Towards a unified turbulence model through multi-objective learning

Zhuo-Ran Liu[1,2], Hao-Chen Wang[1,2], Zhuo-Lin Zhao[1,2], Heng Xiao[1,2,*]

[1]Stuttgart Center for Simulation Science, University of Stuttgart, Stuttgart 70569, Germany
[2]Institute of Aerospace Thermodynamics, University of Stuttgart, Stuttgart 70569, Germany

[*]Corresponding author: heng.xiao@simtech.uni-stuttgart.de

Abstract

Turbulence is a central challenge in classical physics and a critical barrier to accurate flow prediction in climate, aerospace, and energy systems. Despite the widespread reliance on Reynolds-averaged Navier–Stokes (RANS) solvers in industrial simulations, existing turbulence models lack the generalizability to handle diverse regimes, such as separation, secondary flows, and free-shear flows, without manual tuning or switching. We propose a unified data-driven turbulence modeling framework based on multi-objective learning. The goal is to achieve Pareto-optimal performance across heterogeneous flow datasets, each representing distinct mechanisms and quantities of interest. The resulting *unified foundation model* employs a parallel tensor basis neural network with automatic balancing and internal branching to adapt across flow regimes without explicit switching. The parallel architecture enables explicit regularization to promote model parsimony, while the tensor-basis formulation preserves physical symmetries. Trained on five representative flows, the model is evaluated on 27 test cases spanning attached, separated, and secondary flows, as well as two realistic three-dimensional flows of industrial relevance. It improves or matches the performance of the baseline k–ω model in all cases. For specific applications, we show that *specialist models* trained on tailored datasets can further improve accuracy in challenging configurations, such as three-dimensional diffuser flows common in gas turbine aerodynamics, which exhibit simultaneous separation and secondary flows. These results demonstrate that a generalized, deployable turbulence model unifying multiple flow mechanisms within a single architecture is achievable. This work marks significant progress toward unified turbulence modeling for scientific and industrial applications.

Significance

Turbulence remains one of the last grand challenges of classical physics, yet accurate modeling is crucial for applications that shape our society, from weather and climate to aircraft and power plants. Decades of efforts have yet to produce a turbulence model that can accurately and robustly predict turbulent flows across the diverse flow regimes found in nature and industry. We present a machine-learning framework that learns a single, unified turbulence model by balancing multiple competing objectives in different flow mechanics and quantities of interest. The resulting model captures diverse flow mechanisms without manual tuning and improves accuracy in both trained and unseen flows, including the realistic three-dimensional flows. Our results mark a step toward a generalized, data-driven turbulence model that is predictive and deployable for real-world applications.

Introduction

Turbulence is among the last unresolved problems of classical physics Kim and Leonard (2024). Its presence is ubiquitous in natural and engineered flows from sub-meter to planetary scales. Accurate prediction of turbulent flows has far-reaching impacts on modern society, ranging from reducing uncertainties in climate projections and improving fidelity in extreme weather events Qi and Majda (2020) to designing safer and more efficient mission-critical systems such as aircraft engines and power plants Spalart and Venkatakrishnan (2016). The chaotic and multi-scale nature of turbulence Pope (2000) is responsible for critical physics in these systems, such as momentum and heat transport in planetary boundaries and flow through engine turbines and pollutant transport in rivers and oceans. Despite the importance of turbulent transport in such processes, our current modeling of such effects in state-of-the-art simulations is still crude. For example, in weather forecast models, the vertical turbulent mixing is still represented as one-dimensional *parameterization* schemes known as the planetary boundary layer scheme Zhu et al. (2025). In industrial computational fluid dynamics, Reynolds-averaged Navier–Stokes (RANS) solvers are still the workhorse tool for routine simulations Slotnick et al. (2014), where the "averaged effects" of the turbulent fluctuations are represented with *turbulence models* Xiao

and Cinnella (2019); Spalart (2015). This is because these applications require either long-time integration or many simulations for design optimization or uncertainty quantification. It becomes prohibitively expensive to fully resolve all the turbulence scales (as in Direct Numerical Simulations) or even to partially resolve them (as in Large Eddy Simulations) Shen et al. (2024). Despite the growth of computational resources, flow solvers based on pure modeling or parameterization of the turbulence (referred to as "turbulence modeling" collectively hereafter) will remain as backbone in many fields in the years to come. Therefore, the theoretical foundation and practical development of turbulence will continue to have societal and scientific importance.

Pursuit of a Universal Constitutive Relation

More than half a century ago, Lumley Lumley (1970) proposed the existence of a turbulent constitutive relation that maps the mean velocities $\boldsymbol{u}$ to the Reynolds stresses $\boldsymbol{\tau}$. From a philosophical perspective, a constitutive relation is determined phenomenologically from data, constrained by mathematical invariance and symmetry, and thus transferable to similar types of flows in other geometries. Mathematically, they can be expressed as $\boldsymbol{\tau}(\boldsymbol{x}) = \mathcal{F}\{\boldsymbol{u}(\boldsymbol{x}+\boldsymbol{\epsilon})\}, \boldsymbol{\epsilon} \in \Omega$, where the Reynolds stress $\boldsymbol{\tau}$ at point $\boldsymbol{x}$ depends on the mean velocity $\boldsymbol{u}$ in its vicinity Ω. The relation shall be independent of initial or boundary conditions of the Reynolds stress $\boldsymbol{\tau}$ and thus can be expected to hold in regions far removed from solid boundaries, such as free-shear layers and outer portions of boundary layer flows. Through analysis of experimental data, Lumley established that such a constitutive relation exists for homogeneous shear flows and homogeneous strain, both of which behave analogously to viscoelastic fluids with an increasing relaxation timescale Lumley (1970). Despite the conceptual depth, this insight had a limited impact on the subsequent development of turbulence modeling. While numerous models have been developed since then, none have achieved the generality envisioned by Lumley. This is partly because simplicity is as important as accuracy in industrial flow simulations, which constrains the search for constitutive relations, if any, to local ones. That is, the vicinity Ω shall contain only the immediate neighbors of point $\boldsymbol{x}$. Today, simple, empirically tuned models such as Spalart–Allmaras (SA) and k–ω SST models remain dominant in industrial practice Xiao and Cinnella (2019). The current models combine transport equations for turbulent scales with a linear constitutive relation to estimate the Reynolds stress. In the past decades, numerous linear, quadratic Tamaki and Kawai (2024); Shih (1993), and even cubic models Craft et al. (1996) have been proposed in the literature. Theoretical analysis indicates that the most general eddy-viscosity model requires 10 tensor

bases to form a quintic model Pope (2000). Since the coefficients of these bases are unknown, the tensor basis neural network is designed to learn them while preserving the generic frame invariance Ling et al. (2016). So in principle, given enough data (mean velocity field and corresponding Reynolds stress field, referred to as direct data in the context of learning turbulence model), it is possible to verify the existence of a universal turbulent constitutive relation. And if such a relation does exist, the neural network representation of such a relation can be learned. However, so far this venue has been rarely explored. A major difficulty is the lack of comprehensive data, particularly direct data that can be used to learn the model (i.e., Reynolds stresses).

Universality Versus Unification

The past few decades of experience in industrial turbulence simulations strongly indicate that a universal turbulence model that is simple and local probably does not exist. Turbulence in general is intrinsically a nonlocal, non-equilibrium phenomenon, and the averaging process in obtaining the mean flow equations loses information Duraisamy et al. (2019). However, it is a near-consensus in the turbulence modeling community that a monolithic model that seamlessly adapts to a variety of flow mechanisms and geometries (e.g., attached boundary layers, massive separation, and secondary flows) without manual zoning, switching, or blending is highly desirable Rumsey and Coleman (2022); Spalart (2023). In this work, we refer to a model that unifies multiple flow mechanisms as a *unified model*. A unified model that accommodates many flow mechanics, similar to a foundation model in artificial intelligence, is highly desirable. It could address industrial applications such as aircraft aerodynamics or turbine engines, where multiple mechanisms coexist without clear spatial distinctions and interact with each other. Developing such a *unified foundation model* is the main thrust of this work. However, even a unified model to a limited degree, referred to as *specialist model* here, e.g., one that can treat attached boundary layers, separated and secondary flows simultaneously, would be valuable too Rumsey et al. (2022); Wu and Zhang (2023); Tamaki and Kawai (2024).

Learning a Unified Model: Successes and Challenges

If a simple, universal turbulent constitutive relation does not exist, then how can one expect to find a unified model for multiple flows defined above? The key lies in training with a diverse dataset featuring multiple flow mechanisms. A number of strategies have been pursued with varying degree of successes. Multi-case training with symbolic model representation yields models that perform well across several wall-bounded flows, which are among the first attempts towards a unified

model. Fang et al. (2023); Waschkowski et al. (2022). The strategy of "unification through aggregation" relies on expert models tailored to individual flow mechanisms, with a classifier assigning models to each spatial location during prediction Cherroud et al. (2025); Buchanan et al. (2025). Yet another strategy toward a unified model is offering users a generalized model to tune for each flow, referred to as generalized k–ω model Menter and Matyushenko (2025). Similar efforts towards a generalizable closure have also emerged in wall-modeled large eddy simulations, where additive learning captures different flow mechanisms Zhang et al. (2025), and a building-block strategy leverages simple flows to construct unified closures Lozano-Durán and Bae (2023); Arranz et al. (2024). However, to this day, a unified RANS turbulence model that can seamlessly handle coexisting, vastly different flow mechanics in a single flow (e.g., flows with massive separation and swirling simultaneously) does not yet exist. The difficulty lies in the fact that different flow regimes impose *conflicting requirements* on the model: adjustments that improve one regime can degrade another. Derivative-free multi-objective optimization methods Deb et al. (2000) are generally inefficient, which limits the scalability of model training. And symbolic models, while interpretable, are often too restrictive to reconcile such conflicts. As summarized in the NASA Langley Turbulence Modeling Workshop summary Rumsey and Coleman (2022), after a decade of intense research, the community has yet to produce a data-driven turbulence model that exceeds current models in terms of predictability, generality, and robustness.

Unification Through Multi-objective Learning

In this work we pursue a strategy of "unification through balancing and branching". By exploiting the expressive power of neural networks, which scales exponentially with depth, we represent a nonlinear eddy-viscosity model (a local turbulence model in its most general form) using a parallel tensor basis neural network Ling et al. (2016), which ensures Galilean and frame invariance. This flexibility offered by neural networks allows a single model to adapt implicitly to local flow mechanisms without explicit switching or manual intervention. We use a multi-objective learning method to train such a model by seeking Pareto-optimal trade-offs among objectives such as velocity, skin friction, lift and drag from a number of representative flows (five flow mechanisms are selected in this study). This leads to a unified foundation model that can handle a wide range of flows, which is then integrated to the Reynolds-averaged Navier–Stokes equations solver OpenFOAM as in our previous works Zhang et al. (2022, 2023).

We compiled a comprehensive list of benchmark flows in the literature, including both the classical datasets Kim et al. (1987) and our own recent data specifically designed for machine learning of turbulence models Xiao et al. (2020), totaling 32 cases with five identified as training set and 27 as testing set. For each case, we carefully identified the appropriate quantities of interest for training and evaluation, which requires the understanding of the underlying physics and the unique challenges of each flow. This breadth and rigor provide the foundation for developing a unified turbulence model and establish a benchmark for future studies. Unlike weather forecasting, which relies on a single physical system with abundant data Bi et al. (2023), or genomics, where massive datasets are available Callaway (2025), turbulence involves diverse flow mechanisms and limited accessible data. Our dataset is therefore a critical step toward bridging this gap and enabling substantial progress in turbulence modeling powered by machine learning.

However, learning turbulence models from large datasets poses three main challenges: achieving a flexible yet robust model representation, learning from sparse and indirect observations, and balancing multiple objectives across diverse flows. For the first challenge, although the tensor basis neural network can represent the general eddy-viscosity model by using a single network to predict coefficients for all tensor bases, regardless of their degree of nonlinearity, such unconstrained forms often compromise stability and generalizability. To address this, we incorporate structure into the model representation through physics-informed regularization Tian et al. (2023), which keeps the model close to the baseline Parish and Duraisamy (2016); Holland et al. (2019) and favors linear terms over nonlinear ones. This is achieved with a parallel tensor basis neural network architecture that separates linear and nonlinear components, allowing more flexible control of excessive nonlinearity. The second challenge arises because most experimental datasets provide only indirect measurements Zhang et al. (2022), such as velocities, lift, or drag, rather than full-field Reynolds stresses, which serve as the direct training data for turbulence models Zhang et al. (2023b). Since learning from such indirect observations usually requires a differentiable or adjoint-enabled RANS solver Holland et al. (2019); Duraisamy (2021); Michelén Ströfer and Xiao (2021), often unavailable in practice, we employ an regularized ensemble Kalman method Kovachki and Stuart (2019); Zhang et al. (2020). This method updates network weights by correlating model predictions with observation mismatches, thereby avoiding explicit gradient calculation. The third challenge involves balancing multiple objectives, as standard machine learning approaches typically optimize a single aggregated loss, making it difficult to manage trade-offs across flows and target quantities. We address this with a gradient-based multi-objective learning strategy Désidéri (2012); Jaggi

(2013) that seeks a Pareto-optimal solution across all objectives. A solution is Pareto-optimal if no objective can be improved without degrading at least one other. The method computes a weighted average of gradients, identifies the most conflicting directions, and reconciles them by blending with an analytically determined ratio found via line search. This process is repeated until conflicts are resolved or a fixed number of iterations is reached, minimizing interference among objectives. Through this iterative conflict resolution, the resulting model generalizes robustly across diverse flows and quantities. Although formulated in terms of explicit gradients, the approach naturally extends to updates obtained from the regularized ensemble Kalman method.

Therefore, our framework delivers robust, generalizable predictions across diverse flow phenomena without the need for manual zoning or handcrafted blending, while scaling efficiently to multiple competing objectives. By combining neural network expressiveness, indirect data learning, physics-informed regularization, and gradient-based multi-objective optimization, this framework moves toward turbulence models that are accurate, generalized, and practical for large-scale engineering applications. The learned model in this study is more comprehensive than those presented in the literature so far Rumsey and Coleman (2022); Rumsey et al. (2022); Cherroud et al. (2025); Fang et al. (2023). More importantly, the proposed methodology is scalable to a large number of flows and target quantities of interest up to 40 objectives Sener and Koltun (2018), offering the potential of being extended to unify all the benchmark flow mechanics and to handle truly realistic industrial flows.

Results: Unified Foundation Model

We developed a unified turbulence modeling framework that is trained once and generalizes well across a broad range of flows, including attached boundary layers, separated, secondary, and free-shear flows. Based on this framework, we construct a unified foundation model that integrates multiple diverse flow mechanisms into a single formulation. Unlike most existing data-driven models, which are trained for narrow flow categories and tend to degrade outside their training regimes Rumsey et al. (2022), our model performs well on the training flows and improves predictions for unseen flows exhibiting similar flow mechanisms. This directly addresses long-standing challenges emphasized in recent reviews and community discussions Spalart (2023); Rumsey and Coleman (2022); Rumsey et al. (2022), particularly the need for general turbulence models that deliver consistent, broadly applicable improvements while remaining robust and free from manual zoning. Such requirements were central to the NASA 2022 "Collaborative Testing Challenge" Rumsey and Coleman (2022). Our unified model makes substantial progress toward this goal, delivering robust and accurate predictions across multiple flow regimes.

Model Training and Evaluation Setup

The unified foundation turbulence model is trained on five flows representing distinct mechanisms: a plane channel for attached boundary layers, a periodic hill and an S809 airfoil at high angle of attack for internal/external separated flows, a square duct for secondary flows, and a round jet for free-shear flows. Each case provides sparse, indirect, and heterogeneous observations tailored to its nature: sparse velocity measurements for the plane channel, periodic hill, square duct, and round jet; and drag, lift coefficients for the S809 airfoil. Once trained, the model's generalization is tested on 27 unseen cases without manual zoning or parameter tuning, as shown in Fig. 1 and Table 1. Refer to SI Appendix for details.

Evaluation of the Unified Turbulence Model

The unified foundation model performs well across all training cases and generalizes effectively to unseen cases, including both those within the training flow categories and realistic 3D configurations. Its performance is evaluated on diverse test cases spanning attached boundary layers, separated flows, secondary flows, free-shear flows, and realistic 3D flows. The model's performance across training flow categories is shown in Fig. 2. It is evaluated using the normalized misfit $\hat{e} = \frac{e_{\text{unified}} - e_{\text{single}}}{e_{\text{base}} - e_{\text{single}}}$, where e_{unified} is the misfit of the unified foundation model relative to the ground truth, e_{base} is the misfit of the baseline model, and e_{single} is the misfit of a single-case trained model, which defines the upper performance limit. A value of $\hat{e} < 1$ indicates improved performance over baseline, while $\hat{e} > 1$ indicates degradation.

Overall, the unified foundation model generalizes well, achieving lower misfits than the baseline in most cases and comparable performance otherwise. For **attached boundary layers**, such as the zero pressure gradient flat plate, the model matches the baseline model in regimes where linear eddy-viscosity models are already reliable, recovering the law of the wall and maintaining accuracy across geometries and Reynolds numbers. In **free-shear flows**, represented here by a round jet, the model provides more accurate predictions of centerline velocity decay. For **secondary flows** in ducts with different Reynolds numbers and aspect ratios, the model better captures in-plane flows, maintaining accuracy under both geometric and Reynolds number extrapolation. For **separated flows**, the model substantially improves predictions of massive separation, such as periodic hills with varying slope steepness and S809 airfoil at high angles of attack. For flows over bumps and curved step, where separation is incipient, the model performs comparably

Categories	Flows	Control Parameter	Observations	No. Train	No. Test
Attached Boundary Layers	Zero pressure gradient flat plate		Velocities		1
	Plane channel	Reynolds number	Velocities	1	2
	S809 airfoil (attached)	Angle of attack	Lift, drag		2
Separated Flows	Curved step		Velocities		1
	Bump	Height	Wall friction		5
	Periodic hill	Slope steepness	Velocities	1	3
	S809 airfoil (separated)	Angle of attack	Lift, drag	1	4
Secondary Flows	Square duct	Reynolds number	Velocities	1	3
	Rectangular duct	Aspect ratio	Velocities		4
Free-shear Flows	Round jet		Velocities	1	
Realistic 3D Flows	Generic car		Velocities, drag		1
	3D diffuser		Velocities, wall friction		1
			Totals	**5**	**27**

Table 1: Categorized benchmark turbulent flows for evaluating the unified foundation model. The benchmark covers five categories: *attached boundary layers*, *separated flows*, *secondary flows*, *free-shear flows*, and *realistic 3D flows*. Most cases vary by control parameters such as Reynolds number, angle of attack, or geometric features (e.g., bump height, hill slope steepness), while case without such parameter correspond to a single configuration. Available observations include velocity, wall friction, lift, and drag. "No. Train" and "No. Test" denote the numbers of training and test cases, respectively. Representative datasets include flat plate Jespersen et al. (2016), plane channel Kim et al. (1987), S809 airfoil Somers (1997), curved step Bentaleb et al. (2012), bump Matai and Durbin (2019), periodic hill Xiao et al. (2020), square duct Pinelli et al. (2010), rectangular duct Vinuesa et al. (2018), round jet Bridges and Wernet (2010), generic car Ashton et al. (2024), and 3D diffuser Cherry et al. (2008).

to the baseline k–ω model. For **realistic 3D flows**, the generic car case shows improved separation predictions, while the 3D diffuser recovers ground truth skin friction coefficients along the bottom-wall midsection. However, improvements for realistic 3D flows remain limited, as the drag of the generic car is still inaccurate and the separation location in the diffuser is also not correctly predicted. Detailed performance of unified foundation model on six representative cases is illustrated in Fig. 3.

The unified foundation model demonstrates improved generalization across diverse flow regimes, achieving better, or at least comparable, accuracy to the baseline model in all test cases. Detailed case setups, training and evaluation results are provided in the SI Appendix.

Interpretation for Model Unification

The unified foundation model combines two complementary strategies for flow mechanism unification: automatic balancing of overlapping input features and internal branching for case-specific features across multiple flows. In overlapping regions, where identical features occur in different flows, the model resolves conflicts through multi-objective learning. In case-specific regions, it learns distinct mappings for each flow, acting as a piecewise function with internal branching. Together, these mechanisms enable the model to generalize across flows governed by distinct yet interacting flow mechanisms.

The model's input feature analysis reveals that different flows exhibit overlapping features while also maintaining distinct case-specific structures. We project the four frame-invariant input features into two dimensions using t-SNE Maaten and Hinton (2008) for visualization. Fig. 4A presents the results for two representative flows: periodic hill and square duct. The plot shows overlapping regions, as well as distinct regions specific to each case. This demonstrates that the unified foundation model captures common structures shared across flows while preserving case-specific characteristics.

Balancing in Overlapping Regions

Overlapping regions in the feature space serve as critical zones where the model must resolve conflicts among cases to enable joint learning of interacting mechanisms. Because identical features in these regions can correspond to different outputs across cases, deterministic
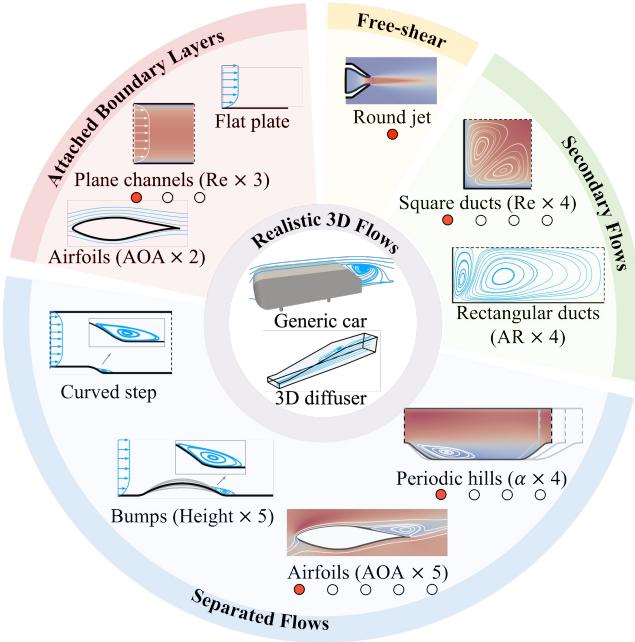
Figure 1: Canonical and realistic flows used to train and evaluate the unified foundation model, comprising 32 cases in total. The canonical flows are grouped into four categories: attached boundary layers, free-shear flows, secondary flows, and separated flows, each shown as a sector of the ring, while the two realistic 3D flows involving multiple mechanisms are placed in the inner circle. Five representative flows (depicted with contours and marked with red circles) form the training set, and the remaining 27 cases are used for testing. The testing set includes 25 canonical flows within the training categories and two realistic 3D flows from automotive and aerospace aerodynamics. Detailed specifications are provided in Table 1. This range of flows demonstrates that the multi-objective learning framework can incorporate diverse flow mechanisms during training and generalize effectively to both canonical and realistic 3D configurations.
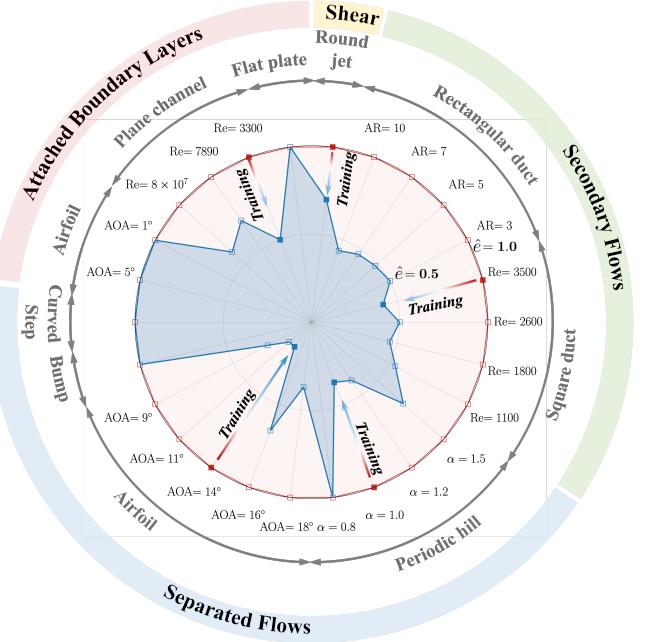

Figure 2: Radar chart comparing the performance of the baseline and unified foundation models across flows within training categories. The chart shows normalized misfits for attached boundary layers, free-shear flows, secondary flows, and separated flows. The red area represents the baseline model, while the blue area represents the unified foundation model. Smaller radial distances correspond to improved performance, and the chart illustrates the robust generalization of the unified foundation model across flow categories. Results for the bumps are nearly identical to the baseline, so only one case is shown.

Branching in Case-Specific Regions

In non-overlapping regions of the feature space, the network learns case-specific mappings, and effectively behaves like a piecewise function. To study this behavior, we extract case-specific features and pass them through the unified foundation model. For each neuron, we compute its activation rate, defined as the fraction of inputs producing a positive output. For rectified linear units (ReLU) Agarap (2018), this is equivalent to the proportion of non-zero outputs. The analysis is performed separately for the periodic hill and square duct.

The activation patterns of the sub-network for the quadratic coefficient g_2 are shown in Fig. 4C. Dark neurons and connections represent pathways activated in both the periodic hill and square duct cases, highlighting their similarities. In the periodic hill, most first-layer neurons are active, with one additional neuron uniquely activated (blue), but the second layer channels information almost entirely through a single pathway. In contrast, in the square duct, second-layer activations are more widely distributed, with several unique path-

models alone cannot eliminate the resulting discrepancies. Multi-objective learning addresses this by adjusting the contribution of each case during training, promoting balanced optimization and compelling the model to reconcile conflicting objectives. As shown in Fig. 4B, the single-case trained models often map overlapping features to significantly different outputs, creating distinct regions in the output space and limiting generalization. The unified foundation model resolves these conflicts by maintaining a balanced representation, achieving an intermediate and more generalizable state. This balancing not only enhances generalization but also preserves the ability to capture genuine interactions where multiple mechanisms coexist.
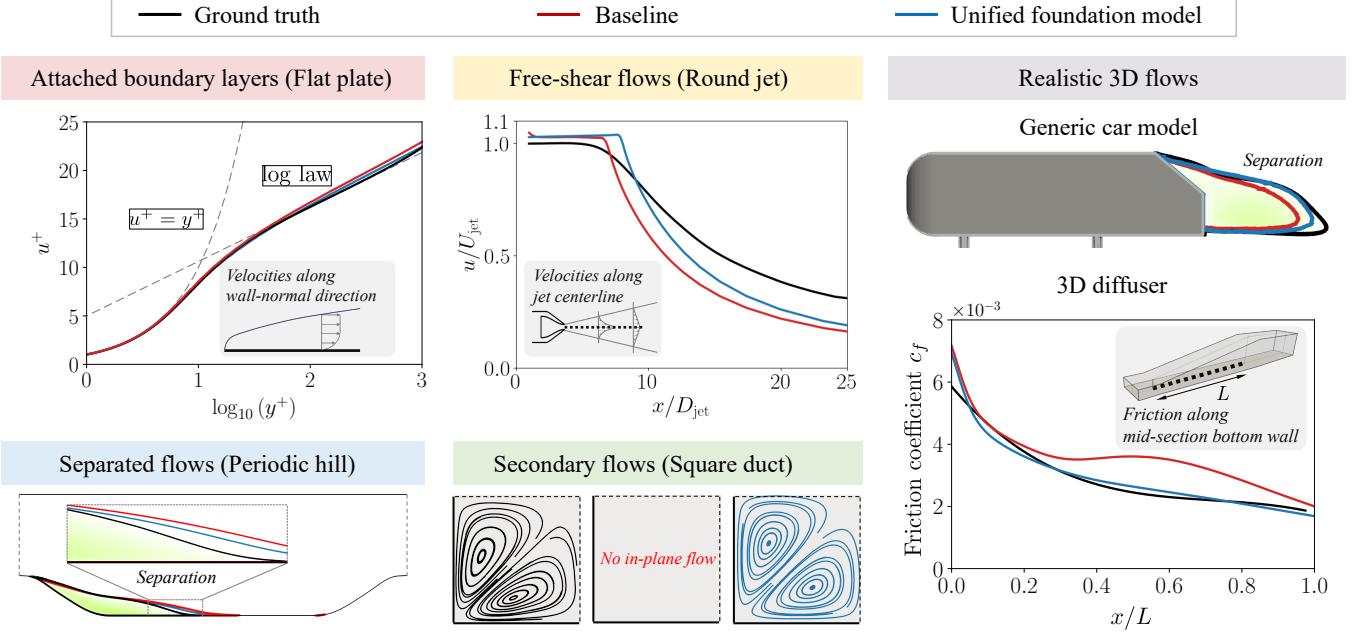
Figure 3: Performance of the unified foundation model across representative cases from training flow categories and two realistic 3D flows. The model is compared with ground truth and baseline model. For attached boundary layers, the flat plate recovers the law of wall. For free-shear flows, the round jet shows improved streamwise velocity decay along the jet centerline. For separated flows, the periodic hill with slope scaling factor $\alpha = 1.2$ demonstrates better prediction of separation size and reattachment location. For secondary flows, the square duct at $Re = 2600$ shows accurate recovery of corner vortices. For realistic 3D flows, the generic car yields improved separation predictions, while the 3D diffuser matches ground-truth skin-friction coefficients along the bottom-wall midsection.
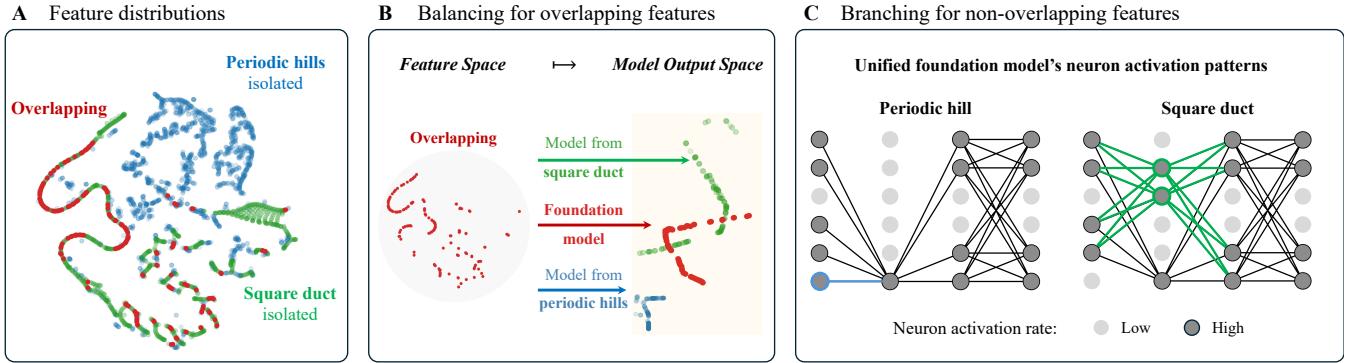

Figure 4: Panel A shows a t-SNE projection of the four-dimensional feature space, highlighting overlapping and case-specific regions for the periodic hill (blue) and the square duct (green), with red indicating overlap. Panel B shows that single-case trained models map overlapping features to different outputs, limiting generalization, whereas the unified foundation model resolves these conflicts and produces balanced predictions. Panel C shows neuron activation patterns in the sub-network for the quadratic coefficient g_2, where periodic hill activates fewer neurons, while square duct features activate more and rely more heavily on g_2.

ways remaining active (green). These distinct connection patterns demonstrate how the model reconfigures its internal branching to adapt to the statistical and structural characteristics of each flow.

Results: Specialist Model

The specialist model improves prediction accuracy for target flows by unifying related flow mechanisms exclusively — whole-device unification. Unlike the unified foundation model trained for broad coverage, it optimizes performance within a tailored set of flows that contain the target mechanisms. Although this specialization

cannot guarantee accuracy on other benchmark flows, it achieves high fidelity where it matters most, reflecting industrial practice in developing narrowly focused yet robust predictive tools.

Model Training and Evaluation Setup

The specialist model targets flows dominated by separation and secondary flow. Training uses three benchmark cases: the periodic hill for separated flows, and square and rectangular ducts for secondary flows. This combination allows the model to learn both secondary and separated flow mechanisms. The obtained model is evaluated on flows within the training regimes, including periodic hills with slope scaling factors α ranging from 0.8 to 1.5, square ducts with Reynolds numbers from 1100 to 2600, and rectangular ducts with aspect ratios from 5 to 10. It is also tested on an asymmetric 3D diffuser flow that involves interactions between the training flow mechanisms. The 3D diffuser is a challenging validation case featuring incompressible, asymmetric internal flow with strong adverse pressure gradients and significant Reynolds-stress anisotropy. The resulting three-dimensional separation closely mirrors practical diffuser behavior, making it an ideal benchmark for evaluating separation and secondary flow prediction capabilities.

Evaluation of the Specialist Model

The specialist model generalizes well across flows within the training categories and also shows significant improvement in the realistic 3D diffuser case. It substantially reduces the baseline's over-prediction of separation in periodic hills and successfully recovers in-plane flows in both square and rectangular ducts, which baseline models fail to reproduce. Moreover, the specialist model captures diffuser separation more accurately, correcting the baseline k–ω model's sidewall separation and aligning it with the ground truth, where separation occurs along the upper wall in expansion region.

Generalization Within Training Categories

Evaluation across varying geometric and flow parameters demonstrates robust performance within the training flow categories. Rectangular ducts with aspect ratios 5 to 10, square ducts with Reynolds numbers 1100 to 2600, maintain substantial improvements comparable to the training cases, and periodic hills with slope steepness variations sustain strong generalizability with modest sensitivity to slope changes. Refer to SI Appendix for details.

Generalization to Realistic 3D Flow

In the 3D diffuser, where secondary flows interacts with separation, the specialist model suppresses spurious side-wall separation and correctly predicts the primary upper-wall separation observed in the ground truth. Cross-sectional analyses show that the specialist model accurately captures the growth and reattachment of the separated region, while the baseline model misplaces both the location and extent of separation. Refer to SI Appendix for details. The specialist model's success on this realistic configuration demonstrates strong generalization beyond the training manifold and highlights its ability to capture interacting flow mechanisms in practical flows.

Discussion

An accurate, predictive turbulence model that performs reliably across a wide range of flow mechanisms, often simultaneously present in industrial flows, has long been a central goal in industrial computational fluid dynamics. In this work, we demonstrate that multi-objective learning enables the training of a unified foundation model on a heterogeneous dataset consisting of four canonical flow categories (e.g., attached and separated boundary layers, secondary flows, free-shear flows) and various target quantities (e.g., velocities, drag, lift). With the strategy of "unification through balancing and branching", this unified foundation model achieves Pareto-optimal performance across a broad range of test cases, including previously unseen and realistic scenarios involving multiple interacting flow mechanisms. Learning from canonical flows provides interpretability, ease of verification, and a scalable route to generalization. In contrast to training directly on application-specific flows, which are often inaccessible or lack clear mechanism labeling, our approach provides a structured path to unification by expanding coverage with canonical flows through multi-objective learning, balancing trade-offs in parallel while avoiding catastrophic forgetting Zhang et al. (2025).

While the unified foundation model yields consistent improvements over the baseline k–ω model, not all test cases show dramatic gains. This is expected, as many engineering flows involve additional complexities beyond the training flows and limited objective set. Nevertheless, the framework proves highly effective in application-specific scenarios where a small number of flow mechanisms dominate. We demonstrated this through two specialist models: one for three-dimensional diffuser flow, where adverse pressure-gradient-induced separation interacts with secondary motions, and the other for external vehicle aerodynamics, trained on flows with separations of varying severity. Refer to SI Appendix for details. These examples show that selecting and combining representative canonical flows enables efficient training of specialist models with strong generalization within their respective domains.

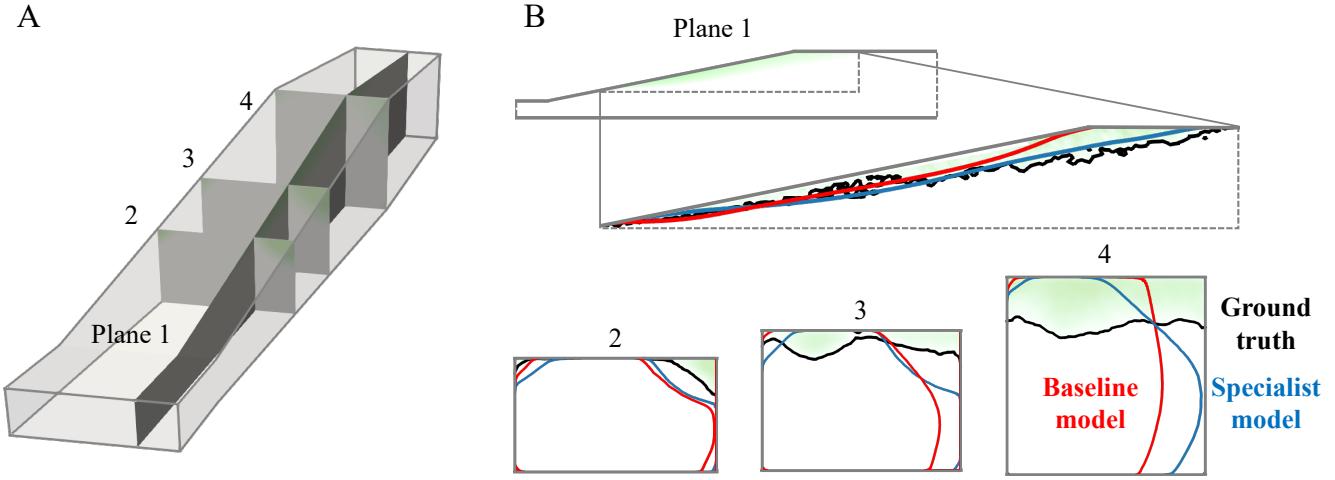The long-term goal is to develop a unified turbulence model that performs robustly across an entire device,

Figure 5: Panel A illustrates the 3D diffuser geometry and the sampling planes used for analysis. Plane 1 is a vertical spanwise plane near the inclined sidewall, while planes 2, 3, and 4 are cross-sectional planes along the streamwise direction within the expansion region. Panel B shows the predicted flow separation compared with the ground truth. Plane 1 reveals that the ground truth exhibits flow separation downstream of the expansion, which is not captured by the baseline model (red). In contrast, the specialist model (blue) accurately predicts separation close to the ground truth (black). In planes 2, 3, and 4, the ground truth shows separation predominantly on the diffuser upper wall. While the baseline model erroneously predicts sidewall separation, the specialist model correctly captures the upper-wall separation. These results demonstrate that the specialist model significantly outperforms the baseline.

rather than in isolated regions. For instance, while a three-dimensional diffuser represents a single component in a gas turbine, a full engine comprises many more interacting components with distinct flow characteristics. Switching among specialist models within a single simulation would be cumbersome and potentially unstable. Ideally, a single model should adapt seamlessly across the system. Achieving this will require scaling the current framework to accommodate a significantly larger set of objectives. Recent advances in multi-objective learning suggest that neural networks can balance up to 40 objectives Sener and Koltun (2018), making full-device unification technically feasible for configurations such as entire gas turbines or aircraft frames.

To achieve more consistent improvements across diverse flows and quantities, several technical enhancements are necessary. First, gradient accuracy for each objective must be improved. While our ensemble-based method is robust and broadly applicable, adjoint-based or fully differentiable solvers Bezgin et al. (2025) could provide more precise gradients, enabling more sophisticated architectures and fine-tuning strategies. Additionally, feature augmentation could better utilize the network's representational capacity by increasing the input space dimensionality, helping to reduce objective conflicts through internal branching. However, the averaging process that maps turbulent fluctuations to mean fields inherently causes information loss, meaning some mechanisms may remain indistinguishable in the input space. In such cases, effective balancing, supported by

accurate gradients, remains essential. The presented framework, with its physics-informed architecture and gradient-aware training, provides a promising foundation for further advances in unified turbulence modeling.

Methods

General Eddy-Viscosity Model and Its Representation

For constant-density, incompressible turbulent flows, the mean flow is governed by the Reynolds-averaged Navier–Stokes (RANS) equations:

$$\nabla \cdot \boldsymbol{U} = 0, \tag{1a}$$

$$\frac{\partial \boldsymbol{U}}{\partial t} + \boldsymbol{U} \cdot \nabla \boldsymbol{U} - \nu \nabla^2 \boldsymbol{U} + \frac{1}{\rho} \nabla P = \nabla \cdot \boldsymbol{\tau}, \tag{1b}$$

where $\boldsymbol{U}$ is the mean velocity, P is the mean pressure normalized by constant density, and $\boldsymbol{\tau}$ is the Reynolds stress tensor capturing the effect of velocity fluctuations. Since these fluctuations are not resolved in RANS, the Reynolds stress $\boldsymbol{\tau}$ requires modeling. The Reynolds stress $\boldsymbol{\tau}$ can be decomposed into a deviatoric part and a isotropic part:

$$\boldsymbol{\tau} = 2k\,\boldsymbol{b} + \frac{2}{3}k\mathbf{I},$$

where k is the turbulent kinetic energy, $\boldsymbol{b}$ is the normalized anisotropy tensor and $\mathbf{I}$ the identity tensor.

In the general eddy-viscosity framework Pope (1975), the normalized anisotropy $\boldsymbol{b}$ is expressed as:

$$\boldsymbol{b} = \sum_{i=1}^{10} g^{(i)}(\theta_1, \ldots, \theta_5)\, \mathbf{T}^{(i)}, \qquad (2)$$

where $\mathbf{T}^{(i)}$ and θ_j denote the tensor bases and scalar invariants derived from the mean velocity $\boldsymbol{U}$, and $g^{(i)}$ are the coefficient functions to be learned. For 2D incompressible turbulence, only the first two invariants and tensor bases are required Zhang et al. (2022).

To represent this model, we introduce a parallel tensor basis neural network. This frame-invariant formulation separates the output into dominant linear and corrective high-order components, extending the original tensor basis neural network Ling et al. (2016). Our model uses the first two invariants θ_1 and θ_2 along with two additional flow features: the viscosity ratio and turbulence Reynolds number Ling and Templeton (2015), as inputs, and employs the first two tensor bases $\mathbf{T}^{(1)}$ and $\mathbf{T}^{(2)}$ to construct Reynolds stress $\boldsymbol{\tau}$. Two parallel sub-networks predict the coefficients of the linear and quadratic terms independently, allowing them to be regularized separately. Refer to SI Appendix for details. This structure not only preserves low-order physics but also improves robustness and interpretability by providing flexible control over nonlinearity, while introducing higher-order corrections only when necessary.

Regularized Ensemble Learning from Indirect Observations

We extend the ensemble Kalman method (EnKF) with a regularization term to train turbulence models from sparse, indirect observation data in a stable and robust manner. The learning objective is to minimize the loss function ℓ Zhang et al. (2020):

$$\ell = \|\mathbf{w}_j^{i+1} - \mathbf{w}_j^i\|_{\mathsf{P}^{-1}}^2 + \|\mathbf{y}_j - \mathcal{H}[\mathbf{w}_j^{i+1}]\|_{\mathsf{R}^{-1}}^2 + \|\mathcal{G}[\mathbf{w}_j^{i+1}]\|_{\mathsf{Q}^{-1}}^2,$$

where $\mathbf{w}$ denotes the model parameters, $\mathbf{y}$ is the observation data, $\mathcal{H}$ is the observation operator mapping the model state to the observation space, and $\mathcal{G}$ represents the regularization. The matrices P, R, and Q are the covariance matrices of the state, observation errors, and regularization constraint, respectively, and define the weighted norms. The first two terms in loss ℓ constitute the standard EnKF objective, enforcing proximity to the prior estimate and agreement with observation data. The last term introduces regularization, defined as:

$$\mathcal{G} = \left[\left(g^{(1)} + 0.09 \right)^2, \ \left(g^{(2)} - 0 \right)^2 \right], \qquad (3)$$

which penalizes deviations from the baseline linear eddy-viscosity model Parish and Duraisamy (2016); Holland et al. (2019) with coefficients $g^{(1)} = -0.09$ and $g^{(2)} =$

0. Refer to SI Appendix for details. This anchoring prevents divergence while still allowing nonlinear data-driven refinements, yielding stable and generalizable turbulence models.

Multi-Objective Learning for Unified Turbulence Model

We aim to build a unified turbulence model that performs reliably across diverse flows. To this end, we adopt a multi-objective learning method in which each target quantity defines a loss $\ell_i(\mathbf{w})$, where $\mathbf{w}$ denotes the trainable model parameters. Optimizing these objectives independently or by naive averaging often biases the model toward some flows while degrading others.

Training with multiple flows is therefore cast as finding a *Pareto-optimal* solution, where no objective can be improved without worsening another Sener and Koltun (2018). By the Karush–Kuhn–Tucker conditions Kuhn and Tucker (1951), Pareto optimality requires a non-negative weight vector $\boldsymbol{\xi}$, with $\sum_{i=1}^{N} \xi_i = 1$, such that the weighted sum of gradients vanishes. This stationary condition can be expressed in practice by solving

$$\sum_{i=1}^{N} \xi_i \nabla_{\mathbf{w}} \ell_i(\mathbf{w}) = \mathbf{0} \quad \Rightarrow \quad \min_{\boldsymbol{\xi}} \left\| \sum_{i=1}^{N} \xi_i \nabla_{\mathbf{w}} \ell_i(\mathbf{w}) \right\|^2. \quad (4)$$

The weights $\boldsymbol{\xi}$ are computed using the Frank–Wolfe algorithm Jaggi (2013). At each step, the algorithm identifies the objective whose gradient conflicts most strongly with the current weighted direction and blends it into the combination with an analytically determined ratio from a line search. To ensure meaningful weighting across heterogeneous objectives, we first scale all losses to the same level. The gradients are then normalized by the product of their corresponding loss value and their L^2 norm. After this normalization, the effective gradient magnitude depends only on the loss value: objectives with smaller losses produce larger normalized gradients, which in turn receive smaller weights. This ensures that objectives already well-optimized contribute less to the combined update. Through this iterative reconciliation, the resulting descent direction improves all objectives jointly, or in some cases slightly worsens a few while still achieving an overall balanced improvement.

Combined with the ensemble learning method, the model parameters $\mathbf{w}$ update formula becomes:

$$\mathbf{w} \leftarrow \mathbf{w} + \sum_{i=1}^{N} \xi_i \, \Delta \mathbf{w}_i,$$

where $\Delta \mathbf{w}_i$ is the per-objective update provided by the regularized ensemble Kalman method Zhang et al. (2020). This coupling enables indirect data learning without requirement of adjoint solver and prevents dominance by any single flow type, yielding balanced improvements across multiple flows. Refer to SI Appendix for details.

Acknowledgments

References

Abien Fred Agarap. Deep learning using rectified linear units (ReLU). *arXiv preprint arXiv:1803.08375*, 2018.

Gonzalo Arranz, Yuenong Ling, Sam Costa, Konrad Goc, and Adrián Lozano-Durán. Building-block-flow computational model for large-eddy simulation of external aerodynamic applications. *Communications Engineering*, 3(1):127, 2024.

Neil Ashton, Danielle C Maddix, Samuel Gundry, and Parisa M Shabestari. AhmedML: High-fidelity computational fluid dynamics dataset for incompressible, low-speed bluff body aerodynamics. *arXiv preprint arXiv:2407.20801*, 2024.

Yacine Bentaleb, Sylvain Lardeau, and Michael A Leschziner. Large-eddy simulation of turbulent boundary layer separation from a rounded step. *Journal of Turbulence*, (13):N4, 2012.

Deniz A Bezgin, Aaron B Buhendwa, and Nikolaus A Adams. Jax-Fluids 2.0: Towards HPC for differentiable CFD of compressible two-phase flows. *Computer Physics Communications*, 308:109433, 2025.

Kaifeng Bi, Lingxi Xie, Hengheng Zhang, Xin Chen, Xiaotao Gu, and Qi Tian. Accurate medium-range global weather forecasting with 3D neural networks. *Nature*, 619(7970):533–538, 2023. doi:10.1038/s41586-023-06185-3.

James Bridges and Mark Wernet. Establishing consensus turbulence statistics for hot subsonic jets. In *16th AIAA/CEAS Aeroacoustics Conference*, 2010.

Tyler Buchanan, Monica Lăcătuş, Alastair West, and Richard P Dwight. Data-driven RANS closures using a relative importance term analysis based classifier for 2D and 3D separated flows. *arXiv preprint arXiv:2504.06758*, 2025.

Ewen Callaway. Deepmind's new AlphaGenome AI tackles the 'dark matter' in our DNA. *Nature*, 643:17–18, 2025. doi:10.1038/d41586-025-01998-w.

Soufiane Cherroud, Xavier Merle, Paola Cinnella, and Xavier Gloerfelt. Space-dependent aggregation of stochastic data-driven turbulence models. *Journal of Computational Physics*, 527:113793, 2025.

Erica M Cherry, Christopher J Elkins, and John K Eaton. Geometric sensitivity of three-dimensional separated flows. *International Journal of Heat and Fluid Flow*, 29(3):803–811, 2008.

TJ Craft, BE Launder, and K Suga. Development and application of a cubic eddy-viscosity model of turbulence. *International Journal of Heat and Fluid Flow*, 17(2):108–115, 1996.

Kalyanmoy Deb, Samir Agrawal, Amrit Pratap, and Tanaka Meyarivan. A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II. In *International conference on parallel problem solving from nature*, pages 849–858. Springer, 2000.

Jean-Antoine Désidéri. Multiple-gradient descent algorithm (MGDA) for multiobjective optimization. *C. R. Math.*, 350(5-6):313–318, 2012. doi:10.1016/j.crma.2012.03.014.

Karthik Duraisamy. Perspectives on machine learning-augmented Reynolds-averaged and large eddy simulation models of turbulence. *Phys. Rev. Fluids*, 6(5):050504, 2021. doi:10.1103/PhysRevFluids.6.050504.

Karthik Duraisamy, Gianluca Iaccarino, and Heng Xiao. Turbulence modeling in the age of data. *Annual Review of Fluid Mechanics*, 51:357–377, 2019. doi:10.1146/annurev-fluid-010518-040547.

Yuan Fang, Yaomin Zhao, Fabian Waschkowski, Andrew SH Ooi, and Richard D Sandberg. Toward more general turbulence models via multicase computational-fluid-dynamics-driven training. *AIAA J.*, 61(5):2100–2115, 2023. doi:10.2514/1.J062572.

Jonathan R Holland, James D Baeder, and Karthikeyan Duraisamy. Field inversion and machine learning with embedded neural networks: Physics-consistent neural network training. In *AIAA Aviation 2019 Forum*, 2019.

Martin Jaggi. Revisiting Frank-Wolfe: Projection-free sparse convex optimization. In *International Conference on Machine Learning*, pages 427–435. PMLR, 2013.

Dennis C Jespersen, Thomas H Pulliam, and Marissa Lynn Childs. Overflow turbulence modeling resource validation results. Technical Report No. ARC-E-DAA-TN35216, 2016.

John Kim and Anthony Leonard. The early days and rise of turbulence simulation. *Annual Review of Fluid Mechanics*, 56(1):21–44, 2024.

John Kim, Parviz Moin, and Robert Moser. Turbulence statistics in fully developed channel flow at low Reynolds number. *J. Fluid Mech.*, 177:133–166, 1987.

Nikola B Kovachki and Andrew M Stuart. Ensemble Kalman inversion: a derivative-free technique for machine learning tasks. *Inverse Problems*, 35(9):095005, 2019.

Harold W. Kuhn and Albert W. Tucker. Nonlinear programming. In Jerzy Neyman, editor, *Proceedings of the Second Berkeley Symposium on Mathematical Statistics and Probability*, pages 481–492, Berkeley, 1951. University of California Press.

Julia Ling and Jeremy Templeton. Evaluation of machine learning algorithms for prediction of regions of high Reynolds-averaged Navier–Stokes uncertainty. *Physics of Fluids*, 27(8), 2015.

Julia Ling, Andrew Kurzawski, and Jeremy Templeton. Reynolds averaged turbulence modelling using deep neural networks with embedded invariance. *J. Fluid Mech.*, 807:155–166, 2016. doi:10.1017/jfm.2016.615.

Adrián Lozano-Durán and H Jane Bae. Machine learning building-block-flow wall model for large-eddy simulation. *Journal of Fluid Mechanics*, 963:A35, 2023.

John L Lumley. Toward a turbulent constitutive relation. *J. Fluid Mech.*, 41(2):413–434, 1970.

Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9(Nov):2579–2605, 2008.

Racheet Matai and Paul Durbin. Large-eddy simulation of turbulent flow over a parametric set of bumps. *J. Fluid Mech.*, 866:503–525, 2019.

Florian R Menter and Alexey Matyushenko. Generalized k–ω (GEKO) two-equation turbulence model. *AIAA J.*, 2025.

Carlos A Michelén Ströfer and Heng Xiao. End-to-end differentiable learning of turbulence models from indirect observations. *Theor. Appl. Mech. Lett.*, 11(4):100280, 2021. doi:10.1016/j.taml.2021.100280.

Eric J Parish and Karthik Duraisamy. A paradigm for data-driven predictive modeling using field inversion and machine learning. *J. Comput. Phys.*, 305:758–774, 2016. doi:10.1016/j.jcp.2015.11.012.

Alfredo Pinelli, Markus Uhlmann, Atsushi Sekimoto, and Genta Kawahara. Reynolds number dependence of mean flow structure in square duct turbulence. *Journal of Fluid Mechanics*, 644:107–122, 2010. doi:10.1017/S0022112009992242.

Stephen B Pope. A more general effective-viscosity hypothesis. *J. Fluid Mech.*, 72(2):331–340, 1975. doi:10.1017/S0022112075003382.

Stephen B. Pope. *Turbulent flows*. Cambridge University Press, 2000. doi:10.1017/CBO9780511840531.

Di Qi and Andrew J Majda. Using machine learning to predict extreme events in complex systems. *Proceedings of the National Academy of Sciences*, 117(1):52–59, 2020.

Christopher L Rumsey and Gary N Coleman. NASA symposium on turbulence modeling: Roadblocks, and the potential for machine learning. Technical Report NASA/TM-20220014930, NASA, 2022.

Christopher L Rumsey, Gary N Coleman, and Li Wang. In search of data-driven improvements to RANS models applied to separated flows. In *AIAA Scitech 2022 Forum*, 2022.

Ozan Sener and Vladlen Koltun. Multi-task learning as multi-objective optimization. *Advances in Neural Information Processing Systems*, 31, 2018.

Weiyu Shen, Jie Yao, and Yue Yang. Designing turbulence with entangled vortices. *Proceedings of the National Academy of Sciences*, 121(35):e2405351121, 2024.

Tsan-Hsing Shih. A realizable Reynolds stress algebraic equation model. Technical Report NASA-TM-105993, 1993. Presented at the Symposium on Turbulence Shear Flows, Kyoto, Japan, Aug 10–18, 1993.

Jeffrey P Slotnick, Abdollah Khodadoust, Juan Alonso, David Darmofal, William Gropp, Elizabeth Lurie, and Dimitri J Mavriplis. CFD vision 2030 study: A path to revolutionary computational aerosciences. Technical Report NASA/CR-2014-218178, NASA, 2014.

Dan M Somers. Design and experimental results for the S809 airfoil. Technical Report No. NREL/SR-440-6918, National Renewable Energy Lab.(NREL), Golden, CO (United States), 1997.

Philippe Spalart. An old-fashioned framework for machine learning in turbulence modeling. *arXiv preprint arXiv:2308.00837*, 2023. doi:10.48550/arXiv.2308.00837.

Philippe R Spalart. Philosophies and fallacies in turbulence modeling. *Progress in Aerospace Sciences*, 74:1–15, 2015.

Philippe R Spalart and V Venkatakrishnan. On the role and challenges of CFD in the aerospace industry. *The Aeronautical Journal*, 120(1223):209–232, 2016.

Yoshiharu Tamaki and Soshi Kawai. Turbulence anisotropy effects on corner-flow separation: Physics and turbulence modelling. *J. Fluid Mech.*, 980:A21, 2024.

Yifeng Tian, Michael Woodward, Mikhail Stepanov, Chris Fryer, Criston Hyett, Daniel Livescu, and Michael Chertkov. Lagrangian large eddy simulations via physics-informed machine learning. *Proceedings of the National Academy of Sciences*, 120(34): e2213638120, 2023.

Ricardo Vinuesa, Philipp Schlatter, and HM Nagib. Secondary flow in turbulent ducts with increasing aspect ratio. *Physical Review Fluids*, 3(5):054606, 2018.

Fabian Waschkowski, Yaomin Zhao, Richard Sandberg, and Joseph Klewicki. Multi-objective CFD-driven development of coupled turbulence closure models. *J. Comput. Phys.*, 452:110922, 2022. doi:10.1016/j.jcp.2021.110922.

Chenyu Wu and Yufei Zhang. Enhancing the shear-stress-transport turbulence model with symbolic regression: A generalizable and interpretable data-driven approach. *Phys. Rev. Fluids*, 8(8):084604, 2023. doi:10.1103/PhysRevFluids.8.084604.

Heng Xiao and Paola Cinnella. Quantification of model uncertainty in RANS simulations: A review. *Progress in Aerospace Sciences*, 108:1–31, 2019. doi:10.1016/j.paerosci.2018.10.001.

Heng Xiao, Jin-Long Wu, Sylvain Laizet, and Lian Duan. Flows over periodic hills of parameterized geometries: A dataset for data-driven turbulence modeling from direct simulations. *Computers & Fluids*, 200:104431, 2020. ISSN 0045-7930. doi:10.1016/j.compfluid.2020.104431.

Fengshun Zhang, Zhideng Zhou, Xiaolei Yang, and Guowei He. Knowledge-integrated additive learning for consistent near-wall modelling of turbulent flows. *Journal of Fluid Mechanics*, 1011:R1, 2025.

Xin-Lei Zhang, Carlos. A Michelén-Ströfer, and Heng Xiao. Regularized ensemble Kalman methods for inverse problems. *J. Computational Physics*, 416:109517, 2020. doi:10.1016/j.jcp.2020.109517.

Xin-Lei Zhang, Heng Xiao, Xiaodong Luo, and Guowei He. Ensemble Kalman method for learning turbulence models from indirect observation data. *Journal of Fluid Mechanics*, 949:A26, 2022. doi:10.1017/jfm.2022.744.

Xin-Lei Zhang, Heng Xiao, Solkeun Jee, and Guowei He. Physical interpretation of neural network-based nonlinear eddy viscosity models. *Aerosp. Sci. Technol.*, 142:108632, 2023a. doi:10.1016/j.ast.2023.108632.

Xin-Lei Zhang, Heng Xiao, Xiaodong Luo, and Guowei He. Combining direct and indirect sparse data for learning generalizable turbulence models. *Journal of Computational Physics*, 489:112272, 2023b.

Ping Zhu, Kwun Yip Fung, Xuejin Zhang, Jun A Zhang, Jian-Wen Bao, Chuan-Kai Wang, Bin Liu, Zhan Zhang, Lucas Harris, Kun Gao, et al. Toward a unified parameterization of three dimensional turbulent transport in high resolution numerical weather prediction models. *npj Climate and Atmospheric Science*, 8(1): 223, 2025.

Supplementary Material

Turbulence is ubiquitous in natural and engineered flows, from weather and climate to mission-critical systems such as aircraft engines and power plants. Its chaotic, multiscale nature governs essential processes such as momentum, heat, and pollutant transport. Because resolving all scales is computationally infeasible, most practical predictions rely on turbulence closures within Reynolds-averaged Navier–Stokes (RANS) solvers Slotnick et al. (2014) or parameterization schemes in weather forecasting Zhu et al. (2025). Despite their central role, such closures remain crude, making turbulence modeling both scientifically and societally important.

More than fifty years ago, Lumley proposed a universal constitutive relation linking mean velocity to Reynolds stresses Lumley (1970). While this hypothesis was verified in simple homogeneous flows, it had little influence on practical modeling, where robustness and simplicity led to the dominance of empirical eddy-viscosity models such as Spalart–Allmaras and k–ω SST models Xiao and Cinnella (2019). Decades of experience suggest that a simple, local universal model does not exist Duraisamy et al. (2019). However, in engineering devices where boundary layers, separation, and secondary flows coexist and interact, a *unified turbulence model*, capable of handling multiple mechanisms seamlessly, without manual zoning or tuning, remains highly desirable but elusive Rumsey and Coleman (2022).

Eddy-viscosity closures form the foundation of recent efforts to build a *unified turbulence model* that captures multiple coexisting flow mechanisms without manual switching. This choice is natural because eddy-viscosity models are local, robust, and computationally efficient, making them reliable for industrial solvers and a strong basis for data-driven extensions Xiao and Cinnella (2019). Conventional eddy-viscosity models combine transport equations for turbulent scales with a linear constitutive relation to estimate the Reynolds stress. This framework can be generalized to nonlinear forms, offering greater flexibility than the standard Boussinesq assumption Speziale (1987); Gatski and Speziale (1993). The general eddy-viscosity model Pope (1975) has been realized through tensor basis neural networks Ling et al. (2016), symbolic formulations Weatheritt and Sandberg (2016); Schmelzer et al. (2018), and random forests Wang et al. (2017); Wu et al. (2018). Other studies have instead focused on improving the turbulence transport equations, for example by introducing data-driven multiplicative correction fields Parish and Duraisamy (2016). Beyond eddy-viscosity models, nonlocal approaches offer an alternative. These include Reynolds stress transport models Launder et al. (1975) and data-driven nonlocal constitutive relations Han et al. (2023); Zhou et al. (2022); Luther and Jenny (2025), which could capture additional physical effects such as non-equilibrium anisotropy. However, nonlocal models face major challenges: they are computationally expensive, less robust, and difficult in boundary condition treatment. Machine learning has also been used to refine specific terms in Reynolds stress transport models, such as the pressure–strain correlation Alaya et al. (2022), but the models themselves remain fragile. Considering these trade-offs, the tensor basis neural network formulation of the general eddy-viscosity model provides the most balanced compromise between accuracy, flexibility, and computational efficiency.

Within the eddy-viscosity framework, several strategies have been explored to achieve unification. Symbolic models trained with multi-objective optimization on diverse datasets have shown promising generalization across wall-bounded flows Fang et al. (2023). However, their restrictive functional forms and reliance on derivative-free optimization method (i.e., non-dominated sorting genetic algorithm II) limit scalability of the model training, particularly when balancing many competing objectives required for unification. Aggregation-based approaches, such as mixture-of-experts or classifier-driven model selection Cherroud et al. (2025); Buchanan et al. (2025), allow expert models for different mechanisms to be combined into a composite framework. However, they require multiple model evaluations and can struggle when the flow involves strong interactions between mechanisms. A different strategy employs adjoint-based field inversion and machine learning to learn multiplicative correction fields Rumsey et al. (2022); Wu et al. (2025). These methods have proven effective when applied to flows dominated by similar mechanisms Wu et al. (2025). Yet, when extended to unify flows with fundamentally different mechanisms, it faces challenges such as non-convergent training and limited generalizability Rumsey et al. (2022). Finally, the generalized k–ω model offers a pragmatic compromise by introducing tunable closure terms that allow systematic adjustments for different flow conditions Menter and Matyushenko (2025). This makes it attractive for practical use in the commercial solver Fluent, but its performance still relies heavily on manual tuning and thus falls short of true unification. Despite these advances, no turbulence model has yet achieved the goal of a unified framework that captures diverse coexisting mechanisms without manual tuning Rumsey and Coleman (2022). Achieving this remains a central open challenge in turbulence modeling.

A   Methodology

Our objective is to develop a unified, data-driven turbulence modeling framework that satisfies the following requirements:

(i)   The Reynolds stress representation will be frame-invariant and sufficiently flexible to represent diverse flow mechanisms.

(ii)   The model will be learned from sparse, indirect observations, and should prefer low-order representation to enhance robustness.

(iii)   The model will unify multiple flow mechanisms within a single framework efficiently, eliminating manual intervention.

A.1   General eddy-viscosity model and its representation via parallel tensor basis neural network

For constant-density, incompressible turbulent flows, the mean flow is governed by the Reynolds-averaged Navier–Stokes (RANS) equations:

$$\nabla \cdot \boldsymbol{U} = 0, \tag{S1a}$$

$$\frac{\partial \boldsymbol{U}}{\partial t} + \boldsymbol{U} \cdot \nabla \boldsymbol{U} - \nu \nabla^2 \boldsymbol{U} + \frac{1}{\rho}\nabla P = \nabla \cdot \boldsymbol{\tau}, \tag{S1b}$$

where $\boldsymbol{U}$ is the mean velocity, P is the mean pressure normalized by constant density, and $\boldsymbol{\tau}$ is the Reynolds stress tensor capturing the effect of velocity fluctuations. Since these fluctuations are not resolved in RANS, the Reynolds stress $\boldsymbol{\tau}$ requires modeling. It can be decomposed into a deviatoric part and an isotropic part as:

$$\boldsymbol{\tau} = \boldsymbol{a} + \frac{2}{3}k\mathbf{I}, \tag{S2}$$

where k is the turbulence kinetic energy, and $\mathbf{I}$ is the second-order identity tensor. According to the general eddy-viscosity formulation Pope (1975), the normalized Reynolds stress anisotropy, $\boldsymbol{b} = \boldsymbol{a}/(2k)$, can be represented as a linear combination of ten tensor bases $\mathbf{T}^{(i)}$ with coefficient functions $g^{(i)}$:

$$\boldsymbol{b} = \sum_{i=1}^{10} g^{(i)}(\theta_1, \ldots, \theta_5)\,\mathbf{T}^{(i)}, \tag{S3}$$

where the tensor bases $\mathbf{T}^{(i)}$ and scalar invariants θ_i are derived from the normalized mean strain rate $\hat{\mathbf{S}}$ and rotation rate $\hat{\mathbf{W}}$:

$$\hat{\mathbf{S}} = \tau_s \mathbf{S}, \quad \hat{\mathbf{W}} = \tau_s \mathbf{W}. \tag{S4}$$

Here, the mean strain rate $\mathbf{S}$ and rotation rate $\mathbf{W}$ are defined as $\mathbf{S} = \frac{1}{2}\left[\nabla \boldsymbol{u} + (\nabla \boldsymbol{u})^\top\right]$ and $\mathbf{W} = \frac{1}{2}\left[\nabla \boldsymbol{u} - (\nabla \boldsymbol{u})^\top\right]$, respectively. The turbulence time scale τ_s can be estimated using either the turbulent kinetic energy k and dissipation rate ε, or the specific dissipation rate ω, as $\tau_s = k/\varepsilon = 1/(C_\mu \omega)$. The first two scalar invariants and tensor bases are given by:

$$\theta_1 = \{\hat{\mathbf{S}}^2\}, \quad \theta_2 = \{\hat{\mathbf{W}}^2\} \tag{S5a}$$

$$\mathbf{T}^{(1)} = \hat{\mathbf{S}}, \quad \mathbf{T}^{(2)} = \hat{\mathbf{S}}\hat{\mathbf{W}} - \hat{\mathbf{W}}\hat{\mathbf{S}}, \tag{S5b}$$

where $\{\cdot\}$ denotes the matrix trace. Full definitions of all ten tensor bases and five scalar invariants are available in the literature Pope (1975); Ling et al. (2016). This formulation ensures frame invariance and provides the most general eddy-viscosity closure. For two-dimensional turbulent flows, the complete representation of Reynolds stress requires only the first two scalar invariants and tensor bases.

The tensor basis neural network (TBNN) Ling et al. (2016) was proposed to represent the general eddy-viscosity model and learns the unknown coefficients $g^{(i)}$ directly from data. Specifically, TBNN constructs a neural network-based mapping, $\mathcal{F}: \{\theta_i\}_{i=1}^5 \mapsto \{g_j\}_{j=1}^{10}$, ensuring model flexibility while maintaining frame invariance. Typically, neural network input features are scaled to $[-1, 1]$ to enhance training convergence. However, conventional min-max normalization, defined as $\hat{\theta} = (\theta - \theta_{\min})/(\theta_{\max} - \theta_{\min})$, can cause feature clustering near singular points and poor

generalizability as it utilizes global quantities, i.e., $\theta_{\max}$ and $\theta_{\min}$. Therefore, in contrast to the commonly used global normalization, we used a local time-scale normalization Wu et al. (2018); Zhang et al. (2023):

$$\hat{\mathbf{S}} = \frac{\mathbf{S}}{|\mathbf{S}| + 1/\tau_s}, \quad \hat{\mathbf{W}} = \frac{\mathbf{W}}{|\mathbf{W}| + 1/\tau_s}, \tag{S6}$$

from which new scalar invariants are subsequently derived, improving generalization.

In this work, we extend the tensor basis neural network to achieve more flexible control of model nonlinearity. The resulting architecture, called the parallel tensor basis neural network, is shown in Fig. S1. Unlike the original tensor basis neural network, which predicts all coefficients with a single network, the parallel tensor basis neural network uses multiple separated sub-networks to predict the coefficient functions $g^{(i)}$. We construct a quadratic eddy-viscosity model that depends on the first two scalar invariants, θ_1 and θ_2, along with two commonly used flow features: the viscosity ratio $q_3 = \frac{\nu_t}{100\nu + \nu_t}$ and the turbulent Reynolds number $q_4 = \min\left(\frac{\sqrt{k}d}{50\nu}, 2\right)$ Ling and Templeton (2015). These four quantities, denoted $\{q_i\}_{i=1}^4$, form the model input. The first two tensor bases, $\mathbf{T}^{(1)}$ and $\mathbf{T}^{(2)}$, are then combined with the predicted coefficients to construct the normalized Reynolds stress anisotropy tensor $\mathbf{b}$. The sub-networks independently predict the coefficient of the linear term, $g^{(1)}$, and the coefficient of the quadratic term, $g^{(2)}$. Although both sub-networks share the same input features $\{q_i\}_{i=1}^4$, their independent structures allow tailored regularization for each component. This parallel design explicitly captures hierarchical interactions by first emphasizing linear contributions and then introducing nonlinear corrections. As a result, the parallel TBNN enhances the interpretability, robustness, and accuracy of turbulence modeling.

(a) Architecture of original TBNN. (b) Architecture of parallel TBNN.
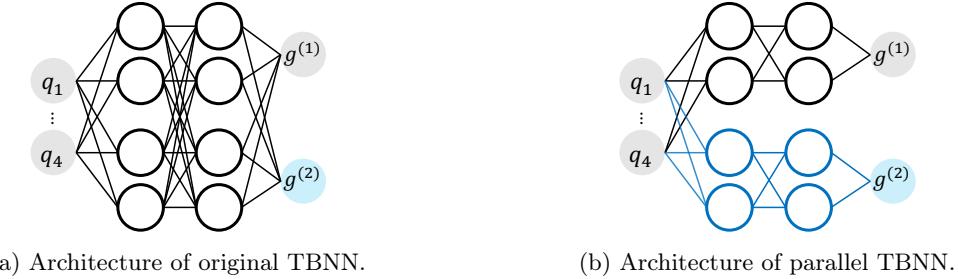
Figure S1: Comparison of the original and parallel tensor basis neural network (TBNN) architectures. (a) The original TBNN predicts the linear coefficient $g^{(1)}$ and the quadratic coefficient $g^{(2)}$ simultaneously with a single network. (b) The parallel TBNN predicts $g^{(1)}$ and $g^{(2)}$ using separate sub-networks, enabling flexible regularization and prioritization of the linear term.

A.2   Regularized ensemble learning from sparse, indirect observations for robustness

To achieve stable and efficient training, we adopt a regularized ensemble Kalman filter (REnKF) method, inspired by Zhang et al. (2020). Turbulence model learning is formulated as a Bayesian inverse problem: given sparse, indirect observations $\mathbf{y}$ and a baseline model as the prior, the goal is to infer the posterior distribution of the turbulence model, parameterized by neural network weights $\mathbf{w}$:

$$\mathbb{P}(\mathbf{w} \mid \mathbf{y}) \propto \mathbb{P}(\mathbf{w})\,\mathbb{P}(\mathbf{y} \mid \mathbf{w}), \tag{S7}$$

where $\mathbb{P}(\mathbf{w})$ is the prior baseline model and $\mathbb{P}(\mathbf{y} \mid \mathbf{w})$ the likelihood function. Under Gaussian and linear assumptions, the maximum a posteriori estimate reduces to the optimization problem:

$$\ell_{\text{EnKF}}(\mathbf{w}) = \|\mathbf{w}^{i+1} - \mathbf{w}^i\|_{\mathsf{P}^{-1}}^2 + \|\mathbf{y} - \mathcal{H}[\mathbf{w}^{i+1}]\|_{\mathsf{R}^{-1}}^2, \tag{S8}$$

where $\mathcal{H}$ is the observation operator that maps the model from state space to the observation space, and matrices P and R are the covariance matrices of the prior model and the observation errors, respectively. For any vector $\mathbf{v}$, the weighted norm is defined as

$$\|\mathbf{v}\|_{\mathsf{P}^{-1}}^2 = \mathbf{v}^\top \mathsf{P}^{-1} \mathbf{v}, \tag{S9}$$

with analogous definitions for $\|\cdot\|_{\mathsf{P}^{-1}}$ and $\|\cdot\|_{\mathsf{R}^{-1}}$. This problem can be solved iteratively using the ensemble Kalman filter (EnKF). However, repeated iterations tend to push the solution far from the baseline, effectively reducing the update to maximum likelihood estimation, which often leads to overfitting and poor generalizability.

To mitigate this issue, we introduce a regularization term that penalizes deviations from the baseline linear eddy-viscosity model, following the same principle of field inversion and machine learning Parish and Duraisamy (2016). The resulting regularized loss function is

$$\ell_{\mathrm{REnKF}}(\mathbf{w}) = \|\mathbf{w}^{i+1} - \mathbf{w}^i\|_{\mathsf{P}^{-1}}^2 + \|\mathbf{y} - \mathcal{H}[\mathbf{w}^{i+1}]\|_{\mathsf{R}^{-1}}^2 + \|\mathcal{G}[\mathbf{w}^{i+1}]\|_{\mathsf{Q}^{-1}}^2, \tag{S10}$$

where the matrices Q denotes the covariance matrix of the regularization constraints. The regularization operator $\mathcal{G}$ penalizes deviations of the model coefficients from their baseline values, $g^{(1)} = -0.09$ and $g^{(2)} = 0$:

$$\boldsymbol{\mathcal{G}} = [\mathcal{G}_1, \mathcal{G}_2] = \left[\left(g^{(1)} + 0.09\right)^2, \ \left(g^{(2)} - 0\right)^2 \right]. \tag{S11}$$

Minimizing this cost function ℓ_{REnKF} leads to the following update scheme, which can be expressed in two stages: a pre-correction step that accounts for the regularization term, followed by the standard ensemble Kalman update:

$$\widetilde{\mathbf{w}}_j^i = \mathbf{w}_j^i - \mathsf{P}\mathcal{G}'[\mathbf{w}_j^i]^\top \mathsf{Q}^{-1}\mathcal{G}[\mathbf{w}_j^i], \tag{S12a}$$

$$\mathbf{w}_j^{i+1} = \widetilde{\mathbf{w}}_j^i + \mathsf{K}\big(\mathbf{y}_j - \mathcal{H}[\widetilde{\mathbf{w}}_j^i]\big), \tag{S12b}$$

$$\mathsf{K} = \mathsf{P}\mathsf{H}^\top \left(\mathsf{H}\mathsf{P}\mathsf{H}^\top + \mathsf{R}\right)^{-1}. \tag{S12c}$$

The detailed implementation procedure is provided in our earlier work Zhang et al. (2020). Overall, this regularized ensemble Kalman filter framework mitigates instabilities in nonlinear eddy-viscosity models and yields turbulence closures that are both robust and generalizable.

A.3   Multi-objective learning for unified turbulence model

A unified turbulence model should maintain robust performance across diverse flows without manual intervention. To achieve this objective, we propose training a unified turbulence model with heterogeneous data (e.g., sparse velocities and force) from a wide range of flows (e.g., attached and separated flows) within a single optimization process. This approach can be formulated as a multi-objective learning strategy and each objective corresponds to minimize the loss ℓ_j in terms of a quantity of interest from a particular flow, totaling N combination:

$$\min_{\mathbf{w}} \left[\ell_1\left(\mathbf{w}\right), \ell_2\left(\mathbf{w}\right), ..., \ell_N\left(\mathbf{w}\right)\right]^\top, \tag{S13}$$

where $\mathbf{w}$ represents the parameters of neural network-based turbulence model. However, this approach can potentially involve conflicting objectives during optimization.

Pareto optimality refers to a solution in multi-objective learning where no objective can be improved without negatively impacting another. In the context of a unified turbulence model, it means identifying a model with parameter $\mathbf{w}$ such that further optimizing loss ℓ_j for one objective would inevitably worsen performance on others. This ensures a balanced trade-off across objectives, improving the overall performance in multiple flow scenarios. When handling multi-objective learning problem, the weighted summation of losses is a common approach, which converts multiple objectives into a single objective by weighting each objective with specified coefficients. However, this approach has inherent limitations. It often requires costly grid searches to determine appropriate weights for multiple objectives or relies on heuristics Kendall et al. (2018); Chen et al. (2018). More critically, it lacks a clear definition of global optimality in the multi-objective setting, making it difficult to effectively balance conflicting objectives, leading to suboptimal performance across objectives. In contrast, the multi-objective optimization problem can be solved with the multi-gradient descent algorithm with dynamically adjusted weights Désidéri (2012):

$$\mathbf{w} \leftarrow \mathbf{w} - \eta \sum_{j=1}^{N} \xi_j \nabla_{\mathbf{w}}\ell_j(\mathbf{w}), \tag{S14}$$

where η is the optimization step size (learning rate) and ξ_j denotes weight corresponding to objective j. Pareto optimality can be formulated as Pareto stationarity by leveraging the Karush–Kuhn–Tucker (KKT) conditions Kuhn and Tucker (1951). Specifically, a solution $\mathbf{w}$ is Pareto stationary if there *exists* a weight vector $\boldsymbol{\xi} = [\xi_1, \xi_2, \ldots, \xi_N]^\top$, where each element ξ_j is non-negative and sum to one, i.e., $\xi_j \geq 0$ and $\sum_{j=1}^{N} \xi_j = 1$, such that the weighted sum of objective gradients $\sum_{j=1}^{N} \xi_j \nabla_{\mathbf{w}}\ell_j(\mathbf{w})$ results in the zero vector. This implies that if the parameter $\mathbf{w}$ is Pareto stationary, there must exist gradients with opposing signs (unless all objective gradients $\nabla_{\mathbf{w}}\ell_j(\mathbf{w})$ are zero). The objective is thus to find such a weight vector $\boldsymbol{\xi}$.

The Karush–Kuhn–Tucker conditions provide a way to identify a Pareto stationary parameter $\mathbf{w}$ by solving an auxiliary optimization problem for the weight vector $\boldsymbol{\xi}$. This involves minimizing the norm of the weighted sum of objective gradients, $\sum_{j=1}^{N} \xi_j \nabla_{\mathbf{w}} \ell_j(\mathbf{w})$. A zero norm indicates that the KKT conditions are satisfied, meaning $\mathbf{w}$ lies at a Pareto stationary point where objectives are in conflict. Conversely, a nonzero norm suggests cooperative optimization, where gradients align to improve all objectives simultaneously. In this case, the optimal weights $\boldsymbol{\xi}^{\mathrm{opt}}$ yield a descent direction $\sum_{j=1}^{N} \xi_j^{\mathrm{opt}} \nabla_{\mathbf{w}} \ell_j(\mathbf{w})$. To ensure meaningful weighting across heterogeneous objectives, all losses are first scaled to a comparable level. Each gradient is then normalized by the product of its loss value and its L^2 norm. After this normalization, the effective gradient magnitude depends only on the loss value: objectives with smaller losses yield larger normalized gradients, which receive smaller weights. This ensures that objectives already well optimized contribute less to the combined update, while underperforming objectives are emphasized. For simplicity, let $\boldsymbol{G}_j$ denote the gradient of objective $\nabla_{\mathbf{w}} \ell_j(\mathbf{w})$, and let $\overline{\boldsymbol{G}}$ denote their weighted sum $\sum_{j=1}^{N} \xi_j \boldsymbol{G}_j$. The Frank–Wolfe algorithm Jaggi (2013) then optimizes $\boldsymbol{\xi}$ by identifying the objective gradient $\boldsymbol{G}_J$ that is most misaligned with $\overline{\boldsymbol{G}}$, indicating conflict between objective J and the aggregated objective. The algorithm increases the weight ξ_J accordingly, while preserving the normalization constraint on $\boldsymbol{\xi}$, thereby balancing the competing objectives.

As an example, we illustrate the Frank–Wolfe algorithm in the scenario of a two objectives in Fig. S2. The algorithm starts by initializing equal weight for each objective $\boldsymbol{\xi} = [0.5, 0.5]^{\top}$ and compute the gradient $\widetilde{\boldsymbol{G}} = \frac{1}{2}(\boldsymbol{G}_1 + \boldsymbol{G}_2)$. In the second step, the algorithm identifies the most conflicting gradient as $\boldsymbol{G}_2$ and resolve the conflict by updating the gradient as $\overline{\boldsymbol{G}} = (1 - \gamma)\widetilde{\boldsymbol{G}} + \gamma \boldsymbol{G}_2$, where the γ is derived from the simple line search. The algorithm will repeat step 2 until it finds a non-conflicting gradient $\widetilde{\boldsymbol{G}}$ or the search algorithm reaches the maximum step. Unlike derivative-free optimization methods (e.g., non-dominated sorting genetic algorithm), this gradient-based approach is highly efficient at managing up to 40 tasks Sener and Koltun (2018), making it ideal for developing a unified turbulence model.
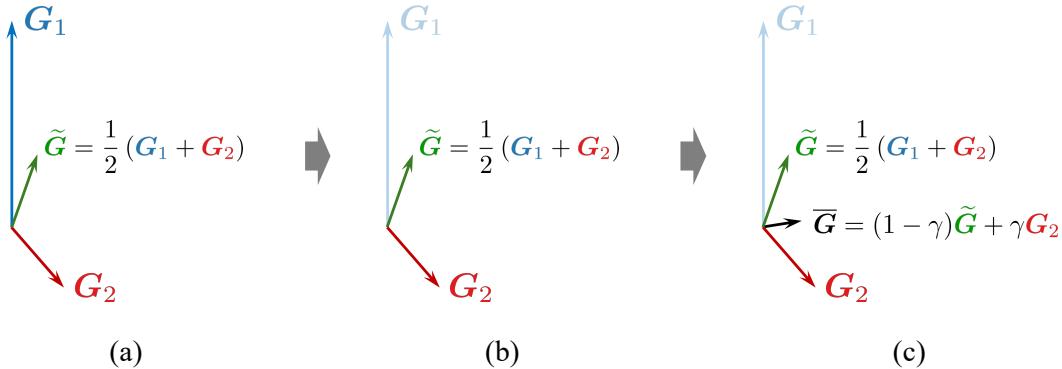

Figure S2: Illustration of Frank–Wolfe algorithm in the scenario of two objectives with conflicts, including: (a) averaging of the gradient of two objectives to obtain mean gradient $\widetilde{\boldsymbol{G}}$, (b) identifying the gradient associated with the most conflicting objective, in this case $\boldsymbol{G}_2$, and (c) under-relaxation of the mean gradient with a ratio with this conflicting gradient with a factor of γ. Such a relaxation amounts to a compromise between the mean gradient and the most conflicting gradient. The process continues, until all conflicts are resolved or the maximum step is achieved.

To integrate the multi-objective learning into the regularized ensemble learning framework, we derive a multi-objective update formula for model parameters $\mathbf{w}$ and it has the following form:

$$\mathbf{w} \leftarrow \widetilde{\mathbf{w}} + \sum_{j=1}^{N} \xi_j \boldsymbol{G}_j, \quad \text{with } \boldsymbol{G}_j = \mathsf{K}_j\big(\mathbf{y}_j - \mathcal{H}_j[\widetilde{\mathbf{w}}]\big) \tag{S15}$$

where K_j is the Kalman gain matrix for objective i, and $\mathcal{H}_j$ represents the corresponding observation operator. The weight vector $\boldsymbol{\xi}$ is derived from the aforementioned Frank–Wolfe algorithm. The details are provided in Algorithm 1.

B Case setup

We present the training and test cases used to evaluate three turbulence models: the unified foundation model, the specialist model for separated flows, and the specialist model for secondary flows with separation. We summarize the setup of all cases together with the corresponding observation data in Fig. S3.

Algorithm 1 multi-objective learning with regularized ensemble Kalman filter method (REnKF)

1: **for** $t = 1, 2, \ldots, T$ **do**

2:       Compute individual objective j's update by REnKF:

$$\boldsymbol{G}_j = \mathsf{K}_j\big(\mathbf{y}_j - \mathcal{H}_j[\widetilde{\mathbf{w}}]\big),$$

3:       Normalize each update:

$$\boldsymbol{G}_j^{\mathrm{n}} = \frac{\boldsymbol{G}_j}{\ell_j \cdot \|\boldsymbol{G}_j\|}$$

4:       Initialize task weights: $\boldsymbol{\xi} \leftarrow (1/N, \ldots, 1/N)$

5:       **repeat**

6:          Compute weighted mean gradient:

$$\overline{\boldsymbol{G}^{\mathrm{n}}} = \sum_{j=1}^{N} \xi_j \boldsymbol{G}_j^{\mathrm{n}}$$

7:          Identify the most conflicting task:

$$J = \arg \min_j \boldsymbol{G}_j^{\mathrm{n}\top} \overline{\boldsymbol{G}^{\mathrm{n}}}$$

8:          Compute step size γ via line search:

$$\gamma = \arg \min_{\gamma \in [0,1]} \big\|(1-\gamma)\overline{\boldsymbol{G}^{\mathrm{n}}} + \gamma \boldsymbol{G}_J^{\mathrm{n}}\big\|^2$$

9:          Update task weights:

$$\boldsymbol{\xi} \leftarrow (1-\gamma)\boldsymbol{\xi} + \gamma \mathbf{e}_J$$

10:      **until** convergence of $\boldsymbol{\xi}$ or maximum steps reached

11:      Update parameters:

$$\mathbf{w} \leftarrow \widetilde{\mathbf{w}} + \eta \left(\sum_{j=1}^{N} \xi_j \boldsymbol{G}_j \right)$$

12: **end for**

13: **return** Final parameters $\mathbf{w}$

B.1 Unified foundation model

The unified foundation model is trained on five representative flows and evaluated across two groups: flows within the training categories and realistic 3D flows. The five training flow configurations are: attached boundary layers (plane channel flow); separated flows (flow over periodic hill and S809 airfoil flow with high angle of attack); secondary flows (flow through square duct); and free-shear flows (round jet flow). An overview of each training and test case is shown in Fig. S4. For each case, Fig. S5 depicts the set of training cases used in the multi-objective learning framework. The following paragraphs elaborate on the training and test cases introduced in Fig. S5.

B.1.1 Training cases

Plane channel flow at a moderate Reynolds number Re = 3300 exhibits distinct viscous, buffer, and logarithmic regions near the wall. The computational domain is periodic in the streamwise direction, with no-slip walls at the top and bottom. The mean velocity profile follows the classical law of the wall, and matching it serves as a fundamental validation step for turbulence models. Four streamwise velocity values are sampled along the wall-normal direction, covering the viscous sub-layer to the logarithmic region, and are used as indirect observations.

     Flow over the S809 airfoil at high angles of attack (AOA > 10°) is dominated by massive flow separation, making

it challenging for traditional turbulence models, which often overestimate the lift force beyond stall Singh and Duraisamy (2016). At $\mathrm{Re} = 2 \times 10^6$ based on chord length and inflow velocity, the suction side experiences a strong adverse pressure gradient, causing early separation. The case at $\mathrm{AOA} = 14°$ is selected for training, using experimentally measured lift and drag coefficients as observations.

Flow over periodic hill is a benchmark for internal separated flows, characterized by turbulence in the presence of periodic smooth hills that induce flow separation and reattachment. The setup operates at $\mathrm{Re}_{h_p} = 10595$, defined using the bulk velocity and hill height h_p, with slope scaling factor $\alpha = 1.0$. The domain applies streamwise-periodic boundary conditions, no-slip walls, and a specified bulk velocity. Velocities from DNS data are extracted at four cross-sections ($x/h_p = 1.0, 3.0, 5.0, 7.0$), yielding 66 velocity components as observations.

Square duct flow introduces secondary flows due to turbulence-induced anisotropy in a non-circular cross-section. The domain assumes fully developed flow with periodic streamwise boundaries and no-slip walls. To reduce computational cost, one-quarter of the cross-section is simulated, with symmetry applied to two sides. Capturing both axial velocity and secondary cross-sectional flows is essential for assessing a model's ability to represent anisotropy. Training observations include mean velocities in three directions at four cross-sections, totaling 193 velocity components. The training case is conducted at a Reynolds number of $\mathrm{Re} = 3500$.

Round jet evolves from a potential core into a turbulent shear layer before mixing with the ambient fluid. The total pressure is imposed at the jet inlet, pressure-outlet conditions are applied at the outlet, velocity is specified for the freestream inflow, and no-slip conditions are applied on the walls. Model validation requires reproducing the centerline velocity decay. Accordingly, 25 streamwise velocities along the centerline are used as training observations.

B.1.2 Test cases

We evaluate the unified foundation model on 27 turbulent flows, divided into two groups: flows within the training categories and realistic three-dimensional flows (Table. S1).

Attached boundary layers The attached boundary layer category contains five test cases: a flat plate, channel flows at two Reynolds numbers, and airfoils at two low angles of attack. The flat plate case employs uniform inflow and no-slip boundary conditions at the wall, with periodicity applied in the spanwise direction. Two velocity values are sampled along the wall-normal direction to capture near-wall behavior, and the case represents the canonical development of a turbulent boundary layer, where viscous effects dominate close to the wall and gradually transition to turbulent momentum transport farther away. The channel flows at two different Reynolds numbers ($\mathrm{Re} = 7890$, $\mathrm{Re} = 8 \times 10^7$) use the same boundary conditions and sampling strategy as the training case, with Reynolds number as the only varying parameter. Four streamwise velocity values are sampled along the wall-normal direction, allowing systematic examination of Reynolds-number effects while retaining the classical structure of viscous, buffer, and logarithmic layers. The attached airfoil test cases at $\mathrm{AOA} = 1°$ and $5°$ follow the same setup as the training cases, differing only in angle of attack. Lift and drag coefficients are calculated over the entire airfoil surface. These low angles maintain attached flow without massive separation, ensuring them remain representative of attached boundary layers.

Separated flows The separated flow category contains four types of test cases: curved step, bump, airfoils at four high angles of attack, and periodic hills at three slope scaling factors. The curved step case uses uniform inflow with a no-slip wall and outflow boundary conditions downstream, with velocity samples collected in the recirculation zone to measure bubble length and reattachment. The bump cases of different heights are simulated with periodic boundary conditions in the streamwise direction and no-slip conditions at the walls, with the wall friction coefficient used to assess the predictions of separation. The airfoil test cases at $\mathrm{AOA} = 9°$, $11°$, $16°$ and $18°$ share the same setup and sampling strategy as the training case, with lift and drag coefficients extracted; at these higher angles, separation and stall dominate the flow behavior. The periodic hill cases at three slope scaling factors ($\alpha = 0.8, 1.2,$ and 1.5) impose periodic streamwise and spanwise boundaries with no-slip walls at top and bottom, and streamwise velocity is sampled along wall-normal lines downstream of the hill crest, with points taken every five computational cell, to determine separation and reattachment lengths. Together, these cases form a diverse testbed of separated turbulent flows, ranging from localized recirculation to fully stalled aerodynamic configurations.

Secondary flows The secondary flow category contains two types of test cases: square ducts at three Reynolds numbers ($\mathrm{Re} = 1100, 1800, 2600$) and rectangular ducts at four aspect ratios ($\mathrm{AR} = 3, 5, 7, 10$). In both geometries, a fully developed turbulent profile is prescribed at the inlet, with no-slip walls along all duct boundaries and a constant mass flux condition to sustain the flow. The domain is periodic in the streamwise direction, ensuring

Table S1: Test cases for the unified foundation model

Case	Control parameter	Value	No. of cases
Flat plate	Reynolds number	5×10^6	1
Plane channel	Reynolds number	7890 8×10^7	2
S809 airfoil	Angle of attack	1° 5° 9° 11° 16° 18°	6
Periodic hill	Slope scaling factor	0.8 1.2 1.5	3
Curved step	Reynolds number	13700	1
Bump	Height	20 mm 26 mm 31 mm 38 mm 42 mm	5
Square duct	Reynolds number	1100 1800 2600	3
Rectangular duct	Aspect ratio	3 5 7 10	4
Generic car	Slant angle	40°	1
3D diffuser	Reynolds number	10000	1
		Totals	**27**

statistical stationarity. Velocity observations are taken in cross-sectional planes, identical to the sampling strategy used in the training case, to capture mean velocity distortion and secondary vortex development. The square duct flows emphasize corner-induced vortices, while the rectangular duct flows show how aspect ratio affects secondary circulations. Together, these cases test the ability of turbulence models to capture inherently secondary motions in duct flows.

Realistic 3D flows The generic car challenges turbulence models with bluff-body wake dynamics, including shear-layer vortices, corner vortices, and a large base recirculation zone. Observation data consist of surface-integrated drag coefficient, which reflects the accuracy of predicted separation and wake structures. The asymmetric 3D diffuser tests turbulence models under adverse pressure gradients and Reynolds stress anisotropy that drive asymmetric separation. Observation data include velocity profiles across the separated shear layer, and wall friction trends that capture onset and reattachment.

B.2 Specialist turbulence model for flow separation

Conventional turbulence models often exhibit poor performance in flows with massive separation. To address this limitation, we develop a specialist model tailored for separated flows. This section describes the training configurations and the test cases selected to evaluate its predictive performance.

B.2.1 Training cases

To obtain a specialist model for flows with large separation, we train the model using three benchmark separated flows: flow over periodic hill (described in detail in Section. B.1.1), flow over a bump, and flow over a curved step as shown in Fig. S6. The bump's convex curvature induces an adverse pressure gradient that produces an incipient separation bubble with reattachment several bump heights downstream, while the curved step's sudden streamline curvature generates a large separation region followed by gradual reattachment along the lower wall. Below, we introduce the bump and the curved step in detail.

Flow over a bump is a two-dimensional wall-mounted configuration formed by a circular arc smoothly blended into a flat plate through convex fillets at both ends. This geometry produces an incipient flow separation on the lee side, making it a demanding test case for turbulence model evaluation. At the selected operating condition, the inflow Reynolds number based on the momentum thickness and freestream velocity is $\mathrm{Re}_{\theta_\infty} = 2500$. While shallow bumps exhibit minimal separation, the present curvature induces a strong adverse pressure gradient downstream of the crest, leading to primary separation near the trailing fillet and a short recirculation bubble. The inlet velocity and turbulence profiles are extracted from a precursor large-eddy simulation of a zero-pressure-gradient turbulent boundary layer at the same $\mathrm{Re}_{\theta_\infty}$. Zero-gradient conditions are applied at the top and outlet boundaries, while the bottom boundary is treated as a no-slip wall. We use the wall skin friction coefficient c_f along the lower surface, sampled at 42 points and obtained from LES, as the training observation and the quantity of interest.

Curved step flow consists of an upstream convex ramp that terminates in a sudden downward step. Unlike the flows over periodic hill and bump, which induce separation after an accelerating boundary layer, this configuration forces the detachment of an already fully developed turbulent boundary layer through abrupt streamline curvature. This setup resolves both the attached upstream boundary layer and the separated region downstream of the step. The flow operates at $\mathrm{Re} = 13700$ and 27 velocity values are sampled along several wall-normal lines, as shown in Fig. S3, and used for model training.

B.2.2 Test cases

Two groups of cases for evaluating the trained specialist model for separated flows are chosen and summarized in Table S2: test cases within the training categories (seven parameter variations: bumps with different heights $(20, 26, 38, 42\,\mathrm{mm})$ and periodic hills with slope factors $\alpha = 0.8, 1.2, 1.5$) and a realistic three-dimensional case: the generic car shown in Fig. S3.

Table S2: Test cases for the specialist model of separated flows

Case	Control parameter	Value	No. of cases
Bump	Height	20 mm 26 mm 38 mm 42 mm	4
Periodic hill	Slope scaling factor	0.8 1.2 1.5	3
Generic car	Slant angle	40°	1
		Totals	**8**

The bumps retain the same configuration as in training while varying the bump height, which systematically alters the adverse pressure gradient and separation strength. The periodic hills use the same flow conditions as in training, but their geometry varies through a slope scaling factor, creating a family of separated flows with different intensities. Together, these cases extend the training set into controlled parametric variations, enabling assessment of model robustness.

The generic car is a standard benchmark in automotive aerodynamics, used to evaluate turbulence models for bluff-body flows. The model is placed in a wind tunnel domain and a uniform inlet yields a Reynolds number $Re = 2.8 \times 10^6$. A zero-pressure outlet boundary condition is applied at the downstream end, no-slip conditions are imposed on all solid surfaces, and symmetry conditions are used on the top and lateral boundaries to exploit geometric symmetry and reduce computational cost. The drag coefficient is calculated over the entire generic car surface as the observation data. The $40°$ slanted rear surface produces a complex three-dimensional wake, including top-shear-layer vortices, corner vortices, and a large base recirculation bubble. These flow structures make the generic car a stringent test for assessing turbulence model capability in predicting complex three-dimensional separation and wake dynamics.

B.3   Specialist turbulence model for secondary flows with separation

We train a specialist turbulence model aimed at accurately capturing secondary flows with separation (multiple mechanisms). Recognizing that practical turbulent flows rarely involve a single dominant mechanism, but instead arise from complex interactions among multiple phenomena, the model is trained on configurations where both effects are prominent. This section outlines the training setups and test cases used to evaluate the model's predictive performance.

B.3.1   Training cases

We select three benchmark cases to train the specialist model for secondary flows with separation shown in Fig. S7, targeting large flow separation and secondary flow mechanisms: flow over a periodic hill and flows through square and rectangular ducts. Both the periodic hill and square duct cases are detailed in Section. B.1.1. The rectangular duct flow exhibits asymmetric corner vortices, with the vortex on the horizontal wall expanding spanwise and dominating while the vertical wall vortex is compressed, producing a skewed concave flow pattern unlike the symmetric structure in square ducts. Training on these cases enhances the model's ability to capture secondary flows, large separations, and their interactions.

B.3.2   Test cases

We select two groups to evaluate the trained specialist model for secondary flows with separation shown in Table. S3: test cases within the training categories (square ducts at $Re = 1100, 1800, 2600$, rectangular ducts with aspect ratio $= 5, 7, 10$, and periodic hills with slope scaling factor $\alpha = 0.8, 1.2, 1.5$) and a realistic three-dimensional diffuser case. All the test cases within training categories are detailed in Section. B.1.2 and we now focus on the 3D diffuser.

The 3D diffuser Cherry et al. (2008) is a challenging validation case for turbulence models. It features incompressible, asymmetric internal flow with strong adverse pressure gradients and significant Reynolds stress anisotropy. The resulting three-dimensional complex separation closely mirrors the flow behavior in practical diffusers, such as those found between compressors and combustors in jet engines. This case is therefore a valuable benchmark for evaluating a model's ability to predict corner separation and secondary flow dynamics.

The diffuser consists of an inlet channel, a diffuser section, and a straight outlet section. Both the upper wall and the sidewall expand. Boundary conditions prescribe a fully developed turbulent inflow with a bulk velocity $U_b = 1$m/s, giving a Reynolds number $Re = 10000$ based on inlet channel height. The diffuser test case is designed to capture three-dimensional separation and reattachment driven by wall expansion. Velocities are sampled at multiple wall-normal locations downstream of the expansion, as indicated by the red markers in Fig. S3, to resolve the separated shear layer and recirculation bubble. In addition to velocity profiles, the wall friction coefficient is monitored along the bottom wall to quantify separation onset and reattachment. In total, 64 observations are used, providing detailed information on both the mean flow field and wall-bounded response.

Table S3: Test cases for the specialist model of secondary flows with separation

Flow	Control parameter	Value	No. of cases
Square duct	Reynolds number	1100 1800 2600	3
Rectangular duct	Aspect ratio	5 7 10	3
Periodic hill	Slope scaling factor	0.8 1.2 1.5	3
3D diffuser	Reynolds number	10000	1
		Totals	**10**

C   Results

C.1   Unified foundation model

We train a unified foundation model on five canonical benchmark cases and evaluate it on 27 diverse test cases. The unified foundation model demonstrates strong generalization across all test categories and consistently reduces the misfit relative to the baseline k–ω model.

We introduce the normalized misfit, denoted $\hat{e}$, to quantify the improvement of the unified foundation model relative to the single-case trained model and to the baseline model; let e_{unified}, e_{single}, and e_{base} denote the misfits of the unified foundation model, the single-case trained model, and the baseline model, respectively, and define

$$\hat{e} = \frac{e_{\text{unified}} - e_{\text{single}}}{e_{\text{base}} - e_{\text{single}}}. \tag{S16}$$

By construction $\hat{e} = 0$ corresponds to performance equal to the single-case trained model and $\hat{e} = 1$ corresponds to the baseline; values $\hat{e} < 1$ indicate improvement relative to the baseline while values $\hat{e} > 1$ indicate degraded performance. For the plane channel case shown in Fig. S8, the unified foundation model attains $\hat{e}$ close to 0, indicating performance comparable to the single-case trained model; the baseline model has $\hat{e} = 1$ and the single-case trained model has $\hat{e} = 0$ by definition, respectively.

The radar plot shown in Fig. S9 summarizes normalized misfit $\hat{e}$ for all training and test cases and highlights the unified foundation model's generalization across four distinct flow categories. The model is trained on five canonical benchmark cases and evaluated on 27 diverse test cases; the plot condenses those results into a single view.

The plot layout makes interpretation straightforward: colored outer bands label the four flow categories – attached boundary layers, secondary flows, separated flows, and free-shear flows, while the radial axes correspond to individual test cases and are annotated with case parameters such as Reynolds number, aspect ratio, and angle of attack. The baseline corresponds to $\hat{e} = 1$ on every axis, training cases are indicated by markers, and the unified foundation model's performance is shown as the filled blue polygon.

Across the majority of axes the unified foundation model's polygon lies inside the baseline ring, indicating $\hat{e} \leq 1$ and therefore reduced misfit relative to the baseline. Training cases cluster near $\hat{e} = 0$, showing performance comparable to single-case trained models; a small number of separated flow cases approach $\hat{e} = 1$, indicating performance only comparable to the baseline.

These patterns demonstrate robust generalization: the unified foundation model reduces misfit broadly across attached boundary layers, secondary flows, free-shear flows and separated flows, while the radar plot also exposes specific cases where improvement is limited and where targeted refinement would be most beneficial.

C.1.1 Attached boundary layers

The unified foundation model matches the baseline k–ω model in five attached boundary layer cases, demonstrating strong accuracy and generalization across Reynolds numbers and geometries. We evaluate the model on channel flows at Re = 3300, 7890, 8×10^7, a turbulent flat plate, with results shown in Fig. S10 and two airfoil cases with low angles of attack (AOA = $1°, 5°$), with results shown in Fig. S14. Among these, the channel flow at Re = 3300 appears in the unified training set. For the airfoil cases at low angles of attack, where the flow remains fully attached, all models perform similarly.

Fig. S10 shows u^+ versus y^+ for ground truth (DNS data, black solid line), the baseline k–ω RANS prediction (red dashed line), and the unified foundation model performance (blue dash-dot line). Because the u^+–y^+ curve reflects momentum diffusion from the wall, it serves as a stringent benchmark for turbulence models across flow regimes. All four cases represent benchmarks where linear eddy viscosity models typically perform reliably.

The unified foundation model reproduces near-wall turbulence with high fidelity in the low and moderate Reynolds number channel flows. At Re = 3300, it closely follows the DNS curve, matching gradients in the near-wall region ($y^+ < 100$) and the logarithmic layer ($y^+ \approx 30$–300), where accurate turbulent transport is critical. At Re = 7890, it continues to match DNS trends and baseline model performance, including in regions where traditional models often overpredict velocities. This consistency shows the model's capability to deliver accurate predictions within the regimes most relevant to engineering applications.

The model generalizes effectively to channel flow at high Reynolds number and unseen zero pressure gradient flat plate. At Re = 8×10^7, all models align closely, but the unified foundation model offers marginal improvements, indicating successful generalization beyond its training range. In the zero pressure gradient flat plate as shown in Fig. S10(d), which differs fundamentally from channel flow, it maintains high accuracy and more effectively captures boundary layer growth than the baseline k–ω model.

These results highlight the unified foundation model's strong generalizability, showing that it captures geometry-agnostic, physically grounded features of near-wall turbulence and transfers them effectively across attached boundary layer flow scenarios.

C.1.2 Free-shear flow

The unified foundation model reproduces centerline velocity decay in a round jet with higher fidelity than the baseline k–ω model. We evaluate it on the same jet flow case used for unified training, with results shown in Fig. S11. Compared to high-fidelity DNS data, which show a gradual velocity decay downstream, the unified foundation model (blue dash-dot line) closely tracks the DNS curve across the entire range. It captures jet entrainment and turbulent mixing more accurately than the baseline k–ω model (red dashed line), particularly in the far-field region where the baseline model typically underpredicts the decay.

C.1.3 Secondary flows

The unified foundation model accurately predicts secondary flows across a wide range of duct configurations, generalizing beyond its training conditions. We evaluate it on eight test cases as shown in Fig. S12 and S13: four square ducts at decreasing Reynolds numbers (Re = 3500, 2600, 1800, 1100) and four rectangular ducts at increasing aspect ratios (AR = 3, 5, 7, 10). Among these, only the square duct at Re = 3500 appears in the training set; all other cases serve as generalization tests. For the square ducts, lower Reynolds numbers increase the sensitivity of secondary flow structures to turbulence model accuracy due to stronger viscous effects. Across all Reynolds number, the unified foundation model outperforms the k–ω baseline model, maintaining closer agreement with DNS results. In rectangular ducts, increasing the aspect ratio amplifies and distorts secondary flows, leading to anisotropic patterns that the baseline k–ω model struggles to capture. The unified foundation model captures both the amplification and spatial shifting of secondary vortices with aspect ratio, preserving their qualitative structure and quantitative magnitude. These results confirm that the model generalizes effectively to both Reynolds number and geometric extrapolation, demonstrating robustness for complex internal flows beyond its training set.

C.1.4 Separated flows

The unified foundation model improves prediction accuracy for many separation-dominated flows but shows limited gains in certain configurations. We evaluate the unified foundation model on four types of benchmark cases: periodic hill flows with different slope scaling factors $\alpha = 0.8, 1.0, 1.2, 1.5$, flow over a bump, flow over a curved step and airfoil cases at five high angles of attack (AOA = $9°, 11°, 14°, 16°, 18°$). Among these, the periodic hill flow with slope scaling factor $\alpha = 1.0$ and the airfoil flow at AOA = $14°$ appear in the unified training set. All four types

of cases are challenging for linear eddy viscosity models because the local equilibrium assumption and Boussinesq hypothesis fail in separated regions. Results are shown in Figs. S14, S15 and S16.

The unified foundation model generalizes flow separation mechanism across a wide range of angles of attack for S809 airfoil more accurately than the baseline k–ω model. We compare drag and lift coefficients (c_d and c_l) over varying angles, with results shown in Fig. S14 alongside ground truth. The unified foundation model closely follows experimental trends across all test angles of attack, particularly in the post-stall region where the baseline substantially overpredicts lift and underpredicts drag. As the angle increases, the baseline model fails to capture the sharp nonlinear rise in c_d and the drop in c_l caused by flow separation, most evident at AOA = 18°. The unified foundation model, in contrast, reproduces these separation-driven transitions with higher fidelity, underscoring its capacity to model untrained flow regimes.

The unified foundation model generalizes effectively across periodic hill flows with varying slope steepness, maintaining high accuracy in separated-flow predictions. In these cases (Fig. S15), steeper slopes intensify adverse pressure gradients, producing stronger separation and delaying reattachment, where the baseline k–ω model often fails. Across all slopes, the unified foundation model more accurately predicts separation and reattachment locations and matches the velocity profiles from ground truth. In the steepest case ($\alpha = 0.8$), it captures curvature and asymmetry absent in the baseline, demonstrating robustness to geometric variation beyond the training slope.

For the bump and curved step cases, which involve incipient flow separation, the unified foundation model offers no clear improvement over the baseline. In Fig. S16(a), the skin friction coefficient c_f distribution matches the baseline closely. In Fig. S16(b), the normalized streamwise velocity profiles at several downstream locations also align with the baseline, without significant enhancement. These findings suggest that the current unified foundation model lacks specialized capability for these configurations, motivating the development of a specialist model for separated flows. Such a model would rely on tailored training case selection to better capture the flow-specific physics through focused supervision.

C.1.5 Realistic 3D flows

The unified foundation model consistently outperforms the baseline model across both realistic 3D cases shown in Fig. S17. In the generic car, the baseline model fails to capture the full extent of the separated wake, producing a region that is significantly contracted relative to the ground truth, while the foundation model more closely aligns with the ground truth, albeit still underpredicting the wake size. In the 3D diffuser, the baseline model markedly overpredicts the wall skin friction c_f, maintaining elevated values far downstream, whereas the foundation model closely follows the gradual decay observed in the ground truth. Overall, the unified foundation model demonstrates a substantial improvement over the baseline, yielding better predictions for the separation region and the wall friction respectively.

C.2 Specialist turbulence model for separated flows

The specialist model for separated flows learns the corrections to the baseline model on the training cases and generalizes well within the training categories, extending even to the unseen generic car. To achieve this, we train the model to capture the physics of different levels of separation across various flows. We then evaluate the model within the training categories: periodic hills with different slope steepness and bumps with varying heights. In addition, we test it on a realistic three-dimensional flow, represented by a generic car with a slant angle of 40°.

C.2.1 Training results

The specialist model achieves improved performance on all training cases. Fig. S18 compares separation bubbles from the ground truth, the baseline k–ω model, and the specialist model across the three canonical training cases. In each case, the baseline overpredicts separation and deviates from the ground truth. After training, the specialist model places the onset and reattachment points closer to the ground truth and produces a separation region that better matches the observations.

C.2.2 Test results on flows within training categories

The specialist model generalizes well to flows within the training categories, outperforming the baseline. We evaluate bump configurations with heights of $h_b = 20, 26, 38,$ and $42\,\text{mm}$, as well as periodic hills with slope scaling factors of $\alpha = 0.8, 1.2,$ and 1.5 (see Fig. S4). Performance in predicting separation is quantified using the normalized misfit $\hat{e}$ and visualized on a radar chart in Fig. S19. For the bumps, the specialist model consistently reduces the error relative to the baseline, demonstrating robust predictive improvement across all test heights. For the periodic hills,

substantial error reductions are observed for slope-scaling factors of $\alpha = 1.0$, 1.2, and 1.5. However, the case with $\alpha = 0.8$ yields results comparable to the baseline. This exception arises from the sharper geometric transitions, which generate flow phenomena beyond the training dataset. Overall, the results show that the specialist model generalizes well to configurations closely related to its training set, but its performance cannot be guaranteed when the test case deviates too far from the training conditions.

C.2.3   Test results on generic car

To evaluate the specialist model's ability to extrapolate beyond its training domain, we apply it to the generic car to predict wake separation. In this case, the drag coefficient c_d is a key quantity of interest. Table S4 reports the drag coefficients predicted by different models. The baseline model substantially overpredicts the drag compared to the ground truth, yielding values with low predictive credibility. Both the unified foundation model and the specialist model significantly improve upon the baseline. The specialist model provides the most accurate estimate, lying closest to the ground truth, though it still shows a noticeable deviation in this case. Fig. S20(a) shows the wake, with the $U_x = 0$ contour line marking the separation region. The comparison of separation regions for the generic car reveals clear differences among the three models, baseline, unified foundation, and specialist, relative to the ground truth. The baseline model substantially underestimates the separation, producing a markedly contracted bubble. The unified foundation model improves the prediction by capturing a larger separation zone that extends further downstream, yet it still deviates from the ground truth. The specialist model achieves the best agreement, accurately reproducing the size of the separated region. The streamline pattern and the distribution of $g^{(1)}$ illustrate how the specialist model adapts its closure in separated flows (Fig. S20(b)). The streamlines delineate the separation bubble and associated vortex structures, while the $g^{(1)}$ field highlights regions where the model applies localized adjustments. Elevated values of $g^{(1)}$ appear near the wake, where the eddy viscosity ν_t Zhang et al. (2023)is reduced according to

$$\nu_t = -\frac{g^{(1)}k}{C_\mu \omega}, \tag{S17}$$

with $C_\mu = 0.09$ being a constant commonly used in eddy-viscosity models. This reduction facilitates separation and yields predictions that are closer to the ground truth.

Table S4: Comparison of drag coefficients among different models

Model	Drag coefficient
Ground truth	0.248
Baseline model	0.983
Unified foundation model	0.587
Specialist model	0.579

In summary, the specialist model for separated flows significantly improves separation prediction across the training cases, placing onset and reattachment points closer to the ground truth. When applied to unseen flows within the training categories, the model consistently reduces error relative to the baseline, demonstrating robust generalization. Finally, in the realistic three-dimensional generic car case, the specialist model sustains its improved performance by concentrating corrections near the separation wake, accurately capturing both the separation region and the associated flow structures.

C.3   Specialist turbulence model for secondary flows with separation

The specialist turbulence model for secondary flows with separation not only corrects the baseline k–ω model's overprediction of separation and reconstructs intricate corner-vortex patterns, but also generalizes well to the three-dimensional diffuser by accurately capturing the interaction between secondary flow and separation.

C.3.1   Training results

The specialist model improves performance across all training cases. For the periodic hill, which characterizes separated flow, Fig. S21(a) shows that it reduces the baseline's overprediction of separation and produces results closer to the ground truth. For the square and rectangular ducts, which characterize secondary flows, Fig. S21(b–c)

compares in-plane flows from the specialist model, the baseline, and the ground truth. The square duct exhibits symmetric corner vortices, while the rectangular duct shows asymmetric vortices: those along the horizontal wall expand in the spanwise direction, weakening those along the vertical wall. Streamline visualizations reveal these vortical structures and highlight the improvements achieved by the specialist model.

C.3.2 Test results on flows within training categories

We evaluate the specialist model's generalization within its training categories. The tests include periodic hills with slope scaling factors of 0.8, 1.2, and 1.5 to assess separated flows; square ducts at Reynolds numbers of 1100, 1800, and 2600; and rectangular ducts with aspect ratios of 5, 7, and 10 to assess secondary flows.

The specialist model consistently outperforms the baseline model. The radar plot in Fig. S22 illustrates these improvement trends. Rectangular ducts with aspect ratios of 5, 7, and 10 show substantial gains over the baseline. For the square duct, trained at Re = 3500, performance decreases monotonically as the Reynolds number drops: Re = 2600 shows moderate improvement, Re = 1800 shows weak improvement, and Re = 1100 shows essentially no improvement. Periodic hills with slope scaling factors $\alpha = 0.8, 1.2, 1.5$ also outperform the baseline, although the gains are smaller than those at the training case $\alpha = 1.0$, indicating modest sensitivity to wall curvature and pressure gradient strength.

C.3.3 Test results on three-dimensional diffuser

Velocity contours across key cross sections show that the specialist model predicts separation behavior with the highest accuracy. To assess its performance on realistic three-dimensional flows that couple secondary flow and separation, we evaluate an asymmetric 3D diffuser. Several representative spanwise and streamwise planes are selected for detailed analysis. Fig. S23 presents streamwise velocity contours U_x/U_b with separation lines defined by $U_x/U_b = 0$ at $x/h_d = 2, 5, 8, 12$, and 15. In the ground truth, the separated region originates near the upper wall and grows downstream while remaining attached to the inclined side wall. In contrast, the baseline model predicts large separation along the inclined side wall and fails to capture the primary upper-wall separation. The specialist model suppresses spurious side-wall separation and relocates the separation line to the upper wall, closely matching the ground truth. The unified foundation model performs between the baseline and specialist models: at $x/h_d = 12$ and 15, it strengthens the upper-wall separation and reduces the side-wall separation, but still deviates from the true separation pattern. Fig. S24 further shows that, along the spanwise cross section near the inclined side wall, the specialist model more accurately captures the propagation of upper-wall separation and its reattachment at the diffuser exit compared to both the unified foundation model and the baseline model.

Overall, the specialist model for secondary flow with separation not only learns the flow mechanisms individually, achieving improved performance within the training categories, but also handles their interaction. Its strong results on the 3D diffuser, eliminating spurious inclined side-wall separation and correctly predicting upper-wall separation, demonstrate generalization beyond the training manifold.
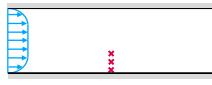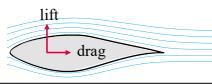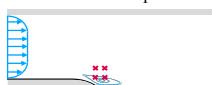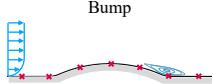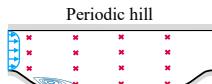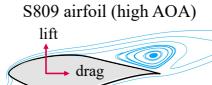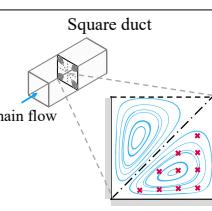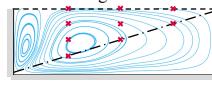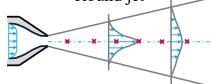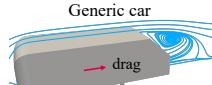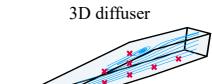
Categories	Flows	Observations	Number of observations	Control parameter	Value
Attached Boundary Layers	Zero pressure gradient flat plate	Velocities	2	Reynolds number	5×10^6
	Plane channel	Velocities	4	Reynolds number	3300 7890 8×10^7
	S809 airfoil (low AOA)	Lift, drag	2	Angle of attack	1° 5°
Separated Flows	Curved step	Velocities	27	Reynolds number	13700
	Bump	Wall friction	42	Height	20 mm 26 mm 31 mm 38 mm 42 mm
	Periodic hill	Velocities	66	Slope scaling factor	0.8 1.0 1.2 1.5
	S809 airfoil (high AOA)	Lift, drag	2	Angle of attack	9° 11° 14° 16° 18°
Secondary Flows	Square duct	Velocities	193	Reynolds number	1100 1800 2600 3500
	Rectangular duct	Velocities	35	Aspect ratio	3 5 7 10
Free-shear Flow	Round jet	Velocities	25	Reynolds number	5601
Realistic 3D Flows	Generic car	Drag	1	Slant angle	40°
	3D diffuser	Velocities, wall friction	64	Reynolds number	10000
		Totals	**32**		

Figure S3: 32 cases with associated observation data used for multi-objective learning and evaluation. Blue arrows indicate the direction of flow, and blue lines represent streamlines. Red arrows show the forces, while red crosses mark representative observation points included schematically for demonstration only. Attached boundary layers include flat plate Jespersen et al. (2016), plane channel Kim et al. (1987) at three Reynolds numbers and S809 airfoil Somers (1997) at two low angles of attack; separated flows include S809 airfoil at five high angles of attack, curved step Bentaleb et al. (2012), bump of five heights Matai and Durbin (2019), and periodic hill Xiao et al. (2020) of four slope scaling factors; secondary flows include square duct Pinelli et al. (2010) at four Reynolds numbers and rectangular duct Vinuesa et al. (2018) of four aspect ratios; free-shear flow includes round jet Bridges and Wernet (2010); realistic 3D flows include generic car Lienhart and Becker (2003) and 3D diffuser Cherry et al. (2008).
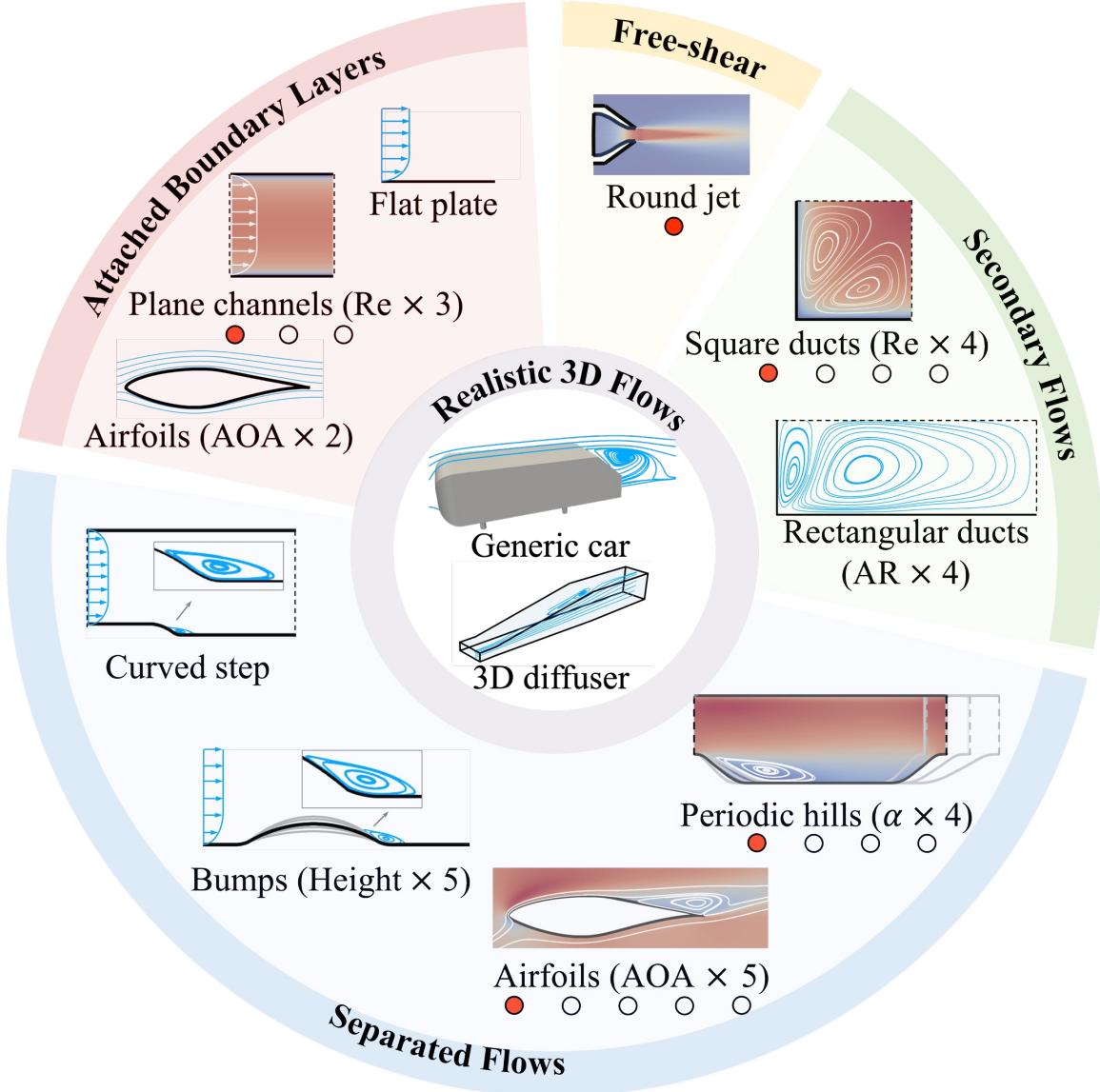
Figure S4: Canonical and realistic flows used to train and evaluate the unified foundation model, totaling 32 cases. The canonical flows are broadly grouped into four categories: attached boundary layers, separated flows, secondary flows, and free-shear flow, each represented as sector of the ring, and the two realistic 3D flows featuring multiple mechanisms are placed in the inner circle. Five representative flows (depicted with contours and marked with red circles) are used as training set, while the remaining 27 cases are used for testing and evaluation. The test set comprises 25 canonical flows within the training categories and two realistic 3D flows from automotive and aerospace aerodynamics applications. Detailed specifications are provided in Fig. S3. The range of flows highlights that the multi-objective learning framework can accommodate diverse flow mechanisms in training and generalize across both trained and unseen configurations, including two realistic 3D cases.
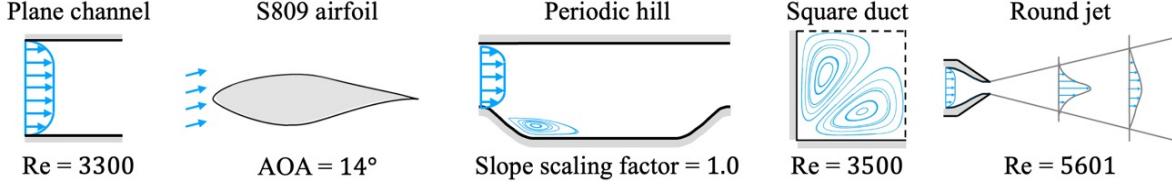
Figure S5: Training cases for the unified foundation model. Training covers five representative flow configurations: plane channel with Reynolds number Re = 3300, S809 airfoil with angle of attack AOA = 14°, periodic hill with slope scaling factor $\alpha = 1.0$, square duct at Re = 3500, and round jet.

Figure S6: Training and cases for the specialist turbulence model for separated flows. Training includes three benchmark configurations: a curved step, a bump with 31 mm height and a periodic hill with slope scaling factor of 1.0.

Figure S7: Training cases for the specialist turbulence model for secondary flows with separation. Training includes three benchmark cases: a square duct at Re = 3500, a rectangular duct with aspect ratio of 3, and a periodic hill with slope scaling factor of 1.0.

Figure S8: An example comparing the normalized misfit $\hat{e}$ between the single-case trained model, the unified foundation model, and baseline k–ω models. The unified foundation model converges to a low normalized misfit that lies between the baseline model, fixed at one, and the single-case trained model, fixed at zero.
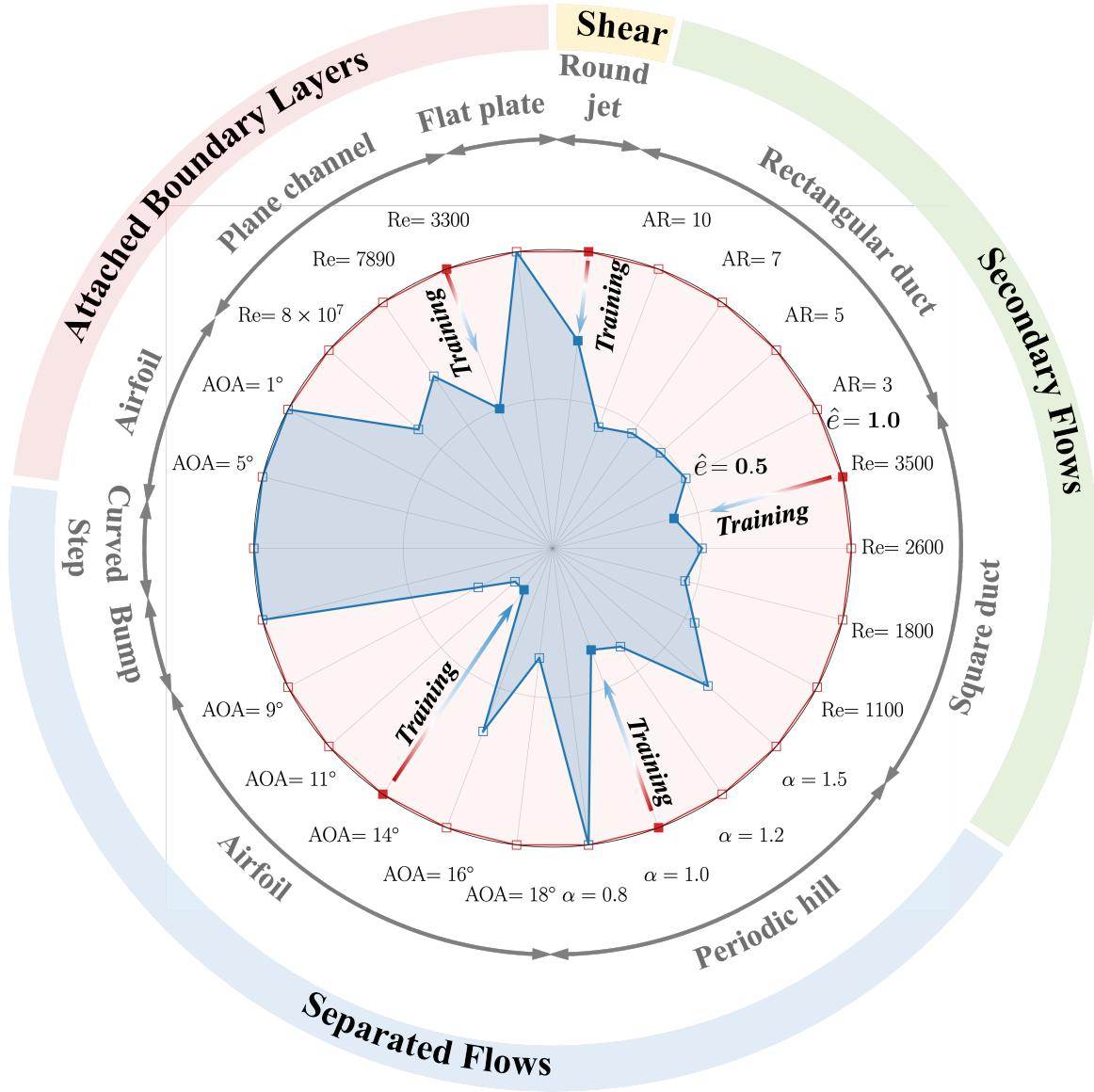
Figure S9: Radar chart comparison of the baseline and the unified foundation model performances across four benchmark flow categories. The radial plot displays relative performance (i.e., normalized misfit $\hat{e}$) across multiple flow configurations, including square and rectangular ducts, periodic hills with varying slope scaling factors, and airfoil at angles of attack. The red curve with open squares represents the baseline model, while the blue curve with filled squares denotes the unified foundation model. Lower radial distances indicate better performance. Since the results among the bumps are roughly the same, only the bump with a height of 31 mm is selected as the representative. The radar plot highlights improved generalization of the unified foundation model on diverse flow scenarios.
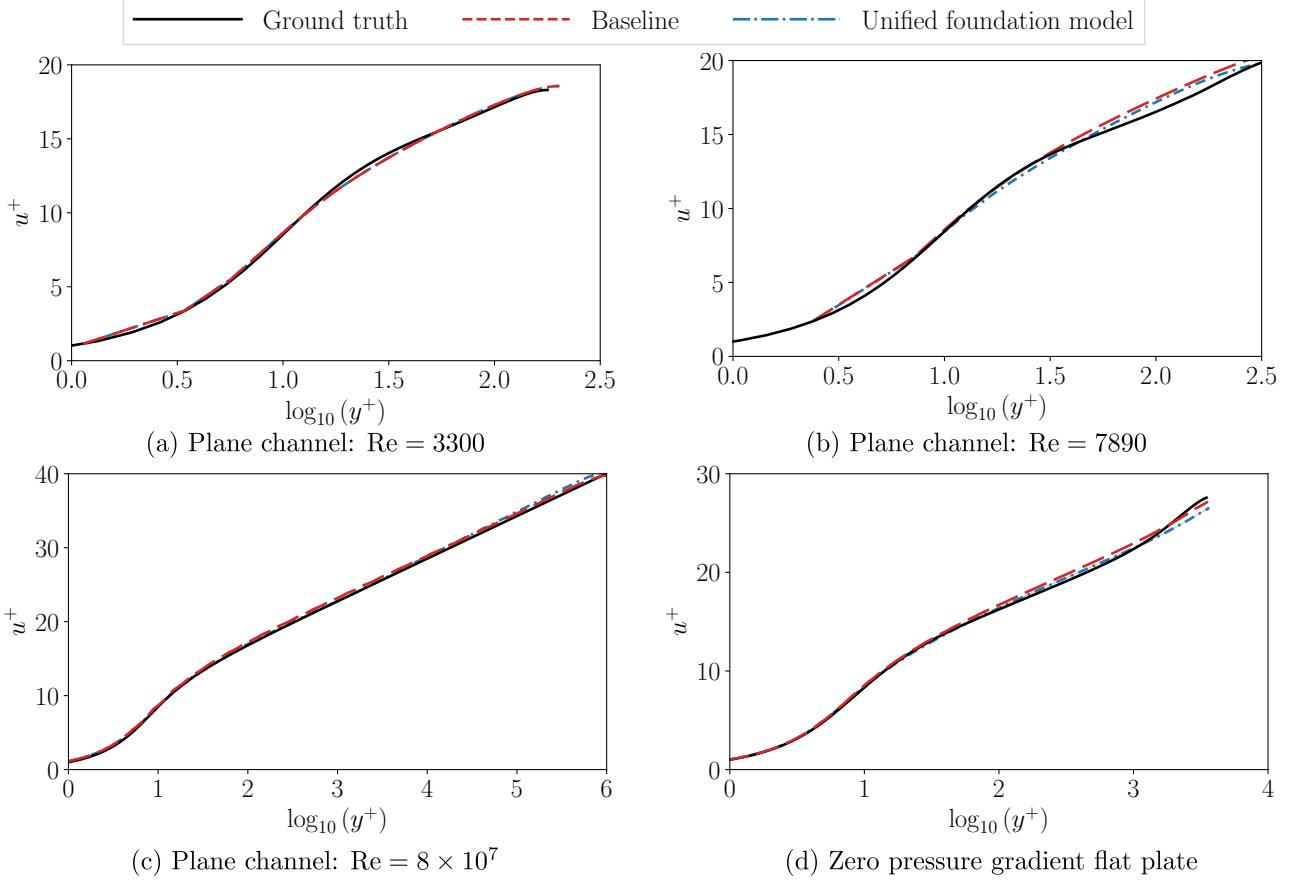
Figure S10: Comparison of u^+ versus y^+ (log scale) across four wall-bounded turbulent flows: (a) plane channel at Re = 3300, (b) plane channel at Re = 7890, (c) plane channel at Re = 8×10^7, and (d) zero pressure gradient flat plate. The ground truth (DNS data, black solid line), the baseline k–ω RANS results (red dashed line), and the unified foundation model results (blue dash-dot line) are shown. The unified foundation model closely matches the DNS profiles, demonstrating strong generalization across flow regimes and Reynolds numbers.
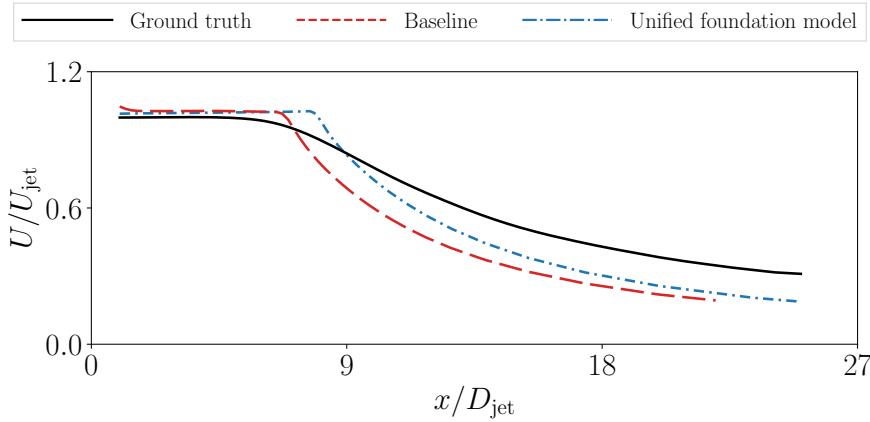
Figure S11: Centerline velocity of a turbulent jet normalized by the jet exit velocity U/U_{jet} as a function of downstream distance normalized by jet diameter x/D_{jet}. The ground truth (DNS data, black solid line), the baseline k–ω RANS results (red dashed line), and the unified foundation model results (blue dash-dot line) are shown. The unified foundation model matches the DNS profiles much closer than the baseline k–ω model.
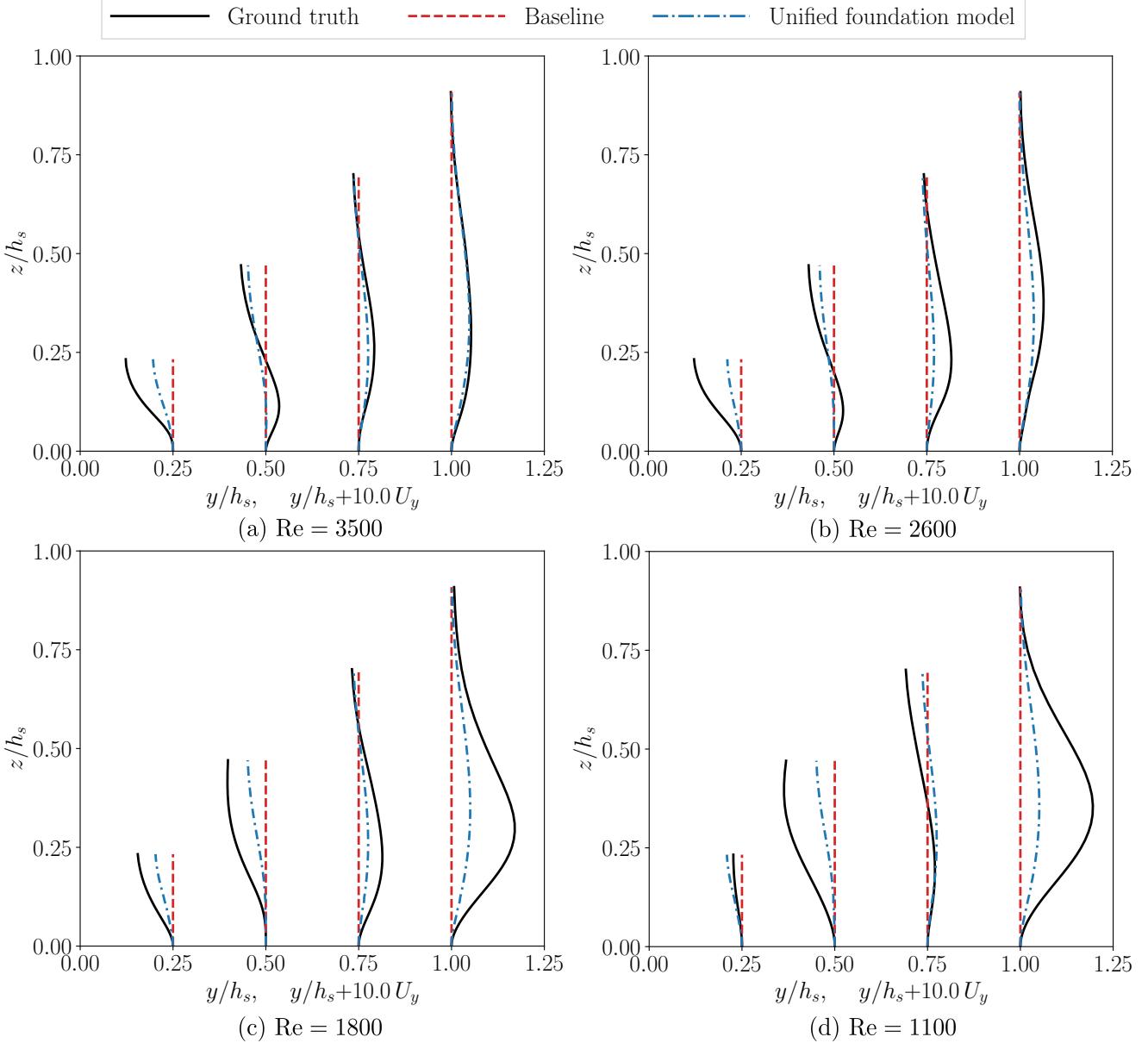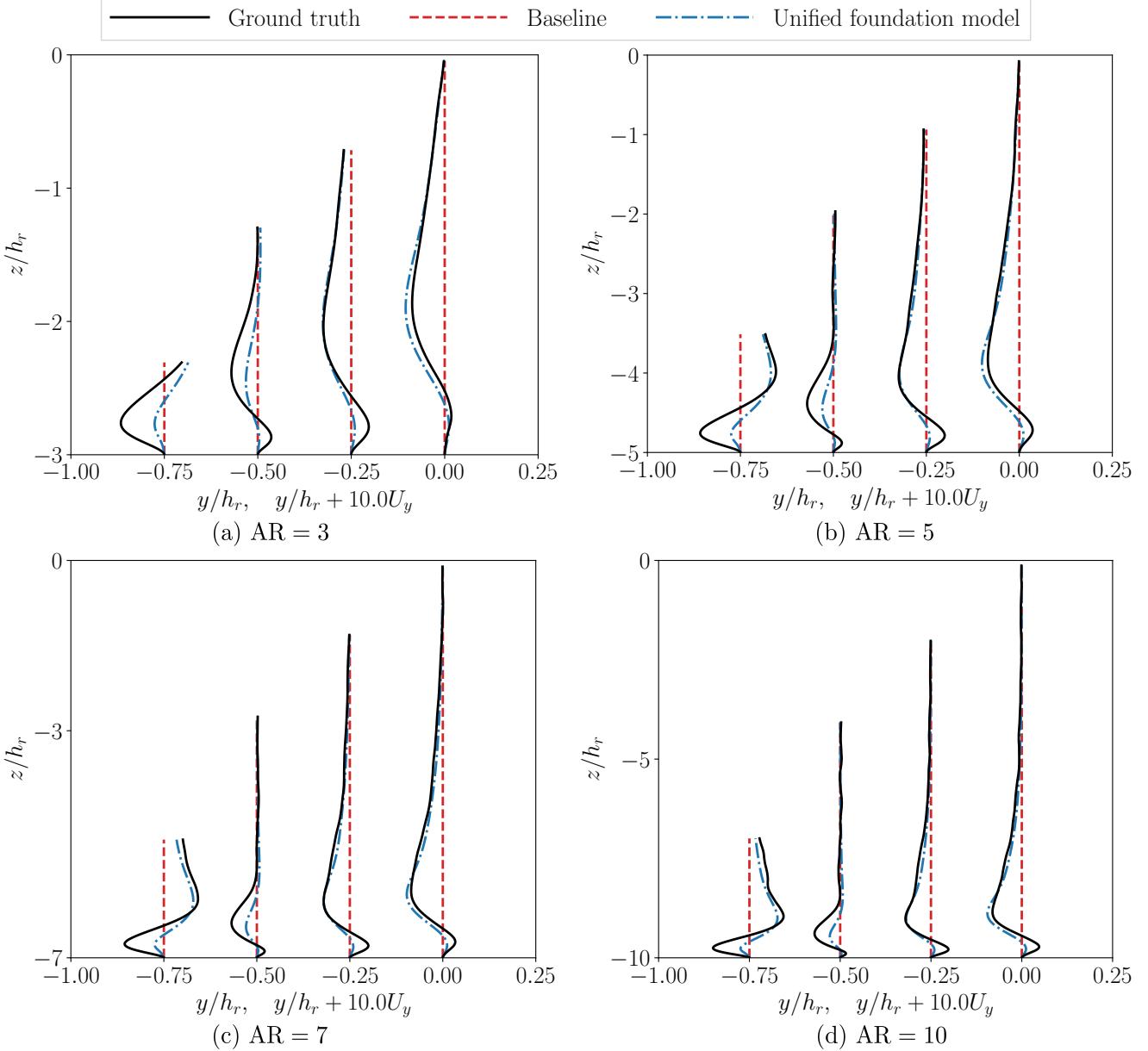
Figure S12: Comparison of secondary flow performances in square ducts using the ground truth (DNS data), the baseline k–ω RANS results, and the unified foundation model results. Predictions for square duct flows at four Reynolds numbers (Re = 3500, 2600, 1800, 1100), where Re = 3500 is the training case for the unified foundation model. Each panel plots cross-sectional velocity profiles (U_y) along multiple z/h_s locations, where h_s denotes half the height of the square duct. The unified foundation model consistently improves over the baseline k–ω model and approximates DNS with higher fidelity. Horizontal offsets of $+10.0U_y$ are applied for visualization clarity.
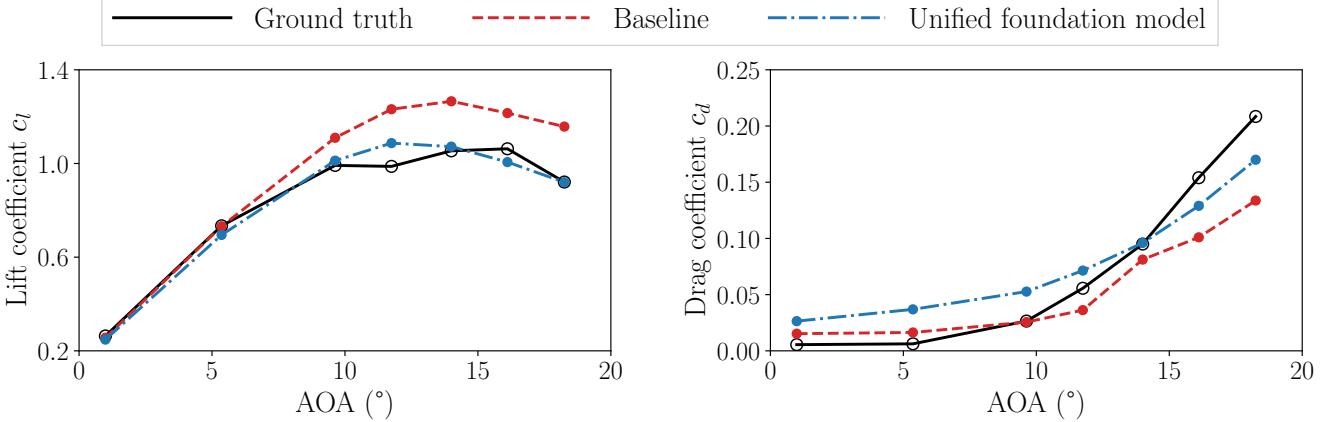
Figure S13: Comparison of secondary flow predictions in rectangular ducts using the ground truth (DNS data), the baseline k–ω RANS results, and the unified foundation model results. Predictions for rectangular ducts with varying aspect ratios (AR = 3, 5, 7, 10). Each panel plots cross-sectional velocity profiles (U_y) along multiple z/h_r locations, where h_r denotes half the height of the rectangular duct. The unified foundation model consistently improves over the baseline k–ω model and approximates DNS with higher fidelity. Horizontal offsets of $+10.0U_y$ are applied for visualization clarity.

Figure S14: Comparison of lift coefficient c_l (left) and drag coefficient c_d (right) across seven different angles of attack. The ground truth (experimental data, black solid line), the baseline k–ω RANS results (red dashed line), and the unified foundation model results (blue dash-dot line) are shown. The unified foundation model captures the lift more accurately than the baseline, while for drag it slightly overpredicts at low angles but follows the sharp rise near stall more faithfully than the baseline, which consistently underpredicts.

(a) Periodic hill, $\alpha = 0.8$

(b) Periodic hill, $\alpha = 1.0$

(c) Periodic hill, $\alpha = 1.2$

(d) Periodic hill, $\alpha = 1.5$
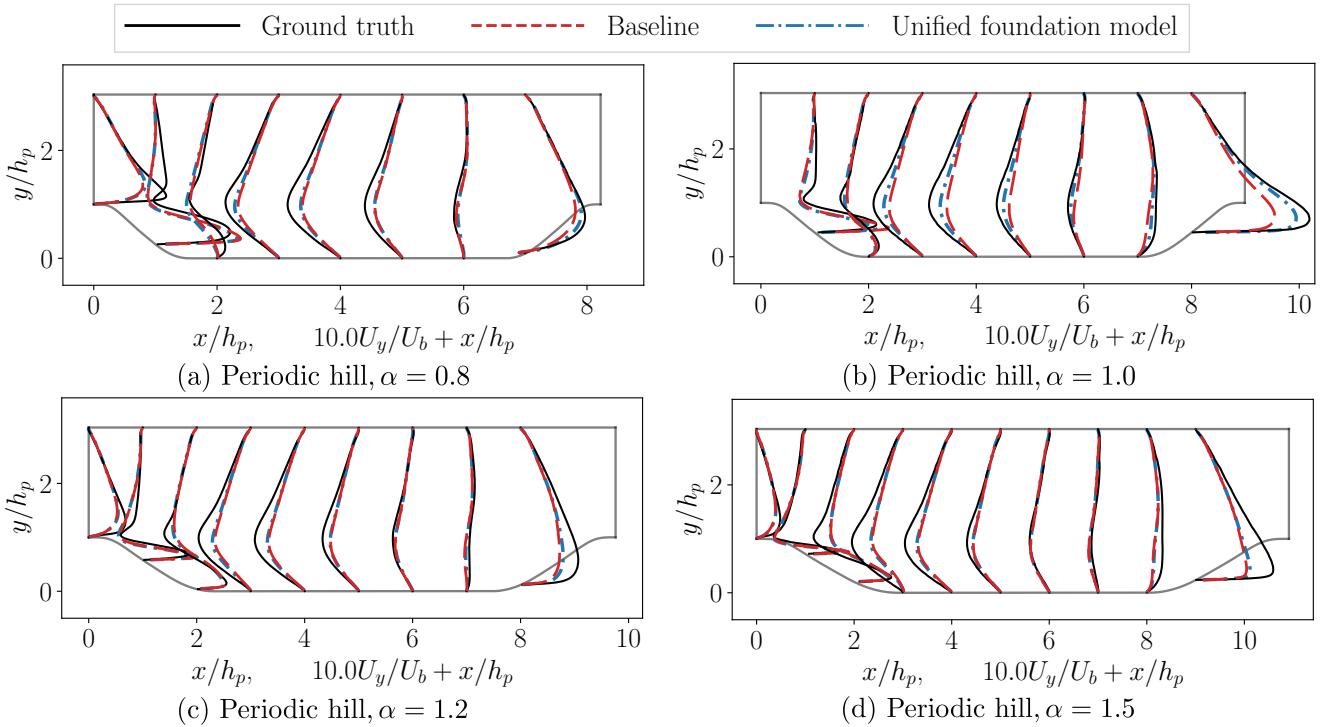
Figure S15: Comparison of vertical velocity profiles U_y normalized by the bulk velocity U_b, plotted as $10\,U_y/U_b + x/h_p$ versus normalized height y/h_p at various streamwise locations x/h_p for different slope scaling factors in a separated turbulent boundary layer. The panels represent results for ramps with slope scaling factor of (a) $\alpha = 0.8$, (b) $\alpha = 1$, (c) $\alpha = 1.2$, and (d) $\alpha = 1.5$. The black solid lines represent the ground truth (DNS data), red dashed lines indicate predictions from the baseline k–ω model, and blue dash-dot lines denote results from the unified foundation model. The unified foundation model consistently provides improved agreement with DNS data across slope configurations, particularly in regions of strong separation and reattachment.
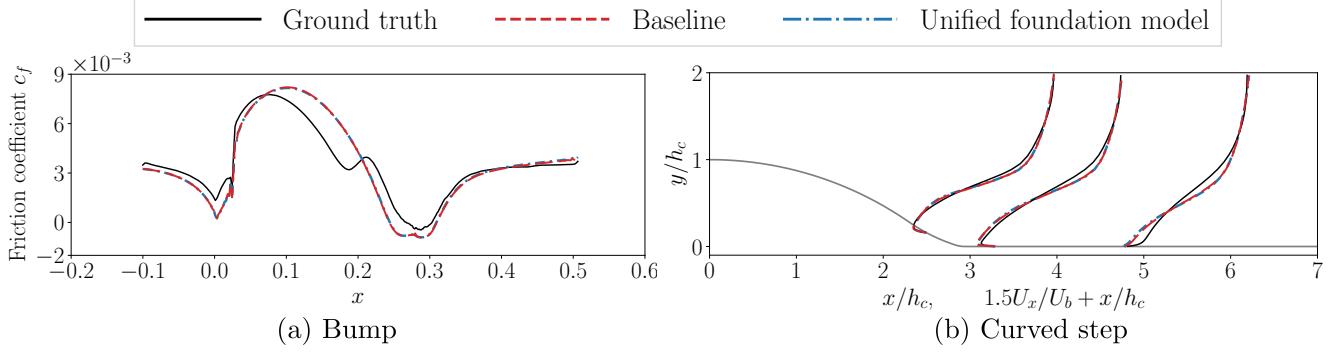
(a) Bump

(b) Curved step

Figure S16: Comparison of model performances for bump flow and curved step flow. (a) Streamwise distribution of the skin friction coefficient c_f. The ground truth (DNS data, solid black line), the baseline k–ω RANS results (red dashed line), and the unified foundation model results (blue dash-dot line) are shown. (b) Streamwise velocity profiles U_x normalized by the bulk velocity U_b, plotted as $1.5\,U_x/U_b + x/h_c$ versus normalized height x/h_c at various streamwise locations x/h_c for the curved step, where h_c is the height of the curved step. The unified foundation model performs comparably to the baseline in both cases without providing notable improvements.
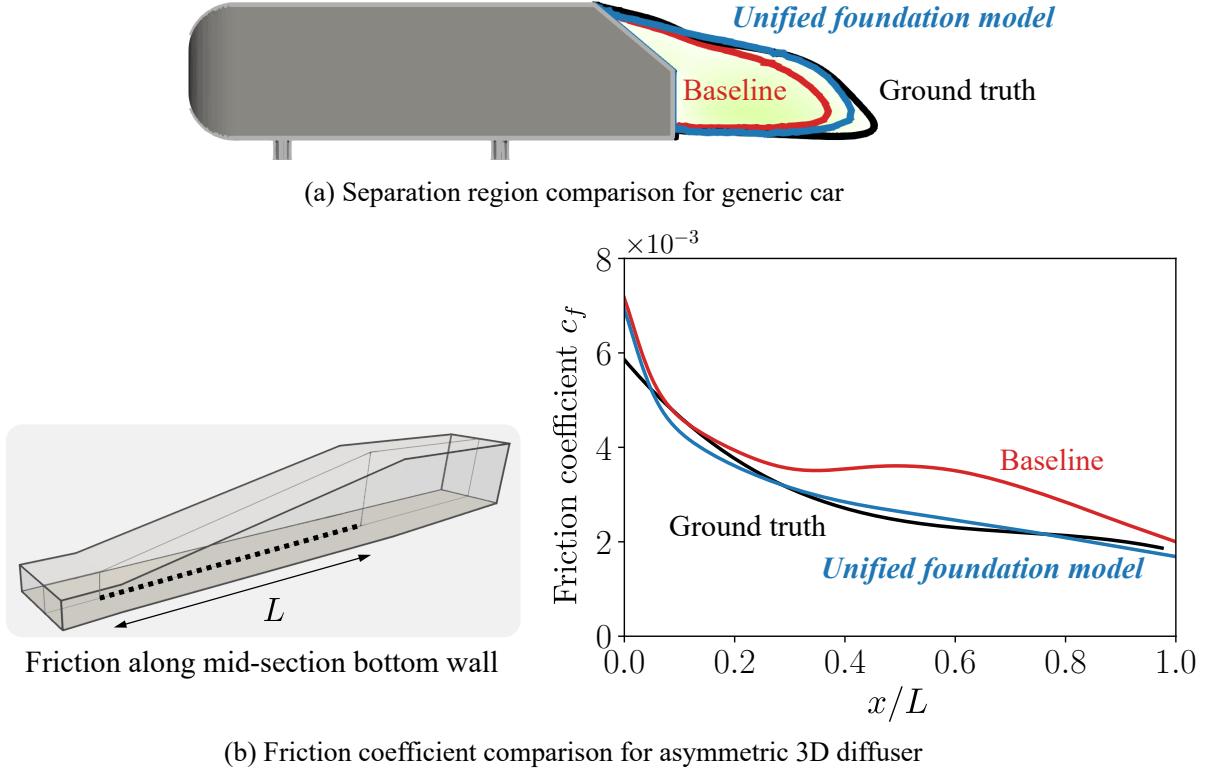

(a) Separation region comparison for generic car

Friction along mid-section bottom wall

(b) Friction coefficient comparison for asymmetric 3D diffuser

Figure S17: Comparison of (a) the separation region for the generic car and (b) the friction coefficient c_f for the asymmetric 3D diffuser. In panel (a), the solid contours depict the wake, with the $U_x = 0$ contour marking the separation region predicted by each model. The unified foundation model predicts the separation region more accurately than the baseline. In panel (b), the friction coefficient is sampled along the mid-section bottom wall to compare the models' predictions. For this case, the unified foundation model also aligns better with the ground truth friction along the bottom wall, whereas the baseline overpredicts friction in the downstream region.

	(a) Periodic hill	(b) Bump	(c) Curved step

Ground truth

Baseline

Specialist model

$-0.2\ 0 \quad 1.2$
U_x/U_b

$-0.1\ 0 \quad 1$
U_x/U_{inlet}

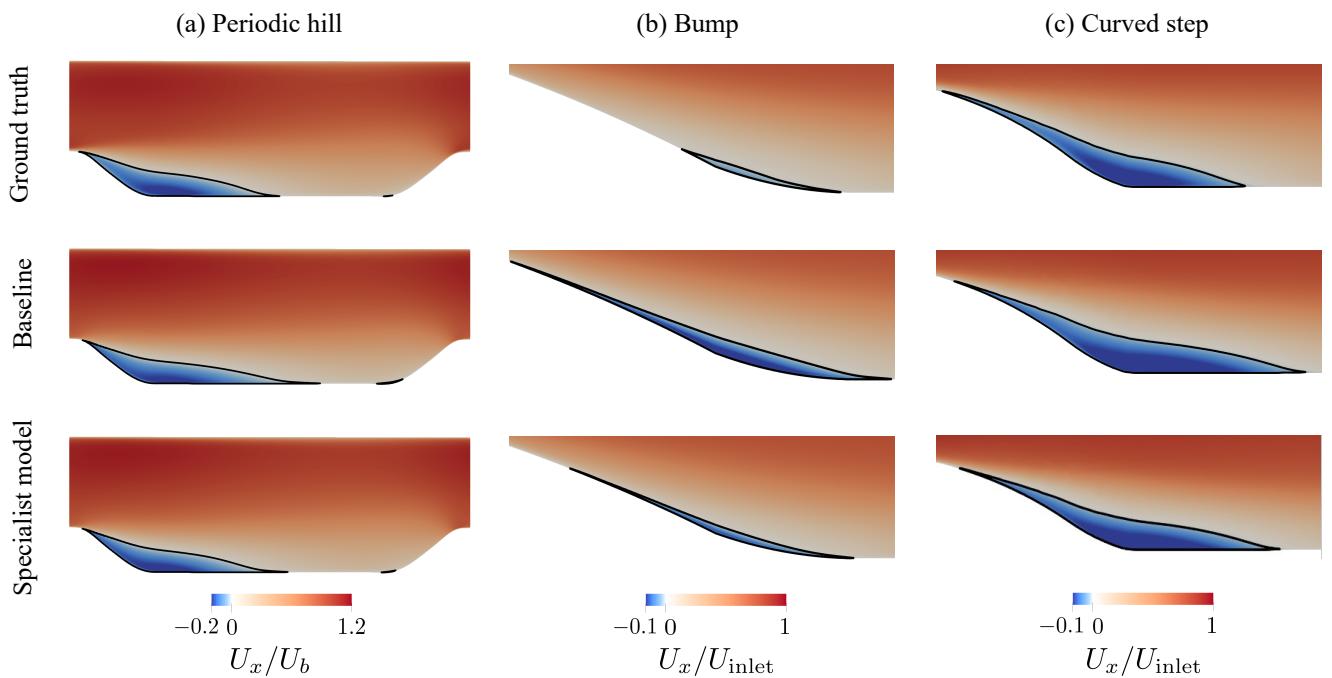$-0.1\ 0 \quad 1$
U_x/U_{inlet}

Figure S18: Streamwise velocity contours for three canonical separated flows: (a) periodic hill, (b) bump, and (c) curved step. For each case, the ground truth (row 1), the baseline model (row 2), and the specialist model for separated flows (row 3) are shown. Contours represent the non-dimensional streamwise velocity (U_x/U_b in the periodic hill, U_x/U_{inlet} in the bump and curved step), with blue indicating reverse flow. The black contour lines corresponds to $U_x = 0$, marking the separated recirculation bubble. Compared with the ground truth, the baseline model over predicts the separation region length of all cases, and the specialist model improves this deficiency.
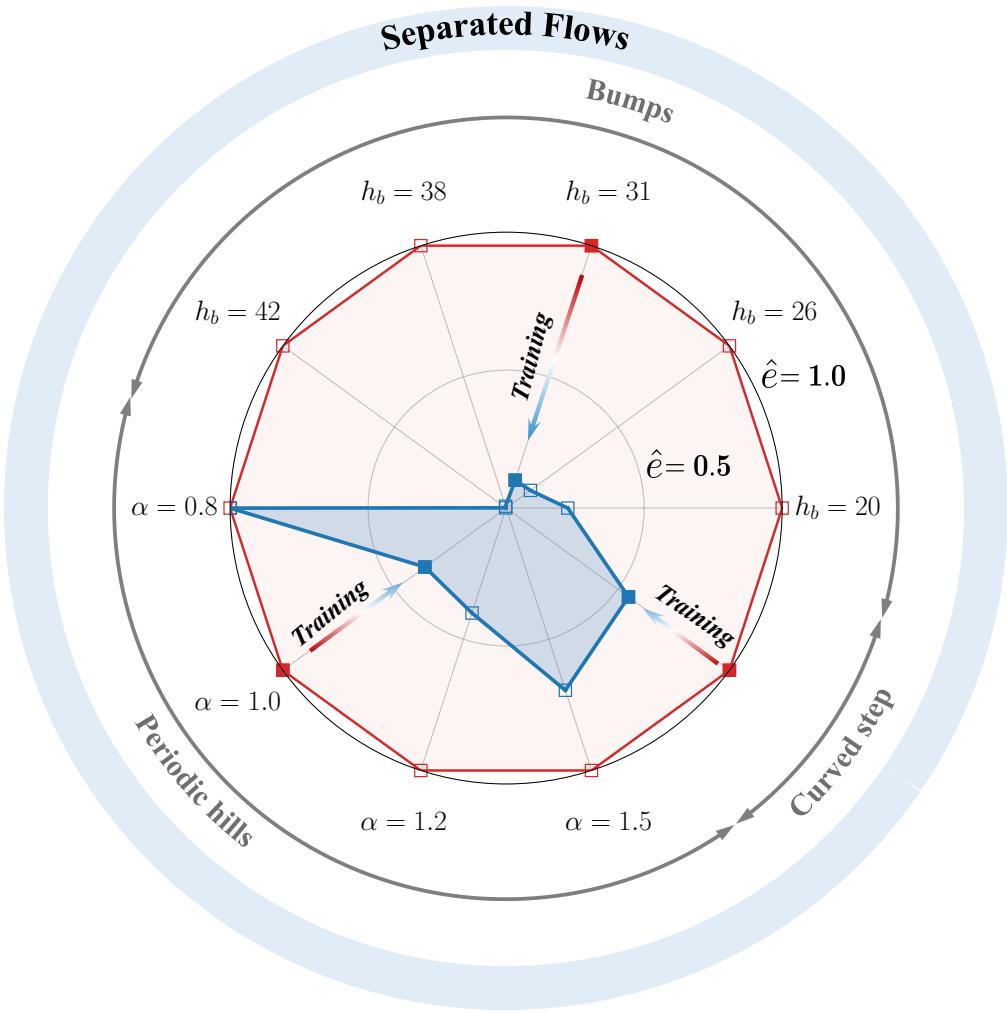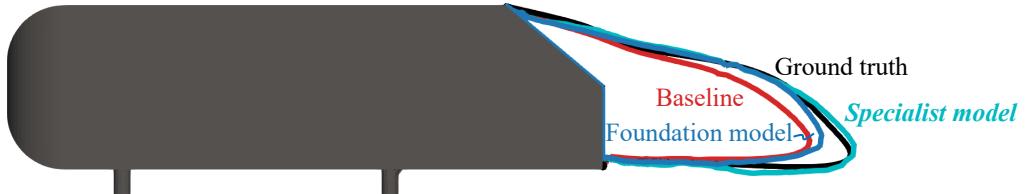
Figure S19: Radar chart comparing the specialist and baseline models on separated flows within the training categories (including the training cases). Three flows are considered: bumps with heights of $h_b = 42, 38, 31, 26, 20$, the curved step, and periodic hills with slope scaling factors of $\alpha = 0.8, 1.0, 1.2, 1.5$. The radial distance represents the normalized misfit $\hat{e}$; points on the unit circle ($\hat{e} = 1$) indicate baseline performance by definition, while points closer to the center indicate better accuracy. Across all cases, the specialist model achieves improved performance, or at least matches the baseline.

(a) Separated region comparison for generic car

(b) Streamline and distribution of $g^{(1)}$ from specialist model

Figure S20: (a) Solid contour lines indicate the separation region, defined by $U_x = 0$, for different models. Compared to the baseline model (red line), both trained models yield predictions that more closely follow the ground truth data (black line), with the specialist model exhibiting the greatest enhancement in accuracy. (b) The improved performance of the specialist model is associated with increased values of the linear tensor basis coefficient $g^{(1)}$ around the wake separation of the generic car, compared to the baseline value of -0.09.
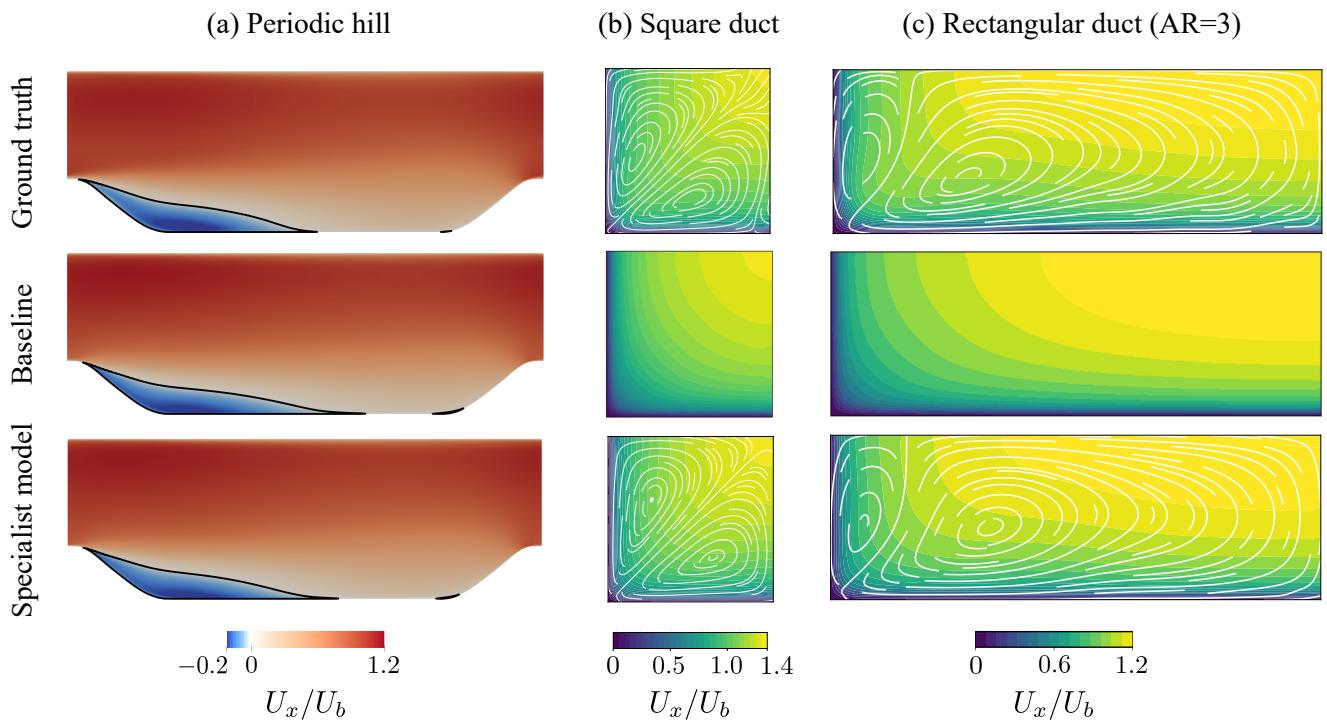

Figure S21: Comparison of different models on the training flows. (a) Periodic hill: contours of the non-dimensional streamwise velocity U_x/U_b, with black contour lines marking the $U_x = 0$ isoline that delineates the separated region. (b) Square duct and (c) rectangular duct with aspect ratio AR $= 3$: contours of U_x/U_b on the cross-sections, overlaid with white streamlines that visualize the in-plane vortices. The specialist model corrects the separation prediction for the periodic hill and captures the secondary flows in the square and rectangular ducts.
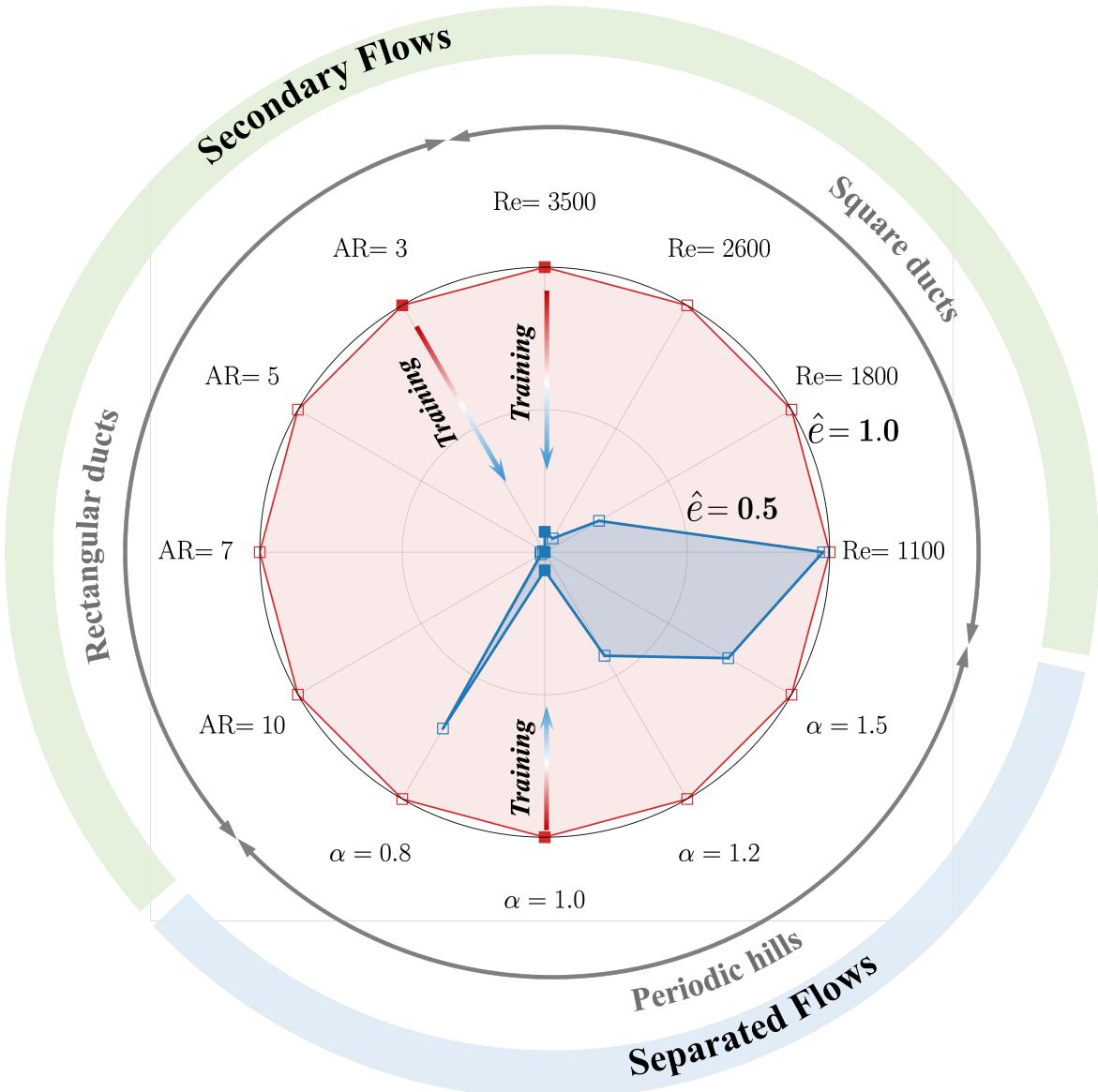
40

Figure S22: Radar chart comparing the baseline model (red) and the specialist model (blue) on flows within the training categories. The specialist model shows substantial improvements over the baseline and performs at least comparably across all cases. Secondary flows: square duct at Reynolds numbers Re = 1100, 1800, 2600, and 3500 and rectangular duct with aspect ratios AR = 3, 5, 7, and 10. Separated flows: periodic hills with slope scaling factors $\alpha = 0.8$, 1.0, 1.2, and 1.5. The radial coordinate represents the normalized prediction error, where a smaller radius indicates better performance.
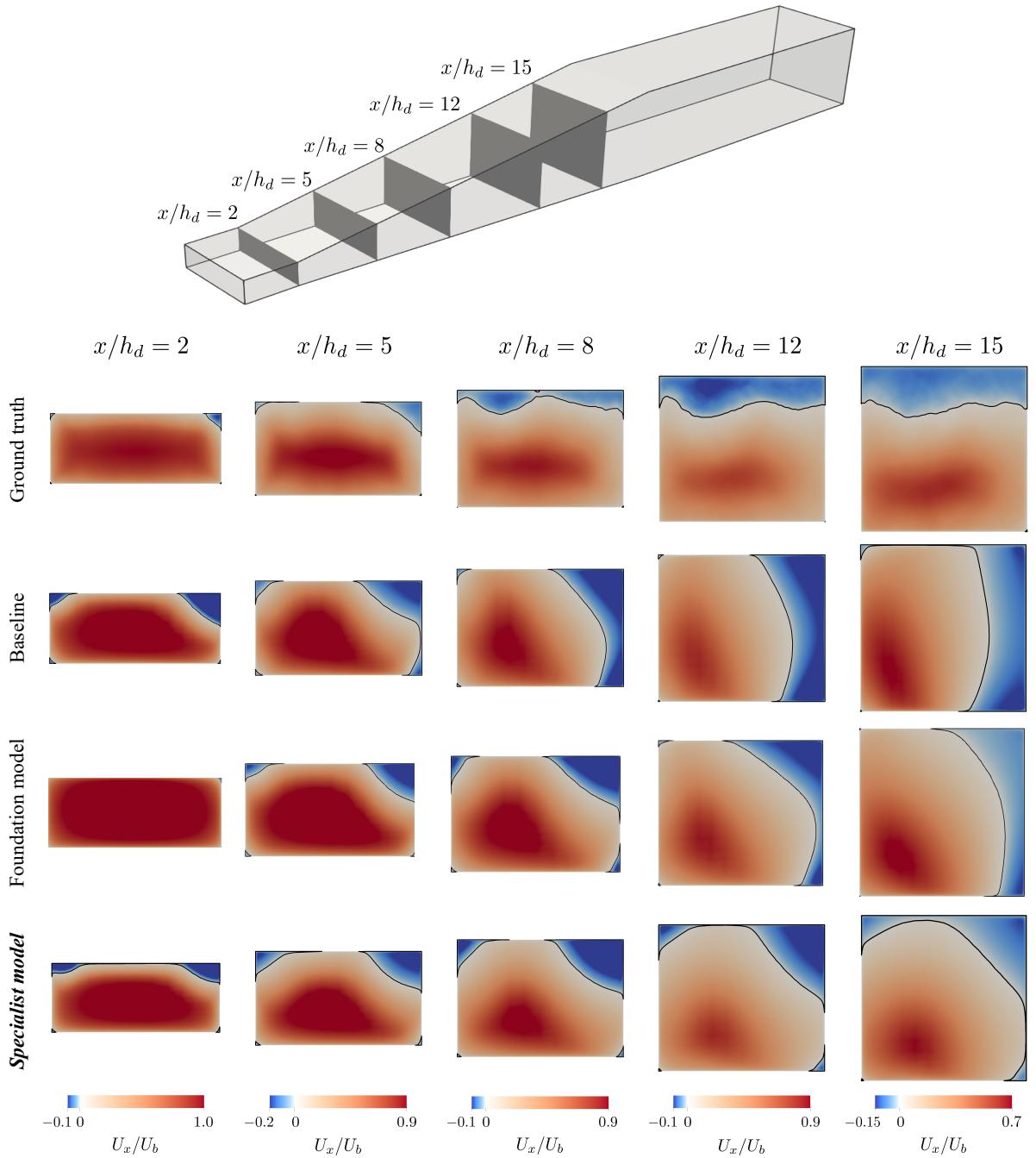
Figure S23: Cross-sectional streamwise velocity contours for the 3D diffuser at $x/h_d = 2$, 5, 8, 12, and 15, where h_d is the inlet height. Columns show downstream stations, while rows present (top to bottom) the ground truth, the baseline model, the unified foundation model, and the specialist model. Colors represent the non-dimensional velocity U_x/U_b (blue = reverse flow), with the black contour line marking the $U_x = 0$ isoline that delineates separation. The ground truth shows separation attached to the upper wall and growing downstream. The baseline predicts spurious side-wall separation and misses the primary upper-wall separation, while the unified foundation reduces side-wall separation but still deviates from the truth. The specialist suppresses side-wall errors, relocates separation to the upper wall, and most closely matches the ground truth, clearly outperforming both the baseline and the unified foundation model.
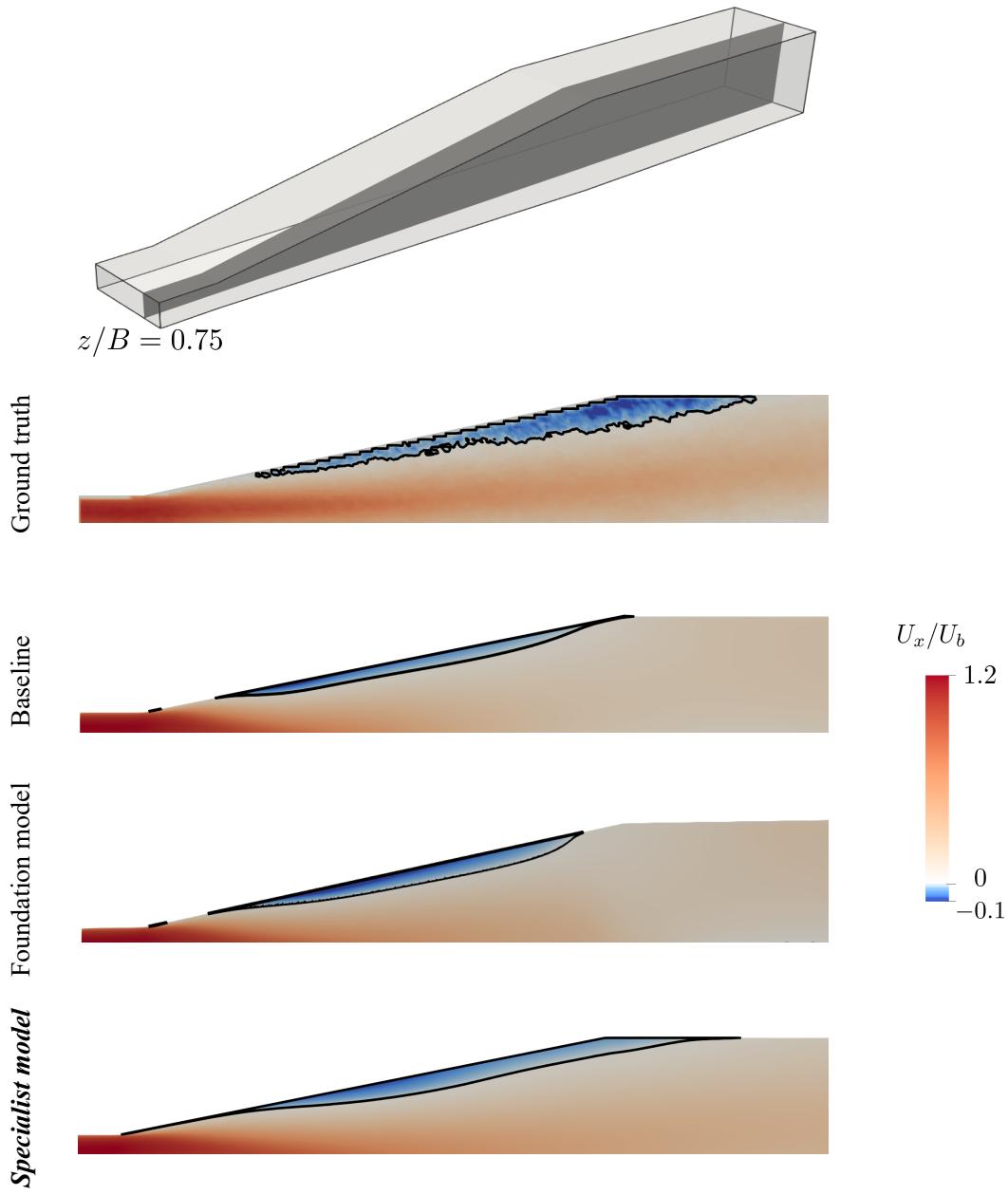
Figure S24: Streamwise velocity field on a spanwise cross section of the 3D diffuser at $z/B = 0.75$, where B is the inlet width. Panels (top to bottom) show the ground truth, the baseline model, the unified foundation model, and the specialist model. Contours depict U_x/U_b, with blue indicating flow separation, and the black contour line marks the $U_x = 0$ isoline outlining the separation bubble along the upper wall. The specialist model most accurately captures the propagation of upper-wall separation and its reattachment at the diffuser exit, outperforming both the baseline and the unified foundation model.

References

Erij Alaya, Cornelia Grabe, and Bernhard Eisfeld. Evolutionary algorithm applied to differential Reynolds stress model for turbulent boundary layer subjected to an adverse pressure gradient. In *AIAA Aviation 2022 Forum*, page 3337, 2022.

Yacine Bentaleb, Sylvain Lardeau, and Michael A Leschziner. Large-eddy simulation of turbulent boundary layer separation from a rounded step. *Journal of Turbulence*, (13):N4, 2012.

James Bridges and Mark P. Wernet. Establishing consensus turbulence statistics for hot subsonic jets. In *16th AIAA/CEAS Aeroacoustics Conference*, 2010. doi:10.2514/6.2010-3751.

Tyler Buchanan, Monica Lăcătuş, Alastair West, and Richard P Dwight. Data-driven RANS closures using a relative importance term analysis based classifier for 2D and 3D separated flows. *arXiv preprint arXiv:2504.06758*, 2025.

Zhao Chen, Vijay Badrinarayanan, Chen-Yu Lee, and Andrew Rabinovich. Gradnorm: Gradient normalization for adaptive loss balancing in deep multitask networks. In *International Conference on Machine Learning*, pages 794–803. PMLR, 2018. doi:10.48550/arXiv.1711.02257.

Soufiane Cherroud, Xavier Merle, Paola Cinnella, and Xavier Gloerfelt. Space-dependent aggregation of stochastic data-driven turbulence models. *Journal of Computational Physics*, 527:113793, 2025.

Erica M Cherry, Christopher J Elkins, and John K Eaton. Geometric sensitivity of three-dimensional separated flows. *International Journal of Heat and Fluid Flow*, 29(3):803–811, 2008.

Jean-Antoine Désidéri. Multiple-gradient descent algorithm (MGDA) for multiobjective optimization. *C. R. Math.*, 350(5-6):313–318, 2012. doi:10.1016/j.crma.2012.03.014.

Karthik Duraisamy, Gianluca Iaccarino, and Heng Xiao. Turbulence modeling in the age of data. *Annual Review of Fluid Mechanics*, 51:357–377, 2019. doi:10.1146/annurev-fluid-010518-040547.

Yuan Fang, Yaomin Zhao, Fabian Waschkowski, Andrew SH Ooi, and Richard D Sandberg. Toward more general turbulence models via multicase computational-fluid-dynamics-driven training. *AIAA J.*, 61(5):2100–2115, 2023. doi:10.2514/1.J062572.

Thomas B Gatski and Charles G Speziale. On explicit algebraic stress models for complex turbulent flows. *J. Fluid Mech.*, 254:59–78, 1993. doi:10.1017/S0022112093002034.

Jiequn Han, Xu-Hui Zhou, and Heng Xiao. An equivariant neural operator for developing nonlocal tensorial constitutive models. *J. Computational Physics*, 488:112243, 2023. doi:10.1016/j.jcp.2023.112243.

Martin Jaggi. Revisiting Frank-Wolfe: Projection-free sparse convex optimization. In *International Conference on Machine Learning*, pages 427–435. PMLR, 2013.

Dennis C Jespersen, Thomas H Pulliam, and Marissa Lynn Childs. Overflow turbulence modeling resource validation results. Technical Report No. ARC-E-DAA-TN35216, 2016.

Alex Kendall, Yarin Gal, and Roberto Cipolla. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7482–7491, 2018. doi:10.1109/CVPR.2018.00781.

John Kim, Parviz Moin, and Robert Moser. Turbulence statistics in fully developed channel flow at low Reynolds number. *Journal of Fluid Mechanics*, 177:133–166, 1987.

Harold W. Kuhn and Albert W. Tucker. Nonlinear programming. In Jerzy Neyman, editor, *Proceedings of the Second Berkeley Symposium on Mathematical Statistics and Probability*, pages 481–492, Berkeley, 1951. University of California Press.

Brian Edward Launder, G Jr Reece, and W Rodi. Progress in the development of a Reynolds-stress turbulence closure. *J. Fluid Mech.*, 68(3):537–566, 1975. doi:10.1017/S0022112075001814.

Hermann Lienhart and Stefan Becker. Flow and turbulence structure in the wake of a simplified car model. *SAE Transactions*, pages 785–796, 2003.

Julia Ling and Jeremy Templeton. Evaluation of machine learning algorithms for prediction of regions of high Reynolds averaged Navier Stokes uncertainty. *Physics of Fluids*, 27(8), 2015.

Julia Ling, Andrew Kurzawski, and Jeremy Templeton. Reynolds averaged turbulence modelling using deep neural networks with embedded invariance. *J. Fluid Mech.*, 807:155–166, 2016. doi:10.1017/jfm.2016.615.

John L Lumley. Toward a turbulent constitutive relation. *J. Fluid Mech.*, 41(2):413–434, 1970.

Jonas Luther and Patrick Jenny. Non-linear super-stencils for turbulence model corrections. *Communications Physics*, 8(1):233, 2025.

Racheet Matai and Paul Durbin. Large-eddy simulation of turbulent flow over a parametric set of bumps. *Journal of Fluid Mechanics*, 866:503–525, 2019.

Florian R Menter and Alexey Matyushenko. Generalized k–ω (GEKO) two-equation turbulence model. *AIAA Journal*, 2025.

Eric J Parish and Karthik Duraisamy. A paradigm for data-driven predictive modeling using field inversion and machine learning. *J. Comput. Phys.*, 305:758–774, 2016. doi:10.1016/j.jcp.2015.11.012.

Alfredo Pinelli, Markus Uhlmann, Atsushi Sekimoto, and Genta Kawahara. Reynolds number dependence of mean flow structure in square duct turbulence. *Journal of Fluid Mechanics*, 644:107–122, 2010. doi:10.1017/S0022112009992242.

Stephen B Pope. A more general effective-viscosity hypothesis. *J. Fluid Mech.*, 72(2):331–340, 1975. doi:10.1017/S0022112075003382.

Christopher L Rumsey, Gary N Coleman, and Li Wang. In search of data-driven improvements to RANS models applied to separated flows. In *AIAA Scitech 2022 Forum*, number 0937, 2022.

CL Rumsey and GN Coleman. NASA symposium on turbulence modeling: Roadblocks, and the potential for machine learning. Technical Report NASA/TM-20220014930, NASA, 2022.

Martin Schmelzer, Richard P. Dwight, and Paola Cinnella. Data-driven deterministic symbolic regression of nonlinear stress-strain relation for rans turbulence modelling. In *48th AIAA Fluid Dynamics Conference*, 2018. doi:10.2514/6.2018-2900.

Ozan Sener and Vladlen Koltun. Multi-task learning as multi-objective optimization. *Adv. Neural Inf. Process. Syst.*, 31, 2018. doi:10.48550/arXiv.1810.04650.

Anand Pratap Singh and Karthik Duraisamy. Using field inversion to quantify functional errors in turbulence closures. *Physics of Fluids*, 28(4):045110, 2016.

Jeffrey P Slotnick, Abdollah Khodadoust, Juan Alonso, David Darmofal, William Gropp, Elizabeth Lurie, and Dimitri J Mavriplis. CFD vision 2030 study: A path to revolutionary computational aerosciences. Technical Report NASA/CR-2014-218178, NASA, 2014.

Dan M Somers. Design and experimental results for the S809 airfoil. Technical Report No. NREL/SR-440-6918, National Renewable Energy Lab.(NREL), Golden, CO (United States), 1997.

Charles G Speziale. On nonlinear k–l and k–ε models of turbulence. *J. Fluid Mech.*, 178:459–475, 1987.

Ricardo Vinuesa, Philipp Schlatter, and HM Nagib. Secondary flow in turbulent ducts with increasing aspect ratio. *Physical Review Fluids*, 3(5):054606, 2018.

Jian-Xun Wang, Jin-Long Wu, and Heng Xiao. Physics-informed machine learning approach for reconstructing Reynolds stress modeling discrepancies based on DNS data. *Physical Review Fluids*, 2(3):034603, 2017. doi:10.1103/PhysRevFluids.2.034603.

Jack Weatheritt and Richard Sandberg. A novel evolutionary algorithm applied to algebraic modifications of the rans stress–strain relationship. *J. Comput. Phys.*, 325:22–37, 2016. doi:10.1016/j.jcp.2016.08.015.

Chenyu Wu, Shaoguang Zhang, and Yufei Zhang. Development of a generalizable data-driven turbulence model: Conditioned field inversion and symbolic regression. *AIAA Journal*, 63(2):687–706, 2025.

Jin-Long Wu, Heng Xiao, and Eric Paterson. Physics-informed machine learning approach for augmenting turbulence models: A comprehensive framework. *Physical Review Fluids*, 3(7):074602, 2018. doi:10.1103/PhysRevFluids.3.074602.

Heng Xiao and Paola Cinnella. Quantification of model uncertainty in RANS simulations: A review. *Progress in Aerospace Sciences*, 108:1–31, 2019. doi:10.1016/j.paerosci.2018.10.001.

Heng Xiao, Jin-Long Wu, Sylvain Laizet, and Lian Duan. Flows over periodic hills of parameterized geometries: A dataset for data-driven turbulence modeling from direct simulations. *Computers & Fluids*, 200:104431, 2020.

Xin-Lei Zhang, Carlos. A Michelén-Ströfer, and Heng Xiao. Regularized ensemble Kalman methods for inverse problems. *J. Computational Physics*, 416:109517, 2020. doi:10.1016/j.jcp.2020.109517.

Xin-Lei Zhang, Heng Xiao, Solkeun Jee, and Guowei He. Physical interpretation of neural network-based nonlinear eddy viscosity models. *Aerosp. Sci. Technol.*, 142:108632, 2023. doi:10.1016/j.ast.2023.108632.

Xu-Hui Zhou, Jiequn Han, and Heng Xiao. Frame-independent vector-cloud neural network for nonlocal constitutive modeling on arbitrary grids. *Comput. Methods Appl. Mech. Eng.*, 388:114211, 2022. doi:10.1016/j.cma.2021.114211.

Ping Zhu, Kwun Yip Fung, Xuejin Zhang, Jun A Zhang, Jian-Wen Bao, Chuan-Kai Wang, Bin Liu, Zhan Zhang, Lucas Harris, Kun Gao, et al. Toward a unified parameterization of three dimensional turbulent transport in high resolution numerical weather prediction models. *npj Climate and Atmospheric Science*, 8(1):223, 2025.