

# Automatic Control 2

## System identification

Prof. Alberto Bemporad

University of Trento



Academic year 2010-2011

# Model identification

- The design of a control system requires a *mathematical model* of the dynamics of the process
- Often a dynamical model can be difficult to obtain due to the complexity of the process, whose dynamics may be even (partially or completely) unknown
- Even if we have a mathematical model, sometimes this is too complex to base a controller design on it (large state dimensions, nonlinearities, etc.)
- Model reduction is a way to go, but requires a (linear) model to start with

*System identification* is a procedure to build a mathematical model of the dynamics of a system from **measured data**

---

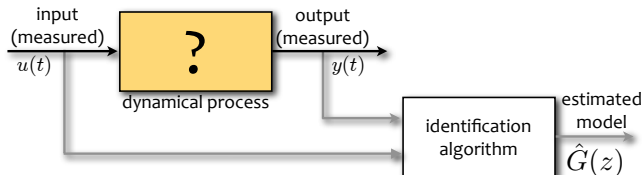
Lecture based on

[1] L. Ljung, "System Identification," Control Systems Handbook (W. Levine ed.), CRC Press, pp. 1033–1054, 1995

More exhaustive reference:

[2] L. Ljung, "System Identification: Theory for the User," Prentice Hall, 1987

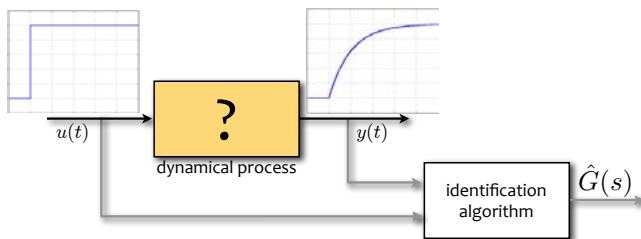
# Model identification



Different types of identification:

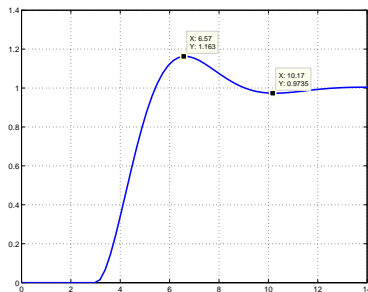
- **White box**: model structure based on first principles (e.g., Newton's law), model parameters estimated from measured data
- **Grey box**: model structure partially known from first principles, the rest is reconstructed from data
- **Black box**: model structure and its parameters completely unknown, they are only estimated from I/O data

# Step-response identification



- Excite the process with a step  $u(t) = \mathbb{I}(t)$ , record output response  $y(t)$
- Observe the shape of  $y(t)$  and reconstruct  $G(s)$   
(1st-order response ? 2nd-order undamped response ? Any delay ? ...)
- Mostly used in process control: excitation experiment is easily done, superposition of effects can be used in the multivariable case to identify each entry  $G_{ij}(s)$  of the *transfer matrix*  $G(s)$  at the time

# Example of step-response identification



The response "looks like" a 2nd-order undamped system response with delay

$$G(s) = \frac{e^{-\tau s}}{1 + \frac{2\zeta}{\omega_n}s + \frac{1}{\omega_n^2}s^2}$$

- Peak overshoot formula:  $\hat{s} = e^{-\frac{\pi\zeta}{\sqrt{1-\zeta^2}}} \rightarrow \zeta = \frac{-\ln\hat{s}}{\sqrt{(\ln\hat{s})^2 + \pi^2}}$   $\hat{s} = 0.163 \rightarrow \zeta \approx 0.5$
- Damped natural frequency formula:  $\omega_d = \omega_n \sqrt{1 - \zeta^2} = \frac{2\pi}{\Delta T}$ , where  $\frac{\Delta T}{2} =$  time interval between two consecutive positive and negative peaks  
 $\omega_n = \frac{2\pi}{\Delta T \sqrt{1-\zeta^2}}$   $\frac{\Delta T}{2} = 10.17 - 6.57 \rightarrow \omega_n \approx 1$
- Delay is estimated by visual inspection  $\tau = 3$

# Black-box identification via linear regression

- Consider the black-box *ARX (AutoRegressive eXogenous)* model

$$y(k) + a_1 y(k-1) + \dots + a_{n_a} y(k-n_a) = b_1 u(k-n_k) + \dots + b_{n_b} u(k-n_k-n_b+1) + e(k)$$

where  $e(k)$  is zero-mean white noise

- We can predict the next output value given previous observations

$$y(k) = -a_1 y(k-1) - \dots - a_{n_a} y(k-n_a) + b_1 u(k-n_k) + \dots + b_{n_b} u(k-n_k-n_b+1) + e(k)$$

- or in more compact form

$$y(k) = \varphi'(k)\theta + e(k)$$

$$\theta = \begin{bmatrix} a_1 & \dots & a_{n_a} & b_1 & \dots & b_{n_b} \end{bmatrix}' \quad \text{unknown parameter vector}$$

$$\varphi(k) = \begin{bmatrix} -y(k-1) & \dots & -y(k-n_a) & u(k-n_k) & \dots & u(k-n_k-n_b+1) \end{bmatrix}'$$

regressor

# Black-box identification via linear regression

- We denote by  $\hat{y}(k|\theta) = \varphi'(k)\theta$  the output prediction to emphasize that the estimate of  $y(k)$  depends from past data and on the parameter vector  $\theta$  ( $e(k) = 0$  is the best we can assume)
- We don't know  $\theta$ , but we have collected a set  $Z^N$  of measured data

$$Z^N = \{u(-n), y(-n), \dots, u(N-1), y(N-1)\}, \quad n = \max\{n_a, n_b + n_k - 1\}$$

- We use the *least-squares method* to estimate the vector  $\theta^*$  that best makes  $\hat{y}(k|\theta)$  fit  $y(k)$

$$\theta^* = \arg \min_{\theta} \{V(\theta, Z^N)\}$$

with

$$V(\theta, Z^N) = \frac{1}{N} \sum_{k=0}^{N-1} (y(k) - \hat{y}(k|\theta))^2 = \frac{1}{N} \sum_{k=0}^{N-1} (y(k) - \varphi'(k)\theta)^2$$

# Black-box identification via linear regression

- $V(\theta, Z^N)$  is a quadratic function of  $\theta$ . We find the minimum by zeroing the derivative of  $V$

$$0 = \frac{d}{d\theta} V_N(\theta, Z^N) = \frac{2}{N} \sum_{k=0}^{N-1} \varphi(k) (y(k) - \varphi'(k)\theta)$$

or

$$\sum_{k=0}^{N-1} \varphi(k)y(k) = \sum_{k=1}^N \varphi(k)\varphi'(k)\theta$$

- The best parameter vector we can choose is therefore

$$\theta^* = \left[ \sum_{k=0}^{N-1} \varphi(k)\varphi'(k) \right]^{-1} \sum_{k=0}^{N-1} \varphi(k)y(k)$$

<b>MATLAB</b>
---------------

$\theta = \text{arx}(Z^N, [n_a \ n_b \ n_k])$
---



# Black-box identification: the general procedure

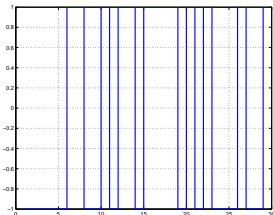
- **Experiment design:** What kind of input excitation  $u(k)$  to apply ?
- **Model structure:** Which class of models do I choose to fit my data ?
- **Fit criterion** between data and model: How do I best choose the model within that class (=the parameter vector) ?
- **Validation criterion:** Is the model that I have identified good enough to reproduce the dynamics of the process ?

# Experiment design

- Collecting data is a very crucial (and most expensive) step
- Some theory is available, as well as some practical rules
- The data set  $Z^N$  should be as *informative* as possible to fully identify the model
- Example:  $u(t) = \sin(\omega t)$  is **not** good, as only  $G(j\omega)$  would be captured !
- The input signal must at least contain as many different frequencies as the order of the chosen structure of linear models
- Step responses are not ideal but ok:  $|\mathcal{F}[\mathbb{I}(t)]| = \frac{1}{\omega}$  has in infinite number of frequencies (although decreasing in amplitude)
- *Pseudo-random binary signals* (PRBS) randomly switching between  $\pm 1$  are a good choice

```
MATLAB
```

```
>> u=idinput(N,'PRBS');
```



# Model structure

- A linear system with additive disturbance  $v(k)$  can be described as

$$y(k) = G(z)u(k) + v(k)$$

where  $G(z)$  is a transfer function ( $z^{-m}x(k) = x(k - m)$ )

$$G(z) = \frac{B(z)}{A(z)} = \frac{b_0 + b_1 z^{-1} + \dots + b_{n_b} z^{-n_b}}{1 + a_1 z^{-1} + a_2 z^{-2} + \dots + a_{n_a} z^{-n_a}}$$

- or, equivalently, as the difference equation

$$\begin{aligned} y(k) = & -a_1 y(k-1) - a_2 y(k-2) - \dots - a_{n_a} y(k-n_a) \\ & + b_0 u(k) + b_1 u(k-1) + \dots + b_{n_b} u(k-n_b) + v(k) \end{aligned}$$

# Model structure

- The disturbance  $v(k)$  is not necessarily white noise, but can be *colored noise*

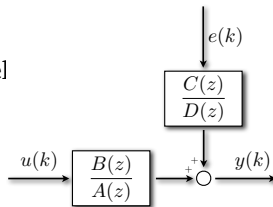
$$v(k) = H(z)e(k)$$

where  $e(k)$  is white noise and  $H(z)$  is another transfer function

$$H(z) = \frac{C(z)}{D(z)} = \frac{1 + c_1 z^{-1} + \dots + c_{n_c} z^{-n_c}}{1 + d_1 z^{-1} + d_2 z^{-2} + \dots + d_{n_d} z^{-n_d}}$$

- The overall model is called *Box-Jenkins (BJ)* model

$$y(k) = \frac{B(z)}{A(z)}u(k) + \frac{C(z)}{D(z)}e(k)$$



## Model structure – Special cases

- *Output Error (OE)* model:  $v(k)$  is white noise ( $C(z) = D(z) = 1$ )

$$y(k) = \frac{B(z)}{A(z)}u(k) + e(k)$$

- *Auto-Regressive Moving-Average with eXogenous variable (ARMAX)* model:  
 $G(z)$  and  $H(z)$  have the same denominator ( $A(z) = D(z)$ )

$$A(z)y(k) = B(z)u(k) + C(z)e(k)$$

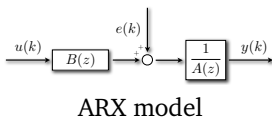
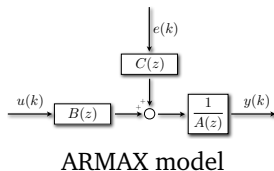
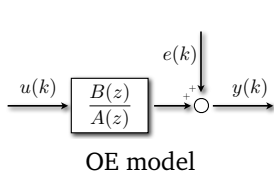
- ARX models are a particular case of ARMAX models ( $C(z) = 1$ )

$$A(z)y(k) = B(z)u(k) + e(k)$$

- ARX and ARMAX models are the most used

# Model structure

- Differently from BJ models, in ARMAX models  $v(k)$  and  $u(k)$  are filtered by the same dynamics  $\frac{1}{D(z)}$
- This is justified if the source of disturbance enters early in the process, together with the input  
Example: in airplanes, the disturbances from wind blasts create the same kind of forces on the airplane as the deflections of the control surfaces
- ARX models are the simplest to compute numerically



# Fit criterion

- Let's consider BJ models, which is the most general structure
- Let  $\theta$  collect all the parameters in the transfer functions  $G(z)$  and  $H(z)$  to be estimated from data

$$y(k) = G(z, \theta)u(k) + H(z, \theta)e(k)$$

$$H^{-1}(z, \theta)y(k) = H^{-1}(z, \theta)G(z, \theta)u(k) + e(k)$$

$$y(k) + H^{-1}(z, \theta)y(k) = y(k) + H^{-1}(z, \theta)G(z, \theta)u(k) + e(k)$$

- Finally, we get

$$y(k) = (1 - H^{-1}(z, \theta))y(k) + H^{-1}(z, \theta)G(z, \theta)u(k) + e(k)$$

- Note that  $1 - H^{-1}(z, \theta) = h_1 z^{-1} + h_2 z^{-2} + \dots$  for some coefficients  $\{h_i\}_{i=1}^{\infty}$

# Fit criterion

- For  $e(k) = 0$  (=the best estimate of  $e(k)$  we can make), the one-step ahead prediction of  $y(k)$  based on previous measurements is

$$\hat{y}(k|\theta) = (1 - H^{-1}(z, \theta))y(k) + H^{-1}(z, \theta)G(z, \theta)u(k)$$

- Assuming we have enough data ( $N \geq \max(n_a, n_b, n_c, n_d)$ ), we compute the *residual*

$$\epsilon(k|\theta) = y(k) - \hat{y}(k|\theta)$$

- The most used fit criterion is

$$V(\theta, Z^N) = \frac{1}{N} \sum_{k=0}^{N-1} \epsilon^2(k|\theta)$$

- The optimal vector  $\theta^*$  is determined by solving the optimization problem

$$\theta^* = \arg \min_{\theta} V(\theta, Z^N)$$



# Complexity

- Ideally  $\epsilon$  should depend linearly on  $\theta$ , so we can get the explicit solution of a least-squares problem. This only happens for ARX models
- Besides choosing the model structure (ARX, ARMAX, etc.) we also need to decide the *order* of the model, i.e., the number of free parameters to optimize
- A small number of parameters could make the model too simple, and not able to explain the data
- A large number of parameters could make the model more complex than we need (and *overfit* the data in  $Z^N$ )
- How to choose the right model complexity?

# Model validation

- Usually to avoid being fooled by overfitting the data set we split  $Z^N$  in two subsets: *estimation data*  $Z_{est}$  and *validation data*  $Z_{val}$ :
  - $Z_{est}$  is used to compute the optimal parameter vector  $\theta^*$
  - $Z_{val}$  is used to see how the estimated model behaves on fresh data
- A validation criterion is to look at one-step prediction errors

$$V(\theta^*, Z_{val}) = \frac{1}{N} \sum_{k=0}^{N-1} (y(k) - \hat{y}(k|\theta^*))^2$$

nothing to optimize here,  
just substitute  $\theta^*$ ,  $Z_{val}$  and evaluate

- Another validation criterion is to simulate the model completely in “open-loop”

$$y_{sim}(k, \theta^*) = G(z, \theta^*)u(k)$$

and to look at

$$\frac{1}{N} \sum_{k=0}^{N-1} (y(k) - y_{sim}(k, \theta^*))^2$$

(or just observe how much the plots of  $y(k)$  and  $y_{sim}(k, \theta^*)$  differ)

# Model validation – Residual analysis

- Ideally the prediction error (or *prediction residual*)  $\epsilon(k|\theta)$  should be white noise and uncorrelated with  $u(k)$
- To test whiteness of  $\epsilon(k|\theta)$  we compute the *auto-correlation* function

$$R_{\epsilon}(\tau) = \frac{1}{N} \sum_{k=0}^{N-\tau-1} \epsilon(k+\tau|\theta)\epsilon(k|\theta)$$

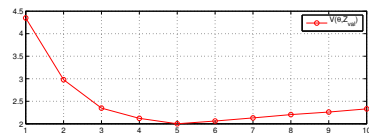
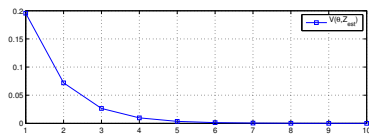
- To test correlation between  $\epsilon(k|\theta)$  and  $u(k)$  we compute the sample covariance

$$R_{\epsilon u}(\tau) = \frac{1}{N} \sum_{k=\tau-1}^{N-1} \epsilon(k|\theta)u(k-\tau)$$

- Both  $R_{\epsilon}(\tau)$  and  $R_{\epsilon u}(\tau)$  should be small

# Model selection

- Which model structure to choose (ARX, ARMAX, OE, etc. )? Which model orders  $n_a$ ,  $n_b$ ,  $n_k$ , etc. ?
- Cross-validation** is the procedure that compares the quality of fit of different models, by validating them on a data set where neither of them was estimated
- Let  $\theta_1^*, \dots, \theta_s^*$  a set of optimal parameters for different model structures
- The best model  $\theta_i^*$  is the one for which  $V(\theta_i^*, Z_{val})$  is smallest
- Often  $V_i(\theta^*, Z_{est})$  decreases as the model complexity increases, while  $V_i(\theta^*, Z_{val})$  starts increasing when the model complexity becomes excessive (=overfit of estimation data)



model index  $i$  (growing complexity)

# Model selection

- If fresh validation data are not available (=no cross-validation), we can use the same performance figures, but in addition penalize overfit (we want a good balance between simplicity and accuracy)
- Let  $d_i$  = number of elements of  $\theta_i^*$  (=model complexity)
- We look for the model that minimizes one of the following figures:

- Akaike's Information theoretic Criterion (AIC):

$$\left(1 + \frac{2d_i}{N}\right) \frac{1}{N} \sum_{k=0}^{N-1} \epsilon^2(k|\theta_i^*)$$

- Akaike's Final Prediction Error (FPE):

$$\left(\frac{1 + \frac{d_i}{N}}{1 - \frac{d_i}{N}}\right) \frac{1}{N} \sum_{k=0}^{N-1} \epsilon^2(k|\theta_i^*)$$

- Rissanen's Minimum Description Length (MDL):

$$\frac{1}{N} \sum_{k=0}^{N-1} \epsilon^2(k|\theta_i^*) + \frac{d_i \cdot \ln N}{N}$$

# Example

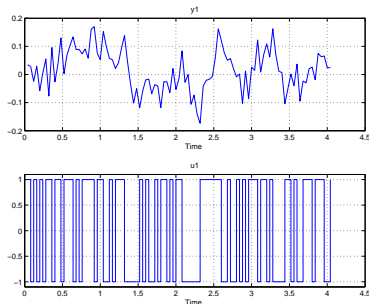
- Assume that the real (unknown) process is the one considered in the “Model reduction” lecture

$$G_c(s) = \frac{s^3 + 11s^2 + 36s + 26}{s^4 + 14.6s^3 + 74.96s^2 + 153.7s + 99.65}$$

- its discrete-time version for sampling time  $T = 0.04$  s is

$$G(z) = \frac{0.03726z^{-1} - 0.09676z^{-2} + 0.08355z^{-3} - 0.024z^{-4}}{1 - 3.464z^{-1} + 4.493z^{-2} - 2.586z^{-3} + 0.5577z^{-4}}$$

- Input excitation: PRBS sequence
- We have 200 samples. The first 100 samples are used for estimation of  $\theta^*$ , the rest for validation

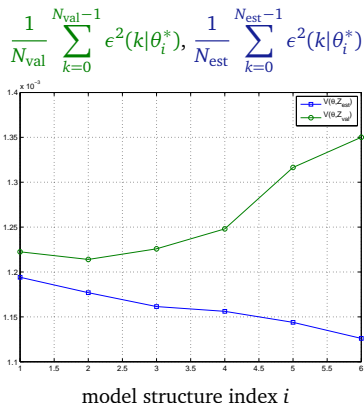


## Example (cont'd)

We try five different ARX model structures  $\text{ARX}(n_a, n_b, n_k)$ :

$$A(z)y(t) = B(z)u(t) + e(t)$$

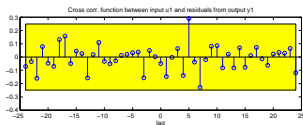
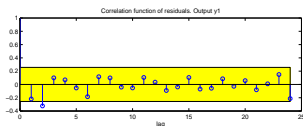
- $i = 1$ :  $\text{ARX}(1,1,1)$
- $i = 2$ :  $\text{ARX}(2,2,1)$
- $i = 3$ :  $\text{ARX}(3,3,1)$
- $i = 4$ :  $\text{ARX}(4,4,1)$
- $i = 5$ :  $\text{ARX}(5,5,1)$
- $i = 6$ :  $\text{ARX}(6,6,1)$



The best model structures are  $i = 2, 3, 4$

# Example (cont'd)

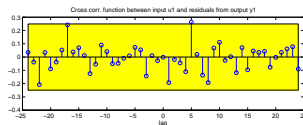
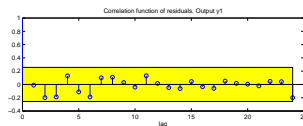
## Residual analysis:



ARX(2,2,1)

$$A(z) = 1 - 0.2153z^{-1} - 0.5624z^{-2}$$

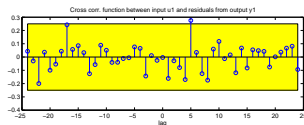
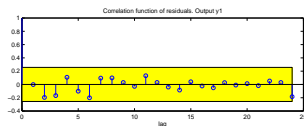
$$B(z) = 0.04041z^{-1} + 0.02456z^{-2}$$



ARX(3,3,1)

$$A(z) = 1 + 0.1228z^{-1} - 0.3396z^{-2} - 0.4444z^{-3}$$

$$B(z) = 0.04014z^{-1} + 0.037z^{-2} + 0.02247z^{-3}$$



ARX(4,4,1)

$$A(z) = 1 + 0.1451z^{-1} - 0.319z^{-2} - 0.4258z^{-3} - 0.03208z^{-4}$$

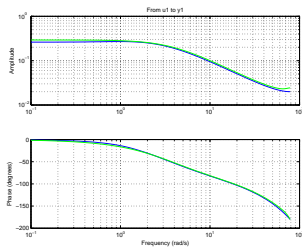
$$B(z) = 0.03912z^{-1} + 0.03826z^{-2} + 0.02476z^{-3} + 0.004177z^{-4}$$

not much different from  
ARX(3,3,1)

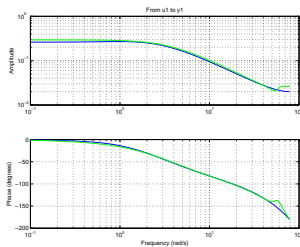


# Example (cont'd)

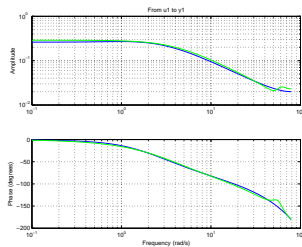
## Compare Bode plots (discrete-time)



ARX(2,2,1)



ARX(3,3,1)





ARX(4,4,1)

# Conclusions

- System identification and control design are complementary: no controller without a model, but identified model only useful for control design purposes
- If model parameters change on-line, one can use *adaptive control*, by identifying the model and changing the controller accordingly in real-time (watch out: closed-loop stability may be an issue !)
- If linear model structures are not able to capture the model well, one should look at nonlinear model structures (like artificial neural networks, piecewise affine functions, and other general function approximation methods)
- In general, the more a-priori knowledge of the process we can exploit (e.g., from physical principles), the better. Sometimes black-box identification fails

# English-Italian Vocabulary

	
system identification least-squares residual adaptive control	<i>identificazione minimi quadrati residuo controllo adattativo</i>

Translation is obvious otherwise.