Mesh generation and adaptation using green AI

Rubén Sevilla

Zienkiewicz Institute for Modelling, Data and AI
Swansea University, Swansea, Wales, UK

Swansea University
Prifysgol Abertawe

Joint work with

Oubay Hassan

Jason Jones

Agustina Felipe

Callum Lock

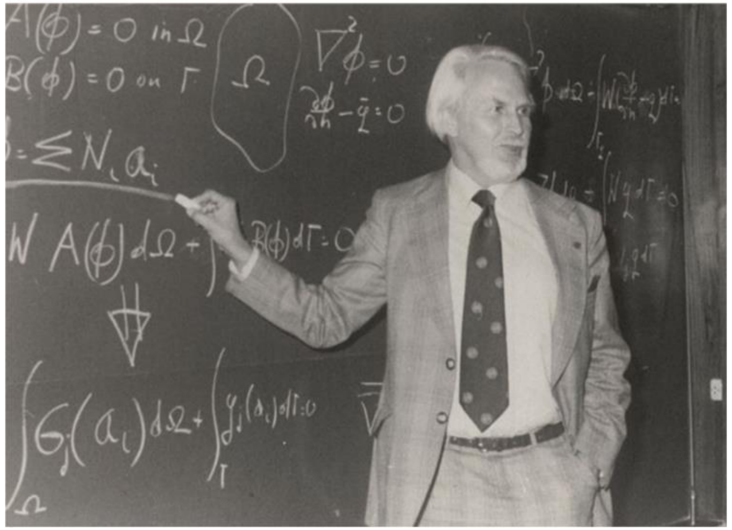
Sergi Sanchez

Zienkiewicz Institute for Modelling, Data and AI
Swansea University, Swansea, Wales, UK

Swansea University
Prifysgol Abertawe

Swansea


Glasgow
Edinburgh
United Kingdom
Isle of Man
Dublin
Ireland
Manchester
Birmingham
Swansea
London



THE MIT PRESS READER

"The Cloud now has a greater carbon footprint than the airline industry. A single data centre can consume th… …es."

HPC wire

"The Exascale era is here and po… consumption for HPC is skyrocke…

ICTFOOTPRINT.EU

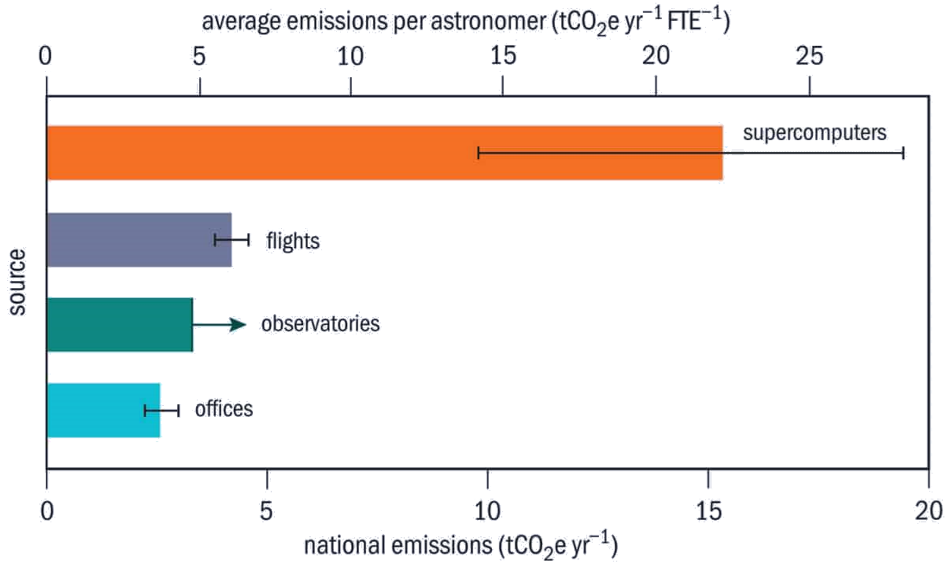
BMW Group moves HPC to Iceland to reduce CO_2 by 3.5K tC02/year

bcs The Chartered Institute for IT

"Carbon footprint, the (not so) hidden cost of HPC."

IOP Publishing

The huge carbon footprint of large-scale computing



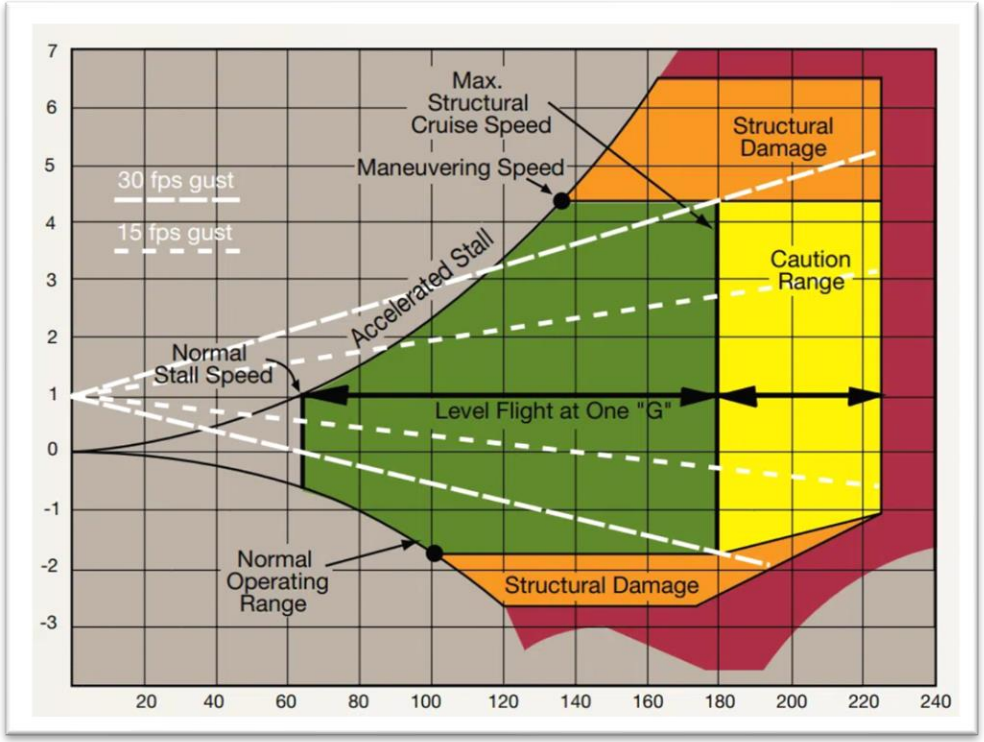
ASIAN SCIENTIST

Powering the world's top 500 supercomputers pumps around 2M tCO_2/year

Source: www.asianscientist.com

The carbon footprint of HPC

CFD for the design of an aircraft

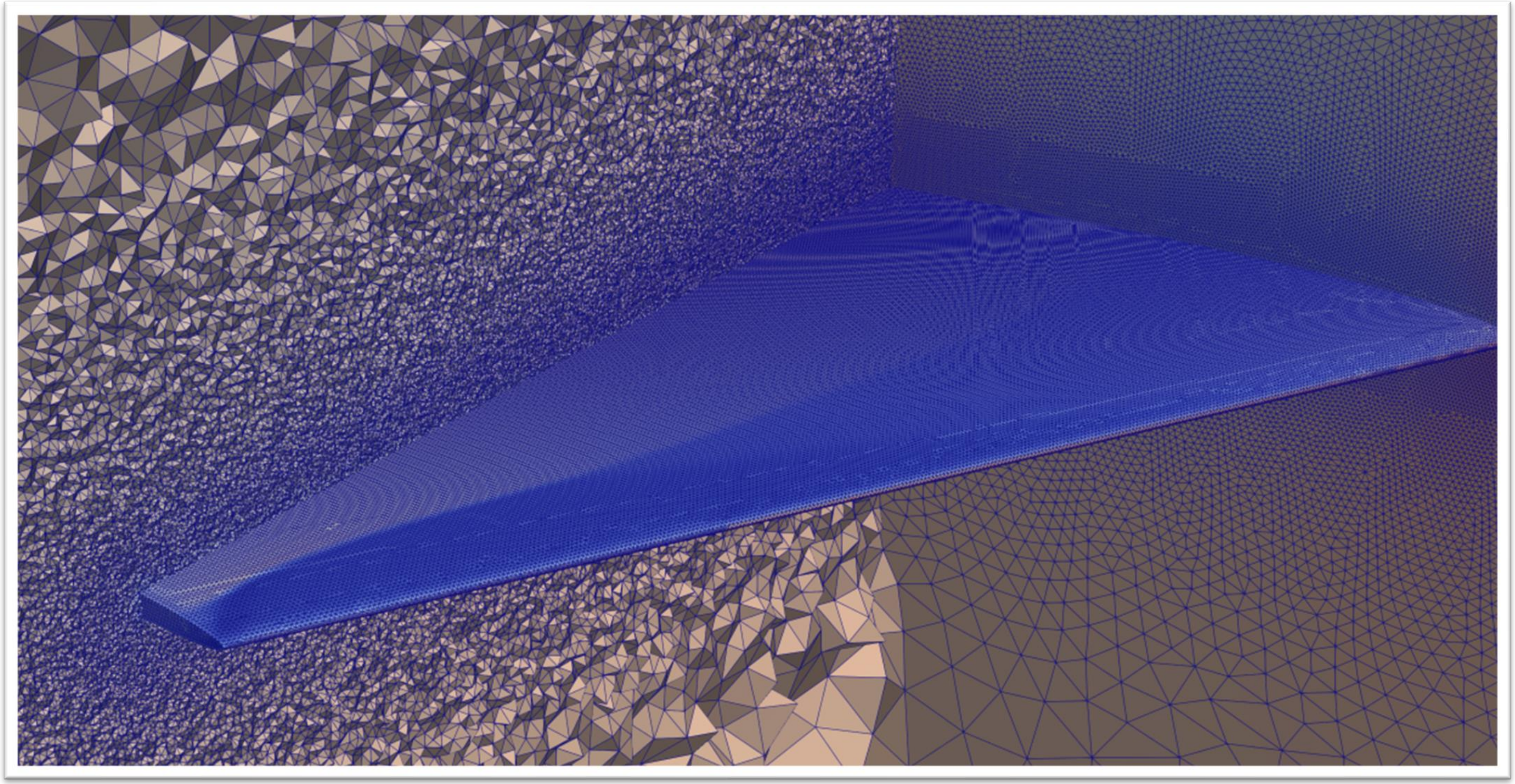


- At least 200 runs to cover the flight envelope
- Each run using ~100M elements (12h using 512 CPU processors)
- 1.7 T CO_2e to perform 200 runs
  - 15.5 MWh
  - 9,840 Km in a passenger car

- NASA estimates that, by 2030, simulations with ~20B elements will be commonplace
- 150 T CO_2e to perform 200 runs
  - 1,350 MWh
  - 858,000 Km in a passenger car (~20 times around the world!)

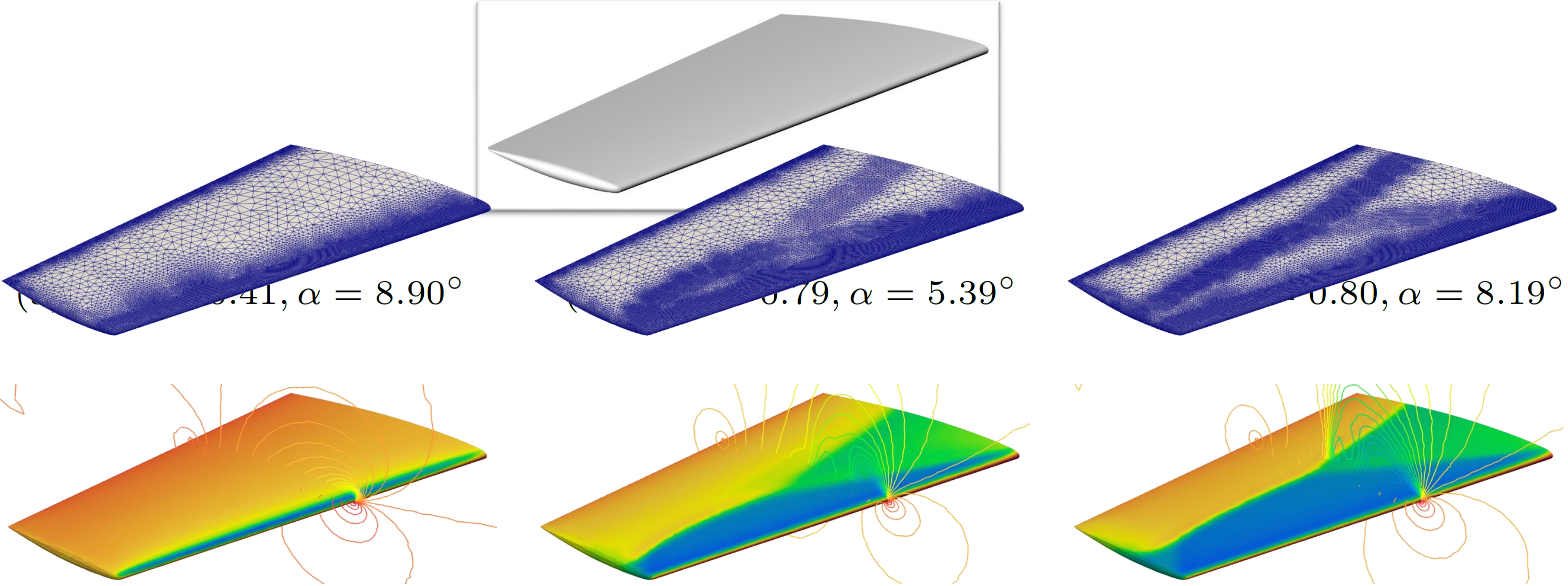
The carbon footprint of HPC

- Current industrial practice involves using over-refined meshes to avoid the requirements of human expertise and the time-consuming process of tailoring meshes



The carbon footprint of HPC

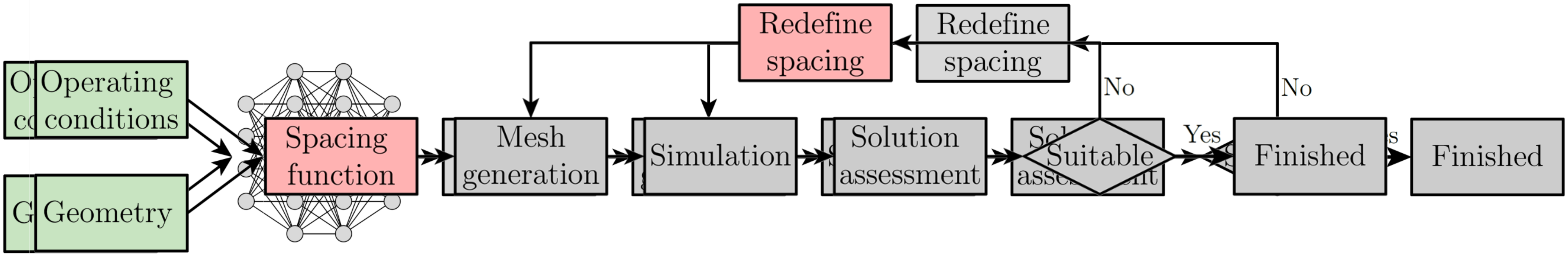
- But from the point of view of simulation efficiency, each operating condition requires a different mesh



Outline

- AI to predict mesh spacing using
  - Mesh sources
  - Background meshes
  - Examples
  - How green is the AI system?
  - Extensions to anisotropic spacing, viscous turbulent flows and CAD integration
- AI to aid mesh adaptation
  - High-order HDG and degree adaptation
  - Examples
- Concluding remarks

AI system to predict mesh spacing



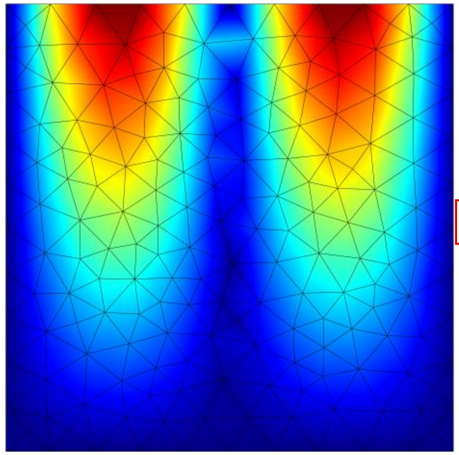
- **Objective: Develop an AI system to predict mesh spacing**

- Increase speed and automation in the mesh generation process
- Accelerate mesh independent studies
- Speed up design processes
- Reduce the carbon footprint of simulations
- Preserve previous knowledge

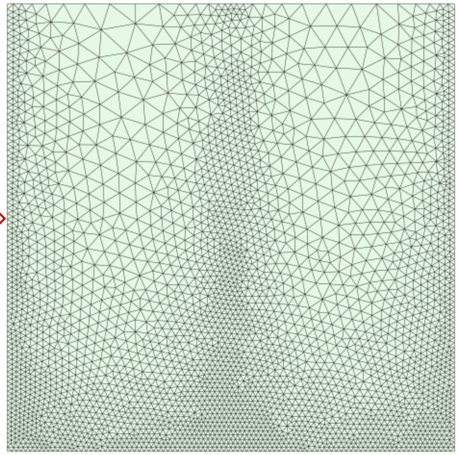
AI system to predict mesh spacing

- How is a spacing function usually defined?

- Background mesh
  - A (discrete) nodal spacing function is defined
  - The spacing at any point is interpolated from the nodal spacing function in the (coarse) background mesh



Spacing function



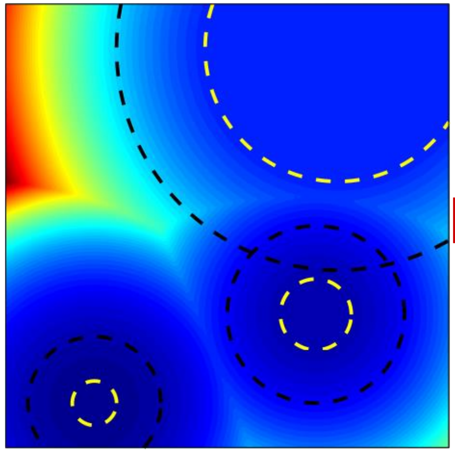
Mesh

AI system to predict mesh spacing

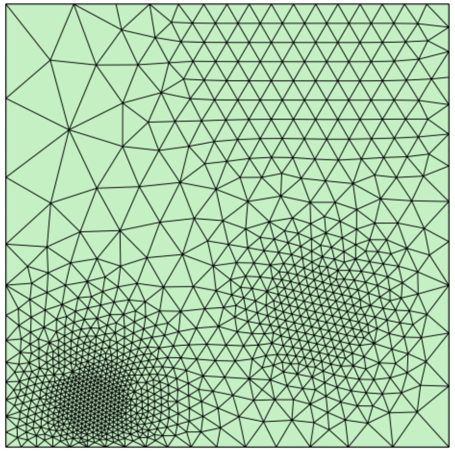
- How is a spacing function usually defined?

- Mesh sources (point, lines, planes, etc)
  - A point source is defined by a position, a desired spacing and a radius of influence
  - The spacing at any point is calculated as the minimum spacing defined by all the sources

$$\boxed{h(\boldsymbol{x};\boldsymbol{x}_c,h_0,r_1,r_2)=\begin{cases} h_0 & \text{if } r\leq r_1 \\ \min\left\{h_\infty, h_0\exp\left[\ln(2)\left(\frac{r-r_1}{r_2-r_1}\right)\right]\right\} & \text{otherwise}\end{cases}}$$



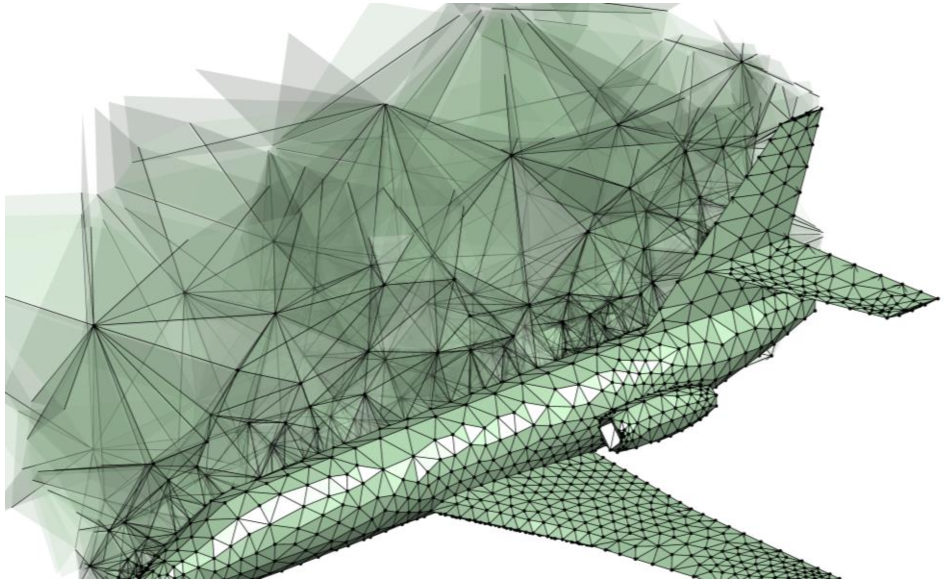
Spacing function



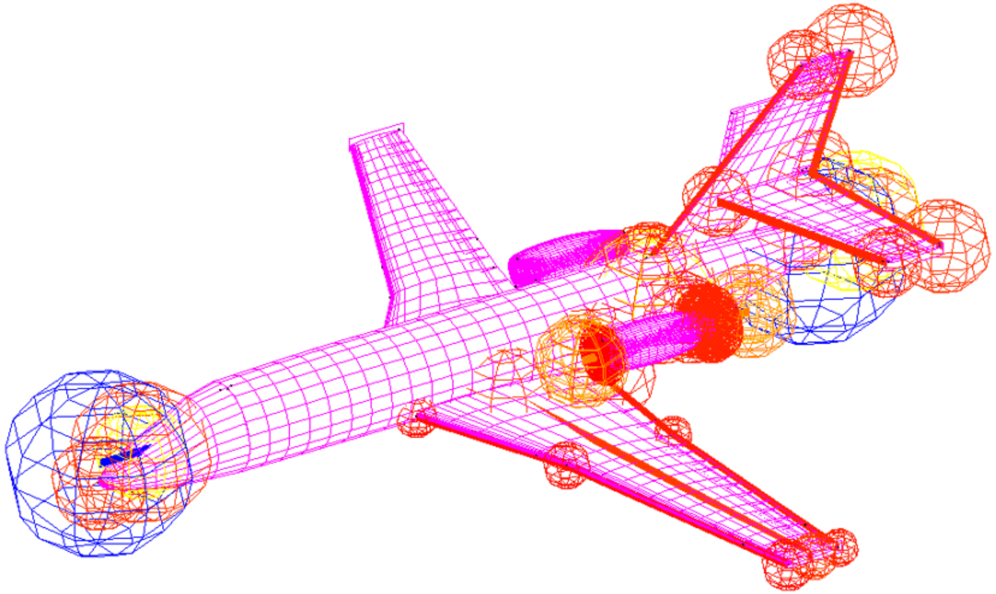
Mesh

AI system to predict mesh spacing

- How is a spacing function usually defined?



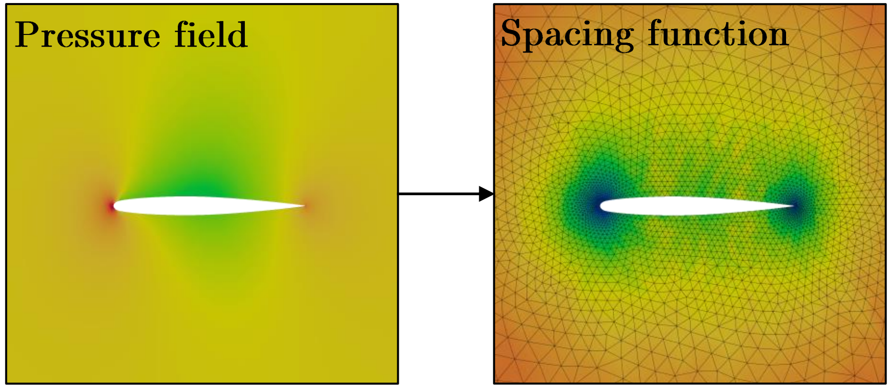
- Background mesh
  - A (discrete) nodal spacing function is defined
  - The spacing at any point is interpolated from the nodal spacing function in the (coarse) background mesh



- Mesh sources (point, lines, planes, etc)
  - A point source is defined by a position, a desired spacing and a radius of influence
  - The spacing at any point is calculated as the minimum spacing defined by all the sources

AI system to predict mesh spacing (using a background mesh)

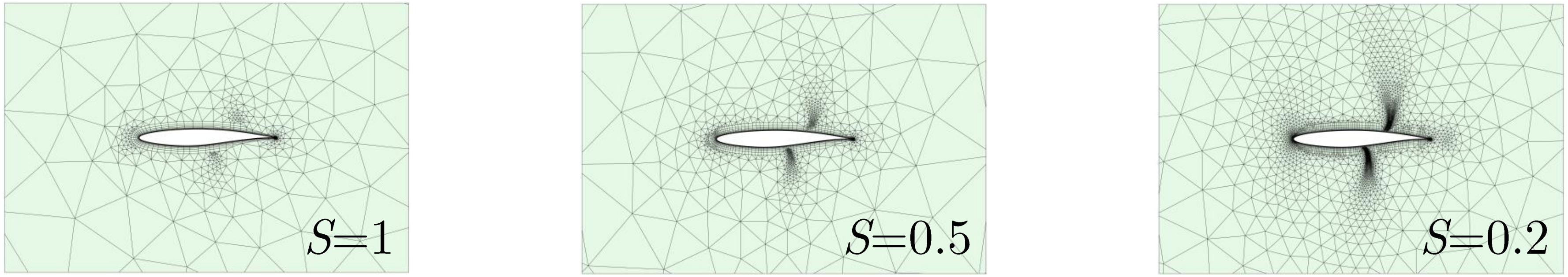
- Generate a spacing function from a given solution



- Spacing at a node $\boldsymbol{x}_i$ along a direction $\boldsymbol{\beta}$ is related to the Hessian of a key variable

$$\delta_{i,\boldsymbol{\beta}}^2 \left(\sum_{k,l=1}^{\mathtt{n_{sd}}} (H_i)_{kl} \beta_k \beta_l \right) = K \qquad \delta_i = \begin{cases} \delta_{min} & \text{if } \lambda_{i,max} > K/\delta_{min}^2, \\ \delta_{max} & \text{if } \lambda_{i,max} < K/\delta_{max}^2, \\ \sqrt{K/\lambda_{i,max}} & \text{otherwise,} \end{cases} \qquad K = S^2 \delta_{min}^2 \lambda_{max}$$

  - $\{\lambda_{i,j}\}_{j=1,\dots,\mathtt{n_{sd}}}$ eigenvalues of the Hessian at node $\boldsymbol{x}_i$ and $\lambda_{i,max} = \max_{j=1,\dots,\mathtt{n_{sd}}}\{\lambda_{i,j}\}$
  - $\lambda_{max} = \max_{i=1,\dots,\mathtt{n_{no}}}\{\lambda_{i,max}\}$ is the maximum for all nodes in the mesh
  - S is a scaling factor

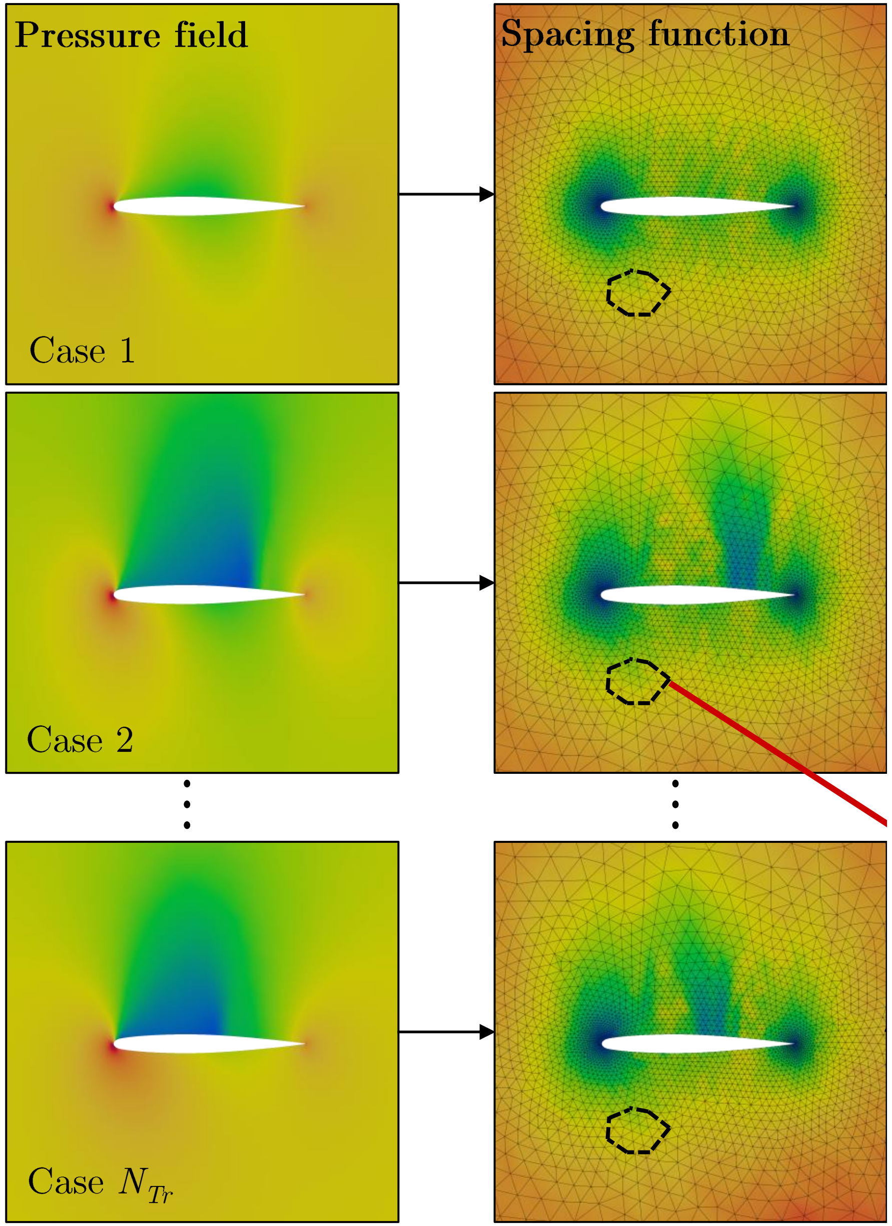


AI system to predict mesh spacing (using a background mesh)

- Generate a spacing function from a given solution

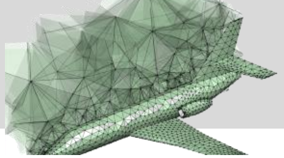
- Interpolate the spacing onto a background mesh

- Take a conservative approach to interpolation

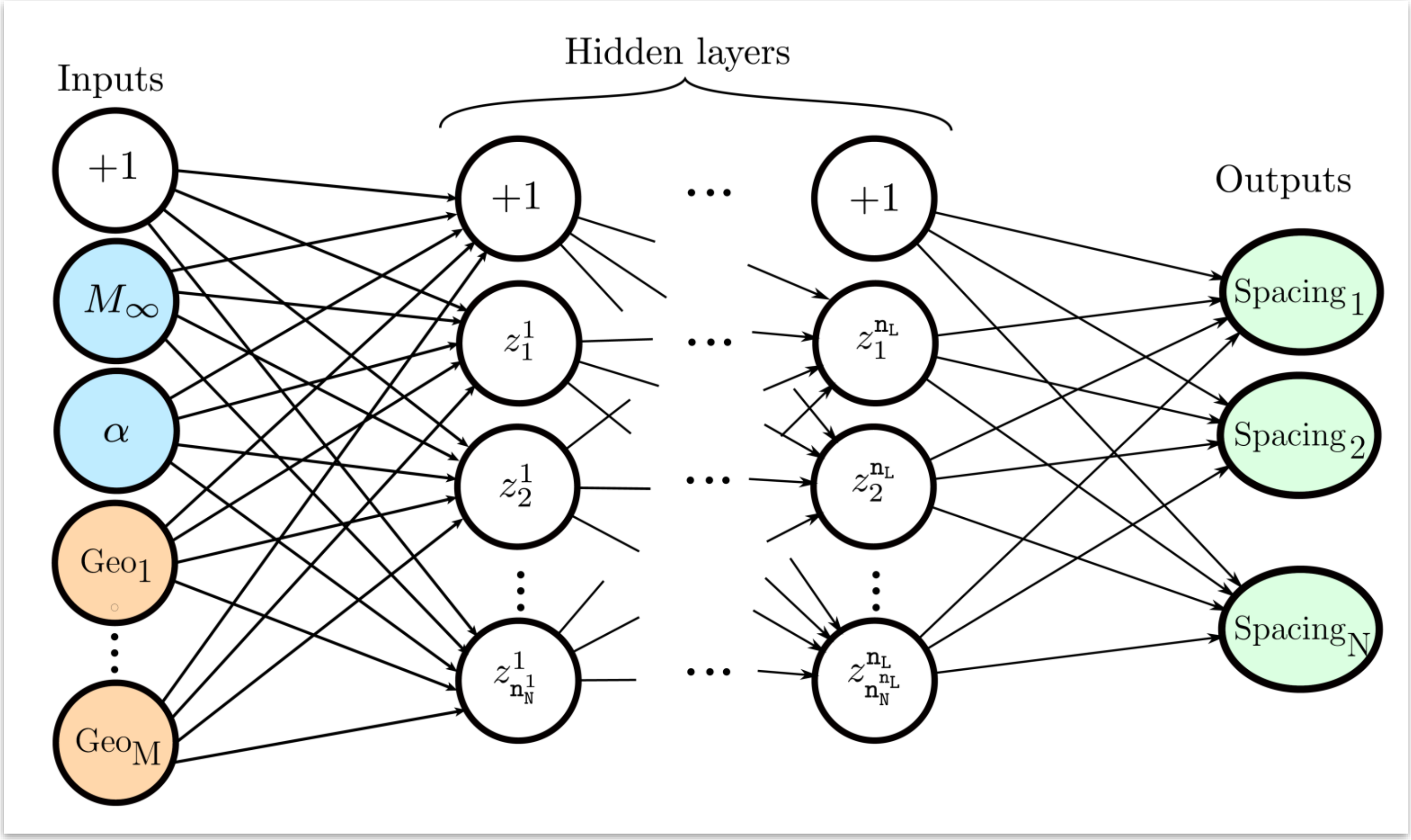


C Lock, O Hassan, RS, J Jones, Predicting the near-optimal mesh spacing for a simulation using machine learning, IMR, 20

AI system to predict mesh spacing (using a background mesh)



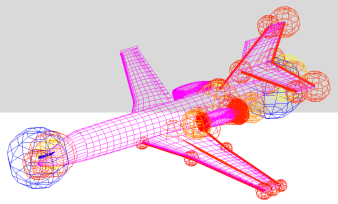
- Neural network architecture



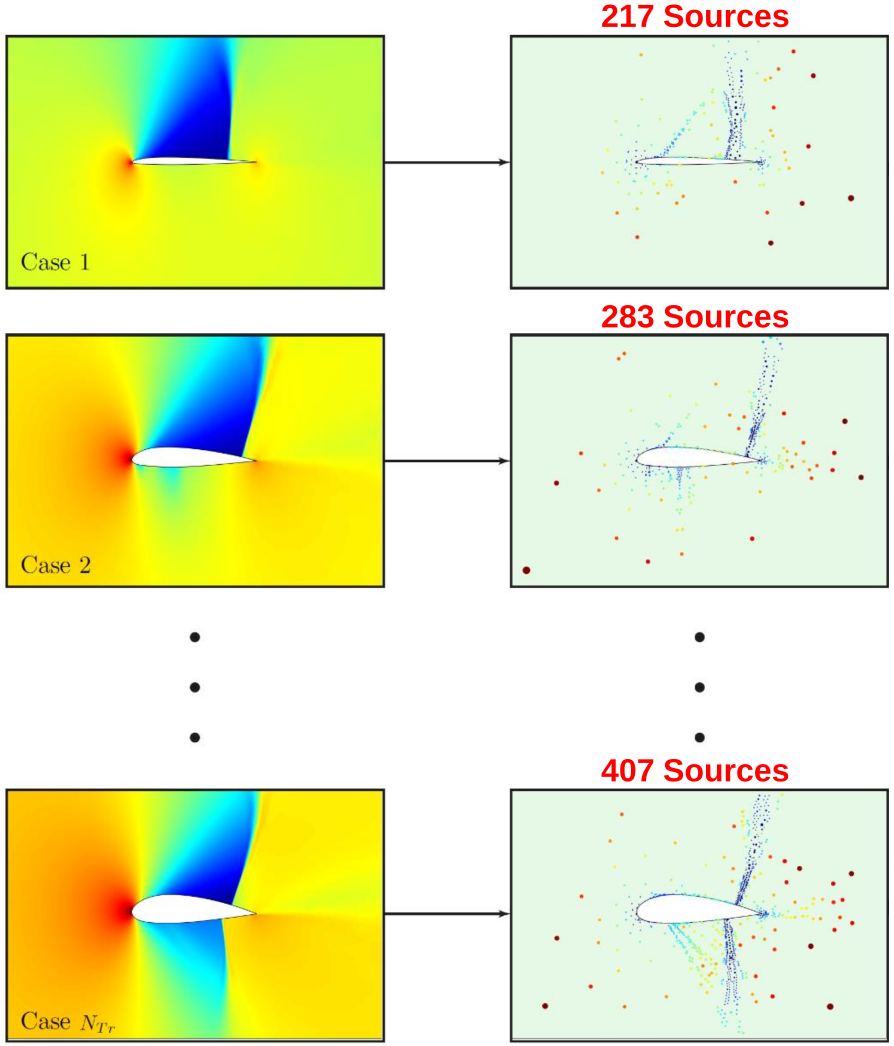
- Each output involves a spacing
- Requires mesh morphing for variable geometric configurations

C Lock, O Hassan, RS, J Jones, Predicting the near-optimal mesh spacing for a simulation using machine learning, IMR, 2023

AI system to predict mesh spacing (using sources)

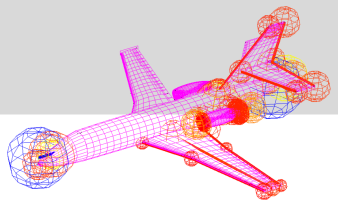


- Step 1 – Generate point sources from a given solution
- Step 2 – Generate global sources from sets of local sources

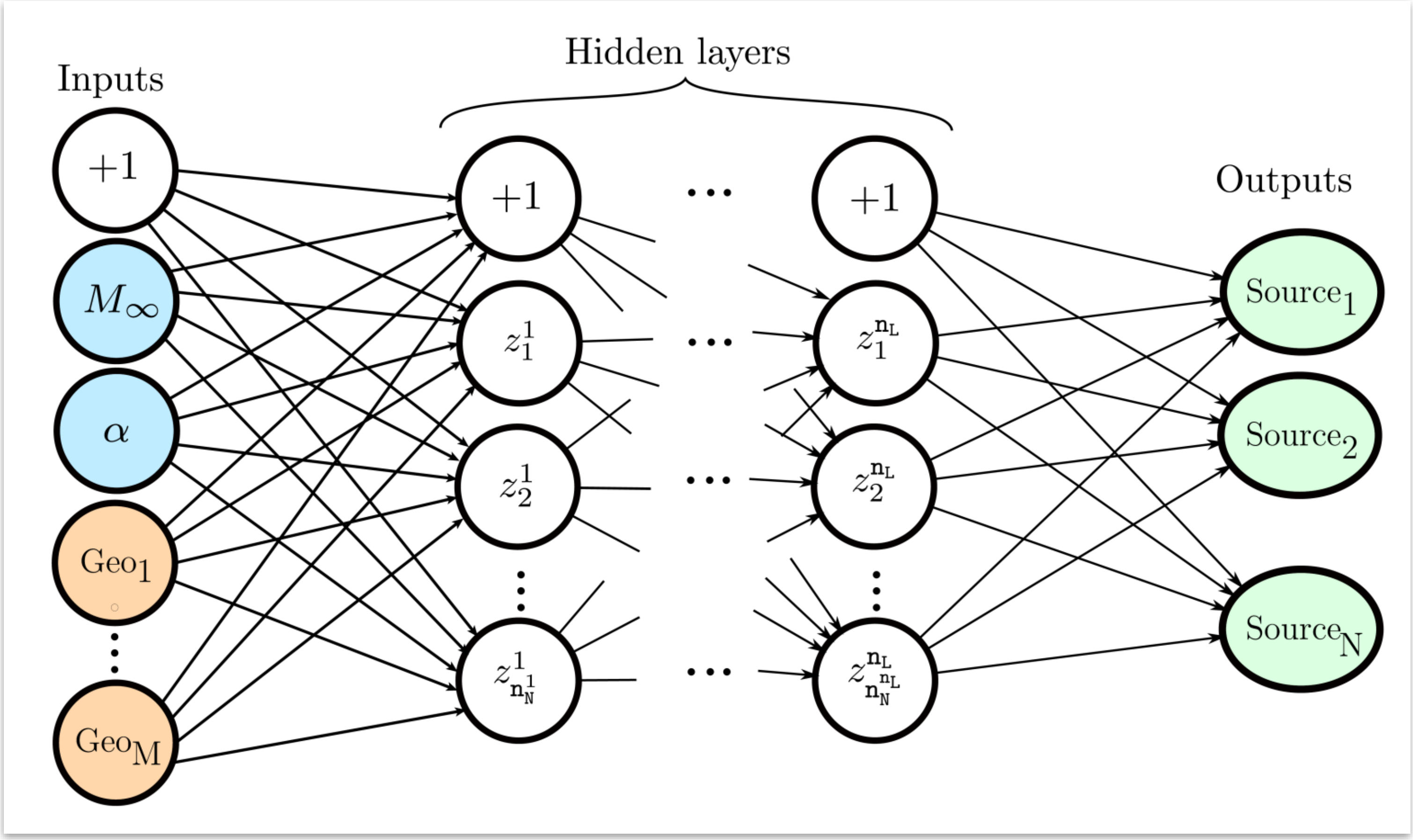


C Lock, O Hassan, RS, J Jones, Meshing using neural networks for improving the efficiency of computer modelling, Engineering with Computers, 2023

AI system to predict mesh spacing (using sources)



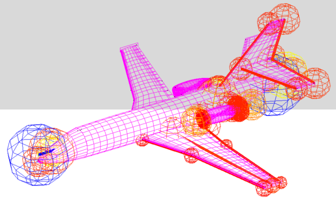
- Neural network architecture



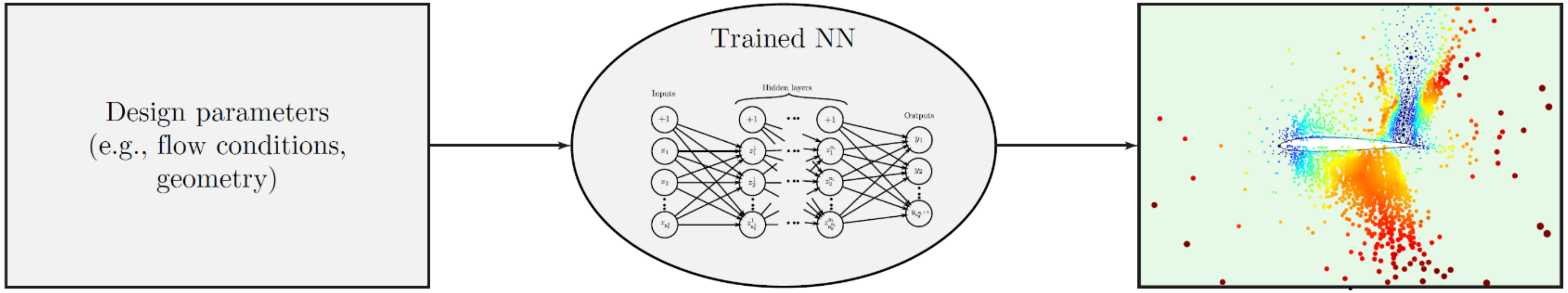
- Each output involves a position (3 coordinates), a spacing and a radius of influence.

C Lock, O Hassan, RS, J Jones, Meshing using neural networks for improving the efficiency of computer modelling, Engineering with Computers, 2023

AI system to predict mesh spacing (using sources)



- Step 3 – Use the trained NN to predict the sources characteristics

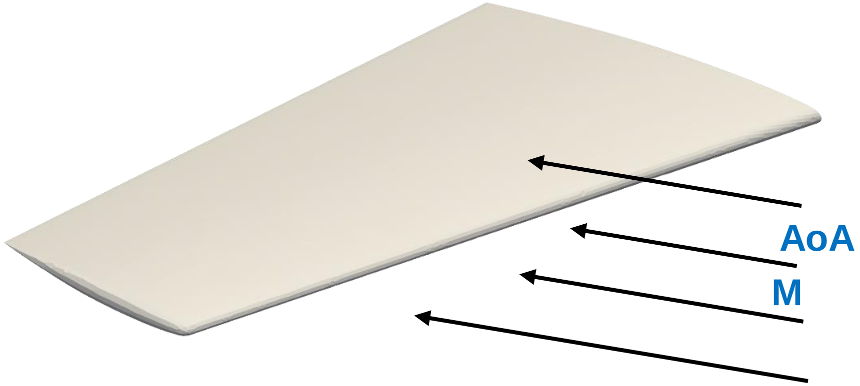


- Step 4 – Reduce the global set of sources

C Lock, O Hassan, RS, J Jones, Meshing using neural networks for improving the efficiency of computer modelling, Engineering with Computers, 2023

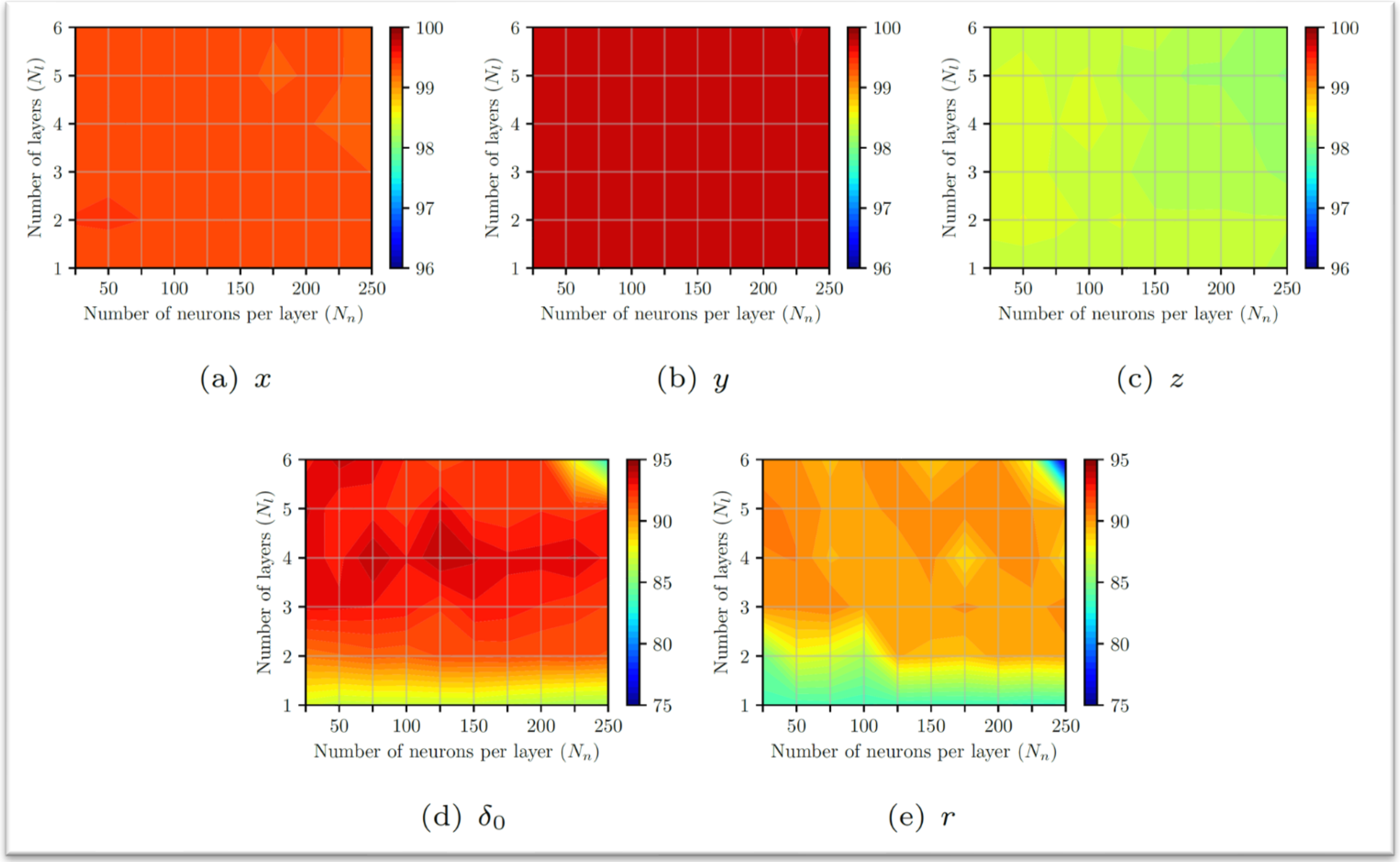
Examples

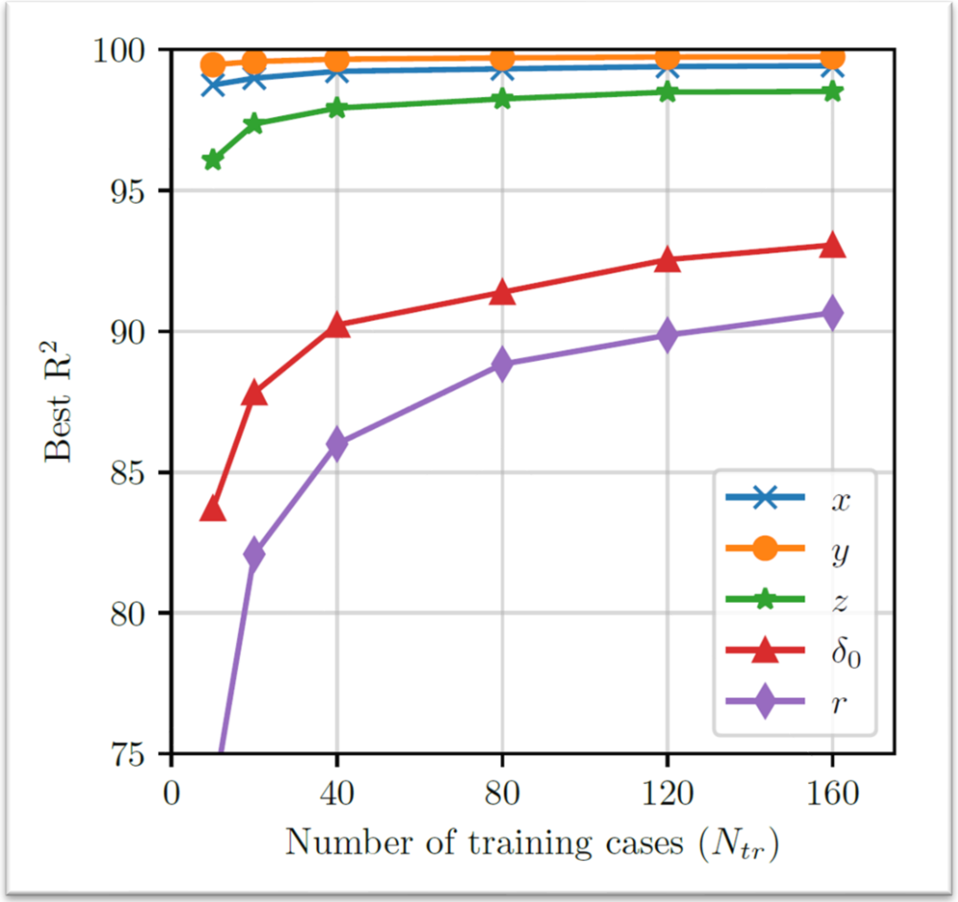
- ONERA M6 wing – Variable flow conditions
  - Mach [0.3, 0.9]
  - Angle of Attack [0, 12]
- Solution from an inviscid compressible flow solver
- 10, 20, 40, 80, 120, 160 training cases, with 100 test cases
- Hyperparameters are tuned
  - Number of hidden layers – 1, 2,…,6
  - Neurons per layer – 25, 50, 75, …, 200, 225, 250
- Using sources
  - 19,345 global point sources
- Using background mesh
  - 14,179 nodes



Examples

- ONERA M6 wing – Variable flow conditions
- NN training

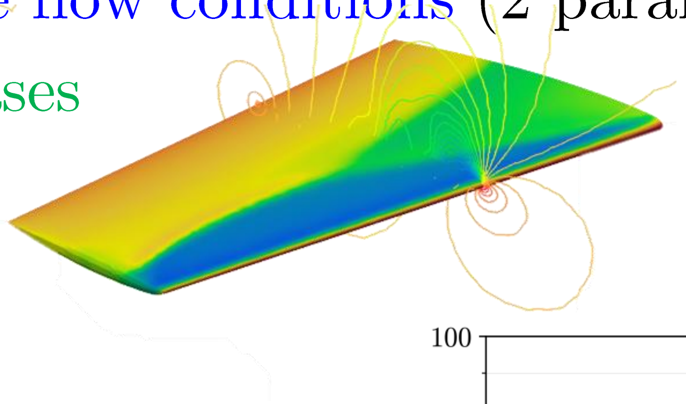




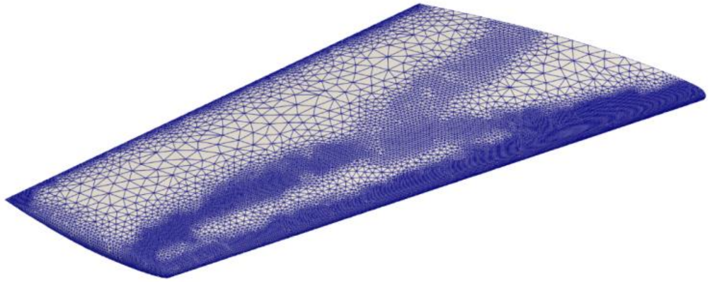
- Shallow networks with 1 or 2 layers are enough to provide accurate predictions
- Spacing and radius of influence are more difficult to predict

Examples

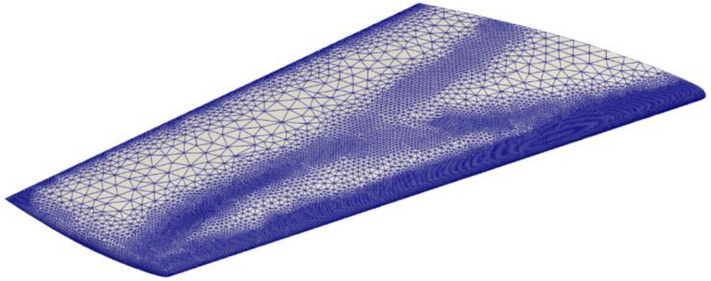
- ONERA M6 wing – Variable flow conditions (2 parameters) – 160 training cases
- Prediction for unseen test cases
  - M=0.79, AoA=5.39°



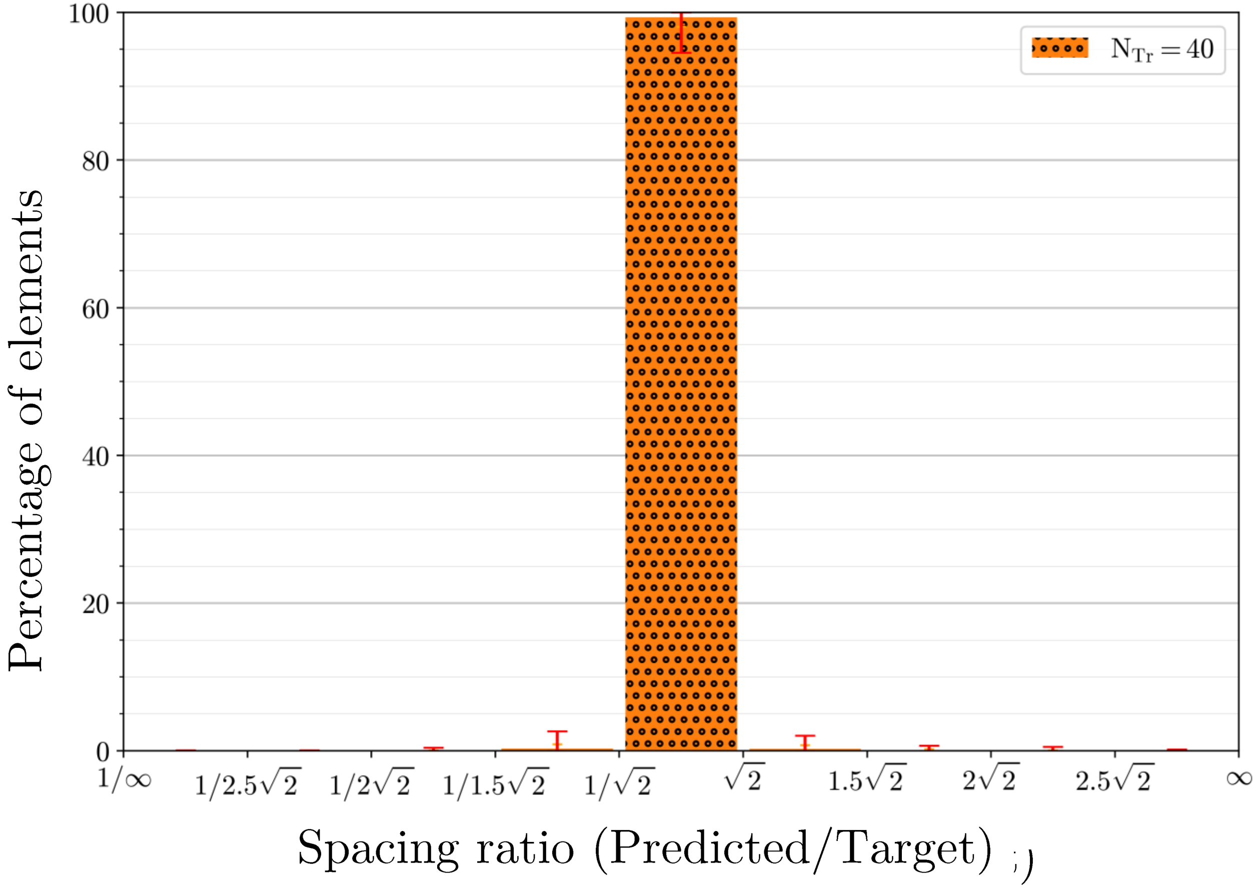
Target



ML Prediction

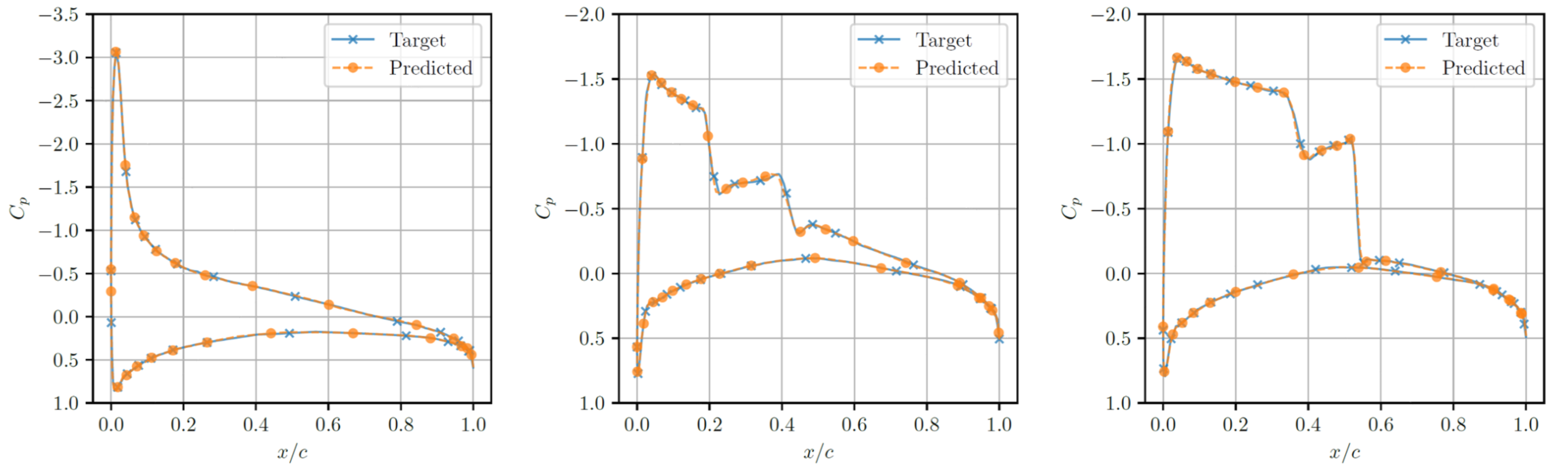


- Using sources, the spacing at 72% of the points are predicted within 5% of the target
- Using a background mesh, the spacing at 94% of the points are predicted within 5% of the target



Examples

- ONERA M6 wing – Variable flow conditions (2 parameters) – 160 training cases
- Suitability of predicted meshes to perform simulations

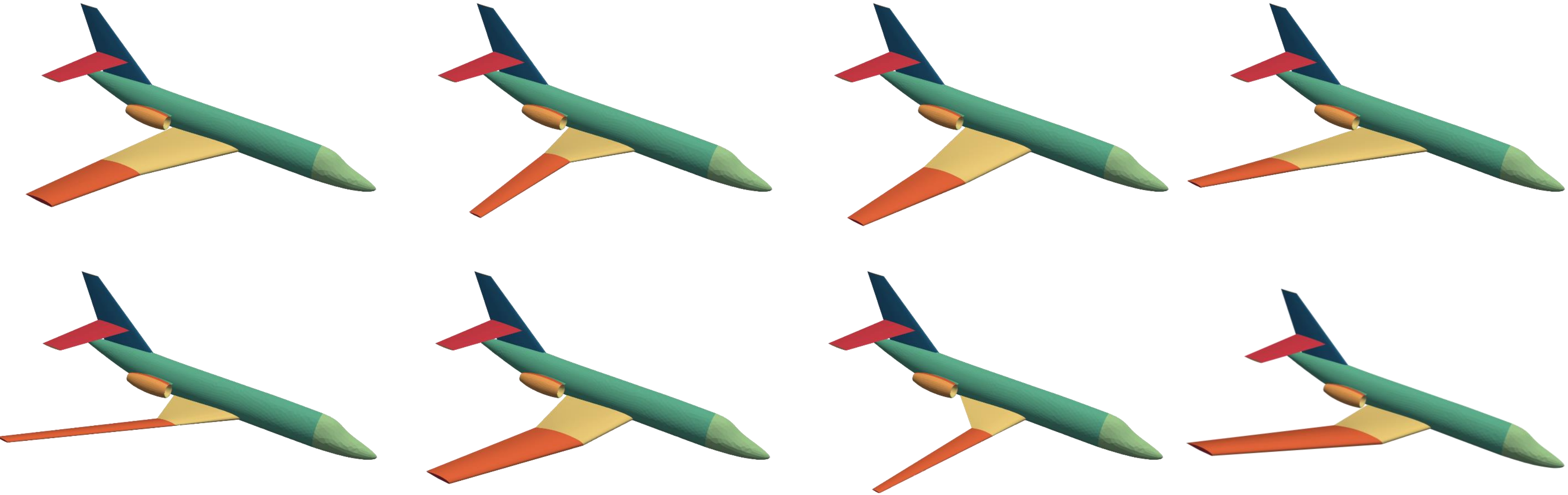


(a) $M_\infty = 0.41, \alpha = 8.90°$ (b) $M_\infty = 0.79, \alpha = 5.39°$ (c) $M_\infty = 0.80, \alpha = 8.19°$

	$M_\infty = 0.41, \alpha = 8.90°$		$M_\infty = 0.79, \alpha = 5.39°$		$M_\infty = 0.80, \alpha = 8.19°$	
	Target	Prediction	Target	Prediction	Target	Prediction
C_L	0.605	0.603	0.469	0.468	0.722	0.723
C_D	0.0342	0.0340	0.0289	0.0287	0.0828	0.0828

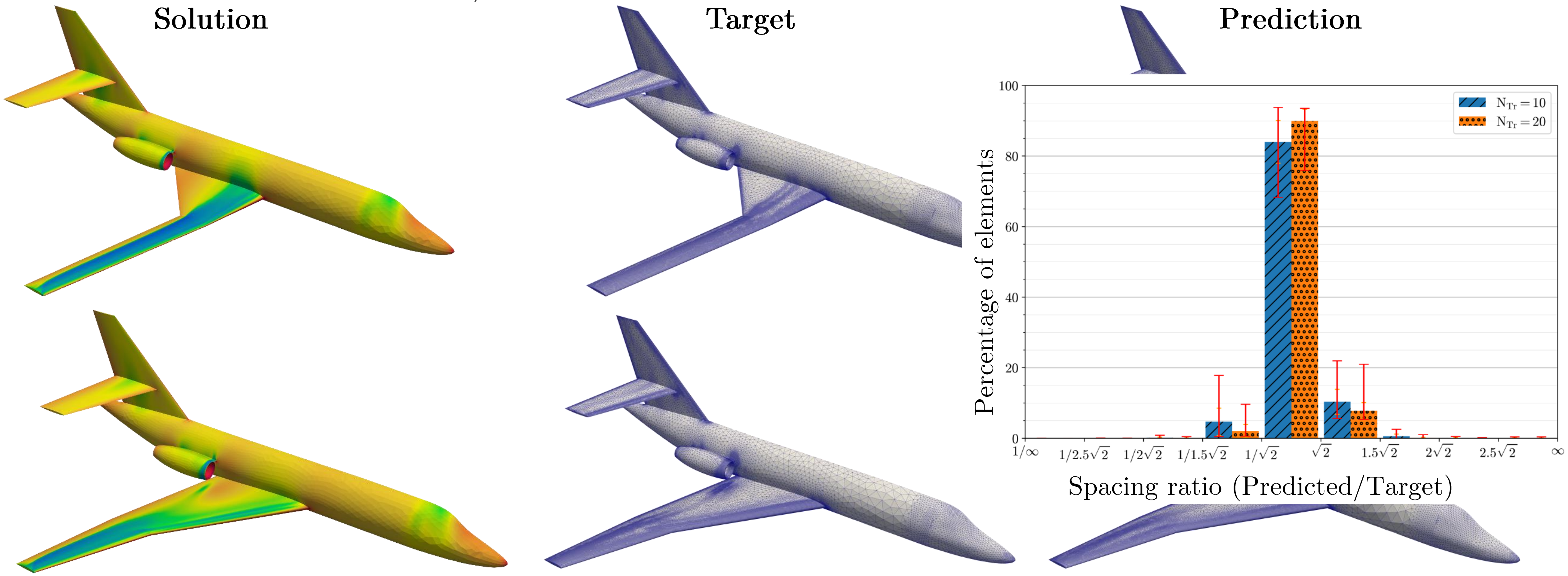
Examples

- Full aircraft – Variable geometry (11 parameters)
  - Spacing prediction using a background mesh
  - Flow conditions – M=0.8, AoA=2°



Examples

- Full aircraft – Variable geometry (11 parameters) – 10 vs 20 training cases
  - Spacing prediction using a background mesh
  - Flow conditions – M=0.8, AoA=2^{o}

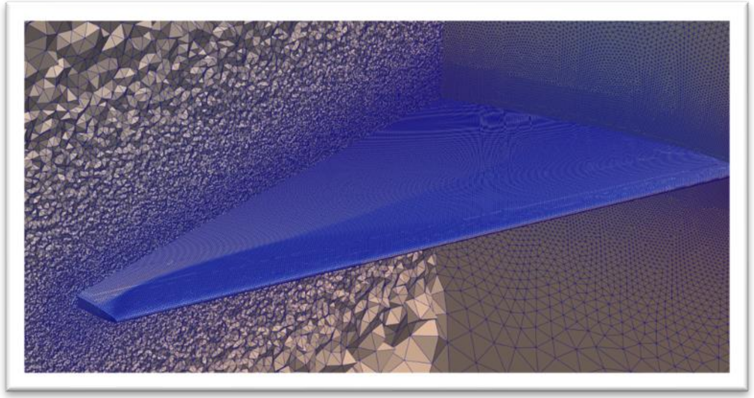


- Spacing at 83% of the points are predicted within 5% of the target

How green is the AI system?

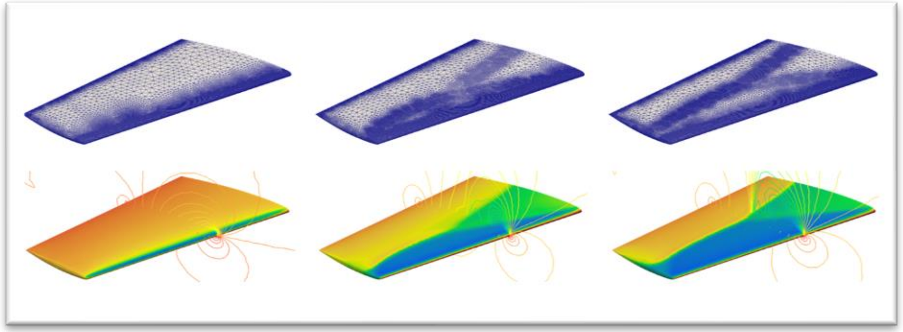
- Carbon footprint for the parametric study using a fixed mesh

Task	Wall clock (H)	Carbon (Kg CO_2e)	Energy (MWh)
Mesh generation	1.0	3.61×10^{-3}	5.89×10^{-5}
CFD solution	3,432.10	527.17	2.28
Total	3,433.0	527.17	2.28



- Carbon footprint for the parametric study using AI predicted meshes

Task	Wall clock (H)	Carbon (Kg CO_2e)	Energy (MWh)
Mesh generation	23.8	0.32	1.40×10^{-3}
CFD solution	143.0	12.36	5.35×10^{-2}
Total	166.8	12.68	0.055

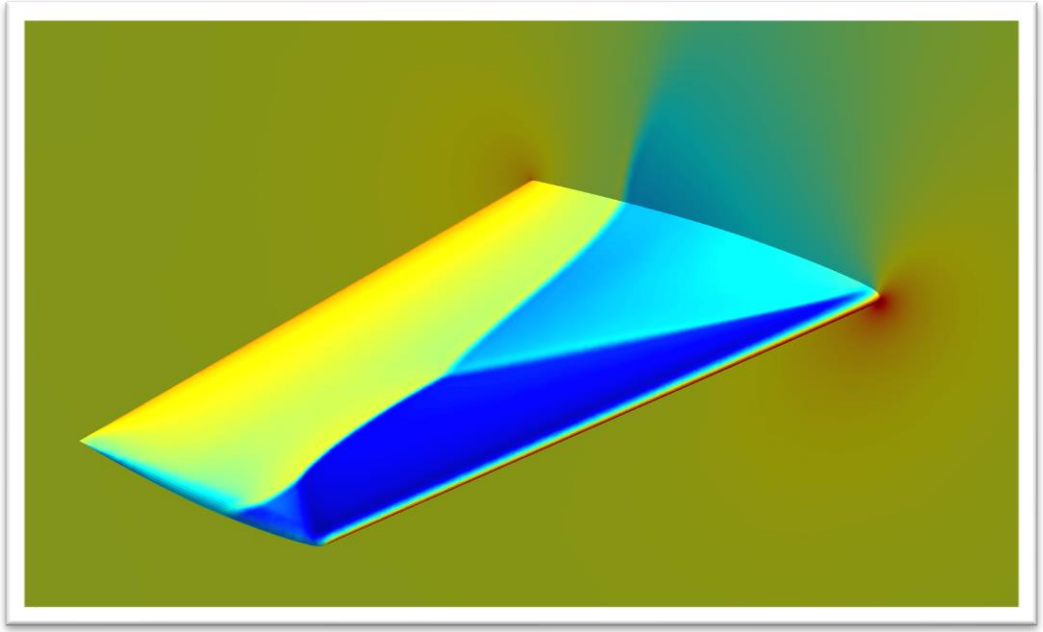


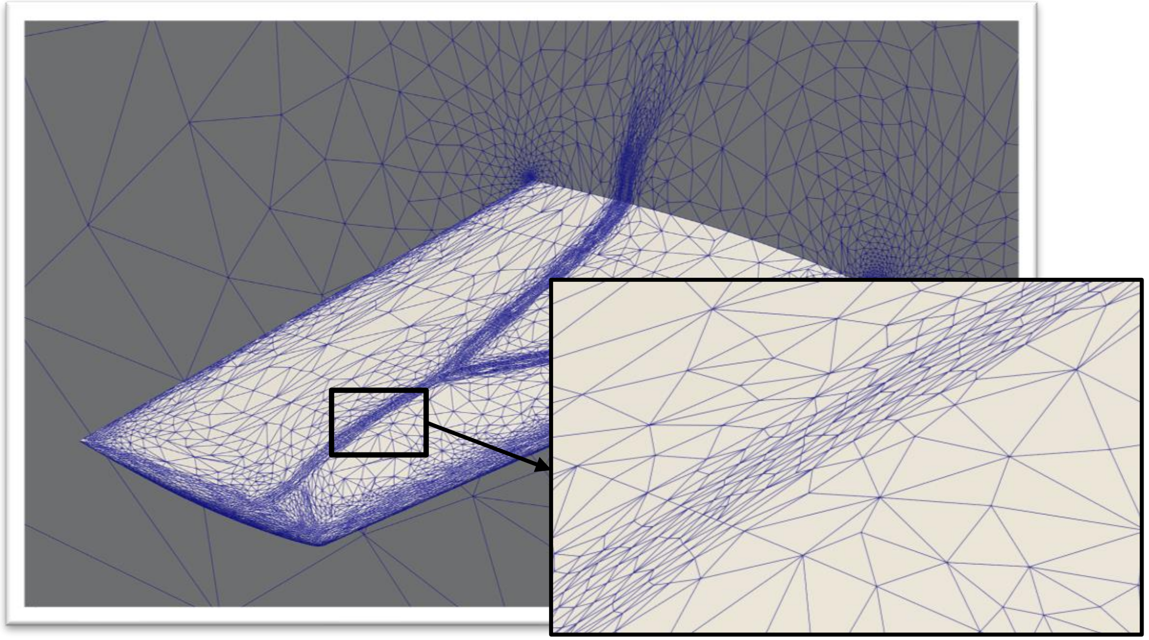
TOTAL carbon footprint is more than 35 times lower

C Lock, O Hassan, RS, J Jones, Meshing using neural networks for improving the efficiency of computer modelling, Engineering with Computers, 2023

Extension to anisotropic spacing

- Isotropic spacing is suboptimal when there is a clear directionality in the solution field

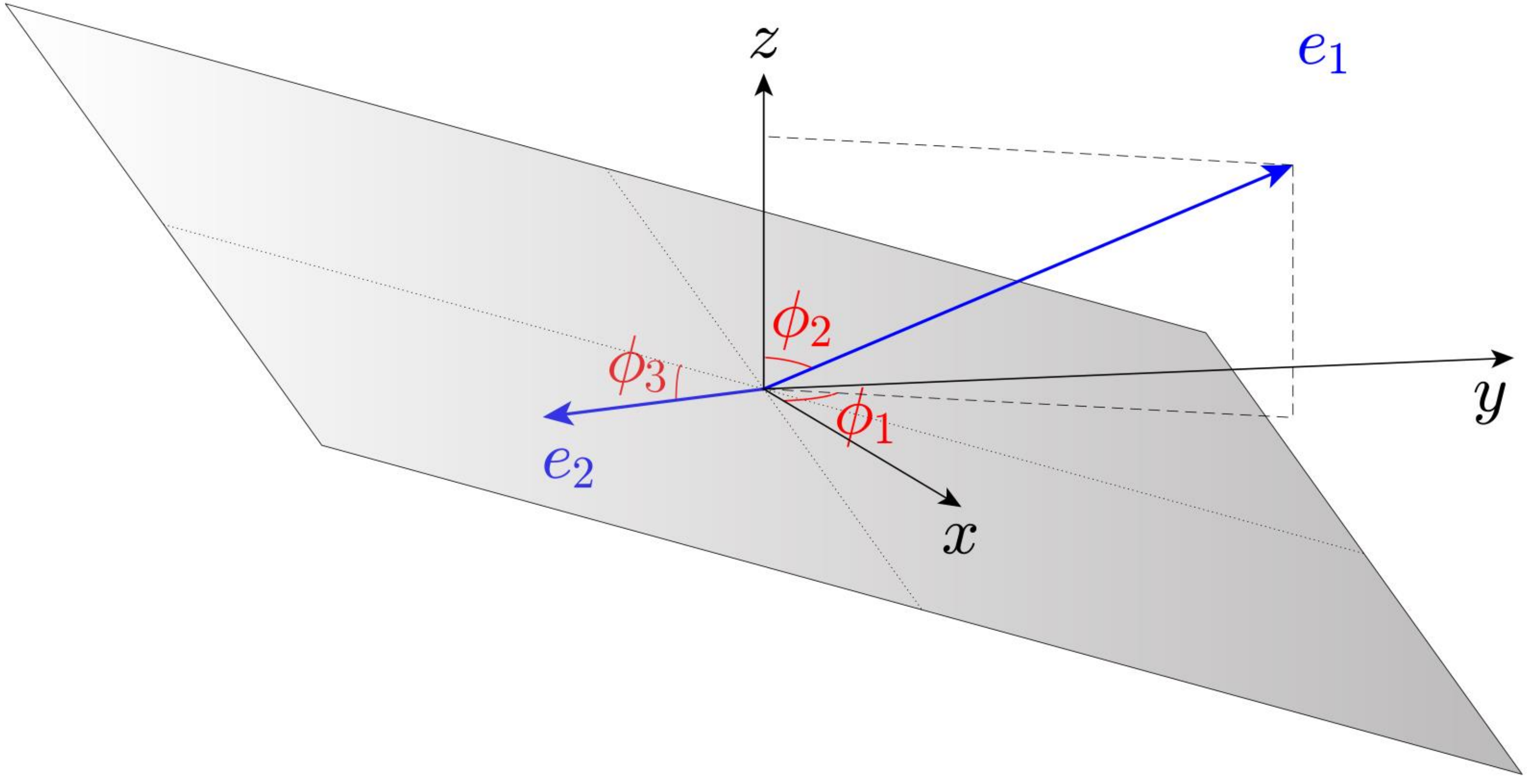




- Anisotropic spacing is described using a metric tensor at each node:
  - Three orthogonal directions (eigenvectors of the Hessian of a key variable)
  - Three spacings (eigenvalues of the Hessian of a key variable)

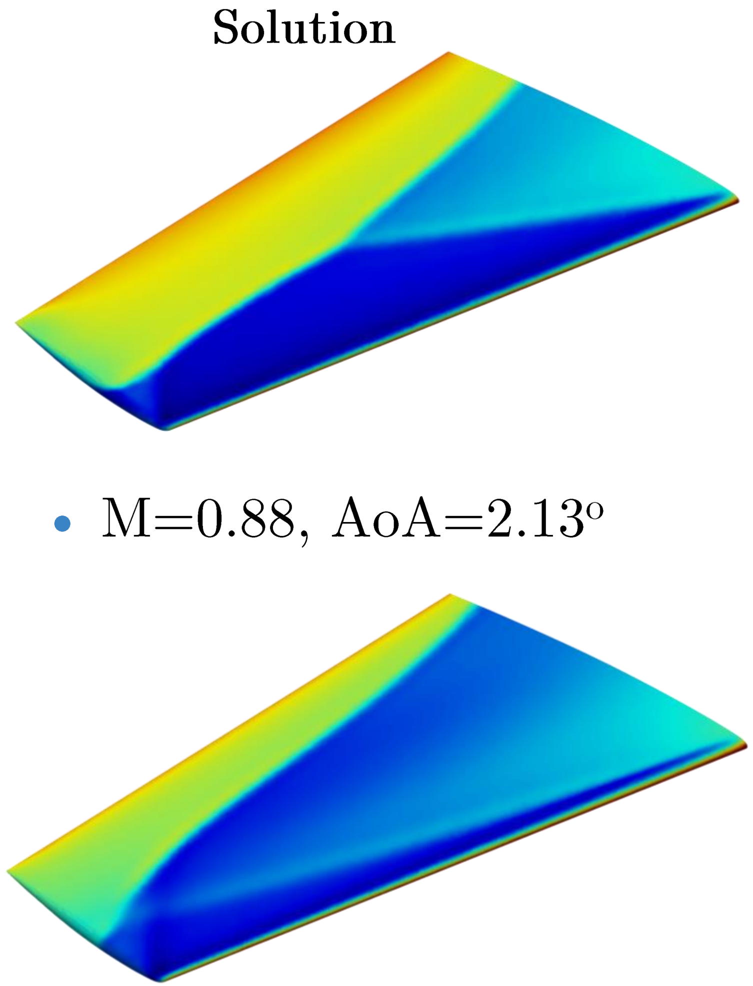
Extension to anisotropic spacing

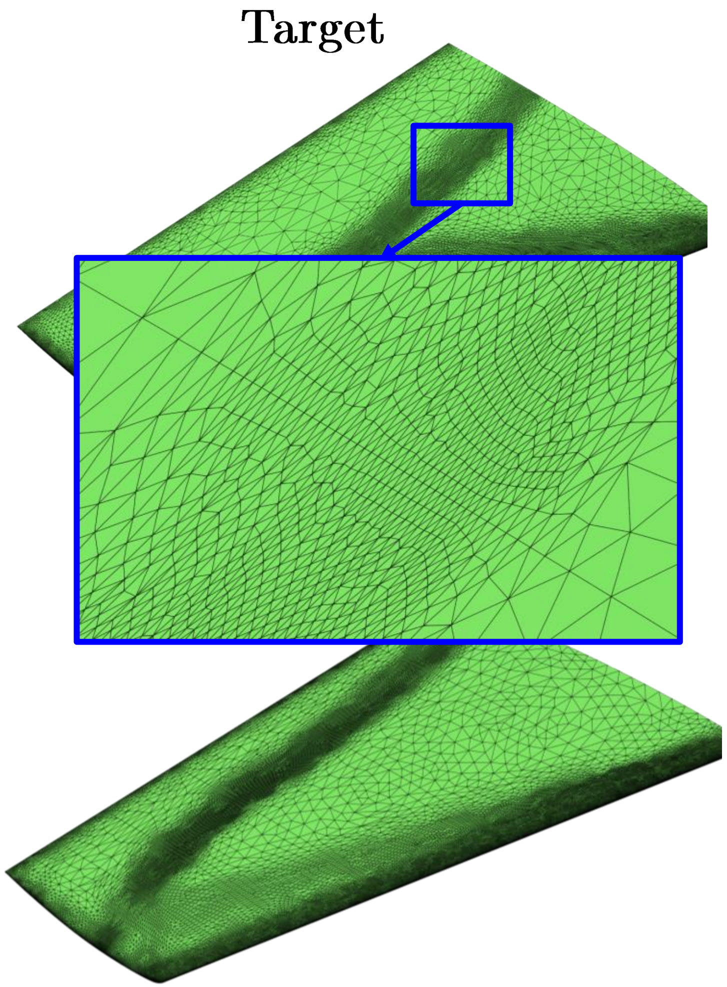
- The AI system needs to predict a metric tensor at each point
- Our strategy consists of
  - expressing the first direction (minimum spacing) in spherical coordinates (two angles)
  - expressing the second direction in polar coordinates in the normal plane to the first direction (one angle)
- The third direction is given by the orthogonality property (no prediction needed)

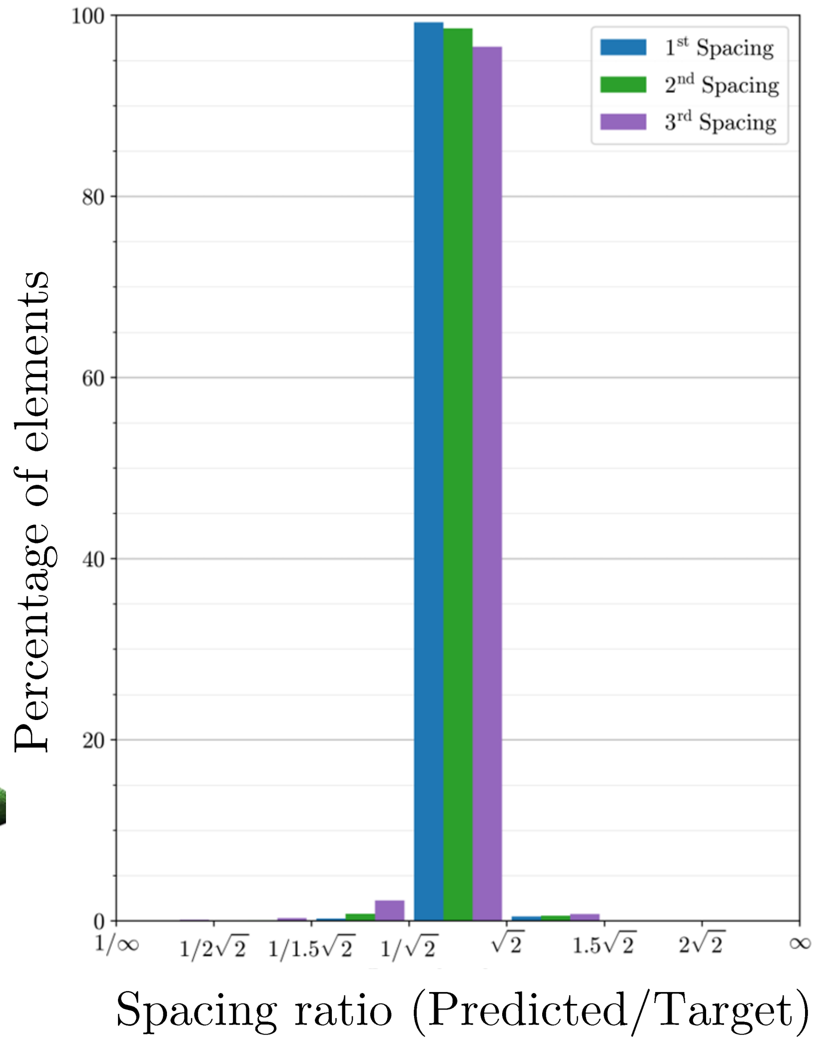


Examples

- ONERA M6 wing – Variable flow conditions (2 parameters) – 80 training cases
- Prediction for unseen test cases
  - M=0.89, AoA=7.12°
  - M=0.88, AoA=2.13°

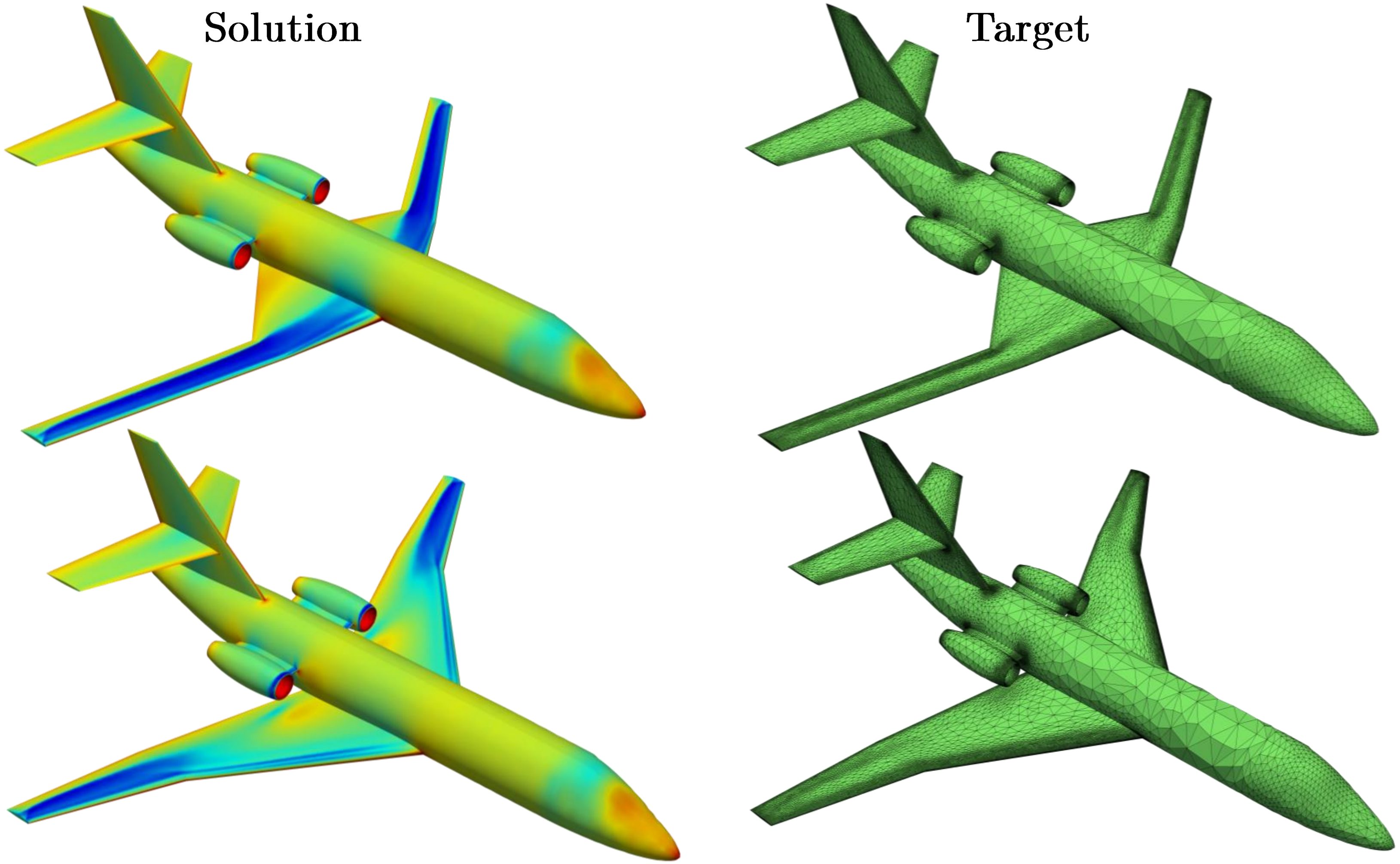


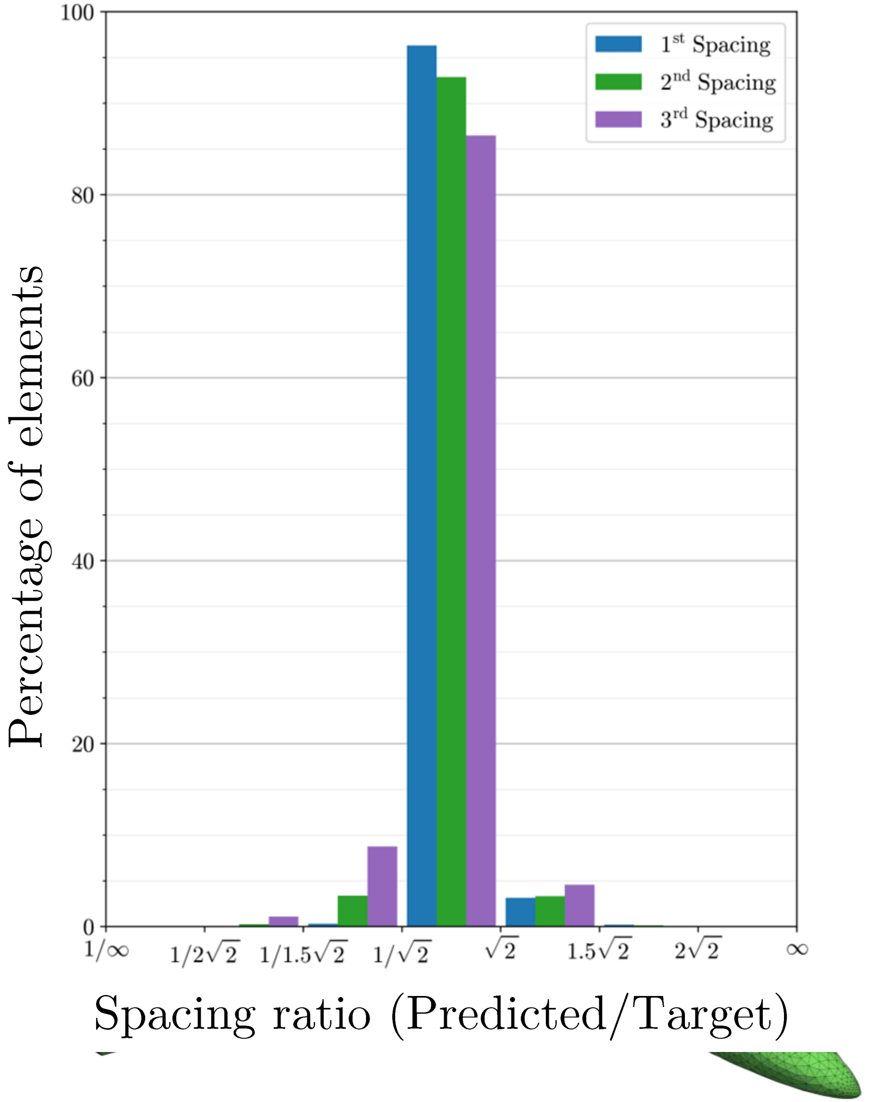




Examples

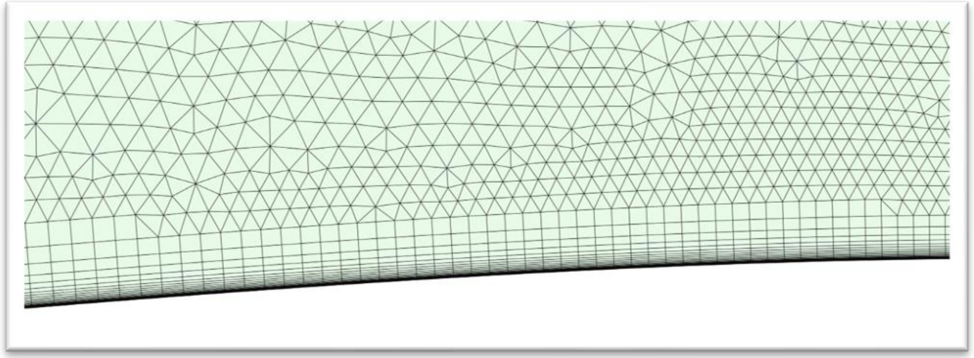
- ONERA M6 wing – Variable geometry (11 parameters) – 40 training cases
- Flow conditions – M=0.8, AoA=2°
- Prediction for unseen test cases

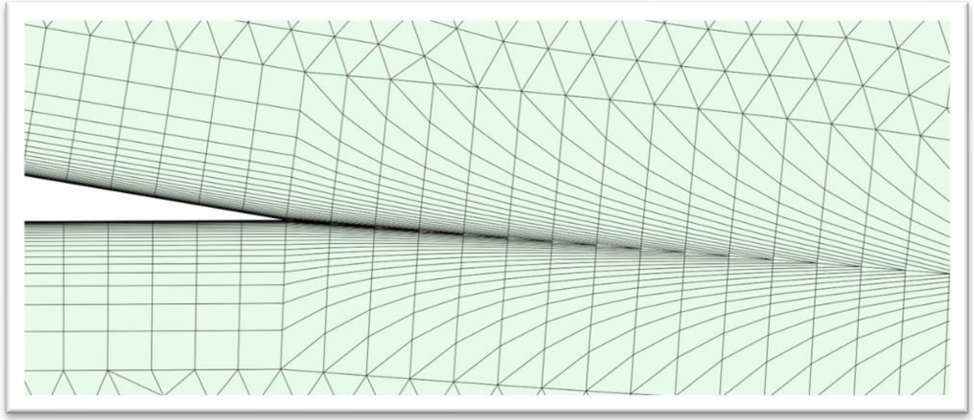




Extension to viscous turbulent flows

- Challenges induced by highly stretched elements (e.g., boundary layer, shear layer)





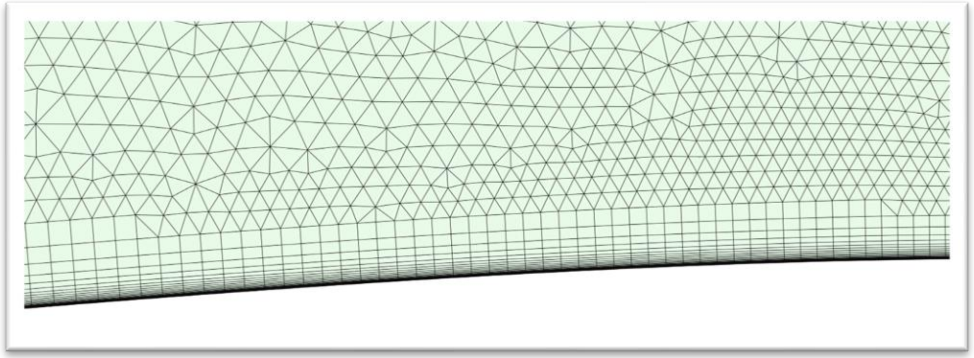
- The computation of the gradient (FEM, FV, FD) involves dividing by an area (for high Reynolds this could be ~10^{-9})

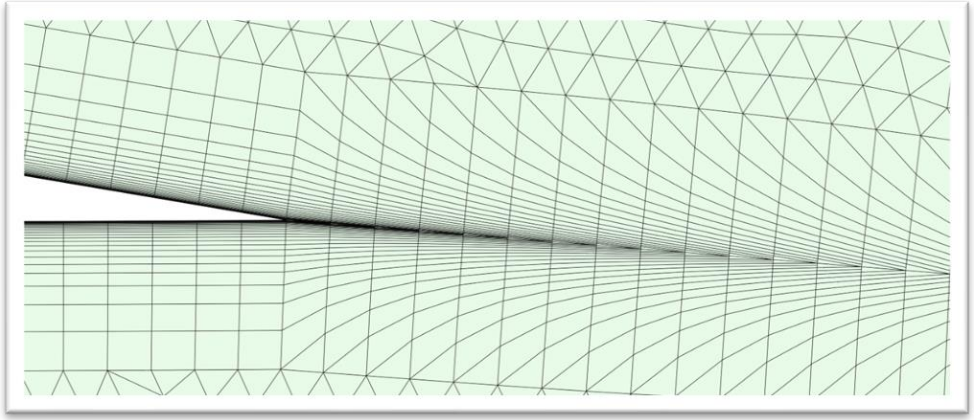
$$\nabla \sigma_i \approx \frac{\displaystyle\sum_{\Omega_e \in \mathcal{P}_i} |\Omega_e| \nabla \sigma_e}{\displaystyle\sum_{\Omega_e \in \mathcal{P}_i} |\Omega_e|}$$

$$\nabla \sigma_i \approx \frac{1}{|V_i|} \left[\sum_{j \in \mathcal{N}_i} \frac{1}{2} (\sigma_i + \sigma_j) \boldsymbol{C}_{ij} + \sum_{j \in \mathcal{N}_i^{\partial}} \sigma_i \boldsymbol{D}_{ij} \right]$$

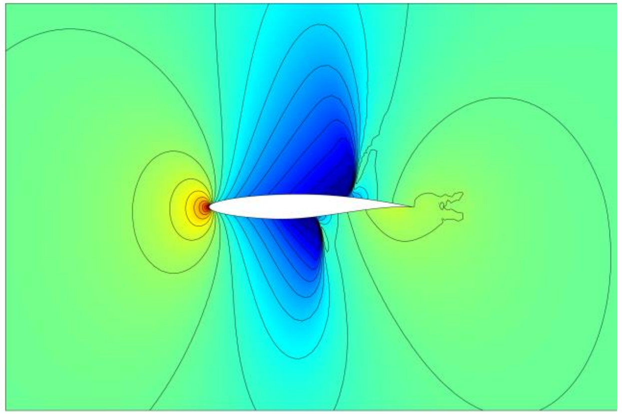
Extension to viscous turbulent flows

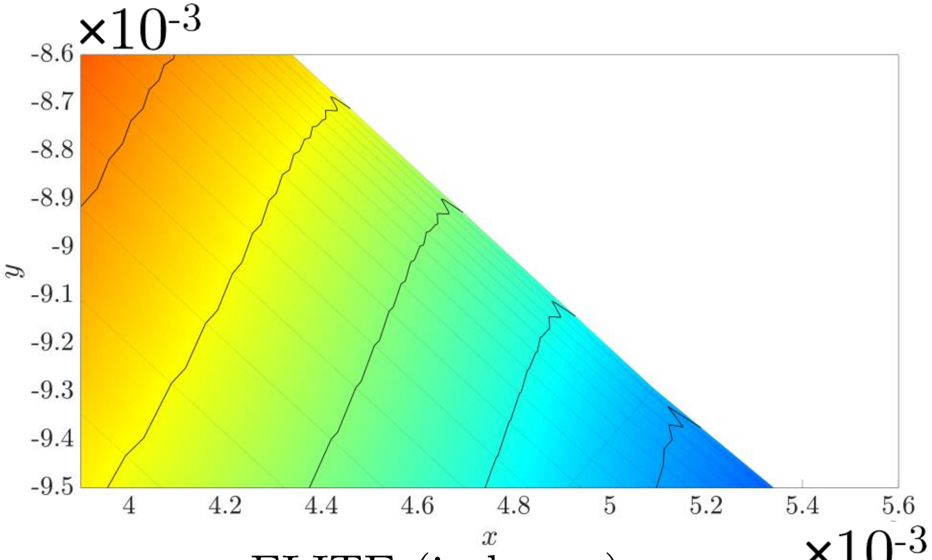
- Challenges induced by highly stretched elements (e.g., boundary layer, shear layer)



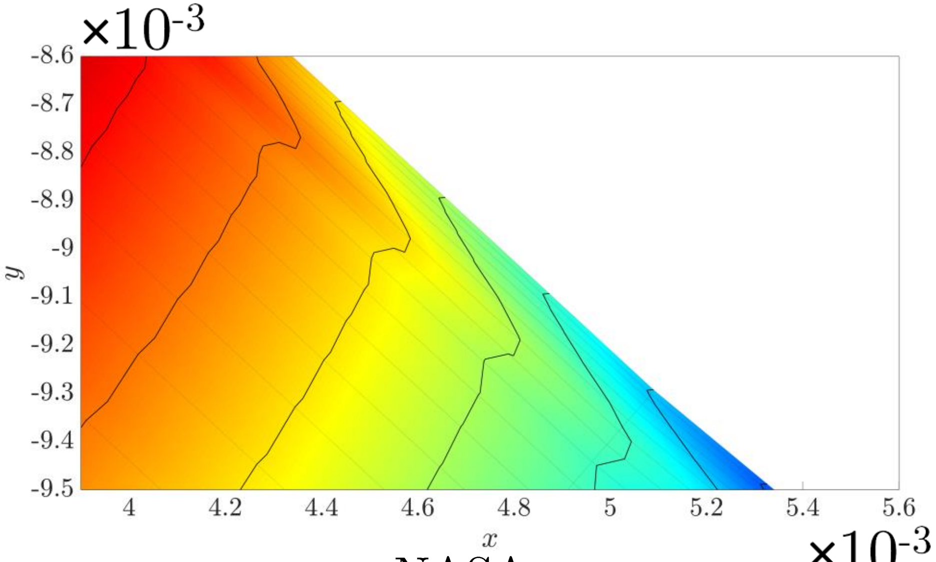


- Pressure field for the RAE2822. M=0.729, AoA=2.31, Re=6.5×10^6





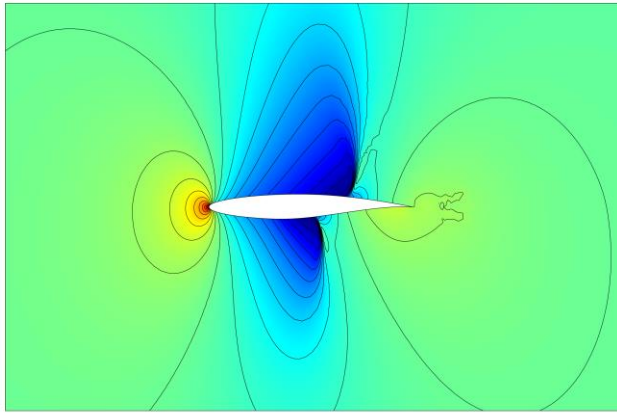
FLITE (in-house)

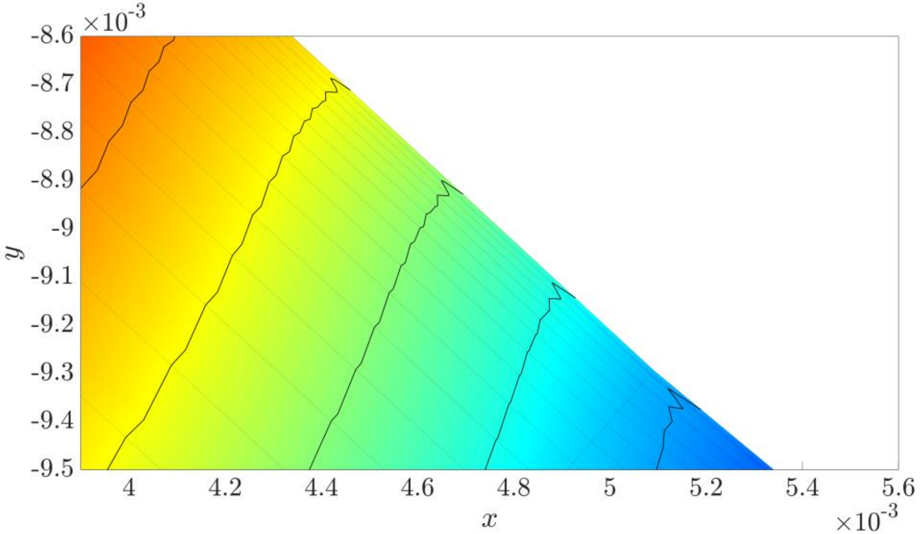


NASA

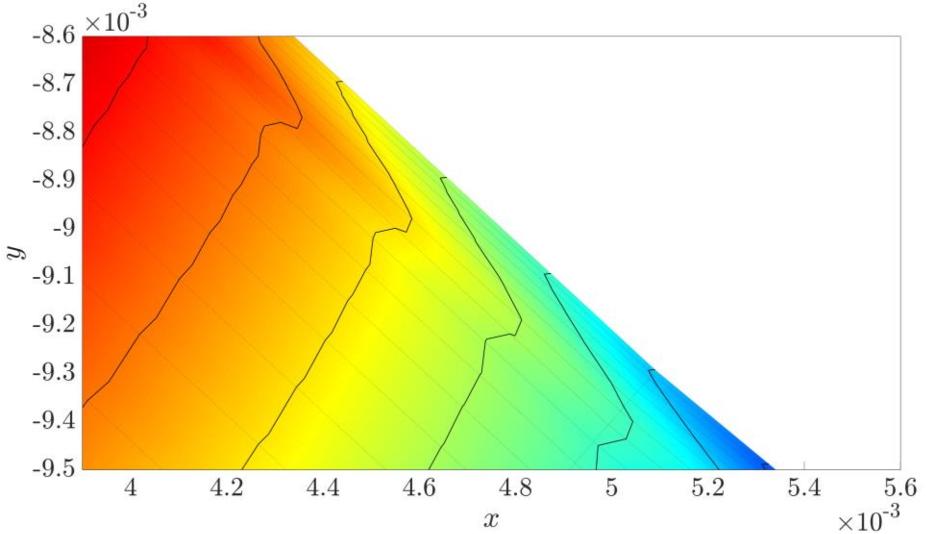
Extension to viscous turbulent flows

- Challenges induced by highly stretched elements (e.g., boundary layer, shear layer)





FLITE (in-house)



NASA

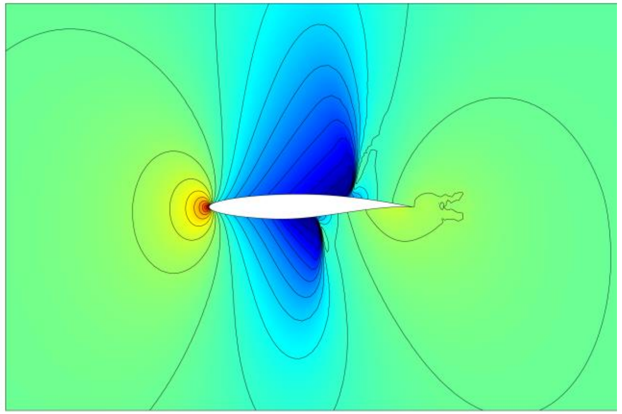
- Very small variations of the pressure can lead to (non-physical) high gradients
- The computation of the gradient (FEM, FV, FD) involves dividing by an area (for high Reynolds this could be ~10^{-9})

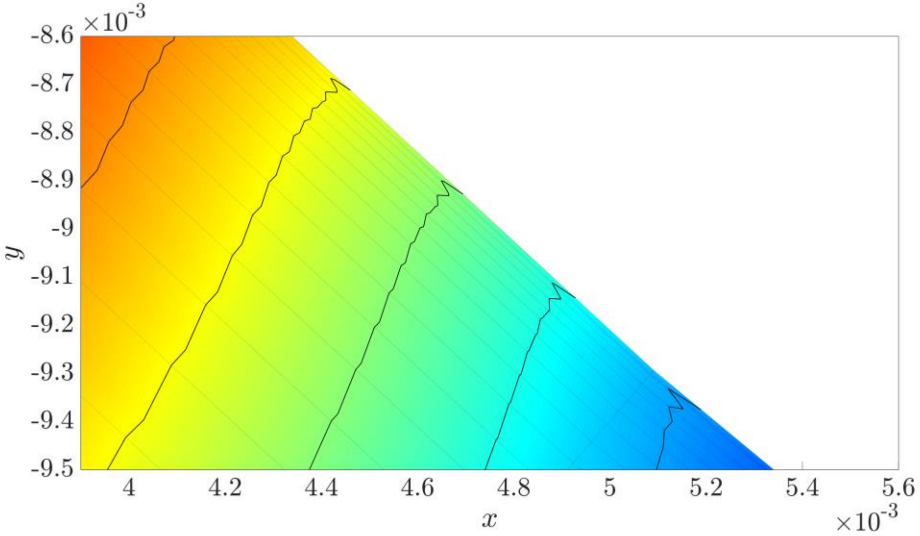
$$\boldsymbol{\nabla}\sigma_i \approx \frac{\displaystyle\sum_{\Omega_e \in \mathcal{P}_i} |\Omega_e| \boldsymbol{\nabla}\sigma_e}{\displaystyle\sum_{\Omega_e \in \mathcal{P}_i} |\Omega_e|}$$

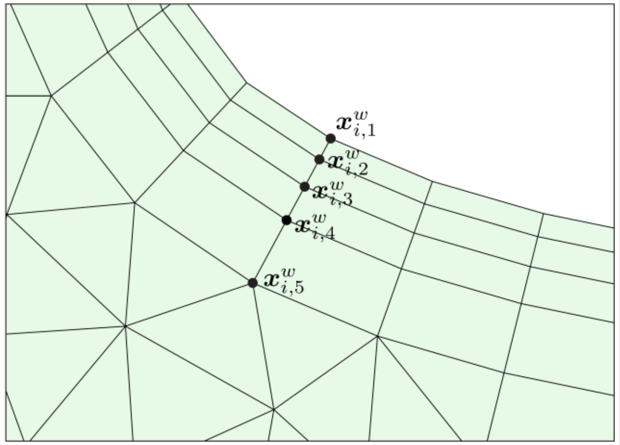
$$\boldsymbol{\nabla}\sigma_i \approx \frac{1}{|V_i|}\left[\sum_{j \in \mathcal{N}_i} \frac{1}{2}(\sigma_i + \sigma_j)\boldsymbol{C}_{ij} + \sum_{j \in \mathcal{N}_i^{\partial}} \sigma_i \boldsymbol{D}_{ij}\right]$$

Extension to viscous turbulent flows

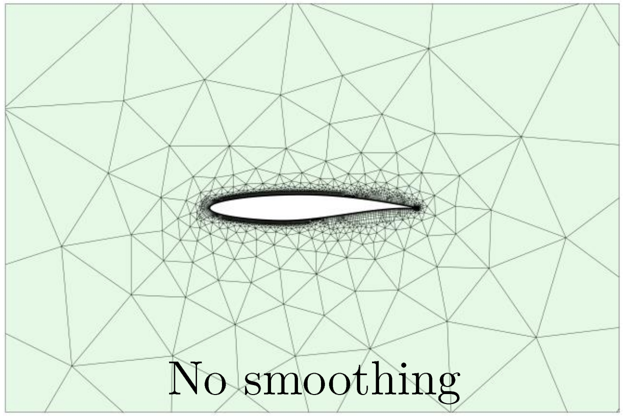
- Challenges induced by highly stretched elements (e.g., boundary layer, shear layer)

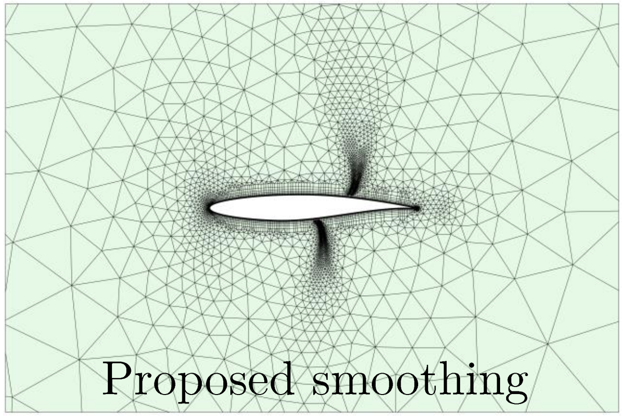






- A smoothing of the pressure in the normal direction is proposed
  - Mesh obtained using the spacing computed with the pressure as key variable

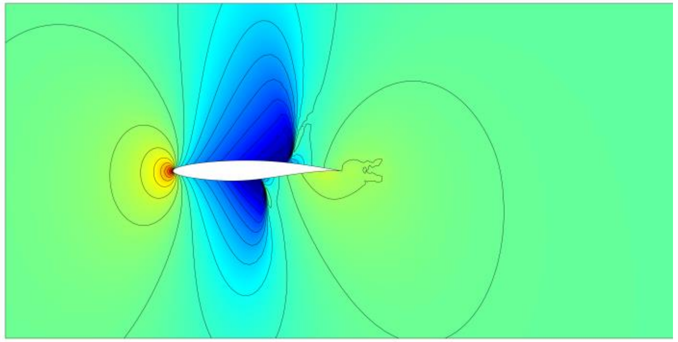

No smoothing


Proposed smoothing

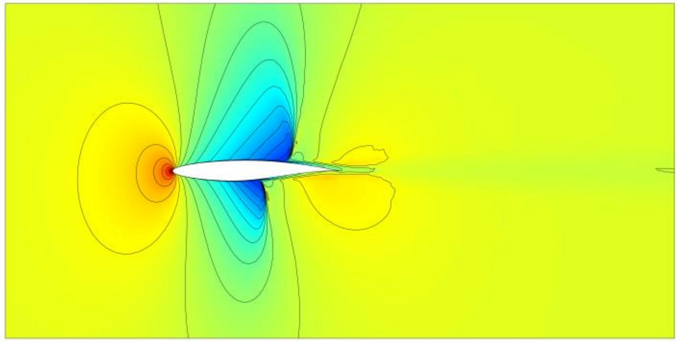
S Sanchez-Gamero, O Hassan, RS, A machine learning approach to predict near-optimal meshes for turbulent compressible flow simulations, arXiv, 2024

Extension to viscous turbulent flows

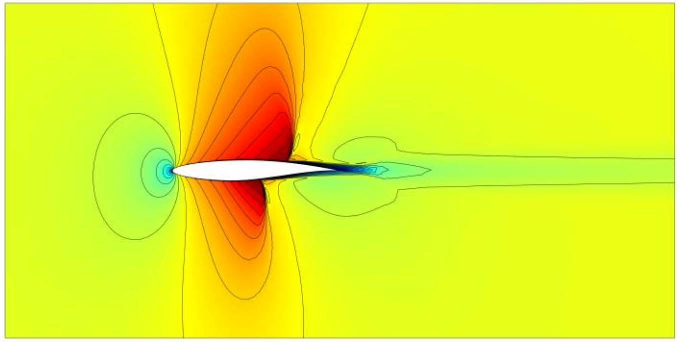
- Several key variables required to define the spacing function capable of capturing all flow features



Pressure

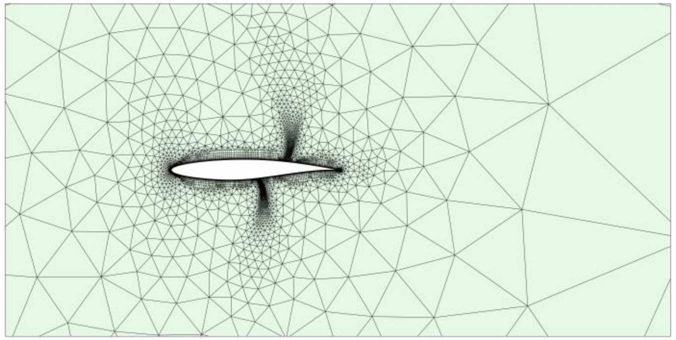


Density

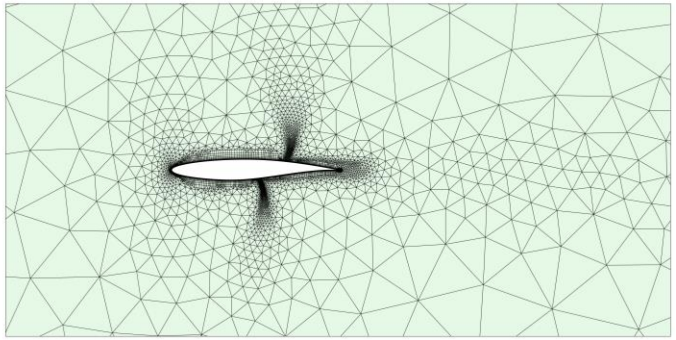


Mach

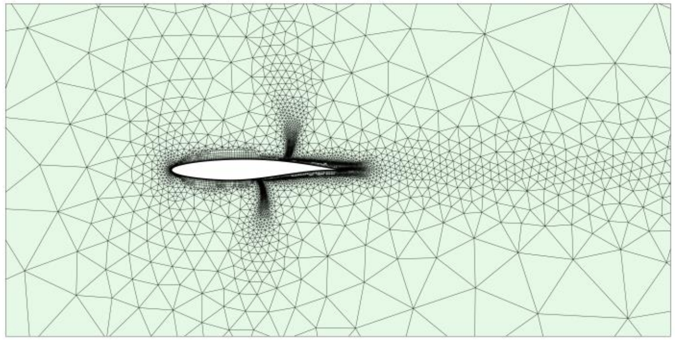
- Mesh obtained using the spacing computed with the different key variables



Key variable: Pressure



Key variable: Pressure & density

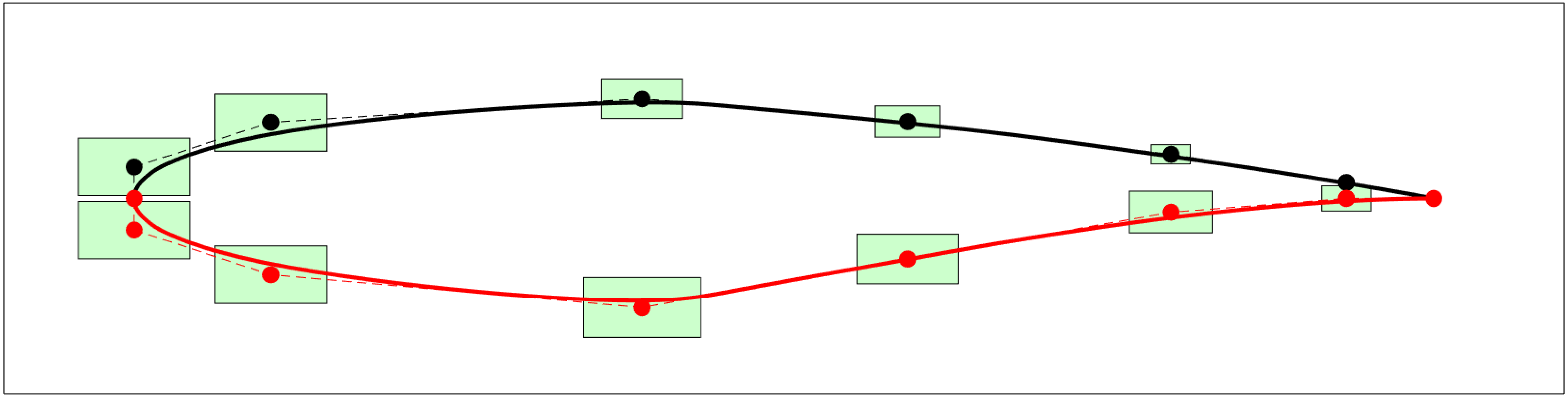


Key variable: Pressure & Mach

S Sanchez-Gamero, O Hassan, RS, A machine learning approach to predict near-optimal meshes for turbulent compressible flow simulations, arXiv, 2024

CAD integration

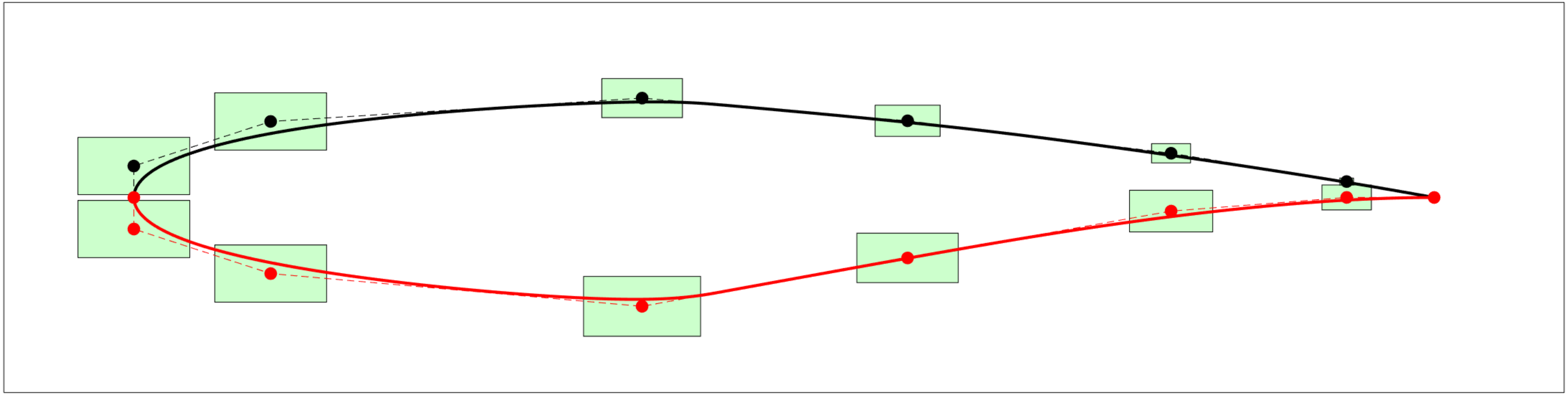
- Use NURBS control points as the design parameters (inputs of the NN)



S Sanchez-Gamero, O Hassan, RS, A machine learning approach to predict near-optimal meshes for turbulent compressible flow simulations, arXiv, 2024

CAD integration

- Use NURBS control points as the design parameters (inputs of the NN)

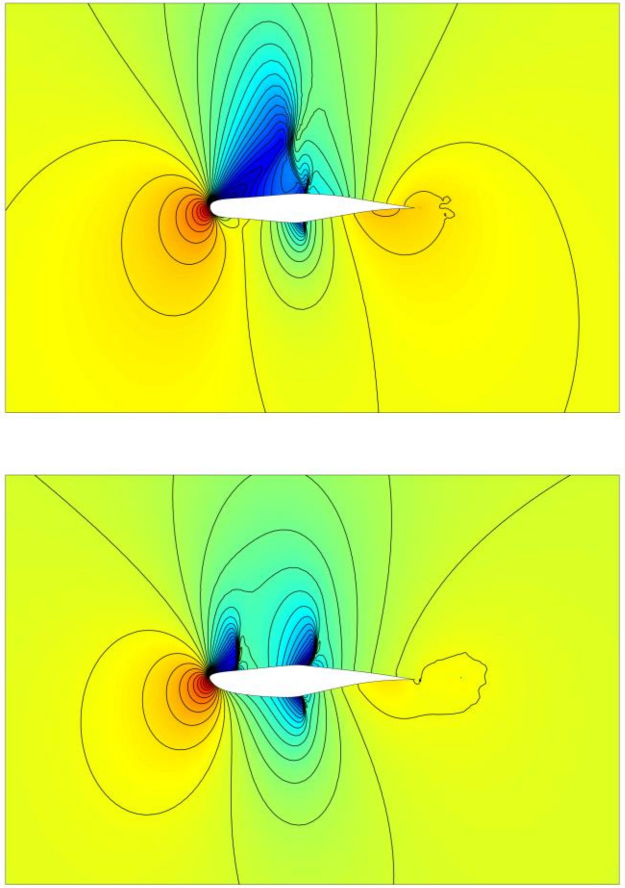


S Sanchez-Gamero, O Hassan, RS, A machine learning approach to predict near-optimal meshes for turbulent compressible flow simulations, arXiv, 2024

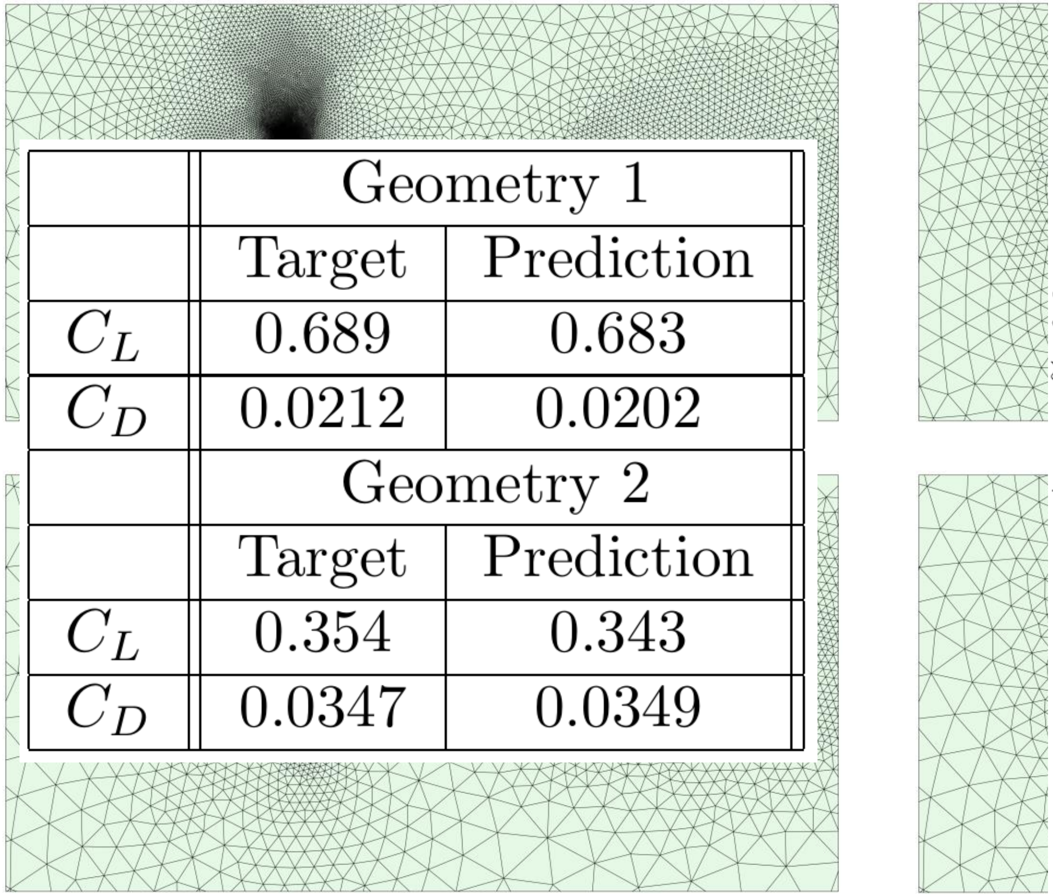
Examples

- 23 geometric parameters (control points of two NURBS curves)

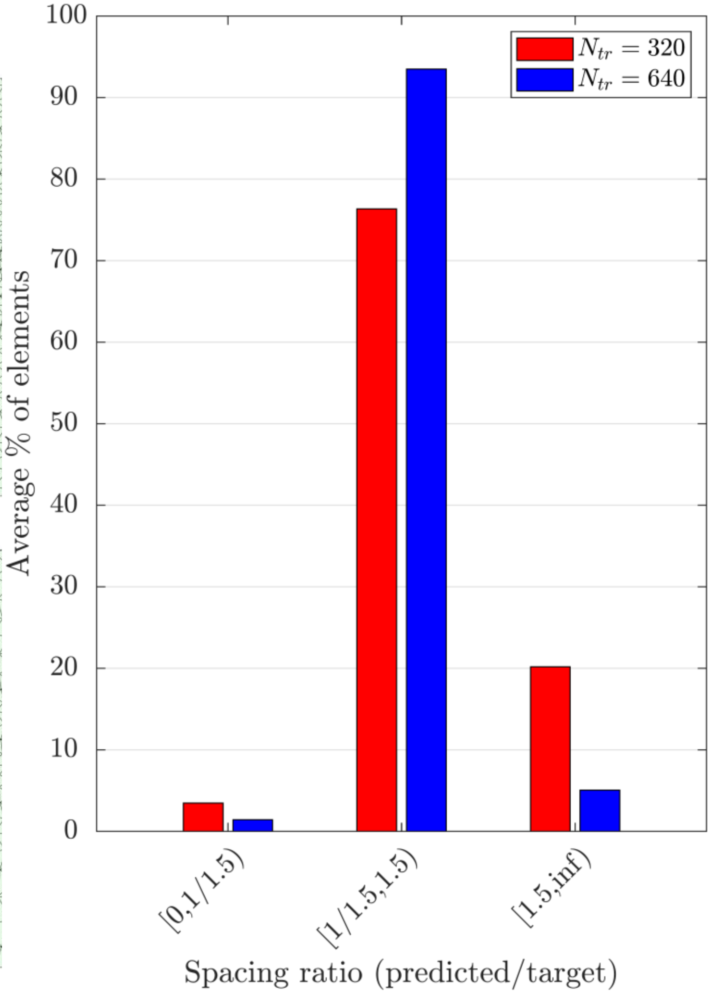
Solution



Target



	Geometry 1	
	Target	Prediction
C_L	0.689	0.683
C_D	0.0212	0.0202
	Geometry 2	
	Target	Prediction
C_L	0.354	0.343
C_D	0.0347	0.0349



S Sanchez-Gamero, O Hassan, RS, A machine learning approach to predict near-optimal meshes for turbulent compressible flow simulations, arXiv, 2024

Outline

- AI to predict mesh spacing using
  - Mesh sources
  - Background meshes
  - Examples
  - How green is the AI system?
  - Extensions to anisotropic spacing, viscous turbulent flows and CAD integration
- AI to aid mesh adaptation
  - High-order HDG and degree adaptivity
  - Examples
- Concluding remarks

High-order HDG and degree adaptivity

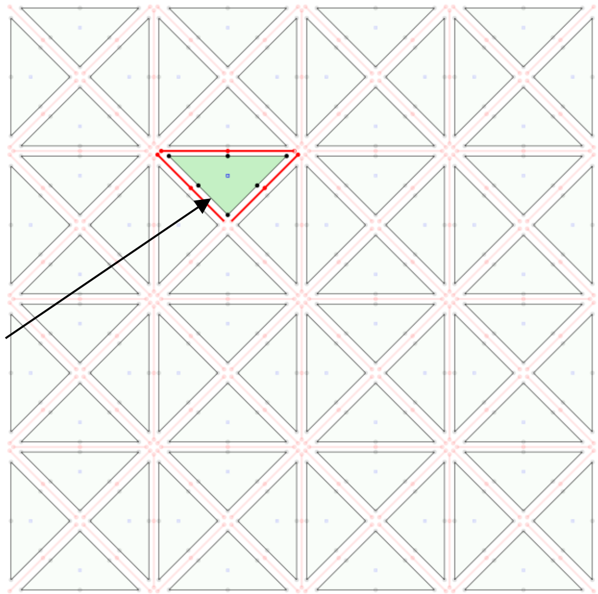
- Incompressible Navier-Stokes equations

$$
\begin{cases}
\boldsymbol{u}_t - \nabla\cdot(2\nu\nabla^{\mathrm{s}}\boldsymbol{u} - p\mathbf{I}) + \nabla\cdot(\boldsymbol{u}\otimes\boldsymbol{u}) = \boldsymbol{s} & \text{in } \Omega\times(0,T],\\
\nabla\cdot\boldsymbol{u} = 0 & \text{in } \Omega\times(0,T],\\
\boldsymbol{u} = \boldsymbol{u}_D & \text{on } \Gamma_D\times(0,T],\\
\left((2\nu\nabla^{\mathrm{s}}\boldsymbol{u} - p\mathbf{I}) - (\boldsymbol{u}\otimes\boldsymbol{u})\right)\boldsymbol{n} = \boldsymbol{t} & \text{on } \Gamma_N\times(0,T].\\
\boldsymbol{u} = \boldsymbol{u}_0 & \text{in } \Omega\times\{0\},
\end{cases}
$$

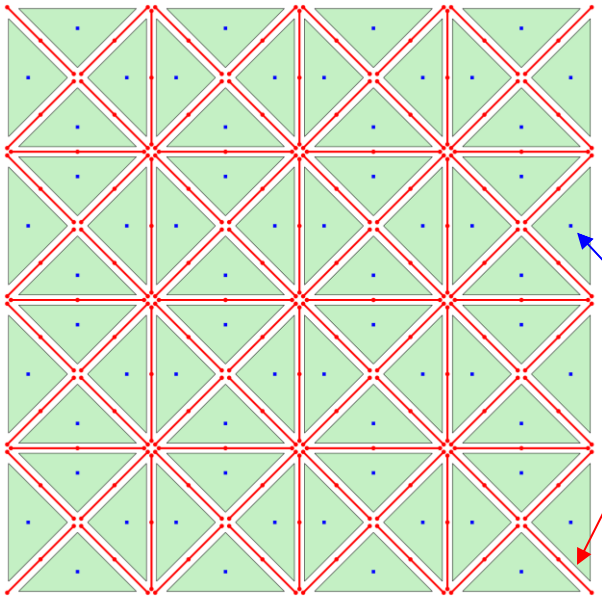
$$
\boldsymbol{u}_t + \nabla\cdot(\sqrt{2\nu}\boldsymbol{L} + p\mathbf{I}) + \nabla\cdot(\boldsymbol{u}\otimes\boldsymbol{u}) = \boldsymbol{s}
$$
$$
\boldsymbol{L} = -\sqrt{2\nu}\nabla^{\mathrm{s}}\boldsymbol{u}
$$

- The HDG method solves the problem in two steps introducing the hybrid velocity

$$
\begin{cases}
\boldsymbol{L} + \sqrt{2\nu}\nabla^{\mathrm{s}}\boldsymbol{u} = \boldsymbol{0} \quad \text{in } \Omega_e,\\
\boldsymbol{u}_t + \nabla\cdot(\sqrt{2\nu}\boldsymbol{L} \quad\quad e,\\
\quad e,\\
\quad e,\\
\quad \partial\Omega_e \cap \Gamma_D,\\
\quad \partial\Omega_e \setminus \Gamma_D,\\
\quad e \times \{0\},
\end{cases}
$$



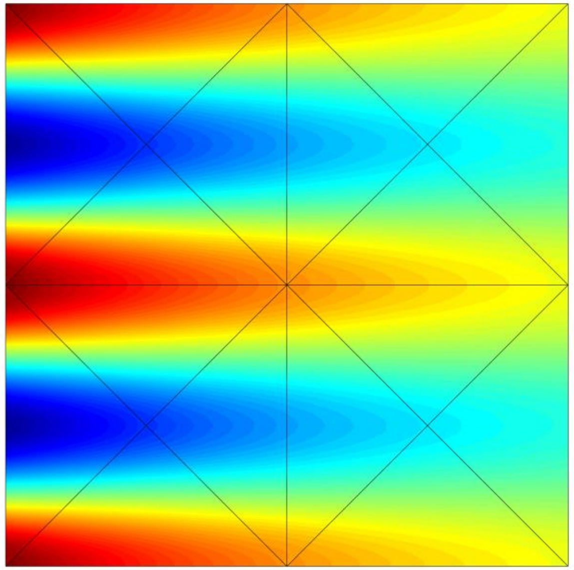
$$
\begin{cases}
[\![\quad = \boldsymbol{0} \quad \text{on } \Gamma,\\
\quad = -\boldsymbol{t} \quad \text{on } \Gamma_N,\\
\langle \hat{\boldsymbol{u}}\cdot\boldsymbol{n}_e, \quad = 0,
\end{cases}
$$

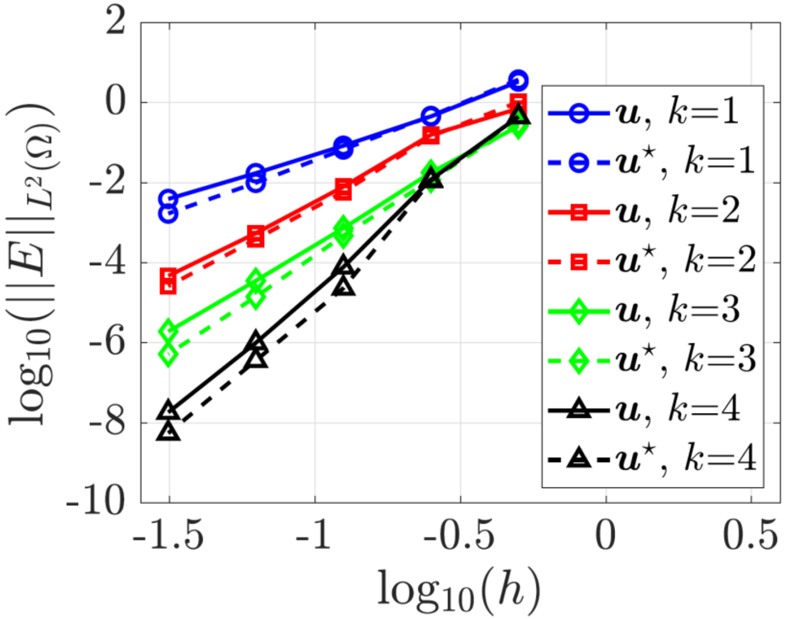


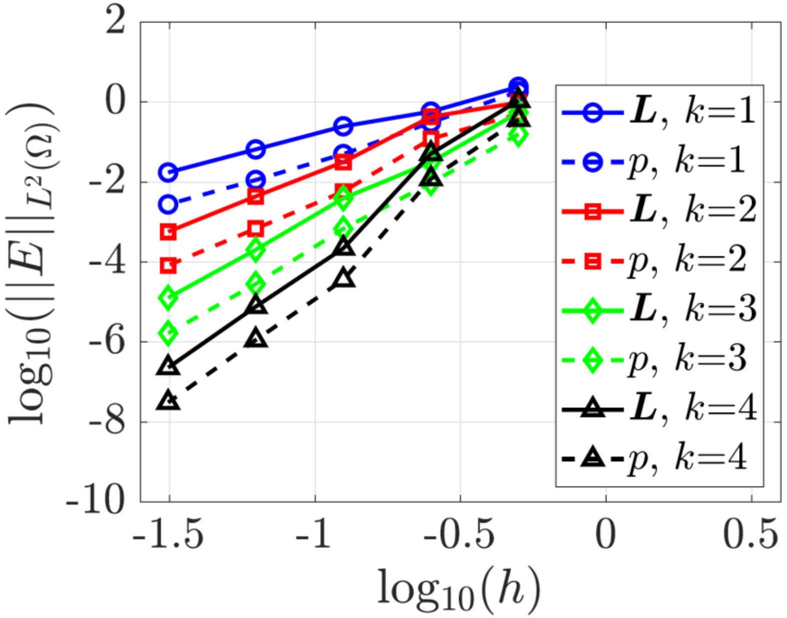
M Giacomini, RS, A Huerta Tutorial on hybridisable discontinuous Galerkin (HDG) formulation for incompressible flow problems, CISM, 2020

High-order HDG and degree adaptivity

- Optimal convergence (*k+1* rate) of the $L^2(\Omega)$ norm of the error for velocity, pressure AND mixed variable (velocity gradient) using polynomials of degree k



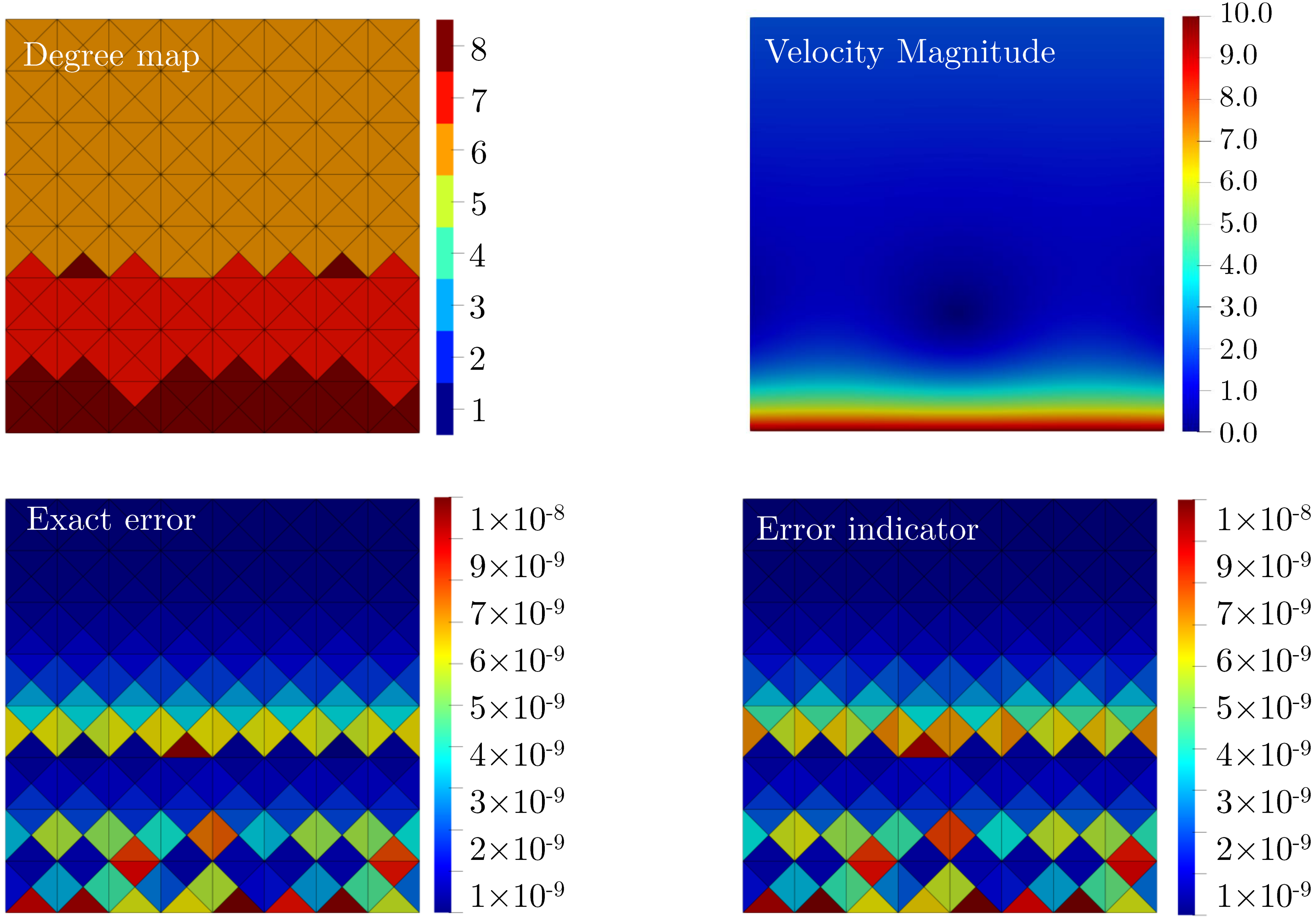




- Enables the computation of a super-convergent velocity (*k+2* rate) by solving element-by-element problems
- A cheap error indicator is readily available (comparing the velocity and the super-convergent velocity)

High-order HDG and degree adaptivity

- Example of degree adaptivity for a steady problem (Wang flow)

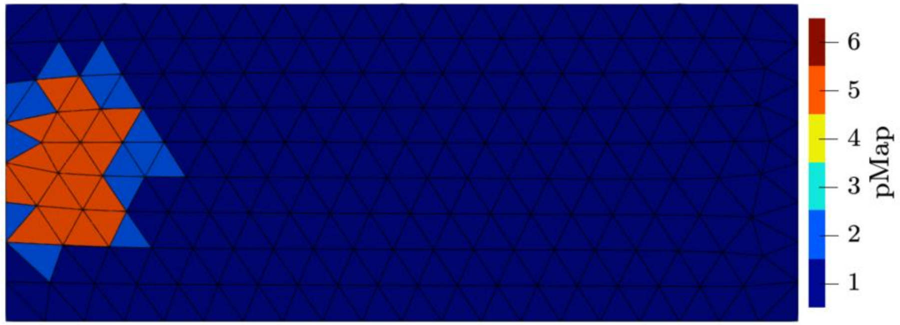


High-order HDG and degree adaptivity

- For transient problems, solution features might travel to elements where the degree of approximation has not been adapted yet
- Example: velocity perturbation (gust) travelling in free space

Reference solution

Degree adaptive solution



Degree of approximation in each element

High-order HDG and degree adaptivity

- For transient problems, solution features might travel to elements where the degree of approximation has not been adapted yet
- Example: velocity perturbation (gust) travelling in free space
- An accurate computation requires repeating each time step

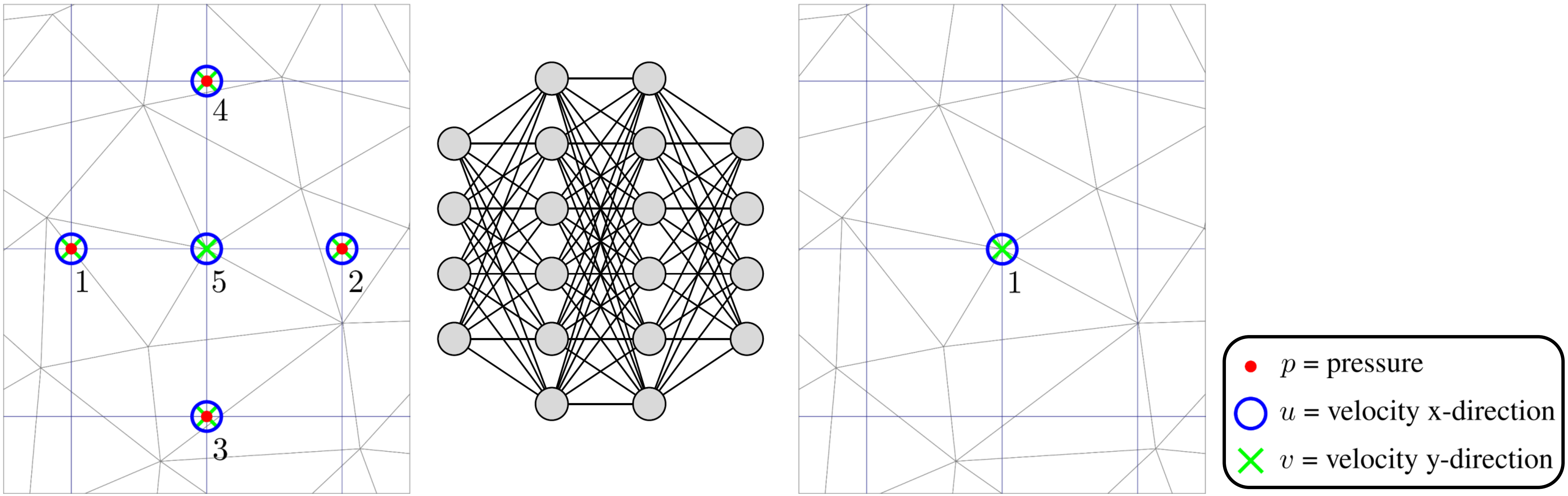
Reference solution

Degree adaptive solution

Degree adaptive solution repeating each time step

NN to aid degree-adaptivity

- A NN is designed to predict the velocity at a node at time t^{n+1}, from the velocity and pressure at a number of "surrounding" points (not nodes) at time t^n

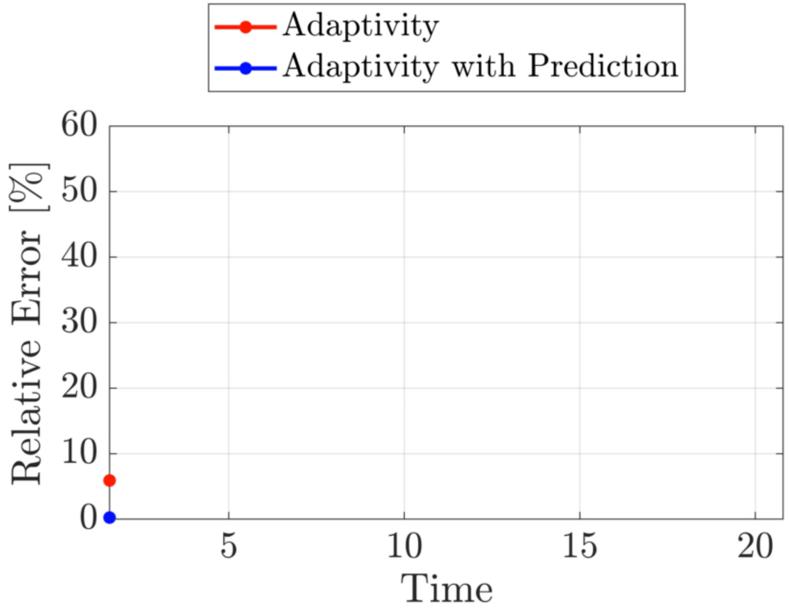


- Data is collected for a number of training cases (in this example varying gust intensity and width)
- To speed up training and reduce bias, we remove redundant information

NN to aid degree-adaptivity

- A trained NN predicts the solution at the next time step so that the degree can be adapted before the solution advances in time, avoiding repeating the time step

Reference solution



Degree adaptive solution

Degree adaptive solution with NN prediction

Concluding remarks

- An AI system to predict near-optimal isotropic and anisotropic mesh spacing for new simulations
  - Operating conditions
  - Geometric variations, including a link with the CAD
- Meshes proved to be suitable to perform simulations of unseen cases
- Reduction of computational cost and carbon emissions compared to current industrial practice

- An AI system to aid degree adaptivity for transient flows
- No need to repeat the time step to guarantee accurate simulations

EPSRC
Pioneering research and skills

- Future work involves
  - Combine sources and a background mesh
  - Simulation of gust impinging on aerodynamic shapes

AIRBUS
DEFENCE & SPACE

Mesh generation and adaptation using green AI

Rubén Sevilla

Zienkiewicz Institute for Modelling, Data and AI
Swansea University, Swansea, Wales, UK

Swansea University
Prifysgol Abertawe