

Consistency requirement of data-driven subgrid-scale modeling in large-eddy simulation

Xinyi Huang ^{1,2,*}, Sze Chai Leung ³, and H. Jane Bae ¹

¹Graduate Aerospace Laboratories, [California Institute of Technology](#), Pasadena, California 91125, USA

²Department of Earth, Environmental and Planetary Sciences, [Brown University](#), Providence, Rhode Island 02906, USA

³Department of Mechanical and Civil Engineering, [California Institute of Technology](#), Pasadena, California 91125, USA



(Received 16 June 2025; accepted 20 November 2025; published 12 January 2026)

Data-driven subgrid-scale (SGS) modeling in the large-eddy simulations (LESs) suffers from an inconsistency between the *a priori* tests and the *a posteriori* tests, which make training accurate SGS models a difficult task. We study the difference in filtered high-fidelity data and LESs to identify the numerical deviation between the two cases, which is a combined impact of commutation error, numerical errors, and error coupling. The impact of the numerical deviation is examined through two SGS model formulations: the eddy-viscosity and the complex nonlinear models. By incorporating numerical deviations into model training, we enhance consistency, stabilize simulations, and improve predictions of energy spectra in *a posteriori* tests. Our findings highlight that data-driven methods introduce significant nonlinearity and equation coupling, exacerbating inconsistencies compared to non-data-driven approaches. Finally, while the impact of the numerical deviation can be generalized, achieving accurate model predictions necessitates a physically grounded model form and an optimal filter width.

DOI: [10.1103/yykb-6rvf](https://doi.org/10.1103/yykb-6rvf)

I. INTRODUCTION

Large-eddy simulation (LES) is a computational approach that resolves the large, energy-containing structures of turbulence while modeling the effects of smaller-scale motions, significantly reducing the computational cost compared to direct numerical simulation (DNS). Over the past few decades, LES has become a powerful tool for investigating a wide range of flows—from canonical configurations to complex real-world scenarios—offering rich physical insights [1–5]. To extend the applicability of subgrid-scale (SGS) models to practical flows, it is essential to incorporate a physical understanding of complex phenomena such as flow separation, pressure gradients, surface roughness, stratification, and compressibility [6–9]. However, developing SGS models for such flows is challenging due to factors including anisotropy, significant numerical discretization effects, and spatial inhomogeneity in resolution [10]. Extracting relevant physical features for model development in these regimes often demands considerable effort. In this context, recent advances in data-driven modeling offer promising opportunities to streamline and enhance SGS model development.

In recent years, data-driven methods have gained traction in SGS modeling due to their potential to uncover complex relationships between SGS terms and resolved quantities using high-fidelity data [11,12]. However, most developments have focused on canonical flows [13–15], with limited

*Contact author: xinyih@brown.edu

success in more complex flow scenarios [16–18]. Several factors contribute to the slow progress in applying data-driven SGS models to complex flows. First, these models face additional challenges in practical applications, such as numerical instability during deployment [12], limited performance gains over traditional models [13,17], and dependence on case-specific global information [16]. Even in canonical settings, data-driven models that appear highly accurate in training (*a priori*) tests can become unstable in simulations, often requiring artificial backscatter clipping and still underperforming classical models in *a posteriori* evaluations. Second, the “black-box” nature of many data-driven approaches makes it difficult to incorporate known physical principles and constraints—an advantage of traditional models. For instance, enforcing Galilean invariance is generally believed to enhance model generalizability [19]. Yet, due to interactions between numerical discretization and the online training process, enforcing seemingly appropriate constraints such as symmetry of the SGS stress tensor can sometimes degrade model performance significantly [16].

A major limitation of data-driven SGS models lies in the inconsistency between their *a priori* and *a posteriori* performance. During model development, i.e., in *a priori* testing, training and validation data are typically generated by filtering high-fidelity simulations. However, in *a posteriori* tests, the model is deployed in LESs, which evolve under different governing equations than those used to produce filtered DNS data. This mismatch arises from commutation errors between discretization and differentiation operators, as well as the coarse spatial resolution inherent to LES [20]. As a result, numerical derivatives of filtered quantities in LES often differ significantly from the filtered derivatives computed from high-resolution data. For example, given a quantity ϕ , unless specific filter and numerical derivative operators are chosen,

$$\frac{\overline{\Delta\phi}}{\Delta x_i} \neq \frac{\Delta\bar{\phi}}{\Delta x_i}, \quad \frac{\overline{\Delta\phi}}{\Delta t} \neq \frac{\Delta\bar{\phi}}{\Delta t}, \quad (1)$$

where the filtered quantities are represented by $\overline{(\cdot)}$. Note that after numerical discretization, the aforementioned two numerical derivatives are not necessarily performed on the same grid resolution, leading to additional numerical discrepancies even when using the same numerical scheme for both the DNS and the LES. By considering all numerical discrepancies, the LES equations for the incompressible Navier-Stokes equations consistent with filtered DNS are

$$\frac{\partial \bar{u}_i}{\partial t} + \frac{\partial \bar{u}_i \bar{u}_j}{\partial x_j} = -\frac{1}{\rho} \frac{\partial \bar{p}}{\partial x_i} + \nu \frac{\partial^2 \bar{u}_i}{\partial x_j \partial x_j} - \frac{\partial \tau_{ij}}{\partial x_j} + \delta_i^{\text{mom}}, \quad \frac{\partial \bar{u}_i}{\partial x_i} + \delta^{\text{div}} = 0, \quad (2)$$

where x_i is the Cartesian coordinate with $i \in \{1, 2, 3\}$, and u_i is the corresponding velocity component in the respective direction. The quantity given by ν is the kinematic viscosity, p is the pressure, and ρ is the fluid density. The SGS stress tensor is defined as $\tau_{ij} = \bar{u}_i \bar{u}_j - \bar{u}_i \bar{u}_j$. The extra terms δ_i^{mom} and δ^{div} are the numerical deviations that differentiate the *a priori* tests from the *a posteriori* tests. Consequently, the actual values are dependent on the specific choice of the filter and the numerical scheme.

The inconsistency between *a priori* and *a posteriori* results is not unique to data-driven SGS model development; it has also been observed in conventional models [21]. In practice, this inconsistency can be mitigated through appropriate choices in the filtering operator and the use of explicit filtering—both aimed at aligning the modeling assumptions used during training with those encountered during LES deployment. For instance, selecting an appropriate filter type can significantly improve *a posteriori* performance. As demonstrated in Ref. [21], combinations such as the mixed model with a Gaussian filter and the Smagorinsky model with a spectral cutoff filter are considered consistent and effective. Moreover, increasing the ratio between the filter width and the grid spacing (i.e., using a larger filter-grid ratio) helps reduce numerical error in LES, as it increases the relative contribution of the subfilter-scale stresses [22,23]. For nonlinear numerical schemes or nonuniform grid spacing, the construction of discrete filters must follow specific design rules to minimize commutation errors [24–26]. In this context, explicit filtering—where an additional filter

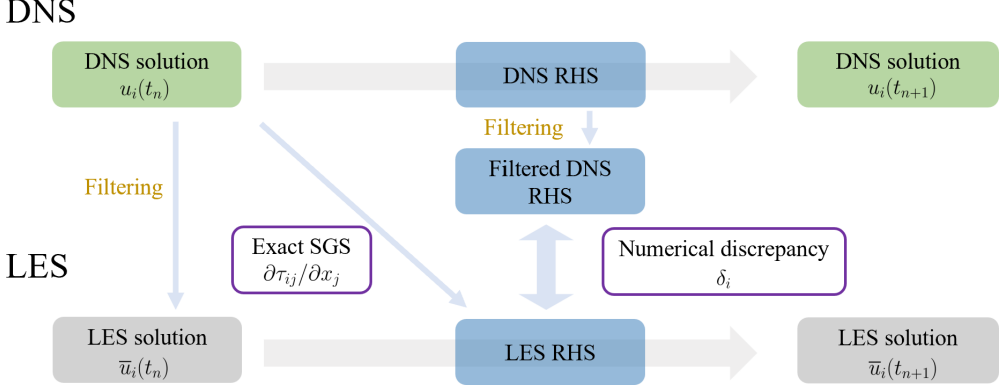
is applied to the nonlinear terms in the Navier-Stokes equations—has been shown to enhance *a posteriori* model performance [27–29].

Several strategies have been proposed to mitigate the issue of inconsistency, including the use of commutative filters, increasing the filter size, and applying explicit filtering techniques. While these approaches have achieved partial success in improving the consistency between *a priori* and *a posteriori* modeling, they do not fully eliminate the numerical discrepancies. These discrepancies are highly dependent on the specific numerical scheme employed, making them difficult to characterize or correct *a priori*. The challenge is further compounded by the strong coupling of various error sources during the evolution of the governing equations, which leads to the accumulation of numerical and modeling errors over time [20]. This issue is particularly severe in data-driven modeling, where nonlinear models are highly sensitive to deviations from the training data distribution. As the simulation evolves, accumulated errors can drive the solution far from the region of state space for which the model was trained, leading to instability and poor predictive performance [30].

To mitigate the issue of accumulating errors, early efforts based on optimal estimation theory aimed to approximate an “ideal” LES using functions of the resolved LES fields [31]. In this context, the ideal LES corresponds to filtered high-resolution (e.g., DNS) data, and the goal of optimal formulations is to reproduce the key statistics of such filtered fields. However, substantial relative errors have been observed across a broad range of spectral scales. Capturing high-wave-number behavior accurately requires access to global information and high-order multipoint correlation data—resources that are typically unavailable during simulations—and the approach has been largely restricted to homogeneous, isotropic turbulence [32,33]. An alternative approach is to directly model the numerical discrepancy in the LES governing Eqs. (2) during the data-driven training phase [34–36]. While this strategy explicitly targets the numerical deviation, models trained in this way often suffer from numerical instability in *a posteriori* tests, again due to the accumulation of error over time [34]. More recent online learning approaches, such as reinforcement learning, have demonstrated improved numerical stability; however, they require access to nonlocal flow information and remain highly case specific, limiting their general applicability [35,36].

Building on previous efforts to improve model consistency, we examine the restrictions that should be imposed on data-driven SGS model development. In practice, SGS models typically rely on locally resolved flow quantities to facilitate generalization across different geometries. While incorporating nonlocal information has shown promise—for example, in dynamic procedures for traditional models [37], and through convolutional neural networks [35] or recurrent neural networks [38] in data-driven frameworks—such approaches often introduce additional challenges related to spatial grid structure or temporal schemes. To maintain flexibility and avoid these complications, we focus exclusively on local input features, eliminating dependencies on grid structure or time history. Simultaneously, we enforce Galilean invariance, ensuring that the model is invariant under translations, rotations, and uniform motions. This constraint guides our model formulation, which is constructed from scalar and tensor invariants of the local velocity gradient, following the classical framework proposed by Pope [39] and later used in complex nonlinear models [40]. In the context of data-driven modeling, imposing physical constraints such as Galilean invariance enhances generalizability and interpretability [19]. Moreover, incorporating a richer set of tensor invariants helps compensate for the lack of a local equilibrium assumption and enables more expressive model forms [41]. Based on these considerations, we explore two representative classes of SGS models in this study: an eddy-viscosity model and a complex nonlinear model.

In this work, we investigate the consistency between *a priori* and *a posteriori* tests in data-driven SGS modeling. Section II describes the DNS-aided LES framework and the procedure for generating training data for the data-driven models. In Sec. III, we analyze how numerical deviations influence both *a priori* training accuracy and *a posteriori* simulation performance. The generalizability of these findings is further discussed in Sec. IV. Finally, the main conclusions are summarized in Sec. V.


 FIG. 1. A sketch of the DNS-aided LES evolving from time step t_n to time step t_{n+1} .

II. SIMULATION DETAILS AND DATA-DRIVEN METHODS

A. Direct numerical simulation aided large-eddy simulation

To accurately quantify the numerical deviation in Eq. (2), we employ a DNS-aided LES framework, in which filtered DNSs and LESs are run side by side to directly evaluate the discrepancies between them [20]. This setup is illustrated schematically in Fig. 1, which shows the evolution from time step t_n to t_{n+1} . At each time step, the DNS flow field is filtered to generate the exact SGS stress divergence $\partial \tau_{ij} / \partial x_j$. By comparing the right-hand side terms of the filtered DNS and the LES, highlighted as blue boxes in the figure, we can isolate the numerical deviation term δ_i . With this deviation accounted for, the LES can be evolved such that its solution remains consistent with that of the filtered DNS. All simulations are performed using an in-house code that employs a staggered-grid, second-order finite-difference scheme in space, and advances the solution in time using a third-order Runge-Kutta method. The DNS solves the discrete incompressible Navier-Stokes equations, with incompressibility enforced via the projection method.

In the present DNS-aided LES framework, both the DNS and the LES are advanced in time using the same time integration scheme and time step, for simplicity. All filtered quantities in Eq. (2) are computed directly from the DNS solution. Notably, after filtering, the resulting fields, including the SGS stress tensor τ_{ij} , remain defined on the DNS grid. To be used in the LES, these filtered quantities are interpolated and downsampled to the LES grid, depending on the resolution and numerical scheme. This interpolation process can be interpreted as an additional filtering operation imposed by the underlying grid structure. In this study, linear interpolation is employed on a staggered grid. Derivative operators are applied after the interpolation to the LES grid. Therefore, the created training dataset aligns with the LES simulations of the *a posteriori* tests, where the derivatives are always computed on the LES grid. Conversely, DNS quantities can also be filtered and interpolated after the derivatives are taken, but this is not adopted in this study. In either case, when the filtering and derivative operators do not commute, the LES equations deviate from the filtered DNS equations. For this study, we neglect the commutation error in the time derivative term. For simplicity, due to the existence of the projection step in both the DNS and the LES, we regard any numerical inconsistency in the continuity equation, denoted δ^{div} , as directly absorbed during the projection step. We therefore focus solely on the momentum equation and drop the superscript “mom” from δ^{mom} without loss of clarity. The numerical deviation δ_i is a combined discrepancy in both the momentum equation and the continuity equation. The LES equation with the numerical deviation can be written as

$$\frac{\partial \bar{u}_i}{\partial t} + \frac{\partial \bar{u}_i \bar{u}_j}{\partial x_j} = -\frac{1}{\rho} \frac{\partial \bar{p}}{\partial x_i} + \nu \frac{\partial^2 \bar{u}_i}{\partial x_j \partial x_j} - \frac{\partial \tau_{ij}}{\partial x_j} + \delta_i, \quad \frac{\partial \bar{u}_i}{\partial x_i} = 0. \quad (3)$$

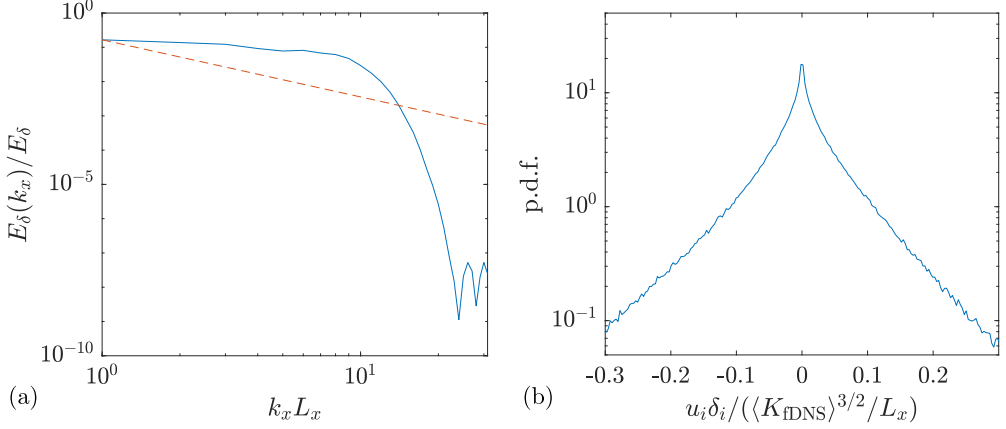


FIG. 2. (a) Spectrum of the numerical deviation $u_i \delta_i$ (solid line). Dashed line indicates the $-5/3$ law. (b) The p.d.f. of the energy transfer for the numerical deviation $u_i \delta_i$ for filter width $\sigma / \Delta_{\text{LES}} = 2.0$.

For further analysis, we can also derive the budget equation of the resolved kinetic energy $K_r = \bar{u}_i \bar{u}_i / 2$ to be

$$\frac{\partial K_r}{\partial t} = \underbrace{-\frac{\partial K_r \bar{u}_j}{\partial x_j}}_{\mathcal{D}_{\text{resolved}}} - \underbrace{\frac{1}{\rho} \frac{\partial \bar{u}_i \bar{p}}{\partial x_i}}_{\mathcal{D}_{\text{pressure}}} + \underbrace{\nu \frac{\partial^2 K_r}{\partial x_j \partial x_j}}_{\mathcal{D}_{\text{viscous}}} - \underbrace{\frac{\partial \bar{u}_i \tau_{ij}}{\partial x_j}}_{\mathcal{D}_{\text{SGS}}} - \underbrace{\nu \frac{\partial \bar{u}_i}{\partial x_j} \frac{\partial \bar{u}_i}{\partial x_j}}_{\varepsilon_{\text{viscous}}} + \underbrace{\tau_{ij} \frac{\partial \bar{u}_i}{\partial x_j}}_{\varepsilon_{\text{SGS}}} + \underbrace{\bar{u}_i \delta_i^{\text{mom}}}_{\varepsilon_{\text{numerical}}}. \quad (4)$$

The right-hand-side budget terms consist of the convection due to the resolved velocity $\mathcal{D}_{\text{resolved}}$, the pressure transport term $\mathcal{D}_{\text{pressure}}$, the SGS transport term $\mathcal{D}_{\text{SGS}}$, the viscous dissipation term $\varepsilon_{\text{viscous}}$, the SGS dissipation term ε_{SGS} , and the numerical dissipation term $\varepsilon_{\text{numerical}}$.

The choice of filtering operator has a significant impact on the inconsistency between DNS and LES solutions, as demonstrated in [42]. One way to reduce the relative importance of the numerical deviation is to employ a larger filter size. However, increasing the filter size also diminishes the resolved energy in the smaller turbulence scales and substantially raises the uncertainty in modeling the unresolved scales. This trade-off and its implications are discussed in detail in Sec. IV B. In this study, we focus on the standard Gaussian filter, where the filtering operator $\bar{(\cdot)}$ is defined as

$$\bar{\phi}(\mathbf{x}, t) = \int G(\mathbf{r}) \phi(\mathbf{x} - \mathbf{r}, t) d\mathbf{r}, \quad G(\mathbf{r}) = \frac{1}{\sqrt{2\pi}\sigma} e^{-|\mathbf{r}|^2/2\sigma^2}. \quad (5)$$

We utilize multiple filter widths, $\sigma / \Delta_{\text{LES}}$ of 0.5, 1.0, 2.0, 4.0, where $\sigma / \Delta_{\text{LES}} = 2.0$ is the baseline filter width.

A DNS-aided LES is carried out for a forced homogeneous isotropic flow at Reynolds number based on a Taylor microscale of $\text{Re}_\lambda = \sqrt{2k/3}\lambda/\nu = 180$, where k is the turbulent kinetic energy, ε is the dissipation rate, and $\lambda = \sqrt{10k/\varepsilon}$. The Kolmogorov length scale for this flow is $\eta = \sqrt[4]{\nu^3/\varepsilon}$, and the Kolmogorov time scale is given by $\tau_\eta = \sqrt{\nu/\varepsilon}$. A linear forcing is used to provide stationary turbulence as proposed by [43]. We use a periodic box of size $(2\pi L_x)^3$ with a resolution of 256^3 for the DNS, where L_x is the characteristic length for the domain, which corresponds to $\Delta_{\text{DNS}} \approx 5.1\eta$. The time step is given by $\Delta t_{\text{DNS}} = 0.013\tau_\eta$. The LES simulations are performed using several different resolutions smaller than that of the DNS, which is explored in Sec. IV B with baseline resolution of 64^3 .

In Fig. 2, we show the spectrum for the numerical deviation δ_i and the probability density function (p.d.f.) of its nondimensional energy transfer $u_i \delta_i$. Compared to the $-5/3$ scaling in the energy spectrum of homogeneous isotropic turbulence, the numerical deviation acts mostly on the intermediate scale. In addition, we observe in Fig. 2(b) that the distribution of the energy transfer

$u_i \delta_i$ is symmetric around 0. Therefore, the numerical deviation term is intended for modifying the spatial distribution of the energy transfer, instead of changing the global energy transfer amount.

In the DNS-aided LES framework, all necessary quantities for training a data-driven SGS model are computed. For instance, the LES velocity field is obtained by filtering the DNS velocity field, as illustrated in Fig. 1, and is used to calculate local invariants as described in Sec. II B. Additionally, quantities derived from the DNS data that require modeling—specifically, the exact SGS stress τ_{ij} and the numerical deviation δ_i —are collected as training targets. Once the SGS model is trained on these data, it is applied in an actual LES simulation for *a posteriori* testing. Ideally, these LES simulations should closely replicate the DNS-aided LES results and accurately predict the filtered DNS fields. The generalizability of the trained models is further examined in Sec. IV B.

B. Training details

We use the DNS-aided LES data for training data-driven SGS models. To achieve generalizability, we use the model form that follows the physical constraint of Galilean invariance. The model involves the strain-rate tensor $\mathbf{S}$ and the rotation-rate tensor $\mathbf{R}$ directly computed from the LES flow field, where

$$S_{ij} = \frac{1}{2} \left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} \right), \quad R_{ij} = \frac{1}{2} \left(\frac{\partial \bar{u}_i}{\partial x_j} - \frac{\partial \bar{u}_j}{\partial x_i} \right). \quad (6)$$

The six scalar invariants are computed from $\mathbf{S}$ and $\mathbf{R}$ as

$$\begin{aligned} I_1 &= \text{tr}(\mathbf{S}^2), & I_2 &= \text{tr}(\mathbf{R}^2), & I_3 &= \text{tr}(\mathbf{S}^3), \\ I_4 &= \text{tr}(\mathbf{S}\mathbf{R}^2), & I_5 &= \text{tr}(\mathbf{S}^2\mathbf{R}^2), & I_6 &= \text{tr}(\mathbf{S}^2\mathbf{R}^2\mathbf{S}\mathbf{R}). \end{aligned} \quad (7)$$

At the same time, the four tensor invariants are also linear or nonlinear forms of $\mathbf{S}$ and $\mathbf{R}$ [39],

$$\begin{aligned} \mathbf{m}_1 &= \mathbf{S}, & \mathbf{m}_2 &= \mathbf{S}^2, \\ \mathbf{m}_3 &= \mathbf{R}^2, & \mathbf{m}_4 &= \mathbf{S}\mathbf{R} - \mathbf{R}\mathbf{S}, \end{aligned} \quad (8)$$

In the present work, two major model forms are used: the eddy viscosity model

$$\boldsymbol{\tau}_{\text{ev}} = -2\nu_T(I_1, \dots, I_6)\mathbf{m}_1, \quad (9)$$

and the complex nonlinear model

$$\begin{aligned} \boldsymbol{\tau}_{\text{cn}} &= -2\nu_{T,1}(I_1, \dots, I_6)\mathbf{m}_1 - 2\nu_{T,2}(I_1, \dots, I_6)\mathbf{m}_2 \\ &\quad - 2\nu_{T,3}(I_1, \dots, I_6)\mathbf{m}_3 - 2\nu_{T,4}(I_1, \dots, I_6)\mathbf{m}_4. \end{aligned} \quad (10)$$

The eddy-viscosity model is dissipative when ν_T is positive definite and the corresponding dissipation is given by ε_{SGS} . The complex nonlinear model includes additional terms to incorporate the anisotropic behavior of the SGS terms.

To capture the nonlinear behavior between the SGS stress tensor $\boldsymbol{\tau}$ and the scalar invariants, an artificial neural network is trained to find the coefficients in front of the tensor invariants for each model, i.e., the eddy viscosity ν_T in the eddy-viscosity model and $\nu_{T,1}, \nu_{T,2}, \nu_{T,3}, \nu_{T,4}$ in the complex nonlinear model. Both the input and the output features are locally nondimensionalized by the characteristic length scale and the strain-rate scale:

$$l_{\text{char}} = \sqrt{\Delta x^2 + \Delta y^2 + \Delta z^2}, \quad S_{\text{char}} = \frac{1}{3} \sqrt{2(\|\mathbf{S}\|^2 + \|\mathbf{R}\|^2)}, \quad (11)$$

where the grid size (thus the LES grid truncation scale) in each direction is denoted by $\Delta x, \Delta y, \Delta z$. The 1/3 scaling of the characteristic strain-rate scale is intended to scale in the input features to a similar order of magnitude, which would be essential for a stabilized training [44]. The nondimensionalized quantities are indicated by the superscript $*$.

We use a fully connected feedforward neural network with three hidden layers, which have 20 neurons in each layer with the sigmoid function as the activation function. Due to the nondimensionalization, the input features $I_1^* + I_2^* = \text{const}$, and thus the input layer does not have a bias parameter. The hyperparameters are determined through trial and error, and they provide efficient training without the risk of overfitting.

The dataset used for model development is obtained from the DNS-aided LES described in Sec. II A. A portion of the dataset is allocated for training and validation, with a 70:30 split between the two, totaling 1.1×10^5 points. A separate subset is reserved for testing, which includes both *a priori* and *a posteriori* evaluations, as discussed in Sec. III. Model training is conducted using the Adam optimization algorithm [45], with an initial learning rate of 0.01. The learning rate is adaptively reduced if the validation error stagnates. Training is terminated when the loss function plateaus, which typically occurs after approximately 100 epochs. The training history of the models is shown in the Appendix.

The loss function is given by the $\mathcal{L}_2$ loss of the local dissipation rate. For comparison, both a model excluding the numerical deviation and a model including the numerical deviation are considered during training; the model formulation and the model implementation are identical between the two models. For a model excluding numerical deviation, the loss function is given by

$$L = \mathcal{L}_2(\varepsilon^*, \varepsilon_{\text{model}}^*) = \sum (S_{ij}^* \tau_{ij}^* - S_{ij}^* \tau_{\text{model},ij}^*)^2, \quad (12)$$

where $\varepsilon^* = \varepsilon_{\text{SGS}}^* = -S_{ij}^* \tau_{ij}^*$ is commonly used in literature for model evaluation [13,46]. For a model including the numerical deviation, the loss function is given by

$$L = \mathcal{L}_2(\varepsilon^*, \varepsilon_{\text{model}}^*) = \sum \left[\left(u_i^* \frac{\partial \tau_{ij}^*}{\partial x_j^*} - u_i^* \delta_i^* \right) - u_i^* \frac{\partial \tau_{\text{model},ij}^*}{\partial x_j^*} \right]^2, \quad (13)$$

$$\text{where } \varepsilon^* = -\mathcal{D}_{\text{SGS}}^* + \varepsilon_{\text{SGS}}^* + \varepsilon_{\text{numerical}}^* = u_i^* \frac{\partial \tau_{ij}^*}{\partial x_j^*} - u_i^* \delta_i^*. \quad (14)$$

Here, the subscript “model” indicates a modeled quantity for both the eddy-viscosity and the complex nonlinear models. The dissipation rate including the numerical deviation is turned into the vector form since the numerical deviation is naturally a vector in Eq. (3). The superscript * indicates that the quantities are nondimensionalized using characteristic scales given by Eq. (11). Note that the loss function is designed to minimize the error in the local dissipation rate using the data-driven methods instead of the domain-averaged dissipation rate in the non-data-driven methods. As observed in the previous literature [40], a constant eddy viscosity does not necessarily provide the optimal local dissipation rate, and the local values have strong scattering.

III. IMPACT OF THE NUMERICAL DEVIATION

In this section, we examine the impact of the numerical deviation in both *a priori* and *a posteriori* tests. The training performance is evaluated and compared between models excluding and including the numerical deviation. All the results are based on the baseline filter width $\sigma / \Delta_{\text{LES}} = 2.0$.

A. *A priori* training performance

The trained models are first evaluated in the *a priori* tests, using the filtered DNS data. For each evaluation, we compare the trained models excluding the numerical deviation and including the numerical deviation. In order to visualize the turbulent structures of different scales, the nondimensionalization is based on the global dissipation of the filtered DNS $\langle \varepsilon_{\text{fDNS}} \rangle$, where $\langle \cdot \rangle$ indicates the spatial averaging.

Figure 3 presents the *a priori* evaluation of the trained eddy-viscosity-type SGS models, comparing the versions that exclude and include the numerical deviation. To qualitatively assess model

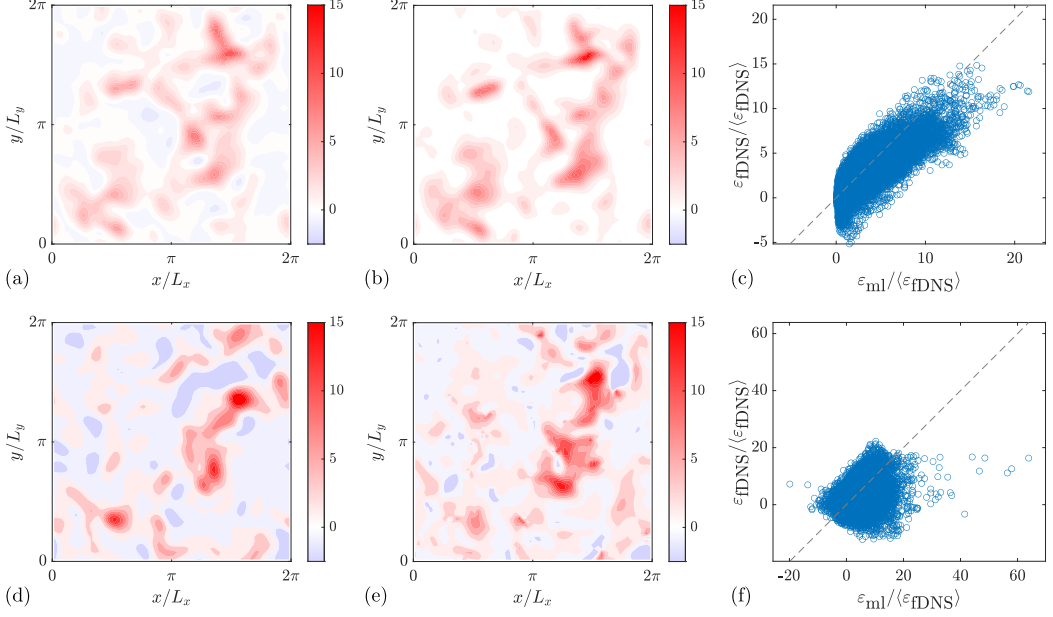


FIG. 3. Instantaneous (a), (d) target dissipation rate $\varepsilon/\langle\varepsilon_{\text{fDNS}}\rangle$ and (b), (e) modeled dissipation rate $\varepsilon_{\text{ml}}/\langle\varepsilon_{\text{fDNS}}\rangle$ on the same x - y plane and (c), (f) scatter plot showing the correlation between ε_{ml} and $\varepsilon_{\text{fDNS}}$ for eddy-viscosity neural-network SGS models (a)–(c) excluding and (d)–(f) including the numerical deviation for baseline filter width $\sigma/\Delta_{\text{LES}} = 2.0$. The dashed line in (c) and (f) indicates identity.

accuracy across different scales, we compare the target dissipation rate with that predicted by the trained models as given by the loss function. Note that the two models have different loss functions, and the target dissipation rate is not identical between the two. The correlation between the modeled and target dissipation rates is shown in the form of scatter plots. Quantitative evaluation is provided in Table I, which reports both the Pearson correlation coefficient and the relative error rate. It is important to note that, due to differences in characteristic scales, this *a priori* evaluation differs from the metrics used during model training.

TABLE I. *A priori* evaluation of the root-mean-square errors (rRMSE) and Pearson correlation coefficient (Corr.) of the dissipation rate $\varepsilon/\langle\varepsilon_{\text{fDNS}}\rangle$. EV indicates eddy-viscosity model, and CN indicates complex nonlinear model. The models excluding and including the numerical deviation are indicated by noND and ND, respectively.

		EV-noND	EV-ND	CN-noND	CN-ND
$\sigma/\Delta_{\text{LES}} = 0.5$	rRMSE	0.7094	1.0149	0.7711	0.9511
	Corr.	0.6366	0.2456	0.6829	0.3860
$\sigma/\Delta_{\text{LES}} = 1.0$	rRMSE	0.5021	0.9538	0.7417	0.6701
	Corr.	0.8322	0.4026	0.8079	0.7777
$\sigma/\Delta_{\text{LES}} = 2.0$ (Baseline)	rRMSE	0.4947	1.0524	0.7624	0.7543
	Corr.	0.8764	0.4181	0.7384	0.7952
$\sigma/\Delta_{\text{LES}} = 4.0$	rRMSE	0.3850	0.7929	1.4408	0.7449
	Corr.	0.8760	0.5092	0.4555	0.7370

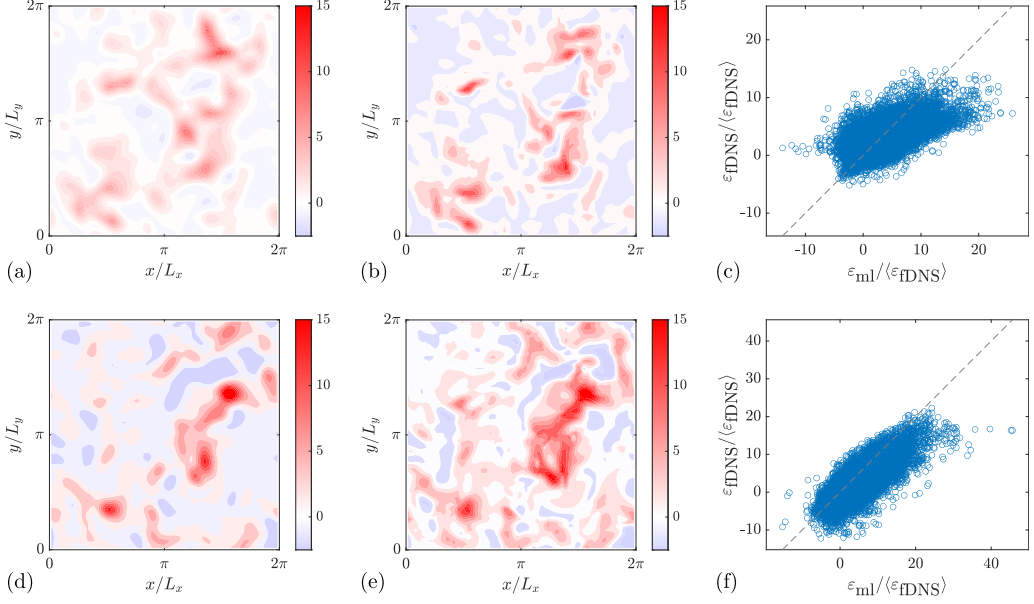


FIG. 4. Instantaneous (a), (d) target dissipation rate $\varepsilon/\langle\varepsilon_{\text{DNS}}\rangle$ and (b), (e) modeled dissipation rate $\varepsilon_{\text{ml}}/\langle\varepsilon_{\text{DNS}}\rangle$ on the same x - y plane and (c), (f) scatter plot showing the correlation between ε_{ml} and ε_{DNS} for complex nonlinear neural-network SGS models (a)–(c) excluding and (d)–(f) including the numerical deviation for baseline filter width $\sigma/\Delta_{\text{LES}} = 2.0$. The dashed line in (c) and (f) indicates identity.

When the numerical deviation is excluded, the target dissipation rate shown in Fig. 3(a) is predominantly positive, reflecting a locally dissipative behavior. As shown in Fig. 3(b), the eddy-viscosity model captures the large-scale features of the dissipation field reasonably well, but fails to accurately reproduce the finer-scale structures. The scatter plot in Fig. 3(c) indicates a strong correlation between the predicted and true dissipation values, supported by a high Pearson correlation coefficient.

In comparison, when the numerical deviation is included, the target dissipation rate in Fig. 3(d) is no longer strictly locally dissipative, and exhibits stronger spatial variability. As shown in Fig. 3(e), the eddy-viscosity neural-network model captures a broader range of flow structures, but fails to accurately reproduce both the large-scale and the small-scale features. The scatter plot in Fig. 3(f) confirms this, showing weaker correlation between the predicted and true values. The model occasionally exhibits large deviations, resulting in higher error in the *a priori* evaluation. This outcome is expected, as including the numerical deviation introduces additional complexity, making it more difficult to extract accurate models when relying solely on local quantities.

In Fig. 4, the visualization of the complex nonlinear model's performance, given by Eq. (10), reveals markedly different behavior compared to the eddy-viscosity model. When the numerical deviation is excluded, as shown in Figs. 4(a) and 4(b), the model prediction exhibits rich small-scale structures, benefiting from the increased expressiveness of the model form. However, the large-scale structures are not as well captured as in the eddy-viscosity case, as they are overshadowed by the amplified small-scale activity. Furthermore, the model is no longer locally dissipative, a consequence of the more flexible nonlinear formulation. The impact of this increased complexity is discussed further in Sec. III B. Despite these differences, the model still achieves a high correlation between predicted and true dissipation values, as seen in Fig. 4(c), although local dissipation is often overestimated.

When the numerical deviation is included [Figs. 4(d) and 4(e)], the model more accurately captures both large-scale and small-scale structures. Notably, the correlation between predictions

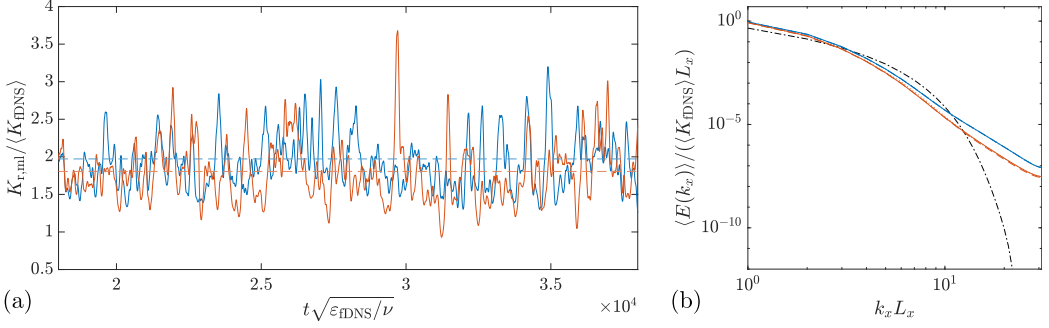


FIG. 5. (a) Time evolution of the instantaneous (solid) and average (dashed) kinetic energy for the eddy-viscosity SGS model excluding (blue) and including (red) the numerical deviation. (b) Energy spectra of the *a posteriori* analysis for the eddy-viscosity SGS model excluding (blue) and including (red) the numerical deviation using time steps of Δt_{DNS} (solid), $2\Delta t_{\text{DNS}}$ (dashed), and $5\Delta t_{\text{DNS}}$ (dotted). Energy spectrum of the filtered DNS is given by the black dot-dashed line.

and true values, shown in Fig. 4(f), is significantly improved compared to the eddy-viscosity model. This suggests that the complex nonlinear model, when trained with the numerical deviation, can better reconcile multiple scales of turbulence.

B. *A posteriori* results

In the data-driven methods, the *a posteriori* tests often fail to align with the *a priori* evaluations. To improve this consistency, we incorporate the numerical deviation in model training. We assess the resulting consistency by examining the energy spectra from *a posteriori* simulations using models trained with and without the numerical deviation. For the *a posteriori* tests, the trained neural-network models are implemented within the second-order finite-difference LES code described in Sec. II A. The filtered DNS flow fields serve as the initial conditions for the LES. To ensure a fair comparison, the simulations are run using the same time step as was used to generate the training data. However, it is important to note that this choice is not a constraint of the modeling framework; the flexibility in time stepping is discussed further in Sec. IV B. The simulations are run until the kinetic energy reaches a statistically converged state as demonstrated in Figs. 5(a) and 6(a). As a

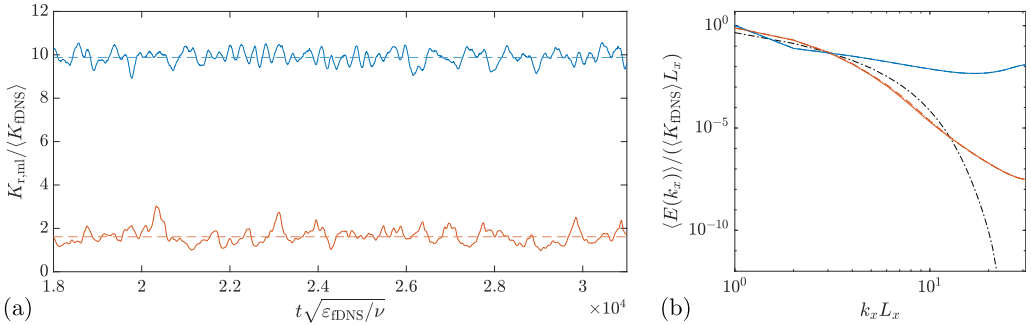


FIG. 6. (a) Time evolution of the instantaneous (solid) and average (dashed) kinetic energy for the complex nonlinear SGS model excluding (blue) and including (red) the numerical deviation. (b) Energy spectra of the *a posteriori* analysis for the complex nonlinear SGS model excluding (blue) and including (red) the numerical deviation using time steps of Δt_{DNS} (solid), $2\Delta t_{\text{DNS}}$ (dashed), and $5\Delta t_{\text{DNS}}$ (dotted). Energy spectrum of the filtered DNS is given by the black dot-dashed line.

reference, the kinetic energy of the filtered DNS fluctuates $\pm 50\%$ around unity value though not shown here, consistent with the fluctuation seen in the literature [43]. No additional stabilization strategies, such as backscatter clipping, are applied during the *a posteriori* simulations.

In Fig. 5, we evaluate the *a posteriori* performance of the eddy-viscosity neural-network SGS models trained both with and without the numerical deviation. Figure 5(a) shows the time evolution of the kinetic energy from these simulations. Both simulations remain numerically stable and evolve toward statistically steady states with higher kinetic energy than the filtered DNS. Notably, the model trained with the numerical deviation yields slightly lower kinetic energy, indicating more dissipative behavior. Though both models are trained for accurate kinetic energy transfer, excluding the numerical deviation will neglect the possible impact of the numerical deviation on the energy spectrum. This leads to different *a posteriori* behavior of the kinetic energy transfer. The interaction between the model and the LES solver will further exaggerate such differences.

This observation is further supported by the energy spectra in Fig. 5(b). Both models show reasonable agreement with the filtered DNS across a range of scales, although the small-scale turbulence is generally overpredicted. This overprediction is consistent with the *a priori* results, where neither model accurately captures the fine-scale structure of the local dissipation rate. Interestingly, both models tend to underpredict intermediate-scale turbulence, though the precise mechanism remains unclear.

Comparing the two, the inclusion of the numerical deviation leads to a modest suppression of small-scale energy, which contributes to slightly better spectral alignment with the filtered DNS. Although the model trained with the numerical deviation performs worse in terms of *a priori* correlation and error metrics, it achieves comparable—if not slightly improved—*a posteriori* performance. This result highlights a key insight: minimizing the *a priori* error in the SGS dissipation rate does not necessarily ensure better *a posteriori* performance. Numerical effects, particularly the numerical deviation, play a critical role and must be accounted for during model development.

In Fig. 6, we present the *a posteriori* evaluation of the complex nonlinear SGS models trained with and without the inclusion of numerical deviation. The use of a richer model form with additional terms significantly enhances the representation of small-scale turbulence in both *a priori* and *a posteriori* tests. However, this added complexity also increases the risk of numerical instability, primarily due to the higher likelihood of local energy backscattering. To maintain a fair evaluation, no backscatter clipping or artificial stabilization is applied.

As shown in Fig. 6(a), the model trained without accounting for numerical deviation results in an *a posteriori* simulation where the kinetic energy grows well beyond the expected level, even after convergence. The corresponding energy spectrum in Fig. 6(b) confirms that the small-scale turbulence is severely overpredicted, and the inaccuracy extends across all resolved scales, distorting the entire spectral shape.

In contrast, the model trained with the numerical deviation yields an *a posteriori* spectrum that closely aligns with the filtered DNS. While the small-scale turbulence remains slightly overpredicted, the magnitude of the deviation is within an acceptable range, and the overall spectrum is much more physical. Furthermore, this model demonstrates improved numerical stability throughout the simulation.

These results emphasize the crucial role of including numerical deviation in training, especially for high-capacity models. Accounting for the numerical discrepancy helps constrain the model in regimes where nonlinear effects dominate, leading to improved physical realism and stable LES performance.

In the *a priori* tests, the neural-network models that exclude the numerical deviation perform well for both the eddy-viscosity and the complex nonlinear model forms. However, these promising *a priori* results do not translate to consistent *a posteriori* performance. In particular, the complex nonlinear model trained without accounting for numerical deviation fails to yield meaningful LES simulations. Despite careful model design and nondimensionalization grounded in physical principles, the combination of model complexity and strong neural-network nonlinearity could bring

the *a posteriori* solutions beyond available training data. Therefore, a data-driven model lacking generalizability could become numerically unstable, especially for a complex model.

For homogeneous isotropic turbulence, a simpler eddy-viscosity-type model is often sufficient. Yet, when modeling more complex flows—where anisotropy, nonequilibrium effects, or additional physical mechanisms are present—more expressive models with nonlinear terms become necessary to capture the essential small-scale dynamics. Unfortunately, the added model capacity also amplifies the inconsistency between the *a priori* and *a posteriori* tests, making data-driven SGS model development for such flows particularly challenging.

To mitigate this inconsistency, incorporating the numerical deviation during training proves to be an effective strategy. This approach helps align the behavior of the model in *a posteriori* simulations with its *a priori* evaluations, thereby enhancing the physical relevance of *a priori* tests and improving the reliability of the training process.

IV. DISCUSSION

A. The relationship to the non-data-driven models

Numerical deviation is always present, regardless of whether data-driven techniques are used for model development. However, data-driven models tend to exhibit a more pronounced inconsistency between the training process and their actual application in LES simulations. To better understand the impact of data-driven methods on *a posteriori* performance, we carry out tests that bridge the gap between data-driven and traditional (non-data-driven) models. Specifically, instead of using neural networks to predict the coefficients of tensor invariants in the eddy-viscosity and complex nonlinear models, we directly “train” constant coefficients from the data. This approach is equivalent to applying linear regression on the local tensor invariants $\mathbf{m}_1$, $\mathbf{m}_2$, $\mathbf{m}_3$, $\mathbf{m}_4$ and allows us to isolate the effects of model nonlinearity from the effects of data-driven optimization. Note that the eddy-viscosity model with a constant coefficient is similar to the traditional Smagorinsky model,

$$\tau_{\text{Smag}} = -2(C_s \Delta)^2 |\mathbf{S}| \mathbf{S} = -2(C_s \Delta)^2 |\mathbf{S}| \mathbf{m}_1, \quad (15)$$

where $C_s = 0.16$ is the typical Smagorinsky coefficient [46], and Δ is the resolved length scale. Similarly, the complex nonlinear model with constant coefficients is similar to the mixed model, which combines the tensor diffusivity model with the eddy-viscosity term [47]. For this model, the SGS stress term is given by

$$\tau_{\text{mixed},ij} = \Delta^2 \frac{\partial u_i}{\partial x_k} \frac{\partial u_j}{\partial x_k} - 2(C_{s'} \Delta)^2 |\mathbf{S}| S_{ij}, \quad (16)$$

or equivalently

$$\tau_{\text{mixed}} = -2(C_{s'} \Delta)^2 |\mathbf{S}| \mathbf{m}_1 + \Delta^2 (\mathbf{m}_2 - \mathbf{m}_3 - \mathbf{m}_4), \quad (17)$$

where $C_{s'}$ is the modified Smagorinsky coefficient. Note that the second part of the mixed model is similar to the tensor diffusivity model, which is derived from a Taylor series of the filter operator [48]. Thus, the mixed model can be seen as the tensor diffusivity model with additional dissipation.

The training procedure follows the method outlined in Sec. II B, with the exception that a constant is used instead of the neural network, resulting in significantly faster training. The resulting constant coefficients are summarized in Table II. For reference, the eddy viscosity of the classical Smagorinsky model is $\nu_T^* = 1.2288$. Among the eddy-viscosity-type models, the data-driven model trained without including the numerical deviation (EV-noND-const) produces a smaller coefficient than expected, whereas the model trained with the numerical deviation (EV-ND-const) yields a larger coefficient. The *a posteriori* results for both models, shown in Fig. 7, confirm that incorporating the numerical deviation results in a modest suppression of small-scale turbulence—consistent with the trends observed in the neural-network-based models.

On the other hand, the complex nonlinear model bears a resemblance to the classical mixed model, which combines an eddy-diffusivity term with the tensor diffusivity model. The tensor dif-

TABLE II. Fitted coefficients for the eddy-viscosity (EV) model [Eq. (9)] and the complex nonlinear (CN) model [Eq. (10)], along with the theoretical prediction of the Smagorinsky model and the tensor diffusivity model.

	EV-noND -const	EV-ND -const	EV-theory	CN-noND -const	CN-ND -const	TD-theory
$\nu_{T,1}^*$	0.9780	1.3323	1.2288	0.8512	1.2519	
$\nu_{T,2}^*$				-0.3046	-0.4042	-0.6667
$\nu_{T,3}^*$				0.8658	0.9194	0.6667
$\nu_{T,4}^*$				4.8161	0.7701	0.6667

fusivity term for the mixed model, which is derived from the Taylor series of the filter operator [48], is not necessarily numerically stable due to lack of dissipation [29]. In the classical formulation, the theoretical coefficients for the tensor diffusivity terms are given by $\nu_{T,2}^* = -\nu_{T,3}^* = -\nu_{T,4}^* = -2/3$, which we adopt as a reference for evaluating the correspondig coefficients in the data-driven model listed in Table II.

We observe that the trained complex nonlinear model with constant coefficients (CN-noND-const), when excluding the numerical deviation, deviates significantly from the tensor diffusivity component of the mixed model—particularly in the value of $\nu_{T,4}^*$. In contrast, the model trained with the numerical deviation included (CN-ND-const) yields coefficients that align more closely with the theoretical values of the tensor diffusivity model, where $\nu_{T,4}^*$ is of the same order of magnitude as $\nu_{T,2}^*$ and $\nu_{T,3}^*$. This discrepancy arises because $\mathbf{m}_4$, when evaluated on a staggered grid, appears nearly nondissipative if the SGS dissipation rate is computed solely as $Si j^* \tau_{ij}^*$. As a result, the model trained without accounting for numerical effects tends to overemphasize the contribution of $\mathbf{m}_4$.

However, $\mathbf{m}_4$ still affects the evolution of local kinetic energy through the SGS transport term and numerical deviation. A more appropriate evaluation function in this context is $u_i^* (\partial \tau_{ij}^* / \partial x_j - \delta_i^*)$, which captures both physical and numerical contributions. As shown in Fig. 7, the *a posteriori* test for CN-noND-const is not available due to numerical instability. Even augmenting the model with an eddy-viscosity term, as in the traditional mixed model, does not stabilize it—likely due to the

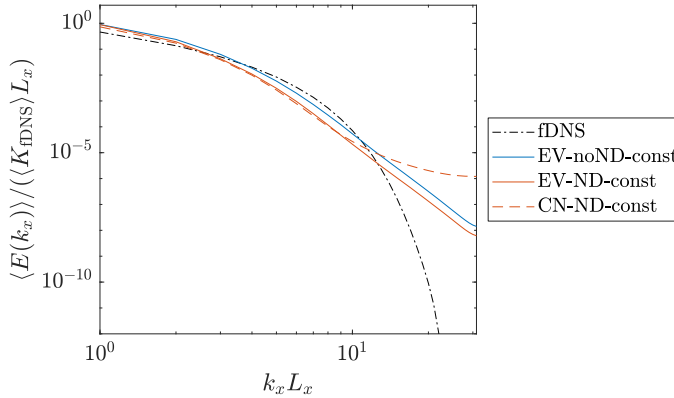


FIG. 7. Energy spectra of the *a posteriori* analysis of the data-driven constant coefficient eddy-viscosity (solid) and complex nonlinear (dashed) models, excluding (blue) and including (red) numerical deviation compared to the filtered DNS (black dot-dashed). The *a posteriori* analysis for CN-noND-const is not available due to numerical instability.

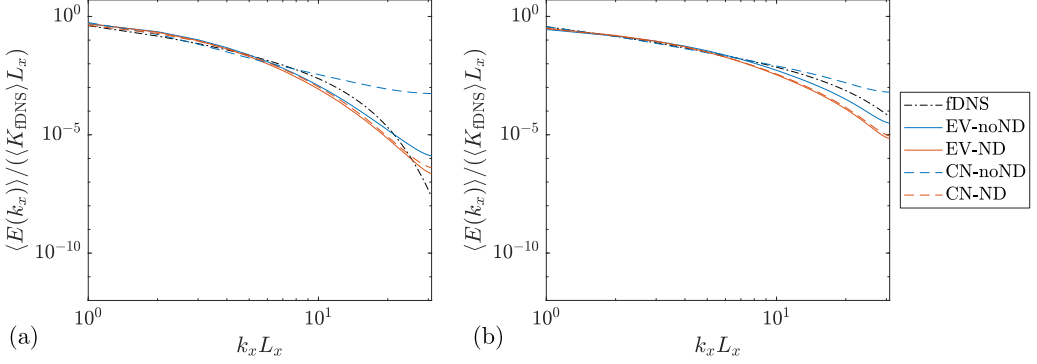


FIG. 8. Energy spectra of the *a posteriori* analysis of the data-driven constant coefficient eddy-viscosity (solid) and complex nonlinear (dashed) models, excluding (blue) and including (red) numerical deviation for filter widths of (a) $\sigma/\Delta_{\text{LES}} = 1.0$ and (b) $\sigma/\Delta_{\text{LES}} = 0.5$, compared to filtered DNS (black dot-dashed).

excessive value of $v_{T,4}^*$. In contrast, the CN-ND-const model is numerically stable and yields a good match with the filtered DNS energy spectrum.

Furthermore, comparing the neural-network-based CN-ND model in Fig. 6(b) with the CN-ND-const model in Fig. 7 shows that the neural network improves small-scale predictions. The additional flexibility from the nonlinear parametrization helps mitigate small-scale overprediction that can arise from complex model structures.

This comparison between the data-driven models and the non-data-driven models indicates the difficulties the data-driven models may have in realizing the physical constraints learned through theoretical derivation, and these difficulties lead to issues such as overpredicting specific nonlinear terms. During the *a priori* tests, the impact of the overprediction is not easy to recognize. However, as the simulation evolves, the error in the solution accumulates and deviates from the filtered DNS. The *a posteriori* tests may have significantly distorted small-scale turbulence and thus could be numerically unstable. The data-driven techniques further magnify the issue because the model form and the neural-network architecture lead to more nonlinearity in the model. The coupling nonlinearity complicates the inconsistency issue and makes it more prominent and more challenging in data-driven modeling. In our experiments, including the numerical deviation eases the challenge of such inconsistency between the *a priori* tests and the *a posteriori* tests in data-driven modeling.

B. Model generalizability

The impact of numerical deviation is further examined by generalizing the results to other numerical setups, beginning with the effect of filter size. Increasing the filter size relative to the grid spacing is known to reduce the relative contribution of numerical errors [23], and it has also been shown to improve the accuracy of data-driven models [42]. We vary the filter width $\sigma/\Delta_{\text{LES}}$ across four values: 0.5, 1.0, 2.0, and 4.0.

For a smaller filter width, $\sigma/\Delta_{\text{LES}} = 1.0$, the *a priori* evaluation (Table I) shows slight degradation compared to the baseline case of $\sigma/\Delta_{\text{LES}} = 2.0$. The corresponding *a posteriori* results, shown in Fig. 8(a), indicate a familiar overprediction of small-scale turbulence, particularly in the complex nonlinear model. As seen previously, models trained without the numerical deviation again exhibit a mismatch between the *a priori* and *a posteriori* behaviors. Incorporating the numerical deviation during training suppresses this overprediction and yields an energy spectrum that is comparably closer to that of the filtered DNS. Surprisingly, the smaller filter width leads to less small-scale overprediction compared to the baseline. Although numerical error may be more pronounced in the *a priori* setting for smaller σ , the interaction between numerical and subfilter-scale terms reduces its impact in the *a posteriori* tests.

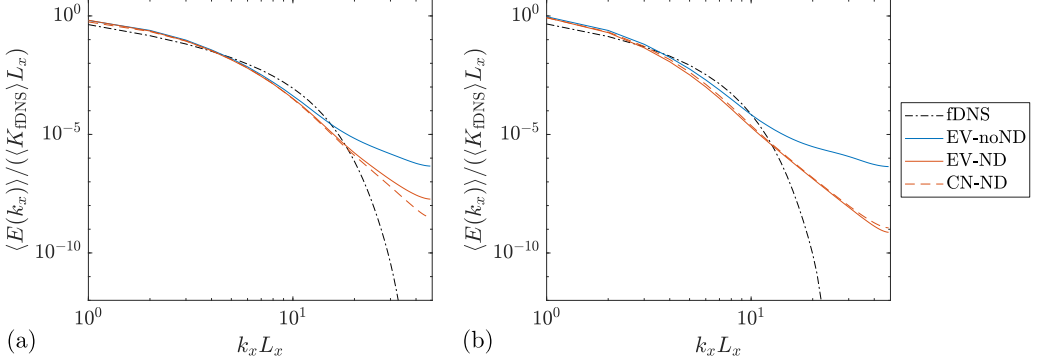


FIG. 9. Energy spectra of *a posteriori* analysis of the data-driven constant coefficient eddy-viscosity (solid) and complex nonlinear (dashed) models, excluding (blue) and including (red) numerical deviation for filter widths of (a) $\sigma/\Delta_{\text{LES}} = 2.0$ and (b) $\sigma/\Delta_{\text{DNS}} = 8.0$, compared to the filtered DNS (black dot-dashed).

Similarly, for an even smaller filter width $\sigma/\Delta_{\text{LES}} = 0.5$, Fig. 8(b) shows that including the numerical deviation continues to suppress small-scale overprediction. However, the overprediction is so minimal that both models—with and without the numerical deviation—provide a similarly accurate fit to the filtered DNS.

At the other extreme, for a larger filter width $\sigma/\Delta_{\text{LES}} = 4.0$, the *a priori* accuracy remains comparable to the baseline (Table I). However, the *a posteriori* simulations fail to sustain turbulence, suggesting that excessive filtering can overdamp the resolved motions, compromising model viability. This highlights the trade-off in choosing filter width: it must be large enough to reduce numerical contamination, yet small enough to preserve essential flow structures. In this context, including the numerical deviation during training aligns the *a priori* and *a posteriori* evaluations more consistently, with the greatest benefit seen for intermediate filter widths.

Next, we demonstrate the generalizability of the trained models with respect to different time step sizes. Since the model development is entirely local and time independent, the implementation is not restricted to the time step used in the DNS-aided LES. In Figs. 5 and 6, the dashed and dotted lines represent simulations using time steps of $2\Delta t$ and $5\Delta t$, respectively. These results are nearly identical to those obtained with the original time step Δt , confirming that the trained models are robust to changes in temporal resolution.

Finally, we examine the impact of grid resolution on model performance. The same comparative analysis is conducted for LES at a finer resolution of 96^3 , and the results are compared with the baseline cases discussed in Sec. III. Two sets of tests are performed: one maintains a constant filter-to-grid ratio, $\sigma/\Delta_{\text{LES}} = 2.0$, while the other maintains a constant absolute filter width, $\sigma/\Delta_{\text{DNS}} = 8.0$. As shown in Fig. 9, the *a posteriori* results at higher resolution reaffirm that including the numerical deviation leads to more consistent and stable LES behavior. In contrast, the complex nonlinear model trained without accounting for the numerical deviation becomes numerically unstable. This highlights that although complex nonlinear models offer greater flexibility in capturing subgrid-scale dynamics, they are also more prone to numerical instability when the numerical deviation is not included during training. Furthermore, grid refinement can destabilize an SGS model that was stable on a coarser grid. For both the eddy-viscosity and the complex nonlinear models, the influence of the numerical deviation becomes more pronounced at finer resolutions, as a greater range of small-scale turbulence is resolved in the flow field.

V. CONCLUSION

In this work, we investigate the inconsistency between the *a priori* and *a posteriori* tests in data-driven SGS modeling. While data-driven approaches often demonstrate improved accuracy

during training (*a priori* tests), their performance in actual LES simulations (*a posteriori* tests) does not always surpass that of traditional models. Furthermore, data-driven models frequently suffer from numerical instability. This inconsistency arises because the governing equations underlying the training data—typically filtered DNS—and those governing the LES simulations differ due to commutation errors and resolution disparities. This discrepancy introduces a numerical deviation that can be locally significant compared to the SGS stress term [42].

To assess the impact of this numerical deviation on model development, we trained neural network SGS models both including and excluding the numerical deviation, considering two model forms: eddy-viscosity and complex nonlinear models. Although the *a priori* performance of models including the numerical deviation may appear inferior to those excluding it, their *a posteriori* kinetic energy spectra are generally comparable or even superior. The nonlinear model captures richer small-scale turbulence behavior but may obscure large-scale structures. Models excluding the numerical deviation tend to significantly overpredict small-scale turbulence, resulting in severely distorted energy spectra. Incorporating the numerical deviation suppresses this overprediction, stabilizing the simulation and yielding energy spectra that align well with filtered DNS data. This effect is particularly critical for complex nonlinear models, which are essential for capturing anisotropic SGS behaviors.

The importance of accounting for numerical deviation is heightened in data-driven approaches due to their inherent nonlinearity. Our comparison of the mixed model and the data-driven nonlinear model with constant coefficients confirms that excluding the numerical deviation leads to exaggerated small-scale turbulence predictions when the model form allows it. Including the numerical deviation results in more physically consistent predictions. Although strategies such as increasing the filter-grid ratio and applying explicit filtering techniques can mitigate numerical errors, they do not fully resolve them. The model inconsistency problem is more challenging in data-driven models because nonlinear interactions cause errors to accumulate, causing the LES solution to drift away from the data-driven model’s valid state space. Thus, incorporating the numerical deviation is crucial for achieving consistency between *a priori* and *a posteriori* tests in data-driven SGS modeling.

We also extend the benefits of including the numerical deviation to different filter widths and LES resolutions. Unlike in non-data-driven methods, increasing the filter width does not necessarily reduce the numerical error term here. While including the numerical deviation generally improves model consistency, the model form must remain physically sound, and the filter width of the filtered DNS should be chosen at an intermediate scale to ensure satisfactory *a posteriori* performance. Importantly, including the numerical deviation does not impose additional constraints on spatial or temporal discretization schemes. The trained model requires only local information and can be applied flexibly across different time steps and grid configurations.

ACKNOWLEDGMENTS

The authors would like to thank Dr. Anthony Leonard for the discussion on the tensor diffusivity model. This work is supported by the Center for Turbulence Research at Stanford University (US) and the Office of Naval Research (US) under Grant No. N00014-23-1-2729.

DATA AVAILABILITY

The data that support the findings of this article are not publicly available upon publication because it is not technically feasible and/or the cost of preparing, depositing, and hosting the data would be prohibitive within the terms of this research project. The data are available from the authors upon reasonable request.

APPENDIX: TRAINING HISTORY

The training history of the examined models is shown in Fig. 10, where the convergence of the loss function of the training dataset and that of the validation dataset indicates that the

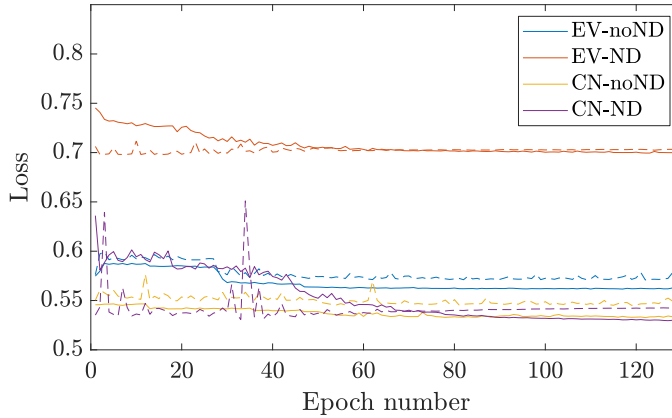


FIG. 10. The training history of the loss function values for the four models. Solid and dashed lines denote training and validation losses, respectively. EV and CN represent eddy-viscosity and complex nonlinear models; noND and ND indicate exclusion and inclusion of numerical deviation.

training is finished without overfitting. All four models have shown a converged performance after approximately 100 epochs.

-
- [1] K. A. Goc, O. Lehmkuhl, G. I. Park, S. T. Bose, and P. Moin, Large eddy simulation of aircraft at affordable cost: A milestone in computational fluid dynamics, [Flow](#) **1**, E14 (2021).
 - [2] T. Okaze, H. Kikumoto, H. Ono, M. Imano, N. Ikegaya, T. Hasama, K. Nakao, T. Kishida, Y. Tabata, K. Nakajima *et al.*, Large-eddy simulation of flow around an isolated building: A step-by-step analysis of influencing factors on turbulent statistics, [Build. Sci.](#) **202**, 108021 (2021).
 - [3] E. Zahn and E. Bou-Zeid, Setting up a large-eddy simulation to focus on the atmospheric surface layer, [Boundary Layer Meteorol.](#) **190**, 12 (2024).
 - [4] X. Huang and Jiaqi J. L. Li, Characteristics of the meandering effect in a stratified wake, [Phys. Rev. Fluids](#) **10**, 014602 (2025).
 - [5] S. Xiao, C. Peng, and D. Yang, Large-eddy simulation of bubble plume in stratified crossflow, [Phys. Rev. Fluids](#) **6**, 044613 (2021).
 - [6] M. Pino Martín, U. Piomelli, and G. V. Candler, Subgrid-scale models for compressible large-eddy simulations, [Theor. Comput. Fluid Dyn.](#) **13**, 361 (2000).
 - [7] R. Stoll and F. Porté-Agel, Dynamic subgrid-scale models for momentum and scalar fluxes in large-eddy simulations of neutrally stratified atmospheric boundary layers over heterogeneous terrain, [Water Resour. Res.](#) **42**, W01409 (2006).
 - [8] R. Agrawal, M. P. Whitmore, K. P. Griffin, S. T. Bose, and P. Moin, Non-boussinesq subgrid-scale model with dynamic tensorial coefficients, [Phys. Rev. Fluids](#) **7**, 074602 (2022).
 - [9] U. Unnikrishnan, H. Huo, X. Wang, and V. Yang, Subgrid scale modeling considerations for large eddy simulation of supercritical turbulent mixing and combustion, [Phys. Fluids](#) **33**, 075112 (2021).
 - [10] R. D. Moser, S. W. Haering, and G. R. Yalla, Statistical properties of subgrid-scale turbulence models, [Annu. Rev. Fluid Mech.](#) **53**, 255 (2021).
 - [11] M. Gamahara and Y. Hattori, Searching for turbulence models by artificial neural network, [Phys. Rev. Fluids](#) **2**, 054604 (2017).
 - [12] J. Park and H. Choi, Toward neural-network-based large eddy simulation: Application to turbulent channel flow, [J. Fluid Mech.](#) **914**, A16 (2021).

- [13] Z. Zhou, G. He, S. Wang, and G. Jin, Subgrid-scale model for large-eddy simulation of isotropic turbulent flows using an artificial neural network, *Comput. Fluids* **195**, 104319 (2019).
- [14] M. Kang, Y. Jeon, and D. You, Neural-network-based mixed subgrid-scale model for turbulent flow, *J. Fluid Mech.* **962**, A38 (2023).
- [15] S. Maejima and S. Kawai, Coarse-grid large-eddy simulation by unsupervised-learning-based sub-grid scale modeling, in *AIAA SciTech 2024 Forum* (2024), p. 1361.
- [16] J. F. MacArt, J. Sirignano, and J. B. Freund, Embedded training of neural-network subgrid-scale turbulence models, *Phys. Rev. Fluids* **6**, 050502 (2021).
- [17] Q. Meng, Z. Jiang, and J. Wang, Artificial neural network-based subgrid-scale models for les of compressible turbulent channel flow, *Theor. Appl. Mech. Lett.* **13**, 100399 (2023).
- [18] M. Kim, J. Park, and H. Choi, Large eddy simulation of flow over a circular cylinder with a neural-network-based subgrid-scale model, *J. Fluid Mech.* **984**, A6 (2024).
- [19] A. Prakash, K. E. Jansen, and J. A. Evans, Invariant data-driven subgrid stress modeling in the strain-rate eigenframe for large eddy simulation, *Comput. Methods Appl. Mech. Eng.* **399**, 115457 (2022).
- [20] H. J. Bae and A. Lozano-Duran, Numerical and modeling error assessment of large-eddy simulation using direct-numerical-simulation-aided large-eddy simulation, [arXiv:2208.02354](https://arxiv.org/abs/2208.02354).
- [21] U. Piomelli, P. Moin, and J. H. Ferziger, Model consistency in large eddy simulation of turbulent channel flows, *Phys. Fluids* **31**, 1884 (1988).
- [22] S. Ghosal, An analysis of numerical errors in large-eddy simulations of turbulence, *J. Comput. Phys.* **125**, 187 (1996).
- [23] F. K. Chow and P. Moin, A further study of numerical errors in large-eddy simulations, *J. Comput. Phys.* **184**, 366 (2003).
- [24] O. V. Vasilyev, T. S. Lund, and P. Moin, A general class of commutative filters for LES in complex geometries, *J. Comput. Phys.* **146**, 82 (1998).
- [25] A. L. Marsden, O. V. Vasilyev, and P. Moin, Construction of commutative filters for LES on unstructured meshes, *J. Comput. Phys.* **175**, 584 (2002).
- [26] S. T. Bose, P. Moin, and D. You, Grid-independent large-eddy simulation using explicit filtering, *Phys. Fluids* **22**, 105103 (2010).
- [27] T. S. Lund, The use of explicit filters in large eddy simulation, *Comput. Math. Appl.* **46**, 603 (2003).
- [28] D. Carati, G. S. Winckelmans, and H. Jeanmart, On the modeling of the subgrid-scale and filtered-scale stress tensors in large-eddy simulation, *J. Fluid Mech.* **441**, 119 (2001).
- [29] G. S. Winckelmans, A. A. Wray, O. V. Vasilyev, and H. Jeanmart, Explicit-filtering large-eddy simulation using the tensor-diffusivity model supplemented by a dynamic Smagorinsky term, *Phys. Fluids* **13**, 1385 (2001).
- [30] F. Sarghini, G. De Felice, and S. Santini, Neural networks based subgrid scale modeling in large eddy simulations, *Comput. Fluids* **32**, 97 (2003).
- [31] J. A. Langford and R. D. Moser, Optimal LES formulations for isotropic turbulence, *J. Fluid Mech.* **398**, 321 (1999).
- [32] J. A. Langford and R. D. Moser, Optimal large-eddy simulation results for isotropic turbulence, *J. Fluid Mech.* **521**, 273 (2004).
- [33] R. D. Moser, N. P. Malaya, H. Chang, P. S. Zandonade, P. Vedula, A. Bhattacharya, and A. Haselbacher, Theoretically based optimal large-eddy simulation, *Phys. Fluids* **21**, 105104 (2009).
- [34] A. Beck, D. Flad, and C.-D. Munz, Deep neural networks for data-driven les closure models, *J. Comput. Phys.* **398**, 108910 (2019).
- [35] M. Kurz, P. Offenhäuser, and A. Beck, Deep reinforcement learning for turbulence modeling in large eddy simulations, *Int. J. Heat Fluid Flow* **99**, 109094 (2023).
- [36] A. Beck and M. Kurz, Toward discretization-consistent closure schemes for large eddy simulation using reinforcement learning, *Phys. Fluids* **35**, 125122 (2023).
- [37] C. Meneveau and J. Katz, Scale-invariance and turbulence models for large-eddy simulation, *Annu. Rev. Fluid Mech.* **32**, 1 (2000).
- [38] M. Kurz and A. Beck, A machine learning framework for LES closure terms, *Electron. Trans. Numer. Anal.* **56**, 117 (2022).

- [39] S. B. Pope, A more general effective-viscosity hypothesis, *J. Fluid Mech.* **72**, 331 (1975).
- [40] T. S. Lund and E. A. Novikov, Parameterization of subgrid-scale stress by the velocity gradient tensor, *Center for Turbulence Research Annual Research Briefs* (Center for Turbulence Research, Stanford, CA, 1993).
- [41] K. Inagaki and H. Kobayashi, Analysis of anisotropic subgrid-scale stress for coarse large-eddy simulation, *Phys. Rev. Fluids* **8**, 104603 (2023).
- [42] X. Huang, S. Leung, M. Whitmore, A. Elnahas, and H. Bae, in *Proceedings of the Summer Program* (Center for Turbulence Research, Stanford, CA, 2024) p. 1.
- [43] T. S. Lundgren, Linearly forced isotropic turbulence, *Center for Turbulence Research Annual Research Briefs 2003* (Center for Turbulence Research, Stanford, CA, 2003).
- [44] Y. LeCun, L. Bottou, G. B. Orr, and K.-R. Müller, *Neural networks: Tricks of the Trade* (Springer, New York, 2002), pp. 9–50.
- [45] D. P. Kingma and J. Ba, Adam: A method for stochastic optimization, [arXiv:1412.6980](https://arxiv.org/abs/1412.6980).
- [46] D. K. Lilly, A proposed modification of the germano subgrid-scale closure method, *Phys. Fluids A* **4**, 633 (1992).
- [47] Y. Zang, R. L. Street, and J. R. Koseff, A dynamic mixed subgrid-scale model and its application to turbulent recirculating flows, *Phys. Fluids A* **5**, 3186 (1993).
- [48] S. Stolz, N. A. Adams, and L. Kleiser, An approximate deconvolution model for large-eddy simulation with application to incompressible wall-bounded flows, *Phys. Fluids* **13**, 997 (2001).