DEEP LEARNING MODELING FOR SUBGRID-SCALE FLUXES IN THE LES OF SCALAR TURBULENCE AND TRANSFER LEARNING TO OTHER TRANSPORT REGIMES

Ali Akhavan-Safaei[1,2] *& Mohsen Zayernouri*[1,3,*]

[1]*Department of Mechanical Engineering, Michigan State University, East Lansing, Michigan 48824, USA*

[2]*Department of Computational Mathematics, Science, and Engineering, Michigan State University, East Lansing, Michigan 48824, USA*

[3]*Department of Statistics and Probability, Michigan State University, East Lansing, Michigan 48824, USA*

*Address all correspondence to: Mohsen Zayernouri, Department of Mechanical Engineering, Michigan State University, East Lansing, MI 48824, USA, E-mail: zayern@msu.edu

Original Manuscript Submitted: 5/9/2023; Final Draft Received: 9/6/2023

We present a deep learning surrogate model in the form of fully-connected feed-forward neural networks to predict the SGS scalar flux in the context of large-eddy simulation of turbulent transport. The deep neural network model is trained and validated using a filtered DNS dataset at $Pe_\lambda = 240$, $Sc = 1$ that includes the filtered scalar and velocity gradients as input features. We provide a sampling strategy in time and space to construct a sufficiently inclusive dataset for successful model optimization. Moreover, using transfer learning notion, we adjust this trained model to perform accurately for turbulent scalar transport regimes with higher Pe_λ and Sc numbers with a fairly low amount of data and computations. In the inference mode on the full-size 3D LES grids over time, we carry out a priori tests for the developed deep learning models, and compare them to the results obtained from conventional models of SGS flux that are defined based on gradients of filtered fields. We show that the deep learning model outperforms traditional SGS models that either suffer from their fundamental restrictions in following the physics or under/overprediction of SGS quantities. For the transfer learned models, these comparisons are made between the base (pretrained) model and the retrained one on the new transport regime, and we discuss the importance of transfer learning in reduction of notably large prediction errors of the pretrained model.

KEY WORDS: *deep learning surrogate, SGS-scalar fluxes, large-eddy simulation (LES), filtered DNS data, gradient-based SGS models, transfer learning, two-point statistics, forward/back-scattering of scalar variance dissipation*

1. INTRODUCTION

Abundance of data, and public access to modern machine learning (ML) libraries have shaped a *data-driven* era virtually for every modeling discipline such as those in the physics-based and engineering fields (Bergen et al., 2019). Over the past decade, the advent of scalable and computationally efficient implementations of deep learning libraries has sparked a growing interest in developing and using a variety of deep neural network (DNN) architectures that constitute predictive models from latent features of datasets even with nonlinear complexities. For instance, learning parametric spaces (Brenowitz and Bretherton, 2018, 2019), reduced-order modeling (Kutz et al., 2016), inverse problems (Arzani et al., 2021; Kumar et al., 2020; Raccuglia et al., 2016; Raissi et al., 2019), solving forward differential equations (Arzani et al., 2021; Han et al., 2018; Raissi et al., 2019; Sirignano and Spiliopoulos, 2018), discovering governing equations (Brunton et al., 2016; Rudy et al., 2017), and data generation (Fukami et al., 2019; Li and McComb, 2022; Wu et al., 2020) are a few examples among important applications in the physics-based modeling via deep learning.

Fluid mechanics is among the challenging branches of physics-based modeling that recently have notably benefited from diverse deep learning architectures (Brunton et al., 2020; Kutz, 2017). In turbulence modeling, promising advances have been made mainly due to the prevalence of high-fidelity simulations and experimental data (Beck and Kurz, 2021; Drygala et al., 2022; Duraisamy, 2021; Duraisamy et al., 2019; Peng et al., 2022; Wu et al., 2018). Many of these contributions were made in the area of Reynolds-averaged Navier-Stokes (RANS) modeling (see e.g., Charalampopoulos and Sapsis, 2022; Li et al., 2022; Ling et al., 2016; Wang et al., 2017; Zangeneh, 2021).

Particularly, in large-eddy simulation (LES) of turbulent transport, researchers have traditionally proposed models for the closure terms that appear in the filtered governing equations, and tried to model the effects of closure (unresolved) terms using resolved-scale flow quantities. These modeling strategies are predominantly categorized into "functional" or "structural" approaches (Sagaut, 2006). In a functional model, the closure term takes the form of a mathematical operator acting on the resolved-scale field. Therefore, functional models are only reproducing the net transfer of turbulence intensity from the resolved to unresolved scales. However, a structural modeling approach would approximate the LES closure in terms of the resolved-scale field, where the subgrid-scale (SGS) structures and statistical characteristics are recovered from the resolved scale information. However, novel DNN surrogate modeling approaches have recently opened a thriving "data-driven" direction to closure modeling for LES. The study by Sarghini et al. is among the earliest reported works focusing on utilizing a neural network architecture for SGS modeling of turbulence (Sarghini et al., 2003). More recently, a study by Beck et al. (2019) managed to test the SGS modeling of the closure term appearing in the filtered momentum equation with deep neural networks in the context of decaying homogeneous isotropic turbulent (HIT) flows. For more recent advancement of the LES modeling in the context of HIT flows, the reader is referred to Wang et al. (2021), Williams et al. (2022), Xie et al. (2020), and Yuan et al. (2020). In addition for the recent developments in the fractional-order modeling and fractional LES modeling in the context of HIT flows, the reader is referred to Lischke et al. (2019), Samiee et al. (2020, 2022), Seyedi and Zayernouri (2022), and Suzuki et al. (2023, 2021). Moreover, studies such as Gamahara and Hattori (2017) on modeling the SGS stress tensor and LES in turbulent wall flows have employed more advanced implementations of a deep neural network [for more recent works see Park and Choi (2021), Liu et al. (2022), Bae and Koumoutsakos (2022), and Kim et al. (2022)], while Yang et al. (2019) presented a

physics-based neural network for wall-modeled LES. Readers interested in constraining and embedding the physical laws to the deep learning LES models are referred to Frezat et al. (2021), MacArt et al. (2021), and Sirignano et al. (2020).

In the SGS modeling for SGS scalar flux, Vollant et al. (2017) managed to represent the closure flux with a deep learning model, and tested their model for homogeneous isotropic turbulent (HIT) flows. Later, Portwood et al. (2021) proposed a deep neural network model for the SGS scalar flux and trained their model on an extensive dataset obtained from filtering the direct numerical simulation of a passive scalar with uniform mean gradient in a HIT flow. They evaluated their model in *a priori* (off-line) and *a posteriori* (on-line) tests and showed their model significantly outperforms the functional and structural gradient-type SGS models. In another study, Frezat et al. (2021) developed a deep learning SGS scalar flux model using a convolutional neural network assembly, while they enforced multiple transformation invariance properties (such as translation, rotation, and Galilean invariance). They showed that enforcing such physical properties is crucial for improving the performance of the SGS model. Looking back to Vollant et al.'s work, they mentioned that they were not successful in testing their trained SGS model on a different scalar transport regime (Vollant et al., 2017). In 2D turbulence settings, Subel et al. (2021) and Guan et al. (2022) conducted successful generalization of pretrained deep learning SGS models to predict on higher Reynolds number regimes using *transfer learning* technique (Yosinski et al., 2014). Inspired by these pioneering studies, here in the context of 3D homogeneous turbulence with a uniform scalar gradient, we seek to develop a deep learning SGS scalar flux model that could feasibly be generalized to other turbulent transport regimes. As a result, the key contributions of our work are summarized as follows:

- We develop a DNN surrogate model for the SGS scalar flux, and utilize the gradients of the resolved-scale scalar and velocity fields as input features. We propose a spatio-temporal sampling approach for constructing a diverse training/validation dataset from filtered DNS (FDNS) results, and seamlessly optimize the DNN model for a specified turbulent transport regime.

- Using the transfer learning concept, we generalize our pretrained DNN model to accurately predict on different scalar transport regimes with higher Schmidt and higher Péclet numbers. We demonstrate that compared to the pretrained model, the amount of required data and the computational cost of model optimization are significantly reduced in the retraining steps.

- We conduct multiple tests to evaluate the performance of optimized models in the inference mode on full-size 3D FDNS grids at multiple time instances. In *a priori* analysis, we show that the base DNN model decreases the time-averaged spatial error in the predicted SGS flux and the SGS dissipation of scalar variance compared to two traditional models. Similar comparisons are presented between the performance of the pretrained and transfer learned models for higher Schmidt and higher Péclet number scalar transport regimes.

- Finally, we test the performance of our base DNN model when utilized in the LES setting, and we show that unlike two other conventional SGS models, the DNN is successful in maintaining the accuracy for predicted time-record resolved-scale scalar variance, and time-averaged two-point structure functions of the LES-resolved scalar concentration field.

The remaining part of this paper is organized as follows: in Section 2, we present the methodologies for numerical simulations and data-driven modeling for base and transfer learned DNN models. In Section 4, we go over *a priori* and *a posteriori* tests of the trained models and compare them to the conventional gradient-based SGS models for scalar flux. Lastly, we conclude our study with a presentation of a summary of contributions and findings.

2. MODELING PARADIGM

2.1 Governing Equations

Considering the incompressible flow regime and the transport of a conserved passive scalar in that flow medium, Navier-Stokes (NS) and advection-diffusion (AD) equations are the set of governing equations constituting the system dynamics (Pope, 2001). In the LES of turbulent transport, a generic spatial filtering operator, $\tilde{\cdot}$, is applied to the NS and AD equations returning the LES governing equations [see, e.g., Sagaut (2006)] as

$$\frac{\partial \tilde{\boldsymbol{u}}}{\partial t} + \tilde{\boldsymbol{u}} \cdot \boldsymbol{\nabla} \tilde{\boldsymbol{u}} = -\frac{1}{\rho} \boldsymbol{\nabla} \tilde{p} + \nu \, \Delta \tilde{\boldsymbol{u}} - \boldsymbol{\nabla} \cdot \boldsymbol{\tau}^R; \quad \boldsymbol{\nabla} \cdot \tilde{\boldsymbol{u}} = 0, \tag{1}$$

$$\frac{\partial \tilde{\Phi}}{\partial t} + \tilde{\boldsymbol{u}} \cdot \boldsymbol{\nabla} \tilde{\Phi} = \mathcal{D} \, \Delta \tilde{\Phi} - \boldsymbol{\nabla} \cdot \boldsymbol{q}^R. \tag{2}$$

In this set of equations, $\boldsymbol{u} = (u_1, u_2, u_3)$, p, and Φ are the velocity, pressure, and scalar concentration fields, respectively. In Eq. (1), ρ denotes the fluid density, while ν represents the viscosity of fluid, and in Eq. (2), $\mathcal{D}$ is the diffusivity of the passive scalar field. Moreover, filtering yields sources of closure in the LES governing equations as the divergence of *residual stress*, $\boldsymbol{\tau}^R = \widetilde{\boldsymbol{u} \otimes \boldsymbol{u}} - \tilde{\boldsymbol{u}} \otimes \tilde{\boldsymbol{u}}$, and *residual scalar flux*, $\boldsymbol{q}^R = \widetilde{\boldsymbol{u}\,\Phi} - \tilde{\boldsymbol{u}}\,\tilde{\Phi}$. Modeling these residual or subgrid terms using the filtered or resolved flow fields is an essential gateway returning a closed set of equations that are suitable for a predictive and numerically stable LES (Meneveau and Katz, 2000; Sagaut, 2006).

The Reynolds decomposition for a general field such as scalar concentration is $\Phi = \langle \Phi \rangle + \phi$, where $\langle \Phi \rangle$ is the ensemble-averaged part of Φ, and ϕ denotes its fluctuating part (Pope, 2001). In our problem setting, we consider a homogeneous isotropic medium for velocity field; therefore, $\langle \boldsymbol{u} \rangle = 0$. For the passive scalar field, we assume the fluctuations are statistically homogeneous while we consider an imposed ensemble-averaged gradient as $\boldsymbol{\nabla} \langle \Phi \rangle = (0, 1, 0)$ (Akhavan-Safaei and Zayernouri, 2023a,b; Overholt and Pope, 1996). As a result, the filtered AD equation (2) is rewritten as

$$\frac{\partial \tilde{\phi}}{\partial t} + \tilde{\boldsymbol{u}} \cdot \boldsymbol{\nabla} \tilde{\phi} = -\tilde{u}_2 + \mathcal{D} \, \Delta \tilde{\phi} - \boldsymbol{\nabla} \cdot \boldsymbol{q}^R. \tag{3}$$

In Eq. (3), the residual scalar flux may be restated as $\boldsymbol{q}^R = \widetilde{\boldsymbol{u}\,\phi} - \tilde{\boldsymbol{u}}\,\tilde{\phi}$. The goal of our study is to develop a *data-driven* model for $\boldsymbol{q}^R$ using a predictive deep neural network (DNN) architecture.

2.2 Deep Neural Network (DNN) Closure Modeling

Using a deep feed-forward neural network architecture shown in Fig. 1, we model the residual scalar flux. This model has L fully connected layers each with N^l neurons in the l-th layer, a

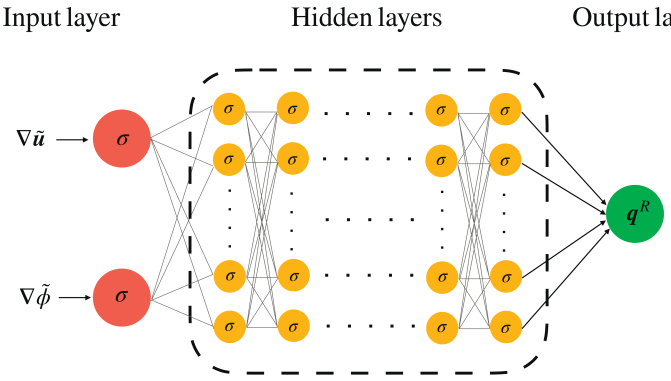

FIG. 1: Schematic of the DNN surrogate model for prediction of SGS scalar flux. σ in the units (neurons) indicates the activation function.

weight tensor of $\boldsymbol{W}^l \in \mathbb{R}^{N^l \times N^{l-1}}$, and a bias vector $\boldsymbol{b}^l \in \mathbb{R}^{N^l}$. For instance, in the l-th layer, y_n^l formulates the output of the n-th neuron as

$$y_n^l = \sigma^l \left(\sum_m^{M^{l-1}} W_{nm}^l \, y_m^{l-1} + b_n^l \right), \tag{4}$$

where σ^l indicates the *activation function*. In this DNN model, $\boldsymbol{y}^0$ represents the input features' layer while $\boldsymbol{y}^L = \boldsymbol{q}_{\mathrm{DNN}}$ is the layer corresponding to prediction of the targets. Through optimization of the DNN's parameters $(\boldsymbol{W}, \boldsymbol{b})$ with FDNS data, $\boldsymbol{q}_{\mathrm{DNN}}$ predicts $\boldsymbol{q}^R = (q_1^R, q_2^R, q_3^R)$. The input features are selected based on the gradients of the filtered (resolved) flow fields, i.e., $\boldsymbol{\nabla}\tilde{\boldsymbol{u}}$ and $\boldsymbol{\nabla}\tilde{\phi}$. Choosing these input features preserves important properties such as frame invariancy and homogeneity of the SGS flux in predictions of the DNN model (Portwood et al., 2021).

3. NUMERICAL AND OPTIMIZATION METHODS

3.1 Simulation Setup for Training/Validation Data

Based on the DNS of passive scalar with a uniform mean gradient of $\boldsymbol{\nabla}\langle\Phi\rangle = (0,1,0)$ in a HIT flow, our required SGS data are constructed. Here, we employ the pseudo-spectral parallel simulation setup elaborated upon in Akhavan-Safaei and Zayernouri (2023a,b). Using the mentioned framework we generate a stationary HIT flow using a computational domain as $\boldsymbol{\Omega} = [0, 2\pi]^3$, which is discretized on a uniform spatial grid of N^3 resolution with $N = 520$. Accordingly, the size of the spatial grid is $\Delta_{\mathrm{DNS}} = 2\pi/N$. A fourth-order Runge-Kutta (RK4) scheme is utilized to perform the time integration with a fixed $\Delta t = 8 \times 10^{-4}$, while the Courant-Friedrichs-Lewy (CFL) number is checked during the simulation to satisfy the $\mathrm{CFL} = (\Delta t/\Delta_{\mathrm{DNS}})\max(|u_1| + |u_2| + |u_3|) < 1$ condition; therefore, the numerical stability is always ensured. Aiming for the Taylor-scale Péclet number ($\mathrm{Pe}_\lambda = \mathrm{Sc}\,\mathrm{Re}_\lambda = 240$) with the Schmidt number ($\mathrm{Sc} = \nu/\mathcal{D} = 1$) and $= \mathrm{Re}_\lambda$ representing the Taylor-scale Reynolds number, at the fully turbulent (statistically stationary) state, we obtain well-resolved velocity and scalar turbulent fields with $k_{\max}\eta_B \approx 1.5$; therefore, we ensure that the small scales in the velocity

and scalar fields are well-resolved (Akhavan-Safaei and Zayernouri, 2023a,b). In order to reach this turbulent state, first the NS equations are resolved from a randomly initialized field for approximately 15 τ_{LE} while an artificial forcing mechanism is enforced to the low wave numbers (energy-containing range) to maintain the turbulent kinetic energy (Akhavan-Safaei and Zayernouri, 2023a,b). Afterwards, we start resolving the AD equation from $\phi_0(\boldsymbol{x}) = 0$ initial fluctuating concentration while imposing a uniform mean gradient. By resolving the NS and AD equations for approximately another 15 large-eddy turnover times, the skewness and flatness records of the fluctuating passive scalar gradient reach a statistically stationary state, specifying the fully-developed turbulent scalar fluctuations. Over a sampling time span in the fully turbulent region, the true values of the residual scalar flux are computed using the box filtering kernel with isotropic filter size Δ, as

$$\mathcal{G}_\Delta(\boldsymbol{x} - \boldsymbol{x}') = \begin{cases} 1/\Delta, & |\boldsymbol{x} - \boldsymbol{x}'| \leq \Delta/2 \\ 0, & \text{Otherwise.} \end{cases} \tag{5}$$

According to the outlined procedure, two other sets of DNS are conducted with (i) $\text{Pe}_\lambda = 240$ and $\text{Sc} = 4$, and (ii) $\text{Pe}_\lambda = 360$ and $\text{Sc} = 1$. In the former case, the spatial discretization consists of the 512^3 uniform grid points with $\Delta t = 8 \times 10^{-4}$, while for the latter case, the spatial grid has a 1024^3 resolution and the temporal resolution is $\Delta t = 5 \times 10^{-4}$.

3.2 Construction of a Spatiotemporal Database

In our study, we examine our model in two filter sizes, $\Delta^* = \Delta/\Delta_{\text{DNS}} = 10, 20$, where the amount of filtered scalar variance for these two filter sizes is 9% and 18%, respectively. Given the time-stationary statistics of our turbulent flow dataset, uniform sampling over the fully turbulent time domain is a fair approach. Therefore, we select a sampling window over approximately 11 τ_{LE} of the fully turbulent state, and uniformly sample 64 time instances. To prepare a randomly generated dataset in the spatial domain (3D periodic cube) for each time sample, we utilize Latin hypercube sampling (LHS) to obtain a randomly drawn index set based on a uniform distribution. This index set returns the location of the spatially sampled data points on the original 3D uniform grid of the filtered DNS solution. For instance, for the case with $\Delta^* = 10$, a set of 23^3 spatially scattered data points is sampled out of the full spatial dataset of 52^3 resolution. This reduction of the data at a sampled time frame helps to construct an unbiased database in space and time (using sampled time snapshots of FDNS data) for training/validation of the DNN model; thus we avoid an unnecessary large size of the entire database. As a result, this spatiotemporal sampling procedure yields a reference FDNS database of approximately 780,000 samples. We hold 5% of the data out for out-of-sample validation during the optimization procedure. Figure 2 illustrates this sampling framework.

3.3 DNN Specifications and Optimization

In this study, the DNN model consists of six nonlinear hidden layers with rectified linear unit (ReLU) activation functions as

$$\sigma(y) = \max(y, 0), \tag{6}$$

and each hidden layer contains 32 neurons (units). Unlike the nonlinear hidden layers, the output layer that returns the DNN approximation for $\boldsymbol{q}^R$ is set to be linear. As a result, this DNN model

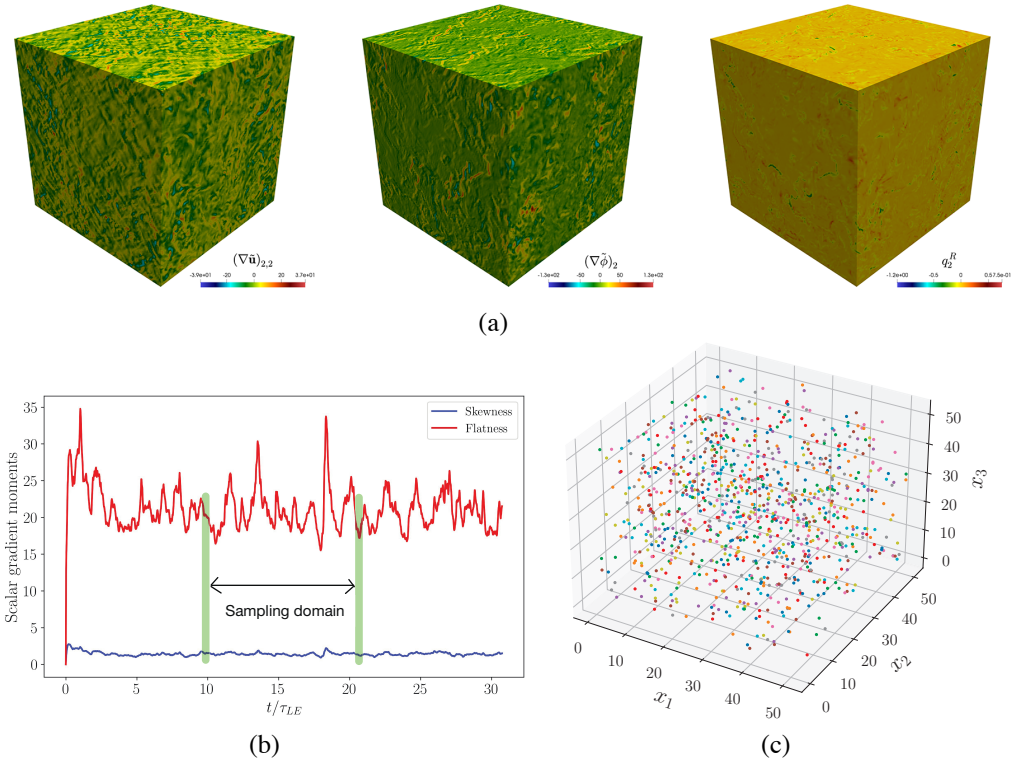

FIG. 2: Spatiotemporal sampling framework for construction of training/validation dataset from FDNS results for scalar transport regime of $\mathrm{Pe}_\lambda = 240$, $\mathrm{Sc} = 1$. (a) Full-size spatial data (input and target features) obtained from FDNS at a sampled time instance. For $\boldsymbol{\nabla}\tilde{\boldsymbol{u}}$ tensor only the component $_{2,2}$, and for $\boldsymbol{\nabla}\tilde{\phi}$ and $\boldsymbol{q}^R$ vectors only the second components are shown on the 3D spatial domain, (b) temporal sampling domain, shown on statistical records of $\boldsymbol{\nabla}\phi$, (c) spatial index sampling to be applied on all time-sampled full-size 3D data.

yields 6000 trainable parameters and these model parameters are initialized based on a uniform distribution prior to model optimization. In order to train the DNN model, the database from spatiotemporal sampling is utilized to minimize the mean-squared error (MSE) loss function:

$$\mathrm{MSE} = \frac{1}{N}\sum_{i}^{N}(\hat{\boldsymbol{q}}_i - \boldsymbol{q}_i)^2. \tag{7}$$

In Eq. (7), $\boldsymbol{q}_i$ denotes the DNN prediction for each sample and $\hat{\boldsymbol{q}}_i$ is the reference labeled data from FDNS. Using the AdamW optimizer with a learning rate of $lr = 10^{-4}$, we manage to minimize the MSE loss function with respect to the model parameters (weights and biases) of the DNN as we utilize PyTorch library (Paszke et al., 2017). Using a batch size of 256 data points while we shuffle the training data in the data loader, we perform model optimization for 150 epochs (with each epoch consisting of 2800 optimizer iterations). At the end of each epoch we perform out-of-sample validations with the updated model. The same batch size is selected for the validation data and the resulting MSE evaluations are averaged over the validation batches. For the DNN model optimized based on the dataset of $\Delta^* = 10$, Fig. 3(a) shows the MSE

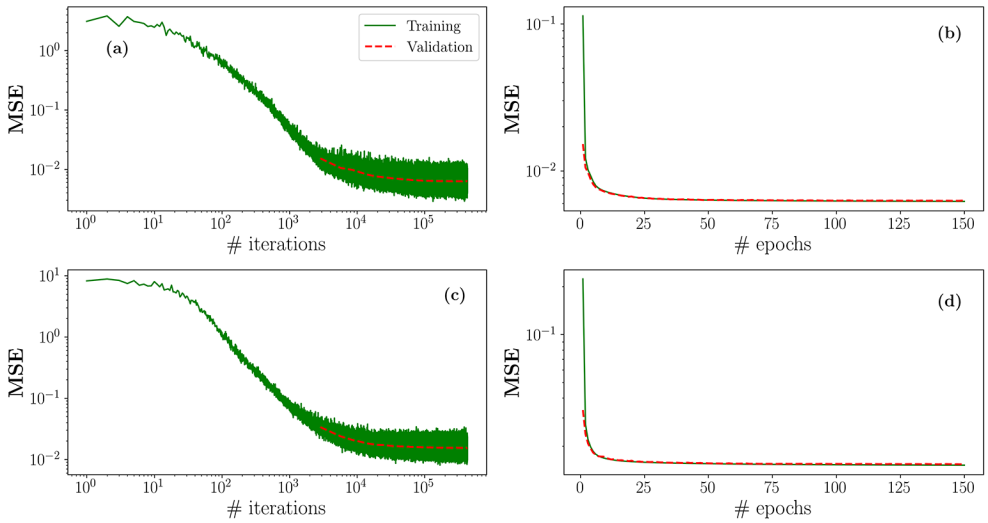

FIG. 3: Training/validation of the DNN surrogate model. Training MSE value after each gradient descent iteration for (a) $\Delta^* = 10$, and (c) $\Delta^* = 20$. Averaged training MSE over evaluations in each epoch for (b) $\Delta^* = 10$, and (d) $\Delta^* = 20$. All of the depicted validation MSEs in (a)–(d), shown with dashed red lines, are averaged MSE values over the validation data batches.

evaluations of the training iterations, and Fig. 3(b) depicts their averaged value over the iterations in each epoch. In both of these plots, the averaged MSE evaluations over the validation data batches are shown with a dashed line. At the end, the convergence of the MSE is achieved, and it is shown that the overfitting is avoided as the averaged MSE values for training and validation are the same over the entire optimization procedure. We observe the same behavior in model training/validation using the $\Delta^* = 20$ filtered dataset as shown in Figs. 3(c) and 3(d).

3.4 Transfer Learning for Generalization to Other Transport Regimes

Due to the increasing computational cost of high-resolution simulations as the Pe_λ number increases in scalar transport, it is not always feasible to obtain a sufficiently large dataset to properly train a DNN to reliably model the SGS flux. On the other hand, it is widely recognized that the data-driven surrogate models for physical systems do not generalize well to accurately predict in physical regimes outside of their training data. Transfer learning (TL) provides a practical approach to this problem, and lets us adjust our pretrained surrogate model (from a physical regime with abundant training data) to work well on another physical regime with less available data. Transfer learning aims to retrain the very final layers of the pretrained DNN to adjust the model parameters learning *specific features* that belong to the new physical regime utilizing its available data. The rationale behind this retraining step is that the *general features* that are common among the initial and new physical regimes are learned in the earlier hidden layers (in the pretrained model), while the *specific features* are learned in the later hidden layers. Therefore, one can freeze the model parameters associated with the earlier layers in the retraining procedure (by setting them as nontrainable parameters in the optimization), and only update those associated with the later hidden layers using the data for the new physical regime. Figure 4 provides a schematic view of the transfer learning in our DNN modeling for SGS scalar flux.

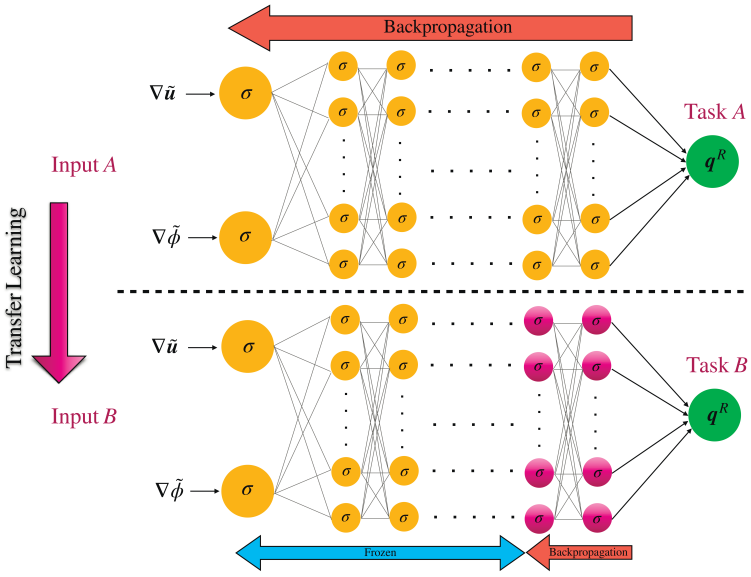

FIG. 4: Schematic of the transfer learning from a DNN model pretrained on data for Task A to a DNN model for Task B. In the initial training for Task A (upper model in the plot), all of the model parameters are optimized, while in the training the pretrained model for Task B, the model parameters in the earlier layers are fixed, and the ones in the final layers are optimized based on the data for Task B (lower model in the plot).

In the previous section, we presented a trained DNN model using the FDNS dataset from a turbulent scalar transport at regime $\mathrm{Pe}_\lambda = 240$ and $\mathrm{Sc} = 1$ (as described in Section 3.2). Taking that pretrained model, we manage to retrain it for transfer learning to the following scalar transport regimes for the specified filter sizes:

- $\mathrm{Pe}_\lambda = 240$ and $\mathrm{Sc} = 4$ ($\Delta^* = 10, 20$),
- $\mathrm{Pe}_\lambda = 360$ and $\mathrm{Sc} = 1$ ($\Delta^* = 8, 16$).

The data for retraining these two transport regimes were obtained from filtering DNS results where the small scales of passive scalar are sufficiently resolved ($k_{\max}\,\eta_B \approx 1.5$). The details of these simulations are similar to the description provided in Section 3.1. In the retraining step for each transport regime, we construct training/validation datasets that contain approximately 5% of sampled data points we utilized for the pretrained model ($\mathrm{Pe}_\lambda = 240$ and $\mathrm{Sc} = 1$). The construction of these training/validation datasets is based on the procedure we explained in Section 3.2; however, in these datasets, we only use four sampled time instances of full-sized 3D FDNS data for spatial sampling. The batch size and learning rate are chosen similar to optimization of the pretrained model, yet in the retraining steps we only aim for 50 epochs. As a result, the number of gradient descent iterations in the retraining steps are approximately 2.5% of iterations in the pretraining. Moreover, in the retraining steps, we optimize the model parameters in the final two layers of the pretrained DNN model, and the rest of the model parameters in other layers remain fixed. For example, Fig. 5 shows the records of the loss function for the retraining and validation for the transfer learning to the SGS model at transport regime $\mathrm{Pe}_\lambda = 360$, $\mathrm{Sc} = 1$. Figures 5(a) and 5(b) show the convergence of the MSE for the model retrained based on data

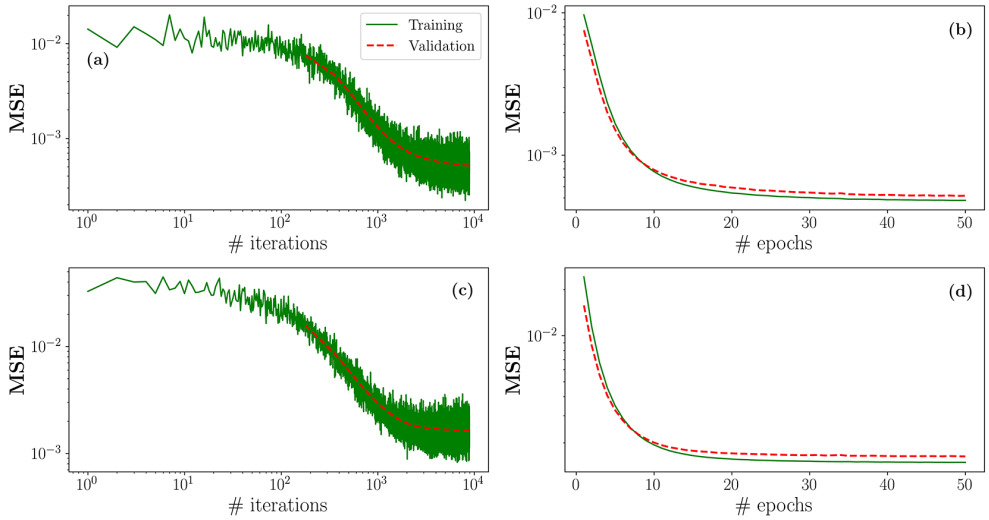

FIG. 5: Training/validation for the transfer learning from the pretrained DNN model DNN using data of $\mathrm{Pe}_\lambda = 240$, $\mathrm{Sc} = 1$ regime (see Fig. 3) to a DNN model for $\mathrm{Pe}_\lambda = 360$, $\mathrm{Sc} = 1$ regime. Training MSE value after each gradient decent iteration for (a) $\Delta^* = 8$, and (c) $\Delta^* = 16$. Averaged training MSE over evaluations in each epoch for (b) $\Delta^* = 8$, and (d) $\Delta^* = 16$. All of the depicted validation MSEs in (a)–(d), shown with dashed red lines, are averaged MSE values over the validation data batches.

with $\Delta^* = 8$ (pretrained model utilized data with $\Delta^* = 10$), while Figs. 5(c) and 5(d) illustrate the converged model when $\Delta^* = 16$ in the retraining data (pretrained model utilized data with $\Delta^* = 20$). Similar to the pretrained models presented in Fig. 3, close behavior of the training and validation MSE records during the retraining steps ensures that the transfer learned models are not overfitted.

4. MODEL ASSESSMENTS IN THE INFERENCE MODE

In this section, we manage to examine the performance of the trained DNN models described in the previous section in the inference of the SGS and resolved flow quantities. Here we employ the full-size 3D time instances of flow fields $(\tilde{\boldsymbol{u}}, \tilde{\phi})$ coming from either the FDNS solution (*a priori* test) or from an LES that utilizes the SGS model (*a posteriori* test). In both of these tests, we assess the performance of the DNN model against the two traditional closure models for the SGS scalar flux:

- Prandtl-Smagorinsky (PSM) model,
- Scalar asymptotic gradient (SAG) model.

In the PSM model, the SGS scalar flux is determined based on the eddy-diffusivity concept, and is written as

$$\boldsymbol{q}_{\mathrm{PSM}} = -\mathcal{D}_t \boldsymbol{\nabla} \tilde{\phi}, \tag{8}$$

where $\mathcal{D}_t = \nu_t / \mathrm{Pr}_t$ denotes the turbulent diffusivity. The constant turbulent Prandtl number, Pr_t, relates $\mathcal{D}_t$ to the turbulent viscosity, $\nu_t = (C_s\,\Delta)^2\,\|\tilde{\boldsymbol{S}}\|$ defined by the Smagorinsky model. Here C_s is the Smagorinsky constant and for the filtered strain-rate tensor $\tilde{\boldsymbol{S}} = 1/2\big(\boldsymbol{\nabla}\tilde{\boldsymbol{u}} + \boldsymbol{\nabla}\tilde{\boldsymbol{u}}^{\mathrm{T}}\big)$,

$\|\tilde{\boldsymbol{S}}\| = \sqrt{2\,\tilde{\boldsymbol{S}} : \tilde{\boldsymbol{S}}}$. For passive scalars in homogeneous turbulence, if the filter size is selected within the inertial subrange, $C_s \approx 0.17$ and $\mathrm{Pr}_t \approx 0.5$ (Antonopoulos-Domis, 1981; Lilly, 1967). Therefore, we adopt these values in our study.

In the SAG model, the residual scalar flux is approximated based on a Taylor series expansion of the filtering operation (Bedford and Yeo, 1993; Clark et al., 1979), and is represented as

$$\boldsymbol{q}_{\mathrm{SAG}} = \frac{\Delta^2}{12}\,\boldsymbol{\nabla}\tilde{\phi}\cdot\boldsymbol{\nabla}\tilde{\boldsymbol{u}}. \tag{9}$$

Although this model has shown a good structural performance in reproducing the SGS flux from FDNS flow fields, it has been identified to be significantly underdissipative, especially when the filter size is sufficiently large compared to the dissipation scales (Leonard, 1975). Therefore, conducting a stable LES (in terms of numerical time integration) using the SAG model is known to require special modifications such as utilizing a "clipping" technique (Lu and Porté-Agel, 2013; Vreman et al., 1997) or combining it with an eddy-diffusivity model (Clark et al., 1979).

4.1 *A Priori* Test

In the *a priori* testing of developed DNN models, we employ full-size 3D FDNS solutions for $\tilde{\phi}$ and $\tilde{\boldsymbol{u}}$. We utilize ten time instances sampled from approximately 5 τ_{LE} away from the time domain where we obtained our training/validation datasets. Using these time instances of FDNS data and the SGS model, we predict the SGS flux on the 3D filtered domain. Considering the mean scalar gradient $\boldsymbol{\nabla}\langle\tilde{\Phi}\rangle$ as the anisotropic source of turbulence production for passive scalar, we categorize the predicted SGS flux terms into *perpendicular* ($q_{\perp\boldsymbol{\nabla}\langle\tilde{\Phi}\rangle} = q_1^R, q_3^R$) and *parallel* ($q_{\parallel\boldsymbol{\nabla}\langle\tilde{\Phi}\rangle} = q_2^R$) components with respect to the mean scalar gradient direction. Moreover, we compute the SGS dissipation of filtered scalar variance,

$$\Pi = -\boldsymbol{q}^R\cdot\boldsymbol{\nabla}\tilde{\phi}, \tag{10}$$

as an indicator for capability of the SGS flux model in reproducing the backward scattering of the scalar variance. The backward scattering phenomenon is the transfer of turbulent intensity from unresolved (SGS) scales to the resolved (filtered) scale, and it is identified with the negative values appearing in the true Π (obtained from FDNS).

Once we compute $\boldsymbol{q}^R$ and Π for each time instance using the SGS model, we take the mean squared error of each quantity with respect to its FDNS solution. In order to average the computed MSEs over the time instances we sampled, each MSE is normalized by the variance of its FDNS solution. Averaged normalized MSEs for $q_{\perp\boldsymbol{\nabla}\langle\tilde{\Phi}\rangle}$, $q_{\parallel\boldsymbol{\nabla}\langle\tilde{\Phi}\rangle}$, and Π are presented for each model. For $q_{\perp\boldsymbol{\nabla}\langle\tilde{\Phi}\rangle}$, the averaged normalized MSE is the mean value of MSEs for both perpendicular components. Finally, the probability distribution function (PDF) of reconstructed Π from the SGS model is compared to the true PDF from FDNS. In the following, we present these metrics for the base SGS model we developed in Section 3.3 as well as the two transfer learned models to other transport regimes we presented in Section 3.4.

4.1.1 Base Model: $Pe_\lambda = 240$, $Sc = 1$

Here we compare the performance of the developed DNN model in Section 3.3 to the performance of the SAG and PSM models. Our comparison is based on the time-averaged normalized MSE metric for predicted $q_{\perp\boldsymbol{\nabla}\langle\tilde{\Phi}\rangle}$, $q_{\parallel\boldsymbol{\nabla}\langle\tilde{\Phi}\rangle}$, and Π at $\Delta^* = 10, 20$ (see Fig. 6). Figure 6(a) shows

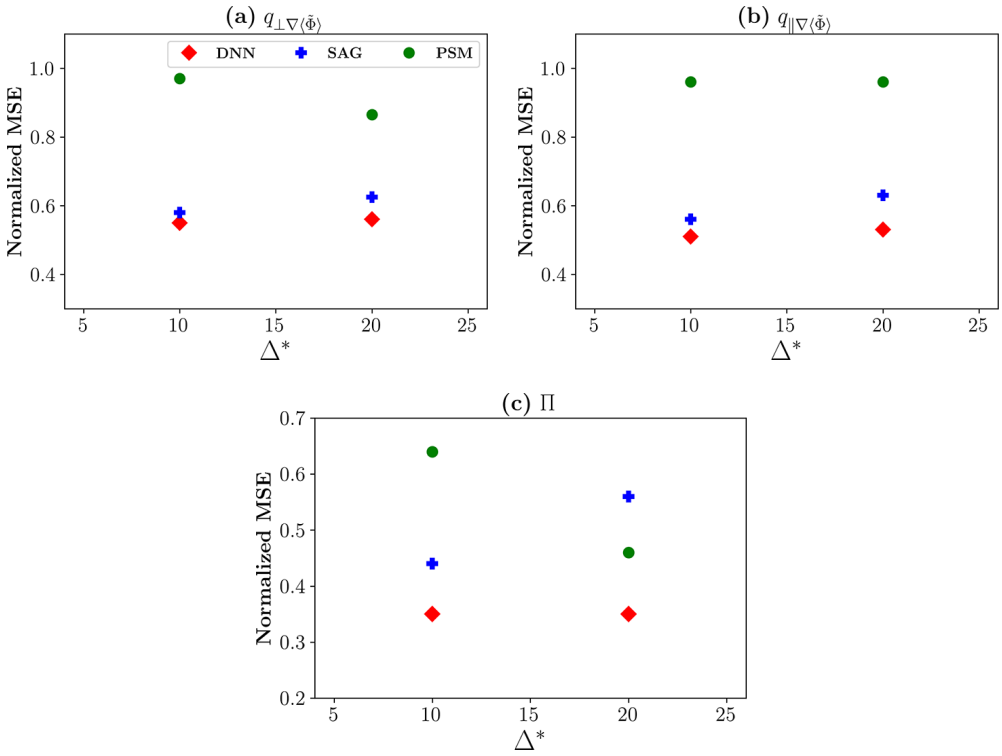

FIG. 6: *A priori* estimation (base model: $\mathrm{Pe}_\lambda = 240$, $\mathrm{Sc} = 1$) of the time-averaged normalized mean squared error of (a) $q_{\perp\boldsymbol{\nabla}\langle\tilde{\Phi}\rangle}$, (b) $q_{\parallel\boldsymbol{\nabla}\langle\tilde{\Phi}\rangle}$, (c) Π for each SGS flux model at each filter size. The normalization is done by the variance of the FDNS solution for each quantity at its filter size, and the time averaging is carried out using ten sampled time instances.

that at $\Delta^* = 10$, the DNN model prediction of $q_{\perp\boldsymbol{\nabla}\langle\tilde{\Phi}\rangle}$ has 5.5% less error compared to the SAG model and 76.4% less error compared to the PSM model, while at $\Delta^* = 20$, its prediction returns 11.6% less error compared to the SAG model and 54.3% less than the error in the PSM model's prediction. In Fig. 6(b), we see that at $\Delta^* = 10$, the DNN model predicts $q_{\parallel\boldsymbol{\nabla}\langle\tilde{\Phi}\rangle}$ with 9.8% less error compared to the SAG model and with 88.2% less error compared to the PSM model, and when $\Delta^* = 20$, the DNN model yields 18.9% less error compared to the SAG model and its error is 81.1% less than what we observe in the prediction of the PSM model. Moreover, from Figs. 6(a) and 6(b) one can realize from the error level in prediction of $q_{\perp\boldsymbol{\nabla}\langle\tilde{\Phi}\rangle}$, $q_{\parallel\boldsymbol{\nabla}\langle\tilde{\Phi}\rangle}$ components, that the DNN model exhibits a fairly consistent behavior; however, such consistency is not necessarily observed in the error level of the predicted flux components in other two models, especially at $\Delta^* = 20$. Finally, for the predicted Π, Fig. 6(c) illustrates that the DNN model has 25.7% less error compared to the SAG model and it returns 82.9% less error compared to the PSM model. However, at $\Delta^* = 20$, the predicted Π by the DNN model returns 60% lower error compared to the SAG model and 31.4% less than what the PSM model predicts. Here, it is noticeable that when the filter size increases the error level in predicted Π form the SAG model increases while the PSM model exhibits an opposite behavior. Nevertheless, the DNN model exhibits a consistent behavior over the filter sizes, while it always maintains error level in the predicted Π significantly lower than the other two models. Given this observation, Fig. 7

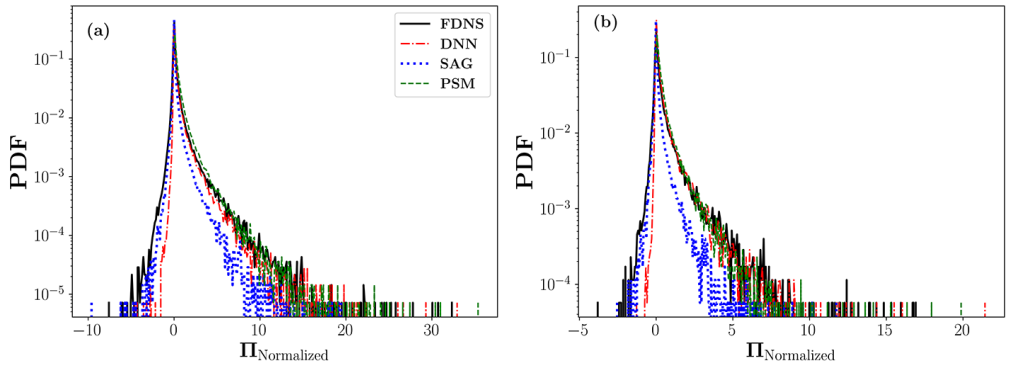

FIG. 7: *A priori* assessment of the PDFs of normalized SGS dissipation obtained from the DNN (base model: $\text{Pe}_\lambda = 240$, $\text{Sc} = 1$), SAG, and PSM models compared to the PDF computed from FDNS for (a) $\Delta^* = 10$, (b) $\Delta^* = 20$. Π for all the PDFs is normalized by the standard deviation of its FDNS solution.

shows that for both filter sizes the DNN model provides a fairly good prediction for the PDF of SGS dissipation of scalar variance compared to the true PDF from FDNS, while the SAG model underpredicts the events in the right side of the PDF (yielding and underdissipated SGS model), and the PSM model provides an overdissipated model since it cannot model the negative-valued events associated with the backward scattering of resolved scalar variance.

4.1.2 Transfer Learned Models

In this section, we make a comparison between the time-averaged normalized MSEs of $q_{\perp\boldsymbol{\nabla}\langle\tilde{\Phi}\rangle}$, $q_{\parallel\boldsymbol{\nabla}\langle\tilde{\Phi}\rangle}$, and Π obtained from *a priori* testing of the TL models we presented in Section 3.4 with the error metrics when we only employ the base (pretrained) model for the predictions in the new scalar transport regime.

Case (I) $\text{Pe}_\lambda = 240$, $\text{Sc} = 4$. In Figs. 8(a) and 8(b), we can see that at $\Delta^* = 10$, the TL model decreases the error in predictions of $q_{\perp\boldsymbol{\nabla}\langle\tilde{\Phi}\rangle}$ and $q_{\parallel\boldsymbol{\nabla}\langle\tilde{\Phi}\rangle}$ by 12% and 46.5%, respectively, compared to the predictions of the pretrained model. Similar comparison at $\Delta^* = 20$ indicates that the TL model returns 27.3% less error for predicted $q_{\perp\boldsymbol{\nabla}\langle\tilde{\Phi}\rangle}$, and 93.2% less error for the predicted $q_{\parallel\boldsymbol{\nabla}\langle\tilde{\Phi}\rangle}$ compared to the pretrained DNN model. Therefore, one can realize that the retraining procedure plays a significant role in improving the prediction of the parallel component of the SGS flux. Moreover, Fig. 8(c) shows that the predicted Π from the pretrained DNN model has 2.6 times higher error at $\Delta^* = 10$ and 3.3 times higher error at $\Delta^* = 20$ compared to the TL model's predictions. In fact, Figs. 9(a) and 9(b) illustrate that these improvements in the prediction of Π mainly originate from correcting the overprediction of forward scattering of SGS dissipation after retraining the DNN model for the higher Sc number regime.

Case (II) $\text{Pe}_\lambda = 360$, $\text{Sc} = 1$. Considering Figs. 10(a) and 10(b), we observe that at both $\Delta^* = 8, 16$ the TL model reduces the error level in the prediction of $q_{\perp\boldsymbol{\nabla}\langle\tilde{\Phi}\rangle}$ by 22 times compared to the pretrained model. However, in the prediction of $q_{\parallel\boldsymbol{\nabla}\langle\tilde{\Phi}\rangle}$, the error level in the pretrained model's performance is reduced by 33 times at $\Delta^* = 8$, and by 37 times when $\Delta^* = 16$ after retraining. Similar to the Case (I), we recognize that TL is playing a more corrective role in enhancing the performance of the pretrained DNN model in prediction of $q_{\parallel\boldsymbol{\nabla}\langle\tilde{\Phi}\rangle}$ compared to $q_{\perp\boldsymbol{\nabla}\langle\tilde{\Phi}\rangle}$. Concerning the prediction of Π, Fig. 10(c) points out that the pretrained DNN model

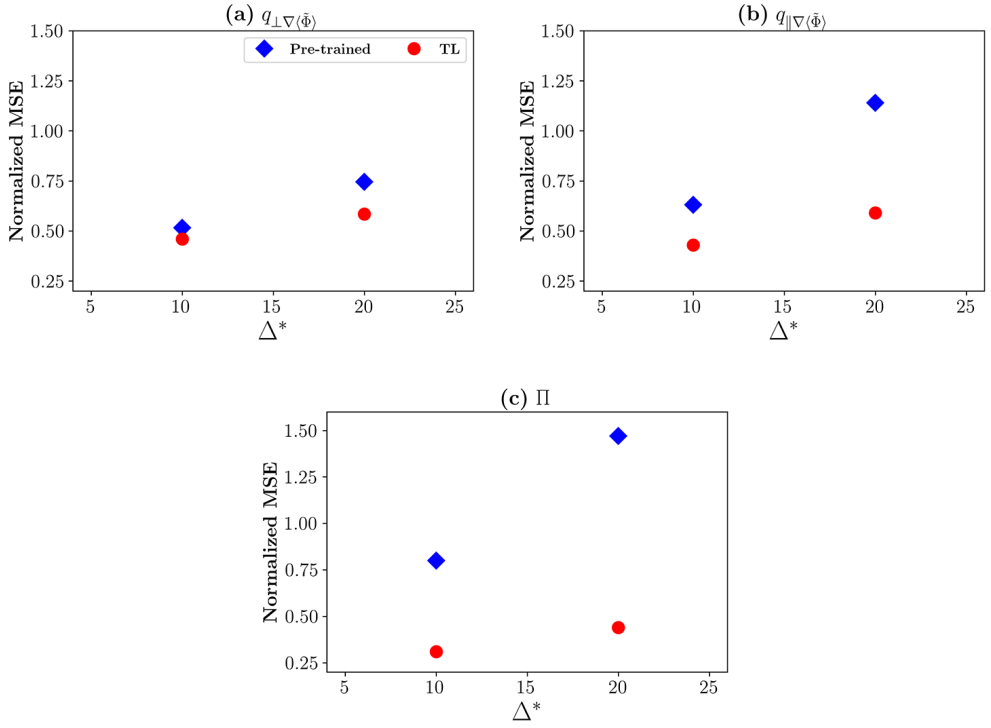

FIG. 8: *A priori* estimation of transfer learned model ($\mathrm{Pe}_\lambda = 240, \mathrm{Sc} = 4$) from the base pretrained model. The subplots show the time-averaged normalized mean squared error of (a) $q_{\perp\boldsymbol{\nabla}\langle\tilde{\Phi}\rangle}$, (b) $q_{\parallel\boldsymbol{\nabla}\langle\tilde{\Phi}\rangle}$, (c) Π for pretrained and TL SGS flux model at each filter size. The normalization is done by the variance of FDNS solution for each quantity at its filter size, and the time averaging is carried out using 10 sampled time instances.

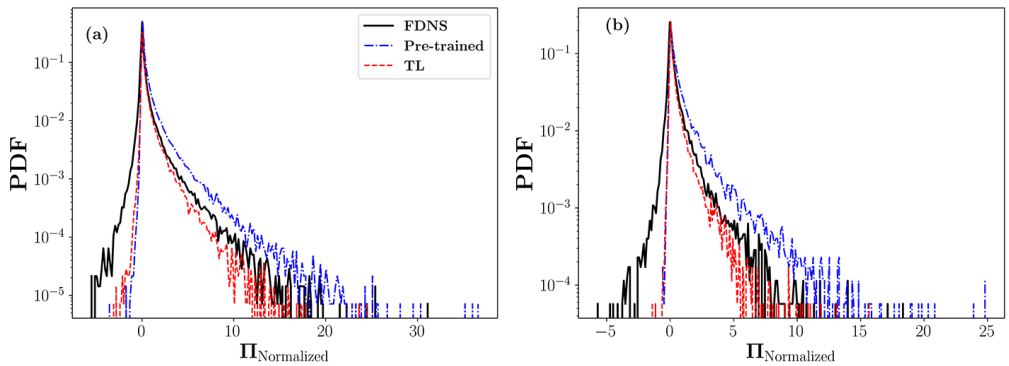

FIG. 9: *A priori* assessment of the PDFs of normalized SGS dissipation obtained from the transfer learned model (for $\mathrm{Pe}_\lambda = 240, \mathrm{Sc} = 4$), and pretrained model (using $\mathrm{Pe}_\lambda = 240, \mathrm{Sc} = 1$ data) compared to the PDF computed from FDNS for (a) $\Delta^* = 10$, (b) $\Delta^* = 20$. Π for all the PDFs is normalized by the standard deviation of its FDNS solution.

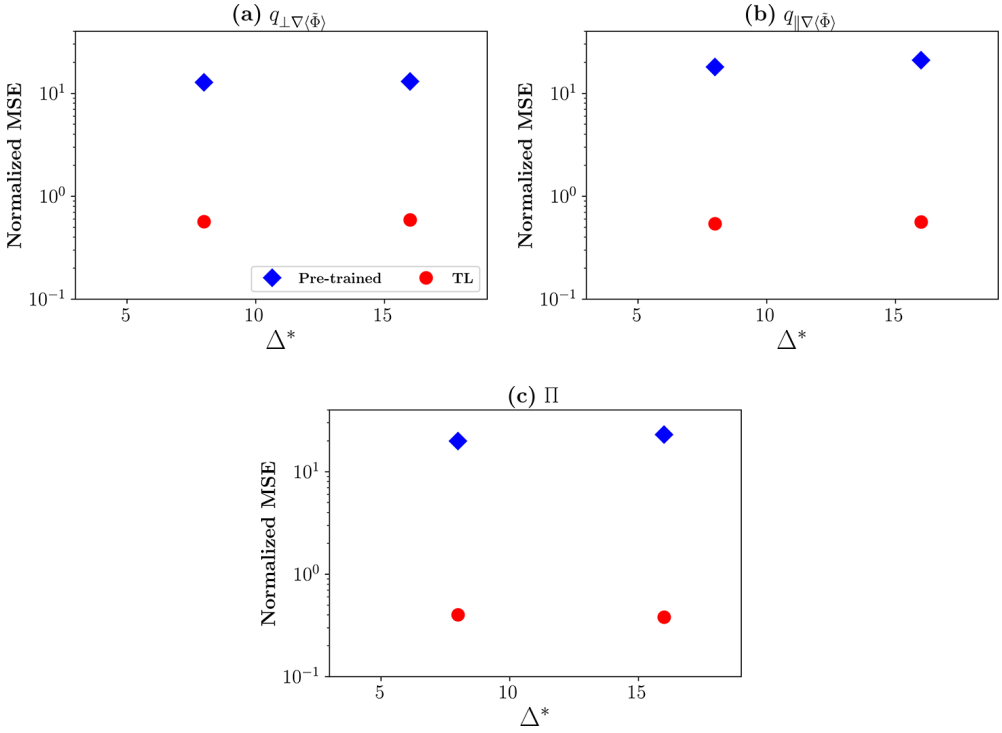

FIG. 10: *A priori* estimation of transfer learned model ($\mathrm{Pe}_\lambda = 360, \mathrm{Sc} = 1$) from the base pretrained model. Similar to Fig. 8, the subplots show the time-averaged normalized mean squared error of (a) $q_{\perp\nabla\langle\tilde{\Phi}\rangle}$, (b) $q_{\parallel\nabla\langle\tilde{\Phi}\rangle}$, (c) Π for pretrained and TL SGS flux model at each filter size. Using the variance of the FDNS solution for each quantity at its own filter size, the MSE values are normalized while the time averaging is performed on ten sampled time instances.

has 49.5 times higher error at $\Delta^* = 8$ and 60.5 times higher error at $\Delta^* = 16$ compared to the TL model. Looking at Figs. 11(a) and 11(b), it is discernible that in the case of TL from the $\mathrm{Pe}_\lambda = 240$ regime to $\mathrm{Pe}_\lambda = 360$, the retrained DNN model not only corrects the considerable overprediction of the forward scattering phenomenon appearing in the pretrained model (see the positive sides of the PDFs), it also improves its overprediction of backward scattering of the SGS dissipation (see the negative sides of the PDFs).

4.2 *A Posteriori* Test

In this section, our main objective is to examine the performance of the developed DNN model when it is utilized as the closure model in an LES setting, a.k.a. *a posteriori* testing. Therefore, we seek to understand if the LES-resolved transport fields (in our case $\tilde{\phi}$) are temporally and spatially following their expected true solution which comes from explicit filtering of the DNS outputs.

To fulfill this goal, we focus on testing the base DNN model we developed in Section 3.3 on the turbulent regime of $\mathrm{Pe}_\lambda = 240$ and $\mathrm{Sc} = 1$. In order to perform the large-eddy simulations on the problem setting introduced in Section 2.1, we employ the pseudo-spectral solver developed in Akhavan-Safaei and Zayernouri (2023a). We modify this DNS framework to account for

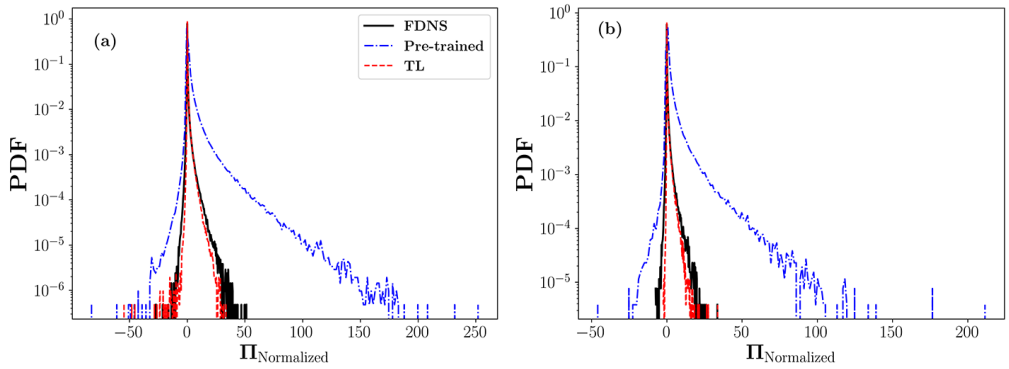

FIG. 11: *A priori* assessment of the PDFs of normalized SGS dissipation obtained from the transfer learned model (for $\text{Pe}_\lambda = 360$, $\text{Sc} = 1$), and pretrained model (using $\text{Pe}_\lambda = 240$, $\text{Sc} = 1$ data) compared to the PDF computed from FDNS for (a) $\Delta^* = 8$, (b) $\Delta^* = 16$. Π for all the PDFs is normalized by the standard deviation of its FDNS solution.

utilization of multiple SGS flux models of interest for $\boldsymbol{q}^R$. In our current study, we utilize the DNN model as well as the PSM and SAG models. Typically, both SGS stresses and fluxes in the filtered NS and AD equations are modeled and utilized in an LES setting. Thus, a specific choice of SGS model for $\boldsymbol{\tau}^R$ could have substantial effects on the solution resolved scalar concentration field (Portwood et al., 2021). Similar to earlier works in literature for the pure purpose of SGS flux model assessment (Portwood et al., 2021; Vollant et al., 2016), we choose to freeze these potentially dominant effects originating from using an SGS stress model and instead directly make use of FDNS solution of velocity field. Therefore, we can only focus on the performance of the utilized SGS flux model in Eq. (3). Accordingly, in our numerical setup, we resolve the NS equations based on the DNS required resolution. By explicit filtering of $\boldsymbol{u}$ after each time step, we obtain the velocity field over the desired LES grid, and use it to solve Eq. (3) in the time-integrator steps. The initial conditions ($\tilde{\phi}_0$ and $\tilde{\boldsymbol{u}}_0$) for our LES tests are adopted from explicit filtering of a well-resolved and fully developed DNS solution, at $\text{Pe}_\lambda = 240$, $\text{Sc} = 1$ as described in Section 3.1.

As we mentioned earlier, for the SAG model we need to employ special treatment for numerical stabilization in LES. Here, we utilize the clipping technique during the time integration, which is based on setting the predicted $\boldsymbol{q}_{\text{SAG}} = 0$ before feeding it to the time integrator whereever $\Pi_{\text{SAG}} < 0$. This essentially helps to improve the extremely underdissipated behavior of the SAG model only for numerical stability (Lu and Porté-Agel, 2013; Vreman et al., 1997). In our tests, the time integration of filtered AD equation (3) is conducted for $5\,\tau_{\text{LE}}$. In the following, we look at two significant indicators in the performance of the SGS scalar flux model.

4.2.1 Temporal Records of Resolved Scalar Variance

Precise time evolution of the resolved scalar variance, $\langle\tilde{\phi}^2\rangle$, is a principal indicator for reliable prediction of the turbulent intensity in LES. In our tests, we manage to record this quantity for the utilized SGS models over the simulation time, while the reference record is obtained from filtering the DNS solution. Here, we compute the relative error between the variance record from the LES with an SGS model $\langle\tilde{\phi}^2\rangle_{\text{LES}}$ and from the FDNS $\langle\tilde{\phi}^2\rangle_{\text{FDNS}}$. Figure 12(a) shows the

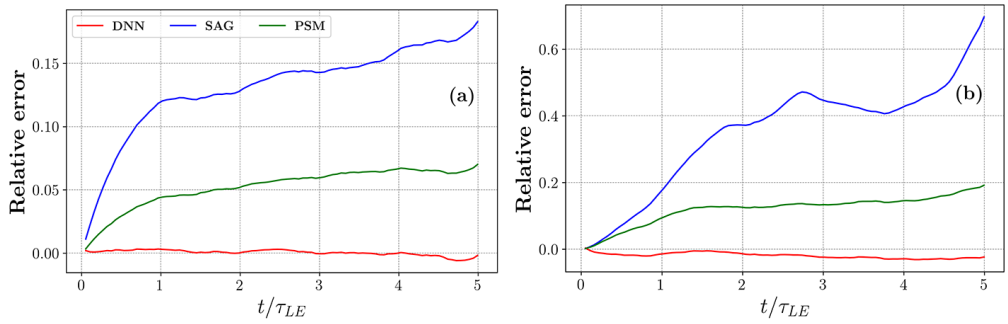

FIG. 12: Relative error (with respect to the FDNS) in the records of resolved-scale scalar variance predicted in the LES tests with the DNN, SAG, and PSM models at (a) $\Delta^* = 10$, and (b) $\Delta^* = 20$. The LES tests are conducted in the $\mathrm{Pe}_\lambda = 240$ and $\mathrm{Sc} = 1$ regime, and the utilized DNN model is trained/validated with the FDNS dataset obtained in the same turbulent regime.

recorded errors for LES tests at $\Delta^* = 10$, and Fig. 12(b) reports it for simulations at $\Delta^* = 20$. At both of the filter sizes, the DNN model has a remarkable performance over time compared to the SAG and PSM models; however, in $\Delta^* = 20$, slight underprediction of resolved-scale variance is observed. It is obvious that in both of the filter sizes, the SAG model exhibits quite a poor performance, especially as time evolves, and this behavior is worse as the filter size increases. On the other hand, the dissipative nature of the PSM model seems to be helpful in terms of a better performance of the model in LES; nevertheless, its considerable errors in prediction of Π (as we observed in Section 4.1) seem to penalize its accuracy in the *a posteriori* tests.

4.2.2 Resolved-Scale Scalar Structure Functions (Two-Point Statistics)

Assessment of an SGS model performance in prediction of the nonlocal behavior of the turbulent transport regime is one of the ultimate tests in the evolution of a turbulent field during LES (Meneveau, 1994). The structure functions of the resolved scalar field are robust two-point statistical measures that return nth-order statistics of resolved-scale scalar increments at a specific direction where $2 \leq n$ (Warhaft, 2000). These structure functions of order n are defined as

$$\langle \delta_r \tilde{\phi}_L^n \rangle = \left\langle \left[\tilde{\phi}_L(\boldsymbol{x} + r\boldsymbol{e}_L) - \tilde{\phi}_L(\boldsymbol{x}) \right]^n \right\rangle; \quad n = 2, 3, \ldots, \tag{11}$$

where r is the size of spatial increment, L represents the *longitudinal* direction (the direction along the imposed uniform mean-gradient) (Akhavan-Safaei and Zayernouri, 2023b; Iyer and Yeung, 2014; Portwood et al., 2021), and $\boldsymbol{e}_L$ specifies the unit vector along the longitudinal direction. Normalizing the $\langle \delta_r \tilde{\phi}_L^n \rangle$ computed from LES (carried out with an SGS model) with its FDNS measurement would provide an informative metric for comparing the SGS models. Focusing on the temporal region $4 < t/\tau_{LE} \leq 5$ where the LES solution has undergone a fairly long time integration, we select uniformly distributed samples of full-size 3D $\tilde{\phi}(\boldsymbol{x})$ in time and compute second- and third-order structure functions. Since we are simulating a statistically-stationary problem, we are allowed to take the temporal average of these normalized structure functions obtained from the sampled structure functions. Therefore, we have a robust indicator measure to examine the performance of each SGS model in predicting nonlocal and high-order statistics of resolved-scale scalar field in a long-time integrated LES. Figure 13 shows this time-averaged $\langle \delta_r \tilde{\phi}_L^n \rangle$ against the normalized spatial shift, r/η_B, for the DNN, SAG, and PMS

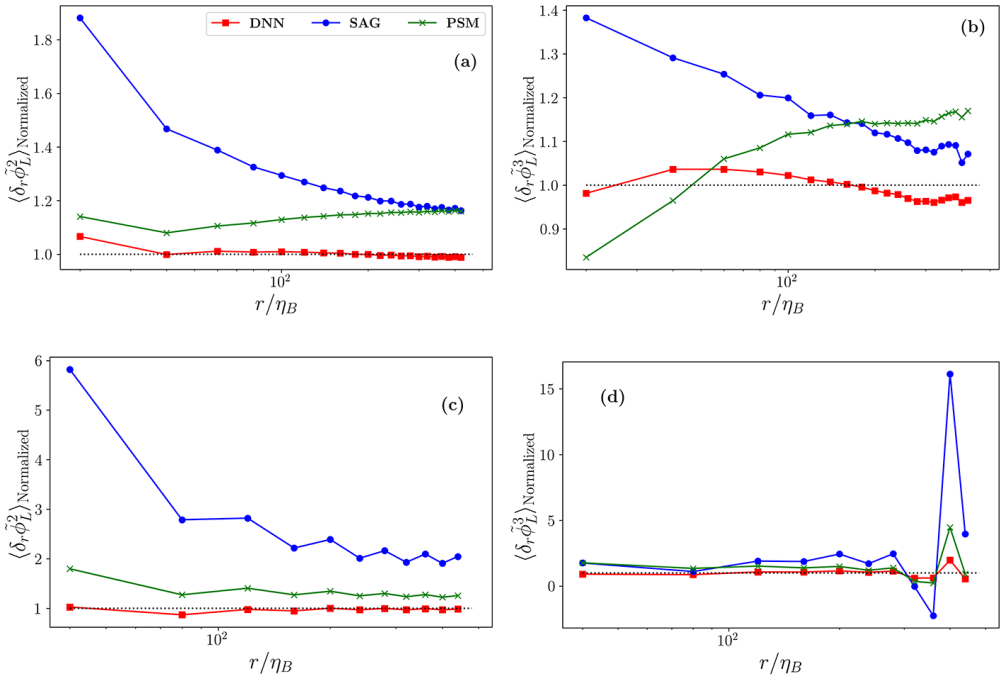

FIG. 13: Time-averaged normalized $\langle \delta_r \tilde{\phi}_L^n \rangle$ from LES with different SGS models with respect to the FDNS result as reference solution. For $\Delta^* = 10$, (a) second-order and (b) third-order illustrate longitudinal structure functions. For $\Delta^* = 20$, (c) second-order and (d) third-order show the longitudinal structure functions. The time-averaging of these two-point structure functions is done over $4 < t/\tau_{LE} < 5$. The spatial shift r equals the filter size of the LES.

models. Figures 13(a) and 13(c) show the relative errors in the even-order structure functions, where the DNPS model has a considerably better performance. Figures 13(a) and 13(c) show the normalized second-order structure functions, for $\Delta^* = 10$ and $\Delta^* = 20$, respectively. Remarkably, we observe that for both of the filter sizes predictions of the DNN model closely follow the temporally-averaged behavior of $\langle \delta_r \tilde{\phi}_L^2 \rangle_{\text{FDNS}}$ over the entire range of spatial shift. On the other hand, we realize that the SAG model has a significant overprediction of $\langle \delta_r \tilde{\phi}_L^2 \rangle$, especially for $r < 100\eta_B$, which becomes worse for the larger filter size. Even in longer range spatial shifts where we observe improvement in its predictions, there is an asymptotic 20% overprediction at $\Delta^* = 10$, and approximately 100% overprediction of $\langle \delta_r \tilde{\phi}_L^2 \rangle$ at $\Delta^* = 20$. Conversely, the PSM model seems to exhibit a better performance in the prediction of $\langle \delta_r \tilde{\phi}_L^2 \rangle$ compared to the SAG model; however, it has an approximate 20% overprediction of the second-order structure function when $\Delta^* = 10$, and at $\Delta^* = 20$, a prolonged 50% overprediction of $\langle \delta_r \tilde{\phi}_L^n \rangle$ is seen (especially for $100\,\eta_B < r$). Moreover, it is well understood that capturing the complex behavior of the third-order scalar structure function is quite a cumbersome task (Warhaft, 2000). However, by looking at Figs. 13(b) and 13(d), we notice that the DNN model manifests a great performance in the prediction of the third-order longitudinal structure function over almost the entire range of spatial shift at both of the investigated filter sizes. For example, at $\Delta^* = 10$, we observe that over the whole range of spatial shift the DNN model predictions have an approximate $\pm 5\%$ deviation from the FDNS values; nonetheless, we can observe an overprediction

causing inaccuracies over r in the range of 10%–40% for the SAG model. Regarding the PSM model, the inaccurate predictions of $\langle \delta_r \tilde{\phi}_L^3 \rangle$ vary from -15% deviations from the true value for $r < 100\,\eta_B$ to approximately 20% deviation for the $100\,\eta_B < r$ spatial shift region. Therefore, the DNN model plays a notable role in maintaining the accuracy of the third-order statistics of $\delta_r \tilde{\phi}_L$.

This overall comparison demonstrates that a well-trained DNN model such as the one we developed could have consequential improvements in the prediction of two-point statistics of the LES resolved scalar concentration.

5. CONCLUSIONS AND REMARKS

In this work, we developed a data-driven surrogate model for the prediction of the closure flux term appearing in the LES equations of a passive scalar. The developed model was a fully connected deep neural network architecture that takes the gradient of the filtered/resolved scale transport fields as input features. In order to provide a rich set of training/validation data from FDNS, we introduced a spatiotemporal procedure to sufficiently sample from a fully-turbulent time domain and also the homogeneous spatial turbulent domain at each sampled time frame. Using the dataset, we managed to optimize the model parameters of our specified DNN model with respect to the MSE loss function. Eventually, we presented a proper optimized DNN model where training and validation loss records have shown perfectly matching behavior; therefore, any overfitting was avoided. Using the notion of transfer learning, we efficiently (in terms of data requirement and computational cost) generalized our initially trained model into a case for a higher Schmidt number, and another case for a higher Péclet number. In training the TL models, we ensured that we are avoiding an overfitting after the retraining procedure. Afterwards, we managed to test the performance of the developed DNN models in the inference mode. Therefore, in an *a priori* assessment, we compared the performance of the base DNN model with the PSM and SAG models at two filter scales, and realized that the DNN model not only reduces the level of error in predictions of SGS flux and dissipation, but also improves the consistency in the prediction of SGS flux with respect to the direction of large-scale anisotropy, $\boldsymbol{\nabla}\langle\tilde{\Phi}\rangle$, compared to the traditional SGS models. For the TL models, we tested the pretrained model's performance in the prediction of a new scalar transport regime against the retrained models and found out that retraining significantly enhances the overprediction of the forward scattering phenomenon (and even the backward scattering of filtered variance at the higher Péclet number regime). Finally, along the SAG and PSM models, we tested our base DNN model in an LES setting. We compared the records of resolved-scale scalar variance from the employed SGS model as well as the second-order and third-order structure functions of the resolved scalar. Compared to the traditional models we tested, we realized that a well-trained DNN model can result in striking error reduction in the recorded scalar variance as well as significant improvement in the prediction of the two-point structure functions.

As a future direction for this study, it is interesting to evaluate the possibility of applying TL to a variety of different tasks and applications in turbulence modeling for LES. For instance, one can adopt the developed base surrogate modeling strategy we presented in this work, and assess if that can be generalized to predict the SGS scalar flux when the turbulent flow medium is a shear flow (e.g., a jet or a wall-bounded flow), turbulent flow around a complex geometry, or an unsteady turbulent flow instead of the HIT flow medium. It is imperative to mention that developing the base model for those types of flows may require additional considerations such as including different or more input features as well as experimentation with different neural

network architectures that might be more suitable for accounting for a different nature of the SGS closure terms in those flows [see, e.g., Radhakrishnan et al. (2006), Bertsch and Girimaji (2015), Momen and Bou-Zeid (2017), and Lav et al. (2019) for challenges in modeling unsteady flows]. However, due to the extra complexities arising from such generalization tasks, employing a proper approach in *meta learning* [see Hospedales et al. (2021) for a comprehensive review] might be a complementary or a superior methodology to the presented TL approach in terms of both implementation and performance.

ACKNOWLEDGMENTS

This research was funded by the ARO Young Investigator Program (YIP) award (W911NF-19-1-0444) and the NSF award (DMS-1923201).

REFERENCES

Akhavan-Safaei, A. and Zayernouri, M., A Parallel Computational–Statistical Framework for Simulation of Turbulence: Applications to Data-Driven Fractional Modeling, *Fractal Fractional*, vol. **7**, no. 6, p. 488, 2023a.

Akhavan-Safaei, A. and Zayernouri, M., A Non-Local Spectral Transfer Model and New Scaling Law for Scalar Turbulence, *J. Fluid Mech.*, vol. **956**, p. A26, 2023b.

Antonopoulos-Domis, M., Large-Eddy Simulation of a Passive Scalar in Isotropic Turbulence, *J. Fluid Mech.*, vol. **104**, pp. 55–79, 1981.

Arzani, A., Wang, J.X., and D'Souza, R.M., Uncovering Near-Wall Blood Flow from Sparse Data with Physics-Informed Neural Networks, *Phys. Fluids*, vol. **33**, no. 7, p. 071905, 2021.

Bae, H.J. and Koumoutsakos, P., Scientific Multi-Agent Reinforcement Learning for Wall-Models of Turbulent Flows, *Nat. Commun.*, vol. **13**, p. 1443, 2022.

Beck, A., Flad, D., and Munz, C.D., Deep Neural Networks for Data-Driven LES Closure Models, *J. Comput. Phys.*, vol. **398**, p. 108910, 2019.

Beck, A. and Kurz, M., A Perspective on Machine Learning Methods in Turbulence Modeling, *Ges. Angew. Math. Mitt.*, vol. **44**, p. e202100002, 2021.

Bedford, K. and Yeo, W., Conjunctive Filtering Procedures in Surface Water Flow and Transport, *Large Eddy Simulation of Complex Engineering and Geophysical Flows*, B. Galperin and S.A. Orszag, Eds., Cambridge, UK: Cambridge University Press, 1993.

Bergen, K.J., Johnson, P.A., de Hoop, M.V., and Beroza, G.C., Machine Learning for Data-Driven Discovery in Solid Earth Geoscience, *Science*, vol. **363**, p. eaau0323, 2019.

Bertsch, R.L. and Girimaji, S.S., Rapid Distortion Analysis of High Speed Homogeneous Turbulence Subject to Periodic Shear, *Phys. Fluids*, vol. **27**, p. 126104, 2015.

Brenowitz, N.D. and Bretherton, C.S., Prognostic Validation of a Neural Network Unified Physics Parameterization, *Geophys. Res. Lett.*, vol. **45**, no. 12, pp. 6289–6298, 2018.

Brenowitz, N.D. and Bretherton, C.S., Spatially Extended Tests of a Neural Network Parametrization Trained by Coarse-Graining, *J. Adv. Model. Earth Syst.*, vol. **11**, no. 8, pp. 2728–2744, 2019.

Brunton, S.L., Proctor, J.L., and Kutz, J.N., Discovering Governing Equations from Data by Sparse Identification of Nonlinear Dynamical Systems, *Proc. Natl. Acad. Sci. U.S.A.*, vol. **113**, no. 15, pp. 3932–3937, 2016.

Brunton, S.L., Noack, B.R., and Koumoutsakos, P., Machine Learning for Fluid Mechanics, *Annu. Rev. Fluid Mech.*, vol. **52**, pp. 477–508, 2020.

Charalampopoulos, A.T.G. and Sapsis, T.P., Machine-Learning Energy-Preserving Nonlocal Closures for Turbulent Fluid Flows and Inertial Tracers, *Phys. Rev. Fluids*, vol. **7**, no. 2, p. 024305, 2022.

Clark, R.A., Ferziger, J.H., and Reynolds, W.C., Evaluation of Subgrid-Scale Models Using an Accurately Simulated Turbulent Flow, *J. Fluid Mech.*, vol. **91**, no. 1, pp. 1–16, 1979.

Drygala, C., Winhart, B., di Mare, F., and Gottschalk, H., Generative Modeling of Turbulence, *Phys. Fluids*, vol. **34**, no. 3, p. 035114, 2022.

Duraisamy, K., Perspectives on Machine Learning-Augmented Reynolds-Averaged and Large Eddy Simulation Models of Turbulence, *Phys. Rev. Fluids*, vol. **6**, no. 5, p. 050504, 2021.

Duraisamy, K., Iaccarino, G., and Xiao, H., Turbulence Modeling in the Age of Data, *Annu. Rev. Fluid Mech.*, vol. **51**, pp. 357–377, 2019.

Frezat, H., Balarac, G., Le Sommer, J., Fablet, R., and Lguensat, R., Physical Invariance in Neural Networks for Subgrid-Scale Scalar Flux Modeling, *Phys. Rev. Fluids*, vol. **6**, no. 2, p. 024607, 2021.

Fukami, K., Fukagata, K., and Taira, K., Super-Resolution Reconstruction of Turbulent Flows with Machine Learning, *J. Fluid Mech.*, vol. **870**, pp. 106–120, 2019.

Gamahara, M. and Hattori, Y., Searching for Turbulence Models by Artificial Neural Network, *Phys. Rev. Fluids*, vol. **2**, no. 5, p. 054604, 2017.

Guan, Y., Chattopadhyay, A., Subel, A., and Hassanzadeh, P., Stable *A Posteriori* LES of 2D Turbulence Using Convolutional Neural Networks: Backscattering Analysis and Generalization to Higher Re via Transfer Learning, *J. Comput. Phys.*, vol. **458**, p. 111090, 2022.

Han, J., Jentzen, A., and E, W., Solving High-Dimensional Partial Differential Equations Using Deep Learning, *Proc. Natl. Acad. Sci. U.S.A.*, vol. **115**, no. 34, pp. 8505–8510, 2018.

Iyer, K. and Yeung, P., Structure Functions and Applicability of Yaglom's Relation in Passive-Scalar Turbulent Mixing at Low Schmidt Numbers with Uniform Mean Gradient, *Phys. Fluids*, vol. **26**, p. 085107, 2014.

Kim, J., Kim, H., Kim, J., and Lee, C., Deep Reinforcement Learning for Large-Eddy Simulation Modeling in Wall-Bounded Turbulence, arXiv:2201.09505, 2022.

Kumar, S., Tan, S., Zheng, L., and Kochmann, D.M., Inverse-Designed Spinodoid Metamaterials, *NPJ Comput. Mater.*, vol. **6**, no. 1, p. 73, 2020.

Kutz, J.N., Brunton, S.L., Brunton, B.W., and Proctor, J.L., *Dynamic Mode Decomposition: Data-Driven Modeling of Complex Systems*, Philadelphia: SIAM, 2016.

Kutz, J.N., Deep Learning in Fluid Dynamics, *J. Fluid Mech.*, vol. **814**, pp. 1–4, 2017.

Lav, C., Sandberg, R.D., and Philip, J., A Framework to Develop Data-Driven Turbulence Models for Flows with Organised Unsteadiness, *J. Comput. Phys.*, vol. **383**, pp. 148–165, 2019.

Leonard, A., Energy Cascade in Large-Eddy Simulations of Turbulent Fluid Flows, *Adv. Geophys.*, vol. **18**, pp. 237–248, 1975.

Li, J.P., Tang, D.G., Yi, C., and Yan, C., Data-Augmented Turbulence Modeling by Reconstructing Reynolds Stress Discrepancies for Adverse-Pressure-Gradient Flows, *Phys. Fluids*, vol. **34**, no. 4, p. 045110, 2022.

Li, M. and McComb, C., Using Physics-Informed Generative Adversarial Networks to Perform Super-Resolution for Multiphase Fluid Simulations, *J. Comput. Inf. Sci. Eng.*, vol. **22**, no. 4, p. 044501, 2022.

Lilly, D.K., The Representation of Small-Scale Turbulence in Numerical Simulation Experiments, in *Proc. of the IBM Computer Science Symposium on Environmental Sciences*, Yorktown Heights, NY, 1967.

Ling, J., Kurzawski, A., and Templeton, J., Reynolds Averaged Turbulence Modelling Using Deep Neural Networks with Embedded Invariance, *J. Fluid Mech.*, vol. **807**, pp. 155–166, 2016.

Lischke, A., Zayernouri, M., and Zhang, Z., Spectral and Spectral Element Methods for Fractional Advection–Diffusion–Reaction Equations, *Handbook of Fractional Calculus with Applications: Numerical Methods*, vol. **3**, pp. 157–183, 2019.

Liu, B., Yu, H., Huang, H., Liu, N., and Lu, X., Investigation of Nonlocal Data-Driven Methods for Subgrid-Scale Stress Modeling in Large Eddy Simulation, *AIP Adv.*, vol. **12**, no. 6, p. 065129, 2022.

Lu, H. and Porté-Agel, F., A Modulated Gradient Model for Scalar Transport in Large-Eddy Simulation of the Atmospheric Boundary Layer, *Phys. Fluids*, vol. **25**, no. 1, p. 015110, 2013.

MacArt, J.F., Sirignano, J., and Freund, J.B., Embedded Training of Neural-Network Subgrid-Scale Turbulence Models, *Phys. Rev. Fluids*, vol. **6**, no. 5, p. 050502, 2021.

Meneveau, C., Statistics of Turbulence Subgrid-Scale Stresses: Necessary Conditions and Experimental Tests, *Phys. Fluids*, vol. **6**, no. 2, pp. 815–833, 1994.

Meneveau, C. and Katz, J., Scale-Invariance and Turbulence Models for Large-Eddy Simulation, *Annu. Rev. Fluid Mech.*, vol. **32**, no. 1, pp. 1–32, 2000.

Momen, M. and Bou-Zeid, E., Mean and Turbulence Dynamics in Unsteady Ekman Boundary Layers, *J. Fluid Mech.*, vol. **816**, pp. 209–242, 2017.

Overholt, M. and Pope, S., Direct Numerical Simulation of a Passive Scalar with Imposed Mean Gradient in Isotropic Turbulence, *Phys. Fluids*, vol. **8**, no. 11, pp. 3128–3148, 1996.

Park, J. and Choi, H., Toward Neural-Network-Based Large Eddy Simulation: Application to Turbulent Channel Flow, *J. Fluid Mech.*, vol. **914**, p. A16, 2021.

Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., and Lerer, A., Automatic Differentiation in PyTorch, in *31st Conf. on Neural Information Processing Systems (NIPS 2017)*, Long Beach, CA, 2017.

Peng, W., Yuan, Z., and Wang, J., Attention-Enhanced Neural Network Models for Turbulence Simulation, *Phys. Fluids*, vol. **34**, no. 2, p. 025111, 2022.

Pope, S.B., *Turbulent Flows*, Cambridge, UK: Cambridge University Press, 2001.

Portwood, G.D., Nadiga, B.T., Saenz, J.A., and Livescu, D., Interpreting Neural Network Models of Residual Scalar Flux, *J. Fluid Mech.*, vol. **907**, 2021.

Raccuglia, P., Elbert, K.C., Adler, P.D., Falk, C., Wenny, M.B., Mollo, A., Zeller, M., Friedler, S.A., Schrier, J., and Norquist, A.J., Machine-Learning-Assisted Materials Discovery Using Failed Experiments, *Nature*, vol. **533**, no. 7601, pp. 73–76, 2016.

Radhakrishnan, S., Piomelli, U., Keating, A., and Lopes, A.S., Reynolds-Averaged and Large-Eddy Simulations of Turbulent Non-Equilibrium Flows, *J. Turbul.*, no. 7, p. N63, 2006.

Raissi, M., Perdikaris, P., and Karniadakis, G.E., Physics-Informed Neural Networks: A Deep Learning Framework for Solving Forward and Inverse Problems Involving Nonlinear Partial Differential Equations, *J. Comput. Phys.*, vol. **378**, pp. 686–707, 2019.

Rudy, S.H., Brunton, S.L., Proctor, J.L., and Kutz, J.N., Data-Driven Discovery of Partial Differential Equations, *Sci. Adv.*, vol. **3**, no. 4, p. e1602614, 2017.

Sagaut, P., *Large Eddy Simulation for Incompressible Flows: An Introduction*, Berlin: Springer Science & Business Media, 2006.

Samiee, M., Akhavan-Safaei, A., and Zayernouri, M., A Fractional Subgrid-Scale Model for Turbulent Flows: Theoretical Formulation and *A Priori* Study, *Phys. Fluids*, vol. **32**, no. 5, 2020.

Samiee, M., Akhavan-Safaei, A., and Zayernouri, M., Tempered Fractional LES Modeling, *J. Fluid Mech.*, vol. **932**, p. A4, 2022.

Sarghini, F., De Felice, G., and Santini, S., Neural Networks Based Subgrid Scale Modeling in Large Eddy Simulations, *Comput. Fluids*, vol. **32**, no. 1, pp. 97–108, 2003.

Seyedi, S.H. and Zayernouri, M., A Data-Driven Dynamic Nonlocal Subgrid-Scale Model for Turbulent Flows, *Phys. Fluids*, vol. **34**, no. 3, 2022.

Sirignano, J., MacArt, J.F., and Freund, J.B., DPM: A Deep Learning PDE Augmentation Method with Application to Large-Eddy Simulation, *J. Comput. Phys.*, vol. **423**, p. 109811, 2020.

Sirignano, J. and Spiliopoulos, K., DGM: A Deep Learning Algorithm for Solving Partial Differential Equations, *J. Comput. Phys.*, vol. **375**, pp. 1339–1364, 2018.

Subel, A., Chattopadhyay, A., Guan, Y., and Hassanzadeh, P., Data-Driven Subgrid-Scale Modeling of Forced Burgers Turbulence Using Deep Learning with Generalization to Higher Reynolds Numbers via Transfer Learning, *Phys. Fluids*, vol. **33**, no. 3, p. 031702, 2021.

Suzuki, J.L., Gulian, M., Zayernouri, M., and D'Elia, M., Fractional Modeling in Action: A Survey of Nonlocal Models for Subsurface Transport, Turbulent Flows, and Anomalous Materials, *J. Peridynam. Nonlocal Model.*, vol. **5**, no. 3, pp. 392–459, 2023.

Suzuki, J.L., Tuttle, T.G., Roccabianca, S., and Zayernouri, M., A Data-Driven Memory-Dependent Modeling Framework for Anomalous Rheology: Application to Urinary Bladder Tissue, *Fractal Fractional*, vol. **5**, no. 4, p. 223, 2021.

Vollant, A., Balarac, G., and Corre, C., A Dynamic Regularized Gradient Model of the Subgrid-Scale Stress Tensor for Large-Eddy Simulation, *Phys. Fluids*, vol. **28**, no. 2, p. 025114, 2016.

Vollant, A., Balarac, G., and Corre, C., Subgrid-Scale Scalar Flux Modelling Based on Optimal Estimation Theory and Machine-Learning Procedures, *J. Turbul.*, vol. **18**, no. 9, pp. 854–878, 2017.

Vreman, B., Geurts, B., and Kuerten, H., Large-Eddy Simulation of the Turbulent Mixing Layer, *J. Fluid Mech.*, vol. **339**, pp. 357–390, 1997.

Wang, J.X., Wu, J.L., and Xiao, H., Physics-Informed Machine Learning Approach for Reconstructing Reynolds Stress Modeling Discrepancies Based on DNS Data, *Phys. Rev. Fluids*, vol. **2**, no. 3, p. 034603, 2017.

Wang, Y., Yuan, Z., Xie, C., and Wang, J., Artificial Neural Network-Based Spatial Gradient Models for Large-Eddy Simulation of Turbulence, *AIP Adv.*, vol. **11**, no. 5, p. 055216, 2021.

Warhaft, Z., Passive Scalars in Turbulent Flows, *Annu. Rev. Fluid Mech.*, vol. **32**, no. 1, pp. 203–240, 2000.

Williams, J., Wolfram, U., and Ozel, A., Neural Stochastic Differential Equations for Particle Dispersion in Large-Eddy Simulations of Homogeneous Isotropic Turbulence, *Phys. Fluids*, vol. **34**, Article ID 113315, 2022.

Wu, J.L., Xiao, H., and Paterson, E., Physics-Informed Machine Learning Approach for Augmenting Turbulence Models: A Comprehensive Framework, *Phys. Rev. Fluids*, vol. **3**, no. 7, p. 074602, 2018.

Wu, J.L., Kashinath, K., Albert, A., Chirila, D., Xiao, H., et al., Enforcing Statistical Constraints in Generative Adversarial Networks for Modeling Chaotic Dynamical Systems, *J. Comput. Phys.*, vol. **406**, p. 109209, 2020.

Xie, C., Wang, J., and Weinan, E., Modeling Subgrid-Scale Forces by Spatial Artificial Neural Networks in Large Eddy Simulation of Turbulence, *Phys. Rev. Fluids*, vol. **5**, no. 5, p. 054606, 2020.

Yang, X., Zafar, S., Wang, J.X., and Xiao, H., Predictive Large-Eddy-Simulation Wall Modeling via Physics-Informed Neural Networks, *Phys. Rev. Fluids*, vol. **4**, no. 3, p. 034602, 2019.

Yosinski, J., Clune, J., Bengio, Y., and Lipson, H., How Transferable Are Features in Deep Neural Networks?, in *Proc. of 27th Int. Conf. on Advances in Neural Information Processing Systems*, vol. **2**, pp. 3320–3328, 2014.

Yuan, Z., Xie, C., and Wang, J., Deconvolutional Artificial Neural Network Models for Large Eddy Simulation of Turbulence, *Phys. Fluids*, vol. **32**, no. 11, p. 115106, 2020.

Zangeneh, R., Data-Driven Model for Improving Wall-Modeled Large-Eddy Simulation of Supersonic Turbulent Flows with Separation, *Phys. Fluids*, vol. **33**, no. 12, p. 126103, 2021.