Nonlinear System Identification of Soft Robot Dynamics Using Koopman Operator Theory

Daniel Bruder, C. David Remy, and Ram Vasudevan, *Member, IEEE*

Abstract— Soft robots are challenging to model due in large part to the nonlinear properties of soft materials. Fortunately, this softness makes it possible to safely observe their behavior under random control inputs, making them amenable to large-scale data collection and system identification. This paper implements and evaluates a system identification method based on Koopman operator theory in which models of nonlinear dynamical systems are constructed via linear regression of observed data by exploiting the fact that every nonlinear system has a linear representation in the infinite-dimensional space of real-valued functions called observables. The approach does not suffer from some of the shortcomings of other nonlinear system identification methods, which typically require the manual tuning of training parameters and have limited convergence guarantees. A dynamic model of a pneumatic soft robot arm is constructed via this method, and used to predict the behavior of the real system. The total normalized-root-mean-square error (NRMSE) of its predictions is lower than that of several other identified models including a neural network, NLARX, nonlinear Hammerstein-Wiener, and linear state space model.

I. INTRODUCTION

Soft robots incorporate non-rigid materials into their morphology to facilitate compliant interactions with the external world. This compliance allows them to manipulate delicate objects, adapt to unstructured environments, and interact safely with coexisting humans [1], [2], [3]. Since their utility is derived from their compliance, control methods that preserve and/or exploit this property are desirable.

Accurate models facilitate better control performance. When an accurate model is available, predictive controllers can be built by using the model to calculate a feedforward term, then adding a feedback term to account for minor model uncertainty and disturbances. If an accurate model is unavailable, feedback must be relied upon more heavily. This poses several problems for soft robots. First, feedback requires sensing, but the morphology of soft robots precludes the use of most conventional sensors. Suitable alternatives are currently in development [4], [5], [6], [7], but are not yet readily available. Second, relying heavily on feedback to compensate for an inaccurate model has been illustrated to reduce the compliance of soft robotic systems [8]. That is, excessive feedback negates the desirable compliance of

The authors are with the Mechanical Engineering Department at the University of Michigan, Ann Arbor, MI 48109, USA {bruderd, cdremy, ramv}@umich.edu
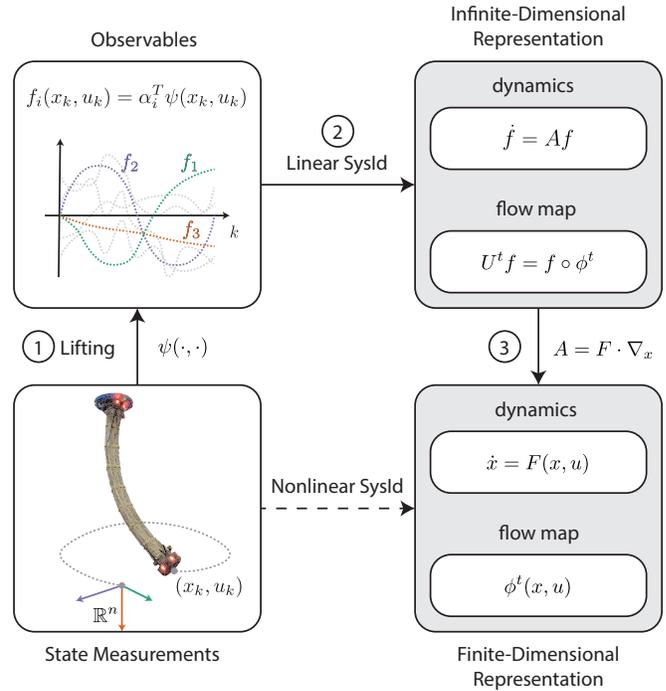
Fig. 1. By providing an infinite-dimensional linear representation of a dynamical system, Koopman operator theory enables linear system identification of nonlinear systems. This process proceeds in three steps, as described in Section II: (1) Measured states of the system are lifted to the space of real-valued functions of the state and input. (2) Least-squares regression is performed on the lifted data to obtain an approximation of the Koopman operator, U^t. (3) An approximation of the nonlinear vector field F is obtained via its one-to-one relationship with the Koopman operator.

a soft robot by replacing its natural dynamics with those of a slower, stiffer system. Therefore, accurate models are required to control soft robots in a manner that reduces dependence on feedback and simultaneously preserves compliance.

Models for soft robots can be separated into two categories: physics-based models and data-driven models. Physics-based models are constructed from observations of component material properties and first-principles, while data-driven models are constructed from observations of system behavior. Physics-based models have the ability to make predictions about a system's behavior before the system is constructed. Thus, they are often used to inform the design and construction of soft robots intended for particular tasks [9], [10], [11], [12], [13]. However, the infinite degrees of freedom and nonlinear behavior of soft materials make

it difficult to construct accurate physics-based models of soft robots without making simplifying assumptions such as constant curvature [14], [15], quasi-static [16], [17], [18], or simplified geometry [19], [20], [21]. These models only describe behavior well in the subset of robot configurations where the assumptions hold; hence, they are limited in applicability.

Data-driven models are more broadly applicable since they do not make structural assumptions about the system. Data-driven models are constructed from observations of system behavior rather than from first-principles. Hence, given sufficient data, they are capable of describing system behavior well over the entire range of observations. By virtue of their soft bodies, it is often possible to safely command arbitrary control inputs without risk of damaging a soft robot or nearby human operators. Soft robots are therefore amenable to data collection over their entire operation range, making them particularly well suited for data-driven system identification techniques.

To capture the characteristic nonlinear behaviors of most soft robots, identification of a nonlinear model is necessary. Unfortunately, identifying a nonlinear model from data typically consists of solving a nonlinear, non-convex optimization problem, for which global convergence is not guaranteed [22]. Furthermore, most nonlinear system identification methods require the manual initialization and tuning of training parameters, which have an obscure impact on the resulting model. A neural network, for example, may be able to capture the nonlinear behavior of a soft robot [23]; however, its accuracy depends on the number of hidden layers, number of nodes per layer, activation function, and termination condition used during training, which must be selected through trial and error until acceptable results are achieved.

Linear model identification, on the other hand, does not suffer from the typical shortcomings of nonlinear identification since linear models can be identified via linear regression [24]. However, linear models are poorly suited to capture the behavior of soft robots since their characteristic behavior is distinctly nonlinear [1].

In this work, we employ Koopman operator theory to generate nonlinear, data-driven models of soft robots via linear regression (see Fig. 1). This approach relies on the idea of lifting nonlinear dynamical systems to an infinite dimensional space, where those systems have a linear representation. In this space, it is possible to describe the dynamic behavior of a system by a linear operator rather than a nonlinear vector field [25]. This linear operator, called the Koopman operator, is identified via linear regression [26], so it does not suffer from the convergence and tuning problems that are characteristic of neural networks and other nonlinear system identification methods.

Our primary contribution is demonstrating, on a real system, a data-driven method for constructing globally valid nonlinear models that does not require the manual tuning of multiple training parameters. To do so, we apply the Koopman based system identification method described in

[27] and [28] to create a dynamic model of a soft robot arm and verify that it captures the system's true dynamic behavior better than the models generated by several other state-of-the-art nonlinear system identification methods including a neural network, a nonlinear auto-regressive with exogenous inputs model (NLARX), a nonlinear Hammerstein-Wiener model, and a state space model. Koopman operator theory has previously been used to perform model-based control of nonlinear systems [29]. However, to the author's knowledge, this technique has never before been used to identify the dynamic model of a real system, much less a soft robot. We believe that this system identification method applied to soft robots will enable the rapid development of new and effective control strategies by making accurate nonlinear dynamic models easier to construct.

The rest of this paper is organized as follows: In Section II we formally introduce the Koopman operator and describe the system identification method. In Section III we describe the soft robot and experimental procedure used for collecting input-output data of the system. In Section IV we summarize the results of applying various nonlinear system identification techniques to the collected data and compare the performances of the models generated. In Section V concluding remarks and perspectives are provided.

II. KOOPMAN OPERATOR METHOD FOR SYSTEM IDENTIFICATION

The system identification method utilized in this work exploits the fact that any finite-dimensional nonlinear system has an equivalent infinite-dimensional linear representation in the space of real-valued functions of the system's state and input. In this space of real-valued functions, the (linear) Koopman operator describes the flow of functions along trajectories of the system. The relationship between the finite and infinite dimensional representations of the system is bijective and well-defined [30]. This enables us to approximate the Koopman operator via linear regression on observed data, then extract the equivalent nonlinear system representation. The remainder of this section summarizes the system identification method presented in [27] and [28] applied to a system with known input, which is later employed and validated on a real soft robotic system.

A. Overview of Lifting Technique

Consider a dynamical system

$$\dot{x} = F(x, u) \tag{1}$$

where $x \in \mathbb{R}^n$ is the state of the system, $u \in \mathbb{R}^m$ is the input and F is continuously differentiable in x. Denote by $\phi(t, x_0, u)$ the solution to (1) at time t when beginning with the initial condition x_0 at time 0 and a constant input u applied for all time between 0 and t. For simplicity, we denote this map, which is referred to as the *flow map*, by $\phi^t(x_0, u)$ instead of $\phi(t, x_0, u)$.

The system can be lifted to an infinite dimensional function space $\mathcal{F} = L^2(X \times U, \mathbb{R})$ where $X \subset \mathbb{R}^n$ and $U \subset \mathbb{R}^m$ are compact subsets and $L^2(X \times U, \mathbb{R})$ is the space of

square integrable real-valued functions with domain $X \times U$. Elements of $\mathcal{F}$ are called *observables*. In $\mathcal{F}$, the flow of the system is characterized by the set of Koopman operators $U^t : \mathcal{F} \to \mathcal{F}$, for each $t \geq 0$, which describes the evolution of the observables $f \in \mathcal{F}$ along the trajectories of the system according to the following definition:

$$U^t f = f \circ \phi^t \tag{2}$$

As desired, U^t is a linear operator even if the system (1) is nonlinear, since for $f_1, f_2 \in \mathcal{F}$ and $\lambda_1, \lambda_2 \in \mathbb{R}$

$$\begin{aligned} U^t(\lambda_1 f_1 + \lambda_2 f_2) &= \lambda_1 f_1 \circ \phi^t + \lambda_2 f_2 \circ \phi^t \\ &= \lambda_1 U^t f_1 + \lambda_2 U^t f_2. \end{aligned} \tag{3}$$

Thus the Koopman operator provides a linear representation of the flow of the system in the infinite-dimensional space of observables (see Fig. 1) [25].

One can show that there is a one-to-one correspondence between the infinite-dimensional Koopman operator and finite-dimensional vector field. To understand this relationship, consider the time derivative of an observable, $\dot{f}$, along trajectories of the system:

$$\dot{f}(x, u) = \frac{\partial f}{\partial x}\frac{dx}{dt} + \frac{\partial f}{\partial u}\frac{du}{dt} \tag{4}$$

$$= \frac{\partial f}{\partial x} F(x, u) \tag{5}$$

$$= (F \cdot \nabla_x) f(x, u) \tag{6}$$

where the second relation follows since u is held constant and where ∇_x is the gradient with respect to x. Since this equation holds for all observables, we introduce the infinitesimal generator of the Koopman operator $A : \mathcal{F} \to \mathcal{F}$ [30, Equation 7.6.5] which is defined in terms of the vector field F as

$$A = F \cdot \nabla_x \tag{7}$$

The infinitesimal generator thus describes the dynamics of the observables along trajectories of the system (i.e. $\dot{f} = Af$). Recalling that the Koopman operator describes the flow of observables, one can show that the Koopman operator is associated with its infinitesimal generator via the following relation:

$$U^t = e^{At} = \sum_{k=0}^{\infty} \frac{t^k}{k!} A^k \tag{8}$$

In Section II-D Equations (7) (8) are used to solve for the vector field F when the Koopman operator U^t is known.

By providing a linear representation of a system with a one-to-one correspondence to its nonlinear representation, Koopman operator theory enables linear system identification of nonlinear systems. This process proceeds in three steps which are summarized by Algorithm 1. In step one, measured states of the system are lifted to the space of observables. Step two consists of performing least-squares regression on the lifted data to obtain an approximation of the Koopman operator. In step three, an approximation of the nonlinear vector field F is obtained using (7) and (8). The following three subsections describe each of these steps in more detail.

Algorithm 1: Koopman-Based System Identification

Input: $\{(x_k, u_k), (x_{k+1}, u_k)\}$ for $k = 1, ..., K$
Step 1: Lift data via (10)
Step 2: Approximate Koopman operator, $\bar{U}^{T_s}$ via (14)
Step 3: Approximate Vector Field, $\bar{F}$ via (18) and (21)
Output: $\bar{F}$

B. Step 1: Lifting the Data

The first step of the Koopman-based system identification method consists of converting empirical data into a form that can be used to identify a linear model in the space of observables. Theoretically this process would consist of "lifting" state measurements into the infinite-dimensional space of observables $\mathcal{F}$. To be implementable, however, measurements can only be lifted into a finite-dimensional subspace. Define $\bar{\mathcal{F}} \subset \mathcal{F}$ to be the subspace of $\mathcal{F}$ spanned by N linearly independent basis functions $\{\psi_k\}_{k=1}^N$ (e.g. monomials, sinusoids, exponentials). Any observable $\bar{f} \in \bar{\mathcal{F}}$ can be written as a linear combination of elements of the basis

$$\bar{f} = \alpha_1 \psi_1 + \cdots + \alpha_N \psi_N. \tag{9}$$

Note that the vector of coefficients $\alpha = \begin{bmatrix} \alpha_1 & \cdots & \alpha_N \end{bmatrix}^T$ provides a *vector representation* for $\bar{f} \in \bar{\mathcal{F}}$. To evaluate $\bar{f}$ at a given state x and constant input u, we introduce the *lifting function* $\psi : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^N$ defined as:

$$\psi(x, u) = \begin{bmatrix} \psi_1(x, u) & \cdots & \psi_N(x, u) \end{bmatrix}^T \tag{10}$$

Then, $\bar{f}(x, u)$ can be expressed in vector form as

$$\bar{f}(x, u) = \alpha^T \psi(x, u). \tag{11}$$

We refer to $\psi(x, u)$ as an N-dimensional "lifted" version of (x, u), since multiplying $\psi(x, u)$ by the vector representation of an observable yields the value of the observable at (x, u).

C. Step 2: Approximating the Koopman Operator

The second step of the Koopman-based system identification method is to identify the Koopman operator that best describes the flow of the lifted versions of measured data points. While the Koopman operator is theoretically infinite-dimensional, for practical purposes we identify a finite-dimensional approximation of it in $\bar{\mathcal{F}}$ which we denote by $\bar{U}^t$. Note that $\bar{U}^t$ can be represented by an $N \times N$ matrix which operates on observables via matrix multiplication:

$$\bar{U}^t \alpha = \beta \tag{12}$$

where α, β are vector representations of observables in $\bar{\mathcal{F}}$. Our goal is to find a $\bar{U}^t$ that describes the action of the infinite dimensional Koopman operator U^t as accurately as possible in the L^2-norm sense on the finite dimensional subspace $\bar{\mathcal{F}}$ of all observables. Therefore, to perfectly mimic the action of U^t acting on an observable in $\bar{\mathcal{F}} \subset \mathcal{F}$, the following should be true

$$(\bar{U}^t \alpha)^T \psi(x, u) = \alpha^T \psi \left(\phi^t(x, u), u \right). \tag{13}$$

Since this is a linear equation, it follows that for a given $x \in \mathbb{R}^n, u \in \mathbb{R}^m$, solving (13) for $\bar{U}^t$ yields the best approximation of U^t on $\bar{\mathcal{F}}$ in the L^2-norm sense:

$$\bar{U}^t = \left(\psi(x,u)^T \right)^\dagger \psi(\phi^t(x,u),u)^T \qquad (14)$$

where superscript $\dagger$ denotes the least-squares pseudoinverse.

To approximate the Koopman operator from a set of experimental data, we take $K{+}1$ discrete state measurements with sampling period T_s. Under the assumption that the control input is constant between samples, we separate the data into a set of K so-called "snapshot pairs" of the form $\{(x_k, u_k), (y_k, u_k)\} \in \mathbb{R}^{(n \times m) \times 2}$ where

$$y_k = \phi^{T_s}(x_k, u_k) + \sigma_k \qquad (15)$$

and σ_k denotes measurement noise. For our basis of $\bar{\mathcal{F}}$, we choose the basis of monomials of x and u with total degree less than or equal to w, which implies $N = (n + m + w)! / \big((n+m)! w! \big)$ [27, Section III]. We then lift all of the snapshot pairs according to (10) and compile them into the following $K \times N$ matrices:

$$\Psi_x = \begin{bmatrix} \psi(x_1, u_1)^T \\ \vdots \\ \psi(x_K, u_K)^T \end{bmatrix} \qquad \Psi_y = \begin{bmatrix} \psi(y_1, u_1)^T \\ \vdots \\ \psi(y_K, u_K)^T \end{bmatrix} \qquad (16)$$

$\bar{U}^{T_s}$ is chosen so that it yields the least squares best fit to all of the observed data, which, following from (14), is given by

$$\bar{U}^{T_s} := \Psi_x^\dagger \Psi_y. \qquad (17)$$

D. Step 3: Obtaining the Vector Field

The final step of the Koopman-based system identification method is to identify the nonlinear vector field by making use of the one-to-one correspondence between the infinite and finite dimensional system representations. As earlier, our goal is to find an $\bar{F}$ that describes the behavior of the vector field F as accurately as possible in the L^2-norm sense on the finite dimensional subspace $\bar{\mathcal{F}}$.

The vector field is related to the Koopman operator through its infinitesimal generator according to Equation (7). With the approximation of the Koopman operator $\bar{U}^{T_s}$ found in Section II-C, we can solve for the infinitesimal generator of the set of Koopman operators $\bar{A}$ by inverting (8):

$$\bar{A} = \frac{1}{T_s} \log \bar{U}^t \in \mathbb{R}^{N \times N} \qquad (18)$$

where $\log$ denotes the principal matrix logarithm [31, Chapter 11]. Note that the principal matrix logarithm is only defined for matrices whose eigenvalues all have non-negative real components, and that $\bar{U}^t$ may have zero or negative eigenvalues when the number of data points is too small [27]. Therefore this method might fail if the number of data points is insufficient. In this instance, more system measurements can be taken to resolve the issue.

With $\bar{A}$ known, (7) can be used to identify $\bar{F}$. Consider A applied to an observable $f \in \mathcal{F}$. According to (7), this is equivalent to the inner product of the vector field F and the gradient of f with respect to x:

$$Af(x,u) = \frac{\partial f(x,u)}{\partial x} F(x,u). \qquad (19)$$

Let $\alpha \in \mathbb{R}^N$ be the vector representation $\bar{f}$, the projection of f onto $\bar{\mathcal{F}}$. Then from (11) the finite-dimensional equivalent of (19) is given by

$$(\bar{A}\alpha)^T \psi(x,u) = \alpha^T \frac{\partial \psi(x,u)}{\partial x} \bar{F}. \qquad (20)$$

We seek the vector field $\bar{F}$ such that (20) holds as well as possible in the L^2-norm sense for all observed data. Therefore we choose the least-square solution to (20) over the set of all observed data points $\big\{ (x_k, u_k) | k = 1, ..., K \big\}$ which is given by

$$\bar{F} = \begin{bmatrix} \frac{\partial \psi(x_1, u_1)}{\partial x} \\ \vdots \\ \frac{\partial \psi(x_K, u_K)}{\partial x} \end{bmatrix}^\dagger \begin{bmatrix} \bar{A}^T & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \bar{A}^T \end{bmatrix} \begin{bmatrix} \psi(x_1, u_1) \\ \vdots \\ \psi(x_K, u_K) \end{bmatrix}. \qquad (21)$$

For a more thorough treatment of this process, see [27], [28].

III. System Identification of a Soft Robot

To demonstrate and evaluate the performance of the method outlined in Section II, we applied it to a continuously deformable soft robot arm and compared the resulting model to those generated by several other nonlinear system identification techniques. In the following, we describe in detail the robotic hardware, experimental setup, measurement procedure, and the data processing involved in the system identification process, as well as the process by which performance was evaluated and compared across models.

A. Hardware Description

The soft robot used in this experiment consisted of three pneumatically driven McKibben actuators (also known as Pneumatic Artificial Muscles or PAMs) adhered together by latex rubber and connected to a common base mount on one end and to an end effector on the other (see Fig. 2). During the trials, the pressure inside each actuator was varied using a pneumatic pressure regulator (Enfield TR-010-g10-s), and the displacement and velocity of the end effector was measured at 60Hz using a commercial motion capture system (Phase Space Impulse X2E).

For this robot, it is our primary interest to control the motion of the end effector. Hence, we chose a state representation of the system which is capable of describing the dynamics of the end effector as an ordinary differential equation, namely the position and velocity of the end effector with respect to a global coordinate frame, as shown in Fig. 2

$$\vec{x} = \begin{bmatrix} x_1 & x_2 & x_3 & \dot{x}_1 & \dot{x}_2 & \dot{x}_3 \end{bmatrix}^T \qquad (22)$$

TABLE I

DATA COLLECTION PARAMETERS

	Trial					
	1	2	3	4	5	6
Length (min)	40:06	31:26	8:16	26:09	28:38	31:35
T_u (s)	3.0	3.5	4	5	5	8

B. Randomized Control Input

To generate a representative sampling of the system's behavior over its entire operation range, a randomized input was applied. The control input into the system u was a set of three $0 - 10$V signals into the pressure regulators corresponding to actuator pressures of $\approx 0 - 140$ kPa

$$u(t) = \begin{bmatrix} u_1(t) & u_2(t) & u_3(t) \end{bmatrix}^T, \qquad u_i \in [0, 10] \quad (23)$$

Before each trial, a $3 \times K_u$ table Υ of uniformly distributed random numbers between zero and ten was generated to be used as an input lookup table. K_u was chosen to be large enough to provide inputs over the entire length of each trial. Each control input was smoothly varied between elements in consecutive columns of the table over a transition period T_u with a time offset of $T_u/3$ between each of the three control signals,

$$u_i(t) = \frac{(\Upsilon_{i,k+1} - \Upsilon_{i,k})}{T_u} \left(t + \frac{(i-1)T_u}{3} \right) + \Upsilon_{i,k} \quad (24)$$

where $k = \text{floor}\left(t/T_u\right)$ is the current index into the lookup table at time t.

C. Data Collection

Data collection proceeded in 6 trials each lasting an average of ≈ 27 minutes. The input transition period T_u varied from trial to trial, taking values between 3 and 8 seconds (see Table I for the specific values for each trial). After collecting data, raw position and velocity measurements were put through a moving average filter with window size of 1s to reduce noise, then sampled uniformly with a period $T_s = 0.02$ seconds. Sampled velocity measurements were put through a second moving average filter with window size of 1 second due to the higher noise content of the velocity signal. The time-series data from each trial was partitioned into training and validation sets. Three 10 second validation sets were extracted from each trial and the remainder of the trial data was used for training.

D. Model Comparison

We generated a state space model from the technique described in section II using a monomial basis of maximum degree $w = 3$ and the collected training data. We then evaluated its accuracy by comparing model simulations to each of the validation data sets (Fig. 3). Goodness of fit for the trajectory of a state y was calculated using the normalized
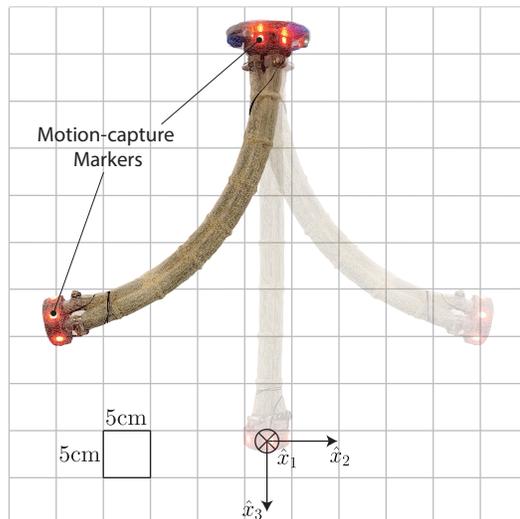

Fig. 2. System identification was performed on a soft robot consisting of three PAMs adhered together. Active motion capture markers on the base and end effector enabled tracking of the position and velocity of the end effector relative to the fixed global coordinate frame marked by unit vectors $\hat{x}_1, \hat{x}_2, \hat{x}_3$ where $\hat{x}_1$ is pointing into the page. The robot's range of motion given control inputs defined in (24) is depicted.

root-mean-square error (NRMSE), defined:

$$\text{RMSE} = \sqrt{\frac{\sum_{k=1}^{N_{\text{total}}} (y_k - \hat{y}_k)^2}{N_{\text{total}}}} \quad (25)$$

$$\text{NRMSE} = \left(\frac{\text{RMSE}}{y_{\max} - y_{\min}} \right) \cdot 100\% \quad (26)$$

where $\hat{y}$ is the simulated value of the state, N_{total} is the total number of points, and $y_{\min/\max}$ are the measured minimum/maximum values of the state observed over all trials.

The performance of our model was benchmarked against a linear state space, nonlinear Hammerstein-Wiener, nonlinear auto-regressive with exogenous inputs (NLARX), and a feedforward neural network model. The models were trained and evaluated on the non-lifted time-series data from the experiments described in Section III, and generated using either the Matlab System Identification Toolbox or Neural Network Toolbox [32]. The state space model was generated using the subspace method [24, Chapter 7] and specified to be 6 dimensional, i.e. the same dimension as the state defined in 22. The neural network model was trained using the Levenberg-Marquardt backpropagation algorithm and sigmoid activation functions. It was trained several times using combinations of 10-30 hidden neurons and 1-10 delays. Only the results for the best of these models, corresponding to 10 hidden neurons and 10 delays, is displayed in Fig. 4 and Table II.

IV. RESULTS

The model generated by the Koopman system identification method has a total RMSE averaged across position states and velocity states of 5.98 mm and 3.66 mm/s, respectively. As shown in Table II, this corresponds to a total NRMSE
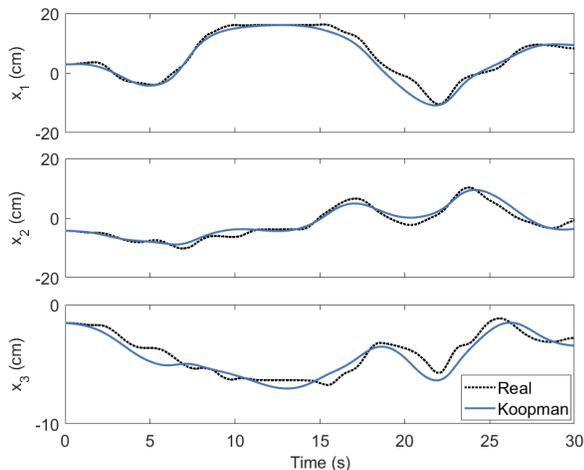
Fig. 3. The measured position of the robot end effector over a 30 second time window (black,dotted) superimposed with the position predicted by the Koopman-based model (blue) given the same initial condition and control inputs. Coordinates are defined with respect to the global coordinate frame depicted in Fig. 2.
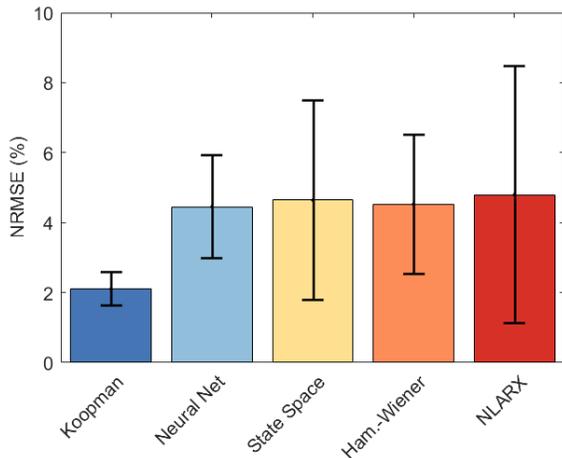

Fig. 4. Shown is the total NRMSE averaged across all states for each of the models, with the standard deviation designated by the black bar. The average NRMSE of the Koopman-based model is less than half of that of the other models, with a standard deviation of less than one third of the other models.

Model	States						Avg.	Std. Dev.
	x_1	x_2	x_3	x_4	x_5	x_6		
Koopman	2.4	2.0	2.9	1.7	1.5	2.0	2.1	0.5
Neural Net	5.8	4.0	6.6	3.9	2.8	3.5	4.5	1.5
State Space	5.1	3.1	9.9	3.0	1.8	4.8	4.6	2.9
Ham.-Weiner	7.0	4.5	6.9	3.0	2.3	3.1	4.5	2.0
NLARX	5.0	3.0	12.0	3.8	2.1	2.8	4.8	3.7

and shown that the model generated outperforms those constructed by several other state-of-the-art nonlinear system identification methods. Perhaps unsurprisingly, the linear state space model was unable to capture the nonlinear dynamics of the robot as well as the Koopman model. As for the nonlinear models, there are several likely reasons why the performance of the Koopman model was superior. Since the Koopman model is a state space model, simulations can be initialized from the same initial condition as the real system. This is not the case for the other learned nonlinear models which do not have an internal state corresponding to the physical state of the robot. Rather, they act as black-box models only capable of mapping inputs to outputs.

Another advantage of the Koopman model is that its quality does not depend on an initial model estimate or tuning parameters. By iterating over the set of all initializations and tuning parameters, one may be able to generate better performing models than those shown; unfortunately, this multivariate trial-and-error process may not affect results in a predictable way. In contrast, the only tuning parameter involved in the Koopman method is the maximum degree of the monomial basis functions, which has a magnitude that is directly proportional to model accuracy.

While the results here are promising, there are practical challenges to extending the Koopman approach to higher dimensional systems. As the dimension of the state space increases, so does the size of the monomial basis set of the finite-dimensional subspace of observables. This greatly increases the size of the matrix equations that must be solved, leading to computational intractability for sufficiently high dimensional systems. However, if some information about the system is known beforehand, this issue could be counteracted by choosing a more suitable basis for the observables. For example, if the system exhibits oscillatory motion, a lower dimensional fourier basis may be more suitable than a monomial basis to represent the behavior. Such an extension of the method is left to future work.

Soft robots are notoriously difficult to model, but amenable to large-scale data collection and data-driven modeling methods. This paper demonstrates the utility of Koopman operator theory to make accurate nonlinear dynamical models easier to construct, enabling the rapid development of control strategies that exploit the unique characteristics of soft robots. Future work will aim to generalize this approach to higher dimensional models, non-polynomial models, and models that account for external loading and contact forces.

averaged across all states of 2.1%. By this metric, it performs more than twice as well as the best competing linear and nonlinear models which have average NRMSEs of 4.6% and 4.5%, respectively (see Fig. 4). The Koopman model also exhibits the smallest standard deviation of the NRMSE across states. This implies that the Koopman model more consistently captures the real behavior of all six states of the system, rather than just a subset of them. Fig. 3 illustrates the ability of the Koopman-based model to predict the position of the end effector over a 30 second time horizon.

V. DISCUSSION AND CONCLUSION

We have successfully applied a system identification technique based on Koopman operator theory to a soft robot

REFERENCES

[1] D. Rus and M. T. Tolley, "Design, fabrication and control of soft robots," *Nature*, vol. 521, no. 7553, p. 467, 2015.

[2] C. Majidi, "Soft robotics: a perspectivecurrent trends and prospects for the future," *Soft Robotics*, vol. 1, no. 1, pp. 5–11, 2014.

[3] H. Lipson, "Challenges and opportunities for design, simulation, and fabrication of soft robots," *Soft Robotics*, vol. 1, no. 1, pp. 21–27, 2014.

[4] W. Felt, S. Lu, and C. D. Remy, "Modeling and design of smart braid inductance sensors for fiber-reinforced elastomeric enclosures," *IEEE Sensors Journal*, vol. 18, no. 7, pp. 2827–2835, 2018.

[5] W. Felt, M. J. Telleria, T. F. Allen, G. Hein, J. B. Pompa, K. Albert, and C. D. Remy, "An inductance-based sensing system for bellows-driven continuum joints in soft robots," *Autonomous Robots*, pp. 1–14, 2017.

[6] S. Yang and N. Lu, "Gauge factor and stretchability of silicon-on-polymer strain gauges," *Sensors*, vol. 13, no. 7, pp. 8577–8594, 2013.

[7] D.-H. Kim, N. Lu, R. Ma, Y.-S. Kim, R.-H. Kim, S. Wang, J. Wu, S. M. Won, H. Tao, A. Islam, *et al.*, "Epidermal electronics," *science*, vol. 333, no. 6044, pp. 838–843, 2011.

[8] C. Della Santina, M. Bianchi, G. Grioli, F. Angelini, M. Catalano, M. Garabini, and A. Bicchi, "Controlling soft robots: balancing feedback and feedforward elements," *IEEE Robotics & Automation Magazine*, vol. 24, no. 3, pp. 75–83, 2017.

[9] J. Bishop-Moser and S. Kota, "Design and modeling of generalized fiber-reinforced pneumatic soft actuators," *IEEE Transactions on Robotics*, vol. 31, no. 3, pp. 536–545, 2015.

[10] D. Bruder, A. Sedal, R. Vasudevan, and C. D. Remy, "Force generation by parallel combinations of fiber-reinforced fluid-driven actuators," *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 3999–4006, Oct 2018.

[11] F. Renda, M. Giorelli, M. Calisti, M. Cianchetti, and C. Laschi, "Dynamic model of a multibending soft robot arm driven by cables," *IEEE Transactions on Robotics*, vol. 30, no. 5, pp. 1109–1122, 2014.

[12] W. Felt and C. D. Remy, "A closed-form kinematic model for fiber-reinforced elastomeric enclosures," *Journal of Mechanisms and Robotics*, vol. 10, no. 1, p. 014501, 2018.

[13] S. Neppalli, M. A. Csencsits, B. A. Jones, and I. D. Walker, "Closed-form inverse kinematics for continuum manipulators," *Advanced Robotics*, vol. 23, no. 15, pp. 2077–2091, 2009.

[14] R. J. Webster III and B. A. Jones, "Design and kinematic modeling of constant curvature continuum robots: A review," *The International Journal of Robotics Research*, vol. 29, no. 13, pp. 1661–1683, 2010.

[15] B. A. Jones and I. D. Walker, "Kinematics for multisection continuum robots," *IEEE Transactions on Robotics*, vol. 22, no. 1, pp. 43–55, 2006.

[16] T. George Thuruthel, Y. Ansari, E. Falotico, and C. Laschi, "Control strategies for soft robotic manipulators: A survey," *Soft robotics*, vol. 5, no. 2, pp. 149–163, 2018.

[17] I. A. Gravagne, C. D. Rahn, and I. D. Walker, "Large deflection dynamics and control for planar continuum robots," *IEEE/ASME transactions on mechatronics*, vol. 8, no. 2, pp. 299–307, 2003.

[18] D. Trivedi, A. Lotfi, and C. D. Rahn, "Geometrically exact models for soft robotic manipulators," *IEEE Transactions on Robotics*, vol. 24, no. 4, pp. 773–780, 2008.

[19] D. Bruder, A. Sedal, J. Bishop-Moser, S. Kota, and R. Vasudevan, "Model based control of fiber reinforced elastofluidic enclosures," in *Robotics and Automation (ICRA), 2017 IEEE International Conference on*. IEEE, 2017, pp. 5539–5544.

[20] A. Sedal, D. Bruder, J. Bishop-Moser, R. Vasudevan, and S. Kota, "A constitutive model for torsional loads on fluid-driven soft robots," in *ASME 2017 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*. American Society of Mechanical Engineers, 2017, pp. V05AT08A016–V05AT08A016.

[21] J. Bishop-Moser, G. Krishnan, C. Kim, and S. Kota, "Design of soft robotic actuators using fluid-filled fiber-reinforced elastomeric enclosures in parallel combinations," in *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*. IEEE, 2012, pp. 4264–4269.

[22] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.

[23] M. T. Gillespie, C. M. Best, E. C. Townsend, D. Wingate, and M. D. Killpack, "Learning nonlinear dynamic models of soft robots for model predictive control with neural networks," in *2018 IEEE International Conference on Soft Robotics (RoboSoft)*. IEEE, 2018.

[24] L. Ljung, *System identification: theory for the user*. Prentice-hall, 1987.

[25] M. Budišić, R. Mohr, and I. Mezić, "Applied koopmanism," *Chaos: An Interdisciplinary Journal of Nonlinear Science*, vol. 22, no. 4, p. 047510, 2012.

[26] M. O. Williams, I. G. Kevrekidis, and C. W. Rowley, "A data–driven approximation of the koopman operator: Extending dynamic mode decomposition," *Journal of Nonlinear Science*, vol. 25, no. 6, pp. 1307–1346, 2015.

[27] A. Mauroy and J. Goncalves, "Linear identification of nonlinear systems: A lifting technique based on the koopman operator," *arXiv preprint arXiv:1605.04457*, 2016.

[28] ——, "Koopman-based lifting techniques for nonlinear systems identification," *arXiv preprint arXiv:1709.02003*, 2017.

[29] I. Abraham, G. de la Torre, and T. Murphey, "Model-based control using koopman operators," in *Proceedings of Robotics: Science and Systems*, Cambridge, Massachusetts, July 2017.

[30] A. Lasota and M. C. Mackey, *Chaos, fractals, and noise: stochastic aspects of dynamics*. Springer Science & Business Media, 2013, vol. 97.

[31] N. J. Higham, *Functions of matrices: theory and computation*. Siam, 2008, vol. 104.

[32] MATLAB, *version 7.10.0 (R2017a)*. Natick, Massachusetts: The MathWorks Inc., 2017.