Greybox Models and Model Reduction
ITWM Kaiserslautern
18 December 2013

Mathematical Methods for Model Order Reduction of Linear and Nonlinear Systems

Peter Benner

Max Planck Institute for Dynamics of Complex Technical Systems
Computational Methods in Systems and Control Theory
Magdeburg, Germany

benner@mpi-magdeburg.mpg.de

Outline

1. Introduction
2. Model Reduction by Projection
3. Interpolatory Model Reduction
4. Balanced Truncation
5. Nonlinear Model Reduction
6. Final Remarks

Outline

1. Introduction
   - Model Reduction for Dynamical Systems
   - Motivating Examples
   - Some Background
   - Qualitative and Quantitative Study of the Approximation Error
2. Model Reduction by Projection
3. Interpolatory Model Reduction
4. Balanced Truncation
5. Nonlinear Model Reduction
6. Final Remarks

Introduction

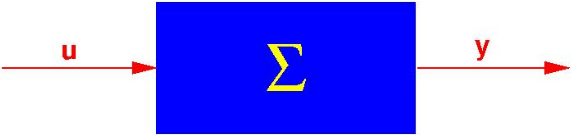
Model Reduction for Dynamical Systems

Dynamical Systems

$$\Sigma : \begin{cases} \dot{x}(t) &= f(t, x(t), u(t)), \quad x(t_0) = x_0, \\ y(t) &= g(t, x(t), u(t)) \end{cases}$$

with

- states $x(t) \in \mathbb{R}^n$,
- inputs $u(t) \in \mathbb{R}^m$,
- outputs $y(t) \in \mathbb{R}^q$.

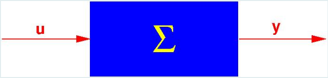


Model Reduction for Dynamical Systems

Original System

$$\Sigma : \begin{cases} \dot{x}(t) = f(t, x(t), u(t)), \\ y(t) = g(t, x(t), u(t)). \end{cases}$$

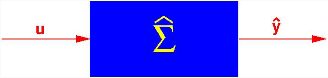
- states $x(t) \in \mathbb{R}^n$,
- inputs $u(t) \in \mathbb{R}^m$,
- outputs $y(t) \in \mathbb{R}^q$.



Reduced-Order Model (ROM)

$$\widehat{\Sigma} : \begin{cases} \dot{\hat{x}}(t) = \widehat{f}(t, \hat{x}(t), u(t)), \\ \hat{y}(t) = \widehat{g}(t, \hat{x}(t), u(t)). \end{cases}$$

- states $\hat{x}(t) \in \mathbb{R}^r$, $r \ll n$
- inputs $u(t) \in \mathbb{R}^m$,
- outputs $\hat{y}(t) \in \mathbb{R}^q$.



Goal:

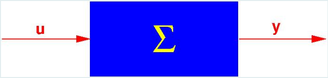
$\|y - \hat{y}\| <$ tolerance $\cdot \|u\|$ for all admissible input signals.

Model Reduction for Dynamical Systems

Original System

$$\Sigma : \begin{cases} \dot{x}(t) = f(t, x(t), u(t)), \\ y(t) = g(t, x(t), u(t)). \end{cases}$$

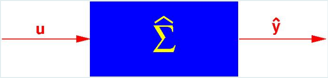
- states $x(t) \in \mathbb{R}^n$,
- inputs $u(t) \in \mathbb{R}^m$,
- outputs $y(t) \in \mathbb{R}^q$.



Reduced-Order Model (ROM)

$$\widehat{\Sigma} : \begin{cases} \dot{\hat{x}}(t) = \widehat{f}(t, \hat{x}(t), u(t)), \\ \hat{y}(t) = \widehat{g}(t, \hat{x}(t), u(t)). \end{cases}$$

- states $\hat{x}(t) \in \mathbb{R}^r$, $r \ll n$
- inputs $u(t) \in \mathbb{R}^m$,
- outputs $\hat{y}(t) \in \mathbb{R}^q$.



Goal:

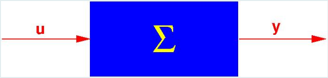
$\|y - \hat{y}\| <$ tolerance $\cdot \|u\|$ for all admissible input signals.

Model Reduction for Dynamical Systems

Original System

$$\Sigma : \begin{cases} \dot{x}(t) = f(t, x(t), u(t)), \\ y(t) = g(t, x(t), u(t)). \end{cases}$$

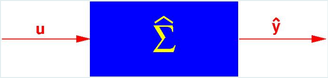
- states $x(t) \in \mathbb{R}^n$,
- inputs $u(t) \in \mathbb{R}^m$,
- outputs $y(t) \in \mathbb{R}^q$.



Reduced-Order Model (ROM)

$$\widehat{\Sigma} : \begin{cases} \dot{\hat{x}}(t) = \widehat{f}(t, \hat{x}(t), u(t)), \\ \hat{y}(t) = \widehat{g}(t, \hat{x}(t), u(t)). \end{cases}$$

- states $\hat{x}(t) \in \mathbb{R}^r$, $r \ll n$
- inputs $u(t) \in \mathbb{R}^m$,
- outputs $\hat{y}(t) \in \mathbb{R}^q$.



Goal:

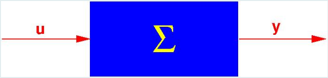
$\|y - \hat{y}\| < \text{tolerance} \cdot \|u\|$ for all admissible input signals.

Model Reduction for Dynamical Systems

Original System

$$\Sigma : \begin{cases} \dot{x}(t) = f(t, x(t), u(t)), \\ y(t) = g(t, x(t), u(t)). \end{cases}$$

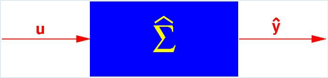
- states $x(t) \in \mathbb{R}^n$,
- inputs $u(t) \in \mathbb{R}^m$,
- outputs $y(t) \in \mathbb{R}^q$.



Reduced-Order Model (ROM)

$$\widehat{\Sigma} : \begin{cases} \dot{\hat{x}}(t) = \widehat{f}(t, \hat{x}(t), u(t)), \\ \hat{y}(t) = \widehat{g}(t, \hat{x}(t), u(t)). \end{cases}$$

- states $\hat{x}(t) \in \mathbb{R}^r$, $r \ll n$
- inputs $u(t) \in \mathbb{R}^m$,
- outputs $\hat{y}(t) \in \mathbb{R}^q$.



Goal:

$\|y - \hat{y}\| <$ tolerance $\cdot \|u\|$ for all admissible input signals.

Secondary goal: reconstruct approximation of x from $\hat{x}$.

Model Reduction for Dynamical Systems

Parameter-Dependent Dynamical Systems

Dynamical Systems

$$\Sigma(p): \begin{cases} E(p)\dot{x}(t;p) = f(t, x(t;p), u(t), p), \quad x(t_0) = x_0, & \text{(a)} \\ y(t;p) = g(t, x(t;p), u(t), p) & \text{(b)} \end{cases}$$

with

- (generalized) states $x(t;p) \in \mathbb{R}^n$ ($E \in \mathbb{R}^{n \times n}$),
- inputs $u(t) \in \mathbb{R}^m$,
- outputs $y(t;p) \in \mathbb{R}^q$, (b) is called output equation,
- $p \in \Omega \subset \mathbb{R}^d$ is a parameter vector, Ω is bounded.

Applications:

- Repeated simulation for varying material or geometry parameters, boundary conditions,
- Control, optimization and design.

Requirement: keep parameters as symbolic quantities in ROM.

Model Reduction for Dynamical Systems

Parameter-Dependent Dynamical Systems

Dynamical Systems

$$\Sigma(p) : \begin{cases} E(p)\dot{x}(t;p) &= f(t, x(t;p), u(t), p), \quad x(t_0) = x_0, & \text{(a)} \\ y(t;p) &= g(t, x(t;p), u(t), p) & \text{(b)} \end{cases}$$

with

- (generalized) states $x(t;p) \in \mathbb{R}^n$ ($E \in \mathbb{R}^{n \times n}$),
- inputs $u(t) \in \mathbb{R}^m$,
- outputs $y(t;p) \in \mathbb{R}^q$, (b) is called output equation,
- $p \in \Omega \subset \mathbb{R}^d$ is a parameter vector, Ω is bounded.

Applications:

- Repeated simulation for varying material or geometry parameters, boundary conditions,
- Control, optimization and design.

Requirement: keep parameters as symbolic quantities in ROM.

Model Reduction for Dynamical Systems

Linear Systems

Linear, Time-Invariant (LTI) Systems

$$
\begin{array}{rclclll}
E\dot{x} & = & f(t,x,u) & = & Ax + Bu, & E, A \in \mathbb{R}^{n\times n}, & B \in \mathbb{R}^{n\times m}, \\
y & = & g(t,x,u) & = & Cx + Du, & C \in \mathbb{R}^{q\times n}, & D \in \mathbb{R}^{q\times m}.
\end{array}
$$

Model Reduction for Dynamical Systems

Linear Systems

Linear, Time-Invariant (LTI) Systems

$$\begin{array}{rclcllr} E\dot{x} & = & f(t,x,u) & = & Ax+Bu, & E,A\in\mathbb{R}^{n\times n}, & B\in\mathbb{R}^{n\times m},\\ y & = & g(t,x,u) & = & Cx+Du, & C\in\mathbb{R}^{q\times n}, & D\in\mathbb{R}^{q\times m}. \end{array}$$

Linear, Time-Invariant Parametric Systems

$$\begin{array}{rcl} E(p)\dot{x}(t;p) & = & A(p)x(t;p)+B(p)u(t),\\ y(t;p) & = & C(p)x(t;p)+D(p)u(t), \end{array}$$

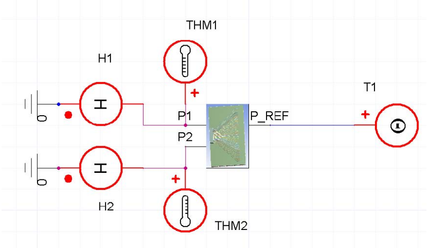
where $A(p), E(p) \in \mathbb{R}^{n\times n}, B(p) \in \mathbb{R}^{n\times m}, C(p) \in \mathbb{R}^{q\times n}, D(p) \in \mathbb{R}^{q\times m}$.

Motivating Examples

Electro-Thermic Simulation of Integrated Circuit (IC)

[Source: Evgenii Rudnyi, CADFEM GmbH]

- SIMPLORER® test circuit with 2 transistors.



- Conservative thermic sub-system in SIMPLORER:
  voltage ⇝ temperature, current ⇝ heat flow.
- Original model: $n = 270,593$, $m = q = 2 \Rightarrow$
  Computing time (on Intel Xeon dualcore 3GHz, 1 Thread):
  - Main computational cost for set-up data $\approx 22min$.
  - Computation of reduced models from set-up data: 44–49sec. ($r = 20$–70).
  - Bode plot (MATLAB on Intel Core i7, 2,67GHz, 12GB):
    7.5h for original system, $<$ 1min for reduced system.
  - Speed-up factor: 18 including / $\geq$ 450 excluding reduced model generation!

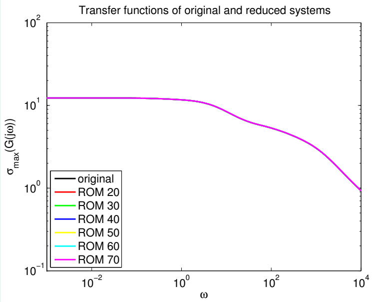
Motivating Examples

Electro-Thermic Simulation of Integrated Circuit (IC)

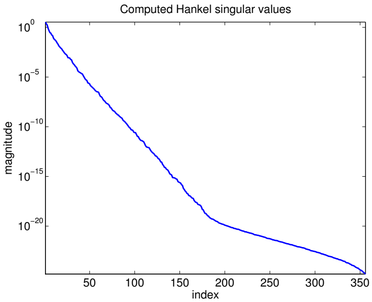
[Source: Evgenii Rudnyi, CADFEM GmbH]

- Original model: $n = 270,593$, $m = q = 2 \Rightarrow$
  Computing time (on Intel Xeon dualcore 3GHz, 1 Thread):
  - Main computational cost for set-up data $\approx 22min$.
  - Computation of reduced models from set-up data: 44–49sec. ($r = 20$–70).
  - Bode plot (MATLAB on Intel Core i7, 2,67GHz, 12GB):
    7.5h for original system, $<$ 1min for reduced system.
  - Speed-up factor: 18 including / $\geq$ 450 excluding reduced model generation!

Bode Plot (Amplitude)



Hankel Singular Values



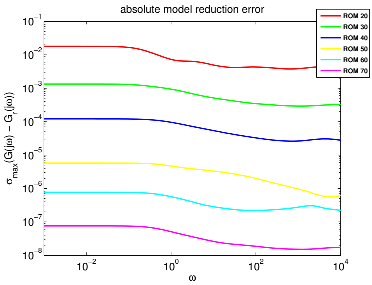
Motivating Examples

Electro-Thermic Simulation of Integrated Circuit (IC)

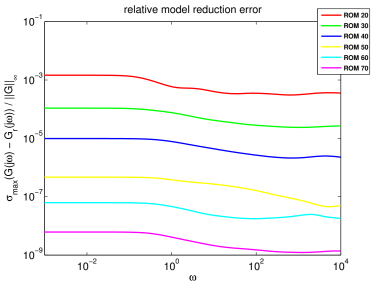
[Source: Evgenii Rudnyi, CADFEM GmbH]

- Original model: $n = 270,593$, $m = q = 2 \Rightarrow$
  Computing time (on Intel Xeon dualcore 3GHz, 1 Thread):
  - Main computational cost for set-up data $\approx 22min$.
  - Computation of reduced models from set-up data: 44–49sec. ($r = 20$–70).
  - Bode plot (MATLAB on Intel Core i7, 2,67GHz, 12GB):
    7.5h for original system, $<$ 1min for reduced system.
  - Speed-up factor: 18 including / $\geq$ 450 excluding reduced model generation!

Absolute Error



Relative Error



Motivating Examples

A Nonlinear Model from Computational Neurosciences: the FitzHugh-Nagumo System

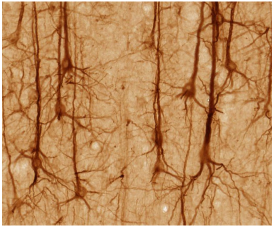
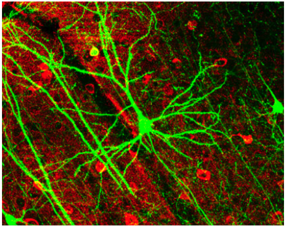
- Simple model for neuron (de-)activation [Chaturantabut/Sorensen 2009]

$$\epsilon v_t(x,t) = \epsilon^2 v_{xx}(x,t) + f(v(x,t)) - w(x,t) + g,$$
$$w_t(x,t) = hv(x,t) - \gamma w(x,t) + g,$$

with $f(v) = v(v-0.1)(1-v)$ and initial and boundary conditions

$$\begin{aligned} v(x,0) &= 0, & w(x,0) &= 0, & x &\in [0,1] \\ v_x(0,t) &= -i_0(t), & v_x(1,t) &= 0, & t &\geq 0, \end{aligned}$$

where $\epsilon = 0.015,\ h = 0.5,\ \gamma = 2,\ g = 0.05,\ i_0(t) = 50{,}000t^3 \exp(-15t)$.

Source: http://en.wikipedia.org/wiki/Neuron

Motivating Examples

A Nonlinear Model from Computational Neurosciences: the FitzHugh-Nagumo System

- Simple model for neuron (de-)activation [Chaturantabut/Sorensen 2009]

$$\begin{aligned} \epsilon v_t(x,t) &= \epsilon^2 v_{xx}(x,t) + f(v(x,t)) - w(x,t) + g, \\ w_t(x,t) &= hv(x,t) - \gamma w(x,t) + g, \end{aligned}$$

  with $f(v) = v(v-0.1)(1-v)$ and initial and boundary conditions

$$\begin{aligned} &v(x,0) = 0, && w(x,0) = 0, && x \in [0,1] \\ &v_x(0,t) = -i_0(t), && v_x(1,t) = 0, && t \geq 0, \end{aligned}$$

  where $\epsilon = 0.015,\ h = 0.5,\ \gamma = 2,\ g = 0.05,\ i_0(t) = 50{,}000 t^3 \exp(-15t)$.

- Parameter g handled as an additional input.
- Original state dimension $n = 2 \cdot 400$, QBDAE dimension $N = 3 \cdot 400$, reduced QBDAE dimension $r = 26$, chosen expansion point $\sigma = 1$.

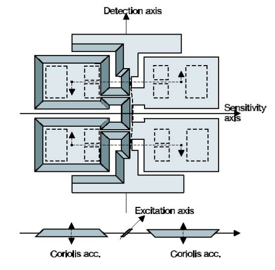
Motivating Examples

A Nonlinear Model from Computational Neurosciences: the FitzHugh-Nagumo System

Motivating Examples
Parametric MOR: Applications in Microsystems/MEMS Design

Microgyroscope (butterfly gyro)

- Application: inertial navigation.



- Voltage applied to electrodes induces vibration of wings, resulting rotation due to Coriolis force yields sensor data.
- FE model of second order: $N = 17.361 \rightsquigarrow n = 34.722,\ m = 1,\ q = 12.$
- Sensor for position control based on acceleration and rotation.

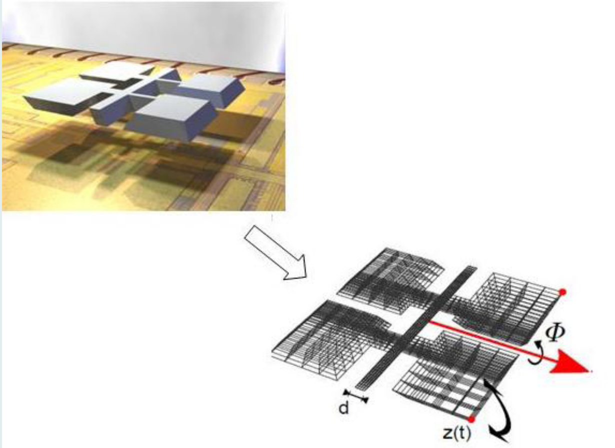
Source: The Oberwolfach Benchmark Collection http://www.imtek.de/simulation/benchmark

Motivating Examples

Parametric MOR: Applications in Microsystems/MEMS Design

Microgyroscope (butterfly gyro)

Parametric FE model: $M(d)\ddot{x}(t) + D(\Phi, d, \alpha, \beta)\dot{x}(t) + T(d)x(t) = Bu(t).$



Motivating Examples

Parametric MOR: Applications in Microsystems/MEMS Design

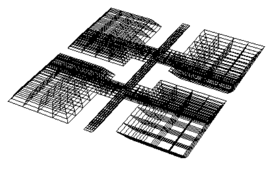
Microgyroscope (butterfly gyro)

Parametric FE model:

$$M(d)\ddot{x}(t) + D(\Phi, d, \alpha, \beta)\dot{x}(t) + T(d)x(t) = Bu(t),$$

wobei

$$\begin{aligned} M(d) &= M_1 + dM_2, \\ D(\Phi, d, \alpha, \beta) &= \Phi(D_1 + dD_2) + \alpha M(d) + \beta T(d), \\ T(d) &= T_1 + \frac{1}{d}T_2 + dT_3, \end{aligned}$$



with

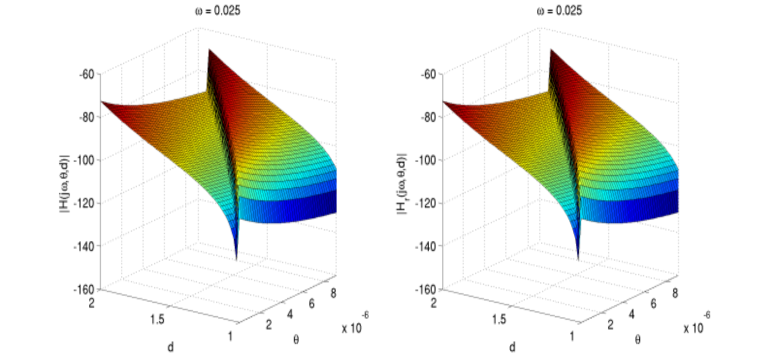
- width of bearing: d,
- angular velocity: Φ,
- Rayleigh damping parameters: α, β.

Motivating Examples

Parametric MOR: Applications in Microsystems/MEMS Design

Microgyroscope (butterfly gyro)

Original. . . and reduced-order model.



Some Background
The Laplace transform

Definition

The Laplace transform of a time domain function $f \in L_{1,\mathrm{loc}}$ with $\mathrm{dom}\,(f) = \mathbb{R}_0^+$ is

$$\mathcal{L} : f \mapsto F, \quad F(s) := \mathcal{L}\{f(t)\}(s) := \int_0^\infty e^{-st} f(t)\, dt, \quad s \in \mathbb{C}.$$

F is a function in the (Laplace or) frequency domain.

Note: for frequency domain evaluations ("frequency response analysis"), one takes $\mathrm{re}\, s = 0$ and $\mathrm{im}\, s \geq 0$. Then $\omega := \mathrm{im}\, s$ takes the role of a frequency (in [rad/s], i.e., $\omega = 2\pi v$ with v measured in [Hz]).

Some Background
The Laplace transform

Definition

The Laplace transform of a time domain function $f \in L_{1,\mathrm{loc}}$ with $\mathrm{dom}\,(f) = \mathbb{R}_0^+$ is

$$\mathcal{L} : f \mapsto F, \quad F(s) := \mathcal{L}\{f(t)\}(s) := \int_0^\infty e^{-st} f(t)\,dt, \quad s \in \mathbb{C}.$$

F is a function in the (Laplace or) frequency domain.

Note: for frequency domain evaluations ("frequency response analysis"), one takes $\mathrm{re}\, s = 0$ and $\mathrm{im}\, s \geq 0$. Then $\omega := \mathrm{im}\, s$ takes the role of a frequency (in [rad/s], i.e., $\omega = 2\pi v$ with v measured in [Hz]).

Lemma

$$\mathcal{L}\{\dot{f}(t)\}(s) = sF(s).$$

Some Background
The Laplace transform

Definition

The Laplace transform of a time domain function $f \in L_{1,\mathrm{loc}}$ with $\mathrm{dom}\,(f) = \mathbb{R}_0^+$ is

$$\mathcal{L} : f \mapsto F, \quad F(s) := \mathcal{L}\{f(t)\}(s) := \int_0^\infty e^{-st} f(t)\,dt, \quad s \in \mathbb{C}.$$

F is a function in the (Laplace or) frequency domain.

Lemma

$$\mathcal{L}\{\dot{f}(t)\}(s) = sF(s).$$

Note: for ease of notation, in the following we will use lower-case letters for both, a function and its Laplace transform!

Some Background

The Model Reduction Problem as Approximation Problem in Frequency Domain

Linear Systems in Frequency Domain

Application of Laplace transform $(x(t) \mapsto x(s),\ \dot{x}(t) \mapsto sx(s))$ to linear system

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

with $x(0) = 0$ yields:

$$sEx(s) = Ax(s) + Bu(s), \quad y(s) = Cx(s) + Du(s),$$

Some Background

The Model Reduction Problem as Approximation Problem in Frequency Domain

Linear Systems in Frequency Domain

Application of Laplace transform $(x(t) \mapsto x(s),\ \dot{x}(t) \mapsto sx(s))$ to linear system

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

with $x(0) = 0$ yields:

$$sEx(s) = Ax(s) + Bu(s), \quad y(s) = Cx(s) + Du(s),$$

$\Longrightarrow$ I/O-relation in frequency domain:

$$y(s) = \Big(\underbrace{C(sE - A)^{-1}B + D}_{=:G(s)} \Big) u(s).$$

$G(s)$ is the transfer function of Σ.

Some Background

The Model Reduction Problem as Approximation Problem in Frequency Domain

Linear Systems in Frequency Domain

Application of Laplace transform $(x(t) \mapsto x(s),\ \dot{x}(t) \mapsto sx(s))$ to linear system

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

with $x(0) = 0$ yields:

$$sEx(s) = Ax(s) + Bu(s), \quad y(s) = Cx(s) + Du(s),$$

$\Longrightarrow$ I/O-relation in frequency domain:

$$y(s) = \Big(\underbrace{C(sE - A)^{-1}B + D}_{=:G(s)} \Big) u(s).$$

$G(s)$ is the transfer function of Σ.

Goal: Fast evaluation of mapping $u \rightarrow y$.

Some Background

The Model Reduction Problem as Approximation Problem in Frequency Domain

Formulating model reduction in frequency domain

Approximate the dynamical system

$$\begin{array}{rcll} E\dot{x} &=& Ax + Bu, & E, A \in \mathbb{R}^{n\times n},\ B \in \mathbb{R}^{n\times m}, \\ y &=& Cx + Du, & C \in \mathbb{R}^{q\times n},\ D \in \mathbb{R}^{q\times m}, \end{array}$$

by reduced-order system

$$\begin{array}{rcll} \hat{E}\dot{\hat{x}} &=& \hat{A}\hat{x} + \hat{B}u, & \hat{E}, \hat{A} \in \mathbb{R}^{r\times r},\ \hat{B} \in \mathbb{R}^{r\times m}, \\ \hat{y} &=& \hat{C}\hat{x} + \hat{D}u, & \hat{C} \in \mathbb{R}^{q\times r},\ \hat{D} \in \mathbb{R}^{q\times m} \end{array}$$

of order $r \ll n$, such that

$$\|y - \hat{y}\| = \|Gu - \hat{G}u\| \leq \|G - \hat{G}\| \cdot \|u\| < \text{tolerance} \cdot \|u\|.$$

Some Background

The Model Reduction Problem as Approximation Problem in Frequency Domain

Formulating model reduction in frequency domain

Approximate the dynamical system

$$\begin{array}{rcll} E\dot{x} &=& Ax + Bu, & E, A \in \mathbb{R}^{n\times n},\ B \in \mathbb{R}^{n\times m}, \\ y &=& Cx + Du, & C \in \mathbb{R}^{q\times n},\ D \in \mathbb{R}^{q\times m}, \end{array}$$

by reduced-order system

$$\begin{array}{rcll} \hat{E}\dot{\hat{x}} &=& \hat{A}\hat{x} + \hat{B}u, & \hat{E}, \hat{A} \in \mathbb{R}^{r\times r},\ \hat{B} \in \mathbb{R}^{r\times m}, \\ \hat{y} &=& \hat{C}\hat{x} + \hat{D}u, & \hat{C} \in \mathbb{R}^{q\times r},\ \hat{D} \in \mathbb{R}^{q\times m} \end{array}$$

of order $r \ll n$, such that

$$\|y - \hat{y}\| = \|Gu - \hat{G}u\| \leq \|G - \hat{G}\| \cdot \|u\| < \text{tolerance} \cdot \|u\|.$$

$\Longrightarrow$ Approximation problem: $\displaystyle\min_{\text{order}\,(\hat{G})\leq r} \|G - \hat{G}\|.$

Some Background

Properties of linear systems

Definition

A linear system

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

is stable if its transfer function $G(s)$ has all its poles in the left half plane and it is asymptotically (or Lyapunov or exponentially) stable if all poles are in the open left half plane $\mathbb{C}^- := \{z \in \mathbb{C} \,|\, \Re(z) < 0\}$.

Lemma

Sufficient for asymptotic stability is that A is asymptotically stable (or Hurwitz), i.e., the spectrum of $A - \lambda E$, denoted by $\Lambda(A, E)$, satisfies $\Lambda(A, E) \subset \mathbb{C}^-$.

Note that by abuse of notation, often *stable system* is used for asymptotically stable systems.

Some Background
Properties of linear systems

Definition

A linear system

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

is stable if its transfer function $G(s)$ has all its poles in the left half plane and it is asymptotically (or Lyapunov or exponentially) stable if all poles are in the open left half plane $\mathbb{C}^- := \{z \in \mathbb{C} \,|\, \Re(z) < 0\}$.

Lemma

Sufficient for asymptotic stability is that A is asymptotically stable (or Hurwitz), i.e., the spectrum of $A - \lambda E$, denoted by $\Lambda(A, E)$, satisfies $\Lambda(A, E) \subset \mathbb{C}^-$.

Note that by abuse of notation, often *stable system* is used for asymptotically stable systems.

Some Background

Realizations of Linear Systems (with $E = I_n$ for simplicity)

Definition

For a linear (time-invariant) system

$$\Sigma : \left\{ \begin{array}{rcl} \dot{x}(t) & = & Ax(t) + Bu(t), \\ y(t) & = & Cx(t) + Du(t), \end{array} \right. \qquad \begin{array}{l} \text{with transfer function} \\ G(s) = C(sI - A)^{-1}B + D, \end{array}$$

the quadruple $(A, B, C, D) \in \mathbb{R}^{n\times n} \times \mathbb{R}^{n\times m} \times \mathbb{R}^{q\times n} \times \mathbb{R}^{q\times m}$ is called a realization of Σ.

Some Background

Realizations of Linear Systems (with $E = I_n$ for simplicity)

Definition

For a linear (time-invariant) system

$$\Sigma : \begin{cases} \dot{x}(t) &= Ax(t) + Bu(t), \\ y(t) &= Cx(t) + Du(t), \end{cases} \qquad \begin{array}{l} \text{with transfer function} \\ G(s) = C(sI - A)^{-1}B + D, \end{array}$$

the quadruple $(A, B, C, D) \in \mathbb{R}^{n\times n} \times \mathbb{R}^{n\times m} \times \mathbb{R}^{q\times n} \times \mathbb{R}^{q\times m}$ is called a realization of Σ.

Definition

The McMillan degree of Σ is the unique minimal number $\hat{n} \geq 0$ of states necessary to describe the input-output behavior completely.
A minimal realization is a realization $(\hat{A}, \hat{B}, \hat{C}, \hat{D})$ of Σ with order $\hat{n}$.

Some Background
Balanced Realizations

Definition

A realization (A, B, C, D) of a linear system Σ is balanced if its infinite controllability/observability Gramians P/Q satisfy

$$P = Q = \mathrm{diag}\,\{\sigma_1, \ldots, \sigma_n\} \quad \text{(w.l.o.g. } \sigma_j \geq \sigma_{j+1},\ j = 1, \ldots, n-1\text{)}.$$

Some Background
Balanced Realizations

Definition

A realization (A, B, C, D) of a linear system Σ is balanced if its infinite controllability/observability Gramians P/Q satisfy

$$P = Q = \mathrm{diag}\,\{\sigma_1, \ldots, \sigma_n\} \quad \text{(w.l.o.g. } \sigma_j \geq \sigma_{j+1},\ j = 1, \ldots, n-1\text{)}.$$

When does a balanced realization exist?

Some Background
Balanced Realizations

Definition

A realization (A, B, C, D) of a linear system Σ is balanced if its infinite controllability/observability Gramians P/Q satisfy

$$P = Q = \operatorname{diag}\{\sigma_1, \ldots, \sigma_n\} \quad (\text{w.l.o.g. } \sigma_j \geq \sigma_{j+1},\ j = 1, \ldots, n-1).$$

When does a balanced realization exist?
Assume A to be Hurwitz, i.e. $\Lambda(A) \subset \mathbb{C}^-$. Then:

Theorem

Given a stable minimal linear system $\Sigma : (A, B, C, D)$, a balanced realization is obtained by the state-space transformation with

$$T_b := \Sigma^{-\frac{1}{2}} V^T R,$$

where $P = S^T S$, $Q = R^T R$ (e.g., Cholesky decompositions) and $SR^T = U \Sigma V^T$ is the SVD of SR^T.

Some Background
Balanced Realizations

Definition

A realization (A, B, C, D) of a stable linear system Σ is balanced if its infinite controllability/observability Gramians P/Q satisfy

$$P = Q = \operatorname{diag}\{\sigma_1, \ldots, \sigma_n\} \quad \text{(w.l.o.g. } \sigma_j \geq \sigma_{j+1},\ j = 1, \ldots, n-1).$$

$\sigma_1, \ldots, \sigma_n$ are the Hankel singular values of Σ.

Note: $\sigma_1, \ldots, \sigma_n \geq 0$ as $P, Q \geq 0$ by definition, and $\sigma_1, \ldots, \sigma_n > 0$ in case of minimality!

Some Background
Balanced Realizations

Definition

A realization (A, B, C, D) of a stable linear system Σ is balanced if its infinite controllability/observability Gramians P/Q satisfy

$$P = Q = \operatorname{diag}\{\sigma_1, \ldots, \sigma_n\} \quad (\text{w.l.o.g. } \sigma_j \geq \sigma_{j+1},\ j = 1, \ldots, n-1).$$

$\sigma_1, \ldots, \sigma_n$ are the Hankel singular values of Σ.

Note: $\sigma_1, \ldots, \sigma_n \geq 0$ as $P, Q \geq 0$ by definition, and $\sigma_1, \ldots, \sigma_n > 0$ in case of minimality!

Theorem

The infinite controllability/observability Gramians P/Q satisfy the Lyapunov equations

$$AP + PA^T + BB^T = 0, \quad A^T Q + QA + C^T C = 0.$$

Some Background
Balanced Realizations

Definition

A realization (A, B, C, D) of a stable linear system Σ is balanced if its infinite controllability/observability Gramians P/Q satisfy

$$P = Q = \operatorname{diag}\{\sigma_1, \ldots, \sigma_n\} \quad (\text{w.l.o.g. } \sigma_j \geq \sigma_{j+1},\ j = 1, \ldots, n-1).$$

$\sigma_1, \ldots, \sigma_n$ are the Hankel singular values of Σ.

Note: $\sigma_1, \ldots, \sigma_n \geq 0$ as $P, Q \geq 0$ by definition, and $\sigma_1, \ldots, \sigma_n > 0$ in case of minimality!

Theorem

The Hankel singular values (HSVs) of a stable minimal linear system are system invariants, i.e. they are unaltered by state-space transformations!

Some Background
Balanced Realizations

Definition

A realization (A, B, C, D) of a stable linear system Σ is balanced if its infinite controllability/observability Gramians P/Q satisfy

$$P = Q = \mathrm{diag}\,\{\sigma_1, \ldots, \sigma_n\} \quad (\text{w.l.o.g. } \sigma_j \geq \sigma_{j+1},\ j = 1, \ldots, n-1).$$

$\sigma_1, \ldots, \sigma_n$ are the Hankel singular values of Σ.

Note: $\sigma_1, \ldots, \sigma_n \geq 0$ as $P, Q \geq 0$ by definition, and $\sigma_1, \ldots, \sigma_n > 0$ in case of minimality!

Remark

For non-minimal systems, the Gramians can also be transformed into diagonal matrices with the leading $\hat{n} \times \hat{n}$ submatrices equal to $\mathrm{diag}(\sigma_1, \ldots, \sigma_{\hat{n}})$, and

$$\hat{P}\hat{Q} = \mathrm{diag}(\sigma_1^2, \ldots, \sigma_{\hat{n}}^2, 0, \ldots, 0).$$

see [Laub/Heath/Paige/Ward 1987, Tombs/Postlethwaite 1987].

Qualitative and Quantitative Study of the Approximation Error
System Norms

Consider transfer function

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

and input functions $u \in \mathcal{L}_2^m \cong L_2^m(-\infty, \infty)$, with the L_2-norm

$$\|u\|_2^2 := \frac{1}{2\pi} \int_{-\infty}^{\infty} u(\jmath\omega)^H u(\jmath\omega)\, d\omega.$$

Qualitative and Quantitative Study of the Approximation Error

System Norms

Consider transfer function

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

and input functions $u \in \mathcal{L}_2^m \cong L_2^m(-\infty, \infty)$, with the L_2-norm

$$\|u\|_2^2 := \frac{1}{2\pi} \int_{-\infty}^{\infty} u(\jmath\omega)^H u(\jmath\omega)\, d\omega.$$

Hardy space $\mathcal{H}_\infty$

Function space of matrix-/scalar-valued functions that are analytic and bounded in $\mathbb{C}^+$.

The $\mathcal{H}_\infty$-norm is

$$\|F\|_\infty := \sup_{\operatorname{re} s > 0} \sigma_{\max}\left(F(s)\right) = \sup_{\omega \in \mathbb{R}} \sigma_{max}\left(F(\jmath\omega)\right).$$

Stable transfer functions are in the Hardy spaces

- $\mathcal{H}_\infty$ in the SISO case (single-input, single-output, $m = q = 1$);
- $\mathcal{H}_\infty^{q\times m}$ in the MIMO case (multi-input, multi-output, $m > 1, q > 1$).

Qualitative and Quantitative Study of the Approximation Error
System Norms

Consider transfer function

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

and input functions $u \in \mathcal{L}_2^m \cong L_2^m(-\infty, \infty)$, with the L_2-norm

$$\|u\|_2^2 := \frac{1}{2\pi}\int_{-\infty}^{\infty} u(\jmath\omega)^H u(\jmath\omega)\,d\omega.$$

Paley-Wiener Theorem (Parseval's equation/Plancherel Theorem)

$$L_2(-\infty,\infty) \cong \mathcal{L}_2, \quad L_2(0,\infty) \cong \mathcal{H}_2$$

Consequently, 2-norms in time and frequency domains coincide!

Qualitative and Quantitative Study of the Approximation Error

System Norms

Consider transfer function

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

Paley-Wiener Theorem (Parseval's equation/Plancherel Theorem)

$$L_2(-\infty,\infty) \cong \mathcal{L}_2, \quad L_2(0,\infty) \cong \mathcal{H}_2$$

Consequently, 2-norms in time and frequency domains coincide!

$\mathcal{H}_\infty$ approximation error

Reduced-order model $\Rightarrow$ transfer function $\hat{G}(s) = \hat{C}(sI_r - \hat{A})^{-1}\hat{B} + \hat{D}$.

$$\|y - \hat{y}\|_2 = \|Gu - \hat{G}u\|_2 \le \|G - \hat{G}\|_\infty \|u\|_2.$$

$\Longrightarrow$ compute reduced-order model such that $\|G - \hat{G}\|_\infty < tol$!

Note: error bound holds in time- and frequency domain due to Paley-Wiener!

Qualitative and Quantitative Study of the Approximation Error

System Norms

Consider stable transfer function

$$G(s) = C\,(sI - A)^{-1}\,B, \quad \text{i.e. } D = 0.$$

Hardy space $\mathcal{H}_2$

Function space of matrix-/scalar-valued functions that are analytic $\mathbb{C}^+$ and bounded w.r.t. the $\mathcal{H}_2$-norm

$$\begin{aligned} \|F\|_2 &:= \frac{1}{2\pi}\left(\sup_{\operatorname{re}\sigma>0}\int_{-\infty}^{\infty}\|F(\sigma+\jmath\omega)\|_F^2\,d\omega\right)^{\frac{1}{2}} \\ &= \frac{1}{2\pi}\left(\int_{-\infty}^{\infty}\|F(\jmath\omega)\|_F^2\,d\omega\right)^{\frac{1}{2}}. \end{aligned}$$

Stable transfer functions are in the Hardy spaces

- $\mathcal{H}_2$ in the SISO case (single-input, single-output, $m = q = 1$);
- $\mathcal{H}_2^{q\times m}$ in the MIMO case (multi-input, multi-output, $m > 1, q > 1$).

Qualitative and Quantitative Study of the Approximation Error

System Norms

Consider stable transfer function

$$G(s) = C\,(sI - A)^{-1}\,B, \quad \text{i.e. } D = 0.$$

Hardy space $\mathcal{H}_2$

Function space of matrix-/scalar-valued functions that are analytic $\mathbb{C}^+$ and bounded w.r.t. the $\mathcal{H}_2$-norm

$$\|F\|_2 \quad = \quad \frac{1}{2\pi}\left(\int_{-\infty}^{\infty} \|F(\jmath\omega)\|_F^2\, d\omega\right)^{\frac{1}{2}}.$$

$\mathcal{H}_2$ approximation error for impulse response $(u(t) = u_0\delta(t))$

Reduced-order model $\Rightarrow$ transfer function $\hat{G}(s) = \hat{C}(sI_r - \hat{A})^{-1}\hat{B}$.

$$\|y - \hat{y}\|_2 = \|Gu_0\delta - \hat{G}u_0\delta\|_2 \leq \|G - \hat{G}\|_2\|u_0\|.$$

$\Longrightarrow$ compute reduced-order model such that $\|G - \hat{G}\|_2 < tol$!

Qualitative and Quantitative Study of the Approximation Error

System Norms

Consider stable transfer function

$$G(s) = C\,(sI - A)^{-1}\,B, \quad \text{i.e. } D = 0.$$

Hardy space $\mathcal{H}_2$

Function space of matrix-/scalar-valued functions that are analytic $\mathbb{C}^+$ and bounded w.r.t. the $\mathcal{H}_2$-norm

$$\|F\|_2 \quad = \quad \frac{1}{2\pi}\left(\int_{-\infty}^{\infty} \|F(\jmath\omega)\|_F^2\, d\omega\right)^{\frac{1}{2}}.$$

Theorem (Practical Computation of the $\mathcal{H}_2$-norm)

$$\|F\|_2^2 = \mathrm{tr}\left(B^T Q B\right) = \mathrm{tr}\left(C P C^T\right),$$

where P, Q are the controllability and observability Gramians of the corresponding LTI system.

Qualitative and Quantitative Study of the Approximation Error
Approximation Problems

Output errors in time-domain

$$
\begin{aligned}
\|y-\hat{y}\|_2 &\leq \|G-\hat{G}\|_\infty \|u\|_2 &\Longrightarrow \|G-\hat{G}\|_\infty < \mathrm{tol} \\
\|y-\hat{y}\|_\infty &\leq \|G-\hat{G}\|_2 \|u\|_2 &\Longrightarrow \|G-\hat{G}\|_2 < \mathrm{tol}
\end{aligned}
$$

Qualitative and Quantitative Study of the Approximation Error

Approximation Problems

Output errors in time-domain

$$\begin{aligned} \|y-\hat{y}\|_2 &\leq \|G-\hat{G}\|_\infty \|u\|_2 &\Longrightarrow \|G-\hat{G}\|_\infty < \mathrm{tol} \\ \|y-\hat{y}\|_\infty &\leq \|G-\hat{G}\|_2 \|u\|_2 &\Longrightarrow \|G-\hat{G}\|_2 < \mathrm{tol} \end{aligned}$$

$\mathcal{H}_\infty$-norm	best approximation problem for given reduced order r in general open; balanced truncation yields suboptimal solution with computable $\mathcal{H}_\infty$-norm bound.
$\mathcal{H}_2$-norm	necessary conditions for best approximation known; (local) optimizer computable with iterative rational Krylov algorithm (IRKA)
Hankel-norm $\|G\|_H := \sigma_{\max}$	optimal Hankel norm approximation (AAK theory).

Introduction
Goals

- Automatic generation of compact models.
- Satisfy desired error tolerance for all admissible input signals, i.e., want

$$\|y - \hat{y}\| < \text{tolerance} \cdot \|u\| \qquad \forall u \in L_2(\mathbb{R}, \mathbb{R}^m).$$

$\Longrightarrow$ Need computable error bound/estimate!

- Preserve physical properties:
  - stability (poles of G in $\mathbb{C}^-$),
  - minimum phase (zeroes of G in $\mathbb{C}^-$),
  - passivity

$$\int_{-\infty}^{t} u(\tau)^T y(\tau)\, d\tau \geq 0 \quad \forall t \in \mathbb{R}, \quad \forall u \in L_2(\mathbb{R}, \mathbb{R}^m).$$

(*"system does not generate energy"*).

Introduction

Goals

- Automatic generation of compact models.
- Satisfy desired error tolerance for all admissible input signals, i.e., want

$$\|y - \hat{y}\| < \text{tolerance} \cdot \|u\| \qquad \forall u \in L_2(\mathbb{R}, \mathbb{R}^m).$$

$\Longrightarrow$ Need computable error bound/estimate!

- Preserve physical properties:
  - stability (poles of G in $\mathbb{C}^-$),
  - minimum phase (zeroes of G in $\mathbb{C}^-$),
  - passivity

$$\int_{-\infty}^{t} u(\tau)^T y(\tau)\, d\tau \geq 0 \quad \forall t \in \mathbb{R}, \quad \forall u \in L_2(\mathbb{R}, \mathbb{R}^m),$$

("system does not generate energy").

Introduction

Goals

- Automatic generation of compact models.
- Satisfy desired error tolerance for all admissible input signals, i.e., want

$$\|y - \hat{y}\| < \text{tolerance} \cdot \|u\| \qquad \forall u \in L_2(\mathbb{R}, \mathbb{R}^m).$$

$\Longrightarrow$ Need computable error bound/estimate!

- Preserve physical properties:
  - stability (poles of G in $\mathbb{C}^-$),
  - minimum phase (zeroes of G in $\mathbb{C}^-$),
  - passivity

$$\int_{-\infty}^{t} u(\tau)^T y(\tau)\, d\tau \geq 0 \quad \forall t \in \mathbb{R}, \quad \forall u \in L_2(\mathbb{R}, \mathbb{R}^m).$$

("system does not generate energy").

Introduction

Goals

- Automatic generation of compact models.
- Satisfy desired error tolerance for all admissible input signals, i.e., want

$$\|y - \hat{y}\| < \text{tolerance} \cdot \|u\| \qquad \forall u \in L_2(\mathbb{R}, \mathbb{R}^m).$$

$\Longrightarrow$ Need computable error bound/estimate!

- Preserve physical properties:
  - stability (poles of G in $\mathbb{C}^-$),
  - minimum phase (zeroes of G in $\mathbb{C}^-$),
  - passivity

$$\int_{-\infty}^{t} u(\tau)^T y(\tau)\, d\tau \geq 0 \quad \forall t \in \mathbb{R}, \quad \forall u \in L_2(\mathbb{R}, \mathbb{R}^m).$$

("system does not generate energy").

Introduction

Goals

- Automatic generation of compact models.
- Satisfy desired error tolerance for all admissible input signals, i.e., want

$$\|y - \hat{y}\| < \text{tolerance} \cdot \|u\| \qquad \forall u \in L_2(\mathbb{R}, \mathbb{R}^m).$$

$\Longrightarrow$ Need computable error bound/estimate!

- Preserve physical properties:
  - stability (poles of G in $\mathbb{C}^-$),
  - minimum phase (zeroes of G in $\mathbb{C}^-$),
  - passivity

$$\int_{-\infty}^{t} u(\tau)^T y(\tau)\, d\tau \geq 0 \quad \forall t \in \mathbb{R}, \quad \forall u \in L_2(\mathbb{R}, \mathbb{R}^m).$$

("system does not generate energy").

Introduction

Goals

- Automatic generation of compact models.
- Satisfy desired error tolerance for all admissible input signals, i.e., want

$$\|y - \hat{y}\| < \text{tolerance} \cdot \|u\| \qquad \forall u \in L_2(\mathbb{R}, \mathbb{R}^m).$$

$\Longrightarrow$ Need computable error bound/estimate!

- Preserve physical properties:
  - stability (poles of G in $\mathbb{C}^-$),
  - minimum phase (zeroes of G in $\mathbb{C}^-$),
  - passivity

$$\int_{-\infty}^{t} u(\tau)^T y(\tau)\, d\tau \geq 0 \quad \forall t \in \mathbb{R}, \quad \forall u \in L_2(\mathbb{R}, \mathbb{R}^m).$$

("system does not generate energy").

Outline

1. Introduction
2. Model Reduction by Projection
   - Projection Methods
   - Projection and Rational Interpolation
3. Interpolatory Model Reduction
4. Balanced Truncation
5. Nonlinear Model Reduction
6. Final Remarks

Model Reduction by Projection

Projection Basics

Definition 3.1 (Projector)

A projector is a matrix $P \in \mathbb{R}^{n \times n}$ with $P^2 = P$. Let $\mathcal{V} = \mathrm{range}\,(P)$, then P is projector onto $\mathcal{V}$.
If $P = P^T$, then P is an orthogonal projector (aka: Galerkin projection), otherwise an oblique projector (aka: Petrov-Galerkin projection).

Model Reduction by Projection

Projection Basics

Definition 3.1 (Projector)

A projector is a matrix $P \in \mathbb{R}^{n \times n}$ with $P^2 = P$. Let $\mathcal{V} = \mathrm{range}\,(P)$, then P is projector onto $\mathcal{V}$.
If $P = P^T$, then P is an orthogonal projector (aka: Galerkin projection), otherwise an oblique projector (aka: Petrov-Galerkin projection).

Lemma 3.2 (Projector Properties)

- If $\{v_1, \ldots, v_r\}$ is a basis of $\mathcal{V}$ and $V = [\, v_1, \ldots, v_r \,]$, then $P = V(V^T V)^{-1} V^T$ is an orthogonal projector onto $\mathcal{V}$.
- Let $\mathcal{W} \subset \mathbb{R}^n$ be another r-dimensional subspace and $W = [\, w_1, \ldots, w_r \,]$ be a basis matrix for $\mathcal{W}$, then $P = V(W^T V)^{-1} W^T$ is an oblique projector onto $\mathcal{V}$ along $\mathcal{W}$.

Model Reduction by Projection

Projection Basics

Definition 3.1 (Projector)

A projector is a matrix $P \in \mathbb{R}^{n \times n}$ with $P^2 = P$. Let $\mathcal{V} = \mathrm{range}(P)$, then P is projector onto $\mathcal{V}$.
If $P = P^T$, then P is an orthogonal projector (aka: Galerkin projection), otherwise an oblique projector (aka: Petrov-Galerkin projection).

Lemma 3.2 (Projector Properties)

- If $\{v_1, \ldots, v_r\}$ is a basis of $\mathcal{V}$ and $V = [\, v_1, \ldots, v_r \,]$, then $P = V(V^T V)^{-1} V^T$ is an orthogonal projector onto $\mathcal{V}$.
- Let $\mathcal{W} \subset \mathbb{R}^n$ be another r-dimensional subspace and $W = [\, w_1, \ldots, w_r \,]$ be a basis matrix for $\mathcal{W}$, then $P = V(W^T V)^{-1} W^T$ is an oblique projector onto $\mathcal{V}$ along $\mathcal{W}$.

Model Reduction by Projection

Projection Methods

Methods:

1. Modal Truncation
2. Rational Interpolation (Padé-Approximation and (rational) Krylov Subspace Methods)
3. Balanced Truncation
4. many more. . .

Joint feature of these methods:
computation of reduced-order model (ROM) by projection!

Model Reduction by Projection

Projection Methods

Joint feature of these methods:
computation of reduced-order model (ROM) by projection!

Assume trajectory $x(t;u)$ is contained in low-dimensional subspace $\mathcal{V}$. Thus, use Galerkin or Petrov-Galerkin-type projection of state-space onto $\mathcal{V}$ along complementary subspace $\mathcal{W}$: $x \approx VW^Tx =: \tilde{x}$, where

$$\mathrm{range}\,(V) = \mathcal{V}, \quad \mathrm{range}\,(W) = \mathcal{W}, \quad W^TV = I_r.$$

Then, with $\hat{x} = W^Tx$, we obtain $x \approx V\hat{x}$ so that

$$\|x - \tilde{x}\| = \|x - V\hat{x}\|,$$

and the reduced-order model is

$$\hat{A} := W^TAV, \quad \hat{B} := W^TB, \quad \hat{C} := CV, \quad (\hat{D} := D).$$

Model Reduction by Projection

Projection Methods

Joint feature of these methods:
computation of reduced-order model (ROM) by projection!
Assume trajectory $x(t; u)$ is contained in low-dimensional subspace $\mathcal{V}$. Thus, use Galerkin or Petrov-Galerkin-type projection of state-space onto $\mathcal{V}$ along complementary subspace $\mathcal{W}$: $x \approx VW^T x =: \tilde{x}$, and the reduced-order model is $\hat{x} = W^T x$

$$\hat{A} := W^T AV, \quad \hat{B} := W^T B, \quad \hat{C} := CV, \quad (\hat{D} := D).$$

Important observation:

- The state equation residual satisfies $\dot{\tilde{x}} - A\tilde{x} - Bu \perp \mathcal{W}$, since

$$W^T \left(\dot{\tilde{x}} - A\tilde{x} - Bu\right) \quad = \quad W^T \left(VW^T \dot{x} - AVW^T x - Bu\right)$$

Model Reduction by Projection

Projection Methods

Joint feature of these methods:
computation of reduced-order model (ROM) by projection!

Assume trajectory $x(t;u)$ is contained in low-dimensional subspace $\mathcal{V}$. Thus, use Galerkin or Petrov-Galerkin-type projection of state-space onto $\mathcal{V}$ along complementary subspace $\mathcal{W}$: $x \approx VW^Tx =: \tilde{x}$, and the reduced-order model is $\hat{x} = W^Tx$

$$\hat{A} := W^TAV, \quad \hat{B} := W^TB, \quad \hat{C} := CV, \quad (\hat{D} := D).$$

Important observation:

- The state equation residual satisfies $\dot{\tilde{x}} - A\tilde{x} - Bu \perp \mathcal{W}$, since

$$\begin{aligned} W^T\left(\dot{\tilde{x}} - A\tilde{x} - Bu\right) &= W^T\left(VW^T\dot{x} - AVW^Tx - Bu\right) \\ &= \underbrace{W^T\dot{x}}_{\dot{\hat{x}}} - \underbrace{W^TAV}_{=\hat{A}}\underbrace{W^Tx}_{=\hat{x}} - \underbrace{W^TB}_{=\hat{B}}u \end{aligned}$$

Model Reduction by Projection

Projection Methods

Joint feature of these methods:
computation of reduced-order model (ROM) by projection!

Assume trajectory $x(t; u)$ is contained in low-dimensional subspace $\mathcal{V}$. Thus, use Galerkin or Petrov-Galerkin-type projection of state-space onto $\mathcal{V}$ along complementary subspace $\mathcal{W}$: $x \approx VW^T x =: \tilde{x}$, and the reduced-order model is $\hat{x} = W^T x$

$$\hat{A} := W^T AV, \quad \hat{B} := W^T B, \quad \hat{C} := CV, \quad (\hat{D} := D).$$

Important observation:

- The state equation residual satisfies $\dot{\tilde{x}} - A\tilde{x} - Bu \perp \mathcal{W}$, since

$$\begin{aligned} W^T\left(\dot{\tilde{x}} - A\tilde{x} - Bu\right) &= W^T\left(VW^T\dot{x} - AVW^Tx - Bu\right) \\ &= \underbrace{W^T\dot{x}}_{\dot{\hat{x}}} - \underbrace{W^TAV}_{=\hat{A}}\underbrace{W^Tx}_{=\hat{x}} - \underbrace{W^TB}_{=\hat{B}}u \\ &= \dot{\hat{x}} - \hat{A}\hat{x} - \hat{B}u = 0. \end{aligned}$$

Model Reduction by Projection

Projection and Rational Interpolation

Projection $\rightsquigarrow$ Rational Interpolation

Given the ROM

$$\hat{A} = W^T AV, \quad \hat{B} = W^T B, \quad \hat{C} = CV, \quad (\hat{D} = D),$$

the error transfer function can be written as

$$G(s) - \hat{G}(s) \quad = \quad \left(C(sI_n - A)^{-1}B + D\right) - \left(\hat{C}(sI_r - \hat{A})^{-1}\hat{B} + \hat{D}\right)$$

Model Reduction by Projection

Projection and Rational Interpolation

Projection $\rightsquigarrow$ Rational Interpolation

Given the ROM

$$\hat{A} = W^T AV, \quad \hat{B} = W^T B, \quad \hat{C} = CV, \quad (\hat{D} = D),$$

the error transfer function can be written as

$$\begin{aligned} G(s) - \hat{G}(s) &= \left(C(sI_n - A)^{-1}B + D\right) - \left(\hat{C}(sI_r - \hat{A})^{-1}\hat{B} + \hat{D}\right) \\ &= C\left((sI_n - A)^{-1} - V(sI_r - \hat{A})^{-1}W^T\right)B \end{aligned}$$

Model Reduction by Projection

Projection and Rational Interpolation

Projection $\rightsquigarrow$ Rational Interpolation

Given the ROM

$$\hat{A} = W^T AV, \quad \hat{B} = W^T B, \quad \hat{C} = CV, \quad (\hat{D} = D),$$

the error transfer function can be written as

$$\begin{aligned} G(s) - \hat{G}(s) &= \left(C(sI_n - A)^{-1}B + D\right) - \left(\hat{C}(sI_r - \hat{A})^{-1}\hat{B} + \hat{D}\right) \\ &= C\left((sI_n - A)^{-1} - V(sI_r - \hat{A})^{-1}W^T\right)B \\ &= C\big(I_n - \underbrace{V(sI_r - \hat{A})^{-1}W^T(sI_n - A)}_{=:P(s)}\big)(sI_n - A)^{-1}B. \end{aligned}$$

Model Reduction by Projection

Projection and Rational Interpolation

Projection $\rightsquigarrow$ Rational Interpolation

Given the ROM

$$\hat{A} = W^T AV, \quad \hat{B} = W^T B, \quad \hat{C} = CV, \quad (\hat{D} = D),$$

the error transfer function can be written as

$$\begin{aligned} G(s) - \hat{G}(s) &= \left(C(sI_n - A)^{-1}B + D\right) - \left(\hat{C}(sI_r - \hat{A})^{-1}\hat{B} + \hat{D}\right) \\ &= C\left((sI_n - A)^{-1} - V(sI_r - \hat{A})^{-1}W^T\right)B \\ &= C\big(I_n - \underbrace{V(sI_r - \hat{A})^{-1}W^T(sI_n - A)}_{=:P(s)}\big)(sI_n - A)^{-1}B. \end{aligned}$$

If $s_* \in \mathbb{C} \setminus (\Lambda(A) \cup \Lambda(\hat{A}))$, then $P(s_*)$ is a projector onto $\mathcal{V}$ $\Longrightarrow$

if $(s_* I_n - A)^{-1}B \in \mathcal{V}$, *then* $(I_n - P(s_*))(s_* I_n - A)^{-1}B = 0$,

Hence

$$G(s_*) - \hat{G}(s_*) = 0 \;\Rightarrow\; G(s_*) = \hat{G}(s_*), \text{ i.e., } \hat{G} \text{ interpolates } G \text{ in } s_*!$$

Model Reduction by Projection

Projection and Rational Interpolation

Projection $\rightsquigarrow$ Rational Interpolation

Given the ROM

$$\hat{A} = W^T AV, \quad \hat{B} = W^T B, \quad \hat{C} = CV, \quad (\hat{D} = D),$$

the error transfer function can be written as

$$\begin{aligned} G(s) - \hat{G}(s) &= \left(C(sI_n - A)^{-1}B + D\right) - \left(\hat{C}(sI_r - \hat{A})^{-1}\hat{B} + \hat{D}\right) \\ \text{Analogously,} \quad &= C(sI_n - A)^{-1}\left(I_n - \underbrace{(sI_n - A)V(sI_r - \hat{A})^{-1}W^T}_{=:Q(s)}\right)B. \end{aligned}$$

If $s_* \in \mathbb{C} \setminus (\Lambda(A) \cup \Lambda(\hat{A}))$, then $Q(s)^H$ is a projector onto $\mathcal{W} \Longrightarrow$

if $(s_* I_n - A)^{-*}C^T \in \mathcal{W}$, *then* $C(s_* I_n - A)^{-1}(I_n - Q(s_*)) = 0.$

Hence

$$G(s_*) - \hat{G}(s_*) = 0 \;\Rightarrow\; G(s_*) = \hat{G}(s_*), \text{ i.e., } \hat{G} \text{ interpolates } G \text{ in } s_*!$$

Model Reduction by Projection

Projection and Rational Interpolation

Theorem [Grimme '97, Villemagne/Skelton '87]

Given the ROM

$$\hat{A} = W^T AV, \quad \hat{B} = W^T B, \quad \hat{C} = CV, \quad (\hat{D} = D),$$

and $s_* \in \mathbb{C} \setminus (\Lambda(A) \cup \Lambda(\hat{A}))$, if either

- $(s_* I_n - A)^{-1} B \in \mathrm{range}(V)$, or
- $(s_* I_n - A)^{-*} C^T \in \mathrm{range}(W)$,

then the interpolation condition

$$G(s_*) = \hat{G}(s_*).$$

in s_* holds.

Note: extension to Hermite interpolation conditions later!

Outline

1. Introduction
2. Model Reduction by Projection
3. Interpolatory Model Reduction
   - Padé Approximation
   - A Change of Perspective: Rational Interpolation
   - $\mathcal{H}_2$-Optimal Model Reduction
4. Balanced Truncation
5. Nonlinear Model Reduction
6. Final Remarks

Padé Approximation

Idea:

- Consider (even for possibly singular E if $\lambda E - A$ regular):

$$E\dot{x} = Ax + Bu, \quad y = Cx$$

with transfer function $G(s) = C(sE - A)^{-1}B$.

Padé Approximation

Idea:

- Consider (even for possibly singular E if $\lambda E - A$ regular):

$$E\dot{x} = Ax + Bu, \quad y = Cx$$

with transfer function $G(s) = C(sE - A)^{-1}B$.

- For $s_0 \notin \Lambda(A, E)$:

$$G(s) \quad = \quad C\big((s_0E - A) + (s - s_0)E\big)^{-1}B$$

Padé Approximation

Idea:

- Consider (even for possibly singular E if $\lambda E - A$ regular):

$$E\dot{x} = Ax + Bu, \quad y = Cx$$

with transfer function $G(s) = C(sE - A)^{-1}B$.

- For $s_0 \notin \Lambda(A, E)$:

$$\begin{aligned} G(s) &= C\left((s_0E - A) + (s - s_0)E\right)^{-1}B \\ &= C\Big(I + (s - s_0)\underbrace{(s_0E - A)^{-1}E}_{:=\tilde{A}}\Big)^{-1}\underbrace{(s_0E - A)^{-1}B}_{:=\tilde{B}} \end{aligned}$$

Padé Approximation

Idea:

- Consider (even for possibly singular E if $\lambda E - A$ regular):

$$E\dot{x} = Ax + Bu, \quad y = Cx$$

with transfer function $G(s) = C(sE - A)^{-1}B$.

- For $s_0 \notin \Lambda(A, E)$:

$$\begin{aligned} G(s) &= C\big((s_0E - A) + (s - s_0)E\big)^{-1}B \\ &= C\Big(I + (s - s_0)\underbrace{(s_0E - A)^{-1}E}_{:=\tilde{A}}\Big)^{-1}\underbrace{(s_0E - A)^{-1}B}_{:=\tilde{B}} \\ &= C\left(I + (s - s_0)\tilde{A}\right)^{-1}\tilde{B} \end{aligned}$$

Padé Approximation

Idea:

- Consider (even for possibly singular E if $\lambda E - A$ regular):

$$E\dot{x} = Ax + Bu, \quad y = Cx$$

with transfer function $G(s) = C(sE - A)^{-1}B$.

- For $s_0 \notin \Lambda(A, E)$:

$$\begin{aligned} G(s) &= C\big((s_0E - A) + (s - s_0)E\big)^{-1}B \\ &= C\Big(I + (s - s_0)\underbrace{(s_0E - A)^{-1}E}_{:=\tilde{A}}\Big)^{-1}\underbrace{(s_0E - A)^{-1}B}_{:=\tilde{B}} \\ &= C\left(I + (s - s_0)\tilde{A}\right)^{-1}\tilde{B} \end{aligned}$$

Neumann Lemma. $\|F\| < 1 \;\Rightarrow\; I - F$ invertible, $(I - F)^{-1} = \sum_{k=0}^{\infty} F^k$.

Padé Approximation

Idea:

- Consider (even for possibly singular E if $\lambda E - A$ regular):

$$E\dot{x} = Ax + Bu, \quad y = Cx$$

with transfer function $G(s) = C(sE - A)^{-1}B$.

- For $s_0 \notin \Lambda(A, E)$:

$$\begin{aligned}
G(s) &= C\left((s_0 E - A) + (s - s_0)E\right)^{-1} B \\
&= C\Big(I + (s - s_0)\underbrace{(s_0 E - A)^{-1} E}_{:=\tilde{A}}\Big)^{-1} \underbrace{(s_0 E - A)^{-1} B}_{:=\tilde{B}} \\
&= C\left(I + (s - s_0)\tilde{A}\right)^{-1} \tilde{B} = C\Big(I - \underbrace{\left(-(s - s_0)\tilde{A}\right)}_{=F}\Big)^{-1} \tilde{B}
\end{aligned}$$

Neumann Lemma. $\|F\| < 1 \ \Rightarrow\ I - F$ invertible, $(I - F)^{-1} = \sum_{k=0}^{\infty} F^k$.

Padé Approximation

Idea:

- Consider (even for possibly singular E if $\lambda E - A$ regular):

$$E\dot{x} = Ax + Bu, \quad y = Cx$$

with transfer function $G(s) = C(sE - A)^{-1}B$.

- For $s_0 \notin \Lambda(A, E)$, and $\tilde{A} = (s_0 E - A)^{-1}E$, $\tilde{B} = (s_0 E - A)^{-1}B$:

$$\begin{aligned} G(s) &= C\left(I + (s - s_0)\tilde{A}\right)^{-1}\tilde{B} = C\Big(I - \underbrace{\big(-(s - s_0)\tilde{A}\big)}_{=F}\Big)^{-1}\tilde{B} \\ &= C\left(\sum_{k=0}^{\infty}(-1)^k(s - s_0)^k\tilde{A}^k\right)\tilde{B} \end{aligned}$$

Neumann Lemma. $\|F\| < 1 \;\Rightarrow\; I - F$ invertible, $(I - F)^{-1} = \sum_{k=0}^{\infty} F^k$.

Padé Approximation

Idea:

- Consider (even for possibly singular E if $\lambda E - A$ regular):

$$E\dot{x} = Ax + Bu, \quad y = Cx$$

  with transfer function $G(s) = C(sE - A)^{-1}B$.

- For $s_0 \notin \Lambda(A, E)$, and $\tilde{A} = (s_0 E - A)^{-1}E$, $\tilde{B} = (s_0 E - A)^{-1}B$:

$$\begin{aligned} G(s) &= C\left(I + (s - s_0)\tilde{A}\right)^{-1}\tilde{B} = C\Big(I - \underbrace{\big(-(s - s_0)\tilde{A}\big)}_{=F}\Big)^{-1}\tilde{B} \\ &= C\left(\sum_{k=0}^{\infty}(-1)^k (s - s_0)^k \tilde{A}^k\right)\tilde{B} \\ &= \sum_{k=0}^{\infty}\underbrace{(-1)^k C\tilde{A}^k\tilde{B}}_{=:\,m_k}(s - s_0)^k \end{aligned}$$

Padé Approximation

Idea:

- Consider (even for possibly singular E if $\lambda E - A$ regular):

$$E\dot{x} = Ax + Bu, \quad y = Cx$$

with transfer function $G(s) = C(sE - A)^{-1}B$.

- For $s_0 \notin \Lambda(A, E)$, and $\tilde{A} = (s_0 E - A)^{-1}E$, $\tilde{B} = (s_0 E - A)^{-1}B$:

$$\begin{aligned}
G(s) &= C\left(I + (s - s_0)\tilde{A}\right)^{-1}\tilde{B} = C\Big(I - \underbrace{\big(-(s - s_0)\tilde{A}\big)}_{=F}\Big)^{-1}\tilde{B} \\
&= C\left(\sum_{k=0}^{\infty}(-1)^k(s - s_0)^k\tilde{A}^k\right)\tilde{B} \\
&= \sum_{k=0}^{\infty}\underbrace{(-1)^k C\tilde{A}^k\tilde{B}}_{=:\, m_k}(s - s_0)^k \\
&= m_0 + m_1(s - s_0) + m_2(s - s_0)^2 + \ldots
\end{aligned}$$

Padé Approximation

Idea:

- Consider (even for possibly singular E if $\lambda E - A$ regular):

$$E\dot{x} = Ax + Bu, \quad y = Cx$$

with transfer function $G(s) = C(sE - A)^{-1}B$.

- For $s_0 \notin \Lambda(A, E)$, and $\tilde{A} = (s_0 E - A)^{-1}E$, $\tilde{B} = (s_0 E - A)^{-1}B$:

$$G(s) \quad = \quad m_0 + m_1(s - s_0) + m_2(s - s_0)^2 + \ldots$$

with $m_k = (-1)^k C\tilde{A}^k\tilde{B}$.

Padé Approximation

Idea:

- Consider (even for possibly singular E if $\lambda E - A$ regular):

$$E\dot{x} = Ax + Bu, \quad y = Cx$$

with transfer function $G(s) = C(sE - A)^{-1}B$.

- For $s_0 \notin \Lambda(A, E)$, and $\tilde{A} = (s_0 E - A)^{-1}E$, $\tilde{B} = (s_0 E - A)^{-1}B$:

$$G(s) = m_0 + m_1(s - s_0) + m_2(s - s_0)^2 + \ldots$$

with $m_k = (-1)^k C\tilde{A}^k\tilde{B}$.

  - For $s_0 = 0$: $m_k := -C(A^{-1}E)^k A^{-1}B \rightsquigarrow$ moments.
    $(m_k = -CA^{-(k+1)}B$ for $E = I_n)$
  - For $s_0 = \infty$ and $E = I_n$: $m_0 = 0$, $m_k := CA^{k-1}B$ for $k \geq 1 \rightsquigarrow$ Markov parameters.

Padé Approximation

Idea:

- Consider (even for possibly singular E if $\lambda E - A$ regular):

$$E\dot{x} = Ax + Bu, \quad y = Cx$$

with transfer function $G(s) = C(sE - A)^{-1}B$.

- For $s_0 \notin \Lambda(A, E)$, and $\tilde{A} = (s_0 E - A)^{-1}E$, $\tilde{B} = (s_0 E - A)^{-1}B$:

$$G(s) \quad = \quad m_0 + m_1(s - s_0) + m_2(s - s_0)^2 + \ldots$$

with $m_k = (-1)^k C\tilde{A}^k\tilde{B}$.

- As reduced-order model use rth Padé approximant $\hat{G}$ to G:

$$G(s) = \hat{G}(s) + \mathcal{O}((s - s_0)^{2r}),$$

i.e., $m_k = \widehat{m}_k$ for $k = 0, \ldots, 2r - 1$

$\rightsquigarrow$ moment matching if $s_0 < \infty$,

$\rightsquigarrow$ partial realization if $s_0 = \infty$.

Padé Approximation

The Padé-Lanczos Connection [Gallivan/Grimme/Van Dooren 1994, Freund/Feldmann 1994]

Theorem [Grimme '97, Villemagne/Skelton '87]

Let $s_* \notin \Lambda(A, E)$ and

$$\begin{aligned} \tilde{A} &:= (s_* E - A)^{-1} E, \quad & \tilde{B} := (s_* E - A)^{-1} B, \\ \tilde{A}^* &:= (s_* E - A)^{-T} E^T, \quad & \tilde{C} := (s_* E - A)^{-T} C^T. \end{aligned}$$

If the reduced-order model is obtained by oblique projection onto $\mathcal{V} \subset \mathbb{R}^n$ along $\mathcal{W} \subset \mathbb{R}^n$, and

$$\begin{aligned} \operatorname{span}\left\{\tilde{B}, \tilde{A}\tilde{B}, \ldots, \tilde{A}^{K-1}\tilde{B}\right\} &\subset \mathcal{V}, \\ \operatorname{span}\left\{\tilde{C}, \tilde{A}^*\tilde{C}, \ldots, (\tilde{A}^*)^{K-1}\tilde{C}\right\} &\subset \mathcal{W}, \end{aligned}$$

then $G(s_*) = \hat{G}(s_*)$, $\frac{d^k}{ds^k} G(s_*) = \frac{d^k}{ds^k} \hat{G}(s_*)$ for $k = 1, \ldots, \ell - 1$, where

$$\ell \geq \begin{cases} 2K & \text{if } m = q = 1; \\ \lfloor \frac{K}{m} \rfloor + \lfloor \frac{K}{q} \rfloor & \text{if } m \neq 1 \text{ or } q \neq 1. \end{cases}$$

Padé Approximation

The Padé-Lanczos Connection [Gallivan/Grimme/Van Dooren 1994, Freund/Feldmann 1994]

Padé-via-Lanczos Method (PVL)

- Padé approximation/moment matching yield:

$$m_k = \frac{1}{k!} G^{(k)}(s_0) = \frac{1}{k!} \hat{G}^{(k)}(s_0) = \hat{m}_k, \quad k = 0, \ldots, 2K-1,$$

  i.e., Hermite interpolation in s_0.

- Recall interpolation via projection result $\Rightarrow$ moments need not be computed explicitly; moment matching is equivalent to projecting state-space onto

$$\mathcal{V} = \mathrm{span}(\tilde{B}, \tilde{A}\tilde{B}, \ldots, \tilde{A}^{K-1}\tilde{B}) =: \mathcal{K}_K(\tilde{A}, \tilde{B})$$

  (where $\tilde{A} = (s_0 E - A)^{-1} E$, $\tilde{B} = (s_0 E - A)^{-1} B$) along

$$\mathcal{W} = \mathrm{span}(\tilde{C}, \tilde{A}^* \tilde{C}^T, \ldots, (\tilde{A}^*)^{K-1}\tilde{C}) =: \mathcal{K}_K(\tilde{A}^*, \tilde{C}).$$

  (where $\tilde{A}^* = (s_* E - A)^{-T} E^T$, $\tilde{C} = (s_* E - A)^{-T} C^T$).

- Computation via unsymmetric Lanczos method.

Padé Approximation

The Padé-Lanczos Connection [Gallivan/Grimme/Van Dooren 1994, Freund/Feldmann 1994]

Padé-via-Lanczos Method (PVL)

- Padé approximation/moment matching yield:

$$m_k = \frac{1}{k!} G^{(k)}(s_0) = \frac{1}{k!} \hat{G}^{(k)}(s_0) = \hat{m}_k, \quad k = 0, \ldots, 2K-1,$$

  i.e., Hermite interpolation in s_0.

- Recall interpolation via projection result $\Rightarrow$ moments need not be computed explicitly; moment matching is equivalent to projecting state-space onto

$$\mathcal{V} = \operatorname{span}(\tilde{B}, \tilde{A}\tilde{B}, \ldots, \tilde{A}^{K-1}\tilde{B}) =: \mathcal{K}_K(\tilde{A}, \tilde{B})$$

  (where $\tilde{A} = (s_0 E - A)^{-1} E$, $\tilde{B} = (s_0 E - A)^{-1} B$) along

$$\mathcal{W} = \operatorname{span}(\tilde{C}, \tilde{A}^* \tilde{C}^T, \ldots, (\tilde{A}^*)^{K-1}\tilde{C}) =: \mathcal{K}_K(\tilde{A}^*, \tilde{C}).$$

  (where $\tilde{A}^* = (s_* E - A)^{-T} E^T$, $\tilde{C} = (s_* E - A)^{-T} C^T$).

- Computation via unsymmetric Lanczos method.

Padé Approximation

The Padé-Lanczos Connection [Gallivan/Grimme/Van Dooren 1994, Freund/Feldmann 1994]

Padé-via-Lanczos Method (PVL)

- Padé approximation/moment matching yield:

$$m_k = \frac{1}{k!} G^{(k)}(s_0) = \frac{1}{k!} \hat{G}^{(k)}(s_0) = \hat{m}_k, \quad k = 0, \ldots, 2K-1,$$

  i.e., Hermite interpolation in s_0.

- Recall interpolation via projection result $\Rightarrow$ moments need not be computed explicitly; moment matching is equivalent to projecting state-space onto

$$\mathcal{V} = \mathrm{span}(\tilde{B}, \tilde{A}\tilde{B}, \ldots, \tilde{A}^{K-1}\tilde{B}) =: \mathcal{K}_K(\tilde{A}, \tilde{B})$$

  (where $\tilde{A} = (s_0 E - A)^{-1} E$, $\tilde{B} = (s_0 E - A)^{-1} B$) along

$$\mathcal{W} = \mathrm{span}(\tilde{C}, \tilde{A}^* \tilde{C}^T, \ldots, (\tilde{A}^*)^{K-1}\tilde{C}) =: \mathcal{K}_K(\tilde{A}^*, \tilde{C}).$$

  (where $\tilde{A}^* = (s_* E - A)^{-T} E^T$, $\tilde{C} = (s_* E - A)^{-T} C^T$).

- Computation via unsymmetric Lanczos method.

Padé Approximation

The Padé-Lanczos Connection [Gallivan/Grimme/Van Dooren 1994, Freund/Feldmann 1994]

Padé-via-Lanczos Method (PVL)

- Padé approximation/moment matching yield:

$$m_k = \frac{1}{k!} G^{(k)}(s_0) = \frac{1}{k!} \hat{G}^{(k)}(s_0) = \hat{m}_k, \quad k = 0, \ldots, 2K-1,$$

  i.e., Hermite interpolation in s_0.

- Recall interpolation via projection result $\Rightarrow$ moments need not be computed explicitly; moment matching is equivalent to projecting state-space onto

$$\mathcal{V} = \mathrm{span}(\tilde{B}, \tilde{A}\tilde{B}, \ldots, \tilde{A}^{K-1}\tilde{B}) =: \mathcal{K}_K(\tilde{A}, \tilde{B})$$

  (where $\tilde{A} = (s_0 E - A)^{-1} E$, $\tilde{B} = (s_0 E - A)^{-1} B$) along

$$\mathcal{W} = \mathrm{span}(\tilde{C}, \tilde{A}^* \tilde{C}^T, \ldots, (\tilde{A}^*)^{K-1}\tilde{C}) =: \mathcal{K}_K(\tilde{A}^*, \tilde{C}).$$

  (where $\tilde{A}^* = (s_* E - A)^{-T} E^T$, $\tilde{C} = (s_* E - A)^{-T} C^T$).

- Computation via unsymmetric Lanczos method.

Remark: Arnoldi (PRIMA) yields only $G(s) = \hat{G}(s) + \mathcal{O}((s - s_0)^r)$.

Padé Approximation

The Padé-Lanczos Connection [Gallivan/Grimme/Van Dooren 1994, Freund/Feldmann 1994]

Padé-via-Lanczos Method (PVL)

Difficulties:

- Computable error estimates/bounds for $\|y - \hat{y}\|_2$ often very pessimistic or expensive to evaluate.
- Mostly heuristic criteria for choice of expansion points.
  Optimal choice for second-order systems with proportional/Rayleigh damping (Beattie/Gugercin '05).
- Good approximation quality only locally.
- Preservation of physical properties only in special cases (e.g. PRIMA/Arnoldi: $V^T AV$ is stable if A is negative definite or dissipative $\rightsquigarrow$ exercises); usually requires post processing which (partially) destroys moment matching properties.

Padé Approximation

The Padé-Lanczos Connection [Gallivan/Grimme/Van Dooren 1994, Freund/Feldmann 1994]

Padé-via-Lanczos Method (PVL)

Difficulties:

- Computable error estimates/bounds for $\|y - \hat{y}\|_2$ often very pessimistic or expensive to evaluate.
- Mostly heuristic criteria for choice of expansion points.
  Optimal choice for second-order systems with proportional/Rayleigh damping (BEATTIE/GUGERCIN '05).
- Good approximation quality only locally.
- Preservation of physical properties only in special cases (e.g. PRIMA/Arnoldi: $V^T AV$ is stable if A is negative definite or dissipative $\rightsquigarrow$ exercises); usually requires post processing which (partially) destroys moment matching properties.

Padé Approximation

The Padé-Lanczos Connection [Gallivan/Grimme/Van Dooren 1994, Freund/Feldmann 1994]

Padé-via-Lanczos Method (PVL)

Difficulties:

- Computable error estimates/bounds for $\|y - \hat{y}\|_2$ often very pessimistic or expensive to evaluate.
- Mostly heuristic criteria for choice of expansion points.
  Optimal choice for second-order systems with proportional/Rayleigh damping (Beattie/Gugercin '05).
- Good approximation quality only locally.
- Preservation of physical properties only in special cases (e.g. PRIMA/Arnoldi: $V^T AV$ is stable if A is negative definite or dissipative $\rightsquigarrow$ exercises); usually requires post processing which (partially) destroys moment matching properties.

Padé Approximation

The Padé-Lanczos Connection [Gallivan/Grimme/Van Dooren 1994, Freund/Feldmann 1994]

Padé-via-Lanczos Method (PVL)

Difficulties:

- Computable error estimates/bounds for $\|y - \hat{y}\|_2$ often very pessimistic or expensive to evaluate.
- Mostly heuristic criteria for choice of expansion points.
  Optimal choice for second-order systems with proportional/Rayleigh damping (Beattie/Gugercin '05).
- Good approximation quality only locally.
- Preservation of physical properties only in special cases (e.g. PRIMA/Arnoldi: $V^T AV$ is stable if A is negative definite or dissipative $\rightsquigarrow$ exercises); usually requires post processing which (partially) destroys moment matching properties.

Interpolatory Model Reduction

A Change of Perspective: Rational Interpolation

Theorem (simplified) [Grimme '97, Villemagne/Skelton '87]

If

$$\begin{aligned} \operatorname{span}\left\{(s_1 I_n - A)^{-1}B, \ldots, (s_k I_n - A)^{-1}B\right\} &\subset \operatorname{Ran}(V), \\ \operatorname{span}\left\{(s_1 I_n - A)^{-T}C^T, \ldots, (s_k I_n - A)^{-T}C^T\right\} &\subset \operatorname{Ran}(W), \end{aligned}$$

then

$$G(s_j) = \hat{G}(s_j), \quad \frac{d}{ds}G(s_j) = \frac{d}{ds}\hat{G}(s_j), \quad \text{for } j = 1, \ldots, k.$$

Interpolatory Model Reduction

A Change of Perspective: Rational Interpolation

Theorem (simplified) [Grimme '97, Villemagne/Skelton '87]

If

$$\begin{aligned} \operatorname{span}\left\{(s_1 I_n - A)^{-1}B, \ldots, (s_k I_n - A)^{-1}B\right\} &\subset \operatorname{Ran}(V), \\ \operatorname{span}\left\{(s_1 I_n - A)^{-T}C^T, \ldots, (s_k I_n - A)^{-T}C^T\right\} &\subset \operatorname{Ran}(W), \end{aligned}$$

then

$$G(s_j) = \hat{G}(s_j), \quad \frac{d}{ds}G(s_j) = \frac{d}{ds}\hat{G}(s_j), \quad \text{for } j = 1, \ldots, k.$$

Remark:

computation of V, W from rational Krylov subspaces, e.g.,

- dual rational Arnoldi/Lanczos [Grimme '97],
- Iterative Rational Krylov-Algo. [Antoulas/Beattie/Gugercin '07].

$\mathcal{H}_2$-Optimal Model Reduction

Best $\mathcal{H}_2$-norm approximation problem

$$\text{Find} \quad \arg\min_{\hat{G}\in\mathcal{H}_2 \text{ of order } \leq r} \|G - \hat{G}\|_2.$$

$\mathcal{H}_2$-Optimal Model Reduction

Best $\mathcal{H}_2$-norm approximation problem

$$\text{Find} \quad \arg\min_{\hat{G}\in\mathcal{H}_2 \text{ of order } \leq r} \|G - \hat{G}\|_2.$$

$\rightsquigarrow$ First-order necessary $\mathcal{H}_2$-optimality conditions:

For SISO systems

$$G(-\mu_i) = \hat{G}(-\mu_i),$$
$$G'(-\mu_i) = \hat{G}'(-\mu_i),$$

where μ_i are the poles of the reduced transfer function $\hat{G}$.

$\mathcal{H}_2$-Optimal Model Reduction

Best $\mathcal{H}_2$-norm approximation problem

$$\text{Find} \quad \arg\min_{\hat{G}\in\mathcal{H}_2 \text{ of order } \leq r} \|G - \hat{G}\|_2.$$

$\rightsquigarrow$ First-order necessary $\mathcal{H}_2$-optimality conditions:

For MIMO systems

$$\begin{aligned} G(-\mu_i)\tilde{B}_i &= \hat{G}(-\mu_i)\tilde{B}_i, && \text{for } i = 1, \ldots, r, \\ \tilde{C}_i^T G(-\mu_i) &= \tilde{C}_i^T \hat{G}(-\mu_i), && \text{for } i = 1, \ldots, r, \\ \tilde{C}_i^T G'(-\mu_i)\tilde{B}_i &= \tilde{C}_i^T \hat{G}'(-\mu_i)\tilde{B}_i, && \text{for } i = 1, \ldots, r, \end{aligned}$$

where $T^{-1}\hat{A}T = \operatorname{diag}\{\mu_1, \ldots, \mu_r\}$ = spectral decomposition and

$$\tilde{B} = \hat{B}^T T^{-T}, \quad \tilde{C} = \hat{C}T.$$

$\rightsquigarrow$ tangential interpolation conditions.

Interpolatory Model Reduction

Interpolation of the Transfer Function by Projection

Construct reduced transfer function by Petrov-Galerkin projection $\mathcal{P} = VW^T$, i.e.

$$\hat{G}(s) = CV\left(sI - W^T AV\right)^{-1} W^T B,$$

where V and W are given as the rational Krylov subspaces

$$V = \left[(-\mu_1 I - A)^{-1}B, \ldots, (-\mu_r I - A)^{-1}B\right],$$
$$W = \left[(-\mu_1 I - A^T)^{-1}C^T, \ldots, (-\mu_r I - A^T)^{-1}C^T\right].$$

Then

$$G(-\mu_i) = \hat{G}(-\mu_i) \quad \text{and} \quad G'(-\mu_i) = \hat{G}'(-\mu_i),$$

for $i = 1, \ldots, r$ as desired.

$\rightsquigarrow$ iterative algorithms (IRKA/MIRIAm) that yield $\mathcal{H}_2$-optimal models.

[Gugercin et al. '06], [Bunse-Gerstner et al. '07], [Van Dooren et al. '08]

Interpolatory Model Reduction

Interpolation of the Transfer Function by Projection

Construct reduced transfer function by Petrov-Galerkin projection $\mathcal{P} = VW^T$, i.e.

$$\hat{G}(s) = CV\left(sI - W^TAV\right)^{-1} W^TB,$$

where V and W are given as the rational Krylov subspaces

$$V = \left[(-\mu_1 I - A)^{-1}B, \ldots, (-\mu_r I - A)^{-1}B\right],$$
$$W = \left[(-\mu_1 I - A^T)^{-1}C^T, \ldots, (-\mu_r I - A^T)^{-1}C^T\right].$$

Then

$$G(-\mu_i) = \hat{G}(-\mu_i) \quad \text{and} \quad G'(-\mu_i) = \hat{G}'(-\mu_i),$$

for $i = 1, \ldots, r$ as desired.

$\rightsquigarrow$ iterative algorithms (IRKA/MIRIAm) that yield $\mathcal{H}_2$-optimal models.

[Gugercin et al. '06], [Bunse-Gerstner et al. '07], [Van Dooren et al. '08]

Interpolatory Model Reduction

Interpolation of the Transfer Function by Projection

Construct reduced transfer function by Petrov-Galerkin projection $\mathcal{P} = VW^T$, i.e.

$$\hat{G}(s) = CV\left(sI - W^T AV\right)^{-1} W^T B,$$

where V and W are given as the rational Krylov subspaces

$$\begin{aligned} V &= \left[(-\mu_1 I - A)^{-1}B, \ldots, (-\mu_r I - A)^{-1}B\right], \\ W &= \left[(-\mu_1 I - A^T)^{-1}C^T, \ldots, (-\mu_r I - A^T)^{-1}C^T\right]. \end{aligned}$$

Then

$$G(-\mu_i) = \hat{G}(-\mu_i) \quad \text{and} \quad G'(-\mu_i) = \hat{G}'(-\mu_i),$$

for $i = 1, \ldots, r$ as desired.

$\rightsquigarrow$ iterative algorithms (IRKA/MIRIAm) that yield $\mathcal{H}_2$-optimal models.

[Gugercin et al. '06], [Bunse-Gerstner et al. '07], [Van Dooren et al. '08]

Interpolatory Model Reduction

Interpolation of the Transfer Function by Projection

Construct reduced transfer function by Petrov-Galerkin projection $\mathcal{P} = VW^T$, i.e.

$$\hat{G}(s) = CV\left(sI - W^TAV\right)^{-1} W^TB,$$

where V and W are given as the rational Krylov subspaces

$$\begin{aligned} V &= \left[(-\mu_1 I - A)^{-1}B, \ldots, (-\mu_r I - A)^{-1}B\right], \\ W &= \left[(-\mu_1 I - A^T)^{-1}C^T, \ldots, (-\mu_r I - A^T)^{-1}C^T\right]. \end{aligned}$$

Then

$$G(-\mu_i) = \hat{G}(-\mu_i) \quad \text{and} \quad G'(-\mu_i) = \hat{G}'(-\mu_i),$$

for $i = 1, \ldots, r$ as desired.
$\rightsquigarrow$ iterative algorithms (IRKA/MIRIAm) that yield $\mathcal{H}_2$-optimal models.

[Gugercin et al. '06], [Bunse-Gerstner et al. '07], [Van Dooren et al. '08]

$\mathcal{H}_2$-Optimal Model Reduction

The Basic IRKA Algorithm

Algorithm 1 IRKA (MIMO version/MIRIAm)

Input: A stable, B, C, $\hat{A}$ stable, $\hat{B}$, $\hat{C}$, $\delta > 0$.
Output: A^{opt}, B^{opt}, C^{opt}

1: **while** $\left(\max_{j=1,\ldots,r} \left\{ \frac{|\mu_j - \mu_j^{\text{old}}|}{|\mu_j|} \right\} > \delta\right)$ **do**
2: $\quad \operatorname{diag}\{\mu_1, \ldots, \mu_r\} := T^{-1}\hat{A}T =$ spectral decomposition, $\tilde{B} = \hat{B}^H T^{-T}$, $\tilde{C} = \hat{C}T$.
3: $\quad V = \left[(-\mu_1 I - A)^{-1} B\tilde{B}_1, \ldots, (-\mu_r I - A)^{-1} B\tilde{B}_r\right]$
4: $\quad W = \left[(-\mu_1 I - A^T)^{-1} C^T\tilde{C}_1, \ldots, (-\mu_r I - A^T)^{-1} C^T\tilde{C}_r\right]$
5: $\quad V = \operatorname{orth}(V)$, $W = \operatorname{orth}(W)$, $W = W(V^H W)^{-1}$
6: $\quad \hat{A} = W^H A V$, $\hat{B} = W^H B$, $\hat{C} = CV$
7: **end while**
8: $A^{opt} = \hat{A}$, $B^{opt} = \hat{B}$, $C^{opt} = \hat{C}$

Outline

1. Introduction
2. Model Reduction by Projection
3. Interpolatory Model Reduction
4. Balanced Truncation
   - The basic method
   - Numerical examples for BT
   - Software
5. Nonlinear Model Reduction
6. Final Remarks

Balanced Truncation

Basic principle:

- Recall: a stable system Σ, realized by (A, B, C, D), is called balanced, if the Gramians, i.e., solutions P, Q of the Lyapunov equations

$$AP + PA^T + BB^T = 0, \qquad A^TQ + QA + C^TC = 0,$$

satisfy: $P = Q = \mathrm{diag}(\sigma_1, \ldots, \sigma_n)$ with $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_n > 0$.

- $\Lambda(PQ)^{\frac{1}{2}} = \{\sigma_1, \ldots, \sigma_n\}$ are the Hankel singular values (HSVs) of Σ.

Balanced Truncation

Basic principle:

- Recall: a stable system Σ, realized by (A, B, C, D), is called balanced, if the Gramians, i.e., solutions P, Q of the Lyapunov equations

$$AP + PA^T + BB^T = 0, \qquad A^TQ + QA + C^TC = 0,$$

satisfy: $P = Q = \mathrm{diag}(\sigma_1, \ldots, \sigma_n)$ with $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_n > 0$.

- $\Lambda\,(PQ)^{\frac{1}{2}} = \{\sigma_1, \ldots, \sigma_n\}$ are the Hankel singular values (HSVs) of Σ.

Balanced Truncation

Basic principle:

- Recall: a stable system Σ, realized by (A, B, C, D), is called balanced, if the Gramians, i.e., solutions P, Q of the Lyapunov equations

$$AP + PA^T + BB^T = 0, \qquad A^TQ + QA + C^TC = 0,$$

satisfy: $P = Q = \mathrm{diag}(\sigma_1, \ldots, \sigma_n)$ with $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_n > 0$.

- $\Lambda\,(PQ)^{\frac{1}{2}} = \{\sigma_1, \ldots, \sigma_n\}$ are the Hankel singular values (HSVs) of Σ.
- Compute balanced realization of the system via state-space transformation

$$\begin{aligned} \mathcal{T} : (A, B, C, D) &\mapsto (TAT^{-1}, TB, CT^{-1}, D) \\ &= \left(\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}, \begin{bmatrix} B_1 \\ B_2 \end{bmatrix}, \begin{bmatrix} C_1 & C_2 \end{bmatrix}, D \right) \end{aligned}$$

- Truncation $\rightsquigarrow (\hat{A}, \hat{B}, \hat{C}, \hat{D}) := (A_{11}, B_1, C_1, D)$.

Balanced Truncation

Basic principle:

- Recall: a stable system Σ, realized by (A, B, C, D), is called balanced, if the Gramians, i.e., solutions P, Q of the Lyapunov equations

$$AP + PA^T + BB^T = 0, \qquad A^TQ + QA + C^TC = 0,$$

satisfy: $P = Q = \mathrm{diag}(\sigma_1, \ldots, \sigma_n)$ with $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_n > 0$.

- $\Lambda\,(PQ)^{\frac{1}{2}} = \{\sigma_1, \ldots, \sigma_n\}$ are the Hankel singular values (HSVs) of Σ.
- Compute balanced realization of the system via state-space transformation

$$\begin{aligned} \mathcal{T} : (A, B, C, D) \quad &\mapsto \quad (TAT^{-1}, TB, CT^{-1}, D) \\ &= \quad \left(\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}, \begin{bmatrix} B_1 \\ B_2 \end{bmatrix}, \begin{bmatrix} C_1 & C_2 \end{bmatrix}, D \right) \end{aligned}$$

- Truncation $\rightsquigarrow (\hat{A}, \hat{B}, \hat{C}, \hat{D}) := (A_{11}, B_1, C_1, D)$.

Balanced Truncation

Implementation: SR Method

1. Compute (Cholesky) factors of the Gramians, $P = S^T S,\ Q = R^T R.$
2. Compute SVD $SR^T = [\,U_1,\ U_2\,] \begin{bmatrix} \Sigma_1 & \\ & \Sigma_2 \end{bmatrix} \begin{bmatrix} V_1^T \\ V_2^T \end{bmatrix}.$
3. ROM is $(W^T AV, W^T B, CV, D)$, where

$$W = R^T V_1 \Sigma_1^{-\frac{1}{2}}, \qquad V = S^T U_1 \Sigma_1^{-\frac{1}{2}}.$$

Balanced Truncation

Implementation: SR Method

1. Compute (Cholesky) factors of the Gramians, $P = S^T S, \; Q = R^T R$.
2. Compute SVD $SR^T = [\, U_1, \; U_2 \,] \begin{bmatrix} \Sigma_1 & \\ & \Sigma_2 \end{bmatrix} \begin{bmatrix} V_1^T \\ V_2^T \end{bmatrix}$.
3. ROM is $(W^T AV, W^T B, CV, D)$, where

$$W = R^T V_1 \Sigma_1^{-\frac{1}{2}}, \qquad V = S^T U_1 \Sigma_1^{-\frac{1}{2}}.$$

Balanced Truncation

Implementation: SR Method

1. Compute (Cholesky) factors of the Gramians, $P = S^T S, \; Q = R^T R.$
2. Compute SVD $SR^T = [\, U_1, \; U_2 \,] \begin{bmatrix} \Sigma_1 & \\ & \Sigma_2 \end{bmatrix} \begin{bmatrix} V_1^T \\ V_2^T \end{bmatrix}.$
3. ROM is $(W^T AV, W^T B, CV, D)$, where

$$W = R^T V_1 \Sigma_1^{-\frac{1}{2}}, \qquad V = S^T U_1 \Sigma_1^{-\frac{1}{2}}.$$

Balanced Truncation

Implementation: SR Method

1. Compute (Cholesky) factors of the Gramians, $P = S^T S, \; Q = R^T R.$
2. Compute SVD $SR^T = [\, U_1, \; U_2 \,] \begin{bmatrix} \Sigma_1 & \\ & \Sigma_2 \end{bmatrix} \begin{bmatrix} V_1^T \\ V_2^T \end{bmatrix}.$
3. ROM is $(W^T AV, W^T B, CV, D)$, where

$$W = R^T V_1 \Sigma_1^{-\frac{1}{2}}, \qquad V = S^T U_1 \Sigma_1^{-\frac{1}{2}}.$$

Note:

$$V^T W \quad = \quad (\Sigma_1^{-\frac{1}{2}} U_1^T S)(R^T V_1 \Sigma_1^{-\frac{1}{2}})$$

Balanced Truncation

Implementation: SR Method

1. Compute (Cholesky) factors of the Gramians, $P = S^T S,\ Q = R^T R$.
2. Compute SVD $SR^T = [\, U_1,\ U_2 \,] \begin{bmatrix} \Sigma_1 & \\ & \Sigma_2 \end{bmatrix} \begin{bmatrix} V_1^T \\ V_2^T \end{bmatrix}$.
3. ROM is $(W^T AV, W^T B, CV, D)$, where

$$W = R^T V_1 \Sigma_1^{-\frac{1}{2}}, \qquad V = S^T U_1 \Sigma_1^{-\frac{1}{2}}.$$

Note:

$$V^T W \;=\; (\Sigma_1^{-\frac{1}{2}} U_1^T S)(R^T V_1 \Sigma_1^{-\frac{1}{2}}) \;=\; \Sigma_1^{-\frac{1}{2}} U_1^T U \Sigma V^T V_1 \Sigma_1^{-\frac{1}{2}}$$

Balanced Truncation

Implementation: SR Method

1. Compute (Cholesky) factors of the Gramians, $P = S^T S,\ Q = R^T R$.
2. Compute SVD $SR^T = [\, U_1,\ U_2\,] \begin{bmatrix} \Sigma_1 & \\ & \Sigma_2 \end{bmatrix} \begin{bmatrix} V_1^T \\ V_2^T \end{bmatrix}$.
3. ROM is $(W^T AV, W^T B, CV, D)$, where

$$W = R^T V_1 \Sigma_1^{-\frac{1}{2}}, \qquad V = S^T U_1 \Sigma_1^{-\frac{1}{2}}.$$

Note:

$$\begin{aligned} V^T W &= (\Sigma_1^{-\frac{1}{2}} U_1^T S)(R^T V_1 \Sigma_1^{-\frac{1}{2}}) = \Sigma_1^{-\frac{1}{2}} U_1^T U \Sigma V^T V_1 \Sigma_1^{-\frac{1}{2}} \\ &= \Sigma_1^{-\frac{1}{2}} [\, I_r,\ 0\,] \begin{bmatrix} \Sigma_1 & \\ & \Sigma_2 \end{bmatrix} \begin{bmatrix} I_r \\ 0 \end{bmatrix} \Sigma_1^{-\frac{1}{2}} \end{aligned}$$

Balanced Truncation

Implementation: SR Method

1. Compute (Cholesky) factors of the Gramians, $P = S^T S, \; Q = R^T R$.
2. Compute SVD $SR^T = [\, U_1, \; U_2 \,] \begin{bmatrix} \Sigma_1 & \\ & \Sigma_2 \end{bmatrix} \begin{bmatrix} V_1^T \\ V_2^T \end{bmatrix}$.
3. ROM is $(W^T AV, W^T B, CV, D)$, where

$$W = R^T V_1 \Sigma_1^{-\frac{1}{2}}, \qquad V = S^T U_1 \Sigma_1^{-\frac{1}{2}}.$$

Note:

$$\begin{aligned} V^T W &= (\Sigma_1^{-\frac{1}{2}} U_1^T S)(R^T V_1 \Sigma_1^{-\frac{1}{2}}) = \Sigma_1^{-\frac{1}{2}} U_1^T U \Sigma V^T V_1 \Sigma_1^{-\frac{1}{2}} \\ &= \Sigma_1^{-\frac{1}{2}} [\, I_r, \, 0 \,] \begin{bmatrix} \Sigma_1 & \\ & \Sigma_2 \end{bmatrix} \begin{bmatrix} I_r \\ 0 \end{bmatrix} \Sigma_1^{-\frac{1}{2}} = \Sigma_1^{-\frac{1}{2}} \Sigma_1 \Sigma_1^{-\frac{1}{2}} = I_r \end{aligned}$$

$\Longrightarrow$ VW^T is an oblique projector, hence balanced truncation is a Petrov-Galerkin projection method.

Balanced Truncation

Properties:

- Reduced-order model is stable with HSVs $\sigma_1, \ldots, \sigma_r$.
- Adaptive choice of r via computable error bound:

$$\|y - \hat{y}\|_2 \leq \left(2 \sum\nolimits_{k=r+1}^{n} \sigma_k\right) \|u\|_2.$$

Balanced Truncation

Properties:

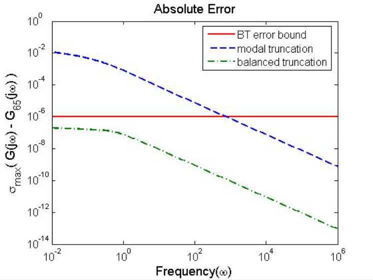
- Reduced-order model is stable with HSVs $\sigma_1, \ldots, \sigma_r$.
- Adaptive choice of r via computable error bound:

$$\|y - \hat{y}\|_2 \leq \left(2 \sum\nolimits_{k=r+1}^{n} \sigma_k\right) \|u\|_2.$$

Balanced Truncation

Numerical examples for BT: Optimal Cooling of Steel Profiles

$n = 1,357$, Absolute Error

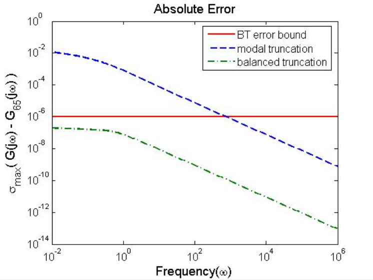


- BT model computed with sign function method,
- MT w/o static condensation, same order as BT model.

Balanced Truncation

Numerical examples for BT: Optimal Cooling of Steel Profiles

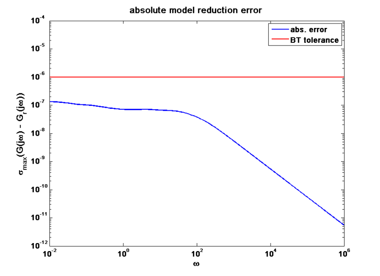
$n = 1,357$, Absolute Error



– BT model computed with sign function method,

– MT w/o static condensation, same order as BT model.

$n = 79,841$, Absolute Error



– BT model computed using M.E.S.S. in MATLAB,

– Computation time: ≈1 min.

Balanced Truncation

Numerical examples for BT: Microgyroscope (Butterfly Gyro)

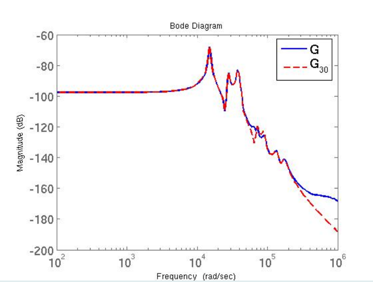
- FEM discretization of structure dynamical model using quadratic tetrahedral elements (ANSYS-SOLID187)
  $\rightsquigarrow$ $n = 34,722$, $m = 1$, $q = 12$.
- Reduced model computed using SpaRed, $r = 30$.

Balanced Truncation

Numerical examples for BT: Microgyroscope (Butterfly Gyro)

- FEM discretization of structure dynamical model using quadratic tetrahedral elements (ANSYS-SOLID187)
  $\rightsquigarrow$ $n = 34,722$, $m = 1$, $q = 12$.
- Reduced model computed using SpaRed, $r = 30$.

Frequency Repsonse Analysis

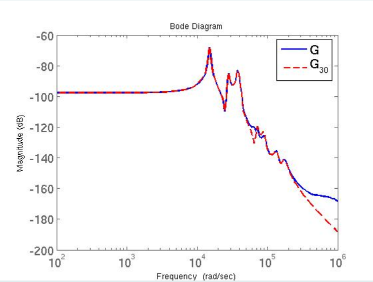


Balanced Truncation

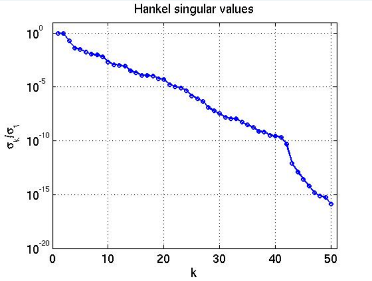
Numerical examples for BT: Microgyroscope (Butterfly Gyro)

- FEM discretization of structure dynamical model using quadratic tetrahedral elements (ANSYS-SOLID187)
  $\rightsquigarrow$ $n = 34,722$, $m = 1$, $q = 12$.
- Reduced model computed using SpaRed, $r = 30$.

Frequency Repsonse Analysis



Hankel Singular Values



Balanced Truncation

Software

Lyapack [Penzl 2000]

MATLAB toolbox for solving

- Lyapunov equations and algebraic Riccati equations,
- model reduction and LQR problems.

Main work horse: Low-rank ADI and Newton-ADI iterations.

Balanced Truncation
Software

Lyapack [Penzl 2000]

MATLAB toolbox for solving

- Lyapunov equations and algebraic Riccati equations,
- model reduction and LQR problems.

Main work horse: Low-rank ADI and Newton-ADI iterations.

M.E.S.S. – Matrix Equations Sparse Solvers [B./Köhler/Saak '08–]

- Extended and revised version of LYAPACK.
- Includes solvers for large-scale differential Riccati equations (based on Rosenbrock and BDF methods).
- Many algorithmic improvements:
  - new ADI parameter selection,
  - column compression based on RRQR,
  - more efficient use of direct solvers,
  - treatment of generalized systems without factorization of the mass matrix,
  - new ADI versions avoiding complex arithmetic etc.
- C and MATLAB versions.

Balanced Truncation
Software

Lyapack [Penzl 2000]

MATLAB toolbox for solving

- Lyapunov equations and algebraic Riccati equations,
- model reduction and LQR problems.

Main work horse: Low-rank ADI and Newton-ADI iterations.

M.E.S.S. – Matrix Equations Sparse Solvers [B./Köhler/Saak '08–]

- Extended and revised version of LYAPACK.
- Includes solvers for large-scale differential Riccati equations (based on Rosenbrock and BDF methods).
- Many algorithmic improvements:
  - new ADI parameter selection,
  - column compression based on RRQR,
  - more efficient use of direct solvers,
  - treatment of generalized systems without factorization of the mass matrix,
  - new ADI versions avoiding complex arithmetic etc.
- C and MATLAB versions.

Balanced Truncation

Software

Lyapack [Penzl 2000]

MATLAB toolbox for solving

- Lyapunov equations and algebraic Riccati equations,
- model reduction and LQR problems.

Main work horse: Low-rank ADI and Newton-ADI iterations.

M.E.S.S. – Matrix Equations Sparse Solvers [B./Köhler/Saak '08–]

- Extended and revised version of LYAPACK.
- Includes solvers for large-scale differential Riccati equations (based on Rosenbrock and BDF methods).
- Many algorithmic improvements:
  - new ADI parameter selection,
  - column compression based on RRQR,
  - more efficient use of direct solvers,
  - treatment of generalized systems without factorization of the mass matrix,
  - new ADI versions avoiding complex arithmetic etc.
- C and MATLAB versions.

Outline

1. Introduction
2. Model Reduction by Projection
3. Interpolatory Model Reduction
4. Balanced Truncation
5. Nonlinear Model Reduction
   - A Brief Introduction
   - Nonlinear Model Reduction by Generalized Moment-Matching
   - Numerical Examples
6. Final Remarks

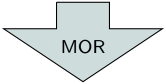
Nonlinear Model Reduction

A Brief Introduction

Given a large-scale control-affine nonlinear control system of the form

$$\Sigma : \begin{cases} \dot{x}(t) = f(x(t)) + bu(t), \\ y(t) = c^T x(t), \quad x(0) = x_0, \end{cases}$$

with $f : \mathbb{R}^n \to \mathbb{R}^n$ nonlinear and $b, c \in \mathbb{R}^n$, $x \in \mathbb{R}^n$, $u, y \in \mathbb{R}$.



$$\hat{\Sigma} : \begin{cases} \dot{\hat{x}}(t) = \hat{f}(\hat{x}(t)) + \hat{b}u(t), \\ \hat{y}(t) = \hat{c}^T \hat{x}(t), \quad \hat{x}(0) = \hat{x}_0, \end{cases}$$

with $\hat{f} : \mathbb{R}^{\hat{n}} \to \mathbb{R}^{\hat{n}}$ and $\hat{b}, \hat{c} \in \mathbb{R}^{\hat{n}}$, $x \in \mathbb{R}^{\hat{n}}$, $u \in \mathbb{R}$ and $\hat{y} \approx y \in \mathbb{R}$, $\hat{n} \ll n$.

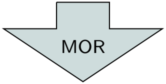
Nonlinear Model Reduction

A Brief Introduction

Given a large-scale control-affine nonlinear control system of the form

$$\Sigma : \begin{cases} \dot{x}(t) = f(x(t)) + bu(t), \\ y(t) = c^T x(t), \quad x(0) = x_0, \end{cases}$$

with $f : \mathbb{R}^n \to \mathbb{R}^n$ nonlinear and $b, c \in \mathbb{R}^n$, $x \in \mathbb{R}^n$, $u, y \in \mathbb{R}$.



$$\hat{\Sigma} : \begin{cases} \dot{\hat{x}}(t) = \hat{f}(\hat{x}(t)) + \hat{b}u(t), \\ \hat{y}(t) = \hat{c}^T \hat{x}(t), \quad \hat{x}(0) = \hat{x}_0, \end{cases}$$

with $\hat{f} : \mathbb{R}^{\hat{n}} \to \mathbb{R}^{\hat{n}}$ and $\hat{b}, \hat{c} \in \mathbb{R}^{\hat{n}}$, $x \in \mathbb{R}^{\hat{n}}$, $u \in \mathbb{R}$ and $\hat{y} \approx y \in \mathbb{R}$, $\hat{n} \ll n$.

Nonlinear Model Reduction

Common Reduction Techniques

Proper Orthogonal Decomposition (POD)

- Take computed or experimental 'snapshots' of full model: $[x(t_1), x(t_2), \ldots, x(t_N)] =: X$,
- perform SVD of snapshot matrix: $X = VSW^T \approx V_{\hat{n}} S_{\hat{n}} W_{\hat{n}}^T$.
- Reduction by POD-Galerkin projection: $\dot{\hat{x}} = V_{\hat{n}}^T f(V_{\hat{n}} \hat{x}) + V_{\hat{n}}^T Bu$.
- Requires evaluation of f
  $\rightsquigarrow$ discrete empirical interpolation [Sorensen/Chaturantabut '09].
- Input dependency due to 'snapshots'!

Trajectory Piecewise Linear (TPWL)

- Linearize f along trajectory,
- reduce resulting linear systems,
- construct reduced model by weighted sum of linear systems.
- Requires simulation of original model and several linear reduction steps, many heuristics.

Nonlinear Model Reduction

Common Reduction Techniques

Proper Orthogonal Decomposition (POD)

- Take computed or experimental 'snapshots' of full model: $[x(t_1), x(t_2), \dots, x(t_N)] =: X$,
- perform SVD of snapshot matrix: $X = VSW^T \approx V_{\hat{n}} S_{\hat{n}} W_{\hat{n}}^T$.
- Reduction by POD-Galerkin projection: $\dot{\hat{x}} = V_{\hat{n}}^T f(V_{\hat{n}} \hat{x}) + V_{\hat{n}}^T Bu$.
- Requires evaluation of f
  $\rightsquigarrow$ discrete empirical interpolation [Sorensen/Chaturantabut '09].
- Input dependency due to 'snapshots'!

Trajectory Piecewise Linear (TPWL)

- Linearize f along trajectory,
- reduce resulting linear systems,
- construct reduced model by weighted sum of linear systems.
- Requires simulation of original model and several linear reduction steps, many heuristics.

Nonlinear Model Reduction by Generalized Moment-Matching

Quadratic-Bilinear Differential Algebraic Equations (QBDAEs)

Consider the class of quadratic-bilinear differential algebraic equations

$$\Sigma : \quad \begin{cases} E\dot{x}(t) = A_1 x(t) + A_2 x(t) \otimes x(t) + Nx(t)u(t) + Bu(t), \\ y(t) = Cx(t), \quad x(0) = x_0, \end{cases}$$

where $E, A_1, N \in \mathbb{R}^{n \times n}, A_2 \in \mathbb{R}^{n \times n^2}$ (Hessian tensor), $B, C^T \in \mathbb{R}^n$ are quite helpful.

- A large class of smooth nonlinear control-affine systems can be transformed into the above type of control system.
- The transformation is exact, but a slight increase of the state dimension has to be accepted.
- Input-output behavior can be characterized by generalized transfer functions $\rightsquigarrow$ enables us to use Krylov-/rational interpolation-based reduction techniques.

Nonlinear Model Reduction by Generalized Moment-Matching

Transformation to QBDAE form via McCormick relaxation

Theorem [Gu'09]

Assume that the state equation of a nonlinear system Σ is given by

$$\dot{x} = a_0 x + a_1 g_1(x) + \ldots + a_k g_k(x) + Bu,$$

where $g_i(x) : \mathbb{R}^n \to \mathbb{R}^n$ are compositions of uni-variable rational, exponential, logarithmic, trigonometric or root functions, respectively. Then, by iteratively taking derivatives and adding algebraic equations, respectively, Σ can be transformed into a system of QBDAEs.

Nonlinear Model Reduction by Generalized Moment-Matching

Transformation to QBDAE form via McCormick relaxation

Theorem [Gu'09]

Assume that the state equation of a nonlinear system Σ is given by

$$\dot{x} = a_0 x + a_1 g_1(x) + \ldots + a_k g_k(x) + Bu,$$

where $g_i(x) : \mathbb{R}^n \to \mathbb{R}^n$ are compositions of uni-variable rational, exponential, logarithmic, trigonometric or root functions, respectively. Then, by iteratively taking derivatives and adding algebraic equations, respectively, Σ can be transformed into a system of QBDAEs.

Example

- $\dot{x}_1 = \exp(-x_2) \cdot \sqrt{x_1^2 + 1}, \quad \dot{x}_2 = -x_2 + u.$
- $z_1 := \exp(-x_2),$
- $\dot{x}_1 = z_1 \cdot z_2,$

Nonlinear Model Reduction by Generalized Moment-Matching

Transformation to QBDAE form via McCormick relaxation

Theorem [Gu'09]

Assume that the state equation of a nonlinear system Σ is given by

$$\dot{x} = a_0 x + a_1 g_1(x) + \ldots + a_k g_k(x) + Bu,$$

where $g_i(x) : \mathbb{R}^n \to \mathbb{R}^n$ are compositions of uni-variable rational, exponential, logarithmic, trigonometric or root functions, respectively. Then, by iteratively taking derivatives and adding algebraic equations, respectively, Σ can be transformed into a system of QBDAEs.

Example

- $\dot{x}_1 = \exp(-x_2) \cdot \sqrt{x_1^2 + 1}, \quad \dot{x}_2 = -x_2 + u.$
- $z_1 := \exp(-x_2)$,
- $\dot{x}_1 = z_1 \cdot z_2$,

Nonlinear Model Reduction by Generalized Moment-Matching

Transformation to QBDAE form via McCormick relaxation

Theorem [Gu'09]

Assume that the state equation of a nonlinear system Σ is given by

$$\dot{x} = a_0 x + a_1 g_1(x) + \ldots + a_k g_k(x) + Bu,$$

where $g_i(x) : \mathbb{R}^n \to \mathbb{R}^n$ are compositions of uni-variable rational, exponential, logarithmic, trigonometric or root functions, respectively. Then, by iteratively taking derivatives and adding algebraic equations, respectively, Σ can be transformed into a system of QBDAEs.

Example

- $\dot{x}_1 = \exp(-x_2) \cdot \sqrt{x_1^2 + 1}, \quad \dot{x}_2 = -x_2 + u.$
- $z_1 := \exp(-x_2), \quad z_2 := \sqrt{x_1^2 + 1}.$
- $\dot{x}_1 = z_1 \cdot z_2,$

Nonlinear Model Reduction by Generalized Moment-Matching

Transformation to QBDAE form via McCormick relaxation

Theorem [Gu'09]

Assume that the state equation of a nonlinear system Σ is given by

$$\dot{x} = a_0 x + a_1 g_1(x) + \ldots + a_k g_k(x) + Bu,$$

where $g_i(x) : \mathbb{R}^n \to \mathbb{R}^n$ are compositions of uni-variable rational, exponential, logarithmic, trigonometric or root functions, respectively. Then, by iteratively taking derivatives and adding algebraic equations, respectively, Σ can be transformed into a system of QBDAEs.

Example

- $\dot{x}_1 = \exp(-x_2) \cdot \sqrt{x_1^2 + 1}, \quad \dot{x}_2 = -x_2 + u.$
- $z_1 := \exp(-x_2), \quad z_2 := \sqrt{x_1^2 + 1}.$
- $\dot{x}_1 = z_1 \cdot z_2,$

Nonlinear Model Reduction by Generalized Moment-Matching

Transformation to QBDAE form via McCormick relaxation

Theorem [Gu'09]

Assume that the state equation of a nonlinear system Σ is given by

$$\dot{x} = a_0 x + a_1 g_1(x) + \ldots + a_k g_k(x) + Bu,$$

where $g_i(x) : \mathbb{R}^n \to \mathbb{R}^n$ are compositions of uni-variable rational, exponential, logarithmic, trigonometric or root functions, respectively. Then, by iteratively taking derivatives and adding algebraic equations, respectively, Σ can be transformed into a system of QBDAEs.

Example

- $\dot{x}_1 = \exp(-x_2) \cdot \sqrt{x_1^2 + 1}, \quad \dot{x}_2 = -x_2 + u.$
- $z_1 := \exp(-x_2), \quad z_2 := \sqrt{x_1^2 + 1}.$
- $\dot{x}_1 = z_1 \cdot z_2, \quad \dot{x}_2 = -x_2 + u,$

Nonlinear Model Reduction by Generalized Moment-Matching
Transformation to QBDAE form via McCormick relaxation

Theorem [Gu'09]

Assume that the state equation of a nonlinear system Σ is given by

$$\dot{x} = a_0 x + a_1 g_1(x) + \ldots + a_k g_k(x) + Bu,$$

where $g_i(x) : \mathbb{R}^n \to \mathbb{R}^n$ are compositions of uni-variable rational, exponential, logarithmic, trigonometric or root functions, respectively. Then, by iteratively taking derivatives and adding algebraic equations, respectively, Σ can be transformed into a system of QBDAEs.

Example

- $\dot{x}_1 = \exp(-x_2) \cdot \sqrt{x_1^2 + 1}, \quad \dot{x}_2 = -x_2 + u.$
- $z_1 := \exp(-x_2), \quad z_2 := \sqrt{x_1^2 + 1}.$
- $\dot{x}_1 = z_1 \cdot z_2, \quad \dot{x}_2 = -x_2 + u, \quad \dot{z}_1 = -z_1 \cdot (-x_2 + u),$

Nonlinear Model Reduction by Generalized Moment-Matching

Transformation to QBDAE form via McCormick relaxation

Theorem [Gu'09]

Assume that the state equation of a nonlinear system Σ is given by

$$\dot{x} = a_0 x + a_1 g_1(x) + \ldots + a_k g_k(x) + Bu,$$

where $g_i(x) : \mathbb{R}^n \to \mathbb{R}^n$ are compositions of uni-variable rational, exponential, logarithmic, trigonometric or root functions, respectively. Then, by iteratively taking derivatives and adding algebraic equations, respectively, Σ can be transformed into a system of QBDAEs.

Example

- $\dot{x}_1 = \exp(-x_2) \cdot \sqrt{x_1^2 + 1}, \quad \dot{x}_2 = -x_2 + u.$
- $z_1 := \exp(-x_2), \quad z_2 := \sqrt{x_1^2 + 1}.$
- $\dot{x}_1 = z_1 \cdot z_2, \quad \dot{x}_2 = -x_2 + u, \quad \dot{z}_1 = -z_1 \cdot (-x_2 + u),$
  $\dot{z}_2 = \frac{2 \cdot x_1 \cdot z_1 \cdot z_2}{2 \cdot z_2} = x_1 \cdot z_1.$

Nonlinear Model Reduction by Generalized Moment-Matching
Variational Analysis and Linear Subsystems

Analysis of nonlinear systems by variational equation approach:

- consider input of the form $\alpha u(t)$,
- nonlinear system is assumed to be a series of homogeneous nonlinear subsystems, i.e. response should be of the form

$$x(t) = \alpha x_1(t) + \alpha^2 x_2(t) + \alpha^3 x_3(t) + \ldots.$$

- Comparison of terms $\alpha^i, i = 1, 2, \ldots$ leads to series of systems

$$\begin{aligned} E\dot{x}_1 &= A_1 x_1 + Bu, \\ E\dot{x}_2 &= A_1 x_2 + A_2 x_1 \otimes x_1 + N x_1 u, \\ E\dot{x}_3 &= A_1 x_3 + A_2 (x_1 \otimes x_2 + x_2 \otimes x_1) + N x_2 u \\ &\vdots \end{aligned}$$

- although i-th subsystem is coupled nonlinearly to preceding systems, linear systems are obtained if terms x_j, $j < i$, are interpreted as pseudo-inputs.

Nonlinear Model Reduction by Generalized Moment-Matching
Variational Analysis and Linear Subsystems

Analysis of nonlinear systems by variational equation approach:

- consider input of the form $\alpha u(t)$,
- nonlinear system is assumed to be a series of homogeneous nonlinear subsystems, i.e. response should be of the form

$$x(t) = \alpha x_1(t) + \alpha^2 x_2(t) + \alpha^3 x_3(t) + \ldots.$$

- Comparison of terms $\alpha^i, i = 1, 2, \ldots$ leads to series of systems

$$\begin{aligned} E\dot{x}_1 &= A_1 x_1 + Bu, \\ E\dot{x}_2 &= A_1 x_2 + A_2 x_1 \otimes x_1 + N x_1 u, \\ E\dot{x}_3 &= A_1 x_3 + A_2 \left(x_1 \otimes x_2 + x_2 \otimes x_1 \right) + N x_2 u \\ &\vdots \end{aligned}$$

- although i-th subsystem is coupled nonlinearly to preceding systems, linear systems are obtained if terms x_j, $j < i$, are interpreted as pseudo-inputs.

Nonlinear Model Reduction by Generalized Moment-Matching

Variational Analysis and Linear Subsystems

Analysis of nonlinear systems by variational equation approach:

- consider input of the form $\alpha u(t)$,
- nonlinear system is assumed to be a series of homogeneous nonlinear subsystems, i.e. response should be of the form

$$x(t) = \alpha x_1(t) + \alpha^2 x_2(t) + \alpha^3 x_3(t) + \ldots.$$

- Comparison of terms $\alpha^i, i = 1, 2, \ldots$ leads to series of systems

$$\begin{aligned} E\dot{x}_1 &= A_1 x_1 + Bu, \\ E\dot{x}_2 &= A_1 x_2 + A_2 x_1 \otimes x_1 + N x_1 u, \\ E\dot{x}_3 &= A_1 x_3 + A_2 (x_1 \otimes x_2 + x_2 \otimes x_1) + N x_2 u \\ &\vdots \end{aligned}$$

- although i-th subsystem is coupled nonlinearly to preceding systems, linear systems are obtained if terms x_j, $j < i$, are interpreted as pseudo-inputs.

Nonlinear Model Reduction by Generalized Moment-Matching

Variational Analysis and Linear Subsystems

Analysis of nonlinear systems by variational equation approach:

- consider input of the form $\alpha u(t)$,
- nonlinear system is assumed to be a series of homogeneous nonlinear subsystems, i.e. response should be of the form

$$x(t) = \alpha x_1(t) + \alpha^2 x_2(t) + \alpha^3 x_3(t) + \dots.$$

- Comparison of terms $\alpha^i, i = 1, 2, \dots$ leads to series of systems

$$\begin{aligned} E\dot{x}_1 &= A_1 x_1 + Bu, \\ E\dot{x}_2 &= A_1 x_2 + A_2 x_1 \otimes x_1 + N x_1 u, \\ E\dot{x}_3 &= A_1 x_3 + A_2 (x_1 \otimes x_2 + x_2 \otimes x_1) + N x_2 u \\ &\vdots \end{aligned}$$

- although i-th subsystem is coupled nonlinearly to preceding systems, linear systems are obtained if terms x_j, $j < i$, are interpreted as pseudo-inputs.

Nonlinear Model Reduction by Generalized Moment-Matching
Variational Analysis and Linear Subsystems

Analysis of nonlinear systems by variational equation approach:

- consider input of the form $\alpha u(t)$,
- nonlinear system is assumed to be a series of homogeneous nonlinear subsystems, i.e. response should be of the form

$$x(t) = \alpha x_1(t) + \alpha^2 x_2(t) + \alpha^3 x_3(t) + \dots .$$

- Comparison of terms $\alpha^i, i = 1, 2, \dots$ leads to series of systems

$$\begin{aligned} E\dot{x}_1 &= A_1 x_1 + Bu, \\ E\dot{x}_2 &= A_1 x_2 + A_2 x_1 \otimes x_1 + N x_1 u, \\ E\dot{x}_3 &= A_1 x_3 + A_2 \left(x_1 \otimes x_2 + x_2 \otimes x_1 \right) + N x_2 u \\ &\vdots \end{aligned}$$

- although i-th subsystem is coupled nonlinearly to preceding systems, linear systems are obtained if terms x_j, $j < i$, are interpreted as pseudo-inputs.

Nonlinear Model Reduction by Generalized Moment-Matching

Generalized Transfer Functions

In a similar way, a series of generalized symmetric transfer functions can be obtained via formal multivariate Laplace transforms:

Nonlinear Model Reduction by Generalized Moment-Matching
Generalized Transfer Functions

In a similar way, a series of generalized symmetric transfer functions can be obtained via formal multivariate Laplace transforms:

$$H_1(s_1) = C \underbrace{(s_1 E - A_1)^{-1} B}_{G_1(s_1)},$$

Nonlinear Model Reduction by Generalized Moment-Matching
Generalized Transfer Functions

In a similar way, a series of generalized symmetric transfer functions can be obtained via formal multivariate Laplace transforms:

$$H_1(s_1) = C\underbrace{(s_1 E - A_1)^{-1} B}_{G_1(s_1)},$$

$$\begin{aligned} H_2(s_1, s_2) = \frac{1}{2!} C \left((s_1 + s_2)E - A_1\right)^{-1} \Big[N \left(G_1(s_1) + G_1(s_2) \right) \\ + A_2 \left(G_1(s_1) \otimes G_1(s_2) + G_1(s_2) \otimes G_1(s_1) \right) \Big], \end{aligned}$$

Nonlinear Model Reduction by Generalized Moment-Matching

Generalized Transfer Functions

In a similar way, a series of generalized symmetric transfer functions can be obtained via formal multivariate Laplace transforms:

$$H_1(s_1) = C\underbrace{(s_1E - A_1)^{-1}B}_{G_1(s_1)},$$

$$\begin{aligned} H_2(s_1, s_2) = \frac{1}{2!}C\left((s_1+s_2)E - A_1\right)^{-1}&\left[N\left(G_1(s_1) + G_1(s_2)\right)\right.\\ &\left. + A_2\left(G_1(s_1)\otimes G_1(s_2) + G_1(s_2)\otimes G_1(s_1)\right)\right],\end{aligned}$$

$$\begin{aligned} H_3(s_1, s_2, s_3) = \frac{1}{3!}C\left((s_1+s_2+s_3)E - A_1\right)^{-1}&\\ &\Big[N(G_2(s_1,s_2) + G_2(s_2,s_3) + G_2(s_1,s_3))\\ &+ A_2\big(G_1(s_1)\otimes G_2(s_2,s_3) + G_1(s_2)\otimes G_2(s_1,s_3)\\ &\quad + G_1(s_3)\otimes G_2(s_1,s_3) + G_2(s_2,s_3)\otimes G_1(s_1)\\ &\quad + G_2(s_1,s_3)\otimes G_1(s_2) + G_2(s_1,s_2)\otimes G_1(s_3)\big)\Big].\end{aligned}$$

Nonlinear Model Reduction by Generalized Moment-Matching
Characterization via Multimoments

For simplicity, focus on the first two transfer functions. For $H_1(s_1)$, choosing σ and making use of the Neumann lemma leads to

$$H_1(s_1) = \sum_{i=0}^{\infty} C \underbrace{\left((A_1 - \sigma E)^{-1} E\right)^i (A_1 - \sigma E)^{-1} B \, (s_1 - \sigma)^i}_{m^i_{s_1,\sigma}}.$$

Nonlinear Model Reduction by Generalized Moment-Matching
Characterization via Multimoments

For simplicity, focus on the first two transfer functions. For $H_1(s_1)$, choosing σ and making use of the Neumann lemma leads to

$$H_1(s_1) = \sum_{i=0}^{\infty} C \underbrace{\left((A_1 - \sigma E)^{-1} E\right)^i (A_1 - \sigma E)^{-1} B \,(s_1 - \sigma)^i}_{m^i_{s_1,\sigma}}.$$

Similarly, specifying an expansion point (τ, ξ) yields

$$H_2(s_1, s_2) = \frac{1}{2} \sum_{i=0}^{\infty} C \left((A_1 - (\tau + \xi)E)^{-1} E\right)^i (A_1 - (\tau + \xi)E)^{-1} \,(s_1 + s_2 - \tau - \xi)^i \cdot$$
$$\left[A_2 \left(\sum_{j=0}^{\infty} m^j_{s_1,\tau} \otimes \sum_{k=0}^{\infty} m^k_{s_2,\xi} + \sum_{k=0}^{\infty} m^k_{s_2,\xi} \otimes \sum_{j=0}^{\infty} m^j_{s_1,\tau}\right) + N \left(\sum_{p=0}^{\infty} m^p_{s_1,\tau} + \sum_{p=0}^{\infty} m^q_{s_2,\xi}\right)\right]$$

Nonlinear Model Reduction by Generalized Moment-Matching

Constructing the Projection Matrix

Goal: $\frac{\partial}{\partial s_1^{q-1}} H_1(\sigma) = \frac{\partial}{\partial s_1^{q-1}} \hat{H}_1(\sigma), \quad \frac{\partial}{\partial s_1^l s_2^m} H_2(\sigma,\sigma) = \frac{\partial}{\partial s_1^l s_2^m} \hat{H}_2(\sigma,\sigma), \; l+m \leq q-1.$

Construct the following sequence of nested Krylov subspaces

Nonlinear Model Reduction by Generalized Moment-Matching

Constructing the Projection Matrix

Goal: $\frac{\partial}{\partial s_1^{q-1}} H_1(\sigma) = \frac{\partial}{\partial s_1^{q-1}} \hat{H}_1(\sigma), \quad \frac{\partial}{\partial s_1^l s_2^m} H_2(\sigma, \sigma) = \frac{\partial}{\partial s_1^l s_2^m} \hat{H}_2(\sigma, \sigma), \; l + m \leq q - 1.$

Construct the following sequence of nested Krylov subspaces

$$V_1 = \mathcal{K}_q \left((A_1 - \sigma E)^{-1} E, (A_1 - \sigma E)^{-1} b \right)$$

Nonlinear Model Reduction by Generalized Moment-Matching
Constructing the Projection Matrix

Goal: $\frac{\partial}{\partial s_1^{q-1}} H_1(\sigma) = \frac{\partial}{\partial s_1^{q-1}} \hat{H}_1(\sigma), \quad \frac{\partial}{\partial s_1^l s_2^m} H_2(\sigma,\sigma) = \frac{\partial}{\partial s_1^l s_2^m} \hat{H}_2(\sigma,\sigma), \; l+m \leq q-1.$
Construct the following sequence of nested Krylov subspaces

$$V_1 = \mathcal{K}_q \left((A_1 - \sigma E)^{-1} E, (A_1 - \sigma E)^{-1} b \right)$$

for $i = 1 : q$

$$V_2^i = \mathcal{K}_{q-i+1} \left((A_1 - 2\sigma E)^{-1} E, (A_1 - 2\sigma E)^{-1} N V_1(:, i) \right),$$

Nonlinear Model Reduction by Generalized Moment-Matching

Constructing the Projection Matrix

Goal: $\frac{\partial}{\partial s_1^{q-1}} H_1(\sigma) = \frac{\partial}{\partial s_1^{q-1}} \hat{H}_1(\sigma), \quad \frac{\partial}{\partial s_1^l s_2^m} H_2(\sigma,\sigma) = \frac{\partial}{\partial s_1^l s_2^m} \hat{H}_2(\sigma,\sigma), \; l+m \leq q-1.$

Construct the following sequence of nested Krylov subspaces

$$
\begin{aligned}
V_1 =& \mathcal{K}_q \left((A_1 - \sigma E)^{-1} E, (A_1 - \sigma E)^{-1} b \right) \\
&\textbf{for } i = 1 : q \\
&\quad V_2^i = \mathcal{K}_{q-i+1} \left((A_1 - 2\sigma E)^{-1} E, (A_1 - 2\sigma E)^{-1} N V_1(:, i) \right), \\
&\qquad \textbf{for } j = 1 : \min(q-i+1, i) \\
&\qquad\quad V_3^{i,j} = \mathcal{K}_{q-i-j+2} \left((A_1 - 2\sigma E)^{-1} E, (A_1 - 2\sigma E)^{-1} A_2 V_1(:, i) \otimes V_1(:, j) \right),
\end{aligned}
$$

$V_1(:, i)$ denoting the i-*th* column of V_1.

Nonlinear Model Reduction by Generalized Moment-Matching
Constructing the Projection Matrix

Goal: $\frac{\partial}{\partial s_1^{q-1}} H_1(\sigma) = \frac{\partial}{\partial s_1^{q-1}} \hat{H}_1(\sigma)$, $\frac{\partial}{\partial s_1^l s_2^m} H_2(\sigma,\sigma) = \frac{\partial}{\partial s_1^l s_2^m} \hat{H}_2(\sigma,\sigma)$, $l + m \leq q-1$.
Construct the following sequence of nested Krylov subspaces

$$V_1 = \mathcal{K}_q\left((A_1 - \sigma E)^{-1} E, (A_1 - \sigma E)^{-1} b\right)$$

for $i = 1 : q$

$$V_2^i = \mathcal{K}_{q-i+1}\left((A_1 - 2\sigma E)^{-1} E, (A_1 - 2\sigma E)^{-1} N V_1(:,i)\right),$$

for $j = 1 : \min(q-i+1, i)$

$$V_3^{i,j} = \mathcal{K}_{q-i-j+2}\left((A_1 - 2\sigma E)^{-1} E, (A_1 - 2\sigma E)^{-1} A_2 V_1(:,i) \otimes V_1(:,j)\right),$$

$V_1(:,i)$ denoting the i-*th* column of V_1. Set $\mathcal{V} = \operatorname{orth}[V_1, V_2^i, V_3^{i,j}]$ and construct $\hat{\Sigma}$ by the Galerkin-Projection $\mathcal{P} = \mathcal{V}\mathcal{V}^T$:

$$\hat{A}_1 = \mathcal{V}^T A_1 \mathcal{V} \in \mathbb{R}^{\hat{n}\times\hat{n}}, \quad \hat{A}_2 = \mathcal{V}^T A_2 (\mathcal{V} \otimes \mathcal{V}) \in \mathbb{R}^{\hat{n}\times\hat{n}^2},$$

$$\hat{N} = \mathcal{V}^T N \mathcal{V} \in \mathbb{R}^{\hat{n}\times\hat{n}}, \quad \hat{b} = \mathcal{V}^T b \in \mathbb{R}^{\hat{n}}, \quad \hat{c}^T = c^T \mathcal{V} \in \mathbb{R}^{\hat{n}}.$$

Nonlinear Model Reduction by Generalized Moment-Matching

Two-Sided Projection Methods

- Similarly to the linear case, one can exploit duality concepts, in order to construct two-sided (Petrov-Galerkin) projection methods.
- Construction the dual Krylov subspaces efficiently requires a bit of tensor calculus.

Nonlinear Model Reduction by Generalized Moment-Matching

Two-Sided Projection Methods

- Similarly to the linear case, one can exploit duality concepts, in order to construct two-sided (Petrov-Galerkin) projection methods.
- Construction the dual Krylov subspaces efficiently requires a bit of tensor calculus.

Nonlinear Model Reduction by Generalized Moment-Matching

Two-Sided Projection Methods

Theorem [B./Breiten 2012]

- $\Sigma = (E, A_1, A_2, N, b, c)$ original QBDAE system.
- Reduced system by Petrov-Galerkin projection $\mathcal{P} = \mathcal{V}\mathcal{W}^T$ with

$$V_1 = \mathcal{K}_{q_1}(E, A_1, b, \sigma), \quad W_1 = \mathcal{K}_{q_1}\left(E^T, A_1^T, c, 2\sigma\right)$$

$$\begin{aligned}
&\textbf{for } i = 1 : q_2 \\
&\quad V_2 = \mathcal{K}_{q_2-i+1}(E, A_1, NV_1(:, i), 2\sigma) \\
&\quad W_2 = \mathcal{K}_{q_2-i+1}\left(E^T, A_1^T, N^T W_1(:, i), \sigma\right) \\
&\quad\quad \textbf{for } j = 1 : \min(q_2 - i + 1, i) \\
&\quad\quad\quad V_3 = \mathcal{K}_{q_2-i-j+2}(E, A_1, A_2 V_1(:, i) \otimes V_1(:, j), 2\sigma) \\
&\quad\quad\quad W_3 = \mathcal{K}_{q_2-i-j+2}\left(E^T, A_1^T, \mathcal{A}^{(2)} V_1(:, i) \otimes W_1(:, j), \sigma\right).
\end{aligned}$$

Then, it holds:

$$\frac{\partial^i H_1}{\partial s_1^i}(\sigma) = \frac{\partial^i \hat{H}_1}{\partial s_1^i}(\sigma), \quad \frac{\partial^i H_1}{\partial s_1^i}(2\sigma) = \frac{\partial^i \hat{H}_1}{\partial s_1^i}(2\sigma), \quad i = 0, \ldots, q_1 - 1,$$

$$\frac{\partial^{i+j}}{\partial s_1^i s_2^j} H_2(\sigma, \sigma) = \frac{\partial^{i+j}}{\partial s_1^i s_2^j} \hat{H}_2(\sigma, \sigma), \quad i + j \le 2q_2 - 1.$$

Numerical Examples
Two-Dimensional Burgers Equation

- 2D-Burgers equation on $\underbrace{(0,1)\times(0,1)}_{:=\Omega}\times[0,T]$

$$u_t = -\left(u \cdot \nabla\right) u + \nu \Delta u$$

with $u(x,y,t) \in \mathbb{R}^2$ describing the motion of a compressible fluid.

Numerical Examples
Two-Dimensional Burgers Equation

- 2D-Burgers equation on $\underbrace{(0,1)\times(0,1)}_{:=\Omega}\times[0,T]$

$$u_t = -(u\cdot\nabla)\,u + \nu\Delta u$$

with $u(x,y,t)\in\mathbb{R}^2$ describing the motion of a compressible fluid.

- Consider initial and boundary conditions

$$\begin{aligned}
&u_x(x,y,0)=\frac{\sqrt{2}}{2}, && u_y(x,y,0)=\frac{\sqrt{2}}{2}, && \text{for } (x,y)\in\Omega_1 := (0,0.5],\\
&u_x(x,y,0)=0, && u_y(x,y,0)=0, && \text{for } (x,y)\in\Omega\backslash\Omega_1,\\
&u_x=0, && u_y=0, && \text{for } (x,y)\in\partial\Omega.
\end{aligned}$$

Numerical Examples
Two-Dimensional Burgers Equation

- 2D-Burgers equation on $\underbrace{(0,1)\times(0,1)}_{:=\Omega}\times[0,T]$

$$u_t = -\,(u\cdot\nabla)\,u + \nu\Delta u$$

  with $u(x,y,t)\in\mathbb{R}^2$ describing the motion of a compressible fluid.
- Consider initial and boundary conditions

$$\begin{aligned}
&u_x(x,y,0) = \frac{\sqrt{2}}{2}, && u_y(x,y,0) = \frac{\sqrt{2}}{2}, && \text{for } (x,y)\in\Omega_1 := (0,0.5],\\
&u_x(x,y,0) = 0, && u_y(x,y,0) = 0, && \text{for } (x,y)\in\Omega\backslash\Omega_1,\\
&u_x = 0, && u_y = 0, && \text{for } (x,y)\in\partial\Omega.
\end{aligned}$$

- Spatial discretization $\rightsquigarrow$ QBDAE system with nonzero I.C. and $N=0 \rightsquigarrow$ reformulate as system with zero I.C. and constant input.

Numerical Examples
Two-Dimensional Burgers Equation

- 2D-Burgers equation on $\underbrace{(0,1)\times(0,1)}_{:=\Omega}\times[0,T]$

$$u_t = -\left(u\cdot\nabla\right)u + \nu\Delta u$$

  with $u(x,y,t)\in\mathbb{R}^2$ describing the motion of a compressible fluid.

- Consider initial and boundary conditions

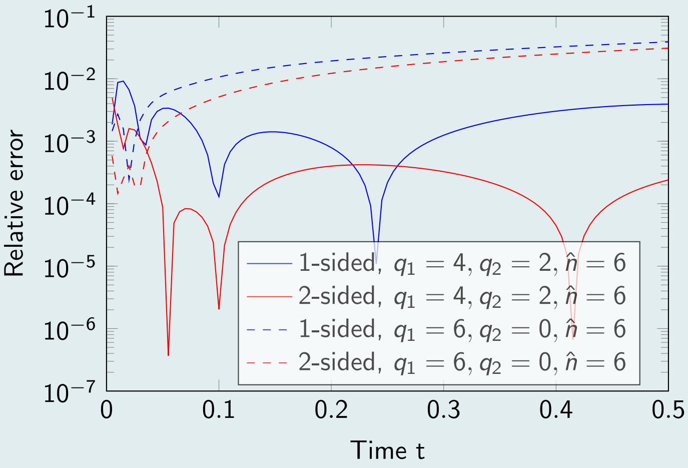
$$\begin{aligned}
&u_x(x,y,0)=\frac{\sqrt{2}}{2}, && u_y(x,y,0)=\frac{\sqrt{2}}{2}, && \text{for } (x,y)\in\Omega_1:=(0,0.5],\\
&u_x(x,y,0)=0, && u_y(x,y,0)=0, && \text{for } (x,y)\in\Omega\backslash\Omega_1,\\
&u_x=0, && u_y=0, && \text{for } (x,y)\in\partial\Omega.
\end{aligned}$$

- Spatial discretization $\rightsquigarrow$ QBDAE system with nonzero I.C. and $N=0\rightsquigarrow$ reformulate as system with zero I.C. and constant input.
- Output C chosen to be average x-velocity.

Numerical Examples

Two-Dimensional Burgers Equation

Comparison of relative time-domain error for $n = 1600$



Numerical Examples
Two-Dimensional Burgers Equation

- 2D-Burgers equation on $\underbrace{(0,1)\times(0,1)}_{:=\Omega}\times[0,T]$

$$u_t = -(u\cdot\nabla)\,u + \nu\Delta u$$

with $u(x,y,t)\in\mathbb{R}^2$ describing the motion of a compressible fluid.

- Now consider initial and boundary conditions

$$\begin{aligned}
&u_x(x,y,0)=0, && u_y(x,y,0)=0, && \text{for } x,y\in\Omega,\\
&u_x=\cos(\pi t), && u_y=\cos(2\pi t), && \text{for } (x,y)\in\{0,1\}\times(0,1),\\
&u_x=\sin(\pi t), && u_y=\sin(2\pi t), && \text{for } (x,y)\in(0,1)\times\{0,1\}.
\end{aligned}$$

Numerical Examples
Two-Dimensional Burgers Equation

- 2D-Burgers equation on $\underbrace{(0,1)\times(0,1)}_{:=\Omega}\times[0,T]$

$$u_t = -\left(u\cdot\nabla\right)u + \nu\Delta u$$

with $u(x,y,t)\in\mathbb{R}^2$ describing the motion of a compressible fluid.

- Now consider initial and boundary conditions

$$\begin{aligned}
u_x(x,y,0) &= 0, & u_y(x,y,0) &= 0, & &\text{for } \ x,y\in\Omega,\\
u_x &= \cos(\pi t), & u_y &= \cos(2\pi t), & &\text{for } \ (x,y)\in\{0,1\}\times(0,1),\\
u_x &= \sin(\pi t), & u_y &= \sin(2\pi t), & &\text{for } \ (x,y)\in(0,1)\times\{0,1\}.
\end{aligned}$$

- Spatial discretization $\rightsquigarrow$ QBDAE system with zero I.C. and 4 inputs $B\in\mathbb{R}^{n\times 4}$, N_1,N_2,N_3,N_4, ROM with $q_1=5, q_2=2, \sigma=0, \hat{n}=52$.

Numerical Examples
Two-Dimensional Burgers Equation

- 2D-Burgers equation on $\underbrace{(0,1)\times(0,1)}_{:=\Omega}\times[0,T]$

$$u_t = -\,(u\cdot\nabla)\,u + \nu\Delta u$$

with $u(x,y,t)\in\mathbb{R}^2$ describing the motion of a compressible fluid.

- Now consider initial and boundary conditions

$$\begin{array}{lll} u_x(x,y,0)=0, & u_y(x,y,0)=0, & \text{for } \ x,y\in\Omega, \\ u_x=\cos(\pi t), & u_y=\cos(2\pi t), & \text{for } \ (x,y)\in\{0,1\}\times(0,1), \\ u_x=\sin(\pi t), & u_y=\sin(2\pi t), & \text{for } \ (x,y)\in(0,1)\times\{0,1\}. \end{array}$$

- Spatial discretization $\rightsquigarrow$ QBDAE system with zero I.C. and 4 inputs $B\in\mathbb{R}^{n\times 4}$, N_1, N_2, N_3, N_4, ROM with $q_1=5, q_2=2, \sigma=0, \hat{n}=52$.
- State reconstruction by reduced model $x\approx V\hat{x}$, max. rel. err $<3\%$.

Numerical Examples
The Chafee-Infante equation

- Consider PDE with a cubic nonlinearity:

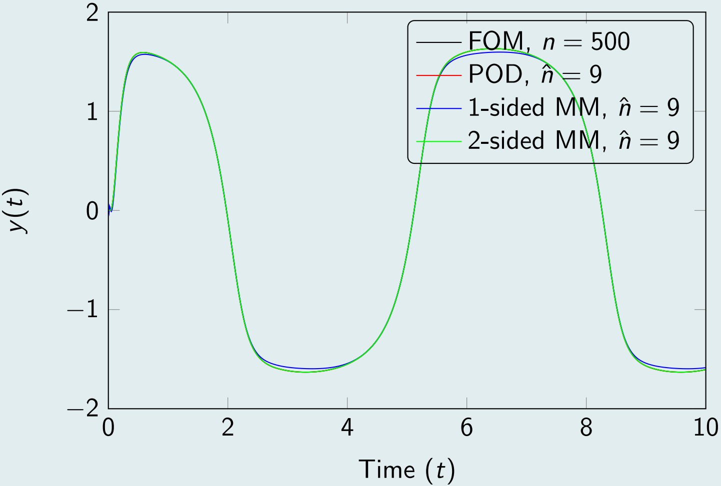
$$
\begin{aligned}
v_t + v^3 &= v_{xx} + v, && \text{in } (0,1) \times (0,T),\\
v(0,\cdot) &= u(t), && \text{in } (0,T),\\
v_x(1,\cdot) &= 0, && \text{in } (0,T),\\
v(x,0) &= v_0(x), && \text{in } (0,1)
\end{aligned}
$$

- original state dimension $n = 500$, QBDAE dimension $N = 2 \cdot 500$, reduced QBDAE dimension $r = 9$

Numerical Examples

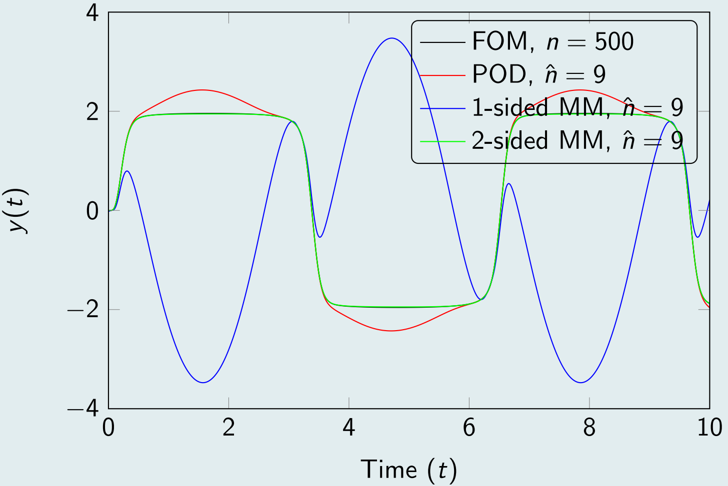
The Chafee-Infante equation

Comparison between moment-matching and POD ($u(t) = 5\cos(t)$)



Numerical Examples
The Chafee-Infante equation

Comparison between moment-matching and POD ($u(t) = 50\sin(t)$)



Numerical Examples
The FitzHugh-Nagumo System

- FitzHugh-Nagumo system modeling a neuron

[Chaturantabut, Sorensen '09]

$$\begin{aligned} \epsilon v_t(x,t) &= \epsilon^2 v_{xx}(x,t) + f(v(x,t)) - w(x,t) + g, \\ w_t(x,t) &= hv(x,t) - \gamma w(x,t) + g, \end{aligned}$$

with $f(v) = v(v-0.1)(1-v)$ and initial and boundary conditions

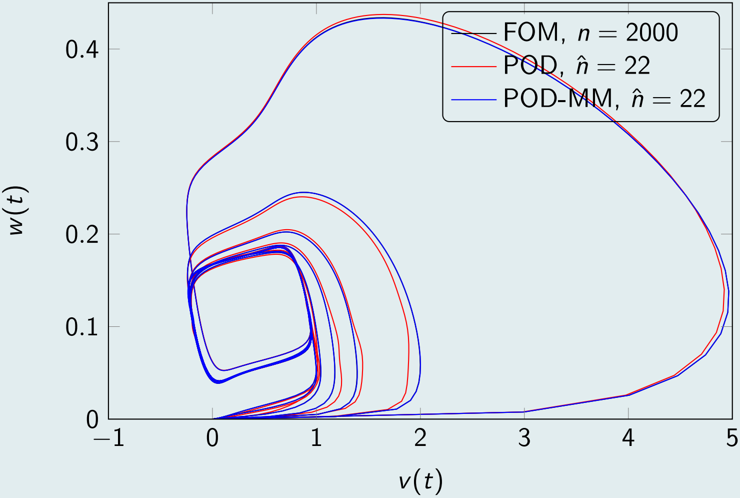
$$\begin{aligned} v(x,0) &= 0, & w(x,0) &= 0, & x &\in [0,1], \\ v_x(0,t) &= -i_0(t), & v_x(1,t) &= 0, & t &\geq 0, \end{aligned}$$

where
$\epsilon = 0.015,\ h = 0.5,\ \gamma = 2,\ g = 0.05,\ i_0(t) = 5 \cdot 10^4 t^3 \exp(-15t)$

- original state dimension $n = 2 \cdot 1000$, QBDAE dimension $N = 3 \cdot 1000$, reduced QBDAE dimension $r = 20$

Numerical Examples
The FitzHugh-Nagumo System

POD via moment-matching (varying input)



Topics Not Covered

Linear Systems:

- Balanced residualization (singular perturbation approximation), yields $G(0) = \hat{G}(0)$.
- Balancing-related methods.
- Special methods for second-order (mechanical) systems.
- Extensions to bilinear and stochastic systems.
- MOR methods for discrete-time systems.
- Extensions to descriptor systems $E\dot{x} = Ax + Bu$, E singular.
- Parametric model reduction:

$$\dot{x} = A(p)x + B(p)u, \quad y = C(p)x,$$

  where $p \in \mathbb{R}^d$ is a free parameter vector; parameters should be preserved in the reduced-order model.

Nonlinear Systems:

- Other MOR techniques like POD, RB, Empirical Gramians.
- Simulation-free methods for parametric systems is widely open!

Further Reading

1. G. Obinata and B.D.O. Anderson.
   Model Reduction for Control System Design.
   Springer-Verlag, London, UK, 2001.
2. R. Freund.
   Model reduction methods based on Krylov subspaces.
   Acta Numerica, 12:267–319, 2003.
3. P. Benner, E.S. Quintana-Ortí, and G. Quintana-Ortí.
   State-space truncation methods for parallel model reduction of large-scale systems.
   Parallel Comput., 29:1701–1722, 2003.
4. P. Benner.
   Solving large-scale control problems.
   IEEE Control Systems Magazine, 24(1):44–59, 2004.
5. P. Benner, V. Mehrmann, and D. Sorensen (editors).
   Dimension Reduction of Large-Scale Systems.
   Lecture Notes in Computational Science and Engineering, Vol. 45,
   Springer-Verlag, Berlin/Heidelberg, Germany, 2005.
6. A.C. Antoulas.
   Lectures on the Approximation of Large-Scale Dynamical Systems.
   SIAM Publications, Philadelphia, PA, 2005.
7. P. Benner, R. Freund, D. Sorensen, and A. Varga (editors).
   Special issue on *Order Reduction of Large-Scale Systems*.
   Linear Algebra Appl., June 2006.
8. W.H.A. Schilders, H.A. van der Vorst, and J. Rommes (editors).
   Model Order Reduction: Theory, Research Aspects and Applications.
   Mathematics in Industry, Vol. 13,
   Springer-Verlag, Berlin/Heidelberg, 2008.
9. P. Benner, J. ter Maten, and M. Hinze (editors).
   Model Reduction for Circuit Simulation.
   Lecture Notes in Electrical Engineering, Vol. 74,
   Springer-Verlag, Dordrecht, 2011.