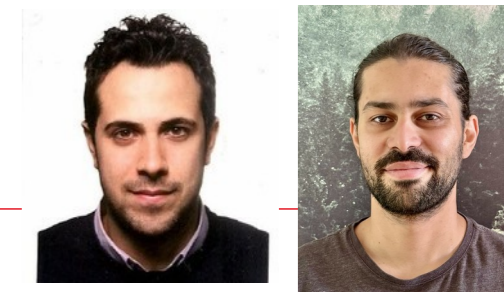


DEEP LEARNING FOR NUMERICAL SIMULATIONS : 2 APPROACHES FOR ML4CFD



Matthieu Nastorg's PhD thesis + towards Large Physics Models

Collaborators:

- Matthieu **Nastorg**, Theofanis **Ifaistos**, Guillaume **Charpiat**, Marc **Schoenauer** (TAU team, LISN / INRIA Saclay)
- Thibault **Faney**, Jean-Marc **Gratien** (IFP Énergies Nouvelles)
- Michele-Alessandro **Bucci** (TAU then Safran Tech)
- Emmanuel **Menier** (IRT-Systemx / TAU)

NB: Matthieu Nastorg and Emmanuel Menier founded a start-up together, NP Co (Augur), after their PhD defences.

Computational Fluid Dynamics (CFD)

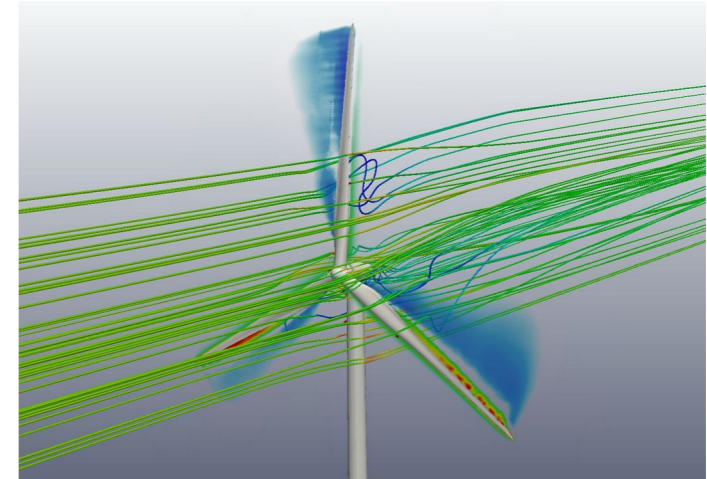
Goal : speed up numerical simulations of fluid flows

Two approaches :

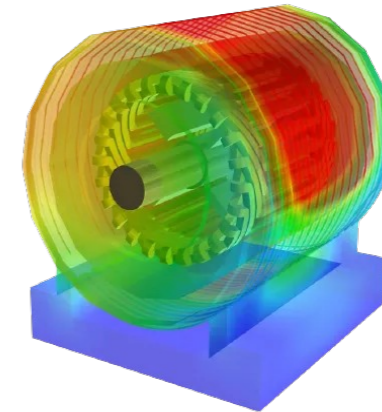
Incorporating physical knowledge vs Brute force data
ex. : GraphCast (GNN) vs ClimaX (Transformer)

Which is best ? depends on your goal.

- 1) Hybridize classical solvers & machine learning
→ speed-up, with convergence guarantees !
- 2) Fully machine learning
→ fast predictions !



Wind turbine design



Electric motor cooling design

1) Hybridize classical solvers & machine learning

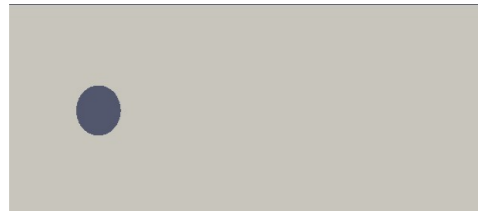


What / where / how ?

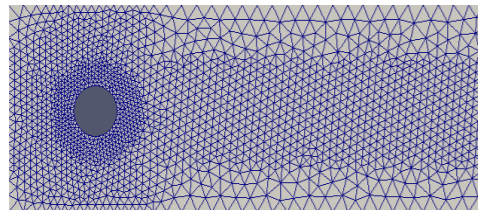


What ?

Pre-Processing



Continuous domain

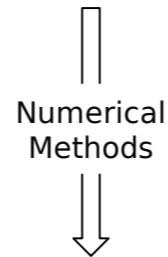


Discretized domain (Mesh)

Solver

$$\begin{cases} \frac{\partial u}{\partial t} + u \cdot \nabla u = -\frac{1}{\rho} \nabla p + \nu \Delta u + f \\ \nabla \cdot u = 0 \end{cases} + \text{Initial \& Boundary conditions}$$

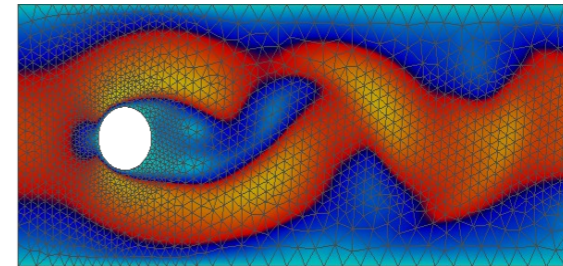
Continuous Inc. Navier-Stokes equations



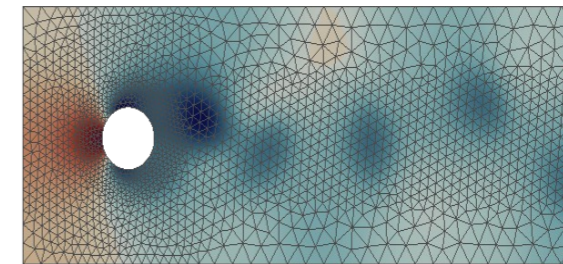
System of discretized equations
(of size at least the nb.of mesh points)



Post-Processing



Velocity magnitude at time $t = 3s$.
Mesh at the background



Pressure at time $t = 3s$.
Mesh at the background



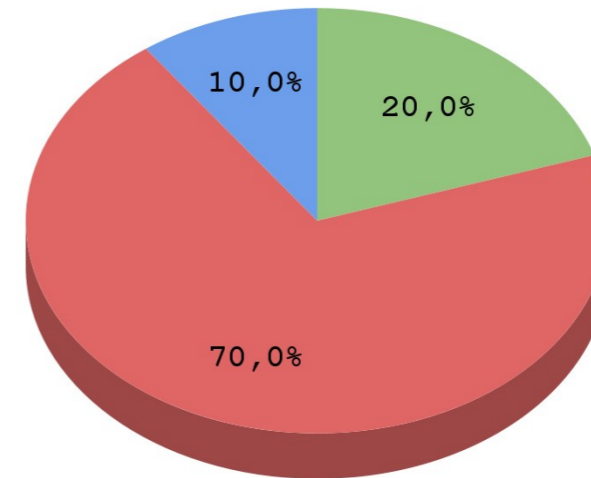
Splitting Schemes

Step 1: Compute tentative velocity \mathbf{u}^*

Step 2: Solve **Poisson Pressure Problem**

$$\Delta p^{n+1} = \Delta p^n + \frac{\rho}{\delta t} \nabla \cdot \mathbf{u}^*$$

Step 3: Update velocity



Splitting scheme time breakdown

Machine Learning model to **enhance** the resolution of **Poisson Pressure** problems

- Wang, Y. (2015). Solving incompressible Navier-Stokes equations on heterogeneous parallel architectures (Doctoral dissertation, Université Paris Sud-Paris XI).
- Langtangen, H. P., Mardal, K. A., & Winther, R. (2002). Numerical methods for incompressible viscous flow. Advances in water Resources, 25(8-12), 1125-1146.



Continuous Poisson problems

$$\left\{ \begin{array}{l} -\Delta u_p = f_p \quad \in \Omega_p \\ u_p = g_p \quad \in \partial\Omega_{p,D} \\ \frac{\partial u_p}{\partial n} = 0 \quad \in \partial\Omega_{p,N} \end{array} \right. \xrightarrow{\text{Numerical Method}}$$

Machine Learning Solver 

$$\hat{\mathbf{u}}_p = \text{GNN-Solver}_\theta (E_{p,h})$$

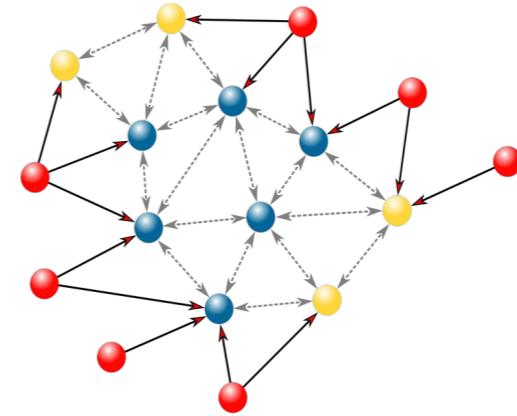
for any shape and boundary conditions !

Build on :

- graph-NN (to handle meshes)
- Deep Statistical Solvers (NeurIPS 2020) by A&O team
- Deep Equilibrium Models (for iterative process)
- domain decomposition (hierarchy for scaling-up)

Discretized Poisson problems

$$A_p \mathbf{u}_p = \mathbf{b}_p$$



● Interior ● Dirichlet ● Neumann

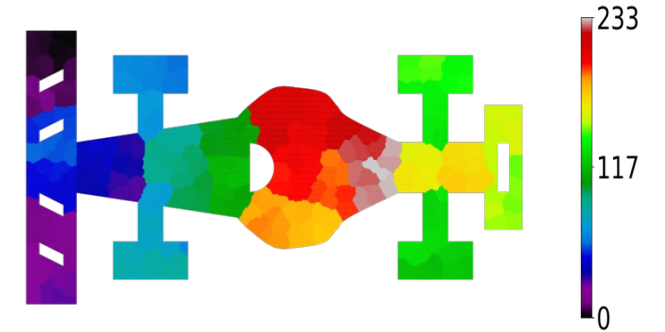
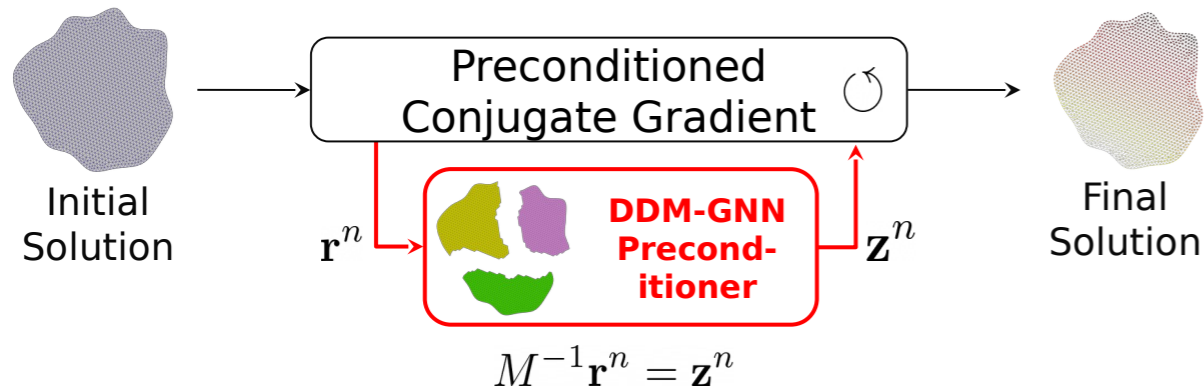


How ?

Training (PINNs) :

$$\theta^* = \underset{\theta}{\operatorname{argmin}} \mathbb{E}_{p \sim \mathcal{D}} \left[\frac{1}{N_p} \|A_p \hat{\mathbf{u}}_p - \mathbf{b}_p\|^2 \right]$$

Guaranteed Accuracy : Preconditioned Conjugate Gradient



Conjugate gradient :

$\mathbf{r}_0 := \mathbf{b} - \mathbf{A}\mathbf{x}_0$
if \mathbf{r}_0 is sufficiently small, then return \mathbf{x}_0 as the result

$\mathbf{p}_0 := \mathbf{r}_0$
 $k := 0$

repeat

$$\alpha_k := \frac{\mathbf{r}_k^\top \mathbf{r}_k}{\mathbf{p}_k^\top \mathbf{A} \mathbf{p}_k}$$

$$\mathbf{x}_{k+1} := \mathbf{x}_k + \alpha_k \mathbf{p}_k$$

$$\mathbf{r}_{k+1} := \mathbf{r}_k - \alpha_k \mathbf{A} \mathbf{p}_k$$

if \mathbf{r}_{k+1} is sufficiently small, then exit loop

$$\beta_k := \frac{\mathbf{r}_{k+1}^\top \mathbf{r}_{k+1}}{\mathbf{r}_k^\top \mathbf{r}_k}$$

$$\mathbf{p}_{k+1} := \mathbf{r}_{k+1} + \beta_k \mathbf{p}_k$$

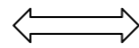
$$k := k + 1$$

end repeat

return \mathbf{x}_{k+1} as the result

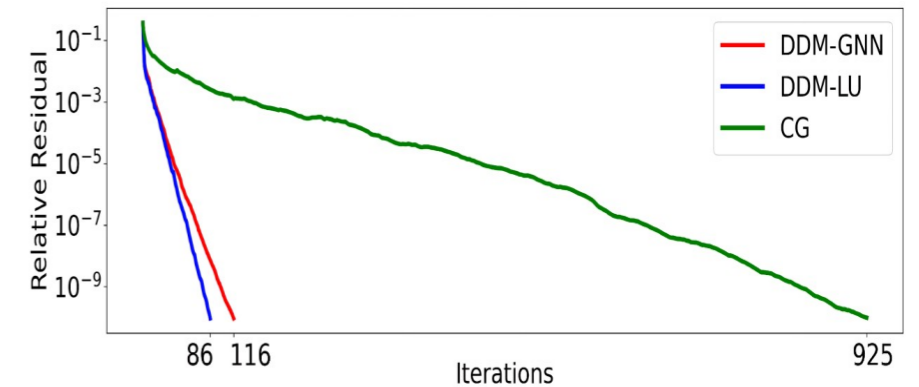
Preconditioner :

$$\mathbf{A} \mathbf{u} = \mathbf{b}$$



$$\mathbf{M}^{-1} \mathbf{A} \mathbf{u} = \mathbf{M}^{-1} \mathbf{b}$$

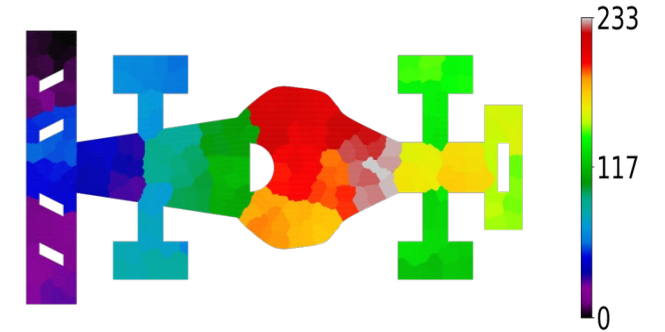
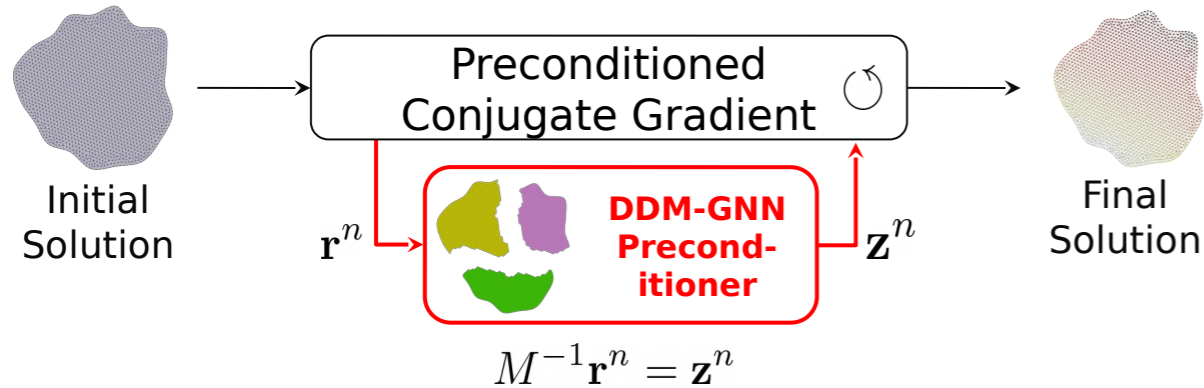
Solve **preconditioned** system
= **enhanced convergence**
of the iterative method



Evolution of the relative residual error for the resolution of a large-scale out-of-distribution Poisson problem using CG, PCG-DDM-GNN and PCG-DDM-LU

[Multi-Level GNN Preconditioner for Solving Large Scale Problems, IPDPSW 2024]
[An Implicit GNN Solver for Poisson-like Problems, Comp. and Math. with Appli, 2024]
[DS-GPS: A Deep Statistical Graph Poisson Solver, ML&physics w at NeurIPS 2022]

Guaranteed Accuracy : Preconditioned Conjugate Gradient



Conjugate gradient :

$\mathbf{r}_0 := \mathbf{b} - \mathbf{A}\mathbf{x}_0$
 Solve: $\mathbf{M}\mathbf{z}_0 := \mathbf{r}_0$
 $\mathbf{p}_0 := \mathbf{z}_0$
 $k := 0$
repeat
 $\alpha_k := \frac{\mathbf{r}_k^\top \mathbf{z}_k}{\mathbf{p}_k^\top \mathbf{A}\mathbf{p}_k}$
 $\mathbf{x}_{k+1} := \mathbf{x}_k + \alpha_k \mathbf{p}_k$
 $\mathbf{r}_{k+1} := \mathbf{r}_k - \alpha_k \mathbf{A}\mathbf{p}_k$
if \mathbf{r}_{k+1} is sufficiently small **then** exit loop **end if**
 Solve $\mathbf{M}\mathbf{z}_{k+1} := \mathbf{r}_{k+1}$
 $\beta_k := \frac{\mathbf{r}_{k+1}^\top \mathbf{z}_{k+1}}{\mathbf{r}_k^\top \mathbf{z}_k}$
 $\mathbf{p}_{k+1} := \mathbf{z}_{k+1} + \beta_k \mathbf{p}_k$
 $k := k + 1$

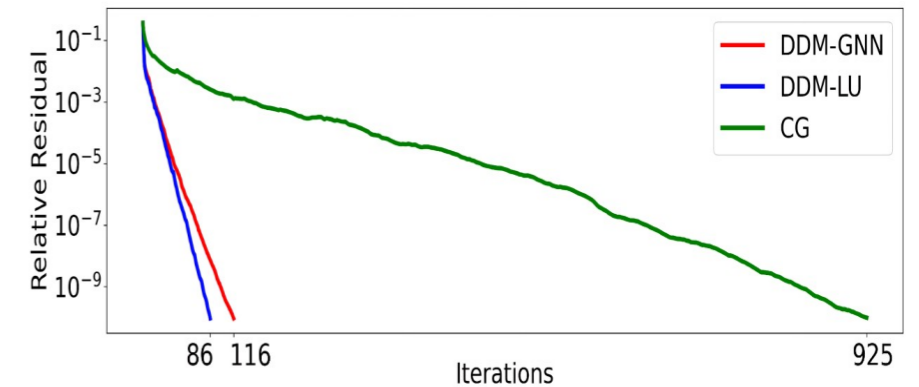
Preconditioner :

$$\mathbf{A}\mathbf{u} = \mathbf{b}$$

$$\iff$$

$$\mathbf{M}^{-1}\mathbf{A}\mathbf{u} = \mathbf{M}^{-1}\mathbf{b}$$

Solve **preconditioned** system
 = **enhanced convergence**
 of the iterative method



Evolution of the relative residual error for the resolution of a large-scale out-of-distribution Poisson problem using CG, PCG-DDM-GNN and PCG-DDM-LU

[Multi-Level GNN Preconditioner for Solving Large Scale Problems, IPDPSW 2024]
 [An Implicit GNN Solver for Poisson-like Problems, Comp. and Math. with Appli, 2024]
 [DS-GPS: A Deep Statistical Graph Poisson Solver, ML&physics w at NeurIPS 2022]

end repeat

The result is \mathbf{x}_{k+1}



Goal : speed up

Two approaches :

1) Hybridize classical solvers & machine learning
→ speed-up, with convergence guarantees !

2) Fully machine learning
→ **fast predictions !**

2) Fully machine learning

→ fast predictions !

INRIA start-up *NP Co*
(was: *Augur*)



Emmanuel Menier



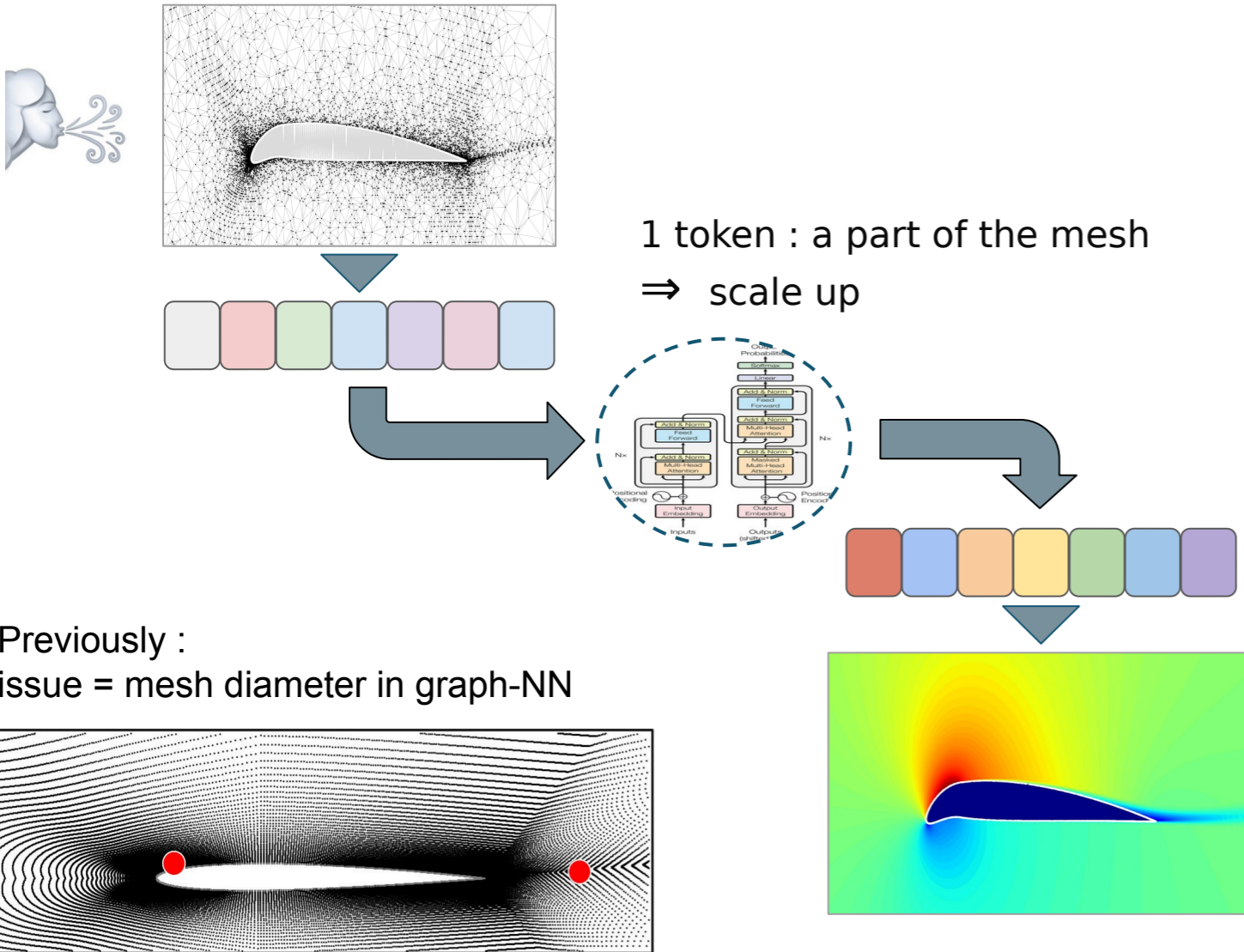
Matthieu Nastorg



Alice Lacan

3 former PhD students :

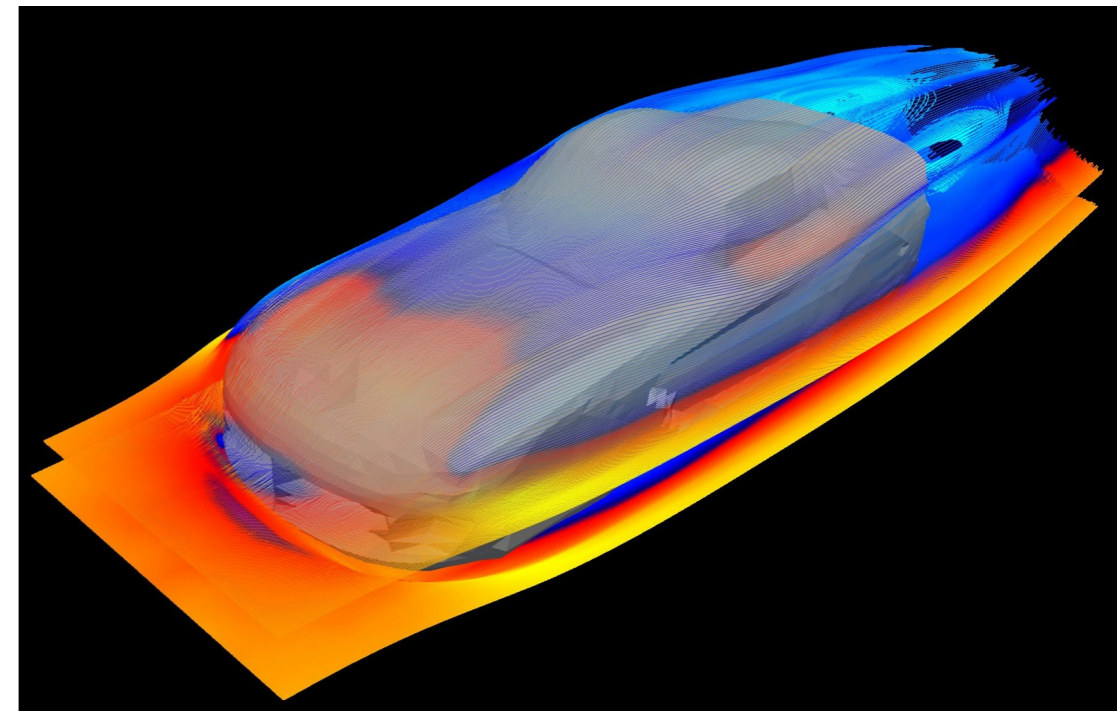
Transformers for CFD



Simulation speed : a game changer

Standard simulation time : hours
 ML approach : 20ms

⇒ enables
 shape
 optimization



ML4CFD conclusion

Two approaches :

- 1) Hybridize classical solvers & machine learning
→ speed-up, with convergence guarantees !
- 2) Fully machine learning
→ fast predictions !

Best one ? depends on your goal.

Flexibility :

- Can adapt to arbitrary geometries
- Can handle varying boundary and simulation conditions

Scalability :

- Can scale to high dimension (>100K points)
- Can scale to large datasets
- Transformer optimisations can be used (flash attention, ZeRO parallelization ...)

Perspectives

Research tracks :

- Multi-scale (graph-NN / transformer)
+ scalability (3D)
- Dynamical systems (unsteady flows)
- PINNs
- Foundations models for CFD : Large Physics Models

Foundation Models for CFD

Dream goal : LPMs : Large Physics Models

Large model :

- typically : transformer

Train on different tasks :

- various shapes and meshes
- various boundary conditions
- *various dynamical systems (diffusion, advection...)*
or parameters of a same equation (viscosity...)

Transfer :

- take pre-trained model
 - fine-tune it
- (as for an LLM)



Theofanis Ifaistos's ongoing PhD

Action exploratoire INRIA

Post-doc position available !

[Multiple Physics Pretraining for Physical Surrogate Models, McCabe et al, 2023]

Questions ?



Publications :

[Multi-Level GNN Preconditioner for Solving Large Scale Problems, IPDPSW 2024]

[An Implicit GNN Solver for Poisson-like Problems, Comp. and Math. with Appli, 2024]

[DS-GPS: A Deep Statistical Graph Poisson Solver, ML&physics w at NeurIPS 2022]

[Deep Statistical Solvers, Donon et al, NeurIPS 2020]

Generalization Capabilities of Transformers in Fluid Dynamics

Theofanis Ifaistos¹,
Matthieu Nastorg², Emmanuel Menier²,
Guillaume Charpiat¹, Marc Schoenauer¹

¹ TAU team, LISN, Université Paris-Saclay, CNRS, Inria

² Augur SAS · Inria spinoff

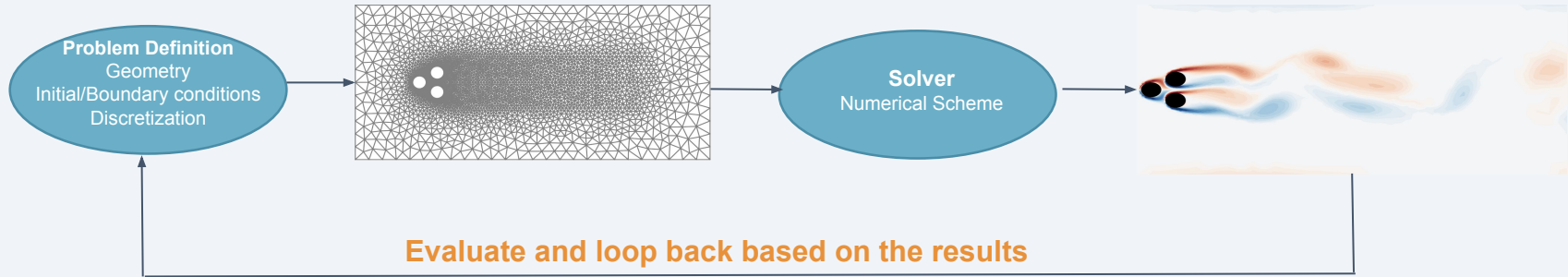


Introduction

Computational Fluid Dynamics

Computational Fluid Dynamics

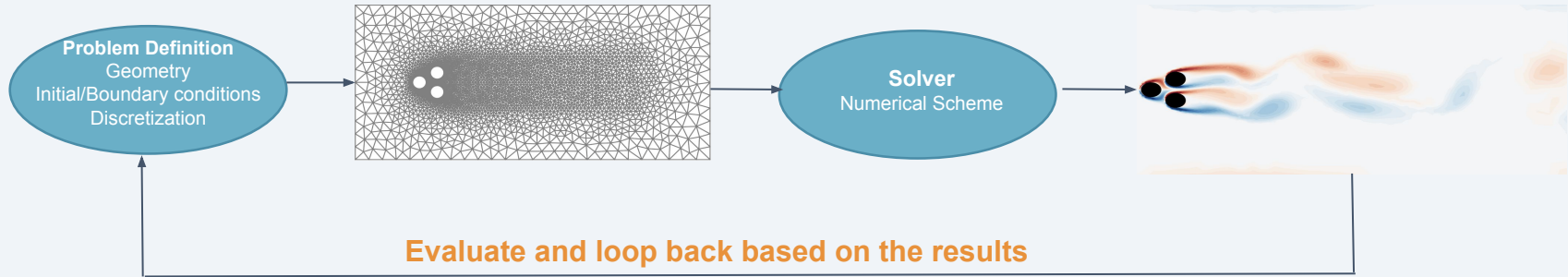
The **Computational Fluid Dynamics** (CFD) pipeline



Unstructured mesh fluidic pinball configuration, Menier et al. Computer Methods in Applied Mechanics and Engineering 2023

Computational Fluid Dynamics

The **Computational Fluid Dynamics** (CFD) pipeline



Unstructured mesh fluidic pinball configuration, Menier et al. Computer Methods in Applied Mechanics and Engineering 2023

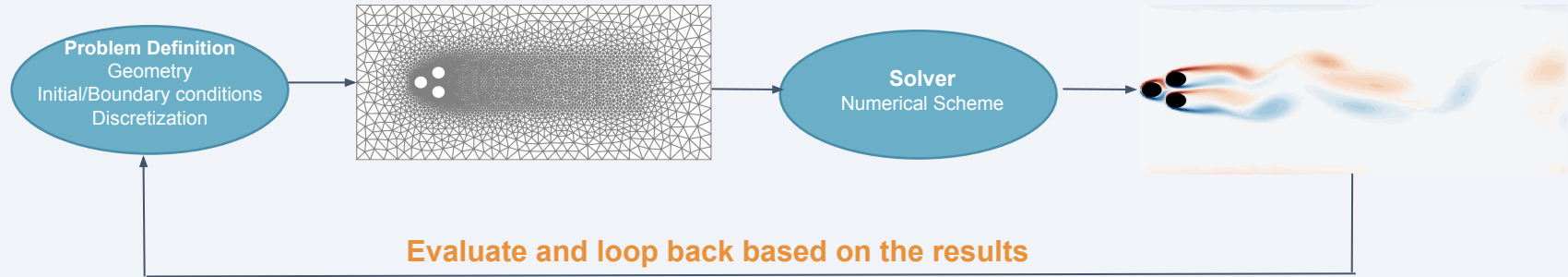


High computational cost

Long compute times and expensive hardware

Computational Fluid Dynamics

The **Computational Fluid Dynamics** (CFD) pipeline



Unstructured mesh fluidic pinball configuration, Menier et al. Computer Methods in Applied Mechanics and Engineering 2023



High computational cost

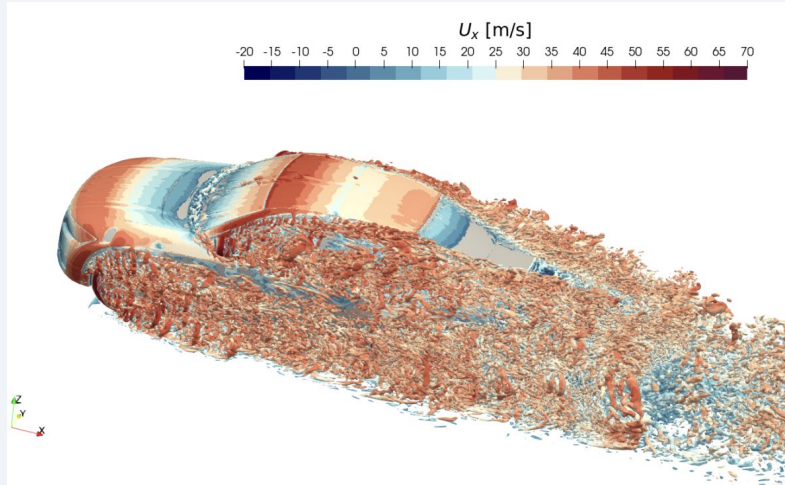
Long compute times and expensive hardware



Design experiments are limited

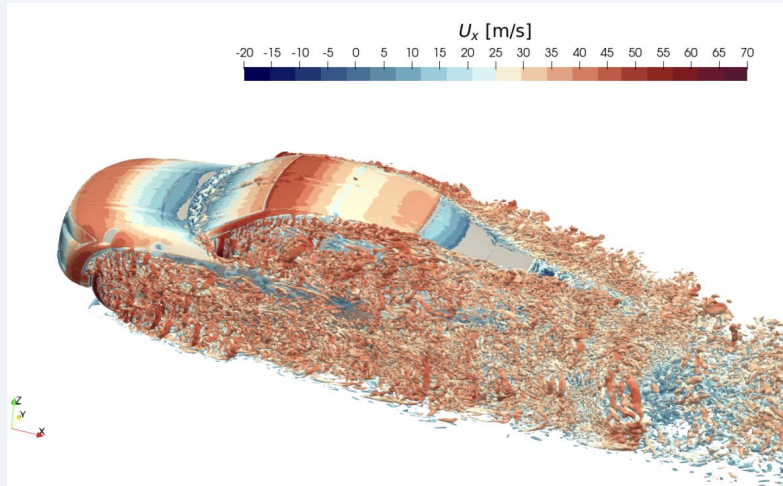
Multiple iterations can take days to week

Computational Fluid Dynamics



DrivaerML: LES simulations of car aerodynamics,
Ashton, N. et al. [arxiv.org](https://arxiv.org/abs/2025.01.08) 2025

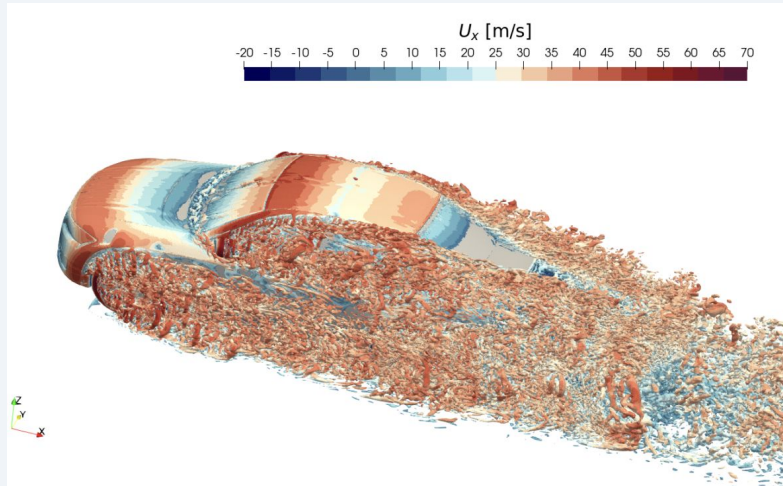
Computational Fluid Dynamics



160M volumetric cells

DrivaerML: LES simulations of car aerodynamics,
Ashton, N. et al. [arxiv.org](https://arxiv.org/abs/2025.01.08123) 2025

Computational Fluid Dynamics

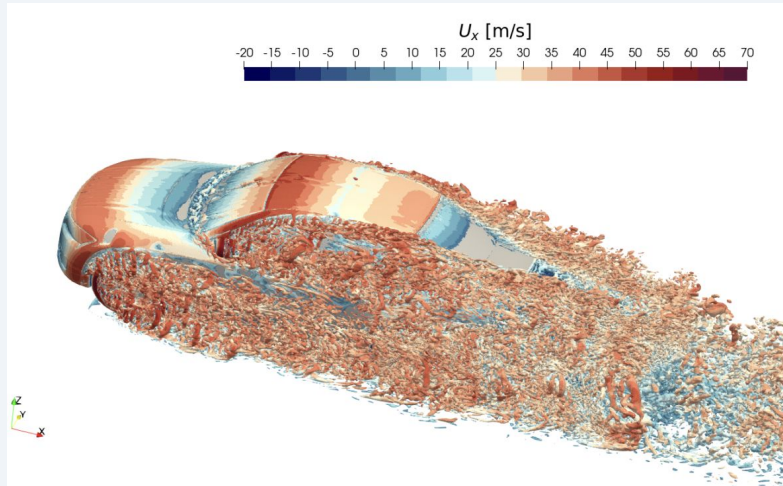


160M volumetric cells

8.8M surface cells

DrivaerML: LES simulations of car aerodynamics,
Ashton, N. et al. [arxiv.org](https://arxiv.org/abs/2025.01.08) 2025

Computational Fluid Dynamics



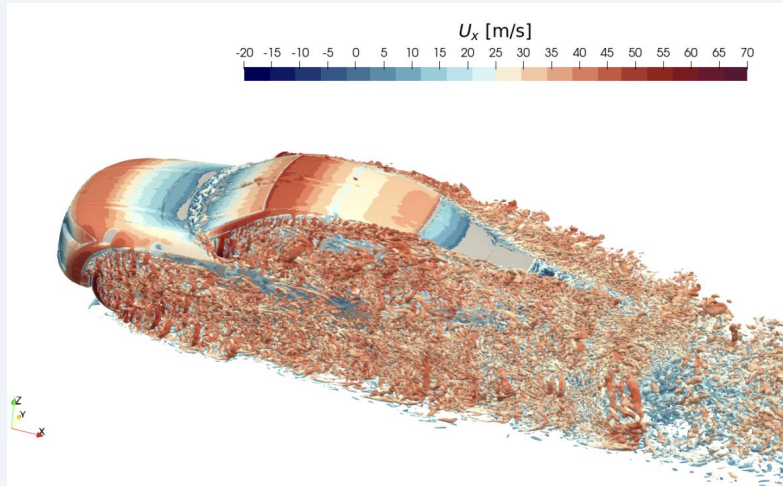
160M volumetric cells

8.8M surface cells

~40h per sim on 1,536 cores

DrivaerML: LES simulations of car aerodynamics,
Ashton, N. et al. [arxiv.org](https://arxiv.org/abs/2025.01.08101) 2025

Computational Fluid Dynamics



DrivaerML: LES simulations of car aerodynamics,
Ashton, N. et al. [arxiv.org](https://arxiv.org/abs/2025.01.08141) 2025

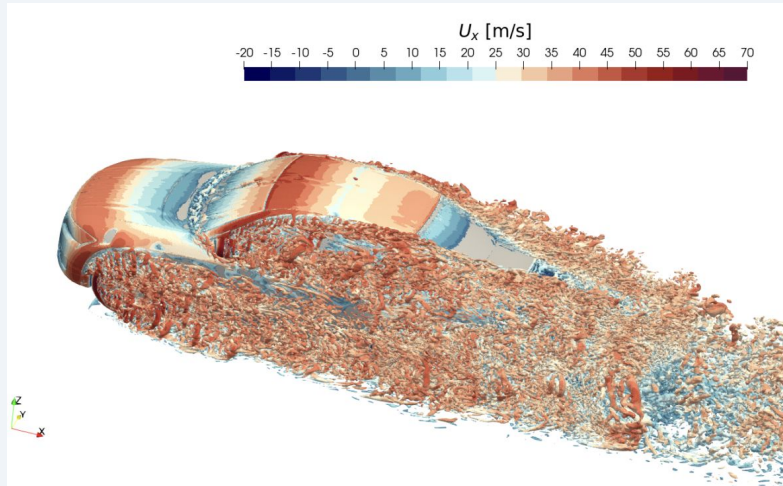
160M volumetric cells

8.8M surface cells

~40h per sim on 1,536 cores

31 TB total dataset of 500 simulations

Computational Fluid Dynamics



DrivaerML: LES simulations of car aerodynamics,
Ashton, N. et al. [arxiv.org](https://arxiv.org/abs/2025.01.08) 2025

160M volumetric cells

8.8M surface cells

~40h per sim on 1,536 cores

31 TB total dataset of 500 simulations

High-fidelity CFD remains extremely expensive • motivating ML-based surrogate models.

Transformers & the Attention Mechanism

Transformer models:

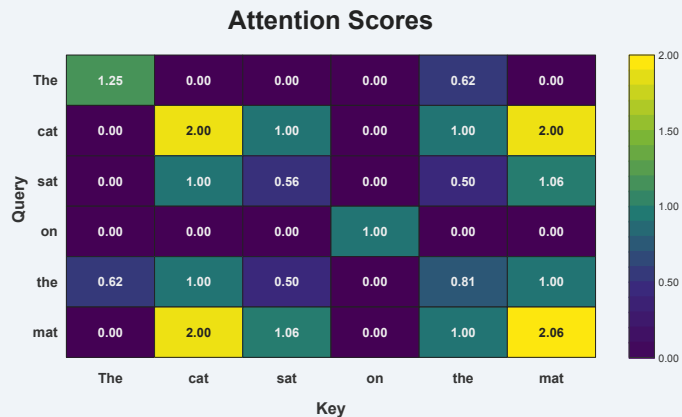
Highly parallelizable architectures based on the **Attention Mechanism**

Transformers & the Attention Mechanism

Transformer models:

Highly parallelizable architectures based on the **Attention Mechanism**

- Computes **relevance scores** between input tokens



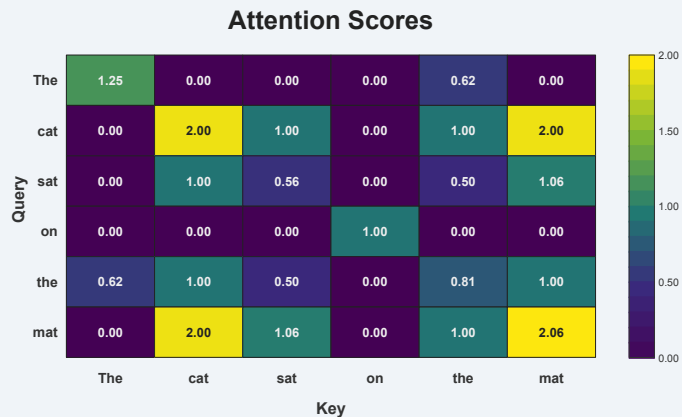
Self-attention scores of a sentence

Transformers & the Attention Mechanism

Transformer models:

Highly parallelizable architectures based on the **Attention Mechanism**

- Computes **relevance scores** between input tokens



Cross attention:

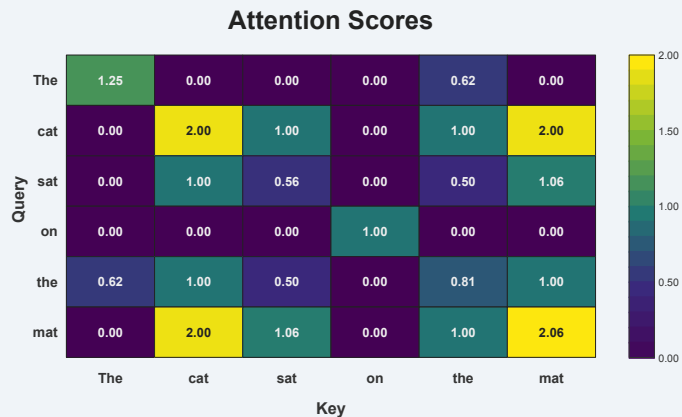
- Scores between two different sequences
- Translation: French word queries the English sentence for predicting the next token

Transformers & the Attention Mechanism

Transformer models:

Highly parallelizable architectures based on the **Attention Mechanism**

- Computes **relevance scores** between input tokens



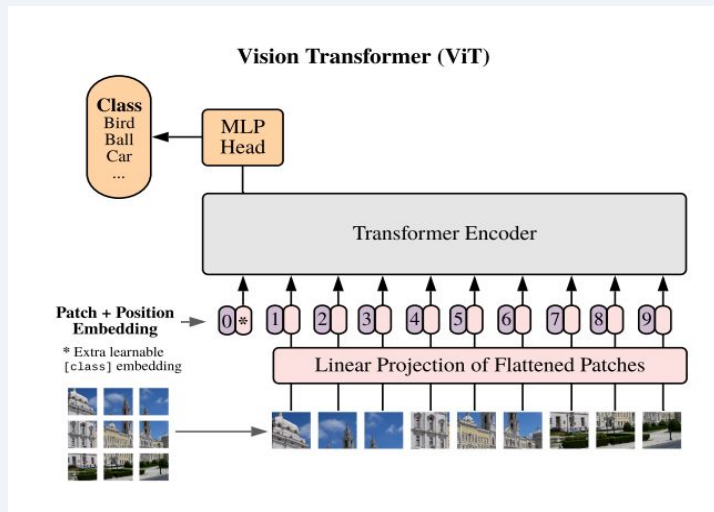
Cross attention:

- Scores between two different sequences
- Translation: French word queries the English sentence for predicting the next token

Capturing long-range interactions between nodes

From ViT to Mesh-Based Transformers

Vision Transformer (ViT) - Application of Transformers to images



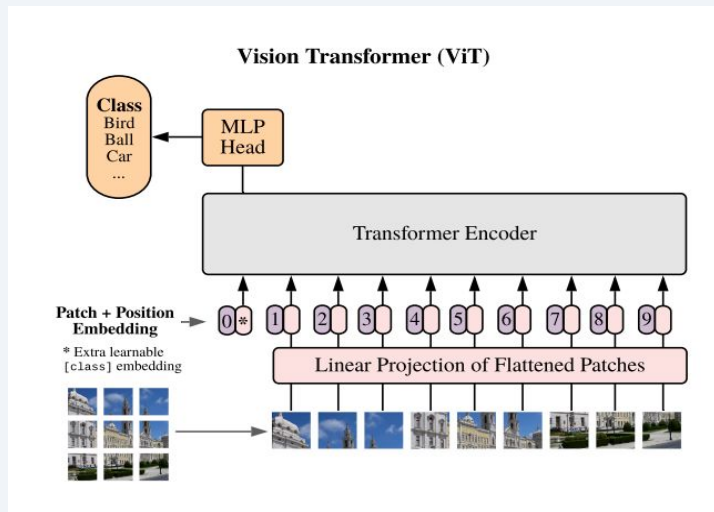
Patch-based tokenization, Dosovitskiy et al. ICLR 2021

¹ Wu et al., "Transsolver: A fast transform solver for PDEs on general geometries", ICML 2024

² Alkin et al., "Universal physics transformers: A framework for efficiently scaling neural operators", NeurIPS 2024

³ Wen et al., "Geometry aware operator transformer as an efficient and accurate neural surrogate for PDEs", NeurIPS 2025

From ViT to Mesh-Based Transformers



Patch-based tokenization, Dosovitskiy et al. ICLR 2021

Vision Transformer (ViT) - Application of Transformers to images

Grid is partitioned to patches downscaling the input

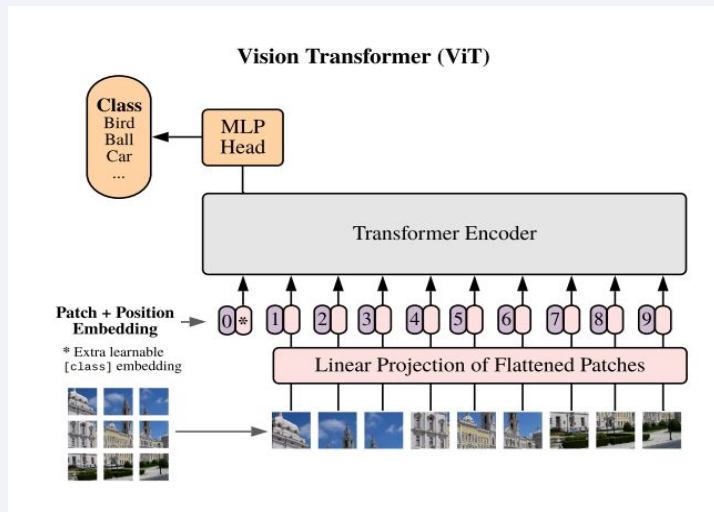
Cost reduced from **quadratic on pixels** to **quadratic on patches**

¹ Wu et al., "Transsolver: A fast transform solver for PDEs on general geometries", ICML 2024

² Alkin et al., "Universal physics transformers: A framework for efficiently scaling neural operators", NeurIPS 2024

³ Wen et al., "Geometry aware operator transformer as an efficient and accurate neural surrogate for PDEs", NeurIPS 2025

From ViT to Mesh-Based Transformers



Patch-based tokenization, Dosovitskiy et al. ICLR 2021

Vision Transformer (ViT) - Application of Transformers to images

Grid is partitioned to patches downscaling the input

Cost reduced from **quadratic on pixels** to **quadratic on patches**

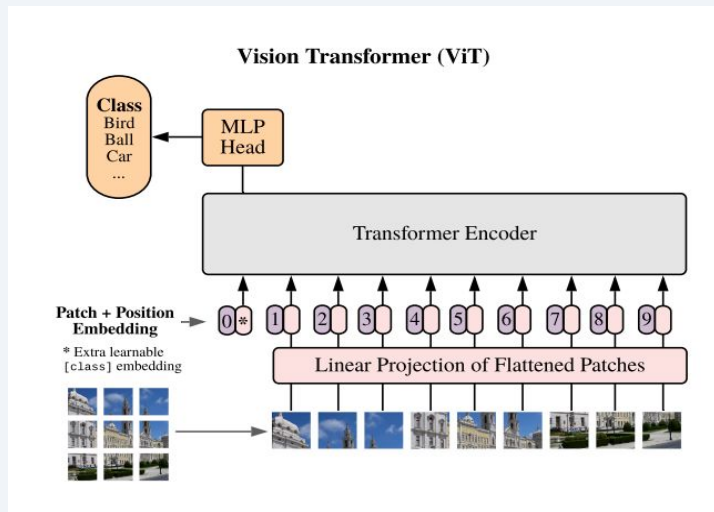
- Cannot be readily applied to irregular meshes as they do are not regularly structured

¹ Wu et al., "Transsolver: A fast transform solver for PDEs on general geometries", ICML 2024

² Alkin et al., "Universal physics transformers: A framework for efficiently scaling neural operators", NeurIPS 2024

³ Wen et al., "Geometry aware operator transformer as an efficient and accurate neural surrogate for PDEs", NeurIPS 2025

From ViT to Mesh-Based Transformers



Patch-based tokenization, Dosovitskiy et al. ICLR 2021

Vision Transformer (ViT) - Application of Transformers to images

Grid is partitioned to patches downscaling the input

Cost reduced from **quadratic on pixels** to **quadratic on patches**

- Cannot be readily applied to irregular meshes as they do are not regularly structured

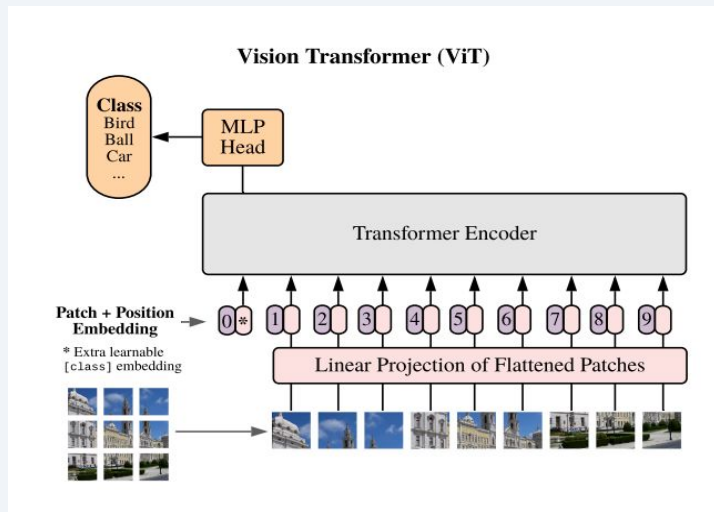
Transolver¹,
UPT², GAOT³

¹ Wu et al., "Transsolver: A fast transform solver for PDEs on general geometries", ICML 2024

² Alkin et al., "Universal physics transformers: A framework for efficiently scaling neural operators", NeurIPS 2024

³ Wen et al., "Geometry aware operator transformer as an efficient and accurate neural surrogate for PDEs", NeurIPS 2025

From ViT to Mesh-Based Transformers



Patch-based tokenization, Dosovitskiy et al. ICLR 2021

Vision Transformer (ViT) - Application of Transformers to images

Grid is partitioned to patches downscaling the input

Cost reduced from **quadratic on pixels** to **quadratic on patches**

- Cannot be readily applied to irregular meshes as they do are not regularly structured

Handle mesh irregularity

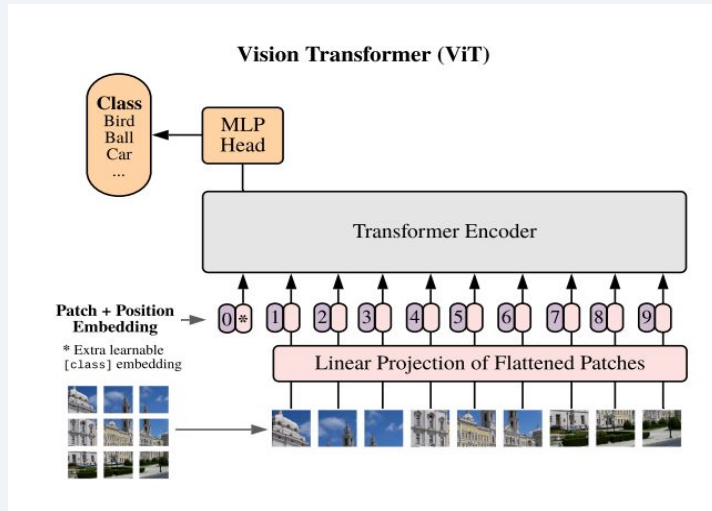
Transolver¹,
UPT², GAOT³

¹ Wu et al., "Transolver: A fast transform solver for PDEs on general geometries", ICML 2024

² Alkin et al., "Universal physics transformers: A framework for efficiently scaling neural operators", NeurIPS 2024

³ Wen et al., "Geometry aware operator transformer as an efficient and accurate neural surrogate for PDEs", NeurIPS 2025

From ViT to Mesh-Based Transformers



Patch-based tokenization, Dosovitskiy et al. ICLR 2021

Vision Transformer (ViT) - Application of Transformers to images

Grid is partitioned to patches downscaling the input

Cost reduced from **quadratic on pixels** to **quadratic on patches**

- Cannot be readily applied to irregular meshes as they are not regularly structured

Handle mesh irregularity

Transolver¹,
UPT², GAOT³

generalization to unseen dynamics

physics faithfulness

fine-tuning capabilities

¹ Wu et al., "Transolver: A fast transform solver for PDEs on general geometries", ICML 2024

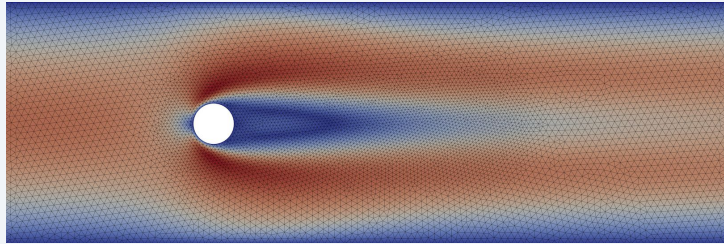
² Alkin et al., "Universal physics transformers: A framework for efficiently scaling neural operators", NeurIPS 2024

³ Wen et al., "Geometry aware operator transformer as an efficient and accurate neural surrogate for PDEs", NeurIPS 2025

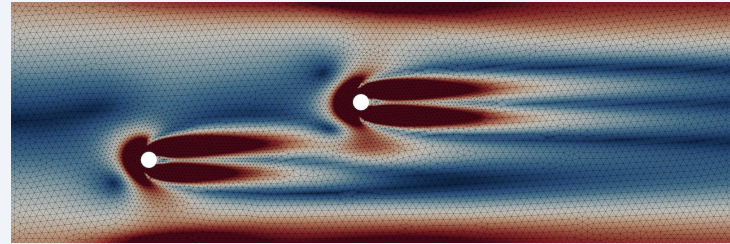
Experimental Setup

Experimental Setup – Overview

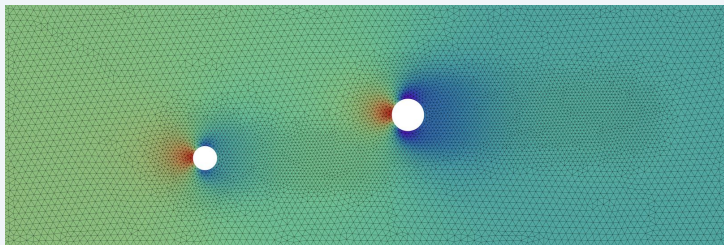
Controlled **benchmark** simulations of incompressible **mean flow fields** around 2D obstacles, using FenicsX¹



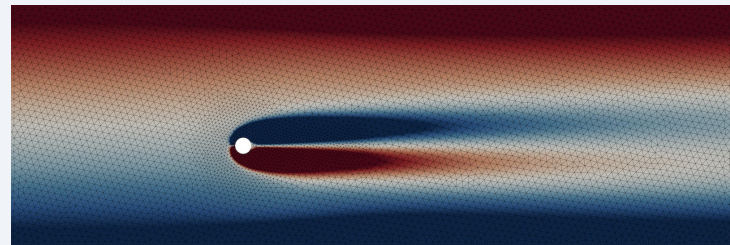
Velocity magnitude



Strain rate



Pressure



Vorticity

Mesh size varies from 6–8k nodes • Reynolds number ranging from 40 to 150

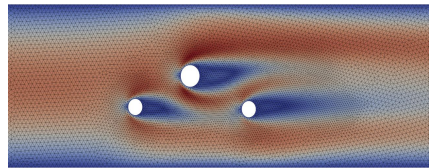
¹ Baratta et al., “DOLFINx: The next generation FEniCS problem solving environment”, preprint 2023

Dataset Generation

Build a dataset rich enough to probe generalization across dynamics, geometry and scale

Dataset Generation

Build a dataset rich enough to probe generalization across dynamics, geometry and scale



Varying cylinder count

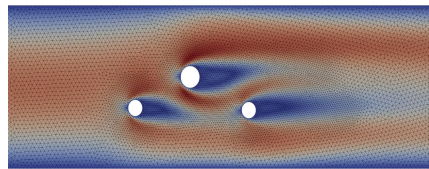
One to four obstacles, testing flow complexity

#150 samples for each cylinder count

600 sims

Dataset Generation

Build a dataset rich enough to probe generalization across dynamics, geometry and scale

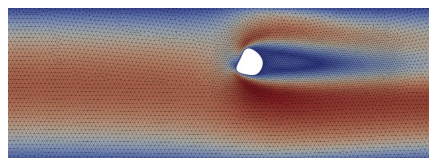


Varying cylinder count

One to four obstacles, testing flow complexity

#150 samples for each cylinder count

600 sims



Irregular shapes

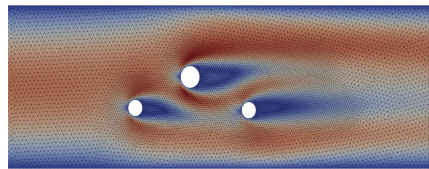
Spline-defined boundaries producing smooth irregular geometries

#30 obstacles for each cylinder count

120 sims

Dataset Generation

Build a dataset rich enough to probe generalization across dynamics, geometry and scale

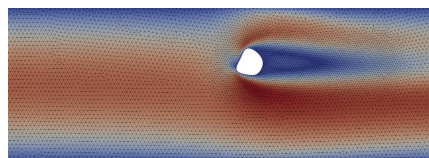


Varying cylinder count

One to four obstacles, testing flow complexity

#150 samples for each cylinder count

600 sims

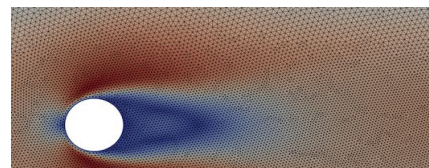


Irregular shapes

Spline-defined boundaries producing smooth irregular geometries

#30 obstacles for each cylinder count

120 sims



Increased resolution

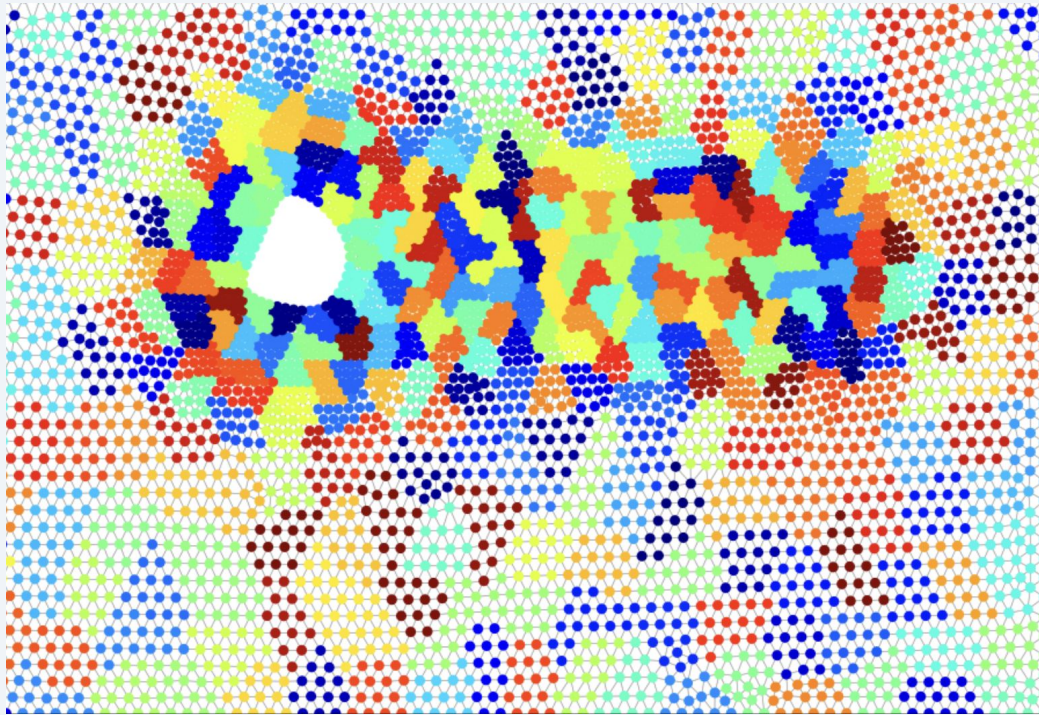
Finer meshes of approximately 30k nodes

Centered obstacles, varying inlet velocity

<10 sims

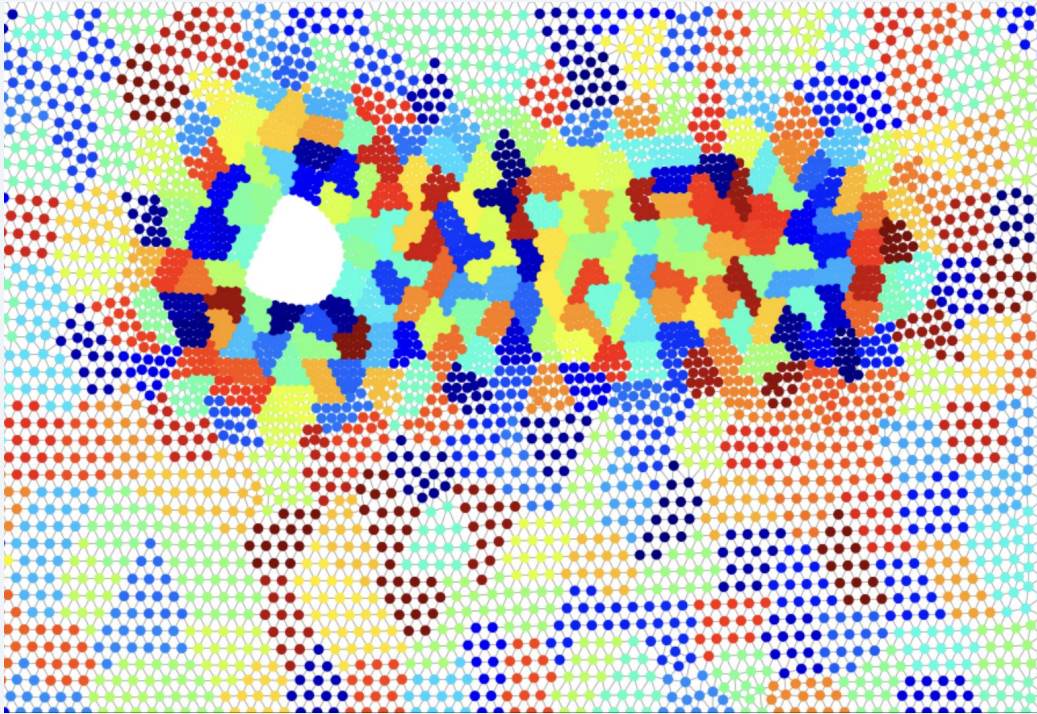
Our Proposed Architecture

Clustering as an alternative to patching



Clustering operation

Clustering as an alternative to patching

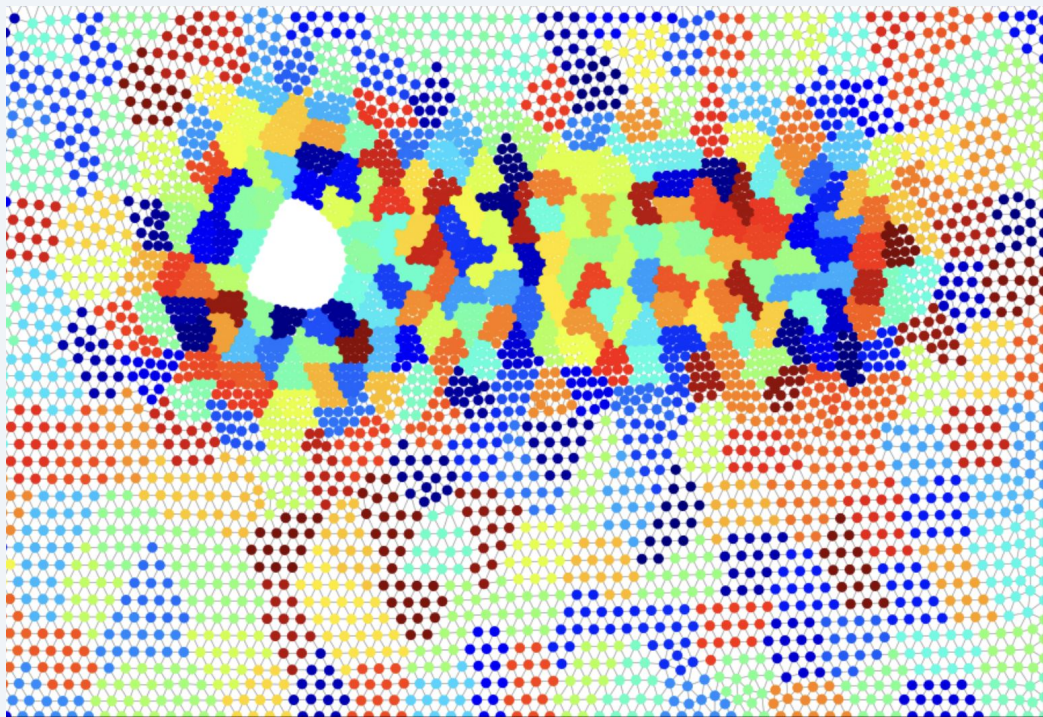


Clustering operation

Fixed cost

Clustering is a one-time offline operation amortized over dataset

Clustering as an alternative to patching



Clustering operation

Fixed cost

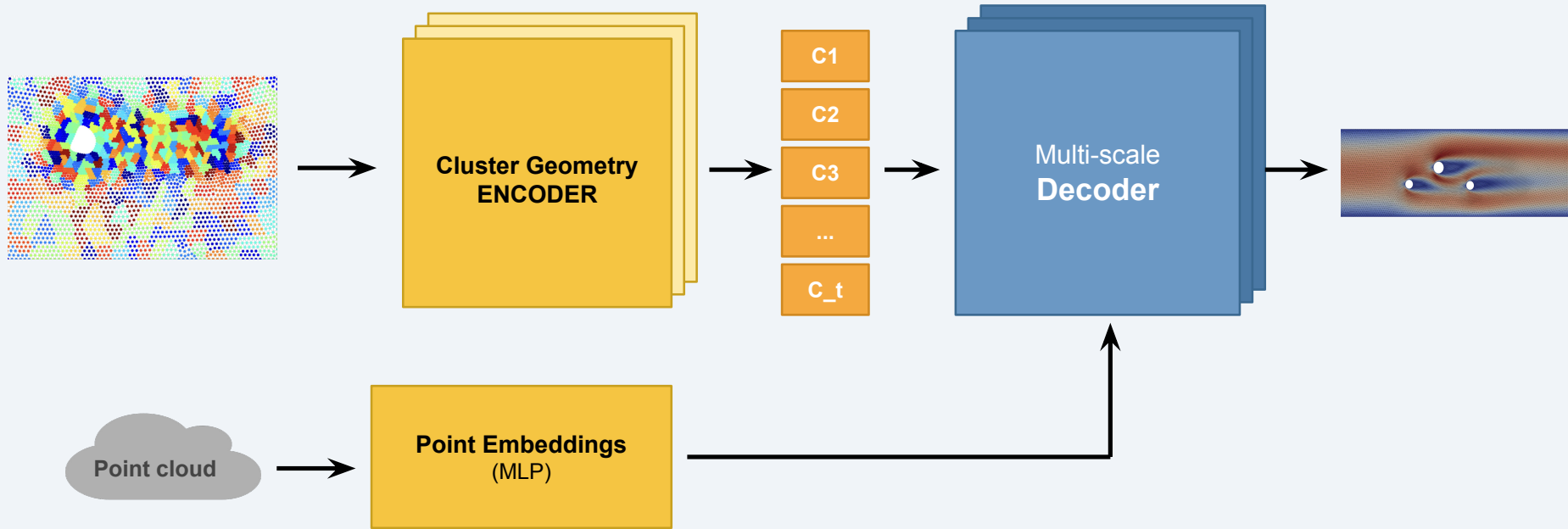
Clustering is a one-time offline operation amortized over dataset

Geometrical patches

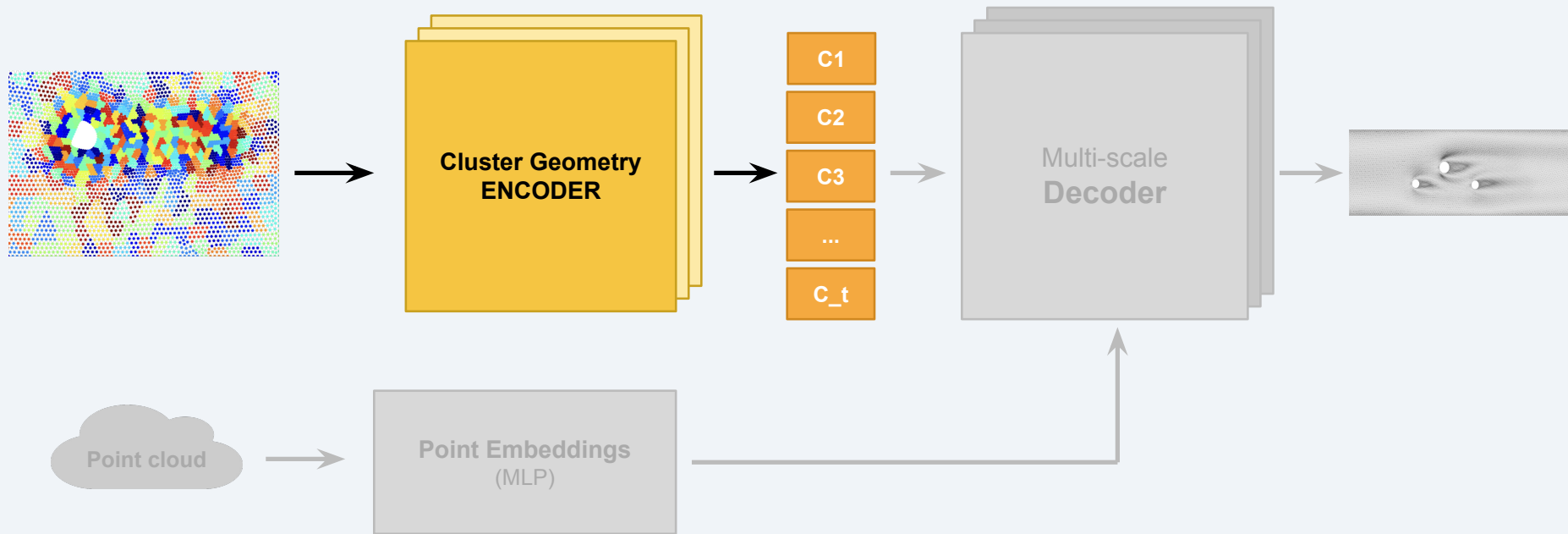
Clusters rely on mesh connectivity

Coarser regions result to larger clusters and vice-versa

Spatial Cluster Transformer – Architecture



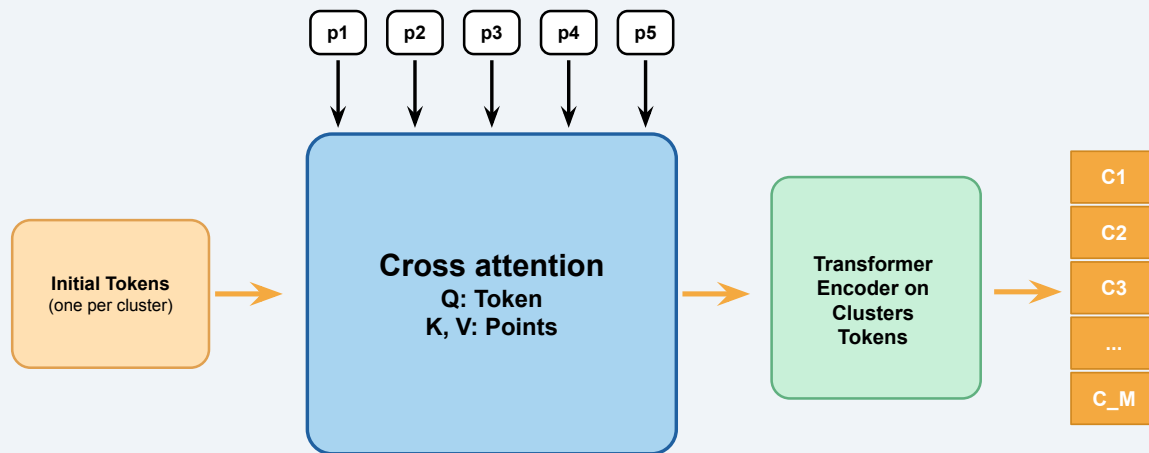
Spatial Cluster Transformer – Architecture



Order-Agnostic Cluster Embeddings

Contribution:

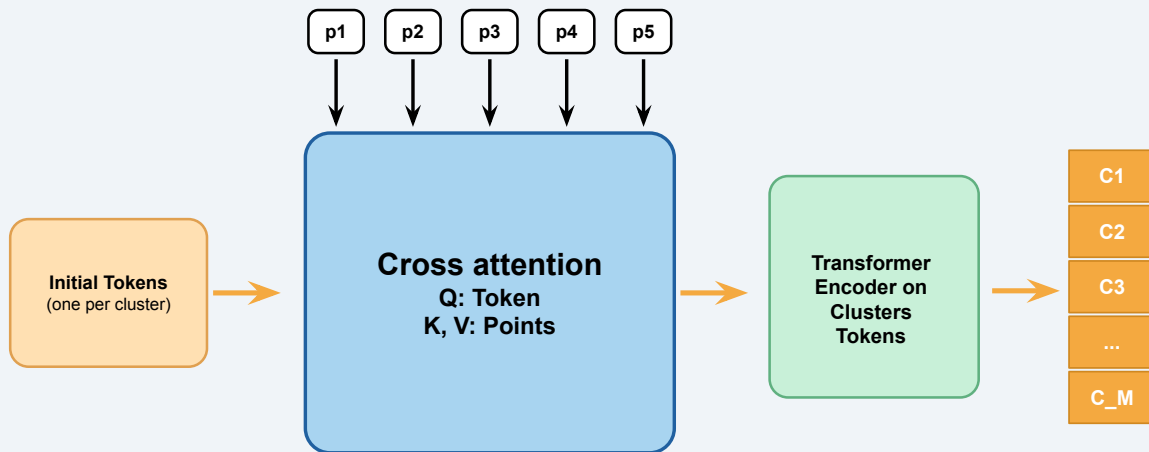
Each cluster is summarized into a single embedding via cross-attention between a learnable token and the cluster's point features.



Order-Agnostic Cluster Embeddings

Contribution:

Each cluster is summarized into a single embedding via cross-attention between a learnable token and the cluster's point features.



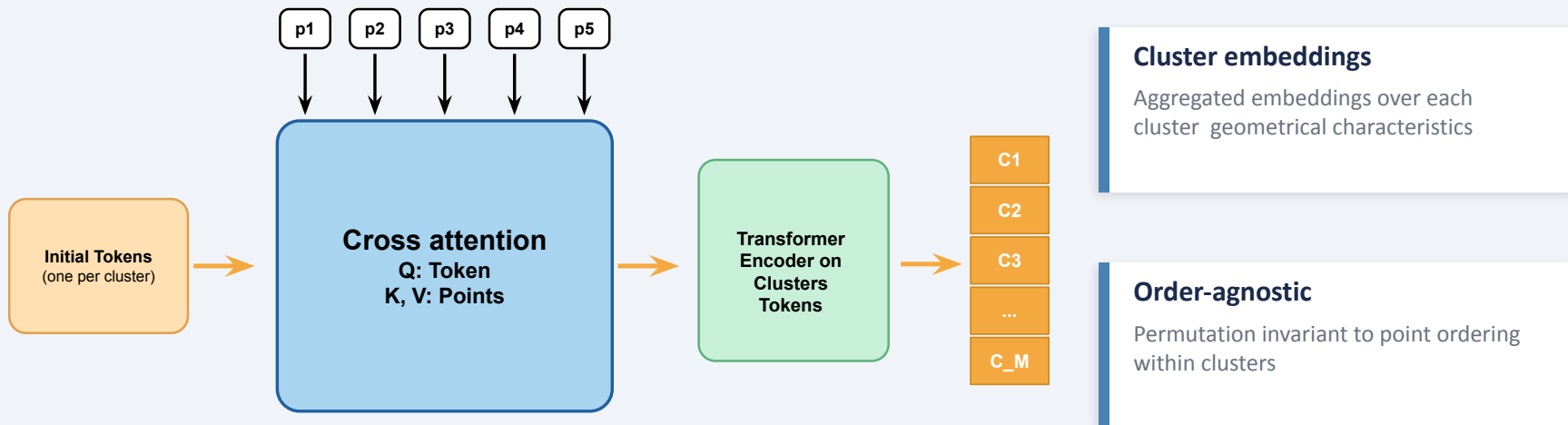
Cluster embeddings

Aggregated embeddings over each cluster geometrical characteristics

Order-Agnostic Cluster Embeddings

Contribution:

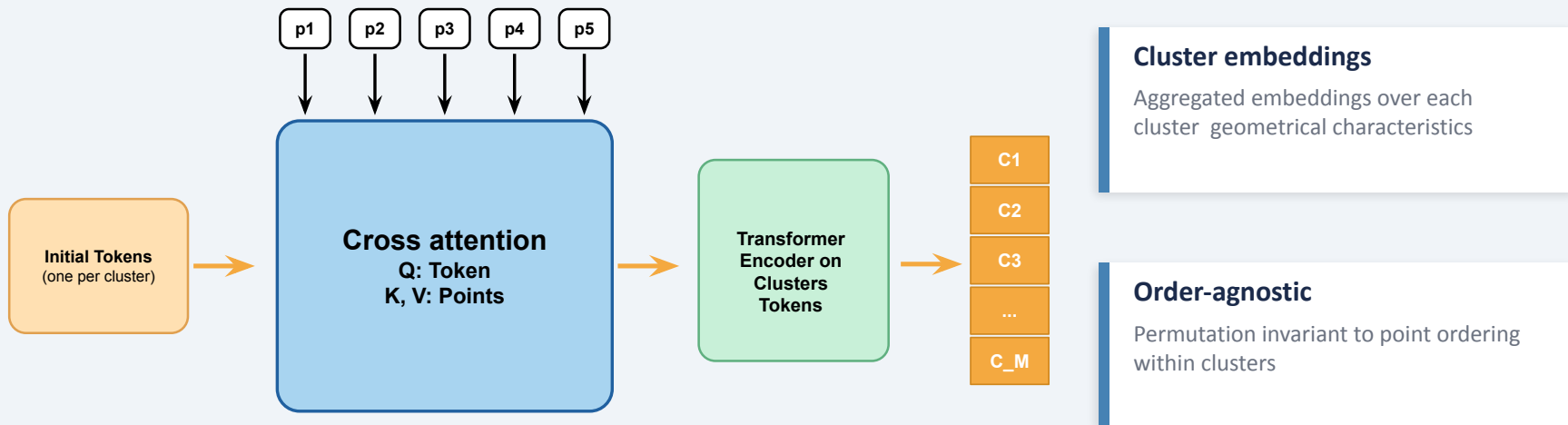
Each cluster is summarized into a single embedding via cross-attention between a learnable token and the cluster's point features.



Order-Agnostic Cluster Embeddings

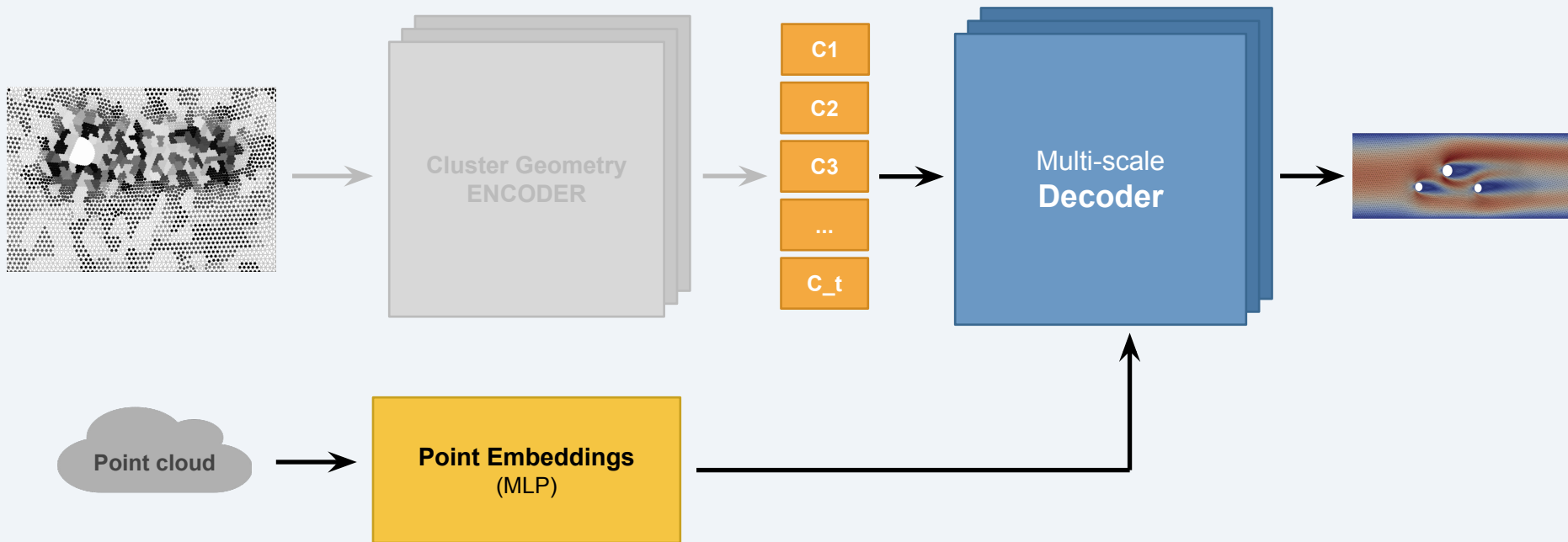
Contribution:

Each cluster is summarized into a single embedding via cross-attention between a learnable token and the cluster's point features.



Reduced complexity Attention cost reduced to $O(M \times n)$ (M = number of clusters) linear to the #nodes

Spatial Cluster Transformer – Architecture



Multi-scale Decoder

Contribution:

Hierarchical combination of point and cloud embeddings • Cross attention scores based on proximity

| | c1 | c2 | c3 |
|----|---------|---------|---------|
| p1 | Score 1 | Score 2 | Score 3 |
| p2 | Score 4 | Score 5 | Score 6 |
| p3 | Score 7 | Score 8 | Score 9 |

Multi-scale Decoder

Contribution:

Hierarchical combination of point and cloud embeddings • Cross attention scores based on proximity

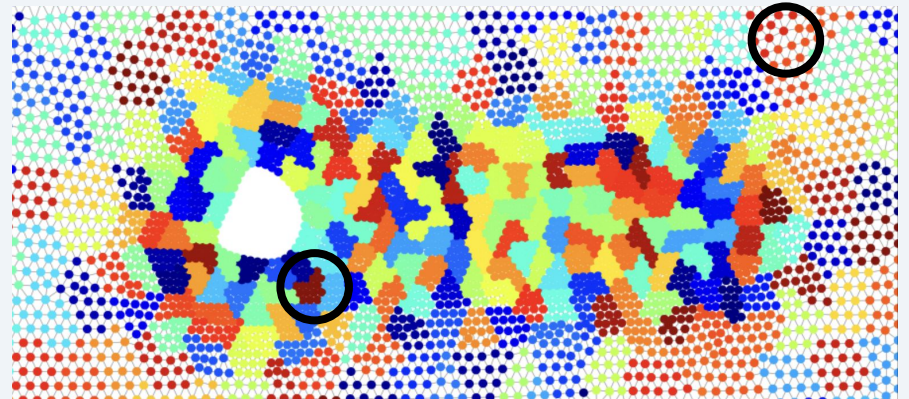
| | c1 | c2 | c3 |
|----|---------|---------|---------|
| p1 | Score 1 | Score 2 | Score 3 |
| p2 | Score 4 | Score 5 | Score 6 |
| p3 | Score 7 | Score 8 | Score 9 |

Spatial attention

Points attend to their nearest clusters

Red scores: spatially relevant pairs

Combination multiple scales of dynamics (point - cluster - neighborhood)



Multi-scale Decoder

Contribution:

Hierarchical combination of point and cloud embeddings • Cross attention scores based on proximity

| | c1 | c2 | c3 |
|----|---------|---------|---------|
| p1 | Score 1 | Score 2 | Score 3 |
| p2 | Score 4 | Score 5 | Score 6 |
| p3 | Score 7 | Score 8 | Score 9 |

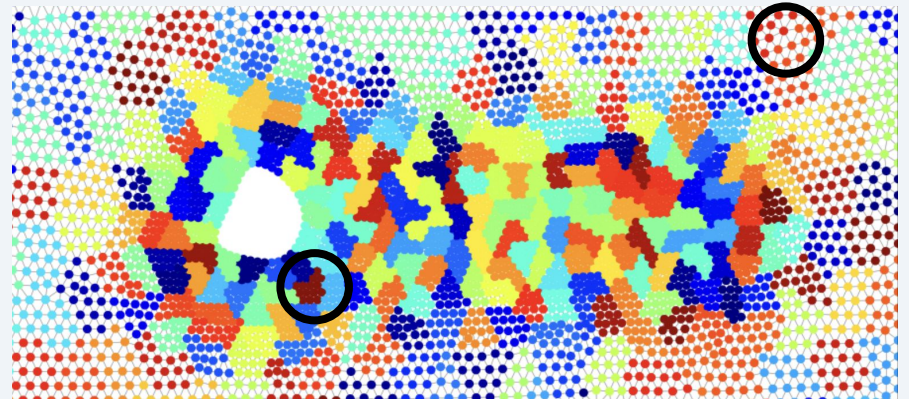
Computation of attention score is able to **focus** - harnessing mesh irregularity

Spatial attention

Points attend to their nearest clusters

Red scores: spatially relevant pairs

Combination multiple scales of dynamics (point - cluster - neighborhood)

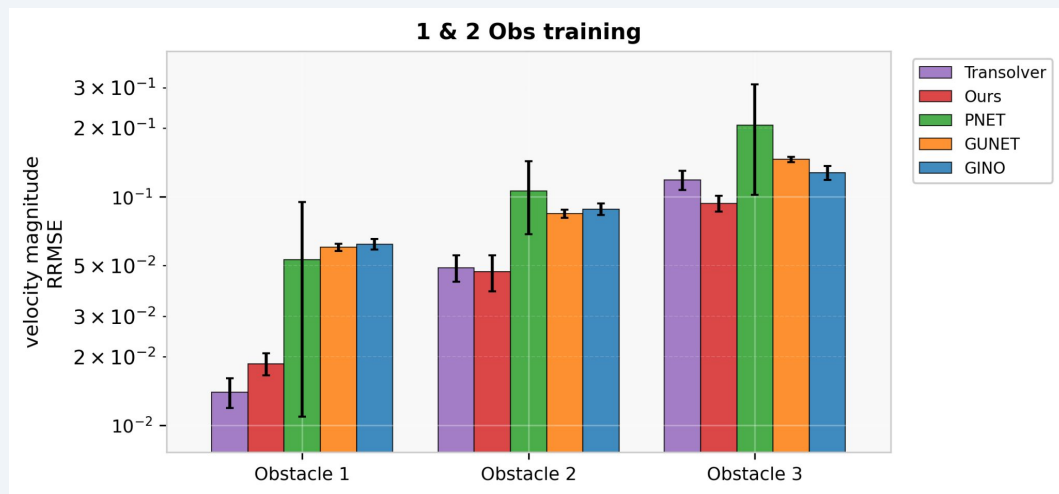


Experimental results

Training configuration details

Architectures evaluated:

Transolver • Ours • PointNet • GraphUNet • GINO



Performance averaged over 5 seeds • 80/20 train/test split

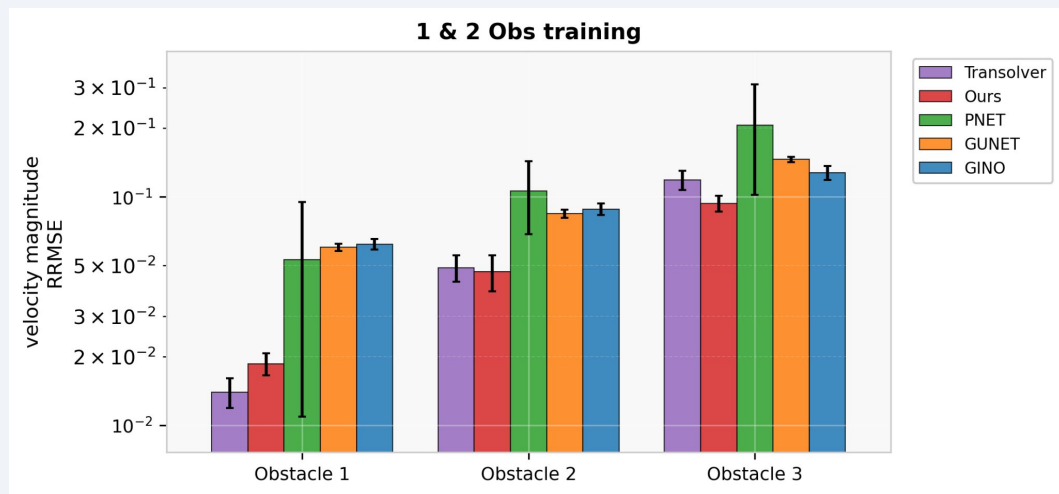
Training configuration details

Architectures evaluated:

Transolver • Ours • PointNet • GraphUNet • GINO

Inputs:

- Positions
- Descriptors
- Features (SDF)

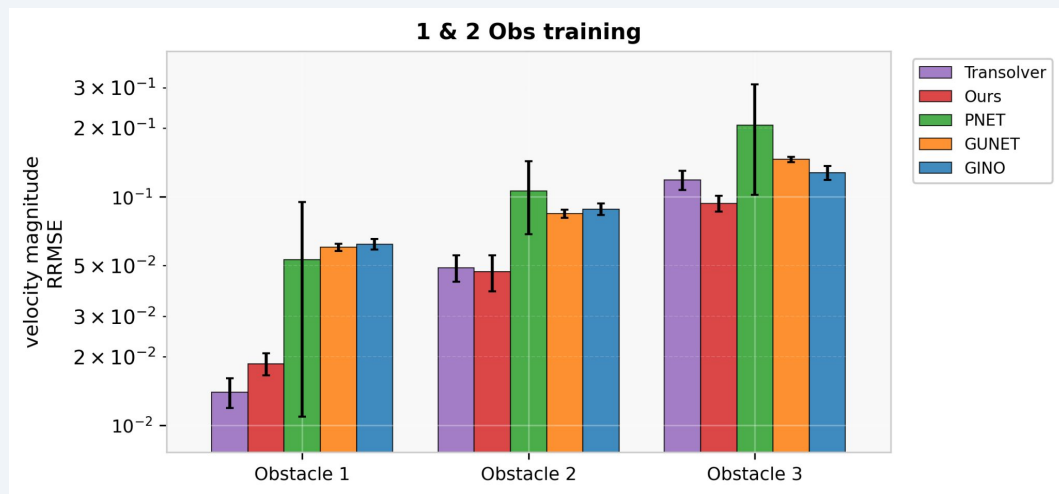


Performance averaged over 5 seeds • 80/20 train/test split

Training configuration details

Architectures evaluated:

Transolver • Ours • PointNet • GraphUNet • GINO



Inputs:

- Positions
- Descriptors
- Features (SDF)

Outputs:

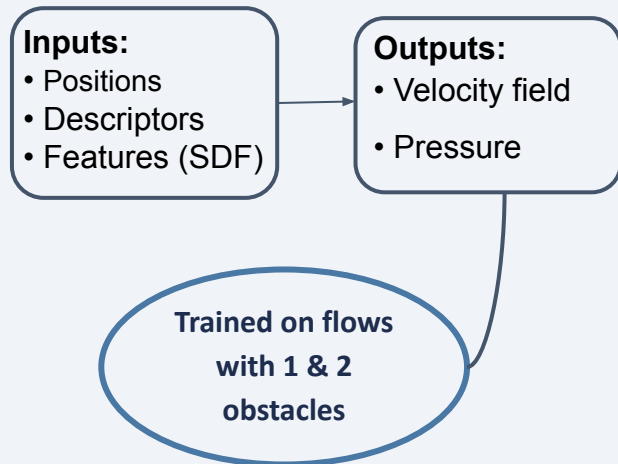
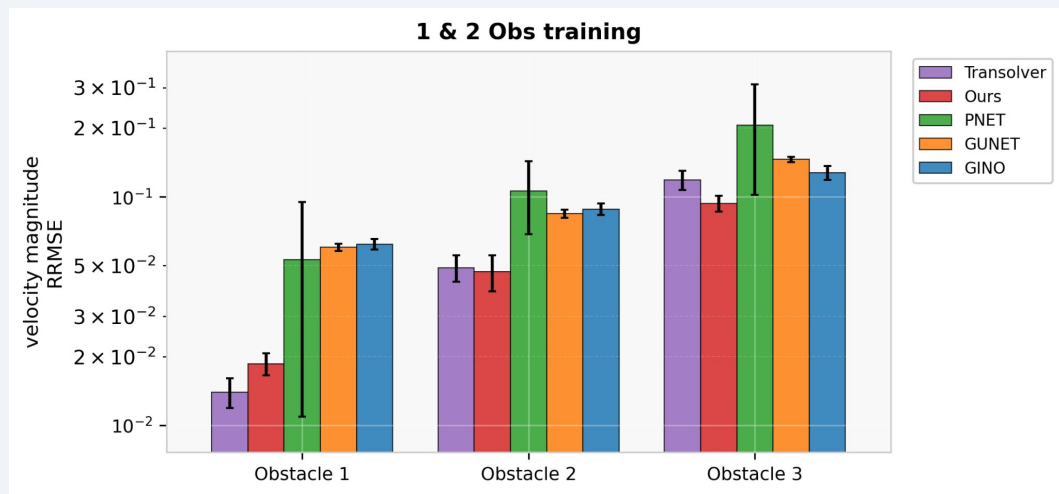
- Velocity field
- Pressure

Performance averaged over 5 seeds • 80/20 train/test split

Training configuration details

Architectures evaluated:

Transolver • Ours • PointNet • GraphUNet • GINO

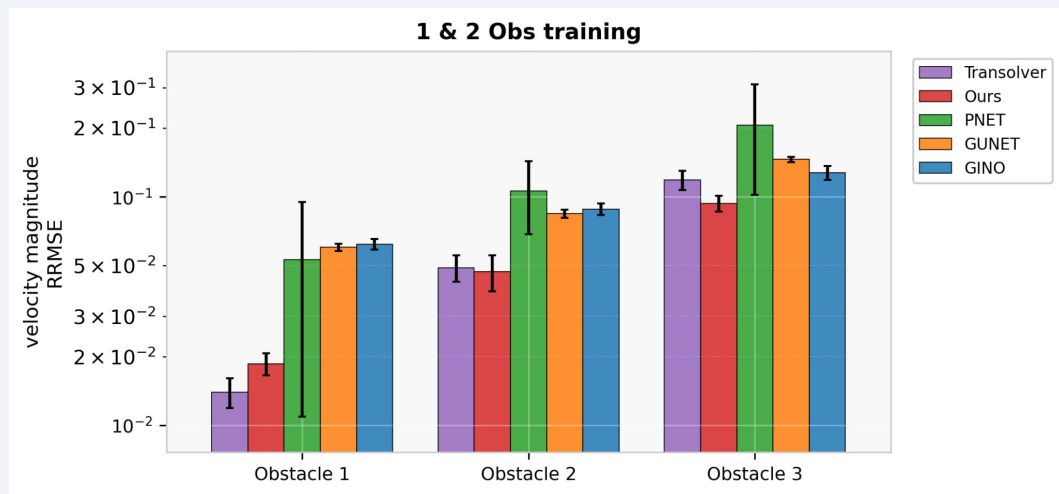


Performance averaged over 5 seeds • 80/20 train/test split

Training configuration details

Architectures evaluated:

Transolver • Ours • PointNet • GraphUNet • GINO



Performance averaged over 5 seeds • 80/20 train/test split

Inputs:

- Positions
- Descriptors
- Features (SDF)

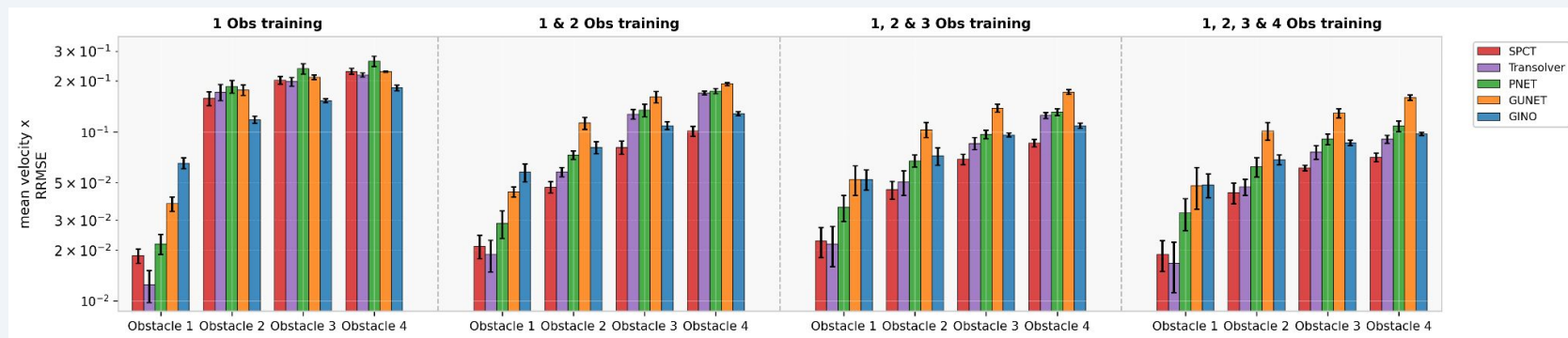
Outputs:

- Velocity field
- Pressure

Trained on flows
with 1 & 2
obstacles

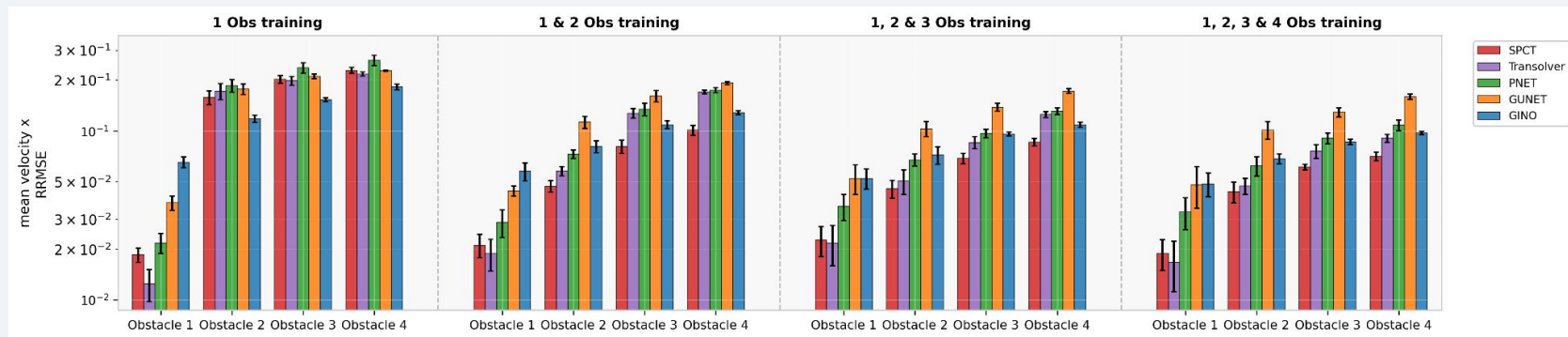
Each group represents evaluation
on specified obstacle count.

Generalization Across Training Configurations



Velocity magnitude **RRMSE** across configurations

Generalization Across Training Configurations

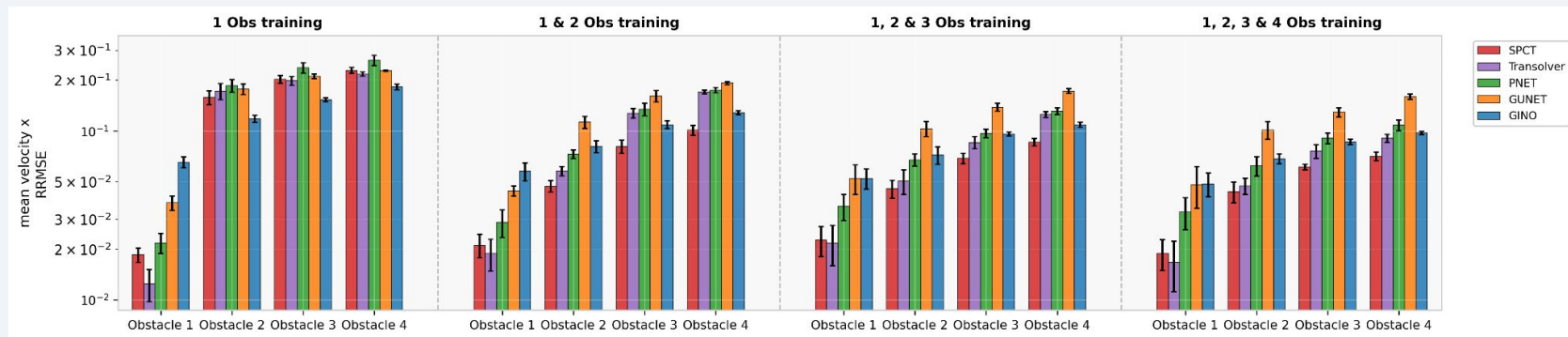


Velocity magnitude **RRMSE** across configurations

Transformers achieve **best performance** on one obstacle inference

Higher obstacle number in the wake signal more complex dynamics not easily resolved

Generalization Across Training Configurations



Velocity magnitude **RRMSE** across configurations

Transformers achieve **best performance** on one obstacle inference

Higher obstacle number in the wake signal more complex dynamics not easily resolved

Spatial Cluster Attention

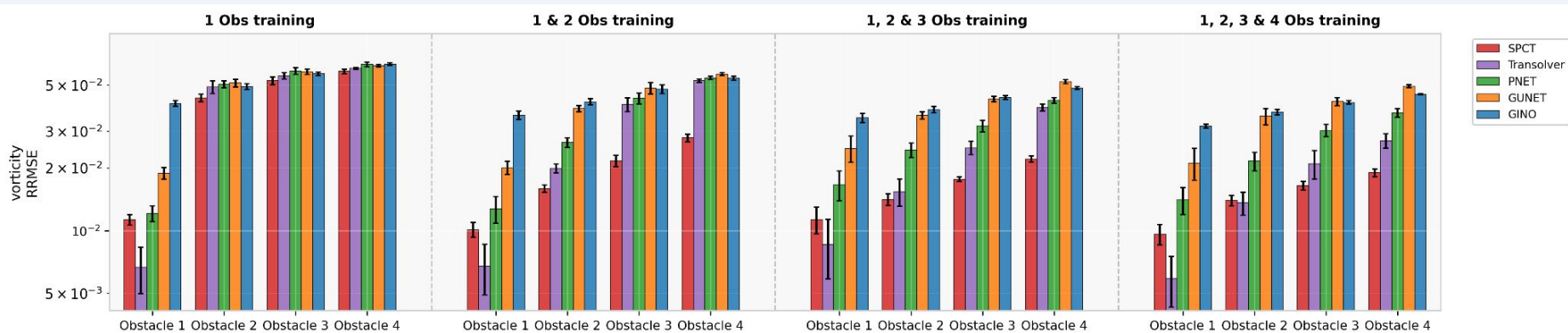
Our architecture achieves comparable performance or even outperforms s.o.t.a. when dynamics get increasingly complex

Reconstruction of derivative quantities

Vorticity $\boldsymbol{\omega} = \nabla \times \mathbf{u}$ computation involves \mathbf{u} gradients — errors reflect the model's ability to capture fine-scale flow details

Reconstruction of derivative quantities

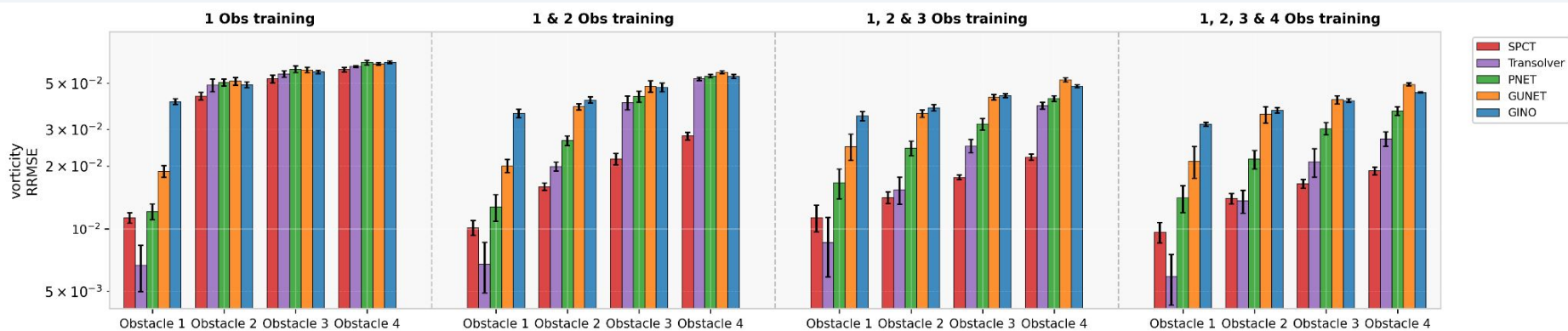
Vorticity $\omega = \nabla \times \mathbf{u}$ computation involves \mathbf{u} gradients — errors reflect the model's ability to capture fine-scale flow details



Vorticity **RRMSE** across configurations

Reconstruction of derivative quantities

Vorticity $\omega = \nabla \times \mathbf{u}$ computation involves \mathbf{u} gradients — errors reflect the model's ability to capture fine-scale flow details

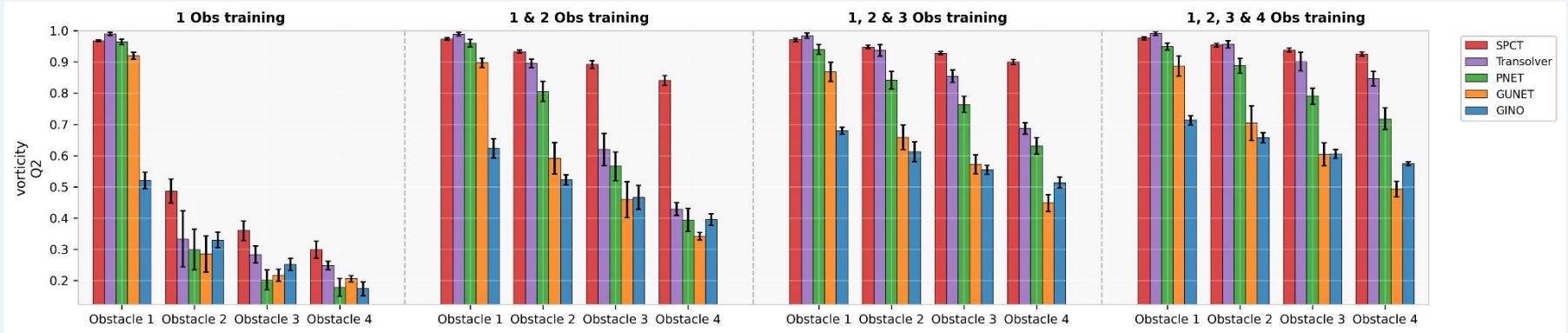


Vorticity **RRMSE** across configurations

Transformers maintain leading performance • Accurate recovery of local velocity gradients

Reconstruction of derivative quantities

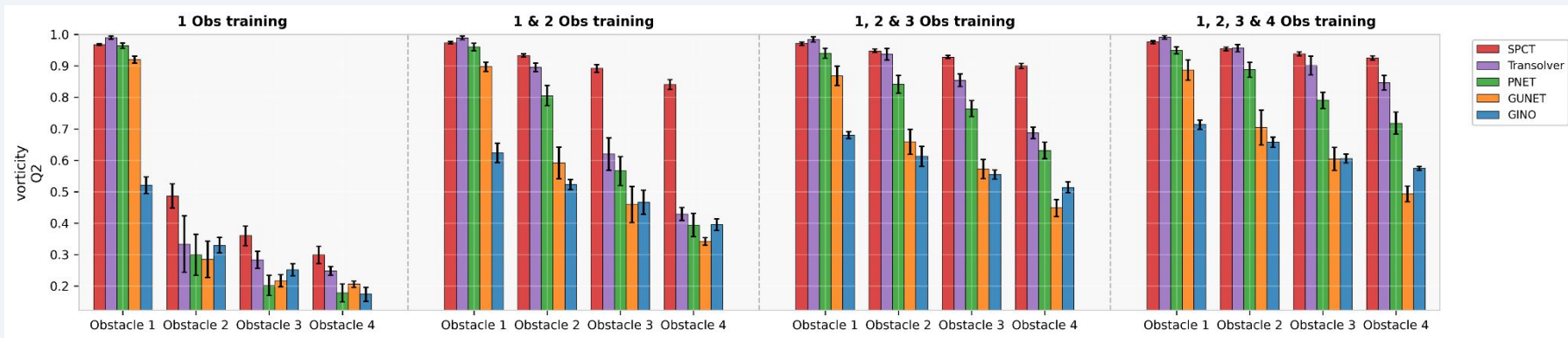
Structure of the vorticity field → spatial patterns and relative magnitudes are captured through Q^2



Coefficient of determination Q^2 across configurations

Reconstruction of derivative quantities

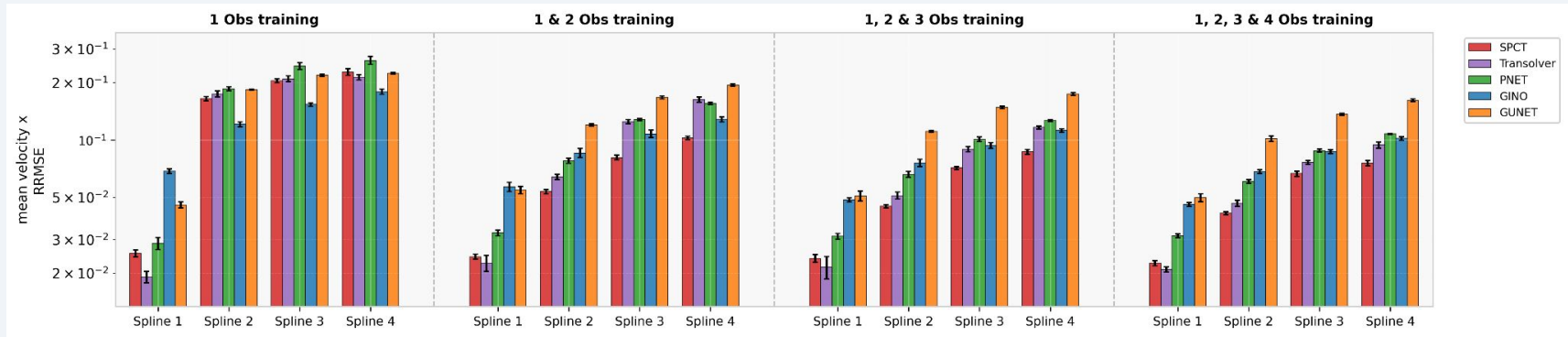
Structure of the vorticity field → spatial patterns and relative magnitudes are captured through Q^2



Coefficient of determination Q^2 across configurations

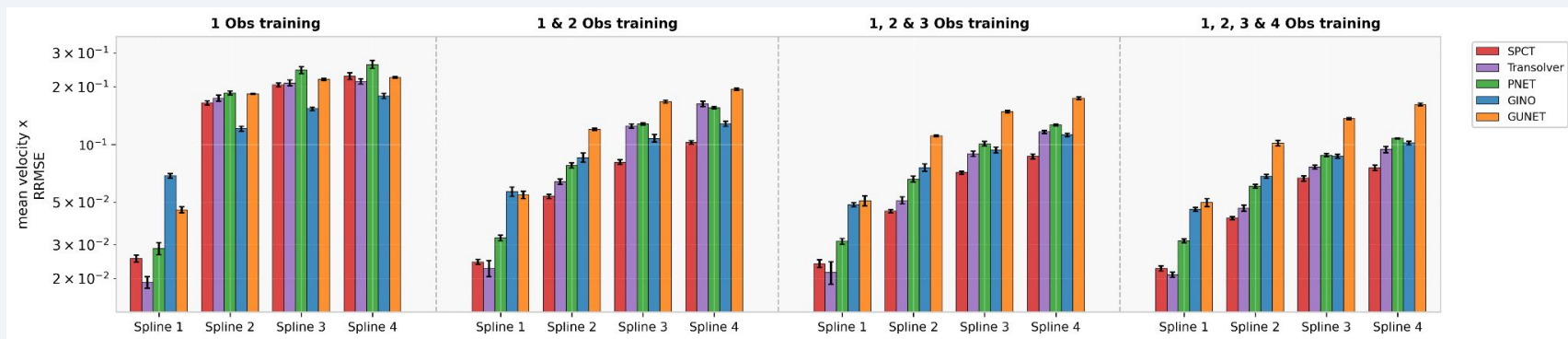
Transformers maintain high Q^2 across all evaluation sets and generalize better beyond their training distribution

Generalization to Unseen Geometries



Velocity magnitude **RRMSE** evaluating irregular obstacles across configurations

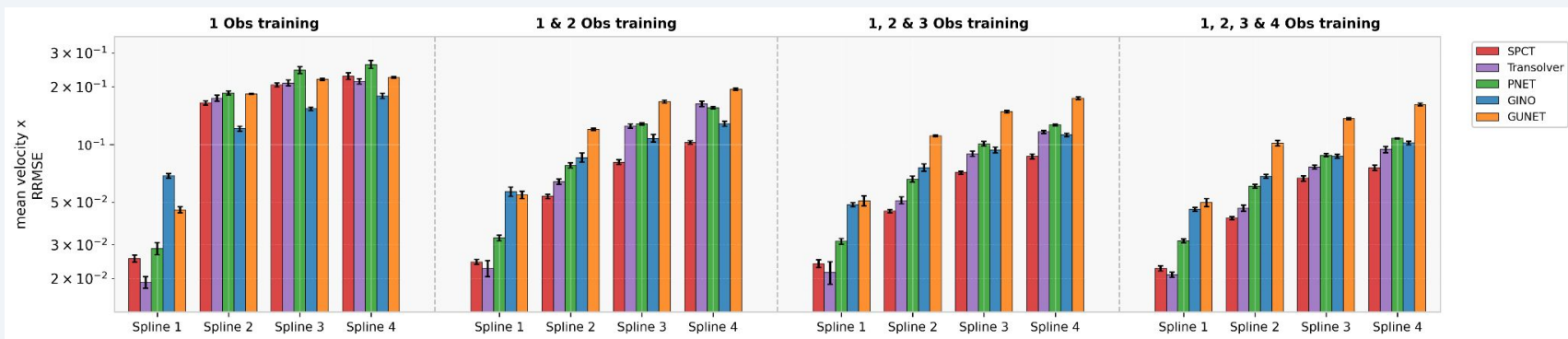
Generalization to Unseen Geometries



Velocity magnitude **RRMSE** evaluating irregular obstacles across configurations

Transformers perform best when obstacle boundaries vary smoothly

Generalization to Unseen Geometries



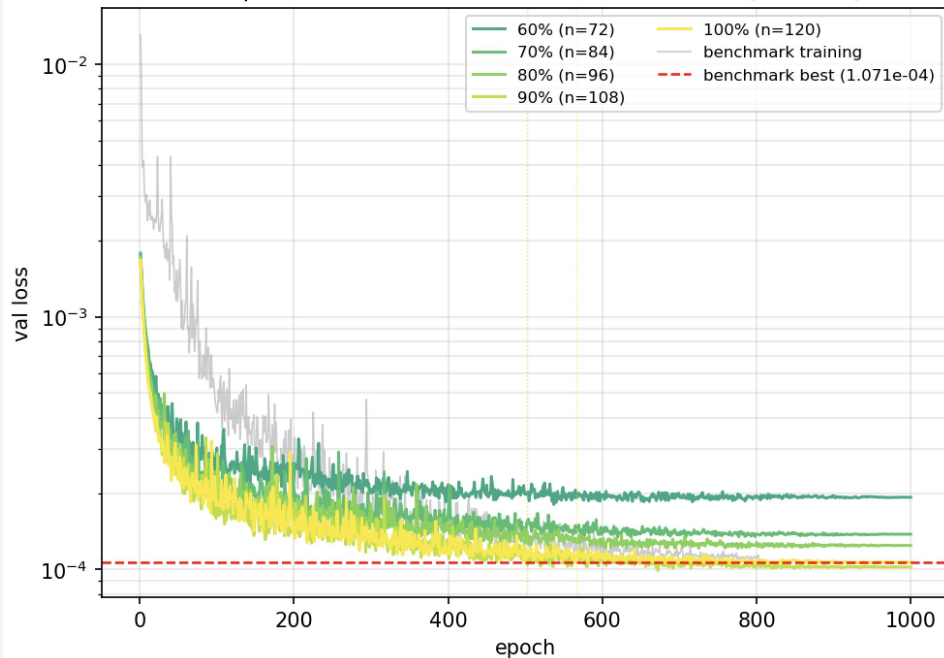
Velocity magnitude **RRMSE** evaluating irregular obstacles across configurations

Transformers perform best when obstacle boundaries vary smoothly

Transformers consistent predictions across groups suggest they fully exploit the available training information

SPCT fine tuning capabilities

SamplePointCloudTransformer obs1 -> obs2 (seed 22)

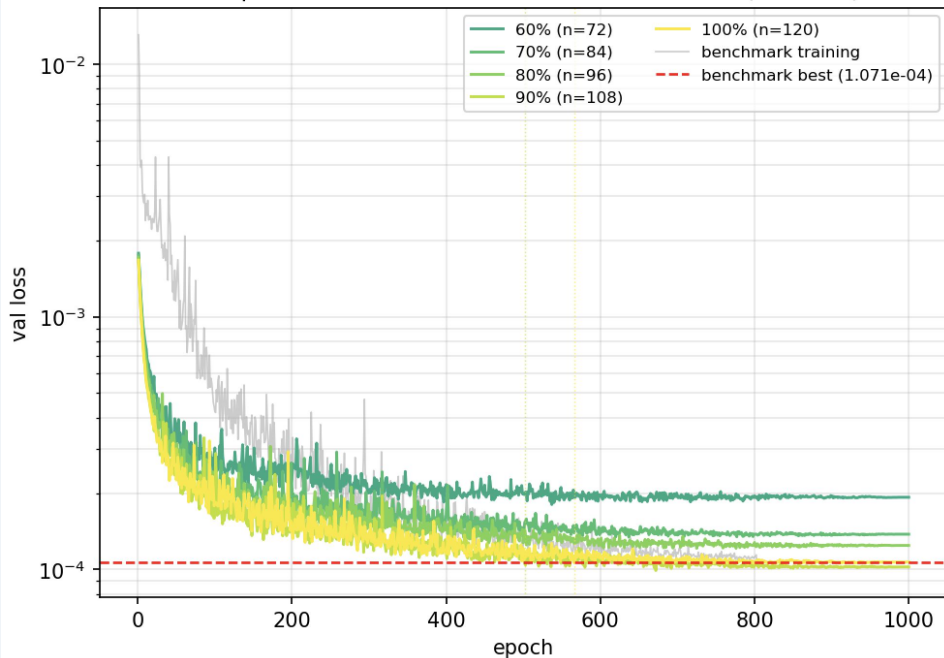


Fine tune performance across different exposure to training distribution data

**Base model
trained on #1
obs flows**

SPCT fine tuning capabilities

SamplePointCloudTransformer obs1 -> obs2 (seed 22)



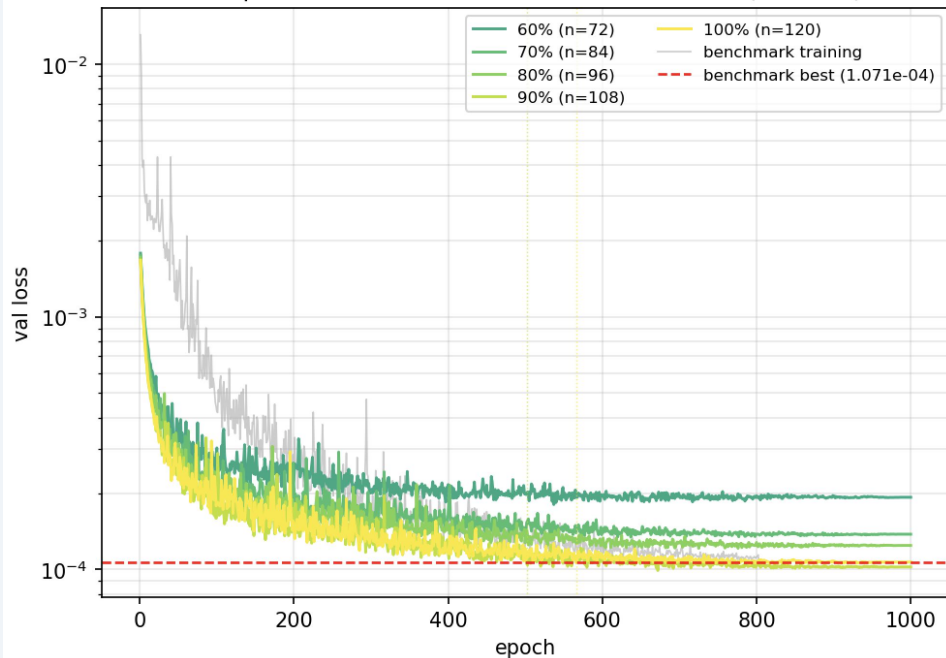
Fine tune performance across different exposure to training distribution data

**Base model
trained on #1
obs flows**

**Target model trained only
on #2 obs flows**

SPCT fine tuning capabilities

SamplePointCloudTransformer obs1 -> obs2 (seed 22)



Fine tune performance across different exposure to training distribution data

Base model trained on #1 obs flows

Target model trained only on #2 obs flows

What percentage of **samples** - training **epochs** required to reach **benchmark** performance

Closing Remarks

Our contributions

- **Novel** Transformer Encoder / Decoder architecture for handling **irregular meshes**

Transformer models:

- Superior **generalization** to complex dynamics of **increased obstacle** counts
- Very accurate **zero-shot inference** for **higher resolution** meshes
- **Leading performance** to out of distribution **obstacle geometries**

Future Work

- Benchmark on **integral quantities**, such as drag & lift coefficients
- **Fine tune** experiments across **obstacle** training configurations
- **Fine tune** to **airfoil** geometries
- Further **architecture refinements** for improved combination of point & cluster embeddings

**Thank you for
your attention!**

Theofanis Ifaistos • Inria – Tau group
(theofanis.ifaistos@inria.fr)

Generalization Capabilities of Transformers in Fluid Dynamics

Theofanis Ifaistos¹,
Matthieu Nastorg², Emmanuel Menier²,
Guillaume Charpiat¹, Marc Schoenauer¹

¹ TAU team, LISN, Universit  Paris-Saclay, CNRS, Inria

² Augur SAS · Inria spinoff

