

# Model Identification and Data Analysis

Matteo Secco

January 2, 2022

# Contents

<b>I</b>	<b>MIDA 1</b>	<b>4</b>
<b>1</b>	<b>Prediction</b>	<b>5</b>
1.1	Probability Recall . . . . .	5
1.1.1	Random Vectors . . . . .	5
1.1.2	Random processes . . . . .	5
1.1.3	Important process classes . . . . .	6
1.2	Spectral Analysis . . . . .	6
1.2.1	Foundamentals . . . . .	6
1.2.2	Fundamental theorem of Spectral Analysis . . . . .	7
1.2.3	Canonical representation of a Stationary Process . . . . .	7
1.3	Moving Average Processes . . . . .	7
1.3.1	MA(1): . . . . .	7
1.3.2	MA(n) . . . . .	8
1.3.3	MA( $\infty$ ) . . . . .	9
1.3.4	Well definition of an MA( $\infty$ ) . . . . .	9
1.4	Auto Regressive Processes . . . . .	10
1.4.1	AR(1) . . . . .	10
1.4.2	AR(n) . . . . .	11
1.5	ARMA Processes . . . . .	12
1.6	Prediction problem . . . . .	12
1.6.1	Fake problem . . . . .	12
1.6.2	True Problem . . . . .	13
1.6.3	Prediction with eXogenous variables . . . . .	13
<b>2</b>	<b>Identification</b>	<b>15</b>
2.1	Prediction Error Minimization . . . . .	15
<b>II</b>	<b>MIDA 2</b>	<b>16</b>
<b>3</b>	<b>Black-Box non-parametric I/O systems</b>	<b>17</b>
3.1	State-space models . . . . .	17
3.1.1	State-space representation . . . . .	17
3.1.2	Transfer function representation . . . . .	18

3.1.3	Convolution of the input with the impulse response . . . .	18
3.2	Converting representations one to another . . . . .	18
3.2.1	State space to Transfer function . . . . .	18
3.2.2	Transfer Function to State Space . . . . .	18
3.2.3	Transfer function to Impulse response . . . . .	19
3.2.4	Impulse response to Transfer function . . . . .	19
3.2.5	State space to Impulse response . . . . .	19
3.3	Controllability and Observability . . . . .	20
3.4	Hankel Matrix . . . . .	20
3.5	Subspace-based State Space System Identification . . . . .	21
3.5.1	Obtain $F, G, H$ from a noise-free IR . . . . .	21
3.6	Obtain $F, G, H$ from a noisy IR . . . . .	22
<b>4</b>	<b>Parametric black-box system identification using frequency-domain approach</b>	<b>23</b>
4.1	Experiment design and data pre-processing . . . . .	23
4.2	Model class selection . . . . .	25
4.3	Performance index . . . . .	25
<b>5</b>	<b>Kalman filter</b>	<b>26</b>
5.1	Motivations and Goals . . . . .	26
5.1.1	Kalman on Basic Systems . . . . .	27
5.1.2	Exogenous input . . . . .	28
5.1.3	Multi-step prediction . . . . .	28
5.1.4	Filter ( $\hat{x}(t t)$ ) . . . . .	28
5.1.5	Time-varying systems . . . . .	29
5.1.6	Non linear system . . . . .	29
5.2	Asymptotic solution of K.F. . . . .	29
5.2.1	Basic idea . . . . .	29
5.2.2	Existence of $\bar{K}$ . . . . .	29
5.3	Extension to non-linear systems . . . . .	30
5.4	Optimization of gain $K$ . . . . .	31
5.4.1	Direct solution . . . . .	31
5.4.2	KF theory solution . . . . .	31
<b>6</b>	<b>Software-sensing with Black box Methods</b>	<b>33</b>
6.1	Linear Time Invariant Systems . . . . .	34
6.2	Non-linear systems . . . . .	35
6.2.1	Recurrent neural network . . . . .	35
6.2.2	FIR architecture . . . . .	35
6.2.3	IRR scheme . . . . .	38
6.2.4	Physical regressors . . . . .	39

<b>7</b>	<b>Grey-box System Identification</b>	<b>40</b>
7.1	Using Kalman Filter . . . . .	40
7.1.1	Problem definition . . . . .	40
7.1.2	State extension . . . . .	40
7.1.3	Design choice . . . . .	41
7.1.4	Applicability . . . . .	41
7.2	Simulation Error Method . . . . .	41
7.2.1	Dataset . . . . .	41
7.2.2	Model . . . . .	41
7.2.3	Performance index . . . . .	42
7.2.4	Optimization . . . . .	42
7.2.5	Limitations . . . . .	42
<b>8</b>	<b>Minimum Variance Control</b>	<b>43</b>
8.1	Setup the problem . . . . .	43
8.1.1	Simplified problem 1 . . . . .	44
8.1.2	Simplified problem 2 . . . . .	45
8.1.3	General solution . . . . .	46
<b>9</b>	<b>Recursive Identification</b>	<b>47</b>
9.1	Least square . . . . .	47
9.1.1	Recursive Least Square . . . . .	48
9.1.2	First form . . . . .	48
9.1.3	Second form . . . . .	48
9.1.4	Third form . . . . .	49
9.1.5	Forgetting factor . . . . .	49
<b>III</b>	<b>Cheatsheet</b>	<b>50</b>
<b>A</b>	<b>MIDA 1</b>	<b>51</b>
A.1	Probability Recall . . . . .	51
A.2	Spectral analysis . . . . .	52
A.3	Moving Average MA(n) . . . . .	52
A.3.1	MA( $\infty$ ) . . . . .	52
A.4	Auto Regressive AR(n) . . . . .	53
A.5	Known predictors . . . . .	53
<b>B</b>	<b>MIDA 2</b>	<b>54</b>
B.1	State-space representation . . . . .	54
B.2	BB representations shift . . . . .	54
B.2.1	State space to Transfer function . . . . .	54
B.2.2	Transfer Function to State Space . . . . .	54
B.2.3	Transfer function to Impulse response . . . . .	55
B.2.4	Impulse response to Transfer function . . . . .	55
B.2.5	State space to Impulse response . . . . .	55

B.3	Controllability and Observability . . . . .	55
B.3.1	Fully observable system . . . . .	55
B.3.2	Fully controllable system . . . . .	55
B.3.3	Hankel Matrix . . . . .	56
<b>C</b>	<b>System Identification</b>	<b>57</b>
C.1	$F, G, H$ from noise-free IR . . . . .	57
C.2	Obtain $F, G, H$ from a noisy IR (TODO: not well understood) . . . . .	58
<b>D</b>	<b>Kalman Filter</b>	<b>59</b>
D.1	Representations . . . . .	59
D.1.1	For basic systems . . . . .	59
D.1.2	With input . . . . .	59
D.1.3	Multi-step prediction . . . . .	59
D.1.4	Filter ( $\hat{x}(t t)$ ) . . . . .	60
D.1.5	Time-varying systems . . . . .	60
D.2	Asymptotic KF . . . . .	60
D.2.1	ARE . . . . .	60
D.3	Non-linear systems . . . . .	61
D.4	Optimizing $K$ . . . . .	61
<b>E</b>	<b>Minimum Variance Control</b>	<b>62</b>

**Part I**

**MIDA 1**

# Chapter 1

## Prediction

### 1.1 Probability Recall

#### 1.1.1 Random Vectors

**Variance**  $Var[v] = E[(v - E[v])^2]$

**Cross-Variance**  $Var[v, u] = E[(v - E[v])(u - E[u])]$

**Variance Matrix** 
$$\begin{vmatrix} Var[v_1] & \cdot & \cdot & Var[v_1, v_k] \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ Var[v_k, v_1] & \cdot & \cdot & Var[v_k] \end{vmatrix}$$

**Covariance coefficient**  $\delta[i, j] = \frac{Var[i, j]}{\sqrt{Var[i]}\sqrt{Var[j]}}$

$\delta[i, j] = 0 \implies i, j$  uncorrelated

$|\delta[i, j]| = 1 \implies i = \alpha j$

#### 1.1.2 Random processes

$v(t, s)$  |  $t$  time instant,  $s$  experiment outcome (generally given)

**Mean**  $m(t) = E[v(t, s)]$

**Variance**  $\lambda^2(t) = Var[v(t)]$

**Covariance function**  $\gamma(t_1, t_2) = E[(v(t_1) - m(t_1))(v(t_2) - m(t_2))] = \gamma(t_2, t_1)$

**Normalized Covariance Function**  $\rho(\tau) = \frac{\gamma(\tau)}{\gamma(0)}$

$\forall$  stationary processes:  $|\rho(\tau)| \leq 1 \quad \forall \tau$

### 1.1.3 Important process classes

#### Stationary process

- $m(t) = m$  constant
- $\lambda^2(t) = \lambda^2$  constant
- $\gamma(t_1, t_2) = f(t_2 - t_1) = \gamma(\tau)$  covariance depends only on time difference  $\tau$

$$|\gamma(\tau)| \leq \gamma(0) \quad \forall \tau$$

**White noise**  $\eta(t) \sim WN(m, \lambda^2)$

- Stationary process
- $\gamma(\tau) = 0 \quad \forall \tau \neq 0$

$$v(t) = \alpha\eta(t) + \beta \quad \eta(t) \sim WN(0, \lambda^2) \quad \implies \quad v(t) \sim WN(\beta, \alpha^2\lambda^2)$$

## 1.2 Spectral Analysis

### 1.2.1 Fundamentals

#### Spectrum

$$\Gamma(\omega) = \overbrace{F(\gamma(\tau))}^{\text{Fourier transform}} = \sum_{\tau=-\infty}^{+\infty} \gamma(\tau) \cdot e^{-j\omega\tau}$$

**Euler formula**  $\Gamma(\omega) = \gamma(0) + 2\cos(\omega)\gamma(1) + 2\cos(2\omega)\gamma(2) + \dots$

#### Spectrum properties

- $\Gamma : \mathbb{R} \rightarrow \mathbb{R}$
- $\Gamma$  is periodic with  $T = 2\pi$
- $\Gamma$  is even [ $\Gamma(-\omega) = \Gamma(\omega)$ ]
- $\Gamma(\omega) \geq 0 \quad \forall \omega$

$$\eta(t) \sim WN(0, \lambda^2) \quad \implies \quad \Gamma_{\eta}(\omega) = \gamma(0) = \text{Var}[\eta(t)] = \lambda^2$$

#### Anti-Transform

$$\gamma(\tau) = \frac{1}{2\pi} \int_{-\pi}^{+\pi} \Gamma(\omega) e^{j\omega\tau} d\omega$$



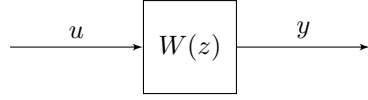
### Complex spectrum

$$\phi(z) = \sum_{\tau=-\infty}^{+\infty} \omega(\tau) z^{-\tau}$$

$$\Gamma(\omega) = \Phi(e^{j\omega})$$

### 1.2.2 Fundamental theorem of Spectral Analysis

**Fundamental theorem of Spectral Analysis** allows to derive the (real and/or complex) spectrum of the output from the input and the transfer function of the system



$$\Gamma_{yy}(\omega) = |W(e^{j\omega})|^2 \cdot \Gamma_{uu}(\omega)$$

$$\Phi_{yy}(z) = W(z)W(z^{-1}) \cdot \Phi_{uu}(z)$$

### 1.2.3 Canonical representation of a Stationary Process

A stationary process can be represented by an infinite number of transfer functions. The canonical representation is the transfer function  $W(z)$  such that:

- Numerator and denominator have same degree
- Numerator and denominator are monic (highest grade coefficient is 1)
- Numerator and denominator are coprime ( $W(z)$  cannot be simplified)
- numerator and denominator are stable polynomials (all poles and zeros of  $W(z)$  are inside the unit disk)

## 1.3 Moving Average Processes

Given  $\eta(t) \sim WN(0, \lambda^2)$

### 1.3.1 MA(1):

**Model**

$$v(t) = c_0 \eta(t) + c_1 \eta(t-1)$$

### Mean

$$\begin{aligned} E[v(t)] &= c_0 \cdot E[\eta(t)] + c_1 \cdot E[\eta(t)] \\ &= c_0 \cdot 0 + c_1 \cdot 0 \end{aligned}$$

$$\boxed{E[v(t)] = 0}$$

### Variance

$$\begin{aligned} Var[v(t)] &= E[(v(t) - \underbrace{E[v(t)]}_0)^2] \\ &= E[(v(t))^2] \\ &= E[(c_0 \cdot \eta(t)^2 + c_1 \cdot \eta(t-1))^2] \\ &= c_0^2 \cdot E[\eta(t)^2] + c_1^2 \cdot E[\eta(t-1)^2] + \underbrace{2c_0c_1 \cdot E[\eta(t)\eta(t-1)]}_0 \\ &= c_0^2 \cdot E[\eta(t)^2] + c_1^2 \cdot E[\eta(t-1)^2] \\ &= c_0^2 \lambda^2 + c_1^2 \lambda^2 \end{aligned}$$

$$\boxed{Var[v(t)] = (c_0^2 + c_1^2) \lambda^2}$$

### Covariance

$$\begin{aligned} \gamma(t_1, t_2) &= E[(v(t_1) - E[v(t_1)]) \cdot (v(t_2) - E[v(t_2)])] \\ &= E[(c_0\eta(t_1) + c_1\eta(t_1-1)) \cdot (c_0\eta(t_2) + c_1\eta(t_2-1))] \\ &= c_0^2 E[\eta(t_1)\eta(t_2)] + c_1^2 E[\eta(t_1-1)\eta(t_2-1)] \\ &\quad + c_0c_1 E[\eta(t_1)\eta(t_2-1)] + c_0c_1 E[\eta(t_1-1)\eta(t_2)] \end{aligned}$$

$$\boxed{\gamma(\tau) = \begin{cases} c_0^2 \lambda^2 + c_1^2 \lambda^2 & \text{if } \tau = 0 \\ c_0c_1 \lambda^2 & \text{if } \tau = \pm 1 \\ 0 & \text{otherwise} \end{cases}}$$

## 1.3.2 MA(n)

### Model

$$\begin{aligned} v(t) &= c_0\eta(t) + c_1\eta(t-1) + \dots + c_n\eta(t-n) \\ &= (c_0 + c_1z^{-1} + \dots + c_nz^{-n})\eta(t) \end{aligned}$$

### Transfer function

$$W(z) = c_0 + c_1z^{-1} + \dots + c_nz^{-n} = \frac{c_0z^n + c_1z^{n-1} + \dots + c_n}{z^n}$$

All poles are in the complex origin

### Mean

$$E[v(t)] = (c_0 + c_1 + \dots + c_n) \underbrace{E[\eta(t)]}_0$$

$$\boxed{E[v(t)] = 0}$$

### Covariance function

$$\gamma(\tau) = \begin{cases} \lambda^2 \cdot \sum_{i=0}^{n-\tau} c_i c_{i-\tau} & |\tau| \leq n \\ 0 & \text{otherwise} \end{cases}$$

### example

$$\gamma(0) = (c_0^2 + c_1^2 + \dots + c_n^2) \lambda^2$$

$$\gamma(1) = (c_0 c_1 + c_1 c_2 + \dots + c_{n-1} c_n) \lambda^2$$

$$\gamma(2) = (c_0 c_2 + c_1 c_3 + \dots + c_{n-2} c_n) \lambda^2$$

...

$$\gamma(n) = (c_0 c_n) \lambda^2$$

$$\gamma(k) = 0 \quad \forall k > n$$

### 1.3.3 MA( $\infty$ )

#### Model

$$v(t) = c_0 \eta(t) + c_1 \eta(t-1) + \dots + c_k \eta(t-k) + \dots = \sum_{i=0}^{\infty} c_i \eta(t-i)$$

#### Variance

$$\gamma(0) = (c_0^2 + c_1^2 + \dots + c_k^2 + \dots) \lambda^2 = \lambda^2 \sum_{i=0}^{\infty} c_i^2$$

### 1.3.4 Well definition of an MA( $\infty$ )

We need to have  $|\gamma(\tau)| \leq \gamma(0)$ , so we must require that

$$\boxed{\gamma(0) = \lambda^2 \sum_{i=0}^{\infty} c_i^2 \text{ is finite}}$$

## 1.4 Auto Regressive Processes

### 1.4.1 AR(1)

**Model**

$$v(t) = av(t-1) + \eta(t)$$

**Mean**

$$\begin{aligned} E[v(t)] &= E[av(t-1)] + \overbrace{E[\eta(t)]}^0 \\ &= aE[v(t-1)] \\ &= aE[v(t)] \\ (1-a)E[v(t)] &= 0 \\ \boxed{E[v(t)]} &= 0 \end{aligned}$$

**Covariance**

**MA( $\infty$ ) method** Observe as an AR(1) can be expressed as an MA( $\infty$ )

$$\begin{aligned} v(t) &= av(t-1) && +\eta(t) \\ &= a[av(t-2) + \eta(t-1)] && +\eta(t) \\ &= a^2v(t-2) && +a\eta(t-1) + \eta(t) \\ &= a^2[v(t-3) + \eta(t-2)] && +a\eta(t-1) + \eta(t) \\ &= \underbrace{a^nv(t-n)}_{\rightarrow 0} + \underbrace{\sum_{i=0}^{\infty} a^i \eta(t-i)}_{MA(\infty)} \end{aligned}$$

In particular, the result depends on an  $MA(\infty)$  having  $\sum_{i=0}^{\infty} c_i = \sum_{i=0}^{\infty} a^i$ . To check if the variance is finite we check  $\gamma(0) = \lambda^2 \sum_{i=0}^{\infty} a^{2i} < \infty$ . The given is a geometric series, convergent for  $|a| < 1$ . Under this hypothesis its value is

$$\gamma(0) = \lambda^2 \sum_{i=0}^{\infty} a^{2i} = \frac{\lambda^2}{1-a^2}$$

Applying the formula of the variance of MA processes we get

$$\begin{aligned}\gamma(1) &= (c_0c_1 + c_1c_2 + \dots)\lambda^2 = (a + aa^2 + \dots)\lambda^2 = a(1 + a^2 + a^4 + \dots)\lambda^2 = a\lambda^2 \sum_{i=0}^{\infty} a^{2i} = a \frac{\lambda^2}{1 - a^2} = a\gamma(0) \\ \gamma(2) &= (c_0c_2 + c_1c_3 + \dots)\lambda^2 = (a^2 + aa^3 + \dots)\lambda^2 = a^2(1 + a^2 + a^4 + \dots)\lambda^2 = a^2\lambda^2 \sum_{i=0}^{\infty} a^{2i} = a^2 \frac{\lambda^2}{1 - a^2} = a^2\gamma(0)\end{aligned}$$

$$\boxed{\gamma(\tau) = a^{|\tau|} \frac{\lambda^2}{1 - a^2}}$$

### Yule-Walkler Equations

$$\begin{aligned}Var[v(t)] &= E[v(t)^2] \\ &= E[(av(t) + \eta(t))^2] \\ &= a^2 \underbrace{E[v(t-1)^2]}_{\substack{=Var[v(t-1)] \\ =Var[v(t)] \\ =\gamma(0)}} + \underbrace{E[\eta(t)^2]}_{=\lambda^2} + 2a \underbrace{E[v(t-1)\eta(t)]}_{\substack{v(t-1) \text{ depends on } \eta(t-2) \\ \eta(t) \text{ independent of } \eta(t-2) \\ \xrightarrow{E[v(t-1)\eta(t)] = 0}}}\end{aligned}$$

$$\gamma(0) = a^2\gamma(0) + \lambda^2$$

$$\boxed{\gamma(0) = \frac{\lambda^2}{1 - a^2}}$$

To find  $\gamma(\tau)$ , we start from the model  $v(t) = av(t-1) + \eta(t)$ .

$$\begin{aligned}v(t) &= av(t-1) + \eta(t) \\ v(t)v(t-\tau) &= av(t-1)v(t-\tau) + \eta(t)v(t-\tau) \\ \underbrace{E[v(t)v(t-\tau)]}_{\gamma(\tau)} &= a \underbrace{E[v(t-1)v(t-\tau)]}_{\gamma(\tau-1)} + \underbrace{E[\eta(t)v(t-\tau)]}_0 \\ \boxed{\gamma(\tau) &= a\gamma(\tau-1)}\end{aligned}$$

We can join the two by inductive reasoning, obtaining

$$\boxed{\gamma(\tau) = a^{|\tau|} \frac{\lambda^2}{1 - a^2}}$$

**Long Division** Leads to same result, but is boring

### 1.4.2 AR(n)

**Model**

$$v(t) = a_1v(t-1) + a_2v(t-2) + \dots + a_nv(t-n) + \eta(t)$$

### Transfer function

$$W(z) = \frac{z^n}{z^n - a_1 z_{n-1} - \dots - a_n}$$

### Mean

$$E[v(t)] = a_1 E[v(t-1)] + a_2 E[v(t-2)] + \dots + a_n E[v(t-n)] + \underbrace{E[\eta(t)]}_0$$

$$m = a_1 m + a_2 m + \dots + a_n m$$

$$(1 - a_1 - a_2 - \dots - a_n)m = 0$$

$E[v(t)] = 0$

## 1.5 ARMA Processes

### Model

$$v(t) = a_1 v(t-1) + \dots + a_{n_a} v(t-n_a) + c_0 \eta(t) + \dots + c_{n_c} v(t-n_c)$$

Can also be expressed as  $V(t) = \frac{C(z)}{A(z)} \eta(t)$ , where

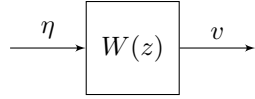
$$C(z) = c_0 + c_1 z^{-1} + \dots + c_{n_c} z^{-n_c}$$

$$A(z) = 1 - a_1 z^{-1} - \dots - a_{n_a} z^{-n_a}$$

Such process is stationary if all the poles of  $W(z)$  are inside the unit disk.

## 1.6 Prediction problem

We want to predict  $v(t+r)$  from  $v(t), v(t-1), \dots$ , where  $r$  is called prediction horizon, of the following stationary process:



### 1.6.1 Fake problem

Having a process with transfer function  $W(z)$ , we can compute it in polynomial form using the long division algorithm

$$W(z) = w_0 + w_1 z^{-1} + w_2 z^{-2} + \dots$$

We can calculate

$$v(t+r) = W(z)\eta(t+r) = \underbrace{w_0 \eta(t+r) + w_1 \eta(t+r-1) + \dots + w_{r-1} \eta(t+1)}_{\alpha(t) \text{ unpredictable: future of } \eta \text{ involved}} + \underbrace{w_r \eta(t) + w_{r+1} \eta(t-1) + \dots}_{\beta(t) \text{ predictable}}$$

The optimal fake predictor is then

$$\boxed{v(t+r|t) = w_r\eta(t) + w_{r+1}\eta(t-1) + \dots} = \beta(t)$$

And the prediction error is

$$\begin{aligned}\epsilon(t) &= v(t+r) & -\hat{v}(t+r|t) \\ &= \alpha(t) + \beta(t) & -\beta(t) \\ &= \alpha(t)\end{aligned}$$

$$\boxed{\epsilon(t) = w_0\eta(t+r) + w_1\eta(t+r-1) + \dots + w_{r-1}\eta(t+1)}$$

$$\boxed{Var[\epsilon(t)] = (w_0^2 + w_1^2 + \dots + w_{r-1}^2)\lambda^2}$$

### 1.6.2 True Problem

We want to estimate  $v(t+r)$  from  $v(t)$ , having transfer function  $W(z)$  and  $\hat{W}_r(z)$  the solution to the fake problem. We can calculate the transfer function of the real predictor from the process as

$$\boxed{W_r(z) = W(z)^{-1} \cdot \hat{W}_r(z)}$$

For ARMA processes a shortcut exists:

$$\hat{v}_{\text{ARMA}}(t|t-1) = \frac{C(z)A(z)}{C(z)} \quad \text{having } W(z) = \frac{C(z)}{A(z)}$$

### 1.6.3 Prediction with eXogenous variables

An exogenous variable is a deterministic input variable in the system

#### ARX model

$$v(t) = a_1v(t-1) + \dots + a_{n_a}v(t-n_a) + b_1u(t-1) + \dots + b_{n_b}u(t-n_b) + \eta(t)A(z)v(t) = B(z)u(t-1) +$$

#### Transfer functions from u and $\eta$

$$W_u(z) = \frac{B(z)}{A(z)} \quad W_\eta(z) = \frac{1}{A(z)}$$

#### ARMAX model

$$\begin{aligned}A(z)v(t) &= C(z)\eta(t) + B(z)u(t-1) \\ y(t) &= W(z)\eta(t) + G(z)u(t)\end{aligned}$$

**Predictor**

$$\hat{y}(t|t-1) = \frac{C(z) - A(z)}{C(z)}y(t) + \frac{B(z)}{C(z)}u(t-1)$$



# Chapter 2

## Identification

Consists of estimating a model from data.

### 2.1 Prediction Error Minimization

Aims to minimize  $\epsilon(t) = v(t) - \hat{v}(t|t-r)$

Steps:

1. **Data collection:** collect  $\vec{u}$  and  $\vec{y}$
2. **Family selection:** choose a family of models  $M(\theta)$ 
  - MA(1)  $\theta = [a]$
  - MA(n)  $\theta = [a_1, \dots, a_n]$
  - ARMA( $n_a, n_c$ )  $\theta = [a_1, \dots, a_{n_a}, c_1, \dots, c_{n_c}]$
  - ...
3. **Select an optimization criterion**
  - Mean Squared error  $J(\theta) = \frac{1}{N} \sum_{t=1}^N \epsilon_\theta(t)^2$
  - Mean absolute error  $J(\theta) = \frac{1}{N} \sum_{t=1}^N |\epsilon_\theta(t)|$
  - ...
4. **Optimization** find  $\hat{\theta} = \operatorname{argmin} J(\theta) \implies \frac{dJ(\theta)}{d\theta} = 0$
5. **Validation** verify if the result satisfies the requirements

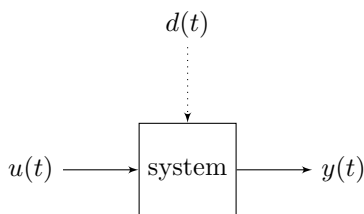
**Part II**

**MIDA 2**

## Chapter 3

# Black-Box non-parametric I/O systems

### 3.1 State-space models



**Known (measured) data**

$\{u(1), \dots, u(N)\}$	input
$\{y(1), \dots, y(N)\}$	output

#### 3.1.1 State-space representation

$$\begin{cases} x(t+1) = Fx(t) + Gu(t) & \text{state equations} \\ y(t) = Hx(t) + Du(t) & \text{output equations} \end{cases}$$

Where  $F_{n \times n}$ ,  $G_{n \times 1}$ ,  $H_{1 \times n}$  and  $D_{1 \times 1}$  are matrices.

**S.S. representation is not unique** Given any invertible matrix  $T$ , let  $F_1 = TFT^{-1}$ ,  $G_1 = TG$ ,  $H_1 = HT^{-1}$ ,  $D_1 = D$ . Then the system  $\{F, G, H, D\}$  is equivalent to the system  $\{F_1, G_1, H_1, D_1\}$ .

### 3.1.2 Transfer function representation

$$W(z) = \frac{B(z)}{A(z)} z^{-k} = \frac{b_0 + b_1 z^{-1} + \dots + b_p z^{-p}}{a_0 + a_1 z^{-1} + \dots + a_n z^{-n}} z^{-k}$$

$W(z)$  is a rational function of the  $z$  operator  $\rightarrow$  is a digital filter

**Infinite impulse response**  $W(z) = \frac{z^{-1}}{1 + \frac{1}{3}z^{-1}}$

**Finite impulse response**  $W(z) = z^{-1} + \frac{1}{2}z^{-2} + \frac{1}{4}z^{-3}$

### 3.1.3 Convolution of the input with the impulse response

Let's call  $\omega(1), \omega(2), \dots$  the values of  $y(t)$  when  $u(t) = \text{impulse}(0)$ , and let's measure the values of  $y$  at different times: . Then it can be proven that for any  $u(t)$

$$y(t) = \sum_{k=0}^{\infty} \omega(k) u(t-k)$$

## 3.2 Converting representations one to another

### 3.2.1 State space to Transfer function

Consider a strictly proper system:

$$\begin{cases} x(t+1) = Fx(t) + Gu(t) \\ y(t+1) = Hx(t) + Du(t) \end{cases} \xrightarrow{0} \begin{cases} x(t+1) = Fx(t) + Gu(t) \\ y(t) = Hx(t) \end{cases}$$

Applying the  $z$  operator we get

$$\begin{aligned} zx(t) &= Fx(t) + Gu(t) \\ x(t)(zI - F) &= Gu(t) \\ x(t) &= (zI - F)^{-1}Gu(t) \\ y(t) &= H(zI - F)^{-1}Gu(t) \end{aligned}$$

And we can extract the transfer function:

$$W(z) = H(zI - F)^{-1}G$$

### 3.2.2 Transfer Function to State Space

We have the transfer function

$$W(z) = \frac{b_0 z^{n-1} + b_1 z^{n-2} + \dots + b_{n-1}}{z^n + a_0 z^{n-1} + \dots + a_n}$$

The formulas for the state space matrices is

$$F = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 \\ 0 & 0 & \dots & 0 & 1 \\ -a_n & -a_{n-1} & \dots & \dots & -a_1 \end{bmatrix} \quad G = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} \quad H = [b_{n-1} \quad b_{n-2} \quad \dots \quad b_0] \quad D = 0$$

### 3.2.3 Transfer function to Impulse response

Obtained by computing the  $\infty$  long division of  $W(z)$

### 3.2.4 Impulse response to Transfer function

**Z-transform** Given a discrete-time signal  $s(t)$  such that  $\forall t < 0 : s(t) = 0$ , it's Z-transform is

$$\mathcal{Z} = \sum_{t=0}^{\infty} s(t)z^{-t}$$

It can be proven that:

$$W(z) = \mathcal{Z}(\omega(t)) = \sum_{t=0}^{\infty} \omega(t)z^{-1}$$

NB: this works only in theory because of the infinite sum

### 3.2.5 State space to Impulse response

Consider the state space model:

$$\begin{cases} x(t+1) = Fx(t) + Gu(t) \\ y(t) = Hx(t) \end{cases}$$

We have that:

$$\begin{aligned} x(1) &= Fx(0) + Gu(0) & &= Gu(0) \\ y(1) &= Hx(1) & &= HGu(0) \end{aligned}$$

$$\begin{aligned} x(2) &= Fx(1) + Gu(1) & &= FGu(0) + Gu(1) \\ y(2) &= Hx(2) & &= HFGu(0) + HG(u1) \end{aligned}$$

$$\begin{aligned} x(3) &= Fx(2) + Gu(2) & &= F^2Gu(0) + FGu(1) + Gu(2) \\ y(3) &= Hx(3) & &= HF^2Gu(0) + HFGu(1) + HGu(2) \end{aligned}$$

$\vdots$

$$y(t) = 0u(t) + HGu(t-1) + HFGu(t-2) + HF^2Gu(t-3) + \dots$$

The IR is:

$$\omega(t) = \begin{cases} 0 & \text{if } t = 0 \\ HF^{t-1}G & \text{if } t > 0 \end{cases}$$

### 3.3 Controllability and Observability

$$\begin{cases} x(t+1) = Fx(t) + Gu(t) \\ y(t) = Hx(t) \end{cases}$$

**Fully observable system** The system is fully observable (from the output)  
 $\iff$  the observability matrix is full rank:

$$O = \begin{bmatrix} H \\ HF \\ \vdots \\ HF^{n-1} \end{bmatrix} \quad \text{rank}(O) = n$$

**Fully controllable system** The system is fully controllable (from the input)  
 $\iff$  the controllability (also called reachability) matrix is full rank:

$$R = [G \quad FG \quad \dots \quad F^{n-1}G] \quad \text{rank}(R) = n$$

### 3.4 Hankel Matrix

Starting from  $\omega(1), \omega(2), \dots, \omega(N)$  where  $N \geq 2n-1$ , we can build the Hankel Matrix of order  $n$ :

$$H_n = \begin{bmatrix} \omega(1) & \omega(2) & \omega(3) & \dots & \omega(n) \\ \omega(2) & \omega(3) & \omega(4) & \dots & \omega(n+1) \\ \omega(3) & \omega(4) & \omega(5) & \dots & \omega(n+2) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \omega(n) & \omega(n+1) & \omega(n+2) & \dots & \omega(2n-1) \end{bmatrix}$$

Knowing that

$$\omega(t) = \begin{cases} 0 & \text{if } t = 0 \\ HF^{t-1}G & \text{if } t > 0 \end{cases}$$

We can rewrite

$$H_n = \begin{bmatrix} HG & HFG & HF^2G & \dots & HF^{n-1}G \\ \vdots & \ddots & & & \vdots \\ \vdots & & \ddots & & \vdots \\ \vdots & & & \ddots & \vdots \\ HF^{n-1}G & \dots & \dots & \dots & HF^{2n-2}G \end{bmatrix} = \begin{bmatrix} H \\ HF \\ \vdots \\ HF^{n-1} \end{bmatrix} \cdot [G \quad FG \quad \dots \quad F^{n-1}G] = O \cdot R$$

### 3.5 Subspace-based State Space System Identification

**Impulse experiment** Measure  $y(t)$  under the input  $u(t) = \text{impulse}(0)(0)$   
How to derive  $F, G, H$  from  $\omega(0), \dots, \omega(n)$ ?

- Assuming the IR measurement to be noise free  $\rightarrow$  easier, not realistic
- Measure  $\omega(\hat{t})$  as a noisy signal and compute  $\omega(t) = \eta(t) - \hat{\omega}(t)$

#### 3.5.1 Obtain $F, G, H$ from a noise-free IR

1. Build the Hankel matrix of increasing order, and compute the rank until  $\text{rank}(H_n) = \text{rank}(H_{n+1})$ . Then,  $n$  is the order of the IR

$$H_1 = [\omega(1)] \quad H_2 = \begin{bmatrix} \omega(1) & \omega(2) \\ \omega(2) & \omega(3) \end{bmatrix} \quad H_3 = \dots \quad \dots \quad H_n = \dots$$

2. Take  $H_{n+1}$  and factorize it in two rectangular matrix of size  $(n+1) \times n$  and  $n \times (n+1)$ :  $H_{n+1} = O_{n+1} \cdot R_{n+1}$ , where

$$O_{n+1} = \begin{bmatrix} H \\ HF \\ \vdots \\ HF^n \end{bmatrix} \quad R_{n+1} = [G \quad FG \quad \dots \quad F^n G]$$

3. Estimate H,F,G:

- Extract F and G from the first element of O and R
- Define:

$$O_1 = \begin{bmatrix} H \\ HF \\ \vdots \\ HF^{n-1} \end{bmatrix} \quad O_2 = \begin{bmatrix} HF \\ \vdots \\ HF^n \end{bmatrix}$$

- Observe that  $O_1 F = O_2$ , so  $F = O_1^{-1} O_2$

### 3.6 Obtain $F, G, H$ from a noisy IR

The measurement is of  $\hat{\omega}(t) = \omega(t) + \eta(t)$ . To identify the process:

1. Build the Hankel matrix from data using all the  $N$  data available in one shot:

$$\hat{H}_{q \times d} = \begin{bmatrix} \hat{\omega}(1) & \hat{\omega}(2) & \dots & \hat{\omega}(d) \\ \hat{\omega}(2) & \hat{\omega}(3) & \dots & \hat{\omega}(d+1) \\ \vdots & \vdots & \ddots & \vdots \\ \hat{\omega}(q) & \hat{\omega}(q+1) & \dots & \hat{\omega}(q+d+1) \end{bmatrix}$$

Where  $q + d + 1 = N$

2. Calculate the Singular Value Decomposition of  $\hat{H}_{q \times d}$ :

$$\hat{H}_{q \times d} = \hat{U}_{q \times q} \cdot \hat{S}_{q \times d} \cdot \hat{V}_{d \times d}^T$$

$\hat{U}$  and  $\hat{V}$  are unitary matrices: they are invertible and their inverses are equal to their transpose.

$$\hat{S} = \begin{bmatrix} \sigma_1 & & & \\ & \sigma_2 & & \\ & & \ddots & \\ & & & \sigma_d \end{bmatrix}$$

3. Plot the singular values ( $\sigma_i$ ) and cut-off the three matrices:
  - Ideally, after a certain  $n$  (the order of the IR) there would be a jump dividing the signal (before) from the noise (after)
  - In reality no clear distinction exists, but it's possible to identify an interval of possible values of  $n$ . A tradeoff between complexity, precision and overfitting takes place
4. Split  $\hat{U}, \hat{S}, \hat{V}^T$  obtaining  $U_{q \times n}, S_{n \times n}, V_{n \times d}^T$  and then recreate  $H_{qd} = USV^T$
5.  $H$  and  $G$  are estimated as for the unnoisy case. To estimate  $F$  we can build  $O_1$  and  $O_2$  as before, but then the system  $O_1 \cdot F = O_2$  cannot be solved directly as  $O_1$  is not square. We can instead compute the approximate least-square solution of the system:

$$F = (O_1^T O_1)^{-1} O_1^T O_2$$



## Chapter 4

# Parametric black-box system identification using frequency-domain approach

### 4.1 Experiment design and data pre-processing

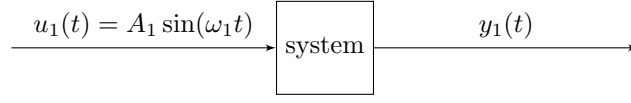
1. Select a set of excitation frequencies  $\{\omega_1, \dots, \omega_H\}$ . Usually  $\omega_i - \omega_{i-1}$  is constant  $\forall i \in \{2, \dots, H\}$ .  $\omega_H$  must be selected according to the bandwidth of the system
2. Make  $H$  independent experiments:
3. Focusing on experiment  $i$ , because of noise the real value of the output will be (unknowns are underlined)

$$\hat{y}_i = \underline{B}_i \sin(\omega_i t + \underline{\phi}_i) = \underline{a}_i \sin(\omega_i t) + \underline{b}_i \cos(\omega_i t)$$

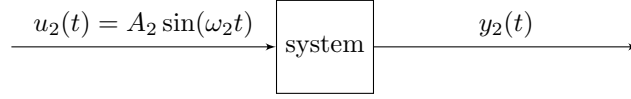
Using the second equation (since it is linear in the unknowns). We want to determine

$$\{\hat{a}_i, \hat{b}_i\} = \arg \min_{\{a_i, b_i\}} J_N(a_i, b_i)$$

$$J_N(a_i, b_i) = \frac{1}{N} \sum_{t=1}^N \left( \underbrace{y_i(t)}_{\text{measurement}} - \underbrace{a_i \sin(\omega_i t) - b_i \cos(\omega_i t)}_{\text{model output}} \right)^2$$

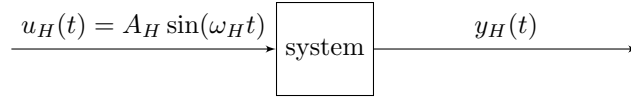


Experiment 1



Experiment 2

$\vdots$



Experiment H

This can be solved explicitly

$$\begin{aligned}\frac{\delta J_N}{\delta a_i} &= \frac{2}{N} \sum_{t=1}^N (-\sin(\omega_i t)) (y_i(t) - a_i \sin(\omega_i t) - b_i \cos(\omega_i t)) = 0 \\ \frac{\delta J_N}{\delta b_i} &= \frac{2}{N} \sum_{t=1}^N (-\cos(\omega_i t)) (y_i(t) - a_i \sin(\omega_i t) - b_i \cos(\omega_i t)) = 0\end{aligned}$$

Which results in the following linear system:

$$\begin{bmatrix} \sum_{t=1}^N \sin(\omega_i t)^2 & \sum_{t=1}^N \sin(\omega_i t) \cos(\omega_i t) \\ \sum_{t=1}^N \sin(\omega_i t) \cos(\omega_i t) & \sum_{t=1}^N \cos(\omega_i t)^2 \end{bmatrix} \begin{bmatrix} a_i \\ b_i \end{bmatrix} = \begin{bmatrix} \sum_{t=1}^N y_i(t) \sin(\omega_i t) \\ \sum_{t=1}^N y_i(t) \cos(\omega_i t) \end{bmatrix}$$

4. We want to move back to sin-only form:

$$\begin{aligned}\hat{\phi}_i &= \arctan\left(\frac{\hat{b}_i}{\hat{a}_i}\right) \\ \hat{B}_i &= \frac{\frac{\hat{a}_i}{\cos \hat{\phi}_i} + \frac{\hat{b}_i}{\sin \hat{\phi}_i}}{2}\end{aligned}$$

5. Repeating H experiments we obtain

$$\begin{aligned} \{\hat{B}_1, \hat{\phi}_1\} &\Rightarrow \frac{\hat{B}_1}{A_1} e^{j\hat{\phi}_1} \\ &\vdots \\ \{\hat{B}_H, \hat{\phi}_H\} &\Rightarrow \frac{\hat{B}_H}{A_H} e^{j\hat{\phi}_H} \end{aligned}$$

So we have H complex numbers representing the frequency response of  $W(z)$ . These numbers are our dataset

## 4.2 Model class selection

$$\mathcal{M}(\theta) : W(z, \theta) = \frac{b_0 + b_1 z^{-1} + \dots + b_p z^{-p}}{1 + a_1 z^{-1} + \dots + a_n z^{-n}} z^{-1} \quad \theta = \begin{bmatrix} a_1 \\ \vdots \\ a_n \\ b_0 \\ \vdots \\ b_p \end{bmatrix}$$

## 4.3 Performance index

$$\begin{aligned} J_H(\theta) &= \frac{1}{H} \sum_{i=1}^H \left( W(e^{j\omega_i}, \theta) - \frac{\hat{B}_i}{A_i} e^{j\hat{\phi}_i} \right)^2 \\ \hat{\theta} &= \arg \min_{\theta} J_H(\theta) \end{aligned}$$

## Chapter 5

# Kalman filter

Based on SS representation:

$$\begin{cases} x(t+1) = Fx(t) + Gu(t) + v_1(t) & v_1 \sim WN \\ y(t) = Hx(t) + Du(t) + v_2(t) & v_2 \sim WN \end{cases}$$

### 5.1 Motivations and Goals

Given a model and noise variances:

- find k-steps ahead predictors of the output  $y$
- find k-steps ahead predictors of the state  $x$
- Find the filter of the state  $\hat{x}(t|t)$  to allow software sensing
- Gray box system identification

Usually a dynamic system has  $m$  inputs,  $n$  states and  $p$  outputs

**Key problem** Usually  $p \ll n$ : physical sensors are much less than system states because:

- Cost
- Cables, power supply
- Maintenance

But we want full state measurements because:

- Control design (using state feedback)
- Monitoring (fault detection, predictive maintenance)

**Software sensing** determine the internal state using the values measured from input and output

### 5.1.1 Kalman on Basic Systems

$$S : \begin{cases} x(t+1) = Fx(t) + \cancel{Gu(t)} + v_1(t) & \text{state equation} \\ y(t) = Hx(t) + v_2(t) & \text{output equation} \end{cases}$$

$$x(t) = \begin{bmatrix} x_1(t) \\ \vdots \\ x_n(t) \end{bmatrix} \quad \cancel{u(t) = \begin{bmatrix} u_1(t) \\ \vdots \\ u_m(t) \end{bmatrix}} \quad y(t) = \begin{bmatrix} y_1(t) \\ \vdots \\ y_p(t) \end{bmatrix}$$

$v_1$  is a vector white noise:

$$v_1 \sim WN(0, V_1) \quad v_1(t) = \begin{bmatrix} v_{11}(t) \\ \vdots \\ v_{1n}(t) \end{bmatrix}$$

- $E[v_1(t)] = \vec{0}$
- $E[v_1(t) \cdot v_1(t)^T] = V_1$ , where  $V_1$  is an  $n \times n$  covariance matrix
- $E[v_1(t) \cdot v_1(t - \tau)^T] = 0 \quad \forall t, \forall \tau \neq 0$

$v_2$  is called output/measurement/sensor noise:

- $E[v_2(t)] = \vec{0}$
- $E[v_2(t) \cdot v_2(t)^T] = V_2$ , where  $V_2$  is an  $p \times p$  covariance matrix
- $E[v_2(t) \cdot v_2(t - \tau)^T] = 0 \quad \forall t, \forall \tau \neq 0$

$v_1$  and  $v_2$  are assumed to have the following relationships:

$$E[v_1(t) \cdot v_2(t - \tau)^T] = \underbrace{V_{12}}_{n \times p} = \begin{cases} 0 & \text{if } \tau \neq 0 \\ \text{any} & \text{if } \tau = 0 \end{cases}$$

So they can only be correlated in the same time instant. Since the system is dynamic we need to define its initial conditions:

$$E[x(1)] = \underbrace{X_0}_{n \times 1} \quad E[(x(1) - x(0))(x(1) - x(0))^T] = \underbrace{P_0}_{n \times n} \geq 0$$

$P_0 = 0 \iff$  the initial state is perfectly known.

We finally assume that  $v_1$  and  $v_2$  are uncorrelated with the initial state:

$$x(1) \perp v_1(t) \quad x(1) \perp v_2(t)$$

## Solution for Basic Systems

$$\begin{array}{ll}
 \hat{x}(t+1|t) = F\hat{x}(t|t-1) + K(t)e(t) & \text{state equation} \\
 \hat{y}(t|t-1) = H\hat{x}(t|t-1) & \text{output equation} \\
 e(t) = y(t) - \hat{y}(t|t-1) & \text{prediction error} \\
 K(t) = (FP(t)H^T + V_{12}) (HP(t)H^T + V_2)^{-1} & \text{gain of the KF} \\
 P(t+1) = (FP(t)F^T + V_1) \\
 \quad - (FP(t)H^T + V_{12}) (HP(t)H^T + V_2)^{-1} (FP(t)H^T + V_{12})^T & \text{difference Riccati equation} \\
 \hline
 \hat{x}(1|0) = E[x(1)] = X_0 & \text{Initial state} \\
 P(1) = \text{var}[x(1)] = P_0 & \text{initial DRE}
 \end{array}$$

### 5.1.2 Exogenous input

$$\begin{array}{ll}
 \hat{x}(t+1|t) = F\hat{x}(t|t-1) + Gu(t) + K(t)e(t) & \text{state equation} \\
 \text{other equations} = \text{unchanged} &
 \end{array}$$

### 5.1.3 Multi-step prediction

Knowing  $\hat{x}(t+1|T)$  from the basic solution we can derive

$$\begin{array}{l}
 \hat{x}(t+2|t) = F\hat{x}(t+1|t) \\
 \hat{x}(t+2|t) = F^2\hat{x}(t+1|t) \\
 \vdots \\
 \hat{x}(t+k|t) = F^{k-1}\hat{x}(t+1|t) \\
 \hline
 \hat{y}(t+k|t) = H\hat{x}(t+k|t)
 \end{array}$$

### 5.1.4 Filter ( $\hat{x}(t|t)$ )

**F invertible**

$$\hat{x}(t+1|t) = F\hat{x}(t|t) \implies \hat{x}(t|t) = F^{-1}\hat{x}(t+1|t)$$

**F not invertible** assuming  $V_{12} = 0$ , then we can re-formulate the K.F. solutions:

$$\begin{array}{l}
 \hat{x}(t|t) = F\hat{x}(t-1|t-1) + Gu(t-1) + K_0(t)e(t) \\
 \hat{y}(t|t-1) = H\hat{x}(t|t-1) \\
 e(t) = y(t) - \hat{y}(t|t-1) \\
 K_0(t) = (P(t)H^T) (HP(t)H^T + V_2)^{-1} \\
 P(t+1) = \text{unchanged}
 \end{array}$$

### 5.1.5 Time-varying systems

$$S: \begin{cases} x(t+1) = F(t)x(t) + G(t)u(t) + v_1(t) \\ y(t) = H(t)x(t) + v_2(t) \end{cases}$$

K.F. equations are unchanged

### 5.1.6 Non linear system

Much more complicated extension. Look for Extended Kalman Filter if interested (I'm not)

## 5.2 Asymptotic solution of K.F.

KF is time variant, because the gain  $K(t)$  is time varying. This causes 2 problems:

- It is difficult to check the stability of the system
- $K(t)$  must be computed at each sampling time, including the inversion of  $(HP(t)H^T)_{p \times p} \Rightarrow$  computationally intensive

Because of this, the asymptotic version of KF is preferred

### 5.2.1 Basic idea

If  $P(t)$  converges to constant  $\bar{P}$ , then also  $K(t)$  will converge to some constant  $\bar{K}$ . Using  $\bar{K}$  instead of  $K(t)$  the KF becomes time-invariant:

$$\begin{aligned} \hat{x}(t+1|t) &= F\hat{x}(t|t-1) + Gu(t) + \bar{K}e(t) \\ &= F\hat{x}(t|t-1) + Gu(t) + \bar{K}(y(t) - \hat{y}(t|t-1)) \\ &= F\hat{x}(t|t-1) + Gu(t) + \bar{K}(y(t) - H\hat{x}(t|t-1)) \\ &= \underbrace{(F - \bar{K}H)}_{\text{new state matrix}} \hat{x}(t|t-1) + Gu(t) + \bar{K}y(t) \end{aligned}$$

If  $\bar{K}$  exists, then the KF is asymptotically stable  $\iff$  all the eigenvalues of  $F - \bar{K}H$  are strictly inside the unit circle

### 5.2.2 Existence of $\bar{K}$

$$\bar{K} = (F\bar{P}H^T + V_{12}) + (H\bar{P}H^T + V_2)^{-1}$$

$\bar{K}$  exists if  $\bar{P}$  exists. DRE is an autonomous discrete time system  $x(t+1) = f(x(t))$ , in equilibrium when  $x(t+1) = x(t) \Rightarrow f(\bar{x}) = \bar{x}$ . Applied to  $P$  this leads to the following Algebraic Riccati Equation:

$$\bar{P} = f(\bar{P}) \iff \bar{P} = (F\bar{P}F^T + V_1) - (F\bar{P}H^T + V_{12}) (H\bar{P}H^T + V_2)^{-1} (F\bar{P}H^T + V_{12})^T$$

### First asymptotic theorem

Assuming  $V_{12} = 0$  and the system is asymptotically stable (all eigenvalues of  $F$  strictly inside the unit circle), then:

- $\exists!$  semi-definite positive solution of ARE:  $\bar{P} \geq 0$
- DRE converges to  $\bar{P} \quad \forall P_0 \geq 0$
- The corresponding  $\bar{K}$  will make the KF asymptotically stable

### Second asymptotic theorem

Assuming  $V_{12} = 0$ ,  $(F, H)$  is observable,  $(F, \Gamma)$  is controllable. Then:

- $\exists!$  semi-definite positive solution of ARE:  $\bar{P} \geq 0$
- DRE converges to  $\bar{P} \quad \forall P_0 \geq 0$
- The corresponding  $\bar{K}$  will make the KF asymptotically stable

## 5.3 Extension to non-linear systems

$$S : \begin{cases} x(t+1) = f(x(t), u(t)) + v_1(t) \\ y(t) = h(x(t)) + v_2(t) \end{cases}$$

Where  $f$  and  $h$  are non-linear functions.

For the gain block of the KF we have 2 types of solutions:

- The gain is a non linear function of  $e(t)$
- The gain is a linear time-varying function

The second solution is preferred, as it allows us to reuse the formulae with just little tweaks. In particular,  $F$  and  $H$  are computed as follows:

$$F(t) = \left. \frac{\delta f(x(t), u(t))}{\delta x(t)} \right|_{x(t)=\hat{x}(t|t-1)}$$
$$H(t) = \left. \frac{\delta h(x(t))}{\delta x(t)} \right|_{x(t)=\hat{x}(t|t-1)}$$

EKF is the time-varying solution of KF, where  $F$  and  $H$  are computed around the last available state prediction  $\hat{x}(t|t-1)$

### Algorithm

1. Take the last available state prediction  $\hat{x}(t|t-1)$
2. Use  $\hat{x}(t|t-1)$  to compute  $F(t)$  and  $H(t)$
3. Compute  $K(t)$  and update the DRE equations
4. Compute  $\hat{x}(t+1|t)$



## 5.4 Optimization of gain $K$

$$S: \begin{cases} x(t+1) = 2x(t) \\ y(t) = x(t) + v(t) \end{cases} \quad v \sim WN(0, 1)$$

### 5.4.1 Direct solution

Starting from the standard observer structure:

$$\begin{cases} \hat{x}(t+1|t) = 2\hat{x}(t|t-1) + K(y(t) - \hat{y}(t|t-1)) \\ \hat{y}(t|t-1) = \hat{x}(t|t-1) \end{cases}$$

Minimizing the variance of the prediction error  $var[\eta(t)] \Rightarrow$  minimizing  $var[x(t) - \hat{x}(t|t-1)]$

$$\begin{aligned} \eta(t) &= 2x(t) - [2\hat{x}(t|t-1) + K(y(t) - \hat{y}(t|t-1))] \\ &= 2x(t) - 2\hat{x}(t|t-1) - K(x(t) + v(t) - \hat{x}(t|t-1)) \\ &= (2-K)(x(t) - \hat{x}(t|t-1)) - Kv(t) \\ \eta(t+1) &= (2-K)\eta(t) - Kv(t) \end{aligned} \quad v \sim WN(0, 1)$$

This is an AR(1) process:

$$\eta(t) = \frac{1}{1 - (2-K)z^{-1}} e(t) \quad e(t) = -Kv(t) \quad e \sim WN(0, K^2)$$

The variance of  $\eta$  is

$$\gamma_\eta(0) = \frac{K^2}{1 - (2-K)^2}$$

Minimizing wrt  $K$ :

$$\frac{\delta \gamma_\eta(0)}{\delta K} = 0 \quad \Rightarrow \quad \begin{cases} K_1 = 0 \\ K_2 = \frac{3}{2} \end{cases}$$

### 5.4.2 KF theory solution

From  $S$  we can derive:

$$\begin{cases} F = 2 \\ H = 1 \\ V_1 = 0 \end{cases} \quad \Rightarrow \quad \begin{cases} \Gamma = 0 \\ V_2 = 1 \\ V_{12} = 0 \end{cases}$$

- $F$  is not asymptotically stable  $\Rightarrow$  cannot use theorem 1
- $(F, \Gamma)$  is not fully reachable  $\Rightarrow$  cannot use theorem 2

$$DRE = 4P(t) - \frac{(2P(t))^2}{P(t) + 1} \dots$$

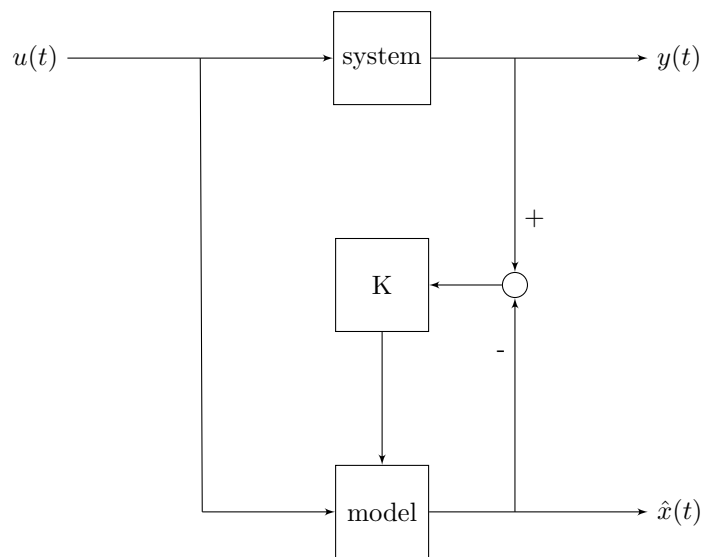
$$P(t+1) = \frac{4P(t)}{P(t) + 1}$$

Solving ARE:

$$\bar{P} = \frac{4\bar{P}}{\bar{P} + 1} \quad \Rightarrow \quad \begin{cases} \bar{P}_1 = 1 \\ \bar{P}_2 = 3 \end{cases} \quad \Rightarrow \quad \begin{cases} K_1 = 0 \\ K_2 = \frac{3}{2} \end{cases}$$

## Chapter 6

# Software-sensing with Blabk box Methods



### Features

- A white-box model is required
- No need of a training dataset
- Works by feedback estimation
- Constructive method
- Can be used to estimate unmeasurable states

## 6.1 Linear Time Invariant Systems

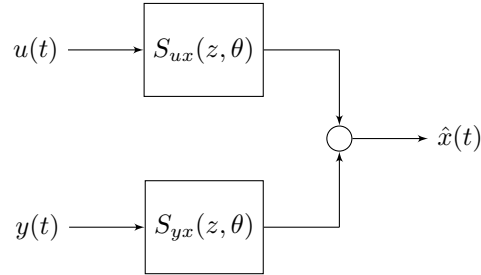
**Known white-box model of the system** Draw the block diagram of the system and the KF, then (from the diagram) calculate  $\hat{x}(t) = f(u(t), y(t))$ . Done.

**Unknown model for the system** A BB estimation is possible iff all the states are measurable.

### Dataset

$$\begin{aligned} &\{u(1), \dots, u(N)\} \\ &\{y(1), \dots, y(N)\} \\ &\{x(1), \dots, x(N)\} \end{aligned}$$

**Model** to be optimized for  $\theta$



### Performance index

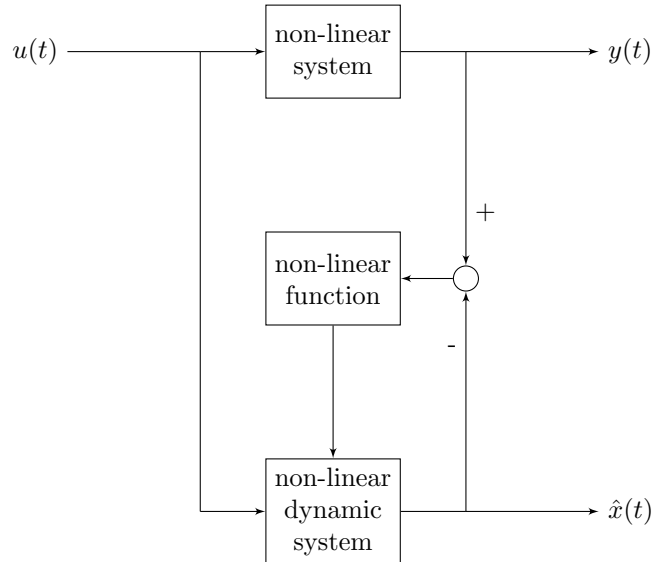
$$J_N(\theta) = \frac{1}{N} \sum_{t=1}^N (x(t) - (S_{ux}(z, \theta)u(t) + S_{yx}(z, \theta)y(t)))^2$$

### Optimization

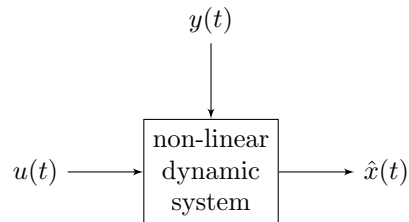
$$\hat{\theta}_N = \arg \min_{\theta} J_N(\theta)$$

We get  $S_{ux}(z, \hat{\theta}_N)$  and  $S_{yx}(z, \hat{\theta}_N)$ , the transfer functions for our software sensors

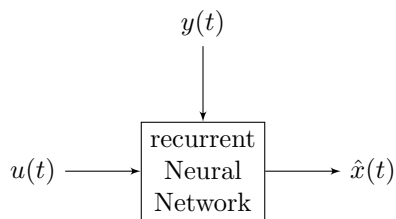
## 6.2 Non-linear systems



**Model**

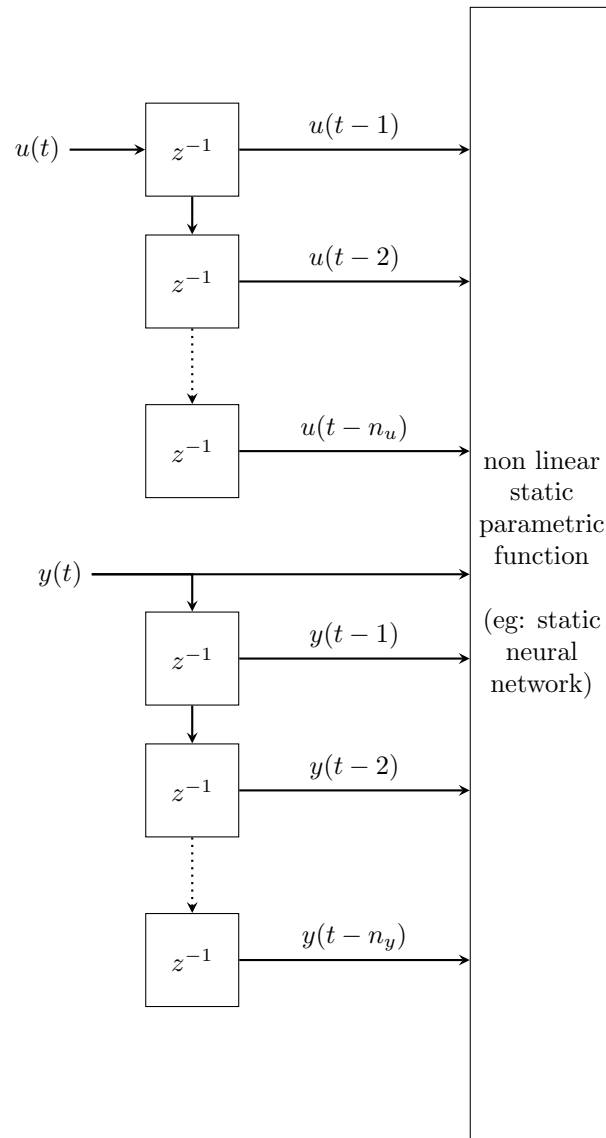


### 6.2.1 Recurrent neural network



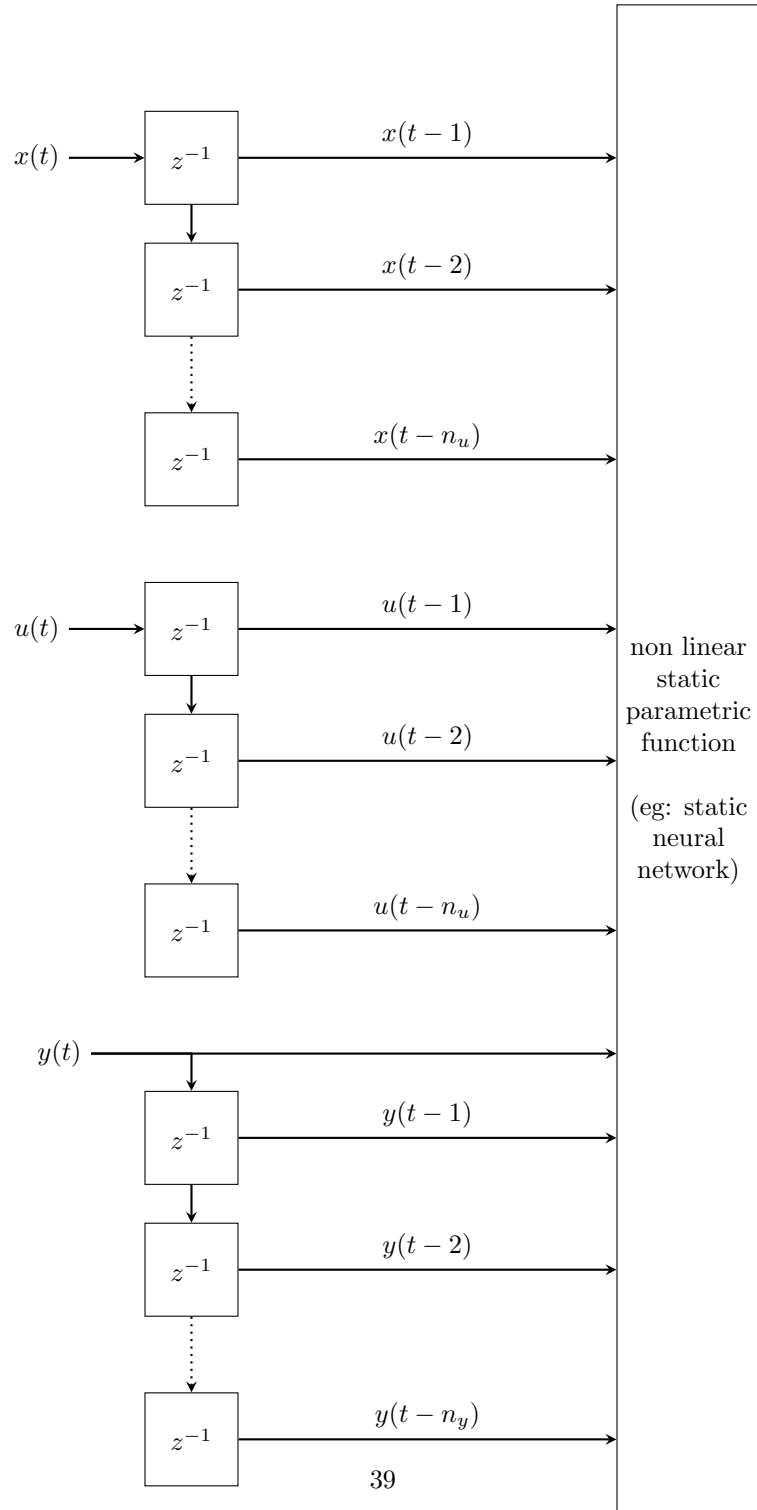
### 6.2.2 FIR architecture

split the system into a static non-linear system and linear dynamics





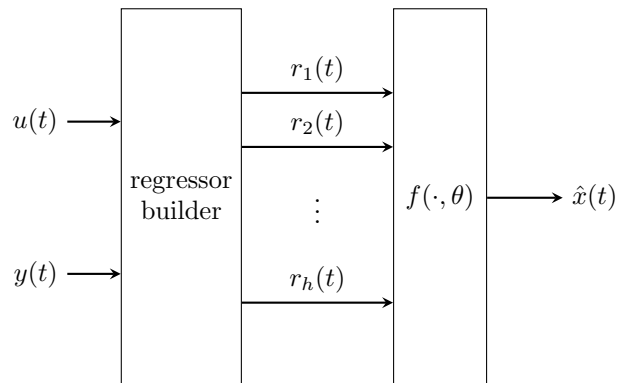
### 6.2.3 IRR scheme





### 6.2.4 Physical regressors

Using the physical knowledge of the system, provide a set of regressors (ie: smaller and more meaningful set of signals) elaborated by a static non-linear system



## Chapter 7

# Grey-box System Identification

### 7.1 Using Kalman Filter

#### 7.1.1 Problem definition

- We have a model:

$$S : \begin{cases} x(t+1) = f(x(t), u(t), \theta) + v_1(t) \\ y(t) = h(x(t), \theta) + v_2(t) \end{cases}$$

- $f$  and  $h$  are functions (linear or not) depending on some unknown parameter  $\theta$  carrying physical meaning (mass, resistance,...)
- We want to estimate  $\hat{\theta}$
- This is achieved by managing the unknown parameters as extended states

#### 7.1.2 State extension

$$S : \begin{cases} x(t+1) = f(x(t), u(t), \theta(t)) + v_1(t) \\ \theta(t+1) = \theta(t) + v_\theta(t) \\ y(t) = h(x(t), \theta(t)) + v_2(t) \end{cases}$$

And the extended state vector is  $x_E = \begin{bmatrix} x(t) \\ \theta(t) \end{bmatrix}$

The noise in the equation of  $\theta$  is added to prevent the KF from getting stuck on the initial conditions.

### 7.1.3 Design choice

The choice of the covariance matrix of  $v_\theta \sim WN(0, V_\theta)$

- Assume  $v_1 \perp v_\theta$  and  $v_2 \perp v_\theta$ :

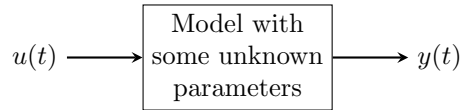
$$V_\theta = \begin{bmatrix} \lambda_{1\theta}^2 & & & \\ & \lambda_{2\theta}^2 & & \\ & & \ddots & \\ & & & \lambda_{n_\theta\theta}^2 \end{bmatrix}_{n_\theta \times n_\theta}$$

- Usually, it is assumed that  $\lambda_{i\theta} = \lambda_{j\theta} \quad \forall i \forall j$
- Assume that  $v_\theta(t)$  is a set of independent  $WN$  all with the same variance  $\lambda_\theta^2$
- Bigger values of  $\lambda_\theta^2$  lead to a quicker convergence, but less stable (stronger oscillations around the steady-state)
- The selection of  $\lambda_\theta^2$  is leaded by application-specific constraints

### 7.1.4 Applicability

In theory, this trick can work with any number of sensors, states, and parameters. In practice it works well only on a limited number of parameters ( $\sim 3$  sensors, 5 states, 2 parameters)

## 7.2 Simulation Error Method



### 7.2.1 Dataset

from an experiment, collect:

$$\begin{aligned} &\{\bar{u}(1), \bar{u}(2), \dots, \bar{u}(N) \\ &\{\bar{y}(1), \bar{y}(2), \dots, \bar{y}(N) \end{aligned}$$

### 7.2.2 Model

$$y(t) = \mathcal{M}(u(t), \bar{\theta}, \theta)$$

$\bar{\theta}$  is the set of known parameters,  $\theta$  the set of unknown parameters

### 7.2.3 Performance index

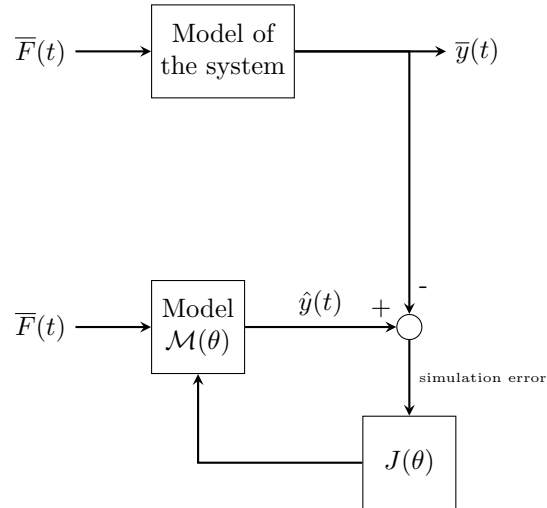
$$J_N(\theta) = \frac{1}{N} \sum_{t=1}^N (\bar{y}(t) - \mathcal{M}(\bar{u}(t), \bar{\theta}, \theta))^2$$

### 7.2.4 Optimization

$$\hat{\theta}_N = \arg \min_{\theta} J_N(\theta)$$

### 7.2.5 Limitations

- Usually  $J_N$  has no analytic expression
- Computing the value of  $J_N$  requires an entire simulation from  $t = 1$  to  $t = N$
- Usually  $J_N$  is non-quadratic and non-convex  $\rightarrow$  iterative and randomized optimization must be used
- Computationally demanding



## Chapter 8

# Minimum Variance Control

The goal is to design a feedback system

- Control design is the aim motivation of system identification and software sensing
- MVC is based on prediction theory

### 8.1 Setup the problem

Consider a generic ARMAX model:

$$y(t) = \frac{B(z)}{A(z)}u(t-k) + \frac{C(z)}{A(z)}e(t) \quad e(t) \sim WN(0, \lambda^2)$$

$$B(z) = b_0 + b_1z^{-1} + \dots + b_pz^{-p}$$

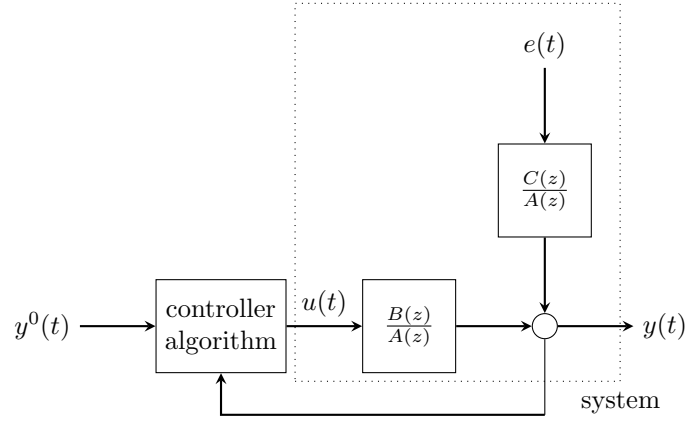
$$A(z) = 1 + a_1z^{-1} + \dots + a_mz^{-m}$$

$$C(z) = 1 + c_1z^{-1} + \dots + c_nz^{-n}$$

#### Assumptions

- $\frac{C(z)}{A(z)}$  is in canonical form
- $b_0 \neq 0 \rightarrow k$  is the actual delay
- $\frac{B(z)}{A(z)}$  is minimum phase  $\iff$  all the roots of  $B(z)$  are strictly inside the unit circle
- $y^0(t) \perp e(t)$
- $y^0(t)$  is known only up to the present time ( $y^0(t)$  is totally unpredictable)

We want to determine the optimal tracking of the desired output behaviour:



Formally, MVC tries to minimize the following performance index:

$$J = E [(y(t) - y^0(t))^2]$$

Which is the variance of the tracking error

### 8.1.1 Simplified problem 1

$$S : y(t) = ay(t-1) + b_0u(t-1) + b_1u(t-2) \quad y(t) = \frac{b_0 + b_1z^{-1}}{1 - az^{-1}}u(t-1)$$

Assuming:

- $y^0(t) = \bar{y}^0$
- no noise
- $b_0 \neq 0$
- Root of numerator inside the unit circle

$$\begin{aligned} J &= (y(t) - y^0(y))^2 \\ &= (y(t) - \bar{y}^0)^2 \\ &= (ay(t-1) + b_0u(t-1) + b_1u(t-2) - \bar{y}^0)^2 \\ &= (ay(t) + b_0u(t) + b_1u(t-1) - \bar{y}^0)^2 \\ \frac{\delta J}{\delta u(t)} &= 2 \left( ay(t) + b_0u(t) + b_1 \underbrace{u(t-1)}_{\substack{\text{number} \\ \text{not variable}}} - \bar{y}^0 \right) (b_0) \end{aligned}$$

$$\frac{\delta J}{\delta u(t)} = 0 \implies ay(t) + b_0u(t) + b_1u(t-1) - \bar{y}^0 = 0 \implies u(t) = (\bar{y}^0 - ay(t)) \frac{1}{b_0 + b_1z^{-1}}$$

### 8.1.2 Simplified problem 2

$$S : y(t) = ay(t-1) + b_0u(t-1) + b_1u(t-2) + e(t) \quad e(t) \sim WN(0, \lambda^2)$$

**Reference variable**  $y^0(t)$

**Performance index**  $E[(y(t) - y^0(t))^2]$

**Trick** rewrite  $y(t)$  as

$$y(t) = \hat{y}(t|t-1) + e(t)$$

$$k = 1 \implies e(t) = e(t) \implies y(t) = \hat{y}(t|t-1) + e(t)$$

$$\begin{aligned} J &= E[(\hat{y}(t|t-1) + e(t) - y^0(t))^2] \\ &= E[(\hat{y}(t|t-1) - y^0(t) + e(t))^2] \\ &= E[(\hat{y}(t|t-1) - y^0(t))^2] + E[e(t)^2] + \cancel{2E[e(t)(\hat{y}(t|t-1) - y^0(t))]} \end{aligned}$$

$$\arg \min_{y^0(t)} E[(\hat{y}(t|t-1) - y^0(t))^2] + E[e(t)^2] = \arg \min_{y^0(t)} E[(\hat{y}(t|t-1) - y^0(t))^2]$$

The best result is when  $\hat{y}(t|t-1) = y^0(t)$ . Writing the system in terms of transfer functions we get:

$$S : y(t) = \frac{b_0 + b_1z^{-1}}{1 - az^{-1}}u(t-1) + \frac{1}{1 - az^{-1}}e(t)$$

Applying the general 1-step predictor for *ARMAX*:

$$\hat{y}(t|t-1) = \frac{b_0 + b_1z^{-1}}{1}u(t-1) + \frac{1 - 1 + az^{-1}}{1}y(t) = (b_0b_1z^{-1})u(t-1) + ay(t-1)$$

Imposing the optimality condition we get:

$$\begin{aligned} b_0u(t-1) + b_1u(t-2) + ay(t-1) &= y^0(t) \\ b_0u(t) + b_1u(t-1) + ay(t) &= y^0(t+1) \\ u(t) &= (y^0(t+1) - ay(t)) \frac{1}{b_0b_1z^{-1}} \end{aligned}$$

Since we cannot predict the future we must approximate:

$$u(t) \approx (y^0(t) - ay(t)) \frac{1}{b_0 + b_1z^{-1}}$$

### 8.1.3 General solution

$$S : y(t) = \frac{B(z)}{A(z)}u(t-k) + \frac{C(z)}{A(z)}e(t) \quad e(t) \sim WN(0, \lambda^2)$$

#### Assumptions

- $b_0 \neq 0$
- $B(z)$  has all roots inside the unit circle
- $\frac{C(z)}{A(z)}$  is in canonical form
- $y^0(t) \perp e(t)$
- $y^0(t)$  is unpredictable

**Trick** rewrite  $y(t) = \hat{y}(t|t-k) + \epsilon(t)$

$$\begin{aligned} J &= E \left[ \left( \hat{y}(t|t-k) + \epsilon(t) - y^0(t) \right)^2 \right] \\ &= E \left[ \left( (\hat{y}(t|t-k) - y^0(t)) + \epsilon(t) \right)^2 \right] \\ &= E \left[ \left( \hat{y}(t|t-k) - y^0(t) \right)^2 \right] + E \left[ \epsilon(t)^2 \right] + 2E \left[ \epsilon(t) (\hat{y}(t|t-k) - y^0(t)) \right] \\ &= E \left[ \left( \hat{y}(t|t-k) - y^0(t) \right)^2 \right] + \text{constant} \end{aligned}$$



## Chapter 9

# Recursive Identification

### 9.1 Least square

$$\hat{\theta}_N = \arg \min_{\theta} \left\{ J_N(\theta) = \frac{1}{N} \sum_{t=1}^N (y(t) - \hat{y}(t|t-1, \theta))^2 \right\}$$

We want to find the predictor model  $\hat{y}(t|t-1, \theta)$

$$y(t) = \phi(t)^T \theta + e(t)$$

Since  $e(t)$  is unpredictable, the best possible 1-step predictor is  $\hat{y}(t|t-1, \theta) = \phi(t)^T \theta$ . Substituting in  $J$ :

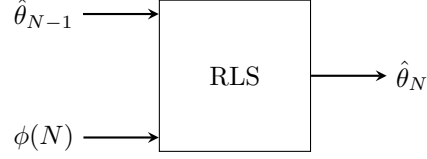
$$J_N(\theta) = \frac{1}{N} \sum_{t=1}^N (y(t) - \phi(t)^T \theta)^2$$

Analytically we can find the minimum:

$$\begin{aligned} \frac{\delta J_N(\theta)}{\delta \theta} &= 0 \\ \hat{\theta}_N &= S(N)^{-1} \sum_{t=1}^N \phi(t) y(t) \\ S(N) &= \sum_{t=1}^N \phi(t) \phi(t)^T \end{aligned}$$

But the procedure must be repeated at each  $t$  step

### 9.1.1 Recursive Least Square



### 9.1.2 First form

$$\begin{aligned}
 \hat{\theta}_N &= S(N)^{-1} \sum_{t=1}^N \phi(t)y(t) \\
 S(N)\hat{\theta}_N &= \sum_{t=1}^N \phi(t)y(t) \\
 S(N-1)\hat{\theta}_{N-1} &= \sum_{t=1}^{N-1} \phi(t)y(t) \\
 \sum_{t=1}^N \phi(t)y(t) &= \sum_{t=1}^{N-1} \phi(t)y(t) + \phi(N)y(N) \\
 \sum_{t=1}^N \phi(t)y(t) &= S(N-1)\hat{\theta}_{N-1} + \phi(N)y(N) \\
 \boxed{\hat{\theta}_N &= \hat{\theta}_{N-1} + S(N)^{-1}\phi(N) \left[ y(N) - \phi(N)^T \hat{\theta}_{N-1} \right]} \\
 S(N) &= S(N-1) + \phi(N)\phi(N)^T
 \end{aligned}$$

However, this way  $S(N) \rightarrow \infty$ , so the computing unit saturates.

### 9.1.3 Second form

Normalize wrt.  $N$

$$\begin{aligned}
 S(N) &= S(N-1) + \phi(N)\phi(N)^T \\
 R(N) &= \frac{1}{N}S(N) \\
 R(N) &= \frac{N-1}{N}R(N-1) + \frac{1}{N}\phi(N)\phi(N)^T \\
 K(N) &= \frac{1}{N}R(N)^{-1}\phi(N) \\
 \epsilon(N) &= y(N) - \phi(N)^T \hat{\theta}_{N-1} \\
 \boxed{\hat{\theta}_N &= \hat{\theta}_{N-1} + K(N)\epsilon(N)}
 \end{aligned}$$

This however requires matrix inversion at each iteration, which is expensive

#### 9.1.4 Third form

It can be proven that the following works

$$\begin{aligned}\epsilon(N) &= y(N) - \phi(N)^T \hat{\theta}_{N-1} \\ V(N) &= V(N-1) - \frac{V(N-1)\phi(N)\phi(N)^T V(N-1)}{1 + \phi(N)^T V(N-1)\phi(N)} \\ K(N) &= V(N)\phi(N) \\ \hat{\theta}_N &= \hat{\theta}_{N-1} + K(N)\epsilon(N)\end{aligned}$$

This does not require matrix inversion

#### 9.1.5 Forgetting factor

If theta contains some parameter changing over time, the objective function should "forget" old data. The new objective function becomes:

$$J_N(\theta) = \frac{1}{N} \rho^{N-t} (\hat{y}(t|t-1, \theta))^2$$

Where  $\rho \in [0, 1]$  is the forgetting factor (the smaller, the more forgetful). The new equations become:

$$\begin{aligned}\epsilon(N) &= y(N) - \phi(N)^T \hat{\theta}_{N-1} \\ S(N) &= \rho S(N-1) + \phi(N)\phi(N)^T \\ K(N) &= S(N)^{-1}\phi(N) \\ \hat{\theta}_N &= \hat{\theta}_{N-1} + K(N)\epsilon(N)\end{aligned}$$

# **Part III**

## **Cheatsheet**

# Appendix A

## MIDA 1

### A.1 Probability Recall

**Cross-Variance**  $Var[v, u] = E[(v - E[v])(u - E[u])]$

**Variance Matrix** 
$$\begin{vmatrix} Var[v_1] & \cdot & \cdot & Var[v_1, v_k] \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ Var[v_k, v_1] & \cdot & \cdot & Var[v_k] \end{vmatrix}$$

**Covariance coefficient**  $\delta[i, j] = \frac{Var[i, j]}{\sqrt{Var[i]}\sqrt{Var[j]}}$

**Stationary process**

- $m$  constant
- $\lambda^2$  constant
- covariance  $\gamma(\tau)$  depends only on time difference
- $|\gamma(\tau)| \leq \gamma(0) \quad \forall \tau$

**White noise**  $\eta(t) \sim WN(m, \lambda^2)$

- Stationary process
- $\gamma(\tau) = 0 \quad \forall \tau \neq 0$
- $v(t) = \alpha\eta(t) + \beta \implies v(t) \sim WN(\beta, \alpha^2\lambda^2)$

### Canonical representation

- Monic
- Same degree
- Coprime
- Poles and zeros in unit disk

## A.2 Spectral analysis

### Spectrum

- $\Gamma(\omega) = \gamma(0) + 2\cos(\omega)\gamma(1) + 2\cos(2\omega)\gamma(2) + \dots$
- Periodic  $T = 2\pi$
- Even
- $\Gamma_\eta(\omega) = \gamma_\eta(0) = \lambda^2$

### Complex spectrum

- $\Phi(z) = \sum_{\tau=-\infty}^{+\infty} \omega(\tau)z^{-\tau}$
- $\Gamma(\omega) = \Phi(e^{j\omega})$

### Fundamental theorem of spectral analysis

- $\Gamma_{\text{out}}(\omega) = |W(e^{j\omega})|^2 \cdot \Gamma_{\text{in}}(\omega)$
- $\Phi_{\text{out}}(z) = W(z)W(z^{-1}) \cdot \Phi_{\text{in}}(z)$

## A.3 Moving Average MA(n)

- $W(z) = \frac{c_0 z^n + c_1 z^{n-1} + \dots + c_n}{z^n}$
- $m = 0$
- $\gamma(\tau) = \begin{cases} (c_0 c_\tau + c_1 c_{1+\tau} + \dots + c_{n-\tau} c_\tau) \lambda^2 & |\tau| \leq n \\ 0 & \text{otherwise} \end{cases}$

### A.3.1 MA( $\infty$ )

- $\gamma(0) = (c_0^2 + c_1^2 + \dots + c_k^2 + \dots) \lambda^2$
- $\gamma(0)$  must converge to a finite value

## A.4 Auto Regressive AR(n)

- $m = 0$
- $W(z) = \frac{z^n}{z^n - a_1 z_{n-1} - \dots - a_n}$
- Covariance calculated by its definition

## A.5 Known predictors

# Appendix B

## MIDA 2

$$\text{AR}(1) \quad \hat{v}(t|t-r) = a^r v(t-r)$$

$$\text{MA}(1) \quad \hat{v}(t|t-1) = v(t-1) - c\hat{v}(t-1|t-2)$$

$$\text{MA}(\mathbf{n}) \quad \hat{v}(t|t-\mathbf{k}) = 0 \quad \forall k > n$$

$$\text{ARMA}(n_a, n_b) \quad \hat{v}(t|t-1) = \frac{C(z)-A(z)}{C(z)}v(t)$$

$$\text{ARMAX}(n_a, n_b) \quad \hat{y}(t|t-1) = \frac{C(z)-A(z)}{C(z)}y(t) + \frac{B(z)}{C(z)}u(t-1)$$

### B.1 State-space representation

$$\begin{cases} x(t+1) = Fx(t) + Gu(t) & \text{state equations} \\ y(t) = Hx(t) + Du(t) & \text{output equations} \end{cases}$$

### B.2 BB representations shift

#### B.2.1 State space to Transfer function

$$W(z) = H(zI - F)^{-1}G$$

#### B.2.2 Transfer Function to State Space

$$W(z) = \frac{b_0 z^{n-1} + b_1 z^{n-2} + \dots + b_{n-1}}{z^n + a_0 z^{n-1} + \dots + a_n}$$

$\Rightarrow$



$$F = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 \\ 0 & 0 & \dots & 0 & 1 \\ -a_n & -a_{n-1} & \dots & \dots & -a_1 \end{bmatrix} \quad G = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} \quad H = [b_{n-1} \quad b_{n-2} \quad \dots \quad b_0] \quad D = 0$$

### B.2.3 Transfer function to Impulse response

$\infty$  long division of  $W(z)$

### B.2.4 Impulse response to Transfer function

Z-transform

$$\mathcal{Z} = \sum_{t=0}^{\infty} s(t)z^{-t}$$

$$W(z) = \mathcal{Z}(\omega(t)) = \sum_{t=0}^{\infty} \omega(t)z^{-1}$$

### B.2.5 State space to Impulse response

$$\omega(t) = \begin{cases} 0 & \text{if } t = 0 \\ HF^{t-1}G & \text{if } t > 0 \end{cases}$$

## B.3 Controllability and Observability

### B.3.1 Fully observable system

$\iff O$  is full rank:

$$O = \begin{bmatrix} H \\ HF \\ \vdots \\ HF^{n-1} \end{bmatrix} \quad \text{rank}(O) = n$$

### B.3.2 Fully controllable system

Fully controllable  $\iff R$  is full rank:

$$R = [G \quad FG \quad \dots \quad F^{n-1}G] \quad \text{rank}(R) = n$$

### B.3.3 Hankel Matrix

Starting from  $\omega(1), \omega(2), \dots, \omega(N)$  where  $N \geq 2n - 1$ , we can build the Hankel Matrix of order  $n$ :

$$H_n = \begin{bmatrix} \omega(1) & \omega(2) & \omega(3) & \dots & \omega(n) \\ \omega(2) & \omega(3) & \omega(4) & \dots & \omega(n+1) \\ \omega(3) & \omega(4) & \omega(5) & \dots & \omega(n+2) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \omega(n) & \omega(n+1) & \omega(n+2) & \dots & \omega(2n-1) \end{bmatrix}$$

$$H_n = \begin{bmatrix} HG & HFG & HF^2G & \dots & HF^{n-1}G \\ \vdots & \ddots & & & \vdots \\ \vdots & & \ddots & & \vdots \\ \vdots & & & \ddots & \vdots \\ HF^{n-1}G & \dots & \dots & \dots & HF^{2n-2}G \end{bmatrix} = \begin{bmatrix} H \\ HF \\ \vdots \\ HF^{n-1} \end{bmatrix} \cdot [G \quad FG \quad \dots \quad F^{n-1}G] = O \cdot R$$

## Appendix C

# System Identification

**Impulse experiment** Measure  $y(t)$  under the input  $u(t) = \text{impulse}(0)(0)$

### C.1 $F, G, H$ from noise-free IR

1.

$$H_1 = [\omega(1)] \quad H_2 = \begin{bmatrix} \omega(1) & \omega(2) \\ \omega(2) & \omega(3) \end{bmatrix} \quad H_3 = \dots \quad \dots \quad H_n = \dots$$

If  $\text{rank}(H_n) = \text{rank}(H_{n+1})$ , then  $n$  is the order of the system

2. Factorize  $H_{n+1}$  as  $H_{n+1} = O_{n+1}[(n+1) \times n] \cdot R_{n+1}[n \times (n+1)]$

$$O_{n+1} = \begin{bmatrix} H \\ HF \\ \vdots \\ HF^n \end{bmatrix} \quad R_{n+1} = [G \quad FG \quad \dots \quad F^n G]$$

3.  $H = O[0]$ ,  $G = R[0]$

4. Define

$$O_1 = \begin{bmatrix} H \\ HF \\ \vdots \\ HF^{n-1} \end{bmatrix} \quad O_2 = \begin{bmatrix} HF \\ \vdots \\ HF^n \end{bmatrix}$$

5.  $F = O_1^{-1} O_2$

## C.2 Obtain $F, G, H$ from a noisy IR (TODO: not well understood)

The measurement is of  $\hat{\omega}(t) = \omega(t) + \eta(t)$ . To identify the process:

1. Build the Hankel matrix using all the  $N$  data:

$$\hat{H}_{q \times d} = \begin{bmatrix} \hat{\omega}(1) & \hat{\omega}(2) & \dots & \hat{\omega}(d) \\ \hat{\omega}(2) & \hat{\omega}(3) & \dots & \hat{\omega}(d+1) \\ \vdots & \vdots & \ddots & \vdots \\ \hat{\omega}(q) & \hat{\omega}(q+1) & \dots & \hat{\omega}(q+d+1 = N) \end{bmatrix}$$

2. Calculate the Singular Value Decomposition of  $\hat{H}_{q \times d}$ :

$$\hat{H}_{q \times d} = \hat{U}_{q \times q} \cdot \hat{S}_{q \times d} \cdot \hat{V}_{d \times d}^T$$

$\hat{U}$  and  $\hat{V}$  are unitary matrices: they are invertible and their inverses are equal to their transpose.

$$\hat{S} = \begin{bmatrix} \sigma_1 & & & \\ & \sigma_2 & & \\ & & \ddots & \\ & & & \sigma_d \end{bmatrix}$$

3. Plot the singular values ( $\sigma_i$ ) and cut-off the three matrices:
  - Ideally, after a certain  $n$  (the order of the IR) there would be a jump dividing the signal (before) from the noise (after)
  - In reality no clear distinction exists, but it's possible to identify an interval of possible values of  $n$ . A tradeoff between complexity, precision and overfitting takes place
4. Split  $\hat{U}, \hat{S}, \hat{V}^T$  obtaining  $U_{q \times n}, S_{n \times n}, V_{n \times d}^T$  and then recreate  $H_{qd} = USV^T$
5.  $H$  and  $G$  are estimated as for the unnoisy case. To estimate  $F$  we can build  $O_1$  and  $O_2$  as before, but then the system  $O_1 \cdot F = O_2$  cannot be solved directly as  $O_1$  is not square. We can instead compute the approximate least-square solution of the system:

$$F = (O_1^T O_1)^{-1} O_1^T O_2$$

# Appendix D

## Kalman Filter

### D.1 Representations

#### D.1.1 For basic systems

$\hat{x}(t+1 t) = F\hat{x}(t t-1) + K(t)e(t)$	state equation
$\hat{y}(t t-1) = H\hat{x}(t t-1)$	output equation
$e(t) = y(t) - \hat{y}(t t-1)$	prediction error
$K(t) = (FP(t)H^T + V_{12}) (HP(t)H^T + V_2)^{-1}$	gain of the KF
$P(t+1) = (FP(t)F^T + V_1)$	
$\quad - (FP(t)H^T + V_{12}) (HP(t)H^T + V_2)^{-1} (FP(t)H^T + V_{12})^T$	difference Riccati equation
$\hat{x}(1 0) = E[x(1)] = X_0$	Initial state
$P(1) = var[x(1)] = P_0$	initial DRE

#### D.1.2 With input

$\hat{x}(t+1 t) = F\hat{x}(t t-1) + Gu(t) + K(t)e(t)$	state equation
other equations = unchanged	

#### D.1.3 Multi-step prediction

Knowing  $\hat{x}(t+1|T)$  from the basic solution we can derive

$$\begin{aligned}\hat{x}(t+k|t) &= F^{k-1}\hat{x}(t+1|t) \\ \hat{y}(t+k|t) &= H\hat{x}(t+k|t)\end{aligned}$$

### D.1.4 Filter ( $\hat{x}(t|t)$ )

$F$  invertible

$$\hat{x}(t+1|t) = F\hat{x}(t|t) \quad \implies \quad \hat{x}(t|t) = F^{-1}\hat{x}(t+1|t)$$

$F$  not invertible assuming  $V_{12} = 0$ :

$$\begin{aligned} \hat{x}(t|t) &= F\hat{x}(t-1|t-1) + Gu(t-1) + K_0(t)e(t) \\ \hat{y}(t|t-1) &= H\hat{x}(t|t-1) \\ e(t) &= y(t) - \hat{y}(t|t-1) \\ K_0(t) &= (P(t)H^T) (HP(t)H^T + V_2)^{-1} \\ P(t+1) &= \text{unchanged} \end{aligned}$$

### D.1.5 Time-varying systems

$$S : \begin{cases} x(t+1) = F(t)x(t) + G(t)u(t) + v_1(t) \\ y(t) = H(t)x(t) + v_2(t) \end{cases}$$

K.F. equations are unchanged

## D.2 Asymptotic KF

### D.2.1 ARE

$$\bar{P} = (F\bar{P}F^T + V_1) - (F\bar{P}H^T + V_{12}) (H\bar{P}H^T + V_2)^{-1} (F\bar{P}H^T + V_{12})^T$$

**First asymptotic theorem**

$V_{12} = 0$  and the system is asymptotically stable (all eigenvalues of  $F$  strictly inside the unit circle)  $\implies$  :

- $\exists!$  solution of ARE:  $\bar{P} \geq 0$
- DRE converges to  $\bar{P} \quad \forall P_0 \geq 0$
- The corresponding  $\bar{K}$  will make the KF asymptotically stable

**Second asymptotic theorem**

$V_{12} = 0$ ,  $(F, H)$  is observable,  $(F, \Gamma)$  is controllable  $\implies$

- $\exists!$  solution of ARE:  $\bar{P} \geq 0$
- DRE converges to  $\bar{P} \quad \forall P_0 \geq 0$
- The corresponding  $\bar{K}$  will make the KF asymptotically stable

### D.3 Non-linear systems

$$F(t) = \left. \frac{\delta f(x(t), u(t))}{\delta x(t)} \right|_{x(t)=\hat{x}(t|t-1)}$$
$$H(t) = \left. \frac{\delta h(x(t))}{\delta x(t)} \right|_{x(t)=\hat{x}(t|t-1)}$$

### D.4 Optimizing $K$

## Appendix E

# Minimum Variance Control

$$y(t) = ay(t-1) + b_0u(t-1) + b_1u(t-2) + e(t) \implies u(t) \approx (y^0(t) - ay(t)) \frac{1}{b_0 + b_1z^{-1}}$$