Data Driven & Physics Constrained Perturbations For Turbulence Model Uncertainty Estimation

Jan Felix Heyse, Aashwin Ananda Mishra, Gianluca Iaccarino
Mechanical Engineering Department, Stanford University, Stanford, CA-94305, USA

Abstract

Turbulence models represent the workhorse for academic and industrial studies involving real life manifestations of fluid turbulence. However, due to the simplifications inherent in their formulation, such turbulence models have a high degree of epistemic uncertainty associated with their predictions. Estimating this model form uncertainty is a critical and long standing problem in turbulence modeling and engineering design. To this end, the direct application of machine learning to estimate turbulence model uncertainties ignores physics based domain knowledge and may even lead to unphysical results. In this light, we outline a framework that utilizes data driven algorithms in conjunction with physics based constraints to generate reliable uncertainty estimates for turbulence models while ensuring that the solutions are physically permissible. The trained machine learning model, utilizing the random forest algorithm, is embedded in a Computational Fluid Dynamics solver and applied to complex problems to test and illustrate its efficacy. This library is to be released as a computational software tool that enables the inclusion of physics based constraints in applications of machine learning in turbulence modeling.

Introduction

Fluid turbulence is a central problem across a variety of disciplines in science and engineering, including Mechanical, Aerospace and Civil Engineering; Biomedical, Oceanographic, Meteorological and Astrophysical Sciences, besides others. The ability to reliably predict the evolution of turbulent flows would lead to seminal advances across these fields. However, in spite of over a century of focused research, no analytical theories to predict the evolution of turbulence have been developed. With the present state of computational resources, a purely numerical resolution of turbulent time and length scales encountered in engineering problems is not viable in industrial design practice. Consequently, almost all investigations have to resort to some degree of modeling. Turbulence models are constitutive relations attempting to relate quantities of interest to flow parameters using assumptions and simplifications derived from physical intuition and observations. Reynolds-averaged Navier-Stokes (RANS)-based models represent the pragmatic recourse for complex engineering flows, with a vast majority of simulations, in both academia and industry, resorting to this avenue. Despite their widespread use, RANS-based models suffer from an inherent structural inability to replicate fundamental turbulence processes and specific flow phenomena, as they introduce a high degree of epistemic uncertainty into the simulations arising due to the model form(Craft, Launder, and Suga 1996; Schobeiri and Abdelfattah 2013).

In this light, uncertainty quantification for RANS-based closures attempts to assess the trustworthiness of model predictions of quantities of interest and is thus of considerable utility in establishing RANS models as reliable tools for engineering applications. To this end, investigators in the recent past have utilized data driven machine learning approaches to engender interval estimates of uncertainty on turbulence model predictions. Large corpora of available data from experiments and higher fidelity simulations present an opportunity to enhance the predictive capabilities of RANS simulations. Traditionally, data has been used in the context of turbulence modeling only for model calibration and to define model corrections. Almost all turbulence models involve some empirical constants which are tuned to optimize the RANS predictions with respect to specific calibration cases (Hanjalić and Launder 1972). Over the last decade, there has been an increasing attempt to utilize data driven approaches to quantify the epistemic uncertainties in RANS models.

As illustrative instances, Wang and Dow (2010) studied the structural uncertainties of the $k-\omega$ turbulence model by modeling the eddy viscosity discrepancy (i.e. the difference between reference high-fidelity data and RANS predictions) as a random field. Their approach is based on Monte Carlo sampling, but given its slow convergence, a considerable number of simulations are required in order to obtain meaningful uncertainty estimates. Wu, Xiao, and Paterson (2018) used data driven algorithms to predict Reynolds stress discrepancies. The target of the machine learning model was a post-hoc, local correction term for the RANS model's predictions. Duraisamy, Iaccarino, and Xiao (2019) provide a more comprehensive review of how data has been used to enhance turbulent flow simulations.

However, such direct application of machine learning models to problems in physical sciences, such as fluid flow

and turbulence modeling, may not completely account for the domain knowledge and more importantly, all the essential physics based constraints required. As an illustration, the Reynolds averaging carried out in turbulence modeling introduces a term in the momentum equations that requires further modeling assumptions or simplifications, i.e. it is *unclosed*. This term is referred to as the Reynolds stress tensor, $R_{ij} = \langle u_i u_j \rangle$, where u_i are components of the fluctuating velocity field after filtering. This is the key Quantity of Interest in turbulence modeling. However, there are essential physics based constraints that any prediction of the Reynolds stresses must follow. These are referred to as realizability constraints. Schumann (1977) was the first to articulate the realizability constraint in the context of turbulence closures, requiring models to yield a Reynolds stress tensor that ensure that $R_{\alpha\alpha} \geq 0$, $R^2_{\alpha\beta} \geq R_{\alpha\alpha} R_{\beta\beta}$ and $det(R) \geq 0$. Unless these constraints are explicitly adhered to, model predictions are unphysical. Furthermore, using machine learning approaches without physics based constraining, leads to issues when such data driven models are integrated in Computational Fluid Dynamics (CFD) software. Unrealizable models can lead to problems in numerical convergence and even numerical instability. Straightforward application of machine learning models to problems in turbulence modeling has led to unrealizable predictions and convergence issues when such data driven models are integrated in CFD software.

In this investigation, we outline a methodology that introduces physics constrained perturbations to estimate structural uncertainty in turbulence models. Thence, we utilize machine learning algorithms to infer these perturbations from labeled data. These two steps together ensure that this framework is both physics constrained and data driven. Finally, we integrate this library into CFD software suites and carry out tests for robustness and reliability.

After an overview of the problem in Section I, we outline the test problem in Section II. Thence, we outline the physics constrained perturbation framework which is applied without and with inference from data (data free and data driven). In the data free results, we utilize the maximum physically permissible perturbations. In the data driven framework, we train a random forest regressor to predict the perturbations using data from other flows and integrate this trained model in the CFD software suite. We conclude with a summary of this work and directions for future research.

Baseline Simulation: Turbulent Diffuser

The test case in this work is the turbulent separated flow in a diffuser. Diffusers are used to decelerate the flow and increase the static pressure of the fluid. The operating principle is simply a change in cross-sectional area, but space constraints and reduction of losses often lead to configurations that are prone to flow separation. Prediction of the turbulent flow in a diffuser represents the challenge in this work.

The turbulent flow in planar asymmetric diffuser first described by Obi, Aoki, and Masuda (1993) is considered. Figure 1 shows the setup: A channel is expanded from inflow width H to outflow width $4.7H$. In the expansion section,

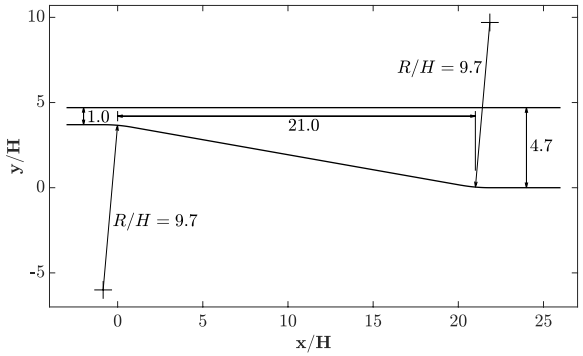

Figure 1: Asymmetric plane diffuser setup. Inflow from the left, outflow to the right.

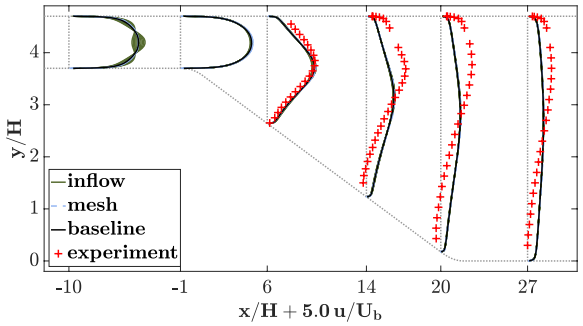

Figure 2: Diffuser simulation. Inflow (green) and mesh (blue) sensitivities, baseline calculation (black). Profiles of streamwise velocity at different x locations with experimental data (red).

the bottom wall is opening up at a 10° angle. The corners at the beginning and the end of that slope are rounded with a radius of $9.7H$.

The inflow is fully turbulent and has the Reynolds number $20,000$, based on the centerline velocity and inflow channel height H. Interesting features of the resulting flow are flow separation, flow reattachment, and development of a new boundary layer. RANS simulations were carried out in OpenFOAM using the $k-\epsilon$ turbulence model. Fully turbulent channel flow was used as inflow condition at $x/H = -10$. The outlet was at $x/H = 60$. The baseline calculation had a structured mesh with $9{,}472$ cells, 148 in the x and 64 in the y direction.

Both a mesh convergence study and an inflow sensitivity study were performed. For mesh convergence, each two coarser and two finer grids were used, with the number of grid cells changing by a factor of 2 between levels. For inflow sensitivity, the inlet velocity profile was distorted at constant flow rate to vary the centerline velocity between 90% and 110% of its nominal value. The results can be seen in figure 2, where the streamwise velocity component is plotted at different x locations across the channel height. All individual mesh and inflow solutions are plotted, giving impression of the respective sensitivities, as well as the base-

line solution and experimental data from Buice and Eaton (2000). Very limited sensitivity to boundary conditions and numerical errors is observed in and after the expanding section, providing confidence in the computations. Yet, while the flow remains attached at all times in the RANS simulations, the experimental data reveals the existence of a large flow separation that has not been captured by the simulation. The simulations are overpredicting the streamwise velocity in the lower half of the channel and underpredicting it in the upper half.

Data free Uncertainty Quantification

Herein, we outline the physics constrained perturbation framework that ensures Reynolds stress realizability. Given an initially realizable Reynolds stress tensor, this framework ensures that the perturbed Reynolds stresses remain positive semi-definite. The Reynolds stress tensor can be decomposed into the anisotropic and deviatoric components as

$$R_{ij} = 2k(b_{ij} + \frac{\delta_{ij}}{3}). \tag{1}$$

Here, $k(=\frac{R_{ii}}{2})$ is the turbulent kinetic energy and $b_{ij}(=\frac{R_{ij}}{2k} - \frac{\delta_{ij}}{3})$ is the Reynolds stress anisotropy tensor. The Reynolds stress anisotropy tensor can be expressed as $b_{in}v_{nl} = v_{in}\Lambda_{nl}$, where v_{nl} is the matrix of orthonormal eigenvectors and Λ_{nl} is the traceless diagonal matrix of eigenvalues λ_k. Multiplication by v_{jl} yields $b_{ij} = v_{in}\Lambda_{nl}v_{jl}$. This is substituted into Equation (1) to yield

$$R_{ij} = 2k(v_{in}\Lambda_{nl}v_{jl} + \frac{\delta_{ij}}{3}). \tag{2}$$

The tensors v and Λ are ordered such that $\lambda_1 \geq \lambda_2 \geq \lambda_3$. In this representation, the shape, the orientation and the amplitude of the Reynolds stress ellipsoid are directly represented by the turbulence anisotropy eigenvalues λ_l, eigenvectors v_{ij} and the turbulent kinetic energy k, respectively.

To account for the errors due to closure assumptions, the tensor perturbation approach introduces perturbations into the modeled Reynolds stress during the CFD solution iterations. This perturbed form is expressed as:

$$R^*_{ij} = 2k^*(\frac{\delta_{ij}}{3} + v^*_{in}\Lambda^*_{nl}v^*_{lj}) \tag{3}$$

where * represents the perturbed quantities. Thus, $k^* = k + \Delta k$ is the perturbed turbulent kinetic energy, v^*_{in} is the perturbed eigenvector matrix, and, Λ^*_{nl} is the diagonal matrix of perturbed eigenvalues, λ^*_l.

In this context, the eigenvalue perturbation can be expressed as a sum of perturbations towards the 3 corners of the barycentric map. The corners of that triangle correspond to limiting states of turbulence with 1, 2, and 3 components, respectively. The expression for the Reynolds stresses with only eigenvalue perturbations is given by $R^*_{ij} = 2k(\frac{\delta_{ij}}{3} + v_{in}\Lambda^*_{nl}v_{lj})$, where Λ^*_{nl} represents the diagonal matrix of perturbed eigenvalues. The perturbed eigenvalues can be expressed by the mapping $\lambda^*_l = B^{-1}\mathbf{x}^*$. Here, $\mathbf{x}^* = \mathbf{x} + \Delta_B(\mathbf{x^t} - \mathbf{x})$ is the representation of the perturbation in the barycentric triangle with $\mathbf{x}$ being the unperturbed state in the barycentric map, $\mathbf{x}^*$ representing the perturbed position, $\mathbf{x^t}$ representing the state perturbed toward and Δ_B is the magnitude of the perturbation. In this context, $\lambda^*_l = B^{-1}\mathbf{x}^*$ can be simplified to $\lambda^*_l = (1-\Delta_B)\lambda_l + \Delta_B B^{-1}\mathbf{x^t}$. Here, B defines a linear map between the perturbation in the barycentric triangle and the eigenvalue perturbations. With the three vertices x_{1C} , x_{2C} , and x_{3C} as the target states, we have $B^{-1}x_{1C} = (2/3, -1/3, -1/3)^T$, $B^{-1}x_{2C} = (1/6, 1/6, -1/3)^T$, and $B^{-1}x_{3C} = (0,0,0)^T$. Figure 3 shows the triangle in the barycentric map as well as one realizable location $\mathbf{x}$ coming from a RANS turbulence model. The eigenvalue perturbations add three perturbed simulations to the baseline calculation, one for each limiting state, leading to a total of four calculations. The uncertainty estimates are constructed by computing the range of values across the four calculations. The minimum and maximum values of the range form envelopes for any quantity of interest.

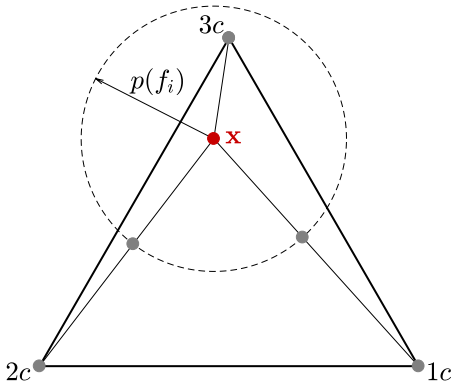

Figure 3: Barycentric domain and eigenvalue perturbation.

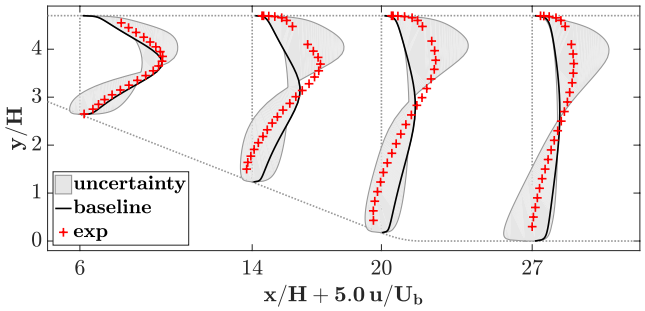

Figure 4: Data free, uniform eigenvalue perturbation. Profiles of streamwise velocity at different x locations.

The framework of eigenvalue perturbations is applied to the present test case. Figure 4 shows the resulting uncertainty envelopes, which cover the experimental results in most locations. Unlike the mesh study and the inflow sensitivity study from the previous section, this analysis correctly indicates that there might be a region of flow recirculation at the bottom wall. The uncertainty estimates, however, go beyond the experimental data, in some regions substantially, in other words seem to overestimate the modeling errors

in some locations. This is expected because the perturbations are targeting all possible extreme states of turbulence anisotropy without consideration of their plausibility. The data free framework perturbs the Reynolds stresses everywhere in the domain all the way to the respective limiting state. Yet, the Reynolds stress predictions of the turbulence model do not have the same level of inaccuracy throughout the domain.

Data driven Uncertainty Quantification

We study a data driven approach to predict a local eigenvalue perturbation strength based on mean flow features. Here, we define the local perturbation strength p as the distance in barycentric coordinates between the unperturbed and the perturbed projection of the Reynolds stress. p is predicted by a machine learning model using physically relevant flow features f_i as input. Figure 3 illustrates the meaning of p in the barycentric map. The original location $\vec{x}_{LF}$ is perturbed towards the same extreme states as in the data free approach, but now the perturbed locations, marked by grey dots, are not more than p away from the original position. In the example from the illustration, that means the 3-component limiting state is reached, while the perturbations towards the 1- and 2-component limits are smaller. The perturbation strength p is directly related to the perturbation magnitude as: $\Delta_B = \min(p/d_t, 1)$, where d_t is the distance in the barycentric map between the unperturbed state and the respective corner towards which it is perturbed. The perturbed locations are still always within the triangle and therefore within the constraints of realizability. This definition of the perturbation strength means that the effective perturbations cannot be greater than for the data free case, but they can be smaller.

A random regression forest is chosen as the machine learning regression model. Random forests are a supervised learning algorithm. They are ensemble learners, meaning that they leverage a number of decorrelated simpler models to make a prediction. In this case of a random forest, the simpler models are regression trees (Breiman et al. 1984). Regression trees are able to learn non-linear functions. They are also robust to extrapolation, since they cannot produce predictions outside the range of the training data labels, and to uninformative features (Ling and Templeton 2015; Milani et al. 2017).

In machine learning models, the mean squared error can be decomposed into the squared bias of the estimate, the variance of the estimate, and the irreducible error:

$$MSE(x) = \underbrace{(E[\hat{f}(x)] - f(x))^2}_{\text{squared bias}} + \underbrace{E[(\hat{f}(x) - E[\hat{f}(x)])^2]}_{\text{variance}} + \underbrace{\sigma_e^2}_{\text{irred. err.}},$$

where $\hat{f}(x)$ is the model prediction and $f(x)$ is the true label. As the name suggests, the irreducible error stems from noise in the data and cannot be reduced through the model. Bias is introduced through assumptions that are made in the model before the training. The more flexible a model is, the lower is its bias. Variance is related to generalization: It measures how much the model predictions would change if trained on different data. High variance indicates strong overfitting and poor generalization.

#1	$\frac{\text{tr}(\mathbf{S})}{\lvert\text{tr}(\mathbf{S})\rvert+\tau_t^{-1}}$	#7	$\frac{P}{\lvert P\rvert+\epsilon}$
#2	$\frac{\text{tr}(\mathbf{S}^2)}{\lvert\text{tr}(\mathbf{S}^2)\rvert+\tau_t^{-2}}$	#8	$\frac{k/\epsilon}{\lvert k/\epsilon\rvert+S^{-1}}$
#3	$\frac{\text{tr}(\mathbf{S}^3)}{\lvert\text{tr}(\mathbf{S}^3)\rvert+\tau_t^{-3}}$	#9	u/c_0
#4	$\frac{\text{tr}(\mathbf{R}^2)}{\lvert\text{tr}(\mathbf{R}^2)\rvert+\tau_t^{-2}}$	#10	$\sqrt{k}/u$
#5	$\frac{\text{tr}(\mathbf{R}^2\mathbf{S}^2)}{\lvert\text{tr}(\mathbf{R}^2\mathbf{S}^2)\rvert+\tau_t^{-4}}$	#11	$\min(\sqrt{k}d_w/50\nu, 4)$
#6	$\frac{W^2-S^2}{W^2+S^2}$	#12	$\lvert g_j s_j\rvert/g*\sqrt{k}/u$

Table 1: Non-dimensional features used for the random regression forest. The following variables are used: the mean rate of strain and rotation $S_{ij} = \frac{1}{2}(\nabla u_{ij} + \nabla u_{ji})$, $W_{ij} = \frac{1}{2}(\nabla u_{ij} - \nabla u_{ji})$; the turbulence time scale $\tau_t = \frac{k}{\epsilon}$; the unit vector along the streamline $s_i = u_i/u$; and the gradient of the streamline aligned velocity $g_i = s_j \frac{\partial u_j}{\partial x_i}$.

In many machine learning models one can vary the model flexibility. A more flexible model is able to learn more complex relationships and will therefore reduce the bias of the predictions. At the same time, a more flexible model increases the likelihood of overfitting to the training data and thereby of increasing the variance. The search for the optimum model complexity to achieve both low bias and low variance is commonly referred to as the bias-variance tradeoff.

Binary decision trees are very flexible and tend to overfit strongly to the training data. Hence, they have a low bias and a high variance. Random forests base their predictions on a number of decorrelated decision trees. Decorrelation is achieved by bagging, which is the training on random subsets of the training data, as well as randomly sampling the active variables at each split. Since the trees are decorrelated, the variance of the random forest predictions is reduced and generalization improved. At the same time, random forests are able to keep the low bias of the decision trees. This makes random forests, despite their simplicity, powerful predictors for a range of applications (Breiman 2001). The random forest is implemented using the OpenCV library.

In the present scenarion, i.e. an incompressible turbulent flow, a set of twelve features was chosen. In order to be able to generalize to cases other than the training data set, all features are non-dimensional. The first eight features were non-dimensionalized such that they lie within the interval $[-1, 1]$. The other ones were non-dimensional quantities common in fluid mechanics, as well as a marker function indicating regions in which the turbulence model is expected to be inaccurate (Gorlé et al. 2014). The computation of the features requires knowledge of the following variables, which are either constant or solved for during the RANS calculations: the mean velocity and its gradient, the turbulent kinetic energy as well as its production and its dissipation rates, the minimum wall distance, the molecular viscosity, and the speed of sound. A list of features is given in table 1.

The periodic wavy wall case was used to obtain training data for the random regression forest model. It is defined as a turbulent channel with a flat top wall and a sinu-

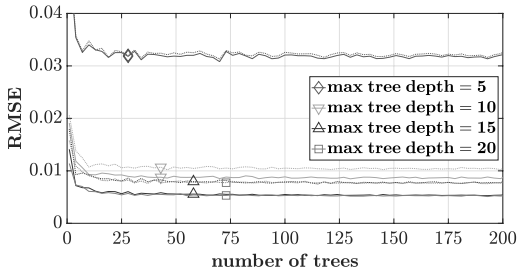

Figure 5: Training error (solid) and validation error (dotted) vs. number of trees for different maximum tree depth values.

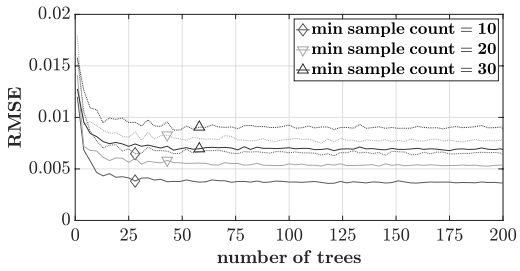

Figure 6: Training error (solid) and validation error (dotted) vs. number of trees for different minimum sample counts.

soidal bottom. The ratio of channel height to wave length is $H/\lambda = 1.0$, and the ratio of wave height to wave length is $2A/\lambda = 0.1$. The wave is repeating periodically, and on its downward slope a flow separation occurs. An unperturbed baseline calculation was run using the $k-\epsilon$ turbulence model with periodic inflow and outflow boundary conditions. A mesh convergence study, as done for the diffuser, suggested only limited numerical errors. Features for the random forest training were computed from the baseline case, and labels were computed from both the baseline case and higher fidelity data. The labels are defined as the actual distances in the barycentric domain between the location predicted by the baseline calculation and the higher fidelity one. The higher fidelity data is DNS data from Rossi (2006).

Hyperparameters are parameters in machine learning models that are not learned during model training, but that are instead set before training and used to define the functional form of the model and control the learning process. The impact of four different hyperparameters on the learning of the random regression forest model is studied: the maximum tree depth, the minimum sample count, the active variable count, and the number of trees. The minimum sample count is the minimum number of samples required at a particular tree node in order to do further splitting. The active variable count is the number of features randomly chose at each node to find the optimal split.

For each of the first three hyperparameters a couple of different values were tested over a range of 1 to 200 regression trees. To improve readability, the figures 5 to 7 show the results for every third number of trees only. The dataset was split into 80% training set and 20% validation set for this

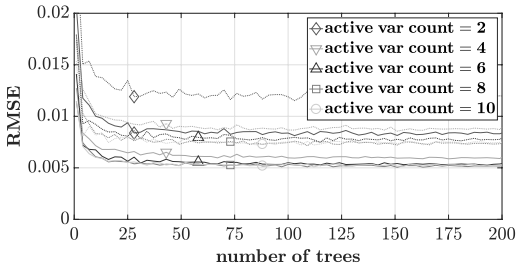

Figure 7: Training error (solid) and validation error (dotted) vs. number of trees for different active variable counts.

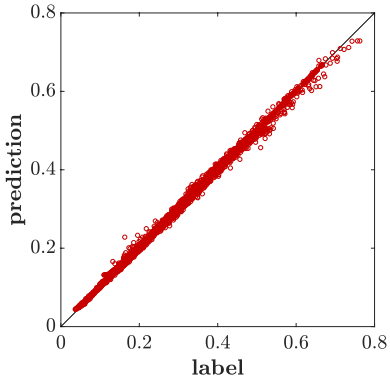

Figure 8: Predictions vs. labels on wavy wall training data.

part. Only after making a choice on the hyperparameters, a final random forest was trained on the full dataset to achieve best performance when employed at the test case.

The results from the maximum tree depth study are shown in figure 5. The training and test errors are plotted in solid and dotted lines, respectively, against the number of trees. The tested values are 5, 10, 15, and 20, and best performance was achieved for 15 and 20. 15 was chosen, because a smaller value means smaller computational costs. Figure 6 shows the results from the minimum sample count study. The tested values are 10, 20, and 30. While the larger values did slightly better in terms of generalization, 10 overall showed the smallest error and was chosen as minimum sample count. The number of active variables was varied between 2 and 10 at an increment of 2 as shown in figure 7. Larger numbers of active variables lead to lower training and test errors, with 8 and 10 yielding the smallest errors.

Only after making a choice on the hyperparameters, a random forest was trained on the full wavy wall dataset to achieve best performance when employed at the diffuser case. Figure 8 shows a scatter plot of predictions vs. labels for the training data. There is a good agreement between the predicted and the true perturbation strengths.

The OpenCV library used to train the random regression forest allows for the computation of feature importance, i.e. a quantitative assessment of the impact that each feature has on the final prediction.

The maximal information coefficient (MIC) is a measure of dependence between variables. It is able to detect both lin-

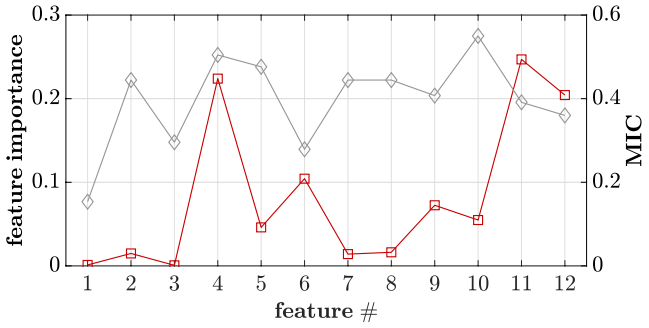

Figure 9: Normalized random forest feature importance scores (squares) and MIC scores (diamonds) for the features.

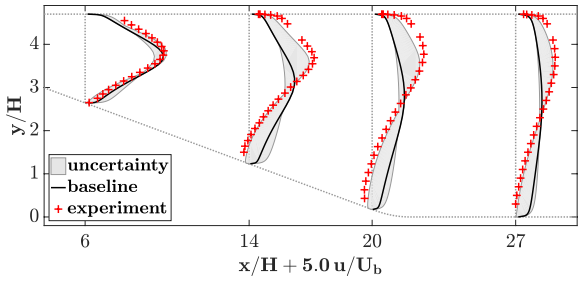

Figure 10: Data driven, local eigenvalue perturbation. Profiles of streamwise velocity at different x locations.

ear and more complex relationships, and it has shown good equitability (Reshef et al. 2016). We can estimate the MIC between the features and the labels and compare the scores to the feature importance scores from the random forest. The MIC was estimated using the tools provided by Albanese et al. (2018). Figure 9 presents the normalized feature importance scores from the random forest as squares and the estimated MIC scores as diamonds.

Two of the most important features of the random forest are #11 the non-dimensional wall distance and #12 the marker indicating deviation from parallel shear flow (Gorlé et al. 2012). The challenging flow features such as flow separation happen at or near the wall, supporting the significance of the wall distance. The marker function was developed specifically to identify regions where the linear eddy viscosity assumption becomes invalid. The importance of this feature indicates that the model was able to recognize this relationship. The features based on combinations of the mean rate of rotation were clearly more important than the ones base on combinations of the mean rate of strain, with #4 the trace of the squared mean rotation rate tensor being ranked as second most important feature. Another important feature is #6 the Q criterion, identifying vortex regions. As expected #1 the divergence of the velocity is not a significant feature for this incompressible flow case. It is important to point out that the feature importance is strongly related to the baseline turbulence model.

The feature importance scores show some trends that are also captured by the MIC, e.g. the ranking between the first five features. This increases our confidence in the learning of the machine learning model. At the same time we notice that there is no perfect agreement. For example, #10 the turbulence intensity was ranked first by the MIC while being not as important to the random forest predictions as other features as. This leaves room for more detailed investigations.

Finally, this new, data driven framework was applied to the planar asymmetric diffuser. The random forest model was used to predict a local perturbation strength at every cell during the RANS calculations. The data driven eigenvalue perturbations lead to an increase in costs for the RANS simulations. Compared to the calculations with the baseline $k-\epsilon$ turbulence model, the observed runtime increases at a factor of $2-3$. There are two reasons for this increase. First, the calculations associated with the Reynolds stress perturbations took some time; and second, the perturbations had an effect of the convergence of the solver, resulting in more iterations that had to be completed depending on the particular limiting state. The time of computing the Reynolds stress perturbations was dominated by the evaluation time of the random forest, which scales linearly with the number of trees, a number that potentially could be reduced.

As for the data free uncertainty envelopes, three perturbed calculations were carried out for the three limiting states of turbulence. Figure 10 shows the results. The uncertainty envelopes still display the same general trend, suggesting an overprediction of the streamwise velocity in the lower half of the channel. As expected from permitting smaller perturbation strengths, the envelopes are narrower than they are when using the data free framework from the previous section. There are no regions where the uncertainty is substantially overestimated: In most regions, the envelopes reach just up to or at least very near to the experimental data. Thus, for the test case the data driven uncertainty estimates give a reasonable estimate of the modeling errors and therefore the true uncertainty in the flow predictions.

Conclusion and Future Work

In this investigation, we outline a physics constrained data driven framework for uncertainty quantification of turbulence models. We outline a methodology that introduces physics constrained perturbations to estimate structural uncertainty in turbulence models while retaining the realizability constraints on the Reynolds stresses. Thence, we utilize machine learning algorithms to infer these perturbations from labeled data. These two steps together ensure that this framework is both physics constrained and data driven. Finally, we integrate this library into CFD software and carry out tests for robustness and reliability. At present, we are testing this framework using different baseline flows and different machine learning algorithms. The software implementation of this physics constrained machine learning library for turbulence model uncertainty quantification will be released soon.

References

Albanese, D.; Riccadonna, S.; Donati, C.; and Franceschi, P. 2018. A practical tool for maximal information coefficient

analysis. *GigaScience* 7(4). ISSN 2047-217X. doi:10.1093/gigascience/giy032. Giy032.

Breiman, L. 2001. Random Forests. *Mach. Learn.* 45(1): 5–32. ISSN 0885-6125. doi:10.1023/A:1010933404324.

Breiman, L.; Friedman, J.; Stone, C. J.; and Olshen, R. A. 1984. *Classification and Regression Trees*. Taylor & Francis. ISBN 9780412048418.

Buice, C. U.; and Eaton, J. K. 2000. Experimental Investigation of Flow Through an Asymmetric Plane Diffuser: (Data Bank Contribution)1. *Journal of Fluids Engineering* 122(2): 433–435. ISSN 0098-2202. doi:10.1115/1.483278.

Craft, T. J.; Launder, B. E.; and Suga, K. 1996. Development and application of a cubic eddy-viscosity model of turbulence. *International Journal of Heat and Fluid Flow* 17(2): 108 – 115. ISSN 0142-727X. doi:https://doi.org/10.1016/0142-727X(95)00079-6.

Duraisamy, K.; Iaccarino, G.; and Xiao, H. 2019. Turbulence Modeling in the Age of Data. *Annual Review of Fluid Mechanics* 51(1): 357–377. doi:10.1146/annurev-fluid-010518-040547. URL https://doi.org/10.1146/annurev-fluid-010518-040547.

Gorlé, C.; Emory, M.; Larsson, J.; and Iaccarino, G. 2012. Epistemic uncertainty quantification for RANS modeling of the flow over a wavy wall. *Center for Turbulence Research Annual Research Briefs* .

Gorlé, C.; Larsson, J.; Emory, M.; and Iaccarino, G. 2014. The deviation from parallel shear flow as an indicator of linear eddy-viscosity model inaccuracy. *Physics of Fluids* 26(5): 051702. doi:10.1063/1.4876577.

Hanjalić, K.; and Launder, B. E. 1972. A Reynolds stress model of turbulence and its application to thin shear flows. *Journal of Fluid Mechanics* 52(4): 609–638. doi:10.1017/S002211207200268X.

Ling, J.; and Templeton, J. 2015. Evaluation of machine learning algorithms for prediction of regions of high Reynolds averaged Navier Stokes uncertainty. *Physics of Fluids* 27(8): 085103. doi:10.1063/1.4927765.

Milani, P. M.; Ling, J.; Saez-Mischlich, G.; Bodart, J.; and Eaton, J. K. 2017. A Machine Learning Approach for Determining the Turbulent Diffusivity in Film Cooling Flows. *Journal of Turbomachinery* 140(2). ISSN 0889-504X. doi:10.1115/1.4038275. 021006.

Obi, S.; Aoki, K.; and Masuda, S. 1993. Experimental and Computational Study of Turbulent Separating Flow in an Asymmetric Plane Diffuser. In *9th International Symposium on Turbulent Shear Flows*, 305. Kyoto, Japan.

Reshef, Y. A.; Reshef, D. N.; Finucane, H. K.; Sabeti, P. C.; and Mitzenmacher, M. 2016. Measuring Dependence Powerfully and Equitably. *Journal of Machine Learning Research* 17(211): 1–63. URL http://jmlr.org/papers/v17/15-308.html.

Rossi, R. 2006. *Passive scalar transport in turbulent flows over a wavy wall*. Ph.D. thesis, Università degli Studi di Bologna, Bologna, Italy.

Schobeiri, M. T.; and Abdelfattah, S. 2013. On the Reliability of RANS and URANS Numerical Results for High-Pressure Turbine Simulations: A Benchmark Experimental and Numerical Study on Performance and Interstage Flow Behavior of High-Pressure Turbines at Design and Off-Design Conditions Using Two Different Turbine Designs. *Journal of Turbomachinery* 135(6). ISSN 0889-504X. doi:10.1115/1.4024787. 061012.

Schumann, U. 1977. Realizability of Reynolds-stress turbulence models. *The Physics of Fluids* 20(5): 721–725.

Wang, Q.; and Dow, E. A. 2010. Quantification of structural uncertainties in the k-omega turbulence model. *Center for Turbulence Research Proceedings of the Summer Program* .

Wu, J.-L.; Xiao, H.; and Paterson, E. 2018. Physics-informed machine learning approach for augmenting turbulence models: A comprehensive framework. *Physical Review Fluids* 3(7). ISSN 2469-990X. doi:10.1103/physrevfluids.3.074602.