

# Pole placement control: state space and polynomial approaches

## Lecture 2

O. Sename<sup>1</sup>

<sup>1</sup> **Gipsa-lab**, CNRS-INPG, FRANCE  
[Olivier.Sename@gipsa-lab.fr](mailto:Olivier.Sename@gipsa-lab.fr)  
[www.gipsa-lab.fr/~o.sename](http://www.gipsa-lab.fr/~o.sename)

November 21, 2017



# Outline

State feedback control

Pole placement control: a state space approach

Specifications

Observer

Observer-based control

Integral Control

Some important features

Pole placement  
control

O.Sename

State feedback  
control

Pole placement  
control: a state  
space approach

Specifications

Observer

Observer-based  
control

Integral Control

Some important  
features

How to design a controller using a state space representation ?

Two cases are possible :

- ▶ **Static controllers (output or state feedback)**
- ▶ Dynamic controllers (output feedback or **observer-based**)

What for ?

- ▶ Closed-loop stability (of state or output variables)
- ▶ disturbance rejection
- ▶ Model tracking
- ▶ Input/Output decoupling
- ▶ Other performance criteria :  $H_2$  optimal,  $H_\infty$  robust...

# State feedback control

Let consider **continuous-time** linear state space system given by :

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t), & x(0) = x_0 \\ y(t) = Cx(t) + Du(t) \end{cases} \quad (1)$$

- ▶  $x(t) \in \mathbb{R}^n$  is the system state (vector of state variables),
- ▶  $u(t) \in \mathbb{R}^m$  the control input
- ▶  $y(t) \in \mathbb{R}^p$  the measured output
- ▶  $A, B, C$  and  $D$  are real matrices of appropriate dimensions
- ▶  $x_0$  is the initial condition.

$n$  is the order of the state space representation.

# Why state feedback and not output feedback?

Example:  $G(s) = \frac{y(s)}{u(s)} = \frac{1}{s^2 - s}$

1. Give the controllable canonical form considering  $x_1 = y$ ,  $x_2 = \dot{y}$ .

2. **Case of output feedback= Proportional control** :  $u = -K_p y$

- ▶ Compute the closed-loop transfer function and check that the closed-loop poles are given by the roots of the characteristic polynomial  $P_{BF}(s) = s^2 - s + K_p$ .
- ▶ Can the closed-loop system be stabilized ?

3. **Case of state feedback** : consider the control law

$$u = -x_1 - 3x_2 + y_{ref}$$

- ▶ Compute the state space representation of the closed-loop system.
- ▶ What are the poles of the closed-loop systems?
- ▶ Is it stable?
- ▶ If yes why this second control solves the problem?

## Definition

A state feedback controller for a continuous-time system is:

$$u(t) = -Fx(t) \quad (2)$$

where  $F$  is a  $m \times n$  real matrix.

When the system is SISO, it corresponds to :

$$u(t) = -f_1 x_1 - f_2 x_2 - \dots - f_n x_n \text{ with } F = [f_1, f_2, \dots, f_n].$$

When the system is MIMO we have

$$\begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_m \end{bmatrix} = \begin{bmatrix} f_{11} & \dots & f_{1n} \\ \vdots & & \vdots \\ f_{m1} & \dots & f_{mn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$$

# Definition of the state feedback control

A state feedback controller for a continuous-time system is:

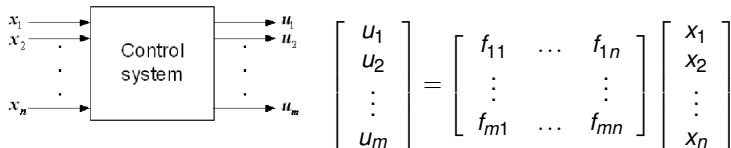
$$u(t) = -Fx(t) \quad (3)$$

where  $F$  is a  $m \times n$  real matrix.

When the system is SISO, it corresponds to :

$u(t) = -f_1 x_1 - f_2 x_2 - \dots - f_n x_n$  with  $F = [f_1, f_2, \dots, f_n]$ .

When the system is MIMO we have





## State feedback (2): stabilization

Using state feedback controllers (6), we get in closed-loop (for simplicity  $D = 0$ )

$$\begin{cases} \dot{x}(t) = (A - BF)x(t), \\ y(t) = Cx(t) \end{cases} \quad (4)$$

The stability (and dynamics) of the closed-loop system is then given by the eigenvalues of  $A - BF$ .

Then the solution  $y(t) = C \exp^{(A-BF)t} x_0$  converges asymptotically to zero!

### ★ Discrete-time systems

Using  $u(k) = F_d x(k)$  we get

$$\begin{cases} x(k+1) = (A_d - B_d F_d)x(k), \\ y(k) = C_d x(k) \end{cases} \quad (5)$$

### Remark

For both cases, the good choice of  $F$  (or  $F_d$ ) may allow to stabilize the closed-loop system.

But what happens if the closed-loop system must track a reference signal  $r$  ?

## State feedback (3): reference tracking

Objective:  $y$  should track some reference signal  $r$ , i.e.

$$y(t) \xrightarrow[t \rightarrow \infty]{} r(t)$$

When the objective is to track some reference signal  $r$ , why not select:

$$u(t) = r(t) - Fx(t) \quad (6)$$

$$\text{or } u(k) = r(k) - F_d x(k) \text{ for discrete-time systems} \quad (7)$$

Can we ensure that  $y$  tracks the reference signal  $r$ , i.e.  $y(t) \xrightarrow[t \rightarrow \infty]{} r(t)$  ?

**No** since, the closed-loop transfer matrix is :

$$\frac{y(s)}{r(s)} = C(sI_n - A + BF)^{-1} B \quad (8)$$

for which the static gain is  $C(-A + BF)^{-1} B$  and may differ from 1 ! (see examples)

## State feedback (4): complete solution for reference tracking

When the objective is to track some reference signal  $r$ , the state feedback control can be selected as:

$$u(t) = -Fx(t) + Gr(t) \quad (9)$$

$G$  is a  $m \times p$  real matrix. Then the closed-loop transfer matrix is :

$$G_{CL}(s) = C(sI_n - A + BF)^{-1}BG \quad (10)$$

$G$  is chosen to ensure a unitary steady-state gain as:

$$G = [C(-A + BF)^{-1}B]^{-1} \quad (11)$$

★ When  $D \neq 0$

$$G_{CL}(s) = [(C - DF)(sI_n - A + BF)^{-1}B + D]G$$

★ Discrete-time systems (with  $D_d =$ )

$$u(k) = -F_dx(k) + G_dr(k) \quad (12)$$

$$\text{with } G_d = [C(I_n - A_d + B_dF_d)^{-1}B_d]^{-1} \quad (13)$$

# Implementation in Simulink

Pole placement  
control

O.Sename

State feedback  
control

Pole placement  
control: a state  
space approach

Specifications

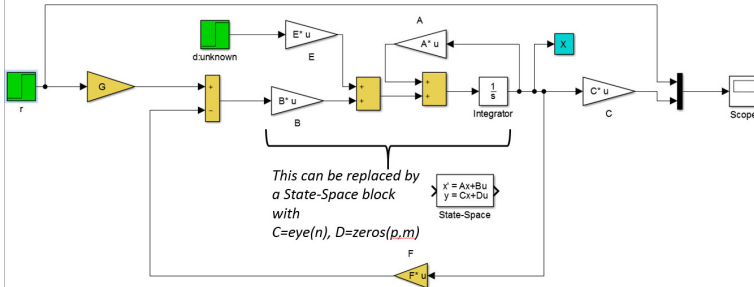
Observer

Observer-based  
control

Integral Control

Some important  
features

Theoretical validation scheme of the state feedback control



# Pole placement control: a state space approach

## Problem definition

Given a linear system (1), does there exist a state feedback control law (6) such that the **closed-loop poles** are in **predefined locations** (denoted  $\gamma_i, i = 1, \dots, n$ ) in the complex plane ?

## Proposition

*Let a linear system given by  $A, B$ , and let  $\gamma_i, i = 1, \dots, n$ , a set of complex elements (i.e. the desired poles of the closed-loop system). There exists a state feedback control  $u = -Fx$  such that the poles of the closed-loop system are  $\gamma_i, i = 1, \dots, n$  **if and only if the pair  $(A, B)$  is controllable.***

## Pole placement control (2): case of the controllable canonical form

Let assume that the system  $G(s) = \frac{c_0 + c_1 s + \dots + c_{n-1} s^{n-1}}{a_0 + a_1 s + \dots + a_{n-1} s^{n-1} + s^n}$  is given by

$$A = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & 0 & \dots \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \vdots & & 0 & 1 \\ -a_0 & -a_1 & \dots & \dots & -a_{n-1} \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} \quad \text{and}$$
$$C = \begin{bmatrix} c_0 & c_1 & \dots & c_{n-1} \end{bmatrix}.$$

Let  $F = [f_1 \quad f_2 \quad \dots \quad f_n]$

Then

$$A - BF = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & 0 & \dots \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \vdots & & 0 & 1 \\ -a_0 - f_1 & -a_1 - f_2 & \dots & \dots & -a_{n-1} - f_n \end{bmatrix} \quad (14)$$

## Pole placement control (3)

From the specifications the **desired** closed-loop polynomial  $(s - \gamma_1)(s - \gamma_2) \dots (s - \gamma_n)$  can be developed as:

$$(s - \gamma_1)(s - \gamma_2) \dots (s - \gamma_n) s^n + \alpha_{n-1} s^{n-1} + \dots + \alpha_1 s + \alpha_0$$

Therefore the chosen solution:

$$f_i = -a_{i-1} + \alpha_{i-1}, i = 1, \dots, n$$

ensures that the poles of  $A - BF$  are  $\{\gamma_i\}, i = 1, n$ .

**Remark:** the case of controllable canonical forms is very important since, when we consider a general state space representation, it is first necessary to use a change of basis to make the system under canonical form, which will simplify a lot the computation of the state feedback control gain  $F$  (see next slide).

**Matlab:** use **F=place(A,B,P)** where  $P$  is the set of desired closed-loop poles (old version **F=acker(A,B,P)**)



## Pole placement control (4)

Procedure for the general case:

1. Check controllability of  $(A, B)$
2. Calculate  $\mathcal{C} = [B, AB, \dots, A^{n-1}B]$ .

$$\text{Note } \mathcal{C}^{-1} = \begin{bmatrix} q_1 \\ \vdots \\ q_n \end{bmatrix}. \text{ Define } T = \begin{bmatrix} q_n \\ q_n A \\ \vdots \\ q_n A^{n-1} \end{bmatrix}^{-1}$$

3. Note  $\bar{A} = T^{-1}AT$  and  $\bar{B} = T^{-1}B$  (which are under the controllable canonical form)
4. Choose the desired closed-loop poles and define the desired closed-loop characteristic polynomial:  
 $s^n + \alpha_{n-1}s^{n-1} + \dots + \alpha_1 s + \alpha_0$
5. Calculate the state feedback  $u = -\bar{F}\bar{x}$  with:

$$\bar{f}_i = -a_{i-1} + \alpha_{i-1}, i = 1, \dots, n$$

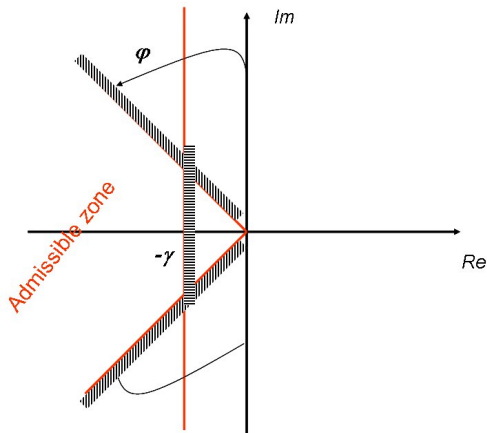
6. Calculate (for the original system):

$$u = -Fx, \text{ with } F = \bar{F}T^{-1}$$

# Specifications: continuous-time case

# Specifications: what should be the closed-loop poles?

The required closed-loop performances should be chosen in the following zone



which ensures a damping greater than  $\xi = \sin \phi$ .

$-\gamma$  implies that the real part of the CL poles are sufficiently negatives.

Some useful rules for selection the desired pole/zero locations (for a second order system):

- ▶ Rise time :  $t_r \simeq \frac{1.8}{\omega_n}$
- ▶ Settling time :  $t_s \simeq \frac{4.6}{\xi \omega_n}$
- ▶ Overshoot  $M_p = \exp(-\pi\xi / \sqrt{1 - \xi^2})$ :
  - $\xi = 0.3 \Leftrightarrow M_p = 35\%$ ,
  - $\xi = 0.5 \Leftrightarrow M_p = 16\%$ ,
  - $\xi = 0.7 \Leftrightarrow M_p = 5\%$ .

## Specifications(3)

Some rules do exist to shape the transient response. The ITAE (Integral of Time multiplying the Absolute value of the Error), defined as:

$$ITAE = \int_0^{\infty} t|e(t)|dt$$

can be used to specify a dynamic response with relatively small overshoot and relatively little oscillation (there exist other methods to do so). The optimum coefficients for the ITAE criteria are given below (see Dorf & Bishop 2005).

| Order | Characteristic polynomials $d_k(s)$   |
|-------|---|
| 1     | $d_1 = [s + \omega_n]$  |
| 2     | $d_2 = [s^2 + 1.4\omega_n s + \omega_n^2]$  |
| 3     | $d_3 = [s^3 + 1.75\omega_n s^2 + 2.15\omega_n^2 s + \omega_n^3]$  |
| 4     | $d_4 = [s^4 + 2.1\omega_n s^3 + 3.4\omega_n^2 s^2 + 2.7\omega_n^3 s + \omega_n^4]$  |
| 5     | $d_5 = [s^5 + 2.8\omega_n s^4 + 5\omega_n^2 s^3 + 5.5\omega_n^3 s^2 + 3.4\omega_n^4 s + \omega_n^5]$                          |
| 6     | $d_6 = [s^6 + 3.25\omega_n s^5 + 6.6\omega_n^2 s^4 + 8.6\omega_n^3 s^3 + 7.45\omega_n^4 s^2 + 3.95\omega_n^5 s + \omega_n^6]$ |

and the corresponding transfer function is of the form:

$$H_k(s) = \frac{\omega_n^k}{d_k(s)}, \quad \forall k = 1, \dots, 6$$

# Specifications(4)

Pole placement  
control

O.Sename

State feedback  
control

Pole placement  
control: a state  
space approach

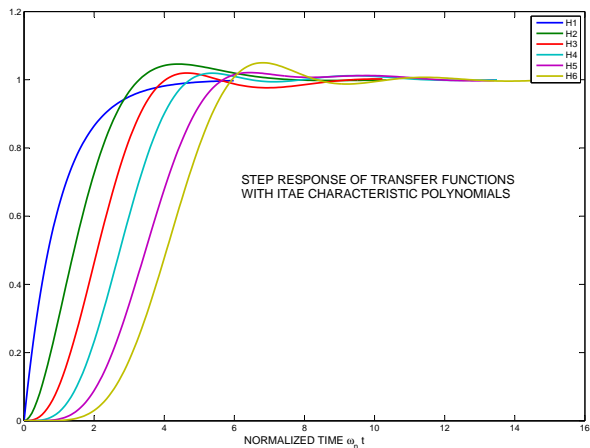
Specifications

Observer

Observer-based  
control

Integral Control

Some important  
features



# Observer

## A first insight

To implement a state feedback control, the measurement of all the state variables is necessary. If this is not available, we will use a state estimation through a so-called *Observer*.

State feedback  
controlPole placement  
control: a state  
space approach

Specifications

Observer

Observer-based  
control

Integral Control

Some important  
features

## Observation or Estimation

The **estimation** theory is based on the famous Kalman contribution to filtering problems (1960), and accounts for noise induced problems. The **observation** theory has been developed for Linear Systems by Luenberger (1971), and does not consider the noise effects.

## Other interest of observation/estimation

In practice the use of sensors is often limited for several reasons: feasibility, cost, reliability, maintenance ...

An *observer* is a key issue to estimate unknown variables (then non measured variables) and to propose a so-called *virtual sensor*.

**Objective:** Develop a dynamical system whose state  $\hat{x}(t)$  satisfies:

- ▶  $(x(t) - \hat{x}(t)) \xrightarrow[t \rightarrow \infty]{} 0$
- ▶  $(x(t) - \hat{x}(t)) \rightarrow 0$  as fast as possible



# Open loop observers: a first approach to estimation from input data

Let consider

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t), & x(0) = x_0 \\ y(t) = Cx(t) + Du(t) \end{cases} \quad (15)$$

Given that we know the plant matrices and the inputs, we can just perform a simulation that runs in parallel with the system

$$\begin{cases} \dot{\hat{x}}(t) = A\hat{x}(t) + Bu(t), & \text{given } \hat{x}(0) \\ \hat{y}(t) = C\hat{x}(t) + Du(t) \end{cases} \quad (16)$$

Therefore, if we would have  $\hat{x}(0) = x(0)$ , then  $\hat{x}(t) = x(t), \forall t \geq 0$ .

**BUT**

- ▶  $x(0)$  is **UNKNOWN** so we cannot choose  $\hat{x}(0) = x(0)$ ,
- ▶ the estimation error ( $e = x - \hat{x}$ ) dynamics is determined by  $A$ , i.e satisfies  $\dot{e}(t) = Ae(t)$  (could be unstable AND cannot be modified)
- ▶ the effects of disturbance and noise cannot be attenuated (leads to estimation bias)

NEED FOR A FEEDBACK FROM MEASURED OUTPUTS TO  
CORRECT THE ESTIMATION !

# Closed-loop Observer: estimation from input AND output data

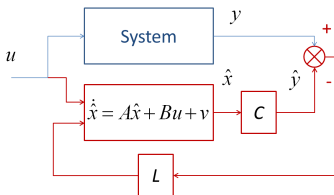
**Objective:** since  $y$  is KNOWN (measured) and is function of the state variables, use an on line comparison of the measured system output  $y$  and the estimated output  $\hat{y}$ .

**Observer description:**

$$\dot{\hat{x}}(t) = A\hat{x}(t) + Bu(t) + \underbrace{L(y(t) - \hat{y}(t))}_{\text{Correction}} \quad (17)$$

$\hat{x}_0$  to be defined

where  $\hat{x}(t) \in \mathbb{R}^n$  is the estimated state of  $x(t)$  and  $L$  is the  $n \times p$  constant observer gain matrix to be designed.



# Analysis of the observer properties

The estimated error,  $e(t) := x(t) - \hat{x}(t)$ , satisfies:

$$\dot{e}(t) = (A - LC)e(t) \quad (18)$$

If  $L$  is designed such that  $A - LC$  is stable, then  $\hat{x}(t)$  converges asymptotically towards  $x(t)$ .

## Proposition

(17) is an observer for system (1) if and only if the pair  $(C, A)$  is observable, i.e.

$$\text{rank}(\mathcal{O}) = n$$

$$\text{where } \mathcal{O} = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{n-1} \end{bmatrix}.$$

# Observer design

The observer design is restricted to find  $L$  such that  $A - LC$  is stable (ensuring that  $(x(t) - \hat{x}(t)) \xrightarrow[t \rightarrow \infty]{} 0$ ), and has some desired eigenvalues (ensuring that  $(x(t) - \hat{x}(t)) \rightarrow 0$  as fast as possible). This is still a pole placement problem.

## Specifications

Select the observer poles according to the systems closed-loop dynamical behavior (see later).

## Design method

- ▶ In order to use the **acker** or **place** Matlab functions, we will use the duality property between observability and controllability, i.e. :  $(C, A)$  observable  $\Leftrightarrow (A^T, C^T)$  controllable.
- ▶ Then there exists  $L^T$  such that the eigenvalues of  $A^T - C^T L^T$  can be randomly chosen. As  $(A - LC)^T = A^T - C^T L^T$  then  $L$  exists such that  $A - LC$  is stable.
- ▶ **Matlab** : use **`L=acker(A',C',Po)`** where **Po** is the set of desired observer poles.

# Theoretical validation scheme using Simulink

Pole placement  
control

O.Sename

State feedback  
control

Pole placement  
control: a state  
space approach

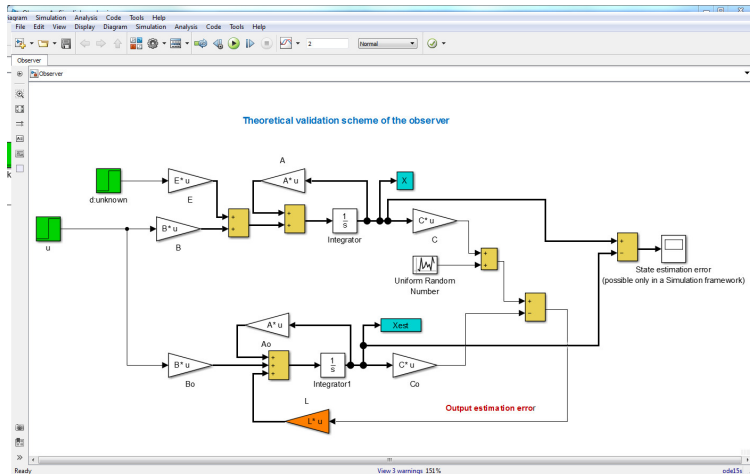
Specifications

Observer

Observer-based  
control

Integral Control

Some important  
features



# How to choose the observer poles ?

## First

This is quite important to avoid that the observer makes the closed-loop system slower. So the observer should be faster than you intend to make the regulator.

## Second

Increasing the observe gain is actually possible since there is no saturation problem.

However the measured outputs are often noisy.

Trade-off between high bandwidth observers (very efficient for estimation but noise sensitive) and low bandwidth ones (less than noise sensitive but slower)

## Rule of thumb

Usually the observer poles are chosen around 5 to 10 times higher than the closed-loop system, so that the state estimation is good as early as possible.

## About the robustness of the observer

Let assume that the systems is indeed given by

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t) + E d_x(t), & x(0) = x_0 \\ y(t) = Cx(t) + D d_y(t) \end{cases} \quad (19)$$

where  $d_x$  can represent input disturbance or modelling error, and  $d_y$  stands for output disturbance or measurement noise.

Then the estimated error satisfies:

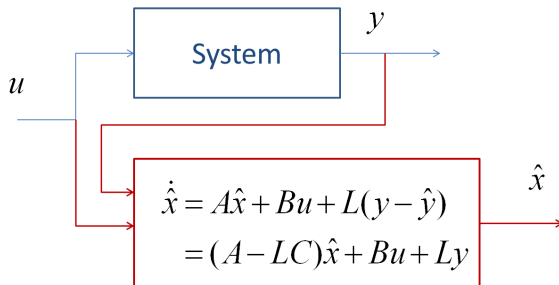
$$\dot{e}(t) = (A - LC)e(t) + E d_x - L D d_y \quad (20)$$

Therefore the presence of  $d_x$  or  $d_y$  may lead to non zero estimation errors due to bias or variations. Then do not forget that you can:

- ▶ Provide an analysis of the observer performances/robustness due to  $d_x$  or  $d_y$  (see later)
- ▶ Design optimal observer when  $d_x$  and  $d_y$  represent noise effects (Kalman - 1960, see next course)
- ▶ Design robust observer using  $H_\infty$  approach (see next year)

## Rules

- ▶ use a state-space block in Simulink
- ▶ enter 'formal' matrices ' $A' = A - LC$ ', ' $B' = [B \quad L]$ ', ' $C' = \text{eye}(n)$ ', ' $D' = \text{zeros}(n, m)$ '
- ▶ Choose  $\hat{x}(0) \neq x(0)$ ,



- ▶ alternative use of **estim**

State feedback  
controlPole placement  
control: a state  
space approach

Specifications

Observer

Observer-based  
control

Integral Control

Some important  
features



# Observer-based control

When an observer is built, we will use as control law:

$$u(t) = -F\hat{x}(t) + Gr(t) \quad (21)$$

The closed-loop system is then (considering here  $D = 0$ )

$$\begin{cases} \dot{x}(t) = (A - BF)x(t) + BF(x(t) - \hat{x}(t)), \\ y(t) = Cx(t) \end{cases} \quad (22)$$

Therefore the fact that  $\hat{x}(0) \neq x(0)$  will have an impact on the closed-loop system behavior.

The stability analysis of the closed-loop system with an observer-based state feedback control needs to consider an extended state vector as:

$$x_e(t) = \begin{bmatrix} x(t) & e(t) \end{bmatrix}^T$$

# Observer-based control: stability analysis

Defining

$$x_e(t) = \begin{bmatrix} x(t) & e(t) \end{bmatrix}^T$$

The closed-loop system with observer (17) and control (21) is:

$$\dot{x}_e(t) = \begin{bmatrix} A - BF & BF \\ 0 & A - LC \end{bmatrix} x_e(t) + \begin{bmatrix} BG \\ 0 \end{bmatrix} r(t) \quad (23)$$

The characteristic polynomial of the extended system is:

$$\det(sI_n - A + BF) \times \det(sI_n - A + LC)$$

If the observer and the control are designed separately then the closed-loop system with the dynamic measurement feedback is stable, given that the control and observer systems are stable and the eigenvalues of (23) can be obtained directly from them.

This corresponds to the so-called **separation principle**.

**Remark:** check **pzmap** of the extended closed-loop system.

State feedback  
control

Pole placement  
control: a state  
space approach

Specifications

Observer

Observer-based  
control

Integral Control

Some important  
features

The closed-loop system from  $r$  to  $y$  is then computed from:

$$y = [C \ 0] \begin{bmatrix} x(t) & e(t) \end{bmatrix}^T$$

which leads to

$$\frac{y}{r} = C(sI_n - A + BF)BG$$

However if some disturbance acts as for:

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t) + Ed(t), & x(0) = x_0 \\ y(t) = Cx(t) \end{cases} \quad (24)$$

where  $d$  is the disturbance, then the extended system writes

$$\dot{x}_e(t) = \begin{bmatrix} A - BF & BF \\ 0 & A - LC \end{bmatrix} x_e(t) + \begin{bmatrix} BG \\ 0 \end{bmatrix} r(t) + \begin{bmatrix} E \\ E \end{bmatrix} d(t) \quad (25)$$

which is a problem for the performances of closed-loop system and of the estimation (see later the Integral control).

State feedback  
controlPole placement  
control: a state  
space approach

Specifications

Observer

Observer-based  
control

Integral Control

Some important  
features

# The controller

The observer-based controller is nothing else than a **2-DOF Dynamic Output Feedback** controller. Indeed it comes from

$$\begin{cases} \dot{\hat{x}}(t) &= A\hat{x}(t) + Bu(t) - L(C\hat{x}(t) - y(t)) \\ u(t) &= -F\hat{x}(t) + Gr(t) \end{cases} \quad (26)$$

which can be written as

$$\begin{cases} \dot{\hat{x}}(t) &= (A - BF - LC)\hat{x}(t) + BGr(t) + Ly(t) \\ u(t) &= -F\hat{x}(t) + Gr(t) \end{cases} \quad (27)$$

We then can write:

$$U(s) = K_r(s)R(s) - K_y(s)Y(s)$$

with

$$K_r(s) = G - F(sI_n - A + BF + LC)^{-1}BG$$

$$K_y(s) = F(sI_n - A + BF + LC)^{-1}L$$

and the analysis can be done through the sensitivity functions.

# Integral Control

# Integral Control or how to ensure disturbance attenuation with a state feedback control?

Preliminary remark: a state feedback controller may not allow to reject the effects of disturbances (particularly of input disturbances).

Let us consider the system:

$$\begin{cases} \dot{x}(t) = Ax(t) + Bu(t) + Ed(t), & x(0) = x_0 \\ y(t) = Cx(t) \end{cases} \quad (28)$$

where  $d$  is the disturbance.

The objective is to keep  $y$  following a reference signal  $r$  even in the presence of  $d$ , i.e

- ▶  $\frac{y}{r} \xrightarrow[t \rightarrow \infty]{} 1$
- ▶  $\frac{y}{d} \xrightarrow[t \rightarrow \infty]{} 0$

# Formulation of the Integral Control

## Introduction

A very useful method consists in adding an integral term (as usual on the tracking error) to ensure a unitary static closed-loop gain, therefore to choose

$$u(t) = -Fx(t) - H \int_0^t (r(\tau) - y(\tau)) d\tau$$

But the question is: how to find  $H$ ?

## The state space method

It consists in extending the system by adding a new state variable:

$$\dot{z}(t) = r(t) - y(t)$$

which leads to define the extended state vector  $\begin{bmatrix} x \\ z \end{bmatrix}$ .

Then the new **open-loop** state space representation is given as:

$$\begin{bmatrix} \dot{x}(t) \\ \dot{z}(t) \end{bmatrix} = \begin{bmatrix} A & 0 \\ -C & 0 \end{bmatrix} \begin{bmatrix} x \\ z \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(t) + \begin{bmatrix} B \\ 0 \end{bmatrix} r(t) + \begin{bmatrix} E \\ 0 \end{bmatrix} d(t)$$
$$y(t) = \begin{bmatrix} C & 0 \end{bmatrix} \begin{bmatrix} x \\ z \end{bmatrix}$$



# Synthesis of the Integral Control

Let us define:

$$A_e = \begin{bmatrix} A & 0 \\ -C & 0 \end{bmatrix}, \quad B_e = \begin{bmatrix} B \\ 0 \end{bmatrix}, \quad C_e = \begin{bmatrix} C & 0 \end{bmatrix}$$

The new state feedback control is now of the form

$$\begin{aligned} u(t) &= -F_e \begin{bmatrix} x \\ z \end{bmatrix} \\ &= -Fx(t) - Hz(t) \text{ denoting } F_e = [F \ H] \end{aligned}$$

Then the synthesis of the control law  $u(t)$  requires:

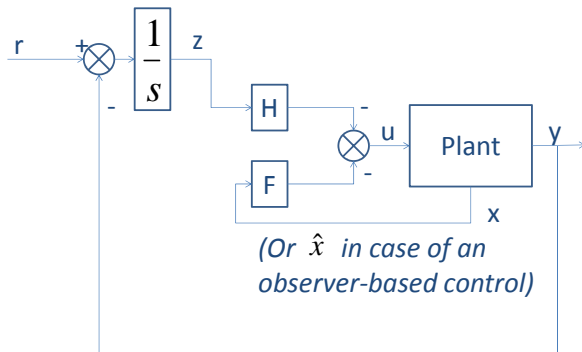
- ▶ the verification of the extended system controllability
- ▶ the specification of the desired closed-loop performances, i.e. a set  $P_e$  of  $n+1$  desired closed-loop poles has to be chosen,
- ▶ the computation of the full state feedback  $F_e$  using  
 **$F_e = \text{acker}(A_e, B_e, P_e)$**

We then get the **closed-loop** system

$$\begin{aligned} \begin{bmatrix} \dot{x}(t) \\ \dot{z}(t) \end{bmatrix} &= \begin{bmatrix} A - BF & BH \\ -C & 0 \end{bmatrix} \begin{bmatrix} x \\ z \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} r(t) + \begin{bmatrix} E \\ 0 \end{bmatrix} d(t) \\ y(t) &= \begin{bmatrix} C & 0 \end{bmatrix} \begin{bmatrix} x \\ z \end{bmatrix} \end{aligned}$$

# Integral control scheme

The complete structure has the following form:



State feedback  
control

Pole placement  
control: a state  
space approach

Specifications

Observer

Observer-based  
control

**Integral Control**

Some important  
features

When an observer is to be used, the control action simply becomes:

$$u(t) = -F\hat{x}(t) - Hz(t)$$

# Integral control scheme

Pole placement  
control

O.Sename

State feedback  
control

Pole placement  
control: a state  
space approach

Specifications

Observer

Observer-based  
control

**Integral Control**

Some important  
features

Compute the closed-loop system representation and check that:

- ▶ the closed-loop system has a unitary gain
- ▶ the effect of the disturbance  $d$  in steady-state is nul

# Some important features

## Some extension: reduced-order observers

In practice, since some output variables are measured it may be non necessary to get an estimation of all the state variables.

For instance, if :

$$x(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix}$$

and  $y(t) = x_1(t)$  is measured, then it possible to estimate only  $x_2(t)$ , which is referred to as a reduced-order observer (order =  $n - p$ ).

N.B: the use of a full-order observer also induces a filter effect on the measurements.

As seen in (27) the general form of a Dynamic Output Feedback controller is 2-DOF state space representation as:

$$\begin{cases} \dot{x}_K(t) &= A_K x_K(t) + B_K \begin{bmatrix} r(t) \\ y(t) \end{bmatrix} \\ u(t) &= C_K x_K(t) + D_K \begin{bmatrix} r(t) \\ y(t) \end{bmatrix} \end{cases} \quad (29)$$

with

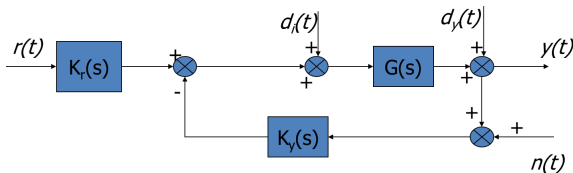
$$A_K = A - BF - LC, \quad B_K = \begin{bmatrix} BG \\ L \end{bmatrix}, \quad C_K = -F, \quad D_K = \begin{bmatrix} G \\ 0 \end{bmatrix}$$

This allows to compute the well known sensitivity functions since, from  $K(s) = D_K + C_K(SI_n - A_K)^{-1}B_K$ , we can write:

$$U(s) = K(s) \begin{bmatrix} R(s) \\ Y(s) \end{bmatrix} := K_r(s)R(s) - K_y(s)Y(s)$$

# Performance analysis (SISO)

The general control scheme is of the form



The closed-loop system satisfies the equations

$$y = \frac{1}{1+G(s)K_y(s)} (GK_r r + d_y - GK_y n + Gd_i)$$
$$u = \frac{1}{1+K_y(s)G(s)} (K_r r - K_y d_y - K_y n - K_y Gd_i)$$

Defining the Sensitivity function:

$$S(s) = \frac{1}{1 + G(s)K_y(s)}$$

The performance sensitivity functions are then

$$\frac{y}{r} = \frac{GK_r}{1+GK_y} = SGK_r \quad \frac{y}{d_i} = SG \quad \frac{y}{d_y} = S \quad \frac{y}{n} = -T$$
$$\frac{u}{r} = K_r S \quad \frac{u}{d_i} = -T_y \quad \frac{u}{d_y} = -K_y S \quad \frac{u}{n} = -K_y S$$

# Performance analysis (MIMO)

The closed-loop system satisfies the equations

$$\begin{aligned}(I_p + G(s)K_y(s))y(s) &= (GK_r r + d_y - GK_r n + Gd_i) \\ (I_m + K_y(s)G(s))u(s) &= (K_r r - K_y d_y - K_y n - K_y Gd_i)\end{aligned}$$

Defining

Output and Output complementary sensitivity functions:

$$S_y = (I_p + GK_y)^{-1}, \quad T_y = (I_p + GK_y)^{-1} GK_y, \quad S_y + T_y = I_p$$

Input and Input complementary sensitivity functions:

$$S_u = (I_m + K_y G)^{-1}, \quad T_u = K_y G (I_m + K_y G)^{-1}, \quad S_u + T_u = I_m$$

and the performance sensitivity functions are then

$$\begin{aligned}\frac{y}{r} &= S_y GK_r & \frac{y}{d_i} &= S_y G & \frac{y}{d_y} &= S_y & \frac{y}{n} &= -T_y \\ \frac{u}{r} &= S_u K_r & \frac{u}{d_i} &= -S_u K_y G & \frac{u}{d_y} &= -S_u K_y & \frac{u}{n} &= -S_u K_y\end{aligned}$$