

Title of Grant / Cooperative Agreement:	A Framework for Turbulence Modeling Using Big D.
Type of Report:	Summary of Research
Name of Principal Investigator:	Karthikeyan Duraisamy
Period Covered by Report:	9/1/15-12/31/17
Name and Address of recipient's institution:	Regents of the University of Michigan, 503 Thomp
NASA Grant / Cooperative Agreement Number:	NNX15AN98A

Reference 14 CFR § 1260.28 Patent Rights (abbreviated below)

The Recipient shall include a list of any Subject Inventions required to be disclosed during the preceding year in the performance report, technical report, or renewal proposal. A complete list (or a negative statement) for the entire award period shall be included in the summary of research.

Subject inventions include any new process, machine, manufacture, or composition of matter, including software, and improvements to, or new applications of, existing processes, machines, manufactures, and compositions of matter, including software.

Have any Subject Inventions / New Technology Items resulted from work performed under this Grant / Cooperative Agreement?	No <input checked="" type="radio"/>	Yes <input type="radio"/>
If yes a complete listing should be provided here: Details can be provided in the body of the Summary of Research report.		

Reference 14 CFR § 1260.27 Equipment and Other Property (abbreviated below)

A Final Inventory Report of Federally Owned Property, including equipment where title was taken by the Government, will be submitted by the Recipient no later than 60 days after the expiration date of the grant. Negative responses for Final Inventory Reports are required.

Is there any Federally Owned Property, either Government Furnished or Grantee Acquired, in the custody of the Recipient?	No <input checked="" type="radio"/>	Yes <input type="radio"/>
If yes please attach a complete listing including information as set forth at § 1260.134(f)(1).		

Attach the Summary of Research text behind this cover sheet.

Reference 14 CFR § 1260.22 Technical publications and reports (December 2003)

Reports shall be in the English language, informal in nature, and ordinarily not exceed three pages (not counting bibliographies, abstracts, and lists of other media).

A Summary of Research (or Educational Activity Report in the case of Education Grants) is due within 90 days after the expiration date of the grant, regardless of whether or not support is continued under another grant. This report shall be a comprehensive summary of significant accomplishments during the duration of the grant.

A Framework for Turbulence Modeling using Big Data: Phase II Final Report

Karthik Duraisamy (PI)
University of Michigan, Ann Arbor, MI

Juan J. Alonso (Co-I)
Stanford University, Stanford, CA

Paul Durbin (Co-I)
Iowa State University, Ames, IA

Contents

1	Introduction	3
2	Relevant prior work	4
3	Data-driven Turbulence modeling: Key issues	5
4	Field Inversion and Machine Learning (FIML) for Turbulence Modeling	6
5	Channel Flow: A tutorial on FIML	6
5.1	Introducing embedded discrepancy	8
5.1.1	Time-scale discrepancy	8
5.1.2	Discrepancy in anisotropy	8
5.2	Field inversion - Formulation	9
5.3	Field inversion - Results	10
5.3.1	Time-scale discrepancy	10
5.3.2	Discrepancy in anisotropy	10
5.4	Machine Learning-augmented prediction and uncertainty quantification	10
6	Inference in complex problems	13
6.1	Boundary layer flow over a curved surface	14
6.2	Boundary layer separation over a wall-mounted hump	14
6.3	Separated flows over airfoils	16
7	Machine learning and predictions in complex problems	16
7.1	Feature selection	16
7.2	Learning	19
7.3	Prediction	19
8	Conclusions	20
9	Symposium on Advances in Turbulence Modeling	22

1 Introduction

Turbulence models based on the Reynolds Averaged Navier–Stokes (RANS) equations remain the workhorse in the computation of high Reynolds number wall-bounded flows. While these models have proven their utility in an industrial setting, their deficiencies in modeling complex flows is well-documented. Even the most sophisticated turbulence models invoke radically simplifying assumptions about the structure of the underlying turbulence. As a result, even if a model is based on physically and mathematically appealing ideas, the model formulation typically devolves into the calibration of a large number of free parameters or functions using a small set of canonical problems.

Over the past decade or so, however, our ability to perform detailed high-fidelity computations and resolved measurements has improved dramatically. At the same time, data science is on the rise because of improvements in computational power and the increased availability of large data sets. This has been accompanied by significant improvements in the effectiveness and scalability of data analytics and machine learning (ML) techniques. Given these advances, we believe that data-driven modeling and machine learning will play a critical role in improving the understanding and modeling of turbulence. This was the thesis of the proposal we had submitted to the NASA LEARN program in the summer of 2013.

The vision for Phase I of our effort, which began in January 2014 was driven by a simple desired future: can we use vast amounts of data, machine learning techniques, and a detailed understanding of the physics of turbulence to setup a framework that can lead to better predictive capabilities in future RANS models? The vision included the delivery of advanced RANS turbulence models not as a set of equations and pre-specified coefficients (such as what would be found in a typical publication) but, rather, as a set of equations and software modules that can be coded into and called from existing and future CFD solvers. While the equations would be fixed, the accompanying software modules could be updated as additional data becomes available for the data-driven generative process, and could be particularized for higher accuracy (by selecting appropriate datasets to particular phenomena and parameter ranges of interest) to specific situations and/or problem domains.

With that in mind, our work in this project was guided by seeking answers to a series of fundamental questions including: 1. Is there merit in the idea of marrying Big Data methods and Turbulence Modeling?

2. What are the best ways to setup well-posed, data-driven-turbulence-modeling problems?

3. What are the most effective ways to use Machine Learning approaches? Are new techniques necessary to transition from standard classification problems (from web applications) to regression approaches that are more appropriate to modeling the physical world?

4. Once a model has been learned, how is it best embedded in an existing CFD solver?

5. What are the best learning algorithms that provide the necessary accuracy but, at the same time lead to convergence of the RANS solver in a reasonable amount of time?

6. What data (and how much data) is needed to improve predictive capabilities? Can different data sets be used in different modeling situations or must we seek a generic model that is uniformly applicable to a broad range of problems?

7. What improvements can be shown in a number of flows of interest?

2 Relevant prior work

With a view towards quantifying model errors, several researchers [1, 2, 3, 4] have used experimental data to infer model parameters. Cheung et al. [5, 6] employ Bayesian model averaging [7] to calibrate model coefficients. Edeling et al. [3] use statistical inference on skin-friction and velocity data from a number of boundary layer experiments to quantify parametric model error. These methods provide insight into parametric uncertainties and address some of the deficiencies of *a priori* processing of data.

Dow and Wang [8, 9] made progress towards addressing non-parametric uncertainties by inferring the spatial structure of the discrepancy in the eddy viscosity coefficient based on a library of direct numerical simulation (DNS) datasets. The discrepancy between the inferred and modeled eddy viscosity was represented as a Gaussian random field and propagated to obtain uncertainty bounds on the mean flow velocities. The research group of Iaccarino [10, 11, 12] introduced adhoc, but realizable perturbations to the non-dimensional Reynolds stress anisotropy tensor a_{ij} to quantify structural errors in eddy viscosity models. Tracey et al. [13] applied neural networks to large eddy simulation data to learn the functional form of the discrepancy in the eigenvalues of a_{ij} and injected these functional forms in a predictive simulation in an attempt to obtain improved predictions. Xiao and co-workers [14] inferred the spatial distribution of the perturbations in a_{ij} and turbulent kinetic energy by assimilating DNS data. Weatheritt [15] uses evolutionary algorithms on DNS data to construct non-linear stress-strain relationships for RANS models.

Ling and Templeton [16] used machine learning-based classifiers to ascertain regions of the flow in which commonly-used assumptions break down. King et al. [17] formulated a damped least squares problem at the test-filter scale to obtain coefficients of a subgrid-scale model. In both of these works, results were demonstrated in an apriori setting.

In the LEARN program [18, 19, 20, 21], we took the first steps towards improving predictive model forms by defining a data-driven modeling paradigm based on field inversion and machine learning (FIML). The FIML approach consists of three key steps : a) Inferring the spatial (non-parametric) distribution of the model discrepancy in a number of problems using Bayesian inversion, b) Transforming the spatial distribution into a functional form (of model variables) using machine learning, and c) Embedding the functional form in a predictive setting. Predictions were demonstrated in turbulent and transitional flows with imposed pressure gradients. A key advance in this technique was the idea of connecting inversion and machine learning towards the end of functional corrections. Note that steps a) and b) involve off-line (training) computations, whereas step c) is on-line (prediction).

Ling et al. [22] and Xiao and co-workers [23] bypass the inference step and use machine learning directly on DNS data to obtain a more comprehensive model (compared to Tracey et al. [13]) for the anisotropy tensor a_{ij} . Ling et al. [22] use neural networks to reconstruct a_{ij} and propagate the model to velocity field predictions. Xiao et al. [23] use random forests to inject the machine learned discrepancy as a one time post-processing step to a computed baseline solution.

The FIML paradigm was further extended in Ref. [24] to the prediction of turbulent separated flows around airfoils and much improved predictions were demonstrated in geometries and flow conditions that were not part of the training data. The resulting machine learned model was embedded in a commercial finite element solver (*Accusolve*) and accuracy improvements were confirmed, thus demonstrating portability.

In this Phase II report, we pursue the following:

1. To describe the paradigm of combining inference and machine learning using a simple turbulence modeling example,

2. To explain the role of inverse modeling as a critical step in data-driven turbulence modeling,
3. To explain how inverse modeling can be used to extract turbulence modeling information in data-sparse and data-rich situations and how machine learning can convert this information into modeling knowledge,
4. To demonstrate how FIML can be used to augment existing turbulence models in a predictive setting

3 Data-driven Turbulence modeling: Key issues

Several observations are pertinent to successfully pursuing data-driven techniques for the development of turbulence models:

1. Predictive models of turbulence are formulated as a set of equations with parameters and functional forms that cannot be determined from theory alone. Additionally, it is not obvious as to how – or what aspects of – a model can benefit from data.

2. Even if individual terms in the closure equations are modeled accurately, small discrepancies can lead to poor results. For instance, Poroseva and Murman [25] observed that even if some terms in a second moment closure are extracted directly from DNS, the overall prediction was not satisfactory. It is thus the **balance** between every term in the model equation that is important.

3. The information required by the models is not immediate in the DNS/LES/experimental data. This information must be created from data. The required information is typically embedded in a number of quantities such as time scales, length scales, etc. Since turbulence models are typically formulated to provide first and second moments, these scale variables assume different values in closure models compared to their values in the real-world or in higher-fidelity simulations.

4. The information being extracted can be defined only in terms of the turbulence closure. This restricts the use of approaches such as direct data mining and pattern recognition applied in isolation to DNS data sets. As an example, the use of a constitutive relationship (or wall damping functions) directly from DNS data may become inconsistent in a predictive modeling setting because intermediate variables such as the dissipation rate will be highly erroneous in a RANS model (the point of application) whereas it would be accurate in the DNS data (the point of learning).

4. Information extracted from *specific* flow configurations has to be converted to more *general* modeling knowledge. Machine learning can enable this transformation, but this issue will be further complicated by the fact that turbulence is a strongly non-local phenomena and that the resulting models must be realizable, satisfy invariance properties, etc.

5. The machine-learned corrections should be inactive when the predictive accuracy of the baseline model is acceptable.

6. A range of uncertainties should be accounted for and propagated to the outputs. This includes uncertainties/errors in the data, numerical errors, uncertainty in the problem setup (especially if the data comes from an experiment), variability in the setup of the inverse problems, probabilistic nature of the machine learning algorithm, etc.

7. When embedded in a predictive setting, the inversion/machine learning-augmented turbulence model should be efficient and have good convergence properties.

4 Field Inversion and Machine Learning (FIML) for Turbulence Modeling

A schematic of the FIML methodology is shown in Figure 1. The following section will offer specific examples of the various quantities used in this work.

Given an existing model, a model correction field ¹ $\beta(\mathbf{x})$ is introduced into the transport equation. Starting with some high fidelity or experimental data \mathbf{G}_d , an inverse problem will be formulated to minimize (an appropriately regularized form of) $\|\mathbf{G}_d - \mathbf{G}_\beta\|$, where \mathbf{G}_β is the predicted output for a given correction field. Note that the intent is not to infer a limited set of parameters that characterizes the correction, but *rather to extract the correction field itself*. In other words, β is sought at every discrete location in the computational domain and inserted into the transport equation. *Thus we are directly addressing structural errors and uncertainties in the model.*

The correction fields will be obtained for an ensemble of problems (on different data sets $\mathbf{G}_d^1, \mathbf{G}_d^2, \dots$ as shown in the figure) representative of the physical phenomena that is to be modeled. Unlike in parameter estimation, in which the inferred parameters can be used directly in simulations, the $\beta(\mathbf{x})$ field by itself is of limited use in quantitative modeling. In other words, having inferred the optimal model correction function β , it remains to **convert the inference into modeling knowledge**.

To be useful in predictive modeling, the spatial dependence of $\beta(\mathbf{x})$ has to be transformed into feature space using machine learning ². Machine learning effectively constructs a functional form $\beta(\boldsymbol{\eta})$, where $\boldsymbol{\eta}(\mathbf{U}(\mathbf{x}))$ are features (derived from flow and turbulence variables \mathbf{U}) that will be available during the solution. The elements of the feature vector $\boldsymbol{\eta}$ have to be preferably locally non-dimensional quantities³ such that the functional corrections can be translated across different problems.

It has to be recognized that the inversion and machine learning are *pre-processing* steps as shown in the red box. During the predictive simulation, (at each time-step or solver iteration), the solver will pass feature vectors $\hat{\boldsymbol{\eta}}$ to the ML evaluation routine and receive appropriate model correction quantities $\hat{\beta}$ for injection into the data-augmented turbulence model.

5 Channel Flow: A tutorial on FIML

For purposes of demonstration, we will focus attention on planar turbulent channel flow in this section. This canonical problem has been heavily utilized to calibrate model parameters throughout the history of turbulence modeling, although most models do not account for the variability due to Reynolds number. The question of choosing the right combination of length and time scales is still an outstanding one in modeling. Rich Direct Numerical Simulation (DNS) datasets[26, 27, 28] exist for such flows. Again, for illustrative purposes, we consider the Wilcox $k - \omega$ model [29].

The Reynolds-averaged momentum equation for a fully developed, incompressible and steady channel flow is given by

$$\frac{\partial}{\partial y} \left[\mu \frac{\partial u}{\partial y} - \rho \overline{u'v'} \right] - \frac{\partial p}{\partial x} = 0, \quad (1)$$

¹ β is a spatial field if the problem is steady and a spatio-temporal field if it is unsteady

²The machine learning step may be avoided by clever trial and error model development

³ $|S|\tau$ is an example of an acceptable feature, where $|S|$ is the magnitude of the strain-rate tensor and τ is a turbulent time-scale.

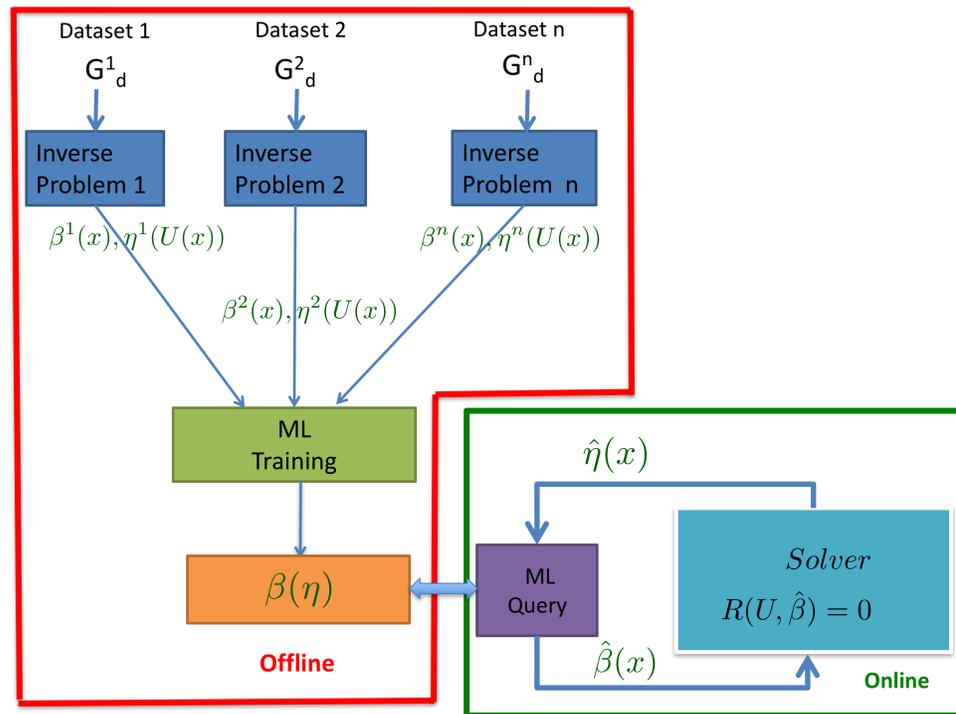


Figure 1: Schematic of field inversion and machine learning framework for data-augmented turbulence modeling

where, u represents the mean streamwise velocity and the Reynolds stress $-\overline{u'v'}$ is modeled as $\nu_t \frac{\partial u}{\partial y}$ in eddy viscosity models. In the Wilcox $k - \omega$ model, $\nu_t = \frac{k}{\omega}$, where k and ω satisfy the following transport equations:

$$\nu_t \left(\frac{\partial U}{\partial y} \right)^2 - \alpha^* k \omega + \frac{\partial}{\partial y} \left[\left(\nu + \sigma^* \frac{k}{\omega} \right) \frac{\partial k}{\partial y} \right] = 0, \quad (2)$$

$$\gamma \left(\frac{\partial U}{\partial y} \right)^2 - \alpha \omega^2 + \frac{\partial}{\partial y} \left[\left(\nu + \sigma \frac{k}{\omega} \right) \frac{\partial \omega}{\partial y} \right] = 0, \quad (3)$$

$$\nu_t = k/\omega ; R_{ii} = \frac{2}{3}k ; R_{12} = R_{21} = -\nu_t \frac{\partial u}{\partial y}.$$

The model assumes standard constants [29]. The system is solved on a geometrically graded mesh consisting of 201 points with the first grid point placed well into the viscous sublayer at $y^+ \approx 0.05$.

5.1 Introducing embedded discrepancy

We pursue two different approaches to introducing embedded discrepancy functions. The first of these focuses on the scale-providing equation. The second approach investigates the perturbations to the predicted anisotropy tensor.

5.1.1 Time-scale discrepancy

In this setting, a function $\beta(y)$ is introduced [30, 31] as a multiplier to the production term in the ω equation.

$$\beta(y) \gamma \left(\frac{\partial U}{\partial y} \right)^2 - \alpha \omega^2 + \frac{\partial}{\partial y} \left[\left(\nu + \sigma \frac{k}{\omega} \right) \frac{\partial \omega}{\partial y} \right] = 0. \quad (4)$$

5.1.2 Discrepancy in anisotropy

In this setting, the eigenvalues of the anisotropy tensor are perturbed [32, 23]. The anisotropy tensor is given as:

$$\mathbf{a} = \frac{\mathbf{R}}{2k} - \frac{1}{3} \mathbf{I} = \mathbf{V} \mathbf{\Lambda} \mathbf{V}^T$$

If $\{\mathbf{\Lambda}, \mathbf{V}\}$ represent the eigen system of \mathbf{a} , then by perturbing the eigenvalues, the Reynolds stress tensor can be modified as below:

$$\mathbf{R}_{pert} = 2k \left[\frac{1}{3} \mathbf{I} + \mathbf{V} (\mathbf{\Lambda} + \boldsymbol{\beta}(y)) \mathbf{V}^T \right]$$

In this case, $\boldsymbol{\beta}(y) = \{\delta_x(y), \delta_y(y)\}$ will be inferred from data.

5.2 Field inversion - Formulation

The next step is to extract the spatial distribution of the embedded discrepancies from data. In addition to determining an optimal discrepancy function, we are also interested in characterizing the impact of uncertainties in the data, model inadequacies and existing knowledge, on the inferred outputs. The Bayesian approach [33] provides a formalism to quantify the posterior probability distribution $q(\beta|d)$, given a prior distribution ($p(\beta)$), data vector, d , and a likelihood function, $h(d|\beta)$. The Bayes theorem states that,

$$q(\beta|d) = \frac{h(d|\beta)p(\beta)}{c}, \quad (5)$$

where $c = \int h(d|\beta)p(\beta)d\beta$. In the current form, the solution to equation 5 is intractable using a sampling based method because of the infinite dimensional nature of β . The problem is made finite-dimensional by re-defining it such that value of β is inferred at every point in the computational domain of the RANS grid⁴.

In this work, the prior probability distribution $p(\beta)$ is assumed to be Gaussian. Further, since the spatially varying form of β is desired, the number of parameters to be estimated is equal to the number of grid points in the computational domain⁵. To promote tractability of the computation of the posterior properties, $p(\beta|d) \equiv e^{-(\beta-\beta_{prior})^T C_{prior}^{-1}(\beta-\beta_{prior})}$ and $h(d|\beta) \equiv e^{-F^T C_{obs}^{-1} F}$ are approximated to be Gaussian. In these expressions, F is a vector with i^{th} element and $f_i = d_{i,RANS} - d_{i,benchmark}$. d_i is the i^{th} data point. C_{obs} and C_{prior} are the observational and the prior covariance matrices. β_{prior} is the prior mean of the parameters (and also corresponds to the *base* model). Under Gaussian assumptions, the maximum a posteriori (MAP) estimate [33] is taken to be the representative of the mean of distribution. The MAP can be computed by maximizing the numerator in Eq. 5, or equivalently by solving the following deterministic minimization problem:

$$\beta_{MAP} = arg \min_{\beta} \mathfrak{J}(\beta) = arg \min_{\beta} \frac{1}{2} \left[F^T C_{obs}^{-1} F + (\beta - \beta_{prior})^T C_{prior}^{-1} (\beta - \beta_{prior}) \right], \quad (6)$$

The resulting optimization problem is still high-dimensional but efficiently solved using adjoint-driven techniques, but the posterior distribution is not determined by sampling methods and instead, the covariance is *approximated* by linearizing about the MAP point. This leads to a definition of the covariance in terms of the inverse of the Hessian of the objective function, $\mathfrak{J}(\beta)$,

$$C_{posterior} = \left[\frac{d^2 \mathfrak{J}(\beta)}{d\beta d\beta} \right]^{-1} \Big|_{\beta_{MAP}}. \quad (7)$$

The error introduced by the Gaussian assumption is difficult to estimate without the use of sampling strategies, which are considered in Ref. [31].

The posterior distribution β can be obtained by first performing a Cholesky decomposition of the form⁶

$$R^T R = C_{posterior}, \quad (8)$$

⁴Note that is still a very high-dimensional problem.

⁵This dimensionality could be reduced to a degree by expressing β as a weighted sum of appropriately defined basis functions

⁶ R is an upper triangular matrix

and generating realizations using

$$\beta = \beta_{MAP} + R^T s, \quad (9)$$

where, s is a vector of the same size as β and contains normally distributed random numbers with zero mean and unit standard deviation. The realizations of the posterior β are used to construct the posterior realizations of various flow quantities by solving the RANS equations for each realization of β . The prior realizations are constructed in a similar manner using C_{prior} .

To summarize, the above inversion procedure involves:

1. Solution of a deterministic optimization problem with the objective function $\mathfrak{J}(\beta)$, typically using a gradient based method, to obtain the MAP model.
2. Building an approximation of the posterior covariance by the inverse of the Hessian around the MAP point.

5.3 Field inversion - Results

5.3.1 Time-scale discrepancy

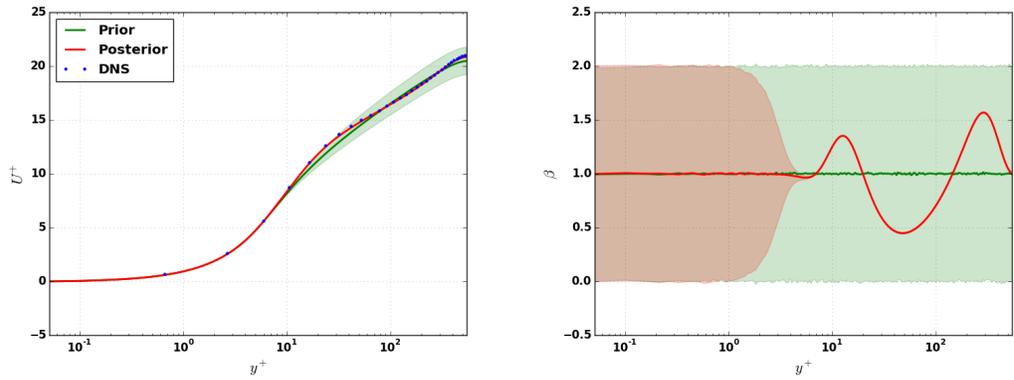
Figure 2 shows the prior, posterior and DNS values at $Re_\tau = 550$ for the inference involving time-scale discrepancy. The posterior velocity profile is almost identical to the DNS profile with the standard deviation collapsing to a very low value. The accuracy of the posterior can be further verified from the plot of $y^+ du^+ / dy^+$. As turbulent production is negligible for $y^+ < 1$, the posterior solution is not affected by $\beta(y)$ and hence $\beta(y)$ remains same as the prior. Correspondingly, there is also no reduction in uncertainty in the viscous sublayer for $y^+ < 2$. Outside of the sublayer, $\beta(y)$ changes to match the specified benchmark velocity profile and a drastic reduction in the posterior standard deviation is noticeable. Figure 3 shows the inferred function, β , for all Reynolds numbers. In every case, β in the overlap and the log-layer scales with wall units. In the outer region, β scales with wall height. This is a physically reasonable variation and represents the systematic nature of modeling deficiency. The large variation in β around $y^+ = 25$ shows the inability of the underlying model to provide the correct time scale near the interface of the overlap region and the log-layer. It is particularly revealing that the nature of the required correction is almost universal (with second-order viscous corrections) for $y^+ < 50$, a fact that can be used to develop improved models.

5.3.2 Discrepancy in anisotropy

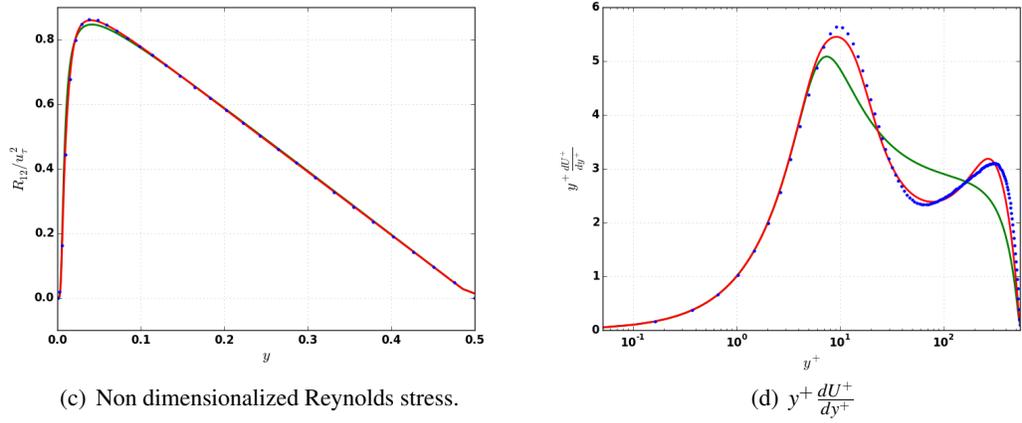
Figure 4a shows the prior and posterior velocity for the perturbation in Barycentric coordinates. The corresponding perturbations are shown in Figure 4b. As expected, the baseline model aligns itself with the plane strain line. The inferred perturbations, which guarantee that the velocity and Reynolds shear stress match the DNS, suggest a different anisotropy compared to the DNS. Figure 5 shows that these perturbations may exhibit universal behavior.

5.4 Machine Learning-augmented prediction and uncertainty quantification

Having inferred the discrepancy for specific problems, the role of machine learning is to map the inferred discrepancy from physical space to feature space. For purposes of brevity, only the anisotropy perturbations are shown here. Gaussian process regression [34] was used for this purpose. The fields



(a) Non dimensionalized velocity, U^+ . Mean is shown with the solid line; shaded region represents the 95% confidence interval. (b) Correction function, β . Mean is shown with the solid line; shaded region represents the 95% confidence interval.



(c) Non dimensionalized Reynolds stress.

(d) $y^+ \frac{dR_{12}}{dy^+}$

Figure 2: Time-scale discrepancy for $Re_\tau = 550$. The prior solution is represented in green, red represents the posterior and blue represents the DNS solution.

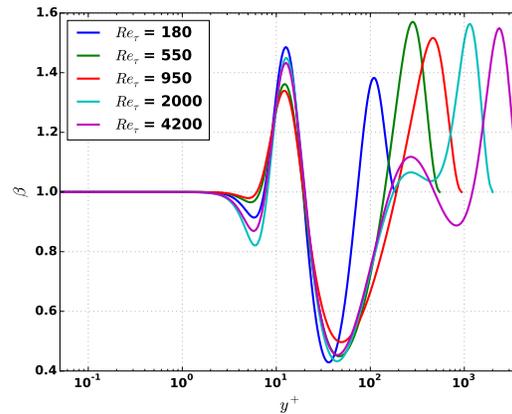


Figure 3: Time-scale discrepancy: The inferred correction function, for different Reynolds numbers.

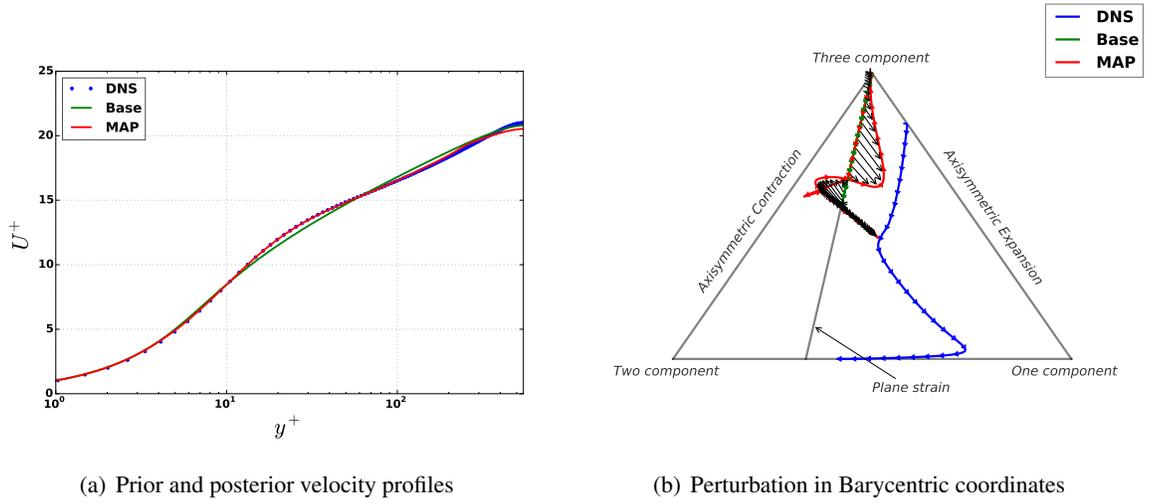


Figure 4: Anisotropy discrepancy for $Re_\tau = 550$. The prior solution is represented in green, red represents the posterior and blue represents the DNS solution.

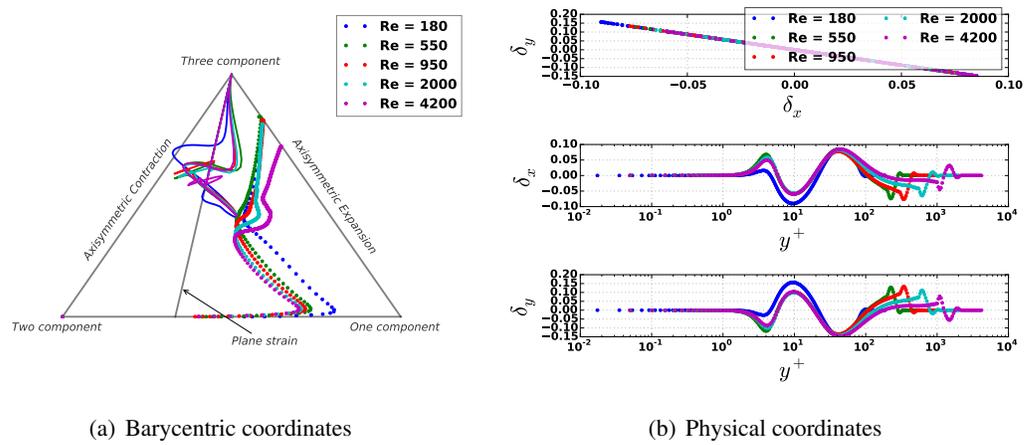


Figure 5: Comparison of the barycentric perturbations for different Reynolds numbers.

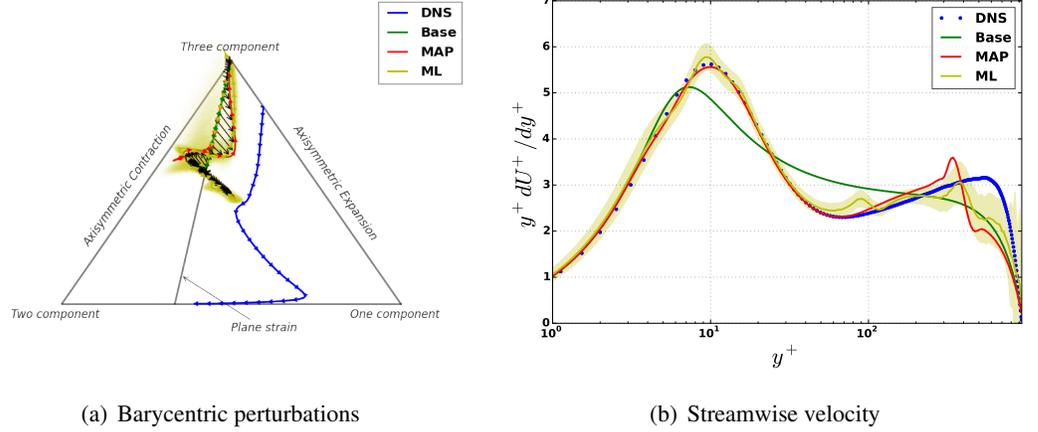


Figure 6: Comparison of the machine learning prediction. Shaded region represents the 95% interval and few representative samples are represented with solid lines.

were separated into two clusters $y^+ < 100$ and $y^+ > 100$ and the following features were used in the machine learning:

$$\delta_x = \begin{cases} F_{GP}(y^+, Sk/\varepsilon, y\sqrt{k}/\nu, u_y^2 k/\omega) & y^+ \leq 100 \\ F_{GP}(y, Sk/\varepsilon, y\sqrt{k}/\nu, u_y^2 k/\omega) & y^+ > 100 \end{cases} \quad (10)$$

$$\delta_y = -1.7321\delta_x \quad (11)$$

Once the GP has been constructed, it was embedded into a predictive model. To propagate the uncertainties in the embedded discrepancy functions, realizations were generated as follows:

$$\delta_x = \mathcal{N}(\delta_{x,GP}, \sigma_{\delta_{x,GP}}^2) \quad (12)$$

$$\delta_y = -1.7320\delta_x. \quad (13)$$

To assess the impact of variabilities in machine learning, an ensemble of realizations (from the posterior of the GP) was used to obtain predictions and predictive bounds as shown in Figure 6.

6 Inference in complex problems

In this section, the inversion procedure will be assessed in a wide range of problems involving flow separation, with the availability of different types of data. For purposes of demonstration, the model correction $\beta(\mathbf{x})$ is introduced as a multiplier of the production term $P(\tilde{\mathbf{v}}, \mathbf{U})$ in the Spalart-Allmaras model. This equation set is notionally represented by

$$\frac{D\tilde{\mathbf{v}}}{Dt} = \beta(\mathbf{x})P(\tilde{\mathbf{v}}, \mathbf{U}) - D(\tilde{\mathbf{v}}, \mathbf{U}) + T(\tilde{\mathbf{v}}, \mathbf{U}). \quad (14)$$

As mentioned in Ref. [21], it must be recognized that the introduction of $\beta(\mathbf{x})$ changes the entire balance of the model, (and need not be interpreted as merely a modification of the production term). It is equivalent to adding a source term $\delta(x) = (\beta(x) - 1)P(x)$. Inferring β , however, leads to a better

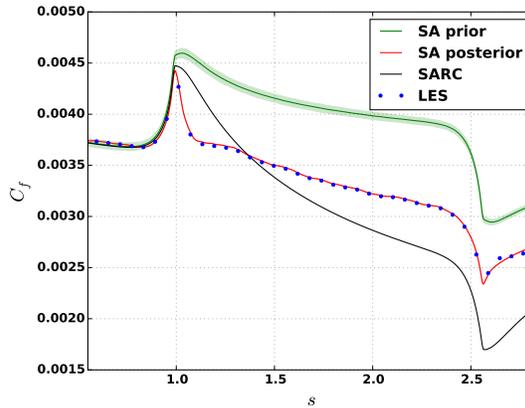


Figure 7: The prior, posterior, SARC and LES coefficient of friction (C_f) at the lower wall for the convex channel. Shaded region represents the 95% confidence interval.

conditioned inverse problem, as β is non-dimensional and has a simple initial value of unity. In these problems, a compressible RANS solver [35] is used along with a discrete adjoint solver. Additional information on the inversion procedure can be found in Ref. [21].

6.1 Boundary layer flow over a curved surface

The benchmark data for boundary layer flow over a convex curve is obtained from a Large Eddy Simulation [36, 37] with an inlet boundary layer with $Re_\theta = 2000$. A comparison of the inferred solution (using the skin friction coefficient C_f from the LES) with the analytically-sensitized rotational correction (SARC) model of Spalart [38] is presented.

Figure 7 shows the prior and the posterior C_f with 95% confidence intervals alongside the LES and SARC results for the convex channel. Figure 8 shows the inferred MAP correction term, δ_{MAP} , and the analytically-defined correction term from the corresponding SARC model. Qualitatively, the correction term is similar for both models, but the magnitudes are locally different. The trend in the correction is consistent with the expectation that the convex curvature reduces the turbulence intensity. The quantitative nature of the results should provide valuable information to the modeler. Figure 9 shows the variation of the streamwise velocity with respect to the distance from the wall at various streamwise locations. The posterior velocity is seen to correlate well with the LES solution compared to the prior and the SARC prediction. The results suggest that the SARC model requires improvements in the log layer.

6.2 Boundary layer separation over a wall-mounted hump

To assess the generality of the inverse modeling and its impact on other field quantities, the inversion was applied to a NASA benchmark test [39] (Fig. 10), which involves separated flow over a smooth hump. In this problem, $\beta(x, y)$ was inferred with the objective of matching the wall pressure distribution in the region $0.5 \leq x/c \leq 1.5$. Even though the objective function was only taken to be the measured surface pressure, improvement is seen in the Reynolds stress predictions (Fig. 10c). As a consequence of the overall improvement in the field solution, the predicted length of the separation bubble was found

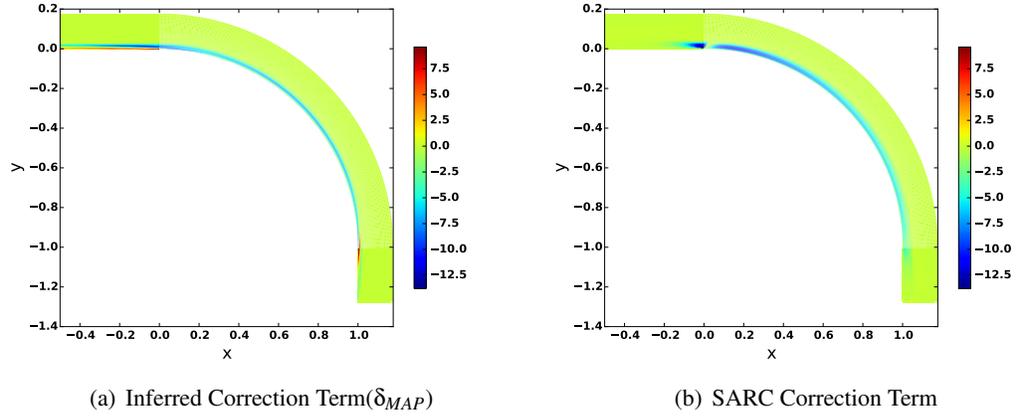


Figure 8: Contour plot showing the inferred correction term (δ_{MAP}) and the SARC correction term for the convex channel.

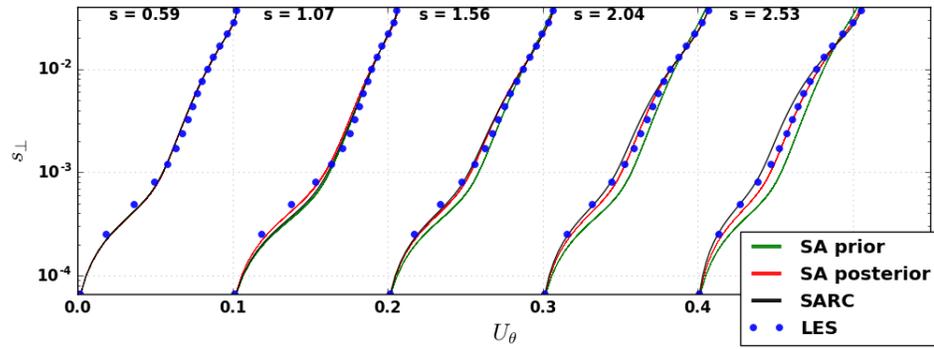


Figure 9: The prior, posterior, SARC and LES stream-wise velocity at various locations for the convex channel. s_{\perp} refers to the perpendicular distance from the lower wall. Refer Fig. 7 for the legend

be 15% more accurate compared to the baseline solution. Results can be improved by considering more information from the experiment [21], but this exercise offers evidence that for separated flows, pressure data can be valuable in inferring model discrepancy.

6.3 Separated flows over airfoils

In this example, high angle of attack turbulent flow over the S809 airfoil [40] is considered. and the flow domain over airfoils is discretized using a C-grid with 291 points in the wraparound direction and 111 points in the wall-normal direction.

To determine the impact of the type of data on the inference, two different inverse problems were solved with respect to the following data sets:

- a) Measured surface pressure coefficients (C_p)
- b) Measured lift coefficient (C_l)

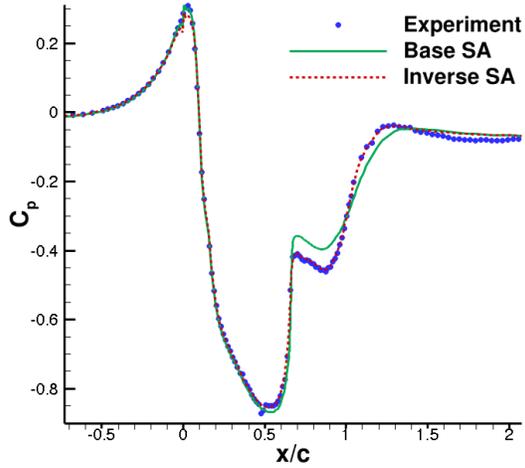
In both of the above inverse problems, the prior solution is assumed to be the baseline SA model and the ratio of the observational covariance to the prior covariance was set to $\lambda = 4 \times 10^{-4}$. The two objective functions were confirmed to lead to a similar solution to the inverse problem (Fig. 11). While there are discrepancies in the post-stall region, the near-wall features in $\beta(x)$ are almost identical, resulting in indistinguishable surface pressures. **This can have significant implications for modeling because many airfoil data sets only provide lift and drag measurements.** This implies a much higher level of confidence in the experimentally measured lift compared to the variability of β . Further, the optimal solution was also confirmed to be relatively insensitive to order of magnitude variations in λ .

7 Machine learning and predictions in complex problems

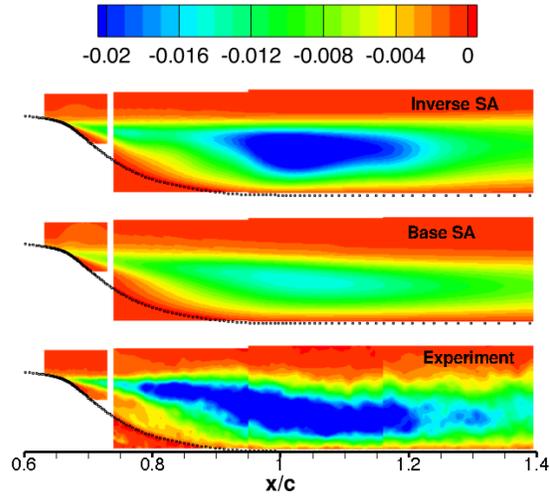
The inverse approach presented in the previous section results in an optimal correction field for a given flow condition and geometry. To be useful in *predictive modeling*, a functional relationship must be developed by considering the output of a number of inverse problems representative of the modeling deficiencies relevant to the predictive problem. Further, as explained below, elements of the feature vector η are chosen to be locally non-dimensional quantities such that the functional relationship $\beta(\eta)$ is useful for different problems in which the η variables are realizable.

7.1 Feature selection

To build a set of features η upon which the functional relationship $\beta(\eta)$ will be based, a logical place to start would be to identify the independent variables in the baseline SA model. The source terms in the SA model are a function of four local flow quantities, v , \hat{v} , Ω , d , which represent the kinematic viscosity, the SA working variable, the vorticity magnitude, and the distance from the wall, respectively. As discussed in Ref. [19], these quantities do not constitute an appropriate choice for the input feature vector to the machine learning algorithm. They are dimensional quantities which may have different numeric values even when two flows are dynamically similar. Thus, the inputs are re-scaled [19] by relevant *local quantities* that are representative of the state of turbulence. An obvious locally non-dimensional quantity in the baseline SA model is $\chi = \hat{v}/v$. We define local scales, $v + \hat{v}$ and d , and

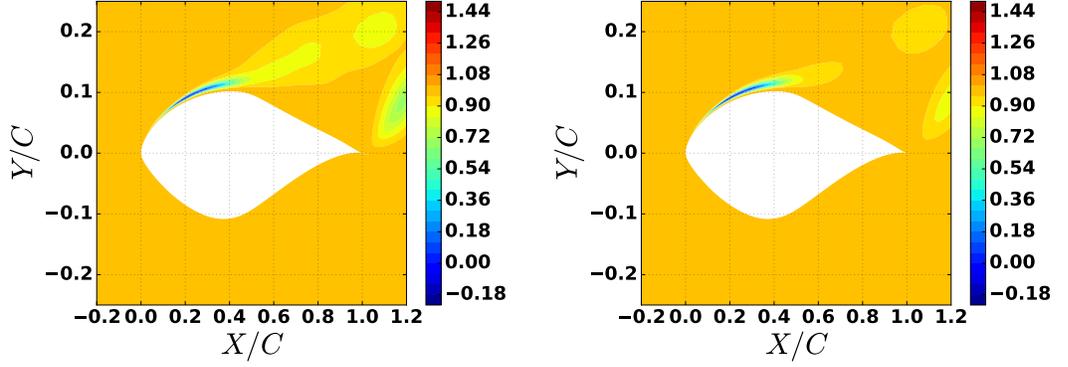


(a) Wall pressure coefficient



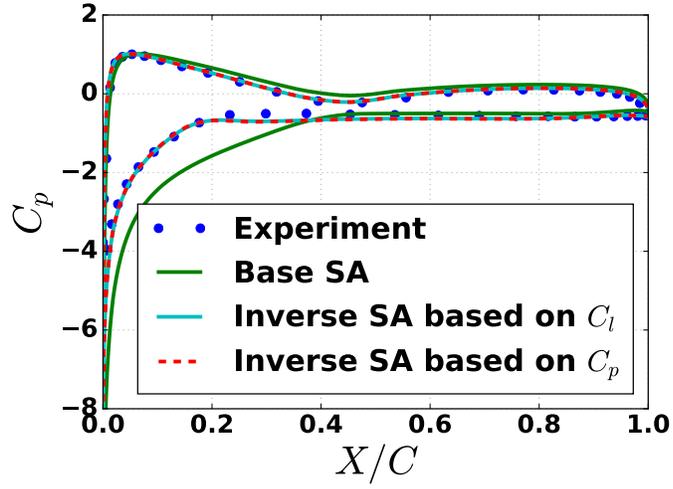
(b) Reynolds shear stress contours $\overline{u'_1 u'_2} / U_\infty^2$

Figure 10: Application of inverse modeling to separated flow over a smooth surface.



(a) $\beta(x)$ field using objective function based on C_l

(b) $\beta(x)$ field using objective function based on C_p



(c) C_p

Figure 11: Inverse solutions using objective function based on lift (C_l) and surface pressure (C_p) coefficients. $\beta(x)$ in the near wall region is unaffected by the choice of objective function resulting in identical inverse C_p .

introduce an additional variable,

$$\bar{\Omega} = \frac{d^2}{\hat{v} + v} \Omega . \quad (15)$$

It can be shown [19, 24] that the locally non-dimensionalized source terms in the baseline SA model are dependent only on $\bar{\Omega}$ and χ .

The set of features that were evaluated includes $\{\bar{\Omega}, \chi, S/\Omega, \tau/\tau_{wall}, P/D\}$, where S, τ, τ_{wall} represent the strain-rate magnitude, magnitude of the Reynolds stress, and the wall shear stress, respectively.

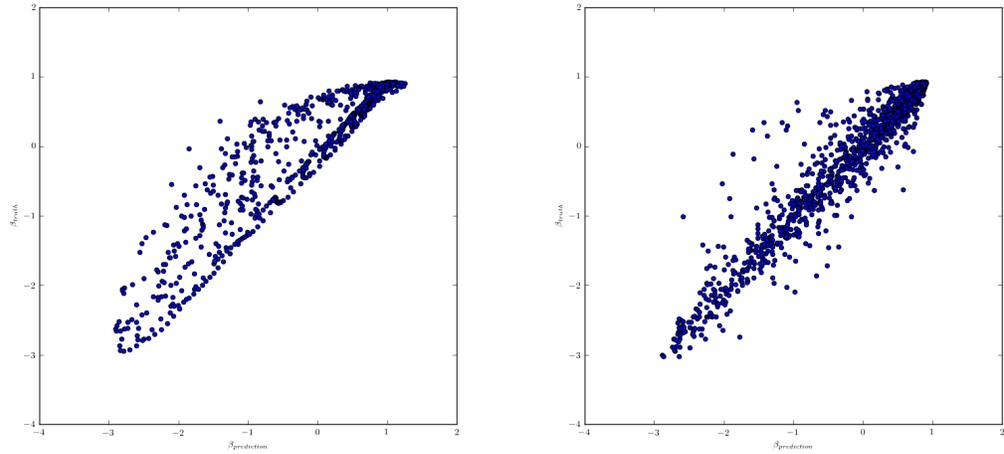
7.2 Learning

The lift-based inferred correction fields were obtained for the S814 airfoil at different combinations of angles of attack and $Re = 1 \times 10^6$ and 2×10^6 . The inversion is followed by employing two different machine learning techniques to reconstruct model corrections:

a) Neural networks (NN): The standard NN algorithm [41] operates by constructing linear combinations of inputs and transforming them through nonlinear activation functions. The process is repeated once for each hidden layer in the network, until the output layer is reached. Tensorflow [42], which utilizes standard back-propagation algorithms [41] is used in this work.

b) Adaptive boosting (Adaboost): Adaboost is an ensemble learning model originally developed by Freund et al. [43]. In this work, we used the enhanced version by Drucker [44] which is implemented in scikit-learn [45]. The basic idea of adaptive boosting is to iteratively use weak learners which iteratively focuses on the data that was predicted poorly in the previous iteration.

Sample evaluations are presented in Figure 12 on a selection of data points which were randomly left out from the training set. For this sample, the error metric (coefficients of determination - R^2) was found to be 0.89 and 0.94, respectively for the neural network and adaboost.



(a) Neural Network

(b) Adaptive Boosting

Figure 12: Results of Machine learning evaluations using two machine learning algorithms on S814 inversion data.

7.3 Prediction

As schematized in Fig. 1, the mapping $\beta(\eta)$ built during the training process is queried for input features $\hat{\eta}$ at every iteration of the flow solver to obtain outputs $\hat{\beta}$ which are embedded into the predictive model. This process is repeated until convergence. Again, it is emphasized that the training data involved only the S814 lift data $Re = 1 \times 10^6, 2 \times 10^6$.

The model performs equally well for airfoil shapes (S805 and S809) and Reynolds numbers ($Re = 3 \times 10^6$) not used in the training set as seen from Figs. 13, 14. Clearly, significant improvement in

stall prediction is evident in the lift prediction. As a consequence, the drag rise is predicted to occur at lower angles of attack than in the baseline model, a trend that is qualitatively correct. Further, there is no evidence of deterioration of accuracy in the low angle of attack regions, where the original model is already accurate. The improvement in the quality of the predictions is further emphasized in Figs. 15, 16. These results confirm that the data-augmented model offers considerable predictive improvements in surface pressure distributions.

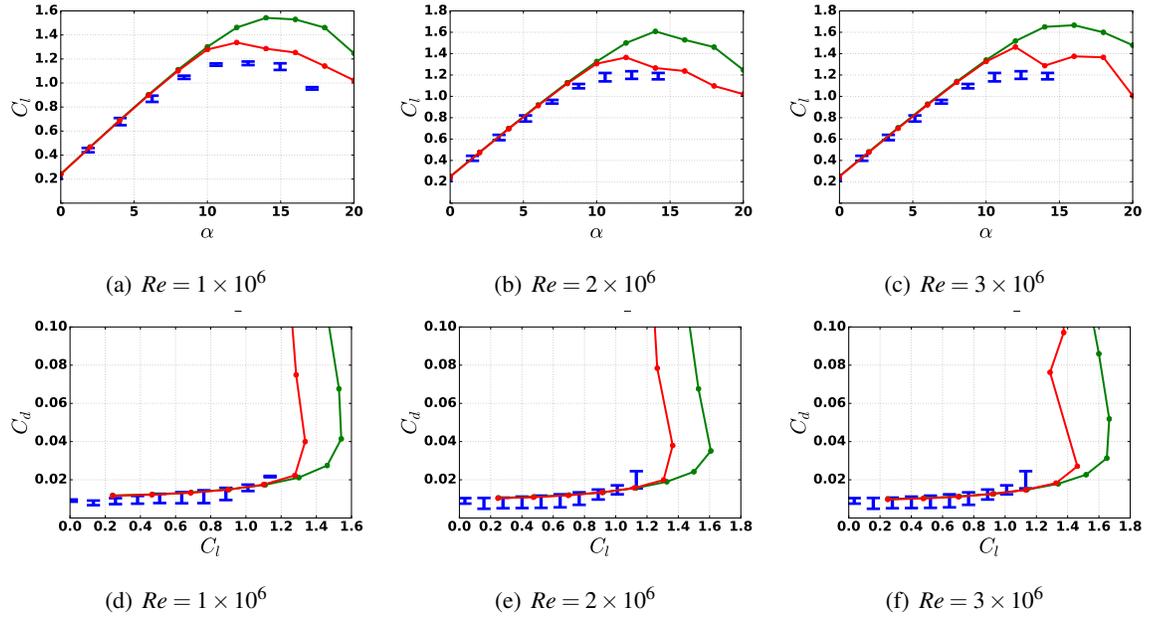


Figure 13: NN-augmented SA prediction for S805 airfoil using data-set P. — Experiment, — base SA and — neural network.

8 Conclusions

The abundance of data from high-fidelity simulations and high resolution experiments provides unprecedented opportunities to comprehensively inform closure models. A paradigm of data-driven model development comprising of full-field inversion and machine learning (FIML) was described. The inference process generates function correction information for *specific* problems. Once the inference is applied over a number of problems, machine learning is used to reconstruct the inferred function in terms of variables that will be available during predictive simulations. The reconstructed function is then embedded into a predictive solver, which queries the machine learned model at every iteration of the flow solver to obtain model corrections. The field inversion process directly provides comprehensive information about model discrepancies, which is of great use to the modeler in the quest to formulate more accurate closures. The machine learning step could be considered as *one* tool that can be used to reconstruct the discrepancy.

A tutorial was presented - in the context of 1D channel flow - on the use of FIML in model development with the goal of addressing two different types of model discrepancy - in time scales and stress anisotropy.

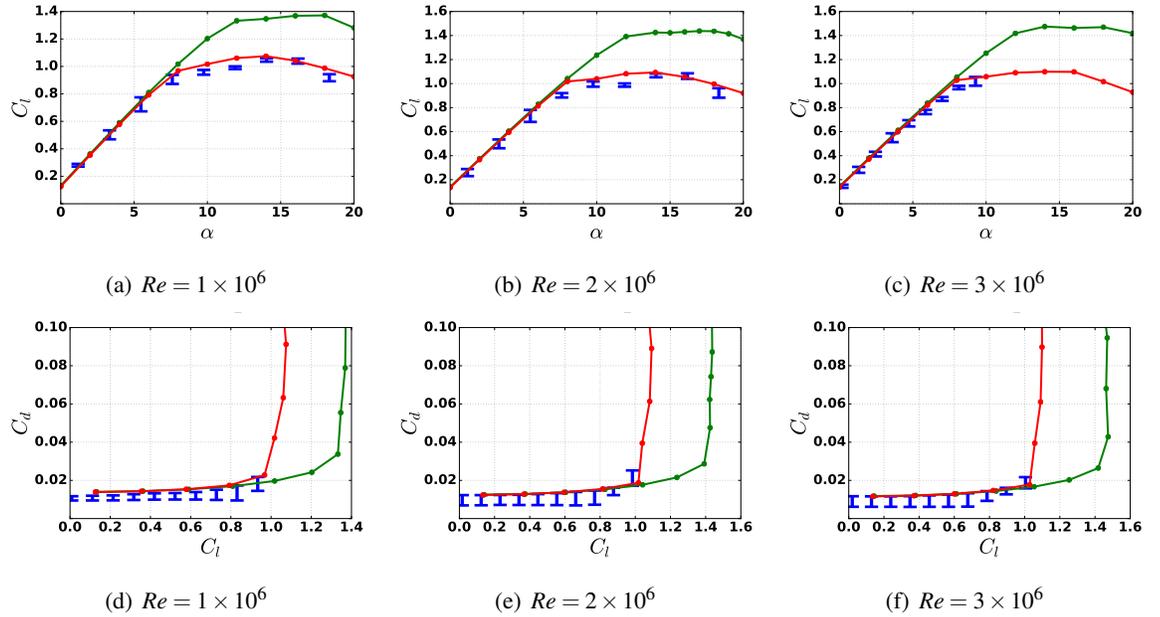


Figure 14: NN-augmented SA prediction for S809 airfoil using data-set P. — Experiment, — base SA and — neural network.

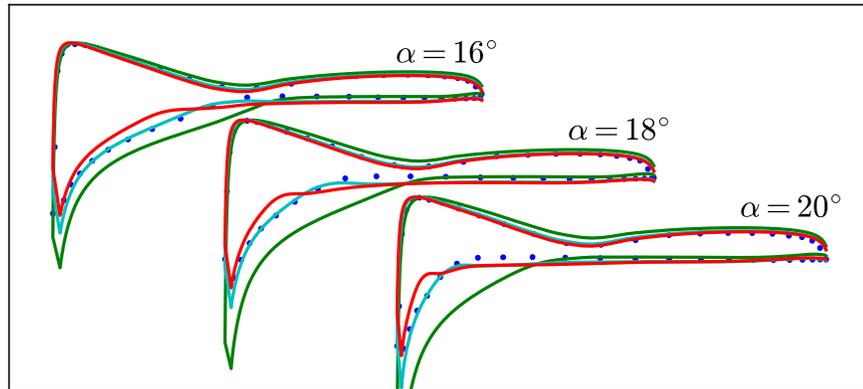


Figure 15: Surface pressure coefficient for S809 airfoil at $Re = 2 \times 10^6$ and $\alpha = \{16^\circ, 18^\circ, 20^\circ\}$. Refer Fig. 11(c) for legend.

In more complicated problems involving flow separation, the inference process was shown to assimilate sparse data while improving the solution over the entire computational domain. This reinforces confidence that the procedure does not overfit the model to the data and that predictive improvements can be realized for the right reasons.

In the case of turbulent flow separation over airfoils, the data-assisted model showed significant improvement over the baseline model in predicting lift and drag coefficients and stall onset angles and the model predictions were confirmed to be excellent in airfoil shapes and flow conditions that were not

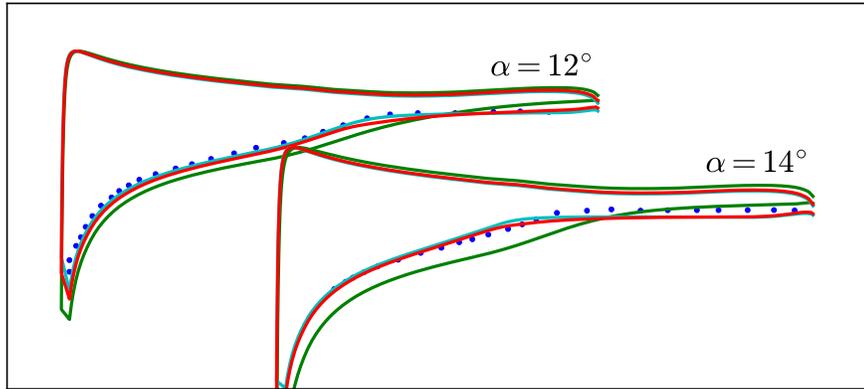


Figure 16: Surface pressure coefficient for S805 airfoil at $Re = 1 \times 10^6$ and $\alpha = \{12^\circ, 14^\circ\}$. Refer Fig. 11(c) for legend. Experimental pressure is shown only for the upper surface.

part of the training set.

The philosophy and formalisms employed in this work are of a general nature and are not restricted to the type of model or the type of model discrepancy that is addressed. In such endeavors, it would be critical to ensure that the model augmentations a) can be developed based on sparse experimental data as well as rich DNS/LES data b) do not influence regions of the flow that are adequately represented by the baseline model (near-wall region in thin boundary layers), and c) do not degrade the convergence properties of the solver. There is much to be gained by carefully exploring a broader set of input features [16, 23] and alternative machine learning methods and approaches [15, 34]. Finally, realizability limits [10, 23] and invariance properties [46] should be respected to constrain the model, especially when the model is operating in an extrapolatory mode.

For the framework to be able to offer improved predictions in practical situations, inverse problems must be solved over a wide class of problems (and over multiple objective functions of interest) that will be representative of the deficient physics in the baseline model. Concurrently, the tendency of the learning process to over-fit data must also be avoided. At every stage of process, the underlying physical insight is irreplaceable and thus it is left to the modeler to make judicious choices about the data, prior information and introduction of one or more correction functions.

9 Symposium on Advances in Turbulence Modeling

The idea of data-driven turbulence modeling which was explored with NASA LEARN funding from 2014-2017 has been received well by the community and several groups around the world are pursuing the idea. With the goal of disseminating results to the broader community and to place the role of data-driven modeling within the context of mainstream turbulence modeling, a three-day Turbulence Modeling Symposium sponsored by the University of Michigan and NASA LEARN was held in Ann Arbor, Michigan in July 2017. This meeting brought together nearly 90 experts from academia, government and industry, with good international participation. Emphasis was placed on turbulence modeling in a predictive context in complex problems, rather than on turbulence theory or descriptive modeling.

A report [47] was published and slides from all talks are available from the symposium website ⁷.

The three-day symposium included 32 participant talks, divided into the following broad areas: new ideas for turbulence modeling, Reynolds stress transport modeling, uncertainty quantification, experiments, data-driven methods, flow solution technologies of the future, LES, direct numerical simulations (DNS), and applications. There were also 6 keynote/plenary invited talks. The symposium was designed to encourage discussion and to be *speculative*. There was well over 8 hours of discussion time in the symposium that often led to very open/candid exchanges of views. Total attendance was 88, with 53 from academia, 20 from national labs and 15 from industry. There was representation from outside of the United States, including 8 participants from Europe, and representation from experimentalists (6 talks). There was also a blend of providers and users of turbulence modeling. Overall, four goals were accomplished:

- In general, a big positive from the symposium was to get a number of turbulence modeling experts and users in one room together for three days of focused interactions, which reflected much goodwill. Although there was an attempt to define some common goals/focus, this proved to be a challenging task, in part because stake-holders had very diverse needs from CFD.
- One key idea of the symposium was to try to get all of the participants to include in their thinking about RANS and to critique the concept of an *ultimate barrier* (at the workshop, the expression “glass ceiling” was used). Here, *ultimate barrier* refers to an unseen and not exactly defined, yet unbreachable barrier. If such a barrier indeed exists, then progress beyond it will be impossible (or at least highly unlikely), despite the ability to “see” the desired goal and even to define it.
- An important goal of the symposium was to discuss some of the more recent topics to emerge in the field of turbulence modeling, primarily uncertainty quantification (UQ) and data-driven modeling. The purpose was two-fold: (1) to educate the participants about the basics behind these topics, and (2) to try to place these topics in the context of mainstream turbulence modeling, as it has been practiced since the middle of the last century.
- A set of recommendations were formulated and shared with the community.

References

- [1] Yarlanki, S., Rajendran, B., and Hamann, H., “Estimation of turbulence closure coefficients for data centers using machine learning algorithms,” *Thermal and Thermomechanical Phenomena in Electronic Systems (ITherm)*, 2012 13th IEEE Intersociety Conference on, 2012, pp. 38–42.
- [2] Kato, H. and Obayashi, S., “Data Assimilation for Turbulent Flows,” *16th AIAA Non-Deterministic Approaches Conference, AIAA SciTech*, (AIAA 2014-1177), Jan 2014.
- [3] Edeling, W., Cinnella, P., Dwight, R. P., and Bijl, H., “Bayesian estimates of parameter variability in the k- ϵ turbulence model,” *Journal of Computational Physics*, Vol. 258, 2014, pp. 73–94.
- [4] Ray, J., Lefantzi, S., Arunajatesan, S., and DeChant, L. J., “Bayesian Calibration of a RANS Model with a Complex Response Surface - A Case Study with Jet-in-Crossflow Configuration,” *45th AIAA Fluid Dynamics Conference Dallas, TX*, Jun 2015.

⁷<http://turbgate.engin.umich.edu/symposium/agenda.html>.

- [5] Cheung, S. H., Oliver, T. A., Prudencio, E. E., Prudhomme, S., and Moser, R. D., “Bayesian uncertainty analysis with applications to turbulence modeling,” *Reliability Engineering & System Safety*, Vol. 96, No. 9, 2011, pp. 1137–1149.
- [6] Oliver, T. A. and Moser, R. D., “Bayesian uncertainty quantification applied to RANS turbulence models,” *Journal of Physics: Conference Series*, Vol. 318, No. 4, 2011, pp. 042032.
- [7] Cheung, S. H. and Beck, J., “New Bayesian updating methodology for model validation and robust predictions based on data from hierarchical subsystem tests,” *Earthquake Engineering Research Laboratory, Caltech*, , No. CaltechEERL:EERL-2008-04, Jan 2009.
- [8] Dow, E. and Wang, Q., “Uncertainty Quantification of Structural Uncertainties in RANS Simulations of Complex Flows,” *20th AIAA Computational Fluid Dynamics Conference Honolulu, Hawaii*, Jun 2011.
- [9] Dow, E. and Wang, Q., “Quantification of Structural Uncertainties in the $k-\omega$ Turbulence Model,” *52nd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference, Denver, Colorado*, Apr 2011.
- [10] Emory, M., Pecnik, R., and Iaccarino, G., “Modeling Structural Uncertainties in Reynolds-Averaged Computations of Shock/Boundary Layer Interactions,” *49th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition Orlando, Florida*, Jan 2011.
- [11] Gorle, C., Emory, M., and Iaccarino, G., “RANS modeling of turbulent mixing for a jet in supersonic cross flow: model evaluation and uncertainty quantification,” *Proceedings of the Seventh International Symposium on Turbulence, Heat and Mass Transfer (ICHMT Digital Library Online, Palermo, Italy, 2012)*, 2012.
- [12] Emory, M., Larsson, J., and Iaccarino, G., “Modeling of structural uncertainties in Reynolds-averaged Navier-Stokes closures,” *Physics of Fluids*, Vol. 25, No. 11, 2013, pp. 110822.
- [13] Tracey, B., Duraisamy, K., and Alonso, J., “Application of Supervised Learning to Quantify Uncertainties in Turbulence and Combustion Modeling,” *51st AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition Grapevine (Dallas/Ft. Worth Region), Texas*, Jan 2013.
- [14] Xiao, H., Wu, J.-L., Wang, J.-X., Sun, R., and Roy, C., “Quantifying and Reducing Model-Form Uncertainties in Reynolds-Averaged Navier-Stokes Equations: An Open-Box, Physics-Based, Bayesian Approach,” *arXiv preprint arXiv:1508.06315*, 2015.
- [15] Weatheritt, J., *The development of data driven approaches to further turbulence closures*, Ph.D. thesis, University of Southampton, 2015.
- [16] Ling, J. and Templeton, J., “Evaluation of machine learning algorithms for prediction of regions of high Reynolds averaged Navier Stokes uncertainty,” *Physics of Fluids*, Vol. 27, No. 8, 2015, pp. 085103.

- [17] King, R., Hamlington, P. E., and Dahm, W., “Autonomic Subgrid-Scale Closure for Large Eddy Simulations,” *53rd AIAA Aerospace Sciences Meeting, AIAA SciTech, Kissimmee, Florida*, Jan 2015.
- [18] Duraisamy, K., Zhang, Z. J., and Singh, A. P., “New Approaches in Turbulence and Transition Modeling Using Data-driven Techniques,” *53rd AIAA Aerospace Sciences Meeting, AIAA SciTech, Kissimmee, Florida*, Jan 2015.
- [19] Tracey, B. D., Duraisamy, K., and Alonso, J. J., “A Machine Learning Strategy to Assist Turbulence Model Development,” *53rd AIAA Aerospace Sciences Meeting, AIAA SciTech, Kissimmee, Florida*, Jan 2015.
- [20] Parish, E. J. and Duraisamy, K., “A paradigm for data-driven predictive modeling using field inversion and machine learning,” *J. Comput. Physics*, Vol. 305, 2016, pp. 758–774.
- [21] Singh, A. P. and Duraisamy, K., “Using field inversion to quantify functional errors in turbulence closures,” *Physics of Fluids*, Vol. 28, No. 4, 2016, pp. 045110.
- [22] Ling, J., Kurzawski, A., and Templeton, J., “Reynolds averaged turbulence modelling using deep neural networks with embedded invariance,” *Journal of Fluid Mechanics*, Vol. 807, 2016, pp. 155–166.
- [23] Wang, J.-X., Wu, J.-L., and Xiao, H., “Physics-Informed Machine Learning for Predictive Turbulence Modeling: Using Data to Improve RANS Modeled Reynolds Stresses,” *arXiv preprint arXiv:1606.07987*, 2016.
- [24] Singh, A. P., Medida, S., and Duraisamy, K., “Machine Learning-augmented Predictive Modeling of Turbulent Separated Flows over Airfoils,” *AIAA Journal*, Vol. 55, No. 7, 2017.
- [25] Poroseva, S. and Murman, S. M., “Velocity/Pressure-Gradient Correlations in a FORANS Approach to Turbulence Modeling,” *44th AIAA Fluid Dynamics Conference, AIAA Aviation, (AIAA 2014-2207)*, Jun 2014.
- [26] Hoyas, S. and Jiménez, J., “Scaling of the velocity fluctuations in turbulent channels up to $Re_\tau = 2003$,” *Physics of Fluids (1994-present)*, Vol. 18, No. 1, 2006, pp. 011702.
- [27] Lee, M. and Moser, R. D., “Direct numerical simulation of turbulent channel flow up to $Re_\tau = 5200$,” *arXiv preprint arXiv:1410.7809*, 2014.
- [28] del Alamo, J. C. and Jimenez, J., “Direct numerical simulation of the very large anisotropic scales in a turbulent channel,” *arXiv preprint arXiv:1309.2322*, 2013.
- [29] Wilcox, D. C., *Turbulence modeling for CFD*, Vol. 3, D C W Industries, 2006.
- [30] Singh, A. P. and Duraisamy, K., “Using field inversion to quantify functional errors in turbulence closures,” *Physics of Fluids (1994-present)*, Vol. 28, No. 4, 2016, pp. 045110.
- [31] Parish, E. J. and Duraisamy, K., “A paradigm for data-driven predictive modeling using field inversion and machine learning,” *J. Comput. Physics*, Vol. 305, 2016, pp. 758–774.

- [32] Emory, M., Larsson, J., and Iaccarino, G., “Modeling of structural uncertainties in Reynolds-averaged Navier-Stokes closures,” *Physics of Fluids (1994-present)*, Vol. 25, No. 11, 2013, pp. 110822.
- [33] Aster, R., *Parameter Estimation and Inverse Problems*, Elsevier Academic Press, 2005.
- [34] Zhang, Z., Duraisamy, K., and Gumerov, N. A., “Efficient Multiscale Gaussian Process Regression using Hierarchical Clustering,” *arXiv preprint arXiv:1511.02258*, 2015.
- [35] Duraisamy, K., McCroskey, W. J., and Baeder, J. D., “Analysis of wind tunnel wall interference effects on subsonic unsteady airfoil flows,” *Journal of Aircraft*, Vol. 44, No. 5, 2007, pp. 1683–1690.
- [36] Arolla, S. K. and Durbin, P. A., “LES of spatially developing turbulent boundary layer over a concave surface,” *Journal of Turbulence*, Vol. 16, No. 1, 2015, pp. 81–99.
- [37] Durbin, P., Personal communication, Iowa State University.
- [38] Shur, M. L., Strelets, M. K., Travin, A. K., and Spalart, P. R., “Turbulence modeling in rotating and curved channels: Assessing the Spalart-Shur correction,” *AIAA journal*, Vol. 38, No. 5, 2000, pp. 784–792.
- [39] Naughton, J. W., Viken, S. A., and Greenblatt, D., “Skin-Friction Measurements on the NASA Hump Model,” *AIAA Journal*, Vol. 44, No. 6, 2016, pp. 1255–1265.
- [40] Somers, D. M., “Design and experimental results for the S809 Airfoil,” *NREL Report, NREL/SR-440-6918*, 1997.
- [41] Bishop, C. M., *Pattern recognition and machine learning*, Springer-Verlag New York, 1st ed., 2006.
- [42] Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., et al., “Tensorflow: Large-scale machine learning on heterogeneous distributed systems,” *arXiv preprint arXiv:1603.04467*, 2016.
- [43] Freund, Y. and Schapire, R. E., “A decision-theoretic generalization of on-line learning and an application to boosting,” *European conference on computational learning theory*, 1995, pp. 23–37.
- [44] Drucker, H., “Improving regressors using boosting techniques,” *ICML*, Vol. 97, 1997, pp. 107–115.
- [45] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al., “Scikit-learn: Machine learning in Python,” *Journal of Machine Learning Research*, Vol. 12, No. Oct, 2011, pp. 2825–2830.
- [46] Ling, J., Jones, R., and Templeton, J., “Machine learning strategies for systems with invariance properties,” *Journal of Computational Physics*, Vol. 318, 2016, pp. 22–35.
- [47] Duraisamy, K., Spalart, P. R., and Rumsey, C. L., “Status, Emerging Ideas and Future Directions of Turbulence Modeling Research in Aeronautics,” 2017.