Combined Applied Math & PDEs Seminar
UC Davis
February 11, 2014

MODEL REDUCTION FOR NONLINEAR SYSTEMS WITHOUT SNAPSHOTS

Peter Benner

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

joint work with
Tobias Breiten (University of Graz, Austria)

Overview

Introduction

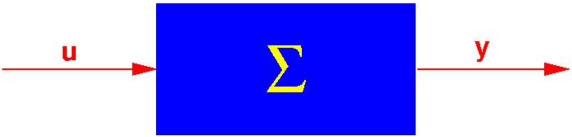
Model Reduction for Dynamical Systems

Dynamical Systems

$$\Sigma : \left\{ \begin{array}{rcl} \dot{x}(t) & = & f(t, x(t), u(t)), \quad x(t_0) = x_0, \\ y(t) & = & g(t, x(t), u(t)) \end{array} \right.$$

with

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

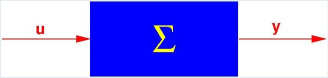


Model Reduction for Dynamical Systems

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 $y(t) \in \mathbb{R}^p$.



Goal:

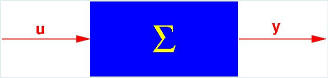
$\|y - \hat{y}\| <$ tolerance $\cdot \|u\|$ for all admissible input signals.

Model Reduction for Dynamical Systems

Original System

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

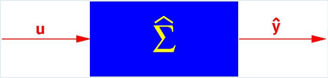
- states $x(t) \in \mathbb{R}^n$,
- inputs $u(t) \in \mathbb{R}^m$,
- outputs $y(t) \in \mathbb{R}^p$.



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 $\hat{y}(t) \in \mathbb{R}^p$.



Goal:

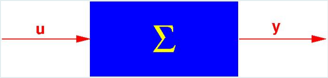
$\|y - \hat{y}\| <$ tolerance $\cdot \|u\|$ for all admissible input signals.

Model Reduction for Dynamical Systems

Original System

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

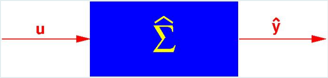
- states $x(t) \in \mathbb{R}^n$,
- inputs $u(t) \in \mathbb{R}^m$,
- outputs $y(t) \in \mathbb{R}^p$.



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 $\hat{y}(t) \in \mathbb{R}^p$.



Goal:

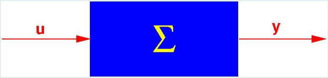
$\|y - \hat{y}\| <$ tolerance $\cdot \|u\|$ for all admissible input signals.

Model Reduction for Dynamical Systems

Original System

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

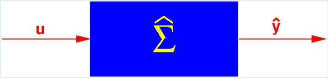
- states $x(t) \in \mathbb{R}^n$,
- inputs $u(t) \in \mathbb{R}^m$,
- outputs $y(t) \in \mathbb{R}^p$.



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 $\hat{y}(t) \in \mathbb{R}^p$.



Goal:

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

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

Application Areas

Historically:

- structural dynamics (1960ies: modal truncation, Guyan/Craig-Bampton reduction),
- computational fluid dynamics/CFD (1970/80ies: proper orthogonal decomposition/POD),
- control design (1980ies: balanced truncation),
- microelectronics/circuit simulation (1990ies: moment matching/Padé approximation **[Freund!]**).

Application Areas

Historically:

- structural dynamics (1960ies: modal truncation, Guyan/Craig-Bampton reduction),
- computational fluid dynamics/CFD (1970/80ies: proper orthogonal decomposition/POD),
- control design (1980ies: balanced truncation),
- microelectronics/circuit simulation (1990ies: moment matching/Padé approximation **[Freund!]**).

Now: many other disciplines in **computational sciences and engineering** like

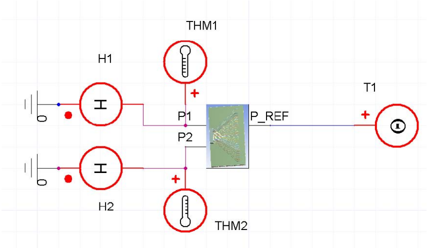
- design of MEMS/NEMS (micro/nano-electrical-mechanical systems),
- computational electromagnetics,
- computational neurosciences,
- computational (*insert your favorite discipline here*),
- chemical process engineering,
- biomedical engineering,
- ...

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: $n = 270.593$, $m = q = 2$ ⇒ Computing time (on Intel Xeon dualcore 3GHz, 1 Thread):
  - Main computational cost for set-up data ≈ 22*min*.
  - 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 / ≥ 450 excluding reduced model generation!

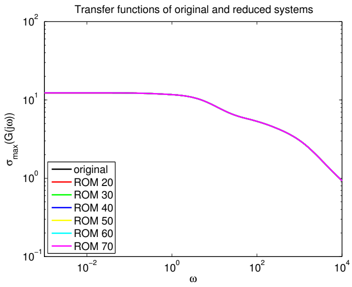
Motivating Examples

Electro-Thermic Simulation of Integrated Circuit (IC)

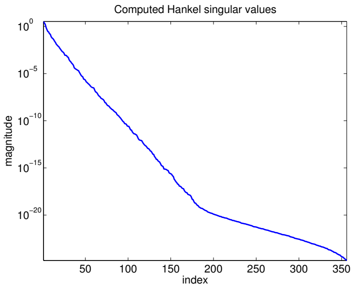
[Source: Evgenii Rudnyi, CADFEM GmbH]

- 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



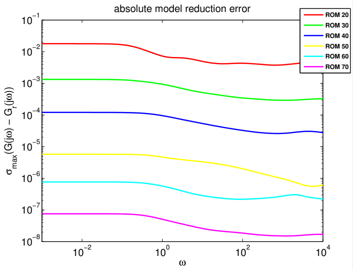
Motivating Examples

Electro-Thermic Simulation of Integrated Circuit (IC)

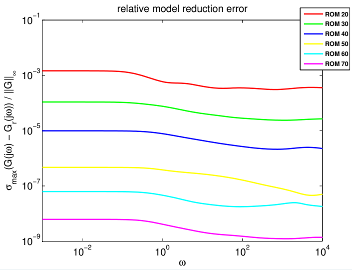
[Source: Evgenii Rudnyi, CADFEM GmbH]

- 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

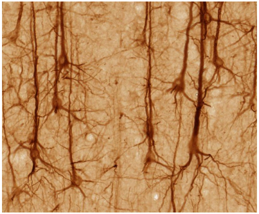
- Simple model for neuron (de-)activation [Chaturantabut/Sorensen 2009]

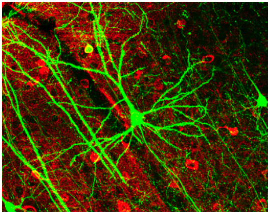
$$\epsilon v_t(x,t) = \epsilon^2 v_{xx}(x,t) + f(v(x,t)) - w(x,t) + g,$$
$$w_t(x,t) = hv(x,t) - \gamma w(x,t) + g,$$

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

$$v(x,0) = 0, \qquad w(x,0) = 0, \qquad x \in [0,1]$$
$$v_x(0,t) = -i_0(t), \qquad v_x(1,t) = 0, \qquad t \geq 0,$$

where $\epsilon = 0.015, h = 0.5, \gamma = 2, g = 0.05, i_0(t) = 50000t^3\exp(-15t)$.





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]

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

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

$$v(x,0) = 0, \qquad w(x,0) = 0, \qquad x \in [0,1]$$
$$v_x(0,t) = -i_0(t), \qquad v_x(1,t) = 0, \qquad t \geq 0,$$

where $\epsilon = 0.015, h = 0.5, \gamma = 2, g = 0.05, i_0(t) = 50000t^3 \exp(-15t)$.

- 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

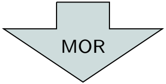
Introduction

Nonlinear Model Reduction

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

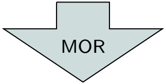
Introduction

Nonlinear Model Reduction

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 $\hat{y} \approx y \in \mathbb{R}$, $\hat{n} \ll n$.

Introduction

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'!

Introduction

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.

Introduction

Linear System Norms

First consider linear systems, i.e. $f(x) = Ax \rightsquigarrow$

$$\dot{x}(t) = Ax(t) + Bu(t), \; y(t) = Cx(t) \quad \simeq \quad Y(s) = C(sI - A)^{-1}BU(s).$$

Two common system norms for measuring approximation quality:

- $\mathcal{H}_2$-norm, $\|\Sigma\|_{\mathcal{H}_2} = \left(\frac{1}{2\pi} \int_0^{2\pi} \operatorname{tr}\left(H^*(-i\omega) H(i\omega) \right) d\omega \right)^{\frac{1}{2}}$,
- $\mathcal{H}_\infty$-norm, $\|\Sigma\|_{\mathcal{H}_\infty} = \sup_{\omega \in \mathbb{R}} \sigma_{max}\left(H(i\omega) \right)$,

where

$$H(s) = C\left(sI - A\right)^{-1}B$$

denotes the corresponding transfer function of the linear system.

Introduction

Linear System Norms

First consider linear systems, i.e. $f(x) = Ax \rightsquigarrow$

$$\dot{x}(t) = Ax(t) + Bu(t), \; y(t) = Cx(t) \quad \simeq \quad Y(s) = C(sI - A)^{-1}BU(s).$$

Two common system norms for measuring approximation quality:

- $\mathcal{H}_2$-norm, $\|\Sigma\|_{\mathcal{H}_2} = \left(\frac{1}{2\pi} \int_0^{2\pi} \mathrm{tr}\left(H^*(-i\omega)H(i\omega) \right) d\omega \right)^{\frac{1}{2}}$,
- $\mathcal{H}_\infty$-norm, $\|\Sigma\|_{\mathcal{H}_\infty} = \sup_{\omega \in \mathbb{R}} \sigma_{max}\left(H(i\omega) \right)$,

where

$$H(s) = C\left(sI - A\right)^{-1} B$$

denotes the corresponding transfer function of the linear system.

We focus on the first one $\rightsquigarrow$ interpolation-based model reduction approaches.

Introduction

Error system and $\mathcal{H}_2$-Optimality [Meier/Luenberger '67]

In order to find an $\mathcal{H}_2$-optimal reduced system, consider the error system $H(s) - \hat{H}(s)$ which can be realized by

$$A^{err} = \begin{bmatrix} A & 0 \\ 0 & \hat{A} \end{bmatrix}, \quad B^{err} = \begin{bmatrix} B \\ \hat{B} \end{bmatrix}, \quad C^{err} = \begin{bmatrix} C & -\hat{C} \end{bmatrix}.$$

Introduction

Error system and $\mathcal{H}_2$-Optimality **[Meier/Luenberger '67]**

In order to find an $\mathcal{H}_2$-optimal reduced system, consider the error system $H(s) - \hat{H}(s)$ which can be realized by

$$A^{err} = \begin{bmatrix} A & 0 \\ 0 & \hat{A} \end{bmatrix}, \quad B^{err} = \begin{bmatrix} B \\ \hat{B} \end{bmatrix}, \quad C^{err} = \begin{bmatrix} C & -\hat{C} \end{bmatrix}.$$

Assuming a coordinate system in which $\hat{A}$ is diagonal and taking derivatives of

$$\|H(\,.\,) - \hat{H}(\,.\,)\|^2_{\mathcal{H}_2}$$

with respect to free parameters in $\Lambda(\hat{A}), \hat{B}, \hat{C} \rightsquigarrow$ **first-order necessary $\mathcal{H}_2$-optimality conditions (SISO)**

$$H(-\hat{\lambda}_i) = \hat{H}(-\hat{\lambda}_i),$$
$$H'(-\hat{\lambda}_i) = \hat{H}'(-\hat{\lambda}_i),$$

where $\hat{\lambda}_i$ are the poles of the reduced system $\hat{\Sigma}$.

Introduction

Error system and $\mathcal{H}_2$-Optimality [Meier/Luenberger '67]

In order to find an $\mathcal{H}_2$-optimal reduced system, consider the error system $H(s) - \hat{H}(s)$ which can be realized by

$$A^{err} = \begin{bmatrix} A & 0 \\ 0 & \hat{A} \end{bmatrix}, \quad B^{err} = \begin{bmatrix} B \\ \hat{B} \end{bmatrix}, \quad C^{err} = \begin{bmatrix} C & -\hat{C} \end{bmatrix}.$$

First-order necessary $\mathcal{H}_2$-optimality conditions (MIMO):

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

where $\hat{A} = R\hat{\Lambda}R^{-T}$ is the spectral decomposition of the reduced system and $\tilde{B} = \hat{B}^T R^{-T}$, $\tilde{C} = \hat{C}R$.

Introduction

Error system and $\mathcal{H}_2$-Optimality **[Meier/Luenberger '67]**

In order to find an $\mathcal{H}_2$-optimal reduced system, consider the error system $H(s) - \hat{H}(s)$ which can be realized by

$$A^{err} = \begin{bmatrix} A & 0 \\ 0 & \hat{A} \end{bmatrix}, \quad B^{err} = \begin{bmatrix} B \\ \hat{B} \end{bmatrix}, \quad C^{err} = \begin{bmatrix} C & -\hat{C} \end{bmatrix}.$$

First-order necessary $\mathcal{H}_2$-optimality conditions (MIMO):

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

$$\Leftrightarrow \ \operatorname{vec}(I_p)^T \left(e_j e_i^T \otimes C\right) \left(-\hat{\Lambda} \otimes I_n - I_{\hat{n}} \otimes A\right)^{-1} \left(\tilde{B}^T \otimes B\right) \operatorname{vec}(I_m)$$

$$= \operatorname{vec}(I_p)^T \left(e_j e_i^T \otimes \hat{C}\right) \left(-\hat{\Lambda} \otimes I_{\hat{n}} - I_{\hat{n}} \otimes \hat{A}\right)^{-1} \left(\tilde{B}^T \otimes \hat{B}\right) \operatorname{vec}(I_m),$$

for $i = 1, \ldots, \hat{n}$ and $j = 1, \ldots, p$.

Introduction

Interpolation of the Transfer Function [Grimme '97]

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

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

Introduction

Interpolation of the Transfer Function [Grimme '97]

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

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

where V and W are given as

$$V = \left[(\sigma_1 I - A)^{-1}B, \ldots, (\sigma_r I - A)^{-1}B\right],$$
$$W = \left[(\sigma_1 I - A^T)^{-1}C^T, \ldots, (\sigma_r I - A^T)^{-1}C^T\right].$$

Introduction

Interpolation of the Transfer Function [Grimme '97]

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

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

where V and W are given as

$$V = \left[(\sigma_1 I - A)^{-1}B, \ldots, (\sigma_r I - A)^{-1}B\right],$$
$$W = \left[(\sigma_1 I - A^T)^{-1}C^T, \ldots, (\sigma_r I - A^T)^{-1}C^T\right].$$

Then

$$H(\sigma_i) = \hat{H}(\sigma_i) \quad \text{and} \quad H'(\sigma_i) = \hat{H}'(\sigma_i),$$

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

Introduction

Interpolation of the Transfer Function [Grimme '97]

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

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

where V and W are given as

$$V = \left[(\sigma_1 I - A)^{-1} B, \ldots, (\sigma_r I - A)^{-1} B\right],$$
$$W = \left[(\sigma_1 I - A^T)^{-1} C^T, \ldots, (\sigma_r I - A^T)^{-1} C^T\right].$$

Then

$$H(\sigma_i) = \hat{H}(\sigma_i) \quad \text{and} \quad H'(\sigma_i) = \hat{H}'(\sigma_i),$$

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

Starting with an initial guess for $\hat{\Lambda}$ and setting $\sigma_i \equiv -\hat{\lambda}_i \rightsquigarrow$ iterative algorithms (IRKA/MIRIAm) that yield $\mathcal{H}_2$-optimal models.

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

$\mathcal{H}_2$-Model Reduction for Bilinear Systems

Bilinear Control Systems

Now consider $\dot{x} = Ax + g(x, u)$ with

$$g(x, u) = Bu + \left[N_1, \ldots, N_m\right] (I_m \otimes x)\, u,$$

i.e. bilinear control systems:

$$\Sigma : \quad \begin{cases} \dot{x}(t) = Ax(t) + \sum_{i=1}^{m} N_i x(t) u_i(t) + Bu(t), \\ y(t) = Cx(t), \quad x(0) = x_0, \end{cases}$$

where $A, N_i \in \mathbb{R}^{n\times n},\ B \in \mathbb{R}^{n\times m},\ C \in \mathbb{R}^{p\times n}$.

$\mathcal{H}_2$-Model Reduction for Bilinear Systems

Bilinear Control Systems

Now consider $\dot{x} = Ax + g(x, u)$ with

$$g(x, u) = Bu + \left[N_1, \ldots, N_m\right] (I_m \otimes x)\, u,$$

i.e. bilinear control systems:

$$\Sigma : \quad \begin{cases} \dot{x}(t) = Ax(t) + \sum_{i=1}^{m} N_i x(t) u_i(t) + Bu(t), \\ y(t) = Cx(t), \quad x(0) = x_0, \end{cases}$$

where $A, N_i \in \mathbb{R}^{n\times n},\ B \in \mathbb{R}^{n\times m},\ C \in \mathbb{R}^{p\times n}$.

- Approximation of weakly nonlinear systems $\rightsquigarrow$ Carleman linearization **[Krener!]**.
- A lot of linear concepts can be extended, e.g. moment matching **[Bai!]**, Gramians, Lyapunov equations, ...
- An equivalent structure arises for some stochastic control systems.

$\mathcal{H}_2$-Model Reduction for Bilinear Systems

Some Basic Facts

Output Characterization (SISO): Volterra series

$$y(t) = \sum_{k=1}^{\infty} \int_0^t \int_0^{t_1} \ldots \int_0^{t_{k-1}} K(t_1, \ldots, t_k) u(t - t_1 - \ldots - t_k) \cdots u(t - t_k) dt_k \cdots dt_1,$$

with kernels $K(t_1, \ldots, t_k) = Ce^{At_k} N_1 \cdots e^{At_2} N_1 e^{At_1} B$.

$\mathcal{H}_2$-Model Reduction for Bilinear Systems

Some Basic Facts

Output Characterization (SISO): Volterra series

$$y(t) = \sum_{k=1}^{\infty} \int_0^t \int_0^{t_1} \dots \int_0^{t_{k-1}} K(t_1, \dots, t_k) u(t - t_1 - \dots - t_k) \cdots u(t - t_k) dt_k \cdots dt_1,$$

with kernels $K(t_1, \dots, t_k) = Ce^{At_k} N_1 \cdots e^{At_2} N_1 e^{At_1} B$.

Multivariate Laplace-transform:

$$H_k(s_1, \dots, s_k) = C(s_k I - A)^{-1} N_1 \cdots (s_2 I - A)^{-1} N_1 (s_1 I - A)^{-1} B.$$

$\mathcal{H}_2$-Model Reduction for Bilinear Systems

Some Basic Facts

Output Characterization (SISO): Volterra series

$$y(t) = \sum_{k=1}^{\infty} \int_0^t \int_0^{t_1} \dots \int_0^{t_{k-1}} K(t_1, \dots, t_k) u(t-t_1-\dots-t_k) \cdots u(t-t_k) dt_k \cdots dt_1,$$

with kernels $K(t_1, \dots, t_k) = Ce^{At_k} N_1 \cdots e^{At_2} N_1 e^{At_1} B$.

Multivariate Laplace-transform:

$$H_k(s_1, \dots, s_k) = C(s_k I - A)^{-1} N_1 \cdots (s_2 I - A)^{-1} N_1 (s_1 I - A)^{-1} B.$$

Bilinear $\mathcal{H}_2$-norm:

$$||\Sigma||_{\mathcal{H}_2} := \left(\operatorname{tr} \left(\sum_{k=1}^{\infty} \int_{-\infty}^{\infty} \dots \int_{-\infty}^{\infty} \frac{1}{(2\pi)^k} \overline{H_k(i\omega_1, \dots, i\omega_k)} H_k^T(i\omega_1, \dots, i\omega_k) \right) \right)^{\frac{1}{2}}.$$

[Zhang/Lam. '02]

$\mathcal{H}_2$-Model Reduction for Bilinear Systems

$\mathcal{H}_2$-Norm Computation

Lemma [B./Breiten '11]

Let Σ denote a bilinear system. Then, the $\mathcal{H}_2$-norm is given as:

$$||\Sigma||^2_{\mathcal{H}_2} = (\mathrm{vec}(I_p))^T (C \otimes C) \left(-A \otimes I - I \otimes A - \sum_{i=1}^{m} N_i \otimes N_i \right)^{-1} (B \otimes B)\,\mathrm{vec}(I_m).$$

Error System

In order to find an $\mathcal{H}_2$-optimal reduced system, define the error system $\Sigma^{err} := \Sigma - \hat{\Sigma}$ as follows:

$$A^{err} = \begin{bmatrix} A & 0 \\ 0 & \hat{A} \end{bmatrix}, \quad N_i^{err} = \begin{bmatrix} N_i & 0 \\ 0 & \hat{N}_i \end{bmatrix}, \quad B^{err} = \begin{bmatrix} B \\ \hat{B} \end{bmatrix}, \quad C^{err} = \begin{bmatrix} C & -\hat{C} \end{bmatrix}.$$

$\mathcal{H}_2$-Model Reduction

$\mathcal{H}_2$-Optimality Conditions

Let us assume $\hat{\Sigma}$ is given by its eigenvalue decomposition:

$$\hat{A} = R\Lambda R^{-1}, \quad \tilde{N}_i = R^{-1}\hat{N}_i R, \quad \tilde{B} = R^{-1}\hat{B}, \quad \tilde{C} = \hat{C}R.$$

$\mathcal{H}_2$-Model Reduction

$\mathcal{H}_2$-Optimality Conditions

Let us assume $\hat{\Sigma}$ is given by its eigenvalue decomposition:

$$\hat{A} = R\Lambda R^{-1}, \quad \tilde{N}_i = R^{-1}\hat{N}_i R, \quad \tilde{B} = R^{-1}\hat{B}, \quad \tilde{C} = \hat{C}R.$$

Using Λ, $\tilde{N}_i$, $\tilde{B}$, $\tilde{C}$ as optimization parameters, we can derive necessary conditions for $\mathcal{H}_2$-optimality, e.g.:

$\mathcal{H}_2$-Model Reduction

$\mathcal{H}_2$-Optimality Conditions

Let us assume $\hat{\Sigma}$ is given by its eigenvalue decomposition:

$$\hat{A} = R\Lambda R^{-1}, \quad \tilde{N}_i = R^{-1}\hat{N}_i R, \quad \tilde{B} = R^{-1}\hat{B}, \quad \tilde{C} = \hat{C}R.$$

Using Λ, $\tilde{N}_i$, $\tilde{B}$, $\tilde{C}$ as optimization parameters, we can derive necessary conditions for $\mathcal{H}_2$-optimality, e.g.:

$$(\operatorname{vec}(I_q))^T \left(e_j e_\ell^T \otimes C\right) \left(-\Lambda \otimes I_n - I_{\hat{n}} \otimes A - \sum_{i=1}^{m} \tilde{N}_i \otimes N_i\right)^{-1} \left(\tilde{B} \otimes B\right) \operatorname{vec}(I_m)$$

$$= (\operatorname{vec}(I_q))^T \left(e_j e_\ell^T \otimes \hat{C}\right) \left(-\Lambda \otimes I_n - I_{\hat{n}} \otimes \hat{A} - \sum_{i=1}^{m} \tilde{N}_i \otimes \hat{N}_i\right)^{-1} \left(\tilde{B} \otimes \hat{B}\right) \operatorname{vec}(I_m).$$

$\mathcal{H}_2$-Model Reduction

$\mathcal{H}_2$-Optimality Conditions

Let us assume $\hat{\Sigma}$ is given by its eigenvalue decomposition:

$$\hat{A} = R\Lambda R^{-1}, \quad \tilde{N}_i = R^{-1}\hat{N}_i R, \quad \tilde{B} = R^{-1}\hat{B}, \quad \tilde{C} = \hat{C}R.$$

Using Λ, $\tilde{N}_i$, $\tilde{B}$, $\tilde{C}$ as optimization parameters, we can derive necessary conditions for $\mathcal{H}_2$-optimality, e.g.:

$$(\mathrm{vec}(I_q))^T \left(e_j e_\ell^T \otimes C\right) \left(-\Lambda \otimes I_n - I_{\hat{n}} \otimes A - \sum_{i=1}^{m} \tilde{N}_i \otimes N_i\right)^{-1} \left(\tilde{B} \otimes B\right) \mathrm{vec}(I_m)$$

$$= (\mathrm{vec}(I_q))^T \left(e_j e_\ell^T \otimes \hat{C}\right) \left(-\Lambda \otimes I_n - I_{\hat{n}} \otimes \hat{A} - \sum_{i=1}^{m} \tilde{N}_i \otimes \hat{N}_i\right)^{-1} \left(\tilde{B} \otimes \hat{B}\right) \mathrm{vec}(I_m).$$

Where is the connection to the interpolation of transfer functions?

$\mathcal{H}_2$-Model Reduction

$\mathcal{H}_2$-Optimality Conditions

Let us assume $\hat{\Sigma}$ is given by its eigenvalue decomposition:

$$\hat{A} = R\Lambda R^{-1}, \quad \tilde{N}_i = R^{-1}\hat{N}_i R, \quad \tilde{B} = R^{-1}\hat{B}, \quad \tilde{C} = \hat{C}R.$$

Using Λ, $\tilde{N}_i$, $\tilde{B}$, $\tilde{C}$ as optimization parameters, we can derive necessary conditions for $\mathcal{H}_2$-optimality, e.g.:

$$(\mathrm{vec}(I_q))^T \left(e_j e_\ell^T \otimes C\right) \left(-\Lambda \otimes I_n - I_{\hat{n}} \otimes A - \sum_{i=1}^m \tilde{N}_i \otimes N_i\right)^{-1} \left(\tilde{B} \otimes B\right) \mathrm{vec}(I_m)$$
$$= (\mathrm{vec}(I_q))^T \left(e_j e_\ell^T \otimes \hat{C}\right) \left(-\Lambda \otimes I_n - I_{\hat{n}} \otimes \hat{A} - \sum_{i=1}^m \tilde{N}_i \otimes \hat{N}_i\right)^{-1} \left(\tilde{B} \otimes \hat{B}\right) \mathrm{vec}(I_m).$$

$$(\mathrm{vec}(I_q))^T \left(e_j e_\ell^T \otimes C\right) \left(-\Lambda \otimes I_n - I_{\hat{n}} \otimes A\right)^{-1} \left(\tilde{B} \otimes B\right) \mathrm{vec}(I_m)$$
$$= (\mathrm{vec}(I_q))^T \left(e_j e_\ell^T \otimes \hat{C}\right) \left(-\Lambda \otimes I_n - I_{\hat{n}} \otimes \hat{A}\right)^{-1} \left(\tilde{B} \otimes \hat{B}\right) \mathrm{vec}(I_m).$$

$\mathcal{H}_2$-Model Reduction

$\mathcal{H}_2$-Optimality Conditions

Let us assume $\hat{\Sigma}$ is given by its eigenvalue decomposition:

$$\hat{A} = R\Lambda R^{-1}, \quad \tilde{N}_i = R^{-1}\hat{N}_i R, \quad \tilde{B} = R^{-1}\hat{B}, \quad \tilde{C} = \hat{C}R.$$

Using Λ, $\tilde{N}_i$, $\tilde{B}$, $\tilde{C}$ as optimization parameters, we can derive necessary conditions for $\mathcal{H}_2$-optimality, e.g.:

$$(\mathrm{vec}(I_q))^T \left(e_j e_\ell^T \otimes C\right) \left(-\Lambda \otimes I_n - I_{\hat{n}} \otimes A - \sum_{i=1}^{m} \tilde{N}_i \otimes N_i\right)^{-1} \left(\tilde{B} \otimes B\right) \mathrm{vec}(I_m)$$

$$= (\mathrm{vec}(I_q))^T \left(e_j e_\ell^T \otimes \hat{C}\right) \left(-\Lambda \otimes I_n - I_{\hat{n}} \otimes \hat{A} - \sum_{i=1}^{m} \tilde{N}_i \otimes \hat{N}_i\right)^{-1} \left(\tilde{B} \otimes \hat{B}\right) \mathrm{vec}(I_m).$$

$$(\mathrm{vec}(I_q))^T \left(e_j e_\ell^T \otimes C\right) (-\Lambda \otimes I_n - I_{\hat{n}} \otimes A)^{-1} \mathrm{vec}(B\tilde{B}^T)$$

$$= (\mathrm{vec}(I_q))^T \left(e_j e_\ell^T \otimes \hat{C}\right) \left(-\Lambda \otimes I_n - I_{\hat{n}} \otimes \hat{A}\right)^{-1} \mathrm{vec}(\hat{B}\tilde{B}^T).$$

$\mathcal{H}_2$-Model Reduction

$\mathcal{H}_2$-Optimality Conditions

Let us assume $\hat{\Sigma}$ is given by its eigenvalue decomposition:

$$\hat{A} = R\Lambda R^{-1}, \quad \tilde{N}_i = R^{-1}\hat{N}_i R, \quad \tilde{B} = R^{-1}\hat{B}, \quad \tilde{C} = \hat{C}R.$$

Using Λ, $\tilde{N}_i$, $\tilde{B}$, $\tilde{C}$ as optimization parameters, we can derive necessary conditions for $\mathcal{H}_2$-optimality, e.g.:

$$(\mathrm{vec}(I_q))^T \left(e_j e_\ell^T \otimes C\right) \left(-\Lambda \otimes I_n - I_{\hat{n}} \otimes A - \sum_{i=1}^m \tilde{N}_i \otimes N_i\right)^{-1} \left(\tilde{B} \otimes B\right) \mathrm{vec}(I_m)$$

$$= (\mathrm{vec}(I_q))^T \left(e_j e_\ell^T \otimes \hat{C}\right) \left(-\Lambda \otimes I_n - I_{\hat{n}} \otimes \hat{A} - \sum_{i=1}^m \tilde{N}_i \otimes \hat{N}_i\right)^{-1} \left(\tilde{B} \otimes \hat{B}\right) \mathrm{vec}(I_m).$$

$$(\mathrm{vec}(I_q))^T \left(e_j e_\ell^T \otimes C\right) \begin{pmatrix} -\lambda_1 I - A & & \\ & \ddots & \\ & & -\lambda_{\hat{n}} I - A \end{pmatrix}^{-1} \begin{pmatrix} B\tilde{B}_1^T \\ \vdots \\ B\tilde{B}_{\hat{n}}^T \end{pmatrix}$$

$$= (\mathrm{vec}(I_q))^T \left(e_j e_\ell^T \otimes \hat{C}\right) \begin{pmatrix} -\lambda_1 I - \hat{A} & & \\ & \ddots & \\ & & -\lambda_{\hat{n}} I - \hat{A} \end{pmatrix}^{-1} \begin{pmatrix} \hat{B}\tilde{B}_1^T \\ \vdots \\ \hat{B}\tilde{B}_{\hat{n}}^T \end{pmatrix}.$$

$\mathcal{H}_2$-Model Reduction

$\mathcal{H}_2$-Optimality Conditions

Let us assume $\hat{\Sigma}$ is given by its eigenvalue decomposition:

$$\hat{A} = R\Lambda R^{-1}, \quad \tilde{N}_i = R^{-1}\hat{N}_i R, \quad \tilde{B} = R^{-1}\hat{B}, \quad \tilde{C} = \hat{C}R.$$

Using Λ, $\tilde{N}_i$, $\tilde{B}$, $\tilde{C}$ as optimization parameters, we can derive necessary conditions for $\mathcal{H}_2$-optimality, e.g.:

$$(\mathrm{vec}(I_q))^T \left(e_j e_\ell^T \otimes C\right) \left(-\Lambda \otimes I_n - I_{\hat{n}} \otimes A - \sum_{i=1}^{m} \tilde{N}_i \otimes N_i\right)^{-1} \left(\tilde{B} \otimes B\right) \mathrm{vec}(I_m)$$

$$= (\mathrm{vec}(I_q))^T \left(e_j e_\ell^T \otimes \hat{C}\right) \left(-\Lambda \otimes I_n - I_{\hat{n}} \otimes \hat{A} - \sum_{i=1}^{m} \tilde{N}_i \otimes \hat{N}_i\right)^{-1} \left(\tilde{B} \otimes \hat{B}\right) \mathrm{vec}(I_m).$$

$$H(-\lambda_\ell)\tilde{B}_\ell^T = \hat{H}(-\lambda_\ell)\tilde{B}_\ell^T$$

$\rightsquigarrow$ tangential interpolation at mirror images of reduced system poles

$\mathcal{H}_2$-Model Reduction

$\mathcal{H}_2$-Optimality Conditions

Let us assume $\hat{\Sigma}$ is given by its eigenvalue decomposition:

$$\hat{A} = R\Lambda R^{-1}, \quad \tilde{N}_i = R^{-1}\hat{N}_i R, \quad \tilde{B} = R^{-1}\hat{B}, \quad \tilde{C} = \hat{C}R.$$

Using Λ, $\tilde{N}_i$, $\tilde{B}$, $\tilde{C}$ as optimization parameters, we can derive necessary conditions for $\mathcal{H}_2$-optimality, e.g.:

$$(\mathrm{vec}(I_q))^T \left(e_j e_\ell^T \otimes C\right) \left(-\Lambda \otimes I_n - I_{\hat{n}} \otimes A - \sum_{i=1}^m \tilde{N}_i \otimes N_i\right)^{-1} \left(\tilde{B} \otimes B\right) \mathrm{vec}(I_m)$$

$$= (\mathrm{vec}(I_q))^T \left(e_j e_\ell^T \otimes \hat{C}\right) \left(-\Lambda \otimes I_n - I_{\hat{n}} \otimes \hat{A} - \sum_{i=1}^m \tilde{N}_i \otimes \hat{N}_i\right)^{-1} \left(\tilde{B} \otimes \hat{B}\right) \mathrm{vec}(I_m).$$

$$H(-\lambda_\ell)\tilde{B}_\ell^T = \hat{H}(-\lambda_\ell)\tilde{B}_\ell^T$$

$\rightsquigarrow$ tangential interpolation at mirror images of reduced system poles

Note: [Flagg 2011] shows equivalence to interpolating the Volterra series!

A First Iterative Approach

Algorithm 1 Bilinear IRKA

Input: A, N_i, B, C, $\hat{A}$, $\hat{N}_i$, $\hat{B}$, $\hat{C}$

Output: A^{opt}, N_i^{opt}, B^{opt}, C^{opt}

while (change in $\Lambda > \epsilon$) **do**

$R\Lambda R^{-1} = \hat{A}$, $\tilde{B} = R^{-1}\hat{B}$, $\tilde{C} = \hat{C}R$, $\tilde{N}_i = R^{-1}\hat{N}_i R$

$$\operatorname{vec}(V) = \left(-\Lambda \otimes I_n - I_{\hat{n}} \otimes A - \sum_{i=1}^{m} \tilde{N}_i \otimes N_i\right)^{-1} \left(\tilde{B} \otimes B\right) \operatorname{vec}(I_m)$$

$$\operatorname{vec}(W) = \left(-\Lambda \otimes I_n - I_{\hat{n}} \otimes A^T - \sum_{i=1}^{m} \tilde{N}_i^T \otimes N_i^T\right)^{-1} \left(\tilde{C}^T \otimes C^T\right) \operatorname{vec}(I_q)$$

$V = \operatorname{orth}(V)$, $W = \operatorname{orth}(W)$

$\hat{A} = \left(W^T V\right)^{-1} W^T A V$, $\hat{N}_i = \left(W^T V\right)^{-1} W^T N_i V$,
$\hat{B} = \left(W^T V\right)^{-1} W^T B$, $\hat{C} = CV$

end while

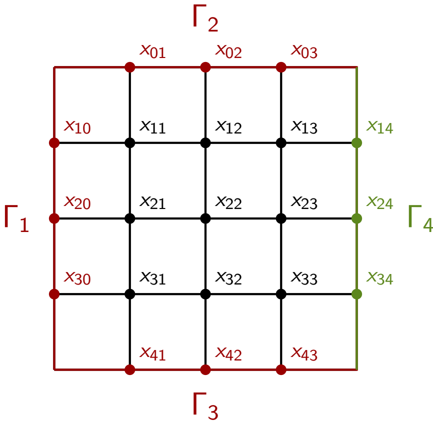
$A^{opt} = \hat{A}$, $N_i^{opt} = \hat{N}_i$, $B^{opt} = \hat{B}$, $C^{opt} = \hat{C}$

Numerical Examples

A Heat Transfer Model

- 2-dimensional heat distribution
  [B./Saak '05]
- Boundary control by spraying intensities of a cooling fluid

$$\begin{aligned} \Omega &= (0,1) \times (0,1), \\ x_t &= \Delta x && \text{in } \Omega, \\ n \cdot \nabla x &= c \cdot u_{1,2,3}(x-1) && \text{on } \Gamma_1, \Gamma_2, \Gamma_3, \\ x &= u_4 && \text{on } \Gamma_4. \end{aligned}$$



- Spatial discretization $k \times k$-grid

$$\Rightarrow \dot{x} \approx A_1 x + \sum_{i=1}^{3} N_i x u_i + Bu$$

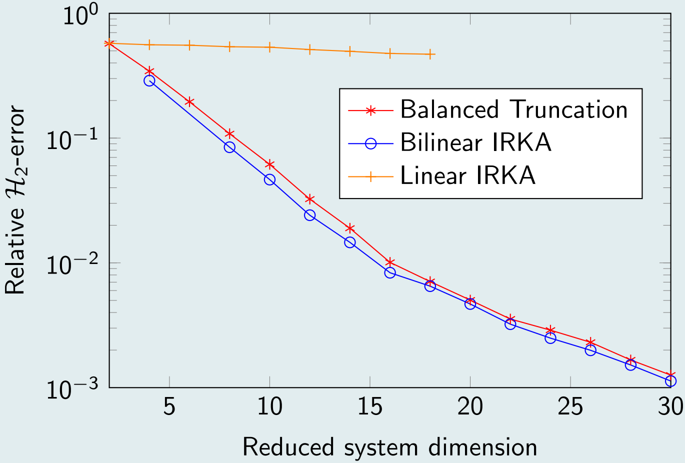
$$\Rightarrow A_2 = 0.$$

- Output: $y = \frac{1}{k^2} \begin{bmatrix} 1 & \dots & 1 \end{bmatrix}$.

Numerical Examples

A Heat Transfer Model

Comparison of relative $\mathcal{H}_2$-error for $n = 10.000$



Numerical Examples

Fokker-Planck Equation

As a second example, we consider a dragged Brownian particle whose one-dimensional motion is given by

$$dX_t = -\nabla V(X_t, t)dt + \sqrt{2\sigma}dW_t,$$

with $\sigma = \frac{2}{3}$ and $V(x, u) = W(x, t) + \Phi(x, u_t) = (x^2 - 1)^2 - xu - x$.
Alternatively, one can consider ([Hartmann et al. '10]) ,

$$\rho(x, t)dx = \mathbf{P}\left[X_t \in [x, x + dx)\right]$$

which is described by the Fokker-Planck equation

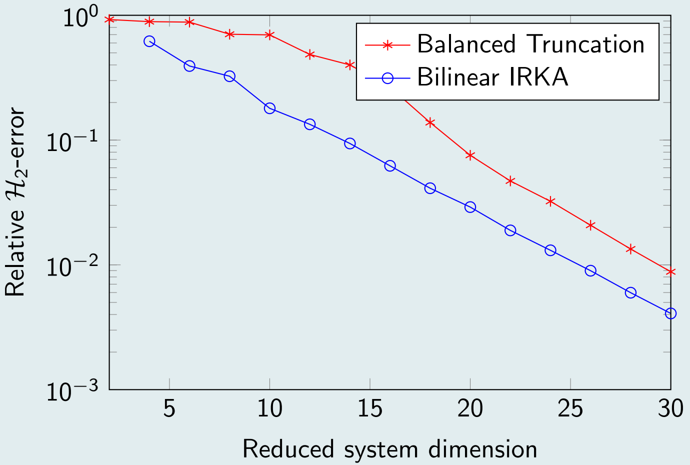
$$\begin{aligned} \frac{\partial \rho}{\partial t} &= \sigma\Delta\rho + \nabla \cdot (\rho\nabla V), && (x, t) \in (-2, 2) \times (0, T], \\ 0 &= \sigma\nabla\rho + \rho\nabla B, && (x, t) \in \{-2, 2\} \times [0, T], \\ \rho_0 &= \rho, && (x, t) \in (-2, 2) \times 0. \end{aligned}$$

Output C discrete characteristic function of the interval $[0.95, 1.05]$.

Numerical Examples

Fokker-Planck Equation

Comparison of relative $\mathcal{H}_2$-error for $n = 500$



Nonlinear Model Reduction

Quadratic-Bilinear Differential Algebraic Equations (QBDAEs)

Coming back to the more general case with nonlinear $f(x)$, we 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-based reduction techniques.

Nonlinear Model Reduction

Transformation 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

Transformation 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

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

Nonlinear Model Reduction

Transformation 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

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

Nonlinear Model Reduction

Transformation 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

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

Nonlinear Model Reduction

Transformation 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

Transformation 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

Transformation 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

Transformation 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

Variational Analysis and Linear Subsystems

Analysis of nonlinear systems by variational equation approach:

Nonlinear Model Reduction

Variational Analysis and Linear Subsystems

Analysis of nonlinear systems by variational equation approach:

- consider input of the form $\alpha u(t)$,

Nonlinear Model Reduction

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

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

Nonlinear Model Reduction

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

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

Nonlinear Model Reduction

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

Generalized Transfer Functions

In a similar way, a series of generalized symmetric transfer functions can be obtained via the growing exponential approach:

Nonlinear Model Reduction

Generalized Transfer Functions

In a similar way, a series of generalized symmetric transfer functions can be obtained via the growing exponential approach:

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

Nonlinear Model Reduction

Generalized Transfer Functions

In a similar way, a series of generalized symmetric transfer functions can be obtained via the growing exponential approach:

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

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

Nonlinear Model Reduction

Generalized Transfer Functions

In a similar way, a series of generalized symmetric transfer functions can be obtained via the growing exponential approach:

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

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

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

Nonlinear Model Reduction

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

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

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

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

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 $i = 1 : q$

$$V_2^i = \mathcal{K}_{q-i+1}\left((A_1-2\sigma E)^{-1}E, (A_1-2\sigma E)^{-1}NV_1(:,i)\right),$$

Nonlinear Model Reduction

Constructing the Projection Matrix

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

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 $i = 1:q$

$$V_2^i = \mathcal{K}_{q-i+1}\left((A_1-2\sigma E)^{-1}E, (A_1-2\sigma E)^{-1}NV_1(:,i)\right),$$

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

$$V_3^{i,j} = \mathcal{K}_{q-i-j+2}\left((A_1-2\sigma E)^{-1}E, (A_1-2\sigma E)^{-1}A_2V_1(:,i)\otimes V_1(:,j)\right),$$

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

Nonlinear Model Reduction

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 $i = 1:q$

$$V_2^i = \mathcal{K}_{q-i+1}\left((A_1-2\sigma E)^{-1}E, (A_1-2\sigma E)^{-1}NV_1(:,i)\right),$$

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

Tensors and Matricizations: A Short Excursion [Kolda/Bader '09, Grasedyck '10]

A tensor is a vector

$$(A_i)_{i \in \mathcal{I}} \in \mathbb{R}^{\mathcal{I}}$$

indexed by a product index set

$$\mathcal{I} = \mathcal{I}_1 \times \cdots \times \mathcal{I}_d, \quad \#\mathcal{I}_j = n_j.$$

Nonlinear Model Reduction

Tensors and Matricizations: A Short Excursion [Kolda/Bader '09, Grasedyck '10]

A tensor is a vector

$$(A_i)_{i\in\mathcal{I}} \in \mathbb{R}^{\mathcal{I}}$$

indexed by a product index set

$$\mathcal{I} = \mathcal{I}_1 \times \cdots \times \mathcal{I}_d, \quad \#\mathcal{I}_j = n_j.$$

For a given tensor A, the t-matricization $A^{(t)}$ is defined as

$$A^{(t)} \in \mathbb{R}^{\mathcal{I}_t \times \mathcal{I}_{t'}}, \quad A^{(t)}_{(i_\mu)_{\mu\in t},\, (i_\mu)_{\mu\in t'}} := A_{(i_1,\ldots,i_d)}, \quad t' := \{1,\ldots,d\}\setminus t.$$

Nonlinear Model Reduction

Tensors and Matricizations: A Short Excursion [Kolda/Bader '09, Grasedyck '10]

A tensor is a vector

$$(A_i)_{i \in \mathcal{I}} \in \mathbb{R}^{\mathcal{I}}$$

indexed by a product index set

$$\mathcal{I} = \mathcal{I}_1 \times \cdots \times \mathcal{I}_d, \quad \#\mathcal{I}_j = n_j.$$

For a given tensor A, the t-matricization $A^{(t)}$ is defined as

$$A^{(t)} \in \mathbb{R}^{\mathcal{I}_t \times \mathcal{I}_{t'}}, \quad A^{(t)}_{(i_\mu)_{\mu \in t},\, (i_\mu)_{\mu \in t'}} := A_{(i_1,\ldots,i_d)}, \quad t' := \{1,\ldots,d\} \backslash t.$$

Example: For a given 3-tensor $A_{(i_1,i_2,i_3)}$ with $i_1, i_2, i_3 \in \{1,2\}$, we have:

$$A^{(1)} = \begin{bmatrix} A_{(1,1,1)} & A_{(1,2,1)} & A_{(1,1,2)} & A_{(1,2,2)} \\ A_{(2,1,1)} & A_{(2,2,1)} & A_{(2,1,2)} & A_{(2,2,2)} \end{bmatrix},$$

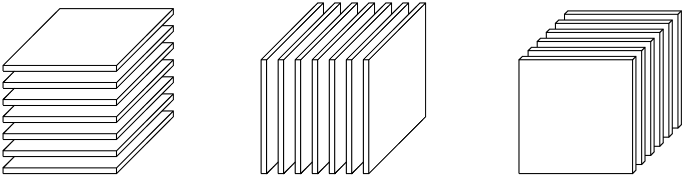
$$A^{(2)} = \begin{bmatrix} A_{(1,1,1)} & A_{(2,1,1)} & A_{(1,1,2)} & A_{(2,1,2)} \\ A_{(1,2,1)} & A_{(2,2,1)} & A_{(1,2,2)} & A_{(2,2,2)} \end{bmatrix}.$$

Nonlinear Model Reduction

Tensors and Matricizations: A Short Excursion [Kolda/Bader '09, Grasedyck '10]

For a given tensor A, the *t-matricization* $A^{(t)}$ is defined as

$$A^{(t)} \in \mathbb{R}^{\mathcal{I}_t \times \mathcal{I}_{t'}}, \quad A^{(t)}_{(i_\mu)_{\mu\in t},\,(i_\mu)_{\mu\in t'}} := A_{(i_1,\ldots,i_d)}, \quad t' := \{1,\ldots,d\}\setminus t.$$



(a) Horizontal slices (b) Lateral slices (c) Frontal slices

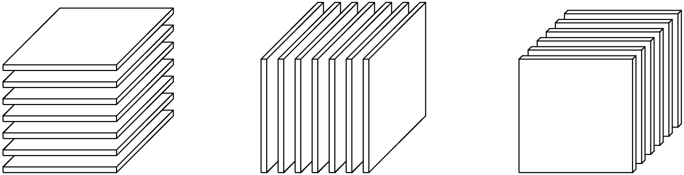
Figure : Slices of a 3rd-order tensor. [Courtesy of Tammy Kolda]

Nonlinear Model Reduction

Tensors and Matricizations: A Short Excursion [Kolda/Bader '09, Grasedyck '10]

For a given tensor A, the *t*-matricization $A^{(t)}$ is defined as

$$A^{(t)} \in \mathbb{R}^{\mathcal{I}_t \times \mathcal{I}_{t'}}, \quad A^{(t)}_{(i_\mu)_{\mu\in t},\,(i_\mu)_{\mu\in t'}} := A_{(i_1,\ldots,i_d)}, \quad t' := \{1,\ldots,d\}\setminus t.$$



(a) Horizontal slices (b) Lateral slices (c) Frontal slices

Figure : Slices of a 3rd-order tensor. [Courtesy of Tammy Kolda]

⇝ Allows to compute matrix products more efficiently.

Nonlinear Model Reduction

Two-Sided Projection Methods

Similarly to the linear case, one can exploit duality concepts, in order to construct two-sided projection methods.

Nonlinear Model Reduction

Two-Sided Projection Methods

Similarly to the linear case, one can exploit duality concepts, in order to construct two-sided projection methods.

Interpreting $\mathcal{A}^{(2)}$ now as the 2-matricization of the Hessian 3-tensor corresponding to A_2, one can show that the dual Krylov spaces have to be constructed as follows

$$W_1 = \mathcal{K}_q\left((A_1 - 2\sigma E)^{-T} E^T, (A_1 - 2\sigma E)^{-T} c\right)$$

for $i = 1 : q$

$$W_2^i = \mathcal{K}_{q-i+1}\left((A_1 - \sigma E)^{-T} E^T, (A_1 - \sigma E)^{-T} N^T W_1(:, i)\right),$$

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

$$W_3^{i,j} = \mathcal{K}_{q-i-j+2}\left((A_1 - \sigma E)^{-T} E^T, (A_1 - \sigma E)^{-T} \mathcal{A}^{(2)} V_1(:, i) \otimes W_1(:, j)\right),$$

Nonlinear Model Reduction

Two-Sided Projection Methods

Similarly to the linear case, one can exploit duality concepts, in order to construct two-sided projection methods.

Interpreting $\mathcal{A}^{(2)}$ now as the 2-matricization of the Hessian 3-tensor corresponding to A_2, one can show that the dual Krylov spaces have to be constructed as follows

$$W_1 = \mathcal{K}_q\left((A_1 - 2\sigma E)^{-T} E^T, (A_1 - 2\sigma E)^{-T} c\right)$$

for $i = 1 : q$

$$W_2^i = \mathcal{K}_{q-i+1}\left((A_1 - \sigma E)^{-T} E^T, (A_1 - \sigma E)^{-T} N^T W_1(:, i)\right),$$

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

$$W_3^{i,j} = \mathcal{K}_{q-i-j+2}\left((A_1 - \sigma E)^{-T} E^T, (A_1 - \sigma E)^{-T} \mathcal{A}^{(2)} V_1(:, i) \otimes W_1(:, j)\right),$$

Note: Due to the symmetry of the Hessian tensor, the 3-matricization $\mathcal{A}^{(3)}$ coincides with $\mathcal{A}^{(2)}$.

Nonlinear Model Reduction

Multimoment matching

Theorem

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

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

$$\textbf{for } i = 1 : q_2$$

$$V_2 = \mathcal{K}_{q_2-i+1}\left(E, A_1, NV_1(:,i), 2\sigma\right)$$

$$W_2 = \mathcal{K}_{q_2-i+1}\left(E^T, A_1^T, N^T W_1(:,i), \sigma\right)$$

$$\textbf{for } j = 1 : \min(q_2 - i + 1, i)$$

$$V_3 = \mathcal{K}_{q_2-i-j+2}\left(E, A_1, A_2 V_1(:,i) \otimes V_1(:,j), 2\sigma\right)$$

$$W_3 = \mathcal{K}_{q_2-i-j+2}\left(E^T, A_1^T, \mathcal{A}^{(2)} V_1(:,i) \otimes W_1(:,j), \sigma\right).$$

Then, it holds:

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

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

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 $\underbrace{(0,1)\times(0,1)}_{:=\Omega}\times[0,T]$

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

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

- Consider initial and boundary conditions

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

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.

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

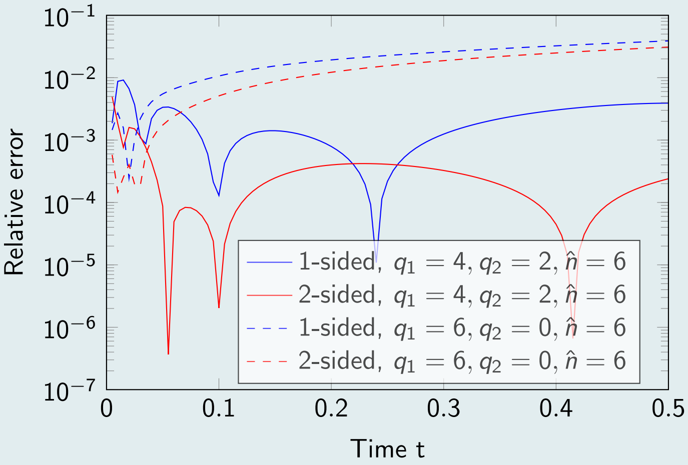
$$\begin{aligned}
&u_x(x,y,0)=\frac{\sqrt{2}}{2}, && u_y(x,y,0)=\frac{\sqrt{2}}{2}, && \text{for } (x,y)\in\Omega_1 := (0,0.5],\\
&u_x(x,y,0)=0, && u_y(x,y,0)=0, && \text{for } (x,y)\in\Omega\backslash\Omega_1,\\
&u_x=0, && u_y=0, && \text{for } (x,y)\in\partial\Omega.
\end{aligned}$$

- 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 $\underbrace{(0,1)\times(0,1)}_{:=\Omega}\times[0,T]$

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

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

- Now consider initial and boundary conditions

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

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

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

- 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 $q_1=5, q_2=2, \sigma=0, \hat{n}=52$.

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 $q_1=5, q_2=2, \sigma=0, \hat{n}=52$.
- 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:

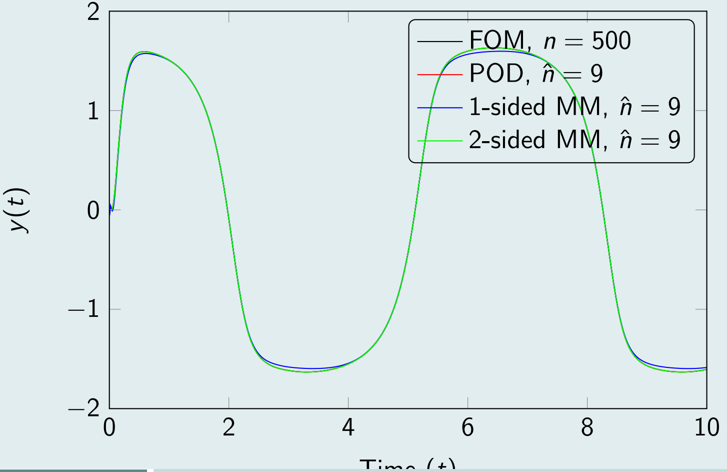
$$\begin{aligned}
v_t + v^3 &= v_{xx} + v, && \text{in } (0,1) \times (0,T), \\
v(0,\cdot) &= u(t), && \text{in } (0,T), \\
v_x(1,\cdot) &= 0, && \text{in } (0,T), \\
v(x,0) &= v_0(x), && \text{in } (0,1)
\end{aligned}$$

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

Nonlinear Model Reduction by Generalized Moment-Matching

The Chafee-Infante equation

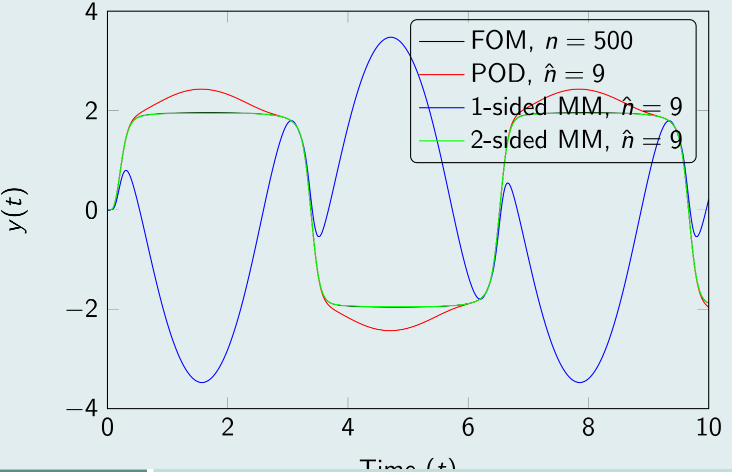
Comparison between moment-matching and POD ($u(t) = 5\cos(t)$)



Nonlinear Model Reduction by Generalized Moment-Matching

The Chafee-Infante equation

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



Nonlinear Model Reduction by Generalized Moment-Matching

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

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

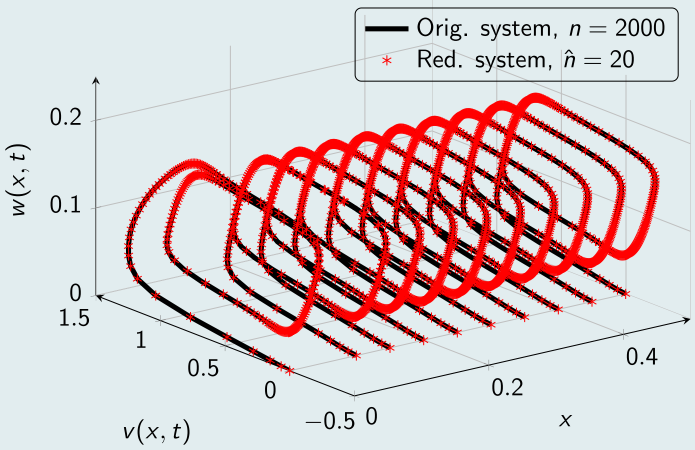
  where

  $\epsilon = 0.015,\ h = 0.5,\ \gamma = 2,\ g = 0.05,\ i_0(t) = 5 \cdot 10^4 t^3 \exp(-15t)$
- original state dimension $n = 2 \cdot 1000$, QBDAE dimension $N = 3 \cdot 1000$, reduced QBDAE dimension $r = 20$

Numerical Examples

The FitzHugh-Nagumo System

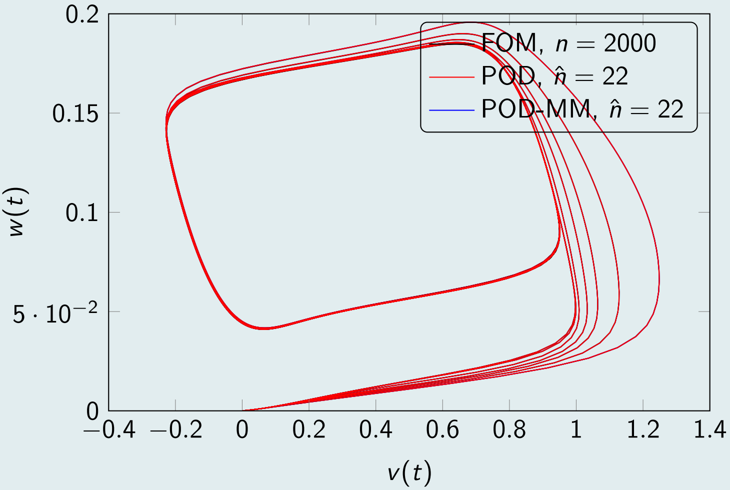
Limit cycle behavior for 1-sided proj. (ROM, $\hat{n} = 20, \sigma = 4$)



Numerical Examples

The FitzHugh-Nagumo System

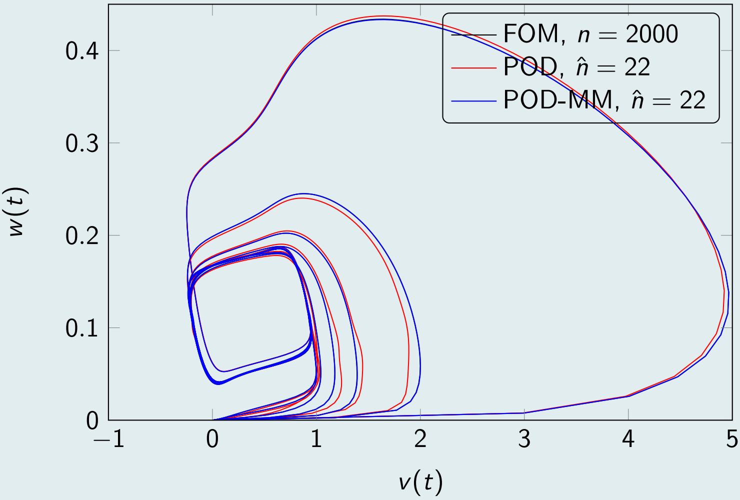
POD via moment-matching (training input)



Numerical Examples

The FitzHugh-Nagumo System

POD via moment-matching (varying input)



Conclusions and Outlook

- Many nonlinear dynamics can be expressed by a system of quadratic-bilinear differential algebraic equations.
- For this type of systems, a frequency domain analysis leads to certain generalized transfer functions.
- There exist Krylov subspace methods that extend the concept of moment-matching $\rightsquigarrow$ using basic tools from tensor theory allows for better approximations.
- In contrast to other methods like TPWL and POD, the reduction process is independent of the control input.

Conclusions and Outlook

- Many nonlinear dynamics can be expressed by a system of quadratic-bilinear differential algebraic equations.
- For this type of systems, a frequency domain analysis leads to certain generalized transfer functions.
- There exist Krylov subspace methods that extend the concept of moment-matching $\rightsquigarrow$ using basic tools from tensor theory allows for better approximations.
- In contrast to other methods like TPWL and POD, the reduction process is independent of the control input.
- Optimal choice of interpolation points?
- Stability/index-preserving reduction possible?

References

P. Benner and T. Breiten.
Interpolation-Based H_2-Model Reduction of Bilinear Control Systems.
SIAM Journal on Matrix Analysis and Applications, 33(3):859–885, 2012.

P. Benner and T. Breiten.
Krylov-Subspace Based Model Reduction of Nonlinear Circuit Models Using Bilinear and Quadratic-Linear Approximations.
In M. Günther, A. Bartel, M. Brunk, S. Schöps, M. Striebel (Eds.), *Progress in Industrial Mathematics at ECMI 2010*, Mathematics in Industry, 17:153–159, Springer-Verlag, Berlin, 2012.

P. Benner and T. Breiten.
Two-Sided Moment Matching Methods for Nonlinear Model Reduction.
MPI Magdeburg Preprint MPIMD/12-12, June 2012.

P. Benner and T. Damm.
Lyapunov Equations, Energy Functionals, and Model Order Reduction of Bilinear and Stochastic Systems.
SIAM Journal on Control and Optimiztion, 49(2):686–711, 2011.

C. Gu.
QLMOR: A Projection-Based Nonlinear Model Order Reduction Approach Using Quadratic-Linear Representation of Nonlinear Systems.
IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 30(9):1307–1320, 2011.