Data-based modeling and control of nonlinear process systems using sparse identification: an overview of recent results

Fahim Abdullah [a], Zhe Wu [b] and Panagiotis D. Christofides [a,c,1]

[a] Department of Chemical and Biomolecular Engineering, University of California, Los Angeles, CA 90095, USA

[b] Department of Chemical and Biomolecular Engineering, National University of Singapore, 117585, Singapore

[c] Department of Electrical and Computer Engineering, University of California, Los Angeles, CA 90095, USA

Abstract

This paper discusses recent developments in the data-based modeling and control of nonlinear chemical process systems using sparse identification of nonlinear dynamics (SINDy). SINDy is a recent nonlinear system identification technique that uses only measurement data to identify the physical laws governing the system in the form of first-order nonlinear differential equations. In this work, the challenges of handling time-scale multiplicities and noisy sensor data when using SINDy are addressed. When applied to two-time-scale systems, to overcome model stiffness, which leads to ill-conditioned controllers, a reduced-order modeling approach is proposed where SINDy is used to model the slow dynamics, and nonlinear principal component analysis is used to algebraically "slave" the fast states to the slow states. The resulting model can then be used in a Lyapunov-based model predictive controller with guaranteed closed-loop stability provided the separation of fast and slow dynamics is sufficiently large. To handle high levels of sensor noise, SINDy is combined with subsampling and co-teaching to improve modeling accuracy. Finally, the challenges of modeling and controlling large-scale systems using noisy industrial data are addressed by using ensemble learning with SINDy.

Keywords

nonlinear processes, sparse identification, chemical processes, subsampling, two-time-scale processes, singular perturbations, co-teaching, dropout, ensemble learning, model predictive control.

Introduction

A central objective of scientific and engineering research is the derivation of the laws governing physical systems in the form of equations. With the explosion in data and computational power over the last two decades, the construction of these equations empirically from data has become more tractable than deriving physics-based first-principles models, especially for highly complex systems, and is gaining momentum in the literature. For many physical systems, the laws governing their dynamics take the form of ordinary differential equations (ODE) or partial differential equations (PDE) with time and/or space as independent variables. Common examples include the Boltzmann equation in thermodynamics and the Navier–Stokes equations in fluid dynamics [1]. The development of such time-series predictive models is often a prerequisite for other objectives in a plant/system engineering context, such as predictive maintenance in operations engineering and advanced control system design in any closed-loop system with strict product requirements. In chemical process systems, model predictive control (MPC) is an advanced control system that has been implemented and accepted widely in industry [2]. As the name suggests, MPC uses a dynamical model such as an ODE to predict the process states and outputs over a user-defined prediction horizon to be able to take the optimal control action based on anticipated possible future trajectories. A large body of literature on data-driven modeling in MPC can be found in [3]. Two of the most common, classical system identification algorithms include singular value decomposition [4] and Numerical algorithms for Subspace State Space System Identification (N4SID) [5]. However, machine learning methods, a type of data-driven modeling with numerous tunable hyperparameters, have demonstrated highly accurate results when applied to complex systems with multiple interacting nonlinearities due to their high degree of freedom. Some examples of machine learning methods include support vector regressors, artificial neural networks, and sparse identification [6], the last of which is the highlight of this article.

In this manuscript, we apply SINDy to model and

[1] Corresponding author. Email: pdc@seas.ucla.edu.

control three types of process systems: 1) processes with time-scale multiplicities, 2) simulated processes with high levels of sensor noise, and 3) large-scale processes corrupted with high levels of industrial noise. Each category of systems has associated challenges and are addressed using different improvements upon the original SINDy algorithm, the details of which will be discussed in the respective section.

Class of Nonlinear Process Systems

We consider the class of nonlinear process systems described by the following first-order ODE:

$$\dot{x}(t) = f(x) + g(x)u + w, \qquad x(t_0) = x_0 \tag{1}$$

where $x \in \mathbb{R}^n$ is the state vector, $u \in \mathbb{R}^r$ is the manipulated input vector, and $w \in \mathbb{R}^n$ is the noise vector. The unknown vector and matrix functions $f \in \mathbb{R}^n$ and $g \in \mathbb{R}^{n\times r}$, respectively, constitute the process model representing the inherent physical laws constraining the system and are assumed to be locally Lipschitz vector and matrix functions of their arguments with $f(0) = 0$. The manipulated input is constricted to be in r nonempty convex sets defined as $\mathcal{U}_i \subseteq \mathbb{R}, i = 1, \ldots, r$. The sensor noise w is assumed to be bounded within the set $W := w \in \mathbb{R}^n : \|w\|_2 \leq \theta, \theta > 0$. The class of systems of the form of eq. (1) is further restricted to the family of stabilizable nonlinear systems, i.e., there exist a sufficiently smooth control Lyapunov function $V(x)$ and a control law $\Phi(x) = [\Phi_1(x) \cdots \Phi_r(x)]^\top$ that renders the nominal ($w \equiv 0$) closed-loop system of eq. (1) asymptotically stable under $u = \Phi(x)$. The stability region Ω_ρ is defined as the largest level set of V where $\dot{V}$ is negative. Without loss of generality, the initial time t_0 is taken to be 0 throughout the article.

Overview of sparse identification of nonlinear dynamics (SINDy)

Based on sparse regression and compressive sensing, sparse identification of nonlinear dynamics (SINDy) is a novel method in the field of system identification and has been applied to a diverse array of engineering problems [7, 8]. The aim of SINDy is to use only input/output data from a system to represent the dynamics in the form of the nominal system of eq. (1),

$$\dot{\hat{x}}(t) = \hat{f}(\hat{x}) + \hat{g}(\hat{x})u \tag{2}$$

where $\hat{x} \in \mathbb{R}^n$ is the state vector of the sparse-identified model, and $\hat{f}$ and $\hat{g}$ are the model parameters that capture the physical laws governing the system.

Since most physical systems contain only a few terms in the right-hand side of eq. (2), if a large number of nonlinear basis functions are considered as possible terms in $\hat{f}$ and $\hat{g}$, the space of all candidate functions considered is rendered sparse. Hence, SINDy aims to identify the small number of active functions in $\hat{f}$ and $\hat{g}$ using algorithms that leverage sparsity. We first sample a discrete set of full-state measurements from open-loop simulations or experiments and concatenate them into a data matrix X and input matrix U,

$$X = \begin{bmatrix} x_1(t_1) & x_2(t_1) & \cdots & x_n(t_1) \\ x_1(t_2) & x_2(t_2) & \cdots & x_n(t_2) \\ \vdots & \vdots & \ddots & \vdots \\ x_1(t_m) & x_2(t_m) & \cdots & x_n(t_m) \end{bmatrix} \tag{3a}$$

$$U = \begin{bmatrix} u_1(t_1) & u_2(t_1) & \cdots & u_r(t_1) \\ u_1(t_2) & u_2(t_2) & \cdots & u_r(t_2) \\ \vdots & \vdots & \ddots & \vdots \\ u_1(t_m) & u_2(t_m) & \cdots & u_r(t_m) \end{bmatrix} \tag{3b}$$

where $x_i(t_\ell)$ and $u_j(t_\ell)$ represent the measurement of the i^{th} state and j^{th} input at the ℓ^{th} sampling time, respectively, where $i = 1, \ldots, n$, $j = 1, \ldots, r$, and $\ell = 1, \ldots, m$. $\dot{X}$, the time-derivative of X, is a required matrix in the sparse identification algorithm and is either measured if possible (e.g., velocity) or estimated from X. Subsequently, a function library $\Theta(X, U)$ is constructed with s nonlinear functions of X and U. These s functions are the candidate nonlinear functions that may be zero or nonzero in the right-hand side of eq. (2). The sparse identification algorithm exploits sparsity to calculate the coefficients associated with the terms in the library, Θ. Given the universality of mononomials, polynomials, and trigonometric functions in engineering systems [6], they are selected as the initial library in Θ. An example of an augmented library is

$$\Theta(X, U) = \begin{bmatrix} | & | & | & | & | & | \\ \mathbf{1} & X & \sin(X) & \mathrm{e}^X & U & UX^2 \\ | & | & | & | & | & | \end{bmatrix} \tag{4}$$

The goal of sparse identification is to find each of the s coefficients associated with the s nonlinear functions considered in Θ and the input term for each row of eq. (2). Each state x_i corresponds to a sparse vector of coefficients, $\xi_i \in \mathbb{R}^s$, that represent the nonzero terms in $\hat{f}_i$ and $\hat{g}_i$ in the respective ODE, $\dot{\hat{x}}_i = \hat{f}_i(\hat{x}_i) + \hat{g}_i(\hat{x}_i)u$. Consequently, there are n such coefficient vectors that must be calculated. In matrix notation, the unknown quantity is

$$\Xi = \begin{bmatrix} \xi_1 & \xi_2 & \cdots & \xi_n \end{bmatrix} \tag{5}$$

which is found by solving the following equation:

$$\dot{X} = \Theta(X, U)\Xi \tag{6}$$

Equation (6) can be solved using sequential thresholded least-squares, wherein all coefficients in Ξ smaller than a threshold λ are zeroed and the resulting equation with zeroed terms is repeatedly solved until convergence of the non-zero coefficients. The iterations converge rapidly due to the sparse structure of Ξ. An alternate algorithm to solve eq. (6) is known as Sparse Relaxed Regularized Regression, which is based on the well-known LASSO operator [9]. After finding Ξ, the identified model can be formulated as the continuous-time differential equation,

$$\dot{x} = \Xi^\top(\Theta(x^\top, u^\top))^\top$$

where $\Theta(x^\top, u^\top)$ is a column vector containing symbolic functions of x and u from the chosen function library, and $x^\top$ represents the transpose of x.

Incorporation of SINDy within MPC

Model predictive control is an advanced control methodology that utilizes a model of the process to predict the states/output over a prediction horizon to compute the optimal control actions by solving an online optimization problem. The formulation of a Lyapunov-based model predictive controller (LMPC) that uses a sparse-identified ODE, $F_{si}(\cdot)$, as the process model is presented below:

$$\mathcal{J} = \min_{u \in S(\Delta)} \int_{t_k}^{t_{k+N}} \mathcal{C}(\tilde{x}(t), u(t))\, \mathrm{d}t \tag{7a}$$

$$\text{s.t.} \quad \dot{\tilde{x}}(t) = F_{si}(\tilde{x}(t), u(t)) \tag{7b}$$

$$u(t) \in U, \ \forall\, t \in [t_k, t_{k+N}) \tag{7c}$$

$$\tilde{x}(t_k) = x(t_k) \tag{7d}$$

$$\dot{\hat{V}}(x(t_k), u) \leq \dot{\hat{V}}(x(t_k), \Phi_{si}(x(t_k))), \quad \text{if } x(t_k) \in \Omega_{\hat{\rho}} \backslash \Omega_{\rho_{si}} \tag{7e}$$

$$\hat{V}(\tilde{x}(t)) \leq \rho_{si}, \ \forall\, t \in [t_k, t_{k+N}), \ \text{if } x(t_k) \in \Omega_{\rho_{si}} \tag{7f}$$

where $\tilde{x}$ is the predicted state trajectory, $S(\Delta)$ represents the set of piece-wise constant functions with a period of Δ, and N is the number of sampling periods within each prediction horizon. $\dot{\hat{V}}(x, u)$ is the time-derivative of the Lyapunov function and is equal to $\frac{\partial \hat{V}(x)}{\partial x} F_{si}(x, u)$. $u = u^*(t)$, $t \in [t_k,\, t_{k+N})$ denotes the optimal input sequence over the prediction horizon, which is provided by the optimizer. The LMPC applies only the first value in $u^*(t_k)$ over the next sampling period $t \in [t_k,\, t_{k+1})$, and solves the optimization again at the next sampling time t_{k+1}.

In the MPC formulation, eq. (7a) is the objective function to be minimized and is chosen to be equal to the integral of $\mathcal{C}(\tilde{x}(t), u(t))$ over the prediction horizon. eq. (7b) describes the sparse-identified model that is used to predict the closed-loop states over the prediction horizon starting from the initial condition of eq. (7d) while u is varied within the constraints defined by eq. (7c). The last two (Lyapunov) constraints of eq. (7e) guarantee that the closed-loop state either moves towards the origin at the next sampling time if the state is outside $\Omega_{\rho_{si}}$ or is contained within $\Omega_{\rho_{si}}$ for the entire prediction horizon once the state enters $\Omega_{\rho_{si}}$.

Reduced-order modeling for two-time-scale systems

Time-scale separation is a common phenomenon found in chemical processes such as distillation columns and catalytic continuous stirred-tank reactors (CSTRs) [10]. If the time-scale separation is not accounted for in a standard nonlinear feedback controller, the controller may be ill-conditioned or even unstable in closed-loop [11]. Due to the distinct slow and fast dynamics in such systems, the process will be represented by stiff ODEs in time when using SINDy without any modification. Such stiff ODEs, when integrated with an explicit integration method such as forward Euler, require a very small integration step size to prevent divergence and yield sufficiently accurate solutions. Hence, in [12], by using the mathematical framework of singular perturbations, we proposed the decomposition of the original two-time-scale system into two lower-order subsystems, each separately modeling the slow and fast dynamics of the original multiscale system. Specifically, following a short transient period, the fast states converge to a slow manifold and can be algebraically related to the slow states using nonlinear functional representations. In our work, we applied nonlinear principal component analysis (NLPCA) proposed by [13] to capture the nonlinear relationship between the slow and fast states, while using sparse identification to derive well-conditioned, reduced-order ODE models for only the slow states that could then be integrated with much larger integration time steps due to numerical stability. Once the slow states are predicted with the ODE model, it is possible to use NLPCA to algebraically predict the fast states without any integration.

Nonlinear principal component analysis is a nonlinear extension of principal component analysis (PCA). PCA is a commonly used dimensionality reduction technique that finds a linear mapping between a higher-dimensional space (of the data) and a lower-dimensional space with minimal loss of information by minimizing the squared sum of orthogonal distances between the data points and a straight line. NLPCA attempts to generalize this to the nonlinear case in two steps: first, a 1-D curve that passes through the "middle" of the data points known as the "principal curve" is found; second, the principal curve is parametrized in terms of distance of each point along the curve by using a feedforward neural network (FNN) with two hidden layers and nonlinear activation functions. Overall, to make a prediction of the state of the two-time-scale system, the measurement of the slow states at the current sampling time is passed to an explicit integrator (such as a Runge-Kutta scheme) that integrates the sparse-identified model to predict the slow states over the prediction horizon, which are then sent to the FNN to yield a prediction of the fast states.

Two-time-scale systems can be written in the form,

$$\dot{x}_s = f_s(x_s, x_f, u, \epsilon) \tag{8a}$$

$$\epsilon \dot{x}_f = f_f(x_s, x_f, u, \epsilon) \tag{8b}$$

where $x_s \in \mathbb{R}^{n_s}$ and $x_f \in \mathbb{R}^{n_f}$ denote the slow and fast states, respectively, with $n_s + n_f = n$. ϵ is a small positive parameter that represents the ratio of slow to fast dynamics of the original system. By making standard assumptions from the singular perturbation framework, the slow subsystem of eq. (8a) can be rewritten in the form required for sparse identification,

$$\dot{\hat{x}}_s = F_{si}(\hat{x}_s, u) := \hat{f}(\hat{x}_s) + \hat{g}(\hat{x}_s)u, \qquad \hat{x}_s(t_0) = x_{s0} \tag{9}$$

where F_{si} is the sparse-identified slow subsystem.

An LMPC that uses eq. (9) as the process model of eq. (7b) may be constructed. Such an LMPC will predict the slow states of the two-time-scale system and optimize the cost function based on the predicted slow states. Due to the coupled nature of the states, it is sufficient to stabilize the slow states to guarantee asymptotic stability for the entire system. However, if computational resources are available, the FNN may be used to predict the fast states, and the LMPC can then account for the full-state of the system. In [14], only the slow subsystem was used to ensure the LMPC optimization can be solved within every sampling period.

Subsampling and co-teaching in the presence of high sensor noise

A key step in the sparse identification procedure is the estimation of the time-derivatives of the states when it cannot be measured directly, as is the case in most process systems. From a survey of the literature, since the conceptualization of SINDy in [6], several advancements in the algorithm have been proposed to handle noisy data. However, most articles that investigate the effect of noise on SINDy add noise to the pre-computed derivatives (from clean data) and/or use very low levels of noise that can be easily smoothened. One example is the SINDy-PI algorithm proposed by [15] and improved by [16]. Through case studies, [16] demonstrated that even the improved algorithm could only handle noise with a maximum variance of 10^{-4}, which is very small in the context of process systems. While a detailed discussion of the comprehensive literature can be found in [17], in summary, one paper proposed an improvement upon the SINDy algorithm in the presence of moderate noise that demonstrated promise and could be developed further. This method proposed by [18], termed subsampling-based threshold sparse Bayesian regression (SubTSBR), involved randomly subsampling a fraction of the entire data set multiple times and selecting the best model by using a model-selection criterion. The issue of noisy data has also been studied in the field of computer science, where fitting a neural network to noisy data often leads to the neural network overfitting the data and capturing the noisy pattern instead. A recent technique proposed to overcome this challenge is co-teaching, where a simplified first-principles process model is used to generate noise-free training data to assist the model training step by reducing overfitting.

Subsampling is a classical statistical technique where a fraction of the total number of samples in a data set are randomly extracted and analyzed to estimate statistical parameters [19] or speed up algorithms [20]. However, subsampling can also be used to instead improve the modeling accuracy of SINDy when the data set is highly noisy. This is because common regression methods such as least squares utilize the complete data set by assuming that only a small fraction of the data samples are highly noisy or outliers. As a result, if the entire data set is used, the higher percentage of "good" data samples should smooth the large noise present in the data set. However, this assumption breaks down if the noise is either very high or uniformly present throughout the data set. In such a case, there are insufficient "good" data samples to smooth out the noise from the very highly corrupted data samples. In the context of SINDy, subsampling refers to selecting random fractions of the data set multiple times in order to sample only the less noisy data points for carrying out the sparse regression. The key requirement for subsampling is that the number of unknown weights to be estimated in the SINDy procedure have to be fewer than the number of total data samples available, which is the case for most practical data sets. Although as a standalone improvement, subsampling greatly improves the performance of SINDy under moderate noise levels, it is insufficient at higher noise levels and co-teaching becomes incumbent.

Co-teaching is a method that has been used in the field of computer science, primarily in image recognition, where neural networks are trained to categorize images into pre-defined classes. However, often, a small proportion of the images in the training data set may be mislabeled, greatly deteriorating the performance of the neural network. As manually relabeling vast amounts of images is not feasible, the method of co-teaching was proposed wherein newly generated noise-free data is fed during model training to reduce the impact of the noisy data. The concept has recently been extended to regression problems, specifically the modeling of dynamical systems using long short-term memory (LSTM) networks [21]. The central idea of co-teaching highlighted and proven in [21] is that neural networks fit simpler patterns in the early iterations of model training, which implies that noise-free data will yield low values of the loss function, while noisy data will tend to produce high loss function values. Therefore, the training can be made more robust to noise and overfitting if the noisy data is augmented with a nonzero proportion of noise-free data from simulations of simplified, approximate first-principle models that can be derived for the complex, original nonlinear system to be studied.

Improving the sparse identification algorithm with both subsampling and co-teaching enables it to tackle consistently noisy data sets where subsampling alone is insufficient. This is because subsampling only subsamples, in the best case scenario, the least noisy data points, which are still too noisy to yield an adequate model. In the proposed method, first, a random subset of the entire data set X_{noisy} is sampled, which is then mixed with noise-free data generated from approximate first-principles models of the process, X_{FP}. The resulting mixed data set is used to solve for the unknown weights of the various terms in the SINDy function library. Once a model is identified, a model-selection criterion is used to evaluate the model performance. Three parameters must be specified in the algorithm: $p \in (0, 1)$ or the subsampling fraction, $q \in (0, 1)$ or the noise-free subsampling fraction, and $L \geq 1$, which is the number of times to independently subsample and identify a SINDy

model. The algorithm randomly subsamples and mixes $p \times m$ data points from the noisy data set with $q \times m$ data points from the noise-free data set to produce the data submatrix X_i for subsample i with $i = 1, 2, \ldots, L$. U_i are the corresponding $(p+q) \times m$ points from U. The sparse regression equation to be solved is then

$$\dot{X}_i = \Theta(X_i, U_i)\Xi_i \tag{10}$$

where Ξ_i are the coefficients associated with each library function that is identified using the data subset X_i. Once Ξ_i is determined, and therefore, the i^{th} ODE model is found, the model selection criterion is used to extract the optimal model. An example of a model selection criterion that balances the error with the model sparsity, which is crucial for SINDy, is the Akaike Information Criterion given by the expression,

$$\text{MSE} = \frac{1}{m}\sum_{j=1}^{m}(x(t_j) - \hat{x}(t_j))^2 \tag{11}$$

$$\text{AIC} = m \log \text{MSE} + 2L_0 \tag{12}$$

where L_0 denotes the 0^{th} norm and is equal to the number of nonzero terms in the sparse-identified model.

The hyperparameters in the algorithm, which require tuning, are the candidate function library, the method for estimating the time-derivative of the noisy data, the optimizer, and the value of λ. Possible choices for the function library are either only polynomials or polynomials with other nonlinear trigonometric or exponential functions based on the problem definition. The total variation regularized derivative (TVRD) as well as the smoothed finite-difference (SFD) (computing finite differences after presmoothing with a Savitzky-Golay filter) have been demonstrated to produce the best results [17], making them reasonable choices. Two simple yet effective optimizers are the sequential thresholded least squares (STLSQ) and the SR3 optimizer based on the LASSO [9]. For the value of λ, depending on the system, a coarse search in steps of 0.1 from 0 to 1 may be sufficient; otherwise, a coarse-to-fine search is most efficient both computationally and in terms of accuracy. Figure 1 shows the flow of the data throughout the algorithm. The complete algorithm can be found in [17].

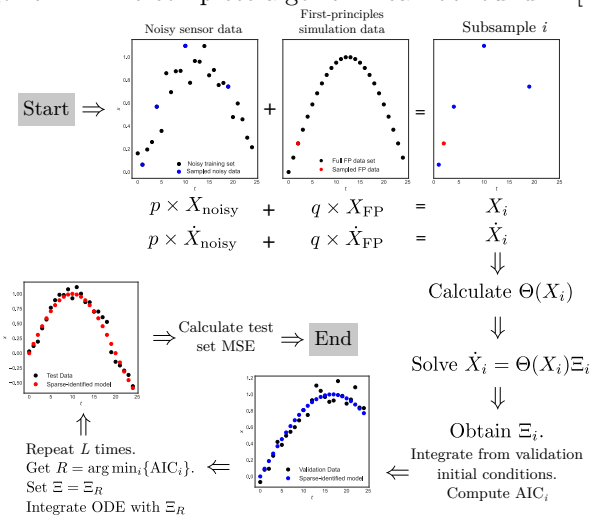

Figure 1: Data flow diagram of subsampling with co-teaching for noisy data.

Ensembled-based dropout-SINDy to model highly noisy industrial data sets

While subsampling with co-teaching is a viable option to overcome the issue of high sensor noise in the data measurements, the primary drawback of co-teaching is its requirement for a first-principles process model that is at least similar to the original system with respect to the dynamics and the final steady-state values. However, in the case of industrial data, especially when corrupted by noise, the dynamics may be far too complex for any theoretically derived ODE to accurately capture the system. Therefore, for the case of dealing with high levels of industrial noise, a new direction and improvement on SINDy is proposed, known as ensemble learning or dropout-SINDy.

Ensemble learning refers to the use of multiple models of either the same type or different types to improve the predictive performance. While subsampling takes a fraction of the data set to create multiple models, dropout-SINDy uses only a fraction of the function library Θ to identify a model. Hence, multiple models can be identified, each with a random subset of the library. Similarly to co-teaching, this can reduce the impact of noisy data and also improve the stability properties of the SINDy models because a large number of nonzero terms (a dense coefficient matrix Ξ) can often lead to instabilities. The sparse regression equation to be solved for dropout-SINDy is similar to eq. (10) but the state and input data sets remain as X and U, respectively, while only the coefficients Ξ_i are varied between models in the ensemble:

$$\dot{X}_i = \Theta(X, U)\Xi_i \tag{13}$$

When using highly noisy industrial data, it is found that basic SINDy is unable to model the dynamics and even the final steady-state of the system, the latter of which greatly affects the performance of a controller. However, when dropout-SINDy is used on the industrial data set, it is able to capture most of the dynamics and correctly predict the final steady-state values of the states. When an MPC is designed with the dropout-SINDy model, it can be demonstrated to achieve closed-loop stability and converge to the steady-state faster and with less energy than a corresponding proportional-controller.

Conclusions and future work

In this paper, we have discussed several recent advancements in sparse identification to overcome the challenges of modeling and controlling two-time-scale systems and noisy data. The methods considered include combining SINDy with nonlinear principal component analysis, feedforward neural networks, subsampling, co-teaching, and ensemble learning. An existing challenge in the SINDy procedure is the choice of the function library. If the correct basis is not considered, the identified model can often be dense and/or unstable. However, once the candidate library is chosen appropriately,

even with minimal hyperparameter tuning, sparse and accurate models can be easily identified. Hence, possible future directions in this field include the use of neural network function approximators in the candidate basis functions to capture nonlinearities that are not among the commonly considered basis functions. If possible, converting these neural network parts of the SINDy model back to symbolic functions will greatly improve the model inference time as explicit nonlinearities are computationally desirable. Such advances have already been initiated in recent papers on modeling biological systems [22].

References

[1] S. Zhang and G. Lin. Robust data-driven discovery of governing physical laws with error bars. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 474(2217):20180305, 2018.

[2] K. S. Holkar and L. M. Waghmare. An overview of model predictive control. *International Journal of Control and Automation*, 3(4):47–63, 2010.

[3] E. Aggelogiannaki and H. Sarimveis. Nonlinear model predictive control for distributed parameter systems using data driven artificial neural network models. *Computers & Chemical Engineering*, 32(6):1225–1237, 2008.

[4] C. Moore. Application of Singular Value Decomposition to the Design, Analysis, and Control of Industrial Processes. In *1986 American Control Conference*, pages 643–650, Seattle, WA, USA, 1986.

[5] P. Van Overschee and B. De Moor. N4SID: Subspace algorithms for the identification of combined deterministic-stochastic systems. *Automatica*, 30(1):75–93, 1994.

[6] S. L. Brunton, J. L. Proctor, and J. N. Kutz. Discovering governing equations from data by sparse identification of nonlinear dynamical systems. *Proceedings of the National Academy of Sciences*, 113(15):3932–3937, 2016.

[7] Z. Bai, T. Wimalajeewa, Z. Berger, G. Wang, M. Glauser, and P. K. Varshney. Low-Dimensional Approach for Reconstruction of Airfoil Data via Compressive Sensing. *AIAA Journal*, 53(4):920–933, 2015.

[8] B. Bhadriraju, M. S. F. Bangi, A. Narasingam, and J. S.-I. Kwon. Operable adaptive sparse identification of systems: Application to chemical processes. *AIChE Journal*, 66(11):e16980, 2020.

[9] P. Zheng, T. Askham, S. L. Brunton, J. N. Kutz, and A. Y. Aravkin. A Unified Framework for Sparse Relaxed Regularized Regression: SR3. *IEEE Access*, 7:1404–1423, 2019.

[10] H.-C. Chang and M. Aluko. Multi-scale analysis of exotic dynamics in surface catalyzed reactions—I: Justification and preliminary model discriminations. *Chemical Engineering Science*, 39(1):37–50, 1984.

[11] P. Kokotović, H. K. Khalil, and J. O'Reilly. *Singular Perturbation Methods in Control: Analysis and Design*. Society for Industrial and Applied Mathematics, 1999.

[12] F. Abdullah, Z. Wu, and P. D. Christofides. Data-based reduced-order modeling of nonlinear two-time-scale processes. *Chemical Engineering Research and Design*, 166:1–9, 2021.

[13] D. Dong and T. McAvoy. Nonlinear principal component analysis—Based on principal curves and neural networks. *Computers & Chemical Engineering*, 20(1):65–78, 1996.

[14] F. Abdullah, Z. Wu, and P. D. Christofides. Sparse-identification-based model predictive control of nonlinear two-time-scale processes. *Computers & Chemical Engineering*, 153:107411, 2021.

[15] N. M. Mangan, S. L. Brunton, J. L. Proctor, and J. N. Kutz. Inferring Biological Networks by Sparse Identification of Nonlinear Dynamics. *IEEE Transactions on Molecular, Biological and Multi-Scale Communications*, 2(1):52–63, 2016.

[16] K. Kaheman, J. N. Kutz, and S. L. Brunton. SINDy-PI: a robust algorithm for parallel implicit sparse identification of nonlinear dynamics. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 476(2242):20200279, 2020.

[17] F. Abdullah, Z. Wu, and P. D. Christofides. Handling noisy data in sparse model identification using subsampling and co-teaching. *Computers & Chemical Engineering*, 157:107628, 2022.

[18] S. Zhang and G. Lin. SubTSBR to tackle high noise and outliers for data-driven discovery of differential equations. *Journal of Computational Physics*, 428:109962, 2021.

[19] B. Efron and C. Stein. The Jackknife Estimate of Variance. *The Annals of Statistics*, 9(3):586 – 596, 1981.

[20] S. H. Rudy, S. L. Brunton, J. L. Proctor, and J. N. Kutz. Data-driven discovery of partial differential equations. *Science Advances*, 3(4):e1602614, 2017.

[21] Z. Wu, D. Rincon, J. Luo, and P. D. Christofides. Machine learning modeling and predictive control of nonlinear processes using noisy data. *AIChE Journal*, 67(4):e17164, 2021.

[22] C. Rackauckas, Y. Ma, J. Martensen, C. Warner, K. Zubov, R. Supekar, D. Skinner, A. Ramadhan, and A. Edelman. Universal differential equations for scientific machine learning. *arXiv preprint arXiv:2001.04385*, 2020.