Multivariable system identification for Model Predictive Control: fundamentals and practice

Prof. Dr.-Ing. Gabriele Pannocchia

University of Pisa, Italy
Department of Civil and Industrial Engineering

Invited lecture for the course on
``Modelling and System Identification"
November 25, 2015

Outline

Introduction and objectives

Conventional feedback control vs. advanced control

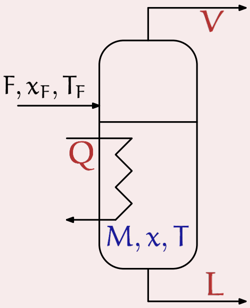
- Conventional **decentralized** control is usually based on PID algorithms:
  - Control action depends only on tracking error: $e(k) = y_s(k) - y(k)$
  - No system model is used by the controller
- Advanced **multivariable** control is designed on a system model:
  - To obtain an LTI controller (e.g., LQR, IMC, H_∞)
  - To solve (repeatedly) optimal control problems (e.g., MPC)

Objectives of this lecture

- Motivate the use of advanced control techniques (MPC)
- Explain the basics of multivariable systems identification
- Discuss the practical issues faced and explain how to deal with them
- Introduce advanced multivariable identification techniques

An example of industrial process: evaporation

Basic concepts



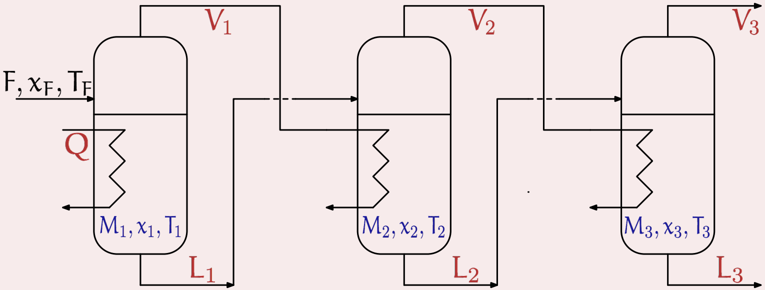
- Evaporation processes aim at concentrating a solution (e.g., sugar) by **removing the solvent** via evaporation
- **Heat** needs to be supplied
- Usually operate at **constant pressure** (hence constant temperature)
- To **minimize operating costs**, they are often arranged in **integrated** multiple stages.

Mass and energy balances

$$\begin{cases} \text{Overall mass balance:} & \frac{dM}{dt} = F - L - V \\ \text{Solute mass balance:} & M\frac{dx}{dt} = Fx_F + (V - F)\,x \\ \text{Energy balance:} & Mc_p\frac{dT}{dt} = Fc_p\,(T_F - T) - V\lambda + Q \end{cases}$$

Process flow diagram of multistage evaporation

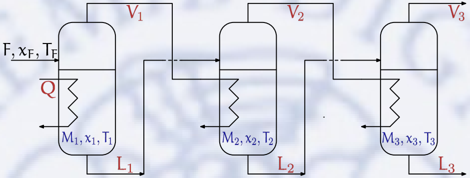
Forward feed triple effect arrangement



Conditions for heat integration

- For heat transfer to be possible: $T_1 > T_2 > T_3$
- This is achieved by operating at decreasing pressures: $p_1 > p_2 > p_3$

Mass and energy balances of multi-stage evaporation



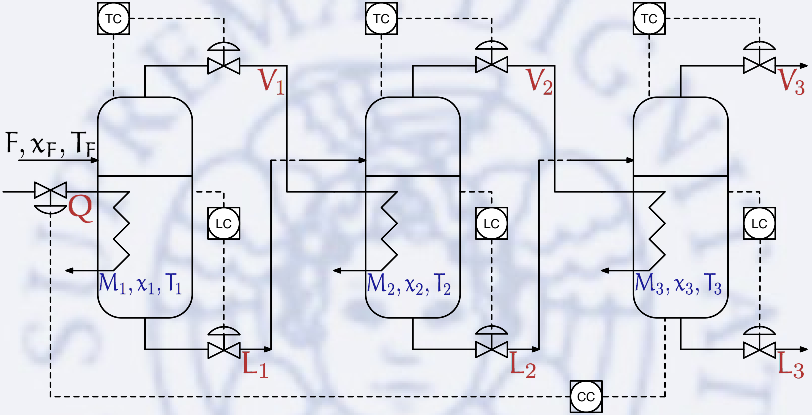
First evaporator

$$\begin{cases} \text{Overall mass balance:} & \frac{dM_1}{dt} = F - L_1 - V_1 \\ \text{Solute mass balance:} & M_1 \frac{dx_1}{dt} = F x_F + (V_1 - F)\, x \\ \text{Energy balance:} & M_1 c_p \frac{dT_1}{dt} = F c_p (T_F - T_1) - V\lambda + Q \end{cases}$$

$i-$th evaporator ($i = 2, 3$)

$$\begin{cases} \text{Overall mass balance:} & \frac{dM_i}{dt} = L_{i-1} - L_i - V_i \\ \text{Solute mass balance:} & M_i \frac{dx_i}{dt} = L_{i-1} x_{i-1} + (V_i - L_{i-1})\, x_i \\ \text{Energy balance:} & M_i c_p \frac{dT_i}{dt} = L_{i-1} c_p (T_{i-1} - T_i) - V_i \lambda_i + V_{i-1} \lambda_{i-1} \end{cases}$$

Conventional control architecture



Decentralized control structure

- Each controlled variable is **paired** with a manipulated variable
- A **SISO PID controller** is used for each pairing

Control issues and objectives

Multivariable system features

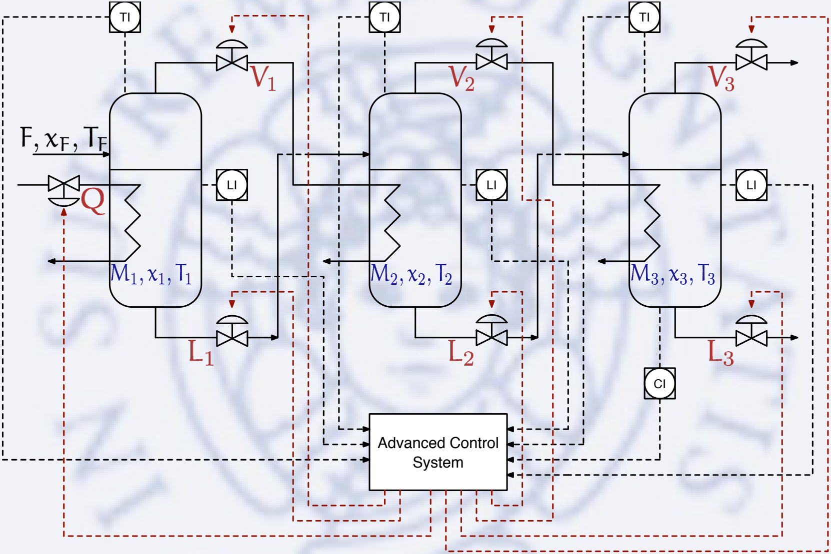
- **Interactions**: each manipulated variable affects more than one controlled variable
- **Directionality**: it is easier to ``move'' the system in certain ``directions'' than in others
- Both manipulated and controlled variable should **satisfy** certain (safety, quality, operation) **constraints**

Opportunities

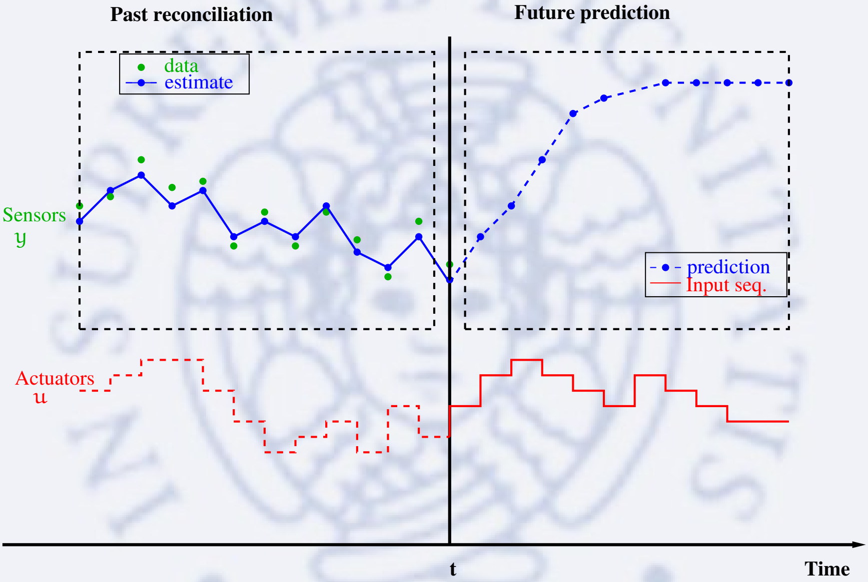
These needs coupled with economic reasons call(ed) for the adoption of advanced optimization based control techniques, able to:

- Control all variables adjusting all manipulated variables **simultaneously**
- **Minimize** energy and cost
- **Respect** constraints

Advanced control architecture



Model predictive control: an introduction



Model predictive control: basic formulation

The optimal control problem

- Given the current state of the system, $x(k)$, solve:

$$\min_{u,x} \sum_{i=0}^{\infty} \ell(x_i, u_i) \qquad \text{subject to:}$$

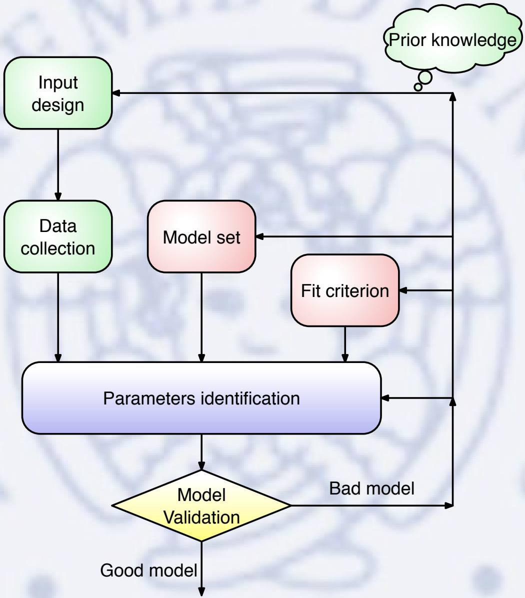
$$\begin{aligned} x_0 &= x(k) && \text{(Initial condition)} \\ x_{i+1} &= \text{model}(x_i, u_i) && \text{(System dynamics)} \\ (x_i, u_i) &\in \mathbb{Z} && \text{(Constraints)} \end{aligned}$$

- Inject the first element of the optimal control sequence: $u^* = (u_0^*, u_1, \dots)$

Linear MPC: role and origin of the model

- In many industrial process model is chosen **linear**
- The identification of a **suitable** model is the **crucial** step for MPC success

The system identification loop



Preliminary tests: objectives and practice

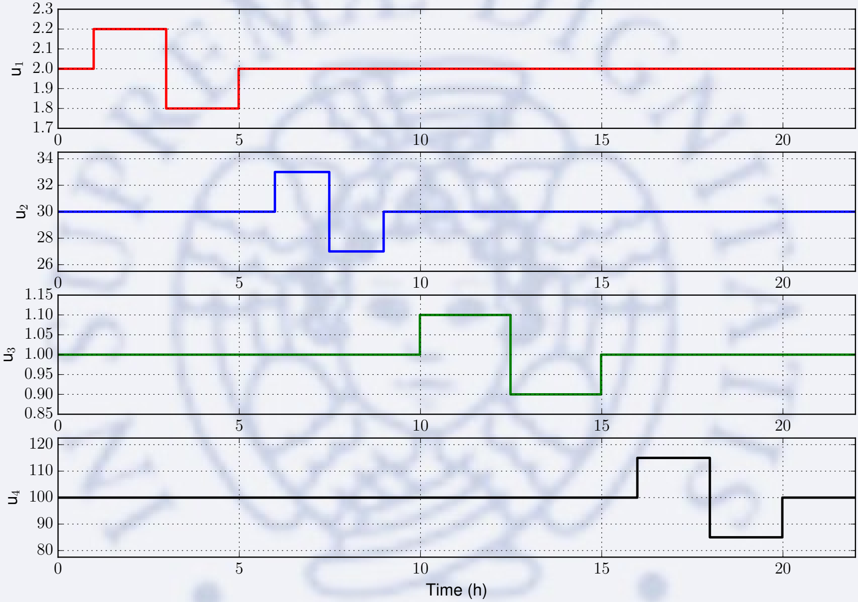
Objectives

- Identification data is(generally) collected during specific **campaigns**
- Test **duration** should be **minimized**, but data should be **informative**

Functional design

- The following **list of variables** are compiled:
  - MV: manipulated variables
  - CV: controlled variables (measurable)
  - DV: disturbance variables (measurable)
- **Instrumentation** (sensors and actuators) may undergo into maintenance before testing
- **Prior knowledge** and/or preliminary tests are used to decide:
  - Amplitude of each MV variation
  - Duration of each MV variation (settling time)

Traditional open-loop step tests



Limitations of step tests

Some useful quantities

- **Autocorrelation** function of a **stationary** stochastic variable $\{u(k)\}$:
$$R_u(\tau) = \mathcal{E}\left(u(k)u(k-\tau)\right)$$
- **Power spectrum** or spectral density
$$\Phi_u(\omega) = \sum_{\tau=-\infty}^{\infty} R_u(\tau)e^{-i\tau\omega}$$

Signals requirements

- Identification signals must have a sufficiently **high power spectrum** mid and low frequency range
- A related property of signals is called **persistent excitation**
- Step signals have **limited** frequency content and do not excite the plant significantly

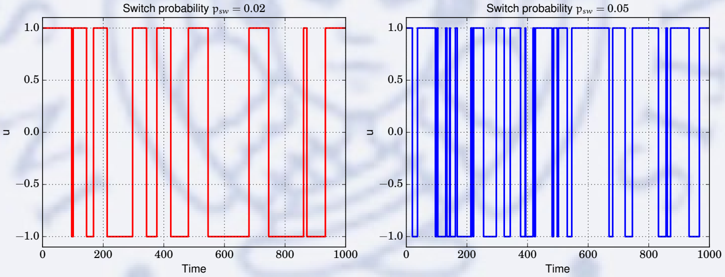
Beyond the step tests

GBN and PRBS

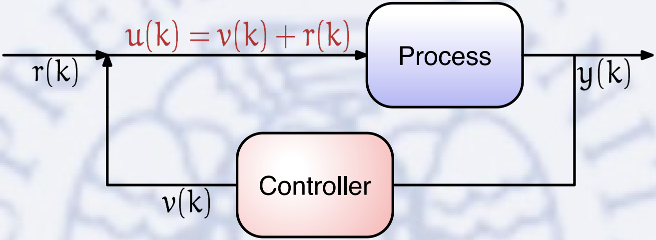
- Generalized Binary Noise (**GBN**) signals are very effective (Zhu, 2001)
- It has **two possible values** $\{+a, -a\}$
- Let $p_{sw} \in (0, 1)$ be the **switching probability**. The signal obeys:

$$\begin{cases} P\left[u(k) = -u_{k-1}\right] = p_{sw} \\ P\left[u(k) = u_{k-1}\right] = 1 - p_{sw} \end{cases}$$

- **PRBS** are similar but **periodic**



Closed-loop tests: basic idea



Basic relations

- **Feedback** is used: $$v(k) = -F(y(k))$$
- **Independent** ``setpoints" are added to $v(k)$: $$u(k) = v(k) + r(k) = -F(y(k)) + r(k)$$
- Setpoints are used to **improve excitations** at higher frequencies

Closed-loop vs open-loop tests

Advantageous features of open-loop signals

- There is **no need** to have a working **controller**
- Identification algorithms are **always applicable** to open-loop data
- **Input variations** (amplitude and duration) defined by the **user**
- Dynamic responses more **easily understood**

Advantageous features of closed-loop signals

- Variations of **outputs** can be **controlled**
- Variations of inputs are **simultaneous**
- Many **studies** report that ``closed-loop data are better suited for **controller design**"

Multivariable data collection

Motivations

- Multivariable signals are more **informative** and excite the system in **several directions**
- The **nonlinearity** is better understood by multivariable signals

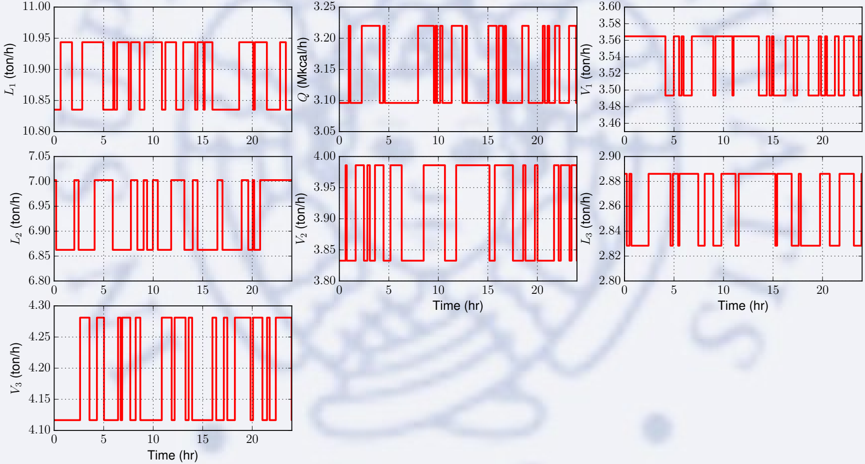
Recommended practice

- **Open-loop** data collection: use **independent** GBN **inputs**
- **Closed-loop** data collection: use **independent** GBN **setpoints**

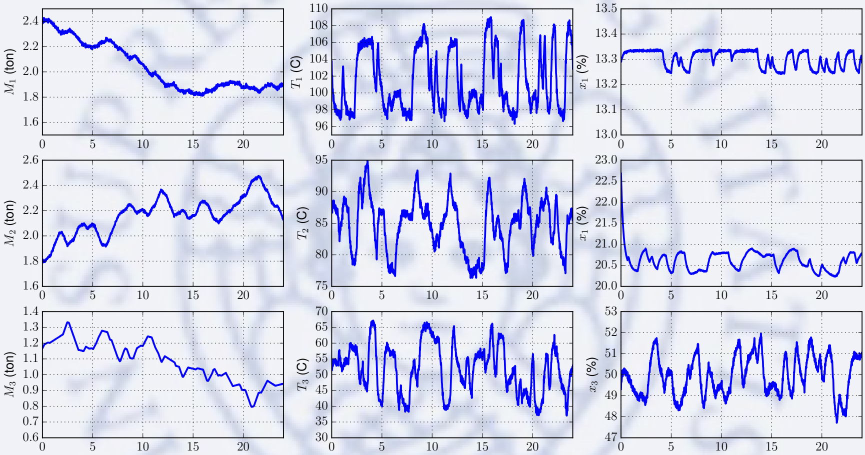
Testing the multi-stage evaporator: inputs

Input design parameters

- Three days of testing using **MIMO**, independent **GBN signals**
- Switch probability $p_{sw} = 0.02$, **minimum switch time** of 4 min



Testing the multi-stage evaporator: outputs



FIR model for SISO systems

Ideal and practical Finite Impulse Response model

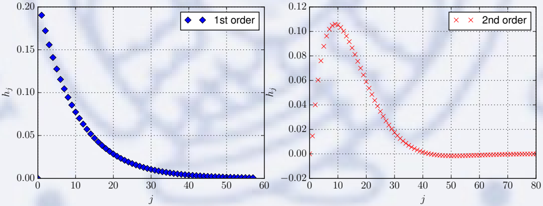
- The **ideal convolution** model in discrete time is:

$$y_k = \sum_{j=1}^{\infty} h_j u_{k-j}$$

with $\{h_j\}$ coefficients of the finite impulse response.

- For **open-loop stable** systems, it follows that: $\lim_{j\to\infty} h_j = 0$
- The **practical FIR** model is ($M > 0$ is the **model horizon**):

$$y_k = \sum_{j=1}^{M} h_j u_{k-j}$$



FIR model identification via least-squares

Linear predictor construction

- Assume **input** and **output data** are available: $[\, u_0, \ldots, u_N \,]$, $[\, y_0, \ldots, y_N \,]$,
- For **each** $k \geqslant M$, write:

$$y_k = g_1 u_{k-1} + g_2 u_{k-2} + \cdots + g_M u_{k-M} + e_k = \varphi_k \theta + e_k$$

where: $\varphi_k = [\, u_{k-1} \; u_{k-2} \; \cdots \; u_{k-M} \,]$, and $\theta = [\, g_1 \; g_2 \; \cdots \; g_M \,]^\top$

- Stack all terms for $k = M, \ldots, N$:

$$\begin{bmatrix} y_M \\ y_{M+1} \\ \vdots \\ y_N \end{bmatrix} = \begin{bmatrix} \varphi_M \\ \varphi_{M+1} \\ \vdots \\ \varphi_N \end{bmatrix} \theta + \begin{bmatrix} e_M \\ e_{M+1} \\ \vdots \\ e_N \end{bmatrix} \Rightarrow \mathbf{y} = \mathbf{\Phi} \theta + \mathbf{e}$$

Least-squares problem and solution

- **Mean Square Error** (MSE) loss function:

$$V_{LS}(\theta) = \frac{1}{N} \sum_{k=M}^{N} e_k^2 = \frac{1}{N} (\mathbf{y} - \mathbf{\Phi}\theta)^\top (\mathbf{y} - \mathbf{\Phi}\theta)$$

- Well known **solution**: $\theta = (\mathbf{\Phi}^\top \mathbf{\Phi})^{-1} \mathbf{\Phi}^\top \mathbf{y} = \mathbf{\Phi}^+ \mathbf{y}$

Multivariable FIR model

Extension to Multiple Input Multiple Output (MIMO) systems

- Consider a system with m **inputs** $(u^{(1)}, u^{(2)}, \ldots, u^{(m)})$ and p **outputs** $(y^{(1)}, y^{(2)}, \ldots, u^{(p)})$
- For **each output**, a Multiple Input Single Output (**MISO**) approach is used

$$
\begin{aligned}
y_k^{(i)} &= \sum_{j=1}^{M} g_j^{(i1)} u_{k-j}^{(1)} + \sum_{j=1}^{M} g_j^{(i2)} u_{k-j}^{(2)} + \cdots + \sum_{j=1}^{M} g_j^{(im)} u_{k-j}^{(m)} + e_k^{(i)} \\
&= \varphi_k^{(1)} \theta^{(i1)} + \varphi_k^{(2)} \theta^{(i2)} + \cdots + \varphi_k^{(m)} \theta^{(im)} + e_k^{(i)}
\end{aligned}
$$

- **Stacking** all terms for $k = M, \ldots, N$ and $\theta^{(i)} = \begin{bmatrix} \theta^{(i1)} & \ldots & \theta^{(im)} \end{bmatrix}^\top$

$$
\mathbf{y}^{(i)} = \mathbf{\Phi} \theta^{(i)} + \mathbf{e}^{(i)} \Rightarrow \theta^{(i)} = \mathbf{\Phi}^{+} \mathbf{y}^{(i)}
$$

Input and output relations

- The user defines **which inputs affect** the response of each output i
- This input/output relations are **decided** using **preliminary tests**

Comments of the FIR model

Good features of FIR models

- Very **little prior knowledge** is required, except which **input/output** coefficients need to be determined
- It is statistically **unbiased** and **consistent**

Bad features of FIR models

- It is **over-parameterized**, and can be **noise sensitive** because the regressor matrix Φ is often **ill-conditioned**
- It is a (very) **high-order** model: **order reduction** may be necessary

Extension to measurable disturbances

- Measurable disturbances are treated as **additional inputs** of the MISO structure

State-space systems: basic definitions

LTI system: **innovation** and **predictor** forms

Innovation form:

$$x_{k+1} = Ax_k + Bu_k + Ke_k$$
$$y_k = Cx_k + e_k$$

Predictor form ($A_K = A - KC$):

$$x_{k+1} = A_K x_k + Bu_k + Ky_k$$
$$y_k = Cx_k + e_k$$

with dimensions: $x \in \mathbb{R}^n$, $u \in \mathbb{R}^m$, $y \in \mathbb{R}^p$

Main assumptions

- (A, B) controllable, (A, C) observable, and $A_K = A - KC$ **strictly Hurwitz**
- The innovation $\{e_k\}$ is a **stationary**, zero mean, white noise process:
$$\mathcal{E}(e_j e_j^\top) = R_e, \qquad \mathcal{E}(e_i e_j^\top) = 0 \quad \text{for } i \neq j$$
- Input $\{u_k\}$ and output $\{y_k\}$ data sequences are available for $k = 0, \ldots, N$.

Indirect routes to get this LTI model

- They can be obtained **via realization** of input-output models
- Often the obtained **order** is quite **high**, with **no** perceivable **advantages**

Subspace identification algorithms: introduction

Motivations

- Multivariable input/output systems identification requires **prior knowledge** or **trial-and-error** to determine the system orders
- Input/output systems identification is always **MISO**, whereas in some cases it would desirable to directly identify **MIMO** models
- Identification of **advanced multivariable** models (e.g., ARMAX, OE, etc.) require solution of large **nonconvex nonlinear programming** problems

Features

- **Direct** identification of an LTI state-space model
- Applicable to both **MIMO** and MISO approaches
- **Compact** multivariable state-space representation
- Very **little prior knowledge** required (an **upper bound** to the order)
- Based on **reliable linear algebra** decompositions

Basic SID algorithm: derivation

An r-step prediction model

- For each k, define an r-step prediction model:

$$\underbrace{\begin{bmatrix} y_k \\ y_{k+1} \\ y_{k+2} \\ \vdots \\ y_{k+r-1} \end{bmatrix}}_{\mathbf{y}_k} = \underbrace{\begin{bmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^{r-1} \end{bmatrix}}_{\Gamma_r} x_k + \underbrace{\begin{bmatrix} 0 & \cdots & \cdots & 0 \\ CB & 0 & \cdots & 0 \\ CAB & CB & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ CA^{r-2}B & CA^{r-3}B & \cdots & 0 \end{bmatrix}}_{H_r^u} \underbrace{\begin{bmatrix} u_k \\ u_{k+1} \\ u_{k+2} \\ \vdots \\ u_{k+r-1} \end{bmatrix}}_{\mathbf{u}_k}$$

$$+ \underbrace{\begin{bmatrix} I & \cdots & \cdots & 0 \\ CK & I & \cdots & 0 \\ CAK & CK & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ CA^{r-2}K & CA^{r-3}K & \cdots & I \end{bmatrix}}_{H_r^e} \underbrace{\begin{bmatrix} e_k \\ e_{k+1} \\ e_{k+2} \\ \vdots \\ e_{k+r-1} \end{bmatrix}}_{\mathbf{e}_k}$$

- **Repeat** for $k \in \{r, \ldots, M = N - r + 1\}$ and **concatenate horizontally**:

$$\underbrace{\begin{bmatrix} \mathbf{y}_r & \cdots & \mathbf{y}_M \end{bmatrix}}_{\mathbf{Y}} = \Gamma_r \underbrace{\begin{bmatrix} x_r & \cdots & x_M \end{bmatrix}}_{\mathbf{X}} + H_r^u \underbrace{\begin{bmatrix} \mathbf{u}_r & \cdots & \mathbf{u}_M \end{bmatrix}}_{\mathbf{U}} + H_r^e \underbrace{\begin{bmatrix} \mathbf{e}_r & \cdots & \mathbf{e}_M \end{bmatrix}}_{\mathbf{E}}$$

Basic SID algorithm: extended observability matrix

The basic relation

- The previous relation is written **compactly** as:

$$\mathbf{Y} = \Gamma_r \mathbf{x} + H_r^u \mathbf{U} + H_r^e \mathbf{E}$$

- Γ_r is called **extended observability matrix**
- H_r^u and H_r^e are **block lower triangular** matrices

Computing the extended observability matrix

- Many **different methods** exist. For instance express:

$$\mathbf{x} = \begin{bmatrix} x_r & \cdots & x_M \end{bmatrix} = A_K^r \begin{bmatrix} x_0 & \cdots & x_{M-r} \end{bmatrix} + \begin{bmatrix} A_K^{r-1}B & \cdots B \end{bmatrix} \mathbf{U}_p + \begin{bmatrix} A_K^{r-1}K & \cdots K \end{bmatrix} \mathbf{Y}_p$$

$$\underbrace{\approx \begin{bmatrix} A_K^{r-1}B & \cdots B & A_K^{r-1}K & \cdots K \end{bmatrix}}_{L_z} \underbrace{\begin{bmatrix} \mathbf{U}_p \\ \mathbf{Y}_p \end{bmatrix}}_{\mathbf{Z}_p}$$

- **Solve** the basic relation: $\mathbf{Y} = \Gamma_r L_z \mathbf{Z}_p + H_r^u \mathbf{U} + H_r^e \mathbf{E}$ to obtain $(\Gamma_r \mathbf{L}_z)$
- Compute Γ_r from a **truncated SVD** of $(\Gamma_r \mathbf{L}_z)$

Basic SID algorithm: obtaining (A, B, C)

Computing (A, C)

- From the extended observability matrix observe:

$$\Gamma_r = \begin{bmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^{r-1} \end{bmatrix} \Rightarrow \begin{bmatrix} CA \\ CA^2 \\ \vdots \\ CA^r \end{bmatrix} = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{r-1} \end{bmatrix} A$$

- Therefore (using Matlab notation):
  - Define C
  $$C = \Gamma_r(1 : p, :)$$
  - Compute A as **least squares** solution of the following overdetermined linear system:
  $$\Gamma_r(p + 1 : pr, :) = \Gamma_r(1 : p(r - 1), :)A$$

Computing B and the rest...

- Having computed (A, C), solving for B can be done again as a **least squares** problem
- Usually also x_0 and K can be computed by **LS** operations

Model validation: prediction error analysis

Output predictions

- Given an input **sequence**, the **model output sequence** is evaluated

$$x_{k+1} = Ax_k + Bu_k$$
$$y_k^{\text{model}} = Cx_k$$

- **Comparative plots** of y_k vs y_k^{model} are useful to assess the model quality

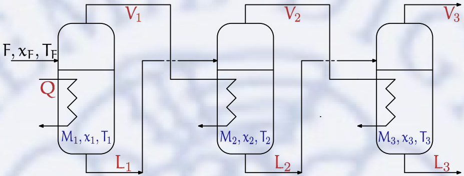
Prediction error analysis

- Compute the **prediction error** sequence:

$$\epsilon_k = y_k - y_k^{\text{model}}$$

- Analyze its **statistical** properties:
  - **Autocorrelation** function (ideally, it should be **white noise**)
  - **Input correlation**: if the prediction error is correlated with the input, then we have **modeling errors**

Multistage evaporator process: input/output relations



From prior knowledge or preliminary tests

- **Thoughtfully** identify which inputs affects each outputs
- Initial **independent step tests** can be used
- If an input/output relation is **very mild**, it is often better to **neglect** it

	L_1	Q_1	V_1	L_2	V_2	L_3	V_3
M_1	✓	-	✓	-	-	-	-
T_1	-	✓	✓	-	-	-	-
x_1	-	-	✓	-	-	-	-
M_2	✓	-	-	✓	✓	-	-
T_2	✓	✓	✓	-	✓	-	-
x_2	✓	-	✓	-	✓	-	-
M_3	-	-	-	✓	-	✓	✓
T_3	✓	✓	✓	✓	✓	-	✓
x_3	✓	-	✓	✓	✓	-	✓

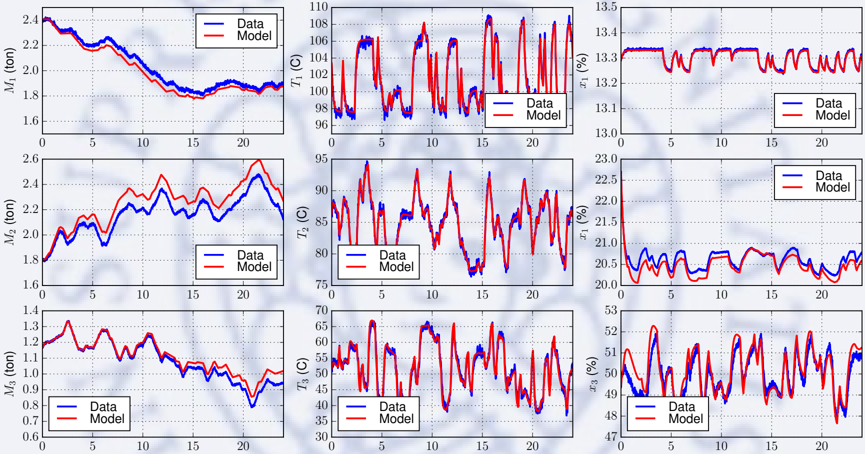
Multistage evaporator process: identified model

Identification algorithm and parameters

- MISO approach
- N4SID algorithm from **System Identification Toolbox** of Matlab
- Automatic order selection, based on singular value threshold

	L_1	Q_1	V_1	L_2	V_2	L_3	V_3
M_1	$-\frac{0.04797}{z-2.717}$	-	$-\frac{0.0339}{z-2.717}$	-	-	-	-
T_1	-	$\frac{0.564}{z-2.509}$	$-\frac{0.1745}{z-2.509}$	-	-	-	-
x_1	-	-	$\frac{0.009394}{z-2.549}$	-	-	-	-
M_2	$\frac{0.05726}{z-2.716}$	-	-	$-\frac{0.07207}{z-2.716}$	$-\frac{0.09465}{z-2.716}$	-	-
T_2	$\frac{0.008029z-0.01856}{z^2-4.913z+6.029}$	$\frac{0.089z-0.02579}{z^2-4.913z+6.029}$	$\frac{0.2431z-0.6396}{z^2-4.913z+6.029}$	-	$\frac{-0.6057z+1.451}{z^2-4.913z+6.029}$	-	-
x_2	$-\frac{0.01418}{z-2.604}$	-	$\frac{0.01038}{z-2.604}$	-	$\frac{0.02976}{z-2.604}$	-	-
M_3	-	-	-	$\frac{0.07503}{z-2.712}$	-	$\frac{0.08504}{z-2.712}$	$-\frac{0.1255}{z-2.712}$
T_3	$\frac{0.001138z+0.03875}{z^2-4.898z+5.986}$	$\frac{-0.02526z+0.3423}{z^2-4.898z+5.986}$	$\frac{0.06671z-0.127}{z^2-4.898z+5.986}$	$\frac{0.09903z-0.2557}{z^2-4.898z+5.986}$	$\frac{2.472z-6.521}{z^2-4.898z+5.986}$	-	$\frac{-2.895z+7.385}{z^2-4.898z+5.986}$
x_3	$\frac{-0.01013z+0.01865}{z^2-5.241z+6.864}$	-	$\frac{0.004064z-0.005355}{z^2-5.241z+6.864}$	$\frac{-0.2224z+0.6029}{z^2-5.241z+6.864}$	$\frac{0.01244z-0.02893}{z^2-5.241z+6.864}$	-	$\frac{0.464z-1.249}{z^2-5.241z+6.864}$

Multistage evaporator process: model validation



Conclusions

- Systems identification is of **paramount importance** for the success of advanced control, especially **Model Predictive Control**
- Systems identification has **grown significantly** since its origins
- **Input design** is at least as important as identification algorithms
- **Multivariable identification** techniques (for input design and identification) are becoming ubiquitous in **process control**