DEEP REINFORCEMENT LEARNING FOR LARGE-EDDY SIMULATION OF WALL-BOUNDED TURBULENCE

Junhyuk Kim
Department of Mechanical Engineering
Yonsei University
50 Yonsei-ro, Seodaemun-gu, Seoul 03722, Korea
junhyuk6@yonsei.ac.kr

Hyojin Kim
Department of Mechanical Engineering
Yonsei University
50 Yonsei-ro, Seodaemun-gu, Seoul 03722, Korea
gywls072@yonsei.ac.kr

Jiyeon Kim
School of Mathematics and Computing
Yonsei University
50 Yonsei-ro, Seodaemun-gu, Seoul 03722, Korea
jykim3994@yonsei.ac.kr

Changhoon Lee
Department of Mechanical Engineering
School of Mathematics and Computing
Yonsei University
50 Yonsei-ro, Seodaemun-gu, Seoul 03722, Korea
clee@yonsei.ac.kr

ABSTRACT

Large-eddy simulation (LES) is actively carried out for many scientific and engineering problems. Recently supervised learning approaches using high-fidelity flow data in training process are applied to LES modeling, however, there are significant disadvantages. For example, the trained model is not effective in actual LES, and such high-fidelity data is usually not available in real-world problems. To overcome these, we employed deep reinforcement learning (DRL) where actual LES and training of subgrid-scale (SGS) model are carried out simultaneously using only target statistics as given information. We additionally applied physical constraints such as reflectional invariance and wall boundary conditions on DRL for reducing the training cost. Through this, we are challenging to find a reliable SGS model for three-dimensional LES of wall-bounded turbulence. The DRL model that produces the local SGS stress based on the local velocity gradient were trained, as a result, we found that in various training environments DRL could discover models that make mean velocity and mean Reynolds shear stress of actual LES be consistent with the target, while the conventional SGS models usually mispredict them. We conclude that DRL would be a effective tool for turbulence modeling in practical problems.

INTRODUCTION

For many problems involving turbulent flows, precise simulation is important, but in real-world problems the cost of direct numerical simulation (DNS) that represents the smallest scale is still too high despite of the rapid development of computational hardware and numerical algorithm. For the purpose of improving the trade-off between cost and accuracy of simulation, several types of turbulence models have been developed.

As one of them, we focus on the large-eddy simulation (LES) where instantaneous vortical motions are represented and the subgrid-scale (SGS) stress should be modeled by the resolved scale information. Many algebraic SGS models based on statistical and physical analysis have been proposed. As a pioneering approach, Smagorinsky (1963) suggested a model that the anisotropic part of SGS stress tensor τ_{ij} is represented by the resolved strain-rate tensor $\bar{S}_{ij}$ and eddy viscosity ν_t. For an effort to apply LES to complex flows, some modifications such as dynamic Smagorinsky model (DSM) (Germano *et al.*, 1991; Lilly, 1992) and Vreman model (Vreman, 2004) were suggested. While those existing SGS models continue to evolve and are successfully applied to practical problems, there is certainly room for significant improvement in terms of accuracy and cost. For example, coarse-grained LES with DSM is quite inaccurate in canonical flows.

In recent years machine learning models using filtered DNS data for training are challenging to the conventional SGS models. In turbulent channel flow Gamahara & Hattori (2017) and Park & Choi (2021) developed neural network (NN)-based model with the classical supervised learning framework, which is composed of four steps; data collection, training, *a priori* test and *a posteriori* test. However, due to the significant mismatch between *a priori* test and *a posteriori* test, it is difficult to develop a reliable model. It is most likely because the model was trained only at the equilibrium state and the effect (reaction) of modeled SGS stress in actual LES was not taken into account in the training process. Another major drawback is that the classical supervised learning requires high-fidelity data for training. In real-world problems, such data is usually not available, and only some partial or statistical data can be collected. As an alternative algorithm, we are expecting that deep reinforcement learning (DRL) would help overcome the disadvantages of classical supervised learning. Very recently, Novati *et al.* (2021) proposed the DRL-LES framework for developing a SGS model in forced homogeneous isotropic turbulence, which could predict energy spectrum quite well. But the learning is not successful in the case that actions are fully collocated in all grids without interpolation. To overcome the previous limitations, we proposed a physics-constrained DRL algorithm that perfectly guarantees reflectional equivariance and boundary conditions of the SGS model. Through this, we extend DRL to LES modeling in wall-bounded turbulence, which remains as a challenging problem due to its inhomogeneity.

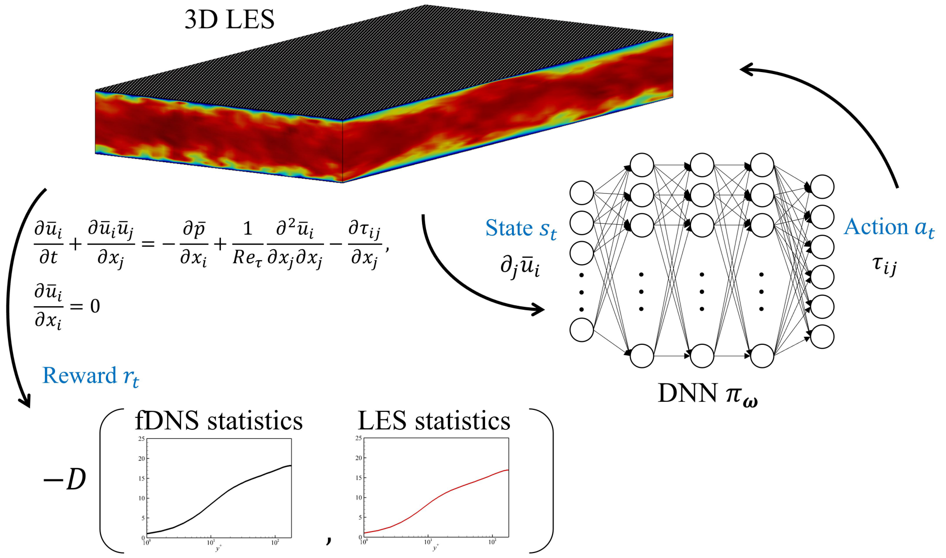

Figure 1. Illustration of present work. A DRL-LES framework for developing a subgrid-scale (SGS) model in wall-bounded turbulence is proposed. Running of three-dimensional (3D) LES and learning of deep neural network (DNN) that produces the SGS stress from the resolved velocity gradient, are carried out simultaneously. Target statistics for training are the mean viscous stress and the mean Reynolds shear stress, used for reward calculation.

METHODOLOGY

The DRL-LES framework consists of a LES solver and a reinforcement learning algorithm. LES of turbulent channel flow with a NN-based SGS model and its training using the data generated from the LES are carried out simultaneously, as shown in figure 1.

Numerical method

The governing equations of LES are filtered incompressible Navier–Stokes equations with a spatial filter operation $(\bar{\cdot})$,

$$\frac{\partial \bar{u}_i}{\partial x_i} = 0, \frac{\partial \bar{u}_i}{\partial t} + \frac{\partial(\bar{u}_i \bar{u}_j)}{\partial x_j} = -\frac{\partial \bar{p}}{\partial x_i} + \frac{1}{Re_\tau}\frac{\partial^2 \bar{u}_i}{\partial x_j \partial x_j} - \frac{\partial \tau_{ij}}{\partial x_j} \quad (1)$$

The equation is non-dimensionalized by the channel half width (δ) and the friction velocity (u_τ). x_1 (x), x_2 (y), and x_3 (z) mean the streamwise, wall-normal, and spanwise directions, and the corresponding velocity components are $u_i (= u, v, w)$, respectively. The simulation parameter is the friction Reynolds number $Re_\tau = \frac{u_\tau \delta}{\nu}$ where ν is the kinematic viscosity. The SGS stress τ_{ij} is modeled by DNN π_ω with trainable parameters ω. The ultimate goal of present work can be defined as below.

$$\text{argmin}_\omega ||S^{\text{target}} - S^{\text{LES}(\pi_\omega)}|| \quad (2)$$

where S is statistics of fDNS and LES. Although the goal of DRL is the same as developing a DNN-based SGS model with high statistical accuracy, its results can be highly dependent on the choice of state and action, the definition of reward, and some important techniques of the algorithm. In the sense that performance of *a posteriori* test is guaranteed if only the training is successful, the construction of the robust learning algorithm is important.

Deep reinforcement learning

DRL is an algorithm that finds the optimal action a_t to receive the maximum reward in a given state s_t of environment. At this time, the target to maximize is the long-term reward ($R_t = \sum_{i=t}^{\infty} \gamma^{i-t} r_i$ with $0 < \gamma < 1$) that the instantaneous reward r_t coming out after the action is accumulated for a time horizon. In our LES SGS modeling problem, the state and action are the resolved flow variables and modeled SGS stresses, respectively, and the reward can be defined as the statistical distance between the target (filtered DNS) and the LES. In the training process, the DRL algorithm optimizes the DNN-based SGS model in the direction of making the statistical accuracy of the LES precise. In this study, one of DRL algorithms, deep deterministic policy gradient (DDPG) is used.

$$\mathbb{E}_{\pi_\omega}\left[\sum_{i=t}^{\infty} \gamma^{i-t} r_i\right] \approx Q_\theta(s_t, a_t) \quad (3)$$

Here, training of Q_θ with deterministic policy is mostly depending on recursive relation of Bellman equation.

Using critic DNN that predicts the long-term reward, we can train the actor DNN π_ω in the direction of increasing the critic value. Its objective function to maximize is as follows :

$$\text{argmax}_\omega \, \mathbb{E}_{s_t}\left[Q_\theta(s_t, \pi_\omega(s_t))\right] \quad (4)$$

Our purpose through DRL is to discover an optimal SGS model π_ω that could make the statistics of LES precise. π_ω is fully connected neural network that produces SGS stress based on the resolved velocity gradient and the local grid size. This local framework can be extended to more complex flow easily. And critic DNN is convolutional network as a surrogate model predicting the future state of flow based on the present state. And the instantaneous reward is defined by two statistical quantities, mean viscous stress and mean Reynolds stress, as follows.

$$r_t(y) = \sum_{i=1}^{2} -c_i \left| S_i^{\text{fDNS}}(y) - S_i^{\text{LES}}(t + \Delta t^{\text{DRL}}, y)\right|^{1/2}, \quad (5)$$

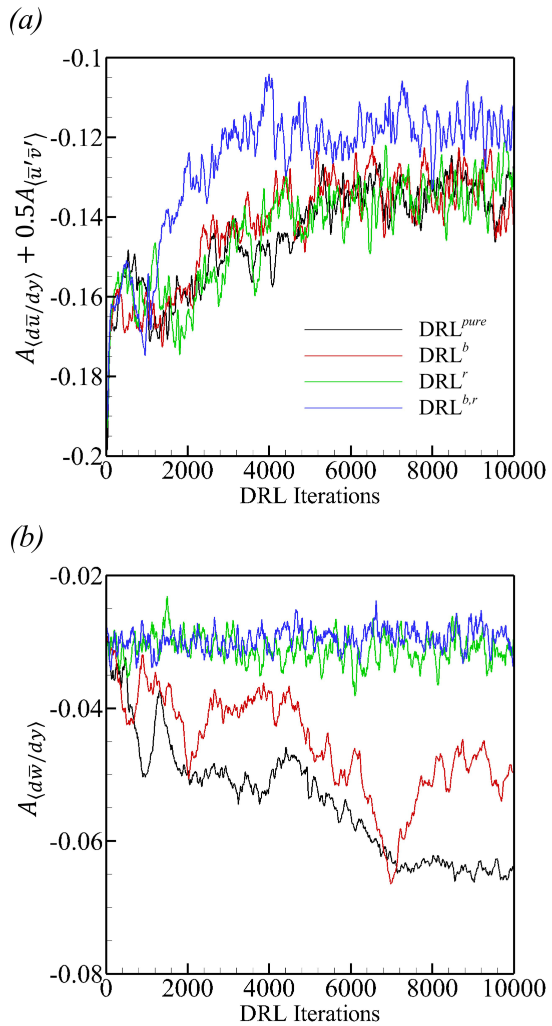

Figure 2. The effect of physical constraints on DRL performance. r and b in the caption denote reflectional invariance and boundary constraints, respectively. (a) is the accuracy of $\langle d\bar{u}/dy\rangle$ and $\langle \bar{u}'\bar{v}'\rangle$ and (b) is the accuracy of $\langle d\bar{w}/dy\rangle$.

where

$$S_1 = \left\langle \frac{1}{Re_\tau}\frac{d\bar{u}}{dy}\right\rangle, \quad S_2 = \langle \bar{u}'\bar{v}'\rangle . \qquad (6)$$

Here, Δt^{DRL} is the period of data collection and learning, and we set it as 30 LES time steps, which corresponds to 5.4 wall time units. And, We carried out DRL for learning steps of 10, 000 – 15, 000 to find the optimal SGS model. And, more details related to DRL algorithms are presented in Kim *et al.* (2022).

RESULTS

DRL was mainly carried out for $Re_\tau = 180$ and filter (grid) size $(\Delta x^+, \Delta z^+) = (70.7, 35.3)$. We applied two kinds of physical constraints, boundary condition and reflectional invariance to DNN and observed their effect on training performance. The statistical accuracy used for the quantification is

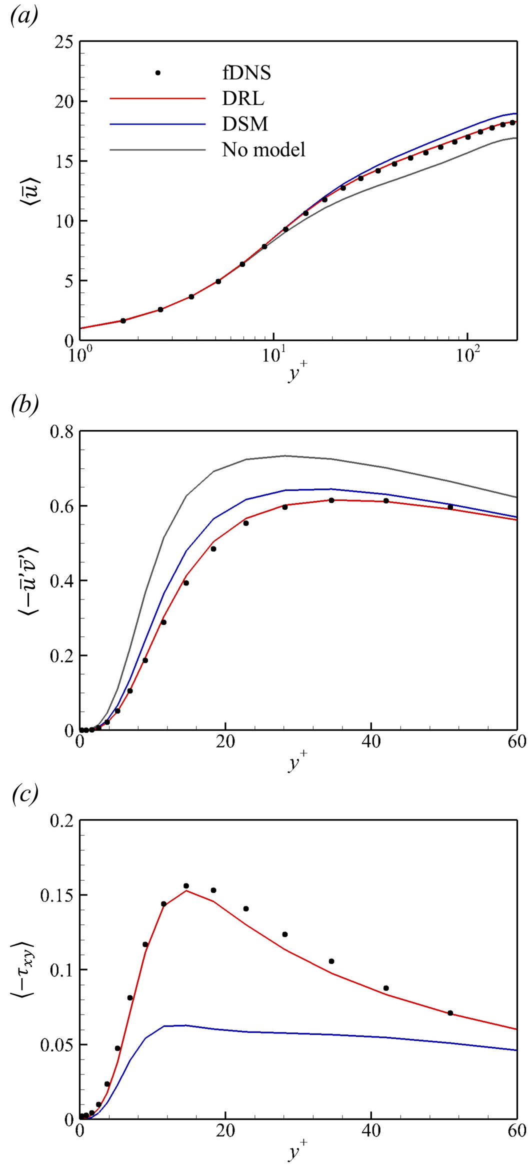

Figure 3. Test for $(\Delta x^+, \Delta z^+) = (70.7, 35.3)$. (a), (b) and (c) are the streamwise mean velocity, the mean Reynolds shear stress and the mean SGS shear stress, respectively.

as follows:

$$A_i = \frac{1}{N_y}\sum_{j=1}^{N_y} -\left|S_i^{\mathrm{fDNS}}(y_j) - S_i^{\mathrm{LES}*}(y_j)\right|^{1/2}. \qquad (7)$$

Here, A_i is the accuracy of statistics. First, the accuracy of mean viscous stress and mean Reynolds stress used in reward is presented in figure 2(a). It is observed that DRL^{b,r} with two physical constraints has better speed and convergence accuracy than DRL^{pure} without physical constraint. Also, as shown in figure 2(b), DRL without reflectional invariance could generate statistically anti-symmetric flows that are unphysical in channel flow. It indicates that applying physical constraints on DRL is useful and necessary.

Next, time-averaged results of actual LES with the trained model are presented in figure 3 with the conventional SGS models including DSM and no model ($\tau_{ij} = 0$). LES with the trained DRL model could predict the target statistics of

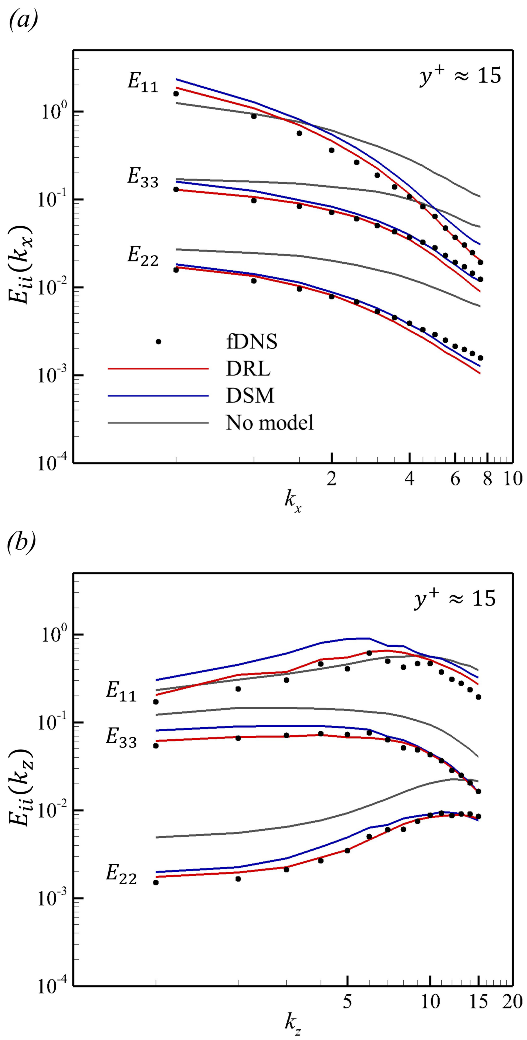

Figure 4. One-dimensional energy spectrum at $y^+ \approx 15$. (*a*) and (*b*) is streamwise and spanwise directions, respectively.

mean velocity and mean Reynolds shear stress almost perfectly, while DSM overpredicts the mean velocity and the magnitude of mean Reynolds shear stress and the No model highly mispredict them. By the total shear stress equation, the DRL model produced mean SGS shear stress $\langle -\tau_{xy} \rangle$ very well compared to DSM. Here, we want to briefly mention the results of classical supervised learning. The supervised learning model with the same input information quite underpredicted the magnitude of $\langle u'v' \rangle$ in the actual LES. In some cases, the actual LES diverged in the model using more input information. This indicates that even when fDNS data are available, there are limitations in developing successful models through the classical supervised learning and that it is important for the development of a successful model to reflect the reaction of the SGS model through the online learning algorithm.

Also, the one-dimensional energy spectrum also supports the results (figure 4). No model significantly overestimated the overall distribution of the energy, while DSM and DRL predict them reasonably. But, it is noticeable that DRL shows better prediction than DSM at low wavenumbers with high energy, although at high wavenumbers some inaccuracies are observed. Actually, (not shown here) LES with the trained DRL model well represents the vortical structures, unlike the No model.

We also present the statistics of SGS dissipation ($\varepsilon_{SGS} = -\tau_{ij}\bar{S}_{ij}/Re_\tau$) and SGS transport ($T_{SGS} = Re_\tau^{-1}\partial(\tau_{ij}\bar{u}_i)/\partial x_j$) in figure 5. We found that to accurately predict the streamwise mean velocity and mean Reynolds shear stress, accurate representation of mean SGS transport is required, but the accuracy of mean SGS dissipation and backscatter are not essential requirements. Accurate prediction of SGS energy transfer might be relevant with the high order statistics or temporal behavior of resolved variables.

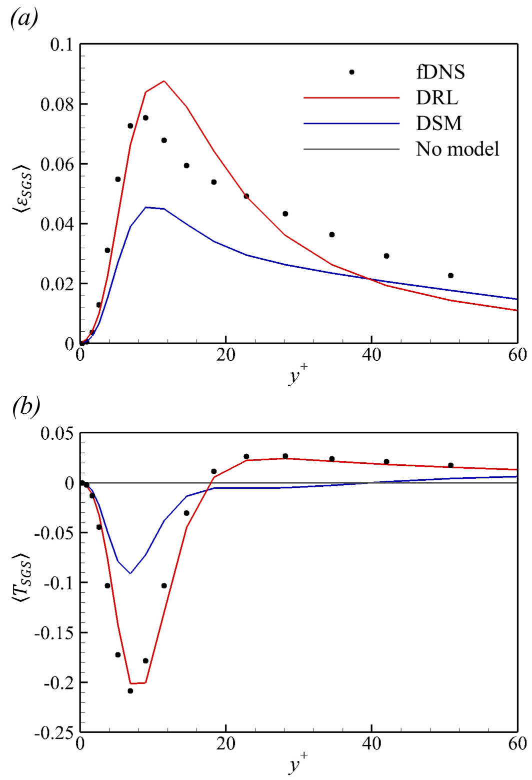

Figure 5. (*a*) and (*b*) are the mean SGS dissipation and SGS transport, respectively.

Finally, we trained the DRL model in a new environment with more coarse grid size $(\Delta x^+, \Delta z^+) = (94.2, 47.1)$. Even though the optimal mean SGS shear stress is two times larger than that of the previous case, DRL could find the optimal SGS model well. Its test results in actual LES are given in figure 6. At this condition, the prediction of DSM highly mismatches with the target statistics, while our DRL model successfully predicts the learning target. It indicates once again that our DRL algorithm could discover a successful model in the actual LES test.

CONCLUSIONS

It is clear that the DRL framework is a promising algorithm for developing the SGS model of LES. And with the increase of the available computational resources, the performance of model will naturally increase. Although we have shown successful applicability only in channel flow, this framework could be extended to various turbulence modeling problems in complex geometries. For the global goal of successful predictions in various complex flows, construction of diverse flow solvers and learning them simultaneously should be carried out. In the near future, we hope to develop the turbulence model that shows practical utility in real-world problems.

Although it was possible to find an optimal model through

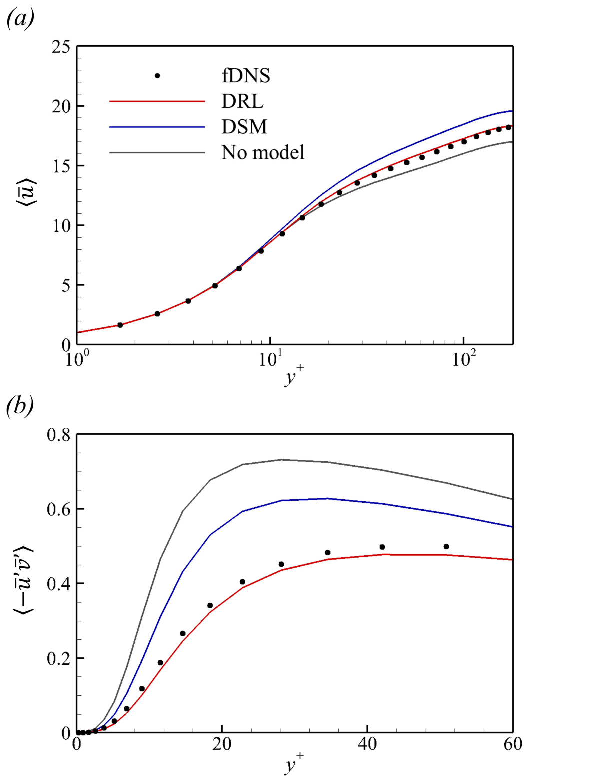

Figure 6. Test for $(\Delta x^+, \Delta z^+) = (94.2, 47.1)$. (*a*) and (*b*) are the streamwise mean velocity and the mean Reynolds shear stresss, respectively.

DRL, the understanding of the working principle of reinforcement learning in relation to turbulence physics, understanding of characteristics of the trained network, and guidance of hyperparameters is poor. In fact, it has been reported that the physical properties inherent in data can be identified through the characteristic analysis of the trained network (Kim & Lee, 2020; Lu *et al.*, 2020), showing its importance. Clearly, we expect that the effort of fundamental understanding would help improve the robustness and generalization of DRL.

Also, in a parallel point of view, a differentiable PDE method (Sirignano *et al.*, 2020; Kochkov *et al.*, 2021) has emerged as a promising algorithm. Although this method should build a differentiable numerical solver and requires huge cost and memory for each learning iteration, it has shown the robust *a posteriori* test results compared to the classical supervised learning. Therefore, a comparative research of online learning algorithms, the DRL and the differentiable PDE method, would be interesting as future work.

ACKNOWLEDGMENTS

This work was supported by a National Research Foundation of Korea (NRF) grant funded by the Korean government (MSIP) (2017R1E1A1A03070282, 2022R1A2C2005538).

REFERENCES

Gamahara, M. & Hattori, Y. 2017 Searching for turbulence models by artificial neural network. *Phys. Rev. Fluids* **2** (5), 054604.

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

Kim, J., Kim, H., Kim, J. & Lee, C. 2022 Deep reinforcement learning for large-eddy simulation modeling in wall-bounded turbulence .

Kim, J. & Lee, C. 2020 Prediction of turbulent heat transfer using convolutional neural networks. *J. Fluid Mech.* **882**, A18.

Kochkov, D., Smith, J. A., Alieva, A., Wang, Q., Brenner, M. P. & Hoyer, S. 2021 Machine learning–accelerated computational fluid dynamics. *Proc. Natl. Acad. Sci.* **118** (21), e2101784118.

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

Lu, P. Y., Kim, S. & Soljačić, M. 2020 Extracting interpretable physical parameters from spatiotemporal systems using unsupervised learning. *Phys. Rev. X* **10**, 031056.

Novati, G., de Laroussilhe, H.L. & Koumoutsakos, P. 2021 Automating turbulence modelling by multi-agent reinforcement learning. *Nat. Mach. Intell.* **3** (1), 87–96.

Park, J. & Choi, H. 2021 Toward neural-network-based large eddy simulation: application to turbulent channel flow. *J. Fluid Mech.* **914**, A16.

Sirignano, J., MacArt, J. F. & Freund, J. B. 2020 DPM: A deep learning PDE augmentation method with application to large-eddy simulation. *J. Comput. Phys.* **423**, 109811.

Smagorinsky, J. 1963 General circulation experiments with primitive equations. *Mon. Weath. Rev.* **91** (3), 99–164.

Vreman, A. W. 2004 An eddy-viscosity subgrid-scale model for turbulent shear flow: Algebraic theory and applications. *Phys. Fluids* **16** (10), 3670–3681.