

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/355092872>

The Potential of Machine Learning to Enhance Computational Fluid Dynamics

Preprint · October 2021

CITATIONS

0

READS

453

2 authors:



Ricardo Vinuesa
KTH Royal Institute of Technology
205 PUBLICATIONS 2,255 CITATIONS

[SEE PROFILE](#)



Steven L. Brunton
University of Washington Seattle
309 PUBLICATIONS 12,533 CITATIONS

[SEE PROFILE](#)

Some of the authors of this publication are also working on these related projects:



Randomized Dimension Reduction [View project](#)



Uncertainty Quantification in CFD and Simulation of Turbulence [View project](#)

The Potential of Machine Learning to Enhance Computational Fluid Dynamics

Ricardo Vinuesa^{1*} and Steven L. Brunton²

¹ FLOW, Engineering Mechanics, KTH Royal Institute of Technology, Stockholm, Sweden

² Department of Mechanical Engineering, University of Washington, Seattle, WA 98195, United States

Abstract

Machine learning is rapidly becoming a core technology for scientific computing, with numerous opportunities to advance the field of computational fluid dynamics. This paper highlights some of the areas of highest potential impact, including to accelerate direct numerical simulations, to improve turbulence closure modelling, and to develop enhanced reduced-order models. In each of these areas, it is possible to improve machine learning capabilities by incorporating physics into the process, and in turn, to improve the simulation of fluids to uncover new physical understanding. Despite the promise of machine learning described here, we also note that classical methods are often more efficient for many tasks. We also emphasize that in order to harness the full potential of machine learning to improve computational fluid dynamics, it is essential for the community to continue to establish benchmark systems and best practices for open-source software, data sharing, and reproducible research.

1 Introduction

The present revolution in machine learning (ML) is enabling numerous advances across a wide range of scientific and engineering disciplines [19, 85, 88, 99, 120]. One subject of significant potential is the numerical simulation of fluid flows, generally known as computational fluid dynamics (CFD). Fluid mechanics is an area of extreme importance, both from a fundamental science perspective and for many industrial engineering applications. Fluid flows are governed by the Navier–Stokes equations, which are partial differential equations (PDEs) modelling the conservation of mass and momentum in a Newtonian fluid. These PDEs are non-linear, due to convection, and they commonly exhibit time-dependent chaotic behavior, known as *turbulence*. Solving the Navier–Stokes equations for turbulent flows requires numerical methods that may be computationally expensive, or even intractable in many cases, due to the wide range of scales in space and time necessary to resolve these flows. As will be discussed below, there are various approaches to numerically solve these equations, involving different levels of fidelity and computational cost. Each of these approaches can benefit from machine learning.

In this contribution, we focus on the potential for machine learning to improve CFD, including possibilities to increase the speed of high-fidelity simulations, develop turbulence models with different levels of fidelity, and produce reduced-order models beyond what can be achieved with classical approaches. Several authors have surveyed the potential of machine learning to improve fluid mechanics [16, 21], including topics beyond the scope of CFD, such as experimental techniques, control applications, and related fields. Others have reviewed more specific aspects of ML for CFD, such as turbulence modeling [1, 31] or modeling and heat-transfer aspects of CFD for aerodynamic optimization [123]. Our discussion will address the middle ground of ML for CFD more broadly, with a schematic representation of topics covered in Fig. 1. Approaches to improve CFD with ML are aligned with a larger effort to incorporate ML into scientific computing, for instance via physics-informed neural networks (PINNs) [53, 97] or to accelerate computational chemistry [87, 88].

Before discussing the areas of high potential in detail, we would like to formulate a number of caveats which may limit the applicability of ML to certain areas of CFD. Firstly, **ML methods, such as deep learning, are often expensive to train and require large amounts of data. It is therefore important to identify areas where ML outperforms classical methods, which have been established for decades, and may be more accurate and efficient.** There is also a question of how the training data is generated, and whether the associated cost is taken into account when benchmarking. In this context, transfer learning is a promising area [46]. It is also worth noting that there are alternative machine learning techniques to deep learning that may be more appropriate for some tasks. Finally, it is important to assess the information about the training data available to the user: certain flow properties (*e.g.* incompressibility, periodicity, etc.) should be embedded in the ML model to increase training efficiency and prediction accuracy.

The remainder of this article is structured as follows: in §2 we discuss the potential of ML to accelerate high-fidelity fluid simulations; in §3 we highlight areas where ML can improve turbulence modelling; in §4 we explore the potential of ML to enhance reduced-order-models (ROMs); and finally, an outlook and emerging possibilities are provided in §5.

* Corresponding author: Ricardo Vinuesa (rvinuesa@mech.kth.se)

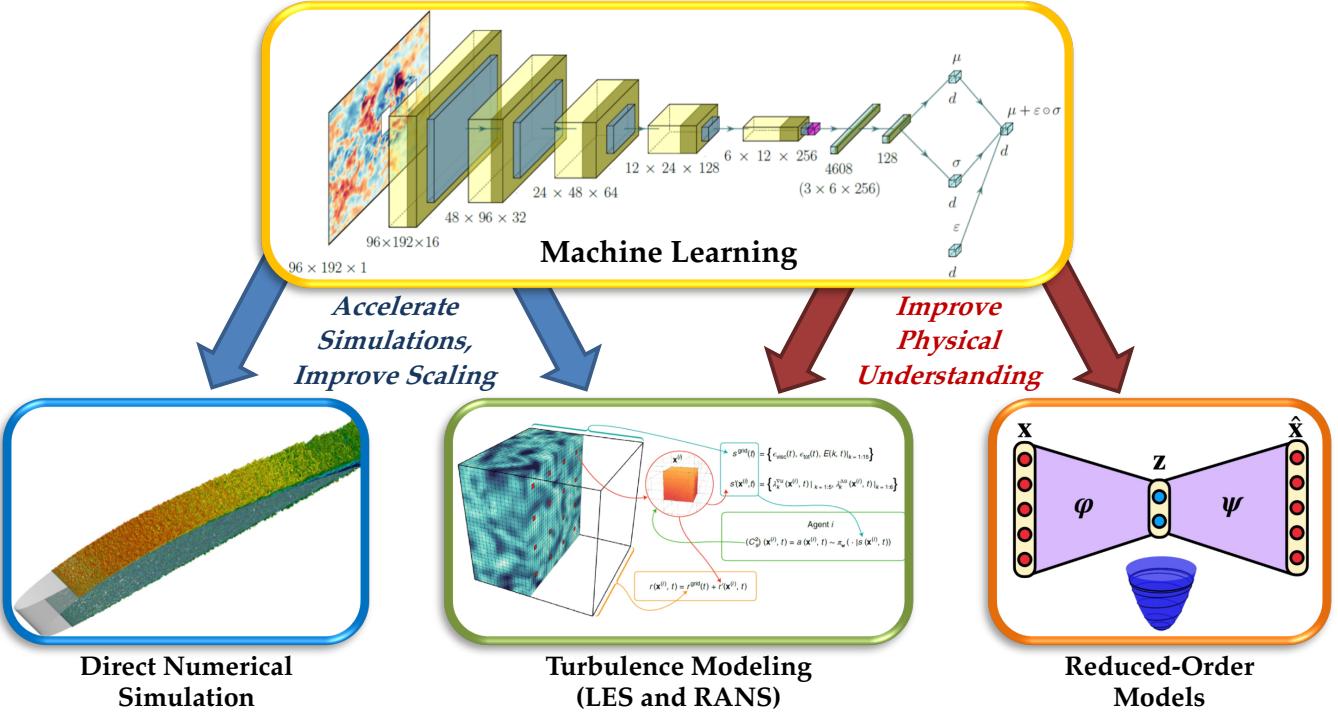


Figure 1: Summary of some of the most relevant areas where machine learning can enhance CFD, in the context of direct numerical simulations, turbulence modelling and reduced-order models. As described below, LES denotes large-eddy simulation and RANS Reynolds-averaged Navier–Stokes. Images reproduced from Refs. [32, 89, 118] with permission of the publishers.

2 Increasing the speed of direct numerical simulations

Direct numerical simulation (DNS) is a high-fidelity approach where the governing Navier–Stokes equations are discretized and integrated in time with enough degrees of freedom to resolve all flow structures. Turbulent flows exhibit a pronounced multi-scale character, with vortical structures across a range of sizes and energetic content. This complexity requires fine meshes and accurate computational methods to avoid distorting the underlying physics with numerical artifacts. With properly designed DNS, it is possible to obtain a representation of the flow field with the highest level of detail among CFD methods. However, the fine computational meshes required to resolve the smallest scales lead to exceedingly high computational costs, which increase with the Reynolds number [27].

A number of machine learning approaches have been developed recently to improve the efficiency of DNS. Bar-Sinai *et al.* [6] proposed a technique based on deep learning to estimate spatial derivatives in low-resolution grids, outperforming standard finite-difference methods. A similar approach was developed by Stevens and Colonius [112] to improve the results of fifth-order finite-difference schemes in the context of shock-capturing simulations. Other strategies to improve the performance of PDE solvers in coarser meshes have been developed by Li *et al.* [62–64]. Recently, Kochkov *et al.* [56] considered the two-dimensional Kolmogorov flow [26], which maintains fluctuations via a forcing term. They leveraged deep learning to develop a correction between fine- and coarse-resolution simulations, obtaining excellent agreement with reference simulations in meshes from 8 to 10 times coarser in each dimension, as shown in Fig. 2. These results promise to significantly reduce the computational cost of relevant fluid simulations, including weather [8], climate [105], engineering [119], and astrophysics [3].

Jeon and Kim [49] proposed to use a deep neural network to simulate the well-known finite-volume discretization scheme [37] employed in fluid simulations. They tested their method with reactive flows, obtaining excellent agreement with reference high-resolution data at one tenth the computational cost. However, they also documented errors with respect to the reference solution which increased with time. Another deep-learning approach, based on a fully-convolutional/long-short-term-memory (LSTM) network, was proposed by Stevens and Colonius [113] to improve the accuracy of finite-difference/finite-volume methods.

It is also possible to accelerate CFD by solving the Poisson equation with deep learning, as proposed by several research groups in various areas [108, 133]. The Poisson equation is frequently used in operator-splitting methods to discretize the Navier–Stokes equations [17], where first the velocity field is advected, and the resulting field \mathbf{u}^* does

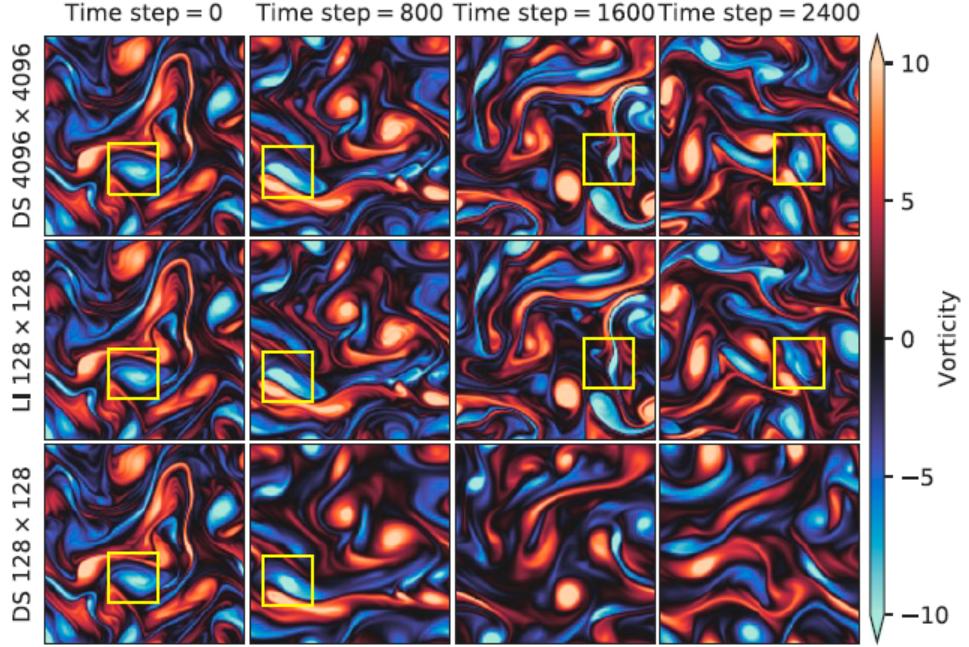


Figure 2: Sample results from the work by Kochkov *et al.* [56], where the instantaneous vorticity field is shown for (top) the simulation with original resolution, (middle) low-resolution data based on the ML model and (bottom) low-resolution data based on a simulation with the same coarse resolution. Four different time steps are shown, and some key vortical structures are highlighted with yellow squares. Reprinted from Ref. [56], with permission of the publisher (United States National Academy of Sciences).

not satisfy the continuity equation (*i.e.* for incompressible flows, \mathbf{u}^* does not satisfy divergence free condition). The second step involves a correction to ensure that \mathbf{u}^* is divergence free, leading to the following Poisson equation:

$$\frac{\Delta t}{\rho} \nabla^2 p = -\nabla \cdot \mathbf{u}^*, \quad (1)$$

where Δt is the simulation time step, ρ is the fluid density, and p is the pressure. Solving this equation is typically the most computationally expensive step of the numerical solver. Therefore, devising alternative strategies to solve it more efficiently is an area of great promise. Ajuria *et al.* [2] proposed using a convolutional neural network (CNN) coupled with a CFD code to solve (1) in incompressible cases and tested it in a plume configuration. Their results indicate that it is possible to outperform the traditional Jacobi solver with good accuracy at low Richardson numbers Ri ; the Richardson number measures the ratio between the buoyancy and the shear in the flow. However, the accuracy degrades at higher Ri , motivating the authors to develop a hybrid CNN-CFD approach. Fully-convolutional neural networks were also used to solve the Poisson problem by decomposing the original problem into a homogeneous Poisson problem plus four inhomogeneous Laplace subproblems [92]. This decomposition resulted in errors below 10%, which motivates using this approach as a first guess in iterative algorithms, potentially reducing computational cost. These approaches may also be used to accelerate simulations of lower fidelity that rely on turbulence models, which will be discussed in the next section.

Numerical simulations can also be accelerated by decreasing the size of the computational domain needed to retain physical properties of the system. For example, artificially producing turbulent inflow conditions or far-field pressure gradients can significantly reduce the size of a computational domain. Fukami *et al.* [40] developed a time-dependent inflow generator for wall-bounded turbulence simulations using a convolutional autoencoder with an MLP. They tested their method in a turbulent channel flow at $Re_\tau = 180$, which is the friction Reynolds number based on channel half height and friction velocity, and they maintained turbulence for an interval long enough to obtain converged turbulence statistics. This is a promising research direction due to the fact that current inflow-generation methods show limitations in terms of generality of the inflow conditions, for instance at various flow geometries and Reynolds numbers. A second approach to reduce the computational domain in external flows is to devise a strategy to set the right pressure-gradient distribution without having to simulate the far field. This was addressed by Morita *et al.* [82] through Bayesian optimization based on Gaussian-process regression, achieving excellent results when imposing concrete pressure-gradient conditions on a turbulent boundary layer.

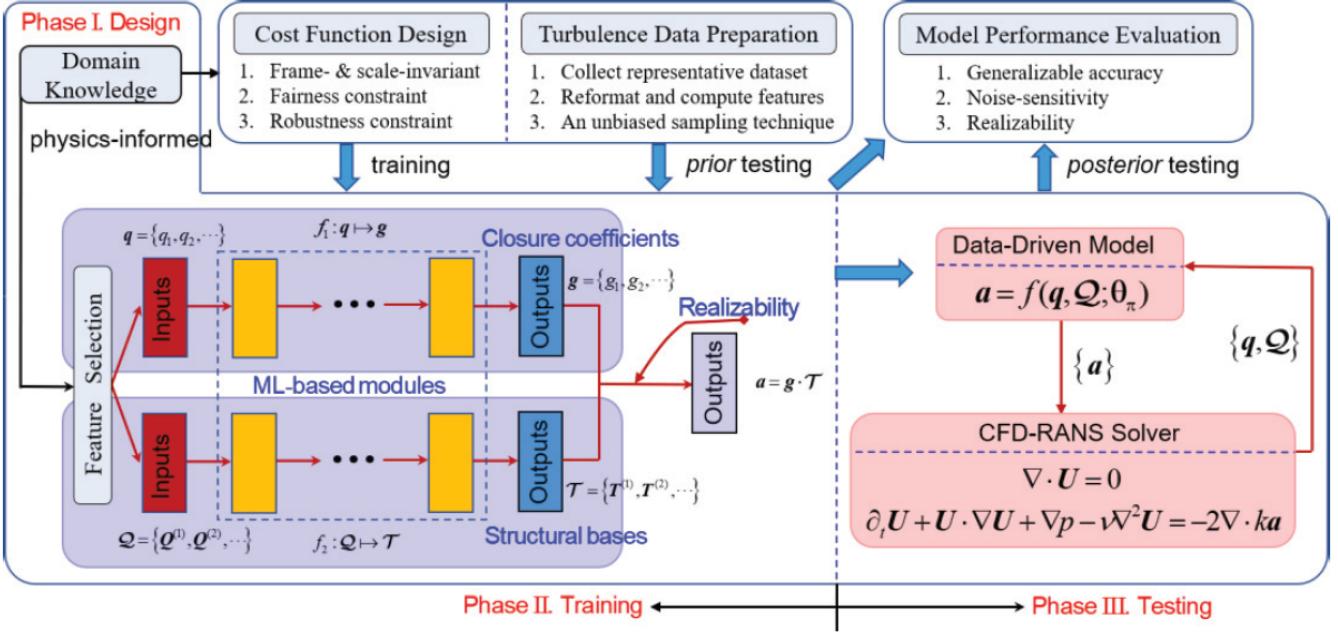


Figure 3: Schematic representation of the interpretable machine-learning framework proposed by Jiang *et al.* [50] for RANS modeling, which includes the three following phases: (i) design of the framework based on the domain knowledge, (ii) training strategy and (iii) performance assessment. Reprinted from Ref. [50], with permission of the publisher (AIP Publishing).

3 Turbulence modelling

DNS is impractical for many real-world applications due to the computational cost associated with resolving all scales for flows with high Reynolds numbers, together with difficulties arising from complex geometries. Industrial CFD typically relies on either Reynolds-averaged Navier–Stokes (RANS) models, where no turbulent scales are simulated, or coarsely resolved large-eddy simulations (LES), where only the largest turbulent scales are resolved and smaller ones are modelled. Here the term *model* refers to an *a-priori* assumption regarding the physics of a certain range of turbulent scales. In the following we discuss novel ML applications to RANS and LES modelling.

3.1 RANS modelling

Duraisamy *et al.* [31] provided a detailed review on ML applications to improve turbulence modelling, mostly in the context of RANS. They emphasized the importance of imposing physical constraints in the models and incorporating uncertainty-quantification (UQ) [34, 79, 101] alongside ML-based models. They note that when using DNS quantities to replace terms in the RANS closure, the predictions may be unsatisfactory [95]. This inadequacy is due to the strict assumptions associated with the RANS model, as well as the potential ill-conditioning of the RANS equations [130]. They propose to take advantage of novel data-driven methods, while also ensuring that uncertainties are identified and quantified. Another interesting review by Ahmed *et al.* [1] discussed both classical and emerging data-driven closure approaches, also connecting with ROMs.

Ling *et al.* [65] demonstrated the feasibility of deep learning for RANS modelling, as reviewed by Kutz [59]. They proposed a novel architecture, including a multiplicative layer with an invariant tensor basis, used to embed Galilean invariance in the predicted anisotropy tensor. Incorporating this invariance improves the performance of the network, which outperforms traditional RANS models based on linear [15] and nonlinear [28] eddy-viscosity models. They tested their models for turbulent duct flow and the flow over a wavy wall, which are challenging to predict with RANS models [72, 110] because of the presence of secondary flows [117]. Other ML-based approaches [124, 131] rely on physics-informed random forests to improve RANS models, with applications to cases with secondary flows and separation. Jiang *et al.* [50] recently developed an interpretable framework for RANS modelling based on a physics-informed residual network (PiResNet), shown in Fig. 3. Their approach relies on two modules to infer the structural and parametric representations of turbulence physics, and includes non-unique

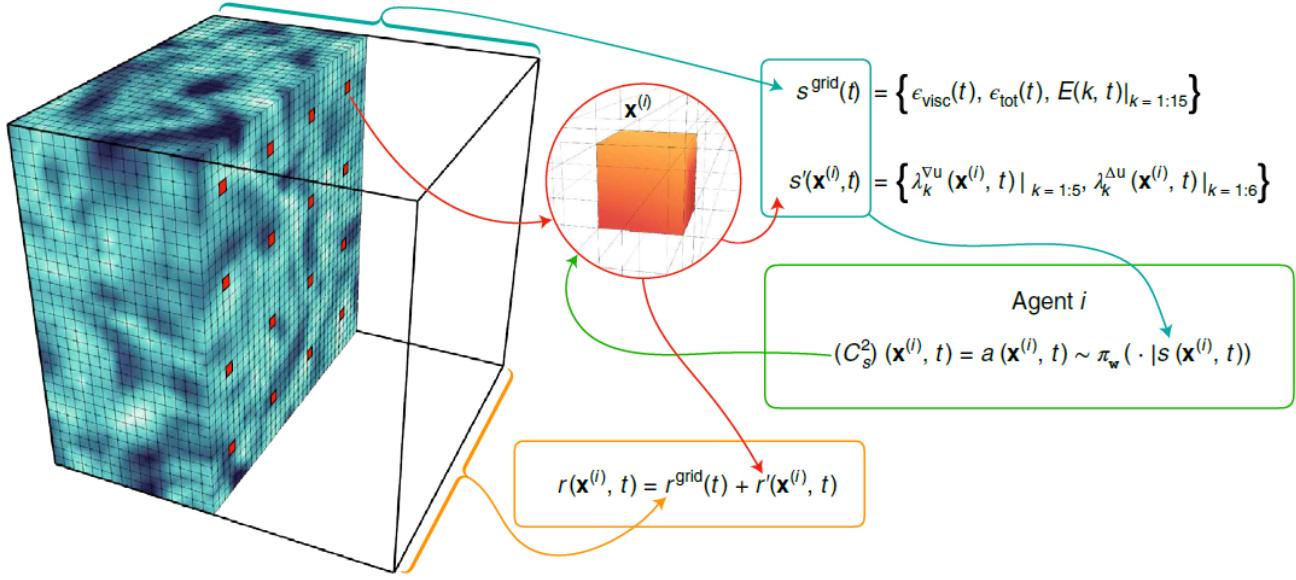


Figure 4: Representation of the RL-based SGS model proposed by Novati *et al.* [89]. The RL agents are located at the red blocks, and they are used to compute the so-called dissipation coefficient for each grid point. Note that the state of agent i at time t is given in terms of local variables (invariants of the velocity gradient and Hessian) and also global ones (energy spectrum, viscous dissipation rate and total dissipation rate). Reprinted from Ref. [89], with permission of the publisher (Springer Nature).

mappings, a realizability limiter, and noise-insensitivity constraints. Interpretable models are essential for engineering and physics, and the interpretability of their framework relies on its constrained model form [103]. Other interpretable RANS models were proposed by Weatheritt and Sandberg [127], using gene-expression programming (GEP), which is a branch of evolutionary computing [57]. GEP iteratively improves a population of candidate solutions by survival of the fittest, with the advantage of producing closed-form models. The Reynolds-stress anisotropy tensor was modelled by the same authors [128] and tested in RANS simulations of turbulent ducts.

Obiols-Sales *et al.* [90] developed a method to accelerate the convergence of RANS simulations based on the very popular Spalart–Allmaras (SA) turbulence model [109] using the CFD code OpenFOAM [129]. In essence, they combined iterations from the CFD solver and evaluation of a CNN model, obtaining convergence from 1.9 to 7.4 times faster than that of the CFD solver, both in laminar and turbulent flows. Multiphase flows, which consist of flows with two or more thermodynamic phases, are also industrially relevant. Gibou *et al.* [43] proposed different directions in which machine learning and deep learning can improve CFD of multiphase flows, in particular when it comes to enhancing the simulation speed. Ma *et al.* [70] used deep learning to predict the closure terms (i.e., gas flux and streaming stresses) in their two-fluid bubble flow, whereas Mi *et al.* [77] analyzed gas-liquid flows and employed neural networks to identify the different flow regimes.

3.2 LES modelling

Machine learning has also been used to develop subgrid-scale (SGS) models in the context of LES of turbulent flows. Beck *et al.* [9] used an artificial neural network based on local convolutional filters to predict the mapping between the flow in a coarse simulation and the closure terms, using a filtered DNS of decaying homogeneous isotropic turbulence. Lapeyre *et al.* [60] employed a similar approach, with a CNN architecture inspired by a U-net model, to predict the subgrid-scale wrinkling of the flame surface in premixed turbulent combustion; they obtained better results than classical algebraic models. Maulik *et al.* [75] employed a multilayer perceptron (MLP) to predict the SGS model in an LES using high-fidelity numerical data to train the model. They evaluated the performance of their method on Kraichnan turbulence [58], which is a classical two-dimensional decaying-turbulence test case. GEP has also been used for SGS modelling [100] in an LES of a Taylor–Green vortex, outperforming standard LES models. An interesting recent approach to LES modelling by Novati *et al.* [89] employed multi-agent reinforcement-learning (RL) to estimate the unresolved subgrid-scale physics. This unsupervised method exhibits favorable generalization properties across grid sizes and flow conditions, and the results are presented for isotropic turbulence. A schematic of this method is shown in Fig. 4. Several other studies have used neural networks for SGS modelling [42, 74, 122].

In certain applications, for instance those involving atmospheric boundary layers (ABLs), the Reynolds number is several orders of magnitude larger than those of most studies based on turbulence models or wind-tunnel experiments [48]. The mean flow in the inertial sublayer has been widely studied in the ABL community, and it is known that in neutral conditions it can be described by a logarithmic law [18]. The logarithmic description of the inertial sublayer led to the use of wall models, which replace the region very close to the wall with a model defining a surface shear stress matching the logarithmic behavior. This is the cornerstone of most atmospheric models, which avoid resolving the computationally expensive scales close to the wall. One example is the work by Giometto *et al.* [44], who studied a real urban geometry, adopting the LES model by Bou-Zeid *et al.* [14] and the Moeng model [81] for the wall boundary condition. Data-driven approaches have been developed to dynamically set this off-wall boundary condition based on information from the outer region. For instance, it is possible to exploit properties of the logarithmic layer and rescale the flow in the outer region to set the off-wall boundary condition in turbulent channels [35, 80]. This may also be accomplished via transfer functions in spectral space [104], convolutional neural networks [4], or modelling the temporal dynamics of the near-wall region via deep neural networks [78]. Another promising approach based on deep learning was tested in channel flow by Moriya *et al.* [83]. Defining off-wall boundary conditions with machine learning is a challenging yet promising area of research.

4 Reduced-order models

Machine-learning is also being used to develop reduced-order models (ROMs) in fluid dynamics. ROMs rely on the fact that even complex flows often exhibit a few dominant coherent structures [102, 114, 115] that may provide coarse, but valuable information about the flow. Thus, ROMs describe the evolution of these coherent structures, providing a lower-dimensional, lower-fidelity characterization of the fluid. In this way, ROMs provide a fast surrogate model for the more expensive CFD techniques described above, enabling optimization and control tasks that rely on many model iterations or fast model predictions. The cost of this efficiency is a loss of generality: ROMs are tailored to a specific flow configuration, providing massive acceleration but a limited range of applicability.

Developing a reduced-order model involves (1) finding a set of reduced *coordinates*, typically describing the amplitudes of important flow structures, and (2) identifying a differential-equation model (*i.e.*, a dynamical system) for how these amplitudes evolve in time. Both of these stages have seen incredible recent advances with machine learning. One common ROM technique involves learning a low-dimensional coordinate system with the proper orthogonal decomposition (POD) [68, 114] and then obtaining a dynamical system for the flow system restricted to this subspace by Galerkin projection of the Navier-Stokes equations onto these modes. Although the POD step is data driven, working equally well for experiments and simulations, Galerkin projection requires a working numerical implementation of the governing equations; moreover, it is often *intrusive*, involving custom modifications to the numerical solver. The related dynamic-mode decomposition (DMD) [107] is a purely data-driven procedure that identifies a low-dimensional subspace and a linear model for how the flow evolves in this subspace. Here, we will review a number of recent developments to extend these approaches with machine learning.

The first broad opportunity to incorporate machine learning into ROMs is in learning an improved coordinate system in which to represent the reduced dynamics. POD [68, 114] provides an orthogonal set of modes that may be thought of as a data-driven generalization of Fourier modes, which are tailored to a specific problem. POD is closely related to principal-component analysis (PCA) and the singular-value decomposition (SVD) [19], which are two core dimensionality-reduction techniques used in data-driven modeling. These approaches provide linear subspaces to approximate data, even though it is known that many systems evolve on a nonlinear manifold. Deep learning provides a powerful approach to generalize the POD/PCA/SVD dimensionality reduction from learning a linear subspace to learning coordinates on a curved manifold. Specifically, these coordinates may be learned using a neural network *autoencoder*, which has an input and output the size of the high-dimensional fluid state and a constriction or bottleneck in the middle that reduces to a low-dimensional latent variable. The map from the high-dimensional state \mathbf{x} to the latent state \mathbf{z} is called the *encoder* and the map back from the latent state to an estimate of the high-dimensional state $\hat{\mathbf{x}}$ is the *decoder*. The autoencoder loss function is $\|\hat{\mathbf{x}} - \mathbf{x}\|_2^2$. When the encoder and decoder consist of a single layer and all nodes have identity activation functions, then the optimal solution to this network will be closely related to POD [5]. However, this shallow linear autoencoder may be generalized to a deep nonlinear autoencoder with multiple encoding and decoding layers and nonlinear activation functions for the nodes. In this way, a deep autoencoder learns nonlinear manifold coordinates that may considerably improve the compression in the latent space, with increasing applications in fluid mechanics [32, 84]. This concept is illustrated in Fig. 5 for the simple flow past a cylinder, where it is known that the energetic coherent structures evolve on a parabolic submanifold in the POD subspace [86]. Lee and Carlberg [61] recently showed that deep convolutional autoencoders may be used to greatly improve the performance of classical ROM techniques based on linear subspaces [12, 24].

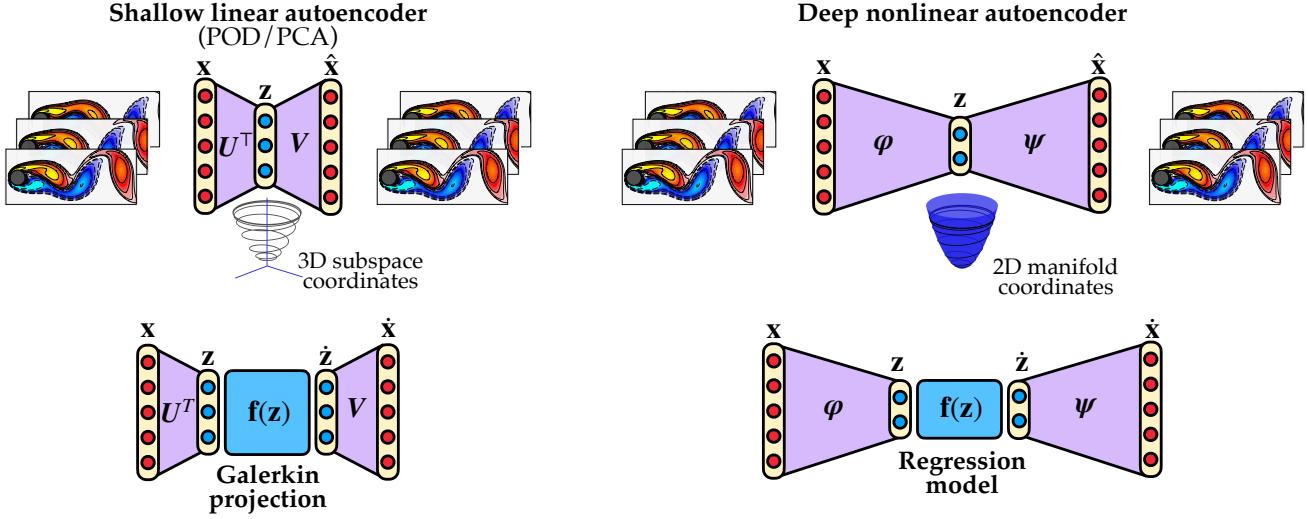


Figure 5: Schematic of classical and modern dimensionality reduction and model identification as a neural-network autoencoder. Classic POD/PCA may be viewed as a shallow autoencoder with a single encoder \mathbf{U}^\top and decoder \mathbf{V} layers, together with linear activation units (top left). A deep, multi-level autoencoder with multilayer encoder φ and decoder ψ , as well as nonlinear activation functions (top right) provides enhanced nonlinear coordinates on a manifold. Similarly, the classic Galerkin-projection model (bottom left) may be replaced with more generic machine-learning regressions (bottom right), such as LSTMs, reservoir networks or SINDy models to represent the nonlinear dynamical system $\dot{\mathbf{z}} = \mathbf{f}(\mathbf{z})$.

Once an appropriate coordinate system is established, there are many machine-learning approaches to model the dynamics in these coordinates. Several neural networks are capable of learning nonlinear dynamics, including the LSTM network [111, 121] and echo-state networks [93], which are a form of reservoir computing. Beyond neural networks, there are alternative regression techniques to learn effective dynamical-systems models. Cluster reduced-order modeling (CROM) [51] is a simple and powerful unsupervised-learning approach that decomposes a time series into a few representative clusters, and then models the transition-state probability between clusters. The operator-inference approach is closely related to Galerkin projection, where a neighboring operator is learned from data [13, 94, 96]. The sparse identification of nonlinear dynamics (SINDy) [20] procedure learns a minimalistic model by fitting the observed dynamics to the fewest terms in a library of candidate functions that might describe the dynamics, resulting in models that are interpretable and balance accuracy and efficiency. SINDy has been used to model a range of fluids [22, 23, 29, 30, 45, 66, 67], including laminar and turbulent wake flows, convective flows, and shear flows. SINDy models have also been used for RANS closure models [10, 11, 106].

The modeling approaches above may be combined with a deep autoencoder to uncover a low-dimensional latent space, as in the SINDy-autoencoder [25]. DMD has also been extended to nonlinear coordinate embeddings through the use of deep-autoencoder networks [69, 71, 91, 116, 132]. In all of these architectures, there is a tremendous opportunity to embed partial knowledge of the physics, such as conservation laws [67], symmetries [45], and invariances [38, 125, 126]. It may also be possible to directly impose stability in the learning pipeline [36, 52, 67].

There are a number of challenges and opportunities that face the integration of machine learning with reduced-order modeling. Machine learning provides improved pattern extraction over traditional techniques, and it also may be able to compensate for basis imperfections in ways that classical Galerkin projection cannot. The ultimate goal for machine-learning ROMs is to develop models that have improved accuracy and efficiency, better generalizability to new initial and boundary conditions, flow configurations, varying parameters, as well as improved model interpretability, ideally with less intrusive methods and less data. Enforcing partially-known physics, such as symmetries and other invariances, along with sparsity, is expected to be critical in these efforts. It is also important to continue integrating these efforts with the downstream applications of control and optimization. Finally, many applications of fluid dynamics are involve safety-critical systems, and therefore certifiable models are essential.

5 Emerging possibilities and outlook

In this contribution we have provided our perspectives on the potential of ML to advance the capabilities of CFD, focusing on three main areas: accelerating simulations, enhancing turbulence models, and improving reduced-order models. In each of these areas, incorporating partial physical knowledge of the fluid dynamics into the machine learning process improves performance.

There are several emerging areas of ML that are promising for CFD. One area is non-intrusive sensing, *i.e.*, the possibility of performing flow predictions based on *e.g.* information at the wall. This task, which has important implications for closed-loop flow control, has been carried out via CNNs in turbulent channels [46]. In connection to this work, there are a number of studies documenting the possibility of performing super-resolution predictions (*e.g.* when limited flow information is available) in wall-bounded turbulence using CNNs, autoencoders and generative-adversarial networks (GANs) [39, 41, 47, 54]. Another promising direction is the imposition of constraints based on physical invariances and symmetries on the ML model, which has been used for SGS modelling [126], ROMs [67], and for geophysical flows [38].

Physics-informed neural networks (PINNs) are another family of methods that are becoming widely adopted for scientific computing, more broadly. This type of network, introduced by Raissi *et al.* [97], uses deep learning to solve PDEs, exploiting the concept of automatic differentiation used in the back-propagation algorithm to calculate partial derivatives and form the equations, which are enforced through a loss function. In certain cases they can solve PDEs more efficiently than traditional numerical methods. This framework also shows promise for biomedical applications, in particular after the recent work by Raissi *et al.* [98], in which the concentration field of a tracer is used as an input to accurately predict the instantaneous velocity fields by minimizing the residual of the Navier-Stokes equations. PINNs have also been used for turbulence modelling [33] and for accelerating traditional solvers, *e.g.* solving the Poisson equation more efficiently [73].

We must also look to grand challenges in CFD that necessitate new methods in ML. One motivating challenging in CFD is to perform accurate coarse-resolution simulations in unforced three-dimensional wall-bounded turbulent flows. The production of turbulent kinetic energy (TKE) in these flows takes place close to the wall [55], and therefore using coarse meshes may significantly distort TKE production; at very high Reynolds numbers, outer-layer production also becomes relevant. In these flows of high technological importance, the TKE production sustains the turbulent fluctuations, and therefore a correction between coarse and fine resolutions may not be sufficient to obtain accurate results. These challenges will require novel techniques to progress the field.

Despite the caveats discussed in §1, we believe that the trend of advancing CFD with ML will continue in the future. This progress will continue to be driven by an increasing availability of high-quality data, high-performance computing, and a better understanding and facility with these emerging techniques. Improved adoption of reproducible research standards [7, 76] is also a necessary step. Given the critical importance of data when developing ML modes, we advocate that the community continue to establish proper benchmark systems and best practices for open-source data and software in order to harness the full potential of ML to improve CFD.

Acknowledgements

RV acknowledges the financial support from the Swedish Research Council (VR). SLB acknowledges funding support from the Army Research Office (ARO W911NF-19-1-0045; program manager Dr. Matthew Munson).

References

- [1] S. E. Ahmed, S. Pawar, O. San, A. Rasheed, T. Iliescu, and B. R. Noack. On closures for reduced order models—a spectrum of first-principle to machine-learned avenues. *Physics of Fluids*, 33(9):091301, 2021.
- [2] E. Ajuria, A. Alguacil, M. Bauerheim, A. Misdariis, B. Cuenot, and E. Benazera. Towards a hybrid computational strategy based on deep learning for incompressible flows. *AIAA AVIATION Forum, June 15–19*, pages 1–17, 2020.
- [3] C. Aloy Torás, P. Mimica, and M. Martínez-Sober. Towards detecting structures in computational astrophysics plasma simulations: using machine learning for shock front classification. In *Artificial Intelligence Research and Development*. Z. Falomir *et al.* (Eds.), pages 59–63, 2018.
- [4] G. B. Arivazhagan, L. Guastoni, A. Güemes, A. Ianiro, S. Discetti, P. Schlatter, H. Azizpour, and R. Vinuesa. Predicting the near-wall region of turbulence through convolutional neural networks. *Proc. 13th ERCOF-TAC Symp. on Engineering Turbulence Modelling and Measurements (ETMM13), Rhodes, Greece, September 16–17*. Preprint arXiv:2107.07340, 2021.

- [5] P. Baldi and K. Hornik. Neural networks and principal component analysis: Learning from examples without local minima. *Neural Networks*, 2(1):53–58, 1989.
- [6] Y. Bar-Sinai, S. Hoyer, J. Hickey, and M. P. Brenner. Learning data-driven discretizations for partial differential equations. *Proceedings of the National Academy of Sciences*, 116(31):15344–15349, 2019.
- [7] L. A. Barba. The hard road to reproducibility. *Science*, 354(6308):142–142, 2016.
- [8] P. Bauer, A. Thorpe, and G. Brunet. The quiet revolution of numerical weather prediction. *Nature*, 525:47–55, 2015.
- [9] A. D. Beck, D. G. Flad, and C.-D. Munz. Deep neural networks for data-driven LES closure models. *Journal of Computational Physics*, 398:108910, 2019.
- [10] S. Beetham and J. Capecelatro. Formulating turbulence closures using sparse regression with embedded form invariance. *Physical Review Fluids*, 5(8):084611, 2020.
- [11] S. Beetham, R. O. Fox, and J. Capecelatro. Sparse identification of multiphase turbulence closures for coupled fluid-particle flows. *Journal of Fluid Mechanics*, 914, 2021.
- [12] P. Benner, S. Gugercin, and K. Willcox. A survey of projection-based model reduction methods for parametric dynamical systems. *SIAM Review*, 57(4):483–531, 2015.
- [13] P. Benner, P. Goyal, B. Kramer, B. Peherstorfer, and K. Willcox. Operator inference for non-intrusive model reduction of systems with non-polynomial nonlinear terms. *Computer Methods in Applied Mechanics and Engineering*, 372:113433, 2020.
- [14] E. Bou-Zeid, C. Meneveau, and M. Parlange. A scale-dependent Lagrangian dynamic model for large eddy simulation of complex turbulent flows. *Physics of Fluids*, 17:025105, 2005.
- [15] J. V. Boussinesq. *Théorie analytique de la chaleur: mise en harmonie avec la thermodynamique et avec la théorie mécanique de la lumière T. 2, Refroidissement et échauffement par rayonnement conductibilité des tiges, lames et masses cristallines courants de convection théorie mécanique de la lumière*. Gauthier-Villars, 1923.
- [16] M. Brenner, J. Eldredge, and J. Freund. Perspective on machine learning for advancing fluid mechanics. *Physical Review Fluids*, 4(10):100501, 2019.
- [17] R. Bridson. Fluid simulation. *A. K. Peters, Ltd., Natick, MA, USA*, 2008.
- [18] R. E. Britter and S. R. Hanna. Flow and dispersion in urban areas. *Annual Review of Fluid Mechanics*, 35: 469–496, 2003.
- [19] S. L. Brunton and J. N. Kutz. *Data-Driven Science and Engineering: Machine Learning, Dynamical Systems, and Control*. Cambridge University Press, 2019.
- [20] S. L. Brunton, J. L. Proctor, and J. N. Kutz. Discovering governing equations from data by sparse identification of nonlinear dynamical systems. *Proceedings of the National Academy of Sciences*, 113(15):3932–3937, 2016.
- [21] S. L. Brunton, B. R. Noack, and P. Koumoutsakos. Machine learning for fluid mechanics. *Annual Review of Fluid Mechanics*, 52:477–508, 2020.
- [22] J. L. Callaham, S. L. Brunton, and J.-C. Loiseau. On the role of nonlinear correlations in reduced-order modeling. *arXiv preprint arXiv:2106.02409*, 2021.
- [23] J. L. Callaham, G. Rigas, J.-C. Loiseau, and S. L. Brunton. An empirical mean-field model of symmetry-breaking in a turbulent wake. *arXiv preprint arXiv:2105.13990*, 2021.
- [24] K. Carlberg, M. Barone, and H. Antil. Galerkin v. least-squares petrov–galerkin projection in nonlinear model reduction. *Journal of Computational Physics*, 330:693–734, 2017.
- [25] K. Champion, B. Lusch, J. N. Kutz, and S. L. Brunton. Data-driven discovery of coordinates and governing equations. *Proceedings of the National Academy of Sciences*, 116(45):22445–22451, 2019.
- [26] G. J. Chandler and R. R. Kerswell. Invariant recurrent solutions embedded in a turbulent two-dimensional Kolmogorov flow. *Journal of Fluid Mechanics*, 722:554–595, 2013.
- [27] H. Choi and P. Moin. Grid-point requirements for large eddy simulation: Chapman’s estimates revisited. *Physics of Fluids*, 24:011702, 2012.
- [28] T. J. Craft, B. E. Launder, and K. Suga. Development and application of a cubic eddy-viscosity model of turbulence. *International Journal of Heat and Fluid Flow*, 17:108–115, 1996.
- [29] N. Deng, B. R. Noack, M. Morzynski, and L. R. Pastur. Low-order model for successive bifurcations of the fluidic pinball. *Journal of Fluid Mechanics*, 884(A37), 2020.
- [30] N. Deng, B. R. Noack, M. Morzyński, and L. R. Pastur. Galerkin force model for transient and post-transient dynamics of the fluidic pinball. *Journal of Fluid Mechanics*, 918, 2021.
- [31] K. Duraisamy, G. Iaccarino, and H. Xiao. Turbulence modeling in the age of data. *Annual Review of Fluid Mechanics*, 51:357–377, 2019.
- [32] H. Eivazi, S. Le Clainche, S. Hoyas, and R. Vinuesa. Towards extraction of orthogonal and parsimonious non-linear modes from turbulent flows. *Preprint arXiv:2109.01514*, 2021.
- [33] H. Eivazi, M. Tahani, P. Schlatter, and R. Vinuesa. Physics-informed neural networks for solving Reynolds-averaged Navier–Stokes equations. *Proc. 13th ERCOFTAC Symp. on Engineering Turbulence Modelling and Measurements (ETMM13), Rhodes, Greece, September 16–17. Preprint arXiv:2107.10711*, 2021.

- [34] M. Emory, J. Larsson, and G. Iaccarino. Modeling of structural uncertainties in Reynolds-averaged Navier-Stokes closures. *Physics of Fluids*, 25:110822, 2013.
- [35] M. P. Encinar, R. García-Mayoral, and J. Jiménez. Scaling of velocity fluctuations in off-wall boundary conditions for turbulent flows. *Journal of Physics: Conference Series*, 506:012002, 2014.
- [36] N. B. Erichson, M. Muehlebach, and M. W. Mahoney. Physics-informed autoencoders for lyapunov-stable fluid flow prediction. *arXiv preprint arXiv:1905.10866*, 2019.
- [37] R. Eymard, T. Gallouët, and R. Herbin. Finite volume methods. *Handbook of Numerical Analysis*, 7:713–1018, 2000.
- [38] H. Frezat, G. Balarac, J. Le Sommer, R. Fablet, and R. Lguensat. Physical invariance in neural networks for subgrid-scale scalar flux modeling. *Physical Review Fluids*, 6(2):024607, 2021.
- [39] K. Fukami, K. Fukagata, and K. Taira. Super-resolution reconstruction of turbulent flows with machine learning. *Journal of Fluid Mechanics*, 870:106–120, 2019.
- [40] K. Fukami, Y. Nabae, K. Kawai, and K. Fukagata. Synthetic turbulent inflow generator using machine learning. *Physical Review Fluids*, 4:064603, 2019.
- [41] K. Fukami, T. Nakamura, and K. Fukagata. Convolutional neural network based hierarchical autoencoder for nonlinear mode decomposition of fluid field data. *Physics of Fluids*, 32:095110, 2020.
- [42] M. Gamahara and Y. Hattori. Searching for turbulence models by artificial neural network. *Physical Review Fluids*, 2:054604, 2017.
- [43] F. Gibou, D. Hyde, and R. Fedkiw. Sharp interface approaches and deep learning techniques for multiphase flows. *Journal of Computational Physics*, 380:442–463, 2019.
- [44] M. G. Giometto, A. Christen, C. Meneveau, J. Fang, M. Krafczyk, and M. B. Parlange. Spatial characteristics of roughness sublayer mean flow and turbulence over a realistic urban surface. *Boundary-Layer Meteorology*, 160:425–452, 2016.
- [45] Y. Guan, S. L. Brunton, and I. Novoselov. Sparse nonlinear models of chaotic electroconvection. *Royal Society Open Science*, 8(8):202367, 2021.
- [46] L. Guastoni, A. Güemes, A. Ianiro, S. Discetti, P. Schlatter, H. Azizpour, and R. Vinuesa. Convolutional-network models to predict wall-bounded turbulence from wall quantities. *Preprint arXiv:2006.12483*, 2020.
- [47] A. Güemes, S. Discetti, A. Ianiro, B. Sirmacek, H. Azizpour, and R. Vinuesa. From coarse wall measurements to turbulent velocity fields through deep learning. *Physics of Fluids*, 33:075121, 2021.
- [48] N. Hutchins, K. Chauhan, I. Marusic, J. Monty, and J. Klewicki. Towards reconciling the large-scale structure of turbulent boundary layers in the atmosphere and laboratory. *Boundary-Layer Meteorology*, 145:273–306, 2012.
- [49] J. Jeon and S. J. Kim. FVM Network to reduce computational cost of CFD simulation. *Preprint arXiv:2105.03332*, 2021.
- [50] C. Jiang, R. Vinuesa, R. Chen, J. Mi, S. Laima, and H. Li. An interpretable framework of data-driven turbulence modeling using deep neural networks. *Physics of Fluids*, 33:055133, 2021.
- [51] E. Kaiser, B. R. Noack, L. Cordier, A. Spohn, M. Segond, M. Abel, G. Daviller, J. Osth, S. Krajnovic, and R. K. Niven. Cluster-based reduced-order modelling of a mixing layer. *J. Fluid Mech.*, 754:365–414, 2014.
- [52] A. A. Kaptanoglu, J. L. Callahan, C. J. Hansen, A. Aravkin, and S. L. Brunton. Promoting global stability in data-driven models of quadratic nonlinear dynamics. *Physical Review Fluids*, 6(094401), 2021.
- [53] G. E. Karniadakis, I. G. Kevrekidis, L. Lu, P. Perdikaris, S. Wang, and L. Yang. Physics-informed machine learning. *Nature Reviews Physics*, 3(6):422–440, 2021.
- [54] H. Kim, J. Kim, S. Won, and C. Lee. Unsupervised deep learning for super-resolution reconstruction of turbulence. *Journal of Fluid Mechanics*, 910:A29, 2021.
- [55] J. Kim, P. Moin, and R. Moser. Turbulence statistics in fully developed channel flow at low Reynolds number. *Journal of Fluid Mechanics*, 177:133–166, 1987.
- [56] D. Kochkov, J. A. Smith, A. Alieva, Q. Wang, M. P. Brenner, and S. Hoyer. Machine learning-accelerated computational fluid dynamics. *Proceedings of the National Academy of Sciences*, 118:e2101784118, 2021.
- [57] J. R. Koza. Genetic Programming: On the Programming of Computers by Means of Natural Selection. *MIT Press*, 1992.
- [58] R. H. Kraichnan. Inertial ranges in two-dimensional turbulence. *Physics of Fluids*, 10:1417–1423, 1967.
- [59] J. N. Kutz. Deep learning in fluid dynamics. *Journal of Fluid Mechanics*, 814:1–4, 2017.
- [60] C. J. Lapeyre, A. Misdariis, N. Cazard, D. Veynante, and T. Poinsot. Training convolutional neural networks to estimate turbulent sub-grid scale reaction rates. *Combustion and Flame*, 203:255, 2019.
- [61] K. Lee and K. T. Carlberg. Model reduction of dynamical systems on nonlinear manifolds using deep convolutional autoencoders. *Journal of Computational Physics*, 404:108973, 2020.
- [62] Z. Li, N. Kovachki, K. Azizzadenesheli, B. Liu, K. Bhattacharya, A. Stuart, and A. Anandkumar. Fourier neural operator for parametric partial differential equations. *arXiv preprint arXiv:2010.08895*, 2020.
- [63] Z. Li, N. Kovachki, K. Azizzadenesheli, B. Liu, K. Bhattacharya, A. Stuart, and A. Anandkumar. Multipole

- graph neural operator for parametric partial differential equations. *arXiv preprint arXiv:2006.09535*, 2020.
- [64] Z. Li, N. Kovachki, K. Azizzadenesheli, B. Liu, K. Bhattacharya, A. Stuart, and A. Anandkumar. Neural operator: Graph kernel network for partial differential equations. *arXiv preprint arXiv:2003.03485*, 2020.
- [65] J. Ling, A. Kurzawski, and J. Templeton. Reynolds averaged turbulence modelling using deep neural networks with embedded invariance. *Journal of Fluid Mechanics*, 807:155–166, 2016.
- [66] J.-C. Loiseau. Data-driven modeling of the chaotic thermal convection in an annular thermosyphon. *Theoretical and Computational Fluid Dynamics*, 34(4):339–365, 2020.
- [67] J.-C. Loiseau and S. L. Brunton. Constrained sparse Galerkin regression. *Journal of Fluid Mechanics*, 838:42–67, 2018.
- [68] J. L. Lumley. The structure of inhomogeneous turbulence. *Atmospheric turbulence and wave propagation, A. M. Yaglom and V. I. Tatarski (eds). Nauka, Moscow*, pages 166–178, 1967.
- [69] B. Lusch, J. N. Kutz, and S. L. Brunton. Deep learning for universal linear embeddings of nonlinear dynamics. *Nature Communications*, 9(1):4950, 2018.
- [70] M. Ma, J. Lu, and G. Tryggvason. Using statistical learning to close two-fluid multiphase flow equations for a simple bubbly system. *Physics of Fluids*, 27:092101, 2015.
- [71] A. Mardt, L. Pasquali, H. Wu, and F. Noé. VAMPnets: Deep learning of molecular kinetics. *Nature Communications*, 9(5), 2018.
- [72] O. Marin, R. Vinuesa, A. V. Obabko, and P. Schlatter. Characterization of the secondary flow in hexagonal ducts. *Physics of Fluids*, 28:125101, 2016.
- [73] S. Markidis. The old and the new: can physics-informed deep-learning replace traditional linear solvers? *Preprint arXiv:2103.09655v2*, 2021.
- [74] R. Maulik and O. San. A neural network approach for the blind deconvolution of turbulent flows. *Journal of Fluid Mechanics*, 831:151–181, 2017.
- [75] R. Maulik, O. San, A. Rasheed, and P. Vedula. Subgrid modelling for two-dimensional turbulence using neural networks. *Journal of Fluid Mechanics*, 858:122–144, 2019.
- [76] O. Mesnard and L. A. Barba. Reproducible and replicable computational fluid dynamics: it’s harder than you think. *Computing in Science & Engineering*, 19(4):44–55, 2017.
- [77] Y. Mi, M. Ishii, and L. H. Tsoukalas. Flow regime identification methodology with neural networks and two-phase flow models. *Nuclear Engineering and Design*, 204:87–100, 2001.
- [78] M. Milano and P. Koumoutsakos. Neural network modeling for near wall turbulent flow. *Journal of Computational Physics*, 182:1–26, 2002.
- [79] A. A. Mishra and G. Iaccarino. Uncertainty estimation for Reynolds-averaged Navier–Stokes predictions of high-speed aircraft nozzle jets. *AIAA Journal*, 55:3999–4004, 2017.
- [80] Y. Mizuno and J. Jiménez. Wall turbulence without walls. *Journal of Fluid Mechanics*, 723:429–455, 2013.
- [81] C. Moeng. A large-eddy-simulation model for the study of planetary boundary-layer turbulence. *Journal of Atmospheric Sciences*, 41:2052–2062, 1984.
- [82] Y. Morita, S. Rezaeiravesh, N. Tabatabaei, R. Vinuesa, K. Fukagata, and P. Schlatter. Applying Bayesian optimization with Gaussian-process regression to Computational Fluid Dynamics problems. *Preprint arXiv:2101.09985*, 2021.
- [83] N. Moriya, K. Fukami, Y. Nabae, M. Morimoto, T. Nakamura, and K. Fukagata. Inserting machine-learned virtual wall velocity for large-eddy simulation of turbulent channel flows. *Preprint arXiv:2106.09271*, 2021.
- [84] T. Murata, K. Fukami, and K. Fukagata. Nonlinear mode decomposition with convolutional neural networks for fluid dynamics. *Journal of Fluid Mechanics*, 882:A13, 2020.
- [85] S. A. Niederer, M. S. Sacks, M. Girolami, and K. Willcox. Scaling digital twins from the artisanal to the industrial. *Nature Computational Science*, 1(5):313–320, 2021.
- [86] B. R. Noack, K. Afanasiev, M. Morzynski, G. Tadmor, and F. Thiele. A hierarchy of low-dimensional models for the transient and post-transient cylinder wake. *Journal of Fluid Mechanics*, 497:335–363, 2003.
- [87] F. Noé, S. Olsson, J. Köhler, and H. Wu. Boltzmann generators: Sampling equilibrium states of many-body systems with deep learning. *Science*, 365(6457):eaaw1147, 2019.
- [88] F. Noé, A. Tkatchenko, K.-R. Müller, and C. Clementi. Machine learning for molecular simulation. *Annual review of physical chemistry*, 71:361–390, 2020.
- [89] G. Novati, H. L. de Laroussilhe, and P. Koumoutsakos. Automating turbulence modelling by multi-agent reinforcement learning. *Nature Machine Intelligence*, 3:87–96, 2021.
- [90] O. Obiols-Sales, A. Vishnu, N. Malaya, and A. Chandramowlishwaran. CFDNet: a deep learning-based accelerator for fluid simulations. *Preprint arXiv:2005.04485*, 2020.
- [91] S. E. Otto and C. W. Rowley. Linearly-recurrent autoencoder networks for learning dynamics. *SIAM Journal on Applied Dynamical Systems*, 18(1):558–593, 2019.
- [92] A. Özbay, A. Hamzehloo, S. Laizet, P. Tzirakis, G. Rizos, and B. Schuller. Poisson CNN: Convolutional neural networks for the solution of the Poisson equation on a Cartesian mesh. *Data-Centric Engineering*, 2:E6, 2021.

- [93] J. Pathak, B. Hunt, M. Girvan, Z. Lu, and E. Ott. Model-free prediction of large spatiotemporally chaotic systems from data: a reservoir computing approach. *Physical Review Letters*, 120(2):024102, 2018.
- [94] B. Peherstorfer and K. Willcox. Data-driven operator inference for nonintrusive projection-based model reduction. *Computer Methods in Applied Mechanics and Engineering*, 306:196–215, 2016.
- [95] S. Poroseva, F. J. D. Colmenares, and S. Murman. On the accuracy of RANS simulations with DNS data. *Physics of Fluids*, 28:115102, 2016.
- [96] E. Qian, B. Kramer, B. Peherstorfer, and K. Willcox. Lift & learn: Physics-informed machine learning for large-scale nonlinear dynamical systems. *Physica D: Nonlinear Phenomena*, 406:132401, 2020.
- [97] 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. *Journal of Computational Physics*, 378:686–707, 2019.
- [98] M. Raissi, A. Yazdani, and G. E. Karniadakis. Hidden fluid mechanics: Learning velocity and pressure fields from flow visualizations. *Science*, 367:1026–1030, 2020.
- [99] B. Recht. A tour of reinforcement learning: The view from continuous control. *Annual Review of Control, Robotics, and Autonomous Systems*, 2:253–279, 2019.
- [100] M. Reissmann, J. Hasslbergerb, R. D. Sandberg, and M. Klein. Application of gene expression programming to a-posteriori LES modeling of a Taylor Green vortex. *Journal of Computational Physics*, 424:109859, 2021.
- [101] S. Rezaeiravesh, R. Vinuesa, and P. Schlatter. On numerical uncertainties in scale-resolving simulations of canonical wall turbulence. *Computers and Fluids*, 227:105024, 2021.
- [102] C. W. Rowley and S. T. Dawson. Model reduction for flow analysis and control. *Annual Review of Fluid Mechanics*, 49:387–417, 2017.
- [103] C. Rudin. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence*, 1:206–215, 2019.
- [104] K. Sasaki, R. Vinuesa, A. V. G. Cavalieri, P. Schlatter, and D. S. Henningson. Transfer functions for flow predictions in wall-bounded turbulence. *Journal of Fluid Mechanics*, 864:708–745, 2019.
- [105] F. Schenk, M. Välimäki, F. Muschitiello, L. Tarasov, M. Heikkilä, S. Björck, J. Brandefelt, A. V. Johansson, J. O. Näslund, and B. Wohlfarth. Warm summers during the Younger Dryas cold reversal. *Nature Communications*, 9:1634, 2018.
- [106] M. Schmelzer, R. P. Dwight, and P. Cinnella. Discovery of algebraic reynolds-stress models using sparse symbolic regression. *Flow, Turbulence and Combustion*, 104(2):579–603, 2020.
- [107] P. J. Schmid. Dynamic mode decomposition of numerical and experimental data. *Journal of Fluid Mechanics*, 656:5–28, Aug. 2010.
- [108] T. Shan, W. Tang, X. Dang, M. Li, F. Yang, S. Xu, and J. Wu. Study on a Poisson’s equation solver based on deep learning technique. *2017 IEEE Electrical Design of Advanced Packaging and Systems Symposium (EDAPS)*, pages 1–3, 2017.
- [109] P. Spalart and S. Allmaras. A one-equation turbulence model for aerodynamic flows. *30th Aerospace Sciences Meeting and Exhibit, AIAA Paper 1992-0439*, 1992.
- [110] P. R. Spalart. Strategies for turbulence modelling and simulations. *International Journal of Heat and Fluid Flow*, 21:252–263, 2000.
- [111] P. A. Srinivasan, L. Guastoni, H. Azizpour, P. Schlatter, and R. Vinuesa. Predictions of turbulent shear flows using deep neural networks. *Physical Review Fluids*, 4:054603, 2019.
- [112] B. Stevens and T. Colonius. Enhancement of shock-capturing methods via machine learning. *Theoretical and Computational Fluid Dynamics*, 34:483–496, 2020.
- [113] B. Stevens and T. Colonius. Finitenet: A fully convolutional LSTM network architecture for time-dependent partial differential equations. *arXiv preprint arXiv:2002.03014*, 2020.
- [114] K. Taira, S. L. Brunton, S. Dawson, C. W. Rowley, T. Colonius, B. J. McKeon, O. T. Schmidt, S. Gordeyev, V. Theofilis, and L. S. Ukeiley. Modal analysis of fluid flows: An overview. *AIAA Journal*, 55(12):4013–4041, 2017.
- [115] K. Taira, M. S. Hemati, S. L. Brunton, Y. Sun, K. Duraisamy, S. Bagheri, S. Dawson, and C.-A. Yeh. Modal analysis of fluid flows: Applications and outlook. *AIAA Journal*, 58(3):998–1022, 2020.
- [116] N. Takeishi, Y. Kawahara, and T. Yairi. Learning koopman invariant subspaces for dynamic mode decomposition. In *Advances in Neural Information Processing Systems*, pages 1130–1140, 2017.
- [117] A. Vidal, H. M. Nagib, P. Schlatter, and R. Vinuesa. Secondary flow in spanwise-periodic in-phase sinusoidal channels. *Journal of Fluid Mechanics*, 851:288–316, 2018.
- [118] R. Vinuesa, S. M. Hosseini, A. Hanifi, D. S. Henningson, and P. Schlatter. Pressure-gradient turbulent boundary layers developing around a wing section. *Flow Turbulence and Combustion*, 99:613–641, 2017.
- [119] R. Vinuesa, P. S. Negi, M. Atzori, A. Hanifi, D. S. Henningson, and P. Schlatter. Turbulent boundary layers around wing sections up to $Re_c = 1,000,000$. *International Journal of Heat and Fluid Flow*, 72:86–99, 2018.
- [120] R. Vinuesa, H. Azizpour, I. Leite, M. Balaam, V. Dignum, S. Domisch, A. Felländer, S. D. Langhans,

- M. Tegmark, and F. Fuso Nerini. The role of artificial intelligence in achieving the Sustainable Development Goals. *Nature Communications*, 11:233, 2020.
- [121] P. R. Vlachas, W. Byeon, Z. Y. Wan, T. P. Sapsis, and P. Koumoutsakos. Data-driven forecasting of high-dimensional chaotic systems with long short-term memory networks. *Proceedings of the Royal Society A*, 474:20170844, 2018.
 - [122] A. Vollant, G. Balarac, and C. Corre. Subgrid-scale scalar flux modelling based on optimal estimation theory and machine-learning procedures. *Journal of Turbulence*, 18:854–878, 2017.
 - [123] B. Wang and J. Wang. Application of artificial intelligence in computational fluid dynamics. *Industrial & Engineering Chemistry Research*, 60:2772–2790, 2021.
 - [124] J. X. Wang, J. L. Wu, and H. Xiao. Physics-informed machine learning approach for reconstructing Reynolds stress modeling discrepancies based on DNS data. *Physical Review Fluids*, 2:034603, 2017.
 - [125] R. Wang, K. Kashinath, M. Mustafa, A. Albert, and R. Yu. Towards physics-informed deep learning for turbulent flow prediction. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1457–1466, 2020.
 - [126] R. Wang, R. Walters, and R. Yu. Incorporating symmetry into deep dynamics models for improved generalization. *arXiv preprint arXiv:2002.03061*, 2020.
 - [127] J. Weatheritt and R. D. Sandberg. A novel evolutionary algorithm applied to algebraic modifications of the RANS stress-strain relationship. *Journal of Computational Physics*, 325:22–37, 2016.
 - [128] J. Weatheritt and R. D. Sandberg. The development of algebraic stress models using a novel evolutionary algorithm. *International Journal of Heat and Fluid Flow*, 68:298–318, 2017.
 - [129] H. G. Weller, G. Tabor, H. Jasak, and C. Fureby. A tensorial approach to computational continuum mechanics using object-oriented techniques. *Computers in Physics*, 12:620–631, 1998.
 - [130] J. Wu, H. Xiao, R. Sun, and Q. Wang. Reynolds-averaged Navier–Stokes equations with explicit data-driven Reynolds stress closure can be ill-conditioned. *Journal of Fluid Mechanics*, 869:553–586, 2019.
 - [131] J.-L. Wu, H. Xiao, and E. Paterson. Physics-informed machine learning approach for augmenting turbulence models: A comprehensive framework. *Physical Review Fluids*, 3:074602, 2018.
 - [132] E. Yeung, S. Kundu, and N. Hodas. Learning deep neural network representations for koopman operators of nonlinear dynamical systems. *arXiv preprint arXiv:1708.06850*, 2017.
 - [133] Z. Zhang, L. Zhang, Z. Sun, N. Erickson, R. From, and J. Fan. Solving Poisson’s equation using deep learning in particle simulation of PN junction. *2019 Joint International Symposium on Electromagnetic Compatibility, Sapporo and Asia-Pacific International Symposium on Electromagnetic Compatibility (EMC Sapporo/APEMC)*, pages 305–308, 2019.