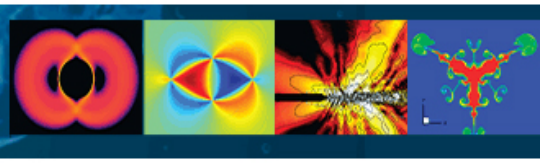
Data-driven recovery of hidden physics in reduced order modeling of fluid flows

Cite as: Phys. Fluids **32**, 036602 (2020); https://doi.org/10.1063/5.0002051
Submitted: 21 January 2020 . Accepted: 20 February 2020 . Published Online: 10 March 2020

Suraj Pawar, Shady E. Ahmed, Omer San, and Adil Rasheed



Physics of Fluids ARTICLE scitation.org/journal/phf

Data-driven recovery of hidden physics in reduced order modeling of fluid flows

Cite as: Phys. Fluids 32, 036602 (2020); doi: 10.1063/5.0002051
Submitted: 21 January 2020 • Accepted: 20 February 2020 •
Published Online: 10 March 2020

Suraj Pawar,[1] Shady E. Ahmed,[1] Omer San,[1,a)] and Adil Rasheed[2]

AFFILIATIONS

[1]School of Mechanical and Aerospace Engineering, Oklahoma State University, Stillwater, Oklahoma 74078, USA
[2]Department of Engineering Cybernetics, Norwegian University of Science and Technology, N-7465 Trondheim, Norway

[a)]**Author to whom correspondence should be addressed:** osan@okstate.edu

ABSTRACT

In this article, we introduce a modular hybrid analysis and modeling (HAM) approach to account for hidden physics in reduced order modeling (ROM) of parameterized systems relevant to fluid dynamics. The hybrid ROM framework is based on using first principles to model the known physics in conjunction with utilizing the data-driven machine learning tools to model the remaining residual that is hidden in data. This framework employs proper orthogonal decomposition as a compression tool to construct orthonormal bases and a Galerkin projection (GP) as a model to build the dynamical core of the system. Our proposed methodology, hence, compensates structural or epistemic uncertainties in models and utilizes the observed data snapshots to compute true modal coefficients spanned by these bases. The GP model is then corrected at every time step with a data-driven rectification using a long short-term memory (LSTM) neural network architecture to incorporate hidden physics. A Grassmann manifold approach is also adopted for interpolating basis functions to unseen parametric conditions. The control parameter governing the system's behavior is, thus, implicitly considered through true modal coefficients as input features to the LSTM network. The effectiveness of the HAM approach is then discussed through illustrative examples that are generated synthetically to take hidden physics into account. Our approach, thus, provides insights addressing a fundamental limitation of the physics-based models when the governing equations are incomplete to represent underlying physical processes.

I. INTRODUCTION

Advances in machine learning algorithms along with the huge amount of data generated from high-fidelity numerical simulations, lab experiments, and sensor data can be integrated with physical modeling to improve the prediction of complex dynamical systems. The physics-based approaches are interpretable, trustworthy, and highly generalizable, implying that they can be applied to a much wider array of problems (in the context of fluid mechanics, atmospheric flows, and flow around microprocessors), provided that they are governed by similar physics. However, they can be computationally very demanding. The machine learning algorithms, on the other hand, can lead to models which are computationally efficient for real-time predictions and can evolve with time as they experience new physics. However, their blackbox nature is currently preventing their full potential from being utilized in engineering applications. The current paper is an effort in the direction of combining these two approaches in the context of real-time prediction to develop a new paradigm in modeling called the *hybrid analysis and modeling* (HAM) which combines the best of both approaches to develop models which are generalizable, trustworthy, computationally efficient, and dynamically evolving in time. Readers are directed to Reichstein *et al.*[1] and references therein for an excellent perspective on the hybridization of physical modeling and machine learning algorithms in the context of geoscientific research. In addition, Karpatne *et al.*[2] offer insights into the paradigm of theory-guided data science for scientific problems involving a complex physical phenomenon and describe several approaches for integrating physical knowledge into data-driven methods. In recent years, there has been growing interest in using machine learning algorithms for fluid flow modeling.[3] In a complementary perspective, Ref. 4 outlined different opportunities and challenges of using machine learning for fluid mechanics. Furthermore, as highlighted by Rasheed *et al.*,[5] both injecting physical knowledge into machine learning models to make them more trustworthy and generalizable (*physics for machine learning*) and correcting physical models by machine

learning to enhance their fidelity and robustness (*machine learning for physics*) are getting more and more attention in physical and applied sciences.

The problems in fluid mechanics are very high-dimensional owing to a wide range of spatial and temporal scales that have to be resolved, which places an enormous computational burden on numerical simulations. There is a range of techniques that aim at constructing a reduced order model (ROM) that captures the essential features of these flows and is also computationally orders of magnitude faster than actual numerical simulations.[6–8] Furthermore, many physical processes are governed by parameterized partial differential equations, and there is increased attention in building parametric ROMs with high-fidelity over a range of parameters.[9] These ROMs are particularly essential for system identification,[10,11] design optimization,[12,13] flow control,[14–16] flow sensing,[17] data assimilation,[18–20] and uncertainty quantification[21] that require multiple forward simulations over a large range of parameters. In addition, in the context of digital twins, a model that is reliably accurate and computationally efficient is sought to allow for real-time and informed decision making strategies.[5,22,23] Hence, a ROM offers a viable and promising enabler for such frameworks as well.[24,25]

Proper orthogonal decomposition (POD) is one of the most popular model reduction methods that decompose the flow field into a set of basis functions that optimally describe the system and select only the most energy-conserving bases to represent the system.[26] The POD was introduced to the fluid mechanics community by Lumley[27] by showing the link between coherent structures in turbulent flows and the spatial structures of the POD modes. Different variants of POD have been introduced such as spectral POD[28] that enables the separation of a phenomenon that occurs at multiple frequencies and multiscale POD[29] that extracts the optimal basis from each scale. Dynamic mode decomposition (DMD)[30] is another model reduction method designed to decompose time-resolved data into modes, each one of which corresponds to a single characteristic frequency and growth rate. Since its introduction, many versions of the DMD approach have been introduced to enhance its robustness by analyzing the spectral properties of the Koopman operator and effectively de-biasing the resulting modes.[31,32] Another dimensionality reduction method is an autoencoder,[33] which can be considered as a nonlinear generalization of the POD.[34] Autoencoders have been successfully applied for lid-driven cavity flow,[35] advection-dominated flows,[36] and viscous flow around a cylinder.[37]

The POD is complemented by a Galerkin projection (GP) in which the dynamics of the system are modeled.[38–40] However, GP models often suffer from instability, and several successful closures have been proposed to model the effect of truncated modes.[41–44] In a recent study, it is also shown that a numerically stable and accurate ROM can be constructed using a Petrov–Galerkin projection framework.[45] We highlight that the GP framework is based on the physical model of the system and requires complete information about equations governing the system's dynamics. In many processes, there might be a discrepancy between the physical model and the observed data. This discrepancy arises due to approximations to complex physical processes, the requirement of parameterizations in physical models, imperfect knowledge about source terms, observational errors, etc. For example, the only information available for many geophysical flows might be the dynamical core represented by the vorticity transport equation. We might not have any prior accurate information about the hidden physics such as wind forcing, bottom friction, rotation, stratification, radiation, or other parameterizations.[46] However, the observed data for such systems will have the correct external forces embedded in it. The observed data can be utilized in a hybrid fashion to build accurate and robust ROMs for these systems.

Recently, there has been a significant effort to use machine learning algorithms to develop nonintrusive ROMs that do not require any information about the full order model (FOM) and solely rely on the data.[47–54] Guo and Hesthaven[55] proposed a combination of the reduced basis method and Gaussian process regression along with an active data selection algorithm for parameterized nonlinear structural analysis. Li *et al.*[56] utilized the LSTM neural network for constructing parametric unsteady aerodynamic ROMs for aeroelastic applications. Maulik *et al.*[57] presented a data-driven parametric ROM framework for advection-dominated systems with convolutional autoencoders for dimensionality reduction and recurrent neural networks for modeling the latent space dynamics.

Machine learning algorithms have also been utilized in identifying and extracting patterns from the observational data that can help us in the understanding of the physical phenomena and discovering the equations governing these phenomena.[58–63] Raissi and Karniadakis[64] proposed a data-efficient learning algorithm to discover underlying physical laws from sparse data using neural networks. Along a similar line, machine learning algorithms can be used to extract patterns from the observed data that are not included in a physical model and will allow us to improve the physical model.[65] Kou and Zhang[66] developed a multifidelity aerodynamic ROM framework based on data fusion with machine learning employed for learning the correction from the low-fidelity model to the high-fidelity model. In this study, instead of employing a purely data-driven method, we develop an algorithm that blends the physical knowledge with a machine learning approach for modeling parameterized systems with hidden physics.

The goal of the present study is to build a modular ROM for any parametrized process that can be defined by

$$\frac{\partial \mathbf{u}}{\partial t}(\mathbf{x}, t; \nu, \gamma) = \mathbf{F}_m(\mathbf{x}, t; \mathbf{u}; \nu) + \mathbf{\Pi}(\mathbf{x}, t; \mathbf{u}; \nu, \gamma), \tag{1}$$

where $\mathbf{u}$ is the prognostic variable, $\mathbf{F}_m$ is the physics-based model (i.e., dynamical core) governing the known process, and $\mathbf{\Pi}$ includes the external empirical parameterizations and unknown physics. Here, ν and γ refer to the control parameters that are relevant to the dynamical core and the hidden physics, respectively. The source term $\mathbf{\Pi}$ is usually unknown and can be learned using machine learning methods from the observed data. However, machine learning methods and, in particular, deep learning models lack interpretability and generalizability, and, hence, are prone to produce physically inconsistent results. Hence, there is active research going on to incorporate the physical knowledge into machine learning methods to make them physically consistent, such as loss regularization based on physical laws,[67,68] designing novel neural network architectures to embed certain physical properties,[69,70] building hybrid models to correct the imperfect knowledge in physical models,[46,71–74] and training neural networks guided by the physics.[75] How should we

inject physics and domain knowledge into machine learning models? When and how might machine learning be a complementary approach for generating more robust surrogate models? These are open research questions that we tackle, and the present study offers a glimpse into addressing these issues and pertains to introducing an effective hybrid modeling approach toward more accurate real-time predictions of fluid flows under epistemic or structural uncertainties.

The motivation behind this work is to synthesize a physics-based approach along with machine learning to model the hidden physics in parameterized ROMs. The paper is structured as follows: We introduce the main philosophy of the hybrid analysis and modeling (HAM) approach in Sec. II. We then discuss a specific formulation of the HAM framework in Sec. III for developing reduced order models for parameterized systems. The numerical results for the one-dimensional Burgers equation and two-dimensional vorticity transport equation with the HAM framework are detailed in Sec. IV. Finally, Sec. V will present conclusions and ideas for the future work.

II. HYBRID ANALYSIS AND MODELING (HAM) FRAMEWORK

In the context of upcoming technologies such as digital twin,[5] the role of cybernetics is to steer the system toward an optimal set-point. In order to do so, the output of the system is continuously monitored and compared against a reference. The difference, called the error signal, is applied as feedback to the controller, which generates a system input to bring the output set-point closer to the reference. With the availability of more and more sensors and communication technologies, an increasingly larger volume of data (in fact big data) is being made available in real-time. The challenge is then to develop a better understanding of the big data before it can be used for control purposes.

Most of these modeling approaches lie in either of the two categories: physics-based modeling or data-driven modeling. Recently, a third approach to modeling which is a combination of the two is fast emerging. However, there remains a lack of formal definition and terminology for the approach. In a nutshell, our hybrid modeling philosophy attempts to combine physics-based models with the versatility of data-driven approaches. Specifically, we define *hybrid analysis and modeling* (HAM) as a modeling approach that combines the interpretability, robust foundation, and understanding of a physics-based approach with the accuracy, efficiency, and automatic pattern-identification capabilities of advanced data-driven machine learning and artificial intelligence algorithms. This concept offers many new perspectives to our rapidly digitized society and its seamless interactions with many different fields.

Here, we pretend that some partially known physics govern the process of the vortex merging problem. The behavior of the vortices can be captured using RADAR or LIDAR. Using our partial understanding of the behavior of the vortices, we might utilize a first set of equations that govern vorticity transport processes. Since we do not have a good understanding of the effect of humidity and temperature on the evolution of these vortices, we simply claim that this unknown physics is captured by the term $\mathbf{\Pi}$. At this stage to explain the vortex behavior, we solve these equations without $\mathbf{\Pi}$. At this first iteration, the model is a pure physics-based model. It is generalizable because we have not tuned it with any case-specific data. However, the numerical solution of these equations can be computationally expensive. To address the problem of computational efficiency, *proper orthogonal decomposition* (POD) can be applied to the observed data to capture case-specific basis functions. Using these basis functions, the partial governing equations from the first stage are projected onto a reduced dimensional space to develop a system of ordinary differential equations which are relatively inexpensive to solve. This projection is commonly referred to as the Galerkin projection. However, at this stage, the unexplained term $\mathbf{\Pi}$ from the previous stage still remains unaccounted for. It only gets projected to a lower dimensional space. At the last stage and only as the last resort, we utilize a blackbox machine learning algorithm, e.g., *long short-term memory* (LSTM) neural network approach, to model this term. It is to be noted that if the LSTM misbehaves, then the mass and momentum conservation represented by the equations will be violated, and we can include this as a way to keep sanity check on the blackbox part of the workflow. Finally, we can also utilize a symbolic regression[65] approach to explicitly learn the functional form of $\mathbf{\Pi}$. Once the functional form is in place, we go and update the previous version of the model we started with. Hence, this way of running HAM will always be more generalizable, trustworthy, dynamically evolving, and computationally efficient.

The current paper is an attempt in this direction to not only formalize the definition of HAM but also to give directions on how this kind of modeling can be done. To this end, in Sec. III, we present a potential application area where hybrid modeling is supposed to make severe impacts. Here, we devise a hybrid reduced order model based on the Galerkin projection and machine learning correction (LSTM-based). The Galerkin projection model founded on POD modes is constructed to capture the core of the flow physics while machine learning is used to uncover the hidden physics. The accuracy of the model is demonstrated for a shock formation (Burgers equation) and vortex merging problem with forcing (2D incompressible flow). However, in the current demonstration, we consider a single family of solutions representing $\mathbf{\Pi}$ in Eq. (1) for quick prototyping. As a starting point, we set it up so that we can train on one set of $\mathbf{\Pi}$ and then test on a structurally similar set of $\mathbf{\Pi}$. However, by expanding the training set, including a wide range of possible physical processes, we might arguably agree that the proposed framework can be utilized in an effective manner for a variety of physical processes relevant to fluid dynamics and turbulence.

III. HAM FOR PARAMETERIZED SYSTEMS IN LOW DIMENSIONS

To construct a set of orthonormal POD basis functions, we collect the data snapshots, $\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_N \in \mathbb{R}^m$, at different time instants. We form the matrix $\mathbf{A} \in \mathbb{R}^{m \times N}$, whose columns are the snapshots $\mathbf{u}_n$, and then perform the singular value decomposition (SVD) of the matrix,

$$\mathbf{A} = \mathbf{W}\mathbf{\Sigma}\mathbf{V}^T = \sum_{k=1}^{N} \sigma_k \mathbf{w}_k \mathbf{v}_k^T, \tag{2}$$

where $\mathbf{W}$ is an $m \times N$ matrix with orthonormal columns $\mathbf{w}_k$, $\mathbf{V}$ is an $N \times N$ matrix with orthonormal columns $\mathbf{v}_k$, and $\mathbf{\Sigma}$ is an $N \times N$ matrix with non-negative diagonal entries, called singular values,

TABLE I. Galerkin projection operators for two prototypical examples, where ϕ_k^u refers to the *k*th *basis function of field u* in the Burgers equation. Similarly, in the vorticity transport equation, ϕ_k^ω and ϕ_k^ψ refer to basis functions of the vorticity and streamfunction, respectively.

Example	$\mathfrak{L}_{ik}$	$\mathfrak{N}_{ijk}$
Burgers equation	$\left\langle \frac{1}{\mathrm{Re}}\frac{\partial^2\phi_i^u}{\partial x^2}, \phi_k^u \right\rangle$	$\left\langle -\phi_i^u \frac{\partial \phi_j^u}{\partial x}, \phi_k^u \right\rangle$
Vorticity transport equation	$\left\langle \frac{1}{\mathrm{Re}}\left(\frac{\partial^2\phi_i^\omega}{\partial x^2}+\frac{\partial^2\phi_i^\omega}{\partial y^2}\right), \phi_k^\omega \right\rangle$	$\left\langle -\left(\frac{\partial\phi_i^\omega}{\partial x}\frac{\partial\phi_j^\psi}{\partial y}-\frac{\partial\phi_i^\omega}{\partial y}\frac{\partial\phi_j^\psi}{\partial x}\right), \phi_k^\omega \right\rangle$

arranged such that $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_N \geq 0$. The vectors $\mathbf{w}_k$ are the POD modes that we denote as ϕ_k in this text, and $\mathbf{\Phi} = \{\phi_k\}_{k=1}^R$ is the set of POD basis functions for any values of $R \leq N$.[6] The representation of the approximated field using the POD modes is as follows:

$$\mathbf{u}(\mathbf{x}, t_n) = \sum_{k=1}^{R} a_k(t_n)\phi_k(\mathbf{x}), \tag{3}$$

where $a_k^{(n)}$ are the time dependent modal coefficients and R refers to the number of modes retained in our model. The GP model equations are obtained by applying projection to our physical system [i.e., using the linear superposition given by Eq. (3) in Eq. (1) and applying the inner product of the resulting equation with the basis functions ϕ_k that are orthonormal to each other]. The resulting system of equations is given below,

$$\frac{da_k}{dt} = \underbrace{\sum_{i=1}^{R}\mathfrak{L}_{ik}a_i + \sum_{i=1}^{R}\sum_{j=1}^{R}\mathfrak{N}_{ijk}a_i a_j}_{\text{Physics-based model } G(a_k)} + \underbrace{\tilde{C}_k}_{\text{Hidden physics}}, \tag{4}$$

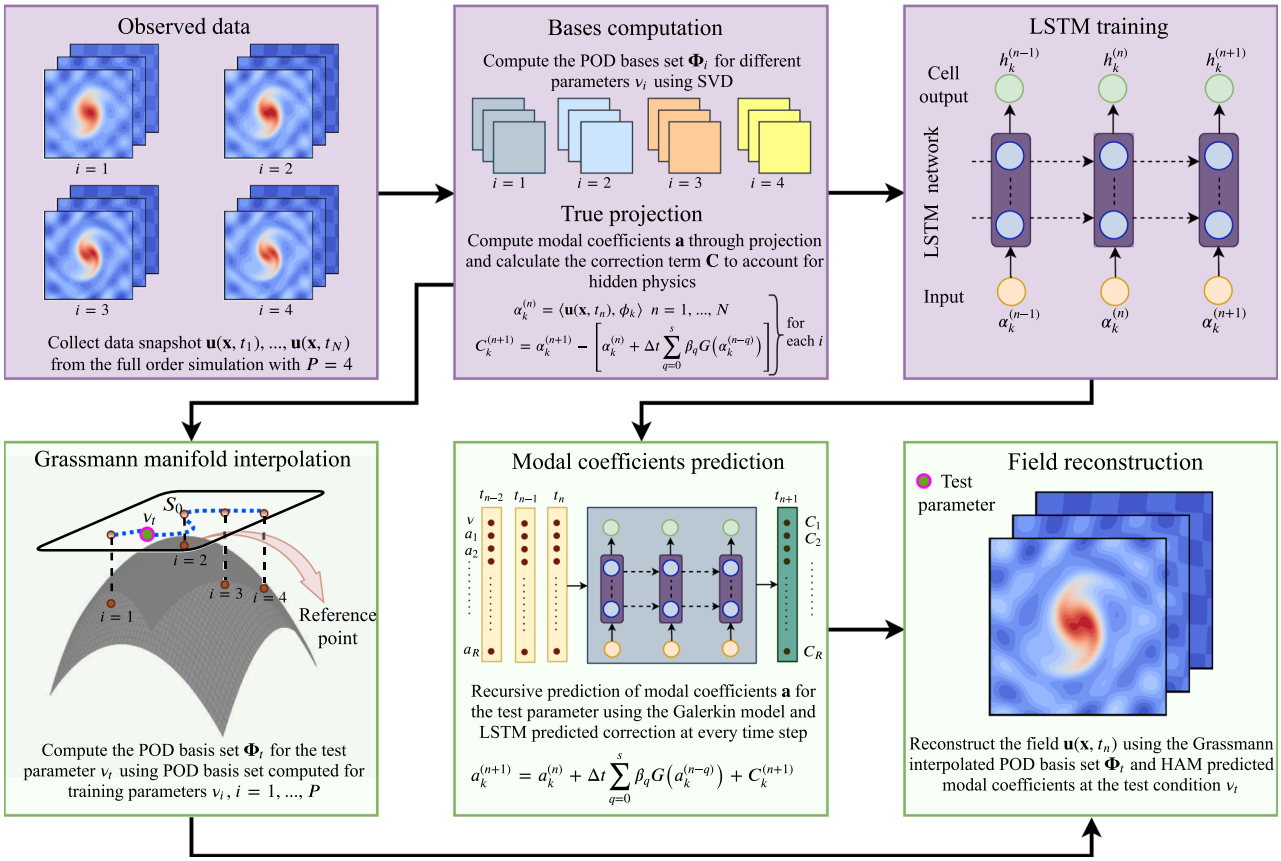


FIG. 1. Hybrid analysis and modeling (HAM) framework for model order reduction. Top and bottom rows refer to offline and online stages, respectively.

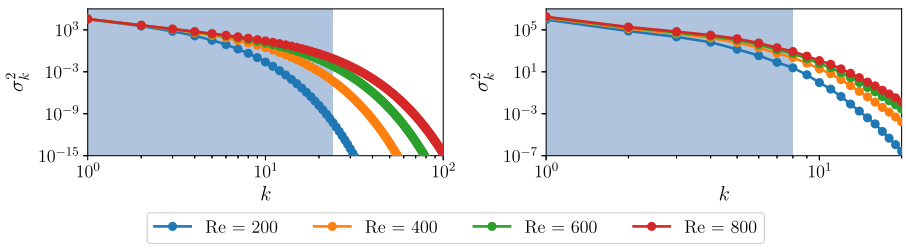


FIG. 2. Square of singular values of the snapshot data matrix **A** (equivalent to eigenvalues of $\mathbf{AA}^T$ or $\mathbf{A}^T\mathbf{A}$) for different Re training datasets obtained by the one-dimensional Burgers equation (left) and the two-dimensional vorticity transport equation with $\gamma = 0.1$ (right).

where $\mathfrak{L}$ and $\mathfrak{N}$ are the linear and nonlinear operators of the physical system, and $\tilde{C}$ is the part of the system for which the physical model is not available. The linear and nonlinear operators for the two prototypical examples investigated in this study are presented in Table I. The angle-parentheses in Table I refer to the Euclidean inner product defined as $\langle \mathbf{x}, \mathbf{y} \rangle = \mathbf{x}^T\mathbf{y} = \sum_{i=1}^{m} x_i y_i$. In a discrete sense, the update formula can be written as

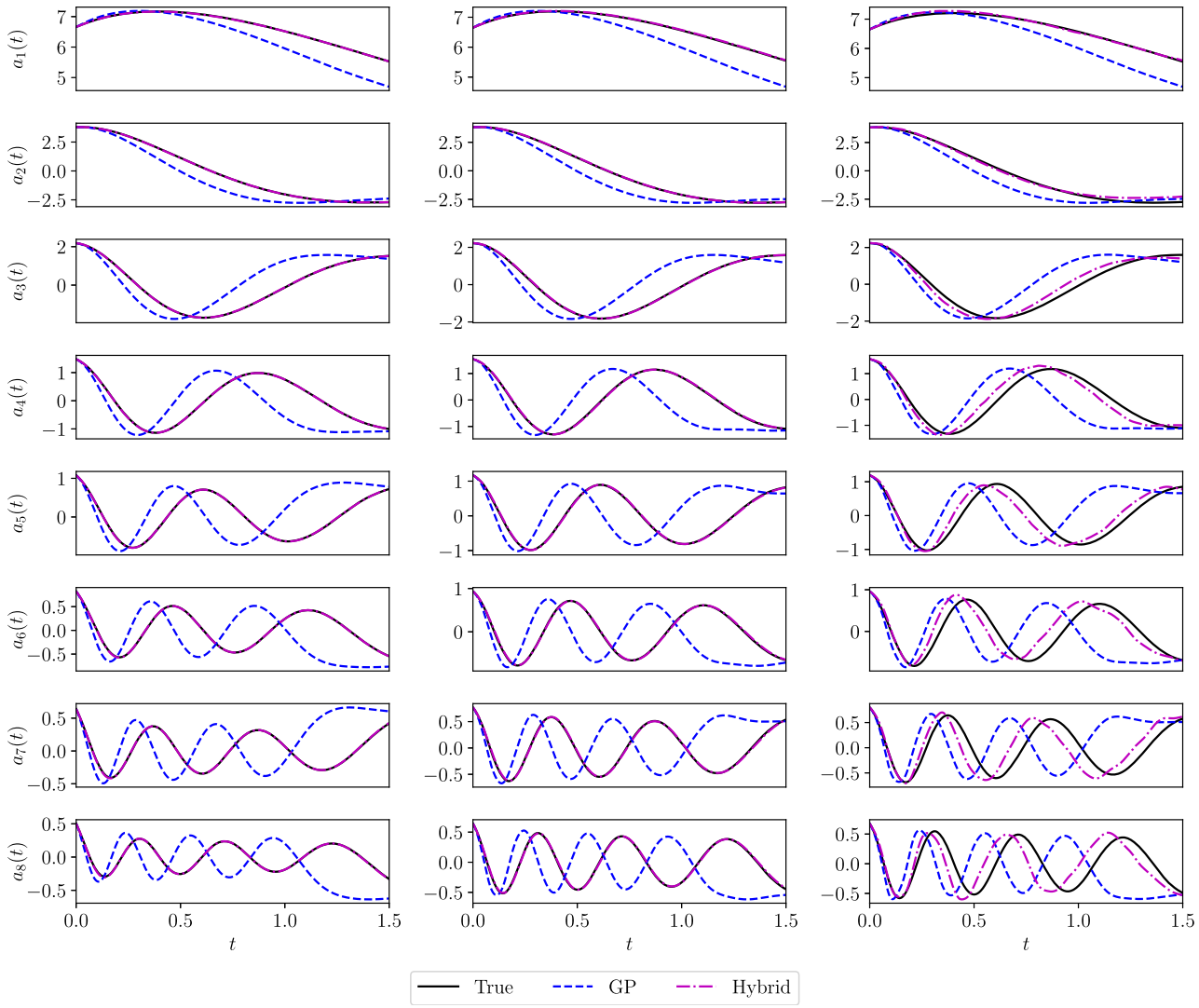


FIG. 3. Temporal evolution of modal coefficients for the Burgers equation at Re = 500 (left), Re = 1000 (middle), and Re = 1500 (right).

$$a_k^{(n+1)} = a_k^{(n)} + \Delta t \sum_{q=0}^{s} \beta_q [G(a_k^{(n-q)}) + \tilde{C}_k^{(n-q)}], \tag{5}$$

where s and β_q depend upon the numerical scheme used for the time integration. We use the third-order Adams–Bashforth (AB3) method in this study for which $s = 2$, $\beta_0 = 23/12$, $\beta_1 = -16/12$, and $\beta_2 = 5/12$. The physics-based GP model at any time t_n is

$$G(a_k^{(n)}) = \sum_{i=1}^{R} \mathfrak{L}_{ik} a_i^{(n)} + \sum_{i=1}^{R} \sum_{j=1}^{R} \mathfrak{N}_{ijk} a_i^{(n)} a_j^{(n)}. \tag{6}$$

In our HAM approach, since we attempt to learn the hidden physics part $\tilde{C}$ by using a supervised data-driven approach, Eq. (5) can be rewritten as

$$a_k^{(n+1)} = a_k^{(n)} + \Delta t \sum_{q=0}^{s} \beta_q G(a_k^{(n-q)}) + C_k^{(n+1)}, \tag{7}$$

where we set apart $\tilde{C}_k$ from the time integrator and represent hidden physics by C_k in the discrete form given by Eq. (7). We can generate a training set from the true projection of observed data onto POD basis. These true projection modal coefficients encompass both the dynamical core of the system and hidden physics. The true projection modal coefficients are computed as below,

$$\alpha_k^{(n)} = \langle \mathbf{u}(\mathbf{x}, t_n), \phi_k \rangle. \tag{8}$$

Using Eqs. (7) and (8), we can recover the hidden physics as below,

$$C_k^{(n+1)} = \alpha_k^{(n+1)} - \left[\alpha_k^{(n)} + \Delta t \sum_{q=0}^{s} \beta_q G(\alpha_k^{(n-q)}) \right]. \tag{9}$$

We can use any of the suitable supervised machine learning algorithms to learn this correction term C_k. In this study, we employ a long short-term memory (LSTM) neural network algorithm to learn the mapping from true modal coefficients to the correction term (e.g., $\{\alpha_1, \ldots, \alpha_R\} \in \mathbb{R}^R \rightarrow \{C_1, \ldots, C_R\} \in \mathbb{R}^R$). At this point, we highlight that the mapping architecture can be constructed in various forms depending on the problem. Since we test our proposed approach against unseen conditions, we include the underlying control parameters in our training (i.e., $\{\nu, \alpha_1, \ldots, \alpha_R\} \in \mathbb{R}^{R+1} \rightarrow \{C_1, \ldots, C_R\} \in \mathbb{R}^R$, where ν is the control parameter parameterizing the system's behavior, e.g., Reynolds number). The LSTM network is particularly suitable for time-series data as it considers

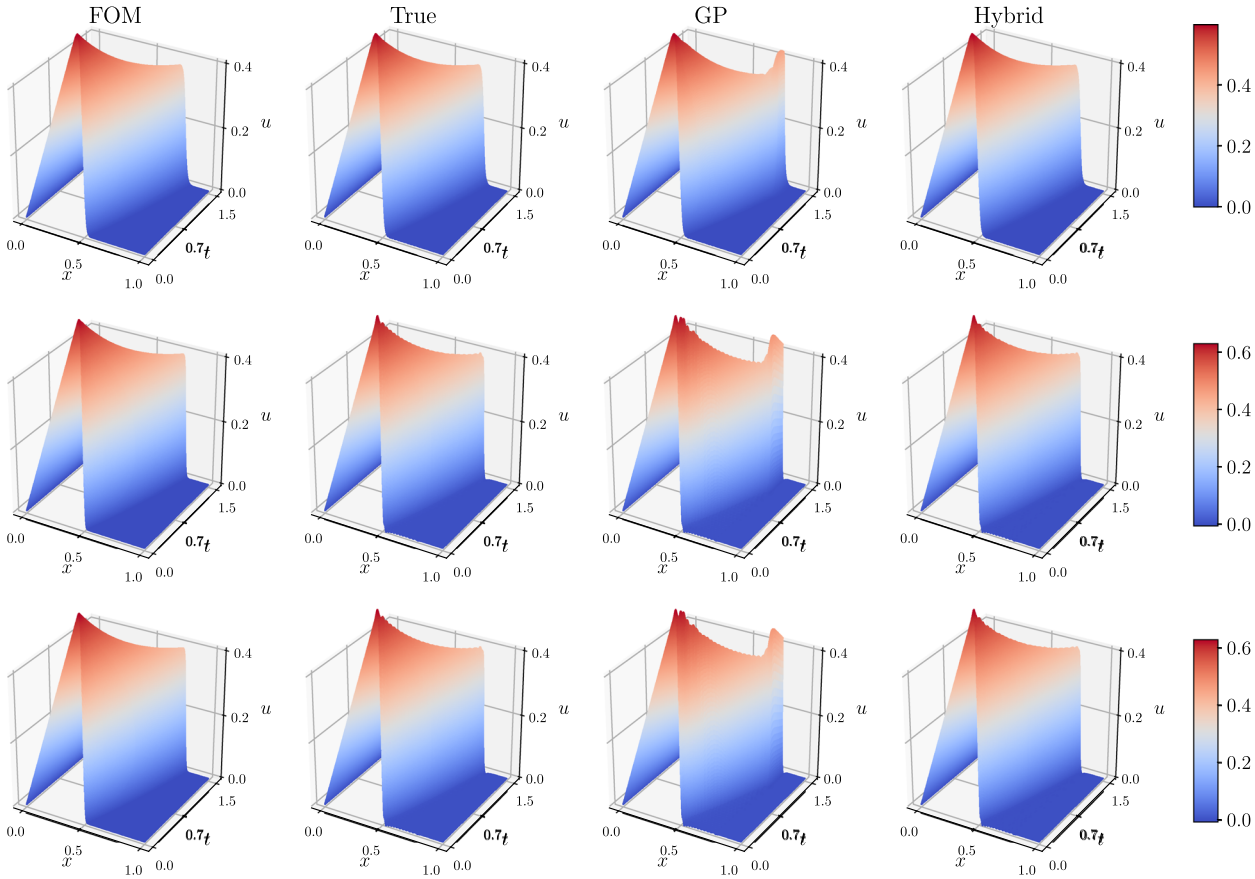


FIG. 4. Space-time solution field for the Burgers equation for an interpolatory test condition at Re = 500 (top), and extrapolatory test conditions at Re = 1000 (middle) and Re = 1500 (bottom).

the effect of the past state of the system in the future prediction of the state of the system. Other architectures might benefit from including the time index into the input layer.[76] Because of its built-in memory embedding capability, the LSTM network has been used in many nonintrusive ROM studies.[49,77,78] The AB3 multi-step scheme requires three previous time steps to perform temporal integration for the forward model. Thus, we use a lookback of three steps in the LSTM architecture to be consistent with the time integration of the GP model (e.g., see the work of Rahman *et al.*[48] for further details). During the deployment, we start with an initial condition and then at every time step, the correction is added to the GP model $G(a_k)$, and we recursively proceed with this process. We would like to note here that the GP model and the correction model are tightly coupled in this dynamic framework. Therefore, it is possible to have stability problems either because of the incompleteness of the Galerkin projection or the incorrect input from the LSTM network. However, in our numerical experiments, the stability issues were not encountered due to the accurate prediction of the correction term at every time step. Additionally, the stability issues[79] due to the truncation of modes were not faced as the number of retained modes capture more than 99.9% energy in our numerical experiments. The parameter governing the behavior of the physical system is taken implicitly into account through a physics-based GP model, and explicitly using the control parameter as an input feature to the LSTM network. This augments the learning process to improve the generalizability of the model.

During training, we compute the POD basis sets $\mathbf{\Phi} = [\phi_1, \phi_2, \ldots, \phi_R]$ for different parameters governing the physical system. We utilize the Grassmann manifold interpolation approach[80,81] to compute the POD basis set for the unseen test parameter from a set of available POD basis functions constructed in the offline phase. The Grassman manifold interpolation consists of choosing a reference

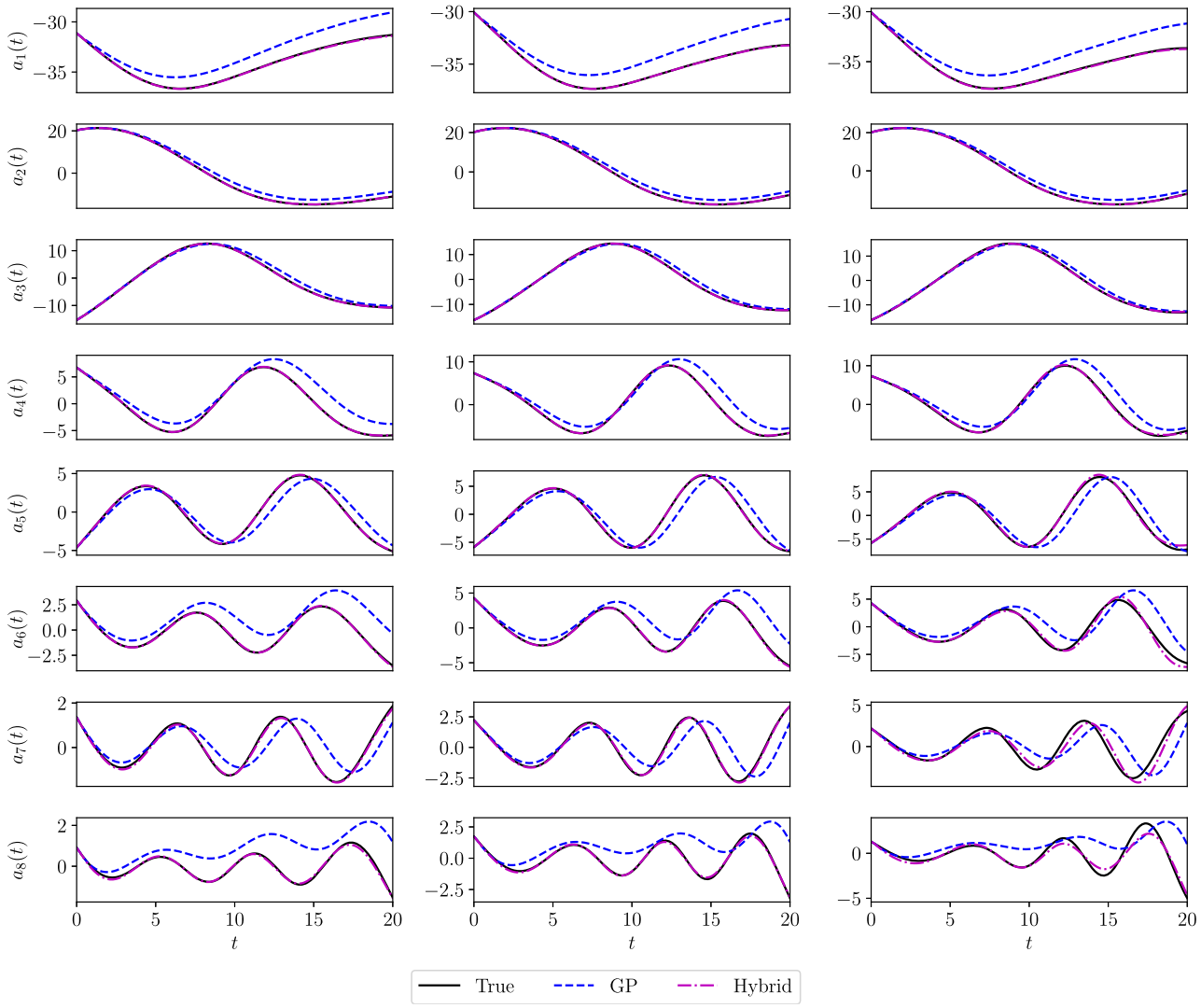


FIG. 5. Temporal evolution of vorticity modal coefficients for the vortex-merger problem at Re = 500 (left), Re = 1000 (middle), and Re = 1500 (right) with $\gamma = 0.01$.

point S_0, corresponding to the basis set $\mathbf{\Phi}_0$, and then mapping each point S_i to a matrix $\mathbf{\Gamma}_i$ which represents a point on the tangent space at S_0 using the logarithmic map $\log_{S_0}$,

$$(\mathbf{\Phi}_i - \mathbf{\Phi}_0\mathbf{\Phi}_0^T\mathbf{\Phi}_i)(\mathbf{\Phi}_0^T\mathbf{\Phi}_i)^{-1} = \mathbf{U}_i\mathbf{\Sigma}_i\mathbf{V}_i^T, \tag{10}$$

$$\mathbf{\Gamma}_i = \mathbf{U}_i\tan^{-1}(\mathbf{\Sigma}_i)\mathbf{V}_i^T. \tag{11}$$

The matrix $\mathbf{\Gamma}_t$ corresponding to the test parameter ν_t is obtained by the Lagrange interpolation of the matrices $\mathbf{\Gamma}_i$ corresponding to ν_i, $i = 1, 2, \ldots, P$,

$$\mathbf{\Gamma}_t = \sum_{i=1}^{P}\left(\prod_{\substack{j=1\\ j\neq i}}^{P}\frac{\nu_t - \nu_j}{\nu_i - \nu_j}\right)\mathbf{\Gamma}_i, \tag{12}$$

where P refers to the number of the control parameters for offline simulations (e.g., $P = 4$ in our numerical examples). The POD basis set $\mathbf{\Phi}_t$ corresponding to the test parameter ν_t is computed using the exponential mapping as follows:

$$\mathbf{\Gamma}_t = \mathbf{U}_t\mathbf{\Sigma}_t\mathbf{V}_t^T, \tag{13}$$

$$\mathbf{\Phi}_t = [\mathbf{\Phi}_0\mathbf{V}_t\cos(\mathbf{\Sigma}_t) + \mathbf{U}_t\sin(\mathbf{\Sigma}_t)]\mathbf{V}_t^T. \tag{14}$$

We note that trigonometric functions apply only to diagonal elements. The main blocks of the hybrid analysis and modeling (HAM) framework are shown in Fig. 1. In our numerical experiments, we use two hidden layers with 80 cells for both Burgers and vorticity transport equations to train the LSTM network. Our experiments with different sets of hyperparameters show that the LSTM network is not highly sensitive to hyperparameters.

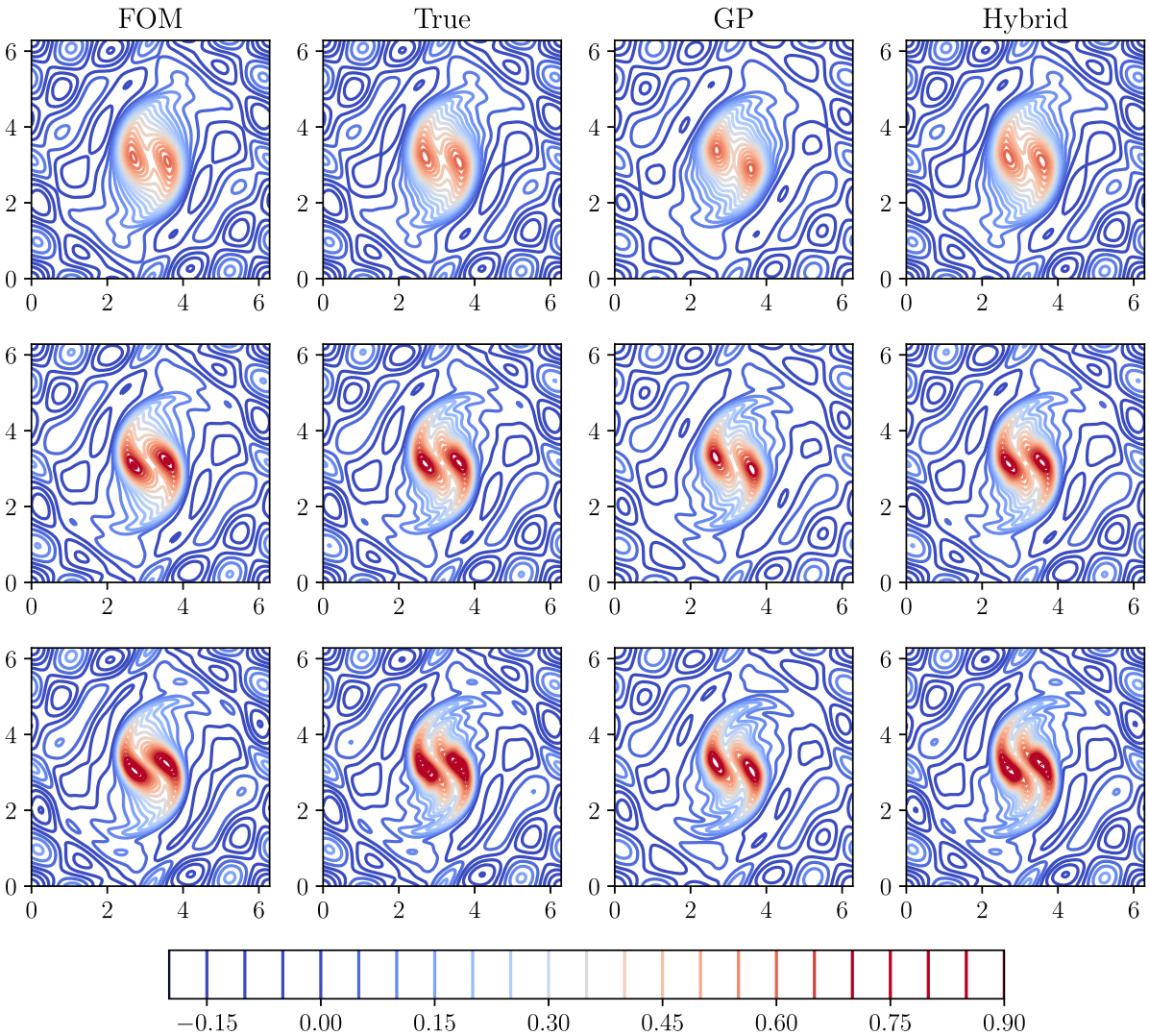


FIG. 6. Vorticity field at time $t = 20$ for the vortex-merger problem testing an interpolatory condition at Re = 500 (top), and extrapolatory conditions at Re = 1000 (middle) and Re = 1500 (bottom) with $\gamma = 0.01$.

IV. NUMERICAL EXPERIMENTS

A. One-dimensional Burgers equation

We test the performance of the HAM framework for the one-dimensional Burgers equation, which is a prototypical example for nonlinear advection–diffusion problems. The Burgers equation can be written as

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} = \frac{1}{\mathrm{Re}}\frac{\partial^2 u}{\partial x^2} + \Pi, \quad x \in [0,1], \quad t \in [0,1.5] \tag{15}$$

and accepts an analytical solution in the form of

$$u(x,t) = \frac{\frac{x}{t+1}}{1+\sqrt{\frac{t+1}{t_0}}\exp\left(\mathrm{Re}\frac{x^2}{4t+4}\right)}, \tag{16}$$

when the hidden physics part is set to $\Pi = 0$. Here, $t_0 = \exp(\mathrm{Re}/8)$ and Re is the Reynolds number that parameterizes the Burgers equation. We add 30% perturbation to this canonical solution to represent the hidden physics with an unknown source term. Therefore, our snapshots read from

$$u(x,t) = 1.3\frac{\frac{x}{t+1}}{1+\sqrt{\frac{t+1}{t_0}}\exp\left(\mathrm{Re}\frac{x^2}{4t+4}\right)}, \tag{17}$$

and we note that $\Pi \neq 0$. In our HAM approach, we model this part Π with a data-driven machine learning model. We generate the data snapshots using the analytical solutions given by Eq. (17) for Re = [200, 400, 600, 800] and evaluate the performance of the HAM framework for Re = 500, 1000, and 1500. The reference point in the Grassmann manifold interpolation is set to Re = 400 for testing at

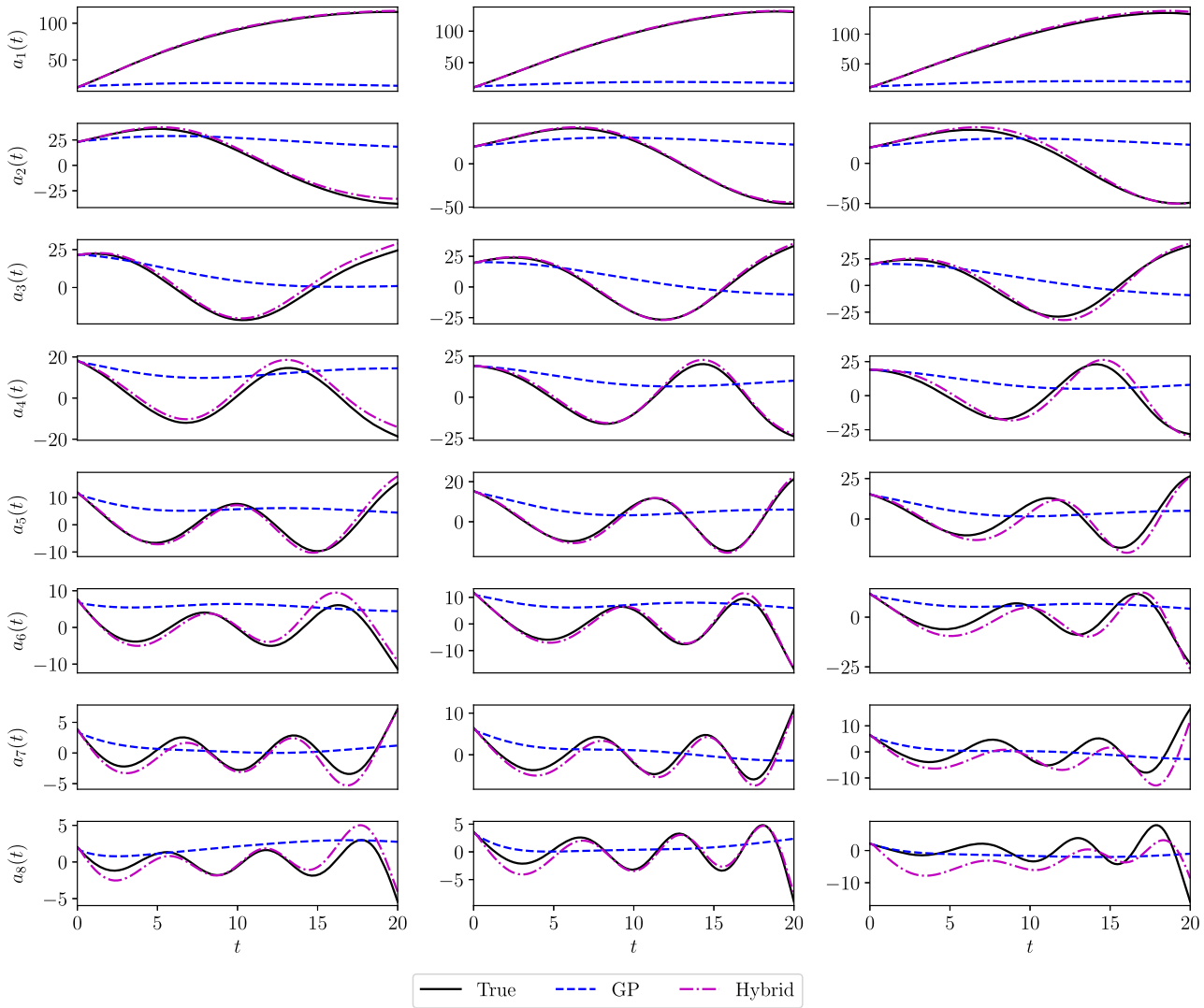


FIG. 7. Temporal evolution of vorticity modal coefficients for the vortex-merger problem at Re = 500 (left), Re = 1000 (middle), and Re = 1500 (right) with $\gamma = 0.1$.

Re = 500, and Re = 800 for testing at Re = 1000 and 1500. The performance of the HAM framework for these test Reynolds numbers can inform us about its interpolatory and extrapolatory prediction capability.

We use $R = 24$ modes in our ROM to represent the Burgers equation, which captures more than 99.99% energy at all Reynolds numbers. The eigenvalues of the Burgers equation for different Reynolds numbers included in training are shown in Fig. 2. From Fig. 3, we observe that in the presence of the source term, the GP is not able to predict the same trajectory as the true projection even with 24 modes (we show only the first eight modes in all plots). The HAM framework adds the correction due to the source term at every time step and follows the trajectory similar to the true projection modal coefficients as shown in Fig. 3 for the interpolatory Reynolds number Re = 500. We note that the HAM framework aims at modeling not only the closure term (due to truncation of higher modes) but also hidden physics. The prediction of modal coefficients is very accurate at Re = 1000, which is slightly extrapolated from the training Reynolds number regime. We see some discrepancy in true and hybrid modal coefficients for Re = 1500, which can be attributed to the limited extrapolation capability of data-driven methods as demonstrated for turbulent isotropic flows.[82] This discrepancy can also be a result of disparity between true basis and interpolated basis functions, which can be minimized by incorporating a richer offline

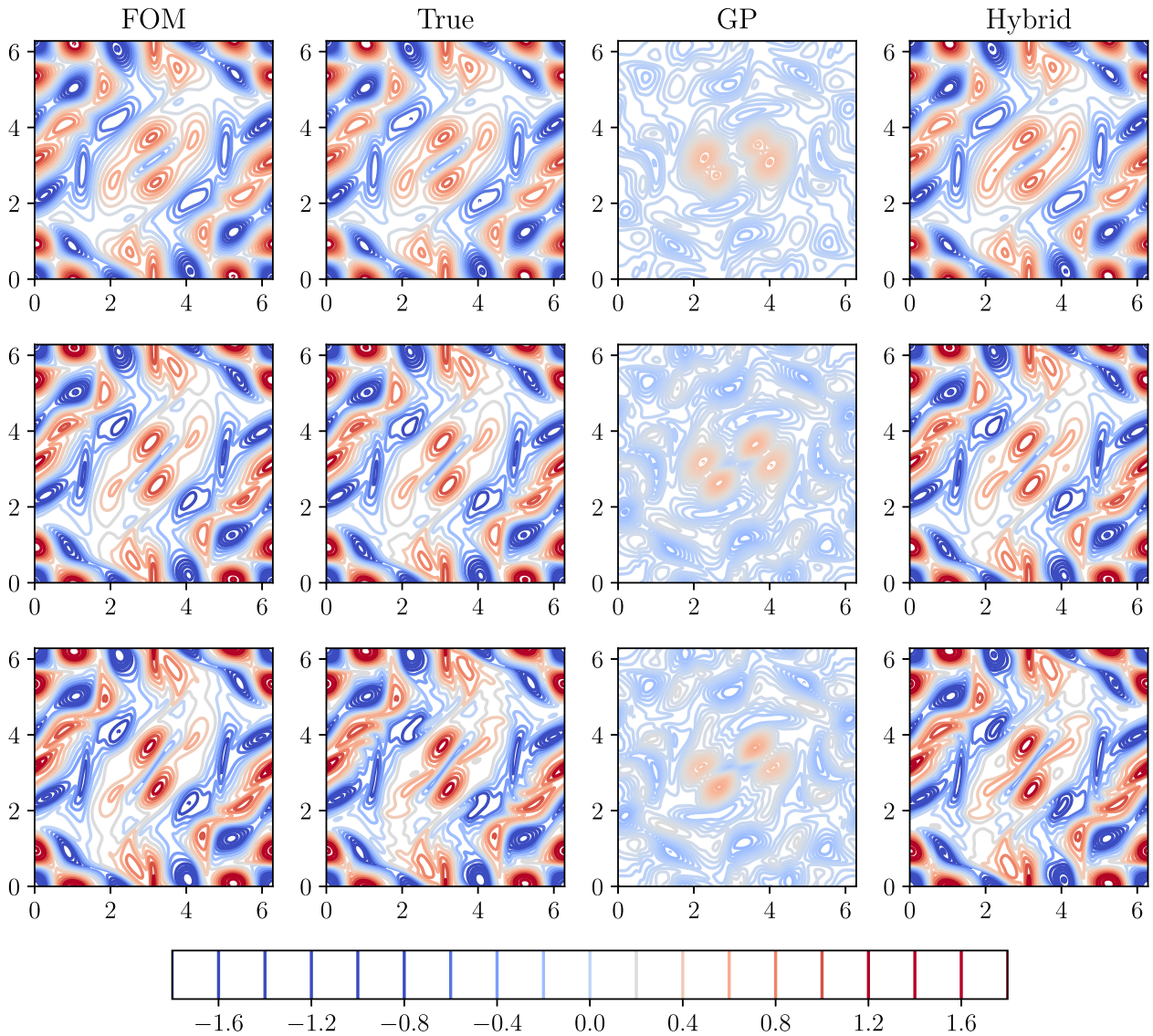


FIG. 8. Vorticity field at time $t = 20$ for the vortex-merger problem testing an interpolatory condition at Re = 500 (top), and extrapolatory conditions at Re = 1000 (middle) and Re = 1500 (bottom) with $\gamma = 0.1$.

dataset. Figure 4 shows the time evolution of the solution field from t = [0, 1.5] for FOM and the reconstructed solution field with true, GP, and hybrid modal coefficients. The HAM framework can produce the solution field similar to the true projection for all test Reynolds numbers, i.e., Re = 500, 1000, and 1500.

B. Two-dimensional vorticity transport equation

After a successful demonstration of the HAM framework for the Burgers equation, we test the HAM framework for two-dimensional vorticity transport equation applied to the vortex-merger problem. This problem is a prototypical example used to study the merging of two co-rotating vortices. Initially, two vortices of the same sign are separated by some distance and end up as a single nearly axisymmetric vortex after some time. The two-dimensional vorticity transport equation in vorticity-streamfunction formulation can be written as

$$\frac{\partial \omega}{\partial t} + \frac{\partial \psi}{\partial y}\frac{\partial \omega}{\partial x} - \frac{\partial \psi}{\partial x}\frac{\partial \omega}{\partial y} = \frac{1}{\mathrm{Re}}\left(\frac{\partial^2 \omega}{\partial x^2} + \frac{\partial^2 \omega}{\partial y^2}\right) + \Pi, \tag{18}$$

$$\frac{\partial^2 \psi}{\partial x^2} + \frac{\partial^2 \psi}{\partial y^2} = -\omega, \tag{19}$$

where ω is the vorticity defined as $\omega = \nabla \times \mathbf{u}$, $\mathbf{u} = [u,v]^T$ is the velocity vector, and ψ is the streamfunction. We start from an initial vorticity field of two Gaussian-distributed vortices,

$$\omega(x,y,0) = \exp\left(-\pi\left[(x-x_1)^2 + (y-y_1)^2\right]\right) + \exp\left(-\pi\left[(x-x_2)^2 + (y-y_2)^2\right]\right), \tag{20}$$

where their centers are initially located at $(x_1, y_1) = (3\pi/4, \pi)$ and $(x_2, y_2) = (5\pi/4, \pi)$. We add a perturbation field (i.e., referring to hidden physics) by utilizing an arbitrary array of decaying Taylor–Green vortices as the source term, which is given below,

$$\Pi = -F(t)\cos(3x)\cos(3y), \tag{21}$$

where $F(t) = \gamma\, e^{-t/\mathrm{Re}}$. We use the computational domain $(x, y) \in [0, 2\pi]$ with periodic boundary conditions. We generate data snapshots for Re = [200, 400, 600, 800] with 256^2 spatial grids and a time step of 0.01 from time $t = 0$ to $t = 20$. We test the HAM framework for out-of-sample conditions at Re = 500 (interpolatory), Re = 1000 and Re = 1500 (extrapolatory). The Grassmann manifold interpolation utilizes the reference dataset from Re = 400 for testing at Re = 500, and Re = 800 is chosen as the reference for testing at Re = 1000 and Re = 1500.

We use eight modes in our ROM for the vorticity transport equation and these eight modes capture more than 99.95% energy for all Reynolds numbers included in the training. Figure 2 shows the eigenvalues for 2D vorticity transport equation. We use two different amplitudes for Taylor–Green vortices (i.e., $\gamma = 0.01$ and $\gamma = 0.1$) to demonstrate the effectiveness of the HAM framework in modeling different magnitudes of hidden physics. First, we present the results for the case of $\gamma = 0.01$ that represents a comparatively easier setting due to its small magnitude compared to the actual vorticity field. Figure 5 shows the trajectories of the vorticity modal coefficients obtained by the true projection, GP, and HAM framework for $\gamma = 0.01$. The GP deviates from the true projection modal coefficients due to the presence of a source term that is not embedded in the

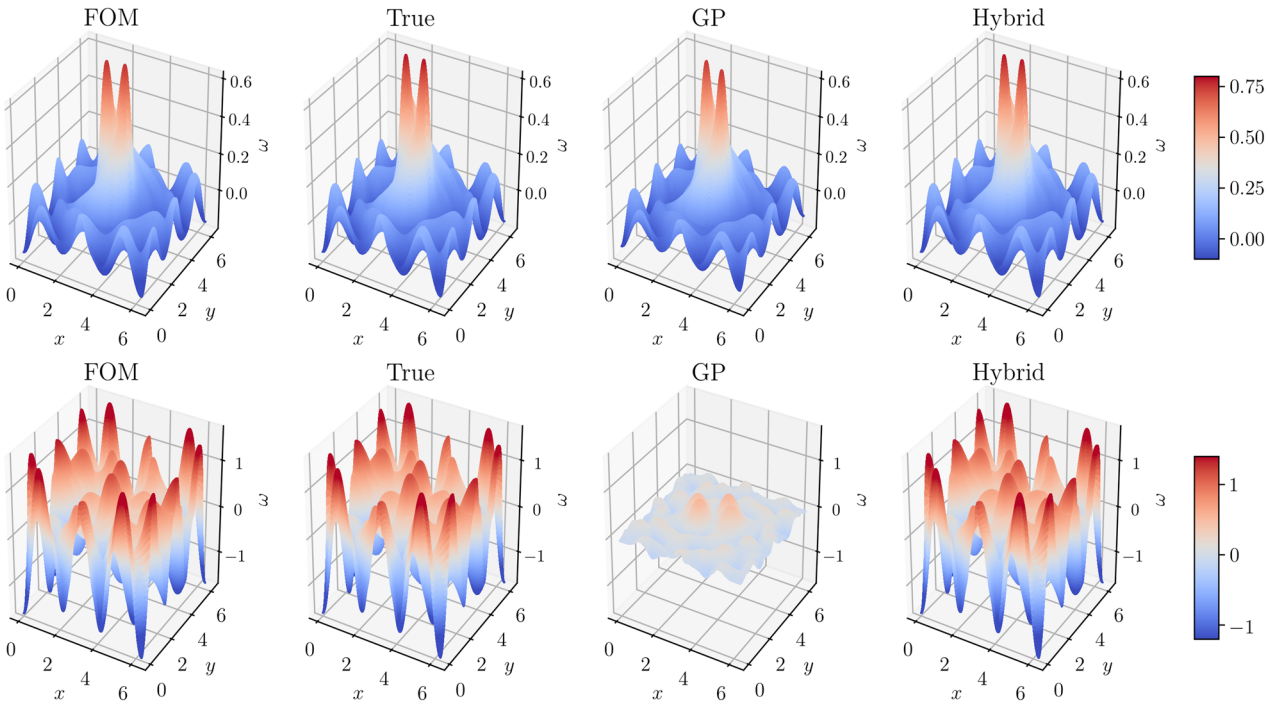


FIG. 9. Three-dimensional vorticity field at time $t = 20$ for the vortex-merger problem testing at Re = 1000 with $\gamma = 0.01$ (top) and $\gamma = 0.1$ (bottom).

GP model. The HAM framework can accurately predict the effect of the source term through the LSTM network at every time step, and we observe that the hybrid modal coefficients follow almost the same trajectory as true modal coefficients. We note a small discrepancy between true and hybrid modal coefficients, especially for later modes (i.e., a_6, a_7 and a_8) for Re = 1500. Figure 6 displays the vorticity field at the final time $t = 20$ for FOM and the reconstructed field using true, GP, and hybrid modal coefficients. The GP fails to capture the correct orientation of two counter-rotating vortices at the final time for all test Reynolds numbers. The HAM framework predicts the correct orientation of vortices with sufficient accuracy in comparison to the FOM vorticity field.

Next, we present the results of our numerical experiments with $\gamma = 0.1$. Figure 7 shows the vorticity modal coefficient trajectories for $\gamma = 0.1$ at all three test Reynolds numbers. The GP modal coefficients deviate significantly from true projection modal coefficients as they do not incorporate any information about the hidden physics, and for this test case, the major contribution comes from the unknown source term. The HAM framework, on the other hand, can produce accurate dynamics for both Re = 500 and 1000 with little deviation near the final time for a few modes. The deviation in true and hybrid modal coefficients is more at the Re = 1500 test case, which is far from the training Reynolds number regime. Figure 8 shows the vorticity field at $t = 20$, and it can be noted that the HAM framework predicts the vorticity field with sufficient accuracy for all Reynolds numbers. To demonstrate the difference in magnitude of source terms, we plot a three-dimensional view of the vorticity field at Re = 1000 for $\gamma = 0.01$ and $\gamma = 0.1$ in Fig. 9. The magnitude of

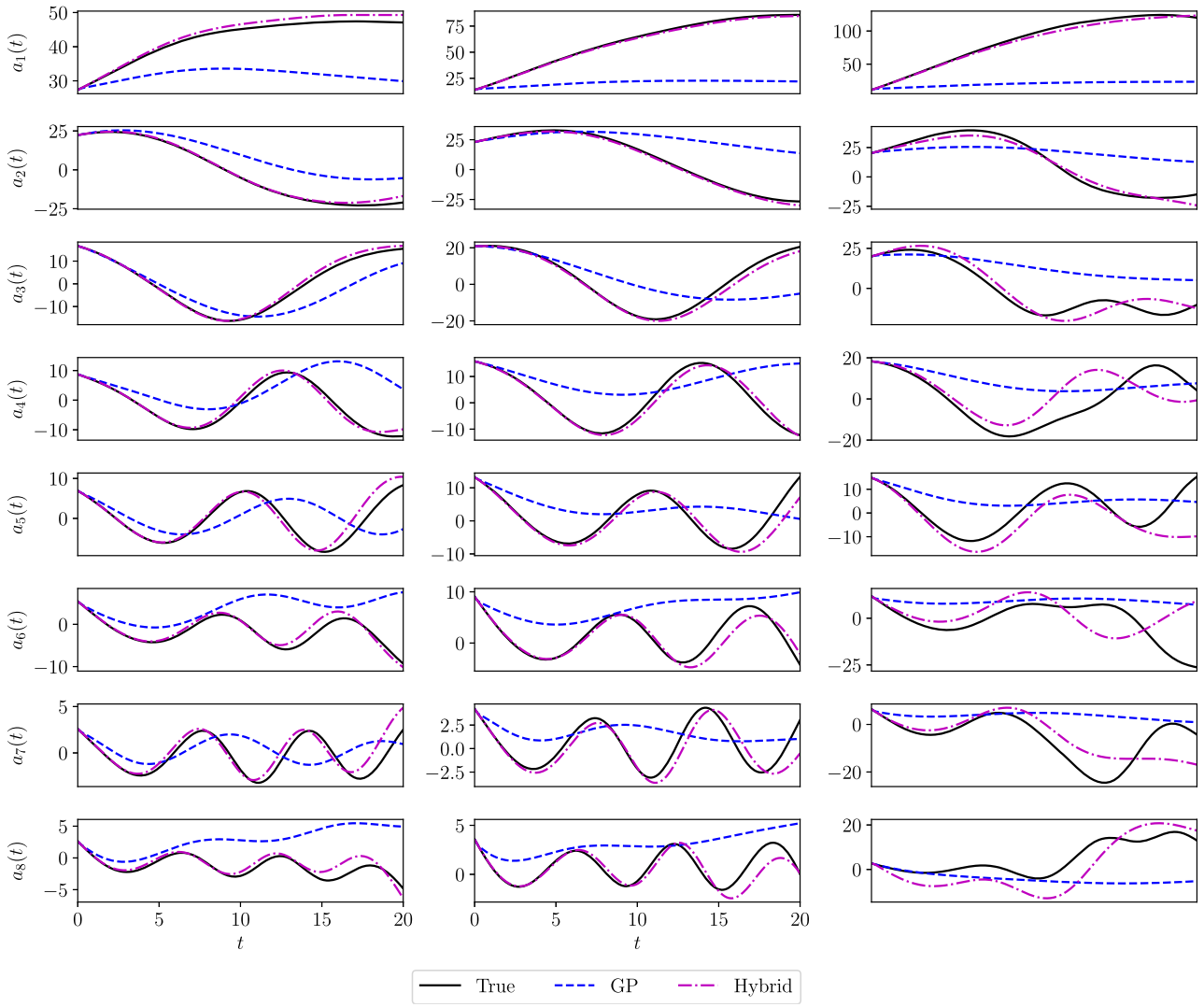


FIG. 10. Temporal evolution of vorticity modal coefficients for the vortex-merger problem for Re = 1000 with $\gamma = 0.03$ (left), $\gamma = 0.06$ (middle), and $\gamma = 0.1$ (right). The model is trained for $\gamma = 0.01$, 0.02, 0.04, 0.05, 0.07, 0.08, and 0.09.

decaying Taylor–Green vortices is very small compared to two Gaussian-distributed vortices for $\gamma = 0.01$. In the case of $\gamma = 0.1$, the magnitude of Taylor–Green vortices is of the same order of magnitude or even more compared to two Gaussian-distributed vortices. This illustrates the advantage of the hybrid framework to model physical systems where the major contribution comes from an unknown source term.

In the previous set of numerical experiments, we looked at the Reynolds number as the control parameter for the vorticity transport equation. We consider the strength of the source term, i.e., γ as the control parameter in the next set of numerical experiments. We keep the Reynolds number constant at Re = 1000, train the LSTM network with different magnitudes of the source term (i.e., $\gamma = 0.01$, 0.02, 0.04, 0.05, 0.07, 0.08, 0.09), and test the hybrid framework for $\gamma = 0.03$, 0.06, and 0.10. Figure 10 shows the evolution of vorticity modal coefficients for different test source terms. We obtain an accurate prediction of modal coefficients for $\gamma = 0.03$ and $\gamma = 0.06$, which lie in the interpolatory regime of training source terms. It can also be observed that the pattern of vorticity modal coefficients is different at $\gamma = 0.1$ compared to source terms of other magnitudes, suggesting a slight discrepancy between true and interpolated basis functions. Figure 11 displays a three-dimensional view of the vorticity field for FOM and compares with the reconstructed vorticity fields using true, GP, and hybrid modal coefficients. The hybrid framework can predict the accurate vorticity field at $\gamma = 0.03$ and $\gamma = 0.06$. We observe a larger deviation of the vorticity field from FOM at $\gamma = 0.1$ in the region where the Gaussian-distributed vortices are generated as an initial condition. This discrepancy is due to both inaccuracy in the interpolation of basis functions and modal coefficients.

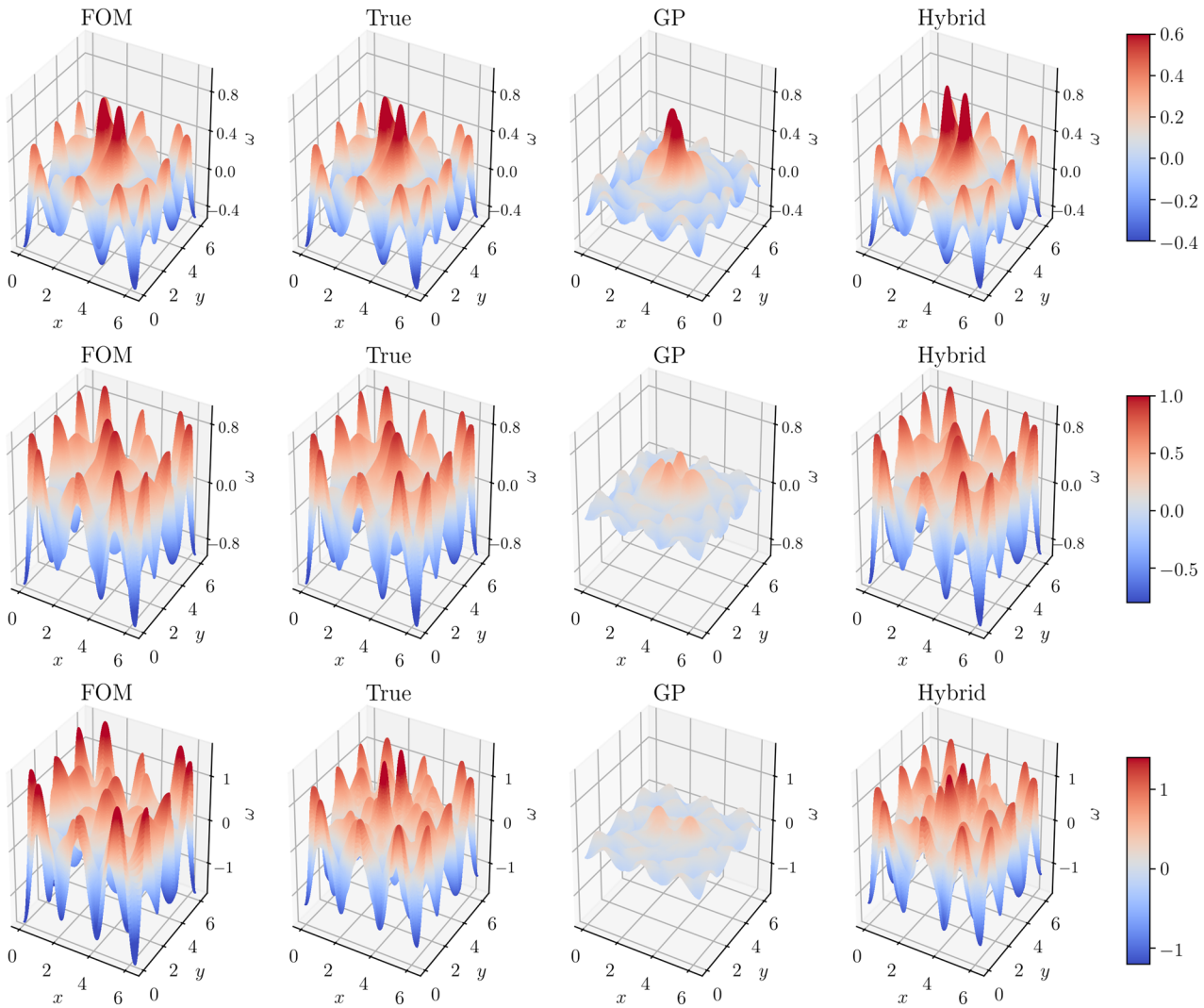


FIG. 11. Vorticity field at time $t = 20$ for the vortex-merger problem for Re = 1000 with $\gamma = 0.03$ (top), $\gamma = 0.06$ (middle), and $\gamma = 0.1$ (bottom). The model is trained for $\gamma = 0.01$, 0.02, 0.04, 0.05, 0.07, 0.08, and 0.09.

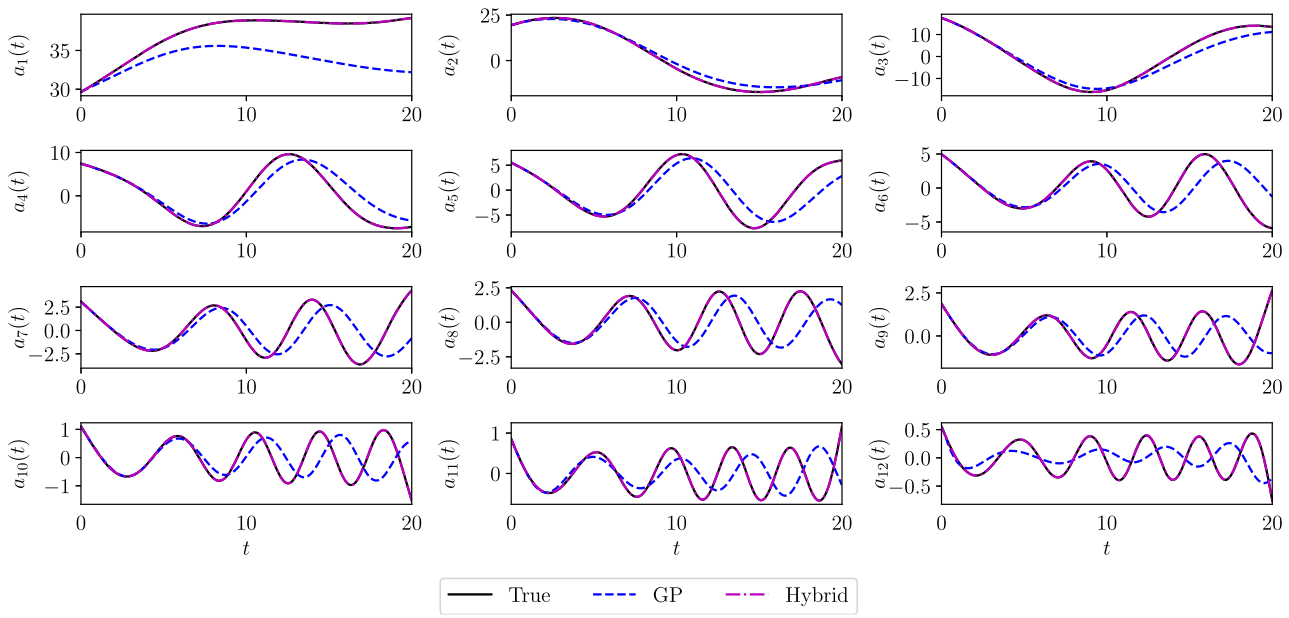


FIG. 12. Temporal evolution of vorticity modal coefficients for the vortex-merger problem for Re = 1000 with $\sigma = 0.01$.

In another example of the vorticity transport equation, we can consider a hypothetical case where a possible friction process occurs in large scale ocean basins. For example, a linear forcing term might represent the hidden physics in our model, which should be modeled as

$$\Pi = \sigma\omega, \tag{22}$$

where σ is the Ekman bottom layer friction coefficient. The parameter σ is set using the Stommel scale, and it accounts for larger scale damping in quasigeostrophic ocean models.[83,84] If the friction term is missing in the physical model, the GP model would use only the vorticity transport equation to model the dynamics of the process. On the other hand, the hybrid framework is able to learn the missing physics from the data. The motivation behind this example is to demonstrate the advantage of the hybrid framework in recovering missing physics from the data itself to augment the physical model.

First, we test the hybrid framework with a smaller magnitude of friction, $\sigma = 0.01$, at Re = 1000. Figure 12 shows the time evolution of vorticity modal coefficients for $\sigma = 0.01$. The GP model predicts modal coefficients that have the same magnitude as the true modal coefficients. The trajectory of GP modal coefficients starts deviating from true modal coefficients after $t \approx 10$. The hybrid framework predicts the same trajectory as the true modal coefficients. Figure 13 compares the vorticity fields reconstructed using true, GP, and hybrid modal coefficients with FOM. The GP is not able to produce the vorticity field with correct magnitude and orientation as the true projection. The hybrid framework, however, calculates the vorticity field with the same level of accuracy as the true projection. Next, we analyze the hybrid framework with a larger

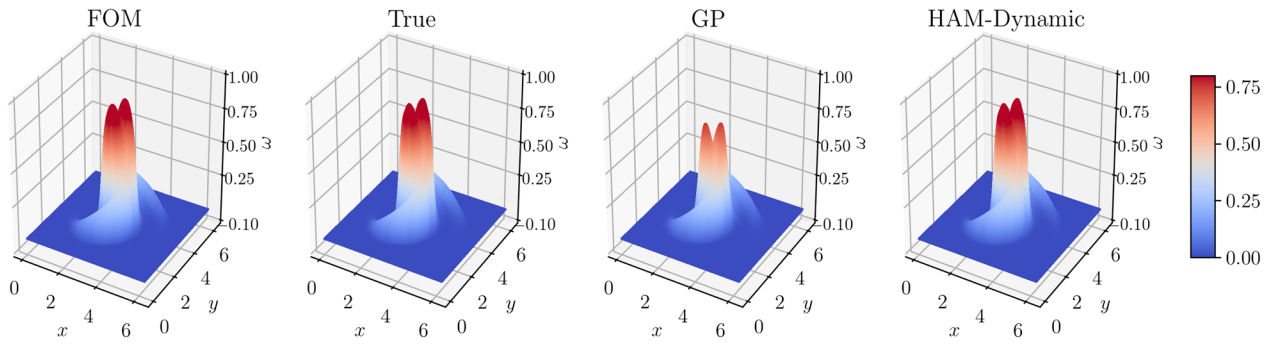


FIG. 13. Vorticity field at time $t = 20$ for the vortex-merger problem for Re = 1000 with $\sigma = 0.01$.

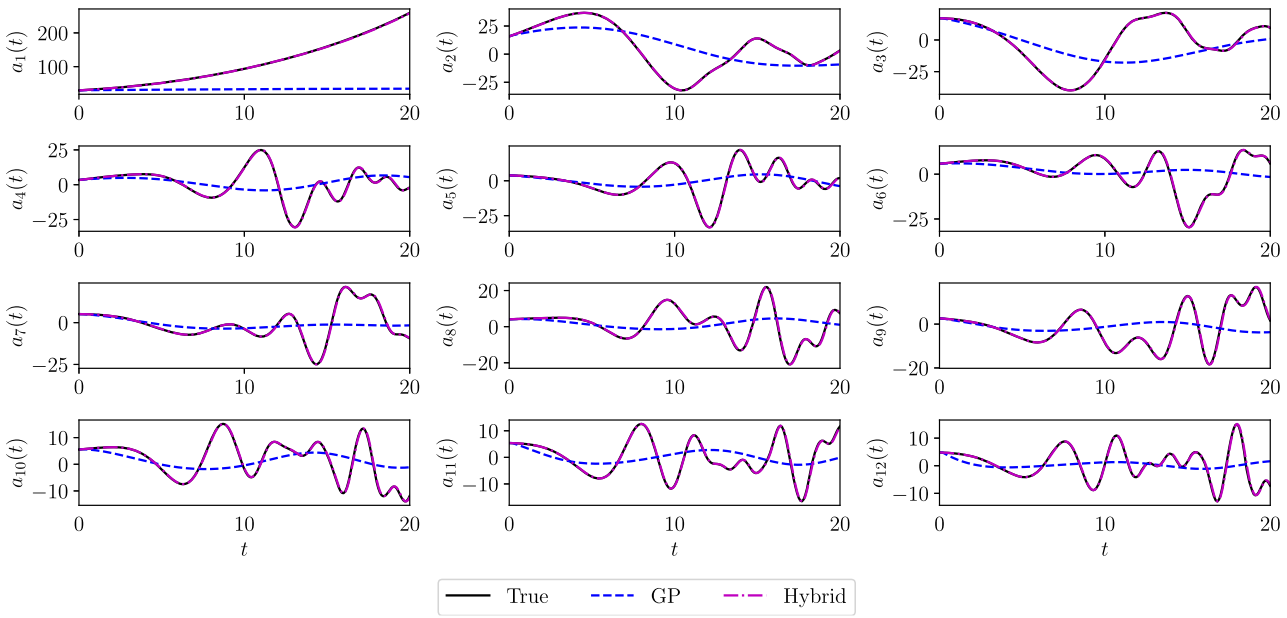


FIG. 14. Temporal evolution of vorticity modal coefficients for the vortex-merger problem for Re = 1000 with $\sigma = 0.1$.

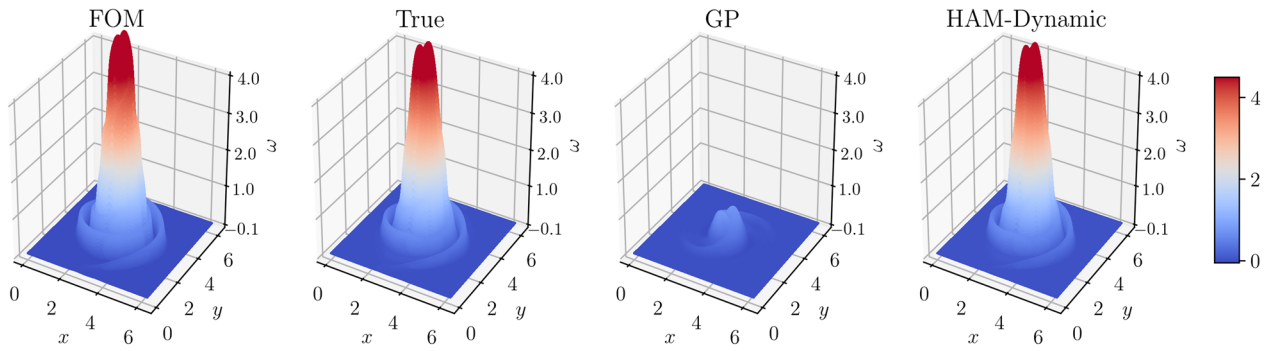


FIG. 15. Vorticity field at time $t = 20$ for the vortex-merger problem for Re = 1000 with $\sigma = 0.1$.

magnitude of friction, $\sigma = 0.1$, using the same Reynolds number. Figure 14 displays the evolution of vorticity modal coefficients for $\sigma = 0.1$. We can see a large variation in true and GP modal coefficients due to the large magnitude of the friction that is not modeled by the GP. The hybrid framework accurately predicts the vorticity modal coefficients even for a large magnitude of the friction. Figure 15 shows the vorticity field for FOM and reconstructed vorticity fields using true, GP, and hybrid modal coefficients. The magnitude of the vorticity field predicted by the GP is very small compared to the true projection as the large magnitude of hidden physics is not accounted for in the GP. The hybrid framework predicts the vorticity field with a sufficient level of accuracy as the FOM. Our findings from numerical experiments suggest that the hybrid framework can be looked into in two ways. The first way to look at the hybrid framework is as a corrector to the physical model using data-driven methods. The other way is as a means to incorporate physics into nonintrusive ROMs to make them more interpretable and generalizable.

V. DISCUSSION AND CONCLUSION

A hybrid ROM framework is presented for parameterized systems with hidden physics that uses a GP model to compute modal coefficients and a LSTM neural network to predict the unknown physics. The parameter governing the physical system is directly embedded in the GP model and the effect on the unknown physics is considered implicitly. The control parameter governing the physical system's behavior is also taken explicitly into account by including it as an input feature to the LSTM network. This type of training is found to augment the learning process making the model more generalizable.

The accuracy of the HAM framework in predicting modal coefficients and reconstructed field is illustrated for two prototypical examples: Burgers and vorticity transport equations. We examine the Burgers equation with 30% perturbation as an unknown source term. The vorticity transport equation is tested using two types of perturbation as the hidden physics, one in with decaying Taylor–Green vortices and the other in the form of linear forcing friction referring to the unknown source term. In both cases, the prediction for the interpolatory parameter by the HAM framework is shown to be considerably more accurate than only a physics-based GP model that cannot model the hidden physics. The evaluation of the HAM framework for the extrapolatory parameter also shows great agreement with the true projection and yields a result superior to the GP model. The HAM framework is found to be robust even when the magnitude of the unknown source term is almost the same or even larger than that of the physical model for both interpolatory and extrapolatory predictions.

The numerical experiments with large magnitudes of hidden physics demonstrate the effectiveness of the HAM framework for processes where there is a considerable difference between the physical model and the observed data. The HAM approach can also be interpreted as a way to inject physics or include physical models into the data-driven nonintrusive ROMs. The input features to the LSTM network in our framework are based on the GP model and, hence, the proposed approach can also be considered as a physics-informed approach rather than relying only on machine learning algorithm prediction.

The analysis of numerical experiments with two examples suggests that the hybrid modeling approaches have the potential to model a multiphysics system where there is a deviation in the physical model and the observed data. In the current study, we present the robustness of our approach using synthetic datasets generated either from an analytical solution or numerical simulation. Our approach can also be viewed as a data assimilation technique on a reduced dimensional space, which is a topic we intend to exploit further. The performance of the HAM framework in the presence of multiple source terms, noisy and realistic observed data, and complex turbulent flows will be investigated as a part of future studies. The ROMs can also be constructed for quasilimit cycle problems such as deferentially heated cavity[85] or flow past a cylinder[86] where the dynamics is to be predicted beyond the training time window. In our future studies, we will test our hybrid framework for quasilimit cycle problems and assess its predictive performance for those settings.

To sum up, in our study, we convey a hybrid modeling philosophy, arguing that one should always rely on the domain knowledge developed over decades of hard and diligent work to begin with and only switch to the blackbox pure data-driven modeling approach as a last resort. This philosophy is presented in our HAM approach for reduced order modeling of fluid flow problems involving known and unknown physics. In such a situation, the last step of blackbox modeling will not only account for the unknown physics but also for the errors resulting from the basic assumptions made to begin with. In fact, this is a powerful aspect of the approach we follow and demonstrate in the present study. In this regard, our hybrid methodology provides a way of catering to industrial needs in fluid dynamics through the generalizability, trustworthiness, computational efficiency, and dynamic adaptation of the models.

ACKNOWLEDGMENTS

This material is based upon work supported by the U.S. Department of Energy, Office of Science, Office of Advanced Scientific Computing Research under Award No. DE-SC0019290. O.S. gratefully acknowledges their support. Disclaimer: This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. The reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of the authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

REFERENCES

[1]M. Reichstein, G. Camps-Valls, B. Stevens, M. Jung, J. Denzler, N. Carvalhais *et al.*, "Deep learning and process understanding for data-driven Earth system science," Nature **566**, 195–204 (2019).

[2]A. Karpatne, G. Atluri, J. H. Faghmous, M. Steinbach, A. Banerjee, A. Ganguly, S. Shekhar, N. Samatova, and V. Kumar, "Theory-guided data science: A new paradigm for scientific discovery from data," IEEE Trans. Knowl. Data Eng. **29**, 2318–2331 (2017).

[3]S. L. Brunton, B. R. Noack, and P. Koumoutsakos, "Machine learning for fluid mechanics," Annu. Rev. Fluid Mech. **52**, 477–508 (2019).

[4]M. Brenner, J. Eldredge, and J. Freund, "Perspective on machine learning for advancing fluid mechanics," Phys. Rev. Fluids **4**, 100501 (2019).

[5]A. Rasheed, O. San, and T. Kvamsdal, "Digital twin: Values, challenges and enablers from a modeling perspective," IEEE Access **8**, 21980–22012 (2020); arXiv:1910.01719.

[6]C. W. Rowley and S. T. Dawson, "Model reduction for flow analysis and control," Annu. Rev. Fluid Mech. **49**, 387–417 (2017).

[7]K. Taira, M. S. Hemati, S. L. Brunton, Y. Sun, K. Duraisamy, S. Bagheri, S. T. Dawson, and C.-A. Yeh, "Modal analysis of fluid flows: Applications and outlook," AIAA J. (to be published).

[8]V. Puzyrev, M. Ghommem, and S. Meka, "*pyROM*: A computational framework for reduced order modeling," J. Comput. Sci. **30**, 157–173 (2019).

[9]P. Benner, S. Gugercin, and K. Willcox, "A survey of projection-based model reduction methods for parametric dynamical systems," SIAM Rev. **57**, 483–531 (2015).

[10]J.-C. Loiseau, B. R. Noack, and S. L. Brunton, "Sparse reduced-order modelling: Sensor-based dynamics to full-state estimation," J. Fluid Mech. **844**, 459–490 (2018).

[11]S. W. Fung and L. Ruthotto, "A multiscale method for model order reduction in PDE parameter estimation," J. Comput. Appl. Math. **350**, 19–34 (2019).

[12]T. Lassila and G. Rozza, "Parametric free-form shape design with PDE models and reduced basis method," Comput. Methods Appl. Mech. Eng. **199**, 1583–1592 (2010).

[13]M. Xiao, P. Breitkopf, R. F. Coelho, C. Knopf-Lenoir, M. Sidorkiewicz, and P. Villon, "Model reduction by CPOD and Kriging," Struct. Multidiscip. Optim. **41**, 555–574 (2010).

[14]M. Bergmann and L. Cordier, "Optimal control of the cylinder wake in the laminar regime by trust-region methods and POD reduced-order models," J. Comput. Phys. **227**, 7813–7840 (2008).

[15]B. R. Noack, M. Morzynski, and G. Tadmor, *Reduced-Order Modelling for Flow Control* (Springer Science and Business Media, New York, 2011), Vol. 528.

[16]S. L. Brunton and B. R. Noack, "Closed-loop turbulence control: Progress and challenges," Appl. Mech. Rev. **67**, 050801 (2015).
[17]K. Willcox, "Unsteady flow sensing and estimation via the gappy proper orthogonal decomposition," Comput. Fluids **35**, 208–226 (2006).
[18]D. N. Daescu and I. M. Navon, "Efficiency of a POD-based reduced second-order adjoint model in 4D-Var data assimilation," Int. J. Numer. Methods Fluids **53**, 985–1004 (2007).
[19]R. Ştefănescu, A. Sandu, and I. M. Navon, "POD/DEIM reduced-order strategies for efficient four dimensional variational data assimilation," J. Comput. Phys. **295**, 569–595 (2015).
[20]D. Xiao, J. Du, F. Fang, C. Pain, and J. Li, "Parameterised non-intrusive reduced order methods for ensemble Kalman filter data assimilation," Comput. Fluids **177**, 69–77 (2018).
[21]S. Guzzetti, L. M. Alvarez, P. Blanco, K. T. Carlberg, and A. Veneziani, "Propagating uncertainties in large-scale hemodynamics models via network uncertainty quantification and reduced-order modeling," Comput. Methods Appl. Mech. Eng. **358**, 112626 (2020).
[22]F. Tao, H. Zhang, A. Liu, and A. Y. Nee, "Digital twin in industry: State-of-the-art," IEEE Trans. Ind. Inf. **15**, 2405–2415 (2018).
[23]R. Ganguli and S. Adhikari, "The digital twin of discrete dynamic systems: Initial approaches and future challenges," Appl. Math. Modell. **77**, 1110–1128 (2020).
[24]D. Hartmann, M. Herz, and U. Wever, "Model order reduction a key technology for digital twins," in *Reduced-Order Modeling (ROM) for Simulation and Optimization* (Springer, 2018), pp. 167–179.
[25]S. Chakraborty, S. Adhikari, and R. Ganguli, "The role of surrogate models in the development of digital twins of dynamic systems," arXiv:2001.09292 (2020).
[26]L. Sirovich, "Turbulence and the dynamics of coherent structures. I. Coherent structures," Q. Appl. Math. **45**, 561–571 (1987).
[27]J. L. Lumley, "The structure of inhomogeneous turbulent flows," in *Atmospheric Turbulence and Radio Wave Propagation*, edited by A. M. Yaglom and V. I. Tatarsky (Nauka, Moscow, 1967), pp. 166–176.
[28]M. Sieber, C. O. Paschereit, and K. Oberleithner, "Spectral proper orthogonal decomposition," J. Fluid Mech. **792**, 798–828 (2016).
[29]M. Mendez, M. Balabane, and J.-M. Buchlin, "Multi-scale proper orthogonal decomposition of complex fluid flows," J. Fluid Mech. **870**, 988–1036 (2019).
[30]P. J. Schmid, "Dynamic mode decomposition of numerical and experimental data," J. Fluid Mech. **656**, 5–28 (2010).
[31]H. Arbabi and I. Mezic, "Ergodic theory, dynamic mode decomposition, and computation of spectral properties of the Koopman operator," SIAM J. Appl. Dyn. Syst. **16**, 2096–2126 (2017).
[32]M. S. Hemati, C. W. Rowley, E. A. Deem, and L. N. Cattafesta, "De-biasing the dynamic mode decomposition for applied Koopman spectral analysis of noisy datasets," Theor. Comput. Fluid Dyn. **31**, 349–368 (2017).
[33]D. DeMers and G. W. Cottrell, "Non-linear dimensionality reduction," in *Advances in Neural Information Processing Systems* (Morgan-Kaufmann, 1993), pp. 580–587.
[34]G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," Science **313**, 504–507 (2006).
[35]F. J. Gonzalez and M. Balajewicz, "Deep convolutional recurrent autoencoders for learning low-dimensional feature dynamics of fluid systems," arXiv:1808.01346 (2018).
[36]K. Lee and K. Carlberg, "Model reduction of dynamical systems on nonlinear manifolds using deep convolutional autoencoders," J. Comput. Phys. **404**, 108973 (2020).
[37]S. E. Otto and C. W. Rowley, "Linearly recurrent autoencoder networks for learning dynamics," SIAM J. Appl. Dyn. Syst. **18**, 558–593 (2019).
[38]D. Rempfer, "On low-dimensional Galerkin models for fluid flow," Theor. Comput. Fluid Dyn. **14**, 75–88 (2000).
[39]C. W. Rowley, T. Colonius, and R. M. Murray, "Model reduction for compressible flows using POD and Galerkin projection," Physica D **189**, 115–129 (2004).
[40]I. Akhtar, A. H. Nayfeh, and C. J. Ribbens, "On the stability and extension of reduced-order Galerkin models in incompressible flows," Theor. Comput. Fluid Dyn. **23**, 213–237 (2009).
[41]J. Borggaard, T. Iliescu, and Z. Wang, "Artificial viscosity proper orthogonal decomposition," Math. Comput. Modell. **53**, 269–279 (2011).
[42]Z. Wang, I. Akhtar, J. Borggaard, and T. Iliescu, "Proper orthogonal decomposition closure models for turbulent flows: A numerical comparison," Comput. Methods Appl. Mech. Eng. **237**, 10–26 (2012).
[43]L. Cordier, B. R. Noack, G. Tissot, G. Lehnasch, J. Delville, M. Balajewicz, G. Daviller, and R. K. Niven, "Identification strategies for model-based control," Exp. Fluids **54**, 1580 (2013).
[44]O. San and T. Iliescu, "Proper orthogonal decomposition closure models for fluid flows: Burgers equation," Int. J. Numer. Anal. Model., Ser. B **5**, 217–237 (2014).
[45]S. Grimberg, C. Farhat, and N. Youkilis, "On the stability of projection-based model order reduction for convection-dominated laminar and turbulent flows," arXiv:2001.10110 (2020).
[46]V. M. Krasnopolsky and M. S. Fox-Rabinovitz, "Complex hybrid models combining deterministic and machine learning components for numerical climate modeling and weather prediction," Neural Networks **19**, 122–134 (2006).
[47]J. Yu, C. Yan, and M. Guo, "Non-intrusive reduced-order modeling for fluid problems: A brief review," Proc. Inst. Mech. Eng., Part G **233**, 5896–5912 (2019).
[48]S. M. Rahman, S. Pawar, O. San, A. Rasheed, and T. Iliescu, "Nonintrusive reduced order modeling framework for quasigeostrophic turbulence," Phys. Rev. E **100**, 053306 (2019).
[49]S. E. Ahmed, S. Rahman, O. San, A. Rasheed, and I. M. Navon, "Memory embedded non-intrusive reduced order modeling of non-ergodic flows," Phys. Fluids **31**(12), 126602 (2019).
[50]D. Xiao, F. Fang, C. Heaney, I. Navon, and C. Pain, "A domain decomposition method for the non-intrusive reduced order modelling of fluid flow," Comput. Methods Appl. Mech. Eng. **354**, 307–330 (2019).
[51]D. Xiao, C. Heaney, L. Mottet, F. Fang, W. Lin, I. Navon, Y. Guo, O. Matar, A. Robins, and C. Pain, "A reduced order model for turbulent flows in the urban environment using machine learning," Build. Environ. **148**, 323–337 (2019).
[52]R. Swischuk, L. Mainini, B. Peherstorfer, and K. Willcox, "Projection-based model reduction: Formulations for physics-based machine learning," Comput. Fluids **179**, 704–717 (2019).
[53]D. Xiao, "Error estimation of the parametric non-intrusive reduced order model using machine learning," Comput. Methods Appl. Mech. Eng. **355**, 513–534 (2019).
[54]T. Murata, K. Fukami, and K. Fukagata, "Nonlinear mode decomposition with convolutional neural networks for fluid dynamics," J. Fluid Mech. **882**, A13 (2020).
[55]M. Guo and J. S. Hesthaven, "Reduced order modeling for nonlinear structural analysis using Gaussian process regression," Comput. Methods Appl. Mech. Eng. **341**, 807–826 (2018).
[56]K. Li, J. Kou, and W. Zhang, "Deep neural network for unsteady aerodynamic and aeroelastic modeling across multiple mach numbers," Nonlinear Dyn. **96**, 2157–2177 (2019).
[57]R. Maulik, B. Lusch, and P. Balaprakash, "Reduced-order modeling of advection-dominated systems with recurrent neural networks and convolutional autoencoders," arXiv:2002.00470 (2020).
[58]M. Schmidt and H. Lipson, "Distilling free-form natural laws from experimental data," Science **324**, 81–85 (2009).
[59]A. J. Majda, C. Franzke, and D. Crommelin, "Normal forms for reduced stochastic climate models," Proc. Natl. Acad. Sci. U. S. A. **106**, 3649–3653 (2009).
[60]S. H. Rudy, S. L. Brunton, J. L. Proctor, and J. N. Kutz, "Data-driven discovery of partial differential equations," Sci. Adv. **3**, e1602614 (2017).
[61]Z. Long, Y. Lu, and B. Dong, "PDE-Net 2.0: Learning PDEs from data with a numeric-symbolic hybrid deep network," J. Comput. Phys. **399**, 108925 (2019).
[62]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**, 281–302 (2019).
[63]R. Iten, T. Metger, H. Wilming, L. Del Rio, and R. Renner, "Discovering physical concepts with neural networks," Phys. Rev. Lett. **124**, 010508 (2020).
[64]M. Raissi and G. E. Karniadakis, "Hidden physics models: Machine learning of nonlinear partial differential equations," J. Comput. Phys. **357**, 125–141 (2018).

[65]H. Vaddireddy, A. Rasheed, A. E. Staples, and O. San, "Feature engineering and symbolic regression methods for detecting hidden physics from sparse sensor observation data," Phys. Fluids **32**, 015113 (2020).
[66]J. Kou and W. Zhang, "Multi-fidelity modeling framework for nonlinear unsteady aerodynamics of airfoils," Appl. Math. Modell. **76**, 832–855 (2019).
[67]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**, 686–707 (2019).
[68]M. Bode, M. Gauding, Z. Lian, D. Denker, M. Davidovic, K. Kleinheinz, J. Jitsev, and H. Pitsch, "Using physics-informed super-resolution generative adversarial networks for subgrid modeling in turbulent reactive flows," arXiv:1911.11380 (2019).
[69]J. Ling, A. Kurzawski, and J. Templeton, "Reynolds averaged turbulence modelling using deep neural networks with embedded invariance," J. Fluid Mech. **807**, 155–166 (2016).
[70]T. Beucler, M. Pritchard, S. Rasp, P. Gentine, J. Ott, and P. Baldi, "Enforcing analytic constraints in neural-networks emulating physical systems," arXiv:1909.00912 (2019).
[71]O. San and R. Maulik, "Neural network closures for nonlinear model order reduction," Adv. Comput. Math. **44**, 1717–1750 (2018).
[72]R. Ibáñez, E. Abisset-Chavanne, D. González, J.-L. Duval, E. Cueto, and F. Chinesta, "Hybrid constitutive modeling: Data-driven learning of corrections to plasticity models," Int. J. Mater. Form. **12**, 717–725 (2019).
[73]S. Pan and K. Duraisamy, "Physics-informed probabilistic learning of linear embeddings of non-linear dynamics with guaranteed stability," SIAM J. Appl. Dyn. Syst. **19**(1), 480–509 (2020).
[74]C. Mou, H. Liu, D. R. Wells, and T. Iliescu, "Data-driven correction reduced order models for the quasi-geostrophic equations: A numerical investigation," Int. J. Comput. Fluid Dyn. 1–13 (2020).
[75]N. Muralidhar, J. Bu, Z. Cao, L. He, N. Ramakrishnan, D. Tafti, and A. Karpatne, "Physics-guided design and learning of neural networks for predicting drag force on particle suspensions in moving fluids," arXiv:1911.04240 (2019).
[76]O. San, R. Maulik, and M. Ahmed, "An artificial neural network framework for reduced order modeling of transient flows," Commun. Nonlinear Sci. Numer. Simul. **77**, 271–287 (2019).
[77]Z. Y. Wan, P. Vlachas, P. Koumoutsakos, and T. Sapsis, "Data-assisted reduced-order modeling of extreme events in complex dynamical systems," PLoS One **13**, e0197704 (2018).
[78]Z. Wang, D. Xiao, F. Fang, R. Govindan, C. C. Pain, and Y. Guo, "Model identification of reduced order fluid dynamics systems using deep learning," Int. J. Numer. Methods Fluids **86**, 255–268 (2018).
[79]T. Lassila, A. Manzoni, A. Quarteroni, and G. Rozza, "Model order reduction in fluid dynamics: Challenges and perspectives," in *Reduced Order Methods for Modeling and Computational Reduction* (Springer, 2014), pp. 235–273.
[80]D. Amsallem and C. Farhat, "Interpolation method for adapting reduced-order models and application to aeroelasticity," AIAA J. **46**, 1803–1813 (2008).
[81]R. Zimmermann, B. Peherstorfer, and K. Willcox, "Geometric subspace updates with applications to online adaptive nonlinear model reduction," SIAM J. Matrix Anal. Appl. **39**, 234–261 (2018).
[82]N. B. Erichson, L. Mathelin, Z. Yao, S. L. Brunton, M. W. Mahoney, and J. N. Kutz, "Shallow learning for fluid flow reconstruction with limited sensors and limited data," arXiv:1902.07358 (2019).
[83]O. San, A. E. Staples, and T. Iliescu, "Approximate deconvolution large eddy simulation of a stratified two-layer quasigeostrophic ocean model," Ocean Modell. **63**, 1–20 (2013).
[84]O. San and A. E. Staples, "An efficient coarse grid projection method for quasi-geostrophic models of large-scale ocean circulation," Int. J. Multiscale Comput. Eng. **11**, 463–495 (2013).
[85]S. Pawar, S. Rahman, H. Vaddireddy, O. San, A. Rasheed, and P. Vedula, "A deep learning enabler for nonintrusive reduced order modeling of fluid flows," Phys. Fluids **31**, 085101 (2019).
[86]M. Mohebujjaman, L. G. Rebholz, and T. Iliescu, "Physically constrained data-driven correction for reduced-order modeling of fluid flows," Int. J. Numer. Methods Fluids **89**, 103–122 (2019).