

MIT Open Access Articles

Data-driven operator inference for nonintrusive projection-based model reduction

The MIT Faculty has made this article openly available. **Please share** how this access benefits you. Your story matters.

Citation: Peherstorfer, Benjamin, and Karen Willcox. "Data-Driven Operator Inference for Nonintrusive Projection-Based Model Reduction." *Computer Methods in Applied Mechanics and Engineering* 306 (July 2016): 196–215 © 2016 Elsevier B.V.

As Published: <http://dx.doi.org/10.1016/J.CMA.2016.03.025>

Publisher: Elsevier BV

Persistent URL: <http://hdl.handle.net/1721.1/117750>

Version: Author's final manuscript: final author's manuscript post peer review, without publisher's formatting or copy editing

Terms of use: Creative Commons Attribution-NonCommercial-NoDerivs License



Data-driven operator inference for nonintrusive projection-based model reduction

Benjamin Peherstorfer^{a,*}, Karen Willcox^a

^a*Department of Aeronautics & Astronautics, MIT, 77 Massachusetts Avenue, Cambridge, MA 02139, USA*

Abstract

This work presents a nonintrusive projection-based model reduction approach for full models based on time-dependent partial differential equations. Projection-based model reduction constructs the operators of a reduced model by projecting the equations of the full model onto a reduced space. Traditionally, this projection is intrusive, which means that the full-model operators are required either explicitly in an assembled form or implicitly through a routine that returns the action of the operators on a given vector; however, in many situations the full model is given as a black box that computes trajectories of the full-model states and outputs for given initial conditions and inputs, but does not provide the full-model operators. Our nonintrusive operator inference approach infers approximations of the reduced operators from the initial conditions, inputs, trajectories of the states, and outputs of the full model, without requiring the full-model operators. Our operator inference is applicable to full models that are linear in the state or have a low-order polynomial nonlinear term. The inferred operators are the solution of a least-squares problem and converge, with sufficient state trajectory data, in the Frobenius norm to the reduced operators that would be obtained via an intrusive projection of the full-model operators. Our numerical results demonstrate operator inference on a linear climate model and on a tubular reactor model with a polynomial nonlinear term of third order.

Keywords: nonintrusive model reduction, data-driven model reduction, black-box full model, inference

1. Introduction

Model reduction seeks to construct reduced models that provide accurate approximations of the full model solutions with orders of magnitude reduction in computational complexity. We consider here projection-based model reduction for full models that are based on parametrized time-dependent partial differential equations (PDEs). Projection-based model reduction first constructs a basis of a low-dimensional reduced space and then projects the equations of the full model onto the reduced space to obtain the operators of the reduced model [1, 2, 3, 4, 5]. The construction of these reduced operators is usually intrusive and requires the full-model operators, which means that the full-model operators need to be available either in an assembled form or through a routine that returns the action of the operators on a given vector. This limits the scope of projection-based model reduction because in many situations the full model is given as a black box, which allows computing the trajectories of the full-model states and outputs for given initial conditions and inputs, but does not provide the full-model operators. Note that we also consider full models as black-box models if the source code of the implementation is available but the complexity of the code renders obtaining the full-model operators time consuming, see the general circulation model (GCM) [6] in Section 4.4. We present nonintrusive operator inference that replaces the classical intrusive reduced operator construction by deriving approximations of the reduced operators directly from data of the full model, without requiring the full-model operators. The data include initial conditions, inputs, trajectories of the full-model states, and outputs. The reduced operators are low-dimensional quantities and therefore the inference underlying

*Corresponding author

Email addresses: pehersto@mit.edu (Benjamin Peherstorfer), +1-617-253-7831 (Benjamin Peherstorfer)

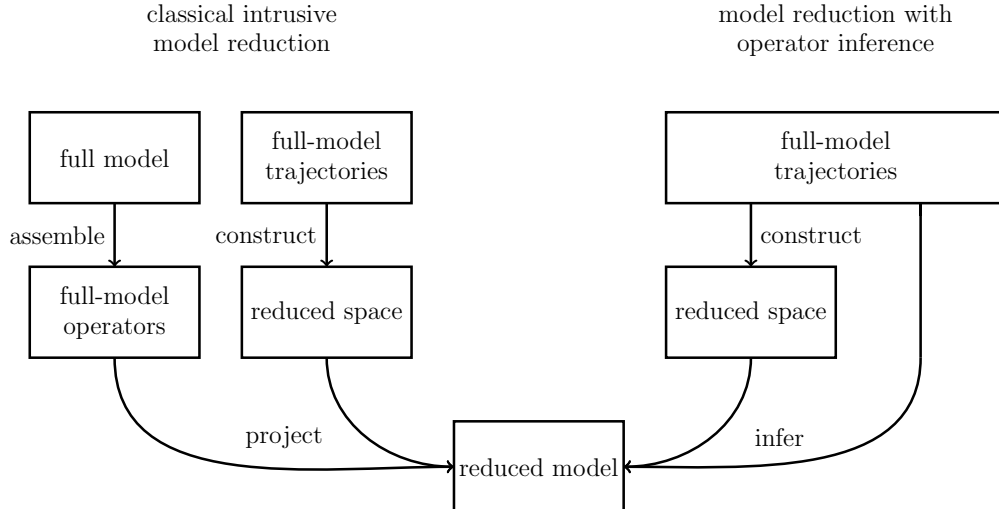


Figure 1: The projection step in classical projection-based model reduction is intrusive because the operators of the full model are required for the construction of the reduced operators. In contrast, our operator inference derives the approximations of the reduced operators directly from data (initial conditions, inputs, trajectories, outputs) of the full model. This operator inference is nonintrusive and therefore applicable to black-box full models.

our approach is feasible with respect to computational costs and with respect to required amount of data. Our operator inference is applicable to nonlinear PDEs with polynomial nonlinear terms of low order. Our operator inference provides a nonintrusive way to construct a reduced model, see Figure 1.

There are several classical projection-based model reduction methods. Proper orthogonal decomposition (POD) constructs a reduced basis from states of the full model [1, 2, 7]. The reduced basis method [3, 8] derives a reduced basis with a greedy approach [9] based on computationally cheap *a posteriori* error estimators of intermediate reduced models. There are also Krylov subspace methods, including multivariate Padé approximations and tangential interpolation [4, 5, 10]. Once the reduced basis is generated, all these methods construct a reduced model and the reduced operators with an intrusive projection step, leading to intrusive model reduction methods. In [11], the coefficients of a reduced state for a given input are approximately derived from a computationally cheap coarse-grid full-model state. This avoids the construction of reduced operators and leads to a nonintrusive approximate reduced basis method; however, the computation of the coefficients requires solving the equations of the coarse-grid full model at each input for which the reduced system of the reduced model is solved.

The Loewner framework provides a nonintrusive approach to construct reduced models of linear time-invariant (LTI) systems. The reduced operators are derived directly from frequency response data, i.e., from transfer function evaluations [12, 13, 14], without requiring the operators of the full LTI system. The Loewner framework has been extended to parametric LTI systems [15] and to LTI systems with multiple outputs [16]; however, compared to intrusive projection-based model reduction, the scope of the Loewner framework is still limited. In particular, an extension of the Loewner framework to nonlinear models remains an open problem. There is also vector fitting [17, 18] that constructs approximate rational interpolants of LTI systems for given frequency response data. Both the Loewner framework and vector fitting rely on frequency response data, which are often unavailable for full models that are formulated in the time domain and marched forward in time with a time stepping scheme.

In contrast to projection-based reduced models, data-fit surrogate models are constructed using interpolation and regression techniques to directly learn the map from inputs to outputs of the full model [19, 20, 21, 22]. The construction of a reduced model and its operators is therefore circumvented and without the intrusive projection step. The approaches in [23, 24, 25, 26, 27, 28, 29, 30] construct a reduced space and learn a surrogate model that maps inputs to coefficients of the representations of full-model states in

the reduced space. Typically, data-fit surrogate models target different applications than projection-based reduced models, e.g., where the notion of state is unimportant and where the interpolation and regression of the outputs provide sufficiently accurate results, see [31]. We do not consider data-fit surrogate models here.

Dynamic mode decomposition (DMD) is a method to analyze the behavior of dynamical systems [32, 33, 34, 35, 36]. DMD derives an approximation of the operator of the full model (dynamical system) directly from full-model trajectories. The obtained approximate operator corresponds to the linear approximation of the full model that best fits the trajectories in the L^2 norm. The approximate operator is then used to approximate the eigenvectors and eigenvalues of the (unknown) system matrix of the full model [32]. One then uses these approximate eigenvectors and eigenvalues to analyze the behavior of the full model. Our operator inference follows a similar approach to DMD. We also optimize for operators that best fit the trajectories in the L^2 norm. In the case of a linear full model and inputs that are constant over time, our inferred operator is the same operator as the one obtained via DMD [35, Algorithm 1]; however, DMD derives a linear approximation of the full model, whereas our approach can handle full models that have polynomial nonlinear terms. Note that a first approach exists to extend DMD to nonlinear full models via the Koopman operator that transforms a finite-dimensional nonlinear system into an infinite-dimensional linear system [37], see also [38, 39]. In contrast to DMD, our approach explicitly takes into account the output of the full model and derives an approximate output operator that describes the map from states to outputs. Furthermore, we consider full models with operators that dependent on parameters. This means that we compute multiple inferred operators and then interpolate between them, whereas DMD is typically applied to full models with parameter-independent operators only [35].

The aim of our operator inference approach is the construction of a projection-based reduced model in a nonintrusive way. We first construct a reduced space from trajectories of the full-model states as in classical projection-based model reduction and then replace the intrusive construction of the reduced operators by our operator inference. The operator inference is based on a least-squares problem that infers the operators from initial conditions, inputs, trajectories, and outputs of the full model. These data are typically available when marching forward a black-box full model in time, in contrast to frequency response data. For sufficient and accurate data, the inferred operators converge in the Frobenius norm to the reduced operators that would be obtained via an intrusive construction. We rely on a similar least-squares problem as [40], where updates to reduced operators are computed without recourse to the full model; however, our approach infers directly the reduced operators, whereas [40] provides only updates to operators of an initial intrusive reduced model. We develop our operator inference for full models with polynomial nonlinear terms in the state, and therefore the scope of our approach goes beyond LTI systems. We note, however, that our approach is limited in practice to polynomial nonlinear terms of low order because the computational costs of the operator inference grow exponentially in the order of the polynomial nonlinear term, see Section 3.5.

The following Section 2 introduces systems of parametrized equations and describes classical intrusive projection-based model reduction based on POD. Section 3 introduces model reduction with operator inference. Section 4 presents numerical experiments, where reduced models are inferred from data of linear and nonlinear models. In particular, we infer a reduced model of a nonlinear tubular reactor model [41] and of a climate model [6]. Section 5 concludes the discussion.

2. Problem setup

Section 2.1 and Section 2.2 introduce systems of parametrized nonlinear ordinary differential equations (ODEs). Of particular interest are high-dimensional systems of ODEs that arise from spatial discretizations of time-dependent PDEs. Section 2.3 briefly discusses intrusive projection-based model reduction for systems of ODEs.

2.1. Systems of parametrized nonlinear ODEs

Consider a parameter $\boldsymbol{\mu} \in \mathcal{D}$ in the domain $\mathcal{D} \subset \mathbb{R}^d$ of dimension $d \in \mathbb{N}$ and consider time $t \in [0, T] \subset \mathbb{R}$ with $0 < T$. Let $\mathbf{u} : [0, T] \times \mathcal{D} \rightarrow \mathbb{R}^p$ be the input with $p \in \mathbb{N}$, $\mathbf{x} : [0, T] \times \mathcal{D} \rightarrow \mathbb{R}^N$ the state vector

with $N \in \mathbb{N}$, and $\mathbf{y} : [0, T] \times \mathcal{D} \rightarrow \mathbb{R}^q$ the output with $q \in \mathbb{N}$. Define with the state vector $\mathbf{x}(t; \boldsymbol{\mu}) = [x_1(t; \boldsymbol{\mu}), \dots, x_N(t; \boldsymbol{\mu})]^T \in \mathbb{R}^N$ the vector

$$\mathbf{x}(t; \boldsymbol{\mu})^2 = \begin{bmatrix} \mathbf{x}^{(1)}(t; \boldsymbol{\mu}) \\ \vdots \\ \mathbf{x}^{(N)}(t; \boldsymbol{\mu}) \end{bmatrix} \in \mathbb{R}^S, \quad (1)$$

where

$$\mathbf{x}^{(i)}(t; \boldsymbol{\mu}) = x_i(t; \boldsymbol{\mu}) \begin{bmatrix} x_1(t; \boldsymbol{\mu}) \\ \vdots \\ x_i(t; \boldsymbol{\mu}) \end{bmatrix} \in \mathbb{R}^i, \quad (2)$$

for $i = 1, \dots, N$. The vector $\mathbf{x}(t; \boldsymbol{\mu})^2$ contains as components all products of two components of the state vector $\mathbf{x}(t; \boldsymbol{\mu})$ and thus $S = \binom{N+1}{2} = N(N+1)/2 \in \mathbb{N}$. Note that the definition of $\mathbf{x}(t; \boldsymbol{\mu})^2$ in (1) accounts for the commutativity of multiplication, i.e., there is only one entry in $\mathbf{x}(t; \boldsymbol{\mu})^2$ for the product $x_1(t; \boldsymbol{\mu})x_2(t; \boldsymbol{\mu})$ and not a second entry with $x_2(t; \boldsymbol{\mu})x_1(t; \boldsymbol{\mu})$. Analogously, for a higher order $2 < \alpha \in \mathbb{N}$, the vector $\mathbf{x}(t; \boldsymbol{\mu})^\alpha \in \mathbb{R}^S$ can be constructed, where $S = \binom{N+\alpha-1}{\alpha} \in \mathcal{O}(N^\alpha)$. Note that S grows exponentially in the order α . Note further that the components of $\mathbf{x}(t; \boldsymbol{\mu})^\alpha$ correspond to the multisets of size α with N symbols. This means that computing the components of $\mathbf{x}(t; \boldsymbol{\mu})^\alpha$ is a standard combinatorics problem, see, e.g., [42].

The system of parametrized ODEs with a nonlinear term of second order in the state \mathbf{x} is

$$\begin{aligned} \frac{d}{dt} \mathbf{x}(t; \boldsymbol{\mu}) &= \mathbf{A}(\boldsymbol{\mu})\mathbf{x}(t; \boldsymbol{\mu}) + \mathbf{F}(\boldsymbol{\mu})\mathbf{x}(t; \boldsymbol{\mu})^2 + \mathbf{B}(\boldsymbol{\mu})\mathbf{u}(t; \boldsymbol{\mu}), \\ \mathbf{y}(t; \boldsymbol{\mu}) &= \mathbf{C}(\boldsymbol{\mu})\mathbf{x}(t; \boldsymbol{\mu}). \end{aligned} \quad (3)$$

The operator $\mathbf{A}(\boldsymbol{\mu}) \in \mathbb{R}^{N \times N}$ corresponds to the linear terms of the system of ODEs and the matrix $\mathbf{F}(\boldsymbol{\mu}) \in \mathbb{R}^{N \times S}$ to the nonlinear term. The matrices $\mathbf{B}(\boldsymbol{\mu}) \in \mathbb{R}^{N \times p}$ and $\mathbf{C}(\boldsymbol{\mu}) \in \mathbb{R}^{q \times N}$ are the input and output matrices, respectively. We further have the initial condition $\mathbf{x}(0; \boldsymbol{\mu}) = \mathbf{x}_0(\boldsymbol{\mu}) \in \mathbb{R}^N$. The operators $\mathbf{A}(\boldsymbol{\mu}), \mathbf{B}(\boldsymbol{\mu}), \mathbf{C}(\boldsymbol{\mu})$ and $\mathbf{F}(\boldsymbol{\mu})$ depend on the parameter $\boldsymbol{\mu}$. Note that the operator $\mathbf{F}(\boldsymbol{\mu})$ corresponding to the nonlinear term in (3) can be reformulated with the Kronecker product $\mathbf{x}(t; \boldsymbol{\mu}) \otimes \mathbf{x}(t; \boldsymbol{\mu})$ instead of the vector $\mathbf{x}(t; \boldsymbol{\mu})^2$. Many reformulations exist because the Kronecker product does not account for the commutativity of the multiplication as discussed in the previous paragraph. Our operator inference will consider the formulation (i.e., the operators of the system (3)) as unknown and will use only the state vectors of (3), which are independent of the formulation of the system (3).

We restrict the following exposition to systems of ODEs with a nonlinear term of second order; however, the methodology is also applicable to systems of ODEs with nonlinear terms of higher order, see the numerical results in Section 4.3.

2.2. Time discretization

Let $0 = t_0 < t_1 < \dots < t_K = T$ be the discretization of the time domain $[0, T]$ into $K \in \mathbb{N}$ equidistant time steps with time step size $0 < \delta t \in \mathbb{R}$. Let further $\mathbf{x}_1(\boldsymbol{\mu}), \dots, \mathbf{x}_K(\boldsymbol{\mu}) \in \mathbb{R}^N$ be the discrete states of the system of ODEs (3) at the time steps t_1, \dots, t_K computed with a time stepping scheme, and let $\mathbf{y}_1(\boldsymbol{\mu}), \dots, \mathbf{y}_K(\boldsymbol{\mu}) \in \mathbb{R}^q$ be the corresponding outputs. We define the trajectory

$$\mathbf{X}(\boldsymbol{\mu}) = [\mathbf{x}_1(\boldsymbol{\mu}), \dots, \mathbf{x}_K(\boldsymbol{\mu})]^T \in \mathbb{R}^{K \times N} \quad (4)$$

with the corresponding output matrix

$$\mathbf{Y}(\boldsymbol{\mu}) = [\mathbf{y}_1(\boldsymbol{\mu}), \dots, \mathbf{y}_K(\boldsymbol{\mu})]^T \in \mathbb{R}^{K \times q} \quad (5)$$

and input matrix

$$\mathbf{U}(\boldsymbol{\mu}) = [\mathbf{u}(t_1; \boldsymbol{\mu}), \dots, \mathbf{u}(t_K; \boldsymbol{\mu})]^T \in \mathbb{R}^{K \times p}.$$

Note that the trajectory $\mathbf{X}(\boldsymbol{\mu})$ and the output matrix $\mathbf{Y}(\boldsymbol{\mu})$ contain the discrete states and the corresponding outputs derived with a time stepping scheme, whereas the input matrix $\mathbf{U}(\boldsymbol{\mu})$ contains the same inputs that are used in the continuous full model (3) at the discrete time steps t_1, \dots, t_K .

The discrete states in $\mathbf{X}(\boldsymbol{\mu})$ are taken at equidistant time steps but the time stepping scheme may vary the time step size during the computation; however, we assume that the time stepping scheme is convergent for the full model (3) in the sense that, for $\delta t \rightarrow 0$, the discrete states $\mathbf{x}_j(\boldsymbol{\mu})$ converge in the L^2 norm to the states $\mathbf{x}(t_j; \boldsymbol{\mu})$ of the system of ODEs (3) for $j = 1, \dots, K$. No further assumptions are made on the time stepping scheme.

2.3. Classical intrusive projection-based model reduction for systems of ODEs

Let $\boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_M \in \mathcal{D}$ be $M \in \mathbb{N}$ parameters and let $\mathbf{U}(\boldsymbol{\mu}_1), \dots, \mathbf{U}(\boldsymbol{\mu}_M) \in \mathbb{R}^{K \times p}$ and $\mathbf{x}_0(\boldsymbol{\mu}_1), \dots, \mathbf{x}_0(\boldsymbol{\mu}_M) \in \mathbb{R}^N$ be the corresponding input matrices and initial conditions, respectively. Let further $\mathbf{X}(\boldsymbol{\mu}_1), \dots, \mathbf{X}(\boldsymbol{\mu}_M) \in \mathbb{R}^{K \times N}$ be the trajectories and $\mathbf{Y}(\boldsymbol{\mu}_1), \dots, \mathbf{Y}(\boldsymbol{\mu}_M) \in \mathbb{R}^{K \times q}$ the output matrices.

Let $\mathbf{v}_1, \dots, \mathbf{v}_n \in \mathbb{R}^N$ be the (global) POD basis [1, 2] derived from the matrix of the concatenated initial conditions and trajectories

$$[\mathbf{x}_0(\boldsymbol{\mu}_1) \quad \mathbf{X}(\boldsymbol{\mu}_1)^T \quad \dots \quad \mathbf{x}_0(\boldsymbol{\mu}_M) \quad \mathbf{X}(\boldsymbol{\mu}_M)^T] \in \mathbb{R}^{N \times (KM+M)},$$

where $n \in \mathbb{N}$ is the number of POD basis vectors. The POD basis vectors are the columns of the POD basis matrix

$$\mathbf{V}_n = [\mathbf{v}_1, \dots, \mathbf{v}_n] \in \mathbb{R}^{N \times n}. \quad (6)$$

Note that the POD basis $\mathbf{v}_1, \dots, \mathbf{v}_n$ is independent of the parameter $\boldsymbol{\mu} \in \mathcal{D}$.

In classical projection-based model reduction, the reduced operators are obtained by projecting the ODEs onto the space spanned by the POD basis vectors via, e.g., Galerkin projection. Thus, the reduced operator $\tilde{\mathbf{A}}(\boldsymbol{\mu}_i) \in \mathbb{R}^{n \times n}$ is constructed as

$$\tilde{\mathbf{A}}(\boldsymbol{\mu}_i) = \mathbf{V}_n^T \mathbf{A}(\boldsymbol{\mu}_i) \mathbf{V}_n \in \mathbb{R}^{n \times n}, \quad (7)$$

for $i = 1, \dots, M$. Note that computing $\tilde{\mathbf{A}}(\boldsymbol{\mu}_i)$ as in (7) requires the full-model operator $\mathbf{A}(\boldsymbol{\mu}_i)$ or a method that provides the matrix-vector product with $\mathbf{A}(\boldsymbol{\mu}_i)$. Analogously, the respective full-model operators are required to construct the reduced input operator as $\tilde{\mathbf{B}}(\boldsymbol{\mu}_i) = \mathbf{V}_n^T \mathbf{B}(\boldsymbol{\mu}_i) \in \mathbb{R}^{n \times p}$ and the reduced output operator as $\tilde{\mathbf{C}}(\boldsymbol{\mu}_i) = \mathbf{C}(\boldsymbol{\mu}_i) \mathbf{V}_n \in \mathbb{R}^{q \times n}$ for $i = 1, \dots, M$. With $s = \binom{n+1}{2} = n(n+1)/2 \in \mathbb{N}$, the reduced operator $\tilde{\mathbf{F}}(\boldsymbol{\mu}_i) \in \mathbb{R}^{n \times s}$ of $\mathbf{F}(\boldsymbol{\mu}_i) \in \mathbb{R}^{N \times s}$ is obtained by first transforming $\mathbf{F}(\boldsymbol{\mu}_i)$ into a tensor of size $N \times N \times N$ and then performing Galerkin projection in each dimension separately. For each parameter $\boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_M$, the reduced model

$$\begin{aligned} \frac{d}{dt} \tilde{\mathbf{x}}(t; \boldsymbol{\mu}_i) &= \tilde{\mathbf{A}}(\boldsymbol{\mu}_i) \tilde{\mathbf{x}}(t; \boldsymbol{\mu}_i) + \tilde{\mathbf{F}}(\boldsymbol{\mu}_i) \tilde{\mathbf{x}}(t; \boldsymbol{\mu}_i)^2 + \tilde{\mathbf{B}}(\boldsymbol{\mu}_i) \mathbf{u}(t; \boldsymbol{\mu}_i) \\ \tilde{\mathbf{y}}(t; \boldsymbol{\mu}_i) &= \tilde{\mathbf{C}}(\boldsymbol{\mu}_i) \tilde{\mathbf{x}}(t; \boldsymbol{\mu}_i) \end{aligned} \quad (8)$$

is derived, with the reduced initial condition $\tilde{\mathbf{x}}_0(\boldsymbol{\mu}_i) = \mathbf{V}_n^T \mathbf{x}_0(\boldsymbol{\mu}_i) \in \mathbb{R}^n$, the reduced state vector $\tilde{\mathbf{x}}(t; \boldsymbol{\mu}_i) \in \mathbb{R}^n$, the reduced output $\tilde{\mathbf{y}}(t; \boldsymbol{\mu}_i) \in \mathbb{R}^q$, and the vector $\tilde{\mathbf{x}}(t; \boldsymbol{\mu}_i)^2 \in \mathbb{R}^s$ defined analogously as in (1). There are several ways to construct the reduced operators of the reduced model for a parameter $\boldsymbol{\mu} \in \mathcal{D}$ that does not coincide with any of the parameters $\boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_M$ [43]. Typically, the operators for $\boldsymbol{\mu}$ are constructed from the operators for $\boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_M$ via elementwise interpolation [44, 45, 46, 47].

3. Operator inference

Our goal is to derive the reduced model (8) for a black-box full model, where the trajectories and outputs can be computed, but where the full-model operators are unavailable. We consider the situation where, for parameters $\boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_M \in \mathcal{D}$, the initial conditions $\mathbf{x}_0(\boldsymbol{\mu}_1), \dots, \mathbf{x}_0(\boldsymbol{\mu}_M) \in \mathbb{R}^N$, the trajectories $\mathbf{X}(\boldsymbol{\mu}_1), \dots, \mathbf{X}(\boldsymbol{\mu}_M) \in \mathbb{R}^{K \times N}$, the input matrices $\mathbf{U}(\boldsymbol{\mu}_1), \dots, \mathbf{U}(\boldsymbol{\mu}_M) \in \mathbb{R}^{K \times p}$, and the output matrices $\mathbf{Y}(\boldsymbol{\mu}_1), \dots, \mathbf{Y}(\boldsymbol{\mu}_M) \in \mathbb{R}^{K \times q}$ are available, but where the corresponding operators $\mathbf{A}(\boldsymbol{\mu}_i), \mathbf{B}(\boldsymbol{\mu}_i), \mathbf{C}(\boldsymbol{\mu}_i)$, and

$F(\boldsymbol{\mu}_i)$ for $i = 1, \dots, M$ are unavailable. Our inference-based approach approximates the reduced operators $\tilde{\mathbf{A}}(\boldsymbol{\mu}_i)$, $\tilde{\mathbf{B}}(\boldsymbol{\mu}_i)$, $\tilde{\mathbf{C}}(\boldsymbol{\mu}_i)$, and $\tilde{\mathbf{F}}(\boldsymbol{\mu}_i)$ by the operators $\hat{\mathbf{A}}(\boldsymbol{\mu}_i) \in \mathbb{R}^{n \times n}$, $\hat{\mathbf{B}}(\boldsymbol{\mu}_i) \in \mathbb{R}^{n \times p}$, $\hat{\mathbf{C}}(\boldsymbol{\mu}_i) \in \mathbb{R}^{q \times n}$, and $\hat{\mathbf{F}}(\boldsymbol{\mu}_i) \in \mathbb{R}^{n \times s}$, which are inferred from the available initial conditions, inputs, trajectories, and outputs. A reduced model is then constructed with these inferred operators for the parameters $\boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_M$ as in (8) and for parameters $\boldsymbol{\mu} \in \mathcal{D}$ with elementwise interpolation as discussed in Section 2.3. Note that we consider the output operators $\mathbf{C}(\boldsymbol{\mu}_1), \dots, \mathbf{C}(\boldsymbol{\mu}_M)$ as unknown. In situations where the relationship of the states and the outputs is known, and thus where the output operators $\mathbf{C}(\boldsymbol{\mu}_1), \dots, \mathbf{C}(\boldsymbol{\mu}_M)$ are available, the operator inference for the output operators can be skipped, see Section 3.3.

The same operator inference is performed for each parameter $\boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_M$, and therefore, and to ease the exposition, we simplify the notation in this section. The discrete state $\mathbf{x}_j(\boldsymbol{\mu}_i)$ at time t_j and parameter $\boldsymbol{\mu}_i$ is denoted as \mathbf{x}_j , and analogously the input $\mathbf{u}(t_j; \boldsymbol{\mu})$ and output $\mathbf{y}_j(\boldsymbol{\mu})$ are denoted as \mathbf{u}_j and \mathbf{y}_j , respectively. The notation is also simplified for the trajectory $\mathbf{X} = \mathbf{X}(\boldsymbol{\mu}_i)$, the input matrix $\mathbf{U} = \mathbf{U}(\boldsymbol{\mu}_i)$ and the output matrix $\mathbf{Y} = \mathbf{Y}(\boldsymbol{\mu}_i)$. Also the full operator $\mathbf{A}(\boldsymbol{\mu}_i)$, the reduced operator $\tilde{\mathbf{A}}(\boldsymbol{\mu}_i)$, and the inferred operator $\hat{\mathbf{A}}(\boldsymbol{\mu}_i)$ are denoted as \mathbf{A} , $\tilde{\mathbf{A}}$, and $\hat{\mathbf{A}}$, respectively. Similarly, we have \mathbf{B} , $\tilde{\mathbf{B}}$, $\hat{\mathbf{B}}$, \mathbf{C} , $\tilde{\mathbf{C}}$, $\hat{\mathbf{C}}$, as well as \mathbf{F} , $\tilde{\mathbf{F}}$, $\hat{\mathbf{F}}$ for parameter $\boldsymbol{\mu}_i$.

Section 3.1 describes the projection of the trajectories onto the reduced space spanned by the POD basis vectors. Sections 3.2 and 3.3 derive optimization problems to infer operators. Algorithm 1 in Section 3.4 summarizes the computational procedure, and Section 3.5 provides a detailed cost analysis of the construction of a reduced model with operator inference. Section 3.6 summarizes practical considerations.

3.1. Data projection

Let $\mathbf{V}_n \in \mathbb{R}^{N \times n}$ be the POD basis matrix containing as columns the first $n \in \mathbb{N}$ POD basis vectors. The projected state $\hat{\mathbf{x}}(t) \in \mathbb{R}^n$ of the full-model state $\mathbf{x}(t) \in \mathbb{R}^N$ is $\hat{\mathbf{x}}(t) = \mathbf{V}_n^T \mathbf{x}(t)$. We have

$$\hat{\mathbf{x}}_j = \mathbf{V}_n^T \mathbf{x}_j, \quad j = 0, \dots, K, \quad (9)$$

for the discrete states $\mathbf{x}_1, \dots, \mathbf{x}_K$, which we assemble into the projected trajectory

$$\hat{\mathbf{X}} = [\hat{\mathbf{x}}_1, \dots, \hat{\mathbf{x}}_K]^T \in \mathbb{R}^{K \times n}. \quad (10)$$

We also define

$$\hat{\mathbf{X}}^{(i)} = [\hat{\mathbf{x}}_1^{(i)}, \dots, \hat{\mathbf{x}}_K^{(i)}]^T \in \mathbb{R}^{K \times i} \quad (11)$$

for $i = 1, \dots, n$, where the vectors $\hat{\mathbf{x}}_1^{(i)}, \dots, \hat{\mathbf{x}}_K^{(i)} \in \mathbb{R}^i$ are derived from the projected states $\hat{\mathbf{x}}_1, \dots, \hat{\mathbf{x}}_K$ analogously as in (2).

3.2. Inference-based reduced operators

Consider the time steps t_1, \dots, t_K with time step size δt , see Section 2.2. For $j = 1, \dots, K$, let $\dot{\hat{\mathbf{x}}}_j \in \mathbb{R}^n$ be an approximation of the derivative $\frac{d}{dt} \hat{\mathbf{x}}(t_j)$ of the projected state $\hat{\mathbf{x}}(t_j)$ at time t_j that converges with $\delta t \rightarrow 0$ to $\frac{d}{dt} \hat{\mathbf{x}}(t_j)$ in the L^2 norm. A concrete example of such an approximation will follow in Section 3.4. We define the inferred operators $\hat{\mathbf{A}} \in \mathbb{R}^{n \times n}$, $\hat{\mathbf{B}} \in \mathbb{R}^{n \times p}$, and $\hat{\mathbf{F}} \in \mathbb{R}^{n \times s}$ to be a solution of the optimization problem

$$\underset{\hat{\mathbf{A}} \in \mathbb{R}^{n \times n}, \hat{\mathbf{B}} \in \mathbb{R}^{n \times p}, \hat{\mathbf{F}} \in \mathbb{R}^{n \times s}}{\text{minimize}} \sum_{j=1}^K \left\| \dot{\hat{\mathbf{x}}}_j - \hat{\mathbf{A}} \hat{\mathbf{x}}_j - \hat{\mathbf{F}} \hat{\mathbf{x}}_j^2 - \hat{\mathbf{B}} \mathbf{u}_j \right\|_2^2, \quad (12)$$

which means that the inferred operators satisfy the equations

$$\dot{\hat{\mathbf{x}}}_j = \hat{\mathbf{A}} \hat{\mathbf{x}}_j + \hat{\mathbf{F}} \hat{\mathbf{x}}_j^2 + \hat{\mathbf{B}} \mathbf{u}_j, \quad j = 1, \dots, K, \quad (13)$$

if the objective of (12) is zero at the optimum.

Let

$$\mathbf{D} = [\hat{\mathbf{X}}, \mathbf{U}, \hat{\mathbf{X}}^{(1)}, \dots, \hat{\mathbf{X}}^{(n)}] \in \mathbb{R}^{K \times (n+p+s)} \quad (14)$$

be the data matrix, where $\hat{\mathbf{X}}$ is the projected trajectory, \mathbf{U} the input matrix, and $\hat{\mathbf{X}}^{(1)}, \dots, \hat{\mathbf{X}}^{(n)}$ the matrices as defined in equation (11). We also define the right-hand side matrix

$$\mathbf{R} = [\hat{\mathbf{x}}_1, \dots, \hat{\mathbf{x}}_K]^T \in \mathbb{R}^{K \times n}. \quad (15)$$

We now show that the optimization problem (12) is a linear least-squares problem with the system matrix \mathbf{D} and the right-hand side matrix \mathbf{R} . We therefore first transpose the terms in the norm in the objective of (12) and obtain

$$\left\| \dot{\hat{\mathbf{x}}}_j^T - \hat{\mathbf{x}}_j^T \hat{\mathbf{A}}^T - (\hat{\mathbf{x}}_j^2)^T \hat{\mathbf{F}}^T - \mathbf{u}_j^T \hat{\mathbf{B}}^T \right\|_2^2. \quad (16)$$

Consider now $\hat{\mathbf{F}}$ in block form as

$$\hat{\mathbf{F}} = [\hat{\mathbf{F}}^{(1)}, \dots, \hat{\mathbf{F}}^{(n)}] \in \mathbb{R}^{n \times s}, \quad (17)$$

with the blocks $\hat{\mathbf{F}}^{(i)} \in \mathbb{R}^{n \times i}$ of size $n \times i$ for $i = 1, \dots, n$, respectively. Because the squared Frobenius norm of a matrix is the sum of the squares of all entries of the matrix, the sum over the K time steps in (12) can be written as a Frobenius norm such that the optimization problem (12) becomes

$$\underset{\hat{\mathbf{A}} \in \mathbb{R}^{n \times n}, \hat{\mathbf{B}} \in \mathbb{R}^{n \times p}, \hat{\mathbf{F}} \in \mathbb{R}^{n \times n^k}}{\text{minimize}} \left\| \mathbf{R} - \hat{\mathbf{X}} \hat{\mathbf{A}}^T - \mathbf{U} \hat{\mathbf{B}}^T - \sum_{i=1}^n \hat{\mathbf{X}}^{(i)} \hat{\mathbf{F}}^{(i)T} \right\|_F^2.$$

We obtain with the data matrix \mathbf{D} and the matrix \mathbf{O} the least-squares problem

$$\underset{\mathbf{O} \in \mathbb{R}^{n \times (n+p+s)}}{\text{minimize}} \|\mathbf{D}\mathbf{O}^T - \mathbf{R}\|_F^2, \quad (18)$$

where $\mathbf{O} \in \mathbb{R}^{n \times (n+p+s)}$ and

$$\mathbf{O} = [\hat{\mathbf{A}}, \hat{\mathbf{B}}, \hat{\mathbf{F}}^{(1)}, \dots, \hat{\mathbf{F}}^{(n)}] \in \mathbb{R}^{n \times (n+p+s)}. \quad (19)$$

The least-squares problem (18) can be transformed into n independent least-squares problems in the $\|\cdot\|_2$ norm, see, e.g., [40, Lemma 1]. Consider the columns of $\mathbf{R} = [\mathbf{r}_1, \dots, \mathbf{r}_n] \in \mathbb{R}^{K \times n}$. We obtain the n least-squares problems

$$\underset{\mathbf{o}_i \in \mathbb{R}^{n+p+s}}{\text{minimize}} \|\mathbf{D}\mathbf{o}_i - \mathbf{r}_i\|_2^2, \quad i = 1, \dots, n \quad (20)$$

that are solved by the columns $\mathbf{o}_1, \dots, \mathbf{o}_n \in \mathbb{R}^{n+p+s}$ of $\mathbf{O}^T \in \mathbb{R}^{(n+p+s) \times n}$. Therefore, from each of the n independent least-squares problems, one column of \mathbf{O}^T is obtained.

Theorem 1. Consider the reduced model (8) and let $\tilde{\mathbf{x}}(t_1), \dots, \tilde{\mathbf{x}}(t_K) \in \mathbb{R}^n$ be the reduced states at time steps t_1, \dots, t_K that satisfy

$$\frac{d}{dt} \tilde{\mathbf{x}}(t_j) = \tilde{\mathbf{A}} \tilde{\mathbf{x}}(t_j) + \tilde{\mathbf{F}} \tilde{\mathbf{x}}(t_j)^2 + \tilde{\mathbf{B}} \mathbf{u}(t_j), \quad j = 1, \dots, K. \quad (21)$$

Define the reduced trajectory as $\tilde{\mathbf{X}} = [\tilde{\mathbf{x}}(t_1), \dots, \tilde{\mathbf{x}}(t_K)]^T \in \mathbb{R}^{K \times n}$ and the vectors $\tilde{\mathbf{X}}^{(1)}, \dots, \tilde{\mathbf{X}}^{(n)}$ analogously as for the projected trajectory in (11). Let $\tilde{\mathbf{D}} \in \mathbb{R}^{K \times (n+p+s)}$ be the data matrix assembled from the reduced trajectory

$$\tilde{\mathbf{D}} = [\tilde{\mathbf{X}}, \mathbf{U}, \tilde{\mathbf{X}}^{(1)}, \dots, \tilde{\mathbf{X}}^{(n)}] \in \mathbb{R}^{K \times (n+p+s)},$$

and $\tilde{\mathbf{R}} \in \mathbb{R}^{K \times n}$ the corresponding right-hand side matrix

$$\tilde{\mathbf{R}} = \left[\frac{d}{dt} \tilde{\mathbf{x}}(t_1), \dots, \frac{d}{dt} \tilde{\mathbf{x}}(t_K) \right]^T.$$

If the time stepping scheme used in the full model is convergent for $\delta t \rightarrow 0$ (see Section 2.2), if $\hat{\mathbf{x}}_j$ converges in the L^2 norm to $\frac{d}{dt} \tilde{\mathbf{x}}(t_j)$ for $\delta t \rightarrow 0$ and $j = 1, \dots, K$, and if the data matrices $\tilde{\mathbf{D}}$ and \mathbf{D} have full column rank $n+p+s$ for all $0 < \delta t \in \mathbb{R}$ and for all $n \leq N$, then there exists for $0 < \epsilon \in \mathbb{R}$ an $n \leq N$ and a time step

size δt such that the Frobenius norm of the difference of the inferred $\hat{\mathbf{A}}$ and the (intrusive) reduced operator $\tilde{\mathbf{A}}$ is below the threshold ϵ

$$\|\hat{\mathbf{A}} - \tilde{\mathbf{A}}\|_F < \epsilon.$$

The same holds for the operators $\hat{\mathbf{B}}$ and $\hat{\mathbf{F}}$.

Proof. Consider (12) and the corresponding optimization problem (18) with the data matrix $\tilde{\mathbf{D}}$ and the right-hand side matrix $\tilde{\mathbf{R}}$ that are assembled from the reduced trajectory $\tilde{\mathbf{X}}$. The reduced operators $\tilde{\mathbf{A}}, \tilde{\mathbf{B}}$, and $\tilde{\mathbf{F}}$ are a solution of the optimization problem because the reduced states $\tilde{\mathbf{x}}(t_1), \dots, \tilde{\mathbf{x}}(t_K)$ satisfy equations (21). Furthermore, the reduced operators are the unique solution because the data matrix $\tilde{\mathbf{D}}$ has full column rank $n + p + s$.

The projected trajectory $\hat{\mathbf{X}}$ can be interpreted as a perturbed matrix of the reduced trajectory $\tilde{\mathbf{X}}$, i.e., $\hat{\mathbf{X}} = \tilde{\mathbf{X}} + \delta\tilde{\mathbf{X}}$ with the perturbation $\delta\tilde{\mathbf{X}} \in \mathbb{R}^{K \times n}$. For $\delta t \rightarrow 0$ and $n \rightarrow N$, the norm of the perturbation converges to zero $\|\delta\tilde{\mathbf{X}}\|_F \rightarrow 0$ because the time stepping scheme of the full model is convergent and the reduced model (8) becomes a reformulation of the full model (3) in the POD basis $\mathbf{V}_n = [\mathbf{v}_1, \dots, \mathbf{v}_n]$ for $n = N$. Note that the data matrix has full column rank for all $n \leq N$ and therefore the POD space of dimension N exists and is \mathbb{R}^N . This means that $\hat{\mathbf{x}}_j$ converges in the L^2 norm to $\frac{d}{dt}\tilde{\mathbf{x}}(t_j)$ for $\delta t \rightarrow 0$ and $n \rightarrow N$, for $j = 1, \dots, K$. Therefore, there exists for all $0 < \epsilon_1 \in \mathbb{R}$ a time step size δt and a dimension $n \leq N$ such that $\|\tilde{\mathbf{D}} - \mathbf{D}\|_F < \epsilon_1$ and $\|\tilde{\mathbf{R}} - \mathbf{R}\|_F < \epsilon_1$. The perturbation analysis of full-rank least-squares problems [48, 49] shows $\|\hat{\mathbf{A}} - \tilde{\mathbf{A}}\|_F < c\epsilon_1$ for a constant $c \in \mathbb{R}$. The constant c is independent of ϵ_1 . Setting $\epsilon_1 = \epsilon/c$ leads to $\|\hat{\mathbf{A}} - \tilde{\mathbf{A}}\|_F < \epsilon$. \square

Theorem 1 shows that the inferred operators converge with $\delta t \rightarrow 0$ to the reduced operators that would be obtained via intrusive Galerkin projection. Our inferred operators therefore will eventually inherit the properties of the reduced operators. In particular, if the reduced model based on the (intrusive) reduced operators is asymptotically stable, the inferred operators will eventually also lead to an asymptotically stable reduced model.

3.3. Inference-based reduced output operators

For the inference of the output operator $\hat{\mathbf{C}} \in \mathbb{R}^{q \times n}$, we consider the least-squares problem

$$\underset{\hat{\mathbf{C}} \in \mathbb{R}^{q \times n}}{\text{minimize}} \sum_{i=1}^K \left\| \mathbf{y}_i - \hat{\mathbf{C}} \hat{\mathbf{x}}_i \right\|_2^2. \quad (22)$$

Optimization problem (22) is cast into a least-squares problem

$$\underset{\hat{\mathbf{C}} \in \mathbb{R}^{q \times n}}{\text{minimize}} \|\hat{\mathbf{X}} \hat{\mathbf{C}}^T - \mathbf{Y}^T\|_F^2, \quad (23)$$

where the operator $\hat{\mathbf{C}}^T$ is obtained columnwise as in (20).

Corollary 1. *Consider the same setting as in Theorem 1. If $\hat{\mathbf{X}}$ has full column rank n for all $0 < \delta t \in \mathbb{R}$ and for all $n \leq N$, then there exists for $0 < \epsilon \in \mathbb{R}$ an $n \leq N$ and a time step size δt such that*

$$\|\hat{\mathbf{C}} - \tilde{\mathbf{C}}\|_F < \epsilon.$$

Proof. The same arguments as in Theorem 1 apply. \square

Algorithm 1 Operator inference for models with second-order nonlinear terms

- 1: **procedure** INFEROP($\mathbf{x}_0, \mathbf{X}, \mathbf{U}, \mathbf{Y}, \mathbf{V}_n$)
- 2: Project $\hat{\mathbf{x}}_0 = \mathbf{V}_n^T \mathbf{x}_0$ and $\hat{\mathbf{X}} = \mathbf{X} \mathbf{V}_n$
- 3: Assemble matrices $\hat{\mathbf{X}}^{(1)}, \dots, \hat{\mathbf{X}}^{(n)}$ following (11)
- 4: Assemble data matrix \mathbf{D} following (14) as

$$\mathbf{D} = [\hat{\mathbf{X}}, \mathbf{U}, \hat{\mathbf{X}}^{(1)}, \dots, \hat{\mathbf{X}}^{(n)}] \in \mathbb{R}^{K \times (n+p+s)}$$

- 5: Assemble right-hand side matrix \mathbf{R} using the approximation (24)

$$\mathbf{R} = \left[\frac{\hat{\mathbf{x}}_1 - \hat{\mathbf{x}}_0}{\delta t}, \dots, \frac{\hat{\mathbf{x}}_K - \hat{\mathbf{x}}_{K-1}}{\delta t} \right]^T \in \mathbb{R}^{K \times n}$$

- 6: Extract columns $\mathbf{R} = [\mathbf{r}_1, \dots, \mathbf{r}_n]$ and solve for $\mathbf{O}^T = [\mathbf{o}_1, \dots, \mathbf{o}_n]$ in

$$\|\mathbf{D} \mathbf{o}_i - \mathbf{r}_i\|_2^2, \quad i = 1, \dots, n$$

- 7: Extract operators $\hat{\mathbf{A}}, \hat{\mathbf{B}}, \hat{\mathbf{F}}^{(1)}, \dots, \hat{\mathbf{F}}^{(n)}$ from \mathbf{O} as in (19)
- 8: Assemble

$$\hat{\mathbf{F}} = [\hat{\mathbf{F}}^{(1)}, \dots, \hat{\mathbf{F}}^{(n)}]$$

- 9: Solve least-squares problem (23) to construct matrix $\hat{\mathbf{C}}$
 - 10: **return** $\hat{\mathbf{A}}, \hat{\mathbf{B}}, \hat{\mathbf{C}}, \hat{\mathbf{F}}$
 - 11: **end procedure**
-

3.4. Computational procedure

In the following, we use the finite difference

$$\dot{\hat{\mathbf{x}}}_j = \frac{\hat{\mathbf{x}}_j - \hat{\mathbf{x}}_{j-1}}{\delta t}, \quad j = 1, \dots, K, \quad (24)$$

where $\hat{\mathbf{X}} = [\hat{\mathbf{x}}_1, \dots, \hat{\mathbf{x}}_K]^T \in \mathbb{R}^{K \times n}$ is the projected trajectory. Note that the finite difference (24) converges to $\frac{d}{dt} \hat{\mathbf{x}}(t_j)$ for $\delta t \rightarrow 0$ and $j = 1, \dots, K$. Note further that once the operators are inferred, any time stepping scheme (any time discretization) can be used to march forward in time the reduced model with the inferred operators, because we are inferring the operators of the system of ODEs and not the operators of the time-discretized system. Also recall that the trajectory \mathbf{X} of the full model can be computed with any time stepping scheme, see Section 2.2.

Algorithm 1 summarizes the inference of the operators from initial conditions, inputs, trajectories, and outputs as discussed in Sections 3.2 and 3.3. Inputs to Algorithm 1 are the initial condition \mathbf{x}_0 , the trajectory $\mathbf{X} \in \mathbb{R}^{K \times N}$, the input matrix $\mathbf{U} \in \mathbb{R}^{K \times p}$, the output matrix $\mathbf{Y} \in \mathbb{R}^{K \times q}$, and the POD basis matrix $\mathbf{V}_n \in \mathbb{R}^{N \times n}$. The first step of Algorithm 1 is to project the trajectories onto the POD space spanned by the POD basis vectors in the matrix \mathbf{V}_n . Note that the trajectory \mathbf{X} contains the transposed discrete state vectors as rows and therefore the projection is a right multiplication of \mathbf{X} with \mathbf{V}_n . The data matrix $\mathbf{D} \in \mathbb{R}^{K \times (n+p+s)}$ and the right-hand side matrix $\mathbf{R} \in \mathbb{R}^{K \times n}$ are assembled as defined in (14) and (15), respectively. The columns of the matrix \mathbf{O}^T are computed with the n independent least-squares problems as in (20), and the inferred operators $\hat{\mathbf{A}}, \hat{\mathbf{B}}, \hat{\mathbf{F}}^{(1)}, \dots, \hat{\mathbf{F}}^{(n)}$ are extracted from \mathbf{O}^T as in (19), from which the operator $\hat{\mathbf{F}}$ is assembled. The output operator $\hat{\mathbf{C}}$ is inferred by solving the least-squares problem (23).

3.5. Computational costs

The following cost analysis ignores the costs of the data generation and the construction of the POD basis because these steps are the same as in classical intrusive projection-based model reduction. We therefore

consider the costs of Algorithm 1 and compare it to the costs of the classical intrusive construction of the reduced operators as presented in Section 2.3.

Let us first consider the costs of Algorithm 1. Note that Algorithm 1 is applicable in case of a second-order nonlinear term; we present a separate cost analysis for higher-order nonlinear terms below. The costs of the projection of the trajectory onto the POD space are in $\mathcal{O}(nNK)$ and thus are bounded linearly in the dimension N of the full model and the number of time steps K . Note that sometimes $K > N$, i.e., the number of time steps is larger than the dimension of the full model. Assembling the matrices $\hat{\mathbf{X}}^{(1)}, \dots, \hat{\mathbf{X}}^{(n)}$ induces costs that are bounded by $\mathcal{O}(Kn^2)$. The data matrix \mathbf{D} and the right-hand side matrix \mathbf{R} are assembled with costs in $\mathcal{O}(K(n+p+s))$ and $\mathcal{O}(Kn)$, respectively. A crude upper bound of the costs of solving one of the n independent least-squares problems is $\mathcal{O}(K(n+p+s)^3)$, see, e.g., [50, Sec. 5.3.3]. Therefore, the costs for solving the least-squares problem (18), and thus the optimization problem (12), are bounded by $\mathcal{O}(nK(n+p+s)^3)$. Similarly, the least-squares problem (22) to derive the output matrix $\hat{\mathbf{C}}$ is constructed with costs in $\mathcal{O}(Kq)$ and solved in $\mathcal{O}(qn^3K)$.

Overall, the costs of Algorithm 1 are bounded linearly in the dimension N of the full model and linearly in the number of time steps K . Only the projection of the trajectories onto the POD space incurs costs that scale with the dimension N of the full model. All other computations in Algorithm 1 have costs independent of the dimension N of the full model. The costs of the classical construction of the reduced operators depend on the sparsity of the full-model operators. If the full-model operators are sparse, then the costs of the classical construction scale linearly in the dimension N . If the full-model operators are dense or sparsity is not exploited, then the costs are quadratic in the dimension N of the full model. Furthermore, the costs of the classical construction of the reduced operators are independent of the number of time steps K . The costs of computing a reduced state with a reduced model are independent of whether the reduced model is constructed with operator inference or classical intrusive projection-based model reduction.

Consider now a full model (3) with a nonlinear term of order greater than 2, i.e., $\alpha > 2$. The vector $\hat{\mathbf{x}}_j^\alpha \in \mathbb{R}^s$ at a time step $j \in \{1, \dots, K\}$ has dimension $s = \binom{n+\alpha-1}{\alpha}$, see Section 2.1. The dimension s is bounded by $\mathcal{O}(n^\alpha)$ and can grow exponentially in the order α . This means that the costs of assembling the data matrix $\mathbf{D} \in \mathbb{R}^{K \times (n+p+s)}$, including the assembly of the matrices $\hat{\mathbf{X}}^{(i)}$, grow exponentially in the order α . Additionally, the costs of solving the least-squares problem (18) grow exponentially in α . Our approach is therefore limited to polynomial nonlinear terms of low order.

3.6. Practical considerations

A data matrix $\mathbf{D} \in \mathbb{R}^{K \times (n+p+s)}$ with a large condition number can introduce significant numerical errors into the solutions of the n least-squares problems (20), and thus into the inferred operators. Monitoring the singular values of the data matrix helps detecting such a situation. In our case, a large condition number typically arises if the states at different time steps are almost linearly dependent. This can happen either because the time step size δt is small, and therefore the states of two successive time steps are similar, or because the states converge quickly to a steady state. One remedy that empirically is found to reduce the condition number of the data matrix \mathbf{D} in case the states converge quickly to a steady state is combining trajectories of different inputs and initial conditions. The data matrix derived from the combined trajectories contains more states in the transient regime than a data matrix derived from a single trajectory only. For combining trajectories of different inputs and initial conditions, let $\mathbf{X}_1, \dots, \mathbf{X}_{M'} \in \mathbb{R}^{K \times N}$ be $M' \in \mathbb{N}$ trajectories corresponding to the input matrices $\mathbf{U}_1, \dots, \mathbf{U}_{M'} \in \mathbb{R}^{K \times p}$ and M' initial conditions. Note that the trajectories $\mathbf{X}_1, \dots, \mathbf{X}_{M'}$ correspond to a single parameter $\boldsymbol{\mu}_i$, see the notation defined at the beginning of Section 3. The matrices

$$\mathbf{X} = \begin{bmatrix} \mathbf{X}_1 \\ \vdots \\ \mathbf{X}_{M'} \end{bmatrix} \in \mathbb{R}^{M'K \times N}, \quad \mathbf{U} = \begin{bmatrix} \mathbf{U}_1 \\ \vdots \\ \mathbf{U}_{M'} \end{bmatrix} \in \mathbb{R}^{M'K \times p} \quad (25)$$

and the corresponding initial conditions are then passed to Algorithm 1 such that the data matrix \mathbf{D} and the right-hand side matrix \mathbf{R} are assembled from (25) and the M' initial conditions. We emphasize that this

is a heuristic strategy and that many other strategies exist that might help to reduce the condition number of the data matrix. For example, if a small time step size leads to a large condition number (see above) then neglecting states to remove linear dependency might reduce the condition number.

The eigenvalues of the inferred operator $\hat{\mathbf{A}} \in \mathbb{R}^{n \times n}$ serve as an indicator for the stability of the reduced model in the following. A linear reduced model is asymptotically stable if the eigenvalues $\lambda_1, \dots, \lambda_n \in \mathbb{C}$ of $\hat{\mathbf{A}}$ have negative real parts [51, Section 5.8]. In case of a nonlinear reduced model, the eigenvalues $\lambda_1, \dots, \lambda_n$ become stability indicators.

4. Numerical experiments

We first demonstrate operator inference on two synthetic examples. Section 4.1 discusses the heat equation and Section 4.2 the Burgers' equation. We then consider in Section 4.3 a tubular reactor with a limit cycle oscillation (LCO) [52]. The governing equations of the tubular reactor include a nonlinear term of third order. Section 4.4 infers a reduced model of the GCM, which we use for an ocean gyre circulation experiment [6].

4.1. Heat equation: linear problem

Let $\Omega = [0, 1] \subset \mathbb{R}$ be the spatial domain and $[0, T] \subset \mathbb{R}$ the time domain with $T = 1$. The parameter domain is $\mathcal{D} = [0.1, 10] \subset \mathbb{R}$. We consider the parametrized heat equation

$$\frac{\partial}{\partial t} x(\omega, t; \mu) - \mu \frac{\partial^2}{\partial \omega^2} x(\omega, t; \mu) = 0, \quad (26)$$

with the state $x(\omega, t; \mu)$ at the spatial coordinate $\omega \in \Omega$, time $t \in [0, T]$, and the parameter $\mu \in \mathcal{D}$. Dirichlet boundary conditions $x(0, t; \mu) = x(1, t; \mu) = 1$ for $t \in [0, T]$ and $\mu \in \mathcal{D}$ are imposed. Discretization of the spatial domain Ω on an equidistant grid with mesh width 2^{-7} with the finite difference method leads to the system of $N = 127$ parametrized linear ODEs

$$\frac{d}{dt} \mathbf{x}(t; \mu) = \mathbf{A}(\mu) \mathbf{x}(t; \mu) + \mathbf{B}(\mu) \mathbf{u}(t), \quad (27)$$

with the state vector $\mathbf{x}(t; \mu) \in \mathbb{R}^N$ and the linear operator $\mathbf{A}(\mu) \in \mathbb{R}^{N \times N}$. The input matrix $\mathbf{B}(\mu) \in \mathbb{R}^{N \times 1}$ imposes the Dirichlet boundary conditions with the constant input $u(t) = 1$ for all $t \in [0, T]$, and thus we set $p = 1$. The initial condition is $\mathbf{x}_0(\mu) = \mathbf{0} \in \mathbb{R}^N$. The output $y(t; \mu) \in \mathbb{R}$ is defined as the average of the components of $\mathbf{x}(t; \mu) \in \mathbb{R}^N$, which leads to

$$\mathbf{C} = \left[\frac{1}{N}, \dots, \frac{1}{N} \right] \in \mathbb{R}^{1 \times N}, \quad (28)$$

and thus $q = 1$. The operators $\mathbf{A}(\mu)$ and $\mathbf{B}(\mu)$ depend on the parameter $\mu \in \mathcal{D}$. The system of ODEs (27) is marched forward in time with the implicit Euler scheme with time step size $\delta t = 10^{-3}$ and $K = 10^3$ time steps.

Let $\mu_1, \dots, \mu_M \in \mathcal{D}$ be $M = 10$ parameters equidistantly distributed in \mathcal{D} , and let $\mathbf{X}(\mu_1), \dots, \mathbf{X}(\mu_M) \in \mathbb{R}^{K \times N}$, $\mathbf{U} = [1, \dots, 1]^T \in \mathbb{R}^{K \times 1}$, and $\mathbf{Y}(\mu_1), \dots, \mathbf{Y}(\mu_M) \in \mathbb{R}^{K \times 1}$ be the corresponding trajectories, input, and output matrices of the system of ODEs (27). Note that the input matrix $\mathbf{U} \in \mathbb{R}^{K \times 1}$ and the initial condition are independent of the parameter $\mu \in \mathcal{D}$ in this example. We construct a POD basis and put the first $w = 8$ POD basis vectors into the POD basis matrix $\mathbf{V}_w \in \mathbb{R}^{N \times w}$ such that the projection error

$$\frac{1}{M} \sum_{i=1}^M \frac{\|\mathbf{X}(\mu_i) - \mathbf{X}(\mu_i) \mathbf{V}_w \mathbf{V}_w^T\|_F^2}{\|\mathbf{X}(\mu_i)\|_F^2} \quad (29)$$

is about 10^{-5} . The operators $\hat{\mathbf{A}}(\mu_i) \in \mathbb{R}^{w \times w}$, $\hat{\mathbf{B}}(\mu_i) \in \mathbb{R}^{w \times 1}$, and $\hat{\mathbf{C}}(\mu_i) \in \mathbb{R}^{1 \times w}$ are inferred with Algorithm 1 for $i = 1, \dots, M$ and used to derive inference-based reduced models of dimension w corresponding

to the parameters μ_1, \dots, μ_M . An inference-based reduced model of dimension $n < w$ is constructed by extracting the submatrix of size $n \times n$ from $\hat{\mathbf{A}}(\mu_i) \in \mathbb{R}^{w \times w}$ that corresponds to the first n POD basis vectors, and analogously for the operators $\hat{\mathbf{B}}(\mu_i)$ and $\hat{\mathbf{C}}(\mu_i)$. Note that this requires performing the operator inference with Algorithm 1 only once. Our approach is equivalent to DMD (see Section 1) in this example (linear full model and input constant over time) and therefore the inferred operators are the same operators that would be obtained via DMD as given in [35, Algorithm 1]. The operators for a parameter $\mu \in \mathcal{D}$ are derived via elementwise spline interpolation, see [46, Section 3.1]. For comparison, we also construct the intrusive reduced models as discussed in Section 2.3. The stability check described in Section 3.6 indicates stability for the intrusive and the inference-based reduced models in this example.

Figure 2 compares the errors of the inference-based and the intrusive reduced models. Figure 2a reports the average error of the states

$$\frac{1}{M} \sum_{j=1}^M \frac{\|\mathbf{X}(\mu_j) - \bar{\mathbf{X}}(\mu_j) \mathbf{V}_n^T\|_F^2}{\|\mathbf{X}(\mu_j)\|_F^2}, \quad (30)$$

where $\bar{\mathbf{X}}(\mu_j)$ is the trajectory obtained with either the inference-based or the intrusive reduced model and $\mathbf{V}_n \in \mathbb{R}^{N \times n}$ the POD basis matrix with the first n POD basis vectors. Figure 2b shows the average error of the states for $M_{\text{test}} = 5$ test parameters $\mu_{M+1}, \dots, \mu_{M+M_{\text{test}}} \in \mathcal{D}$. The test parameters are drawn randomly from a uniform distribution in \mathcal{D} and are different from the training parameters μ_1, \dots, μ_M for which operators were inferred with Algorithm 1. For both the training and the test parameters, the inference-based and the intrusive reduced model achieve a similar error (30). A similar result is obtained for the average output error

$$\frac{1}{M} \sum_{j=1}^M \frac{\|\mathbf{Y}(\mu_j) - \bar{\mathbf{Y}}(\mu_j)\|_F^2}{\|\mathbf{Y}(\mu_j)\|_F^2}, \quad (31)$$

where $\bar{\mathbf{Y}}(\mu_j)$ is the output matrix obtained with either the inference-based or the intrusive reduced model. The average output error is shown in Figures 2c and 2d for the training and the test parameters, respectively.

4.2. Burgers' equation: nonlinear term of second order

Let $\Omega = [0, 1] \subset \mathbb{R}$ be the spatial domain and $[0, T] \subset \mathbb{R}$ the time domain with $T = 1$. Let further $\mathcal{D} = [0.1, 1] \subset \mathbb{R}$ be the parameter domain. The viscous Burgers' equation is

$$\frac{\partial}{\partial t} x(\omega, t; \mu) + x(\omega, t; \mu) \frac{\partial}{\partial \omega} x(\omega, t; \mu) - \mu \frac{\partial^2}{\partial \omega^2} x(\omega, t; \mu) = 0, \quad (32)$$

where $x(\omega, t; \mu)$ is the state at spatial coordinate $\omega \in \Omega$, time $t \in [0, T]$, and parameter $\mu \in \mathcal{D}$. We impose Dirichlet boundary conditions on $x(0, t; \mu) = u(t)$ and $x(1, t; \mu) = -u(t)$ with the input $u : [0, T] \rightarrow \mathbb{R}$. Discretization with the finite difference method on an equidistant grid with mesh width 2^{-7} leads to the system of parametrized nonlinear ODEs

$$\frac{d}{dt} \mathbf{x}(t; \mu) = \mathbf{A}(\mu) \mathbf{x}(t; \mu) + \mathbf{F} \mathbf{x}(t; \mu)^2 + \mathbf{B}(\mu) u(t), \quad (33)$$

where $\mathbf{x}(t; \mu) \in \mathbb{R}^N$ is the state vector, $\mathbf{A}(\mu) \in \mathbb{R}^{N \times N}$ the operator corresponding to the linear terms of the Burgers' equation, and $\mathbf{F} \in \mathbb{R}^{N \times S}$ the operator corresponding to the nonlinear term. Note that \mathbf{F} is independent of the parameter μ in this example. The input matrix $\mathbf{B}(\mu)$ with input $u(t)$ at time t imposes the Dirichlet boundary conditions $x(0, t; \mu) = u(t)$ and $x(1, t; \mu) = -u(t)$. The initial condition is $\mathbf{x}_0(\mu) = \mathbf{0} \in \mathbb{R}^N$. We do not consider an output in this example. The system of ODEs (33) is solved for a parameter $\mu \in \mathcal{D}$ with the semi-implicit Euler scheme and time step size $\delta t = 10^{-4}$ and $K = 10^4$.

Let $\mu_1, \dots, \mu_M \in \mathcal{D}$ be $M = 10$ parameters equidistantly distributed in \mathcal{D} . We generate $M' = 10$ input matrices $\mathbf{U}_1(\mu_i), \dots, \mathbf{U}_{M'}(\mu_i)$ for each parameter $\mu_i, i = 1, \dots, M$, where each component of the input matrices is drawn randomly from a uniform distribution in $[0, 1] \subset \mathbb{R}$. The M' trajectories $\mathbf{X}_1(\mu_i), \dots, \mathbf{X}_{M'}(\mu_i) \in$

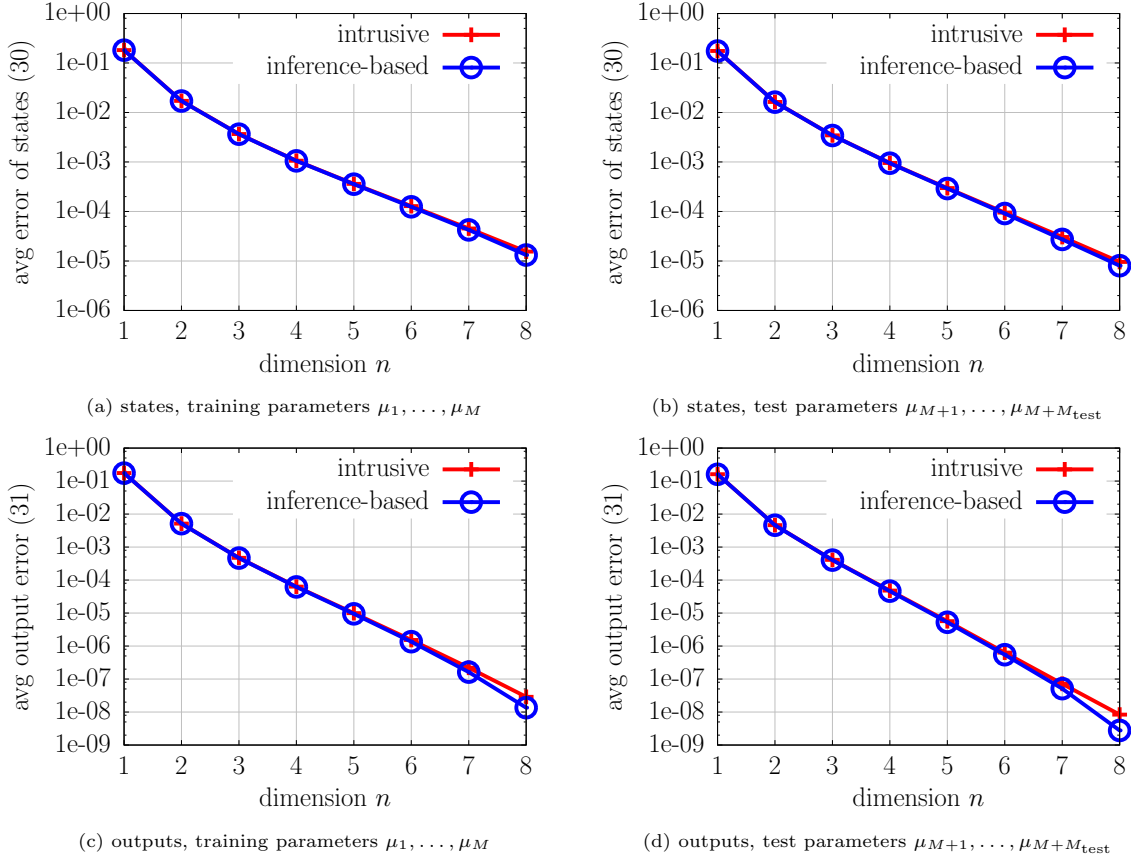


Figure 2: Heat equation: The intrusive reduced model is derived with the classical intrusive construction of the reduced operators and the nonintrusive, inference-based reduced model with the operator inference summarized in Algorithm 1; thus, the construction of the inference-based reduced model does not require the full-model operators. The plots show that the inferred operators lead to a reduced model with a similar behavior to the intrusive reduced model.

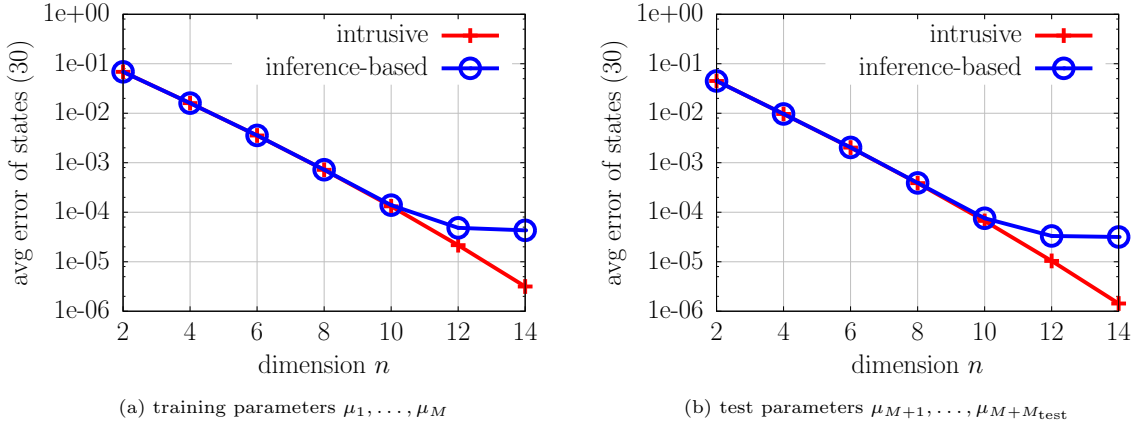


Figure 3: Burgers' equation: The inference-based reduced model shows a similar behavior to the intrusive reduced model. For dimensions larger than 10, our stability check detects that the inference-based reduced models become asymptotically unstable for a few parameters and decrements the dimension until an asymptotically stable inference-based reduced model is recovered, which leads to a leveling off of the error.

$\mathbb{R}^{K \times N}$ corresponding to the M' input matrices are computed with the full model for each parameter μ_i . We then define the trajectory and the input matrix

$$\mathbf{X}(\mu_i) = \begin{bmatrix} \mathbf{X}_1(\mu_i) \\ \vdots \\ \mathbf{X}_{M'}(\mu_i) \end{bmatrix} \in \mathbb{R}^{M'K \times N}, \quad \mathbf{U}(\mu_i) = \begin{bmatrix} \mathbf{U}_1(\mu_i) \\ \vdots \\ \mathbf{U}_{M'}(\mu_i) \end{bmatrix} \in \mathbb{R}^{M'K \times p} \quad (34)$$

for parameter μ_i and $i = 1, \dots, M$ following Section 3.6. Combining trajectories as in (34) is necessary because the data matrix corresponding to a single input matrix $\mathbf{U}_j(\mu_i) \in \mathbb{R}^{K \times 1}$ for $i = 1, \dots, M$ and $j = 1, \dots, M'$ has a large condition number in this example. Note that the number $M' = 10$ was determined empirically by monitoring the condition number of the data matrix as described in Section 3.6.

A POD basis with $w = 15$ POD basis vectors is constructed, which leads to a projection error (29) of about 10^{-5} . The trajectories and input matrices (34) are used to infer the operators for parameters μ_1, \dots, μ_M with Algorithm 1. The inference-based reduced models of dimension $n < w$ are obtained by extracting the submatrices of respective size from the inferred operators, see Section 4.1. For comparison, the intrusive reduced operators are computed, and the corresponding intrusive reduced models are constructed. The operators for a parameter $\mu \in \mathcal{D}$ unequal to μ_1, \dots, μ_M are obtained via elementwise spline interpolation.

The average error of the states (30) of the inference-based and the intrusive reduced model is shown in Figure 3a for the training parameters μ_1, \dots, μ_M and for $M_{\text{test}} = 5$ randomly drawn test parameters $\mu_{M+1}, \dots, \mu_{M+M_{\text{test}}} \in \mathcal{D}$ from a uniform distribution in \mathcal{D} in Figure 3b. The inputs impose the Dirichlet boundary conditions $x(0, t; \mu) = 1$ and $x(1, t; \mu) = -1$ for $t \in [0, T]$, i.e., input $u(t) = 1$ for $t \in [0, T]$. The inference-based reduced model shows a similar behavior to the intrusive reduced model for this nonlinear example. For a few parameters, the inference-based reduced models become asymptotically unstable for dimensions $n > 10$ in this example, because the condition number of the data matrix is large and therefore introduces numerical errors at components corresponding to the less important POD basis vectors. Such a situation can be detected with the stability check discussed in Section 3.6. If we encounter an asymptotically unstable inference-based reduced model of dimension n for a parameter μ , the dimension n is decremented until an asymptotically stable model is obtained, i.e., until components corresponding to only the important POD basis vectors remain. This leads to the leveling off of the error as seen in Figure 3.

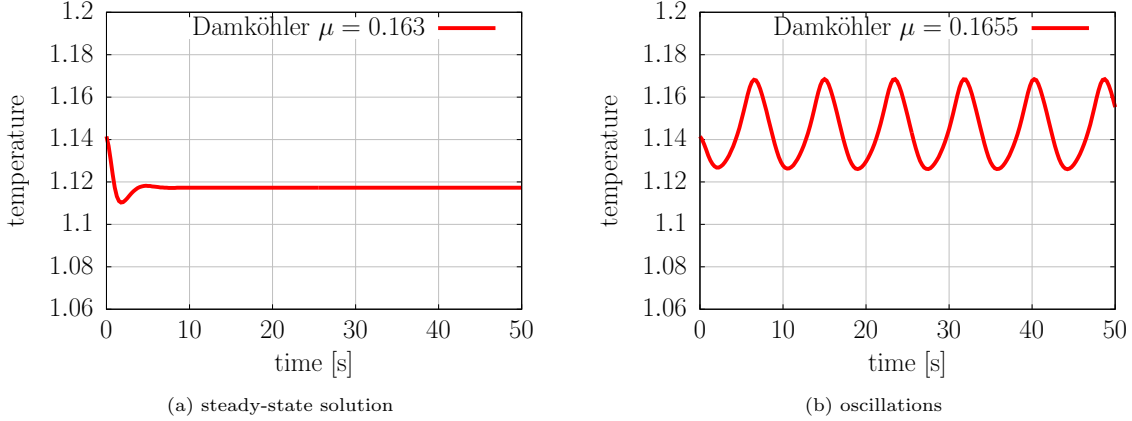


Figure 4: Tubular reactor: For parameters $\mu < \mu^*$ below the critical parameter $\mu^* \in \mathcal{D}$, the tubular reactor converges to a steady-state solution as shown in (a). For parameters $\mu > \mu^*$, the tubular reactor enters an LCO as in (b).

4.3. Limit cycle oscillations in a tubular reactor

In [41] a model of a one-dimensional non-adiabatic tubular reactor with a single reaction and axial mixing is introduced. Let $\Omega = [0, 1] \subset \mathbb{R}$ be the spatial domain of the tubular reactor, $[0, T]$ with $0 < T$ the time domain, and $\mathcal{D} = [0.163, 0.1655] \subset \mathbb{R}$ the parameter domain. The governing equations of the tubular reactor are coupled nonlinear time-dependent convection-diffusion-reaction equations

$$\begin{aligned} \frac{\partial}{\partial t} x^c(\omega, t; \mu) &= \frac{1}{Pe} \frac{\partial^2}{\partial \omega^2} x^c(\omega, t; \mu) - \frac{\partial}{\partial \omega} x^c(\omega, t; \mu) - \mu f(x^c, x^\theta) \\ \frac{\partial}{\partial t} x^\theta(\omega, t; \mu) &= \frac{1}{Pe} \frac{\partial^2}{\partial \omega^2} x^\theta(\omega, t; \mu) - \frac{\partial}{\partial \omega} x^\theta(\omega, t; \mu) - \beta(x^\theta(\omega, t; \mu) - \theta_0) + \epsilon \mu f(x^c, x^\theta), \end{aligned}$$

with the concentration $x^c : \Omega \times [0, T] \times \mathcal{D} \rightarrow \mathbb{R}$, the temperature $x^\theta : \Omega \times [0, T] \times \mathcal{D} \rightarrow \mathbb{R}$, and the nonlinear function

$$f(x^c, x^\theta) = x^c \exp\left(\gamma - \frac{\gamma}{x^\theta}\right). \quad (35)$$

The Péclet number is $Pe = 5$ and $\gamma = 25, \beta = 2.5, \epsilon = 0.5$, and $\theta_0 = 1$ are known constants. The parameter $\mu \in \mathcal{D}$ is the Damköhler number. Robin boundary conditions are imposed at $0 = \omega \in \Omega$ and Neumann boundary conditions at $\omega = 1$. The model does not have an output. The same initial condition $x_0^c : \Omega \times \mathcal{D} \rightarrow \mathbb{R}$ and $x_0^\theta : \Omega \times \mathcal{D} \rightarrow \mathbb{R}$ as in [41, 52] is used in the following. We approximate the nonlinear function f by its Taylor expansion of second order at the initial condition x_0^θ , i.e., we replace f by

$$\hat{f}(x^c, x^\theta) = x^c \exp\left(\gamma - \frac{\gamma}{x_0^\theta}\right) \left(1 - \frac{\gamma}{x_0^\theta} + \frac{\gamma(\gamma - 2x_0^\theta)}{2(x_0^\theta)^2} + \left(\frac{\gamma}{(x_0^\theta)^2} - \frac{\gamma(\gamma - 2x_0^\theta)}{(x_0^\theta)^3}\right) x^\theta + \frac{1}{2} \frac{\gamma(\gamma - 2x_0^\theta)}{(x_0^\theta)^2} (x^\theta)^2\right). \quad (36)$$

As investigated in [41], the tubular reactor exhibits two different regimes, depending on the Damköhler number $\mu \in \mathcal{D}$. There exists a critical Damköhler number $\mu^* \in \mathcal{D}$ such that for all $\mu < \mu^*$ the concentration x^c and the temperature x^θ converge to steady-state solutions, see Figure 4a. For all $\mu > \mu^*$, the tubular reactor enters an LCO as shown in Figure 4b. We define the LCO amplitude $\vartheta \in \mathbb{R}$ as the amplitude of the temperature oscillation around an equilibrium position. Thus, in the steady-state regime, the LCO amplitude is zero $\vartheta = 0$ but for $\mu > \mu^*$, the LCO amplitude is larger than zero $\vartheta > 0$. The goal is to find the critical parameter μ^* that separates the two regimes.

A full model of the tubular reactor model is derived in [52],

$$\frac{d}{dt} \mathbf{x}(t; \mu) = \mathbf{A}(\mu) \mathbf{x}(t; \mu) + \mathbf{F}(\mu) \mathbf{x}(t; \mu)^2 + \mathbf{G}(\mu) \mathbf{x}(t; \mu)^3 + \mathbf{B}(\mu) u(t),$$

where $\mathbf{G}(\mu)$ and $\mathbf{x}(t; \mu)^3$ correspond to the nonlinear term of third order and are derived similarly as for the second-order nonlinear term, see Section 2.1. A third-order nonlinear term is obtained because of the Taylor expansion (36) of the nonlinear function f . The input is constant $u(t) = 1$. The dimension of the full model is $N = 51$. We set $\delta t = 10^{-4}$, $T = 50$, and use the implicit Euler scheme with the Newton method to compute the trajectories

$$\mathbf{X}_1(\mu_1), \dots, \mathbf{X}_1(\mu_M) \in \mathbb{R}^{K \times N} \quad (37)$$

corresponding to the $M = 15$ parameters μ_1, \dots, μ_M that are equidistantly distributed in the parameter domain \mathcal{D} .

We follow Section 3.6 and combine trajectories corresponding to different initial conditions and constant input 1. Let therefore $\mathbf{V}_n \in \mathbb{R}^{N \times n}$ be the POD basis matrix derived with $n = 5$, define $M' = 3000$, and let $\mathbf{z}_2(\mu_i), \dots, \mathbf{z}_{M'}(\mu_i) \in \mathbb{R}^n$ be n -dimensional vectors with components drawn independently from a uniform distribution in $[0, 1] \subset \mathbb{R}$. The initial conditions x_0^c and x_0^θ given above are in the range of about $[0, 1]$ and therefore the range of the random initial conditions is chosen as $[0, 1]$ as well. We then compute the trajectories $\mathbf{X}_j(\mu_i) \in \mathbb{R}^{1 \times N}$ with the initial condition $\mathbf{V}_n \mathbf{z}_j(\mu_i)$ for $j = 2, \dots, M'$ and $i = 1, \dots, M$.

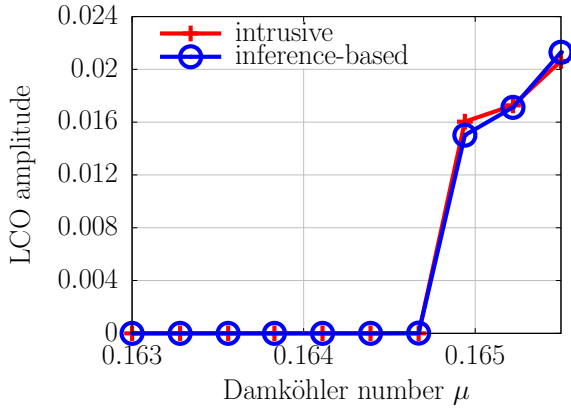
In contrast to the initial conditions x_0^c and x_0^θ given above, these random initial conditions $\mathbf{V}_n \mathbf{z}_j(\mu_i)$ do not necessarily have a physical interpretation. The aim is to obtain trajectories that reflect the behavior of the full model over many different initial conditions, and therefore provide more complete information about the full model. We use the obtained trajectories only for the operator inference. We perform only a single time step. In our experiments, this is sufficient because we are only interested in the transient solution that covers the behavior of the full model near the random initial conditions, whereas steady-state solutions are independent of the initial condition and therefore would not provide additional information about the full model, see Section 3.6. Note that we keep the input constant over time, cf. Section 4.2. Varying the input and the initial conditions might further reduce the condition number of the data matrix. Note further that we can afford a large number $M' = 3000$ of different initial conditions because we perform only a single time step. The number $M' = 3000$ was determined empirically by monitoring the condition number of the data matrix, see Section 3.6.

The trajectories $\mathbf{X}_1(\mu_i), \dots, \mathbf{X}_{M'}(\mu_i)$ are combined into \mathbf{X} as in (25) and passed to Algorithm 1 to infer the operators. Note that the matrices defined in (25) are generalized here to contain trajectories with different numbers of time steps. The dimension is set to $n = 5$. For comparison, we also construct the intrusive reduced model. The stability check of Section 3.6 indicates that the reduced models for $n = 5$ are asymptotically stable in this example. Operators for parameters $\mu \in \mathcal{D}$ are derived via elementwise spline interpolation. Note that the LCO behavior of the full model is not captured by reduced models with $n < 5$ dimensions.

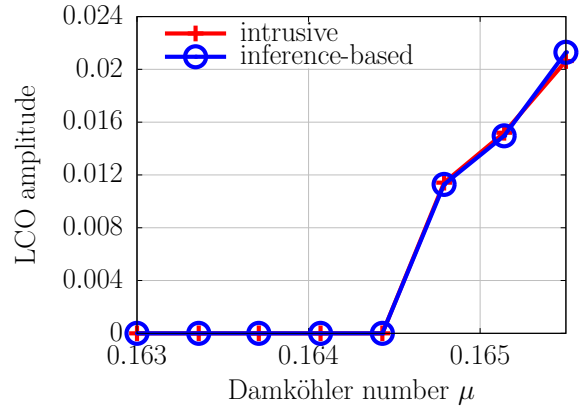
Figure 5a shows the LCO amplitude obtained with the inference-based and the intrusive reduced model. Both reduced models predict a similar critical parameter. Figure 5b plots the LCO amplitude corresponding to $M_{\text{test}} = 8$ test parameters $\mu_{M+1}, \dots, \mu_{M+M_{\text{test}}} \in \mathcal{D}$ that are equidistantly distributed in \mathcal{D} . The inference-based and the intrusive reduced models predict a similar LCO amplitude. Figure 6a reports the absolute errors

$$|\vartheta_i - \bar{\vartheta}_i|, \quad i = 1, \dots, M, \quad (38)$$

where $\vartheta_i \in \mathbb{R}$ is the LCO amplitude computed with the full model for the training parameter μ_i and $\bar{\vartheta}_i$ the corresponding LCO amplitude computed with the inference-based and intrusive reduced model, respectively. The inference-based and the intrusive reduced model lead to similar errors. Figure 6b shows that the inference-based and the intrusive reduced model predict the LCO amplitude for the test parameters with a comparable accuracy as for the training parameters. Note that the purpose of this tubular reactor model is to distinguish between the steady-state and the LCO regime, see above. Figure 5 and Figure 6 confirm that the (intrusive and inference-based) reduced models are sufficiently accurate to clearly distinguish between the two regimes.

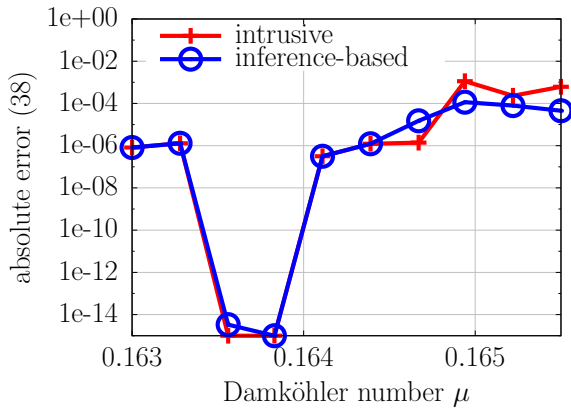


(a) training parameters μ_1, \dots, μ_M

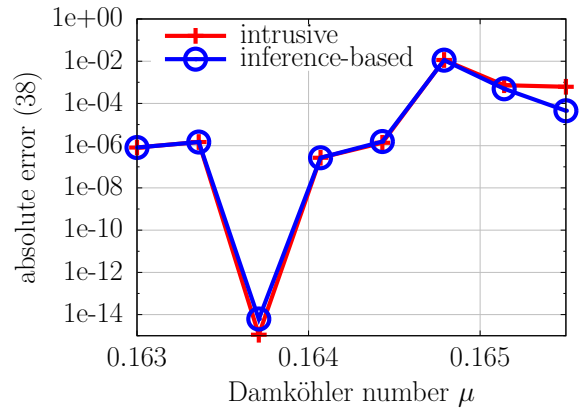


(b) test parameters $\mu_{M+1}, \dots, \mu_{M+M_{\text{test}}}$

Figure 5: Tubular reactor: The inference-based reduced model enters an LCO at the same parameter where the intrusive reduced model enters the LCO. The LCO amplitude corresponding to the inference-based reduced model shows a similar behavior as for the intrusive reduced model.



(a) training parameters μ_1, \dots, μ_M



(b) test parameters $\mu_{M+1}, \dots, \mu_{M+M_{\text{test}}}$

Figure 6: Tubular reactor: The LCO amplitudes computed with the inference-based and the intrusive reduced model lead to a similar absolute error (38) with respect to the LCO amplitude obtained with the full model.

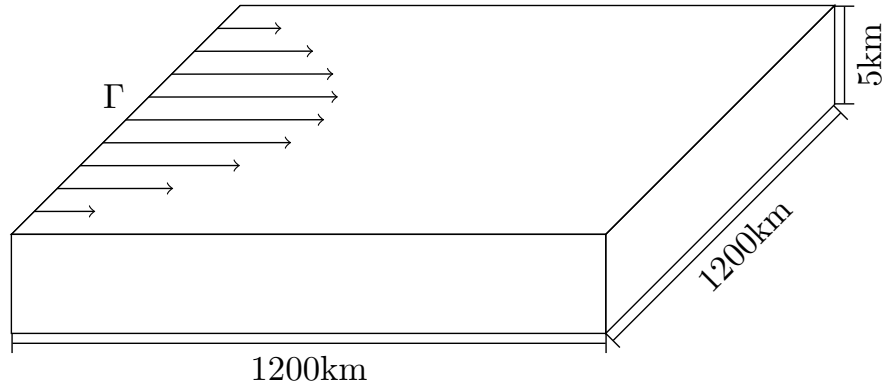


Figure 7: Global circulation model: The ocean gyre circulation experiment considers a fluid in a $1200\text{km} \times 1200\text{km} \times 5\text{km}$ box that is set in motion by a sinusoidal wind stress at boundary Γ . The result of the experiment is the velocity of the flow field and the ocean surface elevation.

4.4. General circulation model

The GCM¹ is a model to solve the equations of motion governing the ocean and Earth’s atmosphere for the numerical study of climate [6]. It is widely used as demonstrated by hundreds of published articles that involve the GCM in some way². The GCM is a typical example of a community code where the source code is available, but where the code has reached such a complexity that understanding the implementation details is tedious and time consuming. Deriving a reduced model with classical intrusive model reduction would therefore require a significant amount of time. We treat the GCM as a black box and derive a reduced model with operator inference.

We use the GCM to investigate a wind-forced ocean gyre circulation experiment as described in detail in [53]. We therefore consider a fluid in a box of size $1200\text{km} \times 1200\text{km} \times 5\text{km}$. The fluid is set in motion by a wind stress that varies sinusoidally in one direction, see Figure 7.

The following problem setup is derived from the documentation of the GCM. Let $\Omega \in [0, 1]^2 \subset \mathbb{R}^2$ be the spatial domain. Note that only a single layer of the fluid is considered and therefore a two-dimensional spatial domain is sufficient [53]. The wind stress $\tau : \Omega \times [0, T] \rightarrow \mathbb{R}$ is defined as

$$\tau(\boldsymbol{\omega}, t) = u(t) \sin\left(\pi \frac{\omega_2}{1200}\right),$$

where $\boldsymbol{\omega} = [\omega_1, \omega_2]^T \in \Omega$ is the spatial coordinate, $t \in [0, T]$ time, and $u : [0, T] \rightarrow \mathcal{U}$ the input with input domain $\mathcal{U} = [0.100, 0.099] \subset \mathbb{R}$. Note that the problem is sensitive with respect to the input and therefore a larger input domain leads to solutions with distinctly different behaviors, which are hard to approximate with a reduced model. The GCM computes the velocities of the fluid in the two spatial directions and the ocean surface elevation. The GCM documentation proposes a spatial discretization of 60×60 , which leads to a system of $N = 3 \times 60 \times 60 = 10800$ linear ODEs. The system of ODEs is solved by the GCM with a time step size of $\delta t = 1200\text{s}$, which guarantees stability. The initial condition is zero. The experiment does not have a parameter or an output. The governing equations, implementation details, time stepping scheme, and the discretized operators are unavailable.

Consider an equidistant grid of six grid points $\gamma_1, \dots, \gamma_6 \in \mathcal{U}$ in the input domain \mathcal{U} and define the inputs $u_1, \dots, u_6 : [0, T] \rightarrow \mathcal{U}$ with $u_i(t) = \gamma_i$ for $i = 1, \dots, 6$ and $t \in [0, T]$. Let further $\mathbf{U}_1, \dots, \mathbf{U}_6 \in \mathbb{R}^{K \times 1}$ be the corresponding input matrices for $T = 1.2 \times 10^6\text{s}$ with $\mathbf{U}_i = [u_i(t_1), \dots, u_i(t_K)] = [\gamma_i, \dots, \gamma_i]^T \in \mathbb{R}^{K \times 1}$ and thus $K = 10^4$. We generate the six trajectories $\mathbf{X}_1, \dots, \mathbf{X}_6 \in \mathbb{R}^{K \times N}$ with the GCM from the input matrices $\mathbf{U}_1, \dots, \mathbf{U}_6$. We then derive a POD basis of dimension $w = 30$ from these six trajectories, which leads to

¹<http://mitgcm.org/>

²<http://mitgcm.org/publications/>

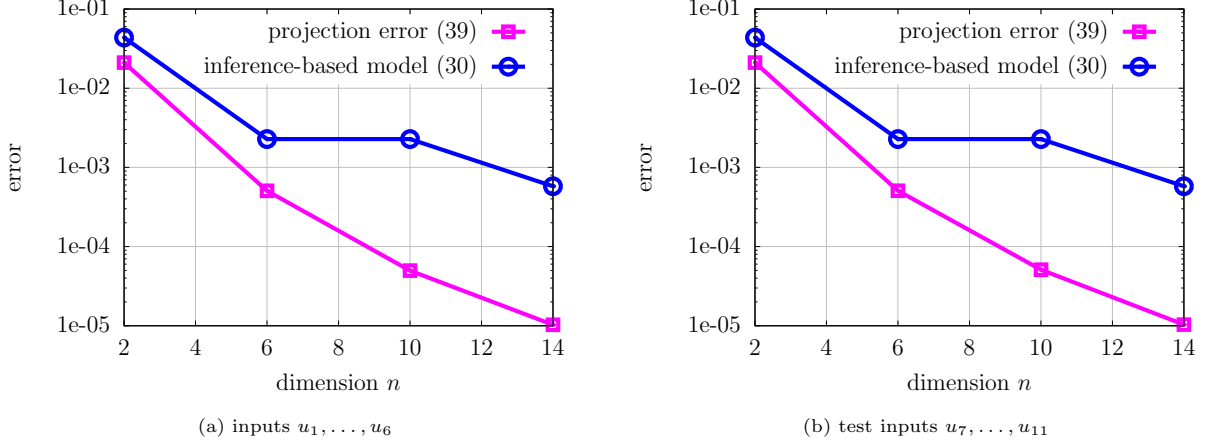


Figure 8: Global circulation model: An intrusive reduced model is unavailable for this experiment and therefore the averaged error (30) of the inference-based reduced model is compared to the projection error (39) of the POD basis. The averaged error of the inference-based reduced model decays with a similar rate as the projection error.

a projection error of about 10^{-7} . The inferred operators $\hat{\mathbf{A}} \in \mathbb{R}^{w \times w}$ and $\hat{\mathbf{B}} \in \mathbb{R}^{w \times 1}$ are constructed with Algorithm 1 from the trajectory and input matrix

$$\mathbf{X} = \begin{bmatrix} \mathbf{X}_1 \\ \vdots \\ \mathbf{X}_6 \end{bmatrix} \in \mathbb{R}^{6K \times 10800}, \quad \mathbf{U} = \begin{bmatrix} \mathbf{U}_1 \\ \vdots \\ \mathbf{U}_6 \end{bmatrix} \in \mathbb{R}^{6K \times 10800}.$$

The inference-based reduced models are marched forward in time with the implicit Euler scheme and the proposed time step size $\delta t = 1200s$. The data matrix corresponding to this example has a large condition number, leading to asymptotically unstable inference-based reduced models of dimension $n > 14$, cf. Section 3.6. We therefore plot in Figure 8a only the averaged error of the states up to dimension $n = 14$. The averaged error of the states is compared to the projection error

$$\frac{1}{6} \sum_{i=1}^6 \frac{\|\mathbf{X}_i - \mathbf{X}_i \mathbf{V}_n \mathbf{V}_n^T\|_F^2}{\|\mathbf{X}_i\|_F^2}, \quad (39)$$

because an intrusive reduced model is unavailable for this experiment. The averaged error of the states of the inference-based reduced model shows a similar behavior as the projection error. No improvement is achieved from step $n = 6$ to $n = 10$ because the operator for $n = 10$ would lead to an asymptotically unstable system and therefore the operator of dimension $n = 6$ is used, cf. the stability check in Section 3.6 and Section 4.2. Figure 8b shows the averaged error of the states of the inference-based reduced model for test inputs $u_7, \dots, u_{11} : [0, T] \rightarrow \mathcal{U}$ with $u_i(t) = (u_{i-6}(t) + u_{i-5}(t))/2$ for $i = 7, \dots, 11$ and $t \in [0, T]$. This means that the test inputs are in between the inputs u_1, \dots, u_6 that were used for the inference of the operators. The behavior of the error on the training and the test inputs is similar. Figure 9 compares the flow fields obtained with the full model to the flow field obtained with the inference-based reduced model for $n = 14$ and input u_1 . The inference-based reduced model predicts a similar flow field as the full model, which confirms the accuracy results reported in Figure 8.

5. Conclusions

We introduced operator inference that leads to a nonintrusive approach for the construction of projection-based reduced models. Our method derives approximations of the reduced operators directly from initial

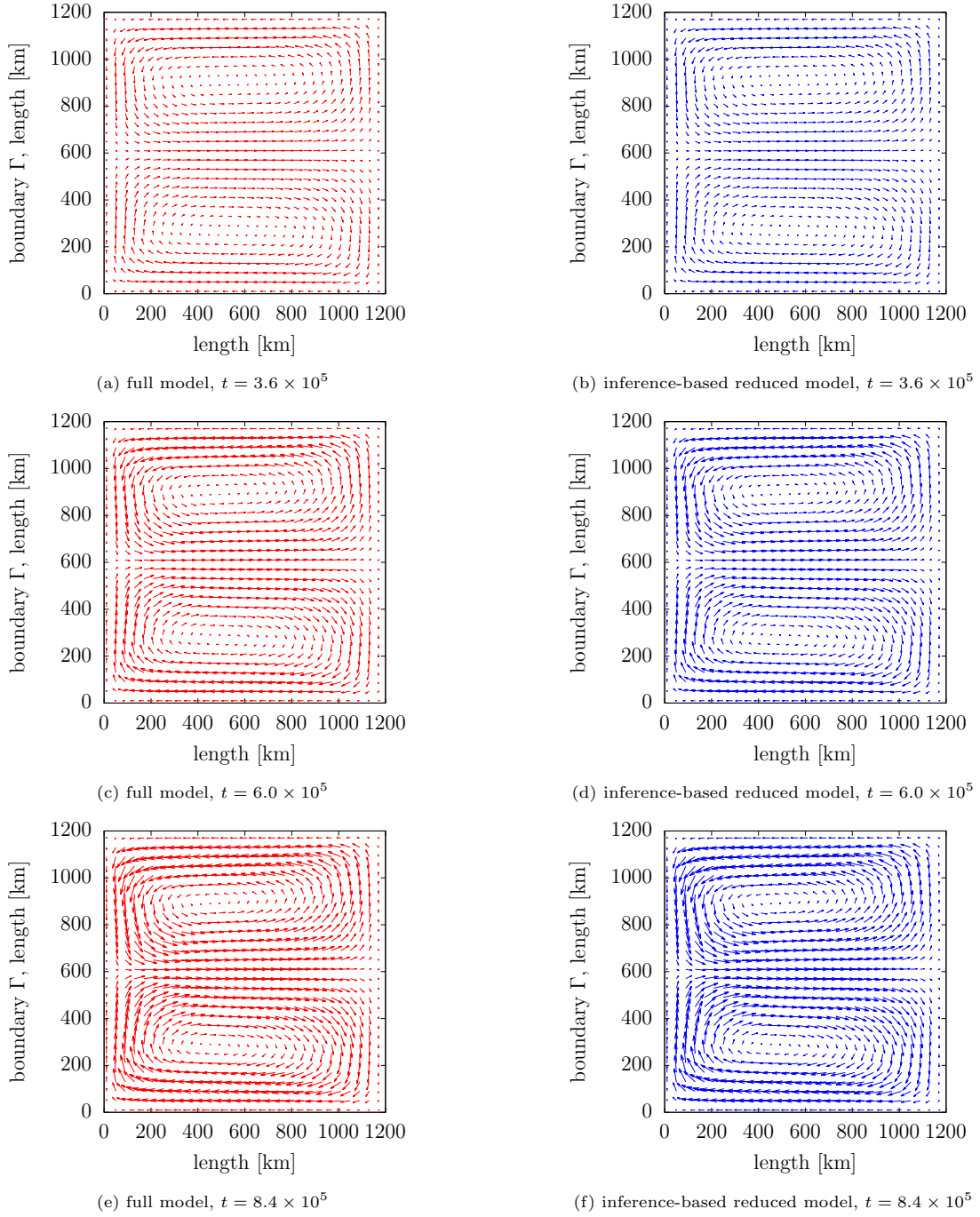


Figure 9: Global circulation model: The experiment considers a fluid in a $1200\text{km} \times 1200\text{km} \times 5\text{km}$ box, which is set in motion by a wind stress at the left boundary Γ of the box. Shown is the top view of the box. The flow field predicted by the inference-based reduced model shows a similar behavior as the flow field of the GCM.

conditions, inputs, trajectories, and outputs of the full model, without an intrusive projection step that requires the full-model operators. These data are typically available for black-box full models that are marched forward in time with a time stepping scheme. The inference of the operators is based on a least-squares problem. We have shown that the inferred operators converge to the reduced operators that would be obtained via an intrusive construction. The computational costs of the inference are bounded linearly in the number of time steps and the number of degrees of freedom of the full model. If the full model has a polynomial nonlinear term, then the costs of our operator inference grow exponentially in the order of the polynomial nonlinear term.

Our numerical experiments on models with nonlinear terms of up to third order and on a large-scale community model demonstrated the wide applicability our approach. Possible further applications of our operator inference methodology are industry codes, where often intrusive model reduction is infeasible due to the complexity of the code that leads to significant time and expertise requirements for deriving a projection-based reduced model.

Acknowledgment

This work was supported in part by the United States Department of Energy, Office of Advanced Scientific Computing Research (ASCR), Applied Mathematics Program, awards DE-FG02-08ER2585 and DE-SC0009297, as part of the DiaMonD Multifaceted Mathematics Integrated Capability Center. Some of the numerical examples were computed on the computer cluster of the Munich Centre of Advanced Computing.

References

- [1] L. Sirovich, Turbulence and the dynamics of coherent structures, *Quarterly of Applied Mathematics* (1987) 561–571.
- [2] G. Berkooz, P. Holmes, J. L. Lumley, The proper orthogonal decomposition in the analysis of turbulent flows, *Annu. Rev. Fluid Mech.* 25 (1) (1993) 539–575.
- [3] G. Rozza, D. Huynh, A. Patera, Reduced Basis Approximation and a Posteriori Error Estimation for Affinely Parametrized Elliptic Coercive Partial Differential Equations, *Archives of Computational Methods in Engineering* 15 (3) (2008) 229–275.
- [4] P. Feldmann, R. Freund, Efficient linear circuit analysis by Padé approximation via the Lanczos process, *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 14 (5) (1995) 639–649.
- [5] R. W. Freund, Model reduction methods based on Krylov subspaces, *Acta Numerica* 12 (2003) 267–319.
- [6] J. Marshall, C. Hill, L. Perelman, A. Adcroft, Hydrostatic, quasi-hydrostatic, and nonhydrostatic ocean modeling, *Journal of Geophysical Research: Oceans* 102 (C3) (1997) 5733–5752.
- [7] M. Rathinam, L. Petzold, A New Look at Proper Orthogonal Decomposition, *SIAM Journal on Numerical Analysis* 41 (5) (2003) 1893–1925.
- [8] M. A. Grepl, Y. Maday, N. C. Nguyen, A. T. Patera, Efficient reduced-basis treatment of nonaffine and nonlinear partial differential equations, *ESAIM: Mathematical Modelling and Numerical Analysis* 41 (03) (2007) 575–605.
- [9] C. Prud’homme, Y. Maday, A. T. Patera, G. Turinici, D. V. Rovas, K. Veroy, L. Machiels, Reliable Real-Time Solution of Parametrized Partial Differential Equations: Reduced-Basis Output Bound Methods, *Journal of Fluids Engineering* 124 (1) (2001) 70–80.
- [10] S. Gugercin, A. Antoulas, C. Beattie, \mathcal{H}_2 Model Reduction for Large-Scale Linear Dynamical Systems, *SIAM Journal on Matrix Analysis and Applications* 30 (2) (2008) 609–638.

- [11] R. Chakir, Y. Maday, A two-grid finite-element/reduced basis scheme for the approximation of the solution of parametric dependent P.D.E, *Comptes Rendus Mathematique* 347 (2009) 435–440.
- [12] S. Lefteriu, A. Antoulas, A new approach to modeling multiport systems from frequency-domain data, *Computer-Aided Design of Integrated Circuits and Systems*, *IEEE Transactions on* 29 (1) (2010) 14–27.
- [13] A. Ionita, A. Antoulas, Case study: Parametrized reduction using reduced-basis and the Loewner framework, in: A. Quarteroni, G. Rozza (Eds.), *Reduced Order Methods for Modeling and Computational Reduction*, Vol. 9 of MS&A - Modeling, Simulation and Applications, Springer International Publishing, 2014, pp. 51–66.
- [14] A. Mayo, A. Antoulas, A framework for the solution of the generalized realization problem, *Linear Algebra and its Applications* 425 (2–3) (2007) 634–662.
- [15] A. Ionita, A. Antoulas, Data-driven parametrized model reduction in the Loewner framework, *SIAM Journal on Scientific Computing* 36 (3) (2014) A984–A1007.
- [16] A. Antoulas, A. Ionita, S. Lefteriu, On two-variable rational interpolation, *Linear Algebra and its Applications* 436 (8) (2012) 2889 – 2915.
- [17] B. Gustavsen, A. Semlyen, Rational approximation of frequency domain responses by vector fitting, *Power Delivery*, *IEEE Transactions on* 14 (3) (1999) 1052–1061.
- [18] Z. Drmač, S. Gugercin, C. Beattie, Quadrature-based vector fitting for discretized \mathcal{H}_2 approximation, *SIAM Journal on Scientific Computing* 37 (2) (2015) A625–A652.
- [19] G. Matheron, Principles of geostatistics, *Economic Geology* 58 (8) (1963) 1246–1266.
- [20] D. Gorissen, I. Couckuyt, P. Demeester, T. Dhaene, K. Crombecq, A surrogate modeling and adaptive sampling toolbox for computer based design, *J. Mach. Learn. Res.* 11 (2010) 2051–2055.
- [21] K. A. Forrester A., Sóbester A., *Engineering design via surrogate modelling: a practical guide*, Wiley, 2008.
- [22] D. Xiao, F. Fang, A. Buchan, C. Pain, I. Navon, A. Muggeridge, Non-intrusive reduced order modelling of the Navier-Stokes equations, *Computer Methods in Applied Mechanics and Engineering* 293 (2015) 522–541.
- [23] C. Audouze, F. De Vuyst, P. B. Nair, Nonintrusive reduced-order modeling of parametrized time-dependent partial differential equations, *Numerical Methods for Partial Differential Equations* 29 (5) (2013) 1587–1628.
- [24] H. V. Ly, H. T. Tran, Modeling and control of physical processes using proper orthogonal decomposition, *Mathematical and Computer Modelling* 33 (1–3) (2001) 223 – 236.
- [25] K. Lee, T. Nam, C. Perullo, D. N. Mavris, Reduced-Order Modeling of a High-Fidelity Propulsion System Simulation, *AIAA Journal* 49 (8) (2011) 1665–1682.
- [26] Jennifer Goss, Kamesh Subbarao, Inlet Shape Optimization Based on POD Model Reduction of the Euler Equations, in: 12th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, Multidisciplinary Analysis Optimization Conferences, American Institute of Aeronautics and Astronautics, 2008, pp. 1–17.
- [27] P. G. Constantine, D. F. Gleich, Y. Hou, J. Templeton, Model reduction with MapReduce-enabled tall and skinny singular value decomposition, *SIAM Journal on Scientific Computing* 36 (5) (2014) S166–S191.

- [28] R. Zimmermann, S. Görtz, Non-linear reduced order models for steady aerodynamics, *Procedia Computer Science* 1 (1) (2010) 165 – 174.
- [29] D. Butnaru, B. Peherstorfer, H. Bungartz, D. Pflüger, Fast insight into high-dimensional parametrized simulation data, in: *Machine Learning and Applications (ICMLA)*, 2012 11th International Conference on, Vol. 2, 2012, pp. 265–270.
- [30] T. Bui-Thanh, Murali Damodaran, Karen Willcox, Proper Orthogonal Decomposition Extensions for Parametric Applications in Compressible Aerodynamics, in: *21st AIAA Applied Aerodynamics Conference, Fluid Dynamics and Co-located Conferences*, American Institute of Aeronautics and Astronautics, 2003, pp. 1–11.
- [31] A. I. J. Forrester, A. J. Keane, Recent advances in surrogate-based optimization, *Progress in Aerospace Sciences* 45 (1-3) (2009) 50–79.
- [32] P. Schmid, S. J., Dynamic mode decomposition of numerical and experimental data, in: *Bull. Amer. Phys. Soc.*, 61st APS meeting, American Physical Society, 2008, p. 208.
- [33] P. Schmid, Dynamic mode decomposition of numerical and experimental data, *Journal of Fluid Mechanics* 656 (2010) 5–28.
- [34] C. Rowley, I. Mezić, S. Bagheri, P. Schlatter, D. Henningson, Spectral analysis of nonlinear flows, *Journal of Fluid Mechanics* 641 (2009) 115–127.
- [35] J. H. Tu, C. W. Rowley, D. M. Luchtenburg, S. L. Brunton, J. N. Kutz, On dynamic mode decomposition: Theory and applications, *Journal of Computational Dynamics* 1 (2) (2014) 391–421.
- [36] J. L. Proctor, S. L. Brunton, J. N. Kutz, Dynamic mode decomposition with control, *ArXiv* 1409.6358.
- [37] S. L. Brunton, B. W. Brunton, J. L. Proctor, J. N. Kutz, Koopman observable subspaces and finite linear representations of nonlinear dynamical systems for control, *ArXiv* 1510.03007.
- [38] S. L. Brunton, J. L. Proctor, J. N. Kutz, Discovering governing equations from data: Sparse identification of nonlinear dynamical systems, *ArXiv* 1509.03580.
- [39] A. Alla, J. N. Kutz, Nonlinear model reduction via dynamic mode decomposition, submitted.
- [40] B. Peherstorfer, K. Willcox, Dynamic data-driven reduced-order models, *Computer Methods in Applied Mechanics and Engineering* 291 (0) (2015) 21–41.
- [41] R. Heinemann, A. Poore, Multiplicity, stability, and oscillatory dynamics of the tubular reactor, *Chemical Engineering Science* 36 (8) (1981) 1411–1419.
- [42] R. P. Stanley, *Enumerative Combinatorics*, Cambridge University Press, 2012.
- [43] P. Benner, S. Gugercin, K. Willcox, A survey of projection-based model reduction methods for parametric dynamical systems, *SIAM Review* 57 (4) (2015) 483–531.
- [44] D. Amsallem, J. Cortial, K. Carlberg, C. Farhat, A method for interpolating on manifolds structural dynamics reduced-order models, *International Journal for Numerical Methods in Engineering* 80 (9) (2009) 1241–1258.
- [45] H. Panzer, J. Mohring, R. Eid, B. Lohmann, Parametric model order reduction by matrix interpolation, *at – Automatisierungstechnik* 58 (8) (2010) 475–484.
- [46] J. Degroote, J. Vierendeels, K. Willcox, Interpolation among reduced-order matrices to obtain parameterized models for design, optimization and probabilistic analysis, *International Journal for Numerical Methods in Fluids* 63 (2) (2010) 207–230.

- [47] D. Amsallem, C. Farhat, An online method for interpolating linear parametric reduced-order models, *SIAM Journal on Scientific Computing* 33 (5) (2011) 2169–2198.
- [48] G. H. Golub, J. Wilkinson, Note on the iterative refinement of least squares solution, *Numerische Mathematik* 9 (2) (1966) 139–148.
- [49] G. H. Golub, V. Pereyra, The differentiation of pseudo-inverses and nonlinear least squares problems whose variables separate, *SIAM Journal on Numerical Analysis* 10 (2) (1973) 413–432.
- [50] G. H. Golub, C. F. V. Loan, *Matrix Computations*, Johns Hopkins University Press, 2013.
- [51] A. Antoulas, *Approximation of Large-Scale Dynamical Systems*, SIAM, 2005.
- [52] Y. B. Zhou, Model reduction for nonlinear dynamical systems with parametric uncertainties, Thesis (S.M.), Aeronautics and Astronautics, Massachusetts Institute of Technology (2012).
- [53] W. Holland, L. Lin, On the Generation of Mesoscale Eddies and their Contribution to the Oceanic-General Circulation. I. A Preliminary Numerical Experiment, *Journal of Physical Oceanography* 5 (4) (1975) 642–657.