Identification of Non-linear Dynamical Systems

Lennart Ljung

Linköping University

Sweden

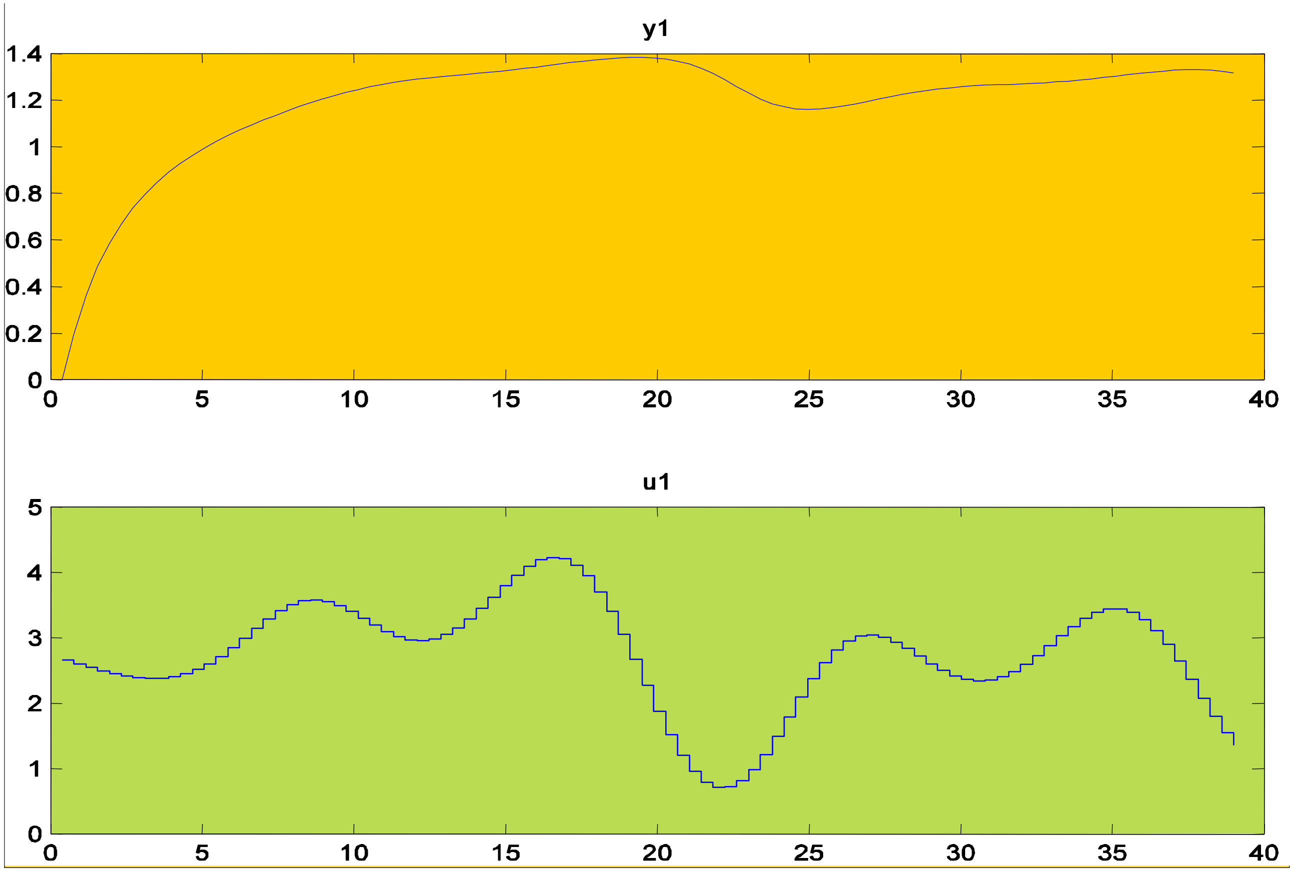
Prologue

Prologue

The PI, the Customer and the Data Set

- C: I have this data set. I have collected it from a cell metabolism experiment. The input is Glucose concentration and the output is the concentration of G6P. Can you help me building a model of this system?

The Data Set



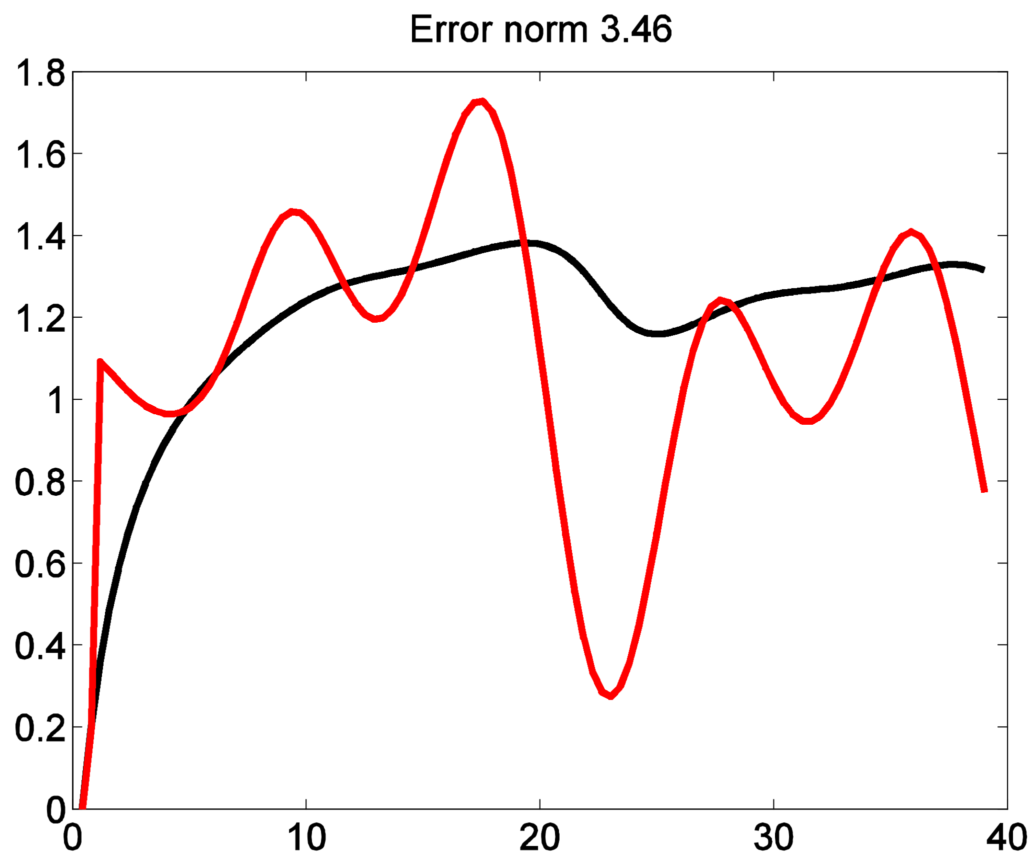
Output

Input

A Simple Linear Model

Red: Model

Black: Measured



Try the simplest model

y(t) = a u(t-1) + b u(t-2)

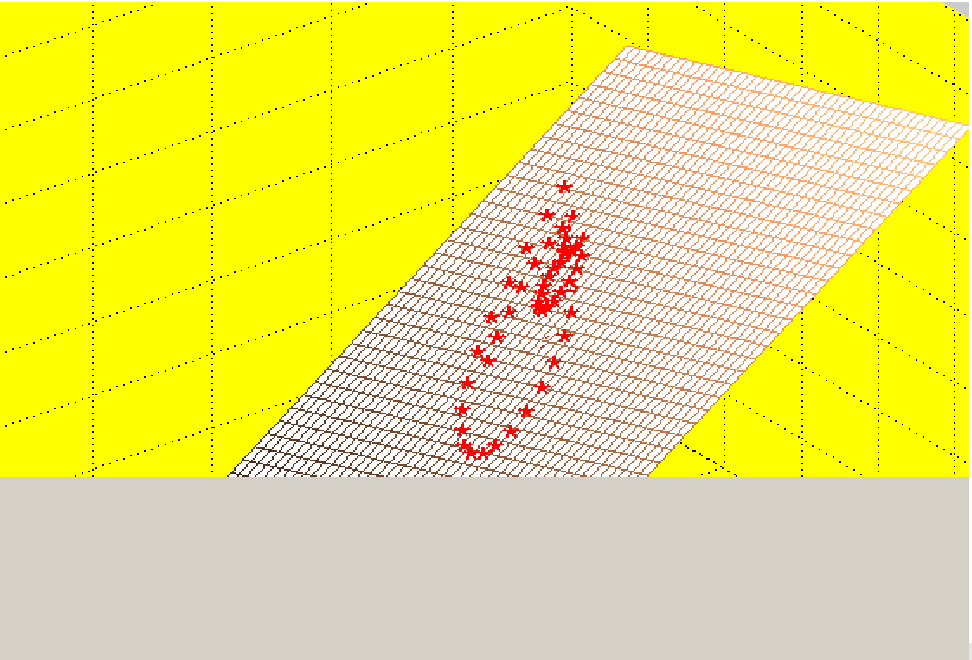
Fit by Least Squares:

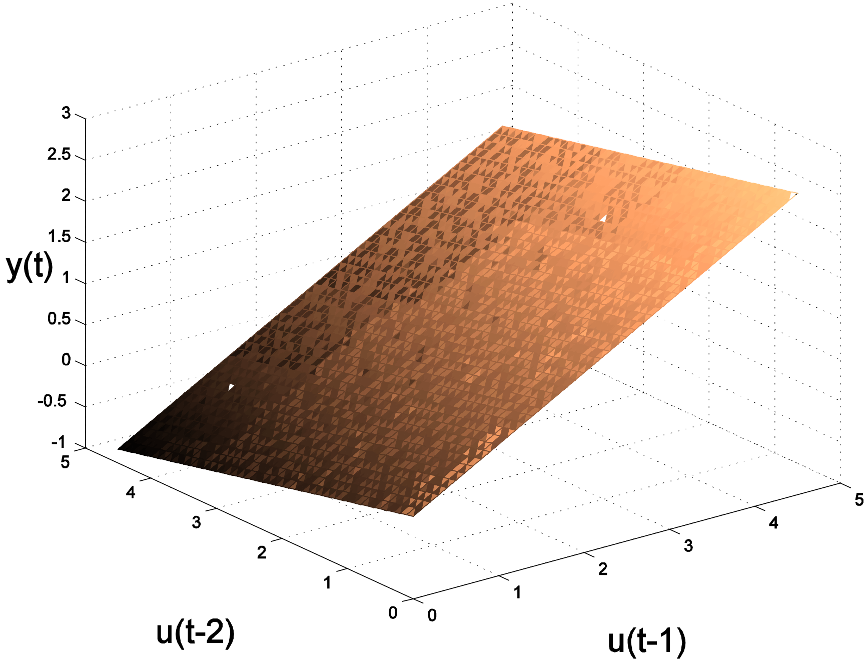
m1=arx(z,[0 2 1])

compare(z,m1)

A Picture of the Model

Depict the model as y(t) as a function of u(t-1) and u(t-2)





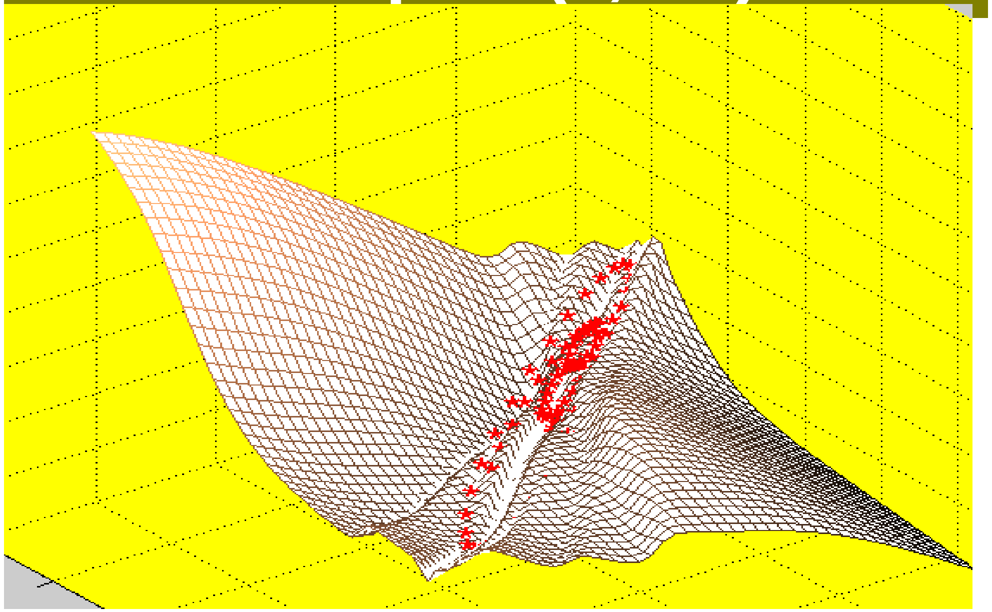
A Nonlinear Model

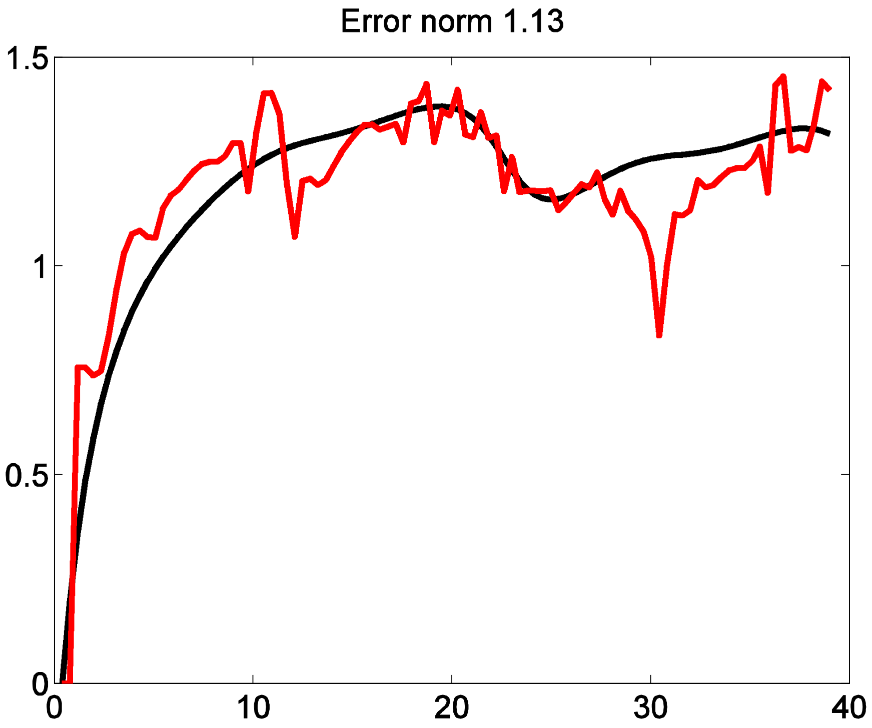
Try a nonlinear model

y(t) = f(u(t-1),u(t-2))

m2 = arxnl(z,[0 2 1],'sigm')

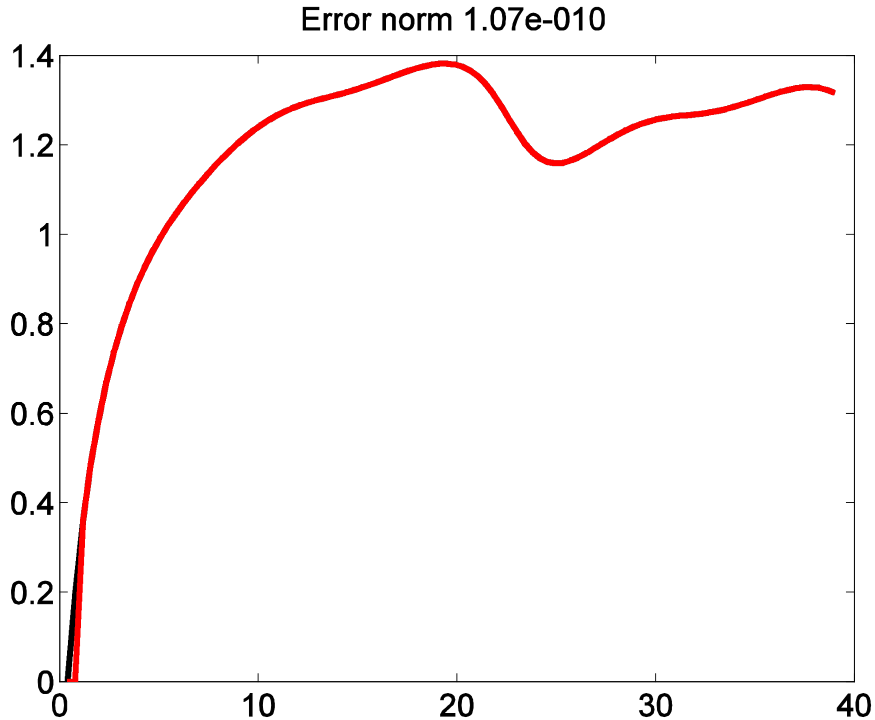
compare(z,m2)

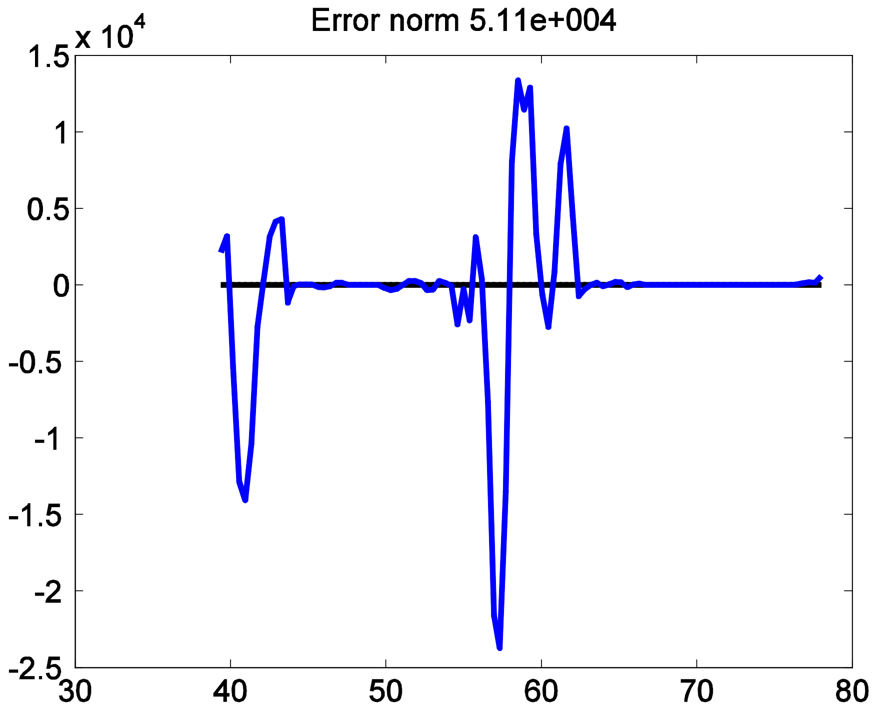




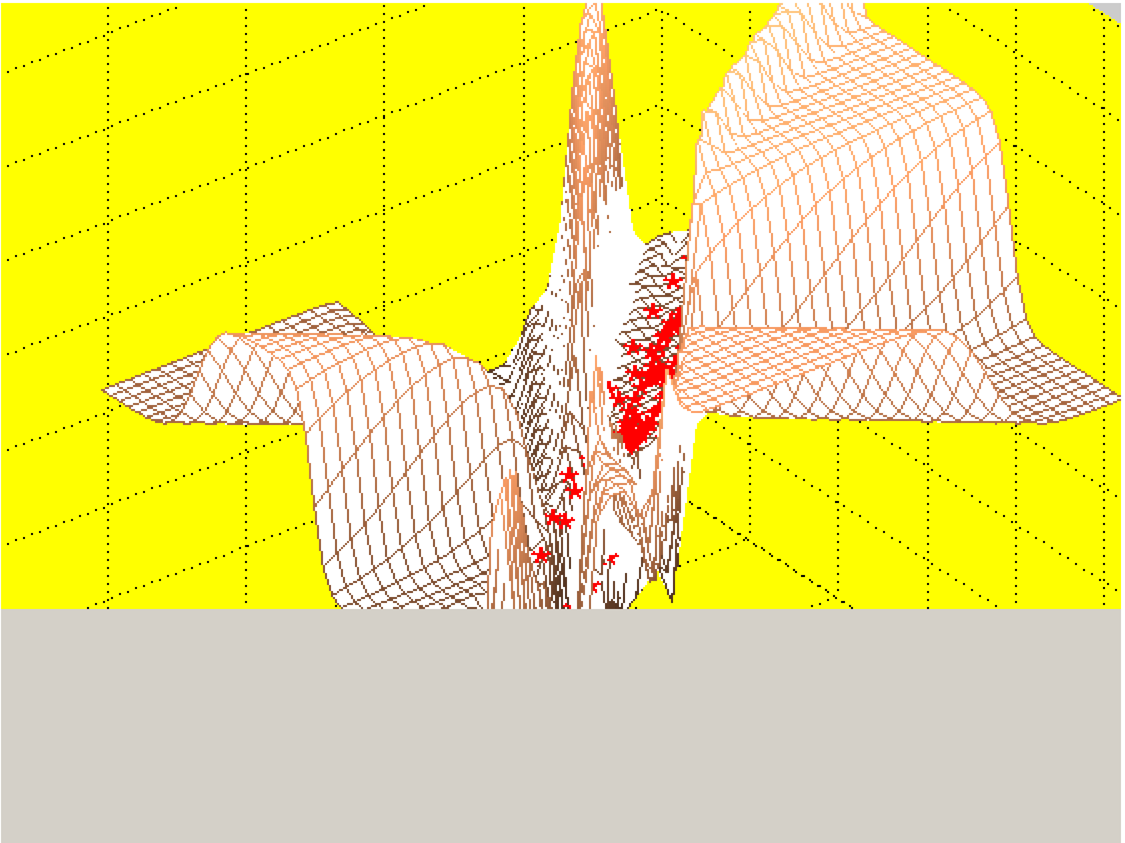
More Flexibility

A more flexible, nonlinear model
y(t) = f(u(t-1),u(t-2))
m3 = arxnl(z,[0 2 1],'sigm','numb',100)
compare(z,m3)
compare(zv,m3)



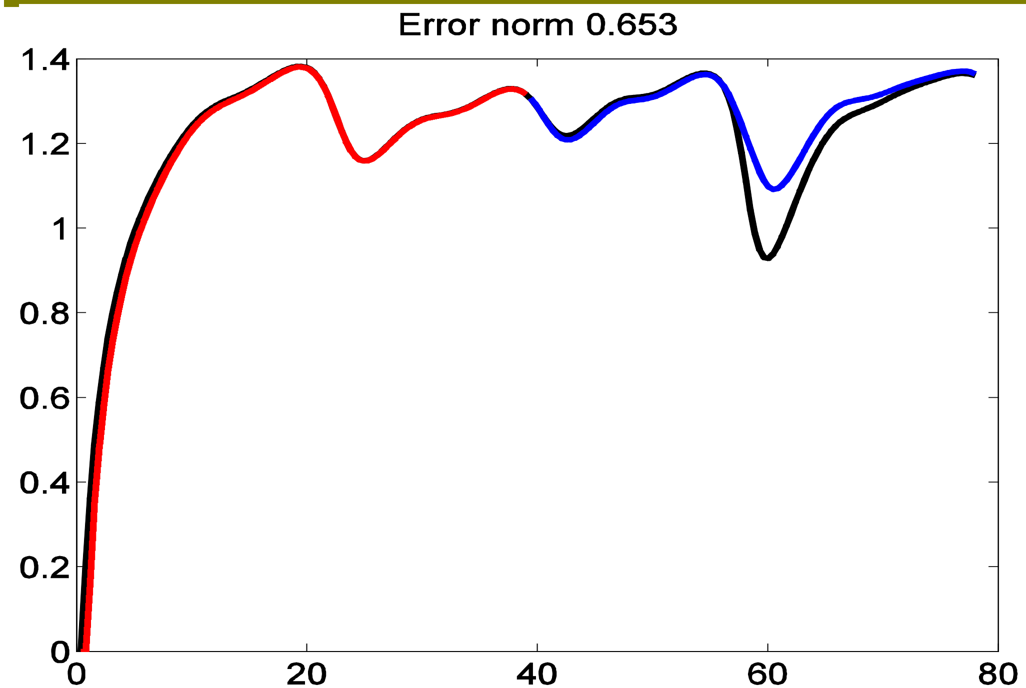


The Fit Between Model and Data



More Regressors

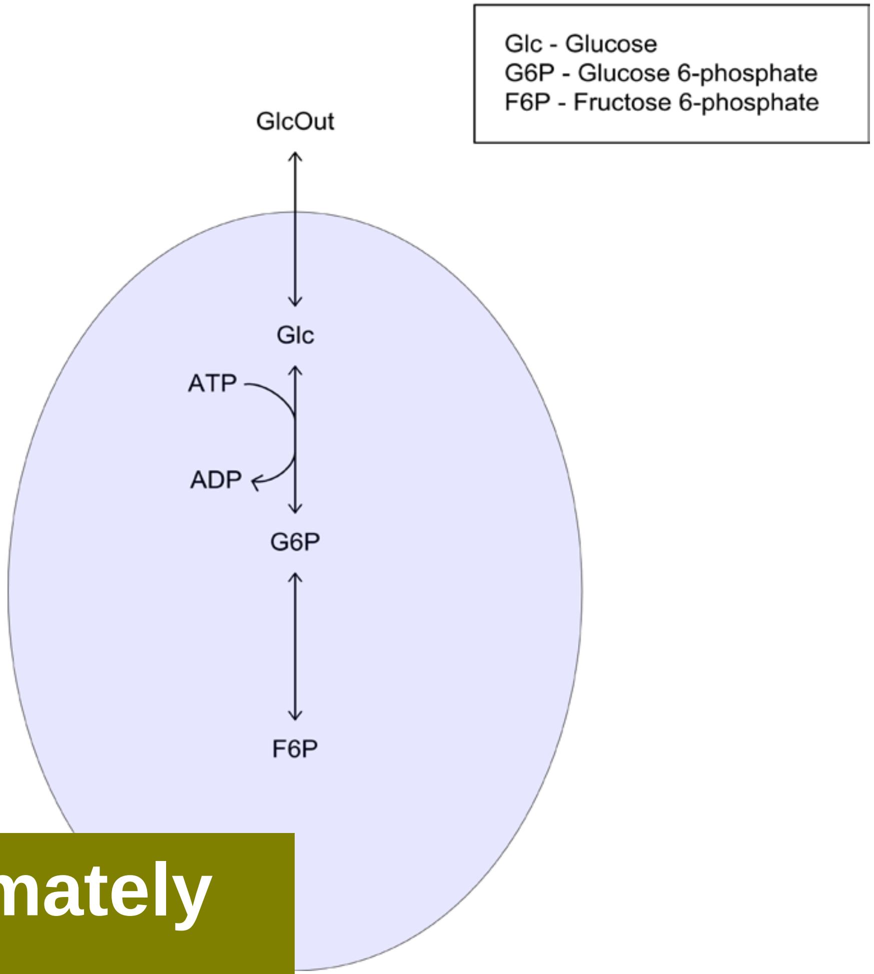
Try other arguments:
y(t) = f(y(t-1),y(t-2),u(t-1),u(t-2))
m4 = arxnl(z,[2 2 1],'sigm')
compare([z;zv],m4)



Biological Insight

Pathway diagram

$$
\begin{aligned}
\dot{x}_1 =& -\theta_1 \frac{x_1/\theta_2 - x_2/\theta_3}{1 + x_1/\theta_2 + x_2/\theta_3} \\
& + \theta_4 \frac{u - x_1}{1 + u/\theta_5 + x_1/\theta_5 + x_1 u/\theta_5^2} \\
\dot{x}_2 =& \theta_1 \frac{x_1/\theta_2 - x_2/\theta_3}{1 + x_1/\theta_2 + x_2/\theta_3} \\
& - \theta_6 \frac{x_2/\theta_7 - \theta_8}{1 + x_2/\theta_7 + \theta_8} \\
y =& x_2
\end{aligned}
$$

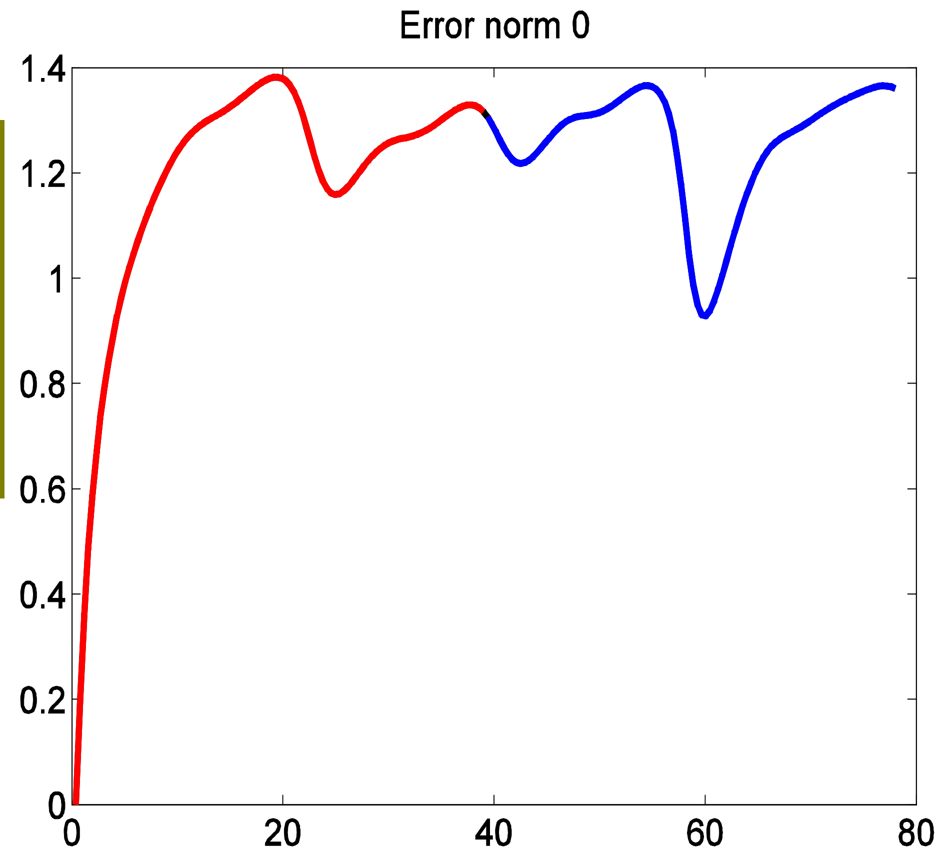


For sampled data, approximately

y(t) = f(y(t-1),y(t-2),u(t-1),u(t-2),θ)

Tailor-made Model Structure

```
cell = nlgrey(eqns,nom_pars)
m5 = pem(z,cell);
compare([z;zv],m5)
```



End of Prologue

Outline

- Problem formulation
- How to parameterize black box predictors
- Using physical insight
- Initialization of parameter search
- LTI approximation of non-linear systems

The Basic Picture

Input u, Output y, $Z^t = \{u(1), y(1), \ldots u(t), y(t)\}$

- State-Space

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

$$y = h(x, u, e)$$

w and e noises

$$\hat{y}(t|t-1) = E(y(t)|Z^{t-1})$$

- Output predictor

$$\hat{y}(t|t-1) = f_0(Z^{t-1})$$

The Predictor Function

General structure $\hat{y}(t|t-1) = f_0(Z^{t-1})$

Common/useful special case:

$$\hat{y}(t|t-1) = f_0(Z^{t-1}) = f_0(\phi(t))$$

$\phi(t) = \phi(Z^{t-1})$ of fixed dimension m (”state”, ”regressors”)

Think of the simple case

$$\phi(t) = \begin{bmatrix} y(t-1) & \ldots & y(t-n_a) & u(t-1) & \ldots & u(t-n_b) \end{bmatrix}$$

The Predictor Function

General structure $\hat{y}(t|t-1) = f_0(Z^{t-1})$

Common/useful special case:

$$\hat{y}(t|t-1) = f_0(Z^{t-1}) = f_0(\phi(t))$$

$\phi(t) = \phi(Z^{t-1})$ of fixed dimension m (”state”, ”regressors”)

Think of the simple case

$$\phi(t) = \begin{bmatrix} y(t-1) & \ldots & y(t-n_a) & u(t-1) & \ldots & u(t-n_b) \end{bmatrix}$$

The Data and the Identification Process

The observed data

$Z^N=[y(1),\phi_1,\ldots y(N),\phi_N]$

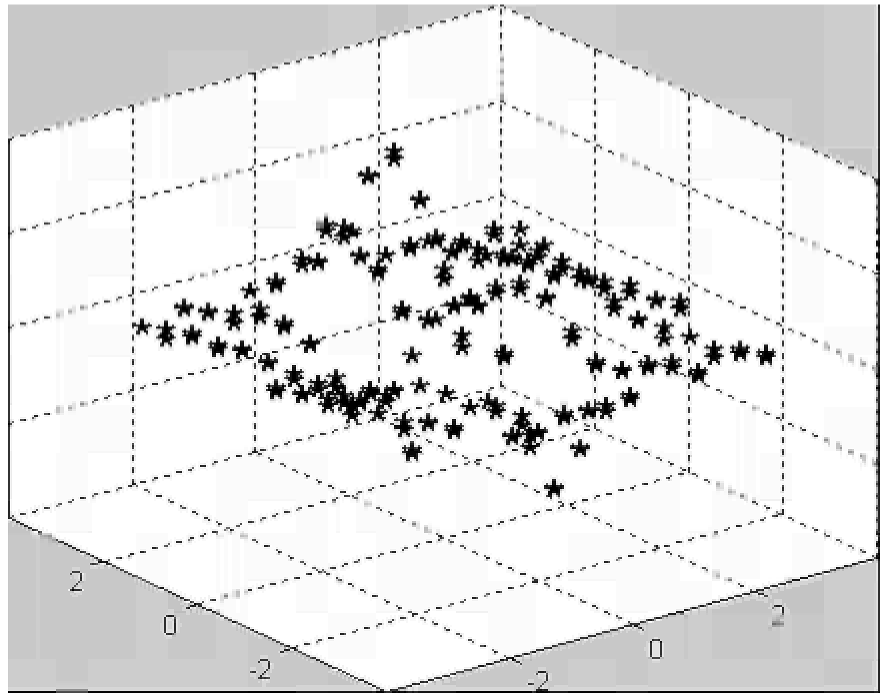
are N points in $\mathbb{R}^{m+1}$

The predictor model

$$\hat{y} = f_0(\phi)$$

is a surface in this space

Identification is to find the predictor surface from the data:



Mathematical Formulation

- Collect observations: Z^N , $y(t)=f_0(\phi(t))$+noise,
- Non-parametric: Smooth the $y(t)$'s locally over selected $\phi(t)$-regions
- Parametric:
  - Parameterize the predictor function: $f(\theta,\phi)$, $f\in\mathcal{F}$ when $\theta \in D$
  - Fit the parameters to the data:

$$\hat{\theta}_N = \arg\min_{\theta\in D} V_N(\theta, Z^N)$$

$$V_N(\theta, Z^N) = \sum_{t=1}^{N} \ell(y(t) - f(\theta, \phi(t))) = \sum_{t=1}^{N} \|y(t) - f(\theta, \phi(t))\|^2$$

- Use model $\hat{f}_N(\phi) = f(\hat{\theta}_N, \phi)$

Outline

- Problem formulation
- Parameterizing black box predictors
- Using physical insight
- Initialization of parameter search
- LTI approximation of non-linear systems

Predictor Function Parameterization

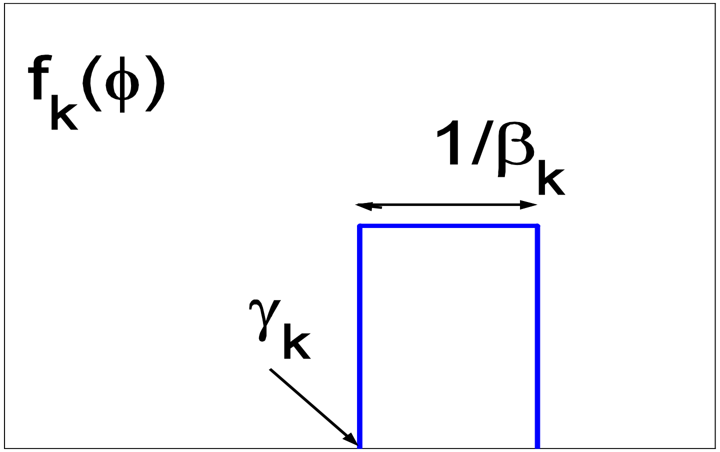
- How to parameterize the predictor function f(θ,ϕ)?
  - Grey-box (Physical insight of some sort)
  - Black-box (Flexible function expansions)

$$f(\theta, \phi) = \sum_{k=1}^{n} \theta_k f_k(\phi)$$

General case: $f_k(\phi) = f_k(\phi(\theta), \theta)$

Choice of Functions: Methods

- Neural Networks
- Radial Basis Neural Networks
- Wavelet-networks
- Neuro-Fuzzy models
- Spline networks
- Support Vector Machines
- Gaussian Processes
- Kriging



$$\gamma_k = \phi(k)$$

$\kappa =$ GaussianBell

ALL THESE USE

$$f(\theta, \phi) = \sum_{k=1}^{n} \alpha_k \kappa(\beta_k(\phi - \gamma_k))$$

Several layers….

$$\theta = \{\alpha_k, \beta_k, \gamma_k\}$$

An Aspect for Dynamical Systems

- Let $\phi(t) = [y(t-1), u(t-1)]^T$
- (One-step ahead) predicted output:

$$\hat{y}_p(t|\theta) = f(\theta, [y(t-1), u(t-1)]^T)$$

- This is normally what is fitted to data.
- A tougher test for the model is to simulate the output from past inputs only:

$$\hat{y}_s(t, \theta) = f(\theta, [\hat{y}_s(t-1, \theta), u(t-1)]^T)$$

- Stability issues!

The Basic Challenge

- Non-linear surfaces in high dimensions can be very complicated and need support of many observed data points.
- How to find parameterizations of such surfaces that both give a good chance of being close to the true system, and also use a moderate amount of parameters?
- The data cloud of observations is by necessity sparse in the surface space.

How to Deal with Sparsity

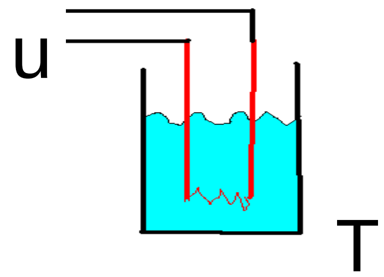
- Need ways to interpolate and extrapolate in the data space
- Leap of Faith: Search for global patterns in observed data to allow for data-driven interpolation
- Use Physical Insight: Allow for few parameters to parameterize the predictor surface, despite the high dimension.

Outline

- Problem formulation
- Parameterizing black box predictors
- Using physical insight
- Initialization of parameter search
- LTI approximation of non-linear systems

Using Physical Insight: Light Version

Semiphysical Modeling



Input: heater voltage u

Output: Fluid temperature T

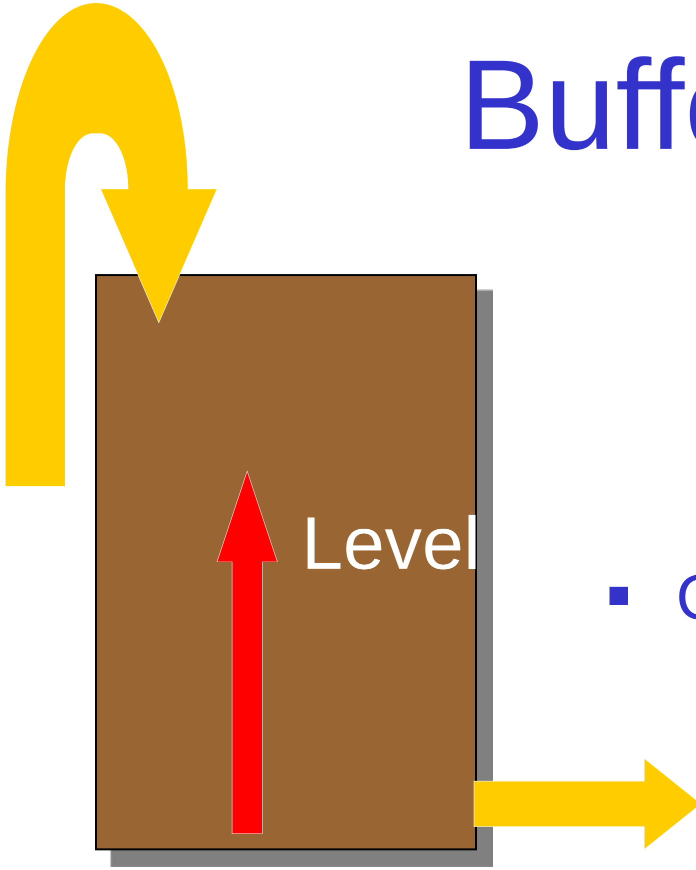
Square the voltage:
$u \rightarrow u^2$

Example: Semiphysical Modeling

Buffer Vessel for Pulp

Inflow

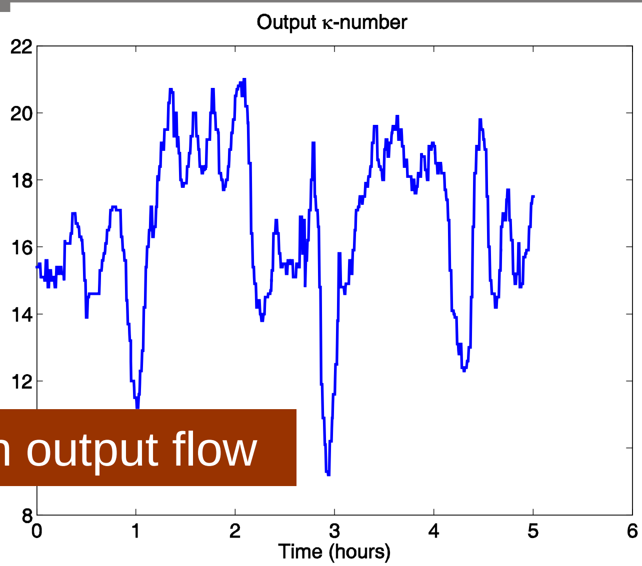
κ-number



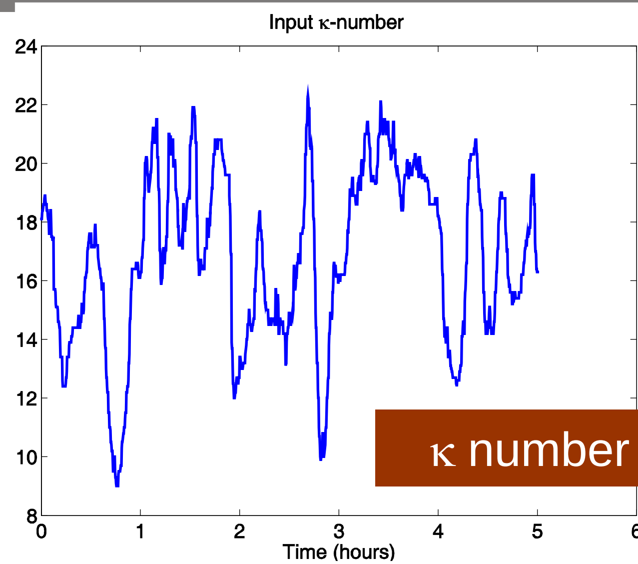
- Outflow
  - Flow
  - κ-number

Find the dynamics of this process!

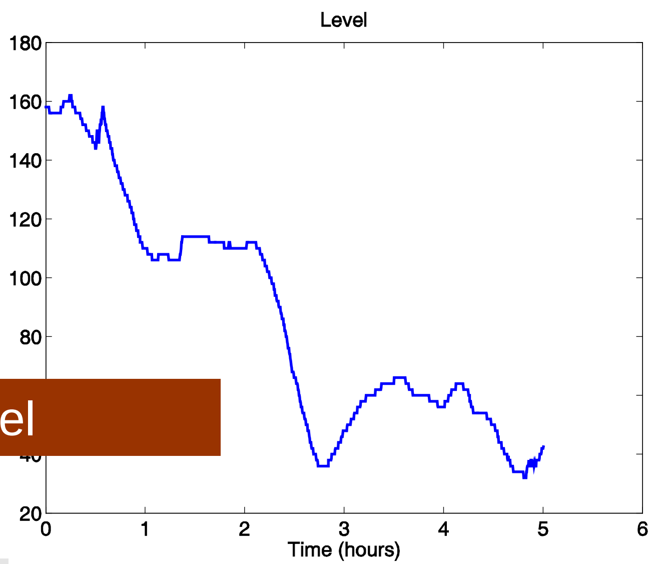
Measured Data from the Vessel



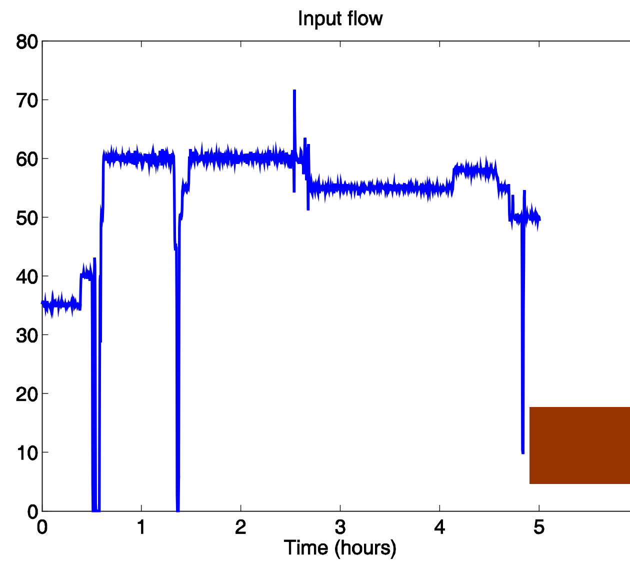
κ number in output flow



κ number in input flow

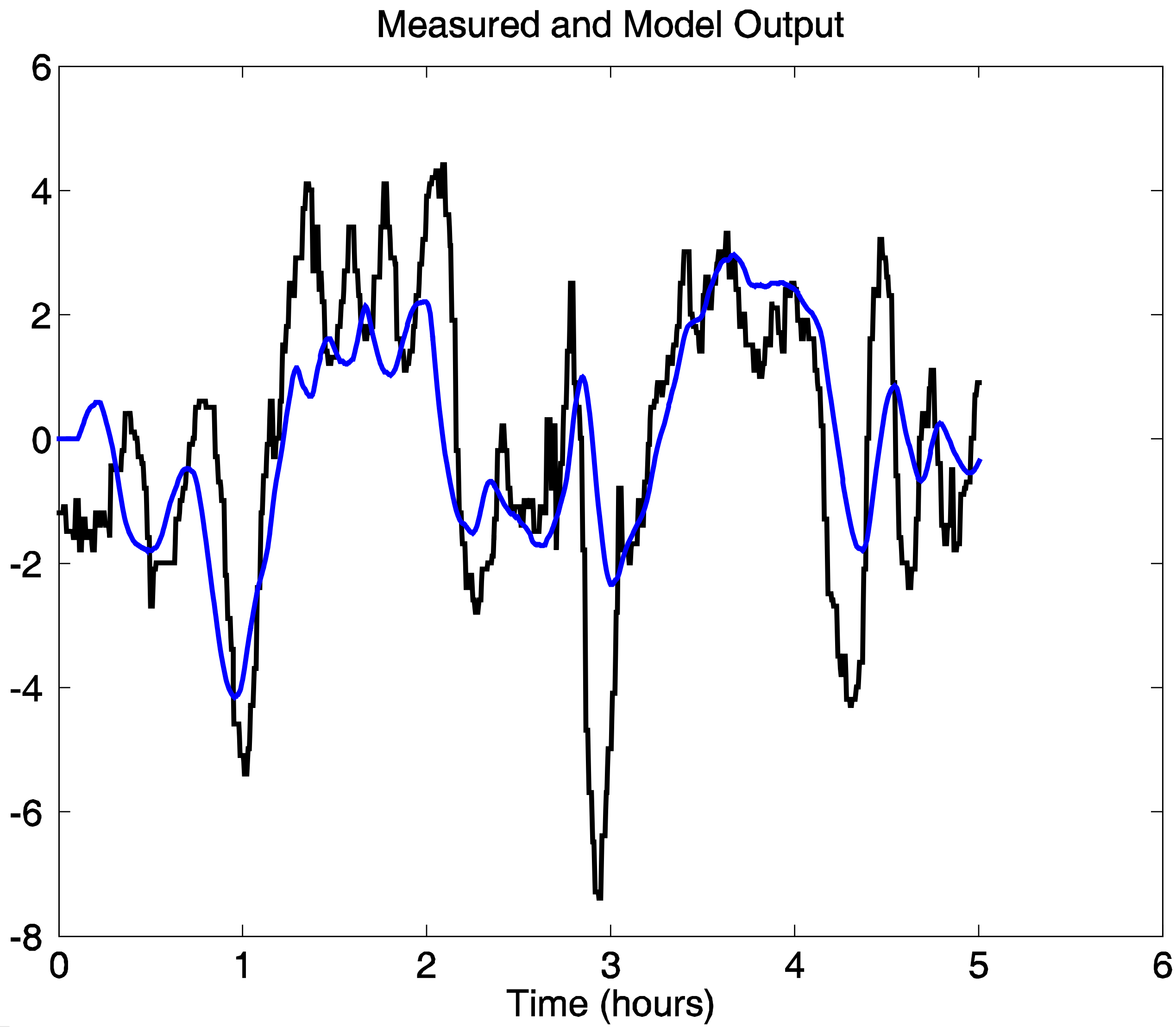


Level

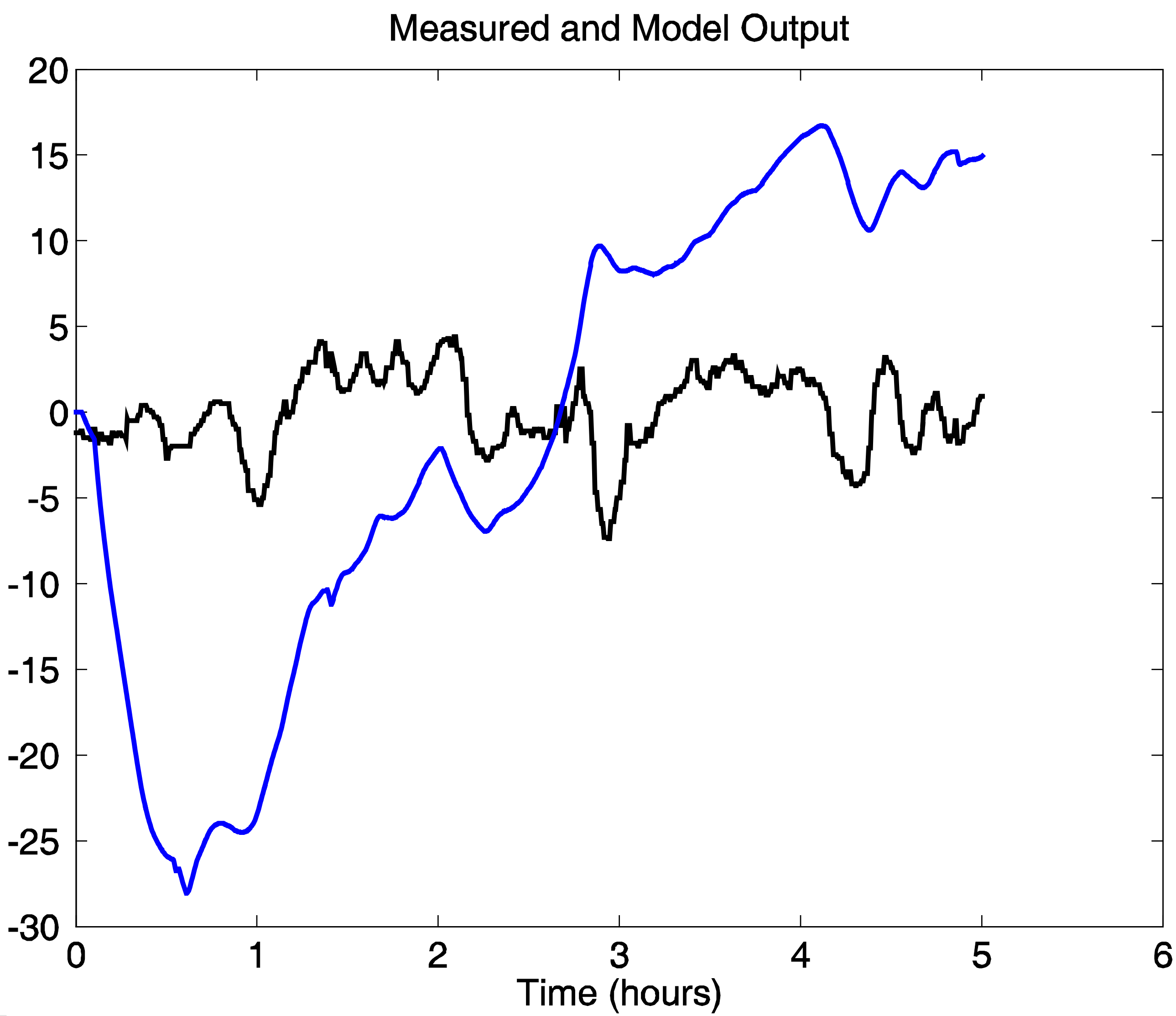


Flow

Fit a Linear Model to Data



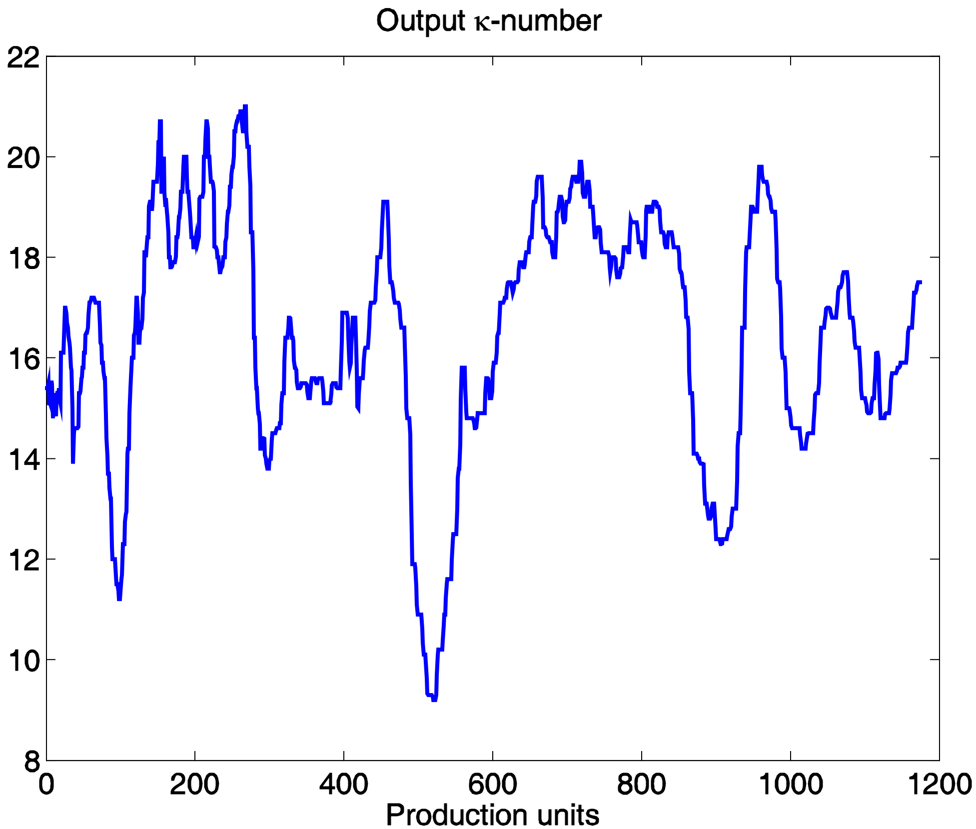
Using All 3 Inputs to Predict the Output

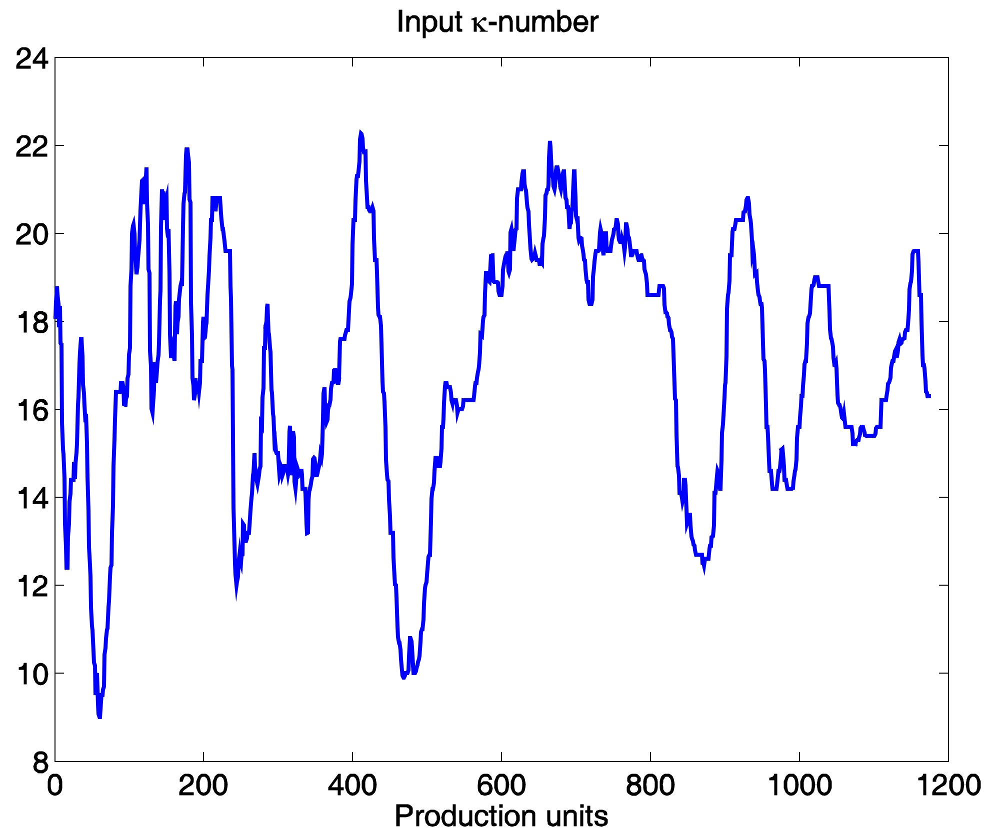


Think …

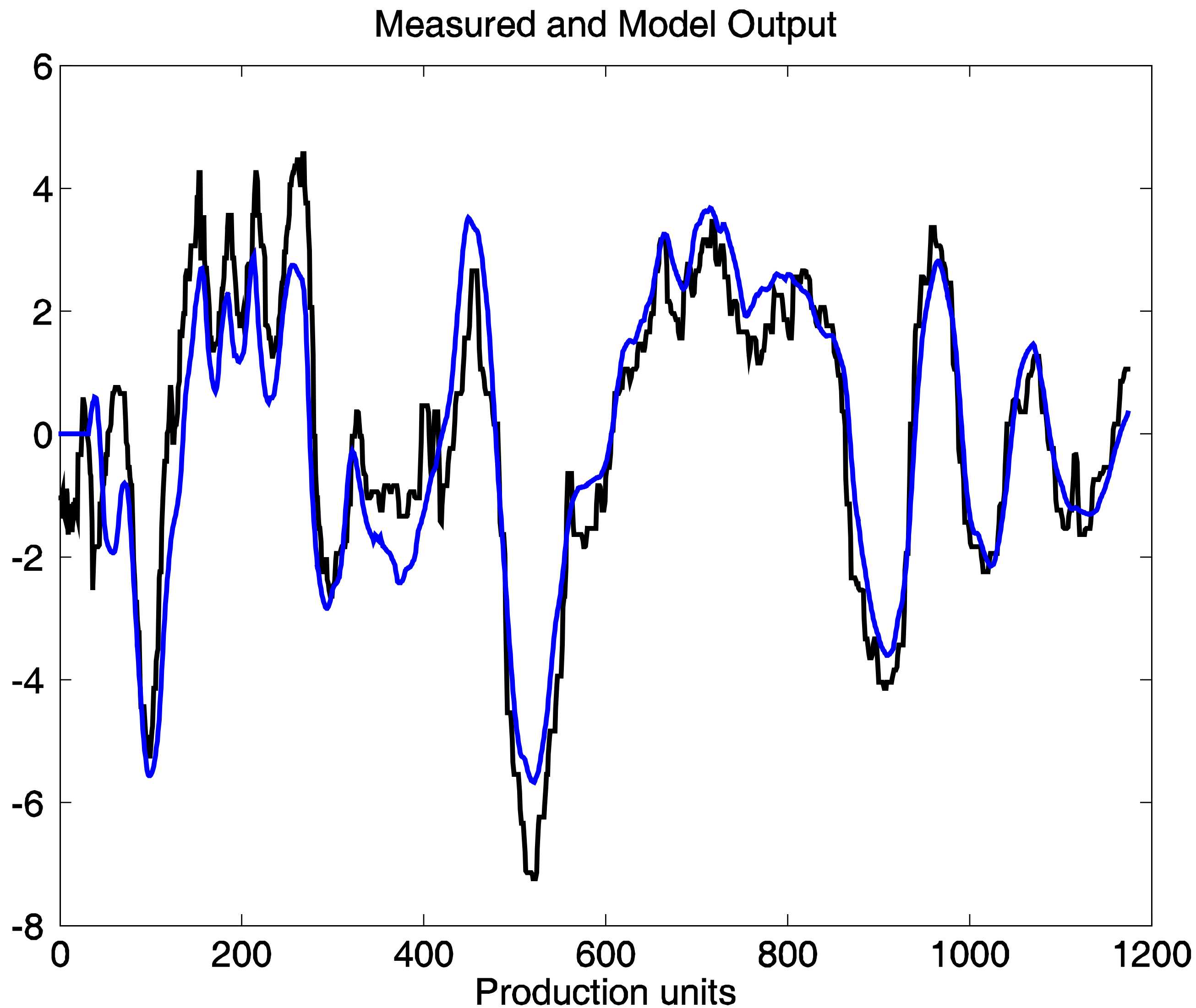
- Plug Flow: The system is a pure time delay of Volume/Flow
- Perfectly stirred tank: First order system with time constant = Volume/Flow
- Natural Time variable: Volume/Flow
- Rescale Time:
- `Pf = Flow/Level`
- `Newtime = interp1(cumsum(Pf),time,[Pf(1):sum(Pf)]);`
- `Newdata = interp1(Time,Data,Newtime);`

The Data with a New Time-scale





Simple Linear Model for Rescaled Data



Using Physical Insight: Serious Version

- Careful modeling leading to systems of Differential Algebraic Equations (DAE) parameterized by physical parameters.
- Support by modern modeling tools.
- The ”statistically correct” approach is to estimate the parameters by the Maximum Likelihood method.

Local Minima of the Criterion

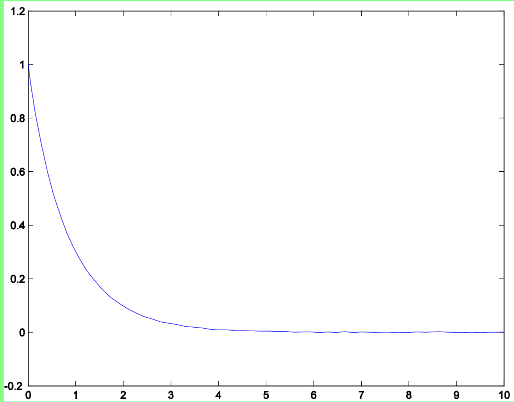
- This sounds like a general and reasonable approach
- Are there any catches?
- Well, to minimize the criterion of fit (maximizing the likelihood function) could be a challenge.
- Can be trapped in local minima….

Maximum Likelihood: The Solution?

- Example: A Michaelis-Menten equation:

$$\dot{x} = \theta_1 \frac{x}{\theta_2 + x} - x + u$$
$$y = x + e$$
$$u = \text{impulse}$$

- The output:



The ML Criterion (Gaussian Noise)

$$V(\theta) = \sum_{k=1}^{100} \|y(t_k) - x(t_k, \theta)\|^2$$

$$\dot{x}(t, \theta) = \theta_1 \frac{x(t, \theta)}{\theta_2 + x(t, \theta)} - x(t, \theta) + u(t)$$

V(θ) as a function of θ

Outline

- Problem formulation
- How to parameterize black box predictors
- Using physical insight
- Initialization of parameter search
- LTI approximation of non-linear systems

Can We Handle Local Minima ?

- Can the observed data be linked to the parameters in a different (and simpler) way?
- Manipulate the equations …

Ex: The Michaelis-Menten Equation

- In our case (noisefree)

$$\dot{y} = \frac{\theta_1 y}{y+\theta_2} - y + u$$

$$\dot{y}y + \theta_2\dot{y} = \theta_1 y - y^2 - \theta_2 y + uy + \theta_2 u$$

$$\dot{y}y + y^2 - uy = \begin{bmatrix}\theta_1 & \theta_2\end{bmatrix}\begin{bmatrix} y \\ u - \dot{y} - y\end{bmatrix}$$

For observed y and u this is a linear regression in the parameters. With noisy observations, the noise structure will be violated, though, which could lead to biased estimates.

Identifiability and Linear Regression

Crucial Challenge for physically parameterized models: Find a good initial estimate

- Result of conceptual interest: (Ljung, Glad, 1994)

A parameterized set of DAEs is globally identifiable

if and only if

the set can be rearranged as a linear regression

Ritt's algorithm from differential algebra provides a finite procedure for constructing the linear regression

Example of Ritt's Algorithm

Original equations

$$\dot{x}_1 = \theta x_2^2$$
$$\dot{x}_2 = u$$
$$y = x_1$$

Differentiate y twice

$$\dot{y} = \dot{x}_1 = \theta x_2^2$$
$$\ddot{y} = 2\theta x_2 \dot{x}_2 = 2\theta x_2 u$$

Square the last expression

$$\ddot{y}^2 = 4\theta\theta x_2^2 u^2 = 4\theta \dot{y} u^2$$

which is a linear regression

Challenge for Parameter Initialization

- Only small examples treated so far. Make the initialization work in bigger problems.
- Potential for important contributions:
  - Handle the complexity by modularization
  - Handle the noise corruption so that good quality initial estimates are secured
- Room for innovative ideas using algebra and semidefinite programming!

S

A Control Aspect

- Despite all the work and results on non-linear models, the most common situation will still be

How to live with an estimated LTI model approximation of a Non-linear system.

Outline

- Problem formulation
- Generalization properties
- How to parameterize black box predictors
- Using physical insight
- Initialization of parameter search
- LTI approximation of non-linear systems

Non-linear System Approximation

- Given an LTI Output-error model structure $y=G(q,\theta)u+e$, what will the resulting model be for a non-linear system?
- Assume that the inputs and outputs u and y are such that the spectra Φ_u and Φ_{yu} are well defined.
- Then the LTI second order equivalent is

$$G_0 = \frac{1}{\lambda L(z)} \left[\frac{\Phi_{yu}(z)}{L(z^{-1})}\right]_{\text{causal}} \qquad \Phi_u(z) = \lambda L(z) L(z^{-1})$$

Note: G_0 depends on u

- The limit model will be

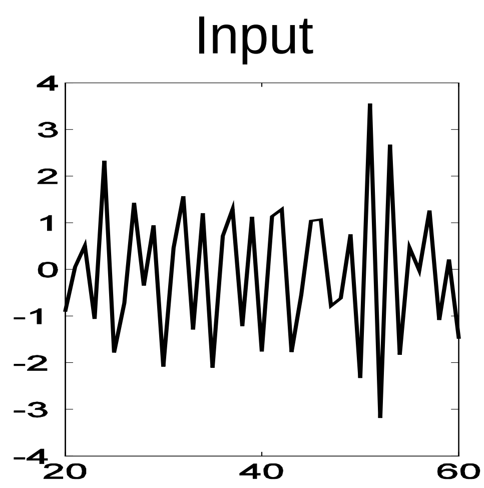
$$\min_\theta \int |G(z,\theta) - G_0(z)|^2 \Phi_u(z) dz$$

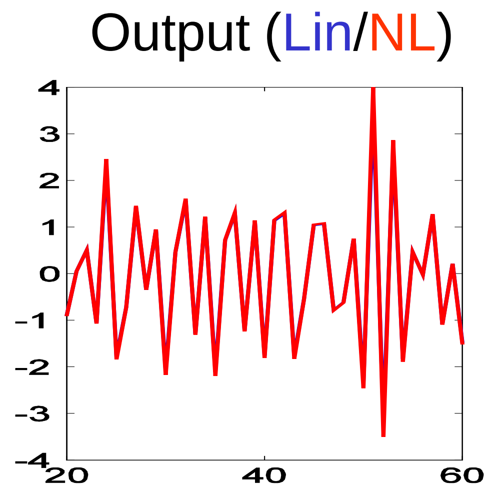
An Example

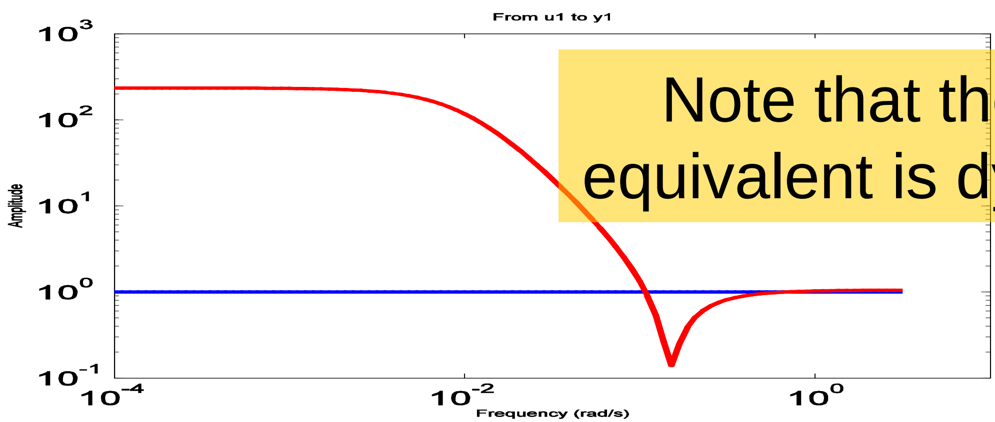
- Two data sets
- Input u and output y
- $y = u$
- $y = u + 0.01u^3$

(Enqvist, 2003)

The corresponding LTI equivalents (amplitude Bode plot)





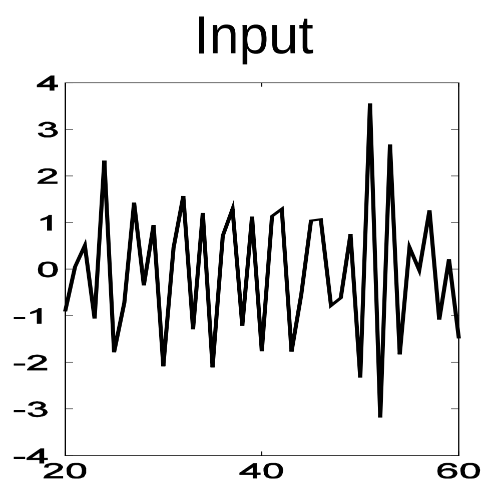


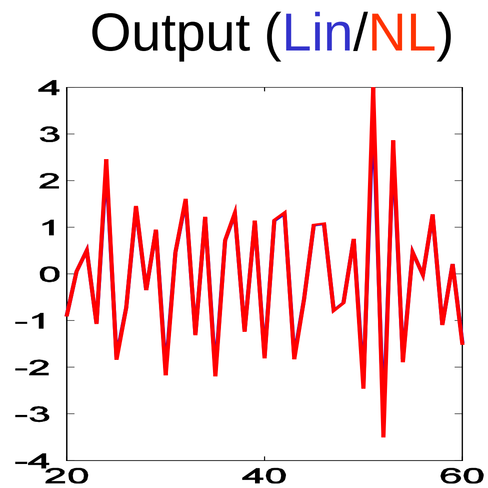
Note that the LTI equivalent is dynamic!

An Example

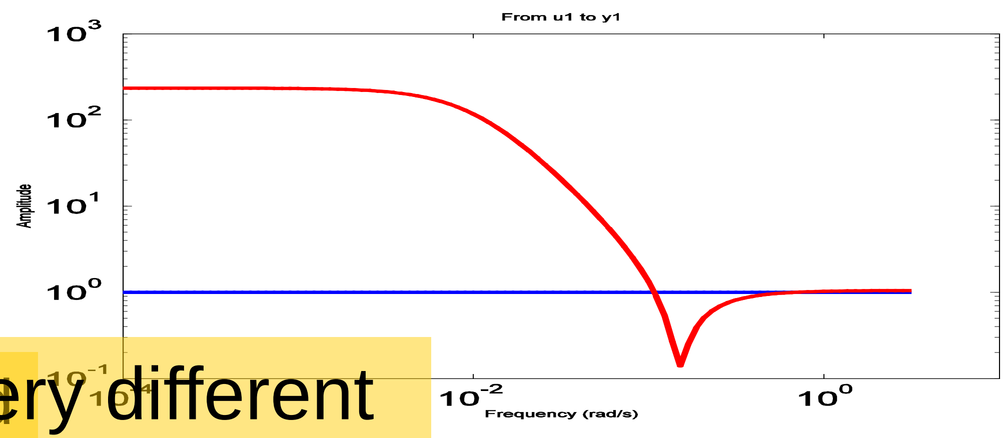
- Two data sets
- Input u and output y
- $y = u$
- $y = u + 0.01u^3$

(Enqvist, 2003)





The corresponding LTI equivalents (amplitude Bode plot)



Is the red Bode plot a good basis for control design?

So, oe(z,[2 2 1]) give very different results for the two data sets!

S

Epilogue: Tasks for the Control Community

- Black-box models
  - Working stability theory: Prediction/Simulation
- Semiphysical Models
  - Tools to generate and test non-linear transformations of data
- Fully integrated software for modeling and identification
  - Object oriented modeling
  - Differential Algebraic Equations – including disturbance modeling
  - Robust parameter initialization techniques
- Understand LTI approximation of nonlinear dynamic systems