

Sparse Identification of Nonlinear Dynamics with Control (SINDYc) ^{*}

Steven L. Brunton ^{*} Joshua L. Proctor ^{**} J. Nathan Kutz ^{***}

^{*} Department of Mechanical Engineering, University of Washington,
Seattle, WA 98195 USA (e-mail: sbrunton@uw.edu)

^{**} Institute for Disease Modeling, Bellevue, WA 98004 USA, (e-mail:
joproctor@intven.com)

^{***} Department of Applied Mathematics, University of Washington,
Seattle, WA 98195 USA, (e-mail: kutz@uw.edu)

Abstract: Identifying governing equations from data is a critical step in the modeling and control of complex dynamical systems. Here, we investigate the data-driven identification of nonlinear dynamical systems with inputs and forcing using regression methods, including sparse regression. Specifically, we generalize the sparse identification of nonlinear dynamics (SINDY) algorithm to include external inputs and feedback control. This method is demonstrated on examples including the Lotka-Volterra predator–prey model and the Lorenz system with forcing and control. We also connect the present algorithm with the dynamic mode decomposition (DMD) and Koopman operator theory to provide a broader context.

© 2016, IFAC (International Federation of Automatic Control) Hosting by Elsevier Ltd. All rights reserved.

Keywords: Dynamical systems, control, system identification, sparse regression

1. INTRODUCTION

The data-driven modeling of complex systems is currently undergoing a revolution. There is unprecedented availability of high-fidelity measurements from historical records, numerical simulations, and experimental data, and recent developments in machine learning and compressed sensing make it possible to extract more from this data. Systems of interest, such as a turbulent fluid, an epidemiological system, a network of neurons, financial markets, or the climate, are high-dimensional, nonlinear, and exhibit multi-scale phenomena in both space and time. However, many systems evolve on a low-dimensional attractor that may be characterized by large-scale coherent structures (Holmes et al., 2012; Holmes and Guckenheimer, 1983).

System identification comprises a large collection of methods to characterize a dynamical system from data. Many techniques in system identification (Ljung, 1999), including dynamic mode decomposition (DMD) (Schmid and Sesterhenn, 2008; Rowley et al., 2009; Schmid, 2010; Tu et al., 2014) and DMD with control (DMDc) (Proctor et al., 2016a), are designed to handle high-dimensional data with the assumption of linear dynamics; historically, there have been relatively few techniques to identify nonlinear dynamical systems from data. However, DMD has strong connections to nonlinear dynamics through Koopman operator theory (Koopman, 1931; Mezić and Banaszuk, 2004; Mezić, 2005), which spurred significant interest and developments (Rowley et al., 2009; Tu et al., 2014; Budišić et al., 2012; Mezić, 2013).

^{*} SLB acknowledges support from the U.S. Air Force Center of Excellence on Nature Inspired Flight Technologies and Ideas (FA9550-14-1-0398). JLP thanks Bill and Melinda Gates for their active support of the Institute of Disease Modeling and their sponsorship through the Global Good Fund. JNK acknowledges support from the U.S. Air Force Office of Scientific Research (FA9550-09-0174).

A recent breakthrough in nonlinear system identification (Schmidt and Lipson, 2009) uses genetic programming (Koza et al., 1999) to construct families of candidate nonlinear functions for the rate of change of state variables in time. A parsimonious model is chosen from this family by finding a Pareto optimal solution that balances model complexity with predictive accuracy. In a related modeling framework, we have developed an algorithm for the sparse identification of nonlinear dynamics (SINDY) from data (Brunton et al., 2016b), relying on the fact that most dynamical systems of interest have relatively few nonlinear terms in the dynamics out of the family of possible terms (i.e., polynomial nonlinearities, etc.). This method uses sparsity promoting techniques to find models that automatically balance sparsity in the number of terms with model accuracy. An earlier related algorithm (Wang et al., 2011) uses compressed sensing (Donoho, 2006; Candès, 2006; Baraniuk, 2007), while our algorithm uses sparse regression (Tibshirani, 1996) to handle measurement noise and overdetermined cases when we have more time snapshots than state measurements.

There are many other interesting methods recently developed to incorporate sparsity and nonlinear dynamics (Schaeffer et al., 2013; Ozoliņš et al., 2013; Mackey et al., 2014). In addition, there are numerous exciting directions in equation-free modeling (Kevrekidis et al., 2003), including the Perron-Frobenius operator (Froyland and Padberg, 2009), cluster reduced-order models based on probabilistic transition between various system behaviors (Kaiser et al., 2014), and methods for uncertainty quantification and subspace analysis in turbulent flows and the climate (Majda and Harlim, 2007; Majda et al., 2009; Sapsis and Majda, 2013).

Beyond modeling, a goal for many complex systems is active feedback control, as in many fluid dynamic appli-

cations (Brunton and Noack, 2015). Extending the data-driven methods above to disambiguate between the effects of dynamics and actuation is a critical step in developing nonlinear input–output models that are suitable for control design. Similar to the extension of dynamic mode decomposition to include the effects of control (Proctor et al., 2016a), here we extend the SINDY algorithm (Brunton et al., 2016b) to include external inputs and control. We also demonstrate the relationship of SINDY, with and without control, to DMD and Koopman methods, concluding that each of these are variations of model identification from data using advanced regression techniques.

2. MODEL IDENTIFICATION VIA REGRESSION

Here we review various techniques in system identification, including dynamic mode decomposition (DMD), Koopman analysis, and the sparse identification of nonlinear dynamics (SINDY). Each of these methods is cast as a regression problem of data onto models, and the schematic overview of these methods is shown in Fig. 1.

2.1 Dynamic mode decomposition

The dynamic mode decomposition (DMD) originated in the fluids community to extract spatial-temporal coherent structures from fluid data sets (Schmid and Sesterhenn, 2008; Rowley et al., 2009; Schmid, 2010; Tu et al., 2014). DMD modes are spatially coherent and oscillate at a fixed frequency and/or growth or decay rate. Since fluids data is typically high-dimensional, DMD is built on the proper orthogonal decomposition (POD) (Holmes et al., 2012), effectively recombining POD modes in a linear combination to enforce the temporal coherence.

First, we collect multiple snapshots of high-dimensional fluid data in time $\mathbf{x}_k = \mathbf{x}(k\Delta t) \in \mathbb{R}^n$, where n represents the number of spatial measurements, which may easily represent millions or billions of degrees of freedom. In DMD, we seek a linear operator \mathbf{A} that approximately relates these snapshots, at least for short periods of time:

$$\mathbf{x}_{k+1} \approx \mathbf{A}\mathbf{x}_k. \quad (1)$$

If we collect $m + 1$ snapshots and arrange in two matrices:

$$\mathbf{X} = \begin{bmatrix} | & | & & | \\ \mathbf{x}_1 & \mathbf{x}_2 & \cdots & \mathbf{x}_m \\ | & | & & | \end{bmatrix}, \quad \mathbf{X}' = \begin{bmatrix} | & | & & | \\ \mathbf{x}_2 & \mathbf{x}_3 & \cdots & \mathbf{x}_{m+1} \\ | & | & & | \end{bmatrix}, \quad (2)$$

it is possible to related these matrices by:

$$\mathbf{X}' \approx \mathbf{A}\mathbf{X}. \quad (3)$$

In principle, for low-dimensional data it is possible to solve directly for the best-fit linear operator \mathbf{A} that minimizes $\|\mathbf{X}' - \mathbf{A}\mathbf{X}\|_F$ using a least-squares regression, where $\|\cdot\|_F$ is the Frobenius norm. Numerically, the singular value decomposition (SVD) is used to apply the pseudo-inverse of \mathbf{X} to both sides of Eq. (3). However, when the state dimension n is large, then \mathbf{A} is high-dimensional with n^2 elements, and might not be representable computationally. Instead, we apply the proper orthogonal decomposition to the data \mathbf{X} and compute a reduced operator $\tilde{\mathbf{A}}$ that acts on

POD coefficients. It is possible to reconstruct the leading eigenvalues and eigenvectors of the high-dimensional \mathbf{A} matrix from the eigendecomposition of $\tilde{\mathbf{A}}$.

- (1) Compute the economy-sized SVD of \mathbf{X} :

$$\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^*, \quad (4)$$

where $\mathbf{U} \in \mathbb{R}^{n \times m}$, $\mathbf{\Sigma} \in \mathbb{R}^{m \times m}$, and $\mathbf{V} \in \mathbb{R}^{m \times m}$.

- (2) Compute the projection of the least-square solution $\bar{\mathbf{A}} = \mathbf{X}'\mathbf{X}^\dagger$ onto POD modes, given by the columns of \mathbf{U} , where $\mathbf{X}^\dagger = \mathbf{V}\mathbf{\Sigma}^{-1}\mathbf{U}^*$ is the pseudo-inverse:

$$\tilde{\mathbf{A}} = \mathbf{U}^*\bar{\mathbf{A}}\mathbf{U} = \mathbf{U}^*\mathbf{X}'\mathbf{V}\mathbf{\Sigma}^{-1}. \quad (5)$$

Note that $\tilde{\mathbf{A}}$ is an $m \times m$ matrix, where m is the number of time snapshots; this matrix advances POD coefficients forward in time.

- (3) Compute the eigendecomposition of $\tilde{\mathbf{A}}$:

$$\tilde{\mathbf{A}}\mathbf{W} = \mathbf{W}\mathbf{\Lambda}. \quad (6)$$

- (4) The eigenvalues in $\mathbf{\Lambda}$ are also eigenvalues of the full \mathbf{A} matrix, and these are called *DMD eigenvalues*. The corresponding eigenvectors of \mathbf{A} , called *DMD modes*, are constructed as (Tu et al., 2014):

$$\Phi = \mathbf{X}'\mathbf{V}\mathbf{\Sigma}^{-1}\mathbf{W}. \quad (7)$$

It is also possible to truncate the SVD at order r , retaining only the first r POD modes, and resulting in an $r \times r$ matrix $\tilde{\mathbf{A}}$. The DMD modes ϕ are spatially coherent and oscillate and/or grow or decay at the fixed frequency λ .

The dynamic mode decomposition has been applied to a wide range of problems including fluid mechanics (Rowley et al., 2009; Tu et al., 2014; Mezić, 2013), epidemiology (Proctor and Echhoff, 2015), neuroscience (Brunton et al., 2016a), robotics (Berger et al., 2015), and video processing (Grosek and Kutz, 2014; Erichson et al., 2015). However, many of these applications have the ultimate goal of closed-loop feedback control.

2.2 Dynamic mode decomposition with control

To disambiguate the effect of internal dynamics from actuation or external inputs, the dynamic mode decomposition with control (DMDc) was developed (Proctor et al., 2016a). In DMDc, the linear state dynamics in Eq. (1) are augmented to include the effect of actuation inputs \mathbf{u} :

$$\mathbf{x}_{k+1} \approx \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k. \quad (8)$$

We still collect the state snapshots from Eq. (2), but now we collect an additional matrix for the control history:

$$\mathbf{\Upsilon} = \begin{bmatrix} | & | & & | \\ \mathbf{u}_1 & \mathbf{u}_2 & \cdots & \mathbf{u}_m \\ | & | & & | \end{bmatrix}. \quad (9)$$

In DMDc, \mathbf{A} and \mathbf{B} are approximated from data via:

$$\mathbf{X}' \approx [\mathbf{A} \ \mathbf{B}] \begin{bmatrix} \mathbf{X} \\ \mathbf{\Upsilon} \end{bmatrix}. \quad (10)$$

2.3 Koopman analysis

DMD is connected to nonlinear systems via the Koopman operator (Mezić and Banaszuk, 2004; Mezić, 2005;

	Dynamics		Dynamics with control	
Linear	Dynamic mode decomposition (DMD) $\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k$		Dynamic mode decomposition with control (DMDC) $\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k$	
Operator	Koopman analysis $\mathcal{K}y(\mathbf{x}_k) = y(\mathbf{f}(\mathbf{x}_k)) = y(\mathbf{x}_{k+1})$		Koopman with inputs and control (KIC) $\mathcal{K}_*y(\mathbf{x}_k, \mathbf{u}_k) = y(\mathbf{f}(\mathbf{x}_k, \mathbf{u}_k), *) = y(\mathbf{x}_{k+1}, *)$	
Nonlinear	Sparse identification of nonlinear dynamics (SINDY) $\frac{d}{dt}\mathbf{x} = \mathbf{f}(\mathbf{x})$		SINDY with control (SINDYc) $\frac{d}{dt}\mathbf{x} = \mathbf{f}(\mathbf{x}, \mathbf{u})$	
	High-Dimensional Data	POD Coefficients	State Data	Input Data
			Regression Model	Regression Model with Control

Fig. 1. Overview of various methods that use regression to identify dynamics from data.

Rowley et al., 2009; Tu et al., 2014). The Koopman operator (Koopman, 1931) is an infinite-dimensional *linear* operator that describes how a measurement function $y(\mathbf{x})$ evolves through *nonlinear* dynamics:

$$\mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k). \quad (11)$$

The Koopman operator \mathcal{K} acts on the Hilbert space of scalar measurement functions $y(\mathbf{x})$ as:

$$\mathcal{K}y(\mathbf{x}_k) = y(\mathbf{f}(\mathbf{x}_k)) = y(\mathbf{x}_{k+1}). \quad (12)$$

That is, the Koopman operator acts on y by the composition of y with the dynamic update \mathbf{f} .

The DMD algorithm approximates the spectrum of the Koopman operator using linear observable functions (i.e., the observable functions are linear functions of the state, as in $y(\mathbf{x}_k) = \mathbf{x}_k$). However, it was recently shown that linear measurements are not sufficiently rich to analyze nonlinear systems (Williams et al., 2014), resulting in the extended DMD (eDMD), which performs a similar DMD regression, but on an augmented data matrix including nonlinear

state measurements. Since this algorithm is expensive numerically, a kernel trick was implemented to make the eDMD method as computationally efficient as standard DMD (Williams et al., 2015).

2.4 Koopman with inputs and control

Similar to how DMD was extended to include inputs and control, Koopman analysis has recently been extended to include inputs and control (Proctor et al., 2016b). In this Koopman with inputs and control (KIC) framework, scalar measurements of the state and control $y(\mathbf{x}, \mathbf{u})$ are advanced through nonlinear dynamics with control:

$$\mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k). \quad (13)$$

The Koopman with control operator \mathcal{K}_* is given by:

$$\mathcal{K}_*y(\mathbf{x}_k, \mathbf{u}_k) = y(\mathbf{f}(\mathbf{x}_k, \mathbf{u}_k), *) = y(\mathbf{x}_{k+1}, *). \quad (14)$$

It is important to note that there is a parameterized family of Koopman with control operators \mathcal{K}_* , as there is a choice of which future control input $*$ to use. It has been shown

that Koopman with inputs and control reduces to DMDC for linear dynamical systems, much as Koopman analysis is numerically computed using DMD for linear systems.

2.5 Sparse identification of nonlinear dynamics (SINDY)

The SINDY algorithm identifies fully nonlinear dynamical systems from measurement data. This relies on the fact that many dynamical systems have relatively few terms in the right hand side of the governing equations:

$$\frac{d}{dt}\mathbf{x} = \mathbf{f}(\mathbf{x}). \quad (15)$$

Given a library of candidate nonlinear functions,

$$\Theta^T(\mathbf{X}) = \begin{bmatrix} \text{---} & \mathbf{1} & \text{---} \\ \text{---} & \mathbf{X} & \text{---} \\ \text{---} & \mathbf{X}^2 & \text{---} \\ & \vdots & \\ \text{---} & \sin(\mathbf{X}) & \text{---} \\ \text{---} & \sin(2\mathbf{X}) & \text{---} \\ & \vdots & \end{bmatrix}, \quad (16)$$

where \mathbf{X} is the same data matrix as in Eq. (2), we may write our dynamical system as:

$$\dot{\mathbf{X}} = \Xi \Theta^T(\mathbf{X}). \quad (17)$$

The coefficients Ξ in this library are *sparse* for most dynamical systems. Therefore, we employ sparse regression to identify a sparse Ξ corresponding to the fewest nonlinearities in our library that give good model performance. Choosing a library of candidate dynamics is a crucial choice in the SINDY algorithm. The algorithm may be extended to include support for more general nonlinearities. It may also be possible to test different libraries (polynomials, trigonometric functions, etc.) and also incorporate partial knowledge of the physics (fluids vs. quantum mechanics, etc.).

Notice that if $\Theta^T(\mathbf{X}) = \mathbf{X}$, then Eq. (17) is equivalent to DMD with $\Xi = \mathbf{A}$. Each row of Eq. (17) represents a row in Eq. (15), and the sparse vector of coefficients ξ_k corresponding to the k -th row of Ξ is found using a sparse regression algorithm, such as LASSO (Tibshirani, 1996):

$$\xi_k = \operatorname{argmin}_{\xi_k} \|\dot{\mathbf{X}}_k - \xi_k \Theta^T(\mathbf{X})\|_2 + \alpha \|\xi_k\|_1, \quad (18)$$

where $\dot{\mathbf{X}}_k$ represents the k -th row of $\dot{\mathbf{X}}$. The $\|\cdot\|_1$ term promotes sparsity in the coefficient vector ξ_k . The parameter α is selected to identify the Pareto optimal model that best balances low model complexity with accuracy. A coarse sweep of α is performed to identify the rough order of magnitude where terms are eliminated and where error begins to increase. Then this parameter sweep may be refined.

To approximate derivatives from noisy state measurements, the SINDY algorithm uses the total variation regularized derivative (Rudin et al., 1992; Chartrand, 2011).

3. SPARSE IDENTIFICATION OF NONLINEAR DYNAMICS WITH CONTROL (SINDYc)

Here, we generalize the SINDY method to include inputs and control. In particular, we now consider the nonlinear dynamical system with inputs \mathbf{u} :

$$\frac{d}{dt}\mathbf{x} = \mathbf{f}(\mathbf{x}, \mathbf{u}). \quad (19)$$

The SINDY algorithm is readily generalized to include actuation, as this merely requires building a larger library $\Theta(\mathbf{x}, \mathbf{u})$ of candidate functions that include \mathbf{u} ; these functions can include nonlinear cross terms in \mathbf{x} and \mathbf{u} . This extension requires measurements of the state \mathbf{x} as well as the input signal \mathbf{u} . This generalization is shown in Fig. 1 in terms of the overarching regression framework.

If the signal \mathbf{u} corresponds to an external forcing, then we solve for the sparse coefficients Ξ in the following:

$$\dot{\mathbf{X}} = \Xi \Theta^T(\mathbf{X}, \mathbf{Y}). \quad (20)$$

However, if the signal \mathbf{u} corresponds to a feedback control signal, so that $\mathbf{u} = \mathbf{k}(\mathbf{x})$, then it is impossible to disambiguate the effect of the feedback control \mathbf{u} with internal feedback terms $\mathbf{k}(\mathbf{x})$ within the dynamical system; namely, the SINDY regression becomes ill-conditioned. In this case, we may identify the actuation \mathbf{u} as a function of the state:

$$\mathbf{Y} = \Xi_u \Theta^T(\mathbf{X}). \quad (21)$$

To identify the coefficients Ξ in Eq. (20), we perturb the signal \mathbf{u} to allow it to be distinguished from $\mathbf{k}(\mathbf{x})$ terms. This may be done by injecting a sufficiently large white noise signal, or occasionally kicking the system with a large impulse or step in \mathbf{u} . An interesting future direction would be to design input signals that *aid* in the identification of the dynamical system in Eq. (19) by perturbing the system in directions that yield high-value information.

4. EXAMPLE SYSTEMS

Here, we demonstrate the SINDY with control algorithm on a simple example predator-prey model with forcing and on the Lorenz equations with external forcing and control.

4.1 Predator-prey model

A predator-prey model with forcing is given by:

$$\dot{x}_1 = ax_1 - bx_1x_2 + u^2, \quad (22a)$$

$$\dot{x}_2 = -cx_2 + dx_1x_2 \quad (22b)$$

The variable x_1 represents the size of the prey population and x_2 represents the size of the predator population; the prey species is actuated with u^2 . The parameters a, b, c , and d represents the various growth/death rates, the effect of predation on the prey population, and the growth of predators based on the size of the prey population.

In this example, we force the system sinusoidally with $u(t) = 2\sin(t) + 2\sin(t/10)$, and the population response is shown in Fig. 2 (grey and black). The first 100 time units are used to train the SINDY and SINDYc algorithms,

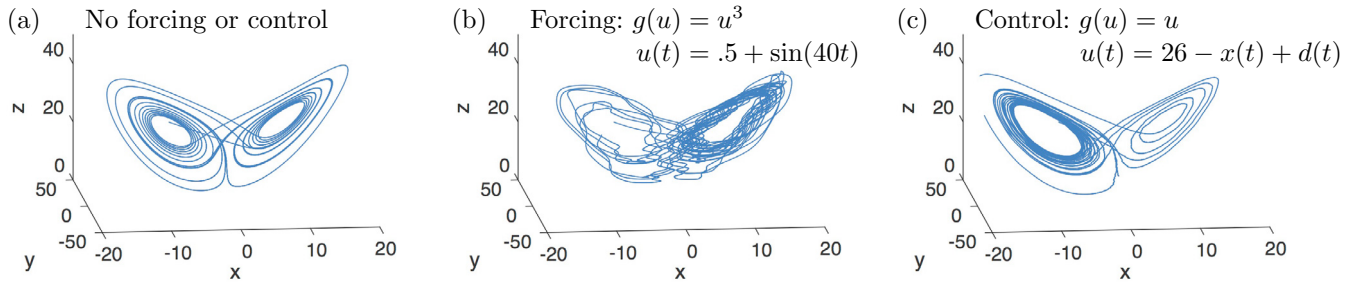


Fig. 3. Lorenz system without forcing or control (a), with forcing (b), and with feedback control (c). In each case, the system is integrated for 50 time units with a $\Delta t = .001$ with the parameters $\sigma = 10$, $\beta = 8/3$, and $\rho = 28$.

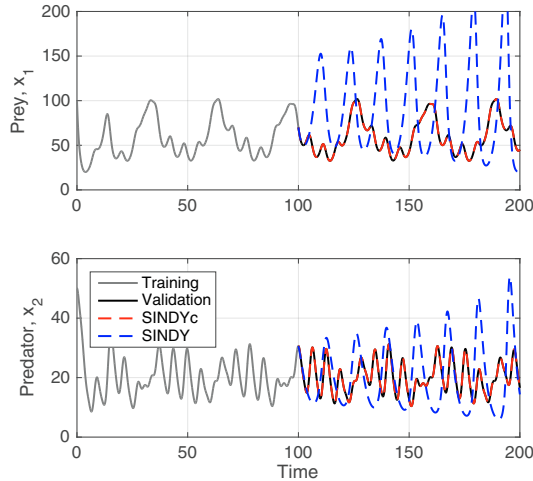


Fig. 2. SINDY and SINDYc model predictions for the force Lotka-Volterra system in Eq. (22). The training data consists of the Lotka-Volterra system with periodic forcing.

after which they are validated on the next 100 time units of forced data. The naive application of SINDY without knowledge of the input results in an unstable model (blue), while the SINDYc algorithm correctly identifies the model structure and parameters in Eq. (22) to within machine precision in the absence of measurement noise; the SINDYc reconstruction is shown in red.

4.2 Lorenz equations

We also test the SINDYc method on the Lorenz equations:

$$\dot{x} = \sigma(y - x) + g(u) \quad (23a)$$

$$\dot{y} = x(\rho - z) - y \quad (23b)$$

$$\dot{z} = xy - \beta z. \quad (23c)$$

These equations are examined with various forcing and control models, as shown in Fig. 3. In the case of an external forcing, as in Fig. 3 (b), the SINDYc algorithm correctly identifies the model and nonlinear input terms.

In the case that the Lorenz system is being actively controlled by state feedback, as in Fig. 3 (c), we must add a perturbation signal $d(t)$ to the input to disambiguate the effect of state feedback via u from internal dynamics. For this problem, we use an additive white noise process. In this example, we train the models using 20 time units of controlled data, and validate them on another 20 time units where we switch the forcing to a periodic signal $u(t) = 50 \sin(10t)$. The SINDY algorithm does not capture the effect of actuation, while SINDYc correctly identifies

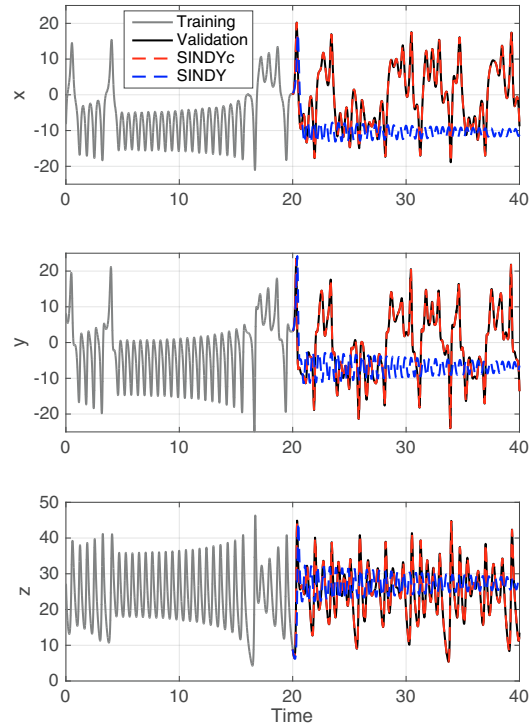


Fig. 4. SINDY and SINDYc predictions for the controlled Lorenz system in Eq. (23). Training data consists of the Lorenz system with state feedback as in Fig. 3 (c). After the training period, the input u switches to a periodic signal $u(t) = 50 \sin(10t)$.

the model and predicts the behavior in response to a new forcing that was not used in the training data.

5. DISCUSSION

In this work, we have generalized the sparse identification of nonlinear dynamics (SINDY) algorithm to include inputs and control. This involved generalizing the library of candidate nonlinear terms to include functions not only of the state \mathbf{x} , but also of the input \mathbf{u} , including cross terms between state and input. This new algorithm is cast in an overarching regression framework in Fig. 1, relating it to other algorithms that determine models from data, including dynamics mode decomposition (DMD), DMD with control, extended DMD, and Koopman analysis.

The new method has been tested on a predator prey model and the Lorenz system with various forcing and control models. The proposed algorithm should scale to the same class of problems where SINDY is useful, since they are built on the same computational architecture.

There are a number of interesting directions to extend this work. First, it is important to determine optimal strategies to disambiguate the effect of a state-feedback control signal from internal state dynamics; this may be achieved by additive white noise on the input signal or occasional kicks to the system, but understanding the tradeoffs and benefits of these strategies will be useful. More importantly, it is likely possible to design input sequences that optimally probe complex systems to extract high-value information that will be useful to characterize the system. For example, perhaps perturbing some systems off-attractor will provide valuable information about nonlinear terms in the dynamics if the on-attractor data may strongly resemble a linear system. If the state and control variables have different levels of sparsity, it may be possible to use a weighted convex optimization to penalize the state and control sparsity separately. It may also be important to improve the model identification if the control law $\mathbf{u} = \mathbf{k}(\mathbf{x})$ is known. These are promising areas of current and future research.

REFERENCES

- Baraniuk, R.G. (2007). Compressive sensing. *IEEE Signal Processing Magazine*, 24(4), 118–120.
- Berger, E., Sastuba, M., Vogt, D., Jung, B., and Amor, H.B. (2015). Estimation of perturbations in robotic behavior using dynamic mode decomposition. *J. Advanced Robotics*, 29(5), 331–343.
- Brunton, B.W., Johnson, L.A., Ojemann, J.G., and Kutz, J.N. (2016a). Extracting spatial-temporal coherent patterns in large-scale neural recordings using dynamic mode decomposition. *Journal of Neuroscience Methods*, 258, 1–15.
- Brunton, S.L. and Noack, B.R. (2015). Closed-loop turbulence control: Progress and challenges. *Applied Mechanics Reviews*, 67, 050801–1–050801–48.
- Brunton, S.L., Proctor, J.L., and Kutz, J.N. (2016b). Discovering governing equations from data by sparse identification of nonlinear dynamical systems. *Proc. Natl. Acad. Sci.*, 113(15), 3932–3937.
- Budišić, M., Mohr, R., and Mezić, I. (2012). Applied Koopmanism a). *Chaos*, 22(4), 047510.
- Candès, E.J. (2006). Compressive sensing. *Proc. International Congress of Mathematics*.
- Chartrand, R. (2011). Numerical differentiation of noisy, nonsmooth data. *ISRN Applied Mathematics*, 2011.
- Donoho, D.L. (2006). Compressed sensing. *IEEE Trans. Information Theory*, 52(4), 1289–1306.
- Erichson, N.B., Brunton, S.L., and Kutz, J.N. (2015). Compressed dynamic mode decomposition for real-time object detection. Preprint. Available: arXiv:1512.04205.
- Froyland, G. and Padberg, K. (2009). Almost-invariant sets and invariant manifolds – connecting probabilistic and geometric descriptions of coherent structures in flows. *Physica D*, 238, 1507–1523.
- Grošek, J. and Kutz, J.N. (2014). Dynamic mode decomposition for real-time background/foreground separation in video. Preprint. Available: arXiv:1404.7592.
- Holmes, P.J., Lumley, J.L., Berkooz, G., and Rowley, C.W. (2012). *Turbulence, coherent structures, dynamical systems and symmetry*. Cambridge Monographs in Mechanics. Cambridge University Press, Cambridge, England, 2nd edition.
- Holmes, P. and Guckenheimer, J. (1983). *Nonlinear oscillations, dynamical systems, and bifurcations of vector fields*, volume 42 of *Applied Mathematical Sciences*. Springer-Verlag, Berlin.
- Kaiser, E., Noack, B.R., Cordier, L., Spohn, A., Segond, M., Abel, M., Daviller, G., Osth, J., Krajnovic, S., and Niven, R.K. (2014). Cluster-based reduced-order modelling of a mixing layer. *J. Fluid Mech.*, 754, 365–414.
- Kevrekidis, I.G., Gear, C.W., Hyman, J.M., Kevrekidis, P.G., Runborg, O., and Theodoropoulos, C. (2003). Equation-free, coarse-grained multiscale computation: Enabling microscopic simulators to perform system-level analysis. *Communications in Mathematical Science*, 1(4), 715–762.
- Koopman, B.O. (1931). Hamiltonian systems and transformation in Hilbert space. *Proc. Natl. Acad. Sci.*, 17(5), 315–318.
- Koza, J.R., Bennett III, F.H., and Stiffelman, O. (1999). Genetic programming as a darwinian invention machine. In *Genetic Programming*, 93–108. Springer.
- Ljung, L. (1999). *System Identification: Theory for the User*. Prentice Hall.
- Mackey, A., Schaeffer, H., and Osher, S. (2014). On the compressive spectral method. *Multiscale Modeling & Simulation*, 12(4), 1800–1827.
- Majda, A.J., Franzke, C., and Crommelin, D. (2009). Normal forms for reduced stochastic climate models. *Proc. Natl. Acad. Sci.*, 106(10), 3649–3653.
- Majda, A.J. and Harlim, J. (2007). Information flow between subspaces of complex dynamical systems. *Proc. Natl. Acad. Sci.*, 104(23), 9558–9563.
- Mezić, I. (2005). Spectral properties of dynamical systems, model reduction and decompositions. *Nonlinear Dynamics*, 41(1-3), 309–325.
- Mezić, I. (2013). Analysis of fluid flows via spectral properties of the Koopman operator. *Ann. Rev. Fluid Mech.*, 45, 357–378.
- Mezić, I. and Banaszuk, A. (2004). Comparison of systems with complex behavior. *Physica D*, 197(1), 101–133.
- Ozoliņš, V., Lai, R., Cafisch, R., and Osher, S. (2013). Compressed modes for variational problems in mathematics and physics. *Proc. Natl. Acad. Sci.*, 110(46), 18368–18373.
- Proctor, J. and Echhoff, P. (2015). Discovering dynamic patterns from infectious disease data using dynamic mode decomposition. *International Health*, 7, 139–145.
- Proctor, J.L., Brunton, S.L., and Kutz, J.N. (2016a). Dynamic mode decomposition with control. *SIAM Journal on Applied Dynamical Systems*, 15(1), 142–161.
- Proctor, J.L., Brunton, S.L., and Kutz, J.N. (2016b). Generalizing Koopman theory to allow for inputs and control. *arXiv preprint arXiv:1510.03007*.
- Rowley, C.W., Mezić, I., Bagheri, S., Schlatter, P., and Henningson, D. (2009). Spectral analysis of nonlinear flows. *J. Fluid Mech.*, 645, 115–127.
- Rudin, L.I., Osher, S., and Fatemi, E. (1992). Nonlinear total variation based noise removal algorithms. *Physica D*, 60(1), 259–268.
- Sapsis, T.P. and Majda, A.J. (2013). Statistically accurate low-order models for uncertainty quantification in turbulent dynamical systems. *Proc. Natl. Acad. Sci.*, 110(34), 13705–13710.
- Schaeffer, H., Cafisch, R., Hauck, C.D., and Osher, S. (2013). Sparse dynamics for partial differential equations. *Proc. Natl. Acad. Sci.*, 110(17), 6634–6639.
- Schmid, P.J. (2010). Dynamic mode decomposition of numerical and experimental data. *Journal of Fluid Mechanics*, 656, 5–28.
- Schmid, P.J. and Sesterhenn, J. (2008). Dynamic mode decomposition of numerical and experimental data. In *61st Annual Meeting of the APS Division of Fluid Dynamics*.
- Schmidt, M. and Lipson, H. (2009). Distilling free-form natural laws from experimental data. *Science*, 324(5923), 81–85.
- Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *J. of the Royal Statistical Society B*, 267–288.
- Tu, J.H., Rowley, C.W., Luchtenburg, D.M., Brunton, S.L., and Kutz, J.N. (2014). On dynamic mode decomposition: theory and applications. *Journal of Computational Dynamics*, 1(2), 391–421.
- Wang, W.X., Yang, R., Lai, Y.C., Kovanis, V., and Grebogi, C. (2011). Predicting catastrophes in nonlinear dynamical systems by compressive sensing. *PRL*, 106, 154101.
- Williams, M.O., Kevrekidis, I.G., and Rowley, C.W. (2015). A data-driven approximation of the Koopman operator: extending dynamic mode decomposition. *J. Nonlin. Sci.*, 25(6), 1307–1346.
- Williams, M.O., Rowley, C.W., and Kevrekidis, I.G. (2014). A kernel approach to data-driven Koopman spectral analysis. *arXiv preprint arXiv:1411.2260*.