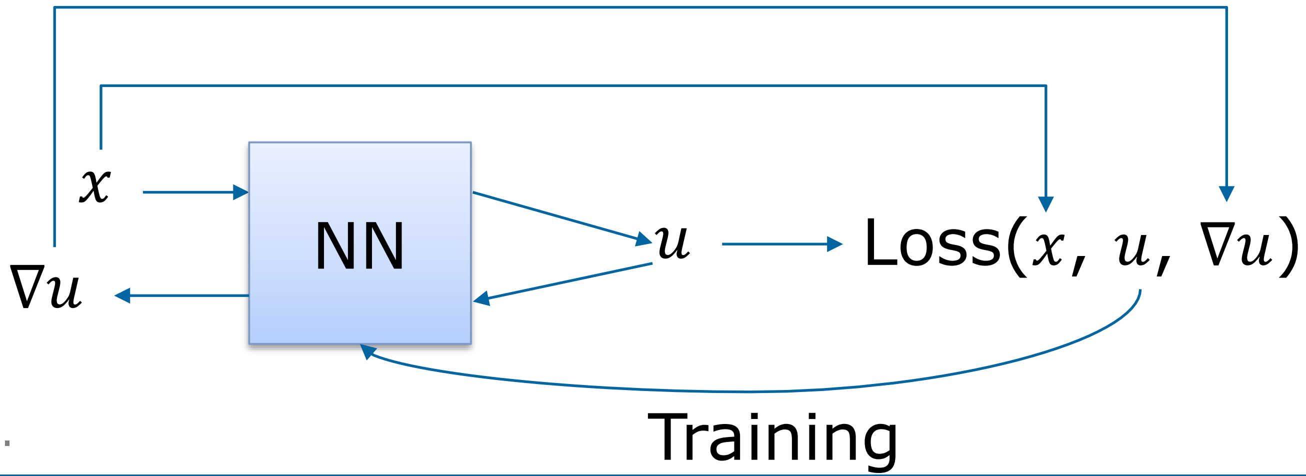
Learning from PDEs: solving equations with PINN
Why PINN ?

- Traditional Supervised ML
  - Known: input data (x, y)



- Physics-Based Supervised ML
  - Known: x, Partial Derivative Equation (PDE), Boundary Conditions (BC)
- Physics-Informed Neural Network
  - Known: x, Partial Derivative Equation (PDE), Boundary Conditions (BC)

(Multi) Loss = PDE residuals + BC residuals



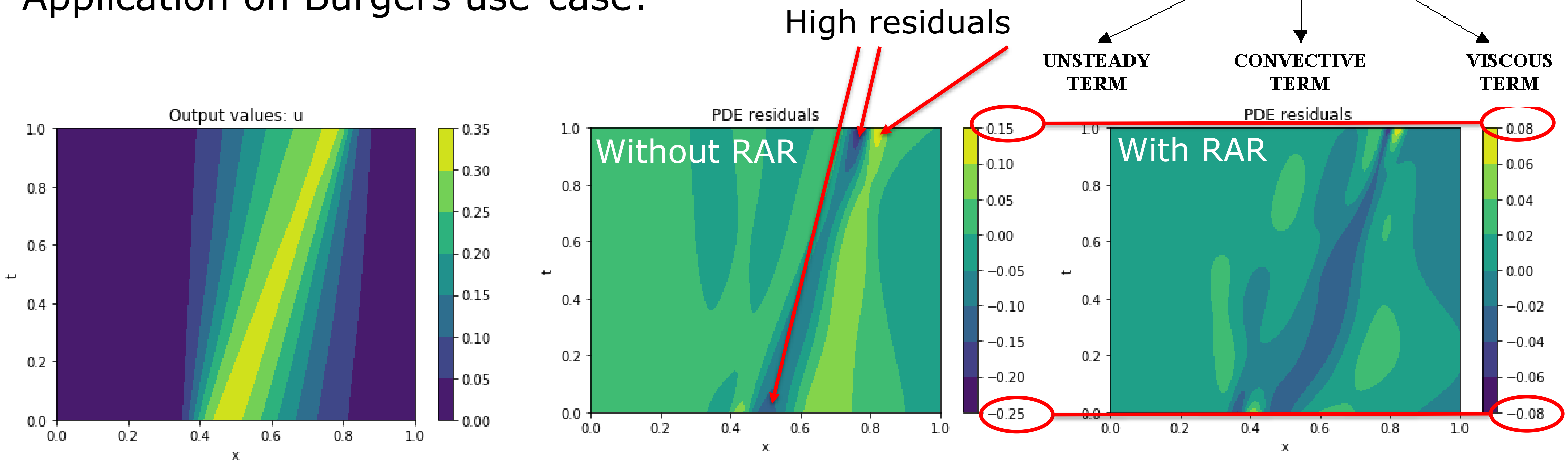
M. Raissi et al (2017). Physics informed deep learning (part i): Data-driven solutions of nonlinear partial differential equations.

Learning from PDEs: solving equations with PINN
Implementation of Residual-based Adaptative Refinement (RAR)

- The idea behind adaptive refinement in the context of PINN is to improve the network training by providing more data in the regions were the residuals are higher

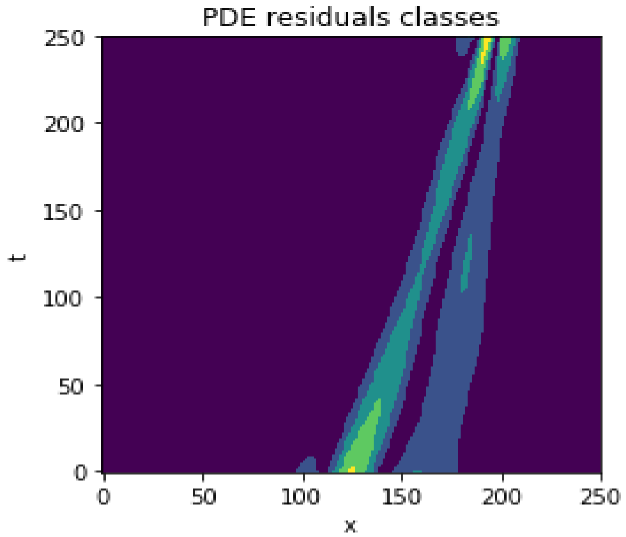
$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} = 0$$

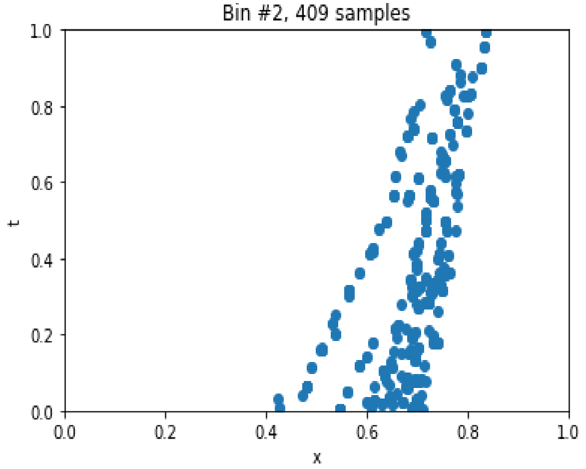
- Application on Burgers use-case:



Learning from PDEs: solving equations with PINN
Implementation of Residual-based Adaptative Refinement (RAR)

- **Classical RAR**
  - We improve the training data set by adding a few selected points with high residuals values
- **Density-based Adaptive Refinement (DAR)**
  - A density map is built from the residual distribution on a set of points (random sampling or mesh)
  - A new training set is generated on the base of this density map
- **Density-based Adaptive Refinement with Bins (DARB)**
  - Different classes are defined on the base of a histogram of residuals
  - Application of the previous DAR to select points from each class





6 The Ai4Sim Library

Ai4Sim Python library
From research to product

One Ai4Sim objective is the maturation and integration of research topics until the highest TRL

Integration

Ai4Sim
Shared BDS R&D AI4Sim

Research
Shared BDS R&D AI4Sim

unittest

- Unit testing coverage (>85%)
- PEP8-compliency, SonarQube, etc.
- TensorFlow 2.x Keras Models
- Continous Integration & Deployment

JFrog

Research topics under productization:

- Physics-Informed Neural Networks (PINN, HNN)
- Latent Space & Physical Constraints
- Hyper-optimization & NN Architecture Search
- Online/Meta Learning
- Simulation Control & Coupling Workflow

Atos

Artificial Intelligence for Simulation

an R&D collaboration program

Thank you !

Any Question ?

For more information on AI4Sim please contact:

Gaël Goret, PhD – Group Leader

+33 476 298 128 gael.goret@atos.net

Atos