Physica D 457 (2024) 133938

Contents lists available at ScienceDirect

Physica D

journal homepage: www.elsevier.com/locate/physd

Probabilistic forecast of nonlinear dynamical systems with uncertainty quantification

Mengyang Gu [a,*,1], Yizi Lin [a], Victor Chang Lee [b], Diana Y. Qiu [b]

[a] Department of Statistics and Applied Probability, University of California, Santa Barbara, 93106, CA, USA
[b] Department of Mechanical Engineering and Materials Sciences, Yale University, New Haven, 06520, CT, USA

ARTICLE INFO

Communicated by Dmitry Pelinovsky

Dataset link: https://github.com/UncertaintyQuantification/forecast_dynamical_systems

Keywords:
Bayesian prior
Generative models
Dynamic mode decomposition
Forecast
Gaussian processes
Uncertainty quantification

ABSTRACT

Data-driven modeling is useful for reconstructing nonlinear dynamical systems when the underlying process is unknown or too expensive to compute. Having reliable uncertainty assessment of the forecast enables tools to be deployed to predict new scenarios unobserved before. In this work, we first extend parallel partial Gaussian processes for predicting the vector-valued transition function that links the observations between the current and next time points, and quantify the uncertainty of predictions by posterior sampling. Second, we show the equivalence between the dynamic mode decomposition and the maximum likelihood estimator of the linear mapping matrix in the linear state space model. The connection provides a probabilistic generative model of dynamic mode decomposition and thus, uncertainty of predictions can be obtained. Furthermore, we draw close connections between different data-driven models for approximating nonlinear dynamics, through a unified view of generative models. We study two numerical examples, where the inputs of the dynamics are assumed to be known in the first example and the inputs are unknown in the second example. The examples indicate that uncertainty of forecast can be properly quantified, whereas model or input misspecification can degrade the accuracy of uncertainty quantification.

1. Introduction

Dynamical systems are ubiquitously used for describing natural phenomena, such as passive motions driven by thermodynamics [1] and phase transition from flocking [2,3], and social behaviors, such as epidemiological processes [4]. As mathematical models typically contain unknown parameters, observations are often used for calibrating the models and filtering the noises to estimate the latent state of the dynamical system. Kalman filter [5] and Rauch–Tung–Striebel smoother [6], for instance, produce fast estimation of latent states for linear dynamical systems with additive Gaussian fluctuations and noises, at a computational cost linearly increasing with the number of time points. When dynamical systems are nonlinear or non-Gaussian, approximate approaches, such as extended Kalman filter [7], particle filters [8] and ensemble Kalman filter [9], were developed to approximate the posterior distributions of latent states. However, these approaches require underlying data-generating models to be known, whereas models that exactly reproduce the reality may be unavailable or too costly to compute in some applications.

Data-driven approaches become useful for estimating dynamical systems when the true data-generating mechanism is unknown. For instance, orthogonal basis is estimated in proper orthogonal decomposition [10,11] to reconstruct the covariance between each of the output coordinates by treating temporal observations as independent measurements. Dynamic mode decomposition [12,13] reconstructs the output vector through linearizing the one-step-ahead transition operator between the input and output pairs, where the eigenpairs of the linear mapping matrix produce a finite-dimensional approximation of the Koopman modes and eigenvalues [14,15]. Extensive variants of Koopman operator have been proposed, such as utilizing longer temporal lag of observations through Hankel method or higher order dynamic mode decomposition [16], and utilizing nonlinear basis functions for lifting the process by the extended dynamic mode decomposition [17]. A few recent techniques, such as sparse regression [18,19], model predictive control [20], and Koopman eigenfunctions [21], were studied for designing the nonlinear basis and estimating the lifted state in extended dynamic mode decomposition. Other nonlinear methods employ neural networks to model differential equations [22–24]. The uncertainty of the estimation by the dynamic mode decomposition and other machine learning approaches, however, may not be available, as the generative models were not well-studied.

* Corresponding author.
E-mail addresses: mengyang@pstat.ucsb.edu (M. Gu), diana.qiu@yale.edu (D.Y. Qiu).
[1] Equal contributions between the first two authors.

https://doi.org/10.1016/j.physd.2023.133938
Received 15 May 2023; Received in revised form 23 August 2023; Accepted 26 September 2023
Available online 10 October 2023

Deploying data-driven models to forecast or extrapolate the input space requires reliable uncertainty assessments of predictions. We evaluate the precision of uncertainty quantification by the percentage of held-out observations covered by the $1-\alpha$ predictive intervals, with $0<\alpha<1$. An efficient approach should have around $1-\alpha$ of the held-out samples covered by a short predictive interval. Gaussian processes have been applied in emulating expensive computer simulations [25–27], and calibrating computer models [28–30]. However, uncertainty assessment of these approaches is typically assessed for interpolating physical input space, while forecast or extrapolation is required in various real-world applications, such as optimizing chemical reaction conditions through Bayesian optimization [31] and controlling the predictive error by active learning [32].

The goal of this paper is to quantify the uncertainty from probabilistic forecasts by different approaches. Our contributions are three-fold. First, we extend a recent approach, called the parallel partial Gaussian process (PP-GP) [33], to forecast dynamical systems with multivariate outputs through predicting the one-step-ahead transition function, and the uncertainty of the forecast can be assessed through posterior sampling. Predicting the one-step-ahead transition function with a finite-dimensional space allows one to transform the forecast problem to an interpolation problem, and under regularity conditions, Gaussian process regression converges to the truth with a minimax rate [34]. Furthermore, the assessed uncertainty from the PP-GP can alert when the forecast becomes less accurate, which allows for timely interventions to control predictive error. The PP-GP is particularly suitable for dynamical systems with massive outputs, as the computational complexity of the PP-GP is linear to the number of outputs at each time point. Second, we introduce a general two-step approach to derive a probabilistic generative model. Based on this approach, we show the estimation of dynamic mode decomposition is equivalent to the maximum likelihood estimator of a linear mapping matrix in a linear state space model, which produces uncertainty assessment from the sampling model. Third, we draw connections between different approaches, including Gaussian processes, proper orthogonal decomposition and dynamic mode decomposition. These connections allow one to examine the inherent generative models of different approaches, and to develop a suitable predictive model for real-world problems.

We compare the approaches for forecasting and uncertainty quantification by two numerical examples. In the first example, we assume the inputs of the process are known, whereas we do not have prior knowledge of the functional form of the process. Hence we cannot use the exact form of the function to form the nonlinear basis. Rather we aim to test default or generic kernels or basis, which can be used for other scenarios. We test this scenario by the Lorenz 96 system [35], a benchmark approach of modeling atmospheric quantities at equally spaced locations along a cycle that induces chaotic behaviors. The PP-GP can detect the time when the predictive error becomes large, based on its internal uncertainty assessment of the forecast.

In the second example, we do not assume the true inputs are known, which is unconventional in designing data-driven approaches, but not uncommon in practice [36]. We consider one of the most challenging problems in condensed matter physics: simulating quantum many-body systems far from equilibrium. Many problems in quantum dynamics, such as the motion of atoms or charge carriers, cannot be modeled by well-established equilibrium methods, including density functional theory (DFT) for the electronic ground state or the GW plus Bethe–Salpeter equation [37,38] which describes excited-state properties in the equilibrium linear response regime. This limits, for example, the understanding of systems under irradiation by ultrafast or intense laser pulses for obtaining information about the electronic structure of a system [39]. Therefore, nonequilibrium simulations that describe how a system responds to an external perturbation and how it evolves from one configuration to another under these circumstances are crucial for a complete understanding of electronic and optical properties of molecules and solids.

A rigorous approach to simulating materials' nonequilibrium dynamics lies in propagating the nonequilibrium Green's function as a two-point correlator of the creation and annihilation field operators on the Keldysh contour [40–42]. This approach has been recently applied to compute various nonlinear and nonequilibrium optical responses from first principles in the adiabatic limit, which limits the time-evolution to a single average time, neglecting memory effects [43–47]. However, even in the adiabatic approximation, the numerical evaluation is far from trivial, requiring millions of CPU hours for systems of only a few atoms. Thus, models that can forecast the time-evolution without the need of the simulator are urgently needed. Recent work [48,49] uses dynamic mode decomposition to approximate the Green's functions where the system is assumed to start from a known non-interacting state, and it is driven by an arbitrary external electromagnetic field. Different representations of the Green's function encode spectroscopic information, which may be measured in experiments. In this work, inputs such as the external field and many-electron interactions are not used in constructing data-driven models to test uncertainty quantification of forecast for the scenario when inputs are misspecified.

The article is organized as follows. In Section 2, we extended the PP-GP for forecasting nonlinear dynamical processes. We introduce a general approach to derive a generative model, and apply this approach to derive a sampling model of the dynamic mode decomposition and its predictive distribution in Section 3. PP-GP, dynamic mode decomposition and proper orthogonal decomposition are compared in Section 4, focusing on generative models of these approaches. In Section 5, we numerically compare different approaches for forecast, and discuss the scenarios when reliable forecast can be constructed even at reasonably long trajectory. We conclude this study and outline future directions in Section 6. The data and code used in this work are publicly available (https://github.com/UncertaintyQuantification/forecast_dynamical_systems).

2. Probabilistic forecast and uncertainty quantification through parallel partial Gaussian processes

The parallel partial Gaussian process (PP-GP) emulator was originally designed as a fast surrogate model to approximate computationally expensive computer models with massive observations [33]. Emulating computer models typically starts with running the computer simulation at a set of 'space-filling' designs, such as Latin hypercube designs [25], for building the emulator. For any other inputs untested before, the predictive distribution of the emulator is used for predictions and quantifying the uncertainty of the predictions. Most of the computer model emulation tasks deal with *interpolation* for a design space, meaning that the distance between the test input and some training inputs is close, as the 'space-filling' inputs fill the input space. However, many scientific tasks, such as designing a new molecule or forecasting dynamical systems, inevitably require *extrapolation* from the existing design space, where reliable uncertainty quantification of the predictions is needed. Here we extend the PP-GP model for forecasting nonlinear dynamical systems that enables uncertainty of the forecast to be quantified in a probabilistic way, which was not studied before.

Suppose we have collected n vectors of real-valued outputs or snapshots, each having m dimensions, where the tth output vector is denoted as $\mathbf{y}(\mathbf{x}_t) = (y_1(\mathbf{x}_t), \ldots, y_m(\mathbf{x}_t))^T$, with $\mathbf{x}_t$ being a p-dimensional input that contains the observations in the prior time points and additional physics input, for $t = 1, \ldots, n$. When the observational vector in the prior time point is used as the input, we have $\mathbf{x}_t = \mathbf{y}(\mathbf{x}_{t-1})$ and thus $p = m$. In general, the dimensions of the input and output vectors can be different.

In the PP-GP model, we assume a distinct mean parameter μ_j and variance parameter σ_j^2 at each coordinate of the output $y_j(\cdot)$, for $j = 1, \ldots, m$, which makes it flexible to capture the scale difference in output coordinates. The correlation of the output at any two coordinates j and

j', on the other hand, is assumed to be the same, i.e. $\mathrm{Cor}(y_j(\mathbf{x}), y_j(\mathbf{x}')) = \mathrm{Cor}(y_{j'}(\mathbf{x}), y_{j'}(\mathbf{x}'))$, which greatly simplifies the computation. The observations from dynamical systems may contain noises, due to numerical or measurement errors. Thus, for any input $\mathbf{x}$, we define the PP-GP model of the jth coordinate below:

$$y_j(\mathbf{x}) = f_j(\mathbf{x}) + \epsilon_j, \tag{1}$$

where $\mathbf{x}$ is p-dimensional input variable, $f_j(\cdot)$ follows a Gaussian process prior with mean μ_j and covariance $\sigma_j^2 K(\cdot,\cdot)$, and ϵ_j is an independent Gaussian noise with variance $\eta\sigma_j^2$. For any $p \times n$ input matrix $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_n]$, integrating the latent Gaussian process $f_j(\cdot)$, the marginal distribution $\mathbf{y}_j = [y_j(\mathbf{x}_1), \ldots, y_j(\mathbf{x}_n)]^T$ follows a multivariate normal distribution:

$$(\mathbf{y}_j \mid \mathbf{X}, \mu_j, \sigma_j^2, \eta, \boldsymbol{\gamma}) \sim \mathcal{MN}\left(\mu_j \mathbf{1}_n, \sigma_j^2 \tilde{\mathbf{K}}\right), \tag{2}$$

where $\mathcal{MN}$ denotes the multivariate normal distribution, μ_j is an unknown mean parameter, $\tilde{\mathbf{K}} = (\mathbf{K} + \eta \mathbf{I}_n)$ with $\mathbf{K}$ being a correlation matrix and η being a nugget parameter due to noises in observations. The (t,t')th entry of $\mathbf{K}$ follows $K_{t,t'} = K(\mathbf{x}_t, \mathbf{x}_{t'})$ with $K(\cdot,\cdot)$ being a kernel function containing a $\tilde{p}$-dimensional range parameter $\boldsymbol{\gamma}$, and $\mathbf{I}_n$ denotes an identity matrix. With a nugget parameter, the prediction of GP will be dragged towards to the mean compared to a noise-free GP [50]. Additional trend or mean basis functions can be included in the mean of the PP-GP model.

Frequently used covariance functions include isotropic covariance and product covariance [25]. The isotropic covariance is a function of Euclidean distance between two inputs: $d = \|\mathbf{x} - \mathbf{x}'\|$ with $\|\cdot\|$ denoting the L_2 norm. For instance, the isotropic power exponential covariance function follows

$$\sigma_j^2 K(d) = \sigma_j^2 \exp\left(-\frac{d^\alpha}{\gamma}\right), \tag{3}$$

where the roughness parameter $0 < \alpha \le 2$ is typically held fixed and γ is a positive range parameter controlling the correlation length, which is often estimated by data. Here the number of range parameters is 1, i.e. $\tilde{p} = 1$.

Another widely used isotropic covariance is the Matérn covariance [51]:

$$\sigma_j^2 K(d) = \sigma_j^2 \frac{2^{1-\alpha}}{\Gamma(\alpha)} \left(\frac{\sqrt{2\alpha} d}{\gamma}\right)^\alpha \mathcal{K}_\alpha \left(\frac{\sqrt{2\alpha} d}{\gamma}\right), \tag{4}$$

where $d = \|\mathbf{x} - \mathbf{x}'\|$ is the distance between inputs, $\Gamma(\cdot)$ is the gamma function and $\mathcal{K}_\alpha(\cdot)$ is the modified Bessel function of the second kind with a positive parameter α. The Matérn covariance has a closed-form expression with an integer roughness parameter $\alpha = \frac{2z+1}{2}$ with $z \in \mathbb{N}$. When $\alpha = 2.5$, for example, the Matérn covariance function has the following expression

$$\sigma_j^2 K(d) = \sigma_j^2 \left(1 + \frac{\sqrt{5} d}{\gamma} + \frac{5d^2}{3\gamma^2}\right) \exp\left(-\frac{\sqrt{5} d}{\gamma}\right). \tag{5}$$

The GP model having a Matérn covariance with a roughness parameter α is $\lfloor \alpha - 1 \rfloor$ mean squared differentiable, an appealing property as the smoothness of the process is directly controlled by the roughness parameter α.

When the input variables have different scales, a product correlation function is more frequently used, as it allows one to have a distinct correlation length parameter for each of the input coordinates:

$$K(\mathbf{x}, \mathbf{x}') = \prod_{l=1}^{p} K_l(x_l, x_l'), \tag{6}$$

where $K_l(x_l, x_l')$ is a kernel function, such as power exponential kernel or Matérn kernel, with range parameter γ_l for $l = 1, \ldots, p$ and we have $\tilde{p} = p$ range parameters. The product form of the kernel in Eq. (6) is widely used for computer model emulation [52–54] and often treated as the default setting in statistical emulator software packages [55, 56], as inputs variable can contain completely different scales and physical meanings, thus requiring distinct correlation lengthscales. In practice, isotropic covariance may be used when Euclidean distance is meaningful for characterizing the distance between two inputs. The product covariance may yield better predictive performance, whereas more range parameters are needed to be estimated.

In the PP-GP model, we have m mean parameters $\boldsymbol{\mu} = (\mu_1, \ldots, \mu_m)^T$, m variance parameters $\boldsymbol{\sigma}^2 = (\sigma_1^2, \ldots, \sigma_m^2)^T$, $\tilde{p}$ covariance range parameters $\boldsymbol{\gamma} = (\gamma_1, \ldots, \gamma_{\tilde{p}})^T$ and a nugget parameter η. We follow the Bayesian procedure to define the prior of parameters. As most of the parameters can be integrated out explicitly, meaning that the uncertainty from the estimates of these parameter is quantified during the Bayesian inference with almost no extra computational cost, whereas a plug-in estimator of the parameters may ignore the uncertainty from parameter estimation. We assume an objective Bayesian prior [33,57] of the parameters below

$$\pi(\boldsymbol{\mu}, \boldsymbol{\sigma}^2, \boldsymbol{\gamma}, \eta) \propto \frac{\pi(\boldsymbol{\gamma}, \eta)}{\prod_{j=1}^m \sigma_j^2}, \tag{7}$$

where $\pi(\boldsymbol{\gamma}, \eta)$ is a prior of the range and nugget parameters. Denote the $m \times n$ matrix of observations by $\mathbf{Y} = [\mathbf{y}(\mathbf{x}_1), \ldots, \mathbf{y}(\mathbf{x}_n)]$. Integrating out the mean and variance parameters, the predictive distribution for $\mathbf{x}_{t^*}$ follows a noncentral Students' t distributions with $n-1$ degrees of freedom [33]:

$$\left(y_j\left(\mathbf{x}_{t^*}\right) \mid \mathbf{X}, \mathbf{Y}, \mathbf{x}_{t^*}, \boldsymbol{\gamma}, \eta\right) \sim \mathcal{T}\left(\hat{y}_j\left(\mathbf{x}_{t^*}\right), \hat{\sigma}_j^2 K^*, n-1\right), \tag{8}$$

where the predictive mean and scale parameters follow

$$\hat{y}_j\left(\mathbf{x}_{t^*}\right) = \hat{\mu}_j + \mathbf{k}^T(\mathbf{x}_{t^*}) \tilde{\mathbf{K}}^{-1}\left(\mathbf{y}_j - \hat{\mu}_j \mathbf{1}_n\right), \tag{9}$$

$$\hat{\sigma}_j^2 = \frac{1}{n-1}\left(\mathbf{y}_j - \hat{\mu}_j \mathbf{1}_n\right)^T \tilde{\mathbf{K}}^{-1}\left(\mathbf{y}_j - \hat{\mu}_j \mathbf{1}_n\right), \tag{10}$$

$$K^* = 1 + \eta - \mathbf{k}^T(\mathbf{x}_{t^*}) \tilde{\mathbf{K}}^{-1} \mathbf{k}(\mathbf{x}_{t^*}) + \frac{\left(1 - \mathbf{1}_n^T \tilde{\mathbf{K}}^{-1} \mathbf{k}(\mathbf{x}_{t^*})\right)^2}{\mathbf{1}_n^T \tilde{\mathbf{K}}^{-1} \mathbf{1}_n}, \tag{11}$$

with $\tilde{\mathbf{K}} = \mathbf{K} + \eta \mathbf{I}_n$, $\hat{\mu}_j = \left(\mathbf{1}_n^T \tilde{\mathbf{K}}^{-1} \mathbf{1}_n\right)^{-1} \mathbf{1}_n^T \tilde{\mathbf{K}}^{-1} \mathbf{y}_j$ being the generalized least square estimator of the mean, $\mathbf{1}_n$ is an n-vector of ones and $\mathbf{k}(\mathbf{x}_{t^*}) = (K(\mathbf{x}_1, \mathbf{x}_{t^*}), \ldots, K(\mathbf{x}_n, \mathbf{x}_{t^*}))^T$ being an n-vector of the covariance between the training inputs and the test input.

The PP-GP model has been implemented in different computational platforms such as MATLAB, Python and R [56]. The predictive mean from distribution in Eq. (8) is typically used for prediction. The uncertainty of the prediction can be quantified by the predictive interval. A few recent studies approximate the transition operator of dynamical systems through kernel flows [58,59]. In comparison, the PP-GP provides a distinct mean and variance for each coordinates, and these parameters are integrated out for calculating the predictive distribution in Eq. (8), making the model more flexible in uncertainty assessment. We will discuss the computational issue and compare PP-GP with other vector-valued GP approaches in Section 2.3.

2.1. Predictions as weighted averages of basis functions and output vectors

The predictive mean or median $\hat{y}_j(\mathbf{x}_{t^*})$ is often used for one-step-ahead prediction of output coordinate j for any test input $\mathbf{x}_{t^*}$, for $j = 1, \ldots, m$. The corollary below shows that the prediction of the PP-GP model can be written as a weighted average of the observations and kernel functions.

Corollary 1. *The predictive mean vector* $\hat{\mathbf{y}}(\mathbf{x}_{t^*}) = (\hat{y}_1(\mathbf{x}_{t^*}), \ldots, \hat{y}_m(\mathbf{x}_{t^*}))^T$ *from Eq. (9) follows*

$$\hat{\mathbf{y}}(\mathbf{x}_{t^*}) = \mathbf{Y}\mathbf{v}^T = \hat{\boldsymbol{\mu}} + \mathbf{W}\mathbf{k}(\mathbf{x}_{t^*}), \tag{12}$$

where $\mathbf{v}$ *is an* n*-dimensional row vector and* $\mathbf{W} = [\mathbf{w}_1^T, \ldots, \mathbf{w}_m^T]^T$ *is a* $m \times n$ *matrix with* $\mathbf{w}_j$ *being an* n*-dimensional row vector defined below:*

$$\mathbf{v} = \frac{(1 - \mathbf{k}^T(\mathbf{x}_{t^*}) \tilde{\mathbf{K}}^{-1} \mathbf{1}_n) \mathbf{1}_n^T \tilde{\mathbf{K}}^{-1}}{\left(\mathbf{1}_n^T \tilde{\mathbf{K}}^{-1} \mathbf{1}_n\right)} + \mathbf{k}^T(\mathbf{x}_{t^*}) \tilde{\mathbf{K}}^{-1}, \tag{13}$$

$$\mathbf{w}_j = \left(\mathbf{y}_j - \hat{\mu}_j \mathbf{1}_n\right)^T \tilde{\mathbf{K}}^{-1}, \tag{14}$$

for $j = 1, \dots, m$.

From Corollary 1, the prediction of a test input at the jth coordinate of the output from the PP-GP model can be written as a weighted average of the observations at the jth coordinate, $\hat{y}_j(\mathbf{x}_{t^*}) = \mathbf{v}\mathbf{y}_j$. Furthermore, the residuals can be written as a weighted average of the kernel function between the test input and training input set, $\hat{y}_j(\mathbf{x}_{t^*}) - \hat{\mu}_j = \mathbf{w}_j \mathbf{k}(\mathbf{x}_{t^*})$, as outlined by the second equality in Eq. (12). When $\hat{\mu}_j = 0$, the predictive mean estimator in Eq. (9) at each output coordinate is equivalent to the kernel ridge regression separately for each coordinate j [60]:

$$\hat{y}_j(\cdot) = \text{argmin}_{f_j \in \mathcal{H}} \left\{ \frac{1}{n} \sum_{t=1}^{n} (y_j(\mathbf{x}_t) - f_j(\mathbf{x}_t))^2 + \frac{\eta}{n} \|f_j\|_{\mathcal{H}}^2 \right\}, \tag{15}$$

where $\mathcal{H}$ is the reproducing kernel Hilbert space (RKHS) [61] attached to the kernel $K(\cdot,\cdot)$ and $\|\cdot\|_{\mathcal{H}}$ is the associated native norm. The loss function in Eq. (15) penalizes the fitting error and complexity of the model simultaneously, which helps avoid the overfitting problem automatically. Compared to the kernel ridge regression in Eq. (15), the uncertainty of the prediction of PP-GP can be quantified based on the predictive distribution in Eq. (8).

Here the range and nugget parameters $(\boldsymbol{\gamma}, \eta)$ can be estimated by maximum marginal posterior distribution described in the Appendix and then these parameters will be plugged into Eq. (8) for computing the predictive distribution. Here the uncertainty of the mean and variance parameters are taken into account in the analysis, whereas the uncertainty in estimating the range and nugget parameters was not considered due to computational feasibility, and confounding issues between kernel parameters [62]. Sampling the parameters from the posterior distribution or residual bootstrap approach [63] can be used for estimating the uncertainty of these kernel parameters.

2.2. Forecast by parallel partial Gaussian processes

Here we focus on estimating the one-step-ahead transition function that maps the input $\mathbf{x}_t$ to the output $\mathbf{y}(\mathbf{x}_t)$ at any time point t. Consider a simple scenario, where the previous snapshot is used for predicting the output at the current time step: $\mathbf{x}_t = \mathbf{y}(\mathbf{x}_{t-1})$ for any $t \geq 1$. Since the function is nonlinear, we may iteratively use the predictive distribution to sample S chains for forecasting from $t^* = n+1, \dots, n+n^*$. Let the input be $\mathbf{x}_{n+1}^{(s)} = \mathbf{y}(\mathbf{x}_n)$ for any chain s, $s = 1, \dots, S$. For each of the chain s, we simulate a new output from the predictive distribution sequentially for $t^* = n+1, \dots, n+n^*$:

$$\mathbf{y}^{(s)}(\mathbf{x}_{t^*+1}^{(s)}) \sim p(\mathbf{y}^{(s)}(\mathbf{x}_{t^*+1}^{(s)}) \mid \mathbf{Y}, \mathbf{X}, \mathbf{x}_{t^*+1}^{(s)}, \boldsymbol{\gamma}, \eta), \tag{16}$$

where $p(\mathbf{y}^{(s)}(\mathbf{x}_{t^*+1}^{(s)}) \mid \mathbf{Y}, \mathbf{X}, \mathbf{x}_{t^*+1}^{(s)}, \boldsymbol{\gamma}, \eta)$ is the predictive distribution of $\mathbf{y}^{(s)}(\mathbf{x}_{t^*+1})$.

As directly sampling from the joint predictive distribution at all output coordinates can be computationally intensive, one may sample the jth coordinate of the output from the marginal predictive distribution $p(y_j^{(s)}(\mathbf{x}_{t^*+1}) \mid \mathbf{Y}, \mathbf{X}, \mathbf{x}_{t^*+1}^{(s)}, \boldsymbol{\gamma}, \eta)$ in Eq. (8) as an approximation. After we obtain predictive samples $y_{t^*,j}^{(s)}$ for $s = 1, \dots, S$, we can use mean or median for predictions, and the lower and upper α quantiles of the samples for constructing the $1-\alpha$ predictive interval for any $0 < \alpha < 1$. Furthermore, the predictive mean $\hat{\mathbf{y}}(\mathbf{x}_{t^*+1})$ may be approximated by using a plug-in estimator of the input $\mathbf{x}_{t^*+1} \approx \hat{\mathbf{y}}(\mathbf{x}_{t^*})$ by Eq. (9). The assessed uncertainty from PP-GP can alert when the forecast becomes less accurate, which allows for timely interventions to control the predictive error, whereas the uncertainty assessment may not be not available in some other machine learning approaches [18,22,23].

Here we transform the forecast problem to predict the one-step-ahead transition function with a finite-dimensional input space. Under regularity conditions, the minimax convergence rate of Gaussian process regression to the truth was studied [34]. We should note that convergence is obtained when the training inputs fill a bounded input domain, whereas sequentially sampled inputs in forecast could move outside the training input domain in practice. When the inputs approach the boundary of input space, the quantified uncertainty becomes large, which can give alert for refining the prediction by expanding the input domain and training the PP-GP emulator with inputs in the expanded input domain. It is of interest to study the convergence of PP-GP forecast in different dynamical systems.

2.3. Computational complexity

Compared with other GP approaches for emulating vectorized output, one advantage of the PP-GP model comes with the computational scalability when the number of output coordinates m is large. Computing the predictive mean of m output coordinates in Eq. (9) requires $\mathcal{O}(nm) + \mathcal{O}(n^3)$ operations, and to obtain S predictive samples of m output vectors at n^* time points for uncertainty quantification requires $\mathcal{O}(n^2 n^* S) + \mathcal{O}(n^2 m)$ operations. The largest cost of PP-GP typically comes from estimating the range and nugget parameters, which requires $\mathcal{O}(\tilde{S}n^3 + \tilde{S}n^2 m)$ operations for $\tilde{S}$ iterations in numerical optimization. When the number of time points n is large, approximation methods, such as the inducing point method [64] and the Vecchia approach [65], may be used for approximating the likelihood function of Gaussian processes.

The computational advantage of PP-GP comes from two assumptions. First, the outputs at different coordinates are assumed to be independent. In Theorem 1 in [33], the authors show that the predictive mean of PP-GP is exactly the same as the predictive mean of a separable Gaussian process of the vectorized output, with the covariance $\boldsymbol{\Sigma}_y \otimes \mathbf{K}$, where $\boldsymbol{\Sigma}_y$ is the covariance of output at different coordinates, and the variance between the two models is similar. The inverse of covariance between output coordinates $\boldsymbol{\Sigma}_y$ generally takes $\mathcal{O}(m^3)$ operations in computing the likelihood function of separable Gaussian process, whereas the complexity of predictions by PP-GP is linear to the number of coordinates (m). Second, the correlation of the output at different inputs $\mathbf{x}$ is shared across output coordinates. Allowing the correlation parameters to differ at each spatial coordinate makes the computational complexity become $\mathcal{O}(n^3 m)$ for computing the predictive mean, which is higher than $\mathcal{O}(nm) + \mathcal{O}(n^3)$. Furthermore, separably estimating $m(\tilde{p}+1)$ range and nugget parameters can be less stable as PP-GP.

3. Probabilistic generative models of dynamic mode decomposition

Mathematical and machine learning approaches often minimize a loss function for estimating parameters. Many loss-minimization approaches can be shown to be equivalent to statistical estimators from probabilistic generative models. We summarize a two-step procedure of building a generative model.

1. Build a probabilistic generative model of untransformed data with parameters that can be identified from data. Generalize the model as much as possible to the extent that the parameters can still be identified from the data.
2. Show equivalence between the estimator that minimizes a loss function and a statistical estimator, such as the maximum likelihood estimator or maximum marginal likelihood estimator, of the probabilistic generative model.

The generative model helps us understand the underlying assumptions from the loss-minimization estimator. A more efficienct estimator can be derived based on the generative model if the loss-minimization estimator is not optimal. We follow this procedure to derive a generative model of the dynamic mode decomposition that enables uncertainty assessment of the estimation.

3.1. Dynamic mode decomposition

Dynamic mode decomposition (DMD) is a data-driven approach to obtain a reduced rank representation of data from complex dynamical

systems [12], which quickly gains popularity for approximating dynamical systems [66]. Here we summarize DMD and derive a generative model of DMD.

Let us split the $m\times n$ real-valued observational matrix at n time points $\mathbf{Y}$ into two matrices, $\mathbf{Y}_{1:(n-1)} = [\mathbf{y}(\mathbf{x}_1), \mathbf{y}(\mathbf{x}_2), \dots, \mathbf{y}(\mathbf{x}_{n-1})]$ and $\mathbf{Y}_{2:n} = [\mathbf{y}(\mathbf{x}_2), \mathbf{y}(\mathbf{x}_3), \dots, \mathbf{y}(\mathbf{x}_n)]$. DMD relies on the approximation: $\mathbf{y}(\mathbf{x}_{t+1}) \approx \mathbf{A}\mathbf{y}(\mathbf{x}_t)$ for $t = 1, \dots, n-1$, where $\mathbf{A}$ is an $m \times m$ matrix. In DMD, $\mathbf{A}$ is estimated by minimizing the loss between the observations and the linear dynamics constructed from previous time steps:

$$\hat{\mathbf{A}} = \underset{\mathbf{A}}{\operatorname{argmin}} \|\mathbf{Y}_{2:n} - \mathbf{A}\mathbf{Y}_{1:n-1}\| = \mathbf{Y}_{2:n}(\mathbf{Y}_{1:n-1})^{+}, \tag{17}$$

where $\|\cdot\|$ is the L_2 norm or Frobenius norm and $(\mathbf{Y}_{1:n-1})^{+}$ is the Moore–Penrose pseudo-inverse of $\mathbf{Y}_{1:n-1}$.

We first introduce the lemma that connects the DMD estimation to the maximum likelihood estimator (MLE) of the linear mapping matrix in a dynamic linear model [67] or linear state space model [68].

Lemma 1 (*Equivalence Between the MLE of the Linear Mapping Matrix in a Linear State Space Model and the DMD Estimation*)**.** *The DMD estimator* $\hat{\mathbf{A}}$ *in Eq. (17) is the MLE of* $\mathbf{A}$ *of the following linear state space model*

$$\mathbf{y}(\mathbf{x}_{t+1}) = \mathbf{A}\mathbf{y}(\mathbf{x}_t) + \boldsymbol{\varepsilon}_{t+1}, \tag{18}$$

where $\boldsymbol{\varepsilon}_{t+1} \sim \mathcal{MN}(\mathbf{0}, \boldsymbol{\Sigma}_\varepsilon)$ *is a vector of Gaussian distributions with a positive definite covariance matrix* $\boldsymbol{\Sigma}_\varepsilon$*, for any* $t = 1, 2, \dots, n$ *and we assume the marginal distribution of initial state* $\mathbf{y}(\mathbf{x}_1)$ *does not depend on* $\mathbf{A}$*. We refer the linear state model by Eq. (18) the DMD-induced process.*

Proof. The log-likelihood of $\mathbf{A}$ and $\boldsymbol{\Sigma}_\varepsilon$ is:

$$\begin{aligned}\mathcal{L}(\mathbf{A}, \boldsymbol{\Sigma}_\varepsilon) &= \log\left\{ p(\mathbf{y}(\mathbf{x}_1)) \prod_{t=2}^{n} p(\mathbf{y}(\mathbf{x}_t) \mid \mathbf{y}(\mathbf{x}_{t-1}), \mathbf{A}, \boldsymbol{\Sigma}_\varepsilon) \right\} \\ &\propto -\frac{n}{2}\log(|\boldsymbol{\Sigma}_\varepsilon|) - \frac{\mathbf{y}(\mathbf{x}_1)^T \boldsymbol{\Sigma}_\varepsilon^{-1} \mathbf{y}(\mathbf{x}_1)}{2} \\ &\quad - \sum_{t=2}^{n} \frac{(\mathbf{y}(\mathbf{x}_t) - \mathbf{A}\mathbf{y}(\mathbf{x}_{t-1}))^T \boldsymbol{\Sigma}_\varepsilon^{-1} (\mathbf{y}(\mathbf{x}_t) - \mathbf{A}\mathbf{y}(\mathbf{x}_{t-1}))}{2}.\end{aligned}$$

Taking the derivative of log-likelihood with respect to $\mathbf{A}$ and $\boldsymbol{\Sigma}_\varepsilon$, we obtain the maximum likelihood estimator of $\mathbf{A}$ and $\boldsymbol{\Sigma}_\varepsilon$:

$$\begin{aligned}\hat{\mathbf{A}} &= \left(\sum_{t=2}^{n} \mathbf{y}(\mathbf{x}_t)\mathbf{y}(\mathbf{x}_{t-1})^T\right)\left(\sum_{t=2}^{n} \mathbf{y}(\mathbf{x}_{t-1})\mathbf{y}(\mathbf{x}_{t-1})^T\right)^{+} \\ &= \mathbf{Y}_{2:n}\mathbf{Y}_{1:n-1}^T(\mathbf{Y}_{1:n-1}\mathbf{Y}_{1:n-1}^T)^{+} = \mathbf{Y}_{2:n}(\mathbf{Y}_{1:n-1})^{+}, \\ \hat{\boldsymbol{\Sigma}}_\varepsilon &= \frac{\mathbf{y}(\mathbf{x}_1)\mathbf{y}(\mathbf{x}_1)^T + \sum_{t=2}^{n}(\mathbf{y}(\mathbf{x}_t) - \hat{\mathbf{A}}\mathbf{y}(\mathbf{x}_{t-1}))(\mathbf{y}(\mathbf{x}_t) - \hat{\mathbf{A}}\mathbf{y}(\mathbf{x}_{t-1}))^T}{n}.\end{aligned}$$

The last equality in the equation of $\hat{\mathbf{A}}$ can be derived by performing the singular value decomposition (SVD) to $\mathbf{Y}_{1:n-1}$ and connecting the SVD with Moore–Penrose pseudo-inverse. □

We notice that the maximum likelihood estimator of $\mathbf{A}$ is equivalent to the solution of DMD shown in Eq. (17). Here $\mathbf{Y}_{1:(n-1)}^{+}$ can be computed by the SVD of $\mathbf{Y}_{1:(n-1)} = \mathbf{U}\mathbf{D}\mathbf{V}^*$ as $\mathbf{Y}_{1:(n-1)}^{+} = \mathbf{V}\mathbf{D}^{-1}\mathbf{U}^*$, where $\mathbf{U}^*$ and $\mathbf{V}^*$ denote the conjugate transpose of $\mathbf{U}$ and $\mathbf{V}$, respectively, and $\mathbf{D} \in \mathbb{R}^{m\times(n-1)}$ is a rectangular diagonal matrix of non-negative singular values. In practice, one can keep the r largest singular values for approximation: $\mathbf{Y}_{1:(n-1)} \approx \mathbf{U}_r\mathbf{D}_r\mathbf{V}_r^*$, where $\mathbf{U}_r$ is the first r columns of $\mathbf{U}$, $\mathbf{D}_r$ is a $r\times r$ diagonal matrix containing the first r largest singular values, with $r \le \min(m, n-1)$, and $\mathbf{V}_r$ is the first r columns of $\mathbf{V}$. Consequently, $\hat{\mathbf{A}}$ can be approximated by

$$\hat{\mathbf{A}} \approx \mathbf{Y}_{2:n}\mathbf{V}_r\mathbf{D}_r^{-1}\mathbf{U}_r^*. \tag{19}$$

As some singular values may be small, the approximation from the right-hand side of Eq. (19) is typically more stable as it avoids numerical error in computing the diagonal terms in $\mathbf{D}^{-1}$.

A primary goal of DMD is to identify the nonzero eigenvalues and their corresponding eigenvectors of $\mathbf{A}$, denoted as $\{\lambda_i, \boldsymbol{\phi}_i\}_{i=1}^r$, which can approximate the Koopman eigenvalues and modes, respectively [15]. However, directly computing the eigenvalues and eigenvectors of a $m \times m$ matrix $\hat{\mathbf{A}}$ in Eq. (19) can be costly when m is large. To reduce the computational cost, we may project $\hat{\mathbf{A}}$ onto the column space of $\mathbf{U}_r$ and define $\tilde{\mathbf{A}}$ as

$$\tilde{\mathbf{A}} = \mathbf{U}_r^*\hat{\mathbf{A}}\mathbf{U}_r \approx \mathbf{U}_r^*\mathbf{Y}_{2:n}\mathbf{V}_r\mathbf{D}_r^{-1}\mathbf{U}_r^*\mathbf{U}_r = \mathbf{U}_r^*\mathbf{Y}_{2:n}\mathbf{V}_r\mathbf{D}_r^{-1}. \tag{20}$$

Denote $\{\lambda_i, \boldsymbol{\omega}_i\}_{i=1}^r$ to be the eigenpairs of $\tilde{\mathbf{A}}$ such that $\lambda_i\tilde{\mathbf{A}} = \tilde{\mathbf{A}}\boldsymbol{\omega}_i$. In [13], the authors show that λ_i is the DMD eigenvalue, and the corresponding eigenvector of $\mathbf{A}$, also known as the DMD mode, can be calculated below

$$\boldsymbol{\phi}_i = \frac{1}{\lambda_i}\mathbf{Y}_{2:n}\mathbf{V}_r\mathbf{D}_r^{-1}\boldsymbol{\omega}_i, \tag{21}$$

for $i = 1, \dots, r$.

The snapshots at any t can be approximated by DMD modes and eigenvalues with a smaller dimension. Denote $\boldsymbol{\Phi} = [\boldsymbol{\phi}_1, \dots, \boldsymbol{\phi}_r]$ and $\boldsymbol{\Lambda} = \operatorname{diag}(\lambda_1, \dots, \lambda_r)$, for any $t \ge 1$, the reconstructed snapshots $\hat{\mathbf{y}}(\mathbf{x}_t)$ can be represented as

$$\hat{\mathbf{y}}(\mathbf{x}_t) = \hat{\mathbf{A}}^{t-1}\mathbf{y}(\mathbf{x}_1) = \boldsymbol{\Phi}\boldsymbol{\Lambda}^{t-1}\boldsymbol{b}, \tag{22}$$

where $\boldsymbol{b} = [b_1, \dots, b_r]^T = \boldsymbol{\Phi}^{+}\mathbf{y}(\mathbf{x}_1)$ represents the mode amplitudes with $\boldsymbol{\Phi}^{+}$ being the pseudo-inverse of $\boldsymbol{\Phi}$. As $\mathbf{y}(\mathbf{x}_1)$ may contain measurement error, an alternative way is to minimize the squared error loss below:

$$\hat{\boldsymbol{b}} = \underset{\boldsymbol{b}}{\operatorname{argmin}} \sum_{t=1}^{n} \|\boldsymbol{\Phi}\boldsymbol{\Lambda}^{t-1}\boldsymbol{b} - \mathbf{y}(\mathbf{x}_t)\|^2, \tag{23}$$

where $\|\cdot\|$ is the L_2 norm or Frobenius norm.

Eq. (22) can be applied to any t, including those $t^* > n$ with n being the number of observed time points, and thus it can be used for forecasts. When the observations are noise-free, a more straightforward way is to let $\hat{\mathbf{y}}(\mathbf{x}_n) = \mathbf{y}(\mathbf{x}_n)$ and forecast output vector on any $t^* > n$ by

$$\hat{\mathbf{y}}(\mathbf{x}_{t^*}) = \hat{\mathbf{A}}^{t^*-n}\mathbf{y}(\mathbf{x}_n). \tag{24}$$

From the DMD-induced process in Eq. (18), we have the following lemma, which gives the posterior distribution for forecast.

Lemma 2. *Conditional on the observations* $\mathbf{Y}$ *with plug-in estimators of* $\hat{\mathbf{A}}$ *and* $\hat{\boldsymbol{\Sigma}}_\varepsilon$*, the posterior distribution of the output vector of DMD-induced process in Eq. (18) at any* $\mathbf{x}_{t^*}$ *follows a multivariate normal distribution*

$$\left(\mathbf{y}(\mathbf{x}_{t^*}) \mid \mathbf{Y}, \hat{\mathbf{A}}, \hat{\boldsymbol{\Sigma}}_\varepsilon\right) \sim \mathcal{MN}\left(\hat{\mathbf{y}}(\mathbf{x}_t^*), \sum_{i=0}^{t^*-n-1} \hat{\mathbf{A}}^i\hat{\boldsymbol{\Sigma}}_\varepsilon(\hat{\mathbf{A}}^T)^i\right), \tag{25}$$

where $\hat{\mathbf{y}}(\mathbf{x}_{t^*})$ *follows Eq. (24) for any* $t^* > n$*.*

Proof. We prove this by induction. For $t^* = n+1$, we have

$$\begin{aligned}\mathbb{E}[\mathbf{y}(\mathbf{x}_{t^*}) \mid \mathbf{Y}, \hat{\mathbf{A}}, \hat{\boldsymbol{\Sigma}}_\varepsilon] &= \hat{\mathbf{A}}\mathbf{y}(\mathbf{x}_n), \\ \mathbb{V}[\mathbf{y}(\mathbf{x}_{t^*}) \mid \mathbf{Y}, \hat{\mathbf{A}}, \hat{\boldsymbol{\Sigma}}_\varepsilon] &= \hat{\boldsymbol{\Sigma}}_\varepsilon.\end{aligned}$$

Assume for $t^* > n+1$, we have

$$\begin{aligned}\mathbb{E}[\mathbf{y}(\mathbf{x}_{t^*}) \mid \mathbf{Y}, \hat{\mathbf{A}}, \hat{\boldsymbol{\Sigma}}_\varepsilon] &= \hat{\mathbf{A}}^{t^*-n}\mathbf{y}(\mathbf{x}_n), \\ \mathbb{V}[\mathbf{y}(\mathbf{x}_{t^*}) \mid \mathbf{Y}, \hat{\mathbf{A}}, \hat{\boldsymbol{\Sigma}}_\varepsilon] &= \sum_{i=0}^{t^*-n-1} \hat{\mathbf{A}}^i\hat{\boldsymbol{\Sigma}}_\varepsilon(\hat{\mathbf{A}}^T)^i.\end{aligned}$$

For any t^*+1, by the sampling model in Eq. (18) and the law of total expectation, the posterior mean follows:

$$\begin{aligned}\mathbb{E}[\mathbf{y}(\mathbf{x}_{t^*+1}) \mid \mathbf{Y}, \hat{\mathbf{A}}, \hat{\boldsymbol{\Sigma}}_\varepsilon] &= \mathbb{E}[\mathbb{E}[\mathbf{y}(\mathbf{x}_{t^*+1}) \mid \mathbf{Y}, \mathbf{y}(\mathbf{x}_{t^*}), \hat{\mathbf{A}}, \hat{\boldsymbol{\Sigma}}_\varepsilon]] \\ &= \mathbb{E}[\hat{\mathbf{A}}\mathbf{y}(\mathbf{x}_{t^*}) \mid \mathbf{Y}, \hat{\mathbf{A}}, \hat{\boldsymbol{\Sigma}}_\varepsilon] \\ &= \hat{\mathbf{A}}\hat{\mathbf{y}}(\mathbf{x}_{t^*}) = \hat{\mathbf{A}}^{t^*+1-n}\mathbf{y}(\mathbf{x}_n).\end{aligned}$$

By the sampling model in Eq. (18) and the law of total covariance, for any t^*+1, the posterior covariance follows

$$\mathbb{V}[\mathbf{y}(\mathbf{x}_{t^*+1}) \mid \mathbf{Y}, \hat{\mathbf{A}}, \hat{\boldsymbol{\Sigma}}_\varepsilon]$$

$$=\mathbb{V}[\mathbb{E}[\mathbf{y}(\mathbf{x}_{t^*+1}) \mid \mathbf{Y}, \mathbf{y}(\mathbf{x}_{t^*}), \hat{\mathbf{A}}, \hat{\boldsymbol{\Sigma}}_\varepsilon]] + \mathbb{E}[\mathbb{V}[\mathbf{y}(\mathbf{x}_{t^*+1}) \mid \mathbf{Y}, \mathbf{y}(\mathbf{x}_{t^*}), \hat{\mathbf{A}}, \hat{\boldsymbol{\Sigma}}_\varepsilon]]$$
$$=\mathbb{V}[\hat{\mathbf{A}}\mathbf{y}(\mathbf{x}_{t^*}) \mid \mathbf{Y}, \hat{\mathbf{A}}, \hat{\boldsymbol{\Sigma}}_\varepsilon] + \hat{\boldsymbol{\Sigma}}_\varepsilon$$
$$=\hat{\mathbf{A}}\left(\sum_{i=0}^{t^*-n-1} \hat{\mathbf{A}}^i \hat{\boldsymbol{\Sigma}}_\varepsilon (\hat{\mathbf{A}}^T)^i\right)\hat{\mathbf{A}}^T + \hat{\boldsymbol{\Sigma}}_\varepsilon = \sum_{i=0}^{t^*-n} \hat{\mathbf{A}}^i \hat{\boldsymbol{\Sigma}}_\varepsilon (\hat{\mathbf{A}}^T)^i. \quad \square$$

Although we can assess the uncertainty of the forecast by DMD from the predictive distribution in Eq. (25), the linear state space model can be restrictive to approximate nonlinear dynamical systems. Furthermore, the uncertainty in estimating $\mathbf{A}$ and $\boldsymbol{\Sigma}_\varepsilon$ is not propagated for predictions. Finally, choosing the rank r in DMD is an open problem as it represents one's belief on the degree of the model is misspecified, which could be hard to be quantified precisely. Typical ways of choosing r include letting the summation of DMD eigenvalues explain a large proportion of the output variability, while this choice could potentially misfit the data, as minimizing the L_2 loss in Eq. (17) cannot avoid overfitting the data.

3.2. Higher order dynamic mode decomposition

One limitation of the DMD approach is that only the observation from the prior time point is used, equivalently inducing a first-order Markov model in Eq. (18). Variants of DMD approaches, such as Higher Order Dynamic Mode Decomposition (HODMD) [16] or Hankel DMD [69], use more observations from longer time lag to construct the dynamics:

$$\mathbf{y}(\mathbf{x}_{t+q}) = \mathbf{A}_1\mathbf{y}(\mathbf{x}_t) + \mathbf{A}_2\mathbf{y}(\mathbf{x}_{t+1}) + \cdots + \mathbf{A}_q\mathbf{y}(\mathbf{x}_{t+q-1}), \tag{26}$$

where $q \geq 1$ is a tunable parameter that determines the number of time-lagged snapshots to be included in the model.

The estimation accuracy from HODMD can be higher than the conventional DMD as multiple time-lagged snapshots are used. For scenarios where the number of time points is larger than the number of output coordinates, including more time-lagged snapshots can increase the upper bound of the number of nonzero singular values, thus potentially capturing complex dynamics in a higher dimensional space. From the theoretical point of view, the eigenfunctions and eigenvalues of HODMD are guaranteed to converge to the Koopman eigenfunctions and eigenvalues for ergodic systems [69].

Let us define $\mathbf{y}^{\text{aug}}(\mathbf{x}_t) = (\mathbf{y}(\mathbf{x}_t)^T, \mathbf{y}(\mathbf{x}_{t+1})^T, \ldots, \mathbf{y}(\mathbf{x}_{t+q-1})^T)^T$, an augmented vector of mq dimensions that contain q snapshots. The linear mapping matrix $\hat{\mathbf{A}}^{\text{HODMD}}$ in HODMD can be obtained by minimizing the Frobenius or L_2 norm between the observations and linear dynamics constructed from the previous time steps: $\hat{\mathbf{A}}^{\text{HODMD}} = \text{argmin}_{\mathbf{A}^{\text{HODMD}}} \|\mathbf{Y}^{\text{aug}}_{2:n} - \mathbf{A}^{\text{HODMD}}\mathbf{Y}^{\text{aug}}_{1:(n-1)}\|$, where $\mathbf{Y}^{\text{aug}}_{2:n} = [\mathbf{y}^{\text{aug}}(\mathbf{x}_2), \ldots, \mathbf{y}^{\text{aug}}(\mathbf{x}_n)]$ and $\mathbf{Y}^{\text{aug}}_{1:(n-1)} = [\mathbf{y}^{\text{aug}}(\mathbf{x}_1), \ldots, \mathbf{y}^{\text{aug}}(\mathbf{x}_{n-1})]$.

However, concatenating q consecutive snapshots increases the number of rows in $\mathbf{Y}^{\text{aug}}_{1:(n-1)}$ from m to mq, and the cost of a singular value decomposition for $\mathbf{Y}^{\text{aug}}_{1:(n-1)}$, leading to higher computational cost. To overcome this limitation, one can use a subsampled version of the data instead of the entire dataset. For instance, one might skip Δt time steps when constructing the data matrices, i.e., $\tilde{\mathbf{Y}}^{\text{aug}}_1 = [\mathbf{y}^{\text{aug}}(\mathbf{x}_1), \mathbf{y}^{\text{aug}}(\mathbf{x}_{1+\Delta t}), \ldots, \mathbf{y}^{\text{aug}}(\mathbf{x}_{1+\lfloor \frac{n-2}{\Delta t} \rfloor \Delta t})]$ and $\tilde{\mathbf{Y}}^{\text{aug}}_2 = [\mathbf{y}^{\text{aug}}(\mathbf{x}_2), \mathbf{y}^{\text{aug}}(\mathbf{x}_{2+\Delta t}), \ldots, \mathbf{y}^{\text{aug}}(\mathbf{x}_{2+\lfloor \frac{n-2}{\Delta t} \rfloor \Delta t})]$. The estimator of $\mathbf{A}^{\text{HODMD}}$ can be computed below

$$\hat{\mathbf{A}}^{\text{HODMD}} = \underset{\mathbf{A}^{\text{HODMD}}}{\text{argmin}} \|\tilde{\mathbf{Y}}^{\text{aug}}_2 - \mathbf{A}^{\text{HODMD}}\tilde{\mathbf{Y}}^{\text{aug}}_1\|. \tag{27}$$

The HODMD contains two prespecified parameters: the number of time-lagged snapshots q to be included in any given time, and the number of skipped time steps Δt in estimation. Note that the estimated $\mathbf{A}^{\text{HODMD}}$ does not preserve the model structure in Eq. (26). Instead, let us consider the following generative model for HODMD with parameters $(q, \Delta t)$

$$\mathbf{y}^{\text{aug}}(\mathbf{x}_{2+i\Delta t}) = \mathbf{A}^{\text{HODMD}}\mathbf{y}^{\text{aug}}(\mathbf{x}_{1+i\Delta t}) + \boldsymbol{\varepsilon}^{\text{aug}}_{2+i\Delta t}, \tag{28}$$

for $i = 1, \ldots, \lfloor (n-2)/\Delta t \rfloor$ with $\boldsymbol{\varepsilon}^{\text{aug}}_{2+i\Delta t} \sim \mathcal{MN}(\mathbf{0}, \boldsymbol{\Sigma}^{aug}_\varepsilon)$. In practice, the model performance depends on the choice of $(q, \Delta t)$, since the underlying data-generating model may rely on multiple time-lagged snapshots and using observations from longer time lag make the model more accurate. The HODMD estimator in Eq. (27) is equivalent to the MLE of $\mathbf{A}^{\text{HODMD}}$ in the generative model in Eq. (28).

3.3. Extended dynamic mode decomposition

The generative model of the DMD algorithm in Eq. (18) is a linear state space model, while some dynamical systems cannot be accurately approximated by linear dynamics. The extended dynamic mode decomposition (EDMD) [17] aims to define a dictionary of $\tilde{m}$ nonlinear basis functions $\mathbf{k}(\cdot) = (k_1(\cdot), ..., k_{\tilde{m}}(\cdot))^T$ to lift the observations to a system that can be approximated by linear dynamics. Denote the linear mapping matrix $\mathbf{A}^{\text{EDMD}}$ to be an approximation of the Koopman operator. In EDMD, the linear mapping matrix $\mathbf{A}^{\text{EDMD}}$ is obtained by minimizing the squared error loss function

$$\hat{\mathbf{A}}^{\text{EDMD}} = \underset{\mathbf{A}^{\text{EDMD}}}{\text{argmin}} \sum_{t=1}^{n-1} \|\mathbf{k}(\mathbf{y}(\mathbf{x}_{t+1})) - \mathbf{A}^{\text{EDMD}}\mathbf{k}(\mathbf{y}(\mathbf{x}_t))\|_2^2. \tag{29}$$

Similar to the DMD-induced process, the estimator of EDMD is equivalent to the maximum likelihood estimator of the linear mapping matrix in a linear state space model defined in the lifted space for $t = 1, \ldots, n-1$:

$$\mathbf{k}(\mathbf{y}(\mathbf{x}_{t+1})) = \mathbf{A}^{\text{EDMD}}\mathbf{k}(\mathbf{y}(\mathbf{x}_t)) + \boldsymbol{\varepsilon}^{\text{EDMD}}_{t+1}, \tag{30}$$

with $\boldsymbol{\varepsilon}^{\text{EDMD}}_{t+1} \sim \mathcal{MN}(\mathbf{0}, \boldsymbol{\Sigma}^{\text{EDMD}}_\varepsilon)$, where $\boldsymbol{\Sigma}^{\text{EDMD}}_\varepsilon$ is a positive definite matrix.

After estimating the linear mapping matrix between the linear state space model, we need to transform it back to predict the future states [20], which may be achieved by defining $\hat{\mathbf{y}}(\mathbf{x}_t) = \mathbf{P}\mathbf{k}(\mathbf{y}(\mathbf{x}_t))$, where $\mathbf{P}$ is a $m \times \tilde{m}$ matrix and can be estimated by $\hat{\mathbf{P}} = \text{argmin}_{\mathbf{P}} \sum_{t=1}^n \|\mathbf{y}(\mathbf{x}_t) - \mathbf{P}\mathbf{k}(\mathbf{y}(\mathbf{x}_t))\|^2$.

Choosing an appropriate set of basis functions is crucial for the EDMD method. A few generic basis functions, such as Hermite polynomials, radial basis functions, and discontinuous spectral elements, were suggested in [17]. The selection of basis functions depends on the context of the problem and domain knowledge may be used as well, whereas misspecifying basis functions can degrade the estimation efficiency of the model. PP-GP is closely connected to EDMD. Assuming the mean is zero, we can write the predictive mean of PP-GP in a matrix form $\hat{\mathbf{Y}} = \mathbf{W}\mathbf{K}$, where $\mathbf{Y}$ is a $m \times n$ observational matrix of n snapshots, and $\mathbf{W} = [\mathbf{w}_1^T, \ldots, \mathbf{w}_m^T]^T$ is an $m \times n$ weight matrix given from Corollary 1. This means the prediction from PP-GP uses the same kernel basis to represent the output for each coordinate, whereas the weights are estimated by solving the linear system of equations with shared coefficients, separately for the output at each coordinate. Compared to EDMD, the PP-GP does not project the output onto the lifted space, and hence we do not need to transform the lifted states back for forecasting. The PP-GP is a flexible model, as the mean and variance parameters are distinct for each coordinate, which can be marginalized out by computing the predictive distribution. Besides, the covariance matrix contains range and nugget parameters, and they are estimated by the maximum marginal posterior distribution, discussed in Appendix.

3.4. Computational complexity

The computational complexity of estimating the transition and covariance matrices in DMD are $\mathcal{O}(\min(m^2n, mn^2))$ and $\mathcal{O}(m^2n)$, respectively. Obtaining the n^*-step forecast with uncertainty quantification by DMD requires $\mathcal{O}(m^2rn^*)$, where r is the rank of the observation matrix $\mathbf{Y}_{1:n-1}$. Similarly, for HODMD with time lag q and thinking parameter Δt, the computational complexity of parameter estimation and n^*-step forecast with uncertainty quantification are

Table 1
Computational complexity for parameter estimation and forecast of n^* time points in DMD, HODMD, EDMD, and PP-GP, where m is the dimension of an output, n is the number of observations, r is the rank of data matrix, q is the number of snapshots stacked in HODMD, Δt is the number of skipped time points in HODMD, $\tilde{m}$ is the number of basis functions in EDMD, T is the number of iterations in K-means, $\tilde{S}$ and S are the iterations of numerical optimization and samples for forecast in PP-GP.

	Parameter estimation	Forecast and UQ
DMD	$\mathcal{O}(m^2 n)$	$\mathcal{O}(m^2 r n^*)$
HODMD	$\mathcal{O}((mq)^2 \lfloor \frac{n}{\Delta t} \rfloor)$	$\mathcal{O}(m^2 q^2 r n^*)$
EDMD	$\mathcal{O}(mn\tilde{m}T)$	$\mathcal{O}(m\tilde{m}n^*)$
PP-GP	$\mathcal{O}(\tilde{S}n^3 + \tilde{S}n^2\ m)$	$\mathcal{O}(Smn^2 n^* + n^3)$

$\mathcal{O}(\min((mq)^2 \lfloor \frac{n}{\Delta t} \rfloor, mq(\lfloor \frac{n}{\Delta t} \rfloor)^2)) + \mathcal{O}((mq)^2 \lfloor \frac{n}{\Delta t} \rfloor)$ and $\mathcal{O}((mq)^2 r n^*)$, respectively. For EDMD using $\tilde{m}$ radial basis functions to lift the data where the centers are determined by the K-means clustering approach, transforming the data and estimating the parameters in the lifted space require $\mathcal{O}(mn\tilde{m}T)$, where T is the number of iterations in K-means. Obtaining the predictions needs $\mathcal{O}(m\tilde{m}n^*)$. For simplicity, here we assume $m > n > \tilde{m}$. The computational complexity for estimating the parameters as well as providing forecast with uncertainty assessment by DMD, HODMD, EDMD, and PP-GP are summarized in Table 1.

4. Connection of different data-driven approaches of modeling dynamical systems with respect to the generative models

Here we compare three classes of data-driven models, namely the proper orthogonal decomposition (POD) [11], DMD and PP-GP approaches. To simplify the notations, we assume the data are properly centered, meaning that the $m \times n$ real-valued output matrix $\mathbf{Y}$ has zero mean.

In POD, the data are decomposed by SVD $\mathbf{Y} = \mathbf{U}_y \mathbf{D}_y \mathbf{V}_y^T$, where $\mathbf{U}_y$ and $\mathbf{V}_y$ are $m \times m$ and $n \times n$ unitary matrix, respectively, and $\mathbf{D}_y$ is a $m \times n$ rectangle diagonal matrix with non-negative singular values in the diagonals. The first $r \leq m$ columns of $\mathbf{U}_y$ associated with the largest r singular values provide the orthogonal basis of a linear subspace to reconstruct the covariance of output at different coordinates by treating the temporal observations as independent measurements: $\mathbf{Y}\mathbf{Y}^T/(n-1) = \mathbf{U}_y \mathbf{D}_y^2 \mathbf{U}_y^T/(n-1)$. This approach is known as the principal component analysis (PCA) [70], which is widely used in unsupervised learning and dimension reduction. The SVD basis from the PCA can be shown to have the same linear subspace to the maximum marginal likelihood estimator of the linear mapping matrix $\mathbf{B}$ after marginalizing out $\mathbf{z}(\mathbf{x}_t)$ in the following generative model [71]:

$$\mathbf{y}(\mathbf{x}_t) = \mathbf{B}\mathbf{z}(\mathbf{x}_t) + \epsilon_t, \tag{31}$$

where $\epsilon_t \sim \mathcal{MN}(\mathbf{0}, \sigma_0^2 \mathbf{I}_m)$ and $\mathbf{z}(\mathbf{x}_t) \sim \mathcal{MN}(\mathbf{0}, \mathbf{I}_r)$ is a r-dimensional latent factors independently following standard normal distributions. Under such model, the covariance of the data at each time follows $\mathbb{V}[\mathbf{y}(\mathbf{x}_t)] = \mathbf{B}\mathbf{B}^T + \sigma_0^2 \mathbf{I}_m$. However, the generative model assumes independence between the observations at different time points, which is restrictive. In [72], $z_l(\cdot)$ is modeled as a GP for each $l = 1, \ldots, r$, and the maximum marginal likelihood estimator of $\mathbf{B}$ under the assumption $\mathbf{B}^T\mathbf{B} = \mathbf{I}_r$ is derived.

Second, the generative model of DMD is given in Eq. (18), where the noise of the data is not modeled. Assuming the initial states follow a multivariate normal distribution with zero mean and covariance $\boldsymbol{\Sigma}_\epsilon$. It is not hard to show that $\mathbb{E}[\mathbf{y}(\mathbf{x}_t)] = \mathbf{0}$ and the covariance between any two output vectors at two time points follows: $\mathrm{Cov}[\mathbf{y}(\mathbf{x}_{t'}), \mathbf{y}(\mathbf{x}_t)] = \mathbf{A}^{t'-t} \sum_{i=0}^{t-1} \mathbf{A}^i \boldsymbol{\Sigma}_\epsilon (\mathbf{A}^T)^i$ for $t' \geq t$. Specifically, the covariance of output at any time point $t \geq 1$ follows $\mathbb{V}[\mathbf{y}(\mathbf{x}_t)] = \sum_{i=0}^{t-1} \mathbf{A}^i \boldsymbol{\Sigma}_\epsilon (\mathbf{A}^T)^i$. Compared with the sampling model in POD, the generative model in the DMD-induced process in Eq. (18) are correlated over time and the strength of the correlation is captured by the linear mapping matrix $\mathbf{A}$ between output vectors at the consecutive time points. The DMD-induced process may not be differentiable with respect to time and the assumption of homogeneous variance at each output coordinate may also be restrictive for applications where the output has different scales.

Third, the PP-GP model in Eq. (9) has the same predictive mean as modeling the output matrix $\mathbf{Y}$ by a matrix-normal distribution [33], with a separable covariance $\mathbb{V}[\mathbf{Y}] = \boldsymbol{\Sigma}_y \otimes \tilde{\mathbf{K}}$, where $\boldsymbol{\Sigma}_y$ is the covariance between output coordinates with the jth diagonal term being σ_j^2, for $j = 1, \ldots, m$ and $\tilde{\mathbf{K}}$ is the correlation matrix between inputs with $\otimes$ denoting the Kronecker product. In comparison, the generative model by DMD in Eq. (18) is a linear state space model, which has a semi-separable covariance structure. The PP-GP induces nonlinear dynamics when using the observations from previous time points as the inputs, and the differentiability of the nonlinear processes induced by PP-GP can be controlled by the choice of kernel function. When the underlying dynamic is smooth, a differentiable GP prior of the nonlinear dynamics may be preferred to have a better convergence rate compared to a nondifferential GP prior [34]. Another advantage of PP-GP is that the range parameters can be estimated by the MLE or maximum marginal posterior mode, which is more flexible than using fixed nonlinear basis functions in EDMD. Lastly, the variance of the output coordinate is distinct and the variance estimator of PP-GP has a closed form expression in Eq. (10), whereas the induced processes by DMD and its variants typically have homogeneous variance. The different variance terms make PP-GP particularly suitable when the output has different scales, which are common in practice.

5. Numerical results

We compare different data-driven forecast approaches for nonlinear dynamical systems, focusing on uncertainty quantification of the forecast. We consider two scenarios. In the first scenario, we assume the underlying dynamical system is modeled by a map from $\mathbb{R}^p \to \mathbb{R}^m$: $d\mathbf{y}/dt = \mathbf{f}(\mathbf{x}_t)$, where the input variables $\mathbf{x}_t$ is a subset of $\mathbf{y}_t$. Here the vector-valued function $\mathbf{f}$ is treated as unknown and required to be approximated, whereas the inputs $\mathbf{x}_t$ are known. For all approaches, we do not include the vector-valued function $\mathbf{f}(\cdot)$ in nonlinear basis functions. Instead, we test uncertainty quantification with generic kernels or nonlinear basis functions that provide default ways of approximation. In the second scenario, the dynamical system is described by $d\mathbf{y}/dt = \mathbf{f}(\mathbf{x}_t, \mathbf{u}_t)$, where we can only observe $\mathbf{x}_t$, whereas the external inputs $\mathbf{u}_t$ and the vector-valued function $\mathbf{f}(\cdot)$ are unobserved.

For both scenarios, we forecast held-out data $\mathbf{y}(\mathbf{x}_{t^*}) = (y_1(\mathbf{x}_{t^*}), \ldots, y_m(\mathbf{x}_{t^*}))^T$ at $t^* = n+1, n+2, \ldots, n+n^*$. The autoregressive model with lag order 1 (AR(1)) separately fitted for each coordinate is used as a benchmark prediction model [73]. Also included are DMD, HODMD and PP-GP. Note that the generative model of the DMD method is equivalent to a noise-free vector autoregressive (VAR) model with lag order 1 [74] and hence we do not include other VAR models for comparison. The criteria include the predictive root of mean squared error (RMSE), the average length of the 95% predictive intervals ($L(95\%)$), and the proportion of the samples covered in the 95% predictive interval ($P(95\%)$):

$$\mathrm{RMSE} = \left(\sum_{j=1}^{m} \sum_{t=n+1}^{n+n^*} (\hat{y}_j(\mathbf{x}_{t^*}) - y_j(\mathbf{x}_{t^*}))^2 \right)^{\frac{1}{2}}, \tag{32}$$

$$L(95\%) = \frac{1}{mn^*} \sum_{j=1}^{m} \sum_{t^*=n+1}^{n^*+n} \mathrm{length}\left\{ CI_{j,t^*}(95\%) \right\}, \tag{33}$$

$$P(95\%) = \frac{1}{mn^*} \sum_{j=1}^{m} \sum_{t^*=n+1}^{n^*+n} 1_{y_j(\mathbf{x}_{t^*}) \in CI_{j,t^*}(95\%)}, \tag{34}$$

where $\hat{y}_j(\mathbf{x}_{t^*})$ is the prediction of the output at coordinate j with input $\mathbf{x}_{t^*}$, $CI_{j,t^*}(95\%)$ is the 95% predictive interval of the output at coordinate j and time t^*, length $\left\{ CI_{j,t^*}(95\%) \right\}$ denotes the length of the predictive interval, and $\bar{y} = \sum_{j=1}^{m} \sum_{t=1}^{n} y_j(\mathbf{x}_t)/(mn)$ is the mean of the

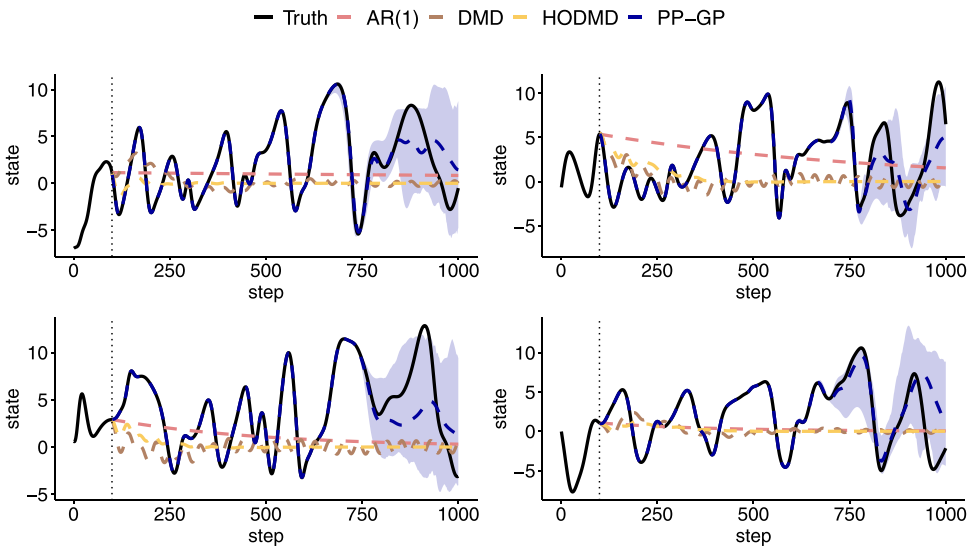

Fig. 1. Forecast of the Lorenz 96 systems for 900 steps by AR(1) (pink dashed curves), DMD (brown dashed curves), HODMD (yellow dashed curves) and PP-GP (blue dashed curve). The 95% predictive interval by PP-GP is graphed as blue shaded area. The blue dashed curves (PP-GP) and black curves (held-out truth) overlap for around the first 500 held-out time steps.

observations in the training data set. An accurate method should have small predictive error quantified by RMSE, short average length of 95% predictive interval ($L(95\%)$) and the proportion of the sample covered by the 95% predictive interval ($P(95\%)$) should be close to the 95% nominal level.

5.1. Lorenz 96 system

We first discuss the Lorenz 96 system for modeling the atmospheric quantities at equally spaced locations along a cycle [35]:

$$\frac{dy_j(t)}{dt} = (y_{j+1}(t) - y_{j-2}(t))y_{j-1}(t) - y_j(t) + F, \tag{35}$$

for $j = 1, \dots, m$, where $m = 40$ and $F = 8$ are typically used for testing. Here $f_j(\mathbf{x}_t) = (y_{j+1}(t) - y_{j-2}(t))y_{j-1}(t) - y_j(t)$, where the 4 dimensional input is $\mathbf{x}_t = \{y_{j-2}(t), y_{j-1}(t), y_j(t), y_{j+1}(t)\}$. The Lorenz 96 system is often used for demonstrating the effectiveness of nonlinear filtering approaches, such as ensemble Kalman filter in data assimilation [75], where the function $f_j(\cdot)$ is typically assumed to be known. Here we assume the underlying dynamics from $f_j(\cdot)$ is unknown.

We test a few methods and compare their performance on uncertainty quantification. We assume both the derivative and output values are available. The data are obtained by the Runge Kutta method of order 4 with step size $h = 0.01$ for 1000 steps. The initial values of the states are sampled from zero mean multivariate normal distribution where covariance matrix is sampled from a Wishart distribution with the scale matrix being identity and m degrees of freedom [76]. Any method can use the $mn = 4000$ observations from the first $n = 100$ time points as training observations, whereas the rest of the 36,000 observations at later $n^* = 900$ time points are held out as test data. For DMD and HODMD, we try both the observed output values and derivatives for forecasting. Since both ways do not work well, we only present results based on the observed output values. When constructing the data matrices for HODMD, 6 observed snapshots ($q = 6$) are concatenated and 3 time points are skipped in the augmented data ($\Delta t = 3$). For PP-GP, we uniformly subsample $n_{training} = 500$ observations from 4000 observations of derivatives in the training time period to estimate the parameters and construct predictive distributions in Eq. (8), because of the high computational cost when n is large. We use the default product Matérn covariance function with roughness parameter being 2.5 in PP-GP. As PP-GP can be considered as an extended version of DMD on projecting the data onto the kernel space discussed in Section 3.3, we do not include any other EDMD approach. The range parameters of the kernel in PP-GP are estimated by the default marginal posterior mode estimation [56], which is more flexible than assuming fixed nonlinear basis function in EDMD.

Table 2
Forecast accuracy and uncertainty assessment on the held-out data. The standard deviation is 3.54 and 3.62 for the 500-step test data and 900-step test data, respectively.

500-step forecast	RMSE	$P(95\%)$	$L(95\%)$
AR(1)	4.60	69.8%	10.8
DMD	4.51	91.6%	20.2
HODMD	4.24	98.8%	39.4
PP-GP	0.0126	93.9%	0.0352
900-step forecast	**RMSE**	$P(95\%)$	$L(95\%)$
AR(1)	4.63	75.4%	12.6
DMD	4.55	93.5%	21.9
HODMD	4.37	99.3%	43.6
PP-GP	1.52	94.6%	2.64

Fig. 1 gives the 900-step forecast by AR(1), DMD, HODMD and PP-GP for the 10th 20th, 30th and 40th states. The uncertainty of the forecast by PP-GP is graphed as the blue shared area in all plots. With the default kernel function and estimation [56], the forecast of PP-GP is reasonably accurate for the first 500 time steps and 95% predictive interval by PP-GP (graphed as the blued shaded area) is almost indistinguishable in this time domain. As the average Lyapunov time for our simulation is around 1.58, the PP-GP can make precise forecast for more than 3 Lyapunov time. The 95% predictive interval becomes noticeably wider at later time steps, and simultaneously, the difference between PP-GP and held-out truth becomes large. The internal uncertainty assessment quantified by the 95% predictive interval of PP-GP provides a time range of reliable forecasts without knowing the held-out truth. Furthermore, Fig. 2 compares the truth to the forecast by PP-GP for all states, which confirms the forecast by PP-GP for the first 500 steps is accurate for all states.

Table 2 summarizes the performance of forecast and uncertainty assessment by different approaches for the Lorenz 96 system. The RMSE by the PP-GP for the first 500 steps is much smaller than the standard deviation of the test data and the length of the 95% predictive interval by PP-GP is also substantially smaller than the variability in the held-out observations. Even if the 95% predictive interval by PP-GP is short, it covers 93.9% of the observations, indicating the uncertainty of the

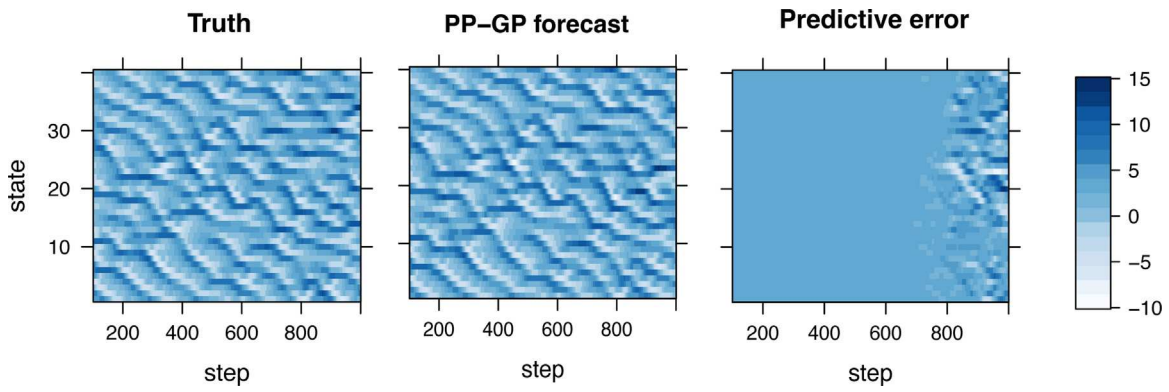

Fig. 2. The truth, 900-step forecast by PP-GP, and their difference for the Lorenz 96 system.

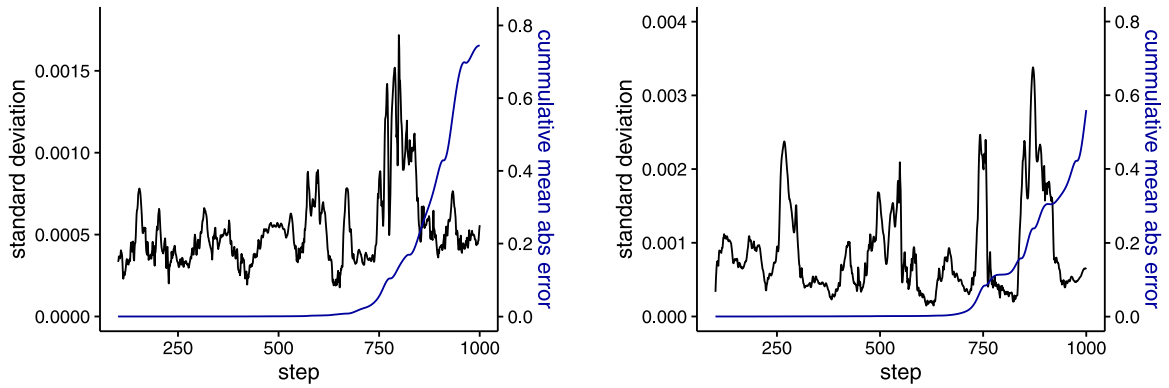

Fig. 3. The predictive standard deviation from the PP-GP model at each step and cumulative mean absolute error $SE_j(t^*) = \sum_{s^*=n+1}^{t^*} |\hat{y}_j(s^*) - y_j(s^*)|/(t^* - n)$ of 900-step forecast of state $j = 1$ and $j = 21$ for $t^* = 101, \ldots, 1000$ and $n = 100$.

forecast is properly quantified. The predictive error by PP-GP becomes larger at later steps due to the accumulation of the approximation error, and the overall predictive error of the 900-step forecast is dominated by the large error at later time points. Note that the length of the 95% predictive interval from PP-GP increases automatically in PP-GP, enabling 94.6% of the held-out data to be covered by the 95% predictive interval. In comparison, although the proportion of the samples covered by the 95% predictive interval by DMD and HODMD is also close to the 95% nominal level, the average length of intervals is substantially larger, as the underlying dynamics cannot be written as a linear combination of previous outputs. It is worth mentioning that the uncertainty of DMD and HODMD is affected by the selected rank to represent the data, here chosen to be the smallest value such that the summation of the eigenvalues explain at least 99% of the variability. A principled way to model the noise and select the rank may improve the uncertainty assessment of these methods.

Fig. 3 presents the predictive standard deviation of PP-GP and the corresponding cumulative mean absolute error between the true values and PP-GP forecasts, for the 1st and 21st states. In the initial 500-step forecasts, both the standard deviation computed by Eq. (8) and the cumulative mean absolute error are relatively small. As the number of forecast steps increases, the cumulative mean absolute error increases. The 95% predictive interval in Fig. 1 can be used to quantify the time when the forecast becomes inaccurate.

The uncertainty assessment by the PP-GP model is reasonably accurate as we convert the challenging problem of forecasting chaotic systems to the problem of predicting the one-step-ahead function in a 4-dimensional input space. In practice, reducing the inputs to a low-dimensional space is helpful for producing reliable forecasts and uncertainty assessments.

5.2. Time-dependent Green's function

In the second example, we apply the data-driven approach to forecast the simulation of the nonequilibrium dynamics of interacting electrons in materials exposed to intense time-varying electric fields, which is one of the most challenging problems in condensed matter physics. Modeling nonequilibrium dynamics from the basic principles of quantum mechanics is exceptionally challenging in computation, and has only been extended to the *ab initio* simulation of real materials in the last decade through the development of a new time-domain approach based on a formalism of Keldysh, Kadanoff and Baym [40–42] for the dynamics of the nonequilibrium Green's function [43,44,77]. We refer to the new computational approach as the time-dependent adiabatic GW (TD-aGW) method, where G stands for the interacting single-particle Green's function and W stands for the screened Coulomb interaction. Details about our implementation of the method can be found in [44], which is built on approximations developed in [43]. In principal, the TD-aGW approach gives quantitatively accurate descriptions of materials' response to electric fields, allowing for the simulation of experiments with femtosecond resolution and the development of new classes of quantum materials whose properties can be switched with laser pulses [45,47,78]. However, the scaling with respect to the size of the problem is computationally prohibitive, and thus, the TD-aGW method is currently limited to systems containing a few atoms and for short timescales.

In the context of many-body perturbation theory and second quantization, the nonequilibrium Green's function is a two-point correlation function comprised of the creation and annihilation field operators that increase and decrease the number of particles in a given state respectively. In general, it is a function of two time variables, but following the work in [43,44], the change in the energy of an electron due interactions with its environment (i.e. the electron self energy) can

be approximated as the equilibrium self energy plus a static nonequilibrium contribution. In this approximation, the key equation that we solve in TD-aGW is an equation of motion for a matrix of single-particle density $\rho(t)$ with index $(m_1 m_2, \mathbf{k})$

$$i\hbar \frac{\partial}{\partial t} \rho_{m_1 m_2, \mathbf{k}}(t) = [\mathbf{H}(t), \rho(t)]_{m_1 m_2, \mathbf{k}}, \tag{36}$$

where $\mathbf{H}(t)$ is a matrix of the Hamiltonian of the system having the same size of $\rho(t)$, the square bracket denotes the commutator of two matrices $\mathbf{A}$ and $\mathbf{B}$ such that $[\mathbf{A}, \mathbf{B}] = \mathbf{AB} - \mathbf{BA}$, and $\hbar$ is the Planck constant. The particle density matrix is written in a basis of electron quantum states with a band index m and a wave vector (or crystal momentum) $\mathbf{k}$. Hence, the matrix element $\rho_{m_1 m_2, \mathbf{k}}(t)$ encodes the entanglement of a state $(m_1, \mathbf{k})$ with another state $(m_2, \mathbf{k})$. The Hamiltonian of the system is calculated as

$$\mathbf{H}(t) = \mathbf{H}_0 - e\mathbf{E}(t) \cdot \mathbf{r} + \boldsymbol{\Sigma}^{GW} + \delta\boldsymbol{\Sigma}(t). \tag{37}$$

Here, $\mathbf{H}_0$ is the system's mean-field Hamiltonian at equilibrium; $\boldsymbol{\Sigma}^{GW}$ is the electron self-energy at equilibrium computed within the GW approximation; and $\delta\boldsymbol{\Sigma}$ is the nonequilibrium correction to the electron self-energy. $e\mathbf{E}(t) \cdot \mathbf{r}$ describes the coupling of the system with the external electric field, such that $\mathbf{E}(t)$ is a spatially-uniform, time-varying electric field; $\mathbf{r}$ is the position operator in quantum mechanics; and e is the fundamental charge of the electron. $\mathbf{H}_0$ and $\boldsymbol{\Sigma}^{GW}$ describe equilibrium properties and thus have no time-dependence. $\delta\boldsymbol{\Sigma}(t)$ is a functional of $\rho(t)$:

$$\delta\Sigma_{m_1 m_2, \mathbf{k}}(t) = \sum_{m_1', m_2', \mathbf{k}'} \rho_{m_1 m_2, \mathbf{k}-\mathbf{k}'}(t) W_{m_1 m_1' m_2 m_2', \mathbf{k}-\mathbf{k}'}, \tag{38}$$

where $W_{m_1 m_1' m_2 m_2', \mathbf{k}-\mathbf{k}'}$ is the equilibrium screened Coulomb interaction computed in the random-phase approximation (RPA); m_1, m_1', m_2, m_2' are band indices, and $\mathbf{k}$ and $\mathbf{k}'$ are vectors in reciprocal space.

We use monolayer MoS_2–a material where the electron self energy is known to be large [79–83]–as our test system. In order to perform the time evolution in Eq. (36), we first need to compute the equilibrium solution before the electric field is turned on to obtain $\rho(t = 0)$, as well as the time-independent matrices $\mathbf{H}_0$ and $\boldsymbol{\Sigma}^{GW}$ in Eq. (36) and $\mathbf{W}$ in Eq. (38). We do this within the one-shot GW approximation [37,38,84] using the following calculation parameters. Our basis includes 4 bands (the top 2 valence bands and the bottom 2 conduction bands) and a uniform grid of $36 \times 36 \times 1$ $\mathbf{k}$-points in reciprocal space. Density functional theory (DFT) calculations with spin–orbit coupling are performed using the Quantum Espresso package [85]. We use norm-conserving fully relativistic PBE pseudopotentials from the SG15 ONCV potential library [86] and a plane wave cutoff energy of 80 Ry. A GW plus Bethe Salpeter equation (BSE) calculation is done as a one-shot calculation on top DFT using the BerkeleyGW package [84]. A dielectric cutoff of 10 Ry and 6000 bands are used in the GW and BSE calculations. We use the results of the equilibrium calculations to setup Eq. (36) at time t=0, and then we propagate the equation in time with an external electric field that mimics a laser pulse polarized along the r_1 direction. The field is $E_{r_1}(t) = A \sin\left(\frac{\pi t}{T}\right)^2 \sin(\omega t)$, where the constants (in Ryd) $A = 0.006$ is the amplitude of the electric field, $\omega = 0.022$ is the frequency of the light, and $T = 160$ fs is the duration of the light pulse, all in Rydberg atomic units. $E_{r_2}(t) = 0$ and $E_{r_3}(t) = 0$. The electric field parameters are values consistent with typical experimental setups in high harmonic generation (HHG) experiments [87]. The 4th order Runge–Kutta method is then used to update $\rho_{m_1 m_2, \mathbf{k}}(t)$ and $\delta\Sigma(t)$ at each time step, and the matrix element $\rho_{m_1 m_2, \mathbf{k}}(t)$ is saved for the single $\mathbf{k}$-point, $\mathbf{k} = \mathbf{0}$, to be used as the test and training data for our data-driven models.

Our focus in this example is on forecasting the real part of the density matrix from Eq. (36), which in turn is related to the two-time Green's function through the Generalized Kadanoff-Baym ansatz (GKBA) [42,88]. Each snapshot is a 16-dimensional vector: $\mathbf{y}_t = \mathrm{Vec}(\boldsymbol{\rho}_{\mathbf{k}}(t))$, where $\boldsymbol{\rho}_{\mathbf{k}}(t)$ is a 4×4 matrix with the (m_1, m_2)th entry being $\rho_{m_1 m_2, \mathbf{k}}(t)$ for $m_1 = 1, \ldots, 4$ and $m_2 = 1, \ldots, 4$, and $\mathbf{k} = \mathbf{0}$. To solve Eq. (36), one needs to compute $\mathbf{E}(t)$, which is a known analytic function, and $\delta\boldsymbol{\Sigma}(t)$, which depends on the density matrix $\rho_{m_1 m_2, \mathbf{k}-\mathbf{k}'}(t)$ at other reciprocal lattice vectors $\mathbf{k} \neq \mathbf{0}$. To evaluate the performance of the data-driven approaches, we only utilize the observations $\mathbf{y}_t$ to construct the model, which only contains the local information at $\mathbf{k} = \mathbf{0}$, whereas the external field $\mathbf{E}(t)$ and interactions terms $\delta\boldsymbol{\Sigma}(t)$ from other lesser Green's function $G^{<}_{m_1 m_2, \mathbf{k}-\mathbf{k}'}(t)$ are not used. For all methods, we test two scenarios with training time steps being 2500 and 3500, respectively. For AR(1) and DMD, all training data are used. For HODMD, we use the same setting as the previous example with $q = 6$ and $\Delta t = 3$. For PP-GP, we use an isotropic kernel and 800 pairs of observations, uniformly sampled from the training data for constructing the model, and the output vector of $m = m_1 m_2 = 16$ dimensions from the previous time point is used as the input for predicting the one-step-ahead transition function.

Table 3
Forecast and uncertainty assessment for the density matrix using the held-out data. Observations at the first 2500 time steps are used to fit the model. The standard deviation is 1.56×10^{-5} and 1.48×10^{-5} of the 1000-step test data and the 2000-step test data, respectively.

1000-step forecast	RMSE	$P(95\%)$	$L(95\%)$
AR(1)	6.20×10^{-6}	23.7%	3.97×10^{-6}
DMD	2.87×10^{-5}	8.65%	2.03×10^{-6}
HODMD	1.30×10^{-5}	9.52%	6.56×10^{-7}
PP-GP	3.53×10^{-6}	65.2%	2.98×10^{-6}
2000-step forecast	RMSE	$P(95\%)$	$L(95\%)$
AR(1)	6.23×10^{-6}	35.8%	5.57×10^{-6}
DMD	2.01×10^{-3}	4.35%	7.29×10^{-5}
HODMD	6.22×10^{-5}	9.22%	1.94×10^{-6}
PP-GP	3.42×10^{-6}	75.3%	3.97×10^{-6}

Table 4
Forecast and uncertainty assessment of the density using the held-out data. Observations from the first 3500-time steps are used to fit the model. The standard deviation is 1.40×10^{-5} and 1.06×10^{-5} of the 1000-step test data and the 2000-step test data, respectively.

1000-step forecast	RMSE	$P(95\%)$	$L(95\%)$
AR(1)	1.13×10^{-5}	31.7%	7.12×10^{-6}
DMD	6.00×10^{-6}	15.9%	1.00×10^{-6}
HODMD	9.29×10^{-6}	29.6%	1.05×10^{-6}
PP-GP	3.93×10^{-6}	81.7%	1.89×10^{-5}
2000-step forecast	RMSE	$P(95\%)$	$L(95\%)$
AR(1)	1.50×10^{-5}	28.5%	9.93×10^{-6}
DMD	8.83×10^{-6}	16.3%	2.98×10^{-6}
HODMD	2.24×10^{-5}	20.2%	1.62×10^{-6}
PP-GP	9.13×10^{-6}	88.8%	4.36×10^{-5}

Table 3 and Table 4 provide the performance of each method when using observations from the first 2500 time steps and first 3500 time step as the training data, respectively. The PP-GP has the smallest RMSE for 2000-step forecast among three approaches in both scenarios, smallest RMSE for 1000-step forecast in the first scenario.

The misspecification of the input variables, however, degrades the accuracy of predictions and uncertainty quantification. The predictive error differs when using two output regimes as the training data, as the trend is different, and neither represents the trend in the held-out test data. The coverage of held-out data by the 95% predictive interval from the PP-GP is the highest among all methods, and having observations from a longer training time period seems to improve the overall coverage of the held-out data.

Fig. 4 and Fig. 5 display the 1000-step forecast by AR(1), DMD, HODMD and PP-GP model using 2500 and 3500 timesteps as training data, respectively. The PP-GP model can make accurate prediction for the first few cycles with short predictive intervals. The prediction error accumulates, and the model automatically detects the inaccuracy of the prediction, leading to larger 95% predictive intervals at later time

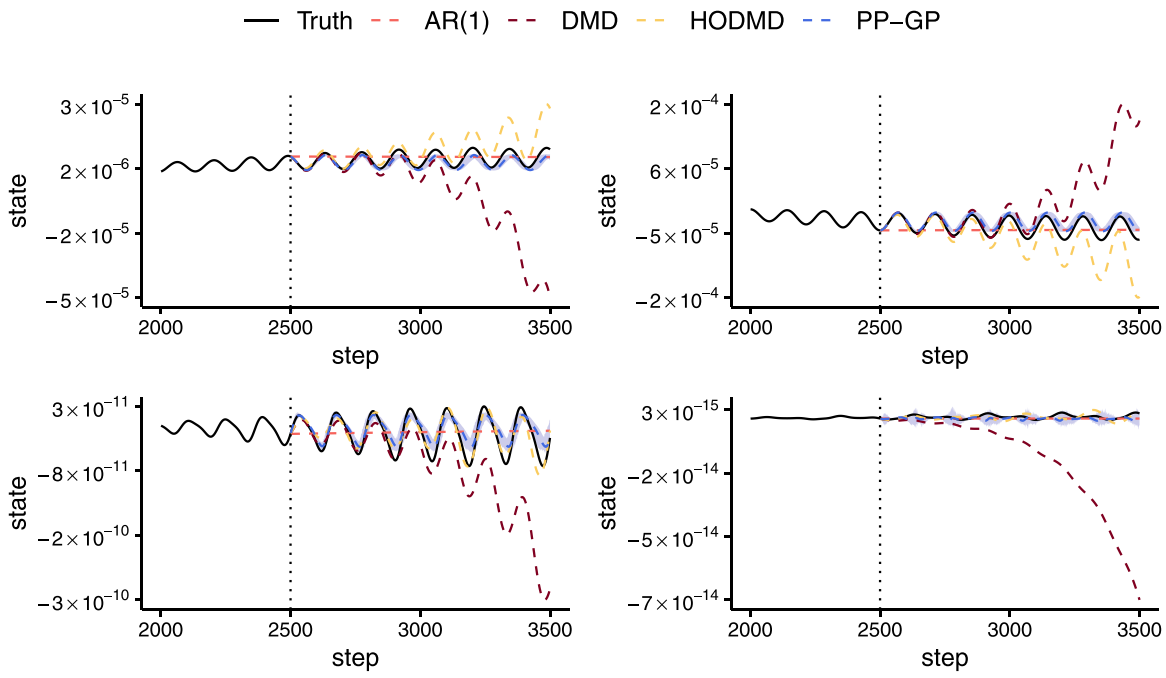

Fig. 4. Forecast of two time Green's function for 1000 steps by AR(1) (orange dashed curves), DMD (brown dashed curves), HODMD (yellow dashed curves) and PP-GP (blue dashed curve). 2500-time steps are used as training data. The 95% predictive interval by PP-GP is graphed as blue shaded area.

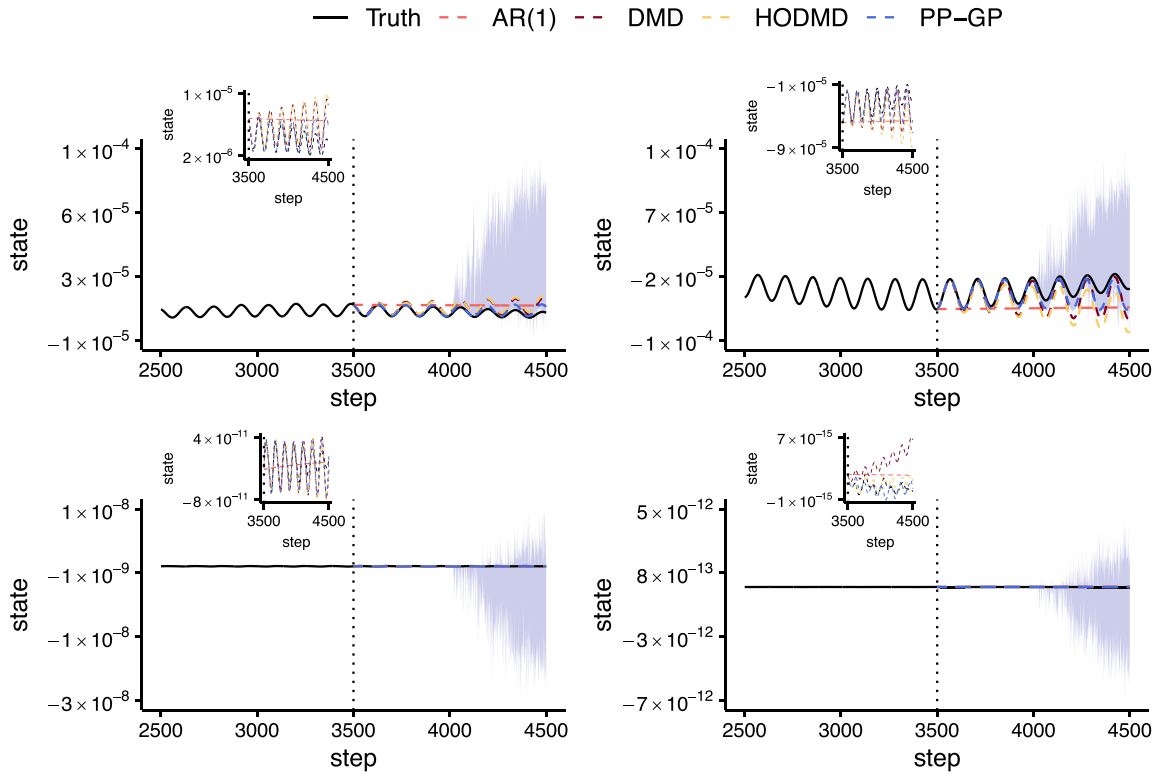

Fig. 5. Forecast of two time Green's function for 1000 steps by AR(1) (orange dashed curves), DMD (brown dashed curves), HODMD (yellow dashed curves) and PP-GP (blue dashed curve). 3500-time steps are used as training data. The 95% predictive interval by PP-GP is graphed as blue shaded area.

points. Note that the scale of the held-out truth is decreasing and the overall trend is not captured by any method. This is because for all four methods, some inputs such as $\mathbf{H}(t)$ and $\rho(t)$ at $\mathbf{k} \neq \mathbf{0}$, are assumed to be unknown. The large predictive intervals from PP-GP indicate substantial differences between the output in the training and forecast period, signaling more information is required to obtain an accurate prediction.

6. Concluding remarks

Quantifying the uncertainty for forecast and extrapolation by data-driven models is a challenging task that was not well-studied. We showed popular approaches for representing dynamical systems, such as the dynamic mode decomposition, can be written as the maximum likelihood estimator of a linear mapping matrix in a linear state space model, and this generative model allows the uncertainty to be quantified of forecast rigorously. We also extended the parallel partial Gaussian process approach to emulate the one-step-ahead transition function that links observations at two nearby time frames, and propagated the uncertainty through posterior sampling for forecasting a longer time. We numerically compared different approaches with correctly specified inputs and misspecified inputs in two examples. We discussed scenarios where the uncertainty can be reliably quantified, and analyzed the factors that can degrade the accuracy of uncertainty assessment.

There is a wide range of open issues to obtain reliable uncertainty quantification for probabilistic forecast of nonlinear dynamical systems. First, restrictive model assumptions, such as equal variance between output coordinates, subjective choice of latent dimensions and lack of models of trends from the forecast period, can degrade the accuracy of uncertainty assessment for forecasting. Having a probabilistic generative model allows one to better understand the model assumptions and hence select data-driven models more suitable for real-world tasks. Second, the kernel representation of vector functions, such as the PP-GP model, can capture nonlinear behaviors of dynamical systems through modeling the one-step-ahead transition function, whereas the Markov assumption of the model can be restrictive. Having inputs from longer time lag period may improve the model performance. Furthermore, when the dimension of input is large, we need to develop a computationally scalable way to reduce the input dimension and form a suitable distance metric between the reduced inputs. Finally, filtering approaches may be used along with the data-driven predictions of one-step-ahead transition function, when the observations contain nonnegligible noises.

CRediT authorship contribution statement

Mengyang Gu: Conceptualization, Methodology, Software, Formal analysis, Validation, Writing – original draft, Writing – review & editing. **Yizi Lin:** Conceptualization, Methodology, Software, Formal analysis, Validation, Writing – original draft, Writing – review & editing. **Victor Chang Lee:** Software, Validation, Writing – reviewing & editing. **Diana Y. Qiu:** Validation, Writing – review & editing.

Declaration of competing interest

The authors have no conflict of interest to declare.

Data availability

Data and code are publicly available: https://github.com/UncertaintyQuantification/forecast_dynamical_systems.

Acknowledgments

This research is supported by the National Science Foundation under Award No. 2053423. Yizi Lin acknowledge the support from the support of the UC Multicampus Research Programs and Initiatives (MRPI) program under Grant No. M23PL5990. Diana Qiu and Victor Chang Lee were supported by the National Science Foundation (NSF) Condensed Matter and Materials Theory (CMMT) program under Grant DMR-2114081. Development of the td-aGW code was supported by Center for Computational Study of Excited-State Phenomena in Energy Materials (C2SEPEM) at the Lawrence Berkeley National Laboratory, funded by the U.S. Department of Energy, Office of Science, Basic Energy Sciences, Materials Sciences and Engineering Division, under Contract No. DE-C02-05CH11231. The calculations used resources of the National Energy Research Scientific Computing (NERSC), a DOE Office of Science User Facility operated under Contract DE-AC02-05CH11231; Anvil at Purdue University through allocation PHY230053 from the Advanced Cyberinfrastructure Coordination Ecosystem: Services & Support (ACCESS) program, which is supported by National Science Foundation grants #2138259, #2138286, #2138307, #2137603, and #2138296 [89]; and the INCITE program at the Oak Ridge Leadership Computing Facility, which is a DOE Office of Science User Facility supported under Contract DE-AC05-00OR22725.

Appendix. Estimation of range and nugget parameters in PP-GP

The range and nugget parameters of PP-GP model can be estimated from mode estimator, such as the maximum likelihood estimator (MLE) or maximum marginal posterior estimator (MMPE). The MLE can be unstable for estimating the these parameters when the sample size is small. Transforming the range parameters to define the inverse range parameter $\beta_l = 1/\gamma_l$, we use the MMPE for estimating the inverse range and nugget parameters [33]:

$$(\hat{\boldsymbol{\beta}}, \hat{\eta}) = \text{argmax}_{\boldsymbol{\beta},\eta} \left\{ \log(\mathcal{L}(\boldsymbol{\beta}, \eta)) + \log(\pi(\boldsymbol{\beta}, \eta)) \right\}. \tag{39}$$

Here the logarithm of the marginal likelihood after integrating out the mean and variance is

$$\log(\mathcal{L}(\boldsymbol{\beta}, \eta)) = c_1 - \frac{m}{2} \log |\tilde{\mathbf{K}}| - \frac{m}{2} \log |\mathbf{1}_n^T \tilde{\mathbf{K}}^{-1} \mathbf{1}_n| - \left(\frac{n-1}{2} \right) \sum_{j=1}^{m} \log \left(S_j^2 \right), \tag{40}$$

where $S_j^2 = (\mathbf{y}_j - \hat{\mu}_j \mathbf{1}_n)^T \tilde{\mathbf{K}}^{-1} (\mathbf{y}_j - \hat{\mu}_j \mathbf{1}_n)$ and c_1 is a normalizing constant not related to $(\boldsymbol{\gamma}, \eta)$. The jointly robust prior [90] is used as a default choice for prior in the RobustGaSP package [56]

$$\log(\pi(\boldsymbol{\beta}, \eta)) = c_2 + a \log(\sum_{l=1}^{\tilde{p}} C_l \beta_l + \eta) - b \left(\sum_{l=1}^{\tilde{p}} C_l \beta_l + \eta \right), \tag{41}$$

where c_2 is a normalizing constant not relevant to $(\boldsymbol{\beta}, \eta)$ and the default choice of prior parameters in the RobustGaSP package is $a = 0.2$, $b = n^{-1/\tilde{p}}(a + \tilde{p})$ and $C_l = n^{-1/\tilde{p}} |x_l^{max} - x_l^{min}|$ with x_l^{max} and x_l^{min} being the largest and lowest input values in the lth coordinate, respectively. For deterministic output values, including the cases where numerical error of the simulations is negligible, the nugget η may be set to be zero.

We transform the estimated inverse range parameters back to get $\hat{\gamma}_l = 1/\hat{\beta}_l$, for $l = 1, \ldots, \tilde{p}$ and compute the predictive distribution after integrating the m mean parameters and m variance parameters

$$p\left(y_j\left(\mathbf{x}_{t^*}\right) \mid \mathbf{X}, \mathbf{Y}, \mathbf{x}_{t^*}, \hat{\boldsymbol{\gamma}}, \hat{\eta}\right) = \int p(y_j\left(\mathbf{x}_{t^*}\right) \mid \mathbf{X}, \mathbf{Y}, \mathbf{x}_{t^*}, \hat{\boldsymbol{\gamma}}, \hat{\eta}, \boldsymbol{\mu}, \sigma^2) \pi(\boldsymbol{\mu}, \sigma^2) d\boldsymbol{\mu} d\sigma^2,$$

where the reference prior of mean and variance parameters follows $\pi(\boldsymbol{\mu}, \sigma^2) \propto 1/\prod_{j=1}^{m} \sigma_j^2$, and $p(y_j\left(\mathbf{x}_{t^*}\right) \mid \mathbf{X}, \mathbf{Y}, \mathbf{x}_{t^*}, \hat{\boldsymbol{\gamma}}, \hat{\eta}, \boldsymbol{\mu}, \sigma^2)$ is the conditional distribution of the output at $\mathbf{x}_{t^*}$ at coordinate j. With the assumption that the outputs are independent across different coordinates, the predictive distribution $p\left(y_j\left(\mathbf{x}_{t^*}\right) \mid \mathbf{X}, \mathbf{Y}, \mathbf{x}_{t^*}, \hat{\boldsymbol{\gamma}}, \hat{\eta}\right)$ follows a Student's distribution in Eq. (8).

References

[1] W. Coffey, Y.P. Kalmykov, The Langevin Equation: With Applications to Stochastic Problems in Physics, Chemistry and Electrical Engineering, Vol. 27, World Scientific, 2012.

[2] T. Vicsek, A. Czirók, E. Ben-Jacob, I. Cohen, O. Shochet, Novel type of phase transition in a system of self-driven particles, Phys. Rev. Lett. 75 (6) (1995) 1226.

[3] J. Toner, Y. Tu, Flocks, herds, and schools: A quantitative theory of flocking, Phys. Rev. E 58 (4) (1998) 4828.

[4] H.W. Hethcote, The mathematics of infectious diseases, SIAM Rev. 42 (4) (2000) 599–653.

[5] R.E. Kalman, A new approach to linear filtering and prediction problems, J. Basic Eng. 82 (1) (1960) 35–45.

[6] H.E. Rauch, F. Tung, C.T. Striebel, Maximum likelihood estimates of linear dynamic systems, AIAA J. 3 (8) (1965) 1445–1450.

[7] S.J. Julier, J.K. Uhlmann, Unscented filtering and nonlinear estimation, Proc. IEEE 92 (3) (2004) 401–422.

[8] G. Kitagawa, Monte Carlo filter and smoother for non-Gaussian nonlinear state space models, J. Comput. Graph. Statist. 5 (1) (1996) 1–25.

[9] G. Evensen, Sequential data assimilation with a nonlinear quasi-geostrophic model using Monte Carlo methods to forecast error statistics, J. Geophys. Res.: Oceans 99 (C5) (1994) 10143–10162.

[10] L. Sirovich, Turbulence and the dynamics of coherent structures, parts I, II and III, Quart. Appl. Math. (1987) 561–590.

[11] G. Berkooz, P. Holmes, J.L. Lumley, The proper orthogonal decomposition in the analysis of turbulent flows, Annu. Rev. Fluid Mech. 25 (1) (1993) 539–575.

[12] P.J. Schmid, Dynamic mode decomposition of numerical and experimental data, J. Fluid Mech. 656 (2010) 5–28.

[13] J.H. Tu, C.W. Rowley, D.M. Luchtenburg, S.L. Brunton, J.N. Kutz, On dynamic mode decomposition: Theory and applications, J. Comput. Dyn. 1 (2) (2014) 391–421.

[14] C.W. Rowley, I. Mezić, S. Bagheri, P. Schlatter, D.S. Henningson, Spectral analysis of nonlinear flows, J. Fluid Mech. 641 (2009) 115–127.

[15] S.L. Brunton, M. Budišić, E. Kaiser, J.N. Kutz, Modern Koopman theory for dynamical systems, SIAM Rev. 64 (2) (2022) 229–340, http://dx.doi.org/10.1137/21M1401243.

[16] S. Le Clainche, J.M. Vega, Higher order dynamic mode decomposition, SIAM J. Appl. Dyn. Syst. 16 (2) (2017) 882–925.

[17] M.O. Williams, I.G. Kevrekidis, C.W. Rowley, A data–driven approximation of the Koopman operator: Extending dynamic mode decomposition, J. Nonlinear Sci. 25 (6) (2015) 1307–1346.

[18] E. Kaiser, J.N. Kutz, S.L. Brunton, Sparse identification of nonlinear dynamics for model predictive control in the low-data limit, Proc. R. Soc. Lond. Ser. A Math. Phys. Eng. Sci. 474 (2219) (2018) 20180335.

[19] S.L. Brunton, J.L. Proctor, J.N. Kutz, Discovering governing equations from data by sparse identification of nonlinear dynamical systems, Proc. Natl. Acad. Sci. 113 (15) (2016) 3932–3937, http://dx.doi.org/10.1073/pnas.1517384113, arXiv:https://www.pnas.org/doi/pdf/10.1073/pnas.1517384113 URL https://www.pnas.org/doi/abs/10.1073/pnas.1517384113.

[20] M. Korda, I. Mezić, Linear predictors for nonlinear dynamical systems: Koopman operator meets model predictive control, Automatica 93 (2018) 149–160.

[21] C. Folkestad, D. Pastor, I. Mezic, R. Mohr, M. Fonoberova, J. Burdick, Extended dynamic mode decomposition with learned Koopman eigenfunctions for prediction and control, in: 2020 American Control Conference, IEEE, 2020, pp. 3906–3913.

[22] R.T. Chen, Y. Rubanova, J. Bettencourt, D.K. Duvenaud, Neural ordinary differential equations, Adv. Neural Inf. Process. Syst. 31 (2018).

[23] N. Kovachki, Z. Li, B. Liu, K. Azizzadenesheli, K. Bhattacharya, A. Stuart, A. Anandkumar, Neural operator: Learning maps between function spaces with applications to PDEs, J. Mach. Learn. Res. 24 (89) (2023) 1–97.

[24] Z. Li, N.B. Kovachki, K. Azizzadenesheli, B. liu, K. Bhattacharya, A. Stuart, A. Anandkumar, Fourier neural operator for parametric partial differential equations, in: International Conference on Learning Representations, 2021, URL https://openreview.net/forum?id=c8P9NQVtmnO.

[25] J. Sacks, W.J. Welch, T.J. Mitchell, H.P. Wynn, Design and analysis of computer experiments, Statist. Sci. 4 (4) (1989) 409–423.

[26] M.J. Bayarri, J.O. Berger, R. Paulo, J. Sacks, J.A. Cafeo, J. Cavendish, C.-H. Lin, J. Tu, A framework for validation of computer models, Technometrics 49 (2) (2007) 138–154.

[27] H. Li, M. Zhou, J. Sebastian, J. Wu, M. Gu, Efficient force field and energy emulation through partition of permutationally equivalent atoms, J. Chem. Phys. 156 (18) (2022) 184304.

[28] M.C. Kennedy, A. O'Hagan, Bayesian calibration of computer models, J. R. Stat. Soc. Ser. B Stat. Methodol. 63 (3) (2001) 425–464.

[29] H. Zhao, J. Kowalski, Bayesian active learning for parameter calibration of landslide run-out models, Landslides 19 (8) (2022) 2033–2045.

[30] W. Chang, B.A. Konomi, G. Karagiannis, Y. Guan, M. Haran, Ice model calibration using semicontinuous spatial data, Ann. Appl. Stat. 16 (3) (2022) 1937–1961.

[31] B.J. Shields, J. Stevens, J. Li, M. Parasram, F. Damani, J.I.M. Alvarado, J.M. Janey, R.P. Adams, A.G. Doyle, Bayesian reaction optimization as a tool for chemical synthesis, Nature 590 (7844) (2021) 89–96.

[32] X. Fang, M. Gu, J. Wu, Reliable emulation of complex functionals by active learning with error control, J. Chem. Phys. 157 (21) (2022) 214109.

[33] M. Gu, J.O. Berger, Parallel partial Gaussian process emulation for computer models with massive output, Ann. Appl. Stat. 10 (3) (2016) 1317–1347.

[34] A.W. van der Vaart, J.H. van Zanten, Rates of contraction of posterior distributions based on Gaussian process priors, Ann. Statist. 36 (3) (2008) 1435–1463.

[35] E.N. Lorenz, Predictability: A problem partly solved, in: Proc. Seminar on Predictability, Vol. 1, No. 1, 1996.

[36] M. Gu, X. Fang, Y. Luo, Data-driven model construction for anisotropic dynamics of active matter, PRX Life 1 (1) (2023) 013009.

[37] M.S. Hybertsen, S.G. Louie, Electron correlation in semiconductors and insulators: Band gaps and quasiparticle energies, Phys. Rev. B 34 (1986) 5390–5413, http://dx.doi.org/10.1103/PhysRevB.34.5390, URL https://link.aps.org/doi/10.1103/PhysRevB.34.5390.

[38] M. Rohlfing, S.G. Louie, Electron-hole excitations and optical spectra from first principles, Phys. Rev. B 62 (2000) 4927–4944, http://dx.doi.org/10.1103/PhysRevB.62.4927, URL https://link.aps.org/doi/10.1103/PhysRevB.62.4927.

[39] D. Sangalli, S. Dal Conte, C. Manzoni, G. Cerullo, A. Marini, Nonequilibrium optical properties in semiconductors from first principles: A combined theoretical and experimental study of bulk silicon, Phys. Rev. B 93 (2016) 195205, http://dx.doi.org/10.1103/PhysRevB.93.195205.

[40] L. Keldysh, Diagram technique for nonequilibrium processes, Sov. Phys.—JETP 20 (4) (1965) 1018.

[41] L.P. Kadanoff, G. Baym, Quantum Statistical Mechanics, 1962.

[42] G. Stefanucci, R. van Leeuwen, Nonequilibrium Many-Body Theory of Quantum Systems: A Modern Introduction, first ed., Cambridge University Press, 2013.

[43] C. Attaccalite, M. Grüning, A. Marini, Real-time approach to the optical properties of solids and nanostructures: Time-dependent Bethe–Salpeter equation, Phys. Rev. B 84 (2011) 245110, http://dx.doi.org/10.1103/PhysRevB.84.245110.

[44] Y.-H. Chan, D.Y. Qiu, F.H. da Jornada, S.G. Louie, Giant exciton-enhanced shift currents and direct current conduction with subbandgap photo excitations produced by many-electron interactions, Proc. Natl. Acad. Sci. 118 (25) (2021) e1906938118, http://dx.doi.org/10.1073/pnas.1906938118, arXiv:https://www.pnas.org/doi/pdf/10.1073/pnas.1906938118.

[45] E. Perfetto, D. Sangalli, A. Marini, G. Stefanucci, First-principles approach to excitons in time-resolved and angle-resolved photoemission spectra, Phys. Rev. B 94 (2016) 245303, http://dx.doi.org/10.1103/PhysRevB.94.245303, URL https://link.aps.org/doi/10.1103/PhysRevB.94.245303.

[46] E. Perfetto, D. Sangalli, M. Palummo, A. Marini, G. Stefanucci, First-principles nonequilibrium Green's function approach to ultrafast charge migration in glycine, J. Chem. Theory Comput. 15 (8) (2019) 4526–4534, http://dx.doi.org/10.1021/acs.jctc.9b00170, PMID: 31314524.

[47] E. Perfetto, S. Bianchi, G. Stefanucci, Time-resolved ARPES spectra of nonequilibrium excitonic insulators: Revealing macroscopic coherence with ultrashort pulses, Phys. Rev. B 101 (2020) 041201, http://dx.doi.org/10.1103/PhysRevB.101.041201, URL https://link.aps.org/doi/10.1103/PhysRevB.101.041201.

[48] J. Yin, Y.-h. Chan, F.H. da Jornada, D.Y. Qiu, S.G. Louie, C. Yang, Using dynamic mode decomposition to predict the dynamics of a two-time non-equilibrium Green's function, J. Comput. Sci. 64 (2022) 101843.

[49] J. Yin, Y.-h. Chan, F.H. da Jornada, D.Y. Qiu, C. Yang, S.G. Louie, Analyzing and predicting non-equilibrium many-body dynamics via dynamic mode decomposition, J. Comput. Phys. 477 (2023) 111909.

[50] I. Andrianakis, P.G. Challenor, The effect of the nugget on Gaussian process emulators of computer models, Comput. Statist. Data Anal. 56 (12) (2012) 4215–4228.

[51] M.S. Handcock, M.L. Stein, A Bayesian analysis of kriging, Technometrics 35 (4) (1993) 403–410.

[52] M.J. Bayarri, J.O. Berger, E.S. Calder, K. Dalbey, S. Lunagomez, A.K. Patra, E.B. Pitman, E.T. Spiller, R.L. Wolpert, Using statistical and computer models to quantify volcanic hazards, Technometrics 51 (2009) 402–413.

[53] S. Conti, A. O'Hagan, Bayesian emulation of complex multi-output and dynamic computer models, J. Statist. Plann. Inference 140 (3) (2010) 640–651.

[54] K.R. Anderson, I.A. Johanson, M.R. Patrick, M. Gu, P. Segall, M.P. Poland, E.K. Montgomery-Brown, A. Miklius, Magma reservoir failure and the onset of caldera collapse at Kīlauea Volcano in 2018, Science 366 (6470) (2019).

[55] O. Roustant, D. Ginsbourger, Y. Deville, DiceKriging, DiceOptim: Two R packages for the analysis of computer experiments by Kriging-based metamodeling and optimization, J. Stat. Softw. 51 (1) (2012) 1–55, http://dx.doi.org/10.18637/jss.v051.i01.

[56] M. Gu, J. Palomo, J.O. Berger, RobustGaSP: Robust Gaussian stochastic process emulation in R, R J. 11 (1) (2019) 112–136, http://dx.doi.org/10.32614/RJ-2019-011.

[57] J.O. Berger, V. De Oliveira, B. Sansó, Objective Bayesian analysis of spatially correlated data, J. Amer. Statist. Assoc. 96 (456) (2001) 1361–1374.

[58] B. Hamzi, R. Maulik, H. Owhadi, Simple, low-cost and accurate data-driven geophysical forecasting with learned kernels, Proc. R. Soc. Lond. Ser. A Math. Phys. Eng. Sci. 477 (2252) (2021) 20210326.

[59] B. Hamzi, H. Owhadi, Learning dynamical systems from data: a simple cross-validation perspective, part I: parametric kernel flows, Physica D 421 (2021) 132817.
[60] M. Gu, F. Xie, L. Wang, A theoretical framework of the scaled Gaussian stochastic process in prediction and calibration, SIAM/ASA J. Uncertain. Quant. 10 (4) (2022) 1435–1460, http://dx.doi.org/10.1137/21M1409949.
[61] H. Wendland, Scattered Data Approximation, Vol. 17, Cambridge University Press, 2004.
[62] H. Zhang, Inconsistent estimation and asymptotically equal interpolations in model-based geostatistics, J. Amer. Statist. Assoc. 99 (465) (2004) 250–261.
[63] A.C. Davison, D.V. Hinkley, Bootstrap Methods and their Application, No. 1, Cambridge University Press, 1997.
[64] E. Snelson, Z. Ghahramani, Sparse Gaussian processes using pseudo-inputs, Adv. Neural Inf. Process. Syst. 18 (2006) 1257.
[65] A.V. Vecchia, Estimation and model identification for continuous spatial processes, J. R. Stat. Soc. Ser. B Stat. Methodol. 50 (2) (1988) 297–312.
[66] P.J. Schmid, Dynamic mode decomposition and its variants, Annu. Rev. Fluid Mech. 54 (2022) 225–254.
[67] M. West, P.J. Harrison, Bayesian Forecasting & Dynamic Models, second ed., Springer Verlag, 1997.
[68] J. Durbin, S.J. Koopman, Time Series Analysis By State Space Methods, Vol. 38, OUP Oxford, 2012.
[69] H. Arbabi, I. Mezic, Ergodic theory, dynamic mode decomposition, and computation of spectral properties of the Koopman operator, SIAM J. Appl. Dyn. Syst. 16 (4) (2017) 2096–2126.
[70] I.T. Jolliffe, Principal Component Analysis for Special Types of Data, Springer, 2002.
[71] M.E. Tipping, C.M. Bishop, Probabilistic principal component analysis, J. R. Stat. Soc. Ser. B Stat. Methodol. 61 (3) (1999) 611–622.
[72] M. Gu, W. Shen, Generalized probabilistic principal component analysis of correlated data, J. Mach. Learn. Res. 21 (13) (2020).
[73] G. Petris, S. Petrone, P. Campagnoli, Dynamic linear models, in: Dynamic Linear Models with R, Springer, 2009.
[74] R. Prado, M. West, Time Series: Modeling, Computation, and Inference, Chapman and Hall/CRC, 2010.
[75] G. Evensen, Data Assimilation: The Ensemble Kalman Filter, Springer Science & Business Media, 2009.
[76] M. Roth, G. Hendeby, C. Fritsche, F. Gustafsson, The ensemble Kalman filter: a signal processing perspective, EURASIP J. Adv. Signal Process. 2017 (1) (2017) 1–16.
[77] E. Perfetto, Y. Pavlyukh, G. Stefanucci, Real-time GW: Toward an ab initio description of the ultrafast carrier and exciton dynamics in two-dimensional materials, Phys. Rev. Lett. 128 (2022) 016801, http://dx.doi.org/10.1103/PhysRevLett.128.016801, URL https://link.aps.org/doi/10.1103/PhysRevLett.128.016801.
[78] Y.-H. Chan, D.Y. Qiu, F.H. da Jornada, S.G. Louie, Giant self-driven exciton-floquet signatures in time-resolved photoemission spectroscopy of MoS_2 from time-dependent GW approach, Proc. Natl. Acad. Sci. 120 (32) (2023) e2301957120, http://dx.doi.org/10.1073/pnas.2301957120, arXiv:https://www.pnas.org/doi/pdf/10.1073/pnas.2301957120 URL https://www.pnas.org/doi/abs/10.1073/pnas.2301957120.
[79] D.Y. Qiu, F.H. da Jornada, S.G. Louie, Optical spectrum of MoS_2: Many-body effects and diversity of exciton states, Phys. Rev. Lett. 111 (2013) 216805, http://dx.doi.org/10.1103/PhysRevLett.111.216805, URL https://link.aps.org/doi/10.1103/PhysRevLett.111.216805.
[80] D.Y. Qiu, F.H. da Jornada, S.G. Louie, Screening and many-body effects in two-dimensional crystals: Monolayer MoS_2, Phys. Rev. B 93 (2016) 235435.
[81] M.M. Ugeda, A.J. Bradley, S.-F. Shi, F.H. da Jornada, Y. Zhang, D.Y. Qiu, W. Ruan, S.-K. Mo, Z. Hussain, Z.-X. Shen, F. Wang, S.G. Louie, M.F. Crommie, Giant bandgap renormalization and excitonic effects in a monolayer transition metal dichalcogenide semiconductor, Nature Mater. 13 (12) (2014) 1091–1095, http://dx.doi.org/10.1038/nmat4061.
[82] A. Chernikov, T.C. Berkelbach, H.M. Hill, A. Rigosi, Y. Li, B. Aslan, D.R. Reichman, M.S. Hybertsen, T.F. Heinz, Exciton binding energy and nonhydrogenic rydberg series in monolayer WS_2, Phys. Rev. Lett. 113 (2014) 076802, http://dx.doi.org/10.1103/PhysRevLett.113.076802, URL https://link.aps.org/doi/10.1103/PhysRevLett.113.076802.
[83] G. Wang, A. Chernikov, M.M. Glazov, T.F. Heinz, X. Marie, T. Amand, B. Urbaszek, Colloquium: Excitons in atomically thin transition metal dichalcogenides, Rev. Modern Phys. 90 (2018) 021001, http://dx.doi.org/10.1103/RevModPhys.90.021001, URL https://link.aps.org/doi/10.1103/RevModPhys.90.021001.
[84] J. Deslippe, G. Samsonidze, D.A. Strubbe, M. Jain, M.L. Cohen, S.G. Louie, BerkeleyGW: A massively parallel computer package for the calculation of the quasiparticle and optical properties of materials and nanostructures, Comput. Phys. Comm. 183 (6) (2012) 1269–1289, URL http://www.sciencedirect.com/science/article/pii/S0010465511003912.
[85] P. Giannozzi, S. Baroni, N. Bonini, M. Calandra, R. Car, C. Cavazzoni, D. Ceresoli, G.L. Chiarotti, M. Cococcioni, I. Dabo, A.D. Corso, S. de Gironcoli, S. Fabris, G. Fratesi, R. Gebauer, U. Gerstmann, C. Gougoussis, A. Kokalj, M. Lazzeri, L. Martin-Samos, N. Marzari, F. Mauri, R. Mazzarello, S. Paolini, A. Pasquarello, L. Paulatto, C. Sbraccia, S. Scandolo, G. Sclauzero, A.P. Seitsonen, A. Smogunov, P. Umari, R.M. Wentzcovitch, QUANTUM ESPRESSO: a modular and open-source software project for quantum simulations of materials, J. Phys.: Condens. Matter 21 (39) (2009) 395502, http://dx.doi.org/10.1088/0953-8984/21/39/395502.
[86] P. Scherpelz, M. Govoni, I. Hamada, G. Galli, Implementation and validation of fully relativistic GW calculations: Spin–orbit coupling in molecules, nanocrystals, and solids, J. Chem. Theory Comput. 12 (8) (2016) 3523–3544, http://dx.doi.org/10.1021/acs.jctc.6b00114, PMID: 27331614.
[87] H. Liu, Y. Li, Y.S. You, S. Ghimire, T.F. Heinz, D.A. Reis, High-harmonic generation from an atomically thin semiconductor, Nat. Phys. 13 (3) (2017) 262–265, http://dx.doi.org/10.1038/nphys3946.
[88] P. Lipavský, V. Špička, B. Velický, Generalized Kadanoff-Baym ansatz for deriving quantum transport equations, Phys. Rev. B 34 (1986) 6933–6942, http://dx.doi.org/10.1103/PhysRevB.34.6933, URL https://link.aps.org/doi/10.1103/PhysRevB.34.6933.
[89] T.J. Boerner, S. Deems, T.R. Furlani, S.L. Knuth, J. Towns, ACCESS: Advancing innovation: Nsf's advanced cyberinfrastructure coordination ecosystem: Services & support, in: Practice and Experience in Advanced Research Computing, Association for Computing Machinery, New York, NY, USA, 2023, pp. 173–176, http://dx.doi.org/10.1145/3569951.3597559.
[90] M. Gu, Jointly robust prior for Gaussian stochastic process in emulation, calibration and variable selection, Bayesian Anal. 14 (3) (2019) 857–885.