Westfälische Wilhelms-Universität Münster

Applied Mathematics Münster

Combined State and Parameter Reduction

Christian Himpe (christian.himpe@uni-muenster.de)
Mario Ohlberger (mario.ohlberger@uni-muenster.de)

WWU Münster
Institute for Computational and Applied Mathematics

GAMM 2014
π-Day

Overview

Control Systems

General Control System:

$$\dot{x}(t) = f(t, x(t), u(t), \theta)$$
$$y(t) = g(t, x(t), u(t), \theta)$$
$$x(0) = x_0$$

- State $x(t) \in \mathbb{R}^N$
- Input $u(t) \in \mathbb{R}^M$
- Output $y(t) \in \mathbb{R}^O$
- Paramter $\theta \in \mathbb{R}^P$
- Vector Field $f : \mathbb{R} \times \mathbb{R}^N \times \mathbb{R}^M \times \mathbb{R}^P \to \mathbb{R}^N$
- Functional $g : \mathbb{R} \times \mathbb{R}^N \times \mathbb{R}^M \times \mathbb{R}^P \to \mathbb{R}^O$

Control Systems

General Control System:

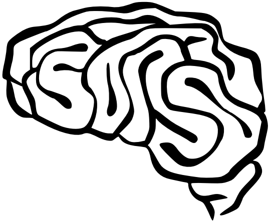
$$\dot{x}(t) = f(t, x(t), u(t), \theta)$$
$$y(t) = g(t, x(t), u(t), \theta)$$
$$x(0) = x_0$$

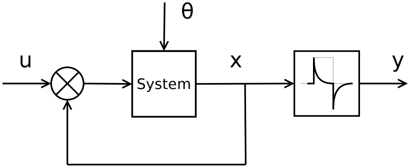
Example:

$$\dot{x}(t) = A(\theta)x(t) + Bu(t)$$
$$y(t) = Cx(t)$$
$$x(0) = x_0$$

- State $x(t) \in \mathbb{R}^N$
- Input $u(t) \in \mathbb{R}^M$
- Output $y(t) \in \mathbb{R}^O$
- Paramter $\theta \in \mathbb{R}^P$
- Vector Field $f : \mathbb{R} \times \mathbb{R}^N \times \mathbb{R}^M \times \mathbb{R}^P \to \mathbb{R}^N$
- Functional $g : \mathbb{R} \times \mathbb{R}^N \times \mathbb{R}^M \times \mathbb{R}^P \to \mathbb{R}^O$

Brain as a Control System[1]





- Inputs: Sensory Input
- States: Brain Regions
- Outputs: Measurement of Regions Activity
- Parameters: Connectivity of Regions

[1] Friston et al 2003

Neuronal Network Models

fMRI / fNIRS[2]

$$\dot{x} = A(\theta)x + Bu$$
$$\dot{z} = f(z, x)$$
$$y = g(z)$$

EEG / MEG[3]

$$\ddot{x} = A_L x + A_N(\theta) f(\dot{x}) + Bu$$
$$y = Cx$$

[2] Friston et al 2003, Stephan et al 2008
[3] David et al 2004, David et al 2006

Model Order Reduction

Model:

$$\dot{x} = f(x, u)$$
$$y = g(x, u)$$

Model Order:

- $\dim(x) \gg 1$
- $\dim(u) \ll \dim(x)$
- $\dim(y) \ll \dim(x)$

Reduced Order Model:

$$\dot{x}_r = f_r(x_r, u)$$
$$y_r = g_r(x_r, u)$$

Reduced Order:

- $\dim(x_r) \ll \dim(x), \ \|y - y_r\| \ll 1$

Model Reduction for Parametrized Systems

Parametrized Model:

$$\dot{x} = f(x, u, \theta)$$
$$y = g(x, u, \theta)$$

Model Order:

- $\theta \in \Theta$
- $\dim(x) \gg 1$
- $\dim(u) \ll \dim(x)$
- $\dim(y) \ll \dim(x)$

Reduced Order Model:

$$\dot{x}_r = f_r(x_r, u, \theta)$$
$$y_r = g_r(x_r, u, \theta)$$

Reduced Order:

- $\dim(x_r) \ll \dim(x),\ \|y - y_r\| \ll 1,\ \forall \theta \in \Theta$

Combined State and Parameter Reduction

Parametrized Model:

$$\dot{x} = f(x, u, \theta)$$
$$y = g(x, u, \theta)$$

Model Order:

- $\dim(\theta) \gg 1$
- $\dim(x) \gg 1$
- $\dim(u) \ll \dim(x)$
- $\dim(y) \ll \dim(x)$

Reduced Order Model:

$$\dot{x}_r = f_r(x_r, u, \theta_r)$$
$$y_r = g_r(x_r, u, \theta_r)$$

Reduced Order:

- $\dim(x_r) \ll \dim(x),\ \|y - y_r\| \ll 1,\ \forall \theta \in \Theta$

Projection-Based Model Reduction

State Projection:

$$V \in \mathbb{R}^{dim(x) \times r},\ r < \dim(x),\ VV^T = \mathbb{1}$$

State Reduced Model:

$$\dot{x}_r = f_r(x_r, u, \theta) = V^T f(Vx_r, u, \theta)$$
$$y_r = g_r(x_r, u, \theta) = g(Vx_r, u, \theta)$$

Projection-Based Model Reduction

Parameter Projection:

$$\mathcal{V} \in \mathbb{R}^{dim(\theta)\times s},\ s < \dim(\theta),\ \mathcal{V}\mathcal{V}^T = \mathbb{1}$$

Parameter Reduced Model:

$$\begin{aligned} \dot{x}_r &= f_r(x, u, \theta_r) = f(x, u, \mathcal{V}\theta_r) \\ y_r &= g_r(x, u, \theta_r) = g(x, u, \mathcal{V}\theta_r) \end{aligned}$$

Projection-Based Model Reduction

State and Parameter Projections:

$$V \in \mathbb{R}^{dim(x) \times r},\ r < \dim(x),\ VV^T = \mathbb{1}$$
$$\mathcal{V} \in \mathbb{R}^{dim(\theta) \times s},\ s < \dim(\theta),\ \mathcal{V}\mathcal{V}^T = \mathbb{1}$$

State and Parameter Reduced Model:

$$\dot{x}_r = f_r(x_r, u, \theta_r) = V^T f(Vx_r, u, \mathcal{V}\theta_r)$$
$$y_r = g_r(x_r, u, \theta_r) = g(Vx_r, u, \mathcal{V}\theta_r)$$

Dual Approach

Gramian-Based

- Approximately balance system,
- based on controllability and observability.
- Truncate least controllable and observable states.
- Treat parameters as constant states.

Optimization-Based

- Iteratively assemble projections,
- utilizing greedy algorithm to find reduced parameters.
- Build parameter projection base vector by vector.
- Parameter projection then induces the state projection.

Gramian-Based Model Reduction[4]

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

Controllability

How well can x be driven by u?
Controllability Gramian:

$$W_C := \int_0^\infty e^{At} B B^T e^{A^T t} \mathrm{d}t$$

Observability

How well is change in x visible in y?
Observability Gramian:

$$W_O := \int_0^\infty e^{A^T t} C^T C e^{At} \mathrm{d}t$$

Cross Gramian:

$$W_X := \int_0^\infty e^{At} B C e^{At} \mathrm{d}t$$

$$\dim(u) = \dim(y)$$

[4] Moore 1981, Fernando and Nicholson 1983

Empirical Gramians[5]

$$W_C := \int_0^\infty e^{At} B B^T e^{A^T t} \mathrm{d}t = \int_0^\infty x(t) x(t)^T \mathrm{d}t$$
$$W_O := \int_0^\infty e^{A^T t} C^T C e^{At} \mathrm{d}t = \int_0^\infty \chi(t) \chi(t)^T \mathrm{d}t$$

- Empirical Gramians equal Classic Gramians
- Extends to nonlinear systems
- POD-related method

[5] Lall et al 1999, Hahn et al 2002

Empirical Cross Gramian[7]

$$W_X := \int_0^\infty e^{At} B C e^{At} \mathrm{d}t = \int_0^\infty x(t) \chi(t)^T \mathrm{d}t$$

Approximate Balanced Truncation[6]:

$$W_X \stackrel{SVD}{=} VDU$$
$$V_r = PV, \quad P : \mathbb{R}^{N \times N} \to \mathbb{R}^{N \times r}$$

[6] Sorensen and Antoulas 2002

[7] Streif et al 2006

Empirical Joint Gramian[8]

Augmented System:

$$\dot{\tilde{x}} = \begin{pmatrix} \dot{x} \\ \dot{\theta} \end{pmatrix} = \begin{pmatrix} f(x, u, \theta) \\ 0 \end{pmatrix}$$
$$\tilde{y} = g(x, u, \theta)$$
$$\tilde{x}_0 = \begin{pmatrix} x_0 \\ \theta \end{pmatrix}$$

Joint Gramian:

$$W_J := \tilde{W}_X = \begin{pmatrix} W_X & w \\ 0 & 0 \end{pmatrix}$$

Cross-Identifiability Gramian:

$$W_{\ddot{I}} := -\frac{1}{2} w^T (W_X + W_X^T)^{-1} w$$

[8] Himpe and Ohlberger 2013 (Preprint)

Optimization-Based[9]

Purpose:

- Developed for large Bayesian inverse problems,
- to reduce model order before parameter optimization.

Core Idea:

- Find optimal reduced order parameter,
- by minimizing error between full and proposed reduced model.

[9] Bashir et al 2007, Lieberman et al 2010

Greedy-Algorithm[10]

1. Initialization:
   $\theta_0 \in \mathbb{R}^P, P_0 = \theta_0, V_0 = \bar{x}(\theta_0)$
2. Maximize error between original and (current) reduced model:
   $\theta_{I+1} = \underset{\tilde{\theta}}{\operatorname{argmax}}\, J(\tilde{\theta}) = \underset{\tilde{\theta}}{\operatorname{argmax}}\, \alpha \|y(\tilde{\theta}) - y_r(\tilde{\theta})\| - \beta \|\tilde{\theta}\|$
3. Snapshot of current proposed reduced model:
   $x_{I+1} = x(\theta_{I+1})$
4. Update parameter projection:
   $\mathcal{V}_{I+1} = \operatorname{orth}\begin{pmatrix}\mathcal{V}_I & \theta_{I+1}\end{pmatrix}$
5. Update state projection:
   $V_{I+1} = \operatorname{orth}\begin{pmatrix}V_I & \bar{x}_{I+1}\end{pmatrix}$

[10] Bui-Thanh et al 2008

Trust-Region[11]

1. Initialization:
   $\theta_0 = 1, P_0 = \theta_0, V_0 = \bar{x}(\theta_0)$
2. Maximize error between original and (current) reduced model:
   $\hat{\theta}_{I+1} = \underset{\tilde{\theta}}{\text{argmax}}\, \alpha \| y(P_I \tilde{\theta}) - y_r(P_I \tilde{\theta}) \| - \beta \| P_I \tilde{\theta} \|$
3. Snapshot of current proposed reduced model:
   $x_{I+1} = x(P_I \hat{\theta}_{I+1})$
4. Update parameter projection:
   $\mathcal{V}_{I+1} = \text{orth}\begin{pmatrix} \mathcal{V}_I & Q\hat{\theta}_{I+1} \end{pmatrix}$
5. Update state projection:
   $V_{I+1} = \text{orth}\begin{pmatrix} V_I & \bar{x}_{I+1} \end{pmatrix}$
6. Enlarge parameter vector:
   $\theta_{I+1} = \begin{pmatrix} \hat{\theta}_{I+1} \\ 0 \end{pmatrix}$

[11] Himpe and Ohlberger 2014 (Preprint)

Data-Driven[12]

Given some output data y_d:

$$J(\tilde{\theta}) = \alpha \|y(\tilde{\theta}) - y_r(\tilde{\theta})\| - \beta \|\tilde{\theta}\|$$

[12] Lieberman et al 2012

Data-Driven[12]

Given some output data y_d:

$$J(\tilde{\theta}) = \alpha \| y(\tilde{\theta}) - y_r(\tilde{\theta}) \| - \beta \| \tilde{\theta} \|$$
$$J(\tilde{\theta}) = \alpha \| y(\tilde{\theta}) - y_r(\tilde{\theta}) \| - \beta \| \tilde{\theta} \| - \gamma \| y_d - y_r(\tilde{\theta}) \|$$

[12] Lieberman et al 2012

Data-Driven[12]

Given some output data y_d:

$$J(\tilde{\theta}) = \alpha \|y(\tilde{\theta}) - y_r(\tilde{\theta})\| - \beta \|\tilde{\theta}\|$$

$$J(\tilde{\theta}) = \alpha \|y(\tilde{\theta}) - y_r(\tilde{\theta})\| - \beta \|\tilde{\theta}\| - \gamma \|y_d - y_r(\tilde{\theta})\|$$

$$J(\tilde{\theta}) = \qquad -\beta \|\tilde{\theta}\| - \gamma \|y_d - y_r(\tilde{\theta})\|$$

[12] Lieberman et al 2012

Software

emgr

- Empirical Controllability Gramian
- Empirical Observability Gramian
- Empirical Cross Gramian
- Empirical Approximate Cross Gramian
- Empirical Sensitivity Gramian
- Empirical Identifiability Gramian
- Empirical Joint Gramian

optmor

- Optional Data-Driven
- Optional Trust-Region
- Configurable Objective Function
- Configurable State Selection
- Configurable Orthogonalization
- Arbitrary Parametrizations
- Lazy Arguments

- Default and Custom Integrators
- (Somewhat) Multi-Core Ready
- Released under Open-Source License
- Compatible with OCTAVE and MATLAB

`http://gramian.de`

`http://j.mp/opt-mor`

Numerical Results

fMRI / fNIRS

$$\dot{x} = A(\theta)x + Bu$$
$$\dot{z} = f(z, x)$$
$$y = g(z)$$

EEG / MEG

$$\ddot{x} = A_L x + A_N(\theta) f(\dot{x}) + Bu$$
$$y = Cx$$

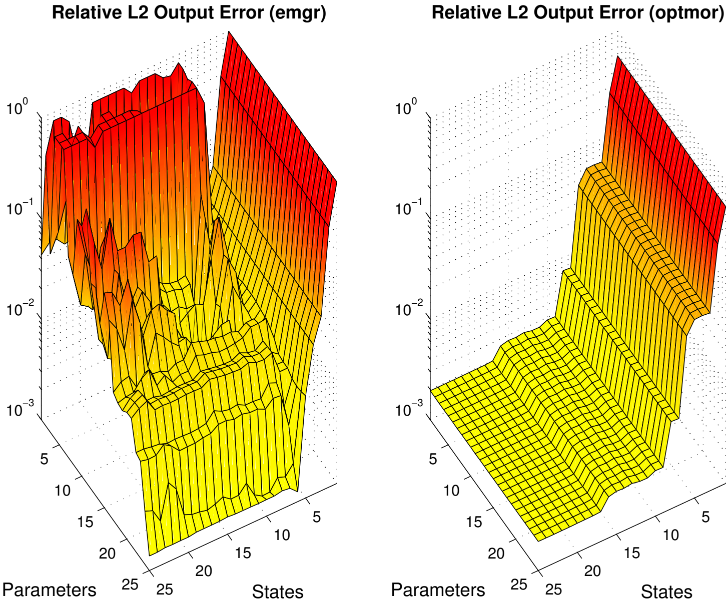
Linearization[13]:

$$\dot{x} = \hat{A}(\theta)x + Bu$$
$$y = \hat{C}x$$

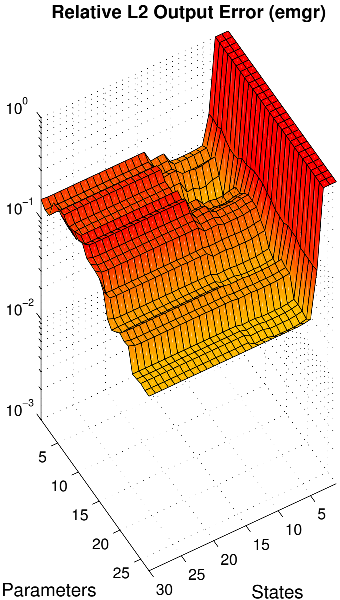
$$\ddot{x} = A_L x + \hat{A}_N(\theta)\dot{x} + Bu$$
$$y = C\dot{x}$$

[13] Kamrani et al 2012 (fMRI), Moran et al 2007 (EEG)

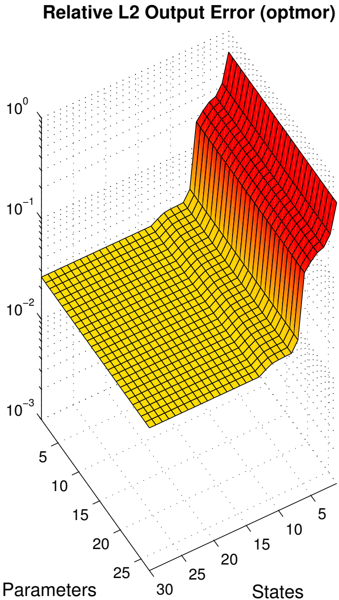
fMRI


Relative L2 Output Error (emgr)
10^0
10^−1
10^−2
10^−3
5
10
15
20
25
Parameters
25
20
15
10
5
States
Relative L2 Output Error (optmor)
10^0
10^−1
10^−2
10^−3
5
10
15
20
25
Parameters
25
20
15
10
5
States

EEG



Relative L2 Output Error (emgr)
10^0
10^{-1}
10^{-2}
10^{-3}
5
10
15
20
25
Parameters
30
25
20
15
10
5
States



Relative L2 Output Error (optmor)
10^0
10^{-1}
10^{-2}
10^{-3}
5
10
15
20
25
Parameters
30
25
20
15
10
5
States

tl;dl

Summary:

- Empirical Cross Gramian works in non-symmmetric setting
- Trust-Region and Data-Driven extensions prevent snapshots

- Empirical Gramians: parameter reduction error dominates
- Optimization-Based: state reduction error dominates

Source Code:

- `http://j.mp/gamm14`

Thanks!