



# New Approaches in Turbulence and Transition Modeling Using Data-driven Techniques

Karthik Duraisamy\* Ze Jia Zhang<sup>†</sup> Anand Pratap Singh<sup>‡</sup>

Department of Aerospace Engineering, University of Michigan, Ann Arbor, MI 48109.

**A data-driven approach to the modeling of turbulent and transitional flows is proposed in this work, with the goal of developing more robust and accurate closure models. The key idea is to (i) infer the functional form of deficiencies in known closure models by applying inverse problems to computational and experimental data, (ii) use machine learning to reconstruct the improved functional forms, and (iii) to inject the improved functional forms in simulations to obtain more accurate predictions. The inverse modeling step, on its own, can yield valuable insight to the modeler, essentially converting data to information. The machine learning step is a tool to convert information into modeling knowledge. Representative examples are used to describe the methodology and to demonstrate its viability. The first example investigates the modeling of a non-equilibrium turbulent boundary layer, and the second involves the modeling of bypass transition to turbulence. Evidence from these problems emphasizes the utility of the proposed approach in offering new routes to closure modeling in general computational physics disciplines.**

## I. Introduction

Accurate simulation of transitional and turbulent flows is of critical importance to many applications in science and engineering. The onset of turbulence is considered undesirable in many situations such as those in turbo-machinery, atmospheric re-entry vehicles, commercial aircraft, *etc.* due to consequences such as increased losses, aerodynamic heating, and decreased fuel efficiency. In contrast, turbulence plays a beneficial role in settings such as mixing and combustion of reacting gases. Despite the tremendous growth in computational resources over the past decade, modeling and simulation of many practical/realistic turbulent flows – to the desired level of accuracy – has remained challenging, and in some cases, even inaccessible. Though Direct Numerical Simulations (DNS) and Large Eddy Simulations (LES) have offered tremendous insight and predictive capabilities in many flows, these techniques continue to be infeasible for high Reynolds number wall-bounded flows. This situation is unlikely to change within the next few decades unless significant advances are made in hybrid techniques that employ a near-wall model in conjunction with an outer-layer LES. Thus, near-wall models in both an LES and in a Reynolds-averaged Navier–Stokes (RANS) context will be the pacing item in applied computational fluid dynamics (CFD). This assertion is also supported by the findings of a recent NASA study.<sup>1</sup>

Popular turbulence (and transition) models are all drastic simplifications to the rich dynamics of turbulence. Turbulence closure equations usually introduce between one and seven additional transport variables and many adjustable constants selected by the engineering judgment of the modeler. These constants are calibrated by a small number of simple test cases such as homogeneous turbulence and thin-shear flows, and given this development process, it is unsurprising that accuracy diminishes as the model is applied to problems which deviate from the calibrated cases. Typical examples in which RANS models are deficient involve adverse pressure gradients, inhomogeneous flow directions, secondary effects and flow separation. Practical turbulence models have been simple out of expediency. However, in attempts to increase accuracy, some work has shifted toward greater complication in turbulence models. As an example, Refs. 2,3 introduce additional tensorial bases to account for inhomogeneity, wall echo, and anisotropies in second moment closures. While these efforts are encouraging, the potential benefits are obscured by the need to determine a number of free parameters from a small set of often idealized test cases. New strategies are required to move beyond these limitations and we believe that data-driven approaches are capable of providing solutions.

\*Assistant Professor

<sup>†</sup>Graduate Research Assistant

<sup>‡</sup>Graduate Research Assistant

Data science is on the rise in many disciplines due to improvements in computational power and the increased availability of large data sets. This has been accompanied by significant improvements in data analytics and machine learning (ML) techniques, both in effectiveness and scalability. Various ML techniques are widely used today in financial and commercial applications such as stock trading, fraud detection, preference choices, etc. and scientific applications such as genomics, astrophysics, fluid mechanics, and natural language processing. Depending on the application, the objectives of the tasks can be a combination of automated clustering and classification, feature extraction, predictive modeling and improved decision making. Specifically in the area of turbulent flows, previous efforts have used neural networks for near-wall modeling through reconstruction of structures in a fully-developed turbulent channel flow,<sup>4</sup> real-time extraction of coherent spatio-temporal structures,<sup>5</sup> and optimization of closure coefficients of the two-equation k- $\epsilon$  turbulence model,<sup>6</sup> *etc.* Other attempts (Refs. 7,8) have approached the problem from the viewpoint of structural uncertainty quantification or Bayesian model averaging.<sup>9,10</sup>

Very recently, data-driven statistical inference to correct for model error (Edeling et al.<sup>11</sup>) has been proposed to address some of the deficiencies of a priori processing. This, and other approaches have focused on estimating the parameters of the standard models to calibrate a set of pre-specified building-block functions. The uniqueness of our approach stems from the focus on inferring and reconstructing deficiencies in the **functional form** of known turbulence models, rather than on the model parameters. Further, in contrast to data-driven *descriptive* modeling approaches, data-driven *predictive* modeling involves additional challenges. The enabling tools in our approach are inverse modeling and machine learning.

The broader goal of the present effort is to develop the formalism and tools to infuse closure models of transition, turbulence and other fields of mechanics with data-driven aspects. In this pursuit, we do not seek to replace decades of modeling knowledge, but rather to supplement it.

The rest of the paper is organized as follows: Section II introduces the application of inverse modeling to infer the functional form of modeling deficiencies in two sample problems in turbulence and transition modeling. Section III introduces the application of Neural networks and Gaussian process regression methods to reconstruct the inferred functional forms in Section II. Injection of the reconstructed data in simulations is addressed in Section IV, after which, a brief summary is provided. In a companion paper (Tracey et al.<sup>12</sup>), more specific details related to the machine learning step of the framework are addressed.

## II. Inverse Modeling

In principle, the proposed techniques can be applied to many computational physics problems, but we will restrict the discussion to application in turbulence and transition models. Further, within the field of turbulence/transition modeling, the proposed techniques can be adapted to sub-grid/wall-modeling in LES or to the end of improving second moment RANS closures. The focus of this work, however, is to introduce the general ideas and highlight the promise of our approach, and hence the demonstrations will be restricted to eddy-viscosity-based closures for transition and turbulence.

### A. Application to a turbulence modeling problem

The closure model that will be considered for the study is the Spalart-Allmaras turbulence model.<sup>14</sup> In this model, a transport equation for a surrogate variable  $\tilde{\nu}$  of the turbulent eddy viscosity  $\nu_t$  is employed. The eddy-viscosity surrogate  $\tilde{\nu}$  is transported according to

$$\frac{D\tilde{\nu}}{Dt} = P(\tilde{\nu}) - D(\tilde{\nu}) + T(\tilde{\nu}), \quad (1)$$

where the production  $P$ , the destruction  $D$  and the diffusion  $T$  terms are non-linear terms that are modeled empirically. The above equation is used with a non-linear functional relationship to derive  $\nu_t$  from  $\tilde{\nu}$ , which is then used in a Boussinesq formulation to compute the Reynolds stresses. Extracting  $\tilde{\nu}$ , say from DNS or LES data is of little use without knowing how it will be used in a model. Further, since these equations contain no explicit physical or data parameters, not much benefit can be derived from parameter estimation. The issue at hand, rather, is that **the functional forms of the terms in Eq. 1 are themselves inaccurate**, and in some cases, substantially so.

The chosen application involves the modeling of a non-equilibrium turbulent boundary layer. Specifically, the evolution of a boundary layer over a convex wall is studied, with the computational domain shown in Fig. 1. The geometry corresponds to that used in the experimental measurements of Webster, Degraaf and Eaton.<sup>16</sup> In this problem, the top and bottom boundaries are treated as viscous walls. The left boundary is an equilibrium turbulent boundary layer at a momentum thickness Reynolds number  $Re_\theta = 12170$ , while the right boundary is a subsonic characteristic

outflow. We compare the outputs of our model to the wall-modeled LES (WMLES) computed by Radhakrishnan and Piomelli.<sup>17</sup>

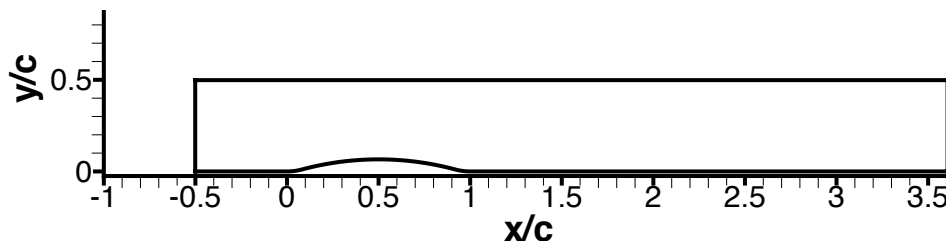


Figure 1. Computational Domain.

As a demonstrative example of the proposed approach, a spatio-temporal function  $\alpha(\mathbf{x}, t)$  is introduced into Eq. 1 in the following form:

$$\frac{D\tilde{\mathbf{v}}}{Dt} = P(\tilde{\mathbf{v}}) - \alpha(\mathbf{x}, t)D(\tilde{\mathbf{v}}) + T(\tilde{\mathbf{v}}). \quad (2)$$

In this paper, we will work with statistically steady problems, so  $\alpha$  will be assumed to just have spatial dependence. Assuming we have data  $\mathbf{G}_d$ , which is a scalar or a vector, a Bayesian<sup>a</sup> (or Least-squares-based frequentist) inverse problem can be constructed to infer  $\alpha|G_d$ . In this specific example, a classical deterministic optimization with no regularization, *i.e.*

$$\alpha|G_d = \arg \min_{\alpha} J = \arg \min_{\alpha} \|\mathbf{G}_d - \mathbf{G}_{\alpha}\|_2 \quad (3)$$

is used. Since skin friction data is available from LES, the following objective function was used:

$$J = \int_w \left[ \tau_w^{data}(s) - \tau_w^{model}(s) \right]^2 ds, \quad (4)$$

where the subscript  $w$  denotes the lower wall of the computational domain.

The resulting values of  $\alpha(\mathbf{x})$  are analogous to the maximum a posteriori (MAP) estimate in a Bayesian inversion with an iteratively updated prior. It may be argued that functional error also exists in the Boussinesq approximation and it could overwhelm the error arising from deficiencies in the transport equations for turbulent scalars. While this assertion may be true in situations such as those with strong secondary flows, an appropriate eddy viscosity (inferred via inverse problems or using a least-squares extraction from high fidelity solutions) was verified to result in a high degree of predictive accuracy of the mean flow quantities in a number of two-dimensional and mildly three dimensional problems that we have investigated. *Thus, before questioning the imperfectness of the Boussinesq approximation, it is important to properly assess the effect of deficiencies within eddy-viscosity models.*

In the above approach,  $\alpha$  is thus sought at every discrete location in the computational domain, and used in Eq. 2, conjoined with the conservation equations for the ensemble-averaged mass, momentum and energy. The resulting inverse problem is extremely high-dimensional and thus an efficient adjoint-based optimization framework is employed. Details are provided later in this section. This problem is statistically time-independent, and involved inferring  $\alpha$  at 14000 spatial locations. Figure 2 compares the baseline solution *i.e.*, with  $\alpha(\mathbf{x}) = 1 \forall \mathbf{x}$  and the solution inferred from the data. Figure 3 shows the distribution of the inferred function  $\alpha$  over part of the computational domain. This exercise helps directly quantify modeling inadequacies – **information that is already extremely valuable to the modeler**. Typical practice is to modify one of the above terms using physics-based arguments (for instance, Ref. 15). The inverse modeling procedure gives a quantitative target for functional modification. Alternately, machine learning methods (introduced in the next section) can also be used. Physically, the higher values of  $\alpha$  over the surface of the bump implies a higher rate of destruction of turbulence over the convex surface, which the original model was lacking. It has to be mentioned that, while the quantitative information on model inadequacy is useful, physical interpretations are somewhat loose because all other aspects of the model are also imperfect.

Further, The choice of introducing  $\alpha$  as a coefficient function of the destruction term of turbulence model, while being a good physical option, is not necessarily restrictive of the model. Equivalently, a more general function  $\delta$  could have been introduced as a model discrepancy in the following form

$$\frac{D\tilde{\mathbf{v}}}{Dt} = P(\tilde{\mathbf{v}}) - D(\tilde{\mathbf{v}}) + T(\tilde{\mathbf{v}}) + \delta. \quad (5)$$

<sup>a</sup>It is understood that the use of a Bayesian framework allows for a more rigorous specification of prior information about the model.

However, both Eq. 2 and 5 are equivalent in that introducing  $\alpha$  or  $\delta$  has fundamentally changed the model, and in fact, if  $D \neq 0$ ,  $\delta = (1 - \alpha)D$ . The advantage of Eq. 2 is that the resulting optimization problem is well-conditioned<sup>b</sup>.

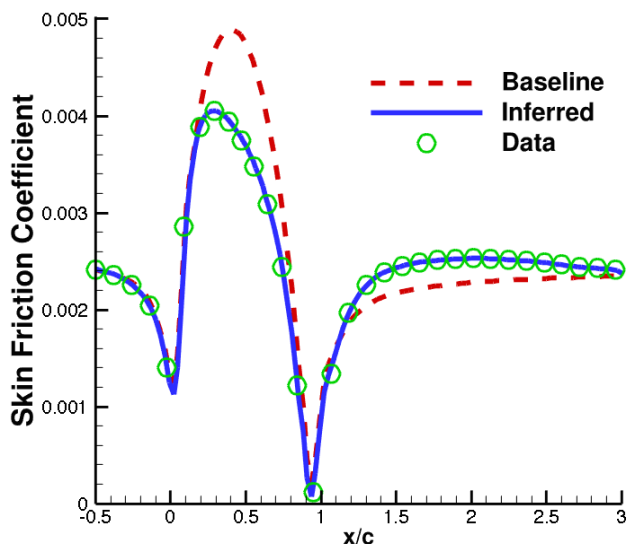


Figure 2. Skin-friction distribution on the lower surface of the computational domain in Figure 1.

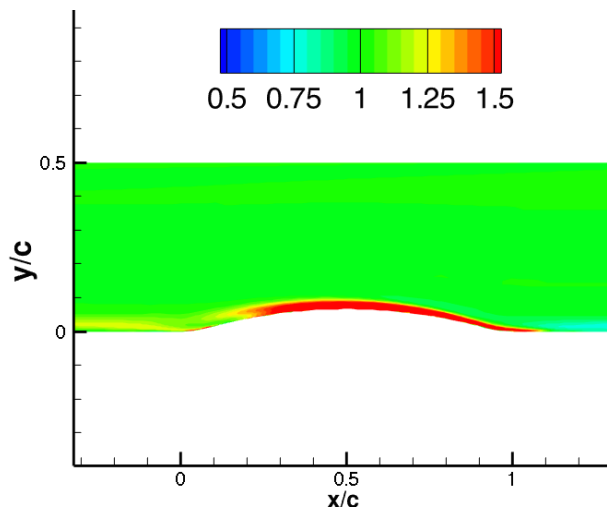


Figure 3. Contour of inferred  $\alpha$ .

## B. Application to a transition modeling problem

When free-stream turbulence levels are about 1% or more, boundary layers typically proceed from laminar to fully turbulent states without the occurrence of linear instability of the base state: this mode of transition is referred to as bypass transition. Models of bypass transition for general CFD codes are a relatively recent development (Ref. 18) compared to natural transition. At a fundamental level, the bypass process occurs as turbulence diffuses into the laminar boundary layer and generates disturbances known as Klebanoff modes. These grow in amplitude, and transition to turbulence occurs.<sup>19</sup> Closure models, at RANS level, are very loosely based on this mechanism. One method is to use the concept of intermittency  $\gamma$  to blend the flow from the laminar to the turbulent regions. Intermittency is associated with the spottiness of turbulence and manifests itself as a non-Gaussian behavior in turbulent flows. An “intermittency

<sup>b</sup>This is because of two reasons: First,  $\alpha$  is unit-less and second, the posterior value of  $\alpha$  can be expected to be of the same order as its prior value of 1, whereas  $\delta$  will have to be initiated at 0.

factor” can be formally defined as a fraction of the time turbulence is active, and modeling strategies are roughly based on this definition. *The intermittency factor is not a property of the simulation/experimental database, but is rather a property of the closure model.*

Consider the  $k - \omega$  closure (with values of the constants given by Wilcox<sup>23</sup>) and the Reynolds-averaged Navier-Stokes equations. Transition can be introduced by multiplying the production term of the  $k$  equation by a function  $\gamma(x)$ .  $\gamma$  is zero in laminar flow, and ramps up to unity in fully turbulent flow.  $\gamma$  appears within the turbulence model only as a factor in the production term of the turbulent kinetic energy transport equation:

$$\frac{Dk}{Dt} = 2\nu_T |S|^2 \gamma - C_\mu k \omega + \partial_j \left[ \left( \nu + \frac{\nu_T}{\sigma_k} \right) \partial_j k \right] \quad (6)$$

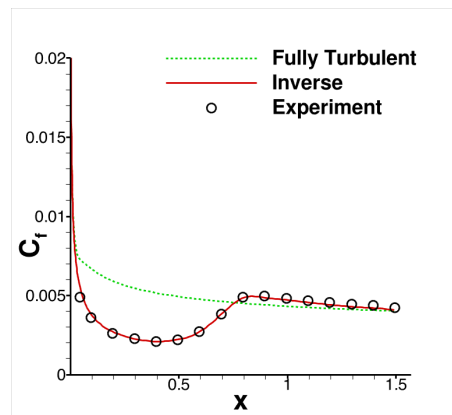
$$\frac{D\omega}{Dt} = 2C_{\omega 1} |S|^2 - C_{\omega 2} \omega^2 + \partial_j \left[ \left( \nu + \frac{\nu_T}{\sigma_\omega} \right) \partial_j \omega \right] \quad (7)$$

The eddy viscosity  $\nu_T$  is  $k/\omega$ .

The model developed by Ge et al.<sup>24</sup> is based on the idea that, in bypass transition under free-stream turbulence, non-zero  $\gamma$  diffuses into the boundary layer, allowing  $k$  to be produced, thereby creating eddy viscosity and further enhancing the diffusion of  $\gamma$ . In this way, transition occurs by penetration of free-stream turbulence into the boundary layer via molecular and turbulent diffusion. An intermittency transport equation is defined with a source term,  $P_\gamma$ , that contributes to producing intermittency inside the boundary layers. A sink term,  $E_\gamma$ , ensures that the boundary layer initially is laminar. The form of the model is

$$\frac{D\gamma}{Dt} = \partial_j \left[ \left( \frac{\nu}{\sigma_\gamma} + \frac{\nu_T}{\sigma_\gamma} \right) \partial_j \gamma \right] + P_\gamma - E_\gamma. \quad (8)$$

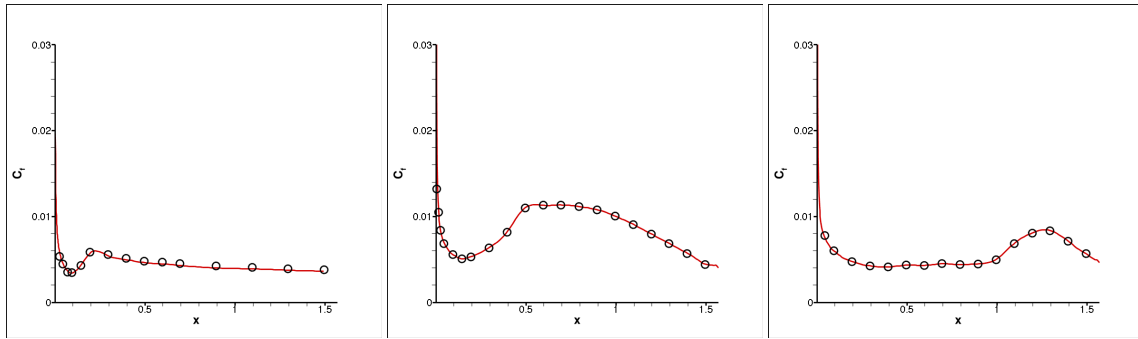
A detailed description of the model can be found in Ref. 24.



**Figure 4. Result of inverse problem to match the experimental skin friction for the T3A test case.<sup>25</sup> The initial condition assumes fully turbulent flow.**

We begin with the question of determining the intermittency field that will be required within the context of Eq. 7 to match given data on transitional flows. In this problem, the objective function in Eq. 4 is again employed. We will start with a fully turbulent assumption (*i.e.*,  $\gamma(\mathbf{x}) = 1$ ) and attempt to minimize  $\mathcal{J}$  by considering every grid point value of  $\gamma$  as parameters in an optimization problem. A sample result of this inverse problem is shown in Figure 4. The output of the problem  $\min_\gamma \mathcal{J}$  is thus a data field  $\gamma(\mathbf{x})$  that is suited to the  $k - \omega$  model. Note that intermittency is defined operationally, in terms of the model and the mechanism of ramping up the production term. It is not a physical variable that can be obtained from data, independently of its use.

Figure 5 shows the result of the inverse solution on the T3-series of test cases,<sup>25</sup> which correspond to bypass transition to turbulence over a flat plate with different turbulent intensities and pressure gradients. The left column in Figure 6 shows the inferred intermittency field. The right column shows the intermittency field obtained using the transition model of Ge et al.<sup>24</sup> with the mean flow imposed from the inferred solution. The most significant difference in the T3A and T3B cases is that the inferred field extends higher in the boundary layer near the inlet. This difference explains the over prediction of  $C_f$  near the inlet in the original Ge et al.<sup>24</sup> model and suggests that the sink term in Eq. 8 needs improvement.



(a) T3B (no pressure gradient,  $Tu_\infty = 6.5\%$ ) (b) T3C1 (pressure gradient,  $Tu_\infty = 10\%$ ) (c) T3C2 (pressure gradient,  $Tu_\infty = 3.7\%$ )

**Figure 5. Result of inverse solution to match skin friction for T3-series of test cases. Symbols: data; lines: computation.**

The T3C1 case has a high level of free-stream turbulence and shows a prompt transition. Again the inferred field shows the low intermittency extending higher in the boundary layer, near the entrance, than the model. The T3C2 case has a lower free-stream intensity and lower Reynolds number than T3C1 and the  $C_f$  prediction in Ref. 24 is fairly accurate; correspondingly, the inferred and modeled fields were confirmed to be fairly close. The cases presented here are representative of the other T3 cases: The inferred intermittency field shows the region of  $\gamma < 1$  extending higher into the boundary layer near the inlet and the  $\gamma = 1$  region is achieved farther inside the downstream boundary layer. The model postulates a sink term that is a function of  $R_t \equiv \nu_T/\nu$  and  $R_v \equiv d^2|\Omega|/2.188\nu$ . It is not clear that the discrepancy between the model and inferred fields can be parametrized by these terms. In the next section, a new parametrization, based on machine learning is proposed to improve the model.

Note that, unlike the turbulence modeling example in which  $\alpha$  was constructed as an unknown functional correction, the inference of  $\gamma$  in the transition modeling example is in a different context.

### C. Optimization procedure

The optimization problem uses a gradient-based Quasi-Newton method employing the limited memory Broyden-Fletcher-Goldfarb-Shanno (LBFGS) algorithm.<sup>22</sup> Since the optimization problem is extremely high dimensional (as the number of parameters equals the number of mesh points), an adjoint solver is required to efficiently compute gradients. Consider the discretized governing equations (including boundary conditions)  $\mathbf{R}_H(\mathbf{U}_H, \alpha) = \mathbf{0}$ , where  $\mathbf{U}_H$  represent the conserved variables in the RANS equations along with the turbulent scalars. For a discrete objective function  $J_H$ , the discrete adjoint equation<sup>20</sup> for the vector of adjoint variables  $\Psi_H$  is given by

$$\left[ \frac{\partial \mathbf{R}_H}{\partial \mathbf{U}_H} \right]^T \Psi_H = - \left[ \frac{\partial J_H}{\partial \mathbf{U}_H} \right]^T. \quad (9)$$

In this work, the software suite ADOL-C<sup>21</sup> has been used for automatic differentiation of the complete set of dependencies including the scalar transport variables. Given the adjoint solution, the gradient of the cost function with respect to the intermittency  $\alpha_i$  at every mesh point can be computed as

$$\frac{dJ_H}{d\alpha_i} = \Psi_H^T \frac{\partial \mathbf{R}_H}{\partial \alpha_i}$$

and used in the optimization loop.

## III. Machine Learning

The inverse approach presented in the previous sections results in a destruction correction field (in the turbulence modeling example) and the intermittency field (in the transition modeling example) that is required for the model to match data. Unlike in parameter estimation, in which the inferred parameters can be used directly in simulations, the  $\alpha(\mathbf{x})$  field by itself is of limited use in quantitative modeling. In other words, having inferred the optimal model correction function  $\alpha$ , it remains to convert the inference into modeling knowledge. To be useful in *predictive modeling*, the spatio-temporal dependence of  $\alpha(\mathbf{x})$  has to be transformed by extracting the functional relationship  $\alpha(\mathbf{x}) \approx y \approx f(\mathbf{q})$ ,

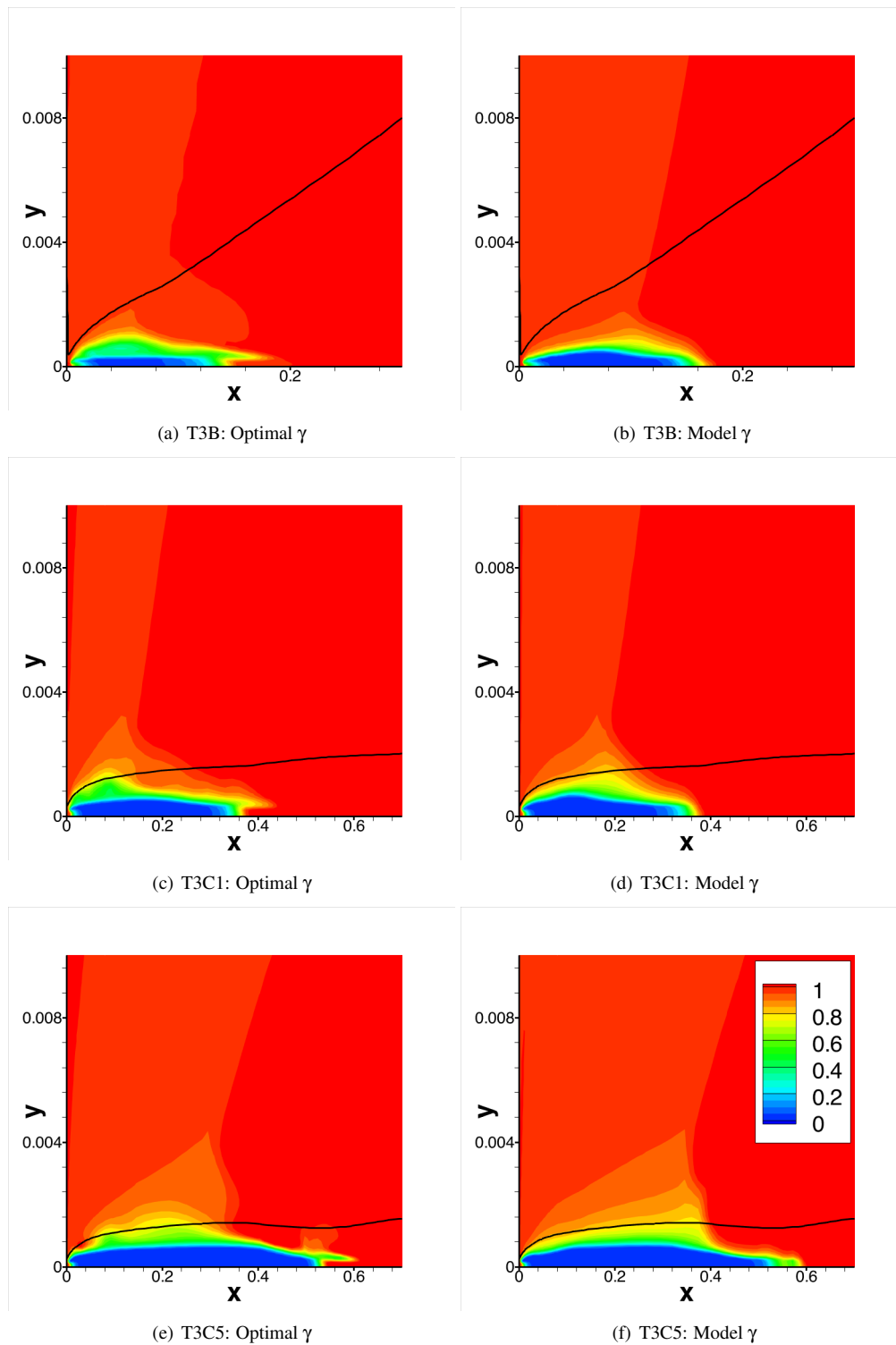


Figure 6. Comparison of inferred intermittency field and the model of Ge et al.<sup>24</sup> for selected cases, with the mean flow imposed from the inferred solution. The line is an iso-contour of 99% of the inlet free-stream velocity.

where  $\mathbf{q}$  are input features (derived from mean-field variables) that will be available **during** the predictive solution process. Such a functional relationship will have to be elicited while considering the output of a large number of inverse problems that are representative of the physical phenomena to be modeled. The elements of the feature vector  $\mathbf{q}$  have to be preferably locally non-dimensional quantities<sup>c</sup> such that the functional relationship  $f(\mathbf{q})$  is useful in a different problem in which  $\mathbf{q}$  variables are realizable.

We intend to use supervised learning techniques, specifically Gaussian processes (GP) and Artificial Neural Networks (NN). The features  $\mathbf{q} = [q_1 \ q_2 \ \dots \ q_M]^T$  are chosen from a number of possibilities through hill-climbing feature selection,<sup>27</sup> an algorithm that appends inputs to the usable set  $\mathbf{q}$  until increasing the size of  $\mathbf{q}$  no longer improves the performance when fitting  $y$ . The performance metric used in the current work for input selection is the sum squared error (SSE) on the validation set, with training performed by a baseline NN or GP.

Given a set of outputs  $\mathbf{Y}$  and its corresponding set of inputs  $\mathbf{Q}$ , normal practice<sup>26</sup> is to divide them into training, validation, and test sets  $(\mathbf{Q}_{train}, \mathbf{Y}_{train})$ ,  $(\mathbf{Q}_{val}, \mathbf{Y}_{val})$ , and  $(\mathbf{Q}_{test}, \mathbf{Y}_{test})$ . A ML algorithm is first applied to the training set to obtain  $f(\mathbf{q})$ . Then, the error on the validation set is used to adjust any hyper-parameters the ML algorithm may possess. The sum of squared errors (SSE) is:

$$\text{SSE} = \|\mathbf{Y} - f(\mathbf{Q})\|^2 \quad (10)$$

Through 10-fold cross-validation,<sup>26</sup> hyper-parameters that minimize the validation SSE are determined. Finally, the SSE of the test set is used to compare the performances of different ML algorithms. For the current work, the model is trained one last time on the entire dataset  $(\mathbf{Q}, \mathbf{Y})$ , using the optimal hyper-parameters, before being linked to the RANS solver. This helps ensure maximum accuracy by taking advantage of all available data.

## A. Neural Networks

A standard feed-forward neural network<sup>26</sup> is used. The NN 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. Figure 7 presents a sample NN. For this sample network, the values of the hidden nodes  $z_{1,1}$  through  $z_{1,H_1}$  would be constructed as

$$z_{1,j} = a_{(1)} \left( \sum_{i=1}^3 w_{ij}^{(1)} q_i \right) \quad (11)$$

where  $a_{(1)}$  and  $w_{ij}^{(1)}$  are the activation function and weights associated with the first hidden layer, respectively. Similarly, the second layer of hidden nodes is constructed as

$$z_{2,j} = a_{(2)} \left( \sum_{i=1}^{H_1} w_{ij}^{(2)} z_{1,i} \right) \quad (12)$$

Finally, the output is

$$y \approx f(\mathbf{q}) = a_{(3)} \left( \sum_{i=1}^{H_2} w_{ij}^{(3)} z_{2,i} \right) \quad (13)$$

Given training data, error back-propagation algorithms<sup>26</sup> are used to find  $w_{ij}^{(n)}$ .

One main advantage of NNs over GPs is efficiency. Once the weights are found, computing  $f(\mathbf{Q}_{test})$  depends only on the number of hidden nodes, and not on the (typically high) volume of the training data. Hyper-parameters of the NN method include the number of hidden layers, the number of nodes in each hidden layer, and the forms of the activation functions. In the current work, the open-source Fast Artificial Neural Network (FANN) library<sup>13</sup> is used for training and for integration with the RANS solver.

## B. Gaussian Processes

The current work employs Gaussian processes with a radial basis function (RBF)<sup>28,29</sup> kernel. The output can be represented as

$$y \approx f(\mathbf{q}) = \sum_{n=1}^N w_n \phi(\mathbf{q}, \mathbf{q}_n) \quad \phi(\mathbf{q}_m, \mathbf{q}_n) = e^{-\frac{\|\mathbf{q}_m - \mathbf{q}_n\|^2}{h^2}} \quad (14)$$

<sup>c</sup> $|S|\tau$  is an example of an acceptable feature, where  $|S|$  is the magnitude of the strain-rate tensor and  $\tau$  is a turbulent time-scale.



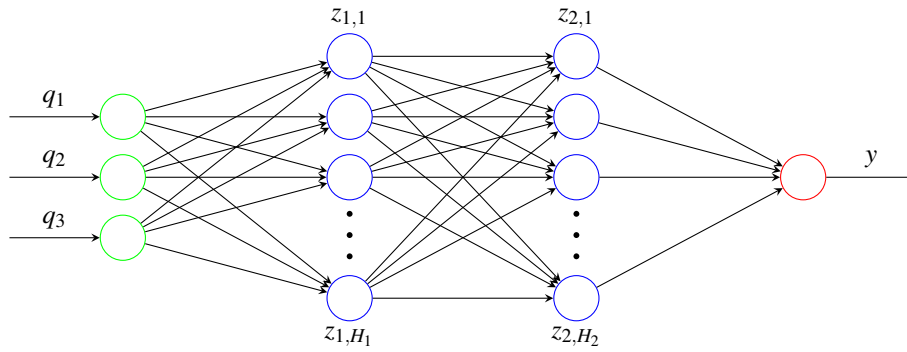


Figure 7. Network diagram for a feed-forward NN with three inputs, two hidden layers, and one output.

where  $w_n$  are weights,  $\phi$  is a Gaussian RBF, and  $\mathbf{q}_n$  are input points from the training data. It is therefore evident that, unlike for NNs, computing the output of a GP depends on  $N$ , the size of the training set.

Once again, the weights must be found. To do so, the design matrix,  $\Phi$ , is constructed:

$$\Phi_{ij} = \phi(\mathbf{q}_j, \mathbf{q}_i) \quad (15)$$

$\mathbf{q}_i, \mathbf{q}_j$  are training inputs, and  $\Phi$  has size  $N \times N$ . The weights are solved for as follows:

$$\mathbf{w} = [w_1 \ w_2 \ \dots \ w_N]^T = (\Phi\Phi^T + \lambda\mathbf{I})^{-1}\Phi\mathbf{Y} \quad (16)$$

$\lambda\mathbf{I}$  is a regularization term, and  $\lambda$  should be representative of the noise in the training data. This allows output  $y_{test}$  at test point  $\mathbf{q}_{test}$  to be written as

$$y_{test} \approx \underline{\phi}\mathbf{w} \quad \underline{\phi}_i = \phi(\mathbf{q}_{test}, \mathbf{q}_i) \quad (17)$$

Finally, the variance of  $y_{test}$  can be found through

$$\sigma^2(y_{test}) = \lambda(1 + \underline{\phi}^T(\Phi\Phi^T + \lambda\mathbf{I})^{-1}\underline{\phi}) \quad (18)$$

In the current implementation, the inputs are scaled individually before being used as training data. Hence, the scaling factors and  $h$  are hyper-parameters of the method, and are optimized.

### C. Results on Transition modeling problem

In the transition modeling problem, we will first write Eq. 8 as

$$\frac{D\gamma}{Dt} = \partial_j \left[ \left( \frac{v}{\sigma_l} + \frac{v_T}{\sigma_\gamma} \right) \partial_j \gamma \right] + S_\gamma. \quad (19)$$

Since convection and diffusion are fundamental transport properties, the functional form of the source term  $S_\gamma$  (*i.e.* production minus destruction) will be targeted for improvement. To be consistent with predictive RANS modeling,  $S_\gamma$  will have to be extracted from the **inferred flow-field** (rather than the DNS or LES) which was computed in the previous section. This can be determined by considering Eq. 19 and computing  $S_\gamma$  by using the following balance (on the optimal mean and intermittency flow-fields)

$$S_\gamma = \frac{D\gamma}{Dt} - \partial_j \left[ \left( \frac{v}{\sigma_l} + \frac{v_T}{\sigma_\gamma} \right) \partial_j \gamma \right] \quad (20)$$

For the T3-series of test cases, the selected features are

$$\mathbf{q} = \left[ k, \omega, \gamma, \frac{\partial u}{\partial y}; \frac{\partial v}{\partial y}; d^2\Omega/\nu \right], \quad (21)$$

where  $\Omega$  is the vorticity magnitude and  $d$  is the distance from the nearest viscous wall. These features were selected from a set that included the full velocity gradient tensor, the transported scalars  $k, \omega, \gamma$ , and three non-dimensional

parameters  $\{v_t/v; d^2\Omega/v; \Omega/\omega\}$  that appear in the original Ge et al.<sup>24</sup> model. A standard hill-climbing<sup>27</sup> algorithm was used to narrow down the feature set. Only points within the boundary layer were considered for Machine Learning. Outside the boundary layer, the analytical source term from the baseline model was used. The RANS output was divided into two parts: 80% of the data was used training and 20% for validation. The validation sets were used to adjust hyper-parameters in both the NN and GP models. Figure 8 plots the original versus the predicted values of  $S_\gamma$  produced by the optimal NN and GP. The SSE for the GP method was found to be four times smaller than that of the NN. It has to be mentioned that the GP method is based on an in-house code for regression and optimization, whereas the NN code uses the FANN<sup>13</sup> library. Though this behavior is representative of the T3-series, this demonstration is preliminary in nature and conclusions about the merits of each ML method cannot be drawn based on the limited number of evaluations. Tracey et al.<sup>12</sup> provide a more detailed discussion of specific issues and challenges of applying Neural Networks to learning turbulence model terms.

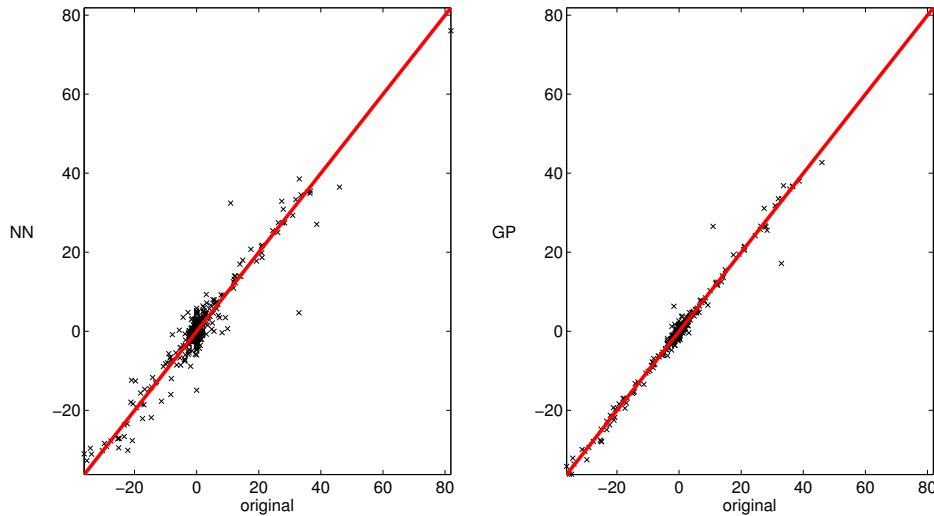


Figure 8. Original versus predicted  $S_\gamma$  for various ML methods for T3C1 transition case.

#### IV. Data Injection

The inversion and machine learning steps should be considered as *pre-processing* or *off-line* steps in the proposed framework. During the training stage of the ML model, theoretical knowledge and intuition should be used to inform the ML step of asymptotic behaviors and in regions of sparsely-populated feature space. An example in turbulence modeling would be to use rapid distortion theory to supplement the data set in regions of high deformation rates. During the predictive simulation, (at each time-step or solver iteration), the solver will pass feature vectors  $\mathbf{q}_*$  to the ML ‘testing’ routine and receive appropriate model correction quantities  $\alpha_* \approx \mathbf{y}_*(\mathbf{q}_*)$  for injection into the data-driven turbulence model. These quantities are requested at every spatial grid point in the computational domain. If the GP model is used, the information on the probabilistic structure of  $\alpha_*$  can be utilized to generate a number of realizations. **The ensemble of simulations can account for the impact of structural uncertainties in the turbulence model.** If an ensemble of simulations are generated, the inevitably large dimension of  $\alpha_*$  will require efficient reduced-order spatial and stochastic representations<sup>30,31</sup> of  $\alpha_*$  to keep the computation tractable.

Simulations of bypass transition, in which the inferred and reconstructed quantity is  $S_\gamma$  in the transition model introduced in the previous section is used as a demonstrative example. Figure 9 compares the predicted source term  $S_\gamma(q)$  versus the actual source term  $S_\gamma(\mathbf{x})$  via the inverse solution for the T3C1 transition case. The close agreement confirms the validity of the new parametrization. The inferred, predicted and baseline model<sup>24</sup> intermittency profiles for the T3C1 case are shown in Figure 10. For the data injection, initial computations were performed with the baseline model. The machine-learned source term  $S_\gamma$  was then blended into the solution in the form  $S_\gamma = (1 - \beta) [S_\gamma]_{baseline} + \beta [S_\gamma]_{ML}$ , gradually increasing  $\beta$  from 0 to 1. The inadequacy of the baseline model in predicting the high levels of intermittency required in the context of the  $k - \omega$  closure is again confirmed. Note that, to attain the intensity of turbulence in the fully developed region, the intermittency variable has to assume a value greater than unity. This is a failure of the closure model to represent the physics in sufficient detail and thus the role of the intermittency as a

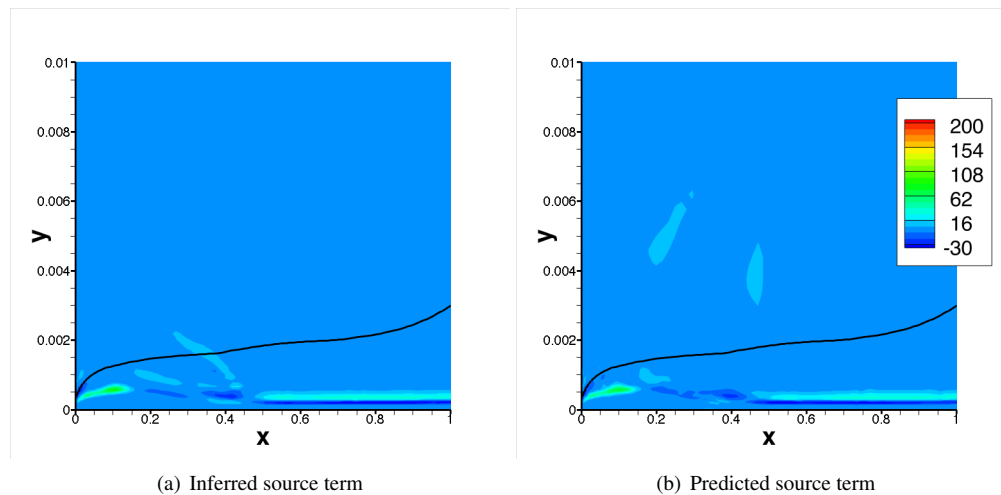


Figure 9. Comparison of inferred source term and neural network prediction for T3C1.

model variable, rather than a physical one is emphasized. Figure 11 shows the ability of the machine learning method to reproduce the inferred skin friction results. The noisy nature of the prediction is partly a result of the lack of detail in the data (a total of six T3-series cases were used for the inference). Improvements can be expected by using a larger set of inverse problems as well as by performing the inference with respect to a wider set of objective functions.

## V. Summary

The traditional approach to closure modeling does not leverage the availability of large amounts of data from high fidelity simulations and high resolution experiments. The proposed set of approaches highlight the potential of inverse modeling and machine learning techniques to quantify and account for deficiencies in turbulence and transition modeling using data from simulations and measurements. The focus on the functional forms of the closure (rather than on parameters in the model) offers the promise of improved predictive models under the premise that the data is diverse enough to characterize the physical phenomena to be modeled. The inverse modeling step, by itself, provides direct information on the model inadequacy which is of value to the modeler. It has to be mentioned that machine learning should be considered as just one tool to convert the inferred information into modeling knowledge. The modeler can, in principle, use information from the inverse problem to make parametric corrections to existing models. The proposed techniques are general enough to be applied in any physical modeling situation in which appropriate data is available.

Continuing work is aimed at inferring model inadequacies in a wider range of problems to specifically target turbulent flow separation and bypass transition. While this paper suggests that there is merit in exploring data-driven techniques, several details are being addressed currently. For instance, the choice of the objective function  $\mathcal{J}$  in the inverse problem is arbitrary; the solution to the inverse problem may be non-unique; the map  $\alpha(\mathbf{q})$  will not be injective, etc. Towards this end, a consistent Bayesian framework for inversion/machine learning/uncertainty propagation is being built, such that the targeted model can offer improved predictions as well as confidence intervals on the predicted outputs.

## Acknowledgments

This work was funded by the NASA Aeronautics Research Institute (NARI) under the Leading Edge Aeronautics Research for NASA (LEARN) program. The authors acknowledge insightful discussions with Paul Durbin, Shivaji Medida, Juan Alonso, Brendan Tracey, and Philippe Spalart.

## References

<sup>1</sup>Slotnick, J., Khodadoust, A., Alonso, J., Darmofal, D., Gropp, W., Lurie, E. & Mavriplis, D., "CFD vision 2030 study: A path to revolutionary computational aerosciences," *NASA Technical Report CR-2014-218178*, 2013.

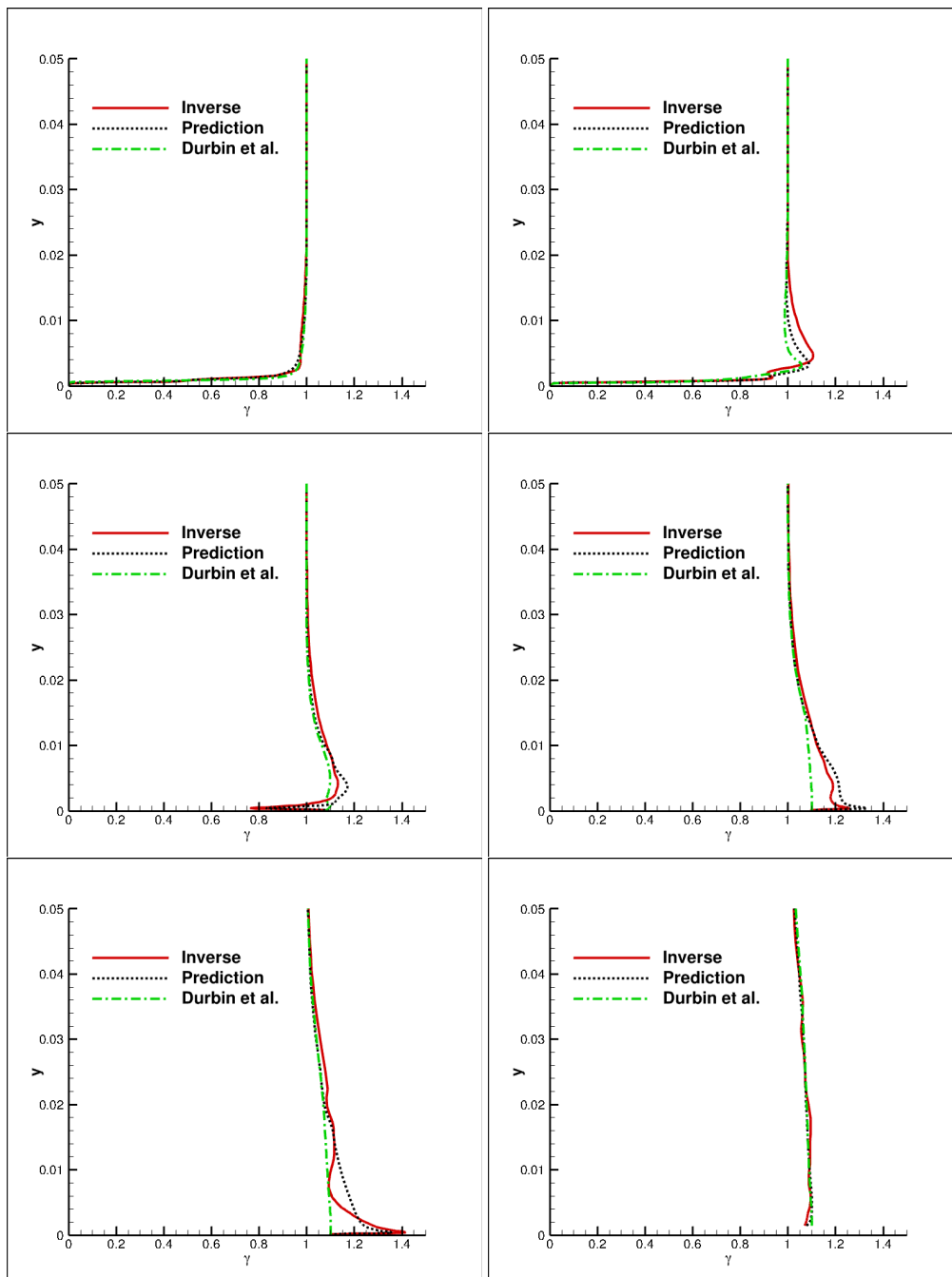


Figure 10. Intermittency field at selected streamwise stations for inferred, predicted and model for T3C1. Locations are  $x=0.1,0.2,0.4,0.6,1.0,1.5$ .

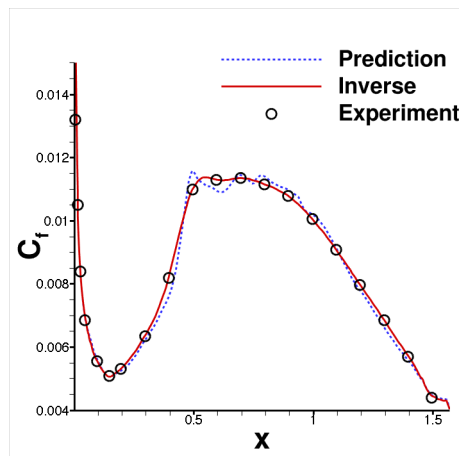


Figure 11. Skin friction prediction for T3C1.

<sup>2</sup>G. Gerolymos, E. Sauret & I. Vallet, "Contribution to single-point closure Reynolds-stress modelling of inhomogeneous flow," *Theoretical Computational Fluid Dynamics*, Vol. 17, pp. 407–431, 2004.

<sup>3</sup>B. Younis, T. Gatski & C. Speziale, "Towards a rational model for the triple velocity correlations of turbulence," *Proc. R. Soc. London A*, Vol. 456, pp. 909–920, 2000.

<sup>4</sup>M. Milano & P. Koumoutsakos, "Neural network modeling for near wall turbulent flow," *Journal of Computational Physics*, Vol. 182(1), pp. 1–26, 2002.

<sup>5</sup>I. Marusic, V. Interrante, P. K. Subbareddy & A. Moss, "Chapter 13 - Real-time feature extraction for the analysis of turbulent flows," *Data Mining for Scientific and Engineering Applications*, 2001.

<sup>6</sup>S. Yarlanki, B. Rajendran & H. Hamann, "Estimation of turbulence closure coefficients for data centers using machine learning algorithms," *13<sup>th</sup> IEEE Intersociety Conference on Thermal and Thermomechanical Phenomena in Electronic Systems*, San Diego, CA, 2012.

<sup>7</sup>Dow, E. & Wang, Q., "Uncertainty Quantification of Structural Uncertainties in RANS Simulations of Complex Flows," *AIAA Paper 2011-3865*, 2011.

<sup>8</sup>Emory, M., Pecnik, R., and Iaccarino, G., "Modeling Structural Uncertainties in Reynolds-Averaged Computations of Shock/Boundary Layer Interactions," *AIAA Paper 2011-0479*, 2011.

<sup>9</sup>Cheung, S. H., Oliver, T. A., Prudencio, E. E., Prudhomme, S., & Moser, R. D., "Bayesian Uncertainty Analysis with Applications to Turbulence Modeling," *Reliability Engineering and System Safety*, Vol. 96, pp. 1137–1149, 2011.

<sup>10</sup>Oliver, T. and Moser, R., "Bayesian uncertainty quantification applied to RANS turbulence models," *Journal of Physics: Conference Series*, Vol. 318, p. 042032, 2011.

<sup>11</sup>Edeling, W., Cinnella, P., Dwight, R., & Bijl, H., "Bayesian estimates of parameter variability in the  $k - \epsilon$  turbulence model," *Journal of Computational Physics*, Vol. 258, pp. 73–94, 2014.

<sup>12</sup>Tracey, B., Duraisamy, K. & Alonso, J., "A Machine Learning Strategy to Assist Turbulence Model Development," *Proc. AIAA Scitech conference*, Orlando, Florida, 2015.

<sup>13</sup>Nissen, S., "Implementation of a fast artificial neural network library (FANN)," *Report, Department of Computer Science University of Copenhagen (DIKU)*, Vol. 31, 2003.

<sup>14</sup>P. R. Spalart & S. R. Allmaras, "A One-Equation Turbulence Model for Aerodynamic Flows," *30<sup>th</sup> Aerospace Sciences Meeting & Exhibit*, Reno, NV, Jan 1992.

<sup>15</sup>Shur, M. L., Strelets, M. K., Travin, A. K. & Spalart, P. R., "Turbulence Modeling in Rotating and Curved Channels: Assessing the Spalart-Shur Correction," *AIAA Journal* Vol. 38(5), pp. 784–792, 2000.

<sup>16</sup>Webster, D., DeGraaff, D. & Eaton, J., "Turbulence characteristics of a boundary layer over a two-dimensional bump," *Journal of Fluid Mechanics*, Vol. 320(1), pp. 53–69, 1996.

<sup>17</sup>Radhakrishnan, S., Keating, A., Piomelli, U. & Silva Lopes, A., "Reynolds-averaged and Large-eddy simulations of turbulent non-equilibrium flows," *Journal of Turbulence*, Vol. 7, 2006.

<sup>18</sup>Menter, F. R., Langtry, R. & Volker, S., "Transition modelling for general purpose CFD codes," *Flow, Turbulence and Combustion* Vol. 77(1-4) pp. 277–303, 2006.

<sup>19</sup>Durbin, P. & Wu, X., "Transition beneath vortical disturbances," *Annual Review of Fluid Mechanics*, Vol. 39, pp. 107–128, 2007.

<sup>20</sup>Giles, M. & Pierce, N., "An introduction to the adjoint approach to design," *Flow, Turbulence and Combustion*, Vol. 65(3-4), pp. 393–415, 2000.

<sup>21</sup>Griewank, A., Juedes, D. & Utke, J., "Algorithm 755: ADOL-C: A Package for the Automatic Differentiation of Algorithms Written in C/C++," *ACM Transactions on Mathematical Software*, Vol. 22(2), pp. 131–167, 1996.

<sup>22</sup>Dennis, J.E. & Moré, J., "Quasi-Newton methods, motivation and theory," *SIAM review*, Vol. 19(1), pp. 46–89, 1977.

<sup>23</sup>Wilcox, D., *Turbulence modeling for CFD*, Vol. 2, DCW industries, 2006.

<sup>24</sup>Ge, X., Arolla, S. & Durbin, P., "A bypass transition model based on the intermittency function," *Flow, Turbulence and Combustion* Vol. 93(1), pp. 37–61, 2014.

- <sup>25</sup>Roach, P. E. & Brierley, D. H., "The influence of a turbulent free-stream on zero pressure gradient transitional boundary layer development Part 1: Test cases T3A and T3B," *Numerical Simulation of Unsteady Flows and Transition to Turbulence*, pp. 319–347, ERCOFTAC, 1992.
- <sup>26</sup>Bishop, C. M., *Pattern Recognition and Machine Learning*, Springer, 2006.
- <sup>27</sup>Kohavi, R. & John, G. H., "Wrappers for Feature Subset Selection," *Artificial Intelligence*, Vol. 97(1) pp. 273–324, 1997.
- <sup>28</sup>Park, J. & Sandberg, I. W., "Universal Approximation Using Radial-Basis-Function Networks," *Neural Computation*, Vol. 3(2) pp. 246–257, 1992.
- <sup>29</sup>Rasmussen, C. E. & Williams, C. K. I., *Gaussian Processes for Machine Learning*, MIT Press, 2006.
- <sup>30</sup>Duraisamy, K. & Chandrashekar, P., "Goal oriented uncertainty propagation using stochastic adjoints," *Computers and Fluids* Vol. 66, pp. 10–20, 2012.
- <sup>31</sup>Lakshminarayan, V., Duraisamy, K. & Alonso, J., "Estimation and Adaptive Control of Spatial, Temporal and Stochastic Errors Using Adjoints for Unsteady Aerodynamic Applications," 51<sup>st</sup> *AIAA Aerospace Sciences Meeting*, Grapevine, TX, 2013.