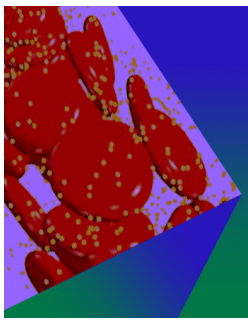
RESEARCH ARTICLE | MAY 09 2023

Log-law recovery through reinforcement-learning wall model for large eddy simulation

Aurélien Vadrot; Xiang I. A. Yang; H. Jane Bae; ... et. al

Physics of Fluids 35, 055122 (2023)
https://doi.org/10.1063/5.0147570



Physics of Fluids ARTICLE pubs.aip.org/aip/phf

Log-law recovery through reinforcement-learning wall model for large eddy simulation

Cite as: Phys. Fluids **35**, 055122 (2023); doi: 10.1063/5.0147570
Submitted: 23 February 2023 · Accepted: 25 April 2023 ·
Published Online: 9 May 2023

View Online Export Citation CrossMark

Aurélien Vadrot,[1] Xiang I. A. Yang,[2] H. Jane Bae,[3] and Mahdi Abkar[1,a)]

AFFILIATIONS

[1]Department of Mechanical and Production Engineering, Aarhus University, Aarhus N 8200, Denmark
[2]Department of Mechanical Engineering, Pennsylvania State University, State College, Pennsylvania 16802, USA
[3]Graduate Aerospace Laboratories, California Institute of Technology, Pasadena, California 91125, USA

[a)]Author to whom correspondence should be addressed: abkar@mpe.au.dk

ABSTRACT

This paper focuses on the use of reinforcement learning (RL) as a machine-learning (ML) modeling tool for near-wall turbulence. RL has demonstrated its effectiveness in solving high-dimensional problems, especially in domains such as games. Despite its potential, RL is still not widely used for turbulence modeling and is primarily used for flow control and optimization purposes. A new RL wall model (WM) called VYBA23 is developed in this work, which uses agents dispersed in the flow near the wall. The model is trained on a single Reynolds number ($Re_\tau = 10^4$) and does not rely on high-fidelity data, as the backpropagation process is based on a reward rather than an output error. The states of the RLWM, which are the representation of the environment by the agents, are normalized to remove dependence on the Reynolds number. The model is tested and compared to another RLWM (BK22) and to an equilibrium wall model, in a half-channel flow at eleven different Reynolds numbers $\{Re_\tau \in [180; 10^{10}]\}$. The effects of varying agents' parameters, such as actions range, time step, and spacing, are also studied. The results are promising, showing little effect on the average flow field but some effect on wall-shear stress fluctuations and velocity fluctuations. This work offers positive prospects for developing RLWMs that can recover physical laws and for extending this type of ML models to more complex flows in the future.

I. INTRODUCTION

Machine learning (ML) techniques have been widely used recently in lots of various domains, including robotics,[1] cybersecurity,[2] biology,[3] games,[4] and others. Computational fluid dynamics (CFD) has not been spared. ML has been applied for turbulence modeling,[5–7] flow control,[8,9] reduced-order modeling,[10] optimization,[11–13] model discovery,[14,15] among others. An overview of ML applications in CFD can be found in Refs. 16–19.

ML elicits as much attention as it does criticism in the turbulence community. First, ML is a subfield of artificial intelligence (AI) that focuses on the development of algorithms and statistical models that enable computers to learn from and make predictions or decisions based on data, without being explicitly programed to do so. ML encompasses a broad spectrum of techniques, ranging from the straightforward, such as linear regression, to the highly complex, such as reinforcement learning (RL). Within the context of turbulence modeling, the term "machine learning" often involves regression, for which the most commonly used tool is the feed-forward neural network (FNN). This type of network processes information in a unidirectional manner, flowing from input nodes through hidden nodes to output nodes.

ML is often seen as a highly effective and versatile solution, and it is applied to a wide range of problems. In the context of modeling, ML is frequently perceived as a method for discovering the ideal model from data, without the need for prior hypotheses, at the expense of physical consideration.[20] ML is frequently referred to as a "black box" because it is difficult to understand the connection between the inputs and the predictions produced by the model. This makes it challenging to form a simplified explanation, or hypothesis, about the relationships in the data. In contrast, traditional modeling techniques involve formulating hypotheses based on observations in a given theoretical framework, constructing a model that represents that hypothesis and testing the model's ability to predict phenomena in this framework.[21] ML often bypass data observations to produce a model, resulting in a lack of prior hypotheses and the discovery of multiple models that perform well on the training set but have unknown theoretical frameworks. When applied to new data, these models often perform poorly and produce unrealistic results,[22] and we have limited control over the

prediction. The solution in ML turbulence modeling involves incorporating physical knowledge to inform, constrain, and/or embed the model.[23,24] ML is commonly perceived as a model in and of itself, but it is actually just a tool. A model should be based on true hypotheses within a physical framework. This does not mean that ML cannot be successful in turbulence modeling, as it is still in its early stages and many possibilities have not been explored yet. Despite its limitations, such as its black-box nature and limited extrapolation capabilities, ML might have the potential to overcome these challenges through the use of various types of neural networks.[25–28] It is important to note that the above-mentioned observations and criticisms about ML are not applicable to all ML techniques and are more commonly made about ML in general rather than specific ML methods.

This paper aims to use a specific ML tool, reinforcement learning (RL), which has limited usage in turbulence modeling,[22,25,29,30] but has various applications in CFD for control and optimization.[9,31] Novati *et al.*[29] were the first to use RL for predicting a sub-grid scale (SGS) turbulence model in the context of large eddy simulations (LES), followed by Refs. 25 and 30. Xiang *et al.*[32] applied RL and accelerated convergence of numerical solutions of the pressure Poisson equation with density discontinuities. Later, Bae and Koumoutsakos[22] resorted to RL for predicting wall-shear stress (τ_w) in the context of LES wall modeling.

RL is an ensemble of algorithms to solve optimization problems. RL does not necessarily require high-fidelity data since the training is done on the fly with a trial-and-error approach. Working directly with *a posteriori* data greatly helps to generalize since the discrepancy between *a priori* and *a posteriori* data (caused by, e.g., discretization and filtering errors) is a main source of errors.[33] The basic idea here is to adjust an agent's or agents' behaviors in an environment to yield desired outcomes.

In the context of LES wall modeling (WM), the WM is the agent, the LES field is the environment, and the desired outcome is an accurate wall-shear stress. The agent is able to learn a relevant policy to maximize its long-term reward. The reward can be local or global, coming from experiment, theory, or high-resolution data. In RL, the reward, serving as the loss function, is not determined by comparing predicted and true outputs, unlike in supervised learning.

RL seems promising in answering the two aforementioned concerns: the "black-box" nature and the limited extrapolation capabilities. Vadrot *et al.*[34] gave promising clue to understand agents' behavior and "predict the prediction" by looking at the states-action map. Moreover, the involvement of physical consideration into RL turbulence modeling is possible through the definition of states, rewards, and actions. The states are the only information about the environment that agents can see. By constraining the states, we can embed physical knowledge in the representation of the environment built by agents. Agents do not need to know all the flow field, and in fact by doing so, we construct a too-specific model that will not be able to adapt to other flows. Bae and Koumoutsakos[22] compare the performance of two models: the velocity-based WM and the log-law-based WM. The latter, with states defined based on the intercept and slope of the log law, achieved better generalization. However, it is not yet perfect, a log-layer mismatch is shown in Ref. 34 at small and large Reynolds numbers that does not result from directly using velocity at the first off-wall grid point.[35,36]

The following paper aims to pursue this work in the context of WM. Modeling near-wall turbulence is a key element for LES. WM consists in a strong simplification of the flow behavior close to the wall. The richness of turbulence composed of streaks and quasi-streamwise vortices is reduced to a single value of the wall-shear stress (τ_w). The outer-layer flow is used to predict the wall-shear stress (or heat flux), which will be itself used to predict the outer-layer flow, following an iterative process. The most basic, yet widely used, form of WM—equilibrium wall model (EWM)[36–40]—is based on a simplified solution of fluid flow near the wall, in which all the non-equilibrium terms (related to, e.g., pressure gradient, acceleration, and buoyancy) are neglected. While EWM is generally effective in predicting the flow behavior, including in non-equilibrium flows,[41] its ability to accurately capture strong non-equilibrium effects in certain flow scenarios can be limited.[42,43] Recent developments in wall modeling, e.g., the integral WM,[44] the non-equilibrium WM,[45] the slip-WM,[46,47] and the Lagrangian relaxation toward equilibrium WM,[43] seek improvements by incorporating a lower level of simplifications. Further efforts are still needed to better predict complex flows with separation, transition, and heat transfer. ML has paved the way for a new branch of WM research that promises to improve precision and complexity beyond what traditional wall modeling techniques can achieve. The ever-increasing availability of high-fidelity simulation data[48–50] has motivated the use of machine-learning wall models (MLWMs). The past few years have seen the development of a number of ML WMs.[5,22,51–59] Yang *et al.*[5] were the first to apply ML in WM using supervised MLWM trained at $Re_\tau = 1000$ to predict the wall-shear stress. Huang *et al.*[51,52] built upon Yang *et al.*'s[5] WM to develop WMs that work well in a spanwise rotating channel and in a channel with arbitrary (in terms of direction) but small (in terms of magnitude) system rotation, while still recovering the law of the wall in basic channels.[34] Bin *et al.*[60] employed a progressive learning approach that emulates the development of empirical WM, gradually increasing the complexity of the flow. Our approach follows a similar progressive methodology, now using RL. Our primary objective is to precisely capture the log law in equilibrium flows prior to attempting more complicated flows. We think it is crucial to proceed with care and not apply RL directly to complex flows without progressively recovering physical laws.

The paper presents an improvement to the original reinforcement-learning wall model (RLWM).[22] The new model, named VYBA23, is trained at a moderately high Reynolds number ($Re_\tau = 10^4$) and uses newly identified states to overcome the limitations of the original BK22 WM.[22] This new model demonstrates a successful recovery of the law of the wall, up to an extremely high Reynolds number ($Re_\tau = 10^{10}$). Moreover, some aspects of RL, such as the impact of the distance between agents, the time step between actions or the range of actions, are still largely under-explored. This paper further investigates these effects. The rest of the paper is organized as follows: The computational setup and RLWMs are detailed in Sec. II. The results are presented in Sec. III, followed by an analysis in Sec. IV and concluding remarks afterward.

II. WALL-MODELED LARGE EDDY SIMULATION DETAILS

A. Flow configuration and flow solver numerics

The configuration is the half-channel flow. The domain size is $L_x \times L_y \times L_z = 2\pi\delta \times 2\pi\delta \times 1\delta$ in the streamwise x, spanwise y, and wall-normal z directions, where δ is the half-channel height. The flow is periodic in both the streamwise and the spanwise directions.

A wall-shear stress boundary condition is imposed at $z = 0$ (wall), and a symmetric condition is imposed at $z = \delta$ (at the half channel height). The flow is driven by a constant pressure gradient in the x direction. The friction Reynolds number is varied in the range $Re_\tau \in [180, 10^3, 2 \times 10^3, 5.2 \times 10^3, 10^4, 10^5, 10^6, 10^7, 10^8, 10^9, 10^{10}]$. Note that the RLWM has been trained using only one Reynolds number within this range ($Re_\tau = 10^4$) and then tested for all Reynolds numbers.

We employ the open-source pseudo-spectral code LESGO, publicly available at https://lesgo.me.jhu.edu.[61] The code uses the spectral method in the x and y directions and the second-order finite difference method in the z direction. The computational domain is divided uniformly into $N_x = 48$, $N_y = 48$, and $N_z = 48$ grid points with the resolution of dx, dy, and dz in the x, y, and z directions, respectively. The grid planes are staggered in the vertical direction, with the first horizontal velocity plane at a distance of $dz/2$ from the surface. The last grid point is just above the physical domain, and therefore, N_z grid points translate to a wall-normal grid spacing of $L_z/(N_z - 1)$. The code has been well validated and extensively used in earlier research publications.[62–67] Furthermore, it has served as a ground for testing SGS models and WMs.[44,68–71] Available SGS models include the constant coefficient,[72] dynamic,[73] Lagrangian dynamic[74] Smagorinsky models, and the minimum dissipation model (AMD).[70,75] Available WMs include EWM,[35,68] the integral WM,[44] the slip-WM,[76] the POD-inspired WM,[77] the supervised MLWMs in Refs. 51 and 53, and the RLWM in Ref. 22.

B. Wall models

Three WMs are considered, namely, the equilibrium wall model, referred to as EWM,[78] the RLWM in Ref. 22, referred to as BK22, and the newly developed RLWM in this paper, referred to as VYBA23. The first two models underwent a thorough comparative analysis in Ref. 34. The latter will be thoroughly investigated and compared to BK22 WM.

1. Empirical WM, EWM

EWM imposes the following law of the wall locally and instantaneously:

$$u^+ = \frac{1}{\kappa} \ln\left(\frac{z}{z_0}\right), \tag{1}$$

where $u^+ = u/u_\tau$ is the inner scaled streamwise velocity, $\kappa \approx 0.4$ is the von Kármán constant, $z_0 = \nu \exp(-\kappa B)/u_\tau$ is a viscous scale, and $B \approx 5$ is the intercept of the log law.[79] The model reads

$$\tau_w = \rho u_\tau^2 = \rho \left[\frac{\kappa \tilde{U}_{\mathrm{LES}}}{\ln(h_{wm}/z_0)}\right]^2, \tag{2}$$

where ρ is the fluid density, U_{LES} is the LES horizontal velocity at a distance h_{wm} from the wall, and $\tilde{(\cdot)}$ denotes possible filtration operation.[35] Equation (2) is implicit and must be solved iteratively. The matching height h_{wm} can be the first, second, or the nth off-wall grid point.[36,53] In this work, we place h_{wm} at $dz/2$, i.e., the first off-wall grid point, and filter the LES velocity to remove the log-layer mismatch.[35] Here, the factor 1/2 is due to the use of a staggered grid.

2. Reinforcement-learning wall model, BK22

Figure 1 shows how the RLWM works. Bae and Koumoutsakos[22] initialized their training with EWM-generated flow fields at $Re_\tau = 2000$, 4200, and 8000. The RLWM was trained using only the initial EWM-generated fields as initial condition, after which it acted independently without further guidance from the EWM-generated fields. Several agents are inserted into the LES flow field, and these agents modify flow fields in order to reach the best policy π. The optimal policy is determined using a neural network featuring two hidden layers, each with 128 neurons, which generate the policy's mean value and standard deviation. The network parameters have been optimized using an off-policy actor-critic algorithm known as V-Racer, with additional information available in Ref. 22. Specifically, the agent produces an action $a_n(x, y)$ on its environment at the instant t_n based on an observation (the states s_n) and a reward r_n, causing the environment to transition from states s_n to states s_{n+1}. The action is to increase or decrease the predicted wall-shear stress by a factor of a_n as

$$\tau_w(x, y, t_{n+1}) = a_n(x, y)\tau_w(x, y, t_n). \tag{3}$$

In Ref. 22, $a_n(x, y) \in [0.9, 1.1]$. We will test the effect of this range on the predictions in Sec. III.

Each agent receives a reward computed as

$$r_n(x, y, t_n) = \frac{|\tau_w^{\mathrm{true}} - \tau_w(x, y, t_n)| - |\tau_w^{\mathrm{true}} - \tau_w(x, y, t_{n-1})|}{\tau_w^{\mathrm{true}}} + \mathrm{Bonus}. \tag{4}$$

τ_w^{true} is the true mean wall-shear stress obtained from the equations of motion for an unidirectional mean flow. The approach of converging to the theoretical mean value is a common wall modeling strategy in LES, such as the equilibrium wall model, which enforces the law of the wall locally and instantaneously. Since LES involves unsteady and dynamic processes, connecting the wall-shear stress with a fluctuating LES velocity can result in varying wall-shear stress, even when the law of the wall is imposed. The objective of RLWM is to accomplish the same effect. In a channel flow, the mean wall-shear stress at the lower wall ($z = 0$) is related to the mean streamwise pressure gradient as

$$\tau_w^{\mathrm{true}} = -\delta \frac{dp_0}{dx}. \tag{5}$$

In LESGO, the flow is driven by a constant pressure gradients, and τ_w^{true} is, thus, known. The bonus reward helps to accelerate the convergence of the model if expressed as

$$\mathrm{Bonus} = \begin{cases} 1 - \dfrac{|\tau_w^{\mathrm{true}} - \tau_w(x, y, t_n)|}{\tau_w^{\mathrm{true}}}, & \text{if } |\tau_w^{\mathrm{true}} - \tau_w(x, z, t_n)|/\tau_w^{\mathrm{true}} < 0.1; \\ 0 & \text{otherwise.} \end{cases} \tag{6}$$

It is worth noting that the agents in Fig. 1 are single-policy agents. That means that from given states, all agents will predict the same action.

In Ref. 22, two models are trained: one called the velocity wall model (VWM) and one called the log-law wall model (LLWM). We consider only LLWM, which was proven to have better extrapolation capabilities because of its definition based on the empirical log law. The states (s_1 and s_2) of LLWM are, respectively, defined as

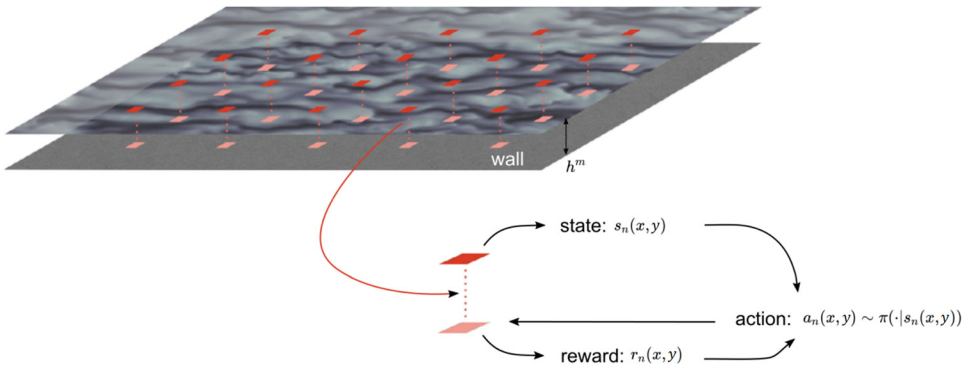


FIG. 1. Several single-policy agents are distributed in the (x, y) plane at h_{wm}. Reprinted with permission from H. J. Bae and P. Koumoutsakos, Nat. Commun. **13**, 1443 (2022). Copyright 2022 Authors, licensed under a Creative Commons Attribution 4.0 International License.

$$s_1 = \frac{1}{\kappa_{wm}} = \frac{h_{wm}}{u_{\tau_{wm}}} \frac{\partial u}{\partial z}\bigg|_{z=h_{wm}}, \tag{7a}$$

$$s_2 = B_{wm} = \frac{u_{\mathrm{LES}}}{u_{\tau_{wm}}} - \frac{1}{\kappa_{wm}} \ln\left(\frac{h_{wm} u_{\tau_{wm}}}{\nu}\right), \tag{7b}$$

where $u_{\mathrm{LES}} = u(h_{wm})$ is the streamwise LES velocity taken at h_{wm}, and $u_{\tau_{wm}} = \sqrt{\tau_w/\rho}$ is the model-computed friction velocity. We place the matching location h_{wm} at dz, i.e., between the first and the second off-wall grid points, for ease of computing the velocity derivative. It may be worth noting that the implementation of RLWM is not straightforward. The present model requires the coupling of an RL library, here the Smarties library[80] with the LES solver.

At first, BK22's results are recovered from this newly coupling between LESGO and the Smarties libraries. Training is performed on $Re_\tau = 10^4$. The results are displayed in Fig. 2. The same log-layer mismatch at large Reynolds numbers is observed for both models. At low Reynolds numbers, the log-layer mismatch is suppressed for the newly trained model. The models under comparison have identical states, rewards, and actions. The differences arise from the Reynolds number used in the training, which is slightly higher in our study, and from the specific RL settings. It is worth noting that the choice of solver may also contribute to differences since the BK22 model was initially trained using a solver with a second-order finite difference spatial scheme, whereas it is now tested in a pseudo-spectral code. A comparison with the original paper[22] can be found in Ref. 34.

3. Reinforcement-learning wall model, VYBA23

The newly developed RLWM is based on the exact same RL algorithm as in Ref. 22. It is designed to address the limitations of BK22 WM at high Reynolds numbers. The states in Eqs. (7a) and (7b) need to be corrected to achieve this. The limitations of BK22 WM are discussed in Ref. 34. This issue arises from the rotation of the neutral line in the states-action map. The neutral line is defined as

$$u^+_{\mathrm{LES}} - u^+_{\mathrm{LL}} = 0. \tag{8}$$

Here, u^+_{LL} is the velocity obtained from the log law with $\kappa = 0.4$ and $B = 5$. Both velocities are evaluated at the matching location h_{wm}. Equation (8) can be explicitly expressed from explicit quantities κ, κ_{wm}, B, B_{wm}, and h_{wm} as

$$\frac{1}{\kappa_{wm}} \ln\left(h^+_{wm}\right) + B_{wm} - \left(\frac{1}{\kappa} \ln\left(h^+_{wm}\right) + B\right) = 0. \tag{9}$$

For states located above this neutral line, the velocity is larger than the log-law value. Given these states, the RLWM should ideally generate an action $a_n > 1$. By doing so, the wall-shear stress would increase,

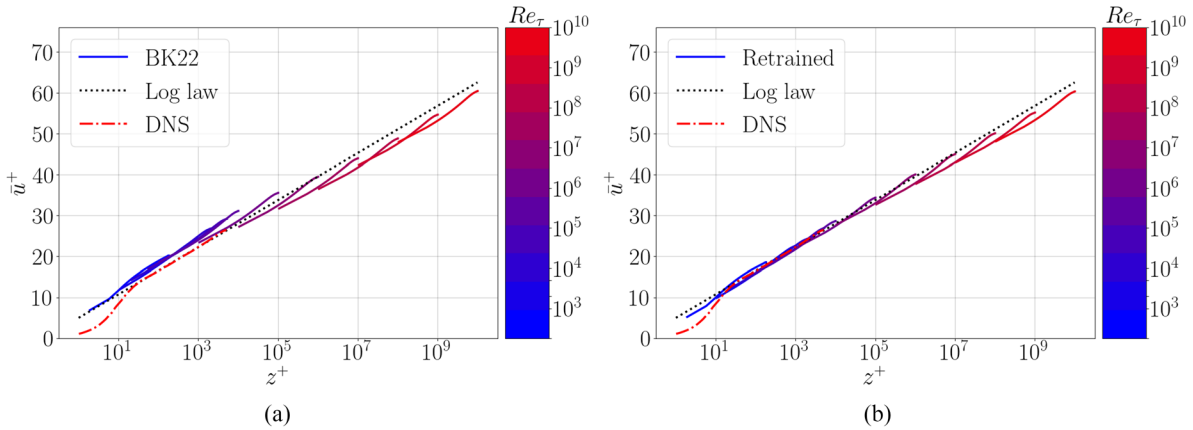
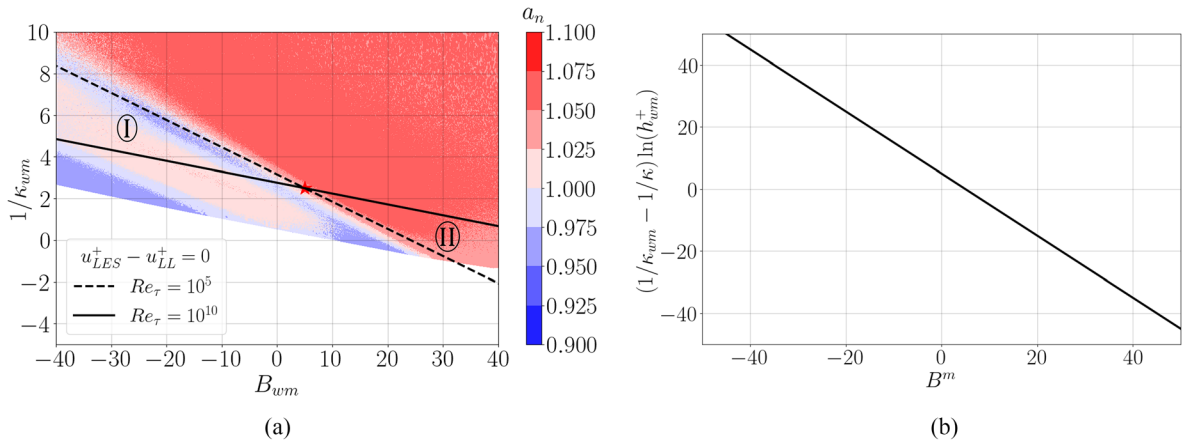


FIG. 2. Mean streamwise velocity $\bar{u}^+$ as a function of the wall-normal direction z^+ at 11 Reynolds numbers between $Re_\tau = 180$ and 10^{10}. (a) BK22 tested in LESGO (Ref. 22) and (b) retrained RLWM in LESGO. DNS result at $Re_\tau = 5200$ is included for comparison purposes.[50] The log law corresponds to $\kappa = 0.4$ and $B = 5$.



(a) (b)

FIG. 3. (a) States-action map at $Re_\tau = 10^{10}$ for BK22 WM. The action contours show in red the increasing actions and in blue decreasing actions. The neutral lines are, respectively, plotted at $Re_\tau = 10^5$ and 10^{10}. (b) States-action map with the new state $s_1 = (1/\kappa_{wm} - 1/\kappa)\ln(h^+_{wm})$. The neutral line is identical for all Re_τ.

which is anticipated to result in a drop in the local velocity, thereby bringing down the velocity to the log-law value. On the contrary, given states below the neutral line, the RLWM should ideally generate an action $a_n < 1$ to bring up the velocity to the log-law value. The reader will notice that RL models are amenable to physical interpretation. Visualizing the states-action map plot can help address the "black-box" criticism frequently attributed to ML models, as it enables a better comprehension of the learned policy of agents and its physical interpretation.

Figure 3(a) displays the states-action map at $Re_\tau = 10^{10}$ for BK22 WM. When increasing the Reynolds number, the neutral line rotates in counterclockwise and rotation saturates at large Reynolds numbers, explaining the saturating log-layer mismatch. This rotation is caused by the decrease in the slope of the neutral line $-1/[\ln(dz/\delta) + \ln(Re_\tau)]$ for increasing Re_τ (see Ref. 34 for further details).

To solve this issue, we propose to normalize the first state to remove the dependency on h^+_{wm}. If so, an agent trained for an arbitrary Reynolds number could be able to extrapolate to any unseen Reynolds number. We propose the following expression for the first state s_1

$$s_1 = \left(\frac{1}{\kappa_{wm}} - \frac{1}{\kappa}\right)\ln(h^+_{wm}). \tag{10}$$

Now, the first state depends on the value of κ, but the neutral lines of the states-action map collapse into a single line whatever the Reynolds number is [see Fig. 3(b)]. The value of κ in the log law is not universal and can depend on various factors, such as the Reynolds number, surface roughness, and boundary layer type. Empirical studies have suggested values of κ ranging from 0.36 to 0.44, with values around 0.4 being commonly used as a default value.[81]

The parameters that are defined for agents in the context of multi-agents RLWM include the action range (Δa), where a_n falls within the range $[1 - \Delta a; 1 + \Delta a]$, the time step between actions (Δt), and the horizontal (Δx) and vertical (Δy) distances between agents. During the training of VYBA23 WM, the parameters specified in Table I were used. The impact of these parameters on the flow will be examined in Sec. III. VYBA23 WM was trained from an LES at $Re_\tau = 10^4$ using EWM, but this process was challenging and time-consuming. The training involved coupling the RL library, Smarties, which is coded in C++, with the Fortran-based CFD solver LESGO. Achieving proper convergence required a significant amount of trial and error with hundreds of simulations and millions of policy gradient updates. It is worth noting that some models may not converge, and several models with different reward functions were trained before settling on the VYBA23 model.

III. RESULTS

We trained a new RLWM, namely, VYBA23, using the new expression for the first state [Eq. (10)] and based on parameters from Table I. In this section, we evaluate this model. BK22 WM (Ref. 22) and EWM (Ref. 36) are sometimes shown for comparison.

A. The effect of reinforcement-learning wall models

Figures 4 and 5 show instantaneous snapshots of the contours of streamwise velocity fluctuations at, respectively, the first off-grid point wall and at $z/\delta = 0.5$. The friction Reynolds number is equal to 10^5. The flow fields are alike between EWM,[36] BK22,[22] and VYBA23. The flow is composed of high-intense small-scale turbulent structures in the near-wall region and of large-scale streaks at $z/\delta = 0.5$. For all the three WMs, both intensity and scale-sizes are similar.

Figure 6 displays the mean velocity profiles. VYBA23 WM captures properly the log law when the matching location h_{wm} is in the log layer (i.e., at $Re_\tau = 180$, the prediction is above the DNS profile). VYBA23 WM outperforms BK22 WM (Ref. 22) (Fig. 2), providing an accurate prediction of the log law outside of its training range ($Re_\tau = 10^4$), in high Reynolds numbers channel flows (up to $Re_\tau = 10^{10}$).

TABLE I. Details of VYBA23.

WM	Δa	Δt	Δx	Δy	N_{agents}	Training Re_τ
VYBA23	0.10	$10dt$	$3dx$	$3dy$	256	10^4

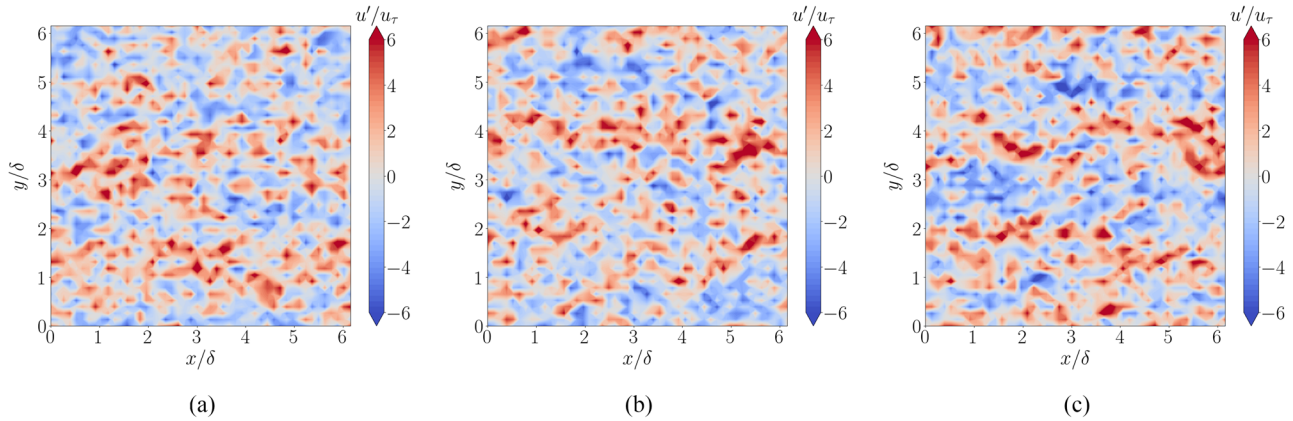


FIG. 4. Contours of the fluctuating streamwise velocity in the x–y plane at the first off-wall grid point $z = dz/2$. The flow is at $Re = 10^5$. (a) EWM,[36] (b) BK22,[22] and (c) VYBA23. The grid resolution is $N_x \times N_y \times N_z = 48^3$.

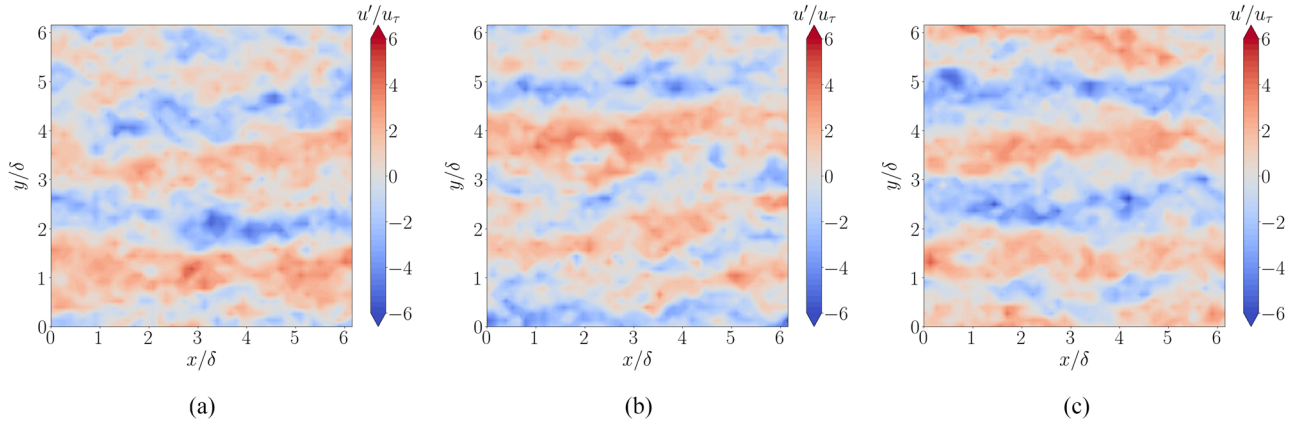


FIG. 5. The same as Fig. 4 but at $z/\delta = 0.5$.

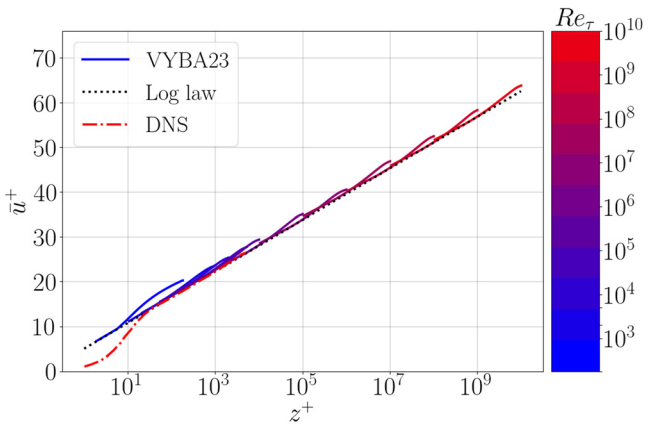


FIG. 6. Mean streamwise velocity $\bar{u}^+$ as a function of the wall-normal direction z^+ at 11 Reynolds numbers between $Re_\tau = 180$ and 10^{10} for VYBA23 WM. DNS result at $Re_\tau = 5200$ is included for comparison purposes.[50] The log law corresponds to $\kappa = 0.4$ and $B = 5$.

Figure 7 displays the root mean square (rms) value of the velocity fluctuations $u^+_{\rm rms}$ for EWM, BK22, and VYBA23 WMs.

The recent DNS and experimental studies of boundary-layer flows at high Reynolds numbers have revealed an outer peak in $u^{+^2}_{\rm rms}$ in addition to the inner peak.[79,82] Here, the inner peak cannot be observed, and the outer peak is barely noticeable. The explanations for this are provided below. The bump caused by the outer layer peak is visible at $z/\delta \approx 0.1$. However, none of WMs capture this bump, likely due to insufficient resolution to accurately represent the outer peak. The inner peak is located at around $z^+ \approx 15$ within the buffer layer, while the outer peak emerges at approximately $z^+ \approx 3\sqrt{Re_\tau}$ for high Reynolds numbers,[79] corresponding to $z/\delta = 3/\sqrt{Re_\tau}$. For $Re_\tau = 10^4$, the outer peak is at $z/\delta \approx 0.03$. As the Reynolds number increases, the outer peak moves closer to the wall in terms of z/δ. Since the grid resolution in WMLES is coarse, a typical WMLES does not resolve either the inner or outer peaks. From the outer peak toward the channel center, the streamwise velocity rms scales as

$$u^{+^2}_{\rm rms} \sim \ln(\delta/z). \tag{11}$$

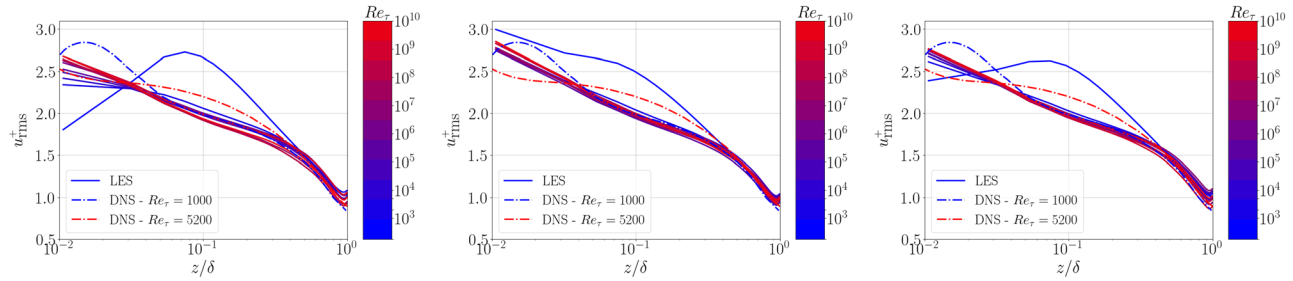


FIG. 7. Root mean square of the streamwise velocity fluctuation $u^+_{\rm rms}$ as a function of the wall-normal co-ordinate z/δ plotted for between $Re_\tau = 180$ and 10^{10}. (a) EWM,[36] (b) BK22,[22] and (c) VYBA23. DNS data at $Re_\tau = 1000$ and $Re_\tau = 5200$ are included for comparison.[49,50]

This scaling is independent of the Reynolds number, which is why the WMLES-predicted $u^+_{\rm rms}$ collapses at sufficiently high Reynolds numbers. The curve at $Re_\tau = 180$ differs from the others because the Reynolds number is too low.

It is important to note that in this study, each LES was carefully checked for statistical convergence by examining the convergence of kinetic energy and the stress balance. It is well established that once equilibrium is reached, the total stress, which is the sum of the filter (modeled) and resolved stresses, becomes a linear function of the wall-normal distance. Figure 8 demonstrates the successful convergence of the total stress for $Re_\tau = 10^8$. The figure shows that the total stress obtained almost perfectly overlaps with the ideal total stress.

B. Further results

In order to assess the impact of agents' parameters on the results, various combinations were tested by varying the agents' spacing (Δx and Δy), the range of actions (Δa), and the time step (Δt). The tested combinations are listed in Table II. Figure 9 illustrates the influence of Δa, Δt, and $N_{\rm agents}$ on the log-layer mismatch. A 5% error in the predicted wall-shear stress is expected due to a 2.5% uncertainty in the von Kármán constant κ.[5] This translates to about 1–2 plus units in the context of log-layer mismatch. The log-layer mismatch problem that was identified in BK22 WM (Ref. 34) has been addressed, with most of the parameters of the agents having limited impact on the mean velocity distribution. The only exception is when the time step between actions becomes too large ($\Delta t > 10dt$). That strengthens the reliability of RLWM methods, where the mean field is largely unchanged by the agents' parameters. However, the parameters may still affect the higher-order statistics of the flow, particularly the fluctuations in the wall-shear stress. RLWM adjusts the wall-shear stress to match its true mean value [as shown in Eq. (5)], leading to varying levels of fluctuations depending on the choice of agents' parameters. It is acknowledged that fluctuations of the wall-shear stress can have a significant impact on the flow.[83] The selection of agents' parameters provides a great deal of flexibility (as seen in Fig. 10).

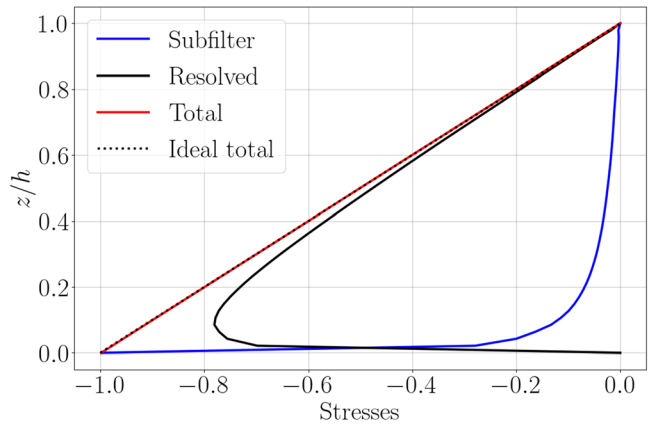


FIG. 8. Vertical profiles of the normalized total and partial shear stress for $Re_\tau = 10^8$ VYBA23 WM.

The fluctuations of the wall-shear stress decrease as the friction Reynolds number increases. This can be explained by rewriting Eq. (2) as

$$\tau_w = \rho u_\tau^2 = \rho \left[\frac{\kappa \tilde{U}_{\rm LES}}{\ln (h^+_{wm}) + B} \right]. \tag{12}$$

Wall-shear fluctuations depend on $1/(\ln (h^+_{wm} + B)$ that causes the wall-shear fluctuations decrease for increasing Reynolds numbers. The range and time step of actions are the most influential parameters, with fluctuations increasing with range and decreasing with time step. The impact of agents' spacing on the wall-shear stress fluctuations is minimal, as states are likely to be similar between two agents and

TABLE II. Test of VYBA23 WM with different running settings.

Δa	Δt	Δx	Δy	$N_{\rm agents}$
0.10	$10dt$	$3dx$	$3dy$	256
0.10	$10dt$	$3dx$	$6dy$	128
0.10	$10dt$	$6dx$	$6dy$	64
0.05	$10dt$	$3dx$	$3dy$	256
0.025	$10dt$	$3dx$	$3dy$	256
0.10	dt	$3dx$	$3dy$	256
0.10	$5dt$	$3dx$	$3dy$	256
0.10	$20dt$	$3dx$	$3dy$	256

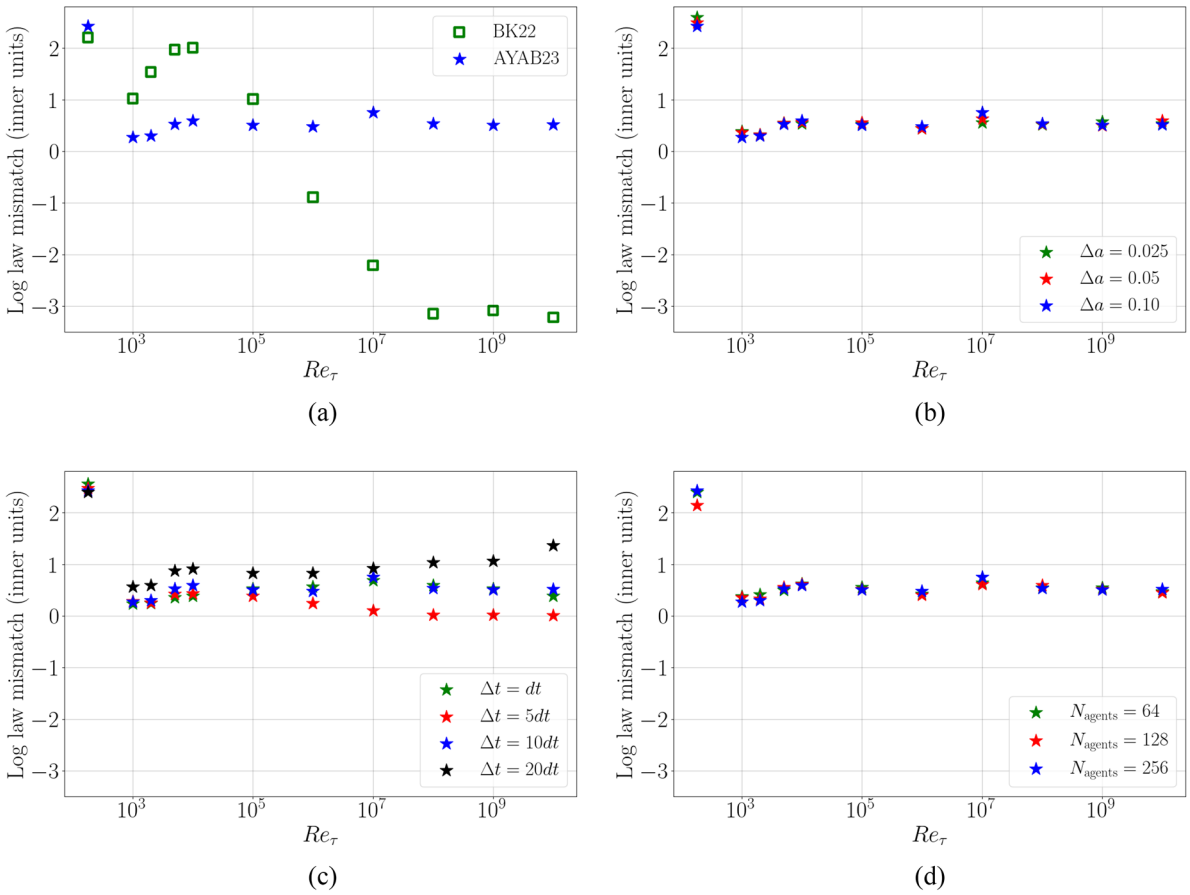


FIG. 9. Log-layer mismatch in inner units as a function of the friction Reynolds number Re_τ for BK22 (Ref. 22) and VYBA23. The agents' parameters are tested individually by changing one parameter at a time while keeping the others at their baseline configuration ($\Delta a = 0.10$, $\Delta t = 10dt$, and $N_{agents} = 256$, see Table I). The baseline log law is $\ln(z^+)/\kappa + B$ with $\kappa = 0.4$ and $B = 5$.

adding an agent between them would not result in significant differences compared to the interpolation of the wall-shear stress. The optimal setting to match filtered DNS based on wall-shear stress fluctuations results is $(\Delta a; \Delta t; \Delta x; \Delta y) = (0.10; 5dt; 3dx; 3dy)$. However, it is important to note that this choice may be influenced by the numerical schemes used.

IV. ANALYSIS

In this section, we pursue a physical interpretation of the RL WM results, a task that is typically challenging and not frequently undertaken in other ML investigations. This RL approach is designed in a simple way to control the wall-shear stress and has a limited number of states, allowing for easy visualization of the actions in a 2D plot. This is not the case for many other RL models with numerous states, nor for most supervised models that have multiple features, making it challenging to visualize in 2D spaces, and for which, other methods, such as the Shapley additive explanations tool (SHAP), need to be used.

Agents learnt a policy through experiences at a Reynolds number of 10^4. They learnt to make decisions regarding the wall-shear stress using only the local values of κ_{wm} and B_{wm} to align with the log law. If the velocity, u^+_{LES}, exceeds the log-law velocity, u^+_{LL}, the agents act to increase the wall-shear stress to bring the velocity back to the log-law profile. Conversely, if the velocity is below the average, the agents decrease the wall-shear stress. The normalization of the first state compensates for an increase in the Reynolds number. The difference between $1/\kappa_{wm}$ and $1/\kappa$ is weighted with $\ln(h^+_{wm})$ to correct the rotation of the neutral line in the states-action map, rather than using raw $1/\kappa_{wm}$. The states-space is designed to be replicable for Reynolds numbers outside the training range, with the exception of when h^+_{wm} is located in the buffer or viscous layers, in which case VYBA23 WM overpredicts the velocity. Figure 11 illustrates the agents' policy at Reynolds numbers 180, 10^3, 10^5, and 10^{10}. The cluster of points in the states-action map spreads along the neutral line as the Reynolds number increases, due to the increase in h^+_{wm}. Despite the wider spread of the cluster of points when compared to the learnt case ($Re_\tau = 10^4$), the policy of VYBA23 WM remains consistent for higher Reynolds numbers.

Additionally, the performance of the agents' policies can be evaluated using quantitative measures, such as precision and recall, defined as follows:

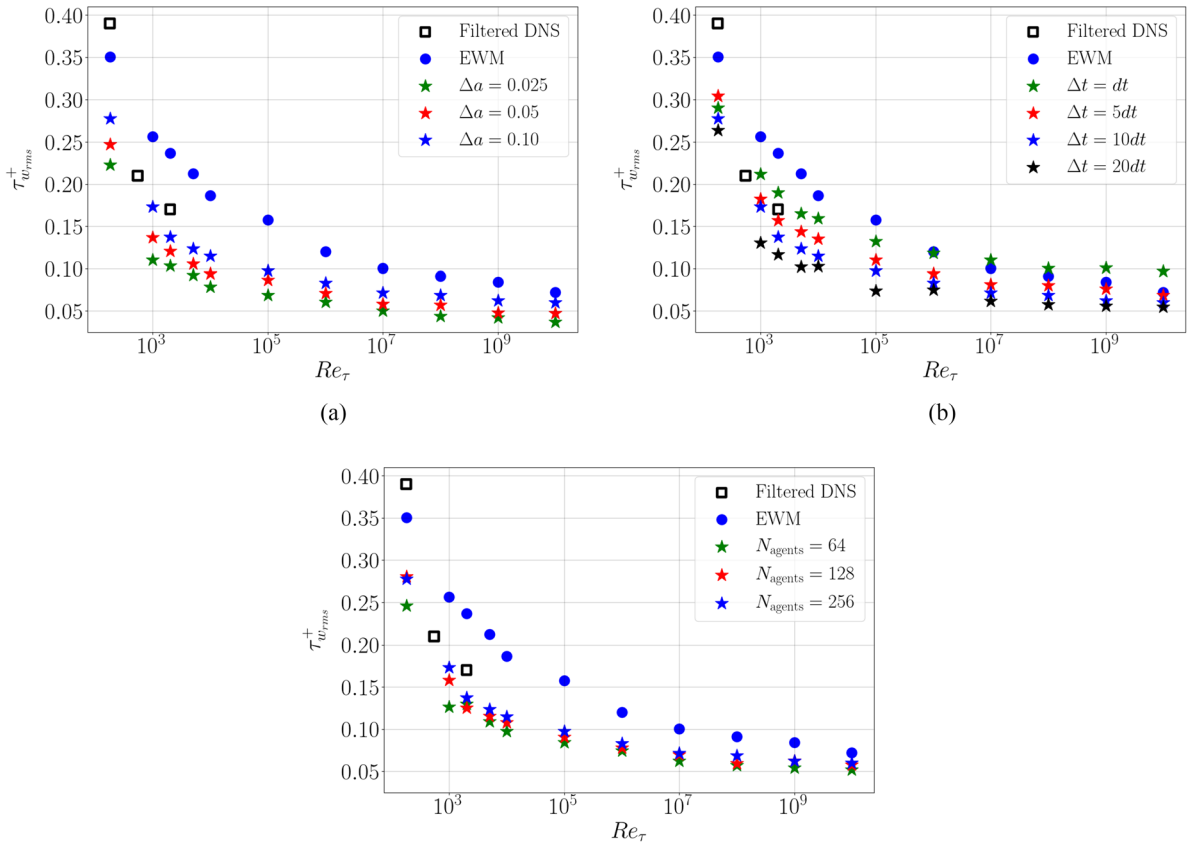


FIG. 10. Evolution of wall-shear stress fluctuations for VYBA23, EWM (Ref. 36), and filtered DNS.[35] DNS are filtered with a top-hat filter. The agents' parameters are tested individually by changing one parameter at a time while keeping the others at their baseline configuration ($\Delta a = 0.10$, $\Delta t = 10dt$, and $N_{agents} = 256$, see Table I).

$$\begin{aligned}\text{precision}(a_n > 1) &= \frac{\text{true positive}}{\text{true positive} + \text{false positive}} \\ &= \frac{\text{instances of A}}{\text{instances of A} + \text{instances of C}}, \\ \text{recall}(a_n > 1) &= \frac{\text{true positive}}{\text{true positive} + \text{false negative}} \\ &= \frac{\text{instances of A}}{\text{instances of A} + \text{instances of B}}.\end{aligned} \tag{13}$$

Similarly, precision and recall can also be defined for action $a_n < 1$. Accuracy can be calculated as

$$\text{Accuracy} = \frac{\text{instances of A and D}}{\text{all instances}}. \tag{14}$$

The results of these measures are presented in Table III at Reynolds numbers of 180, 10^3, 10^5, and 10^{10}. VYBA23 WM consistently demonstrates higher precision, recall, and accuracy levels compared to BK22 WM.[22] Although there is a slight decrease in performance at Reynolds number of 10^{10}, the results are still largely satisfactory.

The agents' action amplitude and time step have a similar impact on wall-shear stress fluctuations (see Fig. 10). To find a similarity between these two parameters, we derived an equation showing that the ratio of action amplitude to time step must be the same for both WMs.

Considering two RLWMs 1 ($\Delta a = a_1; \Delta t = b_1 dt$) and 2 ($\Delta a = a_2; \Delta t = b_2 dt$), between t and $t + \delta t$, the agents act on the shear stress as

$$\tau_{t+\delta t} = (1 + a_1)^{\delta t/(b_1 dt)} \tau_t = (1 + a_2)^{\delta t/(b_2 dt)} \tau_t, \tag{15}$$

so that

$$\frac{1}{b_1} \ln(1 + a_1) = \frac{1}{b_2} \ln(1 + a_2). \tag{16}$$

Considering that $a_1 \ll 1$,

$$\frac{a_1}{b_1} = \frac{a_2}{b_2}. \tag{17}$$

We tested this rule for two cases of action amplitude ratios ($a_1/a_2 = 2$ and $a_1/a_2 = 10$). The results are quite conclusive since curves overlap in both cases (see Fig. 12). However, it is more computationally efficient to increase the action range rather than the time step to have consistent effects.

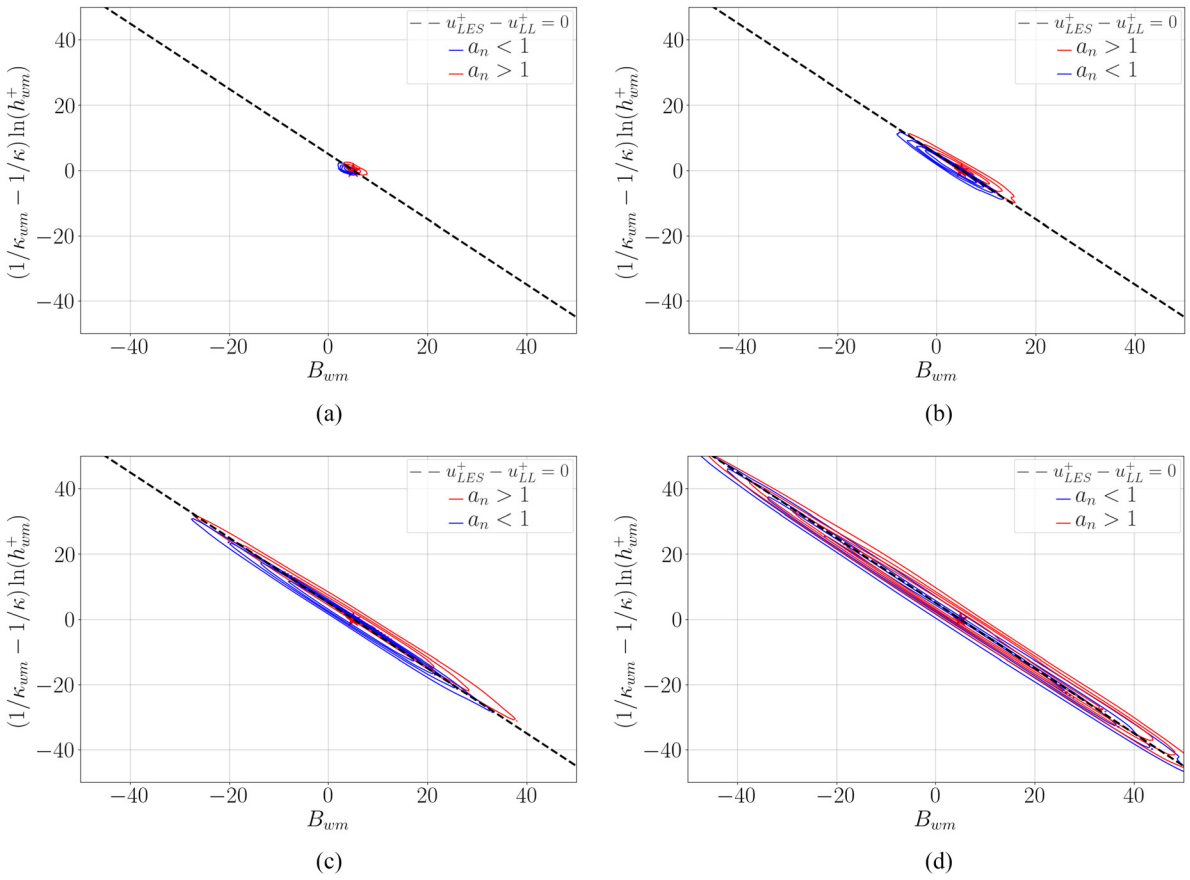


FIG. 11. States-action maps for VYBA23 WM at (a) $Re_\tau = 180$, (b) $Re_\tau = 10^3$, (c) $Re_\tau = 10^5$, and (d) $Re_\tau = 10^{10}$. The following density contours are plotted: 20%, 40%, 60%, and 80%.

It is crucial to be mindful of the consequences of changing both the action range and time step on the flow field, as these changes can impact the Reynolds-number similarity, as shown in Figs. 13–15. Although the curves follow a very similar trend and almost collapse, it is noted that all three parameters have an effect on it. A reduction in the action range leads to more similar results, while for the number of agents or the time step, there is likely an optimal value somewhere in the middle of the tested range. Based on these results, it is recommended to reduce the action range or to set the time step to $\Delta t = 5dt$, which results in wall-shear stress fluctuations that are closer to the filtered DNS (Fig. 10), less log-layer mismatch (Fig. 9), and less impact on the Reynolds number independence of velocity fluctuations (Fig. 14). However, the relationship between agent's parameters and Reynolds-number similarity is not yet well understood, and this decision may largely depend on the numerical solver used.

V. CONCLUDING REMARKS

This paper presents a new WM (VYBA23) based on RL that use agents to control the wall-shear stress in a flow. The model has been developed after an analysis of a previous RLWM (BK22) that was found to produce a log-layer mismatch at large Reynolds numbers due to their choice of states.[34] The VYBA23 model outperforms BK22 WM by normalizing one of the states and is able to accurately predict the log law for large Reynolds numbers (up to $Re_\tau = 10^{10}$). It has also been tested for its ability to predict wall-shear stress fluctuations and has been found to be consistent with filtered DNS[35] and with EWM.[36]

TABLE III. Precision, recall, and accuracy scores for BK22 and VYBA23 WMs.

		Precision		Recall		
WM	Re_τ	$a_n > 1$	$a_n < 1$	$a_n > 1$	$a_n < 1$	Accuracy
BK22	180	0.61	0.38	0.48	0.51	0.49
VYBA23	180	0.91	0.92	0.90	0.93	0.92
BK22	10^3	0.61	0.40	0.49	0.52	0.50
VYBA23	10^3	0.92	0.91	0.90	0.93	0.91
BK22	10^5	0.66	0.40	0.51	0.55	0.53
VYBA23	10^5	0.85	0.80	0.80	0.85	0.83
BK22	10^{10}	0.57	0.42	0.45	0.53	0.49
VYBA23	10^{10}	0.64	0.65	0.67	0.62	0.64

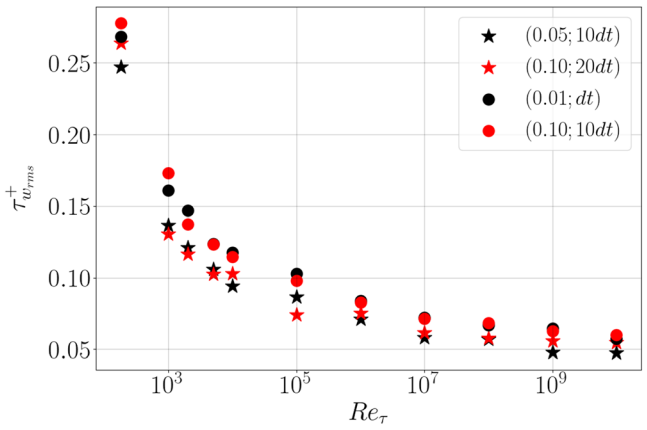


FIG. 12. Wall-shear stress fluctuations as a function of friction Reynolds number. We test the similarity rule between two pairs of RLWMs varying agents' parameters: $(0.05; 10dt)$ with $(0.10; 20dt)$ and $(0.01; dt)$ with $(0.10; 10dt)$. The number of agents is set to $N_{\text{agents}} = 256$.

The effect of changing the parameters of the agents, such as the action range (Δa), time step (Δt), and the number of agents (N_{agents}), was also studied. The predictions of the log law remain unchanged, but it was found that these parameters can affect wall-shear stress fluctuations and velocity fluctuations.

This research provides promising results as the VYBA23 model was trained using only a single Reynolds number ($Re_\tau = 10^4$) without the need for high-fidelity data, yet it still demonstrates good performance in a much broader range. RL algorithms surpass supervised ML as they work directly with a reward rather than an output error, eliminating the need for filtered DNS data and the inconsistency problem between the *a priori* and *a posteriori* computations of the inputs of the model.

Physical knowledge can be incorporated into these models through the use of rewards and states, as proven by the works of Ref. 22 and this research. By shaping the states, the way the RL algorithm represents the environment can be controlled and the dependence on the Reynolds number can be reduced. Furthermore, the states-action map, which serves as the policy of the agents, provides a clear understanding of the actions taken by the RL model. This eliminates the black-box nature of traditional ML models, providing a more interpretable solution.

To implement this type of ML model, a coupling of the RL library with the CFD solver is required, which can be challenging due to the need for communication between different programing languages. In the future, if the superiority of these models is established, it may be worthwhile to explore more efficient methods for extracting the trained model, thus eliminating the need for coupling with the RL library. Such a development could increase exchange and testing of RLWMs within the community, thereby improving their adoption and confidence in their use.

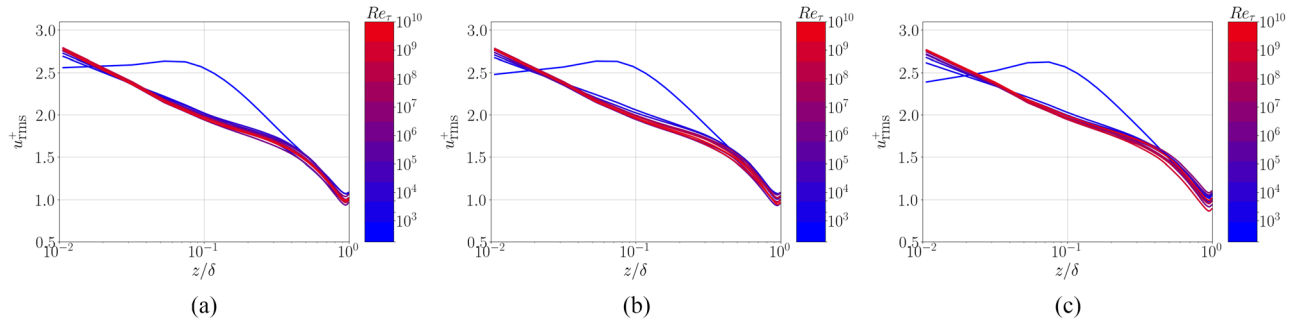


FIG. 13. Velocity fluctuations for (a) $\Delta a = 0.025$, (b) $\Delta a = 0.05$, and (c) $\Delta a = 0.10$. Other parameters are set to the baseline configuration ($\Delta t = 10dt$ and $N_{\text{agents}} = 256$, see Table I).

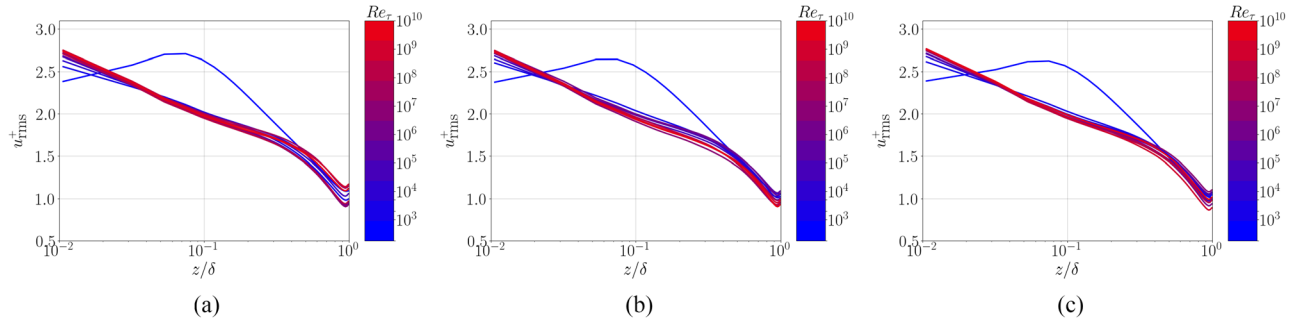


FIG. 14. Velocity fluctuations for (a) $\Delta t = dt$, (b) $\Delta t = 5dt$, and (c) $\Delta t = 10dt$. Other parameters are set to the baseline configuration ($\Delta a = 0.10$ and $N_{\text{agents}} = 256$, see Table I).

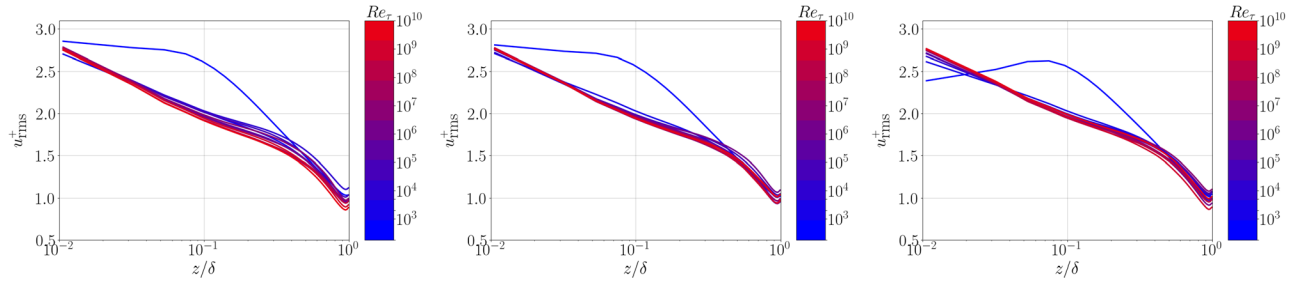


FIG. 15. Velocity fluctuations for (a) $N_{\mathrm{agents}} = 64$, (b) $N_{\mathrm{agents}} = 128$, and (c) $N_{\mathrm{agents}} = 256$. Other parameters are set to the baseline configuration ($\Delta a = 0.10$ and $\Delta t = 10dt$, see Table I).

The strength of the RL approach is discussed in detail in Ref. 22. Conventional empirical models, RLWMs, and other ML WMs all have their own strengths and weaknesses. The purpose of this paper is not to establish the superiority of the RL approach over other approaches. Rather, we aim to address an issue identified in Ref. 22, namely, the log-law recovery. We consider the recovery of the log law to be an essential step toward creating more general WMs—if a model cannot capture the log law, it would be hard for it to capture other flow phenomena. RL has demonstrated its potential in solving very large dimensional problems, with the ability to handle up to 10^{170} states in the game of Go,[84] representing a significant achievement in the field of ML. It is believed that this capability could be useful in turbulence modeling, which is a high dimensional problem due to the wide range of scales that increases with the Reynolds number. Such an approach could prove particularly valuable in predicting complex flow problems, including separated flows, where significant progress could be made.

ACKNOWLEDGMENTS

This research was supported by the Independent Research Fund Denmark (DFF) under the Grant No. 1051–00015B. Yang acknowledges US Office of Naval Research under Contract No. N000142012315, with Dr. Peter Chang as Technical Monitor.

AUTHOR DECLARATIONS

Conflict of Interest

The authors have no conflicts to disclose.

Author Contributions

Aurélien Vadrot: Conceptualization (equal); Data curation (lead); Formal analysis (equal); Investigation (lead); Methodology (equal); Software (lead); Validation (lead); Visualization (lead); Writing – original draft (lead). **Xiang Yang:** Conceptualization (equal); Formal analysis (equal); Methodology (equal); Writing – review & editing (equal). **H. Jane Bae:** Conceptualization (equal); Formal analysis (equal); Methodology (equal); Writing – review & editing (equal). **Mahdi Abkar:** Conceptualization (equal); Formal analysis (equal); Funding acquisition (lead); Methodology (equal); Project administration (lead); Supervision (lead); Writing – review & editing (equal).

DATA AVAILABILITY

The data that support the findings of this study are available from the corresponding author upon reasonable request.

REFERENCES

[1]J. Kober, J. A. Bagnell, and J. Peters, "Reinforcement learning in robotics: A survey," Int. J. Rob. Res. **32**, 1238–1274 (2013).
[2]T. T. Nguyen and V. J. Reddi, "Deep reinforcement learning for cyber security," IEEE Trans. Neural Networks Learn. Syst. (published online 2021).
[3]J. Jumper, R. Evans, A. Pritzel, T. Green, M. Figurnov, O. Ronneberger, K. Tunyasuvunakool, R. Bates, A. Žídek, A. Potapenko *et al.*, "Highly accurate protein structure prediction with AlphaFold," Nature **596**, 583–589 (2021).
[4]D. Silver, T. Hubert, J. Schrittwieser, I. Antonoglou, M. Lai, A. Guez, M. Lanctot, L. Sifre, D. Kumaran, T. Graepel *et al.*, "A general reinforcement learning algorithm that masters chess, shogi, and go through self-play," Science **362**, 1140–1144 (2018).
[5]X. Yang, S. Zafar, J.-X. Wang, and H. Xiao, "Predictive large-eddy-simulation wall modeling via physics-informed neural networks," Phys. Rev. Fluids **4**, 034602 (2019).
[6]R. Maulik, O. San, J. D. Jacob, and C. Crick, "Sub-grid scale model classification and blending through deep learning," J. Fluid Mech. **870**, 784–812 (2019).
[7]A. Eidi, N. Zehtabiyan-Rezaie, R. Ghiassi, X. Yang, and M. Abkar, "Data-driven quantification of model-form uncertainty in Reynolds-averaged simulations of wind farms," Phys. Fluids **34**, 085135 (2022).
[8]F. Ren, H-b Hu, and H. Tang, "Active flow control using machine learning: A brief review," J. Hydrodyn. **32**, 247–253 (2020).
[9]L. Guastoni, J. Rabault, P. Schlatter, H. Azizpour, and R. Vinuesa, "Deep reinforcement learning for turbulent drag reduction in channel flows," arXiv:2301.09889 (2023).
[10]K. Hasegawa, K. Fukami, T. Murata, and K. Fukagata, "Machine-learning-based reduced-order modeling for unsteady flows around bluff bodies of various shapes," Theor. Comput. Fluid Dyn. **34**, 367–383 (2020).
[11]J. Viquerat, J. Rabault, A. Kuhnle, H. Ghraieb, A. Larcher, and E. Hachem, "Direct shape optimization through deep reinforcement learning," J. Comput. Phys. **428**, 110080 (2021).
[12]B. S. Mekki, J. Langer, and S. Lynch, "Genetic algorithm based topology optimization of heat exchanger fins used in aerospace applications," Int. J. Heat Mass Transfer **170**, 121002 (2021).
[13]M. J. Rincón, M. Reclari, X. I. Yang, and M. Abkar, "Validating the design optimisation of ultrasonic flow meters using computational fluid dynamics and surrogate modelling," Int. J. Heat Fluid Flow **100**, 109112 (2023).
[14]J. L. Callaham, J. V. Koch, B. W. Brunton, J. N. Kutz, and S. L. Brunton, "Learning dominant physical processes with data-driven balance models," Nat. Commun. **12**, 1016 (2021).
[15]J. Bakarji, J. Callaham, S. L. Brunton, and J. N. Kutz, "Dimensionally consistent learning with buckingham Pi," Nat. Comput. Sci. **2**, 834–844 (2022).

[16]K. Duraisamy, G. Iaccarino, and H. Xiao, "Turbulence modeling in the age of data," Annu. Rev. Fluid Mech. **51**, 357–377 (2019).
[17]S. L. Brunton, B. R. Noack, and P. Koumoutsakos, "Machine learning for fluid mechanics," Annu. Rev. Fluid Mech. **52**, 477–508 (2020).
[18]R. Vinuesa and S. L. Brunton, "Enhancing computational fluid dynamics with machine learning," Nat. Comput. Sci. **2**, 358–366 (2022).
[19]N. Zehtabiyan-Rezaie, A. Iosifidis, and M. Abkar, "Data-driven fluid mechanics of wind farms: A review," J. Renewable Sustainable Energy **14**, 032703 (2022).
[20]C. M. Legaard, T. Schranz, G. Schweiger, J. Drgoňa, B. Falay, C. Gomes, A. Iosifidis, M. Abkar, and P. G. Larsen, "Constructing neural network-based models for simulating dynamical systems," ACM Comput. Surv. **55**, 1–34 (2021).
[21]P. A. Durbin, "Some recent developments in turbulence closure modeling," Annu. Rev. Fluid Mech. **50**, 77–103 (2018).
[22]H. J. Bae and P. Koumoutsakos, "Scientific multi-agent reinforcement learning for wall-models of turbulent flows," Nat. Commun. **13**, 1443 (2022).
[23]J.-L. Wu, H. Xiao, and E. Paterson, "Physics-informed machine learning approach for augmenting turbulence models: A comprehensive framework," Phys. Rev. Fluids **3**, 074602 (2018).
[24]X. Zhang, J. Wu, O. Coutier-Delgosha, and H. Xiao, "Recent progress in augmenting turbulence models with physics-informed machine learning," J. Hydrodyn. **31**, 1153–1158 (2019).
[25]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).
[26]Z. Deng, C. He, Y. Liu, and K. C. Kim, "Super-resolution reconstruction of turbulent velocity fields using a generative adversarial network-based artificial intelligence framework," Phys. Fluids **31**, 125111 (2019).
[27]C. J. Lapeyre, A. Misdariis, N. Cazard, D. Veynante, and T. Poinsot, "Training convolutional neural networks to estimate turbulent sub-grid scale reaction rates," Combust. Flame **203**, 255–264 (2019).
[28]J. Duvall, K. Duraisamy, and S. Pan, "Discretization-independent surrogate modeling over complex geometries using hypernetworks and implicit representations," arXiv:2109.07018 (2021).
[29]G. Novati, H. L. de Laroussilhe, and P. Koumoutsakos, "Automating turbulence modelling by multi-agent reinforcement learning," Nat. Mach. Intell. **3**, 87–96 (2021).
[30]J. Kim, H. Kim, J. Kim, and C. Lee, "Deep reinforcement learning for large-eddy simulation modeling in wall-bounded turbulence," Phys. Fluids **34**, 105132 (2022).
[31]S. Verma, G. Novati, and P. Koumoutsakos, "Efficient collective swimming by harnessing vortices through deep reinforcement learning," Proc. Natl. Acad. Sci. U. S. A. **115**, 5849–5854 (2018).
[32]T.-R. Xiang, X. Yang, and Y.-P. Shi, "Neuroevolution-enabled adaptation of the Jacobi method for Poisson's equation with density discontinuities," Theor. Appl. Mech. Lett. **11**, 100252 (2021).
[33]K. Duraisamy, "Perspectives on machine learning-augmented Reynolds-averaged and large eddy simulation models of turbulence," Phys. Rev. Fluids **6**, 050504 (2021).
[34]A. Vadrot, X. I. Yang, and M. Abkar, "A survey of machine learning wall models for large eddy simulation," arXiv:2211.03614 (2022).
[35]X. I. A. Yang, G. I. Park, and P. Moin, "Log-layer mismatch and modeling of the fluctuating wall stress in wall-modeled large-eddy simulations," Phys. Rev. Fluids **2**, 104601 (2017).
[36]S. Kawai and J. Larsson, "Wall-modeling in large eddy simulation: Length scales, grid resolution, and accuracy," Phys. Fluids **24**, 015105 (2012).
[37]U. Schumann, "Subgrid scale model for finite difference simulations of turbulent flows in plane channels and annuli," J. Comput. Phys. **18**, 376–404 (1975).
[38]X. I. A. Yang and Y. Lv, "A semi-locally scaled eddy viscosity formulation for LES wall models and flows at high speeds," Theor. Comput. Fluid Dyn. **32**, 617–627 (2018).
[39]P. E. Chen, Y. Lv, H. H. Xu, Y. Shi, and X. I. A. Yang, "LES wall modeling for heat transfer at high speeds," Phys. Rev. Fluids **7**, 014608 (2022).
[40]H. H. Xu, X. I. A. Yang, and P. M. Milani, "Assessing wall-modeled large-eddy simulation for low-speed flows with heat transfer," AIAA J. **59**, 2060–2069 (2021).
[41]J. Larsson, S. Kawai, J. Bodart, and I. Bermejo-Moreno, "Large eddy simulation with modeled wall-stress: Recent progress and future directions," Mech. Eng. Rev. **3**, 15–00418–00418 (2016).
[42]C. Hansen, X. I. Yang, and M. Abkar, "POD-mode-augmented wall model and its applications to flows at non-equilibrium conditions," arXiv:2301.06803 (2023).
[43]M. Fowler, T. A. Zaki, and C. Meneveau, "A Lagrangian relaxation towards equilibrium wall model for large eddy simulation," J. Fluid Mech. **934**, A44 (2022).
[44]X. Yang, J. Sadique, R. Mittal, and C. Meneveau, "Integral wall model for large eddy simulations of wall-bounded turbulent flows," Phys. Fluids **27**, 025112 (2015).
[45]G. I. Park and P. Moin, "An improved dynamic non-equilibrium wall-model for large eddy simulation," Phys. Fluids **26**, 015108–015148 (2014).
[46]S. T. Bose and P. Moin, "A dynamic slip boundary condition for wall-modeled large-eddy simulation," Phys. Fluids **26**, 015104 (2014).
[47]H. J. Bae, A. Lozano-Durán, S. T. Bose, and P. Moin, "Dynamic slip wall model for large-eddy simulation," J. Fluid Mech. **859**, 400–432 (2019).
[48]E. Perlman, R. Burns, Y. Li, and C. Meneveau, "Data exploration of turbulence simulations using a database cluster," in *Proceedings of the ACM/IEEE Conference on Supercomputing* (IEEE, 2007), pp. 1–11.
[49]J. Graham, K. Kanov, X. Yang, M. Lee, N. Malaya, C. Lalescu, R. Burns, G. Eyink, A. Szalay, R. Moser *et al.*, "A web services accessible database of turbulent channel flow and its use for testing a new integral wall model for LES," J. Turbul. **17**, 181–215 (2016).
[50]M. Lee and R. D. Moser, "Direct numerical simulation of turbulent channel flow up to," J. Fluid Mech. **774**, 395–415 (2015).
[51]X. L. Huang, X. I. A. Yang, and R. F. Kunz, "Wall-modeled large-eddy simulations of spanwise rotating turbulent channels: Comparing a physics-based approach and a data-based approach," Phys. Fluids **31**, 125105 (2019).
[52]X. L. Huang and X. I. A. Yang, "A Bayesian approach to the mean flow in a channel with small but arbitrarily directional system rotation," Phys. Fluids **33**, 015103 (2021).
[53]Z. Zhou, G. He, and X. Yang, "Wall model based on neural networks for LES of turbulent flows over periodic hills," Phys. Rev. Fluids **6**, 054610 (2021).
[54]Z. Zhou, X. I. A. Yang, F. Zhang, and X. Yang, "A wall model trained using the periodic hill data and the law of the wall," arXiv:2211.03614 (2022).
[55]D. Zhou, M. Whitmore, K. Griffin, and J. Bae, "Multi-agent reinforcement learning for wall models in LES of flow over periodic hills," in Center for Turbulence Research Summer Program Proceeding, 2022.
[56]A. Lozano-Durán and H. J. Bae, "Self-critical machine-learning wall-modeled LES for external aerodynamics," arXiv:2012.10005 (2020).
[57]R. Bhaskaran, R. Kannan, B. Barr, and S. Priebe, "Science-guided machine learning for wall-modeled large eddy simulation," in *IEEE International Conference on Big Data* (IEEE, 2021), pp. 1809–1816.
[58]S. Radhakrishnan, L. A. Gyamfi, A. Miró, B. Font, J. Calafell, and O. Lehmkuhl, "A data-driven wall-shear stress model for LES using gradient boosted decision trees," in *International Conference on High Performance Computing* (Springer, 2021), pp. 105–121.
[59]N. Moriya, K. Fukami, Y. Nabae, M. Morimoto, T. Nakamura, and K. Fukagata, "Inserting machine-learned virtual wall velocity for large-eddy simulation of turbulent channel flows," arXiv:2106.09271 (2021).
[60]Y. Bin, L. Chen, G. Huang, and X. I. A. Yang, "Progressive, extrapolative machine learning for near-wall turbulence modeling," Phys. Rev. Fluids **7**, 084610 (2022).
[61]See https://lesgo.me.jhu.edu for "LESGO: A Parallel Pseudo-Spectral Large-Eddy Simulation Code."
[62]F. Porté-Agel, C. Meneveau, and M. B. Parlange, "A scale-dependent dynamic model for large-eddy simulation: Application to a neutral atmospheric boundary layer," J. Fluid Mech. **415**, 261–284 (2000).
[63]M. Abkar and F. Porté-Agel, "A new boundary condition for large-eddy simulation of boundary-layer flow over surface roughness transitions," J. Turbul. **13**, N23 (2012).
[64]M. Abkar and F. Porté-Agel, "Influence of atmospheric stability on wind-turbine wakes: A large-eddy simulation study," Phys. Fluids **27**, 035104 (2015).
[65]X. I. A. Yang and M. Abkar, "A hierarchical random additive model for passive scalars in wall-bounded flows at high Reynolds numbers," J. Fluid Mech. **842**, 354–380 (2018).

[66]X. I. Yang, S. Pirozzoli, and M. Abkar, "Scaling of velocity fluctuations in statistically unstable boundary-layer flows," J. Fluid Mech. **886**, A3 (2020).
[67]X. I. A. Yang, P. E. Chen, R. Hu, and M. Abkar, "Logarithmic-linear law of the streamwise velocity variance in stably stratified boundary layers," Boundary-Layer Meteorol. **183**, 199–213 (2022).
[68]E. Bou-Zeid, C. Meneveau, and M. Parlange, "A scale-dependent Lagrangian dynamic model for large eddy simulation of complex turbulent flows," Phys. Fluids **17**, 025105 (2005).
[69]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, https://doi.org/10.1029/2005WR003989 (2006).
[70]M. Abkar, H. J. Bae, and P. Moin, "Minimum-dissipation scalar transport model for large-eddy simulation of turbulent flows," Phys. Rev. Fluids **1**, 041701 (2016).
[71]M. Abkar and P. Moin, "Large-eddy simulation of thermally stratified atmospheric boundary-layer flow using a minimum dissipation model," Boundary-Layer Meteorol. **165**, 405–419 (2017).
[72]J. Smagorinsky, "General circulation experiments with the primitive equations: I. The basic experiment," Mon. Weather Rev. **91**, 99–164 (1963).
[73]M. Germano, U. Piomelli, P. Moin, and W. H. Cabot, "A dynamic subgrid-scale eddy viscosity model," Phys. Fluids **3**, 1760–1765 (1991).
[74]C. Meneveau, T. S. Lund, and W. H. Cabot, "A Lagrangian dynamic subgrid-scale model of turbulence," J. Fluid Mech. **319**, 353–385 (1996).
[75]W. Rozema, H. J. Bae, P. Moin, and R. Verstappen, "Minimum-dissipation models for large-eddy simulation," Phys. Fluids **27**, 085107 (2015).
[76]X. Yang, S. Bose, and P. Moin, "A physics-based interpretation of the slip-wall LES model," *Center for Turbulence Research, Annual Briefs* (Stanford University, 2016), pp. 65–74.
[77]C. Hansen, M. Whitmore, M. Abkar, and X. I. A. Yang, "POD-mode-augmented wall model and its applications to flows at non-equilibrium conditions," in Center for Turbulence Research Summer Program Proceeding, 2022.
[78]C.-H. Moeng, "A large-eddy-simulation model for the study of planetary boundary-layer turbulence," J. Atmos. Sci. **41**, 2052–2062 (1984).
[79]I. Marusic, J. P. Monty, M. Hultmark, and A. J. Smits, "On the logarithmic region in wall turbulence," J. Fluid Mech. **716**, R3 (2013).
[80]G. Novati and P. Koumoutsakos, "Remember and forget for experience replay," in Proceedings of the 36th International Conference on Machine Learning, 2019.
[81]H. M. Nagib and K. A. Chauhan, "Variations of von Kármán coefficient in canonical flows," Phys. Fluids **20**, 101518 (2008).
[82]M. Hultmark, M. Vallikivi, S. C. C. Bailey, and A. Smits, "Turbulent pipe flow at extreme Reynolds numbers," Phys. Rev. Lett. **108**, 094501 (2012).
[83]C. Diaz-Daniel, S. Laizet, and J. C. Vassilicos, "Wall shear stress fluctuations: Mixed scaling and their effects on velocity fluctuations in a turbulent boundary layer," Phys. Fluids **29**, 055102 (2017).
[84]C. J. Maddison, A. Huang, I. Sutskever, and D. Silver, "Move evaluation in go using deep convolutional neural networks," arXiv:1412.6564 (2014).