Model Order Reduction for Nonlinear Systems

Lihong Feng

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

MAX-PLANCK-INSTITUT
DYNAMIK KOMPLEXER
TECHNISCHER SYSTEME
MAGDEBURG

Overview

- Linearization MOR.
- Qudratic MOR.
- Bilinearization MOR.
- Variational analysis MOR.
- Trajectory piece-wise linear MOR.
- Proper orthogonal decomposition (POD).
- References.

Linearization MOR

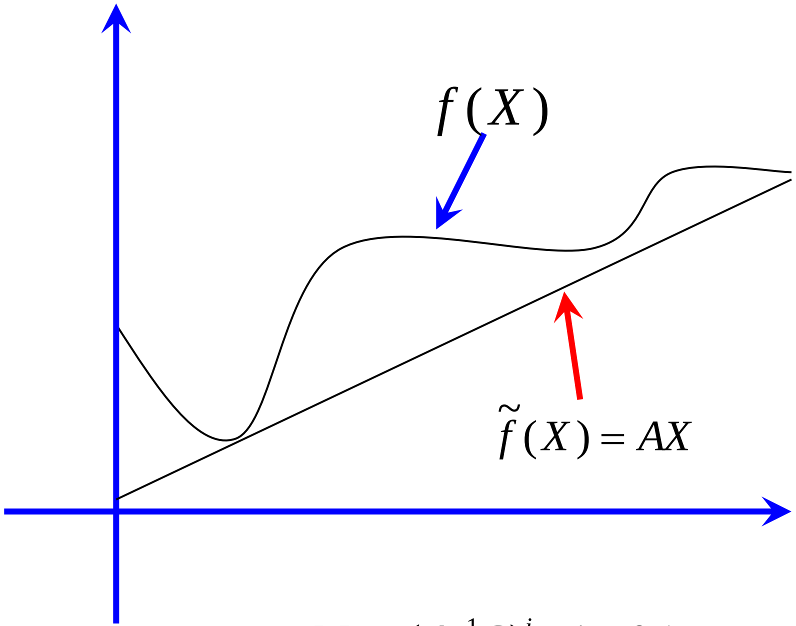
Original large ODE

$$CdX/dt = f(X) + Bu(t)$$
$$y(t) = LX(t)$$

Linearization: approximate $f(X)$ by a linear function

Taylor series expansion:

$$f(X) = f(X_0) + D_f(X - X_0) + \frac{1}{2}(X - X_0)^T H_f(X_0)(X - X_0) + \cdots$$
$$\approx f(X_0) + D_f(X - X_0)$$



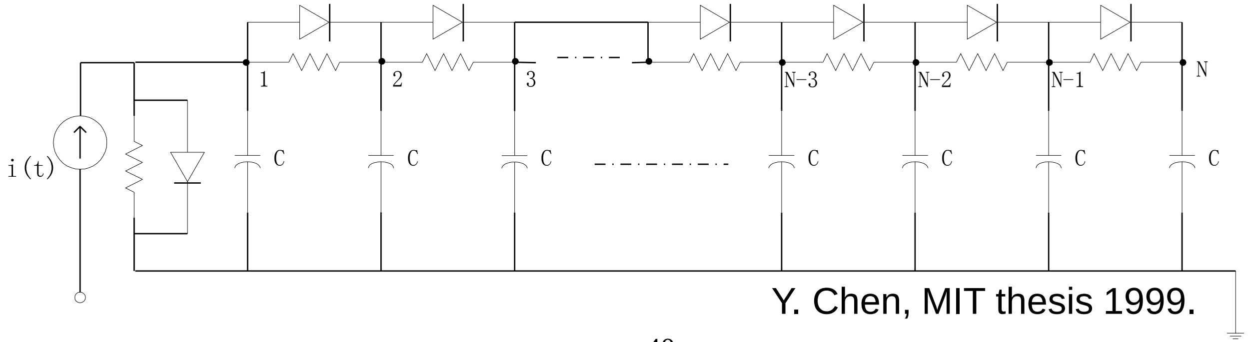
$$CdX/dt = f(X_0) + D_f(X - X_0) + Bu(t)$$
$$\tilde{y}(t) = LX(t)$$

$$CdX/dt = AX + \underbrace{[B, f(X_0) - D_f X_0]}_{\tilde{B}} \begin{pmatrix} u(t) \\ 1 \end{pmatrix}$$
$$\tilde{y}(t) = LX(t)$$

$$M_i = (A^{-1}C)^i r, i = 0,1,\ldots$$

$$V = orthogonalization\{r, M_1 r, M_2 r, \cdots M_j r\}$$
$$r = A^{-1}\tilde{B}, M_i = [(s_0 C - A)^{-1} C]^i r, i = 0,1,\ldots.$$

Example



Y. Chen, MIT thesis 1999.

$$g(x) = e^{40x} + x - 1$$

$$\frac{dX}{dt} = \begin{bmatrix} -g(x_1) - g(x_1 - x_2) \\ g(x_1 - x_2) - g(x_2 - x_3) \\ \vdots \\ g(x_{k-1} - x_k) - g(x_k - x_{k+1}) \\ \\ g(x_{n-1} - x_n) \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} u(t),$$

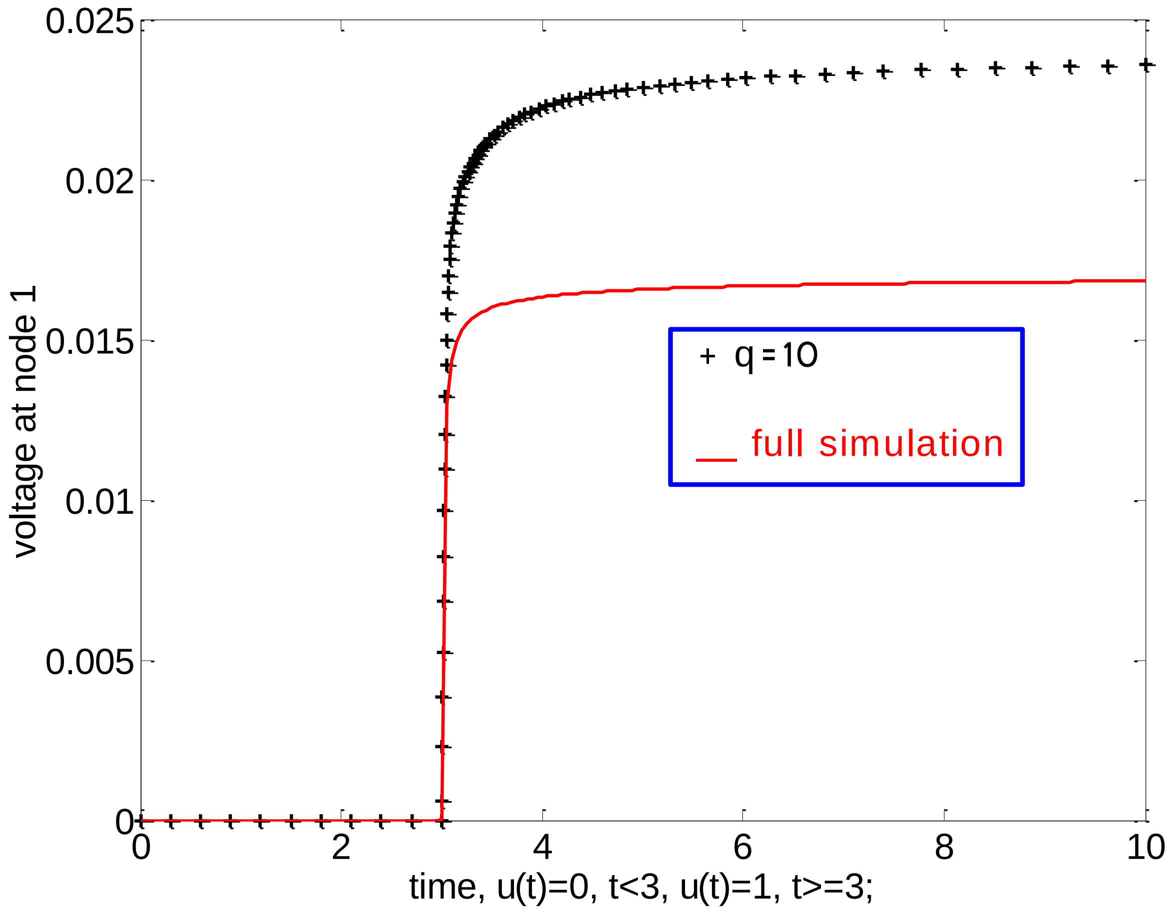
$$y(t) = LX(t)$$

$$g(x) = g(0) + g'(0)x + \frac{g''(0)}{2!}x^2 + \cdots$$
$$\approx g(0) + g'(0)x = 41x$$

$$\frac{dX}{dt} = \begin{bmatrix} -82x_1 + 42x_2 \\ 41x_1 - 82x_2 + 41x_3 \\ \vdots \\ 41x_{k-1} - 82x_k - 41x_{k+1} \\ \\ 41x_{n-1} - 41x_n \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} u(t),$$

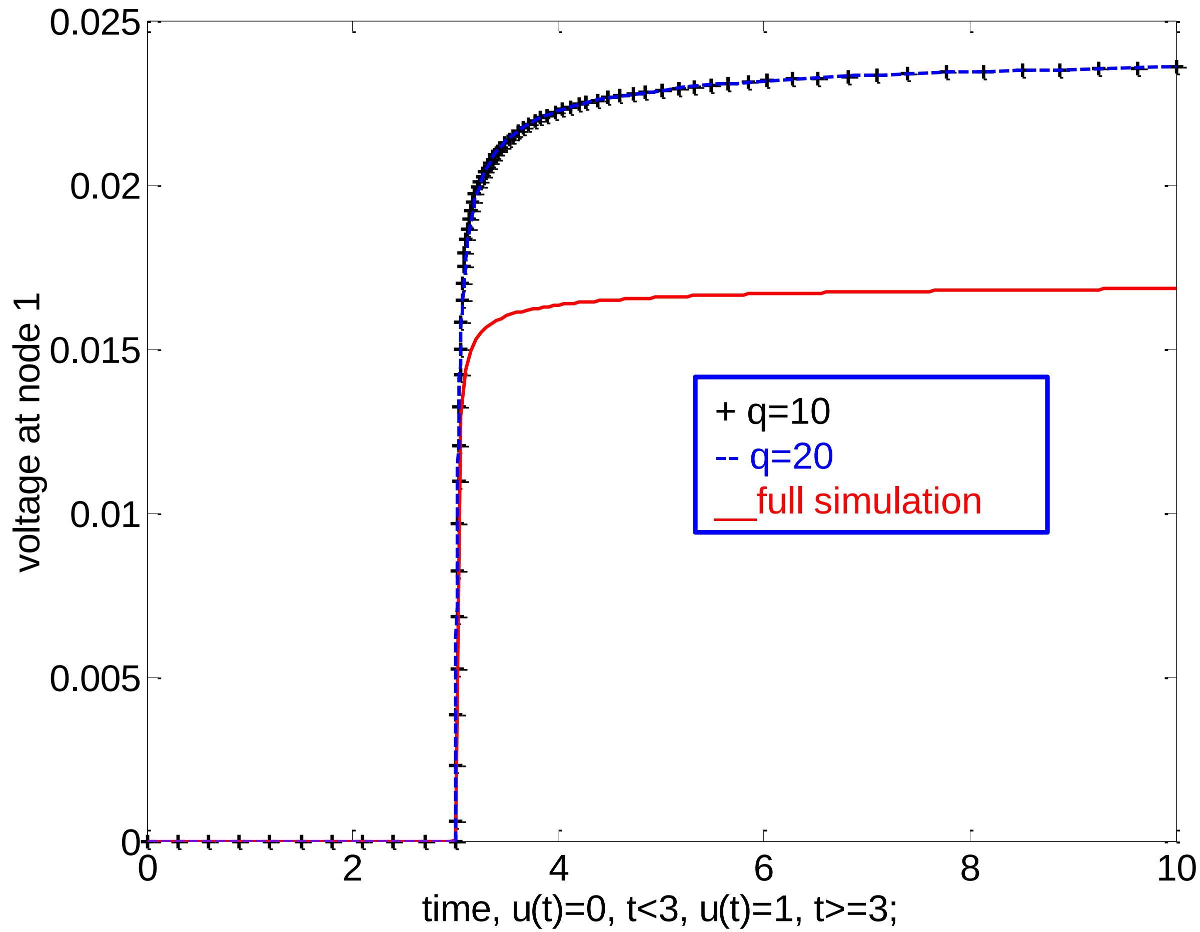
$$y(t) = LX(t)$$

Example

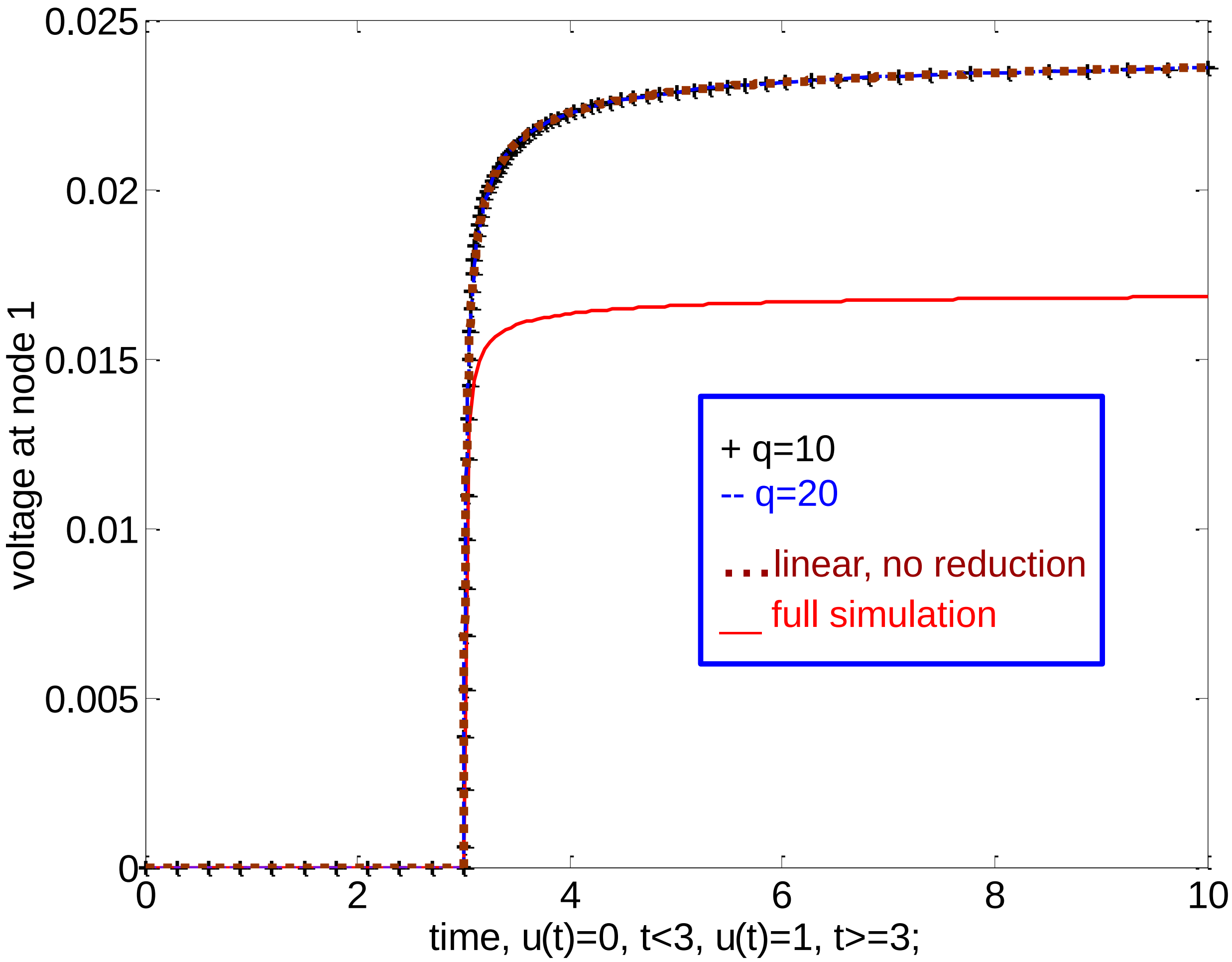


0.025
0.02
0.015
0.01
0.005
0
voltage at node 1
0
2
4
6
8
10
time, u(t)=0, t<3, u(t)=1, t>=3;
+ q=10
__ full simulation

Example


0.025
0.02
0.015
0.01
0.005
0
voltage at node 1
0
2
4
6
8
10
time, u(t)=0, t<3, u(t)=1, t>=3;
+ q=10
-- q=20
__full simulation

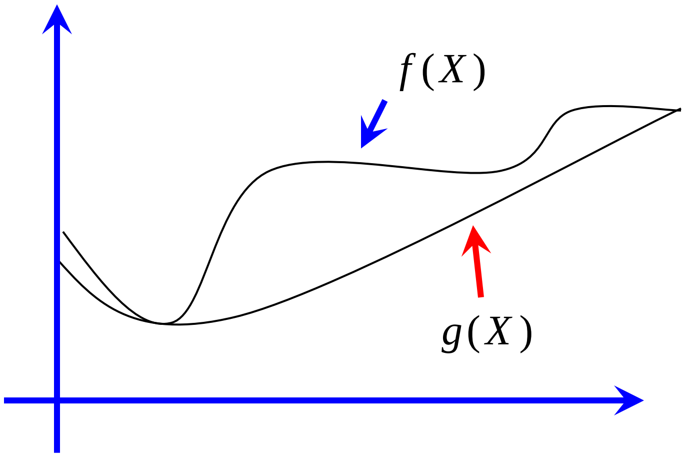
Example


0.025
0.02
0.015
0.01
0.005
0
voltage at node 1
0
2
4
6
8
10
time, u(t)=0, t<3, u(t)=1, t>=3;
+ q=10
-- q=20
... linear, no reduction
__ full simulation

Quadratic MOR

Approximate $f(X)$ by a quadratic polynomial $g(X)$

$$CdX/dt = f(X) + Bu(t)$$
$$y(t) = LX(t)$$



Taylor series expansion:

$$f(X) = f(X_0) + D_f(X - X_0) + \frac{1}{2}(X - X_0)^T H_f(X_0)(X - X_0) + \cdots$$
$$\approx f(X_0) + D_f(X - X_0) + \frac{1}{2}(X - X_0)^T H_f(X_0)(X - X_0)$$

$$CdX/dt = AX + X^T WX + \tilde{B}u(t)$$
$$\tilde{y}(t) = LX(t)$$

$X \approx VZ, Z \in R^q, q << n$

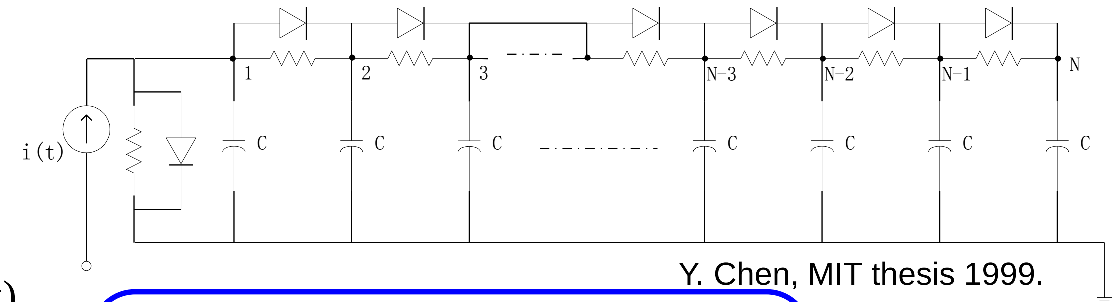
$$V^T CXdZ/dt = V^T AVZ + V^T Z^T V^T WVZ + V^T \tilde{B}u(t)$$
$$\hat{y}(t) = LVZ(t)$$

$$V = orthogonalization\{r, M_1 r, M_2 r, \cdots M_j r\}$$
$$r = (s_0 C - A)^{-1}\tilde{B}, M_i = [(s_0 C - A)^{-1} C]^i r, i = 0,1,\ldots.$$

Example

$$\frac{dX}{dt} = \begin{bmatrix} -g(x_1) - g(x_1 - x_2) \\ g(x_1 - x_2) - g(x_2 - x_3) \\ \vdots \\ g(x_{k-1} - x_k) - g(x_k - x_{k+1}) \\ \\ g(x_{n-1} - x_n) \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} u(t),$$

$$y(t) = LX(t)$$



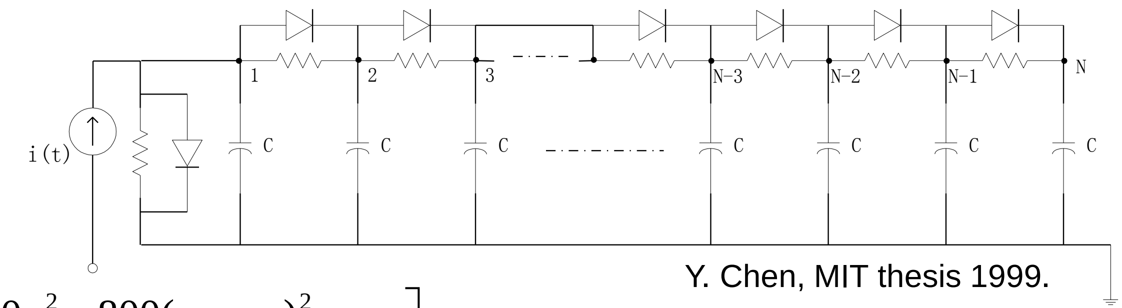
Y. Chen, MIT thesis 1999.

$$g(x) = e^{40x} + x - 1$$
$$g(x) = g(0) + g'(0)x + \frac{g''(0)}{2!}x^2 + \cdots$$
$$\approx g(0) + g'(0)x + \frac{g''(0)}{2!}x^2 = 41x + 800x^2$$

$$\frac{dX}{dt} = \begin{bmatrix} -82x_1 + 42x_2 \\ 41x_1 - 82x_2 + 41x_3 \\ \vdots \\ 41x_{k-1} - 82x_k - 41x_{k+1} \\ \\ 41x_{n-1} - 41x_n \end{bmatrix} + \begin{bmatrix} -800x_1^2 - 800(x_1 - x_2)^2 \\ -800(x_1 - x_2)^2 - 800(x_2 - x_3)^2 \\ \vdots \\ -800(x_{k-1} - x_k)^2 - 800(x_k - x_{k+1})^2 \\ \vdots \\ -800(x_{n-1} - x_n)^2 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} u(t),$$

$$y(t) = LX(t)$$

Example



Y. Chen, MIT thesis 1999.

$$\frac{dX}{dt} = \begin{bmatrix} -82x_1 + 42x_2 \\ 41x_1 - 82x_2 + 41x_3 \\ \vdots \\ 41x_{k-1} - 82x_k - 41x_{k+1} \\ \\ 41x_{n-1} - 41x_n \end{bmatrix} + \begin{bmatrix} -800x_1^2 - 800(x_1 - x_2)^2 \\ -800(x_1 - x_2)^2 - 800(x_2 - x_3)^2 \\ \vdots \\ -800(x_{k-1} - x_k)^2 - 800(x_k - x_{k+1})^2 \\ \vdots \\ -800(x_{n-1} - x_n)^2 \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} u(t),$$

$$y(t) = LX(t)$$

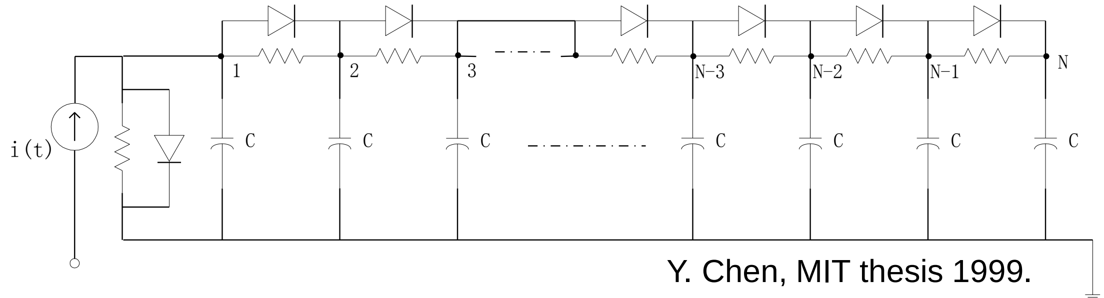
$$CdX/dt = AX + X^TWX + \tilde{B}u(t)$$
$$\tilde{y}(t) = LX(t)$$

W is a tensor, it has n matrices, the ith matrix corresponds to the ith element of the nonlinear vector.

Example

W

$$W^1 \in R^{n\times n} = \begin{pmatrix} -1600 & 800 & 0 & \cdots & 0 \\ 800 & -800 & 0 & \cdots & 0 \\ 0 & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & \cdots & 0 & 0 & 0 \end{pmatrix}$$



Y. Chen, MIT thesis 1999.

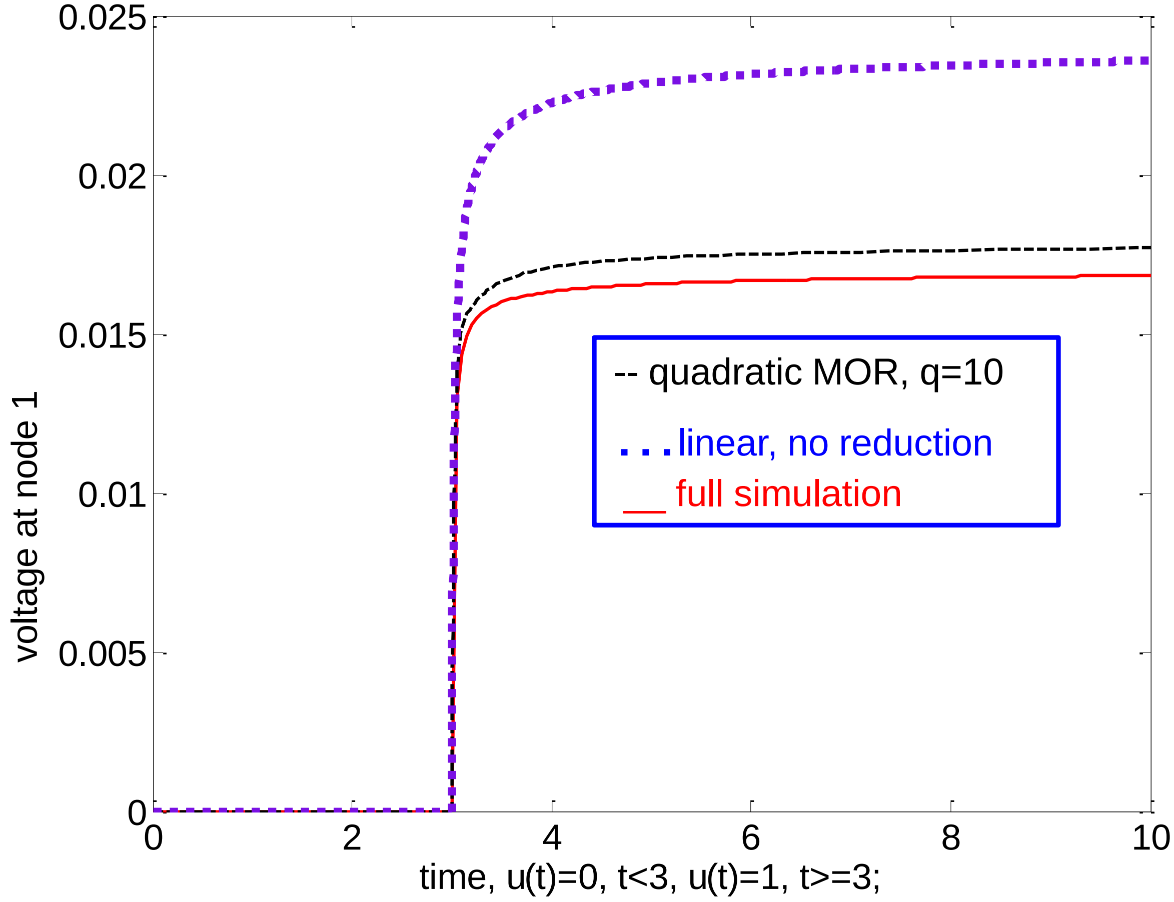
$$W^i \in R^{n\times n} = \begin{pmatrix} 0 & 0 & & 0 & \cdots & & & 0 \\ & \ddots & \vdots & & \vdots & & \ddots & \vdots \\ & & 800 & -800 & 0 & 0 & & 0 \\ \vdots & & -800 & 0 & 800 & 0 & & \\ & & 0 & 800 & -800 & 0 & & \\ & & 0 & 0 & 0 & 0 & & \\ \vdots & \ddots & \vdots & \vdots & \vdots & & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 0 & & 0 \end{pmatrix} \begin{matrix} \\ \\ i-1 \\ i \\ i+1 \\ \\ \\ \\ \end{matrix}$$

$$i-1, \quad i, \quad i+1$$

$$W^n \in R^{n\times n} = \begin{pmatrix} 0 & 0 & 0 & \cdots & 0 \\ 0 & \ddots & 0 & \cdots & \vdots \\ 0 & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & 800 & -800 \\ 0 & \cdots & 0 & -800 & 800 \end{pmatrix}$$

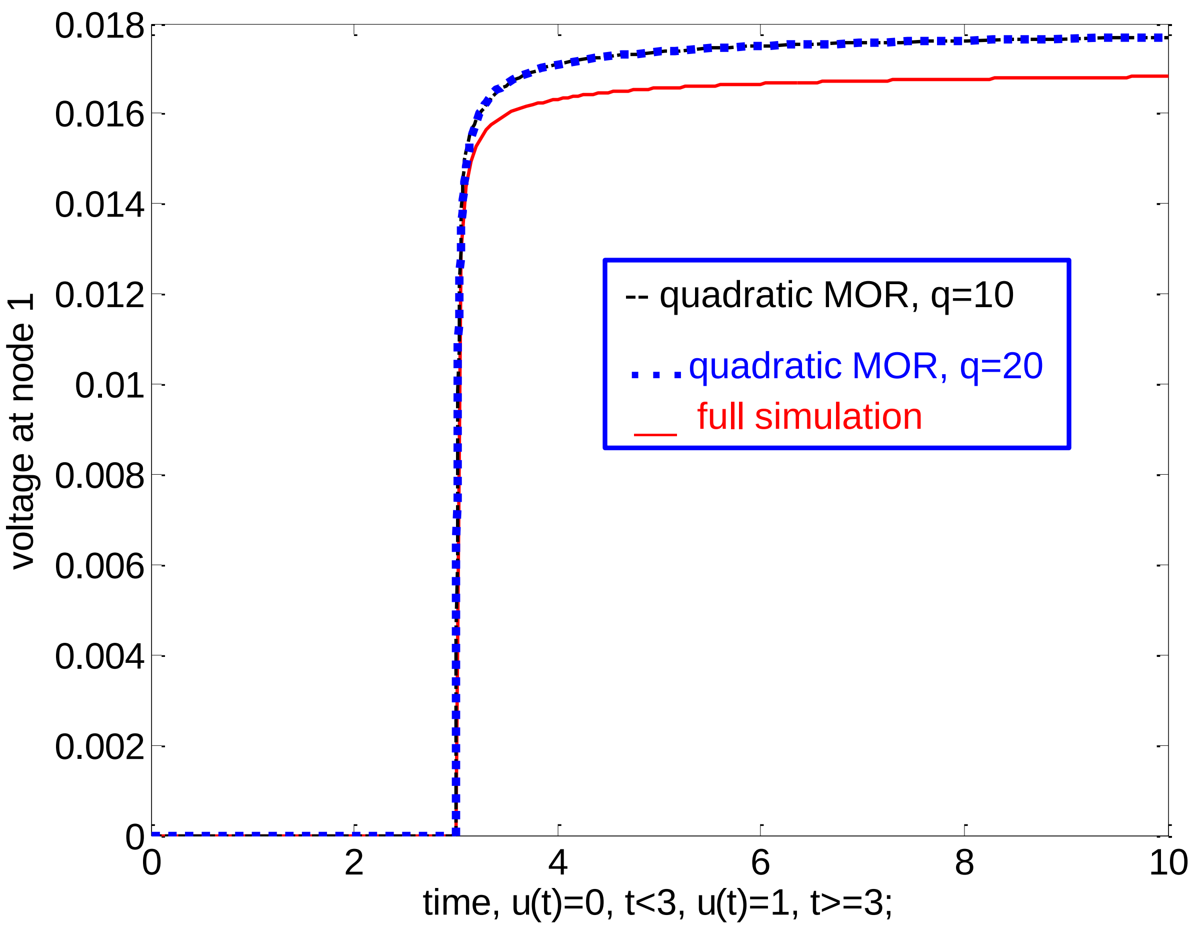
$$X^T W X = \begin{bmatrix} X^T W^1 X \\ \vdots \\ X^T W^i X \\ \vdots \\ X^T W^n X \end{bmatrix} \longrightarrow Z^T V^T W V Z = \begin{bmatrix} Z^T V^T W^1 V Z \\ \vdots \\ Z^T V^T W^i V Z \\ \vdots \\ Z^T V^T W^n V Z \end{bmatrix}$$

Example



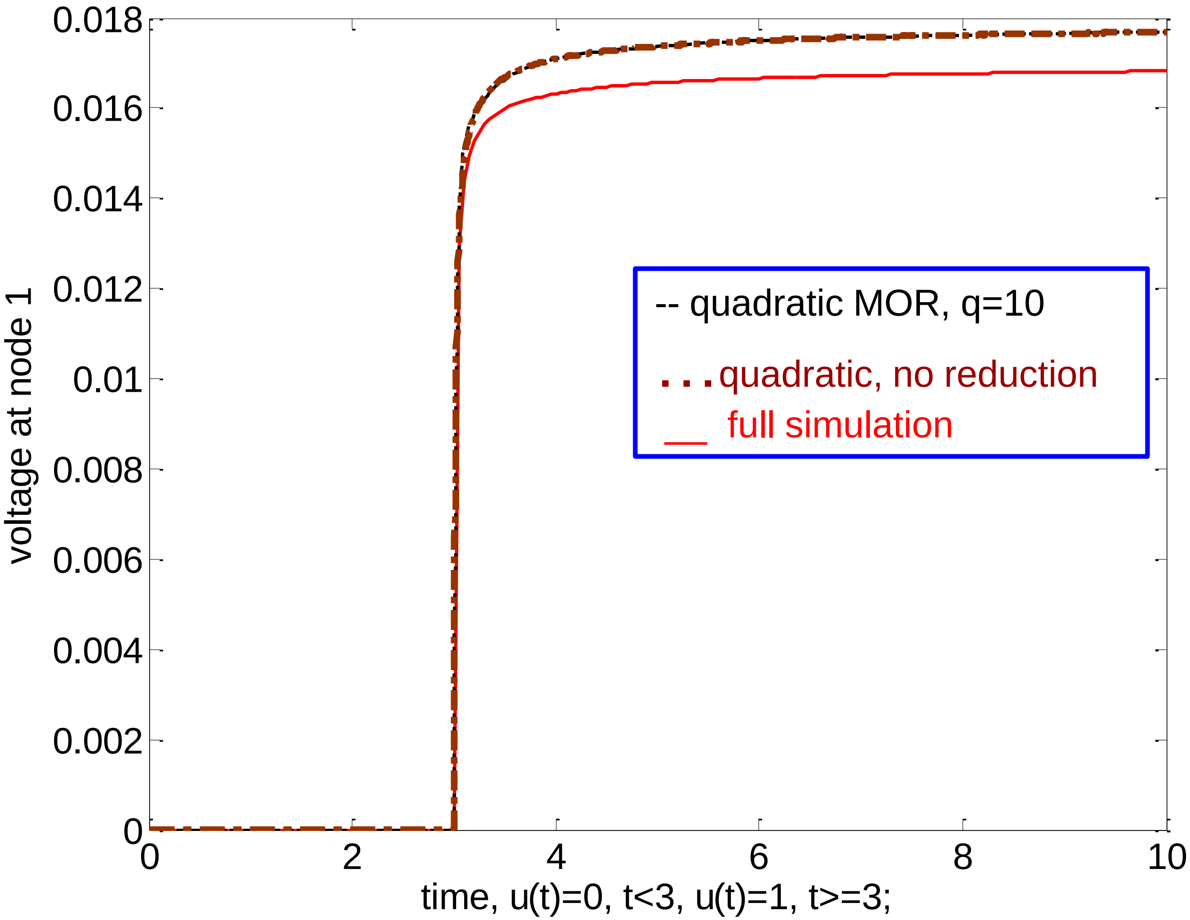
0.025
0.02
0.015
0.01
0.005
0
0
2
4
6
8
10
voltage at node 1
time, u(t)=0, t<3, u(t)=1, t>=3;
-- quadratic MOR, q=10
... linear, no reduction
__ full simulation

Example



0.018
0.016
0.014
0.012
0.01
0.008
0.006
0.004
0.002
0
voltage at node 1
0
2
4
6
8
10
time, u(t)=0, t<3, u(t)=1, t>=3;
-- quadratic MOR, q=10
...quadratic MOR, q=20
__ full simulation

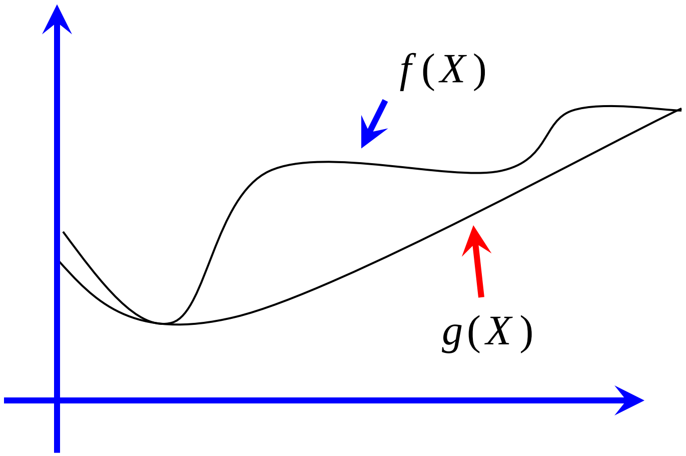
Example


0.018
0.016
0.014
0.012
0.01
0.008
0.006
0.004
0.002
0
voltage at node 1
0
2
4
6
8
10
time, u(t)=0, t<3, u(t)=1, t>=3;
-- quadratic MOR, q=10
...quadratic, no reduction
__ full simulation

Bilinearization MOR

Approximate $f(X)$ by quadratic polynomial $g(X)$, but written into Kronecker product

$$dX/dt = f(X) + Bu(t)$$
$$y(t) = LX(t)$$



Taylor series expansion:

$$f(X) = f(X_0) + D_f(X - X_0) + \frac{1}{2}(X - X_0)^T H_f(X_0)(X - X_0) + \cdots$$
$$\approx f(X_0) + D_f(X - X_0) + \frac{1}{2}(X - X_0)^T H_f(X_0)(X - X_0)$$
$$= f(X_0) + A_1 X + A_2 X \otimes X$$

$$dX^{\otimes}/dt = A^{\otimes} X^{\otimes} + N^{\otimes} X^{\otimes} u(t) + B^{\otimes} u(t)$$
$$y(t) = L^{\otimes} X^{\otimes}(t)$$

$X^{\otimes} \in R^N, N \approx n^2$

$X^{\otimes} \approx VZ, Z \in R^q, q << n$

$$dZ/dt = V^T A^{\otimes} VZ + V^T N^{\otimes} VZu(t) + V^T B^{\otimes} u(t)$$
$$\hat{y}(t) = LVZ(t)$$

Bilinearization MOR

$$dX/dt = f(X) + Bu(t)$$
$$y(t) = LX(t)$$

Carleman bilinearization

$$dX^{\otimes}/dt = A^{\otimes}X^{\otimes} + N^{\otimes}X^{\otimes}u(t) + B^{\otimes}u(t)$$
$$y(t) = L^{\otimes}X^{\otimes}(t)$$

$$A^{\oplus} = \begin{pmatrix} A_1 & A_2 \\ 0 & A_1 \otimes I + I \otimes A_1 \end{pmatrix} \qquad N^{\oplus} = \begin{pmatrix} 0 & 0 \\ B \otimes I + I \otimes B & 0 \end{pmatrix}$$

$$X^{\otimes} = \begin{pmatrix} X \\ X \otimes X \end{pmatrix} \qquad B^{\otimes} = \begin{pmatrix} B \\ 0 \end{pmatrix} \qquad L^{\otimes} = \begin{bmatrix} L & 0 \end{bmatrix}$$

Kronecker product

$$A \otimes B = \begin{pmatrix} a_{11}B & \cdots & a_{1n}B \\ \vdots & \ddots & \vdots \\ a_{m1}B & \cdots & a_{mn}B \end{pmatrix}$$

Carleman bilinearization:

[1] W.J. Rugh, Nolinear System Theory, The John Hopkins University Press, Boltimore, 1981.

[2] S. Sastry, Nonlinear Systems: Analysis, Stability and Control, Springer, New York, 1999.

Bilinearization MOR

How to compute V?

$$dX^{\otimes}/dt = A^{\otimes}X^{\otimes} + N^{\otimes}X^{\otimes}u(t) + B^{\otimes}u(t)$$
$$y(t) = L^{\otimes}X^{\otimes}(t)$$

Volterra series expression of bilinear system

According to the theory in [Rugh 1981], the output response of the bilinear system can be expressed into Volterra series,

$$y(t) = \sum_{n=1}^{\infty} y_n(t)$$

$$y_n(t) = \int_0^t h_n^{(reg)}(t_1, t_2, \cdots, t_n) u(t - t_1 - t_2 - \cdots - t_n) \cdots u(t - t_n) dt_1 \cdots dt_n$$

$$h_n^{(reg)}(t_1, t_2, \cdots, t_n) = L^{\otimes T} e^{A^{\otimes} t_n} N^{\otimes} e^{A^{\otimes} t_{n-1}} \cdots N^{\otimes} e^{A^{\otimes} t_1} B^{\otimes}$$

Laplace transform (drop $\otimes$ for simplicity):

$$h_n^{(reg)}(s_1, s_2, \cdots, s_n) = L^T (s_n I - A)^{-1} N (s_{n-1} I - A)^{-1} N \cdots (s_2 I - A)^{-1} N (s_1 I - A)^{-1} B$$
$$= (-1)^n L^T (I - s_n A^{-1})^{-1} A^{-1} N (I - s_{n-1} A^{-1})^{-1} A^{-1} N \cdots (I - s_2 A^{-1})^{-1} A^{-1} N (I - s_1 A^{-1})^{-1} A^{-1} B$$

Bilinearization MOR

How to compute V?

$$dX^{\otimes}/dt = A^{\otimes}X^{\otimes} + N^{\otimes}X^{\otimes}u(t) + B^{\otimes}u(t)$$

$$y(t) = L^{\otimes}X^{\otimes}(t)$$

Laplace transform:

$$h_n^{(reg)}(s_1, s_2, \cdots, s_n) = L(s_n I - A)^{-1} N(s_{n-1}I - A)^{-1} N \cdots (s_2 I - A)^{-1} N(s_1 I - A)^{-1} B$$
$$= (-1)^n L^T (I - s_n A^{-1})^{-1} A^{-1} N (I - s_{n-1}A^{-1})^{-1} A^{-1} N \cdots (I - s_2 A^{-1})^{-1} A^{-1} N (I - s_1 A^{-1})^{-1} A^{-1} B$$

$$(I - s_n A^{-1})^{-1} = I + A^{-1}s_n + \cdots + A^{-i}s_n^i + \cdots$$

$$h_n^{(reg)}(s_1, s_2, \cdots, s_n) = \sum_{l_n=1}^{\infty} \cdots \sum_{l_1=1}^{\infty} (-1)^n s_n^{l_n - 1} \cdots s_1^{l_1 - 1} \underline{LA^{-l_n} NA^{-l_{n-1}} N \cdots A^{-l_1} B}$$

Multimoments:

$$m(l_n, \cdots, l_1) = (-1)^n LA^{-l_n} NA^{-l_{n-1}} N \cdots A^{-l_1} B$$

Bilinearization MOR

How to compute V?

$$dX^{\otimes}/dt = A^{\otimes}X^{\otimes} + N^{\otimes}X^{\otimes}u(t) + B^{\otimes}u(t)$$
$$y(t) = L^{\otimes}X^{\otimes}(t)$$

$$h_n^{(reg)}(s_1, s_2, \cdots, s_n) = \sum_{l_n=1}^{\infty} \cdots \sum_{l_1=1}^{\infty} (-1)^n s_n^{l_n-1} \cdots s_1^{l_1-1} \underline{LA^{-l_n} NA^{-l_{n-1}} N \cdots A^{-l_1} B}$$

Multimoments:

$$m(l_n, \cdots, l_1) = (-1)^n LA^{-l_n} NA^{-l_{n-1}} N \cdots A^{-l_1} B$$

$$range\{V_1\} = K_{q_1}\{A^{-1}, A^{-1}b\} = span\{A^{-1}B, \ldots, A^{-q_1}B\}$$

$$\vdots$$

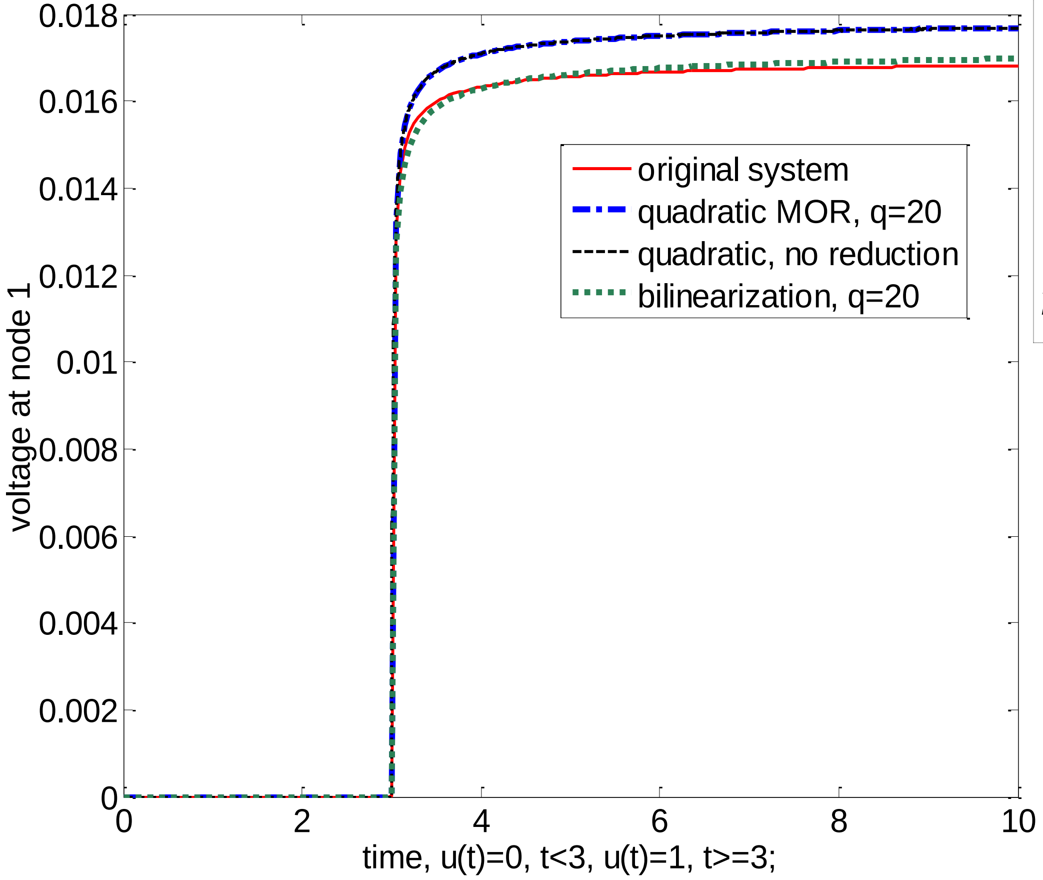
$$range\{V_j\} = K_{q_j}\{A^{-1}, A^{-1}NV_{j-1}\} = \{A^{-1}NV_{j-1,} A^{-1}NV_{j-1}, \ldots, A^{-q_j}NV_{j-1}\}$$

$$range\{V\} = colspan\{V_1, \ldots, V_J\}$$

Reduced model:

$$dZ/dt = V^T A^{\otimes} VZ + V^T N^{\otimes} VZu(t) + V^T B^{\otimes} u(t)$$
$$\hat{y}(t) = LVZ(t)$$

Example



V for bilinearization MOR:

$$range\{V\} = colspan\{V_1, V_2\}$$

$$range\{V_1\} = span\{(A^{\otimes})^{-1}B, \ldots, (A^{\otimes})^{-19}B\}$$

$$range\{V_2\} = span\{(A^{\otimes})^{-1}NV_1(:,1)\}$$

V for quadratic MOR:

$$range\{V\} = span\{A^{-1}B, \ldots, A^{-20}B\}$$

Variational analysis MOR

Original system:

$$dX/dt = f(X) + Bu(t)$$
$$y(t) = LX(t)$$

Taylor series expansion:

$$f(X) = f(X_0) + D_f(X - X_0) + \frac{1}{2}(X - X_0)^T H_f(X_0)(X - X) + \cdots$$
$$\approx f(X_0) + A_1 X + A_2 X \otimes X + A_3 X \otimes X \otimes X + \cdots$$

$$dX/dt = A_1 X + A_2 X \otimes X + \tilde{B}\tilde{u}(t)$$
$$y(t) = LX(t)$$

or

$$dX/dt = A_1 X + A_2 X \otimes X + A_3 X \otimes X \otimes X + \tilde{B}\tilde{u}(t)$$
$$y(t) = LX(t)$$

Variational analysis:

$$dX/dt = A_1 X + A_2 X \otimes X + \tilde{B}\alpha\tilde{u}(t)$$
$$y(t) = LX(t)$$

or

$$dX/dt = A_1 X + A_2 X \otimes X + A_3 X \otimes X \otimes X + \tilde{B}\alpha\tilde{u}(t)$$
$$y(t) = LX(t)$$

$$\Longrightarrow \quad X(\alpha, t) = \alpha X_1(t) + a^2 X_2(t) + \alpha^3 X_3(t) + \cdots$$

Variational analysis MOR

Variational analysis [11]:

Assume : $X(t)=0$, if $u(t)=0$

$$dX/dt = A_1 X + A_2 X\otimes X + A_3 X\otimes X\otimes X + \tilde{B}\alpha\tilde{u}(t)$$
$$y(t) = LX(t)$$

$$\Rightarrow \quad X(t) = \alpha X_1(t) + a^2 X_2(t) + \alpha^3 X_3(t) + \cdots$$

$$\Downarrow$$

$$d(\alpha X_1 + \alpha^2 X_2 + \alpha^3 X_3 + \cdots)/dt = A_1(\alpha X_1 + \alpha^2 X_2 + \alpha^3 X_3 + \cdots)$$
$$+ A_2[(\alpha X_1 + \alpha^2 X_2 + \alpha^3 X_3 + \cdots)\otimes(\alpha X_1 + \alpha^2 X_2 + \alpha^3 X_3 + \cdots)]$$
$$+ A_3[(\alpha X_1 + \alpha^2 X_2 + \alpha^3 X_3 + \cdots)\otimes(\alpha X_1 + \alpha^2 X_2 + \alpha^3 X_3 + \cdots)\otimes(\alpha X_1 + \alpha^2 X_2 + \alpha^3 X_3 + \cdots)] + \tilde{B}\alpha\tilde{u}(t)$$
$$y(t) = LX(t)$$

$$\Downarrow$$

α : $$dX_1(t)/dt = A_1 X_1(t) + \tilde{B}\tilde{u}(t)$$

α^2 : $$dX_2(t)/dt = A_1 X_2(t) + A_2(X_1\otimes X_1)$$

α^3 : $$dX_3(t)/dt = A_1 X_3(t) + A_2(X_1\otimes X_2 + X_2\otimes X_1) + A_3(X_1\otimes X_1\otimes X_1)$$

$$\vdots$$

Variational analysis MOR

Variational analysis:

$$dX/dt = A_1 X + A_2 X \otimes X + A_3 X \otimes X \otimes X + \tilde{B}\alpha\tilde{u}(t)$$
$$y(t) = LX(t)$$

$$\Longrightarrow \quad X(t) = \alpha X_1(t) + a^2 X_2(t) + \alpha^3 X_3(t) + \cdots$$

$$\alpha: \quad dX_1(t)/dt = A_1 X_1(t) + \tilde{B}\tilde{u}(t)$$

$$\alpha^2: \quad dX_2(t)/dt = A_1 X_2(t) + A_2(X_1 \otimes X_1)$$

$$\alpha^3: \quad dX_3(t)/dt = A_1 X_3(t) + A_2(X_1 \otimes X_2 + X_2 \otimes X_1) + A_3(X_1 \otimes X_1 \otimes X_1)$$

$$\vdots$$

$$X_1 \approx V_1 Z_1 \qquad V_1 = span\{A_1^{-1}\tilde{B}, \cdots, A_1^{-q_1}\tilde{B}\}$$

$$X_2 \approx V_2 Z_2 \qquad V_2 = span\{A_1^{-1}A_2, \cdots, A_1^{-q_2}A_2\}$$

$$X_3 \approx V_3 Z_3 \qquad V_3 = span\{A_1^{-1}[A_2, A_3], \cdots, A_1^{-q_2}[A_2, A_3]\}$$

$$\Longrightarrow$$

$$X(t) = \alpha X_1(t) + a^2 X_2(t) + \alpha^3 X_3(t) + \cdots$$
$$\approx \alpha X_1 + \alpha^2 X_2 + \alpha^3 X_3$$
$$\approx \alpha V_1 Z_1 + \alpha^2 V_2 Z_2 + \alpha^3 V_3 Z_3$$
$$\Updownarrow$$
$$X(t) \approx\in span\{V_1, V_2, V_3\}$$

Variational analysis MOR

Original system:

$$dX/dt = f(X) + Bu(t)$$
$$y(t) = LX(t)$$

$\approx$

$$dX/dt = A_1 X + A_2 X \otimes X + A_3 X \otimes X \otimes X + \tilde{B}\tilde{u}(t)$$
$$y(t) = LX(t)$$

$\Downarrow$

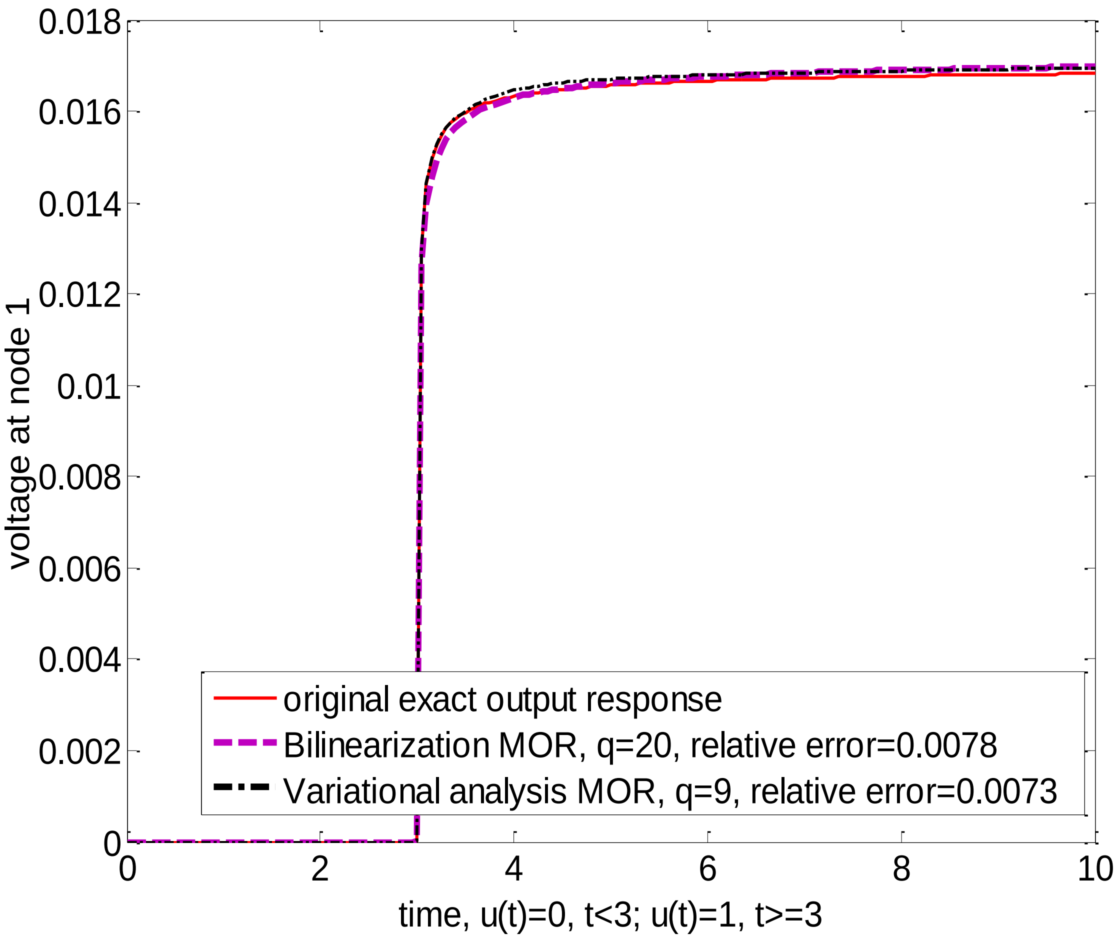
$$X(t) \approx\in span\{V_1, V_2, V_3\}$$

Compute *V*: $range(V) = span\{V_1, V_2, V_3\}$ $\quad X(t) \approx VZ$

Reduced model: $dZ/dt = V^T A_1 VZ + V^T A_2 VZ \otimes VZ + V^T A_3 VZ \otimes VZ \otimes VZ + V^T \tilde{B}\tilde{u}(t)$

$$\hat{y}(t) = LVZ(t)$$

Example



V for bilinearization MOR:

$$range\{V\} = colspan\{V_1, V_2\}$$

$$range\{V_1\} = span\{A^{-1}B, \ldots, A^{-19}B\}$$

$$range\{V_2\} = span\{A^{-1}NV(:,1)\}$$

V for Variational analysis MOR:

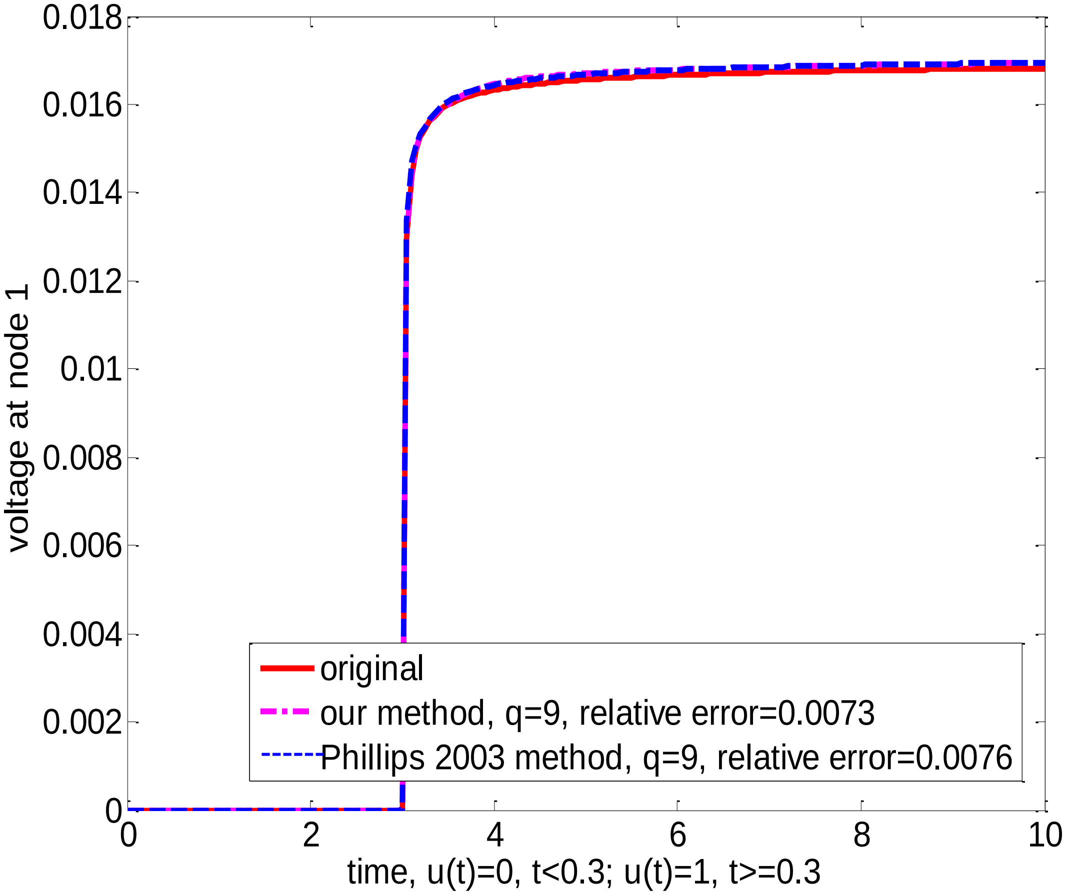
$$range\{V_1\} = span\{A_1^{-1}B, \ldots, A^{-4}B\}$$

$$V_2^0 = orth\{A_2\}$$

$$range\{V_2\} = span\{A_1^{-1}V_2^0(:,1:6)\}$$

$$range\{V\} = span\{V_1, V_2\}$$

Example



V for [Phillips 2000] method:

$$range\{V_1\} = span\{A_1^{-1}B, \ldots, A^{-4}B\}$$
$$V_2^0 = A^{-1}A_2(V_1 \otimes V_1)$$
$$range\{V_2\} = span\{V_2^0(:,1:6)\}$$
$$range\{V\} = span\{V_1, V_2\}$$

V for our method [Feng 2014]:

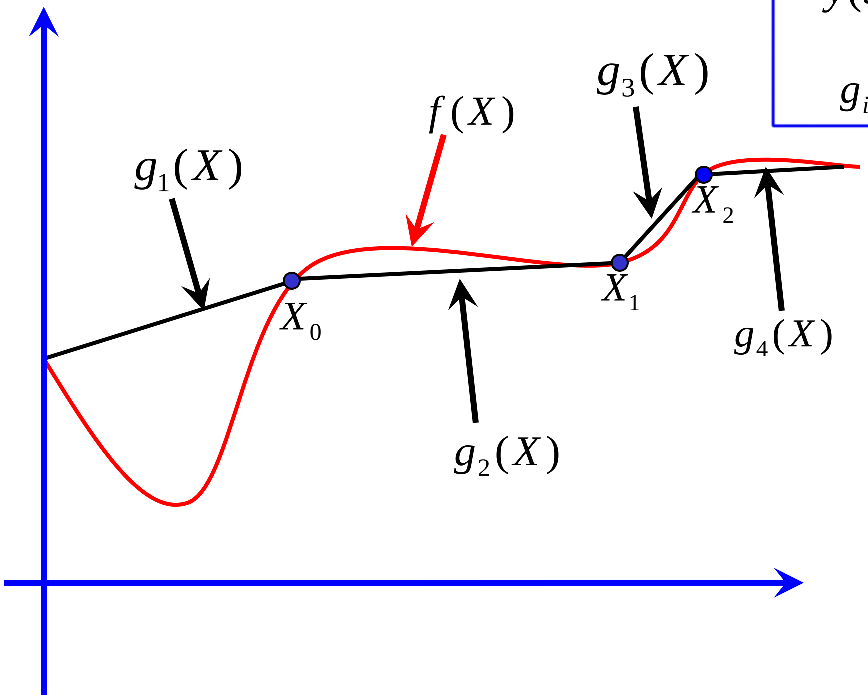
$$range\{V_1\} = span\{A_1^{-1}B, \ldots, A^{-4}B\}$$
$$V_2^0 = orth\{A_2\}$$
$$range\{V_2\} = span\{A_1^{-1}[V_2^0(:,1:6)]\}$$
$$range\{V\} = span\{V_1, V_2\}$$

Trajectory piece-wise linear MOR

Original system:

$$dX/dt = f(X) + Bu(t)$$
$$y(t) = LX(t)$$

$$dX/dt = \sum_{i=0}^{s-1} \tilde{w}_i g_i(X) + Bu,$$
$$y(t) = LX(t)$$
$$g_i(X) = f(X_i) + A_i(X - X_i), \quad i = 0,1,\ldots,s-1$$



$$A_i = (a_{jk}), \qquad a_{jk} = \left.\frac{\partial f_j(X)}{\partial x_k}\right|_{X_i}$$

Trajectory piece-wise linear MOR

Original system:

$$dX/dt = f(X) + Bu(t)$$
$$y(t) = LX(t)$$

$$dX/dt = \sum_{i=0}^{s-1} g_i(X) + Bu,$$
$$y(t) = LX(t)$$
$$g_i(X) = \tilde{w}_i(X) f(X_i) + \tilde{w}_i(X) A_i (X - X_i), \quad i = 0,1,\ldots,s-1$$
$$= \tilde{w}_i A_i X + \tilde{w}_i (f(X_i) - A_i X_i)$$

$$dX/dt = \sum_{i=0}^{s-1} (\tilde{w}_i A_i X + B_0 \tilde{w}_i) + Bu(t),$$
$$y(t) = LX(t)$$
$$\tilde{B} = [B, B_0], B_0 = f(X_i) - AX_i$$

How to compute V?

$$range\{V_i\} = span\{A_i^{-1}\tilde{B}_i, \ldots, A_i^{-q_i}\tilde{B}_i\} \qquad i = 0,1,\ldots,s-1$$
$$range\{V\} = span\{V_1, \ldots, V_{s-1}\}$$

Trajectory piece-wise linear MOR

Original system:

$$dX/dt = f(X) + Bu(t)$$
$$y(t) = LX(t)$$

Trajectory piece-wise linear system:

$$dX/dt = \sum_{i=0}^{s-1} (\tilde{w}_i A_i X + B_0 \tilde{w}_i) + Bu(t),$$
$$y(t) = LX(t)$$

Reduced model:

$$dZ/dt = \sum_{i=0}^{s-1} (\tilde{w}_i V^T A_i VZ + V^T B_0 \tilde{w}_i + V^T Bu(t),$$
$$\hat{y}(t) = LVZ(t)$$

Proper orthogonal decomposition (POD)

MAX-PLANCK-GESELLSCHAFT

POD and SVD

SVD: For any matrix $Y \in R^{m\times n}$, there exist $U = (u_1,\dots,u_m) \in R^{m\times m}$ and $V = (v_1,\dots,v_n) \in R^{n\times n}$, s.t.

$$Y = U\Sigma V^T \quad \text{or} \quad U^T Y V = \begin{pmatrix} D & 0 \\ 0 & 0 \end{pmatrix} := \Sigma \in R^{m\times n}$$

Here, $D = diag(\sigma_1,\dots,\sigma_d)$. Let U^d and V^d be the matrices including the first d columns of U and V respectively.

It is obvious, $Y = (y_1,\dots,y_n) = U^d D (V^d)^T$

$$\Rightarrow y_j = \sum_{i=1}^{d} u_i (D(V^d)^T)_{ij} = \sum_{i=1}^{d} (D(V^d)^T)_{ij} u_i = \sum_{i=1}^{d} ((U^d)^T U^d D (V^d)^T)_{ij} u_i$$

$$= \sum_{i=1}^{d} ((U^d)^T Y)_{ij} u_i = \sum_{i=1}^{d} \left(\sum_{k=1}^{m} U^d_{ki} Y_{kj} \right) u_i = \sum_{i=1}^{d} \left\langle y_j, u_i \right\rangle_{R^m} u_i = \sum_{i=1}^{d} \left\langle u_i, y_j \right\rangle_{R^m} u_i.$$

Y can be represented in terms of *d* linearly independent columns of U^d .

Proper orthogonal decomposition (POD)

Definition For $l \in \{1, \ldots, d\}$ the vectors $\{u_i\}_{i=1}^{l}$ are called POD basis of rank l.

The POD basis $\{u_i\}_{i=1}^{l}$ is optimal, among all rank l approximations, in approximating the columns of Y:

$$\{u_i\}_{i=1}^{l} = \arg \min_{\tilde{u}_1, \ldots, \tilde{u}_l \in R^m} \sum_{j=1}^{n} \varepsilon_j \qquad \text{s.t.} \left\langle \tilde{u}_i, \tilde{u}_j \right\rangle_{R^m} = \delta_{ij}, 1 \le i, j \le l.$$

Here, $\varepsilon_j = \| y_j - \sum_{i=1}^{l} \left\langle y_j, \tilde{u}_i \right\rangle_{R^m} \tilde{u}_i \|_{R^m}^2$

Model Order Reduction using POD

Algorithm MOR using POD

1. Solve the original nonlinear system to get the snapshots

$$X = (x_{t_1}, \ldots x_{t_m})$$

2. Get the POD vectors of rank q from SVD of X

$$X = \tilde{U}\Sigma\tilde{V}^T, V = (\tilde{u}_1, \ldots, \tilde{u}_q)$$

3. Use V to get the ROM

$$V^T E V \frac{dz(t)}{dt} = V^T f(Vz(t)) + V^T Bu(t)$$

How to deal with $f(Vz(t))$?

An effective way is to approximate the nonlinear function by projecting it onto a subspace that approximates the space generated by the nonlinear function, and with dimension $l << n$.

$$f(t) \approx Uc(t), U = (u_1, \ldots, u_l)$$

DEIM

MAX-PLANCK-GESELLSCHAFT

To determine $c(t)$, we select m distinguished rows from the overdetermined system

$$f(t) = Uc(t).$$

In particular, consider a matrix

$$P = [e_{\wp_1}, \ldots, e_{\wp_l}] \in R^{n \times m},$$

Suppose $P^T U$ is nonsingular, then

$$P^T f(t) = P^T U c(t) \Rightarrow c(t) = (P^T U)^{-1} P^T f(t)$$

so that,

$$f(t) \approx Uc(t) = U(P^T U)^{-1} P^T f(t).$$

How to compute U and how to specify the indices $\wp_i, i = 1, \ldots, l$?

Compute U:

1. Collect the snapshots of $f(x(t))$ into a matrix $F = (f(x_{t_1}), \ldots, f(x_{t_m}))$.
2. Apply SVD to F: $F = U^F \Sigma (V^F)^T$
3. $U = (u_1^F, \ldots, u_l^F)$.

DEIM

Using DEIM to decide the indices:

Algorithm Discrete Empirical Interpolation Method (DEIM)

$$\text{Input}: \text{POD basis} \left\{ u_i^F \right\}_{i=1}^{l} \text{ for F}$$

$$\text{Output}: \vec{\wp} = [\wp_1, \ldots \wp_l]^T \in R^l$$

$$1. [|\rho|, \wp_1] = \max\left\{ |u_1^F| \right\}$$

$$2. U = [u_1^F], P = [e_{\wp_1}], \vec{\wp} = [\wp_1]$$

$$3. \text{ for } i = 2 \text{ to } l \text{ do}$$

$$4. \text{ Solve } (P^T U)\alpha = P^T u_i^F \text{ for } \alpha, \text{ where } \alpha = (\alpha_1, \ldots, \alpha_{i-1})^{\mathrm{T}}$$

$$5. r = u_i^F - U\alpha$$

$$6. [|\rho|, \wp_l] = \max\{|r|\}$$

$$7. U \leftarrow [U\ u_i^F], P \leftarrow [P\ e_{\wp_i}], \vec{\wp} \leftarrow \begin{bmatrix} \vec{\wp} \\ \wp_i \end{bmatrix}$$

$$8. \text{ end for}$$

DEIM

Come back to $V^T f(Vz(t))$:

$$f(Vz(t)) \approx U(P^T U)^{-1} P^T f(Vz(t)).$$

If $f(x(t)) = (f_1(x_1(t)), \ldots, f_n(x_n(t)))$, then

$$P^T f(Vz(t)) = f(P^T Vz(t))$$

so that

$$V^T f(Vz(t)) \approx \underline{V^T U(P^T U)^{-1}} f(P^T Vz(t))$$

can be precomputed before solving the ROM

Computation of $V^T f(Vz(t))$ during solving ROM is independent of n.

If $f(x(t))$ is not componentwisely evaluated as above, but each entry f_i only depends on a few entries of $x(t)$, then computation of $V^T f(Vz(t))$ during solving ROM is still independent of n.

References

Quadratic MOR:

[1] Yong Chen, “Model order reduction for nonlinear systems”, MIT master thesis, 1999.

Bilinearization MOR:

[2] J. R. Phillips, “Projection Frameworks for Model Reduction of Weakly Nonlinear Systems”, Proc. IEEE/ACM DAC, 184-189, 2000.

[3] Z. Bai, and D. Skoogh, “A projection method for model reduction of bilinear dynamical systems”, Linear Algebra Appl., 415: 406-425, 2006.

[4] L. Feng and P. Benner, A Note on Projection Techniques for Model Order Reduction of Bilinear Systems, AIP Conf. Proc. 936: 208-211, 2007.

References

Variational analysis MOR:

[5] J. R. Phillips, “Automated Extraction of Nonlinear Circuit Macromodels ”, Proc. of IEEE Custom Itegrated Circuits Conference, 451-454, 2000.

[6] L. Feng, X. Zeng, Ch. Chiang, D. Zhou, Q, Fang “Direct nonlinear order reduction with variational analysis”, Proc. of Design, Automation and Test in Europe Conference and Exhibition, 2:1316-1321, 2004.

Trajectory piece-wise linear (polynomial) MOR:

[7] M. Rewienski and J. White, “ A Trajectory Piecewise-Linear Approach to Model Order Reduction and Fast Simulation of Nonlinear Circuits and Micromachined Devices”, IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst., 22(2):155-170, 2003.

[8] M. Rewienski and J. White, “ Model order reduction for nonlinear dynamical systems based on trajectory piecewise-linear approximations ”, Linear Algebra Appl., 415(2-3):426-454, 2006.

References

POD:

[9] S. Volkwein, "Model reduction using proper orthogonal decomposition ", Lecture notes, June 7, 2010.

[10] S. Schaturantabut and D. C. Sorensen, "Nonlinear model reduction via discrete empirical interpolation", SIAM J. Sci. Comput. 32(5): 2737-2764, 2010.

A very good book for nonlinear system:

[11] W. J. Rugh, "Nonlinear system theory, the Volterra/Wiener Approach", The Johns Hopkins University Press, 1981.