الجمهوريــة الجزائريــة الديمقراطيــة الشعبيــة
The People's Democratic Republic of Algeria

وزارة التعليــم العالـــي والبحــث العلمـــي
Ministry of Higher Education and Scientific Research

Echahid Cheikh Larbi Tebessi University
Faculty of Sciences and Technology
Department of Electrical Engineering

جامعـة الشهيد الشيخ العربي التبسي
كليـة العلـوم و التكنولوجيـا
قسـم الهندسـة الكهربائيـة

Course

In accordance with LMD programs Academic

Bachelor's Degree / Automatics

Sampled Control Systems

Dr. DJEDDI Abdelghani

June 2024

Semestre: 6
Unité d'enseignement: UEF 3.2.1
Matière 1: Systèmes asservis échantillonnés
VHS: 45h00 (Cours: 1h30, TD: 1h30)
Crédits: 4
Coefficient: 2

Objectifs de l'enseignement:

Connaître les techniques d'échantillonnage et de reconstruction des signaux, Etre capable d'étudier la stabilité et d'évaluer la précision d'un système asservis échantillonné, Appliquer quelques méthodes d'analyse et de synthèse des systèmes asservis échantillonnés.

Connaissances préalables recommandées:

Systèmes asservis linéaires et continus, Mathématique de base (Algèbre, analyse, …).

Contenu de la matière:

Chapitre 1. Structure d'un système de commande numérique (1 Semaine)
Historique, Avantages et inconvénients de la commande numérique, Structure générale d'un système de commande numérique, Conversions A/N et N/A, Echantillonneurs/bloqueurs.

Chapitre 2. Echantillonnage des signaux (2 Semaines)
Modélisation des Convertisseurs A/N et N/A, Echantillonnage, Reconstruction des signaux, Bloqueurs, Transmittance en Z et réponse fréquentielle d'un BOZ (bloqueur d'ordre zéro), Théorème d'échantillonnage de Shannon, Considérations pratiques.

Chapitre 3. Représentation des systèmes échantillonnés (3 Semaines)
Définitions, Représentation par les équations aux différences, Opérateurs d'avance/retard, Représentation par la réponse impulsionnelle, Transformée en Z, Transmittance en Z et simplification des blocs/diagrammes, Transformation de pôles/zéro par échantillonnage.

Chapitre 4. Analyse des systèmes échantillonnés (4 Semaines)
Conditions de stabilité, Nature temporelle des signaux du régime transitoire, Critères de stabilité (Schur-Cohn, Jury, Routh-Hurwitz, Nyquist discret, Lieu d'Evans Discret).

Chapitre 5. Synthèse des systèmes échantillonnés (4 Semaines)
Introduction, Rapidité, Précision statique, Régulateurs standard PID, Synthèse dans le plan P et numérisation, Synthèse dans le plan Z, implémentation pratique des régulateurs.

Chapitre 6. Contrôleur RST (1 Semaine)

Mode d'évaluation:

Contrôle continu: 40% ; Examen: 60%.

Références bibliographiques:

1. J.R. Ragazzini, G. F. Franklin, « Les systèmes asservis échantillonnés », Dunod, 1962.
2. Daniel Viault, Yves Quenec'hdu, « Systèmes asservis échantillonnés », ESE, 1977.
3. Christophe Sueur, Philippe Vanheeeghe, Pierre Borne, « Automatique des systèmes échantillonnés : éléments de cours et exercices résolus », Technip, 5 décembre 2000.
4. P. Borne. G.D.Tanguv. J. P. Richard. F. Rotella, I. Zambetalcis, « Analyse et régulation de processus industriels-régulation numérique », Tome 2-Editions Technip, 1993.
5. Emmanuel Godoy, Eric Ostertag, « Commande numérique des systèmes : Approches fréquentielle et polynomiale », Ellipses Marketing ,2004.

Semester: 6
Teaching Unit: UEF 3.2.1
Subject 1: Sampled Control Systems
VHS: 45 hours (Lectures: 1 hour 30 minutes, Tutorials: 1 hour 30 minutes)
Credits: 4
Coefficient: 2

Teaching Objectives:

To understand signal sampling and reconstruction techniques, To be able to study stability and evaluate the accuracy of a sampled control system, To apply some analysis and synthesis methods for sampled control systems.

Recommended Prerequisites:

Linear and continuous control systems, Basic mathematics (Algebra, analysis, etc.).

Subject Content:

Chapter 1. Structure of a digital control system (1 Week)
Historical overview, Advantages and disadvantages of digital control, General structure of a digital control system, Analog-to-digital (A/D) and digital-to-analog (D/A) conversions, Samplers/blockers.

Chapter 2. Signal Sampling (2 Weeks)
Modeling of A/D and D/A Converters, Sampling, Signal Reconstruction, Blockers, Z-Transform and Frequency Response of a ZOH (Zero-Order Holder), Shannon's Sampling Theorem, Practical Considerations.

Chapter 3. Representation of Sampled Systems (3 Weeks)
Definitions, Representation by difference equations, Lead/Lag operators, Representation by impulse response, Z-Transform, Z-Transform and simplification of blocks/diagrams, Pole/zero transformation by sampling.

Chapter 4. Analysis of Sampled Systems (4 Weeks)
Stability conditions, Time nature of transient signals, Stability criteria (Schur-Cohn, Jury, Routh-Hurwitz, Discrete Nyquist, Discrete Evans Locus).

Chapter 5. Synthesis of Sampled Systems (4 Weeks)
Introduction, Speed, Static precision, Standard PID controllers, Synthesis in the P-domain and digitization, Synthesis in the Z-domain, practical implementation of controllers.

Chapter 6. RST Controller (1 Week)
Assessment Method:

Continuous assessment: 40%; Exam: 60%.

Références bibliographiques:

1. J.R. Ragazzini, G. F. Franklin, « Les systèmes asservis échantillonnés », Dunod, 1962.
2. Daniel Viault, Yves Quenec'hdu, « Systèmes asservis échantillonnés », ESE, 1977.
3. Christophe Sueur, Philippe Vanheeeghe, Pierre Borne, « Automatique des systèmes échantillonnés : éléments de cours et exercices résolus », Technip, 5 décembre 2000.
4. P. Borne. G.D.Tanguv. J. P. Richard. F. Rotella, I. Zambetalcis, « Analyse et régulation de processus industriels-régulation numérique », Tome 2-Editions Technip, 1993.
5. Emmanuel Godoy, Eric Ostertag, « Commande numérique des systèmes : Approches fréquentielle et polynomiale », Ellipses Marketing ,2004.

Preface

This document serves as a comprehensive course companion tailored for undergraduate students in control engineering. Designed for a 45-hour course at Echahid Cheikh Larbi Tebessi University - Tebessa, Algeria, its primary aim is to explore essential aspects of digital control systems in a concise yet thorough manner.

Students are expected to have a solid foundation in mathematical concepts such as function manipulation, integral calculus, and a strong understanding of continuous-time linear systems. Building upon this foundation, the course covers key themes including the structure of digital control systems, signal sampling, representation of sampled systems, and analysis and synthesis techniques.

Throughout the course, emphasis is placed on sampling and signal reconstruction techniques, enabling students to study stability and evaluate the precision of sampled control systems effectively. Additionally, students explore the RST controller, a widely-used synthesis method, through a combination of theoretical concepts, practical examples, and hands-on exercises.

In essence, this course equips students with the necessary knowledge and practical skills to comprehend, analyze, and design digital control systems, preparing them for the challenges of modern engineering in this dynamic field.

Tebessa, May 15, 2024

Abdelghani DJEDDI

General introduction

Welcome to the comprehensive course on Digital Control Systems. This course is meticulously designed to equip you with a thorough understanding of the fundamental principles, essential components, and advanced techniques involved in the realm of digital control systems. As we embark on this educational journey, we will explore the historical evolution of digital control, the detailed structure of digital control systems, and the diverse methods employed to achieve optimal performance in these systems.

Understanding the historical context of digital control systems is crucial as it sets the foundation for grasping contemporary advancements. The roots of digital control can be traced back to the early 19th century with significant milestones such as Joseph-Marie Jacquard's invention of the automatic loom using punched cards, and Herman Hollerith's use of similar punched cards for data processing in the 1880s. These early innovations laid the groundwork for the development of numerical control systems in the aerospace industry and beyond. Throughout this course, we will highlight these key historical developments, demonstrating how they have influenced and shaped the modern landscape of digital control systems.

At the heart of digital control systems are several critical components that work in unison to ensure precise and efficient control. We will delve deeply into the roles and functions of analog-to-digital converters (ADCs) and digital-to-analog converters (DACs). Understanding these converters is fundamental as they form the bridge between the analog world of physical processes and the digital realm of computational control. You will learn how ADCs sample continuous signals at regular intervals and convert them into digital values, while DACs perform the reverse operation, transforming digital signals back into continuous analog signals.

The processes involved in digital control systems are intricate and multifaceted. This course will cover the essential aspects of sampling, quantization, and the synthesis of digital control laws. Sampling involves capturing discrete values from a continuous signal at regular time intervals, a process governed by Shannon's Theorem to avoid information loss. Quantization, on the other hand, involves approximating these sampled values to a finite set of levels, introducing

quantization noise, which we must manage to maintain system precision and stability.

The synthesis of digital control laws is another critical area we will explore. This involves formulating algorithms that define the evolution of the digital control signal based on measurements and reference signals. These control laws, often expressed as recurrence relations, are implemented in digital computers to drive the physical systems towards desired performance criteria.

Implementing digital control systems involves several methodical steps, starting with the choice of an appropriate sampling period, followed by the determination and design of digital controllers based on specified performance requirements. We will examine the various methodologies for controller design, including the use of frequency domain analysis and z-transform techniques. Additionally, you will gain insights into the practical challenges associated with digital control, such as managing quantization errors, ensuring real-time processing, and maintaining system robustness in the face of component variations and external disturbances.

The advantages of digital control are numerous, including high precision, flexibility in programming, and the ability to handle complex multi-variable systems. However, these benefits come with challenges such as high initial costs, the need for regular maintenance, and the requirement for specialized knowledge in programming and digital signal processing.

As we advance through the course, we will introduce you to sophisticated control strategies, notably Proportional-Integral-Derivative (PID) and Polynomial RST (Regulator, Simplifier, Tracker) controllers. These controllers are foundational in modern control systems. The PID controller, renowned for its simplicity and effectiveness, will be explored in depth, including its tuning methods and practical implementation. The RST controller, which offers enhanced flexibility and robustness for complex systems, will also be comprehensively covered, with a focus on its polynomial approach and practical applications.

To bridge the gap between theory and practice, we will incorporate practical examples and case studies, utilizing MATLAB for simulations and control system design. MATLAB's powerful tools for control analysis and design will help reinforce the theoretical concepts discussed, providing a hands-on learning experience that is invaluable for mastering digital control systems.

By the end of this course, you will have a solid foundation in digital control systems, enabling you to design, analyze, and optimize control systems for a wide range of industrial applications. You will be well-versed in the historical context, understand the core components and processes, and be equipped with the knowledge to tackle both the advantages and challenges of digital control. Furthermore, you will have practical skills in implementing advanced control strategies using state-of-the-art software tools.

Let's embark on this journey to master the intricacies of digital control systems, unlocking their potential to enhance performance and efficiency in modern technology. This comprehensive understanding will pave the way for innovation and excellence in the field of digital control engineering.

Chapter 1. Structure of a digital control system

1.1 Introduction

This chapter provides an in-depth understanding of the fundamental structure of digital control systems. It begins with a historical overview, highlighting key developments that have shaped the field. The chapter then delves into the essential components and processes involved in digital control systems, including analog-to-digital and digital-to-analog conversions. The implementation steps, advantages, and disadvantages of digital control, as well as the classification of different signals, are thoroughly examined to provide a comprehensive foundation for further study.

1.2 Historical Background

The history of digital control applied to industry can be traced back to the early 19th century in the city of Lyon, where Joseph-Marie Jacquard invented an automatic loom using punched cards to control the weaving of complex patterns. In the 1880s, the American statistician Herman Hollerith had the idea of using punched cards, similar to Jacquard's, for data processing. The first applications of numerical control for machining originated in the aerospace industry.

The problems to be solved for the control of continuous processes involve the following points:

(a) Sampling of a continuous signal: This operation involves sampling information taken by a continuous signal at regular time intervals, called the sampling period. This means that the computer will only consider samples, i.e., the values taken by the signal at the sampling instants.

(b) Conversion of an analog signal to a digital signal: This involves converting the value taken by an analog signal at the sampling instant into a numerical value so that it can be processed by the computer. Such a signal may,

for example, come from a sensor. It will then be referred to as a measurement signal.

(c) Conversion of a digital signal to an analog signal: This operation involves transforming the digital signal generated by the computer at the sampling instant (referred to as a digital control signal) into an analog control signal existing throughout the sampling period, with the aim of controlling the physical system.

(d) Synthesis of a calculation algorithm: This involves establishing a law for the evolution of the digital control signal based on the numerical measurement and reference signals to enable the controlled system to meet certain specifications.

This function is called a digital controller or digital control law. Its objective is to determine the value of the digital control signal at a sampling instant, based on the previous values of the digital control, measurement, and reference signals. In practice, the digital control law is expressed as a recurrence relation, which facilitates its implementation in a digital computer.

Definitions:

- The device used to perform operations (a)-(b) is called an Analog-to-Digital Converter (ADC).
- The device used to perform operation (c) is called a Digital-to-Analog Converter (DAC).

Remarks:

- In practice, the analog signal processed by the ADC is quantized on a finite scale of values. Quantifying the amplitude of the sampled signal therefore involves an approximation of the actual value of the corresponding analog signal. This can lead to the propagation of rounding errors in the computer and have consequences on the overall behavior of the controlled system (precision, stability, robustness, etc.). Although not

the subject of this course, it is important to note that the study of quantization and coding is essential for the practical implementation of digital control.

- In the context of this course, the digital control law is a linear function of the control, measurement, and reference variables with constant coefficients.
- In these lecture notes, we will tend, by abuse of language, to call sampling the entire signal processing chain, including conversion. This has no consequence, since the models we will study take into account the overall description, from signal transformation to the corresponding sequence of numbers.

1.3 General Structure of a Digital Control System

A digital control system typically consists of several key components, each playing a crucial role in the overall operation and performance of the system. The general structure of a digital control system can be outlined as follows:

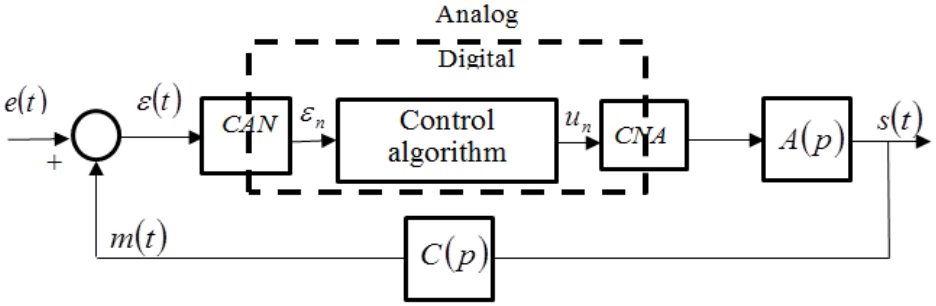


FIG. 1.1 – Typical Structure of Digital Control Implementation

1.4 Analog-to-Digital and Digital-to-Analog Conversions

Digital control is typically achieved through a structure schematized by Figure 1.1 and composed of the following fundamental components:

– **A comparator:** This provides an error signal $\varepsilon(t)$, which represents the difference between the analog reference signal $e(t)$ and the analog measurement signal $m(t)$.
– **An ADC:** This operates at the sampling period $T > 0$. It outputs the digital error signal denoted ε_n.
– **A control algorithm:** This manipulates sequences of numbers and is responsible for developing the control law. It thus produces the digital control signal u_n.
– **A DAC:** This operates at the sampling period $T > 0$. It converts the digital control signal from the computer into the corresponding analog control signal.
– **Transfer functions** $A(p)$ and $C(p)$ representing the dynamics of the system and the sensor, respectively.

In practice, the comparison operation is also done digitally. Thus, another typical control structure can be schematized by Figure 1.2, where we can notice the presence of an additional ADC.

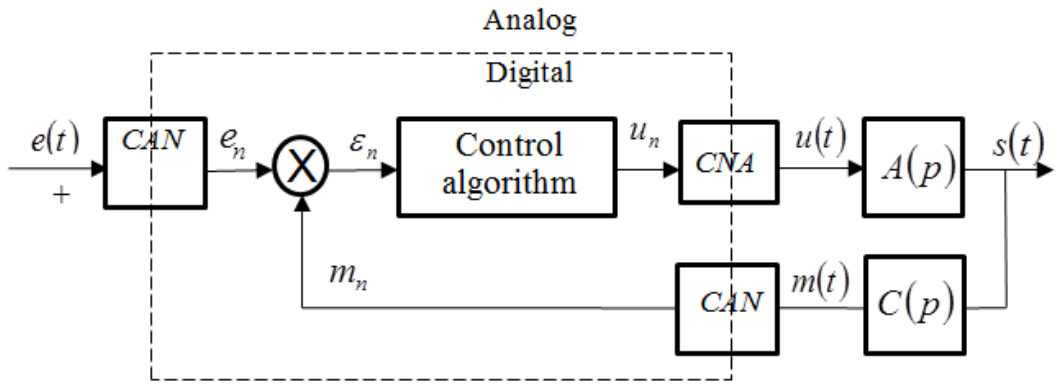


FIG. 1.2 – Another Typical Structure for Digital Control Implementation

In any control structure, a digital computer is inserted, performing, among other tasks, the functions of the control algorithm. Such a computer may be based on

microprocessors and be part of a microcontroller, an electronic card for real-time acquisition and processing, or a DSP type device, which also performs conversion operations.

1.5 Implementation Steps of Control by Computers

- Choice of the sampling period,

- Determination of the digital controller based on a set of specifications,

- The controller, typically calculated in the frequency domain, leads, by returning to the time domain, to the program of the computer (control law). It develops in real-time the command that it will send at each sampling instant or time to the process. The evolution of the system then depends at each instant on the results developed by the computer; this is referred to as real-time control by computers.

1.6 Advantages and Disadvantages of Digital Control

1.6.1 Advantages of CNC Machines

• The flexibility of using the computer (programmable machine) instead of an analog controller (hardwired machine) is remarkable:

• The flexibility of programming allows for the realization of finite controllers, easily adjustable and self-adjusting,

• Provides high precision, solves complex problems (large number of parameters), increases efficiency, improves performance, etc.

• Reduction of unproductive time,

• Reduction of scrap rates,

• Reduction of the need for inspection,

• Ability to process complex geometries,

• Ease of adapting to technical changes,

• Reduction of production setup times,

• Reduction of inventory,

• Reduction of required space,

• Reduction of operator qualification level.

1.6.2 Disadvantages of CNC Machines

• High investment costs,

• More effort must be made for maintenance,

• Programming of CNC equipment is necessary,

• Greater utilization of CNC equipment must be made to maximize profitability.

1.7 Classification of Different Signals

In this section, we present the different types of signals encountered in a digital control chain. Figure 4.8 gathers the main considered signals.

The signal represented by figure C corresponds to the typical output of a DAC with zero-order hold.

The signal represented by figure D is the digital signal as it is considered in practice, i.e., time is sampled and amplitude is quantized.

However, we have chosen, in this course, to define the signal represented by figure B as the digital signal because we never take into account the amplitude quantization step.

The sampled signal $e^*(t)$ cannot be represented in this table, as it is a continuous-time signal but with unquantized amplitude, introduced as an intermediate step to the concept of a digital signal. It is represented by Figure 1.3.

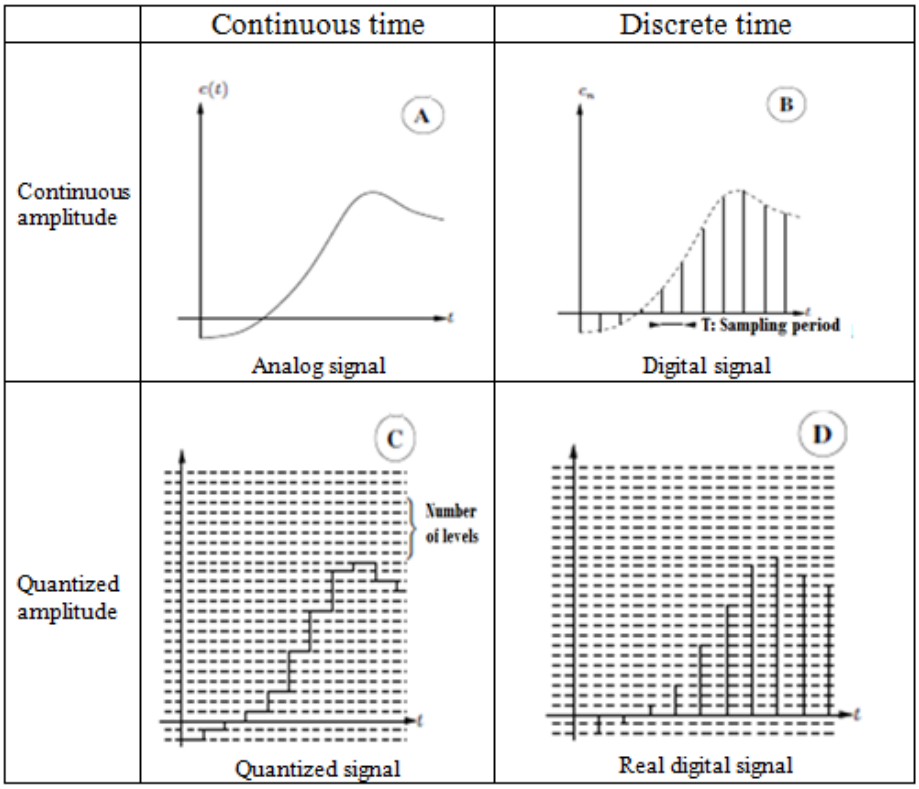


FIG. 1.3 – Classification of Signals.

1.8 Conclusion:

In summary, this chapter has provided an essential overview of digital control systems, highlighting their historical development, core components, and operational steps. Understanding the processes of analog-to-digital and digital-to-analog conversions is key to implementing effective control systems. The advantages such as flexibility and precision, along with the challenges of high costs and maintenance, were also discussed. Mastering these concepts is fundamental for anyone looking to excel in the field of digital control engineering.

Chapter 2. Signal Sampling

2.1. Introduction

In industrial reality, the complexity of systems, as well as the processing tasks to be carried out, often requires the use of digital processing tools (digital computers). These tools cannot accommodate continuous signals; they must be transformed into sequences of numbers to be processed. We speak of discrete-time systems and signals as opposed to the notion of continuous time. Transforming a continuous signal into a discrete signal requires sampling, which involves taking discrete values of the signal at regular time intervals, and then an analog-to-digital conversion that transforms these samples into coded numbers.

2.2. Modeling of Analog-to-Digital and Digital-to-Analog Converters

In Chapter 1, two blocks were presented, the Analog-to-Digital Converter noted ADC, and the Digital-to-Analog Converter noted DAC. These two blocks surround the digital computer. It is these objects that realize, as we will see, the interface between the digital world and the analog world. With the tools presented in this chapter, we can propose a model of the real technological behavior of the converters.

2.2.1 Case of the Analog-to-Digital Converter (ADC)

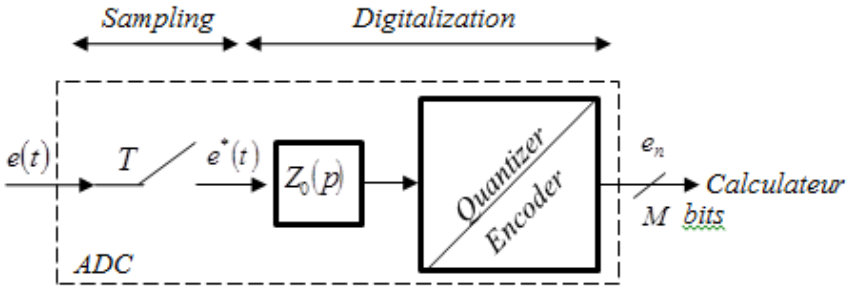


FIG. 2.1 – Modeling of the Analog-to-Digital Converter

The operation of the ADC can be schematized by Figure 4.3 and decomposed into two steps:

a. Sampler

The sampler creates a comb-like signal. It is represented by a switch whose opening and closing are synchronized at the sampling period by the computer clock. In practice, the clock frequency is much higher than the sampling frequency, to allow the computer to deliver the result of the necessary operations before acquiring the next sample.

b. Digitization

- Zero-Order Hold: The zero-order hold maintains each signal sample for the duration of the sampling period.

- Quantizer: The maintained output of the zero-order hold allows the quantizer to associate with the sample a value in an interval of integer numbers $[0,1,...,2^M - 1]$ in binary form, where M is an integer characterizing the number of bits used to quantify the analog signal, thus defining the number of available levels for quantizing the signal.

The integer number obtained from the quantizer is then associated by the encoder with another binary number, but whose value makes sense in relation to the coefficients programmed in the computer, i.e., taking into account its sign or floating-point representation. It should be emphasized here that the zero-order hold is part of the digitization step and its role is to allow the quantizer enough time to establish the quantized binary signal.

In the remainder of the digital control course, the digitization step will never be taken into account, considering that the operation of the ADC is reduced to that of a simple sampler, delivering either the sampled signal $e^*(t)$ or the digital signal e_n interchangeably, depending on the context.

This is why it will always be represented by the schema shown in Figure 2.2.

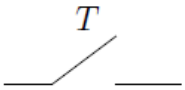


FIG. 2.2 – Standard Representation of the ADC.

2.2.2 Case of the Digital-to-Analog Converter (DAC)

The operation of the DAC can be schematized by Figure 2.3 and decomposed as follows:

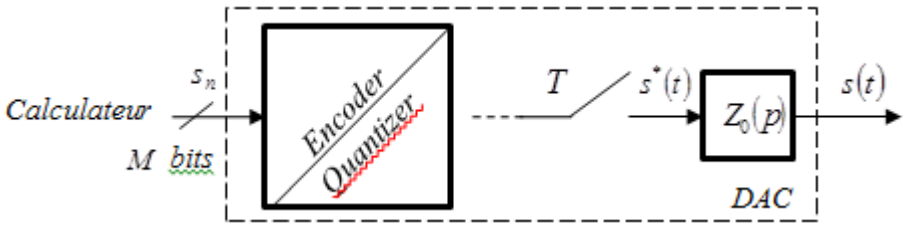


FIG. 2.3 – Modeling of the Digital-to-Analog Converter

The computer provides a value encoded in binary form but taking into account the sign or floating-point representation of the processed number. The encoder transforms this number into another binary number "readable" by the quantizer, which in turn transforms it into a sampled analog signal of appropriate value. This signal is generally held constant throughout the sampling period.

We thus find a **zero-order hold** behavior preceded by a **sampler**. Consequently, a DAC will be systematically represented by the schema illustrated in Figure 2.4.

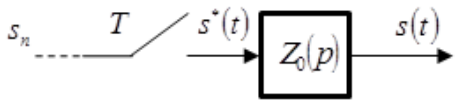


FIG. 2.4 – Standard Representation of the DAC.

In practice, depending on the technology used, during a sampling period, the signal can also evolve linearly using a first-order hold, parabolically using a second-order hold, or even polynomially. Thus, it will be sufficient to consider the expression of the appropriate transfer function of such a hold instead of the zero-order hold in Figure 2.4.

Therefore, the typical structure of digital control shown in Figure 1.2 will be represented by the schema in Figure 2.5.

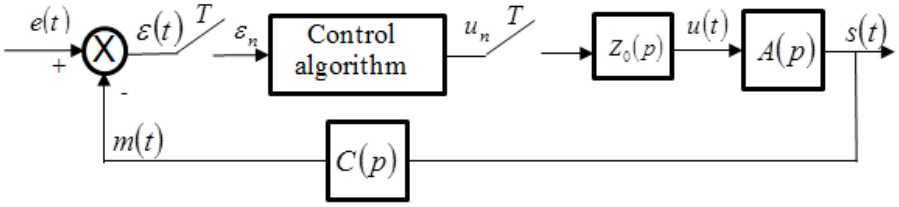


FIG. 2.5 – Typical Structure of Digital Control.

2.3 Sampling:

Sampling the causal signal $e(t)$ involves transforming it into a discrete sequence $e(nT)$ of values taken at the instants nT, where n is a natural number ($n = 0,1,2,...$) and T is the sampling period. The instants nT are called sampling instants. The intervals $[nT,(n+1)T]$ are called sampling intervals.

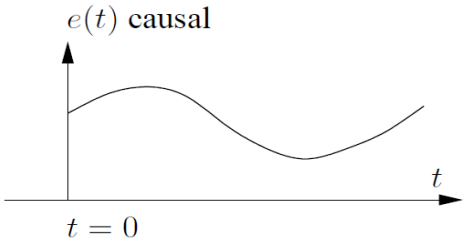


FIG. 2.6 – Causal Signal.

We first define $p(t)$, the sampling function at the instants nT, given by:

$$p(t) = \sum_{n \geq 0} \delta(t - nT) \tag{2.1}$$

where $\delta(t-t_0)$ is the Dirac delta function at time t_0 (arbitrary). The sampling function $p(t)$ is also called a temporal comb. It is represented in Figure 2.7.

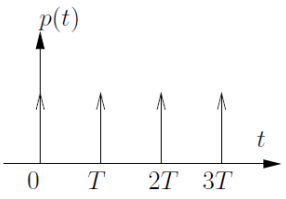


FIG. 2.7 – Causal Temporal Comb.

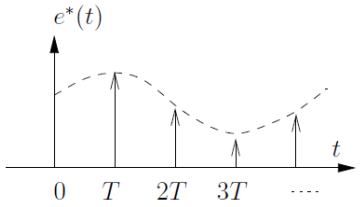


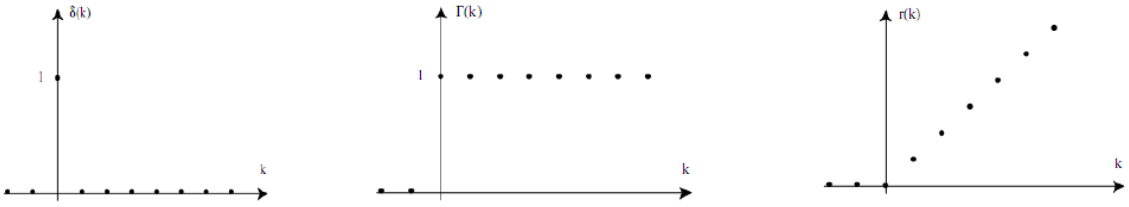
FIG. 2.8 – Temporally Sampled Signal.

Applying the sampling function to the signal $e(t)$ then produces the sampled signal $e^*(t)$ (Figure 2.8), defined at instants nT as follows:

$$e^*(t) = e(t)\,p(t) = e(t)\sum_{n\geq 0}\delta(t-nT) = \sum_{n\geq 0} e(nT)\delta(t-nT) \qquad (2.2)$$

The sequence of numbers $\{e_0, e_1, e_2, ...\}$ *with* $\backslash e_n(t) = e(nT)$ *for* $n \geq 0$ *is called the digital signal associated with* $e(t)$.

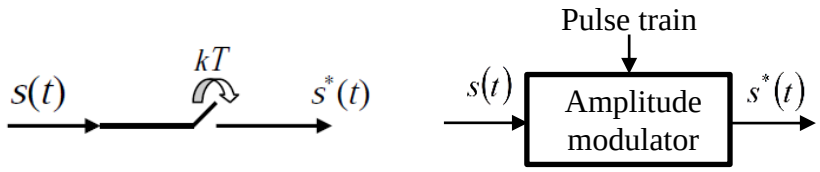
Examples of sampled signals:



Dirac pulse *Unit step* *Ramp or velocity step*

FIG. 2.9 – Examples of Simple Sampled Signals.

Examples of samplers:



Sampler or sampler *Pulse modulator*

2.3.1 Sampling Practical Rule:

One of the essential objectives of sampling is not to lose information during time discretization, which can be translated into the possibility, from the spectrum of the sampled signal, to reconstruct that of the original signal. The spectrum of the original signal given by the Fourier Transform:

$$s^*(t) \rightarrow S^*(f).* f_e \sum_{n=0}^{+\infty} \delta(f - nf_e)$$

f_s *is the sampling frequency, B is the spectral width of the signal.*

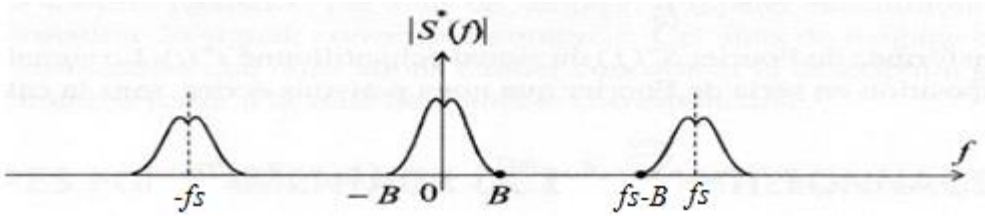


FIG. 2.10 Spectrum of a Sampled Signal.

If B is the spectral width 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 on the frequency* f_s*, extends from* $f_s - B$ *to* $B + f_s$.

The non-overlapping condition is therefore:

$B < f_s - B$, which implies $f_s > 2B$.

This inequality constitutes Shannon's theorem: the sampling frequency $f_s = \frac{1}{T}$ must be chosen to be greater than or equal to twice the upper frequency of the band of the process to be controlled.

2.3.2 Shannon's Theorem:

Among the essential concerns of sampling is the one that ensures no information loss during the temporal discretization of the continuous signal. For this to be possible, one of the conditions to be met is that the signal $e(t) = 1$ to be sampled has a finite spectral width (expressed in Hz or rad/s); this is referred to as a low-pass spectrum. It is recalled that the spectral width of such a signal is defined by the interval $[0, f_{\max}]$, where $f_{\max}$ is the highest frequency present in the frequency spectrum of this signal.

This condition stems from the phenomenon of spectrum folding (or aliasing), which is discussed in another course, and we recall the following fundamental result:

- ***Shannon's Theorem:*** *To preserve the spectral content of a signal* $e(t)$ *during sampling, the sampling frequency,* $f_s = \frac{1}{T}$*, must be greater than twice* $f_{\max}$*, the spectral width of the signal:*

$$f_s > 2 f_{\max} \tag{2.10}$$

Throughout, we assume condition (2.10) is satisfied.

- *Nyquist Frequency:* *The Nyquist frequency is defined as* $f_N = \frac{fs}{2}$.

The Nyquist frequency is of crucial importance as it defines the upper frequency limit for which standard graphical representations in the frequency domain are sufficient to represent all the spectral content of a sampled system (see Chapter 3). Graphical tools such as Bode, Black-Nichols, or Nyquist diagrams are applicable to sampled systems, and their plots can be limited to the frequency band $[0, f_N]$.

2.3.3 Practical Consequence:

In reality, any physically realizable analog signal has a "very large" spectral width, although it is necessarily of finite energy. This can arise, for example, from the presence of noise or additive disturbances whose amplitude spectra are not negligible at high frequencies. In other words, $f_{\max}$ is very large, which necessarily leads to the choice of an even larger f_s. For example, if we directly measure the signal from a stationary accelerometer sensor using a scanning oscilloscope, we find that the order of magnitude of $f_{\max}$ varies from 10^5 to 10^8 Hz!

Caution: Before sampling, it is necessary to filter the raw analog signal through a low-pass analog filter. Such a filter is called an Anti-Aliasing Filter (AAF).

2.4 Reconstruction of a Sampled Signal. Blockers.

The goal is to reconstruct $s(t)$ from the samples $s(nT)$ for $nT \le t$. The Taylor series expansion of $s(t)$ around $t = nT + \varepsilon$ is the basis of the method.

$$s(nT+\varepsilon) \cong s(nT) + \varepsilon s'(nT) + \varepsilon^2 s''(nT) + \ldots$$

We approximate the successive derivatives with finite differences:

$$s'(nT) \cong \frac{1}{T}\left[s(nT) - s\left((n-1)T\right)\right]$$

$$s''(nT) \cong \frac{1}{T^2}\left[s(nT) - 2S\left((n-1)T\right) + S\left((n-2)T\right)\right]$$

This reconstruction requires smoothing of the sampled signal; this is achieved by a zero-order hold circuit. This circuit's function is to maintain the amplitude of the pulse between instants nT and $(n+1)T$ constant and equal to $s(t)$. The computer, in addition to digital control, performs the comparison function (digitally).

2.4.1 Zero-Order Hold (ZOH) Circuit

The transfer function of a zero-order hold circuit is given by $B_0(p) = \frac{1-e^{-Tp}}{p}$.

The impulse response of a ZOH, $b_0(t)$, is therefore a square wave, which is simply written as

$$b_0(t) = u(t) - u(t-T),$$

where u is the unit step function. By applying the Laplace transform, we obtain (4.1).

2.4.2 First-Order Hold (FOH) Circuit

The first-order hold circuit allows linear extrapolation (of degree 1) between nT and $(n+1)T$ based on e_n and e_{n-1}.

The transfer function of a first-order hold circuit is:

$$B_1(p) = T(1+Tp)\left[\frac{1-e^{-pT}}{Tp}\right]^2$$

This property is derived from the extrapolation function:

$$s(nT+\tau) = e_n + \frac{\tau}{T}\left[e_n - e_{n-1}\right] \qquad (4.3)$$

2.5 Conclusion

In conclusion, this chapter has provided a detailed exploration of signal sampling in digital control systems. It highlighted the essential processes of analog-to-digital and digital-to-analog conversion, the practical rules for effective sampling, and the critical importance of Shannon's Theorem in preserving signal integrity. Understanding these concepts is fundamental for ensuring the accuracy and efficiency of digital control systems, laying a solid foundation for more advanced topics in digital signal processing.

Chapter 3. Representation of Sampled Systems

3.1 Introduction:

Controlling a process by a computer, or processor, requires the implementation of digital-to-analog and analog-to-digital conversions. Modeling, therefore, involves considering both continuous-time elements and discrete-time elements simultaneously in the loop. While it is difficult to analyze discrete-time systems as continuous-time systems with non-zero inputs and outputs only at instants, the reverse approach proves to be very fruitful. Thus, analyzing a system controlled by a digital computer involves defining a discrete-time system, including the typically continuous nature of the controlled process, and the digital-to-analog and analog-to-digital converters, which can respectively be assimilated to the zero-order hold circuit and the sampler.

3.2 Z-Transform.

The *Z*-transform is the preferred tool for the analysis and synthesis of linear, stationary, sampled, or discrete systems.

For causal signals (signal null for negative t), the *Z* -transform of the signal *x(t)* is defined as:

$$X(z) = \sum_{n=0}^{\infty} x(k) z^{-n}$$

where *z* is complex like *p*, ($z = e^{Tp}$).

3.2.1 Methods for calculating the *Z*-transform:

- By definition: $X(z) = \sum_{n=0}^{\infty} x(k) z^{-n}$

- Residue Method: $X(z) = \sum_{pôles\,de\,X(p)} \left[Résidus\ de \frac{X(p)}{1 - e^{-Tb} z^{-1}} \right]$

with $R(b)=\frac{1}{(n-1)!}\frac{d^{n-1}}{dp^{n-1}}\left[(p-b)^n\ \frac{1}{1-e^{-T(p-b)}z^{-1}}X(p)\right]$; $R(b)$ being a residue.

- Decomposition into simple elements and use of transform tables.

3.2.2 Properties of the *Z* -transform:

The main properties of the ***Z*** -transform are summarized in Appendix A.3.1.

- Linearity: $Z[\lambda_1 x_1(t)+\lambda_2 x_2(t)]=\lambda_1 Z[x_1(t)]+\lambda_2 Z[x_2(t)]$

- Transforms convolution into product: $TZ[x_1(n)*x_2(n)]=X_1(z).X_2(z)$

- Delay: $TZ(x(n-n_0))=z^{-n_0}X(z)$

- Advance: if $y(n)=x(n+n_0)$, then $Y(z)=z^{k_0}\left[X(z)-x(0)z^0-x(1)z^1-...-x(k_0-1)z^{n_0+1}\right]$

- Multiplication: $TZ\left(a^n x(k)\right)=X\left(\frac{z}{a}\right)$

- Complex translation (a complex): $TZ\left(x(p+a)\right)=X\left(e^{aT}z\right)$

- Summation: $TZ\left[\sum_{i=0}^{\infty}x_i\right]=\frac{z}{z-1}X(z)$

- Initial value theorem: $x(0)=\lim_{z\to+\infty}X(z)$

- Final value theorem: $x(\infty)=\lim_{z\to1}(z-1)X(z)=\lim_{z\to1}\left(1-z^{-1}\right)X(z)$

3.4 Modified Z-Transform:

$$X(z,m) = Z_m[x(t)] = z^{-1}\sum_{n=0}^{\infty} x((n+m)T)\, z^{-n} = \sum_{n=0}^{\infty} x((n-1+m)T)\, z^{-n} \qquad (0 \le m \le 1)$$

Properties:

In this section, we present the properties of the modified *Z*-transform. Most of these properties can be deduced from those of the *Z*-transform. Let $f(n)$, $g(n)$ and $F_m(z)$, $G_m(z)$ be two discrete-time signals, and their respective modified *Z*-transforms.

Linearity: $Z_m\{\alpha f(n) + \beta g(n)\} = \alpha F_m(z) + \beta G_m(z), \quad \forall \alpha, \beta \in R$

Correspondence with the Z-transform: $F(z,0) = z^{-1}F(z)$

Indeed, if $m = 0$ implies $\lambda = 1$, there is a delay of T_s between the two transforms.

$$\lim_{m \to 1} F(z,m) = F(z) - f(0)$$

Indeed, if *m = 1* then *λ=0*, and there is no delay between the two transforms.

Initial Value Theorem: $S(f) = \int_{-\infty}^{+\infty} s(t) e^{-2\pi f t} dt = S\int_{-\infty}^{+\infty} \cos(2\pi f_0 t) e^{-2\pi f t} dt.$

Final Value Theorem: $\lim_{n \to \infty} f((n+m)T_s) = \lim_{z \to 1}(1 - z^{-1})F(z,m)$

3.5. Discrete-Time System

A discrete-time system is defined as an operator between two discrete-time signals. Consider the system represented in Figure 3.1, where u_k represents the general term of the input sequence and y_k the general term of the output sequence. An input-output model, also called an external model, involves only the input variable u_k and the output variable y_k.

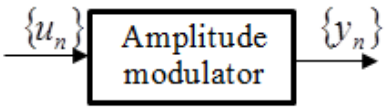


FIG. 3.1 - Discrete-Time System

In this course, we will discuss two complementary types of external models, which are recursive equations and transfer functions.

3.5.1 Recursive Equation

The initial modeling of a discrete-time system often leads to the writing of a recursive equation between different terms of the input and output sequences. The general form of a linear recursive equation can be given by:

$$a_n y_{k+n} + a_{n-1} y_{k+n-1} + \ldots + a_1 y_{k+1} + a_0 y_k = b_m y_{k+m} + b_{m-1} y_{k+m-1} + \ldots + b_1 y_{k+1} + b_0 y_k \quad (1.1)$$

By assumption, $a_n \neq 0$, and n is called the order of the system. The system is said to be causal if the outputs depend only on past events. For this, it must necessarily satisfy $m \leq n$. In this case, it is possible to write the algorithm that determines the system output given the previous inputs/outputs:

$$a_n y_k = -a_{n-1} y_{k-1} - \ldots - a_1 y_{k-n-1} - a_0 y_{k-n} + b_m y_{k+m-n} + \ldots + b_1 y_{k-n-1} + b_0 y_{k-n} \quad (1.2)$$

This formulation of the recursive equation is well-suited for numerical computation.

This is the form in which process control algorithms will be presented. The system is fully defined, and the recursive equation can be solved if the "initial" conditions are specified: $y_0, y_1, ..., y_{n-1}, u_0, u_1, ..., u_{m-1}$.

3.5.2 Z-Transform Transfer Function

In the same way that a continuous-time system is associated with a transfer function by applying the Laplace transform to its differential equation, a discrete-time system can be associated with a Z-transform transfer function by applying the Z-transform to its recursive equation. Under the assumption that the "initial" conditions are zero $y_0 = y_1 = ... = y_{n-1} = u_0 = u_1 = ... = u_{m-1} = 0$, the following relationship holds:

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

or alternatively:

$$Y(z) = \frac{N(z)}{D(z)} U(z)$$

where:

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

is defined as the Z-transform transfer function of the system. In the general case where the initial conditions are non-zero, the representation of the system in Z is written more precisely:

$$Y(z) = \frac{N(z)}{D(z)} U(z) + \frac{I(z)}{D(z)}.$$

where the polynomial $I(z)$ depends only on the initial conditions. It influences the system output without modifying the behavior due to the input signal $U(z)$.

The factorization of the numerator and the denominator leads to the following poles, zeros, gain form:

$$G(z) = \frac{b_m}{a_n} \frac{(z - z_1)(z - z_2)...(z - z_m)}{(z - p_1)(z - p_2)...(z - p_n)}$$

Where : $p_i = 1,...,n$: Poles, $z_j = 1,...,m$: Zeros, $k = \frac{b_m}{a_n}$: Gain.

By definition, the system ***poles*** are the roots of the denominator polynomial, and the system ***zeros*** are the roots of the numerator polynomial. Both of these are by default either real or complex numbers.

Some authors prefer a formulation in z^{-1} of the transfer function. It can be obtained from the z formulation as follows:

$$G(z) = \frac{b_m}{a_n} z^{m-n} \frac{1 + b^*_{m-1} z^{-1} + ... + b^*_0 z^{-m}}{1 + a^*_{n-1} z^{-1} + ... + a^*_0 z^{-n}} \qquad (1.3)$$

with the following notations: $a^*_i = \frac{a_i}{a_n}$ $b^*_j = \frac{b_j}{b_m}$

It corresponds to equation (1.2) as opposed to (1.1).

Its interest lies in representing the system as close as possible to its physical reality, in the sense that z^{-1} represents the "delay" operator which is physically realistic, whereas z assumes predicting future instants. Of course, the formulations in z and z^{-1} are equivalent. The expression (1.3) not only shows the gain, poles, and zeros but also a pure delay z^{m-n} between an input excitation of the system and its effect on the output.

It is also noteworthy that, as in the case of continuous-time systems, the denominator of the transfer function is also called the ***characteristic polynomial*** of the system. Its degree n corresponds to the ***order of the system***, and its roots are the poles of the system:

$$a_n z^n + a_{n-1} z^{n-1} + ... + a_1 z + a_0 = a_n (z - p_1)(z - p_2)...(z - p_n)$$

3.6 System Response:

The response of a system to a given input can be obtained by forming the inverse transform of $G(z)E(z)$.

Example: Consider a system defined by its recurrence equation:

$s(n+1)-0{,}5s(n)=e(n)$ with initial conditions: *s(0) = 0* and the input is a unit step.

We can calculate the output *s(n)* either step by step:

$$s(1)=0{,}5s(0)+e(0)=1$$

$$s(2)=0{,}5s(1)+e(1)=1{,}5$$

$$s(3)=0{,}5s(2)+e(2)=1{,}75$$

...

- Using the Z-transform, allows to obtain: $S(z)=G(z)E(z)=\frac{1}{z-0{,}5}\frac{z}{z-1}$

By decomposing $\frac{G(z)}{z}$ into partial fractions, we find:

$$S(z)=\frac{2z}{z-1}\frac{2z}{z-0{,}5}\Rightarrow s(n)=2-2(0{,}5)^n$$

Note: The higher the order of a system, the longer it takes for its response to manifest.

3.6.1 Pure Delay:

Systems may exhibit a pure delay, which is manifested in the recurrence equation by the appearance of terms in $e(n-r)$ where r is the delay in the number of sampling periods.

***Example**:*

The system defined by the recurrence equation:

$$s(n+3)+3s(n+2)+2s(n+1)+s(n)=e(n+1)+2e(n)+e(n-1)+e(n-2)$$

is of order 3 and has a delay of 2 steps or sampling periods (due to the presence of the term $e(n-2)$).

The input system *e(n)*, output *s(n)*, and transfer function $G(z)=\frac{S(z)}{E(z)}$ exhibit a pure delay of r sampling periods if *G(z)* can be written in the form $G(z)=z^{-r}G_1(z)$.

3.6.2 Integral and Derivative Functions in Digital:

- Derivative Function: $y(t)=\frac{du(t)}{dt}\rightarrow y(n)=\frac{u(n)-h(n-1)}{T}=u(n)\left(\frac{1-z^{-1}}{T}\right)$

The "derivative" transfer function is thus: $\left(\frac{1-z^{-1}}{T}\right)$

- Integral Function:

$$y(t)=\int u(t)dt\rightarrow y(n)=\sum_{0}^{n}Tu(n)\ \Rightarrow\ y(n)(1-z^{-1})=Tu(n)\ \Rightarrow\ y(n)=\left(\frac{T}{1-z^{-1}}\right)u(n)$$

The "integral" transfer function is thus: $\left(\frac{T}{1-z^{-1}}\right)$

3.7 Sampled System

Controlling a process by a computer, or processor, requires the implementation of digital-to-analog and analog-to-digital conversions (see Figure 1.1).

Modeling thus involves considering both continuous-time elements and discrete-time elements simultaneously in the loop. While analyzing discrete-time systems as continuous-time systems with non-zero inputs and outputs only at instants is cumbersome, the reverse approach proves to be very rich.

Thus, analyzing a system controlled by a digital computer involves defining a discrete-time system, including the typically continuous nature of the controlled process, and the digital-to-analog and analog-to-digital converters, which can respectively be assimilated to the zero-order hold circuit and the sampler, according to the scheme in Figure 3.2.

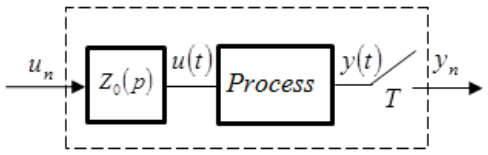


FIG. 3.2 - Sampled Process

The models between u_k and y_k are of the type presented previously. The rest of this section will focus on techniques for determining the discrete-time sampled model obtained from the continuous model of the process.

Before that, it is important to revisit the choice of the sampling period. Shannon's theorem specifies that the sampling frequency $f = 1/T$ must be at least twice the highest frequency contained in the spectrum of the signal to be sampled. This result is only exploitable given a signal. However, the output signal of a system $y(t)$ is not known in the considered problem.

The real problem envisaged is the output sampling of a process for which, for example, its transfer function is known but the system output is unknown because it depends on the input signal $u(t)$, which is not specified.

The method then consists of analyzing the frequencies transmitted by the system. By plotting the Bode diagram, it is possible to determine the cutoff frequency f_c of the system and thus indicate that all frequencies above f_c in the output signal spectrum will be attenuated.

Theorem 1.1 *In practice, it is recommended to choose the sampling frequency in a range of about 6 to 24 times the cutoff frequency of the process.*

Example 1.4:

Thus, for a first-order process: $G(p)=\dfrac{1}{1+\tau p}$

the cutoff frequency is $f_c = 1/(2\pi\tau)$. The sampling frequency $f = 1/T$ will be chosen such that:

$$\frac{6}{2\pi\tau} < \frac{1}{T} < \frac{24}{2\pi\tau}$$

approximately: $\dfrac{\tau}{4} < T < \tau$.

3.7.1 Sampled Transfer Function

In this subsection, the calculation method that allows deducing the discrete-time Z-model of the discrete-time system obtained by sampling given a transfer function of a continuous-time system is explained. It is summarized in the following theorem.

Theorem 1.2 *Let a continuous process be modeled by a transfer function* $G(p)$. *This process, sampled according to the scheme in Figure 1.6, has a Z-transform transfer function such that:*

$$G(z) = Z\left[B_0(p)G(p)\right] = \frac{z-1}{z} Z\left[\frac{G(p)}{p}\right]$$

Before proceeding to prove this result, it is necessary to detail the notation $Z[H(p)]$ *where* $H(p)$ *is a transfer function of a continuous system. This notation covers the following operation:*

$$H(p) \xrightarrow{L^{-1}} h(t) \xrightarrow{T} h_n \xrightarrow{Z} H(z)$$

Given a transfer function $H(p)$*, it is first necessary to calculate its impulse response* $h(t)$*, then sample this signal,* $\{h_n\} = \{h(nT)\}$*, and finally calculate its Z-transform* $Z(h_n) = H(z)$.

Proof of Theorem 1.2:

Let $u^*(t)$ be the continuous-time signal consisting of the samples $u_n = u(nT)$ of the control signal and which is zero everywhere else: $u^*(t) = \sum_{n=0}^{+\infty} u_n \delta(t - nT)$

The Laplace transform of this signal is written as:

By definition of transfer functions, the continuous signals *U(p)* and *Y(p)* satisfy:

Thus, by setting which, applying the delay theorem, gives:

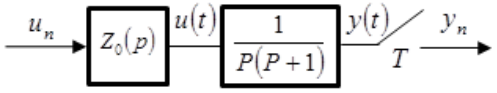


FIG. 3.3 - Sampled System

3.7.2 Properties of the Sampled Model

Following the formulas that allow determining the discrete-time model of a sampled continuous-time system, we can highlight some fundamental properties of this operation:

- A continuous linear system remains linear after sampling.

- The order of the system is preserved.

- The poles of the sampled system are deduced from the poles of the continuous system as follows:

$p_{di} = e^{p_{ci}T} \quad \forall i = 1,...,n$ where p_{ci} are the poles of the continuous system, p_{di} are the poles of the sampled system, and T is the sampling period.

- The sampling period T strongly influences the model of the sampled system.

- Sampling the product of two transfer functions is not equal to the product of their respective sampled models. This latter remark is very important. The calculation of a sampled system only makes sense if it corresponds to a transfer between a zero-order hold circuit and a sampler.

3.8 Sampled Transfer Function

Consider a sampler interposed between a continuous linear system and its input.

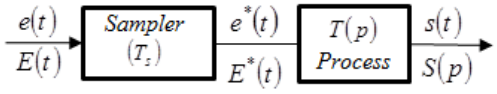


FIG. 3.4 - Sampled Continuous System

Now, the output of a continuous system is continuous, so here: $S(p) = T(p)E^*(p)$, and as seen previously:

$$e^*(t) = e(t)p(t) = e(t)\sum_{n\geq 0}\delta(t-nT_s) = \sum_{n\geq 0}e(nT_s)\delta(t-nT_s)$$

$$e^*(t) = \sum_{n\geq 0}e(nT_s)\delta(t-nT_s) \quad \xrightarrow{TP} \quad E^*(p) = \sum_{n\geq 0}e(nT_s)e^{-nT_sP}$$

Equation 2 $E^*(p) = \sum_{n=0}^{\infty}e(nT_s)e^{-nT_sp}\delta(t-nT_s)$ than $S(p) = \sum_{n=0}^{\infty}e(nT_s)T(p)e^{-nTp}$

In this last expression, we find the term $\Delta_0^*(p) = \int_0^{+\infty}\delta_0^*(t)e^{-pT_s}$, which is the delayed Laplace transform of the transfer function $t(t-nT_s)$.

This allows us to return to the original $s(t)$, thus:

Equation 3 $s(t) = TL^{-1}[S(p)] = \sum_{n=0}^{\infty} e(nT_s)t(t - nT_s)$

Based on the above, we can express the Z-transform of the output $s(t)$ and the input $e(t)$, thus:

So, ultimately:

$$Z[t(t - nT_s)] = z^{-n}T(z)$$

Equation 4 $S(z) = Z[s(t)] = \sum_{n=0}^{\infty} e(nT_s)z^{-n}T(z) = T(z)\sum_{n=0}^{\infty} e(nT_s)z^{-n}$

Equation 5 $E(z) = Z[e(t)] = \sum_{n=0}^{\infty} e(nT_s)z^{-n}$

So, in the end: $S(z) = T(z)E(z)$, that is:

Equation 6 $T(z) = \frac{S(z)}{E(z)}$,

Represented by the following diagram:

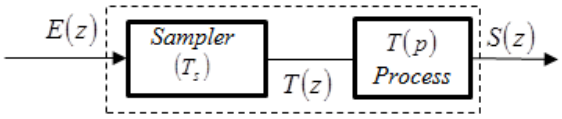


FIG. 3.5 - Transfer Function $T(z)$ of a Sampled System

3.8.1 Influence of Sampler Position

A careful observation of the sampler's position is necessary when calculating the Z-transforms.

Cascade Sampled Systems

Consider at least two sampled systems in cascade.

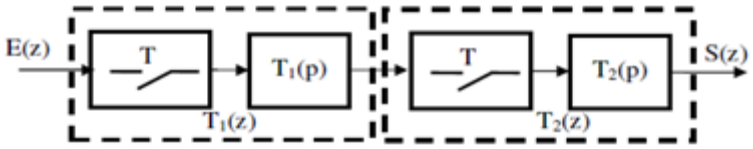


FIG. 3.6 - Cascade Sampled Systems

According to the above, we can write the Z-transforms of each system:

$$S_1(z) = T_1(z)E(z) \quad et \quad S(z) = T_2(z)S_1(z)$$

By substitution, we obtain: $S(z) = T_1(z)T_2(z)E(z)$

from which we derive: $S(z) = T_1(z)T_2(z)E(z)$

Equation 7 $T(z) = T_1(z)T_2(z)$

Sampling of Continuous Systems in Cascade

Consider at least two continuous systems in cascade that we want to sample.

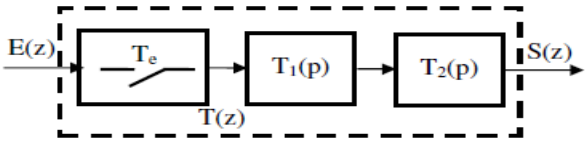


FIG. 3.7 - Sampling of Continuous Systems in Cascade

We are in the same conditions as those used to determine the Z-transform in Equation 6 above, so $S(z) = T(z)E(z)$ with:

Equation 8 $T(z) = \overline{T_1T_2}(z)$

This notation allows distinguishing between the two cases, whose results are different.

3.8.2 Transfer Function of a Process Equipped with a ZOH Block

Consider a continuous process, a Zero-Order Hold (ZOH) block, and a ZOH sampler:

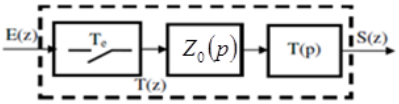


FIG. 3.8 - Continuous Process Equipped with a ZOH and ZOH Sampler

Let's write the output equation:

Equation 9 $T(z) = \overline{T_{ZOH}T}(z)E(z)$

We have seen that: $T_{ZOH}(p) = \frac{1-e^{-T_s p}}{p}$, hence $T_{ZOH}(p)T(p) = \frac{1-e^{-T_s p}}{p}T(p)$.

the sampled transfer function of the entire system is therefore:

Equation 10 $\boxed{\overline{T_{BOZ}T}(z) = (1-z^{-1})Z\left[\frac{T(p)}{p}\right]}$

3.8.3 Z-Transform of a System in Unity Feedback

Consider a process and its controller, which acts as a blocker, to be sampled. In practice, we sample the input, the measurement (feedback), and the output of the controller sampler. The scheme can therefore be reduced to this:

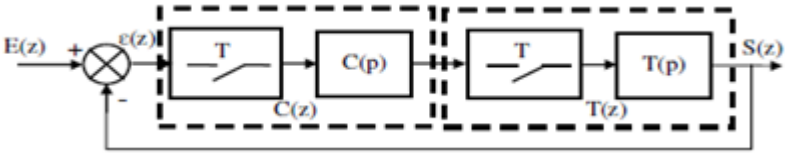


FIG. 3.9 - Sampled Unity Feedback Controlled System

The forward path consists of elements in cascade, each with a sampler, hence:

$S(z) = C(z)T(z)\varepsilon(z)$, we express $\varepsilon(z) = E(z) - S(z)$, which leads to

$S(z) = C(z)T(z)[E(z) - S(z)]$ from which we derive:

Equation 11 $H(z) = \dfrac{C(z)T(z)}{1 + C(z)T(z)}$

3.8.4 Z-Transform of a System in Non-Unity Feedback

Let's conduct the same study for a sampled system in non-unity feedback. The scheme is then as follows:

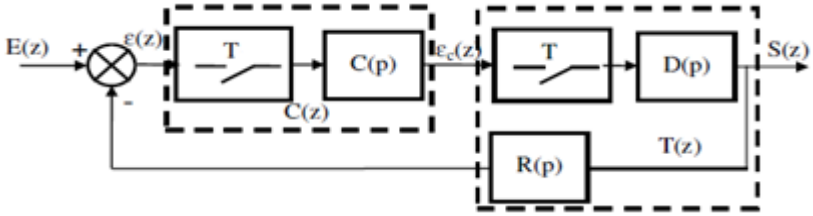


FIG. 3.10 - Sampled Unity Feedback Controlled System

We can establish the following relationships:

$$T(z) = \overline{DR}(z)\varepsilon_c(z), \quad S(z) = C(z)D(z)\varepsilon(z) \quad et \quad \varepsilon(z) = E(z) - T(z)$$

Let:

$$\begin{aligned} S(z) &= C(z)D(z)[E(z) - T(z)] \\ &= C(z)D(z)E(z) - C(z)D(z)\overline{DR}(z)\varepsilon_c(z) \\ &= C(z)D(z)E(z) - C(z)D(z)\overline{DR}(z)\frac{S(z)}{D(z)} \end{aligned}$$

If we group the terms in $S(z)$: $S(z)\left[1 + C(z)\overline{DR}(z)\right] = C(z)D(z)E(z)$, hence finally:

Equation 12 $H(z)=\frac{S(z)}{E(z)}=\frac{C(z)D(z)}{1+C(z)\overline{DR}(z)}$

The sampling period is calculated based on the apparent inertia of the closed-loop system.

3.8.5 Choice of Sampling Frequency

We have seen that the choice of sampling frequency influences the transfer function of sampled systems. However, just like continuous systems, we need to set stability, precision, and speed criteria for sampled systems. We will set the sampling frequency or period within a range that directly depends on the process. Let's examine the cases for a first-order and a second-order system.

Sampling Period for a First-Order Continuous System

Consider a closed-loop first-order continuous system with transfer function: $H(p)=\frac{K}{1+\tau p}$. We know that for this order, we have $\omega_n=\omega_c(-3dB)=\frac{1}{\tau}$.

If we assume that this frequency represents the maximum bandwidth frequency of the system, we can deduce that $f_c(-3dB)=f_M=\frac{1}{2\pi\tau}$. Then, applying the relation we have established, we obtain:

$$\frac{2\pi\tau}{25}<T_s<\frac{2\pi\tau}{5},\ \text{where}: 0{,}25\,\tau<T_s<1{,}25\,\tau,$$

Sampling Period for a Second-Order Continuous System

Consider the first canonical form of $H(p)$: $H(p)=\frac{K}{1+\tau p}$. We can also express the cutoff frequency at $-3dB$, in terms of the natural frequency, as:

$$\omega_c(-3dB)=\omega_n\sqrt{(1-2z^2)+\sqrt{(1-2z^2)+1}}.$$

In most cases, we settle for $z = 0{,}7$, which allows for low overshoot and minimum response time. In this case, the calculation leads to $\omega_n \cong \omega_c(-3dB) = \frac{1}{\tau}$.

Therefore, we can deduce as before:

$$\frac{2\pi}{25\omega_n} < T_s < \frac{2\pi\tau}{5\omega_n}, \text{ where : } \frac{0{,}25\,\tau}{\omega_n} < T_s < \frac{1{,}25\,\tau}{\omega_n},$$

Sampling Period for a System with Pure Time Delay

If the system has a pure time delay R, typically the case in industrial processes, we can choose the sampling period as follows: $T_s < \frac{R}{n}$, where $n \in N$,

Or

$T_s = \frac{R - R'}{n}$, where $n \in R'$ and R' is a delay such that $0 < R' << T_s$, and R' always $\frac{T_s}{2}$.

The polynomial form directly provides the coefficients of the recurrence:

$$G(z) = \frac{b_1 z + b_0}{z^2 + a_1 z + a_0}$$

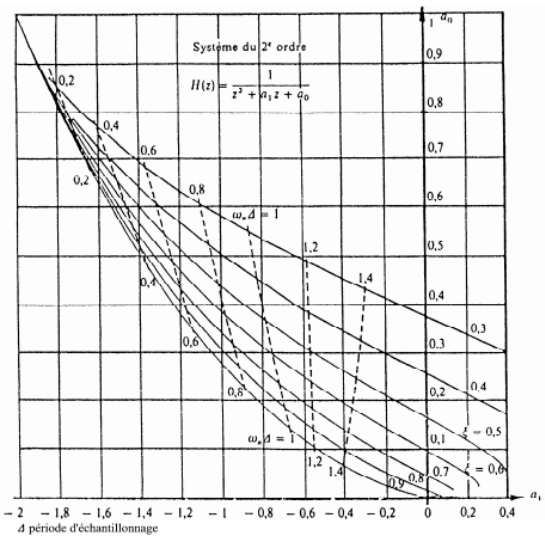
The poles and zeros form allows for an evaluation equation of dynamic performance:

$$G(z) = b_1 \frac{z - z_0}{(z - z_1)(z - \overline{z_1})} \;;$$

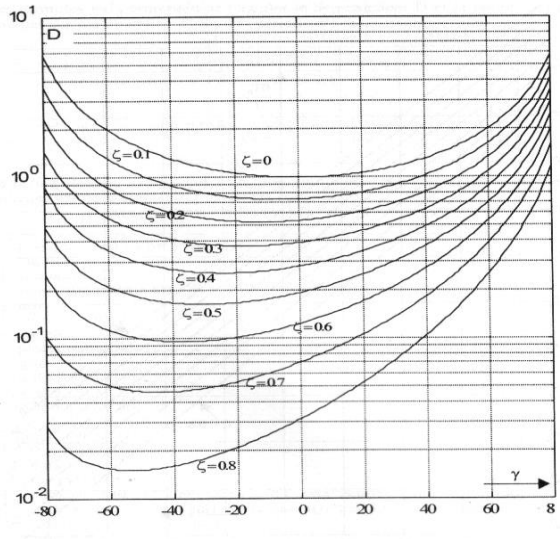
with:

$$
\begin{aligned}
&\alpha = e^{-\xi\omega_n T} \\
&\omega_n' = \omega_n\sqrt{1-\xi^2} \\
&a_0 = \alpha^2 \\
&a_1 = -2\alpha\cos(\omega_n T) \\
&b_0 = \alpha^2 + \alpha\left[\xi\frac{\omega_n}{\omega_n'}\sin(\omega_n' T) - \cos(\omega_n' T)\right] \\
&b_1 = 1 - \alpha\left[\xi\frac{\omega_n}{\omega_n'}\sin(\omega_n' T) + \cos(\omega_n' T)\right] \\
&z_0 = -\frac{b_0}{b_1} \\
&z_1 = e^{(-\xi\omega_n + j\omega_n')T} \\
&\bar{z}_1 = e^{(-\xi\omega_n - j\omega_n')T}
\end{aligned}
$$

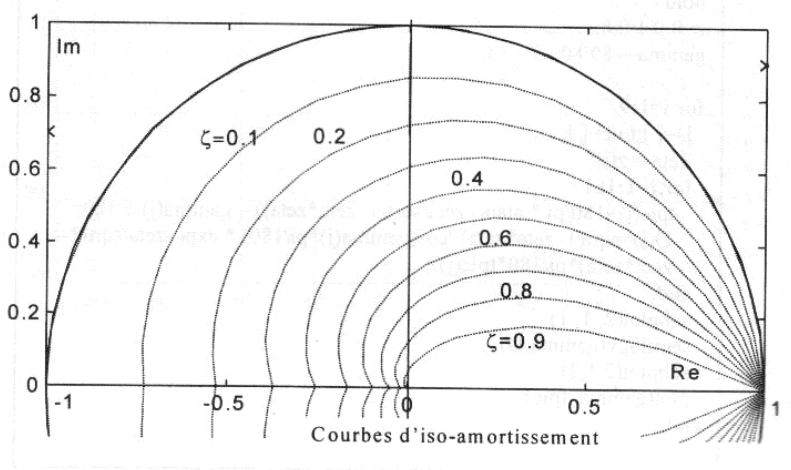
Curves, Charts to Use



(Second-Order Digital Abacus): Parameters



(Second-Order Digital Abacus): Overshoot



(Second-Order Digital Abacus): Constant Damping Pole Location

3.9 Poles - Zeros - Stability - Causality:

3.9.1 Poles:

The poles z_i of a transfer function system $G(z)$ characterize the dynamics of the system (the transient regime). The poles are the roots of the characteristic equation that defines the free regime (related to the initial conditions) of the system.

Causality:

For the transfer function system:

$$G(z) = \frac{S(z)}{E(z)} = \frac{\sum_{k=0}^{n} b_k z^{-k}}{\sum_{k=0}^{m} a_k z^{-k}} = \frac{b_m z^{-m} + ... + b_1 z^{-1} + b_0}{a_n z^{-n} + ... + a_1 z^{-1} + a_0}$$

The causality of the system requires $n \geq m$ otherwise the temporal recurrence is expressed in terms of the future and not the past.

3.9.2 Zeros of a digital system:

Similarly to analog systems, zeros determine whether the system is at minimal phase (a system is at minimal phase if the zeros of its transfer function are stable). An unstable zero (with magnitude >1) is sufficient to make a system non-minimum phase (a system not fast compared to a minimal phase system).

The zeros of a system characterize its speed. A system with zeros is a system whose second term contains (memorizes) terms from the input excitation:

$$e(n)\ ,\ e(n-1)\ ,\ e(n-2),...$$

Introducing a stable zero thus accelerates the system.

Adjusting a zero as an unstable zero, on the other hand, causes a delay in the system. A zero $z_0 = 0$ represents a pure advance of one sampling step. $zG(z)$ is faster than $(z-0{,}5)G(z)$, itself faster than $G(z)$.

Note: There is no simple relationship between the zeros of $G(p)$ and those of $G(z)$. $G(z)$ can have zeros while $G(p)$ does not, as can be seen in the following example:

Consider the system:

$g(t) = -at$ then: $G(p) = \dfrac{1}{p+a}$ and $G(z) = \dfrac{1}{z - e^{-aT_e}}$.

Thus, there exists a pole in z while there is none in p.

3.10 Conclusion

In conclusion, this chapter has provided an essential foundation in understanding the representation of sampled systems, focusing on the Z-transform and its properties, methods of calculating the Z-transform, and the significance of modified Z-transform. The analysis of discrete-time systems, including recursive equations and transfer functions, equips us with the tools to model and analyze digital control systems effectively. By grasping these concepts, we can ensure accurate system responses and optimize the design and implementation of digital control systems in practical applications.

Chapter 4. Analysis of Sampled Systems

4.1 Introduction

In the field of control systems, the analysis of sampled systems is crucial for understanding and improving the performance of digital control applications. Sampled systems are those in which signals are sampled at discrete intervals and processed digitally. This chapter delves into the fundamentals of sampled systems, examining their stability, analyzing different criteria for stability assessment, and exploring algebraic and geometric methods to evaluate and enhance system performance.

The design of a control system must ensure stability in closed-loop operation above all. This chapter is dedicated to presenting tools for studying the stability of control systems. Stability is a key requirement in control systems, ensuring that the system performs its desired function without becoming unstable or oscillatory over time. Various criteria and methods are used to assess and guarantee stability in sampled systems, providing a comprehensive approach to control system design and analysis.

4.2 Study of Stability

The design of a control system must ensure stability in closed-loop operation above all. This chapter is dedicated to presenting tools for studying the stability of control systems.

4.2.1 Stability

4.2.2 General Stability Condition

Definition:

A sampled signal $\sum_{n=0}^{+\infty} e_n \delta_n$ is bounded if the samples in e_n are bounded above, meaning there exists $M > 0$ such that $\forall n \in N, |e_n| \leq M$.

Definition:

A system is said to be stable in the sense of bounded-input bounded-output (BIBO) when every bounded input corresponds to a bounded output.

Property:

A sampled system, with transfer function $G(z)$, is stable if and only if:

The poles of $G(z)$ have magnitudes strictly less than 1. (6.1)

Proof: We will only demonstrate the sufficiency of condition (6.1), meaning that if the poles of $G(z)$ have magnitudes strictly less than 1, then the sampled system is stable.

Suppose all the poles of $G(z)$ have magnitudes strictly less than 1. Let $l = \max|\lambda_i| < 1$ where λ_i are the poles of $G(z)$. By considering the partial fraction decomposition of the rational function $G(z)$, it is easy to see that we can obtain a Laurent series expansion for $|z| > l$. This implies that the unit circle is strictly within the convergence region of the Laurent series, and thus, $G(1)$ is absolutely summable.

We then define: $k = \sum_{0}^{+\infty} |g_n|$

Recall that the output of a digital system is the convolution product of the digital input signal with the digital impulse response of the system (see (5.5)). We then clearly have the following implication:

$si \quad |g_n| \le M \quad \forall n \in N, alors\, |s_n| \le M\,K \quad \forall n \in N$,

meaning BIBO.

A proof of the necessity of (6.1) extends beyond the scope of these notes.

Note:

- This rule is general as it applies equally whether $G(z)$ represents an open-loop system or a closed-loop control system.

- The correspondence between continuous-time stability and discrete-time stability is illustrated in the figure below.

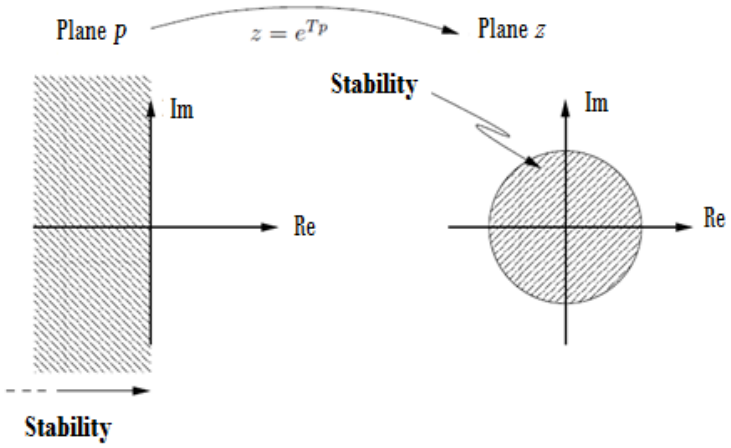


Fig. 4.1 - Equivalence between the s-plane and the Z-plane

The criteria presented below allow assessing the stability of a sampled control system based on knowledge of its open-loop transfer function.

Consider again the structure of the sampled control system represented by the following diagram:

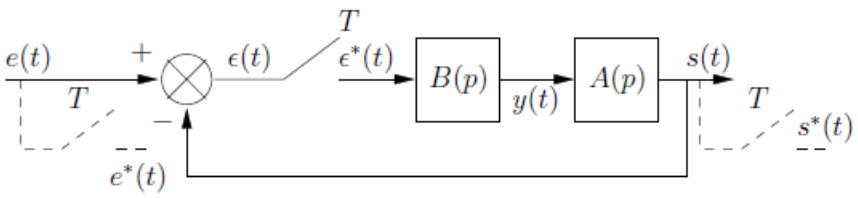


Fig. 4.2 - Sampled control system structure

4.3 Algebraic Criteria

The starting point of these criteria is the characteristic equation of the closed-loop transfer function of the sampled system:

$$\frac{S(z)}{E(z)} = \frac{\overline{AB}(z)}{1 + \overline{AB}(z)} = \frac{\gamma_n z^n + ... + \gamma_0}{c_n z^n + ... + c_0} \tag{4.1}$$

- Definition:

The characteristic equation of the transfer function (Equation (4.2)) is obtained by setting the denominator of the transfer function to zero:

$$c_n z^n + c_{n-1} z^{n-1} + ... + c_1 z + c_0 = 0 \tag{4.2}$$

4.3.1 Schur-Cohn Matrix Criterion

This criterion is applicable regardless of the order of the system. It determines whether all the roots of the characteristic equation have magnitudes less than 1, in which case, the closed-loop system is deemed stable.

Definition:

The Schur-Cohn matrix is expressed in terms of the coefficients of the characteristic equation polynomial:

$$SC_k = \begin{bmatrix} c_0 & 0 & \cdots & \cdots & 0 & c_n & c_{n-1} & \cdots & \cdots & c_{n-k+1} \\ c_1 & \ddots & \ddots & & \vdots & 0 & \ddots & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & \ddots & 0 & \vdots & & \ddots & \ddots & c_{n-1} \\ c_{k-1} & \cdots & \cdots & c_1 & c_0 & 0 & \cdots & \cdots & 0 & c_n \\ c_n & 0 & \cdots & \cdots & 0 & c_0 & c_1 & \cdots & \cdots & c_{k-1} \\ c_{n-1} & \ddots & \ddots & & \vdots & 0 & \ddots & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & \ddots & \ddots & 0 & \vdots & & \ddots & \ddots & c_1 \\ c_{n-k+1} & \cdots & \cdots & c_{n-1} & c_n & 0 & \cdots & \cdots & 0 & c_0 \end{bmatrix} \tag{4.3}$$

Stability analysis involves calculating the determinant of the Schur-Cohn matrix.

Let's denote:

$$\Delta_k = \det(SC_k) \tag{4.4}$$

The Schur-Cohn stability criterion is stated as follows:

Property:

The sampled system will be stable in closed-loop if for all k such that $1 \le k \le n$:

$$\Delta_k < 0 \quad pour \quad k \quad impair$$

$$\Delta_k > 0 \quad pour \quad k \quad pair$$

Example: **System of order** $n = 2$

The characteristic equation of such a system is:

$$c_1 z^2 + c_1 z + c_0 = 0$$

Calculate Δ_k for $k = 1$ and then for $k = 2$:

$\Delta_1 =$

$\Delta_2 =$

What conditions must the coefficients of the characteristic equation polynomial satisfy for the closed-loop system to be stable?

Example: **System of order** $n = 2$

In such a case, verifying the Schur-Cohn criterion amounts to directly solving the characteristic equation:

$$c_1 z + c_0 = 0,$$

giving the expression of the closed-loop system pole. This pole will be stable if its magnitude is less than 1.

Let's consider the following system:

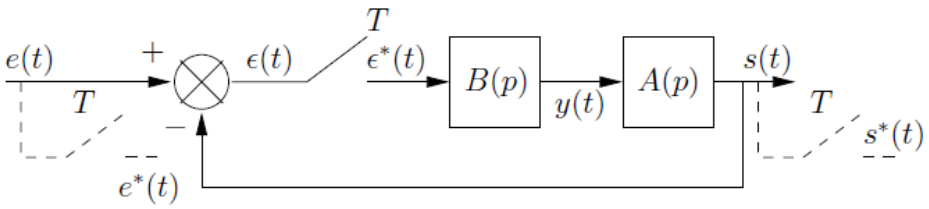


Fig. 4.3 - Control system structure

$$A(p) = \frac{A_0}{p + a} \tag{4.5}$$

$$B(p) = B_0(p) \tag{4.6}$$

$$B(p) = \frac{1 - e^{-Tp}}{p} \tag{4.7}$$

It is recalled that $F(z) = \frac{S(z)}{E(z)}$ has the expression:

$$F(z) = \frac{\overline{AB}(z)}{1 + \overline{AB}(z)} \tag{4.8}$$

and that:

$$\overline{AB}(z) = \frac{A_0}{a} \frac{1 - e^{-aT}}{(z - e^{-aT})} \tag{4.9}$$

- What is the order of the closed-loop system?

- What is its characteristic equation?

- Assuming that A_0 is the only tuning parameter, determine the conditions for closed-loop stability.

4.3.2 Jury criterion

The Jury criterion addresses the stability problem of a discrete-time linear system based on the knowledge of the characteristic polynomial.

Let's consider the discrete-time system with transfer function:

$G(z)=\dfrac{N(z)}{D(z)}$ where $D(z)=a_0+a_1z+...+a_nz^n$ and $a_n>0.$

When the coefficients of $D(z)$ are real, the Jury criterion allows us to conclude on the stability of the system without calculating the roots of the denominator or the characteristic polynomial $D(z)$.

Theorem 4.3.2 (Jury's Criterion): *Let*

$$\begin{cases} a_{0,i}=a_i, \quad \forall i=0,1,...,n, \\ a_{j+1,i}=\begin{vmatrix} a_{j,0} & a_{j,n-j-i} \\ a_{j,n-j} & a_{j,i} \end{vmatrix}, \quad \forall j=0,1,...,n-1 \quad et \quad 0\le i\le n-j-1. \end{cases}$$

The polynomial $D(z)$, with real coefficients, has roots with modulus less than unity if and only if the following inequalities are satisfied:

1. $|a_0|-a_n<0,$
2. $D(1)>0,$
3. $(-1)^nD(-1)>0,$
4. $|a_{j,0}|-|a_{j,n-j}|>0, \quad \forall j=1,2,...,n-2.$

Let's study this criterion for systems of order less than or equal to four:

Case n = 1: Then, $D(z)=a_0+a_1z+a_2z^2$, and the conditions of Theorem 4.3.2 are:

1. $|a_0|<a_1,$
2. $a_0+a_1>0,$
3. $-a_0+a_1>0.$

The fourth condition of Theorem 4.3.2 does not apply because $n<3$.

Case n = 2: Then, $D(z) = a_0 + a_1 z + a_2 z^2$, and the conditions of Theorem 4.3.2 are:

1. $|a_0| < a_2$,
2. $a_0 + a_1 + a_2 > 0$,
3. $a_0 - a_1 + a_2 > 0$.

As with the previous case, the fourth condition of Theorem 4.3.2 does not apply because $n < 3$.

Case n = 3 : Then, $D(z) = a_0 + a_1 z + a_2 z^2 + a_3 z^3$, and the conditions of Theorem 4.3.2 are:

1. $|a_0| < a_3$,
2. $a_0 + a_1 + a_2 + a_3 > 0$,
3. $-a_0 + a_1 - a_2 + a_3 > 0$.
4. To easily express this fourth condition, we establish the following table:

a_0			a_1			a_2	a_3
a_3			a_2			a_1	a_0
$a_{1,0} =$	a_0	a_3	$a_{1,1} =$	a_0	a_2	$a_{1,2} =$	a_0 a_1
	a_3	a_0		a_3	a_1		a_3 a_2

The first and second rows of this table are constructed using the coefficients of *D(z)* in both ascending and descending orders of *z*'s powers.

Subsequently, the third row contains only three coefficients, and each coefficient $a_{1,j}$ is calculated as the determinant of the matrix constructed with the first and the $(n-j+1)$-th columns of the two previous rows. Therefore, condition 4 can be written as $\left|a_0^2 - a_3^2\right| > \left|a_0 a_2 - a_3 a_1\right|$, which is equivalent to: $a_3^2 - a_0^2 > \left|a_0 a_2 - a_3 a_1\right|$.

Case n = 4: Then, $D(z) = a_0 + a_1 z + a_2 z^2 + a_3 z^3 + a_4 z^4$ and the conditions of

Theorem 4.3.2 are:

1. $|a_0| < a_4$,
2. $a_0 + a_1 + a_2 + a_3 + a_4 > 0$,

a_0	a_1	a_2	a_3	a_4
a_4	a_3	a_2	a_1	a_0
$a_{1,0} = \begin{vmatrix} a_0 & a_4 \\ a_4 & a_0 \end{vmatrix}$	$a_{1,1} = \begin{vmatrix} a_0 & a_3 \\ a_4 & a_1 \end{vmatrix}$	$a_{1,2} = \begin{vmatrix} a_0 & a_2 \\ a_4 & a_2 \end{vmatrix}$	$a_{1,3} = \begin{vmatrix} a_0 & a_1 \\ a_4 & a_3 \end{vmatrix}$	
$a_{1,3}$	$a_{1,2}$	$a_{1,1}$	$a_{1,0}$	
$a_{2,0} = \begin{vmatrix} a_{1,0} & a_{1,3} \\ a_{1,3} & a_{1,0} \end{vmatrix}$	$a_{2,1} = \begin{vmatrix} a_{1,0} & a_{1,2} \\ a_{1,3} & a_{1,1} \end{vmatrix}$	$a_{2,2} = \begin{vmatrix} a_{1,0} & a_{1,1} \\ a_{1,3} & a_{1,2} \end{vmatrix}$		

Tab. 4.1 – Table for Jury's Criterion *(n = 4)*

3. $a_0 - a_1 + a_2 - a_3 + a_4 > 0$.

4. We construct Table 5.1.

The first three rows of this table are constructed as in the previous case. The fourth row is constructed from the calculated values $a_{1,j}$ in decreasing order of j. Subsequently, the fifth row contains only three coefficients, and each coefficient $a_{2,j}$ is calculated as the determinant of the matrix constructed with the first and the $(n-j+1)$-th columns of the two previous rows. Therefore, condition 4 can be written as:

$$|a_{1,0}| > |a_{1,3}| \Leftrightarrow a_4^2 - a_0^2 > |a_0 a_3 - a_1 a_4|$$

Jury's Criterion: Presented for the case $n = 2$

Let's reconsider the characteristic equation (6.2) for the case where $n = 2$:

$$c_1 z^2 + c_1 z + c_0 = 0 \tag{4.10}$$

Equation (4.10) is normalized with respect to the coefficient of the highest degree, i.e., dividing the left-hand side by c_2:

$$z^2 + \tilde{c}_1 z + \tilde{c}_0 = 0 \tag{4.11}$$

where:

$$\tilde{c}_1 = \frac{c_1}{c_2} \tag{4.12}$$

$$\tilde{c}_0 = \frac{c_0}{c_2} \tag{4.13}$$

***Definition**:*

The Jury polynomial is written as:

$$J(z) = z^2 + \tilde{c}_1 z + \tilde{c}_0 \tag{4.14}$$

Jury's criterion for a second-order system is stated as follows:

Property:

The second-order system is stable if and only if the Jury polynomial (equation (4.14)) satisfies the following three conditions:

- $J(0) < 1 \;\Rightarrow\; \tilde{c}_0 < 1$
- $J(1) > 0 \;\Rightarrow\; 1 + \tilde{c}_1 + \tilde{c}_0 > 0$
- $J(-1) > 0 \;\Rightarrow\; 1 - \tilde{c}_1 + \tilde{c}_0 > 0$

4.3.3 Routh-Hurwitz Criterion

Transformation to w:

In order to apply the same stability analysis tools for sampled systems as for continuous systems, we seek a conformal transformation that maps every complex number inside the unit circle to a unique complex number in the left half-plane of the complex plane.

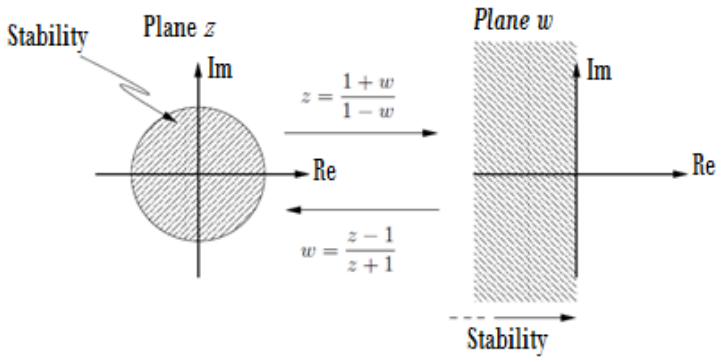


Fig. 4.4 - Transformation into w plan.

For the transition from the z plane to the w plane, this transformation is defined by:

$$z = \frac{1+w}{1-w} \tag{4.15}$$

and for the inverse transition (from the w) plane to the z) plane):

$$w = \frac{z-1}{z+1} \tag{4.16}$$

The main advantage of this transformation is that it preserves the structure of rational fractions of transfer functions. Indeed, taking the previously stated characteristic equation (equation (4.2)) and using (4.15), we get:

$$c_n z^n + c_{n-1} z^{n-1} + \ldots + c_1 z + c_0 = 0 \tag{4.17}$$

$$c_n \left(\frac{1+w}{1-w}\right)^n + c_{n-1} \left(\frac{1+w}{1-w}\right)^{n-1} + \ldots + c_1 \left(\frac{1+w}{1-w}\right) + c_0 = 0 \tag{4.18}$$

which, after expansion, yields the characteristic equation in w:

$$\alpha_n w^n + \alpha_{n-1} w^{n-1} + \ldots + \alpha_1 w + \alpha_0 = 0 \tag{4.19}$$

where the real coefficients α_k, $k = [0,1,\ldots,n]$ depend on the coefficients c_k of the characteristic equation in z.

It is then possible to apply the algebraic criterion of Routh, well known for continuous systems.

Routh-Hurwitz Criterion:

It consists, initially, in forming the following table from the coefficients of the characteristic equation in w (4.19):

Given by the Characteristic equation.	w^n	α_n	α_{n-2}	α_{n-4}	α_{n-6}	…
	w^{n-1}	α_{n-1}	α_{n-3}	α_{n-5}	…	
To fill	w^{n-2}	x_{n-2}	x_{n-3}	x_{n-4}	…	…
	w^{n-3}	y_{n-3}	⋮	⋮		
	⋮	⋮	⋮			
	⋮	⋮				
		$\underbrace{\qquad\qquad}_{valeurs\,de\,même\,signe}$				

$$y_{n-3} = \frac{x_{n-2}\alpha_{n-3} - x_{n-3}\alpha_{n-1}}{x_{n-2}}$$

Given by the char. Eq

$$x_{n-2} = \frac{\alpha_{n-1}\alpha_{n-2} - \alpha_n\alpha_{n-3}}{\alpha_{n-1}}$$

$$x_{n-3} = \frac{\alpha_{n-1}\alpha_{n-4} - \alpha_n\alpha_{n-5}}{\alpha_{n-1}}$$

$$x_{n-3} = \frac{\alpha_{n-1}\alpha_{n-6} - \alpha_n\alpha_{n-7}}{\alpha_{n-1}}$$

$$x_{n-2} = \ldots$$

and the next row:

$$y_{n-3} = \frac{x_{n-2}\alpha_{n-3} - x_{n-3}\alpha_{n-1}}{x_{n-2}}$$

$$y_{n-4} = \ldots$$

Properties:

- ***A Necessary and Sufficient Condition:*** for all roots of the characteristic equation in w to have negative real parts is that the coefficients of the first column of the Routh table are all of the same sign. The number of unstable roots (i.e., with positive real parts) is equal to the number of sign changes in the first column.
- ***A Necessary Condition for all roots***: of the characteristic equation in w to have negative real parts, but not sufficient, is that all coefficients of this polynomial α_k are of the same sign.

Remarks:

- This criterion works regardless of the system's order.

- Such a criterion provides no indication of stability degree like gain margin or phase margin.

- It does not apply to systems containing a delay.

- One way to verify that the Routh table is correctly constructed is to check that the term in the last row of the first column is none other than the coefficient α_0.

Example: Consider the following example:

$$A(p) = \frac{A_0}{p + a} \tag{4.20}$$

whose characteristic equation in z is:

$$(z - e^{-aT}) + \frac{A_0}{a}(1 - e^{-aT}) = 0 \tag{4.21}$$

- Calculate the characteristic equation in w.

- Construct the Routh table.

- Deduce the stability conditions.

4.4 Geometric Criteria

Geometric criteria are commonly used in the study of continuous control systems stability because, in addition to the information "stable" or "unstable", they provide a measure of the stability degree of control systems through gain margin and phase margin. These same criteria are applicable to sampled control systems with some precautions. For these, the poles of the closed-loop transfer function are the roots of a characteristic equation that can be written as:

$$1 + kH(z) = 0 \tag{4.22}$$

where k is a positive scalar.

4.4.1 Nyquist Criterion

The Nyquist criterion allows determining whether the roots of an expression of the form (4.22) with $k = 1$ fixed, have negative real parts by reasoning on the Nyquist plot of the sampled transfer function.

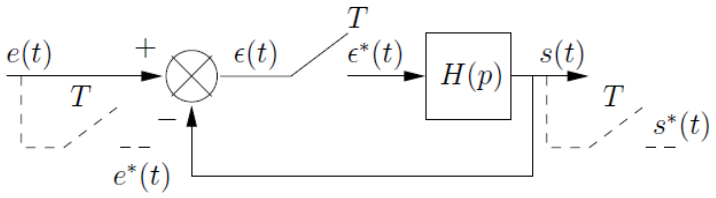


Fig. 4.5 – Sampled System

Method for Plotting the Nyquist Locus for Sampled-Data Systems:

For sampled-data systems, the Nyquist locus can be obtained in two different ways:

- Starting from the transfer function $H(z)$ and replacing z with e^{jwT}. Then, calculate the resulting complex number by evaluating $H(e^{jwT})$. In this case, the frequency w varies from 0 to w_N, the Nyquist frequency where $w_N = 2\pi f_N = \frac{\pi}{T}$ (see Chapter 2).

- Starting from the transfer function in w of $H(z)$. Let's call this transfer function $\tilde{H}(w)$. We have the relation:

$$\tilde{H}(w) = H(z) \tag{4.23}$$

where

$$z = \frac{w+1}{w-1} \tag{4.24}$$

and

$$w = \frac{z-1}{z+1} \tag{4.25}$$

Replacing z with e^{jwT}, it is shown that:

$$w = \tan\left(\frac{wT}{2}\right) \tag{4.26}$$

Now, when w varies from 0 to $\frac{\pi}{T}$, $\tan\left(\frac{wT}{2}\right)$ varies from 0 to $+\infty$. We then define:

$$\tilde{w} = \tan\left(\frac{wT}{2}\right) \tag{4.27}$$

we see that by setting $w = j\tilde{w}$, where $w = j\tilde{w}$ varies from 0 to $+\infty$. This change of variable is analogous to the case of continuous systems when plotting their Nyquist locus.

The second approach is generally preferred because, practically, once $\tilde{H}(w)$ is calculated, the procedure for plotting the Nyquist locus is identical to that of continuous systems.

Note: *The relation (4.27) shows that the variable $\tilde{w}$ does not have the physical significance of the frequency w. When choosing the second approach, one must ensure to make the inverse correspondence:*

$$w = \frac{2}{T}\arctan\tilde{w} \tag{4.28}$$

To plot the Nyquist locus of the transfer function $\tilde{H}(w)$, follow the steps below:

1. Create a table with a first column containing a number of significant $\tilde{w}$ values for the plot. In the second column, record the corresponding actual frequency w. In the third column, calculate the real part of $\tilde{H}(j\tilde{w})$ for each w value, and in the last column, calculate the imaginary part of $\tilde{H}(j\tilde{w})$.

$\tilde{w}$	w	$\mathrm{Re}\left(\tilde{H}(j\tilde{w})\right)$	$\mathrm{Im}\left(\tilde{H}(j\tilde{w})\right)$

TAB. 4.1 –

2. Plot on a graph representing the complex plane, the function $\mathrm{Im}\left(\tilde{H}(j\tilde{w})\right) = f\left(\mathrm{Re}\left(\tilde{H}(j\tilde{w})\right)\right)$ point by point, specifying each value of $\tilde{w}$, and connect these points in the order of increasing $\tilde{w}$.

3. Place the critical point with the coordinate $-1+0j$.

4.4.2 Simplified Nyquist Criterion (or Reverse Criterion)

Assumption: The system is stable in open-loop, meaning the transfer function $H(w)$ in w has no poles with strictly positive real parts.

In this case, the closed-loop system will be stable if, as we traverse the Nyquist locus in the increasing w direction, the critical point remains on the left side.

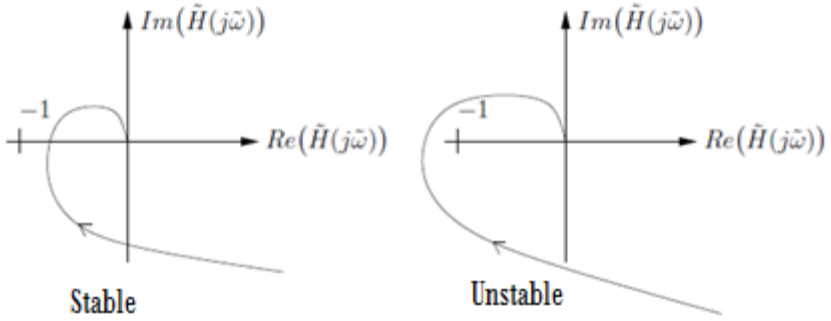


Fig. 4.6 – System Stability

The stability margins (gain margin and phase margin) can then be deduced from the Nyquist locus as in the case of continuous systems.

4.4.3 Bode Plot and Black-Nichols Plot

The Bode and Black-Nichols plots are obtained as in the continuous case when considering ◆ , the W-transform of ◆ . For plotting them, we can use the following table and proceed similarly to the Nyquist plot.

$\tilde{w}$	w	$\tilde{H}(j\tilde{w})$	$\arg\left(\tilde{H}(j\tilde{w})\right)$

TAB. 4.2 –

and proceed in the same manner as for the Nyquist plot. For the Bode plot, we can also make an asymptotic plot of the transfer function $\tilde{H}(w)$.

4.4.4 Root Locus (or Evans Root Locus)

This graph represents the evolution of the roots of equation (6.24) in the complex plane $Z = x + j\,y$ as k varies from 0 to $+\infty$. Thus, it is possible to visualize the trajectory of each pole from its position in the open-loop system. It is of interest because when $k = 1$, the roots of (6.24) are the poles of the closed-loop system.

There are rules, which we will not mention in these notes, for the asymptotic plotting of trajectories. We recommend the reader to refer to some bibliographic sources for more details.

4.4.5 Evans Root Locus

The Evans root locus constitutes a second geometric method for studying the stability of a control system.

Consider the feedback control system in Figure 4.6 with a proportional controller $C(z) = K_c$.

Then, the transfer function o f the open-loop system is:

$$F_{BO}(z) = K_c G(z) H(z) = \frac{N_{BO}(z)}{D_{BO}(z)} \qquad (4.29)$$

and the transfer function of the closed-loop system is:

$$F_{BO}(z) = \frac{K_c G(z)}{1 + K_c G(z) H(z)} = \frac{N_{BF}(z)}{D_{BF}(z)}. \qquad (4.30)$$

We recall that the poles of the control system are the roots of $D_{BF}(z)$, which means they are the roots of the characteristic equation:

$$1 + K_c G(z) H(z) + 1 + F_{BO}(z) = 0. \qquad (4.31)$$

It is noticeable that the roots of this equation depend on K_c. This means that the poles of the closed-loop system are dependent on the gain K_c.

Definition of Evans Root Locus:

It is interesting to study the position of the poles of the control system as the gain K_c varies. One can then synthesize the proportional controller by pole placement, meaning one can deduce the values of K_c for which the control system meets certain specifications (e.g., stable control, fast response, or sufficiently damped response).

Definition:

The curves described by the poles of the control system as K_c varies correspond to the curves described by the roots of the characteristic equation (of the control system) as K_c varies. The collection of these curves is called the root locus or Evans root locus.

It is possible to approximately plot the location of the poles of the closed-loop transfer function as a function of the gain K_c using simple rules. These rules are presented below.

Rules for Constructing the Evans Root Locus:

The rules for constructing the root locus are based solely on the open-loop transfer function $F_{BO}(z)$, which can be factored as:

$$F_{BO}(z) = K_c G(z) H(z) = \underbrace{K_c K_G}_{K} \frac{\prod_{i=1}^{m} (z - z_i)}{\prod_{j=1}^{n} (z - p_j)} \tag{4.32}$$

where z_i for $i = 1,...,m$ are the zeros and p_j for $j = 1,...,n$ are the poles of the open loop.

Subsequently, we assume that $K \in R_+$. Then, the Evans root locus represents the set of points in the convex plane defined by the roots of $1+F_{BO}(z)=0$ as K varies from 0 to ∞. This equation can be rewritten as:

$$\frac{\prod_{i=1}^{m}(z-z_i)}{\prod_{j=1}^{n}(z-p_j)} = -\frac{1}{K} \tag{4.33}$$

and it is also called the characteristic equation of the Evans root locus.

According to equation (4.33), a point *M* with the complex coordinate z_M belongs to the root locus if and only if the following conditions are satisfied:

1. $\frac{\prod_{i=1}^{m}(z_M-z_i)}{\prod_{j=1}^{n}(z_M-p_j)} = \frac{1}{K}$ Called the modulus condition;

2. $\sum_{i=1}^{m}\arg(z_M-z_i)-\sum_{j=1}^{n}\arg(z_M-p_i)=\pi(1+2\lambda)$ With $\lambda \in N$ being any angle, called the angle condition.

The rules for constructing the root locus are stated subsequently. Note that, in applying these rules, the zeros and poles are counted with their multiplicity order.

Rule 1:

Number of locus branches: *n* branches; The number of branches is identical to the number of poles of the open-loop system.

Starting points: The *n* branches start, for $K=0$, from the n poles $\{p_i\}$ of $F_{BO}(z)$

Ending points: The n branches end, for $K \to \infty$, at the *m* zeros $\{z_i\}$ and the $n-m$ infinite zeros of $F_{BO}(z)$.

Therefore, the locus has $n-m$ branches going to infinity.

Rule 2: The locus of roots is symmetrical with respect to the real axis.

Rule 3: Branches of the locus belonging to the real axis: A point M on the real axis belongs to the locus if the number of real poles and zeros of the open-loop system, counted with their multiplicity order, and located to the right of point M, is odd.

Rule 4: Asymptotes of branches at infinity: The $n-m$ asymptotes of branches going to infinity make angles with the real axis as follows:

$$\alpha_\lambda = \frac{(2\lambda+1)}{n-m}, \quad \lambda = 0,1,...,(n-m-1).$$

These $(n-m)$ asymptotes intersect the real axis at a single point with abscissa:

$$\sigma_a = \frac{\sum_{j=1}^{n} p_j - \sum_{i=1}^{m} z_j}{n-m}.$$

Rule 5: Separation points: correspond to the intersection of the locus with the real axis; A separation point on the real axis indicates the existence of a multiple real root that has the property of nullifying the derivative of $F_{BO}(z)$:

$$\frac{d\,F_{BO}(z)}{dz} = 0.$$

Note that this condition is a necessary but not sufficient condition. Therefore, to find a separation point, it is necessary to solve the previous equation and then verify that the solution(s) of this equation belong to the locus of roots.

Angle of branches at the separation point: if N branches of the Evans locus intersect at a separation point, then the angle between two adjacent half-branches is equal to $\frac{\pi}{N}$.

Rule 6: In some cases, for example when the open-loop transfer function contains complex conjugate zeros or poles, it is necessary to know the angle of departure or arrival of a branch in order to correctly construct the root locus. These angles are determined using the angle condition.

Angle of departure of a branch: let p_k be a pole of multiplicity n_k, then the n_k branches starting from p_k make angles β_k with respect to the horizontal, and

$$\beta_k = \frac{1}{n_k}\left(\sum_{i=1}^{m}\arg(p_k - z_i) - \sum_{j=1, j\neq k}^{n}\arg(p_k - z_i) - \pi(1+2\lambda)\right), \quad \lambda = 0,1,...,(n_k - 1).$$

Angle of arrival of a branch: let z_k be a zero of multiplicity m_k, then the m_k branches arriving at z_k make angles γ_k with respect to the horizontal.

Rule 7: Graduation of the locus in terms of K: The value of K at any point M with the complex coordinate z_M of the Evans locus is calculated using the characteristic equation or the modulus condition.

Note:

The intersections of the Evans locus with the unit circle can be determined using the Jury criterion. Determining these intersection points allows finding the value of K at the stability limit or the values of K for which the control systems remain stable.

Example:

Consider the open-loop transfer function:

$$F_{BO}(z) = K\frac{z + 0{,}5}{(z-1)(z-0{,}5)}.$$

The rules of the root locus apply as follows.

Rule 1:

Number of branches: $n = 2$ branches;

Starting points: the poles of the open-loop system $\{1;0,5\};$;

Ending points: the zero of the open-loop system $\{-0,5\}$ and the infinite zeros $\Rightarrow 1$ branch at infinity.

Rule 3: Branches of the locus belonging to the real axis: any point M with an abscissa z_M such that $z_M \in [0,51] \cup [-\infty\ -0,5]$.

Rule 4: Asymptote of the branch at infinity: $\alpha_0 = \pi$.

Rule 5: Separation points:

$$\frac{dF_{BO}(z)}{dz}. = 0 \Rightarrow z_1 = 0,72\ et\ z_2 = -1,72$$

The angles of the branches at the separation points are $\pm\frac{\pi}{2}$. The root locus has been plotted in the following figure:

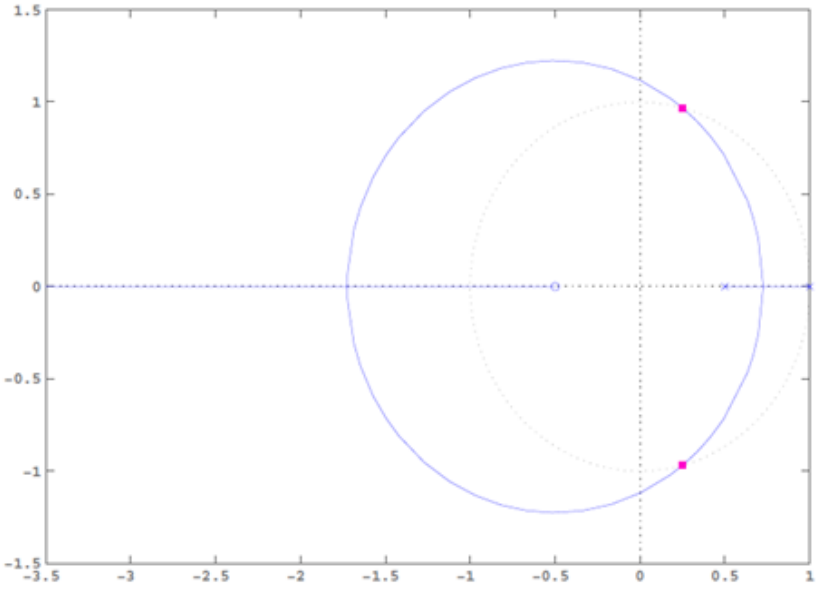


Fig. 4.6 – Root locus.

4.5 Conclusion

In conclusion, the analysis of sampled systems plays a critical role in the field of digital control systems. This chapter has provided an extensive examination of the stability of sampled systems, utilizing various criteria and methods to assess and ensure stability. Key insights include the importance of the Z-transform for analyzing discrete-time systems, the use of algebraic criteria such as the Schur-Cohn and Jury criteria for stability assessment, and the application of geometric criteria like the Nyquist criterion for evaluating the stability margins.

Understanding these concepts is essential for designing robust and efficient digital control systems. By ensuring stability, we can enhance the performance and reliability of these systems, making them suitable for a wide range of industrial applications. The methods and criteria discussed provide a comprehensive toolkit for engineers and practitioners to analyze, design, and improve digital control systems effectively. This foundational knowledge is crucial for anyone involved in the field of digital control and automation, paving the way for more advanced studies and practical implementations.

Chapter 5. Synthesis of Sampled-Data Systems

5.1 Introduction

This chapter focuses on the control approaches of discrete-time control systems and represents the most important step that requires particular attention. Indeed, all the techniques presented in the previous chapters must lead to the expertise related to the selection and calculation of controllers capable of providing the desired performances to a discrete process. These digital controllers, in whatever form they may appear, are characterized by a discrete output $U(z)$ called the control law, which drives the process and conditions its behavior to satisfy these performances. Different control approaches are presented in this chapter to enable the compensation of discrete processes in this regard.

5.2 Objective

The control of discrete-time systems is the fundamental objective of any study aimed at imposing performances that the controlled system must satisfy. This is why we have extensively detailed this chapter where we have presented various techniques for the implementation and calculation of digital controllers. These PID-type controllers, or RST-type controllers, widely used elsewhere, have been the subject of detailed study. They will subsequently need to be translated into control algorithms (not covered in these notes) through computers to drive the studied process. We have presented detailed examples of discrete process control and used MATLAB software to perform calculations or plot curves, recalling each time the syntax used to do so.

Definition of Speed:

Speed is related to the response time, and therefore to the bandwidth. This is itself linked to the dominant time constant or the natural frequency ω_n.

5.3 Static Accuracy of Sampled-Data Control Systems

The control error is defined as the difference between the reference and the controlled variable. The analysis of the error in steady-state is very important because it gives us a measure of the quality of control in terms of static accuracy.

For sampled-data systems, the same performance specifications as for continuous-time systems are defined. This includes the precision of the systems, which is always defined here by the notions of position, velocity, and acceleration errors.

The error $\varepsilon(k)$ in the time domain is obtained only in steady-state by means of the final value theorem: $\lim_{k\to\infty} \varepsilon(k) = \lim_{z\to 1}(z-1)\varepsilon(z)$.

5.3.1 Study of Accuracy:

In addition to stability, a controlled system must have other qualities to ensure satisfactory performance. In this chapter, accuracy is studied. The closed-loop systems considered here are assumed to be stable.

Definition:

Accuracy is the ability of a system to track a specific set of references with an error acceptable according to the specifications.

Two types of accuracy are distinguished:

- Static accuracy, which characterizes, for a given input, the limit of the so-called static error after an infinite time, i.e., much longer than the duration of the system's free response. This is referred to as steady-state or established regime.

- Dynamic accuracy, which takes into account the process's evolution characteristics mainly during the transient regime, i.e., over a time horizon of the same order of magnitude as the duration of the system's free response.

5.3.2 Expression of the Error (Static Accuracy)

Consider the digitally controlled system shown in Figure 5.1.

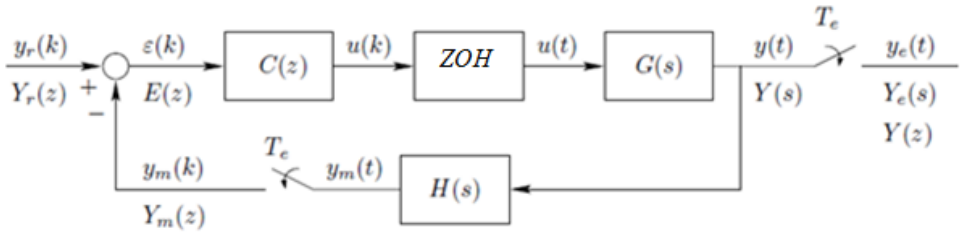


Fig. 5.1 – Closed-loop system with a digital controller

The error signal is:

$$\varepsilon(k) = y_r(k) - y_m(k). \tag{5.1}$$

Using the results from exercise 4.2.2, the z-transform of the error signal is given by:

$$E(k) = \frac{Y_r(z)}{1 + C(z)(1 - z^{-1})Z\left\{\dfrac{G(p)H(p)}{p}\right\}}. \tag{5.2}$$

5.3.3 General Expression of the Error

According to the final value theorem for discrete-time systems (see the properties of the Z-transform, section 3.3), we can calculate the expression of the steady-state error as:

$$\varepsilon_\infty = \lim_{z\to 1}(1 - z^{-1})E(z) = \lim_{z\to 1}\left(\frac{z-1}{z}\right)E(z) = \lim_{z\to 1}(z-1)E(z) \tag{5.3}$$

It should be noted that this relationship is valid provided that all poles of $E(z)$ are inside the unit circle.

Using equations (5.2) and (5.3), the expression of the error for the controlled system is:

$$\varepsilon_\infty = \lim_{z\to 1}\frac{z-1}{z}\frac{Y_r(z)}{1 + F_{BO}(z)}. \tag{5.4}$$

with the open-loop transfer function $F_{BO}(z) = C(z)(1-z^{-1})Z\left\{\frac{G(p)H(p)}{p}\right\}$.

Definition:

Consider the control system depicted in Figure 6.7.

- The static position error ε_p of system *S* is defined as the static error corresponding to a step input.

- The static velocity error ε_v of system *S* is defined as the static error corresponding to a ramp input.

- The static acceleration error ε_a of system *S* is defined as the static error corresponding to a parabolic input.

5.3.4 System Class

In the case of a discrete-time system, integration is characterized by the presence of a pole at $z = e^0 = 1$. Thus, if the sum of the coefficients of the denominator of the transfer function of the considered system is zero, then the system contains at least one integrator.

Now let's highlight the pole $z = 1$ of multiplicity c of the open-loop transfer function:

With $A(z)$, $B(z)$ polynomials of appropriate degrees. Let K be the constant defined by $\frac{A(1)}{B(1)}$. The integer c is called the class of the open-loop system.

The precision of the controlled system can be systematically calculated using (5.4) for different types of input:

Step Input: In this case, the static error is also called zero steady-state error.

Then, the input is written as:

$$Y_r(z) = \frac{E_0 z}{z-1}$$

and thus

$$\varepsilon_\infty = \lim_{z\to 1} \frac{E_0}{1+F_{BO}(z)} = \begin{cases} \frac{E_0}{1+K} & si \quad c = 0 \\ 0 & si\ c > 0 \end{cases}$$

Ramp Input: The input is written as:

$$Y_r(z) = \frac{V_0 z}{(z-1)^2}$$

and thus

$$\varepsilon_\infty = \lim_{z\to 1} \frac{V_0}{(z-1)(1+F_{BO}(z))} = \begin{cases} \infty & si \quad c = 0 \\ \frac{V_0}{K} & si \quad c = 1 \\ 0 & si \quad c > 1 \end{cases}$$

In this case, the static error is called first-order steady-state error.

These results can be extended to the case of polynomial inputs of degree greater than or equal to two. Table 5.1 summarizes the results for polynomial inputs of degree less than or equal to two.

Remark: From Table 5.1, we can deduce that the static precision with respect to the reference depends on the class of the open-loop system, i.e., the number of integrators in the open-loop. A zero steady-state error of order *n* can be canceled if and only if the class of the open-loop is at least $n+1$.

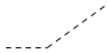
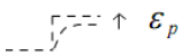
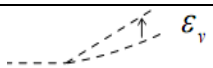
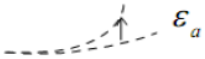
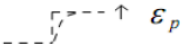
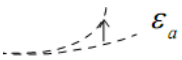
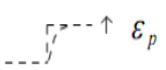
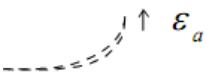
Input	***Static Position Error*** ε_p ***amplitude step*** **(step** (E_0)**)**  $Y_r(z) = E_0 \frac{z}{z-1}$	***Static Speed Error*** ε_v ***speed step*** **(ramp** $(E_0 kT)$**)**  $Y_r(z) = V_0 \frac{z}{(z-1)^2}$	***Static acceleration error*** ε_a ***acceleration step*** **(parable** $\left(\frac{1}{2} W_0 k^2 T^2\right)$**)**  $Y_r(z) = W_0 \frac{z(z+1)}{(z-1)^3}$
c = 0	 $\varepsilon_p = \frac{E_0}{1+KT(1)}$	 □	 □
c = 1	 0	 $\varepsilon_V = \frac{V_0 T}{KT(1)}$	 □
c = 2	 0	 ***0***	 $\varepsilon_V = \frac{W_0 T^2}{KT(1)}$
c > 2	0	***0***	***0***

Table 5.1 - Precision of digital control systems, according to their class

If *T(z)* is the transfer function in open-loop unity feedback;

then: $\varepsilon(z) = \frac{E(z)}{1+T(z)} = \frac{E(z)}{1+k\frac{1}{(z-1)^m}z^{-k}\frac{N(z)}{D(z)}}$ (Standard form),

This is the system's class.

Good precision is achieved through digital integrators that need to be introduced into the open loop via the controller.

5.4 PID Control

Steps of a digital control by computer of a sampled continuous process – blocked:

- Selection of the sampling period T and obtaining, by calculation or identification, the recurrence equation or transfer function of the digital process to be regulated,

- Selection of the desired digital model in closed-loop after correction; guided by the analysis and expression of the specifications,

- Design of the necessary digital controller (calculation of controller parameters),

- The controller, usually calculated in the frequency domain (z-transform), leads, by returning to the time domain, to the control algorithm of the process.

It elaborates in real-time (causality of the system to be controlled) the control *u(k)* to be sent to the digital system under its control at each T-step.

From a formal point of view, digital control is identical to sampled control.

The synthesis consists of conforming to this imposed model, by determining a feasible recurrence uk:

- Causal (the control at order *k* should only depend on previous values of control u, setpoint e, and measurement *y*),

- Stable (poles inside the stable domain), ensuring smooth control and continuous output without hidden oscillations.

The behavior of a sampled (digital) controlled system is the same as that of a continuous controlled system; it must be robust with respect to stability, precision, and speed. However, synthesis or tuning differs due to:

- The constitution of the controller (programmable),

- and the sampling period.

Two types of tuning: either synthesis using continuous methods or so-called digital synthesis.

5.4.1 Synthesis of digital controllers by extension of analog controllers.

This synthesis is a commonly used approach in the industrial domain.

This approach assumes that the synthesis of an analog controller has been performed using methods for continuous systems. A digital algorithm is sought that closely approximates the analog controller, by approximating the Laplace variable p or on the poles and zeros of the transfer function of the analog controller.

- **Command by transposition of analog controllers**

This method exploits analog regulators whose parameters are already established (by various techniques) in a continuous control loop that provides the analog controlled process with the required performances according to specifications imposed by the user.

Starting from a continuous control loop, a digital control loop is established by discretizing the process and implementing the analog regulator, respecting discretization techniques. This results in a discrete control system as presented in figure 5.2.

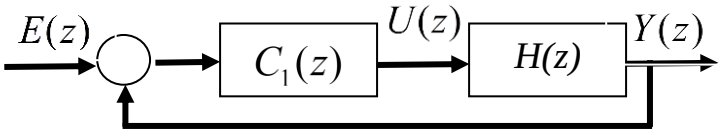


Fig.5.2: Closed-loop of a compensated discrete control system

Steps to follow:

- Choose the appropriate sampling period T_e for the studied process.

- Transpose the regulator $C(p)$ and the transfer function $H(p)$ into digital form, for example by replacing the variable P with $p = \frac{2(z-1)}{T_e(z+1)}$ (Tustin's approximation), or using the discretization techniques previously discussed.

- Calculate the closed-loop transfer function $H_f(z) = \frac{C_1(z)H(z)}{1 + C_1(z)H(z)}$.

- Check the stability (fundamental condition) of this discrete system.

- Establish the recurrence equation $y(k)$.

- By choosing an input $e(k) = 1$, calculate the samples $y(k)$ for $k = 0,1,...$ to compare its evolution to that of the uncorrected system in terms of precision, response time, etc.

- The control law $u(k)$, which must be put in the form of a programmable algorithm, directly follows from the recurrence equation.

Approximation of the variable p:

The principle of this approach is to deduce a discrete corrector $R_d(z)$ from the analog corrector $R_c(z)$ by choosing an approximation of the variable *p*, as shown in

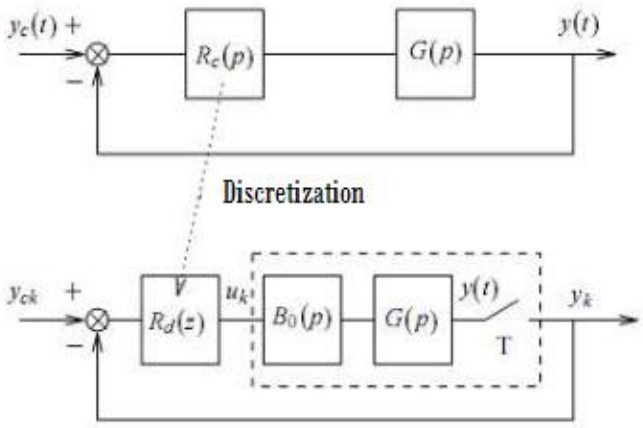


Fig.5.3: Discretization of an analog controller.

- Forward discretization: $p = \frac{z-1}{T}$

Results from approximating the derivative of a function between two sampling instants using the Euler method.

- Backward discretization: $p = \frac{z-1}{zT}$

Tustin's approximation: $p = \frac{2}{T}\frac{z-1}{z+1}$

5.4.2 Synthesis using continuous methods.

The principle of these methods is based on transforming the sampled system into a continuous system that is approximately equivalent using transformations. Once the synthesis using classical continuous methods is done, we return to the discrete regulator.

Examples of transformations:

Method	Principle	$G_n(p)$	$C(z)$
Delay of $\frac{T}{2}$	$B_0(p) \approx e^{-p\frac{T}{2}}$	$e^{-p\frac{T}{2}}G(p)$	$C\left(\frac{z-1}{T}\right)$
Linear approximation	$z \approx 1 - pT$ $P \approx \frac{z-1}{T}$	$G(1+pT)$	$C\left(\frac{z-1}{T}\right)$
Homographic transformation	$z \approx \frac{1+pT/2}{1-pT/2}$ $P \approx \frac{2}{T}\frac{(z-1)}{(z+1)}$	$G\left(\frac{1+pT/2}{1-pT/2}\right)$	$C\left(\frac{2}{T}\frac{(z-1)}{(z+1)}\right)$
Transformation to w-domain	$z = \frac{1+w}{1-w},$ $w = \frac{z-1}{z+1},$ $p = j\omega \Rightarrow w = jtg(\omega T/2) = jv$	$G\left(\frac{1+w}{1-w}\right)$	$C\left(\frac{z-1}{z+1}\right)$

a. Command by implementation of a naturally digital regulator

With this method, a discrete corrector $C(z)$ is directly calculated by imposing that the closed-loop discrete transfer function $H_f(z)$ of the compensated system be equal to a transfer function $G(z)$ \) chosen in advance, which satisfies the desired performance on the regulated system.

Therefore: $H_f(z) = \frac{C_1(z)H(z)}{1 + C_1(z)H(z)} = G(z)$

The expression of $C(z)$ is then deduced as:

$$C(z) = \frac{1}{H(z)} \frac{G(z)}{1 - C(z)} \tag{5.5}$$

If $H(z) = \frac{A(z)}{B(z)}$ and $G(z) = \frac{N(z)}{D(z)}$

The magnitudes of the zeros of $B(z)$ and the zeros of $D(z)$ must be less than one (to respect the stability condition), the expression of $C(z)$ becomes:

$$C(z) = \frac{B(z)}{A(z)} \frac{N(z)}{D(z) - N(z)}.$$

The stability condition must also be verified on this corrector.

According to the expression of the denominator of $C(z)$, this condition can only be ensured if:

The zeros of $A(z)$ also have magnitudes less than one.

The zeros of $D(z) - N(z)$ have magnitudes less than one as well.

This naturally limits the use of this method to a particular category of processes that meet conditions on the magnitudes of both their denominators and numerators.

Example 1:

Suppose we want to control a process represented by its discrete transfer function

$H(z) = \frac{A(z)}{B(z)}$ in open loop.

(All the magnitudes of the zeros of $B(z)$ must be less than one for stability.)

We implement a discrete regulator $C(z)$, and then obtain a closed-loop transfer function of the controlled process in the form:

$$\frac{C(z)H(z)}{1+C(z)H(z)}$$

If we do not want any overshoot in the response of the corrected system, it is necessary for $\frac{C(z)H(z)}{1+C(z)H(z)}$ to be identified with a function $G(z)$ of the form $G(z)=\frac{1-a}{z-a}$ representing a first-order discrete system with unit static gain. Then:

$$\frac{C(z)H(z)}{1+C(z)H(z)}=G(z)=\frac{1-a}{z-a}$$

From which we deduce the expression of $C(z)$:

$$C(z)=\frac{(1-a)B(z)}{(z-1)A(z)}$$

This imposes, for stability, that all the magnitudes of the zeros of $A(z)$ must also be less than 1.

Example 2:

Let the open-loop transfer function of a controlled system be:

$$H(z)=\frac{A(z)}{B(z)}=\frac{0{,}03(z+0.75)}{z^2-1{,}5z+0{,}5}$$

Calculate a digital regulator that provides the corrected system with:

♦ A second-order closed-loop behavior characterized by: a damping coefficient $\xi=0{,}6$ and a natural frequency $\omega_0=3\,rad/s$.

♦ A static error $\varepsilon(\infty)=0$ and a velocity error $\varepsilon_v=0{,}2$.

Solution:

- First, we establish the expression of the closed-loop transfer function:

$$G(z)=\frac{N(z)}{D(z)}=\frac{N(z)}{(z-z_1)(z-z_2)}=\frac{az+b}{(z-z_1)(z-z_2)}$$

Where z_1 and z_2 are the imposed roots of the chosen second order, with expressions: $z_{1/2}=e^{-\xi\omega_0 T_e \pm j\omega_0 T_e\sqrt{1-\xi^2}}$ for $T_e=0{,}2\,s$, we have $z_{1/2}=0{,}526\pm j\,0{,}360$.

To calculate a and b, we exploit the conditions $\varepsilon(\infty)=0$ and $\varepsilon_v=0{,}2$:

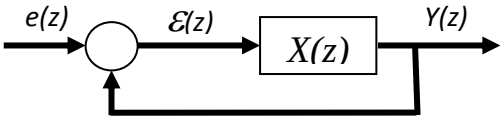


Fig. 5.4 - Closed-loop of a discrete control system

Let's express the error $Z[\varepsilon(t)]$: According to Fig. 37 $\varepsilon(z)=\frac{1}{1+X(z)}e(z)$, knowing that $\frac{y(z)}{e(z)}=\frac{X(z)}{1+X(z)}=G(z)$, then $X(z)=\frac{D(z)}{1-G(z)}$ and therefore $\varepsilon(z)=[1-G(z)]e(z)$.

- For a unit step input $e(z)=\frac{z}{z-1}$, the static error is thus

$$\varepsilon(z)=[1-G(z)]\frac{z}{z-1}$$

Since $\varepsilon(\infty)=\lim_{k\to\infty}\varepsilon(k)=\lim_{z\to 1}\left(1-z^{-1}\right)\varepsilon(z)$, then $\varepsilon(\infty)=[1-G(1)]$, however $\varepsilon(\infty)=0$, by assumption $G(1)=1$,

$$G(1)=\frac{a+b}{1+(z_1+z_2)+(z_1z_2)}.$$

- The velocity error corresponds to a ramp input $e(z) = \frac{T_e z}{(z-1)^2}$.

$\varepsilon_v = \lim_{k\to\infty} \varepsilon_v(k) = \lim_{z\to 1}(1-z^{-1})[1-G(z)]\frac{T_e z}{(z-1)^2} = \frac{1}{K_v}$ Since $\varepsilon_v = 0{,}2$ Then $K_v = \frac{1}{0{,}2}$

We can also exploit L'hôpital's rule (or $T_e = 0{,}2\,s$, by assumption):

$$\left[\frac{dG(z)}{dz}\right]_{z=1} = -\frac{1}{K_v T} = -\frac{1}{\frac{1}{0{,}2}0{,}2} = -1.$$

So, $\dot{G}(1) = -1 = \frac{a(1+z_1+z_2+z_1z_2)-(2+z_1+z_2)+(a+b)}{[1+(z_1+z_2)+(z_1z_2)]^2}$

In conclusion, $G(1) = 1$ and $\dot{G}(1) = -1$ allow us to calculate a and b, and the controller $C(z) = \frac{1}{H(z)}\frac{G(z)}{1-G(z)}$ is then fully defined.

5.4.3 PID Digital Control

The PID controller is the most "popular" of controllers and is widely used, especially in analog control. It is characterized by its three coefficients: proportional K_P, integral K_I, and derivative K_D, which appear in the time-based expression relating the controller's output to its input $\varepsilon(t)$:

$$u(t) = K_P\varepsilon(t) + K_I\int_0^t \varepsilon(\tau)d\tau + K_D\frac{d\varepsilon(t)}{dt}$$

The Laplace transform gives the following transfer function:

$$R(p) = \frac{U(p)}{\varepsilon(p)} = K_P + K_I\frac{1}{p} + K_D p = \frac{K_D p^2 + K_P p + K_I}{p}$$

Note that:

- K_P *and* K_I *improve the response time but may render the system less stable.*

- K_I *additionally eliminates static error but can generate oscillations harmful to stability.*

- K_D *slows down the response but can attenuate oscillations, thus making the system more stable.*

Determining the three coefficients K_P, K_I and K_D in analog studies involves various techniques, both theoretical (e.g., the Naslin method) and empirical (e.g., the Ziegler-Nichols method), extensively covered in basic control engineering literature.

To determine the PID coefficients, one can also conduct an open-loop study on the system corrected by a PID and require the PID numerator to eliminate the largest time constants of the open-loop transfer function of the process.

The coefficients can also be determined from a closed-loop study of the corrected system, where performance requirements such as stability or precision are imposed.

In all cases, a balance must be sought between speed, stability, and static error, while adhering to the conditions set forth in a user-defined specification.

Furthermore, the study of analog control systems regulated by a PID effectively utilizes the step response of the system to highlight:

- *Overshoot:* the deviation of the response from the setpoint in transient regime (which generally should not exceed 20%).

- *Rise Time:* Represents the time required for the system to transition from 10% to 90% of the setpoint and characterizes the system's dynamics.

- *Settling Time:* The time it takes for a system to reach 5% (or 2%) of its setpoint in steady-state.

This study leads to determining the PID coefficients in analog according to the satisfaction of the required system performance.

Subsequently, the analog PID thus obtained can be discretized to obtain its numerical expression, which is then implemented in the control loop of a discrete system.

Note that there exists a direct method for calculating the coefficients of a digital PID, similar to the Ziegler-Nichols method in analog, which uses an empirical approach (based on experimentation), which we do not discuss here.

5.4.4 Numerical Expressions of the PID Controller

a. Approximation of the Integral and Derivative

Starting from an analog controller with the expression:

$u(t) = K_P\left[\varepsilon(t) + \frac{1}{T_i}\int_0^t \varepsilon(\tau)d\tau + T_d\frac{d\varepsilon(t)}{dt}\right]$ where T_i and T_d are its integral and derivative constants, we can discretize the control law $u(p)$ by evaluating it at the sampling instant T_s.

To simplify the notation, assuming a constant sampling period T_s, we replace each function $F(T_s k)$ with $F(k)$ or F_k.

$\varepsilon(p)$ is the error signal, $e(p)$ is the setpoint, and $y(p)$ is the output.

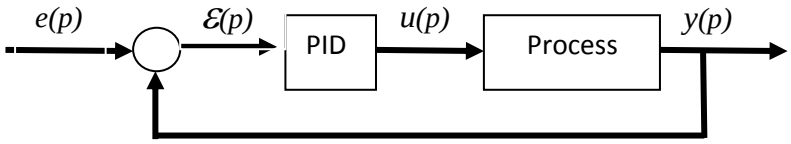


Fig. 5.5 - Analog control loop controlled by PID

To do this, we make the following substitutions:

$$\begin{cases} u(t) \rightarrow u(k), \quad \varepsilon(t) \rightarrow \vdots \varepsilon(k) \\ \dfrac{d}{dt}\varepsilon(t) \rightarrow \dfrac{\varepsilon(k)-\varepsilon(k-1)}{T_e} \end{cases} \qquad \textbf{(5.6)}$$

The integral expression of $\varepsilon(k)$ can be approximated by the sum of elementary rectangles $\varepsilon(l)T_e$ (see Figure 36)

$$\int_0^{kh} \varepsilon(\tau)d\tau \approx \sum_{l=0}^{k-1} \varepsilon(l)\,T_d = \varepsilon(0)T_d + \varepsilon(1)T_d + \ldots \qquad \textbf{(5.7)}$$

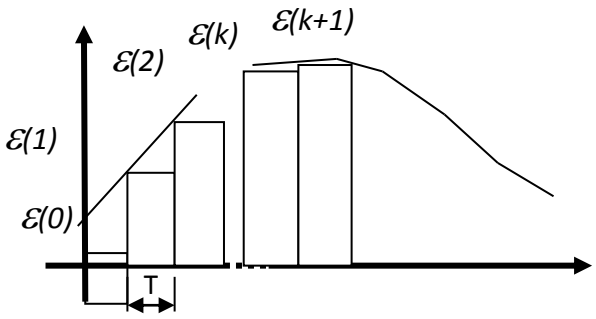


Fig. 5.6 - Approximation of an integral by Elementary Rectangles

In this case, the discrete control law is written as:

$$u(k) = K_p\left[\varepsilon(k) + \frac{1}{T_i}\sum_{l=0}^{k-1}\varepsilon(l)T_s + T_d\,\frac{\varepsilon(k)-\varepsilon(k-1)}{T_s}\right]$$

However, the form of the obtained control law above is not suitable for programming, due to the approximation of the integral.

Indeed, by summing up the input ε, we would need to store all past values of ε starting from $\sum_{l=0}^{k-1}\varepsilon(l)$.

To avoid this constraint, we explicitly express the term $u(k-1)$:

$$u(k-1) = K_p\left[\varepsilon(k-1) + \frac{1}{T_i}\sum_{l=0}^{k-2}\varepsilon(l)T_s + T_d\,\frac{\varepsilon(k-1)-\varepsilon(k-2)}{T_s}\right]$$

And then calculate the difference $u(k)-u(k-1)$, which yields:

$$u(k)-u(k-1) = K_p\left[\varepsilon(k)-\varepsilon(k-1)\right] + \frac{K_p}{T_i}\left[\sum_{l=0}^{k-1}\varepsilon(l)T_s - \sum_{l=0}^{k-2}\varepsilon(l)T_s\right] + K_pT_d\left[\frac{\varepsilon(k)-\varepsilon(k-1)}{T_s}\right]$$
$$- K_pT_d\left[\frac{\varepsilon(k-1)-\varepsilon(k-2)}{T_s}\right]$$

The integral expression then reduces to: $\frac{K_p}{T_i}\left[\sum_{l=0}^{k-1}\varepsilon(l)T_s - \sum_{l=0}^{k-2}\varepsilon(l)T_s\right] = \frac{K_p}{T_i}T_s\varepsilon(k-1)$

And the derivative expression to: $-K_pT_d\varepsilon(k) - \frac{2K_pT_d}{T_s}\varepsilon(k-1) + \frac{K_pT_d}{T_s}\varepsilon(k-2)$

By rearranging, we obtain the final expression of the PID control law:

$$u(k) = u(k-1) + b_0\varepsilon(k) + b_1\varepsilon(k-1) + b_2\varepsilon(k-2) \qquad (5.8)$$

b. Direct Discretization of the Analog PID Transfer Function

The transfer function being: $R(p) = K_p + K_i \frac{1}{p} + K_d p$

It can be written as:

$$R(p) = K\left(1 + \frac{1}{T_i}\frac{1}{p} + T_d p\right); \qquad \textbf{(5.9)}$$

To discretize $R(p)$, we can:

● Either directly exploit the table of transforms, yielding:

A : $$R(z) = K\left(1 + \frac{1}{T_i}\frac{z}{z-1} + T_d \frac{z-1}{z}\right); \qquad \textbf{(5.10)}$$

This is the simplest form in which the sampling period T_s does not appear.

• Or discretize:

- The integral with a sampling period T_s and a zero-order hold, in which case we get: $\frac{T_s}{T_i}\frac{z}{z-1}$

- The derivative term by approximating it as: $\frac{d\varepsilon(t)}{dt} \Rightarrow \frac{\varepsilon(t) - \varepsilon(t-1)}{T_s}$, thus :

$$\frac{T_d}{T_s}\left(1 - z^{-1}\right) = \frac{T_d}{T_s}\frac{z-1}{z}$$

And we obtain:

B: $$R(z) = K\left(1 + \frac{T_s}{T_i}\frac{z}{z-1} + \frac{T_d}{T_s}\frac{z-1}{z}\right); \qquad \textbf{(5.11)}$$

It is worth noting that the derived expression in analog $T_d p$ is not physically realizable, and for the practical implementation of the derivative term, we

multiply $T_d\,p$ by the expression $\dfrac{1}{1+\dfrac{T_d}{N}\,p}$ called the filtering term. (See appendix).

In this case, $T_d\,p$ is replaced by $\dfrac{T_d\,p}{1+\dfrac{T_d}{N}\,p}$, where N is a filtering constant between 3 and 20 (typically $N \approx 10$).

We then have an analog expression of the transfer function:

$R(p) = K\left(1+\dfrac{1}{T_i}\dfrac{1}{p}+\dfrac{T_d\,p}{1+\dfrac{T_d}{N}\,p}\right)$; that we are going to discretize.

Let's write:

$$S(p) = \frac{R(p)}{p} = K\left(\frac{1}{p}+\frac{1}{T_i}\frac{1}{p^2}+\frac{T_d}{1+\dfrac{T_d}{N}\,p}\right);$$

According to the table of transforms, we have $S(z)$:

$$S(z) = K\left(\frac{z}{z-1}+\frac{T_s}{T_i}\frac{z}{(z-1)^2}+N\frac{z}{z-e^{-N\frac{T_s}{T_d}}}\right);$$

As:

$$S(z) = \left[S(p)^*\right] = \left[\frac{R(p)}{p}\right]^* = \left[\frac{B_0(p)R(p)}{1-e^{-T_s p}}\right]^* = \frac{z}{z-1}R(z);$$

Then, the discrete expression $R(z)$ of the PID becomes:

$$\textbf{C :}\quad R(z) = K\left(1+\frac{T_s}{T_i}\frac{1}{(z-1)}+N\frac{(z-1)}{z-e^{-N\frac{T_s}{T_d}}}\right); \qquad \textbf{(5.12)}$$

c. Discretization of an Analog PID, using the Euler Approximation

In this case, we arrive at other expressions for $R(z)$:

• By replacing P with $\frac{z-1}{T_e}$ in $R(p)$, we obtain expression *D*:

$$\textbf{D :} \quad R(z)=K\left(1+\frac{T_s}{T_i}\frac{1}{(z-1)}+N\frac{(z-1)}{z-e^{-N\frac{T_s}{T_d}}}\right); \tag{5.13}$$

• By replacing P with $\frac{1}{T_s}\frac{z-1}{z}$ in $R(p)$, we obtain expression ***E***:

$$\textbf{E :} \; R(z)=K\left(1+\frac{T_s}{T_i}\frac{z}{(z-1)}+N\frac{(z-1)}{\left(1+N\frac{T_s}{T_d}\right)z-1}\right); \tag{5.14}$$

• By replacing P with $\frac{2}{T_s}\frac{z-1}{z+1}$ in $R(p)$, we obtain expression *F*:

$$\textbf{E :} \quad R(z)=K\left(1+\frac{T_s}{2T_i}\frac{z+1}{(z-1)}+N\frac{(z-1)}{\left(1+N\frac{T_s}{T_d}\right)z-\left(1-N\frac{T_s}{2T_d}\right)}\right); \tag{5.15}$$

These different variants of the expression of a digital PID are all equivalent. However, a standard form is generally used, which involves applying the Euler approximation $\frac{z-1}{T_s}$ to discretize the integral, and the approximation to discretize the derivative term.

In this case, the standard form of a digital PID (which we will use later) is:

$$R(z)=K\left(1+\frac{T_s}{T_i}\frac{1}{(z-1)}+N\frac{(z-1)}{\left(1+N\frac{T_s}{T_d}\right)z-1}\right); \tag{5.16}$$

d. The Digital PID in a Control Loop

As in an analog loop, the controller $R(z)$ is placed upstream of the discretized process that one wishes to regulate. $e(z)$ represents the error between the setpoint $Y_C(z)$ and the output $Y(z)$, and $u(z)$ is the output of $R(z)$.

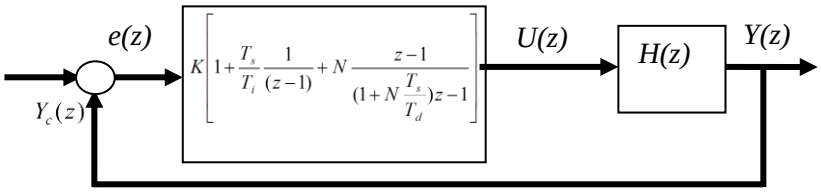


Fig. 5.7 - First configuration of the implementation of a Digital PID

Another implementation scheme where there is separation of the three actions is:

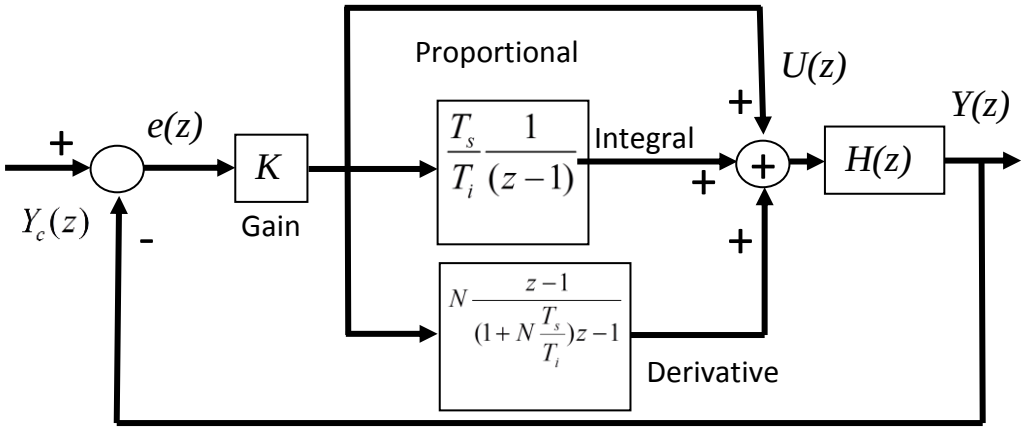


Fig. 5.8 - Second configuration of the implementation of a digital PID

To overcome some disadvantages related to phenomena of strong variations of $u(z)$, caused by the derivative, the derivative of the gain is avoided as explicitly shown in the diagram below.

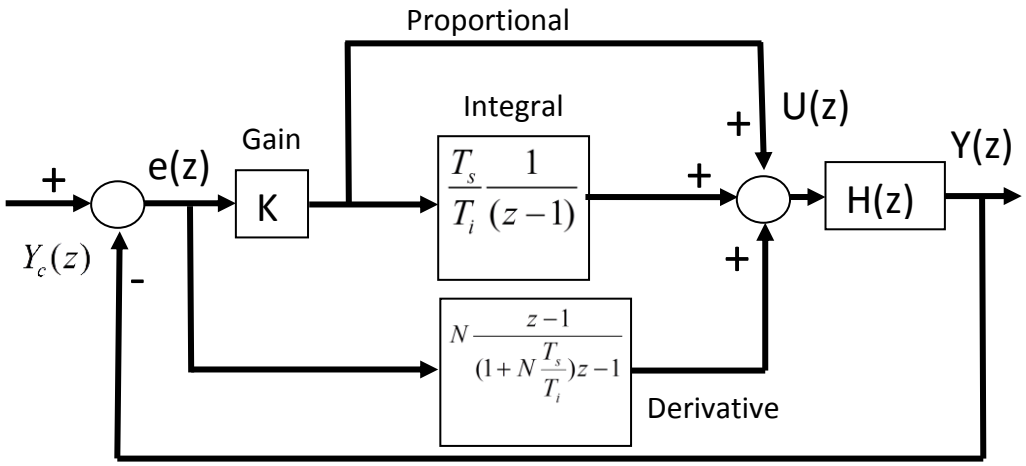
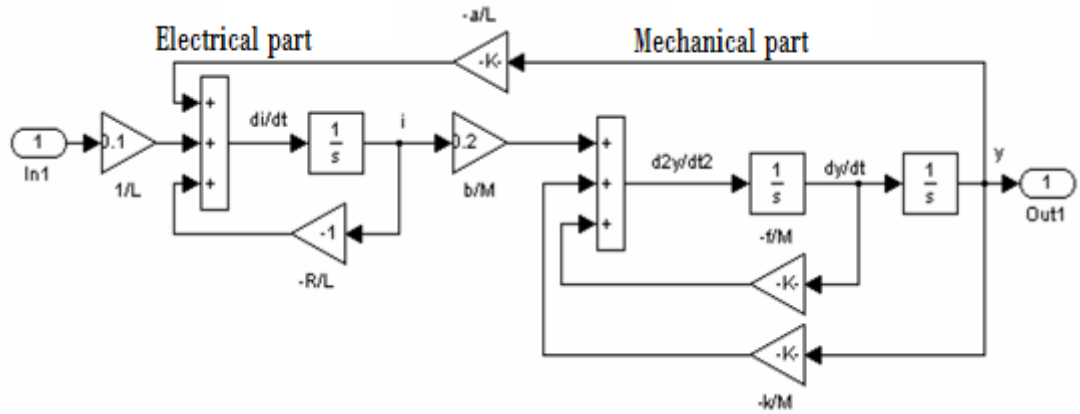


Fig. 5.9 - Third configuration of the implementation of a digital PID

Example of implementation of an analog and digital PID controller

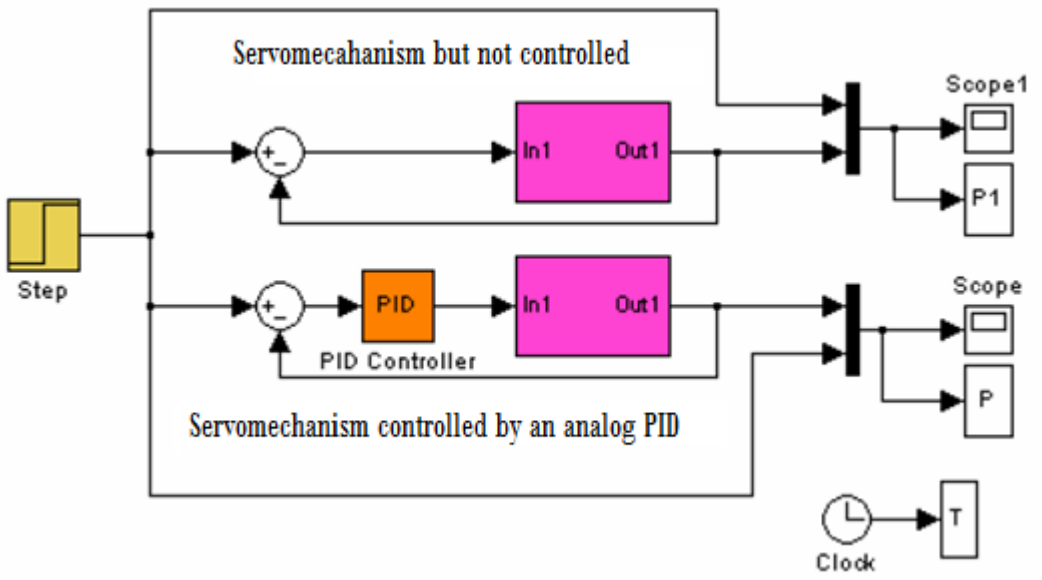
The following wiring diagram represents an electromechanical system.



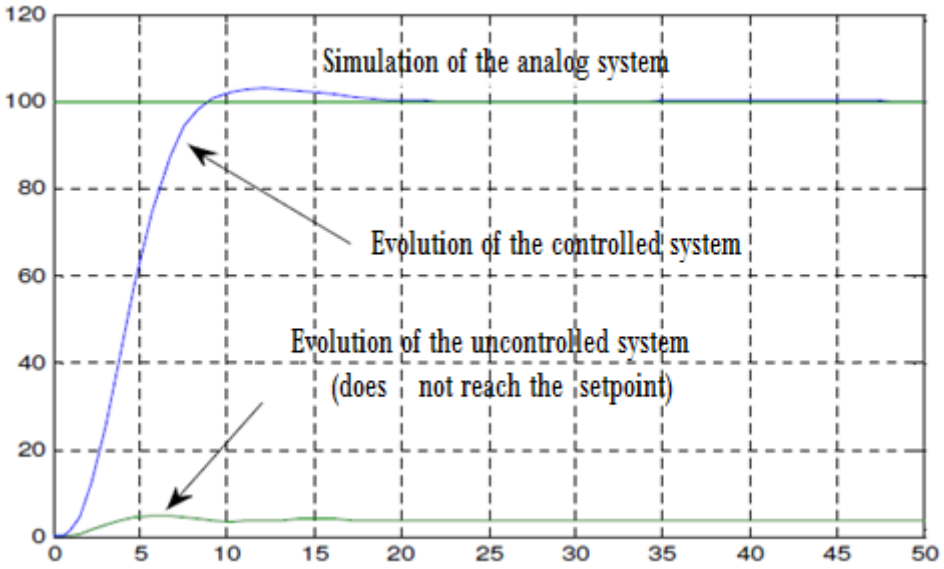
Established in the Simulink environment of Matlab.

A first numerical simulation is carried out on this analog controlled system (see diagram below) represented in an input-output box.

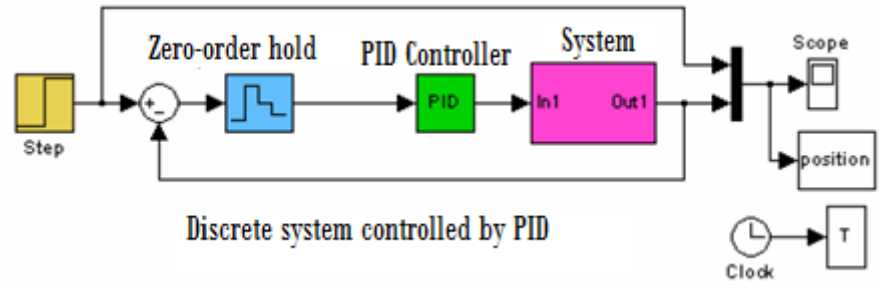
The simulation is performed simultaneously on the system without regulation and with regulation using a PID controller with judiciously chosen coefficients.



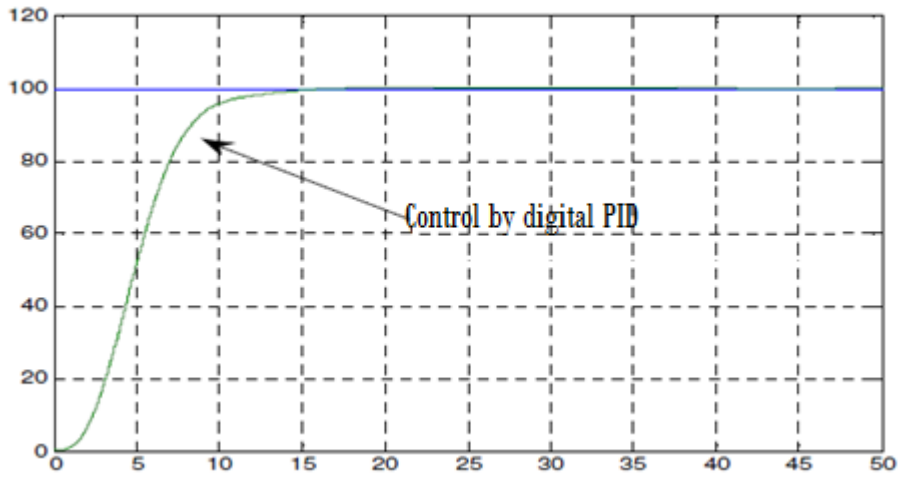
According to the curves obtained through this simulation, we observe that the output *y(t)* shows poor precision for the uncontrolled system, while for the system controlled by a PID, the output reaches its setpoint after 20 seconds and maintains it, with a transient overshoot of no more than 5%.



Then, we conduct a simulation of the discretized and controlled system (see diagram below), where a zero-order hold sampler samples the direct chain containing the PID and the process, with a properly chosen sampling period T_s.



We observe that the digital PID provides a better result on the output of the discrete system, as this output reaches its setpoint in 15 seconds and maintains it, without any overshoot.



5.5 Conclusion

In conclusion, the synthesis of sampled-data systems is a crucial aspect of digital control engineering, focusing on the development of discrete-time controllers to achieve desired system performance. This chapter has highlighted various control strategies, including PID and RST controllers, essential for managing discrete processes. We have examined the importance of static accuracy and the techniques used to ensure the precision and stability of sampled-data systems.

Key takeaways include the need for careful selection of the sampling period, the design of digital controllers to meet specific performance criteria, and the implementation of control laws through digital algorithms. The chapter has also provided detailed examples and practical insights into using MATLAB for calculations and plotting, reinforcing the theoretical concepts discussed.

Ultimately, mastering the synthesis of sampled-data systems equips engineers with the tools necessary to design effective and robust digital control systems, ensuring high performance and reliability in various industrial applications. This foundation is vital for further exploration and innovation in the field of digital control.

Chapter 6. RST Controller

6.1 Introduction:

Control by polynomial approach (RST Regulator)

[T : Tracking; S : Simplify; R : Return]

Even though 90% of regulation and control loops use a classical PID structure, and although PID is relatively easy to calculate, it is not always easy to achieve good performance on the process controlled by PID, especially if this process:

- Has a significant pure delay,

- Its dynamic characteristics vary during operation,

- Its order is higher than 2.

- Overshooting of the setpoint is not tolerated.

The RST regulator is used in a "transfer function" representation to allow the control of a system whose closed-loop transfer function $H_f(z)$ will be modified by the implementation of three polynomials $R(z)$, $S(z)$, and $T(z)$ arranged in a control loop as shown in Figure 6.1 to achieve a control law that takes into account the coefficients of these polynomials and satisfies the desired performance. Thus, the controlled system can:

- Manage the tracking dynamics (tracking) and disturbance rejection dynamics (regulation) independently.

- Specify independently the rise time and overshoot on the setpoint.

- Take into account the pure delay of the process.

- Ensure robust regulation with respect to:

  - Process variations,

  - Changes in setpoints...,

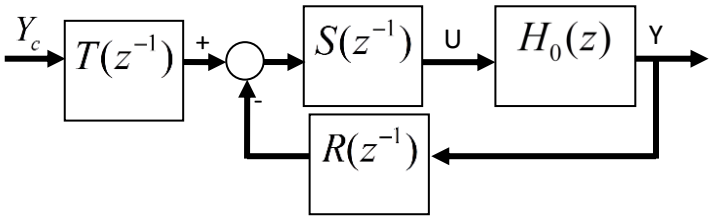


Fig. 6.1 - Block diagram of a control system

6.2 Data on the system to be controlled:

The discrete process is governed by an open-loop transfer function:

$$H_0(z) = \frac{B(z)}{A(z)}$$

Note that the transfer functions of the studied system can relate to either the correspondence regulation $H_m(z)$ (relative to a setpoint) or the maintenance regulation $G_p(z)$ (relative to a disturbance).

Correspondence:

$$H_m(z) = \frac{B_m(z)}{A_m(z)} = \frac{B(z)T(z)}{A(z)R(z) + B(z)S(z)} \tag{6.1}$$

Maintenance:

$$G_p(z) = \frac{B(z)R(z)}{A(z)R(z) + B(z)S(z)} \tag{6.2}$$

Here, we choose to focus solely on the case related to correspondence regulation, namely $H_m(z)$.

The control law is governed by the equation:

$$U(z) = \frac{T(z)}{R(z)} Y_C(z) - \frac{S(z)}{R(z)} Y(z) \tag{6.3}$$

6.3 Data on the reference model

The desired performance in tracking must be specified in a reference model, imposed by the user and which, in closed-loop, is expressed in the form of a rational fraction in *z*:

$$H_m(z) = \frac{B_m(z)}{A_m(z)} = \frac{Y(z)}{Y_C(z)}$$

which we identify as $\frac{B(z)T(z)}{A(z)R(z)+B(z)S(z)}$.

And which must be chosen very carefully:

Firstly, $H_m(z)$ must be chosen so that the static error $\varepsilon(\infty)=0$ is zero (In some study cases, a condition on the tracking error may also be imposed. In this case, the expression of $\varepsilon_v(\infty)$ in response to a discrete ramp input must be explicitly stated.

The static error arises from the expression: $\varepsilon(k) = y_c(k) - y(k)$, and we have

$$\varepsilon(\infty) = 0 = \lim_{k\to\infty}\left[y_C(k) - y(k)\right] = \lim_{z\to 1}(z-1)\left[Y_C(z) - Y(z)\right] \qquad (86)$$

where $Y(z) = Y_C(z)H_m(z)$

for a unit input $Y_C(z) = \frac{z}{z-1}$.

Replacing in (84), we find that $H_m(z)$ must satisfy:

$$H_m(1) = 1 \qquad (6.4)$$

The denominator $A_m(z)$ of $H_m(z)$ must have as zeros the values imposed by the user depending on the dynamics they have chosen.

The numerator $B_m(z)$ will contain the uncompensated zeros of the system as we will see later.

The implementation of the three polynomials $R(z)$, $S(z)$, and $T(z)$ allows us to calculate their coefficients in order to satisfy the performance criteria, which may include precision, overshoot rate, step response, settling time, or closed-loop bandwidth... based on the identification of the expression $\frac{B(z)T(z)}{A(z)R(z)+B(z)S(z)}$ to the reference model $H_m(z)=\frac{B_m(z)}{A_m(z)}$ which summarizing these performances.

We follow the steps to be placed. The conditions imposed on the degrees of the polynomials and the determination of the coefficients of these polynomials are then related to the choice of its poles and the targeted dynamics:

6.4 Design Synthesis of Polynomials RST

First, we deal with the open-loop transfer function of the initial system:

$$H_0(z)=\frac{B(z)}{A(z)}$$

We decompose $B(z)$:

$$B(z)=B^+(z)\,B^-(z) \qquad \textbf{(6.5)}$$

$B^+(z)$ represents the zeros whose modulus is less than one

$B^-(z)$ represents the zeros whose modulus is greater than one

The polynomial $B_m(z)$, numerator of the chosen model $H_m(z)$, must therefore contain $B^-(z)$ to ensure that no zero of $H_0(z)$ whose modulus is greater than "one" will be compensated.

So we set:

$$B_m(z)=B^-(z)\,B_m^{'}(z) \qquad \textbf{(6.6)}$$

The polynomial $B_m^{'}(z)$ remains to be determined.

For $B^+(z)$ to be compensated, the only possibility, to allow simplification of the denominator of $\frac{B(z)T(z)}{A(z)R(z)+B(z)S(z)}$, is that it is compensated by the polynomial *R(z)* $R(z)$ must contain $B^+(z)$.

The closed-loop transfer function relative to the diagram in Figure 45 is therefore:

$$H_f(z) = \frac{B(z)T(z)}{A(z)R(z)+B(z)S(z)}$$

Since *H* (z) must be identified with $H_m(z) = \frac{B_m(z)}{A_m(z)}$, then:

$$H_m(z) = \frac{B_m(z)}{A_m(z)} = \frac{B(z)T(z)}{A(z)R(z)+B(z)S(z)} = \frac{B^-(z)B^+(z)T(z)}{A(z)B^+(z)R^-(z)+B^-(z)B^+(z)S(z)}$$

But $B_m(z) = B^-(z)\,B_m^{'}(z)$, so:

$$\frac{B^-(z)B_m^{'}(z)}{A_m(z)} = \frac{B^-(z)T(z)}{A(z)R^-(z)+B^-(z)S(z)}$$

These two fractions are equal, but we can only identify their numerators and denominators respectively if we introduce a norm polynomial $A_0(z)$, called an observer polynomial, chosen by the designer.

Usually $A_0(z) = z^\alpha$, but it can also be chosen by an expression.

$$A_0(z) = z^\alpha + a_1 z^{\alpha-1} + ... + a_0$$

In any case, $A_0(z)$ must satisfy a well-defined condition on its degree, as will be explained later. It is also imperative that the zeros of $A_0(z)$ are located inside the unit circle (for stability).

$$\frac{B^-(z)B_m^{'}(z)}{A_m(z)} = \frac{B^-(z)T(z)}{A(z)R^-(z)+B^-(z)S(z)} \qquad \textbf{(6.8)}$$

We can then establish:

$$A_0(z) = z^{\alpha} \tag{6.9}$$

and by multiplying by $B^+(z)$ and exploiting (6.5) and (6.6), we obtain:

$$B^+(z)A_m(z)A_0(z) = A(z)R(z) + B(z)S(z) \tag{6.10}$$

This formula, called the Diophantine equation, will be exploited later for the calculation of the coefficients of the polynomials.

By imposing the two numerators of (6.2) to be equal, we also have:

$$T(z) = B_m^{'}(z)A_0(z) \tag{6.11}$$

We verify the constraints on the degree of each polynomial to ensure causality (i.e., feasibility, conditioned by the fact that a rational fraction must have the degree of its denominator strictly greater than the degree of its numerator):

$$\begin{cases} D^0[A] \geq D^0[B] \\ D^0[A_m] - D^0[B_m] \geq D^0[A] - D^0[B] \\ D^0[A_0] \geq 2D^0[A] - D^0[A_m] - D^0[B^+] - 1 \\ D^0[R] \geq D^0[S] \\ D^0[R] \geq D^0[T] \end{cases} \tag{6.12}$$

We impose conditions on the equality of the degrees of the polynomials:

$$\begin{cases} D^0[R] = D^0[B^+] + D^0[A_0] + D^0[A_m] - D^0[A] \\ D^0[S] = D^0[A] - 1 \\ D^0[T] = D^0[A_0] + D^0[B_m] \end{cases} \tag{6.13}$$

We calculate the coefficients r_i, s_j and a_k of the polynomials R, S and $A_0^{'}$:

$$\begin{cases} A(z) = z^{\delta A} + a_1 z^{\delta A - 1} + a_2 z^{\delta A - 2} + ... + a_{\delta A} \\ R(z) = z^{\delta R} + a_1 z^{\delta R - 1} + a_2 z^{\delta R - 2} + ... + a_{\delta R} \\ B(z) = z^{\delta B} + a_1 z^{\delta B - 1} + a_2 z^{\delta B - 2} + ... + a_{\delta B} \\ S(z) = z^{\delta S} + a_1 z^{\delta S - 1} + a_2 z^{\delta S - 2} + ... + a_{\delta S} \end{cases} \tag{6.14}$$

$$A_0(z) = z^{\delta A_0 S} + \alpha_1 z^{\delta A_0 S-1} + \alpha_2 z^{\delta A_0 S-2} + ... + \alpha_{\delta A_0}$$

By replacing $A(z)$, $R(z)$, $B(z)$, $S(z)$ and A_0 with their expressions in the equation:

$$A(z)R(z) + B(z)S(z) = B^+(z)A_m(z)A_0(z)$$

and by performing an identification of the coefficients term by term according to the powers of z, $B^+(z)A_m(z)A_0(z)$ where A_0 satisfies:

$$D^0[A_0] \geq 2D^0[A] - D^0[A_m] - D^0[B^+] - 1$$

with R^+ and A_m being known, we can determine the r_i and the s_j and a_k.

Note 1: if the transfer function $H_0(z) = \frac{B(z)}{A(z)}$ does not include an integrator, the polynomial $R(z)$ must contain an integration to allow the elimination of static error:

$$R(z) = B^+(z)(1-z)Q(z) \qquad \textbf{(6.15)}$$

Note 2: It is convenient for numerical simulation to express transfer functions and polynomials in z^{-1}.

Example 1:

Preliminary: Consider a controlled system defined in open loop by its transfer function:

$$G(z) = \frac{0{,}64}{p^2 + 0{,}16p + 0{,}64}$$

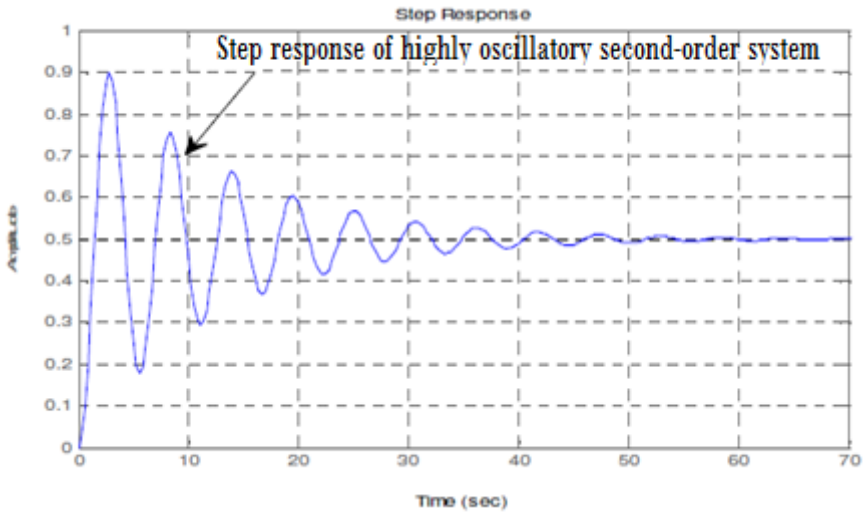
It is a second-order system characterized by a natural frequency $\omega_0 = 0{,}8$ and a low damping factor $\xi = 0{,}1$.

It is highly oscillatory as shown by its step response, plotted using the Matlab software:

G = tf(0.64, [1 0.16 0.64]); % Transfer function syntax

Gf = feedback(G, 1); % Open-loop transfer function

step(Gf), grid % Step response

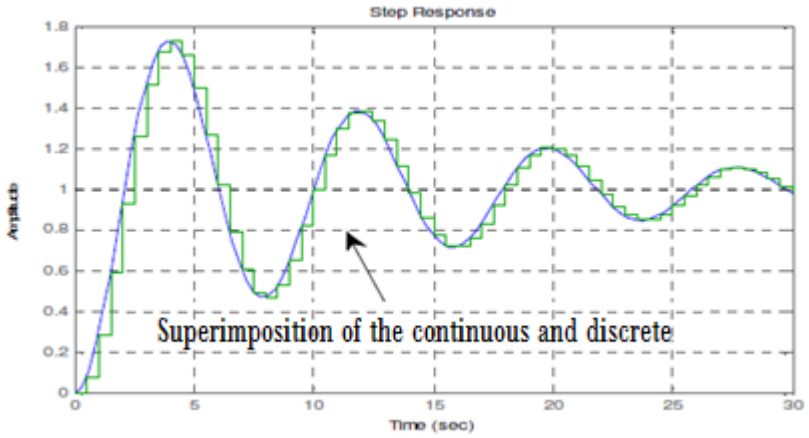


The discretization of $G(z)$ in Matlab with a sampling period less than its smallest time constant, denoted as $T_s = 0{,}5\ s,$ yields:

H = c2d(G, 0.5) *% H = c2d* (syntax for continuous to discrete conversion)

step(G, H, 30), grid

The discrete system is also highly oscillatory, as shown by its step response overlaid with the continuous step response for comparison.



Objective:

We aim to calculate an RST controller that provides the closed-loop system with a second-order behavior with a natural frequency of 0.8 and a damping ratio of $\xi = 0{,}8$ and $\omega_0 = 1{,}4$.

This example is simple because $H_0(z)$ does not contain zeros with magnitudes greater than one.

Step A-1: Let's use the Zpk(H**0)** syntax.

To represent $H_0(z)$ in the form: $H_0(z) = \dfrac{B(z)}{A(z)} = \dfrac{0{,}07688\,(z + 0{,}9735)}{z^2 - 1.771z + 0{,}9231}$

A-2: The zeros of $H_0(z)$ must be decomposed into zeros with magnitudes greater than 'one' $B^-(z)$, and zeros with magnitudes less than 'one' $B^+(z)$.

$$\frac{B(z)}{A(z)} = \frac{B^+(z)B^-(z)}{z^2 - 1.771z + 0{,}9231}$$

We have: $B(z) = 0{,}07688\,(z + 0{,}9735)$

Since there are no zeros with magnitudes greater than one, one might be tempted to set $B^-(z) = 1$; $B^+(z) = 0{,}07688\,(z + 0{,}9735)$ however, the zero $z = -0{,}9735$ is considered too close to 'one' and could be the source of oscillations.

To avoid this, we prefer to choose $B^-(z) = 0{,}07688\,(z + 0{,}9735)$, we do not compensate $B^+(z) = 1$

B) Construction of the transfer function of the model to be tracked.

$$H_m(z) = \frac{B_m(z)}{A_m(z)}$$

Determination of the denominator $A_m(z)$:

$A_m(z)$ should behave like a second-order system with $\xi = 0{,}8$ and $\omega_0 = 1{,}4$.

We calculate an analog second-order system with $\xi = 0{,}8$ and $\omega_0 = 1{,}4$ using Matlab:

[Nbf, Nbf] = ord2(1.4, 0.8) **% (Construction of a second-order system)**

We discretize this second-order system with $T_s = 0{,}5\ s,$ using the syntax:

zp=exp(0.5roots(dbf)

to calculate the discrete roots z_1 and z_2.

We find $z_1 = 0{,}5216 + 0{,}2329\,i$ and $z_2 = 0{,}5216 - 0{,}2329\,i$

Calculation of the denominator $A_m(z)$

Am=[1 -zp(1)-zp(2) zp(1)zp(2)]

which gives:

$$A_m(z) = (z - z_1)(z - z_2) = z^2 - (z_1 + z_2)z + z_1 z_2$$

We want to obtain a static gain equal to 1, so:

$$H_m(1) = \frac{B_m(1)}{A_m(1)} = 1$$

So we choose the numerator $B_m(z) = B(z)\,g$.

Hence:

$$H_m(1) = \frac{B_m(1)}{A_m(1)} = \frac{0{,}07688\,(z + 0{,}9735)g}{z^2 - 1.0431z + 0{,}3263}$$

And we impose:

$g = 1.888$

Finally: $H_m(1) = \dfrac{B_m(1)}{A_m(1)} = \dfrac{0{,}145\,(z + 0{,}9735)}{z^2 - 1.0431z + 0{,}3263}$

C) Verification of the conditions on the degrees of R, S, T, and A_0:

C -1 $D^0[A_m(z)] - D^0[B_m(z)] \geq D^0[A(z)] - D^0[B(z)]$

C -2 $B_m(z) = B^-(z)\,B^*(z) = 0{,}07688\,(z + 0{,}9735)\,g$

C -3 $D^0[A_m(z)] = 2$

D) Identification of the polynomials R, S, T, and the observer polynomial A_0:

D -1

$D^0[A_0(z)] \geq 2D^0[A(z)] - D^0[A_m(z)] - D^0[B^+(z)] - 1$

$D^0[A_0(z)] \geq 4 - 2 - 0 - 1 = 1$ therefore $A_0(z) = z$ (Normalized polynomial)

D -2

$D^0[R(z)] = D^0[A_m(z)] + D^0[A_0(z)] - D^0[A(z)] = 1$

$D^0[R(z)] = 2 + 1 - 2 = 1$ therefore $R(z) = z + r_0$ (Normalized)

D -3

$D^0[S(z)] = D^0[A(z)] - 1 + 1$ then $S(z) = s_1 z + s_0$ (Normalized)

E: Calculation of the coefficients of the polynomials R, S, and T.

We exploit the Diophantine equation to calculate r_0 and s_0:

$A_m(z)A_0(z) = A(z)R(z) + B^-(z)S(z)$

$(z^2 - 1.0431z + 0.3263)z = (z^2 - 1.771z + 0.9231)(z + r_0) + 0.076881(z + 0.9735)(s_1 z + s_0)$

By identifying term by term, we find:

$R(z) = z + 0.365$

$S(z) = -4{,}53z + 4{,}77$

$T(z) = B_m^{'}(1)A_0(z) = g\,A_0(z)$

However $B_m^{'}(1) = \dfrac{A_m(z)}{B^-(z)}$ (because $H_m(1) = \dfrac{B_m(1)}{A_m(1)} = \dfrac{B^-(1)\,B_m(1)}{A_m(1)} = 1$)

$B_m^{'}(1) = \dfrac{0{,}28}{0{,}15} = 0{,}86$

Thuse $T(z) = 1{,}86\,z$

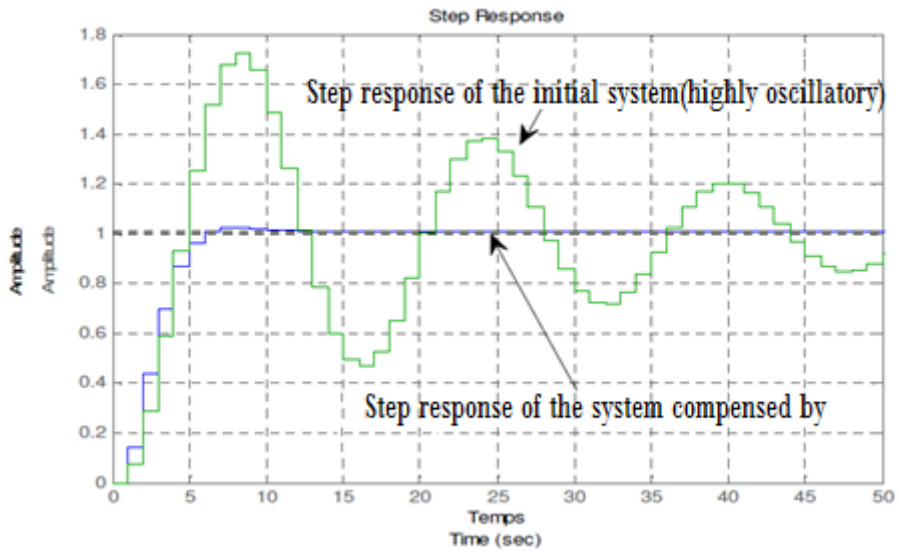
Hence the digital control law: $U(z) = \frac{T(z)}{R(z)} Y_C(z) - \frac{S(z)}{R(z)} Y(z)$

Numerical application (N.A.) $U(z) = \frac{1.86z}{z + 0.365} Y_C(z) - \frac{-4.53z + 4.77}{z + 0.365} Y(z)$

Consequently, the recurrent equation, in the form of an algorithm easily programmable:

$$u_k = -0.365\,u_{k-1} + 1.86\,y_k^c + 4.53\,y_k - 4.77\,y_{k-1}$$

On the diagram below, obtained by numerical simulation, we observe the performance on the step response of the corrected system, which reaches its setpoint in 8 s with very little overshoot, a performance that can be appreciated compared to that of the step response of the uncorrected system:



Example 2:

Consider the system defined by its discrete open-loop function:

$$H_0(z) = \frac{B(z)}{A(z)} = \frac{0.05(z+2)(z+0.3)}{z^3 - 2.2z^2 + 1.8z - 0.5}$$

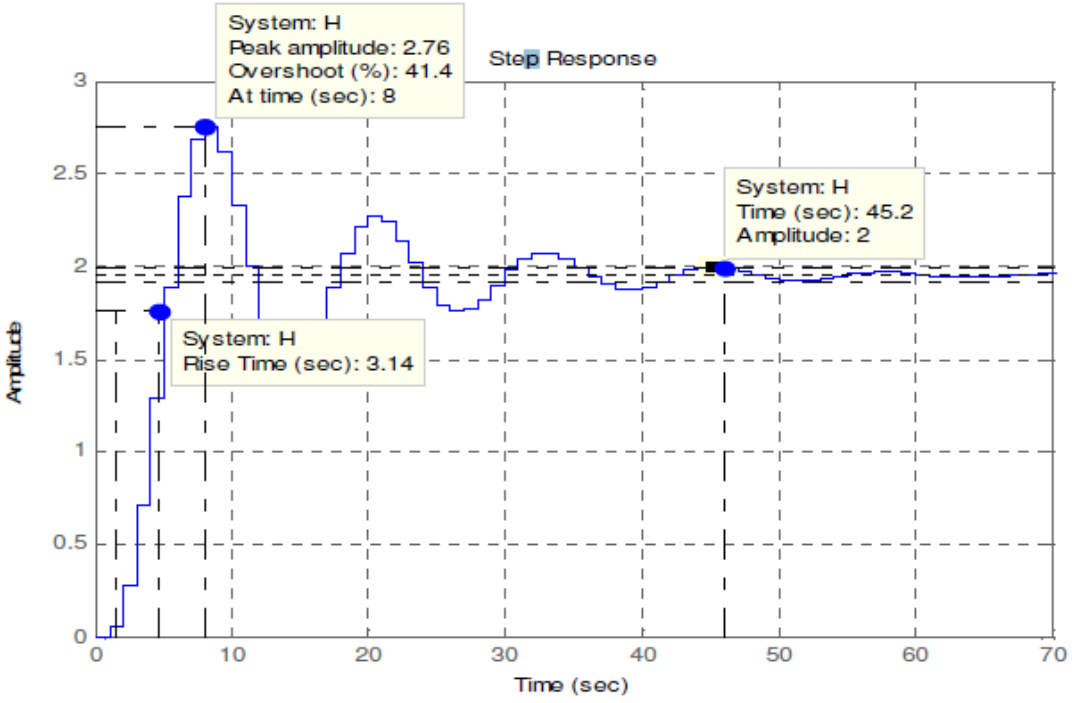
Diagnosis:

1. This system contains a zero at z = -2, located outside the unit circle.

2. Using the syntax $zpk(H_0)$, we obtain its factored form of $H_0(z)$.

From this expression, we deduce that the system has a pole $z_1(z) = 0.5838$ and two complex conjugate poles $z_{2/3}(z) = 0.8080 \pm 0.4513i$, with a modulus equal to *0.9255*, which is considered very close to unity.

2. Furthermore, its step response reveals a highly oscillatory nature with an overshoot exceeding 40% and poor accuracy.



We aim to design an RST controller to place the poles at $z_1 = 0.3$ and $z_{2/3}(z) = 0.2 \pm 0.3i$ to improve the performance of this system and establish the corresponding control law.

Approach:

1. Definition of the model to be tracked:

We decompose the numerator of $H_0(z)$:

$$B(z) = B^+(z)B^-(z) = 0.05(z+0.3)(z+2)$$

$B^-(z) = z+2$ should not be compensated, so it should be included in the numerator of the model to be tracked, $H_m(z)$.

$$B_m(z) = (z+2)B_m^{'}(z)$$

The denominator of $H_m(z)$ must consist of the poles chosen by the designer, denoted as $z_1 = 0.3$ and $z_1 = 0.3$ and $z_{2/3}(z) = 0.2 \pm 0.3i$

$$A_m(z) = (z-3)(z-0.2-0.3i)(z-0.2+0.3i)$$

Therefore, $H_m(z) = \dfrac{(z+2)B_m^{'}}{(z-0.2)(z-0.2-0.3i)(z-0.2+0.3i)}$.

To achieve a unit static gain, it is necessary that:

$H_m(1) = \dfrac{(z+2)B_m^{'}(z)}{(1-0.2)(1-0.2-0.3i)(1-0.2+0.3i)} = 1$ which implies: $k = 17$

$$H_m(z) = \frac{0.17(z+2)}{(z-0.2)(z-0.2-0.3i)(z-0.2+0.3i)}$$

Hence, the condition $D^0[A_m(z)] - D^0[B_m(z)] \geq D^0[A(z)] - D^0[B(z)] = 3-1 > 3-2$ is satisfied.

6.5. Definition of polynomials:

Observer polynomial $A_0(z)$:

Given that $D^0[A_0] \geq 2D^0[A] - D^0[A_m] - D^0[B^+] - 1 = 2*3 - 3 - 1 = 2$, we choose a second-order normalized polynomial.

$A_0(z) = z^2 + a_1 z + a_0$ must have its zeros inside the unit circle.

Polynomial $R(z)$:

As $H_0(z)$ does not contain integration, $R(z)$ must contain integration to eliminate steady-state error.

Since $R(z) = B^+(z)R'(z)$ is chosen such that:

$R(z) = B^+(z)(1-z)^{\alpha} Q'(z)$ with $\alpha = 1$ it follows that: $R(z) = 0.055(z+0.3)(1-z)Q'(z)$

However, $D^0[R] = D^0[B^+] + D^0[A_0] + D^0[A_m] - D^0[A] = 1 + 3 - 2 + 3 - 3 = 2$

Thus, $D^0[Q] = 1$ is a second-order normalized polynomial.

Therefore, $R(z) = 0.055(z+0.3)(1-z)(z+q_0)$.

Polynomial $S(z)$:

From the Diophantine equation $A_m(z)A_0(z) = A(z)R(z) + B^{-1}(z)S(z)$, we can deduce the degree of $S(z)$.

$$S(z) = s_1 z + s_0$$

Polynomial $T(z)$:

$T(z) = B_m' A_0$ however $B_m' = \frac{B_m}{B^-} = \frac{0.17(z+2)}{0.05(z+2)} = 3.4$

$$T(z) = B_m' A_0 = 3.4 A_0(z)$$

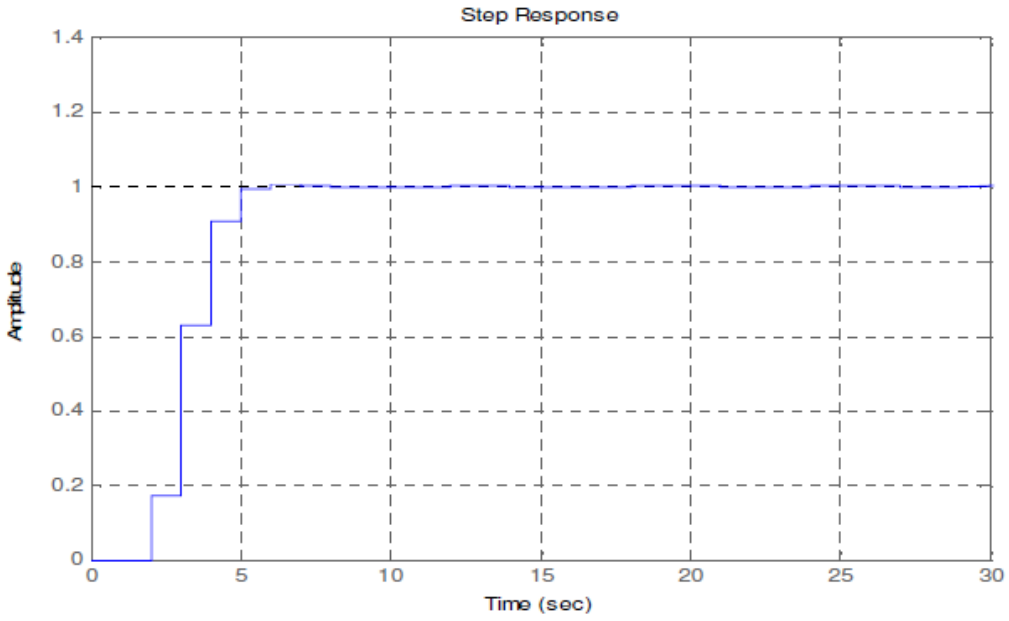
6.6. Calculation of polynomial coefficients

We substitute each polynomial in $A_m(z)A_0(z) = A(z)R(z) + B^{-1}(z)S(z)$ with its expression:

$$(z-0.3)(z-0.2-0.3i)(z^2+a_1z+a_0) = (z^3-2.2z^2+1.8z-0.5)\times 0.055(z+0.3)(z+q_0)+(z+2)(s_1z+s_0)$$

By expanding both sides of this equality and identifying the terms corresponding to the same powers of z, we can determine the coefficients a_0 a_1 s_0 s_1 q_0.

The step response, depicted in the diagram below, shows that the system under this control is no longer oscillatory and reaches its setpoint without overshoot after 5 seconds.



6.7 Conclusion

In summary, the RST controller offers a robust solution for controlling systems with significant pure delays, varying dynamics, and higher-order characteristics that may challenge conventional PID controllers. By using a polynomial approach, RST controllers provide enhanced flexibility in managing tracking dynamics, disturbance rejection, and independently specifying rise times and overshoots. The RST methodology ensures precise control by accommodating process variations and maintaining system stability and performance through carefully designed polynomials *R*, *S*, and *T*. This makes RST controllers a powerful tool in modern control system design, especially for processes requiring high precision and robust performance.

General conclusion

As we conclude this comprehensive course on Digital Control Systems, it is essential to reflect on the depth and breadth of the knowledge we have covered. Our journey began with an exploration of the historical evolution of digital control, setting a foundation that helped us appreciate the advancements and current capabilities in this field. The historical context underscored the importance of early innovations and their lasting impact on modern digital control systems.

We delved into the fundamental components of digital control systems, including analog-to-digital converters (ADCs) and digital-to-analog converters (DACs). These components are crucial for bridging the gap between the analog world of physical processes and the digital realm of computational control. Understanding these converters, along with the processes of sampling and quantization, equipped us with the necessary tools to handle digital signals accurately and efficiently.

Throughout the course, we examined the detailed steps involved in implementing digital control systems. This included choosing appropriate sampling periods, designing digital controllers, and addressing practical challenges such as quantization errors and real-time processing requirements. We also discussed the numerous advantages of digital control, such as high precision and flexibility, while acknowledging the challenges, including the need for specialized knowledge and the costs associated with digital system implementation.

Our exploration extended to advanced control strategies, particularly the design and implementation of PID and RST controllers. These controllers are integral to modern control systems, offering robust solutions for managing complex processes. By understanding their theoretical foundations and practical

applications, we gained insights into designing effective control systems that can achieve desired performance criteria.

The practical aspect of this course was emphasized through the use of MATLAB for simulations and control system design. This hands-on approach reinforced the theoretical concepts, allowing us to see their real-world applications and implications. MATLAB's powerful tools enabled us to model, analyze, and optimize digital control systems, providing a comprehensive learning experience.

By the end of this course, you should have a solid foundation in digital control systems. You are now equipped to design, analyze, and optimize control systems for various industrial applications. This knowledge is crucial for ensuring high performance, reliability, and efficiency in modern technological solutions. Your understanding of the historical context, core components, processes, and advanced strategies positions you to innovate and excel in the field of digital control engineering.

As we conclude, it is important to recognize that the field of digital control is dynamic and continually evolving. Staying updated with the latest advancements and continually honing your skills will be key to your success. This course has provided you with the foundational knowledge and practical tools to embark on this journey. Embrace the challenges and opportunities ahead, and use your expertise to contribute to the ever-growing field of digital control systems.

Bibliographie :

D. Peaucelle, « Système à temps discret, commande numérique des procédés », cours d'ingénieurs, Université Ibn Zohr, Ecole Nationale des Sciences Appliquées, Agadir, Maroc , 7 avril 2003.

S. Tliba, M. Jungers, Y. Chitour, « Commande des processus asservissement numérique, Notes de cours», Master IST1 & IFIPS EI2, UE majeures 421-422, Université Paris-Sud XI - ENS de Cachan, septembre 2005.

I. Bara et M. de Mathelin, « Commande numérique des systèmes», Cours 2ème année, Ecole nationale de physique de Strasbourg, 2004-2005.

M. AKKARI, Les systèmes asservis linéaires Echantillonnés Compensation des systèmes échantillonnée», Université Virtuelle de Tunis.

T. Dumartin, « Rappels Traitement du Signal», Note de cours, Licence Professionnel Optronique, Année 2004 – 2005

R. Gourdeau, et J. Thistle, « Introduction à l'automatisation Commande des systèmes linéaires », école Polytechnique de Montréal, 21 avril 2010.

A. Meghebbar, « Commande Numérique », ´Cours Commande Numérique, Master Automatique.

B. Bergeon, « Représentations d'état des systèmes linéaires à temps discret Université Bordeaux 1, Master 1 EAPS-ESA.

Y. Guettaf, « Asservissements échantillonnés et Régulation Numérique» Centre Universitaire Nour Bachir El Bayadh, Cours UEF 122, 2016-2017.

Table of Contents

Preface	1
General Introduction	2
Chapter 1: Structure of a Digital Control System	5
1.1 Introduction	5
1.2 Historical Background	5
1.3 General Structure of a Digital Control System	7
1.4 Analog-to-Digital and Digital-to-Analog Conversions	7
1.5 Implementation Steps of Control by Computers	9
1.6 Advantages and Disadvantages of Digital Control	9
1.6.1 Advantages of CNC Machines	9
1.6.2 Disadvantages of CNC Machines	10
1.7 Classification of Different Signals	10
1.8 Conclusion	11
Chapter 2: Signal Sampling	12
2.1 Introduction	12
2.2 Modeling of Analog-to-Digital and Digital-to-Analog Converters	12
2.2.1 Case of the Analog-to-Digital Converter (ADC)	12
2.2.2 Case of the Digital-to-Analog Converter (DAC)	14
2.3 Sampling	15
2.3.1 Practical Rule of Sampling	17
2.3.2 Shannon's Theorem	18

2.3.3 Practical Consequence	19
2.4 Reconstruction of a Sampled Signal	19
2.4.1 Zero-Order Hold (ZOH) Circuit	20
2.4.2 First-Order Hold (FOH) Circuit	20
2.5 Conclusion	21
Chapter 3: Representation of Sampled Systems	22
3.1 Introduction	22
3.2 Z-Transform	22
3.2.1 Methods for Calculating the Z-Transform	22
3.2.2 Properties of the Z-Transform	23
3.4 Modified Z-Transform	24
3.5 Discrete-Time System	25
3.5.1 Recursive Equation	25
3.5.2 Z-Transform Transfer Function	26
3.6 System Response	28
3.6.1 Pure Delay	28
3.6.2 Integral and Derivative Functions in Digital	29
3.7 Sampled System	29
3.7.1 Sampled Transfer Function	31
3.7.2 Properties of the Sampled Model	32
3.8 Sampled Transfer Function	33
3.8.1 Influence of Sampler Position	34

3.8.2 Transfer Function of a Process Equipped with a ZOH Block	36
3.8.3 Z-Transform of a System in Unity Feedback	36
3.8.4 Z-Transform of a System in Non-Unity Feedback	37
3.8.5 Digital Control Loops	38
3.8.6 Choice of Sampling Frequency	39
3.9 Poles - Zeros - Stability - Causality	41
3.9.1 Poles	41
3.9.2 Zeros of a digital system	42
3.8 Conclusion	43
Chapter 4: Analysis of Sampled Systems	44
4.1 Introduction	44
4.2 Study of Stability	44
4.2.1 Stability	44
4.2.2 General Stability Condition	44
4.3 Algebraic Criteria	47
4.3.1 Schur-Cohn Criterion	47
4.3.2 Jury Criterion	50
4.3.3 Routh-Hurwitz Criterion	53
4.4 Geometric Criteria	57
4.4.1 Nyquist Criterion	57
4.4.2 Simplified Nyquist Criterion	62
4.4.3 Bode Plot and Black-Nichols Plot	62

4.4.4 Root Locus (or Evans Root Locus)	61
4.4.5 Evans Root Locus	61
4.5 Conclusion	67
Chapter 5: Synthesis of Sampled-Data Systems	68
5.1 Introduction	68
5.2 Objective	68
5.3 Static Accuracy of Sampled-Data Control Systems	69
5.3.1 Study of Accuracy	69
5.3.2 Expression of the Error (Static Accuracy)	70
5.3.3 General Expression of the Error	70
5.3.4 System Class	71
5.4 PID Control	74
5.4.1 Synthesis of Digital Controllers by Extension of Analog Controllers	75
5.4.2 Synthesis Using Continuous Methods	77
5.4.3 Digital PID Control	82
5.4.4 Numerical Expressions of the PID Controller	84
5.5 Conclusion	94
Chapter 6: RST Controller	95
6.1 Introduction	95
6.2 Data on the System to be Controlled	96

6.3 Data on the Reference Model	97
6.4 Design Synthesis of Polynomials RST	98
6.5. Definition of polynomials	110
6.6 Calculation of polynomial coefficients	111
6.7 Conclusion	112
General Conclusion	113
Bibliographic References	115