GPPS | JOURNAL

JOURNAL OF THE GLOBAL POWER AND PROPULSION SOCIETY
journal.gpps.global

Estimating RANS model uncertainty using machine learning

Original article

Article history:
Submission date: 12 November 2020
Final revision date: 6 February 2021
Acceptance date: 18 March 2021
Publication date: 25 May 2021

Check for updates

Jan F. Heyse[1,*], Aashwin A. Mishra[1], Gianluca Iaccarino[1]

[1]*Mechanical Engineering Department, Stanford University, Stanford, CA, USA*

*Correspondence:
JFH: heyse@stanford.edu

Peer review:
Single blind

Keywords:
machine learning; uncertainty quantification; Reynolds-averaged Navier-Stokes; UQ; data-driven

Citation:
Heyse J. F., Mishra A. A., and Iaccarino G. (2021). Estimating RANS model uncertainty using machine learning. Journal of the Global Power and Propulsion Society. Special Issue: Data-Driven Modelling and High-Fidelity Simulations: 1–14.
https://doi.org/10.33737/jgpps/134643

OPEN ACCESS

Abstract

In this work we outline a machine learning strategy developed to estimate the uncertainty introduced by a turbulence model for the prediction of a turbulent separated flows. The approach is based on the introduction of eigenvalue perturbations of the Reynolds stress anisotropy; the amount of perturbation is predicted by a random forest algorithm trained on high-fidelity simulations of the flow over a wavy wall. The proposed method is applied to the flow in an asymmetric diffuser and demonstrates how the approach correctly identifies the regions in which modeling errors occur and accurately quantifies the amount of errors when compared to experimental observations.

Introduction

Reynolds-averaged Navier-Stokes (RANS) simulations are the standard for complex flow simulations in industry, because the higher fidelity direct numerical simulations (DNS) and large eddy simulations (LES) are computationally too expensive for most applications. The RANS averaging 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 and, over the past several decades, a large body of research has been devoted to its representation, namely the introduction of a turbulence model. Linear eddy viscosity models are a popular group of turbulence models that relate the strain rate tensor to the Reynolds stress anisotropy tensor through an isotropic eddy viscosity. Yet, because of their inherent assumptions they do not perform well on many realistic flows (Craft et al., 1996; Schobeiri and Abdelfattah, 2013). Particularly flows involving strong rotation or adverse pressure gradients, as encountered often in turbomachinery applications, represent a great challenge to turbulence models. It is indispensable for meaningful RANS predictions to have an estimate of the uncertainty in the simulations associated with the turbulence model. This type of uncertainty is directly related to the assumptions made in the model formulation, and therefore referred to as *model-form uncertainty*.

Data from experiments or from higher fidelity simulations presents 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. These calibration cases often are canonical flows such as decaying grid turbulence or plane homogeneous shear flow (Hanjalić and Launder, 1972). The quality of the resulting predictions depends on how relevant the calibration cases are to the case of interest. To extend the applicability of a turbulence models, corrections and modifications can be introduced, which also are calibrated on suitable data (Pope, 1978; Sarkar, 1992). Over the last decade, the main advancements of turbulence models, however, were made by capturing more physics, not by expanding the usage of data.

Recently, with the increased popularity of data science tools, efforts have been devoted to train machine learning models for RANS simulations. Some have incorporated sparse data into their simulations to improve predictions. There also have been efforts to use data to predict the Reynolds stresses, or to correct Reynolds stress predictions, as well as to identify and quantify uncertainty associated with the Reynolds stress predictions. Singh et al. (2017) used inverse modeling and trained neural networks to estimate a correction factor to the production term in the eddy viscosity transport equation of the Spalart-Allmaras turbulence model. They did not do formally quantify the uncertainty, but they studied the sensitivity of their predictions with respect to the training data set. 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 approaches 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 et al. (2018) used random forests to predict Reynolds stress discrepancies. The predictions were used to correct the turbulence model predictions and enhance the RANS results, without estimation of the structural uncertainty. Geneva and Zabaras (2019) trained a deep Bayesian neural network to predict directly the Reynolds stress tensor. Their framework incorporates uncertainty estimation for the trained black-box model, but does not extend to other turbulence models, and furthermore uncertainty was estimated was using Monte Carlo sampling. Ling and Templeton (2015) trained support vector machines, Adaboost decision trees, and random forests to identify regions of uncertainty in the flow, without making quantitative estimates of the uncertainty in the RANS predictions.

In the present work, we are seeking an efficient data driven approach to quantifying uncertainty for general RANS models. As mentioned earlier, prior works in the literature either do not provide uncertainty estimates, they do so for specific models only, or they rely on expensive Monte Carlo sampling to characterize the uncertainty.

We are therefore introducing a data-driven framework based on an extension of the eigenvalue perturbations to the Reynolds stress anisotropy tensor developed by Emory et al. (2013). In the original approach, namely the *data-free variant*, the eigenvalues are uniformly perturbed to limiting states corresponding to 1, 2, and 3 component turbulence, leading to three independent simulations, in addition to the (unperturbed) baseline. However, perturbing everywhere in the domain leads to overly conservative uncertainty estimates because the model is assumed to introduce potential errors everywhere. The new, data-driven framework is used to predict a local perturbation strength in every spatial location based on the local mean flow quantities. The predictions carried out using the machine learning perturbations demonstrate that the uncertainty estimates provide credible quantification of the *true* modeling errors, and are more effective than the original data-free uncertainty estimates. We also carry out careful sensitivity analysis to grid resolution and inflow perturbations to confirm that the turbulence closure is the main source of uncertainty in the present test cases.

Turbulent separated flows in a diffuser

In turbomachinery applications, diffusers are used to decelerate the flow and increase the static pressure of the fluid. They are positioned downstream of compressors and turbines to increase their total-to-static isentropic efficiency. 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 and sensitivity to distortions of the incoming flow stream. Predictions of the turbulent flow in a diffuser represents the challenge in this work.

Test case setup

The turbulent flow in planar asymmetric diffuser first described by Obi et al. (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, 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 at $Re = 17{,}800$ based on bulk velocity and inflow channel height H. The resulting flow has several interesting features. First, there is a flow separation on the downward slope. After this expanding section, the flow reattaches to the bottom wall. Finally, a new boundary layer develops downstream of

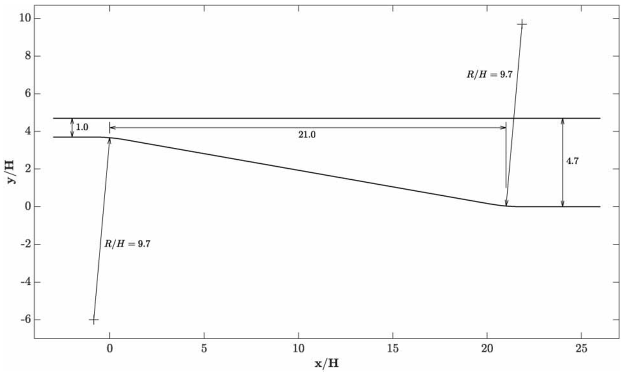

Figure 1. Asymmetric plane diffuser setup.

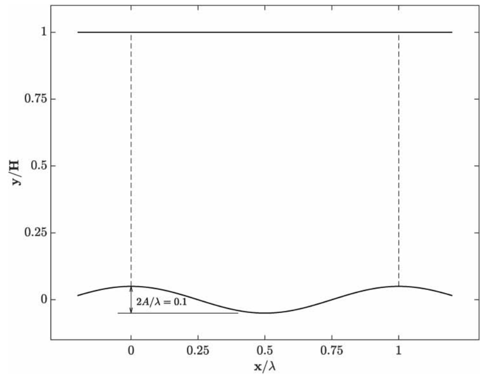

Figure 2. Wavy wall setup.

the reattachment point. All these features are very challenging for turbulence models, making this an interesting test case. Obi et al. (1993) were the first to obtain experimental data for this setup. Later, Buice and Eaton (1995) repeated those experiments, devoting more attention on the reduction of 3 dimensional effects in the flow.

RANS simulations were carried out in OpenFOAM using the $k-\varepsilon$ 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.

Mesh sensitivity

A mesh convergence study was performed using five levels: the baseline mesh and each two coarser and two finder meshes. The individual grid resolutions are given in Table 1, and the non-dimensional grid spacing of the wall-adjacent cells in the inflow section is $\Delta^+ = 3.5$ for the baseline resolution. For each level change, the total number of cells changed by a factor of roughly 2. Profiles of the streamwise velocity for the different meshes are plotted in Figure 2. There is a very good agreement between the results from all mesh resolutions implying a relatively low sensitivity of the flow solutions.

Inflow sensitivity

In order to build confidence in the predictions, the sensitivity of the solution to the inflow conditions was investigated. More specifically, the inlet velocity profile was distorted to vary the centerline velocity between 90% and 110% of its nominal value, while the bulk velocity and thereby the Reynolds number were kept constant. Figure 3 shows the corresponding streamwise velocity profiles. The baseline is plotted in black, and ten simulations with distorted inflows are plotted in red. Near the inflow at $x/H=-10$, the difference in the inflow profiles is well visible. Before the beginning of the expanding section at $x/H=-1$, those differences have already been smeared for the most part, and from there for all downstream segments there is little difference between the profiles, especially with regards to the flow behavior near the bottom wall.

Table 1. Mesh resolutions of diffuser simulations. Baseline computations are based on level 3.

Level	1	2	3	4	5
n_x	75	104	148	209	296
n_y	32	46	64	90	128
n_{tot}	2,400	4,784	9,472	18,810	37,888

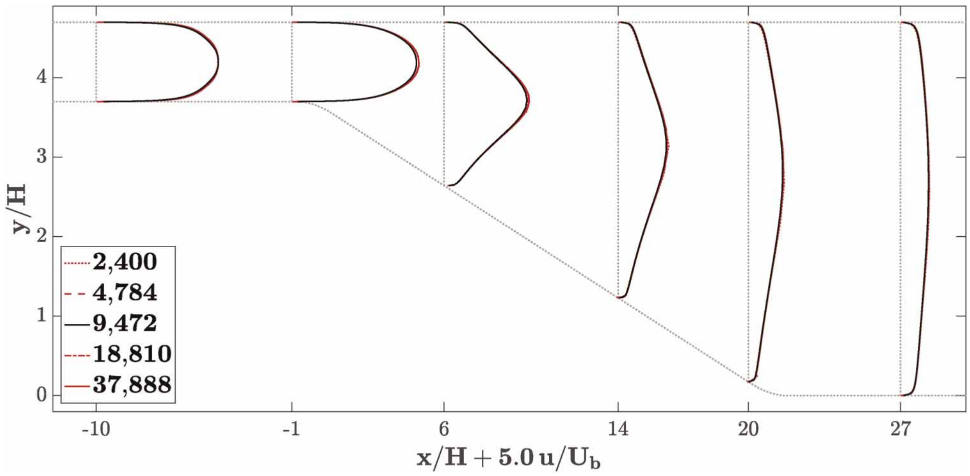

Figure 3. Mesh convergence of diffuser simulations. Profiles of streamwise velocity at different x locations. Baseline is 9,472 cells.

Validation

None of the simulations carried out so far indicate flow separation. Both the grid convergence and the inflow distortion studies provide confidence in the computations as very limited sensitivity to boundary conditions and numerical errors is observed.

The baseline simulation is compared against experimental data from Buice and Eaton (2000) in Figure 4. While the flow remains attached at all times in the RANS simulation, 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

The results presented so far illustrate a typical situation in computational engineering: simulations do not accurately represent experimental findings. Given the simplicity of the present application and the careful sensitivity analysis carried out to assess both numerical errors and boundary conditions, it is clear that the *only* source of inaccuracy in the simulations is the turbulence model. There is no shortage of different turbulence models (Iaccarino, 2001; Cokljat et al., 2003) that are known to provide better representation of the effect of adverse pressure gradients. However, it is useful to establish if it is possible to assess the effect of model inaccuracy and to estimate *bounds* on the predictions to understand the sensitivity of the results to the modeling assumptions.

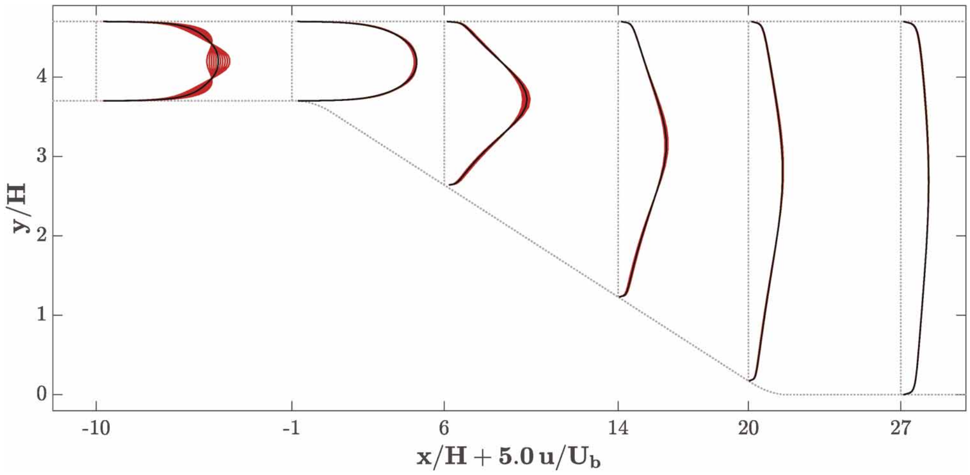

Figure 4. Inflow sensitivity of diffuser simulations. Profiles of streamwise velocity at different x locations. Baseline in black, distorted inflows in red.

Framework

Emory et al. introduced perturbations to eigenvalues of the normalized Reynolds stress anisotropy tensor towards limiting states within the range of realizability (Emory et al., 2001, 2013). This technique is data-free and, in its basic form, parameter-free, and it allows to estimate model form uncertainty stemming from turbulence models. The starting point is to define the Reynolds stress anisotropy tensor

$$a_{ij} = \frac{R_{ij}}{2k} - \frac{1}{3}\delta_{ij} \tag{1}$$

where $R_{ij} = \overline{u'_i u'_j}$ is the Reynolds stress tensor and k is the turbulent kinetic energy. a_{ij} can be diagonalized to

$$a_{ij} = v_{im}\Lambda_{mn}v_{jn} \tag{2}$$

v_{im} is the matrix of orthonormal eigenvectors, and Λ_{mn} is the traceless diagonal matrix of eigenvalues λ_i, with $\lambda_1 \geq \lambda_2 \geq \lambda_3$. These eigenvalues can be visualized as a position in a barycentric map (Banerjee et al., 2007). In that map, all realizable states of turbulence are limited to within a triangle, and the corners of the triangle correspond to limiting states of turbulence with 1, 2, and 3 components respectively. Figure 5 shows this triangle as well as one realizable location inside that triangle $\vec{x}_{LF}$ coming from a low fidelity simulation. This location can be perturbed towards the three limiting states, and a perturbed set of eigenvalues λ_i^* can be reconstructed from the perturbed location $\vec{x}^* = \vec{x}_{LF} + \Delta_B(\vec{x}_{ic} - \vec{x}_{LF})$. In the data-free framework, these perturbed locations are the corners of the triangle ($\Delta_B = 1.0$). A perturbed Reynolds stress anisotropy tensor and a perturbed Reynolds stress tensor follow:

$$a^*_{ij} = v_{im}\Lambda^*_{mn}v_{jn}, \qquad R^*_{ij} = 2k\left(a^*_{ij} + \frac{1}{3}\delta_{ij}\right) \tag{3}$$

These eigenvalue perturbations add three perturbed simulations to the baseline calculations\comma \; 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.

Validation

The framework of eigenvalue perturbations is applied to the present case of the planar asymmetric diffuser. Figure 6 shows the results of the baseline case, the uncertainty envelopes, and the experimental results. The uncertainty envelopes are constructed using the data-free eigenvalue perturbations towards each of the three limiting states with $\Delta_B = 1.0$.

The uncertainty envelopes cover the experimental results in most locations, and in particular at the bottom wall, where the perturbations suggest that the baseline calculation might be overpredicting the streamwise

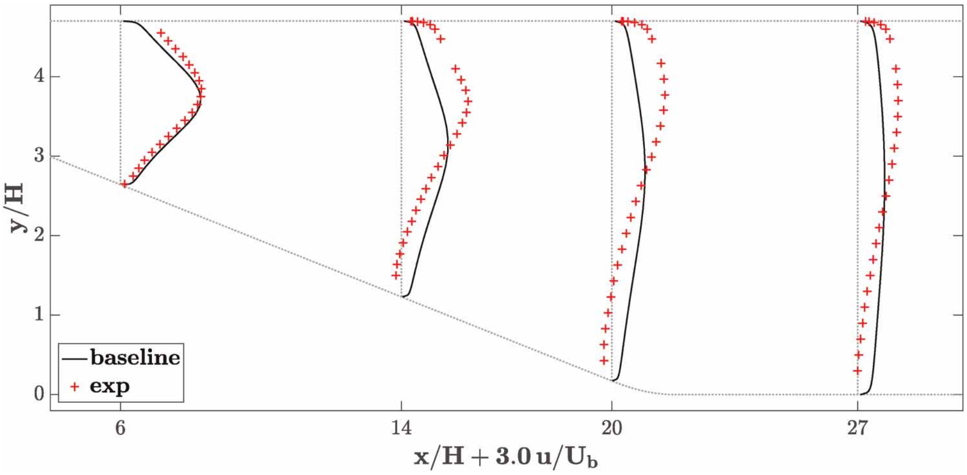

Figure 5. Diffuser simulation. Profiles of streamwise velocity at different x locations with experimental data.

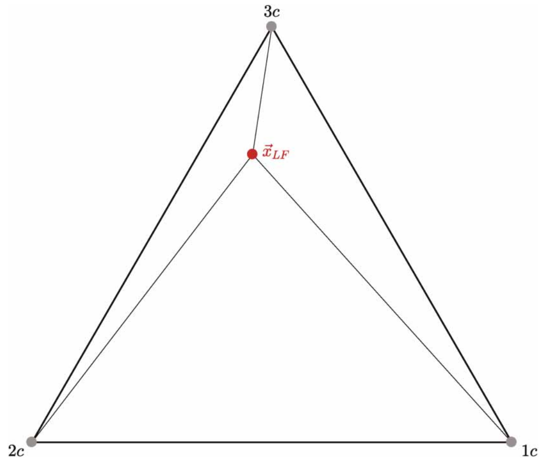

Figure 6. Data-free, uniform eigenvalue perturbation.

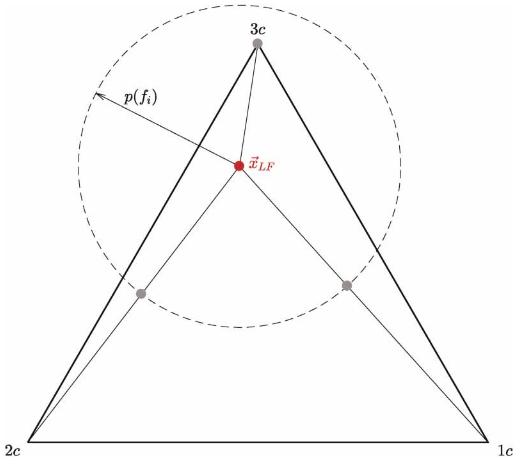

Figure 7. Data-driven, local eigenvalue perturbation. Perturbed locations for 1c, 2c, and 3c limiting state marked as grey dots.

velocity. 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.

For practical applications, it would be useful to bound the uncertainty estimates more tightly, in order to have a more clear indication of the plausible modeling errors. 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. Therefore, it would be desirable to be able to vary the perturbation strength of the eigenvalue perturbations locally. That might allow to reduce the perturbation strength in regions, where less inaccuracy of the turbulence model is expected.

Data-driven uncertainty quantification

Perturbation strength

We study a data-driven approach to predict a local eigenvalue perturbation strength based on 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 7 illustrates the meaning of p in the barycentric map. The original location $\vec{x}_{LF}$ is perturbed towards the same three extreme states as in the data-free approach. 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 perturbed locations are still always within the triangle and therefore within the constraints of realizability; if the predicted perturbation strength is larger than the distance to the respective limiting state then the perturbed location is at that limiting state. This definition of the perturbation strength implies that the effective perturbations cannot be greater than for the data-free case, but they can be smaller.

Machine learning model

A random regression forest is chosen as the machine learning regression model. Random forests are ensemble learners that are based on number of decision trees (Breiman, 2001). The details of regression trees and random forests are discussed in the following subsections. The random forest is implemented using the OpenCV library. More details on the implementation are provided in the appendix.

Regression trees

Regression trees are decision trees that are used to predict ordered variables (Breiman et al., 1984). Decision trees are algorithms that work by defining a hierarchical structure of nodes at which splits are performed. Starting from the root node, at each split one feature is compared against a corresponding threshold value. Depending on the difference between the feature values and the corresponding threshold, the algorithm continues at one of two child nodes. This continues until a so-called leaf node is reached, which assigns a value. Decision trees can be visualized in a tree like structure, hence the name, with the initial root node at the top and the leaves at the bottom.

They are built starting from the root and advancing recursively, finding the optimal split at each node. For the regression problem, the optimal split of a decision tree is the one that minimizes the root mean squared error over all features. Each regression tree is trained on a different random subset of the training set. Furthermore, at each node the optimal split is found for a random subset of features, called the active variables. Adding this randomness ensures that the individual regression trees are decorrelated. The value at a leaf node is the average over all the training samples that fall into that particular leaf node.

This structure enables regression trees 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, which are features that are considered but not relevant for problem (Ling and Templeton, 2015; Milani et al., 2017).

Random forests

Random forests are a supervised learning algorithm. They are ensemble learners, meaning that they leverage a number of simpler models to make a prediction. In this case of a random forest, the simpler models are regression trees.

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

$$\mathrm{MSE}(x) = \underbrace{\left(E\left[\hat{f}(x)\right] - f(x)\right)^2}_{\text{squared bias}} + \underbrace{E\left[\left(\hat{f}(x) - E\left[\hat{f}(x)\right]\right)^2\right]}_{\text{variance}} + \underbrace{\sigma_e^2}_{\text{irreducible error}} \tag{4}$$

where $\hat{f}(x)$ is the model prediction and $f(x)$ is the true value. 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.

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 trade-off.

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).

Features

Decision trees, and machine learning algorithms in general, require the definition of a set of features that represent the mapping from input to outputs.

In the present scenario, 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 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.

The list of features is given in Table 2. For vectors the L_2 norm, and for second-order tensors the Frobenius norm is used. The first five features are traces of different combinations of the mean rate of strain and mean rate of rotation tensors, normalized by the turbulence time scale. One other normalization was tested, using the mean strain and mean rotation time scales, yet using the turbulence time scale yielded better results, i.e. smaller overall MSE. The sixth feature is a non-dimensionalized Q criterion, which is commonly used to define vortex regions. The seventh feature is the ratio of the turbulence kinetic energy production rate to its dissipation rate, governing the turbulence kinetic energy as a key parameter in characterizing the turbulence of the flow. Next are the ratio of turbulence to mean strain time scale, the Mach number, and the turbulence intensity. The turbulence Reynolds number, which can also be interpreted as a non-dimensional wall distance, is the eleventh feature. Finally, a marker function indicating deviation from parallel shear flow is used as a feature, as scalar turbulence viscosity models are geared often geared towards and validated on such parallel shear flows (Gorlé et al., 2012).

Training case setup

Training data is used to determine the model parameters, not to evaluate the model's performance, because unlike the test case it cannot provide an unbiased model assessment. 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 sinusoidal bottom wall as illustrated in Figure 8. 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 $k - \varepsilon$ turbulence model.

Training case mesh sensitivity

The simulations use a structured grid with a resolution of 16,384 cells, 128 in each the x and the y direction. The computational domain covers the space from one wave crest to the next one as indicated by the dashed lines in the figure, with periodic inflow and outflow boundary conditions. A mesh convergence study was carried out with two coarser and two finer meshes. The factor between the total number of cells is approximately 2 between the refinement levels, and the individual grid resolutions are given in Table 3. Figure 9 is showing profiles of the streamwise velocity for the mesh convergence study. There is barely any difference between the different calculations. Features for the random forest training were computed from the baseline case, and labels were computed

Table 2. Non-dimensional features used for the random regression forest.

#	Description	Formula	#	Description	Formula
1	Divergence of velocity	$\mathrm{tr}(\tau_t S)$	7	Ratio of k production to dissipation	$\frac{P}{\varepsilon}$
2	Trace of S^2	$\mathrm{tr}(\tau_t^2 S^2)$	8	Ratio of turbulence to mean strain time scale	$\frac{Sk}{\varepsilon}$
3	Trace of S^3	$\mathrm{tr}(\tau_t^3 S^3)$	9	Mach number	$\frac{\lVert\bar{\mathbf{u}}\rVert_2}{c_0}$
4	Trace of W^2	$\mathrm{tr}(\tau_t^2 W^2)$	10	Turbulence intensity	$\frac{\sqrt{k}}{\lVert\bar{\mathbf{u}}\rVert_2}$
5	Trace of $W^2 S^2$	$\mathrm{tr}\left(\tau_t^4 W^2 S^2\right)$	11	Turbulence Reynolds number	$\min\left(\frac{\sqrt{k}d_w}{50\nu}, 2\right)$
6	Q criterion	$\frac{W^2 - S^2}{W^2 + S^2}$	12	Marker indicating deviation from shear flow	$\frac{\lvert g_j s_j\rvert}{\lVert\mathbf{g}\rVert_2}\frac{\sqrt{k}}{\lVert\bar{\mathbf{u}}\rVert_2}$

The following variables are used: the mean rate of strain and its norm $S_{ij} = (1/2)(\nabla\bar{\mathbf{u}}_{ij} + \nabla\bar{\mathbf{u}}_{ji})$, $S = \lVert S\rVert_F$; the mean rate of rotation and its norm $W_{ij} = (1/2)(\nabla\bar{\mathbf{u}}_{ij} - \nabla\bar{\mathbf{u}}_{ji})$, $W = \lVert W\rVert_F$; the unit vector along the streamline $s_i = \bar{\mathbf{u}}_i/\lVert\bar{\mathbf{u}}\rVert_2$, the gradient of the streamline aligned velocity $g_i = s_j(\partial\bar{\mathbf{u}}_j/\partial x_i)$; and the turbulence time scale $\tau_t = (k/\varepsilon)$.

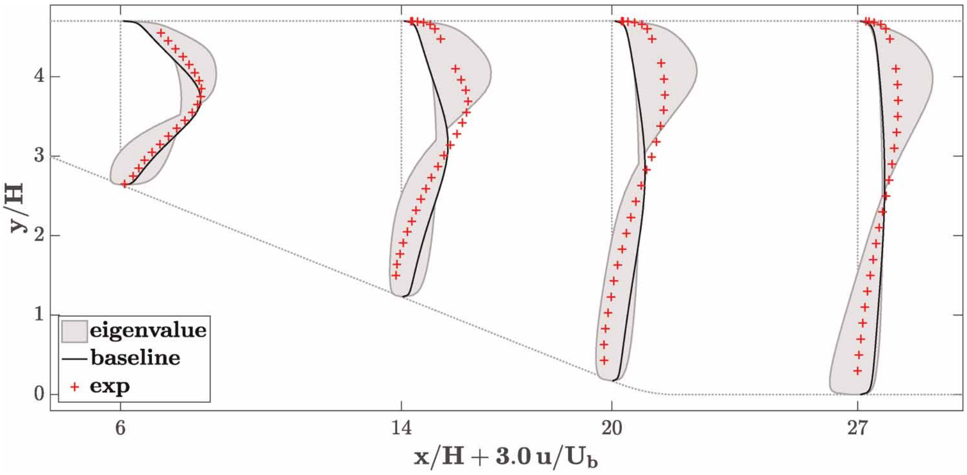

Figure 8. Data-free, uniform eigenvalue perturbation. Profiles of streamwise velocity at different x locations.

from both the baseline case and higher fidelity data, with each RANS grid node being one data point. 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 statistically converged DNS data from Rossi (2006), which was linearly interpolated to the RANS cell locations using the nearest four DNS cell locations for each RANS cell.

Hyperparameters

In machine learning models, hyperparameters are parameters that are not learned during model training, but that instead are 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 node in order to do further splitting. The active variable count is the number of features randomly chosen at each node to find the optimal split. For each of the first three hyperparameters a couple of different values were studied over a range of 1 to 200 regression trees. Figure 10a–c show the training and testing error of the random forest training of those studies plotted against the number of trees. Each solid line corresponds to the training error for a specific value of the particular hyperparameter, and each dotted line corresponds to the testing error. To improve readability, these errors are plotted for every third number of trees only. For these studies, the dataset was first randomly shuffled and then split into a training set comprising 80% of the samples and a testing set comprising 20% of the samples. The shuffling was the same for all individual tests, so that every random forest model in this subsection was trained on the same training samples and tested on the same testing samples. 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 asymmetric diffuser case.

The results from the maximum tree depth study are shown in Figure 10a. 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

Table 3. Mesh resolutions of wavy wall simulations. Baseline is level 3.

Level	1	2	3	4	5
n_x	64	91	128	181	256
n_y	64	91	128	181	256
n_{tot}	4,096	8,281	16,384	32,761	65,536

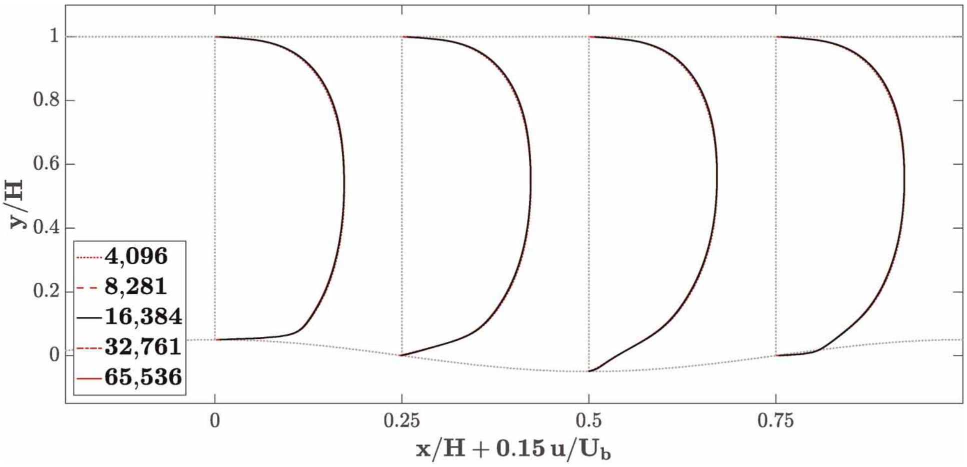

Figure 9. Mesh convergence of wavy wall simulations. Profiles of streamwise velocity at different x locations. Baseline is 16,384 cells.

20. There is no significant difference between 15 and 20. The errors are a bit larger for 10 and significantly larger for 5, which clearly outperformed all other cases in terms of generalization. Because it showed the best performance for training and test error, a maximum tree depth of 15 was chosen. 20 was not chosen, because it increased computational costs without improving prediction performance.

Figure 10b shows the results from the minimum sample count study. The tested values are 10, 20, and 30. All three cases could do better in terms of generalization, with the cases with the larger minimum sample counts doing slightly better. In turn, the overall performance clearly is improving with smaller values. Because the enhanced performance for the smaller values was more clear than the reversed gain in generalisation, 10 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 10c. The results do not vary greatly in terms of generalization. Larger numbers of active variables lead to lower training and test errors. The training errors for 6, 8, and 10 are the lowest and do not differ significantly. The test error for 6 is slightly larger than for 8 and 10. 8 was chosen as active variable count, because shows overall a better performance than smaller values while leading to a stronger decorrelation between the individual regression trees than the similarly performing case with value 10.

Over all three preceding hyperparameter studies the impact of the number of trees can be observed. There is a steep gain in performance for small numbers of trees followed by a quick saturation. The computational costs of evaluation a random forest model scale linearly with the number of trees. Therefore, the minimum number of trees for maximum model performance is sought. In none of the cases, significant improvement was observed for

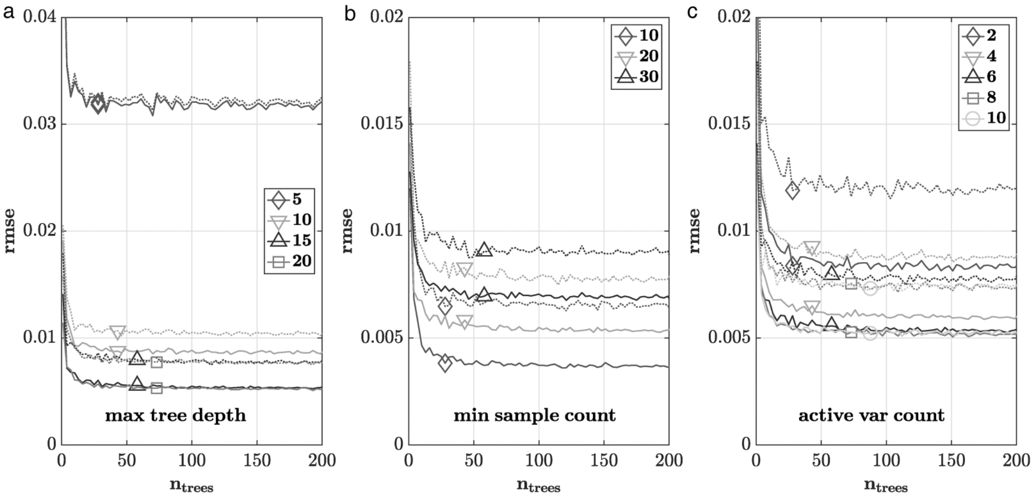

Figure 10. Training error (solid) and testing error (dotted) vs. number of trees for different (a) maximum tree depth values, (b) minimum sample count values, and (c) active variable count values.

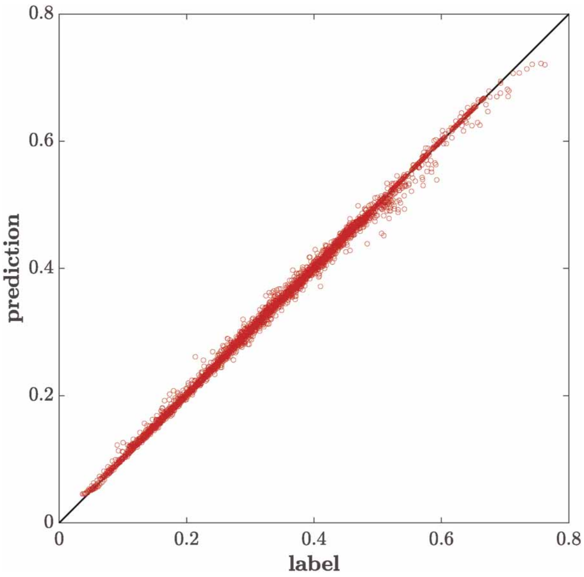

Figure 11. Predictions vs. labels on wavy wall training data for random forest.

a number of trees exceeding 50. Therefore, the final random regression forest was trained for 50 regression trees on the whole dataset. The root mean squared error is 0.00311. Figure 11 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.

Feature importance

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. Table 4 presents the normalized feature importance scores. There is still a lot to learn about the importance of different features for data-driven uncertainty quantification, but we will make a few initial observations.

The two most important features of the trained model are the non-dimensional wall distance and the marker indicating deviation from parallel shear flow. The flow features in the present cases that are the most challenging for turbulence models are flow separation, flow reattachment, and development of a new boundary layer. All these features occur at or near the wall, supporting the significance of the wall distance. The marker for deviation from parallel shear flow was developed specifically to identify regions where the linear eddy viscosity assumption becomes invalid (Gorlé et al., 2014). The importance of this feature indicates that the model was able to highlight this relationship.

Table 4. Normalized feature importance scores.

Feature	Score	#	Score	#	Score
Divergence of velocity	0.000315	Trace of W^2S^2	0.062866	Mach number	0.074620
Trace of S^2	0.012120	Q criterion	0.123786	Turbulence intensity	0.090864
Trace of S^3	0.000271	Ratio of k production to dissipation	0.014759	Turbulence Reynolds number	0.213744
Trace of W^2	0.189448	Ratio of turbulence to mean strain time scale	0.014662	Marker indicating deviation from shear flow	0.202545

The features based on combinations of the mean rate of rotation were clearly more important than the ones based on combinations of the mean rate of strain, with the trace of the squared mean rotation rate tensor being ranked as third most important feature. Other relevant features are the Q criterion, identifying vortex regions, and the turbulence intensity. The turbulence level naturally has an impact on the accuracy of the turbulence model. As expected 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 and related to the training dataset.

Validation

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. As for the data-free uncertainty envelopes, three perturbed calculations were carried out for the three limiting states of turbulence. Figure 12 shows the results of the baseline case, the results of the uncertainty bands from the data-driven eigenvalue perturbations, and the experimental 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

This study is the first step into using data to enhance uncertainty estimation. Based on an existing data-free framework, a new, data-driven approach was introduced in which the perturbations to the Reynolds stress anisotropy eigenvalues is estimated using computable flow features. A random forest was trained on a wavy wall high-fidelity dataset and tested on the prediction of the turbulent separated flow in a diffuser case, with promising early results. The newly predicted uncertainty envelopes closely tracks the experimental data. At the same time, there are still a number of open questions and directions for future work on this topic.

For this study, only one training case and one test case were used. One open question is what impact the choice of the training case(s) has on the results of the test case(s) in this framework. The two cases were relatively similar in terms of the flow features they exhibit, and we expect that this is a necessary condition for accurate uncertainty predictions. Using training data from different cases could make the model applicable in a wider range of test cases.

Just like the choice of test case, the choice of features is an aspect that requires further investigation. This process could be informative for learning which flow features have a stronger impact on uncertainty in the flow predictions. Similarly, the non-dimensionalization of the features gives room for exploration, which again is

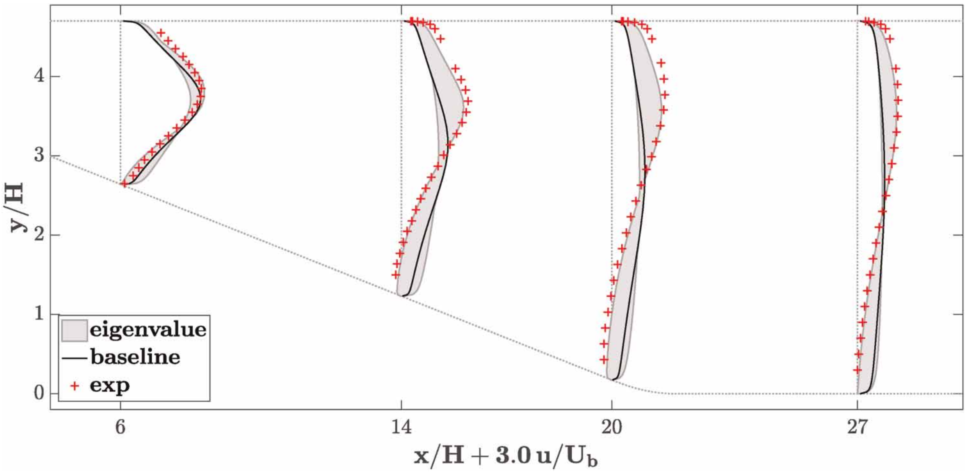

Figure 12. Data-driven, local eigenvalue perturbation. Profiles of streamwise velocity at different x locations.

crucial for generalization. A different time scale was tested for some of the features, but there are more options to consider.

Lastly, different choices could be made regarding the machine learning model. The current choice is a random regression forest. A hyperparameter study was used to decide on a set of parameters, but their effect on the results of the test case are not conclusive. To reduce computational expenses, it would be compelling to find the minimum number of trees needed before the quality of the uncertainty predictions breaks down. And beyond random regression forests, it is useful to compare different types of machine learning approaches.

Appendix: Implementation Notes

Machine learning in OpenCV

The random forest was implemented in C++ using the OpenCV library. Although the OpenCV library was originally developed for computer vision applications, it does include several machine learning models. The `Rtrees` class is used for the random forest. A number of parameters can be set, such as the hyperparameters discussed in the Machine Learning Model section. When the `calculateVarImportance` property is set to true, the importance of the features can be reported after model training. For the random forests used for this study, the training took on the order of 10 s per model. The parameters of the trained models can be exported and imported in a different application.

Connecting OpenFOAM and OpenCV

All RANS calculations were performed using OpenFOAM. For online predictions of the perturbation strength, OpenFOAM and OpenCV were therefore connected. A modified $k-\varepsilon$ turbulence model class was implemented allowing for data-driven eigenvalue perturbations. This turbulence model had an instance of `Rtrees` as member variable. In the constructor of the turbulence model, the random forest was initialized and the model parameters from a trained random forest were imported.

Perturbations in OpenFOAM

During the RANS simulations in OpenFOAM, there are two loops over all cells in the discretized computational domain. For each cell, first all features are computed from the mean flow quantities. Next, the local perturbation strength is predicted using the random forest. Then, an eigenvalue decomposition of the local Reynolds stress anisotropy tensor is performed. The eigenvalues are projected into the barycentric domain and perturbed at the respective perturbation strength, and a perturbed Reynolds stress anisotropy tensor is reconstructed. These loops over all cells are the only difference during runtime between the modified and the standard $k-\varepsilon$ turbulence model.

A test was performed, where the perturbation strength in the modified turbulence model was hard-coded to zero. The results were identical with the results from the standard turbulence model.

Computational costs

The computation of the data-driven eigenvalue perturbations leads to an increase in costs for the RANS simulations. Compared to the calculations with the baseline $k-\varepsilon$ turbulence model, the observed runtime increases by a factor of 2–3. There are two reasons for this increase. First, the calculations associated with the perturbations took some time. Of that time, calculating features, evaluating the random forest, and perturbing the Reynolds stresses accordingly took typically 2–3%, 80–85%, and 15%, respectively. The dominant evaluation time of the random forest scales linearly with the number of trees, so this is a number that potentially could be reduced. Second, the perturbations had an effect of the convergence of the solver, resulting in more iterations that had to be completed. While the data-driven perturbations took roughly the same amount of time for every case, the convergence difficulties were dependent on the particular limiting state.

Acknowledgements

The authors would like to thank Mr. Rossi for providing the DNS dataset of the wavy wall case.

Funding sources

This work was partially supported by a US DOE award funded through the Los Alamos National Laboratory.

Competing interests

Jan F. Heyse declares that he has no conflict of interest. Aashwin A. Mishra declares that he has no conflict of interest. Gianluca Iaccarino declares that he has no conflict of interest.

References

Banerjee S., Krahl R., Durst F., and Zenger C. (2007). Presentation of anisotropy properties of turbulence, invariants versus eigenvalue approaches. *Journal of Turbulence*. 8: N32. https://doi.org/10.1080/14685240701506896

Breiman L. (2001). Random forests. *Machine Learning* 45 (1): 5–32. https://doi.org/10.1023/A:1010933404324

Breiman L., Friedman J., Stone C. J., and Olshen R. A. (1984). *Classification and Regression Trees*, Taylor & Francis, Chapman and Hall/CRC.

Buice C. U. and Eaton J. K. (1995). Experimental investigation of flow through an asymmetric plane diffuser. In: *Center for Turbulence Research Annual Research Briefs*.

Buice C. U. and Eaton J. K. (2000). Experimental investigation of flow through an asymmetric plane diffuser: (data bank contribution). *Journal of Fluids Engineering*. 122 (2): 433–435. https://doi.org/10.1115/1.483278

Cokljat D., Kim S. E., Iaccarino G., and Durbin P. (2003). A comparative assessment of the v2f model for recirculating flows. In: *41st Aerospace Sciences Meeting and Exhibit*, p. 765.

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. https://doi.org/10.1016/0142-727X(95)00079-6

Emory M., Pecnik R., and Iaccarino G. (2001). Modeling structural uncertainties in reynolds-averaged computations of shock/boundary layer interactions. In: *49th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*, p. 479.

Emory M., Larsson J., and Iaccarino G. (2013). Modeling of structural uncertainties in reynolds-averaged navier-stokes closures. *Physics of Fluids*. 25 (11): 110822. https://doi.org/10.1063/1.4824659

Geneva N. and Zabaras N. (2019). Quantifying model form uncertainty in reynolds-averaged turbulence models with bayesian deep neural networks. *Journal of Computational Physics*. 383: 125–147. https://doi.org/10.1016/j.jcp.2019.01.021

Gorlé C., Emory M., Larsson J., and Iaccarino G. (2012). Epistemic uncertainty quantification for rans modeling of the flow over a wavy wall. In: *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. https://doi.org/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. https://doi.org/10.1017/S002211207200268X

Iaccarino G. (2001). Predictions of a turbulent separated flow using commercial CFD codes. *Journal of Fluids Engineering*. 123 (4): 819–828. https://doi.org/10.1115/1.1400749

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. https://doi.org/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): 021006. https://doi.org/10.1115/1.4038275

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*, Kyoto, Japan, p. 305.

Pope S. B. (1978). An explanation of the turbulent round-jet/plane-jet anomaly. *AIAA Journal*. 16 (3): 279–281. https://doi.org/10.2514/3.7521

Rossi R. (2006). Passive scalar transport in turbulent flows over a wavy wall, PhD thesis, Università degli Studi di Bologna, Bologna, Italy.

Sarkar S. (1992). The pressure–dilatation correlation in compressible flows. *Physics of Fluids A: Fluid Dynamics*. 4 (12): 2674–2682. https://doi.org/10.1063/1.858454

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): 061012. https://doi.org/10.1115/1.4024787

Singh A. P., Medida S., and Duraisamy K. (2017). Machine-learning-augmented predictive modeling of turbulent separated flows over airfoils. *AIAA Journal*. 55 (7): 2215–2227. https://doi.org/10.2514/1.J055595

Wang Q. and Dow E. A. (2010). Quantification of structural uncertainties in the k-omega turbulence model. In: *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): 074602. https://doi.org/10.1103/PhysRevFluids.3.074602