Correcting an Algebraic Transition Model using Field Inversion and Machine Learning

Krzysztof J. Fidkowski*
University of Michigan, Ann Arbor, MI, 48188

This paper investigates the feasibility and accuracy of field inversion and machine learning for correcting an algebraic transition model in moderate Reynolds-number flows. An algebraic model is chosen for simplicity and to keep simulation costs low by avoiding additional equations. The discretization is a high-order discontinuous finite-element method. The inverse problem for the correction field is solved using gradient-based optimization with adjoint sensitivities. Data for the inversion comes from unsteady turbulence simulations and, for the purpose of this initial assessment, experiments. The goal of the study is to assess suitability of the algebraic transition model for use in optimization and adaptation of unsteady turbulent flows using the method of dynamic closures. We show that the algebraic model does yield correct local sensitivities for the cases tested. In addition, we identify the need for a new correction approach that incorporates parameter optimization into the field inversion problem to improve learnability of the correction field in the machine-learning stage. The results compare dynamic corrections for RANS with the transition model and without it, and the latter is shown unable to reliably predict transition in spite of the similarity between the turbulent production correction and the intermittency function.

I. Introduction

Whereas advances in computational power have made scale-resolving turbulence simulations more prevalent, the cost of these simulations still prohibits their use in fast-turnaround analyses and designs of practical aerospace vehicles. Furthermore, the chaotic nature of unsteady turbulence prevents a direct application of the adjoint method for local sensitivity calculations, which are required for efficient optimization, uncertainty quantification, and mesh adaptation. Against this backdrop, cheaper turbulence modeling techniques such as RANS remain of high value, despite their inaccuracies. Mitigating these inaccuracies motivates the present work in model corrections.

In our previous work, we have shown how RANS can be augmented with dynamic corrections, driven by unsteady simulations [1, 2]. The dynamic corrections allow for stable adjoint-based sensitivities, which are calculated from the steady RANS model. They also keep the costs of optimization and adaptation simulations low: multiple iterations of optimization/adaptation are done using RANS, whereas unsteady simulations are only needed a few times to rebase the correction.

The accuracy of the adjoint-based sensitivities depends on the suitability of the RANS model, i.e. its proximity to the unsteady simulation, which can be assessed by the magnitude of the correction and whether or not it lends itself to a model that is a function of the local resolved states. In this sense, fully-turbulent RANS models such as Spalart-Allmaras are suitable for correction at high-Reynolds numbers. For moderate Reynolds numbers in the transitional regime, fully-turbulent RANS models become less suitable, requiring larger corrections that, as we show here, cannot always be robustly modeled from the state. Hence, in this work, we investigate field inversion and machine learning for RANS with a transition model.

We apply FIML to the algebraic SA-BCM transition model [3, 4]. For the correction field, we investigate the standard multiplicative factor on the production term, and we demonstrate the need for parameter optimization during inversion to effectively move the transition location. We show that fully turbulent RANS can predict integrated quantities well, but that, as expected, it does not predict the transitional pressure or skin friction distribution. A correction based on an integrated quantity improves the measure of that quantity, but not of the flowfield or surface qunatities. Inverting on a distribution improves the prediction of surface stress, yet a local model for this correction is inaccurate for the small networks tested. In contrast the transition model can predict both integrated quantities and surface distributions well, although its accuracy depends on closure coefficients used. A correction mitigates this dependency and remains accurate

---

[*] Professor, Department of Aerospace Engineering, AIAA Associate Fellow.

with a local model. However, a multiplicative correction is insufficient for controlling production when the intermittency is low, and hence the need for an alternative correction procedure or optimization of the transition parameters. In this work we take the latter approach by including parameter calibration during field inversion.

In this paper, Section II presents the RANS and transition models used in this work. Section III describes the model correction approaches and the field-inversion algorithm for obtaining the correction field. Section IV presents the machine-learning model for the correction and the training approach. Results in Section V demonstrate the feasibility of applying FIML to several transitional flows, and the resulting local sensitivities. Section VI concludes with a summary and future directions.

II. Model and Discretization

In this work, we discretize the compressible Reynolds-Averaged Navier-Stokes (RANS) equations with the Spalart-Allmaras (SA) closure [5] and an algebraic intermittency model [3, 4], using a discontinuous finite-element method.

A. Baseline Turbulence Model

The RANS-SA equations written in index notation with implied summation are

$$\begin{array}{lllll} \partial_t \rho & + \ \partial_j(\rho u_j) & & = & 0 \\ \partial_t(\rho u_i) & + \ \partial_j(\rho u_j u_i + p\delta_{ij}) & - \ \partial_j \tau_{ij} & = & 0 \\ \partial_t(\rho E) & + \ \partial_j(\rho u_j H) & - \ \partial_j(u_i\tau_{ij} - q_j) & = & 0 \\ \partial_t(\rho\tilde{\nu}) & + \ \partial_j(\rho u_j \tilde{\nu}) & - \ \partial_j\left[\frac{1}{\sigma}\rho(\nu + \tilde{\nu} f_n)\partial_j\tilde{\nu}\right] & = & S_{\tilde{\nu}} \end{array} \tag{1}$$

where the source term for the turbulence model $\tilde{\nu}$ equation is

$$S_{\tilde{\nu}} = \tilde{\gamma}\beta P - D - \frac{1}{\sigma}(\nu + \tilde{\nu} f_n)\partial_j\rho\partial_j\tilde{\nu} + \frac{c_{b2}\rho}{\sigma}\partial_j\tilde{\nu}\partial_j\tilde{\nu}. \tag{2}$$

In the above equations, ρ is the density, ρu_j is the momentum, E is the total energy, $H = E + p/\rho$ is the total enthalpy, $p = (\gamma - 1)\left(\rho E - \frac{1}{2}\rho u_i u_i\right)$ is the pressure, γ is the ratio of specific heats, P is the turbulence production, D is the turbulence destruction, $\tilde{\gamma}$ is an intermittency function, β is a production correction term, and i, j index the spatial dimension, dim. The viscous stress, τ_{ij}, is

$$\tau_{ij} = 2(\mu + \mu_t)\bar{\epsilon}_{ij}, \qquad \bar{\epsilon}_{ij} = \frac{1}{2}(\partial_i u_j + \partial_j u_i) - \frac{1}{3}\partial_k u_k \delta_{ij}, \tag{3}$$

where μ is the laminar dynamic viscosity, which is here treated as constant but could also be calculated using Sutherland's law [6]. The eddy viscosity, μ_t, is

$$\mu_t = \begin{cases} \rho\tilde{\nu} f_{v1} & \tilde{\nu} \geq 0 \\ 0 & \tilde{\nu} < 0 \end{cases} \qquad f_{v1} = \frac{\chi^3}{\chi^3 + c_{v1}^3}, \quad \chi = \frac{\tilde{\nu}}{\nu}. \tag{4}$$

The heat flux, q_j, is computed from the temperature gradient, $q_j = (\kappa + \kappa_t)\partial_i T$, where the laminar and turbulent thermal conductivities are $\kappa = C_p\mu/Pr$ and $\kappa_t = C_p\mu_t/Pr_t$ respectively. Pr and Pr_t are the laminar and turbulent Prandtl numbers, and C_p is the specific heat at constant pressure. The production term, P, is

$$P = \begin{cases} c_{b1}\tilde{S}\rho\tilde{\nu} & \chi \geq 0 \\ c_{b1}S\rho\tilde{\nu} & \chi < 0 \end{cases}, \tag{5}$$

where the modified vorticity $\tilde{S}$ is written as

$$\tilde{S} = \begin{cases} S + \overline{S} & \overline{S} \geq -c_{v2}S \\ S + \dfrac{S(c_{v2}^2 S + c_{v3}\overline{S})}{(c_{v3} - 2c_{v2})S - \overline{S}} & \overline{S} < -c_{v2}S \end{cases}, \quad \overline{S} = \frac{\tilde{\nu} f_{v2}}{\kappa^2 d^2}, \quad f_{v2} = 1 - \frac{\chi}{1 + \chi f_{v1}}. \tag{6}$$

In Eqn. 6, $S = \sqrt{2\Omega_{ij}\Omega_{ij}}$ is the vorticity magnitude (summation implied on i, j), $\Omega_{ij} = \frac{1}{2}(\partial_i v_j - \partial_j v_i)$ is the vorticity tensor, and d is the distance to the closest wall. The destruction term, D, is given by

$$D = \begin{cases} c_{w1} f_w \dfrac{\rho\tilde{\nu}^2}{d^2} & \chi \geq 0 \\ -c_{w1} \dfrac{\rho\tilde{\nu}^2}{d^2} & \chi < 0 \end{cases}, \quad f_w = g\left(\frac{1 + c_{w3}^6}{g^6 + c_{w3}^6}\right)^{1/6}, \quad g = r + c_{w2}(r^6 - r), \quad r = \frac{\tilde{\nu}}{\tilde{S}\kappa^2 d^2}. \tag{7}$$

Finally, the multiplier on adding $\tilde{\nu}$ in Eqn. 1 is $f_n = 1$ for positive $\tilde{\nu}$ and

$$f_n = \frac{c_{n1} + \chi^3}{c_{n1} - \chi^3}, \quad \text{when } \chi < 0. \tag{8}$$

The baseline closure coefficients used in this work are

$$\begin{array}{rclrclrcl} c_{b1} & = & 0.1355 & c_{w1} & = & \dfrac{c_{b1}}{\kappa^2} + \dfrac{1 + c_{b2}}{\sigma} & c_{v1} & = & 7.1 \\ c_{b2} & = & 0.622 & c_{w2} & = & 0.3 & \kappa & = & 0.41 \\ \sigma & = & 2/3 & c_{w3} & = & 2 & Pr_t & = & 0.9 \\ c_{n1} & = & 16 & c_{v2} & = & 0.7 & c_{v3} & = & 0.9 \end{array}$$

B. Transition Model

The intermittency function $\tilde{\gamma}$ in Eqn. 2 attenuates turbulent production to maintain laminar flow until transition to turbulence. Various methods exist for modeling $\tilde{\gamma}$, many relying on additional equations. Presently, we use an algebraic model, BCM [4], named after the initial authors. This model is heavily empirical and contains tunable factors, but it is inexpensive and simple to implement. Our goal is to assess whether the BCM model is suitable for predicting sensitivities after correction with FIML.

In the BCM model, the intermittency is

$$\tilde{\gamma} = 1 - \exp\left(-\sqrt{T_1} - \sqrt{T_2}\right), \tag{9}$$

where

$$T_1 = \frac{\max(Re_\theta - Re_{\theta c}, 0)}{c_{t1} Re_{\theta c}}, \qquad T_2 = \max\left(\frac{\nu_t}{c_{t2}\nu}, 0\right). \tag{10}$$

The momentum thickness Reynolds number is correlated to the vorticity Reynolds number, Re_S through [4],

$$Re_\theta = \frac{Re_S}{2.193}, \quad Re_S = \frac{d^2 S}{\nu}, \tag{11}$$

and hence can be measured locally without any integration. The critical momentum thickness Reynolds number is based on the empirical formula

$$Re_{\theta c} = 803.73\,(Tu_\infty + 0.6067)^{-1.027}, \tag{12}$$

where Tu_∞ is the freestream turbulence level that need not correlate directly to $\tilde{\nu}_\infty$ [7].

The closure coefficients used in this work are

$$c_{t1} = 0.002, \qquad c_{t2} = 50, \qquad Tu_\infty = .01. \tag{13}$$

Finally, β that appears in Eqn. 2 is the FIML production correction factor that is applied independently of the intermittency.

C. Finite-Element Discretization

We use a discontinuous Galerkin (DG) finite-element spatial discretization [8], with the Roe [9] convective flux and the second form of Bassi and Rebay (BR2) [10] for the viscous treatment. The state is approximated on an unstructured mesh of non-overlapping elements using polynomials of order p. The semi-discretized form of the equations is

$$\mathbf{M}\frac{d\mathbf{U}}{dt} + \mathbf{R}(\mathbf{U}) = \mathbf{0}, \tag{14}$$

where $\mathbf{U} \in \mathbb{R}^N$ is the discrete state vector, N is the total number of unknowns including the state rank, $\mathbf{R}(\cdot) \in \mathbb{R}^{N\times N}$ is the nonlinear spatial residual, and $\mathbf{M} \in \mathbb{R}^{N\times N}$ is the block-element sparse mass matrix. For steady simulations, the time derivative term drops out, although pseudo-time continuation remains in the solver to drive the steady residual to zero [11]. The solver consists of a Newton-Raphson method with the generalized minimum residual (GMRES) [12] linear solver, preconditioned by an element-line Jacobi smoother with a coarse-level ($p = 1$) correction [13, 14]. For unsteady simulations, we use a third-order modified extended backward difference formula [15] applied to the the semi-discrete form.

Solving a field inversion problem efficiently requires an adjoint, which here is calculated in discrete form. For a scalar output $J(\mathbf{U})$, the discrete steady adjoint vector, $\boldsymbol{\Psi} \in \mathbb{R}^N$, is the local sensitivity of J to perturbations in the steady residual, $\mathbf{R}$ [16]. Linearization of the residual and output shows that the adjoint satisfies the following linear equation,

$$\left(\frac{\partial \mathbf{R}}{\partial \mathbf{U}}\right)^T \boldsymbol{\Psi} + \left(\frac{\partial J}{\partial \mathbf{U}}\right)^T = \mathbf{0}. \tag{15}$$

This equation is solved using the same preconditioned-GMRES method used in the primal solver.

III. Correction Field Inversion

We perform field inversion to obtain the correction field $\beta(\vec{x})$ that appears in Eqn. 2. The objective for the field inversion arises from matching higher-fidelity data, either experimental or numerical, in the form of integrated scalars, surface distributions, or entire flow-fields. In this work we consider both integrated forces and surface-stress distributions.

A. High-Fidelity Data

One way to obtain higher-fidelity data is through unsteady, e.g. scale-resolving, simulations, which can be appropriately averaged for training steady RANS models [1, 2]. Let $\bar{\mathbf{U}}$ be a statistically-steady flow state computed from the time-average of the unsteady simulation after initial transients,

$$\bar{\mathbf{U}} = \frac{1}{T_f - T_i}\int_{T_i}^{T_f} \mathbf{U}(t)\, dt, \tag{16}$$

where T_i is the start time, taken sufficiently large to minimize startup transient effects, and T_f is the final time, taken sufficiently larger than T_i to yield an adequate statistical mean. The goal of field inversion is to determine the $\beta(\vec{x})$ that makes the corrected RANS solution match $\bar{\mathbf{U}}$ or a functional of it.

The field-inversion error, $\mathcal{F}$, measures the accuracy of this match, and presently we consider both scalar and distribution measures,

$$\begin{aligned} \mathcal{F}^{\text{output}}(\mathbf{U}(\beta), \bar{\mathbf{U}}) &= \frac{1}{2}(J(\mathbf{U}(\beta)) - J(\bar{\mathbf{U}}))^2, && (17)\\ \mathcal{F}^{\text{dist}}(\mathbf{U}(\beta), \bar{\mathbf{U}}) &= \frac{1}{2}\int_{\text{airfoil}} \|\boldsymbol{\sigma}(\mathbf{U}(\beta)) \cdot \vec{n} - \boldsymbol{\sigma}(\bar{\mathbf{U}}) \cdot \vec{n}\|^2\, ds, && (18) \end{aligned}$$

where $J(\mathbf{U})$ is a scalar output, such as a drag coefficient, $\boldsymbol{\sigma}$ is the stress tensor, and $\vec{n}$ is the unit normal vector on the surface, which is an airfoil in the problems considered here. Instead of the time-averaged unsteady solution, experimental data can also be used in the above error measures.

B. Field Inversion

We represent the correction field using a $p = 1$ Lagrange DG basis on each element, with a nominal value of 1. Denoting by $\boldsymbol{\beta}$ the vector of basis coefficients in the approximation, the field inversion becomes a discrete optimization problem,

$$\begin{aligned} \min_{\boldsymbol{\beta}} \quad & J^{\text{inv}} \equiv \mathcal{F}(\mathbf{U}(\boldsymbol{\beta}), \bar{\mathbf{U}}) + \frac{\gamma^{\beta}}{2}(\boldsymbol{\beta} - 1)^T \mathbf{M}(\boldsymbol{\beta} - 1) \\ \text{s.t.} \quad & \mathbf{R}(\mathbf{U}, \boldsymbol{\beta}) = \mathbf{0}, \end{aligned} \tag{19}$$

where $\mathcal{F}$ is either the output or distribution measure, and the additional term in the objective is a continuous Tikhonov regularization that makes the inverse problem well-posed by penalizing large deviations of the correction field from the nominal value of one. $\mathbf{M}$ is the mass matrix from the spatial discretization, and γ^{β} is a small user-prescribed parameter set to 10^{-5}, although a wide range of values was found to yield nearly identical results.

The field inversion problem is solved using 20 iterations of a limited-memory Broyden–Fletcher–Goldfarb–Shanno (L-BFGS) algorithm [17], with a history of 10 updates for approximating the inverse Hessian. Each iteration amounts to a steady RANS solution, which is not expensive as the solution from each iteration provides the initial guess for the state at the next iteration. An adjoint is used to evaluate the objective gradient, and the residual linearization with respect to $\boldsymbol{\beta}$ is calculated efficiently using finite differences and only a few residual evaluations [1].

C. Parameter Inversion

Turbulence and transition models have empirical parameters, which may be considered tunable to a certain degree. This is particularly true for the BCM model, which is algebraic and which exhibits a large sensitivity to the parameters c_{t1} and c_{t2}. Correction of these parameters can be incorporated into the inverse problem, without much additional expense since these are scalars, not fields. We apply a multiplicative correction to the parameters to keep the correction dimensionless, and hence invariant with respect to the units used. This assumes that the tunable parameters do not switch signs, which is typically not restrictive.

Let $\mathbf{P}$ denote the vector of multiplicative corrections applied to parameters considered tunable and participating in the inversion problem. The inversion objective in Eqn. 19 becomes

$$\begin{aligned} \min_{\boldsymbol{\beta}, \mathbf{P}} \quad & J^{\text{inv}} \equiv \mathcal{F}(\mathbf{U}(\boldsymbol{\beta}), \bar{\mathbf{U}}) + \frac{\gamma^{\beta}}{2}(\boldsymbol{\beta} - 1)^T \mathbf{M}(\boldsymbol{\beta} - 1) + \frac{\gamma^{P}}{2}(\mathbf{P} - 1)^T (\mathbf{P} - 1) \\ \text{s.t.} \quad & \mathbf{R}(\mathbf{U}, \boldsymbol{\beta}, \mathbf{P}) = \mathbf{0} \end{aligned} \tag{20}$$

The regularization for $\mathbf{P}$, γ^{P}, may differ from that on $\boldsymbol{\beta}$, γ^{β}, although presently we take these as equal. The result of solving Eqn. 20 is the correction field together with scalar values for the parameters. We take these values as locally valid and hold them fixed, as opposed to $\boldsymbol{\beta}$ for which we create a machine-learning model that is a function of the state.

IV. Machine Learning

A. Overview

The correction field obtained from field inversion is static and specific to the geometry and flow conditions used in the inversion. We write it as $\beta(\vec{x})$ to denote that it only depends on the spatial coordinates. In applications such as design optimization and mesh adaptation, we are interested in changes to the geometry and discretization, and hence a static correction field loses utility. We therefore seek a model for β in terms of local flow information, similarly to other terms in the turbulence model.

We model the mapping between local state information and the correction field using an artificial neural network [1, 18–21]. The inputs are the local state ($\mathbf{u}$), its gradient ($\nabla\mathbf{u}$), and the wall distance function (d). In choosing these inputs, which are not scale invariant, we do not aim to derive a general-purpose turbulence model. We instead seek to create a locally-valid corrected model that reproduces the behavior of a higher-fidelity model at geometries, meshes, and conditions that are close to the training case. We expect to retrain the model as an optimization or adaptation progresses, and the utility of the model comes from its ability to yield accurate gradients – this is what we are assessing in the present study of transitional flows.

B. Network Architecture

The neural network maps local flowfield information to the scalar correction field. Figure 1 shows the structure of the network used in this work, which is a single-hidden-layer perceptron. The hidden layer, $\mathbf{x}_1$, contains n_1 neurons, between the input layer, i.e. the features, $\mathbf{x}_0$, and the output layer, which consists of the scalar correction factor, β. The map from the input to the hidden layer involves an entry-wise sigmoid activation function, $\sigma(x) = 1/(1 + e^{-x})$, whereas no activation function is used for the output layer calculation. The parameters associated with the network consist of the weights and biases, $\mathbf{W}_i \in \mathbb{R}^{n_i \times n_{i-1}}$, $\mathbf{b}_i \in \mathbb{R}^{n_i}$, where n_i is the number of neurons in layer i.

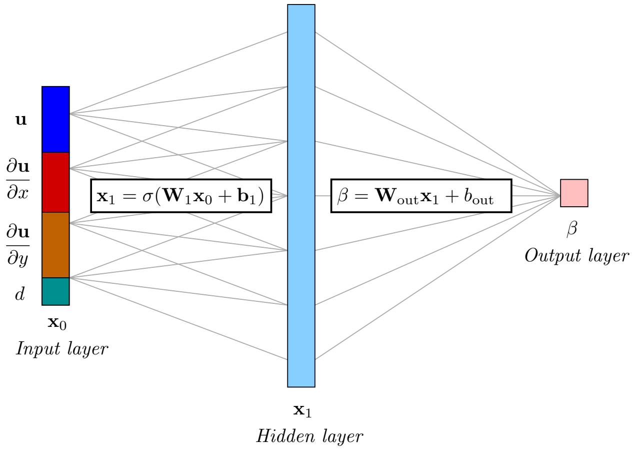


Fig. 1 Structure of the artificial neural networks used to predict the correction field.

The input into the network consists of the state, $\mathbf{u}$, its gradient, $\nabla\mathbf{u}$, and the wall distance, d, for a total of $(1+\text{dim})s+1$ neurons. Based on results of our previous studies [1], the size of the hidden layer was set to $n_1 = 30$. The weights and biases of the network were trained using the adaptive moment (Adam) algorithm in TensorFlow [22]. The true β values for training come from the field inversion correction field for one simulation: β is sampled at the quadrature points of the elements. The training data are broken into mini-batches of size 1000 for the optimizer, and the learning rate is set to .001. Prior to training, the weights and biases are initialized randomly from a unit normal distribution. 500,000 optimization iterations are taken in each training session for all of the presented results, although the mean-squared error typically stabilizes well before this number. Two to three orders of magnitude drop in the loss are usually observed.

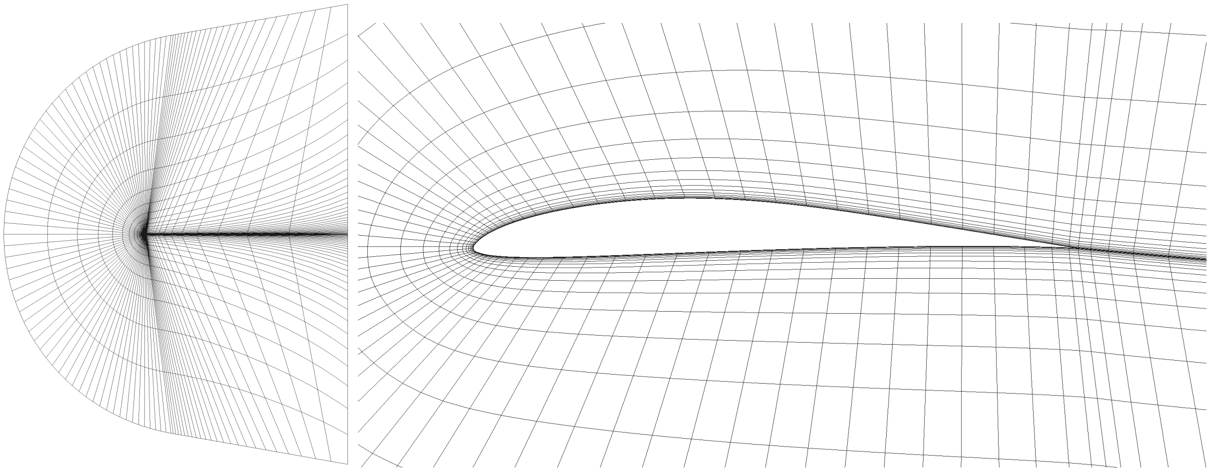
C. Implementation

Once a network is trained, it is implemented as a physics model in the RANS turbulence source calculation. No changes to the inputs to this calculation are required, since the turbulent source already uses the state, its gradient, and the wall distance function. The network is differentiated with respect to the state and its gradient for the Newton solver and the discrete adjoint solver. Due to the nonlinear nature of the network, higher-order accurate quadrature rules are used for the corrected RANS model. Presently, a constant order increment of 2 is added to the baseline $2p + 1$ integration order requirement in the code.

V. Results

This section presents results obtained when using the SA-BCM model in an FIML setting for an Eppler 387 airfoil, for which experimental data are available at various Reynolds numbers [23]. For the field inversion objective, both experimental and unsteady numerical data are used. Comparisons are made between FIML applied to SA versus SA-BCM, in terms of the accuracy of the corrected state and surface distributions, and the magnitude of the correction. We use a free-stream SA working variable of $\tilde{\nu}_\infty = 3\nu$ for the fully-turbulent SA model and $\tilde{\nu}_\infty = 0.1\nu$ for the SA-BCM model [7].

Figure 2 illustrates the computational mesh used in all calculations, which consists of 3584 curved $Q = 3$ quadrilateral elements and extends approximately 100 chord lengths away from the airfoil. The approximation order used is $p = 2$, and the freestream Mach number is 0.1.



(a) Farfield view

(b) Zoom near airfoil

Fig. 2 Mesh used for the Eppler 387 airfoil simulations.

A. Eppler Airfoil, $Re = 200k$

We first perform tests at a Reynolds number of $Re = 200k$. Running the baseline RANS model (SA) and the transitional model (SA-BCM) at different angles of attack produces the lift and drag polars shown in Figure 3. We focus on moderate angles of attack, but not stall, in which both models do not perform well, in part due to their steady nature compared to the large-scale unsteadiness of stall. The results indicate that both models are accurate at predicting the lift, which is not overly difficult when the flow is attached. On the other hand, they generally over-predict the drag, albeit for different reasons: the fully-turbulent SA model over-predicts skin friction drag as it misses the laminar flow before transition, while the SA-BCM model over-predicts pressure drag due to the flow not transitioning to turbulence sufficiently quickly, leaving a larger region of separated flow.

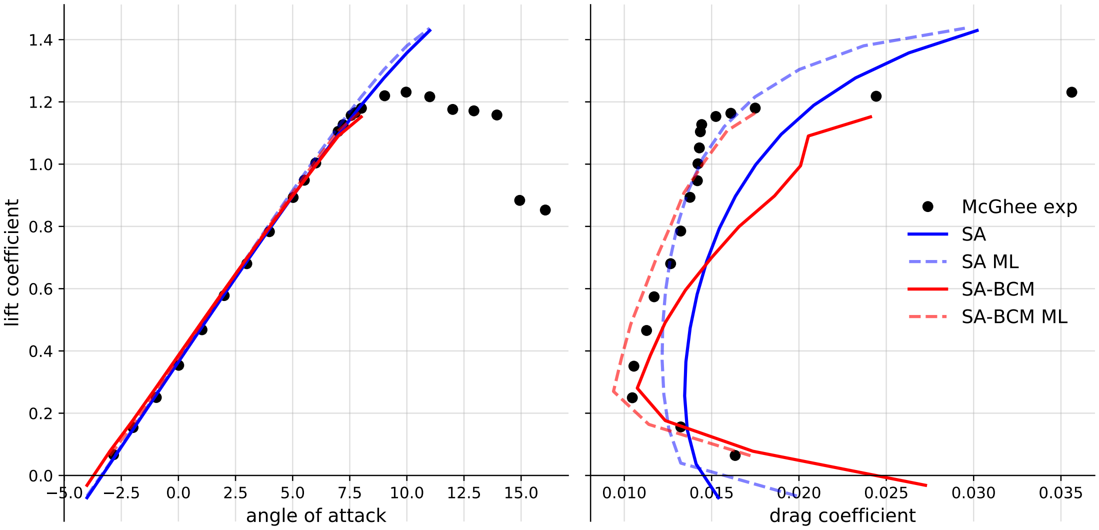
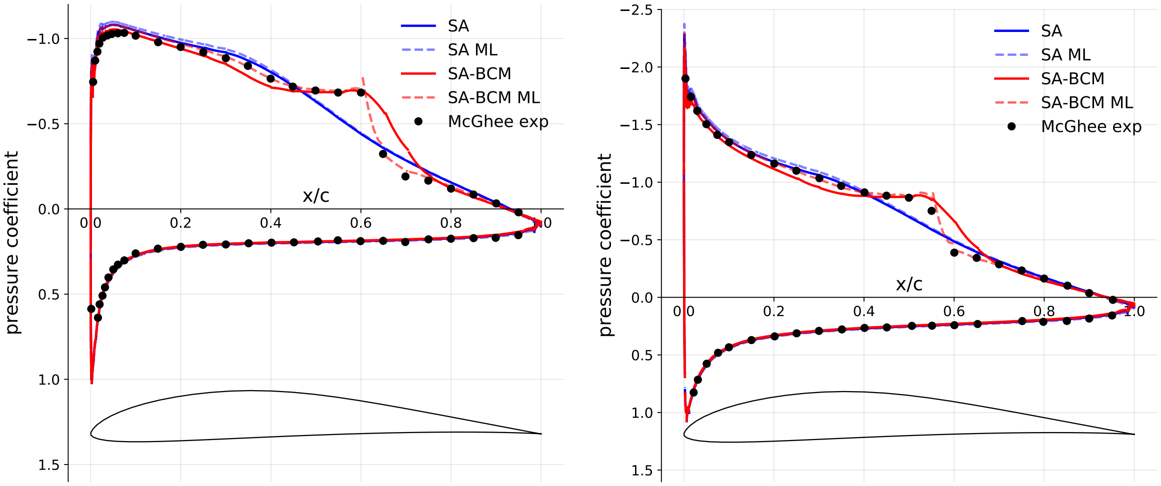


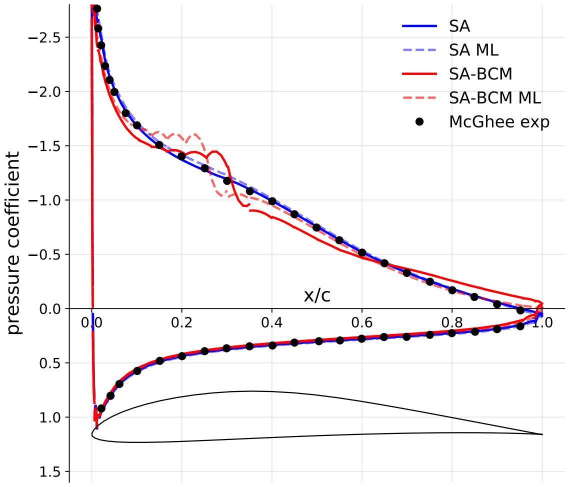
Fig. 3 Eppler airfoil lift and drag polar at $Re = 200{,}000$.

For both the SA and SA-BCM models, we run field inversion at $\alpha = 6^\circ$, using the experimental drag coefficient as the objective, with regularization, $\gamma^\beta = 10^{-5}$. We then generate machine-learning models using the baseline network architecture: 1 hidden layer of 30 neurons. These corrected models, SA-β and SA-BCM-β, are used in the same angle of attack sweep, and Figure 3 includes these results. The lift coefficient does not change much and still agrees with the experiment. The drag coefficient decreases as expected given the lower objective drag for the field inversion. Both models predict the experimental/target c_d at $\alpha = 6°$ well – we note that exact agreement is not guaranteed, due to differences between the machine learning and field inversion corrections. At larger angles of attack, neither model does a good job at predicting the drag dip, which is caused in the experiments by a decrease in pressure drag due to the disappearance of the laminar separation bubble (LSB). However, at lower angles of attack, the SA-BCM results follow the experimental drag polar better, in value and particularly the slope, than SA alone.



(a) $\alpha = 4°$

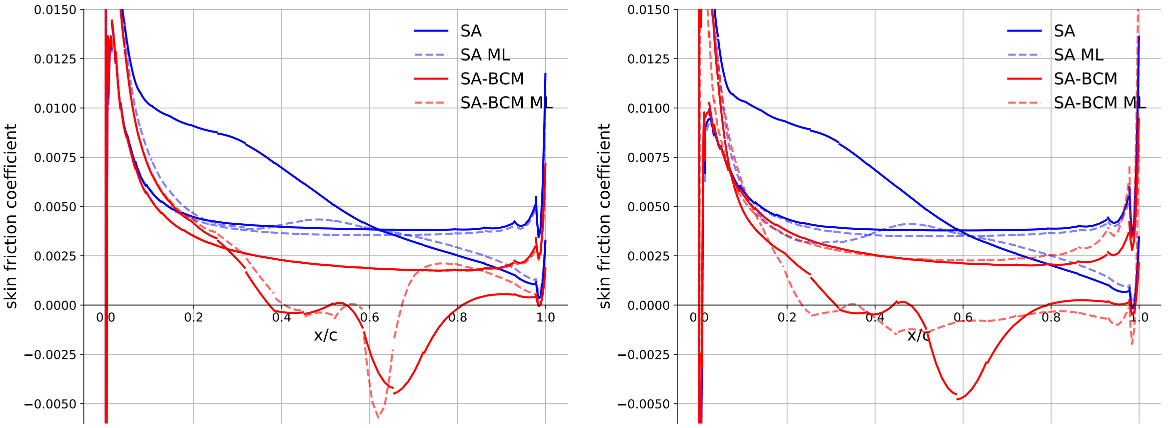
(b) $\alpha = 6°$



(c) $\alpha = 8°$

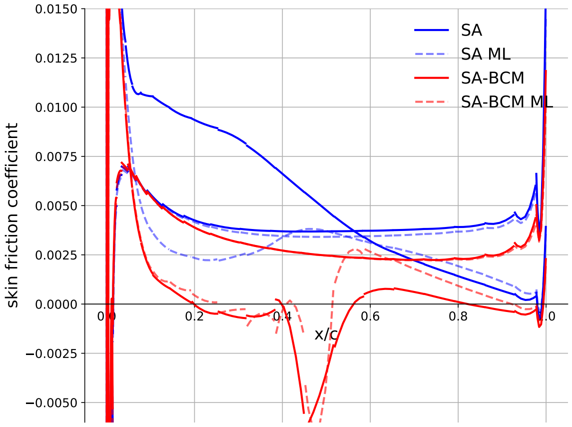
Fig. 4 Eppler airfoil pressure coefficient distributions at $Re = 200{,}000$. The corrected (β) results all use the machine-learning model obtained from field inversion at $\alpha = 6°$.

To further compare the results, we look at the pressure coefficient distributions, shown in Figure 4 for three angles of attack. Transition appears in these plots due to the effect of the LSB on the pressure coefficient. The SA (fully turbulent) model does not predict transition, and its correction barely changes the c_p – it does affect the skin friction, however. The SA-BCM model without correction transitions to turbulence slightly too late, and too slowly at both of the lower angles of attack. Correction moves this transition earlier, virtually on top of the experimental data, through higher β values near the transition location, and this results in lower pressure drag values for SA-BCM-β. We observe improvement at $\alpha = 4°$ even though the inversion and ML training were performed at $\alpha = 6°$, and this illustrates that the correction remains valid locally. The extension to the larger angle of attack, $\alpha = 8°$, is not as accurate due to a regime change: the experimental results no longer show the laminar separation bubble, whereas SA-BCM-β still predicts it.



(a) $\alpha = 4°$

(b) $\alpha = 6°$



(c) $\alpha = 8°$

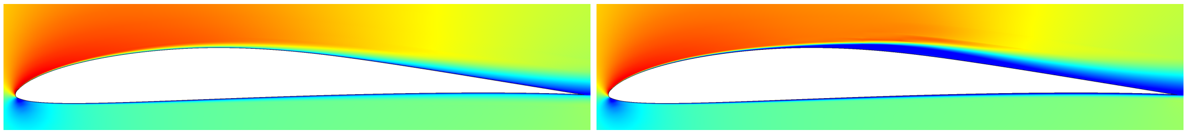
Fig. 5 Eppler airfoil skin friction coefficient distributions at $Re = 200{,}000$. The corrected (β) results all use the machine-learning model obtained from field inversion at $\alpha = 6°$.

The skin friction results in Figure 5 indicate a large drop in c_f for SA-β compared to SA alone, caused by suppression of turbulent production (low β) in what is supposed to be the laminar region. The resulting profiles still differ from the transition model, as they are without an LSB or transition. Only the average c_f, together with the pressure distribution, are consistent with the drag objective, as enforcing matching of an integrated quantity does not guarantee matching of the distribution. For SA-BCM-β, the shift in the transition location on the upper surface is apparent and consistent with the c_p result.

Finally, Figure 6 shows the Mach number and β field plots. The fully-turbulent SA Mach number contours indicate a thin boundary layer without transition. The correction in SA-β reduces the skin friction through low β values, particularly on the front upper portion of the airfoil. The effect on the boundary layer thickness is minimal, and hence the negligible change observed in the c_p distribution. On the other hand, the SA-BCM boundary layer is thicker after the laminar separation. The correction consists of a high β at transition to accelerate the reattachment. This leads to a smaller LSB in SA-BCM-β, and hence less pressure drag.

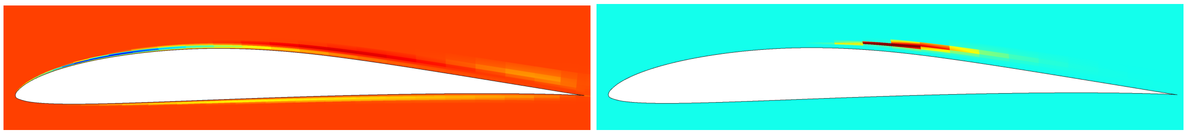
B. Eppler Airfoil, $Re = 60k$

As a second test case, we consider the same Eppler 387 airfoil at a lower Reynolds number, $Re = 60k$, and one angle of attack, $\alpha = 8°$. The flow is now more likely to separate earlier, and transition is important in keeping the flow



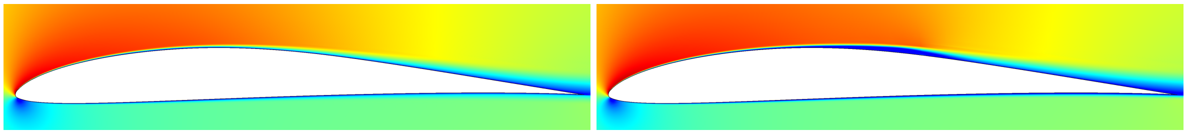
(a) SA: Mach (0, 0.18)

(b) SA-BCM: Mach (0, 0.18)



(c) SA: β (-0.3, 1.3)

(d) SA-BCM: β (0, 3)



(e) SA-β: Mach (0, 0.18)

(f) SA-BCM-β: Mach (0, 0.18)

Fig. 6 Eppler airfoil Mach and correction-field contours at Re = 200,000 and α = 6°.

attached. In addition to experimental data [23], we also generate higher-fidelity numerical data by running an unsteady two-dimensional simulation using the Navier–Stokes equations without any additional models. The pressure distribution from the time-average of the unsteady state actually agrees very well with the experimental data, as shown in Figure 7. This agreement suggests that at this Reynolds number, the dominant transition mechanism is well-approximated by a two-dimensional flow assumption.

Figure 7 also shows the results of the SA and SA-BCM models for this case. SA does well at predicting the average pressure distribution but misses the LSB and transition, as it assumes fully-turbulent flow. SA-BCM does quite poorly: the flow separates in the laminar portion and never re-attaches. Hence, the pressure distribution is vastly different from the other results.

Figure 8 shows the Mach number and correction factor field plots for the various simulations. First, the unsteady plots illustrate the breakdown of the laminar flow and transition at approximately the half-chord location. The SA boundary layer remains thin due to the fully-turbulent assumption, whereas the SA-BCM flowfield exhibits massive separation on the upper surface due to transition not occurring sufficiently early.

In this case, due to the availability of the time-averaged unsteady flowfield data, the airfoil surface stress distribution is used for the field inversion objective. This is more information than just the drag alone, which was used in the previous set of results. Hence, we expect better agreement in the corrected flowfield. Indeed, both Figures 7 and 8 show that this is the case. The pressure distributions for SA-β and SA-BCM-β are nearly identical to the unsteady data, which was the target. For SA-BCM, we observe a massive difference in the flowfield, as the correction causes reattachment and a large change in the pressure distribution. Note the large values of β near the transition location that are responsible for the earlier/faster transition and associated re-attachment. For SA, the change in the flowfield is also noticeable, as the LSB appears. This is caused by suppression of turbulent production, with β playing the role of the transition model intermittency, followed by an increase in β at the desired transition location.

In this case, we see that field inversion produces good results with appropriate corrections for both SA and SA-BCM. A practically-critical question, however, is whether this correction can be learned and computed from the local state information. And this is where we observe failure, for both SA and SA-BCM. The first red flag occurs during the neural network training: the loss function only drops an order of magnitude for the baseline small network consisting a single 30-neuron hidden layer. Increasing the size and/or number of hidden layers helps with the training loss. However, when subsequently used online, these larger networks fail to produce a converged flow solution.

Both SA and SA-BCM fail, but for different reasons. In the case of SA, it is difficult to learn the sudden low-to-high swing in β at transition. This is because the state and wall distance do not change much over a small chord-wise distance. Arguably, by the approximation properties of sufficiently-large neural networks, it should be possible to create a correction function neural network model that produces the desired transition behavior: indeed, this is what the

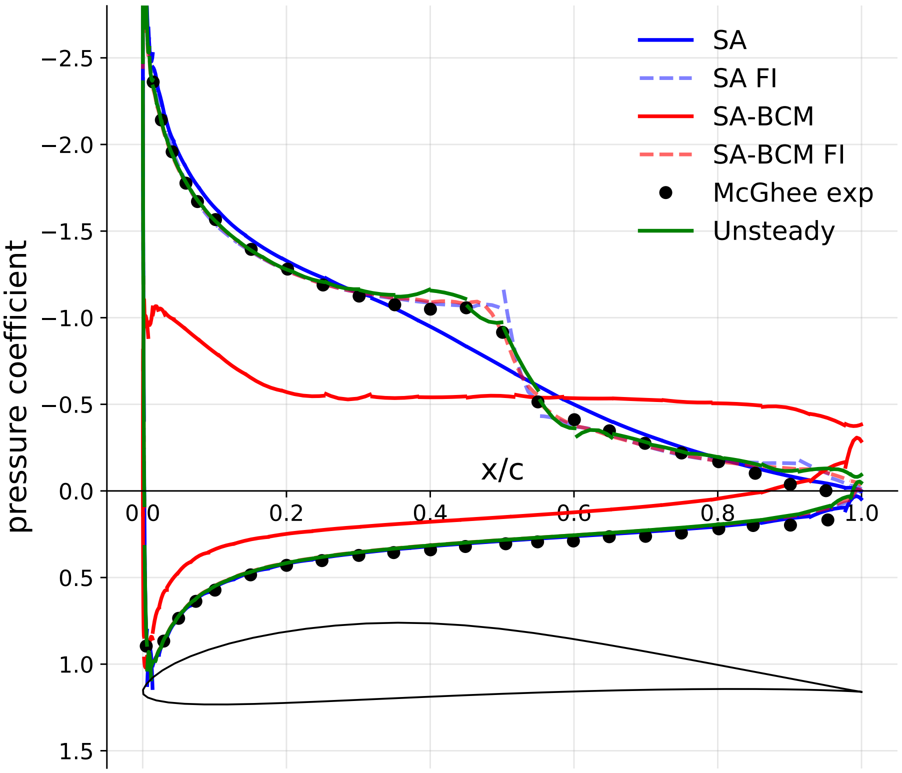
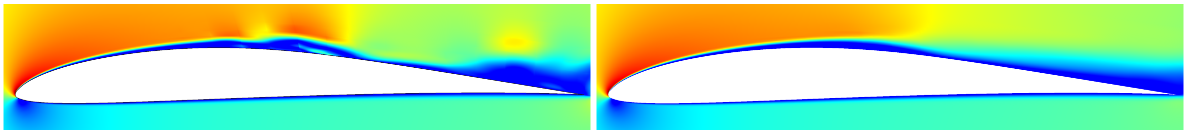
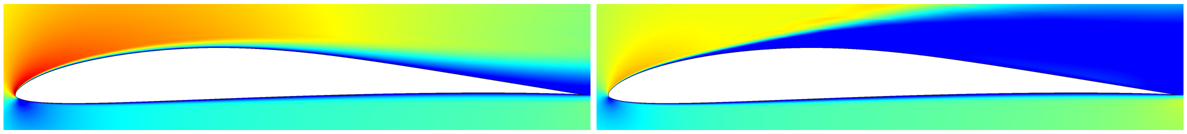


Fig. 7 Eppler airfoil pressure coefficient distributions at Re = 60,000, α = 8°.



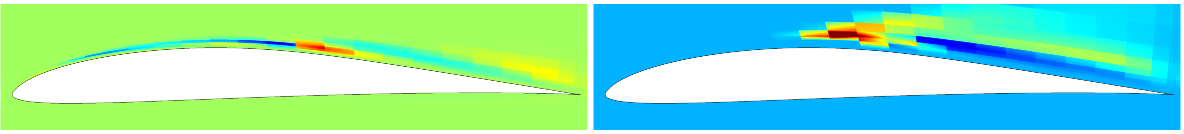
(a) Unsteady: instantaneous state Mach (0, 0.2)

(b) Unsteady: time-averaged state Mach (0, 0.2)



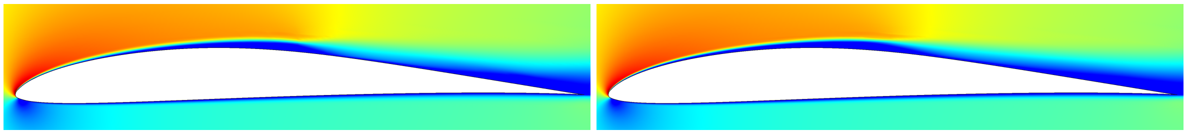
(c) SA: Mach (0, 0.2)

(d) SA-BCM: Mach (0, 0.2)



(e) SA: β (-15, 15)

(f) SA-BCM: β (-5, 15)



(g) SA-β: Mach (0, 0.2)

(h) SA-BCM-β: Mach (0, 0.2)

Fig. 8 Eppler airfoil Mach and correction-field contours at Re = 60,000 and α = 8°.

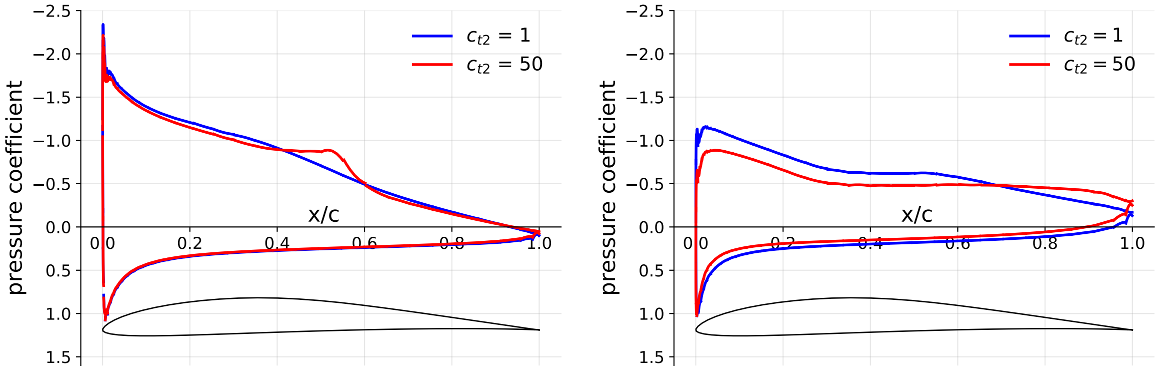
algebraic intermittency model does. However, the small changes in the flow state are drowned out by numerical errors, both from the discretization and the field inversion. Hence, the correction field obtained is not suitable for approximation with an ML model.

In the case of SA-BCM, we see massive separation without correction due to a large region of laminar flow caused by the intermittency staying low for too long. As the β correction is multiplicative, it cannot effectively correct the model in the region of low intermittency. Instead, the recourse is a massive increase in the production (large β) around the transition location that re-attaches the flow, followed by negative β downstream to undo some of this production. This extreme low-to-high-to-low swing in β is difficult to learn and prevents a stable model. In addition, the task of re-attaching the flow poses solver continuation challenge.

C. Transition Model Parameter Sensitivity

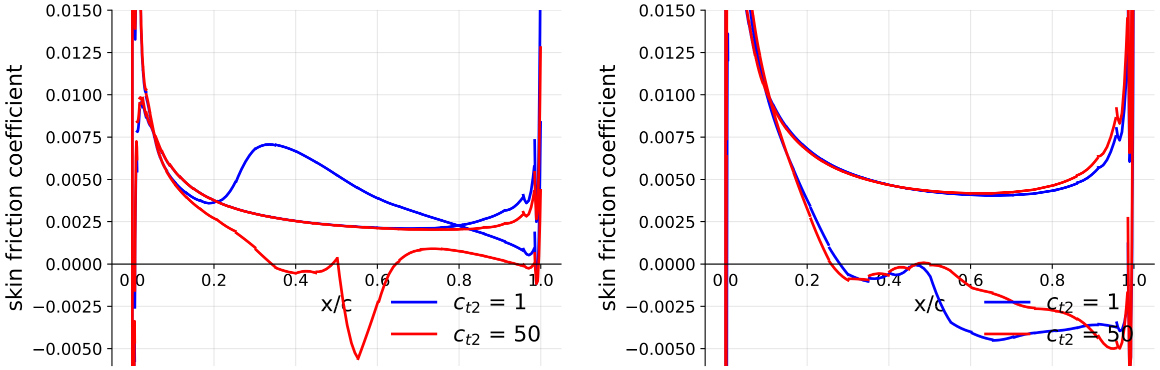
The machine-learning stage failed in the previous example because of inadequacy in the baseline model. SA-BCM with the default parameters did not predict the flow well for the particular test case of $Re = 60k$, $\alpha = 8°$. However, by changing the parameters, we can obtain improved performance. In particular, the c_{t2} parameter that appears in Eqn. 13 has a strong effect on the location of transition. If it is not set appropriately for the case being simulated, the flowfield may be incorrect to a sufficiently high degree that the field inversion yields large-magnitude, unlearnable corrections.

Figure 9 shows the effect of varying c_{t2} on the solutions for the Eppler airfoil at $\alpha = 6°$ for two Reynolds numbers, $Re = 200k$ and $Re = 60k$. We see that for $Re = 200k$, the low value of $c_{t2} = 1$ makes the flow transition earlier, eliminating the separation bubble and reducing the drag by about 10% relative to $c_{t2} = 50$. In the $Re = 60k$ case, the lower c_{t2} value moves the transition earlier, causing a 20% change in drag, but does not eliminate the separation on the upper surface.



(a) $Re = 200k$, c_p

(b) $Re = 60k$, c_p

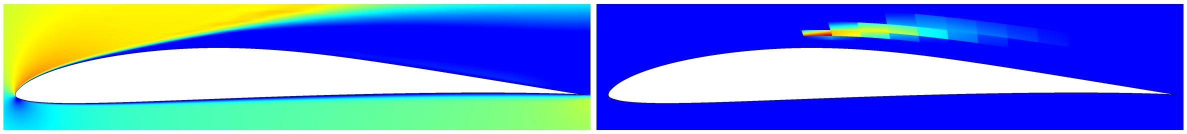


(c) $Re = 200k$, c_f

(d) $Re = 60k$, c_f

Fig. 9 Effect of the BCM parameter c_{t2} on Eppler airfoil solutions at $\alpha = 6°$.

Although changing c_{t2} from 50 to 1 has a minor effect on the flow solution at $Re = 60k$, this change is sufficient to offload some of the correction from the β field and make the machine learning of the dynamic correction possible. Figure 10 shows the initial (uncorrected) Mach number contours and the correction field after field inversion for $c_{t2} = 1$. Compared to the β contours in Figure 8, we now see significantly smaller-magnitude corrections.



(a) SA-BCM: Mach (0, 0.2)

(b) SA-BCM: β (0.9, 3.7)

Fig. 10 Eppler airfoil Mach and correction-field contours at $Re = 60{,}000$ and $\alpha = 8°$, using $c_{t2} = 1$.

Figure 11 shows the corresponding field inversion and machine learning results when using $c_{t2} = 1$. Unlike in the case of $c_{t2} = 50$, machine learning is now possible with the BCM model, and hence this result is shown in the figure. It agrees well with the experimental data and the time-averaged unsteady result. Correction of the baseline SA model without the BCM transition model remains unlearnable for machine learning, due to large swings in β in regions where the state does not change much.

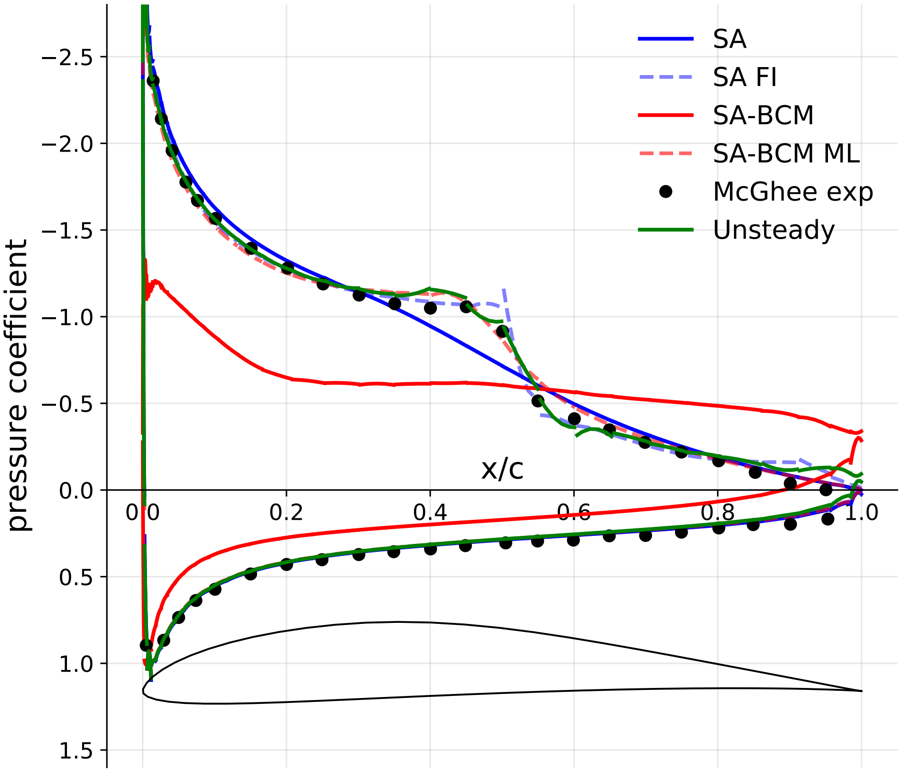


Fig. 11 Eppler airfoil pressure coefficient distributions at $Re = 60{,}000$, $\alpha = 8°$, using $c_{t2} = 1$.

Applying the field inversion and parameter optimization approach presented in Section III.C, we obtain c_{t2} of approximately 1 for the $Re = 60k$, $\alpha = 8°$ case. The correction field also has smaller magnitudes, similarly to what is shown in Figure 10. This result demonstrates the ability of the combined field/parameter inversion to produce dynamically-corrected models with small-magnitude β fields that are amenable to learning as functions of the local state.

VI. Conclusions

This paper extends field inversion and machine learning to transitional flows. The governing equations are RANS with an intermittency multiplier on the production term. The intermittency is a function of the local state through an algebraic expression, so that no additional equations are needed in its calculation. This model, SA-BCM, can predict transition but has tunable constants, in particular the freestream turbulent intensity and a factor that scales the SA working variable. The effect of these tunable parameters can be mitigated to an extent via turbulent production correction, β, used in FIML. The present results for the Eppler 387 airfoil indicate that the corrected SA-BCM results are more accurate than the corrected SA results when inverting on an integrated quantity, such as the drag coefficient. This improvement includes more accurate integrated quantities at nearby conditions, and more accurate surface distributions. When inverting on distribution data, e.g. surface stress, the SA correction is not easily learnable from the local data, due to the need for a large change in the correction from small changes in the state. The SA-BCM correction shows more promise due to the built-in transition model, but the success of the field inversion and machine learning still depends on the applicability of the model, as dictated by its tunable factors. The multiplicative combination of the intermittency and the FIML correction prevents the field production correction from moving the transition location when the intermittency suppresses production. Hence, in addition to solving for a correction field, we have incorporated parameter optimization into the inversion problem. Optimized parameters offload work from the correction field, which then becomes smaller in magnitude and more easily learnable in the machine-learning stage.

References

[1] Fidkowski, K. J., "Gradient-based shape optimization for unsteady turbulent simulations using field inversion and machine learning," *Aerospace Science and Technology*, Vol. 129, 2022, p. 107843. https://doi.org/https://doi.org/10.1016/j.ast.2022.107843.

[2] Fidkowski, K. J., "Output-based error estimation and mesh adaptation for unsteady turbulent flow simulations," *Computer Methods in Applied Mechanics and Engineering*, Vol. 399, 2022, p. 115322. https://doi.org/https://doi.org/10.1016/j.cma.2022.115322.

[3] Cakmakcioglu, S. C., Bas, O., and Kaynak, U., "A correlation-based algebraic transition model," *Journal of Mechanical Engineering Science*, Vol. 232, No. 21, 2018, pp. 3915–3929. https://doi.org/10.1177/0954406217743537.

[4] Cakmakcioglu, S. C., Bas, O., Mura, R., and Kaynak, U., "A Revised One-Equation Transitional Model for External Aerodynamics," AIAA Paper 2020-2706, 2009. https://doi.org/10.2514/6.2020-2706.

[5] Allmaras, S., Johnson, F., and Spalart, P., "Modifications and Clarifications for the Implementation of the Spalart-Allmaras Turbulence Model," Seventh International Conference on Computational Fluid Dynamics (ICCFD7) 1902, 2012.

[6] Fidkowski, K. J., "Three-Dimensional Benchmark RANS Computations Using Discontinuous Finite Elements on Solution-Adapted Meshes," AIAA Paper 2018–1104, 2018. https://doi.org/10.2514/6.2018-1104.

[7] Crivellini, A., Ghidoni, A., and Noventa, G., "Algebraic modifications of the $k-\tilde{\omega}$ and Spalart-Allmaras turbulence models to predict bypass and separation-induced transition," *Computers and Fluids*, Vol. 253, 2023, p. 105791. https://doi.org/10.1016/j.compfluid.2023.105791.

[8] Fidkowski, K. J., "Output error estimation strategies for discontinuous Galerkin discretizations of unsteady convection-dominated flows," *International Journal for Numerical Methods in Engineering*, Vol. 88, No. 12, 2011, pp. 1297–1322. https://doi.org/10.1002/nme.3224.

[9] Roe, P., "Approximate Riemann solvers, parameter vectors, and difference schemes," *Journal of Computational Physics*, Vol. 43, 1981, pp. 357–372. https://doi.org/https://doi.org/10.1016/0021-9991(81)90128-5.

[10] Bassi, F., and Rebay, S., "Numerical evaluation of two discontinuous Galerkin methods for the compressible Navier-Stokes equations," *International Journal for Numerical Methods in Fluids*, Vol. 40, 2002, pp. 197–207. https://doi.org/https://doi.org/10.1002/fld.338.

[11] Ceze, M. A., and Fidkowski, K. J., "Constrained pseudo-transient continuation," *International Journal for Numerical Methods in Engineering*, Vol. 102, 2015, pp. 1683–1703. https://doi.org/10.1002/nme.4858.

[12] Saad, Y., and Schultz, M. H., "GMRES: A Generalized Minimal Residual Algorithm for Solving Nonsymmetric Linear Systems," *SIAM Journal on Scientific Computing*, Vol. 7, No. 3, 1986, pp. 856–869. https://doi.org/https://doi.org/10.1137/0907058.

[13] Fidkowski, K. J., Oliver, T. A., Lu, J., and Darmofal, D. L., "p-Multigrid solution of high–order discontinuous Galerkin discretizations of the compressible Navier-Stokes equations," *Journal of Computational Physics*, Vol. 207, 2005, pp. 92–113. https://doi.org/10.1016/j.jcp.2005.01.005.

[14] Persson, P.-O., and Peraire, J., "Newton-GMRES Preconditioning for Discontinuous Galerkin Discretizations of the Navier-Stokes Equations," *SIAM Journal on Scientific Computing*, Vol. 30, No. 6, 2008, pp. 2709–2733. https://doi.org/https://doi.org/10.1137/070692108.

[15] Cash, J., "The integration of stiff initial value problems in ODEs using modified extended backward differentiation formulae," *Computers & mathematics with applications*, Vol. 9, No. 5, 1983, pp. 645–657. https://doi.org/https://doi.org/10.1016/0898-1221(83)90122-0.

[16] Fidkowski, K. J., and Darmofal, D. L., "Review of Output-Based Error Estimation and Mesh Adaptation in Computational Fluid Dynamics," *AIAA Journal*, Vol. 49, No. 4, 2011, pp. 673–694. https://doi.org/10.2514/1.J050073.

[17] Liu, D. C., and Nocedal, J., "On the Limited Memory BFGS Method for Large Scale Optimization," *Mathematical Programming*, Vol. 45, 1989, pp. 503 – 528. https://doi.org/10.1007/BF01589116.

[18] 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, pp. 2215–2227. https://doi.org/10.2514/1.j055595.

[19] Holland, J. R., Baeder, J. D., and Duraisamy, K., "Towards Integrated Field Inversion and Machine Learning With Embedded Neural Networks for RANS Modeling," AIAA Paper 2019-1884, 2019. https://doi.org/10.2514/6.2019-1884.

[20] Ho, J., and West, A., "Field Inversion and Machine Learning for turbulence modelling applied to three-dimensional separated flows," AIAA Paper 2021–2903, 2021. https://doi.org/https://doi.org/10.2514/6.2021-2903.

[21] Jäckel, F., "A Closed-form Correction for the Spalart-Allmaras Turbulence Model for Separated Flow," AIAA Paper 2022–0462, 2022. https://doi.org/10.2514/6.2022-0462.

[22] Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., and Zheng, X., "TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems," , 2015. URL http://tensorflow.org/, software available from tensorflow.org.

[23] McGhee, R. J., Walker, B. S., and Millard, B. F., "Experimental Results for the Eppler 387 Airfoil at Low Reynolds Numbers in the Langley Low-Turbulence Pressure Tunnel," NASA Technical Memorandum 4062, 1988.