MARIO: Multiscale Aerodynamic Resolution Invariant Operator

Aeroairbus Team (Airbus + ISAE-Supaero)

Giovanni Catalani, Jean Fesquet, Xavier Bertrand, Frédéric Tost, Enrico Foglia, Michael Bauerheim, Joseph Morlier

ML4CFD Competition Workshop

Vancouver, Canada 17/12/2024

Introduction: the ML4CFD Challenge

Industrial Interest

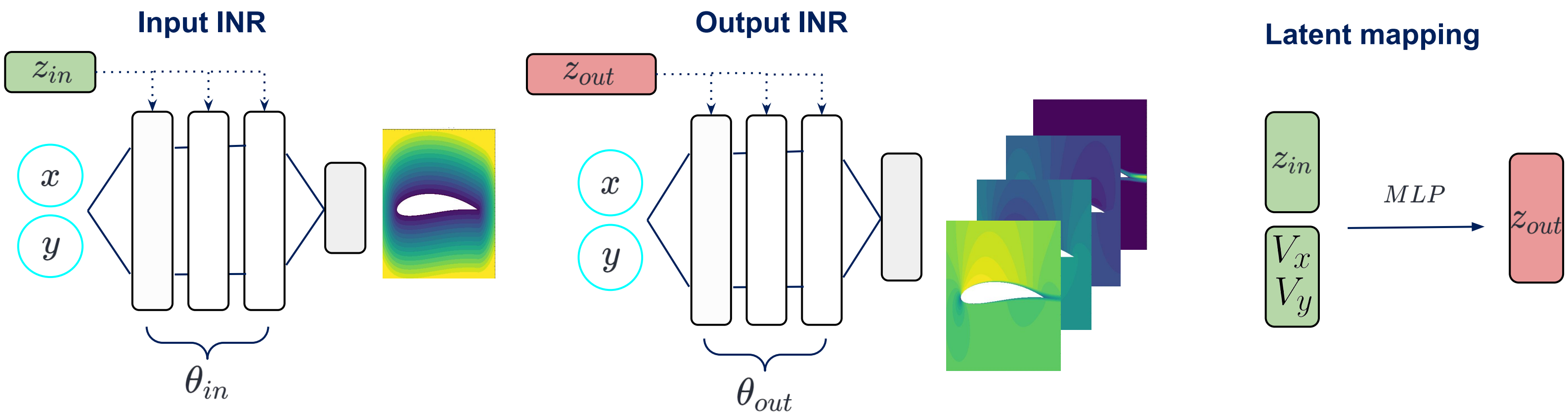
- **Machine Learning task**: predicting physical fields around airfoils and estimating integral force coefficients.
- **Dataset type**: realistic CFD dataset computed using the incompressible RANS equations.
- Relatively large mesh size, shape variations and variation in the Mach number and angle of attack.
- Evaluation not only on accuracy, but also on physical compliance, inference time and generalization.

Challenges

- Data scarcity: only 100 CFD computation available for training.
- Data format: changing unstructured grid, with large node size.
- Strongly non-linear quantities difficult to predict (ex . Velocity in the boundary layer and drag coefficient).

A step back: MARIO 1.0

Encode-process-Decode

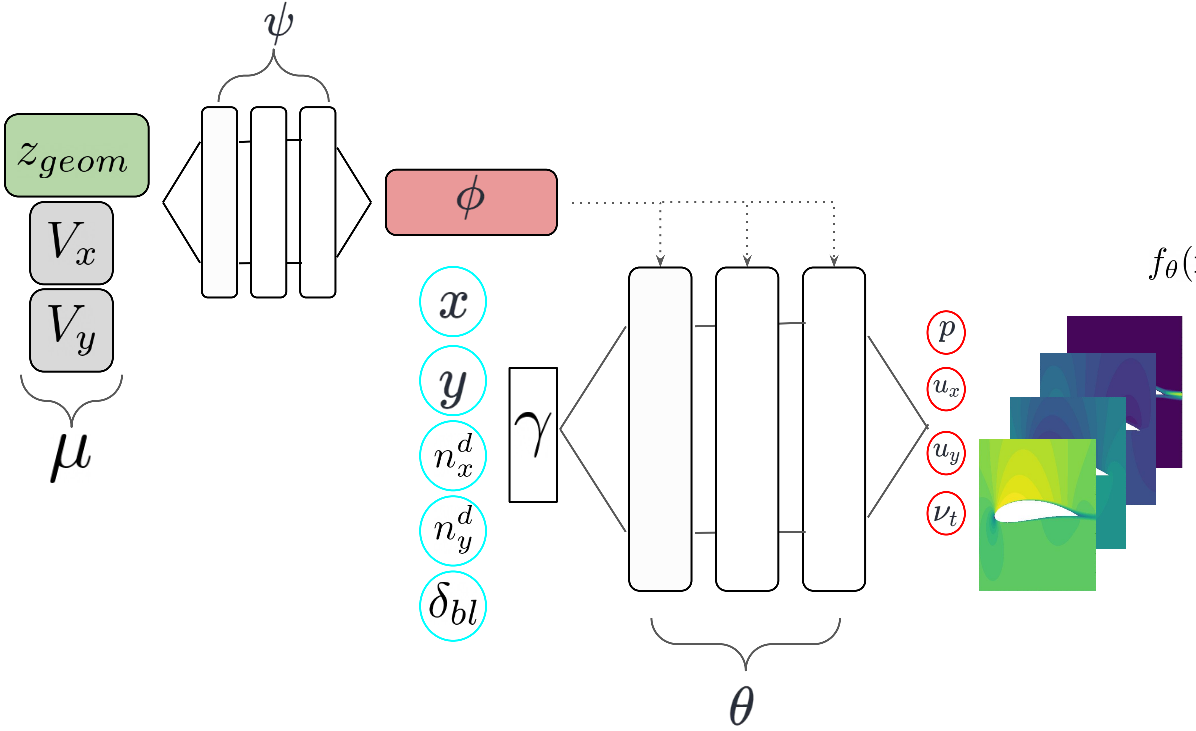


Main Limitations

- On small datasets this approach is prone to **overfitting**.
- The regression step in the latent space introduces large errors.
- The latent vectors predictions are optimized to minimize distance from the ground truth latent vectors and not the output fields.

[1] Serrano, Louis, et al. "INFINITY: Neural Field Modeling for Reynolds-Averaged Navier-Stokes Equations." *arXiv e-prints* (2023): arXiv-2307.

MARIO 2.0 : an end-to end architecture



Main Operator Network

$$f_\theta(\mathbf{x}) \quad = W_L(\eta_{L-1} \circ \eta_{L-2} \circ .... \circ \eta_1 \circ \gamma(\mathbf{x})) + b_L$$

Shift Modulation

$$\eta_l(\cdot) = ReLU(W_l(\cdot) + b_l + \phi_l)$$

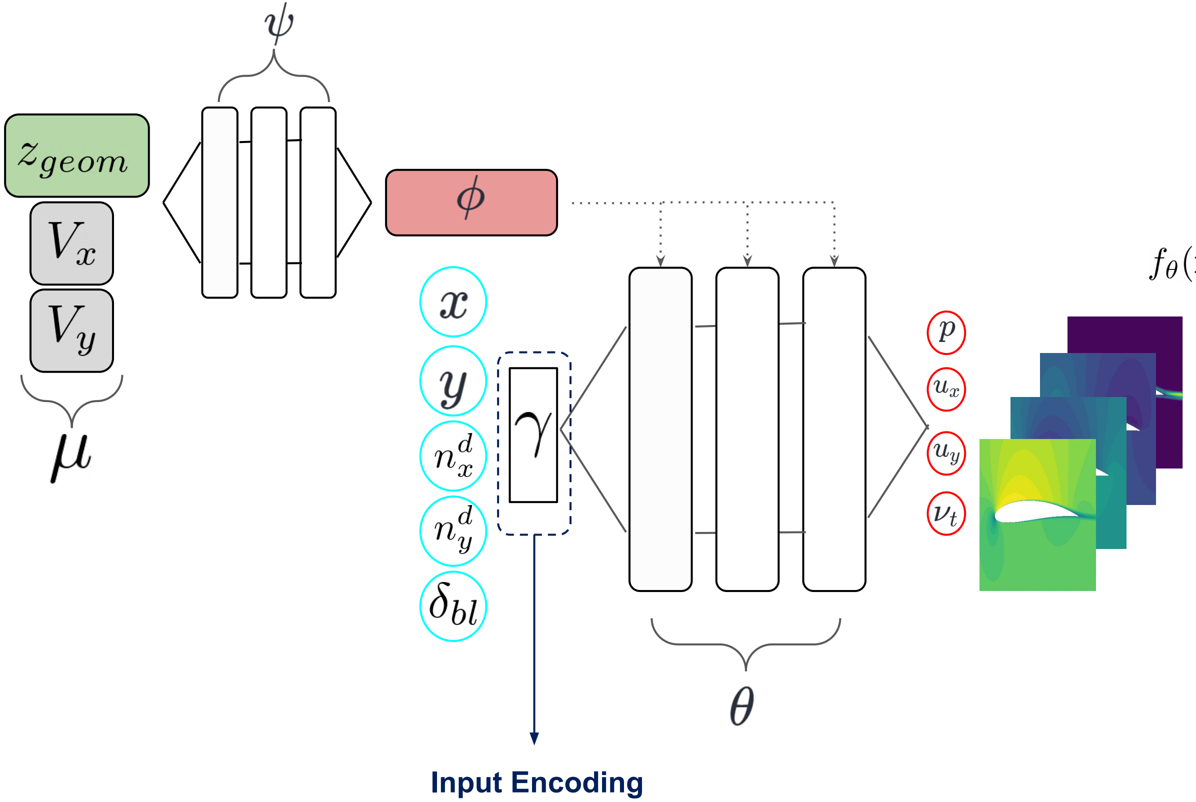
Hypernetwork Layerwise Modulation

$$\phi_l = h^l_\psi(\mu)$$

End-to-end Joint Optimization

$$\min_{\theta,\psi} \mathcal{L} = \sum_i^N \sum_{\mathbf{x}\in\mathcal{M}} mse\left(f_\theta(\mathbf{x}; h_\psi(\mu_i)), u_i(\mathbf{x})\right).$$

MARIO 2.0 : an end-to end architecture



Main Operator Network

$$f_\theta(\mathbf{x}) \quad = W_L(\eta_{L-1} \circ \eta_{L-2} \circ .... \circ \eta_1 \circ \gamma(\mathbf{x})) + b_L$$

Shift Modulation

$$\eta_l(\cdot) = ReLU(W_l(\cdot) + b_l + \phi_l)$$

Hypernetwork Layerwise Modulation

$$\phi_l = h_\psi^l(\mu)$$

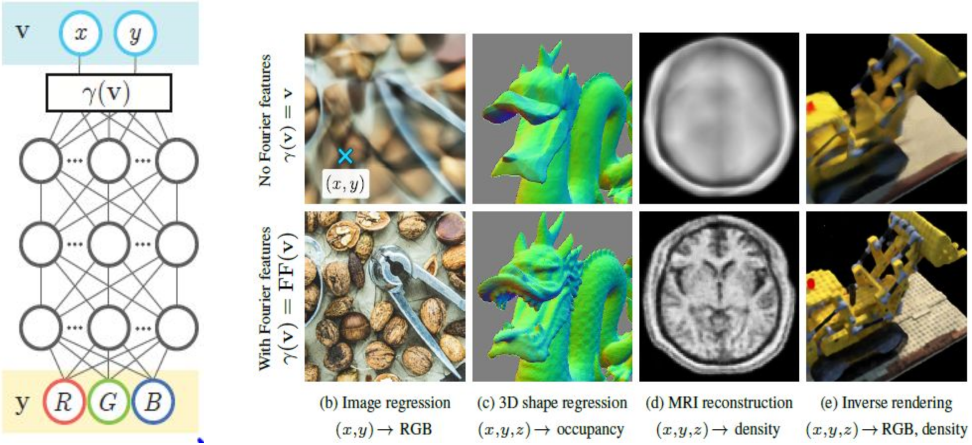
End-to-end Joint Optimization

$$\min_{\theta,\psi} \mathcal{L} = \sum_i^N \sum_{\mathbf{x} \in \mathcal{M}} mse\left(f_\theta(\mathbf{x}; h_\psi(\mu_i)), u_i(\mathbf{x})\right).$$

Spectral Bias and Fourier Features Networks

- Neural Fields are subject to **spectral bias**: difficulty in learning high frequency components of signals.
- **Fourier Features** mapping is used to improve the spectral convergence of simple MLPs.
- Input coordinates are embedded using Fourier Features sampled from a Gaussian Distribution .

$$\gamma(\mathbf{v}) = [cos(2\pi \mathbf{B}\mathbf{v}), sin(2\pi \mathbf{B}\mathbf{v})]^T \qquad b \sim \mathcal{N}(0, \sigma)$$



[2] Tancik, Matthew, et al. "Fourier features let networks learn high frequency functions in low dimensional domains." *Advances in Neural Information Processing Systems* 33 (2020)

Spectral Bias and Fourier Features Networks

- Neural Fields are subject to **spectral bias**: difficulty in learning high frequency components of signals.
- **Fourier Features** mapping is used to improve the spectral convergence of simple MLPs.
- Input coordinates are embedded using Fourier Features sampled from a Gaussian Distribution .

$$\gamma(\mathbf{v}) = [cos(2\pi \mathbf{B}\mathbf{v}), sin(2\pi \mathbf{B}\mathbf{v})]^T \qquad b \sim \mathcal{N}(0, \sigma)$$

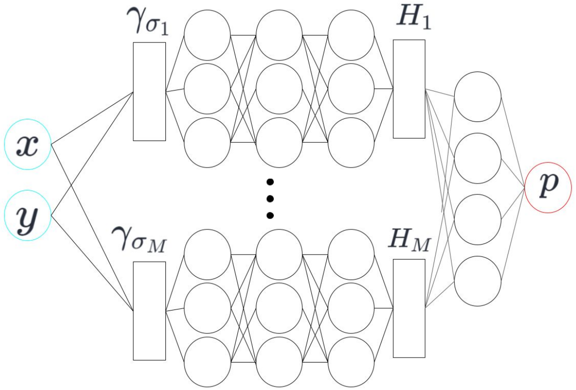
- The choice of the sampling standard deviation controls the **kernel bandwidth** and the optimal is task dependant.

$\sigma = 1$ $\qquad$ $\sigma = 2$ $\qquad$ $\sigma = 10$ $\qquad$ $\sigma = 32$ $\qquad$ $\sigma = 64$

[2] Tancik, Matthew, et al. "Fourier features let networks learn high frequency functions in low dimensional domains." *Advances in Neural Information Processing Systems* 33 (2020)

Main Network: Multiscale Fourier Features

- By operating multiple frequency encodings, we obtain the optimal kernel bandwidth without extensive finetuning.
- For multi output physical prediction, different signals have different spectral content.
- **Example:** velocity presents sharp variation in the boundary layer, while pressure is approx constant

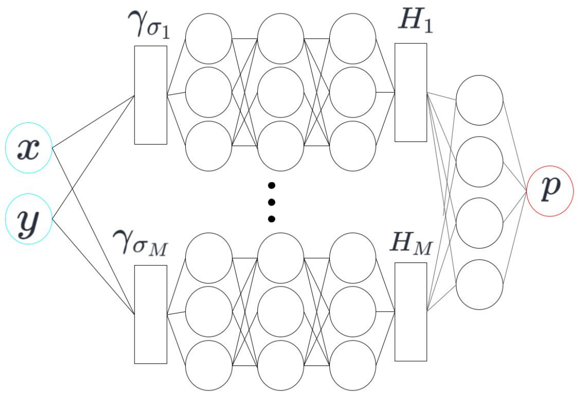


$$\gamma_{\sigma_i}(v) = [sin(2\pi \mathbf{B}_{\sigma_i} v), cos(2\pi \mathbf{B}_{\sigma_i} v)]$$

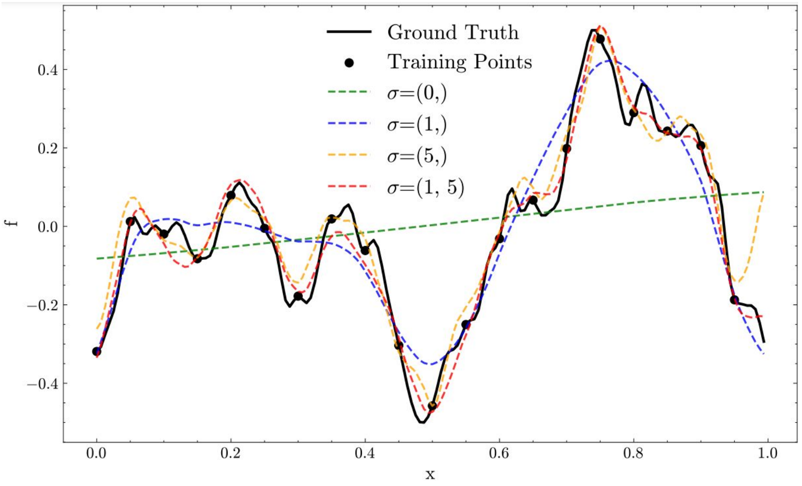
$$b_{\sigma_i} \sim \mathcal{N}(0, \sigma_i^2) \quad i = 1, ..., M$$

Main Network: Multiscale Fourier Features

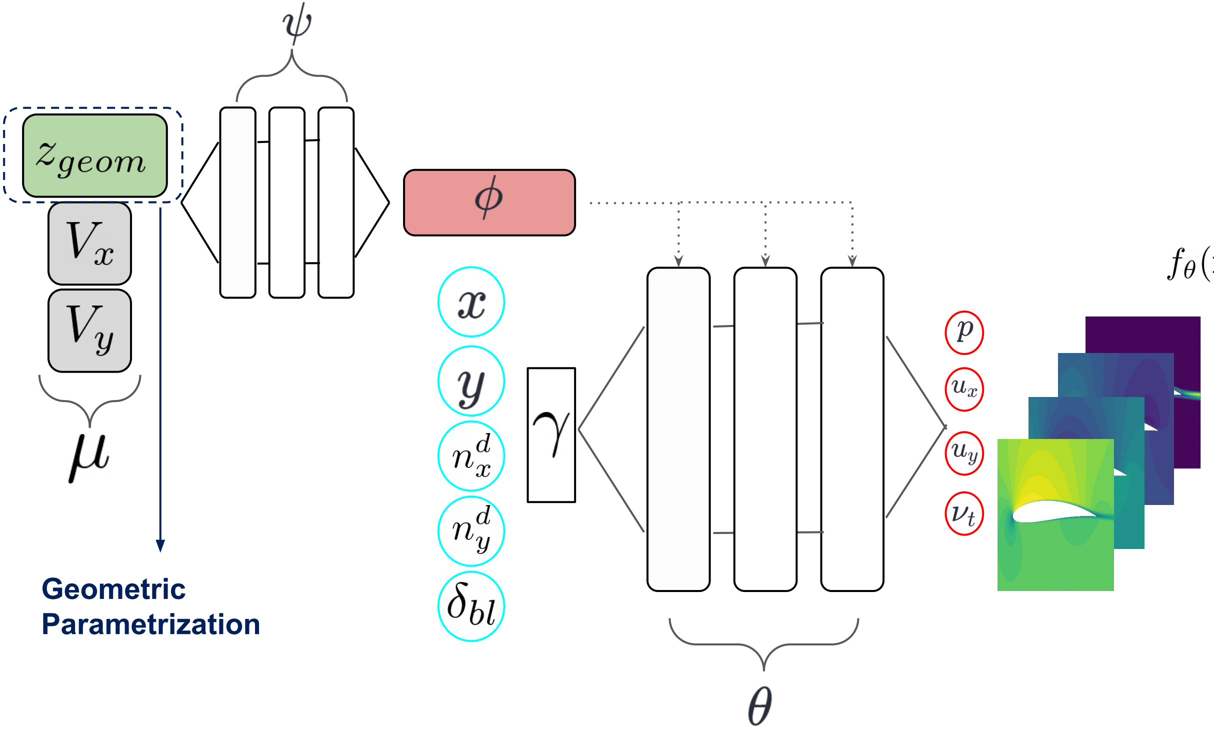
- By operating multiple frequency encodings, we obtain the optimal kernel bandwidth without extensive finetuning.
- For multi output physical prediction, different signals have different spectral content.
- **Example:** velocity presents sharp variation in the boundary layer, while pressure is approx constant



Example: signal fitting in 1D



MARIO 2.0 : an end-to end architecture



Main Operator Network

$$f_\theta(\mathbf{x}) \quad = W_L(\eta_{L-1} \circ \eta_{L-2} \circ .... \circ \eta_1 \circ \gamma(\mathbf{x})) + b_L$$

Shift Modulation

$$\eta_l(\cdot) = ReLU(W_l(\cdot) + b_l + \phi_l)$$

Hypernetwork Layerwise Modulation

$$\phi_l = h_\psi^l(\mu)$$

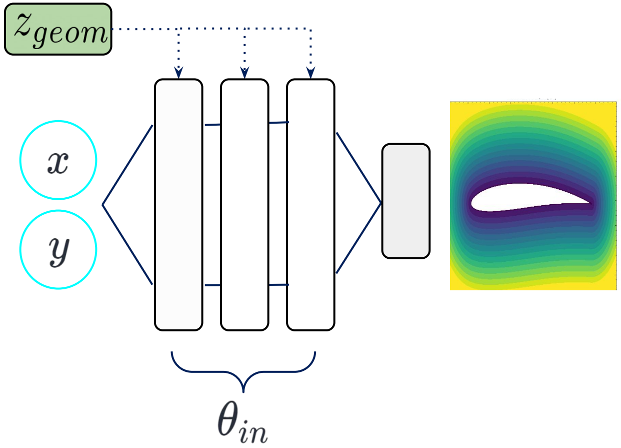
End-to-end Joint Optimization

$$\min_{\theta,\psi} \mathcal{L} = \sum_i^N \sum_{\mathbf{x}\in\mathcal{M}} mse\left(f_\theta(\mathbf{x}; h_\psi(\mu_i)), u_i(\mathbf{x})\right).$$

Geometric Parametrization

SDF Encoding

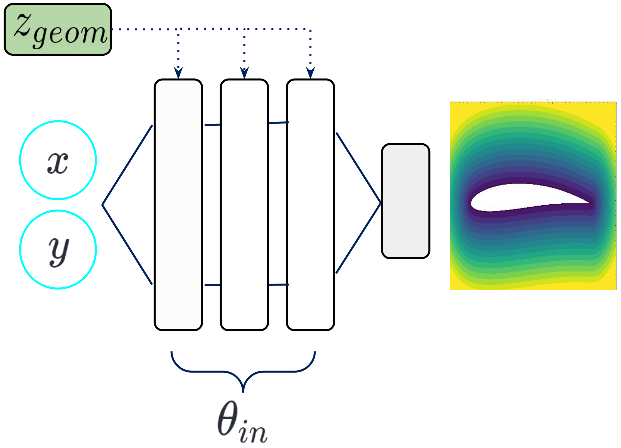
Methodology: Encoding the SDF field into a latent representation using a conditional neural field.
Advantages: General Framework for any shape (also in 3D). Can capture fine shape variations.
Limitations: Relatively slow at inference time.



Geometric Parametrization

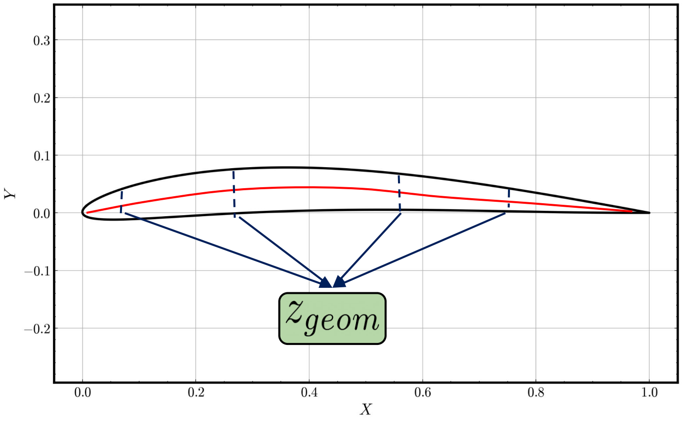
SDF Encoding

Methodology: Encoding the SDF field into a latent representation using a conditional neural field.
Advantages: General Framework for any shape (also in 3D). Can capture fine shape variations.
Limitations: Relatively slow at inference time.

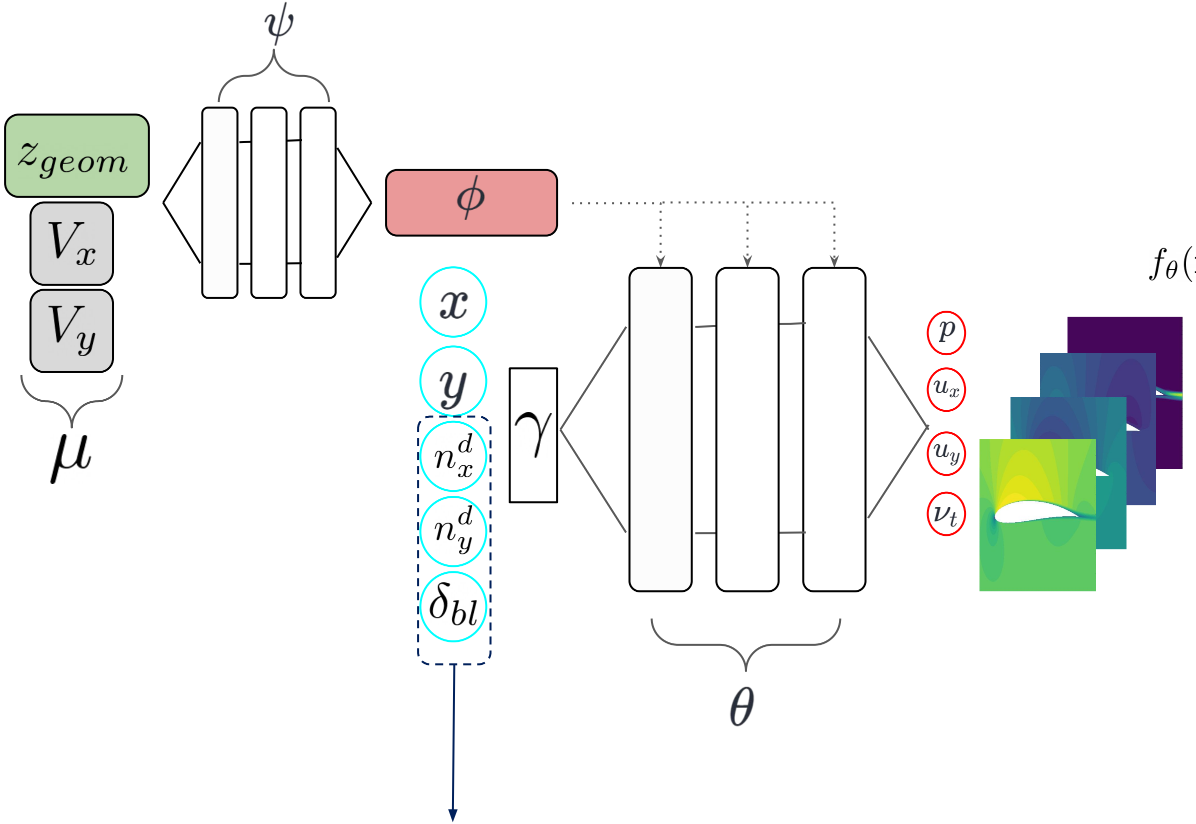


Thickness and Camber distribution

Methodology: Thickness and camber distribution at fixed locations.
Advantages: Fast and reliable for simple 2D airfoil shapes.
Limitations: Impractical for more complex shapes or 3D cases.



MARIO 2.0 : an end-to end architecture



Input Fields: Feature Engineering

Main Operator Network

$$f_\theta(\mathbf{x}) \quad = W_L(\eta_{L-1} \circ \eta_{L-2} \circ \dots\circ \eta_1 \circ \gamma(\mathbf{x})) + b_L$$

Shift Modulation

$$\eta_l(\cdot) = ReLU(W_l(\cdot) + b_l + \phi_l)$$

Hypernetwork Layerwise Modulation

$$\phi_l = h_\psi^l(\mu)$$

End-to-end Joint Optimization

$$\min_{\theta,\psi} \mathcal{L} = \sum_{i}^{N} \sum_{\mathbf{x}\in\mathcal{M}} mse\left(f_\theta(\mathbf{x}; h_\psi(\mu_i)), u_i(\mathbf{x})\right).$$

Input Fields: feature engineering

- The normals are 0 off the surface.
- A single model is used to process all points (on and off the surface).

Two of the five given input fields don't carry useful info for flow domain (most) points.

➔ ~~**Remove normals fields altogether**~~**.**

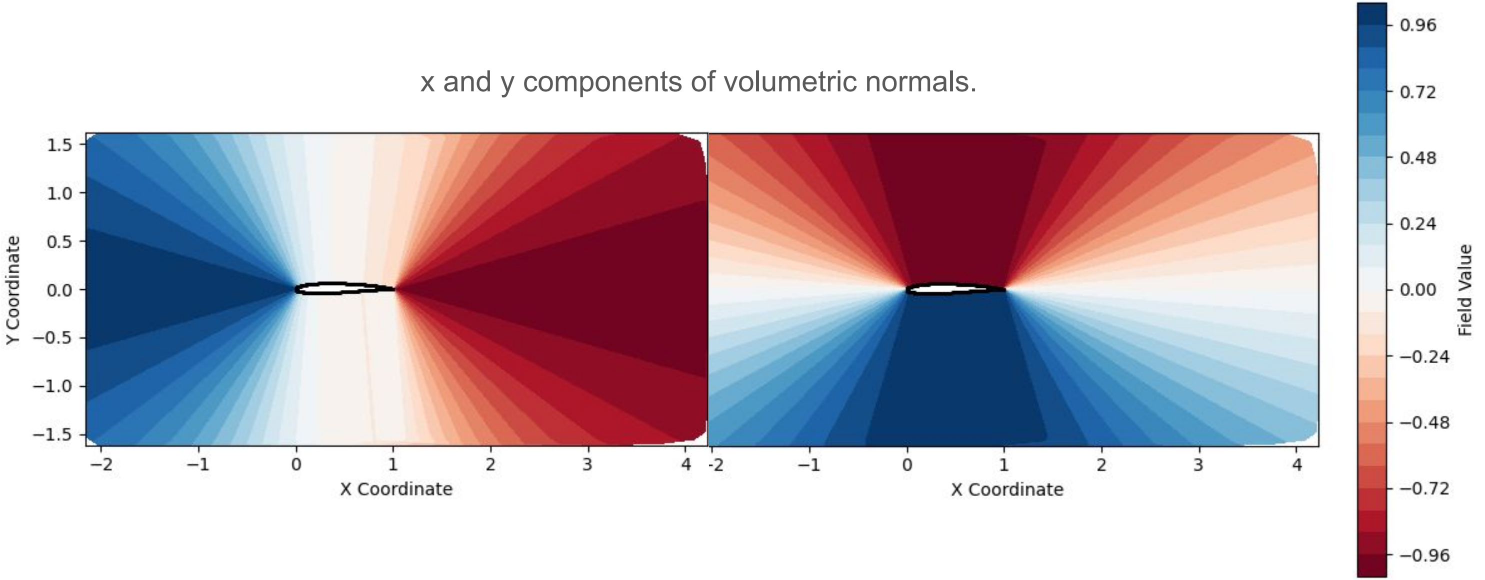
Input Fields: feature engineering

- The normals are 0 off the surface.
- A single model is used to process all points (on and off the surface).

Two of the five given input fields don't carry useful info for flow domain (most) points.

➔ ~~**Remove normals fields altogether.**~~
➔ **Process the normals fields to include direction to the closest point on the surface.**
➔ **Obtain an invariant reference frame with respect to the airfoil shape variation (directional sdf)**

x and y components of volumetric normals.

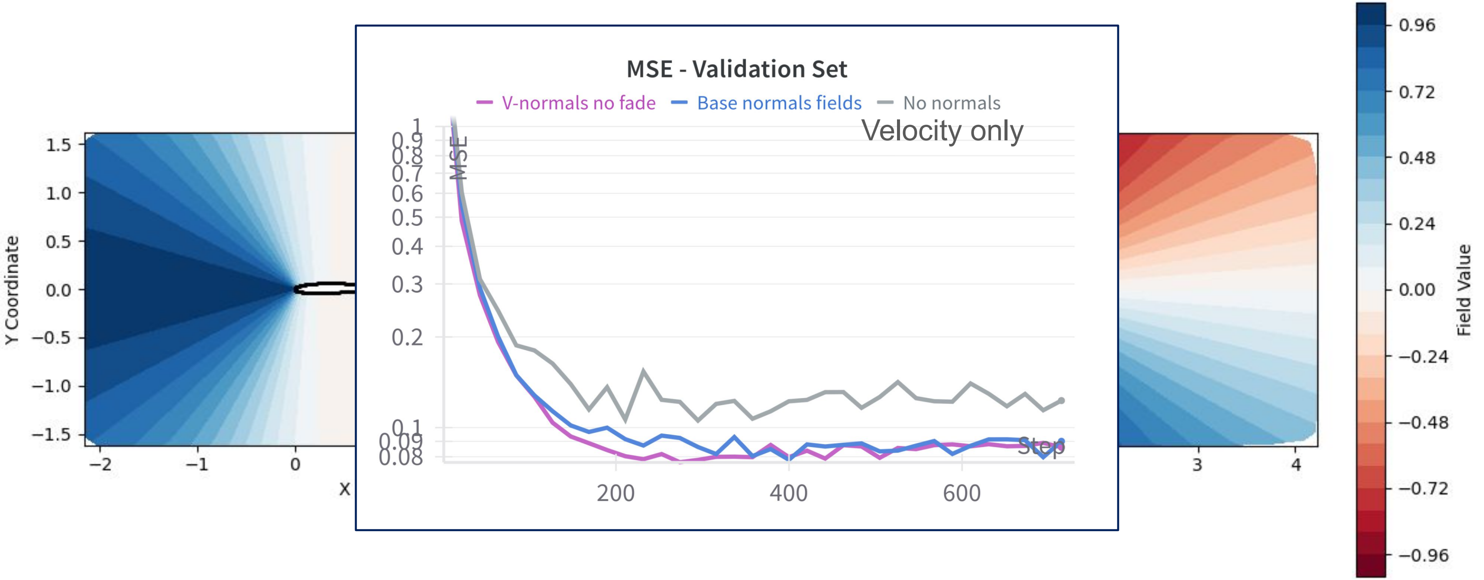


Input Fields: feature engineering

- The normals are 0 in the flow domain.
- A single model is used to process any point (flow domain or surface).

Two of the five given input fields don't carry useful info for flow domain (most) points.

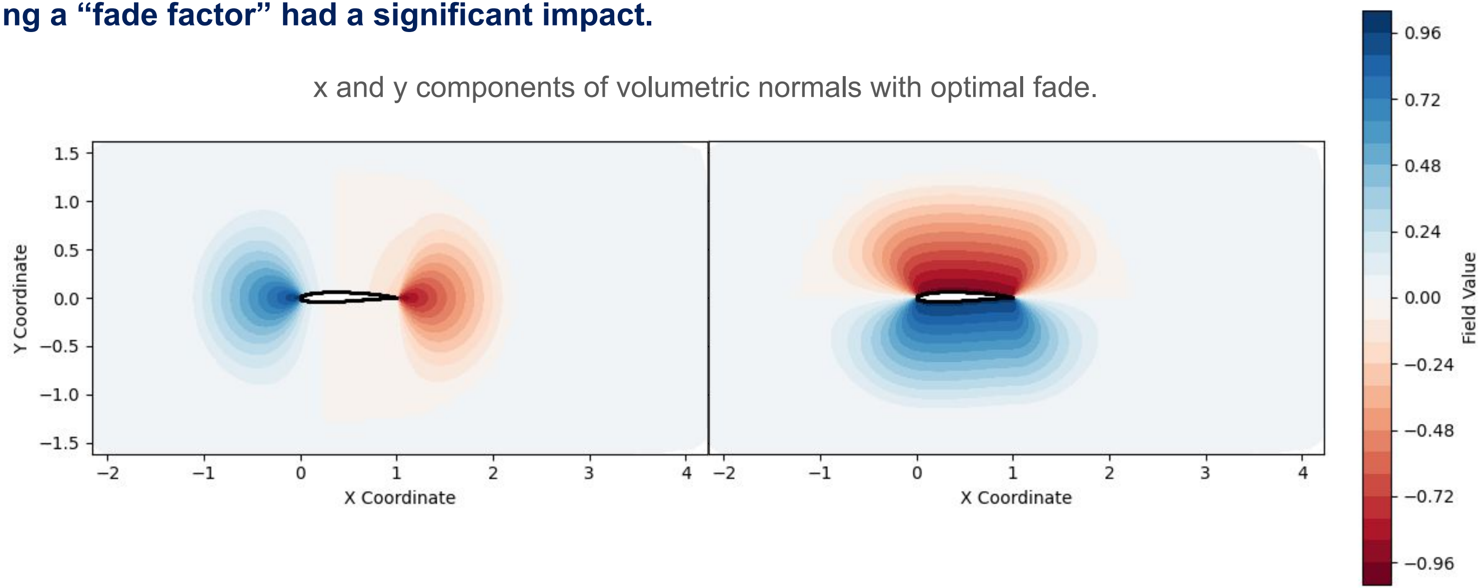
➔ ~~**Remove normals fields altogether**~~**.**
➔ **Process the normals fields to include direction to the closest point on the surface.**
➔ **Obtain an invariant reference frame with respect to the airfoil shape variation (directional sdf).**



Input Fields: feature engineering continued…

Introducing a “fade factor” had a significant impact.

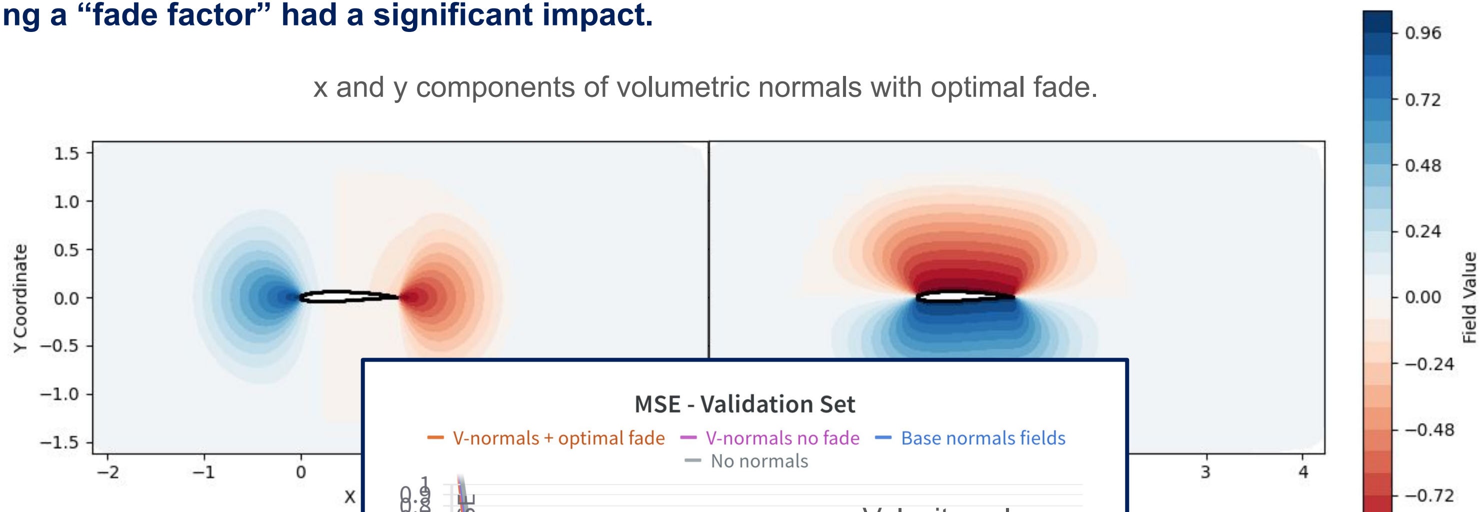
x and y components of volumetric normals with optimal fade.

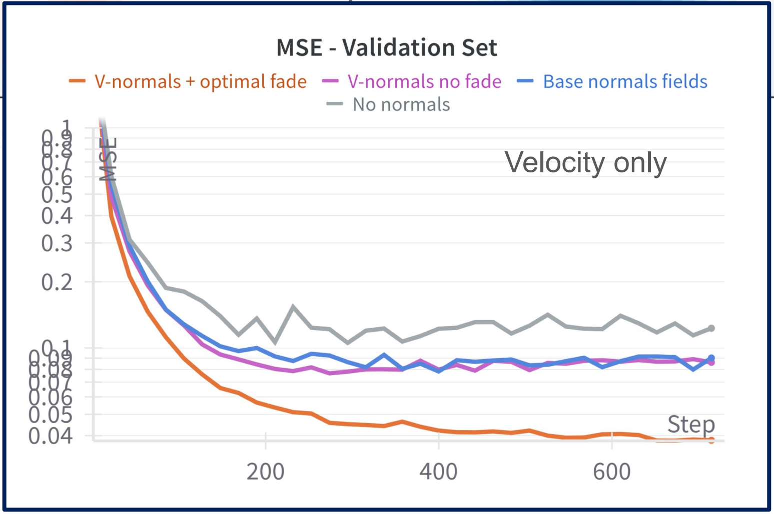


Input Fields: feature engineering continued…

Introducing a “fade factor” had a significant impact.

x and y components of volumetric normals with optimal fade.





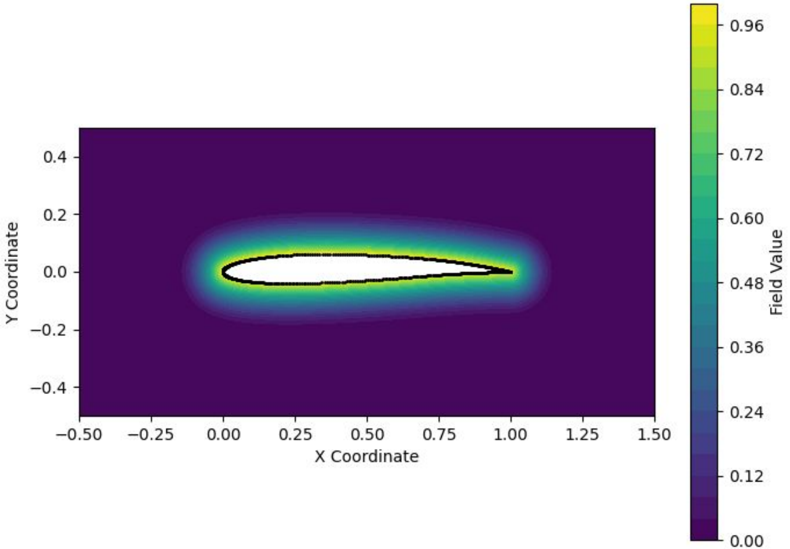
Input Fields: feature engineering continued…

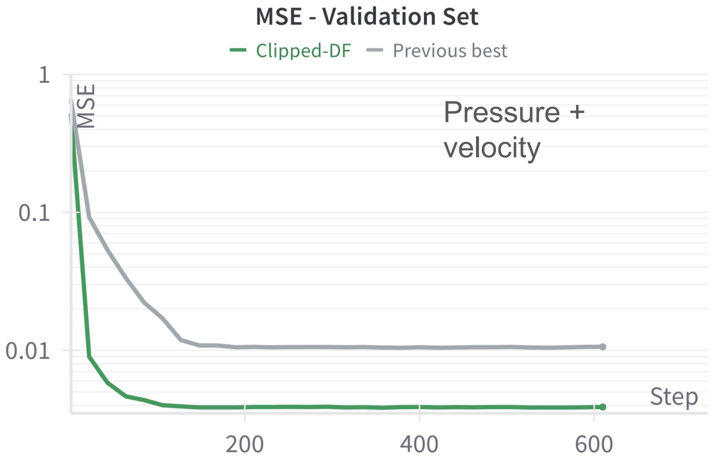
- Training the various physical fields together was leading to poor results.

- Mainly due to the difference in **gradients** between velocity and pressure in the **boundary layer**.

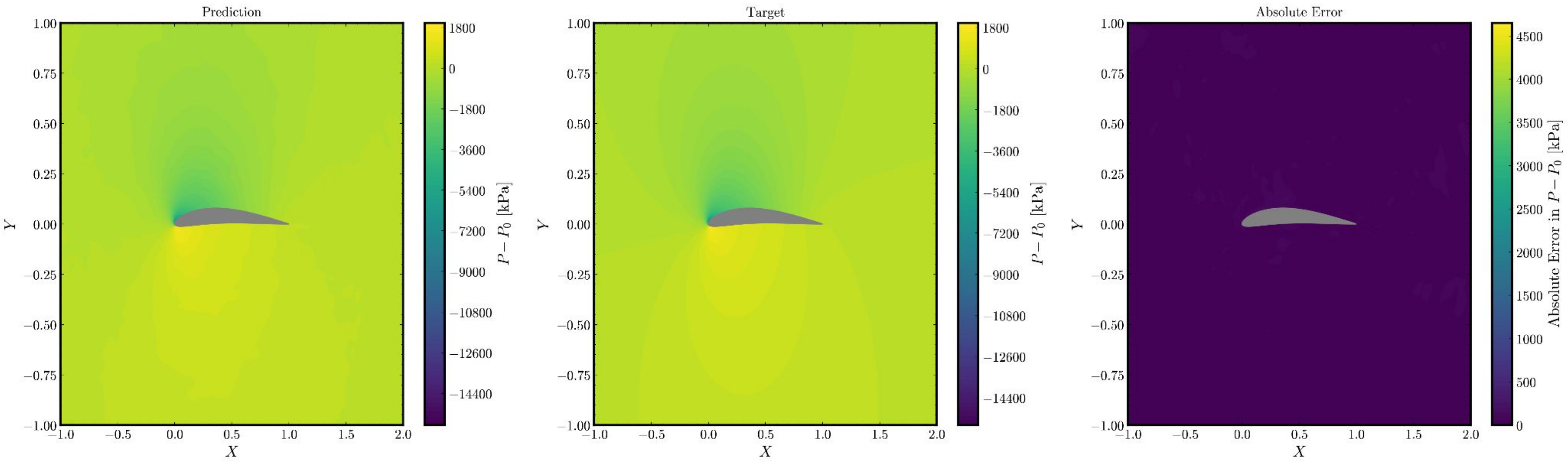
Input Fields: feature engineering continued…

- Training the various physical fields together was leading to poor results.
- Mainly due to the difference in **gradient** between velocity and pressure in the **boundary layer**.
- The addition of a clipped distance-function as input lead to significant improvements.



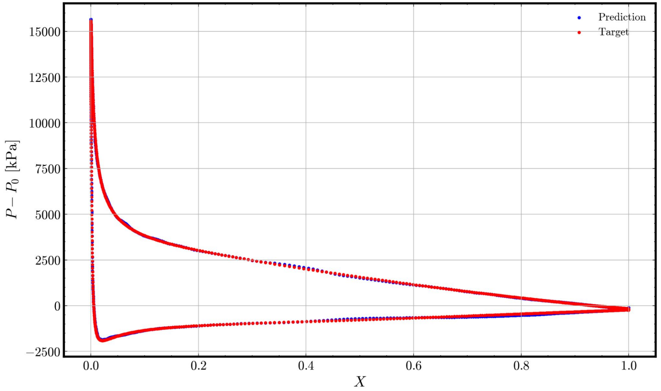


Results and Discussion

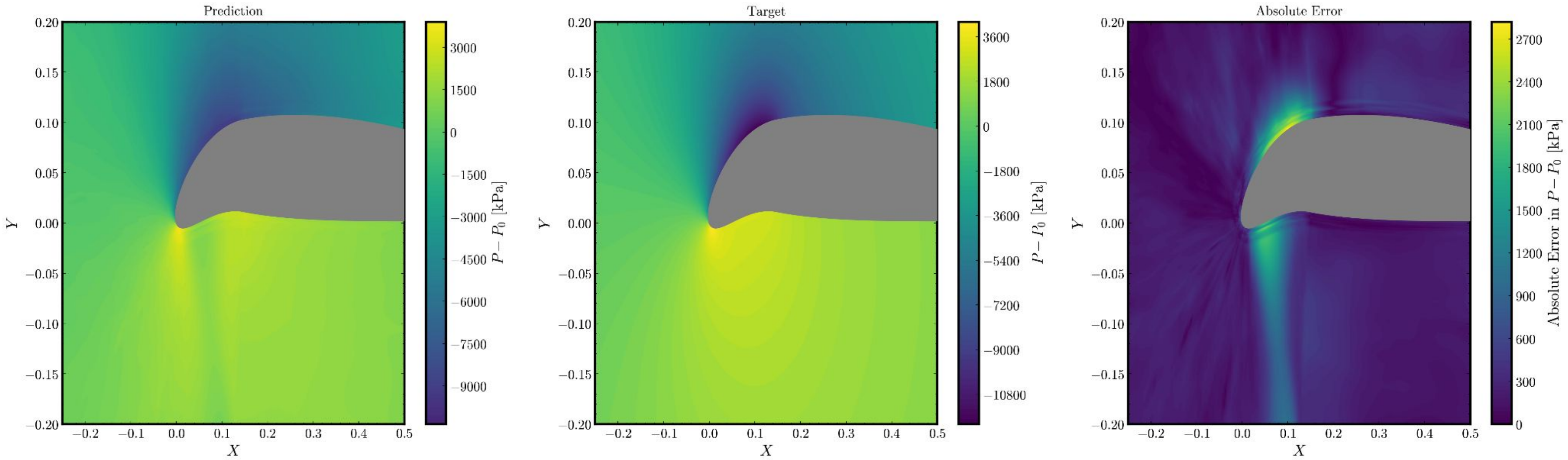


Low errors on simple geometries

- MARIO achieves great accuracy on simple geometries.
- The **main physical features** are well represented.
- It can be used on all NACA 4-5 airfoils with good reliability.

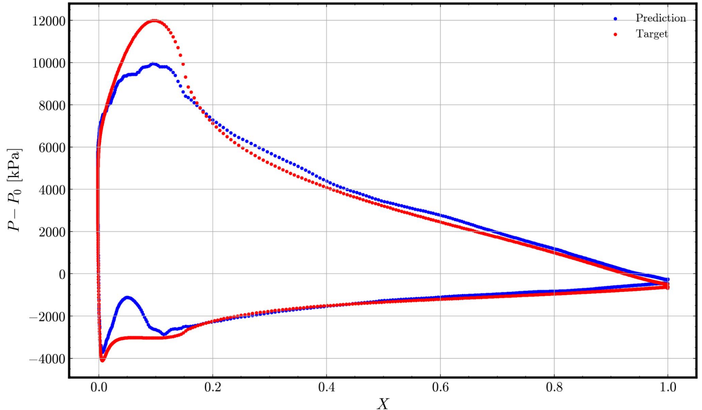


Results and Discussion



Higher errors on unseen geometries

- MARIO struggles to generalize on unseen flow patterns,
- The **MSE error** is driven by the **leading edge** region.



Results and Discussion

Metrics Breakdown

- MARIO 2.0 beats its predecessor (encode-process-decode) by an order of magnitude on MSE accuracy.
- MARIO 2.0 matches the performances of state of the art **deep learning** methods on the Airfrans dataset with **⅛ of the training data**!!!
- Drag coefficient remains difficult to predict accurately.
- The real speedup is much larger than the one measured by the evaluation system.

		MARIO 2.0		MARIO 1.0	
Category	Metric	Test	Test OOD	Test	Test OOD
MSE Volume	u_x	0.0029	0.0042	0.0439	0.0681
	u_y	0.0015	0.0027	0.0302	0.0481
	p	0.0058	0.0201	0.0737	0.1849
	ν_t	0.0278	0.0480	0.0992	0.3573
MSE Surface	p	0.0174	0.0704	0.1664	0.6580
Physics	Spearman drag	0.6410	0.5614	0.3688	0.4197
	Spearman lift	0.9967	0.9925	0.9713	0.9602
	Mean rel. drag	0.3585	0.4455	1.5684	1.9172
	Std rel. drag	0.4762	0.4815	1.2613	1.4234
	Mean rel. lift	0.0712	0.1112	0.3018	0.4690
	Std rel. lift	0.1137	0.1772	0.5133	1.4113
Inference Time	Real Speedup	5000	5000	1000	1000

Beyond the ML4CFD Challenge

Thanks to the mesh-agnostic formulation, surrogate models based on Neural Fields:

- Are discretization invariant: can be trained at **much lower resolution** and perform **super-resolution** at test time.
- Can handle **geometric variations**, including complex **non-parametric** geometries in **3D**.
- Are **fast** at inference time (**5000x** speedup on the Challenge).

Beyond the ML4CFD Challenge

Thanks to the mesh-agnostic formulation, surrogate models based on Neural Fields:

- Are discretization invariant: can be trained at **much lower resolution** and perform **super-resolution** at test time.
- Can handle **geometric variations**, including complex **non-parametric** geometries in **3D**.
- Are **fast** at inference time (**5000x** speedup on the Challenge)

Neural Fields can be used to build scalable aerodynamic data-driven simulators:

- For real **industrial simulations** in 3D over wings and full aircraft simulations.
- For data fusion (experimental data and simulation).
- For shape optimization and multi-disciplinary-optimization (MDO).


scientific reports

Explore content ˅ About the journal ˅ Publish with us ˅

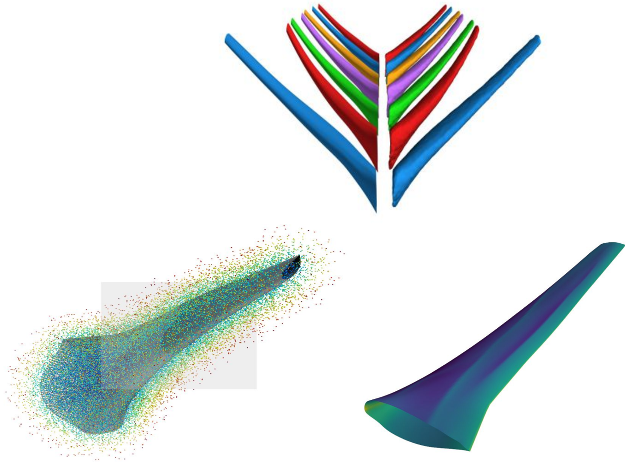
nature › scientific reports › articles › article

Article | Open access | Published: 26 October 2024

Neural fields for rapid aircraft aerodynamics simulations

Giovanni Catalani, Siddhant Agarwal, Xavier Bertrand, Frédéric Tost, Michael Bauerheim & Joseph Morlier

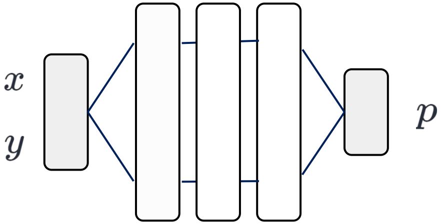
Scientific Reports **14**, Article number: 25496 (2024) | Cite this article



Thanks for listening!

Implicit Neural Representations: continuous representation of data.

Neural Networks can be used as a continuous approximation of signals on general domains: the value of the signal at any spatial input location can be obtained as the output of an **Implicit Neural Representation** (INRs).



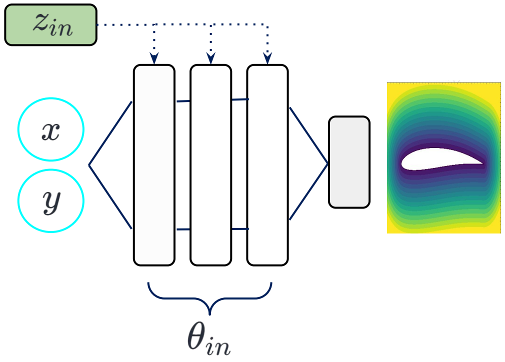
Training a Neural Network: finding the optimal parameters $\theta = \{W_l, b_l\}_{l=1,..,n_l}$ that minimize the reconstruction error

$$\{(x_i, y_i), p_i)\}_{i=1,\dots N} \longrightarrow p = f_\theta(x, y)$$

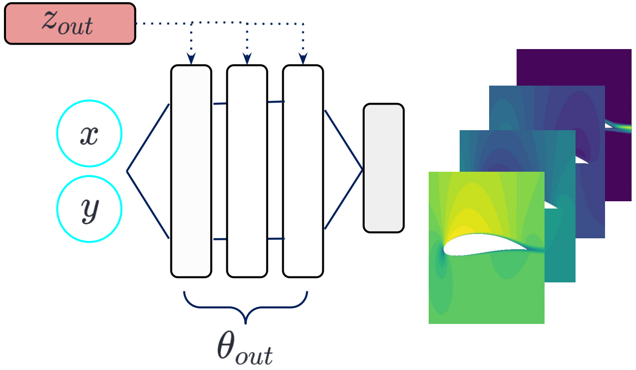
Once we fit a signal with a Neural Network, we can query the Network at any spatial location: we have a **continuous representation.**

Encode-Process-Decode

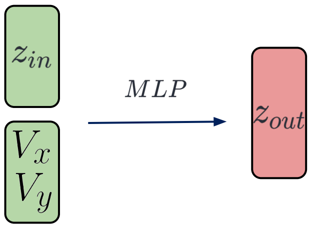
Input INR



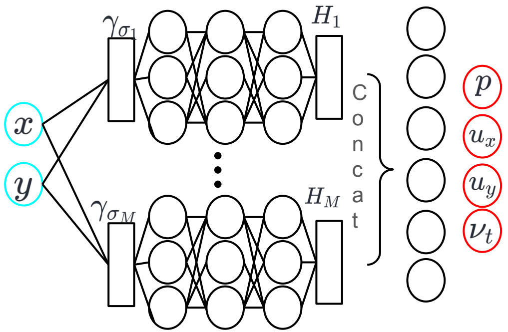
Output INR



Latent Dynamics



INR Architecture: Multiscale Fourier Features MLP



[7] Serrano, Louis, et al. "INFINITY: Neural Field Modeling for Reynolds-Averaged Navier-Stokes Equations." *arXiv e-prints* (2023): arXiv-2307.

ML4CFD Neurips Challenge

https://ml-for-physical-simulation-challenge.irt-systemx.fr/

Objectives:

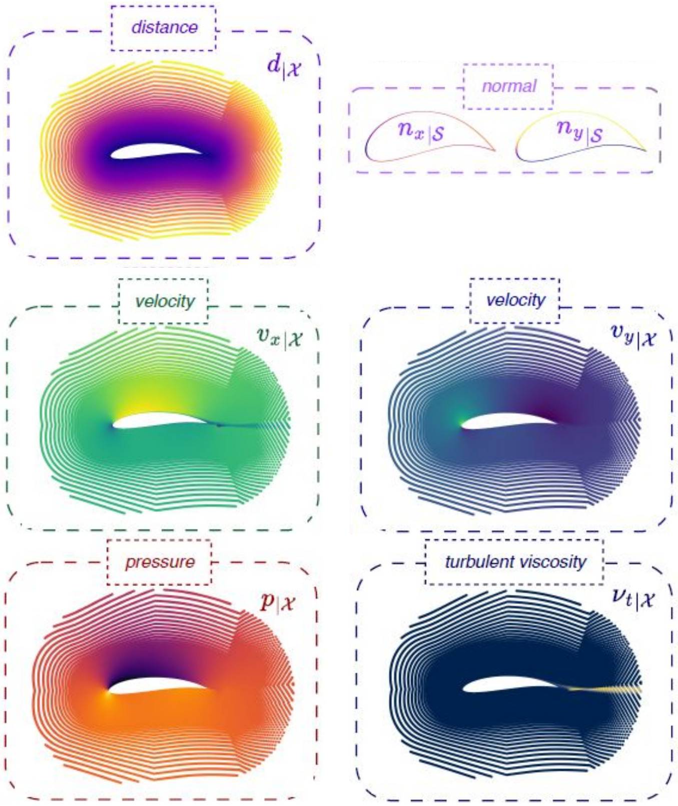
- Open international competition aimed at developing Data Driven physical simulators of Computational Fluid Dynamics.
- Task: **Predict surfacic & volumic fields** around unseen airfoils at test Reynolds numbers, Mach and Angles of Attack.
- Really small training dataset with only 100 CFD computations.

Evaluation metrics:

- Accuracy on prediction on and off the airfoil surface.
- Physical compliance: accuracy in prediction of lift and drag.
- Speed: acceleration compared to CFD solver.
- Out of Distribution performance: tested on ood geometries and ood flow regimes.

Results:

- Our approach based on Neural Fields positioned **3rd** among more than 200 teams.
- **Times 5000 speedup** at inference compared to the high fidelity simulator.



AIRBUS ISAE SUPAERO NEURAL INFORMATION PROCESSING SYSTEMS