In **turbulent flow**, we incorporated **RANS solver** in the **ML** model training, leading to better velocity and pressure prediction, even with limited data. We found **probabilistic** predictions, particularly in regions susceptible to model errors are indispensible.

Probabilistic Machine Learning based Turbulence Model Learning with a Differentiable Solver

Plain language summary

Turbulence is a crucial physical characteristic of a broad range of fluid flows. Grasping this occurrence is vital for intricate designs, environmental simulations, and a myriad of engineering uses. Inspite of the computational advances, approximations like Reynolds-averaged Navier–Stokes (RANS) persist as indispensable for industrial purposes, where precision heavily relies on turbulence closure models. The present research demonstrated that including the RANS solver in the training process can improve the prediction quality even with limited training data. This underscored the impact augmentation of training data with information that can be extracted from the physical simulator can have. Furthermore, we demonstrated a method to account for the involved uncertainties, that are unavoidable when any sort of modeling or learning with finite data is involved. This deep integration of differentiable and probabilistic programming frameworks involving physical simulators is the need of the hour for scientific machine learning. This can revolutionize many fields in physics and engineering like fluid mechanics, molecular dynamics, particle physics, cosmology, material science, drug discovery, to name a few.

Motivation

Simulating fluid dynamics turbulence through Direct Numerical Simulation (DNS), which mandates fine resolution of the Navier-Stokes equations, is prohibitively expensive. Reynolds-averaged Navier-Stokes (RANS) models are the industry standard as it offers a more economical approach to predict mean flow properties. However it hinges on a closure model for the Reynolds stress (RS) tensor [1]. We propose a novel probabilistic, data-driven framework [2] for learning the closure model from **sparse velocity and pressure observations.**

Our framework employs the following:

- A discrete, adjoint-based **differentiable RANS solver** to enable model-consistent, gradient-based learning.
- The RS closure model consists of a parametric part that is expressed with an **invariant neural network** [1], to which a **(latent)** ***stochastic*** **discrepancy tensor** field is added in order to account for the insufficiency of the parametric part.
- A fully Bayesian formulation that enables the **quantification of uncertainties** and their propagation to the predictive estimates (mean velocities, pressure and other relevant quantities.)
- A **prior model** that activates the discrepancy term only in regions of the problem domain where the parametric model is insufficient.

Method Overview

The Reynolds-averaged Navier-Stokes (RANS) equations to obtain the mean velocity and pressure:

$$u_j \frac{\partial u_i}{\partial x_j} - \nu \frac{\partial^2 u_i}{\partial x_j \partial x_j} + \frac{1}{\rho}\frac{\partial p}{\partial x_i} = -\frac{\partial \boxed{\langle \tilde{u}_i \tilde{u}_j \rangle}}{\partial x_j}$$

$$\frac{\partial u_i}{\partial x_i} = 0$$

- Reynolds Stress tensor $\boldsymbol{\mathcal{T}}$
- Unclosed
- Needs modeling

The proposed Reynolds Stress model consists of two parts. An invarient neural network based part $\boldsymbol{\tau}_{\boldsymbol{\theta}}(\boldsymbol{u})$ (as proposed in [1]) and a latent stochastic random variable $\boldsymbol{\epsilon}_\tau$ to account for model uncertainty:

$$\boldsymbol{\tau} = \boldsymbol{\tau}_{\boldsymbol{\theta}}(\boldsymbol{u}) + \boldsymbol{\epsilon}_\tau$$

The neural network based part is given by:

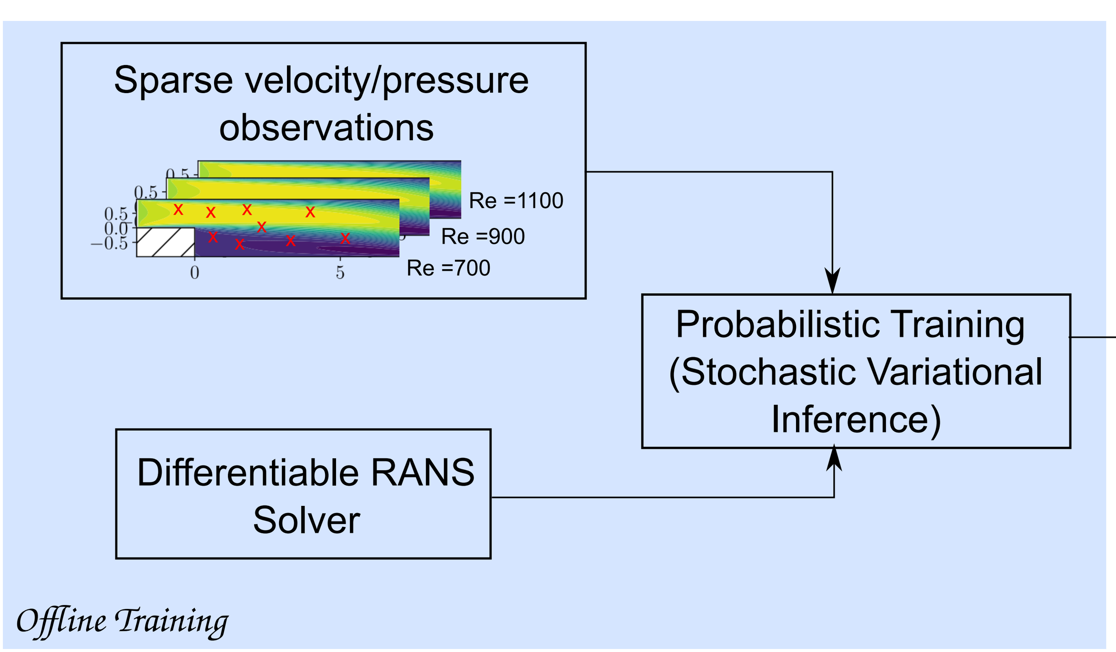
$$\boldsymbol{b}_\theta = \sum_{i=1}^{10} G_{\boldsymbol{\theta}}^{(i)}(\underbrace{\mathcal{I}_1 ... \mathcal{I}_5}_{\substack{Scalar \\ Invarients}})\boldsymbol{\mathcal{T}}^{(i)}; \quad \boldsymbol{\tau}_{\boldsymbol{\theta}} = 2k\boldsymbol{b}_\theta + \frac{2k}{3}\boldsymbol{I};$$

where $\mathcal{T}$ are tensors basis functions and G being scalar coefficient functions (represented by the neural network).
$\mathcal{I}, \mathcal{T} = f(higher\ order \nabla \boldsymbol{u})$

For dimentionality reduction, we employ a piece-wise constant representation $\boldsymbol{\epsilon}_\tau = \boldsymbol{W}\boldsymbol{E}_{\boldsymbol{\tau}}$.(**W** is a Boolean matrix.$dim(\boldsymbol{E}_\tau) << dim(\boldsymbol{\epsilon}_\tau)$)
Sparsity promoting ARD prior for $\boldsymbol{E}_\tau$ i.e, $\boldsymbol{E}_\tau = \boldsymbol{0}$ as long as the right RS is provided by the $\boldsymbol{\tau}_{\boldsymbol{\theta}}(\boldsymbol{u})$ model.

$$p(\boldsymbol{E}_\tau|\boldsymbol{\Lambda}) = \prod_{J=1}^{N_d} p(\boldsymbol{E}_{\tau,J}|\boldsymbol{\Lambda}^{(J)}) = \prod_{J=1}^{N_d} \mathcal{N}\left(\boldsymbol{E}_{\tau,J}\ |\boldsymbol{0},\ diag(\boldsymbol{\Lambda}_J)^{-1})\right)$$

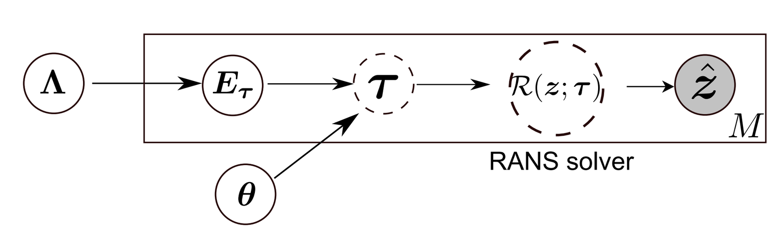
Gamma Hyperprior



With sparse observations of velocity and pressure, we obatin the posterior:

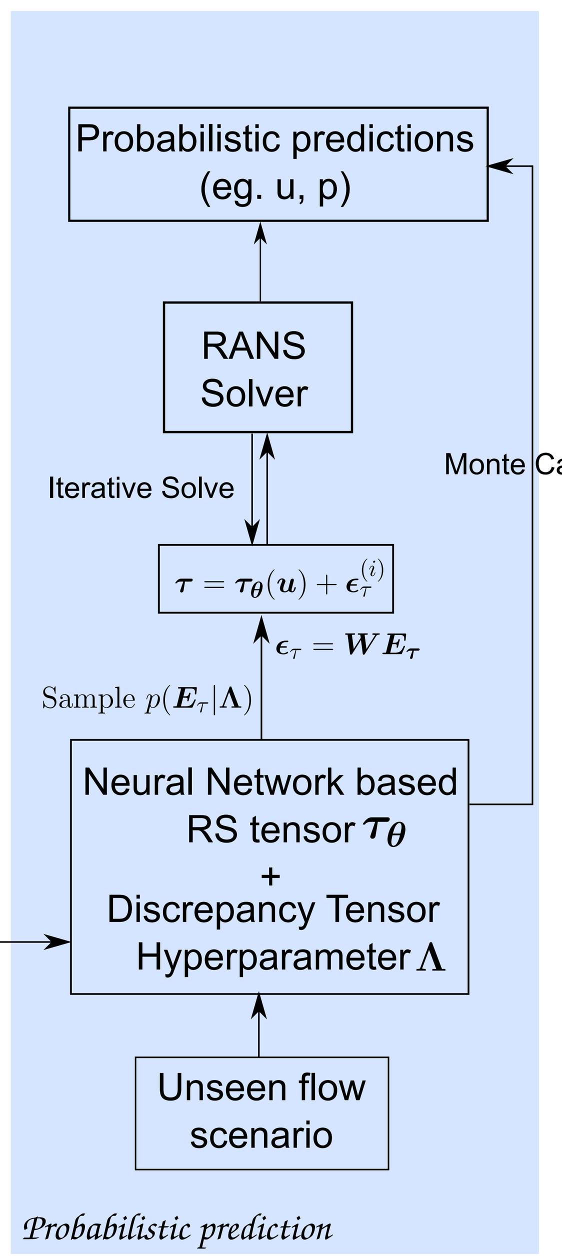
$$p(\boldsymbol{\theta}, \boldsymbol{E}_\tau^{(1:M)}, \boldsymbol{\Lambda}|\ \mathcal{D}) \propto \overbrace{p(\mathcal{D}\ |\ \boldsymbol{\theta}, \boldsymbol{E}_\tau^{(1:M)})}^{Likelihood}\ \overbrace{p(\boldsymbol{E}_\tau^{(1:M)}|\boldsymbol{\Lambda})\ p(\boldsymbol{\theta})\ p(\boldsymbol{\Lambda})}^{prior}$$

$$= \left(\prod_{m=1}^{M} p(\hat{\boldsymbol{z}}^{(m)}\ |\ \boldsymbol{\theta}, \boldsymbol{E}_\tau^{(m)})\ p(\boldsymbol{E}_\tau^{(m)}|\boldsymbol{\Lambda})\right)\ p(\boldsymbol{\theta})\ p(\boldsymbol{\Lambda})$$



Stochastic Variational Inference (SVI) for learning with likelihood ℓ gradients:

$$\frac{d\ell}{d\boldsymbol{\theta}} = \underbrace{\frac{\partial \ell}{\partial \boldsymbol{\tau}}}_{\substack{\text{Adjoint} \\ \text{model}}} \underbrace{\frac{\partial \boldsymbol{\tau}}{\partial \boldsymbol{\theta}}}_{\substack{\text{NN} \\ \text{auto-diff}}} \qquad \frac{d\ell}{d\boldsymbol{E}_\tau} = \boldsymbol{W}^T \frac{d\ell}{d\boldsymbol{\tau}}$$



Results

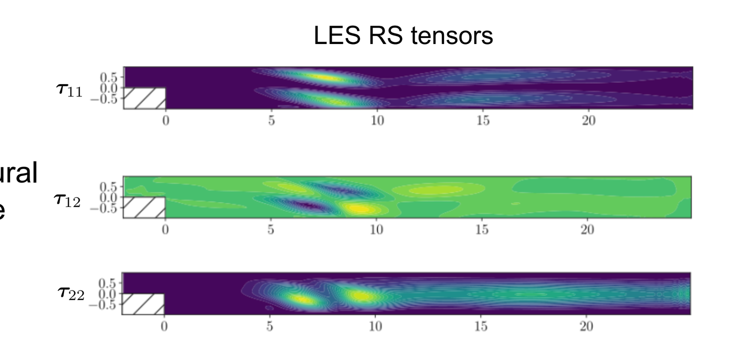
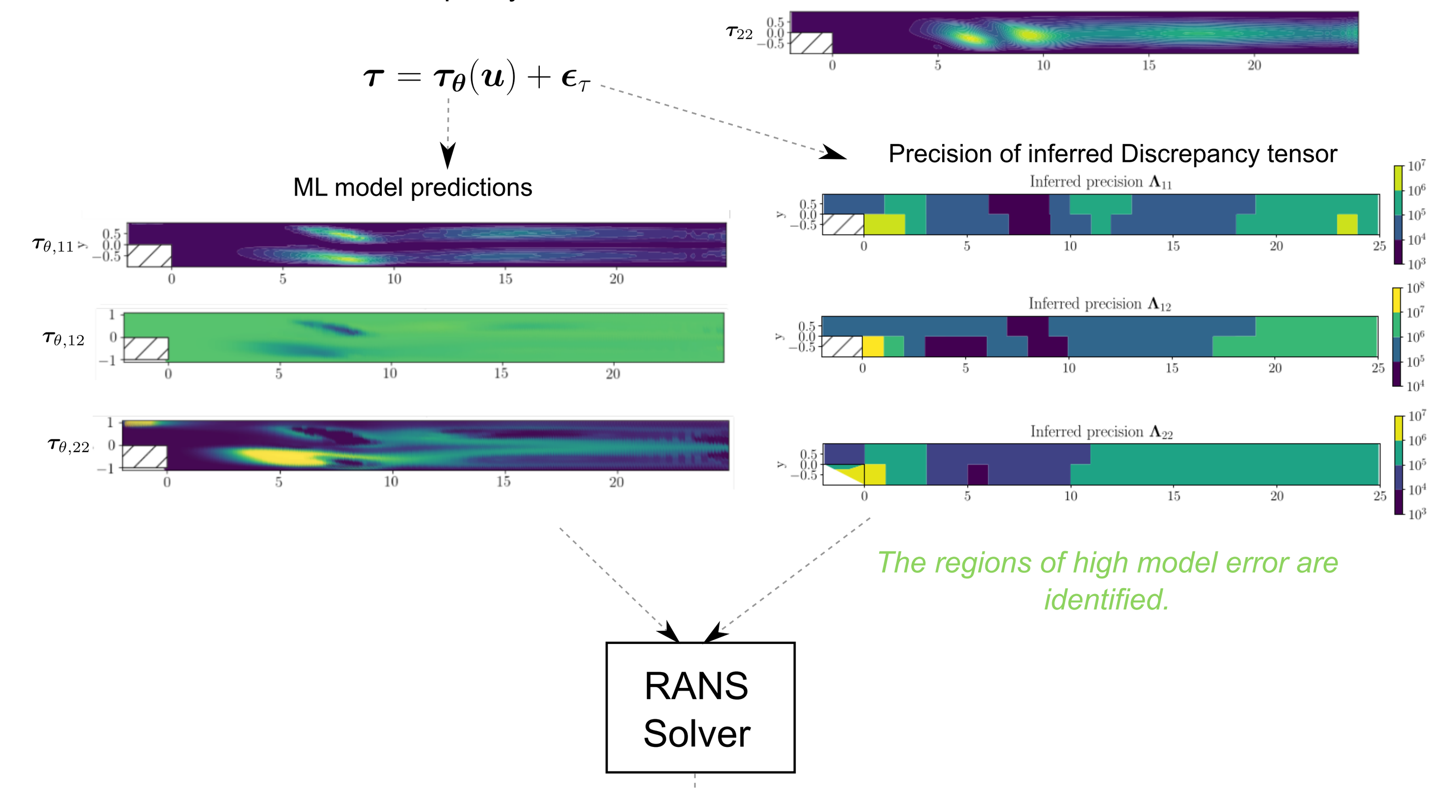


Figure 1 : Comparing LES (high-fidelity sim.), neural network predicted Reynolds Stress tensor and the statistics $\boldsymbol{\Lambda}$ of the inferred discrepancy tensor.



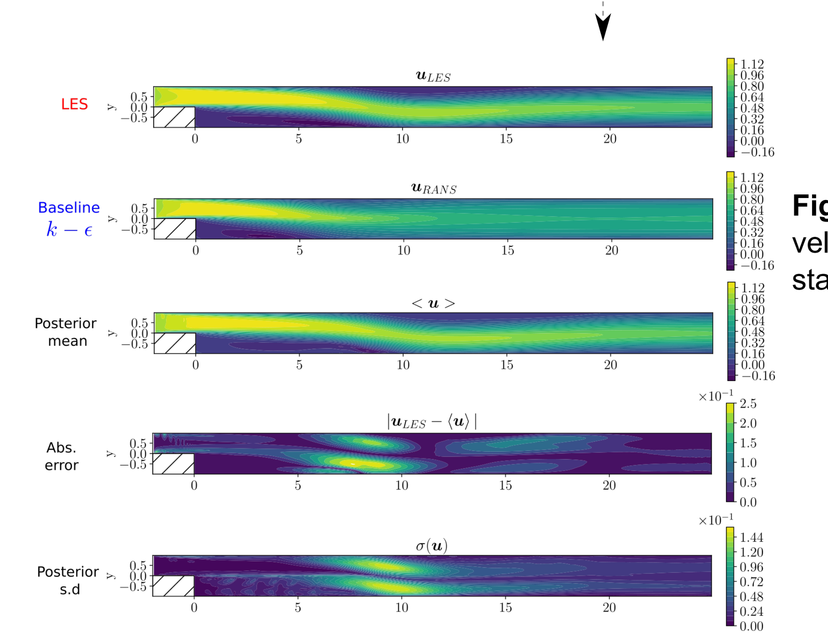


Figure 2 : Contour plots of x-direction velocity for LES, $k-\epsilon$ model (industrial standard) and our probabilistic preditions.

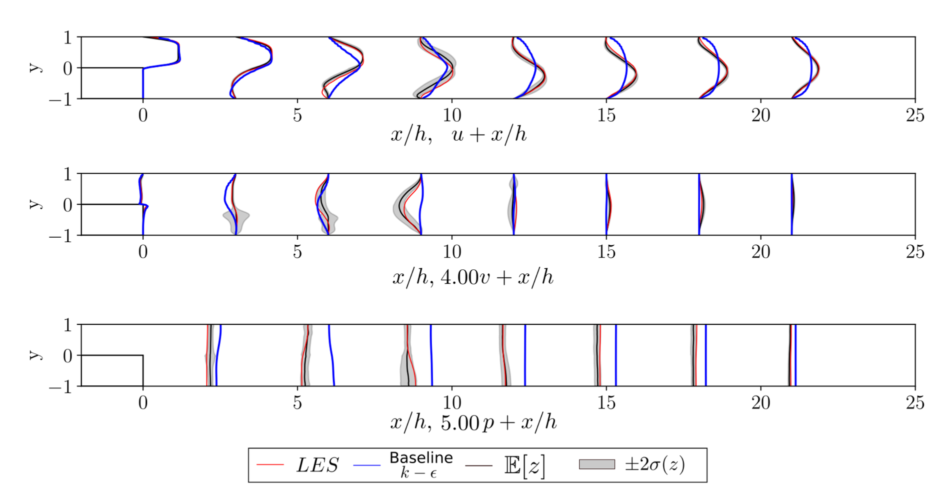


Figure 3 : Section plot compairing x-direction velocity (**u**), y-direction velocity (**v**), and pressure (**p**) for LES, $k-\epsilon$ model and our probabilistic preditions.

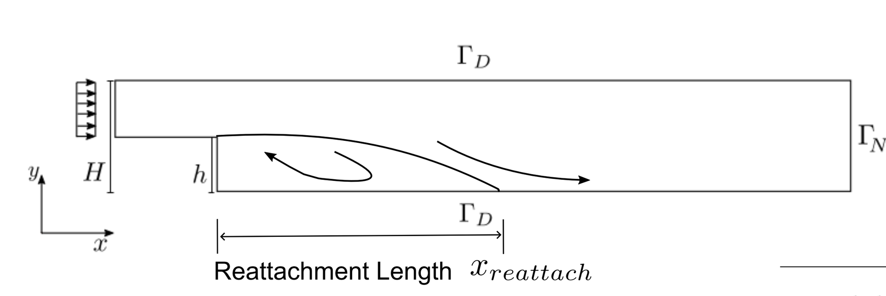


Figure 4 : Reattachment length is notoriously difficult to capture, as evident from the results of the previous works (see table). However, the proposed model is able to envelop the ground truth.

Model	$x_{reattach}$[x/h]
LES (reference)	9.10
Biswas et al.[3] ($H/h = 1.9423$)	8.9
Baseline RANS ($k-\epsilon$ model)	5.61
Geneva et al.[4]	5.52
proposed model	$\mathbf{10.06 \pm 1.21}$

**Biswas el al. is the experiemtal data*

Atul Agrawal*, Phaedon-Stelios Koutsourelakis

atul.agrawal@tum.de

Professorship of Data-Driven Materials Modelling
Technical University of Munich

Get paper and poster:

[1] Julia Ling, Andrew Kurzawski, and Jeremy Templeton. “Reynolds averaged turbulence mod-elling using deep neural networks with embedded invariance”. In: Journal of Fluid Mechanics807 (2016), pp. 155–166
[2] Atul Agrawal and Phaedon-Stelios Koutsourelakis. “A probabilistic, data-driven closure modelfor RANS simulations with aleatoric, model uncertainty”. (*Under review in Journal of Computational Physics)*
[3] Gautam Biswas, Michael Breuer, and Franz Durst. “Backward-facing step flows for variousexpansion ratios at low and moderate Reynolds numbers”. In: J. Fluids Eng. 126.3 (2004),pp. 362–374.
[4] Nicholas Geneva and Nicholas Zabaras. “Quantifying model form uncertainty in Reynolds-averaged turbulence models with Bayesian deep neural networks”. In: Journal of Computa-tional Physics 383 (2019), pp. 125–147.

Technical University of Munich