Order Reduction for Large Scale Finite Element Models: a Systems Perspective

William Gressick, John T. Wen, Jacob Fish

ABSTRACT

Large scale finite element models are routinely used in design and optimization for complex engineering systems. However, the high model order prevents efficient exploration of the design space. Many model reduction methods have been proposed in the literature on approximating the high dimensional model with a lower order model. These methods typically replace a fine scale model with a coarser scale model in schemes such as coarse graining, macro-modeling, domain decomposition and homogenization. This paper takes a systems perspective by stating the model reduction objective in terms of the approximation of the mapping between specified input and output variables. Methods from linear systems theory, including balance truncation and optimal Hankel norm approximation, are reviewed and compared with the standard modal truncation. For high order systems, computational load, numerical stability, and memory storage become key considerations. We discuss several computationally more attractive iterative schemes that generate the approximate gramian matrices needed in the model reduction procedure. A numerical example is also included to illustrate the model reduction algorithms discussed in the paper. We envision that these systems oriented model reduction methods complementing the existing methods to produce low order models suitable for design, optimization, and control.

KEY WORDS

Model Reduction, Large Scale Systems, Balanced Truncation, Finite Element Methods, Approximate Gramian

1. INTRODUCTION

For complex engineering systems such as large mechanical structures, fluid dynamic systems, integrated circuits, and advanced materials, the underlying dynamical models are typically obtained from the finite element method or discretization of partial differential equations. To obtain good approximations of the underlying physical processes, these models are necessarily of very high order. In order to use these models effectively in design optimization and iteration, the high order systems need to be reduced in size while still retaining relevant characteristics. Many model reduction/simplication schemes have been proposed in the past, such as Guyan and the related improved reduced system (IRS) methods [1, 2], hierarchical modeling [3, 4], macro-modeling [5,6], domaining decomposition [7], and others. This paper approaches model reduction from a systems perspective. In contrast to other model reduction techniques for finite element models, the systems approach seeks to retain only the dominant dynamics that are strongly coupled to the specified input and output. This is similar to the goal-oriented adaptive mesh generation method, where the mesh geometry (and hence the approximate model) is governed by its influence on the properties of interests [8]. There has been a recent surge of interests in model reduction for large scale systems from the systems community [9–13]. Well conditioned numerical algorithms have also been developed and become available [14]. The goal of this paper is to present a tutorial of this class of approaches and the underlying algorithms.

The basic problem is as follows: Given an nth order linear time invariant (LTI) system with state space parameters (A, B, C, D), find an rth order reduced order model (A_r, B_r, C_r, D_r), with $r << n$. The goal of model reduction is to make the difference between the full order model and reduced order model small under some appropriate norm. For LTI systems, model reduction methods can be broadly classified as singular value decomposition (SVD) based approach and the classical moment matching method [9]. The SVD based methods can be further separated into model based and data driven. The model based methods assume the available of a high order model. Data driven methods produce a reduced order model based on the input/output data. This is also known as the model identification problem [15, 16]. In this paper, we will focus on model based SVD methods, since a high order model (e.g., obtained from the finite element method) is assumed available. We will consider four different norms for comparison: the H_∞ norm, which is the worst case input/output L_2 gain, the H_2 norm, which is the worst case gain from the peak input spectral density to output power, and the time domain L_∞ (largest amplitude) and L_2 (energy) norms under a specific input of interest. Our discussion will focus on the balanced truncation method which has an *a priori* H_∞ error bound and is stability preserving. However, the method in its original form has computation complexity $\mathcal{O}n^3$, and faces numerical difficulties for stiff high order systems. We then present a number of iterative methods that produce *approximate* balanced truncated models. These methods possess better computational and numerical properties, especially when the system matrix is sparse. A numerical example involving a piezo-composite beam is included to illustrate the various methods discussed in the paper.

This paper is organized as follows. Section 2 reviews the basic description of linear systems. Section 3 presents various model reduction methods, the commonly used modal reduction, balanced truncation, and optimal Hankel norm reduction. Section 4 discusses various approximation techniques for the controllability and observability gramians needed in the balanced truncation. The balanced truncation type of model reduction using the approximate gramians is shown in Section 5. A piezo-composite beam example is included in Section 6 to illustrate the performance of the methods presented.

2. PRELIMINARIES

2.1. System Description

The finite element method typically generates a high order continuous-time LTI system in the state space form:

$$\dot{x}(t) = Ax(t) + Bu(t), \tag{2.1}$$

where $x(t) \in \mathbb{R}^n$ is the state vector and $u(t) \in \mathbb{R}^{n_i}$ is the input vector. For mechanical structures, the model is usually expressed in the generalized second order form

$$M\ddot{q}(t) + F\dot{q}(t) + Kq(t) = Hu(t), \tag{2.2}$$

where q, $\dot{q}$, $\ddot{q}$ are the generalized coordinate, generalized velocity, and generalized acceleration, respectively, M, F, K are the mass, damping, and stiffness matrices, respectively, and H is the influence matrix corresponding to the input $u(t)$. Note that F is difficult to obtain accurately, and is frequently just set

to zero. The second order model can be transformed to the state space form by, for example, defining the state vector as

$$x(t) = \begin{bmatrix} q(t) \\ \dot{q}(t) \end{bmatrix}. \tag{2.3}$$

Then

$$\dot{x}(t) = \begin{bmatrix} 0 & I \\ -M^{-1}K & -M^{-1}F \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ M^{-1}H \end{bmatrix} u(t). \tag{2.4}$$

The state definition is not unique, for example, the Legendre transformation is also a popular choice:

$$x(t) = \begin{bmatrix} q(t) \\ M\dot{q}(t) \end{bmatrix}. \tag{2.5}$$

In this case,

$$\dot{x}(t) = \begin{bmatrix} 0 & M^{-1} \\ -K & -F \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ H \end{bmatrix} u(t). \tag{2.6}$$

In the state space model, we will also consider an output of interest

$$y(t) = Cx(t) + Du(t), \tag{2.7}$$

where $y(t) \in \mathbb{R}^{n_o}$. The input/output, (u, y), may correspond to a particular property of interests, or physical actuators and sensors.

Denote the system with input u and output y by G (Fig. 1). For a given choice of the state, the quadruplet (A, B, C, D) is called the state space representation for G. A state space model can be transformed to other state coordinates through a coordinate transformation:

$$z = T^{-1}x \tag{2.8}$$

where T is any invertible matrix. The resulting state space representation is $(T^{-1}AT, T^{-1}B, CT, D)$.

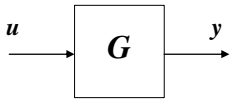

Figure 1: Input/output system under consideration

An LTI system may also be characterized by its impulse response, $g(t)$, which in terms of state space parameters is given by $Ce^{At}B\mathbf{1}(t) + D\delta(t)$, where $\mathbf{1}(t)$ is the unit step function and $\delta(t)$ the unit impulse. In this case, the output is related to the input through a convolution:

$$y(t) = \int_0^t g(t-\tau)u(\tau)\,d\tau + y_{zi}(t) \tag{2.9}$$

where y_{zi} is the zero input (unforced) response due to the initial state.

Another characterization of an LTI system is to transform (2.9) to the Laplace domain:

$$Y(s) = G(s)U(s) + Y_{zi}(s). \tag{2.10}$$

where

$$G(s) = C(sI - A)^{-1}B + D. \tag{2.11}$$

Generalized second order systems (2.2) can also be represented as

$$\begin{bmatrix} I & 0 \\ 0 & M \end{bmatrix} \dot{x} = \begin{bmatrix} 0 & I \\ -K & -F \end{bmatrix} + \begin{bmatrix} 0 \\ H \end{bmatrix} u. \tag{2.12}$$

A system in this form is referred to as a *descriptor system*, where $\dot{x}$ is multiplied by a matrix other than the identity matrix. The key attractions of the descriptor form are the avoidance of mass matrix inversion and sparsity. In the usual state space form, (2.4) and (2.6), the mass matrix inversion can destroy sparsity if the mass matrix is not diagonal. For this reason, model reduction of descriptor systems in their native form is an active area of research. The subject and its associated numerical methods are presented in [17,18] and will not be pursued here.

An LTI system, G, with input $u(t) \in \mathbb{R}^{n_i}$ and output $y(t) \in \mathbb{R}^n_o$ may be regarded as a linear operator mapping $L_p^{n_i}[t_1, t_2]$ to $L_p^{n_o}[t_1, t_2]$, where $1 \le p \le \infty$, and (t_1, t_2) is the time range of interests. The worst case input/output L_p gain is a norm of G (induced by the L_p-norms):

$$\|G\|_{i,p} = \sup_{u \in_p^{n_i}} \frac{\|y\|_{L_p^{n_o}}}{\|u\|_{L_p^{n_i}}}. \tag{2.13}$$

Conceptually, G can be thought of as mapping a unit $L_p^{n_i}$-ball to an ellipsoid in $L_p^{n_o}$ as shown in Fig. 2. Then $\|G\|_{i,p}$ is the length of major axis of the ellipsoid. The most common induced norms are L_2 and L_∞ norms. In the case that $(t_1, t_2) = (0, \infty)$, the L_2 induced norm is the same as the norm of the Hardy space H_∞ and can be calculated by using the transfer function of G (we will soon encounter this again from the frequency domain perspective). The L_∞ induced norm can be shown to be the L_1 norm of the

impulse response of G:

$$\|G\|_{i,\infty} = \int_0^\infty \|g(t)\| \, dt = \|g\|_{L_1} . \tag{2.14}$$

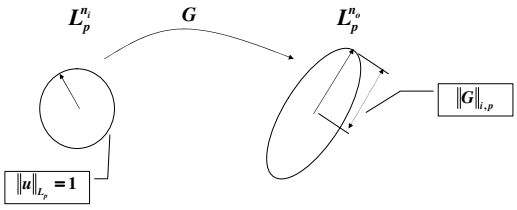

Figure 2: Input/Output Ellipsoid

An operator norm may also be defined for G directly without referring to the input/output norms. A commonly used norm is

$$\|G\|_{H_2} = \left(\int_{t_1}^{t_2} \text{tr} \left[g(t) g^T(t) \right] \, dt \right)^{\frac{1}{2}} \tag{2.15}$$

where **tr** denotes the trace of a matrix, and $g(t)$ is the impulse response of G. This norm can be thought of as the generalization of the matrix Frobenius norm to LTI systems. In the case that $(t_1, t_2) = (0, \infty)$, the norm (2.15) is the same as the norm of the Hardy space H_2 [19].

If the time range is set to $[0, \infty)$, then an LTI system $G : L_p^{n_i}[0, \infty) \to L_p^{n_o}[0, \infty)$ is called an L_p-stable system (these notations may be extended to nonlinear dynamical systems, see [20]). The L_p stability is equivalent to the stability of the state space system, called internal stability, (i.e., all eigenvalues of A in the left half plane) under the stabilizability and detectability conditions (basically ensuring that there is no "hidden" internal dynamics) [20]. For generalized second order systems, stability means that the damping and stiffness matrices are positive definite. The L_p-norm is a natural performance metric, for example, L_p-norm of $G - \hat{G}$, where $\hat{G}$ is the reduced order model.

A stable LTI system may also be regarded as a linear operator under the Fourier transform, where the input is $\hat{u}(j\omega) \in L_p^{n_i}(-j\infty, j\infty)$, the Fourier transform of $u(t)$, and the output is $\hat{y}(j\omega) \in L_p^{n_o}(-j\infty, j\infty)$, the Fourier transform of $y(t)$ (assuming that the Fourier transforms of the input and output signals exist). In this case, the system is represented by the Fourier transform of the impulse response, $G(j\omega)$, or the transfer function evaluated along the imaginary axis. If we regard $G(j\omega)$ as an L_p-mapping, we can again use the induced L_p-norm as the performance metric. The most common choice is the induced L_2-norm, which is also called the H_∞-norm (norm corresponding to the Hardy space H_∞). The H_∞ norm is a direct generalization of the matrix norm induced by the Euclidean norm, which is the maximum singular value of the matrix. The H_∞ norm is related to the transfer function through

$$\|G\|_{H_\infty} = \sup_\omega \|G(j\omega)\| \tag{2.16}$$

where $\|G(j\omega)\|$ denotes the maximum singular value of $G(j\omega)$.

We can also regard $G(j\omega)$ directly as an element of $L_2^{n_o \times n_i}(-j\infty, j\infty)$. The corresponding norm (not an L_p induced norm) is called the H_2-norm (norm for the Hardy space H_2), which may be considered as a generalization of the Frobenius norm. The H_2 norm is related to the transfer function as

$$\|G\|_{H_2} = \left(\frac{1}{2\pi} \int_{-\infty}^{\infty} \text{tr} \left[G(j\omega) G^T(-j\omega) \right] \, d\omega \right). \tag{2.17}$$

By using the Parseval Theorem, the frequency domain expression for the H_2 norm can be shown to be the same as the time domain expression in (2.15). The H_2 norm may be considered as an induced norm for power signals [19]. Let $\mathcal{P}$ be the space of finite power signals with the power of a signal u defined as

$$\|u\|_{\mathcal{P}} = \left(\lim_{T \to \infty} \frac{1}{2T} \int_{-T}^{T} \|u(t)\|^2 \, dt \right)^{\frac{1}{2}} . \tag{2.18}$$

Define the autocorrelation matrix of u as

$$R_{uu}(\tau) = \lim_{T \to \infty} \frac{1}{2T} \int_{-T}^{T} u(t+\tau) u^T(t) \, dt. \tag{2.19}$$

The spectral density $S_{uu}(j\omega)$ of u is the Fourier transform of R_{uu}. Signals with bounded spectral density are denoted by

$$\mathcal{S} = \left\{ u(t) \in \mathbb{R}^{n_i} : \|u\|_{\mathcal{S}} := \|S_{uu}(j\omega)\|_{L_\infty} < \infty \right\}. \tag{2.20}$$

Then H_2 norm is the induced norm from $\mathcal{S}$ to $\mathcal{P}$.

We will use the above norms to evaluate our reduced-order models, by applying them to the difference between the full order LTI and the reduced-order model, $G - \hat{G}$. However, small $\left\| G - \hat{G} \right\|$

have different meanings depending on the norm used. For performance comparison, we will use four different metrics, summarized in Table 1: H_∞ norm, H_2 norm, output L_∞ norm, and output L_2 norm (the latter two for a specific input function). Small $\left\|G-\hat{G}\right\|_{H_\infty}$ may be interpreted in a worst-case sense: the L_2 norm of the output is small for *all* inputs with unit L_2 norms. However, there could be large amplitude errors for short durations. Also, for a given u, this norm could be very conservative, meaning that lower order $\hat{G}$ may be obtained to achieve the same output error norm. Small $\left\|G-\hat{G}\right\|_{H_2}$ means that the L_2 norm of the difference between the impulse response (or, equivalently, between the transfer functions) is small. This does not directly translate to time domain error bound, however. In terms of the interpretation of induced norm from $\mathcal{S}$ to $\mathcal{P}$, small H_2 norm means that under unit Gaussian white noise input (unit spectral density), the power of the output is small. Small output L_2 and L_∞ norms mean that the time domain output response will be small in the L_2 or L_∞ sense for the given input.

3. MODEL REDUCTION METHODS

3.1. Modal Truncation

Modal truncation is perhaps one of the simplest and most well-known model reduction methods. The basic idea is simple: Decompose the transfer function into a sum of "modes" which are transfer functions with a single real pole or a pair of complex poles. A reduced order model is obtained by retaining only the dominant modes (those contributing the most to the transfer function). In many cases, it is the high frequency modes that are discarded, due to damping and bandwidth limitation of actuators and sensors. In terms of the state space representation, modal truncation first transforms the system into the modal form where the system matrix is block diagonal with A_1 containing the dominant modes:

$$\begin{aligned} T^{-1}AT &= \begin{bmatrix} A_1 & 0 \\ 0 & A_2 \end{bmatrix}, T^{-1}B = \begin{bmatrix} B_1 \\ B_2 \end{bmatrix}, \\ CT &= \begin{bmatrix} C_1 & C_2 \end{bmatrix}. \end{aligned} \tag{3.1}$$

The state space representation of the modal truncated model is then (A_1, B_1, C_1, D). The usual approach is to represent A in the Jordan form, and then retain the low frequency eigenvalues only.

Modal truncation method is simple in principle, but is limited in practice by the difficulty to assess the modal dominance of a system. In other words, knowledge of which modes should be retained is not always clear, especially in systems which have closely-spaced eigenvalues, lightly damped high frequency modes, or wide band input excitations. The method also lacks an *a priori* error bound. In terms of implementation, an eigen-decomposition on the full system is required which can be computationally expensive and numerically ill-conditioned for large scaled systems.

3.2. Balanced Realization

Modal truncation is driven by the eigen-structure of A and does not explicitly take the system's input/output properties into account. Another approach to model reduction is to retain only the state dynamics that are strongly coupled to the input/output of the system. To assess how strong this coupling is, we apply the concepts of controllability and observability. We begin by defining controllability. Consider all $u \in L_2^{n_i}[0,T]$ with $\|u\|_{L_2} = 1$ applied to the system initially at rest (zero state). The corresponding states at T, $x(T) \in \mathbb{R}^n$, indicate the strength of coupling between input and state spaces. We consider $x(T)$ strongly coupled to the input $u(t)$ if $\|x(T)\|$ is large and vice versa. If $\|x(T)\| = 0$, then those states are decoupled from the input. This may be visualized as a mapping of a unit ball in $L_2^{n_i}[0,T]$ to an ellipsoid in $\mathbb{R}^n$, called the controllability ellipsoid (see Fig. 3). The principal axes of the ellipsoid indicate the degree of coupling between the state in that direction to the input signal. Denote the mapping of L_2-input to the final state, $x(T)$ as $L_T : L_2^{n_i}[0,T] \to \mathbb{R}^n$:

$$L_T u := \int_0^T e^{A(T-\tau)} B u(\tau)\, d\tau. \tag{3.2}$$

The lengths of the principal axes of the controllability ellipsoid are the singular values of L_T, or, equivalently, the square root of the eigenvalues of

$$P(T) = L_T^* L_T \tag{3.3}$$

where L_T^* is the operator adjoint of L_T and $P(T)$ is an $n \times n$ positive semi-definite matrix, called the controllability gramian (at time T). The controllability gramian may be calculated from a linear matrix differential equation

$$\dot{P}(t) = AP(t) + P(t)A^T + BB^T, \quad P(0) = 0_{n\times n}. \tag{3.4}$$

Error Norm	Interpretation
$\left\|G-\hat{G}\right\|_{H_\infty}$	Worst case L_2 gain
$\left\|G-\hat{G}\right\|_{H_2}$	Worst case spectral density to power gain
$\left\|Gu-\hat{G}u\right\|_{L_\infty}$	Maximum output amplitude with input u
$\left\|Gu-\hat{G}u\right\|_{L_2}$	Maximum output L_2 norm with input u

Table 1: Error norms considered in this paper and their interpretations

The solution can also be written as a matrix integral:

$$P(T) = \int_0^T e^{At} B B^T e^{A^T t}\, dt. \tag{3.5}$$

For stable systems (all eigenvalues of A are in the strict left half plane), $P(t)$ converges to a steady state matrix, P, as $t \to \infty$. In this case, P solves the following linear matrix equation called the Lyapunov equation:

$$AP + PA^T + BB^T = 0. \tag{3.6}$$

The solution can also be written as an integral:

$$P = \int_0^\infty e^{At} B B^T e^{A^T t}\, dt. \tag{3.7}$$

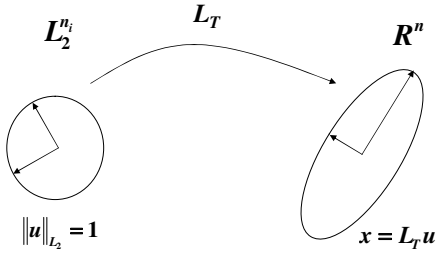

Figure 3: Controllability Ellipsoid

A dual approach considers the state-to-output coupling by using the concept of observability. Since only state and output are considered, let input $u = 0$. Denote the mapping of the initial state $x_0 \in \mathbb{R}^n$ to the output trajectory $y \in L_2^{n_o}[0,T]$ by ℓ_T, then

$$y(t) = (\ell_T x_0)(t) = Ce^{At} x_0. \tag{3.8}$$

We can visualize ℓ_T as a mapping of the unit ball in $\mathbb{R}^n$ to an ellipsoid (at most n-dimensional) in $L_2^{n_o}[0,T]$, called the observability ellipsoid (see

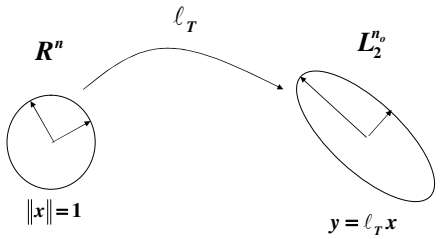

Figure 4: Observability Ellipsoid

Fig. 4). The principal axes of the ellipsoid indicate the degree of coupling between the state and the output signal. The lengths of the principal axes of the observability ellipsoid are the singular values of ℓ_T, or, equivalently, the square root of the eigenvalues of

$$Q(T) = \ell_T \ell_T^* \tag{3.9}$$

where ℓ_T^* is the operator adjoint of ℓ_T and $Q(T)$ is an $n \times n$ positive semi-definite matrix, called the observability gramian (at time T). The observability gramian may be calculated from a linear matrix differential equation

$$\dot{Q}(t) = A^T Q(t) + Q(t) A^T + C^T C, \quad Q(0) = 0_{n\times n}. \tag{3.10}$$

The solution can also be written as a matrix integral:

$$Q(T) = \int_0^T e^{A^T t} C^T C e^{At}\, dt. \tag{3.11}$$

For stable systems, $Q(t)$ converges to a steady state matrix, Q, as $t \to \infty$, which solves the following Lyapunov equation (dual to (3.6):

$$A^T Q + QA + C^T C = 0. \tag{3.12}$$

The solution can also be written as an integral:

$$Q = \int_0^\infty e^{A^T t} C^T C e^{At}\, dt. \tag{3.13}$$

The solution of the Lyapunov equations has been well studied in the literature. Two of the most popular methods are the Bartles and Stewart algorithm [21] and the Hammarling algorithm [22]. These algorithms involve the reduction of the system matrix A to the triangular form via a Schur decomposition, which requires $\mathcal{O}(n^3)$ operations even when A is sparse. This computation cost is acceptable for small to medium scale problems ($n \leq 400$), but is obviously prohibitive for large systems. We will discuss reducing this cost through the use of approximate iterative methods in Section 4.

Once the gramians are found, we are now able to construct a reduced order model by only retaining the states that are strongly coupled to the input or output. For controllability, let the eigenvalue decomposition of P be given by (note that P is symmetric positive semidefinite):

$$P = T_c^T \Sigma_c T_c \tag{3.14}$$

where Σ_c is diagonal and contains the eigenvalues of P sorted in reverse order, and the columns of T_c are the eigenvectors. The transformed system $(T_c A T_c^T, T_c B, C T_c^T, D)$ has the controllability gramian Σ_c. Now partition Σ_c to

$$\Sigma_c = \begin{bmatrix} \Sigma_{c_1} & 0 \\ 0 & \Sigma_{c_2} \end{bmatrix}$$

with Σ_{c_1} containing the dominant eigenvalues and Σ_{c_2} the remainder. Partitioning A, B, C accordingly, the state equation in the transformed coordinate becomes

$$\begin{bmatrix} \dot{z}_1 \\ \dot{z}_2 \end{bmatrix} = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} + \begin{bmatrix} B_1 \\ B_2 \end{bmatrix} u$$
$$y = \begin{bmatrix} C_1 & C_2 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} + Du.$$

A reduced order model may be obtained by direct truncation, i.e., assume z_2 is small. The corresponding state space representation is (A_{11}, B_1, C_1, D). Another way to obtain a reduced order model is through singular perturbation [23] by assuming that z_2 converges to a steady state much faster than z_1. In this case, set $\dot{z}_2 = 0$ to obtain

$$z_2 = -A_{22}^{-1}(A_{21} z_1 + B_2 u).$$

Substitute back into the $\dot{z}_1$ equation to obtain the reduced order system:

$$\begin{aligned} \dot{z}_1 &= (A_{11} - A_{12}A_{22}^{-1}A_{21}) z_1 + (B_1 - A_{12}A_{22}^{-1}B_2) u \\ y &= (C_1 - C_2 A_{22}^{-1} A_{21}) z_1 + (D - C_2 A_{22}^{-1} B_2) u. \end{aligned} \tag{3.15}$$

Note that direct truncation matches the high frequency gains between the full order and reduced order models (i.e., D), and the singularly perturbation model matches the DC gains.

Similarly, we can perform eigen-decomposition on Q as

$$Q = T_o^T \Sigma_o T_o. \tag{3.16}$$

The eigenvalues Σ_o can be partitioned into the dominant and discardable portions. The corresponding partition of (A, B, C, D) can then be used to generate a reduced order system by using either truncation or singular perturbation.

The reduced order model using the input-to-state or state-to-output coupling (through controllability or observability, respectively) is intuitively appealing but unfortunately is coordinate dependent. Let (A, B, C, D) and $(T^{-1}AT, T^{-1}B, CT, D)$ be two state space realizations of the same input/output system, and (P, Q) and $(\overline{P}, \overline{Q})$ the corresponding controllability and observability gramians. Then

$$\overline{P} = T^{-1} P T^{-T}, \quad \overline{Q} = T^T Q T. \tag{3.17}$$

In general, P and $\overline{P}$ (resp. Q and $\overline{Q}$) have different eigenvalues (unless T is orthogonal, i.e., $T^{-1} = T^T$). Therefore, model reduction based only on the controllability gramian (resp. observability gramian) information would yield different reduced order models for the same system by just changing the state representation. In particular, controllability may be increased in certain state directions (e.g., through scaling) at the sacrifice of the observability, and vice versa.

A solution of the coordinate dependence problem is to consider both controllability and observability at the same time. Such a transformation may be found through the eigen-decomposition of PQ:

$$\Sigma = T^{-1} P Q T \tag{3.18}$$

where the diagonal matrix Σ contains the eigenvalues of PQ sorted in descending order and T is the corresponding eigenvector matrix. In this coordinate system, both controllability and observability gramians are Σ:

$$\overline{P} = T^{-1} P T^{-T} = \overline{Q} = T^T Q T = \Sigma, \tag{3.19}$$

and the system is said to be "balanced" (between controllability and observability) [24]. Balancing is usually performed for stable systems using the solutions of the corresponding Lyapunov equations (3.6) and (3.12). In this case, the diagonal entries of Σ are called the *Hankel singular values* of the system

and they are invariant with respect to the coordinate transformation. Hankel singular values describe the degree that a given state contributes to the input-output energy flow of the system. States with small Hankel values are both weakly controllable and weakly observable, and can be removed from the system (through truncation, called balanced truncation, or singular perturbation). Therefore, systems that show a rapid decline in Hankel singular values are easily approximated by a reduced order system. A sharp decrease in the Hankel singular values can indicate a good point to truncate the model [26]. Balanced truncation is the main model reduction tool examined in the paper, and we will later examine approximate, but computationally attractive, solution to the system gramians and their use in balancing.
The most common method of finding the balanced coordinate is the square root method first proposed in [25]. First find the square roots of P and Q:

$$P = Z_P Z_P^T, \quad Q = Z_Q Z_Q^T. \tag{3.20}$$

The $n \times n$ square root matrices are known as the *Cholesky factors* of the gramians. They are upper triangular, and always exist since the gramians are positive semi-definite. Next perform a singular value decomposition:

$$Z_P^T Z_Q = U \Sigma V^T \tag{3.21}$$

where U, V are orthogonal and Σ is diagonal. The next step is to form the coordinate transformation matrix which requires the system to be controllable and observable, i.e., P and Q are positive definite. If this is not the case (which is frequently the case for large scale systems due to numerical accuracy), a model reduction may first need to be performed to remove the nearly uncontrollable and nearly unobservable subsystems (by using truncation or singular perturbation using controllability or observability alone). If P and Q are both positive definite, then Z_P and Z_Q are both invertible. Define

$$T_1 = Z_P U \Sigma^{-\frac{1}{2}}, \quad T_2 = Z_Q V \Sigma^{-\frac{1}{2}}. \tag{3.22}$$

It follows that $T_1^{-1} = T_2^T$. Note, however, neither T_1 nor T_2 require explicit matrix inversion. Using T_1 as the transformation matrix (and use T_2^T instead of T_1^{-1}), the new state space representation is

$$(A_b, B_b, C_b, D_b) = (T_2^T A T_1, T_2^T B, C T_1, D). \tag{3.23}$$

It can be verified that the controllability and observability gramians of this system both equal to Σ.

The Hankel singular values also define an error bound for balanced truncation. For a system G with Hankel singular values $(\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_n)$, the approximation error for a balanced truncation reduced order model of order k, $\hat{G}_k(s)$, satisfies the inequality

$$2(\sigma_{k+1} + ... + \sigma_n) \geq \|G - \hat{G}\|_{H_\infty} \geq \sigma_{k+1}. \tag{3.24}$$

Balanced truncation tends to match gain well but sometimes poorly in phase. A related approach called balanced stochastic truncation has been proposed for square systems [27,28] which balances the spectral factor of $G(s)G^T(-s)$ instead of G itself. This approach tends to approximate the phase better and has a guaranteed relative error bound, i.e., a bound on $\left\|(G - \hat{G})G^{-1}\right\|$. We did not compare this approach in our numerical study in this paper.

3.3. Optimal Hankel Approximation

Balanced realization chooses a state coordinate based on its contribution to the input/output energy flow. The logical next step is to consider the input/output map without explicitly considering the internal state of the system. We first define a linear operator, called the *Hankel operator*, that maps the past input, $L_2^{n_i}(-\infty, 0]$, to the future output, $L_2^{n_o}[0, \infty)$:

$$\Gamma = \Psi_o \Psi_c. \tag{3.25}$$

where $\Psi_c : L_2^{n_i}(-\infty, 0] \to \mathbb{R}^n$ maps the past input to the initial state, and $\Psi_o : \mathbb{R}^n \to L_2^{n_o}[0, \infty)$ maps the initial state to the future output. It can be shown that the induced norm, $\|\Gamma\|$, is the largest Hankel singular value of the system. We can now pose the model reduction as a minimization problem in terms of the Hankel norm of the approximation error (this is called the optimal Hankel norm approximation problem):

$$\min_{\text{order } k \ \hat{G}} \|G - \hat{G}\|_H. \tag{3.26}$$

The solution of this problem is given [29], and can be readily implemented. We also consider this method in the numerical study in the next section. The optimal Hankel norm approximation method gives a tighter guaranteed error bound in terms of the H_∞ norm:

$$\|G - \hat{G}\|_{H_\infty} \leq (\sigma_{k+1} + ... + \sigma_n). \tag{3.27}$$

3.4. Discrete Time Systems

The model reduction approach described above can also be applied to discrete time LTI systems. Discrete time system description arises for computation reasons (as an approximation to continuous time systems – se Section 4) or due to sampled data implementation (zero-order-hold digital-to-analog input and sampled analog-to-digital output).
Consider a discrete time system in a state space representation:

$$\begin{aligned} x(k+1) &= A_d x(k) + B_d u(k) \\ y(k) &= C_d x(k) + D_d u(k), \end{aligned} \tag{3.28}$$

where k is a non-negative integer denoting the time horizon.
As in the continuous time case, controllability can be defined as the mapping, L_N, from an input sequence $u_N = \{u(k) : 0 \le k \le N-1\}$ to a terminal state $x(N)$ (starting from the origin):

$$\begin{aligned} L_N u_N &= \sum_{i=0}^{N-1} A_d^{N-i-1} B_d u(i) \qquad (3.29) \\ &= [A_d^{N-1} B_d, \ldots, B_d] \begin{bmatrix} u(0) \\ \vdots \\ u(N-1) \end{bmatrix}. \end{aligned}$$

The controllability gramian can also be similarly defined as

$$P_N = L_N L_N^T. \tag{3.30}$$

One can visualize L_N as the mapping of a unit $\ell_2^{n_i}[0, N-1]$ ball to an $\mathbb{R}^n$ ellipsoid, the controllability ellipsoid, with the principal axes given by the eigenvectors of P_N and their lengths given by the square roots of the eigenvalues of P_N. The controllability ellipsoid captures the degree of coupling between the input and the state. In the case that the controllability ellipsoid is degenerate (zero length) in certain state direction, then those states cannot be affected by the input (i.e., they are uncontrollable) and can be removed from the system description.
The gramian, P_N, also satisfies the discrete time Lyapunov equation:

$$A_d P_N A_d^T - P_{N+1} + B_d B_d^T = 0 \tag{3.31}$$

and can also be solved explicitly through a finite sum:

$$P_N = \sum_{i=0}^{N-1} A_d^i B_d B_d^T {A_d^i}^T. \tag{3.32}$$

If A_d is stable (i.e., all eigenvalues within the unit circle), then P_N converges to a steady state solution, P, as $N \to \infty$. In this case, P satisfies the steady state Lyapunov equation

$$A_d P A_d^T - P + B_d B_d^T = 0 \tag{3.33}$$

and can be evaluated through an infinite sum:

$$P = \sum_{i=0}^{\infty} A_d^i B_d B_d^T {A_d^i}^T. \tag{3.34}$$

As a dual concept, consider an unforced system (i.e., $u \equiv 0$) with the initial condition $x(0)$ generating an output sequence $y_N = \{y(k) : k = 0, \ldots, N-1\}$. The mapping from $\mathbb{R}^n$ to $\ell_2^{n_o}[0, N-1]$ is then

$$\ell_N x(0) = \begin{bmatrix} C_d \\ C_d A_d \\ \vdots \\ C_d A_d^{N-1} \end{bmatrix} x(0). \tag{3.35}$$

Define the observability gramian as

$$Q_N = \ell_N^T \ell_N. \tag{3.36}$$

We can now visualize ℓ_N as the mapping of a unit $\mathbb{R}^n$ ball to an (at most) n-dimensional $\ell_2^{n_o}[0, N-1]$ ellipsoid, the observability ellipsoid, with the principal axes given by the eigenvectors of Q_N and their lengths given by the square roots of the eigenvalues of Q_N. The observability ellipsoid captures the degree of coupling between the state and the output. In the case that the observability ellipsoid is degenerate in certain state direction, then those states do not generate any output and can be removed from the system description.
The gramian, Q_N, also satisfies the discrete time Lyapunov equation:

$$A_d^T Q_N A_d - Q_{N+1} + C_d^T C_d = 0 \tag{3.37}$$

and can also be solved explicitly through a finite sum:

$$Q_N = \sum_{i=0}^{N-1} {A_d^i}^T C_d^T C_d A_d^i. \tag{3.38}$$

If A_d is stable, then Q_N converges to a steady state solution, Q, as $N \to \infty$. In this case, Q satisfies the steady state Lyapunov equation

$$A_d^T Q A_d - Q + C_d^T C_d = 0 \tag{3.39}$$

and can be evaluated through an infinite sum:

$$Q = \sum_{i=0}^{\infty} {A_d^i}^T C_d^T C_d A_d^i. \tag{3.40}$$

The gramians, P and Q, may be used in exactly the same way as in the continuous time systems to reduce the system order. For balanced truncation, we first perform an eigen-decomposition of PQ (in any coordinate):

$$\Sigma = T^{-1}PQT, \tag{3.41}$$

where the diagonal matrix Σ contains the eigenvalues of PQ (Hankel singular values for the discrete time system) sorted in descending order and T is the corresponding eigenvector matrix. Then by using T as the coordinate transformation, the transformed system, $(T^{-1}A_dT, T^{-1}B_d, C_dT, D_d)$, has identical controllability and observability gramians which are both Σ. The states corresponding to small values of Σ may be truncated as in Section 3.2 to obtain the reduced order model.

3.5. Example

3.5.1. Comparison of Model Reduction Methods

In this section, we examine the performance of the model reduction techniques discussed above applied to a problem of moderate dimension (100 modes or $n = 200$). The purpose is to demonstrate the effectiveness of model reduction for generalized second-order models, and to determine the most effective type of model reduction, which will then be applied to the large-scale problem.
The system under consideration here has been randomly generated, and has a non-monotonically decaying frequency response. We apply three model reduction methods that have been discussed: modal truncation, balanced truncation, and optimal Hankel norm approximation. The performance of these methods is compared by examining the frequency response plots of the original system and the approximated ones.

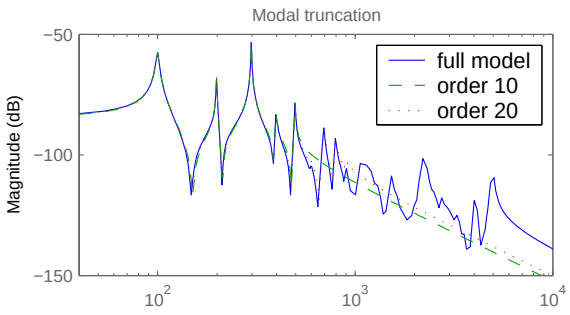

Figure 5: Graphical progression of modal truncation

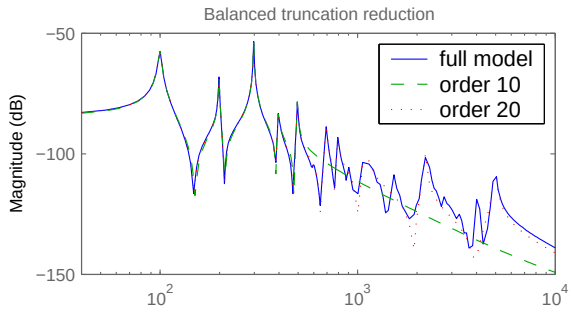

Figure 6: Graphical progression of balanced truncation

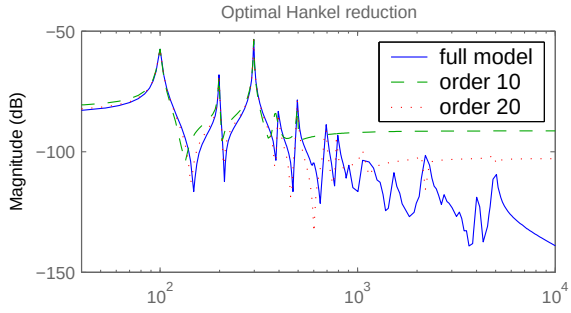

Figure 7: Graphical progression of optimal Hankel-norm approximation

Fig. 5-7 show the frequency response comparison between the full-order model, the modal truncation models, the reduced order balanced truncation models, and optimal Hankel norm reduction models. Balanced truncation retains the most dominant system resonant modes in the frequency response, while modal truncation retains the lowest frequency modes irrespective of their contributions. Hankel norm reduction matches the dominant modes reasonably well but performs poorly at high frequency since the contribution to the error norm is small. Some of the undesirable features of modal and Hankel norm reduction methods may be corrected by selecting modes based on their peak magnitudes or through frequency dependent weighting, but balanced truncation is the overall algorithm of choice since it captures the dominant input/output frequency response, provides an error bound, and preserves system stability.

3.5.2. Maximum Output Prediction with Reduced Order Models

In this section, we present an application of model reduction to the determination of the value and location of the maximum strain in a structure. This information can be used to ensure that a critical yield stress is not exceeded or to optimize the placement of sensors to collect strain data for control purposes. We will show that a reduced order model can be used to predict the value and location of the maximum strain while reducing computational expense. Our motivating example is a piezoelectric composite beam, with a force input applied at a randomly-selected location along its length. The model has been discretized to 400 nodes, giving the full-order model 800 states. Therefore, there are 399 possible locations for the maximum strain.

To apply the model reduction methods presented earlier, we can consider 399 systems, each with a distinct single output corresponding to the strain at a node. Alternatively, we can consider a single system with 399 outputs, and reduce the model only once, obtaining a model with 399 outputs. The first approach has the greatest potential for model reduction, since less information is required to describe the input-output relationship, but the second is much more efficient computationally since it requires only one reduction. The time required to perform 399 individual reductions is much more than the time required to simply compute the full-order model, making the first approach impractical. We thus proceed with the multiple-output approach.

Thirty simulations were performed, in which the input was a bounded sinusoidal force function with random frequency content and amplitude. The force was applied at randomly selected points on the beam, and the full-order model was simulated to find the value and temporal and spatial locations of the maximum strain. Reduced order models with 4 to 200 states were then generated, and the predicted value and location of the maximum strain were found. The results of these predictions were compared with the actual value and location of maximum strain, and the quality of the approximation was then assessed. We define a "successful" prediction as one predicting the strain value within 5 percent, the location within ± 2 nodes, and the time within ± 10 time steps. Our simulation was run for 1000 time steps. The results are presented in graphical form in figure 8, which shows the percentage of tests giving successful approximations for a given reduced-order model size. It is evident that a reasonable prediction can be obtained using reduced order models of order 200 or less in most cases.

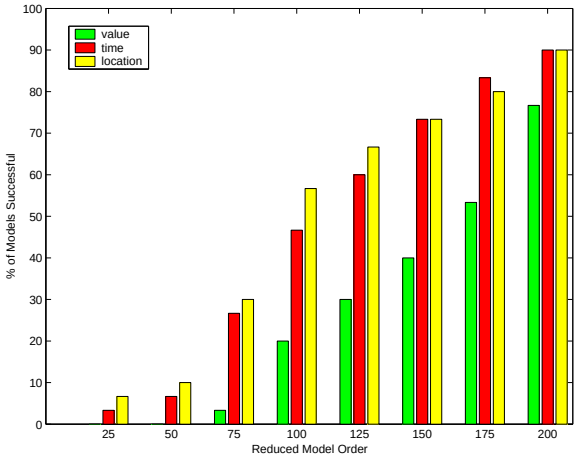

Figure 8: Results for 30 tests of maximum strain prediction

In almost all cases, the time and location were successfully predicted before the strain value itself. It is not known *a priori* when the reduced order model will successfully predict value, time, and location, but it was observed that once an accurate prediction had been reached, it remained accurate when we further increased the model order. Convergence of the maximum strain value was asymptotic, but the behavior of the convergence of location and time was not well-defined, often oscillating between several distinct values before converging.

Although we cannot calculate an analytical error bound for a problem of this type, we can define an *a posteriori* error bound as follows. The induced L_∞ norm in the time domain is the L_1 norm of the impulse response [19]:

$$\|g - \hat{g}\|_{L_1} = \frac{\|y - \hat{y}\|_{L_\infty}}{\|u\|_{L_\infty}} \tag{3.42}$$

where g and $\hat{g}$ are the impulse responses of the full-order and reduced order systems, y and $\hat{y}$ are the respective outputs, and u is the input. If the desired maximum output error bound is, ϵ, then a sufficient condition is

$$\|g - \hat{g}\|_{L_1} \leq \frac{\epsilon}{\|u\|_{L_\infty}}. \tag{3.43}$$

For a given input u, we can choose the order of the reduced model sufficiently high (and $\|g - \hat{g}\|_{L_1}$ sufficiently small), so that (3.43) is satisfied. The con-

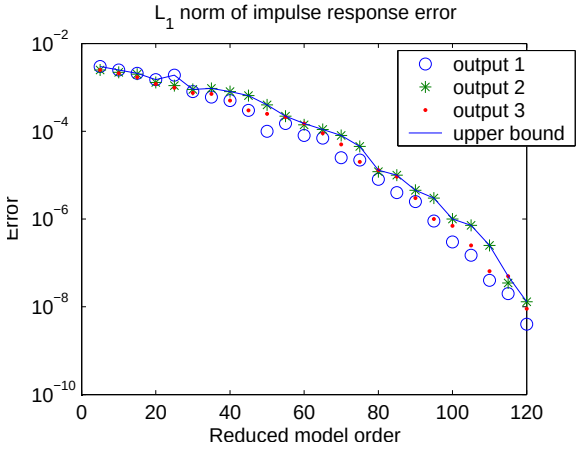

Figure 9: L1 norm of impulse response error for selected output locations

vergence of $\|g - \hat{g}\|_{L_1}$ for several output locations is shown in Fig. 9.

The benefit of model reduction can be seen in the computational time savings shown in Table 2. We incur a one-time cost of reduction (in terms of gramian computation and SVD needed in balancing), and after this we enjoy a significant savings for each iteration in terms of the solution of the time response. The cost to simulate an 200th order model is about $\frac{1}{16}$ of an 800th order model. So after 10 design iterations, the total computation time using the reduced order model is already less than that of the full order model.

4. APPROXIMATE SOLUTIONS OF GRAMIANS

We have shown that balanced truncation is an effective model reduction technique in the previous section. However, its application to large systems is limited by the computational load (of order n^3 for the gramian and SVD calculations needed in balanced transformation) and storage requirements. Additionally, the numerical implementation can become ill-conditioned for stiff systems (widely separated eigenvalues in A). In this section we present several methods that are more numerically efficient to *approximately* compute the system gramians.

For large systems with many state variables and relatively few inputs and outputs, typical of models arising from the finite element method, the Hankel singular values decay rapidly. This implies that the input-output energy coupling is dominated by just a few states. As a result, the gramians have low numerical ranks. We can exploit this fact by computing just the dominant portion of the gramians which can then be sued to calculate a reduced order model. If we can efficiently compute an approximate gramian that has eigenvectors that point roughly in the same state directions as the dominant eigenvectors of the actual gramian, then the approximate gramian will perform nearly as well as the actual one in the model reduction process. This low rank gramian approximation then takes the place of the solution of the full order Lyapunov equations, which is computationally prohibitive for the large-scale problem.

4.1. Discrete-Time Gramian Formulation

Instead of solving for the gramians in continuous-time, we will consider the solution a discrete-time system that has the same gramians. This process allows us to calculate the gramians using an infinite series instead of the integrals in (3.7) and (3.13).

Consider the following bilinear transformation that maps the imaginary axis (in the s domain) to the unit circle (in the z domain)

$$s = p\frac{(1-z)}{(1+z)} \tag{4.1}$$

where $p < 0$ is a shift parameter to be chosen. If the discrete time system is obtained through uniform time domain sampling of the continuous time system, then $p = -2f_s$ with f_s the sampling frequency. Substituting (4.1) into the continuous time transfer function (2.11), we obtain a discrete time transfer function

$$H_p(z) = C_p(zI - A_p)^{-1}B_p + D_p, \tag{4.2}$$

where

$$\begin{aligned} A_p &= (pI + A)^{-1}(pI - A) \\ B_p &= \sqrt{-2p}(pI + A)^{-1}B \\ C_p &= \sqrt{-2p}C(pI + A)^{-1} \\ D_p &= D - C(pI + A)^{-1}B. \end{aligned} \tag{4.3}$$

The corresponding Lyapunov equations for the discrete time controllability and observability gramians are

$$\begin{aligned} A_pPA_p^T - P + B_pB_p^T &= 0 \\ A_p^TQA_p - Q + C_p^TC_p &= 0. \end{aligned} \tag{4.4}$$

Note that for any $p < 0$, these equations are exactly the same as the continuous time Lyapunov

Model	Order	Red Time (s)	Time (1)(s)	Time (10)(s)	Time (100)(s)
Original	800	0	24	240	2400
Reduced	200	125	125+1.5	125+15	125+150

Table 2: Cost savings in maximum strain problem design cycles

equations (3.6) and (3.12). The solutions may be expressed as infinite sums instead of integrals in (3.7) and (3.13):

$$\begin{aligned} P &= \sum_{j=0}^{\infty} A_p^j B_p B_p^T {A_p^T}^j \\ Q &= \sum_{j=0}^{\infty} {A_p^T}^j C_p^T C_p {A_p}^j. \end{aligned} \quad (4.5)$$

Since A_p is stable (i.e., all eigenvalues within the unit circle), these series converge. A natural approximation of P and Q may then be found by truncating these series. We now examine various variations for solving for the approximate gramians efficiently.

4.2. Smith Method

The infinite series (4.5) may be truncated to generate the following kth order approximate gramians:

$$\begin{aligned} P_k &= \sum_{j=0}^{k-1} A_p^j B_p B_p^T {A_p^T}^j \\ Q_k &= \sum_{j=0}^{k-1} {A_p^T}^j C_p^T C_p {A_p}^j. \end{aligned} \quad (4.6)$$

These sums may be iteratively computed:

$$\begin{aligned} P_j &= A_p P_{j-1} A_p^T + B_p B_p^T, \quad P_0 = 0 \\ Q_j &= A_p^T Q_{j-1} A_p + C_p^T C_p, \quad Q_0 = 0, \end{aligned} \quad (4.7)$$

where $j = 1, \ldots, k$. This iterative solution for the approximate gramians is known as the Smith method. The computational cost is $\mathcal{O}(n^3)$ for fully populated A, and $\mathcal{O}(n^2)$ for tridiagonal A.

4.3. ADI Iteration

The Alternating Direction-Implict (ADI) algorithm [30, 31] is a generalization of the Smith method by using distinct shift parameters $p_1, p_2, ..$:

$$\begin{aligned} P_j &= A_{p_j} P_{j-1} A_{p_j}^T + B_{p_j} B_{p_j}^T, \quad P_0 = 0 \\ Q_j &= A_{p_j}^T Q_{j-1} A_{p_j} + C_{p_j}^T C_{p_j}, \quad Q_0 = 0, \end{aligned} \quad (4.8)$$

where A_{p_j}, B_{p_j}, C_{p_j} are the transformed state space matrices as defined in (4.3) with p replaced by the jth shift parameter p_j. When a fixed number of shift parameters are used, they are recycled in the iterations. Referencing (4.7), the ADI iteration simplifies to the Smith method when only one shift parameter is used. However, when multiple shifts are used, the convergence rate is typically faster than the Smith method.

The iterations in (4.8) may be split into two steps to gain efficiency:

$$\begin{aligned} (A + p_j I) P_{temp} &= -BB^T - P_{j-1}(A^T - p_j I) \\ (A + p_j I) P_j &= -BB^T - P_{temp}^T (A^T - p_j I) \\ (A^T + p_j I) Q_{temp} &= -C^T C - Q_{j-1}(A - p_j I) \\ (A^T + p_j I) Q_j &= -C^T C - Q_{temp}^T (A - p_j I) \\ P_0 = Q_0 &= 0 \end{aligned} \quad (4.9)$$

where P_{temp} and Q_{temp} are intermediate matrices. Computationally, each ADI iteration in (4.9) involves two matrix-matrix products, and two matrix-matrix solves. For a full matrix A, a matrix-matrix solve has computational cost $\mathcal{O}(n^3)$, which is impractical for large n. To reduce the computational cost to $\mathcal{O}(n^2)$, a general matrix $\mathbf{A}$ must be made tridiagonal.

4.4. Cyclic Smith Method

The cyclic Smith method combines the ADI and Smith methods by first applying the ADI method for J steps (using all the shift parameters) and then using Smith method to generate the approximate gramians. To show the algorithm, we consider the controllability case only. First write (4.4) with $p = p_J$ (which is equivalent to (3.6)):

$$P = A_{p_J} P A_{p_J}^T + B_{p_J} B_{p_J}^T.$$

Then substitute for P in the right hand side by using

$$P = A_{p_{J-1}} P A_{p_{J-1}}^T + B_{p_{J-1}} B_{p_{J-1}}^T$$

to obtain

$$P = A_{p_J}(A_{p_{J-1}} P A_{p_{J-1}}^T + B_{p_{J-1}} B_{p_{J-1}}^T) A_{p_J}^T + B_{p_J} B_{p_J}^T.$$

Repeat this process to obtain

$$P = \Phi_{0,J} P \Phi_{0,J}^T + P_J \tag{4.10}$$

where

$$\Phi_{k,\ell} := \prod_{i=k+1}^{\ell} A_{p_i}$$

and

$$P_J = \sum_{j=1}^{J} \Phi_{j,J} B_{p_j} B_{p_j}^T \Phi_{j,J}^T$$

is just the Jth iterate of the ADI iteration (4.8). Eq. (4.10) is of the same form as (4.4), therefore, we may apply the Smith method to find an approximate solution:

$$P_j^{(CS)} = \Phi_{0,J} P_{j-1}^{(CS)} \Phi_{0,J}^T + P_J,\ P_0^{(CS)} = 0, \tag{4.11}$$

where $j = 1, \ldots, k$, for the k-term approximation of the infinite series expansion. For the observability gramian, a similar propagation may be used

$$Q_j^{(CS)} = \Phi_{0,J}^T Q_{j-1}^{(CS)} \Phi_{0,J} + Q_J,\ Q_0^{(CS)} = 0, \tag{4.12}$$

where Q_J is the Jth ADI iterate. The computational cost is again $\mathcal{O}(n^3)$ for fully populated A, and $\mathcal{O}(n^2)$ for tridiagonal A.
The advantage of the Cyclic Smith method lies in faster convergence than the Smith method (due to the multiple shifts) while avoiding using a large number of shift parameters.

4.5. Shift Parameter Selection

The convergence of the Smith, cyclic Smith, and ADI algorithms depend on the selection of the shift parameters, p_j (user-selected real numbers or complex conjugate pairs with negative real parts). To increase the speed of convergence of these algorithms, p_j should be chosen so that the eigenvalues A_{p_j} have small magnitudes. The eigenvalues of A_{p_j} is related to the eigenvalues of A by

$$\lambda_i(A_{p_j}) = \frac{p_j - \lambda_i(A)}{p_j + \lambda_i(A)}. \tag{4.13}$$

The selection of the shift parameters (for ADI and cyclic Smith cases) may then be posed as an optimization problem of choosing $p_1, p_2, \ldots, p_J$ to minimize the largest eigenvalue of $\Phi_{0,J}$:

$$\min_{p_1,\ldots,p_J} \max_{\lambda(A)} \left| \prod_{j=1}^{J} \frac{(p_j - \lambda(A))}{(p_j + \lambda(A))} \right|. \tag{4.14}$$

For the Smith method, only one parameter needs to be chosen. If the eigenvalues of A are known, and if p_j's are chosen to be the eigenvalues of A, then the ADI algorithm will produce the exact solution of the gramians in n step. Of course, the goal is to obtain an approximate solution in a much smaller number of iterations, so the number of shift parameters is in general much smaller, i.e., $J << n$.
If the eigenvalues of A are all real, the solution to (4.14) is known, and the optimal parameters may be readily generated. However, second-order FEM models typically have numerous complex eigenvalues, with small real parts. For this case, the problem has no known closed-form solution. Various suboptimal solutions have been proposed [32–34].

4.6. Low Rank Algorithms

The iterative methods presented so far avoid the costly solution of Lyapunov equations in the computation of gramians. However, their application to large scale systems is inherently limited due to the computation and storage requirements in propagating the *full* $n \times n$ system gramians at each iteration. This section discusses the so-called *low rank* methods which propagate the Cholesky factor of the gramian instead of the full gramian [35].

4.6.1. Low-Rank ADI

Low-rank ADI (LR-ADI), proposed in [36], propagates the Cholesky factor of the gramians in ADI instead of the full gramian matrix. As a result, it has reduced computational and storage requirements. Let P_j be the jth ADI iterate. Since $P_j \geq 0$, it can be factored as

$$P_j = Z_{P_j} Z_{P_j}^T.$$

The ADI iteration (4.8) may be written as

$$\begin{aligned} Z_{P_j} Z_{P_j}^T &= A_{p_j} Z_{P_{j-1}} Z_{P_{j-1}}^T A_{p_j}^T + B_{p_j} B_{p_j}^T \\ &= \left[\ A_{p_j} Z_{P_{j-1}} \quad B_{p_j}\ \right] \left[\ A_{p_j} Z_{P_{j-1}} \quad B_{p_j}\ \right]^T. \end{aligned}$$

We can now update Z_{P_j} instead of P_j:

$$Z_{P_j} = \left[\ A_{p_j} Z_{P_{j-1}} \quad B_{p_j}\ \right],\ Z_{P_0} = 0. \tag{4.15}$$

With a little algebra, it can be shown that

$$Z_{P_J} = \left[\ B_{p_J} \quad S_{J-1} B_{p_J} \quad \cdots \quad \prod_{j=1}^{J-1} S_j B_{p_J}\ \right] \tag{4.16}$$

where

$$S_i = \sqrt{\frac{p_i}{p_{i+1}}} (I - (p_{i+1} + p_i)(A + p_i I)^{-1}).$$

The following iterative update may then be used to generate Z_{P_J} (the indexing has been reversed for convenience):

$$\begin{aligned} z_{j+1} &= S_j z_j,\ z_1 = B_{p_1}, \\ S_j &= \sqrt{\frac{p_{j+1}}{p_j}}(I - (p_{j+1}+p_j)(A+p_{j+1}I)^{-1}), \\ Z_{P_j}^{(ADI)} &= \left[\begin{array}{cc} Z_{P_{j-1}}^{(ADI)} & z_j \end{array}\right],\ Z_{P_1}^{(ADI)} = z_1. \end{aligned} \quad (4.17)$$

Similarly, for the observability gramian, we have

$$\begin{aligned} z_{j+1} &= S_j^T z_j,\ z_1 = C_{p_1}^T, \\ Z_{Q_j}^{(ADI)} &= \left[\begin{array}{cc} Z_{Q_{j-1}}^{(ADI)} & z_j \end{array}\right],\ Z_{Q_1}^{(ADI)} = z_1. \end{aligned} \quad (4.18)$$

The iteration terminates when $\|z_j\|$ becomes sufficiently small.

Each iteration of LR-ADI requires only matrix-vector solves, instead of the matrix-matrix products used in the normal ADI method. A matrix-vector solve has cost $\mathcal{O}(n^2)$ if A is full, and $\mathcal{O}(n)$ if A is sparse. Therefore, the LR-ADI algorithm has cost $\mathcal{O}(n)$ if A is tri-diagonal or sparse, and $\mathcal{O}(n^2)$ if A is full. Each LR-ADI iteration adds a number of columns to $Z_{P_j}^{(ADI)}$ and $Z_{Q_j}^{(ADI)}$ corresponding to the number of inputs. LR-ADI becomes most advantageous when the iterations terminate with a small number of columns, therefore saving both storage and computation. There could be further savings if only an orthonormal basis is saved in each iteration.

4.6.2. Low-Rank Cyclic Smith

In a manner similar to the LR-ADI algorithm, we can formulate a low-rank cyclic Smith (LR-CS) method to reduce computation and storage requirements [38]. Consider the cyclic Smith iteration (4.11). Substitute the Cholesky factorizations for P_J, the Jth ADI iterate, and $P_j^{(CS)}$, we get

$$Z_{P_j}^{(CS)} Z_{P_j}^{(CS)^T} = \Phi_{0,J} Z_{P_{j-1}}^{(CS)} Z_{P_{j-1}}^{(CS)^T} \Phi_{0,J}^T + Z_{P_J}^{(ADI)} Z_{P_J}^{(ADI)^T}. \quad (4.19)$$

We can now just update the Cholesky factor instead of the full gramian

$$Z_{P_J}^{(CS)} = \left[\begin{array}{cc} \Phi_{0,J} Z_{P_{j-1}}^{(CS)} & Z_{P_J}^{(ADI)} \end{array}\right]. \quad (4.20)$$

This may be written in an alternate and more efficient update:

$$\begin{aligned} z_{j+1} &= \Phi_{0,J} z_j,\ z_0 = Z_{P_J}^{(ADI)} \\ Z_{P_j}^{(CS)} &= \left[\begin{array}{cc} Z_{P_{j-1}}^{(CS)} & z_j \end{array}\right],\ Z_{P_0}^{(CS)} = z_0. \end{aligned} \quad (4.21)$$

Similarly, for the observability gramian, we have the following iteration:

$$\begin{aligned} z_{j+1} &= \Phi_{0,J}^T z_j,\ z_0 = Z_{Q_J}^{(ADI)} \\ Z_{Q_j}^{(CS)} &= \left[\begin{array}{cc} Z_{Q_{j-1}}^{(CS)} & z_j \end{array}\right],\ Z_{Q_0}^{(CS)} = z_0. \end{aligned} \quad (4.22)$$

The iterations terminate when $\|z_j\|$ is sufficiently small. Similar to the LR-ADI case, since $Z_{P_j}^{(CS)}$ and $Z_{Q_j}^{(CS)}$ in general have low column ranks than the dimension of the system, the LR-CS method has less computational (order $\mathcal{O}(n^2)$ for fully populated A and $\mathcal{O}(n)$ for tridiagonal A) and memory storage requirements.

5. APPROXIMATE BALANCE TRANSFORMATION

Once the approximate gramians are found, we can use them to generate a reduced order model. The square root method presented in Section 3.2 can be directly extended to use the approximate Cholesky factors obtained by using the LR-ADI or LR-CS methods [11,36]. Let the approximate Cholesky factors of the controllability and observability gramians be $\hat{Z}_P \in \mathbb{R}^{n\times k}$ and $\hat{Z}_Q \in \mathbb{R}^{n\times k}$ which are full column rank matrices (note that there are possibly many more rows than columns and, for simplicity, we assume that the matrices have the same number of columns). Next perform an SVD on the following $k \times k$ matrix:

$$\hat{Z}_P^T \hat{Z}_Q = \hat{U}\hat{\Sigma}\hat{V}^T. \quad (5.1)$$

Now define the transformation matrices

$$\hat{T}_1 = \hat{Z}_P \hat{U}\hat{\Sigma}^{-\frac{1}{2}}, \quad \hat{T}_2 = \hat{Z}_Q \hat{V}\hat{\Sigma}^{-\frac{1}{2}}. \quad (5.2)$$

The reduced order system may then be readily obtained:

$$(\hat{A}, \hat{B}, \hat{C}, \hat{D}) = (\hat{T}_2^T A \hat{T}_1, \hat{T}_2^T B, C\hat{T}_1, D). \quad (5.3)$$

The low-rank methods presented earlier can produce $\hat{Z}_P$ and $\hat{Z}_Q$ directly at reduced computation and storage needs as compared to the solution of the full order gramians. If $k << n$, the SVD (an $\mathcal{O}(n^3)$ operation) will also provide considerable savings.

The quality of the reduced order model depends on the how well the low rank Cholesky factors approximate factors of the actual gramians. However, the analytic error bound (3.24) no longer holds. Some error bounds have recently been developed [39] to account for the approximation error as well. Furthermore, the full order balance truncation preserves the

stability of the full order system. With the approximate gramians, this is no longer true and the unstable modes will need to be removed. Other model reduction methods using the approximate Cholesky factors have also been proposed [11].

6. NUMERICAL EXPERIMENT

In this section, we use a fixed-free composite piezoelectric beam presented in [40] to illustrate the model reduction methods discussed in this paper. The beam has been spatially discretized into 450 nodes, resulting in a full-order model with 900 states. The input is chosen as a force applied to node 250, and the output is the strain measured at node 2 (the beam root). The frequency response is shown in Fig. 10. A complete derivation of the model appears in [40] and will not be repeated here. All simulations are performed using MATLAB 6.5, running on a Pentium 4 2.0 GHz PC, with 512 MB of RAM. The approximate gramian computation and balancing routines are drawn from the Lyapack [14] software library.

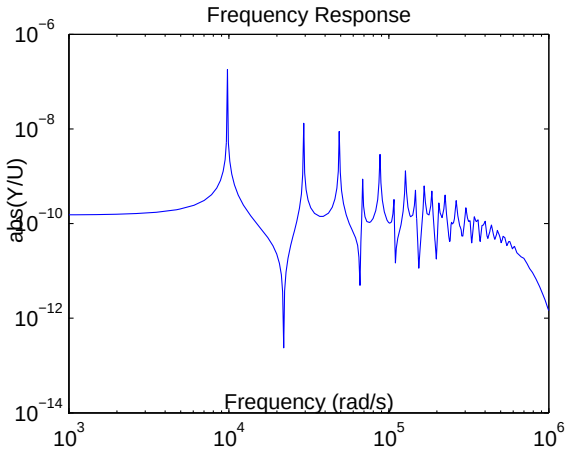

Figure 10: Frequency response of full-order model

Extensive numerical comparison between all the algorithms for gramian approximation have been conducted in [37]. The best performance is obtained by using the LR-ADI algorithm for gramian approximation together with the low-rank square root method for balancing. We will only present the results related to the LR-ADI method here.

6.1. Performance of LR-ADI and LR-Square-Root Methods

The parameters that need to be selected in the model reduction are the number of iterations and the order of the reduced system. In the case of exact balanced truncation, we know the $\mathcal{H}_\infty$ error bound *a priori* and can determine the required model order to achieve the required error tolerance from (3.24). In the approximated-gramian case, we do not have a guaranteed error bound. Therefore, we supply a "requested" model order, and the low-rank square root balancing algorithm produces a reduced order model of size no greater than requested order, and possibly less. A less-than-requested model order can occur when the approximated gramian is of insufficient order. Furthermore, since stability is not preserved under approximate balancing, there may be unstable eigenvalues in the resulting model which will need to be removed, resulting in a lower order model. The presence of unstable states was also noted by other researchers in [11, 38]. Fig. 11 shows the resulting model order for three values of requested model order, as a function of the LR-ADI iterations. Note that the 40th order model attains its full size after about 200 iterations of LR-ADI, meaning that after 200 iterations the gramians have numerical rank sufficient to generate a 40th order model. The 80th order model has a similar behavior, reaching its full size after about 500 iterations. For the 120th order model, the gramian has insufficient rank even after 600 iterations.

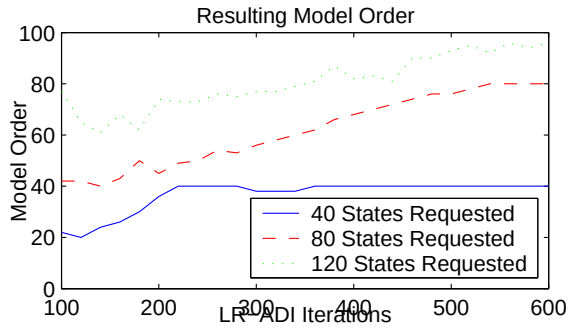

Figure 11: Resulting model order for given requested orders

We next assess the model reduction error as a function of the iteration in LR-ADI. The requested order is chosen to be 120. The $\mathcal{H}_2$ and $\mathcal{H}_\infty$ norms and time-domain L_2 and L_∞ norms of the impulse response of the error system are shown in Fig. 12. These rep-

resent the best results obtained after testing several different shift parameter sequences. Note that for the most part, the error norms decay monotonically in iterations.

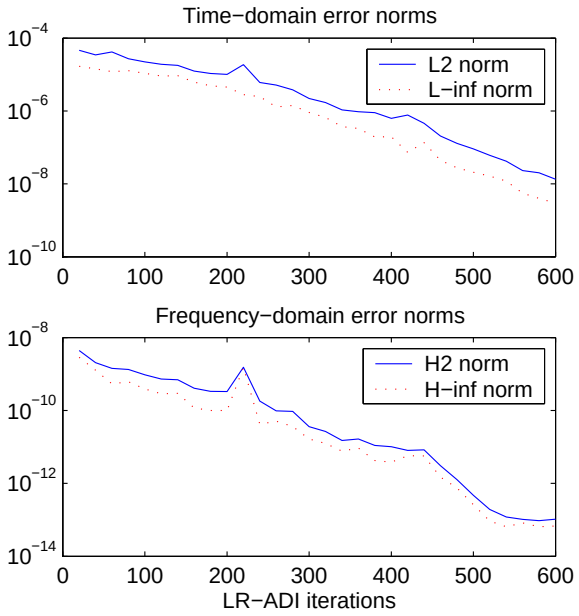

Figure 12: Model reduction error metrics - Beam Model

Fig. 13 shows the error norms as a function of the model order for both the exact and approximate balance truncation methods. We have observed that occasionally the approximate balancing method produces *better* approximations to the original system than the exact method. This is due to the numerical inaccuracies present in the solution of the Lyapunov equations, especially for high order stiff systems.

Fig. 14 shows the execution time of LR-ADI and the LR-square root algorithms. We have used the fully populated A in all the computations. Therefore, the computation is linear in k and quadratic in n. If tridiagonalization is first performed, then the computation load grows linearly in n. The computation load comparison is summarized in Table 3.

Structure	Exact	ADI	LR-ADI
Full $\mathbf{A}$	$\mathcal{O}(n^3)$	$\mathcal{O}(n^3)$	$\mathcal{O}(n^2)$
Tridiagonal $\mathbf{A}$	$\mathcal{O}(n^3)$	$\mathcal{O}(n^2)$	$\mathcal{O}(n)$

Table 3: Computational requirements for generating reduced-order models

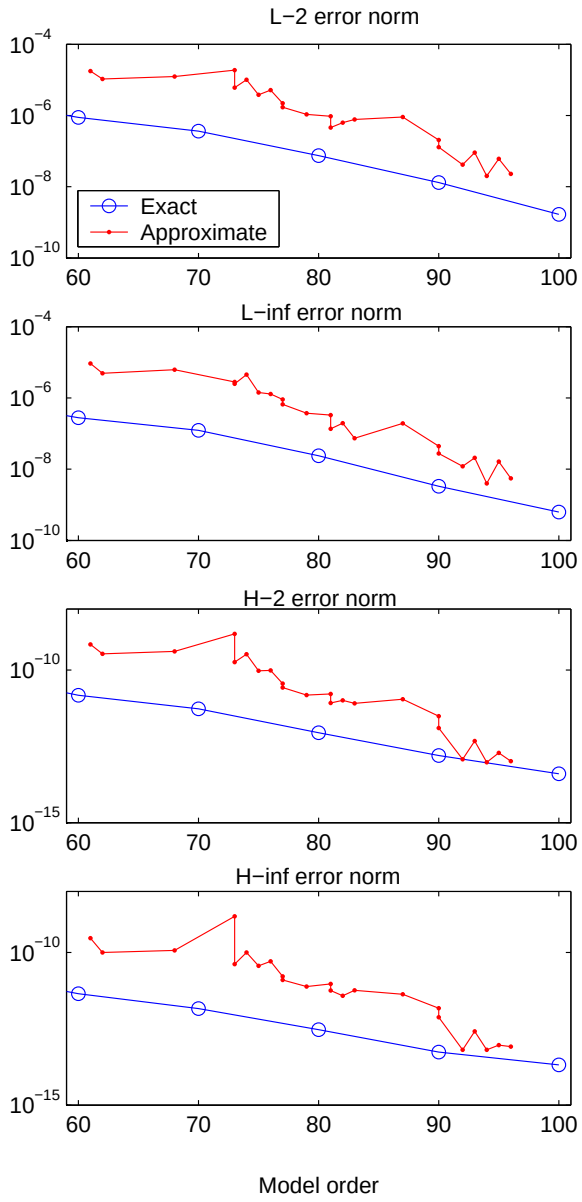

Figure 13: Comparison of approximate balancing to exact balancing

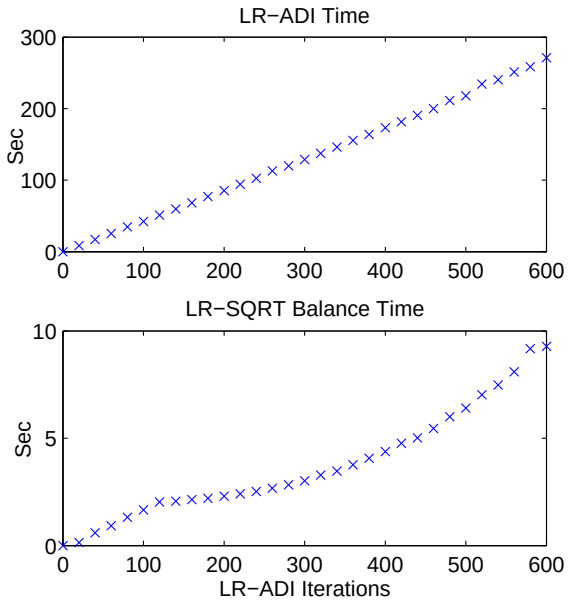

Figure 14: Model reduction times

In general, we observe satisfactory performance of the LR-ADI algorithm with low-rank square root balancing in terms of the modeling error. The LR-ADI algorithm quickly converges to the dominant eigenvectors of the gramians in the numerical example, resulting in reduced order systems whose characteristics closely match those of the original system.

6.1.1. Choice of Shift Parameters

If we know the eigenvalues of the system, we could choose the shift parameters to be the same as the eigenvalues, but the LR-ADI method will take many steps to converge. Since we only want to iterate a small number of steps, the number of shift parameters is limited, and they should approximate the spectrum of the system. We have implemented the shift parameter selection methods by Wachspress [41] and Penzl [11], but the best results were obtained by a heuristic selection procedure that chooses parameters that cover the range of the system eigenvalues, shown in Fig. 15.

If a large number of shift parameters are used, it takes many iterations to cycle through the parameters. If the shift parameters are few and far apart (to cover the spectrum of A), the convergence will also be slow since the spectral radius of A_{p_j} cannot be made small. In general, the number of shift parameters and their locations have to be carefully tuned to obtain the best convergence for a given problem. We have chosen 10 purely real shift parameters, which we found to be a good trade-off between convergence accuracy and convergence rate. We have also noted that the use of complex conjugate parameters (nearer to the system eigenvalues) does not appear to be advantageous. Using 10 complex conjugate pairs with real parts equal to those used in our work results in a slower convergence than the purely real parameters.

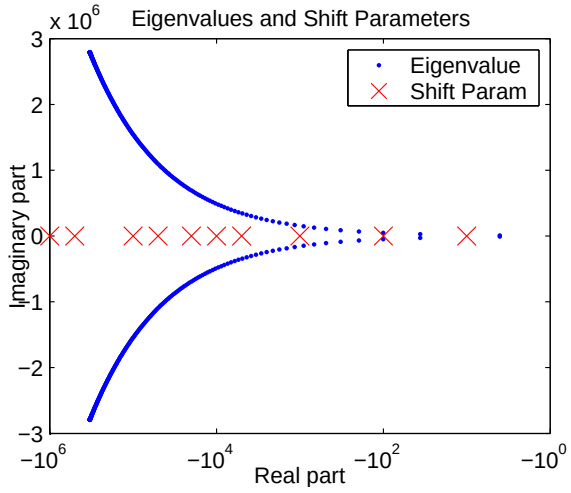

Figure 15: Eigenvalues and shift parameters for beam model

7. CONCLUSION

Large scale dynamical systems arising from the finite element method can often be reduced significantly, since their input-output behavior is dominated by only a small number of internal states. We presented in this paper an overview of the theory and application of model reduction methods based on the input-state and state-output coupling. Among the methods reviewed, balanced truncation is the most attractive as it has a guaranteed H_∞ error bound and produces a reduced-order model that captures well the dominant input-output behavior in both time and frequency domains.

A key step in balanced truncation is the solution of two Lyapunov functions which is computationally intensive and plagued by numerical difficulties for large systems. We presented several iterative methods to generate approximately-balanced reduced order models for large systems. Among them, the

best choice is the low-rank ADI method which balances the computational load and memory storage requirement. However, its effective use requires the selection of a set of shift parameters. The result from low-rank ADI method can be directly used in the low-rank square root method to generate a low order approximate model. A 900-state numerical example is included to show the effectiveness of these methods.

The model reduction algorithms presented here are for LTI systems only, but models of physical systems are invariably nonlinear. However, the gramian concept central to the balanced truncation method may be generalized to nonlinear systems and used for their order reduction [9, 42]. Many physical models also lack damping, such as in molecular dynamics. In this case, finite time model reduction using the same balanced truncation idea could be applied [43].

References

[1] R.J. Guyan. Reduction of mass and stiffness matrices. *American Institute of Aeronautics and Astronautics Journal*, 3(2):380, 1965.

[2] J.C. O'Callahan. A procedure for an improved reduced system IRS model. In *Proceedings of the 6th International Modal Analysis Conference*, pages 17–21, Las Vegas, NV, 1989.

[3] T.I. Zohdi, J.T. Oden, and G.J. Rodin. Hierarchical modeling of heterogeneous bodies. *Comp. Meth. Applied Mech Engineering*, 138:273–298, 1996.

[4] K.J. Bathe, N.S. Lee, and M.L. Bucalem. On the use of hierarchical models in engineering analysis. *Comp. Meth. Appl. Mech. Eng.*, 82:5–26, 1990.

[5] A.C. Cangellaris and L. Zhao. Model order reduction techniques for electromagnetic macromodelling based on finite methods. *Int. Journal of Numerical Model*, 13:181–197, 2000.

[6] A.C. Cangellaris, M. Celik, S. Pasha, and L. Zhao. Electromagnetic model order reduction for system-level modeling. *IEEE Trans. Microwave Theory and Techniques*, 47:840–850, 1999.

[7] J. Rubio, J. Arroyo, and J. Zapata. SFELP: An efficient methodology for microwave circuit analysis. *IEEE Journal on Microwave Theory and Techniques*, 49(3):509–516, March 2001.

[8] J. Tinsley Oden and K. Vemaganti. Estimation of local modeling error and goal-oriented adaptive modeling of heterogeneous materials; part I: Error estimates and adaptive algorithms. *J. Comp. Physics*, 164:22–47, 2000.

[9] A.C. Antoulas, D.C. Sorensen, and S. Gugercin. A survey of model reduction methods for large-scale systems. In *Structured Matrices in Operator Theory, Numerical Analysis, Control, Signal and Image Processing*. AMS, 2001.

[10] P. Benner. Solving large-scale control problems. *IEEE Control Systems Magazine*, 24(1):44–59, February 2004.

[11] T. Penzl, "Algorithms for model reduction of large dynamical systems," T.U. Chemnitz, Germany, Technical Report, 1999.

[12] V. Balakrishnan, Q. Su, and C-K. Koh, "Efficient balance-and-truncate model reduction for large scale systems," *Proceedings of the American Control Conference*, Arlington, VA, 2001.

[13] J-R. Li and J. White, "Efficient Model reduction of interconnects via approximate system gramians," *IEEE/ACM International Conference on Computer Aided Design*, San Jose, CA, 1999.

[14] T. Penzl, "Lyapack - A MATLAB Toolbox for large lyapunov and riccati equations, model reduction problems, and linear-quadratic optimal control problems." Available from http://www.tu-chemnitz.de/sfb393/lyapack/

[15] L. Ljung. *System Identification: Theory for the User*. Prentice-Hall, 1987.

[16] B. De Moor, P. Van Overschee, and W. Favoreel. Numerical algorithms for subspace state space system identification - an overview. In Biswa Datta, editor, *Birkhauser Book Series on Applied an Computational Control, Signals and Circuits*, pages 247–311. Birkhauser, 1999.

[17] T. Stykel, "Model reduction of descriptor systems," Institut für Mathematik, Technische Universität Berlin, Berlin, Germany, Technical Report 720-01, Dec. 2001.

[18] T. Penzl, "Numerical solution of generalized Lyapunov equations," *Advances in Computational Mechanics*, vol. 8, pp. 33-48, 1998.

[19] K. Zhou, J.C. Doyle, and K. Glover. *Robust and Optimal Control*. Prentice-Hall, 1996.

[20] C.A. Desoer and M. Vidyasagar. *Feedback Systems: Input–Output Properties*. Academic Press, New York, 1975.

[21] R. Bartels and G. Stewart, "Algorithm 432, solution of the matrix equation AX + XB =C," *Comm. As. Computer Machinery*, vol. 15, pp. 820-826, 1972.

[22] S. Hammarling, "Numerical solution of the stable, non-negative definite Lyapunov equation,"*IMA Journal of Numerical Analysis*, vol. 2, pp.303-323, 1982.

[23] P. V. Kokotovic, R. E. O'Malley, P. Sannuti, "Singular Perturbations and Order Reduction in Control Theory - an Overview," *Automatica*, vol. 12, pp. 123-132, 1976.

[24] B.C. Moore, "Principal component analysis in linear systems: controllability, observability, and model reduction," *IEEE Transactions on Automatic Control*, vol. AC-26, pp. 17-32, 1981.

[25] M. Tombs and I. Postlethwaite, "Truncated balanced realization of stable, non-minimal state-space systems," *International Journal of Control*, vol. 46, pp. 1319-1330, 1987.

[26] A.C. Antoulas, D.C. Sorensen, and Y. Zhou, "On the decay rate of Hankel singular values and related issues." Rice University, Houston, Texas, Technical Report, 2002.

[27] U.B. Desai and D. Pal. A transformation approach to stochastic model reduction. *IEEE Transaction on Automatic Control*, 29(12):1097–1100, December 1984.

[28] M. Green. A relative-error bound for balanced stochastic truncation. *IEEE Trans. Automat. Control*, 33(10):961–965, 1988.

[29] K. Glover, "All optimal Hankel norm approximations of linear multivariable systems and their L_∞ bound," *Int. Journal of Control,* vol. 39, pp. 1115-1193, 1984.

[30] G. Birkhoff, R. Varga, and D. Young. "Alternating direction implicit methods," in *Advances in Computers*, Vol. 3, New York: Academic Press, pp.189-273, 1962.

[31] E. Wachspress, "Iterative solution of the Lyapunov matrix equation," *Applied Mathematics Letters*, vol. 1, pp.87-90, 1988.

[32] B. Le Bailly and J.P. Thiran, "Optimal rational functions for the generalized zolotarev problem in the complex plane," *SIAM Journal of Numerical Analysis*, vol. 38, no. 5, pp. 1409-1424, 2000.

[33] G. Starke, "Fejer-Walsh points for rational functions and their use in the ADI iterative method," *Journal of Computational and Applied Mathematics*, vol. 46, pp. 129-141, 1993.

[34] G. Starke, "Optimal alternating direction implicit parameters for nonsymmetric systems of linear equations," *SIAM Journal of Numerical Analysis*, vol. 28, no. 5, pp.1431-1445, 1991.

[35] J-R. Li and J. White, "Low rank solutions of Lyapunov equations," *SIAM Journal Matrix Anal. Appl.*, vol. 24, no. 1, pp.260-280, 2002.

[36] J-R Li, "Model reduction of large linear systems via low rank system gramians." Ph.D. dissertation, Massachusetts Institute of Technology, 2000.

[37] W. Gressick. A comparative study of order reduction methods for finite element models. Master's thesis, Rensselaer Polytechnic Institute, Troy, NY., December 2003.

[38] T. Penzl, "A cyclic low-rank Smith method for large sparse Lyapunov equations," *SIAM Journal of scientific computation*, vol. 21, pp.139-144, 2000.

[39] S. Gugercin, D.C. Sorensen, and A.C. Antoulas, "A modified low-rank Smith method for large-scale Lyapunov equations," *Numerical Algorithms*, 32(1), pp.27-55, Jan., 2003.

[40] J. Fish and W. Chen, "Modeling and Simulation of Piezocomposites," *Comp. Meth. Appl. Mech. Engng.*, Vol. 192, pp. 3211-3232, 2003.

[41] E. Wachspress. "The ADI Minimax Problem for Complex Spectra." In *Iterative Methods for Large Linear Systems*, D. Kincaid and L. Hayes, Ed. New York: Academic Press, pp. 251-271, 1990.

[42] S. Lall, J. E. Marsden, and S. Glavaski. Empirical model reduction of controlled nonlinear systems. In *In Proceedings of the IFAC World Congress, Volume F*, pages 473–478, 1999.

[43] M. Barahona, A.C. Doherty, M. Sznaier, H. Mabuchi, and J.C. Doyle. Finite horizon

model reduction and the appearance of dissipation in Hamiltonian systems. In *IEEE Conference on Decision and Control*, pages 4563–4568, December 2002.