Missouri University of Science and Technology

Scholars' Mine

Mathematics and Statistics Faculty Research & Creative Works

Mathematics and Statistics

01 Feb 2009

A Proper Orthogonal Decomposition Approach to Approximate Balanced Truncation of Infinite Dimensional Linear Systems

John R. Singler
Missouri University of Science and Technology, singlerj@mst.edu

Belinda A. Batten

Recommended Citation

A Proper Orthogonal Decomposition Approach to Approximate Balanced Truncation of Infinite Dimensional Linear Systems (SI-CMSSE 07)

John R. Singler and Belinda A. Batten

We extend a method for approximate balanced reduced order model derivation for finite dimensional linear systems developed by Rowley (Internat. J. Bifur. Chaos Appl. Sci. Engrg., 15(3), 997–1013) to infinite dimensional systems. The algorithm is related to standard balanced truncation, but includes aspects of the proper orthogonal decomposition in its computational approach. The method can be also applied to nonlinear systems. Numerical results are presented for a convection diffusion system.

Keywords: balanced truncation, proper orthogonal decomposition, infinite dimensional systems

1.   Introduction and Overview

A large body of research over the last ten to fifteen years has been devoted to the development of reduced order models for a variety of uses including simulation, optimization, optimal design, and controller derivation. A technique that came from the systems literature is balanced realization and truncation, and is used regularly with a good deal of success. This approach involves retaining the states of the system that are both most controllable and most observable. The other states of the system are truncated and a reduced model is formed. For an overview of the method and details of the standard algorithm, see e.g., [1, 2]. Computing the balanced reduced order model for large scale systems, such as those arising from a discretization of an infinite dimensional system, is an active area of research (see [3, 4] and the references therein).

In [5], an approach to computing the balanced realization was presented that uses the proper orthogonal decomposition. The technique was developed for finite dimensional systems. In this work, we extend Rowley's balanced POD algorithm to a class of infinite dimensional systems. One of the main features of the algorithm is that it does not require matrix approximations of the infinite dimensional operators. Such matrices can be difficult or impossible to obtain from certain simulation codes or for certain problems (such as linearized fluid flow). Although we do not attempt to provide a complete convergence analysis of the algorithm in this work, we present numerical results for a model problem that indicate that the algorithm is convergent. Our derivation of the algorithm also points to the analysis that remains in order to prove convergence. We also note how the present algorithm can differ from the popular method of first discretizing the infinite dimensional system and then balancing the resulting finite dimensional system. In particular, we demonstrate by way of example that the latter method can lead to an incorrect reduced

order model if the finite dimensional algorithm is applied without consideration of the underlying approximation of the infinite dimensional system.

The algorithm proposed in this work is a POD-type procedure to design an approximate balanced transformation of an infinite dimensional linear system

$$\dot{x}(t) = Ax(t) + Bu(t), \quad x(0) = x_0, \tag{1a}$$

$$y(t) = Cx(t), \tag{1b}$$

over a Hilbert space X with inner product $(\cdot, \cdot)$. We assume the linear operator $A : D(A) \subset X \to X$ generates an exponentially stable C_0-semigroup e^{At}, and the operators $B : U \to X$ and $C : X \to Y$ are bounded and finite rank. We also assume the input and output spaces are finite dimensional; specifically $U = \mathbb{R}^m$ and $Y = \mathbb{R}^p$.

The general goal of model reduction is to construct a low order model of the form

$$\dot{a}(t) = A_r a(t) + B_r u(t), \quad a(0) = a_0, \tag{2a}$$

$$y(t) = C_r a(t), \tag{2b}$$

that is a "good approximation" to the original system (1) in some sense. Here, the system holds over the state space $\mathbb{R}^r$, where the objective is to have the order r small. The input and output spaces are the same as in (1). One way to measure the distance between the two systems is to compare their transfer functions, $G(s) = C(sI - A)^{-1}B$ and $G_r(s) = C_r(sI_r - A_r)^{-1}B_r$, respectively. Therefore, we look for a system (A_r, B_r, C_r) that makes the error $\|G - G_r\|_\infty$ small. Here, the applied norm is the $\mathcal{H}^\infty$ norm which is the largest singular value of the transfer function along the imaginary axis.

Model reduction via balanced truncation is performed by first determining a realization of the system in which the controllable and observable states of (1) coincide. Specifically, define the controllability and observability operators $\mathcal{B} : L^2(0, \infty; U) \to X$ and $\mathcal{C} : X \to L^2(0, \infty; Y)$ by

$$\mathcal{B}u = \int_0^\infty e^{At} Bu(t)\, dt, \quad [\mathcal{C}x](t) = Ce^{At}x.$$

The adjoint operators $\mathcal{B}^* : X \to L^2(0, \infty; U)$ and $\mathcal{C}^* : L^2(0, \infty; Y) \to X$ are given by

$$[\mathcal{B}^*x](t) = B^*e^{A^*t}, \quad \mathcal{C}^*y = \int_0^\infty e^{A^*t}C^*y(t)\, dt.$$

The controllability and observability Gramians, $L_B \in \mathcal{L}(X)$ and $L_C \in \mathcal{L}(X)$, are defined by

$$L_B x = \mathcal{B}\mathcal{B}^*x = \int_0^\infty e^{At}BB^*e^{A^*t}x\, dt, \quad L_C x = \mathcal{C}^*\mathcal{C}x = \int_0^\infty e^{A^*t}C^*Ce^{At}x\, dt.$$

Balancing provides a new system (A^b, B^b, C^b) that yields the same transfer function $G(s)$ as the original system and for which the Gramians, L_B^b and L_C^b, are equal and diagonal.

The magnitude of the diagonal entries of the balanced Gramians can be thought of as a measure of the controllability and observability of the states. These diagonal

entries are given by the singular values of the Hankel operator $\mathcal{H} : L^2(0, \infty; U) \to L^2(0, \infty; Y)$ defined by

$$[\mathcal{H}u](t) = [\mathcal{C}\mathcal{B}u](t) = \int_0^\infty Ce^{A(t+s)} Bu(s) \, ds.$$

An important fact is that the Hankel singular values are independent of the system realization. Therefore, once the Hankel singular values are ordered from greatest to least, the states corresponding to the "small" singular values are truncated to produce a low order model.

An important property of this method is that we have the following error bound:

$$\|G - G_r\|_\infty \leq 2 \sum_{k>r} \sigma_k, \tag{3}$$

where σ_k is the kth (ordered) Hankel singular value and G_r is the transfer function of the rth order truncated balanced system. Therefore, if the Hankel singular values decay quickly, the error will be small.

In the infinite dimensional setting, the balanced realization exists under certain conditions; in the case that it does exist, the Gramians are equal to a diagonal operator on ℓ^2, the space of square summable sequences (see [6, 7] and the review in [8]). For the theory to hold, the Hankel operator must be nuclear, i.e., the infinite sum of the Hankel singular values must be finite. Of course, this condition is necessary to guarantee that the right hand side of the error bound (3) is finite. It is known that the Hankel operator is nuclear for the class of systems studied in this paper [9, Theorem 4].

The balanced POD algorithm determines a truncated approximate balancing transformation $T_r : \mathbb{R}^r \to X$ and its left inverse $S_r : X \to \mathbb{R}^r$ (i.e., $S_r T_r = I_r$). To obtain a low order model, approximate the solution $x(t)$ of the linear system (1) by Galerkin projection as

$$x(t) \approx x_r(t) = T_r S_r x(t) = T_r a(t), \quad \text{where} \quad a(t) = S_r x(t). \tag{4}$$

Substituting this approximate solution into the linear system yields the reduced order model (2), where $A_r = S_r A T_r$, $B_r = S_r B$, $C_r = C T_r$, and $a_0 = S_r x_0$.

We may apply this Galerkin projection to obtain low order models of more general, in fact nonlinear, systems. For example, suppose the model takes the form

$$\dot{x}(t) = Ax(t) + F(x(t)) + Bu(t) + Dw(t), \quad x(0) = x_0, \tag{5a}$$

$$y(t) = Cx(t) + Ew(t), \tag{5b}$$

where F is a nonlinear operator and w is a disturbance. One can derive the approximate balancing transformation about the linearized system and use the approximation for the solution (4) to obtain the model

$$\dot{a}(t) = A_r a(t) + F_r(a(t)) + B_r u(t) + D_r w(t), \quad a(0) = a_0, \tag{6a}$$

$$y(t) = C_r a(t) + E_r w(t), \tag{6b}$$

where A_r, B_r, C_r, and a_0 are as above, $D_r = S_r D$, $E_r = E$, and $F_r(a) = S_r F(T_r a)$.

2.   Derivation of the Algorithm

We now give a derivation of the balanced POD algorithm for the infinite dimensional setting described above. We do not attempt to rigorously prove every step the derivation; in some cases we simply proceed by analogy with the finite dimensional case. However, we point out places where rigorous convergence analysis of the algorithm is needed; this is an important first step in showing what can be done. Although the formal derivation may seem unnecessary, we show in Section 4.4 that naively applying the finite dimensional approach can lead to incorrect results.

The complete algorithm is presented in Section 3 below. One possible numerical implementation of the algorithm is given in Section 3.1.

2.1.   *Special Forms of the Gramians*

One of the main components of the balanced POD algorithm is to compute approximate factors of the Gramians using simulation data. This is possible because of the special form of the Gramians.

Given the specific assumptions regarding the input and output operators, B and C, in Section 1, they must have the form (see [10, Theorem 6.1])

$$Bu = \sum_{j=1}^{m} b_j u_j, \quad Cx = [\,(c_1, x), \,\ldots, (c_p, x)\,]^T, \tag{7}$$

where $u = [\,u_1, \ldots, u_m\,]^T \in U$, and each b_j and c_j are in X. This allows us to rewrite the Gramians. First, define the functions $w_j(t) = e^{At} b_j$, for $j = 1, \ldots, m$. Then w_j is the solution of the evolution equation

$$\dot{w}_j(t) = A w_j(t), \quad w_j(0) = b_j.$$

The controllability operator $\mathcal{B} : L^2(0, \infty; U) \to X$ defined above takes the form

$$\mathcal{B}u = \int_0^\infty e^{At} Bu(t)\, dt = \int_0^\infty \sum_{j=1}^{m} w_j(t) u_j(t)\, dt,$$

and its adjoint operator $\mathcal{B}^* : X \to L^2(0, \infty; U)$ is easily computed to be

$$[\mathcal{B}^* x](t) = [\,(w_1(t), x), \,\ldots, (w_m(t), x)\,]^T.$$

Therefore, the controllability Gramian $L_B = \mathcal{B}\mathcal{B}^* \in \mathcal{L}(X)$ is given by

$$L_B x = \int_0^\infty \sum_{j=1}^{m} w_j(t)(w_j(t), x)\, dt.$$

To treat the observability Gramian, we need the adjoint operator $C^* \in \mathcal{L}(Y, X)$ given by

$$C^* y = \sum_{j=1}^{p} c_j y_j,$$

where $y = [\, y_1, \ldots, y_p \,]^T \in Y$. We follow a similar procedure as used for B and define $z_j(t) = e^{A^*t} c_j$, for $j = 1, \ldots, p$. Then z_j is the solution of the adjoint equation

$$\dot{z}_j(t) = A^* z_j(t), \quad z_j(0) = c_j.$$

The adjoint of the observability operator $\mathcal{C}^* : L^2(0, \infty; Y) \to X$ takes the form

$$\mathcal{C}^* y = \int_0^\infty e^{A^*t} C^* y(t)\, dt = \int_0^\infty \sum_{j=1}^p z_j(t) y_j(t)\, dt$$

and the operator $\mathcal{C} : X \to L^2(0, \infty; Y)$ is given by $[\mathcal{C}x](t) = [\, (z_1(t), x), \ldots, (z_p(t), x) \,]^T$. Therefore, the observability Gramian $L_C = \mathcal{C}^*\mathcal{C} \in \mathcal{L}(X)$ is

$$L_C x = \int_0^\infty \sum_{j=1}^p z_j(t)(z_j(t), x)\, dt.$$

2.2.   *The Empirical Gramians*

The Gramians can be approximated using time snapshots of the states $w_i(t)$ and $z_i(t)$. Specifically, we approximate the time integrals with the quadratures

$$L_B x = \int_0^\infty \sum_{i=1}^m w_i(t)(w_i(t), x)\, dt \approx L_B^{n_1} x = \sum_{i=1}^m \sum_{j=1}^{n_1} \alpha_j^2 w_i(t_j)(w_i(t_j), x),$$

$$L_C x = \int_0^\infty \sum_{i=1}^p z_i(t)(z_i(t), x)\, dt \approx L_C^{n_2} x = \sum_{i=1}^p \sum_{k=1}^{n_2} \beta_k^2 z_i(t_k)(z_i(t_k), x).$$

Here, $\{\alpha_j^2\}$ and $\{\beta_k^2\}$ are quadrature weights corresponding to the sets of quadrature points $\{t_j\}$ and $\{t_k\}$; different quadrature points and weights can be used for each w_i and z_i if desired. Since w_i are z_i are solutions to linear evolution equations, they are continuous in time and therefore have a well defined value at the quadrature points. The approximate Gramians $L_B^{n_1} \in \mathcal{L}(X)$ and $L_C^{n_2} \in \mathcal{L}(X)$ are called *empirical Gramians*.

Following Rowley in the finite dimensional case, we factor the empirical Gramians. Define "vectors" of weighted snapshots

$$\tilde{w} = [\, \alpha_1 w_1(t_1), \ldots, \alpha_{n_1} w_1(t_{n_1}), \ldots, \alpha_1 w_m(t_1), \ldots, \alpha_{n_1} w_m(t_{n_1}) \,]^T \in X^{N_1}, \quad (8)$$

$$\tilde{z} = [\, \beta_1 z_1(t_1), \ldots, \beta_{n_2} z_1(t_{n_2}), \ldots, \beta_1 z_p(t_1), \ldots, \beta_{n_2} z_p(t_{n_2}) \,]^T \in X^{N_2}, \quad (9)$$

where $N_1 = mn_1$, $N_2 = pn_2$, and $X^q = X \times \cdots \times X$ (q times). These vectors allow the empirical Gramians to be written 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

$$Pa = \sum_{i=1}^{N_1} a_i \tilde{w}_i, \quad Qx = [\, (\tilde{z}_1, x), \ldots, (\tilde{z}_{N_2}, x) \,]^T,$$

and their adjoint operators $P^* : X \to \mathbb{R}^{N_1}$ and $Q^* : \mathbb{R}^{N_2} \to X$ are given by

$$P^*x = [\,(\tilde{w}_1, x), \,\dots\,, (\tilde{w}_{N_1}, x)\,]^T, \quad Q^*a = \sum_{i=1}^{N_2} a_i \tilde{z}_i.$$

Note that P and Q and their adjoints depend on the quadrature points and weights; however, we suppress this dependence for notational simplicity. Note that if the data is smooth, then in fact $P : \mathbb{R}^{N_1} \to D(A)$ and $Q^* : \mathbb{R}^{N_2} \to D(A^*)$. This will allow us to say more about the reduced order model that results.

2.3. *The Approximate Balanced Transformation*

In the finite dimensional case, the eigendecomposition of the product of the Gramians can be used to compute a balancing transformation for the linear system. The balanced system is then truncated to form the reduced order model. We approximate the product of the Gramians $L = L_C L_B$ using the empirical Gramians, i.e., $L \approx L^n = L_C^{n_2} L_B^{n_1}$. Using the above factors, we have $L^n = Q^*QPP^*$. Following Curtain and Zwart [11, Lemma 8.2.9], it is easy to show that L^n is compact and that the nonzero eigenvalues of L^n are equal to the squares of the nonzero singular values of QP.

The operator QP is a bounded linear mapping from $\mathbb{R}^{N_1}$ to $\mathbb{R}^{N_2}$; therefore, it can be represented as an $N_2 \times N_1$ matrix Γ with entries $\Gamma_{ij} = (\tilde{z}_i, \tilde{w}_j)$. Let the singular value decomposition of Γ be given by

$$\Gamma = U\Sigma V^* = [U_1 \ U_2] \begin{bmatrix} \Sigma_1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} V_1^* \\ V_2^* \end{bmatrix} = U_1 \Sigma_1 V_1^*, \tag{10}$$

where $\Sigma_1 \in \mathbb{R}^{s \times s}$ is diagonal and invertible, $s = \text{rank}(\Gamma)$, $U_1^* U_1 = I_s = V_1^* V_1$, and I_s is the identity matrix in $\mathbb{R}^{s \times s}$.

In the finite dimensional case, Rowley showed that an approximate balancing transformation is given by the operators $T_1 : \mathbb{R}^s \to X$ and $S_1 : X \to \mathbb{R}^s$ defined by

$$T_1 = PV_1\Sigma_1^{-1/2}, \quad S_1 = \Sigma_1^{-1/2} U_1^* Q.$$

In this paper, we assume the same is true for the infinite dimensional setting and leave theoretical analysis of the algorithm for future work.

The operators $T_1 : \mathbb{R}^s \to X$ and $S_1 : X \to \mathbb{R}^s$ have the representations

$$T_1 a = \sum_{j=1}^{s} a_j \varphi_j, \quad S_1 x = [\,(\psi_1, x), \,\dots\,, (\psi_s, x)\,]^T,$$

where the (primary) balanced POD modes $\{\varphi_i\}$ and the adjoint balanced POD modes $\{\psi_i\}$ are given by

$$[\,\varphi_1, \,\dots\,, \varphi_s\,]^T = \Sigma_1^{-1/2} V_1^* \tilde{w}, \qquad [\,\psi_1, \,\dots\,, \psi_s\,]^T = \Sigma_1^{-1/2} U_1^* \tilde{z}.$$

As in the finite dimensional case, the primary and adjoint balanced POD modes are biorthogonal, i.e., $(\psi_i, \varphi_j) = \delta_{ij}$. To see this, note $S_1 T_1 a = [\,(\psi_i, \varphi_j)\,]a$ for any $a \in \mathbb{R}^s$. Also, by definition,

$$S_1 T_1 a = \Sigma_1^{-1/2} U_1^* QPV_1 \Sigma_1^{-1/2} a = I_s a.$$

Thus, $[(\psi_i, \varphi_j)] = I_s$, or $(\psi_i, \varphi_j) = \delta_{ij}$. Note that the balanced POD modes have the same smoothness as the solution data (i.e., snapshots).

The approximate balancing transformations are truncated by selecting $r < s$ and setting

$$T_r a = \sum_{j=1}^{r} a_j \varphi_j, \quad S_r x = [\,(\psi_1, x), \, \ldots \,, (\psi_r, x)\,]^T.$$

Thus, only the first r primary and adjoint balanced POD modes need to be computed. Also, we have $S_r T_r = I_r$, and the modes can be computed by

$$[\,\varphi_1, \, \ldots \,, \varphi_r\,]^T = \Sigma_r^{-1/2} V_r^* \tilde{w}, \qquad [\,\psi_1, \, \ldots \,, \psi_r\,]^T = \Sigma_r^{-1/2} U_r^* \tilde{z}, \tag{11}$$

where Σ_r, U_r, and V_r are appropriate truncations of Σ_1, U_1, and V_1.

3.   The Balanced POD Algorithm

The construction of the operators T_r and S_r as shown above completes the balanced POD algorithm. As outlined in Section 1, we use a transformation to obtain the reduced order model (2). The complete procedure can be summarized as follows:

(i) Approximate the solutions w_j of the differential equations

$$\dot{w}_j(t) = A w_j(t), \quad w_j(0) = b_j, \tag{12}$$

for $j = 1, \ldots, m$, where $Bu = \sum_{j=1}^{m} b_j u_j$.

(ii) Approximate the solutions z_j of the adjoint differential equations

$$\dot{z}_j(t) = A^* z_j(t), \quad z_j(0) = c_j, \tag{13}$$

for $j = 1, \ldots, p$, where $Cx = [\,(c_1, x), \, \ldots \,, (c_p, x)\,]^T$.

(iii) Form the matrix Γ, where $\Gamma_{ij} = (\tilde{z}_i, \tilde{w}_j)$, and the weighted snapshot vectors $\tilde{w}$ and $\tilde{z}$ defined in (8) and (9), respectively.

(iv) Compute the singular value decomposition of Γ as in (10), choose $r < \text{rank}(\Gamma)$, and form the first r primary and adjoint balanced POD modes defined in (11):

$$[\,\varphi_1, \, \ldots \,, \varphi_r\,]^T = \Sigma_r^{-1/2} V_r^* \tilde{w}, \qquad [\,\psi_1, \, \ldots \,, \psi_r\,]^T = \Sigma_r^{-1/2} U_r^* \tilde{z},$$

where Σ_r, U_r, and V_r are appropriate truncations of Σ_1, U_1, and V_1.

(v) Use the modes to form the matrices in the reduced order model (2):

$$\begin{aligned}
A_r &= S_r A T_r = [\,(A\varphi_j, \psi_i)\,] \in \mathbb{R}^{r \times r}, \\
B_r &= \ S_r B \ = [\,(b_j, \psi_i)\,] \in \mathbb{R}^{r \times m}, \\
C_r &= \ C T_r \ = [\,(\varphi_j, c_i)\,] \in \mathbb{R}^{p \times r}, \\
a_0 &= \ S_r x_0 \ = [\,(x_0, \psi_1), \, \ldots \,, (x_0, \psi_r)\,]^T \in \mathbb{R}^r.
\end{aligned} \tag{14}$$

The smoothness of the balanced POD modes is important in insuring that A_r, B_r, C_r are well-defined, and hence the reduced order model is also well-defined. If T_r does not map into the domain of A, for example, the operator A_r is nonsensical. Although we do not present all the details here, we conjecture that S_r^* must map

into $D(A^*)$ for convergence of the algorithm. Formal manipulation of the approximations of the infinite dimensional operators can lead to a model that is ill-defined, and numerical results that are meaningless.

As in balancing for finite dimensional systems, one may choose the order r of the reduced system so that the balancing error bound (3) is small enough. Of course, balanced POD will not produce the exact truncated balanced reduced order model; therefore, we cannot expect the balancing error bound to hold. However, the balanced POD reduced order model can be refined in two ways. First, the discretizations of the infinite dimensional differential equations (12) and (13) can be refined to produce more accurate time snapshots. Secondly, as with finite dimensional balanced POD, one may refine the discretization in time by taking a larger number of snapshots of the differential equations. These refinements will affect the resulting Hankel singular values, balancing error bound (3), and balancing modes. As these quantities converge, we anticipate that the balanced POD reduced order model will converge to the exact truncated balanced system. Thus, it is possible that the transfer function error will be close to the balancing error bound (3). As mentioned earlier, we leave this convergence analysis for future work.

3.1.   *Finite Dimensional Galerkin Approximations*

The algorithm presented above is flexible since we may use any procedure to approximate the solutions w_i and z_i of the linear differential equations (12) and (13). We describe the balanced POD algorithm with Galerkin approximations.

Let $W_1 = \text{span}\{\xi_j\}_{j=1}^k \subset D(A)$ and $W_2 = \text{span}\{\eta_j\}_{j=1}^\ell \subset D(A^*)$ be finite dimensional subsets of X. We compute the solutions of the primary and adjoint differential equations by the finite dimensional Galerkin approximations

$$w_\alpha(t) \approx \sum_{j=1}^k r_{j\alpha}(t)\xi_j, \quad z_\beta(t) \approx \sum_{j=1}^\ell s_{j\beta}(t)\eta_j,$$

for $\alpha = 1, \ldots, m$ and $\beta = 1, \ldots, p$. Here, k is the same for each α and ℓ is the same for each β; this is not necessary in general, but it does simplify the resulting algorithm. Using these Galerkin approximations, the balanced POD algorithm takes the following form:

(i) Form the $k \times k$ matrices $\tilde{M}_k = [\,(\xi_j, \xi_i)\,]$ and $\tilde{A}_k = [\,(A\xi_j, \xi_i)\,]$. Approximate the Galerkin coefficient vectors $r_\alpha = [\,r_{1\alpha},\, \ldots,\, r_{m\alpha}\,]^T$ by solving the equations

$$\tilde{M}_k \dot{r}_\alpha(t) = \tilde{A}_k r_\alpha(t), \quad \tilde{M}_k r_\alpha(0) = [\,(b_\alpha, \xi_i)\,], \qquad \alpha = 1, \ldots, m. \qquad (15)$$

(ii) Form the $\ell \times \ell$ matrices $\hat{M}_\ell = [\,(\eta_j, \eta_i)\,]$ and $\hat{A}_\ell = [\,(A^*\eta_j, \eta_i)\,]$. Approximate the Galerkin coefficient vectors $s_\beta = [\,s_{1\beta},\, \ldots,\, s_{p\beta}\,]^T$ by solving the equations

$$\hat{M}_\ell \dot{s}_\beta(t) = \hat{A}_\ell s_\beta(t), \quad \hat{M}_\ell s_\beta(0) = [\,(c_\beta, \eta_i)\,], \qquad \beta = 1, \ldots, p.$$

(iii) Define the weighted snapshot coefficient matrices $R \in \mathbb{R}^{N_1 \times k}$ and $S \in \mathbb{R}^{N_2 \times \ell}$ by

$$R = [\,\alpha_1 r_1(t_1),\, \ldots,\, \alpha_{n_1} r_1(t_{n_1}),\, \ldots,\, \alpha_1 r_m(t_1),\, \ldots,\, \alpha_{n_1} r_m(t_{n_1})\,]^T,$$
$$S = [\,\beta_1 s_1(t_1),\, \ldots,\, \beta_{n_2} s_1(t_{n_2}),\, \ldots,\, \beta_1 s_p(t_1),\, \ldots,\, \beta_{n_2} s_p(t_{n_2})\,]^T.$$

Then the weighted snapshot vectors $\tilde{w}$ and $\tilde{z}$ defined in (8) and (9), respectively, are approximated by

$$\tilde{w} \approx R[\,\xi_1, \, \ldots, \, \xi_k\,]^T, \quad \tilde{z} \approx S[\,\eta_1, \, \ldots, \, \eta_\ell\,]^T.$$

Also, the matrix Γ is approximated by $\hat{\Gamma} = SNR^T$, where the $\ell \times k$ matrix N is given by $N = [(\eta_i, \xi_j)]$.

(iv) Compute the singular value decomposition of $\hat{\Gamma}$ as in (10) and choose $r <$ rank($\hat{\Gamma}$). Then the first r primary and adjoint balanced POD modes are approximated by

$$[\,\varphi_1, \, \ldots, \, \varphi_r\,]^T \approx \Sigma_r^{-1/2} V_r^* R[\,\xi_1, \, \ldots, \, \xi_k\,]^T,$$
$$[\,\psi_1, \, \ldots, \, \psi_r\,]^T \approx \Sigma_r^{-1/2} U_r^* S[\,\eta_1, \, \ldots, \, \eta_\ell\,]^T,$$

where Σ_r, U_r, and V_r are appropriate truncations of Σ_1, U_1, and V_1. Let $\Phi = \Sigma_r^{-1/2} V_r^* R \in \mathbb{R}^{r \times k}$ and $\Psi = \Sigma_r^{-1/2} U_r^* S \in \mathbb{R}^{r \times \ell}$. Then for each i,

$$\varphi_i \approx \sum_{j=1}^{k} \Phi_{ij} \xi_j, \quad \psi_i \approx \sum_{j=1}^{\ell} \Psi_{ij} \eta_j.$$

(v) Substitute the approximate modes into the reduced order model matrices (14):

$$A_r = [\,(A\varphi_j, \psi_i)\,] \approx \Psi[\,(A\xi_j, \eta_i)\,]\Phi^T,$$
$$B_r = [\,(b_j, \psi_i)\,] \approx \Psi[\,(b_j, \eta_i)\,],$$
$$C_r = [\,(\varphi_j, c_i)\,] \approx [\,(\xi_j, c_i)\,]\Phi^T,$$
$$a_0 = [\,(x_0, \psi_1), \, \ldots, \, (x_0, \psi_r)\,]^T \approx \Psi[\,(x_0, \eta_1), \, \ldots, \, (x_0, \eta_\ell)\,]^T.$$

3.2. *Comparison to the Finite Dimensional Method Applied to a Discretized Infinite Dimensional System*

A common method to compute a reduced order model for an infinite dimensional linear system is to first discretize the system and then apply a finite dimensional model reduction algorithm. The Galerkin method presented above gives one way to compare the infinite dimensional balanced POD algorithm presented here, which we term "balance POD then discretize," with the finite dimensional balanced POD algorithm applied to a discretization of an infinite dimensional system, which we call "discretize then balance POD." In this section, we compare the two approaches.

In the "discretize then balance POD" approach, one applies the Galerkin method (or some other discretization scheme) to the linear system (1) to obtain the ordinary differential equation system (15) in step 1 above along with the finite dimensional output equation $y_k = \tilde{C}_k r_\alpha$, where $\tilde{C}_k = [\,(\xi_j, c_i)\,]$. Finite dimensional balanced POD is then performed on this system to obtain a reduced order model.

If certain conditions are satisfied, the "balance POD then discretize" approach presented here produces the same reduced order model as the "discretize then balance POD" approach outlined above. It can be checked that the following conditions are sufficient:

- The Galerkin subspaces W_1 and W_2 must be equal (therefore, $k = \ell$ and $\tilde{M}_k = \hat{M}_k$).

- The Galerkin scheme must satisfy $\tilde{A}_k^* = \hat{A}_k$ (i.e., the conjugate transpose of the matrix approximation of A must equal the matrix approximation of the adjoint operator A^*).
- The same quadrature points and weights are used.
- The inner product for the finite dimensional balanced POD must be weighted by the matrix $\tilde{M}_k$, i.e., $(a, b) = a^T \tilde{M}_k b$.

In this case, the matrices Φ^T and Ψ are produced by the finite dimensional balanced POD algorithm, and the same reduced order model results from both approaches.

We note that certain problems and numerical schemes may not satisfy the first two conditions above. For example, if the domain of A does not equal the domain of A^*, the first condition may be difficult or impossible to satisfy. Also, certain Galerkin schemes may not satisfy the duality property required in the second condition; for an example with a delay equation, see [12]. In these cases, the "discretize then balance" approach may not produce an actual approximate balancing transformation.

In Section 4.4 we demonstrate numerically that if the inner product in the finite dimensional balanced POD is not weighted by the matrix $\tilde{M}_k$, then "discretize then balance POD" can produce incorrect results.

4.   Numerical Results for a Model Problem

In this section, we present numerical results for the convection diffusion system

$$w_t(t, x) = \mu w_{xx}(t, x) - \kappa w_x(t, x) + b(x)u(t),$$

$$y(t) = \int_0^1 c(x)w(t, x)\, dx,$$

$$w(t, 0) = 0, \quad w(t, 1) = 0, \quad w(0, x) = w_0(x).$$

with μ a positive constant, κ a real constant, $t > 0$ and $0 \leq x \leq 1$. The functions $b(x)$ and $c(x)$ are piecewise constant with $b(x) = 1$ when $b_1 < x < b_2$, $c(x) = 1$ when $c_1 < x < c_2$, and both are zero otherwise. First, we discuss properties of this system and give an explicit formula for the transfer function. Next, we approximate the Hankel singular values, balancing modes, and transfer function and compare the results with standard balancing computations and the exact transfer function. Finally, we show that incorrect implementation of finite dimensional balanced POD applied to a discretization of the system results in an inaccurate reduced order model.

4.1.   *Properties of the System*

The convection diffusion system can be written as an infinite dimensional linear system as follows. Let the Hilbert space X equal $L^2(0, 1)$, the space of square integrable functions, with the standard inner product $(f, g) = \int_0^1 f(x)g(x)\, dx$. The system operators are defined by

$$Aw = \mu w_{xx} - \kappa w_x, \quad D(A) = H^2 \cap H_0^1,$$

$$Bu = b(x)u, \quad Cw = (c, w) = \int_0^1 c(x)w(x)\, dx.$$

Here, H^m is the standard Sobolev space of functions with m derivatives all of which are square integrable; also, any function $w \in H_0^1$ must satisfy the Dirichlet boundary conditions $w(0) = 0$ and $w(1) = 0$. The input space U and the output space Y both equal the real numbers. Note that $B : U \to X$ and $C : X \to Y$ are finite rank and bounded since they are of the form (7). We will also require the adjoint of A given by $A^*w = \mu w_{xx} + \kappa w_x$ on the domain $D(A^*) = D(A) = H^2 \cap H_0^1$.

We now briefly show that A generates an exponentially stable C_0-semigroup and provide an exact formula for the transfer function of the system.

PROPOSITION 4.1 *The convection diffusion operator A defined above generates an exponentially stable C_0-semigroup. The transfer function $G(s) = C(sI - A)^{-1}B$ defined for s in the resolvent set of A is given by*

$$G(s) = \sum_{n=1}^{\infty} \frac{1}{s - \lambda_n} \left(\int_{c_1}^{c_2} f_n(x) \, dx \right) \left(\int_{b_1}^{b_2} g_n(x) \, dx \right), \tag{16}$$

where

$$\lambda_n = -\mu n^2 \pi^2 - \kappa^2/4\mu,$$
$$f_n(x) = \sqrt{2} e^{\kappa x/2\mu} \sin(n\pi x),$$
$$g_n(x) = \sqrt{2} e^{-\kappa x/2\mu} \sin(n\pi x).$$

Note that the integrals in the formula for the transfer function can be computed exactly.

Proof It is easily shown that $-A$ takes the form of a Sturm-Liouville operator; therefore, the results in [13] show that A generates a C_0-semigroup and the convection diffusion system (A, B, C) is a Riesz-spectral system (as in [11, Def. 4.1.1]).

It can be shown that the eigenvalues of A are given by λ_n and the corresponding eigenfunctions are given by $f_n(x)$. Furthermore, the eigenvalues of A^* equal the eigenvalues of A and the eigenfunctions are given by $g_n(x)$. The eigenfunctions of A and A^* form a biorthogonal sequence, i.e., $(f_j, g_k) = \delta_{jk}$.

Since the convection diffusion system is a Riesz-spectral system and the eigenvalues of A are all negative and bounded away from the imaginary axis, the C_0-semigroup generated by A is exponentially stable [11, Theorem 2.3.5 d]. Furthermore, once again using the Riesz-spectral property of the system gives [11, Lemma 4.3.10]

$$G(s) = \sum_{n=1}^{\infty} \frac{1}{s - \lambda_n} C f_n B^* g_n.$$

Since $B^*w = (b, w) = \int_0^1 b(x)w(x) \, dx$, the result follows. $\square$

4.2.   *Hankel Singular Values and Balancing Modes*

We now consider the numerical approximation of the Hankel singular values and the balancing modes. We do not have explicit expressions for these quantities, therefore we compare the results of balanced POD with standard balancing computations applied to a finite element discretization of the system. For the balanced POD computations, the solutions of the primary (12) and dual (13) linear differential

12   *John R. Singler and Belinda A. Batten*

equations were approximated with standard piecewise linear finite elements using equally spaced nodes. The solutions were integrated over $0 \leq t \leq 2$ using Matlab's `ode15s` solver with default error tolerances. The quadrature points were chosen as the time points returned from `ode15s` and the trapezoid rule was used for the quadrature weights. Time refinement was performed by decreasing the error tolerances of the ODE solver. For all computations, we chose the system parameters given in Table 1.

Table 1.   System Parameters

μ	κ	b_1	b_2	c_1	c_2
0.1	1	0.1	0.3	0.6	0.7

For this model problem, balanced POD gives identical results with the standard balancing computations when both methods are refined until convergence. In the balanced POD computations, spatial refinement was more important for convergence than time refinement. This is not surprising since the solutions of the primary and dual linear systems are not highly variable in time.

In Figure 1, we show the first 20 approximate Hankel singular values using balanced POD and the matrix representation approach. The methods produce identical results. For each computation, we used 256 equally spaced finite element nodes. The singular values are converged — further refinement in space (and in time for balanced POD) produces little change. The remaining singular values are below machine precision; it is doubtful that these values are accurate.
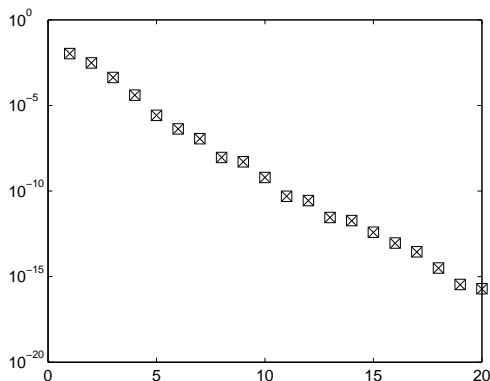
Figure 1. Approximate Hankel singular values for standard balancing ($\square$) and balanced POD ($\times$).

The nuclear norm of the Hankel operator is the sum of the Hankel singular values. When refined, both methods give 0.0143 as an approximation to this sum. The last digit does not appear to converge for either method as they are refined. This is most likely caused by inaccuracies in the singular values below machine precision. However, we may still conjecture that both methods approximate the nuclear norm accurately to the third decimal place.

In Figures 2 and 3, we show primary and adjoint balanced POD modes. All modes are converged and the standard balancing computations produce identical results. In general, the higher numbered modes are slower to converge under refinement for both methods. This behavior is most likely due to the fact that the higher numbered modes tend to oscillate more than the lower numbered modes. For these computations, 128 equally spaced finite element nodes were used.
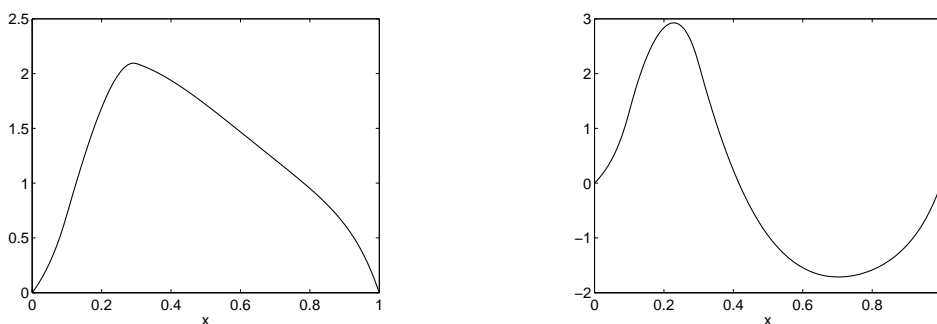
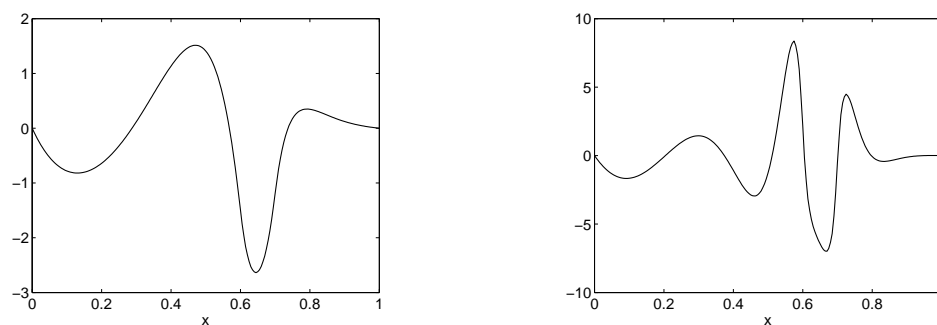Figure 2. Balanced POD mode 1 (left) and mode 2 (right).

Figure 3. Adjoint balanced POD mode 3 (left) and mode 5 (right).

4.3. *Transfer Functions*

Recall that the difference between the transfer functions of the convection diffusion system and the exact rth order truncated balanced system satisfies the error bound

$$\|G - G_r\|_\infty \leq 2 \sum_{k>r} \sigma_k, \tag{17}$$

where σ_k is the kth ordered Hankel singular value. In this section, we numerically investigate the error in the $\mathcal{H}^\infty$ norm using the transfer function of the rth order truncated balanced POD system. We use the same implementation of the algorithm and the system parameters as above.

Figure 4 shows the magnitude of the transfer function of the convection diffusion system evaluated at $s = i\omega$ for $10^{-2} \leq \omega \leq 10^4$. The transfer function is evaluated using (16); the integrals are computed exactly and 40 terms are taken of the infinite sum (this is sufficient for convergence). Recall that the $\mathcal{H}^\infty$ norm of a transfer function is the maximum singular value of the transfer function evaluated along the imaginary axis. Since the convection diffusion system is single input-single output, the $\mathcal{H}^\infty$ norm of the convection diffusion transfer function is given by the largest value of $|G(i\omega)|$ for ω real. Thus, figure 4 gives $\|G\|_\infty \approx 0.016$.

We now check whether the transfer functions from converged balanced POD meet the error bound (17). Figure 5 shows the magnitude of the balanced POD transfer function $G_r(s)$ for $r = 2$ and the error $|G(s) - G_r(s)|$ evaluated at $s = i\omega$ for $10^{-2} \leq \omega \leq 10^4$. We used 128 equally spaced finite element nodes for the computations. The balanced POD transfer function is very close to the true convection diffusion transfer function. For $r = 2$, the error bound is approximately 9.5×10^{-4}. Over the range of ω given above, we have $\max|G(i\omega) - G_r(i\omega)| = 9 \times 10^{-4}$. Thus,
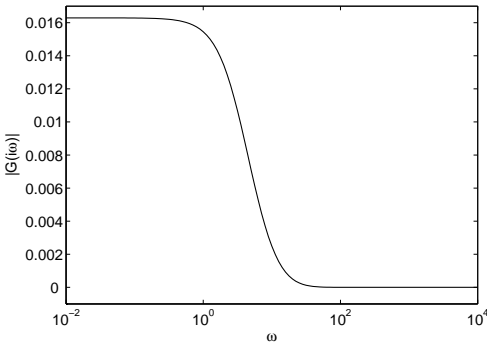
Figure 4.   The magnitude of the transfer function of the convection diffusion system given in (16) evaluated at $s = i\omega$ for $10^{-2} \leq \omega \leq 10^4$. The integrals are computed exactly and 40 terms are taken in the infinite sum.
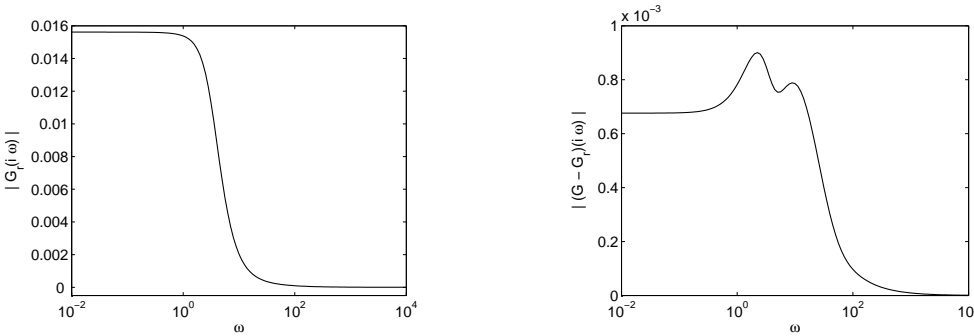

Figure 5.   Magnitude of the balanced POD transfer function $|G_r(s)|$ for $r = 2$ (left) and transfer function error $|G(s) - G_r(s)|$ (right) both evaluated at $s = i\omega$ for $10^{-2} \leq \omega \leq 10^4$.

the balanced POD transfer function satisfies the error bound (17) in this case. Standard balancing computations using finite element matrix approximations produced identical results.

For $r = 3$, the error bound is approximately 8.7×10^{-5}. In this case, 1200 equally spaced finite element nodes are required to bring the error below the bound (see Figure 6). Although a very large number of nodes is required to satisfy the error
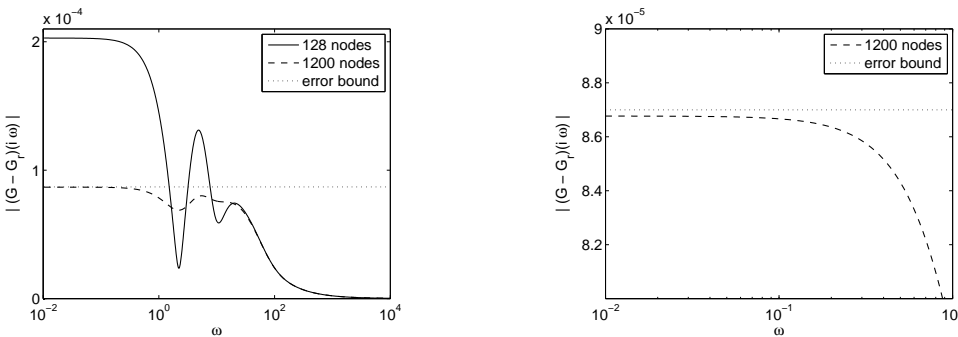

Figure 6.   Balanced POD transfer function error $|G(s) - G_r(s)|$ for $r = 3$ evaluated at $s = i\omega$ for $10^{-2} \leq \omega \leq 10^4$ using 128 and 1200 nodes (left) and the same plot over the range $10^{-2} \leq \omega \leq 1$ (right).

bound (17), 128 equally spaced nodes gives an error of approximately 2×10^{-4}, which is quite near the error bound. Again, standard balancing computations using matrix representations produced identical results.

The balanced POD algorithm is intended to give an approximate truncated balanced realization of the original system. The numerical results presented here indi-

cate that the balanced POD reduced order model is a good approximation to the exact truncated balanced reduced order model. We have seen that an extremely fine discretization of an infinite dimensional system may be necessary for the balanced POD transfer function to satisfy the balancing error bound (17). However, our results also demonstrate that a coarser discretization may still yield a transfer function that nearly satisfies the bound. Therefore, in many cases balanced POD with a coarser discretization may yield a low order model that is a good approximation to the infinite dimensional system.

4.4.   *Discretize then Balance POD: An Example with Incorrect Results*

In this section, we first discretize the convection diffusion system and then apply finite dimensional balanced POD to the discrete system. In Section 3.2, we gave conditions that guarantee when this procedure yields the same reduced order model as balanced POD applied to the infinite dimensional convection diffusion system (with the differential equations discretized by the Galerkin method). The last condition for discretization to "commute" with balanced POD was that the finite dimensional inner product is weighted by $\tilde{M}_k$, the matrix of inner products of the Galerkin basis functions. Below we do *not* weight the finite dimensional inner product by the matrix $\tilde{M}_k$. The numerical results below demonstrate that this incorrect implementation leads to nonconvergent Hankel singular values, balancing modes, and transfer function.

We again use standard piecewise linear finite elements with equally spaced nodes for the spatial discretization. The finite element basis functions are not orthonormal with respect to the L^2 inner product, therefore the matrix $\tilde{M}_k$ is not the identity matrix. The finite dimensional balanced POD algorithm was implemented in exactly the same manner as the infinite dimensional version of the algorithm described in Section 4.2 above. The system parameters were again chosen as in Table 1.

Figure 7 shows the Hankel singular values for 32, 64, 128, and 256 equally spaced nodes. The values no longer converge; the values increase as the refinement level increases. Figure 7 also shows the results of balanced POD applied to the infinite dimensional convection diffusion system. In this case, as the refinement level increases, the Hankel singular values converge (also see Figure 1).
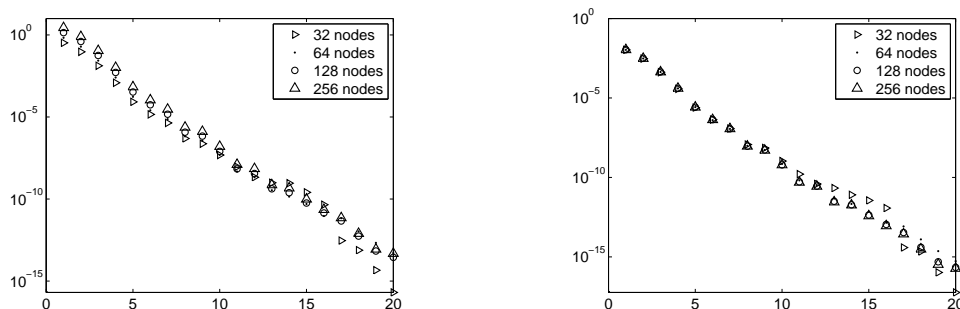

Figure 7.   Hankel singular values for "discretize then balance POD" (left) and "balance POD then discretize" (right).

Figure 8 shows the first balanced POD mode for 32, 64, 128, and 256 equally spaced nodes. The mode no longer converges; the shape of the mode stays the same yet the size of the mode decreases under refinement. Figure 8 also shows the results of balanced POD applied to the infinite dimensional convection diffusion system.

In this case, as the refinement level increases, the balanced POD mode converges (also see Figure 2). Similar results were observed for other balanced POD modes.
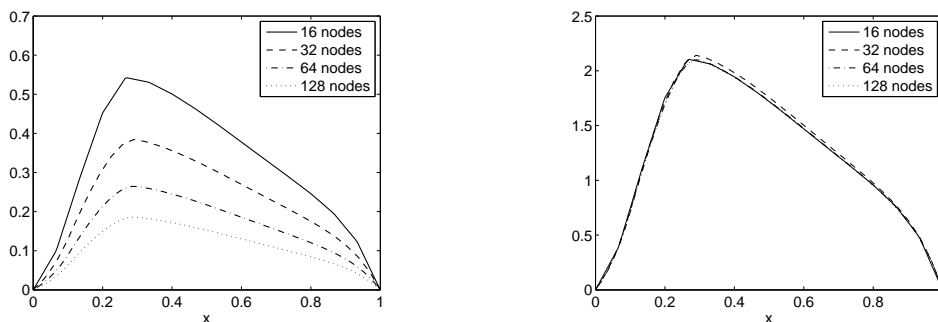
Figure 8.   The first balanced POD mode for "discretize then balance POD" (left) and "balance POD then discretize" (right).

Figure 9 compares the exact transfer function of the convection diffusion system with the balanced POD transfer functions of order $r = 4$ and $r = 7$ computed with 32, 64, 128, and 256 equally spaced nodes. The magnitudes of the transfer functions are evaluated at $s = i\omega$ for $10^{-4} \leq \omega \leq 10^4$. The balanced POD transfer functions do not converge to the true transfer function as the refinement level increases; in fact, the transfer functions do not appear to converge at all. Furthermore, increasing the order r of the reduced order model does not make the balanced POD transfer functions closer to the true transfer function; in fact, increasing r produces no visible change in the balanced POD transfer functions.
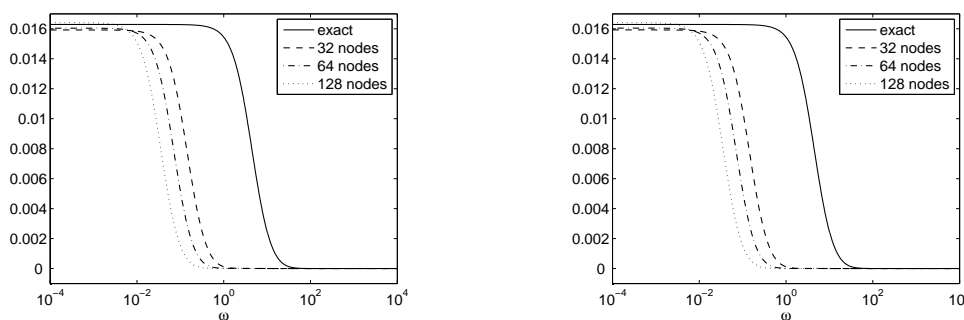
Figure 9.   Exact convection diffusion transfer function versus the "discretize then balance POD" transfer function for $r = 4$ (left) and $r = 7$ (right).

This example shows that one must be cautious when discretizing an infinite dimensional system and then using balanced POD to construct the reduced order model. In particular, if the finite dimensional inner product is not weighted by the matrix $\tilde{M}_k$, then the resulting reduced order model may not be close to the infinite dimensional system. Furthermore, if either of the first two conditions in Section 3.2 are not satisfied by the discretization scheme, then the resulting reduced order model may also suffer from a loss of accuracy. We note that a lack of convergence of the Hankel singular values or balancing modes should be a good indication that the balanced POD reduced order model may not be near the true infinite dimensional system.

5. Conclusions and Future Work

In this paper, we extended Rowley's balanced POD algorithm to a class of infinite dimensional systems. Numerical results for a convection diffusion system indicate convergence of the algorithm by comparing balanced POD with standard balancing computations and the exact transfer function. In addition, we compared finite and infinite dimensional algorithms and gave conditions when balanced POD "commutes" with discretization. Numerical results for the convection diffusion system demonstrated that an incorrect implementation of balanced POD applied to a discretization of an infinite dimensional system can result in an inaccurate reduced order model.

This method shows promise for reduced order model design. In particular, it is computationally tractable for infinite dimensional systems, even if approximating finite dimensional systems have very high dimensions. Additionally, it is applicable even if matrices from approximating systems are not available. One only needs to be able to approximate solutions of standard and dual linear evolution equations. Moreover, there is potential to use error estimators for the solutions of the linear equations to show where to refine to improve accuracy.

We point out, however, that balanced POD may not be feasible for (1) systems with solutions that decay slowly to zero or are highly oscillatory in time because they may need a large number of time quadrature points, or (2) systems that have a large number of inputs and outputs. We note that Rowley's paper treats the case of a large number of outputs using a POD projection.

In a future paper, we will complete the convergence analysis of this method and examine Rowley's output POD projection. In addition, we will compare this approach with balanced truncation methods using large scale matrix Lyapunov solvers (see [3, 4] and the references therein). Even in the case that the matrix solvers are faster, balanced POD may still be preferable due to the advantages listed above. Future work includes extending this approach to systems with unbounded input and output operators.

References

[1] Datta, B.N., 2004 *Numerical Methods for Linear Control Systems* (San Diego, CA: Elsevier Academic Press).
[2] Zhou, K., Doyle, J.C. and Glover, K., 1996 *Robust and Optimal Control* (Prentice-Hall).
[3] Antoulas, A.C., 2005 *Approximation of Large-Scale Dynamical Systems* (Philadelphia, PA: SIAM).
[4] Benner, P., Mehrmann, V. and Sorensen, D.C. (Eds) , 2005 *Dimension Reduction of Large-Scale Systems* (Berlin: Springer-Verlag).
[5] Rowley, C.W., 2005, Model reduction for fluids, using balanced proper orthogonal decomposition. *Internat. J. Bifur. Chaos Appl. Sci. Engrg.*, **15**(3), 997–1013.
[6] Curtain, R.F. and Glover, K., 1986, Balanced realisations for infinite-dimensional systems. In: *Operator Theory and Systems (Amsterdam, 1985)* (Basel: Birkhäuser), pp. 87–104.
[7] Glover, K., Curtain, R.F. and Partington, J.R., 1988, Realization and approximation of linear infinite-dimensional systems with error bounds. *SIAM Journal on Control and Optimization*, **26**(4), 863–898.
[8] Curtain, R.F., 2003, Model reduction for control design for distributed parameter systems. In: *Research Directions in Distributed Parameter Systems* (Philadelphia, PA: SIAM), pp. 95–121.
[9] Curtain, R.F. and Sasane, A.J., 2001, Compactness and nuclearity of the Hankel operator and internal stability of infinite-dimensional state linear systems. *Internat. J. Control*, **74**(12), 1260–1270.
[10] Weidmann, J., 1980 *Linear operators in Hilbert spaces* (New York: Springer-Verlag).
[11] Curtain, R.F. and Zwart, H.J., 1995 *An Introduction to Infinite-Dimensional Linear System Theory* (New York: Springer-Verlag).
[12] Borggaard, J., Burns, J.A., Vugrin, E. and Zietsman, L., 2004, On strong convergence of feedback operators for non-normal distributed parameter systems. In: *Proceedings of the Proceedings of the IEEE Conference on Decision and Control*, Vol. 2, pp. 1526 – 1531.
[13] Delattre, C., Dochain, D. and Winkin, J., 2003, Sturm-Liouville systems are Riesz-spectral systems. *Int. J. Appl. Math. Comput. Sci.*, **13**(4), 481–484.