Q. Yang, C. A. Sing-Long, and E. J. Reed

View Online   Export Citation   CrossMark

ARTICLES YOU MAY BE INTERESTED IN

Reactive SINDy: Discovering governing reactions from concentration data
The Journal of Chemical Physics **150**, 025101 (2019); https://doi.org/10.1063/1.5066099

Combining machine learning with knowledge-based modeling for scalable forecasting and subgrid-scale closure of large, complex, spatiotemporal systems
Chaos: An Interdisciplinary Journal of Nonlinear Science **30**, 053111 (2020); https://doi.org/10.1063/5.0005541

Efficient reduction for diagnosing Hopf bifurcation in delay differential systems: Applications to cloud-rain models
Chaos: An Interdisciplinary Journal of Nonlinear Science **30**, 053130 (2020); https://doi.org/10.1063/5.0004697

Scilight
Summaries of the latest breakthroughs in the **physical sciences**

AIP Publishing

Rapid data-driven model reduction of nonlinear dynamical systems including chemical reaction networks using ℓ^1-regularization

Q. Yang,[1,a] 🔟 C. A. Sing-Long,[2] 🔟 and E. J. Reed[3] 🔟

AFFILIATIONS

[1] Computer Science and Engineering Department, University of Connecticut, Storrs, Connecticut 06269, USA
[2] Institute for Mathematical and Computational Engineering, Pontificia Universidad Católica de Chile, Santiago 7820436, Chile
[3] Department of Materials Science and Engineering, Stanford University, Stanford, California 94305, USA

[a] Author to whom correspondence should be addressed: qyang@uconn.edu

ABSTRACT

Large-scale nonlinear dynamical systems, such as models of atmospheric hydrodynamics, chemical reaction networks, and electronic circuits, often involve thousands or more interacting components. In order to identify key components in the complex dynamical system as well as to accelerate simulations, model reduction is often desirable. In this work, we develop a new data-driven method utilizing ℓ^1-regularization for model reduction of nonlinear dynamical systems, which involves minimal parameterization and has polynomial-time complexity, allowing it to easily handle large-scale systems with as many as thousands of components in a matter of minutes. A primary objective of our model reduction method is interpretability, that is to identify key components of the dynamical system that contribute to behaviors of interest, rather than just finding an efficient projection of the dynamical system onto lower dimensions. Our method produces a family of reduced models that exhibit a trade-off between model complexity and estimation error. We find empirically that our method chooses reduced models with good extrapolation properties, an important consideration in practical applications. The reduction and extrapolation performance of our method are illustrated by applications to the Lorenz model and chemical reaction rate equations, where performance is found to be competitive with or better than state-of-the-art approaches.

Published under license by AIP Publishing. https://doi.org/10.1063/1.5139463

We develop a new data-driven paradigm for efficient model reduction of a broad class of nonlinear dynamical systems. Our model reduction method directly enables the interpretation of key components of the dynamical system, unlike traditional projection-based model reduction methods that focus on reducing computational complexity more than interpretability. Our method is not application specific and is simple to implement on nonlinear dynamical systems arising from a variety of different fields. It requires minimal parameterization using a single parameter to trade-off between model complexity and estimation error. We use a data-driven paradigm to formulate model reduction as an efficient convex optimization problem that scales polynomially in the original size of the complex system, enabling systems with as many as thousands of components to be reduced in a matter of minutes.

I. INTRODUCTION

The numerical solution of nonlinear dynamical systems underlies many problems in computational science from biology[1] to atmospheric chemistry[2,3] to combustion.[4,5] An important challenge in this field is the ability to identify key components of the dynamical system that govern properties of the system of scientific interest. For example, in the case of chemical reaction networks, one might be interested in the time to equilibration and the equilibrium state given an initial chemical composition of the system, and the goal would be to find a small set of key reactions that are sufficient to characterize how the system evolves toward equilibrium. This is difficult to do for complex systems that may consist of thousands or more components.

In this work, we develop a new data-driven paradigm that enables the computationally efficient model reduction of a wide class

of nonlinear dynamical systems. Specifically, we focus on systems of the form

$$\frac{d\boldsymbol{x}(t)}{dt} := \sum_{j=1}^{N} f_j(\boldsymbol{x}(t))$$

$$= \sum_{j=1}^{N} h_j(\boldsymbol{x}(t))\boldsymbol{r}_j, \quad (1)$$

where $\boldsymbol{x}(t) \in \mathbb{R}^M$, $\boldsymbol{r}_j \in \mathbb{R}^M$ are constant, and $h_j : \mathbb{R}^M \to \mathbb{R}$ are possibly nonlinear. We identify every $f_j := h_j \boldsymbol{r}_j$ with an *interpretable* component of the system. The aim is to perform *model reduction*, that is to find a surrogate to (1) that consists of as few of these components as possible while still being able to reproduce the coarse-scale behavior of $\boldsymbol{x}$. Specifically, we focus on finding a surrogate dynamical system,

$$\frac{d\widehat{\boldsymbol{x}}(t)}{dt} = F(\widehat{\boldsymbol{x}}(t), \widehat{\boldsymbol{k}}) := \sum_{j=1}^{N} \widehat{k}_j f_j(\widehat{\boldsymbol{x}}(t)), \quad (2)$$

such that $\widehat{\boldsymbol{k}} \in \mathbb{R}^N$ is a *sparse* vector, namely, a vector such that most of its components are equal to zero. This surrogate system becomes the *reduced model* (or *system*) for (1).

Evidently, $\widehat{\boldsymbol{k}} = 0$ would be a maximally reduced system, but the time evolution of $\widehat{\boldsymbol{x}}$ would bear no resemblance to that of $\boldsymbol{x}$. For this reason, we require that

$$\ell(\boldsymbol{x}, \widehat{\boldsymbol{x}}) \le \epsilon \quad (3)$$

for some suitably defined loss function ℓ and prescribed $\epsilon > 0$. Consequently, model reduction is constrained by the trade-off between the *reduction* achieved, in terms of how many f_j can be neglected in (1), and the *error* incurred, as quantified by the loss function ℓ. Typically, the error will be minimal if $\widehat{\boldsymbol{k}} = \boldsymbol{k}$ and maximal if $\widehat{\boldsymbol{k}} = 0$. For instance, one choice of loss function is

$$\ell(\boldsymbol{x}, \widehat{\boldsymbol{x}}) = \int_0^T \|\boldsymbol{P}\boldsymbol{x}(t) - \boldsymbol{P}\widehat{\boldsymbol{x}}(t)\|_2^2 \, dt,$$

where $\boldsymbol{P}$ is an $m \times m$ projection matrix; in particular, this matrix may select the components of the state vector $\boldsymbol{x}$ that we consider important to reproduce. In this case, we are allowing possibly large errors in the nonimportant components in order to allow a large reduction with a small error in the important components.

In contrast to other model reduction approaches in the literature, such as proper orthogonal decomposition (POD),[6–8] we are not concerned primarily with finding a projection of the state vector $\boldsymbol{x}$ onto a low-dimensional space in order to reduce the number of components of the system and thus its computational complexity. Instead, we are interested in reducing the number of f_j in (1) until only those that are essential to reproduce the behavior of $\boldsymbol{x}$ over time remain. This allows us to interpret what components of the dynamical system are necessary to reproduce the behavior of $\boldsymbol{x}$, as opposed to which subspace on $\mathbb{R}^M$ contains the directions of the largest variability of $\boldsymbol{x}$ over time. It may be the case that by doing this, we also happen to find a reduction in the dimensionality of the relevant state space of $\boldsymbol{x}$, but this is a secondary concern. In particular, we may reduce the dimensionality of the state space if, by removing some of the f_j from the dynamical system, some components $\widehat{x}_i$ of the solution $\widehat{\boldsymbol{x}}$ to the reduced system are constant for $t > 0$. If these components are not considered important, for example, as defined

by the matrix $\boldsymbol{P}$ in the loss function above, then model reduction also leads to a reduction in computational cost.

We illustrate our method with two different types of dynamical systems. We start with a simple example using the widely known Lorenz equations[9] from chaos theory. We show that our model reduction method is not only able to find a reduced model of the Lorenz equations near their stationary point at the origin that corresponds to its linear approximation there but also able to identify a stable dynamical system as the optimal next smaller reduced model, rather than an unstable system that seems to reproduce the full model more closely near the origin but extrapolates poorly. Then, we follow with a larger scale problem in the form of chemical reaction networks described by the reaction rate equations,[10] where each f_j corresponds to a distinct chemical reaction. We would like to find a subset of chemical reactions that are capable of reproducing the molecular concentrations at time t of certain important species with as little error as possible. We study an 89 reaction network that has been used in the combustion community to describe propane pyrolysis and show that our method is not only able to identify a significantly reduced set of reactions that predicts the dynamics of six important molecular species with less than 5% error, but also that the reduced models extrapolate comparably or better over orders of magnitude in time than two previously determined reduced models computed using other methods.

The paper is organized as follows. We discuss background on model reduction for nonlinear dynamical systems in Sec. II, our new algorithm is presented in Sec. III, experimental results are discussed in Sec. IV, and the conclusions follow in Sec. V.

II. BACKGROUND

Model reduction of nonlinear dynamical systems has traditionally been aimed at choosing a subspace of dimension $m \ll M$ to project the state vector $\boldsymbol{x} \in \mathbb{R}^M$ onto this subspace at every instant. Either the singular value decomposition (SVD) or Krylov subspace methods play a key role in many of these methods.[11] This approach has the ability to speed up the computation of solutions to the dynamical system and can help determine the regions of state space that play an important role in the evolution of the dynamical system; in fact, the low-dimensional subspace is typically computed from the directions of large variability of the trajectory. However, projection-based methods are difficult to apply to nonlinear dynamical systems and is an active area of research.[12] They also do not naturally lead to interpretability about the significant components of a system in the sense of (1) since typically, the component functions f_j do not have a one-to-one correspondence to components of $\boldsymbol{x}$.

The ability to find significant components of a complex dynamical system is important from a scientific perspective in many practical applications. In particular, the combustion and biochemistry communities are often interested in the key chemical reactions that are drivers of a reaction or gene regulatory network. Thus, model reaction of chemical reaction networks has been extensively studied in both these communities, among others.[32–37] Popular methods that find reduced mechanisms consisting of a subset of reactions and thus preserve interpretability include sensitivity analysis,[13,14] graph-theoretic methods,[15] and integer programming.[16,17] For example, sensitivity analysis involves computing at various selected times

how the state variables (e.g., molecular concentrations) change with respect to changes in system parameters (e.g., reaction rate constants), building a sensitivity matrix. This sensitivity matrix is then analyzed using methods such as principal component analysis (PCA)[18] in order to find reactions that can be eliminated. The directed relation graph method[15] is another method that computes at selected snapshots of the system a graph that describes whether the rate of change of a particular state variable depends significantly on another state variable. The nodes of the graph are the state variables (e.g., molecular species) and there is a directed edge between nodes if there is a dependency between them. Given a set of state variables that we are interested in, we can then find all other state variables that may be relevant to their dynamics by searching over this graph for dependencies. Then, the reduced reaction model consists of all components (e.g., reactions) that contain the found set of state variables (e.g., molecular species).

These methods involve user expertise in choosing a set of times and system conditions under which to observe the system state and construct a sensitivity matrix or a directed relation graph, respectively, so that the reduced reaction networks consist of the union of reactions found over these different snapshots. They also involve user-specified thresholds to determine which components to retain in the reduced mechanisms, or what qualifies as a significant dependence between states, respectively, and different choices of thresholds can affect the ability to achieve optimal reduction. On the other hand, direct global optimization methods such as integer programming do not scale well and are computationally prohibitive for very large-scale systems.

In contrast, we have designed our algorithm to require minimal parameterization and polynomial-time computational complexity. In fact, we find empirically that the solution to our model reduction procedure is relatively insensitive over a wide range of values to the choice of parameters for all parameters except the one controlling the sparsity of the model. Our method uses ℓ^1-regularization and leverages the linearity in the components f_j in (1) to formulate model reduction as a convex optimization problem, allowing it to scale polynomially in N, the number of components f_j in the full system. Finally, our method is simple to implement and easily applied to any dynamical system in the form of (1). Its simple form gives rise to a number of straightforward variations that can be used to achieve different model reduction goals.

In recent years, several methods leveraging ℓ^1-regularization have been proposed in the literature to solve similar problems. However, the model reduction problem, as we have formulated it, has particular features that makes our method distinct from others proposed in the literature. In particular, it is these differences that preclude us from leveraging the extensive literature on ℓ^1-recovery methods to provide guarantees for our method.

- *Model reduction and model estimation:* In *model estimation* problems, we observe (possibly several) trajectories x_i of an unknown dynamical system. The aim is to estimate the f_j in (1) from these observed trajectories. This is achieved by choosing a collection of functions $\widehat{f_j}$, not necessarily of the form we have considered, and then using a fitting procedure to find a suitable $\widehat{k}$, e.g., by

minimizing

$$\sum_i \int_0^T \left\| \frac{dx_i(t)}{dt} - \sum_j \widehat{k_j}\widehat{f_j}(x_i(t)) \right\|_2^2 dt.$$

If the true f_j are part of the collection $\{\widehat{f_j}\}$, then we expect this procedure to find the unknown dynamical system. In recent years, SINDy[19,20] has been proposed as a method that leverages ℓ^1-regularization of the above loss function to estimate dynamical systems from a set of observed trajectories. Model reduction, as we have formulated it, assumes *full knowledge* of the dynamical system that we aim to reduce. In particular, we do *not* assume that there is any "true" reduced model, i.e., a sparse $\widehat{k}$ for which the error is zero. Instead, we focus on characterizing the trade-off between reduction and error. Although this is a conceptual difference, model estimation methods such as SINDy can be nonetheless applied for model reduction, for instance, by setting $\widehat{f_j} \equiv f_j$ for all j. However, in some multiscale systems, such as chemical reaction networks, where the reaction rate parameters may be several orders of magnitude different from each other, knowing the true model is critical to being able to learn a reduced model from the data. A naïve application of SINDy on chemical reaction networks would encounter challenges due to the vastly different scales of the reaction rates; our method does not have similar difficulties because it leverages the knowledge of the true dynamical system to scale the problem appropriately. This leads to improved reduced models, as we demonstrate in Sec. IV.

- *Model reduction and sparse recovery:* Suppose that there is an unknown sparse vector k_0 in a high-dimensional space representing an object of interest and that the only information we have about this vector is a set of $n \ll N$ measurements of the form $a_i^T k_0$ for some known vectors a_i. The aim of *sparse recovery* is to leverage the sparsity of k_0 to recover it from the n linear measurements available. Compressed sensing[21–23] characterizes the conditions on the sparsity of k_0 and the measurement vectors a_i that ensure we can recover k_0 exactly from this incomplete set of linear measurements. Although model estimation problems are sometimes successfully framed as a sparse recovery problems,[19,24] in our case it is not possible to do so. In principle, we could assume the sparse $\widehat{k}$ that achieves a loss of at most ϵ is unknown and that we observe a *noisy* version $k = 1$, where 1 denotes the vectors of all ones. From this point of view, we want to leverage the sparsity of $\widehat{k}$ to recover it from k. However, we have a *complete* set of *direct* measurements about $\widehat{k}$ and thus critical conditions on the measurements, such as incoherence[25] or the restricted isometry property (RIP),[26] do not hold. In fact, the model reduction problem we have formulated is closer to a denoising problem where we have observations of the form

$$k_j = \widehat{k_j} + w_j,$$

where w is the noise. In this formulation, the properties of w can be deduced from the constraint (3). Consequently, we cannot leverage the results of sparse recovery to provide guarantees to our method.

III. ALGORITHM

Many nonlinear dynamical systems can be expressed in the form (1) reproduced here

$$\frac{d\boldsymbol{x}(t)}{dt} = \sum_{j=1}^{N} h_j(\boldsymbol{x}(t))\boldsymbol{r}_j.$$

For example, if the dynamical system is defined by a polynomial function, then we can assign to each distinct term that appears a scalar function h_j multiplying a constant vector $\boldsymbol{r}_j$. Now, suppose that we are interested in reproducing the dynamics of a subset of state vector components x_i, $i \in \mathcal{C}$, where $\mathcal{C} \subset \{1, \ldots, M\}$ that are important, and we would like to know the smallest set of components f_j that are needed to do so. Ideally, we would like to find a vector $\widehat{\boldsymbol{k}} \in \mathbb{R}^N$ such that as many of the $\widehat{k}_j$ are 0 as possible while minimizing a loss function ℓ that quantifies the discrepancy between $\boldsymbol{x}$ and $\widehat{\boldsymbol{x}}$ over a time interval $t \in [t_a, t_b]$ of interest; here, $\widehat{\boldsymbol{x}}$ is the solution to the reduced model

$$\frac{d\widehat{\boldsymbol{x}}(t)}{dt} = \sum_{j=1}^{N} \widehat{k}_j h_j(\widehat{\boldsymbol{x}}(t))\boldsymbol{r}_j,$$

with initial condition $\widehat{\boldsymbol{x}}(0) = \boldsymbol{x}(0)$.

Since $\boldsymbol{x}$ and $\widehat{\boldsymbol{x}}$ are both solutions to nonlinear dynamical systems, directly minimizing any loss function ℓ that depends on the values of $\boldsymbol{x}$ and $\widehat{\boldsymbol{x}}$ over an interval $[t_a, t_b]$ of interest is computationally challenging, as evaluating $\widehat{\boldsymbol{x}}(t)$ for any $\widehat{\boldsymbol{k}}$ requires integrating a nonlinear ODE, and minimization methods typically involve doing this many times iteratively. Instead, we make the observation that if $d\boldsymbol{x}/dt = d\widehat{\boldsymbol{x}}/dt$, then it is clear that $\boldsymbol{x} = \widehat{\boldsymbol{x}}$ as well. Intuitively then, when $d\boldsymbol{x}/dt \approx d\widehat{\boldsymbol{x}}/dt$ we should have $\boldsymbol{x} \approx \widehat{\boldsymbol{x}}$. Thus, we develop a model reduction algorithm that seeks to minimize the error between $d\boldsymbol{x}/dt$ and $d\widehat{\boldsymbol{x}}/dt$ over a set of sampling times. The idea of minimizing an error on the derivatives can also be found in the literature on collocation methods for solving systems of ordinary differential equations (ODEs).[27]

We make the key observation that $d\widehat{\boldsymbol{x}}/dt$ varies linearly with respect to the vector $\widehat{\boldsymbol{k}} \in \mathbb{R}^N$ at any given time t. For any $\boldsymbol{x} \in \mathbb{R}^M$ and $\boldsymbol{k} \in \mathbb{R}^N$, we can write F in (2) in matrix form as

$$F(\boldsymbol{x}, \boldsymbol{k}) = \underbrace{[\boldsymbol{r}_1 \quad \ldots \quad \boldsymbol{r}_N]}_{:=R} \underbrace{\begin{bmatrix} h_1(\boldsymbol{x}) & \ldots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \ldots & h_N(\boldsymbol{x}) \end{bmatrix}}_{:=D(\boldsymbol{x})} \begin{bmatrix} k_1 \\ \vdots \\ k_N \end{bmatrix}$$

$$= \boldsymbol{R}D(\boldsymbol{x})\boldsymbol{k}.$$

To capture the nonlinearity of the system, we can then sample $\boldsymbol{x}$ at a set of times $t_1, \ldots, t_m$ and stack up all of the corresponding linear systems into the $mM \times N$ linear system

$$\begin{bmatrix} d\boldsymbol{x}(t_1)/dt \\ \vdots \\ d\boldsymbol{x}(t_m)/dt \end{bmatrix} \approx \begin{bmatrix} \boldsymbol{R}D(\boldsymbol{x}(t_1)) \\ \vdots \\ \boldsymbol{R}D(\boldsymbol{x}(t_m)) \end{bmatrix} \widehat{\boldsymbol{k}} \quad \Leftrightarrow \quad \dot{\boldsymbol{x}} \approx F\widehat{\boldsymbol{k}}. \quad (4)$$

This relation has the crucial property that $\dot{\boldsymbol{x}} = F\boldsymbol{1}$. We note that the sampling times $t_1, \ldots, t_m$ that we use to build our data-driven model reduction problem have some effect on the quality of the final reduced model. In our example problems, we choose the sampling

times to be evenly spaced in $t \in [t_a, t_b]$ for a specified sample size m. The reduced models that we find, while fairly robust to the sample size m on our example problems, could in general depend on the density of samples in certain regions where the derivative changes rapidly. It is usually sufficient to use as many samples as are available from the numerical solution of the original ODE.

After setting up the linear system (4), we take the additional step of weighting each *row* of F by its ℓ^2-norm. This can be done by multiplying both F and $\dot{\boldsymbol{x}}$ by a diagonal matrix W whose r-th diagonal entry is equal to the reciprocal of the ℓ^2-norm of the r-th row of F. We do this as a normalization step to guard against differences in the scale of different dx_i/dt. We do not want to ignore perturbations to the rate of change of particular state variables simply because their rate of change is smaller in magnitude than that of others, because especially in a nonlinear system, these small changes may still have a large impact on the overall dynamics of the system. We note that other choices of W are possible to achieve this effect, for example, weighting rows by their ℓ^1-norm. We use the ℓ^2-norm because it has the additional effect of contributing to better numerical stability when solving (5).

Now, we solve the weighted system for sparse $\widehat{\boldsymbol{k}}$, subject to a constraint on the ℓ^1-norm of $\widehat{\boldsymbol{k}}$ in order to induce sparsity in the solution. We add two additional inequality constraints to bound $\widehat{\boldsymbol{k}}$. This leads us to a ℓ^1-regularization problem with box constraints

$$\begin{aligned} \underset{\widehat{\boldsymbol{k}} \in \mathbb{R}^N}{\text{minimize}} \quad & \|WF\widehat{\boldsymbol{k}} - W\dot{\boldsymbol{x}}\|_2 \\ \text{subject to} \quad & \|\widehat{\boldsymbol{k}}\|_1 \leq \lambda, \\ & 0 \leq \widehat{\boldsymbol{k}} \leq \kappa\boldsymbol{1}, \end{aligned} \quad (5)$$

where vector inequalities are interpreted componentwise and $\kappa \geq 1$. The inequality constraints bounding the components of $\widehat{\boldsymbol{k}}$ are intended to promote a solution that is close to that of the integer program. In the special case of chemical reaction networks, as we will discuss below, the lower bound $\widehat{\boldsymbol{k}} \geq 0$ also enforces a physical constraint, namely, that reaction rates must be nonnegative. If we choose $\kappa = 1$, our problem is exactly the convex relaxation of the corresponding integer program where the second constraint in (5) is replaced by $\widehat{\boldsymbol{k}} \in \{0, 1\}^N$. Note that the upper bound $\widehat{\boldsymbol{k}} \leq \kappa\boldsymbol{1}$ is only relevant when $\kappa \leq \lambda$. In practice, we find that for large λ, choosing $\kappa = 10$ or $\kappa = 100$ allows for some added flexibility in finding reduced models where some of the remaining components f_j now need to be weighted more heavily in F. In general, however, the overall solution is not too sensitive to different choices of κ and any improvements from an optimal choice are typically small.

Our problem bears a strong resemblance to the nonnegative garrote[28,29] that was an early inspiration for the popular least absolute shrinkage and selection operator (LASSO) method,[30] which has become an important tool for finding sparse models. By varying λ, we find a family of potential reduced models. The solution to the quadratic program above results in a vector $\widehat{\boldsymbol{k}}$ for which some entries are 0; its sparsity depends on the choice of λ. We use this solution $\widehat{\boldsymbol{k}}$ only for model selection to determine which components to retain, since a well-known property of ℓ^1-regularization is that it tends to underestimate the magnitudes of $\widehat{k}_j$; this phenomenon is known as the shrinkage effect in the statistics literature. While we could choose to only retain components that correspond

to nonzero $\widehat{k}_j$, empirically for λ such that we can find a reasonable reduced model, $\widehat{k}_j$ exhibit a bimodal distribution, so that either $\widehat{k}_j \approx 0$ or $\widehat{k}_j \in [10^{-2}, 10^2]$ (that is, the constrained solution does not vary from the unconstrained solution of $\widehat{k} = 1$, by more than two orders of magnitude). Thus, we retain all components j such that $\widehat{k}_j \geq \epsilon$, where due to the strong separation of the modes the particular choice of ϵ is not very important, but generally $\epsilon = 0.01$ can be used. Finally, we debias the solution by resolving the weighted linear system

$$\underset{\widehat{k}^*}{\text{minimize}} \quad \|WF\widehat{k}^* - W\dot{x}\|_2$$
$$\text{subject to} \quad \mathbf{supp}(\widehat{k}^*) \subset \mathbf{supp}(\widehat{k}),$$
$$0 \leq \widehat{k}^* \leq \kappa \mathbf{1},$$

where $\widehat{k}$ is the optimal solution to (5) and **supp** denotes the support. Therefore, the above problems solves a least squares problem restricting the support of the variable to be that of the optimal solution to (5).

IV. EXPERIMENTAL RESULTS

We demonstrate our method on two different types of nonlinear dynamical systems of the form (1) described above: the Lorenz equations and chemical reaction networks. In both cases, we study the resulting reduced models for their extrapolation capability and other properties of practical interest.

In the following discussions, we evaluate the quality of the reduced model by the error on a subset of important state variables with indices in $\mathcal{C}$. For the Lorenz model, there are only three dimensions and we are interested in all three state variables. In the chemical reaction network example, we choose a small subset of important molecular species, and note that the best reduced model found may be affected by the choice of $\mathcal{C}$. We use the following expression for the error between the full and reduced models (relative norm error), which for comparison purposes is the same as that used by Hannemann-Tamás et al.,[17]

$$\frac{1}{m|\mathcal{C}|} \sum_{i \in \mathcal{C}} \sum_{j=1}^{m} \frac{|\hat{x}_i(t_j) - x_i(t_j)|}{\max(x_i(t_j), \text{Tol}_i)}, \quad (6)$$

where Tol_i is a tolerance parameter used to prevent large relative errors as the denominator decreases to zero. In Hannemann-Tamás et al.[17] as well as in this work, we use $\text{Tol}_i = 10^{-4} \times \max_j x_i(t_j)$. We note that this error metric is different from the objective function of our optimization problem (5), which we have constructed as a surrogate for the error. The empirical relationship between the objective error and the relative norm error is explored in more detail in the experiments below.

We refer to the smallest model with no more than 5% relative norm error as the best reduced model. In general, there is a sharp increase in the error after the best reduced model as we increase λ to try to find smaller models, so again the choice of 5% relative error as a threshold is not a very sensitive parameter.

In all experiments, we choose $\kappa = 100$ and $\epsilon = 0.01$, although as noted above the results are not too sensitive to these parameter choices. We vary λ over a range of values to find a family of reduced models of different sizes, characterizing the trade-off between reduction and error. Note that our problem is scaled so that λ is a rough approximation for the number of components in the reduced model.

A. Lorenz equations

We begin by considering the Lorenz equations as a small illustrative example. The Lorenz system is a nonlinear system of ordinary differential equations that is famous for exhibiting chaotic behavior under certain parameter values and initial conditions. The original model developed by Edward Lorenz in 1963[9] is as follows:

$$\frac{dx}{dt} = \sigma(y - x),$$
$$\frac{dy}{dt} = x(\rho - z) - y,$$
$$\frac{dz}{dt} = xy - \beta z.$$

Lorenz used the parameter values $\sigma = 10$, $\rho = 28$, and $\beta = 8/3$ to demonstrate an example of when the system exhibits chaotic behavior.

The Lorenz system is a dynamical system of the form (1) where h_j are given by the five different terms: x, y, z, xy, and xz. Note that we can rewrite the system as

$$\begin{bmatrix} dx(t)/dt \\ dy(t)/dt \\ dz(t)/dt \end{bmatrix}$$

$$= \underbrace{\begin{bmatrix} -\sigma & \sigma & 0 & 0 & 0 \\ \rho & -1 & 0 & 0 & -1 \\ 0 & 0 & -\beta & 1 & 0 \end{bmatrix}}_{:=R} \mathbf{diag}\underbrace{\left(\begin{bmatrix} x(t) \\ y(t) \\ z(t) \\ x(t)y(t) \\ x(t)z(t) \end{bmatrix}\right)}_{:=D(x(t),y(t),z(t))} \underbrace{\begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}}_{:=\mathbf{k}}.$$

The origin is a fixed point of the system for all values of σ, ρ, and β. We test the ability of our method to find a reduced model at this fixed point. We initialize the system close to the origin and study the system for $t \in [0, 0.1]$ so that the system remains in a regime where the values of x, y, and z are small. We experimented with many different randomly chosen initial conditions close to the origin and find in almost all cases that the smallest reduced model with less than 5% relative norm error is given by

$$\widehat{k} = [1 \quad 1 \quad 1 \quad 0 \quad 0]^t.$$

We plot the results for one such test in Fig. 1 with initial conditions $x(0) = 0.2415$, $y(0) = 0.1434$, and $z(0) = 0.3260$. The reduced models correspond to $\lambda \in [5, 3, 1, 0.5, 0.05]$, which result in reduced models with 5, 4, 3, 2, and 1 terms, respectively. Note that the smallest reduced model before a large growth in error corresponds to omitting the nonlinear xy and xz terms from the Lorenz equations. Our method has thus automatically detected that a good reduced model for the Lorenz equations close to the origin is its linear approximation!

There are a few interesting observations from this experiment. First, our model reduction method does indeed find the correct
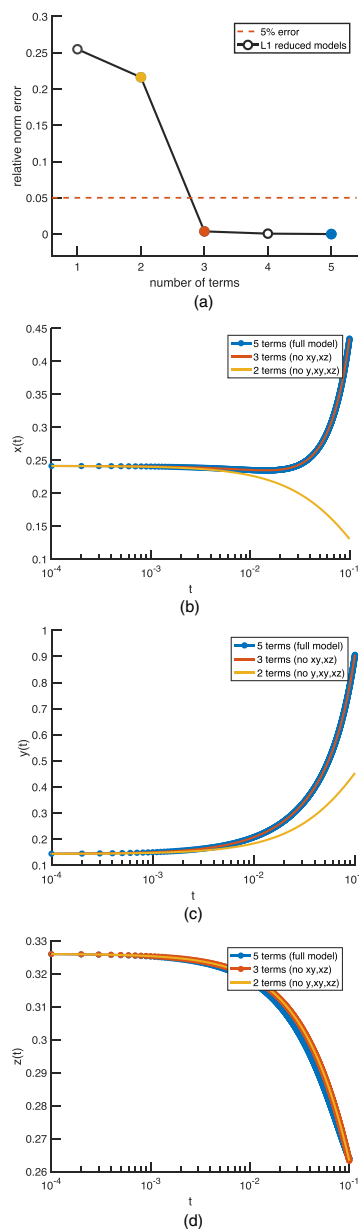
FIG. 1. We apply our model reduction method to the Lorenz equations close to the origin: (a) the smallest reduced model before a large growth in error corresponds to a 3-term model that omits the nonlinear xy and xz terms. This agrees with the fact that the Lorenz equations can be approximated by its linearization close to the origin. (b)–(d) We plot $x(t)$, $y(t)$, and $z(t)$ corresponding to the full model and different reduced models. The full model is plotted in blue. The smallest model with reasonable error, which has both nonlinear terms xy and xz removed, is plotted in red. Additionally, the next best reduced model chosen by our method contains just the x and z terms and is plotted in yellow. It is globally stable for positive values of the parameters and thus has bounded extrapolation error. Our model reduction method preferred this model to a model containing the x and y terms that has better training error but is unstable and has exponentially growing extrapolation error.

smallest reduced model consistently, and its ability to do so is not particularly sensitive to the initial conditions as long as they are close to the origin. Second, we notice that the next term that is removed from the system by our model reduction method is not the z term, which might have been expected given that in the 3-term linear model z has become separated from x and y, so that close to the origin the additional error incurred by simply setting $z \equiv z(0)$ is small. Instead, it is the y term that is removed first. If we compare these two 2-term models, the model that removed the z term first has much lower training error over $t \in [0, 0.1]$ (see Table I). This can make it seem that the 2-term reduced model found by our method is not the "optimal" 2-term model. However, on closer inspection, it turns out that when extrapolating the system to longer times, the error grows more slowly for the reduced model that our method found.

We can see this by directly computing the analytical solution of the two reduced models. In a 2-term model that only contains x and y terms, the analytical solution is

$$\begin{bmatrix} x(t) \\ y(t) \end{bmatrix} = e^{At} \begin{bmatrix} x(0) \\ y(0) \end{bmatrix},$$

where

$$A = \begin{bmatrix} -\sigma & \sigma \\ \rho & -1 \end{bmatrix} \begin{bmatrix} k_1 & 0 \\ 0 & k_2 \end{bmatrix}.$$

For our choices of parameters, $\sigma = 10$ and $\rho = 28$, we see that the magnitude of the eigenvalues of A is much larger than 1, and thus the solution is unstable and will lead to exponential growth in extrapolation error from the solution of the true dynamical system. On the other hand, in a 2-term model that only contains x and z terms, the analytical solution works out to

$$x(t) = x(0)e^{-k_1\sigma t},$$

$$y(t) = y(0) + x(0)\frac{\rho}{\sigma}(1 - e^{-k_1\sigma t}),$$

$$z(t) = z(0)e^{-k_3\beta t}.$$

Then, it is clear that as $t \to \infty$, the system approaches a constant: $x(t), z(t) \to 0$, and $y(t) \to y(0) + \rho x(0)/\sigma$. For positive values of σ and β, this is a globally stable system and has bounded extrapolation error, even though it seemed to have much worse training error. In Table I, we list how the error grows for the three possible 2-term models. For each model, we use the training data to solve for the debiased $\widehat{k}^*$ as described above. We note that the 2-term model that contains only the y and z terms has the same structure as the

TABLE I. Relative norm error of 2-term reduced models for the Lorenz system when extrapolated in time.

2-Term reduced model	Training err	$t \in [0, 1]$	$t \in [0, 10]$	$t \in [0, 100]$
x, z (ℓ^1 **model**)	0.216	911	$2.36 \times 10^{+03}$	$1.26 \times 10^{+03}$
x, y	0.0424	$5.96 \times 10^{+05}$	$4.33 \times 10^{+50}$	∞
y, z	0.251	904	$2.47 \times 10^{+03}$	$1.36 \times 10^{+03}$

2-term model with only the x and z terms; the only difference is in the constants.

B. Chemical reaction networks

Model reduction of chemical reaction networks is an important problem in the combustion literature. Chemical reaction networks can be described by the reaction rate equations (RREs), which are a deterministic set of ordinary differential equations that give the concentration of chemical species continuously over time.[10] The RRE is applicable to chemical systems that obey the law of mass action, under constant pressure and temperature conditions, and is a nonlinear dynamical system of the form (1).

Consider a chemical system with M molecular species and N distinct reactions. Let

$$\boldsymbol{x}(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \\ \vdots \\ x_M(t) \end{bmatrix}$$

be the vector of concentrations of each molecular species at time t. Then, the RRE states that the instantaneous rate of change in the concentration x_i of the i-th molecular species can be expressed as the sum of the changes caused by each reaction in the system for which the i-th molecular species is a reactant or product. The effect of each reaction is proportional to the concentrations of the reactants as well as the stoichiometry of the molecular species in the reaction. This can be summarized in vector notation as

$$\frac{d\boldsymbol{x}(t)}{dt} = \sum_{j=1}^{N} a_j(\boldsymbol{x}(t))\boldsymbol{r}_j. \tag{7}$$

The column vectors $\boldsymbol{r}_j \in \mathbb{R}^M$ correspond to the stoichiometry of reaction j, so that entries are negative if they correspond to reactant species and positive if they correspond to product species. Each a_j is a function proportional to the concentration at time t of the reactants in reaction j. For example, if reaction j is unimolecular with species $i \to$ products, then $a_j(\boldsymbol{x}) = k_j x_i(t)$ where k_j is a reaction-specific rate constant. If reaction j is bimolecular between species i' and i'' so that $i' + i'' \to$ products, then $a_j(\boldsymbol{x}) = k_j x_{i'} x_{i''}$.

Since the functions a_j are nonlinear in $\boldsymbol{x}$ for any reaction that is not unimolecular, the RRE for most chemical reaction networks is a nonlinear dynamical system. However, we can factor out the rate constants k_j from the nonlinear functions a_j to rewrite (7) as

$$\frac{d\boldsymbol{x}(t)}{dt} = \sum_{j=1}^{N} k_j h_j(\boldsymbol{x}(t))\boldsymbol{r}_j.$$

Consider the toy chemical reaction network

$$A + B \to C,$$

$$A \to 2C,$$

$$2C \to A.$$
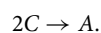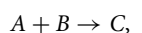
The corresponding reaction rate equations for this toy system are

$$\frac{dA(t)}{dt} = -k_1 A(t)B(t) - k_2 A(t) + k_3 C(t)^2,$$

$$\frac{dB(t)}{dt} = -k_1 A(t)B(t),$$

$$\frac{dC(t)}{dt} = k_1 A(t)B(t) + 2k_2 A(t) - 2k_3 C(t)^2.$$

At any given time t, we can write the rate of change in concentration as a linear function of the vector of reaction rate constants $\boldsymbol{k}$,

$$\frac{d\boldsymbol{x}(t)}{dt} = \begin{bmatrix} dA(t)/dt \\ dB(t)/dt \\ dC(t)/dt \end{bmatrix}$$

$$= k_1 \begin{bmatrix} -1 \\ -1 \\ 1 \end{bmatrix} A(t)B(t) + k_2 \begin{bmatrix} -1 \\ 0 \\ 2 \end{bmatrix} A(t) + k_3 \begin{bmatrix} 1 \\ 0 \\ -2 \end{bmatrix} C(t)^2$$

$$= \underbrace{\begin{bmatrix} -1 & -1 & 1 \\ -1 & 0 & 0 \\ 1 & 2 & -2 \end{bmatrix}}_{:=\boldsymbol{R}} \underbrace{\begin{bmatrix} A(t)B(t) & 0 & 0 \\ 0 & A(t) & 0 \\ 0 & 0 & C(t)^2 \end{bmatrix}}_{:=\boldsymbol{D}(A(t),B(t),C(t))} \underbrace{\begin{bmatrix} k_1 \\ k_2 \\ k_3 \end{bmatrix}}_{:=\boldsymbol{k}}.$$

After choosing m sampling times, we follow the same argument in (4) to obtain a linear system $\dot{\boldsymbol{x}} = \boldsymbol{F}\boldsymbol{k}$. Then, we take the crucial step of first scaling the coefficient matrix in that linear system by the known vector $\boldsymbol{k}$ that parameterizes the full chemical reaction network. In other words, we perform the scaling

$$\boldsymbol{F} \to \boldsymbol{F}\operatorname{diag}(\boldsymbol{k})$$

to solve (5) with this rescaled matrix; the key property is that after rescaling we have $\boldsymbol{F}\mathbf{1} = \dot{\boldsymbol{x}}$. This step is important because the true reaction rates $\boldsymbol{k}$ can span many orders of magnitude, and we would like our ℓ^1-regularization constraint to apply to each reaction uniformly.

We use a medium-sized reaction mechanism to demonstrate the efficiency of our data-driven approach to model reduction. The mechanism describes propane pyrolysis and involves effectively 89 reactions and 32 molecular species (originally 98 reactions but 9 reactions are eliminated since they have reactants that have no formation steps and are not initially present in the system). We compare our reduction method to two works that have studied the same system previously. This network was studied by Turanyi in 1990,[18] where model reduction was done by using principal components analysis on the rate sensitivity matrix. Turanyi's method first identified important and then necessary species in an iterative manner. Then, in 2013, this system was revisited by Hannemann-Tamás et al., who used mixed-integer quadratic programming (MIQP).[17] Both of these methods require careful parameterization. The MIQP method was also too computationally expensive to use directly and required a prefiltering step that first reduced the model to fewer than 40 reactions. In contrast, our method has a single significant parameter, λ, to trade-off between sparsity and error and scales polynomially with the number of reactions in the full mechanism. In Fig. 2, the plotted reduced models correspond to 41 evenly spaced
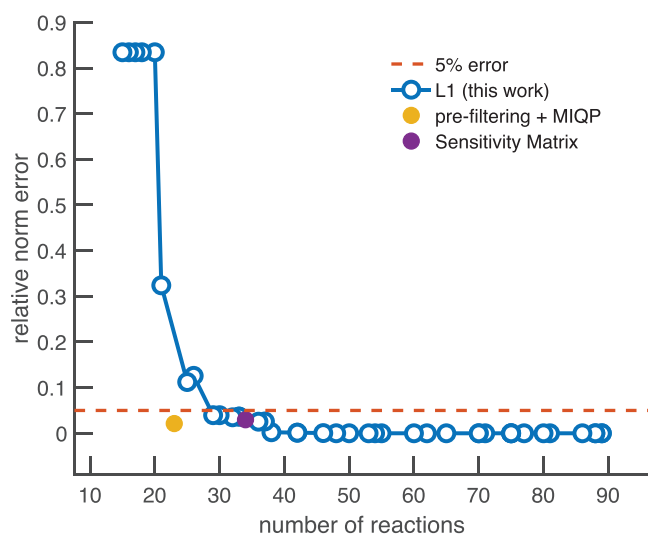
FIG. 2. Our ℓ^1-regularization method is able to find a model at the 5% error threshold that is close to the size of the models found by both the sensitivity matrix and the prefiltering + MIQP methods, at around 30 reactions out of a full network of 89 reactions. We note that the ℓ^1-regularization method is simpler to calculate and scales better than the sensitivity matrix and prefiltering + MIQP methods, and does not require any prereduction steps or manual adjustment of threshold parameters.

FIG. 3. Reducing the number of reactions in the network also leads to reduction in the number of molecular species involved in those reactions. The ℓ^1-regularization method finds reaction networks that involve 23 species, even as the number of reactions decreases.

λ between 10 and 90. We see that our method is able to find a model at the 5% error threshold that is close to the size of the models found by both the sensitivity matrix and prefiltering + MIQP methods at around 30 reactions, a two-thirds reduction in the network.

Reducing the number of reactions in the network also leads to a reduction in the number of molecular species involved in those reactions. From Fig. 3, we see that our method finds reaction networks that consistently involve 23 species, even as the number of reactions decreases. Both the sensitivity matrix and prefiltering + MIQP methods find reduced models involving fewer species.

We note that the prefiltering + MIQP and the sensitivity matrix methods determine quite a different set of reactions from the ℓ^1-regularization method. The prefiltering + MIQP model finds 23 reactions; these 23 reactions happen to be a subset of the 34 reactions found by the Sensitivity Analysis method. By contrast, there are many differences between these models and the ℓ^1-models of approximately the same size and relative norm error.

In the following discussion, we use for illustration the 29-reaction model found by ℓ^1-regularization, which is the smallest model with less than 5% relative error found by the ℓ^1-regularization method. There are 19 reactions that appear in both models. The four reactions that appear in the 23-reaction prefiltering + MIQP model but not in the 29-reaction ℓ^1-model are given in Table II. Note that 1-C_3H_7 indicates a particular isomer of C_3H_7 given by locant 1.

On the other hand, the ten reactions listed in Table III appear in the 29-reaction ℓ^1-regularization model but not the 23-reaction prefiltering + MIQP model. Many of them were reduced in the prefiltering step before solving the MIQP.
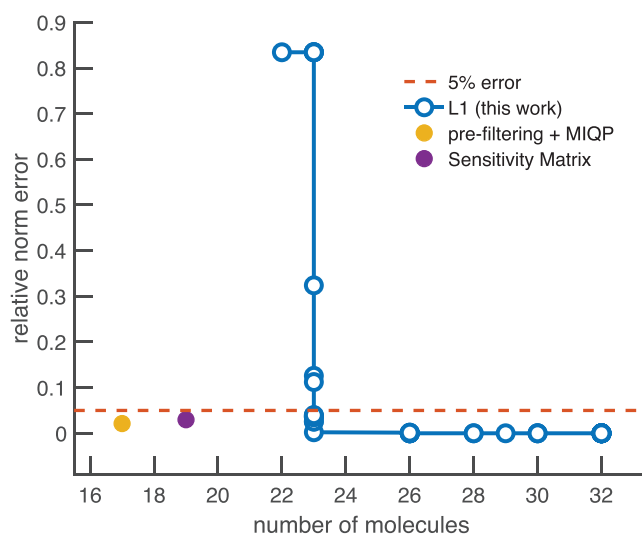
One important and interesting point illustrated by the above is that when evaluating reduced models, especially for large chemical reaction networks where only a small number of important molecular species are tracked, there may be many possible "reduced networks" satisfying a particular error threshold. Our ultimate choice of reduced model may be based on a number of factors. For example, we may simply want the model that results in the smallest state space. In this case, we might, for example, choose the prefiltering + MIQP model given above. However, usually we are interested in using the reduced model in regions it was not trained on. In that case, we may want to choose the model that has the best extrapolation properties, whether in time or initial state. Crucially, our analysis shows that reduced models that at first glance may seem to "miss" well-known chemical mechanisms do not necessarily do worse at extrapolating.

We see in Fig. 4 that both the sensitivity matrix and prefiltering + MIQP models seem optimized for a particular period of time—the MIQP model is highly optimized for the time period it was trained on, while the sensitivity matrix model is optimized for about an order of magnitude longer. We choose two comparable models determined by our ℓ^1-regularization method with around the same number of reactions and approximately the same relative norm error. We see that the smaller ℓ^1-reduced model extrapolates

**TABLE II.** Reactions only in the prefiltering + MIQP model.

$$C_3H_8 + 2{-}C_3H_7 \rightarrow 1{-}C_3H_7 + C_3H_8$$
$$C_3H_8 + C_3H_5 \rightarrow 1{-}C_3H_7 + C_3H_6$$
$$C_3H_6 + CH_3 \rightarrow C_3H_5 + CH_4$$
$$2{-}C_3H_7 + 2{-}C_3H_7 \rightarrow C_3H_6 + C_3H_8$$

TABLE III. Reactions only in the ℓ^1 model.

$$1-C_6H_{12} \rightarrow 1-C_3H_7 + C_3H_5$$
$$H + C_3H_6 \rightarrow 2-C_3H_7$$
$$CH_3 + CH_3 \rightarrow C_2H_6$$
$$CH_3 + 1-C_3H_7 \rightarrow n-C_4H_{10}$$
$$C_2H_5 + 2-C_3H_7 \rightarrow 2-\text{methylbutane}$$
$$1-C_3H_7 + 2-C_3H_7 \rightarrow 2-\text{methylpentane}$$
$$1-C_3H_7 + C_3H_5 \rightarrow 1-C_6H_{12}$$
$$2-C_3H_7 + 2-C_3H_7 \rightarrow \text{product}$$
$$C_2H_5 + C_2H_5 \rightarrow n-C_4H_{10}$$
$$C_2H_5 + 2-C_4H_9 \rightarrow 2-C_4H_8 + C_2H_6$$

to two orders of magnitude longer in time better than the sensitivity matrix model despite containing fewer reactions. The two ℓ^1-reduced models have comparable extrapolation error to the prefiltering + MIQP model, although we note that they do not grow in error until after an order of magnitude in time. Note that propane pyrolysis, the chemical process being modeled in this reaction network, reaches completion after about 10^6 s.

Furthermore, we note that our reduced models can be less sensitive to perturbations in initial conditions than both the sensitivity matrix and prefiltering + MIQP models. The original initial conditions are that there is only propane in the system (1.912×10^{-3} mol dm^{-3} initial propane concentration). In Fig. 4, we see that
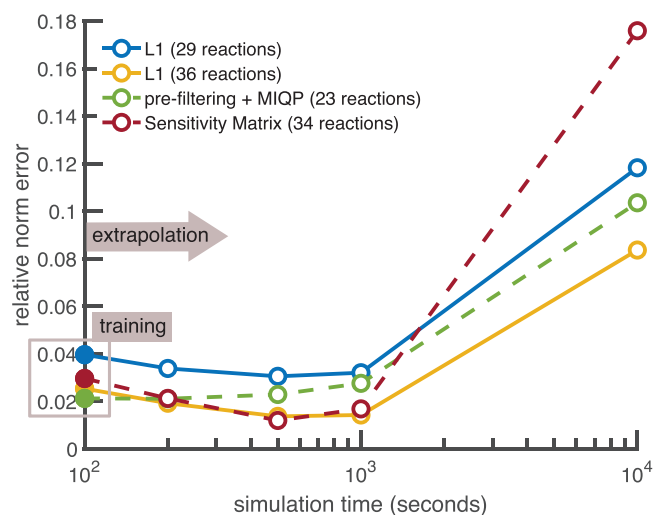
FIG. 4. Predictive power over time of four reduced reaction networks at the 5% error threshold. All reduced models were trained from $t = 0$ to $t = 100$ s. We see that both the sensitivity matrix and the prefiltering + MIQP models seem optimized for a particular period of time—the prefiltering + MIQP model is highly optimized for the time period it was trained on, while the sensitivity matrix model is optimized for about an order of magnitude longer. The ℓ^1-regularization method models extrapolate to longer time better than the sensitivity matrix model despite containing fewer reactions. They have comparable extrapolation error to the prefiltering + MIQP model. Note that propane pyrolysis, the chemical process being modeled in this reaction network, reaches completion after about 10^6 s.
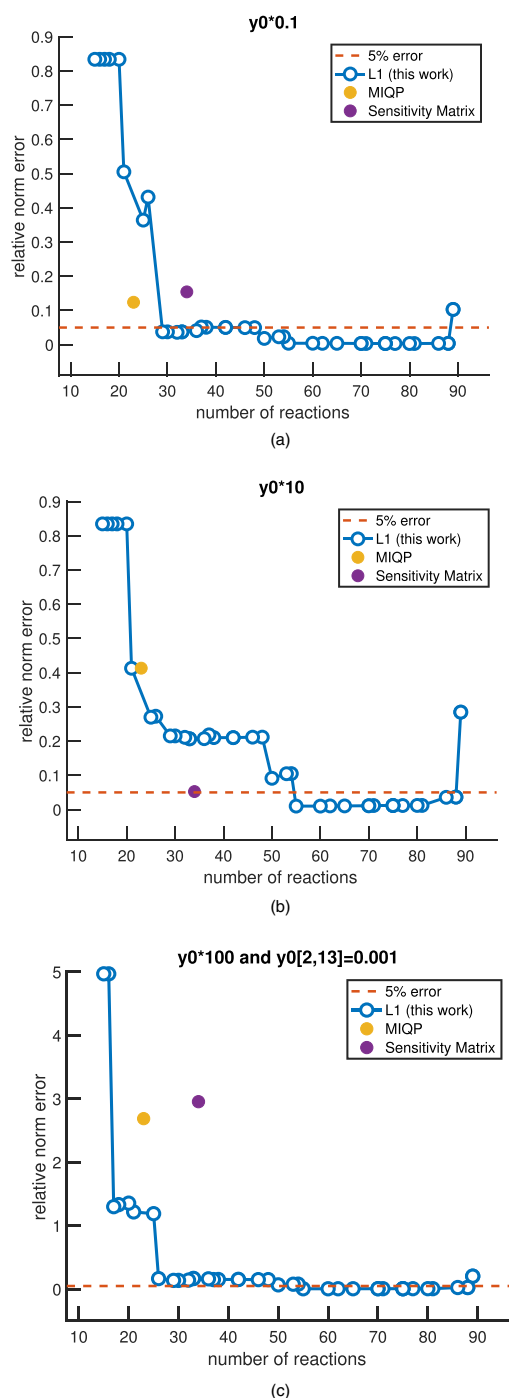
if we decrease the initial propane by a factor of 10, our best reduced models at around 30 reactions trained on the original initial conditions have lower error than both the other models. If, on the other hand, we increase the initial propane by a factor of 10, the sensitivity matrix model is best, but our models are still comparatively better than the prefiltering + MIQP model. Finally, if we change the initial concentration to have two orders of magnitude more propane and also include a little bit of CH_3 and H_2, a practical possibility in combustion systems, all reduced models are much worse than before, but the MIQP and sensitivity matrix models both exhibit catastrophic failure, whereas our models remain within reasonable error (Fig. 5).

To provide empirical intuition for how our model reduction algorithm finds a family of reduced models that exhibits approximately decreasing relative norm error with increasing model size even though the objective function in (5) never directly minimizes the relative norm error, we show in Fig. 6 that the relative norm error (6) is well correlated with the objective error of our model reduction algorithm when we choose the set of significant species $i \in \mathcal{C}$ to be the 6 physically relevant ones (propane and the main products of the propane pyrolysis[18]). However, if we look at the relative norm error overall species, there is a subset of reduced models for which the relative norm error is roughly correlated and then the error plateaus. This is because reduced models necessarily eliminate some of the species that contributes a constant value to the relative norm error.

Since our model reduction method is formulated as a convex optimization problem, specifically quadratic programs, it is fast and can in practice handle reaction networks with thousands of reactions.[31] Due to the computational efficiency of our algorithm, we can compute a family of reduced models that describe propane pyrolysis over a range of temperatures. The size of the resulting models is plotted in Fig. 7. By construction, the reduced models that we find represent an upper bound on the total number of reactions necessary to capture the dynamics of the key molecular species at the given temperature. We can therefore draw a convex hull below the datapoints to reveal the approximate rate of growth in the complexity of the chemical reaction network as a function of temperature.

C. Comparison with SINDy

As noted previously, several methods leveraging ℓ^1-regularization have been proposed in the recent literature to solve similar problems, perhaps most notably SINDy for discovering governing equations from data.[19] We discussed above how a naïve application of SINDy is insufficient for multiscale systems such as chemical reaction networks, where the parameters of interest may vary across many orders of magnitude and also have physical constraints. Here, we demonstrate applying SINDy to model reduction of our propane pyrolysis reaction network, with a few modifications. First, we add the additional physical constraint that $\widehat{k} \geq 0$, since no reaction rates can be negative. Second, in addition to normalizing the columns of F as recommended by the authors of SINDy, we also normalize the rows of F as discussed above to handle differences in scale of the derivatives of the x_i. Finally, we use $\epsilon = 0.01$ as the threshold for retaining components j such that $\widehat{k}_j \geq \epsilon$ as above, noting that without using an ϵ threshold the model is barely reduced before the error

FIG. 5. Model error when extrapolating to perturbed initial conditions: (a) reducing initial propane by a factor of 10, (b) increasing initial propane by a factor of 10, and (c) increasing initial propane by a factor of 100, and also adding H_2 and CH_3 into the system. Our best reduced models around 30 reactions extrapolate better to perturbed initial conditions than the slightly smaller prefiltering + MIQP models, and except in (b), also better than the sensitivity matrix model.
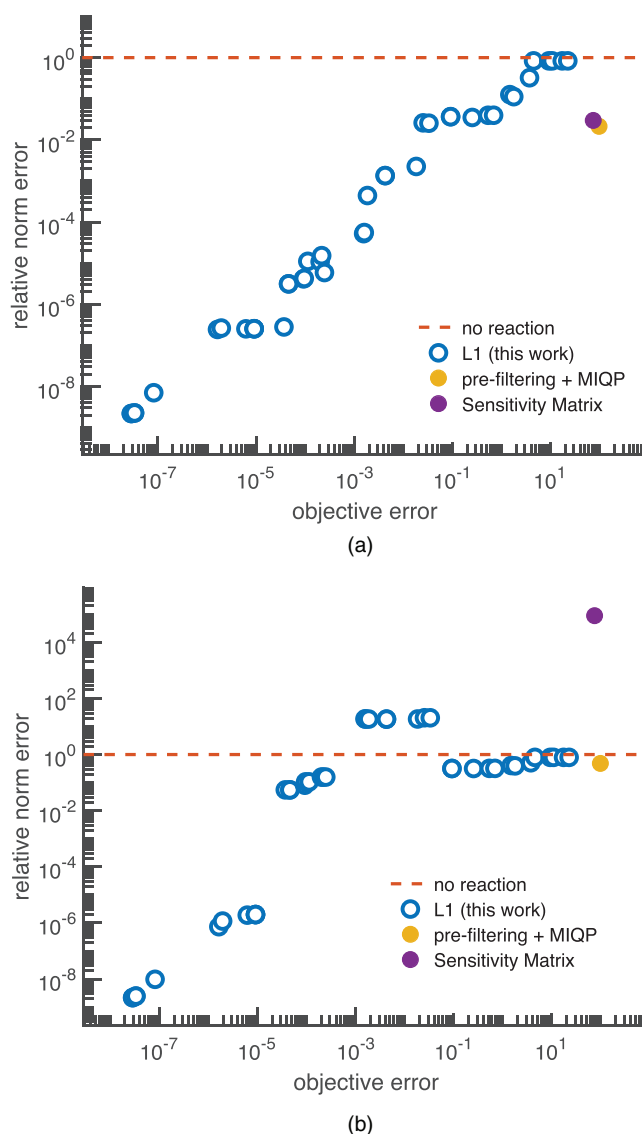

FIG. 6. (a) The relative norm error is well correlated with the objective error of our model reduction algorithm when we choose the set of significant molecular species $i \in \mathcal{C}$ to be the 6 physically relevant ones. (b) Note, however, that if we look at the relative norm error over all species, there is a subset of reduced models for which the relative norm error is roughly correlated, and then the relative norm error plateaus. This is because reduced models necessarily eliminate some of the species, contributing a large error that overwhelms the rest.

grows out of a reasonable range. The key differences of this SINDy-type problem from our proposed method are that: (i) we do not first scale the problem by the true reaction rate parameters, since it is assumed these are not known *a priori*; and (ii) we do not debias the solution. It is important to note that the row normalization and ϵ thresholding, which are **not** in the original proposed SINDy method,
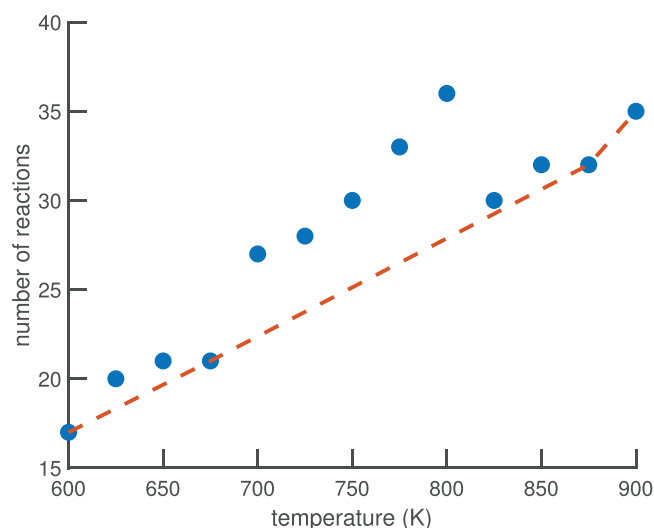
FIG. 7. Fewest number of reactions needed to model propane pyrolysis over 100 s with less than 5% error, as temperature is varied from 600 K to 900 K. The red line corresponds to a convex hull of the data, since the reduced models found using the ℓ^1-regularization method are an upper bound on complexity.

are key to finding a reasonable reduced model for the dynamical system. Even with these additional adjustments to SINDy, however, we find as shown in Fig. 8 that the results are unsatisfactory for finding meaningfully reduced reaction networks.
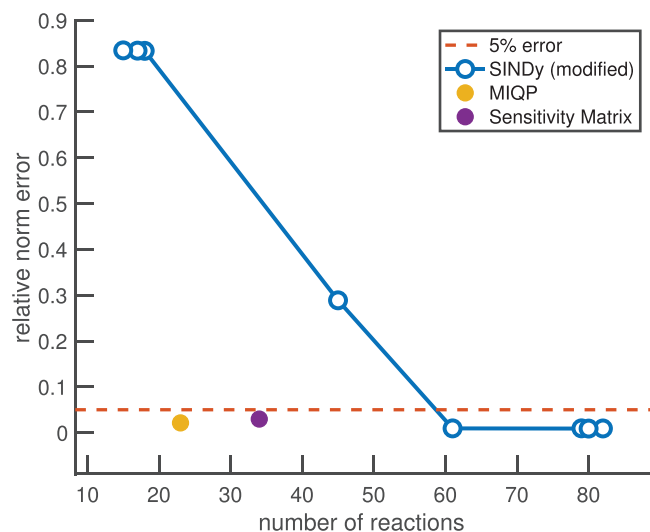
FIG. 8. Results of SINDy on reducing the propane pyrolysis network, after modifications for physical constraints, normalizing rows and columns of F, and thresholding $\hat{k}$. Even with these modifications, SINDy fails to find competitively small reduced models with low error, in contrast to our proposed method.

D. Discussion of validation and test error

We note that in the traditional context of data-driven methods and machine learning, choosing a linear model learned using sampled datapoints would require a validation set, and afterwards the chosen model should be evaluated using a test set. However, this is not necessary in our case because we can compute a true error, for example, the relative norm error given in (6), since we know the exact full model. One might also be interested in whether our reduced model extrapolates to regions it was not trained on. In that sense we would like to know if our model can extrapolate in time or to different initial states (noting that extrapolation in time is equivalent to initializing at a different state along the trajectory); this can also be determined exactly by comparing with the full model. Furthermore, from an application point of view, it may not be necessary or desirable to extrapolate too far away from the trained dataset, since it would not intuitively make sense that a reduced model for propane pyrolysis would be the same relevant reduced model for, for example, methane pyrolysis. As we saw in the family of temperature-dependent reduced models for propane pyrolysis above, as the temperature increased, the size of the best reduced model tended to increase as well. However, in order to effectively use our reduced models as part of multiscale codes, we would like to find models that have stable error under a reasonable amount of extrapolation. The requirements will be different for every application. For a given desired range of extrapolation in initial states, one possible solution is to use a combination of data from several trajectories with different initial conditions to construct our linear system (4) and find reduced models that are applicable across them all. In that case, the prediction error for new initial conditions can be estimated as typically done for data-driven learning problems, by using a collection of test trajectories to measure the expected performance of the model for new problems.

V. CONCLUSIONS

In this work, we have developed an efficient method for model reduction of nonlinear dynamical systems of the form (1) that is capable of reducing systems with thousands of components in a matter of minutes. The key to our method is exploiting the linearity of the dynamics with respect to the components f_j of interest. Then, we use data sampled from the solution of the full dynamical system to capture a series of linear systems that describe the instantaneous rate of change of the system. We use these samples to find a sparse set of linear coefficients, such that as many coefficients are zero as possible so that the corresponding components can be eliminated from the system, while preserving the dynamics of key state variables x_i, $i \in \mathcal{C}$, in the system. To induce sparsity in a computationally efficient way, we use ℓ^1-regularization. We show that our method works in finding a reduced model of the Lorenz equations near the origin that corresponds to its linear approximation. We also find that this method is able to quickly and with minimal parameterization find a significantly reduced chemical reaction network for modeling propane pyrolysis, and furthermore that the reduced model maintains a low approximation error for several orders of magnitude in time and extrapolates better to perturbed initial conditions than comparable methods. The computational efficiency of our method

enables us to study how the complexity of chemical reaction networks changes as parameters such as temperature are varied. We hope that our method will find use for a wide range of dynamical systems, from combustion to biochemistry, atmospheric chemistry, and even very-large-scale integration circuit design, and reduce the amount of time and human expertise required to develop reduced models.

ACKNOWLEDGMENTS

DATA AVAILABILITY

The data that support the findings of this study are available from the corresponding author upon reasonable request.

REFERENCES

[1] O. Radulescu, A. N. Gorban, A. Zinovyev, and V. Noel, "Reduction of dynamical biochemical reactions networks in computational biology," Front. Genet. **3**, 131 (2012).

[2] M. E. Jenkin, L. A. Watson, S. R. Utembe, and D. E. Shallcross, "A common representative intermediates (CRI) mechanism for VOC degradation. Part 1: Gas phase mechanism development," Atmos. Environ. **42**, 7185–7195 (2008).

[3] P. S. J. Lakey, A. Wisthaler, T. Berkemeier, T. Mikoviny, U. Pöschl, and M. Shiraiwa, "Chemical kinetics of multiphase reactions between ozone and human skin lipids: Implications for indoor air quality and health effects," Indoor Air **27**, 816–828 (2017).

[4] T. Lu, C. K. Law, and J. H. Chen, "Development of non-stiff reduced mechanisms for direct numerical simulations," AIAA No. 2008-1010, 2008.

[5] T. Lu and C. K. Law, "Toward accommodating realistic fuel chemistry in large-scale computations," Prog. Energy Combust. Sci. **35**, 192–215 (2009).

[6] P. Holmes, J. L. Lumley, and G. Berkooz, *Turbulence, Coherent Structures, Dynamical Systems and Symmetry* (Cambridge University Press, New York, 1996), p. 402.

[7] J. L. Lumley, *Stochastic Tools in Turbulence* (Academic Press, London 1970).

[8] B. R. Noack, K. Afanasiev, M. Morzyński, G. Tadmor, and F. Thiele, "A hierarchy of low-dimensional models for the transient and post-transient cylinder wake," J. Fluid Mech. **497**, 335–363 (2003).

[9] E. N. Lorenz, "Deterministic nonperiodic flow," J. Atmosp. Sci. **20**, 130–141 (1963).

[10] D. J. Higham, "Modeling and simulating chemical reactions," SIAM Rev. **50**, 347–368 (2008).

[11] A. C. Antoulas, *Approximation of Large-Scale Dynamical Systems* (Society for Industrial and Applied Mathematics, Philadelphia, PA, 2005).

[12] K. Carlberg, C. Farhat, J. Cortial, and D. Amsallem, "The GNAT method for nonlinear model reduction: Effective implementation and application to computational fluid dynamics and turbulent flows," J. Comput. Phys. **242**, 623–647 (2013).

[13] D. Edelson and D. L. Allara, "A computational analysis of the alkane pyrolysis mechanism: Sensitivity analysis of individual reaction steps," Int. J. Chem. Kinet. **XII**, 605–621 (1980).

[14] T. Turányi, T. Bérces, and S. Vajda, "Reaction rate analysis of complex kinetic systems," Int. J. Chem. Kinet. **21**, 83–99 (1989).

[15] T. Lu and C. K. Law, "A directed relation graph method for mechanism reduction," Proc. Combust. Inst. **30**, 1333–1341 (2005).

[16] L. Petzold and W. Zhu, "Model reduction for chemical kinetics: An optimization approach," AIChE J. **45**, 869–886 (1999).

[17] R. Hannemann-Tamás, A. Gábor, G. Szederkényi, and K. M. Hangos, "Model complexity reduction of chemical reaction networks using mixed-integer quadratic programming," Comput. Math. Appl. **65**, 1575–1595 (2013).

[18] T. Turanyi, "Reduction of large reaction mechanisms," New J. Chem. **14**, 795–803 (1990).

[19] S. L. Brunton, J. L. Proctor, and J. N. Kutz, "Discovering governing equations from data: Sparse identification of nonlinear dynamical systems," Proc. Natl. Acad. Sci. U.S.A. **113**, 3932–3937 (2016).

[20] G. Tran and R. Ward, "Exact recovery of chaotic systems from highly corrupted data," Multiscale Model. Simul. **15**, 1108–1129 (2017).

[21] E. J. Candes and J. K. Romberg, "Signal recovery from random projections," Proc. SPIE **5674**, 76–86 (2005). .

[22] D. L. Donoho, "Compressed sensing," IEEE Trans. Inf. Theory **52**, 1289–1306 (2006).

[23] E. Candes, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," IEEE Trans. Inf. Theory **52**, 489–509 (2006), arXiv:0409186 [math].

[24] S. L. Brunton, J. L. Proctor, J. H. Tu, and J. Nathan Kutz, "Compressed sensing and dynamic mode decomposition," J. Comput. Dyn. **2**, 165–191 (2015).

[25] E. Candès and J. Romberg, "Sparsity and incoherence in compressive sampling," Inverse Probl. **23**, 969–985 (2007).

[26] E. J. Candès, "The restricted isometry property and its implications for compressed sensing," C. R. Math. **346**, 589–592 (2008).

[27] J. M. Varah, "A spline least squares method for numerical parameter estimation in differential equations," SIAM J. Sci. Stat. Comput. **3**, 28–46 (1982).

[28] L. Breiman, "Better subset regression using the nonnegative garrote," Technometrics **37**, 373–384 (1995).

[29] M. Yuan and Y. Lin, "On the non-negative garrotte estimator," J. R. Stat. Soc. B **69**, 143–161 (2007).

[30] R. Tibshirani, "Regression shrinkage and selection via the Lasso," J. R. Stat. Soc. B **58**, 267–288 (1996).

[31] Q. Yang, C. A. Sing-Long, and E. J. Reed, "Learning reduced kinetic monte carlo models of complex chemistry from molecular dynamics," Chem. Sci. **8**, 5781–5796 (2017).

[32] M. Mincheva and M. R. Roussel, "Graph-theoretic methods for the analysis of chemical and biochemical networks. I. Multistability and oscillations in ordinary differential equation models," J. Math. Biol. **55**, 61–86 (2007).

[33] E. Chiavazzo, C. W. Gear, C. J. Dsilva, N. Rabin, and I. G. Kevrekidis, "Reduced models in chemical kinetics via nonlinear data-mining," Processes **2**, 112–140 (2014).

[34] M. Frenklach, H. Wang, and M. J. Rabinowitz, "Optimization and analysis of large chemical kinetic mechanisms using the solution mapping method—Combustion of methane," Prog. Energy Combust. Sci. **18**, 47–73 (1992).

[35] T. Turányi and T. Nagy, "Reduction of very large reaction mechanisms using methods based on simulation error minimization," Combust. Flame **156**, 417–428 (2009).

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

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