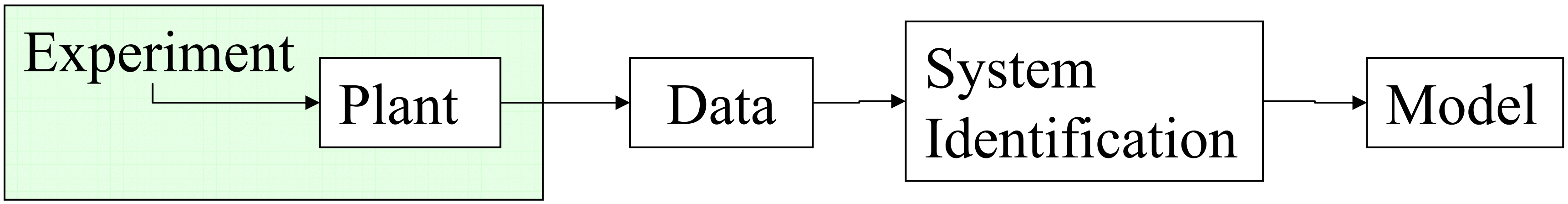
# Lecture 10 - Model Identification

- What is system identification?
- Direct impulse response identification
- Linear regression
- Regularization
- Parametric model ID, nonlinear LS

# What is System Identification?




- White-box identification
  - estimate parameters of a physical model from data
  - Example: aircraft flight model
- Gray-box identification
  - given generic model structure estimate parameters from data
  - Example: neural network model of an engine
- Black-box identification
  - determine model structure and estimate parameters from data
  - Example: security pricing models for stock market

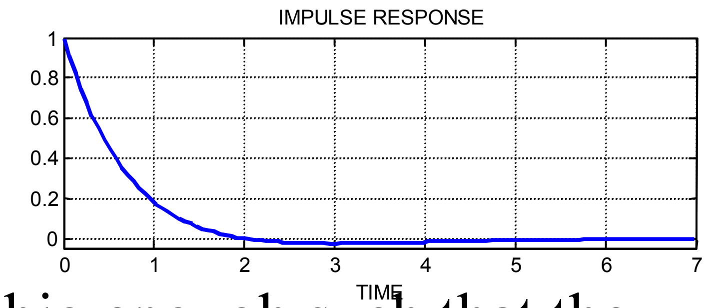
Rarely used in real-life control

# Industrial Use of System ID

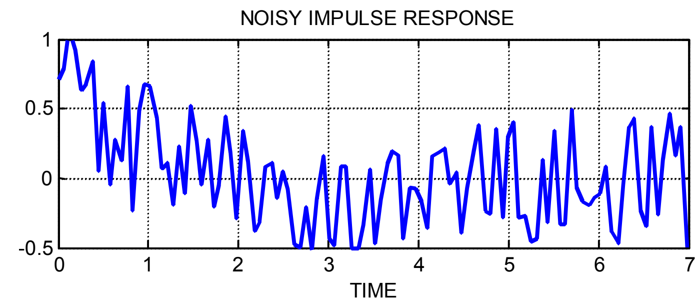
- Process control - most developed ID approaches
  - all plants and processes are different
  - need to do identification, cannot spend too much time on each
  - industrial identification tools
- Aerospace
  - white-box identification, specially designed programs of tests
- Automotive
  - white-box, significant effort on model development and calibration
- Disk drives
  - used to do thorough identification, shorter cycle time
- Embedded systems
  - simplified models, short cycle time

# Impulse response identification

- Simplest approach: apply control impulse and collect the data



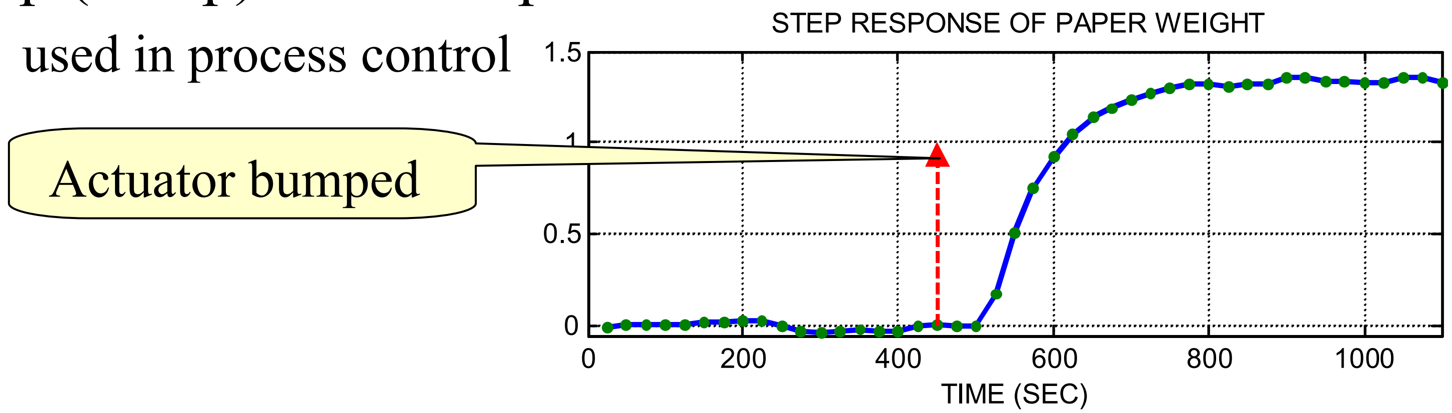

- Difficult to apply a short impulse big enough such that the response is much larger than the noise



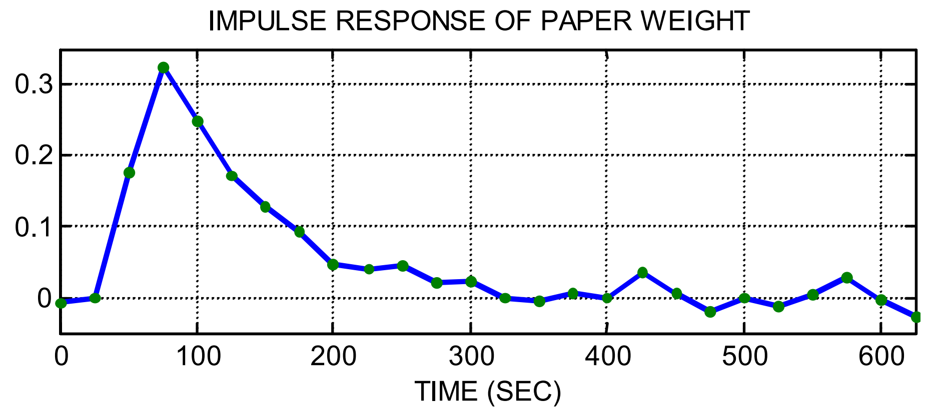

- FIR modeling can be used for building simplified control design models from complex sims

# Step response identification

- Step (bump) control input and collect the data
  - used in process control




- Impulse estimate: impulse(t) = step(t)-step(t-1)
- Still noisy



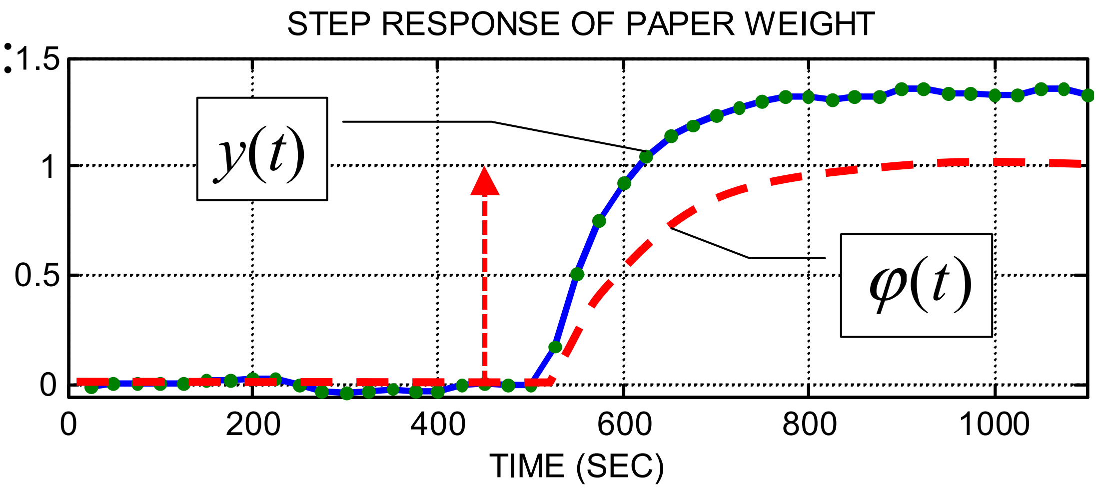
# Noise reduction

Noise can be reduced by statistical averaging:

- Collect data for multiple step inputs and perform more averaging to estimate the step/pulse response
- Use a parametric model of the system and estimate a few model parameters describing the response: dead time, rise time, gain
- Do both in a sequence
  - done in real process control ID packages
- Pre-filter data

# Linear Regression - univariate

- Simple fitting problem:
  - Given model step response $y(t)$
  - And process step response $\varphi(t)$
  - Find the gain factor $\theta$




$$y(t) = \theta\varphi(t) + e(t)$$

$$y = \Phi\theta + e$$

$$y = \begin{bmatrix} y(1) \\ \vdots \\ y(N) \end{bmatrix}, \Phi = \begin{bmatrix} \varphi(1) \\ \vdots \\ \varphi(N) \end{bmatrix}, e = \begin{bmatrix} e(1) \\ \vdots \\ e(N) \end{bmatrix}$$

Solution assuming uncorrelated noise:

$$\theta = \frac{\Phi^T y}{\Phi^T \Phi}$$

# Linear Regression

- Linear regression is one of the main System ID tools

Data — Regression weights — Regressor — Error of the fit

$$y(t) = \sum_{j=1}^{K} \theta_j \varphi_j(t) + e(t)$$

$$y = \Phi\theta + e$$

$$y = \begin{bmatrix} y(1) \\ \vdots \\ y(N) \end{bmatrix}, \Phi = \begin{bmatrix} \varphi_1(1) & \dots & \varphi_K(1) \\ \vdots & \ddots & \vdots \\ \varphi_1(N) & \dots & \varphi_K(N) \end{bmatrix}, \theta = \begin{bmatrix} \theta_1 \\ \vdots \\ \theta_K \end{bmatrix}, e = \begin{bmatrix} e(1) \\ \vdots \\ e(N) \end{bmatrix}$$

# Linear regression - least squares

- Makes sense only when matrix $\Phi$ is tall, $N > K$, more data available than the number of unknown parameters.
  - Statistical averaging
- Least square solution: $\|e\|^2 \to \min$

$$L = (y - \Phi\theta)^T (y - \Phi\theta) \to \min$$

$$\frac{\partial L}{\partial \theta} = -2\Phi^T (y - \Phi\theta) = 0$$

$$\boxed{\hat{\theta} = \left(\Phi^T \Phi\right)^{-1} \Phi^T y}$$


- Can be computed using Matlab `pinv` or left matrix division \

# Linear regression - least squares

- Correlation interpretation of the least squares solution

$$\hat{\theta} = \left(\Phi^T \Phi\right)^{-1} \Phi^T y \quad \Longleftrightarrow \quad \hat{\theta} = R^{-1} c$$

$$R = \frac{1}{N} \Phi^T \Phi \qquad c = \frac{1}{N} \Phi^T y$$

$$R = \frac{1}{N} \begin{bmatrix} \sum_{t=1}^{N} \varphi_1^2(t) & \cdots & \sum_{t=1}^{N} \varphi_K(t)\varphi_1(t) \\ \vdots & \ddots & \vdots \\ \sum_{t=1}^{N} \varphi_1(t)\varphi_K(t) & \cdots & \sum_{t=1}^{N} \varphi_K^2(t) \end{bmatrix}, \qquad c = \frac{1}{N} \begin{bmatrix} \sum_{t=1}^{N} \varphi_1(t) y(t) \\ \vdots \\ \sum_{t=1}^{N} \varphi_K(t) y(t) \end{bmatrix}$$

Information matrix

Correlation vector

# Example: First-order ARMA model

$$y(t) = ay(t-1) + gu(t-1) + e(t)$$

- Linear regression representation

$$\varphi_1(t) = y(t-1) \qquad \theta = \begin{bmatrix} a \\ g \end{bmatrix} \qquad y(t) = \theta_1\varphi_1(t) + \theta_2\varphi_2(t) + e(t)$$
$$\varphi_2(t) = u(t-1)$$

$$\hat{\theta} = \left(\Phi^T\Phi\right)^{-1}\Phi^T y$$

- This (type of) approach is considered in most of the technical literature on identification
- Matlab Identification Toolbox
  – Limited industrial use
- Fundamental issue:
  – Small error in *a* might mean large change in the system response

Lennart Ljung,
*System Identification: Theory for the User,* 2nd Ed, 1999

# Regularization

- Linear regression, where $\Phi^T\Phi$ is ill-conditioned
- Instead of $||e||^2 \rightarrow \min$ solve a regularized problem

$$\|e\|^2 + r\|\theta\|^2 \rightarrow \min$$

$$\boxed{y = \Phi\theta + e}$$

where $r$ is a small regularization parameter
- A.N.Tikhonov (1963)
  - see http://solon.cma.univie.ac.at/~neum/ms/regtutorial.pdf
- Regularized solution

$$\hat{\theta} = \left(\Phi^T\Phi + rI\right)^{-1}\Phi^T y$$

- Cut off the singular values of $\Phi$ that are smaller than $r$

# Regularization

- Analysis through SVD (singular value decomposition)

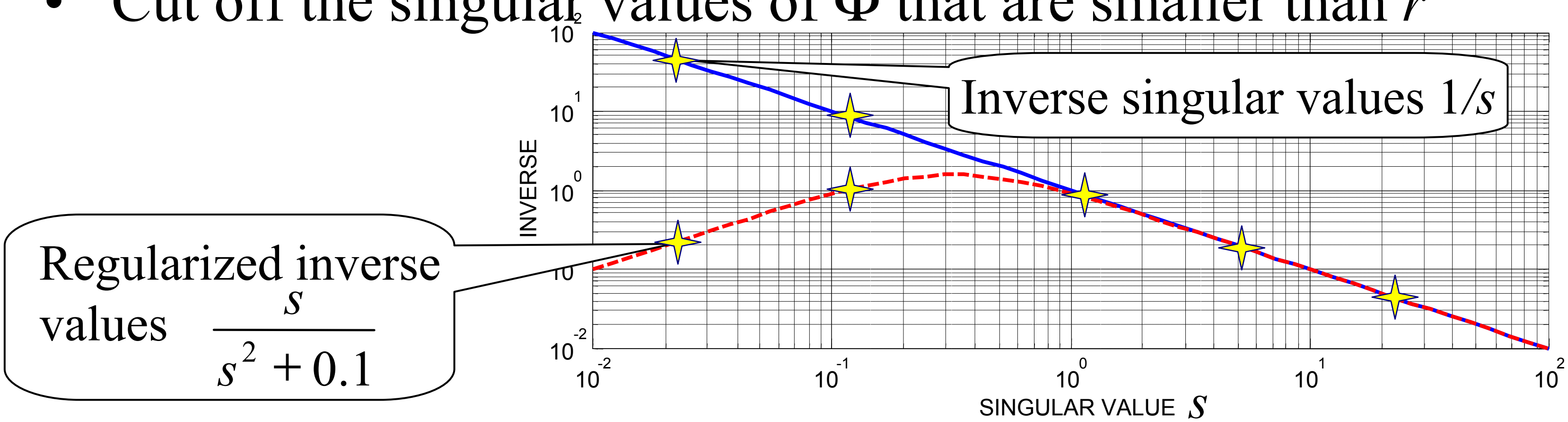
$$\Phi = USV^T \qquad V \in R^{K,K}; U \in R^{N,K}; S = \mathrm{diag}\{s_j\}_{j=1}^{K}$$

$$U^T U = VV^T = I$$

- Regularized solution

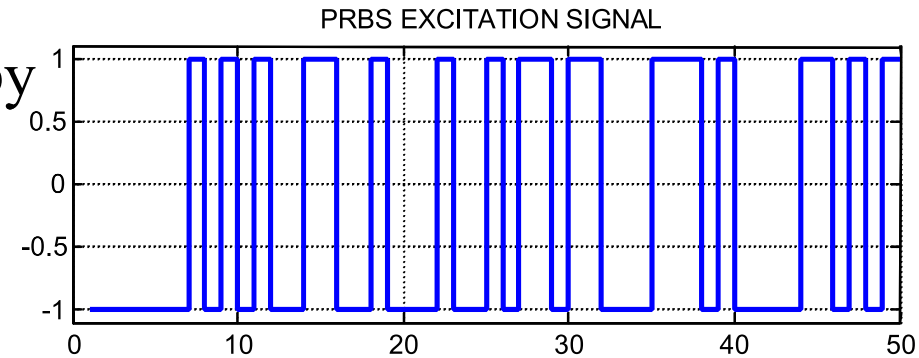
$$\hat{\theta} = \left(\Phi^T \Phi + rI\right)^{-1} \Phi^T y = V\left[\mathrm{diag}\left\{\frac{s_j}{s_j^2 + r}\right\}_{j=1}^{K}\right] U^T y$$

- Cut off the singular values of $\Phi$ that are smaller than $r$



# Linear regression for FIR model

- Identifying impulse response by applying multiple steps
- PRBS excitation signal
- FIR (impulse response) model




$$y(t) = \sum_{k=1}^{K} h(k)u(t-k) + e(t)$$

PRBS =
Pseudo-Random Binary Sequence
See `IDINPUT` in Matlab

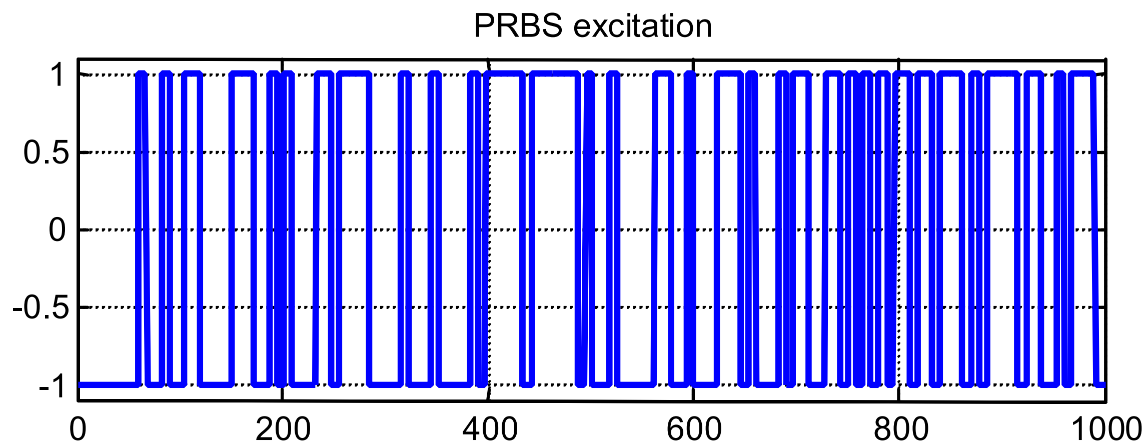
- Linear regression representation

$$\begin{matrix} \varphi_1(t) = u(t-1) \\ \vdots \\ \varphi_K(t) = u(t-K) \end{matrix}, \Phi = \begin{bmatrix} u(t-1) & u(t-2) & \dots & u(t-K) \\ \vdots & \vdots & \ddots & \vdots \\ u(t-N) & u(t-N-1) & \dots & u(t-N-K+1) \end{bmatrix}, \theta = \begin{bmatrix} h(1) \\ \vdots \\ h(K) \end{bmatrix}$$

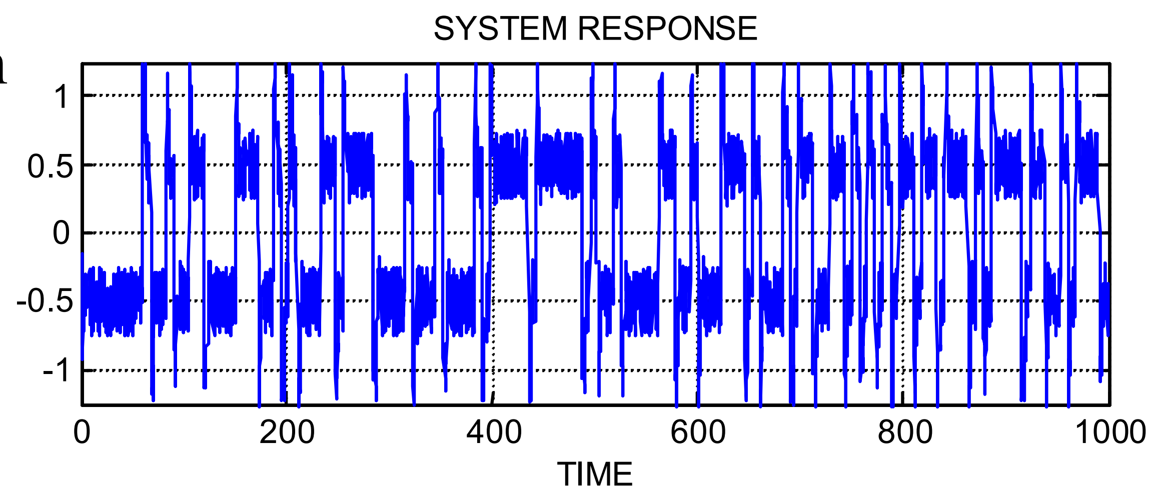
Regularized LS solution: $\hat{\theta} = \left(\Phi^T \Phi + rI\right)^{-1} \Phi^T y$

# Example: FIR model ID

- PRBS excitation input

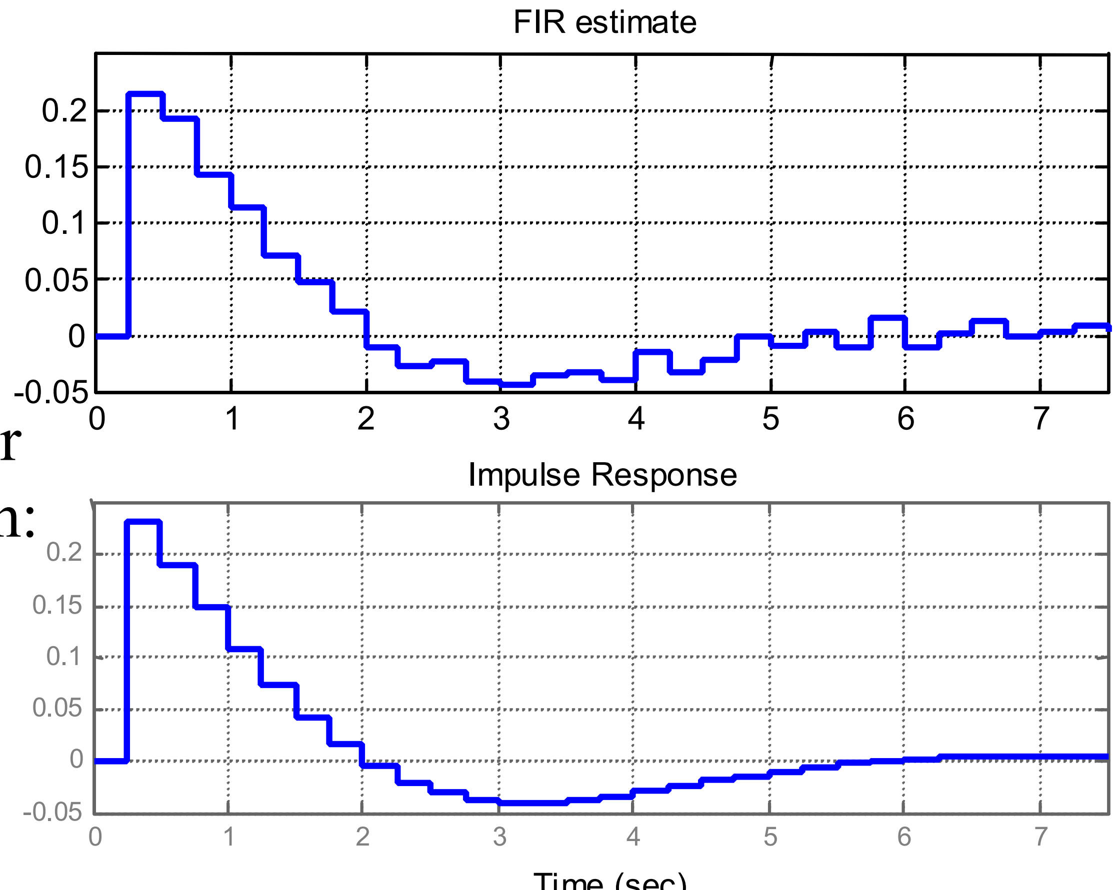



- Simulated system output: 4000 samples, random noise of the amplitude 0.5



# Example: FIR model ID

- Linear regression estimate of the FIR model
- Impulse response for the simulated system:




```
H = tf([1 .5],[1 1.1 1]);
P = c2d(H,0.25);
```

# Nonlinear parametric model ID

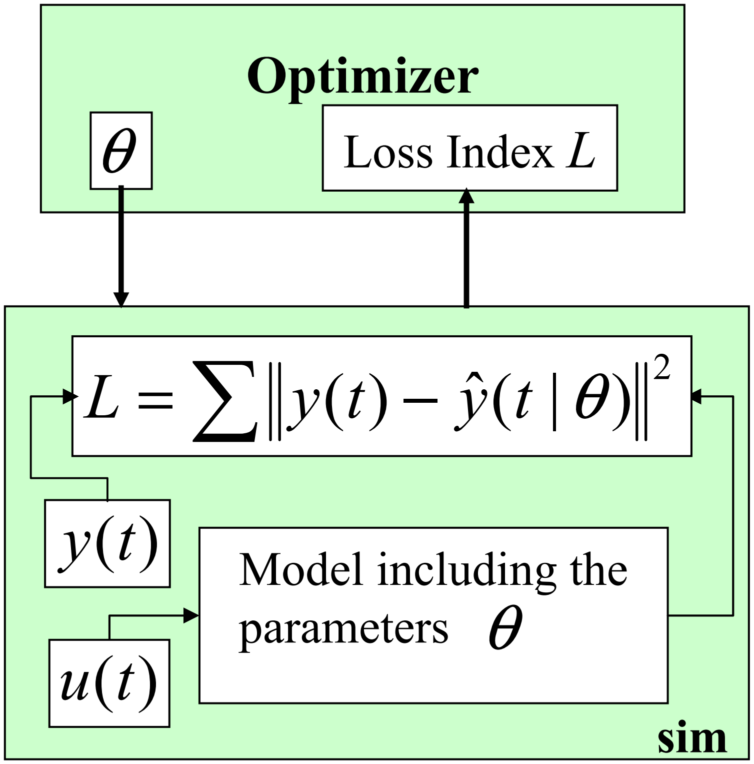
- Prediction model depending on the unknown parameter vector $\theta$

$$u(t) \rightarrow \text{MODEL}(\theta) \rightarrow \hat{y}(t \mid \theta)$$

- Nonlinear regression: loss index

$$L = \sum \left\| y(t) - \hat{y}(t \mid \theta) \right\|^2 \rightarrow \min$$

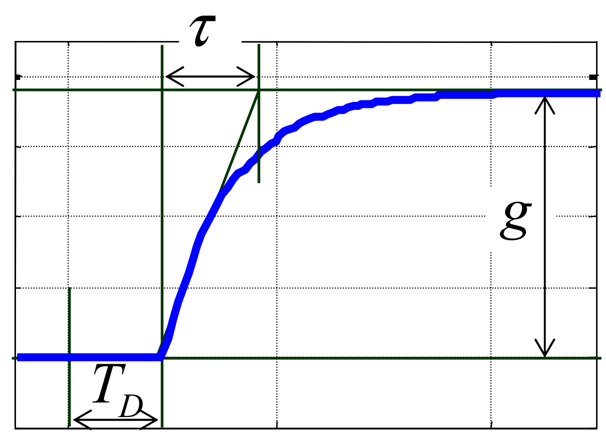
- Iterative numerical optimization. Computation of $L$ as a subroutine




Lennart Ljung, "Identification for Control: Simple Process Models," *IEEE Conf. on Decision and Control*, Las Vegas, NV, 2002

# Parametric SysID of step response

- First order process with deadtime
- Most common industrial process model
- Response to a control step applied at $t_B$

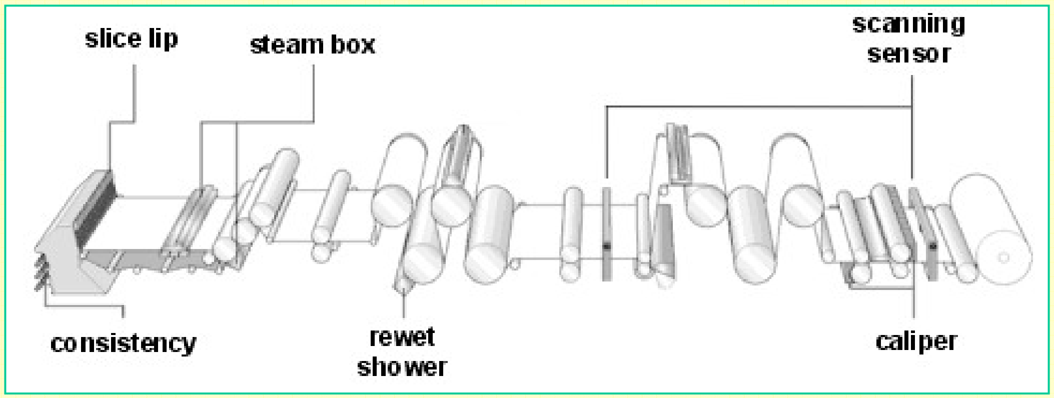



$$y(t\,|\,\theta) = \gamma + \begin{cases} g\left(1 - e^{(t - t_B - T_D)/\tau}\right), & \text{for } t > t_B - T_D \\ 0, & \text{for } t \le t_B - T_D \end{cases}$$

$$\theta = \begin{bmatrix} \gamma \\ g \\ \tau \\ T_D \end{bmatrix}$$

**Example:**

Paper machine process



# Step1: Gain and Offset Estimation

**Two-step approach: linear regression + nonlinear regression**

- For given $\tau, T_D$, the modeled step response can be presented in the form

$$y(t \mid \theta; \tau, T_D) = \gamma + g \cdot y_1(t \mid \tau, T_D)$$

- This is a linear regression

$$y(t \mid \theta; \tau, T_D) = \sum_{k=1}^{2} \theta_k \varphi_k(t) \qquad \begin{aligned} \theta_1 &= g \\ \theta_2 &= \gamma \end{aligned} \qquad \begin{aligned} \varphi_1(t) &= y_1(t \mid \tau, T_D) \\ \varphi_2(t) &= 1 \end{aligned}$$

- Parameter estimate and prediction for given $\tau, T_D$

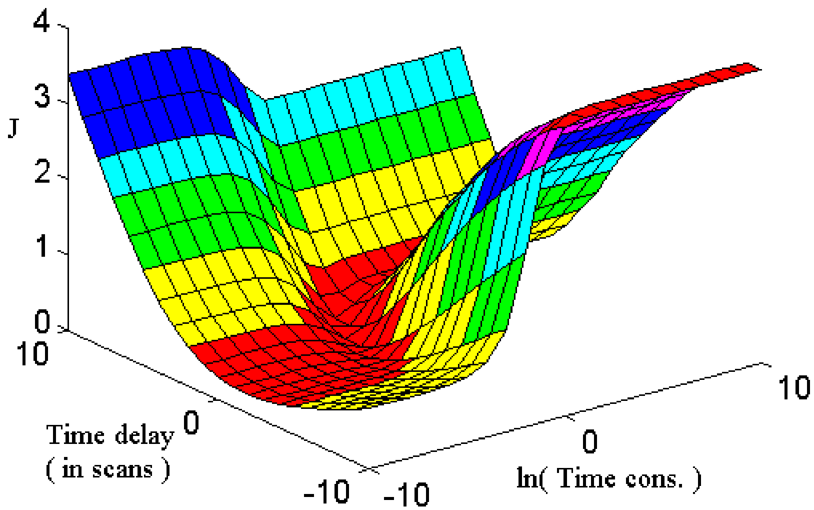
$$\hat{\theta} = \hat{\theta}(\tau, T_D) = \left(\Phi^T \Phi\right)^{-1} \Phi^T y \qquad \hat{y}(t \mid \tau, T_D) = \hat{\gamma} + \hat{g} \cdot y_1(t \mid \tau, T_D)$$

# Step 2: Rise Time & Dead Time Estimation

- For any given $\tau, T_D$, the loss index is

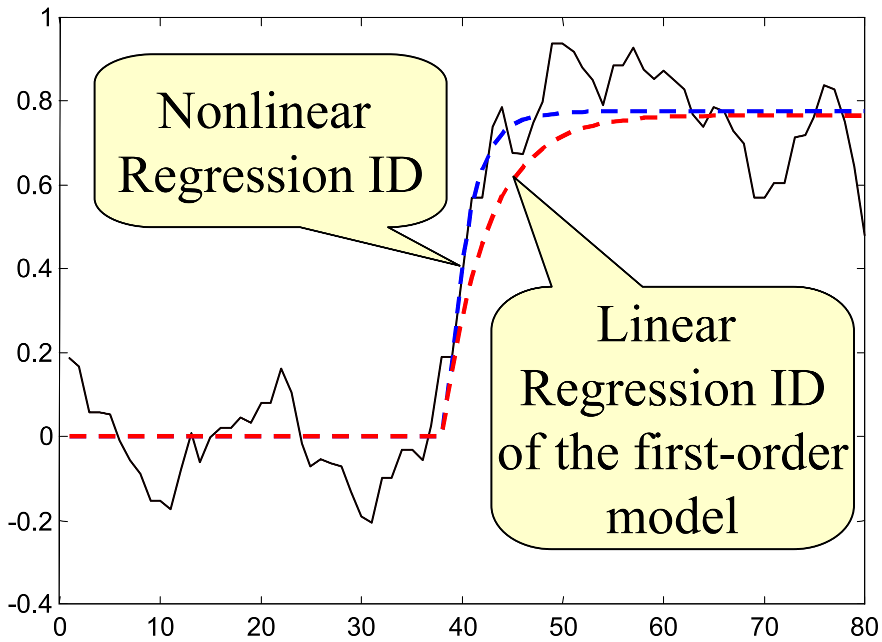
$$L = \sum_{t=1}^{N} \left| y(t) - \hat{y}(t \mid \tau, T_D) \right|^2$$

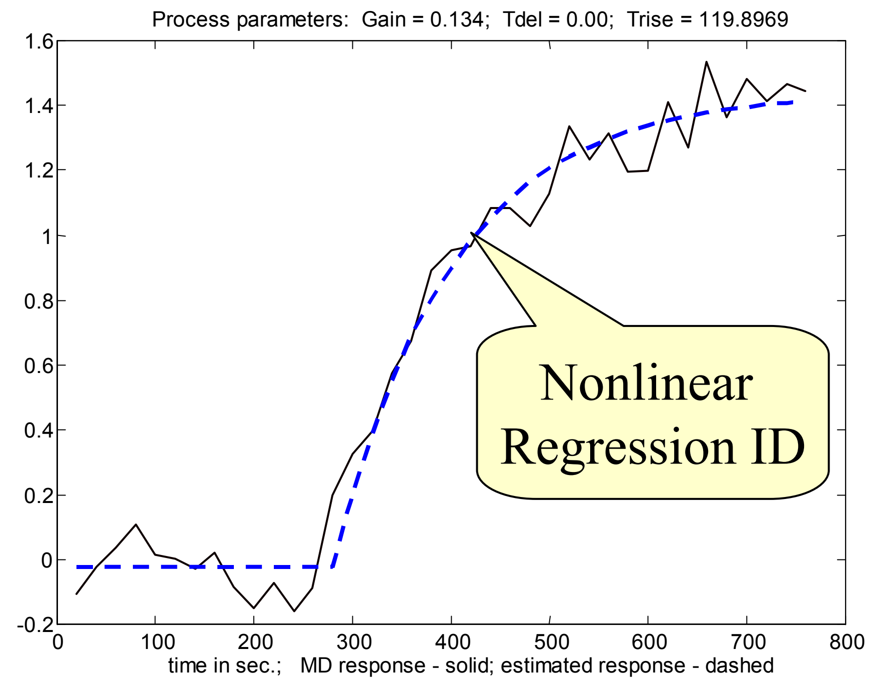
- Grid $\tau, T_D$ and find the minimum of $L = L(\tau, T_D)$



# Examples: Step Response ID

- Identification results for real industrial process data
- This algorithm works in an industrial tool used in 500+ industrial plants, many processes each

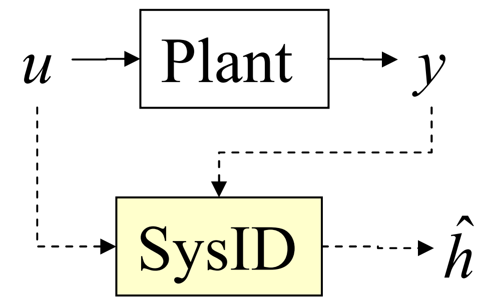


# Linear Filtering in SysID

- A trick that helps: pre-filter data
- Consider data model

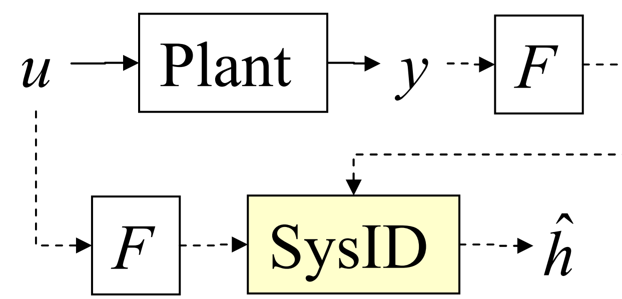
$$y = h * u + e$$




- $F$ is a linear filtering operator, usually LPF

$$\underbrace{Fy}_{y_f} = F(h * u) + \underbrace{Fe}_{e_f}$$

$$F(h * u) = (Fh) * u = h * (Fu)$$

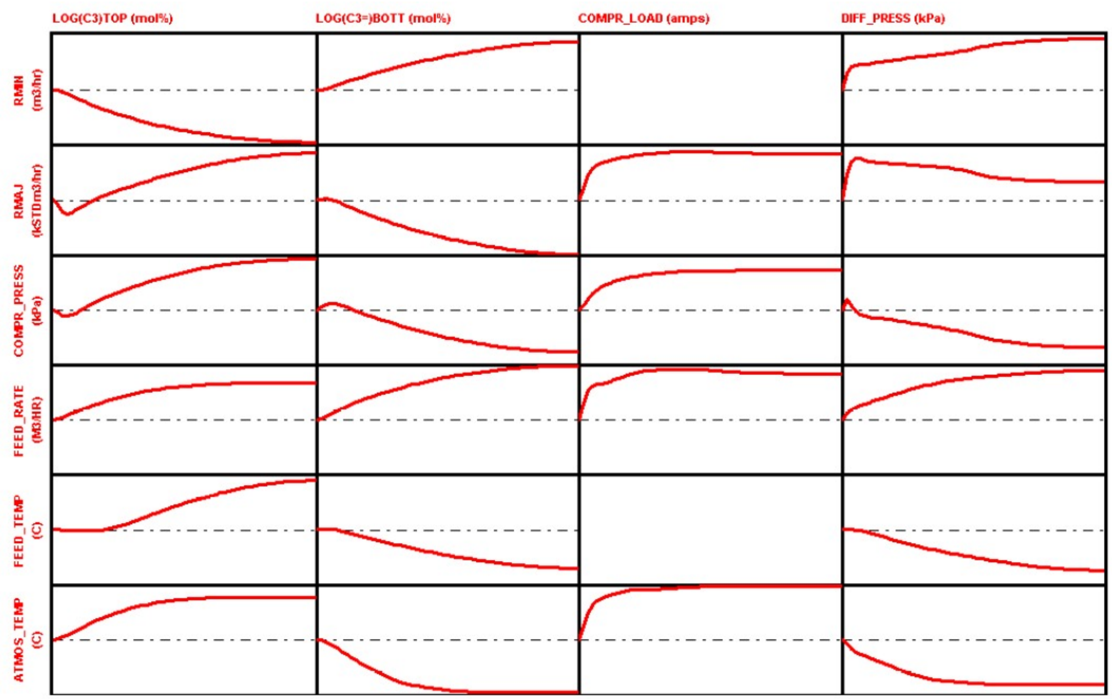



- Can estimate $h$ from filtered $y$ and filtered $u$
- Or can estimate filtered $h$ from filtered $y$ and 'raw' $u$
- Pre-filter bandwidth limits the estimation bandwidth

# Multivariable Identification

- Step/impulse response identification is a key part of the industrial multivariable Model Predictive Control packages

- Apply SISO ID to various input/output pairs
- Need *n* tests: excite each input in turn and collect all outputs at that




**Dynamic Matrix for Propylene/Propane Splitter**