Model order reduction for linear systems via low-rank cross Gramians

Qiu-Yan Song
joint work with Prof. Xin Du and Dr. Umair Zulfiqar

School of Mechatronic Engineering and Automation, Shanghai University

March 27, 2023

上海大学
Shanghai University

Contents

1. The cross Gramian of linear systems

Consider a stable linear time invariant (LTI) input-output system

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

where $x(t) \in \mathbb{R}^n$ is the state, $u(t) \in \mathbb{R}^m$ is the input, and $y(t) \in \mathbb{R}^p$ is the output. $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, and $C \in \mathbb{R}^{p\times n}$ are constant matrices. In general, such system is denoted by $\{A, B, C\}$.

The transfer function of system (1.1) is

$$H(s) = C(sI - A)^{-1}B \in \mathbb{R}^{p\times m},$$

which defines the relation between the output and input in frequency domain.

$\{A, B, C\}$ is square if $m = p$, and is called symmetric if its transfer function is symmetric $H(s) = H^{\mathrm{T}}(s)$.

- Cross Gramian of the square system

For the system $\{A, B, C\}$, its controllability Gramian $W_C(T)$ and observability Gramian $W_O(T)$ are defined as

$$W_C(T) = \int_0^T e^{At} B B^{\mathrm{T}} e^{A^{\mathrm{T}} t} dt, \quad W_O(T) = \int_0^T e^{A^{\mathrm{T}} t} C^{\mathrm{T}} C e^{At} dt.$$

For the square system $\{A, B, C\}$, the cross Gramian $W_{CO}(T)$ is defined as

$$W_{CO}(T) = \int_0^T e^{At} B C e^{At} dt, \tag{1.2}$$

for $0 < T \leq \infty$, which combines controllability and observability information.

- Cross Gramian of the non-square system

Because the above definition (1.2) of cross Gramian is not suitable for non-square systems where $m \neq p$, recently, a strategy was developed to calculate the cross Gramian for non-square systems, which converted the topic to the cross Gramian of the so-called "average" SISO system. At first, we partition B and C as

$$B = \begin{bmatrix} b_1 & b_2 & \cdots & b_m \end{bmatrix}, \quad b_i \in \mathbb{R}^{n\times 1},$$

$$C^{\mathrm{T}} = \begin{bmatrix} c_1^{\mathrm{T}} & c_2^{\mathrm{T}} & \cdots & c_p^{\mathrm{T}} \end{bmatrix}, \quad c_j^{\mathrm{T}} \in \mathbb{R}^{n\times 1},$$

where b_i and c_j $(i = 1, 2, \ldots, m; j = 1, 2, \ldots, p)$ individually represent the i-th column of B and j-th row of C. Each combination of b_i and c_j induces a SISO system $\{A, b_i, c_j\}$.

Let

$$W_C^i(T) = \int_0^T e^{At} b_i b_i^{\mathrm{T}} e^{A^{\mathrm{T}} t} dt, \quad W_O^j(T) = \int_0^T e^{A^{\mathrm{T}} t} c_j^{\mathrm{T}} c_j e^{At} dt,$$

$$W_{CO}^{ij}(T) = \int_0^T e^{At} b_i c_j e^{At} dt$$

be the controllability Gramian, observability Gramian and cross Gramian of the (i, j)-th subsystem $\{A, b_i, c_j\}$, respectively. According to the partition, these Gramians of the mp SISO subsystems relate to the full MIMO Gramians as

$$W_C(T) W_O(T) = \sum_{i=1}^{m} \sum_{j=1}^{p} W_C^i(T) W_O^j(T) = \sum_{i=1}^{m} \sum_{j=1}^{p} W_{CO}^{ij}(T) W_{CO}^{ij}(T).$$

Then, the cross Gramian $W_{CO}(T)$ of the square system $(m = p)$ satisfies

$$W_{CO}(T) = \sum_{i=1}^{m} W_{CO}^{ii}(T).$$

The cross Gramian $W_X(T)$ of the non-square system $m \neq p$ is defined as the sum of the cross Gramians of all mp SISO subsystems

$$W_X(T) = \sum_{i=1}^{m} \sum_{j=1}^{p} W_{CO}^{ij}(T),$$

which yields the following representation:

$$\begin{aligned} W_X(T) &= \sum_{i=1}^{m} \sum_{j=1}^{p} \int_0^T e^{At} b_i c_j e^{At} dt \\ &= \int_0^T e^{At} \sum_{i=1}^{m} \sum_{j=1}^{p} b_i c_j e^{At} dt \\ &= \int_0^T e^{At} \Big(\sum_{i=1}^{m} b_i \Big) \Big(\sum_{j=1}^{p} c_j \Big) e^{At} dt. \end{aligned}$$

Hence, this approximate cross Gramian $W_X(T)$ is equal to the cross Gramian of the SISO system $\{A, \sum_{i=1}^{m} b_i, \sum_{j=1}^{p} c_j\}$.

2. Low-rank decomposition of the cross Gramian via Legendre polynomials

The cross Gramian $W_{CO}^{ij}(T) = \int_0^T e^{At} b_i c_j e^{At} dt$ of the (i,j)-th subsystem $\{A, b_i, c_j\}$ can be interpreted as the integral of the product of the system's impulse response and its dual system's impulse response. These impulse responses are trajectories

$$\dot{x}_i(t) = Ax_i(t) + b_i\delta(t) \Rightarrow x_\delta^i(t) = e^{At}b_i,$$

$$\dot{z}_j(t) = A^{\mathrm{T}}z_j(t) + c_j^{\mathrm{T}}\delta(t) \Rightarrow z_\delta^j(t) = e^{A^{\mathrm{T}}t}c_j^{\mathrm{T}},$$

$$\Rightarrow W_{CO}^{ij}(T) = \int_0^T e^{At}b_ic_je^{At}dt = \int_0^T e^{At}b_i(e^{A^{\mathrm{T}}t}c_j^{\mathrm{T}})^{\mathrm{T}}dt = \int_0^T x_\delta^i(t)(z_\delta^j(t))^{\mathrm{T}}$$

The state impulse responses can be computed approximately by the Legendre polynomials.

As a first step towards the approximation of the impulse response $e^{At}b_i$, we expand the state variable $x_\delta^i(t)$ in the subsystem $\{A, b_i, c_j\}$ as the following approximate form

$$x_\delta^i(t) \approx \sum_{k=0}^{N-1} f_{i,k}\bar{P}_k(t), \tag{2.1}$$

where $f_{i,k} \in \mathbb{R}^n$ $(k = 0, 1, \ldots, N-1)$ are the Legendre coefficient vectors. N is the desired approximation terms of the Legendre series. Let $u_i(t)$ in $\{A, b_i, c_j\}$ be the unit impulse function $\delta(t)$. After integration and assuming zero initial condition, the state equation becomes

$$x_\delta^i(t) = A\int_0^t x_\delta^i(\tau)d\tau + b_i. \tag{2.2}$$

Substitute (2.1) into (2.2) and according to the properties of $\bar{P}_i(t)$, we have

$$\begin{aligned}\sum_{k=0}^{N-1} f_{i,k}\bar{P}_k(t) =& \frac{T}{2}Af_{i,0}\bar{P}_0(t) + \sum_{k=1}^{N}\frac{T}{2(2k-1)}Af_{i,k-1}\bar{P}_k(t)\\ &- \sum^{N-2}\frac{T}{2(2k+3)}Af_{i,k+1}\bar{P}_k(t) + b_i.\end{aligned}$$

Equating the coefficients of $\bar{P}_k(t)$ for $k = 0, 1, \ldots, N-1$, and ignoring the term $\bar{P}_N(t)$, we finally have

$$\begin{cases} (I - \frac{T}{2}A)f_{i,0} + \frac{T}{6}Af_{i,1} = b_i, \\ -\frac{T}{2(2k-1)}Af_{i,k-1} + f_{i,k} + \frac{T}{2(2k+3)}Af_{i,k+1} = 0, \qquad k = 1, 2, \ldots, N \\ -\frac{T}{2(2N-3)}Af_{i,N-2} + f_{i,N-1} = 0, \end{cases} \tag{2.3}$$

where I is an identity matrix. Thus, the coefficient vectors $f_{i,k}$ $(k = 0, 1, \ldots, N-1)$ satisfy the block tridiagonal linear system

$$\begin{bmatrix} I - \frac{T}{2}A & \frac{T}{6}A & & & \\ -\frac{T}{2}A & I & \frac{T}{10}A & & \\ & -\frac{T}{6}A & I & \ddots & \\ & & \ddots & \ddots & \frac{T}{2(2N-1)}A \\ & & & -\frac{T}{2(2N-3)}A & I \end{bmatrix} \begin{bmatrix} f_{i,0} \\ f_{i,1} \\ \vdots \\ f_{i,N-2} \\ f_{i,N-1} \end{bmatrix} = \begin{bmatrix} b_i \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix}. \tag{2.4}$$

Similarly, the coefficient vectors $g_{j,k} \in \mathbb{R}^n$ $(k = 1, 2, \ldots, N-1)$ for the state impulse response of the dual SISO system $\{A^{\mathrm{T}}, c_j^{\mathrm{T}}, b_i^{\mathrm{T}}\}$

$$z_\delta^j(t) = e^{A^{\mathrm{T}} t} c_j \approx \sum_{k=0}^{N-1} g_{j,k} \bar{P}_k(t),$$

satisfy the following block tridiagonal linear system

$$\begin{bmatrix} I - \frac{T}{2}A^{\mathrm{T}} & \frac{T}{6}A^{\mathrm{T}} & & & \\ -\frac{T}{2}A^{\mathrm{T}} & I & \frac{T}{10}A^{\mathrm{T}} & & \\ & -\frac{T}{6}A^{\mathrm{T}} & I & \ddots & \\ & & \ddots & \ddots & \frac{TA^{\mathrm{T}}}{2(2N-1)} \\ & & & -\frac{TA^{\mathrm{T}}}{2(2N-3)} & I \end{bmatrix} \begin{bmatrix} g_{j,0} \\ g_{j,1} \\ \vdots \\ g_{j,N-2} \\ g_{j,N-1} \end{bmatrix} = \begin{bmatrix} c_j^{\mathrm{T}} \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix} \tag{2.5}$$

Suppose that the coefficient vectors $f_{i,k}, g_{j,k}$ $(k = 0, 1, \ldots, N-1)$ have been obtained by solving the block tridiagonal linear equations (2.4) and (2.5). Then, it has

$$\int_0^T x_\delta^i(t)(z_\delta^j)^{\mathrm{T}}(t)dt \approx \int_0^T \sum_{k=0}^{N-1} f_{i,k}\bar{P}_k(t) \sum_{k=0}^{N-1} g_{j,k}^{\mathrm{T}}\bar{P}_k(t)dt.$$

According to the orthogonality of the shifted Legendre polynomials, it leads to

$$W_{CO}^{ij}(T) = \int_0^T x_\delta^i(t)(z_\delta^j)^{\mathrm{T}}(t)dt \approx \sum_{k=0}^{N-1} \frac{T}{2k+1} f_{i,k} g_{j,k}^{\mathrm{T}} = F_i G_j^{\mathrm{T}},$$

where $F_i = [\sqrt{T} f_{i,0}, \sqrt{T/3} f_{i,1}, \cdots, \sqrt{T/(2N-1)} f_{i,N-1}] \in \mathbb{R}^{n \times N}$, and $G_j = [\sqrt{T} g_{j,0}, \sqrt{T/3} g_{j,1}, \cdots, \sqrt{T/(2N-1)} g_{j,N-1}] \in \mathbb{R}^{n \times N}$.

As a result, the approximation to the cross Gramian $W_{CO}(T)$ of the square system $(m = p)$ is

$$W_{CO}(T) = \sum_{i=1}^{m} W_{CO}^{ii}(T) \approx \sum_{i=1}^{m} F_i G_i^{\mathrm{T}} = FG^{\mathrm{T}}, \tag{2.6}$$

where $F = \left[\begin{array}{cccc} F_1 & F_2 & \ldots & F_m \end{array} \right] \in \mathbb{R}^{n \times mN}$,
$G = \left[\begin{array}{cccc} G_1 & G_2 & \ldots & G_m \end{array} \right] \in \mathbb{R}^{n \times mN}$.

Similarly, the cross Gramian $W_X(T)$ of the non-square system has the following low-rank decomposition

$$W_X(T) \approx \tilde{F}\tilde{G}^{\mathrm{T}}, \tag{2.7}$$

where

$$\tilde{F} = [\sqrt{T}\tilde{f}_0, \sqrt{T/3}\tilde{f}_1, \cdots, \sqrt{T/(2N-1)}\tilde{f}_{N-1}] \in \mathbb{R}^{n \times N},$$

$$\tilde{G} = [\sqrt{T}\tilde{g}_0, \sqrt{T/3}\tilde{g}_1, \cdots, \sqrt{T/(2N-1)}\tilde{g}_{N-1}] \in \mathbb{R}^{n \times N},$$

and $\tilde{f}_i, \tilde{g}_i \in \mathbb{R}^n$ $(i = 0, 1, \ldots, N-1)$ are obtained by replacing the right terms b_i and c_j in (2.4) and (2.5) with $\sum_{i=1}^{m} b_i$ and $\sum_{j=1}^{p} c_j$, respectively.

(2.6) is the low-rank decomposition of the cross Gramian $W_{CO}(T)$ for the square system (1.1). Then, we can use the low-rank square-root method (LRSRM) to generate the ROM. Applying the SVD technique to $G^{\mathrm{T}}F$, we obtain

$$G^{\mathrm{T}}F = \begin{bmatrix} U_1 & U_2 \end{bmatrix} \begin{bmatrix} \Sigma_1 & 0 \\ 0 & \Sigma_2 \end{bmatrix} \begin{bmatrix} {V_1}^{\mathrm{T}} \\ {V_2}^{\mathrm{T}} \end{bmatrix},$$

where $\Sigma_1 = \text{diag}\{\tilde{\sigma}_1, \tilde{\sigma}_2, \cdots, \tilde{\sigma}_r\} \in \mathbb{R}^{r\times r}$ is invertible, $\Sigma_2 = \text{diag}\{\tilde{\sigma}_{r+1}, \tilde{\sigma}_{r+2}, \cdots, \tilde{\sigma}_{rN}, 0, \cdots, 0\}$ with $r_N = \text{rank}(G^{\mathrm{T}}F)$. Construct the projection matrices $T_r \in \mathbb{R}^{n\times r}$ and $S_r \in \mathbb{R}^{r\times n}$ by

$$T_r = FV_1\Sigma_1^{-\frac{1}{2}}, \quad S_r = \Sigma_1^{-\frac{1}{2}}{U_1}^{\mathrm{T}}G^{\mathrm{T}}.$$

Obviously, it has $S_rT_r = I_r$.

Then, we can get the resulted ROM of the square system $\{A, B, C\}$ by

$$\begin{cases} \dot{x}_r(t) = A_r x_r(t) + B_r u(t), \\ y_r(t) = C_r x_r(t), \end{cases} \tag{3.1}$$

where $A_r = S_r A T_r$, $B_r = S_r B$ and $C_r = C T_r$. According to the square-root method, the smaller singular values are truncated. As a result, for a given tolerance tol, the order r of the ROM is adaptively chosen by the following indicator

$$\tilde{\delta} = 2 \sum_{j=r+1}^{r_N} \tilde{\sigma}_j \leq tol,$$

where $\{\tilde{\sigma}_1, \tilde{\sigma}_2, \cdots, \tilde{\sigma}_{r_N}\}$ is in decreasing order.

Algorithm 1 Cross Gramian-based low-rank square root method for square systems (CG-LRSRM(W_{CO}))

Input: $A, B, C, T, N,$ *tol*;

Output: ROM of order r: A_r, B_r, C_r;

1. Compute the low-rank factors F and G from (2.4) and (2.5);
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: $\tilde{\delta} = 2\sum_{j=r+1}^{r_N} \tilde{\sigma}_j \leq tol$;
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: $A_r = S_rAT_r$, $B_r = S_rB$, $C_r = CT_r$.

The proposed reduction procedure can be described by Algorithm 1, whose projection is a Petrov-Galerkin projection, where S_r and T_r are different.

The above reduction procedure may lead to numerical errors and instabilities. To alleviate these shortcomings, a modified method is presented, which was using a modification of the dominant subspaces projection model reduction (DSPMR). The modified algorithm can be described by Algorithm 2.

Algorithm 2 Cross Gramian-based DSPMR for square systems (CG-DSPMR(W_{CO}))

Input: A, B, C, T, N, tol;
Output: A_r, B_r, C_r;
1. Compute low-rank factors F and G from (2.4) and (2.5);
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: $\tilde{\delta} = 2\sum_{j=r+1}^{r_N} \tilde{\sigma}_j \leq tol$;
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: $A_r = V^{\mathrm{T}} A V$, $B_r = V^{\mathrm{T}} B$, $C_r = CV$.

4. Cross Gramian-based MOR for non-square systems

For the non-square systems where $m \neq p$, (2.7) is the low-rank decomposition of the cross Gramian $W_X(T)$. Then, we have the following cross Gramian-based low-rank square-root method (LRSRM) (Algorithm 3) for non-square systems.

Algorithm 3 Cross Gramian-based LRSRM for non-square systems (CG-LRSRM(W_X))

Input: A, B, C, T, N, tol;

Output: ROM of order r: A_r, B_r, C_r;

1. Compute low-rank factors F and G of the SISO system $\{A$, $\sum_{i=1}^{m} b_i$, $\sum_{j=1}^{p} c_j\}$ from (2.4) and (2.5);
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: $\tilde{\delta} = 2\sum_{j=r+1}^{r_N} \tilde{\sigma}_j \leq tol$;
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 reduced model: $A_r = S_rAT_r$, $B_r = S_rB$, $C_r = CT_r$.

Meanwhile, analogously to Algorithm 2, we also have Algorithm 4 based on DSPMR for non-square systems.

Algorithm 4 Cross Gramian-based DSPMR for non-square systems (CG-DSPMR(W_X))

Input: A, B, C, T, N, tol;

Output: A_r, B_r, C_r;

1. Compute low-rank factors F and G of the SISO system $\{A$, $\sum_{i=1}^{m} b_i$, $\sum_{j=1}^{p} c_j\}$ from (2.4) and (2.5);
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: $\tilde{\delta} = 2\sum_{j=\tilde{r}+1}^{r_N} \tilde{\sigma}_j \leq tol$;
4. Compute the QR decomposition: $\begin{bmatrix} U_F(:,1:k), & U_G(:,1:k) \end{bmatrix} = QR$, $V = Q(:,1:r)$;
5. Construct the reduced model: $A_r = V^{\mathrm{T}} A V$, $B_r = V^{\mathrm{T}} B$, $C_r = CV$.

Example 1 (CD player): This example is a model of compact disc (CD) player. The model describes the dynamics between the lens actuator and the radial arm position of a portable CD player. This square system has 120 states with two inputs and two outputs

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

Numerical results show that A is stable and $A + A^{\mathrm{T}} < 0$.

For this square system, two parameters in our proposed algorithms are taken as $N = 20$ and $T = 10$. With $tol = 10^{-10}$, the reduced order of Algorithm 1 and 2 is adaptively determined as 34.

The computational times to construct each ROMs and the stability are listed in Table 5.1.

Table 5.1: Computational cost and stability of ROMs for Example 1

Method	ROM size r	Time (second)	A_r	$A_r + A_r^{\mathrm{T}}$
CG-LRSRM(W_{CO})	34	0.012	**u**	**N**
CG-DSPMR(W_{CO})	34	0.013	s	Y
CG-DS	34	0.022	s	Y
BT	34	0.049	s	**N**

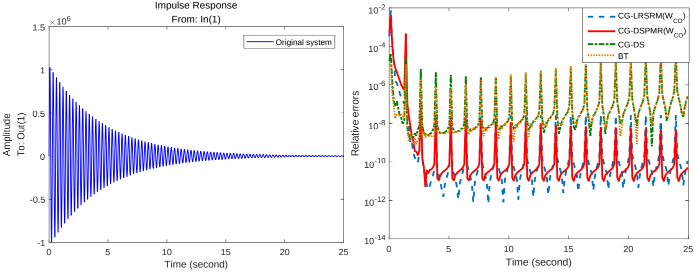

Figure 5.1: Impulse response of the first output for the first input of the original system, and the relative errors ε of the ROMs for Example 1.

The impulse response of the first output for the first input of the original system and the corresponding relative errors are shown in Figure 5.1.

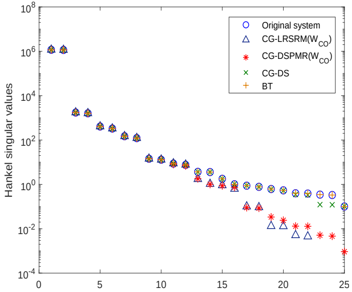

Figure 5.2: Hankel singular values of the ROMs in Example 1.

Figure 5.2 shows the first 25 HSVs of each ROMs.

Example 2 (linear SI5O system): This example is a benchmark problem coming from a discretization of a convective thermal flow problem. The associated linear time-invariant system is given by

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

Some relevant quantities for the model are listed in Table 5.2.

Table 5.2: Some relevant quantities for Example 2.

Example No.	Matrix dimension	Non-zeros in A	Non-zeros in E
1	9669	67391	9669

Note that A is negative definite while E is positive definite, so that the resulting linear time-invariant system is stable.

The computational times for constructing the ROMs by different methods and the stability are reported in Table 5.3.

Table 5.3: Computational cost and stability of ROMs for Example 2

Method	ROM size r	Time (second)	A_r	$A_r + A_r^{\mathrm{T}}$
CG-LRSRM(W_X)	6	35.74	s	N
CG-DSPMR(W_X)	6	35.75	s	Y
CG-DS	17	25.65	s	Y
BT	6	4094.18	s	N

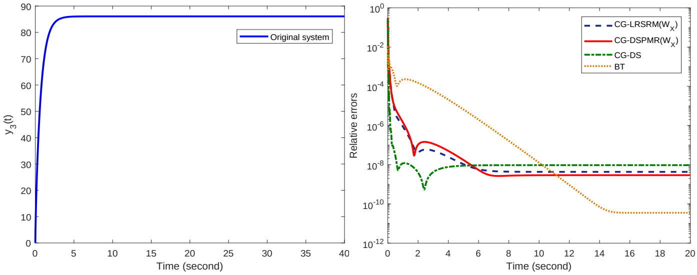

Figure 5.3: The transient response of the third output $y_3(t)$ of the original system, and the relative errors ε of the ROMs for Example 2.

Meanwhile, in Figure 5.3, the relative errors ϵ of the transient response for each constructed ROMs with unit step function $u(t) = \begin{cases} 1, & t \geq 0, \\ 0, & t < 0, \end{cases}$ are plotted.

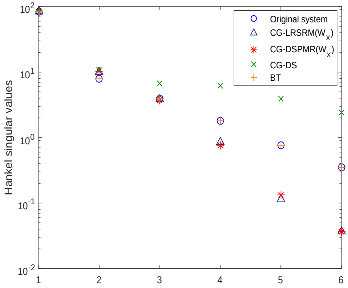

Figure 5.4: Hankel singular values of the ROMs in Example 2.

Figure 5.4 shows the first 6 HSVs of each ROMs.

THANK YOU
FOR YOUR ATTENTION !