Check for updates

Deep learning emulation of subgrid scale processes in turbulent shear flows.

Anikesh Pal[1,2]

[1]National Center for Computational Sciences, Oak Ridge National Laboratory, TN USA
[2]Department of Mechanical Engineering, Indian Institute of Technology Kanpur, U.P. INDIA

Author Manuscript

Key Points:

- DNNs are built to emulate the SGS eddy viscosity and diffusivity for turbulent stratified shear flows.
- These DNNs compute the SGS eddy viscosity and diffusivity $2-4$ times faster than the Dynamic Smagorinsky Model.
- These DNNs emulate the SGS processes accurately such that the energy and variance budgets match with the Dynamic Smagorinsky Model.

This is the author manuscript accepted for publication and has undergone full peer review but has not been through the copyediting, typesetting, pagination and proofreading process, which may lead to differences between this version and the Version of Record. Please cite this article as doi: 10.1029/2020GL087005

Corresponding author: Anikesh Pal, pala@iitk.ac.in

This article is protected by copyright. All rights reserved.

Abstract

Deep neural networks (DNNs) are developed from a dataset obtained from the Dynamic Smagorinsky model to emulate the subgrid-scale viscosity (ν_{sgs}) and diffusivity (κ_{sgs}) for turbulent stratified shear flows encountered in oceans and atmosphere. These DNNs predict ν_{sgs} and κ_{sgs} from velocities, strain rates and density gradients such that the evolution of the kinetic energy budget and density variance budget terms is similar to the corresponding values obtained from the original Dynamic Smagorinsky model. These DNNs also compute ν_{sgs} and κ_{sgs} $\sim 2-4$ times quicker than the Dynamic Smagorinsky model resulting in a $\sim 2-2.5$ times acceleration of the entire simulation. This study demonstrates the feasibility of deep learning in emulating the subgrid scale phenomenon in geophysical flows accurately in a cost-effective manner. In a broader perspective, deep learning based surrogate models can present a promising alternative to the traditional parameterizations of the subgrid-scale processes in climate models.

Plain Language Summary Large eddy simulations (LES) are commonly used to simulate various atmospheric and oceanic flows. In LES the large eddies are resolved whereas the small scale turbulent features, which are the primary source of mixing, are parameterized using physical models. A deep learning based surrogate LES model is developed from the dataset obtained from such a physical model, the Dynamic Smagorinsky model, at moderate Reynolds number (see SI) and resolution. When this surrogate LES model is deployed for 10 times higher Reynolds number at a relatively higher and lower resolution, it was able to capture all the qualitative and quantitative features of the flow accurately at a cheaper computational cost. The effectiveness of deep learning based surrogate models to emulate the small-scale processes is a promising area of research and can potentially be extended for various subgrid scale parameterizations in climate and earth science models.

1 Introduction

The fidelity of climate models and earth system models is compromised due to their inability to resolve the turbulent mixing occurring at the scale of hundreds of meters to a few kilometers. Stratified oceanic and atmospheric mixing is commonly instigated by a shear instability which is the dominant mechanism for converting the fluid motion to turbulence. This investigation uses deep learning as a new technique to accurately parameterize the turbulence phenomenon associated with mixing owing to shear instabilities at high Reynolds numbers in a stratified environment.

In recent years, deep learning (Dechter, 1986; LeCun, Bengio, & Hinton, 2015) has evolved as a compelling and cutting-edge topic of research and has demonstrated a tremendous increase in accuracy in the areas of image and speech recognition. Deep learning achieves its great power and flexibility by learning features incrementally through its hidden layer architecture. Owing to its supremacy in terms of accuracy when trained with a huge amount of data, deep neural networks are gaining popularity in the areas of turbulence modeling (Ling, Kurzawski, & Templeton, 2016; Parish & Duraisamy, 2016; Tracey, Duraisamy, & Alonso, 2013; Zhang & Duraisamy, 2015), climate and earth science research (Anderson & Lucas, 2018; Bolton & Zanna, 2019; Brenowitz & Bretherton, 2018; Gentine, Pritchard, Rasp, Reinaudi, & Yacalis, 2018; Pal, Mahajan, & Norman, 2019; Reichstein et al., 2019; Watson, 2019).

A stratified shear layer, characterized by two parallel fluid streams with different velocities and density (see figure S1 in SI), is commonly observed in oceans and atmosphere. Under favorable conditions such stratified shear layer develops Kelvin-Helmholtz (KH) instabilities and forms large-scale billows which subsequently break down to turbulence and mix the fluid streams. The evolution of a stratified shear layer is extensively studied by using observational techniques, laboratory experiments and numerical simulations (Brucker & Sarkar, 2007; Caulfield & Peltier, 2000; Geyer, Lavery, Scully, & Trowbridge, 2010; Pham,

This article is protected by copyright. All rights reserved.

Author Manuscript

19448007, 2020, 12, Downloaded from https://agupubs.onlinelibrary.wiley.com/doi/10.1029/2020GL087005 by Oak Ridge National Laboratory, Wiley Online Library on [27/03/2023]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

Author Manuscript

19448007, 2020, 12, Downloaded from https://agupubs.onlinelibrary.wiley.com/doi/10.1029/2020GL087005 by Oak Ridge National Laboratory, Wiley Online Library on [27/03/2023]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

Sarkar, & Brucker, 2009; W. D. Smyth & Moum, 2000a, 2000b). Turbulence resolving simulations (direct numerical simulations, DNS) of stratified shear layers have been performed at Reynolds number of 5000 (Pham & Sarkar, 2010; W. Smyth, Moum, & Caldwell, 2001). However, in a more realistic scenario such as in the oceans and atmosphere, the Reynolds numbers associated with these shear layers are $O(10^5 - 10^6)$ (Geyer et al., 2010). DNS at such high Reynolds number are prohibitively expensive. Therefore, large eddy simulations (LES) are used as a cheaper alternative. In LES the larger eddies are resolved whereas the effect of the small scales on these large eddies are parametrized by the subgrid scale fluxes. LES are extensively used in investigating atmospheric boundary layers (Chamecki, Meneveau, & Parlange, 2007; Moeng & Sullivan, 1994), shallow cumulus convection (Siebesma et al., 2003) and stratocumulus-topped planetary boundary layer (Stevens et al., 2005). Similarly LES are also noticeably employed in oceanography (Jalali & Sarkar, 2017; Pham & Sarkar, 2017) and ocean modeling (Chalamalla, Santilli, Scotti, Jalali, & Sarkar, 2017). LES of the canonical shear without stratification has also been performed by Vreman, Geurts, and Kuerten (1997).

Although LES models provide a reasonably accurate representation of the small scale processes, their implementation in climate models to parameterize turbulence is very challenging. Therefore, a deep learning based simpler surrogate model is constructed from a robust LES model (Dynamic Smagorinsky model, Germano, Piomelli, Moin, and Cabot (1991)) as a proof of concept to emulate the unresolved turbulent processes accurately. The Dynamic Smagorinsky model computes ν_{sgs} and κ_{sgs} from velocities, strain rates and density gradients. The deep learning based subgrid scale model will also use velocities, strain rates and density gradients to predict ν_{sgs} and κ_{sgs} in a relatively simpler formulation. Machine learning techniques have become popular in recent years for emulating unresolved atmospheric processes. Deep neural networks are used by Gentine et al. (2018) to emulate the effects of unresolved clouds and convection in an idealized simulation over an aqua-planet using the Super-Parameterized Community Atmosphere Model (SPCAM). Rasp, Pritchard, and Gentine (2018) also used deep neural networks to represent all atmospheric subgrid processes in SPCAM. They have reported close resemblance of mean climate and key aspects of variability between the deep learning based parameterization and the traditional parameterization in SPCAM. Similar neural network based parameterization for apparent sources of heat and moisture are developed by Brenowitz and Bretherton (2018) for a near global aqua-planet simulation using the Community Atmosphere Model. A random forest based parameterization for moist convection is also developed and implemented in a general circulation model (GCM) in an idealized setting by O'Gorman and Dwyer (2018). Deep neural network based surrogate models are also developed by Pal et al. (2019) for radiative transfer in Super-Parameterized Energy Exascale Earth System Model (SP-E3SM). Bolton and Zanna (2019) trained convolutional neural networks (CNNs) on degraded data from a high-resolution quasi-geostrophic ocean model. They demonstrated that their CNNs successfully replicate the spatiotemporal variability of the subgrid eddy momentum forcing, are capable of generalizing to a range of dynamical behaviors, and can be forced to respect global momentum conservation.

In the past few years, neural network based models have emerged as a substitute to parameterize the turbulence processes in Reynolds Averaged Navier-Stokes (RANS) models (Ling et al., 2016; Parish & Duraisamy, 2016; Tracey et al., 2013; Zhang & Duraisamy, 2015). Recently, Srinivasan, Guastoni, Azizpour, Schlatter, and Vinuesa (2019) illustrated the potential of neural networks to predict the temporal dynamics of a low-order model of a turbulent flow. However, the use of deep neural networks to parameterize the turbulent processes in LES models are not explored yet. Therefore, the prime motivation of this investigation is to verify the feasibility of deep learning based subgrid scale models to accurately and efficiently emulate the subgrid scale processes occurring in geophysical flows. Deep neural networks are developed from a dataset obtained from the LES of an evolving shear layer at moderate Reynolds number (10^4) using the Dynamic Smagorinsky model. Later these deep neural networks replace the Dynamic Smagorinsky model to emulate the subgrid scale turbulent processes at a higher Reynolds number (see definition in SI) (10^5) relevant to ocean and atmosphere at different resolutions. The mean and fluctuating statistics obtained from

This article is protected by copyright. All rights reserved.

this deep learning based subgrid scale model are compared with the traditional Dynamic Smagorinsky model to assess the accuracy and effectiveness of this surrogate LES model.

The details of the problem setup and DNN architecture are presented in section 2. Qualitative and quantitative comparisons between the deep learning based surrogate LES model and the traditional Dynamic Smagorinsky model are discussed in section 3, and conclusions are given in section 4.

2 Method

2.1 Problem setup

2.1.1 Stratified shear layer

Following Pham and Sarkar (2014) a stratified shear layer is defined as two parallel streams of fluid with different velocity and density flowing in opposite directions (see figure S1 in SI). The streamwise velocity and density fields are initialized as a function of the vertical direction (x_3) as follows:

$$\langle u^* \rangle = \frac{\Delta U^*}{2} tanh\left(\frac{x_3^*}{0.5\delta_{\omega,0}^*}\right), \tag{1}$$

$$\langle \rho^* \rangle = \rho_0^* + \frac{\Delta \rho^*}{2} tanh\left(\frac{x_3^*}{0.5\delta_{\omega,0}^*}\right), \tag{2}$$

where ΔU^* and $\Delta \rho^*$ are the initial velocity and density differences, respectively. The initial thickness of the shear layer is $\delta_{\omega,0}^*$ calculated by $\Delta U^*/max(|\frac{du^*}{dx_3}|)$. The superscript $*$ denotes dimensional quantities and $\langle \bullet \rangle$ represents averaging in the homogeneous directions $(x_1 - x_2)$.

2.1.2 Governing equations and numerical method

In a LES model, the equations of motion for an incompressible flow with Boussinesq approximation are filtered in space, non-dimensionalized by ΔU^*, $\delta_{\omega,0}^*$ and $\Delta \rho^*$ Pham and Sarkar (2014), and are written as:

mass:

$$\frac{\partial \overline{u}_i}{\partial x_i} = 0, \tag{3}$$

momentum:

$$\frac{\partial \overline{u}_i}{\partial t} + \frac{\partial (\overline{u_j u_i})}{\partial x_j} = -\frac{\partial \overline{P}}{\partial x_i} + \frac{1}{Re_0}\frac{\partial^2 \overline{u}_i}{\partial x_j \partial x_j} + Ri_{b,0}\overline{\rho}' g_i - \frac{\partial \tau_{ij}}{\partial x_j}, \tag{4}$$

density:

$$\frac{\partial \overline{\rho}}{\partial t} + \frac{\partial (\overline{u_j \rho})}{\partial x_j} = \frac{1}{Re_0 Pr}\frac{\partial^2 \overline{\rho}}{\partial x_j \partial x_j} - \frac{\partial Q_j}{\partial x_j} \tag{5}$$

where the overbar denotes the filtered quantities and g is gravity acting in the vertical(x_3) direction. The Reynolds number, $Re_0 = \Delta U^* \Delta_{\omega,0}^* / \nu^*$ (ν^* is the molecular viscosity), the bulk Richardson number $Ri_{b,0} = g^* \Delta \rho^* \Delta_{\omega,0}^* / \rho_0^* \Delta U^{*2}$, and the Prandtl number $Pr_0 = \nu^*/\kappa^*$ (κ^* is the molecular diffusivity) are the three non-dimensional parameters which prescribe the fluid motion. Although the Prandtl number $Pr_0 = 1$ in this investigation is typical to the atmosphere, the Reynolds number and the bulk Richardson number considered in this study are well within the range expected in both the ocean and atmosphere. The subgrid scale stress, τ_{ij} and subgrid buoyancy flux Q_j is parameterized as follows:

$$\tau_{ij} = -2C_d \overline{\Delta}^2 |S| S_{ij}, \tag{6}$$

$$Q_j = -C_\theta \overline{\Delta}^2 |S| \frac{\partial \overline{\rho}}{\partial x_j}, \tag{7}$$

This article is protected by copyright. All rights reserved.

Author Manuscript

19448007, 2020, 12, Downloaded from https://agupubs.onlinelibrary.wiley.com/doi/10.1029/2020GL087005 by Oak Ridge National Laboratory, Wiley Online Library on [27/03/2023]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

where $\overline{\Delta}$ is the filter width, C_d is the model coefficient, $\overline{S}_{ij} = 1/2(\partial\overline{u}_i/\partial x_j + \partial\overline{u}_j/\partial x_i)$ is the resolved strain rate and $|S|$ is defined as $\sqrt{2\overline{S}_{ij}\overline{S}_{ij}}$. The subgrid eddy-viscosity and diffusivity are given by:

$$\nu_{sgs} = C_d\overline{\Delta}^2|S|, \tag{8}$$

$$\kappa_{sgs} = C_\theta\overline{\Delta}^2|S|, \tag{9}$$

respectively. The model coefficients C_d and C_θ are calculated by a dynamic procedure Germano et al. (1991); Lilly (1992) in which a test filter is applied to the resolved velocity fields. The quantities denoted by $\widetilde{\bullet}$ are double-filtered with both LES and test filters. The dynamic coefficient is computed by:

$$C_d = -\frac{1}{2}\frac{\langle L_{ij}M_{ij}\rangle}{\langle M_{ij}M_{ij}\rangle}, \tag{10}$$

where $L_{ij} = \widetilde{\overline{u}_i\overline{u}_j} - \tilde{\overline{u}}_i\tilde{\overline{u}}_j$ and $M_{ij} = \widetilde{\overline{\Delta}}^2\widetilde{|\overline{S}|}\widetilde{\overline{S}}_{ij} - \overline{\Delta}^2\widetilde{|\overline{S}|\overline{S}_{ij}}$. Similarly, the dynamic coefficient for the subgrid buoyancy flux is

$$C_\theta = -\frac{1}{2}\frac{\langle L_i^\theta M_i^\theta\rangle}{\langle M_i^\theta M_i^\theta\rangle}, \tag{11}$$

where $L_i^\theta = \widetilde{\overline{\rho}\overline{u}_i} - \tilde{\overline{\rho}}\tilde{\overline{u}}_i$ and $M_i = \widetilde{\overline{\Delta}}^2\widetilde{|\overline{S}|}\widetilde{\frac{\partial\overline{\rho}}{\partial x_i}} - \overline{\Delta}^2\widetilde{|\overline{S}|\frac{\partial\overline{\rho}}{\partial x_i}}$. The ratio of the test and LES filter, $\widetilde{\overline{\Delta}}/\overline{\Delta} = 6$, and $\langle\bullet\rangle$ denotes the averaging in the homogeneous directions (in this case the horizontal, $x_1 - x_2$ direction). For time-advancement, a semi-implicit, third order Runge-Kutta/Crank-Nicolson formulation is used. The viscous terms in the wall normal direction are treated implicitly whereas the viscous terms in the periodic direction are treated explicitly. All the spatial derivatives are discretized using a central, second-order finite difference scheme on a staggered grid. The pressure Poisson equation which is utilized to project the velocity field into a divergence-free space is solved using a multigrid solver. This numerical solver has been validated and used extensively for a number of free-shear and wall-bounded turbulence problems Brucker and Sarkar (2010); Pal and Chalamalla (2020); Pal, de Stadler, and Sarkar (2013); Pal and Sarkar (2015); Pham and Sarkar (2014, 2018).

The horizontal boundaries have periodic conditions, while the top and bottom boundaries have the following boundary conditions:

$$\overline{u}_1(x_{3,min}) = \frac{1}{2}, \overline{u}_1(x_{3,max}) = -\frac{1}{2}, \tag{12}$$

$$\overline{u}_2(x_{3,min}) = 0, \overline{u}_2(x_{3,max}) = 0, \tag{13}$$

$$\frac{\partial\overline{u}_3}{\partial x_3}(x_{3,min}) = \frac{\partial\overline{u}_3}{\partial x_3}(x_{3,max}) = 0, \tag{14}$$

$$\overline{p}(x_{3,min}) = \overline{p}(x_{3,max}) = 0, \tag{15}$$

$$\frac{\partial\overline{\rho}}{\partial x_3}(x_{3,min}) = \frac{\partial\overline{\rho}}{\partial x_3}(x_{3,max}) = 0. \tag{16}$$

2.2 Deep neural networks

A dense, fully-connected, feed-forward DNN is chosen for this application as follows:

$$h_1 = \varphi_1\left(W_1\mathcal{A}_{in} + b_1\right) \tag{17}$$

$$h_2 = \varphi_2\left(W_2h_1 + b_2\right) \tag{18}$$

$$h_3 = \varphi_3\left(W_3h_2 + b_3\right) \tag{19}$$

$$P_{out} = \left(W_4h_3 + b_4\right). \tag{20}$$

Here $\mathcal{A}_{in}$ denotes the input vector; P_{out} is the predicted output vector; W_1, W_2, W_3, W_4 are matrices of trainable weights; b_1, b_2, b_3, b_4 are "bias vectors"; φ_1, φ_2, φ_3 are the non-linear

This article is protected by copyright. All rights reserved.

Author Manuscript

19448007, 2020, 12, Downloaded from https://agupubs.onlinelibrary.wiley.com/doi/10.1029/2020GL087005 by Oak Ridge National Laboratory, Wiley Online Library on [27/03/2023]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

activation functions; and h_1, h_2, and h_3 are the "hidden" vectors whose scalar components are called "neurons". Note that activation functions are only applied on the hidden vectors. For the present investigation a neural network package called Keras (https://keras.io) has been used, which is a high-level wrapper around Tensorflow (https:// www.tensorflow.org/) written in Python.

2.3 Data collection and normalization

Table 1 lists all the simulated cases for this study. DSM1 represents simulation at Reynolds number $Re = 10^4$ using the Dynamic Smagorinsky model and serves as a source of data collection for a deep neural network (DNN). The domain is decomposed laterally ($x_1 - x_2$ directions) on 480 CPU cores for computation. The input and output variables are collected at each grid point along the vertical column at the center of each CPU core (see the dashed black line in figure S1 in SI) at every 100 time steps. This ensures spatial and temporal variability in the input-output pairs. The Dynamic Smagorinsky model computes ν_{sgs} from velocities, strain rates and the size of the filters (LES and test) at every grid point as equations 20-10. Similarly κ_{sgs} is calculated from velocities, density, strain rates, density gradients in the respective direction and the size of the filters (LES and test) at every grid point as equations 9-11.

The goal is to develop two separate DNNs, one of which will predict ν_{sgs} whereas the other one will predict κ_{sgs}. The velocities $(\overline{u}, \overline{v}, \overline{w})$ and strain rates $(\overline{S}_{ij} = 1/2(\partial\overline{u}_i/\partial x_j + \partial\overline{u}_j/\partial x_i))$ Germano et al. (1991) at every grid point are taken as inputs to the DNN whereas ν_{sgs} at the corresponding locations will be the output from the DNN. Analogously $\overline{u}, \overline{v}, \overline{w}, \overline{\rho}, \overline{S}_{ij}, \partial\overline{\rho}/\partial x_i$ will be the inputs for computing κ_{sgs}. The size of the filters are related to the grid size and are taken into account during strain rate calculation. Therefore the effect of the size of the filters are indirectly associated with the strain rates and are excluded from the inputs to the DNN. Approximately 12 million input-output samples are collected from DSM1.

The following terminology is used to describe the different data sets:

1. Training data set: A random 90% of the saved data from DSM1 is used to train the DNNs over a series of epochs.
2. Testing data set: A random 10% of the saved data from DSM1 is used to probe the accuracy of the DNNs after they have been trained. The DNNs are never trained on this data set during any epoch.
3. Validation data set: During each epoch, a random 10% of the training data set remains unused during a single epoch and is used to track DNN generalization as epochs progress during training.

The input and output variables should be given equal importance while computing the loss function. Therefore, each input and output variable in the present study is normalized by the maximum and minimum values of each variable across the entire dataset using the following relations:

$$\widehat{\mathcal{A}}_j = \frac{\mathcal{A}_j - \min_i(\mathcal{A}_j)}{\max_i(\mathcal{A}_j) - \min_i(\mathcal{A}_j)}, \tag{21}$$

$$\widehat{\mathcal{C}}_k = \frac{\mathcal{C}_k - \min_i(\mathcal{C}_k)}{\max_i(\mathcal{C}_k) - \min_i(\mathcal{C}_k)}, \tag{22}$$

where $\mathcal{A}_j$ $(j = [1, 9])$ are the inputs $(\overline{u}, \overline{v}, \overline{w}, \overline{S}_{xx}, \overline{S}_{yy}, \overline{S}_{zz}, \overline{S}_{xy}, \overline{S}_{xz}, \overline{S}_{yz})$ and $\mathcal{C}_k$ $(k = 1)$ is the actual output (ν_{sgs}) from DSM1, i is the number of training samples. Similarly, $\mathcal{A}_j$ $(j = [1, 13])$ will be $\overline{u}, \overline{v}, \overline{w}, \overline{\rho}, \overline{S}_{xx}, \overline{S}_{yy}, \overline{S}_{zz}, \overline{S}_{xy}, \overline{S}_{xz}, \overline{S}_{yz}, \partial\overline{\rho}/\partial x, \partial\overline{\rho}/\partial y, \partial\overline{\rho}/\partial z$ for $\mathcal{C}_k$ to be κ_{sgs}. $\widehat{\mathcal{A}}_j$ and $\widehat{\mathcal{C}}_k$ are used as inputs and output for training the DNNs. Notice that outputs $\widehat{\mathcal{C}}_k(\widehat{\nu}_{sgs}, \widehat{\kappa}_{sgs})$ are normalized (21), and are non-normalized using $\min_i(\mathcal{C}_k)$ and $\max_i(\mathcal{C}_k)$

This article is protected by copyright. All rights reserved.

Author Manuscript

19448007, 2020, 12, Downloaded from https://agupubs.onlinelibrary.wiley.com/doi/10.1029/2020GL087005 by Oak Ridge National Laboratory, Wiley Online Library on [27/03/2023]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

Author Manuscript

This article is protected by copyright. All rights reserved.

Case	N_1	N_2	N_3	Re_0	Δx_1^*	Δx_2^*	$\Delta x_{3,min}^*$	$\Delta x_{3,max}^*$	LES model	SGS CPU time(s)/time step	NS Solver CPU time (s)/time step
DSM1(coarse)	640	192	256	10^4	0.04	0.04	0.04	0.6	*DSM*	1.14	0.276
DSM2(coarse)	640	192	256	10^5	0.04	0.04	0.04	0.6	*DSM*	1.14	0.276
DLLES2(coarse)	640	192	256	10^5	0.04	0.04	0.04	0.6	*DNN*	0.42	0.276
DSM3(coarser)	512	160	256	10^5	0.05	0.05	0.05	0.37	*DSM*	1.02	0.228
DLLES3(coarser)	512	160	256	10^5	0.05	0.05	0.05	0.37	*DNN*	0.39	0.228
DSM4(fine)	768	256	512	10^5	0.03	0.03	0.03	0.13	*DSM*	2.94	0.66
DLLES4(fine)	768	256	512	10^5	0.03	0.03	0.03	0.13	*DNN*	0.81	0.66
DSM5(finer)	1024	384	1024	10^5	0.022	0.022	0.022	0.067	*DSM*	7.71	1.96
DLLES5(finer)	1024	384	1024	10^5	0.022	0.022	0.022	0.067	*DNN*	1.67	1.96

Table 1. Simulation parameters for shear layer: N_1, N_2, N_3 are the number of grid points in x_1, x_2 and x_3 directions respectively. $Re_0 = \frac{\Delta U \delta_{\omega,0}}{\nu^*}$ is the Reynolds number, where $\Delta U^* = 1m/s$ is the velocity difference, $\delta_{\omega,0}^* = 1m$ is the initial shear layer thickness and $\nu^*(m^2/s)$ is the molecular viscosity. The Prandtl number is 1 and the bulk Richardson number is 0.1 for all the cases (see SI for details).The data from DSM1 is used to train the deep learning based LES model. The size of the computational domain are $L_1/\delta_{\omega,0}^* = 25.6$, $L_2/\delta_{\omega,0}^* = 7.6$, and $L_3/\delta_{\omega,0}^* = 27$ (see figure S1 in SI) in the streamwise, spanwise and vertical directions respectively. The grid spacings are non-dimesionalized by $\delta_{\omega,0}^*$ and are uniform in streamwise and spanwise directions. In the vertical direction the grid spacing is kept uniform $(\Delta x_{3,min}/\delta_{\omega,0}^*)$ in the region $-3 < x_3/\delta_{\omega,0}^* < 3$. Outside this region the grids are stretched at a ratio of $0.3\% - 3\%$.

This manuscript submitted to *Geophysical Research Letters*

19448007, 2020, 12, Downloaded from https://agupubs.onlinelibrary.wiley.com/doi/10.1029/2020GL087005 by Oak Ridge National Laboratory, Wiley Online Library on [27/03/2023]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

from DSM1 to obtain ν_{sgs} and κ_{sgs}. This is applicable for all the cases with deep learning based subgrid scale model (DLLES2, DLLES3, DLLES4 and DLLES5).

Several exploratory training experiments were carried out to refine the architecture of the DNNs. This includes varying the number of hidden layers, number of neurons, different activation functions, different batch sizes and different number of epochs. However, the current configuration results in a balance between the validation accuracy and speed up. The goal was to select a number of hidden layers and neurons per layer that will minimize the training as well as the validation error. These goals were achieved by using 3 "hidden layers" with 16 neurons in each hidden layer. A ReLU activation unit, which uses the Rectifier Linear (ReL) activation function (https: //keras.io/activations/) defined as $f(\mathbf{x}) = \mathbf{max}(\mathbf{0}, \mathbf{w} \bullet \mathbf{x})$, is chosen for this study. The optimizer used for training is RMSprop (https://keras.io/optimizers) with a learning rate of 0.01. The DNNs are trained with a batch size of 240 for 100 epochs with *mean squared error* (https://keras.io/losses) as the loss function. The details of the setup and architecture of the deep neural network such as hidden layers and number of neurons are provided in section 2.2 and figure S2 in SI. Figures S3(a)-(b) in SI show the evolution of the training and validation loss for ν_{sgs} and κ_{sgs}. The training took approximately 3 hours on a single Nvidia Pascal graphical processing unit (GPU).

Figures S4 (a)-(b) compare the ν_{sgs} and κ_{sgs} predicted from the DNNs with the actual values of ν_{sgs} and κ_{sgs} obtained from the Dynamic Smagorinsky model (DSM1) for the first 100 samples of the testing dataset. These samples represent the spatial and temporal variability of ν_{sgs} and κ_{sgs} at $Re_0 = 10^4$. The DNNs are able to capture the qualitative and quantitative details of the variation in ν_{sgs} and κ_{sgs} similar to the Dynamic Smagorinsky model. The rest of the samples in the testing dataset also manifest qualitative and quantitative similarity in ν_{sgs} and κ_{sgs} calculated from the DNNs and DSM1, however, to avoid data clutter in a single plot only the first 100 samples are shown in figures S4 (a)-(b). These two DNNs will now replace the original Dynamic Smagorinsky model for shear layer simulations at $Re_0 = 10^5$.

3 Results and Discussion

Figure 1 (a) and (c) compares the distribution of ν_{sgs} between DLLES2 and DSM2 at $Re = 10^5$ at time $t = 51s, 85s$ on a vertical $(x_1 - x_3)$ plane at the center $(x_2 = 0)$. An analogous contour plot for DLLES4 and DSM4 is shown in figure 1 (b) and (d) respectively. It is evident from Figures 1(a) and (b) that the deep learning based subgrid scale model is able to capture the essential dynamical features such as the formation of Kelvin-Helmholtz billows and their disintegration to turbulence during the evolution of the shear layer similar to the Dynamic Smagorinsky model (figures 1(c),(d)). The DNNs were trained on a dataset obtained from a coarse resolution simulation at $Re_0 = 10^4$ (case DSM1). When these DNNs are employed at $Re_0 = 10^5$, relevant to atmospheric and oceanic flows, they predict ν_{sgs} and κ_{sgs} fairly well for same and finer resolution simulations. This is manifested by the comparison of the vertical profiles of horizontally averaged (see equation 18 in SI) ν_{sgs} at time $t = 51s, 85s$ between the deep learning based subgrid scale model and the Dynamic Smagorinsky model for the respective cases in figures 1 (e) and (f). Analogous comparisons of κ_{sgs} are shown in figures 1 (g) and (h). The differences in the vertical profiles of ν_{sgs} and κ_{sgs} between the cases with the deep learning based subgrid scale model and the Dynamic Smagorinsky model are probably attributed to the difference in Re_0 and grid resolution at which the DNNs are being trained and employed.

The deep learning based subgrid scale model accelerates the computation of ν_{sgs} and κ_{sgs} by $\sim 2 - 4$ times as compared to the Dynamic Smagorinsky model as evident from the SGS CPU time per time step in table 1. The computation of ν_{sgs} using the deep learning emulation requires ~ 0.85 times floating point operations than the Dynamic Smagorinky model. The deep learning based subgrid scale model replaces all the complex calculations for ν_{sgs} (equations 20-10) and κ_{sgs} (equations 9-11) with matrix-vector multiplications (equations 17-

This article is protected by copyright. All rights reserved.

Author Manuscript

19448007, 2020, 12, Downloaded from https://agupubs.onlinelibrary.wiley.com/doi/10.1029/2020GL087005 by Oak Ridge National Laboratory, Wiley Online Library on [27/03/2023]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

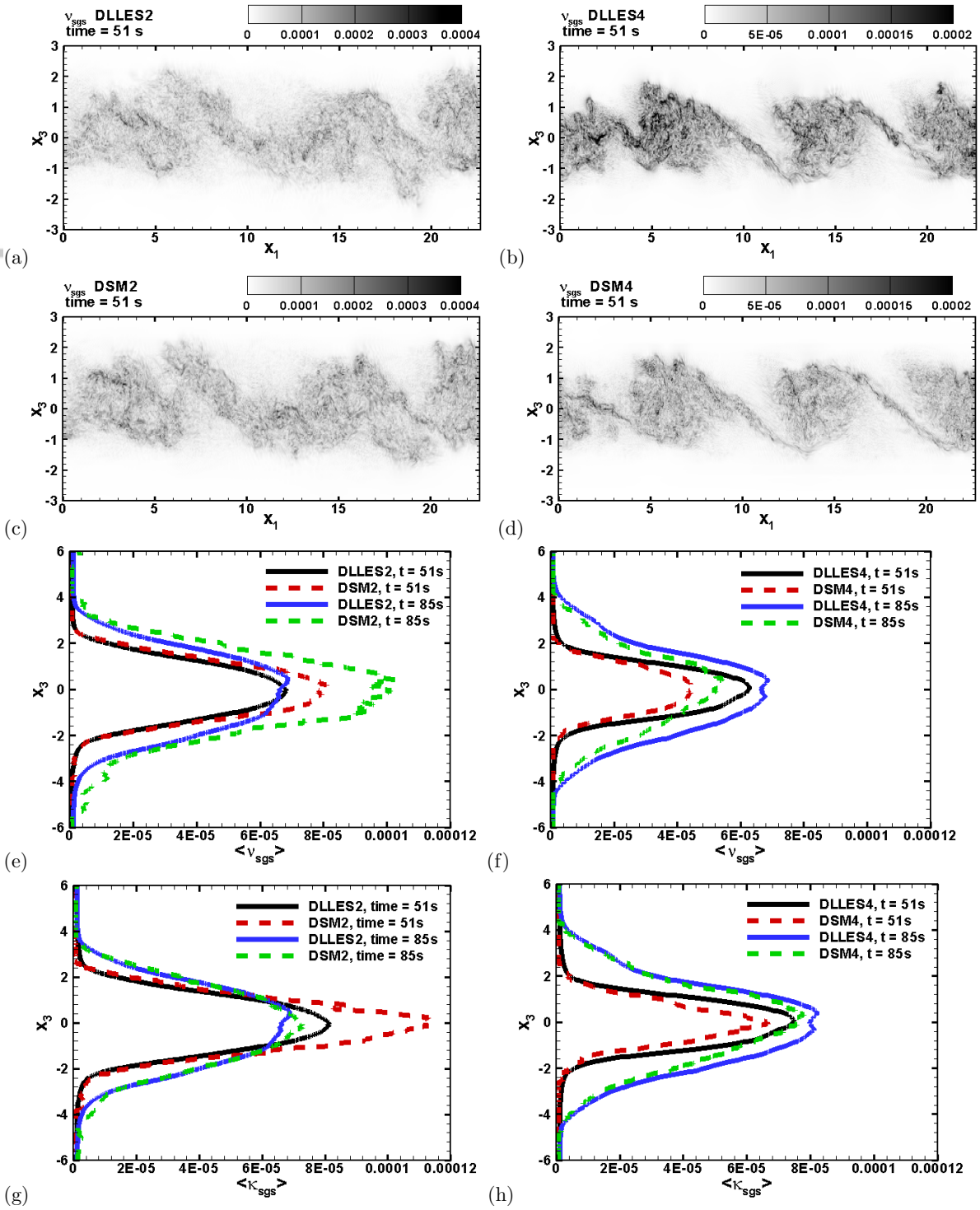


Figure 1. Comparison of ν_{sgs} at central vertical plane ($x_2 = 0, x_1 - x_3$ plane) at coarse and finer resolution obtained from deep learning based LES model (a) DLLES2, (b) DLLES4, and Dynamic Smagorinsky model (c) DSM2, (d) DSM4 at time $t = 51s$. Comparison of vertical profiles of horizontally averaged $\langle \nu_{sgs} \rangle$ for (e) coarse (DLLES2 and DSM2), and (f) finer (DLLES4 and DSM4) resolution simulation at time $t = 51s$ and $Re_0 = 10^5$. Comparison of vertical profiles of horizontally averaged $\langle \kappa_{sgs} \rangle$ for (g) coarse (DLLES2 and DSM2), and (h) finer (DLLES4 and DSM4) resolution simulation time $t = 51s, 85s$ and $Re_0 = 10^5$.

This article is protected by copyright. All rights reserved.

Author Manuscript

19448007, 2020, 12, Downloaded from https://agupubs.onlinelibrary.wiley.com/doi/10.1029/2020GL087005 by Oak Ridge National Laboratory, Wiley Online Library on [27/03/2023]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

Author Manuscript

19448007, 2020, 12, Downloaded from https://agupubs.onlinelibrary.wiley.com/doi/10.1029/2020GL087005 by Oak Ridge National Laboratory, Wiley Online Library on [27/03/2023]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

20). An optimized BLAS operation *dgemm* (http://www.netlib.org/lapack/) is used for these matrix multiplications between the input/hidden vectors and the weight matrices. Therefore the acceleration of the deep learning based subgrid scale model is not only attributed to the lesser number of floating point operations, but also to the use of highly optimized matrix multipliers designed for modern computer architectures. This speed-up is measured by comparing the minimum and maximum CPU time elapsed in calculating ν_{sgs} and κ_{sgs} among all the CPUs during the entire simulation for the deep learning based subgrid scale model and the Dynamic Smagorinsky model. The determination of ν_{sgs} and κ_{sgs}, and the Navier-Stokes solver, which primarily is the Poisson equation (an elliptic equation) for pressure perturbation, are two of the most computationally expensive modules for the simulation of turbulent shear layers. As indicated in table 1 the NS solver CPU time is same for all the respective cases (DSM2 and DLLES2; DSM3 and DLLES3; DSM4 and DLLES4; DSM5 and DLLES5), and is less than the SGS CPU time. Therefore, an acceleration in the computation of ν_{sgs} and κ_{sgs} by the deep learning based subgrid scale model will accelerate the entire simulation by $\sim 2 - 2.5$ times as compared to the simulation using the Dynamic Smagorinsky model.

The DNNs developed in this investigation have learned the underlying physics of the problem resulting in their capability to establish a relationship between the flow variables and eddy viscosity/diffusivity. This is the reason why these DNNs are able to predict ν_{sgs} and κ_{sgs} for a higher Re_0 at different resolutions fairly well. It should be noted that a flow with a different physics will require an additional simulation for data collection, and development of a deep learning based SGS model for that particular flow. However as demonstrated in this study, the deep learning based SGS model can not only provide accuracy but will overall be computationally cheaper (see SI for total computational time details) when used for parametric studies as compared to the Dynamic Smagorinsky model.

A qualitatively similar evolution of the filtered density field ($\overline{\rho}$) is observed between the deep learning based subgrid scale model and the Dynamic Smagorinsky model for the coarse (figures 2(a)-(c)) and fine (figures 2(b)-(d)) grid simulations. An analogous comparison for the coarser grid simulations (DLLES3 and DSM3) is shown in figures S5(a) and (b) in SI. The deep learning based LES model is able to capture the details of the flow field such as the secondary KH instabilities along the braids of the primary KH billows similar to the Dynamic Smagorinsky model. The appearance of such secondary KH instabilities are typical to high Re_0 shear flows (Geyer et al., 2010). It can also be observed from figures 2 (e) and (f) that the deep learning based subgrid scale model imitates the time evolution of the vertically integrated turbulent and mean kinetic energy similar to the Dynamic Smagorinsky model for the coarse (DLLES2, DSM2) and fine (DLLES4, DSM4) grid simulations. The increase in TKE is attributed to the decrease in MKE signifying the conversion of MKE to TKE. This phenomenon is further explained by means of the production term in TKE budget equation in the upcoming paragraphs. Figures S5 (c) and (d) in SI also corroborate the accuracy of the deep learning based subgrid scale model in computing TKE and MKE for coarser (DLLES3) grid simulations.

In LES the large scales are resolved whereas the small scale turbulence is modeled using a subgrid scale model. The performance of a subgrid scale model is determined by its ability to provide sufficient subgrid scale dissipation and scalar dissipation via ν_{sgs} and κ_{sgs} respectively. If the subgrid scale/scalar dissipation is low, energy builds up in small scales resulting in eventual crashing of the solver. If the subgrid scale/scalar dissipation is high the small scale features dissipate resulting in lower turbulence intensities and disappearance of the finer structures in large scales resulting in inaccurate evolution of the large structures. Therefore, to verify the accuracy of the deep learning based subgrid scale model it is imperative to compare the small (TKE, scalar variance budgets) and large scale statistics (MKE budget) obtained from the deep learning algorithm and the Dynamic Sagorinsky model. The TKE (equations $3 - 8$), density variance (equations $13 - 18$), and MKE (equations $9 - 12$) terms are presented in SI.

This article is protected by copyright. All rights reserved.

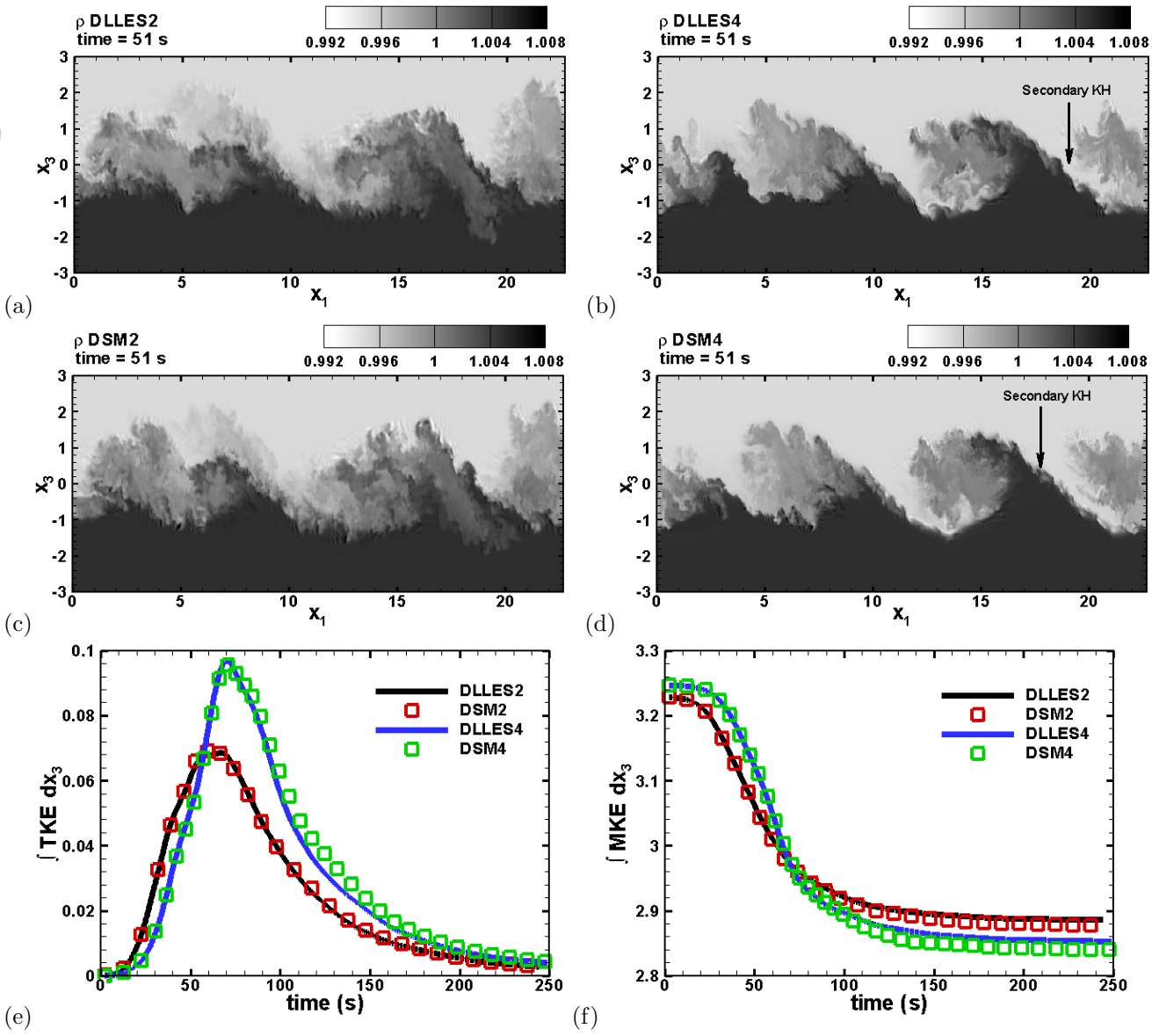


Figure 2. Comparison of filtered density $\overline{\rho}$ field at central vertical plane ($x_2 = 0, x_1 - x_3$ plane) at coarse and fine resolution obtained from deep learning based subgrid scale model (a) DLLES2, (b) DLLES4, and Dynamic Smagorinsky model (c) DSM2, (d) DSM4 at time $t = 51s$. Comparison of the time evolution of vertically integrated (e) TKE, and (f) MKE between the deep learning based subgrid scale model and the Dynamic Smagorinsky model at $Re_0 = 10^5$.

This article is protected by copyright. All rights reserved.

Author Manuscript

19448007, 2020, 12, Downloaded from https://agupubs.onlinelibrary.wiley.com/doi/10.1029/2020GL087005 by Oak Ridge National Laboratory, Wiley Online Library on [27/03/2023]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

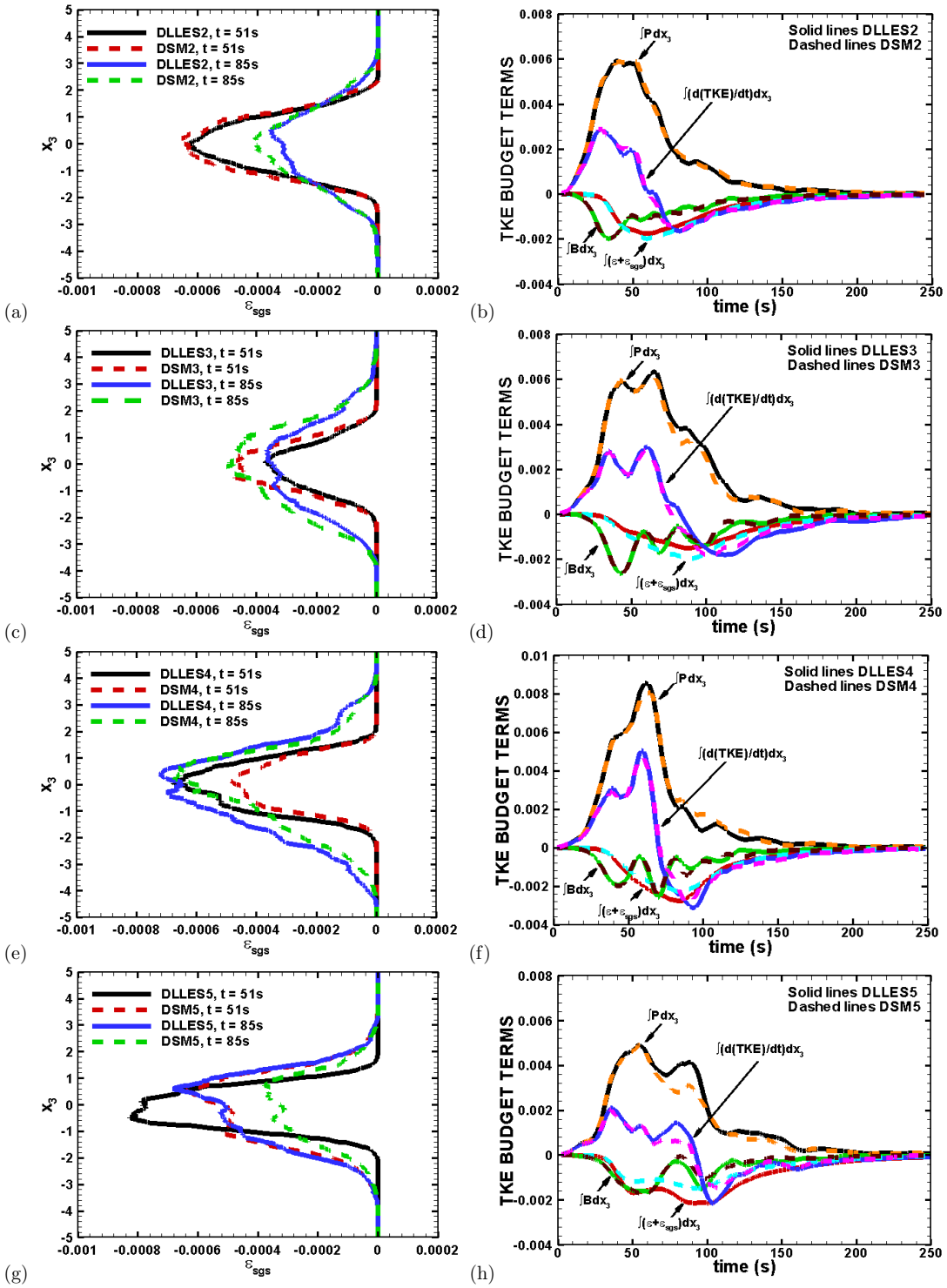


Figure 3. Comparison of vertical profiles ε_{sgs} between (a) DLLES2 and DSM2, (c) DLLES3 and DSM3, (e) DLLES4 and DSM4, and (g) DLLES5 and DSM5 at time $t = 51s, 85s$ and $Re_0 = 10^5$. Comparison of vertically integrated TKE budget terms between (b) DLLES2 and DSM2, (d) DLLES3 and DSM3, (f) DLLES4 and DSM4, and (h) DLLES5 and DSM5 at $Re_0 = 10^5$.

This article is protected by copyright. All rights reserved.

Author Manuscript

19448007, 2020, 12, Downloaded from https://agupubs.onlinelibrary.wiley.com/doi/10.1029/2020GL087005 by Oak Ridge National Laboratory, Wiley Online Library on [27/03/2023]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

Author Manuscript

19448007, 2020, 12, Downloaded from https://agupubs.onlinelibrary.wiley.com/doi/10.1029/2020GL087005 by Oak Ridge National Laboratory, Wiley Online Library on [27/03/2023]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

The vertical profiles of subgrid dissipation (ε_{sgs}, eqn 7 in SI) at time $t = 51s, 85s$ for different grid resolutions are shown in 3 (a), (c) and (e). The resolved dissipation (ε, eqn 5 in SI) is small as compared to the subgrid dissipation for all the cases and is not shown. This is attributed to high Re_0 at which subgrid dissipation contributes more to the TKE budget, making up for the smaller resolved dissipation. Pham and Sarkar (2014) presented LES studies at the $Re_0 = 5000$ in which both the resolved and subgrid scale dissipation contributes equally. The deep learning based LES model is able to determine the subgrid scale dissipation at time $t = 51s, 85$ fairly well for all the grid resolutions when compared with the Dynamic Smagorinksy model. The differences in the vertical profiles of ε_{sgs} is directly related to the differences in the vertical profiles of ν_{sgs}. Geyer et al. (2010) reported that the KH instabilities at $Re_0 = 1.25 \times 10^5$, $Ri_b \sim 0.2 - 0.25$ and $Pr_0 = 700$ has dissipation in the range of $3 - 6.8 \times 10^{-4} m^2 s^{-3}$. Note that Geyer et al. (2010) has defined Reynolds number based on half the velocity difference across the layer and halfwidth of the shear zone. Therefore, their Reynolds number of 5×10^5 equals to $Re_0 = 1.25 \times 10^5$. The dissipation from the deep learning based LES model is in the range $3.6 - 7 \times 10^{-4} m^2 s^{-3}$ (figures 3 (a), (c) and (e)) similar to the observations of Geyer et al. (2010).

A comparison of the time evolution of the vertically integrated TKE budget terms between the deep learning based subgrid scale model and the Dynamic Smagorinsky model for different grid resolutions are shown in figure 3(b), (d) and (f). The resolved and subgrid transport terms when integrated over the computational domain are negligible and are not shown. The initial increase in the production term represents the extraction of energy from the shear, as also manifested by the decrease in MKE in figure 2 (f). This energy is converted to turbulence as shown by the increase in $d(TKE)/dt$ term. At a later time ($t > 60s$) the turbulence decays owing to the loss to dissipation and buoyancy flux. The differences in TKE budget terms among different cases are attributed to the different grid resolutions. The deep learning based subgrid scale model is able to mimic the magnitude and variation of the production (P), buoyancy flux (B), and the advection ($d(TKE)/dt$) terms at different grid resolutions similar to the Dynamic Smagorinsky model. There are small differences in the evolution of the the total dissipation ($\varepsilon + \varepsilon_{sgs}$) which is attributed to the differences in ν_{sgs} computed by the DNNs amd the Dynamic Smagorinsky model.

Comparison of the time evolution of vertically integrated MKE budget terms (figure 4(a),(c),(e)) and density variance budget terms (figure 4(b),(d),(f)) between the deep learning based subgrid scale model and the Dynamic Smagorinsky model for the cases at $Re_0 = 10^5$ are shown in figure 4. The purpose of analyzing the MKE budget is to appraise the performance of the deep learning based subgrid scale model in the conservation of the mean kinetic energy. The MKE transport terms, MKE dissipation terms and mean buoyancy production term (see equations $25 - 28$ in SI) become negligible when integrated in the vertical direction. Therefore production is the only source of the rate of change of MKE. The deep learning based subgrid scale model respect the conservation of MKE for all grid resolutions similar to the Dynamic Smagorinky model as shown in figures 4(a),(c),(e). The scalar transport terms (resolved and subgrid) when integrated vertically also become negligible and are therefore not shown in figure 4(b),(d),(f). The scalar production (P_ρ) term is the source whereas the scalar dissipation ($\chi_\rho + \chi_{\rho,sgs}$) acts as a sink for the evolution of scalar variance advection term ($\frac{d\langle\overline{\rho}'^2\rangle}{dt}$). Once again the deep learning based subgrid scale model successfully captures the variation of the scalar production and scalar variance advection terms similar to the Dynamic Smagorinsky model for all the grid resolutions at $Re_0 = 10^5$ with small differences in the evolution of the scalar dissipation term. These differences are associated with the differences in κ_{sgs} calculated from the deep learning based subgrid scale model and the Dynamic Smagorinsky model.

4 Conclusions

Deep learning is used as a new technique to emulate the subgrid scale viscosity and diffusivity for large eddy simulations of turbulent shear flows at Reynolds number relevant to

This article is protected by copyright. All rights reserved.

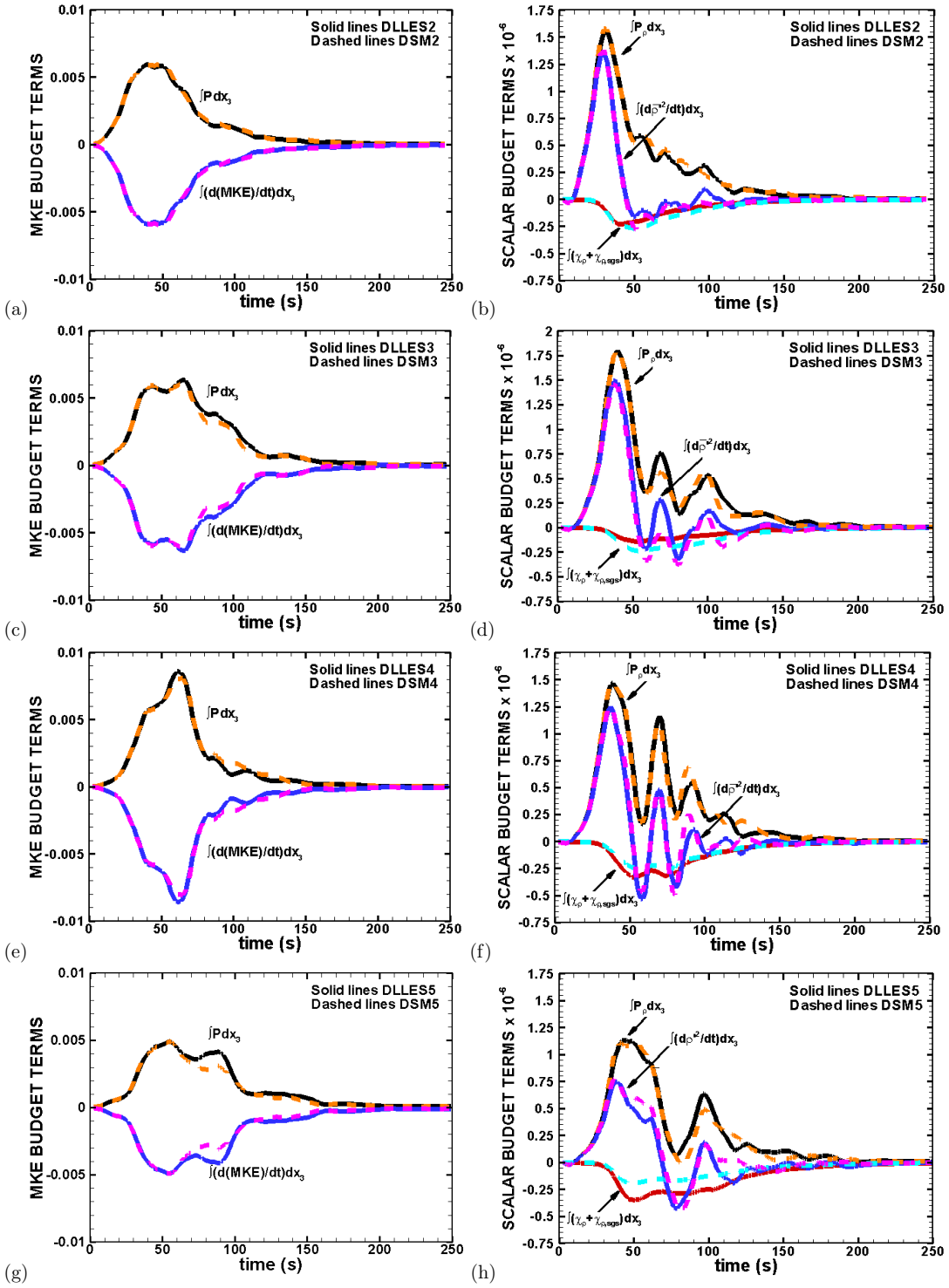


Figure 4. Comparison of vertically integrated MKE budget terms (a) DLLES2 and DSM2, (c) DLLES3 and DSM3, (e) DLLES4 and DSM4, and (g) DLLES5 and DSM5. Comparison of vertically integrated density variance budget terms between (b) DLLES2 and DSM2, (d) DLLES3 and DSM3, (f) DLLES4 and DSM4, and (h) DLLES5 and DSM5 at $Re_0 = 10^5$.

This article is protected by copyright. All rights reserved.

Author Manuscript

19448007, 2020, 12, Downloaded from https://agupubs.onlinelibrary.wiley.com/doi/10.1029/2020GL087005 by Oak Ridge National Laboratory, Wiley Online Library on [27/03/2023]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

that observed in the oceans and atmosphere. The deep learning based subgrid scale model is developed from a dataset obtained by simulating a shear layer using the traditional Dynamic Smagorinsky model at a particular grid resolution at $Re_0 = 10^4$. Once this subgrid scale model is trained and validated, it replaces the Dynamic Smagorinsky model for simulating shear flow at $Re_0 = 10^5$ for different grid configurations. The deep learning based subgrid scale model predicts ν_{sgs} and κ_{sgs} from velocities, strain rates and density gradients such that all the qualitative features during the evolution of the shear layer such as the KH billows, their instabilities, secondary KH instabilities in the braids and eventual transition to turbulence are captured for not only the same grid resolution but for a finer and coarser grid resolution as well. This model also computes the subgrid scale viscosity and diffusivity $\sim 2-4$ times quicker than the Dynamic Smagorinsky model. This acceleration of the deep learning based subgrid scale model is primarily attributed to the simplification of the complex computation of the subgrid scale viscosity and diffusivity to vector matrix multiplications. As the computational time required to determine ν_{sgs} and κ_{sgs} turns out to be the prime computational burden for the present cases, a $\sim 2-4$ times acceleration by the deep learning based subgrid scale model results in a $\sim 2-2.5$ times speed up of the entire simulation. A deep learning based subgrid scale model is computationally efficient when used for parametric studies. The development of deep learning based subgrid scale models may require data from additional simulations. However the consolidated cost of the additional simulations and training the DNNs will be compensated as the deep learning based subgrid scale model accelerate the cases with different parameters.. Quantitative analysis reveal that the deep learning based subgrid scale model conserves mean kinetic energy during the evolution of the shear layer similar to the Dynamic Smagorinsky model. This surrogate LES model is also able to mimic the terms of the turbulent kinetic energy budget and the density variance budget as the Dynamic Smagorinsky model.

The primary motivation of developing a deep learning based subgrid scale model is to verify the feasibility of machine learning algorithms to determine the subgrid scale processes accurately in geophysical flows. This study builds a confidence in experimenting with deep learning alogrithms in emulating the subgrid scale processes in climate models similar to Gentine et al. (2018); Rasp et al. (2018). The subgrid scale processes such as rain, clouds, mixing owing to turbulence are typically represented by physical models in current climate models. These physical models are sometimes complicated and introduce uncertainties in the outcomes of low-resolution climate simulations. The deep learning based subgrid scale models developed from a high resolution climate simulation dataset will be a simpler representation of the subgrid scale processes, and can easily be implemented in the climate model. Although this investigation was carried out as a proof of concept, the results are promising enough to motivate futuristic research in data-driven emulation of the small scale processes in climate and earth science models. The capability of deep neural networks in parameterizing the turbulent processes at finer as well as coarser resolutions accurately at reduced computation cost has been demonstrated in this study. Therefore deep neural network based sgs parameterization can be an alternative to the sgs parameterizations based on physically motivated approaches in ocean (Jansen, Held, Adcroft, & Hallberg, 2015) and climate (Pieroth et al., 2018) models. Attempts similar to Gentine et al. (2018); Pal et al. (2019) will be made in the future to implement deep learning based subgrid scale models to emulate the different subgrid scale processes in a climate model.

Acknowledgments

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.

Data and materials availability: The code for training the DNN is available at **https://doi.org/10.5281/zenodo.3748123** . This repository also contains the weights matrices and bias vectors. The dataset for DNN training and testing amount to several GB and are available from the author upon request.

This article is protected by copyright. All rights reserved.

Author Manuscript

19448007, 2020, 12, Downloaded from https://agupubs.onlinelibrary.wiley.com/doi/10.1029/2020GL087005 by Oak Ridge National Laboratory, Wiley Online Library on [27/03/2023]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

Author Manuscript

19448007, 2020, 12, Downloaded from https://agupubs.onlinelibrary.wiley.com/doi/10.1029/2020GL087005 by Oak Ridge National Laboratory, Wiley Online Library on [27/03/2023]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

References

Anderson, G. J., & Lucas, D. D. (2018). Machine learning predictions of a multiresolution climate model ensemble. *Geophys. Res. Lett.*, *45*(9), 4273–4280.

Bolton, T., & Zanna, L. (2019). Applications of deep learning to ocean data inference and subgrid parameterization. *Journal of Advances in Modeling Earth Systems*, *11*(1), 376–399.

Brenowitz, N. D., & Bretherton, C. S. (2018). Prognostic validation of a neural network unified physics parameterization. *Geophys. Res. Lett.*, *45*(12), 6289–6298.

Brucker, K. A., & Sarkar, S. (2007). Evolution of an initially turbulent stratified shear layer. *Phys. Fluids*, *19*(10), 105105.

Brucker, K. A., & Sarkar, S. (2010). A comparative study of self-propelled and towed wakes in a stratified fluid. *J. Fluid Mech.*, *652*, 373–404.

Caulfield, C., & Peltier, W. (2000). The anatomy of the mixing transition in homogeneous and stratified free shear layers. *J. Fluid Mech.*, *413*, 1–47.

Chalamalla, V. K., Santilli, E., Scotti, A., Jalali, M., & Sarkar, S. (2017). Somar-les: A framework for multi-scale modeling of turbulent stratified oceanic flows. *Ocean Modelling*, *120*, 101–119.

Chamecki, M., Meneveau, C., & Parlange, M. B. (2007). The local structure of atmospheric turbulence and its effect on the smagorinsky model for large eddy simulation. *J. Atmos. Sci.*, *64*(6), 1941–1958.

Dechter, R. (1986). *Learning while searching in constraint-satisfaction problems.*

Gentine, P., Pritchard, M., Rasp, S., Reinaudi, G., & Yacalis, G. (2018). Could machine learning break the convection parameterization deadlock? *Geophys. Res. Lett.*, *45*(11), 5742–5751.

Germano, M., Piomelli, U., Moin, P., & Cabot, W. H. (1991). A dynamic subgrid-scale eddy viscosity model. *Phys. Fluids A: Fluid Dynamics*, *3*(7), 1760–1765.

Geyer, W. R., Lavery, A. C., Scully, M. E., & Trowbridge, J. H. (2010). Mixing by shear instability at high reynolds number. *Geophys. Res. Lett.*, *37*(22).

Jalali, M., & Sarkar, S. (2017). Large eddy simulation of flow and turbulence at the steep topography of luzon strait. *Geophys. Res. Lett.*, *44*(18), 9440–9448.

Jansen, M. F., Held, I. M., Adcroft, A., & Hallberg, R. (2015). Energy budget-based backscatter in an eddy permitting primitive equation model. *Ocean Modelling*, *94*, 15–26.

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, *521*(7553), 436.

Lilly, D. K. (1992). A proposed modification of the germano subgrid-scale closure method. *Phys. Fluids A: Fluid Dynamics*, *4*(3), 633–635.

Ling, J., Kurzawski, A., & Templeton, J. (2016). Reynolds averaged turbulence modelling using deep neural networks with embedded invariance. *J. Fluid Mech.*, *807*, 155–166.

Moeng, C.-H., & Sullivan, P. P. (1994). A comparison of shear-and buoyancy-driven planetary boundary layer flows. *J. Atmos. Sci.*, *51*(7), 999–1022.

O'Gorman, P. A., & Dwyer, J. G. (2018). Using machine learning to parameterize moist convection: Potential for modeling of climate, climate change, and extreme events. *J. Adv. in Modeling Earth Sys.*, *10*(10), 2548–2563.

Pal, A., & Chalamalla, V. K. (2020). Evolution of plumes and turbulent dynamics in deep-ocean convection. *J. Fluid Mech.*, *889*, A35.

Pal, A., de Stadler, M. B., & Sarkar, S. (2013). The spatial evolution of fluctuations in a self-propelled wake compared to a patch of turbulence. *Phys. Fluids*, *25*(9), 095106.

Pal, A., Mahajan, S., & Norman, M. R. (2019). Using deep neural networks as costeffective surrogate models for superparameterized e3sm radiative transfer. *Geophys. Res. Lett.*, *46*(11), 6069-6079.

Pal, A., & Sarkar, S. (2015). Effect of external turbulence on the evolution of a wake in stratified and unstratified environments. *J. Fluid Mech.*, *772*, 361–385.

Parish, E. J., & Duraisamy, K. (2016). A paradigm for data-driven predictive modeling using field inversion and machine learning. *J. Comput. Phys.*, *305*, 758–774.

Pham, H. T., & Sarkar, S. (2010). Transport and mixing of density in a continuously strat-

This article is protected by copyright. All rights reserved.

ified shear layer. *J. Turbulence*(11), N24.
Pham, H. T., & Sarkar, S. (2014). Large eddy simulations of a stratified shear layer. *J. Fluid Eng.*, *136*(6), 060913.
Pham, H. T., & Sarkar, S. (2017). Turbulent entrainment in a strongly stratified barrier layer. *J. Geophys. Res. Ocn.*, *122*(6), 5075–5087.
Pham, H. T., & Sarkar, S. (2018). Ageostrophic secondary circulation at a submesoscale front and the formation of gravity currents. *J. Phys. Oceanogr.*, *48*, 2507–2529.
Pham, H. T., Sarkar, S., & Brucker, K. A. (2009). Dynamics of a stratified shear layer above a region of uniform stratification. *J. Fluid Mech.*, *630*, 191–223.
Pieroth, M., Dolaptchiev, S. I., Zacharuk, M., Heppelmann, T., Gritsun, A., & Achatz, U. (2018). Climate dependence in empirical parameters of subgrid-scale parameterizations using the fluctuation–dissipation theorem. *J. Atmos. Sci.*, *75*(11), 3843–3860.
Rasp, S., Pritchard, M. S., & Gentine, P. (2018). Deep learning to represent subgrid processes in climate models. *Proceedings of the National Academy of Sciences*, *115*(39), 9684–9689.
Reichstein, M., Camps-Valls, G., Stevens, B., Jung, M., Denzler, J., Carvalhais, N., et al. (2019). Deep learning and process understanding for data-driven earth system science. *Nature*, *566*(7743), 195.
Siebesma, A. P., Bretherton, C. S., Brown, A., Chlond, A., Cuxart, J., Duynkerke, P. G., ... others (2003). A large eddy simulation intercomparison study of shallow cumulus convection. *J. Atmos. Sci.*, *60*(10), 1201–1219.
Smyth, W., Moum, J., & Caldwell, D. (2001). The efficiency of mixing in turbulent patches: Inferences from direct simulations and microstructure observations. *J. Phys. Oceanogr.*, *31*(8), 1969–1992.
Smyth, W. D., & Moum, J. N. (2000a). Anisotropy of turbulence in stably stratified mixing layers. *Phys. Fluids*, *12*(6), 1343–1362.
Smyth, W. D., & Moum, J. N. (2000b). Length scales of turbulence in stably stratified mixing layers. *Phys. Fluids*, *12*(6), 1327–1342.
Srinivasan, P., Guastoni, L., Azizpour, H., Schlatter, P., & Vinuesa, R. (2019). Predictions of turbulent shear flows using deep neural networks. *Phy. Rev. Fluids*, *4*(5), 054603.
Stevens, B., Moeng, C.-H., Ackerman, A. S., Bretherton, C. S., Chlond, A., de Roode, S., ... others (2005). Evaluation of large-eddy simulations via observations of nocturnal marine stratocumulus. *Mon. Weather Rev.*, *133*(6), 1443–1462.
Tracey, B., Duraisamy, K., & Alonso, J. (2013). Application of supervised learning to quantify uncertainties in turbulence and combustion modeling. In *51st aiaa aerospace sciences meeting including the new horizons forum and aerospace exposition* (p. 259).
Vreman, B., Geurts, B., & Kuerten, H. (1997). Large-eddy simulation of the turbulent mixing layer. *J. Fluid Mech.*, *339*, 357–390.
Watson, P. A. (2019). Applying machine learning to improve simulations of a chaotic dynamical system using empirical error correction. *J. Adv. in Modeling Earth Sys.*, *https://doi.org/10.1029/2018MS001597*.
Zhang, Z. J., & Duraisamy, K. (2015). Machine learning methods for data-driven turbulence modeling. In *22nd aiaa computational fluid dynamics conference* (p. 2460).

Supporting References:

1. Kundu, P., & Cohen, I. (2010). Fluid mechanics. Elsevier Science. Retrieved from https://books.google.com/books?id=d9B5NElxUKwC
2. Pham, H. T., & Sarkar, S. (2014). Large eddy simulations of a stratified shear layer. J. Fluid Eng., 136 (6), 060913.

This article is protected by copyright. All rights reserved.

Author Manuscript

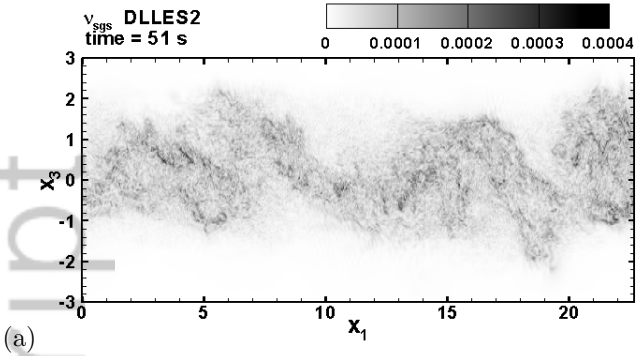
19448007, 2020, 12, Downloaded from https://agupubs.onlinelibrary.wiley.com/doi/10.1029/2020GL087005 by Oak Ridge National Laboratory, Wiley Online Library on [27/03/2023]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

Figure1.

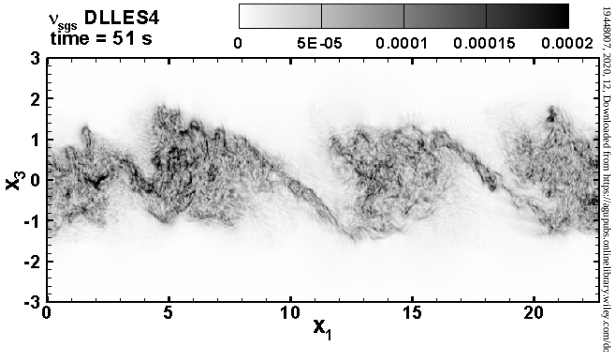
Author Manuscript

This article is protected by copyright. All rights reserved.

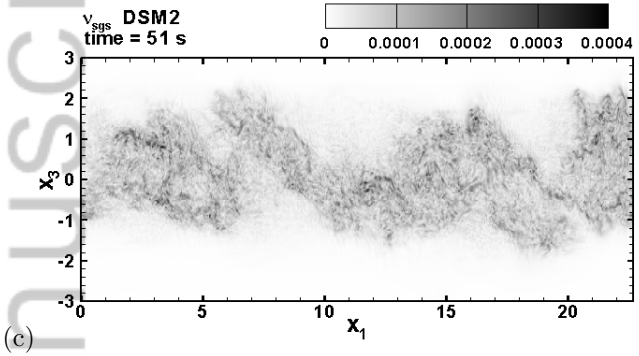
19448007, 2020, 12, Downloaded from https://agupubs.onlinelibrary.wiley.com/doi/10.1029/2020GL087005 by Oak Ridge National Laboratory, Wiley Online Library on [27/03/2023]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License



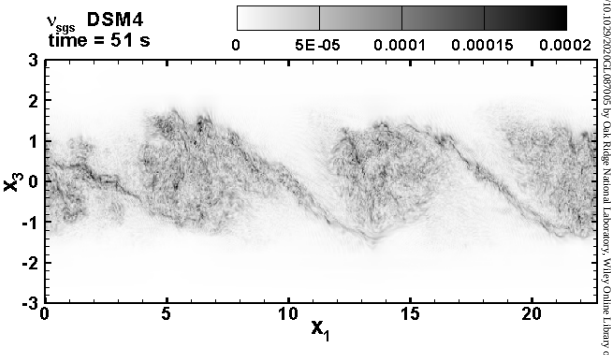
(a)



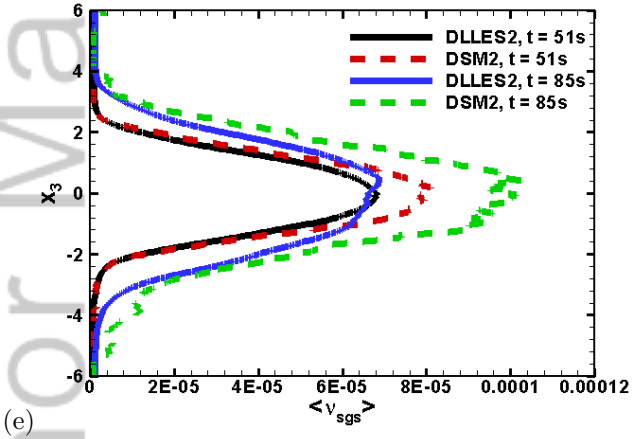
(b)



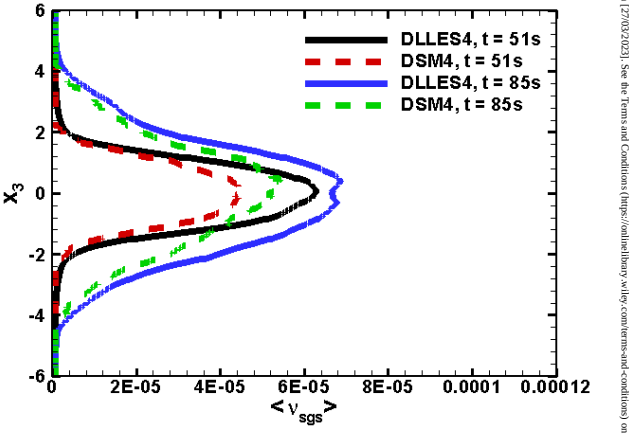
(c)



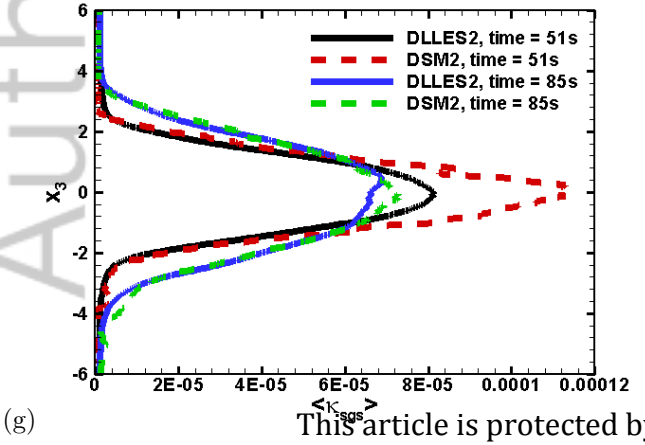
(d)



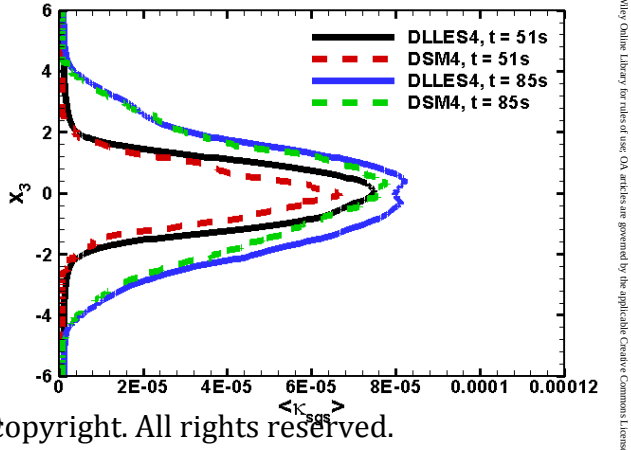
(e)



(f)



(g)



(h)

This article is protected by copyright. All rights reserved.

Author Manuscript

19448007, 2020, 12, Downloaded from https://agupubs.onlinelibrary.wiley.com/doi/10.1029/2020GL087005 by Oak Ridge National Laboratory, Wiley Online Library on [27/03/2023]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

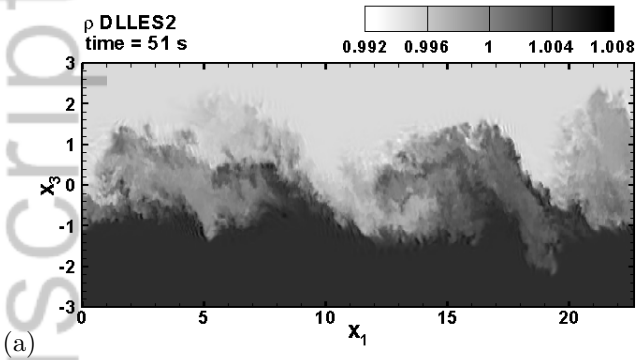
Figure2.

Author Manuscript

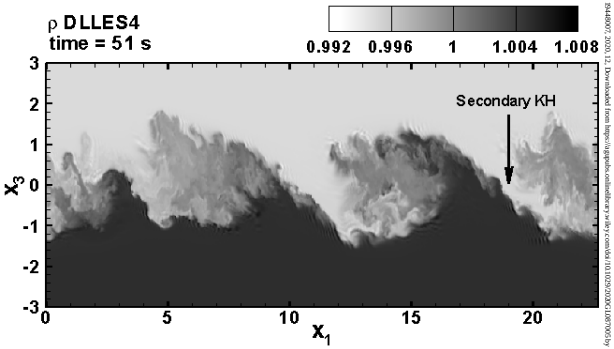
This article is protected by copyright. All rights reserved.

19448007, 2020, 12, Downloaded from https://agupubs.onlinelibrary.wiley.com/doi/10.1029/2020GL087005 by Oak Ridge National Laboratory, Wiley Online Library on [27/03/2023]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

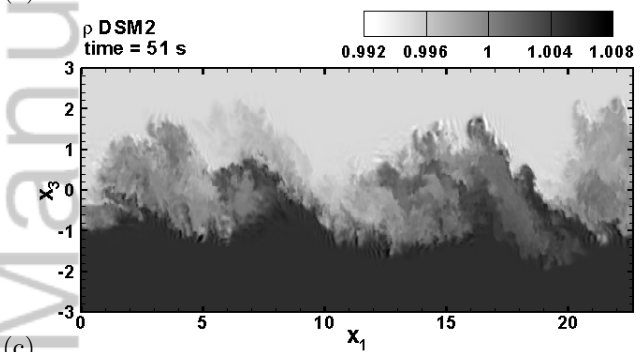
Author Manuscript



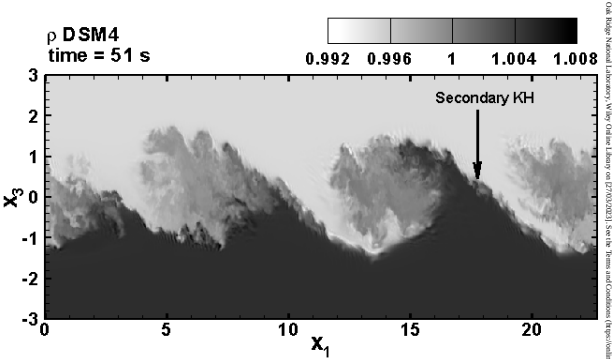
(a)



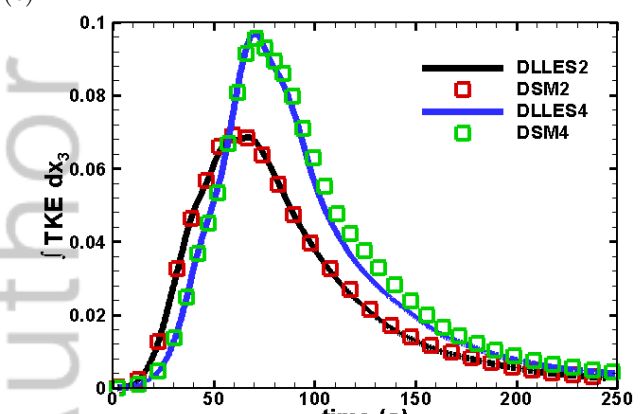
(b)



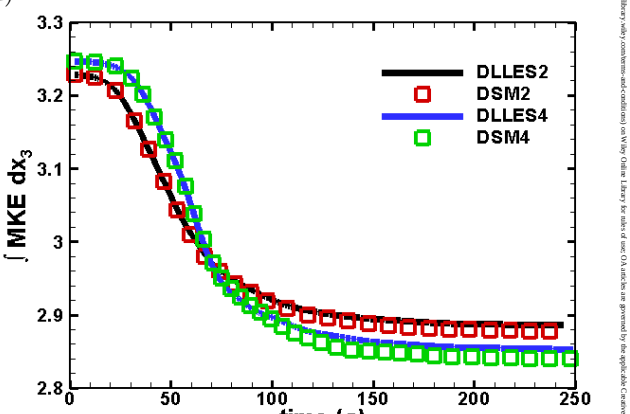
(c)



(d)



(e)



(f)

This article is protected by copyright. All rights reserved.

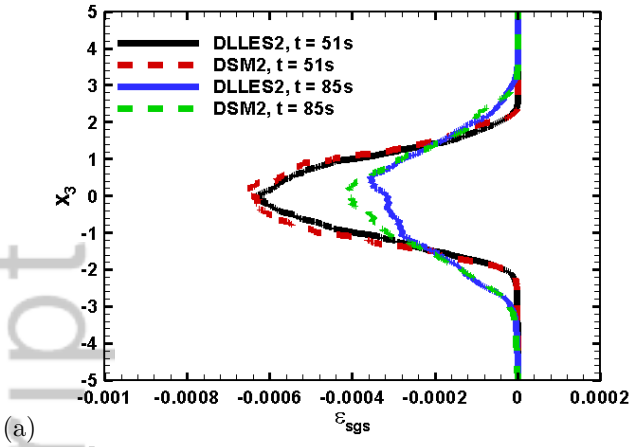
8440007, 2020, 12, Downloaded from https://agupubs.onlinelibrary.wiley.com/doi/10.1029/2020GL087005 by Oak Ridge National Laboratory, Wiley Online Library on [27/10/2021]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

Figure3.

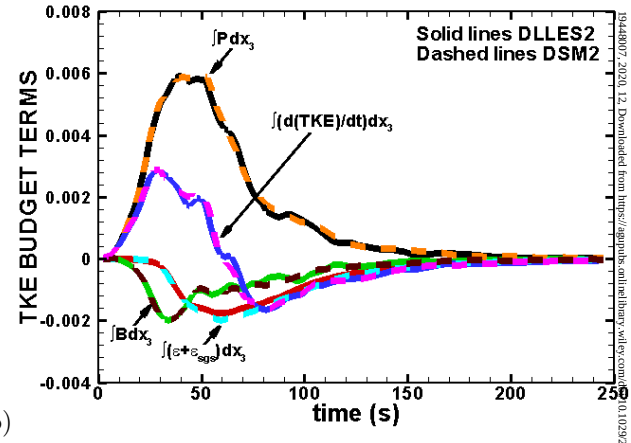
Author Manuscript

This article is protected by copyright. All rights reserved.

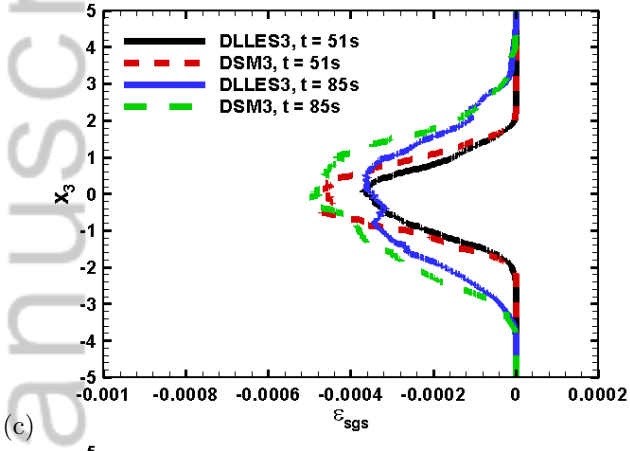
19448007, 2020, 12, Downloaded from https://agupubs.onlinelibrary.wiley.com/doi/10.1029/2020GL087005 by Oak Ridge National Laboratory, Wiley Online Library on [27/03/2023]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License



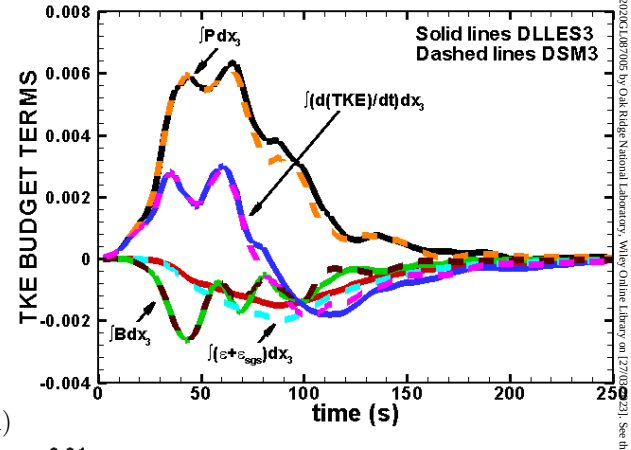
(a)



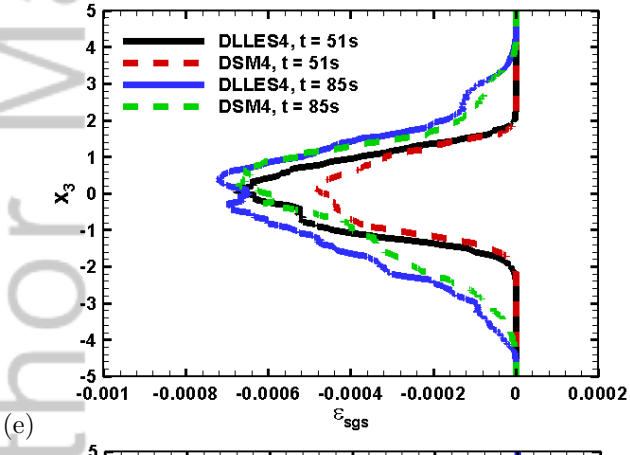
(b)



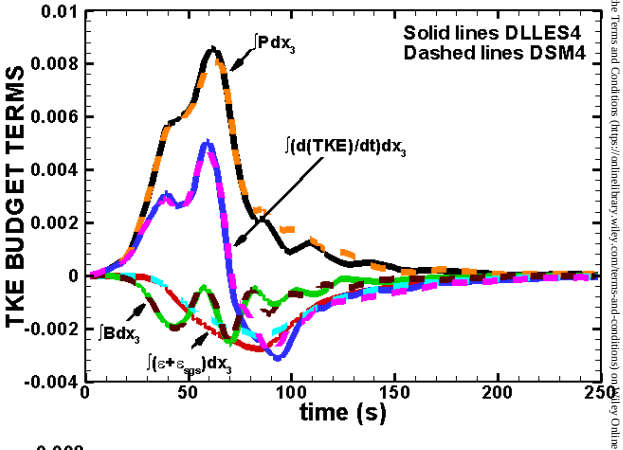
(c)



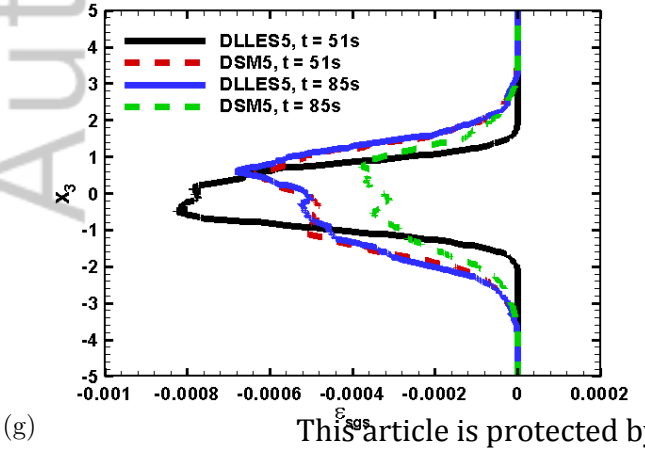
(d)



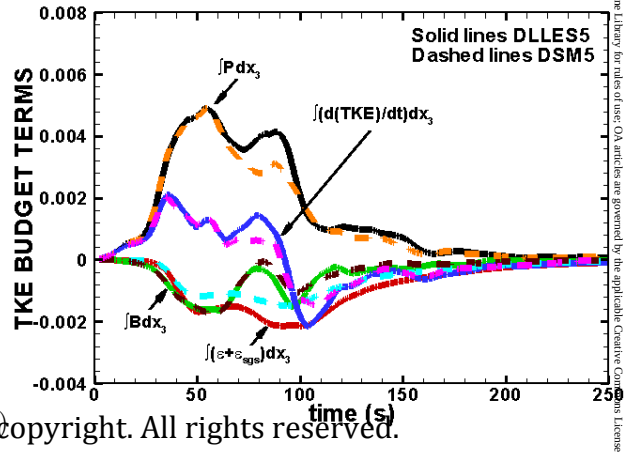
(e)



(f)



(g)



(h)

Author Manuscript

This article is protected by copyright. All rights reserved.

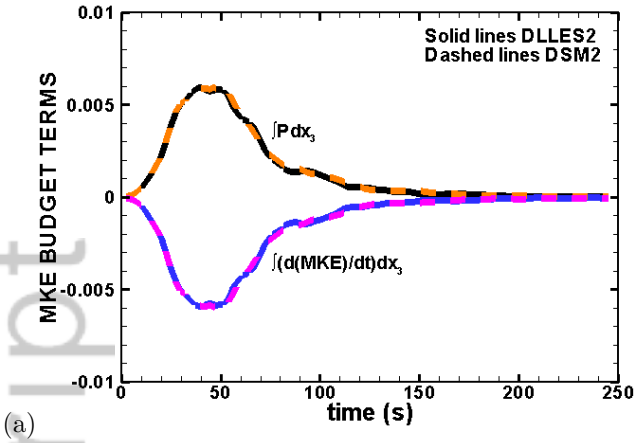
19448007, 2020, 12, Downloaded from https://agupubs.onlinelibrary.wiley.com/doi/10.1029/2020GL087095 by Oak Ridge National Laboratory, Wiley Online Library on [27/07/023]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

Figure4.

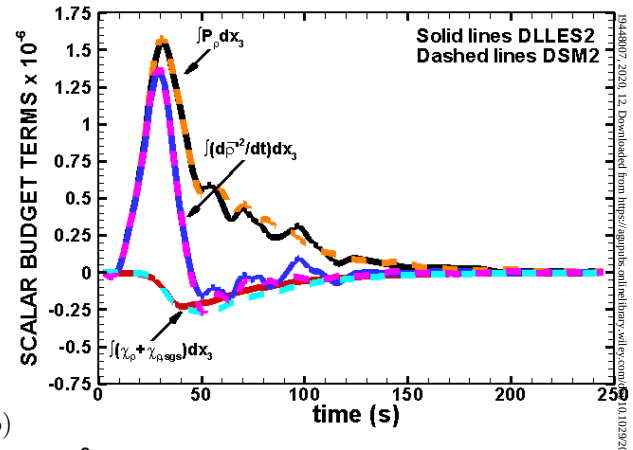
Author Manuscript

This article is protected by copyright. All rights reserved.

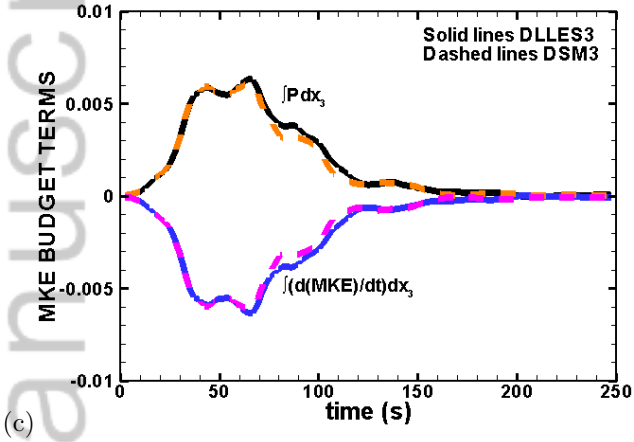
19448007, 2020, 12, Downloaded from https://agupubs.onlinelibrary.wiley.com/doi/10.1029/2020GL087005 by Oak Ridge National Laboratory, Wiley Online Library on [27/03/2023]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License



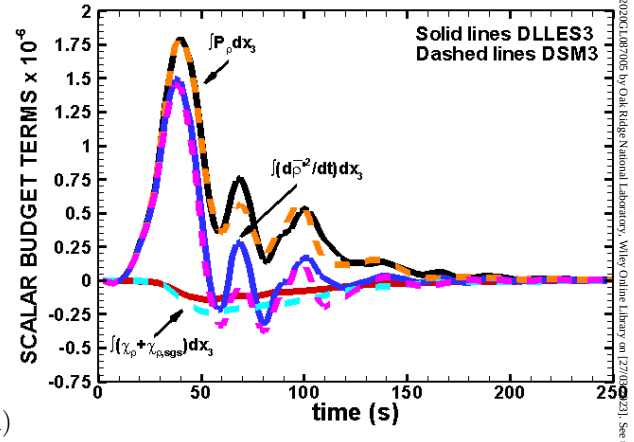
(a)



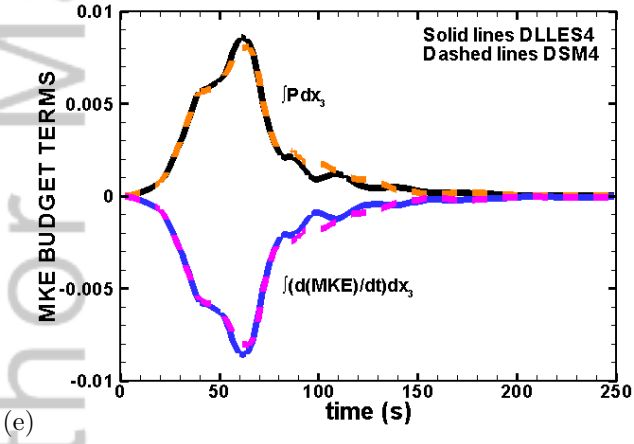
(b)



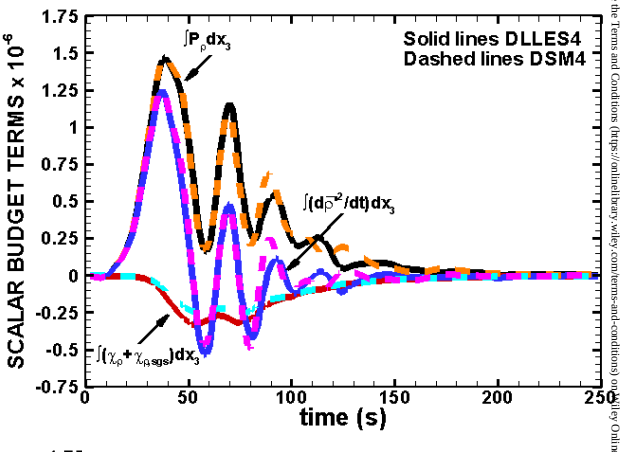
(c)



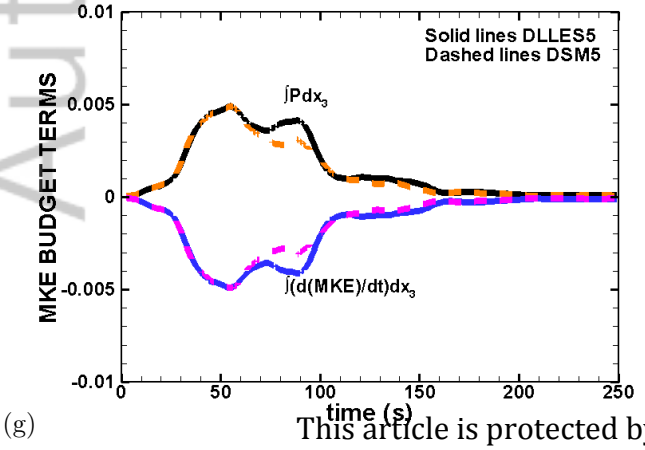
(d)



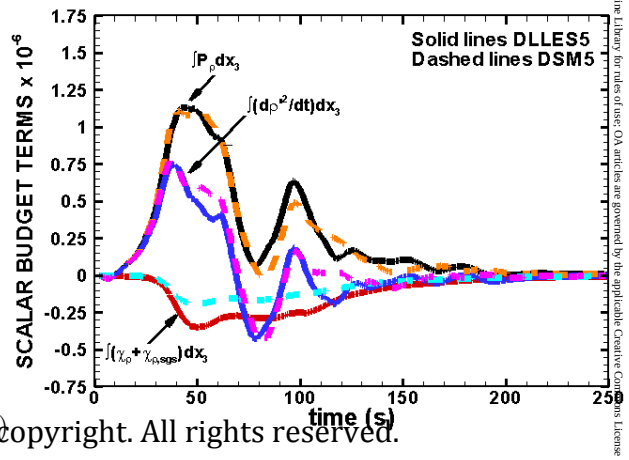
(e)



(f)



(g)

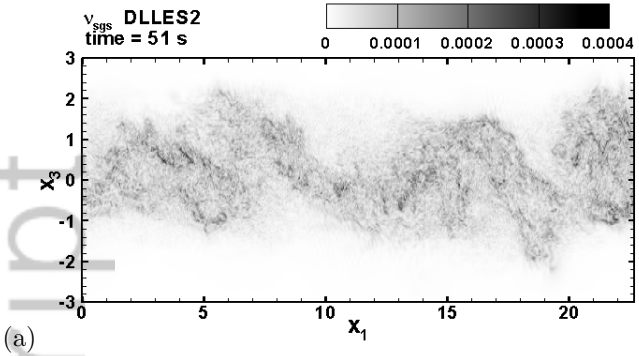


(h)

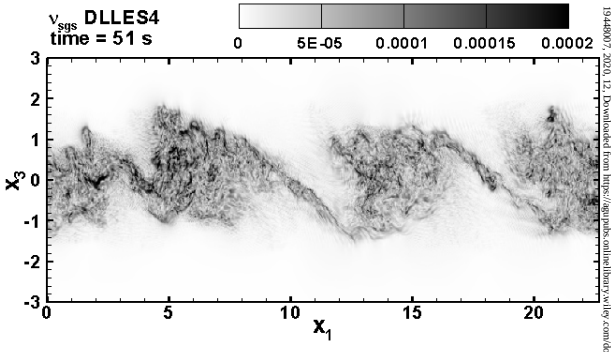
This article is protected by copyright. All rights reserved.

Author Manuscript

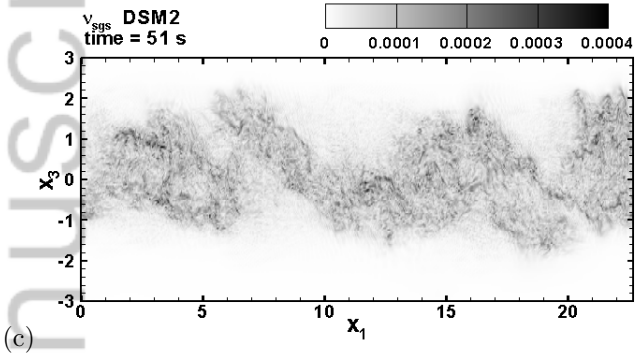
19448007, 2020, 12, Downloaded from https://agupubs.onlinelibrary.wiley.com/doi/10.1029/2020GL087095 by Oak Ridge National Laboratory, Wiley Online Library on [27/07/023]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License



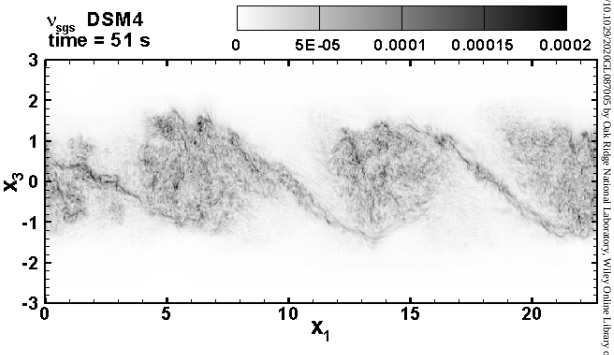
(a)



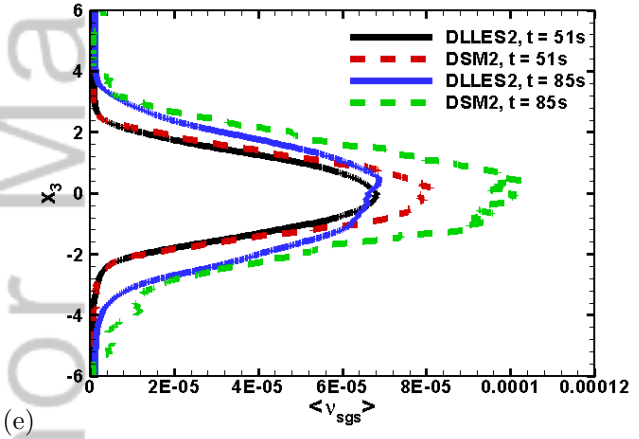
(b)



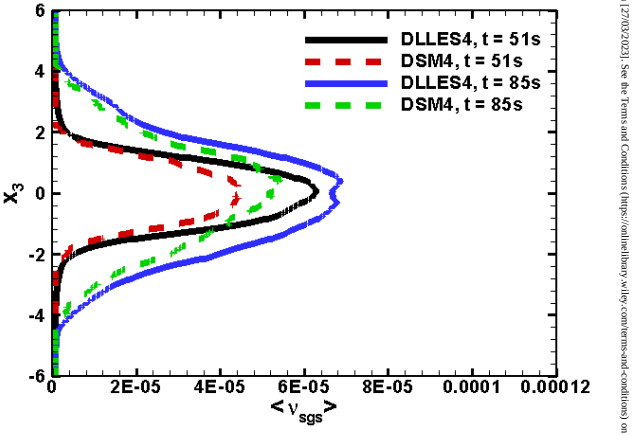
(c)



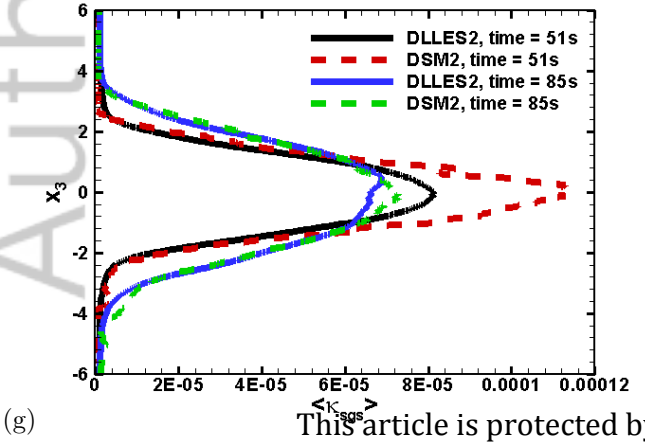
(d)



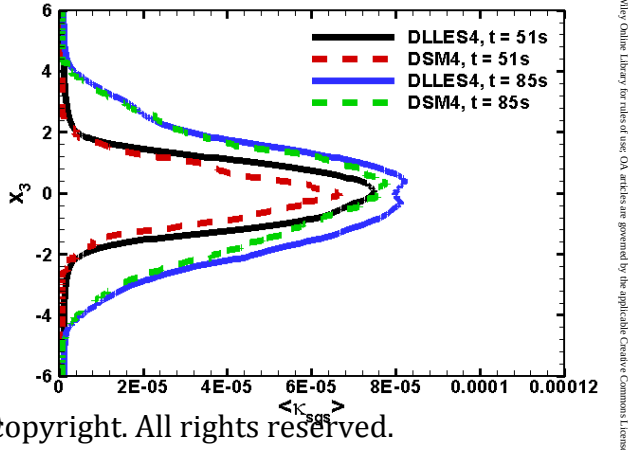
(e)



(f)



(g)



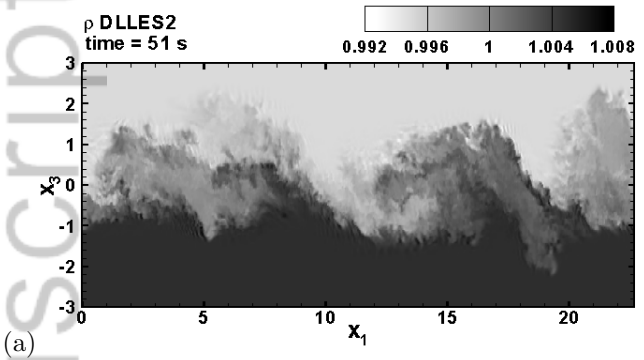
(h)

This article is protected by copyright. All rights reserved.

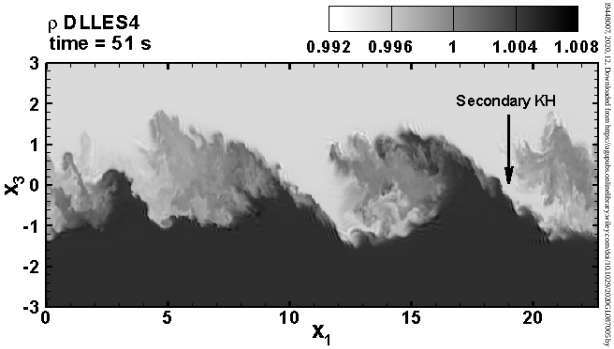
Author Manuscript

19448007, 2020, 12, Downloaded from https://agupubs.onlinelibrary.wiley.com/doi/10.1029/2020GL087005 by Oak Ridge National Laboratory, Wiley Online Library on [27/03/2023]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

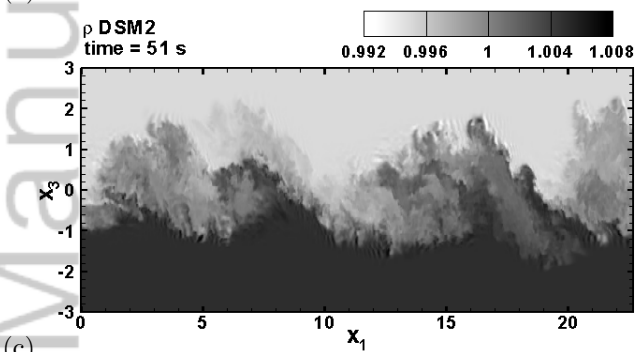
Author Manuscript



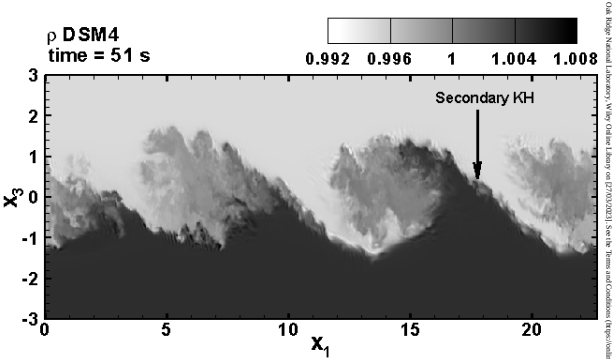
(a)



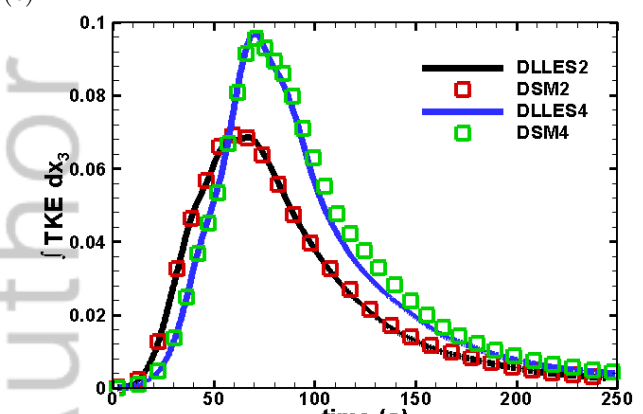
(b)



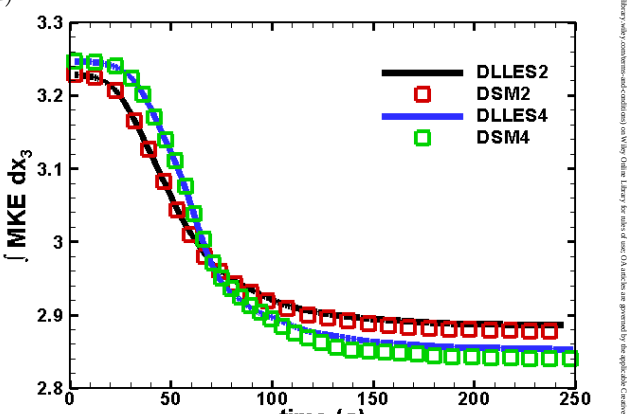
(c)



(d)



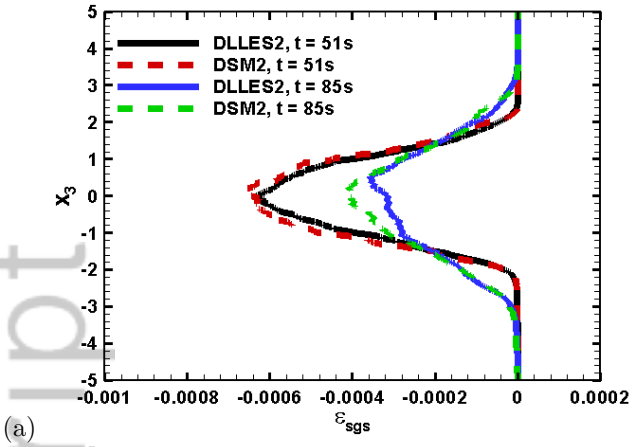
(e)



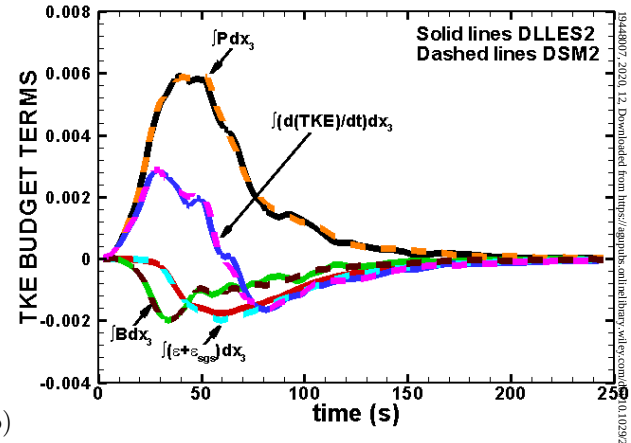
(f)

This article is protected by copyright. All rights reserved.

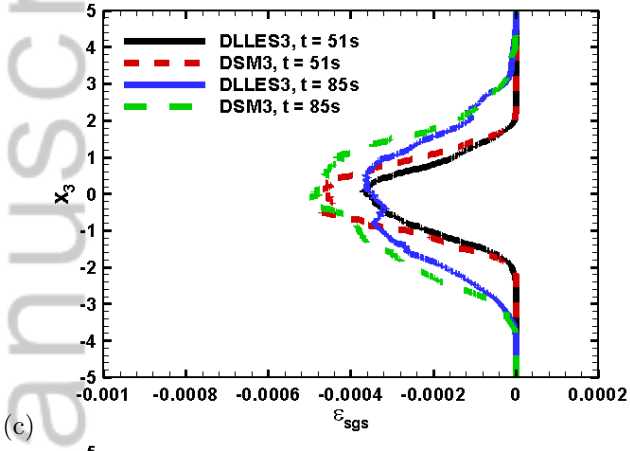
8440007, 2020, 12, Downloaded from https://agupubs.onlinelibrary.wiley.com/doi/10.1029/2020GL087005 by Oak Ridge National Laboratory, Wiley Online Library on [27/10/2021]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License



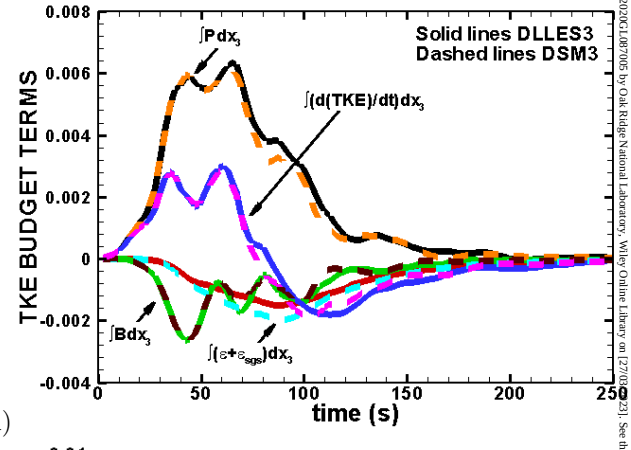
(a)



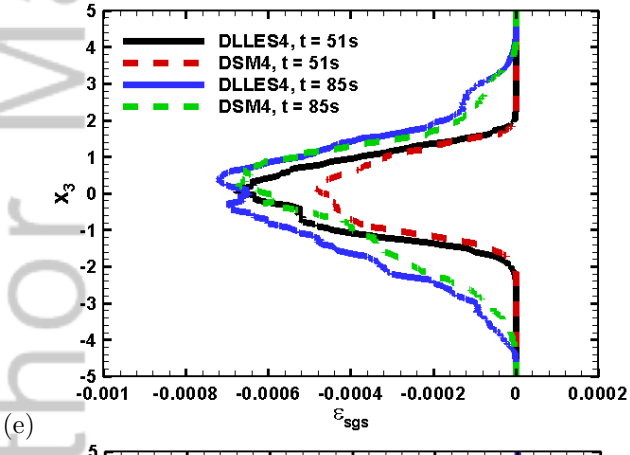
(b)



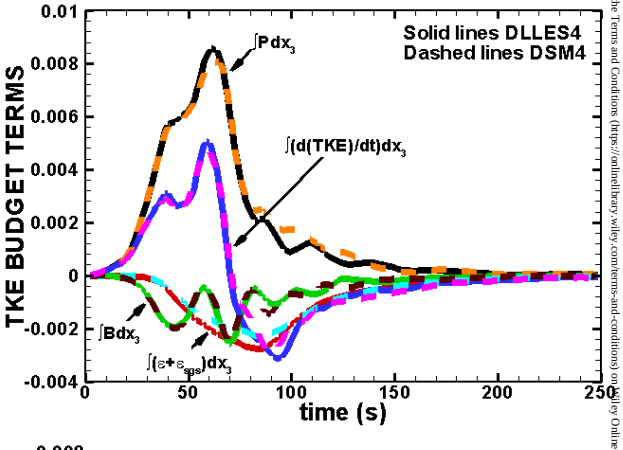
(c)



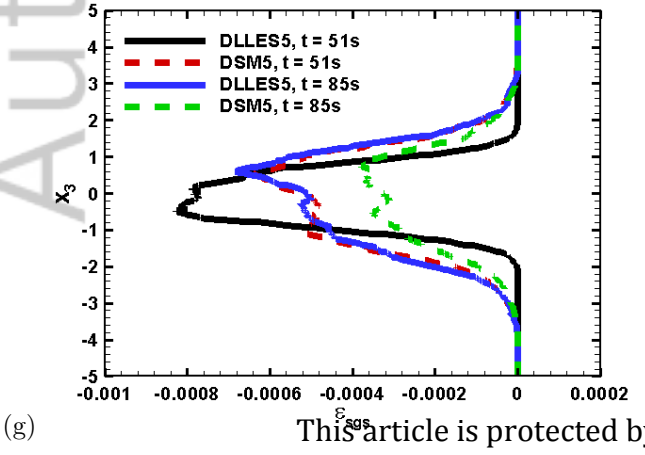
(d)



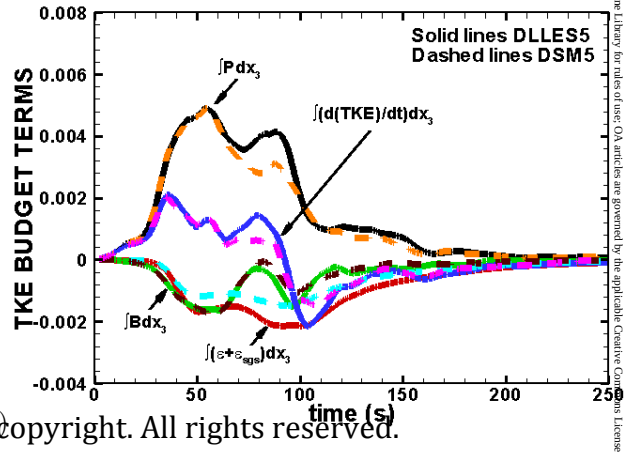
(e)



(f)



(g)

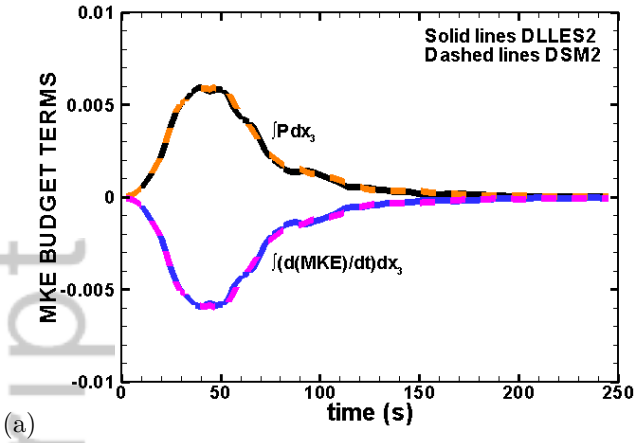


(h)

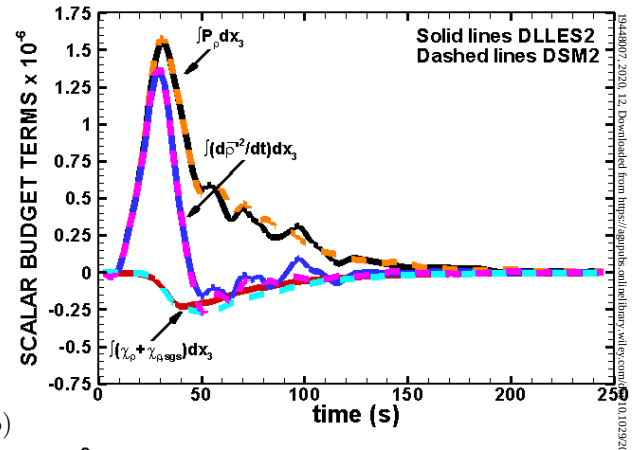
Author Manuscript

This article is protected by copyright. All rights reserved.

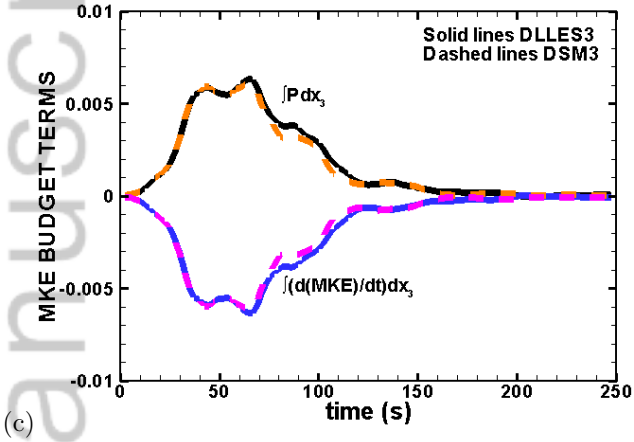
19448007, 2020, 12, Downloaded from https://agupubs.onlinelibrary.wiley.com/doi/10.1029/2020GL087095 by Oak Ridge National Laboratory, Wiley Online Library on [27/07/023]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License



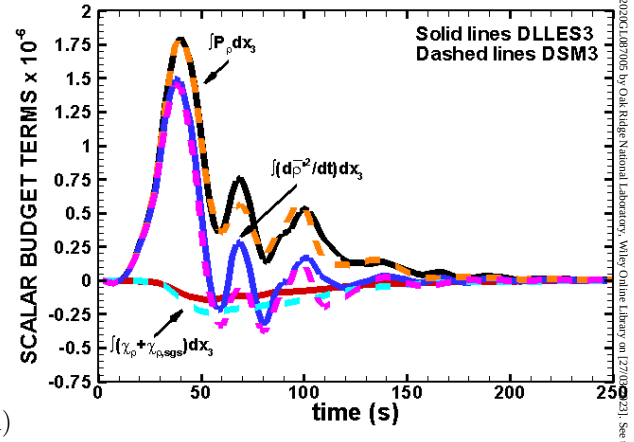
(a)



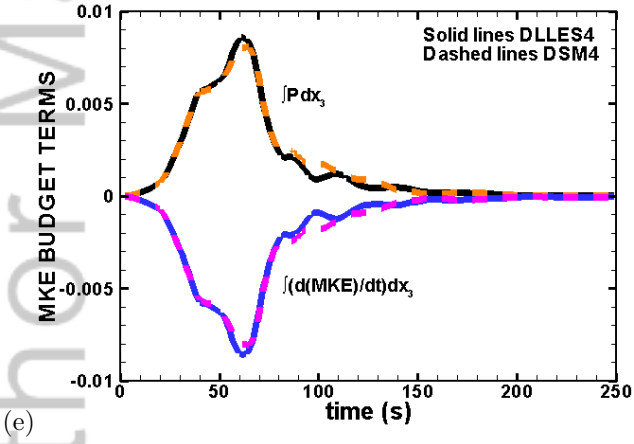
(b)



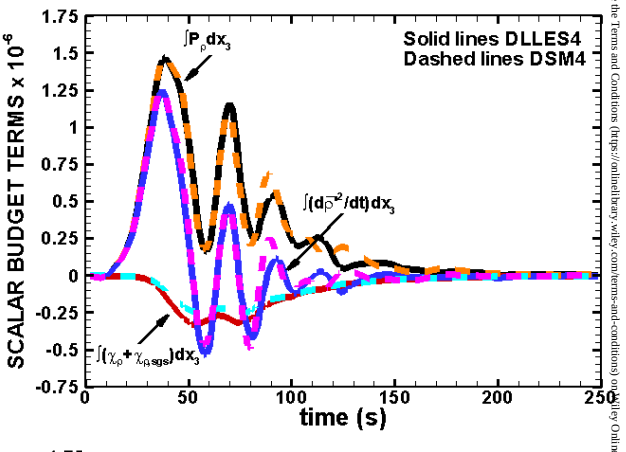
(c)



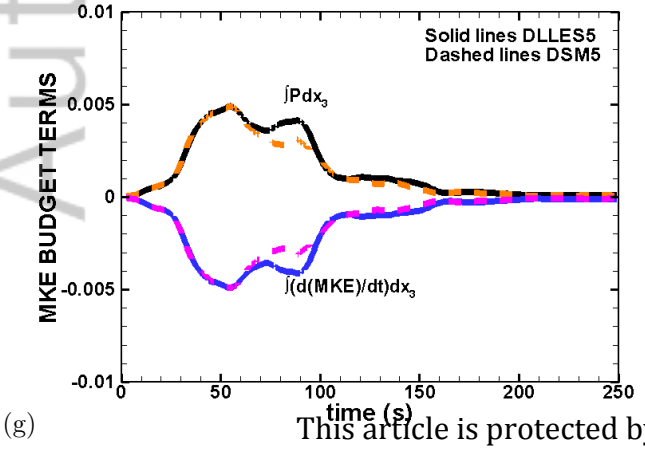
(d)



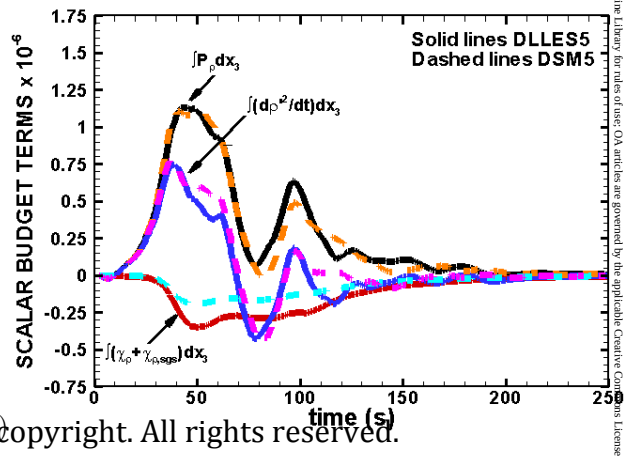
(e)



(f)



(g)



(h)

This article is protected by copyright. All rights reserved.

Author Manuscript

19448007, 2020, 12, Downloaded from https://agupubs.onlinelibrary.wiley.com/doi/10.1029/2020GL087095 by Oak Ridge National Laboratory, Wiley Online Library on [27/07/023]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License