Giuseppe Giordano and Jonas Sjöberg

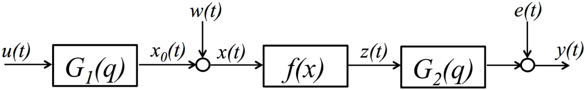
Department of Signals and Systems

Maximum Likelihood identification of Wiener-Hammerstein model in presence of process noise

Outline

Introduction

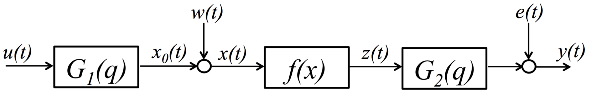
The Wiener-Hammerstein model in presence of process noise



Only signals u and y available for identification (N data points). Noises w, e zero mean and normally distributed.

Introduction

The Wiener-Hammerstein model in presence of process noise

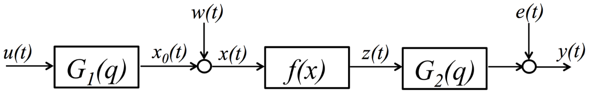


Only signals u and y available for identification (N data points). Noises w, e zero mean and normally distributed.

When $w(t) \equiv 0$ (only measurement noise):

Introduction

The Wiener-Hammerstein model in presence of process noise

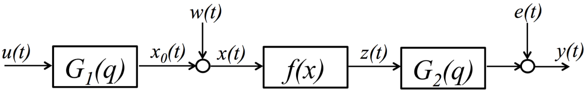


Only signals u and y available for identification (N data points). Noises w, e zero mean and normally distributed.
When $w(t) \equiv 0$ (only measurement noise):

- The Best Linear Approximation (BLA) is a consistent estimate of concatenation of the linear parts: $G_{BLA} = kG_1 G_2$

Introduction

The Wiener-Hammerstein model in presence of process noise

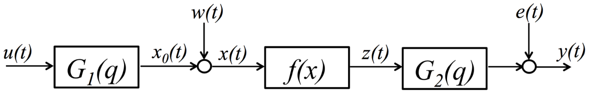


Only signals u and y available for identification (N data points). Noises w, e zero mean and normally distributed.
When $w(t) \equiv 0$ (only measurement noise):

- The Best Linear Approximation (BLA) is a consistent estimate of concatenation of the linear parts: $G_{BLA} = kG_1G_2$
- Main problem: splitting of the dynamics and non-linearity estimation

Introduction

The Wiener-Hammerstein model in presence of process noise

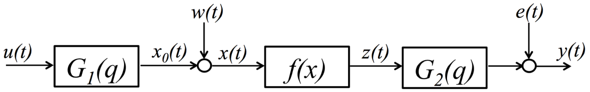


Only signals u and y available for identification (N data points). Noises w, e zero mean and normally distributed.
When $w(t) \equiv 0$ (only measurement noise):

- The Best Linear Approximation (BLA) is a consistent estimate of concatenation of the linear parts: $G_{BLA} = kG_1 G_2$
- Main problem: splitting of the dynamics and non-linearity estimation
- Several solutions to the splitting problem in literature based on BLA

Introduction

The Wiener-Hammerstein model in presence of process noise



Only signals u and y available for identification (N data points). Noises w, e zero mean and normally distributed.
When $w(t) \equiv 0$ (only measurement noise):

- The Best Linear Approximation (BLA) is a consistent estimate of concatenation of the linear parts: $G_{BLA} = kG_1 G_2$
- Main problem: splitting of the dynamics and non-linearity estimation
- Several solutions to the splitting problem in literature based on BLA

Question: Can we use BLA in presence of process noise? How can we implement the split?

The Best Linear Approximation in case of process noise

Data generated from a Wiener-Hammerstein system

$$y(t) = G_2^0(q)f(x(t)) + e(t) \tag{1a}$$

$$x(t) = x_0(t) + w(t) \tag{1b}$$

$$x_0(t) = G_1^0(q)u(t) \tag{1c}$$

The Best Linear Approximation in case of process noise

Data generated from a Wiener-Hammerstein system

$$y(t) = G_2^0(q)f(x(t)) + e(t) \tag{1a}$$

$$x(t) = x_0(t) + w(t) \tag{1b}$$

$$x_0(t) = G_1^0(q)u(t) \tag{1c}$$

Theorem (Consistency of the BLA)

Signals $u(t)$, $w(t)$, $e(t)$ independent, Gaussian, stationary processes. $G(q,\theta)$ arbitrary transfer function parametrization, with freely adjustable gain. Parameter θ is estimated from u and y using an output error method,

$$\hat{\theta}_N = \underset{\theta}{argmin} \quad \frac{1}{N}\sum_{t=1}^{N}(y(t) - G(q,\theta)u(t))^2, \tag{2}$$

then

$$G(q,\hat{\theta}_N) \rightarrow kG_1^0(q)G_2^0(q) \quad w.p.\ 1 \quad as \quad N \rightarrow \infty \tag{3}$$

The splitting problem

The BLA is still a consistent estimate of the linear parts. Can we use a standard splitting algorithm? How will the estimation of the non-linearity look like?

The splitting problem

The BLA is still a consistent estimate of the linear parts. Can we use a standard splitting algorithm? How will the estimation of the non-linearity look like?

A standard splitting algorithm based on the BLA (*exhaustive search*), when $w(t) \equiv 0$:

The splitting problem

The BLA is still a consistent estimate of the linear parts. Can we use a standard splitting algorithm? How will the estimation of the non-linearity look like?

A standard splitting algorithm based on the BLA (*exhaustive search*), when $w(t) \equiv 0$:

- Consider all possible combinations of the poles/zeros of the BLA.

The splitting problem

The BLA is still a consistent estimate of the linear parts. Can we use a standard splitting algorithm? How will the estimation of the non-linearity look like?

A standard splitting algorithm based on the BLA (*exhaustive search*), when $w(t) \equiv 0$:

- Consider all possible combinations of the poles/zeros of the BLA.
- For each combination estimate the non-linearity, using PEM.

The splitting problem

The BLA is still a consistent estimate of the linear parts. Can we use a standard splitting algorithm? How will the estimation of the non-linearity look like?

A standard splitting algorithm based on the BLA (*exhaustive search*), when $w(t) \equiv 0$:

- Consider all possible combinations of the poles/zeros of the BLA.
- For each combination estimate the non-linearity, using PEM.
- When $w(t) \equiv 0$, PEM is equivalent to the Maximum likelihood estimator (ML), hence the estimate is consistent.

The splitting problem

The BLA is still a consistent estimate of the linear parts. Can we use a standard splitting algorithm? How will the estimation of the non-linearity look like?

A standard splitting algorithm based on the BLA (*exhaustive search*), when $w(t) \equiv 0$:

- Consider all possible combinations of the poles/zeros of the BLA.
- For each combination estimate the non-linearity, using PEM.
- When $w(t) \equiv 0$, PEM is equivalent to the Maximum likelihood estimator (ML), hence the estimate is consistent.
- If α_i, β_i is a possible split of the dynamics in the BLA, PEM estimate is

$$\gamma* = \underset{\gamma}{\text{argmin}} \quad \frac{1}{N}\sum_{t=1}^{N}(y(t) - G_2(q,\beta_i)f(G_1(q,\alpha_i)u(t),\gamma))^2. \qquad (4)$$

The splitting problem

The BLA is still a consistent estimate of the linear parts. Can we use a standard splitting algorithm? How will the estimation of the non-linearity look like?

A standard splitting algorithm based on the BLA (*exhaustive search*), when $w(t) \equiv 0$:

- Consider all possible combinations of the poles/zeros of the BLA.
- For each combination estimate the non-linearity, using PEM.
- When $w(t) \equiv 0$, PEM is equivalent to the Maximum likelihood estimator (ML), hence the estimate is consistent.
- If α_i, β_i is a possible split of the dynamics in the BLA, PEM estimate is

$$\gamma * = \underset{\gamma}{\text{argmin}} \quad \frac{1}{N} \sum_{t=1}^{N} (y(t) - G_2(q, \beta_i) f(G_1(q, \alpha_i) u(t), \gamma))^2. \qquad (4)$$

- One estimation will provide the true split (lowest prediction error).

The splitting problem

The BLA is still a consistent estimate of the linear parts. Can we use a standard splitting algorithm? How will the estimation of the non-linearity look like?

A standard splitting algorithm based on the BLA (*exhaustive search*), when $w(t) \equiv 0$:

- Consider all possible combinations of the poles/zeros of the BLA.
- For each combination estimate the non-linearity, using PEM.
- When $w(t) \equiv 0$, PEM is equivalent to the Maximum likelihood estimator (ML), hence the estimate is consistent.
- If α_i, β_i is a possible split of the dynamics in the BLA, PEM estimate is

$$\gamma* = \underset{\gamma}{\text{argmin}} \quad \frac{1}{N}\sum_{t=1}^{N}(y(t) - G_2(q,\beta_i)f(G_1(q,\alpha_i)u(t),\gamma))^2. \qquad (4)$$

- One estimation will provide the true split (lowest prediction error).

Consistency of (4) needed both for the **estimation of non-linearity** and for the **splitting algorithm.**

Impact of the process noise on PEM

Problem: The PEM estimate (4) is not consistent in presence of process noise

Impact of the process noise on PEM

Problem: The PEM estimate (4) is not consistent in presence of process noise
Outcomes:

Impact of the process noise on PEM

Problem: The PEM estimate (4) is not consistent in presence of process noise
Outcomes:

- Even assuming the true split of the dynamics is known, the estimation of the non-linearity will be biased.

Impact of the process noise on PEM

Problem: The PEM estimate (4) is not consistent in presence of process noise
Outcomes:

- Even assuming the true split of the dynamics is known, the estimation of the non-linearity will be biased.
- In the exhaustive search algorithm, a wrong split could provide lowest prediction error. The splitting fails.

First order systems with polynomial non-linearity

Data points (1000) generated from the *true* system:

$$\begin{aligned} x_0(t) =& \frac{q}{q-a_1}u(t) \\ z(t) =& f(x_0(t)+w(t)) \\ y(t) =& \frac{q}{q-a_2}z(t)+e(t) \end{aligned} \tag{5}$$

$f(x)$ is a third degree polynomial:

$$f(x(t)) = c_0 + c_1x(t) + c_2x^2(t) + c_3x^3(t) \tag{6}$$

First order systems with polynomial non-linearity

Data points (1000) generated from the *true* system:

$$\begin{aligned} x_0(t) &= \frac{q}{q - a_1} u(t) \\ z(t) &= f(x_0(t) + w(t)) \\ y(t) &= \frac{q}{q - a_2} z(t) + e(t) \end{aligned} \tag{5}$$

$f(x)$ is a third degree polynomial:

$$f(x(t)) = c_0 + c_1 x(t) + c_2 x^2(t) + c_3 x^3(t) \tag{6}$$

- Input u is white Gaussian noise with standard deviation 1
- Process noise w is white Gaussian with standard deviation 4
- Signals u, w and e are mutually independent

First order systems with polynomial non-linearity

Monte-Carlo simulations used to generate estimation distributions over 1000 sets for

- estimation of the BLA with process noise

Table: BLA estimation

Poles/Zeros	True	Estimated $(\mu \pm \sigma)$
p_1	0.4	0.3988 ± 0.0100
p_2	0.8	0.7998 ± 0.0401

First order systems with polynomial non-linearity

Monte-Carlo simulations used to generate estimation distributions over 1000 sets for

- estimation of the BLA with process noise

Table: BLA estimation

Poles/Zeros	True	Estimated ($\mu \pm \sigma$)
p_1	0.4	0.3988 ± 0.0100
p_2	0.8	0.7998 ± 0.0401

- Splitting algorithm, 2 poles $\rightarrow$ 2 combinations

Table: Splitting

	True split (RMSE)	Wrong split (RMSE)
w/out pc	**1.9474**	1.96089
w/ pc	8.11246	**8.07622**

First order systems with polynomial non-linearity

Monte-Carlo simulations used to generate estimation distributions over 1000 sets for

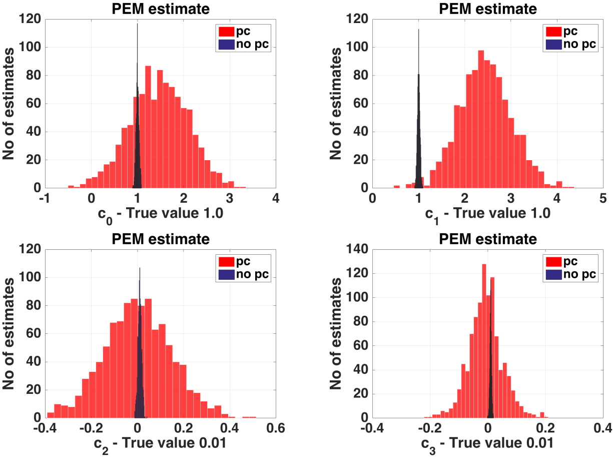
- estimation of the non-linearity with (w/) and without (w/out) process noise, for the true split

Table: Nonlinearity estimation

Par	True	Est w/ pc ($\mu \pm \sigma$)	Est w/out pc ($\mu \pm \sigma$)
c_0	1.0	1.4591 ± 0.3934	1.0010 ± 0.0009
c_1	1.0	2.4337 ± 0.3344	0.9987 ± 0.0008
c_2	0.01	0.0014 ± 0.0213	$0.0095 \pm 6.3 * 10^{-5}$
c_3	0.01	-0.0075 ± 0.0034	$0.0101 \pm 9.5 * 10^{-6}$

First order systems with polynomial non-linearity

Histograms of the estimates in presence of process noise. Parameters c_0 and c_1 show bias in the estimate.



Maximum likelihood in presence of process noise

- Main reason for inconsistency: PEM $\neq$ ML, in presence of process noise.

[1] A. Hagenblad, L. Ljung, and A. Wills. Maximum likelihood identification of Wiener models. Automatica, 44(11):26972705, 2008.

Maximum likelihood in presence of process noise

- Main reason for inconsistency: PEM $\neq$ ML, in presence of process noise.
- Possible solution: implementation of a ML estimate for the problem

$$\alpha*, \beta*, \gamma* = \text{argmax} \quad p_y(\alpha, \beta, \gamma; Z) \tag{7}$$

[1] A. Hagenblad, L. Ljung, and A. Wills. Maximum likelihood identification of Wiener models. Automatica, 44(11):26972705, 2008.

Maximum likelihood in presence of process noise

- Main reason for inconsistency: PEM $\neq$ ML, in presence of process noise.
- Possible solution: implementation of a ML estimate for the problem

$$\alpha*, \beta*, \gamma* = \text{argmax} \quad p_y(\alpha, \beta, \gamma; Z) \tag{7}$$

$Z = \{y^N, u^N\}$. The signal x can be introduced as a nuisance signal. [1]

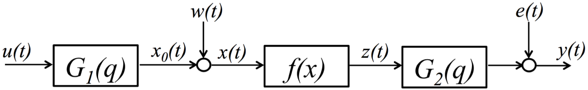
[1] A. Hagenblad, L. Ljung, and A. Wills. Maximum likelihood identification of Wiener models. Automatica, 44(11):26972705, 2008.

Maximum likelihood in presence of process noise

- Main reason for inconsistency: PEM $\neq$ ML, in presence of process noise.
- Possible solution: implementation of a ML estimate for the problem

$$\alpha*, \beta*, \gamma* = \text{argmax} \quad p_y(\alpha, \beta, \gamma; Z) \tag{7}$$

$Z = \{y^N, u^N\}$. The signal x can be introduced as a nuisance signal. [1]



$$\begin{aligned} p_y(y|x) &= p_e(y(t) - G_2(q, \beta) f(x(t), \gamma)) \\ p_x(x|u^N) &= p_w(x(t) - x_0(t)) = p_w(x(t) - G_1(q, \alpha) u(t)) \end{aligned} \tag{8}$$

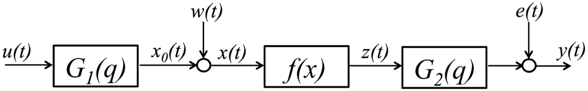
[1] A. Hagenblad, L. Ljung, and A. Wills. Maximum likelihood identification of Wiener models. Automatica, 44(11):26972705, 2008.

Maximum likelihood in presence of process noise

- Main reason for inconsistency: PEM $\neq$ ML, in presence of process noise.
- Possible solution: implementation of a ML estimate for the problem

$$\alpha*, \beta*, \gamma* = \text{argmax} \quad p_y(\alpha, \beta, \gamma; Z) \tag{7}$$

$Z = \{y^N, u^N\}$. The signal x can be introduced as a nuisance signal. [1]



$$\begin{aligned} p_y(y|x) &= p_e(y(t) - G_2(q,\beta) f(x(t),\gamma)) \\ p_x(x|u^N) &= p_w(x(t) - x_0(t)) = p_w(x(t) - G_1(q,\alpha) u(t)) \end{aligned} \tag{8}$$

e and w are normally distributed with zero mean and variances λ_e and λ_w respectively.

[1] A. Hagenblad, L. Ljung, and A. Wills. Maximum likelihood identification of Wiener models. Automatica, 44(11):26972705, 2008.

Maximum likelihood in presence of process noise

Compute p_y by integrating the joint probability p_{yx} over all $x(t)$

Maximum likelihood in presence of process noise

Compute p_y by integrating the joint probability p_{yx} over all $x(t)$

$$p_y(\alpha, \beta, \gamma; Z) = \int_{x \in \mathbf{R}} p_e p_w dx = \left(\frac{1}{2\pi\sqrt{\lambda_e \lambda_w}} \right)^N \prod_{t=1}^{N} \int_{-\infty}^{\infty} \mathrm{e}^{-\frac{1}{2} E(t,\alpha,\beta,\gamma)} dx(t) \tag{9}$$

Maximum likelihood in presence of process noise

Compute p_y by integrating the joint probability p_{yx} over all $x(t)$

$$p_y(\alpha,\beta,\gamma;Z) = \int_{x\in\mathbf{R}} p_e p_w dx = \left(\frac{1}{2\pi\sqrt{\lambda_e\lambda_w}}\right)^N \prod_{t=1}^{N}\int_{-\infty}^{\infty} \mathrm{e}^{-\frac{1}{2}E(t,\alpha,\beta,\gamma)}dx(t) \tag{9}$$

where

$$E(t,\alpha,\beta,\gamma) = \frac{1}{\lambda_e}(y(t)-G_2(q,\beta)f(x(t),\gamma))^2+\frac{1}{\lambda_w}(x(t)-G_1(q,\alpha)u(t))^2 \tag{10}$$

Maximum likelihood in presence of process noise

Compute p_y by integrating the joint probability p_{yx} over all $x(t)$

$$p_y(\alpha,\beta,\gamma;Z) = \int_{x\in\mathbf{R}} p_e p_w dx = \left(\frac{1}{2\pi\sqrt{\lambda_e\lambda_w}}\right)^N \prod_{t=1}^{N}\int_{-\infty}^{\infty} \mathrm{e}^{-\frac{1}{2}E(t,\alpha,\beta,\gamma)}dx(t) \tag{9}$$

where

$$E(t,\alpha,\beta,\gamma) = \frac{1}{\lambda_e}(y(t)-G_2(q,\beta)f(x(t),\gamma))^2 + \frac{1}{\lambda_w}(x(t)-G_1(q,\alpha)u(t))^2 \tag{10}$$

The estimation problem then will be

$$\min \quad -\log(p_y(\alpha,\beta,\gamma;Z)) = C + \frac{N}{2}\log(\lambda_e\lambda_w) - \sum_{t=1}^{N}\log\int_{-\infty}^{\infty}\mathrm{e}^{-\frac{1}{2}E(t,\alpha,\beta,\gamma)}dx(t) \tag{11}$$

Maximum likelihood in presence of process noise

Compute p_y by integrating the joint probability p_{yx} over all $x(t)$

$$p_y(\alpha,\beta,\gamma;Z) = \int_{x\in\mathbf{R}} p_e p_w dx = \left(\frac{1}{2\pi\sqrt{\lambda_e\lambda_w}}\right)^N \prod_{t=1}^{N} \int_{-\infty}^{\infty} \mathrm{e}^{-\frac{1}{2}E(t,\alpha,\beta,\gamma)} dx(t) \tag{9}$$

where

$$E(t,\alpha,\beta,\gamma) = \frac{1}{\lambda_e}(y(t) - G_2(q,\beta)f(x(t),\gamma))^2 + \frac{1}{\lambda_w}(x(t) - G_1(q,\alpha)u(t))^2 \tag{10}$$

The estimation problem then will be

$$\min \quad -\log(p_y(\alpha,\beta,\gamma;Z)) = C + \frac{N}{2}\log(\lambda_e\lambda_w) - \sum_{t=1}^{N} \log \int_{-\infty}^{\infty} \mathrm{e}^{-\frac{1}{2}E(t,\alpha,\beta,\gamma)} dx(t) \tag{11}$$

- Variances λ_e and λ_w can be also estimated.

Maximum likelihood in presence of process noise

Compute p_y by integrating the joint probability p_{yx} over all $x(t)$

$$p_y(\alpha,\beta,\gamma;Z) = \int_{x\in\mathbf{R}} p_e p_w dx = \left(\frac{1}{2\pi\sqrt{\lambda_e\lambda_w}}\right)^N \prod_{t=1}^{N} \int_{-\infty}^{\infty} \mathrm{e}^{-\frac{1}{2}E(t,\alpha,\beta,\gamma)} dx(t) \tag{9}$$

where

$$E(t,\alpha,\beta,\gamma) = \frac{1}{\lambda_e}(y(t) - G_2(q,\beta)f(x(t),\gamma))^2 + \frac{1}{\lambda_w}(x(t) - G_1(q,\alpha)u(t))^2 \tag{10}$$

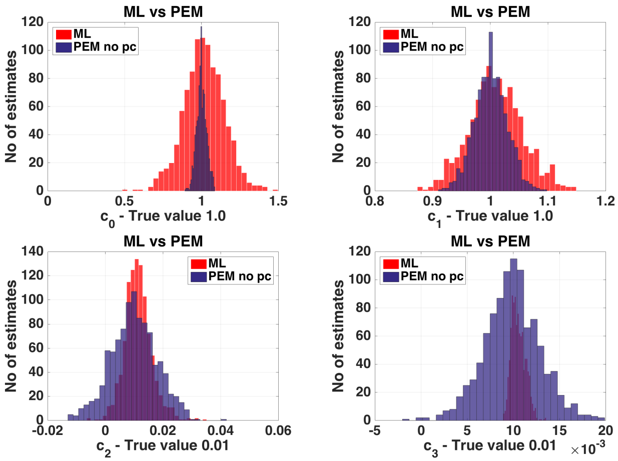
The estimation problem then will be

$$\min \quad -\log(p_y(\alpha,\beta,\gamma;Z)) = C + \frac{N}{2}\log(\lambda_e\lambda_w) - \sum_{t=1}^{N}\log\int_{-\infty}^{\infty} \mathrm{e}^{-\frac{1}{2}E(t,\alpha,\beta,\gamma)} dx(t) \tag{11}$$

- Variances λ_e and λ_w can be also estimated.
- Numerical integration of the integral needed (e.g. Monte-Carlo integration, Importance Sampling)

Illustrative example

Fixed linear parts. Maximum likelihood estimation with process noise compared to PEM estimate with no process noise.



Preliminary identification results using benchmark data

- Identification of a 6th order linear model (BLA)

Preliminary identification results using benchmark data

- Identification of a 6th order linear model (BLA)
- For all possible pole/zero combination the nonlinearity has been estimated using ML

Preliminary identification results using benchmark data

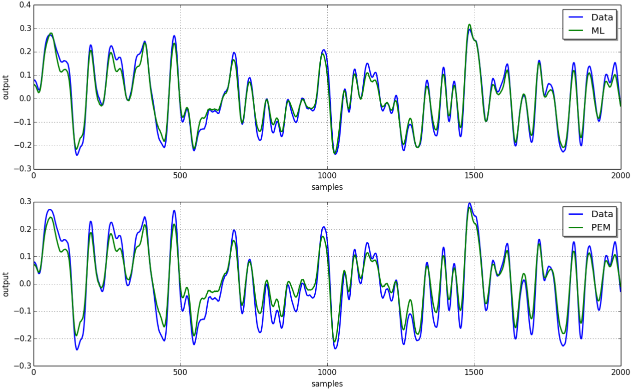
- Identification of a 6th order linear model (BLA)
- For all possible pole/zero combination the nonlinearity has been estimated using ML
- For the best split the following RMSE are obtained

Table: RMSE

Input	BLA	WHwPEM	WHwML
Multisine	0.035	0.0291	0.0162

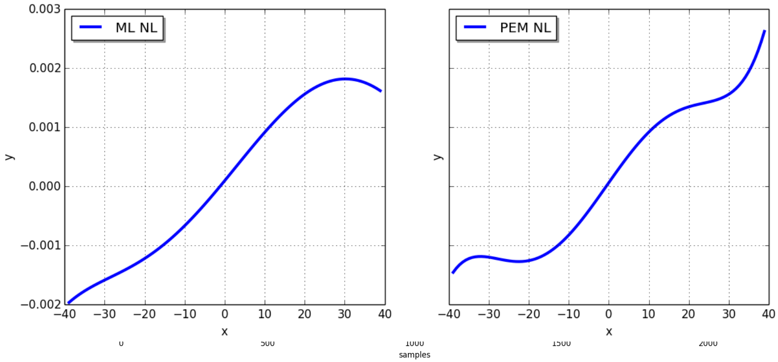
Preliminary identification results using benchmark data

- Simulated output and identified nonlinearity



Preliminary identification results using benchmark data

- Simulated output and identified nonlinearity



Preliminary identification results using benchmark data

Problems

- ML cost function is expensive (time/memory)

Preliminary identification results using benchmark data

Problems

- ML cost function is expensive (time/memory)
- Only small data set can be handled so far

Preliminary identification results using benchmark data

Problems

- ML cost function is expensive (time/memory)
- Only small data set can be handled so far
- The term $G_2(q,\beta)f(x(t),\gamma)$ "correlates" the entire sequence $x(t)$

$$E(t,\alpha,\beta,\gamma) = \frac{1}{\lambda_e}(y(t) - G_2(q,\beta)f(x(t),\gamma))^2 + \frac{1}{\lambda_w}(x(t) - G_1(q,\alpha)u(t))^2$$

Preliminary identification results using benchmark data

Problems

- ML cost function is expensive (time/memory)
- Only small data set can be handled so far
- The term $G_2(q,\beta)f(x(t),\gamma)$ "correlates" the entire sequence $x(t)$

$$E(t,\alpha,\beta,\gamma) = \frac{1}{\lambda_e}(y(t) - G_2(q,\beta)f(x(t),\gamma))^2 + \frac{1}{\lambda_w}(x(t) - G_1(q,\alpha)u(t))^2$$

- Numerical issues during the integration

Preliminary identification results using benchmark data

Problems

- ML cost function is expensive (time/memory)
- Only small data set can be handled so far
- The term $G_2(q,\beta)f(x(t),\gamma)$ "correlates" the entire sequence $x(t)$

$$E(t,\alpha,\beta,\gamma) = \frac{1}{\lambda_e}(y(t) - G_2(q,\beta)f(x(t),\gamma))^2 + \frac{1}{\lambda_w}(x(t) - G_1(q,\alpha)u(t))^2$$

- Numerical issues during the integration
- For colored process noise, include the noise model in the Likelihood function