

RESEARCH ARTICLE

Modeling the wall shear stress in large-eddy simulation using graph neural networks

Dorian Dupuy^{1,*} , Nicolas Odier¹, Corentin Lapeyre¹ and Dimitrios Papadogiannis^{2,*}

¹European Centre for Research and Advanced Training in Scientific Computing, Toulouse F-31057 Cedex 1, France

²Safran Tech, Magny-Les-Hameaux, France

*Corresponding authors. E-mails: lapeyre@cerfacs.fr; dimitrios.papadogiannis@safrangroup.com

Received: 31 August 2022; **Revised:** 22 January 2023; **Accepted:** 04 February 2023

Keywords: Computational fluid dynamics; graph neural networks; large-eddy simulation; wall modeling

Abstract

As the Reynolds number increases, the large-eddy simulation (LES) of complex flows becomes increasingly intractable because near-wall turbulent structures become increasingly small. Wall modeling reduces the computational requirements of LES by enabling the use of coarser cells at the walls. This paper presents a machine-learning methodology to develop data-driven wall-shear-stress models that can directly operate, a posteriori, on the unstructured grid of the simulation. The model architecture is based on graph neural networks. The model is trained on a database which includes fully developed boundary layers, adverse pressure gradients, separated boundary layers, and laminar–turbulent transition. The relevance of the trained model is verified a posteriori for the simulation of a channel flow, a backward-facing step and a linear blade cascade.

Impact Statement

Numerical simulation is an important tool for design optimization in various industries, notably in the transport and aeronautics market. The development of graph-based machine-learning wall-shear-stress models for large-eddy simulation could improve the accuracy of practical industrial simulations, paving the way for a reduction in the energy consumption of aircraft, cars, and ships, resulting in lower emissions and noise.

1. Introduction

This paper investigates the modeling of the wall shear stress (WSS) in large-eddy simulations (LESs) using a graph neural network (GNN) that directly operates on the mesh of the simulation. In LESs of turbulent flows, wall modeling enables the use of coarser meshes, that are not able to resolve the velocity gradients in the inner viscous and buffer regions of the near-wall boundary layers. This ability relieves the grid-point and time-step requirements of LES as the Reynolds number increases (Yang and Griffin, 2021), and is key to the LES of complex industrial flows. Indeed, the use of a no-slip boundary condition requires an accurate resolution of the entire boundary layer, including the viscous sublayer. While LESs do not require the direct computation of the smaller turbulence length scales, even the most energetic integral length scales decrease in size near the walls hence preventing further coarsening in the wall-normal direction and constraining the time step of the simulation. This is often intractable in complex flows and in flows at high Reynolds number, as the number of grid points required to resolve the near-wall region

becomes prohibitive. Wall models have been reviewed in Larsson et al. (2016) and Bose and Park (2018). The most common and classical approach is the algebraic equilibrium wall model, which imposes the WSS as a Neumann-type boundary condition by enforcing the law of the wall locally (Deardorff, 1970; Schumann, 1975; Piomelli et al., 1989). This assumption is relevant in fully developed boundary layers, for which the law of the wall holds in the mean, but cannot generally be assumed to hold in complex flows, notably in the presence of strong adverse pressure gradients, curvature, separated, or transitional boundary layers.

Several alternative wall models have been proposed in the literature. Bose and Moin (2014) and Bae et al. (2019) suggested a slip wall model that provides a boundary condition for the wall-parallel velocity. Chung and Pullin (2009), Inoue and Pullin (2011), Saito et al. (2012), and Gao et al. (2019) have coupled a stretched-vortex subgrid-scale model with a virtual wall model that terminates the LES domain at a finite distance from the wall. A large class of methods models the WSS using a strategy that combines LES and Reynolds-average Navier–Stokes (RANS) simulation. This includes zonal methods that solve the RANS or thin-boundary-layer equations (Cabot, 1995; Balaras et al., 1996; Baurle et al., 2003; Davidson and Peng, 2003; Temmerman et al., 2005; Piomelli, 2008). This is computationally expensive since the grid resolution requirements are not necessarily lower than would be in a wall-resolved LES, although the boundary-layer equations are less expensive to resolve. The integral wall model introduced by Yang et al. (2015) and extended by Catchirayer et al. (2018) use a parameterized velocity profile to avoid the numerical integration of the boundary-layer equations. Wall models based on control theory have also been suggested, in which the WSS is imposed so as to lead to the target mean velocity profile, either given by the law of the law or a coupled RANS model (Nicoud et al., 2001; Templeton et al., 2006; 2008). Recently, the use machine-learning methodologies to model the WSS has finally been proposed by several authors. Yang et al. (2019) developed a data-driven wall-stress model in a turbulent channel flow. Huang et al. (2019) compared data-driven and physics-based approaches in a spanwise rotating channel. Zhou et al. (2021) assessed a data-driven wall model for the turbulent flow over periodic hills. Lozano-Durán and Bae (2020) applied a self-critical machine-learning wall model to a wing-fuselage juncture flow. Zangeneh (2021) developed a data-driven wall model for supersonic separated flows. Bhaskaran et al. (2021) has trained a wall model for a boundary layer in the presence of shock-boundary layer interaction. Bae and Koumoutsakos (2022) assessed a multi-agent reinforcement learning approach to wall modeling in a turbulent channel flow. Radhakrishnan et al. (2021) developed a wall model based on gradient boosted decision trees. Dupuy et al. (2022) investigated the data-driven wall modeling of turbulent separated flows. These efforts may be seen as part of a broader movement to introduce data-driven methods to the field of turbulence simulation and modeling (Duraisamy et al., 2019; Brunton et al., 2020). This has been motivated by the recent emergence of data-driven approaches with capability beyond those of handcrafted models in various complex tasks, such as image classification, speech recognition, natural language processing, short-term weather prediction, or protein structure prediction (LeCun et al., 2015; Jumper et al., 2021; Ravuri et al., 2021). Supervised machine-learning wall-shear-stress models have proven capable of discriminating and operating in non-equilibrium flow regions, provided that similar flow phenomena were encountered in the training dataset (Dupuy et al., 2022).

The studies on machine-learning wall-shear-stress modeling have demonstrated the advantages of going beyond algebraic formulations and capturing topological flow features to improve the wall friction prediction. However, a challenge with nonlocal models is defining this topological extraction in a general manner that applies to any complex geometry. As an example, multipoint stencils might exit the flow domain in small nooks or sharp concave corners. The present study proposes a general formulation that is adaptable to any mesh, without any concern for how to implement it geometrically, by exploiting GNNs that rely on the data mesh itself. This work directly allows the use of data on unstructured meshes similarly to Lozano-Durán and Bae (2020), which is the first study to apply a machine-learning wall-shear-stress model in an unstructured-grid simulation. In their study, the input data from the flow simulation are interpolated on a local Cartesian grid to feed the inputs of the wall-shear-stress model, arranged in a predefined stellated arrangement with uniform spacing. By contrast, the present machine-learning model is a topologically aware wall-shear-stress model that directly operates on the unstructured mesh of the

simulation. This is achieved by using a GNN wherein the mesh of the simulation is represented as a graph. Neural networks operating on graphs were first used by Gori et al. (2005) and have demonstrated their effectiveness in various tasks, for instance, visual scene understanding, multi-agent systems, relational reasoning, or graph classification (Scarselli et al., 2008; Bruna et al., 2013; Sukhbaatar et al., 2016; Bronstein et al., 2017; Gilmer et al., 2017; Raposo et al., 2017; Santoro et al., 2017; Kipf et al., 2018). GNNs were also used by various authors to learn the dynamics of physical systems, including fluid flows (Battaglia et al., 2016; Chang et al., 2016; Watters et al., 2017; Sanchez-Gonzalez et al., 2018; Van Steenkiste et al., 2018; Pfaff et al., 2020; Chen et al., 2021; Gao et al., 2022). A review may be found in Battaglia et al. (2018) or Zhou et al. (2020). The relevance of GNNs to wall-shear-stress modeling can be hypothesized from its potential ability to detect geometric features, such as angles and curvature. In addition, the architecture of GNNs encodes some physical assumptions that are relevant for a wall-shear-stress model and therefore may ease the training process and increase the generalizability of the model, such as translational invariance and an inherent bias toward spatial locality.

The ability of a GNN to model the WSS is investigated in this paper using both a priori tests, based on filtered numerical data, and a posteriori tests, based on LESs implementing such models. A database of seven direct numerical simulations or high-fidelity wall-resolved LESs is used to train the model and validate the model: two channel-flow simulations, the simulation of the flow in a diffuser, two simulations of flows over backward-facing steps (BFSs) and two simulations of flows in a linear blade cascade. This provides a variety of flow phenomena, including fully developed boundary layers, adverse pressure gradients, separated boundary layers, and laminar–turbulent transition. Instantaneous snapshots from each simulation are filtered onto coarse meshes representative of a LES with a wall-shear-stress model, of which only near-wall nodes are used by the model for its prediction. The analysis is restricted to coarse tetrahedral meshes, for their ubiquity in simulations of complex industrial flows, although the method could in principle be applied to any type of mesh. An Encode-Process-Decode architecture (Battaglia et al., 2018) is used for the GNN, with no global features to ensure the spatial locality of the prediction. The inputs of the model are the displacement vector of the mesh edges and the three components of velocity at each node, scaled to ensure the Mach number invariance of the model at the incompressible limit, and augmented by arbitrary rotations and reflections of the coordinate system. The model is implemented a posteriori by coupling the compressible massively parallel flow solver AVBP (Schönfeld and Rudgyard, 1999) to a code dedicated to inference, using a message passing interface (MPI; Serhani et al., 2022). A channel flow, a BFS and a linear blade cascade configuration are simulated for these a posteriori tests. In each case, the simulations are compared to the corresponding simulation using an algebraic model based on the law of the wall to assess the relevance of the model.

The paper is organized as follows. The dataset and the preparation of the data for the machine-learning model is presented in Section 2. The architecture of the GNN is described in Section 3. The results are analyzed and discussed in Section 4.

2. Dataset and Data Preparation

2.1. Dataset

The machine-learning wall-shear-stress model developed in this paper relies on a database of direct numerical simulations or high-fidelity wall-resolved LESs which have been produced by various research groups (Figure 1). Namely, the database includes the simulations of: a fully developed channel flow at friction Reynolds number $Re_\tau = 180$ (CF1), performed at Imperial College London (Agostini and Vincent, 2020) using the high-order flux reconstruction method of Huynh (2007) in the CFD code PyFR (Witherden et al., 2014); a fully developed channel flow at friction Reynolds number $Re_\tau = 950$ (CF2), performed at the Polytechnic University of Madrid (Del Álamo and Jiménez, 2003; Lozano-Durán and Jiménez, 2014; 2015) using a hybrid Fourier–Chebyshev spectral method; a three-dimensional diffuser (3DD) corresponding to the geometry “Diffuser 1” of Cherry et al. (2008), performed at Barcelona Supercomputing Center (Ercoftac, 2022) using the low-dissipation finite element scheme of Lehmkuhl et al. (2019); a BFS, performed at CERFACS (Pouech et al., 2019; Pouech et al., 2021) using a cell-vertex

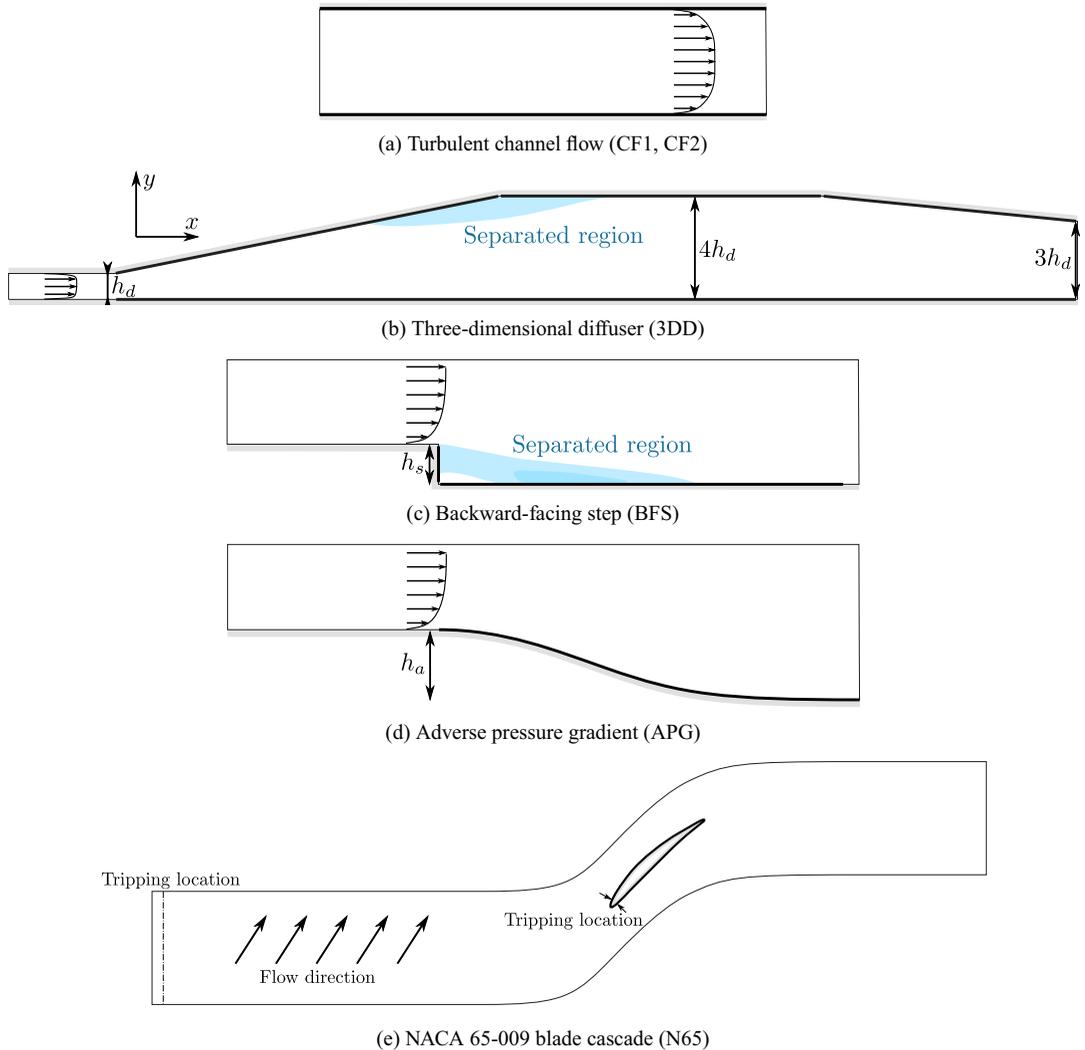


Figure 1. Geometry of the six simulations in the database. The blue regions are separated in the mean.

finite-element method (Schönfeld and Rudgyard, 1999 with second-order accurate convection and diffusion schemes (Lax and Wendroff, 1960); a curved BFS based on the geometry of Disotell and Rumsey (Disotell and Rumsey, 2017; Alaya et al., 2020), hereafter referred to as adverse-pressure-gradient (APG) simulation and performed at the University of Bergamo (Ercoftac, 2022) using discontinuous Galerkin method and a fifth order linearly implicit Rosenbrock scheme (Bassi et al., 2015; 2016); and a NACA 65-009 blade cascade (N65) such as studied experimentally by Ma et al. (2011) and Zambonini et al. (2017), performed using a cell-vertex finite-element method (Schönfeld and Rudgyard, 1999 and a two-step Taylor–Galerkin scheme (Colin and Rudgyard, 2000). The N65 case is subdivided into two sub-configurations that differ only by the inlet boundary condition: a simulation with an incidence angle of 4° (N65a) and a simulation with an incidence angle of 7° (N65b). The two simulations are described in Appendix C. The database is heterogeneous in terms of flow solvers and numerical methods since the 3DD, BFS, N65a, and N65b simulations were performed using finite-element methods,

Table 1. Numerical parameters of the numerical simulation included in the a priori database.

Simulation	Numerical method	Mesh size
Channel flow $Re_\tau = 180$ (CF1)	High-order flux reconstruction	0.07 M cells, \mathbb{P}_5 (\times)
Channel flow $Re_\tau = 950$ (CF2)	Hybrid Fourier–Chebyshev	100 M cells
Three-dimensional diffuser (3DD)	Low-dissipation FE scheme	250 M cells
Backward-facing step (BFS)	Second-order FE scheme	40 M cells
Adverse pressure gradient (APG)	Implicit high-order DG scheme	15 M cells, \mathbb{P}_3 (+)
[App. A] NACA 65-009 blade (N65a)	Taylor–Galerkin FE scheme	273 M cells
[App. A] NACA 65-009 blade (N65b)	Taylor–Galerkin FE scheme	273 M cells

Note. The acronym FE denotes a finite element method, DG a discontinuous Galerkin method, and \mathbb{P}_n indicates the use of n th order polynomials in the case of high-order methods. Note that due to the use of high-order polynomial, the effective number of solution points is (\times) 15 M in the CF1 simulation and (+) 960 M in the APG simulation.

the CF1 and APG simulations using high-order methods and the CF2 simulation using a spectral method. We believe that data-driven modeling approaches should demonstrate their ability to learn from such heterogeneous databases because heterogeneous databases can more easily scale and cover a large diversity of flow configurations. The numerical parameters of each simulation are summarized in Table 1.

A wide variety of flow behavior is represented by the dataset. The CF1 and CF2 simulations are focused on providing data on fully developed boundary layers. The 3DD and BFS simulations include fully separated flow regions. The APG simulation includes regions of incipient separation, that is with a backflow probability between 1 and 20% (Simpson, 1989). The N65a and N65b simulations include transitional boundary layers, through small-scale flow separation, on the blade surface, a light adverse pressure gradient as well as a large region of unsteady three-dimensional corner separation on the junction between the endwall and the suction side of the blade. For a given model to produce accurate predictions in all simulations, an ability to discriminate between these various flow regimes will thus be required.

2.2. Data preparation

The instantaneous fields of the a priori database need to be preprocessed for use by the machine-learning procedure. Namely, the fields are filtered onto coarse meshes representative of a LES with a wall-shear-stress model, as is customary in a priori tests for LES (Leonard, 1974; Sagaut, 2006). We focus on tetrahedral meshes, for their simplicity, their ability to represent arbitrarily complex domains and widespread use in simulations of complex industrial flows. The preparation of the data for the machine-learning algorithm is represented in Figure 1 and is as follows:

1. For each simulation, a series of coarse tetrahedral meshes are constructed using the mesh adaptation library MMG3D (Dobrzynski and Frey, 2008; Dapogny et al., 2014). All coarse tetrahedral meshes have a uniform target refinement, although there are local variations of cell sizes resulting from the meshing algorithm, since regular tetrahedra do not fill space (as may be seen in Figure 2). To select the refinement level, a particular flow region is used to define the nominal WSS of each simulation. Those are: the channel walls in the simulations CF1 and CF2, the bottom wall upstream of the expansion in the 3DD simulation, the boundary layer upstream of the step in the APG and BFS simulations, and the boundary layer upstream of the blade in the N65a and N65b simulations. Coarse tetrahedral meshes are produced for 12 different refinement levels for each simulation, with a nominal edge length in wall units, in the regions listed above, in the range $25 \leq e^+ \leq 80$. This implies that the trained GNN can only operate on meshes that are within this range of mesh refinement in terms of scaled edge length.
2. For each coarse tetrahedral mesh, the neighborhood of the target walls is selected, by preserving only nodes separated from the target walls \mathcal{W} by N_H edges or less. The resulting tetrahedral meshes

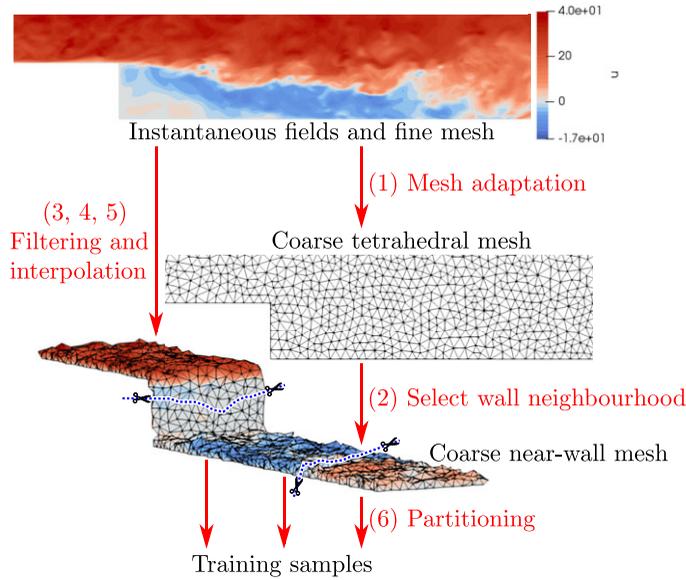


Figure 2. Preprocessing steps required to prepare the data for the training procedure.

are conformal and referred to as coarse near-wall meshes hereafter. Unless otherwise specified, $N_H = 3$ is used.

3. A fine tetrahedral mesh is produced, associated with a nominal edge length $e^+ = 4$. The instantaneous fields of the a priori database are interpolated linearly onto this fine mesh. The resulting fields are used as source data for the filtering operation.
4. The instantaneous data are filtered onto and at the resolution of the coarse near-wall meshes. This low-pass filtering operation attenuates large frequencies, associated with turbulent scales that cannot be resolved using the target computational grid. A surface filter is used to filter the WSS and a volume filter is used to filter the velocity components in the bulk of the domain. We denote $(\bar{\cdot})$ the volume filter associated with the local filter width $\tilde{\Delta}_{p_0} = \tilde{V}_{p_0}^{1/3}$ and $(\tilde{\cdot}^S)$ the surface filter associated with the local filter width $\tilde{\Delta}_{p_0}^S = \tilde{S}_{p_0}^{1/2}$, with \tilde{V}_{p_0} and \tilde{S}_{p_0} , respectively, the nodal volume of node p_0 and the nodal area of p_0 on the surface \mathfrak{W} . The volume and surface filters are truncated Gaussian filters weighted by the local cell volume or face area,

$$\bar{\phi}(p_0) = \sum_{p \in \mathfrak{w}_{p_0}} C_{p_0} V_p e^{-\frac{r_p - r_{p_0}}{2(\sigma_{p_0})^2}} \phi(p), \tag{1}$$

$$\tilde{\phi}^S(p_0) = \sum_{p \in \mathfrak{w}_{p_0}^S} C_{p_0}^S S_p e^{-\frac{r_p - r_{p_0}}{2(\sigma_{p_0}^S)^2}} \phi(p), \tag{2}$$

with C_{p_0} and $C_{p_0}^S$ normalization constants, selected such that $\bar{a} = \bar{a}^S = a$ for any constant a , V_p , and S_p , respectively, the nodal volume and face area associated with the source fine mesh and $\sigma_{p_0} = \tilde{\Delta}_{p_0} / \sqrt{12}$ and $\sigma_{p_0}^S = \tilde{\Delta}_{p_0}^S / \sqrt{12}$, respectively, the standard deviations of the volume and surface Gaussian kernels, following Leonard (1974). The volume filter window \mathfrak{w}_{p_0} associated with a point p_0 of the target coarse near-wall mesh is restricted in the wall-normal direction to avoid biasing the mean velocity profile:

$$\mathfrak{w}_{p_0} = \left\{ p \in \mathfrak{P}_f : \|r_p - r_{p_0}\| \leq \tilde{\Delta}_{p_0} \wedge |D_{\mathfrak{W},p} - D_{\mathfrak{W},p_0}| \leq \frac{1}{2} \tilde{\Delta}_{1,p_0} \right\}, \tag{3}$$

with \mathfrak{P}_f the set of points of the source mesh, \mathbf{r}_p the position vector associated with point p , and $D_{\mathfrak{W},p}$ the shortest distance between p and the target walls \mathfrak{W} . The window $\mathfrak{w}_{p_0}^S = \left\{ p \in \mathfrak{P}_f : \|\mathbf{r}_p - \mathbf{r}_{p_0}\| \leq \tilde{\Delta}_{p_0}^S \wedge D_{\mathfrak{W},p} = 0 \right\}$ of the surface filter is isotropic.

5. The Gaussian filter defined above leads to a small bias on the velocity profile. In particular, the mean Gaussian-filtered velocity is slightly lower than mean unfiltered velocity in the case of a fully developed boundary layer. This bias is compensated for by correcting the filtered velocity as follows:

$$\bar{\mathbf{u}}^* = \bar{\mathbf{u}} + \langle \mathbf{u} \rangle - \langle \bar{\mathbf{u}} \rangle, \tag{4}$$

with $\bar{\mathbf{u}}^*$ the corrected filtered velocity and where $\langle \mathbf{u} \rangle$ and $\langle \bar{\mathbf{u}} \rangle$ are assessed using the time-average fields from each simulation.

6. The resulting data are partitioned into contiguous chunks of similar size using the graph partitioning library METIS (Karypis and Kumar, 1998). This helps producing mini-batches of consistent size to fit in the dynamic memory of the graphics processing unit (GPU) during training. In addition, subdividing the data into smaller chunks allows for more diversity in each mini-batch, by aggregating small regions of different simulations, which reduces noise in the optimization process.

3. Graph Neural Network

The wall-shear-stress model developed in this paper is a GNN, and more specifically uses an Encode-Process-Decode architecture (Battaglia et al., 2018). The GNNs used in the present study are feedforward deep learning models that aim to exploit the structure of the data in high dimension to predict the shear stress at the wall. Indeed, GNNs can involve a relatively large number of points in the prediction, which immerses the learning problem in a high-dimensional space. The high-dimensional nature of the learning problem may reduce the inherent ambiguities of the WSS prediction, by allowing the model to discriminate between different types of local flow configurations. For instance, a minimal capacity would be needed to allow the model to identify that a boundary layer is separated (Dupuy et al., 2022). In addition, the use of information in a region for the prediction might allow the model to take into account the space displacement associated with regions of high correlations with the WSS (Boxho et al., 2022). More specifically, GNNs are used for their ability to work directly with unstructured mesh data, without requiring the prior interpolation of the data. This ability is important for LESs in complex geometries, with angles, corners, or curvature. Finally, the architecture of the GNNs is physically well suited to the modeling of the WSS. First, the model is biased toward spatial locality, such that nearby grid points will more easily influence the prediction than grid points that are far away. Second, the model is invariant to translation as the same learned function is applied everywhere in the domain. In addition to the physical assumptions related to the model architecture, the present learning process ensures the equivariance of the model to Galilean transformations, orthogonal transformations such as rotations or reflections, and “changes of scales,” hereafter referred to as Mach number equivariance. We furthermore assume that the flow is isothermal incompressible and obeys the incompressible Navier–Stokes equations without external body force. This is consistent with the selection of simulations in the database of Section 2.1, which involves neither significant temperature variations nor compressibility effects.

3.1. Model architecture

Each sample mesh produced in Section 2.2 is represented as a symmetric directed graph $\mathfrak{G} = (\mathfrak{V}, \mathfrak{E})$, where \mathfrak{V} is the set of nodes and \mathfrak{E} the set of directed edges. The number of nodes is denoted n_v and the number of edges n_e . Let $v_i \in \mathfrak{V}$ be the i th node of the graph and $e_k = (v_{S(k)}, v_{R(k)}) \in \mathfrak{E}$ an edge pointing from $v_{S(k)}$ to $v_{R(k)}$. The input of the model is the ordered pair $\mathbf{X} = (\mathbf{V}, \mathbf{E})$, where $\mathbf{V} \in \mathbb{R}^{n_v \times d_v}$ is the node

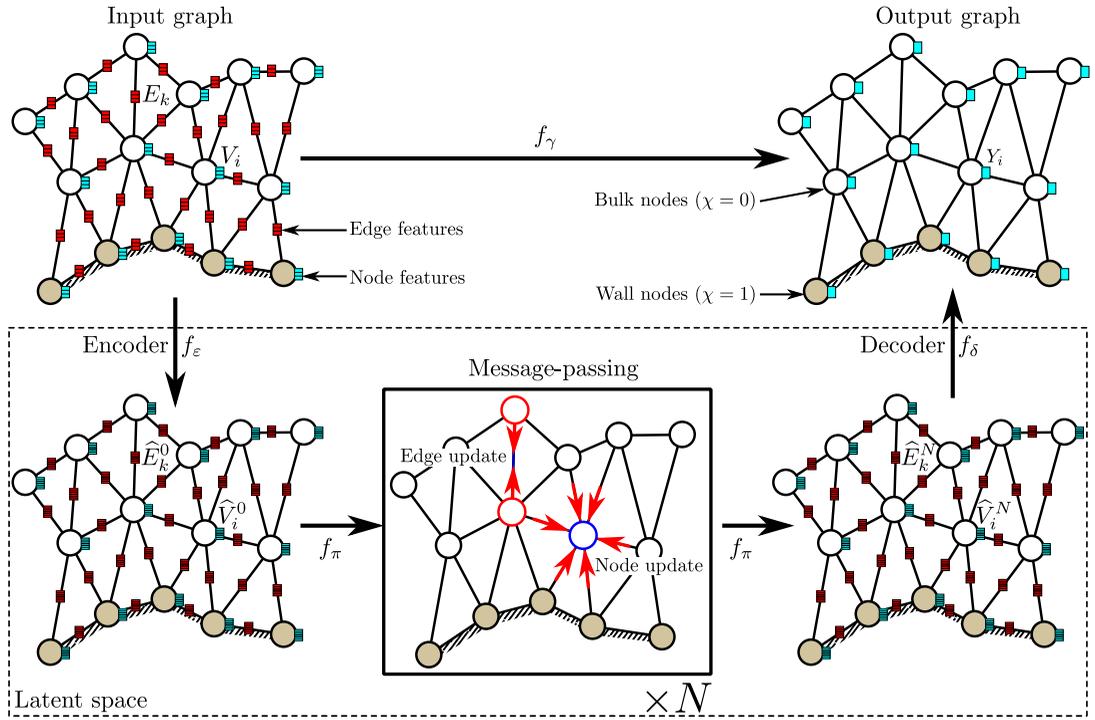


Figure 3. Graphical representation of the Encode-Process-Decode architecture.

input features, whose i th element $V_i \in \mathbb{R}^{d_v}$ represents the input features at node v_i , and $E \in \mathbb{R}^{n_e \times d_e}$ the edge input features, whose k th element $E_k \in \mathbb{R}^{d_e}$ represents the input features at edge e_k . The output of the model is the node vector $Y \in \mathbb{R}^{n_v \times 1}$. With an Encode-Process-Decode architecture (Battaglia et al., 2018), the output of the GNN may be expressed as

$$Y = f_\gamma(X) = f_\delta \circ f_\pi^N \circ \dots \circ f_\pi^2 \circ f_\pi^1 \circ f_\epsilon(X), \tag{5}$$

where $f_\epsilon, f_\pi^1, f_\pi^2, \dots, f_\pi^N$ and f_δ are the encoder, processor, and decoder, respectively (Figure 3). The architecture of the model is described below:

3.1.1. Encoder

The input features are encoded into a latent space of size $d_L = 128$ for both nodes and edges, $L^0 = (\hat{V}^0, \hat{E}^0) = f_\epsilon(X)$, with $\hat{V}^0 \in \mathbb{R}^{n_v \times d_L}$ and $\hat{E}^0 \in \mathbb{R}^{n_e \times d_L}$. The encoding is performed independently at each node or edge by $\hat{V}_i^0 = f_\epsilon^V(V_i)$ and $\hat{E}_k^0 = f_\epsilon^E(E_k)$, where f_ϵ^V and f_ϵ^E are two multilayer perceptrons.

3.1.2. Processor

The latent space is transformed by the successive application of $N = 4$ message-passing steps, with residual connections. The latent space after n steps, $L^n = (\hat{V}^n, \hat{E}^n) = L^{n-1} + f_\pi(L^{n-1})$, is given by $\hat{V}_i^n = \hat{V}_i^{n-1} + f_\pi^V(\hat{V}_i^{n-1}, \sum_{k|R(k)=i} \hat{E}_k^n)$ and $\hat{E}_k^n = \hat{E}_k^{n-1} + f_\pi^E(\hat{E}_k^{n-1}, \hat{V}_{S(k)}^{n-1}, \hat{V}_{R(k)}^{n-1})$, where f_π^V and f_π^E are two multilayer perceptrons.

3.1.3. Decoder

The latent space after the last message-passing step is decoded back to physical space at nodes only, $Y_i = f_\delta^V(\hat{V}_i^N)$, with f_δ^V a multilayer perceptron.

The same encoding, processing, and decoding functions are learned and applied everywhere in the graph. Hence, the message-passing steps in the processor will propagate the information through the edge and nodes to predict the WSS on every wall nodes.

The multilayer perceptrons $f_\varepsilon^V, f_\varepsilon^E, f_\pi^V$, and f_π^E each have $n_\ell = 2$ layers composed of d_L neural units and followed by layer normalization (Ba et al., 2016). The output of the i th neuron of hidden layer ℓ is given by

$$\bar{z}_\ell^{(i)} = \bar{w}_\ell^{(i)} \frac{z_\ell^{(i)} - \langle z_\ell^{(i)} \rangle_{\mathfrak{F}}}{\sqrt{\langle (z_\ell^{(i)} - \langle z_\ell^{(i)} \rangle_{\mathfrak{F}})^2 \rangle_{\mathfrak{F}}}} + \bar{b}_\ell^{(i)}, \tag{6}$$

where $\langle \cdot \rangle_{\mathfrak{F}}$ denotes an average across features and with

$$z_\ell^{(i)} = \sigma \left(\sum_j w_\ell^{(i,j)} \bar{z}_{\ell-1}^{(j)} + b_\ell^{(i)} \right), \tag{7}$$

where the weights and biases $\bar{w}_\ell^{(i)}$, $w_\ell^{(i,j)}$, $\bar{b}_\ell^{(i)}$, and $b_\ell^{(i)}$ are learnable parameters and σ the activation function, namely a rectified linear unit $\sigma(x) = \max(0, x)$ in the present work. The multilayer perceptron f_δ^V has $n_\ell + 1$ hidden layers, the first n_ℓ layers being as described above and the last layer a linear layer with one neural unit,

$$z_{n_\ell} = \sum_j w_{n_\ell}^{(j)} z_{(n_\ell-1)}^{(j)} + b_{n_\ell}. \tag{8}$$

The weights of the network are initialized using a truncated normal distribution, discarding and redrawing values beyond two standard deviations, and trained using the Adam optimizer (Kingma and Ba, 2015) with a base learning rate of 0.001. The loss function is the mean squared error (MSE) between the predicted WSS and the reference WSS,

$$\mathcal{L} = \langle \chi (\bar{\tau}_{\text{ref}}^S - Y)^2 \rangle_{\mathfrak{Q}}, \tag{9}$$

where $\langle \cdot \rangle_{\mathfrak{Q}}$ denotes an average across graph nodes and samples and χ a target boundary mask, with $\chi = 1$ on the target wall nodes \mathfrak{W} and $\chi = 0$ elsewhere. The training time is 20 hr using a single NVIDIA A30 GPU unit.

3.2. Feature selection

The input of the GNN WSS model is given by $d_v = 4$ node input features and $d_e = 4$ edge input features. The node input features are the target boundary mask χ and the three components of the scaled velocity vector $\check{\mathbf{u}}^*$, defined as

$$\check{\mathbf{u}}^* = \bar{\mathbf{u}}^* \langle \|\mathbf{e}\| \rangle_{\mathfrak{G}} / \langle \nu \rangle_{\mathfrak{G}}, \tag{10}$$

where $\langle \|\mathbf{e}\| \rangle_{\mathfrak{G}}$ is the average edge length of a given graph \mathfrak{G} and $\langle \nu \rangle_{\mathfrak{G}}$ the average kinematic viscosity. Note that the average viscosity is included in the scaling since, although all the flows considered are incompressible, some of the numerical methods used are based on the compressible Navier–Stokes equations, which results in negligible but nonzero variations of viscosity. Following Pfaff et al. (2020), the $d_e = 4$ edge input features are the three components of the scaled displacement vector of the edge $\check{\mathbf{e}}$, with $\check{\mathbf{e}} = \mathbf{e} / \langle \|\mathbf{e}\| \rangle_{\mathfrak{G}}$, and its magnitude $\|\check{\mathbf{e}}\|$. The edge and node input features are taken on the entire graph of the neural network. However, only a limited region will influence the prediction of the WSS at a given node. This region is the receptive field of the model, governed by the number of message-passing steps N as represented schematically in Figure 4. In accordance with the scaling of the input velocity (10), the target WSS is scaled as

$$\bar{\tau}_{\text{ref}}^S = \bar{\tau}_{\text{ref}}^S (\langle \|\mathbf{e}\| \rangle_{\mathfrak{G}} / \langle \nu \rangle_{\mathfrak{G}})^2 / \langle \rho \rangle_{\mathfrak{G}}, \tag{11}$$

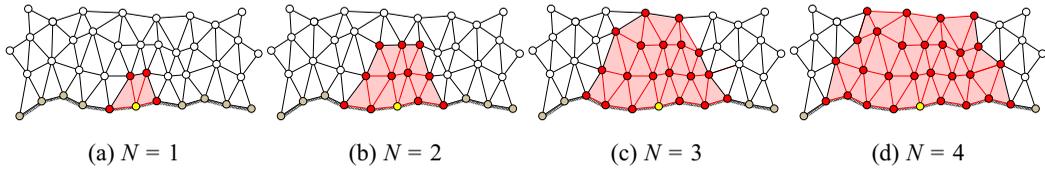


Figure 4. Effect of the number of message-passing steps N on the receptive field of the graph neural network (red shaded area) for the prediction of the wall shear stress on a given target node (yellow node). (a) $N = 1$. (b) $N = 2$. (c) $N = 3$. (d) $N = 4$.

Table 2. Strategies used to enforce the equivariance of the machine-learning model to various transformations.

Equivariance	Strategy
Orthogonal equivariance	Data augmentation
Galilean equivariance	Invariant input features
Mach number equivariance	Scaling of input features

where $\langle \rho \rangle_{\mathcal{G}}$ is the average density and with τ the norm of the shear stress vector $\boldsymbol{\tau} = \boldsymbol{\Sigma} \cdot \mathbf{e}_n - (\mathbf{e}_n \cdot \boldsymbol{\Sigma} \cdot \mathbf{e}_n) \mathbf{e}_n$, where $\boldsymbol{\Sigma} = \mu(\nabla \mathbf{u} + (\nabla \mathbf{u})^T - (2/3)(\nabla \cdot \mathbf{u}) \mathbf{I}_d)$ is the viscous stress tensor and \mathbf{e}_n a unit wall-normal vector.

The equivariance of the model under several transformations is ensured by the selection of input features, as reported in Table 2. The orthogonal equivariance of the model, that is its equivariance under rotations and reflections, is ensured by data augmentation: for each sample graph, a random three-dimensional rotation and reflection is applied onto the input edge displacement vectors and velocity vectors. To ensure that the predictions of the model do not depend on the coordinate system, the components of the scaled edge displacement vector $\check{\boldsymbol{\epsilon}}$ and scaled velocity vector $\check{\mathbf{u}}^*$ are expressed in a basis arbitrarily rotated and reflected for each graph. The Galilean invariance of the model is ensured by using the wall as an intrinsic velocity reference, that is, by using as input feature, the relative velocity of the flow with respect to the wall velocity. The Mach number equivariance of the model, or namely the equivariance of the model to a change of scale of velocity, density, viscosity, or length, is ensured by the scaling (\cdot) of the input and target features. Indeed, the scaled variables tend to a constant as the Mach number of the flow tends to zero, at constant Reynolds number and mesh resolution (Paolucci, 1982). In other words, for an isothermal incompressible flow without external body force, the scaled variables depend on the Reynolds number of the flow, which governs the flow physics, but are unaffected by a change of Mach number, which is not a relevant physical parameter. This scaling is a natural extension of the scaling used by Dupuy et al. (2022) for a Cartesian grid. It is purely local and only involves quantities that can be computed in a LES. Enforcing the equivariance of the model under these transformations increases the sample efficiency of each graph in the numerical dataset and improves the generalization of the model to different configurations.

4. Results

4.1. A priori tests

This section presents a priori tests of the machine-learning wall-shear-stress model, based solely on the numerical database described in Section 2.1. The model is trained using the simulations CF1, CF2, 3DD, and N65a. In each training simulation, a subset of the data is isolated for validation purpose. The split between training and validation datasets is performed independently for each simulation. In the CF1, CF2, and 3DD simulations, the snapshots associated with the later time steps are reserved for validation purpose. The BFS and APG simulations are only used as test simulations, to test the performance of the

Table 3. Simulations used for the training and validation of the graph neural network WSS model, for a priori tests and for a posteriori tests.

Simulation	Train/validation	A priori test	A posteriori test
Channel flow $Re_\tau = 180$ (CF1)	Yes	—	—
Channel flow $Re_\tau = 950$ (CF2)	Yes	—	Yes
Channel flow $Re_\tau = 2000$	—	—	Yes
Three-dimensional diffuser (3DD)	Yes	—	—
Backward-facing step (BFS)	—	Yes	Yes
Adverse pressure gradient (APG)	—	Yes	—
NACA 65-009 blade, $I = 4^\circ$ (N65a)	Yes	—	—
NACA 65-009 blade, $I = 7^\circ$ (N65b)	—	Yes	Yes

model on configurations not seen during training. For the N65 configuration, training is exclusively performed on the sub-configuration N65a, that is with an incidence angle $I = 4^\circ$, while testing is exclusively performed on the sub-configuration N65b, that is with an incidence angle $I = 7^\circ$. This is summarized in Table 3.

In order to assess a priori the relevance of the GNN WSS model, the behavior of the model in each simulation is first analyzed using $u^+(y^+)$ graph wherein the predicted local WSS is used to compute the wall-unit scaling ($^+$). The results are reported in Figure 5 along with corresponding data based on the reference WSS. Agreement or departure of both reference data and model prediction from fully developed equilibrium wall turbulence is emphasized by comparing the instantaneous data points to the analytical profile of Reichardt (1951),

$$u^+(y^+) = \frac{1}{\kappa} \log(1 + \kappa y^+) + 7.8 \left[1 - \exp\left(\frac{-y^+}{11}\right) - \frac{y^+}{11} \exp\left(\frac{-y^+}{3}\right) \right], \quad (12)$$

with $\kappa = 0.41$ the von Kármán constant. It should be emphasized that, except in the case of the channel-flow simulations, Reichardt's law is not a reference but rather as a baseline, indicative of what a model with insufficient capacity would predict. In the two channel-flow simulations (CF1 and CF2), the reference data points are scattered around the law of the wall, with a large variance associated with velocity and WSS fluctuations. This behavior is reproduced by the model, but the variance of the data points around Reichardt's law is clearly underestimated. In the 3DD simulation, most data points are scattered below Reichardt's law in the $u^+(y^+)$ graph, which is consistent with the presence of separated regions. The GNN WSS model reproduces this behavior, although the predictions are more biased toward the law of the wall and extreme points well above Reichardt's law are missing at low y^+ . The presence of data points that deviates from the law of the wall is also correctly predicted by the model in the test BFS and APG simulations, suggesting reliable generalization to other flow configurations with separation. Finally, the NACA 65-009 blade simulation (N65b) is most saliently associated with a transitional boundary layer on the blade surface, and most data points are found above Reichardt's law on the $u^+(y^+)$ graph. This is qualitatively reproduced by the model but with a clearly lower variance. Overall, the model thus leads to markedly different distributions of $u^+(y^+)$ in each simulation, indicating an ability to locally discriminate, to some extent, between the various type of boundary layers, fully developed, separated, or transitional, present in the numerical database.

Several quantitative measures of the agreement between the predicted WSS and corresponding local instantaneous reference WSS are given in Table 4. First, we use the concordance coefficient of Lin (1989) between the predicted WSS Y and the reference WSS τ_{ref} , defined as

$$C_\tau = \frac{\langle \tau_{\text{ref}} Y \rangle_{\mathcal{Q}} - \langle \tau_{\text{ref}} \rangle_{\mathcal{Q}} \langle Y \rangle_{\mathcal{Q}}}{\langle \tau_{\text{ref}}^2 \rangle_{\mathcal{Q}} - \langle \tau_{\text{ref}} \rangle_{\mathcal{Q}}^2 + \langle Y^2 \rangle_{\mathcal{Q}} - \langle Y \rangle_{\mathcal{Q}}^2 + (\langle \tau_{\text{ref}} \rangle_{\mathcal{Q}} - \langle Y \rangle_{\mathcal{Q}})^2}. \quad (13)$$

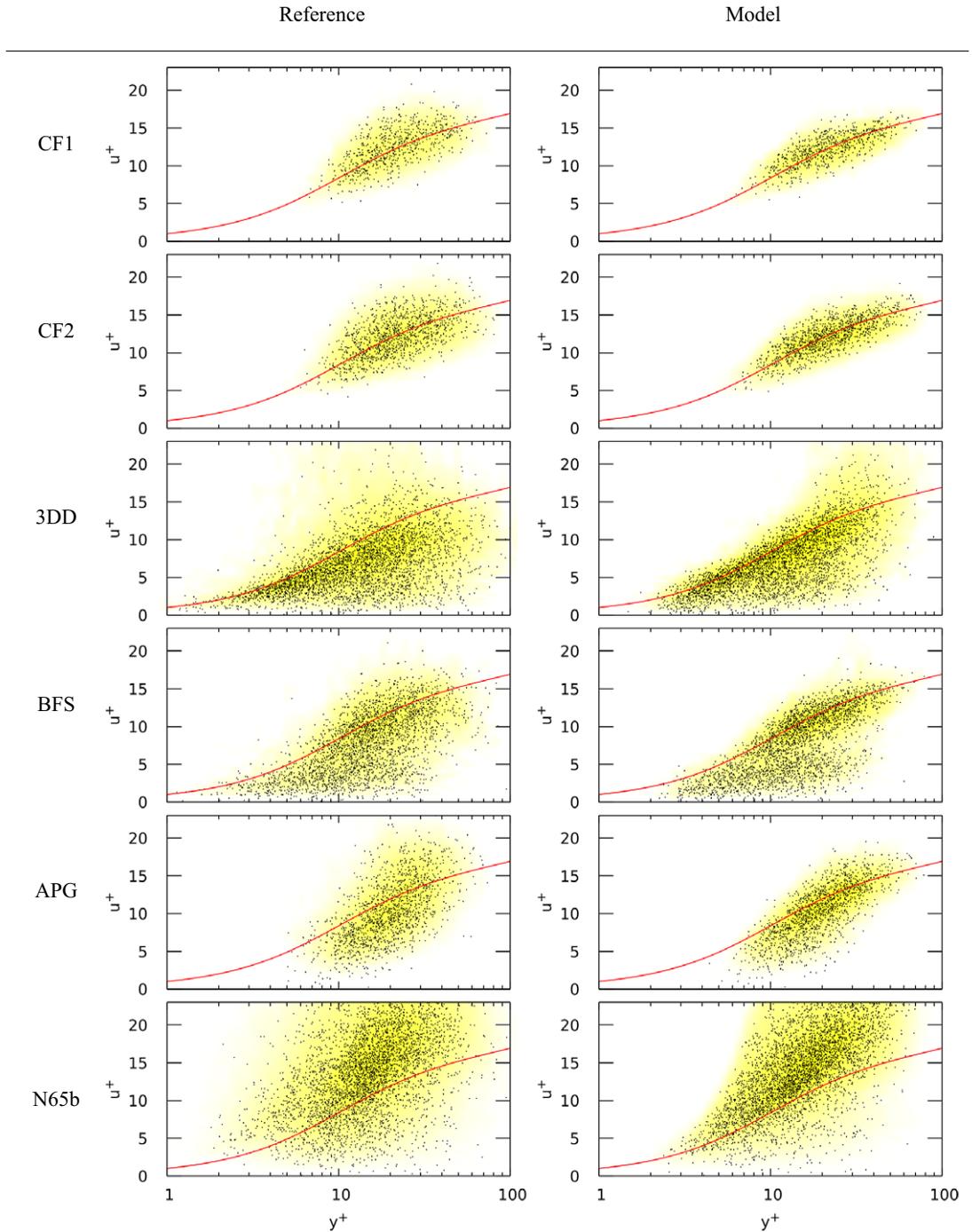


Figure 5. A priori validation: Norm of the scaled tangential velocity u^+ as a function of the scaled distance to the wall y^+ in the CF1, CF2, 3DD, BFS, APG, and N65b datasets using the local reference wall shear stress (left) or the prediction of the graph neural network WSS model (right) to compute the wall unit scaling ($^+$). The red line is Reichardt's law, given by equation (12).

Table 4. *A priori validation: Integral measures of the disagreement between the reference wall shear stress and the prediction of the graph neural network WSS model in the CF1, CF2, 3DD, BFS, APG, and N65b datasets.*

Simulation	Reichardt's law			GNN model		
	C_τ	R_2	$R_{2,\text{Reichardt}}$	C_τ	R_2	$R_{2,\text{Reichardt}}$
CF1	0.48	0.01	0.00	0.60	0.37	0.37
CF2	0.56	0.19	0.00	0.67	0.48	0.36
DD	0.73	0.48	0.00	0.81	0.66	0.34
BFS	0.75	0.53	0.00	0.86	0.76	0.50
APG	0.84	0.60	0.00	0.89	0.77	0.42
N65b	0.57	-0.31	0.00	0.72	0.56	0.67

In addition, the explained variance R_2 is used to compare the variance of the residual of the model to the underlying WSS variance:

$$R_2 = 1 - \frac{\langle (\tau_{\text{ref}} - Y)^2 \rangle_{\mathcal{D}}}{\langle (\tau_{\text{ref}} - \langle \tau_{\text{ref}} \rangle)^2 \rangle_{\mathcal{D}}}. \quad (14)$$

The Reichardt-residual explained variance similarly compares the model residual to the residual that would be obtained using Reichardt's law to compute the WSS,

$$R_{2,\text{Reichardt}} = 1 - \frac{\langle (\tau_{\text{ref}} - Y)^2 \rangle_{\mathcal{D}}}{\langle (\tau_{\text{ref}} - \tau_{\text{Reichardt}})^2 \rangle_{\mathcal{D}}}. \quad (15)$$

The concordance coefficient C_τ is equal to zero if there is no linear correlation between the model and the reference and equal to 1 if and only if the predicted WSS is equal to the reference WSS for each time step. The explained variance R_2 is equal to zero if the variance of the residual is equal to the reference shear stress variance and positive if it is lower. In other words, the coefficient R_2 compares the model to a constant model that always returns the mean value of the data. The Reichardt-residual explained variance $R_{2,\text{Reichardt}}$ instead compares the model to Reichardt's law (12). Namely, $R_{2,\text{Reichardt}}$ is equal to zero if the variance of the residual is equal to the variance of the residual of a model based on Reichardt's law, positive if it is lower, and negative if Reichardt's law is more accurate than the GNN WSS model. This is particularly relevant in flows where the analytical profile of Reichardt (1951) is suitable, such as fully developed turbulent channel flows. As for C_τ , both R_2 and $R_{2,\text{Reichardt}}$ are equal to 1 for an ideal model. Note that the values obtained for different datasets can only be compared with great care, since these three indicators depend not only on the intrinsic performance of the model but also on the underlying distribution of the data. For instance, the value of C_τ associated with Reichardt's law is larger in the 3DD and BFS simulations than in the TCF1 and TCF2 simulations, although the model performs worse in these two simulations. In all simulations, the GNN WSS model agrees well with the reference WSS and improves by all measures the results compared to Reichardt's law baseline. Even in the case of a channel flow (case CF1 or CF2), for which the law of the wall is most well suited, the GNN WSS model reduces the residual variance of a model based on Reichardt's law by around a third. Figure 6 shows that this variance reduction is consistently obtained for a wide range of distances to the wall. In the 3DD, BFS, and N65b simulations on the other hand, variations of model performance with y^+ may be observed. In particular, very low value of y^+ are associated with a very large value of $R_{2,\text{Reichardt}}$ because those corresponds to separated flow regions that are not accurately described by Reichardt's law.

Finally, the instantaneous agreement between the predicted and reference WSS is reflected in an improvement of the time- and space-average predicted WSS, as may be seen Figure 7. Note that the

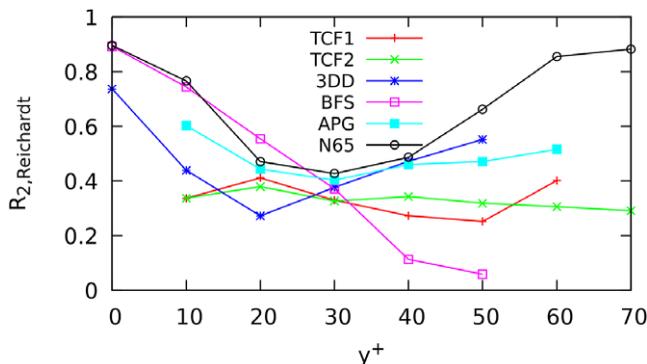
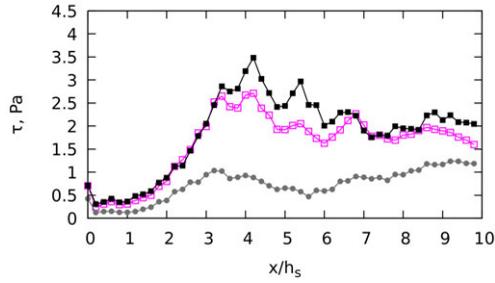


Figure 6. *A priori validation: Reichardt-residual explained variance ($R_{2,\text{Reichardt}}$) in the CF1, CF2, 3DD, BFS, APG, and N65b datasets conditioned on the scaled distance to the wall y^+ .*

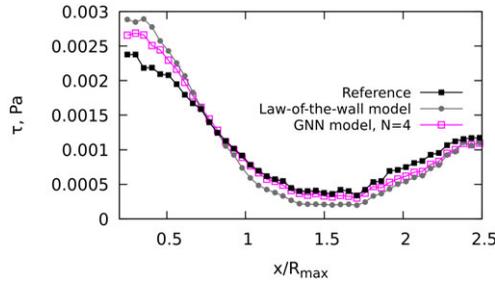
profiles of WSS in Figure 7 have not been scaled in order to explicitly show that the model can operate on flows that have widely different scales, thanks to the scaling used for the model inputs. In accordance with the instantaneous results, a law-of-the-wall model based on Reichardt's law greatly underestimates the mean WSS in the BFS simulation after $x/h_s = 2$ and overestimates the mean WSS in the N65b simulation upstream of the blade and near the leading edge. In the APG simulation, the mean WSS is overestimated in the region of positive curvature ($x/R_{\text{max}} < 1$) and slightly underestimated in the region of negative curvature ($x/R_{\text{max}} > 1$). In each case, the GNN WSS model improves the average WSS prediction and leads to predictions that are close to the reference value. This demonstrates that the model is able to leverage the instantaneous spatial information to improve the mean WSS prediction, even when generalizing to cases that are not seen in the training dataset. A parametric study of the influence of the number of message-passing steps N on this ability has been performed. This parameter is related to the capacity of the GNN since it controls the size of the receptive field of the GNN WSS model (4). The results, provided in Appendix A, demonstrate that a number of message-passing steps $N \geq 3$ is required to properly predict the WSS in the various flows included in our dataset. This is intuitively understood by the fact that the GNN WSS model needs to use information in a relatively large region to immerse the learning problem in the high dimensional space required to reduce the ambiguities associated with the WSS prediction and discriminate between various flow configurations. The value $N = 4$ used in the present study is thus suitable.

4.2. *A posteriori tests*

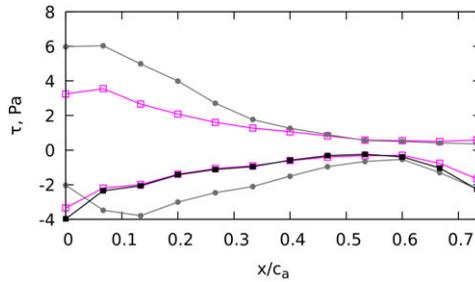
This section investigates the machine-learning wall-shear-stress model a posteriori, that is, by performing simulations implementing the GNN. All simulations presented in this section are performed using a compressible, unstructured and massively parallel flow solver (AVBP; Schönfeld and Rudgyard, 1999), coupled to a code dedicated to the inference of the GNN. The coupling approach is described in Serhani et al. (2022), but the computational cost of the model is also briefly discussed in Appendix B. At each time step, data are extracted from the fields of the simulation in the vicinity of walls to create near-wall graphs, sent to the GNN using the MPI. The inference is distributed to a number of GPU according to a specific data partitioning, independent of the CPU partitioning. The GPU partitions have an overlap corresponding to the number of message-passing steps N of the model, to ensure that each partition contains the receptive field required for the WSS computation (Figure 7), and accordingly that the model prediction is independent of the number of partitions. The resulting predicted WSS is then sent back to main flow solver to impose the boundary condition. The numerical method used in all simulations is similar and relies on the resolution of compressible LES equations with the ideal gas equation of state and an explicit time stepping scheme.



(a) Backward-facing step (BFS)



(b) Adverse pressure gradient (APG)



(c) NACA 65-009 blade cascade (N65b) (†)

Figure 7. *A priori validation: Average prediction of a model based on Reichardt’s law and of the graph neural network WSS model. The average is performed in both time and spanwise direction in the BFS simulation (a), the APG simulation (b), and the N65b simulation (c) on the blade surface. (†) On the blade surface of the N65b simulation, the mean wall shear stress is for clarity reported with positive values for the suction side of the blade and negative values for the pressure side of the blade.*

4.2.1. Governing equations

The governing equations of the LESs are:

$$\partial_t \rho + \partial_j (\rho U_j) = 0, \tag{16}$$

$$\partial_i (\rho U_i) + \partial_j (\rho U_j U_i) = -\partial_i P + \partial_j \Sigma_{ij}, \tag{17}$$

$$\partial_i (\rho E) + \partial_j (\rho U_j H) = -\partial_j Q_j - \partial_j (U_j P) + \partial_j (U_j \Sigma_{ij}), \tag{18}$$

$$P = r \rho T, \tag{19}$$

with ρ the filtered density, t the time, \mathbf{U} the Favre-filtered velocity, \mathbf{x} the Cartesian coordinate, P the filtered pressure, \mathbf{Q} the conductive heat flux, E the total energy per unit mass, $H = E + P/\rho$ the total enthalpy per unit mass, and r the specific ideal gas constant. Density and temperature, which do not experience significant variations in our test cases, are hence resolved nonetheless by the numerical method. The fluid is air. No external body force or heat source are taken into account. Temperature is computed using tabulated data from Stull and Prophet (1971) based on internal energy $e = E - \frac{1}{2}U_i U_i$. The viscous stress tensor and conductive heat flux are computed assuming a Newtonian fluid under Stokes' hypothesis and Fourier's law while subgrid-scale stresses are modeled using an eddy-viscosity model. The stress tensor Σ is, thus, given by

$$\Sigma_{ij} = \left(\mu + \mu_{\text{sgs}} \right) \left(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i} - \frac{2}{3} \frac{\partial U_k}{\partial x_k} \delta_{ij} \right), \quad (20)$$

where δ_{ij} denotes the Kronecker delta.

4.2.2. Channel flow

Several test cases have been selected to verify a posteriori the relevance of the GNN WSS model. First, a simulation of a fully developed turbulent channel flow at friction Reynolds number $Re_\tau = 950$ is performed. The goal is to ensure that the model is able to match in a channel flow the performance of classical algebraic wall stress models. The domain is a rectangular channel of size $12.6h_c \times 2h_c \times 6.3h_c$, where h_c is the half-height of the channel. The mesh is fully tetrahedral and contains 2 million grid points. It is referred to as the original mesh (O) hereafter. The mean edge length is $\Delta^+ = 50$ in wall units, and the mean first off-the-wall grid point height is $y^+ = 37$, which is within the logarithmic layer. A cell-vertex finite-element method is used to discretize equations (16)–(19). The convective scheme is a two-step Taylor–Galerkin scheme with third-order spatial and temporal accuracy (Colin and Rudgyard, 2000), while the diffusive scheme is a centered second-order scheme. The subgrid-scale viscosity is computed using the Smagorinsky's model (Smagorinsky, 1963). The profiles of mean streamwise velocity and standard deviation of streamwise velocity, given in Figure 8, clearly show that a simulation with a GNN WSS model leads to similar results to an algebraic model based on the law of the wall. For both models, there is, however, a non-negligible discrepancy between the LESs and the reference profiles of Hoyas and Jiménez (2008). In particular, there is a mismatch between the logarithmic layers in the velocity profile. This mismatch is well documented in the literature for shear stress models and is due to numerical and subgrid-modeling errors in the first few grid points (Larsson et al., 2016; Bose and Park, 2018). In the context of algebraic wall modeling, Kawai and Larsson (2012) proposed a method to address the issue of logarithmic-layer mismatch by using as input the velocity farther from the wall, at least three cells within the large-eddy-simulation grid, rather than the velocity at the first off-the-wall point.

The approach of Kawai and Larsson (2012) can be adapted to the case of a GNN WSS model, by arguing that since the velocity close to the wall is more affected by numerical and subgrid-scale modeling errors, only the velocity far from the wall should be considered for the prediction of the GNN. This hypothesis is tested by training a dedicated GNN WSS model that only uses the grid points three neighbors away from the wall for its prediction, that is the grid points that are separated from the target walls \mathbb{W} by exactly three edges. The a posteriori results obtained with this model in the case of a channel flow are given in Figure 9. The logarithmic-layer mismatch is greatly reduced using the GNN WSS model that ignores the grid points too close to the wall. This explicitly demonstrates that the approach of Kawai and Larsson (2012), well acknowledged for algebraic wall models, can be readily adapted to the case of a machine-learning wall-shear-stress model. However, this approach has the significant drawback of withholding valuable near-wall data from the model, that although not necessary in a channel flow may be more critical in more complex flows and geometries.

The grid sensitivity of the results is assessed by performing a posteriori LESs of the channel flow for two additional levels of refinement, namely a finer mesh (F) associated with a mean first off-the-wall grid point height $y^+ = 27$ and a coarser mesh (C) associated with a mean first off-the-wall grid point height

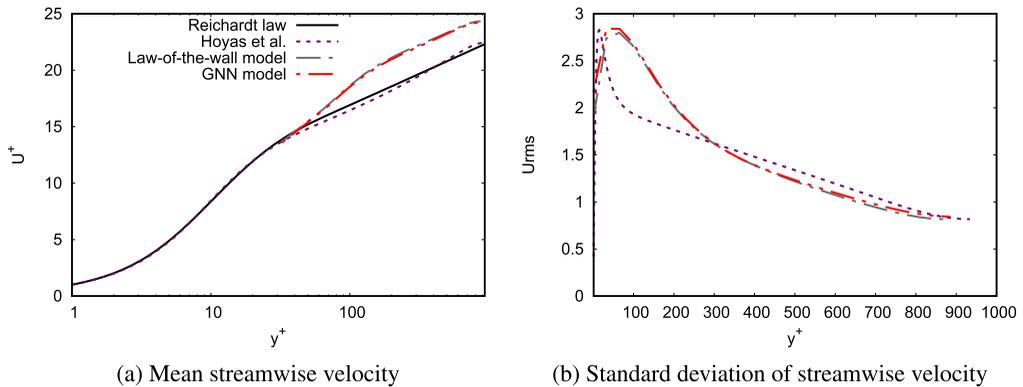


Figure 8. *A posteriori validation: (a) Mean streamwise velocity and (b) standard deviation of streamwise velocity in large-eddy simulations of a channel flow at friction Reynolds number $Re_\tau = 950$ with an algebraic wall stress model and a graph neural network WSS model. The unfiltered direct numerical simulation profile of Hoyas and Jiménez (2008) is given for comparison.*

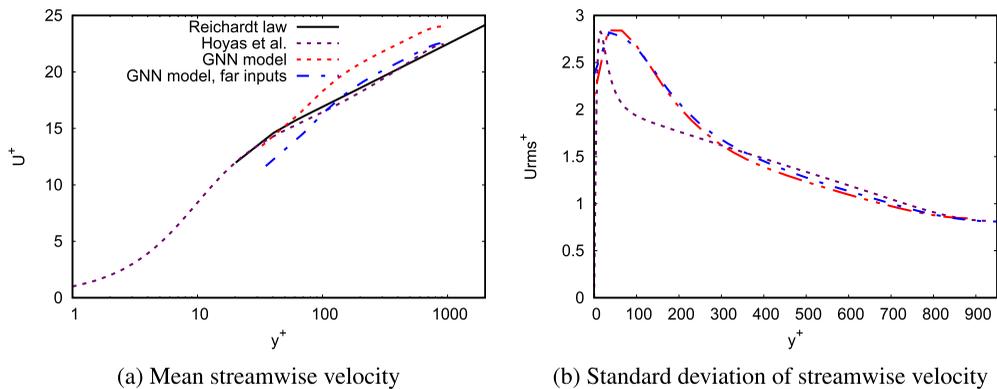


Figure 9. *A posteriori validation: (a) Mean streamwise velocity and (b) standard deviation of streamwise velocity in large-eddy simulations of a channel flow at friction Reynolds number $Re_\tau = 950$ with a graph neural network WSS model without restrictions (GNN model) and a graph neural network WSS model that only uses the grid points at least three neighbors away from the wall for its prediction (GNN model, far inputs). The unfiltered direct numerical simulation profile of Hoyas and Jiménez (2008) is given for comparison.*

$y^+ = 54$. With the F mesh (finer mesh), the GNN WSS model lead to a more accurate velocity profile than the law-of-the-wall model (Figure 10a). This is likely due to the fact that the majority of first off-the-wall grid points are located within the buffer layer, thus outside the domain of validity of the logarithmic law of the wall. With the C mesh (coarser mesh), the GNN WSS model and the law-of-the-wall model lead to almost identical results (Figure 10b). The logarithmic-layer mismatch observed in the velocity profile can, as with the original mesh O, be removed by applying the approach proposed by Kawai and Larsson (2012) to the GNN WSS model with the mesh C. However, this solution does not lead to a significant improvement with the mesh F, which exhibits a less pronounced logarithmic-layer mismatch.

Finally, the simulation of a fully developed turbulent channel flow at friction Reynolds number $Re_\tau = 2000$ is performed to verify that the GNN WSS model can operate for a different Reynolds number, outside of the range of friction Reynolds numbers seen during training ($Re_\tau = 180$ and 950). The geometry and the numerical method are the same as for the LES at $Re_\tau = 950$. The mesh refinement is similar to the mesh refinement of the original mesh (O) used at $Re_\tau = 950$. Namely, the mean first off-the-

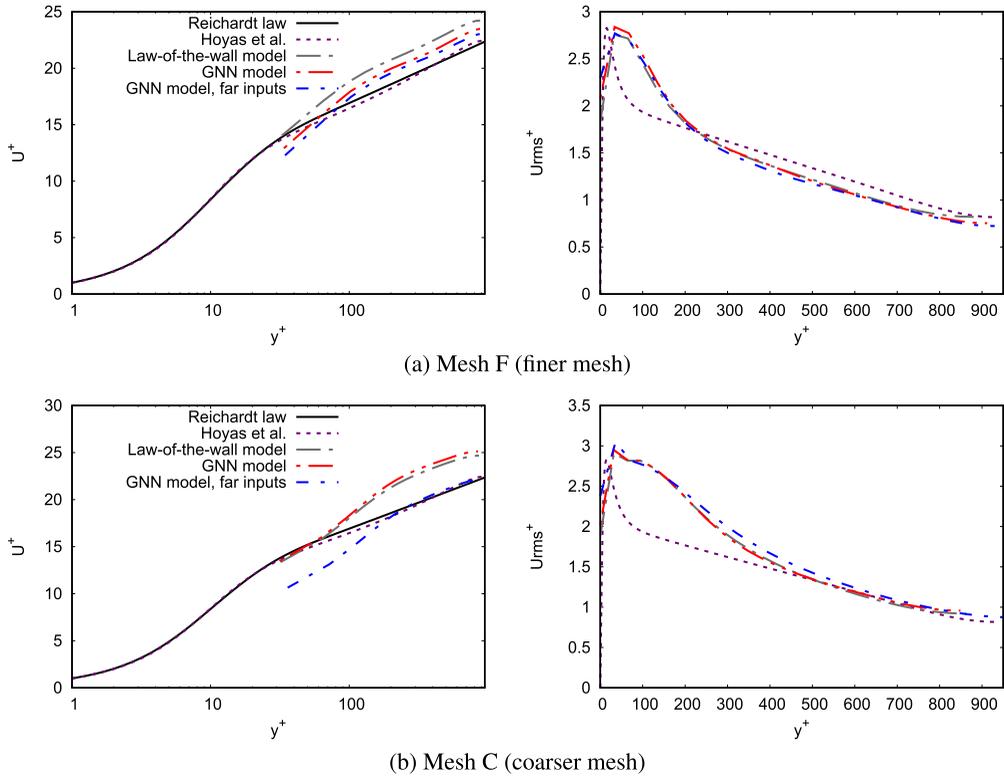


Figure 10. *A posteriori validation: Mean streamwise velocity and standard deviation of streamwise velocity in large-eddy simulations of a channel flow at friction Reynolds number $Re_\tau = 950$ with the (a) meshes F (finer mesh) and (b) C (coarser mesh). The unfiltered direct numerical simulation profile of Hoyas and Jiménez (2008) is given for comparison.*

wall grid point height is $y^+ = 37$. The profiles of mean streamwise velocity and standard deviation of streamwise velocity are given in Figure 11. The results demonstrate that the GNN WSS model performs as well as an algebraic model based on the equilibrium law of the wall at $Re_\tau = 2000$. As in the simulation at $Re_\tau = 950$, the velocity profile features a logarithmic-layer mismatch in both simulations. The logarithmic-layer mismatch can be removed by applying the approach proposed by Kawai and Larsson (2012) to the GNN WSS model (Figure 11), as described above for the case $Re_\tau = 950$. For clarity, we stress that although we did not perform simulations to demonstrate it, the approach of Kawai and Larsson (2012) could be applied to remove the logarithmic-layer mismatch with an algebraic law-of-the-wall model, as was proposed and performed in the literature Kawai and Larsson (2012). Overall, the performance of the present GNN WSS model is thus satisfactory for the a posteriori simulation of a channel flow at both $Re_\tau = 950$ and $Re_\tau = 2000$.

4.2.3. Backward-facing step

The LES of the flow over a BFS using a GNN WSS model is addressed in this section. The configuration is similar to the simulation BFS described in Section 2.1, which was not included in the training dataset. The expansion ratio $ER = (L_y + h_s)/L_y$ of the step, where L_y is the inlet channel height and h_s the step height, is equal to 1.2. The Reynolds number based on h_s and the inflow bulk velocity is $Re_h = 5100$. The inlet is located $10 h_s$ before the step. At the inlet, the mean streamwise velocity profile of the wall-resolved simulation of Dupuy et al. (2022) is imposed with the characteristic nonreflecting boundary condition of Poinot and Lele (1992). In addition, isotropic fluctuations are imposed following the turbulence kinetic

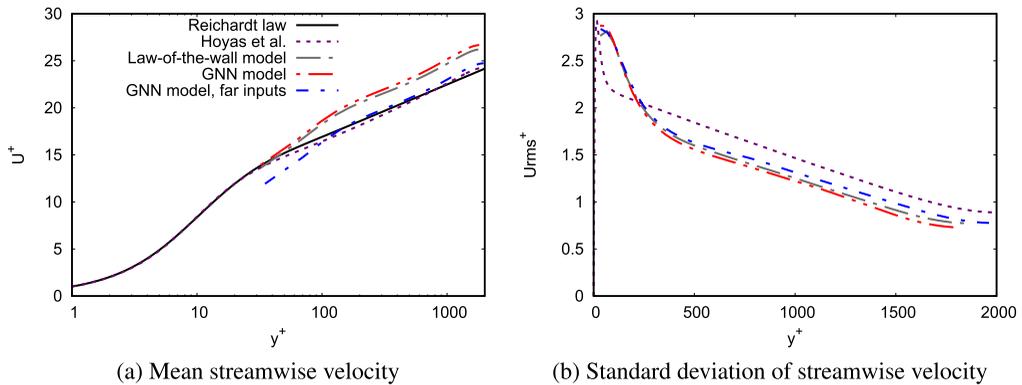


Figure 11. *A posteriori validation: (a) Mean streamwise velocity and (b) standard deviation of streamwise velocity in large-eddy simulations of a channel flow at friction Reynolds number $Re_\tau = 2000$ with an algebraic wall stress model, a graph neural network WSS model without restrictions (GNN model) and a graph neural network WSS model that only uses the grid points at least three neighbors away from the wall for its prediction (GNN model, far inputs). The unfiltered direct numerical simulation profile of Hoyas and Jiménez (2008) is given for comparison.*

energy profile of the wall-resolved simulation using the approach of Kraichnan (1970). At the outlet, located $20 h_s$ after the step, a constant pressure of 1 bar is imposed. The spanwise length of the domain is $12 h_s$. Periodic boundary conditions are used on the side walls. A symmetric boundary condition, that is a zero shear stress and a zero wall-normal velocity, is imposed at the upper boundary of the computational domain. The convective and diffusive schemes are second-order accurate (Lax and Wendroff, 1960). The subgrid-scale viscosity is computed using the Sigma model (Nicoud et al., 2011). The mesh contains 0.9 million grid points. The mean first off-the-wall grid point height is $y^+ = 41$ in wall units, based on the WSS of the boundary layer upstream of the step to compute the wall-unit scaling, and in the range $y^+ = 20\text{--}25$ in the separated region downstream of the step. However, a specific refinement is performed farther from the wall, near the interface between the inlet boundary layer and the separated region. A close-up view of the mesh around the refined region is given in Figure 12.

Wall-modeled LESs performed using an algebraic shear stress model based on the law of the wall and GNN WSS models are compared in Figure 13. Simulations are performed with up to $N = 5$ message-passing steps in order to study a posteriori the influence of the number of steps on the ability of the model to detect separation. The GNN WSS model is only applied on the bottom wall after the step to guarantee that all simulations have the same velocity profile upstream of the step. Overall, the mean velocity downstream of the step is mostly governed by the influence of the shear layer induced by the step. The influence of the GNN WSS model may nevertheless be observed near the onset of the mean backflow region ($x/h_s = 0\text{--}2$). In particular, the presence of a region of positive streamwise velocity at the bottom corner of the step is predicted with $N = 2$ or above, in consistency with the reference high-fidelity simulation, whereas it is absent in the simulation with an algebraic wall stress model. As may be seen in Figure 14, the velocity profiles from $x/h_s = 3$ to further downstream are almost identical regardless of the wall-shear-stress model used.

The profile of mean WSS along the wall is reported in Figure 15. With the algebraic wall stress model, the predicted WSS is accurate near the step ($x/h_s < 2$) and far from the step ($x/h_s > 12$) but underestimated in the backflow and reattachment regions. The GNN WSS models improve the prediction in these non-equilibrium regions, by increasing the predicted WSS. The increase in WSS is negligible with $N = 1$ but substantial already with $N = 2$. The improvements from $N = 3$ to $N = 5$ are rather small. Even with $N = 5$, the peak WSS remain underestimated compared to the reference high-fidelity simulation. It is difficult to disentangle the many possible sources of errors in LESs and determine the contribution of the wall-shear-stress model, the subgrid-scale model and the numerical scheme to the remaining discrepancy

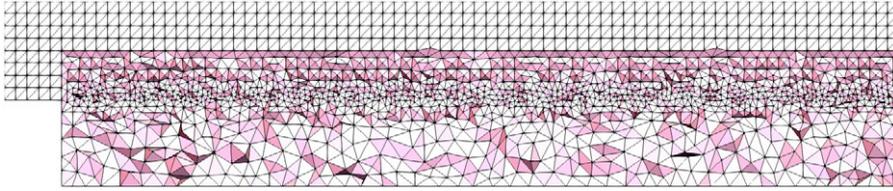


Figure 12. Cross-section of the mesh around the refined region in the x - y plane.

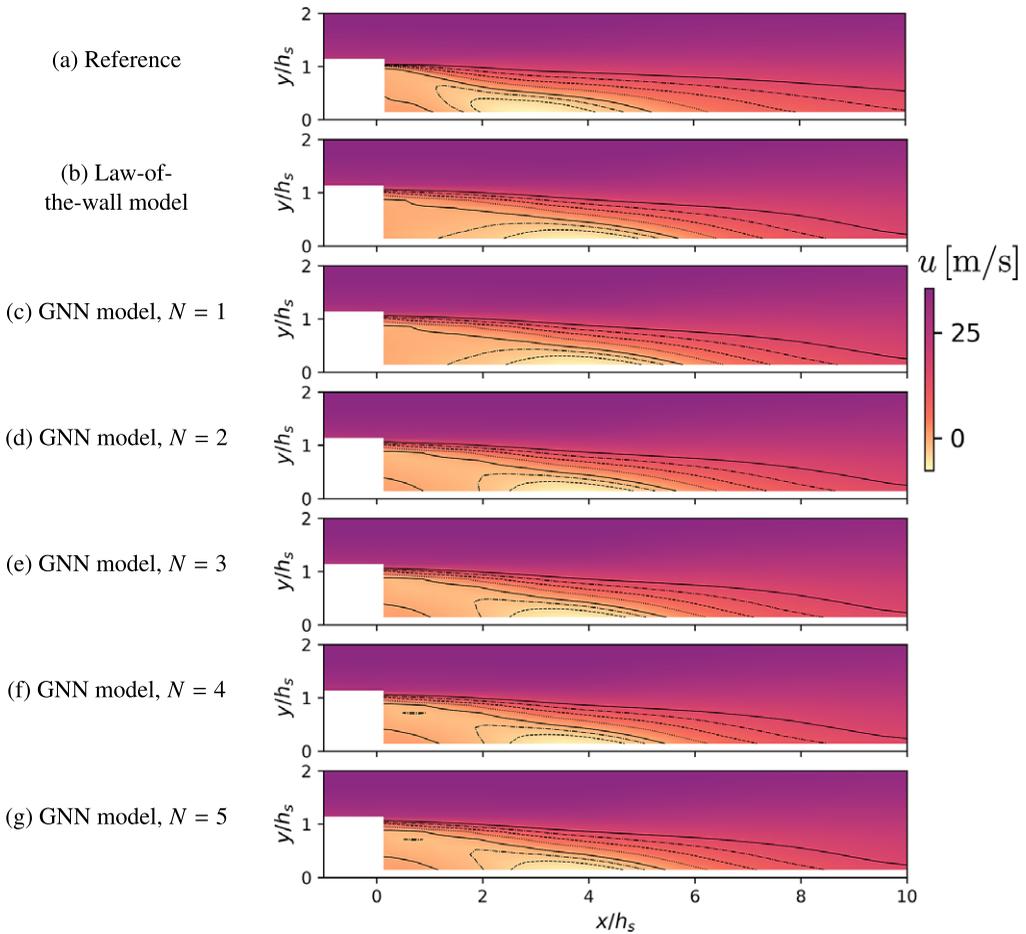


Figure 13. *A posteriori* validation: Mean streamwise velocity in the flow over a backward-facing step according to (a) the reference wall-resolved simulation, (b) a large-eddy simulation with an algebraic wall stress model, and (c–g) graph neural network WSS models. The contour lines denote the level sets -4 , -2 , 0 , 4 , 8 , 12 , and 16 m/s. There are no values within the first large-eddy-simulation cell because the large-eddy simulations do not provide a physical velocity at the wall.

with the reference simulation. Note in addition that since the present shear-stress modeling paradigm only considers the magnitude of the WSS, the shear stress vector is assumed to be aligned with the tangential velocity at the first point above the wall. This assumption is unrealistic near the reattachment point, and leads to a significant underestimation of the reattachment point location in all LESs (Figure 16). The MSE between the shear stress of the LESs and the reference wall-resolved simulation is given in Table 5. The table summarizes the fact that, for this particular simulation, the model associated with $N = 4$ message-

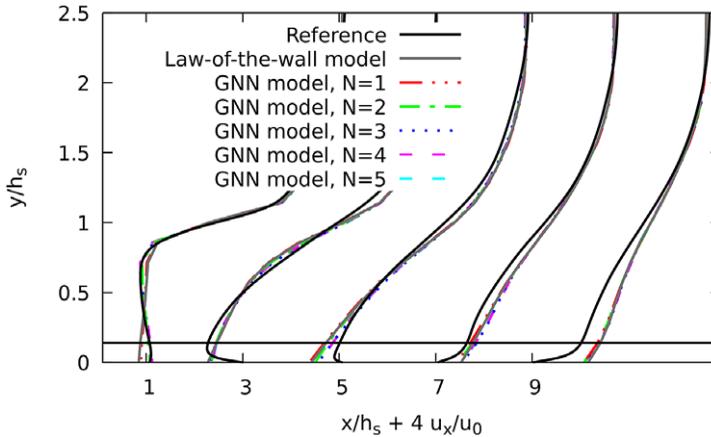


Figure 14. *A posteriori validation: Profile of mean streamwise velocity in the flow over a backward-facing step at the location $x/h_s = 1, 3, 7, 9$, scaled by the free-stream velocity u_0 . The horizontal solid line gives the height of the first point off the wall.*

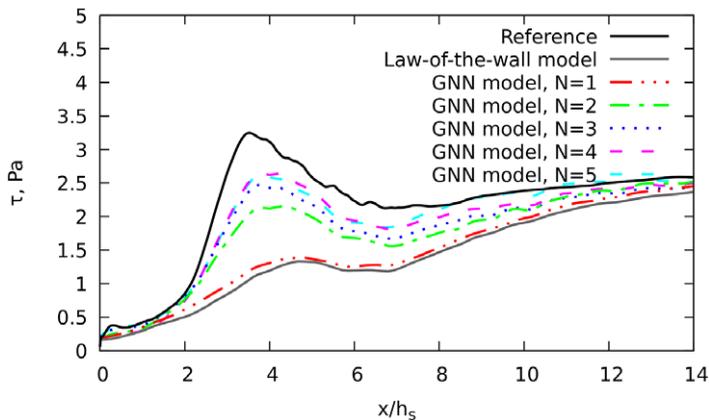


Figure 15. *A posteriori validation: Mean wall shear stress profile in the flow over a backward-facing step.*

passing steps leads to more accurate results than the algebraic model based on the law of the wall. Hence, these a posteriori results are consistent with the a priori tests of Section 4.1 and confirms that a GNN WSS model can lead to large improvements over a model based on the law of the wall for turbulent separated flows.

4.2.4. Linear blade cascade

Finally, the simulation of a linear blade cascade has been performed with an algebraic model based on the law of the wall and a GNN WSS model. The configuration is similar to the simulation N65b described in Section 2.1 and Appendix A, which was not included in the training dataset. The blade geometry is a linearly extruded NACA 65-009 airfoil profile (Abbott et al., 1945), of chord $c = 0.15$ m, attached to flat endwalls with turbulent incoming boundary layers. The geometric parameters of the blade and operating point conditions are reported in Table 6. The Reynolds number of this test case is representative of typical compressor blades. The flow is subsonic, with a low freestream Mach number. The incidence angle is $I = 7^\circ$. The computational domain of the simulation is periodic in the pitchwise direction, implying that each blade is subject to the same flow field. In the spanwise direction, only half a blade has been simulated with a symmetry condition imposed at mid-span. These assumptions have been checked in the

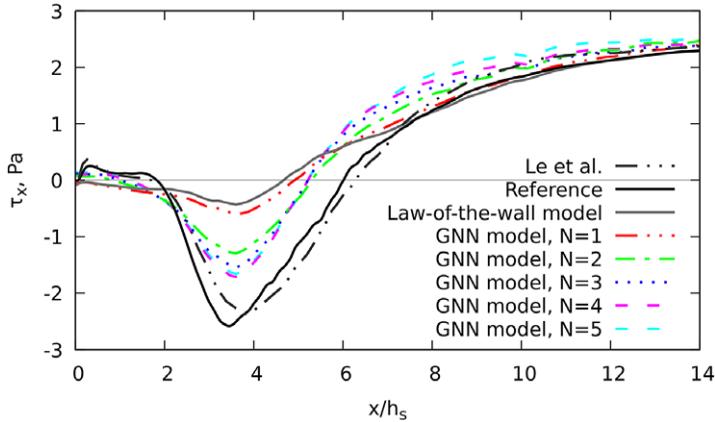


Figure 16. *A posteriori validation: Mean streamwise component of the wall shear stress vector in the flow over a backward-facing step. The numerical results of Le et al. (1997) are given for reference.*

Table 5. *A posteriori validation: Integral measures of the disagreement between the reference wall shear stress in the flow over a backward-facing step and the wall shear stress predicted by large-eddy simulations with an algebraic wall stress model and graph neural network WSS models.*

Simulation	$\int_0^{14h_s} (\tau - \tau_{\text{ref}})^2 dx, \text{N}^2/\text{m}^2$	$\int_0^{14h_s} (\tau_x - \tau_{x\text{ref}})^2 dx, \text{N}^2/\text{m}^2$
LES with law-of-the-wall model	0.806	0.686
LES with GNN model, $N = 1$	0.687	0.585
LES with GNN model, $N = 2$	0.217	0.311
LES with GNN model, $N = 3$	0.120	0.302
LES with GNN model, $N = 4$	0.067	0.308
LES with GNN model, $N = 5$	0.066	0.367

experimental campaigns and were shown to be sufficiently valid (Ma et al., 2011). We denote x the axial direction, y the pitchwise direction, and z the wall-normal direction. The origin of the coordinate system (x, y, z) is placed at the intersection of the leading edge of the blade and the endwall. A curvilinear coordinate s_1 is also defined on the blade surface, where s_1 is zero at the leading edge of the blade, positive on the suction side of the blade and negative on the pressure side of the blade. The inlet plane is located at $x_{\text{inlet}}/c = -3.65$ and the outlet plane at $x_{\text{outlet}}/c = 2.98$. The pitchwise length of the domain is $s = 0.89c$. At the inlet, the experimental velocity profile of Zambonini et al. (2017) is imposed using the characteristic nonreflecting boundary condition of Poinsot and Lele (1992). The nonreflecting boundary condition of Granet et al. (2010) is used to impose a constant pressure of 1 standard atmosphere at the outlet. At the top boundary of the domain, located $h = 1.23c$ above the wall, there is a symmetric boundary condition which imposes the normal velocity and the normal gradients of velocity, temperature, and pressure to zero. A rectangular step of width $0.017c$ and height $0.015c$, located $0.085c$ after the inlet, triggers the transition of the inlet endwall boundary layer. A rectangular step is also placed on the suction side and pressure side of the blade surface, to reproduce the effect of the sandpapers used in the experiments and trigger turbulence transition. The blade steps are sized similarly to the sandpapers present in the experiments. However, the inlet step does not correspond to a feature present in the wind tunnel; the experimental incoming endwall turbulent boundary layer is originating from further upstream and hence cannot be reproduced numerically without simulating the entire wind tunnel.

Two LESs are performed: a simulation using an algebraic shear stress model based on the law of the wall and a simulation using the GNN WSS model. In both cases, a second-order accurate scheme is used

Table 6. Geometric parameters and operating point conditions of the linear cascade (left) and schematic representation of the notations used (right).

Chord, c	0.15 m
Axial chord, c_a	0.110 m
Camber angle, φ	23.22°
Stagger angle, γ	42.70°
Pitch spacing, s	0.134 m
Blade span, h	0.370 m
Free-stream velocity, U_∞	40 m/s
Chord Reynolds number, Re_c	3.82×10^5
Outlet pressure, P_{outlet}	1.01325×10^5 Pa

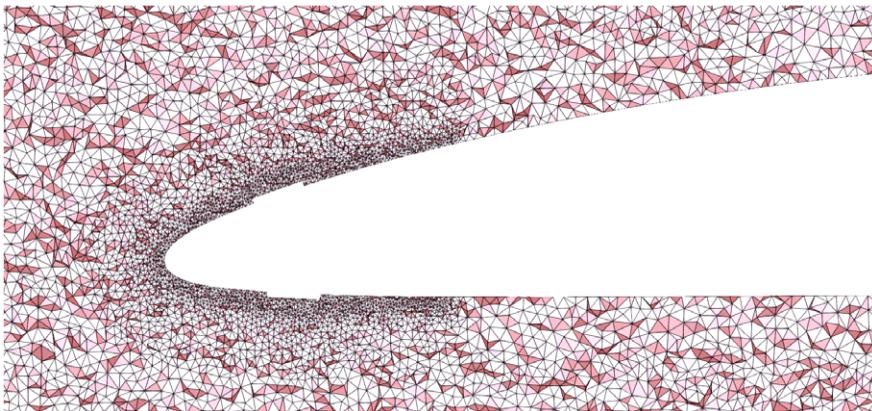
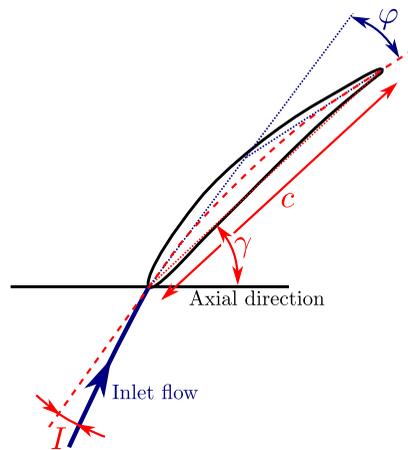


Figure 17. Cross-section of the mesh around the refined region in the x - z plane.

for convection (Lax and Wendroff, 1960) and diffusion. The subgrid-scale viscosity is computed using the Sigma model (Nicoud et al., 2011). The mesh of the LESs is fully tetrahedral and contains 15 million grid points in total. The mesh is more refined around the leading edge in order to resolve the effect of the tripping steps (Figure 17). Outside this region, the mesh refinement near the blade is uniform. In wall units, the edge length varies in the range $\Delta^+ = 40$ – $\Delta^+ = 90$ on the blade surface, and accordingly the first off-the-wall grid point height on the blade surface varies from $y^+ = 30$ to $y^+ = 67$ due to the spatial variation of the WSS.

A comparison of the mean static pressure coefficient on the blade surface obtained in the reference wall-resolved simulation and in the LESs is given in Figure 18. The static pressure coefficient C_p is defined as

$$C_p = \frac{p - p_\infty}{p_{s,\infty} - p_\infty}, \quad (21)$$

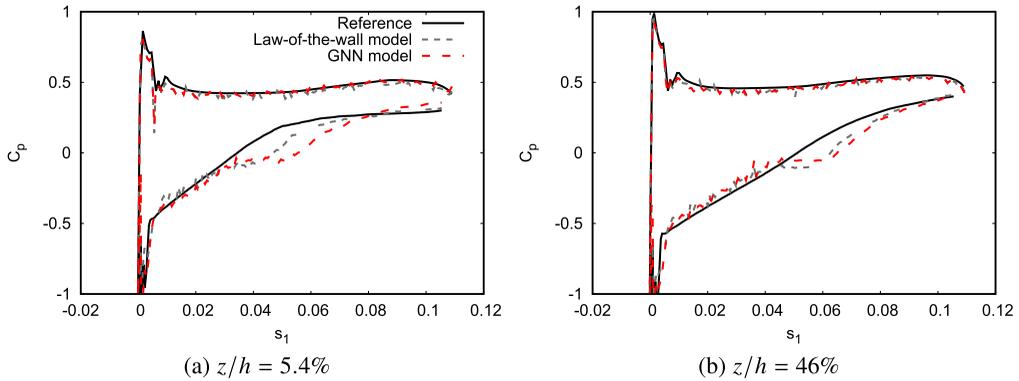


Figure 18. *A posteriori validation: Mean static pressure coefficient on the blade close to the endwall (a) and at mid-height (b). The upper branch corresponds to the pressure side of the blade and the bottom branch corresponds to the suction side of the blade.*

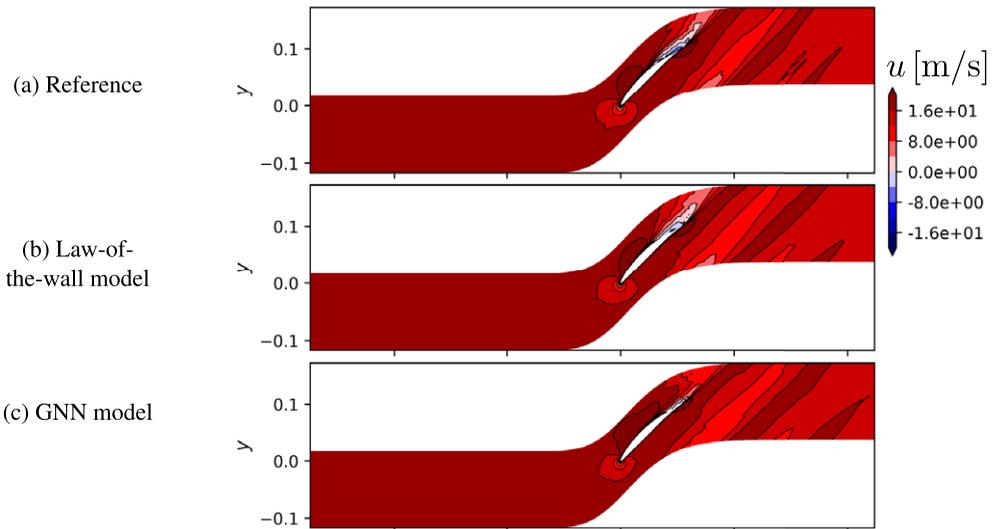


Figure 19. *A posteriori validation: Mean axial velocity on the plane $z/h = 3\%$ in the flow around the NACA 65-009 blade according to (a) the reference wall-resolved simulation, (b) a large-eddy simulation with an algebraic wall stress model, and (c) the graph neural network WSS model.*

where p is the local static pressure and $p_{s,\infty}$ and p_∞ the reference inlet stagnation and static pressures, respectively. The profile of the pressure coefficient on the pressure side of the blade is well reproduced by the LESs both at $z/h = 5.4\%$, near the endwall, and at $z/h = 46\%$, near mid-height. At $z/h = 5.4\%$, the plateau of static pressure coefficient in the aft part of the suction side, associated with the presence of a corner separation region, is shorter in the two LESs, in particular with the GNN WSS model. The contours of mean axial velocity (Figure 19) confirms that the corner separation region on the suction side occurs closer to the trailing edge in the two LESs than in the reference wall-resolved simulation. The coefficient of stagnation pressure loss $\omega_s = (p_{s,\infty} - p_s) / (p_{s,\infty} - p_\infty)$ downstream of the trailing edge of the two LESs is compared to that of the reference wall-resolved simulation in Figure 20. In the two LESs, the wake has a wider footprint and the influence of the corner separation region does not extend as far away from the endwall.

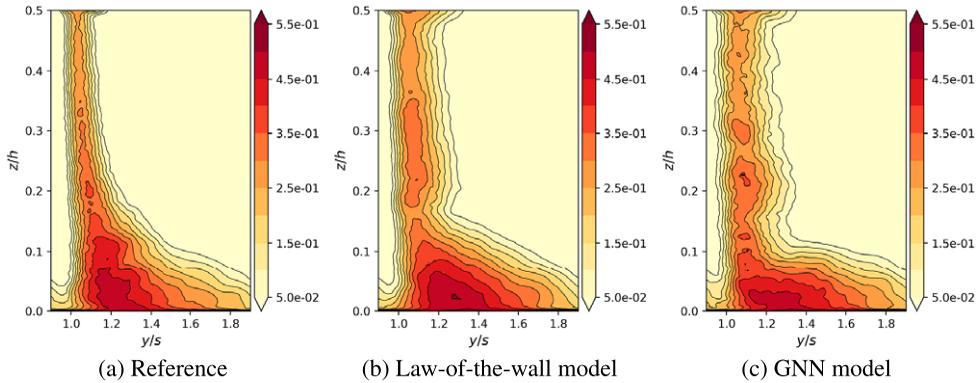


Figure 20. A posteriori validation: Stagnation pressure loss coefficient ω_s at 36.3% axial chord downstream from trailing edge in (a) the reference wall-resolved simulation, (b) a large-eddy simulation with an algebraic wall stress model, and (c) a large-eddy simulation with the graph neural network WSS model.

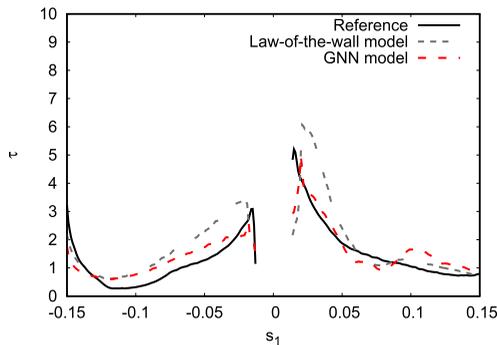


Figure 21. A posteriori validation: Mean wall shear stress profile on the blade surface, averaged in the spanwise direction.

The profile of mean WSS along the blade is reported in Figure 21. The mean WSS is overestimated by the LES with the algebraic wall stress model on the pressure side and on the suction side in the accelerating region near the leading edge. The GNN WSS model leads to a lower WSS amplitude in these regions, which leads to a clear improvement compared to the baseline algebraic model. However, the WSS is overestimated in the region of corner separation by the LES with the GNN WSS model, while the simulation with a law-of-the-wall algebraic model leads to a more accurate prediction in that region. This clearly demonstrates that a priori improvements do not always materialize into a posteriori improvements in all locations. This can be attributed to a discrepancy between the fields resulting from the filtering of direct numerical simulation data and the fields of LESs. It is difficult to assess whether this discrepancy is the responsibility of the wall-shear-stress model or other types of numerical or modeling errors. Overall, we thus consider that the generalizability of the model in the case of compressor blade cascades is still uncertain.

5. Conclusion

A machine-learning wall-shear-stress model has been developed based on reference simulations of the flow in a smooth channel, in a 3DD and in a linear blade cascade. The model directly operates on the unstructured mesh of the simulation and is thus directly applicable in LESs of complex industrial flows. The relevance of the model has been verified a priori using simulations of the flow over a BFS not used

during training and the simulation of a linear cascade at a larger incidence angle than used during training. The model showed a good agreement with the reference WSS and an ability to discriminate between the fully developed turbulent boundary layers, separated flow regions, and transitional boundary layers. The ability of the model to detect separated regions and improve the predictions of a model based on the law of the wall has been demonstrated in an a posteriori setting for the BFS simulation. Furthermore, the model was also able to match the performance of a model based on the law of the wall in a channel flow, for which it is well suited, although it is subject to the same classical issue of logarithmic-layer mismatch. The GNN WSS model also improves the WSS prediction on the pressure side of a blade in a linear cascade, and on the suction side of the blade in the region directly after a tripping step. However, the GNN WSS model is not strictly superior to an algebraic equilibrium wall model in that case, since the WSS is overestimated in the region of corner separation on the suction side of the blade. Overall, this paper demonstrates that graph-based models are a promising tool to address the problem of wall-shear-stress modeling. Improvements to the machine-learning methodology would require a better understanding of the transition from a priori tests to a posteriori tests, and of the relationship between the wall-shear-stress model, the subgrid-scale model and the numerical scheme. This study shows the value of data-driven efforts toward wall-shear-stress modeling in realistic solvers. This justifies further efforts to assemble even more comprehensive and varied databases, which are critical to produce more accurate and robust models.

Acknowledgments. This work was granted access to the computer resources of IDRIS under the allocations A0102A06074 and A0122A06074 made by GENCI. We acknowledge PRACE for awarding us access to Joliot-Curie at GENCI@CEA, France, through the project CornerLES. We are indebted to the European Union's Horizon 2020 research and innovation program (Grant Agreement No. 814837) for producing and making openly available some of the data used in this work. We are grateful to L. Agostini and P. Vincent for providing us the data for the channel-flow simulation at $Re_\tau = 180$, to O. Lehmkuhl and A. Miró for providing us the data for the 3DD simulation, to F. C. Massa and A. Colombo for providing us the data for the APG simulation, to J. Bodet and F. Gao for providing us the data for the stagnation pressure loss coefficient of the NACA 65-009 blade cascade in the experiments at 4° and 7° and in the numerical simulation of Gao et al. (2015).

Author Contributions. Conceptualization: all authors; Data curation: D.D.; Data visualization: D.D.; Investigation: D.D., D.P.; Methodology: D.D., N.O., C.L.; Supervision: N.O., C.L.; Writing original draft: D.D.; Writing review and editing: all authors. All authors approved the final submitted draft.

Competing Interests. The authors declare no competing interests exist.

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

Ethics Statement. The research meets all ethical guidelines, including adherence to the legal requirements of the study country.

Funding Statement. This project has received funding from the European Union's Horizon 2020 research and innovation program (Grant Agreement No. 951,733).

References

- Abbott IH, Von Doenhoff AE and Stivers Jr L (1945) Summary of airfoil data. Technical report, NACA Report no. 824.
- Agostini L and Vincent P (2020) Channel flow data, GitHub repository, https://github.com/LionelAgo/channel_postprocessing.
- Alaya E, Knopp T and Grabe C (2020) TC01: Adverse pressure gradient experiment. Deliverable D6.1-03 of the European Union's Horizon 2020 research and innovation programme no. 814837.
- Ba JL, Kiros JR and Hinton GE (2016). Layer normalization. In I. Guyon, U.V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, & R. Garnett (Eds.), *Advances in Neural Information Processing Systems* (Vol. 30), Curran Associates, Inc., New York, U.S.A., 2017
- Bae HJ and Koumoutsakos P (2022) Scientific multi-agent reinforcement learning for wall-models of turbulent flows. *Nature Communications* 13(1), 1–9.
- Bae HJ, Lozano-Durán A, Bose ST and Moin P (2019) Dynamic slip wall model for large-eddy simulation. *Journal of Fluid Mechanics* 859, 400.
- Balaras E, Benocci C and Piomelli U (1996) Two-layer approximate boundary conditions for large-eddy simulations. *AIAA Journal* 34(6), 1111–1119.

- Bassi F, Botti L, Colombo A, Crivellini A, Ghidoni A and Massa F** (2016) On the development of an implicit high-order discontinuous Galerkin method for DNS and implicit LES of turbulent flows. *European Journal of Mechanics-B/Fluids* 55, 367–379.
- Bassi F, Botti L, Colombo A, Ghidoni A and Massa F** (2015) Linearly implicit Rosenbrock-type Runge–Kutta schemes applied to the discontinuous Galerkin solution of compressible and incompressible unsteady flows. *Computers & Fluids* 118, 305–320.
- Battaglia PW, Hamrick JB, Bapst V, Sanchez-Gonzalez A, Zambaldi V, Malinowski M, Tacchetti A, Raposo D, Santoro A, Faulkner R, Gulcehre C, Song F, Ballard A, Gilmer J, Dahl G, Vaswani A, Allen K, Nash C, Langston V, Dyer C, Heess N, Wierstra D, Kohli P, Botvinick M, Vinyals O, Li Y and Pascanu R** (2018). Relational inductive biases, deep learning, and graph networks. Preprint, [arXiv:1806.01261](https://arxiv.org/abs/1806.01261).
- Battaglia PW, Pascanu R, Lai M, Rezeze D and Kavukcuoglu K** (2016) Interaction networks for learning about objects, relations and physics. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, & R. Garnett, *Advances in Neural Information Processing Systems*, Vol. 29. Curran Associates, Inc., New York, U.S.A., 2017.
- Baurle RA, Tam C-J, Edwards JR and Hassan HA** (2003) Hybrid simulation approach for cavity flows: Blending, algorithm, and boundary treatment issues. *AIAA Journal* 41(8), 1463–1480.
- Bhaskaran R, Kannan R, Barr B and Priebe S** (2021) Science-guided machine learning for wall-modeled large eddy simulation. In *2021 IEEE International Conference on Big Data*. IEEE, New York, U.S.A. pp. 1809–1816.
- Bose ST and Moin P** (2014) A dynamic slip boundary condition for wall-modeled large-eddy simulation. *Physics of Fluids* 26(1), 015104.
- Bose ST and Park GI** (2018) Wall-modeled large-eddy simulation for complex turbulent flows. *Annual Review of Fluid Mechanics* 50, 535–561.
- Boxho M, Rasquin M, Toulorge T, Dergham G, Winckelmans G and Hillewaert K** (2022) Analysis of space-time correlations to support the development of wall-modeled LES. *Flow, Turbulence and Combustion* 109, 1081–1109.
- Bronstein MM, Bruna J, LeCun Y, Szlam A and Vandergheynst P** (2017) Geometric deep learning: Going beyond Euclidean data. *IEEE Signal Processing Magazine* 34(4), 18–42.
- Bruna J, Zaremba W, Szlam A and LeCun Y** (2013) Spectral networks and locally connected networks on graphs. *2nd International Conference on Learning Representations, ICLR 2014, Banff, Canada*.
- Brunton SL, Noack BR and Koumoutsakos P** (2020) Machine learning for fluid mechanics. *Annual Review of Fluid Mechanics* 52, 477–508.
- Cabot W** (1995) Large-eddy simulations with wall models. In *Stanford Center for Turbulence Research Annual Research Briefs*, Vol. 20. Center for Turbulence Research, Stanford University, Palo Alto, CA, U.S.A.
- Catchirayer M, Bousuge J-F, Sagaut P, Montagnac M, Papadogiannis D and Garnaud X** (2018) Extended integral wall-model for large-eddy simulations of compressible wall-bounded turbulent flows. *Physics of Fluids* 30(6), 065106.
- Chang MB, Ullman T, Torralba A and Tenenbaum JB** (2016) A compositional object-based approach to learning physical dynamics. *5th International Conference on Learning Representations, ICLR 2017, Toulon, France*.
- Chen J, Hachem E and Viquerat J** (2021) Graph neural networks for laminar flow prediction around random two-dimensional shapes. *Physics of Fluids* 33(12), 123607.
- Cherry EM, Elkins CJ and Eaton JK** (2008) Geometric sensitivity of three-dimensional separated flows. *International Journal of Heat and Fluid Flow* 29(3), 803–811.
- Chung D and Pullin DI** (2009) Large-eddy simulation and wall modelling of turbulent channel flow. *Journal of Fluid Mechanics* 631, 281–309.
- Colin O and Rudgyard M** (2000) Development of high-order Taylor–Galerkin schemes for LES. *Journal of Computational Physics* 162(2), 338–371.
- Dapogny C, Dobrzynski C and Frey P** (2014) Three-dimensional adaptive domain remeshing, implicit domain meshing, and applications to free and moving boundary problems. *Journal of Computational Physics* 262, 358–378.
- Davidson L and Peng S-H** (2003) Hybrid LES-RANS modelling: A one-equation SGS model combined with a $k-\omega$ model for predicting recirculating flows. *International Journal for Numerical Methods in Fluids* 43(9), 1003–1018.
- Deardorff JW** (1970) A numerical study of three-dimensional turbulent channel flow at large Reynolds numbers. *Journal of Fluid Mechanics* 41(2), 453–480.
- Del Álamo JC and Jiménez J** (2003) Spectra of the very large anisotropic scales in turbulent channels. *Physics of Fluids* 15(6), L41–L44.
- Disotell KJ and Rumsey CL** (2017) Development of an axisymmetric afterbody test case for turbulent flow separation validation. NASA/TM-2017-219680. Technical report.
- Dobrzynski C and Frey P** (2008) Anisotropic delaunay mesh adaptation for unsteady simulations. In Rao V. Garimella (Eds.), *Proceedings of the 17th International Meshing Roundtable*. Springer, Berlin, Heidelberg, Germany. pp. 177–194.
- Dupuy D, Odier N and Lapeyre C** (2022) Data-driven wall modelling for turbulent separated flows. (Submitted for publication).
- Duraisamy K, Iaccarino G and Xiao H** (2019) Turbulence modeling in the age of data. *Annual Review of Fluid Mechanics* 51, 357–377.
- Ercofac** (2022) Direct numerical simulation datasets, Ercofac wiki. Available at https://www.kbwiki.ercofac.org/w/index.php/DNS_Index Accessed 21 February 2023.
- Gao H, Zahr MJ and Wang J-X** (2022) Physics-informed graph neural Galerkin networks: A unified framework for solving PDE-governed forward and inverse problems. *Computer Methods in Applied Mechanics and Engineering* 390, 114502.

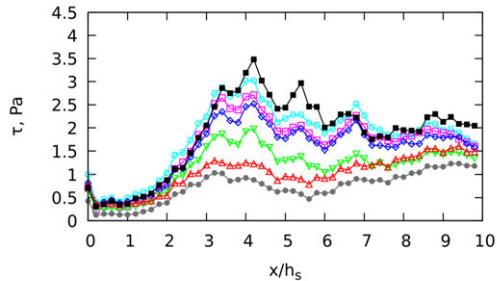
- Gao F, Zambonini G, Boudet J, Ottavy X, Lu L and Shao L** (2015) Unsteady behavior of corner separation in a compressor cascade: Large eddy simulation and experimental study. *Proceedings of the Institution of Mechanical Engineers, Part A: Journal of Power and Energy* 229(5), 508–519.
- Gao W, Zhang W, Cheng W and Samtaney R** (2019) Wall-modelled large-eddy simulation of turbulent flow past airfoils. *Journal of Fluid Mechanics* 873, 174–210.
- Gilmer J, Schoenholz SS, Riley PF, Vinyals O and Dahl GE** (2017) Neural message passing for quantum chemistry. In *International Conference on Machine Learning*. PMLR, pp. 1263–1272.
- Gori M, Monfardini G and Scarselli F** (2005) A new model for learning in graph domains. In *Proceedings. 2005 IEEE International Joint Conference on Neural Networks, 2005*, Vol. 2. IEEE, New York, U.S.A. pp. 729–734.
- Granet V, Vermorel O, Léonard T, Gicquel L and Poinso T** (2010) Comparison of nonreflecting outlet boundary conditions for compressible solvers on unstructured grids. *ALAA Journal* 48(10), 2348–2364.
- Hoyas S and Jiménez J** (2008) Reynolds number effects on the Reynolds-stress budgets in turbulent channels. *Physics of Fluids* 20(10), 101511.
- Huang XLD, Yang X and Kunz R** (2019) Wall-modeled large-eddy simulations of spanwise rotating turbulent channels—Comparing a physics-based approach and a data-based approach. *Physics of Fluids* 31(12), 125105.
- Huynh HT** (2007) A flux reconstruction approach to high-order schemes including discontinuous Galerkin methods. In *18th AIAA Computational Fluid Dynamics Conference*, American Institute of Aeronautics & Astronautics (AIAA), Reston, VA, U.S.A. p. 4079.
- Inoue M and Pullin DI** (2011) Large-eddy simulation of the zero-pressure-gradient turbulent boundary layer up to $Re_\theta = O(1012)$. *Journal of Fluid Mechanics* 686, 507–533.
- Jumper J, Evans R, Pritzel A, Green T, Figurnov M, Ronneberger O, Tunyasuvunakool K, Bates R, Žídek A, Potapenko A, Bridgland A, Meyer C, Kohli SAA, Ballard AJ, Cowie A, Romera-Paredes B, Nikolov S, Jain R, Adler J, Back T, Petersen S, Reiman D, Clancy E, Zielinski M, Steinegger M, Pacholska M, Berghammer T, Bodenstein S, Silver D, Vinyals O, Senior AW, Kavukcuoglu K, Kohli P and Hassabis D** (2021) Highly accurate protein structure prediction with alphafold. *Nature* 596(7873), 583–589.
- Karypis G and Kumar V** (1998) A fast and high quality multilevel scheme for partitioning irregular graphs. *SIAM Journal on Scientific Computing* 20(1), 359–392.
- Kawai S and Larsson J** (2012) Wall-modeling in large eddy simulation: Length scales, grid resolution, and accuracy. *Physics of Fluids* 24(1), 015105.
- Kingma DP and Ba J** (2015) Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, USA*.
- Kipf T, Fetaya E, Wang K-C, Welling M and Zemel R** (2018) Neural relational inference for interacting systems. In Jennifer Dy, Andreas Krause, *International Conference on Machine Learning*. PMLR, pp. 2688–2697.
- Kraichnan RH** (1970) Diffusion by a random velocity field. *Physics of Fluids* 13(1), 22–31.
- Larsson J, Kawai S, Bodart J and Bermejo-Moreno I** (2016) Large eddy simulation with modeled wall-stress: Recent progress and future directions. *Mechanical Engineering Reviews* 3(1), 15-00418.
- Lax P and Wendroff B** (1960) Systems of conservation laws. *Communications on Pure and Applied Mathematics* 13(2), 217–237.
- Le H, Moin P and Kim J** (1997) Direct numerical simulation of turbulent flow over a backward-facing step. *Journal of Fluid Mechanics* 330, 349–374.
- LeCun Y, Bengio Y and Hinton G** (2015) Deep learning. *Nature* 521(7553), 436–444.
- Lehmkuhl O, Houzeaux G, Owen H, Chrysokentis G and Rodriguez I** (2019) A low-dissipation finite element scheme for scale resolving simulations of turbulent flows. *Journal of Computational Physics* 390, 51–65.
- Leonard A** (1974) Energy cascade in large eddy simulations of turbulent fluid flows. *Advances in Geophysics* 18A, 237–248.
- Lin LI-K** (1989) A concordance correlation coefficient to evaluate reproducibility. *Biometrics* 45, 255–268.
- Lozano-Durán A and Bae HJ** (2020) Self-critical machine-learning wall-modeled LES for external aerodynamics. In *Center for Turbulence Research Annual Research Briefs*, Center for Turbulence Research, Stanford University, Palo Alto, CA, U.S.A. pp. 197–210.
- Lozano-Durán A and Jiménez J** (2014) Effect of the computational domain on direct simulations of turbulent channels up to $Re_\tau = 4200$. *Physics of Fluids* 26(1), 011702.
- Lozano-Durán A and Jiménez J** (2015) Turbulent channel flow at $Re_\tau = 950$, DNS database of wall-bounded turbulent flows. Available at http://hal.dmt.upm.es/raw_database/Channels/Re950/2pipi/ Accessed 1 March 2021.
- Ma W, Ottavy X, Lu L, Francis L and Gao F** (2011) Experimental study of corner stall in a linear compressor cascade. *Chinese Journal of Aeronautics* 24(3), 235–242.
- Nicoud F, Baggett JS, Moin P and Cabot W** (2001) Large eddy simulation wall-modeling based on suboptimal control theory and linear stochastic estimation. *Physics of Fluids* 13(10), 2968–2984.
- Nicoud F, Baya Toda H, Cabrit O, Bose S and Lee J** (2011) Using singular values to build a subgrid-scale model for large eddy simulations. *Physics of Fluids* 23(8), 085106.
- Paolucci S** (1982) On the filtering of sound from the Navier–Stokes equations. Technical Report SAND82-8257, Sandia National Laboratories, Livermore, CA.
- Pfaff T, Fortunato M, Sanchez-Gonzalez A and Battaglia PW** (2020) Learning mesh-based simulation with graph networks. *9th International Conference on Learning Representations, ICLR 2021, Vienna, Austria*.

- Piomelli U** (2008) Wall-layer models for large-eddy simulations. *Progress in Aerospace Sciences* 44(6), 437–446.
- Piomelli U, Ferziger J, Moin P and Kim J** (1989) New approximate boundary conditions for large eddy simulations of wall-bounded flows. *Physics of Fluids A: Fluid Dynamics* 1(6), 1061–1068.
- Poinsot TJ and Lele SK** (1992) Boundary conditions for direct simulations of compressible viscous flows. *Journal of Computational Physics* 101(1), 104–129.
- Pouech P, Duchaine F and Poinsot T** (2019) Ignition of a premixed methane-air flow over a turbulent backward-facing step by direct numerical simulation. In *17th International Conference on Numerical Combustion*, Aachen, Germany.
- Pouech P, Duchaine F and Poinsot T** (2021) Premixed flame ignition in high-speed flows over a backward facing step. *Combustion and Flame* 229, 111398.
- Radhakrishnan S, Adu LG, Miró A, Font B, Calafell J and Lehmkuhl O** (2021) A data-driven wall-shear stress model for LES using gradient boosted decision trees. In Heike Jagode, Hartwig Anzt, Hatem Ltaief, Piotr Luszczek, *International Conference on High Performance Computing*. Springer, pp. 105–121.
- Raposo D, Santoro A, Barrett D, Pascanu R, Lillicrap T and Battaglia P** (2017) Discovering objects and their relations from entangled scene representations. *5th International Conference on Learning Representations, ICLR 2017*, Toulon, France, 2017.
- Ravuri S, Lenc K, Willson M, Kangin D, Lam R, Mirowski P, Fitzsimons M, Athanassiadou M, Kashem S, Madge S, Prudden R, Mandhane A, Clark A, Brock A, Simonyan K, Hadsell R, Robinson N, Clancy E and Arribas A** (2021) Skillful precipitation nowcasting using deep generative models of radar. *Nature* 597, 672–677. <https://doi.org/10.1038/s41586-021-03854-z>.
- Reichardt H** (1951) Vollständige darstellung der turbulenten geschwindigkeitsverteilung in glatten leitungen. *ZAMM-Journal of Applied Mathematics and Mechanics/Zeitschrift für Angewandte Mathematik und Mechanik* 31(7), 208–219.
- Sagaut P** (2006) *Large Eddy Simulation for Incompressible Flows: An Introduction*. Springer Science & Business Media, Heidelberg, Germany.
- Saito N, Pullin DI and Inoue M** (2012) Large eddy simulation of smooth-wall, transitional and fully rough-wall channel flow. *Physics of Fluids* 24(7), 075103.
- Sanchez-Gonzalez A, Heess N, Springenberg JT, Merel J, Riedmiller M, Hadsell R and Battaglia P** (2018) Graph networks as learnable physics engines for inference and control. In *International Conference on Machine Learning*. PMLR, pp. 4470–4479.
- Santoro A, Raposo D, Barrett DGT, Malinowski M, Pascanu R, Battaglia P and Lillicrap T** (2017) A simple neural network module for relational reasoning. Preprint, [arXiv:1706.01427](https://arxiv.org/abs/1706.01427).
- Scarselli F, Gori M, Tsoi AC, Hagenbuchner M and Monfardini G** (2008) The graph neural network model. *IEEE Transactions on Neural Networks* 20(1), 61–80.
- Schönfeld T and Rudgyard M** (1999) Steady and unsteady flow simulations using the hybrid flow solver AVBP. *AIAA Journal* 37(11), 1378–1385.
- Schumann U** (1975) Subgrid scale model for finite difference simulations of turbulent flows in plane channels and annuli. *Journal of Computational Physics* 18(4), 376–404.
- Serhani A, Xing V, Dupuy D, Lapeyre C and Staffelbach G** (2022) High-performance hybrid coupling of a CFD solver to deep neural networks. In *33rd International Conference on Parallel Computational Fluid Dynamics*, Alba, Italy.
- Simpson RL** (1989) Turbulent boundary-layer separation. *Annual Review of Fluid Mechanics* 21(1), 205–232.
- Smagorinsky J** (1963) General circulation experiments with the primitive equations: I. the basic experiment. *Monthly Weather Review* 91(3), 99–164.
- Stull DR and Prophet H** (1971) JANAF thermochemical tables, 2nd edition. Technical Report NSRDS-NBS 37, US National Bureau of Standards.
- Sukhbaatar S, Sziám A and Fergus R** (2016) Learning multiagent communication with backpropagation. *Advances in Neural Information Processing Systems* 29, 2244–2252.
- Temmerman L, Hadžiabdić M, Leschziner MA and Hanjalić K** (2005) A hybrid two-layer URANS–LES approach for large eddy simulation at high Reynolds numbers. *International Journal of Heat and Fluid Flow* 26(2), 173–190.
- Templeton JA, Wang M and Moin P** (2006) An efficient wall model for large-eddy simulation based on optimal control theory. *Physics of Fluids* 18(2), 025101.
- Templeton JA, Wang M and Moin P** (2008) A predictive wall model for large-eddy simulation based on optimal control techniques. *Physics of Fluids* 20(6), 065104.
- Van Steenkiste S, Chang M, Greff K and Schmidhuber J** (2018) Relational neural expectation maximization: Unsupervised discovery of objects and their interactions. *6th International Conference on Learning Representations, ICLR 2018*, Vancouver, BC, Canada.
- Watters N, Tacchetti A, Weber T, Pascanu R, Battaglia P and Zoran D** (2017) Visual Interaction Networks: Learning a Physics Simulator from Video. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, & R. Garnett (Eds.), *Advances in Neural Information Processing Systems* (Vol. 30), Curran Associates, Inc., New York, U.S.A., 2017.
- Witherden FD, Farrington AM and Vincent PE** (2014) PyFR: An open source framework for solving advection–diffusion type problems on streaming architectures using the flux reconstruction approach. *Computer Physics Communications* 185(11), 3028–3040.
- Yang XIA and Griffin KP** (2021) Grid-point and time-step requirements for direct numerical simulation and large-eddy simulation. *Physics of Fluids* 33(1), 015108.
- Yang XIA, Sadique J, Mittal R and Meneveau C** (2015) Integral wall model for large eddy simulations of wall-bounded turbulent flows. *Physics of Fluids* 27(2), 025112.

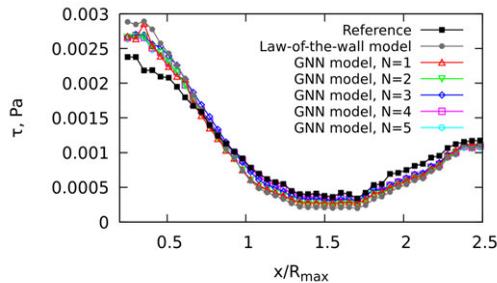
- Yang XIA, Zafar S, Wang J-X and Xiao H** (2019) Predictive large-eddy-simulation wall modeling via physics-informed neural networks. *Physical Review Fluids* 4(3), 034602.
- Zambonini G, Ottavy X and Kriegseis J** (2017) Corner separation dynamics in a linear compressor cascade. *Journal of Fluids Engineering* 139(6).
- Zangeneh R** (2021) Data-driven model for improving wall-modeled large-eddy simulation of supersonic turbulent flows with separation. *Physics of Fluids* 33(12), 126103.
- Zhou J, Cui G, Hu S, Zhang Z, Yang C, Liu Z, Wang L, Li C and Sun M** (2020) Graph neural networks: A review of methods and applications. *AI Open* 1, 57–81.
- Zhou Z, He G and Yang X** (2021) Wall model based on neural networks for LES of turbulent flows over periodic hills. *Physical Review Fluids* 6(5), 054610.

Appendix A. Influence of the Number of Message-Passing Steps

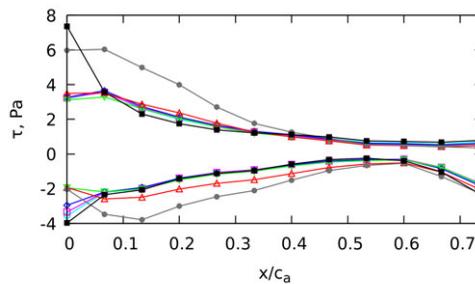
This section studies a priori the influence of the number of message-passing steps N on the model performance. Graph neural network (GNN) wall-shear-stress models with N from 1 to 5 have been trained. Figure A1 shows the mean wall shear stress (WSS) prediction in each case. A very low number of message-passing steps ($N = 1$) is found unable to accurately discriminate separated regions as in particular the WSS increase in the backflow region of the BFS simulation ($x/h_s = 2-4$) is not reproduced by the model. Gradual improvements are observed as larger values of N are used. In the case of the backward-facing step (BFS) simulation, the mean predicted WSS in the separated region is greatly increased from $N = 1$ to $N = 3$, while the improvements from $N = 3$ to $N = 5$ are more limited. In the adverse-pressure-gradient (APG) simulation, more accurate results are obtained throughout the domain with $N = 4$ or 5 message-passing steps. In the NACA 65-009 blade simulation (N65b), the mean WSS prediction slowly converges as the number of message-passing steps is increased, so that almost identical results are obtained with $N = 3$, $N = 4$, or $N = 5$. These results support the choice of the baseline value $N = 4$ in the present study.



(a) Backward-facing step (BFS)



(b) Adverse pressure gradient (APG)



(c) NACA 65-009 blade cascade (N65b) (†)

Figure A1. Average prediction of a model based on Reichardt's law and of graph neural network WSS models with several numbers of message-passing steps N . The average is performed in both time and spanwise direction in the BFS simulation (a), the APG simulation (b), and the N65b simulation (c) on the blade surface. On the blade surface of the N65b simulation, the mean wall shear stress is for clarity reported with positive values for the suction side of the blade and negative values for the pressure side of the blade.

Appendix B. Computational Cost of the Model

The implementation and computational cost of the GNN WSS model has been discussed in Serhani et al. (2022). In particular, we studied the computational cost of the model for the NACA 65-009 simulation. The simulation has been benchmarked on the Jean-Zay super-computing cluster with hybrid nodes containing 40 CPU cores (Intel Cascade Lake 6248 2.5 GHz) and 4 GPUs (Nvidia Tesla V100 SXM2 32GB). The GNN prediction is executed exclusively on the GPUS, concurrently to parts of the LES schemes. We define the deep-learning (DL) overhead as

$$\begin{aligned} \text{DL overhead} = & \text{Cost of wall - shear - stress prediction} \\ & + \text{cost of communications} \\ & - \text{cost of overlapped LES computations.} \end{aligned} \quad (22)$$

It represents the additional cost induced by the addition of the GNN WSS model on a given hybrid node. Figure B1 gives the overhead of the model (DL overhead) for 2–16 hybrid nodes. The deep learning overhead ranges from 0 to 11% of the total iteration cost in that case. Future works should consider the computational cost in the model development.

Appendix C. Wall-Resolved Simulation of the NACA 65-009 Blade Cascade

This section presents the wall-resolved simulations of the NACA 65-009 blade cascade, used to train the GNN WSS model and for a priori and a posteriori tests. The geometric configuration and operating point conditions of the wall-resolved simulations are the same as used for the large-eddy simulations (LESs) with a wall-shear-stress model (Section 4.2.4, Table 6). Two wall-resolved LES have been performed: with an incidence angle $I = 4^\circ$ (N65a) and with an incidence angle $I = 7^\circ$ (N65b). In both cases, the corresponding experimental velocity profiles of Zambonini et al. (2017) are imposed at the inlet, using the characteristic nonreflecting boundary condition of Poinso and Lele (1992). At the outlet, a constant pressure of 1 standard atmosphere is imposed using the nonreflecting boundary condition of Granet et al. (2010). A symmetric boundary condition is used at the top boundary of the domain and a no-slip boundary condition is used on the blade surface and endwall. The computational domain is periodic in the pitchwise direction. The governing equations are the compressible LES equations given in Section 4.2.1. These equations are discretized and advanced in time using a cell-vertex finite-element method with a two-step Taylor–Galerkin scheme of third-order spatial and temporal accuracy (Colin and Rudgyard, 2000) for convection and a second-order centered scheme for diffusion. The numerical simulations are performed using the hybrid flow solver AVBP (Schönfeld and Rudgyard, 1999). The same mesh is used for the two wall-resolved LESs. The mesh is hybrid: the near-wall regions are discretized using triangular prisms and the rest of the domain using tetrahedra. There are 27 prismatic layers on the blade and endwall surface. The prismatic layers on the endwall end at $x/c = 2.4$. Pyramidal cells are used to smoothly transition to tetrahedral cells. The number of prisms is 85×10^6 , the number of pyramids is 1×10^4 and the number of tetrahedra is 188×10^6 . The total number of cells is 273 million. The height of the first point off the wall is $\Delta z^+ = 0.7$ on the incoming endwall boundary layer, and below 1 everywhere except at the leading edge

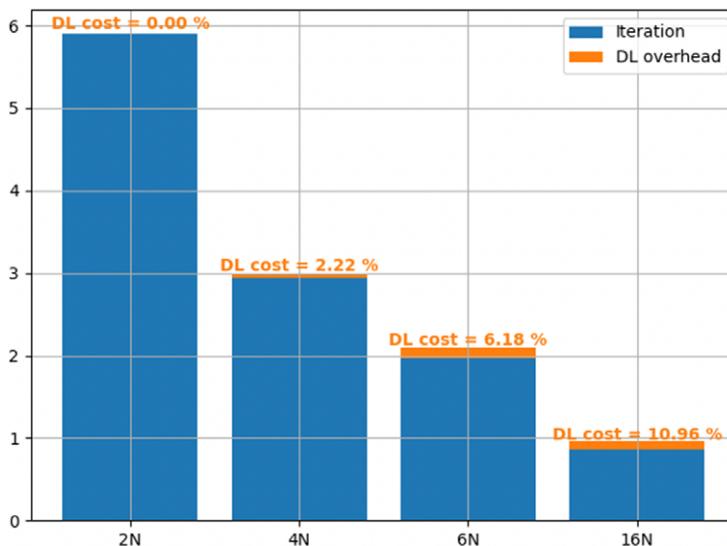


Figure B1. Overhead of the deep-learning wall-shear-stress model per iteration for 2, 4, 6, and 16 hybrid nodes.

of the blade and near the tripping steps, which induce a small separation region. Averaged flow statistics are computed over 10 inlet flow characteristic times c/U_{inlet} .

An overview of the flow is given in Figure C1. The case is characterized by a complex boundary-layer dynamics on the surface of the blade. While the boundary layer is laminar near the leading edge of the blade, transition to turbulence is induced by rectangular tripping steps which aim to reproduce the effect of the experimental sandpapers. There is a three-dimensional separation region in the corner between the suction side of the blade and the endwall. The size of the corner separation region depends on the incidence angle of the incoming boundary layer, as investigated by Ma et al. (2011). The distribution of the mean static pressure coefficient C_p on the blade surface is compared to the reference results of Gao et al. (2015) with $I = 4^\circ$ in Figure C2. The present wall-resolved LES reproduces accurately the expected profiles, both near the endwall and at mid-height. The distribution of C_p with $I = 7^\circ$, also reported in Figure C2, exhibits a larger plateau in the aft part of the suction side, as the corner separation region is larger in that case. Figure C3 compares the coefficient of stagnation pressure loss ω_s downstream of the trailing edge to the reference results of Gao et al. (2015) with $I = 4^\circ$ and Zambonini et al. (2017) with $I = 7^\circ$. The increase in the extent of losses region at the larger incidence angle $I = 7^\circ$ compared to $I = 4^\circ$ is qualitatively reproduced by the wall-resolved LESs. However, the amplitude of the pressure losses is for both incidence angle lower than in the experimental measurements. Finally, Figure C4 compares the evolution of the tangential velocity profiles on the suction side of the blade to the numerical results and experimental laser-Doppler anemometry (LDA) and

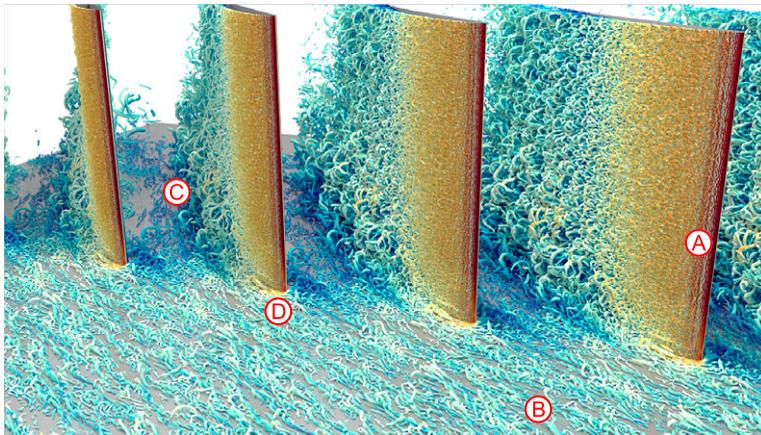


Figure C1. Overview of the flow in the wall-resolved simulation of the linear blade cascade with an incidence angle of 7° . The colors are isocontours of Q -criterion. The flow has been duplicated along the periodic pitchwise direction for clarity. (A): Rectangular tripping step inducing turbulence transition on the blade. (B): Incoming boundary layer. (C): Region of corner separation on the suction side of the blade. (D): Horseshoe vortex at the leading edge of the blade.

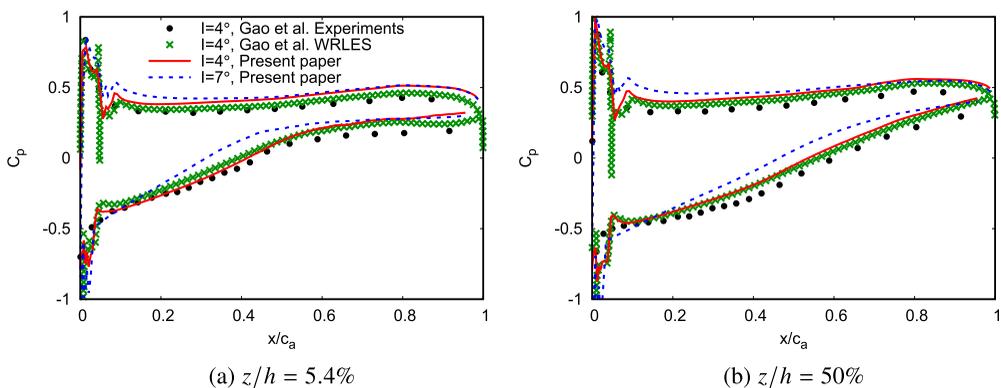


Figure C2. Mean static pressure coefficient C_p on the blade close to the endwall (a) and at mid-height (b), compared to the experimental and numerical results of Gao et al. (2015) with an incidence angle of 4° .

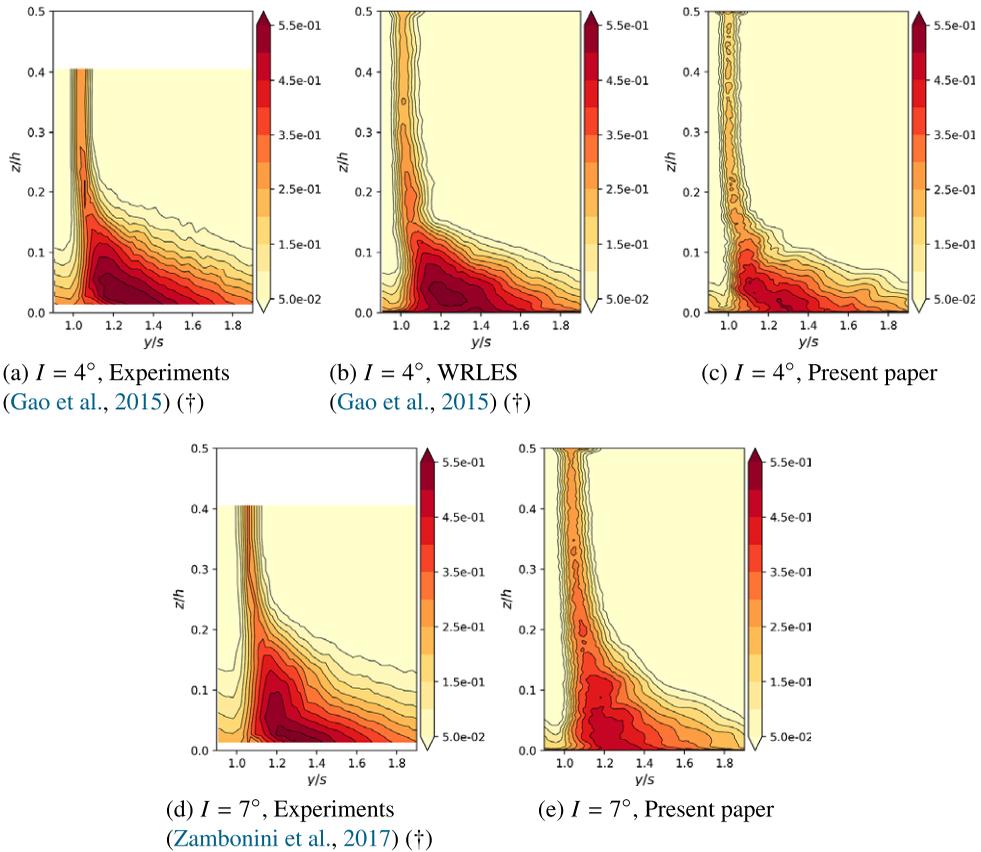


Figure C3. Stagnation pressure loss coefficient ω_s at 36.3% axial chord downstream from trailing edge, compared to the experimental and numerical results of Gao et al. (2015) with an incidence angle of 4° and to the experimental results of Zambonini et al. (2017) with an incidence angle of 7° .

particle image velocimetry (PIV) measurements of Gao et al. (2015), with an incidence angle of 4° . A good agreement is found between the present wall-resolved LES and the references. At mid-height, the increase in boundary layer thickness from the leading edge of the blade to the trailing edge is well reproduced by our simulation. The corresponding decrease of the peak tangential velocity toward the trailing edge of the blade is also consistent between the present simulation and the reference WRLES, PIV, and LDA results. Close to the endwall, the boundary layer is attached in the accelerating region near the leading edge but is detached in the corner separation region near the trailing edge. There is a qualitative agreement between the boundary layer profiles in both regions. However, the onset of the separated region in the present simulation is at $s_1 = 0.68$, which is closer to the trailing edge than in the wall-resolved simulation of Gao et al. (2015). Accordingly, the peak tangential velocity upstream and downstream of the separation region is slightly overestimated and underestimated, respectively, compared to this prior simulation. With a larger incidence angle of 7° , the onset separated region at $z/h = 1.4\%$ is closer to the leading edge, near the station $s_1 = 0.5$ (Figure C4c), since the velocity gradient at this location is close to zero. The separated region also extends far from the wall in that case, since at mid-height the boundary layer close to the trailing edge of the blade is nearly separated (Figure C4f). Overall, the present numerical simulations recover the blade boundary-layer dynamics reported in the literature and capture accurately the impact of the incidence angle on the corner separation.

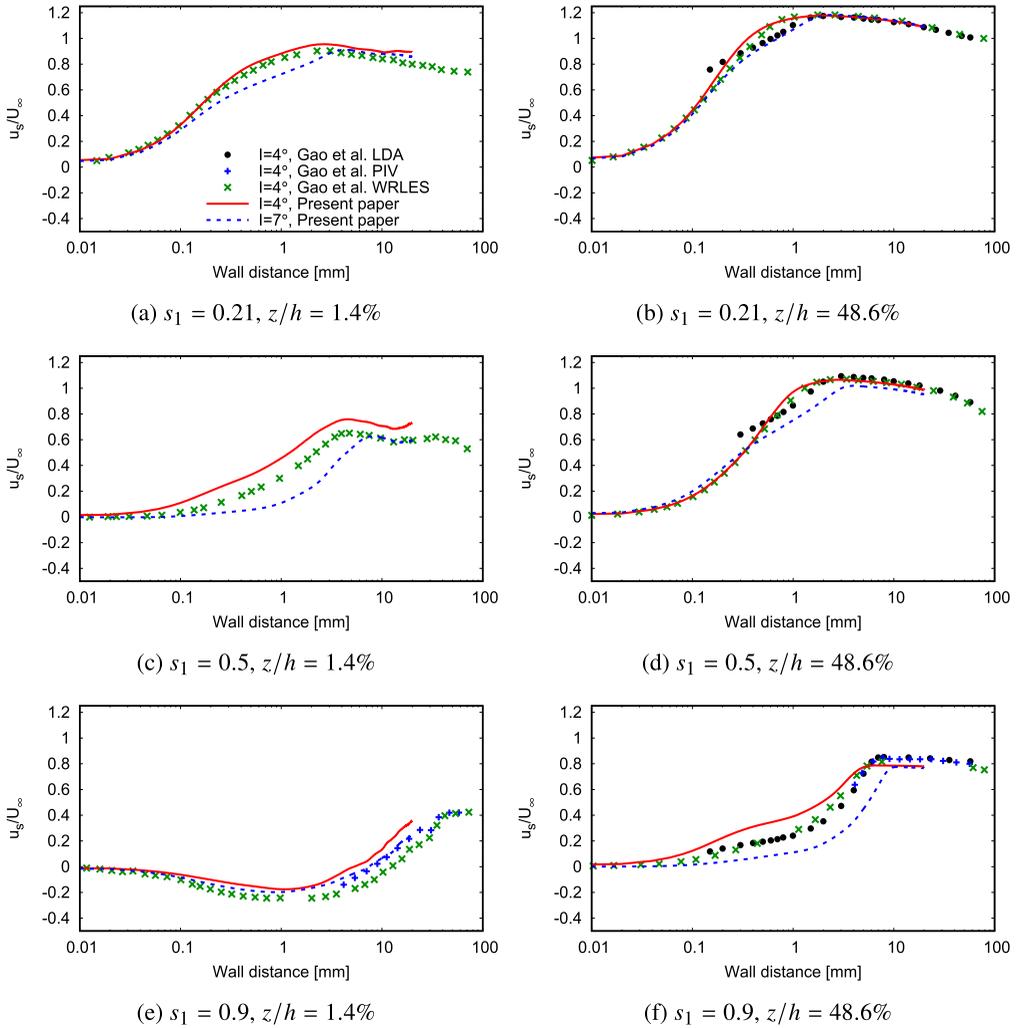


Figure C4. Profile of tangential velocity on the suction side of the blade close to the endwall (a, c, e) and at mid-height (b, d, f), compared to the laser-Doppler anemometry (LDA), particle image velocimetry (PIV), and numerical WRLES results of Gao et al. (2015) with an incidence angle of 4° .