Space-time galerkin pod with application in optimal control of semilinear partial differential equations

Baumann, Manuel; Benner, Peter; Heiland, Jan

DOI
10.1137/17M1135281

Publication date
2018

Document Version
Final published version

Published in
SIAM Journal on Scientific Computing

Citation (APA)
Baumann, M., Benner, P., & Heiland, J. (2018). Space-time galerkin pod with application in optimal control of semilinear partial differential equations. *SIAM Journal on Scientific Computing*, *40*(3), A1611-A1641. https://doi.org/10.1137/17M1135281

SIAM J. Sci. Comput. © 2018 Society for Industrial and Applied Mathematics
Vol. 40, No. 3, pp. A1611–A1641

SPACE-TIME GALERKIN POD WITH APPLICATION IN OPTIMAL CONTROL OF SEMILINEAR PARTIAL DIFFERENTIAL EQUATIONS[*]

MANUEL BAUMANN[†], PETER BENNER[‡], AND JAN HEILAND[‡]

Abstract. In the context of Galerkin discretizations of a partial differential equation (PDE), the modes of the classical method of proper orthogonal decomposition (POD) can be interpreted as the ansatz and trial functions of a low-dimensional Galerkin scheme. If one also considers a Galerkin method for the time integration, one can similarly define a POD reduction of the temporal component. This has been described earlier but not expanded upon—probably because the reduced time discretization globalizes time, which is computationally inefficient. However, in finite-time optimal control systems, time *is* a global variable and there is no disadvantage from using a POD reduced Galerkin scheme in time. In this paper, we provide a newly developed generalized theory for space-time Galerkin POD, prove its optimality in the relevant function spaces, show its application for the optimal control of nonlinear PDEs, and, by means of a numerical example with Burgers' equation, discuss the competitiveness by comparing to standard approaches.

Key words. model reduction, low-rank Galerkin, nonlinear PDE, optimal control

AMS subject classifications. 35Q93, 65L60, 49M25, 65M22

DOI. 10.1137/17M1135281

1. Introduction.

The method of *proper orthogonal decomposition* (POD) is a standard model reduction tool. For a generic dynamical system

$$\dot{v} = f(t, v), \tag{1}$$

on the time interval $(0, T]$ with a solution v with $v(t) \in \mathbb{R}^N$ and using samples $v(t_j)$, POD provides a set of $\hat{n}$ so-called POD modes $\hat{v}_1, \ldots, \hat{v}_{\hat{n}} \in \mathbb{R}^N$ which optimally parametrize the solution trajectory. As a result, the system (1) can be projected down to a system of reduced spatial dimension $\hat{n}$ that often reflects the dynamical behavior of (1) well. If the considered system stems from a *finite element method* (FEM) discretization of a PDE, then the modes $\hat{v}_i$, $i = 1, \ldots, \hat{n}$, can be interpreted as ansatz functions in the finite element space $\mathcal{Y}$ and the projected system as a particular Galerkin projection of the underlying PDE.

In this paper we provide a theoretical framework and show cases for a space-time Galerkin POD method. Some of the underlying ideas for this generalization of POD have been developed and tested in our earlier works [2, 3].

The first innovation of the proposed generalized POD approach is based on the observation that instead of the discrete time samples $v(t_j)$, one may use the projection of v onto the finite dimensional subspace $\mathcal{S} \cdot \mathcal{Y}$, where $\mathcal{S}$ is a, say, k-dimensional subspace of $L^2(0, T)$. The second innovation is that the projection onto $\mathcal{S} \cdot \mathcal{Y}$ can be interpreted as Galerkin discretization in time which can be reduced analoguously to the POD reduction of the space dimension. The resulting scheme is a POD reduced space-time Galerkin discretization.

[*] Submitted to the journal's Methods and Algorithms for Scientific Computing section June 19, 2017; accepted for publication (in revised form) April 9, 2018; published electronically June 5, 2018.
http://www.siam.org/journals/sisc/40-3/M113528.html

[†] Delft Institute of Applied Mathematics, Delft, 2628, Netherlands (m.m.baumann@tudelft.nl).

[‡] Max Planck Institute for Dynamics of Complex Technical Systems, Magdeburg, D-39106, Germany (benner@mpi-magdeburg.mpg.de, heiland@mpi-magdeburg.mpg.de).

This basic idea of a space-time POD has been taken up in [24] but has not progressed since then. We think that this is due to the fact that temporal POD destructs the causality in time, which makes it inefficient for numerical simulations. In fact, the POD reduced time ansatz functions are global such that the space-time Galerkin system has to be solved as a whole rather than in sequences of time stretches as in standard time-stepping or discontinuous Galerkin schemes [17, 22]. Thus, the reduced space-time scheme is likely outperformed by a standard spatial POD combined with a standard Runge–Kutta solver. However, in finite-time optimal control problems, the time *is* a global variable and, as we will show by numerical examples, the space-time Galerkin discretization becomes very competitive.

The need and the potential of also reducing the time dimension of a reduced order model have been discussed in [6]. There—similar to our observation that an SVD of a matrix of measurements also reveals compressed time information—it is proposed to use the right singular vectors of a classical snapshot matrix for forecasting.

We want to point out that the method of *proper generalized decomposition* (PGD) is related to the proposed space-time Galerkin POD only insofar as for PGD also space-time (and parameter) tensor bases are used for the modeling; see, e.g., [8]. However, the PGD approach seeks to successively build up the bases by collocation, *greedy algorithms*, and fixed-point iteration, whereas our approach reduces a given basis on the base of measurements. For the same reasons, the connection of the presented approach to other tensor-based low-dimensional approximation schemes [13, 19] as well as to *reduced basis* approaches [25] is only marginal.

This paper is organized as follows. First, we introduce the mathematical framework and rigorously prove the optimality of the reduced space and time bases. Then we illustrate how the reduced bases can be used for low-dimensional space-time Galerkin approximations. In particular, we address how to treat quadratic nonlinearities, how to incorporate initial and terminal values, and how to set up the bases for a general PDE by means of standard approximation schemes. Finally, we illustrate the performance of the space-time Galerkin POD approach for the optimal control of Burgers' equation and compare it to well-established gradient-based methods combined with standard POD.

2. Space-time Galerkin POD.

In this section, we provide the analytical framework for space-time POD. We introduce the considered function spaces and directly prove the optimality of the POD projection in the respective space-time L^2 norm. For a time interval $(0,T)$ and a spatial domain Ω, consider the space-time function space $L^2(0,T;L^2(\Omega))$. Let

$$\mathcal{S} = \operatorname{span}\{\psi_1,\dots,\psi_s\} \subset L^2(0,T) \quad \text{and} \quad \mathcal{Y} = \operatorname{span}\{\nu_1,\dots,\nu_q\} \subset L^2(\Omega)$$

be finite dimensional subspaces of dimension s and q, respectively, and let

$$\mathcal{X} = \mathcal{S}\cdot\mathcal{Y} \subset L^2(0,T;L^2(\Omega)) \tag{2}$$

be the product space of $\mathcal{S}$ and $\mathcal{Y}$ spanned by $\{\psi_j\cdot\nu_i\}_{j=1,\dots,s}^{i=1,\dots,q}$.

The space-time L^2-orthogonal projection $\bar{x} := \Pi_{\mathcal{S}\cdot\mathcal{Y}}x$ of some $x \in L^2(0,T;L^2(\Omega))$ onto $\mathcal{X}$ is given as

$$\bar{x}(\xi,\tau) = \sum_{j=1}^{s}\sum_{i=1}^{q}\mathbf{x}_{i,j}\nu_i(\xi)\psi_j(\tau), \tag{3}$$

where the coefficients $\mathbf{x}_{i,j}$ are the entries of the matrix

$$\mathbf{X} = \left[\mathbf{x}_{i,j}\right]_{i=1,\ldots,q}^{j=1,\ldots,s} := \mathbf{M}_{\mathcal{Y}}^{-1} \begin{bmatrix} ((x,\nu_1\psi_1))_{\mathcal{S}\cdot\mathcal{Y}} & \ldots & ((x,\nu_1\psi_s))_{\mathcal{S}\cdot\mathcal{Y}} \\ \vdots & \ddots & \vdots \\ ((x,\nu_q\psi_1))_{\mathcal{S}\cdot\mathcal{Y}} & \ldots & ((x,\nu_q\psi_s))_{\mathcal{S}\cdot\mathcal{Y}} \end{bmatrix} \mathbf{M}_{\mathcal{S}}^{-1}, \tag{4}$$

where

$$((x,\nu_i\psi_j))_{\mathcal{S}\cdot\mathcal{Y}} := ((x,\nu_i)_{\mathcal{Y}},\psi_j)_{\mathcal{S}} := \int_0^T \left(\int_\Omega x(\xi,\tau)\nu_i(\xi)\ \mathrm{d}\xi\right)\psi_j(\tau)\ \mathrm{d}\tau.$$

Here, $\mathbf{M}_{\mathcal{Y}}^{-1}$ and $\mathbf{M}_{\mathcal{S}}^{-1}$ are the inverses of the mass matrices with respect to space and time,

$$\mathbf{M}_{\mathcal{Y}} := \left[(\nu_i,\nu_j)_{\mathcal{Y}}\right]_{i=1,\ldots,q}^{j=1,\ldots,q} \quad \text{and} \quad \mathbf{M}_{\mathcal{S}} := \left[(\psi_i,\psi_j)_{\mathcal{S}}\right]_{i=1,\ldots,s}^{j=1,\ldots,s}. \tag{5}$$

Remark 2.1. We will refer to $\mathcal{X} = \mathcal{S}\cdot\mathcal{Y}$ as the measurement space, to the basis functions of $\mathcal{Y}$ and $\mathcal{S}$ as measurement functions, and to $\mathbf{X}$ as the measurement matrix. This means that a function in $L^2(0,T;L^2(\Omega))$ can be measured in $\mathcal{X}$, e.g., via its projection onto $\mathcal{X}$, and, the other way around, an element $\mathbf{X}$ of $\mathcal{X}$ can be seen as a measurement of some functions in $L^2(0,T;L^2(\Omega))$.

We introduce some representations of the inner product and the norm of functions in $\mathcal{S}\cdot\mathcal{Y}$.

LEMMA 2.2 (space-time discrete L^2-product). *Let*

$$x^1 = \sum_{j=1}^{s}\sum_{i=1}^{q} \mathbf{x}_{i,j}^1 \nu_i\psi_j \in \mathcal{S}\cdot\mathcal{Y}, \quad x^2 = \sum_{j=1}^{s}\sum_{i=1}^{q} \mathbf{x}_{i,j}^2 \nu_i\psi_j \in \mathcal{S}\cdot\mathcal{Y};$$

then, with

$$\mathbf{x}^\ell = [\mathbf{x}_{1,1}^\ell,\ldots,\mathbf{x}_{q,1}^\ell,\mathbf{x}_{1,2}^\ell,\ldots,\mathbf{x}_{q,2}^\ell,\ldots,\mathbf{x}_{1,s}^\ell,\ldots,\mathbf{x}_{q,s}^\ell]^\mathsf{T} =: \operatorname{vec}(\mathbf{X}^\ell), \quad \ell = 1,2,$$

the inner product in $\mathcal{S}\cdot\mathcal{Y}$ is given as

$$((x^1,x^2))_{\mathcal{S}\cdot\mathcal{Y}} = \int_0^T\int_\Omega x^1x^2\ \mathrm{d}\xi\ \mathrm{d}\tau = (\mathbf{x}^1)^\mathsf{T}\left(\mathbf{M}_{\mathcal{S}}\otimes\mathbf{M}_{\mathcal{Y}}\right)\mathbf{x}^2 \tag{6}$$

and the induced norm as

$$\|x^\ell\|_{\mathcal{S}\cdot\mathcal{Y}}^2 := ((x^\ell,x^\ell))_{\mathcal{S}\cdot\mathcal{Y}} = \|\mathbf{x}^\ell\|_{\mathbf{M}_{\mathcal{S}}\otimes\mathbf{M}_{\mathcal{Y}}}^2 = \|\mathbf{M}_{\mathcal{Y}}^{1/2}\mathbf{X}^\ell\mathbf{M}_{\mathcal{S}}^{1/2}\|_F^2, \quad \ell = 1,2, \tag{7}$$

where $\|\cdot\|_{\mathbf{M}_{\mathcal{S}}\otimes\mathbf{M}_{\mathcal{Y}}}$ denotes the Euclidean vector norm weighted by $\mathbf{M}_{\mathcal{S}}\otimes\mathbf{M}_{\mathcal{Y}}$, and $\|\cdot\|_F$ is the Frobenius norm.

Proof. The lemma is proved with straightforward calculations. □

Remark 2.3. In practical applications, one uses a *Cholesky factorization* of the mass matrices (5) rather than the square root. If memory is an issue, as it likely is for large-scale three-dimensional (3D) problems, one can also resort to sparse Cholesky factorizations that limit try to minimize the so-called fill-in; see, e.g., [9].

COROLLARY 2.4. *Let $\mathbf{M}_{\mathcal{S}} = \mathbf{L}_{\mathcal{S}}\mathbf{L}_{\mathcal{S}}^\mathsf{T}$ and $\mathbf{M}_{\mathcal{Y}} = \mathbf{L}_{\mathcal{Y}}\mathbf{L}_{\mathcal{Y}}^\mathsf{T}$ be given in factored form. Then, for a given $x \in \mathcal{S}\cdot\mathcal{Y}$ with its coefficient matrix $\mathbf{X}$ and vector $\mathbf{x} = \operatorname{vec}(\mathbf{X})$ it holds that*

$$\|x\|_{\mathcal{S}\cdot\mathcal{Y}}^2 = \|\mathbf{x}\|_{\mathbf{M}_{\mathcal{S}}\otimes\mathbf{M}_{\mathcal{Y}}}^2 = \|\mathbf{L}_{\mathcal{Y}}^\mathsf{T}\mathbf{X}\mathbf{L}_{\mathcal{S}}\|_F^2.$$

Proof.

$$\begin{aligned}\|\mathbf{x}\|^2_{\mathbf{M}_\mathcal{S}\otimes\mathbf{M}_\mathcal{Y}} &= \mathbf{x}^\mathsf{T}(\mathbf{M}_\mathcal{S}\otimes\mathbf{M}_\mathcal{Y})\mathbf{x} = \mathbf{x}^\mathsf{T}(\mathbf{L}_\mathcal{S}\otimes\mathbf{L}_\mathcal{Y})\cdot(\mathbf{L}_\mathcal{S}^\mathsf{T}\otimes\mathbf{L}_\mathcal{Y}^\mathsf{T})\mathbf{x}\\ &= \|(\mathbf{L}_\mathcal{S}^\mathsf{T}\otimes\mathbf{L}_\mathcal{Y}^\mathsf{T})\mathbf{x}\|_2^2 = \|\mathrm{vec}(\mathbf{L}_\mathcal{Y}^\mathsf{T}\mathbf{X}\mathbf{L}_\mathcal{S})\|_2^2 = \|\mathbf{L}_\mathcal{Y}^\mathsf{T}\mathbf{X}\mathbf{L}_\mathcal{S}\|_F^2,\end{aligned}$$

as it follows from basic properties and relations between the Kronecker product, the vectorization operator, and the Frobenius norm. □

From now on, we will always consider the factorized form. In theory, one can always replace the factors by the square roots of the respective mass matrices.

Next, we will consider a given function $x \in \mathcal{S}\cdot\mathcal{Y}$ and determine low-dimensional subspaces of $\mathcal{Y}$ and $\mathcal{S}$ that can provide low-dimensional approximations to x in a norm-optimal way.

LEMMA 2.5 (optimal low-rank bases in space). *Given $x \in \mathcal{S}\cdot\mathcal{Y}$ and the associated matrix of coefficients $\mathbf{X}$. The best-approximating $\hat{q}$-dimensional subspace $\hat{\mathcal{Y}}$ in the sense that the projection error $\|x - \Pi_{\mathcal{S}\cdot\hat{\mathcal{Y}}}x\|_{\mathcal{S}\cdot\mathcal{Y}}$ is minimal over all subspaces of $\mathcal{Y}$ of dimension $\hat{q}$ is given as* $\mathrm{span}\{\hat{\nu}_i\}_{i=1,\dots,\hat{q}}$, *where*

$$\begin{bmatrix}\hat{\nu}_1\\ \hat{\nu}_2\\ \vdots\\ \hat{\nu}_{\hat{q}}\end{bmatrix} = V_{\hat{q}}^\mathsf{T}\mathbf{L}_\mathcal{Y}^{-1}\begin{bmatrix}\nu_1\\ \nu_2\\ \vdots\\ \nu_q\end{bmatrix}, \tag{8}$$

where $V_{\hat{q}}$ is the matrix of the $\hat{q}$ leading left singular vectors of the matrix

$$\mathbf{L}_\mathcal{Y}^\mathsf{T}\mathbf{X}\mathbf{L}_\mathcal{S}.$$

Remark 2.6. Here and in what follows, we use column vectors of the scalar basis functions like $[\nu_1 \ \cdots \ \nu_q]^\mathsf{T}$ as they can be formally treated like a vector and, thus, simplify the notation. We illustrate the use of the formal vectors of functions in the following proof that the reduced bases functions are mutually orthogonal, i.e., that the mass matrix of $\hat{\mathcal{Y}}$ is the identity

$$\begin{aligned}M_{\hat{\mathcal{Y}}} &= \int_\Omega \begin{bmatrix}\hat{\nu}_1(\xi)\\ \vdots\\ \hat{\nu}_{\hat{q}}(\xi)\end{bmatrix}\begin{bmatrix}\hat{\nu}_1(\xi) & \cdots & \hat{\nu}_{\hat{q}}(\xi)\end{bmatrix}\,\mathrm{d}\xi\\ &= V_{\hat{q}}^\mathsf{T}\mathbf{L}_\mathcal{Y}^{-1}\int_\Omega \begin{bmatrix}\nu_1(\xi)\\ \vdots\\ \nu_q(\xi)\end{bmatrix}\begin{bmatrix}\nu_1(\xi) & \cdots & \nu_q(\xi)\end{bmatrix}\,\mathrm{d}\xi\mathbf{L}_\mathcal{Y}^{-\mathsf{T}}V_{\hat{q}} = V_{\hat{q}}^\mathsf{T}\mathbf{L}_\mathcal{Y}^{-1}M_\mathcal{Y}\mathbf{L}_\mathcal{Y}^{-\mathsf{T}}V_{\hat{q}} = I,\end{aligned}$$

where we have used that the singular vectors that constitute $V_{\hat{q}}$ are orthogonal.

Proof of Lemma 2.5. For the time dimension at fixed index j, we consider

$$y := \sum_{i=1}^q \mathbf{x}_{i,j}\nu_i = \begin{bmatrix}\mathbf{x}_{1,j} & \dots & \mathbf{x}_{q,j}\end{bmatrix}\begin{bmatrix}\nu_1\\ \vdots\\ \nu_q\end{bmatrix} \in \mathcal{Y}.$$

Next, we determine the orthogonal projection of y onto $\hat{\mathcal{Y}}$. Therefore, we write y

as a function in $\hat{\mathcal{Y}}$ and a remainder $\hat{R}$ in the orthogonal complement:

$$y = \begin{bmatrix} \mathbf{x}_{1,j} & \dots & \mathbf{x}_{q,j} \end{bmatrix} \begin{bmatrix} \nu_1 \\ \vdots \\ \nu_q \end{bmatrix} = \begin{bmatrix} \beta_1 & \dots & \beta_{\hat{q}} \end{bmatrix} \begin{bmatrix} \hat{\nu}_1 \\ \vdots \\ \hat{\nu}_{\hat{q}} \end{bmatrix} + \hat{R}.$$

We determine the coefficients β_k, $k = 1, \dots, \hat{q}$, by testing against the basis functions of $\hat{\mathcal{Y}}$. By mutual orthogonality of $\hat{\nu}_i$, $i = 1, \dots, \hat{q}$ (see Remark 2.6), and their orthogonality against $\hat{R}$, it follows that

$$\begin{aligned} \beta_k = \left(\sum_{i=1}^{\hat{q}} \beta_i \hat{\nu}_i, \hat{\nu}_k \right)_{\mathcal{Y}} = \left(\hat{R} + \sum_{i=1}^{\hat{q}} \beta_i \hat{\nu}_i, \hat{\nu}_k \right)_{\mathcal{Y}} &= \left(\sum_{i=1}^{q} \mathbf{x}_{i,j} \nu_i, \hat{\nu}_k \right)_{\mathcal{Y}} \\ &\overset{(*)}{=} \begin{bmatrix} \mathbf{x}_{i,j} & \dots & \mathbf{x}_{q,j} \end{bmatrix} \mathbf{M}_{\mathcal{Y}} \mathbf{L}_{\mathcal{Y}}^{-\mathsf{T}} V_{\hat{q},k} \\ &= \begin{bmatrix} \mathbf{x}_{i,j} & \dots & \mathbf{x}_{q,j} \end{bmatrix} \mathbf{L}_{\mathcal{Y}} V_{\hat{q},k}, \end{aligned}$$

where in $\overset{(*)}{=}$ we have used that $\hat{\nu}_k = \begin{bmatrix} \nu_1 & \dots & \nu_q \end{bmatrix} \mathbf{L}_{\mathcal{Y}}^{-\mathsf{T}} V_{\hat{q},k}$ and where $V_{\hat{q},k}$ is the kth column of $V_{\hat{q}}$ in (8). Thus, we find that the coefficients of the orthogonal projection of y onto $\hat{\mathcal{Y}}$ in the bases of $\hat{\mathcal{Y}}$ and $\mathcal{Y}$ are given through

$$\begin{aligned} \hat{y} = \sum_{i=1}^{\hat{q}} \beta_i \hat{\nu}_i = \begin{bmatrix} \beta_1 & \dots & \beta_{\hat{q}} \end{bmatrix} \begin{bmatrix} \hat{\nu}_1 \\ \vdots \\ \hat{\nu}_{\hat{q}} \end{bmatrix} &= \begin{bmatrix} \mathbf{x}_{1,j} & \dots & \mathbf{x}_{q,j} \end{bmatrix} \mathbf{L}_{\mathcal{Y}} V_{\hat{q}} \begin{bmatrix} \hat{\nu}_1 \\ \vdots \\ \hat{\nu}_q \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{x}_{1,j} & \dots & \mathbf{x}_{q,j} \end{bmatrix} \mathbf{L}_{\mathcal{Y}} V_{\hat{q}} V_{\hat{q}}^{\mathsf{T}} \mathbf{L}_{\mathcal{Y}}^{-1} \begin{bmatrix} \nu_1 \\ \vdots \\ \nu_q \end{bmatrix} \\ &=: \begin{bmatrix} \hat{\mathbf{x}}_{1,j} & \dots & \hat{\mathbf{x}}_{q,j} \end{bmatrix} \begin{bmatrix} \nu_1 \\ \vdots \\ \nu_q \end{bmatrix}. \end{aligned}$$

Noting that $\begin{bmatrix} \mathbf{x}_{1,j} & \dots & \mathbf{x}_{q,j} \end{bmatrix}^{\mathsf{T}}$ makes up the jth column of the matrix $\mathbf{X}$ associated with x, we conclude that the matrix $\hat{\mathbf{X}}$ of coefficients associated with $\Pi_{\mathcal{S}\cdot\hat{\mathcal{Y}}} x$ is given as

$$\hat{\mathbf{X}} = \mathbf{L}_{\mathcal{Y}}^{-\mathsf{T}} V_{\hat{q}} V_{\hat{q}}^{\mathsf{T}} \mathbf{L}_{\mathcal{Y}}^{\mathsf{T}} \mathbf{X}$$

and, by Corollary 2.4, we have that

$$\begin{aligned} \|x - \Pi_{\mathcal{S}\cdot\hat{\mathcal{Y}}} x\|_{\mathcal{S}\cdot\mathcal{Y}} &= \|\mathbf{L}_{\mathcal{Y}}^{\mathsf{T}} \mathbf{X} \mathbf{L}_{\mathcal{S}} - \mathbf{L}_{\mathcal{Y}}^{\mathsf{T}} \hat{\mathbf{X}} \mathbf{L}_{\mathcal{S}}\|_F = \|\mathbf{L}_{\mathcal{Y}}^{\mathsf{T}} [\mathbf{X} - \hat{\mathbf{X}}] \mathbf{L}_{\mathcal{S}}\|_F \\ &= \|\mathbf{L}_{\mathcal{Y}}^{\mathsf{T}} \mathbf{X} \mathbf{L}_{\mathcal{S}} - V_{\hat{q}} V_{\hat{q}}^{\mathsf{T}} \mathbf{L}_{\mathcal{Y}}^{\mathsf{T}} \mathbf{X} \mathbf{L}_{\mathcal{S}}\|_F, \end{aligned}$$

which is minimized over all $V_{\hat{q}} \in \mathbb{R}^{q,\hat{q}}$ matrices by taking $V_{\hat{q}}$ as the matrix of the $\hat{q}$ leading left singular vectors of $\mathbf{L}_{\mathcal{Y}}^{\mathsf{T}} \mathbf{X} \mathbf{L}_{\mathcal{S}}$. □

The same arguments apply to the transpose of $\mathbf{X}$.

Lemma 2.7 (optimal low-rank bases in time). *Given $x \in \mathcal{S} \cdot \mathcal{Y}$ and the associated matrix of coefficients $\mathbf{X}$. The best-approximating $\hat{s}$-dimensional subspace $\hat{\mathcal{S}}$ in the sense that the projection error $\|x - \Pi_{\hat{\mathcal{S}}\cdot\mathcal{Y}} x\|_{\mathcal{S}\cdot\mathcal{Y}}$ is minimal over all subspaces of $\mathcal{S}$ of dimension $\hat{s}$ is given as* $\operatorname{span}\{\hat{\psi}_j\}_{j=1,\dots,\hat{s}}$, *where*

$$\begin{bmatrix}\hat{\psi}_1\\ \hat{\psi}_2\\ \vdots\\ \hat{\psi}_{\hat{s}}\end{bmatrix} = U_{\hat{s}}^{\mathsf{T}}\mathbf{L}_{\mathcal{S}}^{-1}\begin{bmatrix}\psi_1\\ \psi_2\\ \vdots\\ \psi_s\end{bmatrix}, \tag{9}$$

where $U_{\hat{s}}$ is the matrix of the $\hat{s}$ leading right singular vectors of

$$\mathbf{L}_{\mathcal{Y}}^{\mathsf{T}}\mathbf{X}\mathbf{L}_{\mathcal{S}}.$$

Remark 2.8. The approximation results Lemmas 2.5 and 2.7 hold in the space-time L^2 norm, which is the appropriate norm for the considered functions and which is not part of the standard POD approach. However, the need for the right norms has been accounted for through the use of weighted inner products or weighted sums. If one lets $\mathcal{S}$ degenerate to a set of Dirac deltas, then Lemma 2.5 reduces to the optimality result [23, Thm. 1.8] for the standard POD approximation in the sense that the inner product is weighted with the FEM mass matrix (see also Remark 4.1). If one chooses $\mathcal{S}$ such that the induced time Galerkin scheme resembles a time discretization by the trapezoidal rule (see [18, sect. 3.3]), then Lemma 2.5 reduces to the optimality conditions for the *continuous POD* approach given in [23, sect. 1.3]. In fact, certain (discontinuous) Galerkin schemes with certain choices of quadrature rules can attain many standard Runge–Kutta schemes [5].

Remark 2.9. The idea of generalized measurements also works as a generalization of POD for a reduction of the state space. Consider the dynamical system (1), and define $\mathbf{X}_{\mathcal{S}} := [(v_i, \psi_j)_{\mathcal{S}}]_{i=1,\dots,q}^{j=1,\dots,s}$, where v_i is the ith component of the vector-valued solution. Then the leading left singular vectors of the matrix $\mathbf{X}_{\mathcal{S}}\mathbf{L}_{\mathcal{S}}^{-1}$ are generalized POD modes and a projection of (1) onto the space spanned by those modes yields a POD-reduced dynamical system as we have previously described under the term *gmPOD* in [3].

3. Space-time Galerkin schemes.

In this section, we briefly describe how to formulate a general space-time Galerkin approximation to a generic PDE. This regression is then followed by the discussion of low-rank space-time Galerkin schemes on the basis of POD reductions of standard Galerkin bases.

Let $\{\hat{\psi}_1, \dots, \hat{\psi}_{\hat{s}}\} \subset H^1(0,T)$ and $\{\hat{\nu}_1, \dots, \hat{\nu}_{\hat{q}}\} \subset H_0^1(\Omega)$ be the POD bases in space and time, respectively. Then, a space-time Galerkin approximation of the generic semilinear equation system

$$\dot{v} - \Delta v + N(v) = f \quad \text{on } (0,T] \times \Omega, \tag{10a}$$

$$v\big|_{\partial\Omega} = 0 \quad \text{on } (0,T], \tag{10b}$$

$$v\big|_{t=0} = v_0 \quad \text{on } \Omega \tag{10c}$$

is given as follows.

The approximate solution $\hat{v}$ is assumed in the space $\hat{\mathcal{S}} \cdot \hat{\mathcal{Y}} := \operatorname{span}\{\hat{\psi}_j\hat{\nu}_i\}_{i=1,\dots,\hat{q}}^{j=1,\dots,\hat{s}}$. We introduce the formal vectors of the coefficient functions

$$\hat{\Upsilon} := \begin{bmatrix}\hat{\nu}_1\\ \vdots\\ \hat{\nu}_{\hat{q}}\end{bmatrix} \quad \text{and} \quad \hat{\Psi} := \begin{bmatrix}\hat{\psi}_1\\ \vdots\\ \hat{\psi}_{\hat{s}}\end{bmatrix}$$

and write $\hat{v}$ as

$$\begin{bmatrix}\hat{\psi}_1 & \dots & \hat{\psi}_{\hat{q}}\end{bmatrix} \otimes \begin{bmatrix}\hat{\nu}_1 & \dots & \hat{\nu}_{\hat{s}}\end{bmatrix}\hat{\mathbf{v}} = [\hat{\Psi}^{\mathsf{T}} \otimes \hat{\Upsilon}^{\mathsf{T}}]\hat{\mathbf{v}}, \tag{11}$$

where $\hat{\mathbf{v}} \in \mathbb{R}^{\hat{s}\hat{q}}$ is the vector of coefficients. We determine the coefficients by requiring them to satisfy the Galerkin projection of (10a) for every basis function $\hat{\nu}_i\hat{\psi}_j$, $i = 1, \dots, \hat{q}$, $j = 1, \dots, \hat{s}$,

$$\int_0^T \int_\Omega \hat{\nu}_i\hat{\psi}_j\dot{\hat{v}} + \hat{\psi}_j\nabla\hat{\nu}_i\nabla\hat{v} + \hat{\nu}_i\hat{\psi}_j N(\hat{v}) \,\mathrm{d}x \,\mathrm{d}t = \int_0^T \int_\Omega \hat{\nu}_i\hat{\psi}_j f \,\mathrm{d}x \,\mathrm{d}t.$$

The latter equations combined give a possibly nonlinear equation system for the vector $\hat{\mathbf{v}}$ of coefficients, which is assembled as follows: For the term with the time derivative we compute

$$\begin{aligned}\int_0^T \int_\Omega [\hat{\Psi} \otimes \hat{\Upsilon}] \frac{\partial \hat{v}}{\partial t} \,\mathrm{d}x \,\mathrm{d}t &= \int_0^T \int_\Omega [\hat{\Psi} \otimes \hat{\Upsilon}] \left[\frac{\partial \hat{\Psi}^\mathsf{T}}{\partial t} \otimes \hat{\Upsilon}^\mathsf{T}\right] \hat{\mathbf{v}} \,\mathrm{d}x \,\mathrm{d}t \\ &= \int_0^T \int_\Omega \left[\hat{\Psi}\frac{\partial \hat{\Psi}^\mathsf{T}}{\partial t} \otimes \hat{\Upsilon}\hat{\Upsilon}^\mathsf{T}\right] \,\mathrm{d}x \,\mathrm{d}t\hat{\mathbf{v}} \\ &= \left[\int_0^T \hat{\Psi}\frac{\partial \hat{\Psi}^\mathsf{T}}{\partial t} \,\mathrm{d}t \otimes \int_\Omega \hat{\Upsilon}\hat{\Upsilon}^\mathsf{T} \,\mathrm{d}x\right] \hat{\mathbf{v}} =: [dM_{\hat{\mathcal{S}}} \otimes M_{\hat{\mathcal{Y}}}]\hat{\mathbf{v}}.\end{aligned}$$

By the same principles, for the term with the spatial derivatives, we obtain

$$\begin{aligned}\int_0^T \int_\Omega [\hat{\Psi} \otimes \nabla\hat{\Upsilon}] \nabla\hat{v} \,\mathrm{d}x \,\mathrm{d}t &= \int_0^T \int_\Omega [\hat{\Psi} \otimes \nabla\hat{\Upsilon}][\hat{\Psi}^\mathsf{T} \otimes \nabla\hat{\Upsilon}^\mathsf{T}]\hat{\mathbf{v}} \,\mathrm{d}x \,\mathrm{d}t \\ &= \left[\int_0^T \hat{\Psi}\hat{\Psi}^T \,\mathrm{d}t \otimes \int_\Omega \nabla\hat{\Upsilon}\nabla\hat{\Upsilon}^\mathsf{T} \,\mathrm{d}x\right] \hat{\mathbf{v}} := [M_{\hat{\mathcal{S}}} \otimes K_{\hat{\mathcal{Y}}}]\hat{\mathbf{v}}.\end{aligned}$$

Note that in higher spatial dimensions, $\nabla\hat{v}$ as well as $\nabla\hat{\nu}_i$ is a vector and, thus, in the preceding derivation, the products and $\nabla\hat{\Upsilon}$ have to be interpreted properly.

Summing up, we can write the overall system as

$$[dM_{\hat{\mathcal{S}}} \otimes M_{\hat{\mathcal{Y}}} + M_{\hat{\mathcal{S}}} \otimes K_{\hat{\mathcal{Y}}}]\hat{\mathbf{v}} + H_{\hat{\mathcal{S}}\hat{\mathcal{Y}}}(\hat{\mathbf{v}}) = f_{\hat{\mathcal{S}}\hat{\mathcal{Y}}}, \tag{12}$$

where

$$M_{\hat{\mathcal{S}}} := [(\hat{\nu}_i, \hat{\nu}_j)]_{i,j=1,\dots,\hat{s}}, \tag{13a}$$

$$dM_{\hat{\mathcal{S}}} := [(\hat{\nu}_i, \dot{\hat{\nu}}_j)]_{i,j=1,\dots,\hat{s}}, \tag{13b}$$

$$M_{\hat{\mathcal{Y}}} := [(\hat{\psi}_l, \hat{\psi}_k)]_{l,k=1,\dots,\hat{q}}, \tag{13c}$$

$$K_{\hat{\mathcal{Y}}} := [(\nabla\hat{\psi}_l, \nabla\hat{\psi}_k)]_{l,k=1,\dots,\hat{q}}, \tag{13d}$$

$$H_{\hat{\mathcal{S}}\hat{\mathcal{Y}}}(\hat{\mathbf{v}}) := [((\hat{\nu}_i\hat{\psi}_l, N(\hat{v})))]_{i=1,\dots,\hat{s};\ l=1,\dots,\hat{q}}, \tag{13e}$$

and

$$f_{\hat{\mathcal{S}}\hat{\mathcal{Y}}} := [((\hat{\nu}_i\hat{\psi}_l, f))]_{i=1,\dots,\hat{s};\ l=1,\dots,\hat{q}}, \tag{13f}$$

are the Galerkin projections of the system operators and the source term assembled in the corresponding inner products.

Remark 3.1. In the space-time Galerkin POD context, the reduced bases are projections of standard finite element bases. Concretely, by virtue of Lemmas 2.5 and 2.7 one has that

$$\hat{\Psi} = U_{\hat{s}}^\mathsf{T}\mathbf{L}_{\mathcal{S}}^{-1}\Psi \quad \text{and} \quad \hat{\Upsilon} = V_{\hat{q}}^\mathsf{T}\mathbf{L}_{\mathcal{Y}}^{-1}\Upsilon,$$

where the columns of $U_{\hat{s}}$ and $V_{\hat{q}}$ are orthonormal and where $\mathbf{L}_{\mathcal{S}}$ and $\mathbf{L}_{\mathcal{Y}}$ are factors of the mass matrices associated with Ψ and Υ. Accordingly, the coefficients in (13) are given as

$$M_{\hat{\mathcal{S}}} := U_{\hat{s}}^{\mathsf{T}}\mathbf{L}_{\mathcal{S}}^{-1}\left[\int_0^T \Psi\Psi^{\mathsf{T}}\,\mathrm{d}s\right]\mathbf{L}_{\mathcal{S}}^{-\mathsf{T}}U_{\hat{s}} = U_{\hat{s}}^{\mathsf{T}}\mathbf{L}_{\mathcal{S}}^{-1}M_{\mathcal{S}}\mathbf{L}_{\mathcal{S}}^{-\mathsf{T}}U_{\hat{s}} = I_{\hat{s}}, \tag{14a}$$

$$dM_{\hat{\mathcal{S}}} := U_{\hat{s}}^{\mathsf{T}}\mathbf{L}_{\mathcal{S}}^{-1}\left[\int_0^T \Psi\dot{\Psi}^{\mathsf{T}}\,\mathrm{d}s\right]\mathbf{L}_{\mathcal{S}}^{-\mathsf{T}}U_{\hat{s}}, \tag{14b}$$

$$M_{\hat{\mathcal{Y}}} := V_{\hat{q}}^{\mathsf{T}}\mathbf{L}_{\mathcal{Y}}^{-1}\left[\int_\Omega \Upsilon\Upsilon^{\mathsf{T}}\,\mathrm{d}x\right]\mathbf{L}_{\mathcal{Y}}^{-\mathsf{T}}V_{\hat{q}} = V_{\hat{q}}^{\mathsf{T}}\mathbf{L}_{\mathcal{Y}}^{-1}M_{\mathcal{Y}}\mathbf{L}_{\mathcal{Y}}^{-\mathsf{T}}V_{\hat{q}} = I_{\hat{q}}, \tag{14c}$$

$$K_{\hat{\mathcal{Y}}} := V_{\hat{q}}^{\mathsf{T}}\mathbf{L}_{\mathcal{Y}}^{-1}\left[\int_\Omega \nabla\Upsilon\nabla\Upsilon^{\mathsf{T}}\,\mathrm{d}x\right]\mathbf{L}_{\mathcal{Y}}^{-\mathsf{T}}V_{\hat{q}}. \tag{14d}$$

Note that, despite their larger size, stiffness matrices of the standard finite element discretization, as they appear in (14b) and (14d), may be assembled efficiently by finite element packages. Thus it might be much faster to assemble and project the standard FEM matrices than to compute the stiffness matrices $dM_{\hat{\mathcal{S}}}$ and $K_{\hat{\mathcal{Y}}}$ as in the formulation given in (13b) and (13d).

4. Implementation issues.

In this section, we address how to compute the measurement matrices by means of standard tools, how to incorporate the initial and terminal values in the time discretization, how to preassemble quadratic nonlinearities, and how to use interpolation for general nonlinearities.

4.1. Computation of the measurements.

We explain how the measurements (cf. Remark 2.1) that are needed for the computation of the optimal low-rank bases (cf. Lemmas 2.5 and 2.7) can be obtained in practical cases.

In the standard *method-of-lines* approach, a $\mathcal{Y}$ will be used as the FE space for a Galerkin spatial discretization that approximates (10a) by an ODE. In a second step, a time integration scheme is employed to approximate the coefficients $v_1, \dots, v_q \colon (0,T] \to \mathbb{R}$ of the solution

$$\bar{v}\colon (0,T] \to \mathcal{Y}\colon t \mapsto \sum_{i=1}^{q} v_i(t)\nu_i$$

of the resulting ODE. With this and with a chosen time measurement space $\mathcal{S}$, a numerical computed measurement in $\mathcal{S}\cdot\mathcal{Y}$ of the actual solution v of (10a) is given as

$$\mathbf{X} = \begin{bmatrix} (v_1,\psi_1)_{\mathcal{S}} & \dots & (v_1,\psi_s)_{\mathcal{S}} \\ \vdots & \ddots & \vdots \\ (v_q,\psi_1)_{\mathcal{S}} & \dots & (v_q,\psi_s)_{\mathcal{S}} \end{bmatrix} \mathbf{M}_{\mathcal{S}}^{-1}. \tag{15}$$

Remark 4.1. For smooth trajectories and for measurements using delta distributions centered at some $t_j \in (0,T)$, $j = 1, \dots, s$, with $\int_0^T v_i\delta(t_j)\,\mathrm{d}t = v_i(t_j)$ the matrix (15) degenerates to the standard POD *snapshot matrix*. In this case, since the delta distributions are not elements of $L^2(0,T)$, there is no way to define an optimal time basis as in Lemma 2.7. However, one can define an optimal low-rank spatial basis by Lemma 2.5 which reduces to the standard POD optimality result with $\mathbf{M}_{\mathcal{S}} = I$; cf. Remarks 2.8 and 2.9.

4.2. Treatment of the initial value. The initial value (10c) requires special consideration. First of all, the solution of the PDE is well defined only when the initial condition is specified. This transfers to the space-time Galerkin discretized system (12) insofar that it is uniquely solvable only if an initial condition is provided. Second, in particular in view of optimal control, the initial value can be subject to changes which should be realizable in the discretized model.

To maintain the prominent role of the initial condition also in the time discretization, we proceed as follows:

1. We choose an $\mathcal{S}$ that is spanned by a nodal basis $\{\psi_1, \dots, \psi_s\}$, where ψ_1 is the basis function associated with the node at $t = 0$.
2. For a given function, we compute $\mathbf{X}_0$ as in (4) or (15) setting $\psi_1 = 0$ and $U_{\hat{s},0}$ as the matrix of the $\hat{s} - 1$ leading right singular vectors of $\mathbf{L}_{\mathcal{Y}}^{\mathsf{T}}\mathbf{X}_0\mathbf{L}_{\mathcal{S}}$.
3. We set
$$U_{\hat{s}} = \begin{bmatrix} \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} & U_{\hat{s},0} \end{bmatrix}$$
and compute the reduced basis as in Lemma 2.7 as
$$\begin{bmatrix} \hat{\psi}_1 \\ \hat{\psi}_2 \\ \vdots \\ \hat{\psi}_{\hat{s}} \end{bmatrix} = U_{\hat{s}}^{\mathsf{T}} \mathbf{L}_{\mathcal{S}}^{-1} \begin{bmatrix} \psi_1 \\ \psi_2 \\ \vdots \\ \psi_s \end{bmatrix}.$$

Under the assumption that $\mathbf{L}_{\mathcal{S}}$ is a lower-triangular factor such that $\mathbf{L}_{\mathcal{S}}\mathbf{L}_{\mathcal{S}}^{\mathsf{T}} = \mathbf{M}_{\mathcal{S}}$, by this construction we obtain that $\hat{\psi}_1 = \frac{1}{\|\psi_1\|}\psi_1$ will be associated with the initial value, whereas $\hat{\psi}_2(0) = \cdots = \hat{\psi}_{\hat{s}}(0) = 0$ will still optimally approximate the trajectory.

A terminal value is treated similarly, though it requires an factorization $\mathbf{L}_{\mathcal{S}}\mathbf{L}_{\mathcal{S}}^{\mathsf{T}} = \mathbf{M}_{\mathcal{S}}$ of the mass matrix with $\mathbf{L}_{\mathcal{S}}$ upper-triangular.

4.3. Assembling of quadratic nonlinearities. As an example, we consider the nonlinearity in the Burgers' equation

$$\frac{1}{2}\partial_x z(t,x)^2 \tag{16}$$

with the spatial coordinate $x \in (0,1)$ and the time variable $t \in (0,1]$.

In the time-space Galerkin projection (11), the *il*-component of the discretized nonlinearity (13e) in the case of (16) is given as

$$\begin{aligned} H_{il}(\hat{\mathbf{v}}) &= \frac{1}{2}\int_0^1 \int_0^1 \hat{\nu}_i \hat{\psi}_l \cdot \partial_x \hat{v}^2 \,\mathrm{d}x \,\mathrm{d}t \\ &= \frac{1}{2}\int_0^1 \int_0^1 \hat{\nu}_i \hat{\psi}_l \cdot \partial_x \left(\left(\left[\hat{\Psi}^{\mathsf{T}} \otimes \hat{\Upsilon}^{\mathsf{T}} \right] \hat{\mathbf{v}} \right)^2 \right) \,\mathrm{d}x \,\mathrm{d}t \\ &= \hat{\mathbf{v}}^{\mathsf{T}} \left[\int_0^1 \hat{\nu}_i \hat{\Psi} \hat{\Psi}^{\mathsf{T}} \,\mathrm{d}t \otimes \frac{1}{2}\int_0^1 \hat{\psi}_l \partial_x (\hat{\Upsilon}\hat{\Upsilon}^{\mathsf{T}})^2 \,\mathrm{d}x \right] \hat{\mathbf{v}}, \end{aligned}$$

where we have used the linearity of the *Kronecker product* and that

$$\hat{v}^2 = ([\hat{\Psi}^{\mathsf{T}} \otimes \hat{\Upsilon}^{\mathsf{T}}]\hat{\mathbf{v}})^2 = \hat{\mathbf{v}}^{\mathsf{T}}[\hat{\Psi} \otimes \hat{\Upsilon}][\hat{\Psi}^{\mathsf{T}} \otimes \hat{\Upsilon}^{\mathsf{T}}]\hat{\mathbf{v}} = \hat{\mathbf{v}}^{\mathsf{T}}[\hat{\Psi}\hat{\Psi}^{\mathsf{T}} \otimes \hat{\Upsilon}\hat{\Upsilon}^{\mathsf{T}}]\hat{\mathbf{v}}.$$

Thus, the evaluation of the discretized nonlinear term can be assisted by precomputing

$$\int_0^1 \hat{\nu}_i \hat{\Psi}\hat{\Psi}^\mathsf{T}\ \mathrm{d}t \quad \text{and} \quad \frac{1}{2}\int_0^1 \hat{\psi}_l(\hat{\Upsilon}\partial_x\hat{\Upsilon}^\mathsf{T} + \partial_x(\hat{\Upsilon})\hat{\Upsilon}^\mathsf{T})\ \mathrm{d}x$$

for all $\hat{\nu}_i$, $i = 1, \dots, \hat{s}$ and $\hat{\psi}_l$, $l = 1, \dots, \hat{q}$. These precomputed quantities are matrices of size $\hat{s} \times \hat{s}$ and $\hat{q} \times \hat{q}$ and can be interpreted as slices of a tensor of size $\hat{s} \times \hat{s} \times \hat{s}$ and $\hat{q} \times \hat{q} \times \hat{q}$ that realize the quadratic form in time and space, respectively.

Remark 4.2. If $V_{\hat{q}}$ is the matrix of the spatial POD modes that transform the FEM basis Υ into the reduced basis $\hat{\Upsilon}$ via $\hat{\Upsilon} = V_{\hat{q}}^\mathsf{T}\mathbf{L}_\mathcal{Y}^{-1}\Upsilon$, then the spatial part of the reduced nonlinearity fulfills

$$\frac{1}{2}\int_0^1 \hat{\psi}_l(\hat{\Upsilon}\partial_x\hat{\Upsilon}^\mathsf{T} + \partial_x(\hat{\Upsilon})\hat{\Upsilon}^\mathsf{T})\ \mathrm{d}x = \frac{1}{2}V_{\hat{q}}^\mathsf{T}\mathbf{L}_\mathcal{Y}^{-1}\int_0^1 \hat{\psi}_l(\Upsilon\partial_x\Upsilon^\mathsf{T} + \partial_x(\Upsilon)\Upsilon^\mathsf{T})\ \mathrm{d}x\mathbf{L}_\mathcal{Y}^{-\mathsf{T}}V_{\hat{q}},$$

where the inner matrix of the latter expression might be efficiently assembled in a FEM package. The same idea applies to the time-related part.

4.4. Interpolation of general nonlinearities. If the nonlinearity N is a general function, one can approximate the evaluation of the space-time integral in (13e) through quadrature. This, however, requires a representation of $\hat{\mathbf{v}}$ that can be evaluated at given quadrature points and, thus, may become computationally expensive. In what follows, we assume that $N\colon \mathbb{R} \to \mathbb{R}$ is a scalar function and deliberately apply to functions v and vectors $\bar{v}$ using the conventions

$$v \mapsto N(v) \leftrightarrow v(t,x) \mapsto N(v(t,x)) \quad \text{and} \quad \bar{v} \mapsto N(\bar{v}) \leftrightarrow \begin{bmatrix} v_1 \\ \vdots \\ v_q \end{bmatrix} \mapsto \begin{bmatrix} N(v_1) \\ \vdots \\ N(v_q) \end{bmatrix}.$$

If the full spaces $\mathcal{S}$ and $\mathcal{Y}$ are spanned by nodal bases, i.e., if the single basis functions are associated with single grid points and if they have the value 1 at these grid points and the value 0 at all other grid points, then for $v = \sum_{j=1}^{s}\sum_{i=1}^{q} v_{i,j}\nu_i\psi_j$, on the space-time grid, the value of $N(v)$ is readily approximated by its interpolant

$$N(v) \approx \sum_{j=1}^{s}\sum_{i=1}^{q} N(v_{i,j})\nu_i\psi_j. \tag{17}$$

Since, generically, the (reduced) bases of $\hat{\mathcal{S}} \cdot \hat{\mathcal{Y}}$ are not nodal, the low-dimensional approximation

$$\hat{v} = [\hat{\Psi}^\mathsf{T} \otimes \hat{\Upsilon}^\mathsf{T}]\hat{\mathbf{v}} \in \hat{\mathcal{S}} \cdot \hat{\mathcal{Y}}$$

needs to be expanded in the basis of $\mathcal{S} \cdot \mathcal{Y}$ first,

$$\hat{v} = [\hat{\Psi}^\mathsf{T} \otimes \hat{\Upsilon}^\mathsf{T}]\hat{\mathbf{v}} = [\Psi^\mathsf{T}\mathbf{L}_\mathcal{S}^{-\mathsf{T}}U_{\hat{s}} \otimes \hat{\Upsilon}^\mathsf{T}\mathbf{L}_\mathcal{Y}^{-\mathsf{T}}V_{\hat{q}}]\hat{\mathbf{v}} = [\Psi^\mathsf{T} \otimes \Upsilon^\mathsf{T}][\mathbf{L}_\mathcal{S}^{-\mathsf{T}}U_{\hat{s}} \otimes \mathbf{L}_\mathcal{Y}^{-\mathsf{T}}V_{\hat{q}}]\hat{\mathbf{v}} \in \mathcal{S} \cdot \mathcal{Y}$$

(see Remark 3.1), before the interpolant can be computed as

$$N(\hat{\mathbf{v}}) \approx [\Psi^\mathsf{T} \otimes \Upsilon^\mathsf{T}]N([\mathbf{L}_\mathcal{S}^{-\mathsf{T}}U_{\hat{s}} \otimes \mathbf{L}_\mathcal{Y}^{-\mathsf{T}}V_{\hat{q}}]\hat{\mathbf{v}}).$$

Then, the nonlinearity $H_{\hat{\mathcal{S}}\hat{\mathcal{Y}}}(\hat{\mathbf{v}})$ as in (13e) can be approximated through interpolation via

$$\begin{aligned} H_{\hat{\mathcal{S}}\hat{\mathcal{Y}}}(\hat{\mathbf{v}}) &= \int_0^T \int_\Omega [\hat\Psi \otimes \hat\Upsilon] N(\hat{\mathbf{v}}) \,\mathrm{d}x \,\mathrm{d}t \\ &\approx \int_0^T \int_\Omega [\hat\Psi \otimes \hat\Upsilon][\Psi^\mathsf{T} \otimes \Upsilon^\mathsf{T}] N([\mathbf{L}_\mathcal{S}^{-\mathsf{T}} U_{\hat s} \otimes \mathbf{L}_\mathcal{Y}^{-\mathsf{T}} V_{\hat q}]\hat{\mathbf{v}}) \,\mathrm{d}x \,\mathrm{d}t \\ &= [U_{\hat s}^\mathsf{T} \mathbf{L}_\mathcal{S}^{-1} \otimes V_{\hat q}^\mathsf{T} \mathbf{L}_\mathcal{Y}^{-1}] \int_0^T \int_\Omega [\Psi \otimes \Upsilon][\Psi^\mathsf{T} \otimes \Upsilon^\mathsf{T}] \,\mathrm{d}x \,\mathrm{d}t N([\mathbf{L}_\mathcal{S}^{-\mathsf{T}} U_{\hat s} \otimes \mathbf{L}_\mathcal{Y}^{-\mathsf{T}} V_{\hat q}]\hat{\mathbf{v}}) \\ &= [U_{\hat s}^\mathsf{T} \mathbf{L}_\mathcal{S}^{-1} \otimes V_{\hat q}^\mathsf{T} \mathbf{L}_\mathcal{Y}^{-1}][\mathbf{M}_\mathcal{S} \otimes \mathbf{M}_\mathcal{Y}] N([\mathbf{L}_\mathcal{S}^{-\mathsf{T}} U_{\hat s} \otimes \mathbf{L}_\mathcal{Y}^{-\mathsf{T}} V_{\hat q}]\hat{\mathbf{v}}) \\ &= [U_{\hat s}^\mathsf{T} \mathbf{L}_\mathcal{S}^\mathsf{T} \otimes V_{\hat q}^\mathsf{T} \mathbf{L}_\mathcal{Y}^\mathsf{T}] N([\mathbf{L}_\mathcal{S}^{-\mathsf{T}} U_{\hat s} \otimes \mathbf{L}_\mathcal{Y}^{-\mathsf{T}} V_{\hat q}]\hat{\mathbf{v}}). \end{aligned} \tag{18}$$

Remark 4.3. The interpolant $N([\mathbf{L}_\mathcal{S}^{-\mathsf{T}} U_{\hat s} \otimes \mathbf{L}_\mathcal{Y}^{-\mathsf{T}} V_{\hat q}]\hat{\mathbf{v}})$ is of dimension sq (the product of the dimensions of the full time and space approximation spaces) and, thus, causes a disproportional computational effort during the setup and solution of the low order model. However, the product structure in $[U_{\hat s}^\mathsf{T} \mathbf{L}_\mathcal{S}^\mathsf{T} \otimes V_{\hat q}^\mathsf{T} \mathbf{L}_\mathcal{Y}^\mathsf{T}] N([\mathbf{L}_\mathcal{S}^{-\mathsf{T}} U_{\hat s} \otimes \mathbf{L}_\mathcal{Y}^{-\mathsf{T}} V_{\hat q}]\hat{\mathbf{v}})$ seems well suited for an extension of the *discrete empirical interpolation method* [7] to a space-time setup that, bluntly put, selects and interpolates the most relevant components through

$$[U_{\hat s}^\mathsf{T} \mathbf{L}_\mathcal{S}^\mathsf{T} \otimes V_{\hat q}^\mathsf{T} \mathbf{L}_\mathcal{Y}^\mathsf{T}] W (P^\mathsf{T} W)^{-1} P N([\mathbf{L}_\mathcal{S}^{-\mathsf{T}} U_{\hat s} \otimes \mathbf{L}_\mathcal{Y}^{-\mathsf{T}} V_{\hat q}]\hat{\mathbf{v}}) \tag{19}$$

with a matrix $W \in \mathbb{R}^{sq,w}$ that spans the interpolation space of dimension $w \ll sq$ and a selector matrix $P \in \mathbb{R}^{w,sq}$ that consists of w chosen unit vectors of the $\mathbb{R}^{sq}$. Thus, if the product $[U_{\hat s}^\mathsf{T} \mathbf{L}_\mathcal{S}^\mathsf{T} \otimes V_{\hat q}^\mathsf{T} \mathbf{L}_\mathcal{Y}^\mathsf{T}] W \in \mathbb{R}^{\hat s \hat q, w}$ is precomputed, the approximative evaluation of N via (19) is independent of the full space dimensions.

Remark 4.4. For a straightforward implementation, the formulation of (18) as

$$\operatorname{vec}\Big(V_{\hat q}^\mathsf{T} \mathbf{L}_\mathcal{Y}^\mathsf{T} N\big(\mathbf{L}_\mathcal{Y}^{-\mathsf{T}} V_{\hat q} \mathbf{X} U_{\hat s}^\mathsf{T} \mathbf{L}_\mathcal{S}^{-1}\big) \mathbf{L}_\mathcal{S} U_{\hat s} \Big),$$

with the coefficients of $\hat{\mathbf{v}}$ as matrix $\mathbf{X}$, that avoids the memory consuming projection and lifting matrices like $\mathbf{L}_\mathcal{S}^{-\mathsf{T}} U_{\hat s} \otimes \mathbf{L}_\mathcal{Y}^{-\mathsf{T}} V_{\hat q} \in \mathbb{R}^{qs, \hat q \hat s}$ is preferable.

5. Application in PDE-constrained optimization.

We consider a generic optimal control problem.

Problem 5.1. *For a given target trajectory $x^* \in L^2(0,T;L^2(\Omega))$ and a penalization parameter $\alpha > 0$, we consider the optimization problem*

$$\mathcal{J}(x,u) := \frac{1}{2}\|x - x^*\|_{L^2}^2 + \frac{\alpha}{2}\|u\|_{L^2}^2 \to \min_{u \in L^2(0,T;L^2(\Omega))} \tag{20}$$

subject to the generic PDE

$$\dot x - \Delta x + N(x) = f + u \quad \text{on } (0,T] \times \Omega, \tag{21a}$$

$$x\big|_{\partial\Omega} = 0 \quad \text{on } (0,T], \tag{21b}$$

$$x\big|_{t=0} = x_0 \quad \text{on } \Omega. \tag{21c}$$

If the nonlinearity is Frechét differentiable, then necessary optimality conditions with respect to Problem 5.1 for (x,u) are given through $u = \frac{1}{\alpha}\lambda$, where λ solves the adjoint equation

$$-\dot\lambda - \Delta\lambda + D_xN(x)^{\mathsf{T}}\lambda + x = x^* \quad \text{on } (0,T]\times\Omega, \tag{22a}$$

$$\lambda\big|_{\partial\Omega} = 0 \quad \text{on } (0,T], \tag{22b}$$

$$\lambda\big|_{t=T} = 0 \quad \text{on } \Omega, \tag{22c}$$

which is coupled to the state equation (21) through x and u; see [20]. Here, D_x denotes the Frechét derivative.

Given low-dimensional spaces $\hat{\mathcal{S}} := \operatorname{span}\{\hat\psi_1, \dots, \hat\psi_{\hat s}\} \subset H^1(0,T)$, $\hat{\mathcal{R}} := \operatorname{span}\{\hat\phi_1, \dots, \hat\phi_{\hat r}\} \subset H^1(0,T)$ and $\hat{\mathcal{Y}} := \operatorname{span}\{\hat\nu_1, \dots, \hat\nu_{\hat q}\} \subset H_0^1(\Omega)$, $\hat\Lambda := \operatorname{span}\{\lambda_1, \dots, \lambda_{\hat p}\} \subset H_0^1(\Omega)$, a tensor space-time Galerkin discretization of the coupled system (21)–(22) reads

(23a)

$$[dM_{\hat{\mathcal{S}}} \otimes M_{\hat{\mathcal{Y}}} + M_{\hat{\mathcal{S}}} \otimes K_{\hat{\mathcal{Y}}}]\hat{\mathbf{v}} + H_{\hat{\mathcal{S}}\hat{\mathcal{Y}}}(\hat{\mathbf{v}}) - \frac{1}{\alpha}[M_{\hat{\mathcal{S}}\hat{\mathcal{R}}} \otimes M_{\hat{\mathcal{Y}}\hat\Lambda}]\hat{\boldsymbol{\lambda}} = f_{\hat{\mathcal{S}}\hat{\mathcal{Y}}},$$

(23b)

$$[-dM_{\hat{\mathcal{R}}} \otimes M_{\hat\Lambda} + M_{\hat{\mathcal{R}}} \otimes K_{\hat\Lambda}]\hat{\boldsymbol{\lambda}} + D_xN^{\mathsf{T}}_{\hat\Lambda\hat{\mathcal{R}}}(\hat{\mathbf{v}})\hat{\boldsymbol{\lambda}} + [M_{\hat{\mathcal{R}}\hat{\mathcal{S}}} \otimes M_{\hat\Lambda\hat{\mathcal{Y}}}]\hat{\mathbf{v}} = [M_{\hat{\mathcal{R}}\hat{\mathcal{S}}} \otimes M_{\hat\Lambda\hat{\mathcal{Y}}}]\hat{\mathbf{v}}^*,$$

with the coefficients $dM_{\hat{\mathcal{R}}}$, $M_{\hat{\mathcal{R}}}$, $M_{\hat\Lambda}$, $K_{\hat\Lambda}$ and the nonlinearity $D_xN^{\mathsf{T}}_{\hat\Lambda\hat{\mathcal{R}}}(\hat{\mathbf{v}})\hat{\boldsymbol{\lambda}}$ defined as in (12), with $M_{\hat{\mathcal{S}}\hat{\mathcal{R}}}$, $M_{\hat{\mathcal{R}}\hat{\mathcal{S}}}$, $M_{\hat{\mathcal{Y}}\hat\Lambda}$, $M_{\hat\Lambda\hat{\mathcal{Y}}}$ denoting the mixed mass matrices like

$$M_{\hat{\mathcal{S}}\hat{\mathcal{R}}} := [\big(\hat\psi_\ell, \hat\phi_k\big)]_{k=1,\dots,\hat r}^{\ell=1,\dots,\hat s} \in \mathbb{R}^{\hat s, \hat r},$$

with $\hat{\mathbf{v}}^*$ representing the target v^* projected onto $\hat{\mathcal{S}} \cdot \hat{\mathcal{Y}}$, with the spatial boundary conditions resolved in the ansatz spaces, and with accounting for the initial and terminal conditions via requiring

$$\hat v(0) = \sum_{j=1}^{\hat s}\sum_{i=1}^{\hat q} \mathbf{x}^1_{i,j}\hat\nu_i\hat\psi_j(0) = \Pi_{\hat{\mathcal{Y}}}x_0 \quad \text{and} \quad \hat\lambda(T) = \sum_{j=1}^{\hat r}\sum_{i=1}^{\hat p} \mathbf{x}^1_{i,j}\hat\mu_i\hat\phi_j(T) = 0;$$

cf. section 4.2.

Remark 5.2. Since the solution x to (21) depends on an initial value at time $t = 0$, since the λ to (22) depends on a terminal value at $t = T$, and since both systems are fully coupled, a numerical approach to them has to either decouple the systems, e.g., in an iterative process, or consider the time as a global variable. In the latter case a reduced model of the time evolution, as provided by the low-dimensional time Galerkin spaces $\hat{\mathcal{S}}$ and $\hat{\mathcal{R}}$, can lead to a significant complexity reduction.

Remark 5.3. In the case that the input enters the system through an input operator $B\colon \mathbb{R}^{n_u} \to L^2(\Omega)$, $n_u \in \mathbb{N}$, and that an output $y = Cx$ with $C\colon L^2(\Omega) \to \mathbb{R}^{n_y}$, $n_y \in \mathbb{N}$, is tracked rather than the full state, the space-time discretized optimality conditions (23) have to be modified as follows:

$$\begin{aligned} \frac{1}{\alpha}[M_{\hat{\mathcal{S}}\hat{\mathcal{R}}} \otimes M_{\hat{\mathcal{Y}}\hat\Lambda}]\hat{\boldsymbol{\lambda}} \quad &\leftarrow \quad \frac{1}{\alpha}[M_{\hat{\mathcal{S}}\hat{\mathcal{R}}} \otimes B_{\hat{\mathcal{Y}}}B^{\mathsf{T}}_{\hat\Lambda}]\hat{\boldsymbol{\lambda}}, \\ [M_{\hat{\mathcal{R}}\hat{\mathcal{S}}} \otimes M_{\hat\Lambda\hat{\mathcal{Y}}}](\hat{\mathbf{v}} - \hat{\mathbf{v}}^*) \quad &\leftarrow \quad [M_{\hat{\mathcal{R}}\hat{\mathcal{S}}} \otimes C^{\mathsf{T}}_{\hat\Lambda}C_{\hat{\mathcal{Y}}}](\hat{\mathbf{v}} - \hat{\mathbf{v}}^*), \end{aligned} \tag{24}$$

with the corresponding space discrete input and output operators.

6. Numerical experiments.

We consider the optimal control of a Burgers' equation as described in [12, 14] and the 2D *Chafee–Infante* equation and test the proposed space-time Galerkin POD approach. To estimate the performance quantitatively, we run similar tests with a well-established gradient-based method.

6.1. Problem and test setup.

For the first setup, in Problem 5.1, we replace the generic PDE (21) by the one-dimensional Burgers' equation, namely,

$$\dot{x} - \nu\partial_{\xi\xi}x + \frac{1}{2}\partial_\xi(x^2) = u \quad \text{on } (0,T]\times(0,L), \tag{25a}$$

$$x\big|_{\xi=0,\xi=L} = 0 \quad \text{on } (0,T], \tag{25b}$$

$$x\big|_{t=0} = x_0 \quad \text{on } (0,L), \tag{25c}$$

where L and T denote the length of the space and time interval and where $\nu > 0$ is the viscosity. We set $T = 1$ and $L = 1$ and, as the initial value, we take the step function

$$x_0\colon (0,1)\to\mathbb{R}\colon \xi\mapsto \begin{cases} 1 & \text{if } \xi \le 0.5, \\ 0 & \text{if } \xi > 0.5. \end{cases} \tag{25d}$$

For the *Chafee–Infante* setup, we consider

$$\dot{x} - \nu(\partial_{\xi_1\xi_1}x + \partial_{\xi_2\xi_2}x) + \beta(x^3 - x) = 0 \quad \text{on } (0,T]\times(0,L)^2, \tag{26a}$$

$$\partial_n x\big|_{\xi_2=0} = 0 \quad \text{on } (0,T], \tag{26b}$$

$$\partial_n x\big|_{\xi_1=L} = g\cdot u \quad \text{on } (0,T], \tag{26c}$$

$$x\big|_{\xi_1=0} = x\big|_{\xi_2=L} = 0 \quad \text{on } (0,T], \tag{26d}$$

where ∂_n denotes the normal derivative, where, again, L and T denote the length of the space and time interval, where $\nu > 0$, again, is a viscosity parameter whereas β is parameter that controls the nonlinearity, and where $g = g(\xi_2)$ is a shape function that models the spatial extension of the control. We set $T = 3$ and $L = 1$ and, as the initial value, we take the zero function.

Instead of tracking the full state, we observe an output $y = Cx$ defined as

$$y(t) = \frac{1}{0.2\cdot 0.2}\int_{0.4}^{0.6}\int_{0.4}^{0.6} x(t,\xi)\,\mathrm{d}\xi_2\,\mathrm{d}\xi_1 \tag{27}$$

which is basically the mean value of x at time t in a domain of observation.

6.1.1. Definition of the optimal control problems.

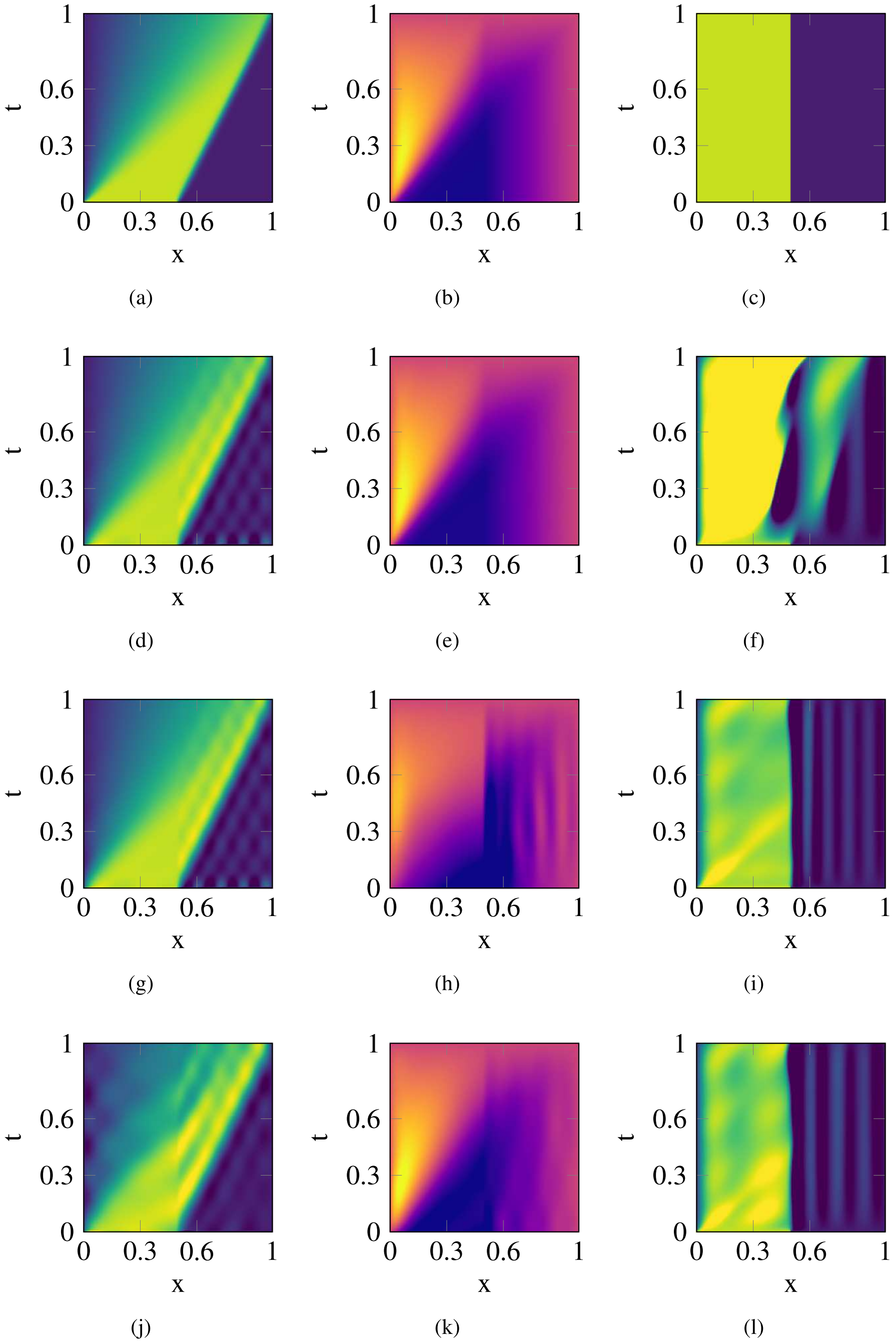
For the first example problem, we define x^* via $x^*(t) = x_0$ as the target. Thus, the concrete optimal control problem which is designed to keep the system in its initial state (cf. Figure 1(c)) reads as follows.

PROBLEM 6.1. *Given parameters ν and α, find $u \in L^2(0,1;L^2(0,1))$ such that*

$$\frac{1}{2}\int_0^1\int_0^1 (x(t,\xi) - x_0)^2\,\mathrm{d}\xi\,\mathrm{d}t + \frac{\alpha}{2}\int_0^1\int_0^1 u^2(t,\xi)\,\mathrm{d}\xi\,\mathrm{d}t \to \min_{u\in L^2(0,1;L^2(0,1))} \tag{28}$$

subject to Burgers' equation (25).

As the second test case, we consider a space-time varying target state. Therefore, we define the function $\chi_\heartsuit\colon (0,1)\times(0,1)\to\{0,1\}$ as the indicator function of a heart-shaped set in the space-time domain as depicted in Figure 2(c).



(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

(j) (k) (l)

FIG. 1. *Illustration of the effect of the choice of the snapshots on the performance of the low-dimensional approximations of the state* (a)*, the adjoint state* (b)*, and the target state* (c) *or the response of the suboptimal controls* (f), (i), (l)*. The second row* (d)–(f) *corresponds to the case that snapshots of the state and the adjoint are used to approximate the state and the adjoint, respectively. For the results depicted in the third row* (g)–(i)*, the optimized basis for the state was used also for the adjoint. The results depicted in the last row* (j)–(l) *were obtained by combining state and adjoint snapshots for the computation of the reduced bases. For a comparable illustration we have used color maps with linear intensity with light color values corresponding to low state values on the intervals* $[-0.1, 1.1]$ *for the states and* $[-0.5, 0.5]$ *for the adjoint states. Values that exceeded these margins were cropped.*

Problem 6.2. *Given parameters ν and α, find $u \in L^2(0,1;L^2(0,1))$ such that*

$$\frac{1}{2}\int_0^1\int_0^1 (x(t,\xi) - \chi_{\heartsuit}(t,\xi))^2 \,\mathrm{d}\xi\,\mathrm{d}t + \frac{\alpha}{2}\int_0^1\int_0^1 u^2(t,\xi)\,\mathrm{d}\xi\,\mathrm{d}t \to \min_{u\in L^2(0,1;L^2(0,1))} \tag{29}$$

subject to Burgers' equation (25).

As the third test case, we consider the optimal control of the *Chafee–Infante* equation (26) toward the output trajectory

$$y^*(t) = \begin{cases} 0 & \text{if } t < 0.5, \\ C[\mathbf{1}] & \text{if } t > 2.5, \\ \frac{1}{2}(1-\cos(\frac{t-0.5}{2}\pi))C[\mathbf{1}] & \text{in between} \end{cases}$$

that describes a smooth transition from the output of the zero function toward the output of the function **1** that is 1 in the whole domain.

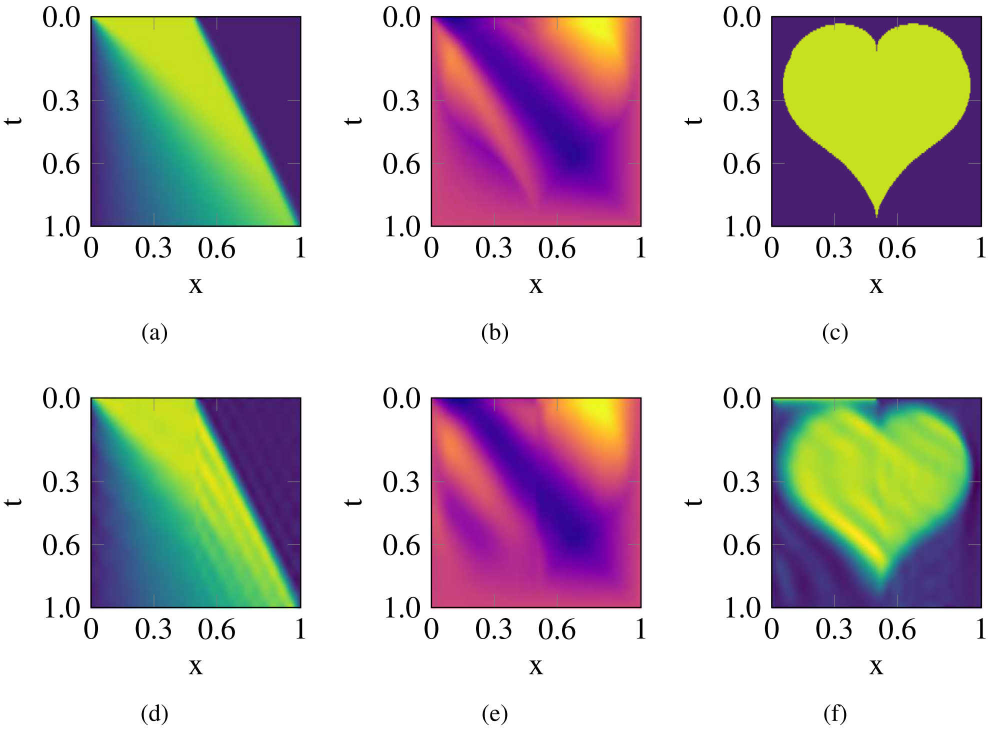


Fig. 2. *Illustration of the optimization problem with a target state varying in space and time as described in section* 6.1.1. *The snapshots are taken from a forward simulation without control* (a) *and the corresponding adjoint solution* (b) *with respect to the target state* (c). *Plots* (d) *and* (e) *show the results of the low-dimensional space-time Galerkin approximation to the primal and the adjoint state. Plot* (f) *illustrates the response of a suboptimal control computed on the basis of the space-time reduced system. For a comparable illustration we have used color maps with linear intensity on the intervals* $[-0.1, 1.1]$ *for the states and* $[-0.3, 0.3]$ *for the adjoint states. Values that exceeded these margins were cropped.*

Problem 6.3. *Given parameters ν and α, find $u \in L^2(0,1;L^2(0,1))$ such that*

$$\frac{1}{2}\int_0^3 (Cx(t) - y^*(t))^2 \,\mathrm{d}t + \frac{\alpha}{2}\int_0^3 u^2(t)\,\mathrm{d}t \to \min_{u\in L^2(0,3;\mathbb{R})} \tag{30}$$

subject to the Chafee–Infante equation (26).

6.1.2. Reduced order optimization approaches. We compute suboptimal solutions to the optimal control problems, Problems 6.1 and 6.2, i.e., we compute optimal controls on the basis of reduced order models that approximate the actual optimal control problems. For that, we consider

- space-time-pod—the space-time Galerkin POD projection of the optimality system (cf. section 5),
- sqp-pod—a *sequential-quadratic programming* (SQP) approach for a POD reduced model.

Details on the implementation are given in sections 6.2 and 6.3 below.

6.1.3. Performance measures. We measure the performance of the suboptimal controls for the actual optimal control problems, Problems 6.1 and 6.2, through

- the tracking error $\frac{1}{2}\|\hat{x} - x_0\|^2_{L^2}$ or $\frac{1}{2}\|\hat{x} - \chi_\heartsuit\|^2_{L^2}$ between the target state and the state $\hat{x}$ achieved by using the suboptimal control $\hat{u}$ in the simulation of the full model,
- the time `walltime` it takes to solve the reduced systems for the suboptimal control $\hat{u}$. We report the lowest measured time out of five runs.

Furthermore, we report

- the norm of the computed control $\|u\|^2_{L^2(0,1;L^2(0,1))}$

and, for the tests in the iterative gradient based approach sqp-pod,

- the numbers of function and gradient evaluations `nfc/ngc`.

6.1.4. Test definitions. We investigate the performance of the space-time Galerkin approach over the given range of parameters by means of the following test suites:

- *Dimension of the reduced model.* We gradually increase the degrees of freedom of the reduced model, equally distributed to the space and time dimension.
- *Varying space and time resolution.* Starting from an equal distribution that has proven to perform well, we gradually increase/decrease the dimension of the reduced model in the time dimension while decreasing/increasing its dimension in the space dimension. In other words, we gradually shift the weighting between spatial and time resolution in the reduced model.
- *Variable viscosity.* For a fixed model dimension, we explore the performance versus varying viscosity parameters.
- *Variable regularization parameter.* For a fixed model dimension, we explore the performance versus the regularization parameter α in the cost functional.

It will turn out that the sqp-pod approach leads to low tracking errors and cost function values but at higher computational costs. As an attempt to reduce the costs, we consider another test suite for the sqp-pod approach:

- *Variable gradient norm.* We gradually change the threshold for the norm of the gradient which is the termination criterion for the minimization.

References to all results are given in Table 1. More details on the particular test setups and an interpretation of the results are given in sections 6.2 and 6.3 for space-time-pod and sqp-pod, respectively. A comparison and an assessment of both methods

Table 1
List of numerical experiments and the corresponding tables of results.

Test setup	Problem 6.1—step function		Problem 6.2—heart shape	
	space-time-pod	sqp-pod	space-time-pod	sqp-pod
Dimension of the reduced model	Table 5	Table 5	Table 13	Table 13
Varying space and time resolution	Table 6	Table 6	Table 14	Table 14
Variable viscosity	Tables 7, 8	Tables 7, 8	Tables 15, 16	Tables 15, 16
Varying regularization parameter	Tables 9, 10	Tables 9, 10	Tables 17, 18	Tables 17, 18
Variable gradient norm	—	Tables 11, 12	—	Tables 19, 20

are given in section 6.6. A 2D map of the performance of the methods with respect to the degrees of freedom in the reduced model is given in Figure 3.

6.1.5. Implementation. The spatial discretization is carried out with the help of the FEM library *FEniCS* [15]. For the time integration, we use *SciPy*'s built-in ODE-integrator `scipy.integrate.odeint`. The norms are approximated in the used FEM space. The implementation and the code for all tests as well as the documentation of the hardware are available from the author's public git repository [11]; see also the note on code availability in section 7.

6.2. Space-time generalized POD for optimal control. The general procedure is as follows:

1. Do at least one forward solve of the state equation (25) and at least one backward solve of the corresponding adjoint equation (cf. (22)) to setup generalized measurement matrices of the state and the costate as explained in section 4.1.
2. Compute optimized space and time bases for the state and the costate as defined in Lemmas 2.5 and 2.7. To account for the initial and the terminal value, one may resort to the procedure explained in section 4.2.
3. Set up the projected closed-loop optimality system (23) and solve for the optimal costate $\hat{\boldsymbol{\lambda}}$ of the reduced system.
4. Lift $\hat{\boldsymbol{u}} = \frac{1}{\alpha}\hat{\boldsymbol{\lambda}}$ up to the full space-time grid and apply it as suboptimal control in the full-fidelity discretization.

To solve the nonlinear system (23) for $\hat{\boldsymbol{\lambda}}$, we use *SciPy*'s `scipy.optimize.fsolve`, which is a wrapper for the method `HYBRD` of the *Fortran* package *MINPACK*. The underlying method is the *Powell hybrid method* that combines a Newton and a steepest descent algorithm. We provided the associated Jacobian as a function which, among others, can be derived from the representation of the nonlinearities as laid out in section 4.3, and we used the prolonged and projected initial state to initialize the forward part of the solution and zero values for the backward part.

The procedure is defined by several parameters. In the presented examples, we fix $\mathcal{Y} = \Lambda$ and $\mathcal{S} = \mathcal{R}$, corresponding to the full-fidelity space and time discretizations, and investigate the influence of the other parameters on the numerical solution of the optimal control problem. See Table 2 for an overview of the parameters and their default values.

Choice of the measurements. The computation of the measurements and the choice of the reduced bases are important parameters of the approach. Generally, the basis of $\hat{\mathcal{S}} \cdot \hat{\mathcal{Y}}$ should be well suited to approximate the state, whereas the basis $\hat{\mathcal{R}} \cdot \hat{\Lambda}$ should well represent the adjoint state. In the optimization case, where the suboptimal input is defined through $\frac{1}{\alpha}\hat{\boldsymbol{\lambda}}$ and its lifting to the full order space, two other conditions

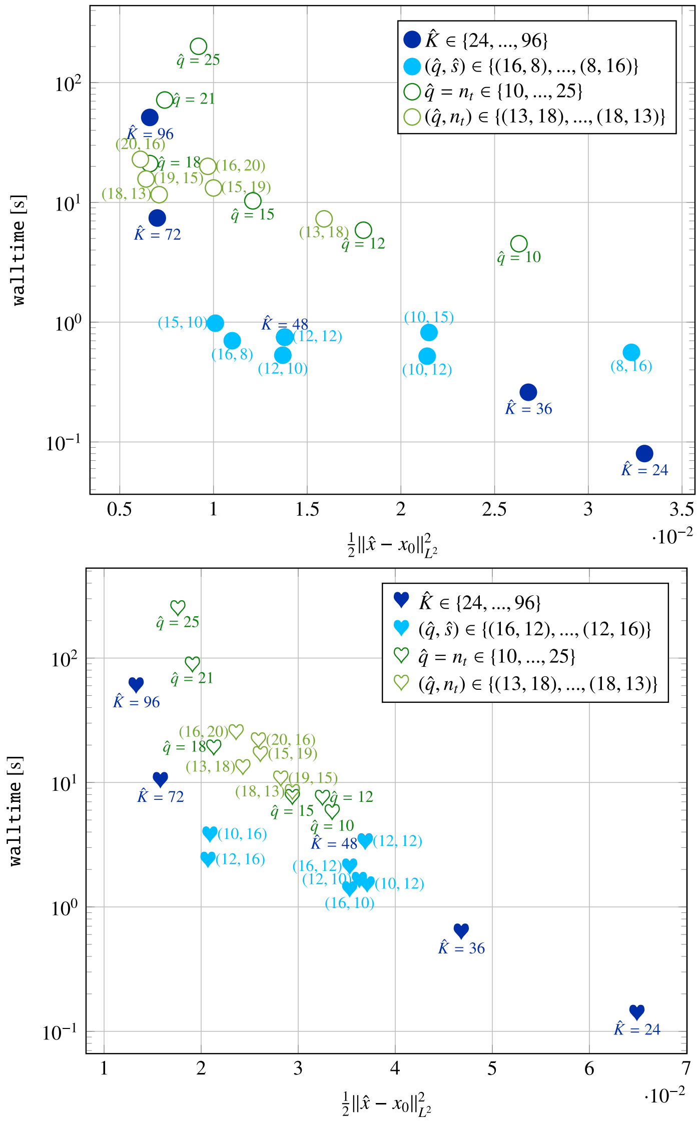


FIG. 3. *Varying temporal and spatial dimensions in the reduced order models for* space-time-pod *(●,♥) versus* sqp-pod *(○,♡) for Problem* 6.1 *(top) and Problem* 6.2 *(bottom).*

emerge. First, the reduced basis of the adjoint state should also well approximate the optimal control. Second, the bases of the state and the adjoint must not be orthogonal or "almost" orthogonal such that the joint mass matrix $[M_{\hat{\mathcal{S}}\hat{\mathcal{R}}} \otimes M_{\hat{\mathcal{Y}}\hat{\Lambda}}]$ degenerates and the contribution of the input in (23a) vanishes.

As illustrated in the plots in Figure 1, the straightforward approach of constructing the bases for the state by means of state measurements and the basis for the adjoint by means of measurements of the adjoint well approximates the state and the adjoint but not the coupled problem. It turned out that taking the state measurements to also construct the reduced space for the adjoint gave a better approximation to the optimality system while, naturally, only poorly approximating the adjoint. The best

TABLE 2

Description and values of the parameters of the numerical tests with space-time-pod*; cf. section* 6.2.

Parameter	Description	Default values	Range
$\mathcal{Y}$, Λ	Space of piecewise linear finite elements on an equidistant grid of dimension q, p	$q = p = 220$	–
$\mathcal{S}$, $\mathcal{R}$	Space of linear hat functions on an equidistant grid of dimension s, r	$s = r = 120$	–
$\hat{\mathcal{Y}}$, $\hat{\Lambda}$	POD reductions of $\mathcal{Y}$ and Λ of dimension $\hat{q}$, $\hat{p}$; cf. Lemma 2.5	$\hat{q} = \hat{p} = 12$	$6 - 24$
$\hat{\mathcal{S}}$, $\hat{\mathcal{R}}$	POD reductions of $\mathcal{S}$ and $\mathcal{R}$ of dimension $\hat{s}$, $\hat{r}$; cf. Lemma 2.7	$\hat{r} = \hat{s} = 12$	$6 - 24$
α	Regularization parameter in the cost functional (28)	$1 \cdot 10^{-3}$	$2.5 \cdot 10^{-4} - 1.6 \cdot 10^{-2}$
ν	Viscosity parameter in the PDE (25)	$5 \cdot 10^{-3}$	$5 \cdot 10^{-4} - 1.6 \cdot 10^{-2}$

results were obtained in combining state and adjoint state measurements to construct the bases.

Thus, for the computation of the optimal bases for the following tests, we combined the measurements obtained from one forward solve with no control and one backward solve with the state from the forward solve and the target state.

In what follows, we report on the performance of space-time-pod in the test setups as defined in section 6.1.4.

Dimension of the reduced model. We set $\hat{q} = \hat{p} = \hat{r} = \hat{s} = \hat{K}/4$ with $\hat{K} \in \{24, 36, 48, 72, 96\}$. Thus, for every setup, the nonlinear system (23) of dimension $\hat{K}$ has to be solved for the optimal costate $\hat{\boldsymbol{\lambda}}$. The results of these tests are reported in Tables 5 and 13, respectively.

As expected, the larger the reduced model, the lower the achieved values of the cost functional. Also, with growing order of the reduced model, the time needed to solve the corresponding nonlinear system increases drastically. Also the memory requirement grows with $\hat{K}^3$, which is mainly due to the nonlinear part; see section 4.3.

Varying space and time reduction. From the previous tests, we found that in the considered setup, an overall number of $\hat{K} = 48$ modes is a good compromise between accuracy and computation time. In this section, we examine how the distribution of modes between space and time affects the quality of the suboptimal control. Therefore, and for varying increments/decrements j and i, we set $\hat{q} = \hat{p} := 12 \mp j$ and $\hat{s} = \hat{r} := 12 \pm i$. Accordingly, the overall number of degrees of freedom stays more or less the same throughout the tests but we add weight on the approximation of either the time or the space component.

The results are listed in Tables 6 and 14. By putting more emphasis on the space component (for Problem 6.1) or on the time component (for Problem 6.2) it is possible to get a significant increase in the performance. Interestingly, the timings `walltime` vary significantly even for the same overall dimensions of the reduced model. This variance is due to different convergence behavior of the optimization algorithm used to solve the nonlinear system.

Variable viscosity. In these tests, we examine how the low-rank space-time Galerkin approach performs over a range of viscosity parameters ν.

The results for Problem 6.1 are listed in Table 8 for $(\hat{q}, \hat{s}) = (\hat{p}, \hat{r}) = (16, 8)$, which was the most beneficial distribution as found in the previous tests, and in Table 7 for $(\hat{q}, \hat{s}) = (\hat{p}, \hat{r}) = (12, 12)$. The optimal distribution $(16, 8)$ has its performance peak at $\nu = 8 \cdot 10^{-3}$ and outperforms the model with the equally distributed modes over almost the whole range.

The results for Problem 6.2 are listed in Tables 16 and 15 for $(\hat{q}, \hat{s}) = (10, 15)$ and for $(\hat{q}, \hat{s}) = (12, 12)$. In terms of the tracking error, the nonequal distribution outperforms the equal distribution over the whole range, while in computation time there is no significant difference. The failure of space-time-pod for $\nu = 1.6e - 2$ in Table 15 is due to a particularly bad performing reduced basis for the adjoint state.

All tests reveal another phenomenon, namely, that the control magnitude $\|\hat{u}\|^2_{L^2}$ increases with the parameter ν. This reflects that, in contrast to forward simulations, in control problems, a larger viscosity makes the system harder to solve especially for nonsmooth target functions. At the other side of the spectrum, for low values of ν, the problem is *convection dominated* and hard to approximate by POD bases.

Regularization parameter. In this section, we examine the influence of the regularization parameter α on the performance—for $(\hat{q}, \hat{s}) = (\hat{p}, \hat{r}) \in \{(16, 8), (12, 12)\}$ (for Problem 6.1) and $(\hat{q}, \hat{s}) = (\hat{p}, \hat{r}) \in \{(10, 15), (12, 12)\}$ (for Problem 6.2). The results are reported in Tables 10 and 9 and Tables 18 and 17, respectively.

With smaller values of α, in all cases, the control magnitude $\|\hat{u}\|^2_{L^2}$ increases. However, the tracking error $\frac{1}{2}\|\hat{x} - x_0\|^2_{L^2}$ reaches a minimum and then increases again. A reason for this increase might be that the increased control action leads to more extreme states that are not well captured in the reduced model and, thus, not well propagated to the full simulation.

6.3. Gradient-based optimal control with POD.

To give also a quantitative estimate of the performance of the space-time POD-reduced Galerkin approach for the solution of PDE-constrained optimal control problems, we tackle the same optimization Problem 6.1 with the established approach of SQP [4, 10] with the *BFGS* approximation combined with standard POD for spatial model order reduction.

Briefly, the SQP method is an iterative scheme for the minimization of the *reduced cost functional* $\tilde{\mathcal{J}}(u) := \mathcal{J}(x(u), u)$ where the iterant u_{k+1} is given as the argument minimum of the Taylor approximation of $\tilde{\mathcal{J}}(u)$ around u_k truncated after the quadratic term,

$$(31) \quad \tilde{\mathcal{J}}_Q(u; u_k) := \tilde{\mathcal{J}}(u_k) + [\nabla_u \tilde{\mathcal{J}}(u_k)](u - u_k) + \frac{1}{2}(u - u_k)^{\mathsf{T}}[\nabla_{uu}\tilde{\mathcal{J}}(u_k)](u - u_k).$$

We approximate the gradient $\nabla_u \tilde{\mathcal{J}}(u_k)$ and the inverse of the Hessian $\nabla_{uu}\tilde{\mathcal{J}}(u_k)$, as needed for the minimization of $\tilde{\mathcal{J}}_Q(u; u_k)$ in (31), by solving the adjoint equation and employing the BFGS approximation formula [10, Chap. 3.2.1]. Accordingly, in the generic case, the cost of every iteration is basically that of a solve of (21) to obtain the state x_k corresponding to u_k and a solve of (22) to obtain λ_k from $x_k(u_k)$ that defines the current gradient $\nabla_u \tilde{\mathcal{J}}(u_k)$ plus, possibly, another few forward solves to determine the step size of the gradient step (line search).

We realize this iteration for the Burgers' problem, Problem 6.1, with the same parameters as before; cf. Table 2. In particular, the numerical solution of the corresponding PDEs bases on the same FEM discretization with $q = 220$ degrees of freedom as in the numerical experiments in section 6.2 and *SciPy*'s built-in ODE-integrator `scipy.integrate.odeint`.

The SQP approach iterates on fully discrete approximations to the input u_k and the BFGS iteration approximates the full Hessian matrix. Thus, for both efficiency and feasibility, the discrete representation needs to be compressed. Therefore, we use the same optimized reduced state spaces $\hat{\mathcal{Y}}$ and $\hat{\Lambda}$ for the forward and adjoint

Table 3

Description and values of the parameters of the numerical examples of section 6.3. *The value in parentheses refers to Problem* 6.2.

Parameter	Description	Base value	Range
$\mathcal{Y}$, Λ	Space of piecewise linear finite elements on an equidistant grid of dimension q, p	$q = p = 220$	–
$\mathcal{S}$, $\mathcal{R}$	Space of linear hat functions on an equidistant grid of dimension s, r – to compute the snapshots for the POD	$s = r = 120$	–
$\hat{\mathcal{Y}}$, $\hat{\Lambda}$	POD reductions of $\mathcal{Y}$ and Λ of dimension $\hat{q}$, $\hat{p}$; cf. Lemma 2.5	$\hat{q} = \hat{p} = 18$	10 – 25
n_t	dimension of the time grid on which u_k is linearly interpolated	18	10 – 25
$\texttt{tol}_\nabla$	Termination tolerance for the norm of the gradient in the SQP iterations	$2.5 \cdot 10^{-4}$	$1.77 \cdot 10^{-4}$ – $1 \cdot 10^{-3}$
α	Regularization parameter in the cost functional (28)	$3.125 \cdot 10^{-5}$ $(6.25 \cdot 10^{-5})$	$7.81 \cdot 10^{-6}$ – $2.5 \cdot 10^{-4}$
ν	Viscosity parameter in the PDE (25)	$5 \cdot 10^{-3}$	$5 \cdot 10^{-4}$ – $1.6 \cdot 10^{-2}$

problem and, accordingly, the spatial dimension of the control as for the space-time Galerkin approach; cf. Table 2. The time dimension of u_k is reduced by considering the linear interpolant on an equidistant time grid of n_t nodes. Thus, the dimension of the discrete u_k that defines the number of unknowns in the optimization is given as $\hat{q} \cdot n_t$.

As further parameters that influence the performance of the SQP-BFGS iteration, we consider $\texttt{tol}_\nabla$—the target tolerance value of the gradient minimization. All approximation defining parameters, as well as the problem parameters ν and α, are listed in Table 3.

We use *SciPy*'s routine `scipy.optimize.fmin_bfgs` to solve the reduced discrete optimization problem and then lift the obtained suboptimal control $\hat{\mathbf{u}}$ to the full space and apply it in the unreduced problem.

Remark 6.4. In the optimal control of systems, the evaluation of the cost functional $\tilde{J}(u_k)$ and its gradient $\nabla_u \tilde{J}(u_k)$ are both based on the same solution $x_k(u_k)$ of the forward problem. In the built-in *SciPy* implementation of the BFGS iteration this redundancy is not considered. To account for that, in the reported `walltime`, we have subtracted the time of the redundant forward solves that we estimate as the number of gradient computations times the average time for one forward solve.

Dimension of the reduced model. In this section, we test how the dimension of the reduced model affects the performance. The results are listed in Tables 5 and 13. Generally, for higher model dimension, the tracking error decreases at the expense of higher computation times. For the easier problem, Problem 6.1, the tracking error, however, goes up again, which might be due to the poorer performance of the optimization algorithm.

Variable viscosity. Here, we examine how a reduced model of fixed dimension performs with respect to the viscosity parameter. The results are listed in Tables 7 and 8 and Tables 15 and 16. For the step-function target (Problem 6.1), the expected behavior can be observed: a performance peak in the middle of the parameter range and a control magnitude that increases with the viscosity (Tables 7, 8). For the harder problem with the heart-shaped target, the performance is good over the whole parameter range without showing the particular patterns except that the computation time increases toward the margins (15, 16).

Regularization parameter. In Tables 9 and 10 and Tables 17 and 18, we tabulate the measured performance for the sqp-pod approach versus varying choices of the regularization parameter α as used in the definition of the cost functionals to penalize the input action. Throughout the investigated range, the performance is equally good. The expected pattern that $\|\hat{u}\|_{L^2}^2$ decreases with increasing α cannot be observed, which is probably due to the termination of the minimization due to *loss of precision*, meaning that the algorithm could not find further descent directions.

Target norm of gradient. We investigate the influence of the termination criterion for the SQP iteration defined through the target value $\mathtt{tol}_\nabla$ of the gradient $\|\nabla_u \tilde{\mathcal{J}}(u_k)\|$. The results are summarized in Tables 11 and 12 and Tables 19 and 20. If the threshold value is increased, the computation time decreases at the expense of a higher tracking error. On the other hand, a threshold below a certain value does not have any further effect. This is due to a stagnation in the minimization process and the termination of the iteration because of *precision loss*.

6.4. Output tracking for the Chafee–Infante equation.

In this section, we compute suboptimal controls for the tracking problem, Problem 6.3, by means of the space-time-pod and sqp-pod approaches as explained in sections 6.2 and 6.3.

The underlying equation (26) has an unstable equilibrium associated with $x = 0$ and two attractors associated with $x = -1$ and $x = 1$. The parameter $\beta \geq 0$ that controls the reaction rate such that with a larger β the attractors are approached faster is subject to the investigation. The viscosity parameter is set to $\nu = 0.1$, which appeared to be a suitable value for the diffusion of the boundary control into the domain.

For the spatial discretization, namely, for $\mathcal{Y}$ and Λ, we use an equally sized triangulation and piecewise linear functions such that $q = p = 900$. For the time discretization we use equispaced piecewise linear functions and $s = r = 90$. For the reduced model, we computed snapshots for the forward problem with the simulation data with the test input

$$u_{\mathsf{test}}(t) = \sin\left(\frac{2}{3}t\pi\right)$$

and snapshots for the backward problem on the base of the corresponding adjoint state. The reduced model was then defined with the manually optimized values $\hat{q} = \hat{p} = 10$ and $\hat{s} = \hat{r} = 15$ for the space-time-pod reduced model and $\hat{q} = n_t = 20$ for the sqp-pod approximation. Since the input is a scalar function, it is only discretized with respect to time. Accordingly, the computation of the sqp-pod solution requires the solution of an optimization problem with $n_t = 20$ parameters. The space-time-pod approach that solves for the reduced forward and adjoint states, nonetheless, requires the solution of a nonlinear system with $\hat{q} \cdot \hat{s} + \hat{p} \cdot \hat{r} = 300$ unknowns. In both approximations the nonlinearity is interpolated as described in section 4.4.

We tested the space-time-pod and the sqp-pod approaches for Problem 6.3 with varying $\beta \in \{1, 3, 5\}$; see Table 4 and Figure 4. In all setups, the problem is challenging since the control acts only on the boundary and since for large β the reaction is fast (see Figures 4(c), (f) and note the steep ascend of the solutions toward the attractor

Table 4

Performance of the space-time-pod *and* sqp-pod *approaches for computing suboptimal controls for the Chafee–Infante equation for various parameter β.*

	β	1.0	3.0	5.0
space-time-pod	$\frac{1}{2}\|C\hat{x} - y^*\|^2$	0.0232	0.0098	0.1731
	walltime	297	351	483
sqp-pod	$\frac{1}{2}\|C\hat{x} - y^*\|^2$	0.0304	0.0378	0.1234
	walltime	116	99.1	125

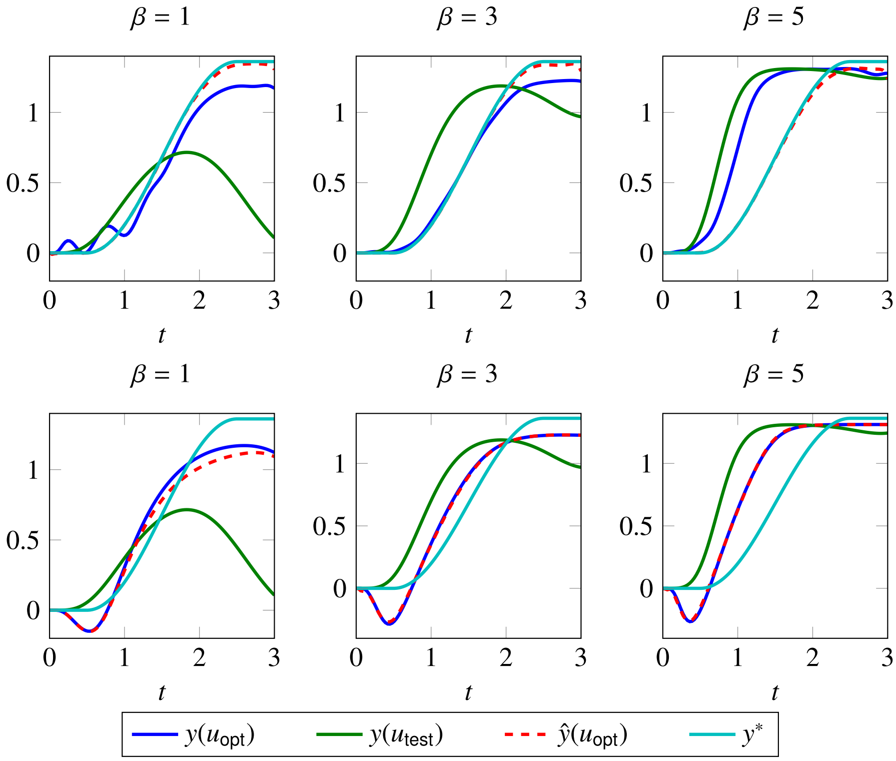


Fig. 4. *Performance of the suboptimal controls u_{opt} for the tracking of the Chafee–Infante equation computed with* space-time-pod *(top row) and* sqp-pod *(bottom row) for varying β. The plots show the response $y(u_{\mathrm{opt}})$ of the full order model, the response $y(u_{\mathrm{opt}})$ of the reduced order model, the target trajectory y^*, and, as an illustration of the dynamics, the response of the test input u_{test} in the full order system.*

and that the action of u_{test} for $t > 1.5$ has but a small effect), whereas for $\beta = 1$ still the high viscosity makes the control more expensive the more that the influence of the zero Dirichlet conditions moves the attractor away from the target of the optimization (see Figures 4(a), (d).

For the $\beta = 5$ case, both approaches led to suboptimal controls that hardly changed the transient behavior toward the attractor. For the other cases, the sqp-pod approach terminated three to four times faster than space-time-pod but with less performant suboptimal controls. This lack of performance was due to the optimizer in the sqp-pod that stagnated after a few iterations. Since the reduced model aligns well with the full order model (see Figures 4(d)–(f)), one may assume that the problem itself with a strong attractor and a relatively weak control is not well suited for an SQP approach.

On the other hand, the space-time-pod suboptimal controls deliver an almost perfect tracking in the reduced order model, though not in the full order model. This suggests that the computed control is indeed optimal in the reduced setting and that the reduced order model does not capture all dynamics. Finally, we note that the relatively long `walltime` for the computation can be reduced significantly if one would provide the Jacobian of system (23).

6.5. POD, space-time POD, and empirical interpolation. As illustrated in section 4.3, for the considered Burgers' equation, the (quadratic) nonlinearity can be reduced in line with the linear terms. However, in more general setups, the question of how to treat a nonlinearity in the reduced equations is immanent.

A generic approach would be an interpolation of $N(\hat{v})$ in the basis of $\hat{\mathcal{S}} \cdot \hat{\mathcal{Y}}$, i.e.,

$$N(\hat{v}) \approx \sum_{i=1}^{\hat{q}} \sum_{j=1}^{\hat{s}} n_{ij} \hat{\nu}_i \hat{\psi}_j,$$

for which the empirical interpolation method [1] might be extended to space-time setups.

Also for the SQP-POD approach of section 6.3, the nonlinearity, which basically is defined through the spatial part of the tensor described in section 4.3, is preassembled and reduced to the reduced dimension. Thus, the nonlinearity can be efficiently evaluated not resorting to the full dimension.

6.6. Summary and interpretation of the numerical results. In the preceding sections, we have used the proposed space-time Galerkin POD approach (space-time-pod) to compute suboptimal controls for a nonlinear PDE.

As a benchmark, we have solved the same problems with a well-established gradient-based method (sqp-pod). The benchmark implementation is highly optimized in terms of runtime and accuracy. In particular, the space dimension of the forward and backward problem is reduced through POD, the nonlinearities are preassembled for efficient evaluation in the reduced dimension, and the numerical time integration as well as the optimization are done by *SciPy*'s built-in routines that call on high-performance *Fortran* packages.

In the scenario of the time-constant target (Problem 6.1), in terms of the tracking error, the sqp-pod outperformed space-time-pod by a factor of 2. If the optimization in the sqp-pod algorithm is stopped on the tracking error level of space-time-pod, the space-time Galerkin approach appears to be faster by a factor of 4; cf. Tables 12 and 11 for $\mathtt{tol}_{\nabla} = 7.07 \cdot 10^{-4}$ versus Table 6 for, e.g., $(\hat{q}, \hat{s}) = (16, 8)$. Thus, for this scenario, the sqp-pod approach leads to good controls, while space-time-pod might be of use for the computation of less optimal controls in significantly shorter times.

In the scenario of the heart-shaped target (Problem 6.2), the space-time-pod approach reaches the tracking error level of sqp-pod while still being faster by a factor of 5; cf., e.g., the performance tabulation with respect to viscosity—Table 16.

7. Conclusion and outlook.

We have presented a novel approach to low-rank space-time Galerkin approximations that is based on a generalization of classical snapshot-based POD which then can be extended to POD reduction of time discretizations. We have proven optimality of the reduced bases in the relevant function spaces and discussed the numerical implementation.

The space-time Galerkin POD reduction applies well to optimal control problems, as we have illustrated for the optimal control of a Burgers' equation. In terms of both computation time for and efficiency of a suboptimal control, the new approach competes well with established gradient-based approaches. In terms of time needed to compute suboptimal controls, the newly proposed approach clearly outperforms the benchmark implementation. In a more challenging setup with a target function varying both with space and time, the proposed space-time Galerkin catches up with the benchmark also in terms of the tracking error.

In the current implementation of the numerical tests, the resulting nonlinear systems were solved by a general purpose routine, namely, *MINPACK*'s `HYBRD` [16] as it is included in *SciPy*. It might be worth investigating whether the performance of the space-time Galerkin approach for optimal control can be improved by better choices and tuning of the optimization routines.

Another possible further improvement and an issue for future work concerning the proposed space-time POD in application to optimal control problems lie in the freedom of the choice of the measurement functions [3]. Moreover, the underlying tensor structure is readily extended to include further directions of the state space like parameter dependencies [2] or inputs. Another issue that needs to be addressed is the treatment of general nonlinearities that cannot be treated by preassembling like in the presented quadratic case. Then, an inclusion of *empirical interpolation* [1] might be needed to achieve efficiency of the reduction. Moreover, it seems worth investigating whether the principles of space-time POD can be used to construct optimized bases for the interpolation.

Code availability.

The source code of the implementations used to compute the presented results can be obtained from

`doi:10.5281/zenodo.583296`

and is authored by Jan Heiland. Please contact Jan Heiland for licensing information

Appendix A. Numerical results for the step-function target.

TABLE 5
Problem 6.1 *with* space-time-pod *(top) and* sqp-pod *(bottom): Performance of the suboptimal control versus varying resolutions of space and time.*

$\hat{K}$	24	36	48	72	96
$\frac{1}{2}\|\hat{x}-x_0\|^2_{L^2}$	0.0322	0.0268	0.0138	0.0070	0.0066
$\|\hat{u}\|^2_{L^2}$	4.1516	5.4906	7.9611	10.944	9.2154
walltime $[s]$	0.08	0.26	0.75	7.41	51.2

$(\hat{q}, n_t)$	$(10,10)$	$(12,12)$	$(15,15)$	$(18,18)$	$(21,21)$	$(25,25)$
$\frac{1}{2}\|\hat{x}-x_0\|^2_{L^2}$	0.0263	0.0180	0.0121	0.0066	0.0074	0.0092
$\|\hat{u}\|^2_{L^2}$	9.6605	11.746	21.044	18.948	13.776	9.0060
walltime	4.52	5.86	10.3	21.1	71.6	201
nfc/ngc	113/101	127/115	140/140	175/174	112/112	93/ 93

Table 6

Problem 6.1 *with* space-time-pod *(top) and* sqp-pod *(bottom): Performance of the suboptimal control versus varying distributions of space and time resolutions; cf. sections* 6.2 *and* 6.3.

$(\hat{q}, \hat{s})/(\hat{p}, \hat{r})$	(16, 8)	(15,10)	(12,10)	(12,12)	(10,12)	(10,15)	(8,16)
$\frac{1}{2}\|\hat{x} - x_0\|^2_{L^2}$	0.0110	0.0101	0.0137	0.0138	0.0214	0.0215	0.0323
$\|\hat{u}\|^2_{L^2}$	9.9313	10.313	7.8039	7.9611	7.1523	6.9013	4.8211
`walltime`	0.70	0.98	0.53	0.75	0.52	0.82	0.56

$(\hat{q}, n_t)$	$(13, 18)$	$(15, 19)$	$(16, 20)$	$(19, 15)$	$(20, 16)$	$(18, 13)$
$\frac{1}{2}\|\hat{x} - x_0\|^2_{L^2}$	0.0159	0.0100	0.0097	0.0064	0.0061	0.0071
$\|\hat{u}\|^2_{L^2}$	11.472	14.295	14.068	19.243	18.885	20.781
`walltime`	7.29	13.2	20	15.7	22.9	11.6
`nfc/ngc`	109/109	153/145	154/143	161/151	132/132	135/135

Table 7

Problem 6.1 *with* space-time-pod *(top) and* sqp-pod *(bottom): Performance of the suboptimal control versus varying diffusion parameters ν for $(\hat{q}, \hat{s}) = (\hat{p}, \hat{r}) = (12, 12)$ for* space-time-pod *and for $(\hat{q}, n_t) = (18, 18)$ for* sqp-pod*; cf. sections* 6.2 *and* 6.3*, respectively.*

ν	$5 \cdot 10^{-4}$	$1 \cdot 10^{-3}$	$2 \cdot 10^{-3}$	$4 \cdot 10^{-3}$	$8 \cdot 10^{-3}$	$1.6 \cdot 10^{-2}$	$3.2 \cdot 10^{-2}$
$\frac{1}{2}\|\hat{x} - x_0\|^2_{L^2}$	0.0206	0.0208	0.0192	0.0150	0.0126	0.0106	0.0134
$\|\hat{u}\|^2_{L^2}$	6.3481	6.5214	6.7930	7.5351	9.3249	10.434	13.365
`walltime`	0.85	0.80	0.75	0.75	0.75	0.80	0.85

ν	$5 \cdot 10^{-4}$	$1 \cdot 10^{-3}$	$2 \cdot 10^{-3}$	$4 \cdot 10^{-3}$	$8 \cdot 10^{-3}$	$1.6 \cdot 10^{-2}$	$3.2 \cdot 10^{-2}$
$\frac{1}{2}\|\hat{x} - x_0\|^2_{L^2}$	0.0169	0.0167	0.0147	0.0087	0.0079	0.0105	0.0138
$\|\hat{u}\|^2_{L^2}$	9.7995	9.7424	9.2163	16.656	16.363	14.363	19.127
`walltime`	21.3	16.6	13.3	17	21.8	20.5	17.1
`nfc/ngc`	203/191	141/141	111/111	134/134	113/113	97/ 97	72/ 72

Table 8

Problem 6.1 *with* space-time-pod *(top) and* sqp-pod *(bottom): Performance of the suboptimal control versus varying diffusion parameters ν for $(\hat{q}, \hat{s}) = (\hat{p}, \hat{r}) = (16, 8)$ for* space-time-pod *and for $(\hat{q}, n_t) = (18, 13)$ for* sqp-pod*; cf. sections* 6.2 *and* 6.3*, respectively.*

ν	$5 \cdot 10^{-4}$	$1 \cdot 10^{-3}$	$2 \cdot 10^{-3}$	$4 \cdot 10^{-3}$	$8 \cdot 10^{-3}$	$1.6 \cdot 10^{-2}$	$3.2 \cdot 10^{-2}$
$\frac{1}{2}\|\hat{x} - x_0\|^2_{L^2}$	0.0224	0.0209	0.0166	0.0121	0.0090	0.0106	0.0140
$\|\hat{u}\|^2_{L^2}$	5.8554	6.4424	7.1055	9.1532	10.204	11.390	13.537
`walltime`	0.78	0.81	0.74	0.74	0.74	0.70	0.63

ν	$5 \cdot 10^{-4}$	$1 \cdot 10^{-3}$	$2 \cdot 10^{-3}$	$4 \cdot 10^{-3}$	$8 \cdot 10^{-3}$	$1.6 \cdot 10^{-2}$	$3.2 \cdot 10^{-2}$
$\frac{1}{2}\|\hat{x} - x_0\|^2_{L^2}$	0.0174	0.0168	0.0172	0.0100	0.0077	0.0098	0.0112
$\|\hat{u}\|^2_{L^2}$	10.142	9.4359	12.516	18.845	17.036	17.809	27.821
`walltime`	12.9	11.7	10.3	10.4	15.5	17.3	23
`nfc/ngc`	166/166	168/160	135/135	127/127	107/107	101/101	109/109

TABLE 9

Problem 6.1 *with* space-time-pod *(top) and* sqp-pod *(bottom): Performance of the suboptimal control versus varying regularization parameters for* $(\hat{q},\hat{s}) = (\hat{p},\hat{r}) = (12,12)$ *for* space-time-pod *and for* $(\hat{q},n_t) = (18,18)$ *for* sqp-pod*; cf. sections* 6.2 *and* 6.3*, respectively.*

α	$2.5\cdot 10^{-4}$	$5\cdot 10^{-4}$	$1\cdot 10^{-3}$	$2\cdot 10^{-3}$	$4\cdot 10^{-3}$	$8\cdot 10^{-3}$	$1.6\cdot 10^{-2}$
$\frac{1}{2}\|\hat{x}-x_0\|_{L^2}^2$	0.0165	0.0151	0.0138	0.0133	0.0142	0.0168	0.0217
$\|\hat{u}\|_{L^2}^2$	10.710	9.5104	7.9611	6.2444	4.6319	3.2791	2.2012
walltime	0.65	0.71	0.75	0.80	0.95	1.19	1.38

α	$3.91\cdot 10^{-6}$	$7.81\cdot 10^{-6}$	$1.56\cdot 10^{-5}$	$3.13\cdot 10^{-5}$	$6.25\cdot 10^{-5}$	$1.25\cdot 10^{-4}$
$\frac{1}{2}\|\hat{x}-x_0\|_{L^2}^2$	0.0067	0.0076	0.0069	0.0066	0.0087	0.0088
$\|\hat{u}\|_{L^2}^2$	18.032	14.408	18.539	18.948	12.454	12.742
walltime	17.6	14.5	21.4	21	14.6	14
nfc/ngc	136/136	113/113	200/189	175/174	136/130	131/122

TABLE 10

Problem 6.1 *with* space-time-pod *(top) and* sqp-pod *(bottom): Performance of the suboptimal control versus varying regularization parameters* α *for for* $(\hat{q},\hat{s}) = (\hat{p},\hat{r}) = (16,8)$ *for* space-time-pod *and for* $(\hat{q},n_t) = (18,13)$ *for* sqp-pod*; cf. sections* 6.2 *and* 6.3*, respectively.*

α	$2.5\cdot 10^{-4}$	$5\cdot 10^{-4}$	$1\cdot 10^{-3}$	$2\cdot 10^{-3}$	$4\cdot 10^{-3}$	$8\cdot 10^{-3}$	$1.6\cdot 10^{-2}$
$\frac{1}{2}\|\hat{x}-x_0\|_{L^2}^2$	0.0120	0.0111	0.0110	0.0121	0.0144	0.0180	0.0239
$\|\hat{u}\|_{L^2}^2$	16.286	13.209	9.9313	7.0006	4.7352	3.1187	1.9797
walltime	0.70	0.71	0.70	0.80	0.91	1.06	1.27

α	$3.91\cdot 10^{-6}$	$7.81\cdot 10^{-6}$	$1.56\cdot 10^{-5}$	$3.13\cdot 10^{-5}$	$6.25\cdot 10^{-5}$	$1.25\cdot 10^{-4}$
$\frac{1}{2}\|\hat{x}-x_0\|_{L^2}^2$	0.0070	0.0070	0.0070	0.0071	0.0073	0.0073
$\|\hat{u}\|_{L^2}^2$	20.186	20.302	20.451	20.781	17.063	17.089
walltime	11.4	11.7	11.6	11.6	12.2	12.6
nfc/ngc	133/133	136/136	134/134	135/135	156/149	159/151

TABLE 11

Problem 6.1 *with* sqp-pod*: Performance of the suboptimal control versus varying target values of the gradient minimization* $\texttt{tol}_\nabla$ *for* $(\hat{q},n_t) = (\hat{p},n_t) = (18,18)$*; cf. section* 6.3.

$\texttt{tol}_\nabla$	$1.77\cdot 10^{-4}$	$2.5\cdot 10^{-4}$	$3.54\cdot 10^{-4}$	$5\cdot 10^{-4}$	$7.07\cdot 10^{-4}$	$1\cdot 10^{-3}$	$1.41\cdot 10^{-3}$
$\frac{1}{2}\|\hat{x}-x_0\|_{L^2}^2$	0.0069	0.0066	0.0078	0.0142	0.0170	0.0224	0.0247
$\|\hat{u}\|_{L^2}^2$	20.045	18.948	14.268	5.9516	4.0778	2.7446	3.4519
walltime	21.1	21.1	16.3	7.18	5.83	4.16	3.39
nfc/ngc	177/176	175/174	143/142	58/ 58	47/ 47	34/ 34	28/ 28

TABLE 12

Problem 6.1 *with* sqp-pod*: Performance of the suboptimal control versus varying target values of the gradient minimization* $\texttt{tol}_\nabla$ *for* $(\hat{q},n_t) = (\hat{p},n_t) = (18,13)$.

$\texttt{tol}_\nabla$	$1.77\cdot 10^{-4}$	$2.5\cdot 10^{-4}$	$3.54\cdot 10^{-4}$	$5\cdot 10^{-4}$	$7.07\cdot 10^{-4}$	$1\cdot 10^{-3}$	$1.41\cdot 10^{-3}$
$\frac{1}{2}\|\hat{x}-x_0\|_{L^2}^2$	0.0071	0.0071	0.0072	0.0078	0.0105	0.0176	0.0186
$\|\hat{u}\|_{L^2}^2$	20.922	20.781	17.206	14.402	8.6073	4.3427	4.3608
walltime	13.7	11.6	10.2	8.61	5.99	3.28	2.88
nfc/ngc	164/152	135/135	117/117	102/102	73/ 73	40/ 40	35/ 35

Appendix B. Numerical results for the heart-shaped target.

Table 13

Problem 6.2 *with* space-time-pod *(top) and* sqp-pod *(bottom): Performance of the suboptimal control versus varying resolutions of space and time; cf. sections* 6.2 *and* 6.3.

$\hat{K}$	24	36	48	72	96
$\frac{1}{2}\Vert\hat{x}-\chi_\heartsuit\Vert_{L^2}^2$	0.0649	0.0468	0.0369	0.0158	0.0133
$\Vert\hat{u}\Vert_{L^2}^2$	4.1219	8.0076	9.4988	12.317	12.681
walltime $[s]$	0.14	0.63	3.34	10.4	60.0

$(\hat{q}, n_t)$	$(10, 10)$	$(12, 12)$	$(15, 15)$	$(18, 18)$	$(21, 21)$	$(25, 25)$
$\frac{1}{2}\Vert\hat{x}-\chi_\heartsuit\Vert_{L^2}^2$	0.0335	0.0325	0.0294	0.0213	0.0191	0.0176
$\Vert\hat{u}\Vert_{L^2}^2$	8.9765	9.2680	10.034	17.073	20.067	21.057
walltime	5.74	7.49	7.60	19.0	88.3	250
nfc/ngc	94/ 84	104/ 93	82/ 70	124/114	165/157	134/125

Table 14

Problem 6.2 *with* space-time-pod *(top) and* sqp-pod *(bottom): Performance of the suboptimal control versus varying distributions of space and time resolutions; cf. sections* 6.2 *and* 6.3.

$(\hat{q}, \hat{s})/(\hat{p}, \hat{r})$	(16,12)	(16,10)	(12,10)	(12,12)	(10,12)	(10,16)	(12,16)
$\frac{1}{2}\Vert\hat{x}-\chi_\heartsuit\Vert_{L^2}^2$	0.0353	0.0353	0.0363	0.0369	0.0371	0.0209	0.0207
$\Vert\hat{u}\Vert_{L^2}^2$	9.6202	8.9154	8.8530	9.4988	9.2766	10.347	10.609
walltime	2.10	1.38	1.64	3.33	1.51	3.80	2.38

$(\hat{q}, n_t)$	$(13, 18)$	$(15, 19)$	$(16, 20)$	$(19, 15)$	$(20, 16)$	$(18, 13)$
$\frac{1}{2}\Vert\hat{x}-\chi_\heartsuit\Vert_{L^2}^2$	0.0243	0.0261	0.0236	0.0282	0.0259	0.0294
$\Vert\hat{u}\Vert_{L^2}^2$	18.968	13.898	17.129	9.2041	12.164	9.0284
walltime	13.2	16.8	25.2	10.7	21.7	8.33
nfc/ngc	108/108	89/ 89	145/139	86/ 74	121/113	84/ 72

Table 15

Problem 6.2 *with* space-time-pod *(top) and* sqp-pod *(bottom): Performance of the suboptimal control versus varying diffusion parameters ν for $(\hat{q}, \hat{s}) = (\hat{p}, \hat{r}) = (12, 12)$ for* space-time-pod *and for $(\hat{q}, n_t) = (18, 18)$ for* sqp-pod*; cf. sections* 6.2 *and* 6.3.

ν	$5 \cdot 10^{-4}$	$1 \cdot 10^{-3}$	$2 \cdot 10^{-3}$	$4 \cdot 10^{-3}$	$8 \cdot 10^{-3}$	$1.6 \cdot 10^{-2}$	$3.2 \cdot 10^{-2}$
$\frac{1}{2}\Vert\hat{x}-\chi_\heartsuit\Vert_{L^2}^2$	0.0421	0.0429	0.0424	0.0390	0.0250	0.1565	0.0417
$\Vert\hat{u}\Vert_{L^2}^2$	8.5599	8.3219	8.3904	9.0221	12.890	34.558	28.385
walltime	1.68	1.65	1.71	1.98	2.36	2.10	1.78

ν	$5 \cdot 10^{-4}$	$1 \cdot 10^{-3}$	$2 \cdot 10^{-3}$	$4 \cdot 10^{-3}$	$8 \cdot 10^{-3}$	$1.6 \cdot 10^{-2}$	$3.2 \cdot 10^{-2}$
$\frac{1}{2}\Vert\hat{x}-\chi_\heartsuit\Vert_{L^2}^2$	0.0273	0.0311	0.0263	0.0223	0.0261	0.0208	0.0210
$\Vert\hat{u}\Vert_{L^2}^2$	20.078	11.560	14.617	20.686	9.8258	17.029	19.335
walltime	42.4	21.2	21.3	22.4	19.7	31.5	35.2
nfc/ngc	256/246	134/122	137/126	144/132	108/ 97	104/ 94	99/ 88

Table 16

Problem 6.2 *with* **space-time-pod** *(top) and* **sqp-pod** *(bottom): for* $(\hat{q},\hat{s}) = (\hat{p},\hat{r}) = (12,16)$ *for* **space-time-pod** *and for* $(\hat{q},n_t) = (13,18)$ *for* **sqp-pod**; *cf. sections* 6.2 *and* 6.3.

ν	$5\cdot10^{-4}$	$1\cdot10^{-3}$	$2\cdot10^{-3}$	$4\cdot10^{-3}$	$8\cdot10^{-3}$	$1.6\cdot10^{-2}$	$3.2\cdot10^{-2}$
$\frac{1}{2}\|\hat{x}-\chi_\heartsuit\|^2_{L^2}$	0.0331	0.0345	0.0314	0.0236	0.0179	0.0286	0.0233
$\|\hat{u}\|^2_{L^2}$	10.016	9.3107	9.2424	9.9573	11.463	16.820	20.378
walltime	3.56	3.75	3.76	6.80	2.49	5.73	3.29

ν	$5\cdot10^{-4}$	$1\cdot10^{-3}$	$2\cdot10^{-3}$	$4\cdot10^{-3}$	$8\cdot10^{-3}$	$1.6\cdot10^{-2}$	$3.2\cdot10^{-2}$
$\frac{1}{2}\|\hat{x}-\chi_\heartsuit\|^2_{L^2}$	0.0346	0.0294	0.0309	0.0269	0.0223	0.0234	0.0243
$\|\hat{u}\|^2_{L^2}$	11.172	16.089	11.540	14.321	17.264	14.183	12.718
walltime	12.9	16.0	10.9	12.9	11.6	18.4	19.3
nfc/ngc	125/114	152/140	111/101	131/119	118/106	95/ 86	86/ 78

Table 17

Problem 6.2 *with* **space-time-pod**: *Performance of the suboptimal control versus varying regularization parameters* α *for* $(\hat{q},\hat{s}) = (\hat{p},\hat{r}) = (12,12)$ *for* **space-time-pod** *and for* $(\hat{q},n_t) = (18,18)$ *for* **sqp-pod**; *cf. sections* 6.2 *and* 6.3.

α	$2.5\cdot10^{-4}$	$5\cdot10^{-4}$	$1\cdot10^{-3}$	$2\cdot10^{-3}$	$4\cdot10^{-3}$	$8\cdot10^{-3}$	$1.6\cdot10^{-2}$
$\frac{1}{2}\|\hat{x}-\chi_\heartsuit\|^2_{L^2}$	0.0536	0.0454	0.0369	0.0311	0.0303	0.0346	0.0431
$\|\hat{u}\|^2_{L^2}$	15.563	12.586	9.4988	6.7663	4.6295	3.0430	1.8744
walltime	1.87	2.26	3.33	1.76	1.17	1.22	1.37

α	$7.81\cdot10^{-6}$	$1.56\cdot10^{-5}$	$3.13\cdot10^{-5}$	$6.25\cdot10^{-5}$	$1.25\cdot10^{-4}$	$2.5\cdot10^{-4}$
$\frac{1}{2}\|\hat{x}-\chi_\heartsuit\|^2_{L^2}$	0.0196	0.0228	0.0229	0.0213	0.0228	0.0281
$\|\hat{u}\|^2_{L^2}$	19.440	13.358	13.688	17.073	13.010	9.0108
walltime	22	16.2	18.8	19.1	19.5	14.6
nfc/ngc	141/130	109/ 97	132/122	124/114	137/127	110/105

Table 18

Problem 6.2 *with* **space-time-pod** *(top) and* **sqp-pod** *(bottom): Performance of the suboptimal control versus varying regularization parameters* α *for* $(\hat{q},\hat{s}) = (\hat{p},\hat{r}) = (12,16)$ *for* **space-time-pod** *and for* $(\hat{q},n_t) = (13,18)$ *for* **sqp-pod**; *cf. sections* 6.2 *and* 6.3.

α	$2.5\cdot10^{-4}$	$5\cdot10^{-4}$	$1\cdot10^{-3}$	$2\cdot10^{-3}$	$4\cdot10^{-3}$	$8\cdot10^{-3}$	$1.6\cdot10^{-2}$
$\frac{1}{2}\|\hat{x}-\chi_\heartsuit\|^2_{L^2}$	0.0217	0.0210	0.0207	0.0225	0.0266	0.0330	0.0421
$\|\hat{u}\|^2_{L^2}$	25.726	15.882	10.609	7.1721	4.8042	3.1383	1.9333
walltime	2.0	4.8	2.4	2.8	2.4	2.5	2.8

α	$7.81\cdot10^{-6}$	$1.56\cdot10^{-5}$	$3.13\cdot10^{-5}$	$6.25\cdot10^{-5}$	$1.25\cdot10^{-4}$	$2.5\cdot10^{-4}$
$\frac{1}{2}\|\hat{x}-\chi_\heartsuit\|^2_{L^2}$	0.0242	0.0242	0.0265	0.0243	0.0266	0.0302
$\|\hat{u}\|^2_{L^2}$	18.339	18.530	13.327	18.968	13.847	9.4158
walltime	13.3	15.1	10.3	13.2	13.5	8.8
nfc/ngc	130/122	147/138	108/ 98	131/120	138/128	100/ 90

Table 19

Problem 6.2 *with* **sqp-pod**: *Performance of the suboptimal control versus varying target values of the gradient minimization* $\mathtt{tol}_\nabla$ *for* $(\hat{q},n_t) = (\hat{p},n_t) = (18,18)$; *cf. section* 6.3.

$\mathtt{tol}_\nabla$	$1.77\cdot10^{-4}$	$2.5\cdot10^{-4}$	$3.54\cdot10^{-4}$	$5\cdot10^{-4}$	$7.07\cdot10^{-4}$	$1\cdot10^{-3}$
$\frac{1}{2}\|\hat{x}-\chi_\heartsuit\|^2_{L^2}$	0.0213	0.0213	0.0213	0.0226	0.0280	0.0300
$\|\hat{u}\|^2_{L^2}$	17.073	17.073	17.073	14.149	9.1307	6.9991
walltime	19.1	19.1	19.1	15.4	10.6	9.44
nfc/ngc	124/114	124/114	124/114	93/ 93	66/ 66	58/ 58

TABLE 20
Problem 6.2 *with* **sqp-pod***: Performance of the suboptimal control versus varying target values of the gradient minimization* `tol`$_\nabla$ *for* $(\hat{q}, n_t) = (\hat{p}, n_t) = (13, 18)$.

`tol`$_\nabla$	$1.77 \cdot 10^{-4}$	$2.5 \cdot 10^{-4}$	$3.54 \cdot 10^{-4}$	$5 \cdot 10^{-4}$	$7.07 \cdot 10^{-4}$	$1 \cdot 10^{-3}$
$\frac{1}{2}\|\hat{x} - \chi_\heartsuit\|_{L^2}^2$	0.0243	0.0243	0.0243	0.0243	0.0302	0.0347
$\|\hat{u}\|_{L^2}^2$	18.968	18.968	18.968	18.968	9.7463	8.0792
`walltime`	`13.1`	`13.1`	`13.1`	`11.2`	`6.29`	`4.69`
`nfc/ngc`	`131/120`	`131/120`	`131/120`	`108/108`	`67/ 67`	`50/ 50`

Acknowledgment. We thank Joost van Zwieten, codeveloper of *Nutils* [21], for providing benchmarks and valuable insight into space-time discretizations of Burgers' equation.

REFERENCES

[1] M. Barrault, Y. Maday, N. C. Nguyen, and A. T. Patera, *An 'empirical interpolation' method: Application to efficient reduced-basis discretization of partial differential equations*, C. R. Math. Acad. Sci. Paris, 339 (2004), pp. 667–672.

[2] M. Baumann, P. Benner, and J. Heiland, *A generalized POD space-time Galerkin scheme for parameter dependent dynamical systems*, presented at MoRePaS III—Workshop on "Model Reduction for Parametrized Systems," 2015, https://doi.org/10.14293/P2199-8442.1.SOP-MATH.P8ECXQ.v1.

[3] M. Baumann, J. Heiland, and M. Schmidt, *Discrete input/output maps and their relation to Proper Orthogonal Decomposition*, in Numerical Algebra, Matrix Theory, Differential-Algebraic Equations and Control Theory, P. Benner, M. Bollhöfer, D. Kressner, C. Mehl, and T. Stykel, eds., Springer, Berlin, 2015, pp. 585–608, https://doi.org/10.1007/978-3-319-15260-8_21.

[4] P. T. Boggs and J. W. Tolle, *Sequential quadratic programming for large-scale nonlinear optimization*, J. Comput. Appl. Math., 124 (2000), pp. 123–137, https://doi.org/10.1016/S0377-0427(00)00429-5.

[5] C. L. Bottasso, *A new look at finite elements in time: A variational interpretation of Runge-Kutta methods*, Appl. Numer. Math., 25 (1997), pp. 355–368, https://doi.org/10.1016/S0168-9274(97)00072-X.

[6] K. Carlberg, J. Ray, and B. van Bloemen Waanders, *Decreasing the temporal complexity for nonlinear, implicit reduced-order models by forecasting*, Comput. Methods Appl. Mech. Engrg., 289 (2015), pp. 79–103.

[7] S. Chaturantabut and D. C. Sorensen, *Nonlinear model reduction via discrete empirical interpolation*, SIAM J. Sci. Comput., 32 (2010), pp. 2737–2764, https://doi.org/10.1137/090766498.

[8] F. Chinesta, A. Ammar, A. Leygue, and R. Keunings, *An overview of the Proper Generalized Decomposition with applications in computational rheology*, J. Non-Newtonian Fluid Mech., 166 (2011), pp. 578–592.

[9] T. A. Davis, *Direct Methods for Sparse Linear Systems*, Fundam. Algorithms 2, SIAM, Philadelphia, 2006.

[10] P. E. Gill and E. Wong, *Sequential quadratic programming methods*, in Mixed Integer Nonlinear Programming, Springer, Berlin, 2012, pp. 147–224.

[11] J. Heiland, *spacetime-genpod-burgers–Python Module for space-time-parameter Generalized POD for Burgers Equation*, https://gitlab.mpi-magdeburg.mpg.de/heiland/spacetime-genpod-burgers, 2015.

[12] M. Heinkenschloss, *Numerical Solution of Implicitly Constrained Optimization Problems*, Technical Report TR08-05, Department of Computational and Applied Mathematics, Rice University, 2008.

[13] B. N. Khoromskij and C. Schwab, *Tensor-structured Galerkin approximation of parametric and stochastic elliptic PDEs*, SIAM J. Sci. Comput., 33 (2011), pp. 364–385, https://doi.org/10.1137/100785715.

[14] K. Kunisch and S. Volkwein, *Control of the Burgers equation by a reduced-order approach using proper orthogonal decomposition*, J. Optim. Theory Appl., 102 (1999), pp. 345–371.

[15] A. Logg, K. B. Ølgaard, M. E. Rognes, and G. N. Wells, *FFC: The FEniCS form compiler*, in Automated Solution of Differential Equations by the Finite Element Method, Springer, Berlin, 2012, pp. 227–238, https://doi.org/10.1007/978-3-642-23099-8_11.

[16] J. More, B. Garbow, and K. Hillstrom, *User guide for MINPACK-*1, Technical Report ANL-80-74, Argonne National Laboratory, 1980.

[17] S. Murman, L. Diosady, A. Garai, and M. Ceze, *A space-time Discontinuous-Galerkin approach for separated flows*, Paper 2016-1059, AIAA, 2016.

[18] T. Schindler and V. Acary, *Timestepping schemes for nonsmooth dynamics based on discontinuous Galerkin methods: Definition and outlook*, Math. Comput. Simulation, 95 (2014), pp. 180–199, https://doi.org/10.1016/j.matcom.2012.04.012.

[19] C. Schwab and R. Stevenson, *Space-time adaptive wavelet methods for parabolic evolution problems*, Math. Comp., 78 (2009), pp. 1293–1318, https://doi.org/10.1090/S0025-5718-08-02205-9.

[20] F. Tröltzsch, *Optimal Control of Partial Differential Equations. Theory, Methods and Applications*, AMS, Providence, RI, 2010.

[21] G. van Zwieten, C. Verhoosel, J. van Zwieten, T. van Opstal, and W. Hoitinga, *Nutils v*2.0, Zenodo, 2016, http://doi.org/10.5281/zenodo.822381.

[22] J. Van Zwieten, R. A. Henkes, D. R. Van Der Heul, P. I. Rosen Esquivel, B. Sanderse, and C. Vuik, *Space-time hp-adaptive DG-FEM schemes for one-dimensional multiphase flow models*, in Proceedings of the 10th International Conference on CFD in Oil & Gas, Metallurgical and Process Industries, 2014, pp. 491–500.

[23] S. Volkwein, *Model Reduction Using Proper Orthogonal Decomposition*, Lecture notes, Institute of Mathematics and Scientific Computing, University of Graz, 2011, http://www.uni-graz.at/imawww/volkwein/POD.pdf.

[24] S. Volkwein and S. Weiland, *An algorithm for Galerkin projections in both time and spatial coordinates*, in Proceedings of the 17th MTNS, (2006).

[25] M. Yano, A. T. Patera, and K. Urban, *A space-time hp-interpolation-based certified reduced basis method for Burgers' equation*, Math. Models Methods Appl. Sci., 24 (2014), pp. 1903–1935, https://doi.org/10.1142/S0218202514500110.