Large Eddy Simulation of Airfoil Flows Using Adjoint-Trained Deep Learning Closure Models

Tom Hickling* and Justin Sirignano†
University of Oxford, UK

Jonathan F. MacArt‡
University of Notre Dame, USA

There is a wide interest in developing methods that reduce the cost of resolving the near-wall and small-scale turbulence in order to enable the use of turbulence-resolving methods (such as large eddy simulation) at higher, more industrially relevant, Reynolds numbers. Existing techniques for this can struggle to accurately predict scenarios with complex flow physics such as transition, large-scale unsteadiness, and smooth-body separation/reattachment. Deep-learning based subgrid-scale (DL-SGS) models may help address this. However, the standard a priori approach for training these is unable to account for interactions of the DL-SGS closure with the numerics and resolved physics. Optimizing the DL-SGS closure over the governing equations using the adjoint equation has previously been found to improve predictions and stability in incompressible flows. In this paper, an adjoint training method is developed for compressible LES, and is applied the flow around a NACA 0012 airfoil at $\mathrm{Re}_c = 5 \times 10^4$ and $\mathrm{Ma} = 0.4$. The DL-SGS closure is trained on a single angle of attack, and improves predictions compared to classical SGS models when extrapolating to out-of-sample angles of attack, even when it has not been trained over fully representative flow physics.

I. Introduction

Turbulence-resolving methods such as large eddy simulation (LES) are increasingly being used to enable accurate predictions in flow regimes containing laminar-to-turbulent transition and smooth-body separation. However, the utility of fully resolved LES remains limited due to the significant cost incurred by mesh resolution requirements, particularly in near-wall regions of the flow [1]. Including the effects of pressure gradients, transition, and heat transfer on the subgrid-scale (SGS) is challenging, particularly near the wall, and attempts at it are not widespread [2]. It is therefore important to develop new SGS closure models which will yield accurate coarse-mesh (in both near-wall and bulk) LES simulations of the complex near-wall flow. Our objective is to use deep learning to develop coarse-mesh LES models which have an accuracy comparable to highly resolved LES or direct numerical simulation (DNS).

Recently, interest has grown in using deep learning for SGS model closures [3]. Most efforts in this direction train the deep learning SGS (DL-SGS) model in an a priori manner, where the SGS closure parameters are estimated offline from filtered DNS data without optimising over the governing equations [4–7]. Although simple to implement with existing supervised machine learning methods, a priori training of the SGS closure has two key disadvantages: optimisation does not commute with a nonlinear function, introducing errors, and the training procedure does not account for numerical errors, which can be significant for a coarse grid LES—even with a "perfect" SGS model derived by coupling to DNS [8]. These issues have been observed to negatively affect both accuracy [9, 10] and even stability [11] of DL-SGS closures, and may play a part in their limited application to only relatively simple canonical flows such as isotropic turbulence, jets, and turbulent channel flows.

To address this limitation, an adjoint-training method has recently been proposed that allows the optimisation to be carried out over the governing equations [9–11]. This has been demonstrated to have superior performance to existing classical and a priori trained DL-SGS models in decaying isotropic turbulence [11], turbulent plane jets [10], and flow over bluff bodies [9].

Motivated by the success of the adjoint-trained deep learning LES methodology in incompressible flows, this paper will develop adjoint training methods and DL-SGS models for compressible flow around low to moderate Reynolds

*Mathematical Institute, University of Oxford, Oxford, UK. tom.hickling@maths.ox.ac.uk.

†Mathematical Institute, University of Oxford, Oxford, UK.

‡Department of Aerospace and Mechanical Engineering, University of Notre Dame, Notre Dame, USA. Member AIAA.

number airfoils. The fluid mechanics here are more complex than previous applications of the methodology—accurately capturing the near-wall behaviour of the laminar separation bubble and turbulent reattachment is important for an accurate prediction [12], especially in a coarse-mesh LES. Reynolds-averaged Navier–Stokes (RANS) models are not considered in this paper, as they consistently struggle with this type of flow due to the large-scale unsteadiness and laminar–turbulent transition [13].

In the remainder of this paper, we will first present the governing equations solved by PyFlowCL, our in-house curvilinear finite-difference solver used throughout this work, along with the mathematical background for our adjoint optimization methodology. We will then detail our DNS testing and training data set, along with some validation against existing DNS simulations at the same conditions in the literature. Next, we discuss the computational set up for our model training and a posteriori simulations, before finally assessing the performance of the DL-SGS model relative to some classical SGS modeling approaches in long time scale a posteriori simulations at various angles of attack.

II. Governing Equations and Methodology

A. Governing Equations and General Approach

The LES equations considered in this paper are derived from the non-dimensional compressible Navier–Stokes equations with an ideal gas equation of state. They are given by

$$\begin{aligned} \frac{\partial \rho}{\partial t} + \frac{\partial \rho u_j}{\partial x_j} &= 0, \\ \frac{\partial \rho u_i}{\partial t} + \frac{\partial \rho u_i u_j}{\partial x_j} + \frac{1}{\mathrm{Ma}^2}\frac{\partial p}{\partial x_i} - \frac{1}{\mathrm{Re}}\frac{\partial \sigma_{ij}}{\partial x_j} &= 0, \\ \frac{\partial \rho E}{\partial t} + \frac{\partial \rho u_j E}{\partial x_j} + \frac{1}{\mathrm{Ma}^2}\frac{\partial p u_j}{\partial x_j} - \frac{1}{\mathrm{Re}}\frac{\partial u_i \sigma_{ij}}{\partial x_j} + \frac{1}{\mathrm{Ma}^2\mathrm{RePr}}\frac{\partial q_j}{\partial x_j} &= 0, \end{aligned} \tag{1}$$

where $x \in \Omega$ is the simulation domain, ρ is the density, u_i is the Cartesian velocity vector, Re, Ma, and Pr are the scaling Reynolds, Mach, and Prandtl numbers respectively, and the viscous stress tensor and the heat-flux vector are $\sigma = \mu(T)(\nabla\mathbf{u} + \nabla\mathbf{u}^\top - \frac{2}{3}(\nabla\cdot\mathbf{u})\mathbf{I})$ and $\mathbf{q} = -\mu\nabla T$. A power-law dependence on temperature is assumed for the viscosity μ. The PDE system (Eq. (1)) is closed using the ideal gas law and by assuming that the gas is calorically perfect. It is completed with appropriate boundary conditions for $x \in \partial\Omega$.

The unweighted and Favre (density-weighted) filtering operations are defined as

$$\overline{\phi} \equiv \int_\Omega G(\mathbf{r},\mathbf{x})\phi(t,\mathbf{x}-\mathbf{r})\,\mathrm{d}\mathbf{r} \qquad \text{and} \qquad \widetilde{\phi} \equiv \frac{1}{\overline{\rho}}\int_\Omega G(\mathbf{r},\mathbf{x})\rho(t,\mathbf{x}-\mathbf{r})\phi(t,\mathbf{x}-\mathbf{r})\,\mathrm{d}\mathbf{r}, \tag{2}$$

respectively, where $G(\mathbf{r},\mathbf{x})$ is an appropriately normalized LES filter kernel. (In most cases this is unknown and determined implicitly by the LES grid.) Applying the filters in Eq. (2) to the compressible Navier–Stokes equations (Eq. (1)) results in the compressible LES equations,

$$\begin{aligned} \frac{\partial \overline{\rho}}{\partial t} + \frac{\partial \overline{\rho}\widetilde{u}_j}{\partial x_j} &= 0, \\ \frac{\partial \overline{\rho}\widetilde{u}_i}{\partial t} + \frac{\partial \overline{\rho}\widetilde{u}_i\widetilde{u}_j}{\partial x_j} + \frac{1}{\mathrm{Ma}^2}\frac{\partial \overline{p}}{\partial x_i} - \frac{1}{\mathrm{Re}}\frac{\partial \widetilde{\sigma}_{ij}}{\partial x_j} &= -\frac{\partial \tau_{ij}^{\mathrm{SGS}}}{\partial x_j} + \epsilon_{u_i}, \\ \frac{\partial \overline{\rho}\widetilde{E}}{\partial t} + \frac{\partial \overline{\rho}\widetilde{u}_j\widetilde{E}}{\partial x_j} + \frac{1}{\mathrm{Ma}^2}\frac{\partial \overline{p}\widetilde{u}_j}{\partial x_j} - \frac{1}{\mathrm{Re}}\frac{\partial \widetilde{u}_i\widetilde{\sigma}_{ij}}{\partial x_j} + \frac{1}{\mathrm{Ma}^2\mathrm{RePr}}\frac{\partial \widetilde{q}_j}{\partial x_j} &= -\frac{\partial f_j^{\mathrm{SGS}}}{\partial x_j} + \epsilon_E, \end{aligned} \tag{3}$$

where $\widetilde{\sigma} = \mu(\widetilde{T})(\nabla\widetilde{\mathbf{u}} + \nabla\widetilde{\mathbf{u}}^\top - \frac{2}{3}(\nabla\cdot\widetilde{\mathbf{u}})\mathbf{I})$ and $\widetilde{\mathbf{q}} = -\mu(\widetilde{T})\nabla\widetilde{T}$. The filtering operation produces several unclosed terms, shown on the right hand side (RHS), due to the nonlinearities of Eq. (1) and several other neglected terms. Historically, the focus has been on the SGS stress tensor and heat-flux vector,

$$\tau_{ij}^{\mathrm{SGS}} \equiv \overline{\rho}\widetilde{u_i u_j} - \overline{\rho}\widetilde{u}_i\widetilde{u}_j \qquad \text{and} \qquad f_j^{\mathrm{SGS}} \equiv \overline{\rho}\widetilde{u_j T} - \overline{\rho}\widetilde{u}_j\widetilde{T}, \tag{4}$$

respectively, which can not be calculated if only the filtered variables (and not the filtered values of their products) are known.

Equation (3) contains additional unclosed terms, denoted ϵ_{u_i} and ϵ_E, that arise due to (a) the noncommutativity of filtering and discrete differentiation operations, and (b) the use of nonlinear constitutive and transport models. These terms are generally neglected in practical LES SGS modeling considerations, even though they can be of a similar magnitude to τ_{SGS} and f_{SGS}. The adjoint-trained DL-SGS model will account for *all* unclosed terms, including these additional terms.

In the DL-SGS model, the unclosed terms are modeled with the aid of a deep neural network (DNN) $\mathbf{h}(\overline{\rho}, \Delta, \widetilde{T}, \nabla\otimes\widetilde{\mathbf{u}}; \theta)$ where Δ is a measure of the local grid length scale, and the parameters θ are to be calibrated to filtered DNS data. In compressible flows, $\mathbf{h}$ is a 4×3 tensor - it has three elements for each momentum equation and 3 elements for the energy equation. Note that, with the 12 inputs $\overline{\rho}$, Δ, $\widetilde{T}$, and $\nabla \otimes \widetilde{\mathbf{u}}$, the closure model $\mathbf{h}$ is Galilean invariant.

B. PyFlowCL Background

The simulations in this paper are conducted in PyFlowCL, an in-house, compressible, curvilinear finite-difference flow solver. PyFlowCL is Python-native and leverages the PyTorch and MPI libraries to enable scalable, distributed simulation and model training with full GPU acceleration. Gradients of the flow variables are calculated using fourth-order central differences, and time is advanced using the explicit fourth-order Runge–Kutta method. To control numerical instabilities from the gradient discretization, a highly discriminating implicit low-pass spatial filter is applied to the conserved variables at the end of each time step.

C. Adjoint Training of Neural Networks in PyFlowCL

Our objective is to select the parameters θ such that the time evolution on the short time interval $[t, t+\tau]$ of the LES solution $\mathbf{Q}(\mathrm{x}, t; \theta) = \{\rho, u, T; \theta\}$ (initialized from trusted high-fidelity data $\mathbf{Q}^*(\mathrm{x}, t)$) matches as closely as possible the evolution of the high-fidelity data. The inclusion of θ in $\mathbf{Q}(\mathrm{x}, t; \theta)$ indicates that it is calculated with the DL-SGS model parameters θ. We will use adjoint equations to optimize over the DL-LES equations to calibrate the parameters θ to the high-fidelity data. As we solve the non-dimensional Navier-Stokes equations, the values of ρ and $\|\mathrm{u}\|_2$ in the freestream flow are 1; the temperature T is divided through by a factor of γ so that it also satisfies this and does not have an disproportionate influence on the loss function. In the equations below, we will consider the fully discretized LES equations. We minimize the following objective function:

$$J(\theta) = \frac{1}{N_p}\left(\left\|\rho(t_f) - \rho^*(t_f)\right\|_1 + \left\|u(t_f) - u^*(t_f)\right\|_1 + \left\|v(t_f) - v^*(t_f)\right\|_1 + \left\|w(t_f) - w^*(t_f)\right\|_1 + \frac{\left\|T(t_f) - T^*(t_f)\right\|_1}{\gamma}\right), \tag{5}$$

where here $t_f = t + \tau$ indicates that the loss is calculated at the final time of the optimization interval $[t, t+\tau]$, $\|\cdot\|_1$ is the ℓ_1 norm, and N_p is the number of LES grid points. This is used instead of a volume integral as it avoids weighting the DL-SGS model away from the more critical areas of the flow where the mesh is finer.

Gradient descent-type algorithms for optimizing the model parameters θ require the gradient $\nabla_\theta J(\theta)$. Adjoint equations are used to efficiently evaluate the gradient $\nabla_\theta J(\theta)$. The adjoint approach only requires the solution of five linear PDEs. The number of adjoint equations does not depend upon the dimension of the neural network parameters θ; this is a key advantage of the adjoint optimization method since deep learning models can have a very large number of parameters.

In order to illustrate the adjoint optimization method, consider the forward discrete update equation for $\mathbf{Q}$ (without a discrete filtering operation):

$$\mathbf{Q}(t + \Delta t) = F(\mathbf{Q}(t); \theta), \tag{6}$$

where F is the discretized PDE update. The adjoint solution $\widehat{\mathbf{Q}}(t) = \partial J / \partial \mathbf{Q}(t)$ is advanced backwards in time (with time step Δt) using the chain rule expression [11]

$$\widehat{\mathbf{Q}}(t) = \left(\frac{\partial F\,(\mathbf{Q}(t); \theta)}{\partial \mathbf{Q}(t)}\right)^\top \widehat{\mathbf{Q}}(t + \Delta t), \tag{7}$$

with the final condition $\widehat{\mathbf{Q}}(t_f) = \partial J / \partial \mathbf{Q}(t_f)$. The RHS in Eq. (7) is evaluated using automatic differentiation. We have extended this to include discrete adjoints for the low-pass filtering operation at the end of each time step and communication across sub-domain boundaries. As far as we are aware, this article is the first to use adjoint optimization to train machine learning closure models for compressible turbulent flows.

Once the adjoint equation has been solved, the gradient of the objective function $\nabla_\theta J(\theta)$ can be evaluated via the following formula:

$$\nabla_\theta J(\theta) = \sum_{i=0}^{N_t} \nabla_\theta F(\mathbf{Q}(t_i);\theta)^\top \widehat{\mathbf{Q}}(t_{i+1}), \tag{8}$$

where $N_t = \tau/\Delta t$ is the number of LES time steps in an optimization interval, and $t_i = t + i\Delta t$.

Once $\nabla_\theta J(\theta^{(k)})$ has been obtained at optimization iteration k, an updated estimate for the parameters can be obtained for the next optimization iteration as

$$\theta^{(k+1)} = \theta^{(k)} - \alpha^{(k)}\nabla_\theta J(\theta^{(k)}), \tag{9}$$

where $\alpha^{(k)}$ is the learning rate.

In practice, optimization of the LES equations with the DL-SGS closure model is carried out using a mini-batch version of the adjoint equation, where multiple instances of the governing equations are solved separately on multiple machines for multiple random sub-samples of filtered DNS data.

The adjoint-training approach used in this work is as follows:

1) Generate a database of DNS data, filtered and down-sampled to the coarse LES grid.
2) Randomly select a short time interval $[t, t+\tau]$.
3) Solve Eq. (3) on $[t, t+\tau]$ (with an initial condition from the filtered and down-sampled DNS data) and calculate the loss function $J(\theta)$ on each machine.
4) Solve the adjoint equation (Eq. (7)) on $[t+\tau, t]$ and calculate $\nabla_\theta J(\theta)$ on each machine.
5) Average the gradients across all machines (i.e. across the mini-batch) and update the parameters θ using the RMSprop algorithm (similar to Eq. (9)).
6) Go back to step 2 and repeat until the loss function has been satisfactorily reduced.

The adjoint solution is fully parallelized and uses the same domain decomposition and overlaps as the forward solution. Due to the very short optimization timescale, Lyapunov divergence due to the chaotic nature of turbulence is not a challenge [11]. Furthermore, optimizing the DL-SGS model over several short timescale simulations is not expected to limit accuracy of the trained model—the small-spatial-scale turbulent flow structures that are most relevant for SGS modeling also have small temporal scales.

III. Training and Testing Datasets

A. DNS overview

We have generated a DNS database of a NACA 0012 airfoil at $\mathrm{Re}_c = 5 \times 10^4$ and Ma = 0.4 at various angles of attack (AoA = 5°, 7.5°, 10.0°, and 15°) for training and testing DL-SGS closures. This airfoil has been chosen as it is a common choice for DNS computations across the literature [14–17]. To maintain consistency with the majority of existing DNS data, all simulations are conducted with a span-wise domain extent that is 20% of the airfoil chord. The DL-SGS model in this paper will only be trained on the AoA = 5° data—this is to test the ability of adjoint-trained models to extrapolate to different angles of attack that may contain unseen flow structures such as large-scale flow separation. We remark that, even when trained on a particular case, the a posteriori coarse mesh LES simulations are quasi-out-of-sample because they are aiming to accurately predict the correct time-average flow and not the short-time evolution of the sub-samples of DNS data that they trained on.

The DNS simulations are conducted on an 187M point O-grid, with 1920 points in the tangential direction, 540 points in the wall-normal direction, and 180 points in the span-wise direction. Snapshots of the instantaneous velocity magnitude at all four flow conditions are shown in Fig. 1. To avoid confinement effects from the far-field boundary condition affecting the results, the computational domain has a radius of 20 chord lengths. We note that this is larger than most DNS simulations of this airfoil, with the exception of [17], who observed slight changes in the pressure at reattachment for domain radii between 15 and 100 chord length. It is reasonable to expect fully separated flow conditions to be more sensitive to this due to the larger domain blockage caused by the separated flow structure. The DNS were run for 15-20 flow-through times before averaging for a further 10 flow-through times to calculate turbulence statistics.

As can be seen in Fig. 1a and b, at AoA = 5° and 7.5°, the flow forms the usual flow structure for low-Reynolds airfoils, consisting of a laminar separation bubble, which breaks down and causes reattachment, followed by a turbulent boundary layer for the rest of the airfoil. In contrast, AoA = 10° and 15° (Fig. 1c and d) exhibit fully separated flow, corresponding to a stalled airfoil.

The fully separated flow at $AoA = 15°$ is consistent with the DNS calculations in [15], but this is not the case at $AoA = 10°$. This is because $AoA \approx 10°$ is roughly where the airfoil stalls: fully separated flow is observed in DNS calculations at $AoA = 9.25°$ in [18], while as previously mentioned, reattaching flow is observed in DNS calculations at $AoA = 10°$ in [15]. It is therefore likely that the flow-field at this condition is extremely sensitive to domain (or tunnel) confinement effects, numerics, and exact details of the mesh resolution. This analysis is backed up by the experimental $C_L - AoA$ curves collated in [19]: different experiments observe this airfoil at this Reynolds number to stall over a wide range of angles of attack, from 6° to 10°.

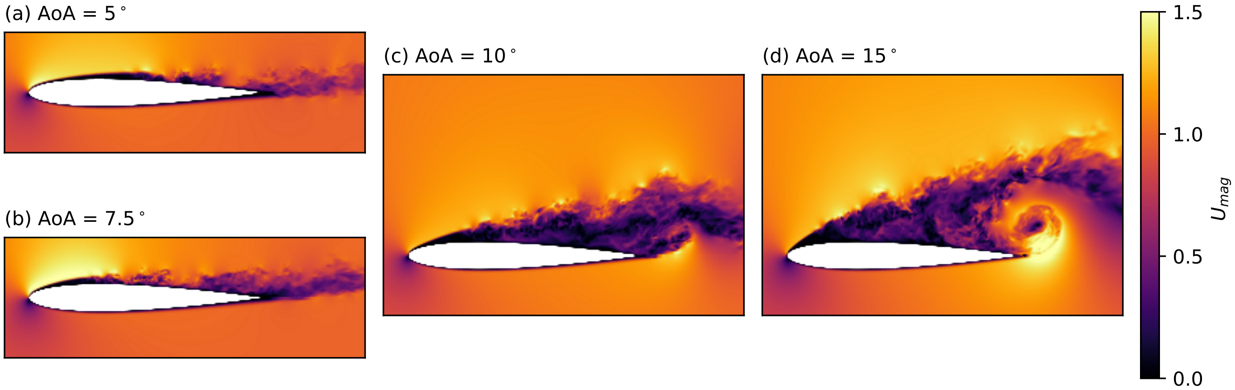

Fig. 1 Snapshots of the instantaneous DNS velocity magnitude, U_{mag}.

The maximum normalized wall spacings from the DNS of the $AoA = 5°$ case are 0.6, 5, and 3 in the wall-normal, stream-wise and span-wise directions. These spacings are well within their recommended limits for DNS [20], indicating adequate resolution of the near-wall flow structures.

B. DNS validation

Our DNS database is validated by comparing time-average pressure and friction coefficients (C_p) to DNS computations of other authors. This is shown in Fig. 2 for $AoA = 5°$, and Fig. 3 for $AoA = 15°$. There is excellent agreement with the DNS data in the literature at $AoA = 5°$, and good agreement at $AoA = 15°$ - it is possible that the latter flow configuration affected by the rounded trailing edge in the present DNS, and/or stronger confinement effects from the smaller domain in the reference DNS data. Further validation is possible at $AoA = 5°$ by comparing our results to the DNS simulations in [17]. Again, the PyFlowCL results match the reference data very closely, both for the time-average wall-parallel velocity (Fig. 4) and for the Reynolds shear stress (Fig. 5).

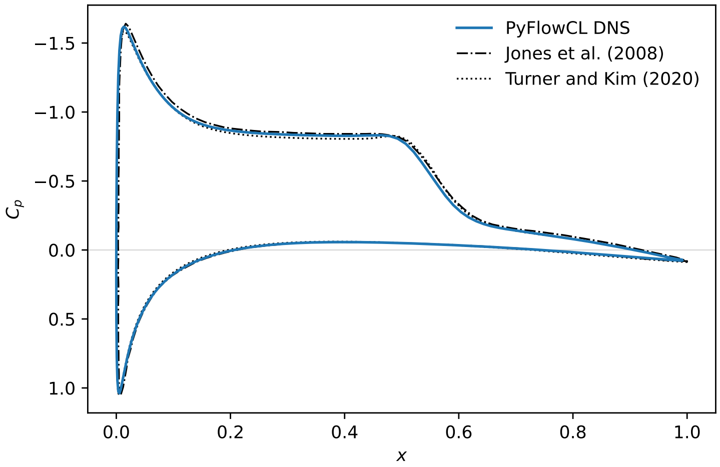

Fig. 2 Time-average DNS pressure coefficients (C_p) at $AoA = 5°$ from PyFlowCL compared to DNS data in the literature [14, 16].

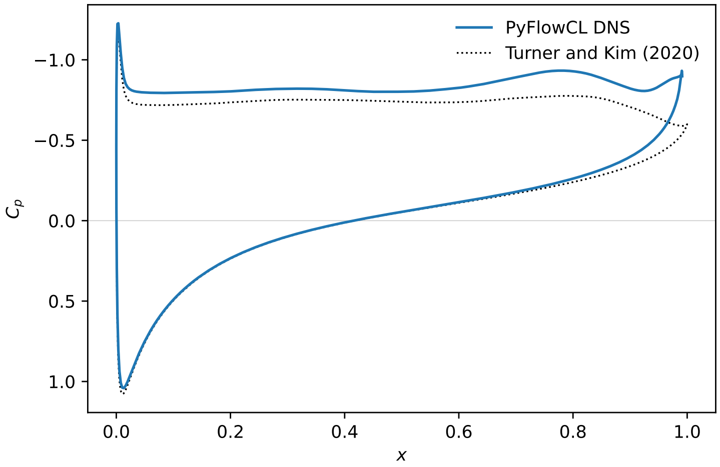

Fig. 3 Time-average DNS pressure coefficients (C_p at $AoA = 15°$ from PyFlowCL compared to DNS data in the literature [15].

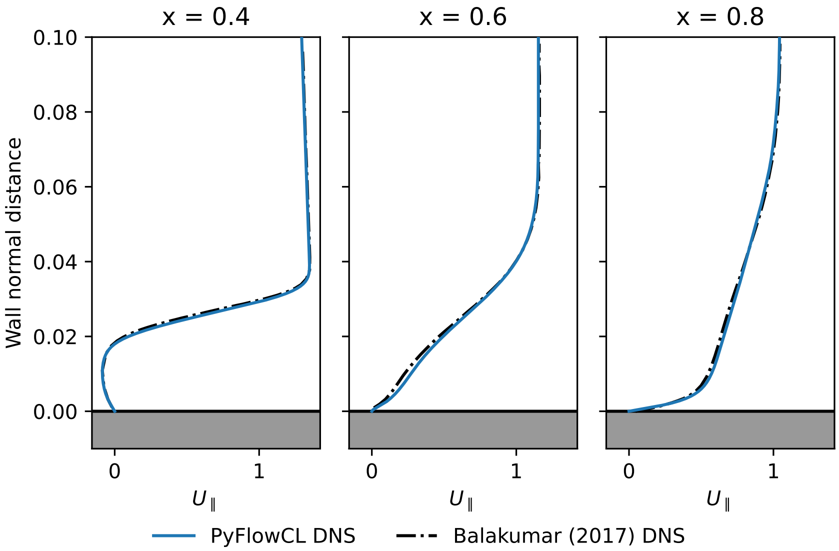

Fig. 4 Time-average DNS wall parallel velocity profiles at $AoA = 5^{\circ}$ compared to the DNS results in [17].

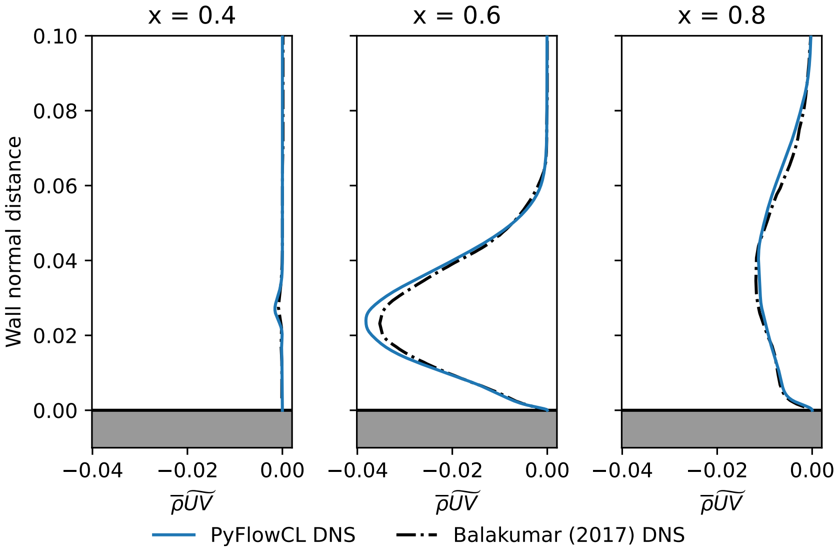

Fig. 5 Time-average DNS wall Reynolds shear stress profiles at $AoA = 5^{\circ}$ compared to the DNS results in [17].

IV. Training and Computational Setup

A. Model formulation

In this paper. we use an anisotropic eddy-viscosity formulation to approximate the SGS stresses. In the anisotropic eddy-viscosity formulation, the SGS stress is approximated as

$$\tau_{ij}^{SGS} + \epsilon_{u_i} \approx \mu(\widetilde{T})\ h_{ij}(\overline{\rho}, \Delta, \widetilde{T}, \nabla \otimes \widetilde{u}) \widetilde{S}_{ij}, \tag{10}$$

and the SGS heat flux is approximated as

$$f_j^{SGS} + \epsilon_E \approx \kappa(\widetilde{T})\ h_{4j}(\overline{\rho}, \Delta, \widetilde{T}, \nabla \otimes \widetilde{u}) \frac{\partial \widetilde{T}}{\partial x_j}, \tag{11}$$

where $\widetilde{S}_{ij}$ is the resolved deviatoric strain rate tensor. An advantage of this approach is that the DL-SGS model is constrained by design to be free-stream preserving. Our experience directly modeling the SGS stresses with a DNN was that this property was complicated to enforce, and was required for stability: spurious SGS stresses were not introduced by changes in grid size, density, or temperature.

B. LES grid and filtering DNS data

The computational grid for the LES simulations is formed by down-sampling the DNS grid by a factor of $8 \times 4 \times 9$ in the tangential, wall normal, and span wise directions. This coarsening results in an LES grid with approximate normalized wall spacings of 40, 4, and 30 in the stream wise, wall-normal, and span wise directions—the wall-normal spacing here is larger than usually recommended for LES, meaning that the DL-SGS model will hopefully learn a degree of wall modeling during the optimization process.

The $8 \times 4 \times 9$ coarsening in a 288-fold reduction in the number of computational points, and an 8-fold increase in the time step supported by the grid. This means that LES simulations with no SGS model have a 2,300 times lower computational cost than the corresponding DNS simulations. In general, we have found that the DL-SGS model has an overhead of about 20% compared to no model, so carrying out DL-SGS simulations is still over three orders of magnitude cheaper than DNS.

A two step process is used for filtering the DNS data. First, the DNS data is averaged in a $9 \times 4 \times 9$ point grid around each LES point. The filtered flow field after this first step still has a 2Δ wavelength component in it. Due to the numerical limitations of collocated finite difference schemes, this is not resolvable on the computational grid—in both DNS and LES we remove it with the implicit filter. To remove these from the filtered DNS data, it is filtered again on the LES grid with a fourth-order explicit low-pass filter. The first step of this filtering process is visualized in Fig. 6.

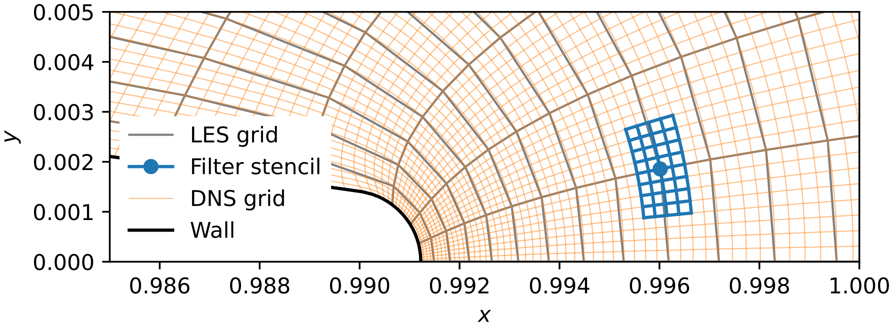

Fig. 6 Visualization of the DNS data filtering and down-sampling to the LES grid near the rounded trailing edge of the NACA 0012.

C. Training hyper-parameters

The DNN $h(\overline{\rho}, \Delta, \widetilde{T}, \nabla \otimes \widetilde{u})$ has four layers and uses ELU (exponential linear unit) activation functions, including on the output layer. A fairly narrow network with 25 hidden units is used - large networks are not desirable in this application, as they increase the cost of both a posteriori simulations and training the model. The use of the exponential linear unit (ELU) functions on the output layer ensures that the net dissipation provided by the eddy viscosity plus the laminar viscosity is always positive: as $x \to -\infty$, $\mathrm{ELU}(x) \to -1$. This allows the model to provide a reasonable amount of back-scatter, while also maintaining solution stability. This avoids having to use clipping methods (such as those proposed in [21]) to limit the back-scatter to a reasonable value.

The training is conducted over 125 time steps on the LES grid; this corresponds to 1% of a chord flow-through time, and a mini-batch size of 24 is used the optimization, with each LES instance being distributed across 2 NVIDIA V100 GPUs. The RMSprop hyper-parameters are kept at their default value in PyTorch, and the learning rate is set to 0.01. Filtered DNS (fDNS) snapshot pairs are randomly selected for each batch (without replacement) from a database of 1,000 snapshot pairs of fDNS data. The out-of-sample performance of the DL-SGS model is periodically checked on a separate test data set consistent of 24 filtered DNS snapshot pairs evenly distributed throughout the temporal extent of the training data. As can be seen in Fig. 7, the optimization converges to approximately a 4% reduction in loss on the test data set, and does not appear to be over fitting to the training data (which would be indicated by the training loss descending wile the test loss levels off or increases).

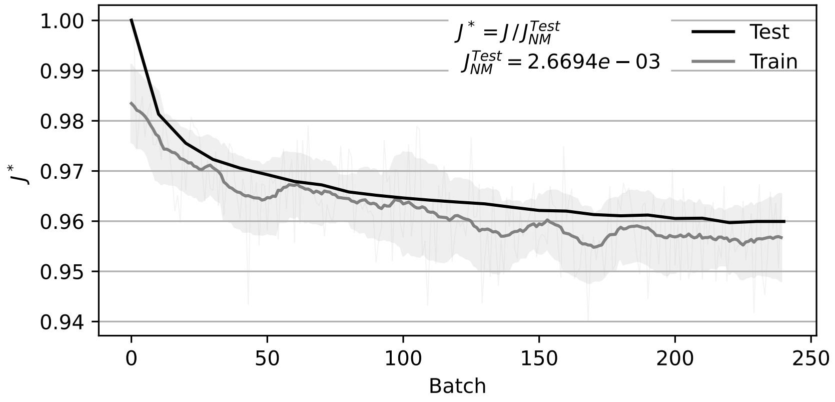

Fig. 7 Convergence of test and training loss (relative to the no model test loss, J_{NM}^{Test}) for the DL-SGS model optimizing over 1% of a chord flow-though time.

V. A Posteriori Results

A. Assessment of DL-SGS model performance

After training, a posteriori LES are run with the DL-SGS model at all four angles of attack considered in this paper. To assess the performance of the DL-SGS model, LES simulations are also run with three classical (i.e. not deep-learning) SGS modelling approaches: no model (or implicit) LES, where the dissipation from the numerical filter is relied upon to provide a similar effect to the SGS stresses [22]; the Smagorinsky model [23] which is well known to have issues in transitional flows; and the wall adapting local eddy viscosity (WALE) model [24] which is formulated in a way that addresses much of the shortcomings of the Smagorinsky model. For the two eddy viscosity based approaches, the SGS heat-flux is modeled assuming a turbulent Prandtl number of 0.85. The values chosen for the Smagorinsky coefficient and the WALE coefficient are $C_s = 0.1$, and $C_w = 0.325$. All LES simulations were run for at least 20 flow through times to expel initial transients and averaged for 10 flow through times to calculate statistics. The DL-SGS model has been verified to result in stable LES simulations at all four conditions for very long run times.

Although the DL-SGS model has been trained over data at $AoA = 5°$, this is considered to be a quasi-out-of-sample case. This is because the model has only been trained over short intervals (1% of a chord flow through time) of filtered DNS (fDNS) data, which is different to the long term evolution of a simulation where the DL-SGS model is able to influence the formation of new flow structures such as shear layer vortices—these typically form on a time scale that is an order of magnitude larger than the optimization period. The $AoA = 7.5°$, 10°, and 15° cases all correspond to fully out of sample flow conditions.

The a posteriori simulations are assessed by comparison to the fDNS simulations. Three figures of merit (FoM) are used for this. The first FoM is in intended to be a time-average equivalent of the loss function used in training (see 5), so that

$$J_{\text{post}} = \frac{1}{N_p}\left(\|\overline{\rho} - \overline{\rho}^*\|_1 + \|\widetilde{u} - \widetilde{u}^*\|_1 + \|\widetilde{v} - \widetilde{v}^*\|_1 + \frac{\|\widetilde{T} - \widetilde{T}^*\|_1}{\gamma} \right), \tag{12}$$

where bars and tildes indicate temporal unweighted- and Favre-averages. J_{post} measures how well a particular SGS model satisfies the goal of the optimization process, but might not necessarily correlate with accurate prediction aof the aerodynamic coefficients. The second and third FoMs are therefore deviations of the LES time-average lift and drag coefficients (C_L and C_D). Tables 1 and 2 shows the values for these error metrics across all four angles of attack for the no model, Smagorinsky, WALE, and DL-SGS model LES simulations. Taken in aggregate across all four angles of attack, the DL-SGS model shows a clear advantage compared to no SGS model and both classical SGS models, with the lowest value in all three error metrics by a comfortable amount (expect for separated cases with the Smagorinsky model). To provide more insight, the flow conditions are broken up in to three groups: reattaching flows ($AoA = 5°$ and 7.5°), flows on the verge of separation ($AoA = 10°$), and fully separated flows ($AoA = 15°$). These three groups are considered in turn in the rest of this section of the paper.

Table 1 Comparison of ℓ_1 error relative to time-average fDNS J_{post} for the no model, Smagorinsky, WALE, and DL-SGS LES.

Angle of attack	5°	7.5°	10°	15°
	Training	Out-of-sample	Out-of-sample	Out-of-sample
No model J_{post}	0.014	0.030	0.178	0.101
Smagorinsky J_{post}	0.083	0.091	0.064	0.047
WALE J_{post}	0.012	0.025	0.169	0.069
DL-SGS J_{post}	0.009	0.018	0.136	0.047

Table 2 Comparison of errors in C_L and C_D for the no model, Smagorinsky, WALE, and DL-SGS LES.

Angle of attack	5°	7.5°	10°	15°
	Training	Out-of-sample	Out-of-sample	Out-of-sample
fDNS C_L	0.605	0.725	0.449	0.788
No model C_L error [%]	6.2	12.5	111.0	3.6
Smagorinsky C_L error [%]	35.3	16.2	54.2	12.9
WALE C_L error [%]	5.8	10.4	105.9	10.9
DL-SGS C_L error [%]	3.0	4.7	84.2	13.2
fDNS C_D	0.0356	0.050	0.125	0.273
No model C_D error [%]	6.8	22.0	44.9	29.6
Smagorinsky C_D error [%]	51.9	93.6	12.6	14.7
WALE C_D error [%]	6.0	17.6	40.1	26.0
DL-SGS C_D error [%]	2.6	7.2	20.9	17.6

1. Reattaching flows

For the two reattaching flow cases ($AoA = 5°$ and $7.5°$), where the model was trained over representative physics, Tab. 1 shows the error in the DL-SGS lift and drag coefficients is about half the next best model (WALE). The DL-SGS time-average flow-field is also much closer to fDNS at these conditions. We consider this to be a significant reduction in modeling error and improvement in predictive ability - even for the out-of-sample flow condition.

The Smagorinsky model does particularly poorly at these conditions and does not offer an improvement over no SGS model at all. The flow feature causing this can be seen in Fig. 8: the DL-SGS model (and no model and WALE) correctly predicts reattaching flow, but the Smagorinsky model misses this—likely due to its well known issues in transitional flows preventing the breakdown of the laminar separation bubble.

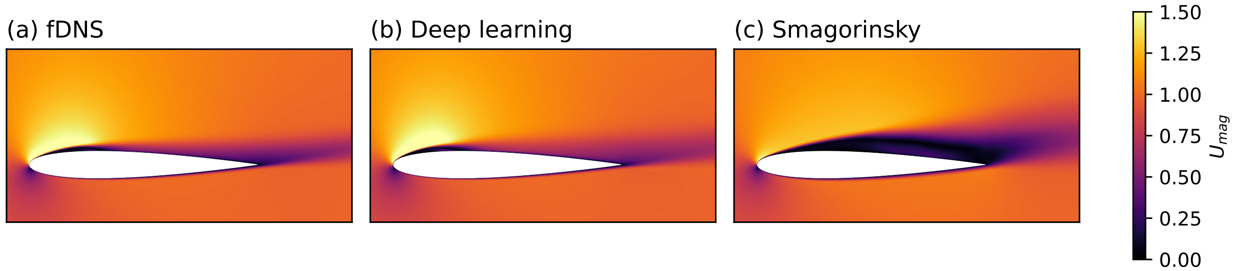

Fig. 8 Time-average velocity magnitude fields for fDNS and the DL-SGS and Smagorinsky models at $AoA = 7.5°$.

It is also interesting to compare the location of separation and reattachment predicted by the different SGS models at these conditions. This is shown in Fig. 9 - for both the quasi-out-of-sample and the out-of-sample case, DL-SGS predicts the first onset of separation and the location of reattachment to be close the where they were observed in the fDNS (although it predicts a small region of temporary reattachment for $x < 0.2$ at $AoA = 7.5°$). In contrast, both the classical SGS modeling approaches (no model and WALE) predict separation and reattachment late at $AoA = 5°$, and reattachment early at $AoA = 7.5°$. The DL-SGS also predicts the correct friction coefficient in the turbulent boundary layer downstream of reattachment (approx $x > 0.6$ and $x > 0.3$ for $AoA = 5°$ and $7.5°$), which neither the no model nor the WALE model are able to do.

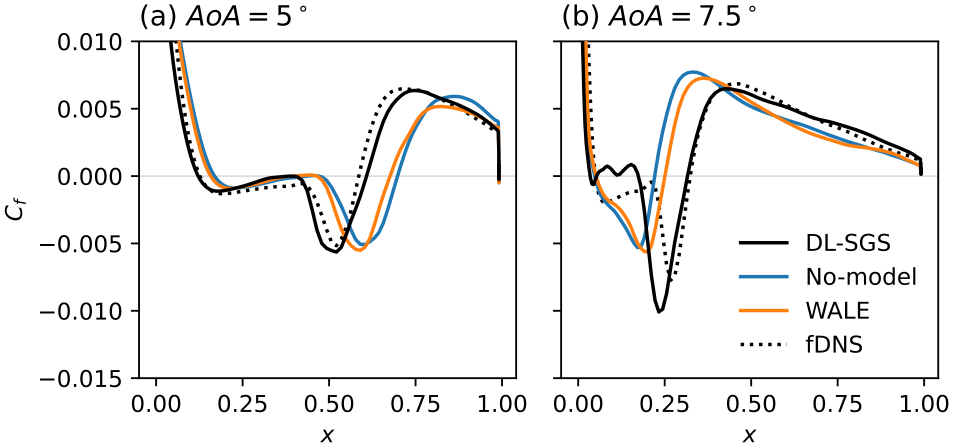

Fig. 9 LES suction surface friction coefficient distributions for the (a) quasi-out-of-sample and (b) out-of-sample reattaching flow conditions.

The local ℓ_1 error in the time-average pressure $\overline{p}$ and Favre-averaged velocity magnitude, $\widetilde{U}_{mag}$ at $AoA = 7.5°$ is shown in Fig. 10. The errors in the DL-SGS simulation (Fig. 10c) are significantly lower than both no model LES and LES with the WALE model (Figs. 10a and b) near the leading edge and on the suction surface of the airfoil. The reason for this can be found in the friction coefficient distributions in Fig. 9 and the boundary layer profiles in Fig. 11.

At $x = 0.2$, the WALE model and the no model simulations are close to predicting reattachment - this results in less blockage from the separation bubble and consequently fuller boundary layer velocity profiles going downstream. The DL-SGS model captures the leading edge separation bubble much more accurately, resulting in boundary layer profiles that are closer to fDNS downstream. The reason for the consistent over-prediction of the near-wall velocity by the DL-SGS is unclear (we also observe this at $AoA = 5°$)—whether it can be reduced by changing the training period or the model inputs requires further investigation. An alternative possibility is that specialist wall-models must be trained to reduce it. In any case, the DL-SGS model predicts more accurate boundary layer profiles than either classical SGS modeling approach.

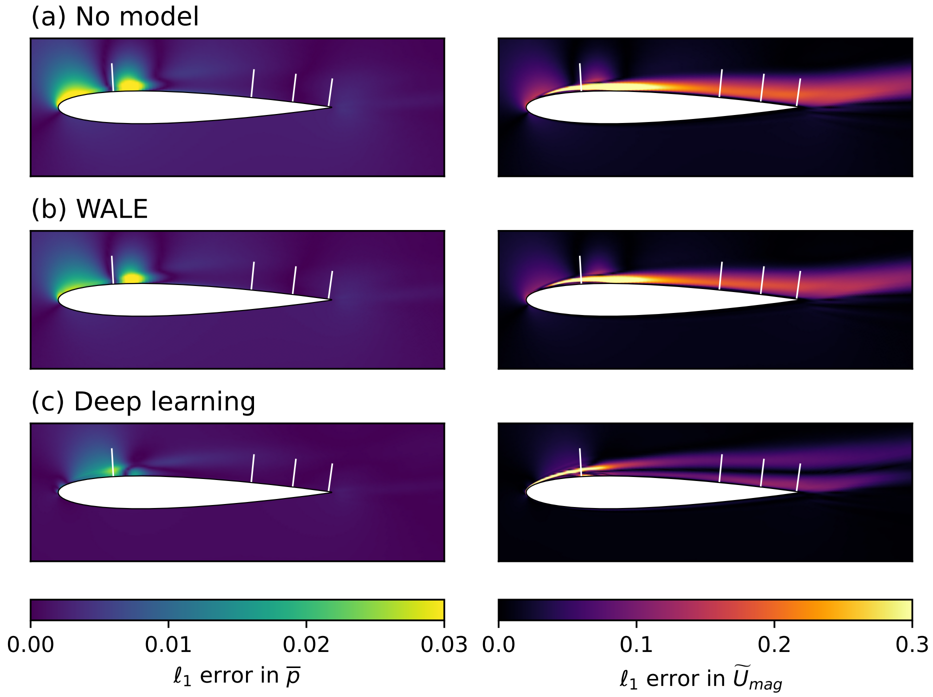

Fig. 10 **ℓ_1 errors (compared to fDNS) in time-average pressure $\overline{p}$ and Favre-averaged velocity magnitude, $\widetilde{U}_{mag}$ at $AoA = 7.5°$. White lines indicate locations of boundary layer profiles.**

2. Flow on the verge of separation

At $AoA = 10°$, the analysis is difficult due to the extremely sensitive nature of the flow field at the verge of stall—as discussed in Sec. III there is a large spread in DNS simulations in the literature around this condition, with some predicting reattachment and some predicting fully-separated flow, and the location of stall in experimental $C_L - AoA$ curves seems to vary widely [19]. As can be seen in Tab. 1 the no model error in time-average flow field J_{post} is 10 times larger than at e.g. $AoA = 5°$, although the DL-SGS model reduces this by 24%, despite never having seen fully separated flow in training.

The Smagorinsky model does well at this condition compared to the other models, although this is a case of a stopped clock being right twice a day: it is not surprising that it predicts fully separated flow given that it does that for all other flow conditions too. This is shown in Fig. 12. Even if it predicts the correct flow structure, the Smagorinsky model still has a large error in C_L (Tab. 2), so it is not accurately predicting the large-scale fDNS flow.

3. Fully separated flow

For the fully separated flow condition ($AoA = 15°$), the Smagorinsky model is again the best performing model, although this time the DL-SGS model is very competitive—J^*_{post} is broadly the same between the two cases, and the errors in C_L and C_D from the Smagorinsky model are only lower by a 0.3 and 2.9 points (Tab. 1). Some brief numerical experiments that we conducted indicated that had we optimized the value of C_s on the $AoA = 5°$ in the same manner as

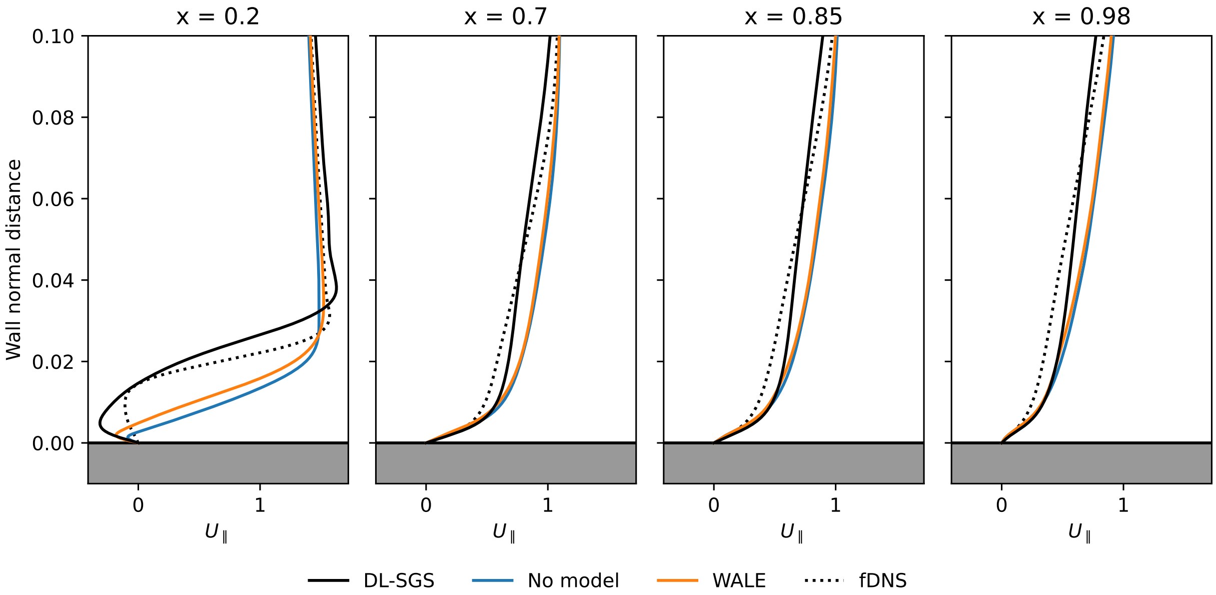

Fig. 11 Wall-normal boundary layer profiles at $AoA = 7.5°$. $U_\parallel$ is the velocity in the wall parallel direction. Profile locations indicated in Fig. 10.

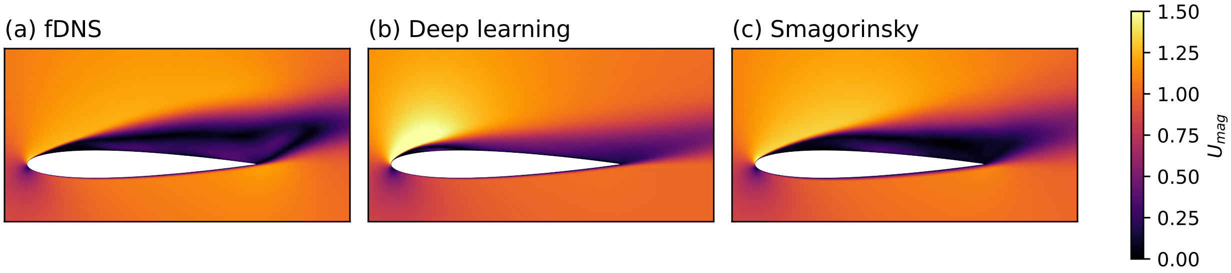

Fig. 12 Time-average velocity magnitude fields for fDNS and the DL-SGS and Smagorinsky models at $AoA = 10.0°$.

the DL-SGS model, C_s would have been $O(0.01)$. This would result in a performance that is on par with the no model LES-including larger errors at high angles of attack. The no model and WALE simulations have a low error in C_L - given their larger error in C_D and larger J_{post}, this seems to be because of a cancellation of errors, and is not indicative of a dependable predictive ability at this angle of attack.

At this far-out-of-sample condition we also see a significant improvement in the behavior of the flow around the laminar separation bubble at the leading edge of the airfoil from the DL-SGS model (Fig. 13). While this might be the main source of improvement from the DL-SGS model at this condition (compared to no model and WALE), if the model had not learned some transferable physics from the re-attaching flow cases, it is likely that the unseen SGS physics downstream of about 50% chord would have been incorrect, corrupting the flow in this region and resulting in worse overall performance than the no-model simulations. This is not reflected in Tab. 1 or Fig. 13. This is extremely promising given that the fully separated flow structure contains large amounts of flow physics that are unseen to the DL-SGS model (large-scale leading edge vortex roll up, fully separated flow, and a Von-Karman vortex street—see Fig. 1), and suggests that the DL-SGS model has learned at least some generalizable features of the SGS physics.

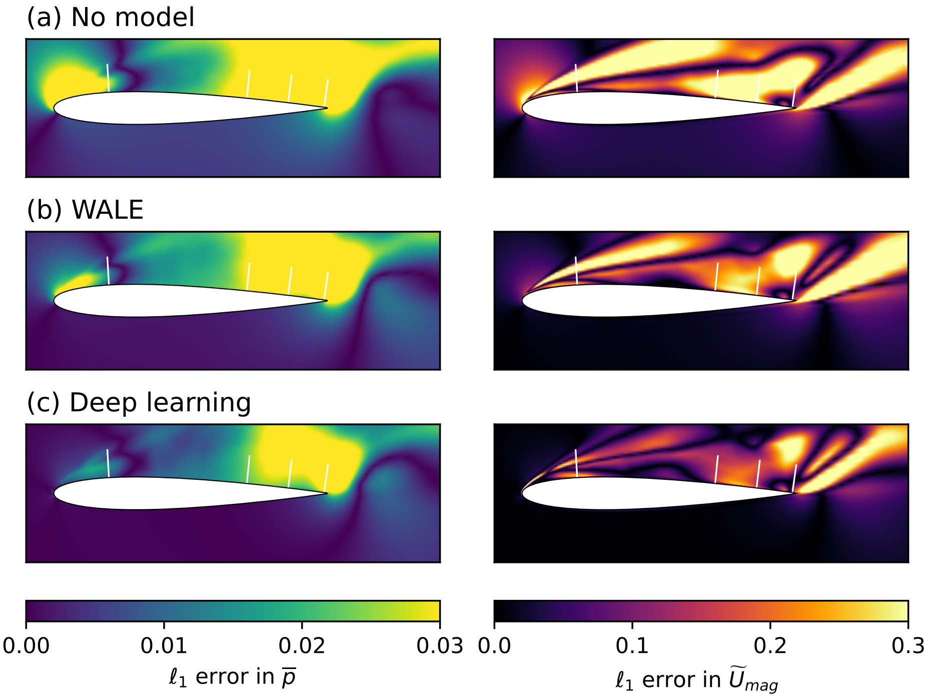

Fig. 13 **ℓ_1 errors (compared to fDNS) in time-average pressure $\overline{p}$ and Favre-averaged velocity magnitude, $\widetilde{U}_{mag}$ at $AoA = 15.0°$.**

VI. Conclusion

The focus of this paper was the development of DL-SGS models for compressible for over a NACA 0012 airfoil at Re = 50, 000 and Ma = 0.4. The results presented in this paper represent a significant improvement in the ability of DL-SGS models to generalize to previously unseen flow conditions and flow physics. We attribute this to the adjoint training method used, where the DL-SGS model is optimized to match the short term evolution of filtered DNS data—this has significant advantages over the a priori method that is usually used for training DL-SGS models, which is not able to take in to account the interaction of the DL-SGS model with the governing equations or the LES numerics.

The DL-SGS model is only trained on filtered DNS data from one angle of attack ($AoA = 5°$) at the lower end of angles of attack investigated. A relatively coarse LES grid is used, corresponding to 288 times fewer grid points than the DNS simulations. The DL-SGS model is tested in a posteriori simulations at $AoA = 5°$, 7.5°, 10°, and 15°. The $AoA = 5°$ a posteriori simulation is considered quasi-out-of-sample, this is because the model has only been trained over short periods (1% of a chord flow-through time), but the a posteriori simulations run for more than 30 flow-through times, including 10 for gathering statistics. Of the fully out-of-sample simulations, the flow field at $AoA = 7.5°$ displays the same broad physics as at $AoA = 5°$, consisting of a laminar separation bubble followed by turbulent reattachment. The flow fields at $AoA = 10°$ and 15° are very far out-of-sample—they are characterized by fully separated flow, trailing edge vortex roll up, and the formation of large coherent structures downstream of the stalled airfoil. The DL-SGS model was not trained over any data that included this physics.

Despite only being trained at a single angle of attack, the DL-SGS model out performs simulations with no SGS model and the WALE model in all four cases, leading to reductions in error of the lift and drag coefficients and the time-average flow field. The Smagorinsky model incorrectly predicts fully separated flow at all angles of attack. This does result in it having a more accurate flow field than DL-SGS compared to the filtered DNS simulation at 10° (this flow condition is likely extremely sensitive as it is on the verge of stall), although DL-SGS is competitive with it at 15° and generalizes an order of magnitude better to lower angles of attack.

For the flow configurations with reattachment (the quasi-out-of-sample $AoA = 5°$ and the fully out-of-sample $AoA = 7.5°$) where the DL-SGS model has been trained over broadly representative flow physics, it results in a significantly more accurate flow field, with errors in the lift and drag coefficients that are 50% smaller than the next best model (WALE). The DL-SGS model correctly predicts the onset of separation and reattachment at both of these flow conditions, which no model, Smagorinsky model, and WALE model simulations fail to do. In addition, in the out-of-sample flow $AoA = 7.5°$ flow condition it enables the LES to properly capture the leading edge laminar separation bubble, a feature that the classical SGS models also fail to accurately predict. Even in the far-our-of-sample case ($AoA = 15°$) where the DL-SGS model has not been trained over any representative flow physics, it is competitive with the best classical SGS model (Smagorinsky), giving a similar ℓ_1 error in the time-average flow field, and only a marginally worse error in the time-average lift coefficient. The strong performance of the DL-SGS model at this flow condition implies that, during training, it has learned at least some generalizable SGS physics that enable it to give reasonable predictions for this far out-of-sample flow condition.

Acknowledgments

This work is supported by the U.K. Engineering and Physical Sciences Research Council grant EP/X031640/1. This material is based in part upon work supported by the U.S. National Science Foundation under Award CBET-22-15472. This research used resources of the Oak Ridge Leadership Computing Facility, which is a DOE Office of Science User Facility supported under Contract DE-AC05-00OR22725 (INCITE award "Extreme-Scale Data Assimilation for Predictive Flow Simulations").

For the purpose of open access, the authors have applied a Creative Commons attribution (CC BY) licence to any accepted manuscript version arising.

References

[1] Spalart, P. R., and Venkatakrishnan, V., "On the role and challenges of CFD in the aerospace industry," *The Aeronautical Journal*, Vol. 120, No. 1223, 2016, p. 209–232.

[2] Bose, S. T., and Park, G. I., "Wall-Modeled Large-Eddy Simulation for Complex Turbulent Flows," *Annual Review of Fluid Mechanics*, Vol. 50, No. 1, 2018, pp. 535–561.

[3] Duraisamy, K., "Perspectives on machine learning-augmented Reynolds-averaged and large eddy simulation models of turbulence," *Physical Review Fluids*, Vol. 6, 2021, p. 050504.

[4] Beck, A., Flad, D., and Munz, C.-D., "Deep neural networks for data-driven LES closure models," *Journal of Computational Physics*, Vol. 398, 2019, p. 108910.

[5] Meng, Q., Jiang, Z., and Wang, J., "Artificial neural network-based subgrid-scale models for LES of compressible turbulent channel flow," *Theoretical and Applied Mechanics Letters*, Vol. 13, No. 1, 2023, p. 100399.

[6] Prakash, A., Jansen, K. E., and Evans, J. A., "Invariant data-driven subgrid stress modeling in the strain-rate eigenframe for large eddy simulation," *Computer Methods in Applied Mechanics and Engineering*, Vol. 399, 2022, p. 115457.

[7] Benjamin, M., Domino, S. P., and Iaccarino, G., "Neural networks for large eddy simulations of wall-bounded turbulence: numerical experiments and challenges," *The European Physical Journal E*, Vol. 46, No. 7, 2023, p. 55.

[8] Bae, H. J., and Lozano-Duran, A., "Numerical and modeling error assessment of large-eddy simulation using direct-numerical-simulation-aided large-eddy simulation," 2022. https://doi.org//10.48550/arXiv.2208.03498.

[9] Sirignano, J., and MacArt, J. F., "Deep learning closure models for large-eddy simulation of flows around bluff bodies," *Journal of Fluid Mechanics*, Vol. 966, 2023, p. A26.

[10] MacArt, J. F., Sirignano, J., and Freund, J. B., "Embedded training of neural-network subgrid-scale turbulence models," *Physical Review Fluids*, Vol. 6, 2021, p. 050502.

[11] Sirignano, J., MacArt, J. F., and Freund, J. B., "DPM: A deep learning PDE augmentation method with application to large-eddy simulation," *Journal of Computational Physics*, Vol. 423, 2020, p. 109811.

[12] Garmann, D. J., Visbal, M. R., and Orkwis, P. D., "Comparative study of implicit and subgrid-scale model large-eddy simulation techniques for low-Reynolds number airfoil applications," *International Journal for Numerical Methods in Fluids*, Vol. 71, No. 12, 2013, pp. 1546–1565.

[13] Slotnick, J. P., Khodadoust, A., Alonso, J., Darmofal, D., Gropp, W., Lurie, E., and Mavriplis, D. J., "CFD Vision 2030 Study: A Path to Revolutionary Computational Aerosciences," *NASA Contractor Report*, 2014.

[14] Jones, L. E., Sandberg, R. D., and Sandham, N. D., "Direct numerical simulations of forced and unforced separation bubbles on an airfoil at incidence," *Journal of Fluid Mechanics*, Vol. 602, 2008, p. 175–207.

[15] Turner, J. M., and Kim, J. W., "Effect of spanwise domain size on direct numerical simulations of airfoil noise during flow separation and stall," *Physics of Fluids*, Vol. 32, No. 6, 2020, p. 065103.

[16] Turner, J. M., and Kim, J. W., "Aerofoil dipole noise due to flow separation and stall at a low Reynolds number," *International Journal of Heat and Fluid Flow*, Vol. 86, 2020, p. 108715.

[17] Balakumar, P., "Direct Numerical Simulation of Flows over an NACA-0012 Airfoil at Low and Moderate Reynolds Numbers," *47th AIAA Fluid Dynamics Conference*, American Institute of Aeronautics and Astronautics, 2017.

[18] Rodríguez, I., Lehmkuhl, O., Borrell, R., and Oliva, A., "Direct numerical simulation of a NACA0012 in full stall," *International Journal of Heat and Fluid Flow*, Vol. 43, 2013, pp. 194–203.

[19] Tank, J., Smith, L., and Spedding, G. R., "On the possibility (or lack thereof) of agreement between experiment and computation of flows over wings at moderate Reynolds number," *Interface Focus*, Vol. 7, No. 1, 2017, p. 20160076.

[20] Georgiadis, N. J., Rizzetta, D. P., and Fureby, C., "Large-Eddy Simulation: Current Capabilities, Recommended Practices, and Future Research," *AIAA Journal*, Vol. 48, No. 8, 2010, pp. 1772–1784.

[21] Prakash, A., Jansen, K. E., and Evans, J. A., "Optimal Clipping of Structural Subgrid Stress Closures for Large-Eddy Simulation," *AIAA Journal*, Vol. 60, No. 12, 2022, pp. 6897–6909.

[22] Grinstein, F., Margolin, L., and Rider, C., *Implicit Large Eddy Simulation: Computing Turbulent Fluid Dynamics*, Cambridge University Press, 2007.

[23] Smagorinsky, J., "General circulations experiments with the primitive equations: I. The basic experiment," *Monthly Weather Review*, Vol. 91, No. 3, 1963, pp. 99–164.

[24] Nicoud, F., and Ducros, F., "Subgrid-Scale Stress Modelling Based on the Square of the Velocity Gradient Tensor," *Flow, Turbulence and Combustion*, Vol. 62, No. 3, 1999, pp. 183–200.