REDUCED ORDER MODELING FOR TIME-DEPENDENT OPTIMIZATION PROBLEMS WITH INITIAL VALUE CONTROLS*

MATTHIAS HEINKENSCHLOSS[†] AND DÖRTE JANDO[‡]

Abstract. This paper presents a new reduced order model (ROM) Hessian approximation for linear-quadratic optimal control problems where the optimal control is the initial value. Such problems arise in parameter identification and data assimilation, where the parameters to be identified appear in the initial data, and also as subproblems in multiple shooting formulations of more general optimal control problems. The new ROM Hessians can provide a substantially better approximation than the underlying basic ROM approximation, and thus can substantially reduce the computing time needed to solve these optimal control problems. The computation of a Hessian vector product requires the solution of the linearized state equation with initial value given by the vector to which the Hessian is applied, followed by the solution of the second order adjoint equation. Projection-based ROMs of these two linear differential equations are used to generate the Hessian approximation while the objective function and gradient are computed exactly using the full model. The challenge is that in general no fixed ROM well-approximates the application of the Hessian to all possible vectors of initial data. The new approach, after having selected a basic ROM, augments this basic ROM by one vector. This vector is either the right-hand side or the vector of initial data to which the Hessian is applied. It is shown that although the size of the ROM increases only by one, this new augmented ROM produces substantially better approximations of the true Hessian vector products as well as of the optimal solution than the basic ROM. The use of these ROM Hessians in a conjugate gradient method is analyzed.

Key words. optimal control, reduced order models, Hessian approximation, initial-condition problems

AMS subject classifications. 49M05, 65F10, 65F15, 93A15

DOI. 10.1137/16M1109084

1. Introduction. We present a new reduced order model (ROM) approach to efficiently compute approximate Hessian information for the following class of large-scale linear quadratic optimal control problems with initial value controls. Given symmetric positive definite matrices $\mathbf{M}, \mathbf{R} \in \mathbb{R}^{n \times n}$, symmetric positive semidefinite matrices $\mathbf{Q}, \mathbf{Q}_T \in \mathbb{R}^{n \times n}$, a matrix $\mathbf{A} \in \mathbb{R}^{n \times n}$, functions $\mathbf{c}, \mathbf{f} \in L^2(0, T; \mathbb{R}^n)$, and a vector $\mathbf{c}_T \in \mathbb{R}^n$, we consider the following linear quadratic optimal control problem:

$$(1.1a) \quad \begin{aligned} \text{minimize } J(\mathbf{s}) = & \int_0^T \frac{1}{2} \mathbf{y}(t)^T \mathbf{Q} \mathbf{y}(t) + \mathbf{c}(t)^T \mathbf{y}(t) dt \\ & + \frac{1}{2} \mathbf{y}(T)^T \mathbf{Q}_T \mathbf{y}(T) + \mathbf{c}_T^T \mathbf{y}(T) + \frac{1}{2} \mathbf{s}^T \mathbf{R} \mathbf{s}, \end{aligned}$$

*Submitted to the journal's Methods and Algorithms for Scientific Computing section December 22, 2016; accepted for publication (in revised form) September 5, 2017; published electronically January 5, 2018.

<http://www.siam.org/journals/sisc/40-1/M110908.html>

Funding: This work was supported in part by NSF grant DMS-1522798, by the DARPA EQUiPS Program, Award UTA15-001068, and by the Excellence Initiative of the German Federal Government and the state governments—funding line Institutional Strategy: DFG project number ZUK 49/2 (“Heidelberg: Realising the Potential of a Comprehensive University”)—via the Heidelberg Mobility Programme.

[†]Department of Computational and Applied Mathematics, MS-134, Rice University, 6100 Main Street, Houston, TX 77005-1892 (heinken@rice.edu).

[‡]Interdisciplinary Center for Scientific Computing (IWR), Heidelberg University, Im Neuenheimer Feld 205, D-69120 Heidelberg, Germany (doerte.jando@iwr.uni-heidelberg.de).

in $\mathbf{s} \in \mathbb{R}^n$, where for given $\mathbf{s}$ the state $\mathbf{y} \in H^1(0, T; \mathbb{R}^n)$ solves

$$(1.1b) \quad \mathbf{M} \frac{d}{dt} \mathbf{y}(t) + \mathbf{A} \mathbf{y}(t) = \mathbf{f}(t), \quad t \in (0, T),$$

$$(1.1c) \quad \mathbf{M} \mathbf{y}(0) = \mathbf{M} \mathbf{s}.$$

We target large-scale problems (1.1) that arise, e.g., from semidiscretizations of optimal control problems governed by partial differential equations (PDEs). Problems of the type (1.1) arise, e.g., in source inversion problems (see, e.g., Akçelik et al. [1], Bashir et al. [5]), as subproblems in data assimilation (see, e.g., Blum, Le Dimet, and Navon [9, sect. 6], Daescu and Navon [12], Rao and Sandu [32], Ștefănescu, Sandu, and Navon [35]), or as subproblems in multiple shooting formulations of optimal control problems (see, e.g., Hesse and Kanschat [19] or Carraro and Geiger [11]). We are motivated by the need to construct faster solvers for these problems, especially for multiple shooting formulations of optimal control problems, and this work for the specific problem (1.1) is a first step to address this need.

A major difficulty in applying projection-based ROMs to the solution of (1.1) arises from the fact that the initial data $\mathbf{s}$ vary in $\mathbb{R}^n$. Because of the large dimension n of the input space and the unknown impact of the initial data on the state $\mathbf{y}$ it is difficult to generate ROMs for these problems. Instead of generating a ROM and replacing (1.1) by its ROM approximation, we compute the objective function J and its gradient ∇J using the original large-scale problem and use ROMs to approximate Hessian information. Since the solution $\mathbf{s}_*$ of (1.1) is characterized by $\nabla J(\mathbf{s}_*) = \mathbf{0}$, the ROMs in our approach do not impact this equation, but are used to accelerate the method with which this equation is solved. The generation of efficient ROMs for Hessian approximations is still challenging and the main focus of this paper.

The problem (1.1) can be solved using the conjugate gradient (CG) method on the optimality condition. Each iteration of the CG method requires the computation of the Hessian $\nabla^2 J$ times a vector of initial data, which in turn requires the expensive sequential solution of two large-scale dynamical systems related to (1.1). To reduce the computational expense, we replace these two dynamical systems with projection-based ROMs which leads to an approximate Hessian. While projection-based ROMs are well established for large classes of linear time invariant systems of the type (1.1b), (1.1c) (see, e.g., the books by Antoulas [3], Quarteroni, Manzoni, and Negri [31], Hesthaven, Rozza, and Stamm [20], and the review article by Benner, Gugercin, and Willcox [7]), they mostly address systems with a small number of inputs in the right-hand side of (1.1b) and fixed initial data (often zero initial data). As mentioned earlier, the difficulty we face is that our initial data vary during the execution of the CG method and the ROM has to capture all possible initial data. To our knowledge this issue has not previously been addressed in the ROM literature. Our new approach, after having selected a basic ROM, augments this basic ROM for each given vector of initial data by adding one basis vector. The resulting ROM is of fixed size, but is generally nonlinear in the vector of initial data. We prove that our new augmented ROM can produce substantially better approximations than the basic ROM. We also show how to use these ROMs in a CG method to solve (1.1). In important special cases, the structures of the problem and of the CG method imply that the original basic ROM is only augmented once, rather than in each CG iteration.

ROMs have been used for some optimal control problems related to (1.1), and they have been used to approximate dynamical systems with inhomogeneous initial values. Next, we will review approaches most closely related to our work and highlight the contributions of our paper over previous work.

There are a few approaches that deal with model reduction of systems with varying initial conditions. The arguably closest (in focus but not approach) paper to ours is Bashir et al. [5].¹ They consider (1.1) with $\mathbf{Q} = \mathbf{C}^T \mathbf{C}$, $\mathbf{Q}_T = \mathbf{C}_T^T \mathbf{C}_T$ (since $\mathbf{Q}, \mathbf{Q}_T$ are positive semidefinite this representation is possible in principle), and $\mathbf{f} = \mathbf{0}$, and they construct a projection-based ROM, such that the error between the outputs $\mathbf{C}\mathbf{y}(t)$, $t \in (0, T)$, $\mathbf{C}_T\mathbf{y}(T)$, of the original system (1.1b), (1.1c), and the outputs of the ROM system is small over a set of initial data. They use a greedy approach which involves the computation of the largest eigenvalues of the Hessian of (1.1a) and associated eigenvectors. The numerical examples in [5] are for a problem where $\mathbf{C}, \mathbf{C}_T$ have a small number of rows, which means only a small number of eigenvalues of the Hessian are relevant, which is not necessarily the case for our target problems. Their approach could be combined with ours by computing a basic ROM following [5] and then augmenting this ROM using our approach. In fact, as we show in section 4, our approach for computing a basic ROM, although different from [5] in general, in special cases also computes the largest eigenvalues and corresponding eigenvectors of the Hessian. The main innovation of our paper is the augmentation of these basic ROMs.

The use of proper orthogonal decomposition (POD)-based ROMs in data assimilation for geophysical fluids is reviewed by Blum, Le Dimet, and Navon [9, sect. 6] and different recent approaches to incorporate POD ROMs are evaluated in Ștefănescu, Sandu, and Navon [35]. The ROM approaches in [9, 35] are iterative. Given a guess of the optimal initial data, the full order state, adjoint equations, and cost function gradient are computed, and a POD ROM is generated from this information. Then a ROM approximation of the data assimilation problem is solved, and the process is repeated. Trust-region model management (see, e.g., [4], [25, sect. 4.1], [34, sect. 3.2]), provides a framework that in principle can be applied to ensure convergence. However, the rigorous application of (especially earlier versions of) this framework requires exact error bounds on objective function and gradient approximations, which is still an issue. Nevertheless, numerically, the ROM approach produces good results on a nonlinear shallow water equation data assimilation problem [35].

ROMs have been used for optimal control problems with controls in the right-hand side, but fixed initial data. See, e.g., the reviews by Gubisch and Volkwein [16], Sachs and Volkwein [34]), and Benner, Sachs, and Volkwein [8]. It is important that ROMs are built so that the objective function and gradient generated using the ROM well-approximate the full order model objective function and gradient. For some optimal control problems with controls in the right-hand side ROMs can be constructed that achieve this approximation property for all control inputs (see, e.g., Benner, Sachs, and Volkwein [8, sect. 4], Antil et al. [2]). In other cases, iterative methods are used that adjust the ROM as the optimization progresses using, e.g., a trust-region framework (see, e.g., Benner, Sachs, and Volkwein [8, sect. 3.3], Sachs and Volkwein [34, sect. 3.1]). In principle the latter approaches could be adjusted to our setting. However, the critical issue remains: how to construct ROMs for varying initial data so that not only objective function and gradient information are approximated, but additional information is captured that makes the ROMs truly efficient? This is the focus of this paper. We note that since we use ROMs only for Hessian approximation, we are not faced with the question of quality of the ROMs for objective and gradient approximation.

¹Bashir et al. [5] consider a discrete time system and $\mathbf{M} = \mathbf{I}$, but their approach can conceptually be extended to the continuous in time case, which is how we present it here.

Heinkenschloss, Reis, and Antoulas [18] study the extension of balanced truncation model reduction (BTMR), which like some other ROM approaches originally assumed zero initial data, to problems with arbitrary but fixed initial data. The approach in [18] is to augment the input matrix by the initial data. Beattie, Gugercin, and Mehrmann [6] use the principle of superposition and compute one ROM for the component of the solution with homogeneous initial condition and varying right-hand side and another ROM for the component of the solution with initial condition $\mathbf{s}$ and zero right-hand side. Both approaches [18] and [6] work for fixed initial data, or initial data in a small dimensional subspace, but become impractical when the initial data can be any vector in $\mathbb{R}^n$, as in our case.

Finally, since the solution of dynamical systems (1.1b), (1.1c) is related to the matrix exponential, one could think about using approximation techniques for the matrix exponential, such as those described, e.g., in Caliarì et al. [10], Frommer and Simoncini [13], or Hochbruck and Lubich [22]. These techniques approximate the matrix exponential applied to fixed initial data. Since our initial data change in every iteration of the CG method applied to (1.1), a new approximation would have to be computed in every iteration of the CG method, which seems excessive. The goal of this paper is to present an approach that efficiently augments a fixed underlying ROM to adjust for the changing initial data.

The paper is organized as follows. Section 2 reviews the optimality conditions and Hessian computations for (1.1), and provides the basic structure for our ROM Hessian approximation. Section 3 specifies a model optimal control problem governed by an advection diffusion reaction PDE, which after finite element semi-discretization in space leads to a problem of the form (1.1), which will be used to illustrate our approach. Section 4 describes our approach to construct the ROM in a special case where $\mathbf{A}$ is symmetric and where after a change of basis for $\mathbf{y}$ the problem (1.1) decouples into n real, scalar optimization problems. This case allows a fairly complete analysis of our ROM augmentation. The basis ROM with $\mathbf{V} = \mathbf{W} \in \mathbb{R}^{n \times k}$, $k \ll n$, is obtained by taking the eigenvalues corresponding to the smallest generalized eigenvalues of $(\mathbf{A}, \mathbf{M})$. We present and analyze two ROM augmentations. The first one augments the ROM once for all initial data by $\mathbf{g} = \nabla J(\mathbf{0})$. This leads to a ROM Hessian approximation that is independent of the initial data. The second one augments the ROM by the initial data, which leads to a ROM Hessian approximation that is nonlinear in the initial data. We provide results on the improvements gained by using these ROM augmentations. The generalization of our approach to compute ROM Hessian approximations for general problems is presented in section 5. Section 6 studies the CG method with ROM Hessian approximations. We review a CG convergence result suitable for our problem structure. More importantly, we show that for an important special case our nonlinear ROM Hessian approximation, which is obtained by augmenting the ROM by the initial data in every step of the CG algorithm, actually is a fixed, linear ROM Hessian approximation and therefore the standard CG algorithm can be used. Section 7 illustrates the performance of our ROM Hessian approximation for several variations of the model problem of section 3.

2. Optimality conditions and Hessian computations. This section reviews the computation of the gradient and the Hessian of the quadratic, strictly convex objective function (1.1a) and provides the basic framework for our reduced basis Hessian approximation.

Given a symmetric positive definite matrix $\mathbf{G} \in \mathbb{R}^{n \times n}$, we endow $\mathbb{R}^n$ with the weighted inner product

$$(2.1) \quad \langle \mathbf{s}_1, \mathbf{s}_2 \rangle = \mathbf{s}_1^T \mathbf{G} \mathbf{s}_2$$

and corresponding norm. Ultimately, $\mathbf{G}$ will be related to $\mathbf{R}$ or $\mathbf{M}$, but for now it can be an arbitrary symmetric positive definite matrix. The gradient and Hessian of J are computed with respect to this inner product. The adjoint of a matrix $\mathbf{H}$ with respect to the inner product (2.1) is $\mathbf{H}^* = \mathbf{G}^{-1} \mathbf{H}^T \mathbf{G}$.

The state space for (1.1b), (1.1c), is $H^1(0, T; \mathbb{R}^n)$, the space of vector valued functions such that each component belongs to the Sobolev space $H^1(0, T)$.

2.1. Full order problem. Since the function (1.1a) is quadratic and strictly convex, (1.1) has a unique solution $\mathbf{s}_*$ and this solution is characterized by the condition $\nabla J(\mathbf{s}_*) = \mathbf{0}$. The gradient of (1.1a) can be computed via the adjoint equation approach (see, e.g., Hinze et al. [21, Chap. 1], or Liberzon [27]) and is given by

$$(2.2) \quad \nabla J(\mathbf{s}) = \mathbf{G}^{-1} \mathbf{M} \mathbf{p}(0) + \mathbf{G}^{-1} \mathbf{R} \mathbf{s} \in \mathbb{R}^n,$$

where $\mathbf{p} \in H^1(0, T; \mathbb{R}^n)$ is the solution of the adjoint equation

$$(2.3a) \quad -\mathbf{M} \frac{d}{dt} \mathbf{p}(t) + \mathbf{A}^T \mathbf{p}(t) = \mathbf{Q} \mathbf{y}(t) + \mathbf{c}(t), \quad t \in (0, T),$$

$$(2.3b) \quad \mathbf{M} \mathbf{p}(T) = \mathbf{Q}_T \mathbf{y}(T) + \mathbf{c}_T.$$

Since J is quadratic its Hessian is independent of $\mathbf{s}$ and $\nabla J(\mathbf{s}) = \nabla^2 J \mathbf{s} + \nabla J(\mathbf{0})$. The Hessian is given by

$$\nabla^2 J = \mathbf{H} + \mathbf{G}^{-1} \mathbf{R},$$

where the application of $\mathbf{H}$ to a vector $\mathbf{v} \in \mathbb{R}^n$ is given by

$$(2.4) \quad \mathbf{H} \mathbf{v} = \mathbf{G}^{-1} \mathbf{M} \mathbf{q}(0),$$

and $\mathbf{q} \in H^1(0, T; \mathbb{R}^n)$ is obtained by first solving

$$(2.5a) \quad \mathbf{M} \frac{d}{dt} \mathbf{z}(t) + \mathbf{A} \mathbf{z}(t) = \mathbf{0}, \quad t \in (0, T),$$

$$(2.5b) \quad \mathbf{M} \mathbf{z}(0) = \mathbf{M} \mathbf{v},$$

and then

$$(2.6a) \quad -\mathbf{M} \frac{d}{dt} \mathbf{q}(t) + \mathbf{A}^T \mathbf{q}(t) = \mathbf{Q} \mathbf{z}(t), \quad t \in (0, T),$$

$$(2.6b) \quad \mathbf{M} \mathbf{q}(T) = \mathbf{Q}_T \mathbf{z}(T).$$

Later we will determine the unique solution $\mathbf{s}_*$ of (1.1) by applying the CG method to

$$(2.7) \quad \nabla^2 J \mathbf{s}_* = -\nabla J(\mathbf{0}).$$

We can give an explicit expression for the Hessian component $\mathbf{H}$, which will be useful for analysis. The solutions of (2.5) and (2.6) are

$$(2.8) \quad \mathbf{z}(t) = \exp(-\mathbf{M}^{-1} \mathbf{A} t) \mathbf{s},$$

$$\mathbf{q}(t) = \exp(-\mathbf{M}^{-1} \mathbf{A}^T (T - t)) \mathbf{M}^{-1} \mathbf{Q}_T \mathbf{z}(T)$$

$$(2.9) \quad + \int_t^T \exp(-\mathbf{M}^{-1} \mathbf{A}^T (\tau - t)) \mathbf{M}^{-1} \mathbf{Q} \mathbf{z}(\tau) d\tau,$$

respectively. Therefore, $\mathbf{H}\mathbf{s} = \mathbf{G}^{-1}\mathbf{M}\mathbf{q}(0)$ gives the expression

$$(2.10) \quad \begin{aligned} \mathbf{H} &= \mathbf{G}^{-1}\mathbf{M} \exp(-\mathbf{M}^{-1}\mathbf{A}^T T) \mathbf{M}^{-1}\mathbf{Q}_T \exp(-\mathbf{M}^{-1}\mathbf{A} T) \\ &+ \int_0^T \mathbf{G}^{-1}\mathbf{M} \exp(-\mathbf{M}^{-1}\mathbf{A}^T \tau) \mathbf{M}^{-1}\mathbf{Q} \exp(-\mathbf{M}^{-1}\mathbf{A} \tau) d\tau. \end{aligned}$$

Since

$$\begin{aligned} \mathbf{H}^T &= \exp(-\mathbf{A}^T \mathbf{M}^{-1} T) \mathbf{Q}_T \mathbf{M}^{-1} \exp(-\mathbf{A} \mathbf{M}^{-1} T) \mathbf{M} \mathbf{G}^{-1} \\ &+ \int_0^T \exp(-\mathbf{A}^T \mathbf{M}^{-1} \tau) \mathbf{Q} \mathbf{M}^{-1} \exp(-\mathbf{A} \mathbf{M}^{-1} \tau) \mathbf{M} \mathbf{G}^{-1} d\tau \end{aligned}$$

and $\exp(-\mathbf{C}\mathbf{N}^{-1}) = \mathbf{N}\mathbf{N}^{-1} \exp(-\mathbf{C}\mathbf{N}^{-1}) = \mathbf{N} \exp(-\mathbf{N}^{-1}\mathbf{C})\mathbf{N}^{-1}$ for any square matrices $\mathbf{C}, \mathbf{N}$ with $\mathbf{N}$ nonsingular, we confirm that $\mathbf{G}^{-1}\mathbf{H}^T\mathbf{G} = \mathbf{H}$, i.e., that $\mathbf{H}$ is in fact self-adjoint. Furthermore, since $\mathbf{Q}, \mathbf{Q}_T$ are symmetric positive semidefinite, (2.10) immediately reveals that $\langle \mathbf{s}, \mathbf{H}\mathbf{s} \rangle \geq 0$ for all $\mathbf{s} \in \mathbb{R}^n$.

2.2. ROM Hessian approximation. Each iteration of the CG method applied to the solution of (2.7) requires the computation of a Hessian-times-vector product, i.e., each iteration requires the solution of (2.5) and (2.6). To reduce the computational cost of these two simulations we will use model order reduction.

Let the matrices for model order reduction be $\mathbf{W}, \mathbf{V} \in \mathbb{R}^{n \times k}$, $k < n$, such that $\mathbf{W}^T \mathbf{M} \mathbf{V}$ is invertible. Often $\mathbf{W}^T \mathbf{M} \mathbf{V} = \mathbf{I}_k$, but this is not necessary. We will discuss later in sections 4.3 and 5 how we compute the matrices $\mathbf{W}, \mathbf{V}$. First we describe how, for given $\mathbf{W}$ and $\mathbf{V}$, our ROM approximation $\hat{\mathbf{H}}$ to the Hessian is computed.

The ROM approximations of the solutions to (2.5) and (2.6) are given by $\mathbf{V}\hat{\mathbf{z}}$ and $\mathbf{W}\hat{\mathbf{q}}$, respectively, where $\hat{\mathbf{z}}, \hat{\mathbf{q}}$ solve

$$(2.11a) \quad \mathbf{W}^T \mathbf{M} \mathbf{V} \frac{d}{dt} \hat{\mathbf{z}}(t) + \mathbf{W}^T \mathbf{A} \mathbf{V} \hat{\mathbf{z}}(t) = \mathbf{0}, \quad t \in (0, T),$$

$$(2.11b) \quad \mathbf{W}^T \mathbf{M} \mathbf{V} \hat{\mathbf{z}}(0) = \mathbf{W}^T \mathbf{M} \mathbf{s}$$

and

$$(2.12a) \quad -\mathbf{V}^T \mathbf{M} \mathbf{W} \frac{d}{dt} \hat{\mathbf{q}}(t) + \mathbf{V}^T \mathbf{A}^T \mathbf{W} \hat{\mathbf{q}}(t) = \mathbf{V}^T \mathbf{Q} \mathbf{V} \hat{\mathbf{z}}(t), \quad t \in (0, T),$$

$$(2.12b) \quad \mathbf{V}^T \mathbf{M} \mathbf{W} \hat{\mathbf{q}}(T) = \mathbf{V}^T \mathbf{Q}_T \mathbf{V} \hat{\mathbf{z}}(T).$$

The ROM approximation of (2.4) is given by

$$(2.13) \quad \hat{\mathbf{H}}\mathbf{s} = \mathbf{G}^{-1} \mathbf{M} \mathbf{W} \hat{\mathbf{q}}(0).$$

Define

$$\hat{\mathbf{M}} = \mathbf{W}^T \mathbf{M} \mathbf{V}, \quad \hat{\mathbf{A}} = \mathbf{W}^T \mathbf{A} \mathbf{V}.$$

Analogously to (2.10) we obtain the explicit representation

$$(2.14) \quad \begin{aligned} \hat{\mathbf{H}} &= \mathbf{G}^{-1} \mathbf{M} \mathbf{W} \exp(-\hat{\mathbf{M}}^{-T} \hat{\mathbf{A}}^T T) \hat{\mathbf{M}}^{-T} \mathbf{V}^T \mathbf{Q}_T \mathbf{V} \exp(-\hat{\mathbf{M}}^{-1} \hat{\mathbf{A}} T) \hat{\mathbf{M}}^{-1} \mathbf{W}^T \mathbf{M} \\ &+ \int_0^T \mathbf{G}^{-1} \mathbf{M} \mathbf{W} \exp(-\hat{\mathbf{M}}^{-T} \hat{\mathbf{A}}^T \tau) \hat{\mathbf{M}}^{-T} \mathbf{V}^T \mathbf{Q} \mathbf{V} \\ &\times \exp(-\hat{\mathbf{M}}^{-1} \hat{\mathbf{A}} \tau) \hat{\mathbf{M}}^{-1} \mathbf{W}^T \mathbf{M} d\tau, \end{aligned}$$

of the ROM approximation. Since

$$\begin{aligned}\widehat{\mathbf{H}}^T &= \mathbf{M}\mathbf{W}\widehat{\mathbf{M}}^{-T} \exp(-\widehat{\mathbf{A}}^T \widehat{\mathbf{M}}^{-T} T) \mathbf{V}^T \mathbf{Q}_T \mathbf{V} \widehat{\mathbf{M}}^{-1} \exp(-\widehat{\mathbf{A}} \widehat{\mathbf{M}}^{-1} T) \mathbf{W}^T \mathbf{M} \mathbf{G}^{-1} \\ &\quad + \int_0^T \mathbf{M}\mathbf{W}\widehat{\mathbf{M}}^{-T} \exp(-\widehat{\mathbf{A}}^T \widehat{\mathbf{M}}^{-T} \tau) \mathbf{V}^T \mathbf{Q} \mathbf{V} \widehat{\mathbf{M}}^{-1} \\ &\quad \times \exp(-\widehat{\mathbf{A}} \widehat{\mathbf{M}}^{-1} \tau) \mathbf{W}^T \mathbf{M} \mathbf{G}^{-1} d\tau,\end{aligned}$$

we have $\mathbf{G}^{-1} \widehat{\mathbf{H}}^T \mathbf{G} = \widehat{\mathbf{H}}$, i.e., the ROM approximation $\widehat{\mathbf{H}}$ is self-adjoint.

The representation (2.14) shows that $\langle \mathbf{s}, \widehat{\mathbf{H}} \mathbf{s} \rangle \geq 0$ for all $\mathbf{s} \in \mathbb{R}^n$. Since $\mathbf{W}, \mathbf{V} \in \mathbb{R}^{n \times k}$, $k < n$, the ROM approximation of the expensive Hessian component has at most rank k ,

$$(2.15) \quad \text{rank}(\widehat{\mathbf{H}}) \leq k.$$

Before we discuss how we construct the matrices $\mathbf{W}, \mathbf{V} \in \mathbb{R}^{n \times k}$, $k < n$, we state a model problem that will be used for illustrations throughout this paper.

3. Model problem. We will use the following model problem in this paper. Let $\Omega = (0, 1)^3$ be the spatial domain, $T = 1$, and let $\Omega_o \subset \Omega$ be the observation region. Given scalars $\kappa > 0$, $\beta > 0$, $\gamma \geq 0$, the advection $a \in \mathbb{R}^2$, and functions $y_d \in L^2(\Omega_o \times (0, T))$, $f \in L^2(\Omega \times (0, T))$, the optimal control problem is

$$(3.1a) \quad \min_{s \in L^2(\Omega)} \frac{1}{2} \int_0^T \int_{\Omega_o} (y(x, t) - y_d(x, t))^2 dx dt + \frac{1}{2} \int_{\Omega_o} (y(x, T) - y_d(x, T))^2 dx + \frac{\beta}{2} \int_{\Omega} s^2(x) dx,$$

where for given $s \in L^2(\Omega)$, $y \in W(0, T)$ is the solution of

$$(3.1b) \quad \frac{\partial y(x, t)}{\partial t} - \kappa \Delta y(x, t) + a \cdot \nabla y(x, t) + \gamma y(x, t) = f(x, t), \quad x \in \Omega, t \in (0, T),$$

$$(3.1c) \quad y(x, t) = 0, \quad x \in \partial\Omega, t \in (0, T),$$

$$(3.1d) \quad y(x, 0) = s(x), \quad x \in \Omega.$$

The existence and uniqueness of a solution $s^* \in L^2(\Omega)$ of (3.1) can be established using standard techniques. See, e.g., Hinze et al. [21, Chap. 1], or Tröltzsch [36].

We discretize (3.1) using P1 piecewise linear finite elements in space. More precisely, we divide $\Omega = (0, 1)^3$ into $n_{x_1} \times n_{x_2} \times n_{x_3}$ cubes of equal size and then divide each cube into six tetrahedra. This leads to a problem of the type (1.1).

The desired/observed states y_d and the resulting $\mathbf{c}, \mathbf{c}_T$ in (1.1a) are computed by setting

$$s(x) = 2e^{-10\|x-x_1\|_2^2} + e^{-5\|x-x_2\|_2^2} + 2e^{-50\|x-x_3\|_2^2} + 2e^{-40\|x-x_4\|_2^2} + e^{-100\|x-x_5\|_2^2},$$

where $x_1 = (0.2, 0.2, 0.2)$, $x_2 = (0.8, 0.8, 0.8)$, $x_3 = (0.5, 0.5, 0.5)$, $x_4 = (0.2, 0.8, 0.8)$, $x_5 = (0.8, 0.2, 0.2)$, solving the discretized state equation (1.1b), (1.1c), and using the solution as the desired state.

4. Augmented ROM Hessians for decoupled scalar problems. It will be illustrative to consider a special case before we describe how to compute the matrices $\mathbf{V}, \mathbf{W}$ used in the construction of the ROM Hessian approximation (2.11)–(2.13) in

the general case. The special case is the problem (1.1) with symmetric $\mathbf{A}$ and matrices in the objective function satisfying the assumptions (4.1a) below. For example, the model problem (3.1) without advection, $a = (0, 0)^T$, and observation in the entire domain, $\Omega_o = \Omega$, falls into this class. As we will see, in this case, after a change of variables, problem (1.1) decouples into n real scalar problems.

Assume that

$$(4.1a) \quad \mathbf{Q} = \mathbf{Q}_T = \mathbf{M}, \quad \mathbf{R} = \beta \mathbf{G}, \quad \mathbf{G} = \mathbf{M}$$

with $\beta > 0$, and assume the existence of matrices $\mathbf{V}_n \in \mathbb{R}^{n \times n}$, $\mathbf{\Lambda} = \text{diag}(\lambda_1, \dots, \lambda_n) \in \mathbb{R}^{n \times n}$,

$$(4.1b) \quad \lambda_1 \leq \dots \leq \lambda_n,$$

such that

$$(4.1c) \quad \mathbf{V}_n^T \mathbf{M} \mathbf{V}_n = \mathbf{I}, \quad \mathbf{A} \mathbf{V}_n = \mathbf{M} \mathbf{V}_n \mathbf{\Lambda}.$$

Thus, the λ_i 's are the generalized eigenvalues of $(\mathbf{A}, \mathbf{M})$ and the columns of $\mathbf{V}_n$ are the corresponding generalized $\mathbf{M}$ -orthonormal eigenvectors. Note that the identity $\mathbf{V}_n^T \mathbf{M} \mathbf{V}_n = \mathbf{I}$ implies that $\mathbf{V}_n$ is invertible.

As we have noted, model problem (3.1) with $a = (0, 0)^T$ and $\Omega_o = \Omega$ leads to a problem that satisfies assumptions (4.1).

We set $\mathbf{g} = \nabla J(\mathbf{0})$. Under assumptions (4.1) the system $\nabla^2 J \mathbf{s} = -\nabla J(\mathbf{0})$ is given by

$$(4.2) \quad (\mathbf{H} + \beta \mathbf{I}) \mathbf{s} = -\mathbf{g}.$$

4.1. Optimal control problem. We transform the optimal control problem (1.1) into a problem with initial data

$$\tilde{\mathbf{s}} \equiv \mathbf{V}_n^T \mathbf{M} \mathbf{s} \in \mathbb{R}^n$$

and with states $\tilde{\mathbf{y}}$ defined through the identify

$$\mathbf{y}(t) = \mathbf{V}_n \tilde{\mathbf{y}}(t).$$

Note that $\tilde{\mathbf{s}} = \mathbf{V}_n^T \mathbf{M} \mathbf{s} = \mathbf{V}_n^T \mathbf{M} \mathbf{V}_n \mathbf{V}_n^{-1} \mathbf{s} = \mathbf{V}_n^{-1} \mathbf{s}$ implies

$$\mathbf{s} = \mathbf{V}_n \tilde{\mathbf{s}}$$

and $\mathbf{s}^T \mathbf{M} \mathbf{s} = \tilde{\mathbf{s}}^T \tilde{\mathbf{s}}$. More generally, for vectors $\mathbf{s}_1 = \mathbf{V}_n \tilde{\mathbf{s}}_1$, $\mathbf{s}_2 = \mathbf{V}_n \tilde{\mathbf{s}}_2$,

$$\mathbf{s}_1^T \mathbf{M} \mathbf{s}_2 = \tilde{\mathbf{s}}_1^T \mathbf{V}_n^T \mathbf{M} \mathbf{V}_n \tilde{\mathbf{s}}_2 = \tilde{\mathbf{s}}_1^T \tilde{\mathbf{s}}_2.$$

Thus the $\mathbf{M}$ inner product in the $\mathbf{s}$ variables becomes the standard Euclidean inner product in the $\tilde{\mathbf{s}}$ variables.

If, in addition to $\tilde{\mathbf{s}}$ and $\tilde{\mathbf{y}}(t)$, we define

$$\tilde{\mathbf{f}}(t) = \mathbf{V}_n^T \mathbf{f}(t), \quad \tilde{\mathbf{c}}(t) = \mathbf{V}_n^T \mathbf{c}(t), \quad \tilde{\mathbf{c}}_T = \mathbf{V}_n^T \mathbf{c}_T,$$

and use (4.1c), then (1.1) is equivalently transformed into

$$(4.3a) \quad \min_{\tilde{\mathbf{s}} \in \mathbb{R}^n} \tilde{J}(\tilde{\mathbf{s}}) = \int_0^T \frac{1}{2} \tilde{\mathbf{y}}(t)^T \tilde{\mathbf{y}}(t) + \tilde{\mathbf{c}}(t)^T \tilde{\mathbf{y}}(t) dt + \frac{1}{2} \tilde{\mathbf{y}}(T)^T \tilde{\mathbf{y}}(T) + \tilde{\mathbf{c}}_T^T \tilde{\mathbf{y}}(T) + \frac{\beta}{2} \tilde{\mathbf{s}}^T \tilde{\mathbf{s}},$$

where for given $\tilde{\mathbf{s}}$ the state $\tilde{\mathbf{y}} \in H^1(0, T; \mathbb{R}^n)$ solves

$$(4.3b) \quad \frac{d}{dt} \tilde{\mathbf{y}}(t) + \mathbf{A} \tilde{\mathbf{y}}(t) = \tilde{\mathbf{f}}(t), \quad t \in (0, T),$$

$$(4.3c) \quad \tilde{\mathbf{y}}(0) = \tilde{\mathbf{s}}.$$

The optimal control problem (4.3) decouples into n independent scalar optimal control problems

$$(4.4) \quad \min_{\tilde{\mathbf{s}}_i \in \mathbb{R}} \tilde{J}_i(\tilde{\mathbf{s}}_i) = \int_0^T \frac{1}{2} \tilde{\mathbf{y}}_i(t)^2 + \tilde{\mathbf{c}}_i(t) \tilde{\mathbf{y}}_i(t) dt + \frac{1}{2} \tilde{\mathbf{y}}_i(T)^2 + \tilde{\mathbf{c}}_{T,i} \tilde{\mathbf{y}}_i(T) + \frac{\beta}{2} \tilde{\mathbf{s}}_i^2,$$

where for given $\tilde{\mathbf{s}}_i$ the state $\tilde{\mathbf{y}}_i \in H^1(0, T; \mathbb{R})$ solves

$$\begin{aligned} \frac{d}{dt} \tilde{\mathbf{y}}_i(t) + \lambda_i \tilde{\mathbf{y}}_i(t) &= \tilde{\mathbf{f}}_i(t), & t \in (0, T), \\ \tilde{\mathbf{y}}_i(0) &= \tilde{\mathbf{s}}_i. \end{aligned}$$

The gradient of $\tilde{J}$ in (4.3a) (the gradient now is the vector of partial derivatives since the inner product in the $\tilde{\mathbf{s}}$ variables is the standard Euclidean inner product) is given by

$$(4.5) \quad \frac{\partial}{\partial \tilde{\mathbf{s}}_i} \tilde{J}(\tilde{\mathbf{s}}) = \tilde{\mathbf{p}}_i(0) + \beta \tilde{\mathbf{s}}_i,$$

where $\tilde{\mathbf{p}}$ is the solution of the adjoint equation

$$(4.6a) \quad -\frac{d}{dt} \tilde{\mathbf{p}}_i(t) + \lambda_i \tilde{\mathbf{p}}_i(t) = \tilde{\mathbf{y}}_i(t) + \tilde{\mathbf{c}}_i(t), \quad t \in (0, T),$$

$$(4.6b) \quad \tilde{\mathbf{p}}_i(T) = \tilde{\mathbf{y}}_i(T) + \tilde{\mathbf{c}}_{T,i}.$$

We set

$$(4.7) \quad \tilde{\mathbf{g}} = \nabla \tilde{J}(\mathbf{0}).$$

The Hessian of $\tilde{J}$ is diagonal and independent of $\tilde{\mathbf{s}}$,

$$\nabla^2 \tilde{J} = \tilde{\mathbf{H}} + \beta \mathbf{I},$$

where

$$\tilde{\mathbf{H}} = \text{diag}(\tilde{\mathbf{H}}_{11}, \dots, \tilde{\mathbf{H}}_{nn}).$$

The application of $\tilde{\mathbf{H}}$ to a vector $\tilde{\mathbf{s}}$ is given by

$$(4.8) \quad \tilde{\mathbf{H}} \tilde{\mathbf{s}} = \tilde{\mathbf{q}}(0),$$

where $\tilde{\mathbf{q}}$ is obtained by first solving

$$(4.9a) \quad \frac{d}{dt} \tilde{\mathbf{z}}(t) + \mathbf{A} \tilde{\mathbf{z}}(t) = 0, \quad t \in (0, T),$$

$$(4.9b) \quad \tilde{\mathbf{z}}(0) = \tilde{\mathbf{s}},$$

and then

$$(4.10a) \quad -\frac{d}{dt} \tilde{\mathbf{q}}(t) + \mathbf{A} \tilde{\mathbf{q}}(t) = \tilde{\mathbf{z}}(t), \quad t \in (0, T),$$

$$(4.10b) \quad \tilde{\mathbf{q}}(T) = \tilde{\mathbf{z}}(T).$$

Of course, (4.9), (4.10) decouple into the n independent systems

$$(4.11a) \quad \frac{d}{dt} \tilde{\mathbf{z}}_i(t) + \lambda_i \tilde{\mathbf{z}}_i(t) = 0, \quad t \in (0, T),$$

$$(4.11b) \quad \tilde{\mathbf{z}}_i(0) = \tilde{\mathbf{s}}_i$$

and

$$(4.12a) \quad -\frac{d}{dt} \tilde{\mathbf{q}}_i(t) + \lambda_i \tilde{\mathbf{q}}_i(t) = \tilde{\mathbf{z}}_i(t), \quad t \in (0, T),$$

$$(4.12b) \quad \tilde{\mathbf{q}}_i(T) = \tilde{\mathbf{z}}_i(T).$$

The solution of (4.11) is $\tilde{\mathbf{z}}_i(t) = \exp(-\lambda_i t) \tilde{\mathbf{s}}_i$ and the solution of (4.12) at the initial time is

$$(4.13) \quad \tilde{\mathbf{q}}_i(0) = e^{-\lambda_i T} \tilde{\mathbf{z}}_i(T) + \int_0^T e^{-\lambda_i \tau} \tilde{\mathbf{z}}_i(\tau) d\tau = \left(e^{-2\lambda_i T} + \int_0^T e^{-2\lambda_i \tau} d\tau \right) \tilde{\mathbf{s}}_i = h(\lambda_i) \tilde{\mathbf{s}}_i,$$

where the function $h : \mathbb{R} \rightarrow \mathbb{R}$ is defined by

$$(4.14) \quad h(\lambda) = \begin{cases} e^{-2\lambda T} + (1 - e^{-2\lambda T}) \frac{1}{2\lambda} & \text{if } \lambda \neq 0, \\ 1 + T & \text{if } \lambda = 0. \end{cases}$$

Since $\tilde{\mathbf{H}}_{ii} \tilde{\mathbf{s}}_i = \tilde{\mathbf{q}}_i(0) = h(\lambda_i) \tilde{\mathbf{s}}_i$ for all $\tilde{\mathbf{s}}_i$,

$$(4.15) \quad \tilde{\mathbf{H}}_{ii} = h(\lambda_i).$$

To estimate the error due to model reduction we will need the following result on the dependence of the diagonal entries (4.15) on λ_i .

LEMMA 1. *For any $T > 0$ the function $h : \mathbb{R} \rightarrow \mathbb{R}$ defined by (4.14) is positive, convex, monotonically decreasing, and converges to zero as $\lambda \rightarrow \infty$.*

Proof. The positivity of h on $[0, \infty)$ and $\lim_{\lambda \rightarrow \infty} h(\lambda) = 0$ follow immediately from the definition of h .

The first two derivatives of h are

$$h'(\lambda) = \begin{cases} \left(-2T + \frac{T}{\lambda} + \frac{1}{2\lambda^2} \right) e^{-2\lambda T} - \frac{1}{2\lambda^2} & \text{if } \lambda \neq 0, \\ -2T - T^2 & \text{if } \lambda = 0, \end{cases}$$

$$h''(\lambda) = \begin{cases} \left(4T^2 - \frac{2T^2}{\lambda} - \frac{2T}{\lambda^2} - \frac{1}{\lambda^3} \right) e^{-2\lambda T} + \frac{1}{\lambda^3} & \text{if } \lambda \neq 0, \\ 4T^2 + \frac{4}{3}T^3 & \text{if } \lambda = 0. \end{cases}$$

Using $e^x \geq 1 + x$ for all x gives

$$\begin{aligned} 2\lambda^2 e^{2\lambda T} h'(\lambda) &= -4T\lambda^2 + 2T\lambda + 1 - e^{2\lambda T} \leq -4T\lambda^2 + 2T\lambda + 1 - 1 - 2\lambda T \\ &= -4T\lambda^2 < 0 \quad \forall \lambda \neq 0, \end{aligned}$$

which implies that $h'(\lambda) < 0$ for all λ . Furthermore, since

$$\lambda^3 e^{2\lambda T} h''(\lambda) = 4T^2\lambda^3 - 2T^2\lambda^2 - 2T\lambda - 1 + e^{2\lambda T} = 4T^2\lambda^3 + \sum_{k=3}^{\infty} \frac{(2\lambda T)^k}{k!} > 0 \quad \forall \lambda > 0,$$

and, using $e^x \leq 1 + x + x^2/2$ for $x \leq 0$,

$$\lambda^3 e^{2\lambda T} h''(\lambda) = 4T^2\lambda^3 - 2T^2\lambda^2 - 2T\lambda - 1 + e^{2\lambda T} \leq 4T^2\lambda^3 < 0 \quad \forall \lambda < 0,$$

it holds that $h''(\lambda) > 0$ for all λ , which implies the convexity of h . $\square$

4.2. Hessian approximation for the decoupled problem.

4.2.1. Basic approximation. Recall that the generalized eigenvalues are ordered $\lambda_1 \leq \dots \leq \lambda_n$ (see (4.1b)). Since the function h defined in (4.14) is positive and monotonically decreasing (see Lemma 1), the best rank- k approximation of $\tilde{\mathbf{H}}$ is given by

$$(4.16) \quad \tilde{\mathbf{H}}^{\text{bsc}} = \text{diag}(\tilde{\mathbf{H}}_{11}, \dots, \tilde{\mathbf{H}}_{kk}, 0, \dots, 0).$$

The approximation $\tilde{\mathbf{H}}^{\text{bsc}} \tilde{\mathbf{s}}$ of $\tilde{\mathbf{H}} \tilde{\mathbf{s}}$, where $\tilde{\mathbf{H}}^{\text{bsc}}$ is given by (4.16), corresponds to projecting the dynamical systems (4.9), (4.10) by $\tilde{\mathbf{V}} = \tilde{\mathbf{W}} = [\mathbf{e}_1, \dots, \mathbf{e}_k] \in \mathbb{R}^{n \times k}$, where $\mathbf{e}_i$ denotes the i th unit vector. In the original $\mathbf{s}$ variables, this corresponds to the reduced order Hessian approximation (2.11), (2.12), (2.13), (2.14) with $\mathbf{V} = \mathbf{W} \in \mathbb{R}^{n \times k}$ given by the matrix corresponding to the first k columns of $\mathbf{V}_n$.

Unfortunately, $\tilde{\mathbf{H}}^{\text{bsc}}$ given by (4.16) is not always a good approximation. To see this, we compare the error between the solution $\tilde{\mathbf{s}}$ of the original system

$$(4.17) \quad (\tilde{\mathbf{H}} + \beta \mathbf{I}) \tilde{\mathbf{s}} = -\tilde{\mathbf{g}}$$

and the solution $\tilde{\mathbf{s}}^{\text{bsc}}$ of the reduced system

$$(4.18) \quad (\tilde{\mathbf{H}}^{\text{bsc}} + \beta \mathbf{I}) \tilde{\mathbf{s}}^{\text{bsc}} = -\tilde{\mathbf{g}}.$$

The componentwise absolute error and relative error are

$$(4.19) \quad |\tilde{\mathbf{s}}_i - \tilde{\mathbf{s}}_i^{\text{bsc}}| = \begin{cases} 0, & i = 1, \dots, k, \\ \tilde{\mathbf{H}}_{ii} / ((\tilde{\mathbf{H}}_{ii} + \beta)\beta) |\tilde{\mathbf{g}}_i|, & i = k+1, \dots, n, \end{cases}$$

and

$$|\tilde{\mathbf{s}}_i - \tilde{\mathbf{s}}_i^{\text{bsc}}| / |\tilde{\mathbf{s}}_i| = \tilde{\mathbf{H}}_{ii} / \beta \leq \tilde{\mathbf{H}}_{k+1, k+1} / \beta \quad \text{for } i \in \{k+1, \dots, n\} \text{ with } \tilde{\mathbf{g}}_i \neq 0,$$

respectively. These errors are only small if $0 \leq \tilde{\mathbf{H}}_{nn} \leq \dots \leq \tilde{\mathbf{H}}_{k+1, k+1}$ are small relative to β . While for some problems (in particular some inverse problems like the one in Bashir et al. [5]) there exists small k for which $\tilde{\mathbf{H}}_{k+1, k+1} < \beta$, this is not true for many others.

Similarly, if we compare the Hessian vector product for a general vector $\tilde{\mathbf{s}}$, then

$$(4.20) \quad \left[(\tilde{\mathbf{H}} + \beta \mathbf{I}) \tilde{\mathbf{s}} - (\tilde{\mathbf{H}}^{\text{bsc}} + \beta \mathbf{I}) \tilde{\mathbf{s}} \right]_i \bigg/ \left[(\tilde{\mathbf{H}} + \beta \mathbf{I}) \tilde{\mathbf{s}} \right]_i = \tilde{\mathbf{H}}_{ii} / (\tilde{\mathbf{H}}_{ii} + \beta) \\ \text{for } i \in \{k+1, \dots, n\} \text{ with } \tilde{\mathbf{s}}_i \neq 0.$$

Again, these errors are only small if $0 \leq \tilde{\mathbf{H}}_{nn} \leq \dots \leq \tilde{\mathbf{H}}_{k+1, k+1}$ are small relative to β .

The deficiencies of this basic ROM approach are illustrated in section 4.2.5 with a diagonal test example.

4.2.2. Augmenting the ROM. The best rank- k approximation (4.16) approximates the Hessian in the matrix-norm, but does not utilize that we want to solve a system with a specific right-hand side $\tilde{\mathbf{g}}$, or that we want to apply the Hessian to a specific vector $\tilde{\mathbf{s}}$. If $\tilde{\mathbf{g}} \in \text{range}(\tilde{\mathbf{V}})$, i.e., if all components $\tilde{\mathbf{g}}_{k+1} = \dots = \tilde{\mathbf{g}}_n = 0$, then $\tilde{\mathbf{s}}^{\text{bsc}} = \tilde{\mathbf{s}}$. Otherwise the solution error (4.19) can be large. Similarly, if $\tilde{\mathbf{s}} \in \text{range}(\tilde{\mathbf{V}})$,

i.e., if all components $\tilde{\mathbf{s}}_{k+1} = \dots = \tilde{\mathbf{s}}_n = 0$, then $\tilde{\mathbf{H}}^{\text{bsc}} \tilde{\mathbf{s}} = \tilde{\mathbf{H}} \tilde{\mathbf{s}}$. Otherwise the error (4.20) can be large.

To improve the ROM we add a vector $\tilde{\mathbf{a}}$, which will be $\tilde{\mathbf{g}}$ or $\tilde{\mathbf{s}}$, to the reduced basis $\tilde{\mathbf{V}} = \tilde{\mathbf{W}} = [\mathbf{e}_1, \dots, \mathbf{e}_k] \in \mathbb{R}^{n \times k}$ to generate $\tilde{\mathbf{V}}^{\text{aug}} = \tilde{\mathbf{W}}^{\text{aug}}$ with

$$\text{range}(\tilde{\mathbf{V}}^{\text{aug}}) = \text{range}([\tilde{\mathbf{V}}, \tilde{\mathbf{a}}]), \quad (\tilde{\mathbf{V}}^{\text{aug}})^T \tilde{\mathbf{V}}^{\text{aug}} = \mathbf{I}.$$

If $\tilde{\mathbf{a}} \in \tilde{\mathbf{V}}$, then $\tilde{\mathbf{V}}^{\text{aug}} = \tilde{\mathbf{V}} = [\mathbf{e}_1, \dots, \mathbf{e}_k]$, i.e., the ROM remains unchanged. If $\tilde{\mathbf{a}} \notin \tilde{\mathbf{V}}$, then

$$(4.21a) \quad \tilde{\mathbf{V}}^{\text{aug}} = [\mathbf{e}_1, \dots, \mathbf{e}_k, \tilde{\mathbf{v}}] \in \mathbb{R}^{n \times (k+1)},$$

where

$$(4.21b) \quad \tilde{\mathbf{v}} = \left(0, \dots, 0, \tilde{\mathbf{a}}_{k+1}, \dots, \tilde{\mathbf{a}}_n\right)^T \bigg/ \left(\sum_{i=k+1}^n \tilde{\mathbf{a}}_i^2\right)^{1/2}.$$

Now we project the systems (4.9), (4.10) with $\tilde{\mathbf{V}}^{\text{aug}}$. Assume $\sum_{i=k+1}^n \tilde{\mathbf{a}}_i^2 > 0$, i.e., $\tilde{\mathbf{a}} \notin \tilde{\mathbf{V}} = [\mathbf{e}_1, \dots, \mathbf{e}_k]$, and define

$$(4.22) \quad \lambda(\tilde{\mathbf{a}}) = \tilde{\mathbf{v}}^T \mathbf{\Lambda} \tilde{\mathbf{v}} = \sum_{i=k+1}^n \lambda_i \tilde{\mathbf{a}}_i^2 \bigg/ \sum_{i=k+1}^n \tilde{\mathbf{a}}_i^2.$$

Projecting the systems (4.9), (4.10) leads to the ODEs (4.11), (4.12), $i = 1, \dots, k$, and the following two additional ODEs,

$$(4.23a) \quad \frac{d}{dt} \tilde{\boldsymbol{\zeta}}(t) + \lambda(\tilde{\mathbf{a}}) \tilde{\boldsymbol{\zeta}}(t) = 0, \quad t \in (0, T),$$

$$(4.23b) \quad \tilde{\boldsymbol{\zeta}}(0) = \tilde{\mathbf{v}}^T \tilde{\mathbf{s}} = \frac{\sum_{i=k+1}^n \tilde{\mathbf{a}}_i \tilde{\mathbf{s}}_i}{\left(\sum_{i=k+1}^n \tilde{\mathbf{a}}_i^2\right)^{1/2}},$$

$$(4.24a) \quad -\frac{d}{dt} \tilde{\boldsymbol{\rho}}(t) + \lambda(\tilde{\mathbf{a}}) \tilde{\boldsymbol{\rho}}(t) = \tilde{\boldsymbol{\zeta}}(t), \quad t \in (0, T),$$

$$(4.24b) \quad \tilde{\boldsymbol{\rho}}(T) = \tilde{\boldsymbol{\zeta}}(T).$$

The solution of the coupled ODEs (4.11), (4.12) at $t = 0$ is $\tilde{\mathbf{q}}_i(0) = \tilde{\mathbf{H}}_{ii} \tilde{\mathbf{s}}_i$, $i = 1, \dots, k$, and the solution of the coupled system (4.23), (4.24) at $t = 0$ is

$$(4.25) \quad \tilde{\boldsymbol{\rho}}(0) = h(\lambda(\tilde{\mathbf{a}})) \frac{\sum_{i=k+1}^n \tilde{\mathbf{a}}_i \tilde{\mathbf{s}}_i}{\left(\sum_{i=k+1}^n \tilde{\mathbf{a}}_i^2\right)^{1/2}},$$

where h is the function defined in (4.14); cf. (4.13).

The Hessian approximation resulting from the subspace $\tilde{\mathbf{V}}^{\text{aug}} = \tilde{\mathbf{W}}^{\text{aug}}$ specified in (4.21) is given by

$$(4.26) \quad \tilde{\mathbf{H}}^{\text{aug}} \tilde{\mathbf{s}} = \sum_{i=1}^k \tilde{\mathbf{q}}_i(0) \mathbf{e}_i + \tilde{\boldsymbol{\rho}}(0) \tilde{\mathbf{v}} = \sum_{i=1}^k \tilde{\mathbf{H}}_{ii} \tilde{\mathbf{s}}_i \mathbf{e}_i + \tilde{\boldsymbol{\rho}}(0) \tilde{\mathbf{v}}.$$

To estimate the error due to model order reduction we will need the following properties of $\lambda(\tilde{\mathbf{a}})$ defined in (4.22). Since $\lambda(\tilde{\mathbf{a}})$ is a weighted arithmetic mean of $\lambda_{k+1}, \dots, \lambda_n$ it holds that $\lambda(\tilde{\mathbf{a}}) \in [\lambda_{k+1}, \lambda_n]$. Moreover, due to the convexity of h (cf. Lemma 1) Jensen's inequality,

$$(4.27) \quad \sum_{i=k+1}^n h(\lambda_i) \tilde{\mathbf{a}}_i^2 \bigg/ \sum_{j=k+1}^n \tilde{\mathbf{a}}_j^2 \geq h(\lambda(\tilde{\mathbf{a}})),$$

holds.

4.2.3. Augmenting the ROM by $\tilde{\mathbf{a}} = \tilde{\mathbf{g}}$. Under the assumptions (4.1) the system $\nabla^2 J \mathbf{s} = -\nabla J(\mathbf{0})$ is equivalent to (4.17). If $\tilde{\mathbf{g}} \notin \text{range}(\tilde{\mathbf{V}})$, then the error between the solution $\tilde{\mathbf{s}}$ of (4.17) and the solution $\tilde{\mathbf{s}}^{\text{bsc}}$ can be large; see (4.19). Therefore, in this subsection, we augment the initial ROM by

$$(4.28) \quad \tilde{\mathbf{a}} = \tilde{\mathbf{g}}.$$

If (4.28) and $\sum_{j=k+1}^n \tilde{\mathbf{g}}_j^2 > 0$, then the augmented ROM of section 4.2.3 gives

$$(4.29a) \quad \tilde{\mathbf{v}} = \left(0, \dots, 0, \tilde{\mathbf{g}}_{k+1}, \dots, \tilde{\mathbf{g}}_n\right)^T \bigg/ \left(\sum_{i=k+1}^n \tilde{\mathbf{g}}_i^2\right)^{1/2},$$

$$(4.29b) \quad \lambda(\tilde{\mathbf{g}}) = \sum_{i=k+1}^n \lambda_i \tilde{\mathbf{g}}_i^2 \bigg/ \sum_{i=k+1}^n \tilde{\mathbf{g}}_i^2,$$

$$(4.29c) \quad \tilde{\boldsymbol{\rho}}(0) = h(\lambda(\tilde{\mathbf{g}})) \frac{\sum_{i=k+1}^n \tilde{\mathbf{g}}_i \tilde{\mathbf{s}}_i}{\left(\sum_{i=k+1}^n \tilde{\mathbf{g}}_i^2\right)^{1/2}},$$

$$(4.29d) \quad \tilde{\mathbf{H}}^{\text{aug}} \tilde{\mathbf{s}} = \sum_{i=1}^k \tilde{\mathbf{H}}_{ii} \tilde{\mathbf{s}}_i \mathbf{e}_i + h(\lambda(\tilde{\mathbf{g}})) \frac{\sum_{i=k+1}^n \tilde{\mathbf{g}}_i \tilde{\mathbf{s}}_i}{\sum_{i=k+1}^n \tilde{\mathbf{g}}_i^2} \left(0, \dots, 0, \tilde{\mathbf{g}}_{k+1}, \dots, \tilde{\mathbf{g}}_n\right)^T.$$

The augmented ROM approximation to the system (4.17) is

$$(4.30) \quad (\tilde{\mathbf{H}}^{\text{aug}} + \beta \mathbf{I}) \tilde{\mathbf{s}}^{\text{aug}} = -\tilde{\mathbf{g}}.$$

LEMMA 2. Let $\sum_{j=k+1}^n \tilde{\mathbf{g}}_j^2 > 0$ and let $\tilde{\mathbf{H}}^{\text{aug}}$ be given by (4.29d). The solution of (4.30) is given by

$$(4.31) \quad \tilde{\mathbf{s}}_i^{\text{aug}} = \begin{cases} -\left(h(\lambda_i) + \beta\right)^{-1} \tilde{\mathbf{g}}_i, & i = 1, \dots, k, \\ -\left(h(\lambda(\tilde{\mathbf{g}})) + \beta\right)^{-1} \tilde{\mathbf{g}}_i, & i = k+1, \dots, n. \end{cases}$$

Proof. The representation (4.29d) of the Hessian vector product and $\tilde{\mathbf{H}}_{ii} = h(\lambda_i)$ imply (4.31) for $i = 1, \dots, k$ and

$$(4.32) \quad h(\lambda(\tilde{\mathbf{g}})) \frac{\sum_{i=k+1}^n \tilde{\mathbf{g}}_i \tilde{\mathbf{s}}_i}{\sum_{i=k+1}^n \tilde{\mathbf{g}}_i^2} \begin{pmatrix} \tilde{\mathbf{g}}_{k+1} \\ \vdots \\ \tilde{\mathbf{g}}_n \end{pmatrix} + \beta \begin{pmatrix} \tilde{\mathbf{s}}_{k+1}^{\text{aug}} \\ \vdots \\ \tilde{\mathbf{s}}_n^{\text{aug}} \end{pmatrix} = - \begin{pmatrix} \tilde{\mathbf{g}}_{k+1} \\ \vdots \\ \tilde{\mathbf{g}}_n \end{pmatrix}.$$

The identity (4.32) implies $(\tilde{\mathbf{s}}_{k+1}^{\text{aug}}, \dots, \tilde{\mathbf{s}}_n^{\text{aug}}) = \alpha(\tilde{\mathbf{g}}_{k+1}, \dots, \tilde{\mathbf{g}}_n)$ for some $\alpha \in \mathbb{R}$. Inserting this representation into (4.32) gives $\alpha = -1/(h(\lambda(\tilde{\mathbf{g}})) + \beta)$, which yields (4.31) for $i = k+1, \dots, n$. $\square$

The componentwise absolute error between the solution $\tilde{\mathbf{s}}$ of (4.17) and the solution $\tilde{\mathbf{s}}^{\text{aug}}$ of (4.30) is

$$(4.33) \quad |\tilde{\mathbf{s}}_i - \tilde{\mathbf{s}}_i^{\text{aug}}| = \begin{cases} 0, & i = 1, \dots, k, \\ \left| -\left(h(\lambda_i) + \beta\right)^{-1} + \left(h(\lambda(\tilde{\mathbf{g}})) + \beta\right)^{-1} \right| |\tilde{\mathbf{g}}_i|, & i = k+1, \dots, n. \end{cases}$$

The next theorem shows that augmenting the ROM does improve the quality of the approximate solution of (4.17).

THEOREM 3. *Let $\sum_{j=k+1}^n \tilde{\mathbf{g}}_j^2 > 0$. If $\tilde{\mathbf{s}}$ denotes the solution of (4.17), $\tilde{\mathbf{s}}^{\text{bsc}}$ is its approximation computed by solving the basic ROM system (4.18), and $\tilde{\mathbf{s}}^{\text{aug}}$ is its approximation computed by solving the augmented ROM system (4.30), then*

$$(4.34) \quad \|\tilde{\mathbf{s}} - \tilde{\mathbf{s}}^{\text{aug}}\|^2 \leq \|\tilde{\mathbf{s}} - \tilde{\mathbf{s}}^{\text{bsc}}\|^2 - \left(\frac{h(\lambda(\tilde{\mathbf{g}}))}{\beta(h(\lambda_{k+1}) + \beta)} \right)^2 \sum_{i=k+1}^n \tilde{\mathbf{g}}_i^2 < \|\tilde{\mathbf{s}} - \tilde{\mathbf{s}}^{\text{bsc}}\|^2.$$

Proof. Equations (4.19) and (4.33) imply

$$(4.35) \quad \begin{aligned} & \|\tilde{\mathbf{s}} - \tilde{\mathbf{s}}^{\text{bsc}}\|^2 - \|\tilde{\mathbf{s}} - \tilde{\mathbf{s}}^{\text{aug}}\|^2 \\ &= \sum_{i=k+1}^n \left[\frac{h(\lambda_i)^2}{\beta^2(h(\lambda_i) + \beta)^2} - \left(\frac{1}{h(\lambda_i) + \beta} - \frac{1}{h(\lambda(\tilde{\mathbf{g}})) + \beta} \right)^2 \right] \tilde{\mathbf{g}}_i^2. \end{aligned}$$

The term in square brackets is

$$(4.36) \quad \begin{aligned} & \frac{h(\lambda_i)^2}{\beta^2(h(\lambda_i) + \beta)^2} - \frac{[h(\lambda(\tilde{\mathbf{g}})) - h(\lambda_i)]^2}{(h(\lambda(\tilde{\mathbf{g}})) + \beta)^2(h(\lambda_i) + \beta)^2} \\ &= \left(\frac{1}{\beta(h(\lambda_i) + \beta)(h(\lambda(\tilde{\mathbf{g}})) + \beta)} \right)^2 \left\{ h(\lambda_i)^2(h(\lambda(\tilde{\mathbf{g}})) + \beta)^2 - [h(\lambda(\tilde{\mathbf{g}})) - h(\lambda_i)]^2 \beta^2 \right\}. \end{aligned}$$

The fraction in (4.36) can be bounded from below, using the facts that the λ_i 's are increasing and $h(\cdot)$ is decreasing due to Lemma 1,

$$(4.37) \quad \left(\frac{1}{\beta(h(\lambda_i) + \beta)(h(\lambda(\tilde{\mathbf{g}})) + \beta)} \right)^2 \geq \left(\frac{1}{\beta(h(\lambda_{k+1}) + \beta)(h(\lambda(\tilde{\mathbf{g}})) + \beta)} \right)^2, \quad i \geq k+1.$$

The term in braces in (4.36) is

$$\begin{aligned}
 & h(\lambda_i)^2 [h(\lambda(\tilde{\mathbf{g}}))^2 + 2h(\lambda(\tilde{\mathbf{g}}))\beta + \beta^2] - [h(\lambda(\tilde{\mathbf{g}}))^2 - 2h(\lambda(\tilde{\mathbf{g}}))h(\lambda_i) + h(\lambda_i)^2]\beta^2 \\
 & = h(\lambda_i)^2 [h(\lambda(\tilde{\mathbf{g}}))^2 + 2h(\lambda(\tilde{\mathbf{g}}))\beta] - [h(\lambda(\tilde{\mathbf{g}}))^2 - 2h(\lambda(\tilde{\mathbf{g}}))h(\lambda_i)]\beta^2 \\
 (4.38) \quad & = h(\lambda(\tilde{\mathbf{g}})) \left(h(\lambda_i)^2 [h(\lambda(\tilde{\mathbf{g}})) + 2\beta] - [h(\lambda(\tilde{\mathbf{g}})) - 2h(\lambda_i)]\beta^2 \right).
 \end{aligned}$$

Inserting (4.36)–(4.38) into (4.35) gives

$$\begin{aligned}
 & \|\tilde{\mathbf{s}} - \tilde{\mathbf{s}}^{\text{bsc}}\|^2 - \|\tilde{\mathbf{s}} - \tilde{\mathbf{s}}^{\text{aug}}\|^2 \\
 & \geq \left(\frac{1}{\beta (h(\lambda_{k+1}) + \beta) (h(\lambda(\tilde{\mathbf{g}})) + \beta)} \right)^2 \\
 (4.39) \quad & \times h(\lambda(\tilde{\mathbf{g}})) \sum_{i=k+1}^n \left(h(\lambda_i)^2 [h(\lambda(\tilde{\mathbf{g}})) + 2\beta] - [h(\lambda(\tilde{\mathbf{g}})) - 2h(\lambda_i)]\beta^2 \right) \tilde{\mathbf{g}}_i^2.
 \end{aligned}$$

Using Jensen's inequality (4.27) with $\tilde{\mathbf{a}} = \tilde{\mathbf{g}}$ we bound the sum in (4.39) as follows:

$$\begin{aligned}
 & \sum_{i=k+1}^n \left(h(\lambda_i)^2 [h(\lambda(\tilde{\mathbf{g}})) + 2\beta] - [h(\lambda(\tilde{\mathbf{g}})) - 2h(\lambda_i)]\beta^2 \right) \tilde{\mathbf{g}}_i^2 \\
 & = [h(\lambda(\tilde{\mathbf{g}})) + 2\beta] \sum_{i=k+1}^n h(\lambda_i)^2 \tilde{\mathbf{g}}_i^2 - h(\lambda(\tilde{\mathbf{g}}))\beta^2 \sum_{i=k+1}^n \tilde{\mathbf{g}}_i^2 + 2\beta^2 \sum_{i=k+1}^n h(\lambda_i) \tilde{\mathbf{g}}_i^2 \\
 (4.40) \quad & \geq [h(\lambda(\tilde{\mathbf{g}})) + 2\beta] \sum_{i=k+1}^n h(\lambda_i)^2 \tilde{\mathbf{g}}_i^2 + h(\lambda(\tilde{\mathbf{g}})) \beta^2 \sum_{i=k+1}^n \tilde{\mathbf{g}}_i^2.
 \end{aligned}$$

Since h is convex and positive, h^2 is convex. Hence we can apply Jensen's inequality (4.27) with $\tilde{\mathbf{a}} = \tilde{\mathbf{g}}$ and h replaced by h^2 to bound

$$\begin{aligned}
 & [h(\lambda(\tilde{\mathbf{g}})) + 2\beta] \sum_{i=k+1}^n h(\lambda_i)^2 \tilde{\mathbf{g}}_i^2 + h(\lambda(\tilde{\mathbf{g}})) \beta^2 \sum_{i=k+1}^n \tilde{\mathbf{g}}_i^2 \\
 & \geq \left([h(\lambda(\tilde{\mathbf{g}})) + 2\beta] h(\lambda(\tilde{\mathbf{g}}))^2 + \beta^2 h(\lambda(\tilde{\mathbf{g}})) \right) \sum_{i=k+1}^n \tilde{\mathbf{g}}_i^2 \\
 (4.41) \quad & = h(\lambda(\tilde{\mathbf{g}})) \left(h(\lambda(\tilde{\mathbf{g}})) + \beta \right)^2 \sum_{i=k+1}^n \tilde{\mathbf{g}}_i^2.
 \end{aligned}$$

Inserting (4.40), (4.41) into (4.39) gives the desired bound

$$\|\tilde{\mathbf{s}} - \tilde{\mathbf{s}}^{\text{bsc}}\|^2 - \|\tilde{\mathbf{s}} - \tilde{\mathbf{s}}^{\text{aug}}\|^2 \geq \left(\frac{h(\lambda(\tilde{\mathbf{g}}))}{\beta (h(\lambda_{k+1}) + \beta)} \right)^2 \sum_{i=k+1}^n \tilde{\mathbf{g}}_i^2 > 0. \quad \square$$

4.2.4. Augmenting the ROM by $\tilde{\mathbf{a}} = \tilde{\mathbf{s}}$. Under the assumptions (4.1) the system $\nabla^2 J \mathbf{s} = -\nabla J(\mathbf{0}) = -\mathbf{g}$ simplifies to (4.2) and (4.17). In this case it was easy to see that the potentially large error between the solution $\tilde{\mathbf{s}}$ of (4.17) and its approximation $\tilde{\mathbf{s}}^{\text{bsc}}$ computed by solving the initial ROM system (4.18) can be reduced by adding $\mathbf{g}$ to the ROM. However, for more general problems this is no longer clear. In fact, when $\mathbf{R} \neq \beta \mathbf{M}$, then $\nabla^2 J \mathbf{s} = -\nabla J(\mathbf{0}) = -\mathbf{g}$ is given by $(\mathbf{H} + \mathbf{M}^{-1}\mathbf{R})\mathbf{s} =$

–**g.** In the general case we apply the CG method and are interested in ROMs that approximate $\mathbf{H} \mathbf{s}$ well or, equivalently, that approximate $\tilde{\mathbf{H}} \tilde{\mathbf{s}}$ well.

If $\tilde{\mathbf{s}} \notin \text{range}(\tilde{\mathbf{V}})$, then the error between $\tilde{\mathbf{H}} \tilde{\mathbf{s}}$ and $\tilde{\mathbf{H}}^{\text{bsc}} \tilde{\mathbf{s}}$ can be large; see (4.20). This leads us to augment the ROM by

$$(4.42) \quad \tilde{\mathbf{a}} = \tilde{\mathbf{s}}.$$

In this subsection we set (4.42) and estimate how well the ROM Hessians $\tilde{\mathbf{H}}^{\text{bsc}}$ and $\tilde{\mathbf{H}}^{\text{aug}}$ approximate the Hessian-times-vector product $\tilde{\mathbf{H}} \tilde{\mathbf{s}}$ for a given $\tilde{\mathbf{s}}$. We still assume that (4.1) holds.

If (4.42) and $\sum_{j=k+1}^n \tilde{\mathbf{s}}_j^2 > 0$, then

$$(4.43a) \quad \tilde{\mathbf{v}} = \left(0, \dots, 0, \tilde{\mathbf{s}}_{k+1}, \dots, \tilde{\mathbf{s}}_n\right)^T \bigg/ \left(\sum_{i=k+1}^n \tilde{\mathbf{s}}_i^2\right)^{1/2},$$

$$(4.43b) \quad \lambda(\tilde{\mathbf{s}}) = \sum_{i=k+1}^n \lambda_i \tilde{\mathbf{s}}_i^2 \bigg/ \sum_{i=k+1}^n \tilde{\mathbf{s}}_i^2,$$

$$(4.43c) \quad \tilde{\rho}(0) = h(\lambda(\tilde{\mathbf{s}})) \left(\sum_{i=k+1}^n \tilde{\mathbf{s}}_i^2\right)^{1/2},$$

$$(4.43d) \quad \begin{aligned} \tilde{\mathbf{H}}^{\text{aug}} \tilde{\mathbf{s}} &= \sum_{i=1}^k \tilde{\mathbf{H}}_{ii} \tilde{\mathbf{s}}_i \mathbf{e}_i + h(\lambda(\tilde{\mathbf{s}})) \left(0, \dots, 0, \tilde{\mathbf{s}}_{k+1}, \dots, \tilde{\mathbf{s}}_n\right)^T \\ &= \tilde{\mathbf{H}}^{\text{bsc}} \tilde{\mathbf{s}} + h(\lambda(\tilde{\mathbf{s}})) \left(0, \dots, 0, \tilde{\mathbf{s}}_{k+1}, \dots, \tilde{\mathbf{s}}_n\right)^T. \end{aligned}$$

Remark 4. In the case (4.42) both $\tilde{\rho}(0)$ and $\tilde{\mathbf{v}}$ depend nonlinearly on $\tilde{\mathbf{s}}$ and, therefore, $\tilde{\mathbf{H}}^{\text{aug}}$ depends nonlinearly on the vector $\tilde{\mathbf{s}}$ it is applied to. We will deal with the impact on the CG method in section 6.

Recall that if $\sum_{j=k+1}^n \tilde{\mathbf{s}}_j^2 = 0$, then $\tilde{\mathbf{H}}^{\text{bsc}} \tilde{\mathbf{s}} = \tilde{\mathbf{H}} \tilde{\mathbf{s}}$ (see (4.16)), and the augmentation of the ROM is not needed. If $\sum_{j=k+1}^n \tilde{\mathbf{s}}_j^2 > 0$, then

$$(4.44) \quad \left[\tilde{\mathbf{H}} \tilde{\mathbf{s}} - \tilde{\mathbf{H}}^{\text{bsc}} \tilde{\mathbf{s}}\right]_i = \begin{cases} 0, & i = 1, \dots, k, \\ h(\lambda_i) \tilde{\mathbf{s}}_i, & i = k+1, \dots, n \end{cases}$$

(see (4.16)), and augmenting the ROM leads to the error

$$(4.45) \quad \left[\tilde{\mathbf{H}} \tilde{\mathbf{s}} - \tilde{\mathbf{H}}^{\text{aug}} \tilde{\mathbf{s}}\right]_i = \begin{cases} 0, & i = 1, \dots, k, \\ \left[h(\lambda_i) - h(\lambda(\tilde{\mathbf{s}}))\right] \tilde{\mathbf{s}}_i, & i = k+1, \dots, n; \end{cases}$$

see the definition (4.15) of the Hessian $\tilde{\mathbf{H}}$ and the definition (4.43d) of the augmented ROM Hessian $\tilde{\mathbf{H}}^{\text{aug}}$.

The next theorem shows that if $\sum_{j=k+1}^n \tilde{\mathbf{s}}_j^2 > 0$ the augmented ROM always leads to a better approximation of the Hessian vector product. (Recall, that no augmentation is needed in the case $\sum_{j=k+1}^n \tilde{\mathbf{s}}_j^2 = 0$.)

THEOREM 5. Let the function h be given by (4.14). If $\sum_{j=k+1}^n \tilde{\mathbf{s}}_j^2 > 0$, the Hessian approximations satisfy

(4.46)

$$\|\tilde{\mathbf{H}} \tilde{\mathbf{s}} - \tilde{\mathbf{H}}^{\text{aug}} \tilde{\mathbf{s}}\|_2^2 \leq \|\tilde{\mathbf{H}} \tilde{\mathbf{s}} - \tilde{\mathbf{H}}^{\text{bsc}} \tilde{\mathbf{s}}\|_2^2 - h(\lambda(\tilde{\mathbf{s}})) \sum_{i=k+1}^n h(\lambda_i) \tilde{\mathbf{s}}_i^2 < \|\tilde{\mathbf{H}} \tilde{\mathbf{s}} - \tilde{\mathbf{H}}^{\text{bsc}} \tilde{\mathbf{s}}\|_2^2.$$

Proof. Equations (4.45) and (4.44) imply

$$\begin{aligned} \|\tilde{\mathbf{H}} \tilde{\mathbf{s}} - \tilde{\mathbf{H}}^{\text{bsc}} \tilde{\mathbf{s}}\|^2 - \|\tilde{\mathbf{H}} \tilde{\mathbf{s}} - \tilde{\mathbf{H}}^{\text{aug}} \tilde{\mathbf{s}}\|^2 &= \sum_{i=k+1}^n \left[h(\lambda_i)^2 - [h(\lambda_i) - h(\lambda(\tilde{\mathbf{s}}))]^2 \right] \tilde{\mathbf{s}}_i^2 \\ &= h(\lambda(\tilde{\mathbf{s}})) \left[2 \sum_{i=k+1}^n h(\lambda_i) \tilde{\mathbf{s}}_i^2 - h(\lambda(\tilde{\mathbf{s}})) \sum_{i=k+1}^n \tilde{\mathbf{s}}_i^2 \right] \\ &= h(\lambda(\tilde{\mathbf{s}})) \left(\sum_{j=k+1}^n \tilde{\mathbf{s}}_j^2 \right) \left[2 \sum_{i=k+1}^n h(\lambda_i) \tilde{\mathbf{s}}_i^2 / \sum_{j=k+1}^n \tilde{\mathbf{s}}_j^2 - h(\lambda(\tilde{\mathbf{s}})) \right] \\ (4.47) \quad &\geq h(\lambda(\tilde{\mathbf{s}})) \left(\sum_{j=k+1}^n \tilde{\mathbf{s}}_j^2 \right) \left[\sum_{i=k+1}^n h(\lambda_i) \tilde{\mathbf{s}}_i^2 / \sum_{j=k+1}^n \tilde{\mathbf{s}}_j^2 \right], \end{aligned}$$

where in the last step we have used Jensen's inequality (4.27) with $\tilde{\mathbf{a}} = \tilde{\mathbf{s}}$. Inequality (4.47) is identical to the first inequality in (4.46). Using Jensen's inequality (4.27) with $\tilde{\mathbf{a}} = \tilde{\mathbf{s}}$ again,

$$h(\lambda(\tilde{\mathbf{s}})) \sum_{i=k+1}^n h(\lambda_i) \tilde{\mathbf{s}}_i^2 \geq h(\lambda(\tilde{\mathbf{s}}))^2 \sum_{i=k+1}^n \tilde{\mathbf{s}}_i^2 > 0,$$

which gives the second inequality in (4.46). $\square$

4.2.5. Example. To illustrate the benefits of augmenting the basic ROM we consider the following decoupled example. The eigenvalues $\lambda_1, \dots, \lambda_n$ come from a two-dimensional in space version of model problem (3.1a)–(3.1c) with homogeneous Dirichlet boundary conditions and parameters $\kappa = 0.1$, $a = (0, 0)^T$, $\gamma = 0.5$, which is discretized using a 30×30 mesh on $\Omega = (0, 1)^2$. This leads to a problem (4.3) of size $n = 841$. We set $\beta = 5 \cdot 10^{-4}$. The vectors $\tilde{\mathbf{g}}$ and $\tilde{\mathbf{s}}$ are constructed the same way; we therefore only specify $\tilde{\mathbf{g}}$.

Except for the components specified below, all entries of $\tilde{\mathbf{g}}$ are zero.

$$(4.48) \quad \text{Case 1: } \tilde{g}_{14} = \tilde{g}_{18} = \tilde{g}_{22} = 0.7, \quad \tilde{g}_{34} = 1.2, \\ \tilde{g}_{90} = \dots = \tilde{g}_{160} = 1, \quad \tilde{g}_n = \tilde{g}_{n-1} = 10^{-5},$$

$$(4.49) \quad \text{Case 2: } \tilde{g}_{14} = \tilde{g}_{18} = \tilde{g}_{22} = 0.7, \quad \tilde{g}_{34} = 1.2, \quad \tilde{g}_{97} = 1, \quad \tilde{g}_n = \tilde{g}_{n-1} = 10^{-5}.$$

We have chosen nonzero entries in $\tilde{g}_{n-1}$ and $\tilde{g}_n$ such that the assumptions $\sum_{j=k+1}^n \tilde{\mathbf{g}}_j^2 > 0$ of Theorem 3 and $\sum_{j=k+1}^n \tilde{\mathbf{s}}_j^2 > 0$ of Theorem 5 are met for all k .

Augmenting by $\tilde{\mathbf{a}} = \tilde{\mathbf{g}}$. We compute approximations to the solution of $\nabla^2 \tilde{J} \tilde{\mathbf{s}} = -\tilde{\mathbf{g}}$ using the basic ROM and the basic ROM augmented by $\tilde{\mathbf{a}} = \tilde{\mathbf{g}}$. The results are depicted in Figure 1.

FIG. 1. Results of the augmented ROM approach with $\tilde{\mathbf{a}} = \tilde{\mathbf{g}}$ applied to the diagonal test example for two different right-hand side vectors $\tilde{\mathbf{g}}$. Top row: Relative errors $\|\tilde{\mathbf{s}} - \tilde{\mathbf{s}}^{\text{bsc}}\|^2 / \|\tilde{\mathbf{s}}\|^2$ (red) and $\|\tilde{\mathbf{s}} - \tilde{\mathbf{s}}^{\text{aug}}\|^2 / \|\tilde{\mathbf{s}}\|^2$ (blue) as well as upper bound (green) on $\|\tilde{\mathbf{s}} - \tilde{\mathbf{s}}^{\text{aug}}\|^2 / \|\tilde{\mathbf{s}}\|^2$ according to Theorem 3 where $\tilde{\mathbf{s}}$ solves (4.17), $\tilde{\mathbf{s}}^{\text{bsc}}$ solves (4.18), and $\tilde{\mathbf{s}}^{\text{aug}}$ solves (4.30). Bottom row: Values of $\tilde{g}_i^2$.

The plots in the top row in Figure 1 show the relative errors between the exact solution of $\nabla^2 \tilde{J} \tilde{\mathbf{s}} = -\tilde{\mathbf{g}}$ and solutions computed using the basic ROM (red curves) and using the augmented ROM (blue curves) for ROM sizes $k \leq 150$. The green curves are the upper bound given by the expression in (4.34) for the error in the solution generated by the augmented ROM. The plots in the bottom row in Figure 1 show the first 150 components of $\tilde{\mathbf{g}}$.

In both cases the error between the exact solution and the solution computed with the basic ROM are large until the basic ROM is large enough to capture all nonzero components of $\tilde{\mathbf{g}}$. In practice such a ROM may be too large to be truly useful.

The augmented ROM performs substantially better. Adding the vector $\tilde{\mathbf{a}} = \tilde{\mathbf{g}}$ to the ROM basis substantially reduces the error between the exact solution and the solution computed with the augmented ROM. The benefits are greater if $\tilde{\mathbf{g}}$ has just a few nonzero entries with index greater than k ; see Case 1 versus Case 2.

In these cases the upper bound given by the expression in (4.34) (green curves) is very pessimistic compared to the true error (blue curves). Since (4.34) bounds the error in the solutions of systems with matrices $\tilde{\mathbf{H}} + \beta \mathbf{I}$ and $\tilde{\mathbf{H}}^{\text{aug}} + \beta \mathbf{I}$ and the same right-hand side $\tilde{\mathbf{g}}$, this is not too surprising. The general error bound (see, e.g., [14, Thm. 2.7.2]) states that the relative error in the solution is bounded by the condition number of $\tilde{\mathbf{H}} + \beta \mathbf{I}$ times the relative error in the matrices. The condition number of $\tilde{\mathbf{H}} + \beta \mathbf{I}$ is $(h(\lambda_1) + \beta) / (h(\lambda_n) + \beta) \approx \beta^{-1} (h(\lambda_1) + \beta) \approx 10^4$ in this example. The estimate (4.34) is of the same type. While reasonably sharp in some cases, it will be pessimistic in many others.

FIG. 2. Results of the augmented ROM approach with $\mathbf{a} = \tilde{\mathbf{s}}$ applied to the diagonal test example for two different vectors $\tilde{\mathbf{s}}$. First row: Relative errors $\|\tilde{\mathbf{H}}\tilde{\mathbf{s}} - \tilde{\mathbf{H}}^{\text{bsc}}\tilde{\mathbf{s}}\|^2 / \|\tilde{\mathbf{H}}\tilde{\mathbf{s}} + \beta\tilde{\mathbf{s}}\|^2$ (red) and $\|\tilde{\mathbf{H}}\tilde{\mathbf{s}} - \tilde{\mathbf{H}}^{\text{aug}}\tilde{\mathbf{s}}\|^2 / \|\tilde{\mathbf{H}}\tilde{\mathbf{s}} + \beta\tilde{\mathbf{s}}\|^2$ (blue) as well as the upper bound (green) on $\|\tilde{\mathbf{H}}\tilde{\mathbf{s}} - \tilde{\mathbf{H}}^{\text{aug}}\tilde{\mathbf{s}}\|^2 / \|\tilde{\mathbf{H}}\tilde{\mathbf{s}} + \beta\tilde{\mathbf{s}}\|^2$ according to Theorem 5. Second row: Values of s_i^2 .

Augmenting by $\tilde{\mathbf{a}} = \tilde{\mathbf{s}}$. We now analyze the ROM augmented by $\tilde{\mathbf{a}} = \tilde{\mathbf{s}}$ and compare the quality of the Hessian vector products. We consider two cases. In Case 1 we use the vector $\tilde{\mathbf{s}} = \tilde{\mathbf{g}}$ given in (4.48) and in Case 2 we use $\tilde{\mathbf{s}} = \tilde{\mathbf{g}}$ given in (4.49). The results are depicted in Figure 2.

The plots in the top row in Figure 2 show the relative errors between the exact Hessian vector product $(\tilde{\mathbf{H}} + \beta)\tilde{\mathbf{s}}$ and the basic ROM Hessian vector product $(\tilde{\mathbf{H}}^{\text{bsc}} + \beta)\tilde{\mathbf{s}}$ (red curves) and the augmented ROM Hessian-vector product $(\tilde{\mathbf{H}}^{\text{aug}} + \beta)\tilde{\mathbf{s}}$ (blue curves). The green curves are the upper bound given by the expression in (4.46) for the error in the augmented ROM Hessian vector product. The plots at the bottom of Figure 2 show the first 150 components of $\tilde{\mathbf{s}}$.

In both cases the error between the exact Hessian vector product and the basic ROM Hessian vector product are large until the basic ROM is large enough to capture most nonzero components of $\tilde{\mathbf{s}}$. The error between the exact Hessian vector product and the augmented ROM Hessian vector product is substantially smaller, and the augmentation of the basic ROM bases by just one vector can dramatically improve the accuracy of the Hessian vector product. This improvement is the larger the more concentrated the nonzero components of $\tilde{\mathbf{s}}$ are. In particular, in Case 2 the augmented ROM Hessian vector product is already accurate for $k \geq 34$ since the contribution of $\tilde{s}_{97}$ is included in the basis $\tilde{\mathbf{V}}$, whereas in the basic ROM this contribution is included only if $k \geq 97$.

The upper bound given by the expression in (4.46) for the error in the augmented ROM Hessian vector product is fairly sharp in both cases.

4.3. ROM augmentation in the original state space. In practice we apply the previous ROM augmentation to the original problem (1.1) instead of the transformed one (4.3). We are given $\mathbf{V} = \mathbf{W} \in \mathbb{R}^{n \times k}$ with $\mathbf{V}^T \mathbf{M} \mathbf{V} = \mathbf{I}$. This ROM is improved by adding the vector $\mathbf{a}$, where $\mathbf{a} = \mathbf{g}$ or $\mathbf{a} = \mathbf{s}$ to the reduced basis $\mathbf{V}$ to generate $\mathbf{V}^{\text{aug}} = \mathbf{W}^{\text{aug}}$ with

$$\text{range}(\mathbf{V}^{\text{aug}}) = \text{range}([\mathbf{V}, \mathbf{a}], \quad (\mathbf{V}^{\text{aug}})^T \mathbf{M} \mathbf{V}^{\text{aug}} = \mathbf{I}.$$

If $\mathbf{a} \in \mathbf{V}$, then $\mathbf{V}^{\text{aug}} = \mathbf{V}$, i.e., the ROM remains unchanged. If $\mathbf{a} \notin \mathbf{V}$, then

$$\mathbf{V}^{\text{aug}} = [\mathbf{V}, \mathbf{v}], \quad \text{where} \quad \mathbf{v} = (\mathbf{a} - \mathbf{V} \mathbf{V}^T \mathbf{M} \mathbf{a}) / \|\mathbf{a} - \mathbf{V} \mathbf{V}^T \mathbf{M} \mathbf{a}\|_{\mathbf{M}}.$$

5. Augmented ROM Hessians. The development of the ROM introduced in section 4 for the special case can be generalized. The basic ROM is now computed from the Jordan normal form. This basic ROM is then augmented by adding $\mathbf{a} = \mathbf{g}$ or $\mathbf{a} = \mathbf{s}$ to the ROM bases. We present our approach in this section and numerically demonstrate its benefit in section 7.2. The analysis of our approach in the general case is subject to future research.

5.1. Basic ROM. We no longer assume that $\mathbf{A}$ is symmetric nor diagonalizable. Let $\lambda_1, \dots, \lambda_n \in \mathbb{C}$ be the generalized eigenvalues of $(\mathbf{A}, \mathbf{M})$ (i.e., eigenvalues of $\mathbf{M}^{-1} \mathbf{A}$). We assume that they are sorted in such that

$$(5.1) \quad \text{Re}(\lambda_1) \leq \dots \leq \text{Re}(\lambda_n).$$

By the Jordan normal form, there exists an invertible matrix $\mathbf{X} \in \mathbb{R}^{n \times n}$ such that

$$(5.2) \quad \mathbf{X}^{-1} \mathbf{M}^{-1} \mathbf{A} \mathbf{X} = \mathbf{J} = \text{diag}(\mathbf{J}_1, \dots, \mathbf{J}_q),$$

where for real eigenvalues λ_j the Jordan block is given by

$$\mathbf{J}_j = \begin{pmatrix} \lambda_j & 1 & & \\ & \ddots & \ddots & \\ & & \ddots & 1 \\ & & & \lambda_j \end{pmatrix},$$

and for complex conjugate pairs of eigenvalues $\lambda_j = \alpha_j \pm i\beta_j$ the Jordan block is given by

$$\mathbf{J}_j = \begin{pmatrix} \mathbf{\Lambda}_j & \mathbf{I} & & \\ & \ddots & \ddots & \\ & & \ddots & \mathbf{I} \\ & & & \mathbf{\Lambda}_j \end{pmatrix} \quad \text{with} \quad \mathbf{\Lambda}_j = \begin{bmatrix} \alpha_j & \beta_j \\ -\beta_j & \alpha_j \end{bmatrix};$$

see, e.g., [26]. Instead of using the Jordan normal form (5.2) one can also use the real Schur form [23, Thm. 2.3.4]. The matrices $\mathbf{V}_n = \mathbf{X}$ and $\mathbf{W}_n = (\mathbf{X}^{-1} \mathbf{M}^{-1})^T$ satisfy

$$(5.3) \quad \mathbf{W}_n^T \mathbf{A} \mathbf{V}_n = \mathbf{J},$$

$$(5.4) \quad \mathbf{W}_n^T \mathbf{M} \mathbf{V}_n = \mathbf{X}^{-1} \mathbf{M}^{-1} \mathbf{M} \mathbf{X} = \mathbf{I}.$$

Our basic ROM $\mathbf{V}, \mathbf{W} \in \mathbb{R}^{n \times k}$ is obtained by taking the first k columns of $\mathbf{V}_n$ and $\mathbf{W}_n$, respectively. This ROM includes the k most important eigenvalues, i.e., those

resulting in slowly decaying solution components; cf. section 4.2.1. Nevertheless, in the same way as in the symmetric case, this basic ROM approach exhibits its weakness.

We note that in our numerical examples in section 7.2 generated from the model problem (3.1) with nonzero advection a , the matrices $\mathbf{M}^{-1}\mathbf{A}$ are diagonalizable but have some complex eigenvalues, i.e., the Jordan blocks are of size $\mathbf{J}_j \in \mathbb{R}$ or $\mathbf{J}_j \in \mathbb{R}^{2 \times 2}$.

5.2. Augmenting the ROM. To improve the basic ROM of section 5.1 we add again a vector $\mathbf{a}$, which will be either $\mathbf{g}$ or $\mathbf{s}$, to the bases $\mathbf{V}, \mathbf{W}$ to generate $\mathbf{V}^{\text{aug}}, \mathbf{W}^{\text{aug}}$ with

$$(5.5a) \quad \text{Range}(\mathbf{V}^{\text{aug}}) = \text{Range}([\mathbf{V}, \mathbf{a}]),$$

$$(5.5b) \quad \text{Range}(\mathbf{W}^{\text{aug}}) = \text{Range}([\mathbf{W}, \mathbf{a}]),$$

$$(5.5c) \quad (\mathbf{W}^{\text{aug}})^T \mathbf{M} \mathbf{V}^{\text{aug}} = \mathbf{I}.$$

Adding $\mathbf{a} = \mathbf{s}$ in (5.5a) guarantees that $\mathbf{s} = \mathbf{V}^{\text{aug}} \hat{\mathbf{s}}$ for some $\hat{\mathbf{s}}$ and that $\hat{\mathbf{s}} = (\mathbf{W}^{\text{aug}})^T \mathbf{M} \mathbf{s}$. Consequently, the initial data $\mathbf{s}$ in (2.11) fully enter the ROM Hessian. In the case $\mathbf{G} = \mathbf{M}$ the ROM Hessian component (2.13) computed with $\mathbf{V}^{\text{aug}}$ and $\mathbf{W}^{\text{aug}}$ satisfies $R(\mathbf{H}^{\text{aug}}) \subset R(\mathbf{W}^{\text{aug}})$, which motivates adding $\mathbf{a} = \mathbf{g}$ in (5.5b). We will see later that for some problems, adding $\mathbf{a} = \mathbf{g}$ to $\mathbf{W}$ in (5.5b) guarantees that the CG directions stay inside the range of $\mathbf{W}^{\text{aug}}$; see Lemma 8.

The addition of $\mathbf{a}$ to $\mathbf{V}, \mathbf{W} \in \mathbb{R}^{n \times k}$ with $\mathbf{V}^T \mathbf{M} \mathbf{W} = \mathbf{I}$ can be done as follows (see, e.g., [24], [30]). The vectors

$$\tilde{\mathbf{v}}_{k+1} = \mathbf{a} - \mathbf{V} \mathbf{W}^T \mathbf{M} \mathbf{a},$$

$$\tilde{\mathbf{w}}_{k+1} = \mathbf{a} - \mathbf{W} \mathbf{V}^T \mathbf{M} \mathbf{a},$$

satisfy $\mathbf{W}^T \mathbf{M} \tilde{\mathbf{v}}_{k+1} = \mathbf{0}$, $\mathbf{V}^T \mathbf{M} \tilde{\mathbf{w}}_{k+1} = \mathbf{0}$, and $\mathbf{a}^T \mathbf{M} \tilde{\mathbf{w}}_{k+1} = \mathbf{a}^T \mathbf{M} \tilde{\mathbf{v}}_{k+1} = \tilde{\mathbf{w}}_{k+1}^T \mathbf{M} \tilde{\mathbf{v}}_{k+1}$. Hence, if $\mathbf{a}^T \mathbf{M} \tilde{\mathbf{w}}_{k+1} = \mathbf{a}^T \mathbf{M} \tilde{\mathbf{v}}_{k+1} = 0$, then $\mathbf{V}^{\text{aug}} = \mathbf{V}$, $\mathbf{W}^{\text{aug}} = \mathbf{W}$ in (5.5). Otherwise $\mathbf{V}^{\text{aug}} = [\mathbf{V}, \mathbf{v}_{k+1}]$, $\mathbf{W}^{\text{aug}} = [\mathbf{W}, \mathbf{w}_{k+1}]$, where $\mathbf{v}_{k+1} = \tilde{\mathbf{v}}_{k+1} / \alpha_{k+1}$ and $\mathbf{w}_{k+1} = \tilde{\mathbf{w}}_{k+1} / \beta_{k+1}$ and $\alpha_{k+1}, \beta_{k+1}$ are nonzero scalars with $\alpha_{k+1} \beta_{k+1} = \tilde{\mathbf{w}}_{k+1}^T \mathbf{M} \tilde{\mathbf{v}}_{k+1}$, e.g., $\alpha_{k+1} = |\tilde{\mathbf{w}}_{k+1}^T \mathbf{M} \tilde{\mathbf{v}}_{k+1}|^{1/2}$, $\beta_{k+1} = \tilde{\mathbf{w}}_{k+1}^T \mathbf{M} \tilde{\mathbf{v}}_{k+1} / \alpha_{k+1}$.

6. CG method.

6.1. CG method with full Hessian. The Hessian $\mathbf{H} + \mathbf{G}^{-1} \mathbf{R}$ is self-adjoint with respect to the inner product (2.1) and positive definite. Therefore we can solve system (2.7) using the CG method with inner product (2.1). We set $\mathbf{g} = \nabla J(\mathbf{0})$ so that the system (2.7) reads

$$(6.1) \quad (\mathbf{H} + \mathbf{G}^{-1} \mathbf{R}) \mathbf{s} = -\mathbf{g}.$$

The CG method for (6.1) is summarized in Algorithm 1.

Convergence results for the CG method can be found, e.g., in the textbooks by Greenbaum [15, Chap. 3], Olshanskii and Tyrtshnikov [29]. For the case $\mathbf{R} = \beta \mathbf{G}$ with $\beta > 0$ the Hessian is of the form

$$\nabla^2 J = \mathbf{H} + \mathbf{G}^{-1} \mathbf{R} = \mathbf{H} + \beta \mathbf{I}$$

and we have the following result from Winther [37]. Since [37, Thm. 2.1] targets the CG method in Hilbert space and states the convergence result slightly differently, we include the theorem and its short proof. The $\nabla^2 J$ inner product is $\langle \mathbf{s}, \mathbf{s} \rangle_{\nabla^2 J} = \langle \mathbf{s}, \nabla^2 J \mathbf{s} \rangle$ and $\|\cdot\|_{\nabla^2 J}$ is the corresponding norm.

Algorithm 1 Conjugate Gradient.

-
0. Given $\text{tol}^{\text{cg}} \in (0, 1)$ and $\mathbf{s}_0$.
Compute $(\mathbf{H} + \mathbf{G}^{-1}\mathbf{R})\mathbf{s}_0$ and set $\mathbf{r}_0 = -\mathbf{g} - (\mathbf{H} + \mathbf{G}^{-1}\mathbf{R})\mathbf{s}_0$, $\mathbf{d}_0 = \mathbf{r}_0$.
 1. For $i = 0, 1, 2, \dots, i_{\max}$
 - (a) If $\|\mathbf{r}_i\| \leq \text{tol}^{\text{cg}}$, return $\mathbf{s}_i$.
 - (b) Compute $\mathbf{h}_i = (\mathbf{H} + \mathbf{G}^{-1}\mathbf{R})\mathbf{d}_i$.
 - (c) $\alpha_i = \langle \mathbf{r}_i, \mathbf{r}_i \rangle / \langle \mathbf{d}_i, \mathbf{h}_i \rangle$
 - (d) $\mathbf{s}_{i+1} = \mathbf{s}_i + \alpha_i \mathbf{d}_i$
 - (e) $\mathbf{r}_{i+1} = \mathbf{r}_i - \alpha_i \mathbf{h}_i$
 - (f) $\mathbf{d}_{i+1} = \mathbf{r}_{i+1} + \frac{\langle \mathbf{r}_{i+1}, \mathbf{r}_{i+1} \rangle}{\langle \mathbf{r}_i, \mathbf{r}_i \rangle} \mathbf{d}_i$
-

THEOREM 6. If $\mathbf{R} = \beta \mathbf{G}$ with $\beta > 0$, and if $\mu_1 \geq \mu_2 \geq \dots \geq \mu_n \geq 0$ are the eigenvalues of $\mathbf{H}$, then the iterates of the CG Algorithm 1 obey

$$\|\mathbf{s}_i - \mathbf{s}_*\|_{\nabla^2 J} \leq \left(\prod_{j=1}^i \frac{\mu_j}{\mu_j + \beta} \right) \|\mathbf{s}_0 - \mathbf{s}_*\|_{\nabla^2 J}.$$

Proof. The eigenvalues $\tilde{\mu}_j$ of $\nabla^2 J$ are given by $\tilde{\mu}_j = \beta + \mu_j$. Standard convergence analysis of the CG method shows that

$$(6.2) \quad \|\mathbf{s}_i - \mathbf{s}_*\|_{\nabla^2 J} \leq \min_{q \in \mathbb{P}_i, q(0)=1} \max_{\ell=1, \dots, n} |q(\tilde{\mu}_\ell)| \|\mathbf{s}_0 - \mathbf{s}_*\|_{\nabla^2 J},$$

where $\mathbb{P}_i$ is the space of polynomials of degree at most i . See, e.g., [15, pp. 50,51], [29], or [33, Lem. 6.5].

The polynomial

$$q(\mu) = \prod_{j=1}^i \frac{\tilde{\mu}_j - \mu}{\tilde{\mu}_j}$$

has degree i and satisfies $q(0) = 1$ and

$$\max_{\ell=1, \dots, n} |q(\tilde{\mu}_\ell)| = \max_{\ell=1, \dots, n} \prod_{j=1}^i \frac{|\tilde{\mu}_j - \tilde{\mu}_\ell|}{|\tilde{\mu}_j|} = \prod_{j=1}^i \frac{|\mu_j - \mu_n|}{\mu_j + \beta} \leq \prod_{j=1}^i \frac{\mu_j}{\mu_j + \beta}.$$

Inserting this bound into (6.2) gives the desired result. $\square$

6.2. CG method with ROM Hessian approximation. The most expensive computation in the CG Algorithm 1 occurs in step 1(b) with the computation of $\mathbf{H}\mathbf{d}_i$, which requires the solution of the two dynamical systems (2.5) (with initial data $\mathbf{d}_i$) and (2.6). Instead we use the reduced order Hessian approximation (2.11), (2.12), (2.13), (2.14). In Algorithm 2 we use a fixed ROM specified by the matrices $\mathbf{W}$, $\mathbf{V} \in \mathbb{R}^{n \times k}$. In the CG Algorithm 2, the fixed ROM could be the basic ROM computed from the (generalized) eigendecomposition of $(\mathbf{A}, \mathbf{M})$ or it could be the basic ROM augmented by a fixed vector $\mathbf{a} = \mathbf{g}$.

Since the rank of $\hat{\mathbf{H}}$ is at most k (c.f. (2.15)), Theorem 6 implies convergence of Algorithm 2 in k steps in the case where $\mathbf{R} = \beta \mathbf{G}$ with $\beta > 0$.

Algorithm 2 CG with Fixed ROM Hessian Approximation.

-
0. Given $\text{tol}^{\text{cg}} \in (0, 1)$ and $\mathbf{s}_0$.
 Generate ROM $\mathbf{W}, \mathbf{V} \in \mathbb{R}^{n \times k}$.
 Compute $(\hat{\mathbf{H}} + \mathbf{G}^{-1}\mathbf{R})\mathbf{s}_0$ and set $\mathbf{r}_0 = -\mathbf{g} - (\hat{\mathbf{H}} + \mathbf{G}^{-1}\mathbf{R})\mathbf{s}_0$, $\mathbf{d}_0 = \mathbf{r}_0$.
 1. For $i = 0, 1, 2, \dots, i_{\max}$
 - (a) If $\|\mathbf{r}_i\| \leq \text{tol}^{\text{cg}}$, return $\mathbf{s}_i$.
 - (b) Compute $\mathbf{h}_i = (\hat{\mathbf{H}} + \mathbf{G}^{-1}\mathbf{R})\mathbf{d}_i$.
 - (c) $\alpha_i = \langle \mathbf{r}_i, \mathbf{r}_i \rangle / \langle \mathbf{d}_i, \mathbf{h}_i \rangle$
 - (d) $\mathbf{s}_{i+1} = \mathbf{s}_i + \alpha_i \mathbf{d}_i$
 - (e) $\mathbf{r}_{i+1} = \mathbf{r}_i - \alpha_i \mathbf{h}_i$
 - (f) $\mathbf{d}_{i+1} = \mathbf{r}_{i+1} + \frac{\langle \mathbf{r}_{i+1}, \mathbf{r}_{i+1} \rangle}{\langle \mathbf{r}_i, \mathbf{r}_i \rangle} \mathbf{d}_i$
-

COROLLARY 7. If $\mathbf{R} = \beta \mathbf{G}$ with $\beta > 0$, and if $\hat{\mu}_1 \geq \dots \geq \hat{\mu}_k \geq \hat{\mu}_{k+1} = \dots = \hat{\mu}_n = 0$ are the eigenvalues of $\hat{\mathbf{H}}$, then the iterates of the CG Algorithm 2 obey

$$\|\mathbf{s}_i - \mathbf{s}_*\|_{\nabla^2 J} \leq \left(\prod_{j=1}^i \frac{\hat{\mu}_j}{\hat{\mu}_j + \beta} \right) \|\mathbf{s}_0 - \mathbf{s}_*\|_{\nabla^2 J}.$$

In particular, $\mathbf{s}_{k+1} = \mathbf{s}_*$.

From the definition of the CG method it follows that the iterates, the residuals, and the direction generated by the CG algorithm are contained in the (shifted) Krylov subspaces. In the notation of Algorithm 2, the iterates, the directions, and the residuals generated by the CG Algorithm 2 obey

$$\begin{aligned} \mathbf{s}_1, \dots, \mathbf{s}_{i+1} &\in \mathbf{s}_0 + \mathcal{K}_{i+1}(\hat{\mathbf{H}} + \mathbf{G}^{-1}\mathbf{R}, \mathbf{r}_0), \\ \mathbf{d}_0, \dots, \mathbf{d}_i &\in \mathcal{K}_{i+1}(\hat{\mathbf{H}} + \mathbf{G}^{-1}\mathbf{R}, \mathbf{r}_0), \\ \mathbf{r}_0, \dots, \mathbf{r}_i &\in \mathcal{K}_{i+1}(\hat{\mathbf{H}} + \mathbf{G}^{-1}\mathbf{R}, \mathbf{r}_0) \end{aligned}$$

with Krylov subspace

$$\mathcal{K}_{i+1}(\hat{\mathbf{H}} + \mathbf{G}^{-1}\mathbf{R}, \mathbf{r}_0) = \text{span}\{\mathbf{r}_0, (\hat{\mathbf{H}} + \mathbf{G}^{-1}\mathbf{R})\mathbf{r}_0, \dots, (\hat{\mathbf{H}} + \mathbf{G}^{-1}\mathbf{R})^i \mathbf{r}_0\}.$$

LEMMA 8. If $\mathbf{R} = \beta \mathbf{G}$, $\mathbf{G} = \mathbf{M}$ with $\beta > 0$, and if $\mathbf{r}_0 \in R(\mathbf{W})$, then $\mathbf{d}_i \in R(\mathbf{W})$ for all i .

Proof. The identity $\mathbf{R} = \beta \mathbf{G}$ with $\beta > 0$, and the definition of the Krylov subspace imply that $\mathcal{K}_{i+1}(\hat{\mathbf{H}} + \mathbf{G}^{-1}\mathbf{R}, \mathbf{r}_0) = \text{span}\{\mathbf{r}_0, \hat{\mathbf{H}}\mathbf{r}_0, \dots, \hat{\mathbf{H}}^i \mathbf{r}_0\}$. Since $\mathbf{G} = \mathbf{M}$, definition (2.14) of $\hat{\mathbf{H}}$ implies

$$\mathcal{K}_{i+1}(\hat{\mathbf{H}} + \mathbf{G}^{-1}\mathbf{R}, \mathbf{r}_0) = \text{span}\{\mathbf{r}_0, \hat{\mathbf{H}}\mathbf{r}_0, \dots, \hat{\mathbf{H}}^i \mathbf{r}_0\} \subset R(\mathbf{W}). \quad \square$$

If we use a ROM Hessian approximation in Algorithm 1, then we want $\mathbf{H}\mathbf{d}_i \approx \hat{\mathbf{H}}\mathbf{d}_i$ for all CG directions $\mathbf{d}_i$. Therefore, one may think of augmenting the basic ROM $\mathbf{V}_0$, $\mathbf{W}_0$ by $\mathbf{d}_i$ in the i th iteration to get $\mathbf{W}_{i+1}$, $\mathbf{V}_{i+1}$, as shown in sections 4.2.4 and 5. Thus the ROM changes from iteration to iteration, and therefore the ROM Hessian approximation $\hat{\mathbf{H}}$ changes from iteration to iteration and $\mathbf{d}_i \mapsto \hat{\mathbf{H}}\mathbf{d}_i$ is nonlinear. In general this requires a modification of the CG algorithm. However, in some cases the following naive implementation is sufficient even though the ROM Hessian approximation $\hat{\mathbf{H}}$ changes from iteration to iteration.

Algorithm 3 CG with Flexible ROM Hessian Approximation.

-
0. Given $tol^{cg} \in (0, 1)$ and $\mathbf{s}_0$.
 Generate a basic ROM $\mathbf{V}_0, \mathbf{W}_0$.
 Compute $(\hat{\mathbf{H}} + \mathbf{G}^{-1}\mathbf{R})\mathbf{s}_0$ and set $\mathbf{r}_0 = -\mathbf{g} - (\hat{\mathbf{H}} + \mathbf{G}^{-1}\mathbf{R})\mathbf{s}_0$, $\mathbf{d}_0 = \mathbf{r}_0$.
 1. For $i = 0, 1, 2, \dots, i_{\max}$
 - (a) If $\|\mathbf{r}_i\| \leq tol^{cg}$, return $\mathbf{s}_i$.
 - (b) Augment the ROM $\mathbf{V}_0, \mathbf{W}_0$ by $\mathbf{d}_i$ to get $\mathbf{V}_{i+1}, \mathbf{W}_{i+1}$.
 - (c) Compute $\mathbf{h}_i = (\hat{\mathbf{H}} + \mathbf{G}^{-1}\mathbf{R})\mathbf{d}_i$.
 - (d) $\alpha_i = \langle \mathbf{r}_i, \mathbf{r}_i \rangle / \langle \mathbf{d}_i, \mathbf{h}_i \rangle$
 - (e) $\mathbf{s}_{i+1} = \mathbf{s}_i + \alpha_i \mathbf{d}_i$
 - (f) $\mathbf{r}_{i+1} = \mathbf{r}_i - \alpha_i \mathbf{h}_i$
 - (g) $\mathbf{d}_{i+1} = \mathbf{r}_{i+1} + \frac{\langle \mathbf{r}_{i+1}, \mathbf{r}_{i+1} \rangle}{\langle \mathbf{r}_i, \mathbf{r}_i \rangle} \mathbf{d}_i$
-

Next we show that for some problems, Algorithms 3 and 2 are identical.

LEMMA 9. If $\mathbf{R} = \beta\mathbf{G}$, $\mathbf{G} = \mathbf{M}$ with $\beta > 0$, and if $\mathbf{r}_0 \in R(\mathbf{W}_0)$, then $\mathbf{d}_i \in R(\mathbf{W}_0)$ and, therefore $R(\mathbf{W}_i) = R(\mathbf{W}_0)$ for all i .

Proof. The proof is essentially a consequence of Lemma 8 and can be obtained by induction. We omit the details. $\square$

THEOREM 10. Let $\mathbf{R} = \beta\mathbf{G}$, $\mathbf{G} = \mathbf{M}$ with $\beta > 0$, and let $\mathbf{W}_0 = \mathbf{V}_0$. If $\mathbf{r}_0 \in R(\mathbf{V}_0)$ and if the ROM in step 1(b) of Algorithm 3 is updated as in (5.5) with $\mathbf{a} = \mathbf{d}_i$, then $\mathbf{W}_i = \mathbf{V}_i = \mathbf{W}_0 = \mathbf{V}_0$ for all i and Algorithms 3 and 2 are identical.

Proof. This result immediately follows from Lemma 9. $\square$

Remark 11. If $\mathbf{s}_0 = \mathbf{0}$ then $\mathbf{r}_0 = -\mathbf{g}$ can be easily incorporated into ROM $\mathbf{W}_0$ as shown in sections 4.2.3 and 5.

7. Numerical results. We consider a P1 finite element semidiscretization of the model problem (3.1)

$$\beta = 10^{-4}, \quad \gamma = 0.5, \quad \Omega_o = \Omega,$$

and diffusion κ and advection a specified in the following subsections. This leads to (1.1) with

$$\mathbf{Q} = \mathbf{Q}_T = \mathbf{M}, \quad \mathbf{R} = \beta\mathbf{M}, \quad \mathbf{G} = \mathbf{M}.$$

To numerically solve (1.1) we discretize the state equation (1.1b) in time using the implicit Euler method and we use a consistent quadrature for the integrals in the objective (1.1a). Our ROMs are applied to the Hessian of the fully discretized problem, which requires no modification in the construction of our ROMs.

7.1. Model problem with symmetric $\mathbf{A}$. In this subsection we consider the implicit Euler time discretization of (1.1) obtained from (3.1) without advection, $a = (0, 0, 0)^T$, and diffusion $\kappa = 0.1$. This leads to a symmetric $\mathbf{A}$. The spatial grid is of size $30 \times 30 \times 30$, which leads to a full order problem of size $n = 29,791$. The number of time steps is 50. We set $\mathbf{g} = \nabla J(\mathbf{0})$. The optimization problem in this subsection satisfies assumptions (4.1).

First we solve $(\mathbf{H}^{\text{bsc}} + \beta\mathbf{I})\mathbf{s}^{\text{bsc}} = -\mathbf{g}$ and $(\mathbf{H}^{\text{aug}} + \beta\mathbf{I})\mathbf{s}^{\text{aug}} = -\mathbf{g}$ using the CG Algorithm 2 with fixed ROM Hessian approximation. We set $tol^{cg} = tol \|\mathbf{g}\|$ with $tol = 10^{-2}$ and $tol = 10^{-4}$ (the results for $tol = 10^{-9}$ are close to those for $tol = 10^{-4}$). Theorem 3 guarantees a decrease in the solution error when the basic ROM Hessian $\mathbf{H}^{\text{bsc}} + \beta\mathbf{I}$ is replaced by the augmented ROM Hessian $\mathbf{H}^{\text{aug}} + \beta\mathbf{I}$, but this theorem

FIG. 3. *Model problem with symmetric A*: Relative errors between the exact solution of $(\mathbf{H} + \beta\mathbf{I})\mathbf{s} = -\mathbf{g}$ and the solution $\mathbf{s}^{\text{bsc}}$ of the system with basic ROM Hessian approximation (red lines) and the solution $\mathbf{s}^{\text{aug}}$ of the system with augmented (by $\mathbf{a} = \mathbf{g}$) ROM Hessian approximation (blue lines). The systems are solved using the CG Algorithm 2 with tolerance $\text{tol}^{\text{cg}} = 10^{-2}\|\mathbf{g}\|$ (left plot) and with $\text{tol}^{\text{cg}} = 10^{-4}\|\mathbf{g}\|$ (right plot). While the augmented ROM Hessian approximation produces better results, a coarse CG stopping tolerance can negatively impact the improvement achieved by the augmented ROM Hessian approximation.

assumes exact solution of the linear systems. The decrease in the solution error using the augmented ROM compared to the basic ROM can also be observed in practice, where the linear system is solved by the CG method; see Figure 3. Nevertheless, with coarse tolerance tol^{cg} the difference in solution quality may be less large, because of early termination of the CG algorithm resulting possibly in a better approximation (see, e.g., basic ROM solutions for $k \geq 140$ in left plot of Figure 3).

Since the optimization problem in this subsection satisfies assumptions (4.1), Theorem 10 applies and CG Algorithms 3 and 2 are identical (in exact arithmetic). This is confirmed by the results in Table 1. We solve the systems $(\mathbf{H} + \beta\mathbf{I})\mathbf{s} = -\mathbf{g}$, $(\mathbf{H}^{\text{bsc}} + \beta\mathbf{I})\mathbf{s}^{\text{bsc}} = -\mathbf{g}$, as well as $(\mathbf{H}^{\text{aug}} + \beta\mathbf{I})\mathbf{s}^{\text{aug}} = -\mathbf{g}$ using CG Algorithms 1, 2, and 3. In all cases the basic ROM size is $k = 20$, the initial CG iterate is $\mathbf{s}_0 = \mathbf{0}$, and the stopping tolerance is $\text{tol}^{\text{cg}} = 10^{-4}\|\mathbf{g}\|$. Again the solutions computed by augmented ROMs are significantly better than that using the basic ROM. In fact, the basic ROM Hessian approximation generates $\mathbf{s}^{\text{bsc}}$ with larger objective function value and gradient norm than those at $\mathbf{s}^0$. Moreover, the last two lines in Table 1 show that the CG Algorithms 3 and 2 generate the same result.

7.2. Model problem with nonsymmetric A. In this subsection we consider the implicit Euler time discretization of (1.1) obtained from (3.1) with advection $\mathbf{a} = (2, 1, 1)^T/\sqrt{6}$, and diffusion $\kappa = 0.1$. This leads to a nonsymmetric $\mathbf{A}$. The spatial grid is of size $30 \times 30 \times 30$, which leads to a full order problem of size $n = 29,791$. The number of time steps is 50. We set $\mathbf{g} = \nabla J(\mathbf{0})$. The matrix $\mathbf{M}^{-1}\mathbf{A}$ is diagonalizable, but it has complex eigenvalues.

Again, we first solve $(\mathbf{H}^{\text{bsc}} + \beta\mathbf{I})\mathbf{s}^{\text{bsc}} = -\mathbf{g}$ and $(\mathbf{H}^{\text{aug}} + \beta\mathbf{I})\mathbf{s}^{\text{aug}} = -\mathbf{g}$ using CG Algorithm 2 with fixed ROM Hessian approximation. We set $\text{tol}^{\text{cg}} = \text{tol} \|\mathbf{g}\|$ with $\text{tol} = 10^{-2}$ and $\text{tol} = 10^{-4}$. The results are illustrated in Figure 4 below. Also for nonsymmetric problems the augmentation of the basic ROM $\mathbf{V}, \mathbf{W}$ by $\mathbf{a} = \mathbf{g}$ improves the approximations to $\mathbf{s}$ significantly. Again the CG tolerance tol^{cg} impacts the difference in solution quality, because of early termination of the CG algorithm.

TABLE 1

Model problem with symmetric $\mathbf{A}$: Solution of the full order system $(\mathbf{H} + \beta \mathbf{I})\mathbf{s}_{\text{full}} = -\mathbf{g}$ and the corresponding system with various ROM Hessian approximations. The basic ROM size is $k = 20$. The relative error is $\text{rel err} = \|\mathbf{s}_{\text{full}} - \mathbf{s}_{\text{rom}}\|^2 / \|\mathbf{s}_{\text{full}}\|^2$. The augmented ROM Hessian approximations $\mathbf{H}^{\text{aug}}$ produce significantly better approximations to $\mathbf{s}$ than the basic ROM Hessian approximation $\mathbf{H}^{\text{bsc}}$. For this problem CG Algorithm 2 ($\mathbf{H}^{\text{aug}}$ with fixed augmentation $\mathbf{a} = \mathbf{g}$) and CG Algorithm 3 ($\mathbf{H}^{\text{aug}}$ with variable augmentation $\mathbf{a} = \mathbf{d}_i$) are identical.

	J	$\ \nabla J\ $	CG iters	ROM size	PDE solves full / ROM	rel err
at $\mathbf{s}_0 = \mathbf{0}$	$1.46e-2$	$5.91e-2$				
Full Hessian	$1.75e-5$	$5.61e-6$	24		50 / 0	
$\mathbf{H}^{\text{bsc}}$	$1.82e+1$	$8.47e-1$	11	20	2 / 22	$5.44e+3$
$\mathbf{H}^{\text{aug}}, \mathbf{a} = \mathbf{g}$	$5.27e-5$	$8.68e-4$	9	21	2 / 18	$3.59e-2$
$\mathbf{H}^{\text{aug}}, \mathbf{a} = \mathbf{d}_i$	$5.27e-5$	$8.68e-4$	9	21	2 / 18	$3.59e-2$

FIG. 4. Model problem with nonsymmetric $\mathbf{A}$: Relative errors between the exact solution of $(\mathbf{H} + \beta \mathbf{I})\mathbf{s} = -\mathbf{g}$ and the solution $\mathbf{s}^{\text{bsc}}$ of the system with basic ROM Hessian approximation (red lines) and the solution $\mathbf{s}^{\text{aug}}$ of the system with augmented (by $\mathbf{a} = \mathbf{g}$) ROM Hessian approximation (blue lines). The systems are solved using CG Algorithm 2 with tolerance $\text{tol}^{\text{cg}} = 10^{-2}\|\mathbf{g}\|$ (left plot) and with $\text{tol}^{\text{cg}} = 10^{-4}\|\mathbf{g}\|$ (right plot). While the augmented ROM Hessian approximation produces better results, a coarse CG stopping tolerance can negatively impact the improvement achieved by the augmented ROM Hessian approximation.

Table 2 below shows the differences between using the full Hessian, the basic ROM Hessian approximation, and the augmented ROM Hessian for different diffusion coefficients κ and different ROM sizes k . We use CG Algorithm 1 and CG Algorithm 2 with fixed ROM Hessian approximation (either basic ROM or augmented ROM by $\mathbf{a} = \mathbf{g}$). Again the initial CG iterate is $\mathbf{s}_0 = \mathbf{0}$, and the stopping tolerance is $\text{tol}^{\text{cg}} = 10^{-4}\|\mathbf{g}\|$. In all settings the augmented (by $\mathbf{a} = \mathbf{g}$) ROM Hessian approach gives better approximations and needs fewer CG iterations than the basic ROM approach. With increasing ROM sizes the approximate solutions become better for both the basic ROM approach and the augmented ROM approach. For decreasing κ (i.e., problem becomes more advection dominated) the CG method needs more iterations and larger ROM sizes are required to meet a particular accuracy. While augmenting the ROM by $\mathbf{a} = \mathbf{g}$ still dramatically improves the quality of the approximate solution, the accuracy of the approximate solutions for fixed ROM size k are less good if the advection becomes more important. Additional reduction of κ further reduces the approximation quality of the ROMs (not reported here). A better understanding of

TABLE 2

Model problem with nonsymmetric $\mathbf{A}$: Solution of the full order system $(\mathbf{H} + \beta \mathbf{I})\mathbf{s} = -\mathbf{g}$ and the corresponding system with various ROM Hessian approximations. The diffusion coefficient κ as well as the basic ROM sizes k vary. The augmented (by $\mathbf{a} = \mathbf{g}$) ROM Hessian approximations $\mathbf{H}^{\text{aug}}$ produce significantly better approximations to $\mathbf{s}$ than the basic ROM Hessian approximations $\mathbf{H}^{\text{bsc}}$. The benefit of augmented ROM Hessian approximations increases for more advection-dominated problems.

$\kappa = 0.1$					
k		J	$\ \nabla J\ $	CG iters	$\frac{\ \mathbf{s}_{\text{full}} - \mathbf{s}_{\text{rom}}\ ^2}{\ \mathbf{s}_{\text{full}}\ ^2}$
	at $\mathbf{s}_0 = \mathbf{0}$ $\mathbf{H}$	1.41e-2 1.75e-5	5.61e-2 4.70e-6	24	
10	$\mathbf{H}^{\text{bsc}}$ $\mathbf{H}^{\text{aug}}, \mathbf{a} = \mathbf{g}$	2.24e+2 4.58e-4	4.42e+0 4.83e-3	10 9	3.43e+4 1.43e-1
30	$\mathbf{H}^{\text{bsc}}$ $\mathbf{H}^{\text{aug}}, \mathbf{a} = \mathbf{g}$	2.34e+1 3.27e-4	1.04e+0 3.63e-3	24 15	6.69e+3 1.18e-1
50	$\mathbf{H}^{\text{bsc}}$ $\mathbf{H}^{\text{aug}}, \mathbf{a} = \mathbf{g}$	4.76e+0 6.61e-5	3.78e-1 1.03e-3	28 19	2.05e+3 4.23e-2
70	$\mathbf{H}^{\text{bsc}}$ $\mathbf{H}^{\text{aug}}, \mathbf{a} = \mathbf{g}$	1.90e+0 5.34e-5	2.14e-1 7.92e-4	31 20	1.01e+3 3.80e-2
$\kappa = 0.07$					
k		J	$\ \nabla J\ $	CG iters	$\frac{\ \mathbf{s}_{\text{full}} - \mathbf{s}_{\text{rom}}\ ^2}{\ \mathbf{s}_{\text{full}}\ ^2}$
	at $\mathbf{s}_0 = \mathbf{0}$ $\mathbf{H}$	1.89e-2 1.76e-5	7.41e-2 6.86e-6	24	
10	$\mathbf{H}^{\text{bsc}}$ $\mathbf{H}^{\text{aug}}, \mathbf{a} = \mathbf{g}$	1.52e+3 2.47e-3	1.52e+1 1.76e-2	12 10	1.36e+5 3.02e-1
30	$\mathbf{H}^{\text{bsc}}$ $\mathbf{H}^{\text{aug}}, \mathbf{a} = \mathbf{g}$	1.73e+2 1.84e-3	3.85e+0 1.28e-2	43 28	2.74e+4 2.92e-1
50	$\mathbf{H}^{\text{bsc}}$ $\mathbf{H}^{\text{aug}}, \mathbf{a} = \mathbf{g}$	4.31e+1 4.96e-4	1.55e+0 5.19e-3	60 36	1.04e+4 1.33e-1
70	$\mathbf{H}^{\text{bsc}}$ $\mathbf{H}^{\text{aug}}, \mathbf{a} = \mathbf{g}$	1.96e+1 2.54e-4	9.52e-1 3.26e-3	75 47	5.70e+3 8.58e-2

the dependence of the ROM quality on the ratio of advection divided by diffusion (i.e., on the Peclet number) is part of ongoing investigations.

Since $\mathbf{A}$ is nonsymmetric, $\mathbf{V} \neq \mathbf{W}$ in the basic ROM and, therefore, the assumptions in Theorem 10 are not satisfied. If CG Algorithm 3 is used, the ROM matrices $\mathbf{V}_i$ and $\mathbf{W}_i$ change from iteration to iteration and CG Algorithm 3 does not converge (it terminates with the message that negative curvature is detected, $\langle \mathbf{d}_i, \mathbf{h}_i \rangle < 0$ in step 1(c) of Algorithm 3). We have also replaced CG Algorithm 3 by a flexible CG following Notay [28], which is designed to handle matrix-vector products that are subject to sufficiently small inexactness. However, the flexible CG also did not converge, since the error due to change in bases is too large.

8. Conclusions. This paper has introduced a new ROM Hessian approximation for linear-quadratic optimal control problems where the optimal control is the initial value. The necessary and sufficient optimality conditions for these problems are given by the linear system $\nabla^2 J \mathbf{s}_* = -\nabla J(\mathbf{0})$, where $\nabla J(\mathbf{0})$ is the gradient of the objective

and $\nabla^2 J$ is the Hessian of the objective, which is independent of the point at which it is computed. The computation of a Hessian vector product $\nabla^2 J \mathbf{s}$ requires the solution of the linearized state equation with initial value given by $\mathbf{s}$, followed by the solution of the second order adjoint equation. Projection-based ROMs of these two linear differential equations are used to generate the ROM Hessian approximation. The basic ROM is computed based on the structure of these two dynamical systems and in important special cases results in a best k -rank Hessian approximation. However, when this basic ROM Hessian is used in the optimization, it can lead to poor results. The difficulty is that the solution of $\nabla^2 J \mathbf{s}_* = -\nabla J(\mathbf{0})$ may have important contributions in subspaces not included in the basic ROM. A fundamental challenge is that in general no fixed ROM well-approximates the application of the Hessian to all possible vectors of initial data.

The main innovation of this paper is an augmentation of the basic ROM by one vector, which is either the right-hand side $\nabla J(\mathbf{0})$ of the linear system, or the vector of initial data $\mathbf{s}$ to which the Hessian is applied. These augmentations are computationally inexpensive, but often lead to substantial improvements in the ROM Hessian approximations. For important special cases, these improvements are quantified theoretically, and for various cases illustrated numerically. When the basic ROM is augmented by the vector of initial data $\mathbf{s}$ to which the Hessian is applied, then the resulting augmented ROM Hessian vector product is nonlinear. While in general, this can negatively impact the application of the CG method, we identify important special cases where only one augmentation is needed, and hence the linear CG method can be applied.

Several extensions of the work presented in this paper are ongoing or future research. First, even though our augmented ROM Hessian generates better approximations $\hat{\mathbf{s}}_*$ of the optimal solution $\mathbf{s}_*$ than the basic ROM, these may not be good enough. One can incorporate our augmented ROM Hessians into an outer iteration, where a sequence of systems $\nabla^2 J \delta \mathbf{s} = -\nabla J(\mathbf{s}_c)$ are solved approximately and the new iterate is $\mathbf{s}_+ = \mathbf{s}_c + \delta \mathbf{s}$. For problems satisfying (4.1) this sequential approach is presented [17]. The development of such a sequential approach for more general problems is a subject of future research. Second, it seems possible to extend our approach to classic PDE constrained problems (3.1) and use different spatial discretization levels for different components of our augmented approach. This is work in progress. Third, when the basic ROM is augmented by the vector $\mathbf{s}$ to which the Hessian is applied, the mapping $\mathbf{s} \mapsto \nabla^2 J \mathbf{s}$ is nonlinear. For sufficiently small nonlinearities, the flexible CG could be used. While in our tests mentioned in section 7.2, the nonlinearities were too large for the flexible CG presented in Notay [28] to converge, a better understanding of when a flexible CG could be applied would be beneficial. Finally, as mentioned earlier, problems of the type (1.1) arise as subproblems in multiple shooting formulations of optimal control problems. Incorporation of the new ROM Hessian approximation into the solution of multiple shooting formulations is part of our future research.

REFERENCES

- [1] V. AKÇELİK, G. BIROS, O. GHATTAS, K. R. LONG, AND B. VAN BLOEMEN WAANDERS, *A variational finite element method for source inversion for convective-diffusive transport*, Finite Elem. Anal. Des., 39 (2003), pp. 683–705.
- [2] H. ANTIL, M. HEINKENSCHLOSS, R. H. W. HOPPE, AND D. C. SORESENSEN, *Domain decomposition and model reduction for the numerical solution of PDE constrained optimization problems with localized optimization variables*, Comput. Vis. Sci., 13 (2010), pp. 249–264.

- [3] A. C. ANTOUNAS, *Approximation of large-scale dynamical systems*, Adv. Des. Control 6, SIAM, Philadelphia, 2005.
- [4] E. ARIAN, M. FAHL, AND E. W. SACHS, *Trust-region Proper Orthogonal Decomposition for Flow Control*, Technical report 2000–25, ICASE, NASA Langley Research Center, Hampton, VA, 2000.
- [5] O. BASHIR, K. WILLCOX, O. GHATTAS, B. VAN BLOEMENWAANDERS, AND J. HILL, *Hessian-based model reduction for large-scale systems with initial-condition inputs*, Internat. J. Numer. Methods Engrg., 73 (2008), pp. 844–868.
- [6] C. BEATTIE, S. GUGERCIN, AND V. MEHRMANN, *Model reduction for systems with inhomogeneous initial conditions*, Systems Control Lett., 99 (2017), pp. 99–106.
- [7] P. BENNER, S. GUGERCIN, AND K. WILLCOX, *A survey of projection-based model reduction methods for parametric dynamical systems*, SIAM Rev., 57 (2015), pp. 483–531.
- [8] P. BENNER, E. SACHS, AND S. VOLKWEIN, *Model order reduction for PDE constrained optimization*, in Trends in PDE Constrained Optimization, G. Leugering, P. Benner, S. Engell, A. Griewank, H. Harbrecht, M. Hinze, R. Rannacher, and S. Ulbrich, eds., Internat. Ser. Numer. Math. 165, Birkhäuser/Springer, Cham, 2014, pp. 303–326.
- [9] J. BLUM, F.-X. LE DIMET, AND I. M. NAVON, *Data assimilation for geophysical fluids*, in Computational Methods for the Atmosphere and the Oceans, R. M. Temam and J. J. Tribbia, eds., Handb. Numer. Anal. 14, Elsevier, Amsterdam, 2009, pp. 385–441.
- [10] M. CALIARI, P. KANDOLF, A. OSTERMANN, AND S. RAINER, *Comparison of software for computing the action of the matrix exponential*, BIT, 54 (2014), pp. 113–128.
- [11] T. CARRARO AND M. GEIGER, *Direct and indirect multiple shooting for parabolic optimal control problems*, in Multiple Shooting and Time Domain Decomposition Methods, MuS-TDD, Heidelberg, 2013, T. Carraro, M. Geiger, S. Körkel, and R. Rannacher, eds., Contrib. Math. Comput. Sci. 9, Springer, Heidelberg, 2015, pp. 35–67.
- [12] D. N. DAESCU AND I. M. NAVON, *Efficiency of a POD-based reduced second-order adjoint model in 4d-var data assimilation*, Internat. J. Numer. Methods Fluids, 53 (2007), pp. 985–1004.
- [13] A. FROMMER AND V. SIMONCINI, *Matrix functions*, in Model Order Reduction: Theory, Research Aspects and Applications, W. H. A. Schilders, H. A. van der Vorst, and J. Rommes, eds., Math. Ind. 13, Springer, Berlin, 2008, pp. 275–303.
- [14] G. H. GOLUB AND C. F. V. LOAN, *Matrix Computations*, Johns Hopkins Ser. Math. Sci. 3, 2nd ed., Johns Hopkins University Press, Baltimore, MD, 1989.
- [15] A. GREENBAUM, *Iterative Methods for Solving Linear Systems*, Front. Appl. Math. 17, SIAM, Philadelphia, 1997.
- [16] M. GUBISCH AND S. VOLKWEIN, *Proper orthogonal decomposition for linear-quadratic optimal control*, in Model Reduction and Approximation: Theory and Algorithms, P. Benner, A. Cohen, M. Ohlberger, and K. Willcox, eds., Comput. Sci. Eng., SIAM, Philadelphia, 2017, pp. 3–64.
- [17] M. HEINKENSCHLOSS AND D. JANDO, *Sequential reduced order modeling for time-dependent optimization problems with initial value controls*, in Reduced-Order Modeling for Simulation and Optimization, W. Keiper, A. Milde, and S. Volkwein, eds., Springer, Heidelberg, to appear.
- [18] M. HEINKENSCHLOSS, T. REIS, AND A. ANTOUNAS, *Balanced truncation model reduction for systems with inhomogeneous initial conditions*, Automatica J. IFAC, 47 (2011), pp. 559 – 564.
- [19] H. K. HESSE AND G. KANSCHAT, *Mesh adaptive multiple shooting for partial differential equations. I. Linear quadratic optimal control problems*, J. Numer. Math., 17 (2009), pp. 195–217.
- [20] J. S. HESTHAVEN, G. ROZZA, AND B. STAMM, *Certified Reduced Basis Methods for Parametrized Partial Differential Equations*, Springer Briefs Math., Springer, New York, 2015.
- [21] M. HINZE, R. PINNAU, M. ULBRICH, AND S. ULBRICH, *Optimization with PDE Constraints*, Math. Model. Theory Appl. 23, Springer, New York, 2008.
- [22] M. HOCHBRUCK AND C. LUBICH, *On Krylov subspace approximations to the matrix exponential operator*, SIAM J. Numer. Anal., 34 (1997), pp. 1911–1925.
- [23] R. A. HORN AND C. A. JOHNSON, *Matrix Analysis*, 2nd ed., Cambridge University Press, Cambridge, 2013.
- [24] L. KOHAUPT, *Introduction to a Gram-Schmidt-type biorthogonalization method*, Rocky Mountain J. Math., 44 (2014), pp. 1265–1279.
- [25] D. P. KOURI, M. HEINKENSCHLOSS, D. RIDZAL, AND B. G. VAN BLOEMEN WAANDERS, *Inexact objective function evaluations in a trust-region algorithm for PDE-constrained optimization under uncertainty*, SIAM J. Sci. Comput., 36 (2014), pp. A3011–A3029.

- [26] A. J. LAUB, *Matrix Analysis for Scientists and Engineers*, SIAM, Philadelphia, 2005.
- [27] D. LIBERZON, *Calculus of Variations and Optimal Control Theory. A Concise Introduction*, Princeton University Press, Princeton, NJ, 2012.
- [28] Y. NOTAY, *Flexible conjugate gradients*, SIAM J. Sci. Comput., 22 (2000), pp. 1444–1460.
- [29] M. A. OLSHANSKII AND E. E. TYRTYSHNIKOV, *Iterative Methods for Linear Systems: Theory and Applications*, SIAM, Philadelphia, 2014.
- [30] B. N. PARLETT, D. R. TAYLOR, AND Z. A. LIU, *A look-ahead Lanczos algorithm for unsymmetric matrices*, Math. Comp., 44 (1985), pp. 105–124.
- [31] A. QUARTERONI, A. MANZONI, AND F. NEGRI, *Reduced Basis Methods for Partial Differential Equations. An Introduction*, Unitext 92, Springer, Cham, 2016.
- [32] V. RAO AND A. SANDU, *A time-parallel approach to strong-constraint four-dimensional variational data assimilation*, J. Comput. Phys., 313 (2016), pp. 583 – 593.
- [33] Y. SAAD, *Iterative Methods for Sparse Linear Systems*, PWS Publishing, Boston, 1996.
- [34] E. W. SACHS AND S. VOLKWEIN, *POD-Galerkin approximations in PDE-constrained optimization*, GAMM-Mitt., 33 (2010), pp. 194–208.
- [35] R. ȘTEFĂNESCU, A. SANDU, AND I. M. NAVON, *POD/DEIM reduced-order strategies for efficient four dimensional variational data assimilation*, J. Comput. Phys., 295 (2015), pp. 569–595.
- [36] F. TRÖLTZSCH, *Optimal Control of Partial Differential Equations: Theory, Methods and Applications*, Grad. Stud. Math. 112, AMS, Providence, RI, 2010.
- [37] R. WINTHER, *Some superliner convergence results for the conjugate gradient method*, SIAM J. Numer. Anal., 17 (1980), pp. 14–17.