2009 American Control Conference
Hyatt Regency Riverfront, St. Louis, MO, USA
June 10-12, 2009

A Comparison of Balanced Truncation Methods for Closed Loop Systems

John R. Singler
Department of Mathematics and Statistics
Missouri University of Science and Technology
Rolla, MO 65409
Email: singlerj@mst.edu

Belinda A. Batten
School of Mechanical, Industrial, and Manufacturing Engineering
Oregon State University
Corvallis, OR 97331–6001
Email: bbatten@engr.orst.edu

Abstract— **Real-time control of a physical system necessitates controllers that are low order. In this paper, we compare two balanced truncation methods as a means of designing low order compensators for partial differential equation (PDE) systems. The first method is the application of balanced truncation to the compensator dynamics, rather than the state dynamics, as was done in [1]. The second method, LQG balanced truncation, applies the balancing technique to the Riccati operators obtained from a specific LQG design. We discuss snapshot-based algorithms for constructing the reduced order compensators and present numerical results for a two dimensional convection diffusion PDE system.**

I. Introduction

Practical methods that can be used to reduce the size of a controller designed for an infinite dimensional system (in particular, a partial differential equation system) have been the focus of much research of the last decade. A challenge is developing a method that preserves properties of the closed loop system and does not discard important dynamics in the reduction process. In this paper, we compare two reduction methods that first compute a converged approximation to the compensator for the infinite dimensional system, and then reduce the compensator. Both methods apply balanced truncation but in different ways. The first method applies it to the Gramians for the compensator system; this method was proposed for finite dimensional systems by Yousuff and Skelton in [1]. The second method, LQG balancing, applies balancing to the solutions of the Riccati equations. Although these methods have been known for some time, only the second appears to have significant theory developed in the context of control design for infinite dimensional systems [2], [3], [4]. The first method has been formally applied to an infinite dimensional system in control of nonlinear convection in [5]. The second method has been applied to PDE systems in [6], [7]. Although there are many aspects to consider in the comparison of such designs, we begin the investigation in this paper by considering the methods as applied to a two dimensional convection diffusion system.

The weakness of many model reduction techniques when applied to systems that are modeled by partial differential equations (PDEs) is that there are typically gaps in what can be proven with regard to convergence of computed controllers to the PDE controllers. In [2], it was shown that LQG balanced truncation followed by a central controller design for the low order model could yield a robust enough controller to handle the lost dynamics in truncation. In this investigation, we instead focus on the two approaches outlined above: (1) balancing and truncating the LQG compensator directly, and (2) computing a reduced LQG controller using an LQG balanced truncated model.

The first step for both of these methods is to compute an approximation to the infinite dimensional compensator. Then, the two balanced truncation methods are applied to this controller to yield low order compensators. Much recent research has focused on algorithms for large-scale matrix equations and model reduction problems for large-scale systems resulting from the discretization of infinite dimensional systems (see, e.g., [8], [9] and the references contained therein). In this work, we use snapshot-based algorithms to construct approximations to the PDE controller and the reduced order models. The algorithms used here are related to those proposed by Wilcox and Peraire [10] and Rowley [11] for finite dimensional systems.

II. The Model Problem

To study the effects of balancing the compensator and LQG balancing, we consider the model problem given by a convection diffusion equation with nonconstant convection coefficients over the spatial domain $\Omega = [0,1] \times [0,1]$. The model problem is given by

$$w_t = \mu(w_{xx} + w_{yy}) - c_1(x,y)w_x - c_2(x,y)w_y + b(x,y)u(t),$$

with Dirichlet boundary conditions on the bottom, right, and top walls:

$$w(t,x,0) = 0, \quad w(t,1,y) = 0, \quad w(t,x,1) = 0,$$

a Neumann boundary condition on the left wall:

$$w_x(t,0,y) = 0,$$

and initial condition

$$w(0,x,y) = w_0(x,y).$$

System measurements are taken of the form

$$\eta(t) = \int_\Omega c(x,y)w(t,x,y)\,dx\,dy.$$

This work was supported in part by the Air Force Office of Scientific Research under grant FA9550-07-1-0540.

We assume the convection coefficients $c_1(x,y)$ and $c_2(x,y)$ are bounded, and we assume the functions $b(x,y)$ and $c(x,y)$ are square integrable over Ω.

We chose this model problem to investigate the feasibility of using the two compensator reduction methods for linearized incompressible fluid flow problems. This model problem shares similarities to linear flow problems, but is a simpler testing platform. In anticipation of using such controller reduction techniques on more complex problems, we use special numerical methods to compute the reduced order compensators.

For the snapshot algorithms below, we require an abstract formulation of the problem. Briefly, this can be done as follows. Let X be the Hilbert space $L^2(\Omega)$ of square integrable functions defined over the domain Ω with standard inner product $(f,g) = \int_\Omega f(x,y)g(x,y)\,dx\,dy$ and norm $\|f\| = (f,f)^{1/2}$. Define the convection diffusion operator $A : D(A) \subset X \to X$ by

$$[Aw](x,y) = \mu(w_{xx} + w_{yy}) - c_1 w_x - c_2 w_y,$$

Roughly, functions in $D(A)$ are twice differentiable and satisfy the above boundary conditions. Define $B : \mathbb{R} \to X$ and $C : X \to \mathbb{R}$ by $[Bu](x,y) = b(x,y)u$ and $Cw = (w,c)$. In this way, the PDE system can be written as the infinite dimensional system

$$\dot{w}(t) = Aw(t) + Bu(t), \quad w(0) = w_0, \quad y(t) = Cw(t),$$

where the dot denotes a time derivative.

III. Background

We now discuss control design and model reduction for a general infinite dimensional system

$$\dot{x}(t) = Ax(t) + Bu(t) + Dw(t), \quad y(t) = Cx(t),$$

holding over a Hilbert space X. We assume the operator $A : D(A) \subset X \to X$ generates a C_0-semigroup, and the control input operator $B : \mathbb{R}^m \to X$, the disturbance input operator $D : \mathbb{R}^n \to X$, and the observation operator $C : X \to \mathbb{R}^p$ are all bounded.

A. Control Design for PDEs

We consider the control objective to minimize the cost

$$J = \int_0^\infty \|Ex(t)\|^2 + \|u(t)\|^2\,dt,$$

where the controlled output operator $E : X \to \mathbb{R}^q$ is also bounded. Under certain assumptions, the solution to this problem is the feedback control law given by

$$u(t) = -Kx_c(t), \quad \dot{x}_c(t) = A_c x_c(t) + Fy(t), \tag{1}$$

where

$$K = B^*\Pi, \quad F = PC^*, \quad A_c = A - BK - FC,$$

and the bounded operators $\Pi : X \to X$ and $P : X \to X$ are the solutions of the control and filter algebraic Riccati equations (AREs)

$$A^*\Pi + \Pi A - \Pi BB^*\Pi + E^*E = 0, \tag{2}$$

$$AP + PA^* - PC^*CP + DD^* = 0, \tag{3}$$

where the asterisk (*) denotes the Hilbert adjoint operator.

Once the gains are computed, a difficulty with the implementation of this control law is that one must solve the infinite dimensional linear differential equation for the state estimate $x_c(t)$ in (1) in real time. Therefore, model reduction is required to create a controller that is implementable in real time.

Both model reduction methods considered below will produce reduced order compensators of the form

$$u(t) = -K_r x_c^r(t), \quad \dot{x}_c^r(t) = A_{cr} x_c^r(t) + F_r y(t), \tag{4}$$

where x_c^r is a vector in $\mathbb{R}^r$, and K_r, A_{cr}, and F_r are matrices of dimensions $m \times r$, $r \times r$, and $r \times p$, respectively. To simulate the performance of the low order compensator, we apply it to the original system (without the disturbance for simplicity) to obtain the closed loop system

$$\begin{bmatrix} \dot{x}(t) \\ \dot{x}_c^r(t) \end{bmatrix} = \begin{bmatrix} A & BK_r \\ F_r C & A_{cr} \end{bmatrix} \begin{bmatrix} x(t) \\ x_c^r(t) \end{bmatrix}, \tag{5}$$

with appropriate initial data. Simulating this system is one way to gain insight into the performance of the low order controller in regulating the PDE.

B. Balanced Model Reduction for PDEs

Both of the controller reduction methods studied in this paper use the standard balanced realization, coupled with truncation. Balancing was applied to finite dimensional systems in [12], [13] and to infinite dimensional systems in [14], [15]. It is typically used in the context of the state equations on the premise that a good low order approximation to the system can be obtained by eliminating any states that are difficult to control and to observe. In particular, the balancing transformation balances the Gramians for the state-space system. In LQG balanced realization, balancing is applied to the solutions of the control and filter Riccati equations. The method was established for systems of ordinary differential equations in [16], [17], [18]. Opmeer and Curtain have extended these results to PDE systems in [2], [3], [4].

We note that there is a restriction to the structure of the system that must be imposed in order to apply LQG balancing that is somewhat impractical in certain situations. Specifically, the measured output and controlled output operators must be identical, $C = E$, and the actuator input and disturbance operators must be identical, $B = D$. The authors are unaware of results that exist that remove those requirements, and that level of specificity of the character of input and outputs limits the applicability of the method.

IV. Snapshot Algorithms for Feedback Gains and Balanced Model Reduction

We now describe snapshot algorithms to compute feedback gains and balanced reduced order models for infinite dimensional systems over a separable Hilbert space X with inner product $(\cdot,\cdot)$ and corresponding norm $\|x\| = (x,x)^{1/2}$. We assume the inner product is real-valued for simplicity.

A. Snapshot Algorithms for Feedback Gains

To begin, we consider the computation of the feedback gain operator $K = B^*\Pi$, where $\Pi : X \to X$ is the solution of the algebraic Riccati equation (2). We assume the operators $B : \mathbb{R}^m \to X$ and $C : X \to \mathbb{R}^p$ are bounded and finite rank. The assumptions on B and C imply that the operators must take the form

$$Bu = \sum_{j=1}^{m} u_j b_j, \quad Cx = [\,(x, c_1), \ldots, (x, c_p)\,]^T,$$

for some vectors $b_1, \ldots, b_m$ and $c_1, \ldots, c_p$ in X (see [19, Theorem 6.1]). For simplicity we focus on the case of a single input and single output; i.e., $m = 1$ and $p = 1$; the algorithms are easily modified for $m > 1$ and $p > 1$. As with most large-scale algorithms for control and model reduction computations, the snapshot algorithms require m and p to be relatively small.

For the case $m = 1$, we have $Bu = bu$ where b is a vector in X. This assumption implies that the feedback operator $K : X \to \mathbb{R}$ given by $K = B^*\Pi$ has the representation $Kx = (x, k)$, where $k = \Pi b$ is a vector in X known as a functional gain. This representation holds since $B^*x = (x, b)$ and therefore $Kx = B^*\Pi x = (\Pi x, b) = (x, \Pi b)$, since Π is self-adjoint. Below, we concentrate on approximating this functional gain.

We first apply a Newton-Kleinman iteration as modified by Banks and Ito [20] to obtain a sequence of Lyapunov equations of the form

$$(A - BK_i)^* S_i + S_i(A - BK_i) + E_i^* E_i = 0, \qquad (6)$$

where K_i is the ith approximation to K, $E_0 x = [K_0 x, Cx]^T$ with K_0 the initial guess, and $E_i = K_i - K_{i-1}$ for $i \geq 1$. To advance to the next iteration, we need to compute $K_1 = B^* S_0$ and then $K_{i+1} = K_i - B^* S_i$ for $i \geq 1$. In the same manner as above, these operators can be represented as follows: $K_i x = (x, k_i)$, where $k_1 = S_0 b$ and $k_{i+1} = k_i - S_i b$ for $i \geq 1$. Therefore, in each iteration we do not need to compute the entire Lyapunov solution S_i, we only need the product $S_i b$. We compute this product using a snapshot algorithm below.

Consider a general infinite dimensional Lyapunov equation

$$A^* S + SA + C^* C = 0, \qquad (7)$$

where we assume $C : X \to \mathbb{R}$ is given by $Cx = (x, c)$ with $c \in X$. It is well known that the solution $S : X \to X$ is given by

$$Sx = \int_0^\infty e^{A^* t} C^* C e^{At} x \, dt.$$

Using the above representation of C, it can be shown [21], [22] that the solution may also be represented by

$$Sx = \int_0^\infty (x, z(t)) z(t) \, dt, \qquad (8)$$

where $z(t) = e^{A^* t} c$ is the solution of the infinite dimensional linear differential equation

$$\dot{z}(t) = A^* z(t), \quad z(0) = c. \qquad (9)$$

This representation leads to the following snapshot algorithm.

Snapshot algorithm [21], [22] to approximate Sx, where S solves the Lyapunov equation (7)

1) Compute an approximation $z^N(t)$ of the solution $z(t)$ of the differential equation (9).
2) Replace $z(t)$ with $z^N(t)$ in the integral representation of Sx in (8) and approximate the integral (by quadrature or some other method).

If $\int_0^\infty \|z^N(t) - z(t)\|^2 \, dt \to 0$, then the resulting approximation converges to Sx [22].

The approximate solution $z^N(t)$ of the differential equation (9) need not be stored to approximate Sx. Instead, a time stepping method can be used to approximate the differential equation and the approximation to the integral can be updated while simultaneously integrating the differential equation. For example, using a piecewise linear approximation to $z(t)$ in time leads to the trapezoid rule to time step the differential equation and the following approximation to the integral.

Trapezoid snapshot algorithm [22] to approximate Sx, where S solves the Lyapunov equation (7)

1) Approximate the solution of the differential equation (9) with the trapezoid rule:
$$(I - \Delta t A^*/2) z_{n+1} = (I + \Delta t A^*/2) z_n,$$
where I is the identity operator.
2) Update the approximation to Sx:
$$[Sx]_{n+1} = [Sx]_n + \Delta t[(x, z_{n+1})/3 + (x, z_n)/6] z_{n+1} + \Delta t[(x, z_{n+1})/6 + (x, z_n)/3] z_n.$$

This updating procedure can be stopped when the norm of the update to Sx (unscaled by Δt) is below a certain tolerance. We note that we used a constant time step for simplicity; this is not necessary in general.

For the Lyapunov equations arising in the modified Newton-Kleinman iterations (6), note that A^* in the Lyapunov equation (7) is replaced by $(A - BK_i)^*$. Thus, in the trapezoid snapshot algorithm, we must invert operators of the form $A_s - B_s K_s$, where $A_s = I - \Delta t A^*/2$, $B_s = -\Delta t K_i^*/2$ and $K_s = B^*$. To compute $(A_s - B_s K_s)^{-1} z$ we use the Sherman-Morrison-Woodbury formula (see, e.g., [23]):

$$(A - BK)^{-1} z = (I + A^{-1} B (I - K A^{-1} B)^{-1} K) A^{-1} z.$$

B. Snapshot Algorithms for Balanced Model Reduction

Next, we consider snapshot algorithms for constructing balanced reduced order models for infinite dimensional linear systems. We consider two related model reduction problems: standard Lyapunov balancing and LQG balancing. As mentioned above, we do not balance the uncontrolled linear system here; rather, we balance and truncate the compensator.

The snapshot algorithm for balanced model reduction of a system governed by ordinary differential equations was proposed by Rowley in [11]. We extended his algorithm to the class of linear infinite dimensional systems considered in this work in [24]. The snapshot algorithm for LQG balancing is an adaptation of Rowley's work, and it appears that it has not been proposed elsewhere.

For a linear system

$$\dot{x}(t) = Ax(t) + Bu(t), \quad y(t) = Cx(t), \tag{10}$$

balancing involves Gramians L_B and L_C, which are the solutions to the infinite dimensional Lyapunov equations

$$AL_B + L_B A^* + BB^* = 0, \quad A^* L_C + L_C A + C^* C = 0.$$

We again assume for simplicity that $Bu = bu$ and $Cx = (x, c)$ for vectors b and c in X. As above, the solutions to these Lyapunov equations are given by

$$L_B x = \int_0^\infty (x, w(t)) w(t)\, dt, \quad L_C x = \int_0^\infty (x, z(t)) z(t)\, dt,$$

where $w(t) = e^{At} b$ and $z(t) = e^{A^* t} c$ satisfy the differential equations

$$\dot{w}(t) = Aw(t),\, w(0) = b, \quad \dot{z}(t) = A^* z(t),\, z(0) = c. \tag{11}$$

In finite dimensions, the balanced realization can be computed using the eigenvalues and eigenvectors of the product of the Gramians $L_C L_B$. Rowley recognized that the eigendecomposition for the finite dimensional problem could be approximated using a variation of the proper orthogonal decomposition. In the infinite dimensional case, this can also be done as follows.

Approximate the Gramians with quadrature:

$$\begin{aligned} L_B x &\approx L_B^{n_1} = \sum_{j=1}^{n_1} \alpha_j^2 (x, w(t_j)) w(t_j) = \sum_{j=1}^{n_1} (x, \tilde{w}_j) \tilde{w}_j, \\ L_C x &\approx L_C^{n_2} = \sum_{k=1}^{n_2} \beta_k^2 (x, z(t_k)) z(t_k) = \sum_{k=1}^{n_2} (x, \tilde{z}_k) \tilde{z}_k, \end{aligned}$$

where $\{\alpha_j^2\}$ and $\{\beta_k^2\}$ are quadrature weights corresponding to the sets of quadrature points $\{t_j\}$ and $\{t_k\}$, $\tilde{w}_i = \alpha_i w(t_i)$, and $\tilde{z}_i = \beta_i z(t_i)$. The approximate Gramians can then be factored as $L_B^{n_1} = PP^*$ and $L_C^{n_2} = Q^* Q$, where the operators $P : \mathbb{R}^{n_1} \to X$ and $Q : X \to \mathbb{R}^{n_2}$ are defined by

$$Pa = \sum_{i=1}^{n_1} a_i \tilde{w}_i, \quad Qx = [\, (x, \tilde{z}_1), \, \ldots \,, (x, \tilde{z}_{n_2}) \,]^T.$$

The eigenvalues and eigenvectors of $L_C^{n_1} L_B^{n_2}$ can then be computed using the singular value decomposition of $\Gamma = QP$, which is an $n_2 \times n_1$ matrix of inner products of weighted snapshots with ij entries $\Gamma_{ij} = (\tilde{z}_i, \tilde{w}_j)$. For the remaining details of the algorithm, including the case of multiple inputs and outputs, see [24].

For LQG balancing, the procedure is similar although we must now solve Riccati equations instead of Lyapunov equations. In finite dimensions, the balancing transformation is given by the eigenvalues and eigenvectors of the product of the solutions Π and P of the Riccati equations (2) and (3), where we assume $B = D$ and $C = E$ as discussed above. As is well known, these Riccati equations can be rewritten as

$$\begin{aligned} (A - BK)^* \Pi + \Pi (A - BK) + K^* K + C^* C = 0, \\ (A - FC) P + P (A - FC)^* + FF^* + BB^* = 0, \end{aligned}$$

where $K = B^* \Pi$ and $F = PC^*$. Now we proceed as above with the representation of Π and P as the solution of these Lyapunov equations.

We again assume $Bu = bu$ and $Cx = (x, c)$, where b and c are vectors in X. Then $Kx = (x, k)$ and $Fy = fy$, where $k = \Pi b$ and $f = Pc$ are the functional gains. Once we compute k and f, we proceed as above and express Π and P in the form

$$\begin{aligned} \Pi x &= \int_0^\infty (x, z_1(t)) z_1(t) + (x, z_2(t)) z_2(t)\, dt, \\ Px &= \int_0^\infty (x, w_1(t)) w_1(t) + (x, w_2(t)) w_2(t)\, dt, \end{aligned}$$

where

$$\dot{z}_1(t) = (A - BK)^* z_1(t), \quad z_1(0) = k, \tag{12}$$

$$\dot{z}_2(t) = (A - BK)^* z_2(t), \quad z_1(0) = c, \tag{13}$$

$$\dot{w}_1(t) = (A - FC) w_1(t), \quad w_1(0) = f, \tag{14}$$

$$\dot{w}_2(t) = (A - FC) w_2(t), \quad w_1(0) = b. \tag{15}$$

Approximate the above integrals with quadrature:

$$\Pi x = \sum_{j=1}^{n_1} (x, \tilde{z}_j) \tilde{z}_j, \quad Px = \sum_{k=1}^{n_2} (x, \tilde{w}_k) \tilde{w}_k,$$

where now the "vectors" $\tilde{z}$ and $\tilde{w}$ contain weighted snapshots of the solutions of the differential equations (12) and (13), and (14) and (15), respectively.

Snapshot algorithm for LQG balanced model reduction of the linear system (10)

1) Approximate the feedback gains $K = B^* \Pi$ and $F = PC^*$, where Π and P solve the AREs (2) and (3), for example, using the snapshot algorithm outlined in Section IV-A.
2) Compute approximate solutions of the differential equations (12)-(15).
3) Form the matrix Γ, where $\Gamma_{ij} = (\tilde{z}_i, \tilde{w}_j)$ and the weighted snapshots $\tilde{w}_j$ and $\tilde{z}_i$ are as above.
4) Compute the singular value decomposition of Γ:

$$\Gamma = UMV^* = [U_1 \; U_2] \begin{bmatrix} M_1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} V_1^* \\ V_2^* \end{bmatrix} = U_1 M_1 V_1^*,$$

where $M_1 \in \mathbb{R}^{s \times s}$ is diagonal and invertible, $s = \operatorname{rank}(\Gamma)$, $U_1^* U_1 = I_s = V_1^* V_1$, and I_s is the identity matrix in $\mathbb{R}^{s \times s}$.
5) Choose $r < \operatorname{rank}(\Gamma)$, and form the first r primary and dual LQG balanced POD modes defined by

$$\begin{aligned} [\, \varphi_1, \, \ldots \,, \varphi_r \,]^T &= M_r^{-1/2} V_r^* \tilde{w}, \\ [\, \psi_1, \, \ldots \,, \psi_r \,]^T &= M_r^{-1/2} U_r^* \tilde{z}, \end{aligned}$$

where M_r, U_r, and V_r are truncations of M_1, U_1, and V_1.

6) Use the modes to form the matrices in the reduced order model:

$$\begin{aligned} A_r &= [\,(A\varphi_j, \psi_i)\,] \in \mathbb{R}^{r\times r}, \\ B_r &= [\,(b, \psi_i)\,] \in \mathbb{R}^{r\times 1}, \\ C_r &= [\,(\varphi_j, c)\,] \in \mathbb{R}^{1\times r}, \end{aligned}$$

It is straightforward to extend this algorithm to multiple inputs and outputs. Convergence theory for the balancing algorithms is underway and will be the subject of future work.

V. Numerical Results

For our numerical experiments with the model problem outlined above, we chose $\mu = 0.05$, convection coefficients $c_1(x, y) = -x\sin(2\pi x)\sin(\pi y)$ and $c_2(x, y) = -y\sin(\pi x)\sin(2\pi y)$, control input function $b(x, y) = 5\sin(\pi x)\sin(\pi y)$ if $x \geq 1/2$ and $b(x, y) = 0$ otherwise, observation function $c(x, y) \equiv 5$, and initial condition $w_0(x, y) = 5\cos(\pi x/2)\sin(\pi y)$.

For the snapshot algorithms, we used standard piecewise linear finite elements for the spatial discretization. For the functional gain computations, we first computed functional gains for $\mu = 0.1$ (using a Newton iteration with $K_0 = 0$) and then used these as initial guesses in the Newton iteration for $\mu = 0.05$. In a similar fashion, we note that one could use the result of one Newton iteration as the initial guess in another Newton iteration with a finer spatial grid (see, e.g., [25]) or time step for the snapshot algorithm. We used the trapezoid rule for the time discretizations required in the snapshot balancing algorithms.

Figures 1 and 2 show approximations to the functional gains $k(x, y)$ and $f(x, y)$ computed using the modified Newton algorithm with the trapezoid snapshot algorithm. We used $\Delta t = 0.01$ for the time step and 41 equally spaced finite element nodes in each coordinate direction. Further refinement in space and time produced little change in the approximations. We note that it may be desirable to use an adaptive time stepping algorithm and this will be explored in future work.

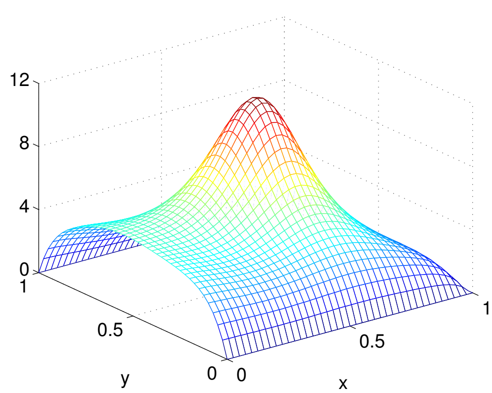

Fig. 1. Approximate control functional gain $k(x, y)$ computed using the snapshot algorithm with $\Delta t = 0.01$ and 41 equally spaced finite element nodes in each coordinate direction.

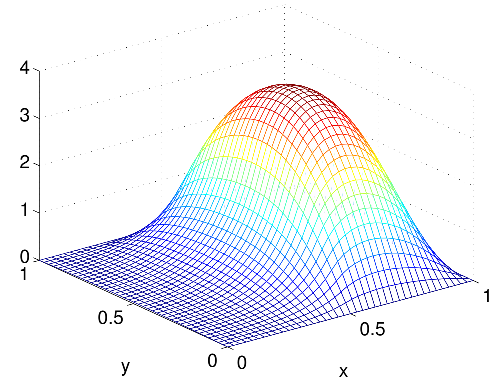

Fig. 2. Approximate observation functional gain $f(x, y)$ computed using the snapshot algorithm with $\Delta t = 0.01$ and 41 equally spaced finite element nodes in each coordinate direction.

Figure 3 shows approximations to the Hankel singular values of the compensator system computed using the balancing snapshot algorithm with $\Delta t = 0.01$ and various equally spaced finite element grids. Many of the larger Hankel singular values are essentially converged on the coarse 21×21 node grid. Further refinement in space caused the smaller values to converge, however this is not needed since these values are not required to reduce the compensator. Also, further refinement in time produced little change. Figure 4 shows similar approximations to the LQG characteristic values of the system computed using the LQG balancing snapshot algorithm. The LQG values do not converge as quickly and further refinement in space and time produced change in most of the values. However, the larger values are converged and only these are used to construct the reduced order compensator. Again, adaptive time stepping algorithms may be advantageous to use for these computations.

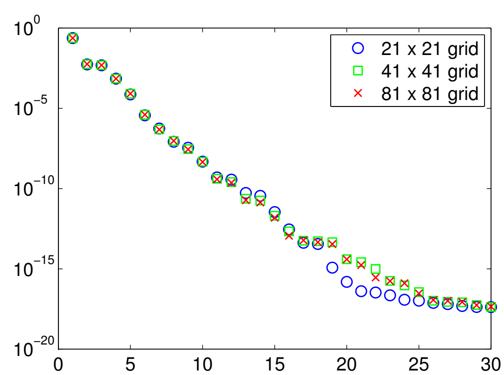

Fig. 3. Approximate compensator Hankel singular values computed using the snapshot algorithm with $\Delta t = 0.01$ and 21, 41, and 81 equally spaced finite element nodes in each coordinate direction.

To compare the two approaches to reducing the compensator, we computed the L^2 norm of the solution of the controlled system. The uncontrolled system is stable, however the solution tends to zero very slowly. The norm of the uncontrolled solution at $t = 20$ is approximately 0.1.

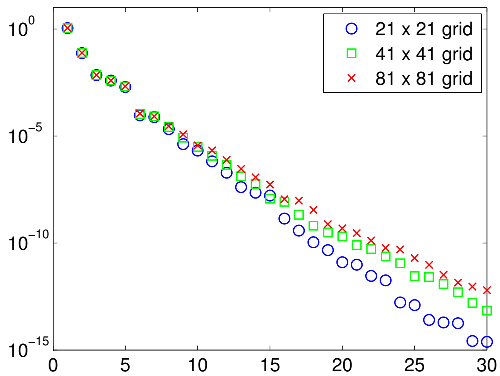

Fig. 4. Approximate LQG characteristic values computed using the snapshot algorithm with $\Delta t = 0.01$ and 21, 41, and 81 equally spaced finite element nodes in each coordinate direction.

Both reduced order controllers (with zero initial data for the compensators) drive the solution to zero at a much faster rate. We chose $r = 4$ states in each reduced order compensator and found that the norm of the solution at $t = 5$ of each closed loop system is on the order of 10^{-4}. Integrating longer in time showed that the controller constructed using LQG balancing drove the solution to zero slightly faster than the controller constructed by balancing the compensator.

Although the performance of the two controllers was very similar, we note that the LQG balanced reduced model was more computationally demanding to construct, and it also converged at a slower rate. This is likely due to the fact that the snapshot algorithm required approximate solutions of the differential equations (13) and (15), whose initial conditions ($c(x, y)$ and $b(x, y)$) are not "smooth" in the sense that they are not twice differentiable functions satisfying the boundary conditions of the governing PDE. Computing the reduced order controllers is done "offline" and so computation time may not necessarily be an issue. However, accurately computing the reduced controller may be essential to guarantee controller performance. Therefore, it appears more care may be required to construct the LQG balanced reduced controller. This may be especially true of more complex problems.

VI. Conclusions and Future Work

We believe that model reduction based on these approaches following robust control design as demonstrated in [2] holds much promise. Both reduced controllers performed well on the model problem. A simple comparison showed the controllers gave similar performance; further investigation is required to give a more thorough comparison of the reduced controllers. Other future work includes further development and analysis of the snapshot algorithms for the controller construction, and comparison of other types of reduced controllers for PDE systems.

References

[1] A. Yousuff and R. E. Skelton, "A note on balanced controller reduction," *IEEE Trans. Automat. Control*, vol. 29, no. 3, pp. 254–257, 1984.

[2] R. F. Curtain, "Model reduction for control design for distributed parameter systems," in *Research Directions in Distributed Parameter Systems*. Philadelphia, PA: SIAM, 2003, pp. 95–121.

[3] M. R. Opmeer and R. F. Curtain, "Linear quadratic Gaussian balancing for discrete-time infinite-dimensional linear systems," *SIAM J. Control Optim.*, vol. 43, no. 4, pp. 1196–1221, 2004.

[4] M. R. Opmeer, "LQG balancing for continuous-time infinite-dimensional systems," *SIAM J. Control Optim.*, vol. 46, no. 5, pp. 1831–1848 (electronic), 2007.

[5] A. C. Or and J. L. Speyer, "Gain-scheduled controller for the suppression of convection at high Rayleigh number," *Physical Review E*, vol. 71, p. 046302, 2005.

[6] K. Evans, *Reduced Order Controllers for Distributed Parameter Systems*. Ph.D. Dissertation, Virginia Polytechnic Institute & State University, 2003.

[7] B. King, N. Hovakimyan, K. Evans, and M. Buhl, "Reduced Order Controllers for Distributed Parameter Systems: LQG Balancing and an Adaptive Approach," *Special Issue of Mathematical and Computer Modeling dedicated to the 50th anniversary of AFOSR*, vol. 43, no. 9-10, pp. 1136–1149, 2006.

[8] A. C. Antoulas, *Approximation of Large-Scale Dynamical Systems*. Philadelphia, PA: SIAM, 2005.

[9] P. Benner, V. Mehrmann, and D. C. Sorensen, Eds., *Dimension Reduction of Large-Scale Systems*. Berlin: Springer-Verlag, 2005.

[10] K. Willcox and J. Peraire, "Balanced model reduction via the proper orthogonal decomposition," *AIAA Journal*, vol. 40, no. 11, pp. 2323–2330, 2002.

[11] C. W. Rowley, "Model reduction for fluids, using balanced proper orthogonal decomposition," *Internat. J. Bifur. Chaos Appl. Sci. Engrg.*, vol. 15, no. 3, pp. 997–1013, 2005.

[12] B. C. Moore, "Principal component analysis in linear systems: controllability, observability, and model reduction," *IEEE Trans. Automat. Control*, vol. 26, no. 1, pp. 17–32, 1981.

[13] L. Pernebo and L. M. Silverman, "Model reduction via balanced state space representations," *IEEE Trans. Automat. Control*, vol. 27, no. 2, pp. 382–387, 1982.

[14] R. F. Curtain and K. Glover, "Balanced realisations for infinite-dimensional systems," in *Operator Theory and Systems (Amsterdam, 1985)*. Basel: Birkhäuser, 1986, pp. 87–104.

[15] K. Glover, R. F. Curtain, and J. R. Partington, "Realization and approximation of linear infinite-dimensional systems with error bounds," *SIAM J. Control Optim.*, vol. 26, pp. 863–898, 1988.

[16] E. A. Jonckheere and L. M. Silverman, "A new set of invariants for linear systems—application to reduced order compensator design," *IEEE Trans. Automat. Control*, vol. 28, pp. 953–964, 1983.

[17] E. I. Verriest, "Low sensitivity design and optimal order reduction for the LQG-problem," *24th Midwest Symposium on Circuits and Systems*, pp. 365–369, 1981.

[18] ——, "Suboptimal LQG-design via balanced realizations," *20th IEEE Conference on Decision and Control*, pp. 686–687, 1981.

[19] J. Weidmann, *Linear Operators in Hilbert Spaces*. New York: Springer-Verlag, 1980.

[20] H. T. Banks and K. Ito, "A numerical algorithm for optimal feedback gains in high-dimensional linear quadratic regulator problems," *SIAM J. Control Optim.*, vol. 29, no. 3, pp. 499–515, 1991.

[21] J. R. Singler, "Approximate low rank solutions of Lyapunov equations via proper orthogonal decomposition," in *Proceedings of the 2008 American Control Conference*, 2008, pp. 267–272.

[22] ——, "Convergent snapshot algorithms for infinite dimensional Lyapunov equations," submitted.

[23] G. H. Golub and C. F. Van Loan, *Matrix Computations*. Baltimore, MD: Johns Hopkins University Press, 1996.

[24] J. R. Singler and B. A. Batten, "A proper orthogonal decomposition approach to approximate balanced truncation of infinite dimensional linear systems," *Int. J. Comput. Math.*, vol. 86, no. 2, pp. 355–371, 2009.

[25] K. Morris and C. Navasca, "Solution of algebraic Riccati equations arising in control of partial differential equations," in *Control and Boundary Analysis*. Chapman & Hall/CRC, Boca Raton, FL, 2005, pp. 257–280.