People's Democratic Republic of Algeria
Ministry of higher education and scientific research

University of Mustapha Stambouli mascara
Faculty of science and technology
Department of electrotechnics

Course handout

Control system 2

3rd-year engineer

Electrotechnics

Specialty: Electrotechnics

Academic year 2024/2025

PREFACE

The course *Control Systems(Asservissement) 02* continues the automation training of third-year engineering students. It aims to deepen the understanding of control systems, particularly in their modeling, analysis, and synthesis, both in continuous and discrete time. In an industrial context where performance, stability, and precision of automated systems are essential requirements, mastering control techniques becomes a key skill for any engineer in automation, electromechanics, or electrical engineering.

This module enables students to develop advanced skills in the design of controllers (P, PI, PD, PID), processing of sampled signals, use of digital regulators, as well as analysis and synthesis in state space. It prepares future engineers to tackle real-world control problems in various industrial sectors: electric machines, automated processes, embedded systems, etc.

This course material is structured into five chapters that support a clear pedagogical progression:

Chapter 1: Introduction to Control

Chapter 2: Compensation of Linear Control Systems

Chapter 3: Analysis of Sampled Systems

Chapter 4: Synthesis of Sampled Control Systems

Chapter 5: Analysis and Synthesis in State Space

Each chapter combines theoretical reviews, design methods, and practical applications to promote both analytical and hands-on understanding of control systems.

This coursebook thus serves as a reference tool for final-year engineering students and may also be useful to anyone wishing to strengthen their knowledge in automatic control and regulation.

Table of contents

Objectives 4

I - Introduction to control systems 5

1. Regulation Concepts 5
2. Components of a Control Loop 6
3. Key Variables in Control Systems 8
4. Representation of a Controlled System 8

II - Correction of Linear Feedback Systems 11

1. Specifications of a Control System 11
2. Necessity of Correction in Control Systems 12
3. Correction (or Compensation) Strategy for Control Systems 13
   3.1. Correction Methods 13
4. Structure of P,I,D, PI, PD, PID correctors 14
   4.1. structure of P,I,D, PI, PD, PID correctors 14
5. Phase delay corrector 19
6. Phase advance corrector 22
7. Selection Criteria and Tuning Methods 24
8. Controller Tuning by Imposing a Reference Model 25
9. Application: Speed Control of a DC Motor 26

III - Discret time system 29

1. Signals sampling 29
   1.1. The principle of sampling 29
   1.2. Shannon's Theorem 30
2. The Z transform -properties and applications 31
   2.1. Properties of the Z transform 32
   2.2. Z-transform of usual signals 32
   2.3. Initial Value and Final Value Theorem of a Sampled System 33
   2.4. Inverse Z Transform 34
3. Sampled transfert functions 36
4. Association of sampled systems 37
5. Harmonic, impulse, and step response 38
6. Z-domain Transmittance and Frequency Response of a Zero-Order Hold 39
7. Analysis of sampled systems, sampled stability 40

IV - Synthesis of sampled control systems 42

1. Stability, Speed (Rapidity), Static Precision 42

2. Precision 42
3. Standard Controllers (PID) and P-Plane Design 44
4. Numerical controller 44
5. Pseudo-Frequency Synthesis & Bilinear Transformation 47
5.1. Discretization of a continuous transfer 48
5.2. Relationship between continuous systems and sampled systems 48
6. Controller Design in Discrete Time: Classical, Modern, and Empirical Methods 49
V - Analysis and synthesis in state space **52**
1. Definition : State-Space Representation 52
2. Stability in State-Space Systems 54
3. Controllability and Observability 54
3.1. Controllable Modes and Observable Modes 56
VI - Bibliographic references **57**
1. Bibliographic references 57

Objectives

Master the principle and structure of control loops and choice of appropriate regulator.

Study of sampled systems. Analyze discrete systems and synthesize discrete regulators (PID, RST and state feedback)

I Introduction to control systems

1. Regulation Concepts

A controlled system is a system that takes into account, during its operation, the evolution of its outputs to modify them and maintain them in accordance with a setpoint.

This branch of automation is divided into two other sub-branches (artificially separated by use):

- **Regulation**: maintaining a determined variable, constant and equal to a value, called a setpoint, without human intervention. Example: Regulation of the temperature of a room.
- **Servo-controlled systems**: varying a determined quantity according to a law imposed by a comparison element. Example: Regulation of the speed of an engine, Trajectory tracking of a missile.

The goal of an automated system is to replace humans in a given task. To establish the structure of an automated system, we will begin by studying the operation of a system in which humans are the "control party".

Example: driver of a vehicle

The driver must follow the road. To do this, he observes the road and its environment and assesses the distance between his vehicle and the edge of the road. He determines, depending on the context, the angle he must give to the steering wheel to follow the road. He acts on the steering wheel (therefore on the system); then again, he begins his observation throughout the duration of the trip. If a gust of wind deviates the vehicle, after observing and measuring the deviation, he acts to oppose this disturbance.

If we want a servo to replace man in various tasks, it will have to have behavior and organs similar to those of a human being. That is to say, it will have to be able to appreciate, compare and act.

Any servo-system will include these three categories of elements that fulfill the 3 major functions necessary for its proper functioning (fig. 1–1):

- Measurement (or observation)
- Comparison between the goal to be achieved and the current position (Reflection)
- Power action

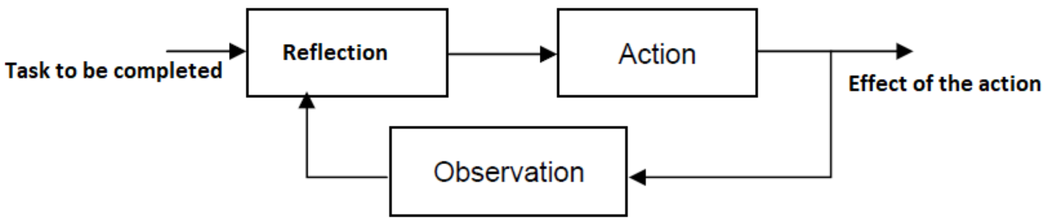

Figure 1.1: General control loop

closed loop and open loop systems

To better understand the concept of a closed-loop system, let's consider an example with two cases. In the first case, we examine an open-loop system and highlight its weaknesses. In the second case, we demonstrate the advantages provided by closing the loop.

First Case: Cannon Fire at a Target

We consider a target to be destroyed and a cannon. To achieve the objective, the firing angle of the cannon and the gunpowder charge of the shell are adjusted based on the target's coordinates and other known parameters at the moment of firing. Once the shell is launched, if these external parameters change, for example, if the target moves, no further adjustments can be made to its trajectory. The shell is essentially left on its own.

Second Case: Cannon Fire at a Target Using a Guided Missile and Radar

Now, consider the same target, but this time with a guided missile. In this case, even if the target moves or a crosswind diverts the missile from its initial trajectory, it will still reach its goal. This is because, at every moment, a radar system provides the respective positions of the missile and the target. By comparing these positions, the trajectory error can be calculated and corrections can be made by adjusting the missile's control surfaces to rectify the error.

In this scenario, the system is no longer left to its own devices because it includes a feedback loop. This loop consists of the radar, which "measures" the missile's position and communicates this information to the operator, and a telecommunication system that allows the trajectory to be adjusted by acting on the missile's control surfaces.

The feedback loop, although introducing a certain level of complexity, provides an enormous gain in precision.

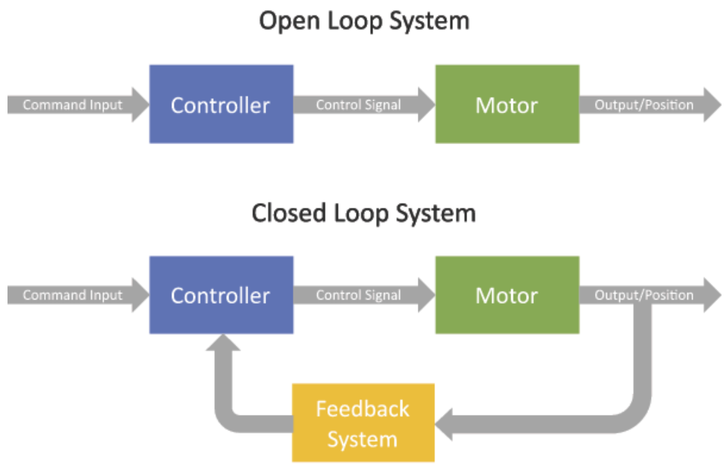

Figure 1.3: open loop Vs closed loop

2. Components of a Control Loop

S Output Variable

The regulated output represents the physical phenomenon the system is designed to control—it is the system's purpose. This can include voltage, displacement, rotation angle, level, speed, etc.

E Input Variable (or reference, or setpoint)

The setpoint is the action input, serving as the controlling variable of the system. Its nature can differ from that of **S**; only its numerical value matters. If **E** and **S** have different natures, a numerical correspondence between these two variables must be defined. For instance, one might specify that one volt at the input represents 100 revolutions per minute.

ε **Error or Deviation** (input - output)

The error, or deviation, is the difference between the setpoint and the output. This measurement can only be made on comparable quantities, so it is generally performed between the setpoint and the measured output. The comparator provides this value, which is proportional to the difference **(E - S')**. It may have a different nature. For example, if **E** and **S'** are voltages, ε could be expressed as a current, such as **ε = (E - S') / R**, where **R** is a resistance

S' Measured Output

The measured output is provided by the feedback chain, generally after some transformation. **S'** must necessarily have the same physical nature as **E**, which is essential to give meaning to the difference **(E - S')**. One of the roles of the feedback chain is thus to ensure the conversion of the measurement of **S** into the physical quantity of **E**.

In general, the system includes:

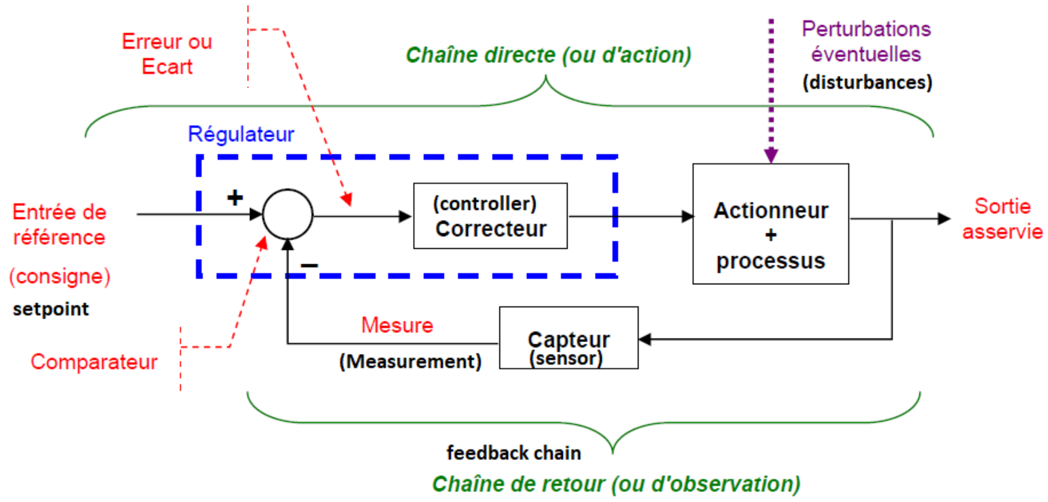

Figure 1.2: Detailed control loop

In any control system, especially in industrial applications, the control loop consists of several interconnected components. Each of these plays a vital role in ensuring the desired output is achieved despite disturbances or variations in system behavior.

1. The Industrial Process (Plant)

- The **process** is the part of the system being controlled. It could be a heating system, a motor, a chemical reactor, a tank level system, etc.
- The goal is to maintain a specific variable (temperature, speed, pressure, level, etc.) at a desired value.

2. Actuators

- **Actuators** are devices that directly influence the process.
- Examples: valves, motors, heaters, pumps.
- They receive control signals from the controller and act upon the process accordingly.

3. Sensors (Measurement Devices)

- **Sensors** measure the current state or output of the process.
- They convert physical quantities (temperature, speed, pressure) into electrical signals.
- Examples: thermocouples, flowmeters, strain gauges.

4. Controllers

- The **controller** compares the measured output to the desired value (setpoint) and computes the control action needed.
- It can be implemented using analog electronics, digital computers, or PLCs.
- Common control types: PID controllers (Proportional, Integral, Derivative).

5. Signal Conditioning

- Signal conditioning devices prepare sensor outputs for the controller.
- They amplify, filter, or convert signals (e.g., analog to digital conversion).
- Important for ensuring accurate and reliable data processing.

6. Setpoint or Reference Signal

- The **setpoint** is the desired value of the output variable.
- It is provided by the user or system and serves as a reference for the controller.

7. Disturbances

- **Disturbances** are unwanted inputs that affect the process output.
- They may come from external sources (e.g., changes in load, temperature) or internal dynamics.
- The control system must compensate for these to maintain stability and performance.

3. Key Variables in Control Systems

To understand how a control system functions, it's essential to identify and distinguish the various types of variables involved. These variables define the behavior of the system and how the control loop reacts to achieve desired performance.

1. Manipulated Variables (Control Variables)

- These are the variables that the controller adjusts to influence the process.
- Example: In a heating system, the manipulated variable could be the power supplied to the heater.
- It is the **output of the controller** and **input to the process** via the actuator.

2. Controlled Variables (Regulated Variables)

- These are the variables that the system aims to maintain at a desired value (setpoint).
- Example: The room temperature in a climate control system.
- The **measured output** of the process that is fed back to the controller.

3. Disturbance Variables

- Disturbances are **unwanted inputs** that affect the controlled variable.
- Example: An open window introducing cold air in a heated room.
- They are not controlled, but the control system must compensate for them.

4. Reference Signal (Setpoint)

- The **desired value** of the controlled variable.
- It is compared to the measured output to compute the error signal used by the controller.

5. Error Signal

- Calculated as:

  $e(t)=r(t)-y(t)$

  where:

  - $e(t)$: error signal
  - $r(t)$: reference (setpoint)
  - $y(t)$: measured output
- This signal guides the control action.

4. Representation of a Controlled System

To understand and analyze a control system, it is essential to use schematic representations (represented above in figure 1.2)and identify the elements of a control loop. These representations make it easier to model, simulate, and optimize control behavior.

Elements of a Control Loop

Component	Function
Process	The system or equipment being controlled (e.g., a tank, motor, heater).

Actuator	Executes the control signal (e.g., motor, valve).
Sensor	Measures the actual output (e.g., temperature sensor, flowmeter).
Controller	Calculates the correction based on the error between setpoint and feedback.
Disturbance	External factors that affect the process (e.g., load, temperature).
Feedback Loop	Returns the measured output to the controller for continuous correction.

Functional Diagrams and Types of Loops

There are two main types of control loops:

- **Open-loop**: no feedback, control action is independent of the output.
- **Closed-loop (Feedback)**: output is measured and compared to setpoint; the error is corrected in real-time.

Performance Criteria of a Control System

To evaluate how well a control system performs, several **quantitative and qualitative criteria** are used. These determine if the system is **stable**, **accurate**, **fast**, and **robust**.

Stability

Stability refers to the ability of a control system to return to a steady state after a disturbance. A stable system ensures that the output does not diverge over time.

- **BIBO Stability (Bounded Input, Bounded Output):**

  A system is BIBO stable if, for any bounded input, the output remains bounded.

In the context of transfer functions, the system is stable if all poles of the transfer function have negative real parts.

Accuracy

Accuracy is a measure of how well the output of the system follows the desired input, often quantified by the **steady-state error**.

- **Steady-State Error:**

  $e_{ss} = \lim_{t \to \infty} [r(t) - y(t)]$

Response Time

Response time measures how quickly the system reaches a steady state after a disturbance or input change.

Key performance metrics related to response time include:

- **Rise Time (tr):** The time it takes for the output to rise from 10% to 90% of its final value.
- **Settling Time (ts):** The time required for the system output to remain within a certain percentage (usually 2% or 5%) of its final value.

For a second-order system, the settling time tst_sts is approximately:

$t_s \approx \frac{4}{\zeta \omega_n}$

Where:

- ζ is the damping ratio
- ω_n is the natural frequency
- **Peak Time (tp):** The time it takes to reach the first peak of the overshoot.

For a second-order system, the peak time is:

$$t_p = \frac{\pi}{\omega_n\sqrt{1-\zeta^2}}$$

- **Overshoot (Mp):** The amount by which the system exceeds the desired value, usually expressed as a percentage.

The overshoot for a second-order system is:

$$M_p = e^{-\frac{\zeta\pi}{\sqrt{1-\zeta^2}}} \times 100$$

Bandwidth

Bandwidth refers to the range of frequencies over which the system can effectively respond to an input. A system with a larger bandwidth can handle a wider range of frequencies, meaning it responds more quickly to changes in the input.

Bandwidth Definition:

The **3 dB bandwidth** is defined as the frequency range where the magnitude of the system's transfer function remains within 3 dB of its peak value.

Robustness

- The system's ability to maintain performance in the face of disturbances or model uncertainties.
- Important in real-world applications where conditions may vary.

Disturbance Rejection

- How well the system can **reject external disturbances** and restore output to the desired value.

Control Effort

- Refers to how much energy or actuation is required to maintain control.
- A good system achieves performance with **minimal actuator effort**.

II Correction of Linear Feedback Systems

Introduction

The correction of linear feedback systems involves adjusting their performance to meet specific criteria such as stability, precision, speed, and robustness.

1. Specifications of a Control System

A **specifications document for a control system** outlines the functional and technical requirements that the system must fulfill. This is a critical step in designing a feedback control system, as it ensures the system meets its intended purpose efficiently and reliably.

Main Components of the Specifications

1. **System Objectives**

Define the purpose of the system, such as maintaining a specific temperature, regulating speed, controlling position, or ensuring stability in dynamic conditions.

2. **Performance Criteria**

Specify quantitative measures to evaluate the system, including:

- **Precision:** The acceptable level of steady-state error or accuracy.
- **Stability:** The system should not oscillate or diverge when subjected to disturbances.
- **Speed of Response:** Characteristics like rise time, settling time, and delay.
- **Robustness:** The system’s ability to maintain performance despite variations in parameters or disturbances.

3. **Input/Output Specifications**

- **Inputs:** Define the type (e.g., voltage, position, force) and range of input signals.
- **Outputs:** Specify the expected range, accuracy, and nature of the system's output.

4. **Operating Conditions**

Outline the environmental and operational constraints, such as:

- Temperature ranges.
- Humidity levels.
- Power supply requirements.
- Physical dimensions and limitations.

5. **Constraints and Limits**

- **System Constraints:** Maximum load, bandwidth, or other physical/technical limits.
- **Safety Constraints:** Ensure safe operation under all conditions, avoiding failures that could cause damage or harm.

6. **Disturbance Rejection**

Specify how the system should handle external disturbances while maintaining output stability and accuracy.

7. **Control Architecture**

Define whether the system will use specific control laws or strategies, such as:

- PID controllers.
- Adaptive control.
- Feedforward or feedback control loops.

Purpose of the Specifications Document

By detailing these requirements, the specifications serve as a blueprint for system design, allowing engineers to:

- Evaluate system performance.
- Ensure compatibility between components.
- Plan for testing and validation phases.

This ensures the system functions correctly in real-world conditions and meets user expectations.

2. Necessity of Correction in Control Systems

Stability and Precision in Control Systems

In the study of control systems, stability and precision are key specifications, often creating a trade-off:

1. **Stability:**
    - Improved by reducing the open-loop gain or bandwidth.
    - Defined by gain margin (lower gain = better stability) and phase margin (lower phase shift = better stability).
2. **Precision:**
    - **Static Precision:** Requires integration in the open-loop transfer function to eliminate steady-state error.
    - **Dynamic Precision:** Improved by increasing the gain or bandwidth.

However, increasing gain for precision reduces stability, making it challenging to achieve both simultaneously. To address this, compensators or controllers adjust the gain in specific frequency ranges to balance stability and precision.

Additionally, a good control system must:

- **Be robust to disturbances (narrow bandwidth).**
- **Respond quickly to input changes (wide bandwidth).**

These conflicting requirements necessitate correctors that shape frequency response (e.g., Bode or Nyquist plots) to:

- Ensure stability with adequate gain and phase margins.
- Maintain high precision at low frequencies.
- Minimize the impact of disturbances without limiting overall bandwidth.

Correctors (electrical, mechanical, or hydraulic) act as filters to balance these requirements effectively.

Controlled systems without proper correction often face limitations, including:

- **Instability**: Poorly designed feedback loops can lead to uncontrolled oscillations.
- **Significant residual error**: Even with feedback, the error between the setpoint and the output may remain high.
- **Slow or imprecise response**: Without correction, dynamic performance (response time, overshoot, damping) may not meet the requirements.

Correction is introduced to address these issues, enhancing both dynamic and static performance of the system.

Objectives of Correction

The goal of correction in controlled systems is to:

- **Minimize or eliminate steady-state error** for specific input types (step, ramp, or parabolic inputs).
- **Stabilize the system** by reducing oscillations and avoiding divergence.
- **Improve dynamic performance** by adjusting:
  - Rise time.
  - Overshoot.
  - Settling time.
  - Bandwidth.
- **Enhance robustness** against disturbances and parameter variations.

Correction is essential in controlled systems to meet industrial requirements and performance standards. It ensures the system achieves the specifications of the design while remaining stable and robust under unexpected conditions.

3. Correction (or Compensation) Strategy for Control Systems

3.1. Correction Methods

Summary: Designing Controllers for Feedback Systems

The purpose of control system analysis tools is to guide the synthesis of corrected or compensated systems, focusing on designing suitable controllers or compensators. This process involves three main steps:

1. **Define System Specifications:**
   - Identify the desired behavior of the controlled system as outlined in the requirements.
2. **Select Controller Configuration:**
   - Decide how the controller will be integrated with the system (e.g., series or parallel).
3. **Tune Controller Parameters:**
   - Calculate the values of the controller's parameters to meet the specified goals.

The ultimate goal is for the output, s(t)s(t), to achieve the desired behavior within a specific time interval by determining the command signal u(t)u(t) that ensures this outcome.

Common Controller Configurations:

Cascade (Series) Correction:

- The controller is placed in series with other elements of the system for direct compensation.

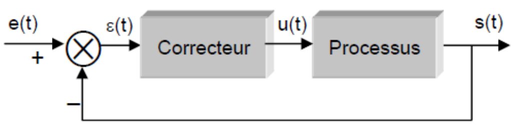

Figure 2.1: Feedback (Parallel) Correction:

Feedback (Parallel) Correction:

- The controller operates in parallel with an element of the system, forming a secondary loop for enhanced correction.

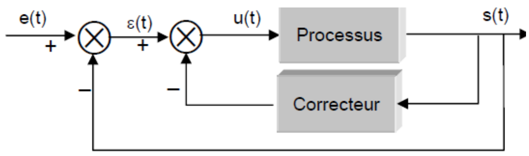

Figure 2.2: Feedback (Parallel) Correction with inner loop

Other configurations, such as **feedforward correction**, are also widely used depending on the system's requirements.

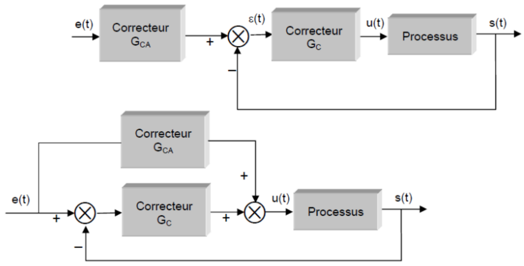

Figure 2.3: Other control loop

4. Structure of P,I,D, PI, PD, PID correctors

4.1. structure of P,I,D, PI, PD, PID correctors

After selecting a correction configuration, the designer must choose the type of controller that, once its parameters are determined, will meet the specifications outlined in the design requirements. However, even here, there is a wide variety of controllers available. In practice, the simplest option is often preferred. The more complex a controller, the higher its cost, the less reliable it is, and the more challenging it is to implement.

The choice of a specific controller for a given application is always based on the designer's experience and, at times, intuition.

Once the controller is selected, the next step is to determine the values of its parameters. These are the coefficients of one or more transfer functions that make up the controller. The fundamental approach involves using the analytical tools discussed in Chapter 5 (Control System Performance) to evaluate how each individual parameter affects the system's overall behavior and, consequently, its performance. Based on this analysis, the controller's parameters are adjusted to ensure all specifications are met.

While this procedure often yields the desired results directly, it frequently requires multiple iterations since certain parameters interact with each other and influence the overall system behavior. For instance, a specific parameter value may satisfy the overshoot requirement, but when another parameter is adjusted to achieve the desired rise time, the overshoot may no longer be acceptable.

It becomes clear that the more specifications there are, the more controller parameters need to be adjusted, and the more complex the design process becomes.

a) General principles

Theoretically, the placement of the controller within the overall system diagram has little impact, as it globally modifies the open-loop transfer function. However, in practice, it cannot be placed arbitrarily.

At the end of the action chain, high-power components are present, making it impractical and uneconomical to position a controller there. On the other hand, the feedback loop carries measurement signals, making it a viable location for a controller, provided it does not significantly alter the output.

In general, cascade controllers are typically placed either at the output of the comparator in the forward chain (before amplification) or within the feedback loop. Controllers are not perfect implementations. Given the desired performance, the actual system design may only approximate the mathematical model within a specific frequency range, and the system's behavior in other domains may differ significantly. This must be considered when integrating the controller into the system.

To determine the type of controller and parameter values, several methods can be used:

- Analyze the time-domain response and evaluate the static and dynamic performance before and after compensation.
- Use the Nyquist curve of the compensated system to deduce the controller structure and parameters by comparison with the desired curve.
- Follow a similar process using Bode plots.
- Use root locus techniques, as the controller introduces new poles and zeros.

In many examples, a simple amplifier with a constant gain KK has been used as the controller. This type of control, known as proportional correction, produces an output u(t)u(t) proportional to the input ε(t)\varepsilon(t).

Additionally, the controller can incorporate the derivative or integral of the input signal ε(t)\varepsilon(t) alongside proportional action. Thus, controllers can generally be viewed as a combination of components such as comparators (adders or subtractors), amplifiers, attenuators, differentiators, and integrators.

To quickly understand the PID controller, each of its actions—Proportional (P), Integral (I), PI (Proportional-Integral), Derivative (D), and PD (Proportional-Derivative)—should be analyzed individually.

b) Proportional action corrector (P)

The relationship between the output u(t) and the error signal ε(t) is:

$$u(t) = K_p \cdot \varepsilon(t)$$

That is:

$$\frac{U(p)}{\varepsilon(p)} = K_p$$

Regardless of the mechanism and the energy source used, the proportional controller is essentially a variable-gain amplifier. Its functional diagram is shown in the figure.

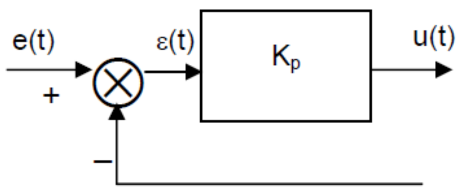

Figure 2.4: Proportional controller

The proportional action **P** generates a control signal **u(t)** that is proportional to the error signal **ε(t)**. It primarily influences the system gain and significantly improves accuracy.

The proportional action:

- Increases the gain, which reduces the steady-state error (improving accuracy), but
- Increases the system bandwidth, which
- Improves system speed, and
- Increases system instability.

The proportional corrector P is generally not used alone. We will see that any corrector has at least the proportional action.

Example

The figure below shows the functional diagram of an example of proportional correction.

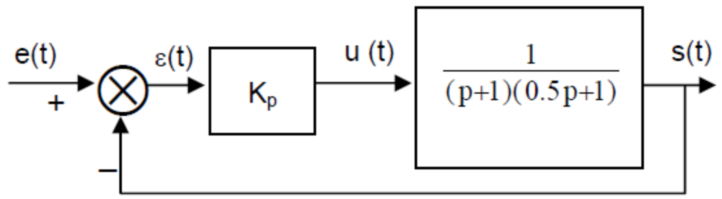

Figure 2.5: Proportional controller in control process

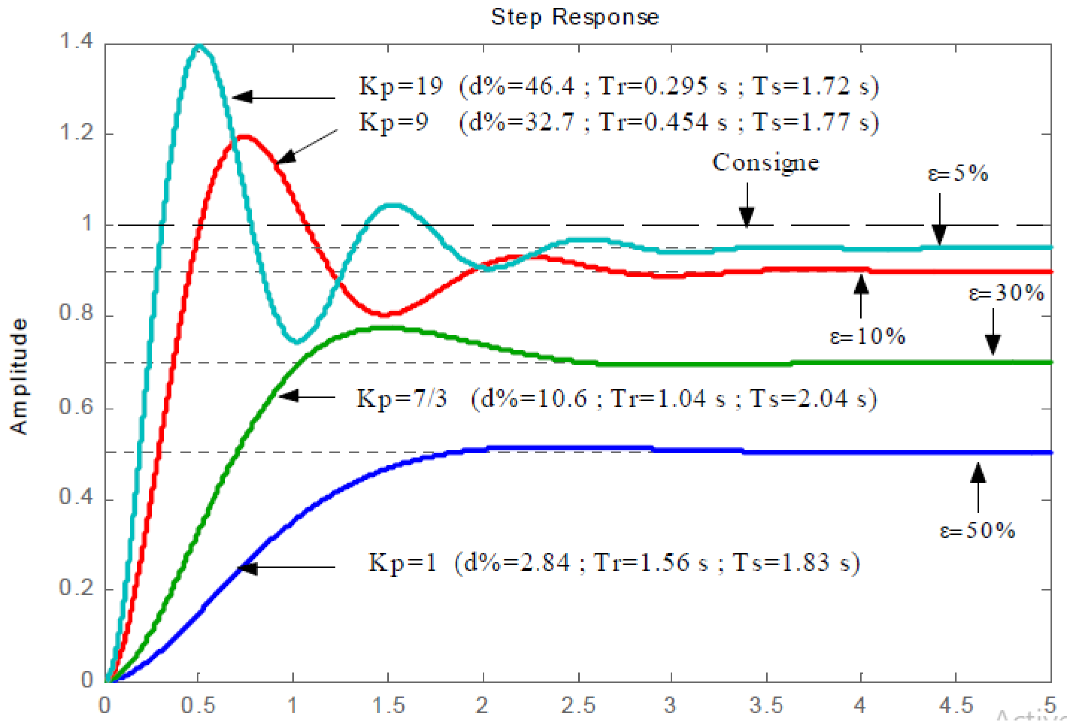

Figure 2.6: Effect of the proportional action

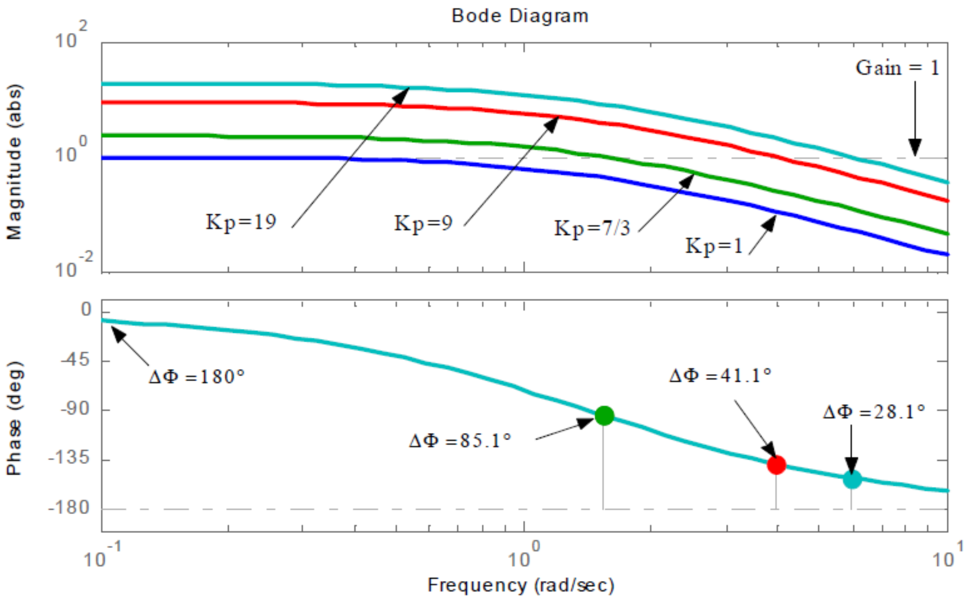

Figure 2.7: Effect of the proportional controller on stability margins

It is observed that increasing Kp leads to:

• An improvement in static error,

- A decrease in rise time,
- A slight improvement in settling time,
- But also a reduction in phase margin and an increase in overshoot (increased system instability).

c) Integral Action Controller (I)

The relationship between the output u(t)u(t)u(t) and the error signal ε(t)\varepsilon(t)ε(t) is:

$\frac{du(t)}{dt} = K_i \varepsilon(t)$

or equivalently,

$u(t) = K_i \int_0^t \varepsilon(t)dt$ That is to say,

$\frac{U(p)}{\varepsilon(p)} = \frac{K_i}{p} = \frac{1}{T_i p}$

where

- K_i is called the **integral gain**
- T_i is called the **integration time constant**

The main advantage of this controller is that it introduces integration into the control loop.

We know that the presence of integration in the open-loop transfer function (OLTF) increases the system's order and reduces or eliminates, depending on the type of input, the system's steady-state error.

The pure integral action:

- Improves accuracy by reducing or eliminating the steady-state error, but
- Introduces a -90° phase shift, which may destabilize the system (reducing the phase margin).

A controller with purely integral action is rarely used in practice due to its slow response and destabilizing effect. It is generally combined with the proportional controller.

d) derivative action corrector (D)

The relationship between the output u(t) and the error signal ε(t) is:

$u(t) = K_d \frac{d\varepsilon(t)}{dt}$

That is to say,

$\frac{U(p)}{\varepsilon(p)} = K_d \cdot p = T_d \cdot p$

Where:

K_d is called "derivative gain",

T_d is called "derivative time constant"

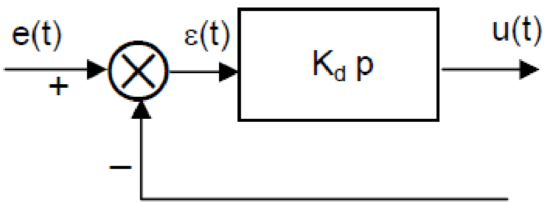

Figure 2.8: Pure Derivative Action Correction

Pure Derivative Action:

- Improves the stability of the system by introducing an additional phase lead of +90° (increasing the phase margin),
- But decreases the system's accuracy,
- And amplifies high-frequency noise.

A controller with exclusively derivative action is almost never used. It is generally combined with a Proportional controller.

e) Proportional, Integral, and Derivative (PID) Controller

The PID corrector combines the actions of the 3 correctors P, I and D.

The relationship between the output u(t)u(t) and the error signal ε(t)ε(t) is:

$$u(t) = K_p \cdot \varepsilon(t) + K_i \int_0^t \varepsilon(t)\,dt + K_d \frac{d\varepsilon(t)}{dt}$$

In the Laplace domain:

$$\frac{U(p)}{\varepsilon(p)} = K_p + \frac{K_i}{p} + K_d\,p$$

Alternatively:

$$\frac{U(p)}{\varepsilon(p)} = \frac{K_p}{p}\left(\frac{K_d}{K_p}p^2 + p + \frac{K_i}{K_p}\right)$$

Or equivalently:

$$\frac{U(p)}{\varepsilon(p)} = K_p\left(1 + T_d\,p + \frac{1}{T_i\,p}\right)$$

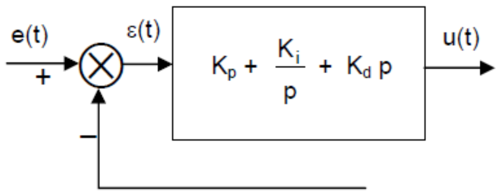

Figure 2.9: Proportional, integral and derivative (PID) correction

This corrector, easy to implement, allows to cancel the static error signal ε∞ and to have a relatively fast and well-damped response. Indeed, the PID corrector increases the system class by one unit and introduces 2 zeros that can be used to improve the transient response.

The root locus method can be used to locate these zeros in order to satisfy a specification on the static and dynamic regimes.

We have seen that a P corrector (Kp) brings speed to the system by reducing the rise time. It also reduces the static error, but does not eliminate it. The integral action (Ki) will have the effect of eliminating the static error. It therefore brings back precision, but degrades the transient response. The derivative action (Kd) improves the stability of the system, reduces overshoots and improves the transient regime.

The effects of each corrector (Kp, Ki and Kd) on the closed-loop response of the system are grouped in Table bellow:

	Rise Time	Overshoot	Settling Time	Steady-State Error
If K_p increases	*Decreases*	*Increases*	*(Little change)*	*Decreases*
If K_i increases	*Decreases*	*Increases*	*Increases*	*Eliminated*
If K_d increases	*(Little change)*	*Decreases*	*Decreases*	*(Little change)*

Figure 2.10: Effect of each action: P I D

There are analytical methods for calculating the components of the PID controller, but they are quite complex and rarely used. Empirical methods exist and greatly simplify the determination of PID controllers (Ziegler-Nichols method, Chien-Hrones method, etc.).

Finally, it is important to remember that it is not mandatory to include all three controllers in a system if it is not necessary. If a PI controller provides satisfactory performance for the output, there is no need to add a D controller to the system. The controller should be designed as simply as possible.

5. Phase delay corrector

The phase lag compensator is a compensator that, despite its name, increases the gain only at low frequencies. It is therefore used to improve the precision of a controlled system.

Its transfer function is:

$C(p) = \frac{a(1+Tp)}{1+aTp} \quad \text{with } a > 1$

To better understand the effect of this compensator, we plot its Bode diagram. There are two cutoff frequencies:

$\frac{1}{T} \quad \text{and} \quad \frac{1}{aT}$

such that :

$\frac{1}{aT} < \frac{1}{T}$

we have

$C(\omega) = \frac{a\sqrt{1+T^2\omega^2}}{\sqrt{1+a^2T^2\omega^2}}$

and

$\varphi(\omega) = \arctan(T\omega) - \arctan(aT\omega)$

When $\omega \to 0\omega \to 0\omega \to 0$, we get:

$C(\omega) \to a$

This zero-slope equivalent is valid from 0 up to the first cutoff frequency, which is expressed as: $\frac{1}{aT}$

The asymptotic slope of the Bode diagram then decreases by one unit, and this new equivalent is valid until the second cutoff frequency $\frac{1}{T}$ beyond which we recover a zero slope.

When$\omega \to +\infty$

we have $20 \log C(\omega) \to 0 \text{ dB}$

Examining the Bode diagram allows us to predict the effect of this compensator. When placed in cascade with the system to be corrected, the two Bode diagrams will add up. The static gain is thus increased by $20 \log a$ which improves precision. By setting the parameter T to a sufficiently small value, this correction only affects low frequencies; the gain at high frequencies is practically unaffected. The additional negative phase shift introduced by the compensator also occurs at low frequencies. Therefore, it does not influence the stability margin, since the 0 dB crossover frequencies are generally located at higher frequency ranges.

In any case, to adjust the phase lag compensator, we choose the value of a to obtain the desired static gain, and then select T such that $\frac{1}{T} \ll \omega_0$

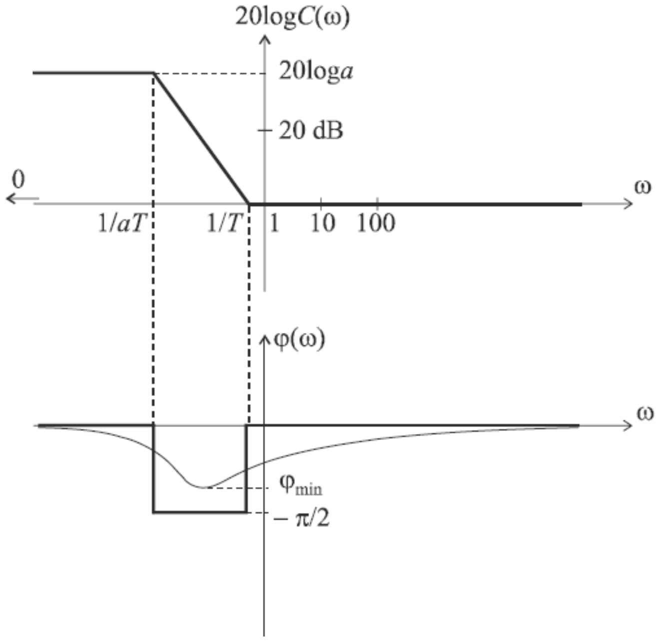

Figure 2.11: Bode plot of a phase delay corrector

Example

Let us consider a system with the transfer function G(p) placed in a unit feedback loop, given by:

$G(p) = \frac{K}{(1+\frac{p}{10})^3}$

The parameter K, the open-loop static gain of the system, is positive and adjustable. We want the closed-loop system to have a position error of $\varepsilon_p = 5\%$ while maintaining a phase margin of $\Delta\varphi = 45°$.

We start by adjusting K to satisfy the phase margin condition:

Since:

$G(j\omega) = \frac{K}{\left(1+\frac{j\omega}{10}\right)^3}$

we have

$\Delta\varphi = \pi - 3\arctan\frac{\omega_{c0}}{10} = \frac{\pi}{4}$

which gives :

$\omega_{c0} = 10 \text{ rad/s}$

thus we get

$G(\omega_{c0}) = \frac{K}{\left(\sqrt{1+\frac{\omega_{c0}^2}{100}}\right)^3} = 1$

solving for K

$K = \left(\sqrt{2}\right)^3 = 2.8 \quad \Rightarrow \quad 20\log K = 8.9 \text{ dB}$

Let us now calculate the position error obtained in closed-loop under these conditions:

$\varepsilon_p = \lim_{p\to 0}[1 - H(p)] = \lim_{p\to 0}\left[1 - \frac{K}{K+(1+\frac{p}{10})^3}\right]$

which gives

$\varepsilon_p = \frac{1}{1+K} = 0.26 = 26\%$

The observed accuracy does not meet the design specifications. To achieve a position error of **5%**, it is necessary to have a static gain K' such that:

$\varepsilon_p = \frac{1}{1+K'} = 0.05 \quad \Rightarrow \quad K' = 19 \quad \Rightarrow \quad 20\log K' = 25.6 \text{ dB}$

Let us introduce a phase lag compensator in the direct path

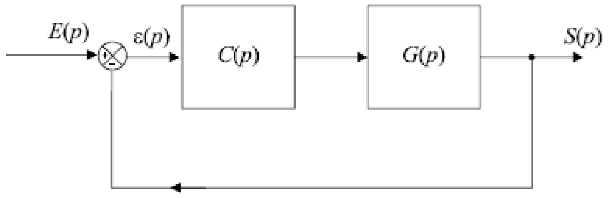

Figure 2.12: Proportional gain control

we have :

$C(p) = \frac{a(1+Tp)}{1+aTp} \quad \text{with } a > 1$

The new open-loop transfer function is:

$G_c(p) = C(p)G(p) = \frac{a(1+Tp)}{1+aTp} \cdot \frac{2.8}{(1+\frac{p}{10})^3} \quad \text{with } a > 1$

The new static gain is:

K' = 2.8a.

Therefore, it is necessary to adjust the parameter a such that:

$a = \frac{19}{2.8} = 6.8 \quad \Rightarrow \quad 20\log a = 16.7 \text{ dB}$

Finally, we just need to choose T so that 1/TT is much lower than the cutoff frequency at 0 dB.

For example, we can take T=10 s.

Thus, we finally get:

$C(p) = \frac{6.8(1+10p)}{1+68p}$

Figure below presents the Bode diagrams comparing the initial system and the corrected system.

Remember that the Bode diagrams of G(p) and C(p) add up to form that of the corrected system $G_c(p)$.

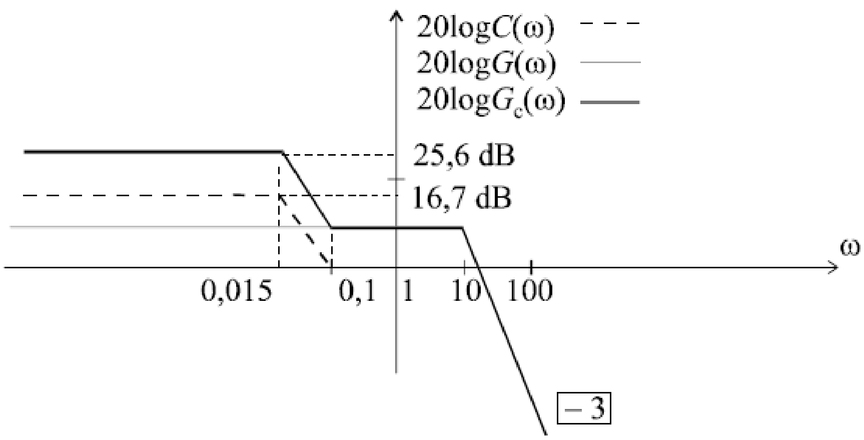

Figure 2.13: Bode margins 1

6. Phase advance corrector

The phase lead compensator is a compensator that, as its name suggests, increases the phase margin of a system. It compensates for insufficient phase shift around the cutoff frequency at 0 dB.

We take:

$$C(p) = \frac{1+aTp}{1+Tp} \quad \text{with } a > 1$$

To better understand the effect of this compensator, let's plot its Bode diagram. There are two cutoff frequencies: 1/T and 1/aT with:

$$\frac{1}{aT} < \frac{1}{T}$$

We have:

$$C(\omega) = \frac{\sqrt{1+a^2T^2\omega^2}}{\sqrt{1+T^2\omega^2}}$$

and

$$\varphi(\omega) = \arctan aT\omega - \arctan T\omega$$

When ω→0, we get:

$$C(\omega) \to 1$$

This zero-slope equivalent is valid from 0 up to the first cutoff frequency, which is expressed as:

$$\frac{1}{aT}$$

At this point, the asymptotic slope of the Bode diagram increases by one unit. This new equivalent remains valid up to the second cutoff frequency (1/T), beyond which the slope returns to zero (Figure below).

When yω→+∞, we get:

$$20 \log C(\omega) \to 20 \log a$$

The advantage of this compensator is visible in its phase diagram: at the frequency

$$\omega_{\max} = \frac{1}{T\sqrt{a}}$$

the phase shift reaches a maximum that we can easily calculate:

$$\varphi_{\max} = \arcsin \frac{a-1}{a+1}$$

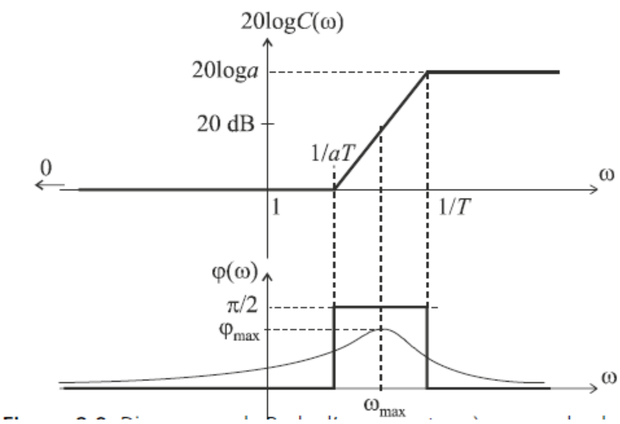

Figure 2.14: Bode margins 2

The principle of corrective action is to align ωmax\omega_{\max}ωmax with the cutoff frequency at 0 dB, $\omega_c 0$, of the system to be corrected, and to adjust $\varphi max \varphi_{\max} \varphi max$, known as the phase boost, to achieve the desired phase margin.

Example

Consider a system with a transfer function G(p)G(p)G(p) placed in a unit feedback loop, given by:

$G(p) = \frac{100}{(p+1)^2}$

We want to correct this system so that its phase margin equals 45°. Let's compute the phase margin before correction.

We have:

$G(\omega) = \frac{100}{1+\omega_0^2} = 1$

from which:

$\omega_0 = \sqrt{99} = 9.95 \text{ rad/s}$

thus

$\Delta\varphi = \pi - 2\arctan\omega_0 = 0.2 \text{ rad} = 11^\circ$

The phase margin is insufficient. To correct it, we need to introduce a phase boost of 34∘ at frequency $\omega_c 0$.

We introduce a phase lead compensator adjusted such that:

$\frac{1}{T\sqrt{a}} = \omega_0 = 9.95 \text{ rad/s} \quad \text{and} \quad \varphi_{\max} = 34^\circ = \arcsin\frac{a-1}{a+1}$

thus:

$\arcsin\frac{a-1}{a+1} = 34^\circ$

which gives:

$a = \frac{1+\sin 34^\circ}{1-\sin 34^\circ} = 3.54$

or equivalently:

$20\log a = 11 \text{ dB}$

then

$\frac{1}{T\sqrt{a}} = \omega_0 \quad \Rightarrow \quad T = \frac{1}{\omega_0\sqrt{a}} = \frac{1}{9.95\sqrt{3.54}} = 0.053 \text{ s}$

so

$T = 18.9 \text{ rad/s} \quad \text{and} \quad aT = 5.3 \text{ rad/s}$

Finally:

$C(p) = \frac{1+0.19p}{1+0.053p}$

The new open-loop transfer function is:

$G_c(p) = C(p)G(p) = \frac{1+0.19p}{1+0.053p} \cdot \frac{100}{(p+1)^2}$

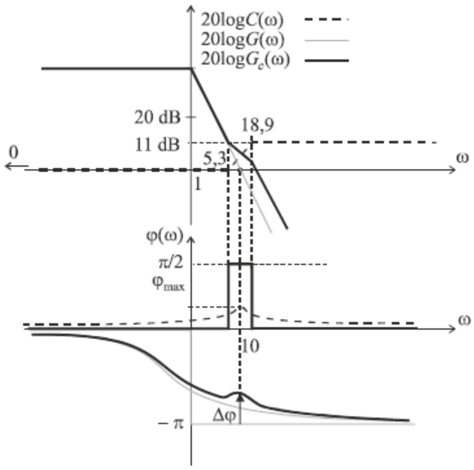

Figure 2.15: Bode margins 3

7. Selection Criteria and Tuning Methods

When designing a control system, selecting the appropriate controller parameters is crucial for ensuring good performance. This process involves:

- Choosing a suitable **performance criterion**
- Applying a **tuning (dimensioning) method** based on system response characteristics

Performance Criteria

Performance criteria help define what makes a controller “good” based on desired behaviors (like fast response, minimal overshoot, etc.).

a. **Flatness Criterion (Critère Méplat)**

This criterion focuses on minimizing oscillations in the system’s step response. A flat response implies low overshoot and a stable approach to the steady-state.

- Preferred in systems where overshoot is undesirable.
- May result in slower response, but higher robustness.

b. **Symmetrical Criterion (Critère Symétrique)**

The symmetrical criterion aims to balance rise time and overshoot for a well-proportioned transient response.

- Useful when both speed and stability are important.
- Often results in a compromise between rapidity and damping.

Controller Tuning Methods

These are practical techniques used to determine the parameters of P, PI, or PID controllers.

a. **Ziegler–Nichols Method**

A widely used empirical method to tune PID controllers based on the system’s ultimate gain and oscillation period.

Steps (Ultimate Gain Method):

1. Set the controller to **P-only** mode.
2. Increase the gain until the output oscillates with constant amplitude.

3. Record:
   - Ku: **Ultimate gain**
   - Tu: **Oscillation period**

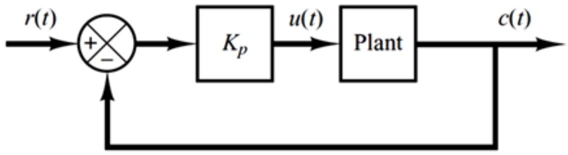

Figure 2.16: proportional gain feedback

Tuning Formulas:

Controller Type	KpK_pKp	TiT_iTi	TdT_dTd
P	0.5Ku0.5 K_u0.5Ku	–	–
PI	0.45Ku0.45 K_u0.45Ku	Tu/1.2T_u / 1.2Tu /1.2	–
PID	0.6Ku0.6 K_u0.6Ku	Tu/2T_u / 2Tu/2	Tu/8T_u / 8Tu/8

8. Controller Tuning by Imposing a Reference Model

1. Objective

This method consists of designing a controller so that the closed-loop system behaves like a desired reference model (also called a tracking model).

The idea is to impose a performance model with known, desirable properties (such as fast response and good stability), and then calculate the controller parameters so that the actual system matches this model.

2. Reference Model Definition

Let the desired closed-loop transfer function (model of tracking) be:

$M(s) = \frac{Y_d(s)}{R(s)} = \frac{b_m}{s^2+a_{m1}s+a_{m0}}$

Where:

- R(s) is the Laplace transform of the input (reference)
- Y_d(s) is the desired output
- M(s) defines the ideal system behavior

3. Example: Second-Order Tracking Model

A common choice for M(s) is a second-order system:

$M(s) = \frac{\omega_n^2}{s^2+2\zeta\omega_n s+\omega_n^2}$

Where:

- \omega_n : natural frequency
- \zeta : damping ratio

This model allows control over:

- Speed of response (ω_n)
- Overshoot and stability (ζ)
- --

4. Designing the Controller

Let the process (plant) be:

$$G(s) = \frac{b}{s^2+a_1s+a_0}$$

We want the closed-loop transfer function:

$$T(s) = \frac{C(s)G(s)}{1+C(s)G(s)} = M(s)$$

Solving for C(s) :

$$C(s) = \frac{M(s)}{G(s)(1-M(s))}$$

5. Implementation Strategy

- Choose a desired model M(s)
- Identify or approximate the system G(s)
- Compute the controller C(s) using the formula above
- Simplify and implement the controller
- --

6. Advantages

Precise control over desired performance

Simple mathematical formulation

Suitable for analog and digital implementation

- --

7. Limitations

Sensitive to modeling errors in G(s)

Can lead to complex (high-order) controllers

Performance affected by disturbances and noise

9. Application: Speed Control of a DC Motor

1. Introduction

DC motors are widely used in industrial systems, robotics, and automation. One of the most important control tasks is to **regulate the speed** of a DC motor to follow a desired reference value, despite disturbances or load variations.

This section introduces the **mathematical model** of a DC motor, and explains how to **design a controller** to regulate its speed.

2. Mathematical Model of the DC Motor

A simplified linear model of a DC motor can be described by the following equations:

Electrical Equation:

$$V(t) = R \cdot i(t) + L \cdot \frac{di(t)}{dt} + K_e \cdot \omega(t)$$

Mechanical Equation:

$J \cdot \frac{d\omega(t)}{dt} + B \cdot \omega(t) = K_t \cdot i(t)$

Where:

- V(t) : input voltage [V]
- i(t) : armature current [A]
- $\omega(t)$: angular speed [rad/s]
- R : armature resistance [Ω]
- L : armature inductance [H]
- K_e : back EMF constant [V·s/rad]
- K_t : torque constant [N·m/A]
- J : moment of inertia [kg·m²]
- B : viscous friction coefficient [N·m·s]

3. Transfer Function

Taking Laplace transforms and assuming zero initial conditions, we can derive the transfer function from input voltage V(s) to output speed \Omega(s) :

$\frac{\Omega(s)}{V(s)} = \frac{K_t}{(JLs^2+(JL)Bs+(K_eK_t+RB)s+RK_eB)}$

For simplification, assuming a dominant time constant and neglecting inductance L , we get a **first-order model**:

$\frac{\Omega(s)}{V(s)} = \frac{K}{\tau s+1}$

Where:

- K : system gain
- \tau : time constant

4. Speed Control Using a PI Controller

A common solution for speed control is to use a **Proportional-Integral (PI)** controller:

$C(s) = K_p + \frac{K_i}{s}$

The closed-loop transfer function becomes:

$T(s) = \frac{C(s)G(s)}{1+C(s)G(s)}$

Where G(s) is the DC motor transfer function.

- --

5. Tuning the PI Controller

Using methods like **Ziegler–Nichols**, **pole placement**, or **model-following**, we tune K_p and K_i to achieve:

- Fast response
- Minimal overshoot
- Good disturbance rejection

Example of pole placement:

If the desired pole is at $s = -\alpha$, then solve for K_p, K_i such that the characteristic equation of the closed-loop system equals:

$s^2 + 2\alpha s + \alpha^2 = 0$

6. Practical Considerations

- **Sensor noise filtering** may be needed for derivative terms.
- Add **current limitation** to protect the motor.
- Use **anti-windup** for integrators to prevent saturation issues.

7. Simulation and Validation

Simulate the system using MATLAB/Simulink to:

- Validate the performance
- Test against step inputs and disturbances
- Compare open-loop vs. closed-loop responses

III Discret time system

1. Signals sampling

In industrial reality, the complexity of the systems, as well as that of the processing to be carried out, often requires the use of digital processing tools: computers, calculators, digital systems of all kinds. Such tools cannot in any case accommodate continuous signals; these must be transformed into sequences of numbers in order to be processed (figure ....). Similarly, these systems deliver, at their output, sequences of digital values, in other words, digital signals.

The operation of digital control is carried out as follows:

1. **Conversion of an analog signal into a digital signal**: This involves converting the information from the analog signal into a digital value so that it can be processed by the controller (ADC block).
2. **Synthesis of a calculation algorithm**: This step involves establishing a digital control law to allow the controlled system to meet the specifications provided by the user (Algorithm block).
3. **Conversion of a digital signal into an analog signal**: In order to control the physical system, this step consists of transforming the digital signal from the controller into an analog control signal present throughout the sampling period (DAC block).

Typically, a digital control loop is presented as follows:

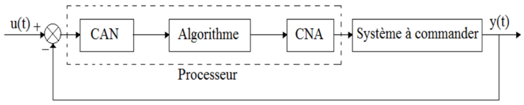

Figure 3.1: discret-continus converters

1.1. The principle of sampling

Sampling a time signal s(t) consists of transforming it into a discrete sequence $s(nT_e)$, with values taken at specific times nT_e.

T_e is called the sampling period, and the times nT_e are referred to as the sampling instants.

In practice, sampling a signal involves multiplying it by a sampling function p(t) , which is zero everywhere except near the times nT_e . This function is often called a "comb" function.

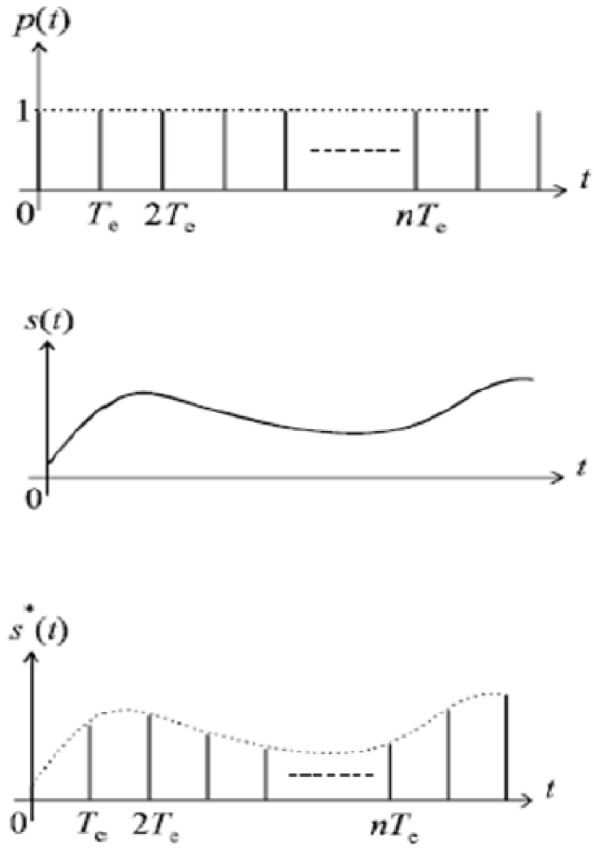

Figure 3.2: sampling steps in Te sampling period

Sampling produces, from a signal s(t)s(t), the sequence

$s(0), s(T_e), s(2T_e), \ldots, s(nT_e)$

which is generally denoted as:

$s^*(t) = \{s_0, s_1, s_2, \ldots, s_n\}$

or alternatively as:

$s(k) = \{s_0, s_1, s_2, \ldots, s_n\}$

1.2. Shannon's Theorem

In order to reconstruct the original signal from the spectrum of the sampled signal, the primary goal of sampling is to ensure no information is lost during the discretization of the continuous signal.

A simple observation of the spectrum \(|U(f)| \) in the figure shows that this is possible if there is no overlap between the different segments of the spectrum.

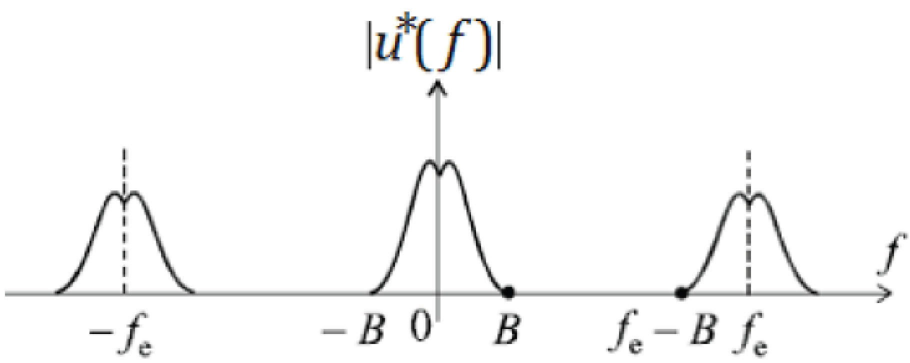

Figure 3.3: signal bandwitdh

If B is the bandwidth of the signal s(t) , in other words, its upper frequency limit, the first shifted segment in the spectrum of s*(t) , which is centered around the frequency f_e, extends from $f_e - B$ to $f_e + B$.

The condition for no overlap is therefore, quite clearly:

B< (f_e-B)=> f_e> 2B

This inequality constitutes Shannon's theorem, which can also be stated as follows:

To preserve the information contained in a signal during sampling, the sampling frequency f_e must be greater than twice the bandwidth of the signal.

> Example
>
> If we want to digitize the analog signal of the telephone network, which has a bandwidth extending from 300 to 3400 Hz, what should be the minimum sampling frequency?
>
> The formula of Shannon's theorem immediately shows us that the sampling frequency must be greater than twice the maximum frequency, which is 6800 Hz. The standard frequency that has been chosen for the digital network is 8 kHz, which satisfies the above conditions.

2. The Z transform -properties and applications

Let us take an analog signal u(t) that we sample at a sampling frequency fe while respecting Shannon's theorem.

We have: *u(k) = u0, u1, u2, . . . , Un* this sequence is the sum of the product of the unit pulses shifted from one another by a period *Te* and the coefficient *uk*.

$$\begin{aligned} u^*(t) &= u_0\delta(t) + u_1\delta(t - T_e) + u_2\delta(t - 2T_e) + \cdots \\ &= \sum_{k=0}^{n} u_k\delta(t - kT_e) \end{aligned}$$

$$u^*(t) = \sum_{k=0}^{n} u_k\delta_k$$

The Laplace transform of the sampled signal u*(t) is:

$u^*(s) = \sum_{k=0}^{n} u_k e^{-skT_e}$

By setting z = exp(sTe), the Z transform of the signal u(t) is defined by

$U(z) = \sum_{k=0}^{n} u_k z^{-k}$

Example

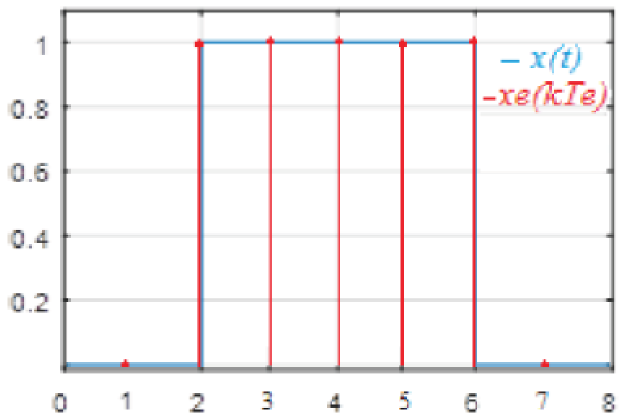

Figure 3.4: square signal sampling

$Y(z) = \sum_{k=0}^{+\infty} x_k z^{-k} = x_0 z^0 + x_1 z^{-1} + x_2 z^{-2} + x_3 z^{-3} + x_4 z^{-4} + x_5 z^{-5} + x_6 z^{-6} = z^{-2} + z^{-3} + z^{-4} + z^{-5} + z^{-6}$

2.1. Properties of the Z transform

- Linearity

$\alpha u_1(t) + \beta u_2(t) \overset{TZ}{\leftrightarrow} \alpha U_1(z) + \beta U_2(z)$

- Time shifts

Delay theorem: The Z transform of the signal u(t) delayed by one time is:

$u(t - aT_e) \overset{\bar{q}}{\longrightarrow} z^{-a} U(z)$

Advance theorem: The Z transform of the signal u(t) advanced by a time aTe, is:

- Derivative with respect to Z

$tu(t) \overset{TZ}{\leftrightarrow} -Tz\frac{dU(z)}{dz}$

- Complex scaling

$x_k = a^k$

$X(z) = U\left(\frac{z}{a}\right)$

- Convolution product

$u(t) * y(t) \overset{Z}{\leftrightarrow} U(z)Y(z)$

- Accumulation

$\sum_{k=0}^{n} u_k \overset{Z}{\leftrightarrow} \frac{z}{z-1} U(z)$

2.2. Z-transform of usual signals

Unit Impulse

$$\begin{cases} \delta_k = 1 & \text{si} \quad k = 0 \\ \delta_k = 0 & \text{ailleurs} \end{cases}$$

Z-transform of Unit Impulse

$\Delta(z) = \sum_{k=0}^{\infty} \delta_k z^{-k} = z^0 = 1$

Unit Step

$$\begin{cases} \Gamma_k = 1 & \forall k \geq 0 \\ \Gamma_k = 0 & \text{ailleurs} \end{cases}$$

Z-transform of Unit Step

$\Gamma(z) = \sum_{k=0}^{\infty} z^{-k} = \sum_{k=0}^{\infty} \left(\frac{1}{z} \right)^k = \frac{1}{1 - \frac{1}{z}} = \frac{z}{z-1}$

Unit Ramp Definition

$$\begin{cases} v(t) = t & \forall t \geq 0 \\ v(t) = 0 & \text{elsewhere} \end{cases}$$

Unit Ramp Expressed with Unit Step

$v(t) = t \cdot \Gamma(t)$

Z-transform of Unit Ramp Using Differentiation

$V(z) = -zTe \frac{d\Gamma(z)}{dz} = -zTe \frac{d}{dz} \left(\frac{z}{z-1} \right)$

% Final Z-transform of Unit Ramp

$V(z) = \frac{zTe}{(z-1)^2}$

Exponential Decreasing

Let *u(t)* be the signal defined by

$$\begin{cases} u(t) = e^{-at}, & \forall t \geq 0 \\ u(t) = 0, & \text{elsewhere} \end{cases}$$

Z-transform of the Signal

$U(z) = \sum_{k=0}^{\infty} e^{-akT_e} z^{-k}$

Simplification of the Z-transform

$= \sum_{k=0}^{\infty} \left(\frac{1}{ze^{aT_e}} \right)^k$

Final Z-transform

$U(z) = \frac{z}{z-e^{-aT_e}}$

2.3. Initial Value and Final Value Theorem of a Sampled System

initial Value Theorem

Let $U(Z)$ be the Z-transform of the signal $u(t)$. The sampling of the signal u(t) gives us the sequence u_k Initial Value Theorem

The initial value u(0) can be evaluated directly from its Z-transform according to the formula:

$u(0) = \lim_{z \to +\infty} U(z)$

finall Value Theorem

The final value theorem allows us to know the value towards which the sequence uk tends when k tends to infinity

$\lim_{k \to +\infty} u_k = \lim_{z \to 1} \left[(1 - z^{-1}) U(z) \right] = \lim_{z \to 1} \left[\frac{z-1}{z} U(z) \right]$

2.4. Inverse Z Transform

The discrete-time Z-transform plays the same role as the Laplace transform in the continuous domain.

Using the Z-transform requires us to be familiar with methods for finding the inverse Z-transform.

The inverse Z-transform of a signal U(z) provides us with the time sequence corresponding to *u(kT)*.

It is very important to note that the inverse Z-transform only provides the sequence *u(kT)* at sampling times *0, T, 2T, . . . , kT*.

This sequence *u(kT)* is unique. However, the signal *u(t)* may not be unique.

The following figure shows an example for which we have a sequence *u(kT)* that is the same for two different signals *u1(t)* and *u2(t)*.

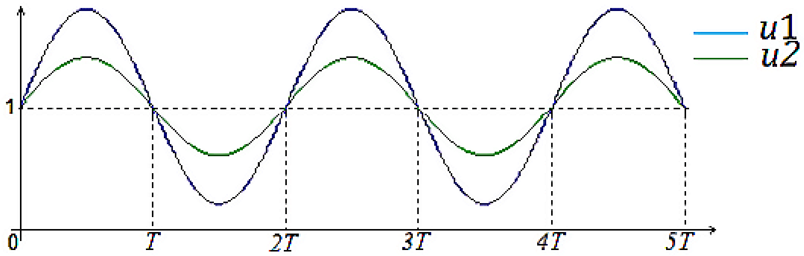

Figure 3.5: sinusoidal signal sampling

There are four methods of inverting the Z transform. Two analytical methods allow us to obtain the original signal as a function of time t, the other two methods are of the digital type which only give the digital values at sampling times of the signal u(t).

a) Analytical methods

Residue Method

let

$u(z) = \sum_{n=0}^{\infty} u(nT) z^{-n}$

with

$u(nT) = \sum_{z_i} \left[\text{Residue}_{z=z_i} \left(z^{(n-1)} U(z)\right)\right]$

where z_i are the poles of *U(z)*

Example

Let a signal U(z) be given by:

$U(z) = \frac{Tz^n}{(z-1)^2}$

This function has a double pole at z=1.

$(nT) = \left[\text{Residue at } z = z_i \left(z^{(n-1)} \frac{Tz}{(z-1)^2}\right)\right] = \frac{d}{dz}\left[(z-1)^2 \frac{Tz^n}{(z-1)^2}\right]\Big|_{z=1} = nT$

Therefore:

$U(z) = T\sum_{n=0}^{\infty} nz^{-n}, \quad \text{and} \quad U(t) = t \cdot \Gamma(t)$

b) Method of decomposition into simple elements

This method consists of decomposing U(z) into simple elements whose originals are found in the Z transform table.

Example:

Let a signal U(z) be given by the following equation:

$U(z) = \frac{z(5z-3.6)}{z^2-1.4z+0.48}$

This function has two poles: z=0.8z = 0.6

The simple element decomposition of U(z)gives us:

$U(z) = \frac{2z}{z-0.8} + \frac{3z}{z-0.6}$

Using the Z transform tables we find:

$u(t) = 2e^{\frac{-0.223}{T}t} + 3e^{\frac{-0.511}{T}t}$

Numerical methods

Division by increasing powers of z−1

This method is used when U(z) is in the form of rational fractions in z (or in z−1), it is sufficient to divide

the numerator by the denominator to obtain a series in z−1, whose coefficients are the desired values

of u(nT).

Example

Let a sampled signal have the following Z-transform:

$U(z) = \frac{z}{z^2-3z+2}$

The division of the polynomial numerator by that of the denominator leads to:

$U(z) = z^{-1} + 3z^{-2} + 7z^{-3} + 15z^{-4} + 31z^{-5} + \ldots$

$u^*(t) = \delta(t-T) + 3\delta(t-2T) + 7\delta(t-3T) + 15\delta(t-4T) + 31\delta(t-5T) + \ldots$

therefore

$u(kT) = 2^k - 1$

Method of recurrent equations

This method consists of deducing the value of the sample u(nT) from the knowledge of the previous samples at the times: (n − 1)T, (n − 2)T, (n − 3)T, · · ·

We proceed in this way iteratively by progressing step by step from period to period.

Example

Let a system have the following discrete transfer function:

$\frac{Y(z)}{U(z)} = \frac{1}{z-0.5} \Rightarrow Y(z) - 0.5z^{-1}Y(z) = z^{-1}U(z$

we then find:

$y(kT) = 0.5y[(k-1)T] + u[(k-1)T]$

3. Sampled transfert functions

Recurrence equations

We call a linear recurrence equation of order 1 with constant coefficients any equation of the type:

$u_{k+1} + \alpha u_k = f(k)$

With$\alpha \in R * \alpha \in \mathbb{R}^* \alpha \in R*$ and f a function with values in R\mathbb{R}R.

The solution to this equation is the real sequence$u(k)u(k)u(k)withk \in Nk \in \mathbb{N}k \in N$, whose terms uku_kuk satisfy the equality for all indices k.

In the general case, a recurrence equation of order n is of the form:

$\sum_{k=0}^{n} a_k y(n-k) = \sum_{k=0}^{m} b_k u(m-k)$

The development of this equation gives us:

$a_0 y(n) + a_1 y(n-1) + \cdots + a_n y(0) = b_0 u(m) + b_1 u(m-1) + \cdots + b_m u(0)$

In control systems, a sampled system of order n, governed by a recurrence equation of the form above, where y represents the output and u represents the input, must satisfy the condition for causality.

For this system to be causal, it is required that n≥m.

Sampled transfer function

Applying the Z transform to the recurrent equation leads us to find the sampled transfer function.

Assuming that the initial conditions are zero, the Z transform of equation in the general form above gives us:

$(a_n z^n + a_{n-1} z^{n-1} + \cdots + a_1 z + a_0) Y(z) = (b_m z^m + b_{m-1} z^{m-1} + \cdots + b_1 z + b_0) U(z)$

and

$G(z) = \frac{b_m z^m + b_{m-1} z^{m-1} + \cdots + b_1 z + b_0}{a_n z^n + a_{n-1} z^{n-1} + \cdots + a_1 z + a_0}$

This is defined as the Z-domain transfer function representing a sampled system.

The transfer function can also be represented by the following model:

$G(z) = \frac{b_m (z-z_1)(z-z_2)\cdots(z-z_m)}{a_n (z-s_1)(z-s_2)\cdots(z-s_n)}$

Where

$z_i, i \in [1, m]$ represents the zeros of the system.

$s_i, i \in [1, n]$ represents the poles of the system.

Example

Let a discrete system be represented by the following recurrence equation:

$y(k) = y(k-1) - ay(k-1) + ae(k-1)$

We assume that the input of the system is eee and its output is yyy. The Z transform of this equation is given by:

$Y(z) = z^{-1} Y(z) - az^{-1} Y(z) + az^{-1} E(z) \Rightarrow [1 + (a-1)z^{-1}] Y(z) = az^{-1} E(z)$

So the transfer function of this system is given by:

$G(z) = \frac{Y(z)}{E(z)} = \frac{az^{-1}}{1+(a-1)z^{-1}} \equiv \frac{a}{z+(a-1)}.$

4. Association of sampled systems

Simplification Rules

a) **Rule 1:** Elements in cascade or serie

$$G = G_1 \cdot G_2 \cdot \ldots \cdot G_n = \prod_{i=1}^{n} G_i$$

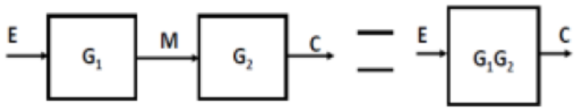

Figure 3.7: cascade

b) **Rule 2:** Elements in parallel

$$G = G_1 + G_2 + \ldots + G_n = \sum_{i=1}^{n} G_i$$

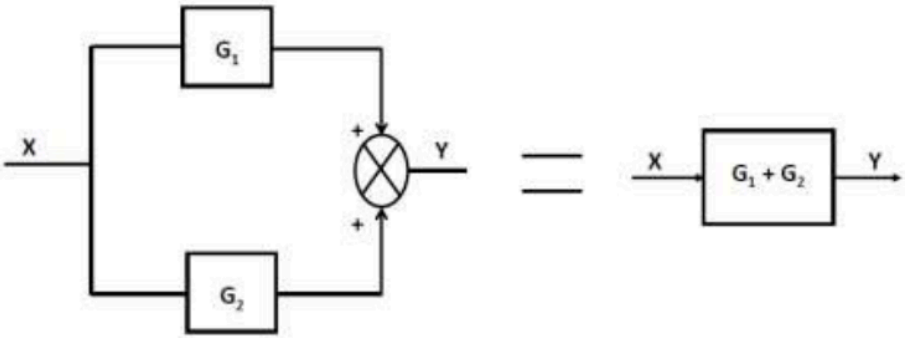

Figure 3.8: parallel representation

c) **Rule 3:** Removal of an element from an action chain

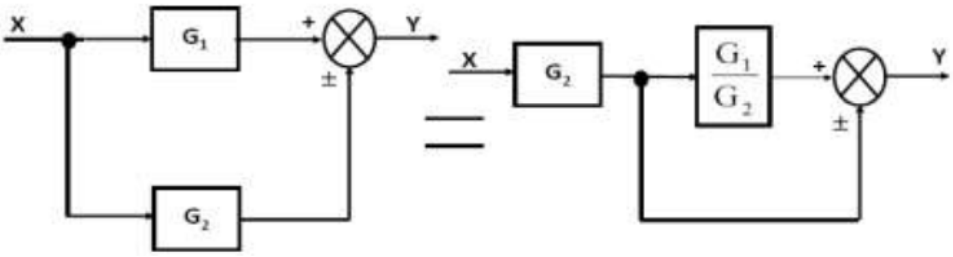

Figure 3.9: summing action

d) **Rule 4:** Elimination of a feedback loop

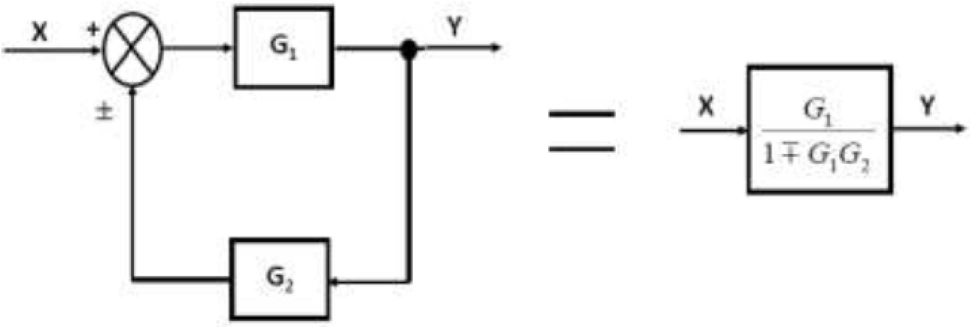

Figure 3.10: feedback loop

There are other rules that we covered in Asservis -1, which apply equally to both continuous and discrete-time systems.

Example

Determine the transfer function of the following system.

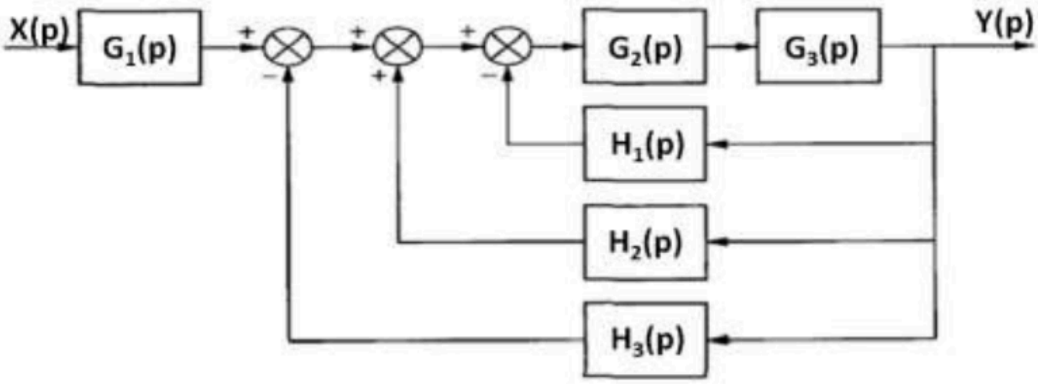

Figure 3.11: complex control schema 1

Solution

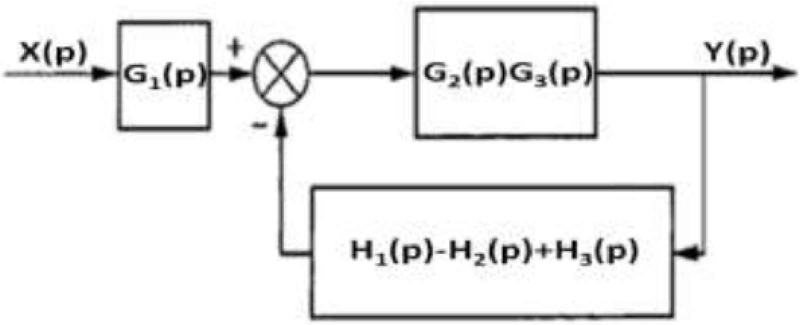

Figure 3.12: complex control schema 2

5. Harmonic, impulse, and step response

A discrete-time system can be characterized by its impulse response s(k) and step response u(k), as shown in Figure below. The impulse response corresponds to the system's output when the input is a discrete impulse δ(k). In the case of the step response, it is obtained when the input is a discrete unit step u(k).

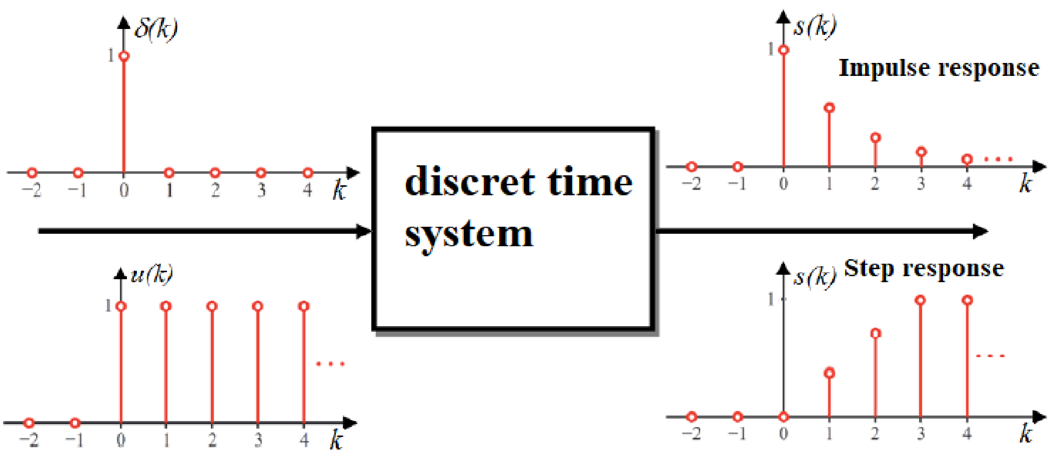

Figure 3.13: complex control schema 3

Example

Consider the transfer function of the system F(z) given by:

$F(z) = \frac{1}{z^2-3z+2}$

since

$F(z) = \frac{S(z)}{E(z)} \Rightarrow S(z) = E(z) \cdot \frac{1}{z^2-3z+2}$

We apply an impulse input e(k)=δ(k).

The Z-transform of the impulse signal δ(k) is E(z)=1.

Thus:

$S(z) = E(z) \cdot \frac{1}{z^2-3z+2} \Rightarrow S(z) = \frac{1}{z^2-3z+2}$

The partial fraction decomposition of S(z)/z is:

$S(z) = \frac{1}{z(z^2-3z+2)} = \frac{\frac{1}{2}}{z} + \frac{\frac{1}{2}}{z-2} - \frac{1}{z-1}$

Thus, we obtain:

$S(z) = \frac{1}{2z} + \frac{z}{2(z-2)} - \frac{z}{z-1}$

The inverse Z-transform is obtained directly using the transform pairs from Table

$s(k) = \mathcal{Z}^{-1}\left\{\frac{1}{2z} + \frac{z}{2(z-2)} - \frac{z}{z-1}\right\}$

Applying inverse transformation:

$s(k) = \frac{1}{2}\delta_k + \frac{1}{2}2^k - 1^k = \frac{1}{2}\delta_k + 2^{k-1} - 1$

Thus, we get:

$$s(0) = \frac{1}{2} + 2^{-1} - 1 = 0$$
$$s(1) = 2^0 - 1 = 0$$
$$s(2) = 2^1 - 1 = 1$$
$$s(3) = 2^2 - 1 = 3$$
$$s(4) = 2^3 - 1 = 7$$
$$\vdots$$

6. Z-domain Transmittance and Frequency Response of a Zero-Order Hold

1. Transmittance in Z of the Zero-Order Hold

The **Zero-Order Hold (ZOH)** is a device used to maintain a constant value between two successive samples in a discrete-time system. It is often used in digital-to-analog conversion.

Its transfer function in the sss-domain is given by:

$H(s) = \frac{1-e^{-Ts}}{s}$

where T is the sampling period.

Applying the Z-transform,the ZOH transfer function in the Z-domain is:

$H(z) = \frac{1-z^{-1}}{s}$

which simplifies to:

$H(z) = \frac{1-z^{-1}}{T}$

2. Frequency Response

The frequency response is obtained by evaluating H(z) on the unit circle, i.e., replacing z with $e^{j\omega T}$:

$H(e^{j\omega T}) = \frac{1-e^{-j\omega T}}{j\omega}$

Simplifying:

$H(e^{j\omega T}) = \frac{1-e^{-j\omega T}}{j\omega}$

This function shows that the ZOH acts as a low-pass filter, smoothing out rapid variations in the signal. Its magnitude and phase are given by

$|H(e^{j\omega T})| = \left|\frac{\sin(\omega T/2)}{\omega T/2}\right|$

$\mathrm{Arg}(H(e^{j\omega T})) = -\frac{\omega T}{2}$

This means that the ZOH introduces attenuation at high frequencies and a phase delay that increases with frequency.

7. Analysis of sampled systems, sampled stability

The study of the stability of a closed-loop system consists in locating the poles of the latter in the complex plane.

In the case of a continuous system, the system is stable if and only if the poles of the latter are in the left half-plane.

For discrete systems, we will begin the study of stability by the relationship between the S plane and the Z plane in order to locate the stability domain in the Z plane.

Let:

$Z = e^{sT} \text{ with } s = \sigma \pm j\omega \Rightarrow Z = e^{\sigma T} e^{\pm j\omega T}$

Then Z can be written as:

$Z = e^{\sigma T} e^{j(\omega T + 2\pi k)}$

And thus:

$|Z| = e^{\sigma T}$

The relationship between the stability region in continuous and discrete systems is given by:

$\sigma < 0 \Rightarrow |Z| < 1$

Calculating poles can be complex for high-order systems, so algebraic methods such as the Routh criterion or De Jury criterion are often used.

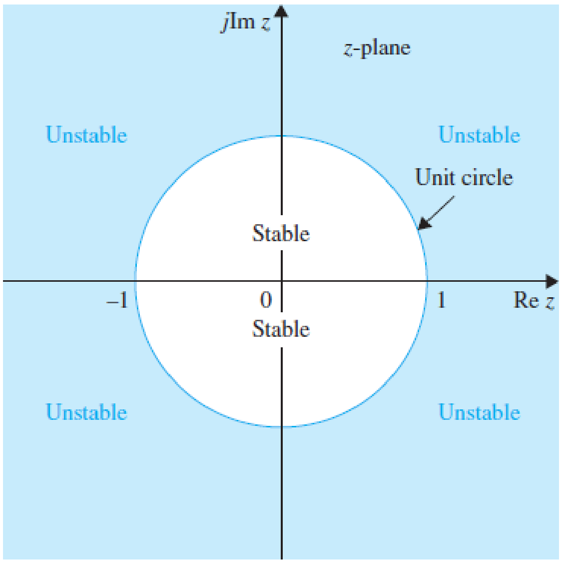

Figure 3.6: complex plane for poles stability

IV Synthesis of sampled control systems

Introduction

The synthesis of **sampled control systems** plays a crucial role in modern digital control, where signals are processed in discrete-time rather than continuous-time. With the increasing use of microcontrollers and digital processors, the need for effective digital control strategies has become essential.

This chapter focuses on the fundamental concepts and methods for designing sampled control systems, emphasizing stability, speed, and steady-state accuracy. It explores various control techniques, including **standard PID controllers**, **digital controllers**, and **pseudo-frequency synthesis with bilinear transformation**. Additionally, it covers the **selection and tuning of regulators** using **classical, modern, and empirical methods** to ensure optimal system performance.

By understanding these techniques, engineers can design efficient digital control systems that meet the performance requirements of various industrial and technological applications.

1. Stability, Speed (Rapidity), Static Precision

Key performance aspects to evaluate and design control systems:

- **Stability**: Ensuring the output remains bounded for any bounded input.
- **Speed (Rapidity)**: How fast the system responds to input changes.

2. Precision

A system is robust if it is characterized by the following three qualities: stability, speed and precision.

In this part we will study the precision of a digital servo system.

The looped systems considered here are assumed to be stable.

Precision can be defined by the ability of a system to follow a given particular instruction with a certain error that must be limited by the specifications.

In this part we are interested in the so-called static precision which is defined by: the error limit after a sufficiently long time (greater than the free response time).

To study static precision, we consider the servo system represented by the following block diagram:

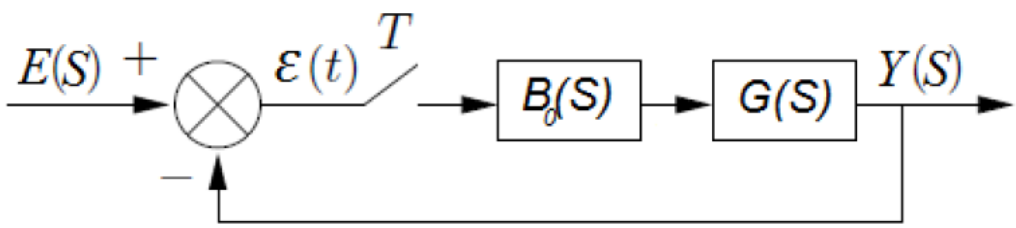

Figure 4.3: sampling system

The Z-transform of the error($\varepsilon(t)$ is given by:

$\varepsilon(z) = \frac{1}{1+T(z)} E(z) \quad \text{where } T(z) \text{ is the Z-transform of } \ [B_0(s)G(s)].$

We consider the system to be stable, so the final value theorem gives us:

$\lim_{k\to+\infty} \varepsilon(kT) = \lim_{z\to1} \left(1 - z^{-1}\right)\varepsilon(z) \Rightarrow \lim_{k\to+\infty} \varepsilon(kT) = \lim_{z\to1} \left(\frac{z-1}{z}\right) \left(\frac{1}{1+T(z)}\right) E(z)$

We clearly see that the steady-state error depends on the input $E(z)$.

The transfer function $G(s)$is of the form:

$G(s) = \frac{k}{s^c}\frac{A(s)}{B(s)} = \frac{k}{s^c}G_p(s)$.

where$c$is the number of integrators in $G(s)$, indicating the system's order.

(T(z) can have the following form:

$T(z) = (1 - z^{-1})Z\left[\frac{k}{s^{c+1}}G_p(s)\right]$.

According to the input E(z) , the type of steady-state error can be defined as follows:

- If the input is a unit step: this is called position steady-state error.
- If the input is a ramp: this is called velocity steady-state error.
- If the input is a parabola: this is called acceleration steady-state error.

The value of the steady-state error can be summarized in the following table:

	c = 0	c = 1	c = 2	c<2
Position Steady-State Error $varepsilon_p$	$\frac{1}{1+k}$	0	0	0
Velocity Steady-State Error $varepsilon_v$	∞	$\frac{T}{k}$	0	0
Acceleration Steady-State Error $varepsilon_a$	∞	∞	$\frac{T^2}{k}$	0

where c: the number of integrators in G(z).

Example Suppose that $G(s) = ks + 1G(s) = \frac{k}{s+1}G(s) = s + 1k$ and we want a position steady-state error less than or equal to 0.05.

$G(s) = \frac{k}{s+1} \Rightarrow T(z) = (1 - z^{-1})\left[\frac{k}{s(s+1)}\right](z)$

where:

$\left[\frac{k}{s(s+1)}\right](z)$

is the Z-transform of

$\frac{k}{s(s+1)}$

thus :

$T(z) = k(1 - z^{-1})\frac{(1-e^{-1})z^{-1}}{(1-z^{-1})(1-e^{-1}z^{-1})} = k\frac{(1-e^{-1})}{z-e^{-1}}$

The steady-state error is given by:

$\varepsilon(z) = \frac{z-e^{-1}}{(z-e^{-1})+k(1-e^{-1})}E(z) \Rightarrow \varepsilon_p = \lim_{z\to 1}\frac{(z-1)}{z}\frac{(z-e^{-1})}{(z-e^{-1})+k(1-e^{-1})}\frac{z}{z-1}$

which simplifies to:

$= \frac{(1-e^{-1})}{1+k}$

To ensure εp≤0.05

$\frac{1}{1+k} \leq 0.05$

Therefore, kkk must be greater than or equal to 19.

3. Standard Controllers (PID) and P-Plane Design

- **PID Controllers**: Widely used due to simplicity and robustness. The transfer function is:

$C(s) = K_p + \frac{K_i}{s} + K_d s$

- **Design in the P-Plane (Pole Placement)**:

Place the closed-loop poles to achieve the desired dynamics.

For a discrete-time system with sampling period T_s design in the Z-plane via mapping:

z = e^{sT_s}

4. Numerical controller

Polynomial methods are among the most commonly used techniques for designing digital controllers. They are highly flexible and relatively easy to implement.

The polynomial RST regulator is described by the canonical structure shown in Figure below), where $R(z^{-1})$, $S(z^{-1})$, and $T(z^{-1})$are polynomials. Such a regulator is called "3-element" (referring to these three polynomials) or "2 degrees of freedom." It can be used for both stable and unstable systems.

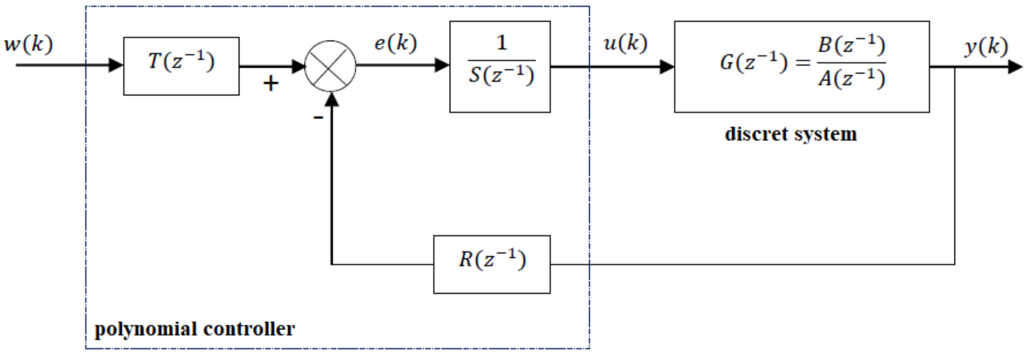

Figure 4.1: polynomial control schema

The transfer function G(z^{-1}) of the process to be controlled can be expressed as follows:

$G(z^{-1}) = \frac{B(z^{-1})}{A(z^{-1})}$

where

$A(z^{-1}) = 1 + a_1 z^{-1} + a_2 z^{-2} + \cdots + a_n z^{-n}$

$B(z^{-1}) = b_1 z^{-1} + b_2 z^{-2} + \cdots + b_m z^{-m}$

The function $G(z^{-1})$ is assumed to be strictly proper, meaning:

$\deg A > \deg B$

The closed-loop transfer function is given as follows:

$H_{BF}(z^{-1}) = \frac{T(z^{-1})B(z^{-1})}{A(z^{-1})S(z^{-1}) + z^{-d}B(z^{-1})R(z^{-1})}$

The characteristic polynomial of $H_{BF}(z^{-1})$ is defined as:

$P(z^{-1}) = A(z^{-1})S(z^{-1}) + B(z^{-1})R(z^{-1})$

Thus, the closed-loop transfer function simplifies to:

$H_{BF}(z^{-1}) = \frac{T(z^{-1})B(z^{-1})}{P(z^{-1})}$

where

$R(z^{-1}) = r_0 + r_1 z^{-1} + \dots$

$S(z^{-1}) = s_0 + s_1 z^{-1} + \dots$

$T(z^{-1}) = t_0 + t_1 z^{-1} + \dots$

The synthesis of the **RST polynomial regulator** is based on the **pole placement strategy** in the closed-loop system.

Pole placement means specifying the closed-loop poles, which correspond to the **roots of the polynomial** $P(z^{-1})$:

$P(z^{-1}) = A(z^{-1})S(z^{-1}) + B(z^{-1})R(z^{-1})$

$P(z^{-1}) = 1 + p_1 z^{-1} + p_2 z^{-2} + \dots$

The choice of the denominator $P(z^{-1})$ of the closed-loop transfer function allows imposing the system's poles.

The polynomials$R(z^{-1})$ and $S(z^{-1})$ must be selected with degrees compatible with the desired degree of P(z^{-1}).

To ensure a **proper regulator**, we impose the following conditions:

$\deg P = 2\deg A - 1, \quad \deg S = \deg R = \deg A - 1$

The resolution of equation above, known as **Bezout's identity** or a **Diophantine equation**, provides the polynomials$R(z^{-1})$ and $S(z^{-1})$by solving the following matrix system.

$S\phi = P$

$\Longleftrightarrow$

$$\begin{bmatrix} a_0 & 0 & 0 & 0 & 0 & \dots & 0 & 0 \\ a_1 & a_0 & 0 & 0 & 0 & \dots & 0 & b_1 \\ \vdots & a_1 & a_0 & 0 & 0 & \dots & b_2 & b_1 \\ a_{n-1} & a_{n-2} & a_{n-3} & \dots & a_0 & b_n & b_{n-1} & \dots \\ a_n & a_{n-1} & a_{n-2} & \dots & a_1 & b_n & b_{n-1} & \dots \\ 0 & a_n & a_{n-1} & \dots & a_2 & b_n & b_{n-1} & \dots \\ 0 & 0 & a_n & \dots & a_3 & b_n & b_{n-1} & \dots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & a_n & b_n & b_{n-1} & b_1 \end{bmatrix} \begin{bmatrix} s_0 \\ \vdots \\ s_{r-1} \\ r_0 \\ \vdots \\ r_{r-1} \end{bmatrix} = \begin{bmatrix} p_0 \\ \vdots \\ p_{r-1} \\ p_n \\ \vdots \\ p_{2n-1} \end{bmatrix}$$

With S being the Sylvester matrix, ϕ representing the vector defining the tuning parameters, and P the vector defining the coefficients of the imposed polynomial.

In fact, in this system, r=max(n,m), a0=1, s0=1, and $P_0 = 1$. For the Sylvester matrix to be invertible, it is necessary and sufficient that the two polynomials $B(z^{-1})$ and $A(z^{-1})$are coprime.

It follows that:

$S\phi = P \Leftrightarrow \phi = S^{-1}P$

On the other hand, the polynomial $T(z^{-1})$ can be chosen as follows:

$T(z^{-1}) = k_B P(z^{-1}$

where

$k_B = \frac{1}{B(1)} \text{ if } B(1) \neq 0, \quad \text{and} \quad k_B = 1 \text{ if } B(1) = 0.$

Example: Designing an RST Controller for a Discrete-Time System

Given:

A discrete-time system with the following transfer function:

$G(z^{-1}) = \frac{0.5z^{-1}}{1-0.8z^{-1}}$

Choose a desired closed-loop characteristic polynomial P(z^{-1}) that places the poles at desired locations for stability and performance. For instance, selecting a pole at z=0.5:

$P(z^{-1}) = 1 - 0.5z^{-1}$

solution

$A(z^{-1}) = 1 - 0.8z^{-1}$

$B(z^{-1}) = 0.5z^{-1}$

Formulate the Diophantine Equation:

The Diophantine equation relates the polynomials A, B, R, and S

$A(z^{-1})S(z^{-1}) + z^{-d}B(z^{-1})R(z^{-1}) = P(z^{-1})$

Assuming a system delay d=1:

$(1 - 0.8z^{-1})S(z^{-1}) + z^{-1}(0.5z^{-1})R(z^{-1}) = 1 - 0.5z^{-1}$

Determine the Degrees of R and S:

To match the degrees on both sides, choose:

$\deg(S) = 0 \quad \text{and} \quad \deg(R) = 0$

let

$S(z^{-1}) = s_0$

$R(z^{-1}) = r_0$

Solve for R and S:

Substitute $S(z^{-1})$ and $R(z^{-1})$ into the Diophantine equation:

$(1 - 0.8z^{-1})s_0 + z^{-1}(0.5z^{-1})r_0 = 1 - 0.5z^{-1}$

Equate coefficients of corresponding powers of z^{-1}:

Forz^0:

$s_0 = 1$

$-0.8s_0 + 0.5r_0 = -0.5$

Substitute s0=1:

$-0.8(1) + 0.5r_0 = -0.5$

$0.5r_0 = 0.3$

$r_0 = 0.6$

thus

$S(z^{-1}) = 1$

$R(z^{-1}) = 0.6$

Determine the T Polynomial:

The T polynomial is typically chosen to ensure proper **reference tracking**. A common choice is:

$T(z^{-1}) = k_B P(z^{-1})$

Where k_B is a gain factor. Assuming $k_B = 1$:

$T(z^{-1}) = 1 - 0.5z^{-1}$

The designed RST controller has the following polynomials:

$R(z^{-1}) = 0.6$

$S(z^{-1}) = 1$

$T(z^{-1}) = 1 - 0.5z^{-1}$

5. Pseudo-Frequency Synthesis & Bilinear Transformation

This sub-chapter focuses on how to design digital controllers using frequency-domain methods, particularly by adapting continuous-time designs to discrete-time systems using **bilinear (Tustin) transformation**.

When designing control systems digitally, we often begin with a continuous-time (analog) model and transform it into a discrete-time equivalent. The **frequency response** of digital systems is not linear with respect to the s-plane due to the nonlinear mapping $s \leftrightarrow z$. To handle this, **pseudo-frequency synthesis** methods are used.

Due to the nonlinear transformation from the s -domain to the z -domain, **frequency warping** occurs. The true frequency ω in the analog domain is not directly equivalent to the frequency Ω in the digital domain.

The transformation is defined as:

$s = \frac{2}{T_s} \cdot \frac{z-1}{z+1}$

This is the **bilinear (Tustin) transformation**, which maps:

- The **left half** of the s-plane into the **inside** of the unit circle in the z-plane.
- The **imaginary axis** (jω) to the **unit circle** (|z| = 1).

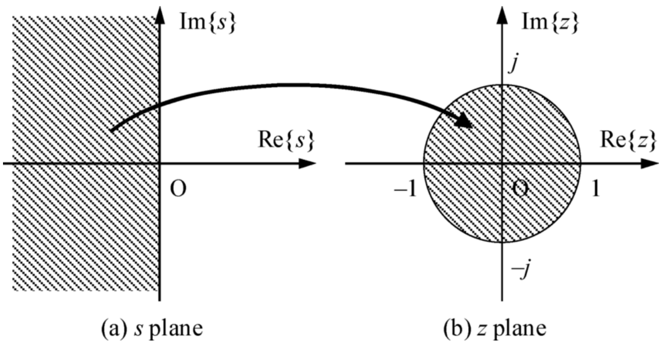

Figure 4.4: stability

To analyze the system as if it were still in continuous-time, we define a **pseudo-frequency** ω_p as follows:

$\omega_p = \frac{2}{T_s} \cdot \tan\left(\frac{\Omega T_s}{2}\right)$

Where:

- Ω is the digital (discrete-time) frequency
- $T_s$is the sampling period

This transformation allows us to apply **Bode plot design** techniques on the digital system by plotting against ω_p instead of Ω .

5.1. Discretization of a continuous transfer

The discretization of a continuous transfer consists of performing a transformation P -> Z.

This transformation is done using an approximation of the relation: Z = e^{sT} .

Several approximation methods allow the transition from the continuous domain to the discrete domain.

The choice of one or the other of these methods depends on the advantages and disadvantages presented by each of these methods, as well as the validity of the results in terms of time and frequency response. Two categories of methods are presented:

— By approximations of the integral operator.

— By invariant techniques.

Approximation of the Integral Operator

This approximation includes the following methods

- **Euler's Method** for which $s = (\frac{z-1}{Tz})$
- **Tustin's Method** .$s = (\frac{2(z-1)}{T(z+1)})$

Example

Consider the following continuous-time transfer function in the s-domain

$(H(s) = \frac{2s+5}{s^2+3s+4}$

Using the Tustin then Euler transformation, find the discrete-time transfer function H(z)

5.2. Relationship between continuous systems and sampled systems

We consider the controlled system represented by the following block diagram:

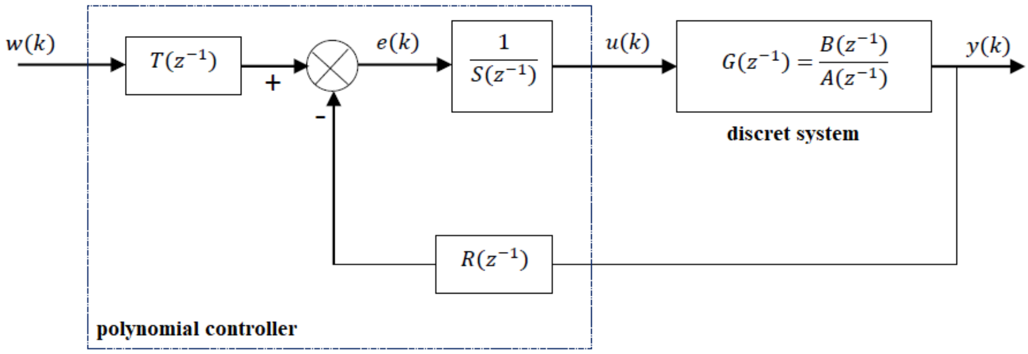

Figure 4.1: polynomial control schema

This is a controlled system with unitary feedback and a single sampler.

For this type of system we have:

$(\varepsilon(t) = e(t) - y(t)$

$\Rightarrow \varepsilon^(t) - y^*(t)$

Let:

$(T(s) = B_0(s)G(s) \Rightarrow T(z) = \text{Z-transform of } [B_0(s)G(s)]$

We have:

$$\begin{cases} \varepsilon(z) = E(z) - Y(z) \\ Y(z) = T(z) \cdot \varepsilon(z) \end{cases}$$

Therefore, the transfer function of the closed-loop system is given by:

$$\frac{Y(z)}{E(z)} = \frac{T(z)}{1+T(z)}$$

6. Controller Design in Discrete Time: Classical, Modern, and Empirical Methods

This section addresses how digital controllers are designed using classical, modern, and empirical approaches. All designs assume discrete-time implementation with a fixed sampling period T_s .

Classical Discrete-Time Design Methods

These methods are direct extensions of continuous-time techniques using z-domain analysis and difference equations.

1. Discrete-Time Root Locus

Given a plant:

$$G(z) = \frac{z-a}{z^2+bz+c}$$

The root locus is plotted for the closed-loop system:

$$T(z) = \frac{KG(z)}{1+KG(z)}$$

We adjust gain \K to achieve desired pole placement inside the unit circle.

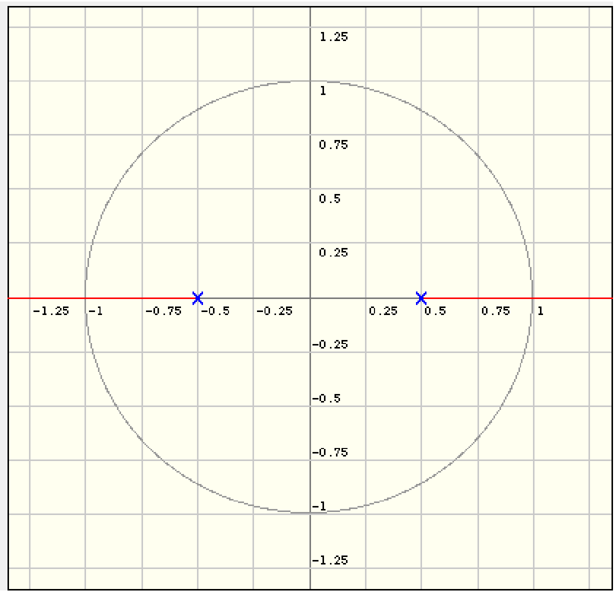

Figure 4.5: complex plane

2.Discrete-Time Bode/Nyquist Design

For frequency domain design:

- Compute the **frequency response** of G(e^{j\Omega})
- Use Bode plots to design discrete compensators like lead/lag.

Compensator in z -domain:

$$C(z) = K \cdot \frac{z-z_0}{z-p_0}$$

Design aims to adjust **gain margin**, **phase margin**, and **bandwidth**.

3. Discrete PID Control

A general digital PID controller:

$$u(k) = u(k-1) + K_p\left[e(k) - e(k-1)\right] + K_i T_s e(k) + K_d \frac{e(k)-2e(k-1)+e(k-2)}{T_s}$$

Or in transfer function form:

$$C(z) = K_p + \frac{K_i T_s}{z-1} + K_d \cdot \frac{z-2+z^{-1}}{T_s}$$

Modern Discrete-Time Control

Modern approaches use state-space and optimal control theory for sampled-data systems.

1. Discrete State-Space Feedback

Given:

$$x(k+1) = Ax(k) + Bu(k), \quad y(k) = Cx(k)$$

We apply state feedback:

$$u(k) = -Kx(k) + r(k)$$

The gain matrix K is designed to place the eigenvalues of $A - BK$inside the unit circle.

2. Discrete Linear Quadratic Regulator (DLQR)

Minimizes cost:

$$J = \sum_{k=0}^{\infty} \left[x(k)^T Q x(k) + u(k)^T R u(k)\right]$$

Solve the **Discrete Algebraic Riccati Equation (DARE)**:

$$P = A^T PA - A^T PB(R + B^T PB)^{-1} B^T PA + Q$$

Then compute:

$$K = (R + B^T PB)^{-1} B^T PA$$

3. Discrete MPC (Model Predictive Control)

- Predicts future outputs using the system model
- Solves an optimization problem at each step
- Handles constraints on u(k) , y(k)

Toolboxes like **MATLAB MPC Toolbox** support this with automatic discretization and constraint handling.

Empirical Discrete-Time Methods

1. Relay Auto-Tuning (Discrete)

- Inject a discrete relay signal (±amplitude)
- Measure output oscillation period T_u and amplitude a
- Estimate ultimate gain K_u and period T_u

Then apply **discrete Ziegler-Nichols tuning rules** for K_p, T_i, T_d

2. Step Response Tuning

- Apply a step input to the plant
- Fit the result to a First-Order Plus Dead-Time (FOPDT) model

From model:

$$G(z) \approx \frac{bz^{-d}}{1-az^{-1}}$$

Use this to calculate PID gains.

3. Auto-Tuning Software

Most industrial PLCs and digital controllers (Siemens, Omron, etc.) include PID autotuning options:

- Perform closed-loop excitation
- Fit a discrete model
- Automatically assign discrete PID parameters

V Analysis and synthesis in state space

Introduction

In this chapter, we explore modern control system design using the state-space approach. Unlike classical methods that rely on transfer functions, the state-space framework allows for a more general and flexible representation of dynamic systems, especially when dealing with multiple inputs and outputs.

We begin by analyzing the internal behavior of systems through key concepts such as **stability**, **controllability**, and **observability**. These properties help determine whether a system can be stabilized, controlled to a desired state, or observed from its outputs.

Once the system is properly analyzed, we move on to **synthesis**—designing state feedback controllers and observers to achieve specific performance objectives. This approach forms the foundation for modern digital and optimal control strategies.

1. Definition : State-Space Representation

Where:

- x(t) : State vector (describes the internal state of the system)

- u(t) : Input vector (external inputs to the system)

- y(t) : Output vector (measured outputs)

- A : System matrix

- B : Input matrix

- C : Output matrix

- D : Feedthrough (or direct transmission) matrix

In the continuous domain, the dynamic aspect of the systems is described by differential equations that we assume in this course to be linear and stationary equations (constant coefficients). Among the representations of dynamic systems,we are interested in this part in the state representation that allows us to use the calculation techniques available in linear algebra, as well as powerful control law synthesis tools such as pole placement, optimal linear quadratic control, Gaussian linear quadratic control, H1 control, μ-synthesis, . . .

In this part we will study the systems in the state space, for this we will start by implementing the state representation through a continuous example.

Example Let the series RLC circuit be represented by the following figure:

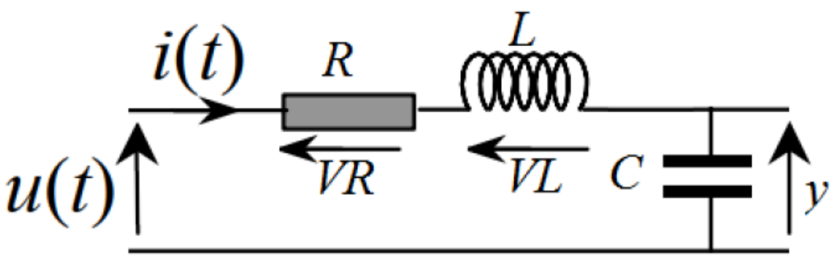

Figure 5.1: RLC circuit

This system can be represented by the following differential equation:

$u(t) = Ri(t) + L\frac{di(t)}{dt} + y(t)$

and

$y(t) = \frac{1}{C}\int i(t)\,d(t) \Rightarrow i(t) = C\frac{dy(t)}{dt}$

By substituting i(t)i(t)i(t) with its expression in equation (2.1), we obtain the following differential equation:

$LC\ddot{y} + RC\dot{y} + y - u = 0$

we define

$$\begin{cases} x_1 = y \\ x_2 = \dot{y} \end{cases}$$

$$\Rightarrow \begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = -\frac{1}{LC}x_1 - \frac{R}{L}x_2 + \frac{1}{LC}u \end{cases}$$

These equations can be written as follows:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{1}{LC} & -\frac{R}{L} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{LC} \end{bmatrix} u(t)$$

$$y = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + 0\,U(t)$$

This representation, called the state-space representation of the system.

The state-space representation is a mathematical model of a physical system expressed as a set of first-order differential (or difference) equations. It describes the system using a state vector, input vector, and output vector.

The general form of the continuous-time state-space representation is:

State equation:

$\dot{x}(t) = Ax(t) + Bu(t)$ n Output equation:

$y(t) = Cx(t) + Du(t)$

where:

u : The input vector of the system, with dimensions $n \times m$, where n is the order of the system and m is the number

y : The output vector of the system, with dimensions $n \times p\,(y \in \mathbb{R}^{n\times p})$, where p is the number of outputs.

x : The state vector, with dimensions $n \times 1\,(x \in \mathbb{R}^{n})$.

A : State matrix, with dimensions $n \times n\,(A \in \mathbb{R}^{n\times n})$.

B : Input matrix, with dimensions $n \times m\,(B \in \mathbb{R}^{n\times m})$.

C : Output matrix, with dimensions $p \times n\,(C \in \mathbb{R}^{p\times n})$.

D : Feedthrough (coupling) matrix, with dimensions $p \times m\,(D \in \mathbb{R}^{p\times m})$.

The state model can be represented by the following diagram:

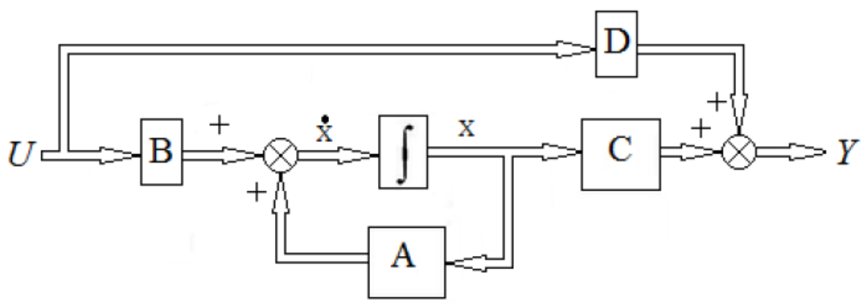

Figure 5.2: state space model

2. Stability in State-Space Systems

Stability is a fundamental property of dynamic systems. In the state-space framework, we determine the stability of a system by examining the eigenvalues of the system matrix A . For a linear time-invariant (LTI) system:

State equation (without input):

$\dot{x}(t) = Ax(t)$

The stability of the system depends on the eigenvalues of the matrix A:

- The system is **asymptotically stable** if all eigenvalues of$A$have strictly negative real parts.

- The system is **marginally stable** if all eigenvalues have non-positive real parts, and any eigenvalues on the imaginary axis are simple (i.e., no repeated eigenvalues).

- The system is **unstable** if any eigenvalue of Ahas a positive real part.

The general solution to the homogeneous state equation is given by:

$x(t) = e^{At}x(0)$

Example

$$A = \begin{bmatrix} 1 & 4 \\ 3 & 2 \end{bmatrix}, \quad B = \begin{bmatrix} 2 \\ 3 \end{bmatrix}, \quad C = \begin{bmatrix} 0 & 1 \end{bmatrix}$$

The poles of this system are given by:

$$|zI - A| = 0 \implies \left| \begin{bmatrix} z & 0 \\ 0 & z \end{bmatrix} - \begin{bmatrix} 1 & 4 \\ 3 & 2 \end{bmatrix} \right| = 0$$

$$\implies \left| \begin{bmatrix} z-1 & -4 \\ -3 & z-2 \end{bmatrix} \right| = 0$$

$$\implies (z-1)(z-2) - (-3)(-4) = 0$$

$$\implies z^2 - 3z - 10 = 0$$

$$\implies \lambda_{1,2} = \frac{3 \pm \sqrt{31}}{2}$$

Thus, the system is not stable.

3. Controllability and Observability

For the study of controllability and observability, we consider the system represented by the following state model:

$$\begin{cases} x = Ax + Bu \\ y = Cx + Du \end{cases}$$

Controllability

One of the objectives of control in the state-space is to transfer the system from any initial state to a desired state. This can only be achieved if the system is controllable.

A necessary and sufficient condition for the system to be controllable is that the rank of the controllability matrix equals the order of the system. In this case, the controllability matrix is said to be of full rank, and the system is completely controllable.

The controllability matrix is given by:

$\varphi = [B \quad AB \quad A^2B \quad A^3B \quad \cdots \quad A^{n-1}B]$

For the system to be controllable, it must satisfy:

$\mathrm{rank}(\varphi) = n$

Observability

The observability of a system in the state space allows us to reconstruct one or more non-measurable states.

For the system to be observable, the observability matrix must be of full rank, i.e.:

$\mathrm{rank}(\vartheta) = n$

Where ϑ\varthetaϑ is the observability matrix, defined as:

$$\vartheta = \begin{bmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^{n-1} \end{bmatrix}$$

Example

Controllability

Let's consider the example of the system governed by the following state equations:

$\dot{x}(t) = [A]\,x(t) + [B]\,e(t)$

with:

$$[A] = \begin{bmatrix} -2 & -2 \\ -1 & -3 \end{bmatrix} \quad \text{and} \quad [B] = \begin{pmatrix} 1 \\ -2 \end{pmatrix}$$

The controllability matrix is formed by two vectors:

$[C]_{([A]|[B])} = [[B] \quad [A][B]]$

Now:

$$[A][B] = \begin{bmatrix} -2 & -2 \\ -1 & -3 \end{bmatrix} \begin{pmatrix} 1 \\ -2 \end{pmatrix} = \begin{pmatrix} 2 \\ 5 \end{pmatrix}$$

Hence:

$$[C]_{([A]|[B])} = \begin{bmatrix} 1 & 2 \\ -2 & 5 \end{bmatrix}$$

This matrix is indeed of rank 2, since its determinant is non-zero:

$$\det\left([C]_{([A]|[B])}\right) = \begin{vmatrix} 1 & 2 \\ -2 & 5 \end{vmatrix} = 9$$

Therefore, the system is fully controllable.

Observability

Let’s consider a system defined by the following state-space representation:

$\dot{x}(t) = [A]\,x(t) + [B]\,u(t)$

$y(t) = [C]\,x(t)$

where:

$$[A] = \begin{bmatrix} 0 & 1 \\ -2 & -3 \end{bmatrix}, \quad [C] = [1 \quad 0]$$

The observability matrix is given by:

$$\mathcal{O} = \begin{bmatrix} C \\ CA \end{bmatrix}$$

We compute:

$$C = [1 \quad 0], \quad CA = [1 \quad 0] \begin{bmatrix} 0 & 1 \\ -2 & -3 \end{bmatrix} = [0 \quad 1]$$

Thus:

$$\mathcal{O} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Since the observability matrix has full rank (rank = 2), the system is fully observable.

3.1. Controllable Modes and Observable Modes

If a system is not completely controllable, that is: $\text{rank}(\varphi) = r < n$ This means there are r controllable modes and n−r uncontrollable modes.

If a system is not completely observable, that is: $\text{rank}(\vartheta) = r < n$ This means there are r observable modes and n−r unobservable modes.

Example

Consider a system represented in the state-space by:

$$\begin{cases} x(k+1) = Ax(k) + Bu(k) \\ y(k) = Cx(k) + Du(k) \end{cases}$$

$$A = \begin{bmatrix} -0.5 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0.5 \end{bmatrix}, \quad B = \begin{bmatrix} -1 \\ 0 \\ 1 \end{bmatrix}, \quad C = [0 \quad 1 \quad 1]$$

The controllability matrix is:

$$\varphi = \begin{bmatrix} -1 & 0.5 & -0.25 \\ 0 & 0 & 0 \\ 1 & 0.5 & 0.25 \end{bmatrix} \quad \Longrightarrow \quad \text{rank}(\varphi) = 2$$

Thus, we have two controllable modes: z=−0.5and z=1 and one uncontrollable mode: z=0.5

The observability matrix is:

$$\vartheta = \begin{bmatrix} 0 & 1 & 1 \\ 0 & 0 & 0.5 \\ 0 & 0 & 0.25 \end{bmatrix} \quad \Longrightarrow \quad \text{rank}(\vartheta) = 2$$

Thus, we have two observable modes: z=1 and z=0.5, and one unobservable mode: z=−0.5.

VI Bibliographic references

1. Bibliographic references

1. E. Dieulesaint, D. Royer, Automatique appliquee, 2001.

2. P. De Larminat, Automatique : Commande des systemeslineaires. Hermes 1993.

3. K. J. Astrom, T. Hagglund, PID Controllers: Theory, Design and Tuning, Instrument Society of

4. America, Research Triangle Park, NC, 1995. A. Datta, M. T. Ho, S. P. Bhattacharyya, Structure and Synthesis of PIDControllers, Springer-

5. Verlag, London 2000.

6. Jean-Marie Flaus, La regulation industrielle, Editions Hermes 1995.

7. P. Borne, Analyse et regulation des processus industriels tome 1:Regulation continue. Editions

8. Technip.

9. T. Hans, P. Guyenot, Regulation et asservissement Editions Eyrolles.

10. R. Longchamp, Commande numerique de systemes dynamiques coursd'automatique, Presses

11. Polytechniques et universitaires romandes 2006.