Session 10

Turbulence

S10-A1	Paul Creusy — Leveraging Machine Learning to discriminate parametric and structural errors in RANS models for Rayleigh-Taylor transition
S10-A3	Thorben Markmann — Generative Modeling of the Spherical Rayleigh-Bénard Convection
S10-A4	Hesam Tofighian — Finite-Volume Convolution for Deep Super-Resolution Reconstruction of Turbulent Flow on Unstructured and Stretched Grids
S10-A5	Maurizio Carbone — Generating turbulence by learning inter-scale cascade dynamics
S10-A9	Mourad Oulghelou — Surrogate-UCB Calibration of RANS Closure Using Sparse Reef Current-Meter Observations
S10-A11	Chiara Calascibetta (Tianyi Li) — Generative Diffusion Models for Turbulent Passive Scalar Fields
S10-P2	Qidi Wu — A Data-Driven Recommendation System for Turbulence-Model Selection
S10-P7	Alejandro Montoya — Archetype-based mixture of experts for
S10-P8	Simon Dalpke — Autoencoder-Based Analysis of Roughness Effects on Heat-Momentum Dissimilarity
S10-P10	Kyriakos Giannakoglou — Data Driven Turbulence Modeling through Eddy Viscosity Adaptation and Symbolic Regression

Leveraging Machine Learning to discriminate parametric and structural errors in RANS models for Rayleigh-Taylor transition

Paul Creusy[1], Benoît-Joseph Gréa[1], Antoine Briard[1], Téo Granger[1]

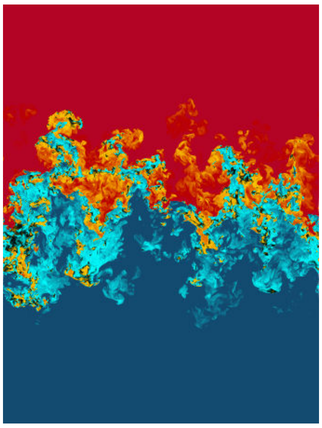
[1] *CEA, DAM, DIF, F-91297 Arpajon, France*

Corresponding author: Paul Creusy (paul.creusy@cea.fr)

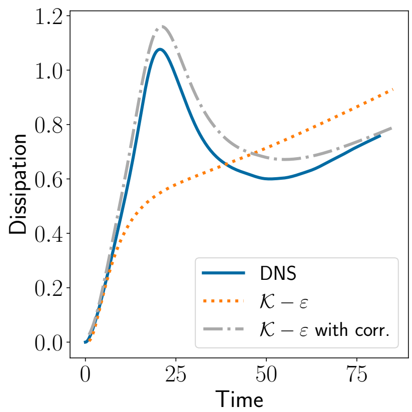
This work proposes a machine learning-based method for calibrating and correcting a 1D RANS $\mathcal{K}$ - ε model to capture the laminar-to-turbulent transition in Rayleigh-Taylor instabilities. The RANS model [1] relies on the turbulent kinetic energy $\mathcal{K}$, its dissipation rate ε, and the heavy fluid mass fraction, all averaged over the statistically homogeneous horizontal plane and extracted from a large DNS database [2].

The calibration framework treats the RANS coefficients as learnable parameters within a *Physics Informed Neural Network* (PINN) [3], enhanced by Residual-Based Attention [4] and Quasi-Newtonian optimizers [5]. Simultaneously, a correction neural network learns the missing forces in the RANS equations, with a penalization to promote minimal intervention. By jointly training both networks, the coefficients converge toward optimal values within the model's valid regime, which is unknown *a priori*, while the correction forces accounts for structural model discrepancies where they exist. The input to the correction neural network consists of the state variables predicted by the PINN, ensuring that the correction is both local and consistent.

A posteriori, symbolic regression using genetic algorithms [6] is applied to the correction neural network to derive an interpretable analytical expression of the augmented model. The integration of the resulting set of equations shows significantly better agreement with DNS data compared to the original model and provides insights into physical phenomena that were not captured initially.



(a) Slice of 3D Rayleigh-Taylor DNS. The color represents, from red (high) to blue (low), the proportion of heavy fluid in the mixing.



(b) Comparison of dissipation on a sample trajectory between DNS simulation, baseline $\mathcal{K}$ - ε model, and an improved version with correction forces.

References

[1] Benoît-Joseph Gréa. The dynamics of the k eps mix model toward its self-similar Rayleigh–Taylor solution. *Journal of Turbulence*, 16(2):184–202, February 2015.

[2] Sébastien Thévenin, Benoît-Joseph Gréa, Gilles Kluth, and Balasubramanya T. Nadiga. Leveraging initial conditions memory for modelling Rayleigh–Taylor turbulence. *Journal of Fluid Mechanics*, 1009:A17, April 2025.

[3] M. Raissi, P. Perdikaris, and G.E. Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, 378:686–707, 2019.

[4] Sokratis J. Anagnostopoulos, Juan Diego Toscano, Nikolaos Stergiopulos, and George Em Karniadakis. Residual-based attention in physics-informed neural networks. *Computer Methods in Applied Mechanics and Engineering*, 421:116805, March 2024.

[5] Jorge F. Urbán, Petros Stefanou, and José A. Pons. Unveiling the optimization process of physics informed neural networks: How accurate and competitive can PINNs be? *Journal of Computational Physics*, 523:113656, February 2025.

[6] Miles Cranmer. Interpretable Machine Learning for Science with PySR and SymbolicRegression, May 2023. arXiv:2305.01582 [astro-ph].

Generative Modeling of the Spherical Rayleigh-Bénard Convection

[1]Hans Harder[*†]; [2]Thorben Markmann[*]; [3]Roshan J. Samuel; [3]Jörg Schumacher; [2]Barbara Hammer; [4,5]Sebastian Peitz

[1]*Department of Computer Science, Paderborn University, Germany*

[2]*Center for Cognitive Interaction Technology, Bielefeld University, Germany*

[3]*Department of Mechanical Engineering, TU Ilmenau, Germany*

[4]*Department of Computer Science, TU Dortmund, Germany*

[5]*Lamarr Institute for Machine Learning and Artificial Intelligence, Dortmund, Germany*

Abstract

Generative modeling has recently achieved remarkable advances across various domains, including fluid dynamics. Applications range from the synthesis of initial conditions to super-resolution, subgrid-scale reconstruction, and the prediction of rare or extreme events. Despite this progress, generative approaches for volumetric flows on spherical domains remain underexplored. We present flow-matching-based generative models for synthesizing physically plausible states of three-dimensional Rayleigh-Bénard convection in a thin spherical shell. This constitutes a widely used simplified description of large-scale atmospheric flows, capturing essential features of long-term climate dynamics and short-term global circulation. Modeling such systems poses unique challenges due to their volumetric structure, spherical topology, and the presence of non-scalar fields. Our key contributions include: (1) a novel extension of 3D UNet architectures to operate on volumetric spherical shells while properly handling non-scalar vector fields, and (2) volumetric extensions of Spherical Neural Operators (SNOs) adapted for shell geometries. We systematically compare these architectures on high-fidelity training data generated using spectral element simulations. The physical fidelity of generated states is assessed by comparing them against DNS data and evaluating physical and statistical properties, including convective heat flux consistency. The results highlight the potential of generative models for volumetric spherical fluid dynamics and establish benchmarks for future developments in climate and geophysical flow modeling.

Keywords:

Flow matching, Generative models, Spherical neural operators, Rayleigh-Bénard convection, Volumetric spherical domains, Atmospheric flows

[*]These authors contributed equally. [†]Corresponding Author, Hans Harder hans.harder@uni-paderborn.de

Finite-Volume Convolution for Deep Super-Resolution Reconstruction of Turbulent Flow on Unstructured and Stretched Grids

[1,2]Hesam Tofighian*; [1]Jordan A. Denev; [2]Nikolai Kornev;

[1]*Scientific Computing Center, Department of Scientific Computing and Mathematics, Karlsruhe Institute of Technology, Karlsruhe 76131, Germany 1*

[2]*Chair of Modeling and Simulation, University of Rostock, Rostock 18051, Germany*

Abstract

We present a finite-volume convolution formulation for deep-learning-based super-resolution on non-uniform and unstructured meshes. The key idea is an exact axis-wise representation of a 3×3×3 three-dimensional convolution as a linear combination of successive one-dimensional convolutions applied along the three coordinate directions. This representation allows a convolution layer to be built from a fixed set of axis-wise operators. While many choices are possible, we adopt three operators, identity, a directional gradient, and a directional average, because each has a direct and consistent finite-volume analogue defined from the local mesh geometry. As a result, the learned convolution remains geometry-aware, and well-defined on general 3D finite-volume grids, including stretched meshes where standard CNN convolutions are not applicable.

The method is assessed on DNS data of turbulent plane channel flow with strongly non-uniform wall-normal spacing. The model achieves high accuracy in reconstructing three-dimensional velocity fields and key turbulence statistics from low-resolution inputs, with particularly strong performance in the near-wall region where gradients and small-scale structures are most pronounced. Compared with a plane-wise 2D CNN baseline, the proposed model yields systematically improved predictions because it can exploit cross-plane information in the wall-normal direction. Cross-plane information is not accessible to plane-wise 2D CNN models. Moreover, standard 3D CNN formulations assume uniform grid spacing and are therefore not directly applicable to non-uniform meshes. We also examine turbulence anisotropy features and observe improved reconstruction of anisotropic behavior, indicating that the model better captures the directional structure and near-wall characteristics of wall-bounded turbulence. Overall, finite-volume convolution provides a practical pathway for accurate 3D super-resolution on realistic CFD grids where conventional convolutional architectures are not applicable.

Keywords: Turbulent flows; Wall-bounded turbulence; Physics-informed machine learning; Super-resolution; Mesh-based neural networks; finite-volume convolution

Generating turbulence by learning inter-scale cascade dynamics

[1]M. Carbone*; [1]V.Valadão; [1]T. Li; [1]M. Sbragaglia; [1]L. Biferale; [1]M. Buzzicotti

[1]*Department of Physics and INFN, University of Rome "Tor Vergata", Rome, Italy*

Abstract

Generating turbulent signals at high Reynolds numbers remains challenging due to the enormous number of active degrees of freedom. While diffusion models can generate and reconstruct time series and fields (e.g., Li *et al.*, *Nat. Mach. Intell.*, 2024), their computational cost may rival that of direct numerical simulation. We propose a scale-recursive generative framework aimed at reducing this burden by compressing the signal into a smaller set of dynamically relevant coefficients. Specifically, building on the long-standing idea that a complex signal with N degrees of freedom may be constructed through the recursive application of a smaller, tractable model, we learn a representation operating on $M \ll N$ variables that reconstructs finer scales from coarse inputs. This enables the generation of consistent multiscale physics up to the gradients statistics and connects the Fokker–Planck description of cascade dynamics (Peinke & Friedrich, *Phys. Rev. Lett.*, 1997) with stochastic interpolants (Albergo *et al.*, *J. Mach. Learn. Res.*, 2025). We benchmark this framework against a wavelet-tree decomposition, where conditional diffusion models generate fine-scale coefficients from coarse ones. While wavelets naturally encode the multiscale structure of the signal (Marchand *et al.*, *Phys. Rev. X*, 2023), direct generation within wavelet subtrees may disrupt spatial correlations. We test the proposed methods on two-dimensional passive scalar turbulence with an extended inertial range (Calascibetta *et al.*, *arXiv*, 2025), focusing on structure functions and anomalous scaling exponents, demonstrating a promising direction for high-fidelity turbulent signal generation.

This work was supported by the Italian Ministry of University and Research (MUR) - Fondo Italiano per la Scienza (FIS2) - 2023 Call, CUP: E53C24003760001, and by the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation program (Grant Agreement Nos. 882340 and 101002724).

Keywords:

Turbulence; Diffusion models; Wavelet analysis.

Surrogate-UCB Calibration of RANS Closure Using Sparse Reef Current-Meter Observations

Mourad Oulghelou*, Tadd Truscott, Sandip Laxman Dighe, and Hannah wilkosz

King Abdullah University of Science and Technology

Abstract

Reef-scale hydrodynamic simulations are central to predicting transport and mixing, yet their predictive skill is often limited by uncertainty in Reynolds-averaged Navier–Stokes (RANS) turbulence closures. We propose a physics-informed, data-driven calibration of an algebraic RANS closure for reef environments using in situ current-meter observations from sparsely distributed loggers deployed across the Shushah reefscape (Red Sea). The closure is formulated as a polynomial expansion of the Reynolds-stress anisotropy in Pope's tensor basis, enforcing frame invariance while providing an interpretable parameterization. We cast calibration as estimating the polynomial coefficients by minimizing a misfit between simulated and observed currents.

Because the observations are sparse and deriving sensitivity information via adjoint methods is cumbersome, we adopt an adjoint-free global optimization strategy based on Gaussian-process (GP) Bayesian optimization. At each iteration, a GP surrogate of the misfit objective is constructed over the coefficient space, providing both a posterior mean prediction and an uncertainty estimate. New candidate coefficients are selected by maximizing an upper confidence bound (UCB) acquisition function, which balances exploration of uncertain regions with exploitation of promising minima. The selected coefficients are evaluated with the RANS solver, the surrogate is updated with the new misfit value, and the procedure is repeated until convergence. This workflow enables sample-efficient calibration of algebraic closure coefficients in data-limited deployments, yielding observation-informed RANS predictions while preserving a physically consistent closure structure.

Keywords: Data-driven RANS closure; reef hydrodynamics; Bayesian optimization, Upper confidence bound.

*Mourad Oulghelou: `mourad.oulghelou@kaust.edu.sa`

Generative Diffusion Models for Turbulent Passive Scalar Fields

[2]Chiara Calascibetta*; [1]Michele Buzzicotti; [1]Tianyi Li; [1]Fabio Bonaccorso; [1]Luca Biferale;

[1]*Department of Physics, University of Rome "Tor Vergata", Via della Ricerca Scientifica 1, 00133 Rome, Italy*

[2]*Université Côte d'Azur, Inria, Calisto team, 06902 Sophia Antipolis, France*

Abstract

Passive scalar fields advected by turbulent flows play a central role in many natural and engineering processes and exhibit highly intermittent multiscale statistics across a wide range of scales, posing a major challenge for data-driven generative modeling. Here we investigate the ability of diffusion-based generative models to synthesize turbulent velocity and scalar fields. Using data from high-resolution numerical simulations of two-dimensional Navier-Stokes turbulence with a passive scalar, characterized by an extended inertial range, we train a diffusion model to generate scalar configurations together with the velocity field in the turbulent inverse cascade regime. The model accurately reproduces key statistical properties of the flow, including multiscale structure functions and scale-dependent scalar fluxes across the inertial range. Furthermore, by treating the unconditional diffusion model as a probabilistic prior, we demonstrate conditional sampling of passive scalar fields guided by partial observations of the velocity field. This approach enables the generation of stochastic realizations of the scalar field conditioned on incomplete flow information. These results highlight the potential of generative diffusion models as flexible probabilistic tools for turbulence modeling, uncertainty quantification, and data assimilation in complex multiscale flows.

This work was supported by the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme Smart-TURB (Grant Agreement No. 882340) and by the Italian Ministry of University and Research (MUR) - Fondo Italiano per la Scienza (FIS2) - 2023 Call, project DeepFL, CUP: E53C24003760001 and the FARE programme (No. R2045J8XAW).

Keywords: passive scalar turbulence, diffusion models, inverse cascade

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

A Data-Driven Recommendation System for Turbulence-Model Selection

[1,2,3]*Qidi Wu;*[1,2,3] *Jiaqing Kou;*[1,2,3]*Shufang Song*;*[1,2,3]*Weiwei Zhang;*

1.School of Aeronautics, Northwestern Polytechnical University, Xi'an 710072, China

2.International Joint Institute of Artificial Intelligence on Fluid Mechanics, Northwestern Polytechnical University, Xi'an 710072, PR China

3.National Key Laboratory of Aircraft Configuration Design, Xi'an 710072, PR China

Keywords: turbulence simulation, turbulence models, intelligent recommendation, machine learning.

Abstract. Turbulence-model selection remains a critical challenge in Reynolds-averaged Navier–Stokes simulations, as different model corrections may lead to notable variations in shock prediction, separation behavior, convergence, and computational cost. This study develops an intelligent recommendation framework for turbulence-model selection in CFD solvers by integrating fuzzy comprehensive evaluation, clustering-based weight optimization, and a random forest classifier. The framework evaluates candidate turbulence-model corrections using accuracy and efficiency indicators, including pressure-coefficient error, convergence time, and iteration number. To reduce the subjectivity of expert-defined weights, K-means clustering and Nelder–Mead optimization are introduced to adaptively adjust indicator weights according to flow-condition similarity. The method is validated on the RAE2822 transonic airfoil under 150 operating conditions, where high-fidelity PHengLEI solutions are used as reference data and SU2 simulations with thirteen SA and SST correction schemes are evaluated. The trained random forest recommender achieves a Top-1 accuracy of 94.0%. A Top-3 recommendation module is further tested on the NACA0012 airfoil, achieving an average coverage rate of 83.3%. The results demonstrate that the proposed framework provides an interpretable and data-driven approach for turbulence-model recommendation with promising generalization capability.

1. Introduction

Computational Fluid Dynamics (CFD) has become an essential tool in aerospace, energy, and biomedical engineering [1], where simulation accuracy and efficiency directly affect design reliability, development cycle, and engineering cost [2]. Among CFD methods, Reynolds-averaged Navier–Stokes (RANS) approaches remain widely used in industrial simulations because they provide a practical balance between computational cost and predictive capability [3,4]. However, their accuracy strongly depends on turbulence-closure selection [5]. Inappropriate model choices may lead to errors in key aerodynamic quantities such as shock location and flow separation, thereby reducing the reliability of engineering simulations [6].

Recent advances in machine learning and artificial intelligence have brought new opportunities to CFD, including improved solver architectures, discretization strategies, and turbulence closures [7–9]. As available methods continue to increase, the practical challenge has shifted from developing new models to selecting suitable ones for specific flow problems. Therefore, automated, interpretable, and transferable recommendation mechanisms are increasingly needed to support reliable CFD decision-making [10,11].

Several studies have investigated data-driven strategies for CFD solver recommendation and parameter tuning. Sousa et al. [12] integrated machine-learning surrogate models with traditional CFD solvers and embedded deep neural networks into the PISO algorithm of OpenFOAM, improving pressure-solver efficiency while maintaining accuracy. Banerjee et al. [13] proposed an active-learning and regression-tree-based method for automatic parameter selection in MESHFREE, achieving performance gains across benchmark cases. Ghorbel et al. [14] combined genetic algorithms with machine-learning surrogate models to optimize solver parameters, thereby improving convergence quality and reducing numerical divergence caused by manual tuning. Yang et al. [15] further showed that aerodynamic optimization results are sensitive to numerical parameters, geometry parameterization methods, and turbulence-closure coefficients. These studies demonstrate the value of intelligent parameter recommendation, but they mainly focus on solver settings rather than the systematic selection of turbulence models.

Recommendation systems and multi-criteria decision-making methods also provide useful methodological support. Jannach et al. [16] introduced multi-objective optimization into recommendation systems to balance accuracy, diversity, and fairness. Zaizi et al. [17] proposed a unified architecture combining multi-objective optimization with recommendation mechanisms, offering a basis for multi-criteria decision-making. Zhang et al. [18] used knowledge-graph embeddings to incorporate prior knowledge and alleviate data sparsity. Muter and Aytekin [19] developed diversity-oriented optimization methods to dynamically balance recommendation accuracy and diversity. Asemi et al. [20] introduced an ANFIS-based recommendation model, improving robustness under uncertainty through fuzzy rules. Alam Bhuiyan and Hammad [21] proposed a multi-criteria decision-support system for engineering material selection, showing the potential of interpretable engineering recommendations. However, these studies are mostly designed for commercial, content, or general engineering scenarios. They do not fully address the physical constraints, accuracy–efficiency trade-offs, convergence behavior, and interpretability requirements involved in CFD turbulence-model selection.

Overall, three gaps remain: existing methods lack generality across complex flow regimes; fixed or single-weight evaluation frameworks are insufficient for multi-objective engineering trade-offs; and many recommendation mechanisms lack physical interpretability, limiting user trust in practical CFD workflows.

To address these issues, this study proposes an intelligent turbulence-model recommendation framework integrating K-means clustering, fuzzy comprehensive evaluation, and random forest learning. K-means clustering is embedded into the recommendation process to identify physical similarity among flow cases and support adaptive weighting in fuzzy evaluation. The random forest model further improves the stability, adaptability, and interpretability of recommendation results. This framework provides a structured basis for advancing CFD toward data-driven and intelligent decision-making.

The remainder of this paper is organized as follows. Section I presents the motivation, related work, and problem statement. Section II introduces the framework architecture, clustering-based fuzzy evaluation, and random forest recommendation model. Section III reports the RAE2822 case study, recommendation-system construction and generalization test on the NACA0012 airfoil. Section IV concludes the paper and discusses future work.

2. Methodology

In this study, a unified recommendation framework is developed for turbulence-model selection in CFD. The framework consists of three modules: fuzzy comprehensive evaluation for solver performance assessment, clustering-based automatic weight optimization, and a machine-learning recommendation model. These modules jointly evaluate solver performance, optimize indicator weights, and predict the most suitable turbulence-model variant under given flow conditions.

From a machine-learning perspective, the recommendation task is formulated as a supervised multi-class classification problem. Given the flow-condition vector composed of Mach number Ma and angle of attack AoA, the model learns a nonlinear mapping from flow conditions to the turbulence-model label. Since thirteen candidate turbulence-model variants are considered, the final recommendation model is constructed as a 13-class classifier.

2.1 Fuzzy Comprehensive Evaluation Model Based on Clustering Optimization

（1）Fuzzy Comprehensive Evaluation Model

The fuzzy comprehensive evaluation (FCE) method [22] is used to assess solver performance under multiple criteria. By introducing membership functions, FCE transforms heterogeneous indicators into a unified fuzzy evaluation space and is therefore suitable for problems involving uncertainty and incomplete information. In this study, the Analytic Hierarchy Process (AHP) [23] is used to construct the indicator hierarchy and determine initial weights through pairwise comparisons.

The FCE-AHP framework can simultaneously consider conflicting criteria such as accuracy and efficiency [24]. However, conventional AHP relies heavily on expert judgment, which may introduce subjectivity and reduce adaptability under different flow conditions. Therefore, a clustering-based automatic weight optimization strategy is introduced to improve the objectivity and adaptability of the evaluation process.

（2）Selection of Evaluation Indicators

Solver performance is evaluated from two aspects: computational accuracy and computational efficiency. Accuracy reflects the ability of the solver to reproduce flow features, while efficiency describes computational cost and convergence behavior. Accordingly, the first-level indicators are defined as computational accuracy and computational efficiency.

At the second level, accuracy is represented by the Root Mean Square Error (RMSE) and Mean Relative Error (MRE), as shown in Eqs. (1) and (2). Efficiency is represented by convergence time and the number of iterations. These indicators together provide a quantitative basis for evaluating solver applicability under different flow conditions.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(C_{p,pred,i} - C_{p,true,i})^2} \tag{1}$$

$$MRE = \frac{\sum_{i=1}^{n}\left|C_{(p,pred,i)} - C_{(p,true,i)}\right|}{\sum_{i=1}^{n}\left|C_{(p,true,i)}\right|} \times 100\% \tag{2}$$

（3）Automatic Weight Optimization Based on Clustering

To reduce the subjectivity of manual weighting, K-means clustering is introduced to optimize the indicator weights. The initial weights are obtained from expert knowledge and then updated through data-driven optimization. In the proposed framework, flow conditions and turbulence-model identities are used to construct the clustering feature space, while FCE scores provide the performance information for weight adjustment.

The optimization process includes four steps. First, an initial scoring function is established using RMSE, MRE, convergence time, and iteration number;

$$S_i(Ma, AoA, T_k) = w_1(w_{11} \cdot E_{RMSE}^{(k)} + w_{12} \cdot E_{MSE}^{(k)}) + w_2(w_{21} \cdot T_{conv}^{(k)} + w_{22} \cdot N_{iter}^{(k)}) \tag{3}$$

Second, a three-dimensional feature space is constructed using Ma, AoA, and the turbulence-model index, and K-means clustering is performed to identify similar flow cases. Third, an entropy-regularized objective function is introduced to maintain consistency among similar cases while avoiding excessive concentration of weights on a single indicator. The objective function is minimized using the Nelder–Mead algorithm, which is suitable for nonlinear and derivative-free optimization problems. Finally, clustering and weight optimization are iteratively updated until the loss variation becomes negligible.

This strategy enables adaptive weight adjustment and encourages similar flow conditions to obtain consistent turbulence-model recommendations. The complete workflow is shown in Figure 1.

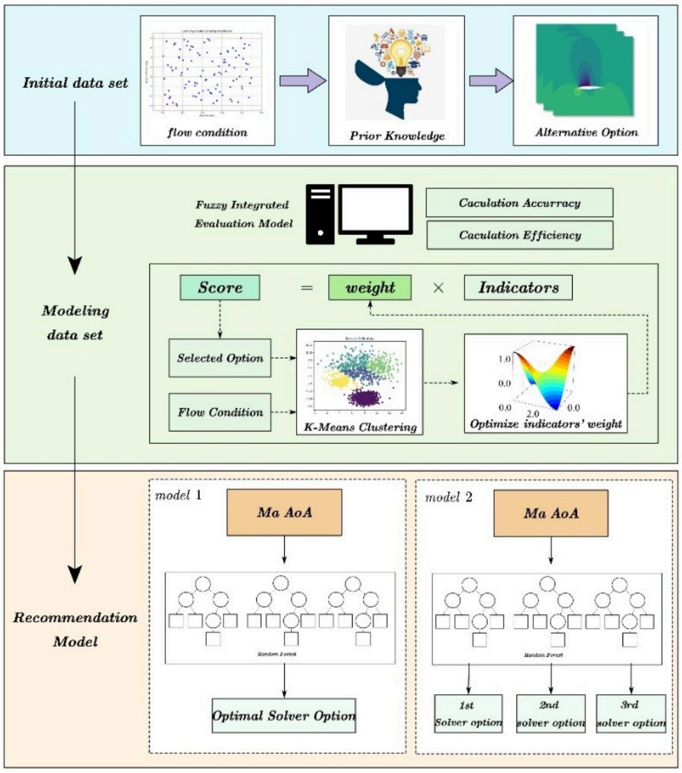


Figure 2. Schematic diagram of the proposed turbulence-model recommendation framework.

3. Sections, subsections and subsubsections

3.1 Solver and Candidate Turbulence Models

All numerical simulations were performed using the open-source CFD solver SU2 [27], which solves the RANS equations with a finite-volume discretization. In this study, turbulence-model correction terms implemented in the SA and SST models were considered as candidate schemes. The SA variants include BCM, COMPRESSIBILITY, EDWARDS, NEGATIVE, NONE, QCR2000, ROTATION, and WITHFT2, while the SST variants include KATO_LAUNDER, SUSTAINING, V1994m, V2003m, and VORTICITY. These thirteen schemes cover the main correction mechanisms used in engineering turbulence modeling. Their theoretical backgrounds and applicable flow regimes are summarized in Tables 1 and 2.

Table 1. Theoretical framework of correction terms for the Spalart–Allmaras (SA) turbulence model.[28-30]

Correction term	Computational principle	Applicable flow regime
BCM	Introduces a nonlinear combination of strain-rate and rotation tensors to improve eddy-viscosity prediction.	Strong shear flows (airfoil separation, shock–boundary-layer interaction).
COMPRESSIBILITY	Modifies eddy viscosity through the turbulent Mach number to account for compressibility effects	High-Mach-number flows (Ma > 3).
EDWARDS	Adjusts the destruction-term coefficient to enhance adaptability under adverse pressure gradients.	Separated flows; may overpredict the extent of separation.
NEGATIVE	Deactivates turbulent viscosity when the modified viscosity variable ($\tilde{\nu}$) becomes negative to prevent numerical instability.	Low-quality grids or transitional flows.

QCR2000	Introduces nonlinear terms of the rotation tensor to improve anisotropic stress representation.	Three-dimensional complex flows; may enlarge predicted separation zones.
ROTATION	Suppresses turbulence production when vorticity exceeds strain rate.	Rotation-dominated flows (e.g., compressor corner separation); tends to yield the lowest peak values.

Table 2. Theoretical framework of correction terms for the Shear Stress Transport (SST) turbulence model.[31-34]

Correction term	Computational principle	Applicable flow regime
KATO_LAUNDER	Replaces vorticity with strain rate in the production term to suppress excessive turbulence generation in separated regions.	Strong adverse pressure-gradient flows; may introduce deviation in heat-flux prediction.
SUSTAINING	Enforces a minimum turbulence level outside the boundary layer to prevent laminarization.	High-Reynolds-number external flows; should be used in conjunction with a transition model.
V1994m	Based on the 1994 Wilcox modification, adjusts the weighting of the vorticity term.	General flows; may underestimate shear-layer development.
V2003m	Derived from the 2003 Wilcox revision, optimizes the balance between vorticity and strain-rate terms.	Separated flows; performs better than V1994m.
VORTICITY	Uses only vorticity for the production term to reduce strain-rate interference.	Rotating machinery; sensitive to shear-dominated flows.

3.2 RAE2822 Airfoil Case

The RAE2822 transonic airfoil was selected as the benchmark case because of its well-documented experimental database. The baseline condition was Ma=0.729, AoA=2.31, Re=6.5×10^6. Experimental pressure data were obtained from the National Space Science Data Center and used for comparison with numerical results (Figure).

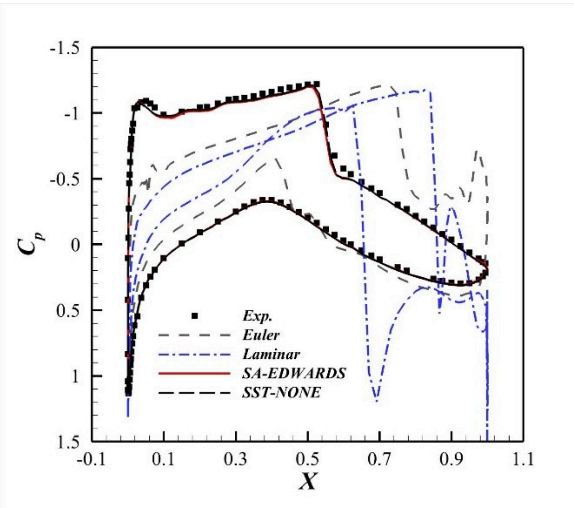


Figure 3. Comparison of computed surface-pressure distributions with experimental data.

The evaluation indicators included RMSE and MRE for computational accuracy, and convergence time and iteration number for computational efficiency. A fuzzy comprehensive evaluation model was used to calculate the overall performance score of each turbulence-model scheme. To reduce the subjectivity of manual weighting, K-means clustering and Nelder–Mead optimization were introduced to automatically update the indicator weights. The optimized results showed that computational accuracy played a dominant role in the final evaluation, which is consistent with the objective of identifying physically reliable turbulence models under transonic conditions.

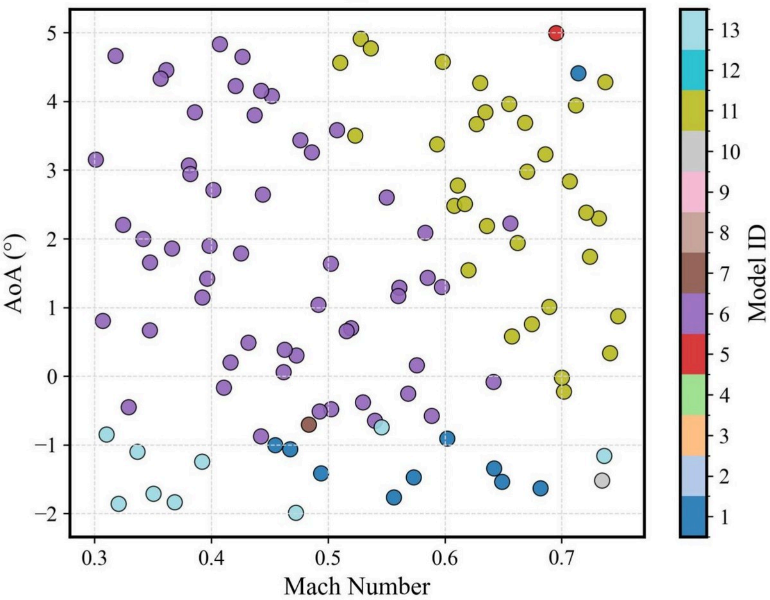


Figure 6. Distribution of optimized turbulence-model schemes after clustering.

Based on the optimized evaluation results, the best-performing turbulence-model scheme under each flow condition was selected as the label for machine-learning training. A random forest classifier was then constructed using Ma and AoA as inputs and the optimal turbulence-model scheme as the output. Cross-validation showed that the model achieved a recommendation accuracy of 94.0%, indicating good predictive capability and robustness (Figure).

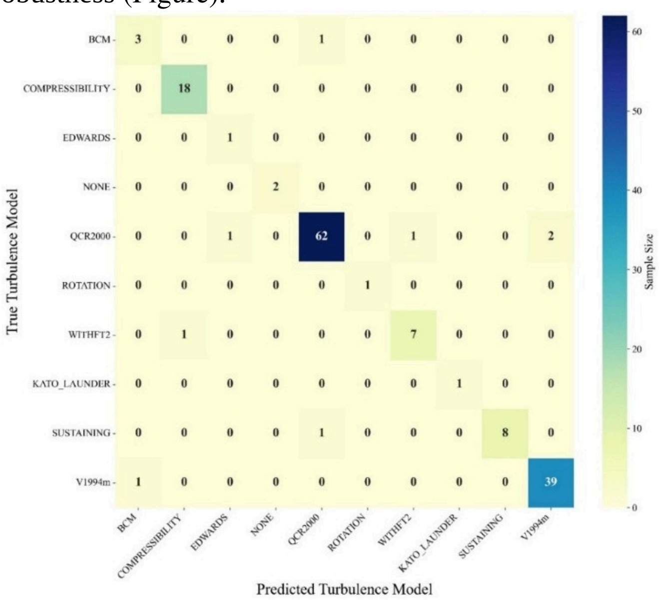


Figure7. Confusion-matrix heatmap of the random-forest model.

To improve engineering applicability, a Top-3 recommendation module was further developed. Instead of providing only one optimal scheme, this module outputs the three highest-ranking turbulence-model schemes, offering alternative choices when the primary recommendation is constrained by convergence, cost, or physical considerations(Figure 8).

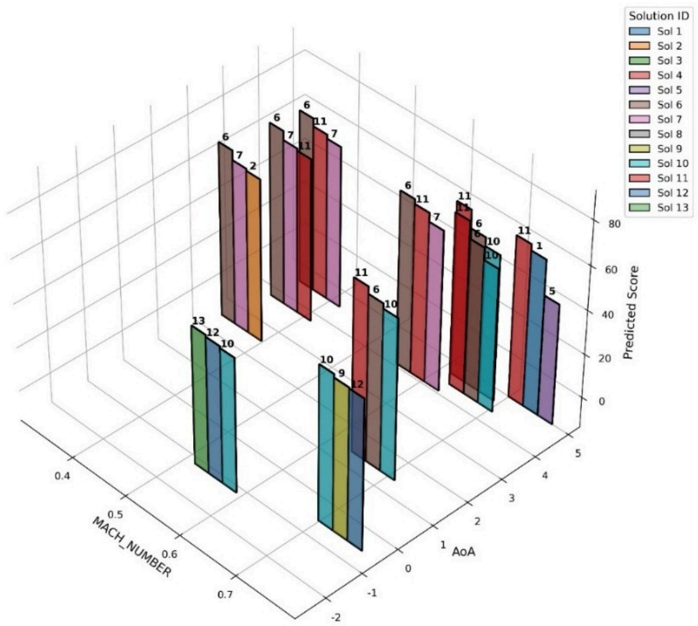


Figure 8. Prediction results of the multi-scheme recommendation model.

3.4 Generalization Verification

To evaluate the generalization capability of the proposed framework, the trained recommendation model was tested on the NACA0012 airfoil. Two grids were generated: a high-fidelity reference mesh with 47,414 cells and an SU2 mesh with 29,654 cells. Since the input features remained Ma and AoA, the model could be directly applied to this new geometry.

Figure 9 compares the predicted and true top-three schemes for the six cases. Most conditions show an overlap of two or more, indicating stable prediction behavior.

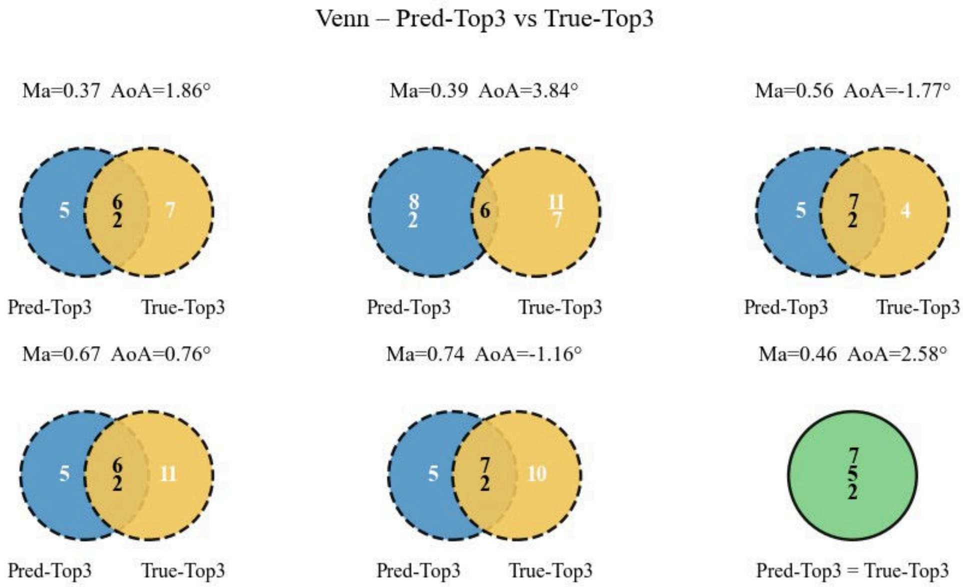


Figure 9 Comparison between predicted Top-3 schemes and true Top-3 schemes across six flow conditions

The predicted Top-3 schemes were compared with the true Top-3 schemes obtained from fuzzy comprehensive evaluation. The results showed that most test cases achieved an overlap of two or more schemes. The success rate was used to characterize the prediction performance of the model, as defined in Eq. (4):

$$\text{Success rate} = \frac{N_{\text{hit}}}{3 \times N_{\text{case}}} \times 100\% \tag{4}$$

Across 20 test conditions, the average Top-3 coverage rate reached 83.3%. This demonstrates that the proposed recommendation model maintains stable performance under an unseen airfoil geometry and has potential for broader aerodynamic applications.

4.Conclusions

In this study, an intelligent recommendation framework was developed for turbulence-model selection in CFD solvers. The framework integrates fuzzy comprehensive evaluation, clustering-based weight optimization, and a random forest classifier, providing a data-driven and interpretable alternative to experience-based model selection.

The proposed method was evaluated on the RAE2822 airfoil under 150 flow conditions, with high-fidelity PHengLEI results used as reference data. The trained recommendation model achieved a Top-1 accuracy of 94.0%, indicating that the framework can effectively identify suitable turbulence-model correction schemes across different Mach numbers and angles of attack. In addition, the Top-3 recommendation module provides alternative model choices for engineering applications.

The generalization capability of the model was further tested on the NACA0012 airfoil. The predicted Top-3 schemes achieved an average coverage rate of 83.3%, demonstrating that the learned recommendation mapping remains effective under moderate geometric variation.

Overall, the proposed framework reduces the subjectivity of turbulence-model selection and improves the consistency of CFD decision-making. Future work will incorporate geometric and mesh-related features to enhance generalization while maintaining interpretability.

References

1. Zhao F, Ji X, Shyy W et al (2022) A compact high-order gas-kinetic scheme on unstructured mesh for acoustic and shock wave computations. J Comput Phys 449:110812
2. Cai S, Mao Z, Wang Z et al (2021) Physics-informed neural networks (PINNs) for fluid mechanics: a review. Acta Mech Sin 37:1727
3. Serra N (2023) Revisiting RANS turbulence modelling used in built-environment CFD simulations. Build Environ 237:110333
4. Liu H, Kong Z, Li G et al (2024) Uncertainty quantification of the standard k-ε turbulence model closure coefficients in predicting aerodynamics of high-speed train. Eng Appl Comput Fluid Mech 18:2430658
5. Cheng C, Chen X, Zhu W et al (2024) Progress in physical modeling of compressible wall-bounded transitional and turbulent flows. Acta Mech Sin 40:323663
6. Economon TD, Palacios F, Copeland SR et al (2016) SU2: An open-source suite for multiphysics simulation and design. AIAA J 54:828
7. Wang J, Liu W, Xie H et al (2025) Advancing buffet onset prediction: a deep learning approach with enhanced interpretability for aerodynamic engineering. Complex Intell Syst 11:1
8. Li J, Du X, Martins JRRA (2022) Machine learning in aerodynamic shape optimization. Prog Aerosp Sci 134:100849
9. Jiang ZH, Yan C, Yu J (2023) A simple a posteriori indicator for discontinuous Galerkin method on unstructured grids. Acta Mech Sin 39:322296
10. Bhattarai B, Hilliard B, Tonina D et al (2024) Evaluation of Reynolds-averaged Navier-Stokes turbulence models in open channel flow over salmon redds. J Hydrodyn 36:741
11. Shankar V, Chakraborty D, Viswanathan V et al (2025) Differentiable turbulence: Closure as a partial differential equation constrained optimization. Phys Rev Fluids 10:024605
12. Sousa P, Rodrigues CV, Afonso A (2024) Enhancing CFD solver with machine learning techniques. Comput Methods Appl Mech Eng 429:117133
13. Banerjee P, Padmanabha M, Sanghavi C et al (2024) Machine learning optimized approach for parameter selection in MESHFREE simulations. arXiv:2403.13672
14. Ghorbel H, Zannini N, Cherif S et al (2019) Smart adaptive run parameterization (SArP): enhancement of user manual selection of running parameters in fluid dynamic simulations using bio-inspired and machine-learning techniques. Soft Comput 23:12031
15. Yang G, Da Ronch A, Drofelnik J et al (2018) Sensitivity assessment of optimal solution in aerodynamic design optimisation using SU2. Aerosp Sci Technol 81:362
16. Jannach D, Abdollahpouri H (2023) A survey on multi-objective recommender systems. Front Big Data 6:1157899
17. Zaizi FE, Qassimi S, Rakrak S (2023) Multi-objective optimization with recommender systems: A systematic review. Inf Syst 117:102233
18. Zhang JC, Li Y, Wu X et al (2024) A review of recommender systems based on knowledge graph embedding. Expert Syst Appl 250:123876
19. Muter I, Aytekin T (2017) Incorporating aggregate diversity in recommender systems using scalable optimization approaches. INFORMS J Comput 29:405
20. Asemi A, Asemi A, Ko A (2023) Adaptive neuro-fuzzy inference system for customizing investment type based on the potential investors' demographics and feedback. J Big Data 10:87
21. Bhuiyan MMA, Hammad A (2023) A hybrid multi-criteria decision support system for selecting the most sustainable structural material for a multistory building construction. Sustainability 15:3128
22. Zadeh LA (1965) Fuzzy sets. Inf Control 8:338
23. Saaty TL (1990) How to make a decision: The analytic hierarchy process. Eur J Oper Res 48:9
24. Wang C, Jin Z, Liu X et al (2024) A coal bursting liability evaluation model based on fuzzy set theory and analysis of three influencing factors. Front Earth Sci 12:1378956
25. Li G, Wang H, Pan T et al (2023) Fuzzy comprehensive evaluation of pilot cadets' flight performance based on G1 method. Appl Sci 13:12058
26. Zhang W, Kou J, Liu Y (2021) Prospect of artificial intelligence empowered fluid mechanics. Acta Aeronaut Astronaut Sin 42:524689
27. Cao W, Shan X, Tang S et al (2025) Solving parametric high-Reynolds-number wall-bounded turbulence around airfoils governed by Reynolds-averaged Navier–Stokes equations using time-stepping-oriented neural network. Phys Fluids 37:015151
28. Shen H, Zhang D, Rinoshika A et al (2025) Flow field prediction with self-supervised learning and graph transformer: A high performance solution for limited data and complex flow scenarios. Phys Fluids 37:045112
29. Yang Z, Shan X, Yang XIA et al (2025) Data-enabled discovery of specific and generalisable turbulence closures. J Fluid Mech 1016:R1
30. Xue Y, Feng Y, Zheng X (2024) Assessment of compressibility corrections on Spalart–Allmaras turbulence model for high-Mach-number flows. AIAA J 62:92–107
31. Hou L, Zhu B, Wang Y (2023) kεNet: discovering the turbulence model and applying for low Reynolds number turbulent channel flow. Acta Mech Sin 39:322326
32. Zeng Y et al (2024) A multi-physics correction of the shear-stress transport turbulence model for supersonic flows. Phys Fluids 36:055105

33. Brunton SL (2021) Applying machine learning to study fluid mechanics: introduction and overview. Acta Mech Sin 37:1718–1726
34. Chedevergne F, Marchenay Y (2019) Transpired turbulent boundary layers: a general strategy for RANS turbulence models. J Turbul 20:681–696
35. Wan Y et al (2023) An efficient communication strategy for massively parallel computation in CFD. J Supercomput 79:7560–7583

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

Archetype-based mixture of experts for interpretable and generalisable data-driven turbulence modelling

Alejandro Montoya Santamaría[1], Paola Cinnella [1]

[1]Institut Jean Le Rond d'Alembert, Sorbonne Université, CNRS, Paris, France

Keywords: Turbulence Modelling, RANS, Variational Autoencoders, Mixture of Experts

Abstract. The present study outlines an interpretable and transferable mixture-of-experts strategy to augment the k-ω SST turbulence model, where expert corrections are associated with learned archetypal flow states. A simplex-constrained variational autoencoder identifies these archetypes and provides local convex weights used to train sparse, interpretable symbolic-regression corrections. The resulting closure is a weighted combination of experts, interpretable as a composition of canonical flow-process corrections rather than a monolithic model. On the ERCOFTAC turbulence challenge benchmark, the model reduces the scaled velocity error from 0.104 for the baseline model to 0.063. It also improves the prediction of the horseshoe vortex in a 3D wing-body junction case, highlighting its potential for complex engineering flows.

1. Introduction

Reynolds-averaged Navier–Stokes (RANS) models remain central to Computational Fluid Dynamics (CFD) because of their favourable cost–accuracy trade-off. In recent years, data-driven approaches have been proposed to augment classical turbulence closures using high-fidelity data, improving predictive accuracy without significantly increasing computational cost [1]. Although these methods can perform remarkably well on flows similar to their training cases, their ability to generalise remains limited [2]. Most data-driven closures rely on a single global correction, whereas realistic flows are inherently heterogeneous. Within a single configuration, multiple physical mechanisms may coexist, including attached boundary layers, shear layers, and secondary flows. As a result, global models often compromise between competing behaviours, leading to overfitting or degraded performance. Mixture-of-experts (MoE) strategies address this limitation by combining specialised models through a gating function. For turbulence modelling, Oulghelou et al. [2] proposed an MoE approach in which experts are trained on separate flow cases and combined through learned weights. While effective, this strategy remains tied to case-based partitioning and does not account for the coexistence of multiple flow mechanisms within a single case.

This study explores an alternative approach based on the identification of "archetypal flow states". We seek a representation in which local flow features can be expressed as convex combinations of a small number of canonical states, or "archetypes" [3]. These archetypes represent extremal patterns in feature space, intended to capture relatively pure occurrences of elementary flow processes. Generic flow regions are then described as mixtures of these archetypes. To learn such a representation, we employ a K-Simplex Variational Autoencoder (KSVAE) [4, 5], which constrains the latent space to a simplex structure, providing for each data point a set of convex weights associated with the archetypes. These weights are then used to train a set of sparse symbolic-regression models using the SBL-SpaRTA framework [6], yielding one expert per archetype. The resulting closure can therefore be interpreted as a composition of process-specific corrections, rather than as a monolithic model. This formulation enables the construction of interpretable and locally adaptive turbulence models, where the contribution of each expert is directly linked to the underlying flow mechanisms. In the following, we describe the proposed methodology and assess its performance on the ERCOFTAC turbulence challenge benchmark [7], and on a three-dimensional wing-body junction case representative of complex engineering flows.

2. Archetype-based MoE model

We investigate the incompressible RANS equations for a fluid with constant properties,

$$\partial_i U_i = 0, \qquad U_j \partial_j U_i = \partial_j \left[-\frac{1}{\rho} P + \nu \partial_j U_i - \tau_{ij} \right], \tag{1}$$

where U_i is the mean velocity, ρ is the constant density, P is the mean pressure and ν is the constant kinematic viscosity. The Reynolds-stress τ_{ij} is the target of modelling. The proposed data-driven turbulence model augments the k-ω SST model by Menter [8], which computes the eddy viscosity ν_t using transport equations for the turbulence kinetic energy k and the specific turbulence dissipation rate ω. This model, which we use as baseline, is augmented using the k-corrective frozen RANS methodology of Schmelzer et al. [9] by adding a correction on the production of turbulence kinetic energy, R_k, and a correction on the Reynolds stress anisotropy, b_{ij}^{Δ}.

2.1. *Learning archetypal flow states with a simplex-constrained latent space*

To identify flow regions needing turbulence model correction, we generate, for each training flow configuration, a baseline k-ω SST solution. Given a set of cellwise baseline RANS data described by prescribed feature vectors $\boldsymbol{\eta}_i \in \mathbb{R}^N$, we seek a low-dimensional representation $\mathbf{z}_i = \mathbf{z}_i(\boldsymbol{\eta}_i) \in \mathbb{R}^K$, $K < N$, in which each data point can be expressed as a convex combination of a small number of canonical states, referred to as archetypes. In contrast to clustering approaches, which assign each point to a single group, this representation allows intermediate flow regions to be described as mixtures of archetypes. Formally, each latent vector $\mathbf{z}_i$ is expressed as a convex combination of M archetype vectors $\boldsymbol{a}_j \in \mathbb{R}^K$, where $M \leq K + 1$,

$$\mathbf{z}_i = \sum_{m=1}^{M} w_{m,i}\boldsymbol{a}_m, \qquad w_{m,i} \geq 0, \qquad \sum_{m=1}^{M} w_{m,i} = 1, \tag{2}$$

where the coefficients $w_{m,i}$ define the archetype weights, which quantify the proximity of a data point i to each archetype in the latent space, and provide a natural basis for local model adaptation.

To learn this representation, we employ a Variational Autoencoder (VAE) architecture [10], presented visually in Figure 1, in which the latent space is constrained to lie on a probability simplex. The encoder maps the input features $\boldsymbol{\eta}$ to a latent variable $\boldsymbol{h}$, from which the archetype weights are obtained via a softmax transformation, $\boldsymbol{w} = \sigma(\boldsymbol{h})$, ensuring positivity and unit sum. The decoder reconstructs the input features from the latent representation, with the first linear layer parameters effectively defining the archetype vectors $\boldsymbol{a}_m$. We regularise the latent space by imposing at each point a Dirichlet prior distribution on the weights vector, $\boldsymbol{w} \sim \mathrm{Dir}(\boldsymbol{\alpha})$, where, for simplicity, $\alpha_m = \alpha$. In practice, the Dirichlet prior is approximated by a logistic-normal distribution [4, 5], which allows standard reparametrisation-based training of the VAE [10]. The model parameters for the probabilistic encoder $q_\phi(\boldsymbol{h}|\boldsymbol{\eta})$, and decoder $p_\theta(\boldsymbol{\eta}|\boldsymbol{h})$, are learned by minimising the negative evidence lower bound (ELBO),

$$\mathcal{L} = -\mathbb{E}_{q_\phi(\boldsymbol{h}|\boldsymbol{\eta})}\left[\log p_\theta(\boldsymbol{\eta}|\boldsymbol{h})\right] + \beta D_{\mathrm{KL}}\left(q_\phi(\boldsymbol{h}|\boldsymbol{\eta}) \,\|\, p(\boldsymbol{h})\right), \tag{3}$$

where the first term is the reconstruction error, the second term promotes the prior distribution $p(\boldsymbol{h})$, and the hyperparameter β controls the balance between reconstruction fidelity and adherence to the simplex prior [11].

For each spatial location, the resulting model provides a set of archetype weights that reflect the local flow character, acting as a soft decomposition of the flow into a limited set of canonical processes. These weights are subsequently used to train and combine the expert models.

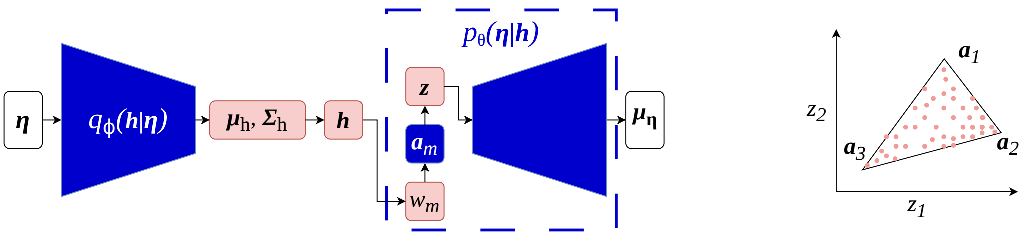


Figure 1: Graphic representation of the KSVAE network. In (a) the architecture of the network is presented. (b) shows an illustrative example scatter plot of data auto-encoded in the simplex-constrained latent space for $K = 2$, $M = 3$.

2.2. *Training symbolic experts conditioned on archetypes*

We express the corrections b_{ij}^{Δ} and R_k using the general eddy-viscosity framework of Pope [12] formulated for two dimensional flows. The corrections are projected onto a minimal integrity basis,

$$b_{ij}^{\Delta} = \sum_{n=1}^{3} \alpha_n^{\Delta}(I_1, I_2)\, T_{ij}^{(n)}(S_{ij}, \Omega_{ij}, \omega), \qquad R_k = \sum_{n=1}^{3} \alpha_n^{R}(I_1, I_2)\, T_{ij}^{(n)}(S_{ij}, \Omega_{ij}, \omega)\partial_j U_i, \tag{4}$$

where $T_{ij}^{(n)}$ are tensor basis functions constructed from the mean-flow strain-rate S_{ij} and rotation-rate Ω_{ij} tensors, non-dimensionalised by ω, and α_n^Δ, α_n^R are scalar functions of tensor invariants I_1, I_2. The coefficients α_n^Δ and α_n^R are identified using the SBL-SpaRTA symbolic regression framework [6], which yields sparse and interpretable algebraic expressions.

Instead of learning a single global model, we train M experts, one per archetype. The archetype weights are used as soft assignments during training, meaning each expert is primarily informed by data points where its corresponding archetype is dominant. The final model is obtained as a convex combination of the expert predictions, as shown in Equation (5). This formulation leads to a closure that is both interpretable and locally adaptive: each expert captures a specific flow mechanism, while the archetype weights determine their spatial combination,

$$b_{ij}^\Delta = \sum_{m=1}^{M} w_m b_{ij}^{\Delta,m}, \qquad R_\text{k} = \sum_{m=1}^{M} w_m R_\text{k}^m. \tag{5}$$

Hence, the model can be viewed as a composition of process-specific corrections rather than as a single global law.

3. Model training

To train the KSVAE, we use k–ω SST RANS data from the ERCOFTAC turbulence challenge benchmark [7]. The vector of input features $\boldsymbol{\eta}$, has size $N = 10$. The first nine are η_{1-9} as presented by Oulghelou et al. [2, 13], to which $\eta_\omega = ||\boldsymbol{\Omega}||k/(||\boldsymbol{\Omega}||k + \varepsilon)$ is added, where ε is the turbulence dissipation rate. The encoder and decoder are fully connected networks with layer sizes (N, 25, 25, 100, M) and (K, 100, 25, 25, N), respectively. We add the constraint $M = K + 1$, meaning that the latent representations $\mathbf{z}$ exist in a K-simplex. Hyperparameters α, β, and K are selected via grid search, where predicted archetype weights are used to train SBL-SpaRTA experts.

The experts are trained on the high-fidelity fields from the training cases of [7], interpolated onto the RANS grids, using an 80/20 train/validation split. The training data are modulated by the archetype weights predicted by the KSVAE. Model selection is based on the Weighted Root Mean Squared Error (WRMSE) computed on the validation set, defined in Equation (6). For SBL-SpaRTA, the sparsity parameter is set to $\lambda^* = 10^5$ [6]. The selected model has $\beta = 0.1$, $\alpha = 0.3$, $K = 5$, and achieves a WRMSE of 0.85.

$$\text{WRMSE} = 0.5\frac{\text{RMSE}(R_{\text{k,pred}}, R_{\text{k,true}})}{\text{RMS}(R_{\text{k,true}})} + 0.5\frac{\text{RMSE}(b^\Delta_{ij,\text{pred}}, b^\Delta_{ij,\text{true}})}{\text{RMS}(b^\Delta_{ij,\text{true}})}. \tag{6}$$

3.1. Interpreting the learned flow archetypes

To analyse the structure of the learned latent space, we project the 5-dimensional latent vectors $\mathbf{z}$ onto two dimensions using the t-SNE algorithm [14], at the cost of losing the simplex structure. The result is shown in Figure 2.

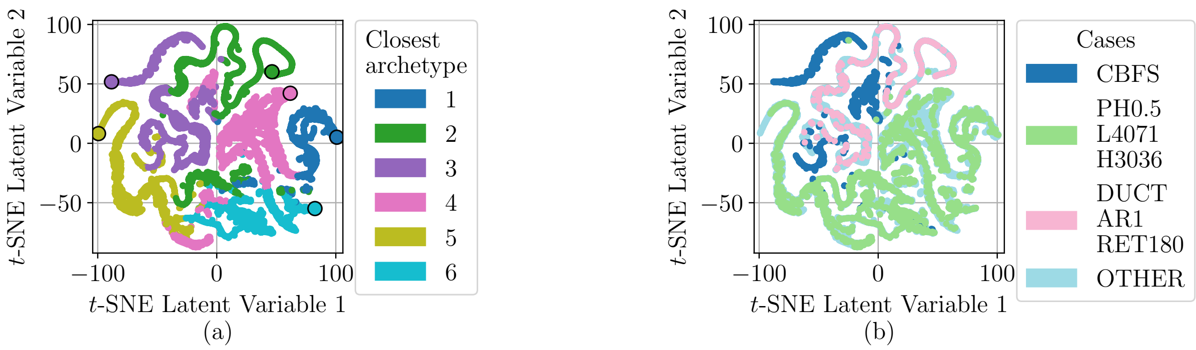
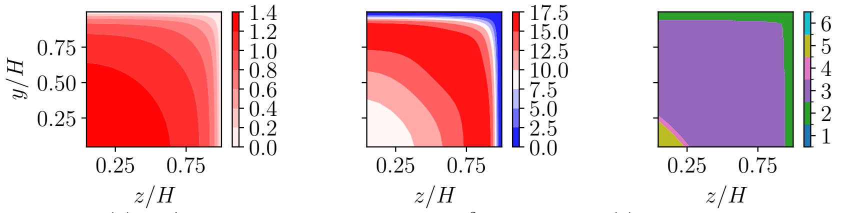


Figure 2: Training data represented in the simplex constrained latent space projected to 2D using the t-SNE algorithm. In (a) coloured based on the closest archetype, in (b) coloured based on the case. In (a) the archetype vectors are plotted using a larger marker.

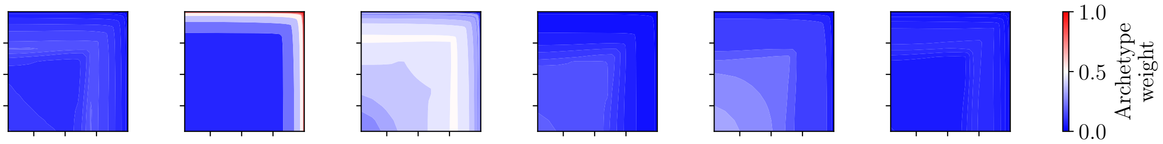
Data points corresponding to different flow configurations occupy partially overlapping regions of the latent space, indicating that the representation is not organised by case, but by local flow features. In particular, similar flow regions arising in different geometries (e.g. periodic hills (PH) and curved backward-facing step (CBFS)) are mapped to neighbouring areas. This suggests that the latent space captures common underlying flow behaviours rather than case-specific characteristics. The archetypes are located on the boundary of the latent space and correspond to extremal patterns in the feature distribution. This representation enables smooth transitions between archetypes, allowing intermediate flow regions to be described as mixtures of multiple canonical processes.

To interpret the physical meaning of the archetypes, we visualise the dominant archetype and the corresponding weights in physical space for representative training cases (Figures 3–6). The fields reveal a consistent organisation of the flow. Archetypes 1 and 2 are most representative of the inner boundary layer, while 4 and 5 are closer to the outer boundary layer and freestream. Archetypes 2 and 5 are more present in areas with a favourable pressure gradient while 4 and 1 are present in areas with an adverse pressure gradient. Finally, archetype 6 is most prevalent in shear layers and separated regions. This indicates that the archetypes align with distinct flow mechanisms, instead of simply reflecting geometric or case-based partitioning. Moreover, the spatial distribution of the weights is smooth, reflecting gradual transitions between flow regimes rather than abrupt changes.



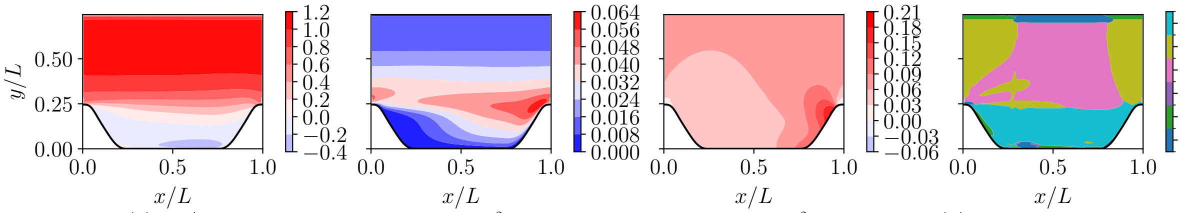
(a) U_x/U_b (b) k/U_b^2 (c) Dominant archetype

Figure 3: U_x, k and dominant archetypes on the k-ω SST simulation of the square duct.



(a) w_1 (b) w_2 (c) w_3 (d) w_4 (e) w_5 (f) w_6

Figure 4: Weights for each of the 6 archetypes on the square duct case.



(a) U_x/U_b (b) k/U_b^2 (c) $2P/\rho U_b^2$ (d) Dominant archetype

Figure 5: U_x, k and dominant archetypes on the k-ω SST simulation of the PH0.5L4071H3036 case.

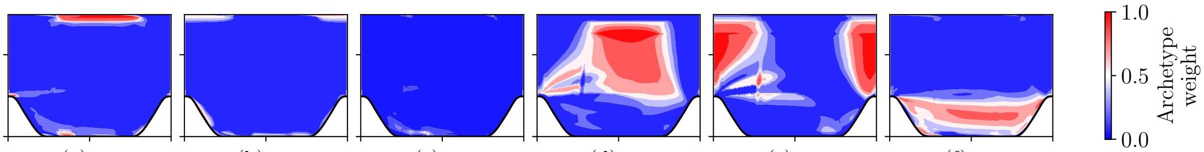
3.2. *Expert Models*

The formulation of the expert models learned by SBL-SpaRTA for b_{ij}^{Δ} is shown in Equation (7). Most experts contain the $T_{ij}^{(2)}$ tensor, which is required to predict secondary flows, with the largest contribution occurring in archetype 1. The outer boundary layer and freestream experts 4 and 5 have smaller coefficients, indicating less pronounced corrections. Archetype 3, which is present mostly in the duct case remains the same as k-ω. The experts for R_k are all equal to zero with the selected sparsity penalty. Different penalties will be examined in the future.

$$\begin{aligned}
b_{ij}^{\Delta,1} &= 12.6\,T_{ij}^{(2)}, & b_{ij}^{\Delta,2} &= (5.2 - 0.5I_1 + 1.5I_2)\,T_{ij}^{(2)} - 1.9\,T_{ij}^{(3)}, \\
b_{ij}^{\Delta,3} &= 0, & b_{ij}^{\Delta,4} &= -0.5\,T_{ij}^{(1)} + 1.9\,T_{ij}^{(2)} + 4.2\,T_{ij}^{(3)}, \\
b_{ij}^{\Delta,5} &= 1.6I_1\,T_{ij}^{(1)} + 3.0\,T_{ij}^{(2)}, & b_{ij}^{\Delta,6} &= -(4.0 + 0.9I_2)\,T_{ij}^{(3)}.
\end{aligned} \tag{7}$$

4. Results

We evaluate our model on the test cases of the turbulence challenge and compare it against the baseline k-ω SST and the intXMA model of [2]. The performance is assessed based on the mean absolute error (MAE) of the predicted



(a) w_1 (b) w_2 (c) w_3 (d) w_4 (e) w_5 (f) w_6

Figure 6: Weights for each of the 6 archetypes on the PH0.5L4071H3036 case.

velocity field at the sampling locations of the benchmark. Figure 7 summarises the results across the test cases. The proposed model consistently improves over the baseline in all but one configuration and outperforms the intXMA model in most cases. On average, KSVAE-SpaRTA achieves a scaled velocity MAE of 0.063, compared to 0.078 for intXMA and 0.104 for the k–ω SST model.

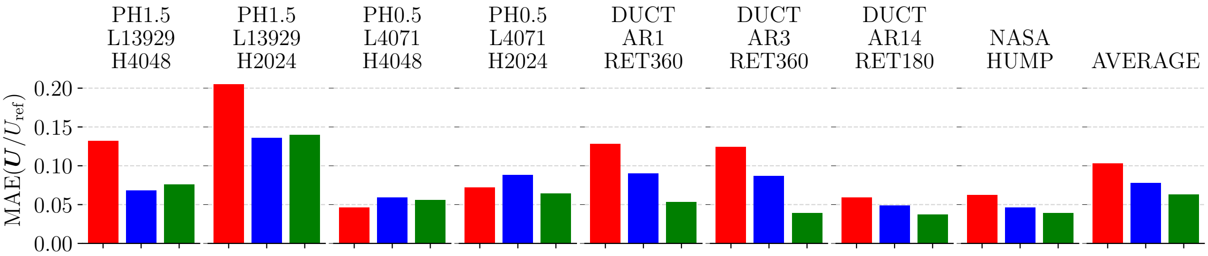


*Figure 7: **U** field MAE of the final solution for the tested models at the sampling locations of the turbulence challenge on the parametric hills cases. k-ω SST in red, the intXMA model in blue and KSVAE-SpaRTA in green.*

To further assess the generalisation capabilities of the model, we consider the three-dimensional wing-body junction configuration [15, 16], which features a complex system of interacting boundary layers and a horseshoe vortex. Figure 8 shows velocity and turbulent kinetic energy profiles on the symmetry plane in corner formed by the wing leading edge and the side wall, where the horseshoe vortex forms. The KSVAE-SpaRTA model provides an improved prediction of the recirculation region compared to both reference models, particularly in the region $-0.4 \leq x/T \leq -0.2$, where the baseline model exhibits significant discrepancies. However, the turbulence intensity is still underestimated, indicating that further improvements in modelling R_k may be required. The improved representation of the flow structure is reflected in the wall pressure distribution. As shown in Figure 9, all models tend to overpredict the strength and upstream position of the vortex, as evidenced by the greater negative error in C_p in front of the leading edge, but the error is reduced for KSVAE-SpaRTA. The impact on the pressure distribution over the wing surface remains limited, suggesting that the main benefits of the model are localised in regions of strong flow interaction.

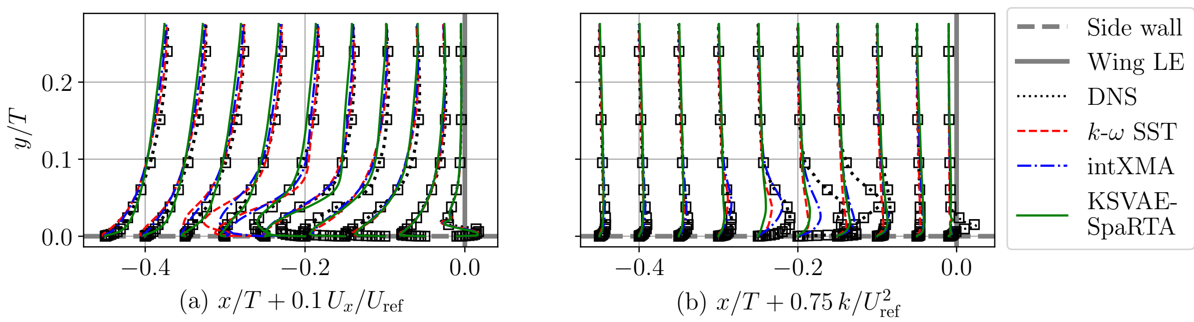


Figure 8: (a) U_x, (b) k and profiles on the wing body junction case. T stands for the thickness of the wing profile.

Overall, these results highlight the ability of the proposed approach to transfer to a complex three-dimensional configuration, despite being trained primarily on canonical two-dimensional flows. This supports the idea that the learned archetypes capture generic flow processes that are not restricted to specific geometries.

5. Conclusion

This work introduced an archetype-based mixture-of-experts strategy for interpretable data-driven turbulence closure. The central idea is to represent local flow states as convex combinations of a limited number of archetypal patterns and to associate each of these patterns with a dedicated symbolic-regression expert. This formulation leads

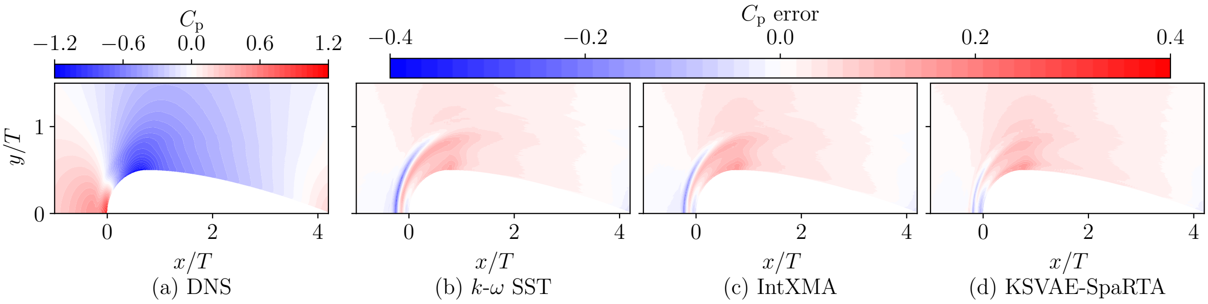


Figure 9: C_p error on wing body junction wall surface.

to a model that is both interpretable and locally adaptive. Instead of a single global correction, the resulting closure is expressed as a composition of process-specific contributions, whose weights are directly linked to the local flow character. This structure enables the model to capture the coexistence of multiple flow mechanisms within a single configuration. The proposed KSVAE-SpaRTA model demonstrates improved predictive performance on the ERCOFTAC turbulence challenge benchmark and shows encouraging transfer to a complex three-dimensional wing-body junction case, where it improves the representation of the horseshoe vortex region. These results suggest that organising data-driven closures around canonical flow states provides a promising route toward more robust and transferable turbulence models. More broadly, the present work supports the view that complex flows may be effectively described through combinations of a limited number of learned canonical processes, opening perspectives for modular and interpretable modelling strategies in fluid mechanics.

References

[1] R. Vinuesa and S. L. Brunton. "Enhancing computational fluid dynamics with machine learning". In: *Nature Computational Science* 2.6 (2022), pp. 358–366.

[2] M. Oulghelou et al. "Machine-learning-assisted blending of data-driven turbulence models". In: *Flow, Turbulence and Combustion* (2025), pp. 1–38.

[3] A. Cutler and L. Breiman. "Archetypal analysis". In: *Technometrics* 36.4 (1994), pp. 338–347.

[4] A. Srivastava and C. Sutton. "Autoencoding variational inference for topic models". In: *arXiv preprint arXiv:1703.01488* (2017).

[5] Y. Li, X. Fu, and W.-K. Ma. "Probabilistic simplex component analysis via variational auto-encoding". In: *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2024, pp. 9671–9675.

[6] S. Cherroud et al. "Sparse Bayesian learning of explicit algebraic Reynolds-stress models for turbulent separated flows". In: *International Journal of Heat and Fluid Flow* 98 (2022), p. 109047.

[7] R. McConkey et al. "The Closure Challenge: a benchmark task for machine learning in turbulence modelling". In: *arXiv preprint arXiv:2603.28884* (2026).

[8] F. R. Menter. *Improved two-equation k-omega turbulence models for aerodynamic flows*. Tech. rep. A, 1992.

[9] M. Schmelzer, R. P. Dwight, and P. Cinnella. "Discovery of algebraic Reynolds-stress models using sparse symbolic regression". In: *Flow, Turbulence and Combustion* 104.2 (2020), pp. 579–603.

[10] D. P. Kingma and M. Welling. "Auto-encoding variational bayes". In: *arXiv preprint arXiv:1312.6114* (2013).

[11] H. Kim and A. Mnih. "Disentangling by factorising". In: *International conference on machine learning*. PMLR. 2018, pp. 2649–2658.

[12] S. B. Pope. "A more general effective-viscosity hypothesis". In: *Journal of Fluid Mechanics* 72.2 (1975), pp. 331–340.

[13] J. Ling and J. Templeton. "Evaluation of machine learning algorithms for prediction of regions of high Reynolds averaged Navier Stokes uncertainty". In: *Physics of Fluids* 27.8 (2015).

[14] L. Van der Maaten and G. Hinton. "Visualizing data using t-SNE." In: *Journal of machine learning research* 9.11 (2008).

[15] D. Apsley and M. Leschziner. "Investigation of advanced turbulence models for the flow in a generic wing-body junction". In: *Flow, Turbulence and Combustion* 67.1 (2001), pp. 25–55.

[16] F. Bassi et al. "On the development of an implicit high-order discontinuous Galerkin method for DNS and implicit LES of turbulent flows". In: *European Journal of Mechanics-B/Fluids* 55 (2016), pp. 367–379.

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

Autoencoder-Based Analysis of Roughness Effects on Heat-Momentum Dissimilarity

Moritz Link[1], Simon Dalpke*[1], Ryo Koshikawa[2], Kai Fukami[2], and Alexander Stroh[1]

[1]Institute of Fluid Mechanics, Karlsruhe Institute of Technology, 76131 Karlsruhe, Germany
[2]Department of Aerospace Engineering, Tohoku University, Sendai, Japan

Keywords: roughness, convolutional autoencoder, turbulent channel flow, momentum & heat transfer

Abstract. Surface roughness strongly alters wall-bounded turbulence, breaking the Reynolds analogy between momentum and heat transfer. Although the resulting dissimilarity is well documented, the geometric features that drive it remain hidden within complex roughness topographies. We address this gap with a convolutional autoencoder (CNN-AE) that learns nonlinear, low-dimensional representations of instantaneous velocity and temperature fields. Instantaneous co-located streamwise velocity $u_1(x_i, t)$ and temperature $\theta(x_i, t)$ are extracted as two-dimensional windows from a dataset of channel flow at friction Reynolds number $Re_\tau \approx 800$ over a wide range of pseudo-random rough surfaces, curated through direct numerical simulations. The windows serve as separate single-channel inputs to a shared CNN-AE. An auxiliary multilayer perceptron is attached to the latent space to classify each embedding as belonging to velocity or passive scalar, encouraging disentanglement of momentum and heat representations. The network is trained jointly with a loss that balances reconstruction error and classification accuracy. At a compression factor of 16 the model reconstructs both u_1 and θ with a global error below 6%, while the MLP head yields a latent space in which velocity and temperature embeddings are clearly separated, as confirmed by principal-component analysis. Distances between co-located latent vectors are then correlated with precise quantitative measures of heat-momentum dissimilarity and roughness variation.

1 Introduction

In many engineering flows, surface roughness is not a minor detail but a primary factor controlling drag, energy loss and heat transfer. The roughness variety ranges from deposits or corrosion in heat exchangers to effects of manufacturing processes. While momentum and heat transfer exhibit strong similarity over hydraulically smooth walls, the presence of roughness decouples these transport mechanisms, leading to a breakdown of classical Reynolds analogy. Hence, the thermo-hydraulic performance of turbulent flow above a rough surface is heavily influenced by different roughness characteristics and its topography (Chung et al., 2021).

This study employs data-driven techniques to investigate the structural differences between streamwise velocity and temperature fields for a variety of rough surfaces in turbulent flow. Direct Numerical Simulations (DNS) are encoded into a compact latent representation, to identify characteristic dissimilarities and roughness dependent behavior in a reduced form. Following the methodology of Fukami & Taira, 2025, we utilize an autoencoder-based framework augmented by a supervised branch to compress the flow fields while enforcing physical interpretability.

2 Methodology

2.1 Roughness Generation and Direct Numerical Simulation

In this study an already existing dataset, generated by Yang et al., 2023, is utilized, where further details can be found. The dataset covers 105 minimal channel Direct Numerical Simulations

*Corresponding Author: simon.dalpke@kit.edu

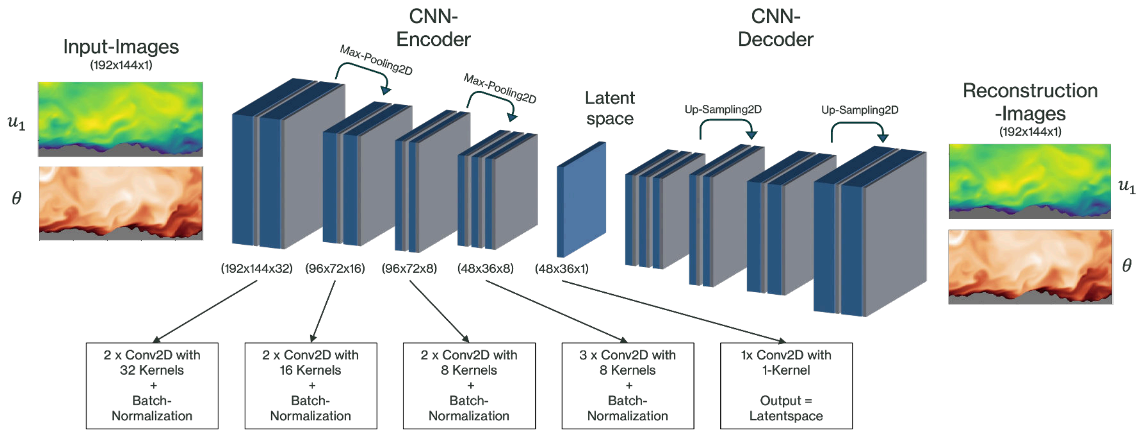


Figure 1: Summary on the architecture of the developed autoencoder. An MLP head is added to the latent space to distinguish between velocity and passive scalar input.

(DNS) of turbulent channel flow over rough walls, performed at a friction Reynolds number of $Re_\tau = 800$ and a Prandtl number of $Pr = 0.71$. Distinct roughness configurations are introduced at the top and bottom wall of the channel via an Immersed Boundary Method (IBM). The rough surfaces are artificially generated using an algorithm by Pérez-Ràfols & Almqvist, 2019 to achieve statistically controlled topographies. The fluid flow is assumed incompressible and Newtonian, with temperature treated as a passive scalar governed by the advection-diffusion equation. The corresponding governing equations are numerically solved using the pseudo-spectral solver SIMSON, providing instantaneous three-dimensional snapshots of the streamwise velocity u_1 and temperature θ fields. A key metric for analysis of the roughness influence are the so-called roughness function ΔU^+ and $\Delta\Theta^+$, which describe the vertical shift in the logarithmic region of the spatio-temporally averaged profiles compared to a smooth wall reference at matched flow conditions. These roughness functions are evaluated for each of the minimal channel simulation in the presented dataset.

In preparation for the machine learning framework, raw spectral velocity and temperature snapshots are transformed into physical space and a binary roughness mask is applied within the solid regions to suppress numerical artifacts introduced by the IBM. Two-dimensional spanwise slices are extracted from the three-dimensional domains, with spacing determined by a correlation length analysis to ensure statistical independence between samples. From these slices, near-wall windows of size $1\delta \times 0.5\delta$ are extracted for both fields at identical spatial coordinates. These are normalized by their global maximum value to account for different magnitudes. To further maintain consistent wall-attached orientation, windows from the upper wall are mirrored in the wall-normal direction, resulting in a total dataset of 42188 windows for training.

2.2 Autoencoder Setup

A nonlinear compression, based a convolutional autoencoder (CNN-AE), is employed to identify a latent representation of the preprocessed windows (Hinton & Salakhutdinov, 2006), as visualized in fig. 1. Either the preprocessed streamwise velocity window or the passive scalar window serve as single-channel input the shared CNN-AE. The chosen architecture compresses the input windows $(192 \times 144 \times 1)$ into a latent tensor of size $48 \times 36 \times 1$, corresponding to 1728 degrees of freedom and a compression ratio of $16 : 1$. The encoder part uses convolutional blocks with a hyperbolic tangent activation function followed by batch normalization and max-pooling layers

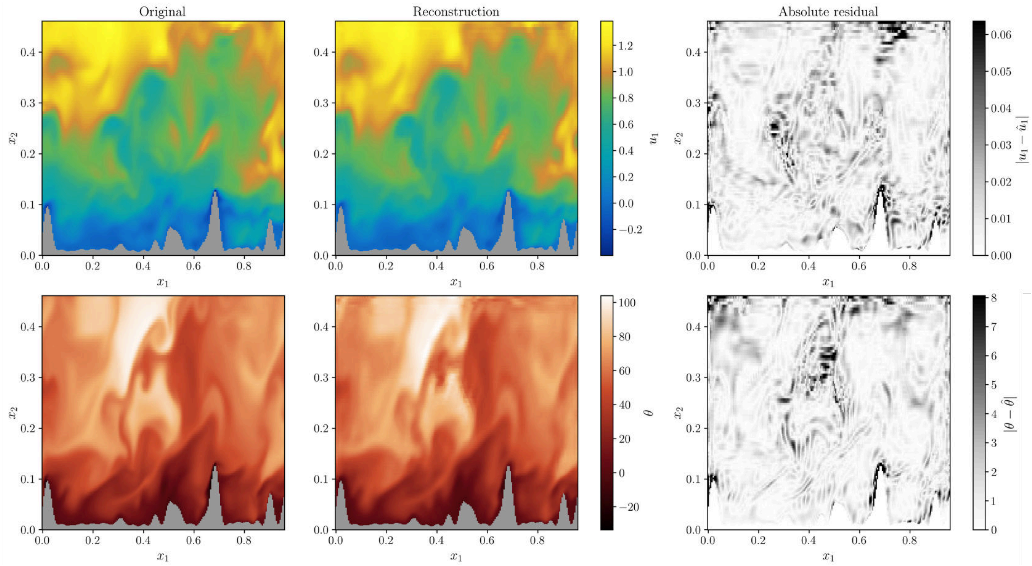


Figure 2: Reconstruction capabilities of the trained autoencoder for the streamwise velocity u_1 (top row) and temperature θ (bottom row).

for down-sampling. The decoder mirrors this encoder architecture. Training of the CNN-AE is governed by minimizing the L_2 reconstruction loss. The bottleneck dimension is selected based on a preliminary analysis of reconstruction error versus latent capacity, identifying the point where further increases in latent dimension yields marginal improvements in fidelity.

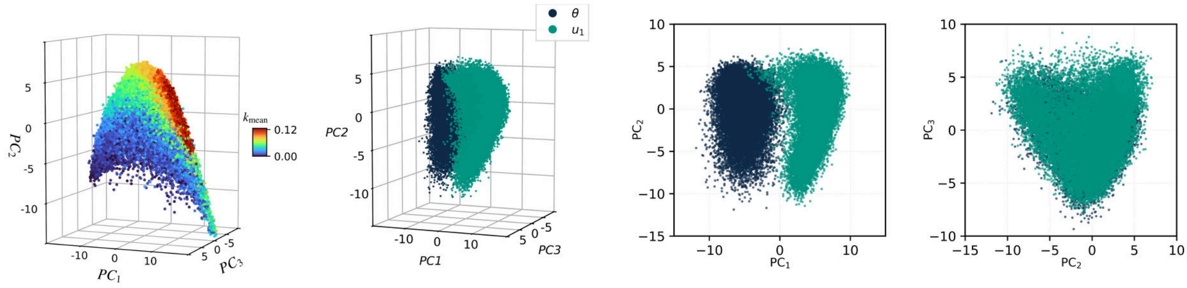
2.3 Additional Multi-Layer Perceptron head

Following Fukami & Taira, 2025 an additional Multi-Layer Perceptron (MLP) network is attached to the latent space representation, to augment the autoencoder with a supervised objective. The MLP head is connected to the flattened latent vector to classify the input field type (velocity vs. temperature) based on the latent space. Hence, the additional network part encourages the latent space to organize according to the underlying transport physics. The whole network was trained by minimizing a composite loss function $L = L_{AE} + \beta L_{BCE}$. The first term represents the L_2 reconstruction error of the autoencoder and the second term is the binary cross-entropy loss, evaluated based on the categorical input field type. The parameter β controls the trade-off between reconstruction accuracy and supervised alignment, and is selected via an L-curve analysis (Hansen & O'Leary, 1993). This procedure identifies a β value (specifically $\beta = 10^{-1}$) by analyzing the autoencoder reconstruction loss and the binary cross-entropy loss for different β-values. The chosen value provides a robust separation in the latent manifold while maintaining an acceptable reconstruction capability.

3 Results

3.1 Reconstruction Quality

The developed autoencoders provide a robust reconstruction capabilities in both configurations. In the case without the MLP head the model achieves high fidelity with global relative L_2 errors of approximately 1.57% for the velocity field and 1.94% for the temperature field. In fig. 2 an example of the autoencoder reconstruction for a selected window of streamwise velocity and



(a) Without supervision (b) With supervision and representation in different projections

Figure 3: Principal components of the latent space representation using an autoencoder without (colored by mean height) and with (colored by classification) classification supervision.

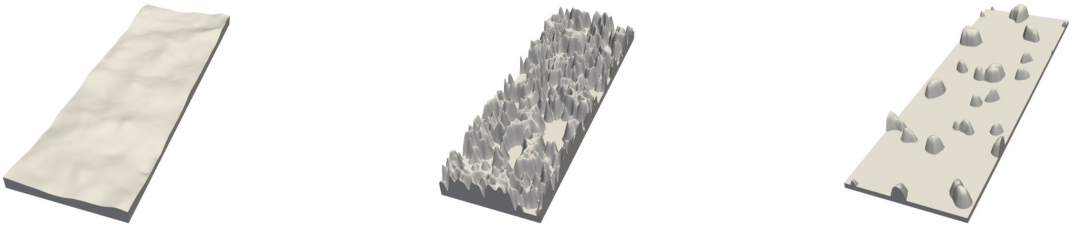
temperature is visualized. Between the original and reconstructed images high visual agreement is achieved and the autoencoder effectively capturing dominant spatial structures. Further analysis of the absolute residual highlights higher local errors in regions with high gradients, e.g., close to the rough surface where a no slip boundary condition is present. Additionally, the fine-scale features in the fluid window are less faithfully reconstructed compared to the larger structures. This is in line with the observations by Fukami & Taira, 2023, nevertheless the autoencoder successfully reconstructs the spatial organization.

Further addition of the MLP head to induce a supervision task and to constrain the latent representation, lead to a small deterioration of the reconstruction error. For the selected balance of the training loss ($\beta = 10^{-1}$), the global reconstruction errors rise to 4.88% for velocity and 6.33% for temperature. This value provides distinct separation between velocity and temperature embeddings, as the classifier network achieves an accuracy of 99.988% over the 42188 windows. This translates into only five misclassifications. While the reconstruction error represents a deterioration compared to the unsupervised baseline, the additional well-founded classification capability is introduced and both errors remain within an acceptable range. This indicates that the latent representation contains sufficient information to recover flow structures while simultaneously distinguish between streamwise velocity and temperature images.

3.2 Latent Space Analysis

For an interpretable representation of the flattened latent space of the autoencoder the Principal Component Analysis (PCA) is performed, providing an orthonormal basis based on decreasing variance. In the pure autoencoder part, PCA revealed a structured "hammock-like" manifold where different regions of the manifold correlate with geometric variations, namely the average height k_{avg} of the roughness in the window (see fig. 3a). These findings suggest that the even unsupervised latent manifold, when viewed through PCA, inherently encodes physically meaningful patterns that reveal key geometric characteristics. However, the embeddings for velocity and temperature fields remained heavily overlapping, and no clear partition is visible.

In contrast, the latent space extracted from an autoencoder with an additional MLP classification head shows a clear separation after PCA between images of streamwise velocity u_1 and temperature θ. This can be seen especially in the first principle component shown in fig. 3b. Here, the green points represent images of streamwise velocity, whereas blue points indicate images of the passive scalar θ. Comparing the first and second principal component (as in the middle graph of fig. 3b) indicates the separation enforced through the binary classification head. This separation is nevertheless a feature of the first principal component, as the visualization



(a) Case A: $\Delta U^+ = 6.01$, $\Delta\Theta^+ = 4.26$

(b) Case B: $\Delta U^+ = 10.55$, $\Delta\Theta^+ = 6.59$

(c) Case C: $\Delta U^+ = 10.51$, $\Delta\Theta^+ = 4.31$

Figure 4: Visualization of the three selected roughness topographies.

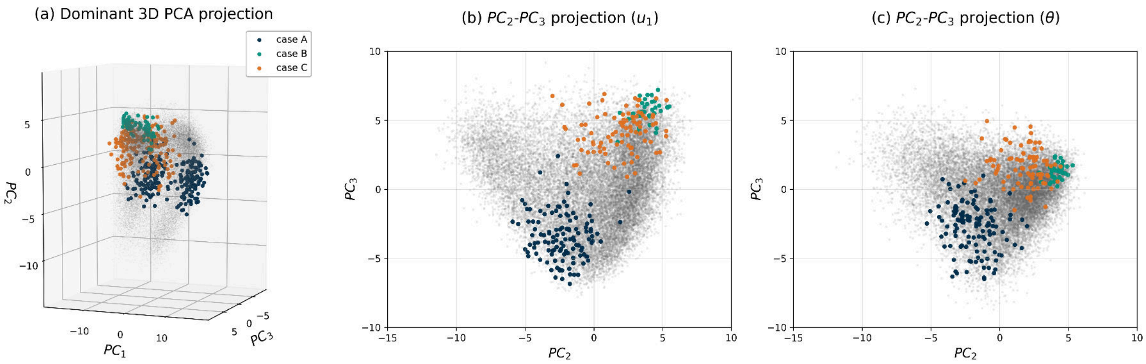


Figure 5: Principal Component representation of the latent space with visualization of the three selected surfaces.

of PC_2 vs. PC_3 do not reveal a partitioning. It should be noted, that the latent space manifold changed its shape from a "hammock-like" representation to a more cloud shape representation when adding the MLP classification head.

To further investigate the latent space manifold three distinct roughness configurations (A, B, and C, shown in fig. 4) are selected based on the analysis of the identical dataset by Dalpke et al., 2025. Case A features a moderate, hilly topography without fine-scale asperities, inducing balanced deficits of $\Delta U^+ = 6.01$ and $\Delta\Theta^+ = 4.26$. In contrast, Case B presents a dense field of sharp asperities that generates the largest thermal shift of the selected surfaces ($\Delta\Theta^+ = 6.59$) alongside a substantial momentum deficit ($\Delta U^+ = 10.55$). Case C exhibits sparse roughness elements placed on a smooth surface; notably, it produces a momentum deficit comparable to Case B ($\Delta U^+ = 10.51$) but a thermal shift similar to Case A ($\Delta\Theta^+ = 4.31$) and hence represents a strong decoupling of the different transport mechanisms. In fig. 5 the points representing the velocity and temperature fields of selected three surfaces are visualized inside the point cloud of the principal component representation of all windows. Furthermore, in this figure the streamwise velocity u_1 and the temperature θ is split in their projection onto the $PC_2 - PC_3$-plane. From a global standpoint, the three roughness topographies are clearly located in distinct regions of the projected manifold, even though the spreading of the data points differ between the surfaces. Especially, the temperature representation of case B has a low spread, compared to the other surfaces. It is furthermore interesting to note, that for the streamwise velocity the two surfaces with similar velocity roughness function are located in a similar region and clearly separated from the low ΔU^+ case. Such an observation to the roughness function cannot be identified in the projection for the temperature.

4 Conclusion

In this contribution, we present a data-driven framework to analyze the momentum and passive scalar transport in turbulent flows over rough surfaces using the latent space of an autoencoder. Based on an existing dataset of 105 Direct Numerical Simulations, we trained a shared convolutional autoencoder to reconstruct co-located streamwise velocity u_1 and temperature θ as separate single-channel inputs. The resulting principal component representation of the latent manifold indicates a learned relation to the mean roughness height but is not able to distinguish between the two separate transport mechanisms. Therefore, a classification multi-layer perceptron head was added to the latent space to further supervise the learning process. The resulting latent representation disentangles the transport modalities, and the MLP head successfully distinguishes the different fields. Furthermore, in the first three principal components of the latent space representation, selected surfaces share similar locations related to their velocity roughness function, even though a clear trend cannot be discovered, especially for the temperature roughness function.

Future work will extend the autoencoder to include the full three-component velocity field, an essential step for accurately modeling the three-dimensional flow structures near wall roughness. We also plan to explore supervised learning approaches that enforce macroscopic roughness functions as physical constraints. Since these functions are spatio-temporally averaged while the autoencoders inputs are local and instantaneous, future architectures must address this scale disparity to prevent the model from overfitting and acting as a simple 'simulation ID' classifier. Furthermore, we aim to advance this AE-architecture toward a β-Variational Autoencoder (β-VAE). Transitioning to a probabilistic framework would impose a regularized, continuous latent distribution, potentially enhancing the robustness of the representation and facilitating the generation of synthetic flow fields based on interpolated latent space variables.

References

Chung, D., Hutchins, N., Schultz, M. P., & Flack, K. A. (2021). Predicting the Drag of Rough Surfaces. *Annual Review of Fluid Mechanics*, 53, 439–471. https://doi.org/10.1146/annurev-fluid-062520-115127

Dalpke, S., Yang, J., Forooghi, P., Frohnapfel, B., & Stroh, A. (2025). Data-driven correlations for thermohydraulic roughness properties. *European Journal of Mechanics - B/Fluids*, 204355. https://doi.org/10.1016/j.euromechflu.2025.204355

Fukami, K., & Taira, K. (2023). Grasping extreme aerodynamics on a low-dimensional manifold. *Nature Communications*, 14(1), 6480. https://doi.org/10.1038/s41467-023-42213-6

Fukami, K., & Taira, K. (2025). Observable-augmented manifold learning for multi-source turbulent flow data. *Journal of Fluid Mechanics*, 1010, R4. https://doi.org/10.1017/jfm.2025.383

Hansen, P. C., & O'Leary, D. P. (1993). The Use of the L-Curve in the Regularization of Discrete Ill-Posed Problems. *SIAM Journal on Scientific Computing*, 14(6), 1487–1503. https://doi.org/10.1137/0914086

Hinton, G. E., & Salakhutdinov, R. R. (2006). Reducing the Dimensionality of Data with Neural Networks. *Science*, 313(5786), 504–507. https://doi.org/10.1126/science.1127647

Pérez-Ràfols, F., & Almqvist, A. (2019). Generating randomly rough surfaces with given height probability distribution and power spectrum. *Tribology International*, 131, 591–604. https://doi.org/10.1016/j.triboint.2018.11.020

Yang, J., Stroh, A., Lee, S., Bagheri, S., Frohnapfel, B., & Forooghi, P. (2023). Prediction of equivalent sand-grain size and identification of drag-relevant scales of roughness – a data-driven approach. *Journal of Fluid Mechanics*, 975, A34. https://doi.org/10.1017/jfm.2023.881

Data Driven Turbulence Modelling through Eddy Viscosity Adaptation and Symbolic Regression

C. Tsolakos; A.A Petrocheilos; K. Batsi; V. Asouti; K. Giannakoglou*;

Parallel CFD & Optimization Unit, School of Mechanical Engineering, National Technical University of Athens, Athens, Greece

Keywords: Eddy Viscosity Adaptation, Field Inversion, Continuous Adjoint, Symbolic Regression, Aerodynamic Shape Optimization

Abstract. In several industrial applications, simulations based on the (steady) Reynolds-Averaged Navier-Stokes (RANS) equations are no more considered as high-fidelity (Hi-Fi) ones, and unsteady (URANS) solutions, supported by, at least, the Detached Eddy Simulation (DES) variant of the Spalart-Allmaras (SA) model, should be used instead. However, the use of unsteady simulations in a shape optimization loop (ShpO) has excessive computational cost. This paper presents and assesses a method to accelerate ShpO loops, at affordable cost. Initially, a continuous adjoint-based loop performs the so-called Eddy Viscosity Adaptation, by computing an eddy viscosity field which, if embedded into the RANS equations, gives solutions which reproduce, as closely as possible, the (time-averaged) outcome of the Hi-Fi solver. Symbolic Regression (SR) is then used to get a closed form expression of the optimized eddy viscosity field, in terms of geometrical and flow quantities. This is, then, used in an adjoint-based ShpO loop, in which the adjoint equations may include terms resulting from the differentiation of the eddy viscosity formula provided by the SR. The new tool/method is demonstrated in a 2D and a 3D aerodynamic shape optimization problem.

1. Introduction

Turbulence modelling as used by RANS solvers remains essential for industrial CFD applications. However, apart from modifications or adaptations for compressibility, transition etc. there have not been significant advances in the development of turbulence models [1]. On the other hand, during the last decade, there has been extensive research into Machine Learning (ML) approaches for modelling turbulence, by integrating data-driven techniques with physics-based models, particularly within the Field Inversion and ML framework [2]. This employs an inverse-modelling strategy in which discrepancies between RANS model predictions and Hi-Fi reference data, such as Large Eddy Simulation (LES), Direct Numerical Simulation (DNS), or experimental measurements, are minimized through the introduction of a spatially varying correction field. Efforts focus on optimizing parameters of widely used turbulence models, such as the production term of the Spalart-Allmaras one in [3], or the destruction term of k–ω model in [4], or performing field inversion directly on the eddy viscosity field in [5].

This paper advances data-driven turbulence modelling by proposing a two-step methodology, and implementing this into a ShpO loop. In the first step, the continuous adjoint method is used to compute the eddy viscosity field (Eddy Viscosity Adaptation, or EVA) which, when used in the standard RANS equations solver by deactivating the turbulence model, reproduces selected outcomes of a dependable Hi-Fi simulation. The so-computed eddy viscosity field can be used in an aerodynamic ShpO loop, by just freezing it (as in [5]). Alternatively, SR can be employed to obtain a closed-form expression of the computed (by EVA) eddy viscosity field. The availability of a closed-form expression allows its differentiation, to modify the adjoint equations that will support the ShpO loop.

Herein, the EVA approach is initially used to reproduce the DNS-computed flow field over 2D periodic hills and is compared against the outcome the RANS supported by either the k-ε or the Spalart-Allmaras (SA) turbulence models. The ShpO of the 3D Ahmed body follows. A closed-form expression of the computed eddy viscosity obtained by SR is used to dynamically update the eddy-viscosity field in each optimization cycle. Also, the SR expression is differentiated and its contributions are considered by the adjoint solver, as explained before. This enables the ShpO without resorting to unsteady DES simulations, and the formulation of the unsteady adjoint equations which typically present significant challenges related to the storage of time-accurate flow fields during the backward-in-time integration [6].

2. Adjoint-based EVA

The time-averaged velocity field () computed by the Hi-Fi flow model is the field to be reproduced by the solution of the RANS equations including an appropriate eddy-viscosity ν_t field, to be computed by the adjoint-based optimization (EVA), by minimizing:

(1)

where might be either the whole computational domain , or part of it. For incompressible fluid flows, the RANS equations are:

, (2)

where are the velocity components, the static pressure divided by density and the constant bulk viscosity. To build the continuous adjoint method to support the EVA process, the Lagrangian is defined and differentiated with respect to (w.r.t.) the unknown field, yielding:

(3)

The mathematical development of equation (3) leads to volume and surface (S) integrals containing derivatives of the flow variables w.r.t. . Setting the multipliers of these derivatives, in all field integrals, to zero yields the field adjoint equations:

, (4)

A similar treatment of the boundary integrals yields the adjoint boundary conditions; at the inlet and solid walls, whereas at the outlet a Robin type condition is imposed on the normal adjoint velocity component. Once all these have been satisfied, the EVA sensitivity derivatives are:

(5)

These derivatives are used to update the field which can, then, be used to simulate the flow either by injecting it, as raw data (RANS/rEV), into equations (2), or by building a closed-form expression for , using SR, to be used in the same equations (RANS/cfEV).

3. Applications – Discussion

Periodic Hills

The first case deals with the flow over 2D periodic hills, a known turbulence model verification benchmark, in which the wall geometry is represented by piecewise third-order polynomial functions [7]. The channel height is and the crest-to-crest distance is , where is the hill height. Periodic (along the left/right boundaries) and no-slip (solid walls) boundary conditions are imposed; the flow is driven by a uniform body force implemented into the x-momentum equation to ensure that , with the bulk velocity at the crest equal to . In this case, the reference flow field results from a DNS database, [7], interpolated to the coarser 2D mesh, with ~14.7K cells, used hereafter. The time-averaged DNS outcome exhibits a large reverse flow region downstream of the hill, as clearly indicated by the plotted streamlines, figure 1a. In the same case, the k-ε model (RANS/k-ε) under-predicts, in contrast to the SA model (RANS/SA) that over-predicts, the reverse flow area (figures 1b, 1c). In both models, at the first cell centers off the solid wall .

The EVA process was firstly tried for equation (1) by forming using 60 mesh cells (half of them are those with the largest RANS/k-ε and DNS differences, and the rest being randomly selected); in this case, the optimization reduced by 65%. Then, EVA was repeated for the entire domain (, achieving a reduction by 95%, and producing a field which, if used in the RANS/rEV code, satisfactorily reproduces the Hi-Fi results, as shown in Figure 1d.

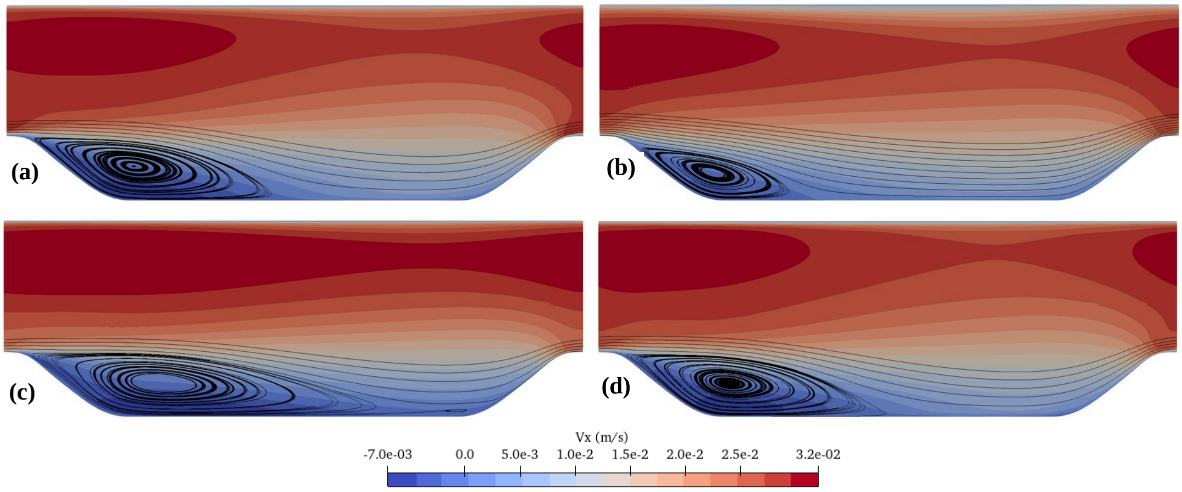


Figure 1. *Periodic Hills: Streamwise () velocity field and streamlines (black lines) obtained using (a) DNS [7], (b) RANS/k-ε, (c) RANS/SA, and (c) RANS/rEV (with J computed over Ω).*

This is confirmed by the velocity profiles plotted in figures 2a, 2b. In figure 2c, skin-friction coefficient () distributions of the along the bottom wall are compared; for the DNS, flow reattachment occurs at , to be compared with for RANS/rEV. The RANS/k-ε solution predicts reattachment much earlier (, while the RANS/SA further downstream (). Overall, the RANS/rEV model satisfactorily captures the DNS velocity profiles near the walls, indicating its suitability for supporting an adjoint-based ShpO loop (see next case).

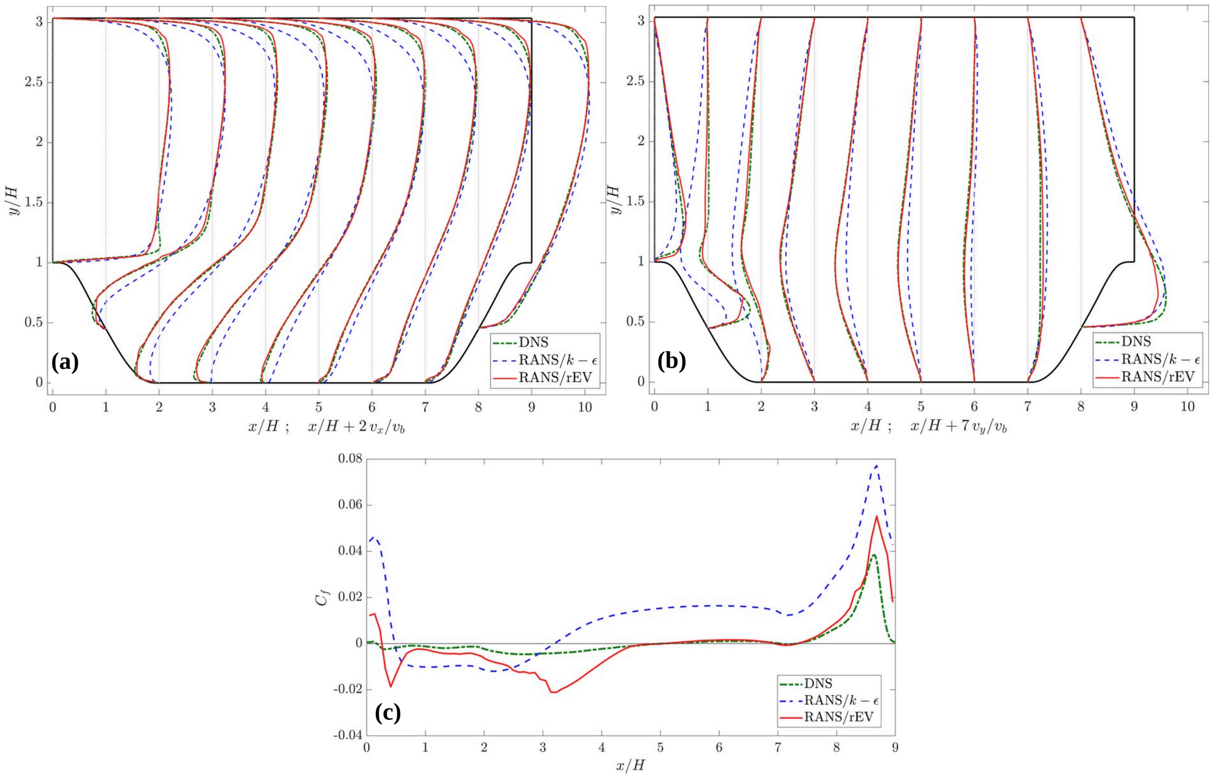


Figure 2. *Periodic Hills: Comparison of (a), (b) velocity profiles and skin friction coefficient (c) across the lower wall for the different models. DNS data are provided by [8].*

Ahmed body

The second study considers the 35° slant-angle Ahmed body [9] in a wind-tunnel-like computational domain with ~2.2M cells, exposed to a uniform inlet velocity (). The inlet and outlet are located 5 and 14 body lengths upstream and downstream, respectively, while the side and top boundaries are placed 4 and 8 body lengths from the body to reduce blockage effects. A no-slip condition is applied on the ground and body surfaces, wall functions are used near the solid walls, and symmetry boundary conditions are imposed on the side boundaries.

The Hi-Fi (reference) solver is the DES/SA model [10]. A simulation of 5 sec was carried out and the time-averaged results of the streamwise velocity component field, computed over the last 3 sec, is shown in figure 4a. These are compared with the steady-state solution of the RANS/SA of figure 4b. The two fields differ significantly, in accordance with differences in the streamwise velocity profiles near the slant area, figure 5a, 5b. The time averaged DES/SA results are very close to the experimental data (from [11]), while the RANS/SA ones differ. The EVA process, minimizing (equation (1)), follows, resulting in a 5% decrease in , compared to the value the same integral takes on using the RANS/SA velocities. The corresponding streamwise velocity field is shown in figure 4c; it is clear that the velocity profiles match the experimental measurements, as shown in figure 5 (red line).

An SR model was then trained on the field resulted from the EVA process, using the PySR library [12]. The algorithm's input variables are flow and geometric quantities, such as coordinates, local velocity components, pressure, strain rate, vorticity and distance from the nearest wall. In the obtained expression, is proportional to the distance of the cell from the nearest wall and the streamwise velocity. The implementation of this expression into the software solving the mean-flow equations (RANS/cfEV) reproduced the velocity profiles with accuracy comparable to those directly obtained from EVA, as shown in figures 4d and 5.

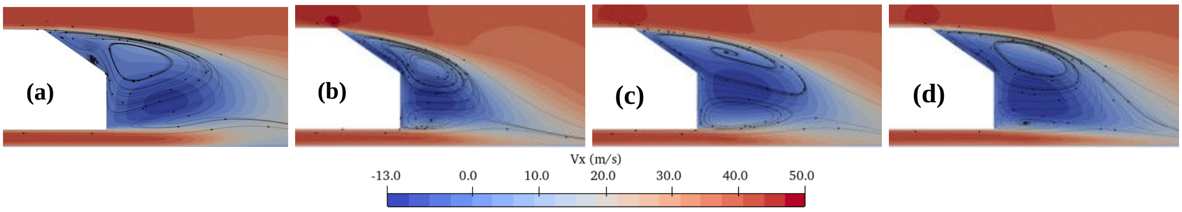


Figure 4. *Ahmed body: Comparison of the velocity field and streamlines (black lines) at the symmetry plane obtained using (a) the time-averaged DES/SA (b) RANS/SA, (c) RANS/rEV, and (d) RANS/cfEV.*

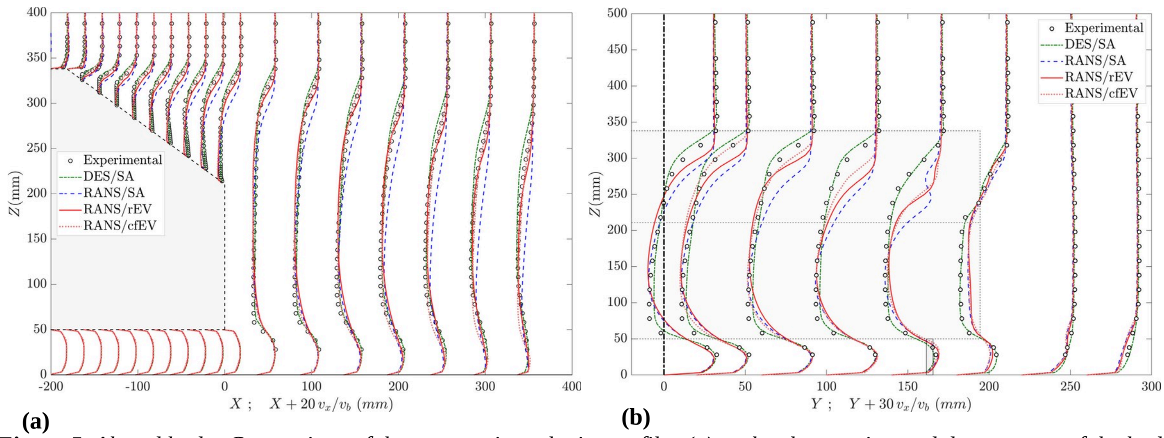


Figure 5. *Ahmed body: Comparison of the streamwise velocity profiles (a) at the slant region and downstream of the body (over the symmetry plane) and (b) at 80mm downstream of the body (rear view).*

The drag coefficient of the RANS/SA model is , while the reference value resulted by the DES/SA is . The RANS/rEV model improved the prediction giving , not so different than that of the RANS/cfEV model ().

The ShpO of the Ahmed body, targeting min. was, then, performed. The design variables are the coordinates of control points of two NURBS morphing lattices, controlling the upper and lower part of the slant. Its control points initially arranged along a line perpendicular to the symmetry plane were obliged to remain on a straight line with the same property. The optimization employed the continuous adjoint method. In case a, ShpO was based on the RANS/SA model, by also solving the adjoint to the turbulence model equation [13], and this resulted in the geometry shown in figure 6a. Then, the primal solver was replaced by either RANS/rEV (case b) or RANS/cfEV (case c), exploiting the outcome of the same EVA process (without or with processing it with the SR tool), figures 6b and 6c; in both of them, in the ShpO loop the adjoint code worked with frozen eddy viscosity fields. Over and above, a last ShpO run was carried out, still based on the RANS/cfEV primal solver, this time though by fully differentiating the closed-form expression of , adding thus extra terms to the adjoint equations (case d, figure 6d). All optimization cases were carried out for 4 cycles, with the exception of case (b) which terminated/failed after just 2 cycles, since the frozen raw assumption proved unreliable for large geometrical changes. In all cases the optimization tends to displace the body by creating a spoiler at the starting point of the slant.

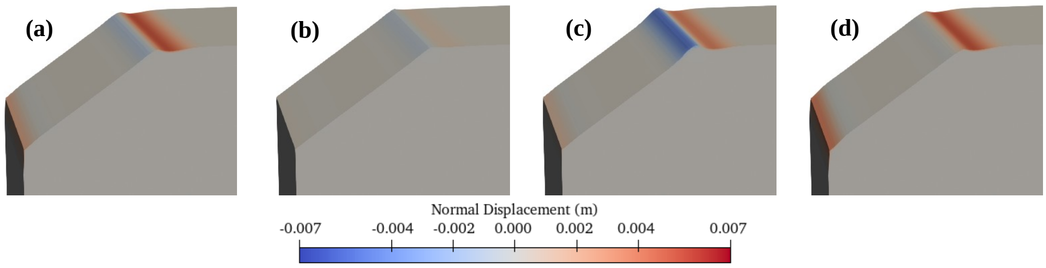


Figure 6. *Comparison of the final Ahmed body slant shapes resulted from the ShpO using the (a) fully differentiated RANS/SA, (b) adjoint to RANS/rEV, (c) adjoint to RANS/cfEV and (d) fully differentiated RANS/cfEV.*

To verify the accuracy of the different approaches, the final geometries in each case were re-evaluated on the DES/SA Hi-Fi solver. The corresponding velocity fields behind the body, on the symmetry plane, are shown in Figure 7. Case (a) yields , marginally better than that of the baseline geometry. Case (b) provides a after 1 cycle, a 10% improvement showing the EVA's efficiency for small geometrical changes. Case (c) achieves after 4 cycles, benefiting from the SR-based closed-form expression of in the primal problem, which takes geometry changes into account during the ShpO. Finally, case (d) gives .

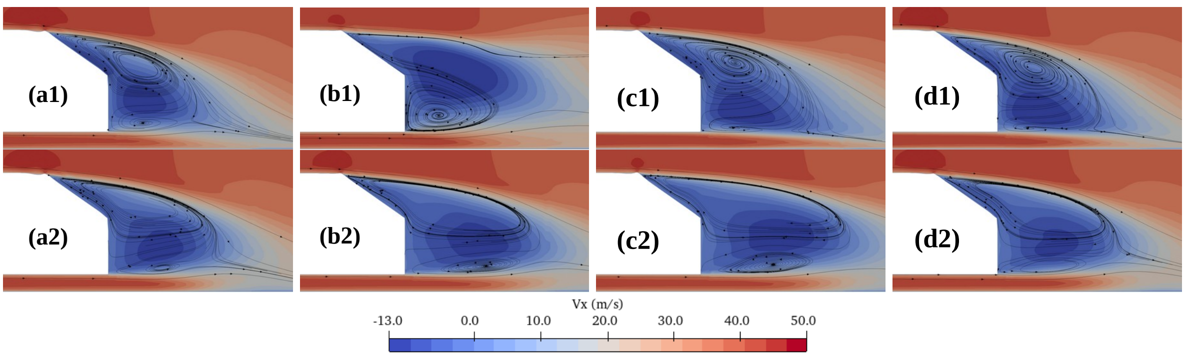


Figure 7. *Streamwise velocity field and streamlines (black lines) on the symmetry plane, computed by the same model (DES/SA) for the final geometries resulted from the ShpO using the a) fully differentiated RANS/SA, (b) adjoint to RANS/rEV, (c) adjoint to RANS/cfEV, and (d) the fully differentiated RANS/cfEV. For each case (x), (x1) corresponds to the original turbulence model, while (x2) corresponds to the reevaluation using DES.*

4. Conclusions

The EVA process, based on continuous adjoint, may compute eddy viscosity fields which, when injected into the mean-flow equations allow a steady flow solver to replicate the outcome of a Hi-Fi flow simulation, such as DNS or DES/SA. In this article, the difference in the computed velocities throughout the flow field is the point of reference to measure the similarity of two flow fields resulting from different solvers; this can certainly by replaced by other field or surface integrals. Having computed such an eddy viscosity field, this can be used into a ShpO loop to compute new shapes with better performance. During the adjoint-based ShpO loop, the eddy viscosity can be frozen to the field computed by the EVA process or, alternatively, can be expressed as a function of geometry and flow quantities, using SR, so as this to adapt to newly generated geometries and their flowfields. Having the eddy viscosity in closed-form expression allows its differentiation and, thus, affects the adjoint equations. This study demonstrated the effectiveness of this approach and revealed deviations when the optimized solutions are re-evaluated on the dependable Hi-Fi tool.

Acknowledgments

Part of this work has been carried out during the internship of the first author at Toyota Motor Europe, supervised by A. Delacroix and G. Smyrlis.

5. References

[1] A. Wissink, A. Cary, N. Wyman, J. Hodges, B. Gropp, N. Georgiadis, B. Smith, Z. J. Wang and N. Wukie ‘CFD Vision 2030 Roadmap Update 2024’, https://doi.org/10.2514/6.2021-2726

[2] E. J. Parish and K. Duraisamy ‘A paradigm for data-driven predictive modeling using field inversion and machine learning’, J. Comput. Phys., 305:758–774, 2016, http://dx.doi.org/10.1016/j.jcp.2015.11.012

[3] Y. Nishi, A. Krumbein, T. Knopp, A. Probst and C. Grabe ‘On the Generalization Capability of a Data-Driven Turbulence Model by Field Inversion and Machine Learning’, Aerospace, 11, 592, 2024, http://dx.doi.org/10.3390/aerospace11070592

[4] C. Wu, S. Zhang and Y. Zhang ‘Development of a Generalizable Data-Driven Turbulence Model: Conditioned Field Inversion and Symbolic Regression’, AIAA J., 63:687–706, 2025, http://dx.doi.org/10.2514/1.J064416

[5] G. Alessi, T. Verstraete, L. Koloszar, B. Blocken and J. van Beeck ‘Adjoint shape optimization coupled with LES-adapted RANS of a U-bend duct for pressure loss reduction’, Comput. Fluids, 228:105057, 2021, http://dx.doi.org/10.1016/j.compfluid.2021.105057

[6] A.-S. Margetis, E. Papoutsis-Kiachagias and K. Giannakoglou ‘The continuous adjoint to the incompressible (D)DES Spalart-Allmaras turbulence models’, Comput. Fluids, 284:106439, 2024, http://dx.doi.org/10.1016/j.compfluid.2024.106439

[7] H. Xiao, J.-L. Wu, S. Laizet, and L. Duan ‘Flows over periodic hills of parameterized geometries: A dataset for data-driven turbulence modeling from direct simulations’, Comput. Fluids, 200:104431, 2020, https://doi.org/10.1016/j.compfluid.2020.104431

[8] B. Krank, M. Kronbichler, and W. A. ‘Wall Direct Numerical Simulation of Flow over Periodic Hills up to Re = 10,595’, Flow, Turbulence and Combustion, 101(2):521–551, 2018, https://doi.org/10.1007/s10494-018-9941-3

[9] S. R. Ahmed, G. Ramm and G. Faltin ‘Some Salient Features of the Time-Averaged Ground Vehicle Wake’, SAE Trans, 93:473-503, 1984, https://doi.org/10.4271/840300

[10] P.R. Spalart, S. Deck, M. L. Shur, K. D. Squires, M. K. Strelets and A. Travin ‘A New Version of Detached-eddy Simulation, Resistant to Ambiguous Grid Densities’, Theor. Comput. Fluid Dyn., 20:181-195, 2006, https://doi.org/10.1007/s00162-006-0015-0

[11] H. Lienhart, C. Stoots, and S. Becker. ‘Flow and turbulence structures in the wake of a simplified car model (Ahmed modell)’, book chapter pp 323-330, New Results in Numerical and Experimental Fluid Mechanics III, Springer Berlin Heidelberg, 2002; doi:10.1007/978-3-540-45466-3_39

[12] M. Cranmer 'Interpretable Machine Learning for Science with PySR and SymbolicRegression.jl', 2023, 10.48550/arXiv.2305.01582.

[13] E. Papoutsis-Kiachagias and K. Giannakoglou. 'Continuous adjoint methods for turbulent flows, applied to shape and topology optimization: Industrial applications', Arch. of Comput. Meth. in Eng., 23(2):255-299, 2014, https://doi.org/10.1007/s11831-014-9141-9