A method to construct reduced-order parameter varying models

Jennifer Annoni* and Peter Seiler

Aerospace Engineering and Mechanics, University of Minnesota

SUMMARY

This paper describes a method to construct reduced-order models for high-dimensional nonlinear systems. It is assumed that the nonlinear system has a collection of equilibrium operating points parameterized by a scheduling variable. First, a reduced-order linear system is constructed at each equilibrium point using state, input, and output data. This step combines techniques from proper orthogonal decomposition, dynamic mode decomposition, and direct subspace identification. This yields discrete-time models that are linear from input to output but whose state matrices are functions of the scheduling parameter. Second, a parameter varying linearization is used to connect these linear models across the various operating points. The key technical issue in this second step is to ensure the reduced-order linear parameter varying system approximates the nonlinear system even when the operating point changes in time. Copyright © 2016 John Wiley & Sons, Ltd.

Received …

KEY WORDS: dynamic mode decomposition; reduced-order modeling; parameter varying

1. INTRODUCTION

This paper describes a method to construct a reduced-order model for high-dimensional nonlinear systems. The work is motivated by the control of systems that involve fluid and/or structural dynamics. One specific example is wind farm control. The overall performance of a wind farm can be improved through proper coordination of the turbines [1]. High-fidelity computational fluid dynamic models have been developed for wind farms [2, 3]. These high-fidelity models are accurate but are not suitable for controller design due to their computational complexity. A second example involves the control of flexible aircraft. More fuel efficient aircraft can be designed by reducing structural weight thus leading to increased flexibility [4]. The vibrational modes can significantly degrade the performance and can even lead to aeroelastic instabilities (flutter) [4, 5]. High-fidelity, computational fluid/structural models also exist for this application [6, 7] but these are also too complex for control design. Simplified, control-oriented models are needed in both of these examples.

A variety of reduced-order modeling techniques have been developed by the fluid dynamics and controls communities. Several of these methods are summarized in Section 2. These methods range from analytical reduced-order modeling, such as balanced truncation [8], to data-driven reduced-order modeling such as system identification [9] where a low-dimensional system is identified to describe the dynamics of a high-dimensional system. Subspace techniques have been applied to balanced truncation to perform model reduction on nonlinear systems [10]. In the fluid dynamics literature, proper orthogonal decomposition (POD) is a standard method where the state is projected onto a low-dimensional subspace of POD modes constructed using data from the high-order system.
Dynamic mode decomposition (DMD) is a more recent approach that fits time-domain data with linear dynamics on a reduced-order subspace [14–16]. This approach has ties to the Koopman operator [17, 18]. In their basic form, both approaches are for autonomous (unforced) systems. An extension of DMD was developed in [19] known as DMD with control (DMDc) to construct reduced-order models with control inputs.

This paper describes an extension of DMDc to construct reduced-order linear parameter varying (LPV) models that approximate high-order nonlinear models. The nonlinear system is assumed to have a parameterized collection of equilibrium operating points. For example, the free-stream wind speed and wind direction parameterizes the equilibrium condition in a wind farm. Related work on this particular topic has been done looking at flexible aircraft [20–22]. Many studies use linear methods such as Krylov methods [23, 24]. The proposed approach in this paper involves two steps. First, POD and direct subspace identification are combined to construct an input-output reduced-order model (IOROM). This terminology has been used in [20]. This step is similar to DMDc [19], which can be used to construct a reduced-order linear model at one operating condition. Specifically, this step uses direct N4SID subspace identification [9] on a low dimensional subspace spanned by POD modes. Second, the reduced-order models constructed at fixed operating conditions are “stitched” together using a parameter varying linearization (Section 3.2). The key technical issue is that the states of the reduced-order model must have a consistent meaning across all operating conditions (as described in Section 3.2.2). This state consistency issue has been addressed in other studies [25]. The difference in this paper is that the parameter is allowed to vary in time for a given simulation. The approach used in the paper handles this issue by constructing a single reduced-order subspace that is used at all operating conditions. This approach and the LPV linearization method are demonstrated on a nonlinear mass-spring-damper example with 200 states as well as a nonlinear actuator disk example often used in wind farm control literature with more than 20,000 states (Section 4). Lastly, Section 5 will address conclusions and future work.

2. BACKGROUND

There are a variety of methods in the controls and fluid dynamics literature that attempt to construct low-order models to describe the dominant dynamics of a high-dimensional system. The goal of this paper is to develop a method that can achieve the following:

- **Handles > 10^5 states:** The approach can be used to develop low-order representations of fluid dynamic problems. Typical fluid dynamic problems can have more than a million states.
- **Handle inputs and outputs:** The objective is to develop low-order models for control design on systems with controllable inputs and measurable outputs, e.g. wind farms and flexible aircraft.
- **Adjoint-free:** Some existing methods require the use of a model adjoint. Our goal is to avoid the use of such adjoints so that the proposed method can be applied to either experimental data or to simulation models.
- **Reduced-order states have physical meaning:** The state of the reduced-order system can be used to approximately reconstruct the full-order state and can assist in the state consistency issue faced by parameter varying systems.
- **Can be used for parameter varying systems:** This is important for nonlinear systems where the dynamics change significantly over the entire operating range, e.g. wind farms and flexible aircraft.

The remainder of this section will briefly address several existing methods as well as their advantages and limitations with respect to the specific criteria defined above. These methods range from analytical approaches such as balanced truncation to data-driven approaches that construct a low-dimensional representation of a high-dimensional system.
2.1. Balanced Truncation

Balanced truncation is a typical model reduction approach used in the controls community [8]. To perform balanced truncation on a linear system, the controllability/observability Gramians need to be computed to understand the influence of the states on the inputs and outputs of the system. The details on this implementation can be found in [8]. The controllability Gramian specifies the minimum control energy required to reach any specific state. States that require less energy to reach are more controllable and hence have a greater influence on the input/output dynamics. Similarly, the observability Gramian specifies the energy in the output measurement when the system evolves from a given initial state (with zero input). States that produce more energy in the output are more observable and hence have a greater influence on the input/output dynamics. The Gramians can be computed by solving the Lyapunov equations. A state coordinate transformation can be constructed to align the controllability and observability Gramians. This allows the most controllable/observable states to be retained. Under this transformed system, the states that are significantly influenced by the inputs are also states that have a significant impact on the outputs. With respect to the overall objectives defined at the beginning of this section, there are advantages and limitations to using this approach. Specifically, balanced truncation is adjoint-free and can handle inputs and outputs. However, this method becomes intractable for large systems (states > 1000) as it requires the solution of the two Lyapunov equations. In addition, the extension of balanced truncation to parameter varying systems requires the solutions of generalized Gramians via semi-definite programming optimizations [26, 27]. This optimization is even more computationally intensive than solving Lyapunov equations.

2.2. System Identification

2.2.1. Numerical Algorithm for Subspace State Space System Identification (N4SID) The idea of N4SID is to use input/output data to form a low-dimensional state-space representation of high-dimensional system. This is done by collecting data from exciting the system using a sine sweep, chirp, etc. Specifically, this is a black-box approach where the inputs and outputs of the system are recorded and used to construct a low-order ordinary differential equation that approximately describes the relationship between the input(s) and the output(s). The data can be used to estimate the low-order states of the system along with the state space matrices associated with the low-order system. The details on the implementation can be found in [9]. This approach is focused on preserving the input-output relationship, which means that it can provide a low-order representation of a high-dimensional system (states > 10^5). In addition, this approach does not require an adjoint. However, the states do not have physical meaning which would help provide some physical intuition about the system. Lastly, when we start addressing parameter varying systems, there is a state consistency issue that stems from the lack of physical meaning in the states.

2.2.2. Eigensystem Realization Algorithm Another system identification technique is known as Eigensystem Realization Algorithm (ERA), which was developed by Juang in [28]. Assuming a linear system, output data is collected in response to an input impulse. Similar to N4SID, this data is used to construct a low-order representation of the state space matrices. The specific details on the implementation can be found in [28], ERA is a special case of system identification and has been shown to be related to dynamic mode decomposition in [16] and has been related to balanced proper orthogonal decomposition in [29]. Similar to N4SID, ERA is also able to provide a low-order representation of a system with many states (states > 10^3). ERA can also handle inputs and outputs and is adjoint-free. The additional benefit is that, for linear systems, ERA can compute the modes and eigenvalues of a system providing some physical intuition about the system. However, as noted above, ERA was developed for linear systems and has difficulty identifying nonlinear systems. Lastly, it has been noted that ERA cannot handle many inputs [16] and there is still a state consistency issue when using ERA.

2.2.3. Linear Parameter Varying Subspace Identification It should be noted that LPV subspace identification has been investigated recently [30–32]. For example, ERA has been extended to
periodic parameter varying systems with observer/kalman filter identification in [33]. This takes care of the state consistency issue exhibited in N4SID and ERA. However, significant amounts of input/output data are required to characterize the response of the system. In addition, the states do not have physical meaning. This method is a good alternative approach when state information is not known. However, in CFD simulations and certain experiments including particle image velocimetry (PIV) [34], the high-dimensional states are known and the state information combined with input/output data can be used to provide a more accurate representation of the system.

2.3. Proper Orthogonal Decomposition

POD provides a low-order approximation of a fluid flow that is capable of capturing the dominant structures in the flow. Specifically, POD can be used to extract dominant spatial features from both simulation and experimental data that is used to uncover the structures in the flow field [11–13]. These dominant structures, or POD modes, can be computed from snapshots of the nonlinear system. A singular value decomposition is done of the snapshots to obtain the POD modes. Additional technical details can be found in [11–13]. The computed POD modes can be used to construct a reduced-order nonlinear model of the flow using the Galerkin projection [10, 13]. In addition, the POD/Galerkin approach has been used in the context of control [35]. This approach can handle many states and is adjoint-free. This approach has been extended to parameter varying systems [36]. Despite the connection of POD/Galerkin to controls, there have been many studies done that show that POD/Galerkin is sensitive inputs and to the data used [37] Also, it can often produce unstable results near stable equilibrium [38].

2.4. Balanced POD

For some systems, a more relevant objective is to compute a low-order model that captures the most relevant flow dynamics that connect the input to the output [40]. For this particular approach, data is collected by performing one forward simulation for each input. In addition, the adjoint of the system has to be formulated and run for each output. Physically, the adjoint system is used to evaluate the sensitivity of the system due to the small perturbations near the output measurements of the system [41]. Snapshots from the forward and adjoint systems can be used to compute the empirical Gramians. The BPOD modes can be computed from these Gramians. BPOD is similar to balanced truncation, but does not require the solution to the Lyapunov equations, which become prohibitive as the number of states increases. By using snapshots from the forward and adjoint simulations, the transformations required to balance the system can be formed at a reasonable cost. More technical details can be found in [40]. This approach can handle many states (states > $10^5$). In addition, it can handle inputs and outputs. However, BPOD has not been extended to parameter varying systems and requires an adjoint. In addition, a linearized simulation needs to be run for every input and the adjoint system needs to be run for every output. BPOD has been compared to ERA in [29]. Specifically, it has been shown that ERA can produce the same balanced models as BPOD without the adjoint. However, standard ERA was developed for linear systems.

2.5. Dynamic Mode Decomposition

Like POD and BPOD, DMD extracts the dominant spatial and temporal information about the flow [14–16]. This method attempts to fit a discrete-time linear system to a set of snapshots from simulation or experiments. The DMD method attempts to fit the snapshots in time using a low rank matrix, obtained through a subspace projection, that captures the dynamics of the dataset. The POD modes of the snapshots are typically chosen as the projection subspace. More technical details are available in [14]. This matrix can be used to construct the DMD spatial modes that correspond to specific temporal frequencies. DMD has been connected to the Koopman operator and can be used to describe nonlinear systems [42, 43]. This method can handle many states (states > $10^5$) and is adjoint-free. The reduced-order states have a physical meaning, i.e. the reduced-order model can be used to construct the approximate solutions to the full-order model using the projection subspace. The standard DMD approach cannot produce input-output models. However, DMD has
been extended to include inputs and outputs in [19]. However, DMD has not been extended to parameter varying systems.

3. REDUCED-ORDER MODELING APPROACH FOR PARAMETER VARYING SYSTEMS

This section describes, in detail, the proposed method that accomplishes the criteria specified in the beginning of Section 2. The proposed method combines approaches in the fluid dynamics and controls communities.

3.1. Input-Output Reduced-Order Modeling

The approach will be summarized in this subsection for time-invariant (LTI) systems at a single operating point, but will be extended to include LPV models in Section 3.2. This paper addresses the use of this technique to develop an input-output reduced-order model (IOROM) that does not require adjoints and where the states have some physical meaning. In this section, an extension of DMDc [19] is presented as a combination of POD and system identification. It should be noted that DMD modes can be constructed from the reduced-order model. This paper will focus on the construction of the reduced-order model rather than the evaluation of the DMD modes.

Consider a discrete-time nonlinear system:

\[ x_{k+1} = f(x_k, u_k) \]  
\[ y_k = h(x_k, u_k) \]

where \( x \in \mathbb{R}^{n_x}, u \in \mathbb{R}^{n_u}, \) and \( y \in \mathbb{R}^{n_y} \) are the state, input, and output vectors.

A collection of snapshot measurements are obtained via simulation or experiments by exciting the system. Snapshots are taken from the nonlinear system and the states, inputs, and outputs are recorded as:

\[ X_0 = \begin{bmatrix} x_0 & x_1 & \ldots & x_{m-1} \end{bmatrix} \in \mathbb{R}^{n_x \times n_s} \]  
\[ X_1 = \begin{bmatrix} x_1 & x_2 & \ldots & x_m \end{bmatrix} \in \mathbb{R}^{n_x \times n_s} \]  
\[ U_0 = \begin{bmatrix} u_0 & u_1 & \ldots & u_{m-1} \end{bmatrix} \in \mathbb{R}^{n_u \times n_s} \]  
\[ Y_0 = \begin{bmatrix} y_0 & y_1 & \ldots & y_{m-1} \end{bmatrix} \in \mathbb{R}^{n_y \times n_s} \]

where \( n_s \) is the number of snapshots.

The proposed variation of DMDc attempts to fit the snapshot measurements to a linear discrete-time system:

\[ x_{k+1} = Ax_k + Bu_k \]  
\[ y_k = Cx_k + Du_k \]

The state matrices \((A, B, C, D)\) have the dimensions compatible to those of \((x, u, y)\). The matrices \((A, B, C, D)\) can be constructed to fit the snapshot data with minimal least-squares cost as in direct N4SID [9]. However, this becomes intractable for large systems. Typical fluid dynamic systems have on the order of millions of states. The proposed solution is to project the state onto a low-dimensional subspace to make the least squares computation tractable. Let \( Q \in \mathbb{R}^{n_x \times r} \) with \( r < n \) have columns that form an orthonormal basis for a projection subspace. The IOROM is expressed in terms of the projected state \( z := Q^T x \in \mathbb{R}^r \) is given by:

\[ z_{k+1} = (Q^T AQ)z_k + (Q^T B)u_k := Fz_k + Gu_k \]  
\[ y_k = (CQ)z_k + Du_k := Hz_k + Du_k \]

The matrices in the reduced-order system have dimensions $F \in \mathbb{R}^{r \times r}$, $G \in \mathbb{R}^{r \times n_u}$, and $H \in \mathbb{R}^{n_y \times r}$. The form of (9) is equivalent to the following low rank approximations for the full-order state matrices:

$$
\begin{bmatrix}
A & B \\
C & D
\end{bmatrix} \approx 
\begin{bmatrix}
QFQ^T & QG \\
HQ^T & D
\end{bmatrix} = 
\begin{bmatrix}
Q & 0 \\
0 & I_{n_y}
\end{bmatrix}
\begin{bmatrix}
F & G \\
H & D
\end{bmatrix}
\begin{bmatrix}
Q^T & 0 \\
0 & I_{n_u}
\end{bmatrix}
$$

(11)

The optimal choice for the reduced-order state matrices $(F, G, H, D)$ can be computed given the subspace spanned by $Q$. This is a similar setup for standard DMD as in in [14–16].

The optimal (reduced-order) state matrices for a given projection space $Q$ are obtained by minimizing the error of the Frobenius norm:

$$
\min_{\begin{bmatrix}
F & G \\
H & D
\end{bmatrix}} \| X_1 Y_1 - \begin{bmatrix}
Q & 0 \\
0 & I
\end{bmatrix}
\begin{bmatrix}
F & G \\
H & D
\end{bmatrix}
\begin{bmatrix}
Q^T & 0 \\
0 & I
\end{bmatrix} U_0 \|_F^2
$$

(12)

As mentioned previously, this is the direct N4SID subspace method for estimating state matrices given measurements of the (reduced-order) state, input, and output. A sub-optimal, but useful, choice for the projection space is given by the POD modes of $X_0$. The POD modes can be computed by taking the singular value decomposition of $X_0 = U \Sigma V^T$. The POD modes are contained in the columns of $U$ and the relative energy of each mode is contained in the singular values of $\Sigma$. These modes provide the spatial component of the flow with the first POD mode being the spatial mode that contains the most energy. The state of the linear system can be approximated on a subspace defined by the first $r$ POD modes of $X_0$, i.e. $Q := U_r$. The optimal reduced-order state matrices for this choice are given by:

$$
\begin{bmatrix}
F & G \\
H & D
\end{bmatrix}_{\text{opt}} = 
\begin{bmatrix}
(U_r^T X_1) \\
Y_0
\end{bmatrix}
\Sigma_r V_r^T \dagger
$$

(13)

As noted earlier, the primary focus of this paper is developing an algorithm that can achieve the criteria in the beginning of Section 2. However, it is important to note that, as with standard DMD, an eigenvalue decomposition of $F_{\text{opt}}$ can be used to construct DMD modes. In other words, the eigenvalue decomposition of $F_{\text{opt}}$ can be used to construct spatio-temporal modes at a specific temporal frequency for the system. This can be important for control design as was shown in [44].

This new methodology also yields input-output information for the model. This proposed method is a tractable implementation of the existing direct N4SID (subspace) method that can be applied to very large systems. This is not simply a black-box (input-output) approach because the state of the reduced-order system $z_k$ can be used to approximately reconstruct the full-order state by:

$$
x_k \approx U_r z_k
$$

(14)

Moreover, the approach requires input/output/state data from the system. Construction and simulation of an adjoint system is not required. The following section will show how to extend this approach to parameter varying systems.

One way to choose the order of the IOROM computed using this approach is to analyze how much energy is captured by the number of modes chosen. It is common to choose the number of modes to capture 99% of the energy (or some similar threshold) in the snapshots. However, for a sufficient model that is suitable for control, the primary metric is the amount of model error incurred from the selection of the number of modes. The model error can be computed using the Frobenius norm:

$$
\| X_1 Y_1 - \begin{bmatrix}
A & B \\
C & D
\end{bmatrix} U_0 \|_F^2
$$

(15)
Again, computation of this model error is intractable for systems with extremely large state dimension. However, the properties of the Frobenius norm can be used to equivalently rewrite this error in a more useful form:

$$\left\| X_1 \right\|_F^2 = \left\| \begin{bmatrix} X_1^T X_1 \\ X_0^T X_0 \end{bmatrix} \right\|_F^2 - \left\| \begin{bmatrix} Q^T X_1 \\ Q^T X_0 \end{bmatrix} \right\|_F^2$$

(16)

The first term represents the model error on the projected subspace. The second and third terms represent the energy lost in the snapshot data $X_1$ by using the projection. Increasing the number of modes will generally decrease the total error. However, there is a point when adding additional modes will not significantly improve the model error. In fact, adding the modes beyond this point could result in a model that is trying to overfit to the nonlinearities in the system resulting in a degradation of the performance.

3.2. Parameter Varying Approach

This subsection extends the proposed approach in Section 3.1 using a parameter varying linearization similar to the one in [45].

3.2.1. Linearization

The parameter varying formulation starts from a nonlinear system that includes an exogenous input, $\rho$:

$$x_k+1 = f(x_k, u_k, \rho_k)$$

$$y_k = h(x_k, u_k, \rho_k)$$

(17)

(18)

The parameter $\rho$ is an exogenous signal that specifies the operating condition. For example, $\rho$ could be a vector of the mean wind speed and direction in the wind farm application or altitude and Mach number for the flexible aircraft example.

Next, assume there is a collection of equilibrium points $(\bar{x}(\rho), \bar{u}(\rho), \bar{y}(\rho))$ such that

$$\bar{x}(\rho) = f(\bar{x}(\rho), \bar{u}(\rho), \rho)$$

$$\bar{y}(\rho) = h(\bar{x}(\rho), \bar{u}(\rho), \rho)$$

(19)

If the state is initialized at $x_0 = \bar{x}(\rho)$, the input is held fixed at $u_k = \bar{u}(\rho)$, and the operating condition is frozen at $\rho_k = \rho$ then the state and output will remain in equilibrium at $x_k = \bar{x}(\rho)$ and $y_k = \bar{y}(\rho)$ for $k = 0, 1, \ldots$. Thus $(\bar{x}(\rho), \bar{u}(\rho), \bar{y}(\rho))$ defines an equilibrium condition for each fixed value of $\rho$.

The nonlinear dynamics can be linearized around the equilibrium points defined for each fixed value of $\rho$. Define perturbations from the equilibrium condition as:

$$\delta x_k := x_k - \bar{x}(\rho), \quad \delta u_k := u_k - \bar{u}(\rho), \quad \delta y_k := y_k - \bar{y}(\rho)$$

(20)

For a fixed operating condition ($\rho_k = \rho$ for $k = 0, 1, \ldots$), a standard linearization yields:

$$\delta x_{k+1} = A(\rho)\delta x_k + B(\rho)\delta u_k$$

$$\delta y_k = C(\rho)\delta x_k + D(\rho)\delta u_k$$

(21)

where the linearized state matrices are defined by

$$A(\rho) := \left. \frac{\partial f}{\partial x} \right|_{(\bar{x}(\rho), \bar{u}(\rho), \rho)} \quad B(\rho) := \left. \frac{\partial f}{\partial u} \right|_{(\bar{x}(\rho), \bar{u}(\rho), \rho)}$$

$$C(\rho) := \left. \frac{\partial h}{\partial x} \right|_{(\bar{x}(\rho), \bar{u}(\rho), \rho)} \quad D(\rho) := \left. \frac{\partial h}{\partial u} \right|_{(\bar{x}(\rho), \bar{u}(\rho), \rho)}$$

(22)
The matrices \((A(\rho), B(\rho), C(\rho), D(\rho))\) define a collection of parameterized LTI systems defined for fixed operating conditions, \(\rho\). Here, we consider the more general case where the operating condition, specified by the parameter \(\rho_k\), varies in time.

Next consider the case where the operating condition, specified by the parameter \(\rho_k\), varies in time. In general \((x_k, u_k, y_k) := (\bar{x}(\rho_k), \bar{u}(\rho_k), \bar{y}(\rho_k))\) is not a valid solution of the nonlinear system. In other words, the parameterized state/input/output values only define an equilibrium condition for fixed values of \(\rho\). Despite this fact, it is still possible to construct a time varying linearization around the parameterized values \((\bar{x}(\rho), \bar{u}(\rho), \bar{y}(\rho))\). Re-define perturbation variables as follows for the case where \(\rho\) varies in time:

\[
\begin{align*}
\delta x_k := x_k - \bar{x}(\rho_k), & \quad \delta u_k := u_k - \bar{u}(\rho_k), & \quad \delta y_k := y_k - \bar{y}(\rho_k)
\end{align*}
\]

A valid (time-varying) linearization can be obtained when \(\rho_k\) is time-varying as follows:

\[
\delta x_{k+1} = f(x_k, u_k, \rho_k) - \bar{x}(\rho_{k+1})
\]

A Taylor series expansion of \(f(x_k, u_k, \rho_k)\) yields:

\[
\begin{align*}
& f(\bar{x}_k(\rho_k) + \delta x_k, \bar{u}_k(\rho_k) + \delta u_k, \rho_k) \approx \bar{x}(\rho_k) + A(\rho_k)\delta x_k + B(\rho_k)\delta u_k \\
& f(x_k, u_k, \rho_k) - \bar{x}(\rho_k) \approx A(\rho_k)\delta x_k + B(\rho_k)\delta u_k \\
& A(\rho_k)\delta x_k + B(\rho_k)\delta u_k
\end{align*}
\]

where \(A\) and \(B\) are as defined in (22). A similar Taylor series approximation can be performed to linearize the output function \(h\) in terms of the matrices \(C\) and \(D\) defined in (22). Combining (24) and (25) leads to the following parameter varying linearization:

\[
\begin{align*}
\delta x_{k+1} &= A(\rho_k)\delta x_k + B(\rho_k)\delta u_k + (\bar{x}(\rho_k) - \bar{x}(\rho_{k+1})) \\
\delta y_k &= C(\rho_k)\delta x_k + D(\rho_k)\delta u_k
\end{align*}
\]

This differs from the standard linearization at a single fixed operating point in two respects. First, \(\rho_k\) varies in time and hence this is a time-varying system. More precisely, the time variations in the state matrices \((A, B, C, D)\) arise due to \(\rho_k\) and hence this is called a linear parameter varying (LPV) system. There are many tools in the controls literature that address this class of systems [46–49]. Second, the equilibrium state varies in time due to the changing operating condition. This effect is retained by the term \(\bar{x}(\rho_k) - \bar{x}(\rho_{k+1})\) which provides a forcing term for the dynamics. This model linearizes the dependence on the state and the input. The linearization approximation is accurate as long as the state, input, and output remain near the manifold of equilibrium points \((\bar{x}(\rho_k), \bar{u}(\rho_k), \bar{y}(\rho_k))\). It is important that the nonlinear dependence on the operating condition (specified by \(\rho_k\)) is retained.

3.2.2. State Consistency Issue Parameter varying linearizations can be constructed using data from steady operating conditions specified by constant values of \(\rho\). Specifically, the linearization only requires state matrices \((A(\rho), B(\rho), C(\rho), D(\rho))\) and equilibrium conditions \((\bar{x}(\rho), \bar{u}(\rho), \bar{y}(\rho))\) at each fixed value of \(\rho\). In principle, the proposed IOROM method or another model reduction method can be used to identify reduced-order state matrices at each operating point, \(\rho\). One key technical issue is state consistency.

To clarify this issue, consider an autonomous system without inputs and outputs. Let \(A(\rho)\) denote the state matrix that appears in the parameter varying linearization. DMD can be used at each fixed value of \(\rho\) to identify a subspace spanned by \(Q(\rho) \in \mathbb{R}^{n_x \times r(\rho)}\) and a reduced-order matrix \(F(\rho) \in \mathbb{R}^{r(\rho) \times r(\rho)}\) such that \(A(\rho) \approx Q(\rho)F(\rho)Q(\rho)^T\). The reduced-order state at the operating point \(\rho\) is defined as \(z := Q(\rho)^T \delta x \in \mathbb{R}^{r(\rho)}\). This reduced-order state, \(z\), will lack consistency if the projection subspace \(Q(\rho)\) depends on the parameter. In other words, the meaning of \(z\) and dimension \(r(\rho)\) will depend on the value of \(\rho\). Hence the state, \(z\), at \(\rho_1\) will not, in general, be consistent in either meaning or dimension with the state \(z\) at \(\rho_2 \neq \rho_1\).

To circumvent this issue, a single consistent subspace \(Q \in \mathbb{R}^{n_x \times r}\) should be used at all operating conditions. The reduced-order state \(z := QT \delta x \in \mathbb{R}^{r}\) then has a consistent meaning for all parameter values. Moreover, a reduced-order matrix \(F(\rho) \in \mathbb{R}^{r \times r}\) can be identified at each value of \(\rho\) such that
Algorithm 1 Reduced-Order Parameter Varying Linearization

1: **Given:** Collection of parameter grid points \((p_g^j)_{j=1}^{n_g}\).

2: **Data Collection:** At each grid point \(j = 1, \ldots, n_g\) do the following:

3: **Equilibrium:** Compute the equilibrium condition \((\bar{x}(p_g^j), \bar{u}(p_g^j), \bar{y}(p_g^j))\) at \(\rho = p_g^j\).

4: **Experiment:** Excite the nonlinear system (17) with fixed \(\rho_k = p_g^j\). The initial condition \(x_0\) and input \(u_0\) should be near the equilibrium condition \((\bar{x}(p_g^j), \bar{u}(p_g^j))\).

5: **Snapshots:** Define the matrices of snapshot deviations from the equilibrium conditions at \(p_g^j:\)

\[
X_0(p_g^j) := \begin{bmatrix} x_0 - \bar{x}(p_g^j), & \ldots, & x_{n_x-1} - \bar{x}(p_g^j) \end{bmatrix} \in \mathbb{R}^{n_x \times n_x}
\]

(28)

\[
X_1(p_g^j) := \begin{bmatrix} x_1 - \bar{x}(p_g^j), & \ldots, & x_{n_x} - \bar{x}(p_g^j) \end{bmatrix} \in \mathbb{R}^{n_x \times n_x}
\]

(29)

\[
U_0(p_g^j) := \begin{bmatrix} u_0 - \bar{u}(p_g^j), & \ldots, & u_{n_s-1} - \bar{u}(p_g^j) \end{bmatrix} \in \mathbb{R}^{n_u \times n_s}
\]

(30)

\[
Y_0(p_g^j) := \begin{bmatrix} y_0 - \bar{y}(p_g^j), & \ldots, & y_{n_s-1} - \bar{y}(p_g^j) \end{bmatrix} \in \mathbb{R}^{n_y \times n_s}
\]

(31)

6: **Construct Subspace for State Reduction:**

7: **Stack Data:** Define matrix of all state data: \(X_0 := [X_0(p_g^1), \ldots, X_0(p_g^{n_g})] \in \mathbb{R}^{n_x \times (n_x n_g)}\).

8: **POD Modes:** Compute POD modes of \(X_0\).

9: **Subspace:** Choose \(r\) modes, denoted \(Q \in \mathbb{R}^{n_x \times r}\), to capture sufficient energy in \(X_0\).

10: **Reduced-Order State Matrices:** At each grid point \(j = 1, \ldots, n_g\) do the following:

11: **IOROM:** Use snapshot data \((X_0(p_g^j), X_1(p_g^j), U_0(p_g^j), Y_0(p_g^j))\) and subspace \(Q\) to compute state matrices \((F(p_g^j), G(p_g^j), H(p_g^j), D(p_g^j))\) via the IOROM steps in Section 3.1.

\[
A(\rho) = QF(\rho)Q^T. \] This would lead to a reduced-order parameter varying linearization of the form:

\[
z_{k+1} = F(\rho_k)z_k + (\bar{z}(\rho_k) - \bar{z}(\rho_{k+1}))
\]

(27)

where \(\bar{z}(\rho) := Q^T \bar{x}(\rho)\) is the equilibrium state projected onto the reduced subspace. The next section presents a method to address this state consistency issue.

3.3. Reduced-Order Parameter Varying Linearizations

One proposed approach for constructing an IOROM for a nonlinear system (1) is shown in Algorithm 1. First, a collection of parameter values are selected (Line 1). Second, data is collected from the nonlinear system at these selected parameter values (Lines 2-5). The collected data includes the equilibrium conditions as well as state/input/output snapshots obtained by exciting the nonlinear system. The algorithm, as written, assumes that the same number of snapshots \(n_s\) are obtained at each grid point. However, the number of snapshots can easily change with each grid point. Third, a single \(r\)-dimensional subspace of the state space is constructed (Lines 6-9). The subspace, defined by an orthogonal matrix \(Q \in \mathbb{R}^{n_x \times r}\), is constructed from the POD modes of the snapshots obtained from all parameter values. Fourth, reduced-order state matrices are computed at each parameter vector using the IOROM approach described in Section 3.1 (Lines 10-11).

The outcome of Algorithm 1 is a single \(r\)-dimensional subspace \(Q\) as well as equilibrium conditions and reduced-order state matrices computed at the selected grid points. This yields a reduced-order parameter varying linearization of the nonlinear system of the form:

\[
z_{k+1} = F(\rho_k)z_k + G(\rho_k)\delta u_k + (\bar{z}(\rho_k) - \bar{z}(\rho_{k+1}))
\]

\[
\delta y_k = H(\rho_k)z_k + D(\rho_k)\delta u_k
\]

(32)

where \(z := Q^T \delta x \in \mathbb{R}^r\) is the reduced-order state and \(\bar{z}(\rho) := Q^T \bar{x}(\rho) \in \mathbb{R}^r\) is the reduced-order equilibrium state. The single subspace defined by \(Q\) is used to construct state matrices at all
Algorithm 2 Hybrid POD/Gram-Schmidt approach to Construct Subspace

1: **Given:** Collection of parameter grid points \( \{p^j\}_{j=1}^{n_p} \) and snapshots \( X_0(p^j_g) \) from each grid point.
2: **Initial Point:** Use standard POD to compute \( r_1 \) modes \( Q(p^j_g) \in \mathbb{R}^{n_x \times r_1} \) to capture sufficient energy in \( X_0(p^j_g) \).
3: **Iterative Processing:** For \( j = 2, \ldots, n_g \), iteratively compute additional modes at each grid point. Given \( Q_1 := [Q(p^1_g), \ldots, Q(p^{n_g}_g)] \), use iterative POD to compute \( r_g \) additional modes \( Q(p^j_g) \in \mathbb{R}^{n_x \times r_g} \) to capture sufficient energy in \( X_0(p^j_g) \).
4: **Subspace:** Stack modes to form a single subspace defined by \( Q := [Q(p^1_g), \ldots, Q(p^{n_g}_g)] \).

Parameter values. Hence the reduced order state \( z \) retains a consistent meaning across the parameter domain. Note that Algorithm 1 only computes the state matrices and equilibrium conditions on a grid of specified parameter values. Interpolation (e.g. linear, spline, etc.) must be used to evaluate the state matrices and equilibrium conditions at any parameter value not contained in this grid. It is assumed that the grid of parameter values is sufficiently dense that this interpolation is accurate. The choice of this grid is problem dependent.

The parameter varying linearization can be used to approximate the response of the nonlinear system for an initial condition \( x_0 \), input \( u_k \), and parameter trajectory \( \rho_k \). Specifically, the initial condition and input for the nonlinear system define a corresponding initial condition \( z_0 = Q^T(x_0 - \bar{x}(\rho_k)) \) and input \( \delta u_k = u_k - \bar{u}(\rho_k) \) for the parameter varying system. The reduced-order, parameter varying linearization (32) can be simulated to obtain the state response \( z_k \) and output \( \delta y_k \). These correspond to the state response \( x_k = Qz_k + \bar{x}(\rho_k) \) and output \( y_k = \delta y_k + \bar{y}(\rho_k) \) for the full-order nonlinear system.

The subspace construction step of Algorithm 1 (Lines 6-9) requires the SVD of the matrix \( X_0 \) that contains the snapshot data from all the operating points. This matrix has \( n_x n_g \) columns and hence the SVD of \( X_0 \) may be computationally intractable if there are many grid points. A suboptimal, but more computationally efficient approach, is to iteratively process the snapshot data from each grid point. The basic idea is to determine a set of modes \( Q(p^1_g) \) that capture the energy in the snapshots at the first grid point. Next, additional modes \( Q(p^2_g) \) are computed so that the combined set \( [Q(p^1_g), Q(p^2_g)] \) captures the energy in the snapshots at the second grid point. The procedure continues iteratively computing new modes \( Q_2 := Q(p^2_g) \) to combine with previously computed modes \( Q_1 := [Q(p^1_g), \ldots, Q(p^{n_g-1}_g)] \). The benefit is that only snapshots obtained from one grid point are required for the calculations. This iterative procedure requires a method to compute the optimal (new) modes \( Q_2 \) that should be added to some given modes \( Q_1 \). Theorem 1 in the appendix provides a POD-type result to perform this iterative calculation. Algorithm 2 gives the detailed steps for the iterative subspace construction. This method can be thought of as a hybrid POD/Gram-Schmidt approach. Also, it is assumed that the snapshots from one simulation or experiment have comparable amounts of energy, which makes this approach reasonable. This iterative method can replace the single-step method described in Lines 6-9 of Algorithm 1. The remaining steps of Algorithm 1 are unchanged even when combined with the iterative subspace calculation.

There are several benefits of the proposed reduced-order LPV models. The main benefit is that the models can be used for standard, gain-scheduled control. In addition, more formal control analysis and synthesis tools have been developed for LPV systems [46–48]. This LPV modeling approach can yield models that are accurate over a wide range of operating conditions for a nonlinear system. This is in contrast to the existing reduced-order modeling approaches described above which, for the most part, are used to construct a single linear model. In addition, the proposed method relies on input/output data from a forced response and does not require linearization of the system or the construction/simulation of the system adjoint as in BPOD. Finally, the reduced-order model involves a projection onto a well-defined reduced-order subspace. This retains a physical meaning in the reduced-order states. In other words, the reduced-order state can be used to approximate the full-order state of the system. This provides insight into the key spatial modes of fluid/structure systems and is not simply a black-box identification technique, often seen in subspace identification.
The last section in this paper emphasizes the application of this method to develop LPV models for high-dimensional systems.

4. APPLICATIONS

This section demonstrates the proposed IOROM approach on two examples. The first example uses a simple nonlinear mass-spring-damper model. The purpose of this example is to show how this method is implemented in detail as well as the feasibility of this method. The second example is a variation of the low Reynolds number actuator disk example that has been used in wind farm literature [50, 51]. This example has approximately 20,000 states and exhibits highly nonlinear behavior. The IOROM algorithm has also been successfully applied to a more realistic wind farm model using a large eddy simulation with 3.6 million states. Details on that example can be found in [52].

4.1. Nonlinear Mass-Spring-Damper Example

4.1.1. Model Formulation A single mass, shown in Figure 1, is connected to a wall by a linear damper and nonlinear spring. Two forces, \( F \) and \( u \), are applied to the block. Newton’s second law yields the following continuous-time, nonlinear model:

\[
\frac{d}{dt} \begin{bmatrix} p \\ v \end{bmatrix} = \begin{bmatrix} \frac{1}{m} (F + u - bv - k_s(p)p) \\ v \end{bmatrix}
\] (33)

where \( p \) is the position of the block [units of m], \( v \) is the velocity [m/s], \( m \) is the mass of the block [kg], \( b \) is the (linear) damping constant [N/(m/s)], and \( k_s(p) \) is a nonlinear spring constant (in N/m). The spring constant has the form \( k_s(p) = k_1 + k_2 p^2 \) and is assumed to be a stiffening spring \( k_2 > 0 \). The corresponding values for these system constants are given in Figure 1. Finally, the block has two forces: an exogenous (disturbance) force \( F \) [N] and a controllable input force \( u \) [N].

A continuous-time parameter varying linearization can be performed. Instead, the model is discretized to align with the discrete-time development used in this the paper. A simple Euler approximation with sample time \( \Delta t \) yields a two-state discrete-time model:

\[
x_{k+1} = f(x_k, u_k, F_k)
\]

where \( x := [p, v]^T \) is the state and the discretized dynamics are given by:

\[
f(x_k, u_k, F_k) := x_k + \Delta t \begin{bmatrix} \frac{1}{m} (F_k + u_k - bv_k - k_s(p_k)p_k) \\ v_k \end{bmatrix}
\] (35)

The disturbance force \( F_k \) is treated as the varying parameter in the dynamics: \( \rho_k := F_k \). If \( u \equiv 0 \) and the disturbance force is held constant at \( F \equiv \rho \) then the mass moves to an equilibrium position \( \bar{p}(\rho) \). This equilibrium position can be determined by solving the cubic equation \( \rho = k_s(\bar{p}) \bar{p} \) for \( \bar{p} \).
Thus a parameterized collection of equilibrium conditions for the mass-spring-damper system is:
\[
(\bar{x}(\rho), \bar{u}(\rho)) := \left( \left[ \begin{array}{c} \bar{p}(\rho) \\ 0 \end{array} \right], 0 \right)
\] (36)

The parameter varying linearization around this collection of equilibrium points is given by:
\[
\delta x_{k+1} = A(\rho_k)\delta x_k + B(\rho_k)\delta u_k + (\bar{x}(\rho_k) - \bar{x}(\rho_{k+1}))
\] (37)

where: 
\[
A(\rho) := I_2 + \Delta t \left[ \begin{array}{cc} 0 & \frac{1}{m} \\ -k_{s,lin}(\rho) - \frac{1}{m} & 0 \end{array} \right] \quad \text{and} \quad B(\rho) := \Delta t \left[ \begin{array}{c} 0 \\ \frac{1}{b} \end{array} \right].
\]

The parameter value. Next, extend the mass-spring-damper example to consist of \( M \) blocks. Each block has mass \( m \) and is connected to its neighboring block by a linear damper and nonlinear spring. In addition, each block is connected to “ground” (fixed wall) by a linear damper and nonlinear spring. All dampers have constant \( b \) and all springs are given by \( k_s(p) = k_1 + k_2p^2 \). Finally, the \( M^{th} \) block has an exogenous (disturbance) force \( F \) and a controllable input force \( u \).

Let \( (p_i, v_i) \) denote the position and velocity of block \( i \). The continuous-time dynamics for this \( M \)-block mass-spring-damper system are described by:
\[
\begin{align*}
 m\ddot{v}_i &= -F_i - F_{i,i-1} - F_{i,i+1} + \sum_{j \in N(i)} F_{i,j} \\
 m\ddot{v}_M &= F + u - F_M - F_{M,M-1}
\end{align*}
\] (38)

where \( F_i \) denotes the force due to a spring and damper connecting block \( i \) to the fixed wall. \( F_{i,j} \) denotes the force due to a spring and damper connecting blocks \( i \) and \( j \). These forces are defined as \( F_i := b(v_i - \bar{v}_i) + k_s(p_i)p_i \) and \( F_{i,j} := b(v_i - v_j) + k_s(p_i - p_j)(p_i - p_j) \). The convention for the first block is \( F_{1,0} \equiv 0 \). The position of the \( M^{th} \) block is the output for the system, i.e. \( y = p_M \). All mass, spring, and damping constants are the same as those given in Figure 1. The result is a nonlinear system with state \( x := [p^T, v^T]^T \in \mathbb{R}^{2M} \), a single controllable input \( u \) and single output \( y \). The exogenous force \( F \) is again considered as a parameter in this model (\( \rho := F \)). All results are given for \( M = 100 \) blocks. This example is small enough that the reduced and full order models can be directly compared to assess the feasibility of the method.

4.1.2. IOROM Method: Single Operating Point The IOROM method described in Section 3.1 is used to construct a single time-invariant model with the exogenous force held constant at \( \rho = 1 \text{N} \). The system is simulated with \( \rho = 1 \) and \( u = 0 \) to determine the equilibrium point \( \bar{x}(\rho) \) for this parameter value. Next, the nonlinear system is excited in continuous-time for \( T_f = 20 \text{s} \) with a chirp excitation input \( u(t) = 0.1 \sin(\omega(t)t) \). The chirp frequency is \( \omega(t) = 0.1 \times (20)^{t/T_f} \) which excites the nonlinear system with frequencies from 0.1 to 2 \( \text{rad/s} \). The nonlinear system is simulated and snapshots of the state/output are collected every \( \Delta t = 0.1 \text{s} \). This yields 200 snapshots over the \( T_f = 20 \text{s} \) simulation.

The left subplot of Figure 2 shows the POD and least-squares fitting errors obtained by the IOROM procedure. The horizontal axis is the number of POD modes selected from the snapshots \( X_0 \). The POD error is the relative energy lost in \( X_0 \) by selecting the specified number of POD modes. The model relative fitting error is obtained by evaluating the IOROM least-squares cost with the optimal fit and dividing by \( \| X_1 \|_F \). Both relative errors decrease rapidly with increasing number of POD modes. The fitting error levels off around 4 modes and hence this was selected for the model fit. The relative fit error does not go to zero because the discrete-time linear system cannot perfectly fit the snapshots from the continuous-time nonlinear system. The right subplot shows a Bode plot for the four-state, reduced-order, discrete time model. The plot also shows a 200-state, full-order linearization. The reduced-order model agrees quite closely with the full order model in the frequency band of input excitation. The mismatch between the two models at high frequencies can be reduced by reducing the snapshot sampling time \( \Delta t \) and increasing the chirp excitation frequency.

It is important to note that the quality of the reduced-order model degrades if too many POD modes (i.e. much more than 4) are selected for the fit. This is due to overfitting a linear model to
a nonlinear response. Inverse modeling techniques may be used to select the number of modes to accurately represent the system without overfitting [53]. This will be addressed in future work. The POD and model fit errors shown in Figure 2 provide useful criteria to avoid such overfitting.

4.1.3. IOROM Parameter Varying Method

Next, the parameter varying IOROM method described in Section 3.3 is used to construct parameter varying reduced-order models. A grid of eleven parameter values was selected for the exogenous forces as \( \{0, 0.2, \ldots, 2\} \). At each fixed value of \( \rho = p_j \) in this grid \( (j = 1, \ldots, 11) \) the system was simulated with \( u = 0 \) to determine the corresponding equilibrium point \( \bar{x}(p_j) \). Next, the nonlinear system is excited for \( T_f = 20 \) s at each \( p_j \) with a chirp excitation input \( u(t) = 0.1 \sin(\omega(t)t) \). The exogenous force \( F = \rho \) stretches the springs and this increases the system speed of response due to the spring stiffening. Thus the chirp frequency was chosen to depend on the parameter value as \( \omega(t) = 0.1 (1 + 10 \rho)^{t/T_f} \). This excites the nonlinear system with frequencies from 0.1 to \((1 + \rho)\) rad/sec. The nonlinear system is simulated and snapshots of the state/output are collected every \( \Delta t = 0.1 \) s. This yields 200 snapshots over the \( T_f = 20 \) s simulation for each parameter grid point.

The basic single-step procedure (Algorithm 1) is used to construct the subspace modes \( Q \). This example is sufficiently small such that it was possible to compute the SVD on snapshots obtained at all grid points. Five modes were selected and reduced-order models were constructed at each grid point using the IOROM procedure. The model relative fitting error was less than 0.05 at each grid point. This indicates that the reduced order models at each “fixed” grid point accurately match the recorded snapshot data. The iterative method (Algorithm 2) was also implemented. This also yielded a five-state subspace \( Q \) although with slightly larger fitting errors. The single-step and iterative approach gave similar results and hence the remainder of the example focuses on the single-step algorithm.

The exogenous disturbance varies as \( F(t) = \rho(t) = -\cos(0.5t) + 1 \). The controllable input is a step \( u(t) = 0 \) for \( t < 25\)sec and \( u(t) = 0.1 \) for \( t \geq 25\)sec. Linear interpolation was used to compute the state matrices and equilibrium point appearing in the LPV model for parameter values not contained in the eleven point grid. The left subplot of Figure 3 shows the position of the \( M^{th} \) block. This is obtained from the parameter varying model as \( y_k = \bar{y}(p_k) + \delta y_k \). The response of the reduced-order model is in close agreement with the nonlinear model. The effect of the step input \( u \) is apparent after \( t = 25\)sec. Again, the states of the reduced model \( z_k \) can be “lifted” to estimate the response of the full order system using the chosen POD modes as \( x_k = \bar{x}(p_k) + \delta x_k \approx \bar{x}(p_k) + Qz_k \). The right subplot of Figure 3 compares a subset of these state estimates (position of blocks 98-100) for the reduced-order system with the true response of the nonlinear system. It should be noted that the output is the last block of the system, i.e. block 100 (bottom plot in Figure 3). Again, there is close agreement.
4.2. Wind Farm Example: Actuator Disk

4.2.1. Model Formulation
The second, more complex, example considered in this paper is the actuator disk model [50, 51]. The actuator disk is used to represent wind turbines in the flow field. In this specific example, we will use two turbines, one directly downstream of the other, see Figure 4. The flow within the two turbine array is modeled using the two-dimensional unsteady, incompressible, Navier-Stokes equations. The typical operating wind speeds in a wind farm do not exceed 25 m/s. This is low relative to the speed of sound at sea level (∼300 m/s) and hence it is sufficient to assume incompressibility [54]. Let \((U, V)\) [m/s] denote the streamwise and spanwise velocity components and \((X, Y)\) [m] denote the streamwise and spanwise distance. Under these assumptions, the nondimensionalized dynamics for \((U, V)\) are governed by the following partial differential equations:

\[
\frac{\partial U}{\partial t} + U \frac{\partial U}{\partial X} + V \frac{\partial U}{\partial Y} = -\frac{\partial P}{\partial X} + \frac{1}{Re} \left( \frac{\partial^2 U}{\partial X^2} + \frac{\partial^2 U}{\partial Y^2} \right) + f_{X,1} + f_{X,2} \tag{40}
\]

\[
\frac{\partial V}{\partial t} + U \frac{\partial V}{\partial X} + V \frac{\partial V}{\partial Y} = -\frac{\partial P}{\partial Y} + \frac{1}{Re} \left( \frac{\partial^2 V}{\partial X^2} + \frac{\partial^2 V}{\partial Y^2} \right) \tag{41}
\]

where \(P\) is pressure in the flow, \(Re\) is the Reynolds number, and \(f_X\) is a force turbines apply to the flow in the \(X\) direction. \(Re\) is defined as the ratio of inertial forces to viscous forces: \(\frac{U_{\infty}D}{\nu}\) where \(U_{\infty}\) is the freestream velocity [m/s]. By changing \(Re\) in this example, it is effectively changing the freestream velocity. The dynamics of the system vary depending on the parameter, \(Re\), that is being simulated.

The loading of each turbine is defined linearly. Specifically, assume that all spatial units have been nondimensionalized by the turbine diameter \(D\). If the hub of the upstream turbine is placed at \(X = X_1\) and \(Y = Y_1\) then the rotor plane lies within \(Y_1 - \frac{1}{2} \leq Y \leq Y_1 + \frac{1}{2}\). The forcing term introduced by the turbines is then given by:
where \( i = 1, 2 \) indicates the upstream and downstream turbines. The constant \( k_T := U_{in,i}, U_{in} \) is the average nondimensionalized velocity across the rotor, and \( C_{T,i} \) is the thrust coefficient of the turbine, which is a function of the turbine operation, i.e. the axial induction factor, \( a \). According to actuator disk theory, \( C_T \) can be related to the axial induction factor by: \( C_T = 4a(1-a) \) [55]. Physically, the axial induction factor is a measure of how much the wind slows down due to the action of the turbine. The optimal operating point is \( a = \frac{1}{3} \). In this particular example, \( a_2 \) is the single input to this system, i.e. \( a_1 \) is fixed at the optimal operating point and \( a_2 \) is allowed to vary about the optimal operating point. This linear profile is smallest at the rotor hub and grows linearly at the blade tips. The loading magnitude can be changed by the thrust coefficient. On utility-scale turbines, this would be equivalent to changing the blade pitch angle or generator torque [1, 55]. These equations are solved using standard CFD methods [56]. Specifically, a central differencing scheme was used for the two-dimensional actuator disk model. The grid is defined by \( N_X \) points in the streamwise \( X \) direction and \( N_Y \) point in the spanwise \( Y \) direction. For this actuator disk model, the typical spacing between grid points is \( \delta X = 0.01 \) and \( \delta Y = 0.01 \) with a typical time step of \( \delta t = 0.01 \).

![Figure 4](image-url) Two-turbine setup (left) and the corresponding baseflow (right) for the low Re flow specified. It should be noted that the flow is depicted as if you are looking at the two-turbine array from above.

The setup has two turbines spaced 5\( D \) apart in the streamwise direction. There is a point 3\( D \) downstream of the second turbine where the wind speed is measured (indicated by the white triangle in Figure 4). For simplicity, as mentioned above, we assume that we can only control the wind speed by actuating the second turbine. As a result, the system that will be identified will be a single-input-single-output system where the input is the axial induction factor of the downstream turbine, \( a_2 \), and the output is the spanwise velocity, \( V \) at the white triangle in Figure 4. In addition, the domain in the streamwise direction was set to 20\( D \) with 201 grid points and in the spanwise direction was set to 5\( D \) with 51 grid points. This would amount to 20,000 states for this particular example by having 2 velocity components per grid point. More realistic, higher fidelity codes for wind farms will have even larger state dimensions [2, 57]. It is important to note that the \( Re \) used in these simulations will be on the order of 10. This is a low \( Re \) in comparison with the \( Re \) typically experienced by wind farms, which is on the order of \( 10^6 \).

4.2.2. IOROM Method: Single Operating Point First, the IOROM method discussed in Section 3.1 will be implemented for the actuator disk problem at one particular operating point, i.e. a single \( Re \). We will use Algorithm 1, but as examples become larger than this, i.e. large eddy simulations where the states are on the order of millions rather than tens of thousands, Algorithm 2 will have to be implemented. One way to do is to use the technique used for large streaming data sets in [58].

Similar to the mass-spring-damper example, the IOROM method is used to construct a single time-invariant model with \( Re \) held constant. The system is simulated with \( Re = 50 \) and \( u = 0 \), i.e. the downstream turbine input is fixed at the optimal operating point, to determine the equilibrium point \( \bar{x}(Re) \) for this parameter value. Next, the nonlinear system (39) is excited for \( T_f = 100 \) s (recall the timestep for this simulation is 0.01 s). This system is excited at the second turbine with
a chirp excitation input $u(t) = 5 \sin(\omega(t)t)$, where $\omega(t) = 0.0628(50)^{0.5} \text{[rad/s]}$. This excites the nonlinear system with frequencies from 0.01 to 0.2 Hz. The nonlinear system is simulated and snapshots of the state/input/output are collected every 0.5 s. This yields 200 snapshots over the 100 s simulation.

Using the IOROM approach, we were able to construct a model with 20 states. This was selected by examining the model fitting error. Because the state dimension is large, the model fitting error cannot be computed directly. Instead, (16) was used to compute the model fitting error. Similar to the mass-spring-damper example, the model error decreases as more modes are used but begins to plateau at 20 modes. At this point, there is minimal benefits to increasing the order of the model. As stated previously, it is possible that the performance of the model will degrade as the model order increases because the model will be overfitting to the nonlinearities of the system.

Figure 5. IOROM Results of the actuator disk example. The top plot shows the flow field reconstructed using 20 modes. The middle plot shows the full-order nonlinear flow. The bottom plot shows the output (indicated by the triangle in the top and middle plot) of the reduced-order model compared to the full-order model. The output for this particular example is the velocity fluctuations at the measurement point 3D downstream of the second turbine.

The reduced-order model was validated using a square wave, $u = 5 \text{square}(0.5t)$, as an input to the second turbine. The results can be seen in Figure 5. One of the main advantages to this method is that the full state can be reconstructed from the reduced-order state. Figure 5 shows the results of the reduced-order model as compared to the full-order model. The top plot shows the reconstructed flow field from the 20 state model identified from the IOROM approach. The middle plot shows the full-order actuator disk model. Visually, the top plot and the middle plot show good agreement. Quantitatively, the error between the full-order and reduced-order model is on the order of $10^{-1}$. The bottom plot shows the output; i.e. the velocity measured 3D downstream of the second turbine. The output of the reduced-order model closely matches the output of the full-order model.

The IOROM technique that was used to identify a low-order model for the actuator disk example was compared to another existing technique, ERA (discussed in Section 2.2.2), that is capable of constructing an input/output model. This IOROM technique has also been compared to BPOD in [59]. A separate simulation was run to obtain the impulse response data necessary for ERA using an impulse as an input. Figure 6 shows the results of the input/output relationship obtained using ERA and the IOROM approach on the actuator disk example. In Figure 6, the input to the system is shown on the top, where $u(t) = 5 \sin(0.5t)$. The output of the reduced-order models are shown on
Figure 6. This plot shows the output of the reduced-order models constructed using ERA and the IOROM approach proposed in this paper. The output for this particular example is the velocity fluctuations at the measurement point $3D$ downstream of the second turbine.

the bottom plot. This bottom plot shows that ERA has a difficult time identifying the system at this particular operating point. This is in part due to the excitation signal. With the IOROM technique, a chirp signal was used, which excites the dynamics at a range of frequencies. With ERA, an impulse is used and is unable to capture the essential dynamics of the system. This IOROM technique has a much larger excitation energy than the impulse used by ERA. This allows the IOROM to capture a better input/output model. However, to use this IOROM approach, the system must be of the form where it is easily excitable by a controllable input.

4.2.3. IOROM Parameter Varying Method
The parameter varying method described in Section 3.3 is used to construct parameter varying reduced-order models. A grid of five parameter values was selected, $Re_j = \{10 : 10 : 50\}$. At each fixed value of $Re$ in this grid, the system was simulated with $u = 0$ to determine the corresponding equilibrium point, $\bar{x}(Re_j)$. Next, the nonlinear system is excited for $T_f = 100$ s at each $Re_j$ with a chirp excitation input $u(t) = 5 \sin(\omega(t)t)$, where $\omega(t) = 0.0628(50)^{T_f}$. The $Re$ changes the effective wind speed of the simulation which changes the dynamics within the system. For example, wake meandering is an oscillation that happens in this particular flow due to the instabilities in the wake. Wake meandering instabilities increase, i.e. the magnitude increases, as the $Re$ increases in this particular example. The chirp signal frequencies were chosen to capture the wake meandering frequency. The chirp signal excites the nonlinear system at frequencies between 0.01 Hz to 0.5 Hz. Again, the nonlinear system is simulated and snapshots of the state/input/output are collected every 0.5 s. This yields 200 snapshots over the $T_f = 100$ s simulation for each parameter grid point.

The basic single step procedure (Algorithm 1) is used to construct the subspace modes $Q$. This example is small enough that it was possible to compute the SVD on snapshots obtained at all grid points. In addition, a MapReduce technique in [60] can be used to compute the SVD of large, tall-skinny matrices. Twenty modes were selected and reduced-order models were constructed at each grid point using the IOROM procedure.

Lastly, Figure 7 compares a time-domain step response of the full-order nonlinear system and the reduced-order LPV model. The $Re$ varies at $Re(t) = 25 - 10 \sin(0.1t)$. The controllable input was set to $u(t) = 5 \text{square}(0.5t)$. The nonlinear system was simulated with a time step of 0.01 s and the reduced-order parameter varying model was simulated with a time step of 0.5 s. Linear interpolation was used to compute the state matrices and equilibrium points appearing in the LPV model for parameter values not contained in the five point grid. It should be emphasized that a different input signal was used for the validation of this LPV model and the simulation is varying.
across a wide range of $Re$. The input/output relationship of the reduced-order LPV model is in close agreement with the full-order nonlinear system. These results indicate that this reduced-order LPV model can be used for predictive purposes and/or control design.

![Graphs showing time domain signal of the Reynolds number, input, and output. The output of the reduced-order LPV model is compared to the full-order nonlinear simulation. Note that the input is the turbine forcing on the downstream turbine and the output is the velocity fluctuations 3D downstream of the second turbine.]

Figure 7. Time domain signal of the Reynolds number, input, and output. The output of the reduced-order LPV model is compared to the full-order nonlinear simulation. Note that the input is the turbine forcing on the downstream turbine and the output is the velocity fluctuations 3D downstream of the second turbine.

5. CONCLUSION

This paper described a method to construct reduced-order models for high-dimensional nonlinear systems. It is assumed that the nonlinear system has a collection of equilibrium operating points. The method has two main components. First, a reduced-order linear system is constructed at each equilibrium point using input/output data. Second, a parameter varying linearization is used to connect these linear models. A nonlinear mass-spring-damper example as well as an actuator disk example were used to demonstrate this method. Future work will apply this method to construct reduced-order parameter varying models for wind farm control using higher-fidelity models.

6. ACKNOWLEDGMENTS

This work was partially supported by the National Science Foundation under Grant No. NSF-CMMI-1254129 entitled “CAREER: Probabilistic Tools for High Reliability Monitoring and Control of Wind Farms”. The work was also supported by the University of Minnesota Institute on the Environment, IREE Grant No. RS-0039-09.

The first author gratefully acknowledges the financial support from the University of Minnesota through the 2015-16 Doctoral Dissertation Fellowship.

REFERENCES
Theorem 1
Let \( Q_1 \in \mathbb{R}^{n_x \times r_1} \) be a given matrix with \( Q_1^TQ_1 = I_{r_1} \). Let \( X \in \mathbb{R}^{n_x \times n_x} \) be given snapshot data. Define the SVD of the projected snapshot matrix (43) as \( U \sum V^T \). Let \( r_2 \) be any non-negative integer such that \( r_2 \leq \text{rank}(X) \) and \( \sigma_{r_2} > \sigma_{r_2+1} \). Then

\[
\min_{Q_2 \in \mathbb{R}^{n_x \times r_2}, Q_1^TQ_2 = I_{r_2}} \| X - Q_1C_1 - Q_2C_2 \|_F^2 = \sum_{k=r_2+1}^{\text{rank}(X)} \sigma_k^2
\]

(43)

An optimal solution is given by \( C_{1,\text{opt}} = Q_1^TX \), \( Q_{2,\text{opt}} = U_r \), and \( Q_{r,\text{opt}} = \Sigma_r V_r^T \) where \( \Sigma_r, U_r, \) and \( V_r \) are associated with the first \( r \) singular values and vectors of \( (I_{n_x} - Q_1Q_1^T)X \).

Proof
Use Gram-Schmidt orthogonalization to construct a matrix \( Q_{1,\perp} \in \mathbb{R}^{n_x \times (n_x - r_1)} \) such that \( [Q_1 \ Q_{1,\perp}] \) is orthogonal. The orthogonal invariance of the Frobenius norm thus implies

\[
\| X - Q_1C_1 - Q_2C_2 \|_F^2 = \left\| \begin{bmatrix} Q_1^T \\ Q_{1,\perp}^T \end{bmatrix} (X - Q_1C_1 - Q_2C_2) \right\|_F^2 = \left\| \begin{bmatrix} Q_1^T X - C_1 Q_1^T Q_2 C_2 \\ Q_{1,\perp}^T X - Q_{1,\perp}^T Q_2 C_2 \end{bmatrix} \right\|_F^2
\]

(44)

The second equality follows from \( Q_1^T Q_1 = I_{r_1} \) and \( Q_{1,\perp}^T Q_1 = 0_{(n_x - r_1) \times r_1} \). The error can be split as:

\[
\| X - Q_1C_1 - Q_2C_2 \|_F^2 = \| Q_1^T X - C_1 Q_1^T Q_2 C_2 \|_F^2 + \| Q_{1,\perp}^T X - Q_{1,\perp}^T Q_2 C_2 \|_F^2
\]

(45)
The second term is unaffected by the choice of \( C_1 \). Moreover, for any \((Q_2, C_2)\) the first term can be made equal to zero by the choice \( C_{1, \text{opt}} = Q_1^T X - Q_1^T Q_2 C_2 \). In fact, \( Q_1^T Q_2 = 0 \) may be assumed without loss of generality. Specifically, the choice of \( Q_2 \) only affects the second term of (45) (assuming the optimal choice for \( C_1 \) just specified). Perform a change of variables \( Q_2 = [Q_1 \quad Q_{1,\perp}] \begin{bmatrix} \hat{Q}_2 \\ \tilde{Q}_2 \end{bmatrix} \). This change of variables from \( Q_2 \) to \( [\hat{Q}_2 \quad \tilde{Q}_2] \) is invertible since \( [Q_1 \quad Q_{1,\perp}] \) is orthogonal. Substitute this change of variables into the second error term to obtain \( \|Q_1^T X - Q_1^T Q_2 C_2\|_F^2 = \|Q_1^T X - \hat{Q}_2 C_2\|_F^2 \). Thus \( \hat{Q}_2 \) has no effect on the second term and may be set to zero. \( Q_2 \) can be selected to have the form \( Q_{1,\perp} \hat{Q}_2 \) and hence \( Q_1^T Q_2 = 0 \). In this case, the optimal choice for \( C_1 \) simplifies to \( C_{1, \text{opt}} = Q_1^T X \).

Retaining the assumption that \( Q_1^T Q_2 = 0 \) as well as \( C_{1, \text{opt}} = Q_1^T X \), the total error is given by:

\[
\|X - Q_1 C_1 - Q_2 C_2\|_F^2 = \|(I_{nx} - Q_1 Q_1^T)X - Q_2 C_2\|_F^2
\]

(46)

By the standard POD result this cost is minimized by the choice \( Q_{2, \text{opt}} = U_r \) and \( C_{2, \text{opt}} = \Sigma_r V_r^T \) where \( \Sigma_r \), \( U_r \), and \( V_r \) are associated with the first \( r \) singular values and vectors of \((I_{nx} - Q_1 Q_1^T)X\). It can be shown that this construction satisfies the assumption \( Q_1^T Q_2 = 0 \). □