Sparse identification of slow timescale dynamics

Jason J. Bramburger[*]

Department of Mathematics and Statistics, University of Victoria, Victoria, British Columbia V8W 2Y2, Canada

Daniel Dylewsky ⬤[†]

Department of Physics, University of Washington, Seattle, Washington 98195, USA

J. Nathan Kutz[‡]

Department of Applied Mathematics, University of Washington, Seattle, Washington 98195, USA

Multiscale phenomena that evolve on multiple distinct timescales are prevalent throughout the sciences. It is often the case that the governing equations of the persistent and approximately periodic fast scales are prescribed, while the emergent slow scale evolution is unknown. Yet the course-grained, slow scale dynamics is often of greatest interest in practice. In this work we present an accurate and efficient method for extracting the slow timescale dynamics from signals exhibiting multiple timescales that are amenable to averaging. The method relies on tracking the signal at evenly spaced intervals with length given by the period of the fast timescale, which is discovered by using clustering techniques in conjunction with the dynamic mode decomposition. Sparse regression techniques are then used to discover a mapping which describes iterations from one data point to the next. We show that, for sufficiently disparate timescales, this discovered mapping can be used to discover the continuous-time slow dynamics, thus providing a novel tool for extracting dynamics on multiple timescales.

I. INTRODUCTION

Many physical phenomena exhibit multiscale dynamics where the fastest timescale is relatively simple to both observe and predict, while the emergent long timescale dynamics are unknown. Examples abound in physics and range from tidal amplitudes [1,2], to molecular-dynamics simulations [3,4], to atmospheric dynamics [5], to the motion of the planets [6–8]. Mathematically, systems which exhibit multiscale dynamics are expensive to simulate since the fastest scales must be accurately resolved, and when the fast timescale is approximately periodic, this expense is primarily used to simulate predictable dynamics. In systems where the scale separation is explicit, it is sometimes the case that one can average the system to obtain the leading-order dynamics of the slow timescale evolution [9–11], although many systems lack an obvious separation of scales or even governing equations, meaning that novel methods for extracting the slow timescale dynamics must be developed.

In this work we present a computationally cheap and efficient method that integrates machine learning and multiscale modeling for extracting and forecasting slow timescale dynamics of a multiscale system. Unlike averaging techniques, the proposed mathematical architecture learns a nonlinear dynamical system characterizing the slow-scale behavior. The setting for the problem is that we are given a signal $\mathbf{\Phi}(t, \varepsilon t)$ in $\mathbb{R}^d$ which contains terms dependent on t contributing to the fast dynamics and terms dependent on εt, for some $0 < \varepsilon \ll 1$, contributing to the slow dynamics. We assume that the fast timescale terms are periodic, so $\mathbf{\Phi}$ satisfies

$$\mathbf{\Phi}(t + T, \varepsilon(t + T)) = \mathbf{\Phi}(t, \varepsilon(t + T)) \approx \mathbf{\Phi}(t, \varepsilon t), \quad (1)$$

for all t and some $T > 0$. The timescales of $\mathbf{\Phi}$ are assumed to be sufficiently disparate so that the slow dynamics are nearly constant on the interval $[t, t + T]$, coming from the size of ε relative to the fast period T. It is through these assumptions that we see that the signal exhibits a fast oscillation and a slow drift, making it amenable to averaging.

Suppose the signal is given on some finite timescale and we wish to understand the physics governing the slow timescale evolution so that the signal can be reconstructed and forecast far beyond the given time window. Since the fast-scale dynamics are relatively simple, we would like to "average" these dynamics out to forecast only the slow-scale variable. Knowing the fast-scale period $T > 0$ naturally leads to tracking the signal after each period,

$$\boldsymbol{x}_n = \mathbf{\Phi}(0, \varepsilon n T), \quad n = 0, 1, 2, 3, \ldots, \quad (2)$$

since $\mathbf{\Phi}(nT, \cdot) = \mathbf{\Phi}(0, \cdot)$ by periodicity. It follows that understanding the signal at $t = nT$ requires an understanding of the slow timescale dynamics. Our goal is to discover a mapping $\boldsymbol{F} : \mathbb{R}^d \to \mathbb{R}^d$ so that

$$\boldsymbol{x}_{n+1} = \boldsymbol{F}(\boldsymbol{x}_n) \quad (3)$$

---
[*]jbramburger7@uvic.ca
[†]dylewsky@uw.edu
[‡]kutz@uw.edu

for all $n \geqslant 0$. Beyond this, if we suppose there exists a function $\boldsymbol{G} : \mathbb{R}^d \to \mathbb{R}^d$ such that the slow timescale dynamics of a signal satisfying Eq. (1) are governed by the ordinary differential equation (ODE)

$$\partial_2 \boldsymbol{\Phi}(0, t) = \boldsymbol{G}(\boldsymbol{\Phi}(0, t)), \tag{4}$$

where ∂_2 is the derivative with respect to the second component, then expanding the left-hand side of Eq. (3) as a Taylor series about $t = nT$ using the definition of $\boldsymbol{x}_{n+1}$ from Eq. (2) gives that

$$\boldsymbol{F}(\boldsymbol{x}) = \boldsymbol{x} + \varepsilon T \boldsymbol{G}(\boldsymbol{x}) + O(\varepsilon^2). \tag{5}$$

Hence, when ε is small, Eq. (5) represents an Euler step of the slow physics. This demonstrates the duality between the functions $\boldsymbol{F}$ and $\boldsymbol{G}$, at least to $O(\varepsilon^2)$. Therefore, discovering the mapping $\boldsymbol{F}$ from data presents a novel method to extract the slow timescale physics of a given signal, something which holds great potential for multiscale systems that are so common in the biological and medical sciences [12].

There are potentially two unknowns that are required to find $\boldsymbol{G}$: the fast period $T > 0$ and the mapping $\boldsymbol{F}$. Here we use the method of sliding-window *dynamic mode decomposition* (DMD) [13] to extract the fast period and the *sparse identification of nonlinear dynamics* (SINDy) algorithm [14,15] to obtain the mapping $\boldsymbol{F}$. We summarize the method visually in Fig. 1 and explain the individual components in more detail in the following section. The advantage our proposed method presents is that it does not attempt to discover a dynamical system for the multiscale signal, since as we show in Sec. III, sparse identification procedures should be expected to fail at this task in general. This is due in part to the numerical approximation of the derivative of the signal, which in the presence of a periodic fast timescale is rapidly changing. Although the challenge of approximating the derivative can be overcome by using instead integral terms [16,17], the present method circumvents this difficulty altogether. Furthermore, the multiscale property of the data presents a unique advantage in the implementation of a regression technique which promotes sparsity by using thresholding: unsupervised timescale separation is expected to introduce error at $O(\varepsilon^2)$, whereas the sparsification procedure amounts to pruning terms of order $O(\varepsilon)$. As a result, the method by construction recovers only the leading order slow timescale dynamics.

The sparse regression framework has been used previously to discover multiscale physics [18], but with a much different procedure for the model discovery process and sampling strategy. Specifically, fast sampling of the dynamics was required since both fast and slow timescale physics needed to be discovered, whereas in this work, the fast scale is known and only the slow dynamics needs to be discovered. Modification to the basic SINDy [15] architecture allows for the discovery of partial differential equations [19], equations of rational expressions [20], and conservation laws [21]. For each of these modifications, one can imagine using the proposed method to handle multiscale temporal dynamics in an efficient manner.

This paper is organized as follows: In Sec. II we present a detailed overview of the method for extracting slow timescale physics. This includes summaries of the two major components of the method: the sliding window DMD technique and the SINDy algorithm. In Sec. III we provide a simple
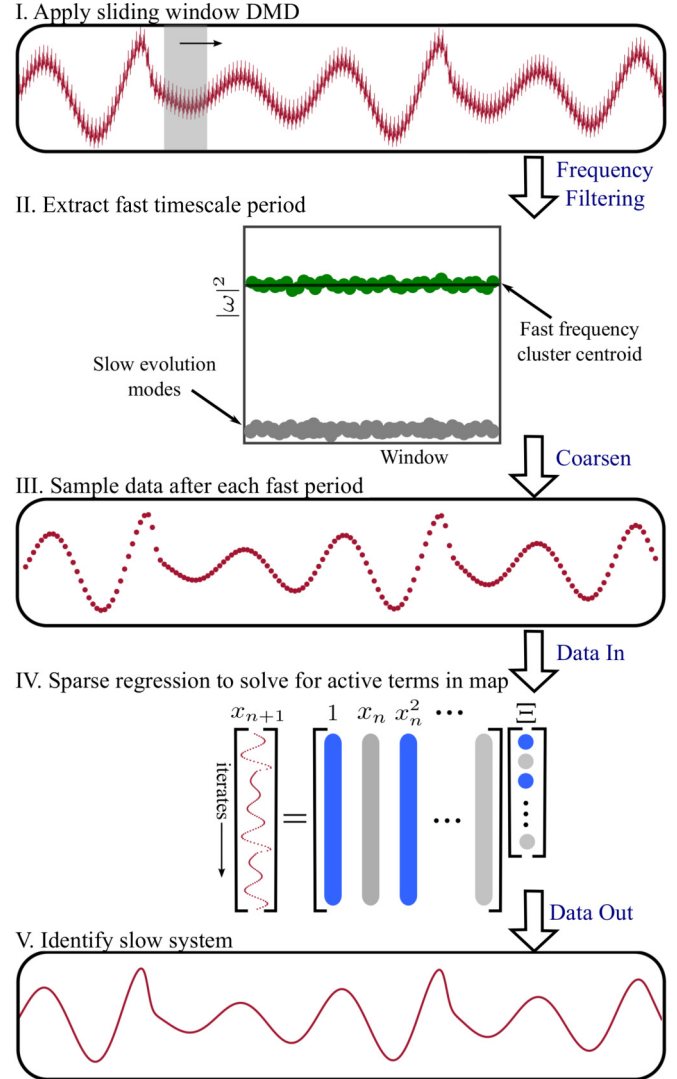

FIG. 1. An overview of the method presented in this work. We first use the sliding window DMD technique [13] to identify the fast period, then track the signal at integer multiples of the fast period, resulting in the coarsened signal $x_n = \boldsymbol{\Phi}(0, \varepsilon n T)$. We then identify a mapping $x_n \mapsto x_{n+1}$ by using the SINDy method [15], and finally use the expansion (5) to extract the slow timescale physics.

example of a multiscale system where naive application of the SINDy method fails to properly identify the full governing equations. We further supplement this discussion by showing that our methods accurately capture the slow evolution of this system once the fast dynamics have been taken out of the signal. Section IV is entirely dedicated to applications of the method, where we present examples with the discovered slow dynamics that are monotone, periodic, and chaotic. Finally, we briefly summarize our findings in Sec. V.

II. METHODS

The method employed by this work has two major components: scale separation using the sliding-window DMD and sparse regression analysis using SINDy. In this section we briefly summarize these methods and direct the reader

to Refs. [13] and [15], respectively, for a more complete discussion of each component. In its totality, our method first employs the sliding window DMD technique to identify the fast period, then tracks the signal at integer multiples of the fast period, resulting in the coarsened signal $x_n = \Phi(0, \varepsilon nT)$. We then identify a mapping F such that $x_{n+1} = F(x_n)$ using the SINDy method. Finally, we use the duality (5) to extract the slow timescale physics when ε is sufficiently small. We summarize the method visually in Fig. 1 and proceed through the following sections with a discussion of the individual components of our method.

A. Scale separation using sliding-window dynamic mode decomposition

The method presented in this work relies on foreknowledge of the timescale disparity between components constituting a multiscale signal. For systems with known governing equations, this can generally be determined by inspection of the coefficients or by perturbation expansion. For systems with unknown equations of motion, however, a data-driven discovery method must suffice. This could be accomplished by identifying peaks on the Fourier spectrum, for example. For multiscale physics, windowed Fourier transforms can help provide improved resolution of signals and their content. Indeed, such windowing procedures are the basis of multiresolution analysis and wavelet decompositions [22]. Multiresolution DMD [23] provides a analogous decomposition of multivariate data, identifying coherent spatial modes and temporal frequencies in multiscale systems.

Dynamic mode decomposition (DMD) is a model regression technique which seeks a best-fit linear representation for observed dynamics [24,25]. Given a data matrix X consisting of m sequential snapshots, DMD identifies a linear operator A which in some optimal (least-squares) sense satisfies the equation $\dot{X} = AX$. This approximation is of course unlikely to be accurate for highly nonlinear systems, but this problem can be circumvented by subsampling X onto shorter time intervals by sliding a window of width d ($d \ll m$) across the full time series. Even when global dynamics are nonlinear, local linear approximations are easily obtainable. Each local operator A has eigenvalues which characterize the timescale content of local dynamics.

In this work we use a technique introduced by Dylewsky *et al.* (2019) which leverages DMD to separate the components of multiscale data and learn local linear models for

the dynamics by clustering on their eigenspectra [13]. This method has the advantages of sparsity, flexibility, and robustness to overlapping or highly disparate dynamical timescales. By gathering eigenspectra across all sliding-window iterations of DMD, one can form a statistical picture of the global spectral content of the data. A simple clustering algorithm can identify the most prominently represented timescales and offer a parsimonious estimate of the scale components present. Even when timescales are separated by many orders of magnitude, the algorithm can be applied recursively with varied window width to properly identify them. These clusters can be depicted visually by plotting the modulus squared of the eigenvalues of the local operators A for each window. In the present scenario of a signal satisfying Eq. (1), at least two distinct clusters should be apparent: one bounded away from 0 representing the fast periodic dynamics and another near 0 representing the slow dynamics (which should appear nearly static over the comparatively short span of a windowed subsample). A cartoon of this clustering is presented in panel II of Fig. 1, while an application of the method to real data is presented in Fig. 4.

B. Sparse identification of nonlinear dynamics

The SINDy method, introduced by Brunton *et al.* (2016), is an algorithm for discovery of a symbolic representation of the governing equations of a system from time series measurements [15]. Given a data matrix comprised of sequential state measurement snapshots:

$$X_1 = \begin{bmatrix} \cdots & x^T(t_1) & \cdots \\ \cdots & x^T(t_2) & \cdots \\ & \vdots & \\ \cdots & x^T(t_m) & \cdots \end{bmatrix}, \qquad (6)$$

along with another data matrix of the same size, X_2, comprised of either the temporal derivative of the data at the same measurement times or the successive iterates of the discrete temporal data. A symbolic representation for the dynamics $\dot{x}(t) = F(x(t))$ or $x(t_{n+1}) = F(x(t_n))$ is constructed from a library of candidate functions for terms of $F(x)$. The chosen functions are evaluated on the measurement data to construct a library matrix Θ whose m rows represent the m measurement snapshots of X_1 lifted into a space of all library observables. For example, a library consisting of polynomials up to degree two for $x = (x_1, x_2) \in \mathbb{R}^2$ would look like

$$\Theta(X_1) = \begin{bmatrix} 1 & x_1(t_1) & x_2(t_1) & [x_1(t_1)]^2 & x_1(t_1)x_2(t_1) & [x_2(t_1)]^2 \\ 1 & x_1(t_2) & x_2(t_2) & [x_1(t_2)]^2 & x_1(t_2)x_2(t_2) & [x_2(t_2)]^2 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & x_1(t_m) & x_2(t_m) & [x_1(t_m)]^2 & x_1(t_m)x_2(t_m) & [x_2(t_m)]^2 \end{bmatrix}. \qquad (7)$$

The claim that the equation of motion $F(x)$ is some linear combination of the chosen library functions is equivalent to the statement

$$X_2 = \Theta(X_1)\Xi \qquad (8)$$

for some coefficient matrix Ξ. For many systems of interest, the governing equations contain only a few terms, so a sparseness requirement can be imposed on Ξ. This can be carried out by using any sparse regression algorithm on the overdetermined system of linear equations (8), e.g., elastic net

methods such as LASSO or Ridge. The approach employed in this paper, as in Ref. [15], is a sequentially thresholded least squares method in which Eq. (8) is initially solved simply by minimizing $\|X_2 - \Theta(X_1)\Xi\|_2$. Elements of Ξ with magnitude smaller than some chosen sparsity parameter $\lambda > 0$ are then thresholded to zero, and the process repeats. This iteration is carried out until all remaining coefficients have absolute values larger than λ. This process is a proxy for ℓ_0 optimization [26]. It has convergence guarantees [27] and performs comparably to LASSO in most cases at significantly reduced computational expense. Moreover, the direct correspondence between the sparsity parameter λ and the obtained dynamical coefficients in Ξ proves useful in the multiscale case treated in this paper: we show that knowledge of the timescale components present in a signal can be used to determine *a priori* a suitable value for λ.

We note that there are many variants of sparse regression, all of which typically attempt to approximate a solution to an *NP*-hard, ℓ_0-norm penalized regression. Sparsity-promoting methods like the LASSO [28,29] use the ℓ_1-norm as a proxy for sparsity since tractable computations can be performed. The iterative least-squares thresholding algorithm of the SINDy algorithm promotes sparsity through a sequential procedure. Recently, Zhang and Schaeffer [27] provided a number of rigorous theoretical results on the behavior and convergence of the SINDy algorithm. Specifically, they prove that the algorithm approximates local minimizers of an unconstrained ℓ_0-penalized least-squares problem. This allows them to provide sufficient conditions for general convergence, the rate of convergence, and conditions for one-step recovery. As shown in Champion *et al.* [30], the SINDy regression framework does not readily accommodate extensions, additional constraints, or improvements in performance. Thus the optimization formulation is extended to include additional structure, robustness to outliers, and nonlinear parameter estimation using the *sparse relaxed regularized regression* (SR3) approach that uses relaxation and partial minimization [26]. Rigorous theoretical bounds are provided for the relaxed formulation which has three advantages over the state-of-the-art sparsity-promoting algorithms: (1) solutions of the relaxed problem are superior with respect to errors, false positives, and conditioning; (2) relaxation allows extremely fast algorithms for both convex and nonconvex formulations; and (3) the methods apply to composite regularizers, essential for total variation (TV) as well as sparsity-promoting formulations using tight frames. Indeed, the SR3 formulation was shown to have superior performance (computational efficiency, higher accuracy, faster convergence rates, and greater flexibility) across a range of regularized regression problems with synthetic and real data, including applications in compressed sensing, LASSO, matrix completion, TV regularization, and group sparsity. Between the three recent papers [26,27,30], rigorous estimates can be established for this model-discovery paradigm.

III. SPARSE IDENTIFICATION OF NONLINEAR DYNAMICS AND MULTISCALE SYSTEMS

In this section we briefly demonstrate how naively applying the SINDy method to a signal in the form of (1) to discover
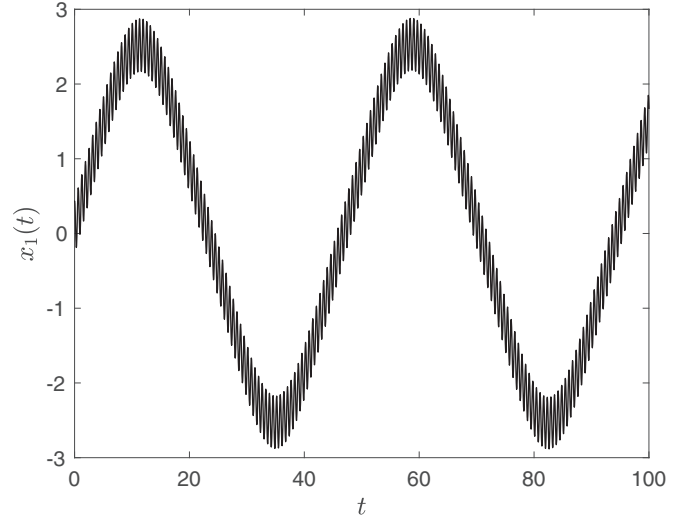

FIG. 2. The first component of the signal $\boldsymbol{x}(t)$ in Eq. (10). There are two distinct timescales and the signal is plotted for slightly more than two periods of the slow timescale. The fast period is approximately $\pi/5$.

both the fast and slow timescales simultaneously should be expected to fail, thus necessitating the method described in this paper. Alternatively, one can us fast sampling strategies to resolve the discovery process [18], but here sampling on the slow scale is all that is needed. In Ref. [13] a simple toy model was created to extract the periods of oscillation for signals formed as linear combinations of periodic phenomena. The model is given by

$$
\begin{aligned}
\dot{v}_1 &= v_2, \\
\dot{v}_2 &= -w_1^2 v_1^3, \\
\dot{w}_1 &= w_2, \\
\dot{w}_2 &= -100 w_1 - 4 w_1^3,
\end{aligned}
\tag{9}
$$

where constants are chosen to appropriately separate the timescales. The (w_1, w_2) variables are governed by an unforced Duffing equation, for which almost all initial conditions fall into steady periodic motion. The (v_1, v_2) variables form a cubic oscillator with a coefficient w_1^2 dependent on the state of w_1. The signal (1) is produced by integrating system (9) using MATLAB's ode23 function with a maximal time step of 10^{-4} and initial conditions $(v_1, v_2, w_1, w_2) = (0, 0.5, 0, 0.5)$. We then take a randomly generated orthogonal matrix $Q \in \mathbb{R}^{4 \times 4}$ to define

$$
\boldsymbol{x}(t) = Q \cdot [v_1(t), v_2(t), w_1(t), w_2(t)]^T, \tag{10}
$$

in an effort to sufficiently mix the disparate temporal dynamics. We direct the reader to Fig. 2 for a visual depiction of the first component of $\boldsymbol{x}(t) = (x_1(t), x_2(t), x_3(t), x_4(t))$, and note that the rest look similar in that we can clearly see the two timescales present in the signal $\boldsymbol{x}(t)$.

If we use the SINDy method for discovery of a continuous-time dynamical systems using Eq. (10) as the training data and a sparsity parameter $\lambda = 10^{-3}$, we discover the differential
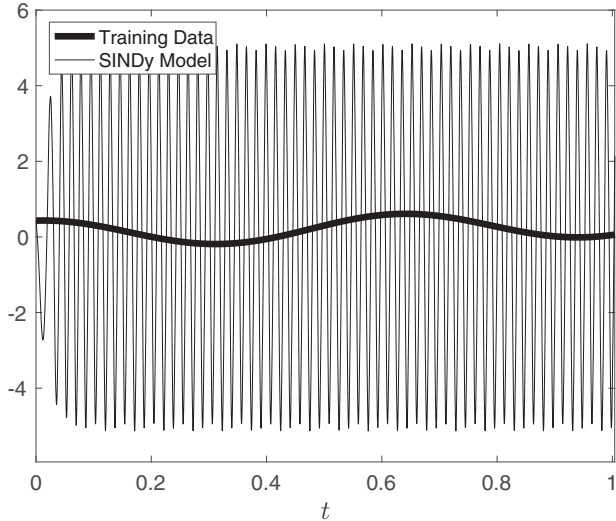
FIG. 3. A comparison of the first component of the training data against the evolution of the discovered continuous-time SINDy model with the same initial data. Notice that the discovered data have a significantly higher frequency of oscillation, as can be observed by comparing the extent of the horizontal axis in this figure with that of Fig. 2.

equation

$$\dot{x}_1 = 0.1 - 788.8x_1 + 788.0x_4 + 3.3x_4^3,$$
$$\dot{x}_2 = -833.3x_1 + 833.3x_4,$$
$$\dot{x}_3 = -0.1 - 877.8x_1 + 878.7x_4 - 3.3x_4^3,$$
$$\dot{x}_4 = -0.3 - 922.3x_1 + 924.0x_4 - 6.6x_4^3. \tag{11}$$

Even without knowing the exact values in Q we immediately note that this discovered system cannot exactly capture the dynamics of the signal since it includes constant terms. In Fig. 3 we plot the first component of the training data against the dynamics of the discovered model (11) with the same initial condition. We can see that the discovered model produces a signal with a faster frequency of oscillation than that of the original signal. This can be observed by comparing the extent of the horizontal axes of Figs. 2 and 3, where the former displays the signal for $t \in [0, 100]$ while the latter uses $t \in [0, 1]$. This significantly shorter time window in Fig. 3 is meant to emphasize the extremely fast frequency of oscillation in the discovered continuous-time SINDy model, so it should be noted that the slow dynamics of Eq. (10) are not yet visible on such a short timescale. We further note that changing the sparsity parameter λ has little effect on the discovered model, always producing a dynamical system which fails to even approximately reproduce the dynamics of the input signal (10). The data repository [31] contains all code related to this example to ensure the ability to reproduce these results and perform further experiments.

This example is merely meant to illustrate what a naive application of the continuous-time SINDy method can result in when applied to multiscale training data. We suspect that the discrepancy between the SINDy model and the training data is due at least in part to the approximation of the derivative. The fast frequency of oscillation present in the multiscale
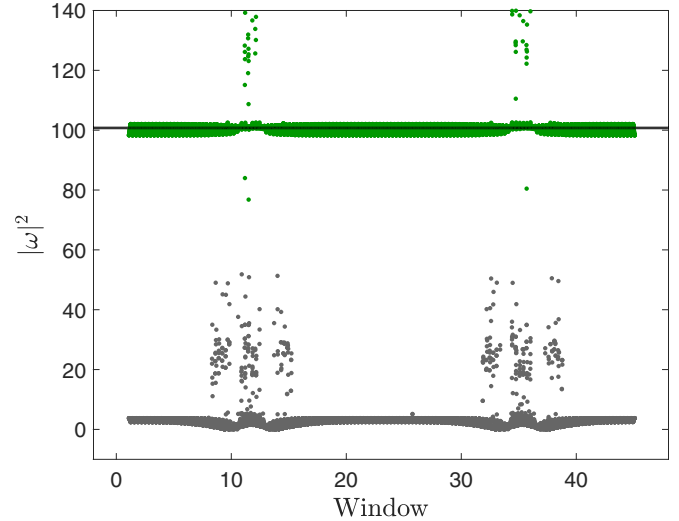

FIG. 4. The result of the sliding window DMD method applied to the signal (10). The horizontal axis represents the value of t at which the window is centered, while the vertical axis plots the square modulus of the purely imaginary eigenvalues ω of the local operator **A** for each window. Colors are used to delineate between the two clusters, representing the fast and slow components of the signal. The horizontal black line represents the centroid of the fast frequency cluster, approximately $|\omega|^2 = 100$.

training data results in a rapidly oscillating derivative, which requires finely spaced data. Hence, one way to improve the performance of the SINDy model might be to take smaller time steps in the training data, but we note that decreasing the maximal step size down a full order of magnitude to 10^{-5} shows little improvement in the discovered continuous-time SINDy model. Of course, in the case of real data it seems unlikely that such a regeneration of the data could be performed and is therefore not a practical method for improving the performance of the discovery algorithm. Another method would be to avoid taking numerical derivatives altogether and instead use weak formulations which replace the derivative with integral terms [16,17]. The strength of the method presented herein is that we are able to consider coarse-grained data which experiences little variation from one data point to the next (representing the slow dynamics), thus avoiding this difficulty altogether.

We now turn to coarsening the signal by tracking it at integer multiples of the fast timescale period. The training data in this case are exactly the same as that used to discovery the continuous-time SINDy model (11). In Fig. 4 we plot the resulting frequencies ω extracted from the sliding window DMD method, where one can clearly see the separation between fast and slow dynamics. The centroid of the fast cluster gives that the fast component of the signal has period given by $T = \pi/5$, and hence we can track the full signal at integer multiples of this fast period. We comment that the frequencies showing the largest variation from the clusters are localized to windows near the extreme points of the slow dynamics, representing regions where the slow dynamics are not approximately constant in the relatively small windows. It is exactly this problem that necessitates the sliding window
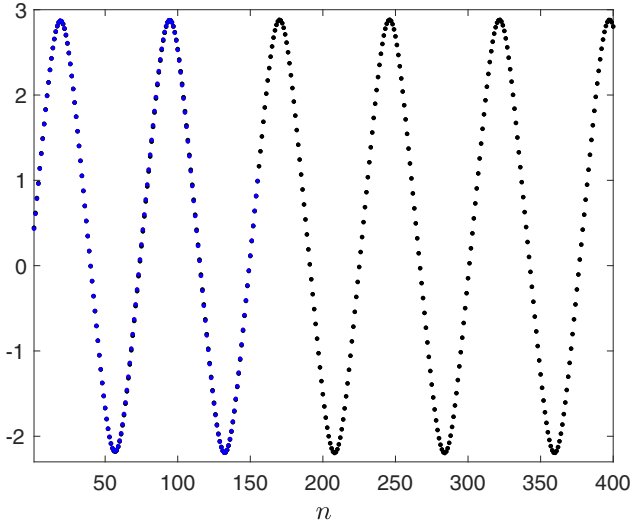
FIG. 5. Iterates of the discovered mapping for the slow timescale dynamics of the first component of (10), seeded with the initial value of the training data. Training data are provided in blue along with the iterates of the discovered mapping in black.

method in the first place, as looking at the whole signal will inevitably introduce large discrepancies since the slow dynamics show large variation on long timescales.

Having now obtained the fast period of oscillation, we are able to apply the sparse identification procedure to obtain a mapping over the slow dynamics. The training data for the SINDy method uses 200 fast periods—long enough to observe at least two full slow periods. The iterates of the component $x_1(nT)$ to the discovered mapping $\boldsymbol{F}$ are presented in Fig. 5 against the training data, while we note that the other components look similar. The cumulative error over all components is 1%, a significant improvement from the model (11). Hence, we see that in the absence of the fast oscillations the SINDy method is able to effectively track the slow evolution of the signal.

IV. APPLICATIONS

In this section we discuss applications of the method to three different systems whose slow timescale dynamics exhibit steady-state, periodic, and chaotic dynamics. All model discovery is performed by using the SINDy method for maps [14,15] with a library of functions containing monomials up to degree five, unless otherwise stated. All code related to these examples can be found in the data repository [31].

A. Singular perturbations and averaging

Let us consider a simple and motivating example. Consider the scalar ODE

$$\dot{x} = \varepsilon x[1 - x + \sin(2\pi t)], \quad x(0) = x_0 \in \mathbb{R}_+, \quad (12)$$

with $0 < \varepsilon \ll 1$. Since the ODE is 1-periodic in the independent variable t, the theory of averaging for dynamical systems [9,11] dictates that there exists a constant $C > 0$ such that the

solution $x(t)$ satisfies $|x(t) - y(t)| \leqslant C\varepsilon$ for all $t \geqslant 0$,[1] where $y(t)$ is a solution of the autonomous ODE

$$\dot{y} = \varepsilon y(1 - y), \quad y(0) = x_0. \quad (13)$$

The solution $y(t)$ is explicitly given by

$$y(t) = \frac{x_0}{x_0 + (1 - x_0)\exp(-\varepsilon t)},$$

where we can therefore see that $y(t)$ evolves on a slow timescale, εt. Hence, $y(t)$ makes up the slow dynamics to at least $O(\varepsilon)$ and, by using Eq. (5), we expect the mapping (3) to be of the form $\boldsymbol{F}(x) = x + \varepsilon x(1 - x) + O(\varepsilon^2)$.

To illustrate the performance of our method, we take $\varepsilon = 10^{-2}$ and a sparsity parameter $\lambda = \varepsilon^2$ to truncate at order ε^2. We discover the mapping (3) here to be

$$\boldsymbol{F}(x) = 1.001x - 0.009778x^2$$
$$\approx x + \varepsilon x(1 - x) + O(\varepsilon^2), \quad (14)$$

conforming with our expectation from the above analysis. Furthermore, the leading-order terms in Eq. (14) represent a forward Euler discretization of Eq. (13) with step size ε, implying that standard error bounding arguments based on the initial condition and the value of ε can be applied to bound the difference between the solutions of (12) at integer values of $n \geqslant 0$ and the iterates of the map (14).

This example illustrates how the method performs against a benchmark ODE where the slow dynamics can be explicitly determined via analysis. In reality, few systems which exhibit multiscale dynamics take the form of singularly perturbed dynamical systems and therefore even determining the fast timescale period to average over is a nontrivial task.

B. Planetary dynamics

As in the toy model example of Sec. III, multiscale phenomena can often be observed when there are two periodic components of the signal with a vast separation between their periods. That is, consider a signal which can approximately be written as

$$\boldsymbol{x}(t) = P_0(t) + P_1(t), \quad (15)$$

where P_0 is periodic with period $T_0 > 0$ and P_1 is periodic with period $T_1 > 0$. Assuming that $T_0 \ll T_1$, naturally leads to the scale separation parameter

$$\varepsilon := \frac{T_0}{T_1} \ll 1.$$

We may rescale $t = T_0\tau$ so that $\tilde{P}_0(\tau) := P_0(T_0\tau)$ is now 1-periodic and $P_1(T_0\tau)$ is ε^{-1}-periodic. Setting

$$\tilde{P}_1(s) := P_1(\varepsilon^{-1}s)$$

makes $\tilde{P}_1$ 1-periodic as well. Hence, the full signal $x(\tau)$ can equivalently be written

$$\tilde{\boldsymbol{x}}(\tau) := \boldsymbol{x}(T_0\tau) = \tilde{P}_0(\tau) + \tilde{P}_1(\varepsilon\tau),$$

---

[1]Technically, the theory dictates that this bound only holds for t on a timescale of length $1/\varepsilon$, but since all positive initial conditions of Eq. (12) evolve towards a global attractor, the bound can be shown to hold for all $t \geqslant 0$ so long as the bound $C > 0$ depends on x_0.
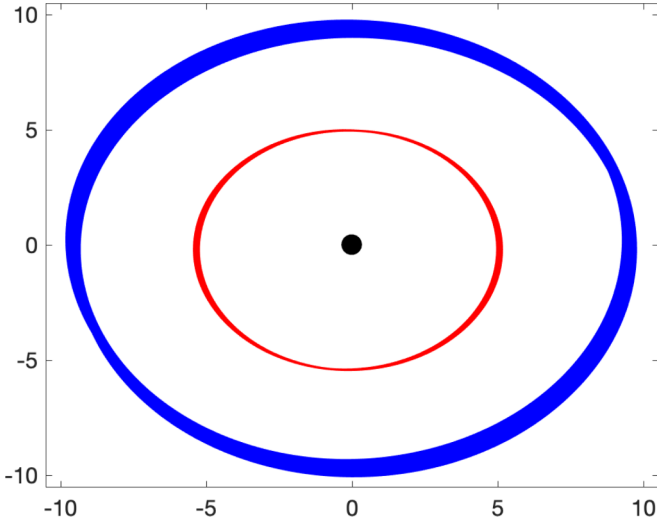
FIG. 6. The procession of Jupiter (red) and Saturn (blue) about the Sun (black dot) in their orbital plane in nondimensionalized coordinates. The fast timescale for each planet constitutes simple procession about the Sun: 11.86 years for Jupiter and 28.5 years for Saturn. Beyond this fast procession are a number of slow scales that drive the eccentricity of the orbit, giving the appearance that they trace out thick circles in the orbital plane over thousands of years.

for which both functions $\tilde{P}_0$ and $\tilde{P}_1$ are 1-periodic and clearly demonstrate the separation of timescales of interest in this work. The mapping (3) tracks $\tilde{x}(n)$ for integers $n \geqslant 0$, or, equivalently, $x(nT_0)$.

Such a phenomenon was exemplified in the model (9) and can further be found in the motion of the planets Saturn and Jupiter around the Sun, as described by a three-body planetary model. For this test case we generate data by constructing a system of three objects subject to pairwise gravitational attraction (using the nonrelativistic Newtonian formulation). The masses and initial conditions of these objects are assigned based on measurements of the true Jupiter-Saturn-Sun system in its center-of-mass reference frame. Time series trajectories in $\mathbb{R}^3$ are obtained via symplectic numerical integration over a period of 10^6 years.

The signal for each planet contains multiple different timescales, and the motion of each planet in its orbital plane can be approximated by a signal of the form (15). The periodic function P_0 describes the primary orbit of the planets around the Sun, for which upon applying the sliding window DMD technique we find periods $T_0 = 11.86$ years for Jupiter and $T_0 = 29.5$ years for Saturn. The function P_1 describes the eccentricity of these orbits, for which the sliding-window DMD technique has shown that $T_1 \approx 46\,800$ years[2] for both planets [13]. This periodic eccentricity of the orbits comes from the interaction of the two planets orbiting the relatively massive Sun and can be observed in Fig. 6 where we plot their orbits around the Sun in their orbital plane.

[2]It was shown in Ref. [13] that Jupiter has a periodic component with period between our T_0 and T_1, but this period is almost exactly $9T_0$ and is expected to be a numerical artifact.

Let us begin by considering our signal to be the motion of Jupiter projected entirely into its orbital plane over time. Here we have $T_0 = 11.86$ with t measured in years and so from the construction above we get $\varepsilon \approx 2.6 \times 10^{-4}$. Denote by x_n^J and y_n^J the x and y components of the position of Jupiter in its orbital plane at time $t = nT_0 = 11.86n$ years. Taking a sparsity parameter λ on the order of $\varepsilon^2 \sim 10^{-8}$ is impractical since numerical error present in either simulating the data or implementing the SINDy method should be expected to show up in the discovered mapping. To overcome this, we note that $T_0\varepsilon \sim 10^{-3}$, and so properly discovering $\boldsymbol{G}$ per Eq. (5) is expected to succeed with $10^{-3} \leqslant \lambda < T_0\varepsilon$ since the lower bound should be large enough to eliminate numerical error and $O(\varepsilon^2)$ terms, while the upper bound allows for discovery of the $O(\varepsilon)$ terms. With $\lambda = 10^{-3}$ we discover the slow-scale mapping

$$\begin{bmatrix} x_{n+1}^J \\ y_{n+1}^J \end{bmatrix} = \begin{bmatrix} 0.0036 \\ -0.0018 \end{bmatrix} + \begin{bmatrix} 0.9999 & 0.0114 \\ -0.0114 & 0.9999 \end{bmatrix} \cdot \begin{bmatrix} x_n^J \\ y_n^J \end{bmatrix} \quad (16)$$

by using training data comprised of 1500 fast timescale periods—long enough to observe at least two full periods of the first slow timescale. The mapping (16) can be used to find that the slow timescale mapping $\boldsymbol{G}$ in this case is given by

$$\boldsymbol{G}(x, y) = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \cdot \begin{bmatrix} 1.17 + 3.69x \\ 0.57 + 3.69y \end{bmatrix} \quad (17)$$

using Eq. (5). Hence, we see that the eccentricity dynamics of Jupiter is given, to leading order, by an ellipse in the orbital plane. We note that the conservative structure of the slow dynamics is not necessarily guaranteed by the conservative structure of the original three-body problem since it could be the case that energy flows from one scale to the next. Despite this, the resulting conservative structure of Eq. (17) could potentially reflect that the training data is stable up to $O(\varepsilon^2)$ perturbations on very long timescales. Hence, we do not expect that Eq. (17) is valid for all time, but only long finite timescales until evolution on even slower timescales begins to influence the $O(\varepsilon)$ dynamics of Jupiter's orbit.

We may proceed in the same way for the procession of Saturn around the Sun to find a similar result. In this case we have $T_0 = 29.5$ years, making up the fast scale period, and therefore the scale separation parameter is given by $\varepsilon \approx 6.3 \times 10^{-4}$. Performing our mapping discovery procedure again with $\lambda = 10^{-3}$, for the same reasons as in the case of Jupiter, results in the mapping

$$\begin{bmatrix} x_{n+1}^S \\ y_{n+1}^S \end{bmatrix} = \begin{bmatrix} -0.0038 \\ 0.0025 \end{bmatrix} + \begin{bmatrix} 1.0000 & -0.0100 \\ 0.0100 & 1.0001 \end{bmatrix} \cdot \begin{bmatrix} x_n^S \\ y_n^S \end{bmatrix}, \quad (18)$$

where the variables (x_n^S, y_n^S) represent the orthogonal components of the position of Saturn in its orbital plane at $t = 29.5n$ years. We can similarly rearrange Eq. (18) via Eq. (5) to find that the eccentricity dynamics of Saturn is confined to an ellipse in the orbital plane.

C. Chaotic slow dynamics

In this example we apply the method to systems for which the slow timescale dynamics are chaotic. We consider a signal

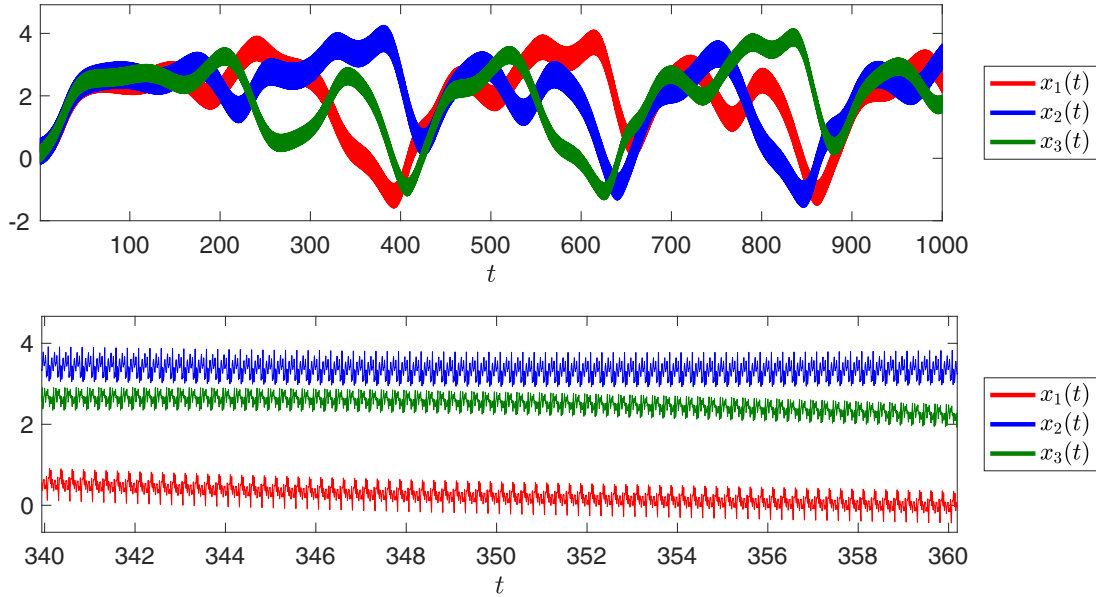$$\boldsymbol{x}(t) = \varepsilon P(t) + C(\varepsilon t), \quad (19)$$

FIG. 7. A signal of the form (19) with periodic fast dynamics given by Eq. (21) with randomly generated Fourier coefficients and slow dynamics governed by the system (20). The top shows the signal over a long timescale so that the chaotic dynamics can be observed. The bottom shows a smaller time window so that only the periodic fast scale can be observed.

where $P(t)$ is periodic with period $T > 0$ and $C(t)$ is a trajectory of a chaotic dynamical system. For the purpose of illustration we will take $C(t)$ to be a trajectory on the circularly symmetric attractor of Thomas [32], given by the three-dimensional dynamical system

$$
\begin{aligned}
\dot{C}_1 &= \sin{(C_2)} - 0.2C_1, \\
\dot{C}_2 &= \sin{(C_3)} - 0.2C_2, \\
\dot{C}_3 &= \sin{(C_1)} - 0.2C_3,
\end{aligned}
\tag{20}
$$

with initial condition $C(0) = (0.3, 0.2, 0.1)^T$. The damping value 0.2 is chosen so that the system is indeed chaotic and the initial condition was chosen arbitrarily to produce a trajectory on the chaotic attractor. The periodic signal $P(t)$ will take the form of a truncated Fourier series

$$
P(t) = a_0 + \sum_{n=1}^{N} \left[a_n \cos{\left(\frac{2\pi t}{T}\right)} + b_n \sin{\left(\frac{2\pi t}{T}\right)} \right]
\tag{21}
$$

for some $N > 1$ fixed and coefficients $a_0, a_n, b_n \in \mathbb{R}^3$. For our work here we will fix $N = 10$ and note that working with larger or smaller N produces nearly identical results. We further take the coefficients a_n and b_n to be drawn from the uniform distribution on the unit cube $[-1, 1]^3$, fix $\varepsilon = 0.1$ and $T = \frac{1}{4}$, which efficiently separates the timescales. The reader is referred to Fig. 7 for a characteristic illustration of the signal.

As before we will fix $\lambda = \varepsilon^2 = 0.01$ to ensure that only $O(\varepsilon)$ terms are present in the discovered equation. However, since the discovered equation contains sinusoidal functions, a library containing only monomials will never be able to fully reproduce the dynamics of the slow scale. Hence, the best one could hope for is to produce a monomial series representation of the dynamics of system (20), but the truncation at $O(\varepsilon^2)$ will inevitably truncate the series represen-

tation as well. Furthermore, the chaotic nature of system (20) leads one to conjecture that SINDy will not necessarily just return a truncated Taylor series representation for $\sin(x)$ since the elements of $C(t)$ do not remain small on large timescales. This is confirmed by implementing the discovery method 100 different times, resulting in 100 different randomized functions $P(t)$, and discovering $\boldsymbol{F}(\boldsymbol{x}) = \boldsymbol{x} + O(\varepsilon^2)$ in every case, meaning that the slow timescale dynamics are unable to be picked up at all with such a monomial basis. This emphasizes that the choice of basis functions plays potentially an even more critical role in the case of discovering the dynamics of the slow timescale system than it potentially would for discovering systems with a single timescale.

The inclusion of both sine and cosine terms into the library immediately remedies the above problem. For example, one implementation of this method resulted in the system

$$
\boldsymbol{F}(\boldsymbol{x}) = \begin{bmatrix} 0.02508 \sin{(x_2)} + 0.9949 x_1 \\ 0.02428 \sin{(x_3)} + 0.9947 x_2 \\ 0.02471 \sin{(x_1)} + 0.9951 x_3 \end{bmatrix},
\tag{22}
$$

which is characteristic of all implementations with randomized Fourier coefficients in Eq. (21). After rearranging, the slow dynamical system $G(\boldsymbol{x})$ is recovered as

$$
\boldsymbol{G}(\boldsymbol{x}) = \begin{bmatrix} 1.0032 \sin{(x_2)} - 0.2040 x_1 \\ 0.9712 \sin{(x_3)} - 0.2120 x_2 \\ 0.9884 \sin{(x_1)} - 0.1960 x_3 \end{bmatrix},
$$

which agrees with Eq. (20) up to $O(\varepsilon^2)$.

V. CONCLUSION

In this work we have seen a computationally cheap and efficient method for discovering the slow timescale dynamical

system of a multiscale signal that is amenable to averaging. This method relies on tracking the signal not continuously, but after each fast period, to produce a mapping which parsimoniously describes the slow physics. The resulting mapping can also be related to the continuous-time dynamics of the slow timescale via the Euler stepping scheme (5). Furthermore, the fast period can be extracted accurately by using the recently developed sliding window DMD technique with clustering of eigenfrequencies [13], which can also be used to reconstruct the fast component of the signal. The result is an algorithm that reliably reproduces the slow evolution of a multiscale signal using DMD for the fast timescale and SINDy for the slow timescale. Given the difficulty in approximating the emergent slow scale evolution dynamics of multiscale systems, the method provides a viable architecture for coarse graining to achieve accurate, interpretable, and parsimonious

dynamical models for slow-scale physics. Finally, we note that, depending on the scaling, averaging can yield trivial slow dynamics, in which case homogenization is the correct procedure [33]. This potentially limits the class of signals for which our techniques are applicable, although our work herein provides an initial step towards data-driven discovery of the slow timescale dynamics of multiscale signals.

The supporting data for this article are available from the data repository [31].

[1] C. Wunsch, The long-period tides, Rev. Geophys. **5**, 447 (1967).

[2] C. Wunsch, D. B. Haidvogel, and M. Iskandarani, Dynamics of the long-period times, Prog. Oceanogr. **40**, 81 (1997).

[3] H. C. Andersen, Molecular dynamics simulations at constant pressure and/or temperature, J. Chem. Phys. **72**, 2384 (1980).

[4] M. Karplus and J. A. McCammon, Molecular dynamics simulations of biomolecules, Nat. Struct. Biol. **9**, 646 (2002).

[5] M. Palus, Multiscale Atmospheric Dynamics: Cross-Frequency Phase-Amplitude Coupling in the Air Temperature, Phys. Rev. Lett. **112**, 078702 (2014).

[6] R. Brasser, A. C. Barr, and V. Dobos, The tidal parameters of trappist-1b and c, Mon. Not. R. Astron. Soc. **487**, 34 (2019).

[7] J. Laskar, Secular evolution of the solar system over 10 million years, Astron. Astrophys. **198**, 341 (1988).

[8] R. Malhotra, M. Holman, and T. Ito, Chaos and stability of the solar system, Proc. Natl. Acad. Sci. USA **98**, 12342 (2001).

[9] J. Guckenheimer and P. Holmes, *Nonlinear Oscillations, Dynamical Systems and Bifurcations of Vector Fields* (Springer-Verlag, New York, 1983).

[10] Y. Kifer, Averaging principle for fully coupled dynamical systems and large deviations, Ergod. Theory Dyn. Syst. **24**, 847 (2004).

[11] J. A. Sanders, F. Verhulst, and J. Murdock, *Averaging Methods in Nonlinear Dynamical Systems* (Springer-Verlag, New York, 2007).

[12] M. Alber, A. B. Tepole, W. R. Cannon, S. De, S. Dura-Bernal, K. Garikipati, G. Karniadakis, W. W. Lytton, P. Perdikaris, L. Petzold, and E. Kuh, Integrating machine learning and multiscale modeling - perspectives, challenges, and opportunities in the biological, biomedical, and behavioral sciences, npj Digit. Med. **2**, 115 (2019).

[13] D. Dylewsky, M. Tao, and J. N. Kutz, Dynamic mode decomposition for multiscale nonlinear physics, Phys. Rev. E **99**, 063311 (2019).

[14] J. J. Bramburger and J. N. Kutz, Poincaré maps for multiscale physics discovery and nonlinear Floquet theory, Phys. D **408**, 132479 (2020).

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

[16] P. A. K. Reinbold, D. R. Gurevich, and R. O. Grigoriev, Using noisy or incomplete data to discover models of spatiotemporal dynamics, Phys. Rev. E **101**, 010203 (2020).

[17] H. Shaeffer and S. McCalla, Sparse model selection via integral terms, Phys. Rev. E **96**, 023302 (2017).

[18] K. P. Champion, S. L. Brunton, and J. N. Kutz, Discovery of nonlinear multiscale systems: Sampling strategies and embeddings, J. Appl. Dyn. Syst. **18**, 312 (2018).

[19] S. H. Rudy, S. L. Brunton, J. L. Proctor, and J. N. Kutz, Data-driven discovery of partial differential equations, Sci. Adv. **3**, e1602614 (2017).

[20] K. Kaheman, J. N. Kutz, and S. L. Brunton, SINDy-PI: A robust algorithm for parallel implicit sparse identification of nonlinear dynamics, J. Open Source Soft. **5**, 2104 (2020).

[21] E. Kaiser, J. N. Kutz, S. L. Brunton, Discovering Conservation Laws from Data for Control, in 2018 IEEE Conference on Decision and Control (CDC), Miami Beach, FL, 6415 (2018).

[22] S. Mallat, *A Wavelet Tour of Signal Processing* (Elsevier, New York, 1999).

[23] J. N. Kutz, X. Fu, and S. L. Brunton, Multiresolution dynamic mode decomposition, J. Appl. Dyn. Syst. **15**, 713 (2016).

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

[25] J. N. Kutz, S. L. Brunton, B. W. Brunton, and J. L. Proctor, *Dynamic Mode Decomposition: Data-Driven Modeling of Complex Systems* (Society for Industrial and Applied Mathematics, Philadelphia, 2016).

[26] P. Zheng, T. Askham, S. L. Brunton, J. N. Kutz, and A. Y. Aravkin, A unified framework for sparse relaxed regularized regression: Sr3, IEEE Access **7**, 1404 (2019).

[27] L. Zhang and H. Schaeffer, On the convergence of the sindy algorithm, Multiscale Model. Simul. **17**, 948 (2019).

[28] W. Su, B. Ma, and E. Candes, False discoveries occur early on the lasso path, Ann. Stat. **45**, 2133 (2017).

[29] R. Tibshirani, Regression shrinkage and selection via the lass, J. Roy. Stat. Soc. B **58**, 267 (1996).

[30] K. Champion, P. Zheng, A. Aravkin, S. L. Brunton, and J. N. Kutz, A unified sparse optimization framework to learn parsimonious physics-informed models from data, arXiv:1906.10612.

[31] J. J. Bramburger, GitHub/jbramburger/Slow-Discovery (2020).

[32] R. Thomas, Deterministic chaos seen in terms of feedback circuits: Analysis, synthesis, 'labyrinth chaos', Int. J. Bifurcation Chaos Appl. Sci. Eng. **9**, 1889 (1999).

[33] D. Givon, R. Kupferman, and A. Stuart, Extracting macroscopic dynamics: Model problems and algorithms, Nonlinearity **17**, R55-R127 (2004).