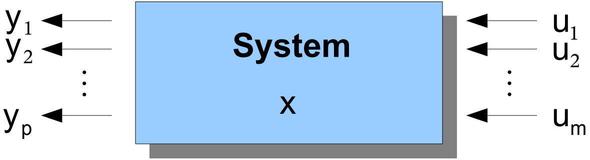
Model Reduction Using Multi-level Substructuring

Frank Blömeling
f.bloemeling@tu-harburg.de

Institute of Numerical Simulation

24.08.2007

Technische Universität Hamburg-Harburg



- system completely determined by the mapping

$$G : u \mapsto y$$

State space description

Linear time-invariant (LTI) dynamical system:

$$
\begin{aligned}
E\dot{x}(t) &= Ax(t) + Bu(t), \quad x(0) = 0 \\
y(t) &= Cx(t) + Du(t),
\end{aligned}
\tag{1}
$$

$A, E \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times m}$, $C \in \mathbb{R}^{p \times n}$ and $D \in \mathbb{R}^{p \times m}$

State space description

Linear time-invariant (LTI) dynamical system:

$$\begin{aligned} E\dot{x}(t) &= Ax(t) + Bu(t), \quad x(0) = 0 \\ y(t) &= Cx(t) + Du(t), \end{aligned} \tag{1}$$

$A, E \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, $C \in \mathbb{R}^{p\times n}$ and $D \in \mathbb{R}^{p\times m}$

- $u(t) \in \mathbb{R}^m$, $y(t) \in \mathbb{R}^p \Rightarrow$ m inputs, p outputs
- $x(t) \in \mathbb{R}^n \Rightarrow$ n state space variables (order)
- abbreviation:

$$G = \left[\begin{array}{c|c} sE - A & B \\ \hline C & D \end{array}\right]$$

State space description

Linear time-invariant (LTI) dynamical system:

$$\begin{aligned} E\dot{x}(t) &= Ax(t) + Bu(t), \quad x(0) = 0 \\ y(t) &= Cx(t) + Du(t), \end{aligned} \tag{1}$$

$A, E \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, $C \in \mathbb{R}^{p\times n}$ and $D \in \mathbb{R}^{p\times m}$

- $u(t) \in \mathbb{R}^m$, $y(t) \in \mathbb{R}^p \Rightarrow$ m inputs, p outputs
- $x(t) \in \mathbb{R}^n \Rightarrow$ n state space variables (order)
- abbreviation:

$$G = \left[\begin{array}{c|c} sE - A & B \\ \hline C & D \end{array}\right]$$

State space description

Linear time-invariant (LTI) dynamical system:

$$\begin{aligned} E\dot{x}(t) &= Ax(t) + Bu(t), \quad x(0) = 0 \\ y(t) &= Cx(t) + Du(t), \end{aligned} \tag{1}$$

$A, E \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times m}$, $C \in \mathbb{R}^{p \times n}$ and $D \in \mathbb{R}^{p \times m}$

- $u(t) \in \mathbb{R}^m$, $y(t) \in \mathbb{R}^p \Rightarrow$ m inputs, p outputs
- $x(t) \in \mathbb{R}^n \Rightarrow$ n state space variables (order)
- abbreviation:

$$G = \left[\begin{array}{c|c} sE - A & B \\ \hline C & D \end{array}\right]$$

Frequency domain

- Laplace transformation: $u \mapsto U(s) := \int_0^\infty e^{-st} u(t) dt$

System output is obtained by simple multiplication:

$$Y = G \cdot U$$

- transfer function:

$$G(s) = C(sE - A)^{-1} B + D$$

Model reduction

Problem: state space dimension n is much too large

$$\Downarrow$$

Replace system matrices

$$\left[\begin{array}{c|c} sE - A & B \\ \hline C & D \end{array}\right] \longrightarrow \left[\begin{array}{c|c} sE^{(k)} - A^{(k)} & B^{(k)} \\ \hline C^{(k)} & D \end{array}\right]$$

Model reduction

Problem: state space dimension n is much too large

$$\Downarrow$$

Replace system matrices

$$\left[\begin{array}{c|c} sE - A & B \\ \hline C & D \end{array}\right] \longrightarrow \left[\begin{array}{c|c} sE^{(k)} - A^{(k)} & B^{(k)} \\ \hline C^{(k)} & D \end{array}\right]$$

- reduced order k is much smaller
- reproduce fundamental input/output behaviour, i.e., $G \approx G^{(k)}$
- preserve essential system properties
- w.l.o.g. $D = 0$

Methods

- small-scale problems:
  - balancing methods (balanced truncation, stochastic balancing, positive real balancing, ...)
  - optimal Hankel norm approximation
  - proper orthogonal decomposition
- large-scale problems:
  - moment-matching methods (Arnoldi, Lanczos, Rational Krylov)
  - low-rank ADI iteration

Methods

- small-scale problems:
  - balancing methods (balanced truncation, stochastic balancing, positive real balancing, ...)
  - optimal Hankel norm approximation
  - proper orthogonal decomposition
- large-scale problems:
  - moment-matching methods (Arnoldi, Lanczos, Rational Krylov)
  - low-rank ADI iteration

What if application not possible or inefficient?

Methods

- small-scale problems:
  - balancing methods (balanced truncation, stochastic balancing, positive real balancing, ...)
  - optimal Hankel norm approximation
  - proper orthogonal decomposition
- large-scale problems:
  - moment-matching methods (Arnoldi, Lanczos, Rational Krylov)
  - low-rank ADI iteration

What if application not possible or inefficient?

Remedy

Combine your favourite method with multi-level substructuring!

Substructuring

Strategy:

1. Decompose large system in many small decoupled ones
2. Reduce solely the small problems

$$\Downarrow$$

Reduction methods are applicable!

Substructuring

Strategy:

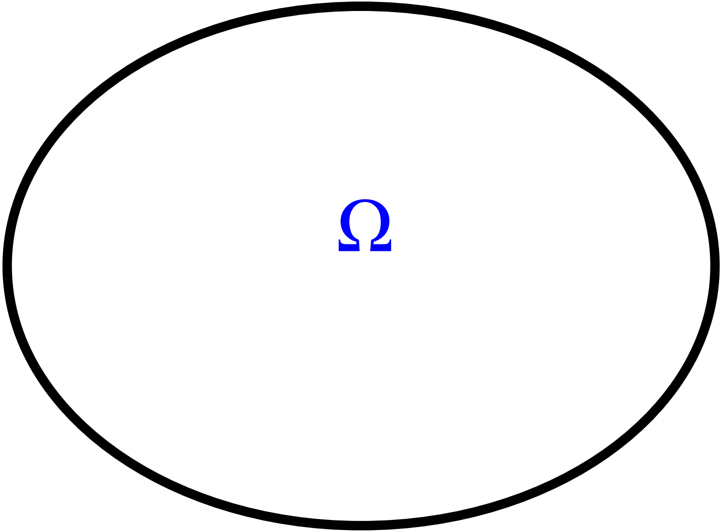
1. Decompose large system in many small decoupled ones
2. Reduce solely the small problems

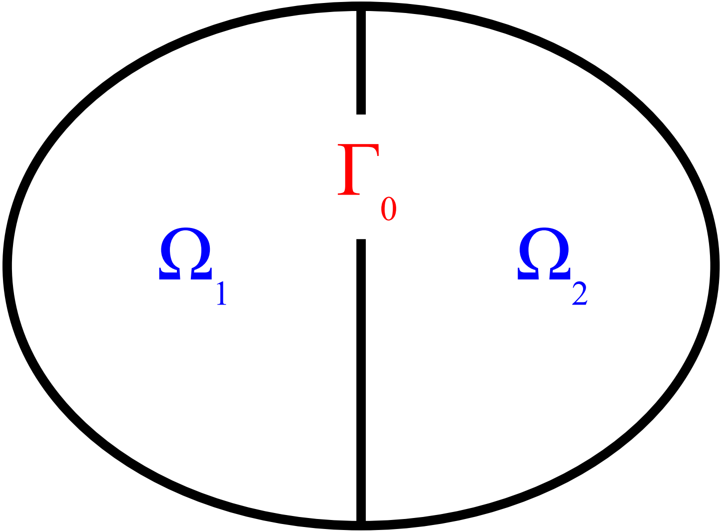
$$\Downarrow$$

Reduction methods are applicable!

Problems:

- decomposition not equivalent to overall system
- arising boundaries too large $\Rightarrow$ multi-level approach


Ω



Γ_0
Ω_1
Ω_2

Splitting of state space:

$$\mathbb{R}^n = \mathbb{R}^{n_1} \times \mathbb{R}^{n_2} \times \mathbb{R}^{n_\Gamma}$$

$$\Downarrow$$

$$A = \begin{pmatrix} A_1 & & A_{1,\Gamma} \\ & A_2 & A_{2,\Gamma} \\ A_{\Gamma,1} & A_{\Gamma,2} & A_\Gamma \end{pmatrix}, \quad E = \begin{pmatrix} E_1 & & E_{1,\Gamma} \\ & E_2 & E_{2,\Gamma} \\ E_{\Gamma,1} & E_{\Gamma,2} & E_\Gamma \end{pmatrix},$$

$$B = \begin{pmatrix} B_1 \\ B_2 \\ B_\Gamma \end{pmatrix}, \quad C = \begin{pmatrix} C_1 & C_2 & C_\Gamma \end{pmatrix}$$

- $n_\Gamma \ll n_i$, $i = 1, 2$

Transformation

State space transformation:

$$(\widehat{A}, \widehat{B}, \widehat{C}, \widehat{E}) = (LAR, LB, CR, LER)$$

$$L = \begin{pmatrix} I & & \\ & I & \\ -A_{\Gamma,1}A_1^{-1} & -A_{\Gamma,2}A_2^{-1} & I \end{pmatrix}, R = \begin{pmatrix} I & & -A_1^{-1}A_{1,\Gamma} \\ & I & -A_2^{-1}A_{2,\Gamma} \\ & & I \end{pmatrix}$$

Transformation

State space transformation:

$$(\widehat{A}, \widehat{B}, \widehat{C}, \widehat{E}) = (LAR, LB, CR, LER)$$

$$L = \begin{pmatrix} I & & \\ & I & \\ -A_{\Gamma,1}A_1^{-1} & -A_{\Gamma,2}A_2^{-1} & I \end{pmatrix}, R = \begin{pmatrix} I & & -A_1^{-1}A_{1,\Gamma} \\ & I & -A_2^{-1}A_{2,\Gamma} \\ & & I \end{pmatrix}$$

- equivalent realization of the system, i.e., $\left[\begin{array}{c|c} sE - A & B \\ \hline C & \end{array}\right] = \left[\begin{array}{c|c} s\widehat{E} - \widehat{A} & \widehat{B} \\ \hline \widehat{C} & \end{array}\right]$
- $\widehat{A}$ block-diagonal
- essential parts of matrices don't change
- twosided block-Gaussian elemination (never build L and R explicitely)

Subsystems

$$\widehat{A} = \begin{pmatrix} A_1 & & \\ & A_2 & \\ & & \widehat{A}_\Gamma \end{pmatrix}, \quad \widehat{E} = \begin{pmatrix} E_1 & & \widehat{E}_{1,\Gamma} \\ & E_2 & \widehat{E}_{2,\Gamma} \\ \widehat{E}_{\Gamma,1} & \widehat{E}_{\Gamma,2} & \widehat{E}_\Gamma \end{pmatrix}$$

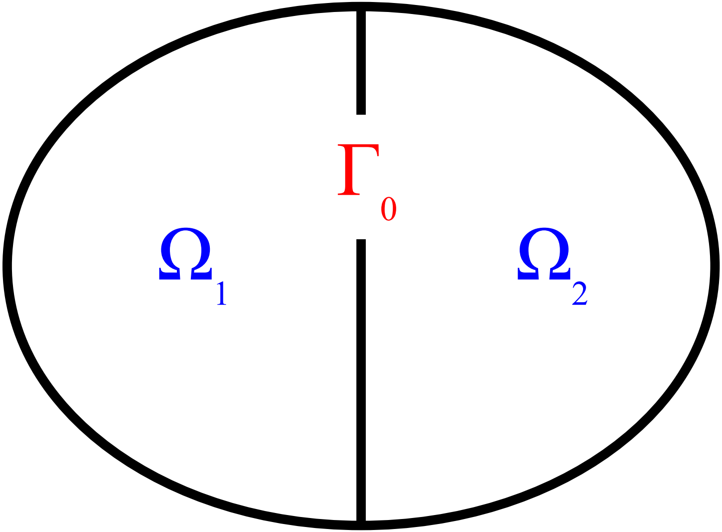
1. Subsystems for subdomains Ω_i, $i = 1, 2$:

$$G_i = \left[\begin{array}{c|c} sE_i - A_i & B_i \\ \hline C_i & \end{array} \right]$$

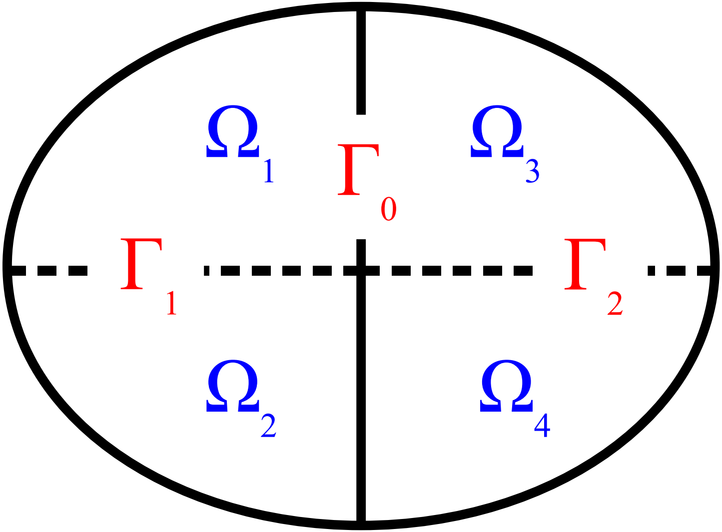
2. Subsystem for interface Γ:

$$G_\Gamma = \left[\begin{array}{c|c} s\widehat{E}_\Gamma - \widehat{A}_\Gamma & \widehat{B}_\Gamma \\ \hline \widehat{C}_\Gamma & \end{array} \right],$$

Multi-level



Multi-level



Nested arrow head structure

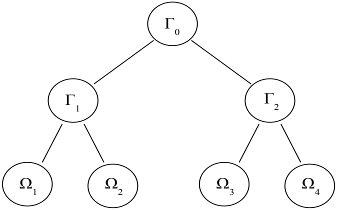
$$A = \left(\begin{array}{ccc|ccc|c}
A_1 & & A_{1,\Gamma_1} & & & & A_{1,\Gamma_0} \\
 & A_2 & A_{2,\Gamma_1} & & & & A_{2,\Gamma_0} \\
A_{\Gamma_1,1} & A_{\Gamma_1,2} & A_{\Gamma_1} & & & & A_{\Gamma_1,\Gamma_0} \\
\hline
 & & & A_3 & & A_{3,\Gamma_2} & A_{3,\Gamma_0} \\
 & & & & A_4 & A_{4,\Gamma_2} & A_{4,\Gamma_0} \\
 & & & A_{\Gamma_2,3} & A_{\Gamma_2,4} & A_{\Gamma_2} & A_{\Gamma_2,\Gamma_0} \\
\hline
A_{\Gamma_0,1} & A_{\Gamma_0,2} & A_{\Gamma_0,\Gamma_1} & A_{\Gamma_0,3} & A_{\Gamma_0,4} & A_{\Gamma_0,\Gamma_2} & A_{\Gamma_0}
\end{array}\right)$$

$$E = \left(\begin{array}{ccc|ccc|c}
E_1 & & E_{1,\Gamma_1} & & & & E_{1,\Gamma_0} \\
 & E_2 & E_{2,\Gamma_1} & & & & E_{2,\Gamma_0} \\
E_{\Gamma_1,1} & E_{\Gamma_1,2} & E_{\Gamma_1} & & & & E_{\Gamma_1,\Gamma_0} \\
\hline
 & & & E_3 & & E_{3,\Gamma_2} & E_{3,\Gamma_0} \\
 & & & & E_4 & E_{4,\Gamma_2} & E_{4,\Gamma_0} \\
 & & & E_{\Gamma_2,3} & E_{\Gamma_2,4} & E_{\Gamma_2} & E_{\Gamma_2,\Gamma_0} \\
\hline
E_{\Gamma_0,1} & E_{\Gamma_0,2} & E_{\Gamma_0,\Gamma_1} & E_{\Gamma_0,3} & E_{\Gamma_0,4} & E_{\Gamma_0,\Gamma_2} & E_{\Gamma_0}
\end{array}\right)$$

Nested arrow head structure

$$\widehat{A} = \left(\begin{array}{ccc|ccc|c} A_1 & & & & & & \\ & A_2 & & & & & \\ & & \widehat{A}_{\Gamma_1} & & & & \\ \hline & & & A_3 & & & \\ & & & & A_4 & & \\ & & & & & \widehat{A}_{\Gamma_2} & \\ \hline & & & & & & \widehat{A}_{\Gamma_0} \end{array}\right)$$

$$\widehat{E} = \left(\begin{array}{ccc|ccc|c} E_1 & & \widehat{E}_{1,\Gamma_1} & & & & \widehat{E}_{1,\Gamma_0} \\ & E_2 & \widehat{E}_{2,\Gamma_1} & & & & \widehat{E}_{2,\Gamma_0} \\ \widehat{E}_{\Gamma_1,1} & \widehat{E}_{\Gamma_1,2} & \widehat{E}_{\Gamma_1} & & & & \widehat{E}_{\Gamma_1,\Gamma_0} \\ \hline & & & E_3 & & \widehat{E}_{3,\Gamma_2} & \widehat{E}_{3,\Gamma_0} \\ & & & & E_4 & \widehat{E}_{4,\Gamma_2} & \widehat{E}_{4,\Gamma_0} \\ & & & \widehat{E}_{\Gamma_2,3} & \widehat{E}_{\Gamma_2,4} & \widehat{E}_{\Gamma_2} & \widehat{E}_{\Gamma_2,\Gamma_0} \\ \hline \widehat{E}_{\Gamma_0,1} & \widehat{E}_{\Gamma_0,2} & \widehat{E}_{\Gamma_0,\Gamma_1} & \widehat{E}_{\Gamma_0,3} & \widehat{E}_{\Gamma_0,4} & \widehat{E}_{\Gamma_0,\Gamma_2} & \widehat{E}_{\Gamma_0} \end{array}\right)$$



1. Subdomains:

$$\left[\begin{array}{c|c} sE_i - A_i & B_i \\ \hline C_i & \end{array}\right], \quad i = 1 \ldots, 4$$

2. Interfaces:

$$\left[\begin{array}{c|c} s\widehat{E}_{\Gamma_j} - \widehat{A}_{\Gamma_j} & \widehat{B}_{\Gamma_j} \\ \hline \widehat{C}_{\Gamma_j} & \end{array}\right], \quad j = 0, 1, 2$$

- partitioning is done automatically by a graph partitioner

Properties of the decomposition

- exact at zero, i.e.,

$$G(0) = \sum_i G_i(0) + \sum_j G_{\Gamma_j}(0)$$

- decomposition of the state space is A-orthogonal in the symmetric case
- derivation from continuous settings $\Rightarrow$ smoothness condition on interfaces
- "small" parts of the matrix $\widehat{E}$ are ignored

MLS model order reduction

Principle steps:

1. Build subsystems by using multi-level substructuring
2. Reduce the subsystems (including interfaces), i.e., determine V_i and W_i s.t.

$$G_i^{(k_i)} = \left[\begin{array}{c|c} sW_i^H \widehat{E}_{i,i} V_i - W_i^H \widehat{A}_{i,i} V_i & W_i^H \widehat{B}_i \\ \hline \widehat{C}_i V_i & \end{array} \right]$$

3. Project the transformed original system onto

$$\mathcal{V} := \text{colspan}\{V_1\} \times \text{colspan}\{V_2\} \times \ldots,$$

i.e.,

$$\begin{aligned} A_{i,i}^{(k)} &= W_i^H \widehat{A}_{i,i} V_i, \\ E_{i,j}^{(k)} &= W_i^H \widehat{E}_{i,j} V_j, \\ B_i^{(k)} &= W_i^H \widehat{B}_i, \\ C_i^{(k)} &= \widehat{C}_i V_i \end{aligned}$$

MLS model order reduction

Principle steps:

1. Build subsystems by using multi-level substructuring
2. Reduce the subsystems (including interfaces), i.e., determine V_i and W_i s.t.

$$G_i^{(k_i)} = \left[\begin{array}{c|c} sW_i^H \widehat{E}_{i,i} V_i - W_i^H \widehat{A}_{i,i} V_i & W_i^H \widehat{B}_i \\ \hline \widehat{C}_i V_i & \end{array}\right]$$

3. Project the transformed original system onto

$$\mathcal{V} := \text{colspan}\{V_1\} \times \text{colspan}\{V_2\} \times \ldots,$$

i.e.,

$$\begin{aligned} A_{i,i}^{(k)} &= W_i^H \widehat{A}_{i,i} V_i, \\ E_{i,j}^{(k)} &= W_i^H \widehat{E}_{i,j} V_j, \\ B_i^{(k)} &= W_i^H \widehat{B}_i, \\ C_i^{(k)} &= \widehat{C}_i V_i \end{aligned}$$

TUHH

MLS model order reduction

Principle steps:

1. Build subsystems by using multi-level substructuring
2. Reduce the subsystems (including interfaces), i.e., determine V_i and W_i s.t.

$$G_i^{(k_i)} = \left[\begin{array}{c|c} sW_i^H \widehat{E}_{i,i} V_i - W_i^H \widehat{A}_{i,i} V_i & W_i^H \widehat{B}_i \\ \hline \widehat{C}_i V_i & \end{array}\right]$$

3. Project the transformed original system onto

$$\mathcal{V} := \text{colspan}\{V_1\} \times \text{colspan}\{V_2\} \times \dots,$$

i.e.,

$$\begin{aligned} A_{i,i}^{(k)} &= W_i^H \widehat{A}_{i,i} V_i, \\ E_{i,j}^{(k)} &= W_i^H \widehat{E}_{i,j} V_j, \\ B_i^{(k)} &= W_i^H \widehat{B}_i, \\ C_i^{(k)} &= \widehat{C}_i V_i \end{aligned}$$

Properties

- $k = \sum_i k_i$
- free choice of reduction method
- applicable in particular if other methods fail due to problem dimension
- multi-level substructuring improves performance for very large systems
- sparse B or $C \Rightarrow$ many subsystems may be neglected

Error sources

Two error sources:

1. Reduction of subsystems $\leftarrow$ controlled by reduction method
2. Multi-level substructuring

- error sources sum up
- error due to MLS depends on the projection error of the coupling blocks

Error sources

Two error sources:

1. Reduction of subsystems $\leftarrow$ controlled by reduction method
2. Multi-level substructuring

- error sources sum up
- error due to MLS depends on the projection error of the coupling blocks

Consequence

Error may be large even if the subsystems are well approximated!

Error sources

Two error sources:

1. Reduction of subsystems $\leftarrow$ controlled by reduction method
2. Multi-level substructuring

- error sources sum up
- error due to MLS depends on the projection error of the coupling blocks

Consequence

Error may be large even if the subsystems are well approximated!

- usually not the case in practical applications

Remark on reduction by Krylov subspace methods

Problem:

$$\mathcal{V} := \text{colspan}\{V_1\} \times \text{colspan}\{V_2\} \times \dots$$

gets too large

Remedy:

1. Perform MLS model order reduction
2. Apply the Krylov method a second time to $(A^{(k)}, B^{(k)}, C^{(k)}, E^{(k)})$

Second-order systems

$$\begin{aligned} E\ddot{x}(t) + D\dot{x}(t) + Ax(t) &= Bu(t) \\ y(t) &= Cx(t), \end{aligned}$$

- structure preserving model reduction method
- avoid enlargement of the state space

MLS model order reduction (second-order systems)

1. Partition the state space recursively
2. State space transformation s.t. A is block-diagonal
3. Build subsystems from diagonal blocks
4. Apply second-order reduction method to subsystems
5. Project the overall system

MLS model order reduction (second-order systems)

1. Partition the state space recursively
2. State space transformation s.t. A is block-diagonal
3. Build subsystems from diagonal blocks
4. Apply second-order reduction method to subsystems
5. Project the overall system

MLS model order reduction (second-order systems)

1. Partition the state space recursively
2. State space transformation s.t. A is block-diagonal
3. Build subsystems from diagonal blocks
4. Apply second-order reduction method to subsystems
5. Project the overall system

MLS model order reduction (second-order systems)

1. Partition the state space recursively
2. State space transformation s.t. A is block-diagonal
3. Build subsystems from diagonal blocks
4. Apply second-order reduction method to subsystems
5. Project the overall system

MLS model order reduction (second-order systems)

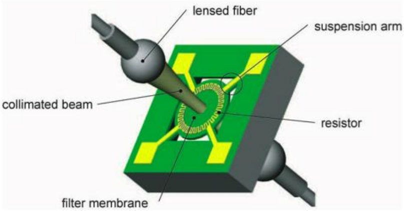
1. Partition the state space recursively
2. State space transformation s.t. A is block-diagonal
3. Build subsystems from diagonal blocks
4. Apply second-order reduction method to subsystems
5. Project the overall system

MLS model order reduction (second-order systems)

1. Partition the state space recursively
2. State space transformation s.t. A is block-diagonal
3. Build subsystems from diagonal blocks
4. Apply second-order reduction method to subsystems
5. Project the overall system

- equivalent to first-order version applied to system in companion form
- reduced system has second-order structure
- other structure may be destroyed

Tunable optical filter



- optical filter device tuned by thermal means
- spatial discretization of the instationary heat equation yields

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

- 106.437 state variables, 1 input and 5 outputs

Partitioning

max. order	SpD	interfaces	domains	levels
500	2	236	237	8

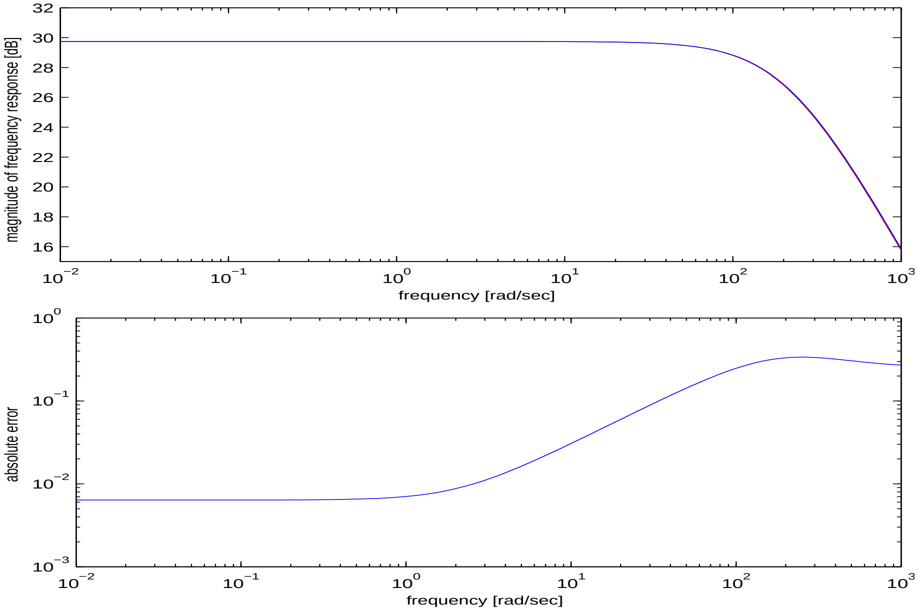
- reduction of subsystems by using balanced truncation
- absolute error tolerance: 10^{-2}

Performance

mem	t_{par}	t_{tr}	t_{red}	t_{total}
480 MB	697 sec	489 sec	438 sec	1624 sec

- 457/473 subsystems dropped

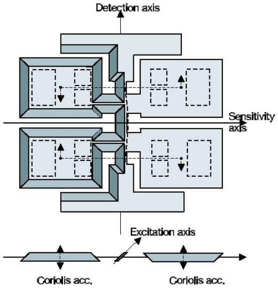
reduced order: 15


magnitude of frequency response [dB]
32
30
28
26
24
22
20
18
16
10^{-2}
10^{-1}
10^{0}
10^{1}
10^{2}
10^{3}
frequency [rad/sec]
absolute error
10^{0}
10^{-1}
10^{-2}
10^{-3}
10^{-2}
10^{-1}
10^{0}
10^{1}
10^{2}
10^{3}
frequency [rad/sec]

Butterfly gyro

- micro-mechanical gyroscope
- spatial discretization of elastodynamic equations yields

$$\begin{aligned} E\ddot{x}(t) + D\dot{x}(t) + Ax(t) &= Bu(t), \\ y(t) &= Cx(t) \end{aligned}$$



- 17.361 state variables, 1 input and 12 outputs

Partitioning

max. order	SpD	interfaces	domains	levels
200	2	80	79	7

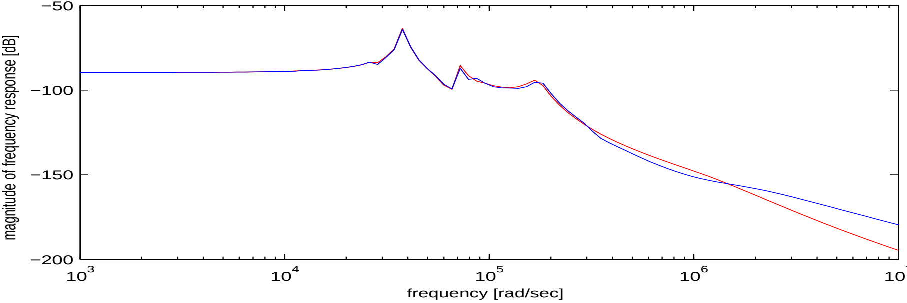
- reduction of subsystems by using SOBT

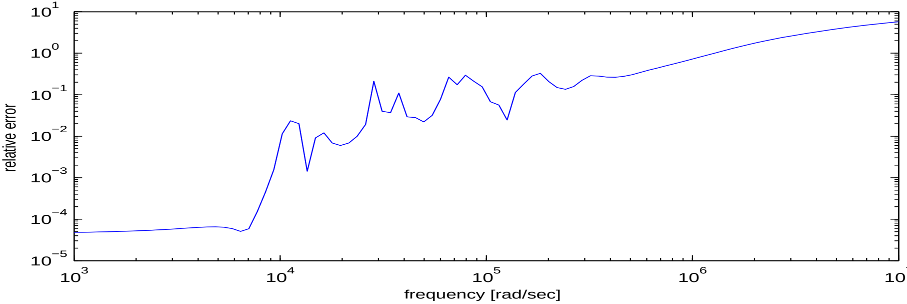
Performance

mem	t_{par}	t_{tr}	t_{red}	t_{total}
330 MB	144 sec	49 sec	12 sec	205 sec

- 151/159 subsystems dropped

reduced order: 40


−50
−100
−150
−200
magnitude of frequency response [dB]
10^3
10^4
10^5
10^6
10^7
frequency [rad/sec]


10^1
10^0
10^{-1}
10^{-2}
10^{-3}
10^{-4}
10^{-5}
relative error
10^3
10^4
10^5
10^6
10^7
frequency [rad/sec]