Low-rank balanced truncation for linear and bilinear systems via Laguerre polynomials

Zhi-Hua Xiao[a]
joint work with Prof. Yao-Lin Jiang[b] and Dr. Zhen-Zhong Qi[c]

[a]School of Information and Mathematics, Yangtze University, Jingzhou, Hubei
[b]School of Mathematics and Statistics, Xi'an Jiaotong University, Xi'an
[c]School of Mathematics, Northwest University, Xi'an

March 27, 2023

Contents

1. Introduction

- Model order reduction (MOR)

Many engineering problems can be described by input-output systems. A large-scale linear time-invariant (LTI) input-output system can be formulated as

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t), \\ y(t) = Cx(t), \end{cases} \tag{1.1}$$

where $x(t) \in \mathbb{R}^n$ is the state vector, $u(t) \in \mathbb{R}^m$ is a vector of inputs, and $y(t) \in \mathbb{R}^p$ is a vector of outputs. $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, and $C \in \mathbb{R}^{p\times n}$ are constant matrices. The transfer function of (1.1) is defined as

$$H(s) = C(sI - A)^{-1}B.$$

When modeling engineering systems, the resulting mathematical modes (1.1) with high order are sometimes too expensive to simulate. It is essential to reduce the dimension for computational purpose and obtain a reduced-order model of the form

$$\begin{cases} \dot{\tilde{x}}(t) = \tilde{A}\tilde{x}(t) + \tilde{B}u(t), \\ \tilde{y}(t) = \tilde{C}\tilde{x}(t), \end{cases} \tag{1.2}$$

where $\tilde{x}(t) \in \mathbb{R}^r$ is the state vector with a drastically reduced state dimension $r \ll n$. The smaller reduced system (1.2) should approximate the input-output behavior of the original system (1.1) well.

In time domain, it is desired that for all feasible input functions $u(t)$,

$$\tilde{y}(t) \approx y(t), \quad \text{for} \quad t \geq 0.$$

With the help of the Laplace transformation, one can also state the approximation problem in the frequency domain, e.g., via

$$\tilde{H}(i\omega) \approx H(i\omega), \quad \text{for} \quad \omega \in \mathcal{R},$$

where $H(i\omega)$ and $\tilde{H}(s)$ are the transfer functions of (1.1) and (1.2).

- Balanced truncation (BT)

There exist different MOR technologies and here we focus on balanced truncation (BT) MOR. The backbone of BT are the infinite controllability Gramian P and observability Gramian Q of (1.1)

$$P = \int_0^\infty e^{At} B B^{\mathrm{T}} e^{A^{\mathrm{T}} t} dt, \tag{1.3}$$

$$Q = \int_0^\infty e^{A^{\mathrm{T}} t} C^{\mathrm{T}} C e^{At} dt, \tag{1.4}$$

which are the unique solutions to the following two algebraic Lyapunov equations:

$$AP + PA^{\mathrm{T}} + BB^{\mathrm{T}} = 0, \tag{1.5}$$

$$A^{\mathrm{T}} Q + QA + CC^{\mathrm{T}} = 0. \tag{1.6}$$

The Hankel singular values (HSVs) of (1.1) are the eigenvalues of the product PQ, and they are system invariants under state space transformations. The magnitude of the HSVs enables to identify states that are weakly controllable and observable.

In BT this is achieved by first transforming (1.1) into a balanced realization such that $P = Q = \sum = \text{diag}(\sigma_1, \cdots, \sigma_n)$. Then, truncating all states corresponding to small values σ_j gives the reduced order model.

With exact Gramian factors, i.e., $P = FF^{\mathrm{T}}$, $Q = GG^{\mathrm{T}}$, BT is known to always generate a stable ROM for which the error bound

$$\| \tilde{H}(s) - H(s) \|_{\mathcal{H}_\infty} = \sup_{\omega \in \mathcal{R}} (\| \tilde{H}(s) - H(s) \|_2) \leq 2 \sum_{j=r+1}^{n} \sigma_j,$$

holds.

Solving above Lyapunov equations for the required Gramians is the computationally most demanding part of BT. For large-scale systems one therefore uses low-rank approximations of the Gramians instead, e.g.,

$$P \approx FF^{\mathrm{T}}, Q \approx GG^{\mathrm{T}},$$

with low-rank solution factors $F \in \mathcal{R}^{n\times k_p}$, $G \in \mathcal{R}^{n\times k_q}$, $\mathrm{rank}(F) = k_p$, $\mathrm{rank}(G) = k_q$ and $k_p, k_q \ll n$.

This strategy is backed up by the often numerically observed and theoretically explained rapid eigenvalue decay of solutions of Lyapunov equations which causes P, Q to have a small numerical rank. The computation of the low-rank factors F, G can be done efficiently by some numerical algorithms for solving large Lyapunov equations, such as Alternating Direction Iteration method (ADI).

BT using low-rank factors F, G of the Gramians (1.5) and (1.6) is illustrated in Algorithm 1, which called as low-rank square-root method (LRSRM).

Algorithm 1 Square-root balanced truncation with low-rank factors (LRSRM)

Input: A, B, C, tol: tolerance for the approximation error of the ROM;
Output: ROM of order r: $\tilde{A}, \tilde{B}, \tilde{C}$;

1. Compute the low-rank factors F and G in (1.5) and (1.6);
2. Compute the SVD: $G^{\mathrm{T}}F = U\Sigma V^{\mathrm{T}}$, $U_r = U(:, 1:r)$, $V_r = V(:, 1:r)$, and $\Sigma_r = \Sigma(1:r, 1:r)$; r is adaptively chosen by given tolerance: $\delta = 2\sum_{j=r+1}^{r_N} \tilde{\sigma}_j \leq tol$;
3. Construct projection matrices: $T_r = FV_r\Sigma_r^{-\frac{1}{2}}, \quad S_r = \Sigma_r^{-\frac{1}{2}}{U_r}^{\mathrm{T}}G^{\mathrm{T}}$;
4. Construct the ROM: $\tilde{A} = S_rAT_r$, $\tilde{B} = S_rB$, $\tilde{C} = CT_r$.

2. Laguerre approximation of the matrix exponential function

The i-th Laguerre polynomial is defined as follows

$$l_i(t) = \frac{e^t}{i!}\frac{d^i}{dt^i}(e^{-t}t^i), \quad i = 0, 1, \ldots$$

and the scaled Laguerre functions are defined as

$$\phi_i^\alpha(t) = \sqrt{2\alpha}e^{-\alpha t}l_i(2\alpha t).$$

In frequency domain, the sequence of Laplace transforms of the scaled Laguerre functions can be written as

$$\Phi_i^\alpha(s) = \mathcal{L}(\phi_i^\alpha(t)) = \frac{\sqrt{2\alpha}}{s+\alpha}\left(\frac{s-\alpha}{s+\alpha}\right)^i, \quad i = 0, 1, \ldots$$

The Laguerre functions are orthonormal over $[0, \infty)$

$$\int_0^\infty \phi_i^\alpha(t)\phi_j^\alpha dt = \delta_{ij}.$$

For $Re(a) < 0$, we have the Laguerre expansion

$$e^{at} = \sum_{k=0}^{\infty} a_k \phi_k^{\alpha}(t) \quad t \geq 0,$$

with the coefficients $\{a_k\}_{k=0}^{\infty}$ defined by

$$\begin{aligned} a_k &= \int_0^{\infty} e^{at} \phi_k^{\alpha}(t) dt = \frac{\sqrt{2\alpha}}{\alpha - a} \left(\frac{a+\alpha}{a-\alpha} \right)^k \\ &= (-1)^k \sqrt{2\alpha} (a+\alpha)^k (\alpha - a)^{-(k+1)}. \end{aligned}$$

We now extend the above Laguerre expansion from e^{at} to e^{At}. Then, if A is stable, it has

$$e^{At} = \sum_{k=0}^{\infty} A_k \phi_k^{\alpha}(t),$$

with the coefficient matrices $\{A_k\}_{k=0}^{\infty}$ satisfying

$$A_k = (-1)^k \sqrt{2\alpha} (\alpha I + A)^k (\alpha I - A)^{-(k+1)},$$

and e^{At} can be optimally approximated in the L_2 norm sense by truncating above expansion as $e^{At} \approx \sum_{k=0}^{N-1} A_k \phi_k^{\alpha}(t)$.

3. Low-rank BT of linear systems via low-rank Gramian approximation

• Low-rank decomposition of the Gramians P, Q based on Laguerre functions

First, we expand the matrix exponential function $e^{At}B$ by a finite term Laguerre functions as the following approximate form:

$$e^{At}B \approx \sum_{i=0}^{N-1} A_i B \phi_i^{\alpha}(t),$$

where
$A_i = (-1)^i \sqrt{2\alpha}\big(\alpha I + A\big)^i \big(\alpha I - A\big)^{-(i+1)} \in \mathbb{R}^{n\times n}$ $(i = 0, 1, \ldots, N-1)$ are the Laguerre coefficient matrices. Then, it has

$$P = \int_0^{\infty} e^{At}BB^{\mathrm{T}}e^{A^{\mathrm{T}}t}dt = \int_0^{\infty} (e^{At}B)(e^{At}B)^{\mathrm{T}}dt$$
$$\approx \int_0^{\infty} \Big(\sum_{i=0}^{N-1} A_i B \phi_i^{\alpha}(t)\Big)\Big(\sum_{i=0}^{N-1} A_i B \phi_i^{\alpha}(t)\Big)^{\mathrm{T}}dt.$$

According to the orthogonality of the Laguerre functions, we have

$$P \approx FF^{\mathrm{T}}, \tag{3.1}$$

where $F = \begin{bmatrix} A_0B & A_1B & \cdots & A_{N-1}B \end{bmatrix}$.

Similarity to the decomposition of P, the observability Gramian Q has the following low-rank approximation form:

$$Q \approx GG^{\mathrm{T}}, \tag{3.2}$$

where $G = \begin{bmatrix} A_0^{\mathrm{T}}C^{\mathrm{T}} & A_1^{\mathrm{T}}C^{\mathrm{T}} & \cdots & A_{N-1}^{\mathrm{T}}C^{\mathrm{T}} \end{bmatrix}$. The Laguerre coefficient matrices A_i can be calculated by the following recurrence formula:

$$\begin{aligned} A_0 &= \sqrt{2\alpha}(\alpha I - A)^{-1}, \\ A_i &= \left[(A + \alpha I)(A - \alpha I)^{-1}\right] A_{i-1}, \quad i = 1, 2, \ldots, N-1. \end{aligned}$$

- The connection with the ADI iteration

The CF-ADI iteration for computing the low-rank factors $Z_j Z_j^{\mathrm{T}}$ for the solution P of the Lyapunov equation (1.5) is given by

$$z_1 = \sqrt{-2\alpha_1}(A+\alpha_1 I)^{-1}B, \quad Z_1 = \left[\begin{array}{c} z_1 \end{array}\right],$$

$$\beta_i = \left(\frac{\sqrt{-2\alpha_{i+1}}}{\sqrt{-2\alpha_i}}\right)[I-(\alpha_{i+1}+\alpha_i)(A+\alpha_{i+1}I)^{-1}],$$

$$z_j = \beta_{j-1}z_{j-1}, \quad Z_j = \left[\begin{array}{cc} Z_{j-1} & z_j \end{array}\right], \quad \text{for j = 2, 3, . . . , N,}$$

with CF-ADI parameters $\{\alpha_1, \alpha_2, \ldots, \alpha_N\}, Re\{\alpha_j\} < 0$.

Assume that all CF-ADI parameters are the same as $-\alpha$, then we have

$$z_1 = \sqrt{2\alpha}(A-\alpha I)^{-1}B = -A_0 B,$$

$$\beta_i = \left(\frac{\sqrt{2\alpha}}{\sqrt{2\alpha}}\right)[I+2\alpha(A-\alpha I)^{-1}] = (A+\alpha I)(A-\alpha I)^{-1},$$

$$z_j = \beta_{j-1}z_{j-1} = (A+\alpha I)(A-\alpha I)^{-1}z_{j-1} = -A_{j-1}B, \text{for j = 2, 3, . . . , N.}$$

Therefore, except for the difference in the negative sign, the low-rank factor F is the same as the low-rank approximation Z_N of the CF-ADI iteration with all parameters α_j being the same as α.

- Basic algorithm

According to the low-rank factors F and G, we can use the Algorithm 1 to generate the ROM. The reduction procedure can be described by Algorithm 2.

Algorithm 2 Laguerre-Gramian-based LRSRM for linear systems (LG-LRSRM(L))

Input: A, B, C, N, α, tol: tolerance for the approximation error of the ROM;

Output: ROM of order r: $\tilde{A}, \tilde{B}, \tilde{C}$;

1. Compute the low-rank factors F and G from (3.1) and (3.2);
2. Compute the SVD: $G^{\mathrm{T}}F = U\Sigma V^{\mathrm{T}}$, $U_r = U(:, 1:r)$, $V_r = V(:, 1:r)$, and $\Sigma_r = \Sigma(1:r, 1:r)$; r is adaptively chosen by given tolerance: $\delta = 2\sum_{j=r+1}^{r_N} \tilde{\sigma}_j \leq tol$;
3. Compute projection matrices: $T_r = FV_r\Sigma_r^{-\frac{1}{2}}, \quad S_r = \Sigma_r^{-\frac{1}{2}}U_r{}^{\mathrm{T}}G^{\mathrm{T}}$;
4. Construct the ROM: $\tilde{A} = S_rAT_r$, $\tilde{B} = S_rB$, $\tilde{C} = CT_r$.

- The choice of the parameter α

The choice of the Laguerre parameter α directly affects the accuracy of the approximation and stability of the ROM. According to the connection with the CF-ADI, the Laguerre parameter α can be obtained by solving the following minimax optimal problem

$$\min_{\alpha>0} \max_{\lambda\in\sigma(\mathbb{A})} \left|\frac{\lambda-\alpha}{\lambda+\alpha}\right|, \tag{3.3}$$

where $\sigma(A)$ denotes the eigenvalues of A.

- The choice of the parameter α

The choice of the Laguerre parameter α directly affects the accuracy of the approximation and stability of the ROM. According to the connection with the CF-ADI, the Laguerre parameter α can be obtained by solving the following minimax optimal problem

$$\min_{\alpha>0} \max_{\lambda\in\sigma(\mathbb{A})} \left|\frac{\lambda-\alpha}{\lambda+\alpha}\right|, \tag{3.3}$$

where $\sigma(A)$ denotes the eigenvalues of A.

Algorithm 2 is a Petrov-Galerkin projection $(S_r \neq T_r)$. The main disadvantage is that it may lead to numerical errors and instabilities. To alleviate such shortcoming, we use a modification of the dominant subspaces projection model reduction (DSPMR) to modify the above algorithm. The modified algorithm is given by Algorithm 3 as follows.

Algorithm 3 Laguerre-Gramian-based DSPMR for linear systems (LG-DSPMR(L))

Input: A, B, C, N, α, tol;
Output: $\tilde{A}, \tilde{B}, \tilde{C}$;

1. Compute low-rank factors F and G from (3.1) and (3.2);
2. Compute the SVDs: $F = U_F \Sigma_F V_F^{\mathrm{T}}$, $G = U_G \Sigma_G V_G^{\mathrm{T}}$;
3. Choose $\tilde{r}/2 \leq k \leq min\{r_F, r_G\}$, $\tilde{r}$ is adaptively chosen by given tolerance: $\delta = 2\sum_{j=r+1}^{r_N} \tilde{\sigma}_j \leq tol$;
4. Compute the QR decomposition: $\left[U_F(:, 1:k), \quad U_G(:, 1:k)\right] = QR$, $V = Q(:, 1:r)$;
5. Construct the ROM: $\tilde{A} = V^{\mathrm{T}} A V$, $\tilde{B} = V^{\mathrm{T}} B$, $\tilde{C} = CV$.

An extension of the DSPMR method is called refined DSPMR. The eponymous refinement is given by weighting factors $1/\|F\|_F$ and $1/\|G\|_F$ which are selected as the Frobenius norm of the respective low-rank factors for the controllability and observability Gramians. This normalization aims to equilibrate the influence of controllability and observability which may be skewed, i.e., due to different scaling of B and C. The corresponding procedure is given by Algorithm 4 as follow.

Algorithm 4 Laguerre-Gramian-based refined DSPMR for linear systems (LG-RDSPMR(L))

Input: A, B, C, N, α, r;
Output: $\tilde{A}, \tilde{B}, \tilde{C}$;

1. Compute low-rank factors F and G from (3.1) and (3.2);
2. Compute the SVD: $\begin{bmatrix} \frac{1}{\|F\|_F} F & \frac{1}{\|G\|_F} G \end{bmatrix} = U\Sigma V^{\mathrm{T}}$, $S = U(:, 1:r)$;
3. Construct the ROM: $\tilde{A} = S^{\mathrm{T}} A S$, $\tilde{B} = S^{\mathrm{T}} B$, $\tilde{C} = CS$.

---

4. Low-rank BT of bilinear systems via low-rank Gramian approximation

In this section, we extend the above approach from linear systems to the following time invariant multi-input multi-output (MIMO) bilinear system

$$\begin{cases} \dot{x}(t) = Ax(t) + \sum_{i=1}^{m} N_i x(t) u_i(t) + Bu(t), \\ y(t) = Cx(t), \end{cases} \tag{4.1}$$

where t is the time variable, $x(t) \in \mathbb{R}^n$ is the state of the system, n is the dimension of the state space. $u(t) \in \mathbb{R}^m$ and $y(t) \in \mathbb{R}^p$ are the input and output functions, and $u_i(t)$ is the ith component of $u(t)$.
$A \in \mathbb{R}^{n\times n}, B \in \mathbb{R}^{n\times m}, C \in \mathbb{R}^{n\times p}, N_i \in \mathbb{R}^{n\times n}$ for $i = 1, 2, \ldots, m$, are constant matrices.

The controllability ($\mathcal{P}$) and observability ($\mathcal{Q}$) Gramians of bilinear system (4.1) are defined as

$$\mathcal{P} = \sum_{i=1}^{\infty} \int_0^{\infty} \cdots \int_0^{\infty} \bar{P}_i \bar{P}_i^{\mathrm{T}} dt_1 \cdots dt_i, \tag{4.2}$$

$$\mathcal{Q} = \sum_{i=1}^{\infty} \int_0^{\infty} \cdots \int_0^{\infty} \bar{Q}_i \bar{Q}_i^{\mathrm{T}} dt_1 \cdots dt_i, \tag{4.3}$$

where

$$\bar{P}_1 = P_1(t_1) = e^{At_1} B, \quad \bar{Q}_1 = Q_1(t_1) = e^{A^{\mathrm{T}} t_1} C^{\mathrm{T}},$$

$$\bar{P}_i = P_i(t_1, \ldots, t_i) = \left[\begin{array}{cccc} e^{At_i} N_1 \bar{P}_{i-1} & e^{At_i} N_2 \bar{P}_{i-1} & \ldots & e^{At_i} N_m \bar{P}_{i-1} \end{array} \right],$$

$$\bar{Q}_i = Q_i(t_1, \ldots, t_i) = \left[\begin{array}{cccc} e^{A^{\mathrm{T}} t_i} N_1^{\mathrm{T}} \bar{Q}_{i-1} & e^{A^{\mathrm{T}} t_i} N_2^{\mathrm{T}} \bar{Q}_{i-1} & \ldots & e^{A^{\mathrm{T}} t_i} N_m^{\mathrm{T}} \bar{Q}_{i-1} \end{array} \right].$$

They satisfy the following generalized algebraic Lyapunov equations:

$$A\mathcal{P} + \mathcal{P}A^{\mathrm{T}} + \sum_{i=1}^{m} N_i \mathcal{P} N_i^{\mathrm{T}} + BB^{\mathrm{T}} = 0,$$

$$A^{\mathrm{T}}\mathcal{Q} + \mathcal{Q}A + \sum_{i=1}^{m} N_i^{\mathrm{T}} \mathcal{Q} N_i + C^{\mathrm{T}}C = 0.$$

According to (3.1), we have

$$\mathcal{P}_1 = \int_0^\infty \bar{P}_1 \bar{P}_1^{\mathrm{T}} dt_1 = \int_0^\infty e^{At_1} B B^{\mathrm{T}} e^{A^{\mathrm{T}} t_1} dt_1 \approx \mathcal{F}_1 \mathcal{F}_1^{\mathrm{T}},$$

where $\mathcal{F}_1 = \begin{bmatrix} F_{1,0} & F_{1,1} & \dots & F_{1,N-1} \end{bmatrix} \in \mathbb{R}^{n \times Nm}$ and

$$\begin{aligned} F_{1,0} &= \sqrt{2\alpha}(\alpha I - A)^{-1} B, \\ F_{1,j} &= \left[(A + \alpha I)(A - \alpha I)^{-1}\right] F_{1,j-1}, \quad j = 1, 2, \dots, N-1. \end{aligned}$$

Consider the first two terms in the series in (4.2), which are dependent on the first two kernels of the Volterra series of the bilinear system

$$\begin{aligned} \mathcal{P}_2 &= \int_0^\infty \bar{P}_1 \bar{P}_1^{\mathrm{T}} dt_1 + \int_0^\infty \int_0^\infty \bar{P}_2 \bar{P}_2^{\mathrm{T}} dt_1 dt_2 \\ &\approx \mathcal{F}_1 \mathcal{F}_1^{\mathrm{T}} + \mathcal{F}_2 \mathcal{F}_2^{\mathrm{T}}, \end{aligned}$$

where $\mathcal{F}_2 = \begin{bmatrix} F_{2,0} & F_{2,1} & \dots & F_{2,N-1} \end{bmatrix} \in \mathbb{R}^{n \times N^2 m^2}$ and

$$\begin{aligned} F_{2,0} &= \sqrt{2\alpha}(\alpha I - A)^{-1} \begin{bmatrix} N_1 \mathcal{F}_1 & N_2 \mathcal{F}_1 & \dots & N_m \mathcal{F}_1 \end{bmatrix}, \\ F_{2,j} &= \left[(A + \alpha I)(A - \alpha I)^{-1}\right] F_{2,j-1}, \quad j = 1, 2, \dots, N-1. \end{aligned}$$

Then, for the truncated Gramian with the first l terms $\mathcal{P}_l$ of the bilinear system, we have the following low-rank decomposition:

$$\mathcal{P}_l = \sum_{i=1}^{l} \int_0^\infty \cdots \int_0^\infty \bar{P}_i \bar{P}_i^{\mathrm{T}} dt_1 \cdots dt_i \approx \mathcal{F}\mathcal{F}^T, \tag{4.4}$$

where $\mathcal{F} = \begin{bmatrix} \mathcal{F}_1 & \mathcal{F}_2 & \dots & \mathcal{F}_l \end{bmatrix}$, $\mathcal{F}_i = \begin{bmatrix} F_{i,0} & F_{i,1} & \dots & F_{i,N-1} \end{bmatrix}$ and

$$\begin{cases} F_{1,0} = \sqrt{2\alpha}(\alpha I - A)^{-1} B, \\ F_{1,j} = \left[(A + \alpha I)(A - \alpha I)^{-1}\right] F_{1,j-1}, \quad j = 1, 2, \dots, N-1 \\ F_{i,0} = \sqrt{2\alpha}(\alpha I - A)^{-1} \begin{bmatrix} N_1 \mathcal{F}_{i-1} & N_2 \mathcal{F}_{i-1} & \dots & N_m \mathcal{F}_{i-1} \end{bmatrix}, \\ F_{i,j} = \left[(A + \alpha I)(A - \alpha I)^{-1}\right] F_{i,j-1}, \quad i = 2, 3, \dots, l. \end{cases}$$

Similarity to the decomposition of $\mathcal{P}_l$, the truncated observability Gramian with the first l terms $\mathcal{Q}_l$ of the bilinear system has the following low-rank approximation form:

$$\mathcal{Q}_l = \sum_{i=1}^{l} \int_0^\infty \cdots \int_0^\infty \bar{Q}_i \bar{Q}_i^{\mathrm{T}} dt_1 \cdots dt_i \approx \mathcal{G}\mathcal{G}^T, \tag{4.5}$$

where $\mathcal{G} = \begin{bmatrix} \mathcal{G}_1 & \mathcal{G}_2 & \dots & \mathcal{G}_l \end{bmatrix}$, $\mathcal{G}_i = \begin{bmatrix} G_{i,0} & G_{i,1} & \dots & G_{i,N-1} \end{bmatrix}$ and

$$\begin{cases} G_{1,0} = \sqrt{2\alpha}(\alpha I - A^{\mathrm{T}})^{-1} C^{\mathrm{T}}, \\ G_{1,j} = \left[(A^{\mathrm{T}} + \alpha I)(A^{\mathrm{T}} - \alpha I)^{-1}\right] G_{1,j-1}, \quad j = 1, 2, \dots, N-1 \\ G_{i,0} = \sqrt{2\alpha}(\alpha I - A^{\mathrm{T}})^{-1} \begin{bmatrix} N_1^{\mathrm{T}} \mathcal{G}_{i-1} & N_2^{\mathrm{T}} \mathcal{G}_{i-1} & \dots & N_m^{\mathrm{T}} \mathcal{G}_{i-1} \end{bmatrix}, \\ G_{i,j} = \left[(A^{\mathrm{T}} + \alpha I)(A^{\mathrm{T}} - \alpha I)^{-1}\right] G_{i,j-1}, \quad i = 2, 3, \dots, l. \end{cases}$$

Similar to the above LG-LRSRM and LG-DSPMR methods for linear systems, the reduced bilinear model can be generated by the low-rank factors $\mathcal{F}$ and $\mathcal{G}$. The corresponding reduction methods are described by Algorithm 5, Algorithm 6 and Algorithm 7, respectively.

Algorithm 5 Laguerre-Gramian-based LRSRM for bilinear systems (LG-LRSRM(BL))

Input: $A, B, C, N_i, N, l, \alpha$, tol: tolerance for the approximation error of the ROM;

Output: ROM of order r: $\tilde{A}, \tilde{N}_i, \tilde{B}, \tilde{C}$;

1. Compute the low-rank factors $\mathcal{F}$ and $\mathcal{G}$ from (4.4) and (4.5);
2. Compute the SVD: $\mathcal{G}^{\mathrm{T}}\mathcal{F} = \mathcal{U}\Sigma\mathcal{V}^{\mathrm{T}}$, $\mathcal{U}_r = \mathcal{U}(:, 1:r)$, $\mathcal{V}_r = \mathcal{V}(:, 1:r)$, and $\Sigma_r = \Sigma(1:r, 1:r)$; r is adaptively chosen by given tolerance: $\delta = 2\sum_{j=r+1}^{r_N} \tilde{\sigma}_j \leq tol$;
3. Compute projection matrices: $\mathcal{T}_r = \mathcal{F}\mathcal{V}_r{\Sigma_r}^{-\frac{1}{2}}, \quad \mathcal{S}_r = {\Sigma_r}^{-\frac{1}{2}}{\mathcal{U}_r}^{\mathrm{T}}\mathcal{G}^{\mathrm{T}}$;
4. Construct the ROM: $\tilde{A} = \mathcal{S}_r A \mathcal{T}_r$, $\tilde{N}_i = \mathcal{S}_r N_i \mathcal{T}_r$, $\tilde{B} = \mathcal{S}_r B$, $\tilde{C} = C\mathcal{T}_r$.

Algorithm 6 Laguerre-Gramian-based DSPMR for bilinear systems (LG-DSPMR(BL))

Input: $A, B, C, N_i, N, \alpha, r$;
Output: $\tilde{A}, \tilde{B}, \tilde{C}, \tilde{N}_i$;

1. Compute low-rank factors $\mathcal{F}$ and $\mathcal{G}$ from (4.4) and (4.5);
2. Compute the SVDs: $\mathcal{F} = \mathcal{U}_F \Sigma_F \mathcal{V}_F^{\mathrm{T}}$, $\mathcal{G} = \mathcal{U}_G \Sigma_G \mathcal{V}_G^{\mathrm{T}}$;
3. Choose $\tilde{r}/2 \leq k \leq min\{r_{\mathcal{F}}, r_{\mathcal{G}}\}$, $\tilde{r}$ is adaptively chosen by given tolerance: $\delta = 2\sum_{j=r+1}^{r_N} \tilde{\sigma}_j \leq tol$;
4. Compute the QR decomposition: $\left[\mathcal{U}_F(:, 1:k), \quad \mathcal{U}_G(:, 1:k)\right] = QR$, $\mathcal{V} = Q(:, 1:r)$;
5. Construct the ROM: $\tilde{A} = \mathcal{V}^{\mathrm{T}} A \mathcal{V}$, $\tilde{N}_i = \mathcal{V} N_i \mathcal{V}$, $\tilde{B} = \mathcal{V}^{\mathrm{T}} B$, $\tilde{C} = C\mathcal{V}$.

Algorithm 7 Laguerre-Gramian-based refined DSPMR for bilinear systems (LG-RDSPMR(BL))

Input: $A, B, C, N_i, N, \alpha, r$;

Output: $\tilde{A}, \tilde{B}, \tilde{C}, \tilde{N}_i$;

1. Compute low-rank factors $\mathcal{F}$ and $\mathcal{G}$ from (4.4) and (4.5);
2. Compute the SVD: $\begin{bmatrix} \frac{1}{\|\mathcal{F}\|_F}\mathcal{F} & \frac{1}{\|\mathcal{G}\|_F}\mathcal{G} \end{bmatrix} = \mathcal{U}\Sigma\mathcal{V}^{\mathrm{T}}$, $\mathcal{S} = \mathcal{U}(:, 1:r)$;
3. Construct the ROM: $\tilde{A} = \mathcal{S}^{\mathrm{T}} A \mathcal{S}$, $\tilde{N}_i = \mathcal{S}^{\mathrm{T}} N_i \mathcal{S}$, $\tilde{B} = \mathcal{S}^{\mathrm{T}} B$, $\tilde{C} = C\mathcal{S}$.

5. Numerical experiments

Example: Nonlinear Heat Transfer Model. In this example, a bilinear model for a nonlinear heat transfer problem is constructed. The physical system to be modeled in Figure 5.1 is the heat transfer along a 1D beam with length L, cross sectional area S, and nonlinear heat conductivity represented by a polynomial in temperature $T(x,t)$ of arbitrary degree N

$$\kappa(T) = a_0 + a_1 T + \cdots + a_N T^N. \tag{5.1}$$

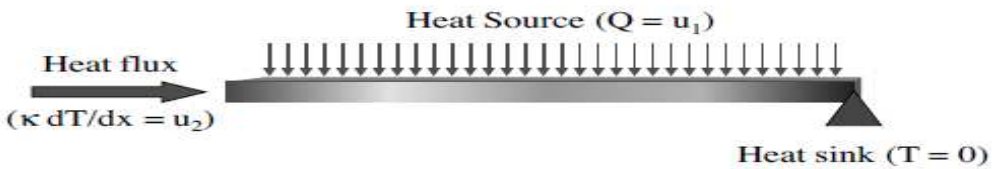

Figure 5.1: The modeled beam with heat flux inputs and heat sink.

The right end of the beam (at $x = L$) is fixed at ambient temperature. The model has two inputs: a time-dependent uniform heat flux $u_1(t)$ at the left end (at $x = 0$) and a time-dependent heat source $u_2(t)$ distributed along the beam.

In our numerical simulation, the number of nodes of the original nonlinear system is 100, yielding a bilinear system of dimension 10100 with 2 inputs and 1 output, taken as the first or the middle node in the discretization. The computational times for constructing and simulating ROMs by different methods and maximum relative errors are reported in Table 5.1.

Table 5.1: Computational times and maximum relative errors of ROMs.

Method	Order	Time	Relative error	Speed up
Bilinear approximation system	10100	445.26	$-$	$-$
LG-LRSRM(BL)	4	32.55	6.44×10^{-2}	13
LG-DSPMR(BL)	4	31.72	1.92×10^{-1}	14
LG-RDSPMR(BL)	4	33.44	1.88×10^{-1}	13
LRBT-TGrams	18	20.37	6.39×10^{-3}	21

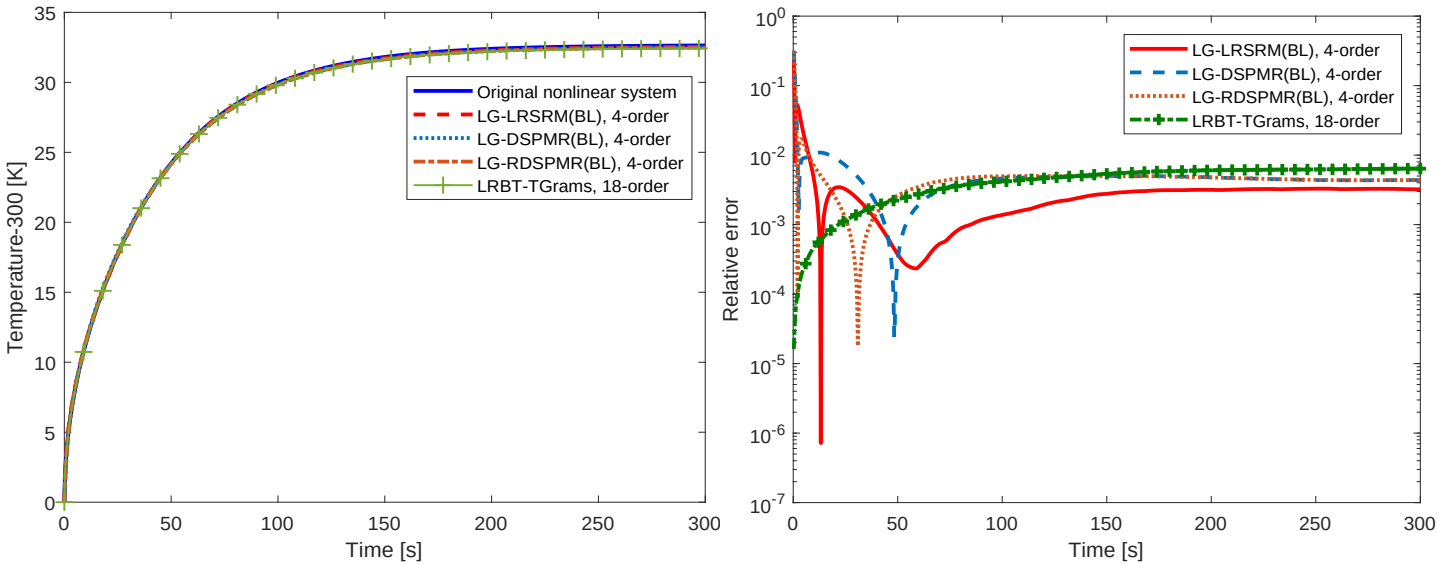

Figure 5.2: Transient responses of the nonlinear heat transfer model for the temperature at the leftmost end for an input of $5 \cdot 10^4 W/m^2$, and the relative errors ε of the ROMs with respect to the original system.

Figure 5.2 shows the steady state temperatures of the original nonlinear heat transfer model at the leftmost of the beam for an input of $5 \cdot 10^4 W/m^2$, and the corresponding relative errors of these ROMs.

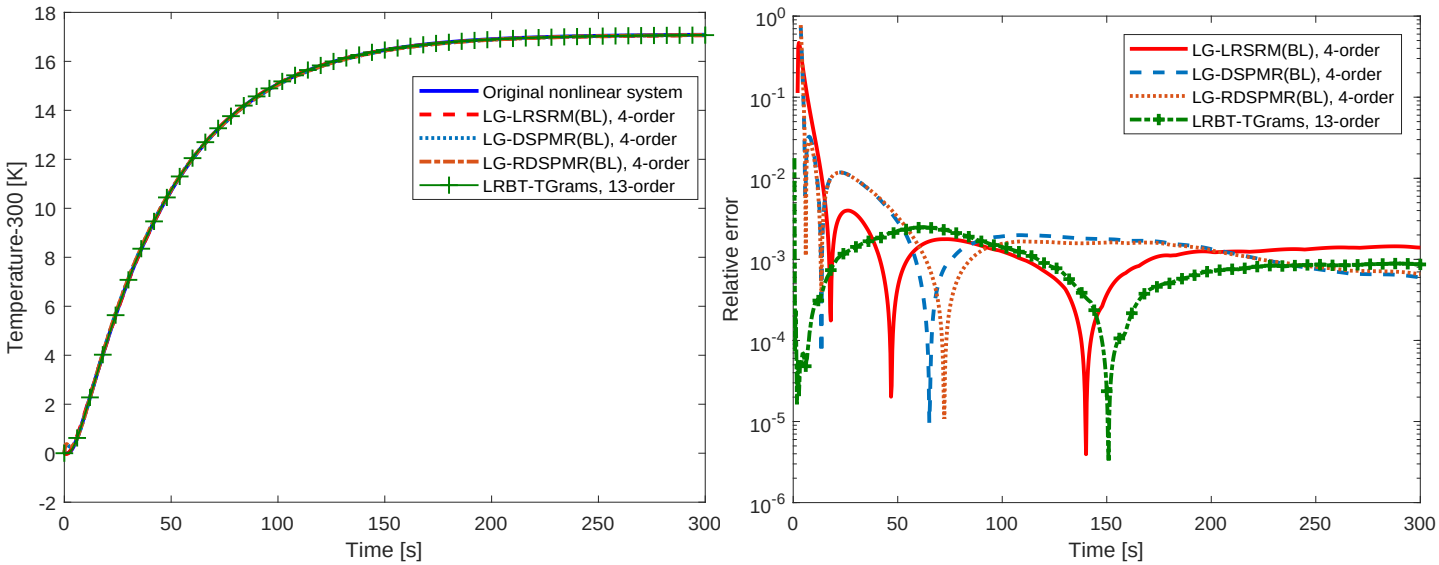

Figure 5.3: Transient responses of the nonlinear heat transfer model for the temperature at the middle of the beam for an input of $5 \cdot 10^4 W/m^2$, and the relative errors ε of the ROMs with respect to the original system.

Figure 5.3 shows the steady state temperatures of the original nonlinear heat transfer model at the middle of the beam for an input of $5 \cdot 10^4 W/m^2$, and the corresponding relative errors of these ROMs.

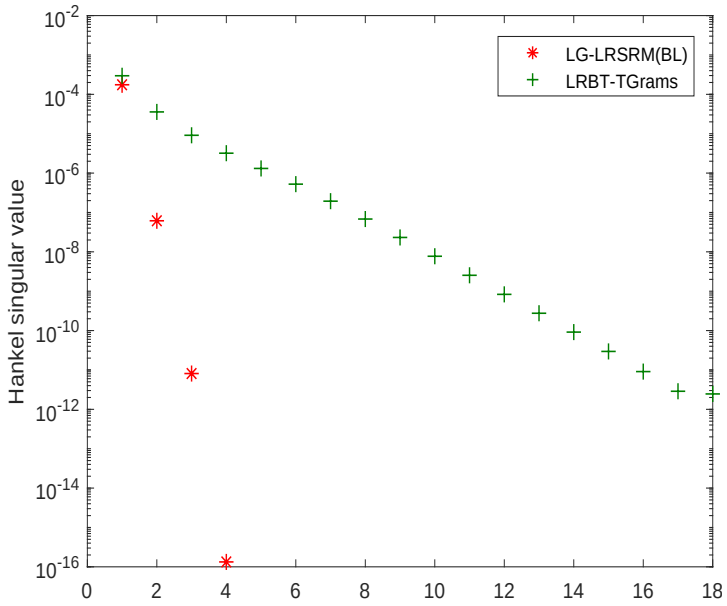

Figure 5.4: Generalized Hankel singular values of the ROMs.

Furthermore, the approximate generalized HSVs of Algorithm 5 and LRBT-TGrams are plotted in Figure 5.4.

Related publications

[1] Z.H. Xiao, Y.X. Fang, Y.L. Jiang, Laguerre-based low-rank balanced truncation of discrete-time systems, *IEEE Transactions on Circuits and Systems II: Express Briefs*, doi: 10.1109/TCSII.2023.3253159, 2023.

[2] Z.H. Xiao, Q.Y. Song, Y.L. Jiang, Z.Z. Qi, Model order reduction of linear and bilinear systems via low-rank Gramian approximation, *Applied Mathematical Modelling* 106 (2022) 100-113.

[3] L. Dai, Z.H. Xiao, R.Z. Zhang, Y.L. Jiang, Laguerre-Gramian-based structure-preserving model order reduction for second-order form systems, *Transactions of the Institute of Measurement and Control* 2022.

[4] L. Dai, Z.H. Xiao, R.Z. Zhang, Laguerre-Gramian-based model order reduction of bilinear systems, *Chinese Control Conference, CCC,* 2021, 2021-July, pp. 1195-1200.

[5] Z.H. Xiao, Y.L. Jiang, Z.Z. Qi, Finite-time balanced truncation for linear systems via shifted Legendre polynomials, *Systems & Control Letters* 126 (2019) 48-57.

[6] Z.H. Xiao, Y.L. Jiang, Z.Z. Qi, Structure preserving balanced proper orthogonal decomposition for second-order form systems via shifted Legendre polynomials, *IET Control Theory & Applications* 13 (8) (2019) 1155-1165.

6. Conclusions

- A series of low-rank balanced truncation MOR algorithms for linear and bilinear systems via Laguerre functions are presented.
- The proposed methods are much efficient due to that the low-rank factors are constructed from a recurrence formula which is equivalent to the CF-ADI iteration method.
- At the same time, our modified algorithms can produce stable ROMs under certain conditions.

6. Conclusions

- A series of low-rank balanced truncation MOR algorithms for linear and bilinear systems via Laguerre functions are presented.
- The proposed methods are much efficient due to that the low-rank factors are constructed from a recurrence formula which is equivalent to the CF-ADI iteration method.
- At the same time, our modified algorithms can produce stable ROMs under certain conditions.
- Low-rank balanced truncation for time varying and nonlinear systems;
- Data-driven modeling and model order reduction of complex dynamical systems;
- Model order reduction of structured network systems.

THANK YOU
FOR YOUR ATTENTION !