Session 1

Machine Learning

S1-A1	Jiacheng Sheng — A Neural Network-Based Method for Bird Strike Risk Assessment of an Aircraft Wing
S1-A2	Mingkun Xia — Symbolic Regression for Progressive Discovery of Physical Laws in Fluid Mechanics
S1-A4	Anja Bartelmei — Closed-loop prediction of thermal plume dynamics in Rayleigh–Bénard convection using delay-embedded optical reservoir computing
S1-A5	Ryan Santoso — Machine learning-based framework for predicting effective properties of 3D partially saturated porous materials
S1-A6	Monica Lacatus — Surrogate Quantum Circuit Design for the Lattice Boltzmann Collision Operator
S1-A9	Ioannis Kyritsopoulos — An Improved Data-driven RANS Modelling Approach for Separated Flows
S1-A10	Liu Jun — Rapid Prediction of Total Pressure Field at the Exit of a 3D S-Shaped Expansion Duct Using an Improved U-Net
S1-A14	Wilfried Genuist / Savin Eric — Predictive modeling of incompressible fluid flows using conditional score-based diffusion models
S1-A16	Dev Hathi — Learning Lattice Boltzmann collision operator using Graph Neural Networks
S1-A17	Bharath Ravikumar — Scope of machine learning in the development of many-body dissipative particle dynamics simulations of complex fluids
S1-A18	Jan Rottmayer — Bayesian Neural Network Surrogates for Aerodynamic Shape Design
S1-P3	Partha Dutta — Deep Learning-Enhanced Background Oriented Schlieren for Droplet Evaporation
S1-P7	Ron Barron — Hard Boundary Condition Enforcement for Learning-Based Boundary Value Problem Solvers
S1-P8	Tobias Rentschler — Transformer-Based Quadrilateral Block-Structured Mesh Generation
S1-P11	Ni Shuzhi — Physics-Guided Graph Neural Networks for Joint Optimization of Sensor Placement and Sparse Flow Reconstruction
S1-P12	Yimin Zhang — Explicit Embedding of Flow Directionality and Advection Operators in Neural Architectures for High-Fidelity Fluid Dynamics Prediction
S1-P13	Michael Vipin — Learning viscoelastic constitutive models from mesoscopic simulations using scientific machine learning
S1-P15	Chris R. Jung — Generative Indoor Flow Prediction: Drifting Models for Surrogate Flow Modeling

A Neural Network-Based Method for Bird Strike Risk Assessment of an Aircraft Wing

Jiacheng Sheng[a], Jun Liu[a]

[a] *School of Aeronautics, Northwestern Polytechnical University, Xi'an 710072, China*

Abstract

Bird strike, Bird strike events, characterized by high-kinetic-energy impacts, pose a persistent threat to aircraft structural integrity and operational safety.[1] Traditional reliance on high-fidelity numerical simulations, while valuable, remains computationally intensive and time-consuming, particularly during the iterative preliminary design phase. To address this challenge, this study explores the integration of deep learning techniques, specifically neural networks, into the aircraft structural design and certification workflow.

A reliable finite element model for bird strike simulation was first established and validated against experimental data. Subsequently, a neural network-based surrogate modeling framework was developed. The capability of this approach was initially demonstrated on a simplified impact scenario (sphere-on-plate), where the trained model successfully and rapidly predicted key mechanical field variables. The core contribution lies in the application of this framework to a representative aircraft wing leading-edge structure. By automating the generation of a diverse parametric simulation dataset, a deep fully connected neural network was trained to serve as a fast and accurate predictive tool for bird strike response. This model enables near-instantaneous evaluation of structural performance metrics—such as deformation and stress—under various impact conditions[2].

The proposed method significantly reduces the computational burden associated with conventional simulation-based design cycles. It provides a practical and intelligent tool for engineers to rapidly screen design alternatives, assess airworthiness compliance risks early, and optimize the impact resistance of leading-edge structures. This research contributes a data-driven framework that merges computational mechanics with machine learning, promoting more efficient and automated design processes aimed at enhancing aviation safety.

Keywords: Bird Strike; Wing Leading Edge; Numerical Simulation; Neural Network; Impact Dynamics; Structural Design

References

[1] Heimbs S, "Computational methods for bird strike simulations: A review", Computers & Structures, Vol. 89(23-24), pp. 2093-2112, 2011.

[2] Hasılcı Z, Boğoçlu M E, Dalkılıç A S, et al., "Development of a prediction model using fully connected neural networks in the analysis of composite structures under bird strike" Journal of Mechanical Science and Technology, Vol. 36(2), pp. 709-722, 2022.

Symbolic Regression for Progressive Discovery of Physical Laws in Fluid Mechanics

[1,2,3]Weiwei Zhang*; [1,2,3]Mingkun Xia;

[1]*School of Aeronautics, Northwestern Polytechnical University, Xi'an 710072 ， China*

[2]*International Joint Institute of Intelligent Fluid Mechanics, Northwestern Polytechnical University, Xi'an 710072, China*

[3]*National Key Laboratory of Aircraft Configuration Design, Xi'an 710072, China*

Abstract

Symbolic regression is a powerful tool for knowledge discovery, enabling the extraction of interpretable mathematical expressions directly from data. However, conventional symbolic discovery typically follows an end-to-end, "one-step" paradigm, which often yields lengthy and physically meaningless expressions when applied to complex fluid systems. This limitation fundamentally stems from its deviation from the basic pathway of scientific discovery: physical laws do not exist in a single form but follow a hierarchical and progressive pattern from simplicity to complexity. Motivated by this principle, we propose Chain of Symbolic Regression (CoSR), a novel framework that models the discovery of physical laws as a chain of symbolic knowledge. This knowledge chain is formed by progressively combining multiple knowledge units with clear physical meanings along a specific logic, ultimately enabling the precise discovery of the underlying physical laws from data. CoSR is applied to three problems in fluid mechanics: turbulent Rayleigh–Bénard convection, rough circular pipe flow, and laser–metal interaction, demonstrating its ability to improve classical scaling theories. Finally, CoSR showcases its capability to discover new knowledge in the complex engineering problem of aerodynamic coefficients scaling for different aircraft, highlighting its potential as a powerful AI tool for data-driven fluid mechanics and scientific discovery.

Keywords: Progressive learning, Symbolic regression, Fluid mechanics, Scientific discovery, Data-driven modeling

Closed-loop prediction of thermal plume dynamics in Rayleigh–Bénard convection using delay-embedded optical reservoir computing

[1,2]Anja Bartelmei*; [1]Stefan Sinzinger; [2]Jörg Schumacher

[1] *Optical Engineering, Technische Universität Ilmenau, P.O.Box 100565, D-98684 Ilmenau, Germany*

[2] *Fluid Mechanics, Technische Universität Ilmenau, P.O.Box 100565, D-98684 Ilmenau, Germany*

Abstract

We present an optical implementation of reservoir computing for predicting the formation of thermal plumes in the near-wall region of a three-dimensional Rayleigh-Bénard convection layer. To this end, we analyse horizontal slices of the vertical temperature derivative at the wall – a blueprint of the complex structure formation process. The data are obtained from direct numerical simulations of Rayleigh–Bénard convection with a Rayleigh number $Ra = 10^5$ and a high temporal resolution. Optical computing has many advantages. Besides processing speed, massive parallelism and energy efficiency, it offers the possibility for the implementation of scalable neural networks. One concept that is straightforward to implement is brain-inspired reservoir computing. The backbone of this algorithm is a sparsely connected recurrent neural network, the reservoir, which maps sequential inputs in a high-dimensional data space. In contrast to other methods, the output weights are trained only. Our implementation is based on a free space laser-optical imaging system. The setup builds on the use of an optical input device – the spatial light modulator – which, in combination with two polarizers, ensures a nonlinear processing of the reservoir state vectors. With full HD, the input device has enough pixels for the system to process large amounts of data in parallel without the necessity of a dimensionality reduction. This allows the implementation of delay embedding, which increases the memory of the network and improves the general performance over a broader parameter range. By loading the input and reservoir states from the previous prediction step on the same chip, the closed-loop scenario of reservoir computing, a cascading memory is created, which represents several past time steps in parallel. Trained hyperparameters are the size of the input array and the number of delays. We investigate the scalability of our experimental setup with an increasingly complex learning task.

Keywords: Rayleigh–Bénard convection, Reservoir computing, Optical computing

Machine learning-based framework for predicting effective properties of 3D partially saturated porous materials

[1]Ryan Santoso*; [1]Lara Wegner; [1,2]Yuankai Yang; [1]Jenna Poonoosamy

[1]*Institute of Fusion Energy and Nuclear Waste Management (IFN) - Nuclear Waste Management (IFN-2), Forschungszentrum Jülich GmbH, Germany*

[2]*Now at Advanced Space Propulsion and Energy Laboratory (ASPEL), School of Astronautics, Beihang University, China*

Abstract

Knowledge of effective properties of 3D porous materials is critical for electrochemical applications (e.g., fuel cells, batteries) and subsurface energy and storage applications (e.g., geothermal extraction, nuclear waste disposal). These properties govern the movement of fluids, solutes, and charges through the materials, depending on geometries and phase configuration. Accurate estimation of such properties is critical for optimum electrochemical device and subsurface operations design.

Effective properties are computed via direct numerical simulations on high-resolution porous geometries reconstructed from microtomography experiments, using methods such as the Lattice-Boltzmann or finite volume. However, these simulations are computationally expensive as accurately resolving complex 3D porous geometries requires high-resolution discretizations, resulting in high dimensionality. The cost further increases in multiphase systems, where simulations are performed in two steps: first to determine phase configurations, then to compute effective properties based on these configurations.

We propose a machine learning-based framework to accelerate this two-step computation. In the first step, a saturation-conditioned U-Net predicts phase configurations from microtomography images, with gas saturation incorporated via a Neural Network conditioning in the skip connections. In the second step, effective properties are estimated using K-Means clustering combined with a Taylor approximation, based on physical parameters including Euler characteristic, gas saturation, and porosity. Effective properties are expressed as derivatives with respect to these parameters, enabling efficient and interpretable predictions. The framework accurately estimates effective properties of real experimental data using a small number of training samples, achieving speedup over 1000 times. This approach can reduce labor-intensive experiments and enable efficient analysis of large experimental datasets.

Keywords: digital rock physics, multiphase system, U-Net, K-Means clustering, derivative-based surrogate model, microtomography

Surrogate Quantum Circuit Design for the Lattice Boltzmann Collision Operator

[1]Monica Lăcătuș*; [1]Matthias Möller

[1]*Delft Institute of Applied Mathematics, Delft University of Technology, Mekelweg 4, 2628 CD Delft, The Netherlands*

Abstract

Interest in the development of quantum computing hardware and algorithms has steadily increased over the last decade due to their promise of accelerating certain computational tasks beyond classical capabilities. In fluid flows, the wide range of spatiotemporal scales that emerge as the Reynolds number increases makes direct numerical simulation (DNS) increasingly intractable, restricting DNS to low Reynolds numbers and simple geometries. This has motivated the development of quantum computational fluid dynamics methods, particularly the quantum lattice Boltzmann method (QLBM), which offers exponential memory compression: while classical LBM scales linearly in the number of grid points and lattice velocities, quantum implementations scale polylogarithmically, suggesting that fault-tolerant devices with on the order of 100 logical qubits could, in principle, access DNS-scale regimes for complex geometries. Despite this promise, progress in QLBM has been slow due to the nonlinear and dissipative nature of the collision operator, which violates the linear, unitary, and reversible structure of quantum computation. Existing approaches either rely on linearization techniques such as Carleman linearization [1], which introduce probabilistic circuits and lack convergence guarantees, or neglect nonlinearity entirely [2], preventing recovery of viscous stresses.

In this work, we propose a new strategy rooted in classical surrogate modeling for fluid flows [3]. We design and classically train a surrogate quantum circuit (SQC) to learn the Bhatnagar-Gross-Krook collision operator. Trained within a quantum circuit learning framework [4], the SQC preserves essential physical constraints including mass and momentum conservation, lattice symmetries, and scale equivariance while remaining unitary. The resulting four-qubit circuit provides a compact realization of the collision step compatible with quantum hardware. We validate the model on benchmark flows such as Taylor-Green vortex decay and lid-driven cavity flows, demonstrating accurate reproduction of dissipative dynamics and flow recirculation. This approach establishes a practical collision operator for QLBM and highlights the potential of quantum machine learning for physics-informed operator learning in quantum computational fluid dynamics.

Keywords: surrogate modelling, quantum lattice Boltzmann method, quantum machine learning

[1] C. Sanavio, W. A. Simon, A. Ralli, P. Love, S. Succi, *Carleman–lattice– Boltzmann quantum circuit with matrix access oracles*, **Physics of Fluids** 37 (3) (2025) 037123. doi:10.1063/5.0254588.

[2] L. Budinski, *Quantum algorithm for the advection–diffusion equation simulated with the lattice Boltzmann method*, **Quantum Information Processing** 20 (2021) 57. doi:10.1007/s11128-021-02996-3.

[2] A. Corbetta, A. Gabbana, V. Gyrya, D. Livescu, J. Prins, F. Toschi, *Toward learning lattice Boltzmann collision operators*, **European Physical Journal** E 46 (3) (2023) 10. doi:10.1140/epje/s10189-023-00267-w.

[3] K. Mitarai, M. Negoro, M. Kitagawa, K. Fujii, *Quantum circuit learning*, **Phys. Rev. A** 98 (3) (2018) 032309, published 10 September 2018. doi: 10.1103/PhysRevA.98.032309.

An Improved Data-driven RANS Modelling Approach for Separated Flows

[1]Ioannis Kyritsopoulos*; [1]Michael Mays; [1]Saleh Rezaeiravesh; [1]Alistair Revell;

[1]*School of Engineering, The University of Manchester, Manchester, UK*

Abstract

We present a data-driven RANS turbulence model to address the limitations of eddy viscosity models in predicting flows with separation. The model is based on the SST-CC (Curvature Correction) and SST-RM (Reattachment Modification) models, which use correlations to modify the turbulent production term of the TKE equation, enhancing turbulence in the separating shear layer and promoting flow reattachment. A physics-based data-driven framework is used for the calibration and development of correlations in these turbulence models, and applied to flow separation. The calibration phase leverages a simple algebraic correction function, with coefficients specifically designed for flow separation prediction, avoiding ad-hoc corrections often used in methods like field inversion. Bayesian optimisation (BO) is used for the calibration of the correction function in wall-bounded cases and then generalised using Symbolic Regression (SR). The BO-SR training set includes periodic hills, parametric bumps, and a curved backward-facing step. The SR model was evaluated on the FAITH hill case and the Ahmed body, with promising results for velocity, and turbulent kinetic energy predictions. The developed reattachment modification model significantly improved predictions for flows with large separations and shows good generalisation.

Keywords: Turbulence Modeling, RANS models, Machine Learning, Bayesian Optimisation, Symbolic Regression

Rapid Prediction Methods for the Outlet Flow Field Structure of a Three-Dimensional S-Duct Diffuser

[1]Tianlanqi; [1]Liujun*; [1]Yuanhuacheng;

[1]*College of Energy and Power Engineering, Nanjing University Of Aeronautics And Astronautics, Nanjing, China*

Abstract

Your abstract. Maximum 2000 characters including spaces (word count) **This study addresses the three-dimensional complex geometry of a square-to-round S-duct diffuser and develops two efficient and accurate rapid flow field prediction models, enabling fast prediction of exit flow field structures under multi-parameter coupled conditions involving various inlet Mach numbers (Ma), inlet total pressures (P), exit back pressures (Pb), and S-duct distortion degree design variables (ym).The range of design variables are presented in Table 1.Considering the high computational cost of CFD numerical simulations, this research employs Latin hypercube sampling combined with genetic algorithms to optimize sampling points, designing and selecting 104 representative sample points, where ym is a discrete variable with 8 levels, while Ma, P, and Pb are continuous variables. Numerical simulations are conducted using Fluent to obtain flow field data, which is subsequently preprocessed to establish a database for model training. Model I is a reduced-order prediction model based on Proper Orthogonal Decomposition (POD) and Random Forest. This model adopts a strategy of "dimensionality reduction first, prediction second, and reconstruction third": first, the POD method is employed to retain the first 20 dominant modes of high-dimensional flow field data, extracting the primary energy characteristics of the flow field; then, the Random Forest regression algorithm is utilized to establish the mapping relationship between design variables and POD modal coefficients; finally, rapid flow field reconstruction is achieved through the predicted modal coefficients. This model features fast training speed and strong interpretability. Model II is an end-to-end prediction model based on the U-Net convolutional neural network. Initially, polar coordinate interpolation is employed to map the irregular physical domain onto a regular computational grid. The trained model comprises 5 encoder layers and 4 decoder layers, with skip connections established between corresponding encoder and decoder layers to preserve multi-scale spatial detail information. Distinguishing from the standard U-Net architecture, this network innovatively introduces dual-channel zero tensors as initial feature maps at the input layer, employing Feature-wise Linear Modulation (FiLM) to inject conditional information into feature maps, rather than the conventional image-to-image translation paradigm. Both prediction models can accurately predict flow field information for non-sample operating conditions within the sample space, the prediction results are illustrated in Figs. 1 and 2. Specifically, the prediction error for exit total pressure distribution is less than 1%, the prediction error for Mach number distribution is less than 2.5%, and the prediction error for total temperature distribution is less than 2%. In the current study, the sampling values for ym levels are relatively limited, resulting in**

significant prediction errors in operating regions where ym values are absent. Future research will expand the sampling to increase the number of ym levels and introduce transfer learning strategies to further enhance the global prediction performance and boundary extrapolation capability of the models.

Keywords: Three-Dimensional S-Duct Diffuser;Proper Orthogonal Decomposition; Random Forest; U-Net convolutional neural network; Rapid flow field prediction

Table 1 Range of Design Variables

Variable	Lower Bound	Upper Bound
ym	0.6	1.5
Ma	0.3	0.8
P (Kpa)	40.4	107.8
Pb	1.0	1.08

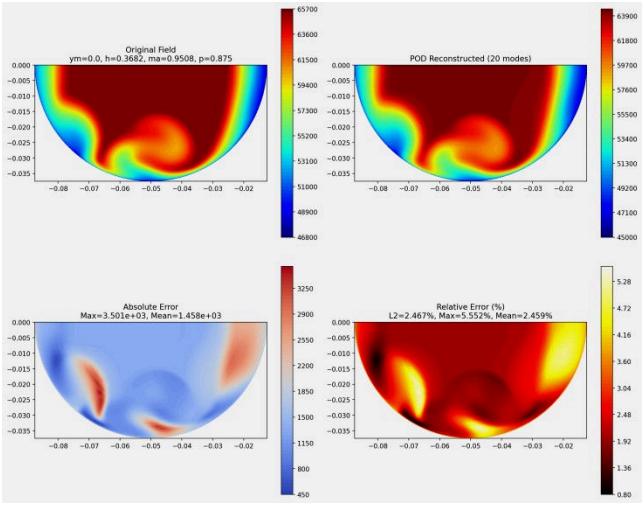


Figure 1. Prediction example of Model 1

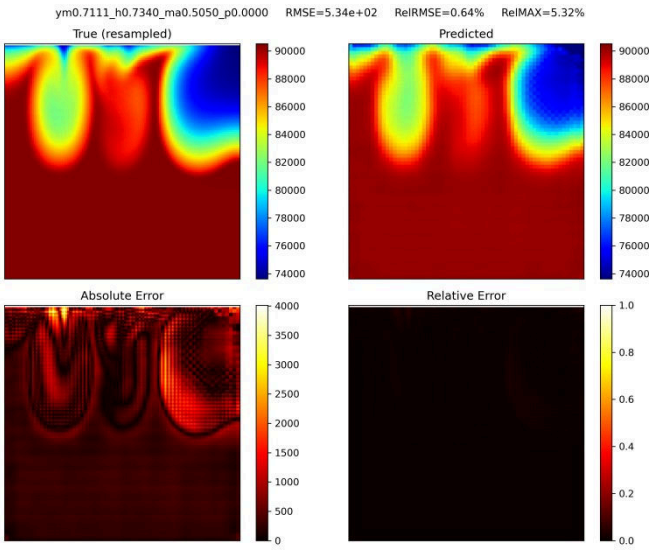


Figure 2. Prediction example of Model 2

Predictive modeling of incompressible fluid flows using conditional score-based diffusion models

Wilfried Genuist,[1,2] **Éric Savin**,[1,*] Filippo Gatti,[2] and Didier Clouteau[2]

[1]Université Paris-Saclay, ONERA
6, chemin de la Vauve aux Granges
91120 Palaiseau, France
{wilfried.genuist,eric.savin}@onera.fr
[*]Corresponding author eric.savin@onera.fr

[2]Université Paris-Saclay, CentraleSupélec
ENS Paris-Saclay, CNRS, LMPS
3, rue Joliot-Curie, 91190 Gif-sur-Yvette, France
{filippo.gatti,didier.clouteau}@centralesupelec.fr

Key Words: *Computational fluid dynamics, Stochastic differential equations, Diffusion models, Score matching*.

ABSTRACT

Well-posedness of the Navier-Stokes equations in both the deterministic and stochastic settings is an open problem [1, 2]. The existence of global weak solutions has been proven in both cases, however uniqueness has not been established so far. Besides, the existence and uniqueness of strong solutions is guaranteed only locally in time. Stochastic Navier-Stokes equations with random forcing have been considered for various reasons. From a mathematical point of view, a conjecture by Kolmogorov suggests that the addition of noise could possibly reduce the number of physically meaningful invariant measures, which are not unique in general. From an engineering point of view, structural vibrations for flow fields in aeronautical applications, or sun heating and industrial pollutions in atmospheric dynamics, for example, can be modeled as random loads. From a computational point of view, this framework paves the way for generative models based on score-matching [3], which learn the logarithmic gradient of the probability density of an underlying diffusion process. By parameterizing this field with a neural network, the statistical geometry of the velocity fields can be reconstructed without explicit knowledge of their probability distribution, allowing the flow statistics to be reconstructed or sampled via inverse diffusion processes. These methods rely on the rise of generative diffusion models in scientific machine learning, a class of models particularly powerful for learning complex distributions [4, 5]. Building on recent advances in generative modeling for computational fluid dynamics, we show in this communication how conditional score-based diffusion models can be adapted to account for physical constraints pertaining to the flow properties, in particular incompressibility or turbulence spectra, and illustrate our approach with numerical simulations [6, 7].

REFERENCES

[1] F. Flandoli. An introduction to 3D stochastic fluid dynamics. In *SPDE in Hydrodynamic: Recent Progress and Prospects (G. Da Prato, M.Rückner, eds.)*, Lecture Notes in Mathematics **1942**, pp. 51-150. Springer, Berlin (2008).

[2] J. C. Robinson. The Navier-Stokes regularity problem. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* **378**(2174), 20190526 (2020).

[3] P. Vincent. A connection between score matching and denoising autoencoders. *Neural Computation* **23**(7), 1661-1674 (2011).

[4] J. Sohl-Dickstein, E. Weiss, N. Maheswaranathan, S. Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. *Proceedings of Machine Learning Research* **37**, 2256–2265 (2015).

[5] Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, Abhishek, S. Ermon, B. Poole. Score-based generative modeling through stochastic differential equations. `arXiv:2011.13456` (2020).

[6] W. Genuist, É. Savin, F. Gatti, D. Clouteau. Autoregressive regularized score-based diffusion models for multi-scenarios fluid flow prediction. `arXiv:2505.24145` (2025).

[7] W. Genuist, É. Savin, F. Gatti, D. Clouteau. Divergence-free diffusion models for incompressible fluid flows. `arXiv:2601.19368` (2026).

Learning Lattice Boltzmann collision operator using Graph Neural Networks

[1]Dev S. Hathi *; [1]Josef M. Winter; [1]Steffen J. Schmidt; [1,2]Nikolaus A. Adams;

[1]*Technical University of Munich, School of Engineering & Design, Department of Engineering Physics and Computation, Chair of Aerodynamics, Boltzmannstr. 15, Garching, 85748, Bavaria, Germany*

[2]*Technical University Munich, Munich Institute of Integrated Materials, Energy and Process Engineering, Lichtenbergstr. 4a, Garching, 85748, Bavaria, Germany*

Abstract

The Lattice Boltzmann Method (LBM) is a computational approach for simulating fluid dynamics, rooted in statistical mechanics. Unlike traditional numerical methods that solve the Navier-Stokes equations directly, LBM models the fluid using a mesoscopic perspective, tracking the evolution of particle distribution functions on a discrete lattice grid. The method has gained significant traction in recent years due to its inherent parallelizability, its straightforward handling of complex geometries, and its ability to recover the macroscopic hydrodynamic equations in the appropriate limit. One of the key components of the LBM is the collision operator, which modifies the distribution functions locally by relaxing them to an equilibrium state. The choice of collision operator critically influences the accuracy, stability, and computational cost of the simulation, and the design of operators capable of robustly handling a wide range of flow regimes remains an open challenge.

Developing collision models that can be used to simulate a broad range of applications is a difficult task, where using neural network surrogates could open new paths for research. Previous works have shown that it is possible to learn simple collision operators using neural networks by embedding physical properties such as conservation laws and lattice symmetries. However, these approaches can only model a part of the collision operator or can only be used to enhance specific operators, limiting their applicability and generalization to more demanding flow configurations. The need for a learned collision model that is both expressive enough to capture the full operator and structured enough to respect the underlying physics motivates the present work.

In this contribution, we present an architecture based on Graph Neural Networks (GNNs) to learn the entropic Multi-Relaxation Time (MRT) collision operator. The entropic MRT relies on a moment representation and an entropic stabilizer to significantly increase the stability range of LBM simulations, making it particularly attractive for high Reynolds number flows where standard Bhatnagar-Gross-Krook (BGK) operators tend to fail. By leveraging the natural locality of the collision step and the symmetry properties of the lattice, the GNN architecture is well-suited to encode the structure of the operator while remaining computationally tractable. The model is parameterized by the relaxation frequency, allowing a single trained network to operate across a range of viscosities without retraining.

We show that the GNN is able to sufficiently learn the full collision operator parameterized by the relaxation frequency and capture the long-term dynamics of benchmark flow problems. For the Taylor-Green vortex decay and the doubly periodic shear layer at high Reynolds numbers, the model accurately reproduces the evolution of kinetic energy and enstrophy over long time horizons. In the lid-driven cavity test case, it correctly predicts the location of primary and secondary vortices, demonstrating robustness in the presence of complex boundary conditions. Finally, for the flow over a circular cylinder, the learned operator yields lift and drag coefficients in close agreement with reference data, correctly capturing the Kármán vortex street and the associated Strouhal number. Together, these results suggest that GNN-based surrogates for entropic MRT collision operators constitute a promising step toward general-purpose, data-driven LBM solvers.

Keywords: Lattice Boltzmann Method, Machine learning, Graph neural networks, Equivariant networks

Scope of machine learning in the development of many-body dissipative particle dynamics simulations of complex fluids

[1]Bharath Ravikumar*; [1]Ioannis K. Karathanassis; [1,2]Manolis Gavaises, [3]Timothy Smith, [3]Gareth Brown

[1]*City St George's, University of London, London, United Kingdom*

[2]*Otto von Guericke University, Magdeburg, Germany*

[3]*Lubrizol Limited, Hazelwood, United Kingdom*

Abstract

Many-body dissipative particle dynamics (MDPD) has emerged as the primary mesoscopic model for capillarity, wetting, and vapor–liquid coexistence because its density-dependent conservative term closes the van der Waals loop that standard pairwise DPD cannot. Despite extensive applications, choosing MDPD parameters that yield target thermophysical properties for real liquids remains non-trivial. The scope of machine learning to model MDPD parameters is still left open as analytical expressions do not enforce thermodynamic and hydrodynamic consistency simultaneously. In this work, we perform a thorough investigation to select the ideal machine learning-based regression model connecting the thermophysical properties such as surface tension, pressure, bonded interactions and Schmidt numbers of the liquids to the MDPD model parameters. The relationship between the different parameters and the target properties established by the regression models are explained using SHAP values. The chosen models are subsequently used to optimise the parameters via evolutionary algorithms (EA) to provide an automated pathway to perform MDPD simulations. The quality of the models is verified by comparing the target properties of four dielectric thermal fluids formulated by mixing polyisobutylene (PIB) polymer in polyalphaolefin oil (PAO). The findings indicate that the Schmidt numbers, bond energies and surface tension values predicted by the ML models and the MDPD simulations with EA optimised parameters align with the ground truth data within an error of one order of magnitude. However, simultaneously matching bulk pressure to the ground truth data while they remain a significant challenge due to the virial form of pressure computation indicating the need for robust characteristic quantities (length, mass and time) while defining the liquid.

Keywords: Polymer solutions | Engineered liquids | Mesoscale model | Machine learning

Bayesian Neural Network Surrogates for Aerodynamic Shape Design

[1]Jan Rottmayer*; [1]Emre Özkaya; [1]Long Chen; [1]Nicolas R. Gauger;

[1]RPTU Kaiserslautern-Landau

Abstract

This paper investigates Bayesian Neural Networks (BNNs) as surrogate models within surrogate-based optimization (SBO) for aerodynamic shape optimization (ASO) problems. SBO is a well-established paradigm for optimizing computationally expensive black-box objectives by iteratively fitting a probabilistic surrogate and selecting new candidates through an acquisition function, which provides an exploration-exploitation tradeoff. Gaussian Process Regression (GPR), or more commonly known as Kriging, remains a standard choice due to the strong inductive bias via kernels and a closed-form posterior, but is limited by the kernel assumptions and poor scaling to higher-dimensional design spaces. In contrast, BNNs offer a more flexible and scalable alternative, offering less restrictive model bias and superior scaling, while retaining uncertainty quantification under a Bayesian treatment of network parameters.

BNNs generalize deterministic neural networks by placing distributions over weights and biases, inducing a predictive distribution via Bayesian inference. Since exact posterior inference is intractable, we consider approximate inference schemes suitable for SBO loops with limited computational budget. We consider Monte Carlo Dropout (MCD) and Infinite-width BNN (IBNN) as variational inference methods and compare against Hamiltonian Monte Carlo (HMC), which draws from the theoretical neural network posterior distribution, and its approximate counterpart Stochastic Gradient Hamiltonian Monte Carlo (SGHMC).

We validate the proposed methodology on two separate ASO problems. For both problems we use the SU2 RANS solver and consider transonic flow conditions. The first ASO considers a NACA0012 profile, decomposed into camberline and thickness distribution, where the camberline is parameterized by 12 Hicks-Henne bump functions. We minimize drag and maximize lift in a multi-objective formulation of the SBO algorithm. Additionally, we include an ONERA wing configuration parameterized using free-form deformation (FFD) boxes to consider an alternative design space structure with increased dimensions (90). The optimization problem considers drag minimization and lift maximization while also providing some geometric thickness constraints.

Results indicate BNN-based surrogates achieve performance comparable to GPR while exhibiting improved scalability and less model bias. Overall, BNNs constitute an alternative modeling choice for CFD-constrained optimization workflows, combining robust function approximation and uncertainty-aware decision making in the SBO framework. This positions BNN-based SBO as promising direction for practical ASO, especially when simulation budgets are tight and reliable uncertainty quantification is essential for guiding the optimization process.

Keywords: bayesian neural networks, bayesian optimization, aerodynamic shape optimization, surrogate-based optimization

Deep Learning-Enhanced Background Oriented Schlieren for Droplet Evaporation

Partha Dutta[1] and Dr. Kiran Raj M.[1]

[1]Department of Applied Mechanics and Biomedical Engineering, Indian Institute of Technology, Madras

Abstract. Background-Oriented Schlieren (BOS) imaging provides a non-intrusive approach for visualizing and quantifying vapor transport during droplet evaporation. In the present study, BOS is applied to investigate post-impact droplet evaporation on a heated substrate, with an emphasis on the quantitative reconstruction of vapor concentration fields. The method is demonstrated for measuring the transient concentration field after the impact of volatile droplets on solid surfaces. Optical displacement fields induced by refractive-index gradients are estimated using both a classical cross-correlation method and a deep-learning-based optical flow model (IRR-PWC Network), employing a physics based irrotational displacement estimation technique trained from a pretrained neural network to obtain a high resolution displacement field. The results demonstrate the potential of combining traditional cross-correlation and physics-based machine-learning displacement estimation techniques to accurately capture the complex and highly transient nature of droplet evaporation, particularly by enabling a comparative assessment of irrotational flow characteristics consistent with BOS.

Keywords: droplet evaporation, vapor concentration field, cross-correlation, optical flow in IRR-PWC Net.

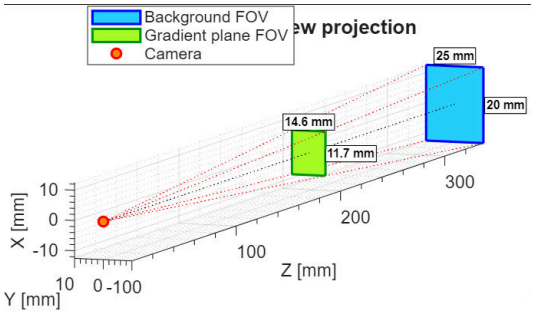
1 Introduction

The droplet evaporation involves coupled fluid dynamics and heat transfer, with applications in inkjet printing and combustion. The interaction of the droplet with the surrounding vapor after impact is critical in many engineering processes. Techniques such as Q-PLIF, PDPA, and He–Ne interferometry provide high-resolution measurements but suffer from limitations, including scattering, alignment sensitivity, and geometric constraints [(1; 2; 3)]. BOS measures vapor through image displacement; while dual-camera setups improve focus [(5)], single-camera systems remain cost-effective. Combined with PIV cross-correlation and physics-informed deep learning, irrotational BOS displacement fields can be estimated with improved spatial resolution [(6)] after the impact of a volatile droplet,enabling accurate reconstruction of refractive-index gradients and quantitative analysis of vapor concentration distribution during droplet evaporation.

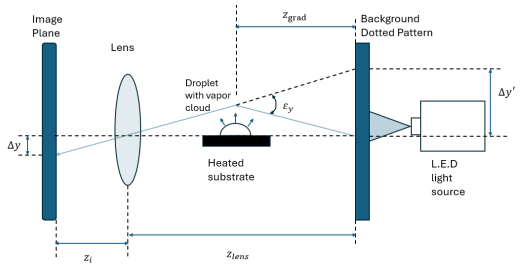
2 Experimental Setup and BOS Method

The experimental configuration utilizes a high-power LED to illuminate a 25×20 mm background pattern (10 μm dots, 50% coverage). Imaging is performed via a FASTCAM Mini UX (1280×1024 px, 2000 fps, 10 μm pitch) equipped with a 70 mm focal length lens (f/3.5–5.6). A 2.5 mm 1:3 (w/w) acetone–water droplet, corresponding to 25 wt.% acetone, is released from a height of 12.5 cm onto a 100°C heated hydrophilic surface. Optimized geometric parameters include $Z_{\text{lens}} = 337$ mm and $Z_{\text{grad}} = 140$ mm, the

distances from the Background FOV to the camera lens and from the Gradient plane FOV to the Background FOV, respectively, as shown in (Fig. 1a) and the two-dimensional schematic (Fig. 1b). Reference and distorted frames were extracted from 2000 fps sequences using ImageJ, cropped to 512×512 px, and spatially aligned to the FOV.



(a) Distance of lens and gradient plane of BOS setup



(b) Geometry of BOS

Figure 1: Experimental setup.

3 Mathematical Formulation

The deflection angles ε_x and ε_y (Fig. 1b) are related to the refractive-index gradients by [(4)]:

$$\varepsilon_x = \frac{1}{n_0}\int_{Z_{\text{grad}}-\Delta Z_{\text{D}}}^{Z_{\text{grad}}+\Delta Z_{\text{D}}} \frac{\partial n}{\partial x}\,\mathrm{d}z = \frac{2}{n_0}\frac{\partial \bar{n}}{\partial x}\Delta Z_{\text{D}}, \quad \varepsilon_y = \frac{1}{n_0}\int_{Z_{\text{grad}}-\Delta Z_{\text{D}}}^{Z_{\text{grad}}+\Delta Z_{\text{D}}} \frac{\partial n}{\partial y}\,\mathrm{d}z = \frac{2}{n_0}\frac{\partial \bar{n}}{\partial y}\Delta Z_{\text{D}} \tag{1}$$

The corresponding schlieren displacements are

$$\Delta x = K\varepsilon_x, \quad \Delta y = K\varepsilon_y \tag{2}$$

where $K = \frac{Z_{\text{grad}}Z_i}{Z_{\text{lens}}}$ is given in equation 3 by [(5)], and where Z_{grad}, Z_i, and Z_{lens} denote the background-to-flow, lens-to-image, and background-to-lens distances, respectively. Combining Eqs. (1) and (2) gives $\frac{\partial \bar{n}}{\partial x} = C\Delta x$ and $\frac{\partial \bar{n}}{\partial y} = C\Delta y$, where $C = \frac{n_0}{2Z_{\text{grad}}\Delta Z_D}$ and n_0 are the refractive index of air, and ΔZ_D is the calibrated half-width of the field of view. The final refractive-index field is obtained by taking the median of the three reconstructed fields as $n_{\text{final}}(x,y) = \text{median}\left(n_{\text{left}}, n_{\text{right}}, n_{\text{top}}\right)$. Boundary conditions set the left, right, and top boundaries to n_{air}, while the bottom boundary remains unconstrained.

4 Displacement Field Estimation

The displacement field is determined by comparing reference and distorted images via PIV cross-correlation and the IRR-PWCNet model for both clean and noisy datasets (Fig. 2). PIV based cross-correlation relies on FFT-based interrogation windows—limiting resolution to low-pass, window-averaged vectors—IRR-PWCNet uses a seven-level feature pyramid and warped cost volumes. This architecture facilitates iterative residual refinement to produce a dense, per-pixel displacement field, resolving high-frequency gradients that are typically suppressed by traditional windowing.

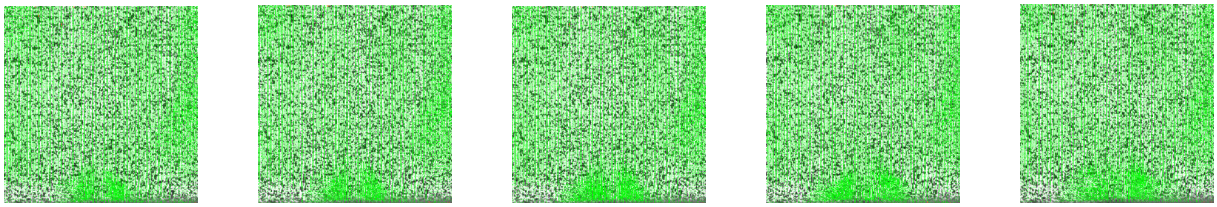


Figure 2: Displacement vector.

4.1 Synthethic data generation for model

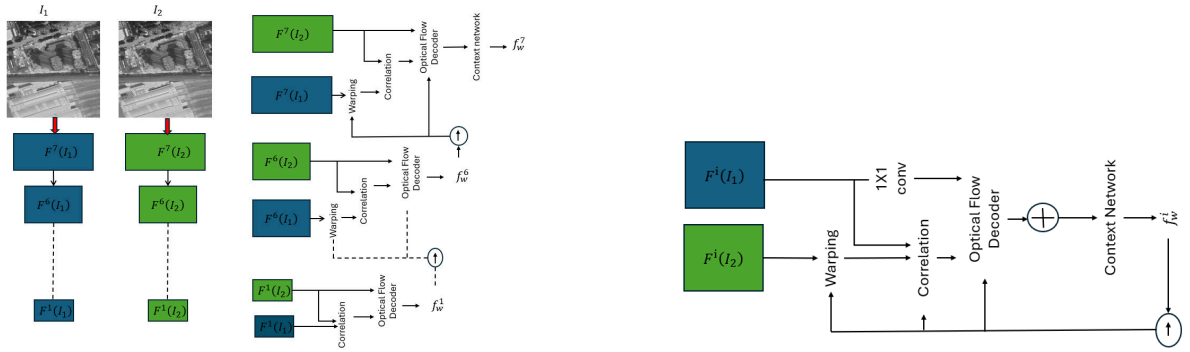
For an irrotational displacement field, the vorticity is

$$\omega_z = \frac{\partial \Delta y}{\partial x} - \frac{\partial \Delta x}{\partial y}. \tag{3}$$

Substituting Eqs. (1)–(2) into Eq. (3) yields $\omega_z = \frac{2K}{n_0} \int \left[\frac{\partial}{\partial x} \left(\frac{\partial n}{\partial y} \right) - \frac{\partial}{\partial y} \left(\frac{\partial n}{\partial x} \right) \right] dz = 0$. Based on this, synthetic data were generated by convolving a random scalar field with a Gaussian filter to obtain a smooth potential given in C. Mucignat[(6)].

4.2 Network architecture used for BOS

The IRR-PWC-Net [(7)], based on PWC-Net [(8)], employs iterative residual refinement with shared weights for efficient and accurate optical flow and occlusion estimation. Unlike PIV cross-correlation, IRR-PWCNet processes images through a seven-level feature pyramid and uses a single shared decoder to iteratively refine flow across levels, resulting in a compact and accurate model (Fig. 3a, Fig. 3b).



(a) PWC-Net **(b) IRR-PWC-Net**

Figure 3: PWC-NET Vs IRR-PWC-Net.

4.3 Loss Function evaluation for irrotational displacement

The model uses satellite images where CMOS-based physics noise [(9)] was added, and transfer learning from 22,872 FlyingChairsOcc planar-motion image pairs was utilized. Following the PIV based cross-correlation-style formulation, only flow weight components are used, while occlusion weight components are excluded. The error at each layer is measured using the average End Point Error (EPE), defined as the L_2 norm shown in Eqs. (4), where M is the number of pixels. Since the displacement field is expected to be irrotational, a curl regularization term is added to the total loss, as shown in Eqs. (5)

$$EPE = \frac{1}{M} \sum_{k=1}^{M} \| f_k^i - (f_{GT}^i)_k \|_2 \tag{4}$$

$$\text{Loss} = \sum_{i=1}^{N} \left(\alpha_i \| f^i - f_{GT}^i \|_2 + \beta_i (\nabla \times f^i) \right) \tag{5}$$

Here, α_i and β_i weight the EPE and curl terms, respectively, at each scale N.

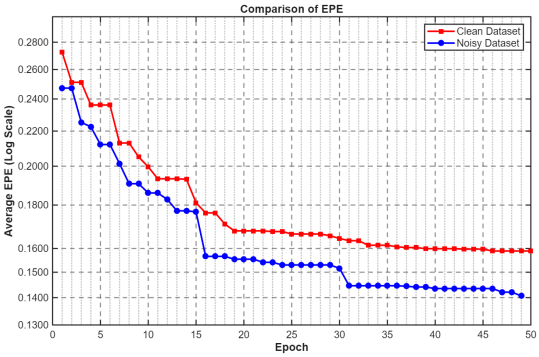
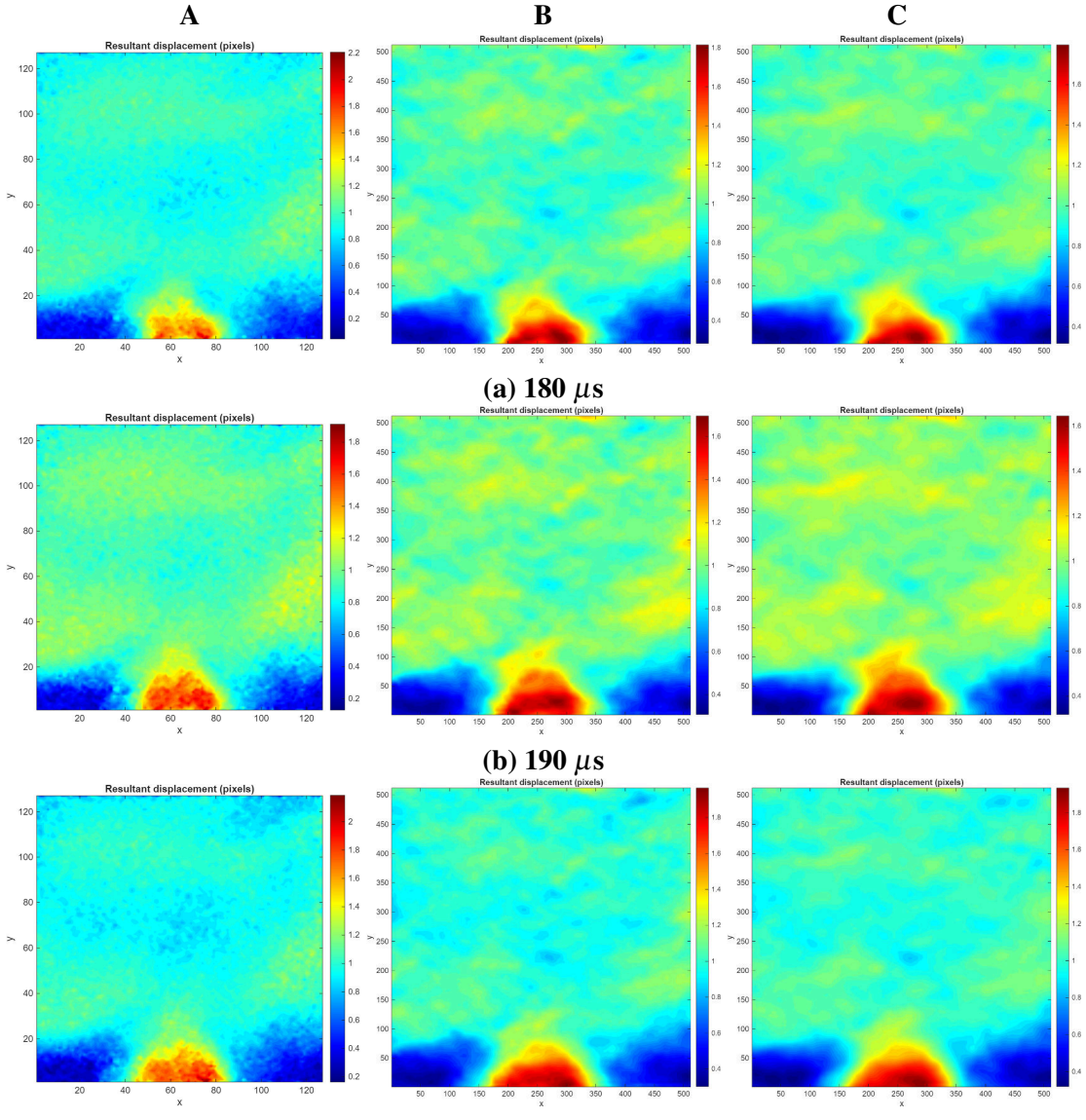


Figure 4: Comparison of End Point Error (EPE) for the clean dataset and noisy data.

5 Results and Discussion

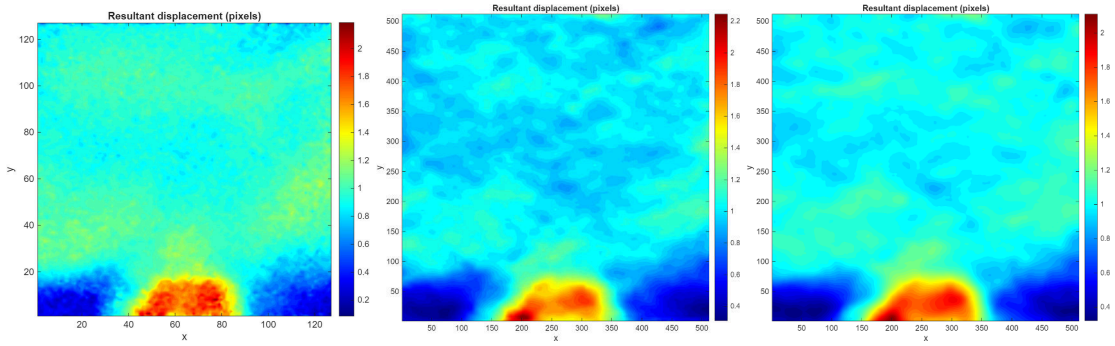
5.1 Comparison of displacement feild



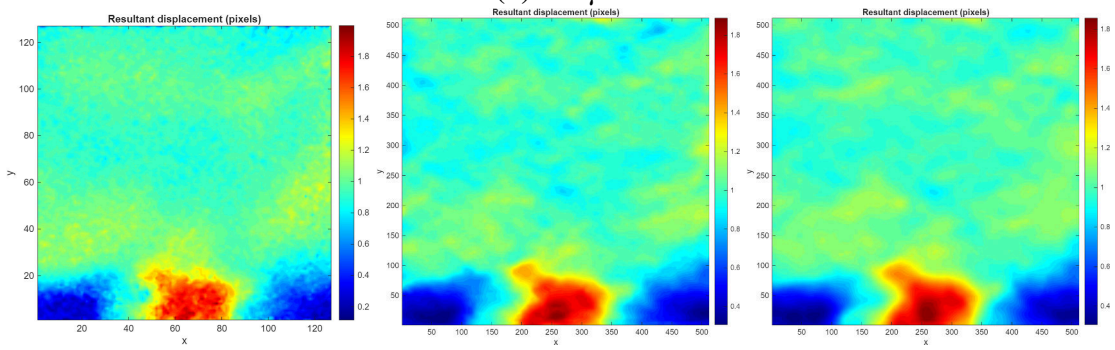
(a) 180 μs

(b) 190 μs

(c) 200 μs



(d) 210 μs



(e) 220 μs

Figure 5: Comparison of the resultant displacement fields: (A) PIV cross-correlation, (B) clean-dataset IRR-PWC-Net, and (C) noisy-dataset IRR-PWC-Net.

PIV cross-correlation captures droplet evaporation structure, but for a 512 × 512 image, it resolves only 128 vectors (512/4) due to interrogation windows (32 × 32 to 4 × 4), averaging motion within each window and blurring fine features such as the thin vapor layer, especially in noisy conditions. In contrast, IRR-PWCNet optical flow uses a seven-level image pyramid, starting from coarse resolution to capture large displacements and iteratively refining through upsampling to Level 1, producing a dense 512 × 512 flow field that preserves both large and fine-scale structures after training with rotationality constraints on clean and noisy datasets

5.2 Comparison of rotationality

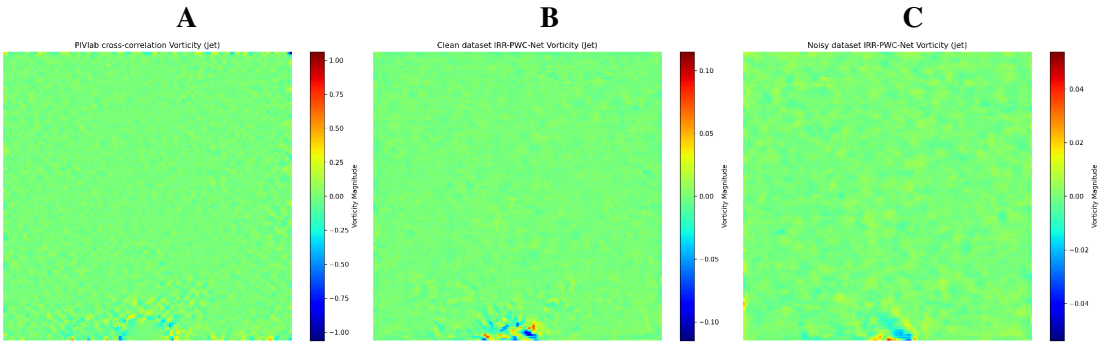


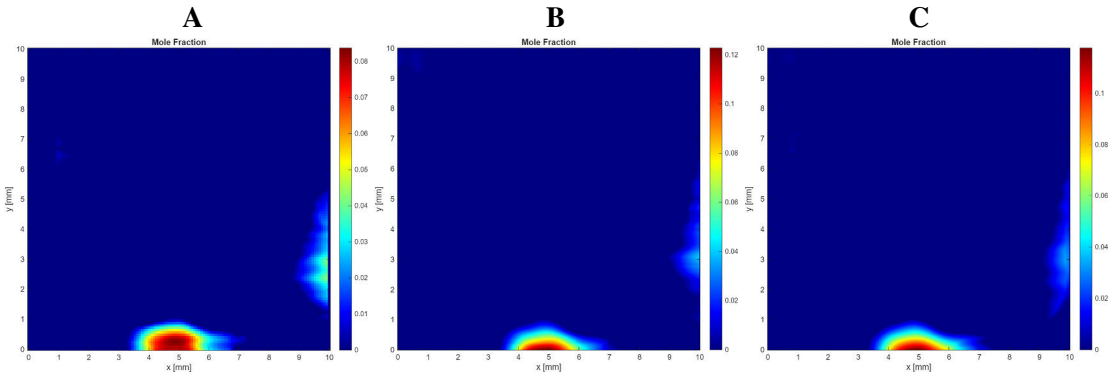
Figure 6: Comparison of rotationality: (A) PIV cross-correlation, (B) clean-dataset IRR-PWC-Net, and (C) noisy-dataset IRR-PWC-Net.

5.3 Comparison of molefraction

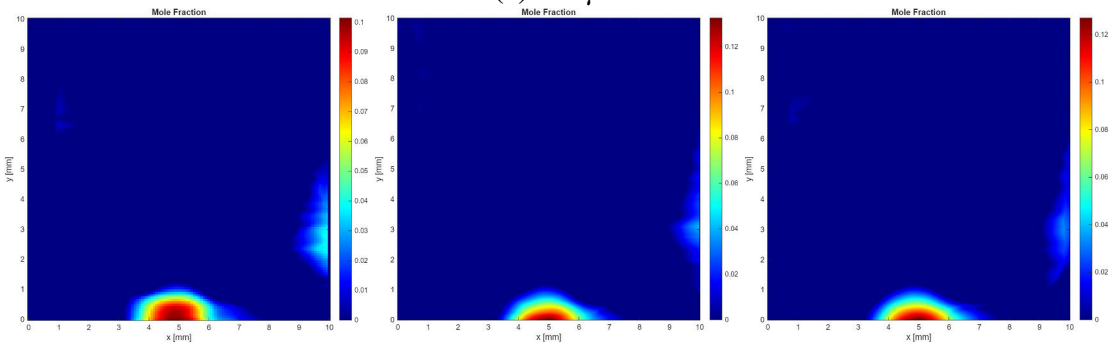
The displacement field obtained in PIV cross-correlation,clean dataset, and noisy dataset can be used to find the refractive index of the mixture .The refractive index of the mixture obtained can be used to

find the concentration of droplet evaporation using the mole fraction derived from the Gladstone-Dale equation $n_{\text{mixture}} - 1 = K\rho$ given in equation (6), and their comparison is shown in (Fig. 7).

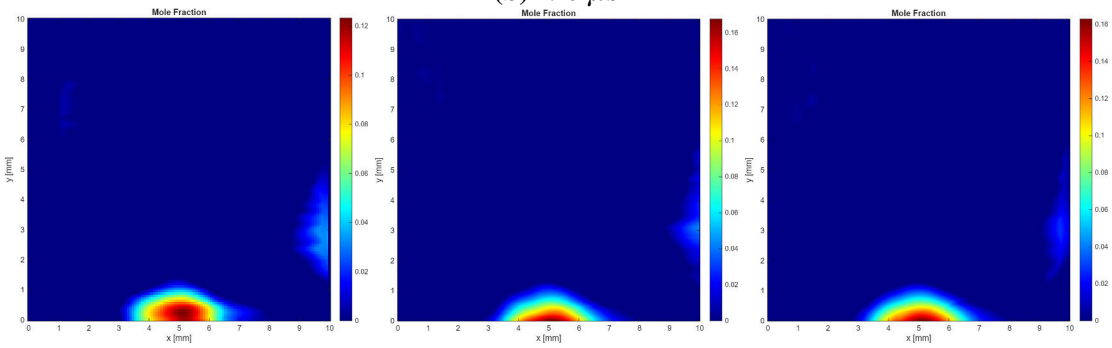
$$\chi = \frac{\Delta n}{n_{\text{vapor}} - n_{\text{air}}}, \tag{6}$$



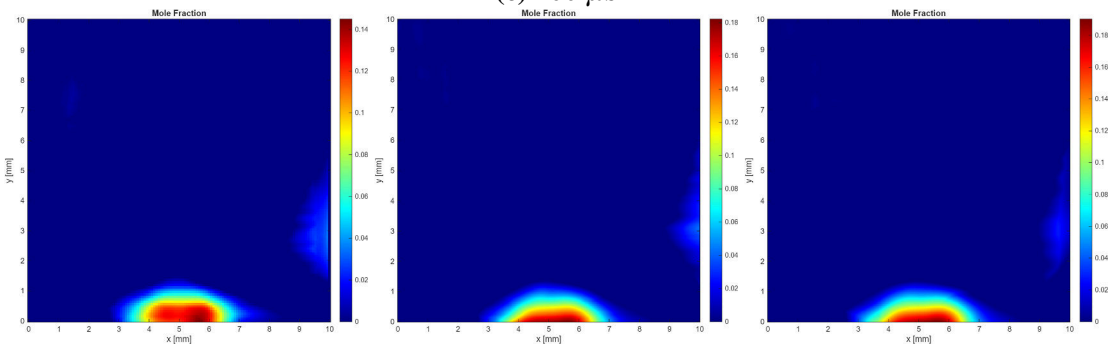
(a) 180 μs



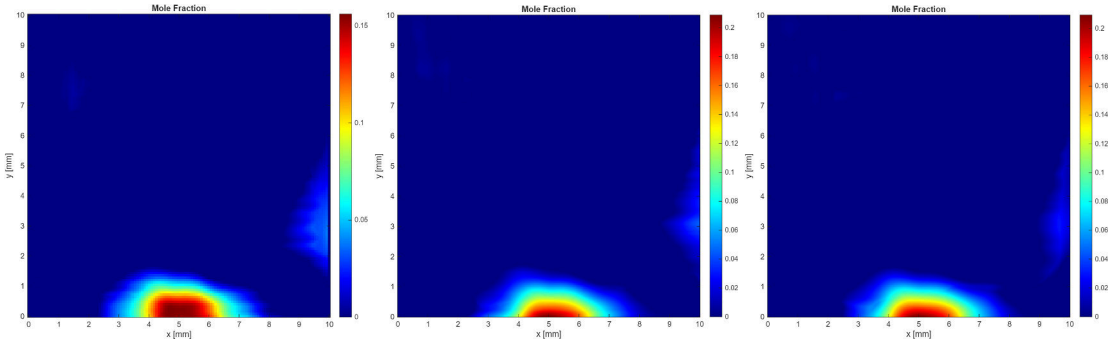
(b) 190 μs



(c) 200 μs



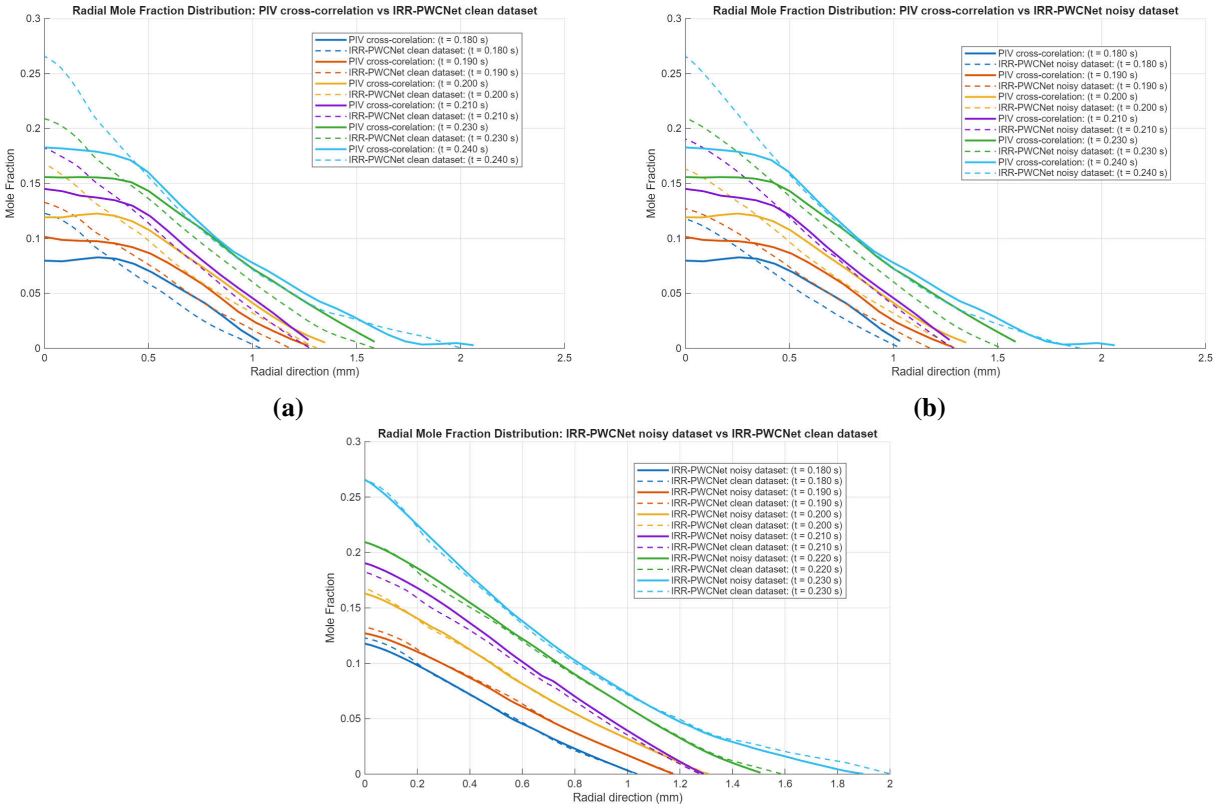
(d) 210 μs



(e) 220 μs

Figure 7: Comparison of molefraction : (A) PIV cross-correlation, (B) clean-dataset IRR-PWC-Net, and (C) noisy-dataset IRR-PWC-Net.

5.4 Comparison of radial molefraction



(a) **(b)**

(c)

Figure 8: Comparison of radial molefraction (a) PIV cross-correlation vs clean dataset IRR-PWC-Net,(b) PIV cross-correlation vs noisy dataset IRR-PWC-Net and (c) clean dataset IRR-PWC-Net vs noisy dataset IRR-PWC-Net

The evaporation dynamics of the droplet, just after impact on a heated substrate, show the droplet oscillating back and forth in the transverse direction, which can be clearly seen in the resultant displacement (Fig. 5), as well as in the corresponding reconstructed concentration fields given in (Fig. 7) from **180 μs** to **220 μs** . The primary mechanism driving this lateral oscillation is the competition between the droplet initial kinetic energy and the forces resisting its spread.The concentration field plot shows radial variation starting from r=0. Comparisons between PIV cross-correlation and the IRR-PWCNet model

for clean and noisy datasets (Fig. 8a, Fig. 8b) indicate that the irrotationality constraint in BOS modifies the displacement field, reducing blurriness in droplet evaporation dynamics relative to standard PIVlab cross-correlation. Additionally, Fig. 8c compares the trained model on clean and noisy datasets, demonstrating reduced noise in the reconstructed field.

6 Conclusion

The IRR-PWCNet framework represents a significant advancement in Background-Oriented Schlieren (BOS) imaging by seamlessly integrating the fundamental physics of irrotational displacement fields into a deep-learning architecture. By penalizing curl errors and maintaining a full 512×512 spatial resolution, this approach effectively overcomes the resolution limits and blurring artifacts typical of traditional cross-correlation methods in PIVlab. Furthermore, the model's robustness to sensor-calibrated noise ensures high-fidelity reconstructions that are both physically consistent and exceptionally sharp. Ultimately, this physics-based approach facilitates the visualization of transient evaporation structures and vapor mole-fractions with a level of detail that was previously lost to standard analysis, bridging the gap between complex fluid dynamics and neural network precision.

References

[1] J. V. Pastor, J. J. López, J. E. Juliá, and J. V. Benajes,'Planar Laser-Induced Fluorescence fuel concentration measurements in isothermal Diesel sprays', *Optics Express*, vol. 10, no. 7, pp. 309–323, 2002.

[2] V. Duke-Walker, B. J. Musick, and J. A. McFarland, 'Experiments on the breakup and evaporation of small droplets at high Weber number', *International Journal of Multiphase Flow*, vol. 161, pp. 104389, 2023

[3] A. M. J. Edwards, J. Cater, J. J. Kilbride, P. Le Minter, C. V. Brown, D. J. Fairhurst, and F. F. Ouali, 'Interferometric measurement of co-operative evaporation in 2D droplet arrays', *Applied Physics Letters*, vol. 119, no. 15, pp. 151601, 2021.

[4] L. Venkatakrishnan and G. E. A. Meier,'Density measurements using the Background-Oriented Schlieren technique', *Experiments in Fluids*, vol. 37, no. 2, pp. 237–247, 2004.

[5] P. Zhang, Z. Xu, T. Wang, and Z. Che, 'A method to measure vapor concentration of droplet evaporation based on background-oriented Schlieren', *International Journal of Heat and Mass Transfer*, vol. 168, pp. 120880, 2021.

[6] C. Mucignat, L. Manickathan, J. Shah, T. Rösgen, and I. Lunati, 'A lightweight convolutional neural network to reconstruct deformation in BOS recordings', *Experiments in Fluids*, vol. 64, pp. 1–16, 2023.

[7] J. Hur and S. Roth, 'Iterative Residual Refinement for Joint Optical Flow and Occlusion Estimation', in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5747–5756, 2019.

[8] D. Sun, X. Yang, M.-Y. Liu, and J. Kautz, 'PWC-Net: CNNs for Optical Flow Using Pyramid, Warping, and Cost Volume', *arXiv preprint arXiv:1709.02371*, 2018.

[9] K. Wei, Y. Fu, Y. Zheng, and J. Yang, 'Physics-Based Noise Modeling for Extreme Low-Light Photography', *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 11, pp. 8520–8537, 2022.

Hard Boundary Condition Enforcement for Learning-Based Boundary Value Problem Solvers

[1]V. Shah; [1]L. Fadia; [2,3]R. Barron*; [4]M. Hassanzadeh

[1]*Department of Electrical and Computer Engineering, University of Windsor, Canada*

[2]*Department of Mechanical, Automotive and Materials Engineering, University of Windsor, Canada*

[3]*Department of Mathematics and Statistics, University of Windsor, Canada*

[4] *Department of Electrical and Computer Engineering / Mathematics and Computer Science, Lawrence Technological University, U.S.A.*

Keywords: hard boundary conditions, learning-based BVP solvers, convection–diffusion equation, CNN

Abstract. Partial differential equations (PDEs) are the mathematical foundation for computational modelling in science and engineering. Classical mesh-based schemes (finite difference/volume/element) are accurate and reliable simulation tools but can become computationally expensive and may require stabilization or refinement strategies that complicate implementation and can damp important solution features. Machine-learning (ML) methods have emerged as attractive alternatives for accelerating PDE solvers or building fast surrogates. On structured grids, convolutional neural networks (CNNs) naturally mimic stencil-like locality. However, a persistent practical limitation remains, in that boundary conditions (BCs) are often enforced softly via penalty terms in the loss. This work addresses the BC issue by introducing a hard (explicit) BC constraint within a learning-based PDE solver framework. The central idea is to keep prescribed boundary information fixed and use a finite-difference discretization of the PDE to actively correct boundary-adjacent behaviour during training. Comparisons with and without the hard BC constraint show that explicit boundary enforcement improves stability and solution fidelity of the CNN predictions, particularly in near-wall regions.

1. Introduction

The behaviour of many phenomena in science and engineering can be described by partial differential equations (PDEs). In fluid mechanics and heat transfer, a set of convection-diffusion equations describe how quantities like velocity, pressure, density, and temperature change in space and time [1]. Two problems consistently show up as stress tests for numerical methods in PDEs: the Poisson and the convection-diffusion boundary value problems (BVPs), which are an elliptic (steady flow) or parabolic (unsteady flow) PDE that appears in many CFD formulations [1]. Traditionally, these BVPs are solved using mesh-based numerical schemes such as finite difference, finite volume, or finite element methods. These approaches are reliable and well understood, but they can be computationally expensive, particularly when the mesh is very fine and when many iterations are needed [1]. For most practical flow scenarios, the solution quality depends heavily on how accurately wall boundary conditions and near-wall behaviour are captured.

Recently, machine learning (ML) approaches have been explored as alternative PDE solvers or fast surrogates. Convolutional neural networks (CNNs) are attractive on structured grids because they naturally learn local stencil-like relationships. However, they generally suffer from the practical limitation that boundary conditions are often enforced softly through penalty terms in the loss function. As a result, predicted boundary values may not exactly match prescribed values unless the training is carefully tuned [2]. This is not a small issue since BVPs are known to be strongly controlled by their boundaries. Ren et al. [3] have noted that physical constraint mismatches can account for 20% of the total error in neural network models.

In this work, the mismatch at the boundary is addressed by introducing a hard boundary condition (BC) constraint within a learning-based PDE solver framework, focusing on Poisson and convection-diffusion problems. The key idea is to keep the boundary information fixed and use the PDE discretization to

*Corresponding Author, R. Barron, az3@uwindsor.ca

correct the near-boundary region during training. This correction is applied repeatedly (for example, at the end of each epoch), helping the learned solution remain compatible with the boundary constraints while the network focuses on learning the interior behaviour. Results with and without the hard BC constraint are compared to demonstrate how the explicit boundary enforcement improves robustness and overall solution quality.

2. Proposed Methodology

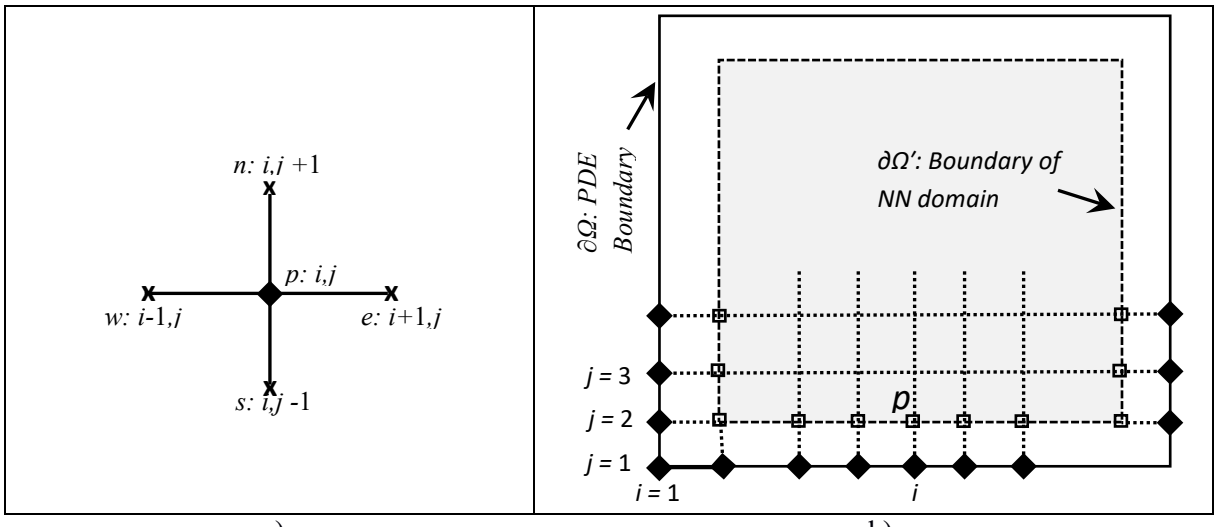
The CNN architecture for implementation of the hard boundary constraint for the Poisson and 2D convection–diffusion problems is designed on a decomposed domain, where the computational region is split into two parts: an inner square and an outer ring adjacent to the boundary. The model learns the solution over the inner subdomain while maintaining communication with the physical boundary conditions through the interface with the outer ring. The boundary values of the inner square are updated using information from the outer boundary [4]. For both soft and hard BC constraints, the network consists of an input layer with 2 neurons for the spatial coordinates (x,y), followed by two hidden layers with 64 neurons each and tanh activation functions, and a final output layer with 1 neuron for the scalar solution field. The model is trained in a coarse-to-fine manner on meshes of size 65x65, 129x129, 257x257 and 513x513, which allows the learned parameters from a coarser mesh to initialize the next finer mesh and improve convergence.

To formulate the hard boundary constraint, consider the general 2nd order convection-diffusion

$$u\frac{\partial \emptyset}{\partial x} + v\frac{\partial \emptyset}{\partial y} - \Gamma\nabla^2\emptyset = S(x,y) \tag{1}$$

defined on a domain Ω, with prescribed Dirichlet values on the boundary ∂Ω. Approximating derivatives with finite differences yields the corresponding finite difference equation. For first and second order discretizations, the difference operators at grid point (x_i, y_j) can be expressed as a linear combination of neighbor (*nb*) values on the stencil centered at (*i,j*), as illustrated in figure 1a, that is,

$$a_p\phi_p = a_w\phi_w + a_e\phi_e + a_s\phi_s + a_n\phi_n + S_p. \tag{2}$$



a) b)

Figure 1 a) Finite difference stencil, b) Physical and neural network domains (◆ physical boundary points for PDE; □ boundary points for NN).

For example, if all derivatives in equation (1) are approximated with second order central differencing at any grid point *p* in the solution domain, the coefficients in (2) are given by

$$a_p = 2\Gamma\left[\frac{1}{(\Delta x)^2} + \frac{1}{(\Delta y)^2}\right], \quad a_w = \Gamma\frac{1}{(\Delta x)^2} + \frac{u_p}{2\Delta x}, \quad a_e = \Gamma\frac{1}{(\Delta x)^2} - \frac{u_p}{2\Delta x}$$

*Corresponding Author, R. Barron, az3@uwindsor.ca

$$a_s = \Gamma \frac{1}{(\Delta y)^2} + \frac{v_p}{2\Delta y}, \quad a_n = \Gamma \frac{1}{(\Delta y)^2} - \frac{v_p}{2\Delta y}. \tag{3}$$

Implementation of hard boundary constraints on a rectangular domain Ω is illustrated in figure 1b. The physical boundary ∂Ω is isolated and the neural network is applied on the reduced domain Ω′ containing only the interior nodes of domain Ω. At the end of each epoch, the values of ϕ at the neural network boundary nodes are adjusted based on the discretized PDE at these nodes. Considering the neural network boundary node *p*:(*i*,2) in figure 1b and applying eqn. (2), the solution at *p* is updated according to

$$\phi_p = \left(a_w\phi_w + a_e\phi_e + a_B\phi_B + a_n\phi_n + S_p\right)/a_p \tag{4}$$

where ϕ_B is the prescribed boundary value at (*i*,1).

3. Results and Discussion

3.1 Poisson equation

Consider the Poisson equation $\nabla^2\emptyset = -\{2 + \pi^2 x(1-x)\}cos(\pi y)$ which has the exact solution $\emptyset = x(1-x)cos(\pi y)$ shown on the domain in figure 2. The corresponding Dirichlet boundary conditions for this "manufactured problem" are obtained from the exact solution, i.e., $\emptyset(-1,y) = -2\cos(\pi y)$, $\emptyset(1,y) = 0, \emptyset(x,-1) = \emptyset(x,1) = x(x-1), \emptyset(1-y,y) = y(1-y)\cos(\pi y)$.

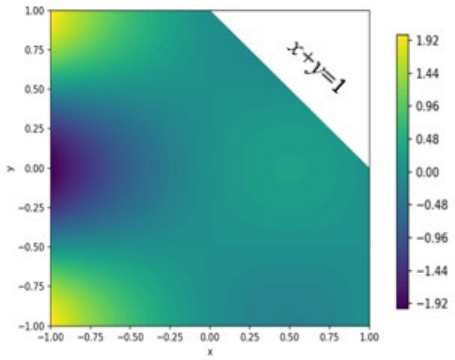


Figure 2 Exact solution of Poisson BVP on a cut-shaped domain.

3.1.1 Soft BC constraint for Poisson equation solver on cut-shaped domain. Table 1a evaluates cross-mesh generalization in which the model with soft boundary constraints is trained on a coarse mesh and evaluated on finer meshes. The overall trend is that errors reduce as the training mesh becomes finer, which is expected because a higher-resolution training mesh exposes the network to more detailed geometry and stronger gradients. For example, training on meshes 65x65, 129x129 and 257x257, and testing on the 513x513 mesh shows a reduction in MSE from 1.29e-6 to 5.97e-7, and the MAE drops from 6.73e-4 to 4.55e-4. This indicates that mesh resolution at training time strongly influences the achievable error floor. A second important observation is that testing on progressively finer meshes does not cause instability. For instance, when the model is trained on mesh 65x65 and tested on 129x129, 257x257 and 513x513, the error generally stays controlled and often decreases slightly. At the same time, because the boundary conditions are enforced as penalties (soft constraints), small boundary mismatches can persist. For elliptic problems like Poisson, boundary errors influence the entire domain, so even small boundary inconsistencies can prevent the method from reaching the lowest possible error.
3.1.2 Hard BC constraint for Poisson equation solver on cut-shaped domain. Table 1b repeats the same train and test mesh comparisons for hard boundary enforcement through boundary-adjacent correction. The pattern is consistent, with hard constraints yielding lower MSE and MAE across all listed train and test pairs relative to the soft constraint results. The improvement is already visible at coarse training. For example, training on 65x65 and testing on 129x129 reduces the MSE and MAE from 1.40e-6 and 6.96e-4 (soft), to 1.15e-6 and 6.52e-4 (hard), respectively. This shows that even when the solution is learned from a relatively coarse mesh, stronger boundary enforcement prevents the network from

*Corresponding Author, R. Barron, az3@uwindsor.ca

drifting near the walls. The largest gains appear at the highest resolution transfer. This is a meaningful reduction and supports the conclusion that strong enforcement of boundary conditions is essential for PDEs in which the solution is predominately boundary-driven.

Table 1. Effect of boundary constraints for Poisson BVP, a) soft constraints, b) hard constraints.

Training mesh	Test mesh	MSE	MAE
65x65	129 x129	1.40e-6	6.96e-4
65x65	257x257	1.32e-6	6.81e-4
65x65	513x513	1.29e-6	6.73e-4
129x129	257x257	7.79e-7	5.19e-4
129x129	513x513	7.58e-7	5.13e-4
257x257	513x513	5.97e-7	4.55e-4

(a)

Training mesh	Test mesh	MSE	MAE
65x65	129 x129	1.15e-6	6.52e-4
65x65	257x257	1.09e-6	6.40e-4
65x65	513x513	1.06e-6	6.33e-4
129x129	257x257	0.74e-6	5.01e-4
129x129	513x513	0.72e-6	4.96e-4
257x257	513x513	0.53e-6	4.22e-4

(b)

To further compare the soft and hard BC approaches, absolute errors between the exact and predicted solutions are plotted in figure 3, along the vertical line $x = 0.95$. In all cases, the hard BC constraint yields a more accurate solution, the difference being most noticeable where the boundary effects are strongest. Overall, figure 3 shows that while both methods are effective, the hard BC approach gives a more accurate solution and better error control for the Poisson equation.

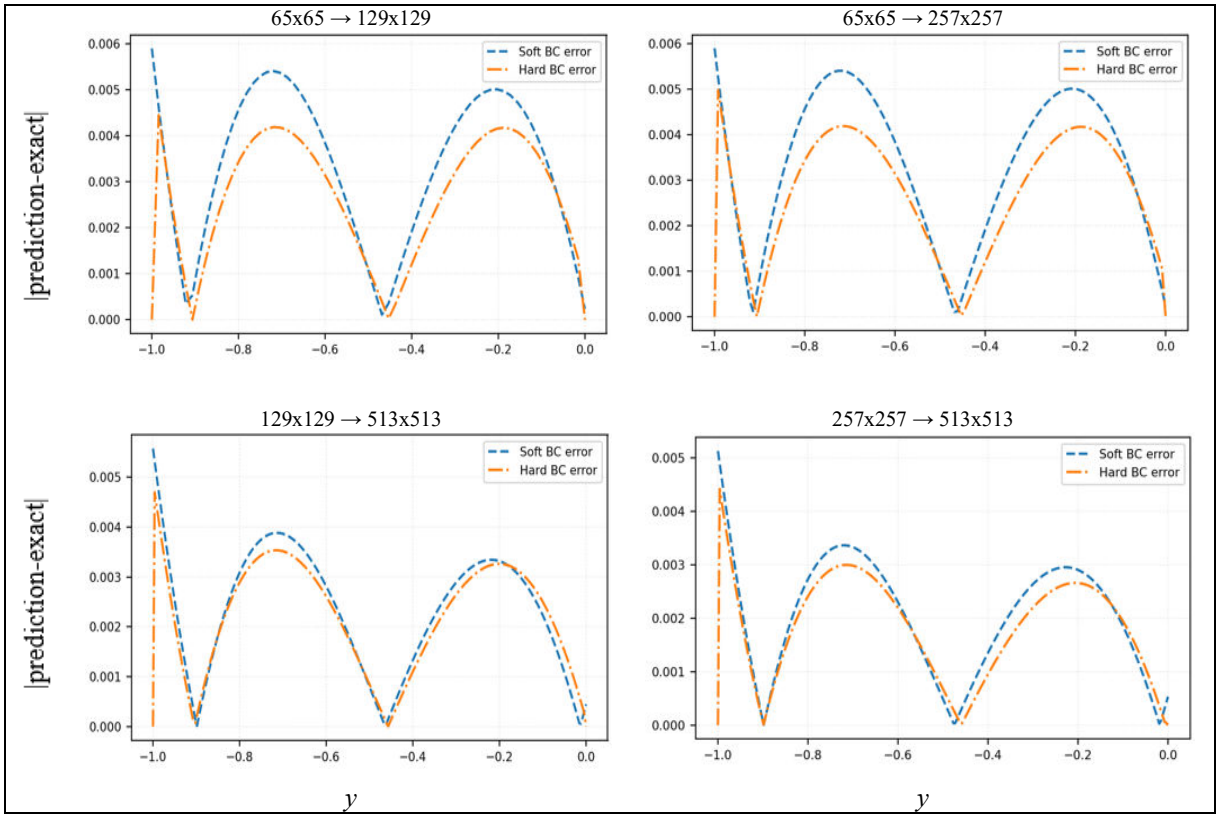


Figure 3 Absolute error plots along $x = 0.95$, for comparison of soft and hard implementation of boundary conditions (Poisson equation on a cut-domain).

3.2 Convection-diffusion equation

Ge and Zhang [5] investigated the effect of Reynolds number on the convection-diffusion equation (1), taken in non-dimensional form with $\Gamma = \mathrm{Re}^{-1}$. The convective velocity is chosen to be $u(x, y) = x(x - 1)(1 - 2y)$, $v(x, y) = -y(y - 1)(1 - 2x)$ and the source term is given by $S(x, y) = -(u +$

*Corresponding Author, R. Barron, az3@uwindsor.ca

$1)b'(x)\{1-b(y)\}-(v+1)b'(y)\{1-b(x)\}$, where $b(\xi)=\{\exp(-\xi Re)-\exp(-Re)\}/\{1-\exp(-Re)\}$. The exact solution for this problem is $\emptyset(x,y)=[1-b(x)][1-b(y)]$, satisfying boundary conditions $\emptyset(0,y)=0,\ \emptyset(1,y)=1-b(y),\ \emptyset(x,0)=0,\ \emptyset(x,1)=1-b(x)$.

3.2.1 Soft boundary constraint for 2D convection-diffusion equation solver. For all Re in table 2, the test mesh is 513x513. Table 2a shows that, as Re increases and the PDE becomes more convection-dominated, the learning problem is more challenging. For Re = 1.0, the errors are relatively stable, with MSE of the order of 1.0e-5 and MAE around 2.5e-3. For Re = 10, errors increase and become more sensitive to the mesh pairing. This indicates that, at this Re, near-boundary behaviour is starting to dominate performance, and soft enforcement struggles to reproduce consistent boundary-layer structure when transferring between resolutions. For Re = 100, the error increases dramatically. This large jump is typical of strongly convection-dominated regimes, where thin layers form and small near-boundary inaccuracies create large global errors. In fact, close analysis of the exact solution indicates very high gradients near the boundaries x = 0 and y = 0, where the solution increases from a value of 0 on the boundary to a maximum value of 1 within a very thin boundary layer.

3.2.2 Hard boundary constraint for 2D convection-diffusion equation solver. The data in table 2b show that hard constraints generally improve performance for moderate convection influence. However, there are also cases where hard constraints do not clearly outperform soft constraints, such as training on 257 and testing on 513 at Re = 10, where the hard result is only comparable to or slightly worse than the soft result. This suggests that for this particular pairing the bottleneck is not purely boundary enforcement. Optimization stability and representation of steep gradients likely become the limiting factors. For Re = 1, hard constraints provide significant gains in transfers shown in table 2b. This indicates that even in smoother regimes, enforcing boundary consistency can significantly improve cross-resolution transfer. For Re = 100, hard constraint errors remain high, supporting a clear and honest assessment that, when the problem is strongly convection-dominated, boundary hardening alone is not enough, and additional stabilization is needed, such as stronger near-layer sampling, more robust discretization-consistent training, or regime-specific tuning.

Table 2. Effect of boundary constraints for convection-diffusion BVP, a) soft, b) hard.

Re	Training mesh	MSE	MAE
1	65x65	1.02e-5	2.57e-3
1	129 x129	0.92e-5	2.43e-3
1	257x257	1.18e-5	2.70e-3
10	65x65	1.90e-5	2.74e-3
10	129 x129	4.28e-5	3.90e-3
10	257x257	4.94e-5	3.52e-3
100	257x257	134.0e-5	14.3e-3

(a)

Re	Training mesh	MSE	MAE
1	65x65	0.72e-5	2.04e-3
1	129x129	0.16e-5	0.95e-3
1	257x257	0.84e-5	2.64e-3
10	65x65	0.69e-5	1.39e-3
10	129x129	0.41e-5	1.26e-3
10	257x257	5.03e-5	3.62e-3
100	257x257	225.8e-5	16.95e-3

(b)

Figure 4 compares absolute errors between the exact solution and those predicted by the soft BC and hard BC approaches, along a vertical section near x = 0.95 for Re equal to 1 and 10. For these values of Re, this figure shows that both boundary constraint methods produce relatively small errors throughout most of the domain, confirming that both can approximate the exact solution well. However, the soft BC results show larger oscillations and higher error peaks, especially near the boundary regions. In contrast, the hard BC curves are smoother and remain lower, indicating better boundary enforcement and improved stability. Overall, these figures demonstrate that both approaches are capable of capturing the solution behaviour, but the hard BC method consistently provides higher accuracy and better stability, particularly near the boundaries.

*Corresponding Author, R. Barron, az3@uwindsor.ca

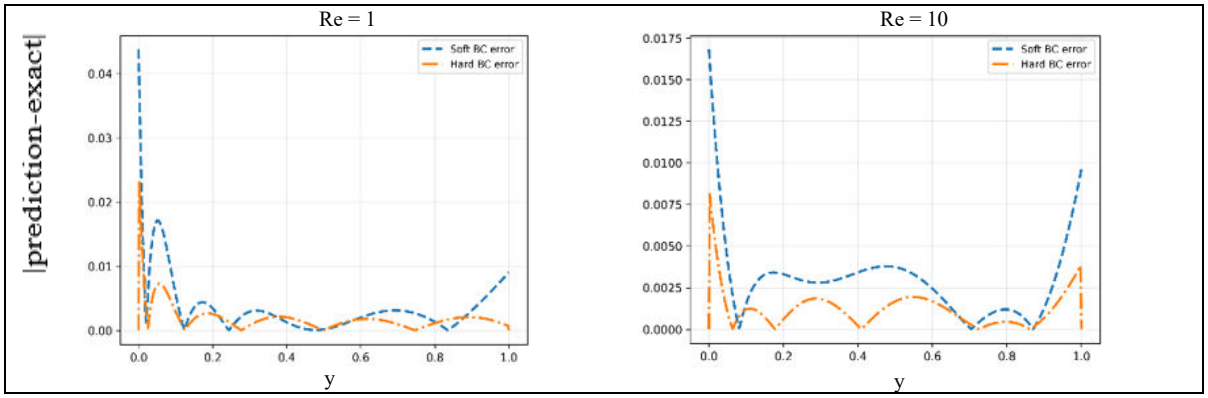


Figure 4 Absolute error plots along x=0.95 for Re equal to 1 and 10 for soft and hard BC constraints for the convection-diffusion equation

4. Conclusions

In this work, the effect that boundary condition enforcement has on the performance of learning-based PDE solvers has been investigated, focusing on Poisson and 2D convection–diffusion BVPs. It is observed that soft boundary condition enforcement, which is commonly used in neural network approaches, can lead to mismatches at the boundaries. These errors may seem minor, but it is shown that they can influence the entire solution and reduce overall accuracy. To address this issue, a hard boundary condition approach has been introduced, where the boundary values are strictly enforced during training using a finite difference-based correction at boundary-adjacent nodes. This allows the model to maintain consistency with the governing equations while improving the solution near the boundaries.

The results show that the hard boundary condition approach consistently improves accuracy for Poisson problems across different mesh resolutions. For the convection–diffusion equation, the method performs well in moderate convective flow regimes (Re = 1 and 10), where boundary effects are important. However, in strongly convection-dominated cases (Re = 100), the errors remain high even with hard boundary enforcement, indicating that the main difficulty comes from failure to capture sharp boundary-layer gradients and interior physics rather than just boundary conditions.

Overall, the study shows that enforcing boundary conditions explicitly can improve stability and accuracy in learning-based PDE solvers. At the same time, it also highlights that additional techniques are needed for more challenging regimes. Future work can explore combining this approach with stabilization methods, better sampling strategies near boundary layers, and improved model architectures.

References

[1] H. K. Versteeg, W. Malalasekera, An Introduction to Computational Fluid Dynamics: The Finite Volume Method, 2nd ed. Harlow, U.K.: Pearson Education, 2007.
[2] I. E. Lagaris, A. Likas, D. I. Fotiadis, ‘Artificial neural networks for solving ordinary and partial differential equations,’ IEEE Trans. Neural Networks, vol. 9, no. 5, pp. 987–1000, 1998. DOI: 10.1109/72.712178
[3] Z. Ren, S. Zhou, D. Liu, Q. Liu, ‘Physics-Informed Neural Networks: A Review of Methodological Evolution, Theoretical Foundations, and Interdisciplinary Frontiers Toward Next-Generation Scientific Computing,’ Appl. Sci., vol. 15, 8092-8109, 2025. https://doi.org/10.3390/app15148092.
[4] C. M. Jiang, K. Kashinath, Prabhat, P. Marcus, ‘Enforcing physical constraints in CNNs through differentiable PDE layer,’ ICLR 2020 Workshop on Integration of Deep Neural Models and Differential Equations, 2020.
[5] L. Ge, J. Zhang, ‘Accuracy, robustness and efficiency comparison in iterative computation of convection diffusion equation with boundary layers,’ Num. Methods in PDEs, vol. 16, no. 4, pp. 379-394, 2000. https://doi.org/10.1002/1098-2426(200007)16:4<379::AID-NUM3>3.0.CO;2-I.

*Corresponding Author, R. Barron, az3@uwindsor.ca

Adjacency-Preserving Tokenization for Transformer-Based Quadrilateral Mesh Generation

[1]T. Rentschler*; [1]A. Tismer; [1]S. Riedelbauch;
[1]Institute of Fluid Mechanics and Hydraulic Machinery, University Stuttgart, Germany

Keywords: Quadrilateral Mesh, Block-Structured, Transformer, Tokenization, Turbomachinery

Abstract.

Transformers have demonstrated significant advancements in mesh generation. However, their quadratically increasing memory and computational requirements significantly limit their applicability to CFD simulations with millions of mesh elements. To reduce the number of tokens to be generated per mesh, we present two methods on different levels. First, we generate coarse quadrilateral block structures suitable for refinement via transfinite interpolation to produce high-quality CFD meshes, thereby reducing the number of mesh faces. Second, we propose an adjacency-preserving tokenization strategy that chains consecutive faces sharing edges, improving convergence and prediction quality compared to conventional lexicographic ordering. Building upon this, we introduce sparse row-structured tokenization that eliminates redundant vertex tokens through shared edges, reducing sequence length by $\sim$40% and GPU memory by $\sim$50%. We demonstrate effective training on consumer-grade GPUs, producing block structures suitable for turbomachinery CFD applications.

1. Introduction

The generation of meshes that are suitable for CFD simulations typically involves millions of elements, a task that is particularly challenging for Transformer-based architectures. This issue is addressed in two distinct levels. Initially, the number of faces per mesh is reduced. Subsequently, the number of tokens required to represent a mesh for the Transformer is decreased.

The number of faces needed to generate by the Transformer autoregressively is reduced by generating coarse block structures using a quadrilateral domain partition algorithm that divides the geometry into large quadrilateral regions. Data augmentation is employed to generate coarse blocks that are structured based on these quadrilateral subregions. These block structures can be refined to high resolution through transfinite interpolation in a post processing step.

Conventional tokenization approaches to autoregressive mesh generation, as introduced by Nash et al. [6], employ lexicographic ordering where vertices are sorted in yzx order (yx in 2D) within each face, and faces are arranged by their first vertex [3]. This produces a layer-by-layer arrangement for uniform grids where adjacent faces sharing edges are consecutive in the token sequence. However, for meshes around objects like turbomachinery blades and coarse block structures with varying element sizes, lexicographic ordering breaks down: consecutive faces in the token sequence are often not adjacent faces within the mesh, creating discontinuities in the token representation and increasing prediction difficulty. The blade cut-out amplifies this effect, introducing frequent jumps even for simple geometries.

In order to ensure that adjacent faces appear as consecutive tokens in the sorted sequence, a half-edge data structure [2] is employed to traverse faces by edge connectivity. This eliminates the need for duplication of vertex tokens, as vertices shared between consecutive faces are tokenized only once instead of once per face, thereby reducing computational overhead. Discontinuities are reduced compared to the lexicographic sorting and only appear between rows of chained faces.

2. Data

The dataset under consideration incorporates 2000 generated turbine blades, with a random scaling factor varying from 0.50 to 0.95 and random rotation within the interval $[-\pi/4, \pi/4]$ applied to each individually. Each profile is decomposed into a set of streamlines via a domain partition algorithm [7, 5], dividing the geometry into quadrilateral regions. The process consists of: (1) computing boundary crosses from local normal and tangent vectors, (2) mapping to representative vectors, (3) propagating the frame field to interior nodes, (4) detecting singularities using the Poincaré index, and (5) generating streamlines from singularities by following the cross field direction. Since configurations with 4 or more singularities already yield partitions with a large number of quadrilateral elements and significant size variation, the dataset was restricted to meshes containing exactly 2 singularities, producing a base block structure of 6 faces, as shown in Figure 1.

As illustrated in Figure 1, augmentation is performed by subdividing each edge once (4 blocks, center) and twice (9 blocks, right).

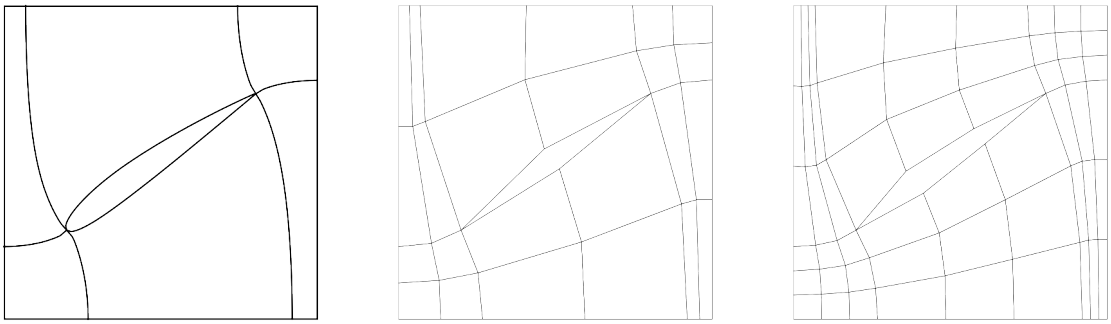
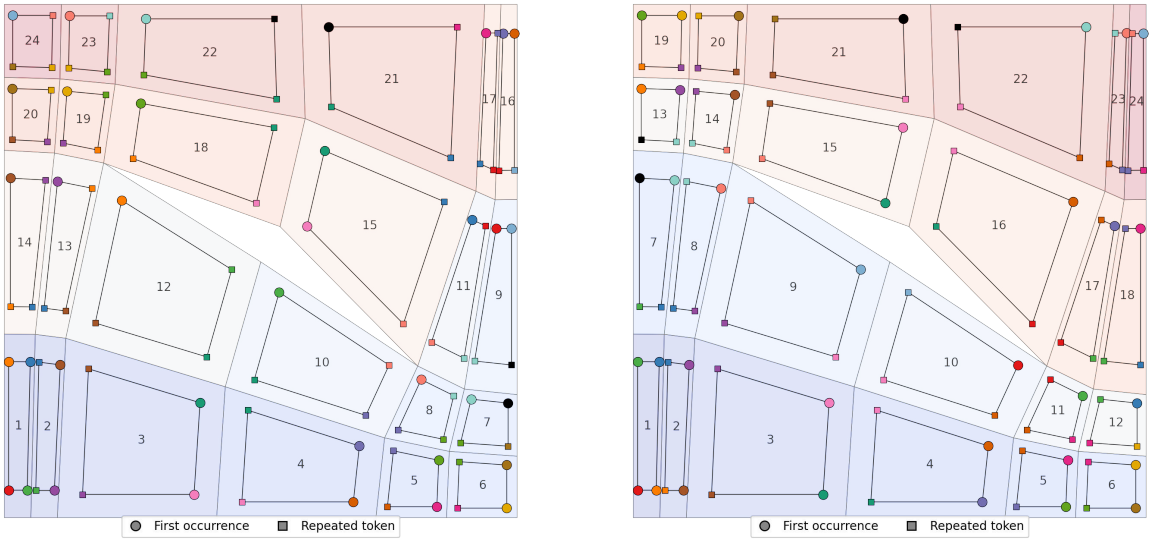


Figure 1. Streamlines (left), 1x subdivision — 4 blocks (center), 2x subdivision — 9 blocks (right).

3. Tokenization

Tokenization is defined as the process of converting raw data into a set of discrete vectors, thereby transforming the data into a format suitable for transformer-based architectures. As Transformers learn statistical patterns from training data, a consistent ordering of mesh elements that yields the same token sequence and clear patterns for each mesh is essential for autoregressive generation. In the initial tokenization stage, equation (1) is employed to transform the coordinates into a discrete set of integers, within the range of $[0, n–1]$. The parameter n represents the total number of integers and, consequently, the number of discretized coordinates. Larger n yields finer resolution but increases model capacity requirements. The lower bound is imposed by the shortest edge length in the dataset.

In the subsequent step, a unique vector is assigned to each integer. In addition, the application of special vectors are required. This includes the implementation of vectors for Start-Of-Sequence (SOS), End-Of-Sequence (EOS), End-of-Row (EOR), and Padding (PAD) to mark the sequence boundaries and



(a) Lexicographic ordering

(b) Adjacency-preserving ordering

Figure 2. Comparison of lexicographic (left) and adjacency-preserving (right) ordering. Face order is color-coded from low (blue) to high (red), plotted at the center of each face. Repeated tokens are marked with squares, first occurrences with circles.

padding.

$$C_{\text{discretized}} = \text{round}\left(\frac{x_i - x_{\min}}{x_{\max} - x_{\min}} \cdot (n-1) \right) \tag{1}$$

3.1 Lexicographic Ordering

Nash et al. [6] introduced lexicographic ordering where vertices are sorted in yx order and faces are arranged by their first vertex. For uniform grids, this produces a layer-by-layer arrangement where adjacent faces appear consecutively in the token sequence. However, for meshes around objects like turbomachinery blades and coarse block structures with varying element sizes, consecutive faces in the token sequence are often not adjacent, creating discontinuities as shown in Figure 2a and 3.

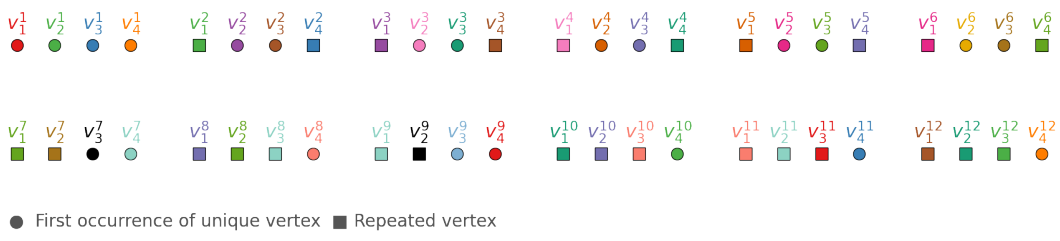


Figure 3. The vertex sequence displayed is generated by lexicographic ordering method applied to the first twelve faces of Figure 2a). The index and superscript are used to denote order and face number in the sequence, respectively. The initial occurring tokens are indicated by the use of circles. Repeated tokens, represented by squares, do not possess a clearly defined position within the sequence.

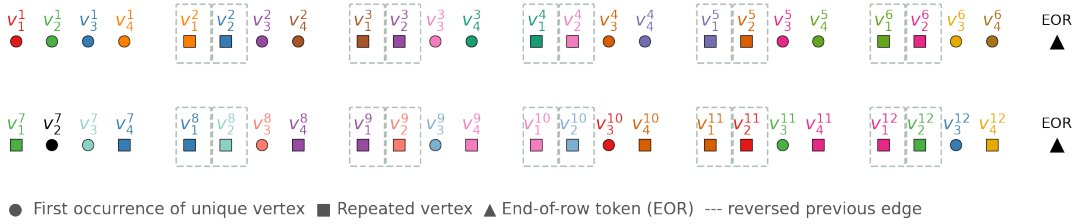


Figure 4. The vertex sequence displayed is generated by adjacency-preserving face ordering method applied to the first two rows from figure 2b. The exit edge of face i becomes the entrance edge of face $i + 1$ in reversed direction. Redundant tokens, highlighted by the gray dotted box, possess a distinct position and can be eliminated. End-of-row tokens are represented by stars.

3.2 Adjacency-Preserving Ordering

The proposed method preserves adjacency through a half-edge-based traversal strategy [2] that chains consecutive faces sharing an edge. Starting from the lexicographically first face, faces are traversed in ascending x-direction until no adjacent face in this direction exists. The next row starts with the lexicographically first unvisited face.

To ensure the desired adjacency pattern, displayed in Figure 2b and 4, where the last two vertices of face i equal the first two vertices of face $i + 1$ in reversed order, all rows progress left-to-right: rows progressing right-to-left are reversed. Vertices within a face are ordered clockwise starting from the bottom-left corner. The exit edge of face i (vertices $v_3^i \rightarrow v_4^i$) thereby becomes the entrance edge of face $i + 1$ (vertices $v_1^{i+1} \rightarrow v_2^{i+1}$) in reversed direction, enabling a sparse row-structured representation.

3.3 Sparse Row-Structured Tokenization

The application of adjacent-preserving face order, in combination with vertex sorting as previously described, allows for the efficient use of sparse row-structured tokenization. The utilization of all four vertices is only necessary at the initial face in a row, due to the presence of repetitive redundant tokens. Subsequent faces in a row do not require the generation of the first two vertices. Consequently, only the remaining two are considered, in contrast to the conventional approach which utilizes all four vertices for each face. Consequently, the implementation of an end-of-row (EOR) token is essential to denote the end of each row. This strategy reduces the sequence length by 40% in the employed dataset, from a maximum of 448 to 272 tokens, and decreased GPU memory usage by 50%.

4. Architecture

The Transformer architecture [9] employed for quadrilateral mesh generation consists of a stack of n stages, where each stage comprises m masked self-attention layers followed by a single cross-attention layer. Token representations are processed through the self-attention layers with causal masking to enforce autoregressive generation, while conditioning information from geometric context encoders is injected via cross-attention after each stage. Geometric context is provided by cross-attention using two parallel encoders /refmeshtron: A Point Perceiver encoder [4] and a Face Count encoder based on sinusoidal encoding. The Point-Perceiver encoder processes the input point cloud, which is generated by a Delaunay triangulation, into a fixed set of latent embeddings capturing global geometry. The Face-Count encoder maps the desired number of faces to a learned embedding using sinusoidal encoding, providing explicit control over mesh density. These conditional embeddings are concatenated and supplied as key and value inputs to the cross-attention mechanism, ensuring that geometric guidance is available at every

layer. Rotary Positional Embeddings (RoPE) [8] are employed throughout the architecture, encoding relative position information directly into the attention computation. RoPE supports extrapolation to sequence lengths beyond those seen during training, which is essential when generating meshes with varying face counts.

5. Results

The three tokenization strategies were evaluated using a three-stage Optuna optimization pipeline [1] based on bits-per-token (equation 2) on validation data. Two independent optimization runs were conducted, with parameter ranges refined between runs. Each stage builds upon the best configuration from the previous stage:

- **Stage A (Architecture):** Model dimension d_{model}, number of attention heads, number of stages (2 to 5), layers per stage (2, 4, 6), and latent embeddings (4, 8, 16, 32, 64).
- **Stage B (Hyperparameters):** Learning rate (log-scale), warmup steps, dropout, and weight decay.
- **Stage C (Training Dynamics):** Batch size, gradient accumulation steps.

Optuna's MedianPruner with Tree-structured Parzen Estimator (TPE) sampling [1] was used to prune unpromising trials. In stage A, 50 trials were completed for each method, while 30 trials were completed in stages B and C. Final model evaluation was performed by training the optimized configurations with three independent random seeds across two independent optimization runs to ensure reproducibility.

5.1 Evaluation Metrics

The primary metric for model comparison is bits-per-token (BPT), defined as:

$$\text{BPT} = -\frac{1}{N}\sum_{i=1}^{N}\log_2 p(x_i) \tag{2}$$

where N is the sequence length and $p(x_i)$ is the predicted probability of token x_i. BPT quantifies the mean information content per predicted token. Lower values correspond to higher compression and prediction quality.

The vocabulary consists of 256 discrete coordinate tokens, in addition to three special tokens used for the Lexicographic and Adjacency-Preserving (SOS, EOS, and PAD) methods. The Sparse Row-Structured method employs the same three unique tokens and additional one EOR token. To evaluate the holistic efficiency, the bits-per-mesh (BPM) are computed by multiplying the BPT with the number of tokens per mesh, combining prediction quality with sequence length.

5.2 Bits-per-Mesh Analysis

Table 1. Final comparison of the three tokenization strategies.

Strategy	BPT	Tokens	GPU Memory	BPM	Accuracy
Lexicographic	0.388 ± 0.038	448	$\sim$15866 MiB	173.8	76.4%
Adjacency-preserving	$\mathbf{0.340} \pm 0.009$	448	$\sim$15866 MiB	152.3	**79.0%**
Sparse Row-Structured	0.551 ± 0.012	272	**$\sim$8379 MiB**	**149.9**	68.3%

Table 1 reveals an important distinction between the two evaluation metrics. While Adjacency-preserving achieves the lowest per-token loss (0.340 BPT, 79.0% accuracy), Sparse Row-Structured achieves the highest BPT (0.551, 68.3% accuracy) yet the lowest bits-per-mesh (149.9) due to its 39.2% shorter sequence length. Lexicographic (0.388 BPT, 76.4% accuracy) occupies the middle ground. This demonstrates that the row-compressed tokenization compensates for its higher per-token prediction difficulty through reduced sequence length - an effect that becomes increasingly significant as mesh complexity grows. The GPU memory requirements represent peak inference memory for the 54-face base meshes, with Sparse Row-Structured achieving 52.8% reduction ($\sim$8379 MiB vs $\sim$15866 MiB) compared to the other strategies.

The adjacency-preserving strategy demonstrates superiority over lexicographic sorting in several respects. The structural coherence ensures that adjacent faces remain adjacent in the token sequence, thereby accelerating convergence and consistently achieving better local minima. Additionally, the adjacency-preserving strategy exhibits reduced variance across multiple trainings, suggesting more stable optimization dynamics.

The row-compressed tokenization strategy, which builds upon the adjacency-preserving approach, demonstrates acceptable performance with significant memory reduction. By eliminating redundant vertex tokens in consecutive faces, it achieves a $\sim$40% shorter sequence length while maintaining prediction quality through the reduced sequence length.

Acknowledgments

This work was funded by the German Research Foundation (DFG) under the special priority program SPP-2353: 501932169

References

[1] Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. Optuna: A next-generation hyperparameter optimization framework. *arXiv preprint arXiv:1907.10902*, 2019.

[2] Mario Botsch, S. Steinberg, Stephan Michael Bischoff, and Leif Kobbelt. OpenMesh - a generic and efficient polygon mesh data structure. In *OpenSGplus Workshop Proceedings*. OpenSGplus Workshop, 2002.

[3] Zekun Hao, David W. Romero, Tsung-Yi Lin, and Ming-Yu Liu. Meshtron: High-fidelity, artist-like 3d mesh generation at scale, 2024.

[4] Andrew Jaeger and et al. Perceiver: General perception with iterative attention. *arXiv preprint arXiv:2103.03206*, 2021.

[5] Nicolas Kowalski, Franck Ledoux, and Pascal Frey. A pde based approach to multidomain partitioning and quadrilateral meshing. In *Proceedings of the 21st International Meshing Roundtable*, pages 137–154, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.

[6] Charlie Nash, Yaroslav Ganin, S. M. Ali Eslami, and Peter W. Battaglia. Polygen: An autoregressive generative model of 3d meshes, 2020.

[7] Tobias Rentschler, Alexander Tismer, and Stefan Riedelbauch. Frame field prediction for quadrilateral domain partition. In Lucas A. Hof, Giuseppe Di Labbio, Antoine Tahan, Marlène Sanjosé, Sébastien Lalonde, and Nicole R. Demarquette, editors, *Proceedings of the CSME-CFDSC-CSR 2025 International Congress*, number 8 in Progress in Canadian Mechanical Engineering, 2025.

[8] Jianlin Su, Yu Lu, Shengfeng Pan, Bo Wen, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *CoRR*, abs/2104.09864, 2021.

[9] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2023.

Physics-Guided Graph Neural Networks for Joint Optimization of Sensor Placement and Sparse Flow Reconstruction

[1, 2]Shuzhi Ni; [1]Zi-Xiang Tong; [1]Yimin Zhang; [1]Jianqin Zhu*

[1]*School of Energy and Power Engineering, Beihang University, Beijing, 100191, China*

[2]*Shen Yuan Honors College, Beihang University, Beijing, 100191, China*

Abstract
High-fidelity flow reconstruction from sparse sensors is critical for engineering condition monitoring, yet challenged by modal truncation bias in traditional linear reduced-order models and the lack of physical constraints in black-box deep learning. Furthermore, existing paradigms typically fail to jointly optimize sensor placement and field prediction. To address this, we propose a physics-informed Graph Neural Network (GNN) framework to achieve joint optimization of discrete sensor layouts and global flow reconstruction. By injecting singular-value-weighted Proper Orthogonal Decomposition (POD) modes as physical priors and employing a continuous probability relaxation via the Straight-Through Gumbel-Softmax estimator, the framework overcomes the non-differentiability of discrete sensor selection. Evaluated on a cylinder wake dataset at Re=100, our strategy outperforms traditional POD and QR pivoting baselines. It effectively suppresses error amplification and leverages nonlinear mapping to compensate for the flow dynamics of the truncated modes. Moreover, the autonomously optimized sensors efficiently track shear layer separation lines rather than clustering locally. Across 1% to 20% measurement noise levels, the framework demonstrates reliable consistent robustness and statistical stability. These findings validate the efficacy of physics-informed joint optimization in low Reynolds number flow, offering a foundational step toward estimation in complex turbulent environments.

Keywords: Sparse flow reconstruction; Sensor placement; Graph neural network; Physics-informed deep learning

1. Introduction

High-fidelity physical field reconstruction of complex flows is crucial for equipment condition monitoring and digital twins [1]. However, in practical industrial applications, acquiring dense data is often impossible due to limited physical space, harsh environments, and high sensor costs. Typically, only sparse local observations are available [2]. Consequently, achieving high-precision and robust reconstruction of global flow fields from limited sparse sensors remains a significant challenge in data-driven modeling.

Traditionally, methods based on Reduced-Order Model (ROM) are widely used for this problem. Among them, the sensor placement strategy combining Proper Orthogonal Decomposition (POD) and QR pivoting (Q-DEIM) serves as a standard baseline for low-fidelity reconstruction due to its solid mathematical foundation [3]. Nevertheless, these traditional methods rely on linear modal superposition. Since the underlying POD bases are extracted from Computational Fluid Dynamics (CFD) data, low-rank POD truncation inevitably introduces simulation bias [4]. This inherent truncation error makes the linear baseline prone to error amplification and reconstruction failure when dealing with local complex flow patterns unresolved by low-energy modes or in high noise level environments.

Recently, deep learning methods have demonstrated excellent nonlinear fitting capabilities in predicting complex physical fields [5]. However, existing purely data-driven paradigms have two major limitations. First, they often separate sensor placement optimization and flow field reconstruction into two independent stages, failing to achieve joint optimization [6]. Second, purely data-driven black-box models lack the constraints of underlying fluid mechanics. They could struggle with convergence and may yield non-physical predictions under unseen conditions or noisy inputs [7].

*Corresponding Author, Jianqin Zhu zhujianqin@buaa.edu.cn

To address these limitations, this paper proposes a physics-informed Graph Neural Network (GNN) framework. Through a three-stage deep graph learning, it achieves the joint optimization of discrete sensor placement (Selector-GNN) and flow field reconstruction (Reconstructor-GNN). Unlike unguided black-box models, our method injects traditional POD physical features into the neural network as prior knowledge. This retains the nonlinear fitting capacity of deep learning while significantly narrowing the hypothesis space for reconstruction. Specifically, the framework designs static physical representations for graph nodes, including POD energy-weighted modes and time-averaged spatial gradients. A multi-mask adaptation mechanism is introduced during training to enhance robustness against varying discrete sensor layouts and observation noise. Furthermore, since sensor selection is inherently a non-differentiable discrete combinatorial optimization problem, we introduce the Straight-Through (ST) Gumbel-Softmax estimator and soft-thresholding techniques [8]. This effectively enables gradient backpropagation through the discrete mask, ensuring stable joint convergence of both the selector and the reconstructor.

2. Method

2.1 Physical Feature Extraction and Graph Construction

This study investigates the two-dimensional incompressible flow past a cylinder at a Reynolds number of *Re*=100. We utilize a high-fidelity Direct Numerical Simulation (DNS) dataset open-sourced by Fukami et al. [5]. The dataset is chronologically divided into a training set (2500 snapshots), a validation set (1250 snapshots), and a test set (1250 snapshots). To explicitly inject fluid mechanics priors into the deep learning model, we first subtract the mean field from the training snapshots. Singular Value Decomposition (SVD) is then performed on the resulting fluctuating flow matrix to extract the first r POD modes $\boldsymbol{\Phi}$ and their corresponding singular values. The POD modes represent the basic coherent structures in the flow field, while the singular values indicate the relative importance of each mode. Detailed mathematical derivations of the POD method can be found in [9].

For the graph representation, the original 112×192 computational grid is mapped into graph nodes. Since different POD modes contribute differently to the dynamic reconstruction, the GNN should prioritize its learning accordingly. Therefore, we use the physical priors extracted from the training set to weight the node input POD modes with their singular values. This guides the network to focus on dominant flow structures containing high energy. Specifically, by combining normalized spatial coordinates and time-averaged flow information, the comprehensive static physical feature vector $\boldsymbol{x}_i$ for node i is defined as:

$$\boldsymbol{x}_i = \left[\frac{\sigma_1}{\sigma_1}\Phi_{i,1}, \frac{\sigma_2}{\sigma_1}\Phi_{i,2}, \cdots, \frac{\sigma_r}{\sigma_1}\Phi_{i,r}, \bar{u}_i, |\nabla \bar{u}_i|, x_i, y_i \right]^{\mathrm{T}}$$

Here, $\Phi_{i,k}$ is the spatial distribution value of the *k*-th mode at node i. The weighting coefficient is the ratio of the corresponding singular value σ_k to the first singular value σ_1, serving as a numerical mapping of modal importance. $\bar{u}_i$ and $|\nabla \bar{u}_i|$ represent the time-averaged value and the spatial gradient magnitude at the node, respectively. x_i and y_i are the normalized spatial coordinates.

Furthermore, real flow fields contain large areas with extremely low energy that contribute little to global reconstruction, such as the far-field wake region. During the final graph construction, we truncate 75% of the low-energy nodes based on their cumulative POD energy. This physical pruning strategy eliminates low-efficiency nodes in advance. It significantly reduces the search space within the massive computational domain, thereby improving the efficiency of sensor sampling and GNN optimization.

2.2 Joint Optimization Network Architecture

Based on the aforementioned graph structure, we design an end-to-end network architecture comprising three core modules.

*Corresponding Author, Jianqin Zhu zhujianqin@buaa.edu.cn

The first module is the Selector-GNN, which is responsible for sensor location optimization. Selecting a specific node as a sensor is inherently a binary discrete decision. Directly outputting discrete variables $\{0,1\}$ leads to gradient truncation during backpropagation, failing to guide network updates. Therefore, our network first outputs a raw score indicating the likelihood of a node becoming a sensor, which is then mapped into a continuous probability $z_{soft,i} \in (0,1)$. During the forward pass, the model is forced to select *p* discrete sensor nodes that are not spatially clustered, generating a strict binary mask $z_{hard,i} \in \{0,1\}$.

The second module is the SensorSetEncoder. After obtaining signals from discrete sensors, this module locally encodes the sparse observations and their physical coordinates to generate a global latent variable g_b. This variable represents the current transient flow features. To prevent g_b from degenerating into meaningless numerical fitting, the network explicitly uses it to predict the true POD temporal coefficients $\boldsymbol{a} = \boldsymbol{\Phi}^{\mathrm{T}} \boldsymbol{u}'$ of the current snapshot. This enforces the global encoding to strictly carry the physical evolutionary state of the flow field.

The third module is the Reconstructor-GNN, which undertakes the final global decoding task. Built upon a GraphSAGE backbone, it incorporates a gated residual connection mechanism. The cross-layer feature update is defined as:

$$\boldsymbol{h}^{(l)} = \boldsymbol{h}^{(l)}_{\mathrm{new}} + \tanh(\boldsymbol{\alpha})\boldsymbol{h}^{(l-1)}$$

This reconstructor jointly processes the local physical features of nodes, the sensor mask, and the global variable g_b. It effectively preserves shallow, local high-frequency flow details during deep information passing, ultimately outputting a high-fidelity fluctuating flow field.

3. Result and discussion

To comprehensively evaluate the performance of the joint optimization framework $\mathrm{GNN_{sel}}$+$\mathrm{GNN_{rec}}$ under sparse observations, this section conducts a quantitative analysis with a fixed sensor budget ($p = 20$). We explore two dimensions: physical prior truncation strength (number of POD modes r) and measurement noise robustness.

3.1 Impact of Modal Truncation on Global Reconstruction Accuracy

In ROM-based flow reconstruction, the number of truncated modes r directly dictates the prediction performance. This study presents the reconstructed instantaneous velocity fields and absolute error distributions at transient moment (snapshot 4375 in the test set) for $r = 10$ and $r = 5$.

As shown in figure 1, with $r = 10$, the traditional QR+POD method roughly recovers the macroscopic flow structure, yielding a relative L_2 error of 6.53%. However, our proposed joint optimization framework $\mathrm{GNN_{sel}}$+$\mathrm{GNN_{rec}}$ further reduces this error to 3.57%. The absolute error contour demonstrates effective error suppression in the wake evolution region. Observing the spatial sensor distribution, the QR algorithm tends to cluster sensors locally in the core vortex evolution region directly behind the cylinder to ensure linear independence of the column space of the reconstruction matrix. In contrast, the sensors optimized by $\mathrm{GNN_{sel}}$ distribute along the shear layer separation lines and the main wake envelope. This broader spatial coverage helps the reconstruction network capture global vortex dynamics boundaries. Furthermore, two cross-decoupled experiments were evaluated. Specifically, traditional QR sensors were used to drive the deep reconstructor, and GNN-optimized sensors were used to drive the linear POD reconstructor. In both configurations, the reconstruction errors surged to 21.30%

*Corresponding Author, Jianqin Zhu zhujianqin@buaa.edu.cn

and 21.69%, respectively, revealing severe local distortion. This indicates that the sensor distribution must be highly aligned with its corresponding reconstruction prediction algorithm. Isolated sensor optimization or pure black-box reconstruction generates significant bias, confirming the necessity of the joint optimization design.

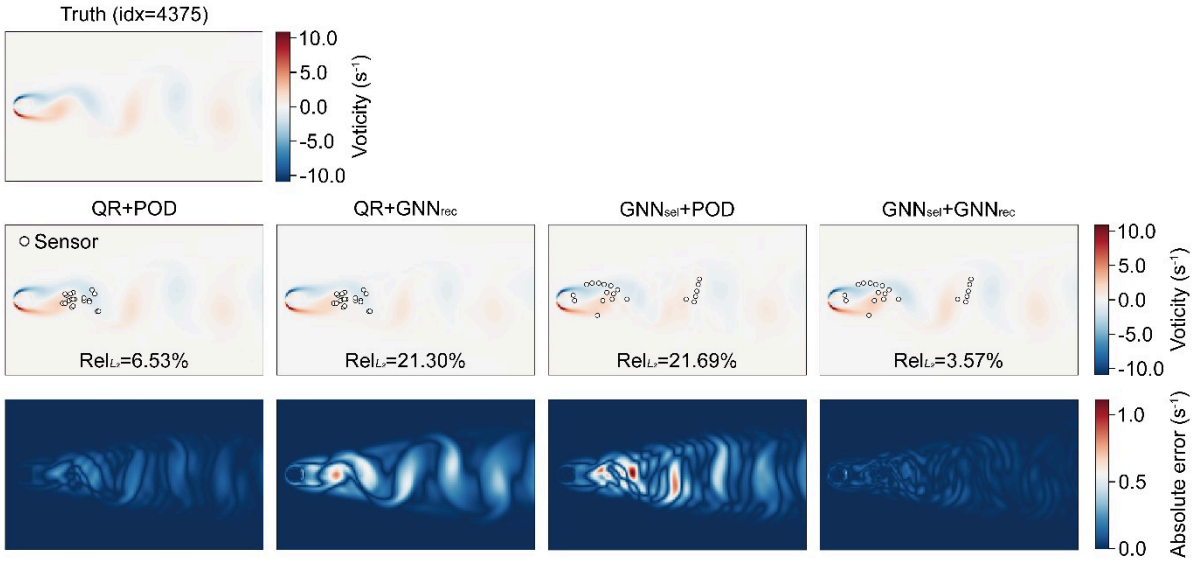


Figure 1: Comparison of the reconstructed vorticity fields and absolute error distributions (,). White circles indicate the spatial layout of the 20 discrete sensors; QR and GNN_{sel} denote the sensor selection strategies, while POD and GNN_{rec} represent the flow field reconstruction methods.

$r = 5$

As illustrated in figure 2, when the physical prior is severely compressed to , the first 5 modes cannot adequately contain the high-frequency vortex shedding information in the wake. Consequently, the error of QR+POD climbs to 10.42%, exhibiting distinct stripe modal omission features. By contrast, GNN_{sel}+GNN_{rec} maintains the error at 8.40%. This shows that the residual GNN not only efficiently utilizes low-fidelity physical priors but also leverages its nonlinear mapping capability to partially compensate for missing flow dynamics unresolved by the low rank POD basis. Regarding sensor distribution with fewer modes, the array learned by GNN_{sel} still maintains a layout tracking the shear layer and wake trajectory. This likely provides stronger informational representation for fluid dynamics feature extraction compared to traditional dense sensor placement.

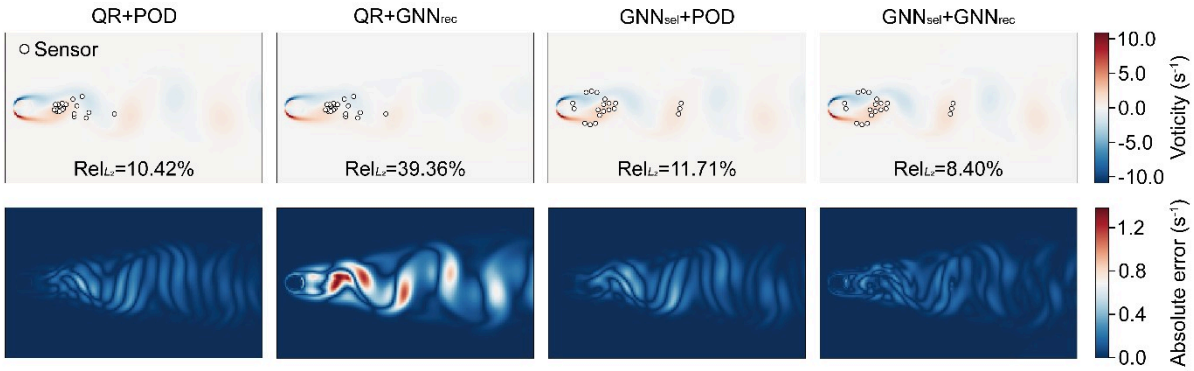


Figure 2: Comparison of the reconstructed vorticity fields and absolute error distributions under extreme modal truncation ($r = 5$, $p = 20$). White circles indicate the spatial layout of the 20 discrete sensors; QR and GNN_{sel} denote the sensor selection strategies, while POD and GNN_{rec} represent the flow field reconstruction methods.

3.2 Robustness Analysis Under Measurement Noise

*Corresponding Author, Jianqin Zhu zhujianqin@buaa.edu.cn

In practical engineering applications, signals acquired by sparse sensors are inevitably contaminated by measurement noise. To objectively evaluate the model's anti-interference capability under noisy inputs, we conducted 100 Monte Carlo sampling evaluations for all transient moments in the test set under Gaussian white noise levels ranging from 1% to 20%. Figure 3 displays the average relative L_2 errors and their standard deviations for different methods across various noise levels. Given the severe performance degradation of the cross-decoupled configurations demonstrated earlier, this robustness evaluation focuses exclusively on the baseline QR+POD and the proposed GNN_{sel}+GNN_{rec} framework.

As shown in figure 3(a), the traditional QR+POD method exhibits greater sensitivity to noise. This accuracy drop becomes especially pronounced when retaining more modes. Specifically, as the noise level increases from 1% to 20%, the mean reconstruction error for r=5 grows by only 1.13%, from 13.53% to 14.66%. In contrast, the error for r=10 degrades much faster, exhibiting a 2.99% surge, from 6.95% to 9.94%. This error amplification effect stems primarily from the inherent sensitivity of the linear basis inverse solver to random noise. Comparatively, the proposed GNN_{sel}+GNN_{rec} exhibits stronger robustness under both modal truncations. Its error curves remain relatively flat across the entire noise spectrum, stabilizing around 5% for r=10 and 10% for r=5. Further analysis of the error standard deviations in figure 3(b) reveals that, for both modal configurations, the reconstruction standard deviations of both methods rise with increasing noise levels. Moreover, for both methods, the standard deviation at r=10 is consistently higher than at r=5. This indicates that in solving inverse problems, introducing more high-order modes exacerbates the ill-posedness of the system, leading to a surge in uncertainty for some local snapshots. Additionally, the absolute standard deviation values of our GNN_{sel}+GNN_{rec} across both modes are consistently lower than those of the QR+POD method, with slower growth trend. This also demonstrates that the relatively dispersed sensor combinations extracted by GNN_{sel} possess better anti-noise capabilities against local measurement noise.

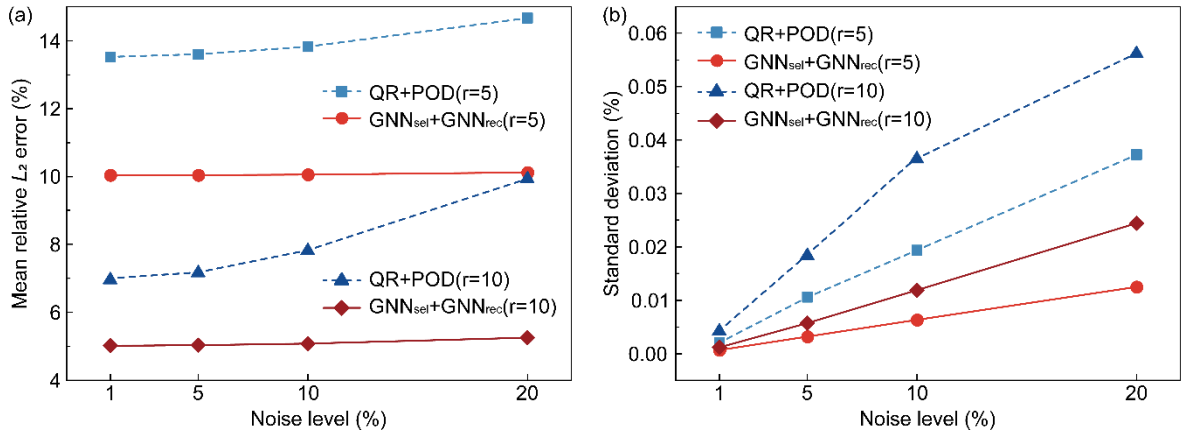


Figure 3: Robustness analysis across noise levels ranging from 1% to 20%. (a) Comparison of mean relative L_2 error. (b) Standard deviation of the relative L_2 errors. The values in (a) and (b) represent the mean relative L_2 errors and one standard deviation, respectively, obtained from 100 Monte Carlo sampling evaluations.

4. Conclusion

To address the challenge of high-fidelity flow reconstruction under sparse observations, this paper proposes a physics-informed GNN framework. By injecting singular value weighted POD modal features and integrating differentiable continuous probability relaxation sampling, this framework successfully achieves joint optimization of sensor placement and global flow reconstruction. Evaluations show that the proposed strategy effectively suppresses error amplification caused by decoupled optimization. It leverages nonlinear mapping to partially compensate for truncated high-frequency flow features. Regarding spatial layout, the network's autonomously optimized sensors track the shear layer separation lines and the main wake envelope, exhibiting superior efficiency in capturing fluid dynamics features. Furthermore, for the evaluated cylinder wake at Re=100, our framework

*Corresponding Author, Jianqin Zhu zhujianqin@buaa.edu.cn

effectively mitigates error amplification under strong measurement noise ranging from 1% to 20%, demonstrating reliable noise robustness and statistical stability. While this study provides a reference approach for sparse flow reconstruction, future work will focus on extending and validating this joint optimization framework across more complex, high Reynolds number turbulent flows.

Acknowledgement
This work was supported by the National Natural Science Foundation of China (U2541279).

References

[1] Kapteyn M G, Pretorius J V R, Willcox K E. ‘A probabilistic graphical model foundation for enabling predictive digital twins at scale’[J]. Nature Computational Science, 2021, 1(5): 337-347.

[2] Brunton S L, Noack B R, Koumoutsakos P. ‘Machine learning for fluid mechanics’[J]. Annual review of fluid mechanics, 2020, 52(1): 477-508.

[3] Manohar K, Brunton B W, Kutz J N, et al. ‘Data-driven sparse sensor placement for reconstruction: Demonstrating the benefits of exploiting known patterns’[J]. IEEE Control Systems Magazine, 2018, 38(3): 63-86.

[4] Fukami K, Fukagata K, Taira K. ‘Super-resolution analysis via machine learning: a survey for fluid flows’[J]. Theoretical and Computational Fluid Dynamics, 2023, 37(4): 421-444.

[5] Fukami K, Maulik R, Ramachandra N, et al. ‘Global field reconstruction from sparse sensors with Voronoi tessellation-assisted deep learning’[J]. Nature Machine Intelligence, 2021, 3(11): 945-951.

[6] Yang F, Kong C, Liu J, et al. ‘Deep reinforcement learning-based optimization of sparse sensor placement for supersonic isolator flowfield monitoring’[J]. Aerospace Science and Technology, 2025: 111307.

[7] Karniadakis G E, Kevrekidis I G, Lu L, et al. ‘Physics-informed machine learning’[J]. Nature Reviews Physics, 2021, 3(6): 422-440.

[8] Jang E, Gu S, Poole B. ‘Categorical reparameterization with gumbel-softmax’[J]. arXiv preprint arXiv:1611.01144, 2016.

[9] Berkooz G, Holmes P, Lumley J L. ‘The proper orthogonal decomposition in the analysis of turbulent flows’[J]. Annual review of fluid mechanics, 1993, 25(1): 539-575.

*Corresponding Author, Jianqin Zhu zhujianqin@buaa.edu.cn

Proceedings of the 2nd international Symposium on AI and Fluid Mechanics
Paper No []

Advection-Aware Graph Neural Networks with Efficient Graph Transformers for Airfoil Flow-Field Prediction

[1]Yimin Zhang; [1]Zi-Xiang Tong*; [1]Shuzhi Ni; [1]Jianqin Zhu;

[1]*School of Energy and Power Engineering, Beihang University, Beijing, 100191, China*

Abstract

Accurate and efficient prediction of aerodynamic flow fields around airfoils is essential for rapid aerodynamic analysis and data-driven design optimization. However, conventional neural surrogate models often struggle to simultaneously capture local transport mechanisms and long-range aerodynamic dependencies on irregular CFD meshes. To address this issue, this study proposes an advection-aware graph neural network enhanced by an efficient Galerkin Transformer for airfoil flow-field prediction. The CFD mesh is represented as a geometric graph, where node features encode spatial coordinates, freestream conditions, signed distance functions and surface-normal information, while edge features describe relative geometric relations between neighboring points. Advection-aware message-passing layers are designed to model local directional information transport through edge-dependent modulation. To overcome the limited receptive field of local graph aggregation, a Galerkin Transformer layer is further introduced to enable global information exchange with linear complexity. The proposed framework is evaluated on the AirfRANS dataset for predicting pressure fields and aerodynamic coefficients over unstructured airfoil meshes. Results show that the Advection GNN–Galerkin Transformer achieves the best overall accuracy, significantly reducing volume-field, surface-field and lift-coefficient errors compared with baseline models. This demonstrates the effectiveness of combining physics-inspired local propagation with efficient global attention for aerodynamic surrogate modeling.

Keywords: Advection-Aware Graph Neural Network; Graph Transformer; Galerkin Attention; Airfoil Flow Prediction;

1.Introduction

Accurate and efficient prediction of aerodynamic flow fields around airfoils is a fundamental task in computational fluid dynamics and aerodynamic design. The distributions of velocity, pressure and turbulent viscosity directly determine aerodynamic quantities such as lift and drag, and therefore play a crucial role in performance evaluation and design optimization. High-fidelity Reynolds-averaged Navier–Stokes simulations can provide detailed flow-field information, but they are still computationally expensive when a large number of airfoil geometries and operating conditions need to be evaluated. To reduce the computational cost, deep learning surrogate models have been increasingly investigated for fast approximation of RANS solutions around airfoils [1]. In this context, the AirfRANS dataset provides a representative benchmark for data-driven aerodynamic prediction, containing high-fidelity RANS solutions over two-dimensional airfoils with varying geometries, angles of attack and inlet velocities [2].

For flow-field prediction on irregular CFD meshes, graph neural networks provide a natural modeling framework. By representing mesh points as nodes and local spatial relations as edges, GNNs can directly operate on unstructured meshes without interpolating the flow field onto a regular grid. Mesh-based graph networks have demonstrated strong capability in learning simulation dynamics on complex discretized domains [3], and message-passing neural PDE solvers further show that local graph aggregation can be used to approximate physical interactions in partial differential equations [4]. Representative graph architectures, such as GraphSAGE [5] and Graph U-Net [6], have also provided useful baselines for learning on graph-structured data. However, conventional message-passing GNNs mainly rely on local neighborhood aggregation. Their receptive field is limited by the number of message-passing layers, which may be insufficient for aerodynamic flows where pressure redistribution, wake evolution and boundary-layer effects are influenced by long-range interactions across the entire computational domain.

Transformer-based architectures offer a promising solution for modeling such global dependencies. Recent studies have applied graph Transformers to inverse physics and flow-field reconstruction around arbitrary airfoils [7], while learning-based reconstruction methods with sensor placement have also shown the potential of neural networks to infer complete physical fields from limited observations [8]. Nevertheless, standard self-attention requires constructing a dense node-to-node attention matrix, leading to quadratic computational complexity with respect to the number of mesh points. This becomes expensive for dense CFD point clouds. To improve efficiency, Galerkin attention reformulates attention from an operator-learning perspective and enables global information exchange with linear

*Corresponding Author, Zi-Xiang Tong zxtong@buaa.edu.cn

complexity under fixed latent dimensions [9]. More recently, Transformer-based PDE solvers on general geometries have further demonstrated the effectiveness of attention mechanisms for physical field prediction on complex domains [10].

Motivated by these developments, this work proposes an advection-aware graph neural network enhanced by an efficient Galerkin Transformer for airfoil flow-field prediction on unstructured meshes. The proposed model first represents each CFD case as an attributed graph, where node features include spatial coordinates, freestream velocity components, signed distance to the airfoil surface and surface-normal information, while edge features encode relative displacement and distance between neighboring points. To better capture the directional transport behavior in aerodynamic flows, advection-aware message-passing layers are designed with an edge-dependent modulation mechanism. This allows local information propagation to be guided by geometric directionality, providing a learnable graph analogue of the advection operator. After local physical representations are extracted, a Galerkin Transformer layer is introduced to perform efficient global information exchange and capture long-range aerodynamic dependencies.

The proposed framework is evaluated on the AirfRANS dataset for full-field aerodynamic prediction over two-dimensional airfoils. Three model variants are compared: a pure Advection GNN, an Advection GNN followed by a Galerkin Transformer, and a Galerkin Transformer followed by an Advection GNN. The results are further compared with representative baseline models, including Simple MLP, GraphSAGE [5] and Graph U-Net [6]. Quantitative results show that applying the Galerkin Transformer after the advection-aware graph layers achieves the best overall performance in volume pressure prediction, surface pressure prediction and lift-coefficient estimation. These results indicate that local advection-aware message passing and global Galerkin attention are complementary: the former captures geometry-dependent directional transport, while the latter improves long-range aerodynamic coupling.

2.Method

2.1 Graph Construction

The input of each sample is represented as an attributed graph , where each node corresponds to a CFD mesh point and each edge denotes a local spatial relation. For a case with mesh points, the node feature matrix is denoted by . Each node contains its spatial coordinates, freestream velocity components, signed distance to the airfoil surface and local surface normal information,

where (is the point coordinate, () is the inlet velocity vector broadcast to all nodes of the same case, is the signed distance function, and is the wall-normal vector, set to zero for non-surface points.

A radius graph is constructed from the point coordinates. For each node, neighbour points within a radius are connected, with a maximum number of neighbors . In the experiments, each training case is randomly subsampled to a fixed number of nodes to control memory usage. For an edge from node to node , the edge feature is defined by relative geometry,

This edge descriptor preserves the relative direction and distance from the sender node to the receiver node.

The target physical field at node is

including the two velocity components, pressure and turbulent viscosity.

2.2 Advection-aware message passing

In fluid dynamics, the advection operator

*Corresponding Author, Zi-Xiang Tong zxtong@buaa.edu.cn

describes the directional transport of a physical quantity along the local velocity . Structurally, this operator consists of a directional transport coefficient and a spatial variation term. Inspired by this form, we design a graph message-passing layer in which each edge message is decomposed into a learned directional modulation term and a learned state-interaction term.

For a directed edge , let and denote the latent states of the sender and receiver nodes, and let denote the geometric edge feature. The directional modulation term is computed as

where plays a role analogous to the directional projection coefficient in the continuous advection operator. The state-interaction term is computed as

which represents the learnable counterpart of the state variation along the edge. The final edge message is generated by

where denotes element-wise multiplication. The incoming messages are then aggregated at node :

Finally, the node state is updated by a residual mapping:

After advection-aware message-passing layers, the local node representation is obtained as

This formulation is not an explicit finite-difference discretization of the advection equation; rather, it provides a learnable graph analogue of the advection operator. The term controls the direction and intensity of edge-wise information transport, while encodes the state interaction between neighboring nodes.

2.3 Graph Transformer

Conventional GNN layers rely on local neighborhood aggregation, and their receptive field is therefore limited by the number of message-passing layers. This makes it difficult to capture long-range aerodynamic dependencies such as pressure redistribution, wake development, and far-field effects. To address this limitation, a graph Transformer layer is introduced before or after the local advection-aware message-passing blocks to enable global information exchange among nodes. In this work, we mainly consider the structure in which the graph Transformer is applied after the local advection-aware message-passing layers.

Directly applying standard self-attention to H^{loc} requires constructing an N×N attention matrix, which leads to quadratic complexity with respect to the number of nodes. This is computationally expensive for dense CFD point clouds. Therefore, we adopt Galerkin attention as an efficient global information exchange module.

The query, key, and value matrices are computed by linear projections:

The global attention output is then formulated as

Z=

*Corresponding Author, Zi-Xiang Tong zxtong@buaa.edu.cn

with appropriate normalization. This formulation has linear complexity in the number of nodes for fixed latent dimension and is therefore suitable for dense CFD graphs. The globally enhanced representation is then obtained by a residual update:

Finally, the physical field is predicted by a pointwise decoder:

where is an MLP decoder and denotes the -th row of . The model is trained by minimizing the mean-squared error between predicted and simulated physical fields. When surface and volume regions are distinguished, a weighted loss can be used:

where and denote errors on non-surface and surface points, respectively. This encourages the model to maintain accuracy both in the far-field flow and near the airfoil boundary, where aerodynamic quantities are most sensitive.

3. Results

To quantitatively evaluate the effectiveness of the proposed advection-aware graph neural architecture, three variants are trained and compared with representative baseline models, including Simple MLP, GraphSAGE and Graph U-Net. The three proposed variants are: the pure Advection GNN, the Advection GNN followed by a Galerkin Transformer, and the Galerkin Transformer followed by an Advection GNN. The comparison is conducted on unstructured AirfRANS meshes for full-field aerodynamic prediction. The evaluation metrics include the relative error of the volume pressure field, the relative error of the surface pressure field, the relative error of the lift coefficient, and the Spearman's rank correlation of the lift coefficient. Lower relative errors indicate higher prediction accuracy, while a Spearman's correlation closer to 1 suggests a better ability to preserve the aerodynamic performance ranking among different airfoil cases. The quantitative results are summarized in Table 1

Table 1: Quantitative Performance Comparison of Baseline Models and Proposed Advection-Aware Graph Transformer Architectures on Airfoil Flow-Field Prediction. In addition to the relative L2 of the surrounding (Volume) and surface (Surf) physics fields, the relative L_2 lift coefficient (C_L) is also recorded, along with a Spearman's rank correlations ρ_L. A Spearman's correlation close to 1 indicates better performance.

Model	Volume: $\bar{p}\downarrow$	Surf: $\bar{p}\downarrow$	$C_L\downarrow$	$\rho_L\uparrow$
Simple MLP	0.0081	0.0200	0.2108	0.9932
GraphSAGE	0.0083	0.0179	0.1511	0.9953
Graph U-Net	0.0076	0.0146	0.1843	0.9922
Advection GNN	0.0073	0.0184	0.1216	0.9970
Advection GNN-Galerkin Transformer	**0.0045**	**0.0112**	**0.1067**	0.9967
Galerkin Transformer - Advection GNN	0.0066	0.0159	0.1139	**0.9971**

As shown in Table 1, the proposed hybrid architectures improve most quantitative error metrics compared with the baseline models. The pure Advection GNN achieves a lower lift-coefficient error than Simple MLP, GraphSAGE and Graph U-Net, indicating that advection-aware message passing is beneficial for aerodynamic coefficient estimation. However, its surface pressure error is not consistently better than all graph-based baselines, suggesting that local directional propagation alone is insufficient for accurate full-field prediction. By introducing the Galerkin Transformer

*Corresponding Author, Zi-Xiang Tong zxtong@buaa.edu.cn

after the Advection GNN, the model achieves the lowest errors in volume pressure, surface pressure and lift coefficient, with values of 0.0045, 0.0112 and 0.1067, respectively. Compared with the pure Advection GNN, this corresponds to reductions of approximately 38.4%, 39.1% and 12.3%. Compared with the best baseline results, the reductions are approximately 40.8%, 23.3% and 29.4%, respectively. These results demonstrate the complementary roles of local advection-aware propagation and global Galerkin attention. The Galerkin Transformer–Advection GNN variant obtains the highest Spearman's rank correlation, but its error metrics are inferior to those of the Advection GNN–Galerkin Transformer. Therefore, applying global attention after local physics-aware message passing appears to be the more effective architecture for airfoil flow-field prediction on unstructured meshes.

4.Conclusion

This study develops an advection-aware graph neural network enhanced by an efficient Galerkin Transformer for airfoil flow-field prediction on unstructured CFD meshes. The proposed model combines local physics-inspired message passing with global information exchange. The advection-aware graph layers capture geometry-dependent directional transport through edge-dependent modulation, while the Galerkin Transformer improves the modeling of long-range aerodynamic dependencies with reduced computational complexity. Tests on the AirfRANS dataset demonstrate that the proposed framework outperforms representative baseline models, including Simple MLP, GraphSAGE and Graph U-Net. Among the tested variants, the Advection GNN–Galerkin Transformer achieves the best overall performance in volume pressure prediction, surface pressure prediction and lift-coefficient estimation. These results suggest that applying global attention after local advection-aware message passing is more effective than using the Transformer before local graph propagation. The proposed method provides an accurate and efficient neural surrogate modeling framework for aerodynamic prediction on irregular meshes. Future work will extend the approach to three-dimensional aerodynamic configurations, stronger physical constraints and broader flow conditions.

Acknowledgement

This work was supported by the National Natural Science Foundation of China (U2541279).

References

1. N. Thuerey, K. Weißenow, L. Prantl and X. Hu 'Deep Learning Methods for Reynolds-Averaged Navier–Stokes Simulations of Airfoil Flows', AIAA Journal, 58(1), pp. 25–36, 2020, doi: 10.2514/1.J058291.

2. F. Bonnet, J. Mazari, P. Cinnella and P. Gallinari 'AirfRANS: High Fidelity Computational Fluid Dynamics Dataset for Approximating Reynolds-Averaged Navier–Stokes Solutions', Advances in Neural Information Processing Systems 35, NeurIPS 2022, Datasets and Benchmarks Track, New Orleans, USA, 28 November–9 December 2022.

3. T. Pfaff, M. Fortunato, A. Sanchez-Gonzalez and P. W. Battaglia 'Learning Mesh-Based Simulation with Graph Networks', International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, 3–7 May 2021.

4. J. Brandstetter, D. E. Worrall and M. Welling 'Message Passing Neural PDE Solvers', International Conference on Learning Representations, ICLR 2022, Virtual Event, 25–29 April 2022.

5. W. L. Hamilton, Z. Ying and J. Leskovec 'Inductive Representation Learning on Large Graphs', Advances in Neural Information Processing Systems 30, NIPS 2017, Long Beach, USA, 4–9 December 2017, pp. 1024–1034.

6. H. Gao and S. Ji 'Graph U-Nets', Proceedings of the 36th International Conference on Machine Learning, ICML 2019, Long Beach, USA, 9–15 June 2019, PMLR 97, pp. 2083–2092.

7. G. Duthé, I. Abdallah and E. Chatzi 'Graph Transformers for Inverse Physics: Reconstructing Flows Around Arbitrary Two-Dimensional Airfoils', AIAA Journal, pp. 1–15, 2026, doi: 10.2514/1.J065387.

*Corresponding Author, Zi-Xiang Tong zxtong@buaa.edu.cn

8. R. Li, G. Wan, Z. Huang, Z. Liu, H. Wang, X. Luo, W. Wang and Y. Sun ‘Flow Field Reconstruction with Sensor Placement Policy Learning’, Advances in Neural Information Processing Systems 38, NeurIPS 2025, Main Conference Track, 2025.

9. S. Cao ‘Choose a Transformer: Fourier or Galerkin’, Advances in Neural Information Processing Systems 34, NeurIPS 2021, Virtual Conference, 6–14 December 2021, pp. 24924–24940.

10. H. Wu, H. Luo, H. Wang, J. Wang and M. Long ‘Transolver: A Fast Transformer Solver for PDEs on General Geometries’, Proceedings of the 41st International Conference on Machine Learning, ICML 2024, Vienna, Austria, 21–27 July 2024, PMLR 235, pp. 53681–53705.

*Corresponding Author, Zi-Xiang Tong zxtong@buaa.edu.cn

Learning viscoelastic constitutive models from mesoscopic simulations using scientific machine learning

[1]Vipin Michael*; [1]Bharath Ravikumar; [1]Ioannis Karathanasis; [1]Mehdi Seddiq; [1,2]Manolis Gavaises.

[1]*City St George's University of London, Northampton Square, London EC1V 0HB*

[2]*OVGU, Magdeburg, Germany*

Abstract

Viscoelastic fluids arise in a wide range of natural and industrial processes, yet the development of constitutive models capable of accurately describing their behaviour across diverse flow conditions remains a longstanding scientific challenge. Simulating the fluid rheology at a mesoscopic scale offers distinct advantages over classical continuum scale models as it allows the viscoelastic stress tensor to emerge naturally from the interactions of the coarse-grained molecules that govern the rheological response. However, their computational cost limits their direct application to engineering-scale problems. In this work, we present a scientific machine learning framework for learning continuum-scale constitutive models directly from rheological data generated by mesoscopic simulations. The approach combines universal differential equations and differentiable physics to augment conventional constitutive models through data-driven discovery of unresolved rheological behaviour. Rheological datasets generated using dissipative particle dynamics (DPD) simulations are used to train the models while preserving the mathematical structure and physical constraints required for continuum-scale flow prediction. The resulting constitutive models can be seamlessly integrated into computational fluid dynamics (CFD) solvers, providing a pathway for bridging mesoscopic simulations and engineering-scale predictions of complex viscoelastic flows.

Keywords: universal differential equations, mesoscopic simulations, rheology

Proceedings of the 2nd international Symposium on AI and Fluid Mechanics
Paper No

Drifting Models for Surrogate Flow Modeling

[1]Chris R. Jung[†]; [1]Markus Dörr[†]; [2]Natalie Jüngling; [2]Jennifer Niessner; [1]Adam T. Müller[‡*]; [1]Nicolaj C. Stache[‡]
[1]*Center for Machine Learning (ZML)*
[2]*Institute for Flow in Additively Manufactured Porous Structures (ISAPS)*
Heilbronn University of Applied Sciences, 74081 Heilbronn, Germany

Abstract
While Computational Fluid Dynamics (CFD) provides high-fidelity flow fields for optimizing indoor environments, its computational cost limits rapid exploration. To solve this problem generative surrogates offer better distribution modeling than deterministic networks, but iterative sampling is slow. To enable high-quality, single-pass generation, we adapt the novel generative drifting framework to fluid mechanics. We introduce a conditional architecture that performs drifting in a learned VAE latent space and uses label-aware masking to align generated samples with their boundary conditions. Our label-conditioned model matches iterative diffusion in accuracy and flow consistency while running two orders of magnitude faster. Additionally, we propose a spatial-conditioning variant that establishes a promising path towards generalization to unseen geometries. Ultimately, conditional drifting serves as a highly efficient alternative to diffusion based approaches, unlocking real-time CFD surrogates where inference speed is critical.
Keywords: *drifting models; generative surrogate modeling; indoor flow prediction; data-driven fluid dynamics*

1. Introduction
Indoor air quality (IAQ) is critical for human health, since humans typically spend most of their time in indoor environments [1]. Ventilation (e.g. window ventilation or mechanical ventilation) effectively decreases the concentration of all indoor contaminants. Designing them efficiently requires an accurate understanding of complex indoor airflow patterns, which are highly sensitive to room geometry, inlet locations, and flow velocities [2, 3].

While Computational Fluid Dynamics (CFD) provides the high-fidelity flow fields necessary to optimize such environments [4], its notable computational cost precludes rapid design exploration and real-time ventilation control [5, 6]. This creates a need for data-driven surrogate models capable of delivering reliable predictions with reduced computational demands [7].

While deterministic architectures like Fourier Neural Operators (FNOs) [8] and coordinate-based networks [9] offer rapid inference, capturing environments with complex multimodal flow distributions remains challenging. Generative approaches, such as diffusion models, successfully capture this physical complexity. However, their iterative nature makes inference computationally heavy [10].

In this work, we explore the potential of adapting novel generative modeling via drifting [11] to the domain of fluid mechanics. Originally designed to evolve pushforward distributions during training to enable high-quality, one-step generation without iterative inference, we demonstrate an adapted conditional drifting architecture on 2D indoor flow modeling as a foundational proof-of-concept. Our main contributions are as follows: **(1)** We prove that conditional drifting models can accurately and efficiently generate flow fields in a single step. **(2)** We introduce essential modifications for physical conditioning, namely latent-space drifting via a learned Variational Autoencoder (VAE) and label-aware positive masking to enforce boundary conditions. **(3)** We develop a spatial encoding variant to explore generalization across unseen room geometries.

2. Related Work
Data-Driven Surrogates for Flow Prediction. Deep learning architectures, including convolutional neural networks (e.g., U-Nets) and FNOs [9, 12], are increasingly used as surrogates of traditional CFD solvers [8, 9]. While deterministic surrogates can offer dramatic inference speedups and establish strong global flow priors, they are mostly designed to minimize standard regression losses like Mean Squared Error. Consequently, such surrogates tend to predict a smoothed expectation of the flow. This can fail to capture the complex, high-frequency, and multimodal distributions characteristic of indoor environments, where minor variations in boundary conditions or obstacles dictate drastically different, yet equally valid, flow regimes [2].
Generative Modeling for Physical Dynamics. To address this smoothing and enable uncertainty-aware predictions, generative architectures have gained traction. Diffusion model based approaches have been

[†] Equal contribution, [‡] Equal senior contribution
*Corresponding Author, Adam T. Müller, adam-theo.mueller@hs-heilbronn.de

established as surrogates for aerodynamic simulations, successfully capturing complex, physically plausible flows [10]. Advanced autoregressive variants extend these capabilities to temporal predictions [13]. However, diffusion relies on an iterative denoising process. The resulting hundreds of neural function evaluations (NFE) render generation computationally heavy, contradicting the goal of rapid surrogate modeling. Flow matching approaches reduce this burden by learning continuous-time conditional flows, requiring fewer sampling steps while extending to unstructured geometries [14, 15]. Nevertheless, flow matching still relies on numerically integrating an ordinary differential equation during inference [16]. To bypass such iterative sampling, we build upon the recently introduced framework of generative modeling via drifting [11]. Originally designed for image generation and grounded in optimal transport [17], drifting evolves pushforward distributions during training. By enabling high-quality, one-step generation (NFE $= 1$), it provides a promising foundation for rapid fluid surrogates.

3. Method

3.1. Fundamentals of Drifting Models.

A drifting model [11] is a single-pass network $f : \mathbb{R}^C \to \mathbb{R}^D$ that maps a prior sample $\epsilon \sim p_\epsilon$ to an output $x = f(\epsilon)$, inducing a pushforward distribution $q = f_{\#}p_\epsilon$ that is trained to match the data distribution p. Rather than learning an iterative noise-to-data trajectory as in diffusion or flow matching, drifting models evolve q at training time through a drifting field $V_{p,q}(x)$ that prescribes, at each iteration, the direction in which a generated sample should move so that q approaches the data distribution p. The field is constructed to satisfy $V_{p,q}(x) = 0$ whenever $q = p$, so equilibrium coincides with distribution matching. Training thus reduces to a fixed-point update $f_{\theta_{i+1}}(\epsilon) \leftarrow f_{\theta_i}(\epsilon) + V_{p,q_{\theta_i}}(f_{\theta_i}(\epsilon))$, realised as the stop-gradient loss:

$$\mathcal{L} = \mathbb{E}_\epsilon\big[\big\| f_\theta(\epsilon) - \mathrm{stopgrad}\big(f_\theta(\epsilon) + V_{p,q_\theta}(f_\theta(\epsilon))\big)\big\|^2\big]. \tag{1}$$

In practice $V_{p,q}$ is instantiated as a kernel-based attraction–repulsion field driven by positive samples drawn from the data and negative samples from q, with the negative samples maintained in a memory bank to amortise their cost across iterations. As f_θ is a single-pass network, inference is one-step (NFE=1), in contrast to the multi-step solvers required by diffusion and flow-based baselines.

3.2. Adjustments for Surrogate Modelling

We adapt the original drifting model along three axes that reflect the different demands of CFD surrogate modelling compared to natural-image generation: the operating space, the conditioning interface, and the construction of the drifting field.

Latent-space drifting via a learned VAE. Unlike the original work using an ImageNet-pretrained MAE, we trained a convolutional VAE on our simulation data to ensure a domain-specific feature extractor. The VAE maps 64×64 velocity fields to a $4 \times 4 \times 16$ latent representation $z = E(\mathbf{u})$ with encoder E. By performing drifting within this compact, semantically organized latent space and reconstructing via the decoder D, we maintain structural integrity while avoiding suboptimal pre-trained features.

Conditioning. Whereas the original generator is conditioned on a single class label, each of our samples carries a condition comprising the inlet position, the outlet position, the room geometry, and a scalar inlet velocity magnitude $v_{\mathrm{in}} \in \mathbb{R}$. We investigate two encoder variants that differ in how the geometric information is presented to the network:

- **Label-based.** The *label-based* variant treats inlet and outlet as discrete identifiers (inlet_id, outlet_id) $\in \mathbb{N}$, drawn from a fixed catalogue of room configurations and embeds each through a learned embedding table. The room geometry is implicit in the identifier. The label-based variant is restricted to the configurations enumerated in its embedding table.
- **Spatial.** The *spatial* variant replaces these identifiers with a three-channel binary mask M with $3 \times H \times W$ that explicitly encodes the inlet, outlet, and obstacle locations on the simulation grid, and processes it through a small convolutional encoder that produces a fixed-dimensional vector. The spatial variant is more flexible than the label-based variant and can theoretically generalise better to arbitrary room layouts, including geometries unseen during training.

In both cases v_{in} is embedded by a small Multilayer Perceptron (MLP) and summed with the geometry embedding, where the resulting conditioning vector c drives the adapted Diffusion Transformer (DiT) backbone [18].

Label-aware positive masking. To adapt the kernelized drifting formulation [11] to our data-sparse setting, we introduced a binary compatibility mask. Unlike the original work [11], which assumes large pools of ground-truth samples per class and processes them in uniform batches, our approach allows mixed conditions within a batch. We gate the attraction kernel by matching $(\text{inlet}, \text{outlet}, \text{geometry})$ triples, ensuring generated samples only drift toward compatible positives. While the global repulsion term remains unchanged, this masking prevents the attraction term from averaging across incompatible flow regimes, which would otherwise force samples toward a condition-agnostic mean.

3.3. Latent Diffusion Baseline.

We adopted a Latent Diffusion Model baseline [19] inspired by Liu and Thuerey's state-of-the-art work in RANS-based surrogates [20]. Operating in the same 4×4 latent space as our drifting model, we used a lightweight DiT with a 1,000-step cosine schedule. To adapt the model for physical surrogates, we integrated heterogeneous boundary conditions and inflow parameters via cross-attention layers. Inference utilized DDIM sampling.

4. Experiments

4.1. Experimental Setup

Training data comprised 2,025 two-dimensional CFD simulations using a steady-state solver executed in Simcenter STAR-CCM+ using a RANS k-ϵ turbulence model for incompressible air flow. The computational domain consisted of a square room measuring 1.5 m × 1.5 m. In each simulation, one 0.1 m wide inlet and one 0.1 m wide outlet were active, located on opposite walls. We evaluated three distinct geometric configurations: an empty room without obstacles, and two configurations containing an obstacle with a diameter of 0.25 m. This obstacle was positioned either squarely in the center of the domain or offset laterally.

For each geometry, we systematically swept 15 inlet positions, 15 outlet positions, and three constant inlet velocities (0.1, 0.2, and 0.3 m/s), resulting in 675 simulations per configuration. The outlet was defined as an atmospheric pressure outlet. The domain was discretized using a polyhedral mesh, with near-wall prism layers applied around the obstacle, while all walls were treated as no-slip wall boundaries. Achieving convergence typically required 750-800 seconds of CPU time per simulation. Following convergence to a steady state, the final flow fields were averaged over the last 100 iterations to mitigate residual numerical fluctuations. Finally, these high-fidelity velocity vectors were interpolated onto a 64×64 uniform grid, with a randomly sampled 10% of the complete dataset reserved as the held-out test set.

4.2. Metrics

We evaluate the predicted velocity field $\hat{\mathbf{u}} = (\hat{u}, \hat{v})$ against the ground truth $\mathbf{u} = (u, v)$ along two axes: field accuracy and flow-structure consistency. All metrics are computed per sample and reported as mean $\pm$ standard deviation over the test set.

Field accuracy. Following established relative-RMSE metrics in CFD-surrogate evaluation [21], we report the range-normalised RMSE (nRMSE), in which the per-sample RMSE is normalised by the target field's range. As a scale-free complement we report the coefficient of determination (R^2), a standard measure of explained variance in indoor-airflow surrogates [22, 23]. To capture directional agreement independently of magnitude, we additionally report the cosine similarity between predicted and target vector fields over locations with non-negligible target velocity, which is sensitive to directional disagreement that nRMSE penalises only indirectly [23].

Flow-structure consistency. To probe whether predictions preserve the structural properties of the reference fields, we report two differential diagnostics. We monitor the vorticity as structural quantity relative to the ground truth [12, 24], to detect over- or under-smoothing of rotational structure. As a complementary diagnostic we monitor the divergence relative to the ground truth; while divergence is often used as a structural constraint in physics-informed learning [25], we employ it as a passive diagnostic against the target divergence. For both quantities we report the gap between predicted and target magnitudes and the predicted-to-target ratio as a measure of relative structural fidelity.

4.3. Results

4.3.1. Quantitative

Table 1 reports the field-accuracy and flow-structure metrics for the diffusion baseline (chapter 3.3) and the two drifting variants on the held-out test set. The diffusion baseline is the strongest model overall, the label-based drifting variant trails it by a small margin, and the spatial drifting variant comes in third with significantly higher variance.

Table 1 *Quantitative comparison between the diffusion baseline and our drifting model with label-based and spatial conditioning. The metrics were calculated on obstacle-free test data in pixel space.*

Category	Metric	Diffusion (baseline)	Drifting (label)	Drifting (spatial)
Field accuracy	nRMSE (range) ↓	$\mathbf{0.0592 \pm 0.0204}$	0.0684 ± 0.0209	0.1076 ± 0.0453
	R^2 ↑	$\mathbf{0.8476 \pm 0.1228}$	0.8019 ± 0.1152	0.4251 ± 0.4783
	Cos. sim. ↑	$\mathbf{0.8108 \pm 0.0576}$	0.7772 ± 0.0661	0.7846 ± 0.1122
Flow-struct. consistency	Div. gap (pred vs. target) ↓	0.4503 ± 0.0849	$\mathbf{0.4224 \pm 0.0842}$	0.4680 ± 0.3316
	Relative divergence ↓	8.0713 ± 2.1104	$\mathbf{7.5579 \pm 1.6388}$	8.2887 ± 5.4017
	Vort. gap (pred vs. target) ↓	0.2895 ± 0.1290	$\mathbf{0.2634 \pm 0.1312}$	0.5519 ± 0.4618
	Relative vorticity ↓	1.1635 ± 0.0820	$\mathbf{1.1449 \pm 0.0861}$	1.3127 ± 0.2789

Field accuracy. The gap between diffusion and label-based drifting is moderate, supporting our central claim that a single-pass drifting generator can reach competitive accuracy. The spatial variant sacrifices a substantial amount of pixel fidelity, particularly on the nRMSE of 0.108 versus 0.059 for the label variant. Cosine similarity shows a smaller gap, indicating that all three models recover the dominant flow direction even where their magnitudes disagree.

Flow-structure consistency. The label-based drifting model is competitive with the diffusion baseline. It has a smaller divergence gap (0.422 vs. 0.450) and a closer relative-vorticity ratio (1.14 vs. 1.16). The spatial variant again lags, with a vorticity gap that roughly doubles (0.552) and a relative vorticity ratio of 1.31, indicating systematic over-prediction of rotational structure. The increased standard deviations in the spatial column further suggest that the failure mode is concentrated on a subset of geometries rather than spread uniformly across the test set.

Although the label-based drifting model yields stronger quantitative results in our experiments, its reliance on enumerated training configurations limits its flexibility. We view the spatial encoding as the long-term interface for this class of surrogate, as it fundamentally generalizes to unseen room geometries. We attribute the current performance gap to the foundational nature of this implementation, anticipating that future hyperparameter tuning and encoder refinements will naturally improve accuracy.

Inference speed. We measured per-sample inference latency on a single NVIDIA A100, timing the full inference pipeline with CUDA events. Under this protocol, the drifting model is two orders of magnitude faster than the diffusion baseline (table 2). The low standard deviations confirm that both distributions are tight and outlier-free. The current gap reflects a comparison in a less optimised regime. However, the structural advantages of single-pass inference are clearly visible.

Table 2 *Per-sample inference latency on a single NVIDIA A100-SXM4 (80 GB) GPU at batch size 1, over* 1000 *runs after* 50 *warmup iterations.*

Model	Conditioning	NFE	Latency (ms)
Diffusion	label-based	1000	1870.2 ± 77.1
Drifting	label-based	1	$\mathbf{6.74 \pm 0.36}$

4.3.2. Qualitative

In the empty-room scenario (figure 1), both the diffusion baseline and label-based drifting model successfully recover the primary flow topology, including the dominant jet and corner recirculation zones. Although being a single-pass generator, the drifting model (row 1) achieves high structural fidelity. While diffusion (row 2) shows slightly lower peak errors within the jet, the drifting model's error distribution is comparably low across most of the domain. This confirms that drifting can match iterative

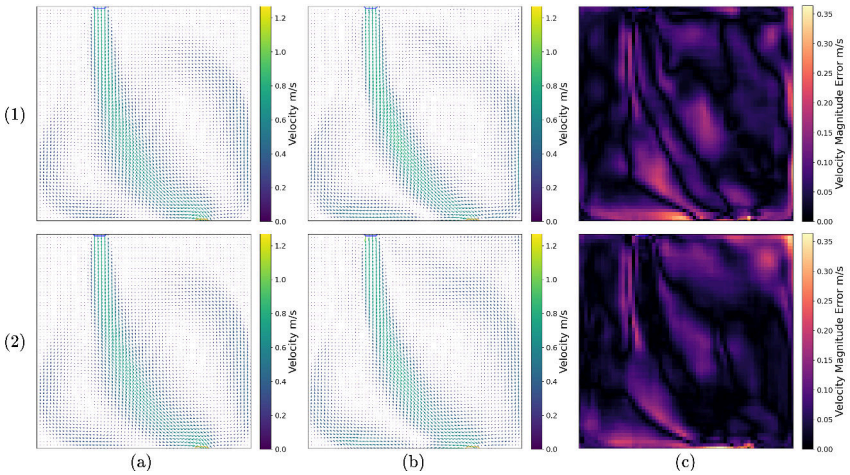


Figure 1 *Comparison of predicted velocity fields for the empty-room configuration. (a) is the ground-truth flow field from CFD, (b) the predicted velocity vector fields for (1) label-based drifting model and (2) diffusion baseline, respectively. (c) illustrates the absolute velocity magnitude error compared to the CFD ground truth. Both models were conditioned on an inlet velocity of 0.20 m/s with identical inlet (blue) and outlet (yellow) positions.*

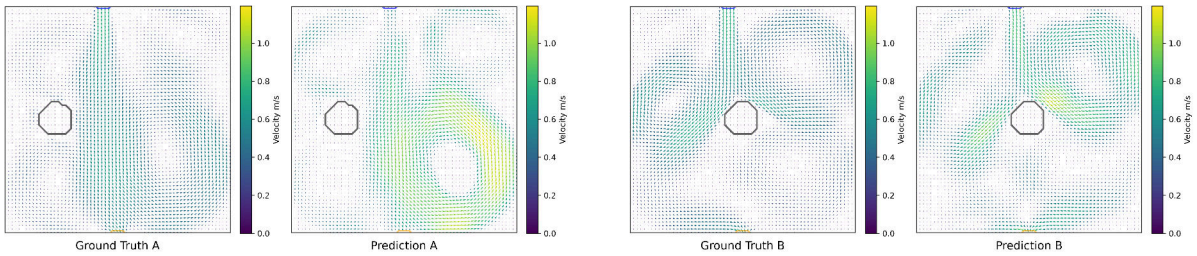


Figure 2 *Comparison of two pairs of velocity fields generated by the drifting model with spatial conditioning. Within each pair, the ground-truth simulation is on the left and the model prediction is on the right. Both cases utilize fixed inlet (7) and outlet (8) positions. The left pair features a left-offset obstacle with inlet velocity of 0.3 m/s, while the right pair features a centered obstacle with inlet velocity 0.1 m/s.*

diffusion performance with lower computational cost while preserving essential physics. Both models exhibit peak errors at jet boundaries, likely due to spatial compression within the VAE architecture.

Figure 2 illustrates the spatial-conditioning variant's performance with obstacles. While trailing the label-based approach (see table 1), the spatial encoded model captures essential topology, including the primary jet trajectory, flow splitting, and boundary separation around the cylinder. It also reasonably approximates wake and corner recirculation locations. The primary discrepancy is in velocity scaling, where the model produces localized over-predicted speed and rotational structures. These errors directly align with the elevated relative vorticity metrics observed in the quantitative analysis.

5. Limitations

Our evaluations focus on 2D steady-state flows on fixed grids to isolate generative fundamentals. However, the dataset's sparsity limits the kernel-based drifting field's potential. Extending to transient 3D flows and multi-state datasets will better leverage the model's distribution-matching and NFE=1 efficiency. Performance is currently limited by the VAE's 16× spatial compression; closing the gap to diffusion requires optimizing latent resolution and patch size. Additionally, the spatial conditioning variant's lag likely stems from lossy geometry mask compression, suggesting a need for richer interfaces like cross-attention. Finally, drifting's two-order-of-magnitude speed advantage is an inherent structural property, providing predictable, high-speed inference without the quality-speed trade-offs of distilled diffusion. Future benchmarks should include tuned models and direct-regression surrogates to further validate drifting's utility in steady-state environments.

6. Conclusion

By adapting a drifting model to 2D steady-state CFD, we achieved performance closely matching a 1000-step diffusion baseline. Although running two orders of magnitude faster, the label-conditioned drifting variant produced an normalised RMSE of 0.068 compared to the baseline's 0.059. While diffusion remains more accurate, the narrow gap makes single-step drifting a viable surrogate for cost-constrained applications. Our methodological contributions, latent-space drifting, spatial-scalar conditioning, and label-aware attraction masks, provide a reusable framework for scientific computing surrogates. Future work will target the spatial-conditioning gap and extend the model to transient 3D regimes to solidify drifting as a competitive generative surrogate.

References

[1] Carmela Concilio *et al.* "CFD simulation study and experimental analysis of indoor air stratification in an unventilated classroom: A case study in Spain". In: *Heliyon* 10.12 (June 30, 2024). DOI: 10.1016/j.heliyon.2024.e32721.
[2] Zhang Lin *et al.* "CFD study on effect of the air supply location on the performance of the displacement ventilation system". In: *Building and Environment* 40.8 (Aug. 1, 2005), pp. 1051–1067. DOI: 10.1016/j.buildenv.2004.09.003.
[3] Ihab Hasan Hatif, Azian Hariri, and Ahmad Fu'ad Idris. "CFD Analysis on Effect of Air Inlet and Outlet Location on Air Distribution and Thermal Comfort in Small Office". In: *CFD Letters* 12.3 (2020), pp. 66–77. ISSN: 2180-1363.
[4] Fatih Topak *et al.* "Collective comfort optimization in multi-occupancy environments by leveraging personal comfort models and thermal distribution patterns". In: *Building and Environment* 239 (2023), p. 110401. DOI: 10.1016/j.buildenv.2023.110401.
[5] Yuexin Bian, Oliver Schmidt, and Yuanyuan Shi. "Operator learning for energy-efficient building ventilation control with computational fluid dynamics simulation of a real-world classroom". In: *Applied Energy* 404 (Feb. 1, 2026), p. 127035. ISSN: 0306-2619. DOI: 10.1016/j.apenergy.2025.127035.
[6] Adrian Tobisch *et al.* "Reducing indoor particle exposure using mobile air purifiers—Experimental and numerical analysis". In: *AIP Advances* 11.12 (Dec. 9, 2021), p. 125114. ISSN: 2158-3226. DOI: 10.1063/5.0064805.
[7] Rui Mao *et al.* "Rapid CFD Prediction Based on Machine Learning Surrogate Model in Built Environment: A Review". In: *Fluids* 10.8 (Aug. 2025), p. 193. ISSN: 2311-5521. DOI: 10.3390/fluids10080193.
[8] Tanya Marwah *et al.* "Deep Equilibrium Based Neural Operators for Steady-State PDEs". In: *Advances in Neural Information Processing Systems* 36 (Dec. 15, 2023), pp. 15716–15737.
[9] Adam T. Müller *et al.* "Reducing Experimental Testing in Space Propulsion Film Cooling Analyses by Pixelwise Generative Image Interpolation". In: *EUCASS* (2025). DOI: 10.13009/EUCASS2025-285.
[10] Dongyu Luo *et al.* "DiffFluid: Plain Diffusion Models are Effective Predictors of Flow Dynamics". In: *arXiv* arXiv:2409.13665 (Sept. 20, 2024). DOI: 10.48550/arXiv.2409.13665. arXiv: 2409.13665[cs.LG].
[11] Mingyang Deng *et al.* "Generative Modeling via Drifting". In: *arXiv* arXiv:2602.04770 (Feb. 6, 2026). DOI: 10.48550/arXiv.2602.04770. arXiv: 2602.04770[cs].
[12] Zongyi Li *et al.* "Fourier Neural Operator for Parametric Partial Differential Equations". In: *arXiv* (2020).
[13] Georg Kohl, Li-Wei Chen, and Nils Thuerey. "Benchmarking autoregressive conditional diffusion models for turbulent flow simulation". In: *Neural Networks* 199 (July 1, 2026), p. 108641. DOI: 10.1016/j.neunet.2026.108641.
[14] Giacomo Baldan *et al.* "Physics vs Distributions: Pareto Optimal Flow Matching with Physics Constraints". In: *The Fourteenth International Conference on Learning Representations* (Oct. 8, 2025).
[15] David Ramos *et al.* "FluidFlow: a flow-matching generative model for fluid dynamics surrogates on unstructured meshes". arXiv.org. In: *arXiv* (Mar. 30, 2026).
[16] Yaron Lipman *et al.* "Flow Matching for Generative Modeling". arXiv.org. In: *arXiv* (Oct. 6, 2022).
[17] Ping He *et al.* "Sinkhorn-Drifting Generative Models". arXiv.org. In: *arXiv* (Mar. 12, 2026).
[18] William Peebles and Saining Xie. "Scalable Diffusion Models with Transformers". In: *2023 ICCV*. Paris, France: IEEE, Oct. 1, 2023, pp. 4172–4182. DOI: 10.1109/ICCV51070.2023.00387.
[19] Robin Rombach *et al.* "High-Resolution Image Synthesis with Latent Diffusion Models". In: *arXiv e-prints*, arXiv:2112.10752 (Dec. 2021), arXiv:2112.10752. DOI: 10.48550/arXiv.2112.10752. arXiv: 2112.10752 [cs.CV].
[20] Qiang Liu and Nils Thuerey. "Uncertainty-aware Surrogate Models for Airfoil Flow Simulations with Denoising Diffusion Probabilistic Models". In: *arXiv* (Dec. 2023). DOI: 10.48550/arXiv.2312.05320.
[21] Reza Behrou *et al.* "Physics-informed multi-fidelity surrogate modeling of fluid flow in porous media". In: *APL Machine Learning* 3.3 (Sept. 8, 2025), p. 036116. ISSN: 2770-9019. DOI: 10.1063/5.0279064.
[22] Ibrahim Reda *et al.* "Rapid indoor airflow prediction using a hybrid residual learning regression model". In: *Energy and Buildings* 352 (Feb. 1, 2026), p. 116827. ISSN: 0378-7788. DOI: 10.1016/j.enbuild.2025.116827.
[23] Wentao Wang *et al.* "Condition monitoring of axial piston pumps based on machine learning-driven real-time CFD simulation". In: *Engineering Applications of Computational Fluid Mechanics* 19.1 (Dec. 31, 2025), p. 2474676. ISSN: 1994-2060. DOI: 10.1080/19942060.2025.2474676.
[24] Dmitrii Kochkov *et al.* "Machine learning–accelerated computational fluid dynamics". In: *Proceedings of the National Academy of Sciences* 118.21 (May 25, 2021), e2101784118. DOI: 10.1073/pnas.2101784118.
[25] Arvind T. Mohan *et al.* "Embedding Hard Physical Constraints in Convolutional Neural Networks for 3D Turbulence". In: *ICLR 2020 Workshop on Integration of Deep Neural Models and Differential Equations*. Feb. 26, 2020.