

Reconstruction of Reynolds-averaged Navier-Stokes solutions from boundary data with partial Reynolds stress transport constraints

Jincheng Zhang ^{*}*School of Engineering, University of Warwick, Coventry CV4 7AL, United Kingdom*

(Received 17 February 2026; accepted 15 September 2026; published 6 October 2026)

Physics-informed neural networks (PINNs), a scientific machine learning framework for solving inverse and forward problems involving partial differential equations, are seeing many successes in fluid research recently. Based on PINNs, this paper develops a physics-informed framework for reconstructing incompressible Reynolds-averaged Navier-Stokes solutions from boundary data through the incorporation of Reynolds stress transport (RST) formulations without introducing empirical models for the unclosed terms. Specifically, the RST equations are first formulated with the exact forms for the advection and production terms as derived from the Reynolds averaging process while the unclosed terms that need empirical modeling in traditional turbulence models are absorbed into a single inadequacy term. The resulting underdetermined systems are then embedded via the PINNs approach to reconstruct turbulent flow fields based on the mean velocity components and Reynolds stress components at the domain boundary. A mechanism of enforcing the realizability condition for the Reynolds stress is further proposed to enhance the reconstruction performance, via a customized neural network design. The proposed approach is applied and evaluated via a set of flow cases, including a zero-pressure-gradient turbulent boundary layer flow, an adverse-pressure-gradient turbulent boundary layer flow, and a set of flow cases over periodic hills of parameterized geometries. The results show that for all the test flow cases, the mean flow quantities are reconstructed very accurately, and a decent accuracy is obtained for the turbulence quantities including both the normal and shear Reynolds stress components.

DOI: [10.1103/rn1j-32n4](https://doi.org/10.1103/rn1j-32n4)

I. INTRODUCTION

The fast development of machine learning (ML), in particular deep learning, has brought a lot of exciting successes in fluid dynamics in recent years, including both fundamental research and engineering applications [1,2]. Due to machine learning's strong capability in handling systems of high nonlinearity and extreme complexity, it has shown great potential in revolutionizing many subfields of fluid research, including surrogate modeling of complex flows [3–5], uncertainty quantification [6–8], flow control [9–11], computational fluid dynamics [12,13], and multiscale and multiphysics flow modeling [14,15]. In particular, a significant amount of works have been carried out on the data-driven turbulence modeling using machine learning techniques [16,17]. In Ref. [18], a tensor basis deep neural network (TBNN) was developed for Reynolds-averaged Navier-Stokes

^{*}Contact author: jincheng.zhang.1@warwick.ac.uk

Published by the American Physical Society under the terms of the [Creative Commons Attribution 4.0 International](https://creativecommons.org/licenses/by/4.0/) license. Further distribution of this work must maintain attribution to the author(s) and the published article's title, journal citation, and DOI.

(RANS) turbulence modeling, which embedded Galilean invariance into the network architecture to ensure the predicted Reynolds stress anisotropy tensor is on an invariant tensor basis. The TBNN and its follow-up studies, e.g., Ref. [19], are built on Pope’s general eddy-viscosity model [20], where the Reynolds stress anisotropy is expressed as a local, algebraic function of the mean strain-rate and rotation-rate tensors, nondimensionalized by the turbulent timescale k/ϵ . Such models therefore require the k and ϵ (or ω) being computed from an underlying two-equation baseline RANS model, and are typically deployed as a local correction on top of that baseline closure rather than being embedded within the Reynolds stress transport equation itself. Because the mapping from local mean-flow invariants to Reynolds stress anisotropy is generally not unique, as different flow histories can produce the same local strain/rotation-rate state but different Reynolds stress anisotropies, this local, algebraic formulation is also subject to a “multivalued” mapping issue [19]. In Ref. [21], a machine learning approach for predicting Reynolds stress discrepancies arising from traditional RANS models was developed, where a set of mean flow features derived from RANS simulations were carefully chosen as the ML input. The discrepancy between the direct numerical simulation (DNS) data and the RANS-predicted Reynolds stress was decomposed into physically meaningful projections, i.e., the turbulent kinetic energy giving the stress magnitude, the anisotropy eigenvalues represented through the barycentric triangle giving its shape, and the eigenvector orientation angles, with each projection’s discrepancy predicted separately as the ML output. In Ref. [22], three ways of introducing perturbations to Reynolds stress tensor were investigated, aiming at providing further insights toward quantification of model-form uncertainties in RANS simulations and assisting development of ML-based turbulence models. In Ref. [23], the Reynolds force vector instead of the Reynolds stress tensor was proposed as the learning target, aiming at addressing the error amplification from the corrected Reynolds stress tensor to the predicted mean flow quantities in the data-assisted turbulence model. In Ref. [24], a vectorial source term was added to correct the RANS-predicted flow field toward its high-fidelity reference via data assimilation, and the rotational invariance of the vectorial output was accounted via data augmentation. As turbulent flows are of particular interest for many engineering applications, more and more works on machine learning based turbulence modeling are still emerging, such as the study of wall-bounded turbulence via explainable deep learning approach based on Shapley additive explanation (SHAP) values [25] and the use of only sparse data for RANS model correction [26].

From the perspective of machine learning models, in most of the aforementioned works, physical insights related to the underlying fluid flows are often taken into account through the problem formulation, the model construction, and/or the design of the training input and the training target, while the training process (i.e., the updating of the training variables via an optimizer to minimize the loss function) is still purely driven by data. The recent development of physics-informed neural networks (PINNs), as introduced in Ref. [27] for solving inverse and forward problems involving nonlinear partial differential equations, opens a new paradigm of machine learning for fluid dynamics. Due to the versatility of PINNs in fusing various forms of information through composing the loss function, it has attracted a lot of attention across the research spectrum, from data-driven, data and physics fusion, and to purely physics-informed learning without any simulation data inside the flow domain. In Ref. [28], PINNs were used to infer the vortex-induced vibrations based on scattered sparse flow measurements and Navier-Stokes (NS) equations. In Ref. [29], a general framework was developed to infer velocity and pressure fields from flow visualizations. It was applied to a range of physical and biomedical flows and the results showed that accurate and robust prediction of flow fields was achieved based on flow visualization and NS equations. In Ref. [30], PINNs were used to integrate the remote sensing Lidar technologies and the NS equations to tackle the in situ prediction of turbulent wind farm wake flows in the atmospheric boundary layer. In Ref. [31], the 2D and 3D flow fields of blood flows were estimated based on the 2D images and the NS equations. In Ref. [32], PINNs were used to solve two forms of NS equations (i.e., the velocity-pressure formulation and the vorticity-velocity formulation) and applied to both laminar and turbulent flows, where the prediction accuracy and convergence were extensively investigated and the scenarios of complete boundary conditions and missing/noisy boundary conditions were both studied. In Ref. [33], turbulent natural

convection flows were investigated where a set of computational techniques were proposed to mitigate the training difficulties of PINNs for this complex turbulent flow. With more and more works on the applications and method developments of PINNs [34–36], understanding why and when PINNs fail is also gaining attention [37–39].

Most of the PINNs studies for predicting fluid flows employed the NS equations as the governing equation. Although it is demonstrated that both laminar and turbulent flows can be handled [32,33], it requires careful training and data settings to predict turbulent flows. However, for turbulent flows, the averaged flow quantities are often the quantities of interest in many engineering applications. Thus the natural next direction is to incorporate RANS equations into the PINNs framework for predicting turbulent flows. In Ref. [40], a comprehensive comparison study was carried out, where the standard k - ω model, the mixing length model, an equation-free ν_t and an equation-free pseudo-Reynolds stress model were implemented using the PINNs approach for RANS simulation of the backward-facing step flow. In Ref. [41], the standard k - ϵ turbulence model was applied for RANS simulations based on PINNs where a pretraining step was employed using data-driven loss. More recent developments in applying PINNs to RANS include the application to transonic flows around a cylinder [42] and compressor cascade flow [43], the inverse parameter identification of RANS model [44], as well as the turbulent flow reconstruction based on PINNs without RANS model [45] and with the RANS model, e.g., Spalart-Allmaras model [46]. It is worth noting that the flow reconstruction problem tackled in Ref. [46], while also based on PINNs, differs from the present work in its ill-posedness. The flow field is reconstructed from scattered measurement points distributed across the flow domain (a coarse-grid super-resolution problem), rather than from information available only at the domain boundary as considered here. In addition, the Spalart-Allmaras model used in Ref. [46] was specifically modified to remove nondifferentiable operations and to improve PINNs trainability, e.g., the $1/d^2$ wall-distance singularity and a clipping operation on the eddy viscosity. The network reconstructs the mean flow velocity together with the divergence-free component of the Reynolds-stress forcing vector and the Spalart-Allmaras model's transport variable, rather than the full Reynolds stress tensor. As a result, the irrotational component of the forcing cannot be separated from the mean pressure, and the Reynolds stress tensor cannot be reconstructed by design. The research in employing RANS equations within the PINNs framework is becoming more and more active.

Because the Reynolds averaging process leads to an under-determined system, the Reynolds stress closure, a vital subfield of turbulence research for many decades, needs to be tackled to solve the RANS equations by numerical approaches. However, the complete modeling of Reynolds stress is not compulsory in the PINNs framework. This is particularly true in ill-posed problems such as flow reconstruction. In particular, in Ref. [47], the RANS solutions were inferred using PINNs approach where the Reynolds stresses were left unmodeled and simply treated as the neural network output. The results in Ref. [47] demonstrated that the turbulent flows can be successfully reconstructed based on only the information (including both the mean flow quantities and second-order statistics) at the domain boundaries and the RANS equations without any specific model for turbulence.

In this work, different from the work in Ref. [47] which left the Reynolds stress completely unmodeled, the RANS equations, along with the Reynolds stress transport (RST) formulations, are incorporated into the PINNs framework, where the advection term and the production term of the Reynolds stress, which can be directly derived from the Reynolds averaging process and do not require any empirical modeling, are used in the designed PINNs model, while all the other terms that require extensive empirical modeling efforts from domain experts, are left unmodeled and absorbed through an inadequacy term. This leads to the proposed baseline Reynolds stress transport net (RSTnet), which is designed to embed the RANS equations through a PINNs structure that is informed by as much physical information as possible while staying as simple as possible to ensure training stability and generalizability. Next, to further improve the performance, mechanisms of imposing the realizability condition of the Reynolds stress tensor are proposed and incorporated into the RSTnet. This leads to two variants of the proposed RSTnet, where in the first variant

the nonnegativity of the normal stress components of the Reynolds stress tensor is imposed in the neural network design, while in the second variant the realizability conditions, including both the nonnegativity of the normal stress components and the Cauchy-Schwarz inequality for the shear stress components, are imposed in the neural network design. To evaluate the performance of the proposed RSTnet and its two variants, a set of flow cases are investigated, including a zero-pressure-gradient (ZPG) turbulent boundary layer flow, an adverse-pressure-gradient (APG) turbulent boundary layer flow, and a set of flow cases over parameterized periodic hills. For each of these flow cases, the influence of the neural network hyperparameters and the convergence with respect to the training boundary points and collocation points are first studied. Then the comparison between the baseline RSTnet and its two variants is carried out to demonstrate the importance of the proposed mechanism for enforcing Reynolds stress realizability. Finally the influence of the weighting coefficients for the individual loss terms forming the total loss, which was shown to have large impact on the training process in previous works [32], is studied.

It is also worth noting that the Reynolds stress closure problem, rather than an analogous closure problem for a passive scalar (e.g., species concentration) transport equation, is chosen as the focus of this work as it is the more consequential and more extensively studied of the two in the turbulence modeling literature. However, the proposed methodology is not restricted to the Reynolds stress transport equation, and its extension to other averaged transport equations with an unclosed flux term is also of great interest and is discussed in Sec. IV as a direction for future work.

In the rest of the paper, the methodology is described in Sec. II. Then the applications to the ZPG and APG turbulent boundary layer flows and the flows over periodic hills are given in Sec. III. The conclusions are drawn in Sec. IV.

II. METHODOLOGY

This section presents the proposed RSTnets. To avoid confusion, it is worth noting that the RSTnets are designed to explore a new way of reconstructing RANS solutions from boundary data by fusing various physical principles without any empirical models, rather than developing new data-driven RANS closure models. In the following part, the PINNs framework, a framework of using neural networks to solve nonlinear partial differential equations, is briefly described first. Then the RANS equations with RST formulations are introduced, where the turbulence advection and production terms are directly derived from the Reynolds averaging process and all the other terms are absorbed through an inadequacy term. Next, the RSTnets, which incorporate the RANS-RST formulations and impose the realizability conditions of the Reynolds stress tensor in the neural network design, are described. Finally the evaluation metrics are given.

A. Physics-informed neural networks

PINNs, different from supervised machine learning approaches, can predict the flow field in the computational domain without any training target inside the domain. This is achieved by using the residual of the governing PDEs, evaluated at the neural network output via automatic differentiation, as a physics-informed loss term [27], which effectively transforms the problem of solving the equation systems into an optimization problem of training the neural networks. The training process guides the neural networks to find the solution of the incorporated governing equations, by updating the training weights so that the neural network output obeys the constraints imposed by both the governing equations and the boundary/initial conditions.

To approximate a complex flow system within PINNs, a deep neural network with a large number of degrees of freedom is usually required, which is designed to take the spatiotemporal coordinates as the input and return the flow states as the output. It can be expressed as

$$\mathbf{y} = \mathcal{N}_w(\mathbf{x}), \quad (1)$$

where $\mathcal{N}_W$ represents the deep neural network with the training variable W , $\mathbf{x}$ is the neural network input which can be specified as the spatial coordinate $[x, y, z]$ for steady flows and the spatiotemporal coordinate $[x, y, z, t]$ for unsteady flows, and $\mathbf{y}$ is the neural network output which can be specified as the flow quantities $[\mathbf{u}, p]$ for the NS equations or $[\mathbf{u}, p, \boldsymbol{\tau}]$ for the RANS equations. Here $\mathbf{u}$ represents the velocity vector, $\boldsymbol{\tau}$ represents the Reynolds stress tensor, and p represents the pressure normalized by the fluid density. It is worth mentioning that the input coordinate here is continuous rather than discrete like in many other deep learning models, e.g., convolutional neural networks and recurrent neural networks.

The governing equations describing the fluid flows, such as the NS equations and the RANS equations, can be expressed in the general form as

$$\mathcal{F}(\mathbf{y}) = 0, \mathbf{x} \in \Omega, \quad (2)$$

where $\mathcal{F}$ represents the nonlinear differential operator and Ω is the input spatial and/or temporal domain. To incorporate such a PDE system into the PINNs, the physics-informed loss term is first derived based on the neural network $\mathcal{N}_W$ using automatic differentiation. The first-order derivative can be obtained via the automatic differentiation of the neural network output $\mathbf{y}$ with respect to the neural network input $\mathbf{x}$. For example, using Tensorflow, the derivative of the j th component of the output with respect to the i th input dimension can be obtained via `tf.gradients(yj, xi)`. The higher-order derivatives in the operator $\mathcal{F}$ can be derived similarly based on the low-order derivatives. All the terms in $\mathcal{F}$ are finally added together to form the physics-informed residual term, which are hereby expressed as $\mathcal{F}_W(\mathbf{x})$. This term, which depends on the same training variable W and can be seen as the representation of the operator $\mathcal{F}$ via the neural network $\mathcal{N}$, returns the PDE residuals as the output given any coordinate $\mathbf{x}$ as the input. After constructing the physics-informed part, the training loss function is finally composed as

$$\mathcal{L} = \mathcal{L}_e + \mathcal{L}_b, \quad (3)$$

where

$$\mathcal{L}_e = \frac{1}{N_e} \sum_{i=1}^{N_e} |\mathcal{F}_W(\mathbf{x}^i)|^2 \quad (4)$$

and

$$\mathcal{L}_b = \frac{1}{N_b} \sum_{j=1}^{N_b} |\mathcal{N}_W(\mathbf{x}^j) - \mathbf{y}^*(\mathbf{x}^j)|^2. \quad (5)$$

Here $\mathcal{L}_e$ is the physics-informed loss where $\{\mathbf{x}^i, 1 \leq i \leq N_e\}$ is the collocation points for enforcing the PDE constraints, and $\mathcal{L}_b$ is the supervised part where the boundary/initial conditions at the points $\{\mathbf{x}^j, 1 \leq j \leq N_b\}$ are imposed through minimizing the mean square error (MSE) between the neural network output and the corresponding reference values $\{\mathbf{y}^*(\mathbf{x}^j), 1 \leq j \leq N_b\}$. Finally, the training is carried out via an optimizer (e.g., the Adam optimizer) to minimize the composed loss $\mathcal{L}$. Through the training process, the PINNs learn the solution inside the flow domain based on the boundary and initial conditions (i.e., the supervised learning part) and the incorporated governing equations (i.e., the physics-informed learning part).

B. RANS equations with Reynolds stress transport formulations

Following the standard Reynolds averaging procedure, the instantaneous velocity and pressure fields are decomposed into a mean and a fluctuating component, i.e., $\tilde{\mathbf{u}} = \mathbf{U} + \mathbf{u}'$ and $\tilde{p} = P + p'$, where the capitalized/overbarred quantities denote the Reynolds-averaged fields and the primed quantities denote the fluctuations. The Reynolds stress tensor is defined as the covariance of the velocity fluctuations, $\boldsymbol{\tau} = -\overline{\mathbf{u}'\mathbf{u}'}$, and P denotes the mean pressure normalized by the fluid density. In addition, the equations and flow quantities in this paper are all presented in nondimensional form,

normalized by characteristic length and velocity scales before being fed into the PINNs training. The RANS and RST equations presented in this section describe statistically steady, incompressible flows, and can be expressed as

$$\nabla \cdot \mathbf{U} = 0, \quad (6)$$

$$\mathbf{U} \cdot \nabla \mathbf{U} = -\nabla P + \frac{1}{\text{Re}} \nabla^2 \mathbf{U} + \nabla \cdot \boldsymbol{\tau}, \quad (7)$$

$$\mathbf{U} \cdot \nabla \boldsymbol{\tau} = \mathcal{P} + \Phi + \mathcal{D} + \mathcal{E}. \quad (8)$$

Here $\mathcal{P}$, Φ , $\mathcal{D}$, and $\mathcal{E}$ represent the production term, the pressure-strain term, the diffusion term, and the dissipation term. Among these terms, the production term $\mathcal{P}$ can be directly derived from the Reynolds averaging process as

$$\mathcal{P} = -(\boldsymbol{\tau} \cdot \nabla \mathbf{U} + (\boldsymbol{\tau} \cdot \nabla \mathbf{U})^T). \quad (9)$$

All the other terms, i.e., Φ , $\mathcal{D}$, and $\mathcal{E}$, require extensive empirical modeling and calibration to close the equation systems (6)–(8). In traditional second-moment Reynolds stress transport closures, each of these terms needs to be modeled separately [48–50]. The dissipation tensor $\mathcal{E}$ is conventionally assumed to be isotropic at high Reynolds number, i.e., $\mathcal{E} \approx \frac{2}{3}\epsilon \mathbf{I}$ [48,51], with the scalar dissipation rate ϵ supplied by a separate transport equation that itself contains further unclosed production, destruction, and diffusion terms requiring calibration against canonical flows. This is the seventh transport equation of a traditional seven-equation Reynolds stress transport model, in addition to the six Reynolds stress components. The pressure-strain correlation Φ requires empirical closure for both a slow (return-to-isotropy) part and a rapid part depending on the mean velocity gradients. In the model proposed in Ref. [48], the slow part is modeled via the linear Rotta return-to-isotropy model, and the rapid part is modeled as a linear function of the mean velocity gradients. Later nonlinear return-to-isotropy models [51–53] have been shown to improve upon the linear Rotta model across a wide range of homogeneous turbulence. More recently, data from anisotropically forced homogeneous turbulence simulations have been used to inform and assess a cubic-order decay model capturing both the pressure-strain correlation and the dissipation terms [54], with the forcing technique enabling the probe of a wide range of turbulence states. Moreover, the diffusion term $\mathcal{D}$ is typically closed via a gradient-transport term with an effective diffusivity capturing both viscous and turbulent diffusion. For a more comprehensive description of these closures, the interested readers may refer to Refs. [49,55,56].

To tackle the equation systems (6)–(8), traditional numerical approaches focus on the complete modeling of the unclosed terms as described above, while the recent work based on PINNs [47] takes a starkly different approach of leaving the Reynolds stress completely unmodeled. Different from these approaches, the present work focuses on making use of physical information as much as possible while keeping the PINNs model as simple as possible, to reconstruct the flow solution based on only the boundary data. Particularly, this work formulates the RST equation by employing the advection and production terms which can be directly derived from Reynolds averaging process and considering all the other terms which need empirical modeling into a single additive inadequacy term η . This leads to the formulation of RST equation as

$$\mathbf{U} \cdot \nabla \boldsymbol{\tau} = -(\boldsymbol{\tau} \cdot \nabla \mathbf{U} + (\boldsymbol{\tau} \cdot \nabla \mathbf{U})^T) + \eta. \quad (10)$$

It is worth noting that the diffusion term $\mathcal{D}$ contains a viscous contribution that is, in principle, exact, and thus could be separated from η without requiring any additional empirical modeling. This is not done in the present formulation. Across most of the flow domain away from the immediate viscous sublayer, viscous diffusion is often dominated by turbulent diffusion, by one to three orders of magnitude. Therefore, explicitly retaining only the viscous part in the proposed method would still leave the dominant yet unclosed turbulent diffusion contribution inside η , at the cost of introducing an additional second-order term into the RST residual. Close to the wall where viscous diffusion

becomes comparable to turbulent diffusion, this simplification is less well justified. To further tackle this issue, an interesting future research direction would be to treat the effective diffusivity, e.g., $\tilde{\nu}$ in the typical closure form $\mathcal{D} \approx \nabla \cdot (\tilde{\nu} \nabla \tau)$ adopted in standard practice [50], as an additional free, underdetermined field, rather than lumping the whole diffusion term into η . In this way, the flow field reconstruction and the inference of the eddy viscosity may be achieved simultaneously.

In this work, the RST equation is formulated as Eq. (10). A natural alternative to this formulation would be to adopt one of the traditional second-moment closures outlined above or a simpler two-equation eddy-viscosity model, instead of the inadequacy term η . This is not done here, for three reasons. First, the objective here is to reconstruct the flow field from boundary data using only relations that follow exactly from the Reynolds averaging process, so that the reconstruction is conditioned on as few modeling assumptions as possible. Adopting a closure would instead tie the reconstructed solution to the particular functional form and empirical coefficients of that closure, together with the associated model-form uncertainty [16], which is a different problem from the one posed here. Second, embedding such closures within PINNs raises practical difficulties, as many of them involve nondifferentiable operations, e.g., min/max limiters and near-wall damping functions, which are well known to complicate PINNs training and convergence. For example, the modified Spalart-Allmaras model used in Ref. [46] required removing the $1/d^2$ wall-distance singularity by multiplying the governing equation by d^2 , and replacing a clipping operation on the eddy viscosity with a smooth reparametrization, to improve PINNs trainability. Similarly, the work in Ref. [40] found that introducing additional, more complex transport-equation-based closures can impede PINNs optimization.

Third, and independently of these considerations, the quantity to be reconstructed in this work, in addition to the mean flow quantities, is the full Reynolds stress tensor, which eddy-viscosity closures cannot accurately provide. A one-equation model such as the Spalart-Allmaras model transports only a scalar eddy-viscosity-like quantity and never solves for the turbulent kinetic energy k . Therefore only the deviatoric part of the Boussinesq relation, $\tau \approx 2\nu_t \mathbf{S}$, is available. This deviatoric estimate is self-consistent for the shear stress, as the missing isotropic contribution only affects the diagonal. The corresponding normal-stress estimates, however, are missing the isotropic $\frac{2}{3}k\mathbf{I}$ contribution entirely, which is typically the dominant part of the normal stress magnitude. Even a two-equation model that does transport k still relies on the linear Boussinesq assumption, $\tau \approx 2\nu_t \mathbf{S} - \frac{2}{3}k\mathbf{I}$, which returns near-isotropic normal stresses whenever the normal mean strain rates are small compared to the mean shear, as is the case for the shear-dominated ZPG and APG boundary layer flows considered in this work. A related but distinct anisotropy limitation of linear eddy-viscosity closures arises from nonequilibrium effects associated with flow separation and recirculation, such as in the periodic hill flows considered here. All these anisotropy limitations are well known for linear eddy-viscosity models, and it is also one of the main motivations driving the tensor-basis and other nonlinear closure developments in turbulence modeling [16,57].

By contrast, the advection, production, and realizability constraints incorporated into the proposed RSTnet are all exact consequences of the Reynolds averaging process and the mathematical requirements on the Reynolds stress tensor, involving no flow-specific calibration and no nondifferentiable operations. They are therefore applied unchanged across all the test cases considered in this work, spanning the attached boundary layers and the separated flows over periodic hills. The proposed formulation also requires fewer transport equations. A dedicated transport equation for the dissipation rate ϵ (or the specific dissipation rate ω) is required in the traditional seven-equation models to provide ϵ to the closure models for Φ and $\mathcal{E}$ described above. Since these terms are now absorbed into the single inadequacy term η , no unclosed term remains for an ϵ -equation (or ω -equation) to feed. Therefore it is omitted entirely in the proposed RST formulation.

In general, it is also straightforward to model the inadequacy as a multiplicative term, which leads to

$$\mathbf{U} \cdot \nabla \tau = -(\tau \cdot \nabla \mathbf{U} + (\tau \cdot \nabla \mathbf{U})^T)(\mathbf{I} + \eta). \quad (11)$$

However, numerical tests, as given in Appendix A, show that the implementation of multiplicative formulation via PINNs performs worse. Therefore the additive formulation is chosen in this work.

C. Reynolds stress transport net

Based on the equations formulated in Sec. II B, the RSTnets are developed which incorporate the RANS equations [i.e., Eqs. (6) and (7)] along with the RST formulation [i.e., Eq. (10)] via the PINNs approach. The overall structure of the RSTnets is illustrated in Fig. 1, which includes the baseline RSTnet for the general two-dimensional flow setup and its two variants by considering the realizability conditions of Reynolds stresses. As shown in Fig. 1(a), a fully connected deep neural network is first constructed which takes the spatial coordinates as the input and returns the mean flow and turbulence quantities as the output. The tanh activation is applied for the intermediate layers and no activation is applied for the output layer. After constructing the deep neural network $\mathcal{N}_w$, the physics-informed part is derived via automatic differentiation which outputs the residuals of RANS equations (i.e., $[e_{\text{div}}, e_U, e_V]$) and the residuals of RST equations (i.e., $[e_{\overline{uu}}, e_{\overline{uv}}, e_{\overline{vv}}]$). A uniform hidden-layer width is used in this work, rather than an autoencoder-type bottleneck architecture. As noted in Sec. II A, the input to the PINNs is the continuous spatial coordinate rather than a discretized representation of the flow field such as an image or a mesh-based snapshot in convolutional neural networks. An autoencoder-type bottleneck, which is often the preferred choice in CNN-based architectures, is therefore not motivated here as it will limit the representation capability of PINNs. This is also consistent with the uniform-width architectures commonly adopted in existing PINNs studies [32–34, 45–47]. The hidden-layer width and depth are then tuned directly via the hyperparameter study reported for each flow case (e.g., Table I).

Next, the training loss function is composed as [the superscript * denotes the reference value of a quantity at the domain boundary, as introduced in Eq. (5)]

$$\mathcal{L} = \alpha_1 \mathcal{L}_{e,\text{mean}} + \alpha_2 \mathcal{L}_{e,\text{turb}} + \beta_1 \mathcal{L}_{b,\text{mean}} + \beta_2 \mathcal{L}_{b,\text{turb}}, \quad (12)$$

$$\mathcal{L}_{e,\text{mean}} = \frac{1}{N_e} \sum_{i=1}^{N_e} (|e_{\text{div}}(\mathbf{x}^i)|^2 + |e_U(\mathbf{x}^i)|^2 + |e_V(\mathbf{x}^i)|^2), \quad (13)$$

$$\mathcal{L}_{e,\text{turb}} = \frac{1}{N_e} \sum_{i=1}^{N_e} (|e_{\overline{uu}}(\mathbf{x}^i)|^2 + |e_{\overline{uv}}(\mathbf{x}^i)|^2 + |e_{\overline{vv}}(\mathbf{x}^i)|^2), \quad (14)$$

$$\mathcal{L}_{b,\text{mean}} = \frac{1}{N_b} \sum_{j=1}^{N_b} (|U(\mathbf{x}^j) - U^*(\mathbf{x}^j)|^2 + |V(\mathbf{x}^j) - V^*(\mathbf{x}^j)|^2 + |P(\mathbf{x}^j) - P^*(\mathbf{x}^j)|^2), \quad (15)$$

$$\mathcal{L}_{b,\text{turb}} = \frac{1}{N_b} \sum_{j=1}^{N_b} (|\overline{uu}(\mathbf{x}^j) - \overline{uu}^*(\mathbf{x}^j)|^2 + |\overline{uv}(\mathbf{x}^j) - \overline{uv}^*(\mathbf{x}^j)|^2 + |\overline{vv}(\mathbf{x}^j) - \overline{vv}^*(\mathbf{x}^j)|^2). \quad (16)$$

Here $[\alpha_1, \alpha_2, \beta_1, \beta_2]$ are the weighting coefficients (with their default values set as 1) for balancing the data-driven part and the physics-informed part as well as balancing the equations governing the mean flow quantities and the turbulence quantities. $\{\mathbf{x}^i, 1 \leq i \leq N_e\}$ are the collocation points inside the flow domain and $\{\mathbf{x}^j, 1 \leq j \leq N_b\}$ are the points at the domain boundary. Finally, the training is carried out through first using Adam optimizer by feeding mini-batches of data into the RSTnet, and then continuing the training process with the L-BFGS optimizer to obtain the final converged results.

This two-stage training procedure, where a first-order optimizer is used to bring training into the vicinity of a good solution before a quasi-Newton method delivers higher final accuracy, follows common practice in the PINNs literature. The original PINNs formulation adopted L-BFGS, a quasi-Newton method, as its optimizer [27], and the two-stage combination of an initial Adam phase followed by a quasi-Newton phase has since become standard practice, including in the closely related flow reconstruction studies based on PINNs [45–47]. The main rationale behind this is not

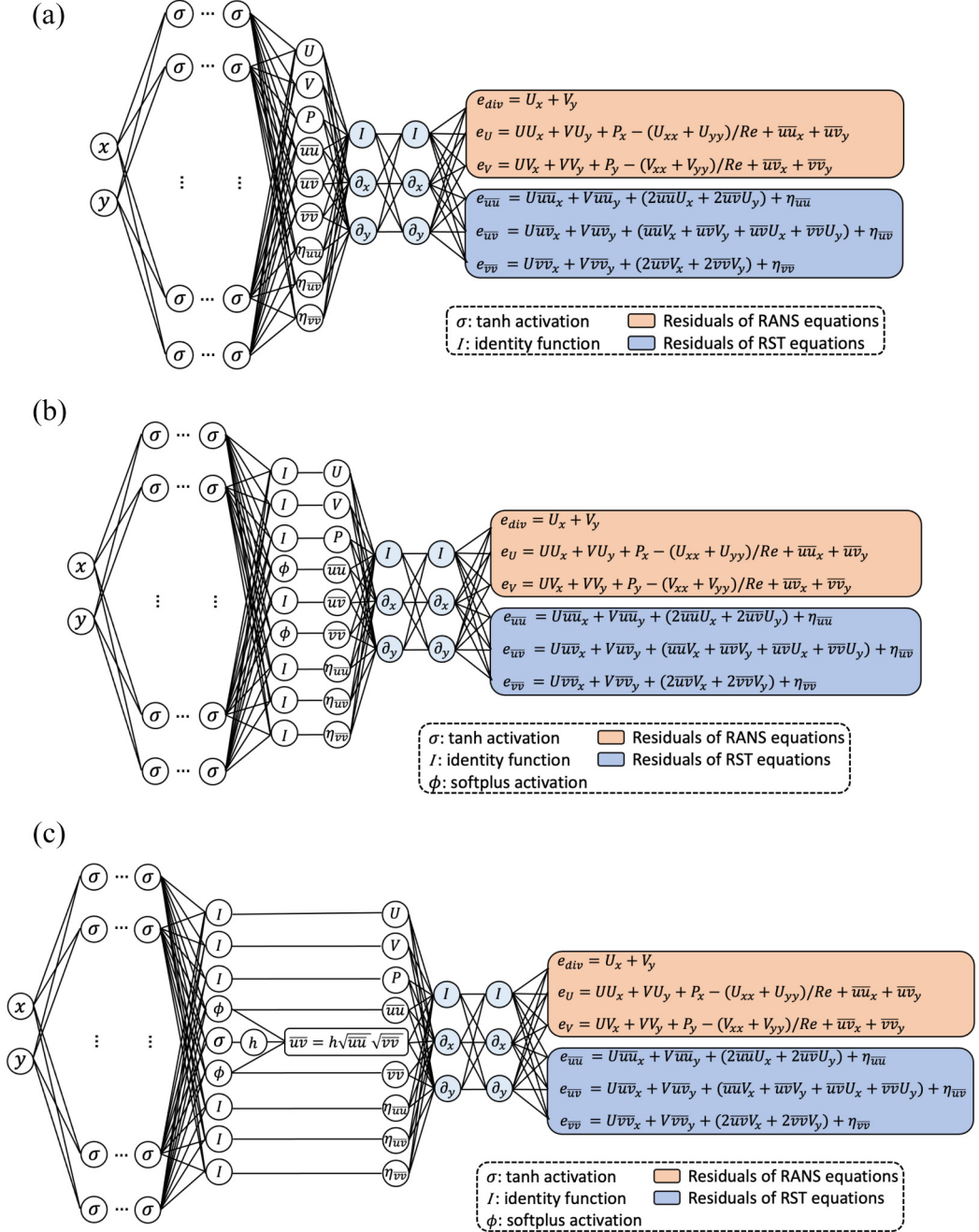


FIG. 1. Illustration of the proposed RSTnets for reconstructing RANS solutions from boundary data with partial Reynolds stress transport constraints, including (a) the baseline RSTnet, (b) the RSTnet imposing nonnegative normal Reynolds stresses, and (c) the RSTnet imposing Reynolds stress realizability.

TABLE I. Hyperparameter tuning details for the ZPG turbulent boundary layer flow case. Here the neural network size is denoted by the number of hidden layer neuron number $\times$ the number of hidden layer.

NN size	LR	ϵ_U	ϵ_V	ϵ_{uu}	ϵ_{uv}	ϵ_{vv}
16×4	10^{-3}	3.43%	9.36%	20.89%	28.43%	29.68%
16×4	10^{-4}	1.83%	7.89%	28.11%	19.94%	24.24%
16×8	10^{-3}	1.79%	10.63%	17.87%	9.72%	22.26%
16×8	10^{-4}	2.27%	11.11%	18.58%	13.65%	26.84%
32×4	10^{-3}	2.05%	10.03%	22.25%	12.51%	23.85%
32×4	10^{-4}	2.46%	8.97%	25.26%	10.93%	20.45%
$32 \times 8^*$	10^{-3^*}	0.80%	5.41%	6.30%	7.92%	8.36%
32×8	10^{-4}	1.66%	15.25%	16.77%	14.95%	17.57%
64×4	10^{-3}	14.50%	20.78%	85.28%	108.37%	100.00%
64×4	10^{-4}	3.75%	18.25%	49.19%	21.85%	28.90%
64×8	10^{-3}	1.68%	7.02%	6.93%	10.93%	13.36%
64×8	10^{-4}	2.98%	9.72%	28.41%	78.67%	17.57%

the reduction of the iteration time, but that the convergence of a quasi-Newton method depends on its initial guess, which the Adam phase is used to provide at an acceptable loss level [46]. Consistent with this, the work in Ref. [32] tunes the number of initial Adam iterations against the final accuracy rather than the training time, reports that the two-stage procedure gives more consistent results, and skips the Adam phase altogether in a transfer-learning case where the initialization is already close to the solution. A recent systematic study of quasi-Newton optimizers for PINNs across a range of challenging PDEs also employs the two-stage training procedure, and reports that the quasi-Newton methods consistently achieve lower errors than using the Adam optimizer alone [58].

Based on the baseline RSTnet presented above, two variants of the RSTnet are further developed to encode the realizability conditions of the Reynolds stress into the neural network design. They are illustrated in Figs. 1(b) and 1(c). The realizability conditions, which serve as the minimum requirements for preventing nonphysical results in traditional turbulence modeling, can be expressed as

$$\overline{u_i u_j} \geq 0, \quad \text{if } i = j, \quad (17)$$

$$\frac{|\overline{u_i u_j}|}{\sqrt{\overline{u_i u_i}} \sqrt{\overline{u_j u_j}}} \leq 1, \quad \text{if } i \neq j. \quad (18)$$

To encode the condition (17), as shown in Fig. 1(b), the output layer of the neural network is designed with the softplus activation for the neurons returning the normal stresses of the Reynolds stress tensor and the linear activation for all the other neurons, therefore imposing the positivity condition. Next, to encode the condition (18), as shown in Fig. 1(c), the output layer is further modified with the tanh activation applied for the neuron that returns the shear stress and at the same time the output is designed as the shear stress normalized by the corresponding normal stresses. In this way, the Cauchy-Schwarz inequality condition is imposed in hard form as the tanh activation has a range of $(-1, 1)$. It is worth emphasizing that both of these constraints are architectural, hard constraints, i.e., for any value of the network weights, the softplus and tanh activation functions ensure that the nonnegativity and Cauchy-Schwarz inequality conditions are satisfied exactly, rather than being only softly encouraged through an additional penalty term in the training loss. An alternative approach would be to enforce the realizability conditions via constrained optimization applied on top of the Adam or L-BFGS optimizer, e.g., a trust-region method. This is not pursued here, as architectural enforcement achieves exact constraint satisfaction throughout training without turning the optimizer into a constrained solver. It also reduces, by construction, the feasible solution space that the optimizer searches over, which is expected to ease

and accelerate convergence in practice. The constrained optimization approach, however, remains of great interest for enforcing more generic forms of constraints where a hard architectural encoding is not available. A hard constraint on the nonnegativity of the normal stresses could alternatively be imposed via a ReLU activation, and softplus is preferred here, as ReLU has a zero gradient for all negative preactivations (which can lead to inactive neurons during training) while softplus provides a smooth, everywhere-nonzero gradient and still guarantees strict positivity. In general, the use of smooth activation functions has been found to be important for PINNs trainability [46]. The three RSTnets above are hereby denoted as RSTnet-a, RSTnet-b, and RSTnet-c, respectively.

D. Evaluation metrics

After the training converges, the RSTnet can be used to reconstruct both the mean flow quantities and the turbulence quantities in the flow domain. In this paper, the prediction accuracy is evaluated using the relative L_2 error which is defined as

$$\epsilon_q = \|q - q^*\|_2 / \|q^*\|_2, \quad (19)$$

where q and q^* represent the predicted and the reference values of the flow quantity q at all the test points in the flow domain. Specifically, q can be specified as the flow velocity components (e.g., U , V), or the Reynolds stress components (e.g., $\overline{uu}$, $\overline{uv}$, $\overline{vv}$).

In addition, it is worth noting that machine learning-based surrogate modeling studies often rely on similar evaluation metrics such as the one defined above. However, there are key differences. First, surrogate models often employ similar or same training losses as the evaluation metrics and the access to the reference flow fields q^* is often required during the training process. Second, for surrogate modeling, the evaluated errors will be small as long as the test samples are not drastically different from the training samples. However, for the proposed RSTnets, the reference flow fields q^* are not required for the training. Therefore, there are no “training cases” but only “test cases” in the sense that the reference flow fields q^* for any flow cases are not leaked to the machine learning model. The reference flow fields and the evaluation metrics defined above are only used for evaluation purposes.

III. NUMERICAL RESULTS

This section gives the numerical results of the proposed RSTnets for a set of flow cases, including a ZPG turbulent boundary layer flow, an APG turbulent boundary layer flow, and a set of flow cases over periodic hills of parameterized geometries. The results for these flow cases are then analyzed together in two further comparisons, including the comparison against a fully unmodeled Reynolds stress baseline in Sec. III D and a joint discussion of the realizability violation in Sec. III E. To avoid confusion with other studies where machine learning models are trained on the training cases and then applied to the test cases, it is worth noting that the proposed RSTnets are independently trained for the five test cases in this work. In terms of computational cost, the typical training times required for the ZPG, APG, and periodic hill cases are 6400 s, 6900 s, and 2700 s, respectively, using a single NVIDIA RTX PRO 5000 GPU. Since the trained network provides the solution directly, the training time here constitutes essentially the whole computational cost, as evaluating the network to obtain the flow solution is negligible by comparison.

A. ZPG turbulent boundary layer flow

In this section, the proposed RSTnets are applied to the ZPG turbulent boundary layer flow with a Reynolds number Re_θ up to 8300 [59]. The high-fidelity LES data from the work in Ref. [59] is used to provide the mean flow quantities and second-order statistics at domain boundaries as the training data for the RSTnets and to provide the reference flow field for performance evaluations. Specifically, the LES dataset is first nondimensionalized based on the characteristic length scale, i.e., the displacement thickness of the inflow boundary layer δ_0 and the characteristic velocity scale,

i.e., the free stream velocity U_∞ . The flow domain of interest is then set as $0 \leq y \leq 130$ for the wall-normal direction and covers the streamwise domain with $1000 \leq \text{Re}_\theta \leq 7000$. Next, the velocity components $[U, V]$ and the Reynolds stress components $[\overline{uu}, \overline{uv}, \overline{vv}]$ at the domain boundaries are extracted as the training data for the RSTnets. The boundary points are clustered toward the wall, inheriting the near-wall resolution of the underlying LES dataset [59]. This gives a first off-wall point at $y^+ \approx 0.1$ and approximately 10 points below $y^+ = 10$. The benefit of using wall-clustered point distribution compared to uniform distribution is quantified in Appendix C.

The number of data points used at the domain boundaries is 200 for the left and right boundaries and 499 for the top and bottom boundaries (thus in total $N_b = 1398$), and the number of collocation points in the flow domain for minimizing the physics-informed residuals is 200×499 (i.e., $N_e = 99800$). The network structure is set as $2 \times 32 \times 32 \times 32 \times 32 \times 32 \times 32 \times 32 \times 9$ and the learning rate is set as 10^{-3} . The corresponding hyperparameter tuning details are given in Table I, where the results for a set of hidden layer neuron numbers, numbers of hidden layers, and learning rates are included. As shown, the best performance is achieved with 8 hidden layers, 32 hidden neuron numbers, and a learning rate of 10^{-3} . The training convergence regarding the number of boundary points N_b and the number of the collocation points N_e is then tested through a set of cases where N_b and N_e are reduced successively. The results are given in Figs. 2(a) and 2(b) which shows the relative L_2 errors of the mean flow quantities and the turbulence quantities with respect to N_e and N_b . As shown, the L_2 error decreases initially and stabilizes toward the considered maximum values of N_e and N_b , demonstrating the training convergence with the current choice of N_b and N_e .

Finally, the three RSTnet structures, i.e., the baseline RSTnet-a, the RSTnet-b which incorporates the mechanism of enforcing nonnegativity of normal Reynolds stresses, and the RSTnet-c which incorporates the mechanism of enforcing realizability of Reynolds stresses, are tested and compared. The results are given in Table II. As shown, the RSTnet-b clearly outperforms the RSTnet-a, showing the importance of the proposed mechanism for imposing nonnegativity for the normal stresses. However, the RSTnet-c, by further imposing the Cauchy-Schwarz inequality condition for the shear stress components compared to the RSTnet-b, performs much worse than the RSTnet-b. The reasons for this deteriorated performance, and how they relate to previous findings in the Reynolds stress transport modeling literature, are discussed jointly with other test cases in Sec. III E. Because of the superior performance of the RSTnet-b compared to the other two structures, it is hereby used as the default network structure in this paper.

The flow fields predicted by the RSTnet-b are given next. The training curves, in term of the total loss $\mathcal{L}$ and the individual loss terms, i.e., $\mathcal{L}_{e,\text{mean}}$, $\mathcal{L}_{e,\text{turb}}$, $\mathcal{L}_{b,\text{mean}}$, and $\mathcal{L}_{b,\text{turb}}$, are examined first. The results are given in Fig. 3. As shown, the training loss is quite noisy during the Adam iterations and continue to drop to about 10^{-8} after the L-BFGS iterations. Then the flow fields predicted by the RSTnet are given in Fig. 4 including both the mean flow quantities and the turbulence quantities. The corresponding LES results are also shown for comparison. As shown, based on only information at boundaries, the RSTnet is able to reconstruct the flow field in the domain very accurately for the mean flow quantities and a decent accuracy is also achieved for the turbulence quantities. In particular, the relative L_2 errors are just 0.80% and 5.41% for U and V , and are just 6.30%, 7.92%, and 8.36% for $\overline{uu}$, $\overline{uv}$, and $\overline{vv}$.

To further quantify the prediction accuracy, the profiles of the flow velocity, Reynolds stresses, and turbulence kinetic energy at three streamwise locations (i.e., $\text{Re}_\theta = 2000, 4000, 6000$), as well as the skin-friction coefficient, are extracted from the predicted flow fields. The results are given in Figs. 5 and 6, respectively. As shown, the RSTnet results match well with the LES results for both the direct outputs of the RSTnet (i.e., U , V , $\overline{uu}$, $\overline{uv}$, $\overline{vv}$) and the derived quantity of interest (i.e., C_f).

The effects of the weighting coefficients $[\alpha_1, \alpha_2, \beta_1, \beta_2]$ are then examined and the results are given in Table III. The balancing weights between the physics-informed and data-driven parts, as well as between the governing equations for the mean flow and the Reynolds stress, are both investigated. The results show that different from previous studies based on NS equations [32],

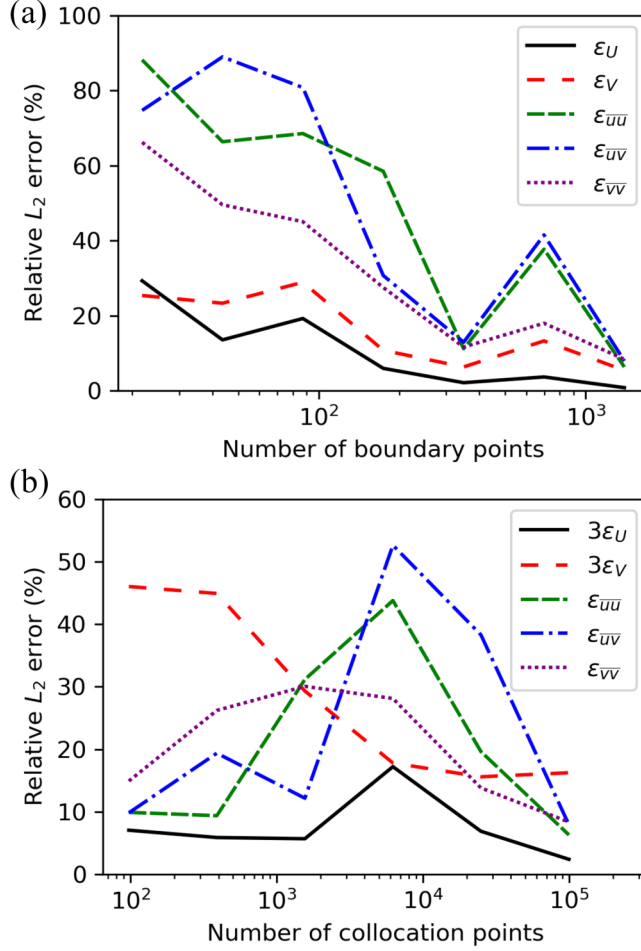


FIG. 2. Training convergence regarding (a) the number of boundary points and (b) the number of the collocation points, at the ZPG turbulent boundary layer flow case.

unbalanced weighting between individual loss terms does not lead to performance improvements for the RSTnet.

B. APG turbulent boundary layer flow

This section focuses on the application of the RSTnets for the turbulent boundary layer flow with adverse pressure gradient, where the Reynolds number range is $910 < \text{Re}_\theta < 3360$ and the Clauser pressure-gradient parameter is $\beta \simeq 1$ [60]. The high-fidelity LES data from Ref. [60] is

TABLE II. Performance comparison of the three RSTnets at the ZPG turbulent boundary layer flow case.

Method	ϵ_U	ϵ_V	$\epsilon_{\overline{uu}}$	$\epsilon_{\overline{uv}}$	$\epsilon_{\overline{vv}}$
RSTnet-a	3.78%	11.75%	21.90%	10.57%	32.80%
RSTnet-b*	0.80%	5.41%	6.30%	7.92%	8.36%
RSTnet-c	3.47%	8.10%	28.07%	29.54%	22.64%

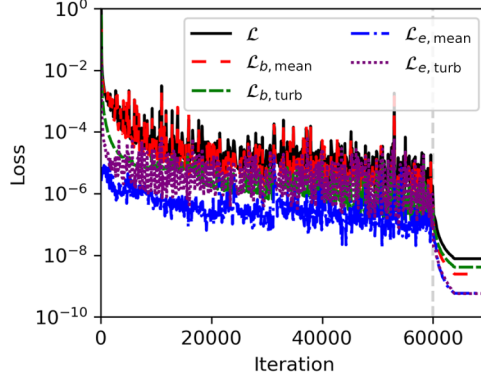


FIG. 3. Training losses at the ZPG turbulent boundary layer flow case, including the total loss $\mathcal{L}$, the physical loss $\mathcal{L}_{e,\text{mean}}$ and $\mathcal{L}_{e,\text{turb}}$, and the boundary loss $\mathcal{L}_{b,\text{mean}}$ and $\mathcal{L}_{b,\text{turb}}$. The vertical dashed line separates the Adam iterations and the L-BFGS iterations.

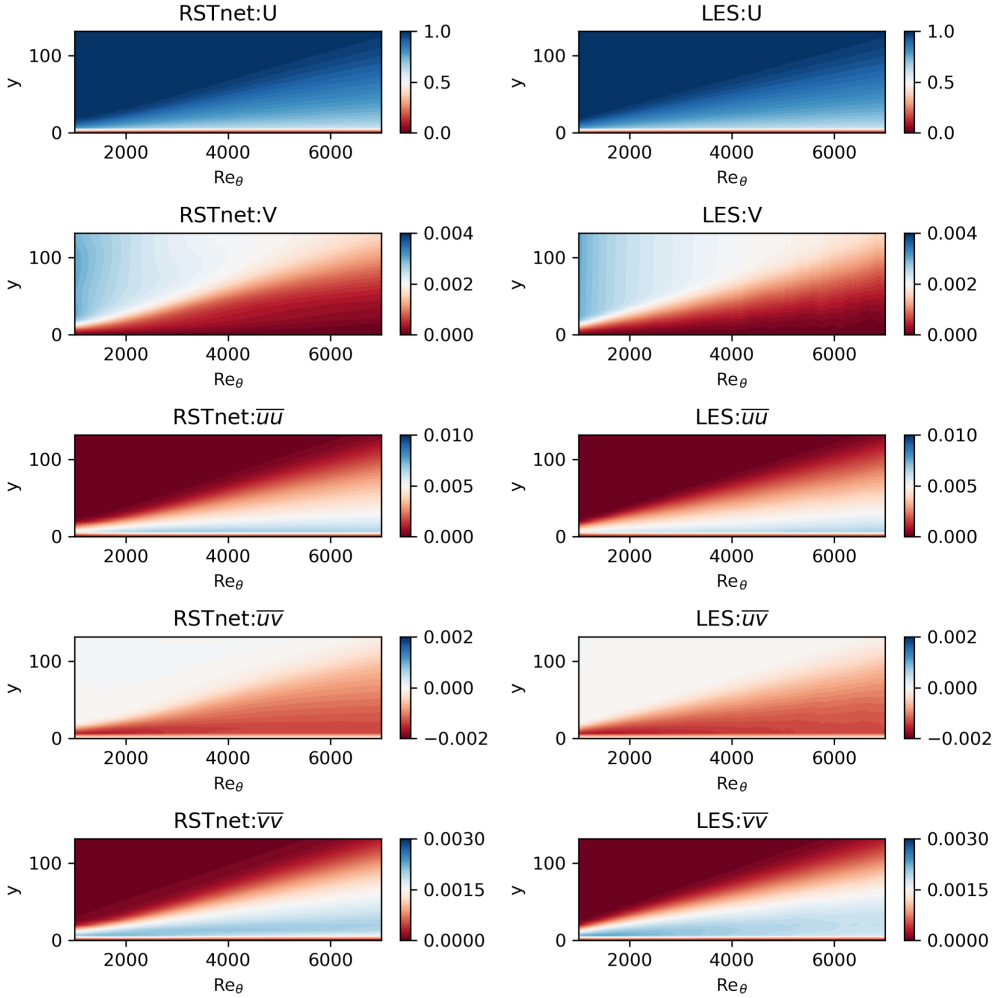


FIG. 4. Flow field prediction at the ZPG turbulent boundary layer flow case, including both the mean flow quantities and the turbulence quantities. The corresponding LES results are also shown for comparison.

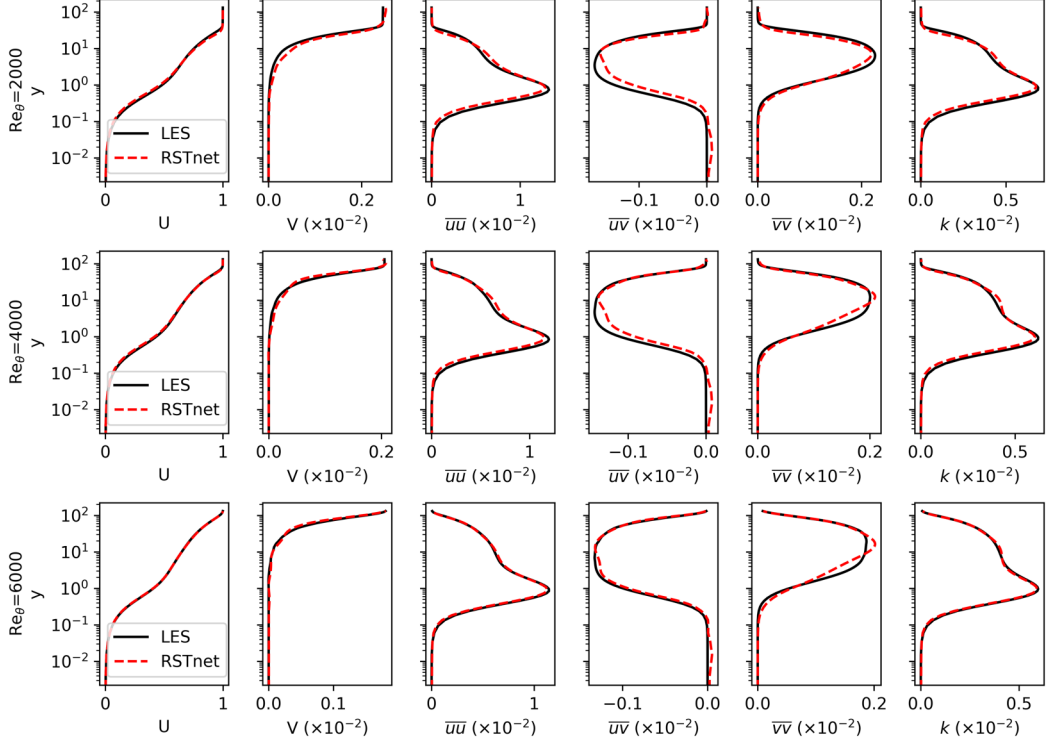


FIG. 5. Extracted profiles of the flow velocity, Reynolds stresses, and turbulence kinetic energy at three streamwise locations (i.e., $Re_\theta = 2000, 4000, 6000$), at the ZPG turbulent boundary layer flow case.

used to provide the first- and second-order flow quantities at domain boundaries as the training data for the RSTnets and to provide the reference flow field for performance evaluation. The LES dataset is first nondimensionalized based on the displacement thickness of the inflow boundary layer δ_0 and the free stream velocity U_∞ . The flow domain of interest is then set as $0 \leq y \leq 140$ for the wall-normal direction and covers the streamwise domain with $1000 \leq Re_\theta \leq 3000$. Next, the velocity components $[U, V]$, the pressure P , and the Reynolds stress components $[\overline{uu}, \overline{uv}, \overline{vv}]$ at the domain boundaries are extracted as the training data for the RSTnets. As in the ZPG case, the boundary points are clustered toward the wall, inheriting the near-wall resolution of the underlying LES dataset [60]. This gives the same wall-unit characteristics as the ZPG case, i.e., a first off-wall point at $y^+ \approx 0.1$ and approximately 10 points below $y^+ = 10$. The benefit of this wall-clustered point distribution is quantified in Appendix C.

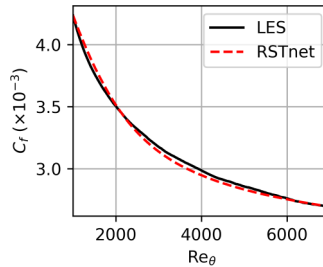


FIG. 6. Skin friction distribution for the ZPG turbulent boundary layer flow case.

TABLE III. Influence of different weighting coefficients for different loss terms to form the total training loss, at the ZPG turbulent boundary layer flow case.

α_1	α_2	β_1	β_2	ϵ_U	ϵ_V	$\epsilon_{\overline{uu}}$	$\epsilon_{\overline{uv}}$	$\epsilon_{\overline{vv}}$
1	10^{-2}	1	10^{-2}	2.58%	16.81%	99.94%	31.01%	99.52%
1	10^{-1}	1	10^{-1}	4.79%	24.02%	33.08%	69.72%	20.79%
1*	10^{0*}	1*	10^{0*}	0.80%	5.41%	6.30%	7.92%	8.36%
1	10^1	1	10^1	4.00%	16.39%	44.68%	55.07%	39.03%
1	10^2	1	10^2	2.94%	6.63%	30.11%	17.46%	27.51%
1	1	10^{-1}	10^{-1}	7.24%	21.21%	41.91%	72.88%	36.30%
1*	1*	10^{0*}	10^{0*}	0.80%	5.41%	6.30%	7.92%	8.36%
1	1	10^1	10^1	1.71%	11.53%	8.78%	11.27%	13.17%
1	1	10^2	10^2	2.49%	19.45%	8.72%	14.15%	13.57%

The number of data points used at the domain boundaries is 301 for the left and right boundaries and 337 for the top and bottom boundaries, and the number of collocation points in the flow domain for minimizing the physics-informed residuals is 301×337 . The network structure is set as $2 \times 32 \times 32 \times 32 \times 32 \times 32 \times 32 \times 32 \times 9$ and the learning rate is set as 10^{-3} . The corresponding hyperparameter tuning details are reported in Table IV. The training convergence regarding the number of boundary points N_b and the number of the collocation points N_e is then examined and the results are given in Figs. 7(a) and 7(b). As shown, the relative L_2 error decreases with the increase of N_e and N_b initially and converges toward the considered maximum values of N_e and N_b . Finally the proposed three RSTnets are tested and compared. The results are given in Table V. A clear advantage of the RSTnet-b is observed over the other two structures. Particularly, the RSTnet-a fails completely in capturing the turbulence quantities in this case, demonstrating the importance of enforcing nonnegativity for normal Reynolds stresses.

The flow field predictions are given next. The training curves are first examined in Fig. 8. Similar as the ZPG turbulent boundary layer case, the training loss is quite noisy during the Adam iterations and continue to drop to about 10^{-8} after the L-BFGS iterations. The flow fields reconstructed by the RSTnet are then given in Fig. 9. As shown, the RSTnet is able to predict the flow fields accurately for both the mean flow quantities and the turbulence quantities. In particular, the relative L_2 errors are just 0.53% and 0.98% for U and V , and are just 10.14%, 14.58%, and 15.04% for $\overline{uu}$, $\overline{uv}$, and

TABLE IV. Hyperparameter tuning details for the APG turbulent boundary layer flow case. Here the neural network size is denoted by the number of hidden layer neuron number $\times$ the number of hidden layer.

NN size	LR	ϵ_U	ϵ_V	$\epsilon_{\overline{uu}}$	$\epsilon_{\overline{uv}}$	$\epsilon_{\overline{vv}}$
16×4	10^{-3}	2.76%	5.25%	28.17%	19.02%	36.07%
16×4	10^{-4}	2.08%	2.44%	26.38%	24.53%	25.09%
16×8	10^{-3}	0.67%	1.67%	32.31%	15.15%	22.87%
16×8	10^{-4}	0.74%	2.26%	13.99%	9.88%	30.29%
32×4	10^{-3}	1.04%	1.99%	61.63%	12.86%	36.67%
32×4	10^{-4}	1.55%	2.43%	33.60%	23.84%	24.71%
$32 \times 8^*$	10^{-3*}	0.53%	0.98%	10.14%	14.58%	15.04%
32×8	10^{-4}	0.45%	3.03%	46.22%	24.68%	17.84%
64×4	10^{-3}	5.22%	3.25%	90.62%	49.97%	74.76%
64×4	10^{-4}	1.17%	2.56%	53.88%	37.05%	31.66%
64×8	10^{-3}	1.09%	1.91%	24.07%	38.26%	25.02%
64×8	10^{-4}	1.40%	1.11%	38.62%	38.62%	22.99%

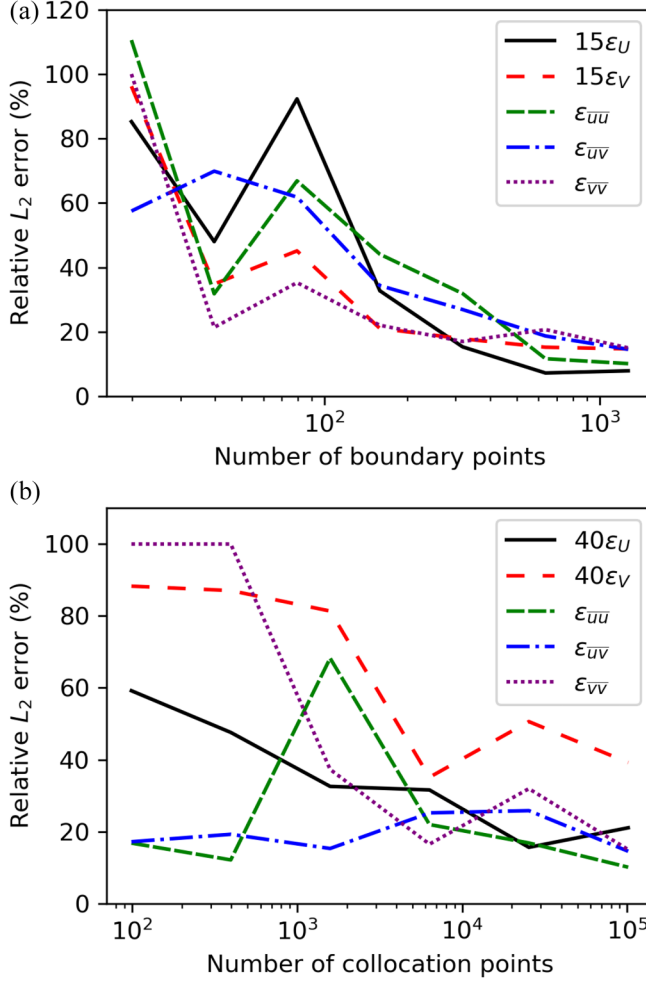


FIG. 7. Training convergence regarding (a) the number of boundary points and (b) the number of the collocation points, at the APG turbulent boundary layer flow case.

$\bar{v}\bar{v}$. To further quantify the prediction accuracy, the flow profiles at three streamwise locations (i.e., $Re_\theta = 1200, 2000, 2800$) and the skin-friction coefficient are extracted and shown in Figs. 10 and 11, respectively. Overall, the RSTnet results agree well with the LES results. Compared with the ZPG flow case, the discrepancy observed for the Reynolds stress components in the APG case is larger, which is also reflected in the reported relative L_2 errors. However, the proposed RSTnet is still able to accurately capture the magnitude and location of the peak for the $\bar{u}\bar{v}$ component as well as the skin friction distribution, clearly outperforming the earlier PINN-RANS study [47].

TABLE V. Performance comparison of the three RSTnets at the APG turbulent boundary layer flow case.

Method	ϵ_U	ϵ_V	$\epsilon_{\bar{u}\bar{u}}$	$\epsilon_{\bar{u}\bar{v}}$	$\epsilon_{\bar{v}\bar{v}}$
RSTnet-a	4.34%	2.67%	224.89%	38.38%	226.78%
RSTnet-b*	0.53%	0.98%	10.14%	14.58%	15.04%
RSTnet-c	1.92%	2.25%	34.75%	26.39%	22.14%

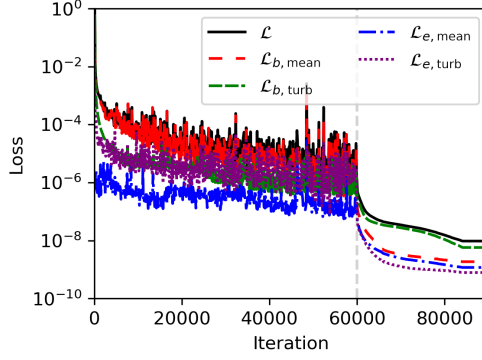


FIG. 8. Training losses at the APG turbulent boundary layer flow case, including the total loss $\mathcal{L}$, the physical loss $\mathcal{L}_{e,\text{mean}}$ and $\mathcal{L}_{e,\text{turb}}$, and the boundary loss $\mathcal{L}_{b,\text{mean}}$ and $\mathcal{L}_{b,\text{turb}}$. The vertical dashed line separates the Adam iterations and the L-BFGS iterations.

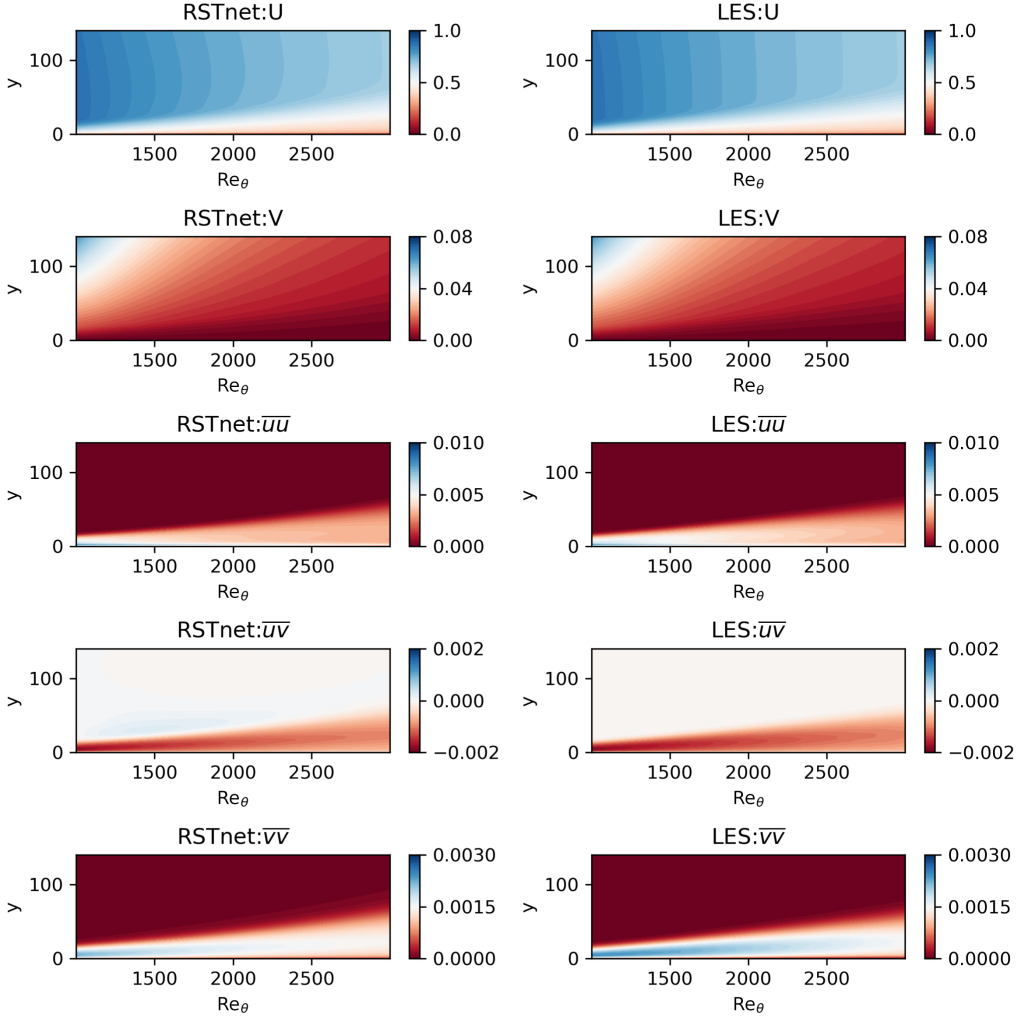


FIG. 9. Flow field prediction at the APG turbulent boundary layer flow case, including both the mean flow quantities and the turbulence quantities. The corresponding LES results are also shown for comparison.

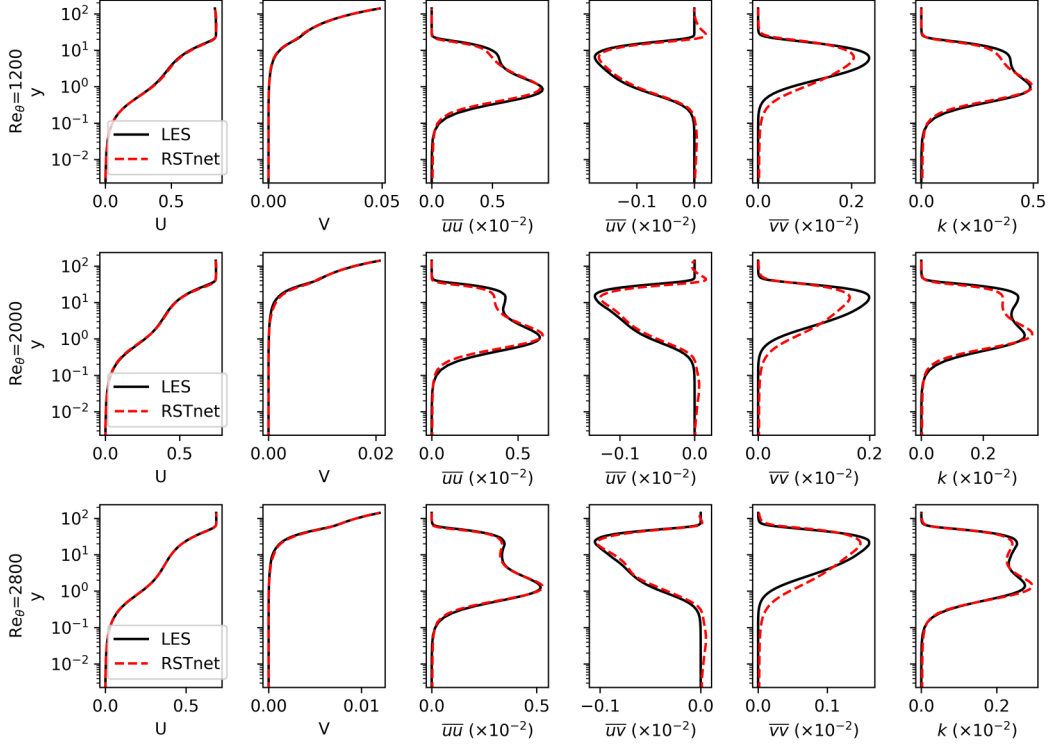


FIG. 10. Extracted profiles of the flow velocity, Reynolds stresses, and turbulence kinetic energy at three streamwise locations (i.e., $Re_\theta = 1200, 2000, 2800$), at the APG turbulent boundary layer flow case.

Finally, the effects of the weighting coefficients $[\alpha_1, \alpha_2, \beta_1, \beta_2]$ are examined. The results are given in Table VI which shows that unbalanced weight for the individual loss terms leads to worse performances for all the test cases.

C. Flows over periodic hills of parameterized geometries

This section is devoted to applying the RSTnets for the flows over periodic hills, where the hill geometries are specified according to the work in Ref. [61] with the slope of the hill parameterized by the elongating factor α . The Reynolds number is 5600 based on the hill height H and the bulk velocity U_b . The high-fidelity DNS database from the work in Ref. [61] is used to provide the flow quantities at the domain boundaries and to provide the ground truth for performance evaluation. The

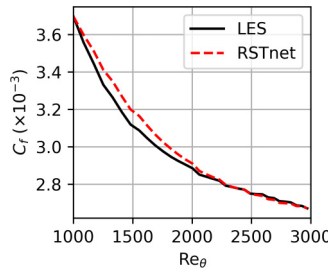


FIG. 11. Skin friction distribution for the APG turbulent boundary layer flow case.

TABLE VI. Influence of different weighting coefficients for different loss terms to form the total training loss, at the APG turbulent boundary layer flow case.

α_1	α_2	β_1	β_2	ϵ_U	ϵ_V	$\epsilon_{\overline{uu}}$	$\epsilon_{\overline{uv}}$	$\epsilon_{\overline{vv}}$
1	10^{-2}	1	10^{-2}	2.17%	2.54%	99.93%	39.74%	99.72%
1	10^{-1}	1	10^{-1}	5.41%	3.14%	69.75%	52.08%	100.00%
1*	10^{0*}	1*	10^{0*}	0.53%	0.98%	10.14%	14.58%	15.04%
1	10^1	1	10^1	1.07%	1.54%	40.48%	35.54%	30.48%
1	10^2	1	10^2	0.78%	4.01%	36.68%	26.20%	36.38%
1	1	10^{-1}	10^{-1}	1.99%	1.61%	35.38%	44.93%	87.95%
1*	1*	10^{0*}	10^{0*}	0.53%	0.98%	10.14%	14.58%	15.04%
1	1	10^1	10^1	2.13%	2.79%	43.43%	29.15%	100.00%
1	1	10^2	10^2	4.32%	3.12%	65.68%	51.26%	48.35%

dataset is first nondimensionalized based on the hill height H and the bulk velocity U_b . Then the flow domain of interest is set as $y/H \in [0, 3]$ in the wall-normal direction and $x/H \in [1, 5]$ in the streamwise direction over the parameterized hill geometries. This streamwise extent, a subregion of the full domain considered in Ref. [61], is chosen to match the domain used in Ref. [47], to the author's knowledge the only other study that reconstructs turbulent flow fields via PINNs from boundary data alone. This enables a direct and fair comparison between the two approaches. Next, the flow quantities including the velocity components $[U, V]$ and the Reynolds stress components $[\overline{uu}, \overline{uv}, \overline{vv}]$ at the domain boundaries are extracted as the training data for the RSTnets.

For the training, the number of points at the domain boundaries is set to about 1700 and the number of collocation points is set to about 130 000. The neural network structure is set as $2 \times 64 \times 64 \times 64 \times 64 \times 64 \times 64 \times 64 \times 9$ and the learning rate is set as 10^{-3} . The corresponding hyperparameter tuning details are reported in Table VII. As shown, the network structure with the best performance has a neuron number of 64 and a layer number of 8. The further increase of the neural network size does not improve the performance further. With the

TABLE VII. Hyperparameter tuning details for the periodic hill flow case. Here the neural network size is denoted by the number of hidden layer neuron number $\times$ the number of hidden layer.

NN size	LR	ϵ_U	ϵ_V	$\epsilon_{\overline{uu}}$	$\epsilon_{\overline{uv}}$	$\epsilon_{\overline{vv}}$
16×4	10^{-3}	3.96%	29.19%	32.88%	13.21%	34.09%
16×4	10^{-4}	4.07%	31.16%	35.30%	23.54%	34.27%
16×8	10^{-3}	4.99%	37.91%	68.75%	45.67%	75.99%
16×8	10^{-4}	5.07%	35.36%	38.09%	32.07%	42.78%
32×4	10^{-3}	3.87%	27.78%	38.30%	19.63%	37.19%
32×4	10^{-4}	3.37%	25.25%	32.72%	16.30%	34.60%
32×8	10^{-3}	3.63%	26.66%	29.95%	18.49%	29.71%
32×8	10^{-4}	4.83%	32.91%	25.64%	20.44%	28.07%
64×4	10^{-3}	2.09%	11.37%	26.86%	19.52%	23.58%
64×4	10^{-4}	3.55%	13.62%	31.90%	31.96%	31.33%
$64 \times 8^*$	10^{-3*}	1.73%	8.71%	15.56%	16.52%	12.52%
64×8	10^{-4}	4.47%	31.04%	37.38%	22.73%	36.23%
128×4	10^{-3}	2.19%	10.70%	34.21%	26.86%	41.97%
128×4	10^{-4}	3.51%	12.50%	31.79%	22.55%	34.03%
128×8	10^{-3}	1.94%	9.59%	28.56%	15.08%	21.15%
128×8	10^{-4}	3.22%	14.02%	28.87%	31.97%	24.82%

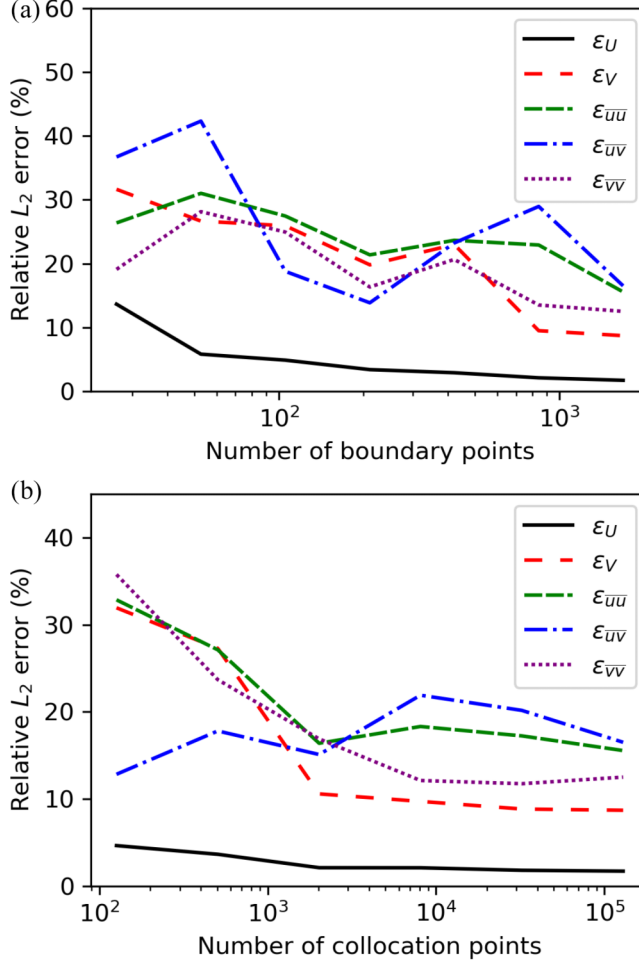


FIG. 12. Training convergence regarding (a) the number of boundary points and (b) the number of the collocation points, at the periodic hill flow case.

optimized hyperparameter settings, the training convergence with respect to the number of the boundary points N_b and the number of the collocation points N_e is then investigated. The results are given in Fig. 12, which shows the influence of N_b and N_e on the prediction accuracy. As shown in Fig. 12(a), the relative L_2 error decreases with the increase of the boundary points initially and then stabilizes toward the considered maximum value for N_b . A similar trend is observed for the training convergence regarding N_e , as shown in Fig. 12(b). Next, a comparison study is carried out for the three RSTnet structures. The results are given in Table VIII, which shows the clear benefit

TABLE VIII. Performance comparison of the three RSTnets at the periodic hill flow case.

Method	ϵ_U	ϵ_V	$\epsilon_{\bar{u}\bar{u}}$	$\epsilon_{\bar{u}\bar{v}}$	$\epsilon_{\bar{v}\bar{v}}$
RSTnet-a	4.11%	30.48%	22.99%	18.92%	12.25%
RSTnet-b*	1.73%	8.71%	15.56%	16.52%	12.52%
RSTnet-c	1.56%	12.67%	15.74%	12.01%	12.45%

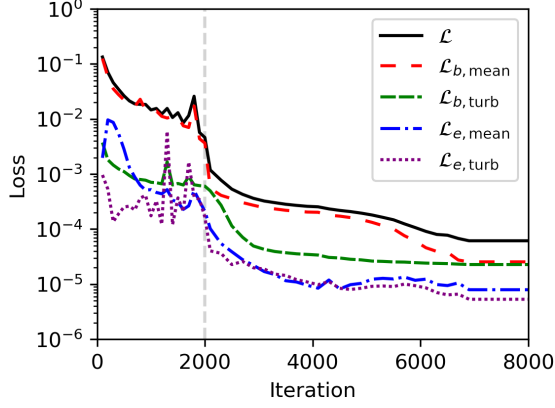


FIG. 13. Training losses at the periodic hill flow case, including the total loss $\mathcal{L}$, the physical loss $\mathcal{L}_{b,\text{mean}}$ and $\mathcal{L}_{b,\text{turb}}$, and the boundary loss $\mathcal{L}_{e,\text{mean}}$ and $\mathcal{L}_{e,\text{turb}}$. The vertical dashed line separates the Adam iterations and the L-BFGS iterations.

of enforcing the nonnegativity of normal stresses in the proposed RSTnet-b. A performance gain is also observed for the prediction of the Reynolds shear stress using the proposed RSTnet-c, where the realizability condition for the shear stress is enforced in the network design, in contrast to the ZPG and APG turbulent boundary layer cases where the same additional constraint leads to worse performance. This contrast is discussed further in Sec. III E.

The flow fields predicted by the RSTnet-b are given next. The training loss, including the total loss and the individual loss terms, are examined first. As shown in Fig. 13, the decrease of the training loss is quite noisy with Adam iterations and the training loss eventually converges after the L-BFGS iterations. The flow fields are then given in Fig. 14 and 15 for the mean flow quantities and the turbulence flow quantities, respectively. The profiles of the streamwise velocity and the shear stress component of the Reynolds stress tensor are also extracted from the predicted flow fields and the results are shown in Fig. 16. As can be seen, based on only information at domain boundaries, the flow field information in the domain is predicted accurately by the RSTnet. In particular, the

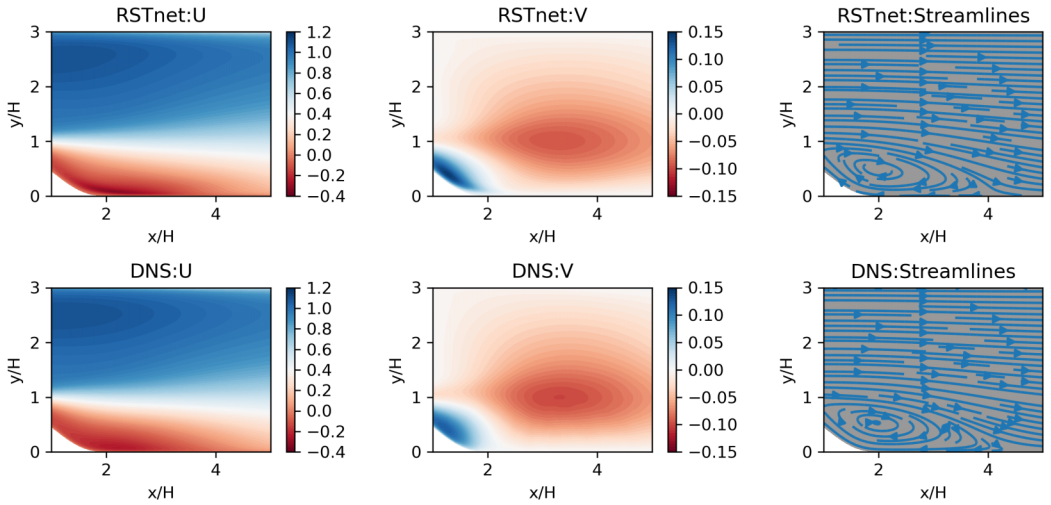


FIG. 14. Flow field of the mean flow quantities predicted by the RSTnet, at the periodic hill flow case with an elongating factor of $\alpha = 1.0$. The corresponding DNS results are also shown for comparison.

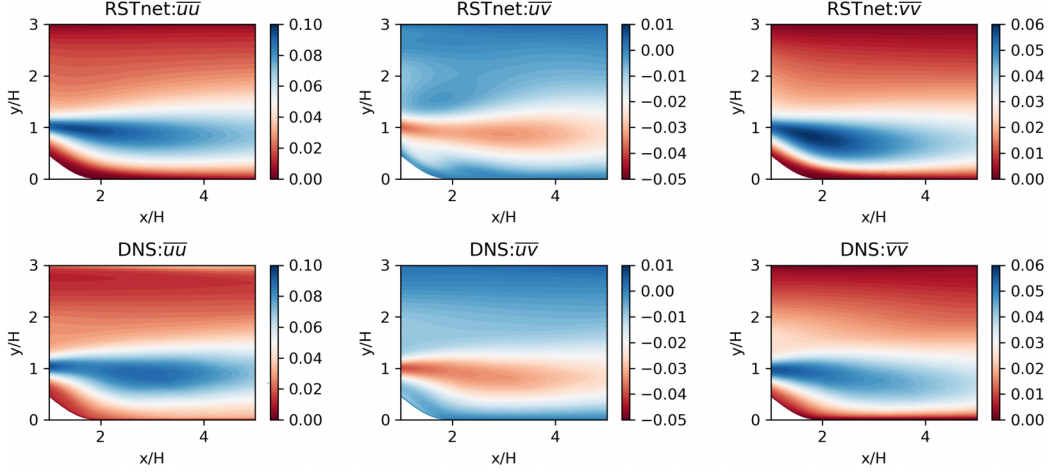


FIG. 15. Flow field of the turbulence quantities predicted by the RSTnet, at the periodic hill flow case with an elongating factor of $\alpha = 1.0$. The corresponding DNS results are also shown for comparison.

recirculation bubble is successfully captured, and all the Reynolds stress components, including the shear stress component, are predicted quite accurately. The relative L_2 errors for the velocity components are 1.73% and 8.71% for U and V , respectively, while the errors for the Reynolds stress components are 15.56%, 16.52%, and 12.52% for $\overline{uu}$, $\overline{uv}$, and $\overline{vv}$, respectively. Finally, the influence of the weighting coefficients balancing individual loss terms is investigated. The results are given in Table IX, which shows that similar as the ZPG and APG turbulent boundary layer flow cases, unbalanced weight leads to worse performances for all the test cases.

With the neural network structure and the training settings obtained above for the periodic hill case with $\alpha = 1$, the RSTnet-b is then applied to two new cases with different hill slopes, i.e., $\alpha = 1.25$ and $\alpha = 1.5$. The results are given in Figs. 17–19 for the case with $\alpha = 1.25$ and are given in Figs. 20–22 for the case with $\alpha = 1.5$. As shown, the RSTnet achieves great accuracy for all the flow quantities and the flow separation is characterized accurately for both cases. In particular, for the $\alpha = 1.25$ case, the relative L_2 errors are just 1.73% and 8.69% for U and V , and are 19.81%, 14.15%, and 14.03% for $\overline{uu}$, $\overline{uv}$, and $\overline{vv}$, respectively. For the $\alpha = 1.5$ case, the relative L_2 errors are just 2.01% and 9.52% for U and V , and are 23.84%, 14.35%, and 23.14% for $\overline{uu}$, $\overline{uv}$, and $\overline{vv}$, respectively. Finally, the skin friction distributions are extracted from the predicted flow fields. The results are shown in Fig. 23. Overall, for all three cases, the RSTnet predictions match with the

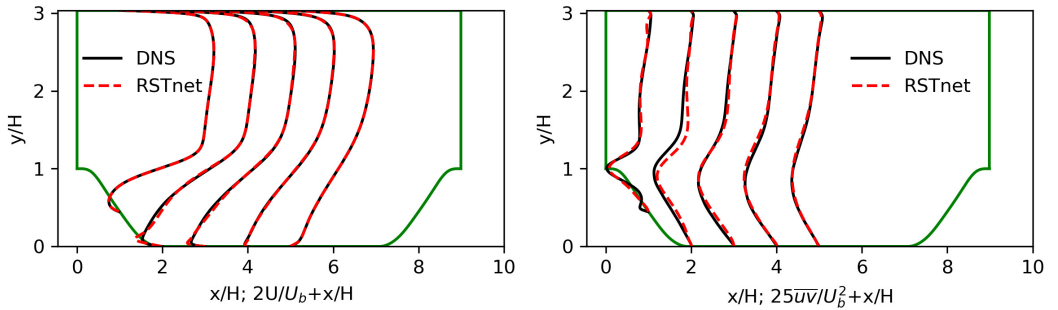


FIG. 16. Profiles of the streamwise velocity and the shear stress component of the Reynolds stress tensor predicted by the RSTnet, at the periodic hill flow case with an elongating factor of $\alpha = 1.0$. The corresponding DNS results are also shown for comparison.

TABLE IX. Influence of different weighting coefficients for different loss terms to form the total training loss, at the periodic hill flow case.

α_1	α_2	β_1	β_2	ϵ_U	ϵ_V	$\epsilon_{\bar{u}\bar{u}}$	$\epsilon_{\bar{u}\bar{v}}$	$\epsilon_{\bar{v}\bar{v}}$
1	10^{-2}	1	10^{-2}	2.87%	21.40%	71.97%	50.40%	58.52%
1	10^{-1}	1	10^{-1}	2.45%	13.19%	32.80%	41.00%	34.66%
1*	10^{0*}	1*	10^{0*}	1.73%	8.71%	15.56%	16.52%	12.52%
1	10^1	1	10^1	3.33%	24.94%	24.92%	24.34%	25.92%
1	10^2	1	10^2	4.75%	31.72%	40.82%	33.34%	38.06%
1	1	10^{-1}	10^{-1}	3.96%	29.62%	29.20%	15.48%	29.91%
1*	1*	10^{0*}	10^{0*}	1.73%	8.71%	15.56%	16.52%	12.52%
1	1	10^1	10^1	2.15%	16.10%	22.79%	14.19%	18.18%
1	1	10^2	10^2	3.45%	22.37%	13.73%	27.21%	12.49%

LES results quite well in the region after the hill, while clear discrepancies are observed over the hill slopes where the calculations are more sensitive to the domain boundary. It is worth noting that the discrepancies of C_f predictions over the hill slopes are also often quite large among traditional RANS models and LES, such as in Ref. [62]. Finally, it is worth noting that the flow reattachment locations, as indicated by the $C_f = 0$ line, are also predicted quite well by the RSTnet for all three test cases.

D. Comparison against a fully unmodeled Reynolds stress baseline

To isolate the individual contributions of the proposed RST formulation and the realizability constraint, two baseline variants of the RSTnet are further trained here, where the Reynolds stress tensor τ is treated as a completely free neural network output. The RST equation [i.e., Eq. (10)] is not used at all for the two variants, and only the continuity and momentum equations [Eqs. (6) and (7)] are enforced. The first variant, denoted as free- τ (unconstrained), additionally applies no realizability constraint on the output layer, following the same underlying approach as Ref. [47]. The second variant, denoted as free- τ (constrained), applies the same design as RSTnet-b to

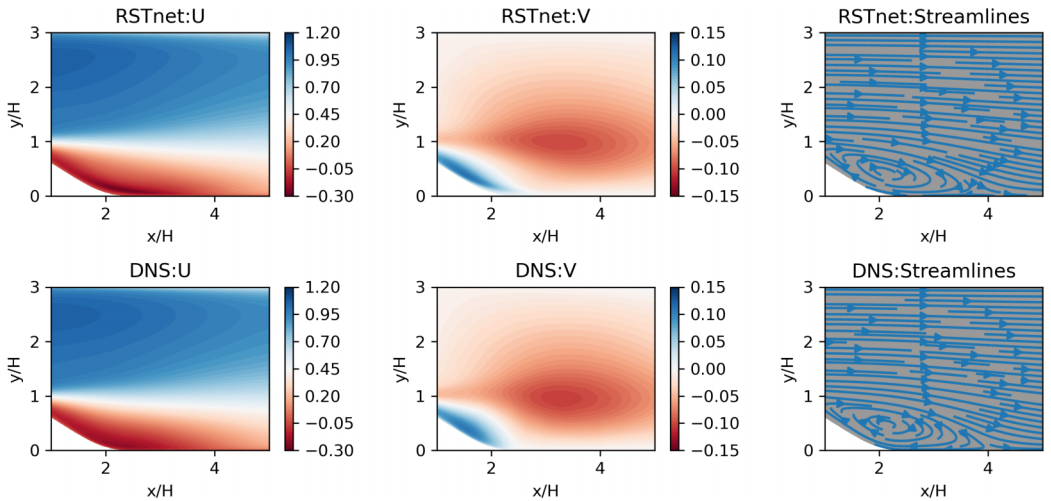


FIG. 17. Flow field of the mean flow quantities predicted by the RSTnet, at the periodic hill flow case with an elongating factor of $\alpha = 1.25$. The corresponding DNS results are also shown for comparison.

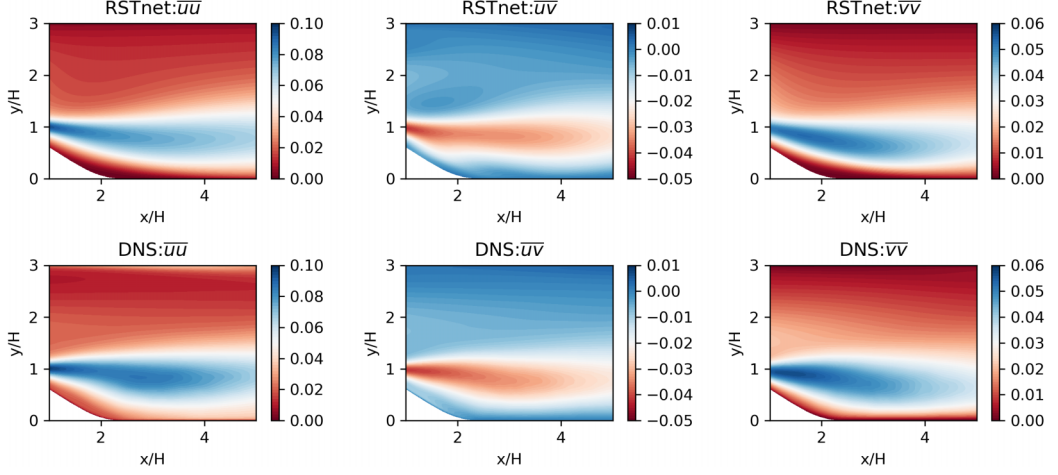


FIG. 18. Flow field of the turbulence quantities predicted by the RSTnet, at the periodic hill flow case with an elongating factor of $\alpha = 1.25$. The corresponding DNS results are also shown for comparison.

enforce the nonnegativity of the normal stress components. Both variants are trained independently for each of the ZPG, APG, and periodic hill ($\alpha = 1.0$) test cases, using the same boundary data, collocation points, network architecture, and training procedure as the corresponding RSTnet-b case in each flow. The resulting comparison therefore isolates the effect of the RST formulation from any difference in hyperparameters or training data. Along with the RSTnet-a and RSTnet-b results already reported in Secs. III A–III C, these two additional variants allow the analysis of the individual contributions from the proposed RST formulation and the realizability constraint. The results are given in Table X.

As shown in Table X, the proposed RSTnet-b, which combines the RST equation with the realizability constraint, achieves the best performance across all flow cases, offering the clearest improvement for the Reynolds stress components. Neither the RST equation alone nor the realizability constraint alone recovers the full accuracy of RSTnet-b, even though both individually improve substantially over the fully unconstrained free- τ baseline in most cases.

The relative contribution of the two mechanisms is, however, strongly flow-dependent. For the APG case the realizability constraint is clearly the dominant one, reducing the relative L_2 errors for $\overline{uu}$ and $\overline{vv}$ from values exceeding 100% to 19.37% and 32.99%, respectively, while the RST equation alone leaves them at 224.89% and 226.78%. Here the values exceeding 100%

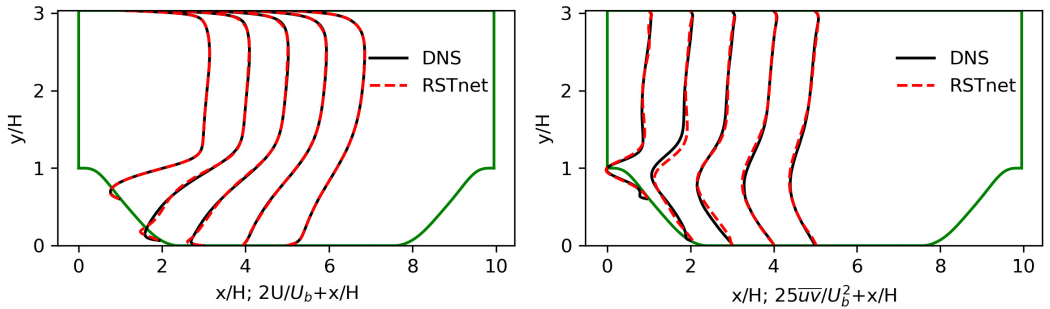


FIG. 19. Profiles of the streamwise velocity and the shear stress component of the Reynolds stress tensor predicted by the RSTnet, at the periodic hill flow case with an elongating factor of $\alpha = 1.25$. The corresponding DNS results are also shown for comparison.

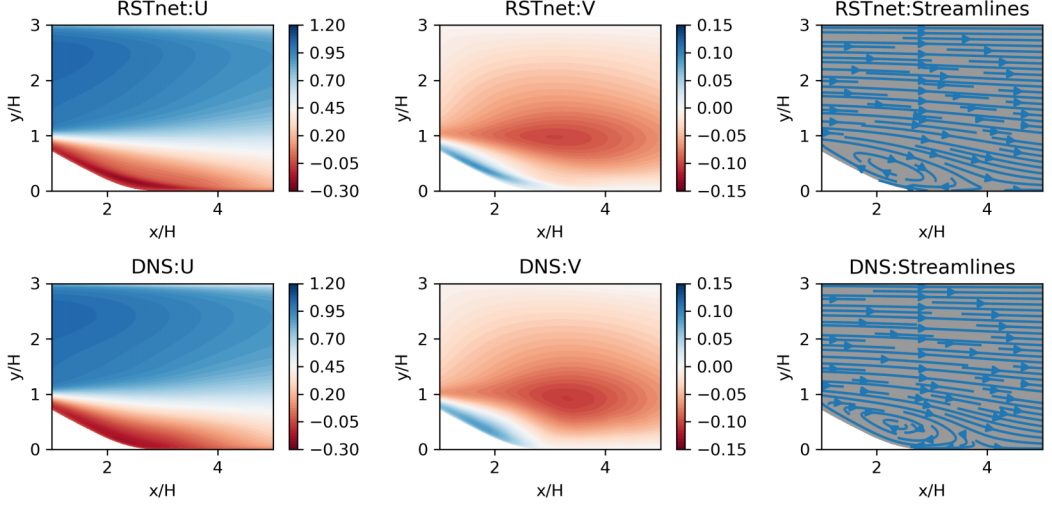


FIG. 20. Flow field of the mean flow quantities predicted by the RSTnet, at the periodic hill flow case with an elongating factor of $\alpha = 1.5$. The corresponding DNS results are also shown for comparison.

reflect nonphysical, diverging predictions in the absence of constraints on the network output. A similarly large, though less extreme, reduction is observed for $\overline{v\overline{v}}$ in the ZPG case (86.23% down to 13.04%). For the periodic hill flow case, the contributions of the realizability constraint and the RST equation are comparable overall. The RST equation alone reduces the errors in $\overline{u\overline{u}}$ and $\overline{v\overline{v}}$ from 49.24% to 22.99% and from 26.32% to 12.25%, respectively, while the realizability constraint alone reduces the errors in U and V from 4.07% to 1.71% and from 15.48% to 9.90%, respectively. The relative contribution across cases follows the extent to which the unconstrained network output actually leaves the physically admissible region, which is quantified in Sec. III E and is the highest for the APG case and the lowest for the periodic hill case. The realizability constraint is therefore most valuable where the unconstrained solution would otherwise drift far outside the realizable

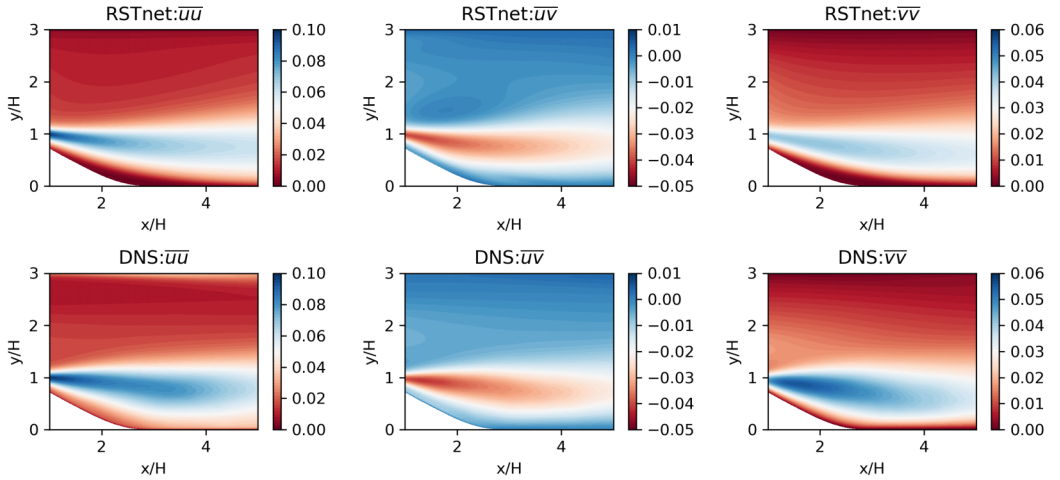


FIG. 21. Flow field of the turbulence quantities predicted by the RSTnet, at the periodic hill flow case with an elongating factor of $\alpha = 1.5$. The corresponding DNS results are also shown for comparison.

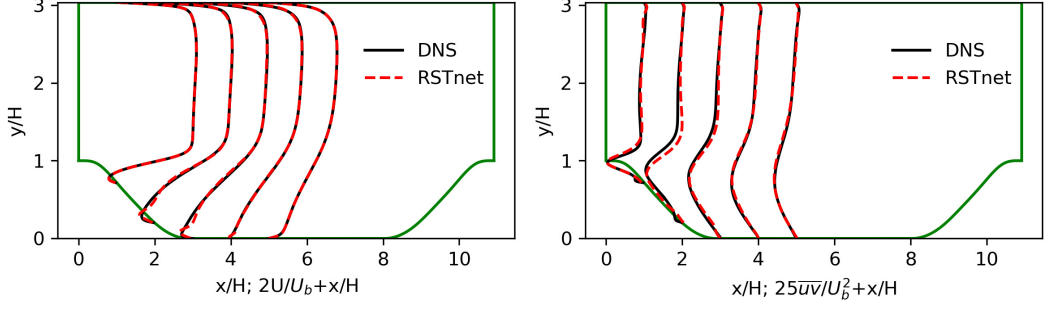


FIG. 22. Profiles of the streamwise velocity and the shear stress component of the Reynolds stress tensor predicted by the RSTnet, at the periodic hill flow case with an elongating factor of $\alpha = 1.5$. The corresponding DNS results are also shown for comparison.

region, while the RST equation contributes most where that drift is small and the advection and production physics is instead the contributing factor.

Imposing the RST equation without the realizability constraint can also degrade individual quantities relative to the unconstrained baseline, most visibly for the wall-normal velocity V in the periodic hill case (30.48% versus 15.48%). A plausible explanation is that enforcing the RST residual on a Reynolds stress field that is not constrained to be realizable introduces a demanding additional residual into the loss, which can be reduced through a nonphysical stress field at the expense of the mean flow. Of the reported quantities, V is the most sensitive in this respect, owing to its small magnitude relative to the streamwise velocity. Consistent with this interpretation, the degradation disappears once the two mechanisms are combined, with RSTnet-b recovering V to 8.71%.

Overall, these results support combining the RST formulation with the realizability constraint rather than relying on either mechanism in isolation, and indicate that the two address different deficiencies rather than duplicating one another. RSTnet-b is therefore the best variant overall, performing consistently well across all the considered flow cases.

E. Partial versus full enforcement of the realizability conditions

Having presented the performance comparison between RSTnet-a, RSTnet-b, and RSTnet-c for all flow cases in Secs. III A–III C, two aspects of the results are best discussed jointly here. These are the degree of realizability violation by the unconstrained and partially constrained networks,

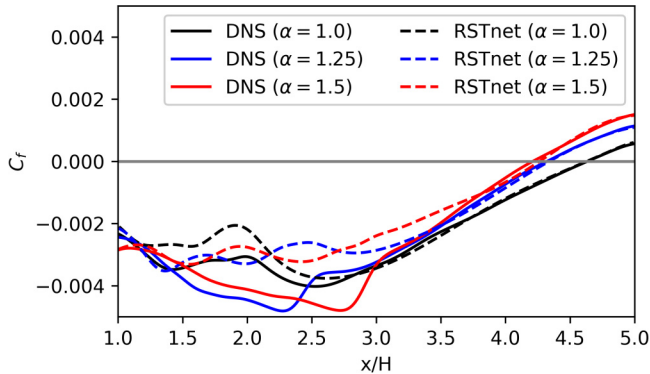


FIG. 23. Skin friction distributions for the periodic hill flow cases.

TABLE X. Performance comparison across the cases with/without the RST formulation and with/without the realizability constraint. The “No/No” and “No/Yes” rows are the two new free- τ baseline variants. The “Yes/No” and “Yes/Yes” rows correspond to RSTnet-a and RSTnet-b whose results were already reported in Secs. III A–III C.

Flow case	RST equation	Realizability	ϵ_U	ϵ_V	$\epsilon_{\overline{uu}}$	$\epsilon_{\overline{uv}}$	$\epsilon_{\overline{vv}}$
ZPG turbulent boundary layer	No	No	4.96%	18.94%	22.59%	16.17%	86.23%
	No	Yes	1.77%	21.28%	9.86%	22.69%	13.04%
	Yes	No	3.78%	11.75%	21.90%	10.57%	32.80%
	Yes	Yes	0.80%	5.41%	6.30%	7.92%	8.36%
APG turbulent boundary layer	No	No	4.55%	2.62%	388.04%	52.25%	631.48%
	No	Yes	0.99%	1.84%	19.37%	21.50%	32.99%
	Yes	No	4.34%	2.67%	224.89%	38.38%	226.78%
	Yes	Yes	0.53%	0.98%	10.14%	14.58%	15.04%
Periodic Hill ($\alpha = 1.0$)	No	No	4.07%	15.48%	49.24%	21.27%	26.32%
	No	Yes	1.71%	9.90%	17.30%	32.60%	23.91%
	Yes	No	4.11%	30.48%	22.99%	18.92%	12.25%
	Yes	Yes	1.73%	8.71%	15.56%	16.52%	12.52%

and the resulting comparison between RSTnet-b, which enforces the nonnegativity condition, and RSTnet-c, which enforces the full set including the Cauchy-Schwarz inequality.

As the nonnegativity and Cauchy-Schwarz conditions are equivalent to the Reynolds stress tensor being positive semi-definite, a normalized deficit can be defined to directly quantify the realizability violation as

$$\delta = \frac{\max(0, -\lambda_{\min})}{\tau_{\text{ref}}}, \quad (20)$$

where $\lambda_{\min}$ is the smaller eigenvalue of the predicted in-plane Reynolds stress tensor and τ_{ref} is the maximum of $\frac{1}{2}(\overline{uu} + \overline{vv})$ over the reference field. The normalization uses this global scale rather than the local turbulence intensity, so that the measure does not become ill-conditioned in the outer region where all the Reynolds stresses approach zero. The maximum value of δ and its mean over the points where it is nonzero are reported for all three RSTnets in Table XI. By construction δ is strictly zero for RSTnet-c, while RSTnet-b enforces nonnegativity alone, so the deficit δ reflects the Cauchy-Schwarz violation only. For RSTnet-a the deficit is largely due to negative normal stresses, which is exactly the failure mode that the RSTnet-b tackles.

TABLE XI. Maximum $\delta_{\max}$ and the mean value $\bar{\delta}$ of the normalized realizability deficit δ [which is defined by Eq. (20)] for the three RSTnets and all three flow cases. By construction, δ is strictly zero for RSTnet-c.

Flow case	Method	$\delta_{\max}$	$\bar{\delta}$
ZPG turbulent boundary layer	RSTnet-a	0.0875	0.0238
	RSTnet-b	0.0078	0.0031
	RSTnet-c	0.0000	0.0000
APG turbulent boundary layer	RSTnet-a	1.0354	0.2104
	RSTnet-b	0.0356	0.0036
	RSTnet-c	0.0000	0.0000
Periodic Hill ($\alpha = 1.0$)	RSTnet-a	0.0457	0.0164
	RSTnet-b	0.0268	0.0123
	RSTnet-c	0.0000	0.0000

As shown in Table XI, the realizability violation for the unconstrained RSTnet-a varies by more than an order of magnitude across the flow cases, from $\delta_{\max} = 0.0457$ for the periodic hill case and 0.0875 for ZPG to 1.0354 for APG. At the most severe case the negative eigenvalue is comparable to the peak turbulence intensity of the reference field. This is also consistent with the severe error blow-up already noted for the unconstrained free- τ baseline in the APG case in Sec. III D, further demonstrating the benefits of architecturally enforcing realizability. For RSTnet-b the residual deficit is smaller in every flow case, never exceeding $\delta_{\max} = 0.0356$, and smaller than that of RSTnet-a by more than an order of magnitude for the two boundary layer cases. The averaged deficit for RSTnet-b is at most 0.0123. Therefore, the residual Cauchy-Schwarz violation left by RSTnet-b is small in comparison with the turbulence intensity, even at the most severe point in the flow domain.

This directly informs the comparison between RSTnet-b and RSTnet-c. RSTnet-c, which additionally enforces the Cauchy-Schwarz inequality for the Reynolds shear stress, underperformed RSTnet-b for the ZPG and APG boundary layer cases, while for the periodic hill case it achieved a clear improvement specifically in the shear stress $\overline{uv}$ prediction, with a mixed picture for the other quantities. Three complementary explanations are discussed below.

First, RSTnet-c's specific architectural implementation of the Cauchy-Schwarz inequality constructs the shear stress output as the normal stresses multiplied by a bounded factor, therefore coupling the shear stress prediction to the normal stress predictions. In turbulent boundary layer flows, the velocity field is primarily dictated by the Reynolds shear stress rather than the normal stresses, so the considered underdetermined system, constrained only by boundary data, will generally yield a more reliable reconstruction of the shear stress than of the normal stresses. Imposing a dependence of the more reliably reconstructed shear stress on the less reliably reconstructed normal stresses may therefore propagate additional error into the shear stress prediction, which would explain the deteriorated performance of RSTnet-c specifically for the boundary layer cases.

Second, little violation remains for the additional constraint to correct. The residual deficit left by RSTnet-b is bounded by $\delta_{\max} = 0.0356$ in every flow case, so the realizability violation available to be removed is small, and the deterioration seen for the two boundary layer cases plausibly reflects optimization difficulties introduced by the architecture itself rather than the physical validity of the constraint. A concrete candidate for such a difficulty is the saturating gradient of the tanh activation used to construct the normalized shear stress, which can produce very small gradients for large preactivation magnitudes and thus impede training process.

Third, a more physically grounded explanation is that enforcing the full set of conditions may be overly restrictive on the admissible solution space. The same argument is made by the work in Ref. [54], where the strong realizability is judged overly restrictive as a constraint on the model coefficients because the limiting states of turbulence are not expected to be reached. This is also in line with the earlier findings in Refs. [52,63] which support the use of the weak form of realizability in practical RST closures. However, it is worth noting that the explanation here is analogous rather than a direct implication, as the conditions in RST modeling constrain the evolution of the Reynolds stress along its decay trajectory, while the conditions here act directly on the predicted state.

These three explanations should be treated as plausible interpretations rather than a conclusively established mechanism. To further isolate the effect of the physical constraint from the specific enforcing mechanism, future work may involve the use of a wider set of test cases spanning more varied anisotropy states, as well as alternative mechanisms for imposing the Cauchy-Schwarz condition, e.g., different architectural designs and constrained-optimization approaches.

IV. CONCLUSIONS

This work proposed the Reynolds stress transport nets (RSTnets) for reconstructing RANS solutions from the boundary data through embedding the Reynolds stress transport formulation via physics-informed neural networks. Different from previous works which either rely on the empirical modeling of all the unclosed terms or leave the Reynolds stress completely unmodeled,

the proposed RSTnets were developed to reconstruct the flow fields in the domain based on the first- and second-order flow quantities at the domain boundary through incorporating as much physical information as possible while avoiding empirical modeling of the unclosed terms. Specifically, the Reynolds stress transport equations were first formulated where the exact forms of the advection and the production terms, which can be directly derived from the Reynolds averaging process, were employed and all the other unclosed terms were absorbed into a single inadequacy term. The unclosed equation systems were then embedded via the PINNs approach. Such design, on the one hand, makes use of the power of PINNs for tackling ill-posed problems and, on the other hand, ensures the simplicity and generalizability of the proposed RSTnets. Furthermore, the realizability condition for the Reynolds stress, which was often considered as the minimum requirement for preventing RANS models from producing nonphysical results, was also incorporated in the proposed RSTnet to improve its performance. This was achieved via customized neural network designs that strictly (architecturally) enforce either the nonnegativity condition alone (RSTnet-b) or the full set of realizability conditions, including the Cauchy-Schwarz inequality for the shear stress components (RSTnet-c).

For performance evaluation, the proposed RSTnets were applied to a set of flow cases including a ZPG turbulent boundary layer flow, an APG turbulent boundary layer flow, and a set of flow cases over periodic hills of parameterized geometries. For each test case, a comprehensive hyperparameter study was first carried out and then the training convergence regarding the boundary points and the collocation points was investigated. After that, a comparison study was carried out to demonstrate the importance of imposing the realizability condition for the Reynolds stress. The results showed that the RSTnet with a customized design enforcing nonnegativity of normal stresses significantly improved the prediction performance for all the test cases, while the further consideration of the Cauchy-Schwarz inequality for Reynolds shear stresses in the RSTnet did not lead to further improvement overall. In more detail, the performance of RSTnet-c deteriorated for the ZPG and APG boundary layer flows, while for the periodic hill case it improved the shear stress prediction specifically, so the less restrictive nonnegativity condition alone (RSTnet-b) was the preferable choice. Several factors may contribute to this contrast, among them that enforcing the full set of conditions is overly restrictive, which is the argument advanced in Ref. [54] for the analogous restriction of a model's coefficient space, and is consistent with the difficulty of imposing the strong form of realizability in practical closures reported in Ref. [63].

For all the flow cases, the proposed RSTnet with mechanism of enforcing nonnegativity of normal stresses achieved great accuracy for all the flow quantities as well as the derived quantity of interest, i.e., the skin-friction distribution. For the ZPG turbulent boundary layer flow, the relative L_2 errors for the mean flow quantities $[U, V]$ were just 0.8% and 5.41%, respectively, while the relative L_2 error for turbulence quantities was on average 7.53%. For the APG turbulent boundary layer flow, the relative L_2 errors for the mean flow quantities $[U, V]$ were just 0.53% and 0.98%, respectively, while the relative L_2 error for turbulence quantities was on average 13.25%. For the flows over periodic hills of parameterized geometries, the relative L_2 errors for the mean flow quantities $[U, V]$ were, on average, 1.82% and 8.97%, respectively, while the relative L_2 error for turbulence quantities was on average 17.10%. Last but not least, the influence of the weights balancing individual loss terms in the total loss was also studied for all the test cases. The results showed that unbalanced weighting between physics-informed and data-driven parts as well as between the governing equations for mean flow quantities and Reynolds stresses both led to worse performances. The unbalanced weighting is thus not recommended for the RSTnets.

The prediction results for both the mean flow and the turbulence quantities are encouraging. The scope of that gain should, however, be kept in mind. Flows like the ZPG and APG boundary layers can be relatively well modeled by much simpler closures, e.g., one- or two-equation eddy-viscosity models, so the improvement offered by the present formulation is expected to be modest for the mean flow quantities. Its main contribution lies in the Reynolds stress field prediction, which such closures cannot provide accurately. Another important consideration is that, as a new way of reconstructing RANS solutions from boundary data via fusing Reynolds stress advection, production,

and realizability without any empirical models, the proposed RSTnets face the inevitable challenge due to the underdeterminedness of the system. The proposed RSTnets, although performing well for all the test cases, will generally be biased toward “smoother” solutions that obey the imposed soft and hard constraints, due to the spectral bias of artificial neural networks. Therefore, for underdetermined systems like the one investigated in this work, the key consideration lies in incorporating additional physics-based constraints into the system so that the ML model converges to a physically admissible solution, eliminating or discriminating the solutions that do not obey these physical constraints. To the best of the author’s knowledge, this is the key reason why the proposed RSTnets perform better than the previous work [47]. Injecting additional physical constraints is also the key reason why the flow field reconstruction is improved in Ref. [46], which also involves an underdetermined system. However, it is worth noting that, if the RSTnets were applied to a particular flow setup where there are multiple physically admissible solutions and the target solution is the one with the higher-frequency content, the correct solution will likely be missed. In this case, techniques for tackling spectral bias, such as injecting prior knowledge of the characteristic length scale via a Fourier-encoding layer, may be applied to guide the training to converge to the correct solution.

In addition, it is worth noting that, from the perspective of system underdeterminedness, this work can be seen as an augmentation of the work in Eivazi *et al.* (2022) by extending a system with six unknowns and three equations to a system with nine unknowns and six equations. Through this augmentation, turbulence advection, production, and realizability are then effectively leveraged to make the training of the RSTnets converge to the right solution. Regarding solution multiplicity in PINNs, it is by itself a very interesting research area and the reader may refer to the work in Ref. [64] for further insights. However, while the NN initialization plays a dominant role when solving determined systems with multiple solutions [64], this study shows that solving underdetermined systems with PINNs mainly depends on the training procedure (including the model specification and the loss composition) rather than the initialization of NN parameters.

To further improve the proposed RSTnets, future work may involve the development of new network architectures for enforcing realizability conditions, the consideration of other physical principles for further narrowing/constraining the solution space, and the extension of the flow field prediction with uncertainty quantification. In particular, two directions appear promising for injecting further physical structure into the unmodeled terms. First, physical constraints such as realizability could be imposed on the inadequacy term η rather than on the Reynolds stresses, in the spirit of the work in Ref. [54]. This is also the form in which realizability enters traditional second-moment closures [63]. In the present formulation the unclosed terms are collected in η , which is an output of the network and thus could be constrained either through architectural design, as done here for the Reynolds stresses, or through additional loss terms. The demonstrated capability of PINNs for solving ill-posed problems means that such constraints could be imposed without prescribing a fixed functional form for the pressure-strain and dissipation terms, although a fixed form, e.g., the linear and nonlinear closure forms proposed in Refs. [48,51–54], could equally be adopted. These options span a range between the formulation used here, which is conditioned on as few modeling assumptions as possible, and a traditional closure, and exploring it would allow the trade-off between empirical calibration and the underdeterminedness left to the network to be examined directly. Second, a tensor-basis expansion, following the works in Refs. [18,19], could be embedded within the present RST formulation. Unlike the standard TBNN formulation where the anisotropy is a local, algebraic function of the mean strain-rate and rotation-rate tensors alone, such a basis would need to be augmented with explicit dependence on the Reynolds stress tensor τ itself. Any model operating within the Reynolds-stress-transport-equation framework requires the current anisotropy state as an input, which has not been investigated in any TBNN studies yet. Such an extension would thus be very interesting for data-driven turbulence modeling in general. It would also improve the interpretability of the proposed approach, as the network output would become a small set of scalar coefficients multiplying known tensors, rather than an unstructured field produced by a black-box model.

TABLE XII. Performance comparison between the multiplicative inadequacy and the additive inadequacy formulations for the RSTnet. The results are all obtained with the RSTnet-b.

Flow case	Formulation	ϵ_U	ϵ_V	$\epsilon_{\overline{uu}}$	$\epsilon_{\overline{uv}}$	$\epsilon_{\overline{vv}}$
ZPG turbulent boundary layer	Multiplicative	2.17%	6.62%	14.02%	22.18%	13.74%
	Additive	0.80%	5.41%	6.30%	7.92%	8.36%
APG turbulent boundary layer	Multiplicative	0.48%	2.06%	26.60%	32.49%	28.87%
	Additive	0.53%	0.98%	10.14%	14.58%	15.04%
Periodic hill ($\alpha = 1.0$)	Multiplicative	1.83%	9.13%	17.86%	14.42%	10.72%
	Additive	1.73%	8.71%	15.56%	16.52%	12.52%
Periodic hill ($\alpha = 1.25$)	Multiplicative	2.01%	10.44%	20.24%	13.72%	14.57%
	Additive	1.73%	8.69%	19.81%	14.15%	14.03%
Periodic hill ($\alpha = 1.5$)	Multiplicative	3.70%	22.42%	31.34%	13.86%	35.33%
	Additive	2.01%	9.52%	23.84%	14.35%	23.14%

The proposed RSTnets require flow information only at the domain boundary rather than throughout the domain interior. They are therefore of potential interest for reconstructing turbulent flow fields from real experimental measurements that are naturally confined to boundaries or sparse sensor locations. Examples include wall-mounted pressure and shear-stress sensors or particle image velocimetry planes positioned at the domain edges, a setup of considerable practical interest for the sparse sensing of turbulent flows. Extending the framework to accommodate the measurement noise, along with the uncertainty quantification extension noted above, is a natural next step in this direction. More broadly, the underlying boundary-data-only reconstruction methodology enforces the exact terms in the transport equation through the PINNs approach and leaves the unclosed term to the network, while further imposing physical constraints. The overall methodology is not restricted to the Reynolds stress transport equation, and could equally be applied to other averaged transport equations with an unclosed flux term, such as the passive scalar transport equation for species concentration reconstruction from boundary data.

ACKNOWLEDGMENTS

This work was supported by Royal Academy of Engineering under the Research Fellowship scheme (Grant No. RF-2425-24-411). The author also acknowledges the Scientific Computing Research Technology Platform (SCRTP) at the University of Warwick for providing High-Performance Computing resources.

DATA AVAILABILITY

The codes for this work are publicly available in GitHub at [65].

APPENDIX A: MULTIPLICATIVE INADEQUACY FORMULATION FOR THE RSTNET

The results by employing the multiplicative inadequacy formulation, i.e., Eq. (11), for the RSTnet are given in this section. The prediction accuracy is given in Table XII, which includes the relative L_2 errors for both the mean flow quantities and the turbulence quantities at all the test cases, i.e., the ZPG turbulent boundary layer, the APG turbulent boundary layer, and the flows over periodic hills with $\alpha = 1.0$, $\alpha = 1.25$, and $\alpha = 1.5$. The results by the additive inadequacy formulation, which has been reported in the main paper, are also included in Table XII for convenience. As shown, the additive formulation performs better than the multiplicative formulation at all the test cases, supporting the choice of the additive formulation in the main paper.

TABLE XIII. The performance comparison between considering only the advection term in the RSTnet and considering both the advection and production terms.

Flow case	Formulation	ϵ_U	ϵ_V	$\epsilon_{\overline{uu}}$	$\epsilon_{\overline{uv}}$	$\epsilon_{\overline{vv}}$
ZPG turbulent boundary layer	Advection only	2.79%	17.02%	11.11%	30.23%	19.91%
	Both	0.80%	5.41%	6.30%	7.92%	8.36%
APG turbulent boundary layer	Advection only	0.74%	1.31%	26.43%	24.25%	21.49%
	Both	0.53%	0.98%	10.14%	14.58%	15.04%
Periodic hill ($\alpha = 1.0$)	Advection only	2.11%	9.61%	24.60%	28.68%	22.83%
	Both	1.73%	8.71%	15.56%	16.52%	12.52%
Periodic hill ($\alpha = 1.25$)	Advection only	2.28%	11.93%	18.66%	14.77%	17.36%
	Both	1.73%	8.69%	19.81%	14.15%	14.03%
Periodic hill ($\alpha = 1.5$)	Advection only	3.96%	24.14%	36.35%	21.36%	42.06%
	Both	2.01%	9.52%	23.84%	14.35%	23.14%

APPENDIX B: EFFECTS OF THE TURBULENCE PRODUCTION TERM IN THE RSTNET

To illustrate the benefits of incorporating the turbulence production term in the proposed RSTnet, numerical tests are carried out here, where the cases with consideration of only the advection terms in the Reynolds stress transport equations are compared with the cases considering both the advection and production terms. The results for all the cases are given in Table XIII. As shown, the prediction is improved clearly by considering both the production and the advection terms, demonstrating the importance of the production term in the proposed RSTnet.

From the general perspective of Reynolds stress transport modeling, unlike the pressure-strain, diffusion, and dissipation terms absorbed into η , which require closure in traditional second-moment models, the production term is exact and is always retained explicitly. Lumping it into the inadequacy term will therefore discard exact physical information that can be directly obtained from the local mean velocity gradient and the current Reynolds stress state, forcing the network to relearn a known relationship. In addition, production is typically the dominant source term balancing the other terms in the Reynolds stress transport equation. Therefore it is retained explicitly in this work, and this choice is also justified by the results in Table XIII.

APPENDIX C: UNIFORM VERSUS WALL-CLUSTERED POINT DISTRIBUTION FOR THE BOUNDARY LAYER FLOW CASES

To quantify the benefit of using the wall-clustered point distribution for the ZPG and APG turbulent boundary layer cases, the RSTnet-b is retrained for each case with the same total number of boundary points redistributed uniformly in the wall-normal direction instead. All other hyperparameters and training settings are kept unchanged. The results are given in Table XIV. As shown, redistributing the points uniformly degrades the accuracy for every reported quantity in both flow cases, confirming the benefit of the wall-clustered point distribution used in the main results.

TABLE XIV. Performance comparison between uniform and wall-clustered point distributions at the ZPG and APG turbulent boundary layer flow cases. The results are all obtained with the RSTnet-b. The wall-clustered rows reproduce the results already reported in Secs. III A and III B.

Flow case	Point distribution	ϵ_U	ϵ_V	$\epsilon_{\overline{uu}}$	$\epsilon_{\overline{uv}}$	$\epsilon_{\overline{vv}}$
ZPG turbulent boundary layer	Uniform	2.16%	9.46%	8.45%	14.23%	27.23%
	Wall-clustered	0.80%	5.41%	6.30%	7.92%	8.36%
APG turbulent boundary layer	Uniform	0.87%	1.14%	21.68%	19.59%	21.55%
	Wall-clustered	0.53%	0.98%	10.14%	14.58%	15.04%

- [1] S. L. Brunton, B. R. Noack, and P. Koumoutsakos, Machine learning for fluid mechanics, *Annu. Rev. Fluid Mech.* **52**, 477 (2020).
- [2] J. N. Kutz, Deep learning in fluid dynamics, *J. Fluid Mech.* **814**, 1 (2017).
- [3] R. Maulik, K. Fukami, N. Ramachandra, K. Fukagata, and K. Taira, Probabilistic neural networks for fluid flow surrogate modeling and data recovery, *Phys. Rev. Fluids* **5**, 104401 (2020).
- [4] X. Zhang, F. Xie, T. Ji, Z. Zhu, and Y. Zheng, Multi-fidelity deep neural network surrogate model for aerodynamic shape optimization, *Comput. Methods Appl. Mech. Eng.* **373**, 113485 (2021).
- [5] M. Tang, Y. Liu, and L. J. Durlofsky, Deep-learning-based surrogate flow modeling and geological parameterization for data assimilation in 3D subsurface flow, *Comput. Methods Appl. Mech. Eng.* **376**, 113636 (2021).
- [6] J. Zhang and S. Fu, An efficient approach for quantifying parameter uncertainty in the SST turbulence model, *Comput. Fluids* **181**, 173 (2019).
- [7] S. Shahane, N. R. Aluru, and S. P. Vanka, Uncertainty quantification in three dimensional natural convection using polynomial chaos expansion and deep neural networks, *Int. J. Heat Mass Transf.* **139**, 613 (2019).
- [8] N. Geneva and N. Zabaras, Quantifying model form uncertainty in Reynolds-averaged turbulence models with Bayesian deep neural networks, *J. Comput. Phys.* **383**, 125 (2019).
- [9] J. Rabault, M. Kuchta, A. Jensen, U. Réglade, and N. Cerardi, Artificial neural networks trained through deep reinforcement learning discover control strategies for active flow control, *J. Fluid Mech.* **865**, 281 (2019).
- [10] D. Fan, L. Yang, Z. Wang, M. S. Triantafyllou, and G. E. Karniadakis, Reinforcement learning for bluff body active flow control in experiments and simulations, *Proc. Natl. Acad. Sci. USA* **117**, 26091 (2020).
- [11] P. Garnier, J. Viquerat, J. Rabault, A. Larcher, A. Kuhnle, and E. Hachem, A review on deep reinforcement learning for fluid mechanics, *Comput. Fluids* **225**, 104973 (2021).
- [12] D. Kochkov, J. A. Smith, A. Alieva, Q. Wang, M. P. Brenner, and S. Hoyer, Machine learning-accelerated computational fluid dynamics, *Proc. Natl. Acad. Sci. USA* **118**, e2101784118 (2021).
- [13] R. Vinuesa and S. L. Brunton, Enhancing computational fluid dynamics with machine learning, *Nat. Comput. Sci.* **2**, 358 (2022).
- [14] C. Lin, M. Maxey, Z. Li, and G. E. Karniadakis, A seamless multiscale operator neural network for inferring bubble dynamics, *J. Fluid Mech.* **929**, A18 (2021).
- [15] S. Cai, Z. Wang, L. Lu, T. A. Zaki, and G. E. Karniadakis, DeepM&Mnet: Inferring the electroconvection multiphysics fields based on operator approximation by neural networks, *J. Comput. Phys.* **436**, 110296 (2021).
- [16] K. Duraisamy, G. Iaccarino, and H. Xiao, Turbulence modeling in the age of data, *Annu. Rev. Fluid Mech.* **51**, 357 (2019).
- [17] K. Duraisamy, Perspectives on machine learning-augmented Reynolds-averaged and large eddy simulation models of turbulence, *Phys. Rev. Fluids* **6**, 050504 (2021).
- [18] J. Ling, A. Kurzawski, and J. Templeton, Reynolds averaged turbulence modelling using deep neural networks with embedded invariance, *J. Fluid Mech.* **807**, 155 (2016).
- [19] J. Cai, P.-E. Angeli, J.-M. Martinez, G. Damblin, and D. Lucor, Revisiting tensor basis neural network for Reynolds stress modeling: Application to plane channel and square duct flows, *Comput. Fluids* **275**, 106246 (2024).
- [20] S. B. Pope, A more general effective-viscosity hypothesis, *J. Fluid Mech.* **72**, 331 (1975).
- [21] J.-X. Wang, J.-L. Wu, and H. Xiao, Physics-informed machine learning approach for reconstructing Reynolds stress modeling discrepancies based on DNS data, *Phys. Rev. Fluids* **2**, 034603 (2017).
- [22] J.-L. Wu, R. Sun, S. Laizet, and H. Xiao, Representation of stress tensor perturbations with application in machine-learning-assisted turbulence modeling, *Comput. Methods Appl. Mech. Eng.* **346**, 707 (2019).
- [23] M. A. Cruz, R. L. Thompson, L. E. Sampaio, and R. D. Bacchi, The use of the Reynolds force vector in a physics informed machine learning approach for predictive turbulence modeling, *Comput. Fluids* **192**, 104258 (2019).

- [24] P. S. Volpiani, M. Meyer, L. Franceschini, J. Dandois, F. Renac, E. Martin, O. Marquet, and D. Sipp, Machine learning-augmented turbulence modeling for RANS simulations of massively separated flows, *Phys. Rev. Fluids* **6**, 064607 (2021).
- [25] A. Cremades, S. Hoyas, R. Deshpande, P. Quintero, M. Lellep, W. J. Lee, J. P. Monty, N. Hutchins, M. Linkmann, I. Marusic, and R. Vinuesa, Identifying regions of importance in wall-bounded turbulence through explainable deep learning, *Nat. Commun.* **15**, 3864 (2024).
- [26] R. Villiers, V. Mons, D. Sipp, E. Lamballais, and M. Meldi, Enhancing unsteady Reynolds-averaged Navier-Stokes modelling from sparse data through sequential data assimilation and machine learning, *Flow Turbul. Combust.* **115**, 991 (2025).
- [27] M. Raissi, P. Perdikaris, and G. E. Karniadakis, Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations, *J. Comput. Phys.* **378**, 686 (2019).
- [28] M. Raissi, Z. Wang, M. S. Triantafyllou, and G. E. Karniadakis, Deep learning of vortex-induced vibrations, *J. Fluid Mech.* **861**, 119 (2019).
- [29] M. Raissi, A. Yazdani, and G. E. Karniadakis, Hidden fluid mechanics: Learning velocity and pressure fields from flow visualizations, *Science* **367**, 1026 (2020).
- [30] J. Zhang and X. Zhao, Digital twin of wind farms via physics-informed deep learning, *Energy Convers. Manage.* **293**, 117507 (2023).
- [31] S. Cai, H. Li, F. Zheng, F. Kong, M. Dao, G. E. Karniadakis, and S. Suresh, Artificial intelligence velocimetry and microaneurysm-on-a-chip for three-dimensional analysis of blood flow in physiology and disease, *Proc. Natl. Acad. Sci. USA* **118**, e2100697118 (2021).
- [32] X. Jin, S. Cai, H. Li, and G. E. Karniadakis, NSFnets (Navier-Stokes flow nets): Physics-informed neural networks for the incompressible Navier-Stokes equations, *J. Comput. Phys.* **426**, 109951 (2021).
- [33] D. Lucor, A. Agrawal, and A. Sergeant, Simple computational strategies for more effective physics-informed neural networks modeling of turbulent natural convection, *J. Comput. Phys.* **456**, 111022 (2022).
- [34] L. Yang, X. Meng, and G. E. Karniadakis, B-PINNs: Bayesian physics-informed neural networks for forward and inverse PDE problems with noisy data, *J. Comput. Phys.* **425**, 109913 (2021).
- [35] Y. Yang and P. Perdikaris, Adversarial uncertainty quantification in physics-informed neural networks, *J. Comput. Phys.* **394**, 136 (2019).
- [36] X. Meng and G. E. Karniadakis, A composite neural network that learns from multi-fidelity data: Application to function approximation and inverse PDE problems, *J. Comput. Phys.* **401**, 109020 (2020).
- [37] A. Krishnapriyan, A. Gholami, S. Zhe, R. Kirby, and M. W. Mahoney, Characterizing possible failure modes in physics-informed neural networks, *Adv. Neural Info. Process. Syst.* **34**, 26548 (2021).
- [38] S. Wang, Y. Teng, and P. Perdikaris, Understanding and mitigating gradient flow pathologies in physics-informed neural networks, *SIAM J. Sci. Comput.* **43**, A3055 (2021).
- [39] S. Wang, X. Yu, and P. Perdikaris, When and why PINNs fail to train: A neural tangent kernel perspective, *J. Comput. Phys.* **449**, 110768 (2022).
- [40] F. Pioch, J. H. Harmening, A. M. Müller, F.-J. Peitzmann, D. Schramm, and O. el Moctar, Turbulence modeling for physics-informed neural networks: Comparison of different RANS models for the backward-facing step flow, *Fluids* **8**, 43 (2023).
- [41] S. Ghosh, A. Chakraborty, G. O. Brikis, and B. Dey, RANS-PINN based simulation surrogates for predicting turbulent flows, *arXiv:2306.06034*.
- [42] X. Ren, P. Hu, H. Su, F. Zhang, and H. Yu, Physics-informed neural networks for transonic flow around a cylinder with high Reynolds number, *Phys. Fluids* **36**, 036129 (2024).
- [43] Z. Li, F. Montomoli, and S. Sharma, Investigation of compressor cascade flow using physics-informed neural networks with adaptive learning strategy, *AIAA J.* **62**, 1400 (2024).
- [44] S. Yazdani and M. Tahani, Data-driven discovery of turbulent flow equations using physics-informed neural networks, *Phys. Fluids* **36**, 035107 (2024).
- [45] L. Sliwinski and G. Rigas, Mean flow reconstruction of unsteady flows using physics-informed neural networks, *Data-Centric Eng.* **4**, e4 (2023).
- [46] Y. Patel, V. Mons, O. Marquet, and G. Rigas, Turbulence model augmented physics-informed neural networks for mean-flow reconstruction, *Phys. Rev. Fluids* **9**, 034605 (2024).

- [47] H. Eivazi, M. Tahani, P. Schlatter, and R. Vinuesa, Physics-informed neural networks for solving Reynolds-averaged Navier–Stokes equations, *Phys. Fluids* **34**, 075117 (2022).
- [48] B. E. Launder, G. J. Reece, and W. Rodi, Progress in the development of a Reynolds-stress turbulence closure, *J. Fluid Mech.* **68**, 537 (1975).
- [49] S. B. Pope, *Turbulent Flows* (Cambridge University Press, Cambridge, UK, 2000).
- [50] Turbulence Model Benchmarking Working Group, AIAA Fluid Dynamics Technical Committee, Turbulence modeling resource, <https://tmbwg.github.io/turbmodels/> (2026), accessed 2026-08-05.
- [51] C. G. Speziale, S. Sarkar, and T. B. Gatski, Modelling the pressure-strain correlation of turbulence: An invariant dynamical systems approach, *J. Fluid Mech.* **227**, 245 (1991).
- [52] S. Sarkar and C. G. Speziale, A simple nonlinear model for the return to isotropy in turbulence, *Phys. Fluids* **2**, 84 (1990).
- [53] M. K. Chung and S. K. Kim, A nonlinear return-to-isotropy model with Reynolds number and anisotropy dependency, *Phys. Fluids* **7**, 1425 (1995).
- [54] T. Homan, O. B. Shende, and A. Mani, Reynolds stress decay modeling informed by anisotropically forced homogeneous turbulence, *Phys. Rev. Fluids* **9**, 094608 (2024).
- [55] P. A. Durbin and B. A. Pettersson Reif, *Statistical Theory and Modeling for Turbulent Flows* (John Wiley & Sons, New York, NY, 2011).
- [56] G. Alfonsi, Reynolds-averaged Navier-Stokes equations for turbulence modeling, *Appl. Mech. Rev.* **62**, 040802 (2009).
- [57] P. A. Durbin, Some recent developments in turbulence closure modeling, *Annu. Rev. Fluid Mech.* **50**, 77 (2018).
- [58] E. Kiyani, K. Shukla, J. F. Urbán, J. Darbon, and G. E. Karniadakis, Optimizing the optimizer for physics-informed neural networks and Kolmogorov–Arnold networks, *Comput. Methods Appl. Mech. Eng.* **446**, 118308 (2025).
- [59] G. Eitel-Amor, R. Örlü, and P. Schlatter, Simulation and validation of a spatially evolving turbulent boundary layer up to $Re_\theta = 8300$, *Int. J. Heat Fluid Flow* **47**, 57 (2014).
- [60] A. Bobke, R. Vinuesa, R. Örlü, and P. Schlatter, History effects and near equilibrium in adverse-pressure-gradient turbulent boundary layers, *J. Fluid Mech.* **820**, 667 (2017).
- [61] H. Xiao, J.-L. Wu, S. Laizet, and L. Duan, Flows over periodic hills of parameterized geometries: A dataset for data-driven turbulence modeling from direct simulations, *Comput. Fluids* **200**, 104431 (2020).
- [62] P. Luo, Y. Tang, and Y. Liu, Modification of Reynolds-stress model with two-component limit for predicting separated flows, *Phys. Fluids* **37**, 085172 (2025).
- [63] C. G. Speziale, R. Abid, and P. A. Durbin, On the realizability of Reynolds stress turbulence closures, *J. Sci. Comput.* **9**, 369 (1994).
- [64] Z. Zou, Z. Wang, and G. E. Karniadakis, Learning and discovering multiple solutions using physics-informed neural networks with random initialization and deep ensemble, *Proc. R. Soc. A* **481**, 20250205 (2025).
- [65] J. Zhang, RSTnets, GitHub, 2026, <https://github.com/zhangxcii/RSTnets>.