

Sparsity-promoting algorithms for the discovery of informative Koopman-invariant subspaces

Shaowu Pan^{1,†}, Nicholas Arnold-Medabalimi¹ and Karthik Duraisamy¹

¹Department of Aerospace Engineering, University of Michigan, Francois-Xavier Bagnoud Building, 1320 Beal Ave, Ann Arbor, MI, USA

(Received 24 February 2020; revised 2 January 2021; accepted 23 March 2021)

Koopman decomposition is a nonlinear generalization of eigen-decomposition, and is being increasingly utilized in the analysis of spatio-temporal dynamics. Well-known techniques such as the dynamic mode decomposition (DMD) and its linear variants provide approximations to the Koopman operator, and have been applied extensively in many fluid dynamic problems. Despite being endowed with a richer dictionary of nonlinear observables, nonlinear variants of the DMD, such as extended/kernel dynamic mode decomposition (EDMD/KDMD) are seldom applied to large-scale problems primarily due to the difficulty of discerning the Koopman-invariant subspace from thousands of resulting Koopman eigenmodes. To address this issue, we propose a framework based on a multi-task feature learning to extract the most informative Koopman-invariant subspace by removing redundant and spurious Koopman triplets. In particular, we develop a pruning procedure that penalizes departure from linear evolution. These algorithms can be viewed as sparsity-promoting extensions of EDMD/KDMD. Furthermore, we extend KDMD to a continuous-time setting and show a relationship between the present algorithm, sparsity-promoting DMD and an empirical criterion from the viewpoint of non-convex optimization. The effectiveness of our algorithm is demonstrated on examples ranging from simple dynamical systems to two-dimensional cylinder wake flows at different Reynolds numbers and a three-dimensional turbulent ship-airwake flow. The latter two problems are designed such that very strong nonlinear transients are present, thus requiring an accurate approximation of the Koopman operator. Underlying physical mechanisms are analysed, with an emphasis on characterizing transient dynamics. The results are compared with existing theoretical expositions and numerical approximations.

Key words: low-dimensional models, machine learning

† Email address for correspondence: shawnp@umich.edu

© The Author(s), 2021. Published by Cambridge University Press. This is an Open Access article, distributed under the terms of the Creative Commons Attribution licence (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted re-use, distribution, and reproduction in any medium, provided the original work is properly cited.

1. Introduction

Complex unsteady flow phenomena such as turbulence (Pope 2001), flow instability (Drazin & Reid 2004; Lietz, Johnsen & Kushner 2017) and fluid–structure interactions (Dowell & Hall 2001) are prevalent in many physical systems. To analyse and understand such phenomena, it is useful to extract coherent modes associated with important dynamical mechanisms and track their evolution. Koopman operator theory (Budišić, Mohr & Mezić 2012) offers an elegant framework to reduce spatio-temporal fields associated with nonlinear dynamics as a linear combination of time evolving modes ordered by frequency and growth rates (Rowley *et al.* 2009).

Consider a general continuous nonlinear dynamical system $\dot{\mathbf{x}} = \mathbf{F}(\mathbf{x})$, where the system state \mathbf{x} evolves on a manifold $\mathcal{M} \subset \mathbb{R}^N$. Here $\mathbf{F} : \mathcal{M} \mapsto T\mathcal{M}$ is a vector-valued smooth function and $T\mathcal{M}$ is the tangent bundle, i.e. $\forall p \in \mathcal{M}, \mathbf{F}(p) \in T_p\mathcal{M}$. The aforementioned task of decomposition is equivalent to finding a set of L observables associated with the system, $\{g_i\}_{i=1}^L$ ($g_i : \mathcal{M} \mapsto \mathbb{C}$), that evolve linearly in time while spanning the system state \mathbf{x} . The significance of the above statement is that it represents a global linearization of the nonlinear dynamical system (Budišić *et al.* 2012; Brunton *et al.* 2016a).

Formally, this idea can be traced back to Koopman operator theory for Hamiltonian dynamical systems introduced by Koopman (1931) to study the evolution of observables g on $L^2(\mathcal{M})$, which is a vector space of square integrable functions defined on the manifold \mathcal{M} . More generally, for a certain vector space of observable functions \mathcal{F} defined on the manifold \mathcal{M} and $t > 0$, the Koopman semigroup (parameterized by t) is the set $\{\mathcal{K}_t\}_{t \in \mathbb{R}^+}$: $\mathcal{K}_t : \mathcal{F} \mapsto \mathcal{F}$ that governs the dynamics of observables in the form $\mathcal{K}_t g(\mathbf{x}(0)) \triangleq g(S(t, \mathbf{x}_0)) = g(\mathbf{x}(t))$, where $S(t, \cdot)$ is the flow map of the dynamical system (Mezić 2013). (Note that for a discrete dynamical system resulting from the discretization (with time step Δt) of a continuous dynamical system, the corresponding discrete-time Koopman operator is an element of this semigroup: $\mathcal{K}_{\Delta t}$.) Here $\mathcal{K} = \lim_{t \rightarrow 0} (\mathcal{K}_t f - f)/t$ is referred to as the continuous-time Koopman operator, i.e. Koopman generator. While the Koopman operator is linear over the space of observables, \mathcal{F} is most often infinite dimensional, e.g. $L^2(\mathcal{M})$, which makes the approximation of the Koopman operator a difficult problem. Throughout this work, we assume that $\mathcal{F} = L^2(\mathcal{M})$. Readers interested in a detailed discussion of the choice of \mathcal{F} are referred to Bruce, Zeidan & Bernstein (2019).

A special subspace of \mathcal{F} , referred to as a minimal Koopman-invariant subspace \mathcal{G} , has the following property: for any $\phi \in \mathcal{G}$, for any $t \in \mathbb{R}^+$, $\mathcal{K}_t \phi \in \mathcal{G}$ and $\mathbf{x} \in \mathcal{G}$. Existing techniques such as the extended dynamic mode decomposition (Williams, Kevrekidis & Rowley 2015) are capable of approximating Koopman operators, but typically yield high-dimensional spaces (L -dimensional space in figure 1). In this work we are interested in accurately approximating such a subspace – but with the minimal possible dimension and Koopman invariance – using learning techniques, as illustrated in figure 1. This can yield useful coordinates for a multitude of applications including modal analysis and control (Arbabi, Korda & Mezić 2018).

In general, physical systems governed by partial differential equations, e.g. fluid dynamics, are infinite dimensional. From a numerical viewpoint, the number of degrees of freedom can be related to the spatial discretization (for example, the number of grid points). Although a finite-dimensional manifold can be extracted (Holmes *et al.* 2012), e.g. $O(10)$ – $O(10^3)$ dimensions via proper orthogonal decomposition (POD), finding a Koopman-invariant subspace on such manifolds is still challenging. Currently, the most popular method to approximate the Koopman operator is the dynamic

Extraction of informative Koopman-invariant subspaces

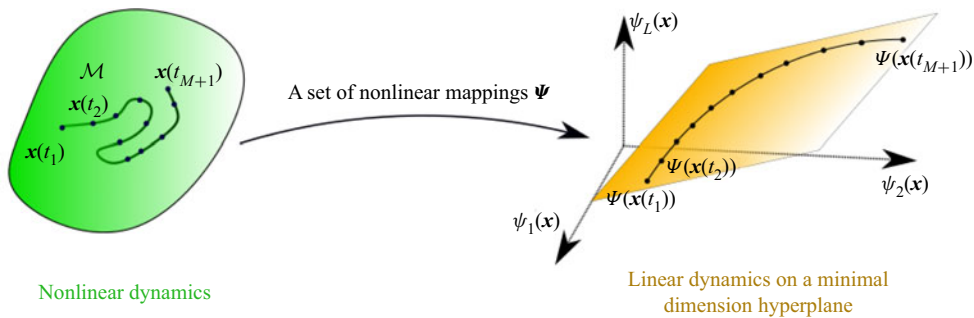


Figure 1. Schematic of transformation of a nonlinear system to a linear L -dimensional space and the extraction of a minimal Koopman-invariant subspace.

mode decomposition (DMD) (Schmid 2010; Rowley & Dawson 2017) mainly for two reasons. First, it is straightforward and computationally efficient compared with nonlinear counterparts such as extended DMD (EDMD) (Williams *et al.* 2015) and kernel DMD (KDMD) (Williams, Rowley & Kevrekidis 2014). Second, the essence of DMD is to decompose a spatio-temporal field into several temporally growing/decaying travelling/stationary harmonic waves, which are prevalent in fluid mechanics. However, the accuracy of DMD is limited by the assumption that the Koopman-invariant subspace lies in the space spanned by snapshots of the state \mathbf{x} . Thus, DMD is used to mainly identify and visualize coherent structures. Indeed, DMD can be interpreted as an L^2 projection of the action of the Koopman operator on the linear space spanned by snapshots of the system state (Korda & Mezić 2018b).

To overcome the above limitations, it is natural to augment the observable space with either the history of the state (Arbabi & Mezić 2017a; Brunton *et al.* 2017; Le Clainche & Vega 2017; Kamb *et al.* 2018) or nonlinear observables of the state (Williams *et al.* 2014, 2015). Time-delay embedding can be very useful in reduced-order modelling of systems for which sparse measurements can be easily obtained, assuming the inputs and outputs are not high dimensional (Korda & Mezić 2018a). Although time-delay embedding is simple to implement and has strong connections to Takens' embedding (Kamb *et al.* 2018; Pan & Duraisamy 2019), the main practical issue arises in reduced-order modelling of high-fidelity simulations in a predictive setting due to the requirement of a large number of snapshots of the full-order model. Furthermore, if one is only interested in the post-transient dynamics of the system state on an attractor, linear observables with time delays are sufficient to extract an informative Koopman-invariant subspace (Mezić 2005; Arbabi & Mezić 2017a,b; Brunton *et al.* 2017; Röjssel 2017; Pan & Duraisamy 2019). However, if one is interested in the strongly nonlinear transient dynamics leading to an attractor or reduced-order modelling for high-fidelity numerical simulations (Carlberg *et al.* 2013; Huang, Duraisamy & Merkle 2018; Xu & Duraisamy 2019; Parish, Wentland & Duraisamy 2020; Xu, Huang & Duraisamy 2020), time-delay embedding may become less appropriate as several delay snapshots of the full-order model are required to initialize the model. In that case, nonlinear observables may be more appropriate.

Driven by the interest in modal analysis and control of transient flow phenomena, we consider augmentation of the observable space with nonlinear functions of the state, e.g. EDMD (Williams *et al.* 2015)/KDMD (Williams *et al.* 2014). Although it has been reported that KDMD allows for a set of more interpretable Koopman eigenvalues (Williams *et al.* 2014) and better accuracy (Röjssel 2017), issues such as mode selection, spurious modes (Kaiser, Kutz & Brunton 2017; Zhang *et al.* 2017) and choice of

dictionary/kernel in EDMD/KDMD remain open. In fact, the choice of kernel type and hyperparameter can significantly affect the resulting eigenmodes, distribution of eigenvalues (Kutz *et al.* 2016) and the accuracy of predictions (Zhang *et al.* 2017).

Searching for an accurate and informative Koopman-invariant subspace has long been a pursuit in the DMD community. Rowley *et al.* (2009) and Schmid, Violato & Scarano (2012) considered selecting dominant DMD modes in order of their amplitudes. However, following such a criterion (Tu *et al.* 2013; Kou & Zhang 2017) may result in the selection of irrelevant modes that may have large amplitudes, but decay rapidly. As a result, Tu *et al.* (2013) considered weighting the loss term by the magnitude of eigenvalues to penalize the retention of fast decaying modes. Sparsity-promoting DMD (referred to as ‘spDMD’ throughout the paper) developed by Jovanović, Schmid & Nichols (2014) recasts mode selection in DMD as an optimization problem with a ℓ_1 penalty. With a preference of stable modes over fast decaying ones, Tissot *et al.* (2014) proposed a simpler criterion based on time-averaged-eigenvalue-weighted amplitude. This was followed by Kou & Zhang (2017) who used a similar criterion but computed the ‘energy’ of each mode, yielding similar performance to spDMD at a lower computational cost. Based on the orthonormal property of pseudo-inverse, Hua *et al.* (2017) proposed an ordering of Koopman modes by defining a new ‘energy’. Compared with previous empirical criteria, the ‘energy’ for each mode involves a pseudo-inverse which combines the influence from all eigenmodes, and, therefore, the contribution from each mode cannot be isolated. Instead of selecting modes from a ‘reconstruction’ perspective, Zhang *et al.* (2017) studied the issue of spurious modes by evaluating the deviation of the identified eigenfunctions from linear evolution in an *a priori* sense. Furthermore, optimized DMD (Chen, Tu & Rowley 2012; Askham & Kutz 2018) combines DMD with mode selection simultaneously, which is the forerunner of recently proposed neural network-based models for Koopman eigenfunctions in spirit (Li *et al.* 2017; Takeishi, Kawahara & Yairi 2017; Lusch, Kutz & Brunton 2018; Otto & Rowley 2019; Yeung, Kundu & Hodas 2019; Pan & Duraisamy 2020). Regardless of the above issues related to non-convex optimization (Dawson *et al.* 2016), extension of optimized DMD to EDMD/KDMD is not straightforward. Further, neural network-based models require large amounts of data, are prone to local minima and lack interpretability. There have been a few attempts towards mode selection in EDMD/KDMD. Brunton *et al.* (2016a) present an iterative method that augments the dictionary of EDMD until a convergence criterion is reached for the subspace. This is effectively a recursive implementation of EDMD. Recently, Haseli & Cortés (2019) showed that given a sufficient amount of data, if there is any accurate Koopman eigenfunction spanned by the dictionary, it must correspond to one of the obtained eigenvectors. Moreover, they proposed the idea of mode selection by checking if the reciprocal of identified eigenvalue also appears when the temporal sequence of data is reversed, which is similar to the idea of comparing eigenvalues on the complex plane from different trajectories, as proposed by Hua *et al.* (2017). In contrast to the ‘bottom-up’ method of Brunton *et al.* (2016a) with subspace augmentation, Hua *et al.* (2017) proposed a ‘top-down’ subspace subtraction method relying on iteratively projecting the features onto the null space. A similar idea can be traced back to Kaiser *et al.* (2017) who propose a search for the sparsest vector in the null space.

As the main contribution of this work, we propose a novel EDMD/KDMD framework equipped with the following strategy to extract an accurate yet minimal Koopman-invariant subspace.

- (i) We first evaluate the normalized maximal deviation of the evolution of each eigenfunction from linear evolution in *a posteriori* fashion.

- (ii) Using the above criteria, we select a user-defined number of accurate EDMD/KDMD modes.
- (iii) Among the accurate EDMD/KDMD modes obtained above, informative modes are selected using multi-task feature learning (Argyriou, Evgeniou & Pontil 2008a; Bach *et al.* 2012).

To the best of our knowledge, this is the first model agnostic attempt to address sparse identification of an accurate minimal Koopman-invariant subspace. Our contribution also extends the classical spDMD (Jovanović *et al.* 2014) in a more general setting that includes EDMD/KDMD. The applications are focused on strongly transient flows, and new classes of stable Koopman modes are identified.

The organization of the paper is as follows: In § 2 we provide a review of EDMD/KDMD in discrete time and present corresponding extensions to continuous time. Following this, we discuss current challenges in standard EDMD/KDMD. In § 3 we propose the framework of sparse identification of informative Koopman-invariant subspaces for EDMD/KDMD with hyperparameter selection and provide implementation details. A novel optimization procedure that penalizes departure from linear evolution is detailed in §§ 3.1 and 3.2. Differences and connections between spDMD, Kou's empirical criterion and our proposed framework is shown from the viewpoint of optimization. In § 4 numerical verification and modal analysis on examples from a fixed point attractor to a transient cylinder wake flow, and a transient ship airwake are provided. In § 5 conclusions are drawn.

2. Review of EDMD/KDMD

In this section we provide a summary of our framework to extract the Koopman operator using EDMD/KDMD in both discrete time and continuous time. Although the original EDMD is seldom used for data-driven Koopman analysis in fluid flows due to its poor scaling, it has recently been reported that a scalable version of EDMD can be applied to high-dimensional systems (DeGennaro & Urban 2019). Since our algorithm to extract the Koopman-invariant subspace is agnostic to the type of technique to compute the original set of Koopman modes, we include both EDMD and KDMD for completeness.

2.1. Extended DMD

2.1.1. Discrete-time formulation

For simplicity, consider M sequential snapshots sampled uniformly in time with Δt on a trajectory, $\{\mathbf{x}_i\}_{i=1}^M$. The EDMD algorithm (Williams *et al.* 2015) assumes a dictionary of L linearly independent functions $i = 1, \dots, L$, $\psi_i(\mathbf{x}) : \mathcal{M} \mapsto \mathbb{C}$ that approximately spans a Koopman-invariant subspace \mathcal{F}_L ,

$$\mathcal{F}_L = \text{span}\{\psi_1(\mathbf{x}), \dots, \psi_L(\mathbf{x})\}. \quad (2.1)$$

Thus, we can write for any $g \in \mathcal{F}_L$, as $g(\mathbf{x}) = \Psi(\mathbf{x})\mathbf{a}$ with $\mathbf{a} \in \mathbb{C}^L$, where the feature vector $\Psi(\mathbf{x})$ is

$$\Psi(\mathbf{x}) = [\psi_1(\mathbf{x}) \quad \dots \quad \psi_L(\mathbf{x})]. \quad (2.2)$$

Consider a set of L' observables as $\{g_l\}_{l=1}^{L'} = \{\Psi\mathbf{a}_l\}_{l=1}^{L'}$, where $\mathbf{a}_l \in \mathbb{C}^L$ is arbitrary. After the discrete-time Koopman operator $\mathcal{K}_{\Delta t}$ is applied on each g_l , given data $\{\mathbf{x}_i\}_{i=1}^M$, we have

the following for $l = 1, \dots, L'$, and $i = 1, \dots, M - 1$,

$$\mathcal{K}_{\Delta t} g_l(\mathbf{x}_i) = g_l(\mathbf{x}_{i+1}) = \Psi(\mathbf{x}_{i+1}) \mathbf{a}_l = \Psi(\mathbf{x}_i) \mathbf{K} \mathbf{a}_l + r_{i,l}, \quad (2.3)$$

where $r_{i,l}$ is simply the residual for the l th function on the i th data point. The standard EDMD algorithm seeks a constant matrix $\mathbf{K} \in \mathbb{C}^{L \times L}$ that governs the linear dynamics in the observable space such that the sum of the square of the residuals $r_{i,l}$ from (2.3) over all samples and functions,

$$J(\mathbf{K}, \{\mathbf{a}_l\}_{l=1}^{L'}) = \sum_{l=1}^{L'} \sum_{m=1}^{M-1} |(\Psi(\mathbf{x}_{m+1}) - \Psi(\mathbf{x}_m) \mathbf{K}) \mathbf{a}_l|^2 = \|(\Psi_{x'} - \Psi_x \mathbf{K}) \mathbf{A}'\|_F^2, \quad (2.4)$$

is minimized over $\{\mathbf{x}_i\}_{i=1}^{M+1}$. In the above equation,

$$\Psi_x = \begin{bmatrix} \psi_1(\mathbf{x}_1) & \dots & \psi_L(\mathbf{x}_1) \\ \vdots & \vdots & \vdots \\ \psi_1(\mathbf{x}_{M-1}) & \dots & \psi_L(\mathbf{x}_{M-1}) \end{bmatrix}, \quad \Psi_{x'} = \begin{bmatrix} \psi_1(\mathbf{x}_2) & \dots & \psi_L(\mathbf{x}_2) \\ \vdots & \vdots & \vdots \\ \psi_1(\mathbf{x}_M) & \dots & \psi_L(\mathbf{x}_M) \end{bmatrix}, \quad (2.5a,b)$$

$$\mathbf{A}' = [\mathbf{a}_1 \quad \dots \quad \mathbf{a}_{L'}]. \quad (2.6)$$

Considering $\partial J / \partial \mathbf{K} = \mathbf{0}$, we obtain $\Psi_x^H \Psi_{x'} \mathbf{A}' \mathbf{A}'^H = \Psi_x^H \Psi_x \mathbf{K} \mathbf{A}' \mathbf{A}'^H$. Thus, the corresponding minimizer \mathbf{K}_{opt} is

$$\mathbf{K}_{opt} = \mathbf{G}^+ \mathbf{A} (\mathbf{A}' \mathbf{A}'^H) (\mathbf{A}' \mathbf{A}'^H)^+, \quad (2.7)$$

where $+$ denotes the pseudo-inverse and

$$\mathbf{G} = \sum_{m=1}^{M-1} \Psi(\mathbf{x}_m)^H \Psi(\mathbf{x}_m) = \Psi_x^H \Psi_x, \quad (2.8)$$

$$\mathbf{A} = \sum_{m=1}^{M-1} \Psi(\mathbf{x}_m)^H \Psi(\mathbf{x}_{m+1}) = \Psi_x^H \Psi_{x'}, \quad (2.9)$$

where H denotes conjugate transpose. Note that when the set of observables fully span \mathcal{F}_L , i.e. \mathbf{A}' is full rank, $(\mathbf{A}' \mathbf{A}'^H) (\mathbf{A}' \mathbf{A}'^H)^+$ reduces to identity. Then we have the more familiar \mathbf{K}_{EDMD} as

$$\mathbf{K}_{EDMD} = \mathbf{G}^+ \mathbf{A}, \quad (2.10)$$

which is independent of the choice of $\{\mathbf{a}_l\}_{l=1}^{L'}$.

Assuming that all eigenvalues of \mathbf{K}_{EDMD} are simple (this is an immediate consequence for ergodic system Parry 2004), for $i = 1, \dots, L$, the corresponding Koopman eigenfunctions φ_i associated with Koopman eigenvalues λ_i are

$$\varphi_i(\mathbf{x}) = \Psi(\mathbf{x}) \mathbf{v}_i, \quad (2.11)$$

where $\mathbf{K}_{EDMD} \mathbf{v}_i = \lambda_i \mathbf{v}_i$. Finally, the Koopman modes of a vector-valued Q -dimensional observable function $\mathbf{g} : \mathcal{M} \mapsto \mathbb{C}^Q$ are the vectors of weights $\{\mathbf{b}_i\}_{i=1}^L$ associated with the

expansion in terms of eigenfunctions $\{\varphi_i\}_{i=1}^L$ as

$$\mathbf{g}(\mathbf{x}) = \sum_{i=1}^L \varphi_i(\mathbf{x}) \mathbf{b}_i, \quad (2.12)$$

where \mathbf{b}_i is often numerically determined in practice by ordinary least squares,

$$\mathbf{B} = \begin{bmatrix} \mathbf{b}_1 \\ \vdots \\ \mathbf{b}_L \end{bmatrix} = (\Psi_x V)^+ \begin{bmatrix} \mathbf{g}(\mathbf{x}_1) \\ \vdots \\ \mathbf{g}(\mathbf{x}_M) \end{bmatrix}, \quad (2.13)$$

with $V = [\mathbf{v}_1 \ \dots \ \mathbf{v}_L]$.

2.1.2. Continuous-time formulation

Consider M data snapshots of the dynamical system with state \mathbf{x} sampled over \mathcal{M} as $\{\mathbf{x}_i, \dot{\mathbf{x}}_i\}_{i=1}^M$, where $\dot{\mathbf{x}}_i = \mathbf{F}(\mathbf{x}_i)$. Recall the generator of the semigroup of Koopman operators $\mathcal{K} : \mathcal{D}(\mathcal{K}) \mapsto \mathcal{F}$, $\mathcal{K} = \lim_{t \rightarrow 0} (\mathcal{K}_t f - f)/t$, where $\mathcal{D}(\mathcal{K})$ is the domain in which the aforementioned limit is well defined and the closure of $\mathcal{D}(\mathcal{K})$ is \mathcal{F} . One can have the evolution of any observable $\mathbf{g} = \Psi \mathbf{a} \in \mathcal{F}_L$ as

$$\mathcal{K} \mathbf{g} = \dot{\mathbf{g}} = \mathbf{F} \cdot \nabla_x \Psi \mathbf{a} = \Psi \mathbf{K} \mathbf{a} + r, \quad (2.14)$$

where r is the residual. Similarly, one can find a \mathbf{K} that minimizes the sum of the square of residual r minimized solution,

$$\mathbf{K}_{EDMD} = \mathbf{G}^+ \mathbf{A}, \quad (2.15)$$

where

$$\mathbf{G} = \sum_{m=1}^M \Psi(\mathbf{x}_m)^H \Psi(\mathbf{x}_m), \quad (2.16)$$

$$\mathbf{A} = \sum_{m=1}^M \Psi(\mathbf{x}_m)^H (\mathbf{F} \cdot \nabla_x) \Psi(\mathbf{x}_m) = \sum_{m=1}^M \Psi(\mathbf{x}_m)^H (\dot{\mathbf{x}}_m \cdot \nabla_x) \Psi(\mathbf{x}_m). \quad (2.17)$$

Consider eigenvalues $\{\mu_i\}_{i=1}^L$ and eigenvectors $\{\mathbf{v}_i\}_{i=1}^L$ of \mathbf{K}_{EDMD} . Koopman eigenfunctions are in the same form as that in discrete-time formulations while continuous-time Koopman eigenvalues μ_i can be converted to the aforementioned discrete-time sense as $\lambda_i = e^{\mu_i \Delta t}$.

2.2. Kernel DMD

2.2.1. Discrete-time formulation

Instead of explicitly expressing a dictionary of candidate functions, one can instead implicitly define a dictionary of candidate functions via the kernel trick, which is essentially the dual form of EDMD (Williams *et al.* 2014). Note that, from the EDMD formulation in (2.4), any vector in the range of \mathbf{K} orthogonal to the range of Ψ_x^H is annihilated, and, therefore, cannot be inferred (Williams *et al.* 2014). Assuming Ψ_x is of rank r , we can obtain a full singular value decomposition (SVD) $\Psi_x = \mathbf{Q} \Sigma \mathbf{Z}^H$ and

the corresponding economical-SVD as $\mathbf{Q}_r \boldsymbol{\Sigma}_r \mathbf{Z}_r^H$. With the aforementioned definitions, we have $\mathbf{K} = \mathbf{Z}_r \hat{\mathbf{K}} \mathbf{Z}_r^H$ without loss (Otto & Rowley 2019).

Since the multiplication by a unitary matrix preserves the Frobenius norm, we have

$$J(\mathbf{K}, \{\mathbf{a}_l\}_{l=1}^{L'}) = \|(\boldsymbol{\Psi}_{x'} - \boldsymbol{\Psi}_x \mathbf{K}) \mathbf{A}'\|_F^2 \quad (2.18)$$

$$= \|(\mathbf{Q}^H \boldsymbol{\Psi}_{x'} - \mathbf{Q}^H \mathbf{Q}_r \boldsymbol{\Sigma}_r \hat{\mathbf{K}} \mathbf{Z}_r^H) \mathbf{A}'\|_F^2 \quad (2.19)$$

$$= \|(\mathbf{Q}_r^H \boldsymbol{\Psi}_{x'} - \boldsymbol{\Sigma}_r \hat{\mathbf{K}} \mathbf{Z}_r^H) \mathbf{A}'\|_F^2 + \|\mathbf{Q}_{r,\perp}^H \boldsymbol{\Psi}_{x'} \mathbf{A}'\|_F^2, \quad (2.20)$$

where $\mathbf{Q}_{r,\perp}^H$ are the last $m - r$ rows of \mathbf{Q}^H . By taking derivatives with respect to $\hat{\mathbf{K}}$, one can obtain the minimal-norm minimizer as

$$\hat{\mathbf{K}}_{opt} = \boldsymbol{\Sigma}_r^+ \mathbf{Q}_r^H \boldsymbol{\Psi}_{x'} \mathbf{A}' \mathbf{A}'^H \mathbf{Z}_r (\mathbf{Z}_r^H \mathbf{A}' \mathbf{A}'^H \mathbf{Z}_r)^+. \quad (2.21)$$

Note that, since any column in the span of \mathbf{A}' that is orthogonal to the span of \mathbf{Z}_r will be annihilated by \mathbf{Z}_r^H and, thus, cannot be utilized to determine $\hat{\mathbf{K}}$, it is reasonable to restrict \mathbf{a}_l within the column space of \mathbf{Z}_r . Assuming L' is sufficiently large such that the column space of \mathbf{A}' fully spans \mathbf{Z}_r , (2.22) can be proved (details in Appendix A),

$$\mathbf{A}' \mathbf{A}'^H \mathbf{Z}_r (\mathbf{Z}_r^H \mathbf{A}' \mathbf{A}'^H \mathbf{Z}_r)^+ = \mathbf{Z}_r. \quad (2.22)$$

With (2.22), we can rewrite (2.21) as the familiar KDMD formulation,

$$\hat{\mathbf{K}}_{KDMD} = \boldsymbol{\Sigma}_r^+ \mathbf{Q}_r^H \boldsymbol{\Psi}_{x'} \mathbf{Z}_r = \boldsymbol{\Sigma}_r^+ \mathbf{Q}_r^H \boldsymbol{\Psi}_{x'} \boldsymbol{\Psi}_x^H \mathbf{Q}_r \boldsymbol{\Sigma}_r^+, \quad (2.23)$$

where $\boldsymbol{\Psi}_x \boldsymbol{\Psi}_x^H = \mathbf{Q}_r \boldsymbol{\Sigma}_r^2 \mathbf{Q}_r^H$ with $(\boldsymbol{\Psi}_x \boldsymbol{\Psi}_x^H)_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$, $(\boldsymbol{\Psi}_{x'} \boldsymbol{\Psi}_x^H)_{ij} = k(\mathbf{x}_{i+1}, \mathbf{x}_j)$ for $1 \leq i, j \leq M - 1$. Again, such a minimizer is independent of the choice of \mathbf{A}' .

Note that, to compute Koopman eigenvalues and eigenfunctions, one would only need access to $\boldsymbol{\Psi}_{x'} \boldsymbol{\Psi}_x^H$ and $\boldsymbol{\Psi}_x \boldsymbol{\Psi}_x^H$, i.e. the inner product among features on all data points. Fortunately, on a compact domain \mathcal{M} , it is well known from Mercer's theorem (Mercer 1909) that once a suitable non-negative kernel function $k(\cdot, \cdot) : \mathcal{M} \times \mathcal{M} \mapsto \mathbb{R}$ is defined, $k(\mathbf{x}, \mathbf{y})$ is the inner product among a potentially infinite-dimensional feature vector $\boldsymbol{\Psi}$ evaluated at $\mathbf{x}, \mathbf{y} \in \mathcal{M}$. Note that the choice of such a feature vector might not be unique but the corresponding reproducing kernel Hilbert space (RKHS) is (Aronszajn 1950). In the case of a Gaussian kernel, one can choose the canonical feature vector $k(\cdot, \mathbf{x})$ which are 'bumps' of a certain bandwidth distributed on \mathcal{M} . From the view point of kernel PCA (Williams *et al.* 2014), \mathbf{Q}_r resulting from the finite-dimensional rank truncation on the Gram matrix $\boldsymbol{\Psi}_x \boldsymbol{\Psi}_x^H$ is a numerical approximation to the r most dominant variance-explained mode shapes in the RKHS evaluated on the data points (Rasmussen 2003), and \mathbf{Z}_r represents the r dominant variance-explaining directions in terms of the feature vector in the RKHS.

Similar to EDMD, given $\hat{\mathbf{K}}_{KDMD} = \hat{\mathbf{V}} \hat{\boldsymbol{\Lambda}} \hat{\mathbf{V}}^{-1}$, $\hat{\mathbf{V}} = [\hat{\mathbf{v}}_1 \dots \hat{\mathbf{v}}_r]$, for $i = 1, \dots, r$, the corresponding i th Koopman eigenfunctions φ_i and Koopman modes for a vector observable \mathbf{g} are

$$\varphi_i(\mathbf{x}) = \boldsymbol{\Psi}(\mathbf{x}) \boldsymbol{\Psi}_x^H \mathbf{Q}_r \boldsymbol{\Sigma}_r^+ \hat{\mathbf{v}}_i, \quad (2.24)$$

$$\mathbf{B} = (\boldsymbol{\Psi}_x \boldsymbol{\Psi}_x^H \mathbf{Q}_r \boldsymbol{\Sigma}_r^+ \hat{\mathbf{V}})^+ \begin{bmatrix} \mathbf{g}(\mathbf{x}_1) \\ \vdots \\ \mathbf{g}(\mathbf{x}_M) \end{bmatrix}. \quad (2.25)$$

Kernel type	$k(\mathbf{x}, \mathbf{x}')$	$\nabla_{\mathbf{x}} k(\mathbf{x}, \mathbf{x}')$	Hyperparameter
Linear	$\mathbf{x}\mathbf{x}'^H$	\mathbf{x}'^H	
Polynomial	$(1 + \mathbf{x}\mathbf{x}'^H)^\alpha$	$\alpha(1 + \mathbf{x}\mathbf{x}'^H)^{\alpha-1} \mathbf{x}'^H$	α
Isotropic Gaussian	$\exp(-\ \mathbf{x} - \mathbf{x}'\ ^2/\sigma^2)$	$\frac{-2(\mathbf{x} - \mathbf{x}')^H}{\sigma^2} \exp(-\ \mathbf{x} - \mathbf{x}'\ ^2/\sigma^2)$	σ

Table 1. Common choice of differentiable kernel functions.

2.2.2. Continuous-time formulation

To the best of our knowledge, continuous-time KDMD has not been previously reported in the literature. This can be helpful when non-uniform multi-scale samples are collected. For the kernel trick to be applied in the continuous formulation, we write $\Psi_{\mathbf{x}'}$ as

$$\Psi_{\mathbf{x}'} = \begin{bmatrix} F(\mathbf{x}_1) \cdot \nabla_{\mathbf{x}} \Psi(\mathbf{x}_1) \\ \vdots \\ F(\mathbf{x}_M) \cdot \nabla_{\mathbf{x}} \Psi(\mathbf{x}_M) \end{bmatrix} = \begin{bmatrix} \dot{\mathbf{x}}_1 \cdot \nabla_{\mathbf{x}} \Psi(\mathbf{x}_1) \\ \vdots \\ \dot{\mathbf{x}}_M \cdot \nabla_{\mathbf{x}} \Psi(\mathbf{x}_M) \end{bmatrix}. \quad (2.26)$$

To compute $\Psi_{\mathbf{x}'} \Psi_{\mathbf{x}}^H$, denoting the q th component of F as f_q ,

$$\begin{aligned} (\Psi_{\mathbf{x}'} \Psi_{\mathbf{x}}^H)_{ij} &= F(\mathbf{x}_i) \cdot \nabla_{\mathbf{x}} \Psi(\mathbf{x}_i) \Psi^H(\mathbf{x}_j) \\ &= \sum_{l=1}^L \sum_{q=1}^N \left(f_q(\mathbf{x}) \frac{\partial \psi_l(\mathbf{x})}{\partial x_q} \right) \bigg|_{\mathbf{x}=\mathbf{x}_i} \overline{\psi_l(\mathbf{x})} \big|_{\mathbf{x}=\mathbf{x}_j} \\ &= \sum_{q=1}^N f_q(\mathbf{x}_i) \frac{\partial}{\partial x_q} \sum_{l=1}^L (\psi_l(\mathbf{x}) \overline{\psi_l(\mathbf{x}')}) \bigg|_{\mathbf{x}=\mathbf{x}_i, \mathbf{x}'=\mathbf{x}_j} \\ &= F(\mathbf{x}_i) \cdot \nabla_{\mathbf{x}} k(\mathbf{x}, \mathbf{x}') \big|_{\mathbf{x}=\mathbf{x}_i, \mathbf{x}'=\mathbf{x}_j} \\ &= \dot{\mathbf{x}}_i \cdot \nabla_{\mathbf{x}} k(\mathbf{x}, \mathbf{x}') \big|_{\mathbf{x}=\mathbf{x}_i, \mathbf{x}'=\mathbf{x}_j}, \end{aligned} \quad (2.27)$$

where the overline symbol is the complex-conjugate operator, and the appearance of Jacobian indicates that a differentiable kernel function is required for the extension to continuous time. For common kernels used in Koopman analysis, the kernel function, Jacobian and hyperparameters are listed in [table 1](#).

2.3. Challenges in EDMD/KDMD

In this section we briefly discuss two broad challenges in the use of EDMD and KDMD for Koopman analysis.

2.3.1. Mode selection

The number of approximated Koopman tuples (eigenfunction, eigenvalue, modes) from EDMD grows with the dictionary size, whereas the KDMD grows with the number of snapshots. However, in most cases, a significant number of the eigenfunctions fail to evolve linearly, or are redundant in contribution to the reconstruction of the state \mathbf{x} . For example, as shown by Budišić *et al.* (2012), the Koopman eigenfunctions that vanish nowhere form an Abelian group under pointwise products of functions, while polynomial observables evolve linearly for a general linear system. These eigenfunctions, associated with the

polynomial observables, are redundant in terms of providing an intrinsic coordinate for the linear dynamics.

When the number of features is larger than the number of data snapshots, EDMD eigenvalues can be misleading (Otto & Rowley 2019) and often plagued with spurious eigenfunctions that do not evolve linearly even when the number of data snapshots is sufficient. Analytically, it is clear that a Koopman eigenfunction in the span of the dictionary will be associated with one of the eigenvectors obtained from EDMD, given Ψ_x is full rank, and contains sufficient snapshots M (Haseli & Cortés 2019). Indeed, the EDMD is an L_2 projection of the Koopman operator under the empirical measure (Korda & Mezić 2018b). As a result, we seek a Koopman-invariant subspace following the standard EDMD/KDMD. Since KDMD can be viewed as an efficient way of populating a dictionary of nonlinear features in high-dimensional spaces, the above arguments apply to KDMD as well. It should be noted that numerical conditioning can play a critical role since full rank matrices can be ill-conditioned.

2.3.2. Choice of dictionary (for EDMD) or kernel (for KDMD)

Although the use of a kernel defines an infinite-dimensional feature space, the resulting finite number of effective features can still be affected by both the type of kernel and the hyperparameters in the kernel, as clearly shown by Kutz *et al.* (2016). Compared with EDMD/KDMD, which are based on a fixed dictionary of features, neural network approaches (Lusch *et al.* 2018; Otto & Rowley 2019; Pan & Duraisamy 2020) have the potential to be more expressive in searching for a larger Koopman-invariant subspace. From a kernel viewpoint (Cho & Saul 2009), feedforward neural networks enable adaptation of the kernel function to the data. Such a characteristic could become significant when the underlying Koopman eigenfunction is discontinuous. From an efficiency standpoint, a kernel-guided scalable EDMD (DeGennaro & Urban 2019) may be pursued. This can be achieved by generating kernel-consistent random Fourier features or approximating a few components of the feature vector constructed from Mercer's theorem, i.e. the eigenfunctions of the Hilbert–Schmidt integral operator on the RKHS.

3. Sparse identification of informative Koopman-invariant subspace

To address the challenges described in § 2.3, we develop a novel framework that uses EDMD/KDMD modes to identify a sparse, accurate and informative Koopman-invariant subspace. Our framework first prunes spurious, inaccurate eigenmodes and second determines a sparse representation of the system state x from the accurate eigenmodes. In addition to the training data, as required in standard EDMD/KDMD, a validation trajectory data set is required to avoid overfitting on training data. The terms spEDMD/spKDMD will refer to filtered mode selections of EDMD and KDMD, respectively.

3.1. Pruning spurious modes by a posteriori error analysis

Given a validation trajectory $x(t)$ where $t \in [0, T]$ associated with the nonlinear dynamical system, for $i = 1, \dots, L$, we define the goodness of i th eigenfunctions in a *a posteriori* way as the maximal normalized deviation from linear evolution conditioned on trajectory $x(t)$ as Q_i in the form

$$e_{i,x(0)}(t) = \frac{|\varphi_i(x(t)) - e^{\lambda_i t} \varphi_i(x(0))|}{\|\varphi_i(x)\|_2}, \quad (3.1)$$

$$Q_i \triangleq e_{i,x(0)}^{\max} = \max_t e_{i,x(0)}(t), \quad (3.2)$$

where $\|\varphi_i(\mathbf{x})\|_2 \triangleq \sqrt{(1/T) \int_0^T \varphi_i^*(\mathbf{x}(t))\varphi_i(\mathbf{x}(t)) dt}$. In practice, we evaluate the above integral terms discretely in time. A similar *a priori* and less restrictive method has been previously proposed (Zhang *et al.* 2017). In contrast, in the proposed method, the maximal error is evaluated in an *a posteriori* way to better differentiate spurious modes from accurate ones. For any $1 \leq \hat{L} \leq L$, we can always select top \hat{L} accurate eigenmodes out of L eigenmodes denoting their index in eigen-decomposition as $\{i_1, i_2, \dots, i_{\hat{L}}\}$, i.e. $Q_{i_1} \leq \dots \leq Q_{i_{\hat{L}}} \leq \dots \leq Q_{i_L}$. Then, for the next sparse reconstruction step, we simply use φ defined as follows to reconstruct the state \mathbf{x} ,

$$\varphi_{\hat{L}}(\mathbf{x}(t)) \triangleq [\varphi_{i_1}(\mathbf{x}(t)) \quad \dots \quad \varphi_{i_{\hat{L}}}(\mathbf{x}(t))] \in \mathbb{C}^{\hat{L}}. \quad (3.3)$$

To choose an appropriate \hat{L} to linearly reconstruct the system state \mathbf{x} , we monitor the normalized reconstruction error for the aforementioned set of top \hat{L} accurate eigenmodes in the form

$$R_{\hat{L}} \triangleq \frac{\|(I - \Psi_{\hat{L}} \Psi_{\hat{L}}^+)X\|_F}{\|X\|_F}, \quad (3.4)$$

where I is the identity matrix, and

$$X = \begin{bmatrix} \mathbf{x}_1 \\ \vdots \\ \mathbf{x}_M \end{bmatrix}, \quad \Psi_{\hat{L}} = \begin{bmatrix} \varphi_{\hat{L}}(\mathbf{x}_1) \\ \vdots \\ \varphi_{\hat{L}}(\mathbf{x}_M) \end{bmatrix}. \quad (3.5a,b)$$

As a result, the evaluation of (3.4) for each \hat{L} is of similar expense to least-square regression. For an increasing number of selected eigenfunctions \hat{L} , the reconstruction error $R_{\hat{L}}$ decreases, while the largest linear evolution error $Q_{i_{\hat{L}}}$ increases. Then, a truncation \hat{L} can be defined by the user to strike a balance between linear evolution error $Q_{i_{\hat{L}}}$ and reconstruction error $R_{\hat{L}}$. In the next subsection we will further select a subset of eigenmodes for spanning the minimal Koopman-invariant subspace.

3.2. Sparse reconstruction via multi-task feature learning

Numerical experiments revealed that, in the selected set of \hat{L} most accurate eigenfunctions, two types of redundant eigenfunctions were found:

- (i) nearly constant eigenfunctions with eigenvalues close to zero;
- (ii) pointwise products of Koopman eigenfunctions introduced by nonlinear observables, not useful in linear reconstruction.

To filter the above modes, we consider sparse regression with \hat{L} most accurate eigenfunctions as features and the system state \mathbf{x} as target. Note that, since we have guaranteed the accuracy of selected eigenmodes, one can either choose features *a priori* $\varphi_i(\mathbf{x}(t))$ or *a posteriori* (multi-step prediction) $e^{\lambda_i t} \varphi_i(\mathbf{x}(0))$. Here we choose the latter since it is directly related to prediction, and can actually be reused from the previous step without additional computational cost. We denote the corresponding multi-step prediction feature

matrix as $\hat{\Psi}_{\hat{L}}$,

$$\hat{\Psi}_{\hat{L}} = \begin{bmatrix} \varphi_{\hat{L}}(\mathbf{x}_1) \\ \varphi_{\hat{L}}(\mathbf{x}_1)e^{\Delta t \Lambda_{\hat{L}}} \\ \vdots \\ \varphi_{\hat{L}}(\mathbf{x}_1)e^{(M-1)\Delta t \Lambda_{\hat{L}}} \end{bmatrix} \in \mathbb{C}^{M \times \hat{L}}, \quad (3.6)$$

where $\Lambda_{\hat{L}} = \text{diag}(\lambda_{i_1}, \dots, \lambda_{i_{\hat{L}}})$. Note that similar features $\hat{\Psi}_{\hat{L}}$ were also considered in spDMD (Jovanović *et al.* 2014) and optimized DMD (Chen *et al.* 2012). Finally, the fact that there is no control over the magnitudes of the implicitly defined features in the standard KDMD may cause unequal weighting between different features. Thus, we consider scaling the initial value of all eigenfunctions to be unity in (3.7),

$$\hat{\Psi}_{\hat{L}, \text{scaled}} = \hat{\Psi}_{\hat{L}} \Lambda_{ini}^{-1} = \begin{bmatrix} 1 & \dots & 1 \\ e^{\Delta t \lambda_{i_1}} & \dots & e^{\Delta t \lambda_{i_{\hat{L}}}} \\ \vdots & \vdots & \vdots \\ e^{(M-1)\Delta t \lambda_{i_1}} & \dots & e^{(M-1)\Delta t \lambda_{i_{\hat{L}}}} \end{bmatrix}, \quad (3.7)$$

where

$$\Lambda_{ini} = \text{diag}([\varphi_{i_1}(\mathbf{x}_1) \quad \dots \quad \varphi_{i_{\hat{L}}}(\mathbf{x}_1)]). \quad (3.8)$$

Since \mathbf{x} is finite dimensional, searching for a sparse combination of $\hat{\Psi}_{\hat{L}}$ to reconstruct \mathbf{x} is equivalent to the solution of a multi-task feature learning problem with preference over a relatively small size of features. Note that this type of problem has been studied extensively in the machine learning community (Argyriou *et al.* 2008a; Zhao *et al.* 2015). In this work, given \mathbf{X} and $\hat{\Psi}_{\hat{L}, \text{scaled}}$, we leverage the multi-task ElasticNet (Pedregosa *et al.* 2011) to search for a row-wise sparse \mathbf{B}' , which solves the following convex optimization problem:

$$\mathbf{B}'^* = \arg \min_{\mathbf{B}' \in \mathbb{C}^{\hat{L} \times N}} \frac{1}{2M} \|\mathbf{X} - \hat{\Psi}_{\hat{L}, \text{scaled}} \mathbf{B}'\|_F^2 + \alpha \rho \|\mathbf{B}'\|_{2,1} + \frac{\alpha(1-\rho)}{2} \|\mathbf{B}'\|_F^2 \quad (3.9)$$

and

$$\mathbf{B} = \Lambda_{ini}^{-1} \mathbf{B}'^*. \quad (3.10)$$

Here $\|\cdot\|_{2,1}$ defined in (3.11) is the so-called ℓ_1/ℓ_2 norm for a matrix \mathbf{W} ,

$$\|\mathbf{W}\|_{2,1} \triangleq \sum_i \sqrt{\sum_j \mathbf{W}_{ij}^2} = \sum_i \|\mathbf{w}_i\|_2, \quad (3.11)$$

and \mathbf{W}_i is i th row of \mathbf{W} . This norm is special in that it controls the number of shared features learned across all tasks, i.e. i th Koopman mode \mathbf{b}_i is either driven to a zero vector or not while the standard ℓ_1 only controls the number of features for each task independently.

As a simple illustration, the ℓ_1/ℓ_2 norm for three different $N \times N$ square matrices (here $N = 5$) with 0-1 binary entries is displayed in figure 2. Since $\sqrt{N} \leq 1 + \sqrt{N-1} \leq N$, minimizing the ℓ_1/ℓ_2 norm leads to a penalty on the number of rows. As shown in the second term on the right-hand side of (3.9), minimizing the ℓ_1/ℓ_2 norm penalizes the number of Koopman eigenmodes.

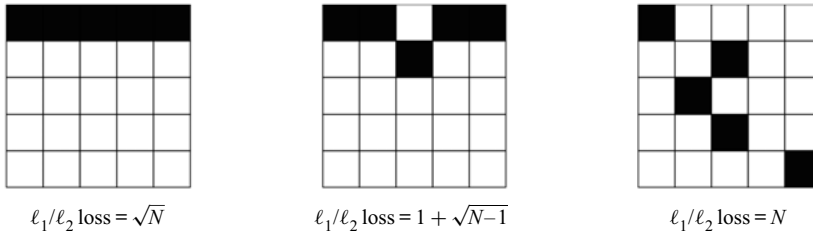


Figure 2. Illustration of ℓ_1/ℓ_2 norm (defined in (3.11)) for different $N \times N$ 0-1 binary matrices.

The above procedure not only serves the purpose of selecting modes that explain the behaviour of all components in the state, but is also particularly natural for EDMD/KDMD since Koopman modes are obtained via regression. Here α is a penalty coefficient that controls the amount of total regularization in the ℓ_1/ℓ_2 and ℓ_2 norms, while ρ is the ElasticNet mixing parameter (Zou & Hastie 2005) that ensures uniqueness of the solution when highly correlated features exist. In our case, we choose $\rho = 0.99$ and sweep α over a certain range with L_r non-zero features denoted as $\hat{\Psi}_{L_r}$ for each α , while monitoring the normalized residual $\min_B \|X - \hat{\Psi}_{L_r} B\|_F / \|X\|_F$ to choose an appropriate α . It has to be mentioned that sometimes the sparsest solution from a multi-task ElasticNet was found to shrink to a small number instead of zero. This is a consequence of the insufficiency of the current optimization algorithm which employs coordinate descent (Pedregosa *et al.* 2011). Hence, for each target component, we consider an additional hard-thresholding step by setting the corresponding magnitude of the coefficient, i.e. contribution of any mode, to zero if it is smaller than a certain threshold $\epsilon \in [10^{-2}, 10^{-3}]$.

Finally, we refit the Koopman modes as $B_{L_r} = \hat{\Psi}_{L_r}^+ X$ which avoids the bias introduced by the penalty term (spDMD does not refit B) in (3.9). To summarize, the general idea of the framework is illustrated in figure 3. As a side note for interested readers, if one only performs multi-task feature learning without hard thresholding and refitting, one would obtain a smooth ElasticNet path instead of a discontinuous one with hard thresholding and refitting. However, the smooth ElasticNet can lead to difficulties in choosing the proper α visually, especially when the given dictionary of EDMD/KDMD is not rich enough to cover an informative Koopman-invariant subspace. Further discussion on the computational complexity of our framework is presented in appendix B.

Thus far, we have presented our main contribution: a novel optimization-based framework to search for an accurate and minimal Koopman-invariant subspace from data. An appealing aspect of our framework is the model agnostic property, which makes the extension easy from the standard EDMD/KDMD to more advanced approximation methods (Azencot, Yin & Bertozzi 2019; Jungers & Tabuada 2019; Mamakoukas *et al.* 2019). In the following subsection we present two mathematical insights: 1) multi-task feature learning generalizes spDMD under a specific constraint; 2) a popular empirical criterion can be viewed as a single step of proximal gradient descent.

3.2.1. Relationship between spDMD, Kou's criterion and multi-task feature learning

For simplicity, neglecting the ElasticNet part (i.e. using $\rho = 1$), (3.9) with L modes leads to a multi-task Lasso problem,

$$\min_{B' \in \mathbb{C}^{L \times N}} \frac{1}{2M} \|X - \hat{\Psi}_{L, \text{scaled}} B'\|_F^2 + \alpha \|B'\|_{2,1}. \quad (3.12)$$

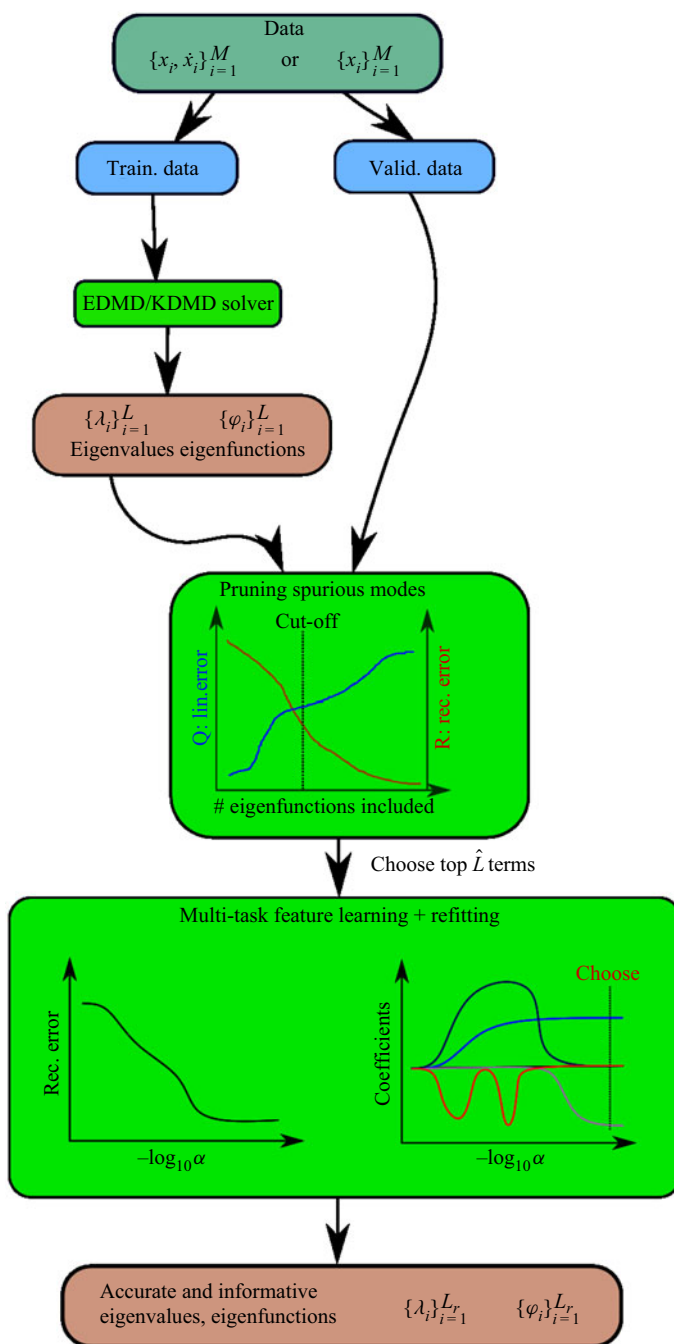


Figure 3. Schematic illustrating the idea of sparse identification of Koopman-invariant subspaces for EDMD and KDMD.

Recall that in spDMD (Jovanović *et al.* 2014), DMD modes ϕ_1, \dots, ϕ_L with $\|\phi_i\|_2 = 1$ remain the same as standard DMD. Similarly, if we posit a structural constraint on \mathbf{B}' in (3.12) by enforcing the modes as those from DMD, then there exist $\alpha_1, \dots, \alpha_L$ such that

$$\mathbf{B}' = \begin{bmatrix} \alpha_1 & & \\ & \ddots & \\ & & \alpha_L \end{bmatrix} \begin{bmatrix} \phi_1^T \\ \vdots \\ \phi_L^T \end{bmatrix}. \quad (3.13)$$

Note the fact that $\|\mathbf{B}'\|_{2,1} = \sum_i^L |\alpha_i|$. Hence, we recover the ℓ_1 optimization step in the spDMD (Jovanović *et al.* 2014) from (3.12) as

$$\min_{\alpha_1, \dots, \alpha_L \in \mathbb{C}} \frac{1}{2M} \left\| \mathbf{X} - \begin{bmatrix} 1 & \dots & 1 \\ \vdots & & \vdots \\ e^{(M-1)\Delta t \lambda_{i_1}} & \dots & e^{(M-1)\Delta t \lambda_{i_L}} \end{bmatrix} \begin{bmatrix} \alpha_1 & & \\ & \ddots & \\ & & \alpha_L \end{bmatrix} \begin{bmatrix} \phi_1^T \\ \vdots \\ \phi_L^T \end{bmatrix} \right\|_F^2 + \alpha \sum_{i=1}^L |\alpha_i|, \quad (3.14)$$

where ϕ_1, \dots, ϕ_L are the DMD modes corresponding to eigenvalues as $\lambda_1, \dots, \lambda_L$. Hence, multi-task feature learning solves a less constrained optimization than spDMD in the context of DMD.

Kou & Zhang (2017) proposed an empirical criterion for mode selection by ordering modes with ‘energy’ I_i defined as

$$I_i = \sum_{j=1}^M |\alpha_i e^{(j-1)\Delta t \lambda_i}| = \begin{cases} \frac{|\alpha_i|(1 - |e^{\Delta t \lambda_i}|^M)}{1 - |e^{\Delta t \lambda_i}|} & \text{if } |e^{\Delta t \lambda_i}| \neq 1, \\ M|\alpha_i| & \text{otherwise.} \end{cases} \quad (3.15)$$

From an optimization viewpoint, consider *a posteriori* prediction matrix \mathbf{X}_{DMD} from DMD,

$$\mathbf{X} \approx \mathbf{X}_{DMD} = \begin{bmatrix} 1 & \dots & 1 \\ \vdots & & \vdots \\ e^{(M-1)\Delta t \lambda_1} & \dots & e^{(M-1)\Delta t \lambda_L} \end{bmatrix} \begin{bmatrix} \alpha_1 & & \\ & \ddots & \\ & & \alpha_L \end{bmatrix} \begin{bmatrix} \phi_1^T \\ \vdots \\ \phi_L^T \end{bmatrix}, \quad (3.16)$$

where \mathbf{X}_{DMD} is determined from DMD using the snapshot pair $(\mathbf{X}, \mathbf{X}')$. Here \mathbf{X}_{DMD} is a rank-1 summation of contributions from different modes (Schmid 2010). Hence, a general mode selection technique with a user-defined preference weighting \mathbf{w} is the following weighted ℓ_0 non-convex optimization problem:

$$\min_{\mathbf{a} \in \mathbb{C}^L} \left\| \mathbf{X}_{DMD} - \begin{bmatrix} 1 & \dots & 1 \\ \vdots & & \vdots \\ e^{(M-1)\Delta t \lambda_1} & \dots & e^{(M-1)\Delta t \lambda_L} \end{bmatrix} \text{diag}(\mathbf{a}) \begin{bmatrix} \phi_1^T \\ \vdots \\ \phi_L^T \end{bmatrix} \right\|_F^2 + \lambda \|\mathbf{a}\|_{\mathbf{w},0}. \quad (3.17)$$

Here $\|\mathbf{a}\|_{\mathbf{w},0} \triangleq \sum_i w_i |a_i|^0$, $|a_i|^0$ is one if $a_i \neq 0$ and zero otherwise. Note that this pseudo-norm can be viewed as a limiting case of a weighted composite sine function smoothed ℓ_0 regularization (Wang *et al.* 2019).

To solve this non-convex optimization problem, compared with the popular ℓ_1 relaxation method such as the one in spDMD, a less known but rather efficient way is iterative least-squares hard thresholding. This has been used in sparse identification of dynamical

systems (SINDy) (Brunton, Proctor & Kutz 2016b), and convergence to a local minimum has been proved (Zhang & Schaeffer 2019). Indeed, a more rigorous framework that generalizes such an algorithm is the proximal gradient method (Parikh & Boyd 2014). Much like Newton's method is a standard tool for unconstrained smooth optimization, the proximal gradient method is the standard tool for constrained non-smooth optimization. Here, it is straightforward to derive the iterative algorithm that extends to the weighted ℓ_0 norm from step k to step $k + 1$ as

$$\mathbf{a}^{k+1} = \text{prox}_{(\lambda/2)\eta_k \|\cdot\|_{w,0}}(\mathbf{a}^k - \eta_k \nabla_a \mathcal{Q}(\mathbf{a}^k)), \quad (3.18)$$

where

$$\mathcal{Q}(\mathbf{a}) = \frac{1}{2} \left\| \mathbf{X}_{DMD} - \begin{bmatrix} 1 & \dots & 1 \\ \vdots & \vdots & \vdots \\ e^{(M-1)\Delta t \lambda_1} & \dots & e^{(M-1)\Delta t \lambda_L} \end{bmatrix} \text{diag}(\mathbf{a}) \begin{bmatrix} \phi_1^T \\ \vdots \\ \phi_L^T \end{bmatrix} \right\|_F^2, \quad (3.19)$$

and η_k is the step size at step k . Note that the weighted ℓ_0 norm is a separable sum of a_i . After some algebra, we have the proximal operator as

$$\text{prox}_{(\lambda/2)\eta_k \|\cdot\|_{w,0}}(\mathbf{a}) = [\mathcal{H}_{\sqrt{\lambda\eta_k}}(a_1/\sqrt{w_1}) \quad \dots \quad \mathcal{H}_{\sqrt{\lambda\eta_k}}(a_L/\sqrt{w_L})]^T, \quad (3.20)$$

where $\mathcal{H}_{\sqrt{\lambda\eta_k}}(a)$ is an element-wise hard-thresholding operator defined as a if $|a| < \sqrt{\lambda\eta_k}$ and zero otherwise.

Particularly if one considers the initial step size to be extremely small $\eta_1 \ll 1$ then the second term in (3.18) can be neglected. Thus, for $i = 1, \dots, L$, with the following weighting scheme that penalizes fast decaying modes,

$$w_i = 1/\beta_i^2, \quad \beta_i = \begin{cases} \frac{1 - |e^{\Delta t \lambda_i}|^M}{1 - |e^{\Delta t \lambda_i}|} & \text{if } |e^{\Delta t \lambda_i}| \neq 1, \\ M & \text{otherwise,} \end{cases} \quad (3.21a,b)$$

one immediately realizes the thresholding criterion for i th entry of \mathbf{a} becomes

$$\sqrt{\lambda\eta_k} > |\alpha_i/\sqrt{w_i}| = |\alpha_i\beta_i|. \quad (3.22)$$

Then plugging (3.21a,b) in (3.18), the first iteration in (3.18) reduces to mode selection with Kou's criterion in (3.15). Normally, β_i is very large for unstable modes and small for decaying modes. It is important to note that (a) such a choice of w preferring unstable/long-lasting modes over decaying modes is still user defined; (b) optimization is in an *a priori* sense to obtain DMD. Thus, the insufficiency of the *a priori* formulation to account for temporal evolution is indeed compensated by this criterion, while DMD in an *a posteriori* formulation (e.g. spDMD) includes such an effect implicitly in the optimization. Hence, it is possible that in some circumstances spDMD and Kou's criterion could achieve similar performance (Kou & Zhang 2017).

Lastly, as summarized in figure 4, it is important to mention the similarities and differences between spKDMD and spDMD: (1) spKDMD will refit Koopman modes while spDMD does not; and (2) the amplitude for all the modes in spKDMD is fixed as unity while it has to be determined in spDMD.

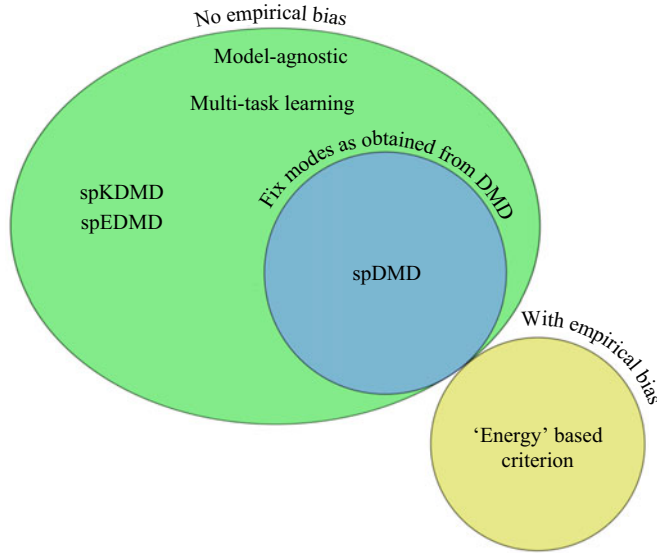


Figure 4. Differences and similarities among existing mode selection methods.

3.3. Hyperparameter selection

For simplicity, hyperparameter selection for KDMD is only discussed in this section. To fully determine a kernel in KDMD, one would have to choose the following:

- (i) kernel type;
- (ii) kernel parameters, e.g. scale parameters σ ;
- (iii) rank truncation r .

In this work, for simplicity, we fix the kernel type to be an isotropic Gaussian. Motivated by previous work on error evaluation in Koopman modes by Zhang *et al.* (2017), we consider evaluation with cross-validation on *a priori* mean normalized accuracy defined in (3.23) and (3.24) for the i th eigenfunction,

$$\text{discrete form: } Q_i^a = \frac{1}{M-1} \sum_{j=1}^{M-1} \frac{|\varphi_i(\mathbf{x}_{j+1}) - \lambda_i \varphi_i(\mathbf{x}_j)|}{\sqrt{\frac{1}{M} \sum_{k=1}^M \varphi_i^*(\mathbf{x}_k) \varphi_i(\mathbf{x}_k)}}, \quad (3.23)$$

$$\text{continuous form: } Q_i^a = \frac{1}{M} \sum_{j=1}^M \frac{|\dot{\mathbf{x}}_j \cdot \nabla_{\mathbf{x}} \varphi_i(\mathbf{x}_j) - \lambda_i \varphi_i(\mathbf{x}_j)|}{\sqrt{\frac{1}{M} \sum_{k=1}^M \varphi_i^*(\mathbf{x}_k) \varphi_i(\mathbf{x}_k)}}, \quad (3.24)$$

on validation data for a different number of rank truncation and kernel parameters.

Note that evaluation on maximal instead of mean normalized accuracy would lead to the error metric to be strongly dependent on the local sparsity of training data in the feature space. This is particularly true for when a single trajectory for a high-dimensional dynamical system is used, and random shuffled cross-validation is performed (Pan & Duraisamy 2018).

For each set of hyperparameters, we first compute the number of eigenfunctions of which the error defined in (3.23) and (3.24) is below a certain threshold on both training and validation data for each fold of cross-validation. Then we compute the average number of such eigenfunctions over all folds which indicates the quality of the corresponding subspace. Finally, we plot the average number vs rank truncation r and kernel scale parameters σ to select hyperparameters.

3.4. Implementation

We implement the described framework in Python with moderate parallelism in each module. We use `scipy.special.hermitemnorm` (Jones, Oliphant & Peterson 2001) to generate normalized Hermite polynomials and `MultiTaskElasticNet` in the `scikit-learn` (Pedregosa *et al.* 2011) for multi-task feature learning where we set the maximal iteration as 10^5 and tolerance as 10^{-12} . Message passing interface parallelism using `mpi4py` (Dalcin *et al.* 2011) is used for the grid search in hyperparameter selection. To prepare data with hundreds of gigabytes collected from high-fidelity simulations, a distributed SVD written in C++ named the parallel data processing (PDP) tool is developed for dimension reduction. A brief description of this tool is given in [appendix D](#).

4. Results and discussion

4.1. Two-dimensional fixed point attractor with the known finite Koopman-invariant subspace

We first consider a simple fixed point nonlinear dynamical system which has an analytically determined, finite-dimensional non-trivial Koopman-invariant subspace (Brunton *et al.* 2016a; Kaiser *et al.* 2017) to show the effectiveness of the proposed method. We consider a continuous-time formulation. The governing equation for the dynamical system is given as

$$\dot{x}_1 = \mu x_1, \quad (4.1)$$

$$\dot{x}_2 = \lambda(x_2 - x_1^2), \quad (4.2)$$

where $\mu = -0.05$, $\lambda = -1$. One natural choice of the minimal Koopman eigenfunctions that linearly reconstructs the state is (Brunton *et al.* 2016a)

$$\varphi_1(\mathbf{x}) = x_2 - \lambda x_1^2 / (\lambda - 2\mu), \quad \varphi_2(\mathbf{x}) = x_1, \quad \varphi_3(\mathbf{x}) = x_1^2, \quad (4.3a-c)$$

with eigenvalues $\lambda = -1$, $\mu = -0.05$, $2\mu = -0.1$, respectively.

The way we generate training, validation and testing data is described below with distribution of the data shown in [figure 5](#).

- (i) Training data: a point cloud with $M = 1600$ pairs of $\{\mathbf{x}^{(i)}, \dot{\mathbf{x}}^{(i)}\}_{i=1}^M$ is generated by Latin hypercube sampling (Baudin 2015) within the domain $x_1, x_2 \in [-0.5, 0.5]$.
- (ii) Validation data: a single trajectory with initial condition as $x_1(0) = x_2(0) = 0.4$, sampling time interval $\Delta t = 0.03754$ from $t = 0$ to $t = 30$.
- (iii) Testing data: a single trajectory with initial condition as $x_1(0) = x_2(0) = -0.3$, sampling time interval $\Delta t = 0.06677$ from $t = 0$ to $t = 40$.

As an illustration, we consider two models to approximate the Koopman operator from training data:

Extraction of informative Koopman-invariant subspaces

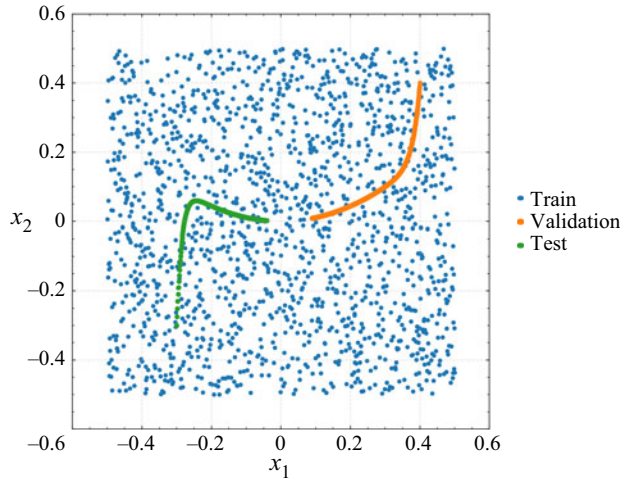


Figure 5. Data distribution for a 2-D fixed point attractor.

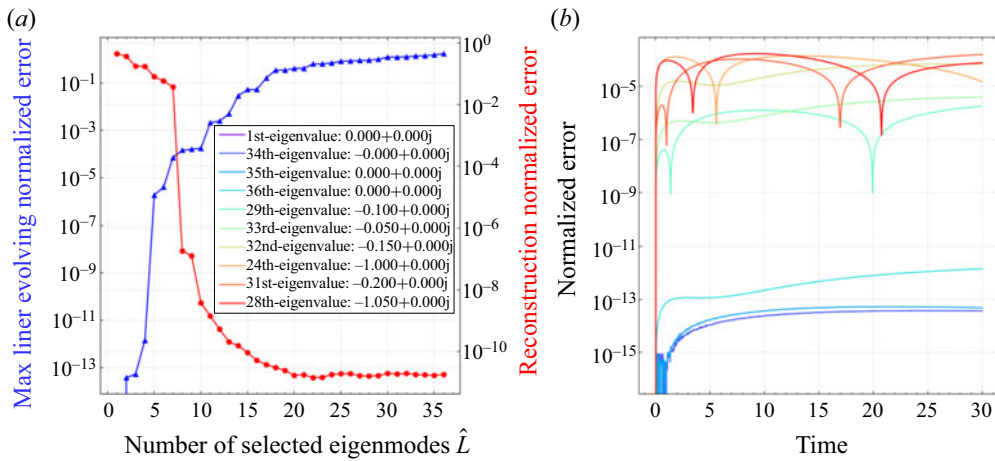


Figure 6. Error analysis of 36 eigenmodes from continuous-time EDMD for the 2-D fixed point attractor. (a) Trends of linearly evolving error Q and reconstruction error R . (b) Temporal evolution of relative error for top $\hat{L} = 10$ accurate eigenmodes.

- (i) a continuous-time EDMD with normalized Hermite polynomials up to fifth-order with $L = 36$ features;
- (ii) a continuous-time KDMD with isotropic Gaussian kernel $\sigma = 2$ with reduced rank $r = L = 36$.

Details of the above choices based on the steps of hyperparameter selection in § 3.3 are given in § C.1.

4.1.1. Results for continuous-time EDMD with mode selection

As displayed in figure 6, we begin with an error analysis of all of the eigenmodes on validation data in figure 5 according to linearly evolving error Q defined in (3.2)

and R defined in (3.4). From figure 6(a), considering both the linearly evolving error and the quality of the reconstruction, we choose the cut-off threshold at $\hat{L} = 10$. We observe a sharp cut-off in figure 6(a) around the number of selected eigenmodes $\hat{L} = 8$. This is a reasonable choice, since from the eigenvalues in figure 6(b), we notice the analytic Koopman eigenmodes are not covered until first eight accurate eigenmodes are selected. Note that the legend in figure 6(a) is ordered by the maximal deviation from linear evolution, e.g. the second most accurate mode is the 34th mode with zero eigenvalue. Indeed, the first four eigenfunctions (index = 1, 34, 35, 36) are redundant in terms of reconstruction in this problem (this could be interesting if the system is instead Hamiltonian). The fifth (index = 29) and sixth (index = 33) eigenmodes correspond to two of the analytic eigenfunctions that span the system, and the seventh (index = 32) eigenmode is indeed the product of the fifth and sixth eigenfunctions. Similarly, the ninth and tenth eigenfunctions (index = 31, 28) also appear to be the polynomial combination of the true eigenfunctions.

According to (3.9), to further remove redundant modes, we perform multi-task feature learning on the $\hat{L} = 10$ eigenmodes. The corresponding ElasticNet path is shown in figure 7. Note that each α corresponds to a minimizer of (3.9). To choose a proper α so as to find a proper Koopman-invariant subspace, it is advisable to check the trend of the normalized reconstruction error and number of non-zero features. Given the dictionary, for simple problems for which there exists an exact Koopman-invariant subspace that also spans system state, a proper model can be obtained by selecting $\alpha \approx 10^{-6}$ which ends up with only three eigenfunctions, as shown in figure 7. Moreover, as is common for EDMD with polynomial basis (Williams *et al.* 2014, 2015), a pyramid of eigenvalues appears in figure 7.

As shown in figure 8, both the identified eigenvalues, and contour of the phase angle and magnitude of selected eigenfunctions from spEDMD match the analytic eigenfunctions given in (4.3a–c) very well. As expected, the prediction on unseen testing data is also excellent. Note that the indices of true eigenfunctions φ_1 , φ_2 and φ_3 ordered by Kou's criterion in (3.15) are 8, 5 and 6. In this case, all of the true eigenfunctions are missing in the top three modes chosen by Kou's criterion. Indeed, the top three modes chosen by Kou's criterion have nearly zero eigenvalues.

4.1.2. Results of continuous-time KDMD with mode selection

The mode selection algorithm presented above can be applied in precisely the same form to KDMD, given a set of eigenfunctions and eigenvalues. Error analysis of eigenfunctions is shown in figure 9, from which we choose $\hat{L} = 10$ as well. As before, eigenvalues ordered by maximal deviation from linear evolution are shown in the legend in figure 9(a). Again, in figure 9(a), we observe a sharp decrease in the reconstruction error after the four most accurate modes are included. This is expected, as the second to fourth most accurate modes are analytically exact from the figure 9(b). As shown in figures 10 and 11, it is confirmed that both spEDMD and spKDMD arrive at the same analytic eigenfunctions with difference up to a constant factor. It should be noted that, although polynomials are not analytically in the RKHS (Minh 2010), good approximations can still be achieved conditioned on the data we have, i.e. $x_1, x_2 \in [-0.5, 0.5]$. Again, the indices of true eigenfunctions φ_1 to φ_3 ordered by Kou's criterion are 8, 2 and 3. Hence, φ_1 is missing in the top three modes chosen by Kou's criterion. Similarly, the first mode chosen by Kou's criterion has a zero eigenvalue.

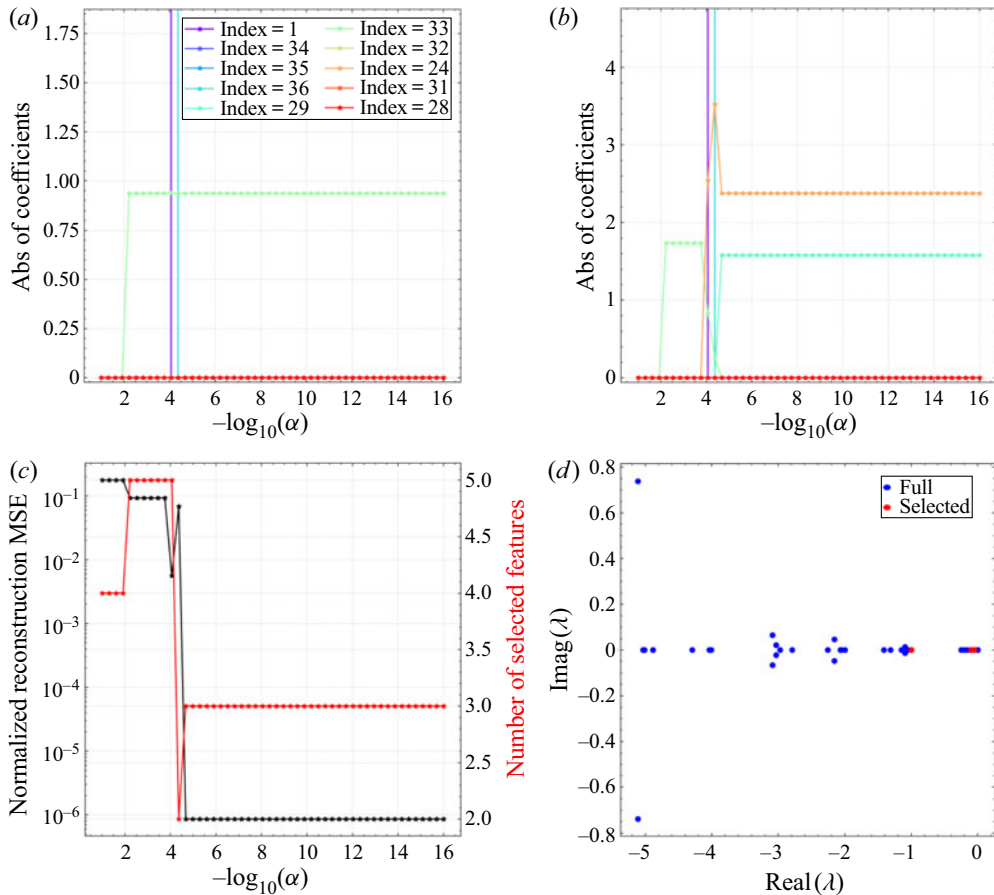


Figure 7. Result of multi-task feature learning on top $\hat{L} = 10$ accurate eigenmodes from continuous-time EDMD for the 2-D fixed point attractor. (a) ElasticNet path for x_1 . (b) ElasticNet path for x_2 . (c) Trends of normalized reconstruction error and number of non-zero terms vs α . (d) Selected continuous-time eigenvalues.

4.1.3. Effect of SVD regularization

Singular value decomposition truncation is a standard regularization technique in the solution of a potentially ill-conditioned linear system. In the standard EDMD in (2.7), for example, G could be potentially ill-conditioned, leading to spurious eigenvalues in K . Hence, Williams *et al.* (2014) recommend SVD truncation in (2.8) to obtain a robust solution of K . Effectively, it shrinks the number of EDMD/KDMD modes. It has to be recognized, however, that the mode reduction from SVD truncation is not the same as mode selection. Most importantly, one should not confuse numerical spuriousness from poor numerical conditioning with functional spuriousness from the orthogonal projection error of the Koopman operator (Korda & Mezić 2018b). Indeed, SVD truncation does not always lead to better approximation of a Koopman-invariant subspace. It is rather a linear dimension reduction that optimally preserves the variance in the feature space conditioned on the training data without knowing the linear evolution property of each feature.

For demonstration, we take the above fixed point attractor system where we use the same data and standard EDMD algorithm with the same order of Hermite polynomials. The results of prediction on the unseen testing data shown in figure 12 indicate that even

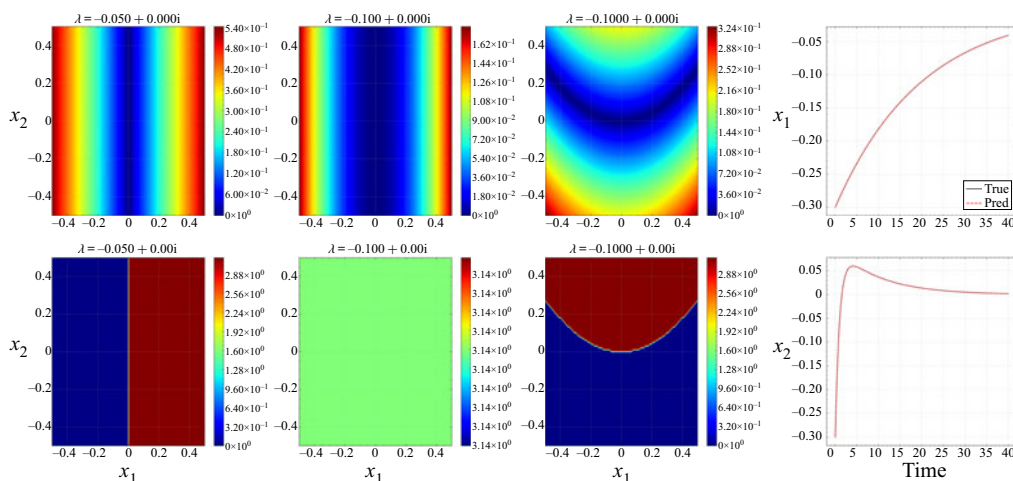


Figure 8. Sparsely selected eigenfunctions and eigenvalues from continuous-time EDMD for the 2-D fixed point attractor with the corresponding prediction on testing data with an unseen initial condition $x_1(0) = x_2(0) = -0.3$. From left to right, the top three figures show contours of magnitude of eigenfunctions, while the bottom three figures are those of the phase angle of eigenfunctions. Last column: comparison between prediction and ground truth for an unseen testing trajectory.

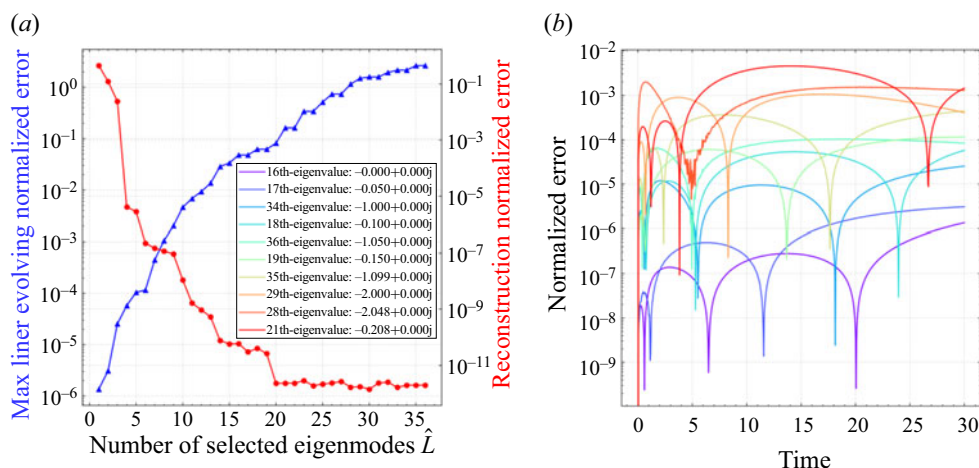


Figure 9. Error analysis of 36 eigenmodes from continuous-time KDMD for the 2-D fixed point attractor. (a) Trends of linearly evolving error Q and reconstruction error R . (b) Temporal evolution of relative error for top $\hat{L} = 10$ accurate eigenmodes.

though only three eigenfunctions (indeed three features in the Hermite polynomial) are required, standard EDMD fails to identify the correct eigenfunctions with 26 SVD modes while the results improve with 31 modes retained. The sensitivity of standard EDMD with respect to SVD truncation is likely a result of the use of normalized Hermite polynomials where SVD truncation would lead to a strong preference over the subspace spanned by the higher-order polynomials. We did not observe such a sensitivity for KDMD, unless the subspace is truncated below 10 modes.

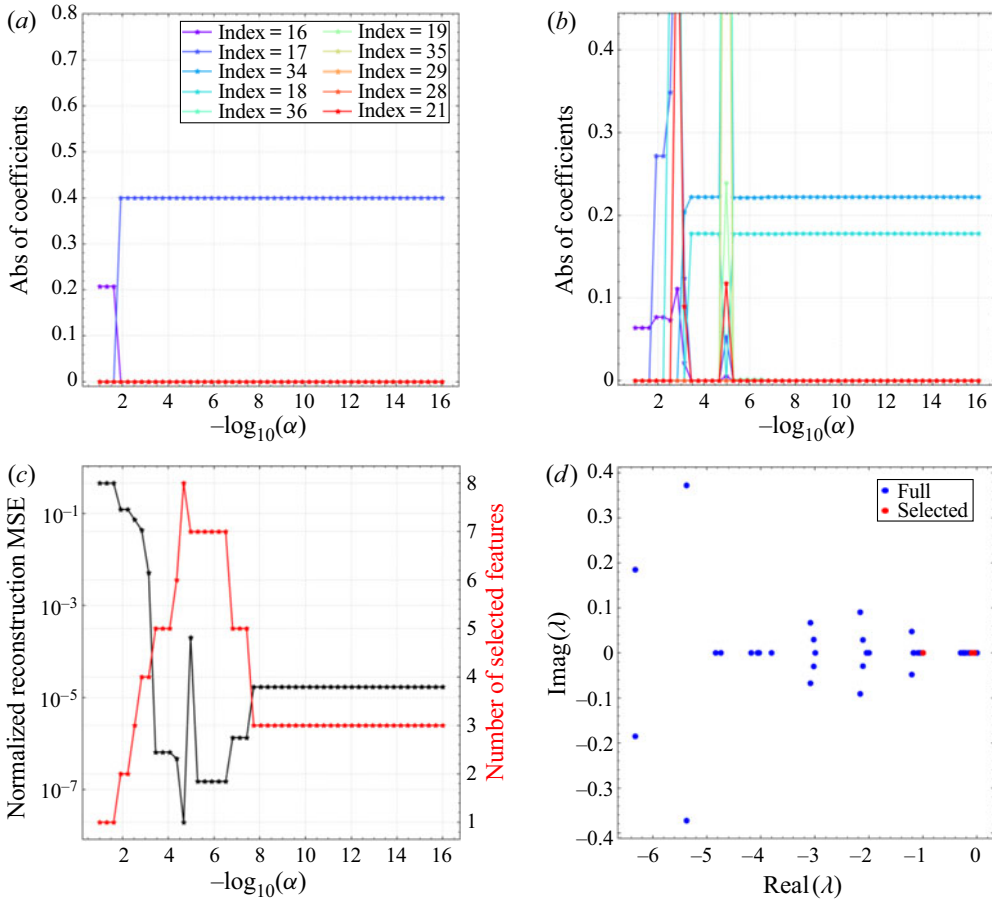


Figure 10. Result of multi-task feature learning on top $\hat{L} = 10$ accurate eigenmodes from continuous-time KDMD for the 2-D fixed point attractor. (a) ElasticNet path for x_1 . (b) ElasticNet path for x_2 . (c) Trends of normalized reconstruction error and number of non-zero terms vs α . (d) Selected continuous-time eigenvalues.

4.2. Two-dimensional, transient flow past a cylinder

As a classical example for Koopman analysis in fluid dynamics (Bagheri 2013; Williams *et al.* 2014; Otto & Rowley 2019), transient two-dimensional flow past a cylinder (figure 13) is considered at different Reynolds numbers ($Re = U_\infty D/\nu$), where $U_\infty = 1$ is the free stream velocity, $D = 2$ is the diameter of the cylinder and ν is the kinematic viscosity. The two-dimensional incompressible Navier–Stokes equations govern the dynamics with far-field boundary conditions for pressure and velocity and no-slip velocity on the cylinder surface. Numerical simulations are performed using the icoFoam solver in OpenFOAM (Jasak, Jemcov & Tukovic 2007) solving the two-dimensional incompressible Navier–Stokes equations. We explore $Re = 70, 100, 130$ by changing the viscosity. The pressure field is initialized with independent and identically distributed (i.i.d.) Gaussian noise $\mathcal{N}(0, 0.3^2)$. The velocity is initialized with a uniform free stream velocity superimposed with i.i.d. Gaussian noise $\mathcal{N}(0, 0.3^2)$. It should be noted that the noise is generated on the coarsest mesh shown in figure 13, and interpolated to the finer meshes. Grid convergence with increasing mesh resolution is assessed by comparing the temporal evolution of the drag coefficient C_D and lift coefficient C_L .

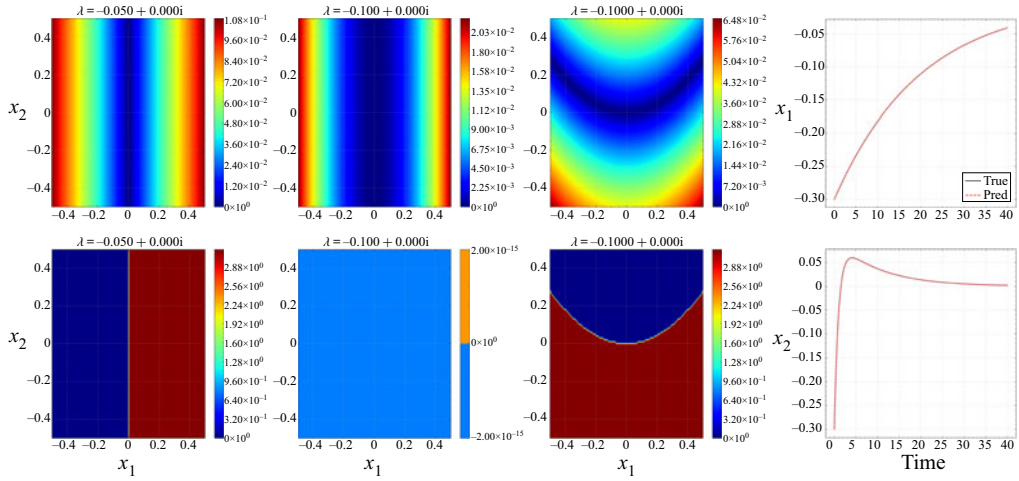


Figure 11. Sparsely selected eigenfunctions and eigenvalues from continuous-time KDMD for the 2-D fixed point attractor with corresponding prediction on testing data with an unseen initial condition $x_1(0) = x_2(0) = -0.3$. From left to right, the top three figures show contours of the magnitude of eigenfunctions, while the bottom three figures are those of the phase angle of eigenfunctions. Last column: comparison between predictions and ground truth for an unseen testing trajectory.

Note that the dynamics of a cylinder wake involves four regimes: near-equilibrium linear dynamics, nonlinear algebraic interaction between equilibrium and the limit cycle, exponential relaxation rate to the limit cycle and periodic limit cycle dynamics (Chen *et al.* 2012; Bagheri 2013). Instead of considering data only from each of these regimes separately (Chen *et al.* 2012; Taira *et al.* 2019) or with only the last two regimes where exponential linear dynamics is expected (Bagheri 2013), we start collecting data immediately after the flow field becomes unstable, and stop after the flow field experiences several limit cycles. Note that the regime with algebraic interaction is non-modal (Schmid 2007) and, therefore, cannot be expressed as individual exponential terms (Bagheri 2013). This becomes a challenging problem for DMD (Chen *et al.* 2012). The sampling time interval is $\Delta t = 0.1t_{ref}$, where $t_{ref} = D/U_\infty$.

For each Re , 891 snapshots of full velocity field U and V with sampling time interval Δt are collected as two matrices of size $N_{grid} \times N_{snapshots}$. Following this, each velocity component is shifted and scaled (normalized) between $[-1, 1]$. Since space-filling sampling in any high-dimensional space would be extremely difficult, we split the trajectory into training, validation and testing data by sampling with strides similar to the ‘even-odd’ sampling scheme previously proposed by Otto & Rowley (2019). As illustrated in figure 14, given index i , if $i \bmod 3 = 0$, the i th point belongs to training set while $i \bmod 3 = 1$ corresponds to validation, and $i \bmod 3 = 2$ for testing data. Consequently, the time interval in the training, testing, validation trajectory is tripled as $3\Delta t = 0.3t_{ref}$. Thus, training, validation and testing data are split into 297 snapshots each. Finally, we stack data matrices along the first axis corresponding to the number of grid points, and perform a distributed SVD described in § 3.4. For all three cases, the top 20 POD modes are retained, corresponding to 99 % of kinetic energy. Next, we apply our algorithm to discrete-time KDMD with isotropic Gaussian kernel on this reduced-order nonlinear system. We choose the hyperparameters $\sigma = 3$ and $r = 180$. Further details are given in § C.2.

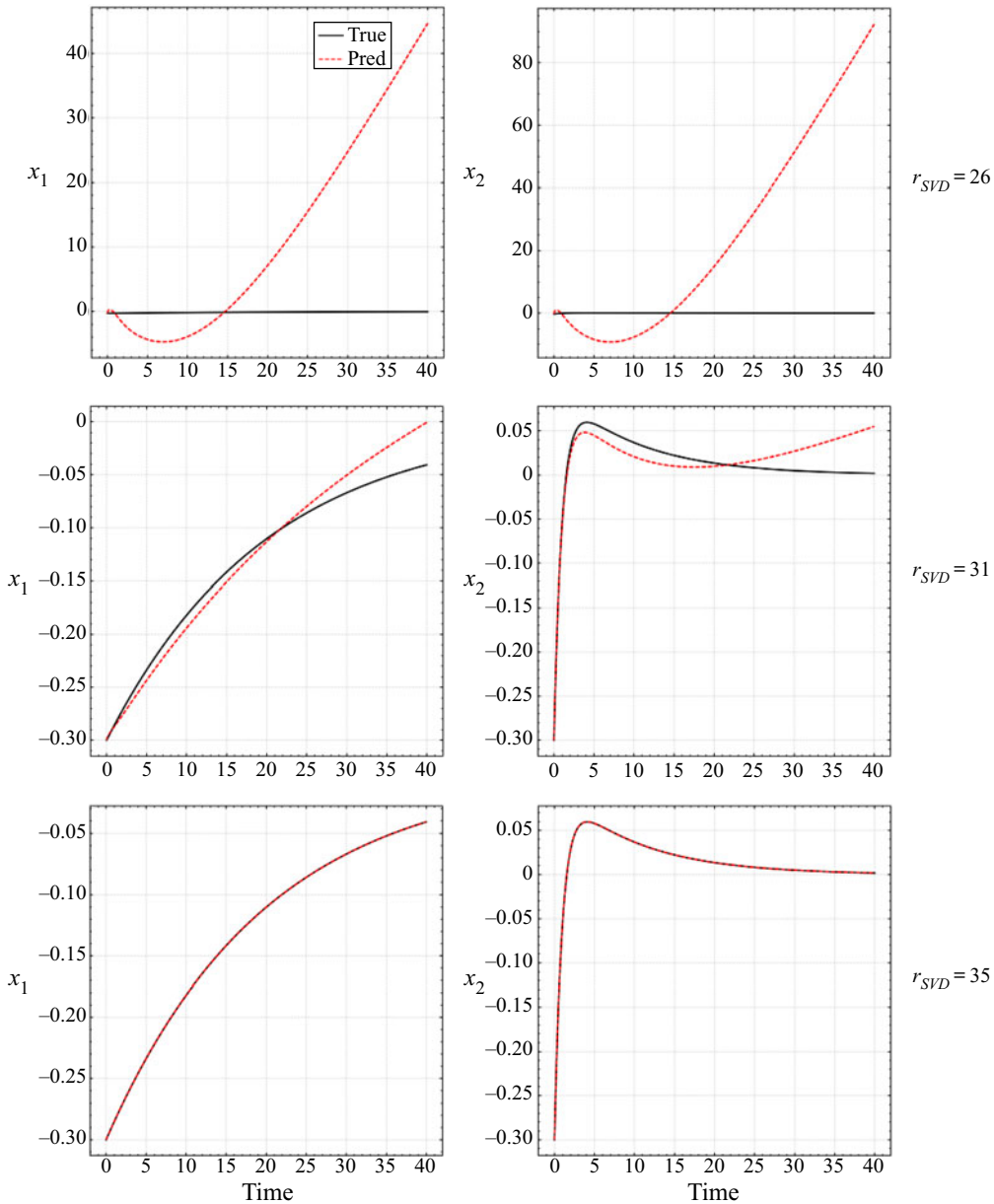


Figure 12. Standard EDMD prediction on an unseen trajectory with different SVD truncations for a fixed point attractor.

4.2.1. Results of discrete-time KDMD with mode selection

For all three Reynolds numbers, *a posteriori* error analysis is shown in figure 15. A good choice of the number of accurate modes \hat{L} retained for reconstruction is around 60 since the corresponding maximal deviation from linear evolution is still around 5 % while the reconstruction error reaches a plateau after $\hat{L} > 60$.

After the mode selection on validation data, a α -family of solutions is obtained with corresponding reconstruction error and the number of non-zero terms as shown

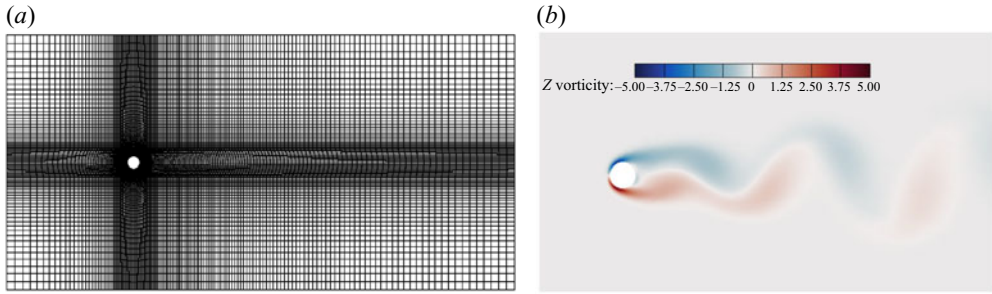


Figure 13. (a) Illustration of computational mesh for a two-dimensional cylinder wake problem (coarsest). (b) Contour of vorticity ω_z for $Re = 70$ when vortex shedding is fully developed ($t = 175$).

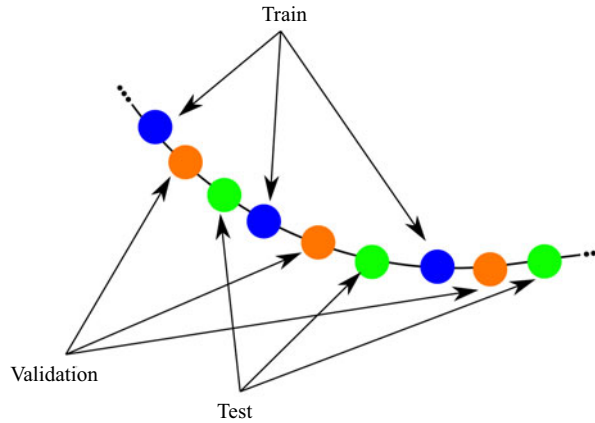


Figure 14. Illustration of splitting a uniformly sampled single trajectory in high-dimensional phase space into training, validation and testing sets.

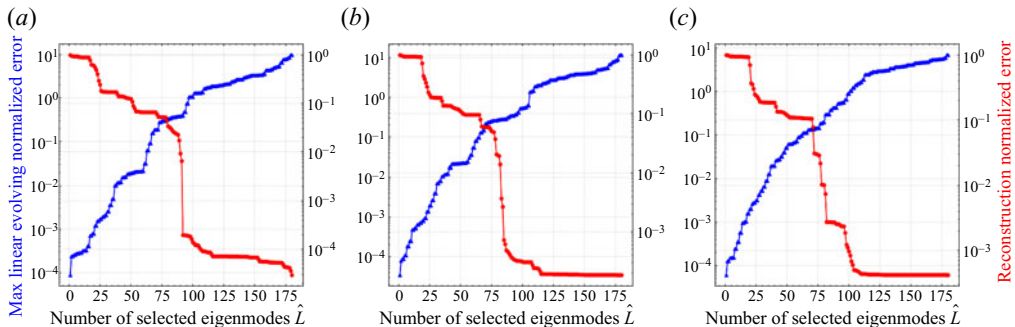


Figure 15. Trend of linear evolution error Q and reconstruction error R from discrete-time KDMD for the two-dimensional cylinder wake flow case; (a) $Re = 70$, (b) $Re = 100$, (c) $Re = 130$.

in figure 16. Note that the chosen solution is highlighted as blue circles. As shown in table 2, nearly half of the accurate KDMD eigenmodes identified are removed with the proposed sparse feature selection. Note that for all three cases, the number of selected modes (around 32 to 34) is still larger than that required in neural network models (around 10) (Otto & Rowley 2019; Pan & Duraisamy 2020). This is because the subspace spanned by KDMD/EDMD relies on a predetermined dictionary rather than being data-adaptive

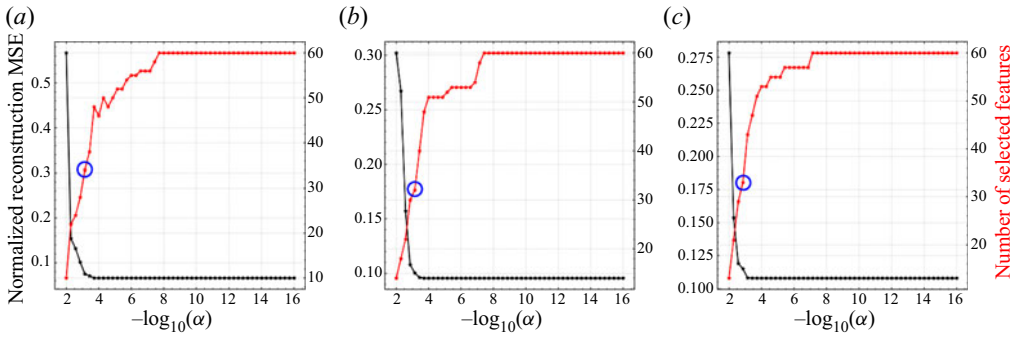


Figure 16. Variation of reconstruction error R and number of non-zero terms for the two-dimensional cylinder wake flow; (a) $Re = 70$, (b) $Re = 100$, (c) $Re = 130$. The blue circle corresponds to selected α .

	$Re = 70$	$Re = 100$	$Re = 130$
α_{select}	7.19×10^{-4}	7.19×10^{-4}	1.39×10^{-3}
Number of selected modes	34	32	33
Number of total modes	297	297	297
Normalized reconstruction error R	0.075	0.105	0.113

Table 2. Summary of mode selection for discrete-time KDMD on two-dimensional cylinder wake flow.

like neural network models. Nevertheless, due to the additional expressiveness from nonlinearity, we will see in § 4.2.3 that spKDMD performs significantly better than DMD (Schmid 2010) and spDMD (Jovanović *et al.* 2014), while enjoying the property of convex optimization at a much lower computational cost than the inherently non-convex and computationally intensive neural network counterparts.

The predictions of the top eight POD coefficients (denoted as x_1 to x_8) on testing data are displayed in figures 17–19. The results match very well with ground truth for all three cases. Figure 21 shows that there appear to be five clusters of selected eigenvalues while most of the modes are removed by the proposed algorithm. Similar observations were also made in Bagheri (2013) when DMD is applied to the full transient dynamics. This pattern consisting of a stable eigenvalue on the unit circle surrounded by several decaying eigenvalues is observed for all clusters. The stable eigenvalue contributes to limit cycle behaviour, while the decaying eigenvalues account for the transient phenomenon. Due to symmetry, only eigenvalues in the first quadrant are shown in the bottom row of figure 21. It is observed that the frequency associated with the type-II cluster is approximately twice that of type-I. This is in good agreement with previous analytical results from the weakly nonlinear theory (Bagheri 2013). The frequency f is normalized as $St = fD/U_\infty$, where St is the Strouhal number.

Recall that in the laminar parallel shedding region ($47 < Re < 180$), the characteristic Strouhal number St scales with $-1/\sqrt{Re}$ (Fey, König & Eckelmann 1998). Therefore, it is expected that St of both types tend toward higher frequency as Re increases from 70 to 130. Furthermore, it is interesting to note that the corresponding Strouhal numbers for lift and drag when the system is on the limit cycle, St_L and St_D (we observe that each lift and drag coefficient exhibits only one frequency at the limit cycle regime for the range of Re studied in this work.), coincide with the stable frequency of type-I and II, respectively,

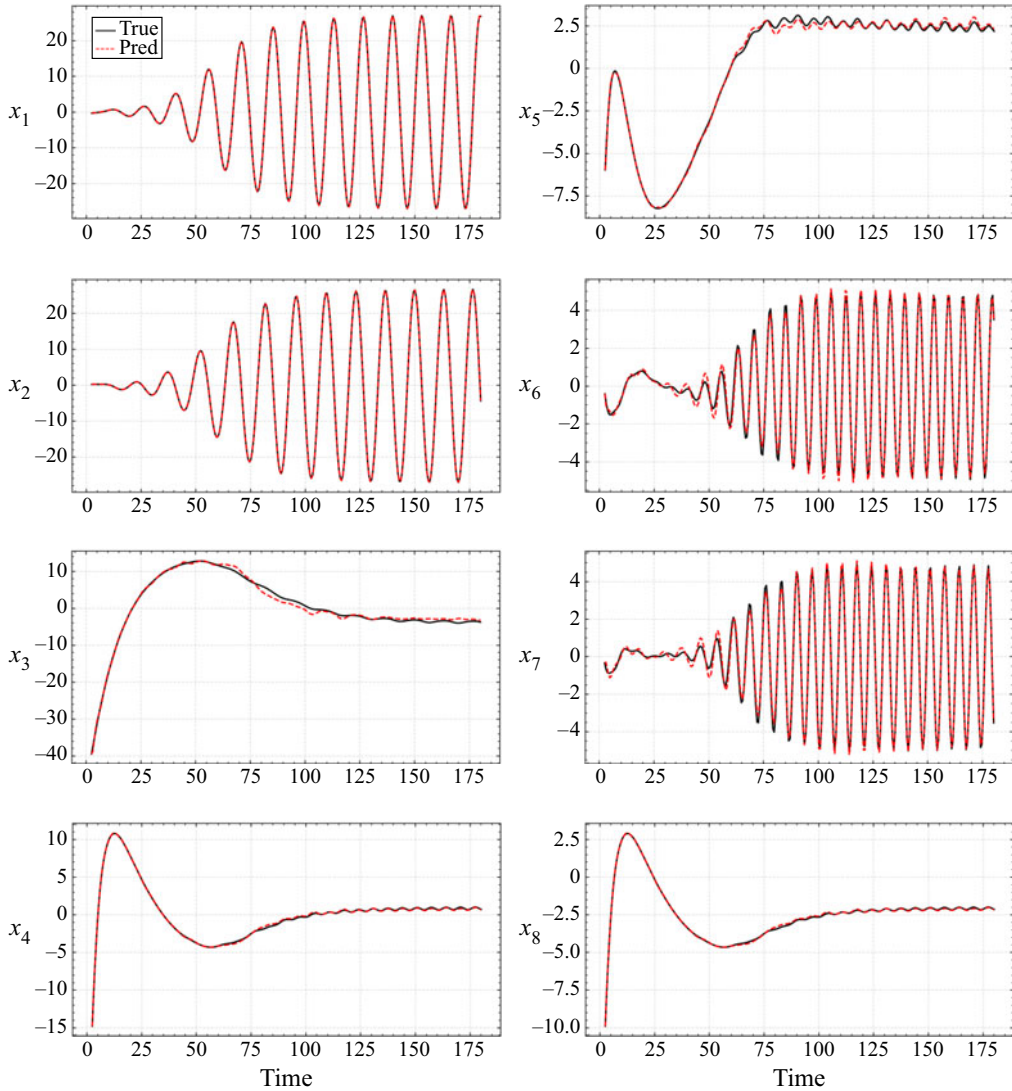


Figure 17. *A posteriori* prediction of testing trajectory for $Re = 70$ in terms of the top eight POD coefficients with spKDMD.

as indicated in figure 21. This is due to the anti-symmetrical/symmetrical structure of the velocity field of type-I/II Koopman mode, respectively, as can be inferred from figure 22. A schematic is also shown in figure 20.

The higher frequency mode is symmetrical (along the free stream direction) in U and anti-symmetrical in V . As a consequence, this only contributes to the oscillation of drag. The lower frequency mode is anti-symmetrical in U and symmetrical in V , and only contributes to the oscillation of lift. Thus, the fluctuation in the lift mostly results from the stable mode in type-I, while that for drag results from the stable mode in type-II with twice the frequency.

Finally, several representative Koopman modes from spKDMD for three Re are shown in figures 22–24. For a better comparison of mode shapes, contributions from the stable modes of type-I and II with a threshold of 0.001 at $t = 0$ is displayed in the top left

Extraction of informative Koopman-invariant subspaces

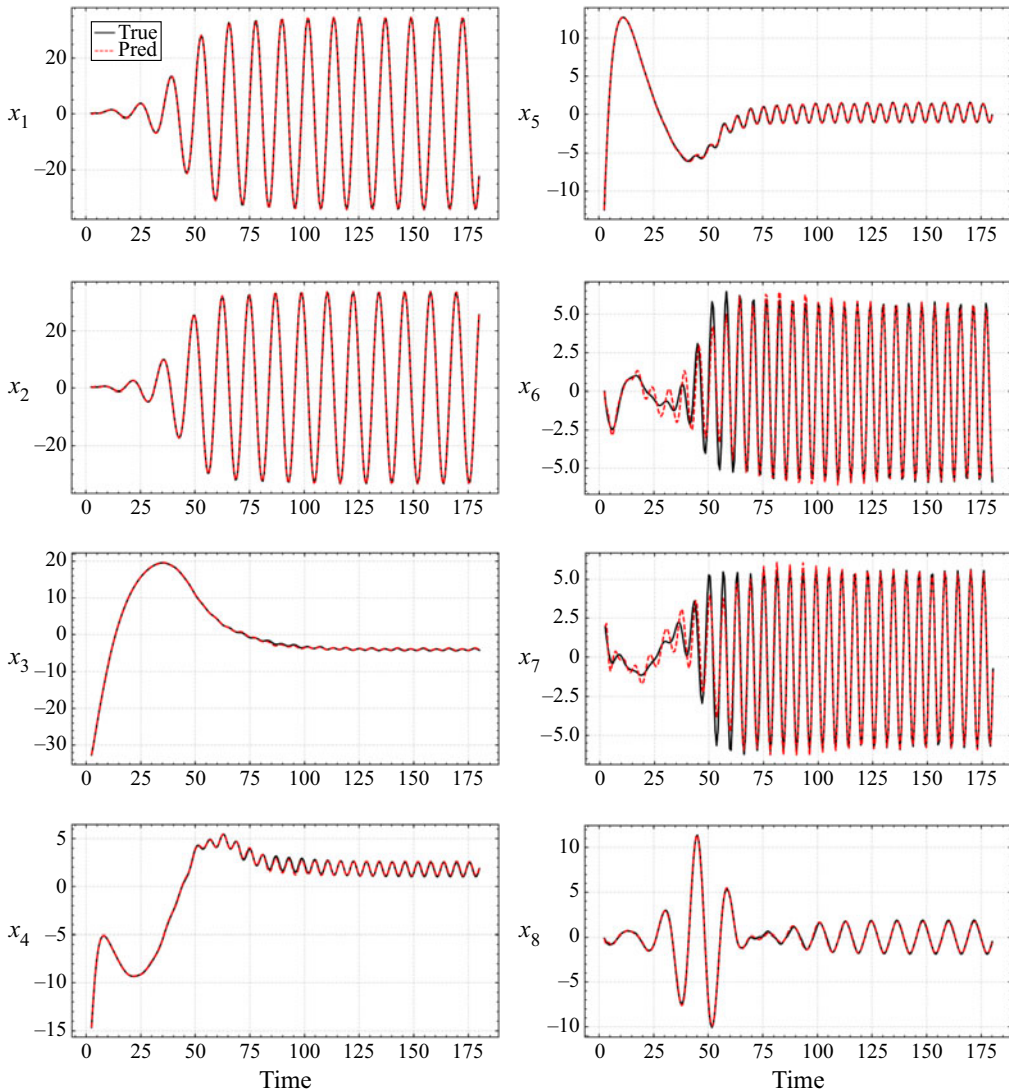


Figure 18. *A posteriori* prediction of testing trajectory for $Re = 100$ in terms of the top eight POD coefficients with spKDMD.

of [figure 25](#). To remove the effect of time, the ‘envelope’ of the mode shape, i.e. time average of the isocontours is shown in the top right of [figure 25](#). From these results, we observe the following interesting phenomena.

- The minimal dimension of the Koopman-invariant subspace that approximately captures the limit cycle attractor for all three Re that fall into laminar vortex shedding regime (White & Corfield 2006) is five, which is consistent with previous multi-scale expansion analysis near the critical Re (Bagheri 2013).
- The lobes of stable Koopman modes in the type-I cluster show an approximately 50 % larger width than those in a type-II cluster.
- Similarity/colinearity among Koopman modes within each cluster is observed. Such a finding is previously reported in the theoretical analysis by Bagheri (2013).

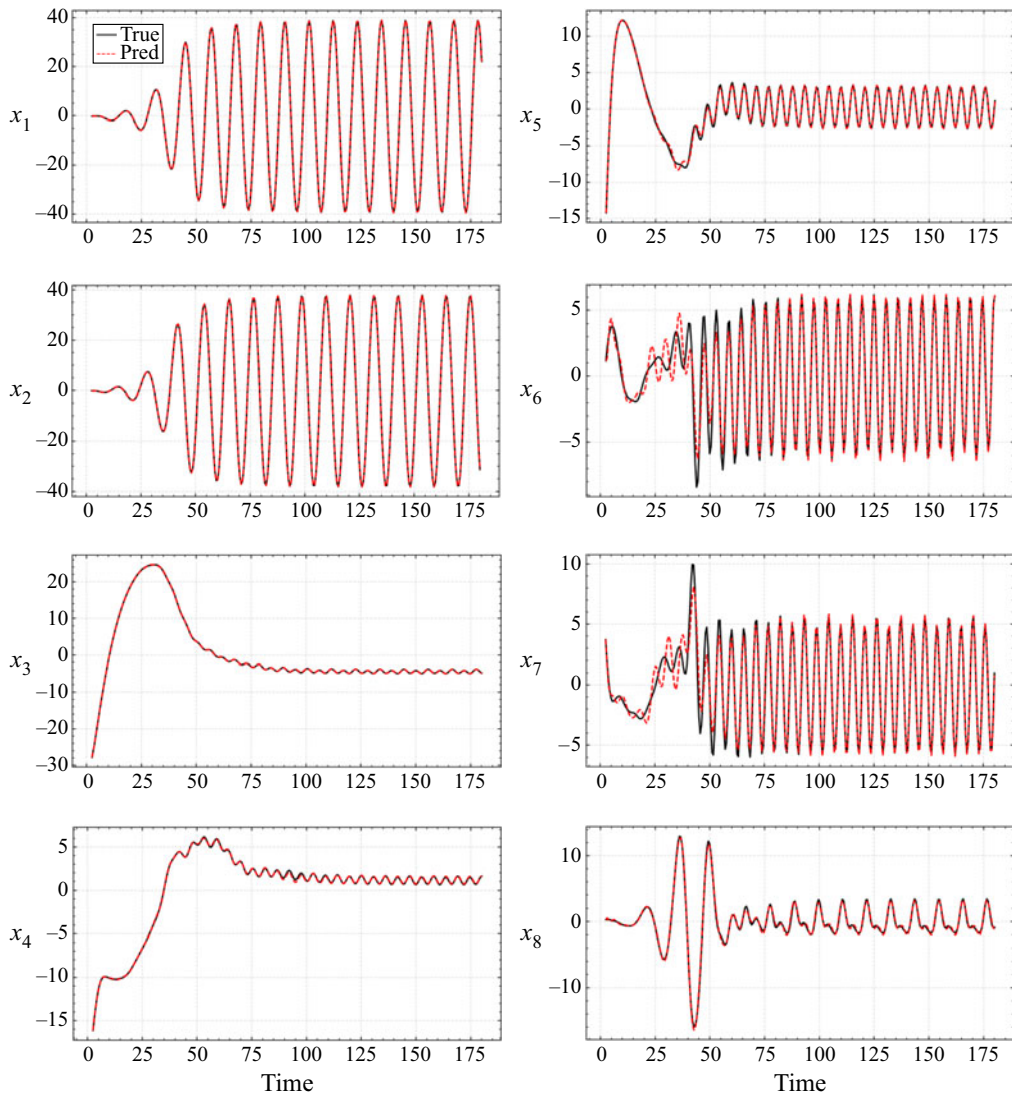


Figure 19. A posteriori prediction of testing trajectory for $Re = 130$ in terms of the top eight POD coefficients with spKDMD.

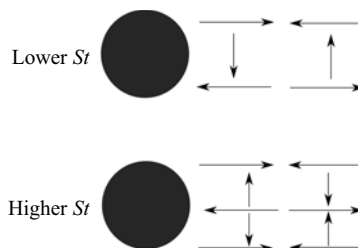


Figure 20. Illustration of the structure of velocity field for the lower (top) and higher frequency (bottom) Koopman modes. The arrow roughly indicates the velocity direction.

Extraction of informative Koopman-invariant subspaces

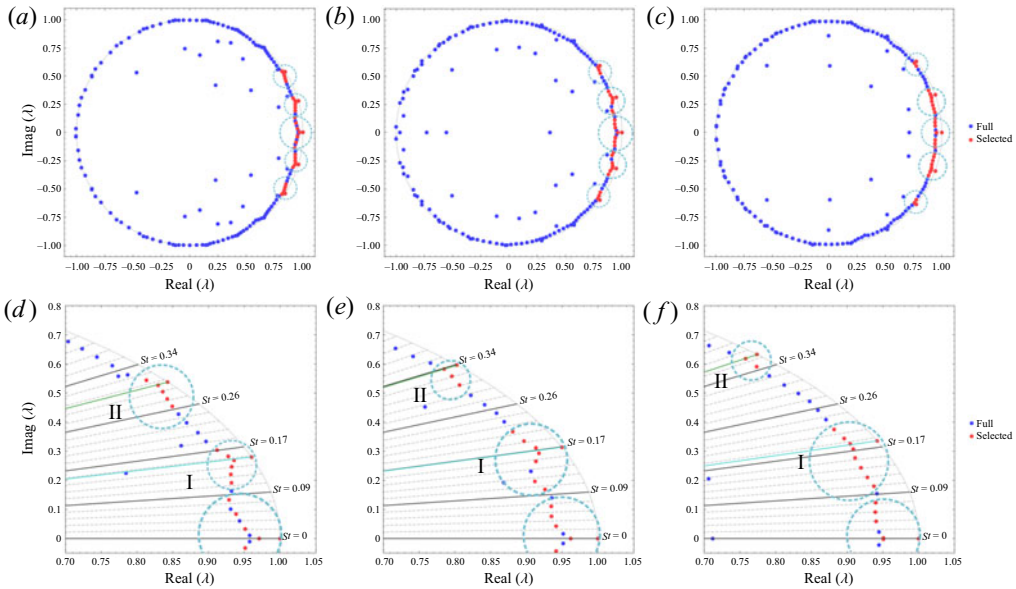


Figure 21. Discrete-time eigenvalue distribution of full KDMD and spKDMD; (a) $Re = 70$, (b) $Re = 100$, (c) $Re = 130$. Blue dot: full KDMD eigenvalues. Red dot: spKDMD eigenvalues. Bottom row: zoomed image. Roman numerals I and II correspond to two types of eigenvalue clusters of distinct frequencies, with each of them enclosed by cyan dashed circles. The green/cyan solid line corresponds to St_D/St_L .

A similarity in spatial structure exists among the Koopman modes belonging to the same family, even though the structures are clearly phase lagged.

- As Re increases from 70 to 130, mode shapes flatten downstream while expanding upstream.
- At $Re = 70$, the shear layer in the stable Koopman modes continues to grow within the computational domain. However, at $Re = 100, 130$, the shear layer stops growing after a certain distance that is negatively correlated with Re .

4.2.2. Net contribution of clustered Koopman modes

By examining figures 22–24, we observe that the colinearity of the spatial structures among each cluster can cause cancellations. A famous example of such non-oscillatory cancellation is the ‘shift mode’ defined by Noack *et al.* (2003), in conjunction with two oscillating modes. As the ‘shift mode’ decays, the stationary component of the flow transitions from the unstable equilibrium to the time-averaged mean. The existence of such non-oscillatory decaying Koopman modes is also confirmed by weakly nonlinear analysis (Bagheri 2013). Interestingly, our algorithm is able to identify not only the non-oscillatory cancellation (from the ‘shift mode’) but also oscillatory cancellations from two clusters with distinct frequencies. Such cancellations elegantly explain why no unstable Koopman eigenvalues appear in this flow given the coexistence of an attracting limit cycle and an unstable equilibrium. These modes could be understood as ‘oscillatory shift modes’, as a generalization of the model proposed by Noack *et al.* (2003).

Since modes within each cluster appear to be colinear to each other, it is intriguing to investigate the net contribution from each cluster. For the above $Re = 70$ case, effects from different clusters at different times in the transient regime are shown in figure 26. We note the following interesting observations throughout the transient period from $t = 80$ to $t = 200$.

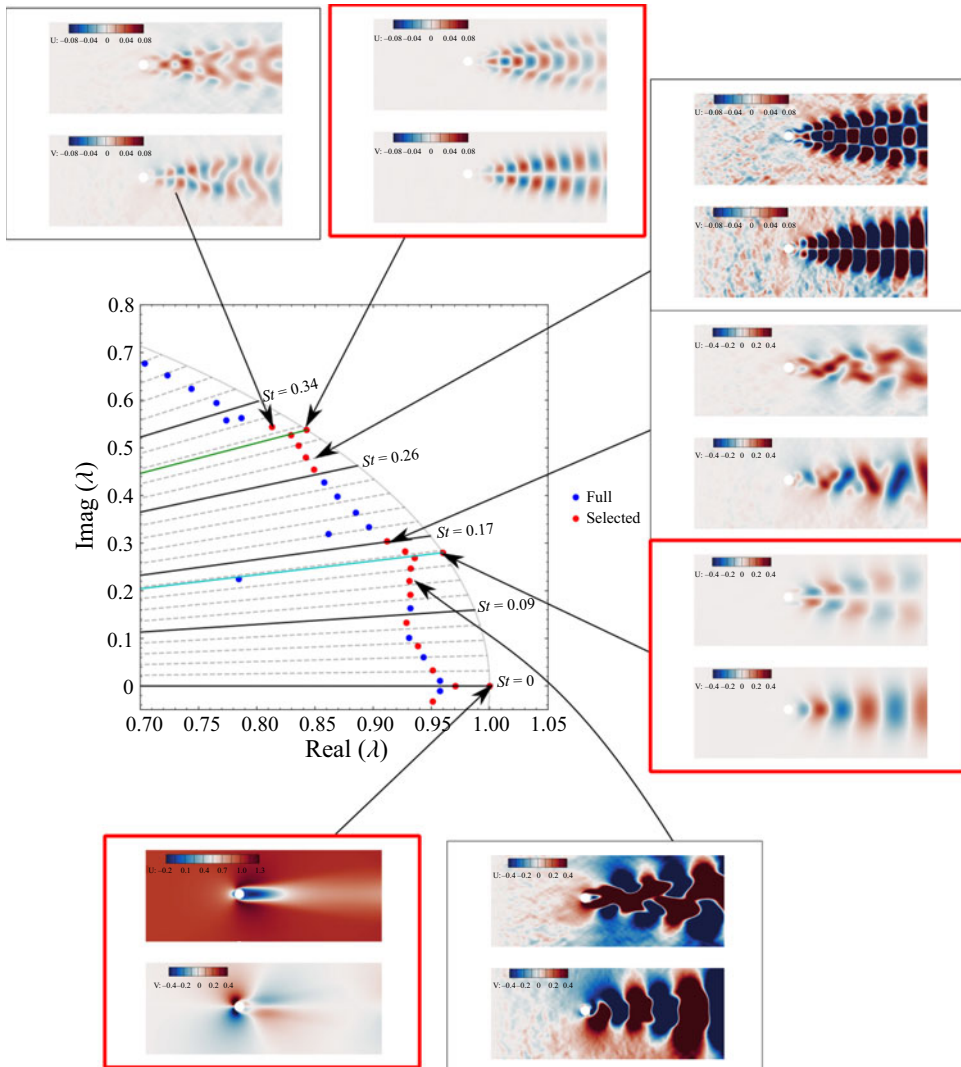


Figure 22. Contours of Koopman modes of $Re = 70$ cylinder wake flow at $t = 0$. Red squares indicate stable modes.

- The net contribution from ‘cluster 0’ does not exhibit strong oscillations. For the contribution from ‘cluster 0’, the U component shows a decrease in the length of the reverse flow region behind the cylinder with an increase in the low speed wake layer thickness while the V component remains unchanged. This is similar to the effect of ‘shift mode’ which also characterizes the decay of recirculation behind the cylinder.
- Initially at $t = 80$, the net contribution from ‘cluster I’ is rather weak primarily due to the ‘cancellation’ from the lagged phase. Furthermore, the development of vortex shedding downstream from the top and bottom surfaces of the cylinder is nearly parallel. This corresponds to the initial wiggling of the low speed wake flow downstream. As time increases, the pattern of corresponding vortices develops away from the centreline.

Extraction of informative Koopman-invariant subspaces

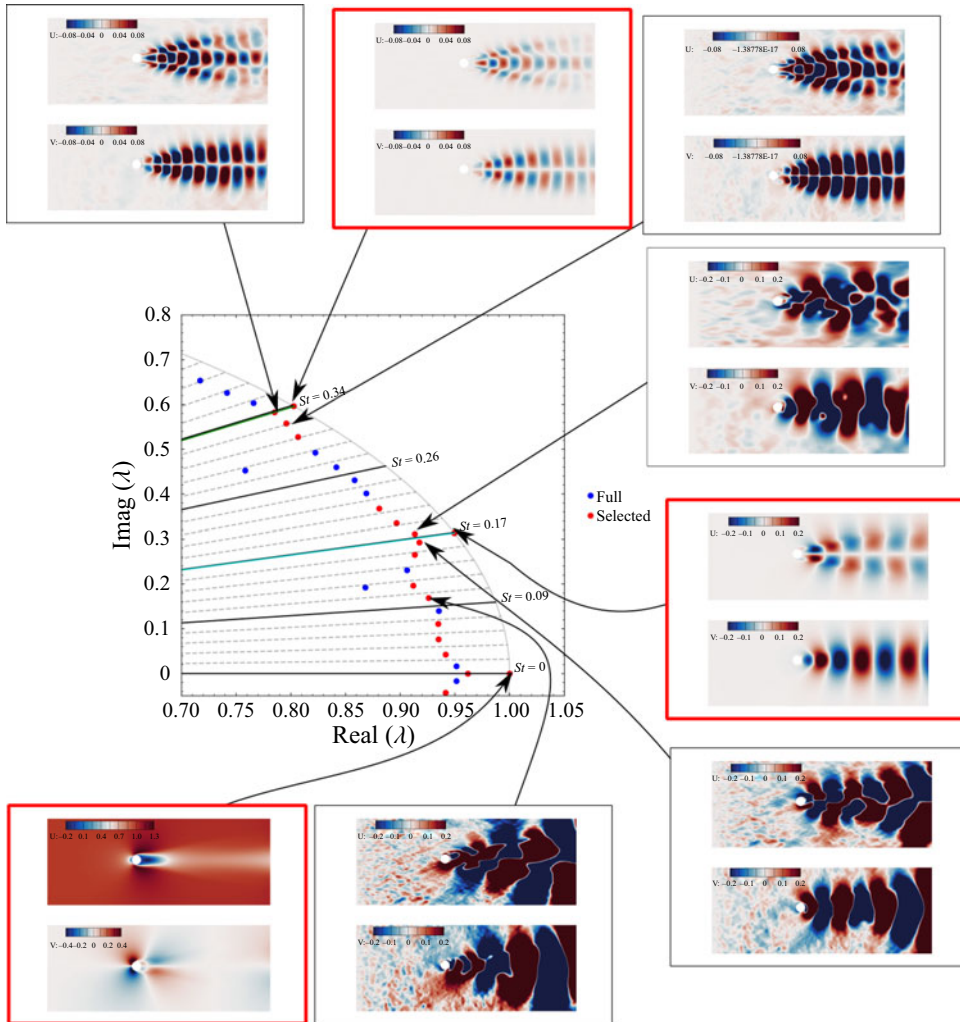


Figure 23. Contours of Koopman modes of $Re = 100$ cylinder wake flow at $t = 0$. Red squares indicate stable modes.

- Although the net contribution from ‘cluster 0+I’ captures most of the flow features from ‘full modes’ throughout the transient regime, with increasing time, the net contribution from ‘cluster II’ becomes more important and contributes to the strength of vortex shedding downstream.

4.2.3. Comparison with DMD and spDMD at $Re = 70$

To confirm the advantage of spKDMD over DMD (Schmid 2010) and spDMD (Jovanović *et al.* 2014), we compare the following three models on the unsteady cylinder wake flow at $Re = 70$:

- spKDMD on the top 20 POD coefficients with 34 modes selected;
- DMD on the top 20 POD coefficients;

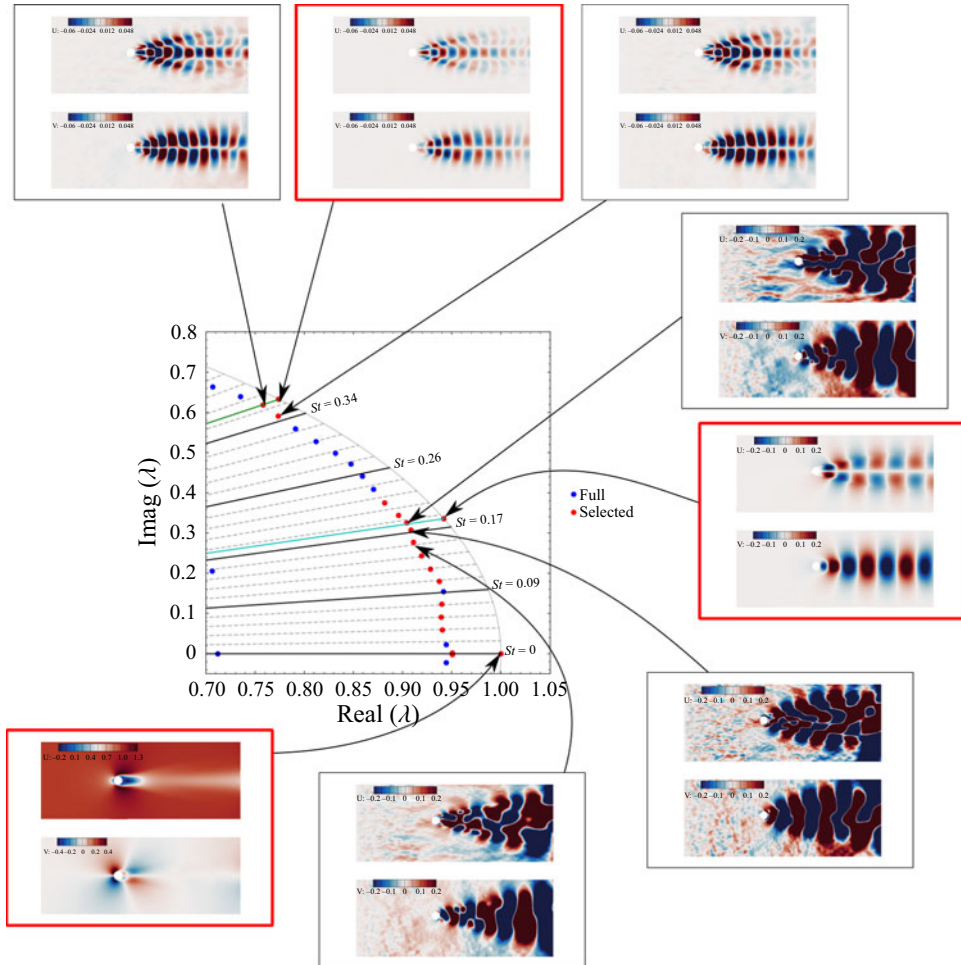


Figure 24. Contours of Koopman modes of $Re = 130$ cylinder wake flow at $t = 0$. Red squares indicate stable modes.

- (iii) spDMD (we used the original Matlab code from <http://people.ece.umn.edu/users/mihailo/>) on the top 200 POD coefficients with α chosen carefully such that only 34 modes are selected.

Note that DMD with the top 200 POD coefficients, i.e. $r = 200$ in SVD-DMD (Schmid 2010), contains 10 times stable/decaying harmonics as DMD on the top 20 modes. Hence, it is not surprising to expect that the corresponding prediction of the evolution of the top 20 POD coefficients to be very good (not shown for clarity). However, to make a fair comparison against spKDMD, we consider spDMD (Jovanović *et al.* 2014) on the top 200 POD coefficients with 34 modes selected. (Although there are 200 POD coefficients used for spDMD and 20 for KDMD, it is not an unfair comparison given that the same number of spatial modes are selected. Furthermore, these are reduced-order models of the same full-order dynamical system.) As shown in figure 27, given the same number of eigenmodes, spKDMD performs remarkably better than spDMD, especially in the transient regime. This is likely due to the inability of DMD in capturing nonlinear algebraic growth in the transient regime (Bagheri 2013). Sparsity-promoting DMD overestimates the

Extraction of informative Koopman-invariant subspaces

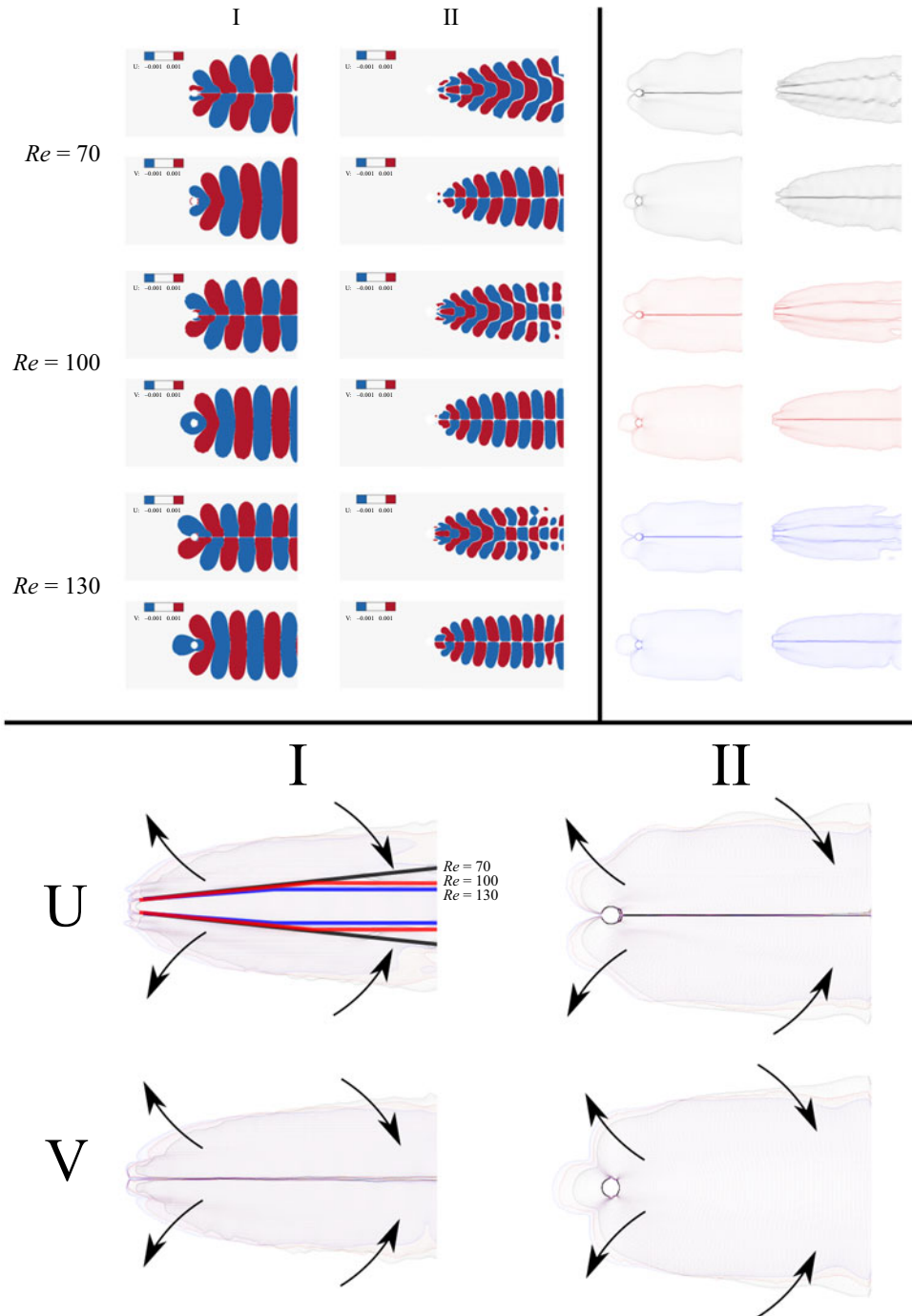


Figure 25. Top left: contribution of stable Koopman modes corresponding to type-I and type-II clusters for $Re = 70, 100, 130$ at $t = 0$ visualized with threshold 0.001. Top right: time-averaged isocontour of top left plot. Bottom: tendency of 'envelope' of type-I and II modes as Re increases. Separation lines in U component of type-I are drawn for $Re = 70$ (black), $Re = 100$ (red) and $Re = 130$ (blue).

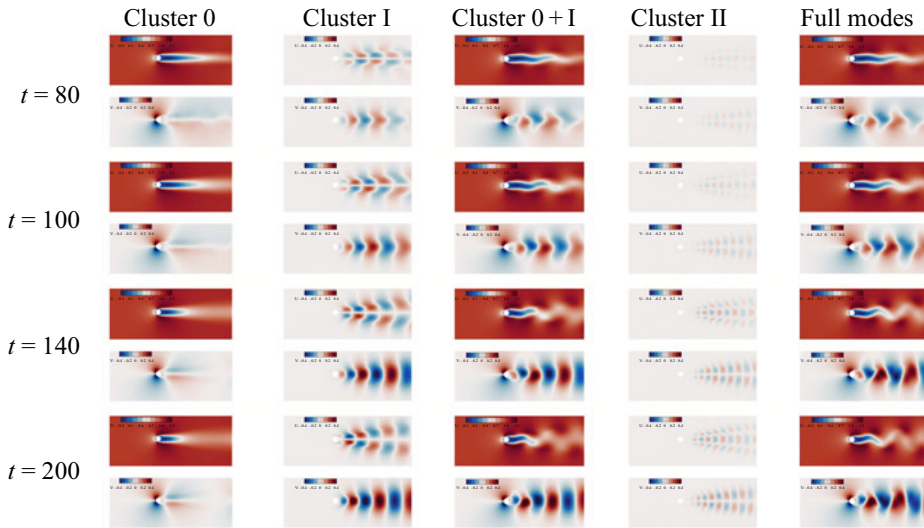


Figure 26. Contribution of Koopman modes at cluster level in the transient regime of $Re = 70$ case. Here ‘cluster 0’ denotes the cluster near the real axis in figure 21; ‘cluster I’ and ‘cluster II’ take the effect of a mirror cluster in the fourth quadrant into account; ‘full modes’ denotes the aggregated contribution of Koopman modes.

growth in x_1 and x_2 while ignoring a turnaround near the onset of the transient regime in x_5 and x_8 . As expected, DMD with 20 POD coefficients performs the worst especially for the modes where transient effects are dominant. Given the results in figure 27, among all of the top eight POD coefficients, x_6 and x_7 appear to be most challenging to model: DMD and spDMD cannot match the limit cycle while spKDMD performs very well. Notice that the frequency in x_6 and x_7 corresponds to St_D . Hence, there will be a difference in predicting the fluctuation of C_D between spDMD and spKDMD.

Finally, comparison of the identified Koopman eigenvalues between DMD, spDMD and spKDMD is shown in figure 28. On one hand, both spDMD and spKDMD exactly capture the stable eigenmodes that correspond to St_D and St_L . This is expected since DMD with 200 POD coefficients represents the dynamics very well, and deviation from limit cycle behaviour would be penalized in spDMD. On the other hand, several erroneous stable DMD modes are obtained by DMD. This explains the deviation of *a posteriori* prediction from the ground truth limit cycle in figure 27. For those decaying modes, similarity/colinearity is observed between two clusters of eigenvalue from spDMD and spKDMD. However, spKDMD contains more high-frequency modes than spDMD. Finally, it is interesting to note that, although the correct stable eigenvalues are captured accurately by both spDMD and spKDMD, the former does not capture accurate amplitudes for stable eigenvalues of type-II as seen in figure 27.

As a side note, when the temporal mean was used instead of maximal error in the definition of Q in (3.2), spKDMD with the above setting was found to not find a stable eigenvalue corresponding to ST_D .

4.3. Three-dimensional transient turbulent ship airwake

Understanding unsteady ship-airwake flows is critical to design safe shipboard operations, such as takeoff and landing of fixed or rotary wing aircraft (Forrest & Owen 2010), especially when the wind direction becomes stochastic. Here we obtain snapshots from

Extraction of informative Koopman-invariant subspaces

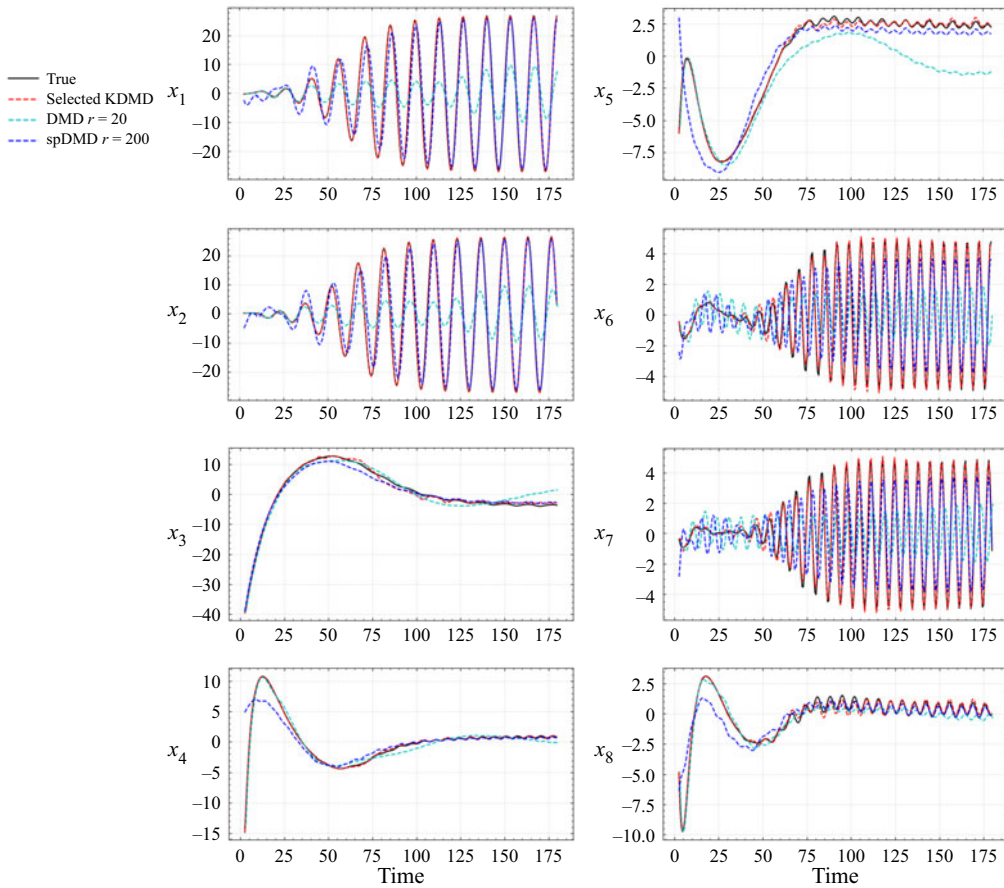


Figure 27. Comparison of *a posteriori* prediction on the top eight POD coefficients of the testing trajectory between spKDMD, DMD (Schmid 2010) and spDMD (Jovanović *et al.* 2014) for the two-dimensional cylinder flow at $Re = 70$. Here x_i denotes the i th POD coefficient.

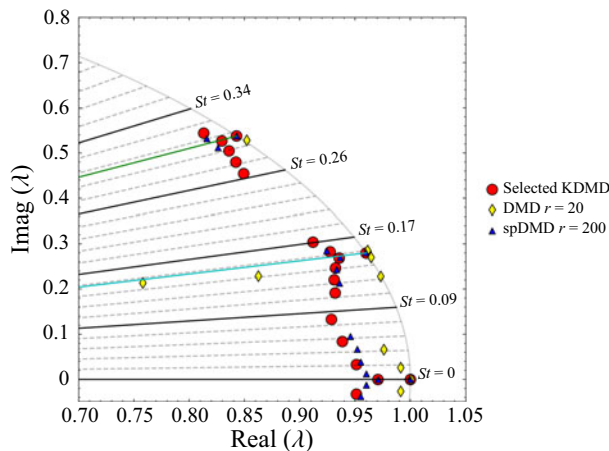


Figure 28. Comparison of identified eigenvalues between spKDMD, DMD (Schmid 2010) and spDMD (Jovanović *et al.* 2014) for the two-dimensional cylinder flow at $Re = 70$.

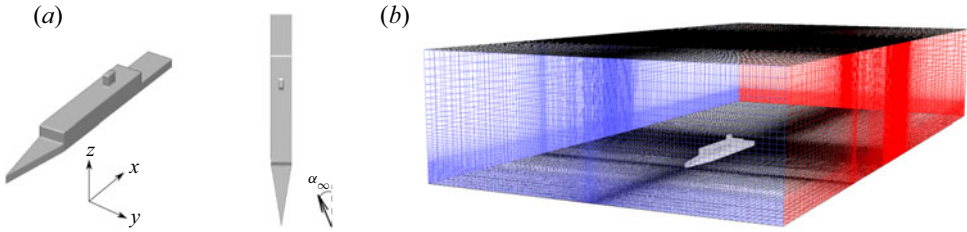


Figure 29. (a) Geometry of the ship (SFS2). (b) Generated computational mesh.

an unsteady Reynolds averaged Navier–Stokes (URANS) simulation of a ship airwake using FLUENT (Ansys 2016) with the shear layer corrected $k-\omega$ two-equation turbulence model. Unsteadiness arises from both bluff-body separation, and an abrupt change in wind direction. We consider a conceptual ship geometry called simple frigate shape 2 (SFS2). For the details of the geometry, readers are referred to Yuan, Wall & Lee (2018). Configuration of the simulation set-up is shown in figure 29, where α_∞ denotes the angle of side wind.

To prepare a proper initial condition, a URANS simulation for $U_\infty = 15 \text{ m s}^{-1}$ with $\alpha_\infty = 0^\circ$, i.e. no side wind, is conducted to reach a physical initial condition. Following this, the last snapshot is used as the initial condition for a new run with an impulsive change in the wind direction from $\alpha_\infty = 0^\circ$ to $\alpha_\infty = \alpha_0 = 5^\circ$. The boundary conditions for outlet/input is pressure outlet/velocity inlet while the top and bottom are set as symmetry for simplicity. No-slip conditions are used at the surface of the ship. Further details on the simulation set-up are provided in Sharma *et al.* (2019).

The sampling time interval is $\Delta t = 0.1 \text{ s}$ with 500 consecutive samples of the three velocity components. This corresponds to several flow through times over the ship length. The domain of interest is a Cartesian region of mesh size $24 \times 40 \times 176$ starting on the rear landing deck. For dimension reduction, the trajectory of the top 40 POD coefficients (temporal mean subtracted) are collected, yielding $>99\%$ kinetic energy preservation. Discrete-time KDMD with an isotropic Gaussian kernel is employed to perform nonlinear Koopman analysis where $\sigma = 200$, $r = 135$ is chosen. Details of hyperparameter selection are provided in § C.3.

4.3.1. Results of discrete-time KDMD with mode selection

First, the error analysis of eigenfunctions is shown in figure 30, where we choose $\hat{L} \approx 60$ for good reconstruction. However, the level of deviation from linear evolution is around 10%. This error will be reflected later as deviation in *a posteriori* prediction on the testing trajectory. (This implies difficulties in finding an accurate yet informative Koopman operator with isotropic Gaussian kernels. However, choosing an optimal kernel type is beyond the scope of this work.)

Second, the result of mode selection is summarized in table 3. Note that nearly 2/3 of the modes are removed. Furthermore, model performance in terms of *a posteriori* prediction on the testing trajectory is evaluated. Comparison between KDMD and the ground truth on contours of velocity components on a special z -plane (1.2 m above the landing deck) is shown in figure 31. Effects of an impulse change in wind direction in the following are observed from $t = 1.5 \text{ s}$ to $t = 30 \text{ s}$ and well captured by spKDMD:

- growth of a large shear layer in U on the rear (left) side of the superstructure on the ship;

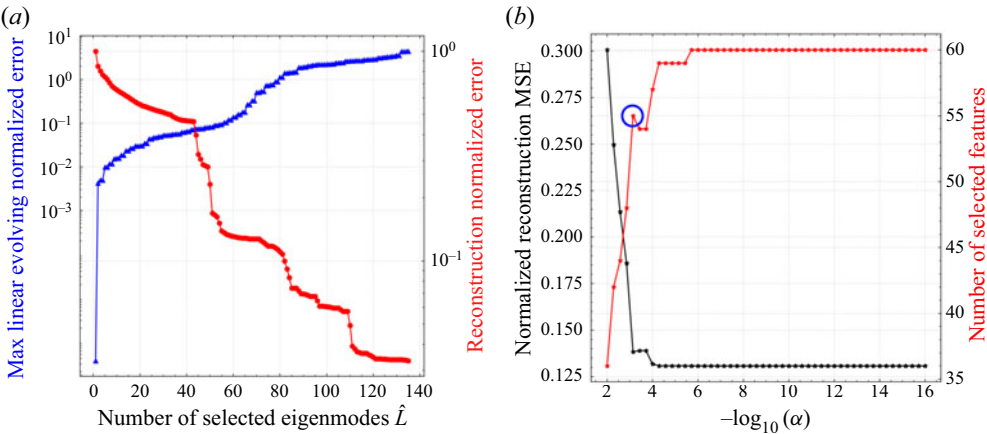


Figure 30. (a) Trend of linearly evolving error Q and reconstruction error R from discrete-time KDMD for the ship airwake. (b) Trend of linearly evolving error Q and reconstruction error R from discrete-time KDMD.

α_{select}	7.19×10^{-4}
Number of selected modes	55
Number of total modes	167
Normalized reconstruction error R	0.133

Table 3. Summary of mode selection for discrete-time KDMD on ship airwake.

- a strong side wind sweep in V above the landing deck propagating downstream;
- development of a vortex on the upwind (right) downstream side of the ship.

Furthermore, three velocity components of the Koopman mode decomposition on the previously mentioned z -plane is shown in figure 32 together with the isocontour of vorticity coloured by the velocity magnitude for the two stable harmonics. Note that the frequency is normalized using U_∞ as the reference velocity and funnel width of the ship $L = 3.048$ m as the characteristic length scale (Forrest & Owen 2010). As expected, the spKDMD yields a large number of decaying modes with only three non-trivial stable harmonics, since significant transient effects are present in the data. From the Koopman mode decomposition in figure 32, we observe the following:

- modes with eigenvalues close to each other exhibit a similar spatial structure;
- modes associated with higher frequency are dominated by smaller scales;
- the stable harmonic mode near $St = 0.09$ associated with a strong cone-shape vortex originating from the upwind (right) rear edge of the superstructure on the ship;
- the stable harmonic mode near $St = 0.068$ corresponds to vortex shedding induced by the funnel;
- the slowly decaying mode near $St = 0.022$ shows unsteady circulation directly behind the superstructure;
- the steady mode ($St = 0$) is consistent with the large circulation behind the superstructure, the roll-up wind on the side of the landing deck, and vertical suction towards the floor on the landing deck.

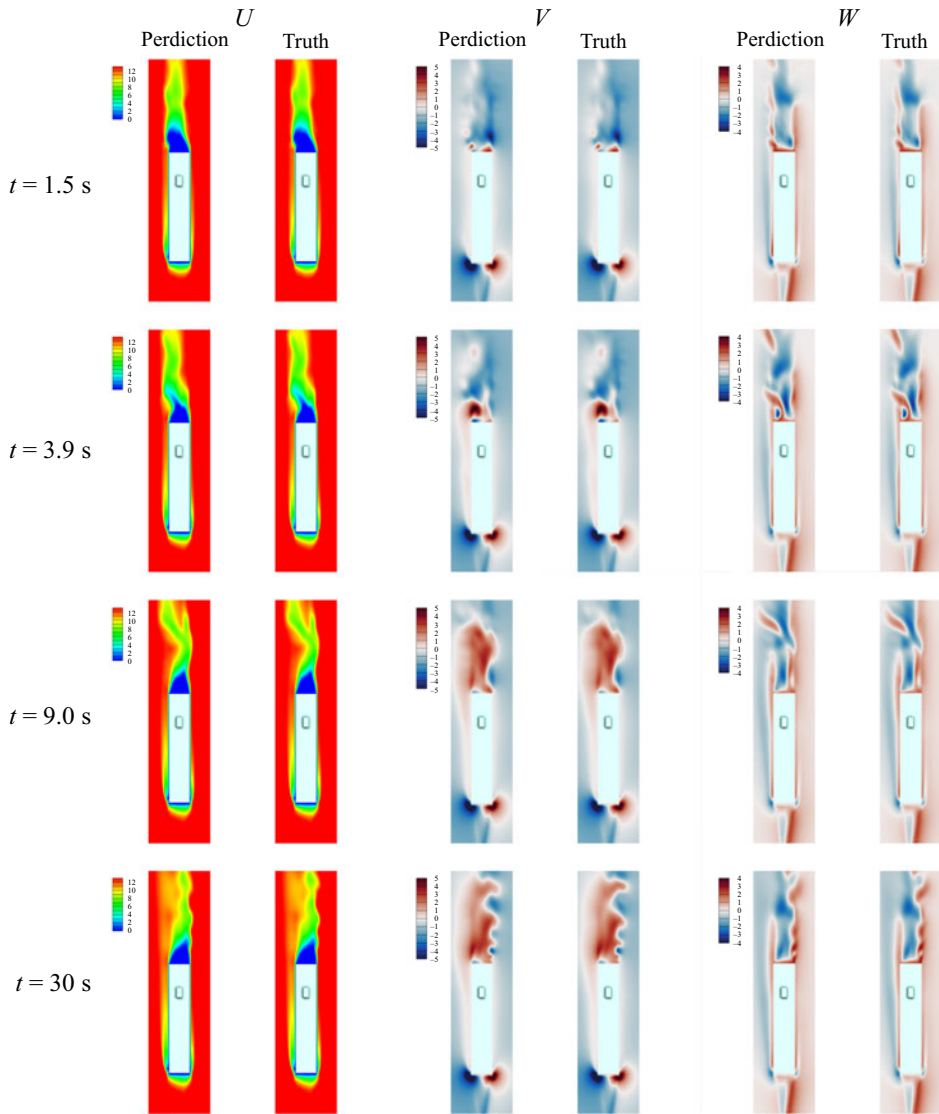


Figure 31. Contour of velocity components near the ship on the z -plane slice at $t = 1.5s, 3.9s, 9.0s, 30s$. For each subfigure, (a) prediction from KDMD; (b) ground truth.

4.3.2. Comparison with spDMD

We again repeat the comparison between our spKDMD and spDMD (Jovanović *et al.* 2014). Note that DMD on the first 40 POD modes performs poorly similar to § 4.2.3 and, therefore, is not shown here. To make a fair comparison against the spKDMD from the previous subsection, however, we collect the first 200 POD coefficients for spDMD to ensure that a sufficient number of modes are used to fit the trajectory well. We then carefully choose the penalty coefficient in spDMD to ensure that the same number of modes are retained as in spKDMD. As shown in figure 33, within the time horizon $t < 50$, *a posteriori* evaluation shows that spKDMD offers much improved predictions compared with spDMD (Jovanović *et al.* 2014) on the testing trajectory.

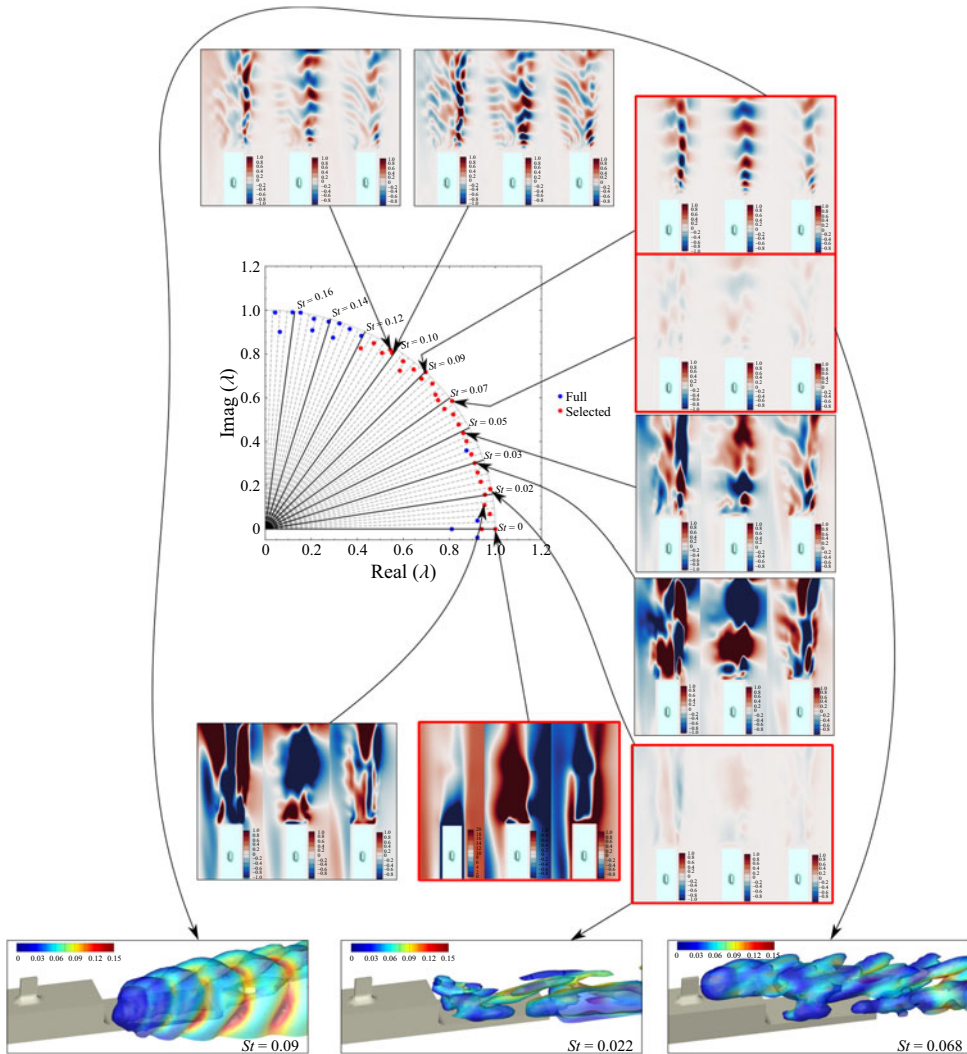


Figure 32. Contours of Koopman modes of ship airwake on the z -plane at $t = 0$. For each subfigure, left: U , middle: V , right: W . Red squares indicate stable modes. Bottom: isocontour of vorticity coloured by velocity magnitude zoomed in near the landing deck.

Moreover, as further illustrated in figure 34(a), eigenvalues identified from spKDMD only contain two stable modes while nearly all eigenvalues from spDMD are located near the unit circle, among which there are 30 out of 56 slightly unstable modes. These unstable modes inevitably lead to the identified system being numerically unstable when predicting beyond the current training time horizon, whereas spKDMD predicts a ‘physically consistent’ limit cycle behaviour. As indicated in figure 34(b), such instability is related to the inability of the (linear) features to approximate the Koopman-invariant subspace, where only eight modes are within 10% of maximal deviation from linear evolution, compared with 60 modes in KDMD. We note that similar observations of the drastically different eigenvalue distribution were reported in the original KDMD paper (Williams *et al.* 2014).

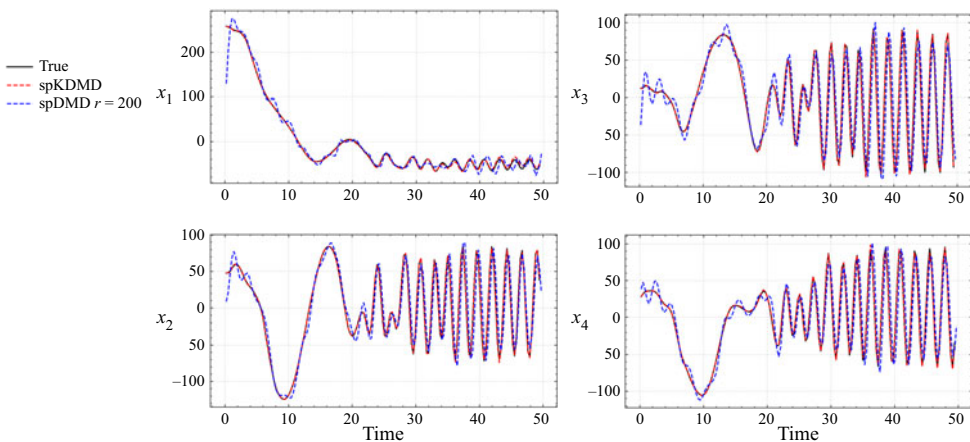


Figure 33. Comparison of *a posteriori* prediction of the four most significant POD coefficients of the testing trajectory between spKDMD and spDMD (Jovanović *et al.* 2014) for the three-dimensional (3-D) ship-airwake flow. Here x_i denotes the i th POD coefficient.

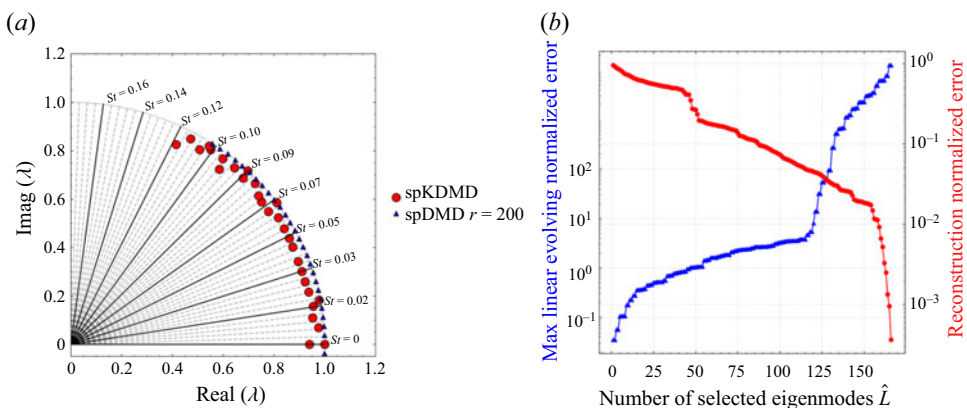


Figure 34. (a) Comparison of identified eigenvalues between spKDMD and spDMD (Jovanović *et al.* 2014) for the 3-D ship-airwake flow. (b) Trend of linear evolving error Q and reconstruction error R from DMD for the 3-D ship-airwake flow.

5. Conclusions

Two classical nonlinear approaches for the approximation of Koopman operator: EDMD and KDMD are revisited. From an algorithmic perspective, the main contributions of this work are (a) sparsity-promoting techniques based on *a posteriori* error analysis, and (b) multi-task learning techniques for mode selection as an extension of spDMD into the nonlinear variants. Furthermore, analytical relationships between spDMD, Kou's criterion and the proposed method are derived from the viewpoint of optimization. The algorithm (code available at <https://github.com/pswpswpsw/SKDMD>) is first evaluated in detail on a simple two-state dynamical system, for which the Koopman decomposition is known analytically.

If one is only interested in the post-transient dynamics of the system on an attractor, linear observables with time delays are sufficient to extract an informative

Koopman-invariant subspace. Thus, the present techniques are evaluated on two fluid flows which involve strong transients: two-dimensional flow over a cylinder at different Reynolds numbers, and a three-dimensional (3-D) ship airwake. We demonstrate the effectiveness of discovering accurate and informative Koopman-invariant subspaces from data and constructing accurate reduced-order models from the viewpoint of Koopman theory. Furthermore, with the proposed algorithm, the parametric dependency of Koopman mode shapes on the Reynolds number is investigated for the cylinder flows. In this case, as Re increases from 70 to 130, the shape of stable modes becomes flatter downstream and larger upstream. Moreover, the similarity of mode shapes between Koopman modes with similar eigenvalues is observed in both fluid flows. Specifically, five clusters of eigenvalues are observed in the case of a two-dimensional cylinder wake flow which is confirmed with weakly nonlinear theoretical analysis from Bagheri (2013). Type-I, II clusters are found to correspond to fluctuations in lift and drag, respectively. We identify non-oscillatory as well as oscillatory cancellations from the above two clusters with distinct frequencies. These modes could be understood as ‘oscillatory shift modes’, as a generalization of the model proposed by Noack *et al.* (2003). For the 3-D ship-airwake case, two stable modes, and one slowly decaying mode with distinct frequencies and mode shapes resulting from vortex shedding are extracted, and accurate predictive performance is observed in the transient regime.

Acknowledgements. We thank the three anonymous reviewers whose comments and suggestions helped improve and clarify this manuscript. This research was supported by the DARPA Physics of AI Program under the grant ‘Physics Inspired Learning and Learning the Order and Structure of Physics’, and by the NSF CMMI program under the grant ‘A Diagnostic Modeling Methodology for Dual Retrospective Cost Adaptive Control of Complex Systems’. Computing resources were provided by the NSF via the grant ‘MRI: Acquisition of ConFlux, A Novel Platform for Data-Driven Computational Physics’.

Declaration of interests. The authors report no conflict of interest.

Author ORCIDs.

 Shaowu Pan <https://orcid.org/0000-0002-2462-362X>.

Appendix A. Proof for the identity for (2.22)

Consider the economy SVD of $A' = USV^H$. Since the column space of A' spans the column space of Z_r , there exists an invertible matrix P such that $Z_r P = U$. Hence,

$$A' Z_r = V S U^H Z_r = V S P^H Z_r^H Z_r = V S P^H, \quad (A1)$$

and $((A' Z_r)^H A' Z_r)^+ = (P^{-1})^H S^{-2} P^{-1}$. Thus,

$$A' A'^H Z_r (Z_r^H A' A'^H Z_r)^+ = (USV^H)(V S P^H)(P^{-1})^H S^{-2} P^{-1} = U P^{-1} = Z_r. \quad (A2)$$

Appendix B. Computational complexity

Computational cost of the proposed framework can be divided in three parts:

- (i) distributed SVD;
- (ii) KDMD/EDMD algorithms;
- (iii) multi-task feature learning with the parameter sweep (solving (3.9) with different values of α).

Step	Computational complexity
SVD (QR iteration with vectors)	$O(Mn^2)$ (Dongarra <i>et al.</i> 2018)
KDMD	$O(M^3)$
Error evaluation & pruning	$O(M^2n)$
Multi-task ElasticNet with a fixed penalty coef. α	$O(N_{iter}\hat{L}^2r)$
Multi-task ElasticNet with N_α coefs. (worst case)	$O(N_\alpha N_{iter}\hat{L}^2r)$

Table 4. Computational complexity of each step in the proposed sparsity-promoting framework.

The computational complexity for each step in the algorithm is summarized in table 4, where n is the dimension of the system state, M is the number of snapshots in the training (for conciseness, we assume that the number of training snapshots equals the number of validation snapshots.), r is the rank of the reduced system after SVD, \hat{L} is the user-defined cut-off for ‘accurate’ features and N_{iter} is the maximal number of iterations user defined to achieve a residual threshold, e.g. 10^{-12} .

As shown in table 4, the theoretical computational complexity for multi-task ElasticNet with an α sweep is $O(N_\alpha N_{iter}\hat{L}^2r)$. Note that this is a worst case simply because – except for the first α – we reuse the previous optimal solution as the initial condition for the new objective function to start the iterative optimization process. Also, thanks to SVD-based dimension reduction, the cost scales linearly with the reduced system rank r . Moreover, the user-defined linearly evolving error truncation \hat{L} helps reduce that complexity as well instead of scaling with the number of snapshots M . Lastly, there is a cubic theoretical complexity associated with the number of snapshots when applying KDMD. The number of snapshots in a typical high-fidelity simulation is $O(10^3)$. That is to say, $r < 10^3$ and $\hat{L} < 10^3$. We note that computational efficiency can be further improved, but this will be left for future work.

Appendix C. Hyperparameter selection

C.1. Two-dimensional fixed point attractor

We perform grid search in parallel for the selection of σ and the truncated rank r over the range $\sigma \in [10^{-1}, 10]$ with 80 points uniformly distributed in the log sense and $r = 36, 50, 70$ to find a proper combination of r and σ . As shown in figure 35, the higher rank r leads to a larger number of linearly evolving eigenfunctions. Thus, it is more crucial to choose a proper scale σ than r from figure 35. However, considering the simplicity of this problem, $\sigma = 2$ and $r = 36$ would suffice.

C.2. Cylinder flow case

As the flow is not turbulent, we choose hyperparameters for the $Re = 70$ case and fix them for all the other cases. We sweep σ from $[1, 10^5]$ with 30 points uniformly distributed in the log sense and $r = 120, 140, 160, 180, 200$, as shown in figure 36. From the plot we choose $r = 180$ and $\sigma = 3$ for an approximate choice of the hyperparameter. Again, we observe that the number of accurate eigenfunctions first increases then decreases with σ increasing and the saturation of rank truncation at higher σ which is related to the variation in the characteristic scale of the features with respect to σ .

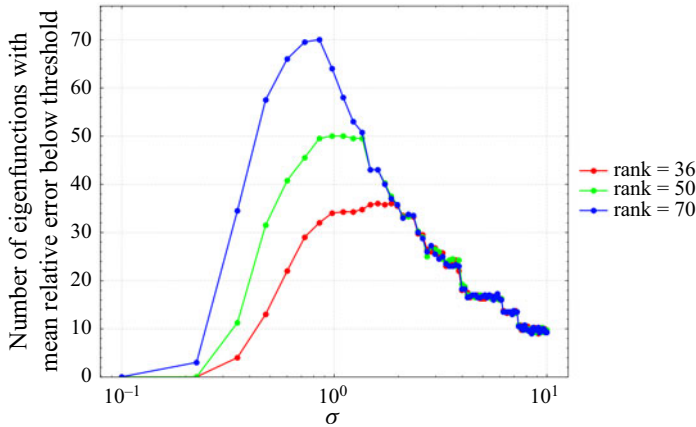


Figure 35. Hyperparameter search for isotropic Gaussian KDMD on the 2-D fixed point attractor.

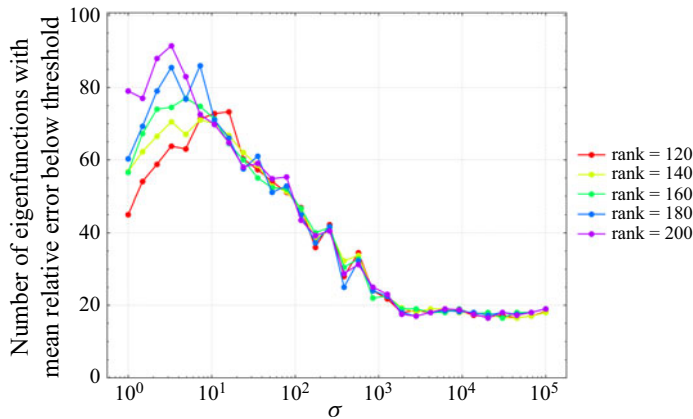


Figure 36. Hyperparameter search for isotropic Gaussian KDMD on transient cylinder wake flows.

C.3. Turbulent ship-airwake case

Grid search in parallel for the selection of σ and r is performed over the range $\sigma \in [1, 10^5]$ with 50 points uniformly distributed in the log sense, $r = 40, 80, 120, 130$. As shown in figure 37, a good choice of σ can be 200 for the case of $\alpha_\infty = 5^\circ$. Note that since the hyperparameter selection is performed with a five-fold cross-validation on the training data, we only have up to $166 \times 0.8 \approx 132$ data points, i.e. maximal possible rank is 132. While in the actual training, we have maximal rank up to 166. Note that as long as the system is well conditioned, the higher the rank, the richer the subspace. Here we take $\sigma = 200$ and $r = 135$.

Appendix D. Parallel data processing tool

The application of data-driven methodologies to increasingly large data sets has exposed the need for scalable tools. Method development is easily achieved in scripting environments, but for application to large multidimensional problems, memory and data I/O becomes a limiting factor. To illustrate this, consider a state-of-the art 3-D computation. Given the fact that there are multiple variables associated with each grid cell,

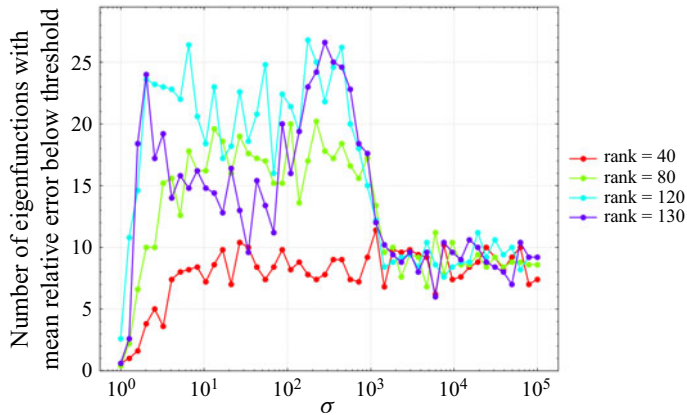


Figure 37. Hyperparameter search for isotropic Gaussian KDMD on a transient ship airwake.

and that reasonable meshes can be expected to contain millions to hundreds of millions of cells, work-space memory requirements of terabytes may be required. Utilizing available distributed tools and I/O strategies a generalized toolset, PDP, has been developed for application of complex methods to arbitrarily large data sets. Originally developed for use in reduced-order modelling, the generality of the toolset allows for a range of complex methodologies to be applied. This was leveraged for use in this work for the application of methods, as the problem size is intractable on desktop machines. The toolset leverages commonly available packages in the form of ScaLAPACK, PBLAS and PBLACS (Blackford *et al.* 1997) routines. ScaLAPACK is widely available on most computing resources where the original simulations are run.

REFERENCES

- ANSYS 2016 Ansys fluent user's guide, release 17.0. Ansys.
- ARBABI, H., KORDA, M. & MEZIC, I. 2018 A data-driven Koopman model predictive control framework for nonlinear flows. [arXiv:1804.05291](https://arxiv.org/abs/1804.05291).
- ARBABI, H. & MEZIC, I. 2017a Ergodic theory, dynamic mode decomposition, and computation of spectral properties of the Koopman operator. *SIAM J. Appl. Dyn. Syst.* **16** (4), 2096–2126.
- ARBABI, H. & MEZIC, I. 2017b Study of dynamics in post-transient flows using Koopman mode decomposition. *Phys. Rev. Fluids* **2** (12), 124402.
- ARGYRIOU, A., EVGENIOU, T. & PONTIL, M. 2008a Convex multi-task feature learning. *Mach. Learn.* **73** (3), 243–272.
- ARGYRIOU, A., PONTIL, M., YING, Y. & MICCHELLI, C.A. 2008b A spectral regularization framework for multi-task structure learning. In *Advances in Neural Information Processing Systems*, pp. 25–32.
- ARONSZAJN, N. 1950 Theory of reproducing kernels. *Trans. Am. Math. Soc.* **68** (3), 337–404.
- ASKHAM, T. & KUTZ, J.N. 2018 Variable projection methods for an optimized dynamic mode decomposition. *SIAM J. Appl. Dyn. Syst.* **17** (1), 380–416.
- AZENCOT, O., YIN, W. & BERTOZZI, A. 2019 Consistent dynamic mode decomposition. *SIAM J. Appl. Dyn. Syst.* **18** (3), 1565–1585.
- BACH, F., JENATTON, R., MAIRAL, J. & OBOZINSKI, G. 2012 Optimization with sparsity-inducing penalties. *Found. Trends Mach. Learn.* **4** (1), 1–106.
- BAGHERI, S. 2013 Koopman-mode decomposition of the cylinder wake. *J. Fluid Mech.* **726**, 596–623.
- BAUDIN, M. 2015. pyDOE. <https://github.com/tisimst/pyDOE>.
- BLACKFORD, L.S., *et al.* 1997 *ScaLAPACK Users' Guide*. Society for Industrial and Applied Mathematics.
- BRUCE, A.L., ZEIDAN, V.M. & BERNSTEIN, D.S. 2019 What is the Koopman operator? a simplified treatment for discrete-time systems. In *2019 American Control Conference (ACC)*, pp. 1912–1917. IEEE.

- BRUNTON, S.L., BRUNTON, B.W., PROCTOR, J.L., KAISER, E. & KUTZ, J.N. 2017 Chaos as an intermittently forced linear system. *Nat. Commun.* **8** (1), 19.
- BRUNTON, S.L., BRUNTON, B.W., PROCTOR, J.L. & KUTZ, J.N. 2016a Koopman invariant subspaces and finite linear representations of nonlinear dynamical systems for control. *PLoS one* **11** (2), e0150171.
- BRUNTON, S.L., PROCTOR, J.L. & KUTZ, J.N. 2016b Discovering governing equations from data by sparse identification of nonlinear dynamical systems. *Proc. Natl Acad. Sci.* **113** (15), 3932–3937.
- BUDIŠIĆ, M., MOHR, R. & MEZIĆ, I. 2012 Applied koopmanism. *Chaos* **22** (4), 047510.
- CARLBERG, K., FARHAT, C., CORTIAL, J. & AMSALLEM, D. 2013 The gnat method for nonlinear model reduction: effective implementation and application to computational fluid dynamics and turbulent flows. *J. Comput. Phys.* **242**, 623–647.
- CHEN, K.K., TU, J.H. & ROWLEY, C.W. 2012 Variants of dynamic mode decomposition: boundary condition, Koopman, and fourier analyses. *J. Nonlinear Sci.* **22** (6), 887–915.
- CHO, Y. & SAUL, L.K. 2009 Kernel methods for deep learning. In *Advances in Neural Information Processing Systems*, pp. 342–350. Curran Associates, Inc.
- DALCIN, L.D., PAZ, R.R., KLER, P.A. & COSIMO, A. 2011 Parallel distributed computing using python. *Adv. Water Resour.* **34** (9), 1124–1139.
- DAWSON, S.T.M., HEMATI, M.S., WILLIAMS, M.O. & ROWLEY, C.W. 2016 Characterizing and correcting for the effect of sensor noise in the dynamic mode decomposition. *Exp. Fluids* **57** (3), 42.
- DEGENNARO, A.M. & URBAN, N.M. 2019 Scalable extended dynamic mode decomposition using random kernel approximation. *SIAM J. Sci. Comput.* **41** (3), A1482–A1499.
- DONGARRA, J., GATES, M., HAIDAR, A., KURZAK, J., LUSZCZEK, P., TOMOV, S. & YAMAZAKI, I. 2018 The singular value decomposition: anatomy of optimizing an algorithm for extreme scale. *SIAM Rev.* **60** (4), 808–865.
- DOWELL, E.H. & HALL, K.C. 2001 Modeling of fluid-structure interaction. *Annu. Rev. Fluid Mech.* **33** (1), 445–490.
- DRAZIN, P.G. & REID, W.H. 2004 *Hydrodynamic Stability*. Cambridge University Press.
- FEY, U., KÖNIG, M. & ECKELMANN, H. 1998 A new Strouhal–Reynolds-number relationship for the circular cylinder in the range $47 < Re < 2 \times 10^5$. *Phys. Fluids* **10** (7), 1547–1549.
- FORREST, J.S. & OWEN, I. 2010 An investigation of ship airwakes using detached-eddy simulation. *Comput. Fluids* **39** (4), 656–673.
- HASELI, M. & CORTÉS, J. 2019 Efficient identification of linear evolutions in nonlinear vector fields: Koopman invariant subspaces. [arXiv:1909.01419](https://arxiv.org/abs/1909.01419).
- HOLMES, P., LUMLEY, J.L., BERKOOZ, G. & ROWLEY, C.W. 2012 *Turbulence, Coherent Structures, Dynamical Systems and Symmetry*. Cambridge University Press.
- HUA, J.-C., NOORIAN, F., MOSS, D., LEONG, P.H.W. & GUNARATNE, G.H. 2017 High-dimensional time series prediction using kernel-based Koopman mode regression. *Nonlinear Dyn.* **90** (3), 1785–1806.
- HUANG, C., DURAISAMY, K. & MERKLE, C. 2018 Challenges in reduced order modeling of reacting flows. In *2018 Joint Propulsion Conference*, p. 4675. AIAA.
- JASAK, H., JEMCOV, A. & TUKOVIC, Z. 2007 Openfoam: a c++ library for complex physics simulations. In *International Workshop on Coupled Methods in Numerical Dynamics*, vol. 1000, pp. 1–20. IUC Dubrovnik Croatia.
- JONES, E., OLIPHANT, T. & PETERSON, P. 2001 *SciPy: Open Source Scientific Tools for Python*.
- JOVANOVIĆ, M.R., SCHMID, P.J. & NICHOLS, J.W. 2014 Sparsity-promoting dynamic mode decomposition. *Phys. Fluids* **26** (2), 024103.
- JUNGERS, R.M. & TABUADA, P. 2019 Non-local linearization of nonlinear differential equations via polyflows. In *2019 American Control Conference (ACC)*, pp. 1–6. IEEE.
- KAISER, E., KUTZ, J.N. & BRUNTON, S.L. 2017 Data-driven discovery of Koopman eigenfunctions for control. [arXiv:1707.01146](https://arxiv.org/abs/1707.01146).
- KAMB, M., KAISER, E., BRUNTON, S.L. & KUTZ, J.N. 2018 Time-delay observables for Koopman: theory and applications. [arXiv:1810.01479](https://arxiv.org/abs/1810.01479).
- KOOPMAN, B.O. 1931 Hamiltonian systems and transformation in Hilbert space. *Proc. Natl Acad. Sci. USA* **17** (5), 315.
- KORDA, M. & MEZIĆ, I. 2018a Linear predictors for nonlinear dynamical systems: Koopman operator meets model predictive control. *Automatica* **93**, 149–160.
- KORDA, M. & MEZIĆ, I. 2018b On convergence of extended dynamic mode decomposition to the Koopman operator. *J. Nonlinear Sci.* **28** (2), 687–710.
- KOU, J. & ZHANG, W. 2017 An improved criterion to select dominant modes from dynamic mode decomposition. *Eur. J. Mech. (B/Fluids)* **62**, 109–129.

- KUTZ, J.N., BRUNTON, S.L., BRUNTON, B.W. & PROCTOR, J.L. 2016 *Dynamic Mode Decomposition: Data-Driven Modeling of Complex Systems*. SIAM.
- LE CLAINCHE, S. & VEGA, J.M. 2017 Higher order dynamic mode decomposition. *SIAM J. Appl. Dyn. Syst.* **16** (2), 882–925.
- LI, Q., DIETRICH, F., BOLLT, E.M. & KEVREKIDIS, I.G. 2017 Extended dynamic mode decomposition with dictionary learning: a data-driven adaptive spectral decomposition of the Koopman operator. *Chaos* **27** (10), 103111.
- LIETZ, A.M., JOHNSEN, E. & KUSHNER, M.J. 2017 Plasma-induced flow instabilities in atmospheric pressure plasma jets. *Appl. Phys. Lett.* **111** (11), 114101.
- LUSCH, B., KUTZ, J.N. & BRUNTON, S.L. 2018 Deep learning for universal linear embeddings of nonlinear dynamics. *Nat. Commun.* **9** (1), 4950.
- MAMAKOUKAS, G., CASTANO, M.L., TAN, X. & MURPHEY, T. 2019 Local Koopman operators for data-driven control of robotic systems. In *Robotics: Science and Systems*.
- MERCER, J. 1909 Functions of positive and negative type and their connection with the theory of integral equations. In *Philosophical Transactions of the Royal Society of London, Series*.
- MEZIĆ, I. 2005 Spectral properties of dynamical systems, model reduction and decompositions. *Nonlinear Dyn.* **41** (1–3), 309–325.
- MEZIĆ, I. 2013 Analysis of fluid flows via spectral properties of the Koopman operator. *Annu. Rev. Fluid Mech.* **45**, 357–378.
- MINH, H.Q. 2010 Some properties of gaussian reproducing Kernel Hilbert spaces and their implications for function approximation and learning theory. *Construct. Approx.* **32** (2), 307–338.
- NOACK, B.R., AFANASIEV, K., MORZYŃSKI, M., TADMOR, G. & THIELE, F. 2003 A hierarchy of low-dimensional models for the transient and post-transient cylinder wake. *J. Fluid Mech.* **497**, 335–363.
- OTTO, S.E. & ROWLEY, C.W. 2019 Linearly recurrent autoencoder networks for learning dynamics. *SIAM J. Appl. Dyn. Syst.* **18** (1), 558–593.
- PAN, S. & DURAISAMY, K. 2018 Long-time predictive modeling of nonlinear dynamical systems using neural networks. *Complexity* **2018**, 4801012.
- PAN, S. & DURAISAMY, K. 2019 On the structure of time-delay embedding in linear models of non-linear dynamical systems. [arXiv:1902.05198](https://arxiv.org/abs/1902.05198).
- PAN, S. & DURAISAMY, K. 2020 Physics-informed probabilistic learning of linear embeddings of nonlinear dynamics with guaranteed stability. *SIAM J. Appl. Dyn. Syst.* **19** (1), 480–509.
- PARIKH, N. & BOYD, S. 2014 Proximal algorithms. *Found. Trends® Optim.* **1** (3), 127–239.
- PARISH, E.J., WENTLAND, C.R. & DURAISAMY, K. 2020 The adjoint Petrov–Galerkin method for non-linear model reduction. *Comput. Meth. Appl. Mech. Engng* **365**, 112991.
- PARRY, W. 2004 *Topics in Ergodic Theory*, vol. 75. Cambridge University Press.
- PDRÉGOSA, F., *et al.* 2011 Scikit-learn: machine learning in Python. *J. Mach. Learn. Res.* **12**, 2825–2830.
- POPE, S.B. 2001 *Turbulent Flows*. Cambridge University Press.
- RASMUSSEN, C.E. 2003 Gaussian processes in machine learning. In *Summer School on Machine Learning*, pp. 63–71. Springer.
- RÖJSEL, J. 2017 Koopman mode analysis of the side-by-side cylinder wake. *Tech. Rep.* Royal Institute of Technology, Department of Mechanics.
- ROWLEY, C.W. & DAWSON, S.T.M. 2017 Model reduction for flow analysis and control. *Annu. Rev. Fluid Mech.* **49**, 387–417.
- ROWLEY, C.W., MEZIĆ, I., BAGHERI, S., SCHLATTER, P. & HENNINGSON, D.S. 2009 Spectral analysis of nonlinear flows. *J. Fluid Mech.* **641**, 115–127.
- SCHMID, P.J. 2007 Nonmodal stability theory. *Annu. Rev. Fluid Mech.* **39**, 129–162.
- SCHMID, P.J. 2010 Dynamic mode decomposition of numerical and experimental data. *J. Fluid Mech.* **656**, 5–28.
- SCHMID, P.J., VIOLATO, D. & SCARANO, F. 2012 Decomposition of time-resolved tomographic PIV. *Exp. Fluids* **52** (6), 1567–1579.
- SHARMA, A., XU, J., PADTHE, A.K., FRIEDMANN, P.P. & DURAISAMY, K. 2019 Simulation of maritime helicopter dynamics during approach to landing with time-accurate wind-over-deck. In *AIAA Scitech 2019 Forum*, p. 0861. AIAA.
- TAIRA, K., HEMATI, M.S., BRUNTON, S.L., SUN, Y., DURAISAMY, K., BAGHERI, S., DAWSON, S.T.M. & YEH, C.-A. 2019 Modal analysis of fluid flows: applications and outlook. *AIAA J.* **58** (3), 998–1022.
- TAKEISHI, N., KAWAHARA, Y. & YAIRI, T. 2017 Learning Koopman invariant subspaces for dynamic mode decomposition. In *Advances in Neural Information Processing Systems*, pp. 1130–1140.
- TISSOT, G., CORDIER, L., BENARD, N. & NOACK, B.R. 2014 Model reduction using dynamic mode decomposition. *C. R. Méc* **342** (6–7), 410–416.

- TU, J.H., ROWLEY, C.W., LUCHTENBURG, D.M., BRUNTON, S.L. & KUTZ, J.N. 2013 On dynamic mode decomposition: theory and applications. [arXiv:1312.0041](https://arxiv.org/abs/1312.0041).
- WANG, L., WANG, J., XIANG, J. & HUI YUE, H. 2019 A re-weighted smoothed l_0 -norm regularized sparse reconstructed algorithm for linear inverse problems. *J. Phys. Commun.* **3** (7), 075004.
- WHITE, F.M. & CORFIELD, I. 2006 *Viscous Fluid Flow*, vol. 3. McGraw-Hill.
- WILLIAMS, M.O., KEVREKIDIS, I.G. & ROWLEY, C.W. 2015 A data-driven approximation of the Koopman operator: extending dynamic mode decomposition. *J. Nonlinear Sci.* **25** (6), 1307–1346.
- WILLIAMS, M.O., ROWLEY, C.W. & KEVREKIDIS, I.G. 2014 A kernel-based approach to data-driven Koopman spectral analysis. [arXiv:1411.2260](https://arxiv.org/abs/1411.2260).
- XU, J. & DURASAMY, K. 2019 Multi-level convolutional autoencoder networks for parametric prediction of spatio-temporal dynamics. [arXiv:1912.11114](https://arxiv.org/abs/1912.11114).
- XU, J., HUANG, C. & DURASAMY, K. 2020 Reduced-order modeling framework for combustor instabilities using truncated domain training. *AIAA J.* **58** (2), 618–632.
- YEUNG, E., KUNDU, S. & HODAS, N. 2019 Learning deep neural network representations for Koopman operators of nonlinear dynamical systems. In *2019 American Control Conference (ACC)*, pp. 4832–4839. IEEE.
- YUAN, W., WALL, A. & LEE, R. 2018 Combined numerical and experimental simulations of unsteady ship airwakes. *Comput. Fluids* **172**, 29–53.
- ZHANG, H., DAWSON, S., ROWLEY, C.W., DEEM, E.A. & CATTAFESTA, L.N. 2017 Evaluating the accuracy of the dynamic mode decomposition. [arXiv:1710.00745](https://arxiv.org/abs/1710.00745).
- ZHANG, L. & SCHAEFFER, H. 2019 On the convergence of the sindy algorithm. *Multiscale Model. Simul.* **17** (3), 948–972.
- ZHAO, L., SUN, Q., YE, J., CHEN, F., LU, C.-T. & RAMAKRISHNAN, N. 2015 Multi-task learning for spatio-temporal event forecasting. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1503–1512. ACM.
- ZOU, H. & HASTIE, T. 2005 Regularization and variable selection via the elastic net. *J. R. Stat. Soc. Ser. B (Stat. Methodol.)* **67** (2), 301–320.