SIAM J. SCI. COMPUT.
Vol. 44, No. 1, pp. A128–A149

COMPUTING SENSITIVITIES IN EVOLUTIONARY SYSTEMS: A REAL-TIME REDUCED ORDER MODELING STRATEGY*

MICHAEL DONELLO†, MARK H. CARPENTER‡, AND HESSAM BABAEE†

Abstract. We present a new methodology for computing sensitivities in evolutionary systems using a model-driven low-rank approximation. To this end, we formulate a variational principle that seeks to minimize the distance between the time derivative of the reduced approximation and sensitivity dynamics. The first order optimality condition of the variational principle leads to a system of closed form evolution equations for an orthonormal basis and corresponding sensitivity coefficients. This approach allows for the computation of sensitivities with respect to a large number of parameters in an accurate and tractable manner by extracting correlations between different sensitivities on the fly. The presented method requires solving *forward* evolution equations, sidestepping the restrictions imposed by the *forward/backward* workflow of adjoint sensitivities. For example, the presented method, unlike the adjoint equation, does not impose any input/output load and can be used in applications in which real-time sensitivities are of interest. We demonstrate the utility of the method for three test cases: (1) computing sensitivity with respect to model parameters in the Rössler system, (2) computing sensitivity with respect to an infinite-dimensional forcing parameter in the chaotic Kuramoto–Sivashinsky equation, and (3) computing sensitivity with respect to reaction parameters for species transport in a turbulent reacting flow.

Key words. reduced order model, time-dependent basis, chaos, sensitivity

AMS subject classifications. 37E99, 76V05, 65P99

DOI. 10.1137/20M1388565

1. Introduction.

Sensitivity analysis is required in a diverse set of evolutionary systems that are governed by differential equations in the form of $\dot{\mathbf{v}} = \mathbf{g}(\mathbf{v}; \boldsymbol{\alpha})$, where $\dot{(\sim)} = d(\sim)/dt$, $\mathbf{v} \in \mathbb{R}^n$ is the state space variable and $\boldsymbol{\alpha} \in \mathbb{R}^d$ is the design space. These sensitivities, denoted by $\mathbf{v}_i' = \partial \mathbf{v}/\partial \alpha_i$, $i = 1, \ldots, d$, are needed in numerous applications such as gradient based optimization [1, 2], optimal control [3], grid adaptivity [4], and parameter identification [5], to name a few. The sensitivities are commonly computed via finite difference or by directly solving a forward sensitivity equation. The computational cost of using finite difference or a sensitivity equation scales linearly with the number of parameters—making them impracticable when sensitivities with respect to a large number of parameters are needed. On the other hand, an adjoint equation (AE) can be solved for computing the sensitivity of an objective function that depends on the forward sensitivity, $\mathbf{v}_i'$. The computational cost of solving an AE is independent of the number of parameters, and it requires solving a single ordinary/partial differential equation (ODE/PDE) for each objective function of interest.

While AE is certainly a preferred approach for computing sensitivities for stationary problems, for time-dependent problems, the *forward-backward* workflow of

*Submitted to the journal's Methods and Algorithms for Scientific Computing section December 28, 2020; accepted for publication (in revised form) September 1, 2021; published electronically January 18, 2022.

https://doi.org/10.1137/20M1388565

Funding: This work was supported by NASA grant no. 80NSSC18M0150. The work of the third author was supported by the NSF under grant CBET-2042918.

†Mechanical Engineering and Materials Sciences, University of Pittsburgh, Pittsburgh, PA 15261 USA (mid47@pitt.edu, h.babaee@pitt.edu).

‡Computational Modeling and Simulation Branch, NASA Langley Research Center, Hampton, VA 23681 USA (mark.h.carpenter@nasa.gov).

the adjoint solver can pose several challenges. In particular, solving an AE imposes a significant storage cost as the AE must be solved backward in time. On the other hand, the adjoint operator utilizes the forward time-resolved solution of the nonlinear dynamical system, i.e., $\mathbf{v}$. As a result, the dynamical system must be solved forward in time, and its time-resolved solution must be stored. The adjoint solver is then solved backward in time, in which the nonlinear state is read from the disk at every time step. This workflow is not adequate for problems where real-time sensitivities are required, e.g., grid adaptivity for time-dependent problems [4]. Moreover, for high-dimensional dynamical systems, i.e., $n \sim \mathcal{O}(10^{10})$, the imposed input/output (I/O) operations in the AE workflow could lead to insurmountable limitations.

The I/O limitations will become more restrictive in future high performance computing architectures, and it is one of the major challenges in the transition from current sub-petascale and petascale computing to exascale computing [6]. For example, in high fidelity simulations of turbulent reactive flows, the solution can only be stored at every 400th time step in order to maintain I/O overhead at a reasonable level, while important events such as the ignition kernel occur rapidly on the order of 10 simulation time steps [7]. Storing the time-resolved solution for these problems is required for AE, and it is currently exceeding the acceptable I/O levels; this trend continues to become even more unfavorable for exascale computing. This alone gives rise to a growing need for algorithms that can accurately compute sensitivities while minimizing or eliminating I/O requirements, and this is one of the motivations of the method presented in this paper.

Sensitivities of a dynamical system with respect to different parameters are often highly correlated, and therefore they are amenable to low-rank approximations. To this end, a new low-dimensional model was recently presented in [8] that can describe transient instabilities in high-dimensional nonlinear dynamical systems. This approach is based on a time-dependent basis known as the optimally time-dependent (OTD) modes. The evolution equation for the OTD modes is obtained by minimizing the functional

$$\mathcal{F}(\dot{\mathbf{u}}_1, \dot{\mathbf{u}}_2, \dots, \dot{\mathbf{u}}_r) = \sum_{i=1}^{r} \left\| \dot{\mathbf{u}}_i - \mathbf{L}(\mathbf{v}(t), t)\mathbf{u}_i(t) \right\|^2 \tag{1.1}$$

subject to the orthonormality of the OTD modes, i.e., $\mathbf{u}_i^T \mathbf{u}_j = \delta_{ij}$, where $\mathbf{u}_i \in \mathbb{R}^n, i = 1, \dots, r$, are the OTD modes. In the above functional, $\|\mathbf{u}\|^2 = \mathbf{u}^T\mathbf{u}$ and $\mathbf{L}(\mathbf{v}(t), t) := \nabla_{\mathbf{v}}\mathbf{g}$ is the instantaneous linearized operator. The optimality condition of the above variational principle leads to a closed form evolution equation for the OTD subspace: $\dot{\mathbf{U}} = (\mathbf{I} - \mathbf{U}\mathbf{U}^T)\mathbf{L}\mathbf{U}$, where $\mathbf{U} = [\mathbf{u}_1|\mathbf{u}_2|\dots|\mathbf{u}_r] \in \mathbb{R}^{n\times r}$ and $\mathbf{I} \in \mathbb{R}^{n\times n}$ is the identity matrix. It was shown later that the OTD subspace converges exponentially fast to the eigendirections of the Cauchy–Green tensor associated with the most intense finite-time instabilities [9]. In this sense, the OTD reduction can be interpreted as a low-rank subspace that approximates the evolution of the perturbed initial condition in all directions of the phase space. One of the computational advantages of OTD is that it only requires solving forward equations. Moreover, the computational complexity of solving OTD reduction scales linearly with respect to the number of modes. The OTD reduction has also been used for flow control [10], building precursors for bursting phenomena [11], and detection of edge manifolds in infinite-dimensional dynamical systems [12]. We also note that time-dependent bases have been developed in the context of stochastic reduced order modeling (see, for example, [13, 14, 15, 16, 17])

and for reduced order modeling of passive and reactive species transport equations [18] as well as skeletal kinetic model reduction [19].

Our objective in this paper is to approximate sensitivities with respect to a large number of parameters using forward low-rank systems similar to OTD. In particular, we seek to reduce the computational cost of solving forward sensitivity equations by exploiting the real-time *correlations* between various sensitivities. However, OTD is not adequate when applied to systems subject to perturbations in a parametric space. These perturbations are governed by the forced linear sensitivity equation, $\partial \mathbf{v}_i'/\partial t = \mathbf{L}\mathbf{v}_i' + \partial \mathbf{g}/\partial \alpha_i$, and in general, the OTD subspace is not an optimal basis for the evolution of $\mathbf{v}_i'$. To this end, we present a new approach based on a time-dependent basis for solving reduced order models (ROMs) of time-varying linear systems forced by a high-dimensional function.

The contributions of this paper are twofold: (i) We present a new variational principle, whose optimality conditions lead to forward real-time low-rank evolution equations for the approximation of the forced sensitivity equation. We name this approach "forced OTD," which we will simply refer to as f-OTD. (ii) We extend the application of the presented method to compute *tensor-like* sensitivities. An example of tensor-like sensitivities is in reactive flows where the goal is to compute the sensitivity of n_s species with respect to n_r reaction parameters. In these systems, the full sensitivities can be represented as a third order tensor, where the first dimension is the number of grid points, the second dimension represents species (n_s), and the third dimension represents the parameters (n_r). We show that with a single set of orthonormal modes, we can approximate sensitivities by exploiting correlations between *all* sensitivities. We compare the computational cost of f-OTD against adjoint based sensitivities where one adjoint variable for each species must be solved [20, 21, 22] for each objective function of interest. We show how the presented approach can be used for computing sensitivities with respect to a large number of parameters by solving forward low-rank evolution equations without the need to store the state variables. Unlike AE, the sensitivities we compute are not tied to an objective function and can be used to directly evaluate sensitivities of any derived quantity of interest.

In the sections that follow, we present the formulation of the f-OTD method and demonstrate a number of outcomes. We start in section 2 with the variational principle whose optimality conditions lead to a set of closed form evolution equations for a low-rank approximation of the forced sensitivity equation. In section 3, we present three demonstration cases: (1) sensitivity with respect to model parameters in the Rössler system, (2) sensitivity with respect to an infinite-dimensional forcing parameter in the chaotic Kuramoto–Sivashinsky equation, and (3) sensitivity with respect to reaction parameters for species transport in a turbulent reacting flow. In section 4, we present the conclusions and implications of our work.

2. Methodology.

2.1. Preliminaries.

We denote $\mathbf{u}(x,t)$ to be a time-dependent field variable. We denote the spatial domain as $D \subset \mathbb{R}^m$, where $m = 1$, 2, or 3. The spatial coordinate is denoted by $x \in D$, and the function is evaluated at time t. We introduce a quasimatrix notation to represent a set of functions in matrix form and denote the quasimatrix $\mathbf{U}(x,t) \in \mathbb{R}^{\infty \times r}$ as [23]

$$\mathbf{U}(x,t) = \Big[\, \mathbf{u}_1(x,t) \;\Big|\; \mathbf{u}_2(x,t) \;\Big|\; \dots \;\Big|\; \mathbf{u}_d(x,t) \,\Big]_{\infty \times r},$$

where the first dimension is infinite and represents the continuous state space contained by D and the second dimension is discrete. Similarly, we use the term quasitensor for tensors whose first dimension is infinity. For example, $\boldsymbol{\mathcal{T}} \in \mathbb{R}^{\infty \times r_1 \times r_2}$ is a third order quasitensor. We define the columnwise inner product of two quasimatrices $\mathbf{U}(x,t) \in \mathbb{R}^{\infty \times r}$ and $\mathbf{V}(x,t) \in \mathbb{R}^{\infty \times d}$ as

$$\mathbf{S}(t) = \big\langle \mathbf{U}(x,t), \mathbf{V}(x,t) \big\rangle,$$

where $\mathbf{S}(t) \in \mathbb{R}^{r \times d}$ is a matrix with components

$$S_{ij}(t) = \int_D \mathbf{u}_i(x,t)\mathbf{v}_j(x,t)dx,$$

where $\mathbf{u}_i(x,t)$ and $\mathbf{v}_j(x,t)$ are the ith and jth columns of $\mathbf{U}(x,t)$ and $\mathbf{V}(x,t)$, respectively. The discrete analogue of this operation is the matrix multiplication, $\mathbf{U}(t)^T\mathbf{W}\mathbf{V}(t)$, where $\mathbf{U}(t) \in \mathbb{R}^{n \times r}$ and $\mathbf{V}(t) \in \mathbb{R}^{n \times d}$ are space discrete with n grid points and $\mathbf{W} \in \mathbb{R}^{n \times n}$ is a diagonal weight matrix. For the case of single-column quasimatrices $\mathbf{u}(x,t) \in \mathbb{R}^{\infty \times 1}$ and $\mathbf{v}(x,t) \in \mathbb{R}^{\infty \times 1}$, i.e., functions, the above definition reduces to an inner product between two functions, which induces an L_2 norm:

$$\big\langle \mathbf{u}(x,t), \mathbf{v}(x,t) \big\rangle = \int_D \mathbf{u}(x,t)\mathbf{v}(x,t)dx, \qquad \|\mathbf{u}(x,t)\|_2 = \big\langle \mathbf{u}(x,t), \mathbf{u}(x,t) \big\rangle^{\frac{1}{2}}.$$

The Frobenius norm of a quasimatrix is defined as

$$\Big\| \mathbf{U}(x,t) \Big\|_F = \sqrt{\operatorname{trace} \big\langle \mathbf{U}(x,t), \mathbf{U}(x,t) \big\rangle}.$$

Finally, we define multiplication between a quasimatrix and a vector:

$$\mathbf{c}(x,t) = \mathbf{U}(x,t)\mathbf{b}(t),$$

where $\mathbf{b}(t) = (b_1(t), b_2(t), \ldots, b_r(t))^T \in \mathbb{R}^{r \times 1}$ is an arbitrary vector and $\mathbf{c}(x,t) \in \mathbb{R}^{\infty \times 1}$ is a function given by $\mathbf{c}(x,t) = b_i(t)\mathbf{u}_i(x,t)$. We use index notation, and the same indices imply summation.

We consider the nonlinear PDE for the evolution of $\mathbf{v}(x,t)$:

$$\frac{\partial \mathbf{v}(x,t)}{\partial t} = \mathcal{N}\left((\mathbf{v}(x,t); \boldsymbol{\alpha}\right), \quad t \in [0, T_f], \tag{2.1}$$

where $\mathcal{N}$ is in general a nonlinear differential operator. Our goal is to compute the sensitivity of $\mathbf{v}(x,t)$ with respect to the design parameters $\boldsymbol{\alpha}$, which can be either infinite-dimensional, i.e., a function $\boldsymbol{\alpha} = \boldsymbol{\alpha}(x,t)$, or finite-dimensional, i.e., a vector $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_d)$. For the sake of simplicity in the exposition we consider the finite-dimensional parametric space. Differentiating (2.1) with respect to design parameter α_i leads to an evolution equation for the sensitivity of the dynamical system:

$$\frac{\partial \mathbf{v}_i'(x,t)}{\partial t} = \mathcal{L}\left(\mathbf{v}_i'(x,t)\right) + \mathbf{f}_i'(x,t;\boldsymbol{\alpha}), \tag{2.2}$$

where $\mathbf{v}_i' = \partial \mathbf{v}/\partial \alpha_i$ is the sensitivity of $\mathbf{v}(x,t)$ with respect to α_i, $\mathcal{L}(\sim) = \partial \mathcal{N}/\partial \mathbf{v}(\sim)$ is the linearized operator, and $\mathbf{f}_i' = \partial \mathcal{N}/\partial \alpha_i$ is the forcing term.

2.2. Variational principle for reduced order modeling. Different sensitivities in a dynamical system tend to be highly correlated at any given time, and therefore, these sensitivities can potentially be approximated effectively by a low-rank time-dependent subspace. In this section, we present a real-time reduced order modeling strategy that aims to extract this subspace and utilize it for building sensitivity ROMs. In particular, we present a variational principle, whose first order optimality conditions lead to the evolution equations of a time-dependent subspace and its coefficients. We estimate the sensitivities using the low-rank decomposition:

$$\mathbf{V}'(x,t) = \mathbf{U}(x,t)\mathbf{Y}(t)^T + \mathbf{E}(x,t), \tag{2.3}$$

where $\mathbf{V}'(x,t) = [\mathbf{v}_1'(x,t) \mid \mathbf{v}_2'(x,t) \mid \ \dots \ \mid \mathbf{v}_d'(x,t)]_{\infty\times d}$ is the sensitivities quasimatrix, $\mathbf{U}(x,t) = [\mathbf{u}_1(x,t) \mid \mathbf{u}_2(x,t) \mid \ \dots \ \mid \mathbf{u}_r(x,t)]_{\infty\times r}$ is a quasimatrix representing a rank-r time-dependent orthonormal basis in which $\langle \mathbf{u}_i(x,t), \mathbf{u}_j(x,t)\rangle = \delta_{ij}$, $\mathbf{Y}(t) = [\mathbf{y}_1(t) \mid \mathbf{y}_2(t) \mid \ \dots \ \mid \mathbf{y}_r(t)]_{d\times r}$ is the coefficient matrix, and $\mathbf{E}(x,t) \in \mathbb{R}^{\infty\times d}$ is the approximation error. The f-OTD decomposition is shown schematically in Figure 1.

We formulate a variational principle with control parameters $\dot{\mathbf{U}}(x,t)$ and $\dot{\mathbf{Y}}(t)$ that seeks to optimally update the subspace $\mathbf{U}(x,t)$ and its coefficients $\mathbf{Y}(t)$ by minimizing the distance between the time derivative of the low-rank approximation and the full-dimensional sensitivity dynamics:

$$\mathcal{F}(\dot{\mathbf{U}}(x,t), \dot{\mathbf{Y}}(t)) = \left\| \frac{\partial(\mathbf{U}(x,t)\mathbf{Y}(t)^T)}{\partial t} - \mathcal{L}\left(\mathbf{U}(x,t)\mathbf{Y}(t)^T\right) - \mathbf{F}'(x,t;\boldsymbol{\alpha}) \right\|_F^2, \tag{2.4}$$

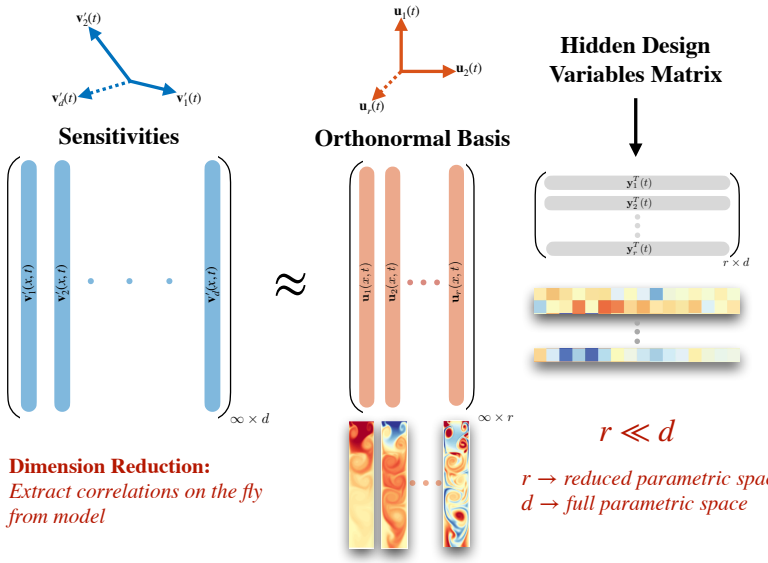


FIG. 1. *Overview of the reduced order modeling strategy. Shown on left in blue is the full-dimensional system of sensitivities that we seek to model using the f-OTD low-rank approximation. Shown on right is the low-rank approximation which consists of a set of temporally evolving orthonormal modes (red) and hidden design variables (gray). The hidden design variables are coefficients that map the orthonormal basis to each sensitivity in the full-dimensional system. That is, each of the d sensitivities are approximated as a linear combination of the r orthonormal modes, where $r \ll d$. It is important to note that the orthonormal basis and hidden design variables are model-driven and evolve based on the linear sensitivity dynamics. Thus, the proposed method only requires solving a system of r PDEs and r ODEs for the modes and coefficients, respectively.*

where $\mathbf{F}'(x,t) = \left[\mathbf{f}'_1(x,t) \mid \mathbf{f}'_2(x,t) \mid \ldots \mid \mathbf{f}'_d(x,t)\right]_{\infty\times d}$. Taking the time derivative of the orthonormality condition leads to the following constraint for the minimization problem:

$$\left\langle \dot{\mathbf{u}}_i(x,t), \mathbf{u}_j(x,t)\right\rangle + \left\langle \mathbf{u}_i(x,t), \dot{\mathbf{u}}_j(x,t)\right\rangle = 0. \tag{2.5}$$

We denote $\boldsymbol{\phi}_{ij}(t) = \left\langle \mathbf{u}_i(x,t), \dot{\mathbf{u}}_j(x,t)\right\rangle$, in which $\boldsymbol{\Phi}(t) = [\phi_{ij}(t)] \in \mathbb{R}^{r\times r}$. It is easy to see that $\boldsymbol{\Phi}(t)$ must be a skew-symmetric matrix in order to satisfy (2.5), i.e., $\boldsymbol{\phi}_{ji}(t) = -\boldsymbol{\phi}_{ij}(t)$. Incorporating this constraint leads to the following unconstrained optimization problem functional:

$$\begin{aligned}\mathcal{G}(\dot{\mathbf{U}}(x,t), \dot{\mathbf{Y}}(t), \lambda(t)) = &\left\| \frac{\partial(\mathbf{U}(x,t)\mathbf{Y}(t)^T)}{\partial t} - \mathcal{L}\left(\mathbf{U}(x,t)\right)\mathbf{Y}(t)^T - \mathbf{F}'(x,t;\boldsymbol{\alpha})\right\|_F^2 \\ &+ \sum_{i,j=1}^{r} \lambda_{ij}(t)\big(\left\langle \mathbf{u}_i(x,t), \dot{\mathbf{u}}_j(x,t)\right\rangle - \boldsymbol{\phi}_{ij}(t)\big),\end{aligned} \tag{2.6}$$

where $\lambda(t) = [\lambda_{ij}(t)] \in \mathbb{R}^{r\times r}$ are Lagrange multipliers. To derive the optimality conditions, we follow a procedure similar to the one that was recently presented in [24]. In Appendix A, we show that minimizing the above functional with respect to $\dot{\mathbf{U}}(x,t)$ and $\dot{\mathbf{Y}}(t)$ leads to closed form evolution equations for the modes and corresponding sensitivity coefficients (ROM):

$$\frac{\partial \mathbf{u}_i(x,t)}{\partial t} = \mathcal{L}\left(\mathbf{u}_i\right) - \left\langle \mathbf{u}_j, \mathcal{L}\left(\mathbf{u}_i\right)\right\rangle \mathbf{u}_j + \left[\mathbf{F}'\mathbf{y}_k - \left\langle \mathbf{u}_j, \mathbf{F}'\mathbf{y}_k\right\rangle \mathbf{u}_j\right] C_{ik}^{-1} - \phi_{ij}\mathbf{u}_j, \tag{2.7}$$

$$\frac{d\mathbf{y}_i(t)}{dt} = \left\langle \mathbf{u}_i, \mathcal{L}(\mathbf{u}_j)\right\rangle \mathbf{y}_j + \left\langle \mathbf{F}', \mathbf{u}_i\right\rangle - \phi_{ij}\mathbf{y}_j, \tag{2.8}$$

where $\mathbf{C}(t) = [C_{ik}(t)] \in \mathbb{R}^{r\times r}$ is the low-rank *correlation* matrix, in which $C_{ik}(t) = \mathbf{y}_i(t)^T\mathbf{y}_k(t)$. These equations are initialized by solving (2.2) for a single time step and computing the singular value decomposition (SVD) of $\mathbf{V}'(x, t = \Delta t)$ such that $\mathbf{U}(x, t = \Delta t)$ contains the first r left singular vectors and $\mathbf{Y}(t = \Delta t)$ is the matrix multiplication of the first r right singular vectors and singular values; see section 2.4. We show in section 2.3 that the skew-symmetric matrix ϕ_{ij} can be taken to be zero, i.e., $\phi_{ij} = 0$.

In the following, we make several observations about (2.7) and (2.8): (i) (2.7) determines the evolution of the f-OTD subspace. For $\phi_{ij} = 0$, the right-hand side of (2.7) is equal to the projection of $\mathcal{L}\left(\mathbf{U}\right) + \mathbf{F}\mathbf{Y}\mathbf{C}^{-1}$ onto the complement of the space spanned by $\mathbf{U}$. Therefore, if $\mathcal{L}\left(\mathbf{U}\right) + \mathbf{F}\mathbf{Y}\mathbf{C}^{-1}$ is in the span of $\mathbf{U}$, the f-OTD subspace does not evolve, i.e., $\dot{\mathbf{U}} = \mathbf{0}$. However, when $\mathcal{L}\left(\mathbf{U}\right) + \mathbf{F}\mathbf{Y}\mathbf{C}^{-1}$ is not in the span of $\mathbf{U}$, the f-OTD subspace evolves optimally to follow the right-hand side. Equation (2.8) is the f-OTD ROM that determines the evolution of the sensitivities within the f-OTD subspace. (ii) We observe that if we set $\mathbf{F}'(x,t) = \mathbf{0}$ in the above equations, we recover the OTD evolution equations presented in [8]. However, unlike the OTD equations, where the evolution of the OTD modes are independent of the evolution of the coefficients ($\mathbf{Y}$), there is a two-way nonlinear coupling between the f-OTD evolution equations for $\mathbf{U}$ and $\mathbf{Y}$. (iii) From the above equations, it is clear to see that f-OTD extracts the low-rank approximation directly from the sensitivity evolution equation. In that sense, it is different from data-driven low-rank approximations such as proper orthogonal decomposition [25, 26, 27] or dynamic mode decomposition

[28, 29], in which the low-rank subspace is extracted from preexisting data. The need to generate data simply does not exist in the f-OTD workflow. (iv) The computational cost of solving the f-OTD (2.7) and (2.8) is roughly equivalent to that of solving r forward sensitivity equations. This is because the evolution of the f-OTD modes described by (2.7) inherits the same differential operators from the sensitivity equation. In fact, (2.7) can be formulated as a forced linear system $\partial \mathbf{u}_i/\partial t = \mathcal{L}(\mathbf{u}_i) + \mathbf{g}_i$. Assuming the discrete f-OTD modes have the size of $\mathbf{U} \in \mathbb{R}^{n\times r}$ and also assuming that $\mathcal{L}$ represents differential operators that can be represented discretely with a matrix of size $n \times n$, for implicit time integration, the cost of solving a linear system often exceeds that of computing $\mathbf{g}_i$. Evaluating $\mathbf{g}_i$ involves computing (i) the low-rank matrix $\mathbf{L}_{r_{ij}} = \langle \mathbf{u}_j, \mathcal{L}(\mathbf{u}_i)\rangle$, which has the computational complexity of $\mathcal{O}(r^2 n)$, when $\mathcal{L}$ is sparse and $\mathcal{O}(r^2 n^2)$ when $\mathcal{L}$ is a full matrix; (ii) $\langle \mathbf{u}_j, \mathbf{F}'\mathbf{y}_k\rangle = \langle \mathbf{u}_j, \mathbf{F}'\rangle \mathbf{y}_k$ which has computational complexity $\mathcal{O}(nrd + dr^2)$, and (iii) the correlation matrix inversion $\mathbf{C}^{-1}$ which has computational complexity $\mathcal{O}(r^3)$. Since r is often much smaller than n, the cost of inverting $\mathbf{C}$ is negligible. Equation (2.8) is an ODE, and therefore its computational cost is negligible compared to the f-OTD modes, which are governed by a PDE. The cost of computing the terms that appear on the right-hand side of (2.8) is already accounted for in (2.7). Also, the computational storage requirement of solving r f-OTD modes is equivalent to that of solving r forward sensitivity equations, as the storage cost of each f-OTD mode is equivalent to a single sensitivity field and the storage cost of $\mathbf{Y}$ is negligible.

2.3. Equivalence. It is important to note that the choice of ϕ_{ij} in (2.7) and (2.8) is not unique, and any skew-symmetric matrix yields an equivalent reduction. Similar to the OTD equations [8], we choose $\phi_{ij} = 0$, which corresponds to the dynamically orthogonal condition. This property is summarized in the theorem below.

THEOREM 2.1. *Let* $\{\mathbf{U}(x,t), \mathbf{Y}(t)\}$ *and* $\{\tilde{\mathbf{U}}(x,t), \tilde{\mathbf{Y}}(t)\}$ *represent two reductions that satisfy* (2.7) *and* (2.8) *with corresponding skew-symmetric matrices* $\mathbf{\Phi}(t)$ *and* $\tilde{\mathbf{\Phi}}(t)$*, respectively. If the reductions are equivalent at* $t = 0$*, i.e., they are initially related by an orthogonal rotation matrix* $\mathbf{R}_0 \in \mathbb{R}^{r\times r}$ *as* $\mathbf{U}(x,0) = \tilde{\mathbf{U}}(x,0)\mathbf{R}_0$ *and* $\mathbf{Y}(0) = \tilde{\mathbf{Y}}(0)\mathbf{R}_0$*, then the two reductions will remain equivalent for* $t > 0$ *with rotation matrix* $\mathbf{R}(t)$ *governed by* $\dot{\mathbf{R}} = \mathbf{R}\mathbf{\Phi} - \tilde{\mathbf{\Phi}}\mathbf{R}$.

For proof of the above theorem see Appendix B.

2.4. Approximation error. The approximation error of estimating sensitivities using f-OTD can be expressed as $e(t) = \|\mathbf{V}'(x,t) - \mathbf{U}(x,t)\mathbf{Y}(t)^T\|_F$. This error can be properly analyzed and better understood by considering two types of error: (i) the resolved error denoted by $e_r(t)$ and (ii) the unresolved error denoted by $e_u(t)$. The resolved error is the discrepancy between approximating the sensitivities with rank-r f-OTD and the optimal rank-r approximation: $e_r(t) = \|\mathbf{U}(x,t)\mathbf{Y}(t)^T - \tilde{\mathbf{U}}(x,t)\tilde{\mathbf{Y}}(t)^T\|_F$, where $\tilde{\mathbf{U}}(x,t) \in \mathbb{R}^{\infty\times r}$ and $\tilde{\mathbf{Y}}(t) \in \mathbb{R}^{d\times r}$ are the optimal rank-r orthonormal modes and their coefficients, respectively. The unresolved error is the error of the optimal rank-r approximation: $e_u(t) = \|\tilde{\mathbf{U}}(x,t)\tilde{\mathbf{Y}}(t)^T - \mathbf{V}'(x,t)\|_F$, which is a direct result of truncating the $d - r$ least energetic modes. Thus, the optimal rank-r approximation is obtained by minimizing

$$\mathcal{E}_u(\tilde{\mathbf{U}}(x,t), \tilde{\mathbf{Y}}(t)) = \left\| \tilde{\mathbf{U}}(x,t)\tilde{\mathbf{Y}}(t)^T - \mathbf{V}'(x,t) \right\|_F \tag{2.9}$$

subject to the orthonormality condition of $\tilde{\mathbf{U}}(x,t)$ modes. The optimal decomposition can be obtained by performing instantaneous SVD of the sensitivity matrix, where

$\tilde{\mathbf{U}}(x,t)$ is the matrix of r most dominant left singular vectors of $\mathbf{V}'(x,t)$ and $\tilde{\mathbf{Y}}(t) = \tilde{\mathbf{Z}}(t)\tilde{\mathbf{\Sigma}}(t)$, where $\tilde{\mathbf{Z}}(t) \in \mathbb{R}^{d\times r}$ and $\tilde{\mathbf{\Sigma}}(t) = \mathrm{diag}(\tilde{\sigma}_1(t), \tilde{\sigma_2}(t), \ldots, \tilde{\sigma}_r(t))$ are the matrix of the r most dominant right singular vectors and the matrix of singular values, respectively. It is straightforward to show that $e_u(t) = (\sum_{i=r+1}^{d} \tilde{\sigma}_i^2(t))^{1/2}$. The error $e_u(t)$ represents the minimum error that any rank-r approximation can achieve, and therefore, it amounts to a lower bound for the f-OTD error: $e(t) \geq e_u(t)$. On the other hand, as with any ROM of a time-dependent system, the unresolved subspace induces a *memory error* in the f-OTD approximation. This means that the unresolved error *drives* the resolved error $e_r(t)$, and under appropriate conditions, it has been shown that for similar time-dependent basis low-rank approximations, $e_r(t)$ can be bounded by $e_r(t) \leq c_1 e^{c_2 t} \int_{t_0}^{t} e_u(s)ds$ [30] for $c_1, c_2 > 0$. The interplay between $e_u(t)$ and $e_r(t)$ can be more rigorously studied within Mori–Zwanzig formalism [31]. These error estimates can guide an adaptive f-OTD, in which modes are added or removed to maintain the error below some threshold value [16]; however, these aspects are not in the scope of this paper and are not explored any further here. Since sensitivities can either be very small or very large with errors following the same trend, we compute the relative error percentages as shown here:

$$\% \text{ Error} = \frac{e(t)}{\|\mathbf{V}'(x,t)\|_F} \times 100. \tag{2.10}$$

Similar quantities are computed for $e_u(t)$ and $e_r(t)$.

2.5. Mode ranking. In this section we present a procedure to rank the f-OTD modes and their coefficients according to their significance. To this end, we start by considering the reduced correlation matrix $\mathbf{C}(t)$, which is in general a full matrix. This implies that the sensitivity coefficients are correlated and there exists a linear mapping from the correlated coefficients, $\mathbf{Y}(t)$, to the uncorrelated coefficients, $\hat{\mathbf{Y}}(t)\mathbf{\Sigma}(t)$, where $\hat{\mathbf{Y}}(t)$ are the orthonormal coefficients and $\mathbf{\Sigma}(t) = \mathrm{diag}(\sigma_1(t), \sigma_2(t), \ldots, \sigma_r(t))$ is a diagonal matrix of singular values. To find such a mapping, we consider the eigendecomposition of $\mathbf{C}(t)$ as follows:

$$\mathbf{C}(t)\mathbf{R}(t) = \mathbf{R}(t)\mathbf{\Lambda}(t), \tag{2.11}$$

where $\mathbf{R}(t) \in \mathbb{R}^{r\times r}$ is a matrix whose columns contain the eigenvectors of $\mathbf{C}(t)$ and $\mathbf{\Lambda}(t) = \mathrm{diag}(\lambda_1(t), \lambda_2(t), \ldots, \lambda_r(t))$ is a diagonal matrix containing the eigenvalues of $\mathbf{C}(t)$. Since $\mathbf{C}(t)$ is a symmetric positive matrix, the matrix $\mathbf{R}(t)$ is an orthonormal matrix, i.e., $\mathbf{R}(t)^T\mathbf{R}(t) = \mathbf{I}$, and the eigenvalues are all nonnegative and can be sorted as $\lambda_1(t) > \lambda_2(t) > \cdots > \lambda_r(t) \geq 0$. It is also straightforward to show that the singular values of the f-OTD low-rank approximation are $\sigma_i(t) = \lambda_i(t)^{1/2}$ for $i = 1, 2, \ldots, r$.

The ranked f-OTD components can be defined as

$$\hat{\mathbf{Y}}(t) = \mathbf{Y}(t)\mathbf{R}(t)\mathbf{\Sigma}^{-1}(t), \qquad \hat{\mathbf{U}}(x,t) = \mathbf{U}(x,t)\mathbf{R}(t),$$

where the columns of $\hat{\mathbf{Y}}(t)$ and $\hat{\mathbf{U}}(x,t)$ are ranked by energy (σ_i^2) in descending order. We shall refer to $\{\hat{\mathbf{Y}}(t)\mathbf{\Sigma}(t), \hat{\mathbf{U}}(x,t)\}$ as the bi-orthonormal form of the reduction. Since the above equations are simply an in-subspace rotation, $\{\hat{\mathbf{Y}}(t)\mathbf{\Sigma}(t), \hat{\mathbf{U}}(x,t)\}$ and $\{\mathbf{Y}(t), \mathbf{U}(x,t)\}$ yield equivalent low-rank approximations of the full-dimensional dynamics. This is easily verified by considering the bi-orthonormal form of the low-rank approximation as $\hat{\mathbf{U}}(x,t)\mathbf{\Sigma}(t)\hat{\mathbf{Y}}(t)^T = \mathbf{U}(x,t)\mathbf{Y}(t)^T$, where we have made use of the identity $\mathbf{R}(t)^T\mathbf{R}(t) = \mathbf{I}$. We refer to $\hat{\mathbf{Y}}$ as the *hidden* parametric space, as

each column of matrix $\hat{\mathbf{Y}}$ can be taken as a new ranked parameter that represents the contribution of all parameters ($\boldsymbol{\alpha}$). In the following sections, all figures will be presented in bi-orthonormal form.

3. Demonstration cases.

3.1. Rössler system. We first present a simple demonstration of f-OTD by computing sensitivities of the Rössler system. The Rössler system is governed by

$$\frac{dv_1}{dt} = -v_2 - v_3, \qquad \frac{dv_2}{dt} = v_1 + \alpha_1 v_2, \qquad \frac{dv_3}{dt} = \alpha_2 + v_3(v_1 - \alpha_3). \tag{3.1}$$

In the above equations, we set $\alpha_1 = \alpha_2 = 0.1$ and $\alpha_3 = 14$, which are common values used to study the chaotic behavior of the attractor. The goal is to calculate the sensitivity of $\mathbf{v}$ with respect to the model parameters $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \alpha_3)$ as $\partial \mathbf{v}/\partial \boldsymbol{\alpha}$. To this end, we take the derivative of the above system of equations with respect to model parameter α_i to obtain the sensitivity equation

$$\frac{d\mathbf{V}'}{dt} = \mathbf{L}\mathbf{V}' + \mathbf{F}', \tag{3.2}$$

where

$$\mathbf{L} = \begin{bmatrix} 0 & -1 & -1 \\ 1 & \alpha_1 & 0 \\ v_3 & 0 & v_1 - \alpha_3 \end{bmatrix}, \quad \mathbf{V}' = \begin{bmatrix} | & | & | \\ \mathbf{v}_1' & \mathbf{v}_2' & \mathbf{v}_3' \\ | & | & | \end{bmatrix}, \quad \mathbf{F}' = \begin{bmatrix} 0 & 0 & 0 \\ v_2 & 0 & 0 \\ 0 & 1 & -v_3 \end{bmatrix},$$

and $\mathbf{v}_i'$ is the sensitivity of the position with respect to α_i and $\mathbf{L} \in \mathbb{R}^{n \times n}$ and $\mathbf{F}' \in \mathbb{R}^{n \times d}$. We choose a subspace with dimension $r = 2$ for the low-rank approximation of the three-dimensional ($d = 3$) sensitivities ($\mathbf{V}'$). Although it is obvious that OTD modes are not based on parametric sensitivities and they are based on perturbations in the initial condition (IC) in all directions of the phase space, we believe it is instructive to contrast the OTD versus f-OTD to better understand f-OTD. To this end, we build two real-time ROMs using OTD modes and f-OTD modes. In the case of OTD, we solve the OTD evolution equation, and we project the forced sensitivity equation (3.2) onto the OTD modes, resulting in

$$\frac{d\mathbf{U}_{otd}}{dt} = (\mathbf{I} - \mathbf{U}_{otd}\mathbf{U}_{otd}^T)\mathbf{L}\mathbf{U}_{otd} \quad \text{and} \quad \frac{d\mathbf{Y}_{otd}}{dt} = \mathbf{Y}_{otd}\mathbf{U}_{otd}^T\mathbf{L}^T\mathbf{U}_{otd} + \mathbf{F}'^T\mathbf{U}_{otd}.$$

We also solved the f-OTD evolution equations (2.7) and (2.8) for the finite-dimensional system. Both OTD and f-OTD modes are initialized with the same subspace, and the evolution equations are solved for $T_f = 10$ units of time. These subspaces are initialized by first solving the full-dimensional sensitivity equation (3.2) for one $\Delta t = 10^{-2}$ and then computing the OTD and f-OTD subspaces as the first two left singular vectors of $\mathbf{V}'(x, t = \Delta t)$. In Figure 2(a), both OTD and f-OTD subspaces are visualized along with the attractor of the Rössler system. The OTD subspace is shown at only one instant for clarity, and that point corresponds to the case where the nonlinear dynamics is in the $v_1 - v_2$ plane. At this point, the OTD subspace is oriented such that it nearly coincides with the $v_1 - v_2$ plane. This result is to be expected since the OTD subspace follows the sensitivities associated with the perturbations in the IC and we know that the IC-perturbed solutions will lie on the *same* attractor. On the other hand, the f-OTD subspace is correctly oriented along the most sensitive subspace for perturbations in the model parameters, i.e., $\delta\boldsymbol{\alpha} = (\delta\alpha_1, \delta\alpha_2, \delta\alpha_3)$, which

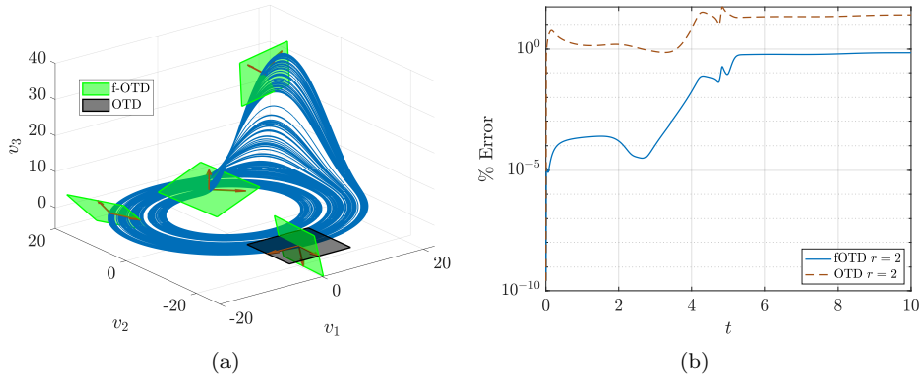


FIG. 2. (a) *Chaotic Rössler attractor with optimal f-OTD subspace shown in green and OTD subspace shown in black for* $r = 2$. *Red arrows depict the orthonormal sensitivity vectors that define each subspace.* (b) *Percent error for* $e(t)$ *plotted versus time for the f-OTD and OTD subspaces.*

lead to perturbations in the attractor itself. That is, the perturbed solutions lie on *different* attractors which can readily be seen as $\delta\boldsymbol{\alpha}$ results in nonzero δv_3, despite $v_3 \simeq 0$. This results in the f-OTD subspace having a large out-of-plane component in the v_3 direction, which the OTD subspace fails to capture in Figure 2(a). In Figure 2(b), the percent errors of $e(t)$ are shown for OTD and f-OTD, which confirms that f-OTD performs significantly better than OTD. This simple example demonstrates that the OTD basis is not optimal and may be inaccurate for reduced order modeling of the forced sensitivity equation.

3.2. Chaotic Kuramoto-Sivashinsky equation. The objective of this example is to evaluate the performance of f-OTD in computing sensitivities of a chaotic system with many positive Lyapunov exponents and a high-dimensional parametric space. The intent of this example is not to compute the gradient of a time-averaged quantity for a chaotic system but rather to compute the solution of the sensitivity equation for a chaotic system with much larger unstable directions than the rank of the f-OTD subspace. For computing sensitivities of time-averaged quantities, one can use f-OTD in conjunction with Ruelle's linear response formula [32, 33] to compute ensemble sensitivities. We also refer the reader to references for methods related to long-term sensitivities in chaotic systems [34, 35]. To this end, we consider the sensitivity of the Kuramoto–Sivashinsky (KS) equation with respect to a time-dependent forcing parameter $\alpha(t)$. The KS equation is a fourth order PDE given by

$$\frac{\partial \mathbf{v}}{\partial t} + \frac{1}{2}\frac{\partial \mathbf{v}^2}{\partial x} + \frac{\partial^2 \mathbf{v}}{\partial x^2} + \nu\frac{\partial^4 \mathbf{v}}{\partial x^4} = \alpha(t)\sin(2\pi x/L), \quad x \in [0, L], \tag{3.3}$$

where $\mathbf{v} = \mathbf{v}(x, t)$. Approximately 110 positive Lyapunov exponents exist for the parameters used in this study: $\nu = 1$ and $L = 1000$. Here $\alpha(t)$ represents an infinite-dimensional parametric space.

To compute the sensitivities numerically, we consider a discrete representation of $\alpha(t)$ in the interval $t_i \in [0, T_s]$, where $T_s \leq T_f$ is a subset of the full integration time T_f and t_i is a discrete instance in time. To this end, we consider the value of $\alpha(t)$ at discrete time $t_i = (i-1) \times \Delta t$, where Δt is the time step. This results in a vector, $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_d)$, where $\alpha_i = \alpha(t_i)$ and $d = T_s/\Delta t$ is the number of instances in time (i.e., number of parameters). In general, Δt can be chosen independently

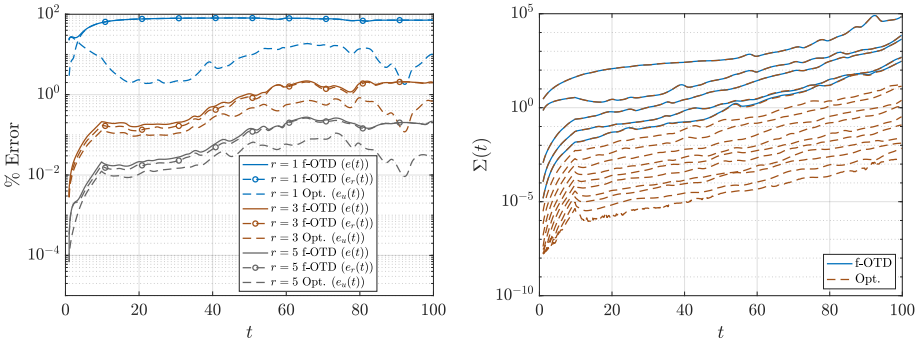
of the numerical time integration step size; however, for simplicity, we use the same value of Δt for both the parametric discretization and numerical integration of the nonlinear solver and f-OTD equations. In this example, we consider $\Delta t = 10^{-2}$ and $T_s = 10$, which results in $d = 1000$ parameters. This leads to the sensitivity of $\mathbf{v}$ with respect to the value of $\alpha(t)$ at 1000 evenly spaced instances in time. We evolve these sensitivities over the interval $t \in [0, T_f]$ with $T_f = 100$. We also choose $\alpha(t) = 0$ for $t_i \in [0, T_f]$, and therefore, the nonlinear solver $\mathbf{v}(t)$ is the solution of the unforced KS equation.

We consider the time-discrete form of (3.3) and differentiate with respect to design parameter α_i. This leads to an evolution equation for the sensitivity of $\mathbf{v}$ with respect to α_i, in which the linear operator and forcing terms are

(3.4)
$$\mathcal{L}(\mathbf{v}_i') = -\left[\frac{\partial(\mathbf{v}\mathbf{v}_i')}{\partial x} + \frac{\partial^2 \mathbf{v}_i'}{\partial x^2} + \nu \frac{\partial^4 \mathbf{v}_i'}{\partial x^4}\right] \text{ and } \mathbf{f}_i' = \delta(t - t_i) \sin(2\pi x/L), \ i = 1, 2, \ldots, d,$$

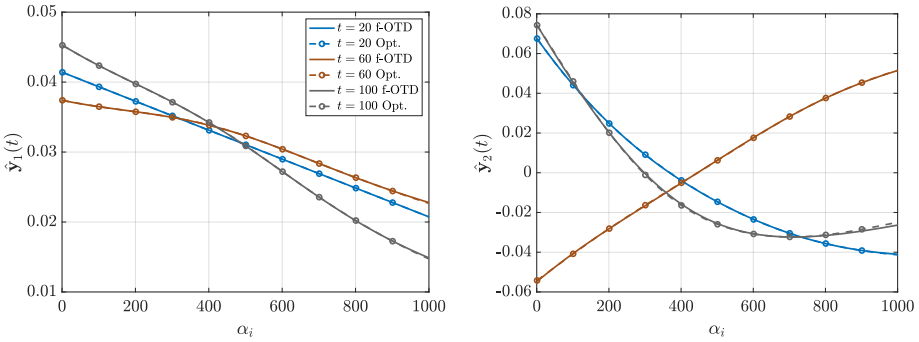
where $\delta(t - t_i) = 0$ for $t \neq t_i$ and $\delta(t - t_i) = 1$ for $t = t_i$. Our goal is to solve (3.4) using f-OTD. We discretize the KS equation and the f-OTD equations using $n = 2^{13} = 8192$ Fourier modes and use an exponential time-differencing Runge–Kutta fourth order time stepping scheme [36]. We verify our solution by directly solving (3.4) for all 1000 sensitivities. Further decreasing Δt and increasing the number of Fourier modes did not change our results. We also compare the f-OTD error with that of optimal instantaneous same-rank approximation of the full sensitivities, which is obtained by computing the SVD of $\mathbf{V}'(x, t)$ at each time. In Figure 3(a), we compare the reconstruction error of f-OTD ($e(t)$) with the reconstruction error of same-rank SVD ($e_u(t)$). We also show the resolved error $e_r(t)$, which measures the discrepancy between the f-OTD approximation and the optimal same-rank approximation. We compute these errors for $r = 1, 3$, and 5. While the optimal low-rank approximation with a single mode captures approximately 99% of the system energy of the full sensitivity (see Figure 3(b)), the f-OTD approximation performs poorly with only a single mode, i.e., a dramatic reduction for 1000 sensitivities. This is a direct result of the memory effect from the lost interactions with the unresolved modes ($e_r(t)$) that ultimately dominate the error for long-term integration. By increasing the number of f-OTD modes, both $e(t)$ and $e_r(t)$ decrease. It is possible to control the error in real time through an adaptive strategy that adds/removes modes with an appropriate criterion. For example, a candidate criterion could be $p = \sigma_r^2(t)/\sum_{i=1}^{r} \sigma_i^2(t)$, where for $p < p_{th}$ the last mode can be removed and for $p > p_{th}$ a new mode can be added. See [16] for similar strategies for adaptive mode addition and removal.

In Figure 3(b), we compare the 15 largest instantaneous singular values of quasi-matrix $\mathbf{V}'(x, t)$ with those obtained from f-OTD with rank $r = 5$, which shows that f-OTD closely captures the most dominant subspace. In Figure 4(a) and (b) the orthonormalized coefficients of the first two dominant f-OTD modes for the case of $r = 5$ are compared to the right singular vectors from the instantaneous SVD of $\mathbf{V}'(x, t)$. These coefficients represent the hidden parametric space: for example, $\hat{\mathbf{y}}_1$ is a series of weights that represent the contribution of each of the $d = 1000$ sensitivities to the most dominant direction of the full sensitivity matrix, $\hat{\mathbf{u}}_1$. Due to the chaotic nature of this problem, we observe that these coefficients can be highly time-dependent, especially for the lower energy modes; see $\hat{\mathbf{y}}_2$. Nevertheless, we have demonstrated that f-OTD extracts the most dominant subspace and associated coefficients of the sensitivity matrix for a chaotic system with large number of unstable directions and parameters.



(a) (b)

FIG. 3. (a) *Comparison of the reconstruction error between f-OTD approximation* $(e(t))$ *and optimal rank-r approximation* $(e_u(t))$ *for different reduction sizes. Resolved error,* $e_r(t)$*, dominates the f-OTD error for long-term integration. Error decreases as the number of modes increases.* (b) *Comparison of singular values between f-OTD and optimal low-rank decomposition for* $r = 5$.



(a) (b)

FIG. 4. *KS The first two columns of the orthonormalized design variable matrix shown at different instances in time:* (a) $\hat{\mathbf{y}}_1(t)$, (b) $\hat{\mathbf{y}}_2(t)$. *The horizontal axis corresponds to the ith design parameter* α_i.

3.3. Species transport equation: Turbulent reactive flow. In this example, we show how a single set of f-OTD modes can lead to significant computational gains for computing sensitivities in problems with multiple coupled field variables, where each field variable has a different linear operator. We consider a species transport problem, where parameter identification via sensitivity analysis plays an important role in allocating computational and experimental resources to reduce parameter uncertainty. Moreover, the sensitivity analysis is used to create reduced reaction mechanisms for complex chemical systems involving a large number of species and reactions. See [20, 21, 22].

3.3.1. Problem setup. To this end, we consider a 2D incompressible turbulent reactive flow:

$$\frac{\partial \mathbf{v}_i}{\partial t} + (\mathbf{w} \cdot \nabla)\, \mathbf{v}_i = \tilde{\kappa}_{ik} \nabla^2 \mathbf{v}_k + \mathbf{s}_i, \tag{3.5}$$

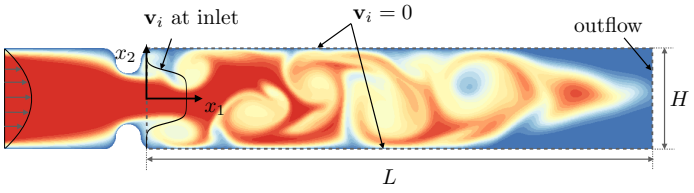


FIG. 5. *Schematic of the flow visualized with a passive scalar.*

where $\mathbf{w} = (\mathbf{w}_{x_1}(x_1, x_2, t), \mathbf{w}_{x_2}(x_1, x_2, t))$ is the velocity field from the 2D incompressible Navier–Stokes equations, $\mathbf{v}_i = \mathbf{v}_i(x_1, x_2, t)$ is the concentration of species i, $\tilde{\kappa}_{ik} \in \mathbb{R}^{n_s \times n_s}$ is the diffusion coefficient matrix, and $\mathbf{s}_i = \mathbf{s}_i(\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_{n_s}; \boldsymbol{\alpha})$ is the nonlinear reactive source term. We choose a diagonal diffusion coefficient matrix, where the ith diagonal entry is the diffusion coefficient of the ith species, and n_s is the number of species. For the reactive source term $\mathbf{s}_i$, we consider the biological reactions used in [37]. These terms are listed in Table 1 in Appendix C for reference. A schematic of the flow is shown in Figure 5, where L and H are the channel length and height, respectively. The no-slip boundary condition is enforced at the top and bottom walls while the outflow boundary condition is enforced downstream. At the inlet a parabolic velocity with the average inlet velocity of $\overline{w}$ is prescribed. The Reynolds number, based on a reference length of half the height $(H/2)$ and the kinematic viscosity, ν is $Re = \overline{w}H/2\nu = 1000$. The inlet boundary condition is $\mathbf{v}_i(0, x_2, t) = 1/2\big(\tanh(x_2 + H/2)/\delta - \tanh(x_2 - H/2)/\delta\big)$ for all species, where $\delta = 0.1$.

The velocity field is governed by a 2D incompressible Navier–Stokes equation. We solved the velocity field once as it is independent from the species using the spectral/hp element method with 4008 quadrilateral elements and polynomial order 5. For more details on the spectral element method see, for example, [38, 39, 40]. We then solve the species transport equations and f-OTD equations in the rectangular domain shown by dashed lines in Figure 5. In the rectangular domain, we used structured spectral elements with 50 elements in the x_1 direction and 15 elements in the x_2 direction. We used spectral polynomial of order 5 in each direction. The velocity field was interpolated onto this grid. The f-OTD equations, which are presented in the next sections, and the species transport equation are integrated forward in time using fourth order Runge–Kutta with $\Delta t = 5 \times 10^{-4}$.

3.3.2. f-OTD formulation. Our goal is to calculate sensitivity of the species concentration with respect to the reaction parameters $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_{n_r})$, where n_r is the number of reaction parameters. To this end, we take the derivative of the above equation with respect to reaction parameter α_j to obtain an evolution equation for the sensitivity:

$$\frac{\partial \tilde{\mathbf{v}}'_{ij}}{\partial t} + (\mathbf{w} \cdot \nabla)\, \tilde{\mathbf{v}}'_{ij} = \tilde{\kappa}_{ik} \nabla^2 \tilde{\mathbf{v}}'_{kj} + \tilde{\mathcal{L}}_{\mathbf{s}_{ik}} \tilde{\mathbf{v}}'_{kj} + \tilde{\mathbf{s}}'_{ij}, \tag{3.6}$$

where $\tilde{\mathbf{v}}'_{ij} = \partial \mathbf{v}_i / \partial \alpha_j \in \mathbb{R}^{\infty \times 1}$ is the sensitivity of the concentration of species $\mathbf{v}_i$ with respect to reaction parameter α_j, $\tilde{\mathcal{L}}_{\mathbf{s}_{ik}} = \partial \mathbf{s}_i / \partial \mathbf{v}_k$ is the linearized reactive source term, and $\tilde{\mathbf{s}}'_{ij} = \partial \mathbf{s}_i / \partial \alpha_j$. In the above equation, $\tilde{\mathcal{L}}_{\mathbf{s}_{ik}} \tilde{\mathbf{v}}'_{kj}$ should be interpreted as a matrix-matrix multiplication for any (x_1, x_2) point in the physical space. In this notation, sensitivities are represented by a quasitensor, i.e., $\tilde{\mathbf{V}}' = [\mathbf{v}'_{ij}]$ with $i = 1, 2, \ldots, n_s$ and $j = 1, 2, \ldots, n_r$, where $\tilde{\mathbf{V}}' \in \mathbb{R}^{\infty \times n_s \times n_r}$ is the third order quasitensor

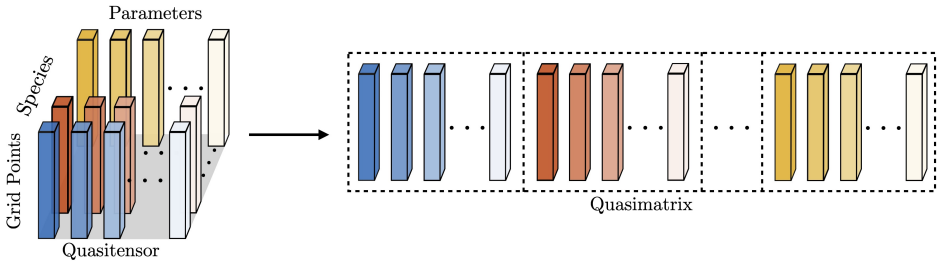


FIG. 6. *Schematic of the tensor flattening from a 3D quasitensor to a 2D quasimatrix.*

depicted in the left-hand side of Figure 6. Here $\tilde{\cdot}$ denotes terms associated with the tensor equation. In the discrete representation of $\tilde{\mathbf{V}}'$, the dimension ∞ is replaced with the number of grid points.

Solving for sensitivities involving $\tilde{\mathbf{v}}'_{ij}$ using an adjoint approach would require solving n_s AEs: one adjoint field for each species. See, for example, [20, 21, 22]. However, it is important to note that these AEs are tied to a specific objective function and do not directly compute $\tilde{\mathbf{v}}'_{ij}$. Consequently, each subsequent objective function would require solving another n_s AEs. To directly solve for $\tilde{\mathbf{v}}'_{ij}$ using f-OTD, one could also solve for n_s sets of f-OTD modes, i.e., one set of f-OTD modes for each species. This straightforward approach would only exploit the correlation between sensitivities of each species separately, i.e., correlations between $\mathbf{v}'_{ij}$ for a fixed i, while leaving the correlations between sensitivities of different species unexploited. In this example, we demonstrate how a single set of f-OTD modes can be used to accurately model the entire sensitivity tensor. Therefore, the compression ratio both in terms of memory and computational cost in comparison to the full sensitivity equation is r/d. In comparison to AE, the compression ratio is r/n_s. Also, the f-OTD is a forward system and does not impose any I/O operation. To this end, we flatten the sensitivity tensor, as shown in Figure 6, which results in a quasimatrix of size $\infty \times d$. Here, $d = n_s \times n_r$, where $n_s = 23$ and $n_r = 34$. This leads to a total of $d = 782$ sensitivity equations that we seek to compute. In Appendix D, we show that the flattened sensitivity evolution equation is

$$\frac{\partial \mathbf{v}'_m}{\partial t} + (\mathbf{w} \cdot \nabla)\mathbf{v}'_m = \kappa_{mn}\nabla^2 \mathbf{v}'_n + \mathcal{L}_{\mathbf{s}_{mn}}\mathbf{v}'_n + \mathbf{s}'_m, \tag{3.7}$$

where $m(i,j) = j+(i-1)n_r$ and $n(i',j') = j'+(i'-1)n_r$, resulting in $m, n = 1, 2, \ldots, d$. Equation (3.6) is a tensor evolution equation, whereas (3.7) is the equivalent matrix evolution equation. The tensor flattening carried out here is similar to the unfolding carried out in the Tucker tensor decomposition [41]. However, unlike Tucker tensor decomposition, we do not consider flattening the tensor in the other two dimensions of species and parameters. Each $\mathbf{y}_k(t)$ is a vector of size $(n_s n_r) \times 1$ and contains coefficients for species and parameters. Once the sensitivity tensor is flattened to a quasimatrix, we use f-OTD to extract a low-rank structure from the quasimatrix. In (3.7), the linear operator changes from one species to the other due to the different diffusion coefficients κ_{mn}. In Appendix D we show how f-OTD evolution equations can be derived for this case, which is different from the previous demonstration cases.

We solve (D.2) and (D.3) for different f-OTD ranks along with the species transport equation (3.5). In Figure 7(a) the f-OTD error ($e(t)$) and optimal low-rank approximation error ($e_u(t)$) are shown using three different ranks of $r = 2, 5$, and 8. Again, we observe that the growth of $e(t)$ surpasses $e_u(t)$ for long-term integration as a direct result of the lost interactions with the unresolved modes. However, with only

5–8 modes, we have shown that f-OTD can approximate 782 sensitivities with error on the order of 0.1%. These results can be explained by studying Figure 7(b), where we observe that more than 99% of the system energy is captured by the reduction. The % energy is calculated from the singular values as $\%\ \text{En.} = \sum_{i=1}^{r} \sigma_i^2 / \sum_{i=1}^{d} \sigma_i^2 \times 100$ and can be used to get a sense of the dimensionality of the system when expressed in the time-dependent basis. Since the system is truly low-dimensional in the time-dependent basis, the f-OTD algorithm is able to extract the latent features associated with the most dominant singular values and successfully approximate the full sensitivity tensor with a high degree of accuracy.

In Figure 8, the time-dependent evolution of the three most dominant f-OTD modes is shown. These modes are energetically ranked where low mode numbers correspond to larger (higher energy) structures and high mode numbers correspond to finer (lower energy) structures in the flow. As opposed to static bases, such as proper orthogonal decomposition (POD) or dynamic mode decomposition (DMD), the f-OTD modes evolve with the flow and exploit the instantaneous correlations between sensitivities. While this system is low-dimensional in the time-dependent

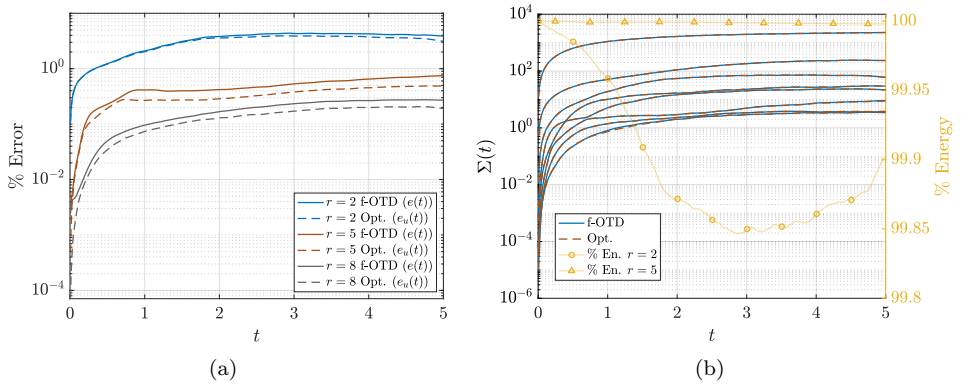


FIG. 7. (a) *Percent error plotted as a function of time. Error decreases as the number of modes r increases.* (b) *Singular values plotted as a function of time for* $r = 8$.

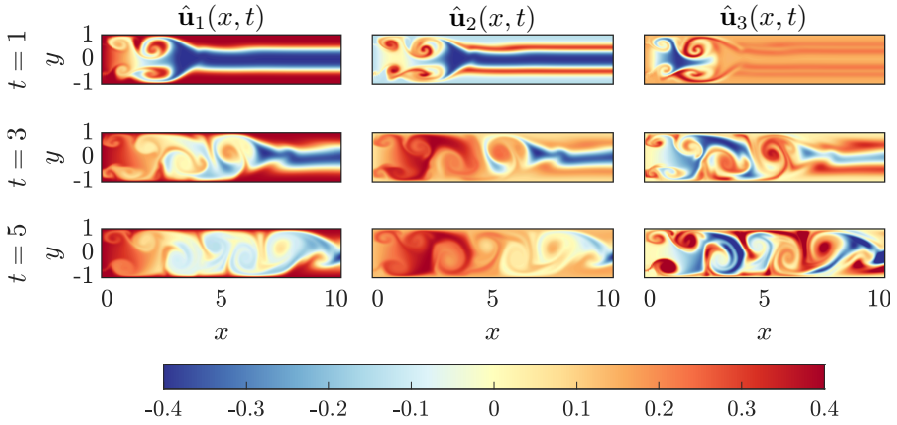


FIG. 8. *First three orthonormal f-OTD modes shown for* $r = 8$. *Each row shows the modes at a different instance in time.*

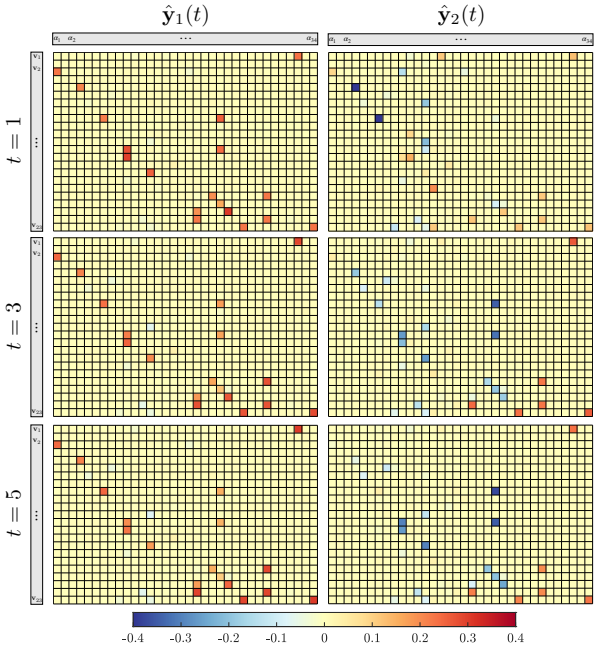


FIG. 9. *Orthonormalized f-OTD coefficients* $\hat{\mathbf{y}}_1(t)$ *and* $\hat{\mathbf{y}}_2(t)$ *visualized as a matrix with rows corresponding to species concentration and columns corresponding to reaction parameters. Color map shows most dominant sensitivities at different time instances.*

basis, when expressed in a POD or DMD basis, the system is high-dimensional and many modes are needed to capture the complex spatiotemporal evolution of $\mathbf{V}'$. See [24] for comparison between a time-dependent basis versus POD and DMD, and see [42] for a recent review of ROM techniques.

To demonstrate the interpretability of the f-OTD decomposition, we show how the hidden parameter space represented by $\hat{\mathbf{Y}}(t)$ can be used to identify the most important reaction parameters. In this context, importance refers to a parameter for which a small change in its value elicits a large change in the response of the system (i.e., highly sensitive). To demonstrate this capability of f-OTD, the first two sensitivity coefficients are visualized as matrices in Figure 9, where each $\hat{\mathbf{y}}_i$ is a $d \times 1$ vector that has been reshaped into an $n_s \times n_r$ matrix. In this form, each $\mathbf{v}'_{ij}$ is visualized using a heat map of the sensitivity coefficients, with rows corresponding to species i and columns corresponding to reaction parameter j. Using this heat map, Figure 9 shows that only a handful of sensitivities are nonzero, while the majority have zero contribution for the entire duration of the simulation.

4. Conclusions. We present a real-time reduced order modeling approach for the computation of sensitivities in evolutionary systems governed by time-dependent ODEs/PDEs. The computational cost of solving the f-OTD equations of rank r is roughly equivalent to that of solving r forward sensitivity equations. We demonstrated that the rank of f-OTD for two diverse applications is much smaller than the number of sensitivity fields. In contrast to adjoint based methods, f-OTD is not tied to an objective function and requires solving a system of *forward* equations that does not require any I/O operation. We showed that a single set of f-OTD modes can be formulated to compress the sensitivities of multivariable PDEs. We demonstrated

this capability by computing sensitivities of multiple species with respect to reaction parameters in a turbulent reactive flow.

We have contrasted the f-OTD with OTD and demonstrated why OTD is not appropriate for parametric/forced sensitivity analysis. In contrast to traditional ROM approaches, f-OTD extracts the low-rank approximation directly from the sensitivity equations as opposed to a data-driven approach, such as POD or DMD, which requires the full-dimensional sensitivity data. The data-driven techniques have the computational advantage that the modes are computed once and the cost of solving ROM is usually insignificant. However, the low-rank subspace in the data-driven approach is fine-tuned to particular operating conditions, whereas the f-OTD subspace evolves with the dynamics of the system and does not require such fine-tuning. As such, f-OTD is an *on the fly* model compression that is achieved by extracting instantaneous correlated structures in the solution.

We conclude with a word of caution for using the presented approach to compute sensitivities for a generic system. Although we have only dealt with multivariable PDEs with homogeneous fields and homogeneous parameters, special care must be taken when dealing with heterogeneous fields and/or parameters in order for (2.7)–(2.8) and (D.2)–(D.3) to remain dimensionally consistent. Therefore, we make the following recommendations: (i) when dealing with sensitivities with respect to heterogeneous parameters (i.e., parameters with different physical dimension), a weighted inner product in the parametric space should be applied to appropriately scale the sensitivity fields and extract meaningful correlations between different parameters, and (ii) when dealing with multivariable PDEs with heterogeneous fields, the PDE should be nondimensionalized and normalized so that the above tensor unfolding remains valid and correlations can be extracted across sensitivities of different field variables.

Appendix A. Optimality conditions of the variational principle. For the sake of brevity, we forgo the explicit written dependencies on x, t, and $\boldsymbol{\alpha}$ in the following derivation. Using index notation, we start by expanding (2.6):

$$\begin{aligned}\mathcal{G}(\dot{\mathbf{U}},\dot{\mathbf{Y}},\lambda) = &\left\langle \dot{\mathbf{u}}_i,\dot{\mathbf{u}}_j\right\rangle \left(\mathbf{y}_i^T\mathbf{y}_j\right) + \left\langle \mathbf{u}_i,\mathbf{u}_j\right\rangle \left(\dot{\mathbf{y}}_i^T\dot{\mathbf{y}}_j\right) + 2\left\langle \dot{\mathbf{u}}_i,\mathbf{u}_j\right\rangle \left(\mathbf{y}_i^T\dot{\mathbf{y}}_j\right)\\ &- 2\left\langle \dot{\mathbf{u}}_i,\mathcal{L}(\mathbf{u}_j)\right\rangle \left(\mathbf{y}_i^T\mathbf{y}_j\right) - 2\left\langle \mathbf{u}_i,\mathcal{L}(\mathbf{u}_j)\right\rangle \left(\dot{\mathbf{y}}_i^T\mathbf{y}_j\right)\\ &+ \left\langle \mathcal{L}(\mathbf{u}_i),\mathcal{L}(\mathbf{u}_j)\right\rangle \left(\mathbf{y}_i^T\mathbf{y}_j\right) - 2\left\langle \dot{\mathbf{u}}_i,\mathbf{F}'\mathbf{y}_i\right\rangle - 2\left\langle \mathbf{u}_i,\mathbf{F}'\dot{\mathbf{y}}_i\right\rangle\\ &+ 2\left\langle \mathcal{L}(\mathbf{u}_i),\mathbf{F}'\mathbf{y}_i\right\rangle + \left\|\mathbf{F}'\right\|_F^2 + \lambda_{ij}\left(\left\langle \mathbf{u}_i,\dot{\mathbf{u}}_j\right\rangle - \phi_{ij}\right).\end{aligned}$$

The first order optimality condition requires that the derivative of $\mathcal{G}$ with respect to $\dot{\mathbf{U}},\dot{\mathbf{Y}}$, and λ vanish. The derivative of $\mathcal{G}$ with respect to λ produces the time derivative of the orthonormality constraint given by (2.5). Provided that the f-OTD modes are orthonormal at $t=0$, the time integration of (2.5) reproduces the orthonormality condition of the f-OTD modes for $t>0$: $\left\langle \mathbf{u}_i,\mathbf{u}_j\right\rangle = \delta_{ij}$. To take the derivative of $\mathcal{G}$ with respect to $\tilde{\dot{\mathbf{u}}}_k$ we use the Fréchet differential as follows:

$$\mathcal{G}'|_{\dot{\mathbf{U}}} \triangleq \lim_{\epsilon\to 0}\frac{\mathcal{G}(\dot{\mathbf{U}}+\epsilon\dot{\mathbf{U}}',\dot{\mathbf{Y}},\lambda)-\mathcal{G}(\dot{\mathbf{U}},\dot{\mathbf{Y}},\lambda)}{\epsilon}.$$

Using the above definition we have

$$\begin{aligned}\mathcal{G}'|_{\dot{\mathbf{u}}_k} = &2\left\langle \dot{\mathbf{u}}',\dot{\mathbf{u}}_j\right\rangle \left(\mathbf{y}_k^T\mathbf{y}_j\right) + 2\left\langle \dot{\mathbf{u}}',\mathbf{u}_j\right\rangle \left(\mathbf{y}_k^T\dot{\mathbf{y}}_j\right) - 2\left\langle \dot{\mathbf{u}}',\mathcal{L}(\mathbf{u}_j)\right\rangle \left(\mathbf{y}_k^T\mathbf{y}_j\right)\\ &- 2\left\langle \dot{\mathbf{u}}',\mathbf{F}'\mathbf{y}_k\right\rangle + \lambda_{jk}\left\langle \dot{\mathbf{u}}',\mathbf{u}_j\right\rangle = 0.\end{aligned}$$

The above equation can be written as $\langle \dot{\mathbf{u}}', \nabla_{\dot{\mathbf{u}}_k} \mathcal{G} \rangle$, and we observe that for any arbitrary direction $\dot{\mathbf{u}}'$, we must satisfy $\nabla_{\dot{\mathbf{u}}_k} \mathcal{G} = \mathbf{0}$. This leads to the following condition:

$$\nabla_{\dot{\mathbf{u}}_k} \mathcal{G} = 2\dot{\mathbf{u}}_j \left(\mathbf{y}_k^T \mathbf{y}_j\right) + 2\mathbf{u}_j \left(\mathbf{y}_k^T \dot{\mathbf{y}}_j\right) - 2\mathcal{L}(\mathbf{u}_j) \left(\mathbf{y}_k^T \mathbf{y}_j\right) - 2\mathbf{F}' \mathbf{y}_k + \lambda_{jk} \mathbf{u}_j = \mathbf{0}. \tag{A.1}$$

To eliminate λ_{jk}, we take the inner product of $\mathbf{u}_l$ with (A.1) to obtain

$$\begin{aligned} \langle \mathbf{u}_l, \nabla_{\dot{\mathbf{u}}_k} \mathcal{G} \rangle = {} & 2\phi_{lj} (\mathbf{y}_k^T \mathbf{y}_j) + 2\delta_{lj} \left(\mathbf{y}_k^T \dot{\mathbf{y}}_j\right) - 2\langle \mathbf{u}_l, \mathcal{L}(\mathbf{u}_j) \rangle \left(\mathbf{y}_k^T \mathbf{y}_j\right) \\ & - 2\langle \mathbf{u}_l, \mathbf{F}' \mathbf{y}_k \rangle + \lambda_{jk} \delta_{lj} = 0, \end{aligned}$$

where we have used $\langle \mathbf{u}_l, \dot{\mathbf{u}}_j \rangle = \phi_{lj}$ and $\langle \mathbf{u}_l, \mathbf{u}_j \rangle = \delta_{lj}$. Rearranging for λ_{lk} gives

$$\lambda_{lk} = 2 \left[-\phi_{lj} (\mathbf{y}_k^T \mathbf{y}_j) - \left(\mathbf{y}_k^T \dot{\mathbf{y}}_l\right) + \langle \mathbf{u}_l, \mathcal{L}(\mathbf{u}_j) \rangle \left(\mathbf{y}_k^T \mathbf{y}_j\right) + \langle \mathbf{u}_l, \mathbf{F}' \mathbf{y}_k \rangle \right].$$

Dividing (A.1) by 2 and substituting λ_{lk} gives

$$\left[\dot{\mathbf{u}}_j - \mathcal{L}(\mathbf{u}_j) + \langle \mathbf{u}_l, \mathcal{L}(\mathbf{u}_j) \rangle \mathbf{u}_l - \phi_{lj} \mathbf{u}_l\right] \left(\mathbf{y}_k^T \mathbf{y}_j\right) - \mathbf{F}' \mathbf{y}_k + \langle \mathbf{u}_l, \mathbf{F}' \mathbf{y}_k \rangle \mathbf{u}_l = \mathbf{0}.$$

Rearranging the above equation for $\dot{\mathbf{u}}_j$ we get

$$\boxed{\dot{\mathbf{u}}_j = \mathcal{L}(\mathbf{u}_j) - \langle \mathbf{u}_l, \mathcal{L}(\mathbf{u}_j) \rangle \mathbf{u}_l + \left[\mathbf{F}' \mathbf{y}_k - \langle \mathbf{u}_l, \mathbf{F}' \mathbf{y}_k \rangle \mathbf{u}_l\right] C_{kj}^{-1} + \phi_{lj} \mathbf{u}_l,}$$

where $C_{kj} = \mathbf{y}_k^T \mathbf{y}_j$. Similarly, the first order optimality condition of $\mathcal{G}$ with respect to $\dot{\mathbf{y}}_k$ requires that

$$\frac{\partial \mathcal{G}}{\partial \dot{\mathbf{y}}_k} = \langle \mathbf{u}_k, \mathbf{u}_j \rangle \dot{\mathbf{y}}_j + \langle \dot{\mathbf{u}}_j, \mathbf{u}_k \rangle \mathbf{y}_j - \langle \mathbf{u}_k, \mathcal{L}(\mathbf{u}_j) \rangle \mathbf{y}_j - \langle \mathbf{F}', \mathbf{u}_k \rangle = \mathbf{0}.$$

Again, we use $\langle \mathbf{u}_k, \mathbf{u}_j \rangle = \delta_{kj}$ and $\langle \dot{\mathbf{u}}_j, \mathbf{u}_k \rangle = -\phi_{jk}$. Rearranging for $\dot{\mathbf{y}}_k$ gives

$$\boxed{\dot{\mathbf{y}}_k = \langle \mathbf{u}_k, \mathcal{L}(\mathbf{u}_j) \rangle \mathbf{y}_j + \langle \mathbf{F}', \mathbf{u}_k \rangle + \phi_{jk} \mathbf{y}_j.}$$

Appendix B. Equivalence of reductions.

Proof. We prove the equivalence by using the evolution equation for $\mathbf{U}, \mathbf{Y}$ and using the matrix differential equation for the rotation matrix $\mathbf{R}$ and recovering the evolution equations for $\tilde{\mathbf{U}}, \tilde{\mathbf{Y}}$. To this end, we substitute $\mathbf{U} = \tilde{\mathbf{U}}\mathbf{R}$ and $\mathbf{Y} = \tilde{\mathbf{Y}}\mathbf{R}$ into the quasimatrix form of (2.7) and (2.8). The evolution equation for the orthonormal modes becomes

$$\begin{aligned} \dot{\mathbf{U}} &= \dot{\tilde{\mathbf{U}}}\mathbf{R} + \tilde{\mathbf{U}}\dot{\mathbf{R}} \\ &= \mathcal{L}(\tilde{\mathbf{U}})\mathbf{R} - \tilde{\mathbf{U}}\mathbf{R}\langle \tilde{\mathbf{U}}\mathbf{R}, \mathcal{L}(\tilde{\mathbf{U}})\mathbf{R} \rangle + [\mathbf{F}'\tilde{\mathbf{Y}}\mathbf{R} - \tilde{\mathbf{U}}\mathbf{R}\langle \tilde{\mathbf{U}}\mathbf{R}, \mathbf{F}'\tilde{\mathbf{Y}}\mathbf{R} \rangle] + \tilde{\mathbf{U}}\mathbf{R}\mathbf{\Phi}. \end{aligned}$$

Substituting $\dot{\mathbf{R}} = \mathbf{R}\mathbf{\Phi} - \tilde{\mathbf{\Phi}}\mathbf{R}$ and solving for $\dot{\tilde{\mathbf{U}}}$ yields

$$\begin{aligned} \dot{\tilde{\mathbf{U}}} = {} & \left[\mathcal{L}(\tilde{\mathbf{U}})\mathbf{R} - \tilde{\mathbf{U}}\mathbf{R}\langle \tilde{\mathbf{U}}\mathbf{R}, \mathcal{L}(\tilde{\mathbf{U}})\mathbf{R} \rangle + [\mathbf{F}'\tilde{\mathbf{Y}}\mathbf{R} - \tilde{\mathbf{U}}\mathbf{R}\langle \tilde{\mathbf{U}}\mathbf{R}, \mathbf{F}'\tilde{\mathbf{Y}}\mathbf{R} \rangle] \right. \\ & \left. + \tilde{\mathbf{U}}\mathbf{R}\mathbf{\Phi} - \tilde{\mathbf{U}}[\mathbf{R}\mathbf{\Phi} - \tilde{\mathbf{\Phi}}\mathbf{R}]\mathbf{R}^T. \right. \end{aligned}$$

Simplifying the above equation and using $\langle \tilde{\mathbf{U}}\mathbf{R}, \cdot \rangle = \mathbf{R}^T \langle \tilde{\mathbf{U}}, \cdot \rangle$ and $\mathbf{R}^{-1} = \mathbf{R}^T$, since $\mathbf{R}$ is an orthonormal matrix, results in

$$\dot{\tilde{\mathbf{U}}} = \mathcal{L}(\tilde{\mathbf{U}}) - \tilde{\mathbf{U}}\langle \tilde{\mathbf{U}}, \mathcal{L}(\tilde{\mathbf{U}}) \rangle + [\mathbf{F}'\tilde{\mathbf{Y}} - \tilde{\mathbf{U}}\langle \tilde{\mathbf{U}}, \mathbf{F}'\tilde{\mathbf{Y}} \rangle]\tilde{\mathbf{C}}^{-1} + \tilde{\mathbf{U}}\tilde{\mathbf{\Phi}},$$

where $\tilde{\mathbf{C}} = \mathbf{R}\mathbf{C}\mathbf{R}^T$ and $\tilde{\mathbf{C}}^{-1} = \mathbf{R}\mathbf{C}^{-1}\mathbf{R}^T$, where $\mathbf{C}$ and $\tilde{\mathbf{C}}$ are similar matrices and thus have the same eigenvalues. Following a similar procedure, the evolution equation for the coefficients becomes

$$\begin{aligned}\dot{\mathbf{Y}} &= \dot{\tilde{\mathbf{Y}}}\mathbf{R} + \tilde{\mathbf{Y}}\dot{\mathbf{R}} \\ &= \tilde{\mathbf{Y}}\mathbf{R}\langle\mathcal{L}(\tilde{\mathbf{U}})\mathbf{R}, \tilde{\mathbf{U}}\rangle\mathbf{R} + \langle\mathbf{F}', \tilde{\mathbf{U}}\rangle\mathbf{R} + \tilde{\mathbf{Y}}\mathbf{R}\boldsymbol{\Phi}.\end{aligned}$$

Substituting $\dot{\mathbf{R}} = \mathbf{R}\boldsymbol{\Phi} - \tilde{\boldsymbol{\Phi}}\mathbf{R}$ and solving for $\dot{\tilde{\mathbf{Y}}}$ yields

$$\begin{aligned}\dot{\tilde{\mathbf{Y}}} &= \big[\tilde{\mathbf{Y}}\mathbf{R}\mathbf{R}^T\langle\mathcal{L}(\tilde{\mathbf{U}}), \tilde{\mathbf{U}}\rangle\mathbf{R} + \langle\mathbf{F}', \tilde{\mathbf{U}}\rangle\mathbf{R} + \tilde{\mathbf{Y}}\mathbf{R}\boldsymbol{\Phi} - \tilde{\mathbf{Y}}[\mathbf{R}\boldsymbol{\Phi} - \tilde{\boldsymbol{\Phi}}\mathbf{R}]\big]\mathbf{R}^T \\ &= \tilde{\mathbf{Y}}\langle\mathcal{L}(\tilde{\mathbf{U}}), \tilde{\mathbf{U}}\rangle + \langle\mathbf{F}', \tilde{\mathbf{U}}\rangle + \tilde{\mathbf{Y}}\tilde{\boldsymbol{\Phi}}.\end{aligned}$$

Thus, we have shown that the evolutions of $\{\mathbf{U}(x,t), \mathbf{Y}(t)\}$ and $\{\tilde{\mathbf{U}}(x,t), \tilde{\mathbf{Y}}(t)\}$ according to (2.7) and (2.8) are equivalent. ∎

Appendix C. Reactive source term specification.

TABLE 1

Reactive source terms with species concentration denoted by $[\cdot]$. *Each* $\mathbf{s}_i$ *is scaled by* 10^2 *for time scale adjustment with the flow and the parameter values are assigned as follows:* $\alpha_1 = 2.54 \times 10^{-2}$, $\alpha_2 = 160$, $\alpha_3 = 3.74 \times 10^{-5}$, $\alpha_4 = 0.449$, $\alpha_5 = 1.12 \times 10^5$, $\alpha_6 = 5.13 \times 10^{-4}$, $\alpha_7 = 2.36 \times 10^{-2}$, $\alpha_8 = 14.6$, $\alpha_9 = 6.24 \times 10^{-2}$, $\alpha_{10} = 140.5$, $\alpha_{11} = 3.93 \times 10^{-4}$, $\alpha_{12} = 2.36 \times 10^{-2}$, $\alpha_{13} = 14.6$, $\alpha_{14} = 5.523$, $\alpha_{15} = 160$, $\alpha_{16} = 8.01 \times 10^{-4}$, $\alpha_{17} = 1.11 \times 10^{-3}$, $\alpha_{18} = 3.105$, $\alpha_{19} = 1060$, $\alpha_{20} = 1.65 \times 10^{-3}$, $\alpha_{21} = 8.177$, $\alpha_{22} = 3160$, $\alpha_{23} = 3.456$, $\alpha_{24} = 2.50 \times 10^5$, $\alpha_{25} = 1.80 \times 10^{-5}$, $\alpha_{26} = 50$, $\alpha_{27} = 3.70 \times 10^{-6}$, $\alpha_{28} = 3.00 \times 10^{-8}$, $\alpha_{29} = 9.01 \times 10^{-2}$, $\alpha_{30} = 3190$, $\alpha_{31} = 1.52 \times 10^{-9}$, $\alpha_{32} = 2.77 \times 10^{-2}$, $\alpha_{33} = 18$, *and* $\alpha_{34} = 2.22 \times 10^{-4}$.

$\mathbf{s}_1 = (\alpha_1[13][2])/(\alpha_2 + [2]) - \alpha_3[1][15]$
$\mathbf{s}_2 = -(\alpha_1[13][2]/(\alpha_2 + [2])$
$\mathbf{s}_3 = (\alpha_4[9][4]/(\alpha_5 + [4]) - \alpha_6[3] - (\alpha_7[17][3])/(\alpha_8 + [3])$
$\mathbf{s}_4 = (\alpha_4[9][4])/(\alpha_5 + [4])$
$\mathbf{s}_5 = (\alpha_9[9][6])/(\alpha_{10} + [6]) - \alpha_{11}[5] - (\alpha_{12}[17][5])/(\alpha_{13} + [5])$
$\mathbf{s}_6 = -(\alpha_9[9][6]/(\alpha_{10} + [6])$
$\mathbf{s}_7 = (\alpha_{14}[24][8])/(\alpha_{15} + [8]) - \alpha_{16}[7][15] - \alpha_{17}[16][7]$
$\mathbf{s}_8 = -(\alpha_{14}[24][8])/(\alpha_{15} + [8])$
$\mathbf{s}_9 = (\alpha_{18}[25][10])/(\alpha_{19} + [10]) - \alpha_{20}[9][15]$
$\mathbf{s}_{10} = -(\alpha_{18}[25][10])/(\alpha_{19} + [10])$
$\mathbf{s}_{11} = (\alpha_{21}[9][12])/(\alpha_{22} + [12]) - (\alpha_{23}[21][11])/(\alpha_{24} + [11])$
$\mathbf{s}_{12} = -(\alpha_{21}[9][12])/(\alpha_{22} + [12])$
$\mathbf{s}_{13} = (\alpha_{25}[9][14])/(\alpha_{26} + [14]) - \alpha_{27}[13][15] - \alpha_{28}[13][19]$
$\mathbf{s}_{14} = -(\alpha_{25}[9][14])/(\alpha_{26} + [14])$
$\mathbf{s}_{15} = -(\alpha_3[1] + \alpha_{16}[7] + \alpha_{20}[9] + \alpha_{27}[13])[15]$
$\mathbf{s}_{16} = -\alpha_{17}[16][7]$
$\mathbf{s}_{17} = (\alpha_{29}[9][18])/(\alpha_{30} + [18]) - \alpha_{31}[17][19]$
$\mathbf{s}_{18} = -(\alpha_{29}[9][18])/(\alpha_{30} + [18])$
$\mathbf{s}_{19} = -\alpha_{31}[17][19] - \alpha_{28}[13][19]$
$\mathbf{s}_{20} = 0$
$\mathbf{s}_{21} = (\alpha_{32}[20][22])/(\alpha_{33} + [22]) - \alpha_{34}[21][23]$
$\mathbf{s}_{22} = -(\alpha_{32}[20][22])/(\alpha_{33} + [22])$
$\mathbf{s}_{23} = -\alpha_{34}[21][23]$

Appendix D. f-OTD derivation for tensor sensitivities. We start by considering the third order quasitensor $\tilde{\mathbf{V}}' = [\tilde{\mathbf{v}}'_{ij}] \in \mathbb{R}^{\infty \times n_s \times n_r}$ that we seek to flatten into a quasimatrix $\mathbf{V}' = [\mathbf{v}'_m] \in \mathbb{R}^{\infty \times d}$. For ease of reference, we rewrite the tensor evolution equation (3.6) below:

$$\frac{\partial \tilde{\mathbf{v}}'_{ij}}{\partial t} + (\mathbf{u} \cdot \nabla)\, \tilde{\mathbf{v}}'_{ij} = \tilde{\kappa}_{ik} \nabla^2 \tilde{\mathbf{v}}'_{kj} + \tilde{\mathcal{L}}_{\mathbf{s}_{ik}} \tilde{\mathbf{v}}'_{kj} + \tilde{\mathbf{s}}'_{ij},$$

where $i, k = 1, 2, \ldots, n_s$ and $j = 1, 2, \ldots, n_r$. We define the indices $m(i, j) = j + (i - 1)n_r$ and $n(i', j') = j' + (i' - 1)n_r$, where $i' = 1, 2, \ldots, n_s$ and $j' = 1, 2, \ldots, n_r$. In the above equation, the terms $\tilde{\mathbf{v}}'_{ij}$ and $\tilde{\mathbf{s}}'_{ij}$ are flattened by replacing the index pair ij with the single index m: $\mathbf{v}'_{m(i,j)} = \tilde{\mathbf{v}}'_{ij}$ and $\mathbf{s}'_{m(i,j)} = \tilde{\mathbf{s}}'_{ij}$. Next, we define a new diffusion coefficient matrix $\kappa_{mn} \in \mathbb{R}^{d \times d}$ such that the mth diagonal entry is equal to the diffusion coefficient of the ith species. That is, κ_{mn} is independent of parameter index j and remains constant across all sensitivities of a given species i. Finally, the linearized reactive source term is defined as $\mathcal{L}_{\mathbf{s}_{m(i,j)n(i',j')}} = \tilde{\mathcal{L}}_{\mathbf{s}_{ii'}} \delta_{jj'}$, where $\delta_{jj'}$ is the Kronecker delta and n is a dummy index corresponding to $\mathbf{v}'_n$. From this definition, $\delta_{jj'}$ results in nonzero contribution to the summation over n only for sensitivities with respect to parameter $j' = j$. Putting this all together, the above equation can be written as

$$\frac{\partial \mathbf{v}'_m}{\partial t} + (\mathbf{w} \cdot \nabla)\mathbf{v}'_m = \kappa_{mn} \nabla^2 \mathbf{v}'_n + \mathcal{L}_{\mathbf{s}_{mn}} \mathbf{v}'_n + \mathbf{s}'_m, \tag{D.1}$$

where $\mathcal{L}_{\mathbf{s}_{mn}} \mathbf{v}'_n$ should be interpreted as a matrix-vector multiplication for any (x_1, x_2) point in the physical space. As a result of the parametric dependence of the linear operator, (2.7) and (2.8) do not hold for the tensor flattened equation. Therefore, we must derive new evolution equations for the f-OTD modes and coefficients for tensor flattened quantities. Substituting the approximation $\mathbf{v}'_m = \sum_{i=1}^{r} \mathbf{u}_i Y_{mi}$ into the above equation, it is straightforward to show that the evolution equations for the f-OTD modes and coefficients are

$$\begin{aligned} \dot{\mathbf{u}}_i = &- \left[(\mathbf{w} \cdot \nabla)\mathbf{u}_i - \mathbf{u}_j \langle \mathbf{u}_j, (\mathbf{w} \cdot \nabla)\mathbf{u}_i \rangle\right] + \left[\nabla^2 \mathbf{u}_k - \mathbf{u}_j \langle \mathbf{u}_j, \nabla^2 \mathbf{u}_k \rangle\right] Y_{nk} \kappa_{mn} Y_{ml} C_{il}^{-1} \\ &+ \left[\mathcal{L}_{\mathbf{s}_{mn}} \mathbf{u}_k - \mathbf{u}_j \langle \mathbf{u}_j, \mathcal{L}_{\mathbf{s}_{mn}} \mathbf{u}_k \rangle\right] Y_{nk} Y_{ml} C_{il}^{-1} + \left[\mathbf{s}'_m - \mathbf{u}_j \langle \mathbf{u}_j, \mathbf{s}'_m \rangle\right] Y_{ml} C_{il}^{-1} \end{aligned} \tag{D.2}$$

and

$$\begin{aligned} \dot{Y}_{mj} = &- \langle \mathbf{u}_j, (\mathbf{w} \cdot \nabla)\mathbf{u}_i \rangle Y_{mi} + \langle \mathbf{u}_j, \nabla^2 \mathbf{u}_i \rangle Y_{ni} \kappa_{mn} \\ &+ \langle \mathbf{u}_j, \mathcal{L}_{\mathbf{s}_{mn}} \mathbf{u}_i \rangle Y_{ni} + \langle \mathbf{u}_j, \mathbf{s}'_m \rangle, \end{aligned} \tag{D.3}$$

where $\mathbf{Y} = [Y_{mi}]$ and the indices $m, n = 1, 2, \ldots, d$ and $i, j, k, l = 1, 2, \ldots, r$.

Acknowledgment. We would like to thank Joseph Derlaga for his support and intellectual insight throughout the process.

REFERENCES

[1] M. B. Giles and N. A. Pierce, *An introduction to the adjoint approach to design*, Flow Turbul. Combust., 65 (2000), pp. 393–415.

[2] M. D. Gunzburger, *Sensitivities, adjoints and flow optimization*, Internat. J. Numer. Methods Fluids, 31 (1999), pp. 53–78.

[3] T. R. Bewley, P. Moin, and R. Temam, *DNS-based predictive control of turbulence: An optimal benchmark for feedback algorithms*, J. Fluid Mech., 447 (2001), pp. 179–225.
[4] K. T. Doetsch and K. J. Fidkowski, *Combined entropy and output-based adjoint approach for mesh refinement and error estimation*, AIAA J., 57 (2019), pp. 3213–3230.
[5] G. Esposito and H. K. Chelliah, *Skeletal reaction models based on principal component analysis: Application to ethylene–air ignition, propagation, and extinction phenomena*, Combust. Flame, 158 (2011), pp. 477–489.
[6] Exascale Mathematics Working Group, *Applied Mathematics Research for Exascale Computing*, U.S. Department of Energy, Office of Science, Advanced Scientific Computing Research Program, 2014.
[7] J. C. Bennett, H. Abbasi, P. Bremer, R. Grout, A. Gyulassy, T. Jin, S. Klasky, H. Kolla, M. Parashar, V. Pascucci, P. Pebay, D. Thompson, H. Yu, F. Zhang, and J. Chen, *Combining in-situ and in-transit processing to enable extreme-scale scientific analysis*, in SC '12: Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis, 2012, pp. 1–9.
[8] H. Babaee and T. P. Sapsis, *A minimization principle for the description of modes associated with finite-time instabilities*, Proc. A, 472 (2016), 20150779.
[9] H. Babaee, M. Farazmand, G. Haller, and T. P. Sapsis, *Reduced-order description of transient instabilities and computation of finite-time Lyapunov exponents*, Chaos, 27 (2017), 063103.
[10] A. Blanchard, S. Mowlavi, and T. Sapsis, *Control of linear instabilities by dynamically consistent order reduction on optimally time-dependent modes*, Nonlinear Dynam., 95 (2019), pp. 2745–2764.
[11] M. Farazmand and T. P. Sapsis, *Dynamical indicators for the prediction of bursting phenomena in high-dimensional systems*, Phys. Rev. E, 94 (2016), 032212.
[12] M. Beneitez, Y. Duguet, P. Schlatter, and D. S. Henningson, *Edge manifold as a Lagrangian coherent structure in a high-dimensional state space*, Phys. Rev. Res., 2 (2020), 033258.
[13] T. P. Sapsis and P. F. J. Lermusiaux, *Dynamically orthogonal field equations for continuous stochastic dynamical systems*, Phys. D, 238 (2009), pp. 2347–2360.
[14] M. Cheng, T. Y. Hou, and Z. Zhang, *A dynamically bi-orthogonal method for time-dependent stochastic partial differential equations* I*: Derivation and algorithms*, J. Comput. Phys., 242 (2013), pp. 843–868.
[15] E. Musharbash, F. Nobile, and T. Zhou, *Error analysis of the dynamically orthogonal approximation of time dependent random pdes*, SIAM J. Sci. Comput., 37 (2015), pp. A776–A810.
[16] H. Babaee, M. Choi, T. P. Sapsis, and G. E. Karniadakis, *A robust bi-orthogonal/dynamically-orthogonal method using the covariance pseudo-inverse with application to stochastic flow problems*, J. Comput. Phys., 344 (2017), pp. 303–319.
[17] P. Patil and H. Babaee, *Real-time reduced-order modeling of stochastic partial differential equations via time-dependent subspaces*, J. Comput. Phys., 415 (2020), 109511.
[18] D. Ramezanian, A. G. Nouri, and H. Babaee, *On-the-fly reduced order modeling of passive and reactive species via time-dependent manifolds*, Comput. Methods Appl. Mech. Engrg., 382 (2021), 113882.
[19] A. G. Nouri, H. Babaee, P. Givi, H. K. Chelliah, and D. Livescu, *Skeletal model reduction with forced optimally time dependent modes*, Combust. Flame, in press.
[20] K. Braman, To. A. Oliver, and V. Raman, *Adjoint-based sensitivity analysis of flames*, Combust. Theory Model., 19 (2015), pp. 29–56.
[21] M. Lemke, L. Cai, J. Reiss, H. Pitsch, and J. Sesterhenn, *Adjoint-based sensitivity analysis of quantities of interest of complex combustion models*, Combust. Theory Model., 23 (2019), pp. 180–196.
[22] R. Langer, J. Lotz, L. Cai, F. vom Lehn, K. Leppkes, U. Naumann, and H. Pitsch, *Adjoint sensitivity analysis of kinetic, thermochemical, and transport data of nitrogen and ammonia chemistry*, Proc. Combust. Inst. 38, (2021), pp. 777–785.
[23] Z. Battles and L. Trefethen, *An extension of matlab to continuous functions and operators*, SIAM J. Sci. Comput., 25 (2004), pp. 1743–1770.
[24] H. Babaee, *An observation-driven time-dependent basis for a reduced description of transient stochastic systems*, Proc. A, 475 (2019), 20190506.
[25] N. Aubry, *On the hidden beauty of the proper orthogonal decomposition*, Theoret. Comput. Fluid Dyn. 2 (1991), pp. 339–352.
[26] L. Alvergue, H. Babaee, G. Gu, and S. Acharya, *Feedback stabilization of a reduced-order model of a jet in crossflow*, AIAA J., 53 (2015), pp. 2472–2481.
[27] P. Arrué, N. Toosizadeh, H. Babaee, and K. Laksari, *Low-rank representation of head impact kinematics: A data-driven emulator*, Front. Bioeng. Biotech., 8 (2020), 555493.

[28] P. J. Schmid, *Dynamic mode decomposition of numerical and experimental data*, J. Fluid Mech., 656 (2010), pp. 5–28.
[29] K. Laksari, M. Kurt, H. Babaee, S. Kleiven, and D. Camarillo, *Mechanistic insights into human brain impact dynamics through modal analysis*, Phys. Rev. Lett., 120 (2018), 138101.
[30] O. Koch and C. Lubich, *Dynamical low-rank approximation*, SIAM J. Matrix Anal. Appl., 29 (2007), pp. 434–454.
[31] A. J. Chorin, O. H. Hald, and R. Kupferman, *Optimal prediction with memory*, Phys. D, 166 (2002), pp. 239–257.
[32] D. Ruelle, *Differentiation of SRB states*, Comm. Math. Phys., 187 (1997), pp. 227–241.
[33] G. L. Eyink, T. W. N. Haine, and D. J. Lea, *Ruelle's linear response formula, ensemble adjoint schemes and Lévy flights*, Nonlinearity, 17 (2004), 1867.
[34] Q. Wang, R. Hu, and P. Blonigan, *Least squares shadowing sensitivity analysis of chaotic limit cycle oscillations*, J. Comput. Phys., 267 (2014), pp. 210–224.
[35] A. Garai and S. M. Murman, *Stabilization of the adjoint for turbulent flows*, AIAA J., 59 (2021), pp. 2001–2013.
[36] A.-K. Kassam and L. N. Trefethen, *Fourth-order time-stepping for stiff PDEs*, SIAM J. Sci. Comput., 26 (2005), pp. 1214–1233.
[37] Z. Li, A. Yazdani, A. Tartakovsky, and G. E. Karniadakis, *Transport dissipative particle dynamics model for mesoscopic advection-diffusion-reaction problems*, J. Chem. Phys., 143 (2015), 014101.
[38] G. E. Karniadakis and S. J. Sherwin, *Spectral/hp Element Methods for Computational Fluid Dynamics*, Oxford University Press, Oxford, 2005.
[39] H. Babaee, S. Acharya, and X. Wan, *Optimization of forcing parameters of film cooling effectiveness*, J. Turbomachinery, 136 (2014), 061016.
[40] H. Babaee, X. Wan, and S. Acharya, *Effect of uncertainty in blowing ratio on film cooling effectiveness*, J. Heat Transfer, 136 (2014), 031701.
[41] T. G. Kolda and B. W. Bader, *Tensor decompositions and applications*, SIAM Rev., 51 (2009), pp. 455–500.
[42] P. Benner, S. Gugercin, and K. Willcox, *A survey of projection-based model reduction methods for parametric dynamical systems*, SIAM Rev., 57 (2015), pp. 483–531.