ORIGINAL PAPER

Fast data-driven model reduction for nonlinear dynamical systems

Joar Axås · **Mattia Cenedese** · **George Haller**

Abstract We present a fast method for nonlinear data-driven model reduction of dynamical systems onto their slowest nonresonant spectral submanifolds (SSMs). While the recently proposed reduced-order modeling method SSMLearn uses implicit optimization to fit a spectral submanifold to data and reduce the dynamics to a normal form, here, we reformulate these tasks as explicit problems under certain simplifying assumptions. In addition, we provide a novel method for time-lag selection when delay-embedding signals from multimodal systems. We show that our alternative approach to data-driven SSM construction yields accurate and sparse rigorous models for essentially nonlinear (or *non-linearizable*) dynamics on both numerical and experimental datasets. Aside from a major reduction in complexity, our new method allows an increase in the training data dimensionality by several orders of magnitude. This promises to extend data-driven, SSM-based modeling to problems with hundreds of thousands of degrees of freedom.

J. Axås · M. Cenedese · G. Haller (✉)
Institute for Mechanical Systems, ETH Zürich,
Leonhardstrasse 21, 8092 Zurich, Switzerland
e-mail: georgehaller@ethz.ch

J. Axås
e-mail: jgoeransson@ethz.ch

1 Introduction

Nonlinear dynamical systems are omnipresent in nature and engineering. Examples include beam and plate buckling [1], turbulent fluid flows [24], vibrations in jointed structures [33], sloshing in fluid tanks [54], and even traffic jams [40]. As computational resources have grown, so has the interest in data-driven methods, which take input data from experiments or simulations and return a reduced model of the underlying system dynamics. To date, however, no rigorous method has been accepted as a standard for nonlinear system identification and reduced modeling.

Model simplicity (or parsimony) is vital for interpretability, control, and response prediction for mechanical devices [31]. This has motivated reduction methods based partially or fully on linearization of the underlying dynamics, such as the proper orthogonal decomposition [3,38] and the dynamic mode decomposition (DMD) [48,49]. Specifically, DMD obtains the best fit of a linear dynamical system to the data in an equation-free manner [32], often utilizing delay embedding to secure sufficiently many observables [17].

Many nonlinear mechanical systems, however, exhibit phenomena that cannot occur in linear systems, such as coexisting isolated steady states, and therefore cannot

be captured by linear models [41]. We refer to such phenomena as *non-linearizable*. Machine learning methods can potentially capture such phenomena [7,14,23,39], but tend to provide models that lack interpretability and perform poorly outside their training range [37].

Here, we propose spectral submanifolds (SSMs) to obtain sparse models of nonlinearizable phenomena. An SSM is the smoothest nonlinear continuation of a nonresonant spectral subspace emanating from a steady state, both in autonomous systems and systems with periodic or quasiperiodic forcing. SSMs are unique, attracting, and persistent invariant manifolds in the phase space, that lend themselves well to model reduction [22].

A concept related to SSMs is a nonlinear normal mode (NNM), which was originally defined as a periodic motion of a nonlinear system [30,45,55]. A later definition of NNMs [50] targets invariant manifolds tangent to spectral subspaces of the linearized system without focusing on the unique, smoothest such manifold. In the sense of this third NNM definition, SSMs are special NNMs with precisely known smoothness and uniqueness properties. They can be constructed over an arbitrary number of linearized modes which may also be in resonance with each other. Recently, it has been proven that two-dimensional SSMs converge to conservative NNMs composed of periodic orbits when the damping in the system tends to zero [36]. Closely linked to SSMs, invariant spectral foliations are the basis of another rigorous approach for extending linear modal analysis to nonlinear systems [51].

Spectral submanifolds can be rigorously constructed for mechanical systems defined by differential equations [43]. Given a Taylor expansion of a set of autonomous ODEs, the coefficients of an SSM can be efficiently computed up to any order and dimension [26]. This methodology has been used for model reduction of systems with hundreds of thousands of degrees of freedom, accurately predicting responses to small harmonic forcing [25,27,42], as well as the bifurcations of those responses [34,35].

These developments have motivated applications of SSM theory to locate reduced-order models from data. As a first step, Ref. [52] fitted a multivariate polynomial to the sampling map and computed SSMs on the resulting Taylor expanded dynamics. This yielded good reconstructions of the backbone curves in a clamped-clamped beam experiment. Typically, however, due to the rapid growth in the number of terms, a polynomial expansion of the flow map in the full observable space is sensitive to overfitting and quickly becomes intractable in higher-dimensional systems.

Recently, another data-driven SSM identification method was proposed to alleviate the overfitting of the sampling map by separating the model reduction into two steps: manifold identification and reduced dynamics fitting [10]. Thus, the polynomial fit of the dynamics takes place only on the SSM, and hence the number of terms no longer grows with the data dimensionality. First, the data is embedded in a user-defined observable space. Then a polynomial representation of the SSM is fitted to the data and the data dimensionality is reduced by projection onto the tangent space of the SSM. Using nonlinear optimization techniques, finally a transformation from the reduced coordinates to a normal form is computed, maximizing sparsity while retaining essential nonlinearities [21].

While data-driven SSM-based model reduction has been successfully applied to both numerical and experimental data for fluid problems, structural dynamics, and fluid-structure interaction [10,11,29], the required implicit optimization can be computationally demanding for high-dimensional systems. This paper introduces an alternative method that explicitly computes an SSM followed by a classic normal form transformation in the reduced coordinates. Our key assumption is that the tangent space of the SSM at a fixed point can be found by singular value decomposition (SVD). This enables a major simplification and speedup at the expense of giving up the more general applicability of the method in Ref. [10]. In consequence, our proposed method is faster than previous SSM-based model reduction from data both in terms of implementation and execution. We do not undertake any formal speed comparison with other methods here.

After reduction to the computed SSM and numerical fitting of the reduced dynamics, we compute the normal form of the reduced dynamics analytically. This is in contrast to Ref. [10], which fits the normal form directly to data simultaneously at all required orders. The aims of our new approach are accessibility to practitioners, major speedup for rapid prototyping, analysis of higher-dimensional observable spaces, and insight into the differences between numerically and analytically computed normal forms.

The structure of this paper is as follows. First, Sect. 2 gives a brief introduction to SSM theory, summarizes SSM-based data-driven modeling, and details

our new method. In Sect. 3, we discuss how to select an observable space using delay embedding and use SSM-reduced models for forced response prediction. In Sect. 4, we apply our newly derived SSM-based model reduction method to experiments from a sloshing tank, simulations of a von Kármán beam, and experiments on an internally resonant beam. Finally, in Sect. 5, we draw conclusions from the examples, suggest additional applications, and outline possible further enhancements to dynamics-based machine learning.

2 Model order reduction on spectral submanifolds

We consider nonlinear autonomous dynamical systems of class $\mathscr{C}^l$, $l \in \{\mathbb{N}^+, \infty, a\}$, with a denoting analyticity, in the form

$$\dot{\boldsymbol{y}} = \boldsymbol{A}\boldsymbol{y} + \boldsymbol{f}(\boldsymbol{y}), \quad \boldsymbol{y} \in \mathbb{R}^p, \quad \boldsymbol{f} \sim \mathscr{O}(|\boldsymbol{y}|^2),$$
$$\boldsymbol{f} : \mathbb{R}^p \to \mathbb{R}^p. \tag{1}$$

We assume $\boldsymbol{A} \in \mathbb{R}^{p \times p}$ is diagonalizable and the real parts of its eigenvalues are either all strictly negative or all strictly positive. Note that this condition requires the origin to be a hyperbolic fixed point, which excludes rigid body modes from the analysis. We denote by $\mathscr{V}_d$ the slowest d-dimensional spectral subspace of $\boldsymbol{A}$, i.e., the span of the eigenvectors corresponding to the d eigenvalues with real part closest to zero. When all eigenvalues have negative real parts, the dynamics of the linearized system will converge to $\mathscr{V}_d$ in forward time.

If the eigenvalues corresponding to $\mathscr{V}_d$ are non-resonant with the remaining $p - d$ eigenvalues of $\boldsymbol{A}$, $\mathscr{V}_d$ admits a unique smoothest, invariant, nonlinear continuation $\mathscr{M}$ under addition of the higher-order terms [9]. We refer to $\mathscr{M}$ as a spectral submanifold (SSM) [11,22]. In the case of a resonance between $\mathscr{V}_d$ and the rest of the spectrum of $\boldsymbol{A}$, we can include the resonant modal subspace in $\mathscr{V}_d$ and thus obtain a higher-dimensional SSM. If all eigenvalues of $\boldsymbol{A}$ have negative real parts, $\mathscr{M}$ will be an attracting slow manifold for system (1), just as $\mathscr{V}_d$ is for the linear part of (1).

A numerical package, SSMTool, has been developed for the computation of SSMs from arbitrary finite-dimensional nonlinear systems [25,26]. More recently, a data-driven method was developed to compute SSMs purely from observables of the dynamical system [10,11]. In the following, we first review the literature on data-driven reduced-order modeling on

SSMs, and then propose a simplified approach that is applicable under further assumptions.

2.1 Learning spectral submanifolds from data

We seek a parametrization of $\mathscr{M}$ from trajectories in an observable space, which may be the full phase space or a suitable embedding, as described in Sect. 3.1. The data-driven methods we will put forward consist of two steps: manifold fitting and normal form computation.

We first outline the method recently proposed by Ref. [10], as implemented in the algorithm SSMLearn [12]. In that approach, one computes a parametrization of the SSM as a multivariate polynomial

$$\boldsymbol{y}(\boldsymbol{\xi}) = \boldsymbol{M}\boldsymbol{\xi}^{1:m} = \boldsymbol{V}\boldsymbol{\xi} + \boldsymbol{M}_{2:m}\boldsymbol{\xi}^{2:m},$$
$$\boldsymbol{M} = [\boldsymbol{V}, \boldsymbol{M}_2, \ldots, \boldsymbol{M}_m], \quad \boldsymbol{M}_i \in \mathbb{R}^{p \times d_i}, \tag{2}$$

with the reduced coordinates $\boldsymbol{\xi} = \boldsymbol{V}^\top \boldsymbol{y}$ and the matrix $\boldsymbol{V} \in \mathbb{R}^{p \times d}$ containing d orthonormal vectors spanning the spectral subspace $\mathscr{V}_d$. Here, d_i denotes the number of d-variate monomials at order i. Throughout this paper, the superscript $(\cdot)^{l:r}$ denotes a vector of all monomials at orders l through r. For example, if $\boldsymbol{x} = [x_1, x_2]^\top$, then

$$\boldsymbol{x}^{1:3} = \left[x_1, x_2, x_1^2, x_1 x_2, x_2^2, x_1^3, x_1^2 x_2, x_1 x_2^2, x_2^3 \right]^\top.$$

Minimization of the distance of the parametrized SSM from the training data points $\boldsymbol{y}_j$ yields the optimal coefficient matrix $\boldsymbol{M}^\star \in \mathbb{R}^{p \times d_{1:m}}$, with $d_{1:m}$ denoting the number of d-variate monomials from orders 1 up to m, as

$$\boldsymbol{M}^\star = [\boldsymbol{V}^\star, \boldsymbol{M}_{2:m}^\star]$$
$$= \underset{\boldsymbol{V}, \boldsymbol{M}_{2:m}}{\arg\min} \sum_j \left\| \boldsymbol{y}_j - \boldsymbol{V}\boldsymbol{V}^\top \boldsymbol{y}_j - \boldsymbol{M}_{2:m}(\boldsymbol{V}^\top \boldsymbol{y}_j)^{2:m} \right\|, \tag{3}$$

subject to the constraints

$$\boldsymbol{V}^\top \boldsymbol{V} = \boldsymbol{I}, \quad \boldsymbol{V}^\top \boldsymbol{M}_{2:m} = \boldsymbol{0}. \tag{4}$$

Next, one computes the reduced dynamics $\dot{\boldsymbol{\xi}} = \boldsymbol{R}\boldsymbol{\xi}^{1:r}$ on the SSM in polynomial form, with coefficients in the matrix $\boldsymbol{R} \in \mathbb{R}^{d \times d_{1:r}}$, by minimizing

$$\underset{\boldsymbol{R}}{\arg\min} \sum_j \left\| \dot{\boldsymbol{\xi}}_j - \boldsymbol{R}\boldsymbol{\xi}_j^{1:r} \right\|. \tag{5}$$

Finally, one computes the normal form of the reduced dynamics of $\boldsymbol{\xi}$ on the SSM [21]. This amounts

to seeking a nonlinear transformation $t^{-1} : \boldsymbol{\xi} \mapsto z$ to new coordinates $z \in \mathbb{C}^d$, that reduces the number of coefficients in $\boldsymbol{R}$ to a minimum set $\boldsymbol{N} \in \mathbb{C}^{d \times d_{1:n}}$. The transformation and normalized dynamics are given by

$$
\begin{aligned}
z &= t^{-1}(\boldsymbol{\xi}) = \boldsymbol{H}\boldsymbol{\xi}^{1:n} = \boldsymbol{W}^{-1}\boldsymbol{\xi} + \boldsymbol{H}_{2:n}\boldsymbol{\xi}^{2:n}, \\
\dot{z} &= \boldsymbol{n}(z) = \boldsymbol{N}z^{1:n} = \boldsymbol{\Lambda}z + \boldsymbol{N}_{2:n}z^{2:n},
\end{aligned}
\tag{6}
$$

where $\boldsymbol{H} \in \mathbb{C}^{d \times d_{1:n}}$, and $\boldsymbol{W} \in \mathbb{C}^{d \times d}$ is the matrix of eigenvectors of the linear part $\boldsymbol{R}_1 = \boldsymbol{W}\boldsymbol{\Lambda}\boldsymbol{W}^{-1}$ of the reduced dynamics. $\boldsymbol{N}$ is potentially a very sparse matrix, simplifying the dynamics to $\boldsymbol{n}(z)$. The locations of nonzero elements in $\boldsymbol{N}$ are determined by any approximate inner resonances between the eigenvalues in $\boldsymbol{\Lambda}$ [10]. An example is given in (17).

The transformation to the normal form on the SSM is computed by minimization of the conjugacy error

$$
\underset{\boldsymbol{N}, \boldsymbol{H}}{\operatorname{argmin}} \sum_j \left\| \nabla_{\boldsymbol{\xi}} t^{-1}(\boldsymbol{\xi}_j) \dot{\boldsymbol{\xi}}_j - \boldsymbol{n}(t^{-1}(\boldsymbol{\xi}_j)) \right\|.
\tag{7}
$$

Data-driven model reduction to SSMs has been successfully applied to both numerical and experimental datasets [10,11,29]. Yet, for very high-dimensional systems, the implicit optimization problems (3) and (7) may be intractable. Further, their solution would benefit from a simplified approach. Motivated by these challenges, in the remainder of this paper we discuss alternative, explicit formulations of the SSM geometry and its reduced normal form, which are possible under additional assumptions detailed below.

2.2 Fast spectral submanifold identification

In order to turn the SSM identification into an explicit problem, we assume that the tangent space of the SSM at the origin can be approximately obtained by a singular value decomposition (SVD) of the data. This is typically satisfied sufficiently close to an equilibrium point. Moreover, since the main variance in the data tends to occur along the tangent space of the SSM, the span of the first d principal directions given by SVD produces a good candidate for this subspace. A third motivation for our main assumption is that the image of $\mathcal{M}$ in a delay-embedded space tends to be flat, even if $\mathcal{M}$ is strongly curved in the full phase space [10]. An overview of the fast SSM construction method exploiting this simplification is shown in Fig. 1.

Specifically, we approximate the tangent space $\mathcal{V}_d$ by d-dimensional truncated SVD on the snapshot matrix

$\boldsymbol{Y} \in \mathbb{R}^{p \times N}$ as

$$
\boldsymbol{Y} \approx \boldsymbol{U}_d \boldsymbol{S}_d \hat{\boldsymbol{V}}_d^\top,
\tag{8}
$$

where $\boldsymbol{U}_d \in \mathbb{R}^{p \times d}, \boldsymbol{S}_d \in \mathbb{R}^{d \times d}, \hat{\boldsymbol{V}}_d \in \mathbb{R}^{N \times d}$. Defining the tangent space approximation as

$$
\boldsymbol{V} = \boldsymbol{U}_d \boldsymbol{S}_d^{-1},
\tag{9}
$$

we can project the trajectories available from data onto this approximate subspace to obtain d reduced coordinates $\boldsymbol{\xi}$ as

$$
\boldsymbol{\xi} = \boldsymbol{V}^\top \boldsymbol{y}.
\tag{10}
$$

We denote by $\boldsymbol{\Xi} \in \mathbb{R}^{d \times N}$ the projection of $\boldsymbol{Y}$ onto $\boldsymbol{V}$. Numerically, it is beneficial to first normalize each column of $\boldsymbol{V}$ with the maximum absolute value of the corresponding row in $\hat{\boldsymbol{V}}_d$.

This first step is equivalent to proper orthogonal decomposition [3,38], but our main interest in this reduced subspace is its use for fitting a nonlinear parametrization of the SSM. The manifold parametrization coefficients $\boldsymbol{M} \in \mathbb{R}^{p \times d_{1:m}}$ are obtained by polynomial regression:

$$
\boldsymbol{y} \approx \boldsymbol{M}\boldsymbol{\xi}^{1:m}, \quad \boldsymbol{M} = \boldsymbol{Y}(\boldsymbol{\Xi}^{1:m})^\dagger,
\tag{11}
$$

with $(\cdot)^\dagger$ denoting the Moore–Penrose pseudo-inverse.

The reduced dynamics are approximated by an $\mathcal{O}(r)$ polynomial regression, with the coefficient matrix $\boldsymbol{R} \in \mathbb{R}^{d \times d_{1:r}}$, as

$$
\dot{\boldsymbol{\xi}} \approx \boldsymbol{R}\boldsymbol{\xi}^{1:r}, \quad \boldsymbol{R} = \dot{\boldsymbol{\Xi}}(\boldsymbol{\Xi}^{1:r})^\dagger.
\tag{12}
$$

The time derivative $\dot{\boldsymbol{\xi}}$ may be computed numerically. The quality of this approximation is important for model accuracy. Therefore, we use a central finite difference method including 4 adjacent points in each time direction [20].

Next, diagonalizing the linear part $\boldsymbol{R}_1 = \boldsymbol{W}\boldsymbol{\Lambda}\boldsymbol{W}^{-1}$, we again apply regression to rewrite $\boldsymbol{R}$ in the form $\boldsymbol{G} \in \mathbb{C}^{d \times d_{1:r}}$ with the eigenvalues in the linear part $\boldsymbol{G}_1 = \boldsymbol{\Lambda}$ and modal coordinates $\boldsymbol{\zeta} = \boldsymbol{W}^{-1}\boldsymbol{\xi}$, such that

$$
\dot{\boldsymbol{\zeta}} \approx \boldsymbol{G}\boldsymbol{\zeta}^{1:r}.
\tag{13}
$$

To compute the normal form of the SSM-reduced dynamics (13), we seek a near-identity polynomial transformation $t : z \mapsto \boldsymbol{\zeta}$ with coefficients $\boldsymbol{T} \in \mathbb{C}^{d \times d_{1:n}}$ from new coordinates $z \in \mathbb{C}^d$ such that

$$
\begin{aligned}
\boldsymbol{\zeta} &= t(z) = \boldsymbol{T}z^{1:n} = z + \boldsymbol{T}_{2:n}z^{2:n} \\
\dot{z} &= \boldsymbol{n}(z) = \boldsymbol{N}z^{1:n} = \boldsymbol{\Lambda}z + \boldsymbol{N}_{2:n}z^{2:n}.
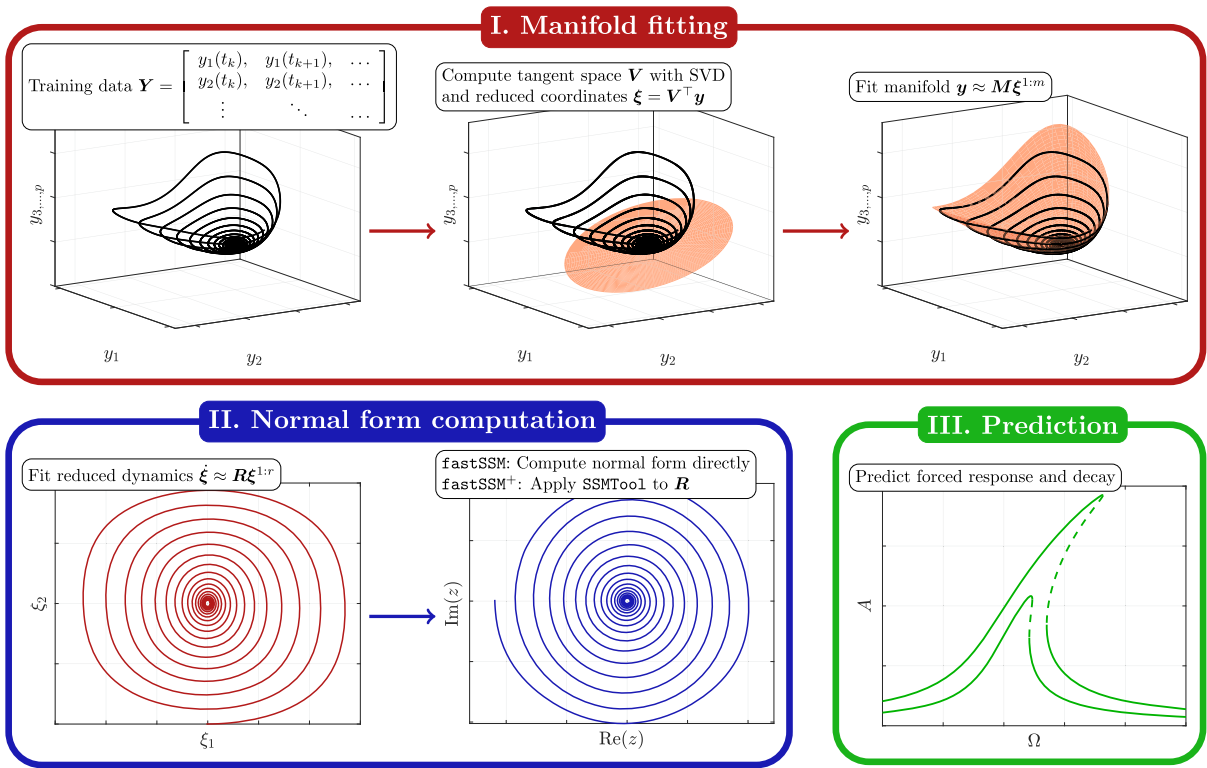\end{aligned}
\tag{14}
$$

Fig. 1 Our simplified data-driven SSM construction consists of two main parts, (I) fitting an SSM with SVD and polynomial regression, and (II) computing a normal form on the manifold by analytical solution of the cohomological equations. The obtained reduced model can then be used for (III) prediction of the autonomous system evolution or the response of the forced system

We enforce conjugacy between the normal form and reduced dynamics by plugging (14) into (13):

$$\nabla_z(\boldsymbol{T}\boldsymbol{z}^{1:n})\boldsymbol{N}\boldsymbol{z}^{1:n} = \boldsymbol{G}(\boldsymbol{T}\boldsymbol{z}^{1:n})^{1:r}. \tag{15}$$

We can compute $\boldsymbol{T}$ and $\boldsymbol{N}$ by solving (15) recursively at increasing orders. See Ref. [25] for details on the computations.

The simplest non-trivial normal form arises on a 2D manifold emanating from a spectral subspace corresponding to two complex conjugate eigenvalues $(\lambda, \bar{\lambda})$ with small real part. We denote the coordinates in the normal form $(z_1, z_2) = (z, \bar{z})$. The cubic normal form, with $\gamma \in \mathbb{C}$, is

$$
\begin{aligned}
\dot{z} &= \lambda z + \gamma z^2 \bar{z} + \mathscr{O}(5) \\
\dot{\bar{z}} &= \bar{\lambda} \bar{z} + \bar{\gamma} z \bar{z}^2 + \mathscr{O}(5),
\end{aligned} \tag{16}
$$

for which we obtain

$$\boldsymbol{N} = \begin{bmatrix} \lambda & 0 & 0 & 0 & 0 & 0 & \gamma & 0 & 0 \\ 0 & \bar{\lambda} & 0 & 0 & 0 & 0 & 0 & \bar{\gamma} & 0 \end{bmatrix}. \tag{17}$$

Solving (15) yields

$$\boldsymbol{T} = \begin{bmatrix} 1 & 0 & T_3 & T_4 & T_5 & T_6 & 0 & T_8 & T_9 \\ 0 & 1 & \bar{T}_5 & \bar{T}_4 & \bar{T}_3 & \bar{T}_9 & \bar{T}_8 & 0 & \bar{T}_6 \end{bmatrix}, \tag{18}$$

where

$$T_3 = \frac{G_3}{\lambda}, \quad T_4 = \frac{G_4}{\bar{\lambda}}, \quad T_5 = \frac{G_5}{2\bar{\lambda} - \lambda},$$

$$T_6 = \frac{2G_3 T_3 + G_4 \bar{T}_5 + G_6}{2\lambda},$$

$$T_8 = \frac{2G_3 T_5 + G_4 \bar{T}_3 + G_4 T_4 + 2G_5 \bar{T}_4 + G_8}{2\bar{\lambda}},$$

$$T_9 = \frac{G_4 T_5 + 2G_5 \bar{T}_3 + G_9}{3\bar{\lambda} - \lambda}, \tag{19}$$

and G_i refers to element i of the top row in $\boldsymbol{G}$ [52].

We refer to the implementation of this simplified method for cubic-order, 2D normal form computations as fastSSM. The full script is written out in "Appendix A.1". We have the reduced dynamics and

normal form orders $r = 3$ and $n = 3$, and we are free to set the order m of the manifold.

While a cubic order normal form will suffice to model a number of datasets accurately [10,11], for stronger nonlinearities we must include higher orders. In the following, we refer to the extension of the manifold fitting and normal form computation to arbitrary dimension and order as fastSSM$^+$. This implementation solves (15) by automatically applying SSMTool to the coefficients of $\boldsymbol{R}$. The full code is written out in "Appendix A.2". In principle, we are free to choose any orders m, r, and n, but to avoid overfitting we must limit the manifold order m and reduced dynamics order r. As an objective function for choosing these parameters, we minimize the mean error of the predictions of the training trajectories, as defined in (32). In addition, we need to pick a large enough n to make the tail of the SSM-reduced dynamics small enough.

Unlike (3) and (7), all computations in our proposed method are explicit. Therefore, we expect this approach to be faster than SSMLearn, which will generally come at the cost of increased modeling errors. Further, we note that in this simplified approach, the normal form on the SSM is computed analytically, rather than numerically, from a polynomial fit of the data.

In practice we must also approximate the inverse of $\boldsymbol{t}$ to transform initial conditions to the normal form before the integration. The most accurate and simplest way is a numerical solution for each initial condition, but a general inverse map can also be computed. Accordingly, we use an explicit 3rd-order polynomial expansion of the inverse of $\boldsymbol{t}$ in the fastSSM algorithm. In contrast, in the fastSSM$^+$ algorithm for higher SSM dimensions and normal form orders, we fit a polynomial map by regression on the training data.

Finally, we note that the simplified SSM computation described here requires the data to lie sufficiently close to the SSM. We achieve this by removing any initial transients from the training dataset, as identified by a spectral analysis on the input signal [11]. Since the SSM is unique and attracting, this procedure ensures that we train on relevant data.

3 Pre- and post-processing of the data

We present here considerations for the delay embedding of observable functions to facilitate SSM identification. We also show how to use the obtained normal form dynamics to predict the forced response of the full system.

3.1 Embedding spectral submanifolds in the observable space

In practice, observing trajectories $\boldsymbol{x}(t)$ in the full phase space is often intractable for high-dimensional systems. An experimentalist might only observe a scalar quantity $s(t) = \zeta(\boldsymbol{x}(t))$, say, the displacement or acceleration at one point in a vibrating structure. Nevertheless, if the observable function $\zeta : \mathbb{R}^{n_{\text{full}}} \to \mathbb{R}$ is generic, it can be used to reconstruct an invariant manifold of the system in an observable space via delay embedding.

To this end, we collect p measurements separated by a timelag $\tau > 0$ to form a vector $\boldsymbol{y}$ in an observable space $\mathbb{R}^p$ as

$$\boldsymbol{y}(t) = [s(t), s(t + \tau), s(t + 2\tau), \dots, s(t + (p - 1)\tau)]^\top.$$

Takens' embedding theorem implies that if $\boldsymbol{x}(t)$ lies on an invariant d-dimensional manifold $\mathcal{M}$ in the original phase space, then $\boldsymbol{y}(t)$ lies on a diffeomorphic copy of $\mathcal{M}$ in $\mathbb{R}^p$ with probability one, if $p \geq 2d + 1$ and $\zeta(\boldsymbol{x})$ is a generic observable [53]. This result also extends to generic multivariate functions $\boldsymbol{\zeta} : \mathbb{R}^{n_{\text{full}}} \to \mathbb{R}^{n_{\text{obs}}}$ as long as the total observable space dimension exceeds $2d$ [15,47]. Even with a high-dimensional observable, however, we apply delay embedding to diversify the data and unveil information about the time derivative of the signal in our later sloshing example [11].

The timelag τ is typically a multiple of the sampling time or timestep Δt of the available dataset. The default choice is $\tau = \Delta t$, but one can also increase the timelag to $\tau = k\Delta t$, $k \in \mathbb{N}_+$. Delay-embedding an observed time series $[s(t_1), \dots, s(t_N)]$ thus yields the snapshot matrix

$$\boldsymbol{Y} = \begin{bmatrix} s(t_1) & s(t_2) & \dots s(t_{N+(1-p)k}) \\ s(t_{k+1}) & s(t_{k+2}) & \dots s(t_{N+(2-p)k}) \\ s(t_{2k+1}) & s(t_{2k+2}) & \dots s(t_{N+(3-p)k}) \\ \vdots & \vdots & \ddots & \vdots \\ s(t_{(p-1)k+1}) & s(t_{(p-1)k+2}) \dots & s(t_N) \end{bmatrix}.$$
$$(20)$$

Takens' embedding theorem requires at least $k = 1$, $p = 2d + 1$, which typically suffices for SSM identification when $d = 2$.

If we observe a smooth scalar signal $s(t)$, then for small τ, we have

$$s(t + j\tau) = s(t) + \dot{s}(t)j\tau + \mathcal{O}((j\tau)^2).$$

Delay-embedding $s(t)$ then yields

$$
\mathbf{y}(t) = \begin{bmatrix} s(t) \\ s(t+\tau) \\ s(t+2\tau) \\ \vdots \\ s(t+(p-1)\tau) \end{bmatrix} = s(t) \begin{bmatrix} 1 \\ 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}
$$

$$
+ \dot{s}(t)\tau \begin{bmatrix} 0 \\ 1 \\ 2 \\ \vdots \\ p-1 \end{bmatrix} + \mathcal{O}((p\tau)^2). \tag{21}
$$

As noted in Ref. [10], if τ and p are small in comparison to $\dot{s}$, then the diffeomorphic copy of $\mathcal{M}$ in the observable space is approximately a plane spanned by the two constant vectors appearing in (21).

For multimodal signals, however, p and k must be increased to allow distinction between the modes in the observable space. The modal subspaces of the full phase space have corresponding planes in the observable space, which must be identified to provide reduced coordinates for the SSM. A low delay embedding dimension and an overly short timelag can make these planes close to parallel, which complicates their identification. In contrast, we want to pick the timelag and embedding dimension such that the images of the modal subspaces in the observable space are close to orthogonal.

To illustrate this, we consider an observed superposition of two harmonics $s(t) = a\cos(\omega_1 t + \psi_1) + b\cos(\omega_2 t + \psi_2)$, a reasonable model of a typical signal close to the origin. Using a trigonometric identity, we can rewrite

$$
\begin{aligned}
s(t+j\tau) &= a\cos(\omega_1 t + \psi_1)\cos(\omega_1 j\tau) \\
&\quad - a\sin(\omega_1 t + \psi_1)\sin(\omega_1 j\tau) \\
&\quad + b\cos(\omega_2 t + \psi_2)\cos(\omega_2 j\tau) \\
&\quad - b\sin(\omega_2 t + \psi_2)\sin(\omega_2 j\tau).
\end{aligned} \tag{22}
$$

The delay-embedded observable vector is therefore

$$
\begin{aligned}
\mathbf{y}(t) &= \begin{bmatrix} s(t) \\ s(t+\tau) \\ s(t+2\tau) \\ \vdots \\ s(t+(p-1)\tau) \end{bmatrix} \\
&= \mathbf{V} \begin{bmatrix} a\cos(\omega_1 t + \psi_1) \\ -a\sin(\omega_1 t + \psi_1) \\ b\cos(\omega_2 t + \psi_2) \\ -b\sin(\omega_2 t + \psi_2) \end{bmatrix},
\end{aligned} \tag{23}
$$

where

$$
\mathbf{V}^\top = \begin{bmatrix} 1 & \cos(\omega_1\tau) & \cos(2\omega_1\tau) & \dots & \cos((p-1)\omega_1\tau) \\ 0 & \sin(\omega_1\tau) & \sin(2\omega_1\tau) & \dots & \sin((p-1)\omega_1\tau) \\ 1 & \cos(\omega_2\tau) & \cos(2\omega_2\tau) & \dots & \cos((p-1)\omega_2\tau) \\ 0 & \sin(\omega_2\tau) & \sin(2\omega_2\tau) & \dots & \sin((p-1)\omega_2\tau) \end{bmatrix}.
$$

$$\tag{24}$$

The trajectories will reside on a 4-dimensional hyperplane given by the *constant* matrix $\mathbf{V}$ of delay-embedded harmonic functions. Picking τ and p too small complicates manifold identification, because the columns of $\mathbf{V}$ are close to linearly dependent. Instead, we choose a timelag such that these planes are as close to orthogonal as possible. Based on these considerations, we select a timelag $\tau = k\Delta t$ such that $\omega_2\tau \approx \frac{\pi}{2}$, where ω_2 is the second eigenfrequency of the observed system. This argument generalizes to any number of modes, although the choice of k becomes more complicated. The embedding dimension p is set to at least $2d + 1$. Further increases to p can facilitate identification of a manifold but also complicate its geometry far from the origin.

3.2 Backbone curves and forced response

If all eigenvalues of the reduced dynamics are complex conjugate pairs, we can rewrite the normal form on the SSM in polar coordinates $(\boldsymbol{\rho}, \boldsymbol{\theta})$ with $z_\ell = \rho_\ell e^{i\theta_\ell}$, $z_{\ell+1} = \rho_\ell e^{-i\theta_\ell}$. This yields the SSM-reduced dynamics in the form

$$
\begin{aligned}
\dot{\rho}_\ell &= c_\ell(\boldsymbol{\rho}, \boldsymbol{\theta})\rho_\ell, \\
\dot{\theta}_\ell &= \omega_\ell(\boldsymbol{\rho}, \boldsymbol{\theta}).
\end{aligned} \tag{25}
$$

If the imaginary parts of all eigenvalue pairs are nonresonant with each other, then the functions c_ℓ and ω_ℓ

depend only on $\boldsymbol{\rho}$ [43]. For example, the cubic 2D normal form (16) in polar coordinates reads

$$
\begin{aligned}
\dot{\rho} &= \mathrm{Re}\,(\lambda)\rho + \mathrm{Re}\,(\gamma)\rho^3 + \mathscr{O}(\rho^5), \\
\dot{\theta} &= \mathrm{Im}\,(\lambda) + \mathrm{Im}\,(\gamma)\rho^2 + \mathscr{O}(\rho^4).
\end{aligned}
\tag{26}
$$

For the general 2D case, we define an amplitude as

$$
\mathscr{A}(\rho) = \max_{\theta\in[0,2\pi)}\left|\alpha\left(\boldsymbol{Wt}\left(\begin{bmatrix}\rho\mathrm{e}^{i\theta} \\ \rho\mathrm{e}^{-i\theta}\end{bmatrix}\right)\right)\right|,
\tag{27}
$$

where $\alpha : \mathbb{R}^p \to \mathbb{R}$ is a function mapping from the observable space to some amplitude of a particular degree of freedom or the norm of total displacements. The backbone curve of that amplitude is then defined as the parametrized curve

$$
\mathscr{B}(\rho) = (\omega(\rho), \mathscr{A}(\rho)),
\tag{28}
$$

which is broadly used in the field of nonlinear vibrations to illustrate the overall effect of nonlinearities in the system.

Next, following Refs. [6,44], we use the data-driven normal form on the SSM to predict the response of the system under additional, time-periodic external forcing. This amounts to adding a forcing term with amplitude f and frequency Ω to the general 2D normal form to obtain

$$
\begin{aligned}
\dot{\rho} &= c(\rho)\rho + f\sin\psi, \\
\dot{\psi} &= \omega(\rho) - \Omega + \frac{f}{\rho}\cos\psi,
\end{aligned}
\tag{29}
$$

where we have introduced the phase difference $\psi = \theta - \Omega t$. The forced response curve (FRC) is then defined as the bifurcation curve of the fixed points (ρ_0, ψ_0) of (29) under varying Ω. Squaring and adding the equations in (29) yields a parametrization of the FRC for the forcing frequency Ω and the phase lag ψ_0 in the form

$$
\begin{aligned}
\Omega(\rho_0, f) &= \omega(\rho_0) \pm \sqrt{\frac{f^2}{\rho_0^2} - (c(\rho_0))^2}, \\
\psi_0(\rho_0, f) &= -\arcsin\left(\frac{c(\rho_0)\rho_0}{f}\right).
\end{aligned}
\tag{30}
$$

Since the relation between the experimental forcing and the normal form forcing f is unknown, a calibration of each FRC to at least one observation of a forced response is necessary. Following Refs. [10,11], we achieve this by prescribing a single intersection point on the FRC in the frequency-amplitude plane. We use the calibration frequency Ω_{cal} and amplitude u_{cal} of the maximal experimental response at each forcing level. The mapping to an equivalent point $\boldsymbol{y}_{\mathrm{cal}}$ in the observable space can then be approximated by delay-embedding a cosine signal of amplitude u_{cal} and frequency Ω_{cal}. We compute the calibration amplitude ρ_{cal} by projecting $\boldsymbol{y}_{\mathrm{cal}}$ onto the manifold and transforming to the normal form. The forcing f is then computed from the relationship

$$
f^2 = (c(\rho_{\mathrm{cal}}))^2\rho_{\mathrm{cal}}^2 + (\omega(\rho_{\mathrm{cal}}) - \Omega)^2\rho_{\mathrm{cal}}^2.
\tag{31}
$$

Finally, under periodic forcing, the SSM parametrization becomes time-dependent with the addition of a small periodic term. We ignore this contribution here to simplify the analysis, but note that this small non-autonomous correction can be exactly computed if needed [10,11].

4 Examples

We now apply the two variants of our simplified SSM computation to three datasets: sloshing experiments in a water tank, simulation of a clamped-clamped von Kármán beam, and experiments on an internally resonant beam.

We quantify the quality of our SSM-reduced models with the *normalized mean trajectory error* (NMTE) [10]. Given a test trajectory with N snapshots $\boldsymbol{y}(t_j)$ and the model-based reconstruction $\hat{\boldsymbol{y}}(t_j)$ obtained by integrating the normal form dynamics and mapping back to the observable space, the NMTE is defined as

$$
\mathrm{NMTE} = \frac{1}{N\max_j \|\boldsymbol{y}(t_j)\|}\sum_{j=1}^{N}\|\hat{\boldsymbol{y}}(t_j) - \boldsymbol{y}(t_j)\|.
\tag{32}
$$

4.1 Tank sloshing

A tank partially filled with a liquid responds nonlinearly to horizontal harmonic excitation [54]. Stronger fluid oscillation gives rise to more shearing against the tank wall, so that the damping of the system increases nonlinearly with the amplitude [19]. In addition, the instantaneous frequency decreases at higher amplitudes. Both phenomena are crucial for predicting the forced response amplitude of the liquid. The industrial applications for sloshing models are numerous, ranging from the transportation of fluids in trucks [13] and container ships [19] to the design of fuel tanks for spacecraft [2,16].

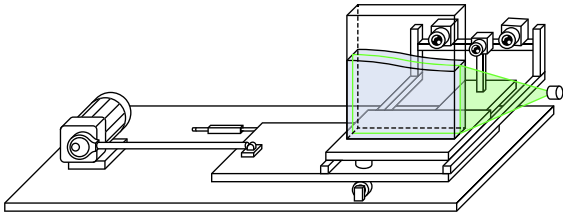In Ref. [10], SSMLearn was applied to experimental sloshing data from Ref. [8]. With nonlinear models

Fig. 2 Sloshing experiment setup with the partially filled tank and the cameras, mounted on a platform excited by a motor

of both frequency and damping, the forced response of the first oscillation mode was accurately predicted from unforced decaying center of mass signals. Here, we will apply the fastSSM algorithm to a much higher-dimensional observable space that includes fine-resolved surface profile data.

The experiments described in Ref. [8] were performed in a rectangular tank of width $w = 500$ mm and depth $l = 50$ mm, partially filled with water up to a height of 400 mm. The tank was mounted on a moving platform excited harmonically by a motor at different frequencies Ω and amplitudes A. A camera was mounted on the moving platform, and the surface profile $\boldsymbol{h}$ was detected via image processing with the sampling time $\Delta t = 0.033$ s. Figure 2 displays the experimental setup.

From the surface profile, the sloshing amplitude can be quantified by computing the horizontal position $\hat{X}$ of the liquid's center of mass at each time, normalized by the tank width. $\hat{X}$ is a physically meaningful quantity, relevant for engineering applications and robust against small image evaluation errors. The tank was excited at the tested frequencies until a steady state was reached, and the driving was turned off. The oscillation amplitude then decayed along the backbone curve defined in (28).

Here we append the $\hat{X}$ signal with the measurements of the surface elevation $\boldsymbol{h}$ at 1 531 points. We delay-embed these signals using 10 subsequent measurements to create a 15 320-dimensional observable space, in which a 2D, 7th-order SSM, shown in Fig. 3a, is identified. We fit the reduced dynamics on the SSM up to 3rd order, and then compute its 3rd-order normal form as

$$\dot{\rho} = -0.062\rho - 0.019\rho^3,$$
$$\rho\dot{\theta} = 7.81\rho - 0.628\rho^3. \tag{33}$$

The backbone curves obtained from this normal form are shown in Figure 3b.

In Fig. 3c and d, we compute forced response curves for the center of mass position, $\hat{X}$, and compare to its experimentally generated values obtained along frequency sweeps from both directions. We find the response prediction to be accurate. In particular, the nonlinear damping term of the normal form helps capturing the width of the forced response curve at higher amplitudes.

We also use the normal form (33) to predict the development of the decaying full surface profile $\boldsymbol{h}$ in Fig. 4. Here, we take the initial surface profile and transform it to an initial condition in the normal form coordinates. We integrate the normal form from this initial condition to predict its development in the observable space. We observe that the prediction is in close agreement with experiments, yielding a total NMTE of 2.05% over the entire phase space.

Finally, we compare execution times for fastSSM and SSMLearn when they are trained on the surface profile data. These computations were performed on MATLAB version 2020b, installed on an iMac with 2.3 GHz 18-Core Intel Xeon W and 128 GB RAM.

In Fig. 3e, the computational effort of fitting a 2D SSM is plotted against the dimensionality of the training data, i.e., the number of included surface points multiplied by the delay embedding dimension 10. Due to its explicit coefficient fitting, fastSSM achieves a major speedup, and enables analysis of significantly higher-dimensional data than SSMLearn. Since our proposed SSM fitting method is based on SVD and polynomial regression, we expect the computational effort to grow linearly with the input data dimensionality. This indeed holds approximately for the highest-dimensional training datasets in the present example.

After reduction to the SSM, both SSMLearn and fastSSM compute the $\mathcal{O}(3)$ normal form in less than a second. In order to compare the computational effort for higher-order normal forms, we apply fastSSM$^+$. Figure 3f shows the time required to compute a 2D normal form after fitting the SSM to the sloshing data. At a given order, fastSSM$^+$ is on average 15 times faster than SSMLearn. While both methods are fast in this example, the difference becomes significant at higher dimensions. It should be noted, however, that in strongly nonlinear cases, an analytical expansion may require a higher-order normal form to converge, and so the difference to the practitioner may be smaller. This difference in the convergence of normal forms is closer examined in the next example.
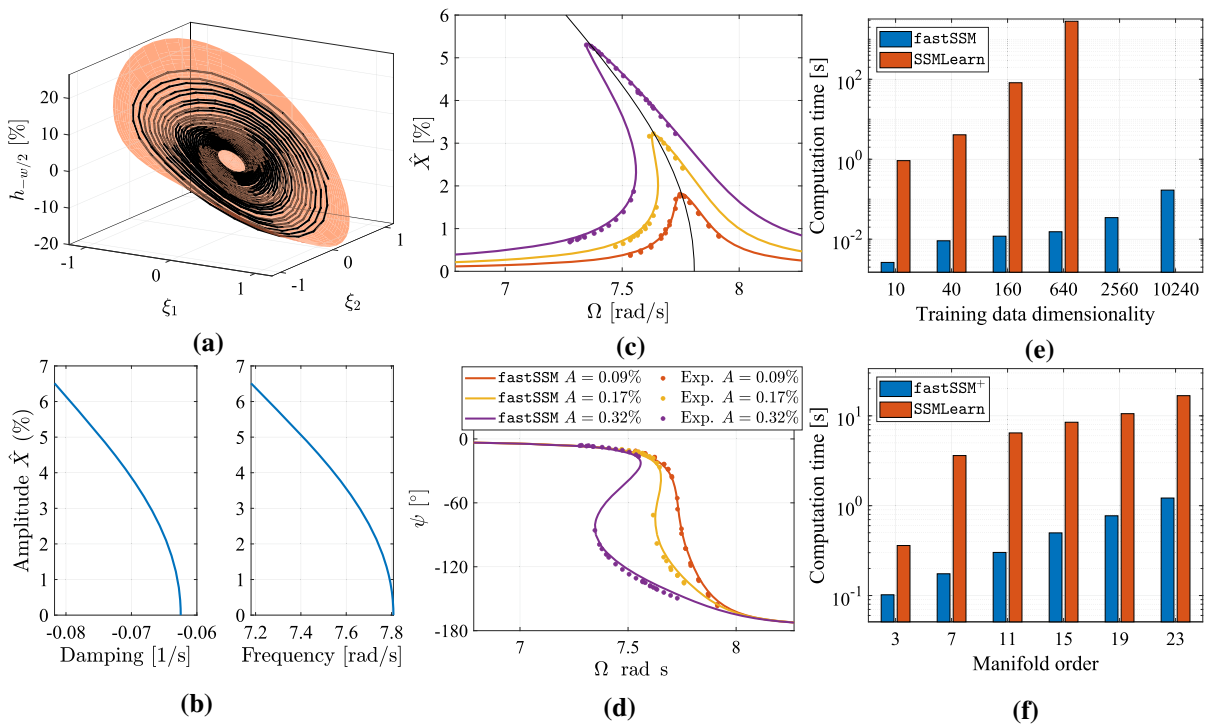
(a)

(b)

(c)

(d)

(e)

(f)

Fig. 3 **a** A visualization of the identified 2D SSM, with the wall surface amplitude on the vertical axis. **b** The computed instantaneous damping and frequency show significant nonlinearity. **c** Prediction of the forced response amplitude and **d** phase lag compared with experimental results for three different forcing amplitudes. **e** The computational cost of manifold fitting is orders of magnitude lower with fastSSM than with SSMLearn, and scales better with higher dimensionality. **f** fastSSM$^+$ normal form computation is also faster, although SSMLearn tends to require a lower order for convergence

4.2 von Kármán beam

We consider data from numerical simulations of a finite-element model of a clamped-clamped von Kármán nonlinear beam [27]. Each element has three degrees of freedom: axial deformation u_0, transverse deflection w_0, and rotation w_0'. The nonlinear von Kármán axial strain approximation is

$$\epsilon_{11} = u_0'(x_1) + \frac{1}{2}\left(w_0'(x_1)\right)^2 - zw_0''(x_1), \tag{34}$$

where the second term sets this model apart from the linear Euler–Bernoulli beam. The axial stress is modeled as

$$\sigma = E\epsilon_{11} + \kappa\dot{\epsilon}_{11}, \tag{35}$$

where E is the Young's modulus and κ is the material rate of viscous damping.

After a convergence analysis, we set the number of elements to 12, which results in a 33-degree of freedom mechanical system, i.e., a 66-dimensional phase space. As initial condition, we compute the response to a static transverse load of 14 kN at the midpoint, removed at the simulation start. Our observable function is the midpoint displacement, and the objective is to reconstruct the SSM and its dynamics in the observable space by delay-embedding the signal. A sketch of the system is shown in Fig. 5a.

We set $E = 70$ GPa, $\kappa = 1.0 \times 10^6$ Pa · s, length 1000 mm, width 50 mm, and thickness 20 mm. The sampling time is $\Delta t = 0.0955$ s. To satisfy the conditions of Takens' embedding theorem, we set the delay embedding dimension to $p = 5$. The maximum displacement in the training data is 15.9 mm.

A cubic normal form is insufficient to describe the higher amplitude oscillations, which prompts us to compute higher orders with SSMTool. We use a 1st-order SSM, 5th-order SSM-reduced dynamics, and obtain an 11th-order normal form
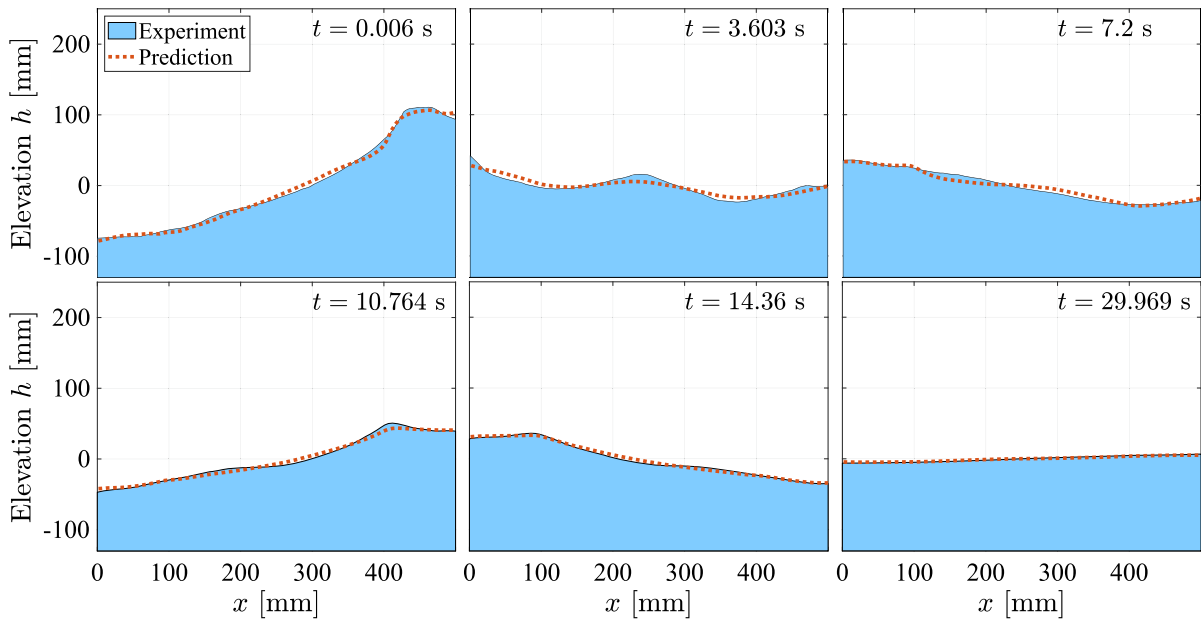
Fig. 4 Snapshots of the fastSSM prediction of the surface profile decay. The initial condition is transformed to the SSM-reduced system, which is integrated to produce a 2D trajectory in normal form coordinates. By mapping the trajectory back to the observ-able space, we can predict the full surface profile and observe close agreement with the experiments in phase, amplitude, and shape

$$\dot{\rho} = -3.12\rho - 0.899\rho^3 + 4.12\rho^5$$
$$-12.4\rho^7 + 16.0\rho^9 - 17.7\rho^{11}, \tag{36a}$$

$$\rho\dot{\theta} = 658\rho + 263\rho^3 - 155\rho^5$$
$$+151\rho^7 - 181\rho^9 + 237\rho^{11}, \tag{36b}$$

which yields a NMTE of 0.63% on the training data (Fig. 5b).

We compute the FRC and verify it against a numerical integration frequency sweep in Fig. 5c. Clearly, the autonomous SSM obtained from our new fastSSM$^+$ algorithm predicts the forced response very accurately even with strong nonlinearities. Figure 5d shows a representation of the SSM geometry, which, as predicted in Sect. 3.1, is almost flat due to the small timelag.

There is, however, a limit to the range of validity of the normal form (36), and so the forced response prediction is not guaranteed to be accurate for arbitrarily high amplitudes. To explore this limitation, we increase the initial point load to 35 kN, equivalent to a maximum displacement of 24.4 mm in the training data. We plot backbone curves computed numerically with SSMLearn and analytically at increasing orders with fastSSM$^+$ in Fig. 5e. For reference, we also compute an approximation of the instantaneous frequency with Peak Finding and Fitting (PFF) from Ref. [28].

Above approximately 20 mm transverse displacement of the beam midpoint, increasing the order of the normal form computation no longer improves the model, as the radius of analyticity seems to be reached. On the other hand, the SSMLearn model remains valid far beyond this limit. This is a clear advantage of the numerical normal form approach over analytical computations.

Taking this observation further, a similar convergence study is shown in Fig. 5f for increasing orders of SSMTool computation on the full finite-element system of equations. The SSMLearn backbone curve agrees with the PFF estimate at a larger range than the analytical normal form converges. This shows that data-driven reduced-order modeling can surpass analytical methods in terms of range of validity, even when the full system is known.

4.3 Resonant beam experiments

Our final dataset comprises velocity measurements from an internally resonant beam structure detailed in Ref. [11]. It consists of an internal beam jointed with three bolts to the left midpoint of an external C-shaped beam, which is clamped to ground at its rightmost edges (Fig. 6a). The bolts give rise to nonlinear frictional slip
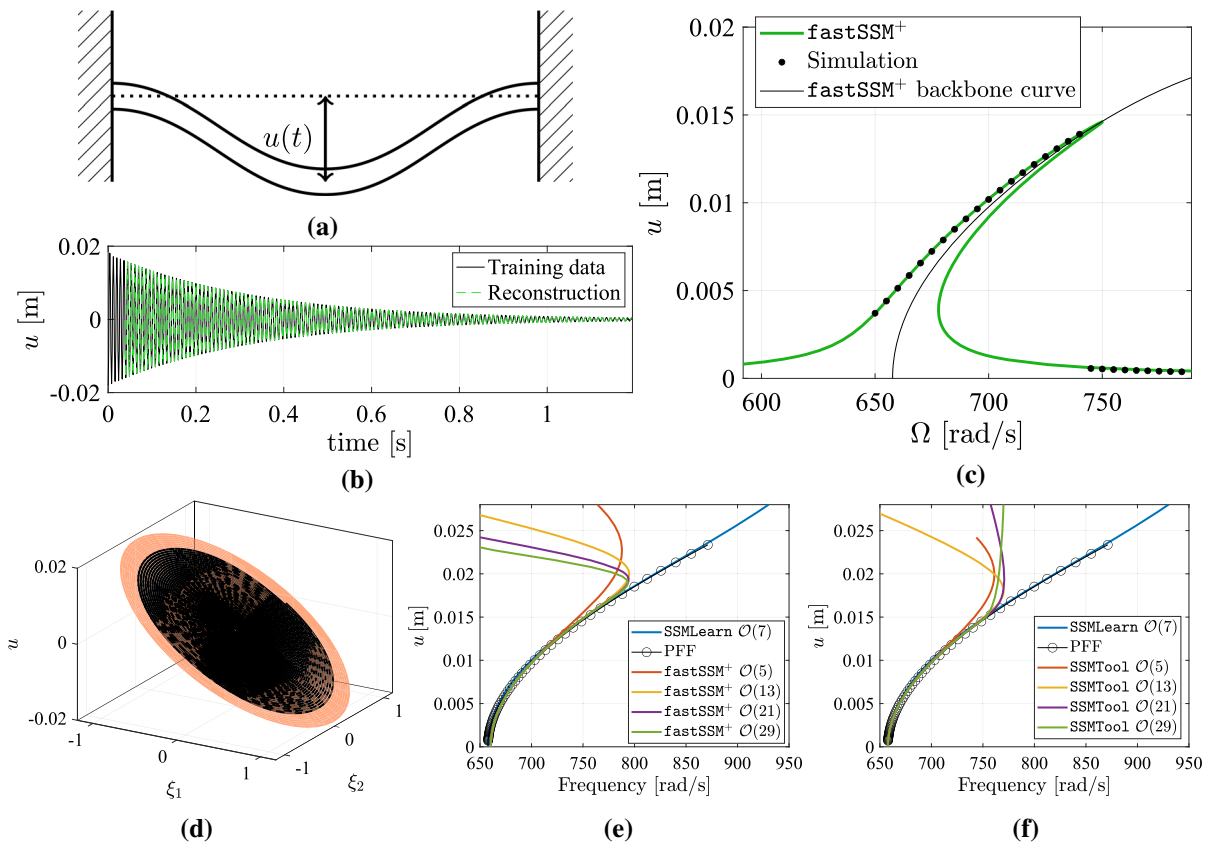
(a)

(b)

(c)

(d)

(e)

(f)

Fig. 5 **a** The observable is the midpoint displacement $u(t)$ of a 12-element von Kármán clamped-clamped beam. **b** Based on an $\mathcal{O}(5)$ reduced dynamics model, we compute using the fastSSM$^+$ algorithm a normal form that predicts the displacement decay accurately. **c** Prediction of the forced response amplitude. **d** The 2D SSM visualized in a 3D representation of the observable space. Above 20 mm deflection, increased orders of **e** fastSSM$^+$ and **f** SSMTool no longer improve the backbone curve approximation. Higher amplitudes lie beyond the radius of convergence of the SSM Taylor expansion. SSMLearn, however, continues to approximate the backbone curve well beyond this limit

[5,18], and the beam lengths are tuned so that the system has a 1:2 resonance between its slowest out-of-plane bending eigenfrequencies, measured to be 122.4 Hz and 243.4 Hz. Vibrations are induced with an impulse hammer at three different impact locations and the transient out-of-plane velocity of the inner beam tip is measured at 5120 Hz. Varying the impact location and force for data diversity, 12 different trajectories were obtained. Frequency analysis shows that only the two slowest eigenfrequencies are present in the signals, so we aim to identify a 4D internally resonant SSM in an observable space for our model reduction.

With more than one mode present in the data, we must judiciously choose the delay embedding parameters. In the experiment, the sampling time is $\Delta t = 0.0001953$ s, which with $\omega_2 = 2\pi \times 243.4$ rad/s pro-

duces $\tau = 5\Delta t$ according to the method in Sect. 3.1. This timelag results in a good reconstruction already at the minimum Takens dimension, $p = 9$. However, increasing the dimensionality further increases modal orthogonality and thus improves trajectory reconstruction. Motivated by this, we select a delay embedding dimension of $p = 12$, as the reconstructions improve only marginally beyond this number. We show a representation of the test trajectories in this observable space in Fig. 6b.

We select three trajectories as the test set—one for each hammer impact location—and use the remaining nine trajectories for training. We set the order of expansion for the SSM $m = 3$, the SSM-reduced dynamics order $r = 3$, and obtain the 3rd-order normal form. The results do depend on the order of reduced dynamics,

(a)

(b)
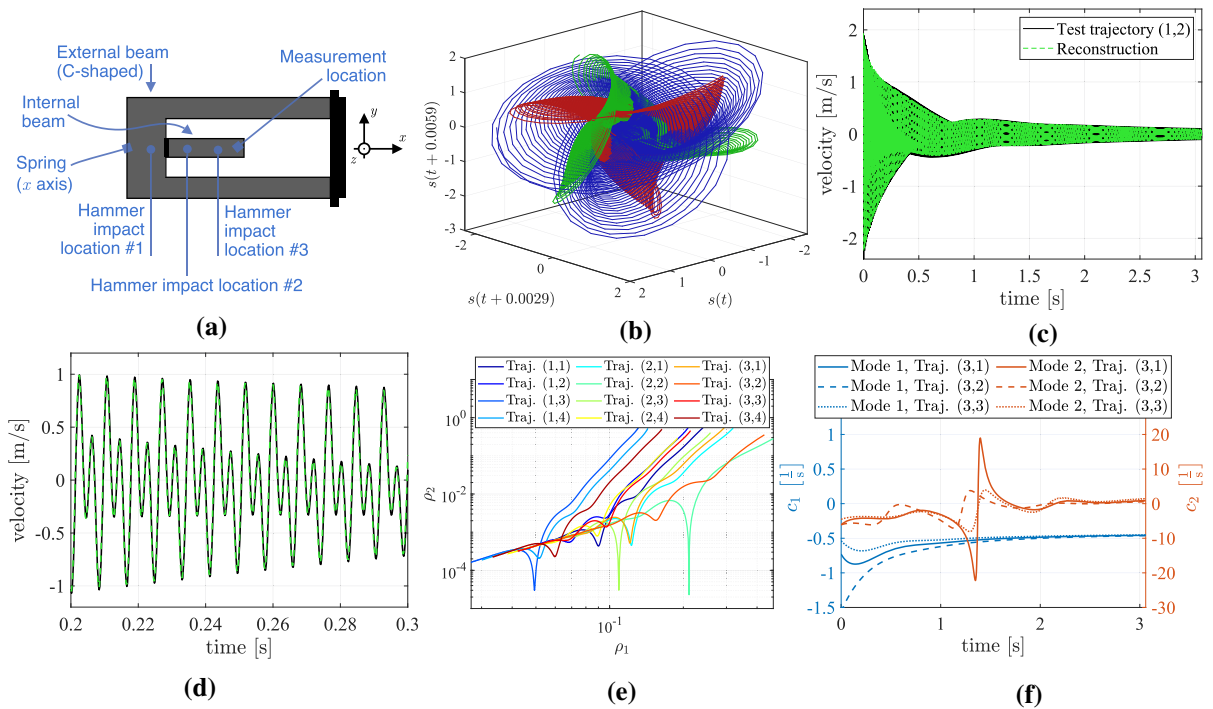

(c)

(d)

(e)

(f)

Fig. 6 **a** Sketch of the experimental setup for the resonant beam tester [11]. **b** The test trajectories embedded in the delay-embedded observable space, in which we fit the SSM. **c, d** The reconstruction from the obtained 4D normal form dynamics agrees with the decay of measured velocities. **e** Plotting the modal amplitudes ρ against one another shows that ρ_2 dies out more quickly than ρ_1, but, due to modal interaction, the decay rates vary. **f** The instantaneous damping of the second mode is at times positive, indicating modal energy interchange

but are insensitive to the orders of computation for the SSM and the normal form on it. Our fastSSM$^+$ algorithm detects an inner resonance from the data and we obtain the normal form

$$\dot{\rho}_1 = -0.4426\rho_1 + 0.0837\rho_1\rho_2^2 - 5.550\rho_1^3$$
$$+ (0.2683\cos\psi - 0.2639\sin\psi)\,\rho_1\rho_2, \quad (37a)$$

$$\rho_1\dot{\theta}_1 = 768.9\rho_1 - 0.7541\rho_1\rho_2^2 - 17.90\rho_1^3$$
$$+ (-0.2639\cos\psi - 0.2683\sin\psi)\,\rho_1\rho_2, \quad (37b)$$

$$\dot{\rho}_2 = -3.125\rho_2 - 4.750\rho_2^3 - 12.00\rho_1^2\rho_2$$
$$+ (1.340\cos\psi - 0.1161\sin\psi)\,\rho_1^2, \quad (37c)$$

$$\rho_2\dot{\theta}_2 = 1529\rho_2 - 10.54\rho_2^3 - 13.86\rho_1^2\rho_2$$
$$+ (0.1161\cos\psi + 1.340\sin\psi)\,\rho_1^2, \quad (37d)$$

where the automatically fitted trigonometric terms involving $\psi = 2\theta_1 - \theta_2$ account for the phase-dependence induced by the internal resonance. The reconstruction of the first test trajectory obtained by integrating the normal form is shown in Fig. 6c, with

a zoomed-in version in Fig. 6d. The NMTE on the test set computed in the first coordinate is 1.28%.

Figure 6e plots the first modal amplitude, ρ_1, against the second one, ρ_2, for each simulated trajectory. The second mode clearly decays faster and trajectories are eventually dominated by the slower mode ρ_1. However, due to the modal coupling terms in the normal form, the decay is not monotonic. Rather, there is a wealth of different decaying patterns depending on initial conditions.

Finally, Fig. 6f shows the instantaneous damping $\dot{\rho}_1/\rho_1$ and $\dot{\rho}_2/\rho_2$ for the reconstructions corresponding to the third impact location. For the slow mode, we observe strong nonlinearity as the instantaneous damping varies dramatically. The fluctuations are even larger for the fast mode. At times, its instantaneous damping reaches positive values, which indicates that energy is transferred from the slow to the fast mode, as previously observed in Refs. [4,46]. Overall, the model obtained with our fastSSM$^+$ algorithm agrees with the one returned by SSMLearn in Ref. [11].

5 Conclusions

We have introduced a fast alternative to a recent data-driven reduction method for nonlinear dynamical systems. Our approach is inspired by the SSMLearn method [10, 11], but formulates the fitting of an SSM as an explicit problem by assuming that its tangent space can be obtained by singular value decomposition on the data. Furthermore, we compute the normal form on the manifold explicitly and recursively, rather than solving an implicit minimization problem.

Our approach is based on rigorous existence and uniqueness results for SSMs constructed near hyperbolic fixed points of smooth dynamical systems. In addition, the spectral subspace over which the SSM is constructed may contain either stable directions or unstable directions but not a mixture of these. These assumptions formally exclude applications of our methods near non-hyperbolic fixed points, over spectral subspaces of mixed stability type and to systems with non-smooth features such as hysteretic phenomena and impacts. Our ongoing work seeks to extend SSM theory in a way that overcomes these limitations.

We have applied our alternative model-reduction method to three datasets: tank sloshing experiments, a nonlinear beam finite-element simulation, and experiments from an internally resonant mechanical structure. In all three problems, we obtained a model that accurately predicted the decay of the autonomous system. In addition, we have demonstrated that a forcing term can be added to the autonomous model for highly accurate prediction of the forced response amplitude and phase. Training on the beam experimental data, the internal resonance was automatically detected and our method generated a model that could predict energy repartition among the modes.

The assumptions made for our new method drastically speed up model identification on SSMs from data, increase the possible dimensionality of observable spaces we can tackle, and significantly simplify the numerical implementation. In comparison with the previous method, however, we sacrifice some model accuracy. Perhaps more significantly, we have found large differences in normal form convergence domains to the benefit of SSMLearn.

Specifically, we demonstrate on our simulated beam problem that the numerical normal form has a considerably larger range of validity than the analytical normal forms of the full system and consequently those of our new data-driven SSM construction. In other words, data-driven analysis can outperform analytical methods in terms of model validity, even when the full equations of the system are known. This suggests that a plausible approach to obtaining a reduced-order model of a finite-element structure would be to simulate the system and train on a small number of trajectories, rather than formulating the full equation system and computing SSMs analytically. This idea is a subject of our ongoing work.

Data availability The data supporting the findings of this work can be openly accessed at www.github.com/haller-group/SSMLearn/tree/main/fastSSM.

Declarations

A Code

A.1 fastSSM

The fastSSM code requires no external packages and can be applied to data out-of-the-box. This implementation is limited to finding 2D SSMs and computing

their 3rd order normal form. For strong nonlinearities or higher-dimensional SSMs, we recommend installing SSMTool and running fastSSM$^+$. fastSSM is also available at www.github.com/haller-group/SSMLearn/tree/main/fastSSM.

```matlab
function [Mmap, iMmap, Tmap, iTmap, Nflow, Yrec] = fastSSM(t, Y, mfdorder)
% [Mmap, iMmap, Tmap, iTmap, Nflow, Yrec] = fastSSM(t, Y, mfdorder)
% Computes a 2D SSM and O(3) normal form from data. By Joar Axas (jgoeransson@ethz.ch)
%
% INPUT:                          OUTPUT:
% t     (1xn) time values         Mmap   function SSM parametrization y=M(xi)^{1:m}
% Y     (pxn) snapshots at times t iMmap  function SSM chart
% mfdorder int SSM polynomial order Tmap   function normal form transformation xi=T(z)
%                                  iTmap  function inverse normal form transformation
%                                  Nflow  function normal form dynamics zdot=N(z)
%                                  Yrec   (pxn)   fastSSM reconstruction of Y

%% Fit manifold
[u,s,v] = svds(Y, 2);
V = (s\u'./max(abs(v'),[],2))'; % tangent space
iMmap = @(y) V'*y;
Xi = iMmap(Y);
Mcoeff = Y/phi(Xi, 1:mfdorder);
Mmap = @(xi) Mcoeff*phi(xi, 1:mfdorder);

%% Compute finite differences and reduced dynamics
idx = 5:(size(Xi,2)-4);
dXi = 4/5*(Xi(:,idx+1)-Xi(:,idx-1)) -1/5*(Xi(:,idx+2)-Xi(:,idx-2)) + ...
   4/105*(Xi(:,idx+3)-Xi(:,idx-3)) -1/280*(Xi(:,idx+4)-Xi(:,idx-4));
Xidot = dXi./(t(idx+1)-t(idx));

R = Xidot/phi(Xi(:,idx), 1:3); % reduced dynamics
[W,Lambda] = eig(R(1:2,1:2));
G = W\Xidot/phi(W\Xi(:,idx), 1:3); % diagonalize R

%% Compute transformation to normal form
ll = Lambda(1,1); % lambda_l
lc = conj(ll);    % lambda_l+1
lj = [ll; lc];    % lambda_j
T1 = eye(2);
T2 = [G(:,3)./(2*ll-lj), G(:,4)./(ll+lc-lj), G(:,5)./(2*lc-lj)];
T3 = horzcat(([2,1].*G(:,3:4)*T2(:,1) + G(:,6))./(3*ll-lj), ...
   [0;1].*([2,1].*G(:,3:4)*T2(:,2) + [1,2].*G(:,4:5)*T2(:,1) + G(:,7))./(2*ll+lc-lj), ...
   [1;0].*([2,1].*G(:,3:4)*T2(:,3) + [1,2].*G(:,4:5)*T2(:,2) + G(:,8))./(ll+2*lc-lj), ...
   ([1,2].*G(:,4:5)*T2(:,3) + G(:,9))./(3*lc-lj));
T = [T1, T2, T3];
Tmap = @(z) real(W*T*phi([z;conj(z)], 1:3));

gamma = [2,1].*G(1,3:4)*T2(:,2) + [1,2].*G(1,4:5)*T2(:,1) + G(1,7);
Nflow = @(t,z) ll*z + gamma*z.^2.*conj(z);

%% Approximate inverse to the normal form transformation
iT1 = [1,0];
iT2 = -T2(1,:);
iT3 = -T3(1,:) + horzcat(2*T(1,3).^2 + T(1,4)*conj(T(1,5)), ...
   3*T(1,3)*T(1,4) + T(1,4)*conj(T(1,4)) + 2*T(1,5)*conj(T(1,5)), ...
   2*T(1,3)*T(1,5) + T(1,4)*conj(T(1,3)) + T(1,4)*T(1,4) + 2*T(1,5)*conj(T(1,4)), ...
   T(1,4)*T(1,5) + 2*T(1,5)*conj(T(1,3)));
iT = [iT1, iT2, iT3];
iTmap = @(xi) iT*phi(W\xi, 1:3);

%% Evaluate model
fprintf('\dot{\rho} = %6.3f\rho %+6.3f\rho^3\n', real(ll), real(gamma))
fprintf('\dot{\theta} = %6.3f %+6.3f\rho^2\n', imag(ll), imag(gamma))
[~, Zrec] = ode45(Nflow, t, iTmap(Xi(:,1))), odeset('RelTol', 1e-6));
Yrec = Mmap(Tmap(Zrec.'));
plot(t, Y(1,:), '-', t, Yrec(1,:), '--', 'linewidth', 1.5);
xlabel('t'); ylabel('Y_1'); legend({'data','reconstruction'})
end

%% Subfunction: multivariate polynomial
function Sigma=exponents(d, k)
   B = repmat({0:max(k)}, 1, d);
   A = combvec(B{:}).';
   Sigma = A(ismember(sum(A, 2), k),:);
   [~,ind] = sort(sum(Sigma, 2));
   Sigma = Sigma(ind,:);
end
function u = phi(xi, r) % return monomials
   x = reshape(xi, 1, size(xi, 1), []);
   exps = exponents(size(xi, 1), r);
   u = reshape(prod(x.^exps, 2), size(exps, 1), []);
end
```

A.2 fastSSM$^+$

This implementation applies SSMTool to extend normal form computations to arbitrary dimension and order. The fastSSM$^+$ code is also available at www.github.

com/haller-group/SSMLearn/tree/main/fastSSM. Requires installation of SSMTool [26].

```matlab
function [Mmap, iMmap, Tmap, iTmap, Nflow, Yrec] = fastSSMplus(t, Y, mfddim, mfdorder, ...
   romorder, nforder)
% [Mmap, iMmap, Tmap, iTmap, Nflow, Yrec] = fastSSMplus(
%                    t, Y, mfddim, mfdorder, romorder, nforder)
% Computes an SSM and normal form from data with SSMTool. By Joar Axas (jgoeransson@ethz.ch)
%
% INPUT:                          OUTPUT:
% t      (1xn) time values         Mmap   function SSM parametrization y=M(xi)^{1:m}
% Y      (pxn) snapshots at times t iMmap  function SSM chart
% mfddim  int   SSM dimension       Tmap   function normal form transformation xi=T(z)
% mfdorder int  SSM polynomial order iTmap  function inverse normal form transformation
% romorder int  reduced dynamics order Nflow function normal form dynamics zdot=N(z)
% nforder int   normal form order     Yrec   (pxn)  fastSSMplus reconstruction of Y

%% Fit manifold
[u,s,v] = svds(Y, mfddim);
V = (s\u'./max(abs(v'),[],2))'; % tangent space
iMmap = @(y) V'*y;
Xi = iMmap(Y);
Mcoeff = Y/phi(Xi, 1:mfdorder);
Mmap = @(xi) Mcoeff*phi(xi, 1:mfdorder);

%% Compute finite differences and reduced dynamics
idx = 5:(size(Xi,2)-4);
dXi = 4/5*(Xi(:,idx+1)-Xi(:,idx-1)) -1/5*(Xi(:,idx+2)-Xi(:,idx-2)) + ...
   4/105*(Xi(:,idx+3)-Xi(:,idx-3)) -1/280*(Xi(:,idx+4)-Xi(:,idx-4));
Xidot = dXi./(t(idx+1)-t(idx));
R = Xidot/phi(Xi(:,idx), 1:romorder);

%% Rearrange coefficients into tensors for SSMTool
Rcol = 1;
for order = 1:romorder
   exps = exponents(mfddim, order);
   fnl{order}.ind = exps;
   fnl{order}.coeffs = R(:,Rcol:Rcol+size(exps,1)-1);
   Rcol = Rcol + size(exps, 1);
   fnl{order} = multi_index_to_tensor(fnl{order}.coeffs,fnl{order}.ind);
end

%% Compute normal form with SSMTool
DS = DynamicalSystem();
set(DS.Options, 'Emax', mfddim, 'Nmax', 100, 'notation', 'multiindex')
set(DS, 'A', R(1:mfddim,1:mfddim), 'B', eye(mfddim), 'fnl', fnl(2:end));
S = SSM(DS);
set(S.Options, 'reltol', 0.1, 'notation', 'multiindex');
S.choose_E(1:mfddim)
[Tcoeff, Ncoeff] = S.compute_whisker(nforder);
Tmap = @(z) real(tpoly(Tcoeff, z));
Nflow = @(t,z) tpoly(Ncoeff, z);

%% Approximate inverse to the normal form transformation
[W, ~] = eig(R(1:mfddim,1:mfddim));
invpoints = W\Xi;
iT = invpoints/phi(Tmap(invpoints),1:nforder);
iTmap = @(xi) iT*phi(xi, 1:nforder);

%% Evaluate model
options = struct('isauto', 1, 'isdamped', 1, 'numDigits', 3);
symexp = reduced_dynamics_symbolic(DS.spectrum.Lambda(1:mfddim), Ncoeff, options);
sympref('FloatingPointOutput',true);
fprintf('\dot{\rho}_%u =\n\\rho_{%u}\dot{\theta_%u} =\n', ceil(0.1:1/3:mfddim/2))
disp(symexp)

[~, Zrec] = ode45(Nflow, t, iTmap(Xi(:,1))), odeset('RelTol', 1e-6));
Yrec = Mmap(Tmap(Zrec.'));
plot(t, Y(1,:), '-', t, Yrec(1,:), '--', 'linewidth', 1.5);
xlabel('t'); ylabel('Y_1'); legend({'data','reconstruction'})
end

%% Subfunction: multivariate polynomial
function Sigma=exponents(d, k)
   B = repmat({0:max(k)}, 1, d);
   A = combvec(B{:}).';
   Sigma = A(ismember(sum(A, 2), k),:);
   [~,ind] = sort(sum(Sigma, 2));
   Sigma = Sigma(ind,:);
end
function u = phi(xi, r) % return monomials
   x = reshape(xi, 1, size(xi, 1), []);
   exps = exponents(size(xi, 1), r);
   u = reshape(prod(x.^exps, 2), size(exps, 1), []);
end

%% Subfunction: Express tensor as polynomial function
function xi = tpoly(T, z)
   xi = zeros(size(T{1}.coeffs,1),size(z, 2));
   z = reshape(z, 1, size(z, 1), []);
   for o = 1:numel(T)
      if ~isempty(T{o}.ind)
         xi = xi + T{o}.coeffs*reshape(prod(z.^T{o}.ind,2), size(T{o}.ind,1), []);
      end
   end
end
```

References

1. Abramian, A., Virot, E., Lozano, E., Rubinstein, S., Schneider, T.: Nondestructive prediction of the buckling load of imperfect shells. Phys. Rev. Lett. **125**, 225504 (2020)

2. Abramson, H. (ed.): The dynamic behavior of liquids in moving containers: with applications to space vehicle technology. In: Norman Abramson, H. (eds) NASA SP-106. Scientific and Technical Information Division. National Aeronautics and Space Administration, Washington, DC (1966)

3. Awrejcewicz, J., Krys'ko, V.A., Vakakis, A.F.: Order Reduction by Proper Orthogonal Decomposition (POD) Analysis, pp. 279–320. Springer, Berlin (2004)

4. Balachandran, B., Nayfeh, A., Pappa, R., Smith, S.: Identification of nonlinear interactions in structures. J. Guid. Control Dyn. **17**, 257–262 (1994). https://doi.org/10.2514/3.21191

5. Brake, M.: The Mechanics of Jointed Structures: Recent Research and Open Challenges for Developing Predictive Models for Structural Dynamics. Springer International Publishing, Berlin (2018)

6. Breunung, T., Haller, G.: Explicit backbone curves from spectral submanifolds of forced-damped nonlinear mechanical systems. Proc. R. Soc. A **474**, 20180083 (2018)

7. Brunton, S.L., Proctor, J.L., Kutz, J.N.: Discovering governing equations from data by sparse identification of nonlinear dynamical systems. Proc. Natl. Acad. Sci. **113**(15), 3932–3937 (2016)

8. Bäuerlein, B., Avila, K.: Phase lag predicts nonlinear response maxima in liquid-sloshing experiments. J. Fluid Mech. **925**, A22 (2021). https://doi.org/10.1017/jfm.2021.576

9. Cabré, X., Fontich, E., de la Llave, R.: The parameterization method for invariant manifolds I: manifolds associated to non-resonant subspaces. Indiana Univ. Math. J. **52**(2), 283–328 (2003)

10. Cenedese, M., Axås, J., Bäuerlein, B., Avila, K., Haller, G.: Data-driven modeling and prediction of non-linearizable dynamics via spectral submanifolds. Nat. Commun. **13**(1), 1–13 (2022). https://doi.org/10.1038/s41467-022-28518-y

11. Cenedese, M., Axås, J., Yang, H., Eriten, M., Haller, G.: Data-driven nonlinear model reduction to spectral submanifolds in mechanical systems. Philos. Trans. R. Soc. A Math. Phys. Eng. Sci. **380**(229), 20210194 (2022). https://doi.org/10.1098/rsta.2021.0194

12. Cenedese, M., Axås, J., Haller, G.: SSMLearn (2021). http://www.georgehaller.com

13. Cheli, F., D'Alessandro, V., Premoli, A., Sabbioni, E.: Simulation of sloshing in tank trucks. Int. J. Heavy Veh. Syst. **20**, 1–16 (2013). https://doi.org/10.1504/IJHVS.2013.051099

14. Daniel, T., Casenave, F., Akkari, N., Ryckelynck, D.: Model order reduction assisted by deep neural networks (rom-net). Adv. Model. Simul. Eng. Sci. **7**, 105786 (2020)

15. Deyle, E.R., Sugihara, G.: Generalized theorems for nonlinear state space reconstruction. PLOS ONE **6**(3), 18295 (2011). https://doi.org/10.1371/journal.pone.0018295

16. Dodge, F.: The New Dynamic Behavior of Liquids in Moving Containers. Southwest Research Institute, San Antonio (2000)

17. Dylewsky, D., Kaiser, E., Brunton, S.L., Kutz, J.N.: Principal component trajectories for modeling spectrally continuous dynamics as forced linear systems. Phys. Rev. E **105**, 015312 (2022). https://doi.org/10.1103/PhysRevE.105.015312

18. Eriten, M., Kurt, M., Luo, G., McFarland, D., Bergman, L., Vakakis, A.: Nonlinear system identification of frictional effects in a beam with a bolted joint connection. Mech. Syst. Signal Process. **39**(1), 245–264 (2013). https://doi.org/10.1016/j.ymssp.2013.03.003

19. Faltinsen, O., Timokha, A.: Sloshing. Cambridge University Press, Cambridge (2009)

20. Fornberg, B.: Generation of finite difference formulas on arbitrarily spaced grids. Math. Comput. **51**(184), 699–706 (1988). https://doi.org/10.1090/s0025-5718-1988-0935077-0

21. Guckenheimer, J., Holmes, P.: Nonlinear oscillations. In: Dynamical Systems and Bifurcation of Vector Fields. Springer, New York (1983)

22. Haller, G., Ponsioen, S.: Nonlinear normal modes and spectral submanifolds: existence, uniqueness and use in model reduction. Nonlinear Dyn. **86**(3), 1493–1534 (2016)

23. Hartman, D., Mestha, L.K.: A deep learning framework for model reduction of dynamical systems. In: 2017 IEEE Conference on Control Technology and Applications (CCTA), pp. 1917–1922 (2017)

24. Holmes, P.J., Lumley, J.L., Berkooz, G., Rowley, C.W.: Turbulence, coherent structures, dynamical systems and symmetry. In: Cambridge Monographs on Mechanics, 2nd edn. Cambridge University Press (2012)

25. Jain, S., Haller, G.: How to compute invariant manifolds and their reduced dynamics in high-dimensional finite element models. Nonlinear Dyn. **107**(2), 1417–1450 (2021). https://doi.org/10.1007/s11071-021-06957-4

26. Jain, S., Thurner, T., Li, M., Haller, G.: SSMTool: Computation of invariant manifolds and their reduced dynamics in high-dimensional mechanics problems. (2021). https://doi.org/10.5281/zenodo.4614201. http://www.georgehaller.com

27. Jain, S., Tiso, P., Haller, G.: Exact nonlinear model reduction for a von Kármán beam: slow-fast decomposition and spectral submanifolds. J. Sound Vib. **423**, 195–211 (2018)

28. Jin, M., Chen, W., Brake, M.R.W., Song, H.: Identification of instantaneous frequency and damping from transient decay data. J. Vib. Acoust. **142**(5), 051111 (2020). https://doi.org/10.1115/1.4047416

29. Kaszás, B., Cenedese, M., Haller, G.: Dynamics-based machine learning of transitions in couette flow. Phys. Rev. Fluids (2022) (In press)

30. Kerschen, G., Peeters, M., Golinval, J., Vakakis, A.: Nonlinear normal modes, Part I: A useful framework for the structural dynamicist. Mech. Syst. Signal Process. **23**(1), 170–194 (2009). https://doi.org/10.1016/j.ymssp.2008.04.002

31. Kutz, J.N., Brunton, S.L.: Parsimony as the ultimate regularizer for physics-informed machine learning. Nonlinear Dyn. **107**(3), 1801–1817 (2022). https://doi.org/10.1007/s11071-021-07118-3

32. Kutz, J.N., Brunton, S.L., Brunton, B.W., Proctor, J.L.: Dynamic Mode Decomposition. SIAM, Philadelphia, PA (2016)

33. Lacayo, R., Pesaresi, L., Groß, J., Fochler, D., Armand, J., Salles, L., Schwingshackl, C., Allen, M., Brake, M.: Nonlinear modeling of structures with bolted joints: a comparison of two approaches based on a time-domain and frequency-

domain solver. Mech. Syst. Signal Process. **114**, 413–438 (2019). https://doi.org/10.1016/j.ymssp.2018.05.033

34. Li, M., Haller, G.: Nonlinear analysis of forced mechanical systems with internal resonance using spectral submanifolds—Part II: Bifurcation and quasi-periodic response. Nonlinear Dyn. (2022). https://doi.org/10.1007/s11071-022-07476-6

35. Li, M., Jain, S., Haller, G.: Nonlinear analysis of forced mechanical systems with internal resonance using spectral submanifolds—Part I: Periodic response and forced response curve. Nonlinear Dyn. (2022)

36. de la Llave, R., Kogelbauer, F.: Global persistence of Lyapunov subcenter manifolds as spectral submanifolds under dissipative perturbations. SIAM J. Appl. Dyn. Sys. **18**(4), 2099–2142 (2019)

37. Loiseau, J.C., Brunton, S.L., Noack, B.R.: From the POD-Galerkin Method to Sparse Manifold Models, pp. 279–320. De Gruyter, Berlin (2020)

38. Lumley, J.L.: The structure of inhomogeneous turbulent flows. In: Atmospheric Turbulence and Radio Wave Propagation, pp. 166–177 (1967)

39. Lusch, B., Kutz, J.N., Brunton, S.L.: Deep learning for universal linear embeddings of nonlinear dynamics. Nature Commun. **9**(4950), 1–10 (2018)

40. Orosz, G., Stépán, G.: Subcritical Hopf bifurcations in a car-following model with reaction-time delay. Proc. R. Soc. A **462**(2073), 2643–2670 (2006)

41. Page, J., Kerswell, R.: Koopman mode expansions between simple invariant solutions. J. Fluid Mech. **879**, 1–27 (2019)

42. Ponsioen, S., Jain, S., Haller, G.: Model reduction to spectral submanifolds and forced-response calculation in high-dimensional mechanical systems. J. Sound Vib. **488**, 115640 (2020)

43. Ponsioen, S., Pedergnana, T., Haller, G.: Automated computation of autonomous spectral submanifolds for nonlinear modal analysis. J. Sound Vib. **420**, 269–295 (2018)

44. Ponsioen, S., Pedergnana, T., Haller, G.: Analytic prediction of isolated forced response curves from spectral submanifolds. Nonlinear Dyn. **98**, 2755–2773 (2019)

45. Rosenberg, R.M.: The normal modes of nonlinear n-degree-of-freedom systems. J. Appl. Mech. **29**(1), 7–14 (1962). https://doi.org/10.1115/1.3636501

46. Sapsis, T., Quinn, D., Vakakis, A., Bergman, L.: Effective stiffening and damping enhancement of structures with strongly nonlinear local attachments. J. Vib. Acoust. Stress Reliab. Des. **134**(1), 011016 (2012). https://doi.org/10.1115/1.4005005

47. Sauer, T., Yorke, J., Casdagli, M.: Embedology. J. Stat. Phys. **65**, 579–616 (1991)

48. Schmid, P.: Dynamic mode decomposition of numerical and experimental data. J. Fluid Mech. **656**, 5–28 (2010)

49. Schmid, P.J.: Dynamic mode decomposition and its variants. Annu. Rev. Fluid Mech. **54**(1), 225–254 (2022). https://doi.org/10.1146/annurev-fluid-030121-015835

50. Shaw, S., Pierre, C.: Normal modes for non-linear vibratory systems. J. Sound Vib. **164**(1), 85–124 (1993)

51. Szalai, R.: Invariant spectral foliations with applications to model order reduction and synthesis. Nonlinear Dyn. **101**, 2645–2669 (2020)

52. Szalai, R., Ehrhardt, D., Haller, G.: Nonlinear model identification and spectral submanifolds for multi-degree-of-freedom mechanical vibrations. Proc. R. Soc. A **473**(2202), 20160759 (2017)

53. Takens, F.: Detecting strange attractors in turbulence. In: Rand, D., Young, L. (eds.) Dynamical Systems and Turbulence, Warwick, 1980, pp. 366–381. Springer, Berlin (1981)

54. Taylor, G.: An experimental study of standing waves. Proc. R. Soc. Lond. Ser. A Math. Phys. Sci. **218**(1132), 44–59 (1953)

55. Vakakis, A.: Non-linear normal modes (NNMs) and their applications in vibration theory: an overview. Mech. Syst. Signal Process. **11**(1), 3–22 (1997). https://doi.org/10.1006/mssp.1996.9999