



HAL
open science

Linear Servo Loop Systems

Thierry Miquel

► **To cite this version:**

| Thierry Miquel. Linear Servo Loop Systems. Master. France. 2020, pp.208. hal-02987746v2

HAL Id: hal-02987746

<https://cel.hal.science/hal-02987746v2>

Submitted on 17 Oct 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution - NonCommercial 4.0 International License

Linear Servo Loop Systems

Lecture notes

Thierry Miquel
thierry.miquel@enac.fr

September 12, 2022

Course overview

Following Wikipedia *a system is a set of interacting or interdependent components forming an integrated whole.*

System control exists for more than 2000 years. Water clocks control was one of the first application of system control. Nevertheless the scientific approach appears after the mid of the 19th century. In 1868 James Clerk Maxwell (1831-1879) published a paper entitled "On Governors" within which he described how to derive the linear differential equations for various governor mechanisms. In 1798 James Watt (1736-1819) developed the automatic control of the speed of a steam motor.

The theory of control has been developed during the first half of the 20th century both in the western countries and in the ex-URSS but the motivations were different. Indeed they were based on mathematical interests in the East. For example Nicholas Minorsky (1885-1970) presented in 1922 a clear analysis of the control involved in position control systems and formulated a control law that we now refer to PID control. On the other side engineering issues were the main drivers in the West. In 1914 Lawrence Burst Sperry (1892-1923) invented the first autopilot which he demonstrated with success in France. In the US telephone repeaters based on electronic amplification of the signal were used around 1920 but the distortion they introduced limited the number that could be used in series. Harold Stephen Black (1898-1983) developed the electronic amplifier with negative feedback in 1927 to overcome the issue.

The so called *classical control* period extends approximatively from 1935 to 1960. By the end of this period the classical control techniques had been established. The design methodologies were for linear SISO (Single Input and Single Output) systems; those systems can be described by linear differential equations with constant coefficients.

Starting in the late 50's – beginning of the 60's, the *modern control* period tackles problems involving the analysis and control of linear MIMO (Multiple Input and Multiple Output) systems and nonlinear systems (e.g. Sputnik in 1957). An important step was accomplished with Rudolf (Rudi) Emil Kalman's (born in Budapest in 1930) treatment of the linear multivariable optimal control problem with a quadratic performance index.

You may read the paper entitled *A Brief History of Automatic Control, IEEE Control Systems, June 1996* by Stuart Bennett for further details.

This course will focus on *classical control*. More specifically, the objectives are the following:

- to learn how to model dynamic systems and to present mathematical representation of these models;
- to learn how to analyse dynamic properties of open loop (and also closed loop) systems;
- to learn how to design basic feedback control loop to achieve stability and precision goals.

This lecture is organized as follows: the first chapter introduces dynamical systems and the way to linearize a nonlinear differential equation. Then we present the Laplace transform, which is the dedicated mathematical tool to study systems driven by linear differential equations. The third chapter introduces Linear Time Invariant (LTI) systems and focuses on their time and frequency responses. The fourth chapter is dedicated to the stability analysis of LTI systems. Then basic control designs based on steady state analysis or transient response specifications are provided in chapter five. The last chapter focuses on discrete time systems and more specifically on the \mathcal{Z} transform and the modeling of continuous time system with sampled input and output.

References

- [1] *Introduction au contrôle des mini-drones : de la conception à la mise en œuvre*, Thierry Miquel, Jean-Philippe Condomines, Bordeneuve-Guibé Joël, Brière Yves, Ellipses Formations & Techniques
- [2] *Introduction to Feedback Control* Li Qiu, Kemin Zhou, Pearson Prentice Hall
- [3] *Modern Control Systems*, Richard C. Dorf, Robert H. Bishop, Prentice Hall
- [4] *Feedback Control of Dynamic Systems*, Gene F. Franklin, J Powel, Abbas Emami-Naeini, Pearson
- [5] *Linear Control System Analysis and Design*, Constantine H. Houpis, Stuart N. Sheldon, John J. D'Azzo, CRC Press
- [6] *Modern Control Engineering*, P.N. Paraskevopoulos, CRC Press
- [7] *Control Engineering: An Introductory Course*, Michael Johnson, Jacqueline Wilkie, Palgrave Macmillan
- [8] *Feedback Control System*, Charles L. Phillips and Royce D. Harbor, Prentice-Hall

Contents

1	Introduction to dynamical systems	11
1.1	Chapter overview	11
1.2	Refresher on derivatives	11
1.3	Differential equations	13
1.4	From nonlinear to linear systems	14
2	Laplace transform	21
2.1	Chapter overview	21
2.2	Reference signals	21
2.2.1	Dirac delta function	21
2.2.2	Unit step function	23
2.2.3	Complex exponential function	23
2.3	Definition	24
2.4	Properties	25
2.4.1	Linearity	25
2.4.2	Derivation	26
2.4.3	Integration	26
2.4.4	Multiplication by t	27
2.4.5	Time domain translation (delay)	27
2.4.6	Frequency domain translation	29
2.4.7	Initial value theorem	29
2.4.8	Final value theorem	30
2.4.9	Convolution	30
2.4.10	Evaluating improper integrals	30
2.4.11	Link with Fourier transform	30
2.5	Usual Laplace transform pairs	31
2.6	Inverse Laplace transform	31
2.6.1	Partial fraction expansion	32
2.6.2	Mellin-Fourier integral	33
2.7	Norm of vector signal	36
2.8	Mellin transform	37
3	Linear Time Invariant systems	39
3.1	Chapter overview	39
3.2	Properties	39
3.3	Convolution integral	40

3.4	System defined by a linear differential equation	41
3.5	Frequency response	42
3.5.1	Relation with transfer function	42
3.5.2	Bode plot	44
3.6	Model reduction	46
3.7	First order model	52
3.7.1	Time response	52
3.7.2	Frequency response	55
3.8	Second order model	56
3.8.1	Differential equation and unit step response	56
3.8.2	Overdamped model	58
3.8.3	Underdamped model	59
3.8.4	Critically damped model	61
3.8.5	Frequency response	62
4	Analysis of Linear Time Invariant systems	65
4.1	Open-loop versus closed-loop	65
4.2	Stability	68
4.3	Routh-Hurwitz criterion	69
4.4	Roots in a specified trapezoidal region	75
4.5	Hermite's criterion	76
4.6	Nyquist stability criterion	79
4.7	Phase and gain margins	83
4.8	Root locus	86
4.9	Effect of zeros on the system response	93
4.10	Step response of non-minimum phase transfer functions	94
4.11	Strong stabilizability and parity interlacing property	95
5	Basic feedback control design	97
5.1	Chapter overview	97
5.2	Lead-lag controller	97
5.2.1	Steady state error	97
5.2.2	Controller presentation	101
5.2.3	Lag controller tuning: graphical method	103
5.2.4	Lead controller tuning: graphical method	104
5.2.5	Example	105
5.2.6	Lead-lag controller tuning: computational method	107
5.3	Transient response specifications	108
5.4	ITAE performance index	110
5.5	PID controller	113
5.5.1	Controller presentation	113
5.5.2	Ziegler and Nichols tuning method	116
5.5.3	Phillips and Harbor tuning method	117
5.5.4	Real dominant pole placement with PI or PD controller	120
5.5.5	Complex conjugate dominant pole placement with PID controller	122
5.6	Prefiltering	125

5.6.1	Purpose	125
5.6.2	Transfer function with negative real-parts zeros	126
5.6.3	Transfer function with positive real-parts zeros	127
5.6.4	Design steps	130
5.7	Algebraic controller design	132
5.7.1	Polynomial product	132
5.7.2	Sylvester's Theorem	135
5.7.3	One degree of freedom controller design	138
5.7.4	Prefilter design	141
5.7.5	Youla parameterization of all stabilizing controllers	147
5.7.6	Two degrees of freedom controller design	150
5.7.7	Equivalent two-degree of freedom feedback loops	156
5.8	Usefulness of integral action	158
5.9	Anti-windup	160
6	Discrete time systems	167
6.1	Chapter overview	167
6.2	Nyquist–Shannon sampling theorem	168
6.3	Reference sequences	169
6.3.1	Unit impulse	170
6.3.2	Sampled unit step	171
6.3.3	Sampled exponential	172
6.4	\mathcal{Z} transform	172
6.4.1	Definition	172
6.4.2	Properties	173
6.4.3	Usual \mathcal{Z} transform pairs	175
6.4.4	Inverse \mathcal{Z} transform	176
6.4.5	Relation with Laplace transform	178
6.5	Discrete linear time invariant systems	181
6.5.1	Discrete time transfer function	181
6.5.2	Stability analysis	182
6.5.3	Frequency response	185
6.6	Continuous time plant with Zero-Order Hold (ZOH) and sampler	188
6.7	State-space realization	193
6.7.1	Discretization	193
6.7.2	Linear system perturbed by white Gaussian noises	195
6.7.3	State space realization of $(1 - z^{-1}) \mathcal{Z} \left[\frac{F(s)}{s} \right]$	195
6.7.4	First order approximation	197
6.7.5	Miscellaneous	197
6.8	Discretization using Adomian series	197
6.8.1	Differential equations	197
6.8.2	Nonlinear dynamical systems	199
6.9	Derivation of discrete time controller from continuous time controller	203
6.9.1	Tustin's approximation	203
6.9.2	State-space realization of bilinear conformal mapping	205

6.9.3	Matched Pole-Zero (MPZ) mapping method	206
-------	--	-----

Chapter 1

Introduction to dynamical systems

1.1 Chapter overview

Usually an appropriate mathematical model of the system to be controlled is needed before controlling it. The mathematical model may be obtained either from physical laws or experimental data. In this chapter we recall what are differential equations. As far as most physical systems are described by nonlinear differential equations we present how to get a linearized model around an equilibrium point in order to be in position to apply all the mathematical tools dedicated to linear time invariant systems.

1.2 Refresher on derivatives

Let's consider the curve in Figure 1.1 which represents a function $y = f(t)$.

At any value of t the slope of the curve between two points is defined as the

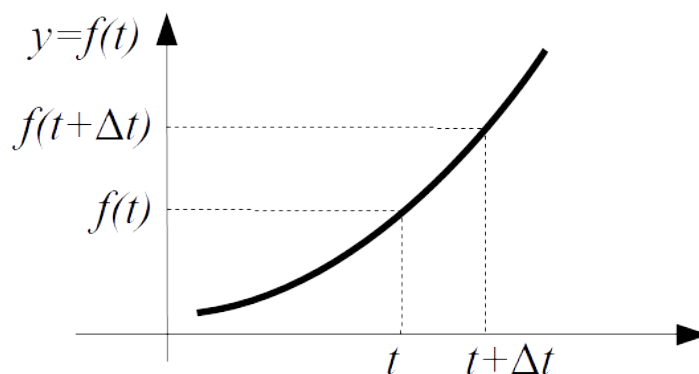


Figure 1.1: Curve $y = f(t)$

change in y divided by the change in t :

$$\text{slope} = \frac{\text{change in } y}{\text{change in } t} = \frac{f(t + \Delta t) - f(t)}{\Delta t} \quad (1.1)$$

The derivative of function f at t is defined as the limit of the slope when Δt tends towards 0:

$$f'(t) = \lim_{\Delta t \rightarrow 0} \frac{f(t + \Delta t) - f(t)}{\Delta t} \quad (1.2)$$

The following notations are equivalent to denote the first derivative of $y = f(t)$ with respect to time t :

- Lagrange's notation: $f'(t)$
- Leibniz's notation: $\frac{dy}{dt}$ or $\frac{d}{dt}f(t)$
- Newton's notation: $\dot{y}(t)$
- Euler's notation uses a differential operator D which is applied to a function $f(t)$ to give the first derivative $Df(t)$

We recall on Tables 1.1 and 1.2 some basic differentiation formulas and basic differentiation rules for combined functions¹.

$\frac{d}{dt}t^n = nt^{n-1}$
$\frac{d}{dt}e^{f(t)} = e^{f(t)} \times \frac{d}{dt}f(t)$
$\frac{d}{dt}(a^t) = \frac{d}{dt}(e^{t \ln(a)}) = (\ln(a)) \times a^t$
$\frac{d}{dt}e^t = e^t$
$\frac{d}{dt} \log_a(f(t)) = \frac{d}{dt} \frac{\ln(f(t))}{\ln(a)} = \frac{1}{\ln(a)} \frac{d}{dt}f(t)$
$\frac{d}{dt} \ln(t) = \frac{1}{t}$
$\frac{d}{dt} \sin(t) = \cos(t)$
$\frac{d}{dt} \cos(t) = -\sin(t)$
$\frac{d}{dt} \tan(t) = 1 + \tan^2(t)$

Table 1.1: Basic differentiation formulas

Derivative of constant multiple	$\frac{d}{dt}(a \times f(t)) = a \times \frac{d}{dt}f(t)$
Derivative of sum	$\frac{d}{dt}(a_1 f_1(t) + a_2 f_2(t)) = a_1 \frac{d}{dt}f_1(t) + a_2 \frac{d}{dt}f_2(t)$
Product Rule	$\frac{d}{dt}(f_1(t) \times f_2(t)) = \frac{d}{dt}f_1(t) \times f_2(t) + f_1(t) \times \frac{d}{dt}f_2(t)$
Quotient Rule	$\frac{d}{dt} \frac{f_1(t)}{f_2(t)} = \frac{\frac{d}{dt}f_1(t) \times f_2(t) - f_1(t) \times \frac{d}{dt}f_2(t)}{(f_2(t))^2}$
Chain Rule	$\frac{d}{dt}f_1(f_2(t)) = \frac{d}{dt}f_1(t) \Big _{t=f_2(t)} \times \frac{d}{dt}f_2(t)$

Table 1.2: Basic differentiation rules for combined functions

The derivative of order n of $f(t)$ is equivalently denoted by $f^{(n)}(t)$, $\frac{d^n y}{dt^n}$, $\frac{d^n}{dt^n}f(t)$ or $D^n f(t)$.

¹<https://www.math.wustl.edu/freiwald/Math131/derivativetable.pdf>

The derivative of order n of $f(t)$ is the derivative of the $n - 1^{th}$ derivative of $f(t)$:

$$\frac{d^n}{dt^n} f(t) = \frac{d}{dt} \left(\frac{d^{n-1}}{dt^{n-1}} f(t) \right) \quad (1.3)$$

1.3 Differential equations

Differential equations first appears with Gottfried Wilhelm von Leibniz (1646-1716) and Isaac Newton (1642-1727)²

A differential equation is a mathematical equation that relates some function with its derivatives.

Differential equations can be divided into several types:

- Ordinary differential equation: an ordinary differential equation (*ODE*) is a differential equation that contains unknown one-variable or multi-variable functions and their derivatives.

An example of ordinary differential equation with respect to the unknown one-variable-variable function $y(t)$ is the following:

$$\frac{d^2}{dt^2} y(t) - 5y(t) = \cos(t) \quad (1.4)$$

- Partial differential equation: a partial differential equation (*PDE*) is a differential equation that contains unknown multi-variable functions and their *partial* derivatives. The *partial* derivative of a function of several variables is its derivative with respect to one of those variables with the others held constant.

An example of partial differential equation with respect to the unknown multi-variable function $u(x, y)$ is the following:

$$\frac{\partial u(x, y)}{\partial x} y + \frac{\partial u(x, y)}{\partial y} x = 0 \quad (1.5)$$

- Linear differential equation: linear differential equations are differential equations (either ODE or PDE) having solutions whose linear combination form further solutions³. We recall that a linear combination is an expression constructed from a set of terms by multiplying each term by a constant and adding the results⁴.

For example the general form of first-order linear differential equations with respect to the unknown one-variable-variable function $y(t)$ reads:

$$\frac{d}{dt} y(t) + a_0(t)y(t) = f(t) \quad (1.6)$$

²History of Ordinary Differential Equations: the First Hundred Years, John E. Sasser

³https://en.wikipedia.org/wiki/Linear_differential_equation

⁴https://en.wikipedia.org/wiki/Linear_combination

Then if $y_1(t)$ is a solution of (1.6) and $y_2(t)$ is another solution of (1.6) then $\alpha_1 y_1(t) + \alpha_2 y_2(t)$, where α_1 and α_2 are constants, is also a solution of (1.6).

More generally a linear ordinary differential equation can be written as follows where $a_{n-1}(t), \dots, a_1(t), a_0(t)$ and $f(t)$ are known functions and where function $y(t)$ has to be found:

$$\frac{d^n}{dt^n} y(t) + a_{n-1}(t) \frac{d^{n-1}}{dt^{n-1}} y(t) + \dots + a_1(t) \frac{d}{dt} y(t) + a_0(t) y(t) = f(t) \quad (1.7)$$

- The order of a differential equation is the order of the *highest* derivative that appears in the equation. For example the following differential equation is a differential equation of order 2:

$$\frac{d^2}{dt^2} y(t) + 3 \frac{d}{dt} y(t) + 2y(t) = t^2 \quad (1.8)$$

1.4 From nonlinear to linear systems

Applying fundamental physical rules (i.e. Euler-Lagrange or Newton dynamic principles for mechanical systems, Kirchhoff's rules for electrical systems, Maxwell rules for electro-magnetic systems, ...) the modeling of a system always leads to an input-output relation between inputs $u(t)$ and outputs $y(t)$ and their derivatives:

$$h(y(t), \dot{y}(t), \ddot{y}(t), \dots, u(t), \dot{u}(t), \dots) = 0 \quad (1.9)$$

Function h is called the input-output relation. The model defined by the relation h can be:

- Either a discrete time or a continuous time model. For discrete time models the values of the time variable t are defined at distinct points in time. This is not the case for continuous time models;
- Either a time invariant or a time variant model. For time variant model time t appears explicitly in the input-output relation h .
- Either a nonlinear or a linear model. The model is linear if h is a linear combination of $y(t), \dot{y}(t), \ddot{y}(t), \dots, u(t), \dot{u}(t), \dots$;
- Either a SISO (Single Input and Single Output) or a MIMO (Multiple Input and Multiple Output) model.

When the input-output relation h is nonlinear there exists quite few mathematical tools which enable to catch the intrinsic behavior of the system. Nevertheless this situation radically changes when h is linear and the good news is that it is quite simple to approximate a nonlinear model with a linear model around an equilibrium point. We will first define what we mean by *equilibrium point* and then we will see how to get a linear model from a nonlinear model.

An equilibrium point is a constant value of the pair (y, u) , which will be denoted (y_e, u_e) , such that:

$$h(y_e, u_e) = 0 \quad (1.10)$$

It is worth noticing that as soon as (y_e, u_e) is a constant value then we have $\dot{y}_e = \ddot{y}_e = \dots = 0$ and $\dot{u}_e = \ddot{u}_e = \dots = 0$.

Then the linearization process consists in computing the Taylor series expansion of h around the equilibrium point (y_e, u_e) and to stop it at the order 1. The linearization of a multivariable function $h(\underline{x})$ around the equilibrium point \underline{x}_e reads:

$$h(\underline{x}) \approx h(\underline{x}_e) + \nabla h(\underline{x})|_{\underline{x}=\underline{x}_e} (\underline{x} - \underline{x}_e) \quad (1.11)$$

Where ∇ is the Jacobian of h that is the vector (or matrix) composed with all first-order partial derivatives of h :

$$\nabla^T = \left[\frac{\partial}{\partial x_1} \quad \dots \quad \frac{\partial}{\partial x_n} \right] \quad (1.12)$$

Applying this relation to the implicit differential equation (1.9) reads:

$$\begin{aligned} 0 = h(y(t), \dot{y}(t), \ddot{y}(t), \dots, u(t), \dot{u}(t), \dots) &\approx h(y_e, u_e) + \\ &\frac{\partial h}{\partial y} \Big|_{y=y_e, u=u_e} \delta y + \frac{\partial h}{\partial \dot{y}} \Big|_{y=y_e, u=u_e} \delta \dot{y} + \dots \\ &+ \frac{\partial h}{\partial u} \Big|_{y=y_e, u=u_e} \delta u + \frac{\partial h}{\partial \dot{u}} \Big|_{y=y_e, u=u_e} \delta \dot{u} + \dots \end{aligned} \quad (1.13)$$

Where:

$$\left\{ \begin{array}{l} \delta y = y - y_e \\ \delta \dot{y} = \dot{y} - \dot{y}_e = \dot{y} \\ \vdots \\ \delta u = u - u_e \\ \delta \dot{u} = \dot{u} - \dot{u}_e = \dot{u} \\ \vdots \end{array} \right. \quad (1.14)$$

Using the fact that $h(y_e, u_e) = 0$ the preceding relation can be rewritten as follows:

$$\begin{aligned} a_0 \delta y + a_1 \frac{d}{dt} \delta y + \dots + a_{n-1} \frac{d^{n-1}}{dt^{n-1}} \delta y + a_n \frac{d^n}{dt^n} \delta y \approx \\ b_0 \delta u + b_1 \frac{d}{dt} \delta u + \dots + b_{m-1} \frac{d^{m-1}}{dt^{m-1}} \delta u + b_m \frac{d^m}{dt^m} \delta u \end{aligned} \quad (1.15)$$

Where:

$$\left\{ \begin{array}{l} a_0 = \frac{\partial h}{\partial y} \Big|_{y=y_e, u=u_e} \\ a_1 = \frac{\partial h}{\partial \dot{y}} \Big|_{y=y_e, u=u_e} \\ \vdots \\ b_0 = - \frac{\partial h}{\partial u} \Big|_{y=y_e, u=u_e} \\ b_1 = - \frac{\partial h}{\partial \dot{u}} \Big|_{y=y_e, u=u_e} \\ \vdots \end{array} \right. \quad (1.16)$$

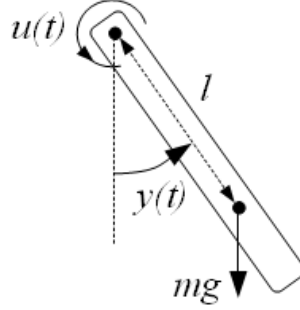


Figure 1.2: Robot arm

Usually the δ sign which indicates that the system approximation is made around an equilibrium point is omitted and the system model (1.15) simply reads:

$$a_0 y(t) + a_1 \frac{dy(t)}{dt} + \cdots + a_{n-1} \frac{d^{n-1}y(t)}{dt^{n-1}} + a_n \frac{d^n y(t)}{dt^n} = b_0 u(t) + b_1 \frac{du(t)}{dt} + \cdots + b_{m-1} \frac{d^{m-1}u(t)}{dt^{m-1}} + b_m \frac{d^m u(t)}{dt^m} \quad (1.17)$$

Consequently the nonlinear input-output relation h can be approximated around an equilibrium point by a linear differential equation of the form (1.17). Nevertheless is worth noticing that the linearization process is an approximation that is only valid around a region close to the equilibrium point.

Example 1.1. Again let's consider Figure 1.2 where a robotic arm is depicted: $u(t)$ is the torque applied by a motor drive and $y(t)$ is the angular position of the arm. In addition we denote m the mass of the arm, l the distance between the axis of the motor and the centre of mass of the arm, b the viscous friction coefficient, J its inertia and g the acceleration of gravity. The dynamics of the robotic arm is the following:

$$J\ddot{y}(t) + b\dot{y}(t) + mlg \sin(y(t)) = u(t) \quad (1.18)$$

This system model is :

- a continuous time model;
- a time invariant model;
- a nonlinear model;
- a SISO (Single Input and Single Output) model.

The input-output relation h is defined as follows:

$$h(y(t), \dot{y}(t), \ddot{y}(t), u(t)) = J\ddot{y}(t) + b\dot{y}(t) + mlg \sin(y(t)) - u(t) \quad (1.19)$$

As far as equilibrium points are concerned, every couples (y_e, u_e) which satisfy the following relation are possible:

$$mlg \sin(y_e) = u_e \quad (1.20)$$

Imposing $u_e = 0$ (no torque applied on the arm at the equilibrium point) the preceding relation reduces to:

$$mlg \sin(y_e) = 0 \quad (1.21)$$

Consequently $(y_e = 0, u_e = 0)$ and $(y_e = \pi, u_e = 0)$ are two equilibrium points. Let's linearize the nonlinear model around those two equilibrium points:

- Around $(y_e = 0, u_e = 0)$ where the arm is vertical and downward oriented we get:

$$\begin{cases} a_0 = \left. \frac{\partial h}{\partial y} \right|_{y=0, u=0} = mlg \cos(y)|_{y=0} = mlg \\ a_1 = \left. \frac{\partial h}{\partial \dot{y}} \right|_{y=0, u=0} = b \\ a_2 = \left. \frac{\partial h}{\partial \ddot{y}} \right|_{y=0, u=0} = J \\ b_0 = - \left. \frac{\partial h}{\partial u} \right|_{y=0, u=0} = 1 \end{cases} \quad (1.22)$$

So the linearized model of the robotic arm around $(y_e = 0, u_e = 0)$ reads:

$$J\delta\ddot{y}(t) + b\delta\dot{y}(t) + mlg\delta y(t) = \delta u(t) \quad (1.23)$$

- Around $(y_e = \pi, u_e = 0)$ where the arm is vertical and upward oriented we get:

$$\begin{cases} a_0 = \left. \frac{\partial h}{\partial y} \right|_{y=\pi, u=0} = mlg \cos(y)|_{y=\pi} = -mlg \\ a_1 = \left. \frac{\partial h}{\partial \dot{y}} \right|_{y=\pi, u=0} = b \\ a_2 = \left. \frac{\partial h}{\partial \ddot{y}} \right|_{y=\pi, u=0} = J \\ b_0 = - \left. \frac{\partial h}{\partial u} \right|_{y=\pi, u=0} = 1 \end{cases} \quad (1.24)$$

So the linearized model of the robotic arm around $(y_e = \pi, u_e = 0)$ reads:

$$J\delta\ddot{y}(t) + b\delta\dot{y}(t) - mlg\delta y(t) = \delta u(t) \quad (1.25)$$

In both situation we get a linear differential equation with constant coefficients. ■

Example 1.2. Let's consider the following nonlinear differential equation:

$$\sin(\ddot{y}(t)) + u^2(t) - 2 + e^{y(t)} = 0 \quad (1.26)$$

The equilibrium points satisfy the following relation:

$$u_e^2 - 2 + e^{y_e} = 0 \quad (1.27)$$

We arbitrarily choose the following equilibrium point:

$$\begin{cases} u_e = 1 \\ y_e = 0 \end{cases} \quad (1.28)$$

Let's linearize the nonlinear model around those the equilibrium point ($y_e = 0, u_e = 1$). We get:

$$\begin{aligned} h(y(t), \dot{y}(t), \ddot{y}(t), u(t)) &= \sin(\ddot{y}(t)) + u^2(t) - 2 + e^{y(t)} \\ \Rightarrow \begin{cases} a_0 = \left. \frac{\partial h}{\partial y} \right|_{y=0, u=1} = e^y|_{y=0} = 1 \\ a_1 = \left. \frac{\partial h}{\partial \dot{y}} \right|_{y=0, u=1} = 0 \\ a_2 = \left. \frac{\partial h}{\partial \ddot{y}} \right|_{y=0, u=1} = \cos(\ddot{y})|_{y=0} = 1 \\ b_0 = - \left. \frac{\partial h}{\partial u} \right|_{y=0, u=1} = -2u|_{u=1} = -2 \end{cases} \end{aligned} \quad (1.29)$$

So the linearized model around ($y_e = 0, u_e = 1$) reads:

$$\ddot{y}(t) + \delta y(t) = -2\delta u(t) \Leftrightarrow \ddot{y}(t) = -\delta y(t) - 2\delta u(t) \quad (1.30)$$

Figure 1.3 represents the response of the linear and the nonlinear model for $u(t) = 1 + 0.01\sin(t)$ over 30 seconds. We can see that both responses are quite similar.

To simulate the response of the system we create the following state vector:

$$\begin{cases} x_1(t) = y(t) \\ x_2(t) = \dot{y}(t) \end{cases} \quad (1.31)$$

Consequently the nonlinear differential equation reads:

$$\begin{aligned} \sin(\ddot{y}(t)) + u^2(t) - 2 + e^{y(t)} &= 0 \\ \Leftrightarrow \begin{cases} \dot{y}(t) = \dot{x}_1(t) = x_2(t) \\ \ddot{y}(t) = \dot{x}_2(t) = \text{asin}(2 - e^{x_1(t)} - u^2(t)) \end{cases} \end{aligned} \quad (1.32)$$

For $\text{asin}()$ to exist the inequality $|2 - e^{x_1(t)} - u^2(t)| \leq 1$ shall be checked. The Scilab code to simulate both the nonlinear and the linear responses is the following:

```
function xdot = nonLinearModel(t,x,A)
    xdot = zeros(2,1);
    xdot(1) = x(2);
    ue = 1; //equilibrium point
    du = A*sin(t);
    u = ue + du;
    siny2dot = 2 - exp(x(1))-u^2;
    xdot(2) = asin(siny2dot);
endfunction
```

```
function xdot = linearModel(t,x,A)
    xdot = zeros(2,1);
```

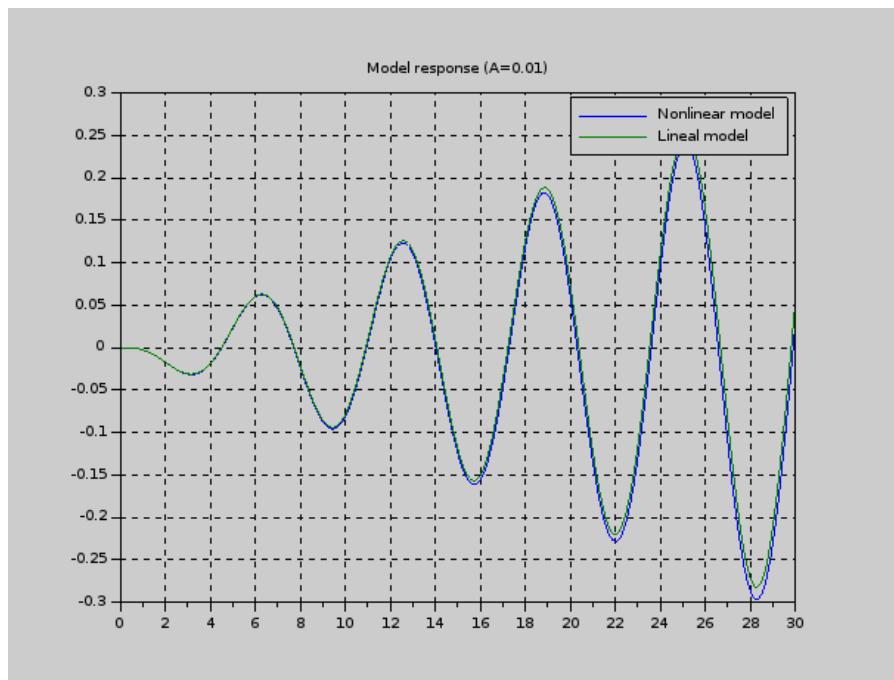


Figure 1.3: Time response of nonlinear and linearized system

```

    xdot(1) = x(2);
    du = A*sin(t);
    xdot(2) = -x(1) - 2*du;
endfunction

ye = 0; //equilibrium point

y0=0;
ydot0 = 0;

A = 0.01;
t=0:0.1:30;
yNonLinear = ode([y0;ydot0],0,t,list(nonLinearModel,A));
dyLinear = ode([y0;ydot0],0,t,list(linearModel,A));
figure();
plot(t,[yNonLinear(1,:); ye + dyLinear(1,:)]);
xgrid();
legend(['Nonlinear model'; 'Lineal model'])
title('Model response (A=0.01)')

```

■

Chapter 2

Laplace transform

2.1 Chapter overview

This chapter introduces the Laplace transform for continuous time functions. This is a powerful tool to solve linear differential equations with constant coefficients. Indeed linear operators like derivation or integration are transformed thanks to the Laplace transform into algebraic operations involving a complex variable which will be denoted s . Consequently the Laplace transform changes a constant coefficients linear differential equation into an algebraic (i.e. polynomial) equation. Assuming that this algebraic equation can be solved for the Laplace transform of the function to be found then the function itself which solves the linear differential equation is obtained by the inverse Laplace transform; this is accomplished either by partial fraction expansion and the use of a table of Laplace transform pairs or by the use of the Mellin-Fourier (or Bromwich) integral. The purpose of this chapter is to present the Laplace transform and to overview its main properties as well as Laplace transform inversion methods.

2.2 Reference signals

A causal signal is a signal which is zero for negative values of time:

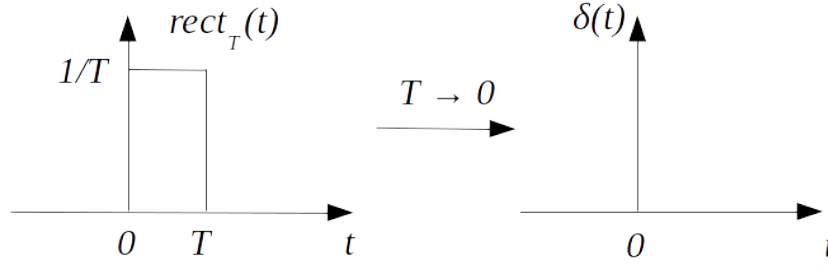
$$g(t) = 0 \quad \forall t < 0 \tag{2.1}$$

In the following we will assume that all the signals are causal.

The aim of this section is to present important signals which are currently used to study continuous time systems: more specifically we will focus on the Dirac delta function, the unit step function and the complex exponential function.

2.2.1 Dirac delta function

The Dirac delta function or $\delta(t)$ function is a generalized function (or distribution) which is zero everywhere except at $t = 0$ with an integral of one

Figure 2.1: Dirac delta function $\delta(t)$

over the entire values of t :

$$\delta(t) = \begin{cases} 0 & \forall t \neq 0 \\ \infty & \text{when } t = 0 \end{cases} \quad \text{s.t.} \quad \int_0^{\infty} \delta(t) dt = 1 \quad (2.2)$$

Dirac delta function can be seen as the limit as $T \rightarrow 0$ of function $rect_T(t)$ which is equal to $1/T \forall t \in]0, T[$ and 0 elsewhere. The schematic representation of the Dirac delta function is a vertical arrow at $t = 0$ as depicted on figure 2.1.

The following properties of the Dirac delta function are often used:

- Multiplication between a signal $x(t)$ and the Dirac delta function translated by $t_0 \in \mathbb{R}$:

$$x(t)\delta(t - t_0) = x(t_0)\delta(t - t_0) \quad (2.3)$$

- Convolution: the convolution product $*$ between two causal signals $f(t)$ and $g(t)$ is defined by:

$$f(t) * g(t) = \int_0^{\infty} f(\tau)g(t - \tau)d\tau \quad (2.4)$$

Using the fact that $g(t)$ is causal we get:

$$g(t) = 0 \quad \forall t < 0 \Leftrightarrow g(t - \tau) = 0 \quad \forall (t - \tau) < 0 \Leftrightarrow \tau > t \quad (2.5)$$

Thus the convolution integral reduces to:

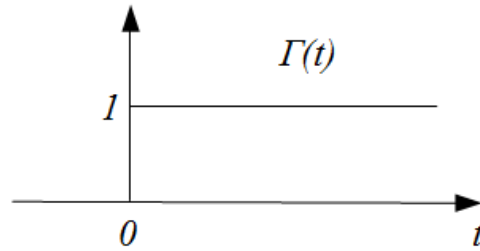
$$f(t) * g(t) = \int_0^t f(\tau)g(t - \tau)d\tau \quad (2.6)$$

The convolution product is commutative:

$$f(t) * g(t) = g(t) * f(t) \quad (2.7)$$

Indeed:

$$\begin{aligned} f(t) * g(t) &= \int_0^t f(\tau)g(t - \tau)d\tau = \int_{-\infty}^{\infty} f(\tau)g(t - \tau)d\tau \\ u = t - \tau &\Rightarrow f(t) * g(t) = -\int_{\infty}^{-\infty} f(t - u)g(u)du \\ \Leftrightarrow f(t) * g(t) &= \int_{-\infty}^{\infty} f(t - u)g(u)du \\ &= \int_0^t f(t - u)g(u)du = g(t) * f(t) \end{aligned} \quad (2.8)$$

Figure 2.2: Unit step function $\Gamma(t)$

As far as the convolution product is concerned, the Dirac delta function satisfies the following property:

$$f(t) * \delta(t - t_0) = f(t - t_0) \quad \forall t_0 \in \mathbb{R} \quad (2.9)$$

Taking $t_0 = 0$ we get:

$$f(t) * \delta(t) = f(t) \quad (2.10)$$

From the preceding equation it is clear that the Dirac delta function is the neutral element of the convolution product.

2.2.2 Unit step function

The unit step function (or Heaviside step function) which will be denoted $\Gamma(t)$ is a discontinuous function whose value is zero for negative values of t and one for positive values of t .

$$\Gamma(t) = \begin{cases} 0 & \forall t < 0 \\ 1 & \forall t > 0 \end{cases} \quad (2.11)$$

The schematic representation of the unit step function is depicted on figure 2.2.

The derivative of the Unit step function involves the Dirac delta function:

$$\frac{d}{dt}\Gamma(t) = \delta(t) \quad (2.12)$$

2.2.3 Complex exponential function

The complex exponential function with natural frequency ω_0 is a signal $x(t)$ where $t \in \mathbb{R}$ and $x(t) \in \mathbb{C}$ defined by:

$$x(t) = e^{j\omega_0 t} \quad (2.13)$$

The complex exponential function verifies the law of exponents:

$$e^{j(\omega_1 + \omega_2)t} = e^{j\omega_1 t} e^{j\omega_2 t} \quad (2.14)$$

The Euler's formula gives the relation between the complex exponential function and $\cos()$ and $\sin()$ functions:

$$e^{j\omega_0 t} = \cos(\omega_0 t) + j \sin(\omega_0 t) \quad (2.15)$$

The complex exponential function is used to compute the frequency response of linear and time invariant systems which will be discussed in the next chapter.

2.3 Definition

The Laplace transform enables to solve linear differential equations with initial conditions. More specifically it transforms a constant coefficients linear differential equation into an algebraic (i.e. polynomial) equation.

The Laplace transform of a continuous time function $x(t)$ which is causal (i.e. $x(t) = 0 \forall t < 0$) is a function of the complex variable s and reads:

$$\mathcal{L}[x(t)] \equiv X(s) = \int_0^{\infty} x(t)e^{-st} dt \quad (2.16)$$

As far as the integral involved in the Laplace transform is limited to positive values of t (indeed $x(t)$ is assumed to be causal) the integral converges as soon as $x(t)$ has an exponential order when $t \rightarrow +\infty$. This means that there exists three positive constants T , M and σ_0 such that:

$$|x(t)| < Me^{\sigma_0 t} \forall t > T \quad (2.17)$$

Consequently the integral involved in the Laplace transform converges as soon as the complex variable s which is used in the Laplace transform is chosen such that its real part is greater than the abscissa of convergence σ_0 :

$$\text{Re}(s) > \sigma_0 \Rightarrow \int_0^{\infty} x(t)e^{-st} dt < \infty \quad (2.18)$$

From a mathematical point of view the Laplace transform is limited to positive values of t because the integral of $e^{-\sigma t}$ where $\sigma > \sigma_0$ is divergent when $t \rightarrow -\infty$. As a consequence the Laplace transform is limited to causal signals, i.e. signals which are null for $t < 0$. It is worth noticing that the abscissa of convergence σ_0 is never computed from a practical point of view.

In addition it is worth noticing that the Laplace transform is a linear operator, meaning that if $x_1(t)$ and $x_2(t)$ are two causal signals and a_1 and a_2 two constants then the Laplace transform of the linear combination between $x_1(t)$ and $x_2(t)$ is the linear combination of the Laplace transform of $x_1(t)$ and $x_2(t)$:

$$\mathcal{L}[a_1 x_1(t) + a_2 x_2(t)] = a_1 \mathcal{L}[x_1(t)] + a_2 \mathcal{L}[x_2(t)] = a_1 X_1(s) + a_2 X_2(s) \quad (2.19)$$

Example 2.1. Compute the Laplace transform of the causal exponential function $x(t) = e^{-at} \forall t \geq 0$.

The Laplace transform of e^{-at} is given by:

$$\mathcal{L}[e^{-at}] = \int_0^{\infty} e^{-at} e^{-st} dt = - \left. \frac{e^{-(s+a)t}}{s+a} \right|_{t=0}^{\infty} \quad (2.20)$$

As far as the real part of the complex variable s is assumed to be greater than the abscissa of convergence σ_0 (which is never computed from a practical point of view), the value of $e^{-(s+a)t}$ converges to 0 when $t \rightarrow \infty$. Consequently the Laplace transform of e^{-at} reduces to:

$$\mathcal{L}[e^{-at}] = \frac{1}{s+a} \quad (2.21)$$

Notice that when $a = 0$ this reduces to the Laplace transform of the unit step function $\Gamma(t)$:

$$\Gamma(t) = e^{-at} \Big|_{a=0} \Rightarrow \mathcal{L}[\Gamma(t)] = \frac{1}{s+a} \Big|_{a=0} = \frac{1}{s} \quad (2.22)$$

■

2.4 Properties

This section presents the main properties of the Laplace transform without demonstration. In the following $x(t)$ is a causal signal and $X(s)$ its Laplace transform.

2.4.1 Linearity

For any signals $x_1(t)$ and $x_2(t)$ and any constants a_1 and a_2 in \mathbb{R} or \mathbb{C} :

$$\mathcal{L}[a_1 x_1(t) + a_2 x_2(t)] = a_1 X_1(s) + a_2 X_2(s) \quad (2.23)$$

This property comes from the linearity of integration.

Example 2.2. Compute the Laplace transform of the causal signal $x(t) = \cos(\omega_0 t) \forall t \geq 0$.

Using the Euler's formula we get:

$$\cos(\omega_0 t) = \frac{e^{j\omega_0 t} + e^{-j\omega_0 t}}{2} \quad (2.24)$$

Using the linearity property, the Laplace transform of $\cos(\omega_0 t)$ is given by:

$$\begin{aligned} \mathcal{L}[\cos(\omega_0 t)] &= X(s) = \int_0^{\infty} \frac{e^{j\omega_0 t} + e^{-j\omega_0 t}}{2} e^{-st} dt \\ &= \frac{1}{2} \int_0^{\infty} e^{j\omega_0 t} e^{-st} dt + \frac{1}{2} \int_0^{\infty} e^{-j\omega_0 t} e^{-st} dt \\ &= \frac{1}{2} \int_0^{\infty} e^{(j\omega_0 - s)t} dt + \frac{1}{2} \int_0^{\infty} e^{-(j\omega_0 + s)t} dt \\ &= \frac{1}{2} \left(\left. \frac{e^{(j\omega_0 - s)t}}{j\omega_0 - s} \right|_{t=0}^{\infty} - \left. \frac{e^{-(j\omega_0 + s)t}}{j\omega_0 + s} \right|_{t=0}^{\infty} \right) \end{aligned} \quad (2.25)$$

As far as the real part of the complex variable s is assumed to be greater than the abscissa of convergence σ_0 (which is never computed from a practical

point of view), the values of $e^{(j\omega_0-s)t}$ and $e^{-(j\omega_0+s)t}$ converge to 0 when $t \rightarrow \infty$. Consequently the Laplace transform of $\cos(\omega_0 t)$ reduces to:

$$\begin{aligned}\mathcal{L}[\cos(\omega_0 t)] &= \frac{1}{2} \left(-\frac{1}{j\omega_0-s} + \frac{1}{j\omega_0+s} \right) \\ &= \frac{1}{2} \left(\frac{-j\omega_0-s}{-\omega_0^2-s^2} + \frac{j\omega_0-s}{-\omega_0^2-s^2} \right)\end{aligned}\quad (2.26)$$

We finally get:

$$\mathcal{L}[\cos(\omega_0 t)] = \frac{s}{s^2 + \omega_0^2} \quad (2.27)$$

■

2.4.2 Derivation

$$\mathcal{L} \left[\frac{d}{dt} x(t) \right] = sX(s) - x(0^+) \quad (2.28)$$

For the second time derivative we have:

$$\mathcal{L} \left[\frac{d^2}{dt^2} x(t) \right] = s\mathcal{L} \left[\frac{d}{dt} x(t) \right] - \dot{x}(0^+) = s^2 X(s) - sx(0^+) - \dot{x}(0^+) \quad (2.29)$$

And so on:

$$\mathcal{L} \left[\frac{d^n}{dt^n} x(t) \right] = s^n X(s) - s^{n-1} x(0^+) - s^{n-2} \dot{x}(0^+) - \dots - \dot{x}^{(n-1)}(0^+) \quad (2.30)$$

Example 2.3. Use the expression of the Laplace transform of $\cos(\omega_0 t)$ to compute the Laplace transform of $\sin(\omega_0 t) \forall t \geq 0$.

We have:

$$\frac{d}{dt} \cos(\omega_0 t) = -\omega_0 \sin(\omega_0 t) \quad (2.31)$$

Then using the property of the Laplace transform with respect to the derivative we get:

$$\begin{aligned}\mathcal{L}[\sin(\omega_0 t)] &= -\frac{1}{\omega_0} \mathcal{L} \left[\frac{d}{dt} \cos(\omega_0 t) \right] \\ &= -\frac{1}{\omega_0} (s\mathcal{L}[\cos(\omega_0 t)] - \cos(\omega_0 t)|_{t=0^+}) \\ &= -\frac{1}{\omega_0} \left(\frac{s^2}{s^2 + \omega_0^2} - 1 \right) = -\frac{1}{\omega_0} \left(\frac{s^2}{s^2 + \omega_0^2} - \frac{s^2 + \omega_0^2}{s^2 + \omega_0^2} \right)\end{aligned}\quad (2.32)$$

We finally get:

$$\mathcal{L}[\sin(\omega_0 t)] = \frac{\omega_0}{s^2 + \omega_0^2} \quad (2.33)$$

■

2.4.3 Integration

$$\mathcal{L} \left[\int x(\tau) d\tau \right] = \frac{X(s)}{s} + \frac{1}{s} \int x(\tau) d\tau \Big|_{\tau=0} \quad (2.34)$$

Note that the integral shall be computed without taking into account the integration constant.

Example 2.4. Use the expression of the Laplace transform of $\sin(\omega_0 t)$ to compute the Laplace transform of $\cos(\omega_0 t) \forall t \geq 0$.

Without taking into account the integration constant we have:

$$\int \sin(\omega_0 \tau) d\tau = -\frac{\cos(\omega_0 \tau)}{\omega_0} \quad (2.35)$$

Then using the property of the Laplace transform with respect to the integration we get:

$$\begin{aligned} \mathcal{L}[\cos(\omega_0 t)] &= -\omega_0 \mathcal{L}\left[\int \sin(\omega_0 \tau) d\tau\right] \\ &= -\omega_0 \left(\frac{\mathcal{L}[\sin(\omega_0 t)]}{s} + \frac{1}{s} \int \sin(\omega_0 \tau) d\tau \Big|_{\tau=0} \right) \\ &= -\omega_0 \left(\frac{\omega_0}{s(s^2 + \omega_0^2)} - \frac{1}{s} \frac{\cos(\omega_0 \tau)}{\omega_0} \Big|_{\tau=0} \right) \\ &= -\omega_0 \left(\frac{\omega_0}{s(s^2 + \omega_0^2)} - \frac{1}{\omega_0 s} \right) \\ &= -\frac{\omega_0^2}{s(s^2 + \omega_0^2)} + \frac{1}{s} \\ &= -\frac{\omega_0^2}{s(s^2 + \omega_0^2)} + \frac{s^2 + \omega_0^2}{s(s^2 + \omega_0^2)} \end{aligned} \quad (2.36)$$

We finally get:

$$\mathcal{L}[\cos(\omega_0 t)] = \frac{s}{s^2 + \omega_0^2} \quad (2.37)$$

■

2.4.4 Multiplication by t

Assuming that n is a positive integer then:

$$\mathcal{L}[t^n x(t)] = (-1)^n \frac{d^n}{ds^n} X(s) \quad (2.38)$$

2.4.5 Time domain translation (delay)

It can be shown that:

$$\mathcal{L}[x(t - T)] = e^{-sT} X(s) \quad (2.39)$$

It is worth noticing time delay does not yield to rational function in the Laplace domain. In order to get a rational function in the Laplace domain the usual approximations of e^{-sT} are the following:

- One method is to write $e^{-sT} = \frac{e^{-\frac{sT}{2}}}{e^{\frac{sT}{2}}}$ and to use the Taylor series expansion of $e^{-\frac{sT}{2}}$ and $e^{\frac{sT}{2}}$. We recall that:

$$e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \dots = \sum_{k=0}^{\infty} \frac{x^k}{k!} \quad (2.40)$$

Consequently e^{-sT} can be approximated by the following rational function where the polynomials at the numerator and the denominator have the same degree. This is set to an arbitrary integer denoted n hereafter:

$$e^{-sT} = \frac{e^{-\frac{sT}{2}}}{e^{\frac{sT}{2}}} \approx \frac{\sum_{k=0}^n \frac{1}{k!} \left(-\frac{sT}{2}\right)^k}{\sum_{k=0}^n \frac{1}{k!} \left(\frac{sT}{2}\right)^k} \quad (2.41)$$

Usually $n = 1$ and we get:

$$e^{-sT} \approx \frac{1 - \frac{sT}{2}}{1 + \frac{sT}{2}} \quad (2.42)$$

– An other method is to approximate e^{-sT} by the following rational function:

$$e^{-sT} \approx \left(\frac{1 - \frac{sT}{2n}}{1 + \frac{sT}{2n}} \right)^n \quad n = 1, 2, \dots \quad (2.43)$$

– Padé approximation consists in identifying the $2n + 1$ first terms of the Taylor series expansion of e^{-sT} with the $2n + 1$ first terms of the Taylor series expansion of the following rational function $\frac{N(sT)}{D(sT)}$ around $sT = 0$:

$$\frac{N(sT)}{D(sT)} = \frac{b_n(sT)^n + b_{n-1}(sT)^{n-1} + \dots + b_1 sT + b_0}{a_n(sT)^n + a_{n-1}(sT)^{n-1} + \dots + a_1 sT + 1} \quad (2.44)$$

It is worth noticing that the coefficient a_0 has been set to 1 without loss of generality.

From a practical point of view the Taylor series expansion of $\frac{N(sT)}{D(sT)}$ around $sT = 0$ can be obtained as the quotient of the polynomial long division, that is the division according to the increasing power of s (this is very close to the Euclidean division, but on the other way):

$$\frac{\begin{array}{l} b_0 + b_1 sT + \dots + b_n(sT)^n \\ - (b_0 + b_0 a_1 sT + \dots + b_0 a_n (sT)^n) \\ \hline (b_1 - b_0 a_1) sT + \dots \end{array}}{\left| \frac{1 + a_1 sT + \dots + a_n (sT)^n}{b_0 + (b_1 - b_0 a_1) sT + \dots} \right.} \quad (2.45)$$

The $2n + 1$ coefficients $(b_n, \dots, b_1, b_0, a_n, \dots, a_1)$ are obtained by identifying each coefficient of $(sT)^k$, $k = 0, \dots, 2n$ in the following equality:

$$b_0 + (b_1 - b_0 a_1) sT + \dots = 1 - sT + \frac{1}{2!}(sT)^2 + \dots \approx e^{-sT} \quad (2.46)$$

It can be shown that Padé approximation of e^{-sT} reads:

$$e^{-sT} \approx \frac{\sum_{i=0}^n q_i (-sT)^i}{\sum_{i=0}^n q_i (sT)^i} \quad \text{where } q_i = \frac{(2n - i)!n!}{2n!i!(n - i)!} \quad (2.47)$$

Padé approximation of e^{-sT} for degree $n = 1, 2$ and 3 is given on Table 2.1.

Degree n	Padé approximation of e^{-sT}
1	$\frac{1 - \frac{sT}{2}}{1 + \frac{sT}{2}}$
2	$\frac{1 - \frac{sT}{2} + \frac{(sT)^2}{12}}{1 + \frac{sT}{2} + \frac{(sT)^2}{12}}$
3	$\frac{1 - \frac{sT}{2} + \frac{(sT)^2}{10} - \frac{(sT)^3}{120}}{1 + \frac{sT}{2} + \frac{(sT)^2}{10} + \frac{(sT)^3}{120}}$

Table 2.1: Padé approximation of e^{-sT}

2.4.6 Frequency domain translation

It can be shown that:

$$\mathcal{L} [e^{-\omega_0 t} x(t)] = X(s + \omega_0) \quad (2.48)$$

Example 2.5. Use the expression of the Laplace transform of $\Gamma(t)$ to compute the Laplace transform of $e^{-at} \forall t \geq 0$.

We get:

$$\mathcal{L} [\Gamma(t)] = X(s) = \frac{1}{s} \Rightarrow \mathcal{L} [e^{-at}] = \mathcal{L} [e^{-at} \Gamma(t)] = X(s + a) = \frac{1}{s + a} \quad (2.49)$$

■

2.4.7 Initial value theorem

Assuming that the value of $x(t)$ exists when $t \rightarrow 0^+$ then:

$$\lim_{t \rightarrow 0^+} x(t) = \lim_{s \rightarrow \infty} sX(s) \quad (2.50)$$

Example 2.6. Knowing that the Laplace transform of $\cos(\omega_0 t)$ is $\frac{s}{s^2 + \omega_0^2}$ compute $\lim_{t \rightarrow 0^+} \cos(\omega_0 t)$.

We check that:

$$\lim_{t \rightarrow 0^+} \cos(\omega_0 t) = \lim_{s \rightarrow \infty} s \frac{s}{s^2 + \omega_0^2} = \lim_{s \rightarrow \infty} \frac{s^2}{s^2 + \omega_0^2} = 1 \quad (2.51)$$

Obviously we rediscover the fact that $\lim_{t \rightarrow 0^+} \cos(\omega_0 t) = \cos(0) = 1$.

■

Example 2.7. Knowing that the Laplace transform of $\sin(\omega_0 t)$ is $\frac{\omega_0}{s^2 + \omega_0^2}$ compute $\lim_{t \rightarrow 0^+} \frac{d}{dt} \sin(\omega_0 t)$.

Using the derivative property we check that:

$$\begin{aligned} \lim_{t \rightarrow 0^+} \frac{d}{dt} \sin(\omega_0 t) &= \lim_{s \rightarrow \infty} s \left(s \frac{\omega_0}{s^2 + \omega_0^2} - \sin(\omega_0 t) \Big|_{t=0^+} \right) \\ &= \lim_{s \rightarrow \infty} \frac{s^2 \omega_0}{s^2 + \omega_0^2} \\ &= \omega_0 \end{aligned} \quad (2.52)$$

Obviously we rediscover the fact that $\lim_{t \rightarrow 0^+} \frac{d}{dt} \sin(\omega_0 t) = \omega_0 \cos(\omega_0 t) \Big|_{t=0^+} = \omega_0$.

■

2.4.8 Final value theorem

Assuming that the value of $x(t)$ exists when $t \rightarrow \infty$ then:

$$\lim_{t \rightarrow \infty} x(t) = \lim_{s \rightarrow 0} sX(s) \quad (2.53)$$

Example 2.8. Knowing that the Laplace transform of the unit step function $\Gamma(t)$ is $\frac{1}{s}$ compute $\lim_{t \rightarrow \infty} \Gamma(t)$.

We check that:

$$\lim_{t \rightarrow \infty} \Gamma(t) = \lim_{s \rightarrow 0} s \frac{1}{s} = \lim_{s \rightarrow 0} 1 = 1 \quad (2.54)$$

■

It is worth noticing that the final value theorem gives the correct answer as soon as the value of $x(t)$ exists when $t \rightarrow \infty$. For example if we apply the final value theorem on $\cos(\omega_0 t)$ which has no limit when $t \rightarrow \infty$ then result is not correct.

2.4.9 Convolution

We have seen that the convolution product $*$ between two causal signals $x(t)$ and $y(t)$ is defined by:

$$x(t) * y(t) = \int_0^t x(\tau)y(t - \tau)d\tau \quad (2.55)$$

Then the Laplace transform of the convolution product between two causal signals $x(t)$ and $y(t)$ is the product of the Laplace transforms of $x(t)$ and $y(t)$:

$$\mathcal{L}[x(t) * y(t)] = X(s)Y(s) \quad (2.56)$$

2.4.10 Evaluating improper integrals

It can be shown¹ that the following property holds to evaluate improper integrals thanks to Laplace transform:

$$\int_0^{\infty} \frac{y(t)}{t} dt = \int_0^{\infty} Y(s) ds \quad (2.57)$$

2.4.11 Link with Fourier transform

The Fourier transform is widely used in communication theory to assess spectral properties of a signal. The Fourier transform $X(\omega)$ of a signal $x(t)$ is defined as follows:

$$X(\omega) = \mathcal{F}[x(t)] = \int_{-\infty}^{+\infty} x(t)e^{-j\omega t} dt \quad (2.58)$$

Compared with the Laplace transform the integral involved in the Fourier transform starts at $-\infty$ and the complex variable s in the exponential which

¹https://en.wikipedia.org/wiki/Laplace_transform

$x(t) \forall t \geq 0$	$X(s)$
$\delta(t)$	1
$\Gamma(t)$ (step function)	$\frac{1}{s}$
t	$\frac{1}{s^2}$
t^n	$\frac{n!}{s^{n+1}}$
e^{-at}	$\frac{1}{s+a}$
$\frac{t^{n-1}}{(n-1)!}e^{-at}$	$\frac{1}{(s+a)^n}$
$a^{\frac{t}{T}}$	$\frac{1}{s - \frac{\ln(a)}{T}}$
$\sin(\omega_0 t)$	$\frac{\omega_0}{s^2 + \omega_0^2}$
$e^{-at} \sin(\omega_0 t)$	$\frac{\omega_0}{(s+a)^2 + \omega_0^2}$
$\cos(\omega_0 t)$	$\frac{s}{s^2 + \omega_0^2}$
$e^{-at} \cos(\omega_0 t)$	$\frac{s+a}{(s+a)^2 + \omega_0^2}$

Table 2.2: Usual Laplace transform pairs

appears in the Laplace transform is replaced by a pure imaginary number $j\omega$ in the Fourier transform. From those properties defining the integral of the two transforms we obtain formulas for getting from one to the other:

– If $x(t)$ is even then:

$$\mathcal{F}[x(t)] = \mathcal{L}[x(t)]_{s=j\omega} + \mathcal{F}[x(t)]_{s=-j\omega} \quad (2.59)$$

– If $x(t)$ is odd then:

$$\mathcal{F}[x(t)] = \mathcal{L}[x(t)]_{s=j\omega} - \mathcal{F}[x(t)]_{s=-j\omega} \quad (2.60)$$

2.5 Usual Laplace transform pairs

Usual Laplace transform pairs are provided in Table 2.2.

2.6 Inverse Laplace transform

In the following $G(s)$ is a rational expression involving polynomials in its numerator and denominator which are respectively denoted $N(s)$ and $D(s)$:

$$G(s) = \frac{N(s)}{D(s)} \quad (2.61)$$

We will assume in the following that the degree of $D(s)$ is strictly greater than the degree of $N(s)$; then $G(s)$ is said strictly proper.

2.6.1 Partial fraction expansion

The roots of $N(s)$ are called the zeros of $G(s)$ whereas the roots of $D(s)$ are called the poles of $G(s)$. Without loss of generality we will assume that $G(s)$ can be written as follows where $\lambda_1, \dots, \lambda_n$ are the poles of $G(s)$ with multiplicity n_1, \dots, n_n respectively:

$$G(s) = \frac{N(s)}{(s - \lambda_1)^{n_1} \cdots (s - \lambda_n)^{n_n}} \quad (2.62)$$

Then the partial fraction expansion of $G(s)$ is:

$$G(s) = G_1(s) + \cdots + G_k(s) + \cdots + G_n(s) \quad (2.63)$$

Where $G_k(s)$ is the partial fraction expansion of $G(s)$ associated to the pole λ_k of multiplicity n_k ($k = 1, \dots, n$):

$$G_k(s) = \sum_{j_k=1}^{n_k} \frac{b_{j_k}}{(s - \lambda_k)^{j_k}} \quad (2.64)$$

Constant coefficient which multiplies the rational expression $\frac{1}{s - \lambda_k}$ (the power of $s - \lambda_k$ is 1) is called the residue of $G(s)$ around pole $s = \lambda_k$.

Coefficients b_{j_k} are computed thanks to the following formula:

$$b_{j_k} = \frac{1}{(n_k - j_k)!} \left. \frac{d^{n_k - j_k}}{ds^{n_k - j_k}} (s - \lambda_k)^{n_k} G(s) \right|_{s=\lambda_k} \quad k = 1, \dots, n_k \quad (2.65)$$

It is worth noticing that when the multiplicity of pole λ_k is 1 this formula reduces to:

$$n_k = j_k = 1 \Rightarrow b_{j_k} = (s - \lambda_k) G(s) \Big|_{s=\lambda_k} \quad (2.66)$$

Notice that the residues can be computed using the function *residue* with Matlab™ and *pfss* with Scilab.

Once $G(s)$ has been broken up into components:

$$G(s) = G_1(s) + \cdots + G_n(s) \quad (2.67)$$

Then the inverse Laplace transform of $G(s)$ is the causal signal $g(t)$ given by:

$$g(t) = \mathcal{L}^{-1} [G(s)] = \mathcal{L}^{-1} [G_1(s)] + \cdots + \mathcal{L}^{-1} [G_n(s)] = g_1(t) + \cdots + g_n(t) \quad (2.68)$$

The advantage of the partial fraction expansion is that the individual terms of $G(s)$ resulting from the expansion into the partial fraction form are very simple functions of s and their inverse Laplace transform are readily available. We can use for example Table 2.2 of usual Laplace transform pairs.

Example 2.9. Compute the inverse Laplace transform of $G(s) = \frac{s^2 + 2s + 3}{(s+1)^3}$.

Laplace transform $G(s)$ has a single pole $\lambda_1 = -1$ with multiplicity $n_1 = 3$.

The partial fraction expansion of $G(s)$ reads:

$$G(s) = \frac{b_1}{s+1} + \frac{b_2}{(s+1)^2} + \frac{b_3}{(s+1)^3} \quad (2.69)$$

Where:

$$\begin{aligned} b_3 &= \frac{1}{(3-3)!} \frac{d^{3-3}}{ds^{3-3}} (s+1)^3 G(s) \Big|_{s=-1} \\ &= s^2 + 2s + 3 \Big|_{s=-1} = 2 \end{aligned} \quad (2.70)$$

$$\begin{aligned} b_2 &= \frac{1}{(3-2)!} \frac{d^{3-2}}{ds^{3-2}} (s+1)^3 G(s) \Big|_{s=-1} \\ &= \frac{1}{1!} \frac{d}{ds} (s^2 + 2s + 3) \Big|_{s=-1} \\ &= 2s + 2 \Big|_{s=-1} = 0 \end{aligned} \quad (2.71)$$

$$\begin{aligned} b_1 &= \frac{1}{(3-1)!} \frac{d^{3-1}}{ds^{3-1}} (s+1)^3 G(s) \Big|_{s=-1} \\ &= \frac{1}{2!} \frac{d^2}{ds^2} (s^2 + 2s + 3) \Big|_{s=-1} \\ &= \frac{1}{2} \frac{d}{ds} (2s + 2) \Big|_{s=-1} \\ &= \frac{1}{2} 2 \Big|_{s=-1} = 1 \end{aligned} \quad (2.72)$$

Thus the partial fraction expansion of $G(s)$ is:

$$G(s) = \frac{b_1}{s+1} + \frac{b_2}{(s+1)^2} + \frac{b_3}{(s+1)^3} = \frac{1}{s+1} + \frac{2}{(s+1)^3} \quad (2.73)$$

With the use of Table 2.2 of usual Laplace transform pairs we are able to compute the inverse Laplace transform of $G(s)$ as:

$$\mathcal{L}^{-1}[G(s)] = \mathcal{L}^{-1}\left[\frac{1}{s+1}\right] + \mathcal{L}^{-1}\left[\frac{2}{(s+1)^3}\right] = e^{-t} + 2 \frac{t^{3-1}}{(3-1)!} e^{-t} \quad (2.74)$$

That is:

$$g(t) = \mathcal{L}^{-1}[G(s)] = e^{-t} + t^2 e^{-t} = (1 + t^2) e^{-t} \quad \forall t \geq 0 \quad (2.75)$$

■

2.6.2 Mellin-Fourier integral

The Mellin-Fourier integral originates from complex analysis, traditionally known as the theory of functions of a complex variable, and more specifically from the computation of the Mellin-Fourier (or Bromwich) integral which can be evaluated by means of the Cauchy's residue theorem. Those notions are out of the scope of this lecture and we will just use some results to compute the inverse Laplace transform of $G(s)$, assuming that $G(s)$ is a rational expression of polynomial strictly proper (i.e. the degree of the numerator is strictly greater than the degree of the denominator).

The Mellin-Fourier integral reads:

$$\begin{aligned} G(s) &= \frac{N(s)}{D(s)} \text{ where } \deg(N(s)) < \deg(D(s)) \\ \Rightarrow g(t) &= \mathcal{L}^{-1}[G(s)] = \sum_k \text{Res}_{s=\lambda_k} [G(s)e^{st}] \quad \forall t \geq 0 \end{aligned} \quad (2.76)$$

Where the residue $Res_{s=\lambda_k} [G(s)e^{st}]$ shall be computed around each pole λ_k of $G(s)$. Assuming that λ_k is a pole of multiplicity n_k then the residue of $G(s)$ around pole λ_k is given by:

$$Res_{s=\lambda_k} [G(s)e^{st}] = \frac{1}{(n_k - 1)!} \frac{d^{n_k-1}}{ds^{n_k-1}} (s - \lambda_k)^{n_k} G(s)e^{st} \Big|_{s=\lambda_k} \quad (2.77)$$

Example 2.10. Compute the inverse Laplace transform of $G(s) = \frac{s^2+2s+3}{(s+1)^3}$.

Laplace transform $G(s)$ has a single pole $\lambda_1 = -1$ with multiplicity $n_1 = 3$.

Using Mellin-Fourier integral we get:

$$g(t) = \mathcal{L}^{-1} [G(s)] = Res_{s=\lambda_1} [G(s)e^{st}] \quad \forall t \geq 0 \quad (2.78)$$

Where:

$$\begin{aligned} Res_{s=-1} [G(s)e^{st}] &= \frac{1}{(3-1)!} \frac{d^{3-1}}{ds^{3-1}} (s+1)^3 G(s)e^{st} \Big|_{s=-1} \\ &= \frac{1}{2!} \frac{d^2}{ds^2} (s^2 + 2s + 3) e^{st} \Big|_{s=-1} \end{aligned} \quad (2.79)$$

We will first compute the first and second derivative of $(s^2 + 2s + 3) e^{st}$ before evaluating them at $s = -1$:

$$\begin{aligned} \frac{d}{ds} (s^2 + 2s + 3) e^{st} &= (2s + 2) e^{st} + (s^2 + 2s + 3) t e^{st} \\ \Rightarrow \frac{d^2}{ds^2} (s^2 + 2s + 3) e^{st} &= 2e^{st} + (2s + 2) t e^{st} \\ &\quad + (2s + 2) t e^{st} + (s^2 + 2s + 3) t^2 e^{st} \\ &= (2 + 2(2s + 2) t) e^{st} + (s^2 + 2s + 3) t^2 e^{st} \\ \Rightarrow \frac{d^2}{ds^2} (s^2 + 2s + 3) e^{st} \Big|_{s=-1} &= (2 + 2t^2) e^{-t} \end{aligned} \quad (2.80)$$

We finally get:

$$\begin{aligned} g(t) &= Res_{s=-1} [G(s)e^{st}] = \frac{1}{2!} \frac{d^2}{ds^2} (s^2 + 2s + 3) e^{st} \Big|_{s=-1} \\ &= (1 + t^2) e^{-t} \quad \forall t \geq 0 \end{aligned} \quad (2.81)$$

■

Example 2.11. Compute the inverse Laplace transform of $G(s) = \frac{s+2}{(s+3)(s+4)^2}$.

Laplace transform $G(s)$ has two poles:

– $\lambda_1 = -3$ with multiplicity $n_1 = 1$. We get for the residue:

$$\begin{aligned} Res_{s=-3} [G(s)e^{st}] &= \frac{1}{(1-1)!} \frac{d^{1-1}}{ds^{1-1}} (s+3)^1 G(s)e^{st} \Big|_{s=-3} \\ &= \frac{s+2}{(s+4)^2} e^{st} \Big|_{s=-3} \\ &= -e^{-3t} \end{aligned} \quad (2.82)$$

– $\lambda_2 = -4$ with multiplicity $n_2 = 2$. We get for the residue:

$$\begin{aligned} Res_{s=-4} [G(s)e^{st}] &= \frac{1}{(2-1)!} \frac{d^{2-1}}{ds^{2-1}} (s+4)^2 G(s)e^{st} \Big|_{s=-4} \\ &= \frac{d}{ds} \frac{s+2}{s+3} e^{st} \Big|_{s=-4} \\ &= \frac{(e^{st} + (s+2)te^{st})(s+3) - (s+2)e^{st}}{(s+3)^2} \Big|_{s=-4} \\ &= -e^{-4t} + 2te^{-4t} + 2e^{-4t} \\ &= (1 + 2t)e^{-4t} \end{aligned} \quad (2.83)$$

Using Mellin-Fourier integral the inverse Laplace transform of $G(s)$ is obtained by summing the two residues:

$$g(t) = \mathcal{L}^{-1}[G(s)] = \text{Res}_{s=-3}[G(s)e^{st}] + \text{Res}_{s=-4}[G(s)e^{st}] \quad \forall t \geq 0 \quad (2.84)$$

We finally get:

$$g(t) = -e^{-3t} + (1 + 2t)e^{-4t} \quad \forall t \geq 0 \quad (2.85)$$

■

Example 2.12. Use Laplace transform to find the solution to the following differential equation:

$$\begin{cases} 2\dot{y}(t) - y(t) = 2e^{2t} \\ y(0) = 3 \end{cases} \quad (2.86)$$

Taking the Laplace transform of the differential equation yields:

$$\begin{aligned} 2(sY(s) - y(0)) - Y(s) &= \frac{2}{s-2} \\ \Leftrightarrow Y(s)(2s-1) &= \frac{2}{s-2} + 6 = \frac{6s-10}{s-2} \\ \Leftrightarrow Y(s) &= \frac{6s-10}{(2s-1)(s-2)} = \frac{3s-5}{(s-0.5)(s-2)} \end{aligned} \quad (2.87)$$

Then we will compute the inverse Laplace transform of $Y(s)$ thanks to the Mellin-Fourier integral:

$$y(t) = \mathcal{L}^{-1}[Y(s)] = \sum_k \text{Res}_{s=\lambda_k}[Y(s)e^{st}] \quad \forall t \geq 0 \quad (2.88)$$

Laplace transform $Y(s)$ has two poles:

– $\lambda_1 = 0.5$ with multiplicity $n_1 = 1$. We get for the residue:

$$\begin{aligned} \text{Res}_{s=0.5}[Y(s)e^{st}] &= \frac{1}{(1-1)!} \frac{d^{1-1}}{ds^{1-1}} (s-0.5)^1 Y(s)e^{st} \Big|_{s=0.5} \\ &= \frac{3s-5}{s-2} e^{st} \Big|_{s=0.5} \\ &= \frac{7}{3} e^{0.5t} \end{aligned} \quad (2.89)$$

– $\lambda_2 = 2$ with multiplicity $n_2 = 1$. We get for the residue:

$$\begin{aligned} \text{Res}_{s=2}[Y(s)e^{st}] &= \frac{1}{(1-1)!} \frac{d^{1-1}}{ds^{1-1}} (s-2)^1 Y(s)e^{st} \Big|_{s=2} \\ &= \frac{3s-5}{s-0.5} e^{st} \Big|_{s=2} \\ &= \frac{2}{3} e^{2t} \end{aligned} \quad (2.90)$$

We finally get:

$$y(t) = \frac{7}{3} e^{0.5t} + \frac{2}{3} e^{2t} \quad (2.91)$$

We check that $y(0) = 3$. ■

2.7 Norm of vector signal

Let $\underline{x}(t) = \begin{bmatrix} x_1(t) \\ \cdot \\ x_n(t) \end{bmatrix} \in \mathbb{R}^n$ be a vector depending of time $t \geq 0$. Vector $\underline{x}(t)$ is called a vector signal. Assuming that the following integral is defined its represents the L_2 norm of vector signal $\underline{x}(t)$:

$$\|\underline{x}(t)\|_2 = \sqrt{\int_0^\infty \underline{x}^T(t)\underline{x}(t)dt} < \infty \quad (2.92)$$

The L_2 norm of a vector signal is also called its energy. Vector signal $\underline{x}(t)$ is said of finite energy, or equivalently $\underline{x}(t) \in L_2[0, \infty)$ where $L_2[0, \infty)$ is the Hilbert space of finite energy signal defined $\forall t \geq 0$, as soon as the integral in Equation (2.92) exists.

Let $\underline{X}(s)$ be the Laplace transform of vector signal $\underline{x}(t)$. As soon as the vector signal $\underline{x}(t) \in L_2$ its Laplace transform $\underline{X}(s)$ is strictly proper and all its poles have negative real part (or equivalently $\underline{x}(t) \in L_2[0, \infty) \Leftrightarrow \underline{X}(s) \in H_2$ where notation H is used after the mathematician G.H. Hardy). By Parseval's theorem we have:

$$\|\underline{x}(t)\|_2 = \sqrt{\frac{1}{2\pi j} \oint_{\mathcal{C}^-} \underline{X}^T(-s)\underline{X}(s)ds} \quad (2.93)$$

where \mathcal{C}^- is a contour following the imaginary axis of the complex plane and then around an infinite semicircle in the left half plane.

The use of the residue theorem finally leads to the following expression:

$$\|\underline{x}(t)\|_2 = \sqrt{\sum_{\text{poles with negative real part}} \text{Res}(\underline{X}^T(-s)\underline{X}(s))} \quad (2.94)$$

Let λ_i be a pole (with negative real part) of $\underline{X}(s)$ with multiplicity n_i . Residue $\text{Res}(\underline{X}^T(-s)\underline{X}(s))$ on pole λ_i is defined as follows:

$$\text{Res}_{s=\lambda_i}(\underline{X}^T(-s)\underline{X}(s)) = \frac{1}{(n_i - 1)!} \frac{d^{n_i-1}}{ds^{n_i-1}}(s - \lambda_i)^{n_i} \underline{X}^T(-s)\underline{X}(s) \Big|_{s=\lambda_i} \quad (2.95)$$

The L_∞ norm of vector signal $\underline{x}(t)$ is defined by:

$$\|\underline{x}(t)\|_\infty = \sup_{t \in \mathbb{R}^+} \max_i |x_i(t)| \quad (2.96)$$

Example 2.13. Let $x(t)$ be defined as follows where τ is a positive time constant:

$$x(t) = e^{-\frac{t}{\tau}} \quad \forall t \geq 0 \quad (2.97)$$

The square of the L_2 norm of signal $\underline{x}(t)$ is:

$$\begin{aligned} \|\underline{x}(t)\|_2^2 &= \int_0^\infty \underline{x}^T(t)\underline{x}(t)dt \\ &= \int_0^\infty e^{-\frac{2t}{\tau}} dt \\ &= \frac{\tau}{2} e^{-\frac{2t}{\tau}} \Big|_0^\infty \\ &= \frac{\tau}{2} \end{aligned} \quad (2.98)$$

And consequently:

$$\|\underline{x}(t)\|_2 = \sqrt{\frac{\tau}{2}} \quad (2.99)$$

The square of the L_2 norm of signal $\underline{x}(t)$ can equivalently be computed using the Laplace transform $X(s)$ of $x(t)$:

$$\begin{aligned} X(s) &= \frac{1}{s + \frac{1}{\tau}} = \frac{\tau}{1 + s\tau} \\ \Rightarrow X(-s)X(s) &= \frac{\tau^2}{(1 + s\tau)(1 - s\tau)} \end{aligned} \quad (2.100)$$

Assuming $\tau > 0$ the unique pole $\lambda_1 = -\frac{1}{\tau}$ of $X(s)$ has a negative real part and its multiplicity n_1 is equal to 1. Then residue of $\underline{X}^T(-s)\underline{X}(s)$ on pole $\lambda_1 = -\frac{1}{\tau}$ is computed as follows:

$$\begin{aligned} \text{Res}_{s=-\frac{1}{\tau}} (X^T(-s)X(s)) &= \frac{1}{(1-1)!} \frac{d^{1-1}}{ds^{1-1}} (s - \lambda_i)^{n_i} X^T(-s)X(s) \Big|_{s=-\frac{1}{\tau}} \\ &= \left(s + \frac{1}{\tau} \right) \frac{\tau^2}{(1+s\tau)(1-s\tau)} \Big|_{s=-\frac{1}{\tau}} \\ &= \frac{\tau}{1-s\tau} \Big|_{s=-\frac{1}{\tau}} \\ &= \frac{\tau}{2} \end{aligned} \quad (2.101)$$

We finally get:

$$\|\underline{x}(t)\|_2 = \sqrt{\sum_{\text{poles with negative real part}} \text{Res} (\underline{X}^T(-s)\underline{X}(s))} = \sqrt{\frac{\tau}{2}} \quad (2.102)$$

The L_∞ norm of signal $x(t)$ is defined by:

$$\|\underline{x}(t)\|_\infty = \sup_{t \in \mathbb{R}^+} |e^{-\frac{t}{\tau}}| = 1 \quad (2.103)$$

■

2.8 Mellin transform

The mellin transform of a continuous time function $x(\tau)$ which is causal (i.e. $x(\tau) = 0 \forall \tau < 0$) is a function of the complex variable s and reads:

$$\mathcal{M}[x(\tau)] = \int_0^\infty x(\tau) \tau^{s-1} d\tau \quad (2.104)$$

Mellin's transform is closely related to an extended form of Laplace's. Indeed let's consider the following change of variables:

$$\tau = e^{-t} \Leftrightarrow t = -\ln(\tau) \quad (2.105)$$

This change of variables transforms the integral (2.104) into:

$$\begin{aligned} d\tau = -e^{-t} dt \Rightarrow \mathcal{M}[x(e^{-t})] &= - \int_{+\infty}^{-\infty} x(e^{-t}) (e^{-t})^{s-1} e^{-t} dt \\ &= \int_{-\infty}^{+\infty} x(e^{-t}) e^{-st} dt \end{aligned} \quad (2.106)$$

The previous relation defines the *two-sided* Laplace transform of $y(t) = x(e^{-t})$ defined by:

$$\mathcal{L}_{2s}[y(t)] = \int_{-\infty}^{+\infty} y(t)e^{-st} dt \quad (2.107)$$

This can be written symbolically as:

$$\mathcal{M}[x(\tau)] = \mathcal{L}_{2s}[x(e^{-t})] \quad (2.108)$$

Chapter 3

Linear Time Invariant systems

3.1 Chapter overview

This chapter introduces linear time invariant (LTI) systems. First the properties of linear time invariant (LTI) systems are presented. Then concept of transfer function is presented and the time response as well as the frequency response of linear time invariant systems are discussed. The last part of this chapter is dedicated to model reduction where we will see that first and second order models constitute the ultimate approximation of a system.

3.2 Properties

Linear and Time Invariant (LTI) systems satisfy the two following properties:

- Linearity: this means that the relation between the input and the output of the system is a linear: if input $u_1(t)$ produces response $y_1(t)$ and input $u_2(t)$ produces response $y_2(t)$ then the linear combination of the inputs $a_1u_1(t) + a_2u_2(t)$ produces the linear combination of the outputs $a_1y_1(t) + a_2y_2(t)$ where a_1 and a_2 are real scalars. Denoting by \mathcal{S} the system operator we get:

$$\begin{aligned} & \begin{cases} \mathcal{S}[u_1(t)] = y_1(t) \\ \mathcal{S}[u_2(t)] = y_2(t) \end{cases} \\ \Rightarrow \mathcal{S}[a_1u_1(t) + a_2u_2(t)] &= a_1\mathcal{S}[u_1(t)] + a_2\mathcal{S}[u_2(t)] \\ \Leftrightarrow \mathcal{S}[a_1u_1(t) + a_2u_2(t)] &= a_1y_1(t) + a_2y_2(t) \end{aligned} \quad (3.1)$$

Linearity leads to the superposition principle. The superposition principle states that, for all linear systems, the net response caused by two or more inputs is the sum of the responses which are caused by each input individually.

- Time invariance: this means that if the output due to input $u(t)$ is $y(t)$ at time t then the output due to input $u(t - T)$ after a delay of T will be $y(t - T)$:

$$\mathcal{S}[u(t)] = y(t) \Rightarrow \mathcal{S}[u(t - T)] = y(t - T) \quad \forall T \geq 0 \quad (3.2)$$

We will see that a linear system is time invariant as soon as initial conditions are null.

3.3 Convolution integral

The time response of a linear system to a causal input $u(t)$ ($u(t) = 0 \forall t < 0$) can be computed as follows:

- First use the fact that the delta Dirac function is the neutral element for the convolution product:

$$u(t) = u(t) * \delta(t) = \int_0^\infty u(\tau)\delta(t - \tau)d\tau \quad (3.3)$$

- Then apply the system operator \mathcal{S} on the input $u(t)$:

$$\mathcal{S}[u(t)] = \mathcal{S}\left[\int_0^\infty u(\tau)\delta(t - \tau)d\tau\right] \quad (3.4)$$

- Using the fact that the system is linear we can reverse \int_0^∞ and \mathcal{S} operators:

$$\mathcal{S}\left[\int_0^\infty u(\tau)\delta(t - \tau)d\tau\right] = \int_0^\infty \mathcal{S}[u(\tau)\delta(t - \tau)]d\tau \quad (3.5)$$

- In the expression $\mathcal{S}[u(\tau)\delta(t - \tau)]$ the only term with depends on time t is $\delta(t - \tau)$. Indeed term $u(\tau)$ is viewed as an independent term of time t because the integral is envisioned as an infinite sum for different values of τ . Thus as far as the system is linear we get:

$$\mathcal{S}[u(\tau)\delta(t - \tau)] = u(\tau)\mathcal{S}[\delta(t - \tau)] \quad (3.6)$$

- Let's us define $h(t)$ as the time response of the system to the delta Dirac function; this is the impulse response of the system:

$$h(t) = \mathcal{S}[\delta(t)] \quad (3.7)$$

As soon as the system is time invariant we get:

$$\mathcal{S}[\delta(t - \tau)] = h(t - \tau) \quad (3.8)$$

- Consequently the system time response can be written as:

$$\mathcal{S}[u(t)] = \int_0^\infty u(\tau)\mathcal{S}[\delta(t - \tau)]d\tau = \int_0^\infty u(\tau)h(t - \tau)d\tau \quad (3.9)$$

Thus we have showed that a Linear and Time Invariant (LTI) system is basically a convolutor, meaning that the relation between the input $u(t)$ and

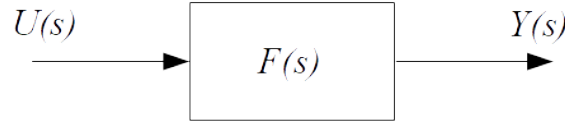


Figure 3.1: Block diagram representation of a LTI system

the output $y(t)$ of the system is obtained thanks to a convolution product. Denoting by $h(t)$ the impulse response of the system we get:

$$h(t) = \mathcal{S}[\delta(t)] \Rightarrow y(t) = \mathcal{S}[u(t)] = u(t) * h(t) = \int_0^{\infty} u(\tau)h(t - \tau)d\tau \quad (3.10)$$

Using the fact that the convolution product is commutative the preceding relation can also be written as follows:

$$y(t) = h(t) * u(t) = \int_0^{\infty} h(\tau)u(t - \tau)d\tau \quad (3.11)$$

The convolution product is usually quite cumbersome to compute. Hopefully the Laplace transform changes the convolution product into a simple product, meaning that the Laplace transform of the system output is obtained thanks to the simple product between the Laplace transform of the system impulse response and the Laplace transform of the input signal:

$$\begin{cases} \mathcal{L}[u(t)] := U(s) \\ \mathcal{L}[h(t)] := F(s) \\ \mathcal{L}[y(t)] := Y(s) \end{cases} \Rightarrow Y(s) = F(s)U(s) \quad (3.12)$$

The Laplace transform of the impulse response $h(t)$ of the system, which is denoted $F(s)$, is called the transfer function of the system. Figure 3.1 provides the block diagram representation of a linear time invariant (LTI) system.

3.4 System defined by a linear differential equation

We have seen that a nonlinear and time invariant system can be approximated by a linear differential equation with constant coefficients. Taking the Laplace transform of (1.17) and assuming *zero initial conditions* we get:

$$\begin{aligned} a_0Y(s) + a_1sY(s) + \cdots + a_{n-1}s^{n-1}Y(s) + a_ns^nY(s) = \\ b_0U(s) + b_1sU(s) + \cdots + b_{m-1}s^{m-1}U(s) + b_ms^mU(s) \end{aligned} \quad (3.13)$$

Thus the transfer function of the system is defined by:

$$F(s) = \frac{Y(s)}{U(s)} = \frac{b_0 + b_1s + \cdots + b_{m-1}s^{m-1} + b_ms^m}{a_0 + a_1s + \cdots + a_{n-1}s^{n-1} + a_ns^n} \quad (3.14)$$

Notice that the coefficient a_n is equal to 1 without loss of generality.

The denominator of the transfer function is called the characteristic polynomial and the order of the model is the degree of the characteristic polynomial (which is n for the transfer function (3.14)). In addition the transfer function $F(s)$ is said to be strictly proper if the degree of its denominator is strictly greater than the degree of its numerator, i.e. $n > m$.

Coming back to the time domain and having in mind that the Laplace transform of the convolution product between two causal signals $h(t)$ and $u(t)$ is the simple multiplication between the Laplace transform of $h(t)$ and $u(t)$ we get:

$$Y(s) = F(s)U(s) \Leftrightarrow y(t) = h(t) * u(t) \quad (3.15)$$

As a consequence a system defined by a linear differential equation is a Linear and Time Invariant system (LTI) as soon as *initial conditions are equal to zero*.

If $u(t)$ is a causal signal then $\lim_{s \rightarrow \infty} U(s) < \infty$. Similarly if $F(s)$ is a fraction involving polynomials in its numerator and denominator then the corresponding system is said to be causal when $\lim_{s \rightarrow \infty} F(s) < \infty$. This means that the degree of the denominator is greater than or equal to the degree of the numerator. From a practical point of view a system is said to be causal when the output depends on past and current inputs but not on future inputs (non-anticipative system).

Settling time t_s is the time required by the system response to reach and stay within a range about the final value. The range is specified by absolute percentage of the final value which is usually 5%.

The static gain G_s of a continuous time system is the limit as $t \rightarrow \infty$ of the rational expression between the output signal $y(t)$ and the input signal $u(t)$. Assuming that the limit exists and applying the final value theorem leads to the following expression of the static gain:

$$G_s = \lim_{t \rightarrow \infty} \frac{y(t)}{u(t)} = \lim_{s \rightarrow 0} \frac{sY(s)}{sU(s)} = \lim_{s \rightarrow 0} \frac{Y(s)}{U(s)} = F(0) \quad (3.16)$$

3.5 Frequency response

In this section we will focus on *stable* linear time invariant systems, meaning that all the poles of the transfer function have negative real part. Stability analysis will be tackled in the next chapter.

3.5.1 Relation with transfer function

The frequency response of a system is the response of the system to the causal input $u(t) = e^{j\omega t}$ where ω is the frequency of the complex exponential function. More specifically the frequency response consists in the magnitude and phase relation between the input and the output signal when the input signal $u(t)$ is the causal complex exponential function $e^{j\omega t}$.

We recall that a linear system is a convolutor. Specializing relation (3.11)

to the case where $u(t)$ is the complex exponential function reads:

$$u(t) = e^{j\omega t} \Rightarrow y(t) = \int_0^{\infty} h(\tau) e^{j\omega(t-\tau)} d\tau \quad (3.17)$$

Using the fact that the integration is a linear operation we get:

$$y(t) = e^{j\omega t} \int_0^{\infty} h(\tau) e^{-j\omega\tau} d\tau \quad (3.18)$$

We recognize in the term $\int_0^{\infty} h(\tau) e^{-j\omega\tau} d\tau$ the Laplace transform of the impulse response $h(t)$ where the Laplace variable s is specialized to be $j\omega$:

$$\int_0^{\infty} h(\tau) e^{-j\omega\tau} d\tau = \int_0^{\infty} h(\tau) e^{-s\tau} d\tau \Big|_{s=j\omega} = F(s) \Big|_{s=j\omega} \quad (3.19)$$

Note that $F(s) \Big|_{s=j\omega} = F(j\omega)$ is called the frequency response of the system.

As a consequence the frequency response of a linear system to the complex exponential function reads:

$$u(t) = e^{j\omega t} \Rightarrow y(t) = e^{j\omega t} F(s) \Big|_{s=j\omega} = F(j\omega) e^{j\omega t} \quad (3.20)$$

The preceding relation shows that frequency response of a linear system is closely linked to its transfer function. The frequency of the output signal is the same than the frequency of the input signal; only the amplitude and the phase of the input signal are changed by the linear system. In addition the phase of the output signal is independent of the amplitude of the input signal. Denoting by $\|F(j\omega)\|$ the norm of $F(j\omega)$ and by $\Phi(\omega)$ the phase of $F(j\omega)$ the preceding relation reads:

$$\begin{cases} u(t) = e^{j\omega t} \\ F(j\omega) = \|F(j\omega)\| e^{j\Phi(\omega)} \end{cases} \Rightarrow y(t) = \|F(j\omega)\| e^{j\omega t} e^{j\Phi(\omega)} = \|F(j\omega)\| e^{j(\omega t + \Phi(\omega))} \quad (3.21)$$

Where Re stands for real part and Im stands for imaginary part:

$$\begin{cases} \|F(j\omega)\| = \sqrt{(ReF(j\omega))^2 + (ImF(j\omega))^2} \\ \Phi(\omega) = \arg(F(j\omega)) = \arctan\left(\frac{ImF(j\omega)}{ReF(j\omega)}\right) \end{cases} \quad (3.22)$$

As far as numerator and denominator of rational transfer function $F(j\omega)$ have real coefficients we have:

$$\begin{cases} Re(F(j\omega)) = Re(F(-j\omega)) \\ Im(F(j\omega)) = -Im(F(-j\omega)) \end{cases} \Rightarrow \begin{cases} \|F(j\omega)\| = \|F(-j\omega)\| \\ \arg(F(j\omega)) = -\arg(F(-j\omega)) \end{cases} \quad (3.23)$$

Thus $\|F(j\omega)\|$ is an even function of ω whereas $\arg(F(j\omega))$ is an odd function of ω . Consequently $\|F(j\omega)\|$ and $\arg(F(j\omega))$ are only studied for positive values of ω , which consistently corresponds to physically achievable frequencies:

$$\omega \geq 0 \quad (3.24)$$

It is worth noticing that when the input signal $u(t)$ is not the complex exponential function but a real sinusoidal signal then the output signal $y(t)$ is obtained thanks to the real (or imaginary) part of the complex output signal:

$$\begin{aligned} u(t) &= \cos(\omega t) = \operatorname{Re}(e^{j\omega t}) \\ \Rightarrow y(t) &= \operatorname{Re}(\|F(j\omega)\| e^{j(\omega t + \Phi(\omega))}) = \|F(j\omega)\| \cos(\omega t + \Phi(\omega)) \end{aligned} \quad (3.25)$$

And similarly:

$$\begin{aligned} u(t) &= \sin(\omega t) = \operatorname{Im}(e^{j\omega t}) \\ \Rightarrow y(t) &= \operatorname{Im}(\|F(j\omega)\| e^{j(\omega t + \Phi(\omega))}) = \|F(j\omega)\| \sin(\omega t + \Phi(\omega)) \end{aligned} \quad (3.26)$$

3.5.2 Bode plot

Bode plot consists in drawing the log magnitude $20 \log_{10} \|F(j\omega)\|$ (this is the Bode magnitude plot) and the phase $\arg(F(j\omega))$ (this is the Bode phase plot) of the frequency response $F(j\omega)$ over the frequency range from $\omega = 0$ to infinity. The plot is usually presented on a semi-log graph where the ω axis is plotted on a logarithmic scale.

Two units are used to express frequency ratios: there are octave and decade. An octave is a frequency band from ω to 2ω whereas a decade is a frequency band from ω to 10ω .

As far as the magnitude of the frequency response is concerned the unit used for the logarithm of the magnitude is the *decibel* (dB). The dB value of 2 is 6 dB and the dB value of $\frac{1}{2}$ is -6 dB. In addition as a number increases by a factor 10 the decibel values increases by 20 dB. More generally as a number increases by a factor 10^n the decibel values increases by $20n$ dB.

The Bode plot of the frequency response is easily obtained when considering the factorized form of the transfer function. Let's us suppose that the transfer function $F(s)$ can be factorized into the following first and second order terms where k is a real (either positive or negative) and α, a_i, b_j are integers either positive or negative:

$$F(s) = \frac{k}{s^\alpha} \prod_i (1 + \tau_i s)^{a_i} \prod_j \left(1 + \frac{2m_j}{\omega_j} s + \frac{s^2}{\omega_j^2} \right)^{b_j} \quad (3.27)$$

Integer α is the number of integrations in the transfer function $F(s)$. First order terms are of the form $1 + \tau_i s$, $\tau_i > 0$ and are related to the real poles ($a_i < 0$) and zeros ($a_i > 0$) of the transfer function whereas second order terms are of the form $1 + \frac{2m_j}{\omega_j} s + \frac{s^2}{\omega_j^2}$, $m_j, \omega_j > 0$ are related to the complex poles ($b_j < 0$) and zeros ($b_j > 0$) of the transfer function.

Replacing the Laplace variable s by $j\omega$ and rewriting the transfer function in polar form yields:

$$F(j\omega) = \frac{\|k\|}{\|\omega\|^\alpha} \prod_i \|1 + \tau_i s\|^{a_i} \prod_j \left\| 1 + \frac{2m_j}{\omega_j} s + \frac{s^2}{\omega_j^2} \right\|^{b_j} \times e^{j\Phi(\omega)} \quad (3.28)$$

Taking the log magnitude of the frequency response reads:

$$\begin{aligned}
20 \log_{10} \|F(j\omega)\| &= 20 \log_{10} \|k\| - 20\alpha \log_{10} (\omega) \\
&+ \sum_i 20a_i \log_{10} \|1 + j\omega\tau_i\| \\
&+ \sum_j 20b_j \log_{10} \left\| 1 - \left(\frac{\omega}{\omega_j}\right)^2 + 2m_j \frac{\omega}{\omega_j} j \right\| \quad (3.29)
\end{aligned}$$

Thus when considering the factorized form of the transfer function the log magnitude of the frequency response is readily obtained by summing the log magnitude of the individual factors.

Similarly the phase of the frequency response is obtained by summing the phase of the individual factors of the factorized form of the transfer function:

$$\begin{aligned}
\Phi(\omega) &= \arg(k) - \arg((j\omega)^\alpha) + \sum_i \arg(1 + j\omega\tau_i)^{a_i} \\
&+ \sum_j \arg\left(1 - \left(\frac{\omega}{\omega_j}\right)^2 + 2m_j \frac{\omega}{\omega_j} j\right)^{b_j} \quad (3.30)
\end{aligned}$$

That is:

$$\Phi(\omega) = \arg(k) - \alpha \frac{\pi}{2} + \sum_i a_i \arctan(\omega\tau_i) + \sum_j b_j \arctan\left(\frac{2m_j \omega_j \omega}{\omega_j^2 - \omega^2}\right) \quad (3.31)$$

Notice that $\arg(k) = 0$ if $k > 0$ and $\arg(k) = \pi$ if $k < 0$.

Asymptotic Bode plot is obtained when drawing the shape of the individual factors of $F(j\omega)$ for the limit values of ω that are $\omega \rightarrow 0$ and $\omega \rightarrow \infty$.

For the log magnitude we get:

$$\lim_{\omega \rightarrow 0} 20 \log_{10} \|F(j\omega)\| = 20 \log_{10} \|k\| - 20\alpha \log_{10} (\omega) \quad (3.32)$$

And:

$$\begin{aligned}
\lim_{\omega \rightarrow \infty} 20 \log_{10} \|F(j\omega)\| &= 20 \log_{10} \|k\| - 20\alpha \log_{10} (\omega) \\
&+ \sum_i 20a_i \log_{10} (\omega\tau_i) + \sum_j 40b_j \log_{10} \left(\frac{\omega}{\omega_j}\right) \quad (3.33)
\end{aligned}$$

Table 3.1 summarizes the main characteristics of the asymptotic Bode plot of first order and second order individual factors where ω_c stands for the corner frequency and n is a positive or negative integer:

	slope in dB/decade		phase in deg	
	$\omega < \omega_c$	$\omega > \omega_c$	$\omega < \omega_c$	$\omega > \omega_c$
s^α	20α		90α	
$(1 + \tau_i s)^{a_i}, \omega_c = 1/\tau_i$	0	$20a_i$	0	$90a_i \operatorname{sgn}(a_i)$
$\left(\frac{s^2 + 2m_j\omega_j s + \omega_j^2}{\omega_j^2} \right)^{b_j}, \omega_c = \omega_j$	0	$40b_j$	0	$180b_j \operatorname{sgn}(b_j)$

Table 3.1: Main characteristics of asymptotic Bode plot

3.6 Model reduction

Model reduction consists in computing an approximate model of lower order of a system in order to facilitate the control design. Several methods exist. We will present in that section only the Padé (or dominant pole) approximation which is one of the simplest method.

We will assume in the following that the transfer function $F(s)$ of the model to be reduced is strictly proper and that all the roots of its polynomial characteristics have strictly negative real parts (i.e. all the poles of $F(s)$ are situated in the left half part of the complex plane):

$$\begin{cases} F(s) = \frac{b_0 + b_1 s + \dots + b_{n-1} s^{n-1}}{a_0 + a_1 s + \dots + a_{n-1} s^{n-1} + s^n} = \frac{N(s)}{D(s)} \\ D(\lambda_i) = 0 \Rightarrow \operatorname{Re}(\lambda_i) < 0 \end{cases} \quad (3.34)$$

Notice that without loss of generality coefficient a_n has been set to 1.

The Padé approximation of order (q, r) of $F(s)$ is the transfer function $\tilde{F}_{q,r}(s)$ whose Taylor series expansion around $s = 0$ has the same first $q + r$ coefficients than those of $F(s)$.

$$\tilde{F}_{q,r}(s) = \frac{\tilde{b}_0 + \tilde{b}_1 s + \dots + \tilde{b}_q s^q}{\tilde{a}_0 + \tilde{a}_1 s + \dots + \tilde{a}_{r-1} s^{r-1} + s^r} \quad (3.35)$$

From a practical point of view Taylor series expansion around $s = 0$ can be obtained through the quotient of the polynomial long division, that is the division according to the increasing power of s . It is worth noticing that contrary to the Euclidean division where the polynomials are written according to the decreasing powers of s , in the polynomial long division the polynomials are written according to the *increasing* powers of s .

As far as Taylor series expansion is obtained around $s = 0$ the Padé approximation keeps the low frequency shape of the frequency response of $F(s)$.

In the following we will set $q = 0$ and we will denote $\tilde{F}_{0,r}(s) \equiv \tilde{F}(s)$:

$$\tilde{F}_{0,r}(s) \equiv \tilde{F}(s) = \frac{\tilde{b}_0}{\tilde{a}_0 + \tilde{a}_1 s + \dots + \tilde{a}_{r-1} s^{r-1} + s^r} = \frac{\tilde{b}_0}{\tilde{D}(s)} \quad (3.36)$$

The denominator $\tilde{D}(s)$ of the reduced model $\tilde{F}(s)$ is built thanks to the r poles of $F(s)$ whose real part are the closest to the imaginary axis. Those poles

are called the *dominant* poles of $F(s)$. If the values obtained when dividing the real part of the others poles of $F(s)$ by the real part of the dominant poles of $F(s)$ are greater than 3 (or 2) then the low frequency model obtained through the Padé approximation is quite representative of the *long term response* of the system.

Once $\tilde{D}(s)$ is known, the coefficient \tilde{b}_0 could be obtained thanks to the identification of the Taylor development of $\tilde{F}(s)$ and $F(s)$ around $s = 0$. In order to avoid the computation of the first terms of the Taylor series expansion it is worth noticing that coefficient \tilde{b}_0 can also be obtained by identifying the static gain of the actual transfer function $F(s)$ and the static gain of the reduced model $\tilde{F}(s)$.

It is worth noticing that first and second order models are widely studied because they can be viewed as reduced order models of actual transfer functions. Indeed:

- When the dominant pole is real then $r = 1$: this leads a first order reduced model.
- When the dominant poles are complex conjugate then $r = 2$: this leads to a second order reduced model.

Example 3.1. *Let's consider the following transfer function:*

$$F(s) = \frac{15}{s^2 + 6s + 5} = \frac{N(s)}{D(s)} \quad (3.37)$$

The poles of $F(s)$ are the following (you may use function roots with Scilab):

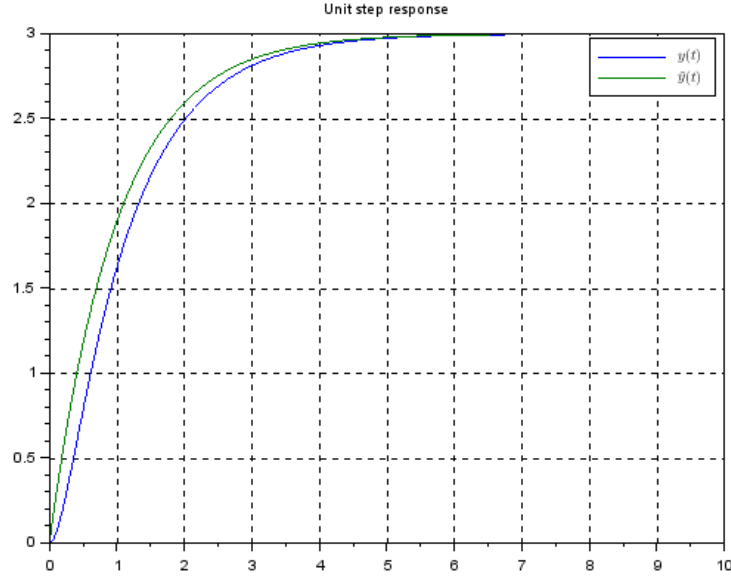
$$\begin{aligned} D(s) &= s^2 + 6s + 5 = (s + 1)(s + 5) = 0 \\ \Rightarrow \begin{cases} \lambda_1 = -1 \\ \lambda_2 = -5 \end{cases} \end{aligned} \quad (3.38)$$

All the poles of $F(s)$ have negative real part and $D(0) \neq 0$. The dominant poles of $F(s)$ is -1 and the values obtained when dividing the real part of the others poles of $F(s)$ by the real part of the dominant poles are greater than 3 (or 2). As a consequence the system is subject to Padé approximation. First we kept the dominant poles of $F(s)$ to build the denominator $\tilde{D}(s)$ of the approximated transfer function $\tilde{F}(s)$:

$$\tilde{F}(s) = \frac{\tilde{b}_0}{s + 1} \quad (3.39)$$

Then we compute the quotient of the polynomial long division (i.e. division according to the increasing power of s) of $\tilde{F}(s)$:

$$\begin{array}{r|l} \tilde{b}_0 & 1 + s \\ - (\tilde{b}_0 + \tilde{b}_0 s) & \tilde{b}_0 + \dots \\ \hline & -\tilde{b}_0 s \end{array} \quad (3.40)$$

Figure 3.2: Unit step responses of $F(s)$ and $\tilde{F}(s)$

We also compute the quotient of the polynomial long division (i.e. division according to the increasing power of s) of $F(s)$:

$$\begin{array}{r|l} 15 & 5 + 6s + s^2 \\ - (15 + 18s + 3s^2) & 3 + \dots \\ \hline -18s & \end{array} \quad (3.41)$$

The identification of the two quotients leads to the expression of coefficients \tilde{b}_0 and \tilde{b}_1 :

$$\tilde{b}_0 = 3 \quad (3.42)$$

Thus the Padé approximation of $F(s)$ reads:

$$\tilde{F}(s) = \frac{3}{s+1} \quad (3.43)$$

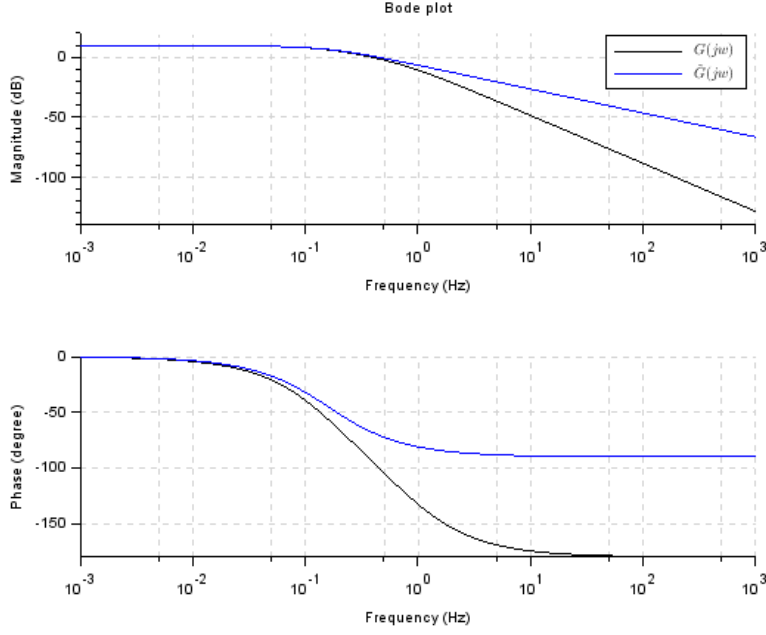
Figures 3.2 and 3.3 show the unit step responses as well as Bode plots of the model $F(s)$ and its Padé approximation $\tilde{F}(s)$; it is clear that $\tilde{F}(s)$ exhibits the same low frequency shape than $F(s)$. ■

Example 3.2. Let's consider the following transfer function:

$$F(s) = \frac{1}{(2s+1)(3s+1)(4s+1)} = \frac{1}{1+9s+26s^2+24s^3} \quad (3.44)$$

It is clear that the poles of $F(s)$ are the following:

$$\begin{cases} \lambda_1 = -\frac{1}{2} \\ \lambda_2 = -\frac{1}{3} \\ \lambda_3 = -\frac{1}{4} = -0.25 \end{cases} \quad (3.45)$$

Figure 3.3: Bode plots of $F(s)$ and $\tilde{F}(s)$

All the poles of $F(s)$ have negative real part and $D(0) \neq 0$. The dominant poles of $F(s)$ is -0.25 . Despite the fact that all the values obtained when dividing the real part of the others poles of $F(s)$ by the real part of the dominant poles are not greater than 3 (or 2) we will test Padé approximation. First we kept the dominant poles of $F(s)$ to build the denominator $\tilde{D}(s)$ of the approximated transfer function $\tilde{F}(s)$:

$$\tilde{F}(s) = \frac{\tilde{b}_0}{s + 0.25} \quad (3.46)$$

Then we compute the quotient of the polynomial long division (i.e. division according to the increasing power of s) of $\tilde{F}(s)$:

$$\frac{\tilde{b}_0}{-(\tilde{b}_0 + 4\tilde{b}_0s)} \left| \begin{array}{l} 0.25 + s \\ 4\tilde{b}_0 + \dots \\ -4\tilde{b}_0s \end{array} \right. \quad (3.47)$$

We also compute the quotient of the polynomial long division (i.e. division according to the increasing power of s) of $F(s)$:

$$\frac{1}{-(1 + 9s + \dots)} \left| \begin{array}{l} 1 + 9s + 26s^2 + 24s^3 \\ 1 + \dots \\ -9s + \dots \end{array} \right. \quad (3.48)$$

The identification of the two quotients leads to the expression of coefficients \tilde{b}_0 and \tilde{b}_1 :

$$4\tilde{b}_0 = 1 \Rightarrow \tilde{b}_0 = 0.25 \quad (3.49)$$

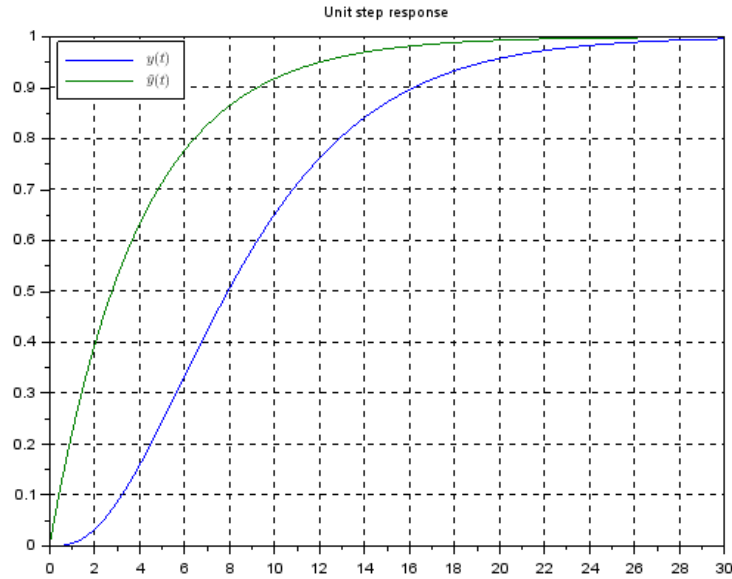


Figure 3.4: Unit step responses of $F(s)$ and $\tilde{F}(s)$

Thus the Padé approximation of $F(s)$ reads:

$$\tilde{F}(s) = \frac{0.25}{s + 0.25} = \frac{1}{4s + 1} \quad (3.50)$$

Figures 3.4 and 3.5 show the unit step responses as well as Bode plots of the model $F(s)$ and its Padé approximation $\tilde{F}(s)$. We can see in that example that the unit step response of the Padé approximation is not close from the unit step response of the full model; nevertheless the two responses move towards the same final values. Indeed both transfer functions have the same static gain: $F(0) = \tilde{F}(0)$. ■

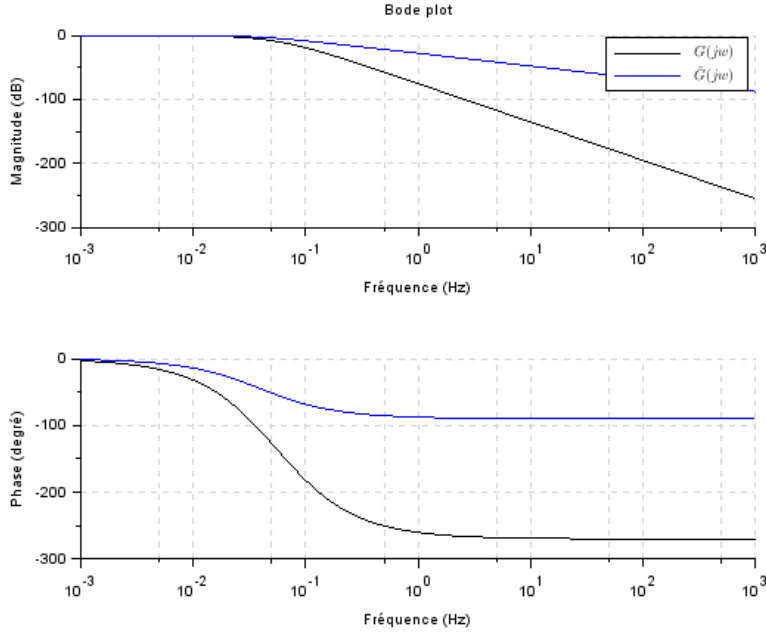
Example 3.3. Let's consider the following transfer function:

$$F(s) = \frac{156 + s + 2s^3 + s^4}{156 + 268s + 216s^2 + 84s^3 + 15s^4 + s^5} = \frac{N(s)}{D(s)} \quad (3.51)$$

The poles of $F(s)$ are the following (you may use function roots with Scilab):

$$\begin{aligned} D(s) &= 156 + 268s + 216s^2 + 84s^3 + 15s^4 + s^5 = 0 \\ \Rightarrow \begin{cases} \lambda_1 = -5 + j \\ \lambda_2 = -5 - j \\ \lambda_3 = -3 \\ \lambda_4 = -1 + j \\ \lambda_5 = -1 - j \end{cases} \end{aligned} \quad (3.52)$$

All the poles of $F(s)$ have negative real part and $D(0) \neq 0$. The dominant poles of $F(s)$ are $-1 \pm j$ and the values obtained when dividing the real part

Figure 3.5: Bode plots of $F(s)$ and $\tilde{F}(s)$

of the others poles of $F(s)$ by the real part of the dominant poles are greater than 3 (or 2). As a consequence the system is subject to Padé approximation. First we kept the dominant poles of $F(s)$ to build the denominator $\tilde{D}(s)$ of the approximated transfer function $\tilde{F}(s)$:

$$\tilde{F}(s) = \frac{\tilde{b}_0 + \tilde{b}_1 s}{(s + 1 - j)(s + 1 + j)} \quad (3.53)$$

Then we compute the quotient of the polynomial long division (i.e. division according to the increasing power of s) of $\tilde{F}(s)$:

$$\frac{\tilde{b}_0 + \tilde{b}_1 s}{(\tilde{b}_1 - \tilde{b}_0)s - \frac{\tilde{b}_0}{2} s^2} \left| \begin{array}{l} 2 + 2s + s^2 \\ \frac{\tilde{b}_0}{2} + \frac{\tilde{b}_1 - \tilde{b}_0}{2} s + \dots \end{array} \right. \quad (3.54)$$

We also compute the quotient of the polynomial long division (i.e. division according to the increasing power of s) of $F(s)$:

$$\frac{156 + s + 2s^3 + s^4}{-267s - 216s^2 - 82s^3 - 14s^4 - s^5} \left| \begin{array}{l} 156 + 268s + 216s^2 + 84s^3 + 15s^4 + s^5 \\ 1 - \frac{267}{156}s + \dots \end{array} \right. \quad (3.55)$$

The identification of the two quotients leads to the expression of coefficients \tilde{b}_0 and \tilde{b}_1 :

$$\begin{cases} \frac{\tilde{b}_0}{2} = 1 \Rightarrow \tilde{b}_0 = 2 \\ \frac{\tilde{b}_1 - \tilde{b}_0}{2} = -\frac{267}{156} \Rightarrow \tilde{b}_1 = 2 \left(1 - \frac{267}{156}\right) \approx -2 \times 0.711 \end{cases} \quad (3.56)$$

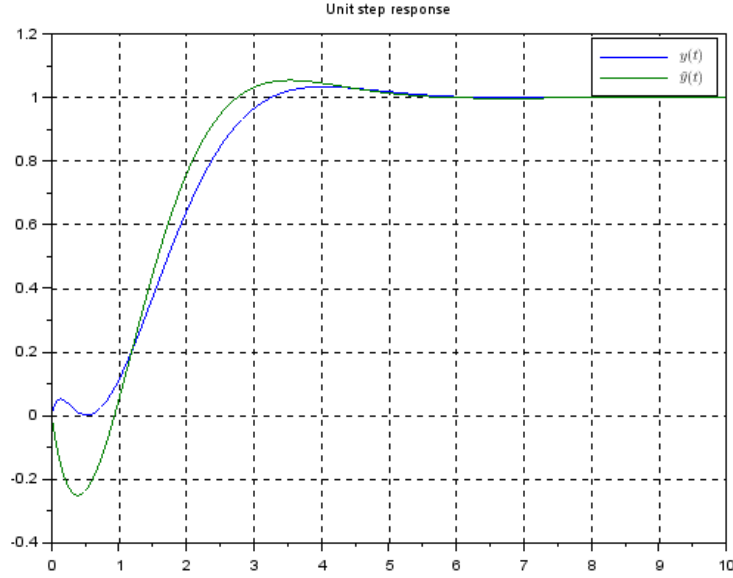


Figure 3.6: Unit step responses of $F(s)$ and $\tilde{F}(s)$

Thus the Padé approximation of $F(s)$ reads:

$$\tilde{F}(s) = 2 \frac{1 - 0.711s}{2 + 2s + s^2} \quad (3.57)$$

Figures 3.6 and 3.7 show the unit step responses as well as Bode plots of the model $F(s)$ and its Padé approximation $\tilde{F}(s)$; it is clear that $\tilde{F}(s)$ exhibits the same low frequency shape than $F(s)$.

Nevertheless the Padé approximation of $F(s)$ is a non-minimum phase transfer function, meaning that some of its zeros have positive real part. As a consequence the response starts in the opposite direction of the final value. The derivative at the starting time can be computed thanks to the initial value theorem:

$$\lim_{t \rightarrow 0} \dot{y}(t) = \lim_{s \rightarrow \infty} s \mathcal{L}[\dot{y}(t)] = \lim_{s \rightarrow \infty} s^2 Y(s) = \lim_{s \rightarrow \infty} s^2 F(s) U(s) \quad (3.58)$$

As far as $u(t)$ is the unit step function whose Laplace transform is $1/s$ we finally obtain:

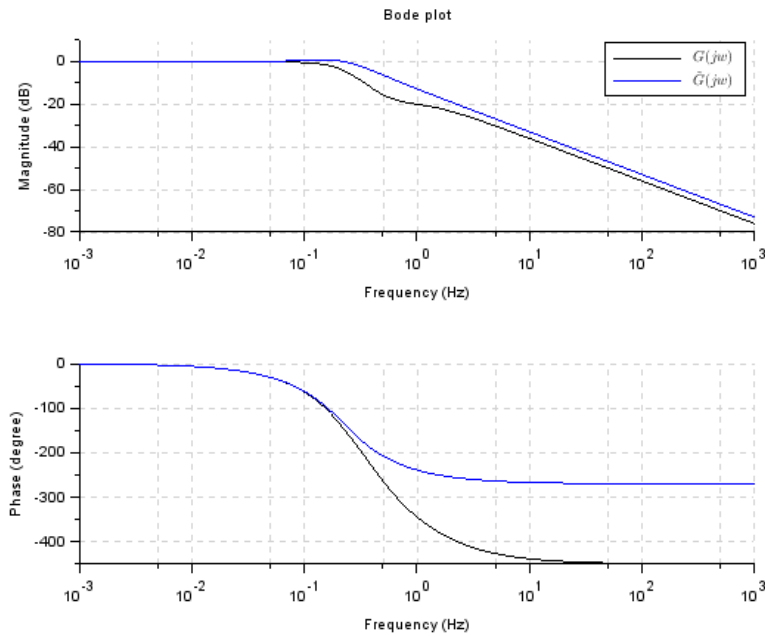
$$U(s) = \frac{1}{s} \Rightarrow \lim_{t \rightarrow 0} \dot{y}(t) = \lim_{s \rightarrow \infty} s F(s) \quad (3.59)$$

■

3.7 First order model

3.7.1 Time response

An first order system is described by a linear differential equation of the first order with constant coefficients. Denoting by τ ($\tau > 0$) the time constant and

Figure 3.7: Bode plots of $F(s)$ and $\tilde{F}(s)$

by K the static gain, first order model reads:

$$\tau \frac{dy}{dt} + y(t) = Ku(t) \quad (3.60)$$

We will assume in the following that:

$$y(0) = 0 \quad (3.61)$$

The transfer function of such system reads:

$$F(s) = \frac{K}{1 + \tau s} \quad (3.62)$$

The unit step response of a first order systems will be computed thanks to the Laplace transform. Since the Laplace transform of the unit step function is $1/s$ and substituting $U(s) = 1/s$ we obtain:

$$Y(s) = F(s)U(s) = \frac{K}{s(1 + \tau s)} \quad (3.63)$$

Expanding $Y(s)$ into partial fractions gives:

$$Y(s) = \frac{K}{s} - \frac{K\tau}{1 + \tau s} = \frac{K}{s} - \frac{K}{1/\tau + s} \quad (3.64)$$

Taking the inverse Laplace transform of the preceding equation yields:

$$y(t) = K - Ke^{-\frac{t}{\tau}} \quad \forall t \geq 0 \quad (3.65)$$

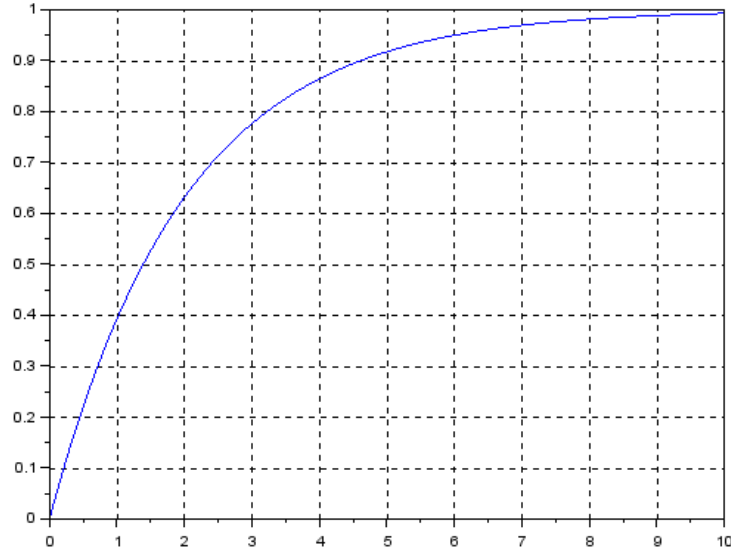


Figure 3.8: First order system unit step response where $K = 1$ and $\tau = 2$

This equation states that the output is initially zero (that is $y(0)$) and that it finally moves towards K when $t \rightarrow \infty$ as soon as $\tau > 0$. Note that the smaller the time constant τ is the faster the system response is.

We can check that $y(t) = h(t) * u(t) = \int_0^\infty h(\tau)u(t - \tau)d\tau$. Indeed:

$$h(t) = \mathcal{L}^{-1}[F(s)] = \mathcal{L}^{-1}\left[\frac{K}{\tau} \frac{1}{\left(\frac{1}{\tau} + s\right)}\right] = \frac{K}{\tau} e^{-\frac{t}{\tau}} \quad (3.66)$$

And:

$$\begin{aligned} y(t) &= h(t) * u(t) = \int_0^\infty h(x)u(t - x)dx = \int_0^t h(x)dx \\ \Leftrightarrow y(t) &= \frac{K}{\tau} \int_0^t e^{-\frac{x}{\tau}} dx = -K e^{-\frac{x}{\tau}} \Big|_0^t = K - K e^{-\frac{t}{\tau}} \quad \forall t \geq 0 \end{aligned} \quad (3.67)$$

The step response of a first order model has no overshoot. In addition the slope of the tangent at $t = 0$ is $\frac{K}{\tau}$ since:

$$\left. \frac{dy}{dt} \right|_{t=0} = \left. \frac{K}{\tau} e^{-\frac{t}{\tau}} \right|_{t=0} = \frac{K}{\tau} \quad (3.68)$$

The unit step response of a first order system is sketched in Figure 3.8.

For a first order system the 5% settling time is:

$$t_s = 3\tau \quad (3.69)$$

Indeed:

$$0.95K = K \left(1 - e^{-\frac{t_s}{\tau}}\right) \Rightarrow 0.05 = e^{-\frac{t_s}{\tau}} \Rightarrow t_s = \tau \ln\left(\frac{1}{0.05}\right) \approx 3\tau \quad (3.70)$$

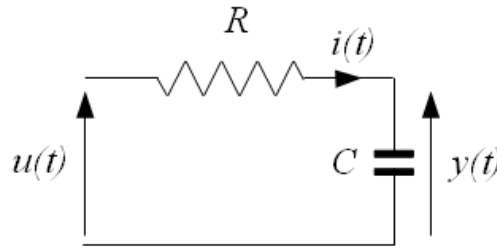


Figure 3.9: RC Filter

Notice that time responses of linear time invariant systems can be computed thanks to functions *syslin* and *csim* (and the use of symbol *%s* to define polynomials) with Scilab.

Example 3.4. *Let's consider Figure 3.9 where an RC filter is depicted: $u(t)$ is the voltage applied to the circuit, $y(t)$ is the voltage across the capacitor and $i(t)$ the current through the capacitor.*

The relation between the input $u(t)$ and the output $y(t)$ is the following:

$$\begin{cases} u(t) = Ri(t) + y(t) \\ i(t) = C\dot{y}(t) \end{cases} \quad (3.71)$$

$$\Rightarrow \dot{y}(t) - \frac{i(t)}{C} = 0$$

$$\Leftrightarrow RC\dot{y}(t) + y(t) = u(t)$$

Comparing the differential equation which drives the dynamic of RC filter with equation (3.60) we conclude that this system is a model for a first order system with time constant $\tau = RC$ and static gain $K = 1$.

■

3.7.2 Frequency response

Let's consider the following first order model where the static gain K has been set to 1:

$$F(s) = \frac{1}{1 + \tau s} \Rightarrow F(j\omega) = \frac{1}{1 + j\omega\tau} \quad (3.72)$$

The log magnitude of the frequency response reads:

$$20 \log_{10} \|F(j\omega)\| = 20 \log_{10} \left(\frac{1}{\sqrt{1 + (\omega\tau)^2}} \right) = -10 \log_{10} (1 + (\omega\tau)^2) \quad (3.73)$$

Thus when $\omega\tau \rightarrow 0$ the log magnitude of $F(j\omega)$ tends towards 0 whereas when $\omega\tau \rightarrow \infty$ the log magnitude of $F(j\omega)$ is approximated by $-10 \log_{10} ((\omega\tau)^2)$ that is $-20 \log_{10} (\omega\tau)$. The two asymptotes cross each other

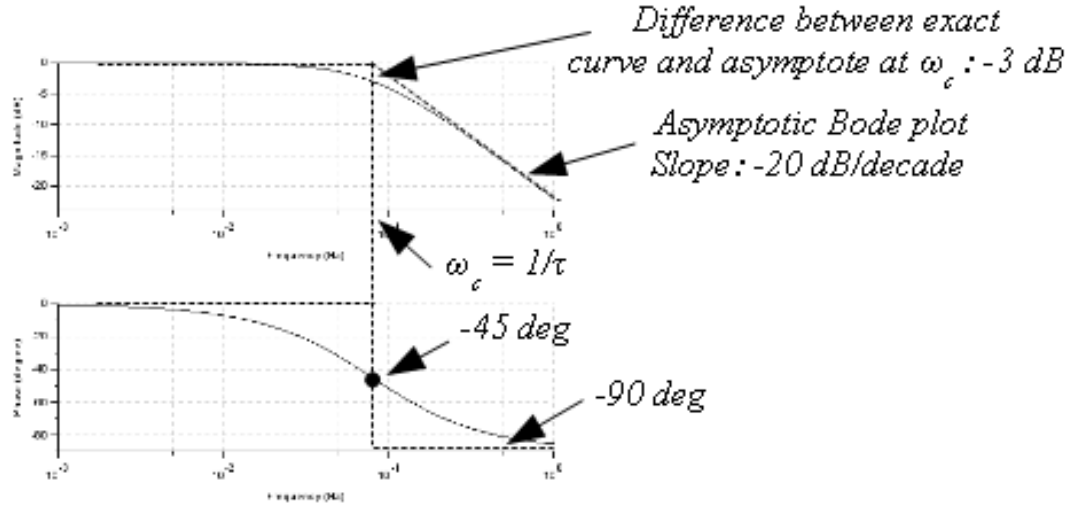


Figure 3.10: Bode plot and asymptotic Bode plot of a first order model where $K = 1$ and $\tau = 2$

when the frequency ω is equal to the corner frequency ω_c . Corner frequency ω_c is defined by the following equation:

$$0 = -20 \log_{10}(\omega_c \tau) \Leftrightarrow \omega_c = \frac{1}{\tau} \quad (3.74)$$

The phase of the frequency response reads:

$$\Phi(\omega) = \arg(F(j\omega)) = -\arctan(\omega\tau) \quad (3.75)$$

Thus when $\omega\tau \rightarrow 0$ the phase of $F(j\omega)$ tends towards 0 whereas when $\omega\tau \rightarrow \infty$ the phase of $F(j\omega)$ is approximated by -90 deg . At the corner frequency ω_c the phase is equal to $-\arctan(1) = -45 \text{ deg}$.

Figure 3.10 presents the Bode plot as well as the asymptotic Bode plot for a first order model where $K = 1$ and $\tau = 2$.

3.8 Second order model

3.8.1 Differential equation and unit step response

A second order model is described by a linear differential equation of the second order with constant coefficients. Denoting by ω_0 ($\omega_0 > 0$) the undamped natural frequency of the model, by m ($m > 0$) the damping ratio and by K the static gain, second order model reads:

$$\frac{1}{\omega_0^2} \frac{d^2 y}{dt^2} + \frac{2m}{\omega_0} \frac{dy}{dt} + y(t) = Ku(t) \quad (3.76)$$

The term $m\omega_0$ is referred as the damping factor.

We will assume in the following that:

$$y(0) = \dot{y}(0) = 0 \quad (3.77)$$

The transfer function of such system reads:

$$F(s) = K \frac{\omega_0^2}{s^2 + 2m\omega_0 s + \omega_0^2} \quad (3.78)$$

The unit step response of a second order model will be computed thanks to the Laplace transform. Since the Laplace transform of the unit step function is $1/s$ and substituting $U(s) = 1/s$ we obtain:

$$Y(s) = F(s)U(s) = K \frac{\omega_0^2}{s(s^2 + 2m\omega_0 s + \omega_0^2)} \quad (3.79)$$

Denoting by λ_1 and λ_2 the two roots of the characteristic polynomial $s^2 + 2m\omega_0 s + \omega_0^2$ and assuming that those two values are distinct the preceding relation reads:

$$Y(s) = F(s)U(s) = K \frac{\omega_0^2}{s(s - \lambda_1)(s - \lambda_2)} \quad (3.80)$$

Expanding $Y(s)$ into partial fractions and using the fact that $\lambda_1 \lambda_2 = \omega_0^2$ (we also have $\lambda_1 + \lambda_2 = -2m\omega_0$ when identifying $s^2 + 2m\omega_0 s + \omega_0^2$ with $(s - \lambda_1)(s - \lambda_2)$) gives

$$\begin{aligned} Y(s) &= K \frac{\lambda_1 \lambda_2}{s(s - \lambda_1)(s - \lambda_2)} \\ &= K \left(\frac{1}{s} + \frac{1}{\lambda_1 - \lambda_2} \left(\frac{\lambda_2}{s - \lambda_1} - \frac{\lambda_1}{s - \lambda_2} \right) \right) \end{aligned} \quad (3.81)$$

Taking the inverse Laplace transform of the preceding equation yields:

$$y(t) = K \left(1 + \frac{1}{\lambda_1 - \lambda_2} \left(\lambda_2 e^{\lambda_1 t} - \lambda_1 e^{\lambda_2 t} \right) \right) \quad \forall t \geq 0 \quad (3.82)$$

This equation states that the output is initially zero (that is $y(0)$) and that it finally moves towards K when $t \rightarrow \infty$ as soon as $Re(\lambda_1) < 0$ and $Re(\lambda_2) < 0$. In addition as far as we have imposed $\dot{y}(0) = 0$ we can check that the slope of the tangent at $t = 0$ is 0 since:

$$\left. \frac{dy}{dt} \right|_{t=0} = K \left(\frac{1}{\lambda_1 - \lambda_2} \left(\lambda_2 \lambda_1 e^{\lambda_1 t} - \lambda_1 \lambda_2 e^{\lambda_2 t} \right) \right) \Big|_{t=0} = 0 \quad (3.83)$$

The shape of $y(t)$ depends on the roots of the characteristic polynomial $s^2 + 2m\omega_0 s + \omega_0^2$:

- The first case deals with the case where $m > 1$
- The second case deals with the case where $m < 1$
- The second case deals with the case where $m = 1$

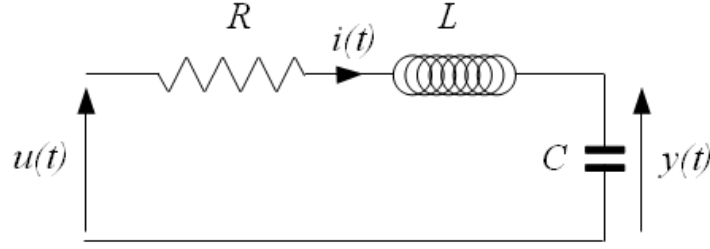


Figure 3.11: RLC Filter

Example 3.5. Let's consider Figure 3.11 where an RLC filter is depicted: $u(t)$ is the voltage applied to the circuit, $y(t)$ is the voltage across the capacitor and $i(t)$ the current through the capacitor.

The relation between the input $u(t)$ and the output $y(t)$ is the following:

$$\begin{cases} u(t) = Ri(t) + L\frac{di}{dt} + y(t) \\ i(t) = C\dot{y}(t) \end{cases} \quad (3.84)$$

$$\Rightarrow u(t) = RC\dot{y}(t) + LC\ddot{y}(t) + y(t)$$

$$\Leftrightarrow LC\ddot{y}(t) + RC\dot{y}(t) + y(t) = u(t)$$

Comparing the differential equation which drives the dynamic of RLC filter with equation (3.76) we conclude that this system is a model for a second order system with undamped natural frequency $\omega_0 = \frac{1}{\sqrt{LC}}$ and damping ratio m defined by $\frac{2m}{\omega_0} = RC$ that is $m = \frac{R}{2}\sqrt{\frac{C}{L}}$. The damping factor is equal to $m\omega_0 = \frac{R}{2L}$. ■

3.8.2 Overdamped model

Overdamped second order model corresponds to the case where $m > 1$. In that situation the roots λ_1 and λ_2 are real and distinct:

$$m > 1 \Rightarrow \begin{cases} \lambda_1 = -m\omega_0 + \omega_0\sqrt{m^2 - 1} \in \mathbb{R} \\ \lambda_2 = -m\omega_0 - \omega_0\sqrt{m^2 - 1} \in \mathbb{R} \end{cases} \quad (3.85)$$

The unit step response of a second order system where $m = 1.2$, $\omega_0 = 1$ and $K = 1$ is sketched in Figure 3.12. It is worth noticing that there is no overshoot on the step response.

Time t_i at which the sign of the second derivative of $y(t)$ changes is given by:

$$\ddot{y}(t) = 0 \Rightarrow \lambda_1 e^{\lambda_1 t_i} = \lambda_2 e^{\lambda_2 t_i} \Rightarrow t_i = \frac{1}{\lambda_1 - \lambda_2} \ln\left(\frac{\lambda_2}{\lambda_1}\right) \quad (3.86)$$

After time t_i the step response of a second order model where $m > 1$ is very close to the step response of a first order model. Indeed exponential $e^{\lambda_1 t}$ involves much greater values than the other exponential.

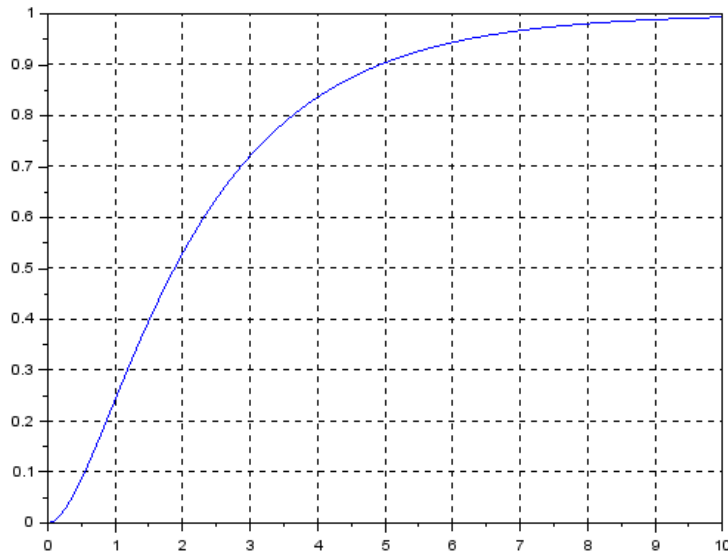


Figure 3.12: Second order system unit step response where $m = 1.2$, $\omega_0 = 1$ and $K = 1$

In addition the case where $m > 1$ is not representative of a second order system as far as this kind of system is equivalent to two cascaded first order systems.

3.8.3 Underdamped model

Underdamped second order model corresponds to the case where $m < 1$. In that situation the roots λ_1 and λ_2 are complex and conjugate:

$$m < 1 \Rightarrow \begin{cases} \lambda_1 = -m\omega_0 + j\omega_0\sqrt{1-m^2} \in \mathbb{C} \\ \lambda_2 = -m\omega_0 - j\omega_0\sqrt{1-m^2} \in \mathbb{C} \end{cases} \quad (3.87)$$

As soon as λ_1 and λ_2 are complex and conjugate we can write them using the Euler's formula :

$$\begin{aligned} \lambda_{1,2} &= \omega_0 \left(-m \pm j\sqrt{1-m^2} \right) = \omega_0 e^{\pm j\Phi} \\ \text{where } \begin{cases} \cos(\Phi) &= -m \\ \sin(\Phi) &= \sqrt{1-m^2} \end{cases} \end{aligned} \quad (3.88)$$

Denoting $a = \text{Re}(\lambda_1) = -\omega_0 m$ the real part of λ_1 and $b = \text{Im}(\lambda_1) = \omega_0 \sqrt{1-m^2}$ the imaginary part of λ_1 the unit step response (3.82) can be rewritten as follows:

$$\begin{aligned} y(t) &= K \left(1 + \frac{1}{2j\sqrt{1-m^2}} (e^{-j\Phi} e^{at+jbt} - e^{j\Phi} e^{at-jbt}) \right) \\ &= K \left(1 + \frac{e^{at}}{2j\sqrt{1-m^2}} (e^{j(bt-\Phi)} - e^{-j(bt-\Phi)}) \right) \\ &= K \left(1 + \frac{e^{at}}{\sqrt{1-m^2}} \sin(bt - \Phi) \right) \end{aligned} \quad (3.89)$$

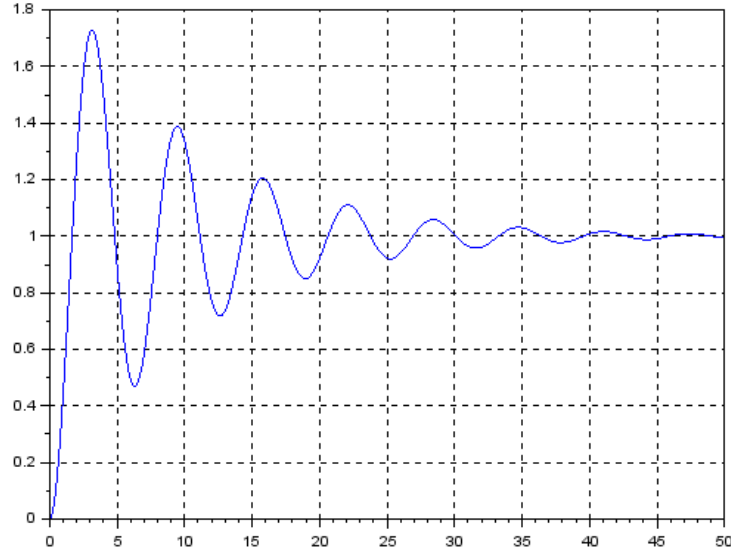


Figure 3.13: Second order system unit step response where $m = 0.1$, $\omega_0 = 1$ and $K = 1$

Using the expression $a = -\omega_0 m$ and $b = \omega_0 \sqrt{1 - m^2}$ we finally get:

$$y(t) = K \left(1 + \frac{e^{-m\omega_0 t}}{\sqrt{1 - m^2}} \sin \left(\omega_0 t \sqrt{1 - m^2} - \Phi \right) \right) \quad \forall t \geq 0 \quad (3.90)$$

The unit step response of a second order system where $m = 0.1$, $\omega_0 = 1$, and $K = 1$ is sketched in Figure 3.13.

Unlike the case where $m > 1$ the step response when $m < 1$ is oscillating and overshoots appear. The envelope of the sinusoid function is the an exponential function involving the real part of the roots of the characteristic polynomial whereas the frequency of the sinusoid is given by the positive imaginary part of the roots of the characteristic polynomial:

$$\begin{cases} \operatorname{Re}(\lambda_1) = -\omega_0 m & \text{gives the exponential rate of decay/growth} \\ \operatorname{Im}(\lambda_1) = \omega_0 \sqrt{1 - m^2} & \text{gives the oscillation frequency} \end{cases} \quad (3.91)$$

The oscillation frequency $\omega_0 \sqrt{1 - m^2}$ is also called the damped natural frequency.

Local maximums are obtained when the first derivative of the step response is null. This happen at time t_k , where k is an integer, defined by:

$$t_k = \frac{k\pi}{\omega_n \sqrt{1 - m^2}} \quad (3.92)$$

When $k = 1$ the first relative overshoot D with respect to the final value is given by:

$$D = \frac{y(t_1) - K}{K} = e^{-\frac{\pi m}{\sqrt{1 - m^2}}} \quad (3.93)$$

Consequently relative overshoot decreases when the damping ratio m increases but is independent of the undamped natural frequency ω_0 . Furthermore the first relative overshoot will be lower than 10% as soon as $m > 0.6$:

$$D < 0.1 \Rightarrow m > 0.6 \quad (3.94)$$

Conversely we can find the damping ratio m to obtain a specific relative overshoot D from the above:

$$D = e^{-\frac{\pi m}{\sqrt{1-m^2}}} \Leftrightarrow m = \frac{-\ln(D)}{\sqrt{\pi^2 + \ln^2(D)}} \quad (3.95)$$

Finally there is no simple expression which gives the settling time of a second order system. Indeed the settling time is not a monotonic function of m : when $m = 0.4$ the settling time is $t_s = \frac{7.5}{\omega_0}$ whereas when $m = 0.7$ the settling time is $t_s = \frac{2.85}{\omega_0}$. Nevertheless when the damping ratio is lower than 0.9 the settling time can be approximated through the following relation:

$$t_s \approx \frac{4}{m\omega_0} \quad (3.96)$$

The preceding relation indicated that the settling time is closely linked to the damping factor $m\omega_0$, that is to the opposite of the real part of the roots of the characteristic polynomial: the faster the system is the higher the damping factor $m\omega_0$ is.

3.8.4 Critically damped model

Critically damped second order model corresponds to the case where the damping ratio m is equal to 1. In that case the roots λ_1 and λ_2 are real and equal:

$$m = 1 \Rightarrow \lambda_1 = \lambda_2 = -\omega_0 \in \mathbb{R} \quad (3.97)$$

In that situation the roots of the characteristic polynomial are no more distinct. Thus the Laplace transform of the output signal $y(t)$ reads:

$$Y(s) = F(s)U(s) = K \frac{\omega_0^2}{s(s + \omega_0)^2} \quad (3.98)$$

Expanding $Y(s)$ into partial fractions gives

$$Y(s) = K \left(\frac{1}{s} - \frac{1}{s + \omega_0} - \frac{\omega_0}{(s + \omega_0)^2} \right) \quad (3.99)$$

Taking the inverse Laplace transform of the preceding equation yields:

$$y(t) = K (1 - (1 + \omega_0 t) e^{-\omega_0 t}) \quad \forall t \geq 0 \quad (3.100)$$

The step response present no overshoot in that case.

3.8.5 Frequency response

Let's consider the following second order model where the static gain K has been set to 1:

$$\begin{aligned} F(s) &= \frac{\omega_0^2}{s^2 + 2m\omega_0 s + \omega_0^2} \\ \Rightarrow F(j\omega) &= \frac{\omega_0^2}{\omega_0^2 - \omega^2 + 2jm\omega_0\omega} = \frac{1}{1 - \left(\frac{\omega}{\omega_0}\right)^2 + 2jm\frac{\omega}{\omega_0}} \end{aligned} \quad (3.101)$$

Denoting by r the ratio $\frac{\omega}{\omega_0}$ the log magnitude of the frequency response reads:

$$r = \frac{\omega}{\omega_0} \Rightarrow 20 \log_{10} \|F(jr)\| = -10 \log_{10} \left((1 - r^2)^2 + (2mr)^2 \right) \quad (3.102)$$

Thus when $r \rightarrow 0$ the log magnitude of $F(j\omega)$ tends towards 0 whereas when $r \rightarrow \infty$ the log magnitude of $F(j\omega)$ is approximated by $-10 \log_{10} (r^4)$ that is $-40 \log_{10} (r)$. The two asymptotes cross each other when the frequency ω is equal to the corner frequency ω_c . Corner frequency ω_c is defined by the following equation:

$$0 = -40 \log_{10} (r) \Leftrightarrow r = 1 \Leftrightarrow \omega_c = \omega_0 \quad (3.103)$$

Furthermore it is worth noticing that second order model exhibits a resonant peak as soon as $0 < m < \frac{1}{\sqrt{2}}$. The frequency of the resonant peak can be computed by setting to zero the first derivative of the magnitude of $F(jr)$:

$$\begin{aligned} \frac{d}{dr} |F(jr)|_{r=r_m} = 0 &\Leftrightarrow \frac{d}{dr} \left((1 - r^2)^2 + (2mr)^2 \right)_{r=r_m} = 0 \\ \Rightarrow \left\{ \begin{array}{l} r_m = \sqrt{1 - 2m^2} \quad \forall m < \frac{1}{\sqrt{2}} \approx 0.707 \\ 20 \log_{10} |F(jr_m)| = 20 \log_{10} \left(\frac{1}{2m\sqrt{1 - m^2}} \right) \end{array} \right. \end{aligned} \quad (3.104)$$

Thus the resonant frequency is lower than the natural undamped frequency ω_0 and is given by the relation:

$$\omega_r = \omega_0 r_m = \omega_0 \sqrt{1 - 2m^2} < \omega_0 \quad \forall 0 < m < \frac{1}{\sqrt{2}} \approx 0.707 \quad (3.105)$$

The phase of the frequency response reads:

$$\Phi(\omega) = \arg(F(j\omega)) = -\arctan \left(\frac{2mr}{1 - r^2} \right) \quad (3.106)$$

Thus when $r \rightarrow 0$ the phase of $F(j\omega)$ tends towards 0 whereas when $r \rightarrow \infty$ the phase of $F(j\omega)$ is approximated by -180 deg . At the corner frequency $\omega_c = \omega_0$ (i.e. $r = 1$) the phase is equal to -90 deg .

Figure 3.14 presents the Bode plot as well as the asymptotic Bode plot for a second order model where $m = 0.1$, $\omega_0 = 1$ and $K = 1$.

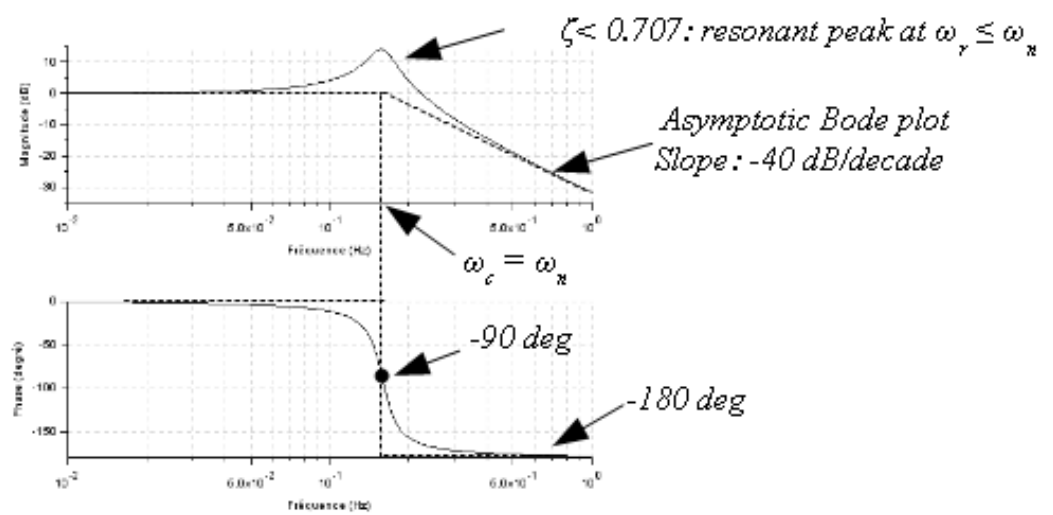


Figure 3.14: Bode plot and asymptotic Bode plot of a second order model where $m = 0.1$, $\omega_0 = 1$ and $K = 1$

Chapter 4

Analysis of Linear Time Invariant systems

4.1 Open-loop versus closed-loop

Usually a plant alone do not fit with the industrial constraints within which it will be used. Typically a plant without controller will not be neither enough precise nor fast. Thus a controller shall be added to the plant (equipped with sensors and actuators) to satisfy industrial specifications. We will denote:

- $F(s)$ the transfer function of the plant;
- $C(s)$ the transfer function of the controller;
- $u(t)$ the plant input (actuator signal) and $y(t)$ the plant output (sensor signal) whose Laplace transform are respectively $\mathcal{L}(y(t)) = Y(s)$ and $\mathcal{L}(u(t)) = U(s)$;
- $r(t)$ the reference input whose Laplace transform is $\mathcal{L}(r(t)) = R(s)$. Reference input $r(t)$ represents what we would like $y(t)$ to be;
- $\epsilon(t) = r(t) - y(t)$ the tracking error as depicted in Figure 4.2. Its Laplace transform is $\mathcal{L}(\epsilon(t)) = \epsilon(s)$.

In that section we will compare open-loop control versus closed-loop control. In open-loop control the output signal $y(t)$ of the plant to be controlled has no effect upon the input of the plant to be controlled as depicted in Figure 4.1. It is not the case for closed-loop control.

Denoting by $C_o(s)$ the open-loop controller, simple algebra shows that the input output relation $G_o(s)$ of the open-loop control depicted in Figure 4.1 reads:

$$Y(s) = C_o(s)F(s)R(s) \Rightarrow G_o(s) = \frac{Y(s)}{R(s)} = C_o(s)F(s) \quad (4.1)$$

On the other hand closed-loop control, or feedback control loop, is a control pattern within which the output signal $y(t)$ of the plant to be controlled is returned back and compared to the reference input to form the system control as depicted in Figure 4.2.

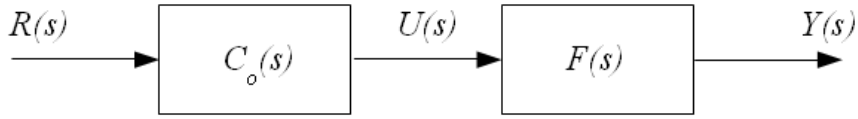


Figure 4.1: Open-loop control

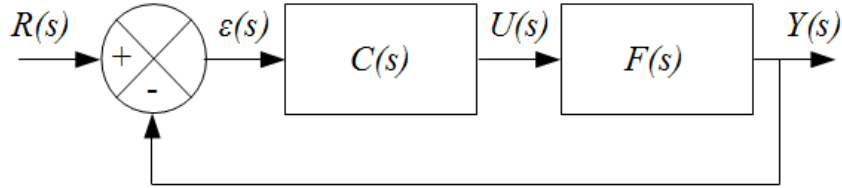


Figure 4.2: Closed-loop control with controller in the direct path

Denoting by $C(s)$ the closed-loop controller, simple algebra shows that the input output relation $G(s)$ of the closed-loop control depicted in Figure 4.2 reads:

$$Y(s) = C(s)F(s) (R(s) - Y(s)) \Rightarrow G(s) = \frac{Y(s)}{R(s)} = \frac{C(s)F(s)}{1 + C(s)F(s)} \quad (4.2)$$

It is worth noticing that other feedback loop configurations exist. For example the closed-loop controller $C(s)$ may be put in the feedback path as depicted in Figure 4.3.

The input output relation of Figure 4.3 reads:

$$Y(s) = F(s) (R(s) - C(s)Y(s)) \Rightarrow \frac{Y(s)}{R(s)} = \frac{F(s)}{1 + C(s)F(s)} \quad (4.3)$$

That is:

$$Y(s) = \frac{C(s)F(s)}{1 + C(s)F(s)} \frac{R(s)}{C(s)} \quad (4.4)$$

The preceding relation indicates that Figure 4.3 is equivalent to Figure 4.2 when the reference input $R(s)$ is replaced by $\frac{R(s)}{C(s)}$.

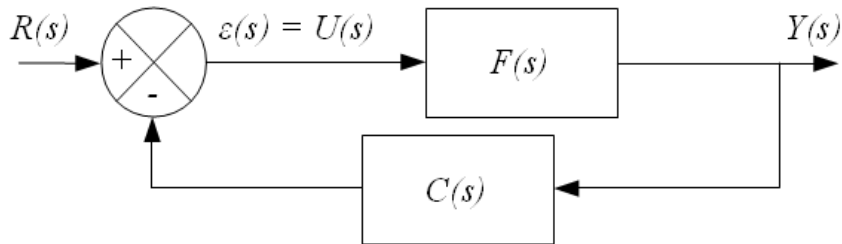


Figure 4.3: Closed-loop control with controller in the feedback path

In the following we will focus on the feedback loop where the controller is situated in the direct path as depicted in Figure 4.2.

When comparing (4.1) and (4.2) it is clear that the open-loop controller $C_o(s)$ can be obtained from the closed-loop controller $C(s)$ by choosing:

$$C_o(s)F(s) = \frac{C(s)F(s)}{1 + C(s)F(s)} \Rightarrow C_o(s) = \frac{C(s)}{1 + C(s)F(s)} \quad (4.5)$$

Consequently it seems that open-loop control and closed-loop control are equivalent control. Nevertheless we shall have in mind that the plant model $F(s)$ often comes from linearization and simplification, and thus is uncertain. Thus we will study the sensitivity of both open-loop scheme and closed-loop scheme with respect to the plant model uncertainty thanks to the sensitivity function S_α^H defined by:

$$S_\alpha^H = \frac{\partial H/H}{\partial \alpha/\alpha} = \frac{\alpha}{H} \frac{\partial H}{\partial \alpha} \quad (4.6)$$

Basically S_α^H relates the relative change of quantity H with respects to the relative change of quantity α .

Specializing the sensitivity function definition to the case where H is the open-loop transfer function $G_o(s)$ and α the uncertain transfer function $F(s)$ of the plant we get:

$$S_F^{G_o} = \frac{F}{G_o} \frac{\partial G_o}{\partial F} = \frac{F}{C_o F} \frac{\partial}{\partial F} (C_o F) = \frac{F}{C_o F} C_o = 1 \quad (4.7)$$

The preceding relation indicates that any change in the plant transfer function $F(s)$ is totally transferred into the open-loop control scheme whatever controller $C_o(s)$ is.

On the other hand we will now specialize the sensitivity function definition to the case where H is the closed-loop transfer function $G(s)$ and α the uncertain transfer function $F(s)$ of the plant:

$$S_F^G = \frac{F}{G} \frac{\partial G}{\partial F} = \frac{F}{\frac{CF}{1+CF}} \frac{\partial}{\partial F} \left(\frac{CF}{1+CF} \right) = \frac{1+CF}{C} \frac{\partial}{\partial F} \left(\frac{CF}{1+CF} \right) \quad (4.8)$$

Let's compute the following expression:

$$\frac{\partial}{\partial F} \left(\frac{CF}{1+CF} \right) = \frac{C(1+CF) - CFC}{(1+CF)^2} = \frac{C}{(1+CF)^2} \quad (4.9)$$

We finally get:

$$S_F^G = \frac{1}{1+CF} \quad (4.10)$$

Thus the sensitivity function $S(s)$ of the closed-loop system reads:

$$S_F^G \equiv S(s) = \frac{1}{1 + C(s)F(s)} \quad (4.11)$$

The preceding relation clearly indicates that as soon as the product $C(s)F(s)$ is *high* within the frequency range of the uncertain plant $F(s)$ then

the closed-loop control scheme allows a great reduction of the sensitivity of the controlled system with respects to uncertainties. Same result can be achieved when comparing the sensitivity of open-loop control scheme and closed-loop control scheme with respect to external disturbances.

As a consequence the central idea to control a plant is the feedback loop where the output signal $y(t)$ of the plant to be controlled is returned back and compared to the reference input to form the system control as depicted in Figure 4.2.

4.2 Stability

A causal signal $y(t)$ is bounded if there is a finite value $B > 0$ such that the signal magnitude never exceeds B , that is $\|y(t)\| \leq B \forall t \geq 0$. A system is said to be Bounded-Input Bounded-Output (BIBO) stable if the output is bounded for every bounded input.

Assuming that 0 is not a pole of the transfer function $H(s)$ and that all of the poles of $H(s)$ are unique the transfer function $H(s)$ reads:

$$H(s) = \frac{b_0 + b_1s + \dots + b_{m-1}s^{m-1} + b_ms^m}{a_0 + a_1s + \dots + a_{n-1}s^{n-1} + s^n} = \frac{K(s + z_1) \cdots (s + z_m)}{(s + \lambda_1) \cdots (s + \lambda_n)} \quad (4.12)$$

We recall that z_i are the zeros of the transfer function $H(s)$ whereas λ_i are the poles of $H(s)$: the magnitude of transfer function $H(s)$ will go to zero at the zeros and to infinity at the poles.

Let's compute the unit step response of a linear time invariant system. We recall that the Laplace transform of the unit step function is $1/s$. Thus the Laplace transform of the unit step response reads:

$$Y(s) = \frac{H(s)}{s} = \frac{1}{s} \frac{K(s + z_1) \cdots (s + z_m)}{(s + \lambda_1) \cdots (s + \lambda_n)} \quad (4.13)$$

Under the condition that $m \leq n$ the Laplace transform $Y(s)$ of the unit step response $y(t)$ can be decomposed into partial fraction expansion where each coefficient α_i , $i \neq 0$ is the residue around the pole λ_i of $H(s)$:

$$Y(s) = \frac{\alpha_0}{s} + \frac{\alpha_1}{s + \lambda_1} + \dots + \frac{\alpha_n}{s + \lambda_n} \quad (4.14)$$

The inverse Laplace transform of $\frac{\alpha_0}{s}$ is $\alpha_0\Gamma(t)$ (we recall that $\Gamma(t)$ denotes the unit step function) and the inverse Laplace transform of $\frac{\alpha_i}{s + \lambda_i}$ is $\alpha_i e^{-\lambda_i t} \forall t \geq 0$. As a consequence the unit step response $y(t)$ is composed of:

- one constant and bounded term, $\alpha_0\Gamma(t)$,
- and of a sum of terms of the form $\alpha_i e^{-\lambda_i t}$. Those terms decay within an exponential envelope as soon as each pole λ_i has a negative real part.

If one of the pole λ_i of $H(s)$ is zero then the term $\alpha_i e^{-\lambda_i t}$ never decays or grows in amplitude; this is called marginal stability. If at least one pole λ_i has a positive real part then at least one element of the response grows without

bound and the system is said to be unstable. It is worth noticing that the same conclusion appears when the multiplicity of a pole is greater than one; indeed we have seen that the inverse Laplace transform of $\frac{1}{(s+\lambda_i)^n}$ is $\frac{t^{n-1}}{(n-1)!}e^{-\lambda_i t} \forall t \geq 0$: as a consequence this term tends towards 0 when $t \rightarrow \infty$ as soon as the real part of the pole λ_i is negative.

As a consequence a *linear time invariant system is stable if and only if all the poles of its transfer function, that are also the roots of the characteristic polynomial, occur in the left half of the complex plane*. Marginal stability occurs when some poles have a null real part whereas instability occurs when at least one pole occurs in the right half of the complex plane.

Notice that the equivalent condition for BIBO stability of linear time invariant (LTI) system is that its impulse response $h(t)$ be absolutely integrable:

$$\int_0^{\infty} |h(t)| dt < \infty \quad (4.15)$$

For linear time invariant (LTI) systems Bounded-Input Bounded-Output (BIBO) stability is also equivalent to *exponential* stability, meaning that after a certain amount of time the impulse response of the system is such that its norm is lower than $\alpha e^{-\gamma t}$ for finite α and $\gamma > 0$.

4.3 Routh-Hurwitz criterion

A simple means of determining the stability of a system can be obtained by the Routh stability criterion; this criterion indicates whether any of the roots of a polynomial have positive real parts, without actually solving for the roots. Consider the following polynomial equation:

$$a_n s^n + a_{n-1} s^{n-1} + \dots + a_1 s + a_0 = 0 \quad (4.16)$$

Necessary but not sufficient conditions so that no roots of Equation (4.49) have positive real parts are that:

- All the coefficients must exist
- All the coefficients of the equation must have the same sign

The first stage to apply the Routh criterion is to build the Routh array, which contains $n + 1$ rows; the two first rows contain coefficients a_i of the polynomial arranged alternatively as indicated in (4.17).

s^n	$a_n \equiv r_{n,1}$	$a_{n-2} \equiv r_{n,2}$	$a_{n-4} \equiv r_{n,3}$	\dots
s^{n-1}	$a_{n-1} \equiv r_{n-1,1}$	$a_{n-3} \equiv r_{n-1,2}$	$a_{n-5} \equiv r_{n-1,3}$	\dots
s^{n-2}	$r_{n-2,1}$	$r_{n-2,2}$	\vdots	
\vdots	\vdots	\vdots	\vdots	
s^1	$r_{1,1}$	$r_{1,2}$	0	
s^0	$r_{0,1}$	0	0	

(4.17)

Coefficients $r_{i,j}$ are computed as follows:

$$r_{i,j} = \frac{r_{i+1,1}r_{i+2,j+1} - r_{i+2,1}r_{i+1,j+1}}{r_{i+1,1}} \quad (4.18)$$

For example:

$$\begin{cases} r_{n-2,1} = \frac{r_{n-1,1}r_{n,2} - r_{n,1}r_{n-1,2}}{r_{n-1,1}} = \frac{a_{n-1}a_{n-2} - a_n a_{n-3}}{a_{n-1}} \\ r_{n-2,2} = \frac{r_{n-1,1}r_{n,3} - r_{n,1}r_{n-1,3}}{r_{n-1,1}} = \frac{a_{n-1}a_{n-4} - a_n a_{n-5}}{a_{n-1}} \end{cases} \quad (4.19)$$

Then the Routh array is continued horizontally and vertical until the last row is completed.

The last step consists in investigating the signs of the numbers in the first column of the Routh array. Routh stability criterion states:

- if all the number of the first column have the same sign then all the roots of the polynomial have negative real parts; when the polynomial is the denominator of the transfer function the system is therefore stable;
- if the numbers in the first column change sign then the number of sign changes indicates the number of roots of the polynomial having positive real parts; when the polynomial is the denominator of the transfer function the system is therefore unstable;

An Hurwitz polynomial, named after Adolf Hurwitz (1859 - 1919, German mathematician), is a polynomial with positive coefficients and for which the real part of every root is zero or negative.

Example 4.1. Consider the following polynomial:

$$D(s) = s^3 + 14s^2 + 41s - 56 \quad (4.20)$$

All the coefficients of $D(s)$ exist and have the same sign; so the necessary condition for $D(s)$ to have all its roots in the left half (complex) plane (LHP) are satisfied. The Routh array reads:

s^3	1	41	(4.21)
s^2	14	-56	
s^1	$\frac{14 \times 41 - 1 \times (-56)}{14} = 45$		
s^0	-56		

There is one change in sign in the first column; consequently $D(s)$ has one root in the right half (complex) plane (RHP). ■

Example 4.2. Consider the following polynomial:

$$D(s) = s^4 + 5s^3 + s^2 + 10s + 1 \quad (4.22)$$

All the coefficients of $D(s)$ exist and have the same sign; so the necessary condition for $D(s)$ to have all its roots in the left half (complex) plane (LHP) are

satisfied. The Routh array reads:

s^4	1	1	1
s^3	5	10	
s^2	-1	1	
s^1	15		
s^0	1		

(4.23)

There are two changes in sign in the first column; consequently $D(s)$ has two roots in the right half (complex) plane (RHP). ■

It is worth noticing that:

- Applying the Routh criterion on the polynomial $D(a + s)$ enables to compute the number of roots of $D(s)$ with a real part greater than a (indeed $s = a + s - a$ and $\text{Re}(s) < 0 \Leftrightarrow \text{Re}(a + s) < a$).

Example 4.3. Let's check how many roots with real part greater than $a = -2$ has the following polynomial:

$$D(s) = s + 1 \quad (4.24)$$

First we form the polynomial $D(a + s) = D(s - 2) = (s - 2) + 1 = s - 1$ and then we apply the Routh criterion on the following Routh array:

s^1	1
s^0	-1

(4.25)

There is one change in sign in the first column; consequently $D(s)$ has one root with a real part greater than $a = -2$ (it is obviously -1)

■

Example 4.4. Let's check how many roots with real part greater than $a = 2$ has the following polynomial:

$$D(s) = (s - 1)(s - 4) \quad (4.26)$$

First we form the polynomial $D(a + s) = D(s + 2) = (s + 2 - 1)(s + 2 - 4) = s^2 - s - 2$ and then we apply the Routh criterion on the following Routh array:

s^2	1	-2
s^1	-1	
s^0	-2	

(4.27)

There is one change in sign in the first column; consequently $D(s)$ has one root with a real part greater than $a = 2$ (it is obviously 4)

■

- To multiply all the numbers in a row by a strictly positive number do not change the conclusion of the Routh criterion.

- As far as the sign of the real part of a complex number is the same than the sign of the real part of its inverse, it is equivalent to test either polynomial $D(s)$ or polynomial $D(1/s)$ where s has been replaced by $1/s$.

Obviously if 0 appears in the first position of a row, then the elements in the following row will be infinite and the Routh criterion breaks down. Similarly the Routh criterion breaks down when all the numbers in a row are 0.

If a first column term in any row is 0, but the remaining terms are not 0 or there is no remaining term, then the 0 term is replaced by a very small number ϵ with the same sign than the coefficient above the 0 and the rest of the array is evaluated.

- If the sign of the coefficient above the $0 \rightarrow \epsilon$ is the same that below it, it indicates that there are a pair of imaginary roots.
- If however the sign of the coefficient above the $0 \rightarrow \epsilon$ is opposite that below it, it indicates that there is one sign change.

Alternatively, when a zero occurs in the first column, it is possible to create the Routh table using the polynomial $D(1/s)$ that has the reciprocal roots of the original polynomial $D(s)$. Tsu-Shuan Chang and Chi-Tsong Chen¹ suggest to find the greatest common divisor $d(s)$ of $D_1(s)$ and $D_2(s)$, where $D_1(s)$ and $D_2(s)$ are the odd and the even part of $D(s)$, and then to apply the Routh criterion to $D(s)/d(s)$.

Example 4.5. Consider the following polynomial:

$$D(s) = s^3 - 3s + 2 \quad (4.28)$$

The Routh array reads:

s^3	1	-3	(4.29)
s^2	$0 \rightarrow \epsilon^+$	2	
s^1	$\frac{-3\epsilon^+ - 2}{\epsilon^+} = -3 - \frac{2}{\epsilon^+}$		
s^0	2		

As soon as $\epsilon^+ \rightarrow 0$, there is one sign change between ϵ^+ (which has the same sign than the coefficient above, that is 1, which is positive) and $-3 - \frac{2}{\epsilon^+}$ (which is negative as soon as $\epsilon^+ \rightarrow 0$) and another sign change between $-3 - \frac{2}{\epsilon^+}$ (which is negative as soon as $\epsilon^+ \rightarrow 0$) and 2. Consequently, we conclude that $D(s)$ has two roots with positive real part. Actually polynomial $D(s)$ reads:

$$D(s) = (s - 1)^2(s + 2) \quad (4.30)$$

Routh table coming from $D(1/s)$ reads as follows:

$$D(1/s) = (1/s)^3 - 3(1/s) + 2 = \frac{1 - 3s^2 + 2s^3}{s^3} \quad (4.31)$$

¹Tsu-Shuan Chang and Chi-Tsong Chen, On the Routh-Hurwitz criterion, IEEE Transactions on Automatic Control, vol. 19, no. 3, pp. 250-251, June 1974, doi: 10.1109/TAC.1974.1100537.

The Routh array of the numerator of $D(1/s)$, that is polynomial $1 - 3s^2 + 2s^3$, reads:

s^3	2	0
s^2	-3	1
s^1	$\frac{-2}{-3} = \frac{2}{3}$	
s^0	1	

(4.32)

We get the same conclusion than previously: as far as there are two changes in sign in the first column, we conclude that $D(1/s)$, and also $D(s)$ has two roots with a real part greater than zero. ■

Example 4.6. Consider the following polynomial:

$$D(s) = s^3 + 2s^2 + s + 2 \quad (4.33)$$

All the coefficients of $D(s)$ exist and have the same sign; so the necessary condition for $D(s)$ to have all its roots in the left half (complex) plane (LHP) are satisfied. The Routh array reads:

s^3	1	1
s^2	2	2
s^1	$0 \rightarrow \epsilon^+$	
s^0	2	

(4.34)

As far as the sign of the coefficient above ϵ^+ is the same that below it, it indicates that there are a pair of imaginary roots. Actually $D(s)$ has two roots at $\pm j$.

Routh table coming from $D(1/s)$ reads as follows:

$$D(1/s) = (1/s)^3 + 2(1/s)^2 + (1/s) + 2 = \frac{1 + 2s + s^2 + 2s^3}{s^3} \quad (4.35)$$

The Routh array of the numerator of $D(1/s)$, that is polynomial $1 + 2s + s^2 + 2s^3$, reads:

s^3	2	2
s^2	1	1
s^1	$0 \rightarrow \epsilon^+$	
s^0	1	

(4.36)

We get the same conclusion than previously: as far as the sign of the coefficient above ϵ^+ is the same that below it, it indicates that there are a pair of imaginary roots. Actually $D(s)$ has two roots at $\pm j$. ■

If all the coefficients in any derived row are zeros it indicates that polynomial $D(s)$ has roots of equal magnitude lying radially opposite in the s -plane, that is two real roots with equal magnitude and opposite sign and/or two conjugate imaginary roots. In such a case the evaluation of the rest of the array can be continued by forming an auxiliary polynomial $Q(s)$ with the coefficients of the

last non-zero's row and by replacing the zero's row by the coefficients of the derivative of this polynomial. It is worth noticing that in this situation the roots of the auxiliary polynomial $Q(s)$ are also roots of the polynomial $D(s)$ under consideration.

Example 4.7. *The previous example where $D(s) = s^3 + 2s^2 + s + 2$ can be tackled with this rule. Indeed all the coefficients (in fact just one) in the row s^1 are zeros:*

$$\begin{array}{c|c|c} s^3 & 1 & 1 \\ \hline s^2 & 2 & 2 \\ \hline s^1 & 0 & \end{array} \quad (4.37)$$

Then we form an auxiliary polynomial with the coefficients of the last non-zero row:

$$Q(s) = 2s^2 + 2 \quad (4.38)$$

It can be checked that the roots of $Q(s)$, which are $(+j, -j)$, are also roots of $D(s)$.

The derivative of $Q(s)$ reads:

$$\frac{d}{ds}Q(s) = 4s \quad (4.39)$$

The Routh array can be continued by replacing the zero's row by the coefficients of the derivative of $Q(s)$, here 4:

$$\begin{array}{c|c|c} s^3 & 1 & 1 \\ \hline s^2 & 2 & 2 \\ \hline s^1 & 4 & \leftarrow \text{coefficients of } \frac{d}{ds}Q(s) \end{array} \quad (4.40)$$

The complete Routh array finally reads:

$$\begin{array}{c|c|c} s^3 & 1 & 1 \\ \hline s^2 & 2 & 2 \\ \hline s^1 & 4 & \\ \hline s^0 & 2 & \end{array} \quad (4.41)$$

As far as there is no sign change in the coefficients of the first column we conclude that $D(s)$ has no root with strictly positive real part. Actually the third root of $D(s)$ is -2 , which obviously has a negative real part and explains why there is no sign change in the Routh array. ■

Example 4.8. *Consider the following polynomial:*

$$D(s) = s^5 + 2s^4 + 24s^3 + 48s^2 + 25s + 50 \quad (4.42)$$

All the coefficients of $D(s)$ exist and have the same sign; so the necessary condition for $D(s)$ to have all its roots in the left half (complex) plane (LHP) are satisfied. The first rows of the Routh array reads:

$$\begin{array}{c|c|c|c} s^5 & 1 & 24 & 25 \\ \hline s^4 & 2 & 48 & 50 \\ \hline s^3 & 0 & 0 & \end{array} \quad (4.43)$$

All the coefficients of the s^3 'row are zeros. Then we form an auxiliary polynomial with the coefficients of the last non-zero row:

$$Q(s) = 2s^4 + 48s^2 + 50 \quad (4.44)$$

It can be checked that the roots of $Q(s)$, which are denoted $(\lambda_1, \lambda_2, \lambda_3, \lambda_4)$, are also roots of $D(s)$:

$$\begin{cases} \lambda_{1,2}^2 = \frac{-48 + \sqrt{48^2 - 4 \times 2 \times 50}}{4} \Rightarrow \lambda_{1,2} = \pm j \sqrt{\frac{48 - \sqrt{1904}}{4}} \\ \lambda_{3,4}^2 = \frac{-48 - \sqrt{48^2 - 4 \times 2 \times 50}}{4} \Rightarrow \lambda_{3,4} = \pm j \sqrt{\frac{48 + \sqrt{1904}}{4}} \end{cases} \quad (4.45)$$

The derivative of $Q(s)$ reads:

$$\frac{d}{ds}Q(s) = 8s^3 + 96s \quad (4.46)$$

The Routh array can be continued by replacing the zero's row by the coefficients of the derivative of $Q(s)$, here 8 and 96:

s^5	1	24	25
s^4	2	48	50
s^3	8	96	\leftarrow coefficients of $\frac{d}{ds}Q(s)$

(4.47)

The complete Routh array finally reads:

s^5	1	24	25
s^4	2	48	50
s^3	8	96	
s^2	24	50	
s^1	238/3		
s^0	50		

(4.48)

As far as there is no sign change in the coefficients of the first column we conclude that $D(s)$ has no root with strictly positive real part. Actually the fifth root of $D(s)$ is -2 , which obviously has a negative real part and explains why there is no sign change in the Routh array. ■

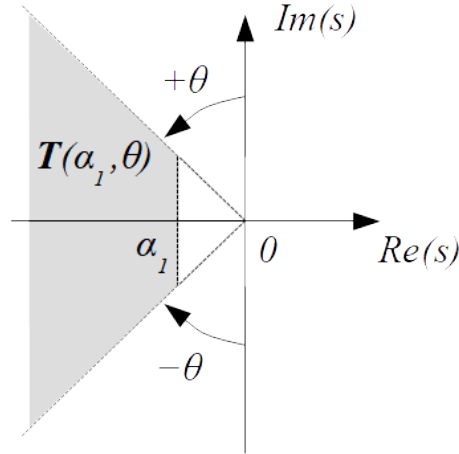
4.4 Roots in a specified trapezoidal region

Let $D(s)$ be the following real polynomial:

$$D(s) = a_n s^n + a_{n-1} s^{n-1} + \cdots + a_1 s + a_0 \quad (4.49)$$

Let \bar{s} be the complex conjugate of s and define the following mapping polynomials:

$$\begin{cases} D_\theta(w) := D(s)D(\bar{s})|_{s=we^{j\theta}} = D(we^{j\theta})D(we^{-j\theta}) \\ D_{\alpha_1}(w) := D(s)|_{s=\alpha_1+w} = D(\alpha_1+w) \end{cases} \quad (4.50)$$

Figure 4.4: Trapezoidal region $T(\alpha_1, \theta)$

Following Kuo & al.² all the roots of $D(s)$ are situated in the trapezoidal region $T(\alpha_1, \theta)$ shown in Figure 4.4 if and only all the roots $D_\theta(w)$ and $D_{\alpha_1}(w)$ are situated in the left half of the w -plane. This can be tested thanks to the Routh-Hurwitz criterion.

This result can be extended to the case where we add the constraint that all the roots of $D(s)$ shall have a real part *greater* than α_2 . This will be the case if and only all the roots of $D_\theta(w)$, $D_{\alpha_1}(w)$ and $D_{\alpha_2}(w)$ are situated in the left half of the w -plane, where:

$$D_{\alpha_2}(w) := D(s)|_{s=\alpha_2-w} = D(\alpha_2 - w) \quad (4.51)$$

4.5 Hermite's criterion

Let $D(s)$ be the following real polynomial:

$$D(s) = a_n s^n + a_{n-1} s^{n-1} + \dots + a_1 s + a_0 \quad (4.52)$$

Let polynomials $u(s)$ and $v(s)$ be defined as follows where $j^2 = -1$:

$$D(j s) := u(s) + j v(s) \Leftrightarrow \begin{cases} u(s) = \text{Re}(D(j s)) \\ v(s) = \text{Im}(D(j s)) \end{cases} \quad (4.53)$$

Hermite's criterion³ states that real polynomial $D(s)$ has all its roots with negative real part if and only if the $n \times n$ Bezoutian matrix $\mathbf{B} := [b_{ij}]$ is positive

². C. Kuo and Y. J. Huang, Pole placement in a specified trapezoidal region for uncertain linear control systems, 7th Asian Control Conference, 2009, pp. 1286-1289.

³Akin Delibasi and Didier Henrion, Hermite matrix in Lagrange basis for scaling static output feedback polynomial matrix inequalities, 2010 International Journal of Control, Vol. 83, pp. 2494 - 2505

definite. The entries $[b_{ij}]$ of symmetric matrix \mathbf{B} are defined by:

$$\begin{aligned} \sum_{i=0}^{n-1} \sum_{j=0}^{n-1} b_{ij} x^i y^j &= [1 \quad x \quad \cdots \quad x^{n-1}] \mathbf{B} \begin{bmatrix} 1 \\ y \\ \vdots \\ y^{n-1} \end{bmatrix} \\ &= \frac{\begin{vmatrix} v(x) & u(x) \\ v(y) & u(y) \end{vmatrix}}{x-y} = \frac{v(x)u(y)-v(y)u(x)}{x-y} \end{aligned} \tag{4.54}$$

By treating $\frac{1}{x-y}$ as the formal power series $\sum_{k=0}^{\infty} x^{-k} y^{k-1}$, we can write⁴ the quotient as a polynomial of degree $n - 1$ in x :

$$\begin{aligned} \frac{1}{x-y} &= \sum_{k=0}^{\infty} x^{-k} y^{k-1} = x^{-1} + yx^{-2} + y^2x^{-3} + \cdots \\ &\Rightarrow \frac{v(x)u(y)-v(y)u(x)}{x-y} = \sum_{i=0}^{n-1} \left(\sum_{j=i+1}^n c_j(y) y^{j-i-1} \right) x^i \end{aligned} \tag{4.55}$$

Alternatively, write $u(s)$ and $v(s)$ as polynomials with the *same* degree n :

$$\begin{cases} v(s) = v_0 + v_1 s + \cdots + v_n s^n \\ u(s) = u_0 + u_1 s + \cdots + u_n s^n \end{cases} \tag{4.56}$$

Let $L_i = [v_i \quad u_i]$ and $R_i = \begin{bmatrix} -u_i \\ v_i \end{bmatrix}$. Then it can be shown⁴ that the $n \times n$ symmetric matrix \mathbf{B} can be computed as the product of a lower triangular matrix $[L_i]$ of size $n \times 2n$ and an upper triangular matrix $[R_j]$ of size $n \times 2n$:

$$\begin{aligned} \begin{cases} L_i = [v_i \quad u_i] \\ R_i = \begin{bmatrix} -u_i \\ v_i \end{bmatrix} \end{cases} \Rightarrow \mathbf{B} &= \begin{bmatrix} b_{0,0} & \cdots & b_{0,n-1} \\ \vdots & \ddots & \vdots \\ b_{n-1,0} & \cdots & b_{n-1,n-1} \end{bmatrix} \\ &= \begin{bmatrix} L_0 & & 0 \\ \vdots & \ddots & \\ L_{n-1} & \cdots & L_0 \end{bmatrix} \begin{bmatrix} R_1 & \cdots & R_n \\ \vdots & \ddots & \\ R_n & & 0 \end{bmatrix} \end{aligned} \tag{4.57}$$

Finally, we remind that a real $n \times n$ symmetric matrix $\mathbf{B} = \mathbf{B}^T$ is positive definite if and only if we have either:

- $\underline{x}^T \mathbf{B} \underline{x} > 0$ for all $\underline{x} \neq 0$;
- All eigenvalues of \mathbf{B} are strictly positive;
- All of the leading principal minors are strictly positive (the leading principal minor of order k is the minor of order k obtained by deleting the last $n - k$ rows and columns).

⁴Eng-Wee Chionh, Ming Zhang, Ronald N. Goldman, Fast Computation of the Bezout and Dixon Resultant Matrices, Journal of Symbolic Computation, Volume 33, Issue 1, 2002, Pages 13-29, ISSN 0747-7171, <https://doi.org/10.1006/jsc.2001.0462>.

Example 4.9. We consider the following polynomial where K is a design parameter:

$$D(s) = 24s^3 + 26s^2 + 9s - 1 + K \quad (4.58)$$

Then we have:

$$\begin{cases} u(s) = \operatorname{Re}(D(js)) = -26s^2 + K - 1 \\ v(s) = \operatorname{Im}(D(js)) = -24s^3 + 9s \end{cases} \quad (4.59)$$

To get coefficients $[b_{ij}]$ of the 3×3 Bezoutian matrix \mathbf{B} we can write:

$$\begin{aligned} \frac{v(x)u(y) - v(y)u(x)}{x-y} &= [1 \quad x \quad \dots \quad x^{3-1}] \mathbf{B} \begin{bmatrix} 1 \\ y \\ \vdots \\ y^{3-1} \end{bmatrix} \\ &= [1 \quad x \quad x^2] \begin{bmatrix} b_{0,0} & b_{0,1} & b_{0,2} \\ b_{1,0} & b_{1,1} & b_{1,2} \\ b_{2,0} & b_{2,1} & b_{2,2} \end{bmatrix} \begin{bmatrix} 1 \\ y \\ y^2 \end{bmatrix} \end{aligned} \quad (4.60)$$

where:

$$\begin{aligned} \begin{bmatrix} b_{0,0} & b_{0,1} & b_{0,2} \\ b_{1,0} & b_{1,1} & b_{1,2} \\ b_{2,0} & b_{2,1} & b_{2,2} \end{bmatrix} &= \begin{bmatrix} L_0 & 0 & 0 \\ L_1 & L_0 & 0 \\ L_2 & L_1 & L_0 \end{bmatrix} \begin{bmatrix} R_1 & R_2 & R_3 \\ R_2 & R_3 & 0 \\ R_3 & 0 & 0 \end{bmatrix} \\ &= \begin{bmatrix} 0 & K-1 & 0 & 0 & 0 & 0 \\ 9 & 0 & 0 & K-1 & 0 & 0 \\ 0 & -26 & 9 & 0 & 0 & K-1 \end{bmatrix} \\ &\quad \begin{bmatrix} 0 & 26 & 0 \\ 9 & 0 & -24 \\ 26 & 0 & 0 \\ 0 & -24 & 0 \\ 0 & 0 & 0 \\ -24 & 0 & 0 \end{bmatrix} \\ &= \begin{bmatrix} 9K-9 & 0 & 24-24K \\ 0 & 258-24K & 0 \\ 24-24K & 0 & 624 \end{bmatrix} \end{aligned} \quad (4.61)$$

We conclude that $D(s)$ has all its roots with negative real part if and only if symmetric matrix $\mathbf{B} = \mathbf{B}^T$ is positive definite, that is if and only if all of the leading principal minors of \mathbf{B} are strictly positive.

Let's compute the principal minors of \mathbf{B} :

$$\begin{cases} m_1 = 9K - 9 \\ m_2 = \det \left(\begin{bmatrix} 9K-9 & 0 \\ 0 & 258-24K \end{bmatrix} \right) = -54(4K-43)(K-1) \\ m_3 = \det(\mathbf{B}) = 13824K^3 - 311040K^2 + 1894752K - 1597536 \end{cases} \quad (4.62)$$

Thus:

$$\begin{cases} m_1 > 0 \\ m_2 > 0 \\ m_3 > 0 \end{cases} \Rightarrow 1 < K < \frac{43}{4} \quad (4.63)$$

■

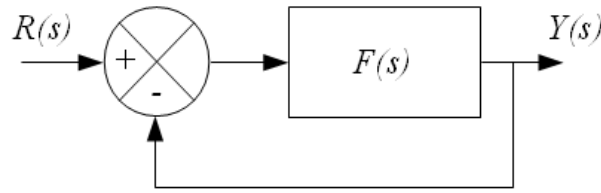


Figure 4.5: Unity feedback loop

4.6 Nyquist stability criterion

Consider Figure 4.5 where the loop transfer function $F(s)$ is assumed to be strictly proper (i.e. $\lim_{s \rightarrow \infty} F(s) = 0$).

The transfer function of the closed-loop system is:

$$G(s) = \frac{Y(s)}{R(s)} = \frac{1}{1 + F(s)} \quad (4.64)$$

The Nyquist stability criterion determines the stability of a closed-loop from the open-loop poles and its open-loop frequency response. More specifically the Nyquist stability criterion determines the stability of $G(s)$ from poles of $F(s)$ and the Nyquist plot of $F(j\omega)$.

Denoting by Re the real part and by Im the imaginary part, the Nyquist plot of $F(j\omega)$ is the locus of vector $(ImF(j\omega), ReF(j\omega))$ when frequency ω varies from $-\infty$ to $+\infty$. As far as $F(s)$ is a fraction where the numerator and the denominator are real coefficients polynomial the Nyquist plot of $F(j\omega)$ is symmetrical with respect to the real axis. In addition as soon as $F(s)$ is assumed to be a strictly proper transfer function the Nyquist plot of $F(j\omega)$ moves toward zero as ω tends to infinity:

$$\lim_{s \rightarrow \infty} F(s) = 0 \Rightarrow \lim_{\omega \rightarrow \pm\infty} F(j\omega) = 0 \quad (4.65)$$

In examining the stability of the unity feedback loop of Figure 4.5 the Nyquist stability criterion states that:

$$Z = P + N \quad (4.66)$$

Where:

- Z is the number of zeros of $1 + F(s)$ in the right half s -plane, that is the number of unstable poles of the closed-loop system whose transfer function is $G(s)$
- P is number of poles of $F(s)$ in the right half s -plane
- N is the number of clockwise encirclements of the $-1 + j0$ point by the Nyquist plot of $F(j\omega)$; point $-1 + j0$ is called the critical point

The Nyquist stability criterion states that *the number of unstable closed-loop poles is equal to the number of unstable open-loop poles plus the number of encirclements of the critical point $-1 + j0$ by the Nyquist plot of $F(j\omega)$* . The encirclement is counted positive in the clockwise direction and negative otherwise.

In others words:

- If there are one or more counterclockwise encirclements of the $-1 + j0$ critical point by the Nyquist plot then the closed-loop system is stable if the number of counterclockwise encirclements is the same as the number of poles of $F(s)$ in the right half s -plane; otherwise the closed-loop system is unstable.
- If there are one or more clockwise encirclements of the $-1 + j0$ critical point this implies that the closed-loop system is unstable.

In current situations $F(s)$ has no pole in the right half s -plane; consequently if the Nyquist plot of $F(j\omega)$ has no encirclement of the $-1 + j0$ critical point this implies that the closed-loop system is stable; otherwise the closed-loop system is unstable.

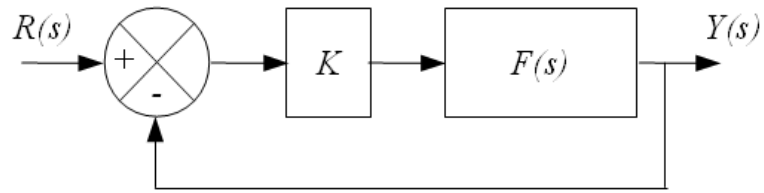
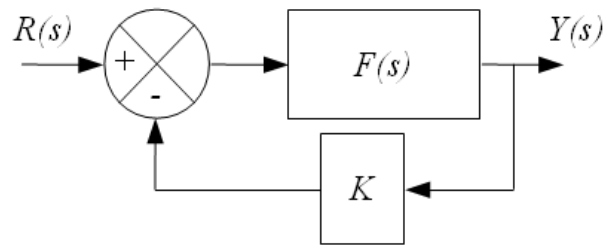
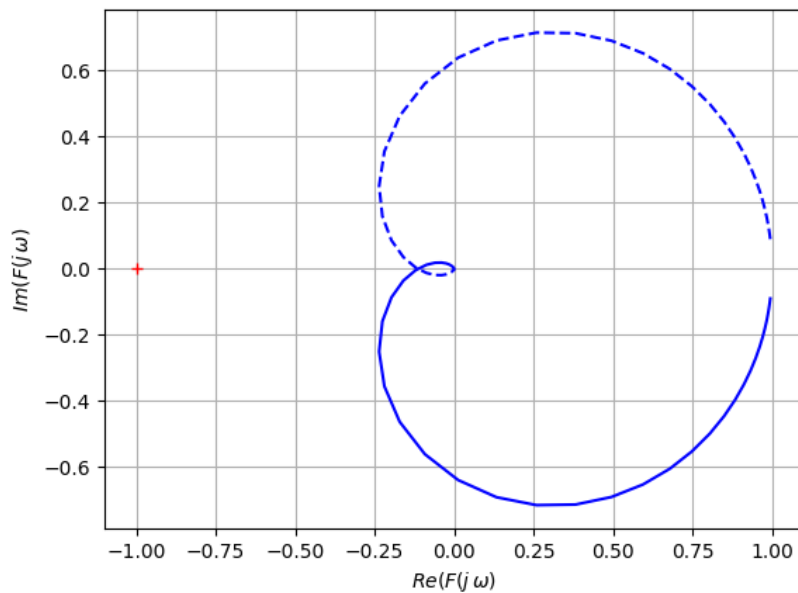
An easy way to determine the number of encirclements of the $-1 + j0$ critical point is to draw a line out from the critical point, in any directions. Then by counting the number of times that the Nyquist plot crosses the line in the clockwise direction (i.e. left to right) and by subtracting the number of times it crosses in the counterclockwise direction then the number of clockwise encirclements of the $-1 + j0$ critical point is obtained. A negative number indicates counterclockwise encirclements.

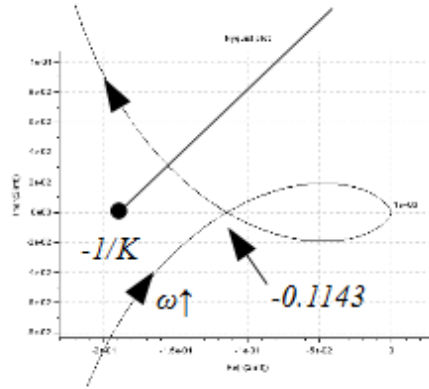
It is worth noticing that when a positive gain K ($K > 0$) is added in the loop, either in the direct path as shown in Figure 4.6 or in the feedback path as shown in Figure 4.7, the denominator of the closed-loop transfer function becomes $1 + KF(s)$. As a consequence the critical point $-1 + j0$ changes to be $-\frac{1}{K} + j0$. The Nyquist stability criterion can then still be applied in both cases by assessing the encirclements of the $-\frac{1}{K} + j0$ critical point where $K > 0$. Consequently the Nyquist stability criterion do not change when positive gain K is added either in the direct path or in the feedback path because the denominator of the closed-loop transfer function $G(s) = \frac{Y(s)}{R(s)}$ remains the same.

More generally we have seen that the denominator of the closed-loop transfer function of feedback loop systems in Figures 4.3 and 4.2 reads $1 + C(s)F(s)$. Thus comparing this denominator with the denominator of $G(s)$ obtained with the feedback loop in Figure 4.5 we conclude that the Nyquist stability criterion still applies on closed-loop transfer function of feedback loop systems in Figures 4.3 and 4.2 as soon as $F(s)$ is replaced by $C(s)F(s)$. In is worth noticing that $C(s)F(s)$ represents the loop transfer function, which is sometimes denoted $L(s)$.

Example 4.10. *Let's consider the following transfer function:*

$$F(s) = \frac{1}{(2s + 1)(3s + 1)(4s + 1)} \quad (4.67)$$

Figure 4.6: Feedback loop with constant gain K in the direct pathFigure 4.7: Feedback loop with constant gain K in the feedback pathFigure 4.8: Nyquist plot of $F(s) = \frac{1}{(2s+1)(3s+1)(4s+1)}$

Figure 4.9: Zoom of the Nyquist plot of $F(j\omega)$

The Nyquist plot of $F(j\omega)$ is shown in Figure 4.8. It is clear that the Nyquist plot of $F(j\omega)$ is symmetrical with respect to the real axis and $\lim_{\omega \rightarrow \pm\infty} F(j\omega) = 0$. We can also check that $F(0) = 1$.

Figure 4.9 is a zoom of Figure 4.8 near the loop of the Nyquist plot.

First we will assume an unit feedback gain (i.e. $K = 1$) in the closed-loop as in Figure 4.5. Then applying the Nyquist stability criterion on the Nyquist plot we conclude that the closed-loop is stable because the number P of poles of $F(s)$ in the right half s -plane is zero and the number N of clockwise encirclements of the $-1 + j0$ critical point by the Nyquist plot is also zero. As a consequence the number Z of zeros of $1 + F(s)$ in the right half s -plane is $N + P = 0$, meaning that there is no unstable pole in the closed-loop system transfer function $G(s)$.

Now we will assume that a constant gain K is inserted in the feedback loop as depicted either in Figure 4.6 or Figure 4.7. Then the critical point becomes $-\frac{1}{K} + j0$. As shown in Figure 4.9 the number of encirclements of the critical point remains to be zero as soon as the positive feedback loop gain K , ($K > 0$) satisfies:

$$-\frac{1}{K} < -0.1143 \Leftrightarrow K < \frac{1}{0.1143} \approx 8.75 \quad (4.68)$$

Consequently the closed-loop is stable as soon as $0 < K < 8.75$. As soon as $K > 8.75$ the number of clockwise encirclements of the critical point is 2, meaning that the closed-loop system is unstable because it has 2 poles in the right half s -plane (RHP).

In order to check this conclusion we can use the Routh criterion. The denominator $D(s)$ of the closed-loop transfer function with constant gain K reads:

$$F(s) = \frac{1}{(2s+1)(3s+1)(4s+1)} = \frac{1}{24s^3 + 26s^2 + 9s + 1} \quad (4.69)$$

$$\Rightarrow D(s) = 24s^3 + 26s^2 + 9s + 1 + K$$

The Routh array of $D(s)$ is the following:

s^3	24	9
s^2	26	$1 + K$
s^1	$\frac{26 \times 9 - 24 \times (1 + K)}{26} = \frac{210 - 24K}{26}$	
s^0	$1 + K$	

(4.70)

Consequently there is no sign change in the first column as soon as :

$$\begin{cases} 210 - 24K > 0 \Leftrightarrow K < \frac{210}{24} \approx 8.75 \\ 1 + K > 0 \Leftrightarrow K > -1 \end{cases} \quad (4.71)$$

When the feedback gain K becomes greater than $\frac{210}{24} \approx 8.75$ then two sign changes appear in the first column of the Routh array and the closed-loop system is then unstable with 2 unstable poles. This is the same conclusion than the one obtained thanks to the Nyquist stability criterion. ■

4.7 Phase and gain margins

In that section we refer to Figure 4.5 and we will assume that the loop transfer function $F(s)$ is stable as well as its inverse $1/F(s)$, meaning that $F(s)$ has both all its poles and zeros in the left half s -plane. In other words transfer function $F(s)$ is minimum phase.

In general the closer the Nyquist plot of the loop transfer function $F(s)$ comes to encircling the $-1 + j0$ critical point, the more oscillatory is the closed-loop system response. Phase and gain margins measures how close is the closed-loop system from the verge of instability. For stable loop transfer function the Nyquist stability criterion states that the closed-loop system is stable as soon as there is no encirclement of the critical point. This can be readily visualized on the Bode plot.

The gain crossover frequency is the frequency ω_c at which the magnitude $\|F(j\omega)\|$ of the loop transfer function $F(s)$ is unity.

At the gain crossover frequency ω_c the phase angle $\Phi(\omega_c) = \arg(F(j\omega_c))$ of the loop transfer function reads as follows where M_Φ represents the phase margin of the closed-loop system:

$$\begin{aligned} \|F(j\omega_c)\| = 1 &\Rightarrow M_\Phi = 180^\circ + \arg(F(j\omega_c)) \quad (\text{modulo } 360^\circ) \\ &\Leftrightarrow \Phi(\omega_c) = \arg(F(j\omega_c)) = -180^\circ + M_\Phi \quad (\text{modulo } 360^\circ) \end{aligned} \quad (4.72)$$

For closed-loop system to be stable the phase margin M_Φ must be positive; it indicates the amount of phase lag which is required to bring the closed-loop system unstable.

The phase crossover frequency is the frequency ω_π at which the phase angle of the loop transfer function $F(s)$ is equal to -180° . At the phase crossover frequency ω_π the magnitude $\|F(j\omega)\|$ of the loop transfer function is the reciprocal of the gain margin Gm :

$$\arg(F(j\omega_\pi)) = -180^\circ \quad (\text{modulo } 360^\circ) \Rightarrow Gm = \frac{1}{\|F(j\omega_\pi)\|} \quad (4.73)$$

Or in terms of decibels:

$$GmdB = 20 \log_{10}(Gm) = -20 \log_{10} \|F(j\omega_\pi)\| \quad (4.74)$$

For a closed-loop system to be stable the gain margin in decibels $GmdB$ must be positive; it indicates how much the gain must be increased before the closed-loop system becomes unstable.

It is worth noticing that the gain margin as well as the phase crossover frequency ω_π at which the gain margin is measured can be obtained through the Routh array. To obtain those values first compute the denominator of the closed-loop transfer function $\frac{K_{cr}F(s)}{1+K_{cr}F(s)}$ where a fictitious gain K_{cr} is used. Then build the corresponding Routh array and compute the value of K_{cr} such that the coefficient in the row s^1 is equal to zero. The gain margin is then obtained as $Gm = K_{cr}$. The phase crossover frequency ω_π is obtained by setting to zero the polynomial extracted from the row s^2 and by replacing s by $j\omega_\pi$.

Similarly, the phase margin as well as the gain crossover frequency ω_c at which the phase margin is measured can be obtained through the Routh array⁵. To obtain those values first compute the denominator of the closed-loop transfer function $\frac{G(s)}{K_{cr}+G(s)}$ where a fictitious gain K_{cr} is used and where $G(s) = \frac{1}{2} \left(F(s) + \frac{1}{F(s)} \right)$. Then build the corresponding Routh array and compute the value of K_{cr} such that the coefficient in the row s^1 is equal to zero. Among all the possible values of K_{cr} such that $|K_{cr}| < 1$ choose the closest value to -1 . The phase margin is then obtained as $M_\phi = \arccos(K_{cr})$. The gain crossover frequency ω_c is obtained by setting to zero the polynomial extracted from the row s^2 and by replacing s by $j\omega_c$.

Figure 4.10 represents the Bode plot of an open-loop stable transfer function $F(s)$ as well as the phase and gain margins of both stable and unstable closed-loop systems.

It is noted that the gain margin of a first or second order system is infinite since the phase for such systems do not cross the -180° phase axis.

For system having two or more gain or phase crossover frequencies the phase or gain margin is measured at the closest point of the critical point which is usually the point at the highest crossover frequency.

Example 4.11. *Figure 4.11 is the Bode plot of the transfer function $F(s)$ which has been studied in the previous example.*

Using the function `g_margin` provided by Scilab, it can be stated that the phase crossover frequency is the frequency $\omega_\pi = 2\pi \times 0.0974621$ rad/sec (on the very right in Figure 4.11) whereas the gain margin in decibels is $GmdB = 18.84$ dB. The maximum gain $K = 8.75$ which let stable the closed-loop system is retrieved through the following computation:

$$20 \log_{10}(K) = 18.84 \Rightarrow K = 10^{\frac{18.84}{20}} = 8.75 \quad (4.75)$$

⁵Yung C.F., Tsai Y.W., Shyu K.K., Applications of Routh-Hurwitz criterion to evaluation of phase margin and phase lead compensation, Electronics Letters, Volume: 27, Issue: 11, 23 May 1991

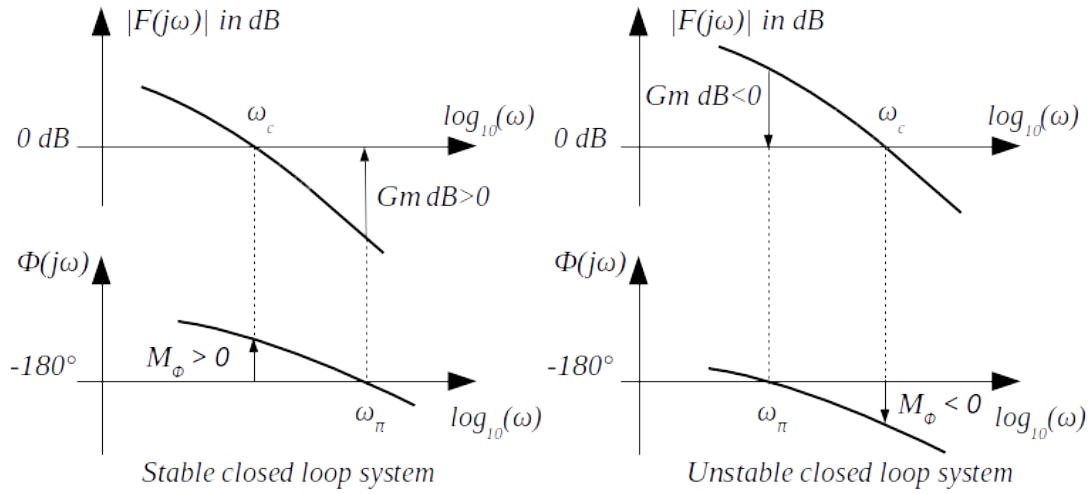


Figure 4.10: Phase and gain margins of stable and unstable systems

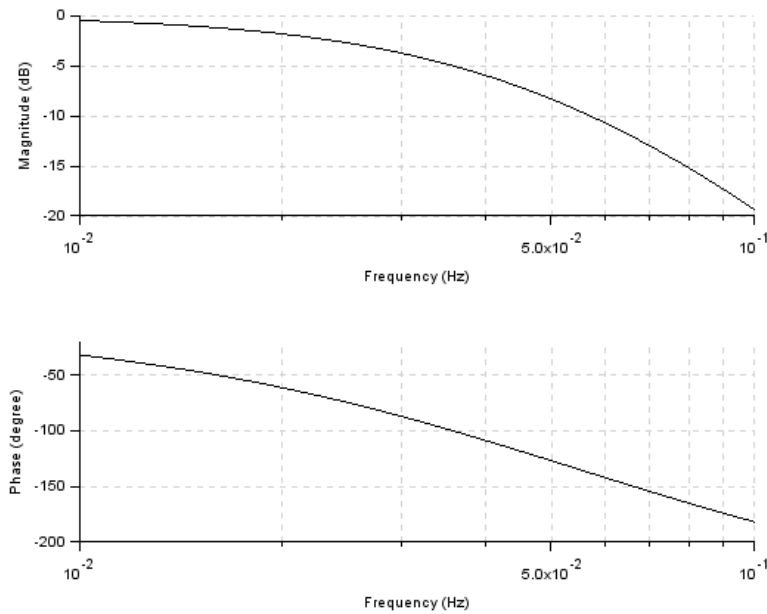


Figure 4.11: Bode plot of $F(s) = \frac{1}{(2s+1)(3s+1)(4s+1)}$

To obtain those values thanks to the Routh array we will first compute the denominator of the closed-loop system where a gain K_{cr} is inserted:

$$\frac{K_{cr}F(s)}{1 + K_{cr}F(s)} = \frac{K_{cr}}{24s^3 + 26s^2 + 9s + 1 + K_{cr}} \quad (4.76)$$

The Routh array has been obtained in (4.70). The value of K_{cr} such that the coefficient in the row s^1 is equal to zero is obtained by solving:

$$\frac{210 - 24K_{cr}}{26} = 0 \Rightarrow K_{cr} = \frac{210}{24} = 8.75 \quad (4.77)$$

The gain margin is then obtained as $Gm = K_{cr} = 8.75$.

The phase crossover frequency ω_π is obtained by setting to zero the polynomial extracted from the row s^2 and by replacing s by $j\omega_\pi$. We get:

$$\begin{aligned} 26s^2 + (1 + K_{cr})|_{s=j\omega_\pi} &= -26\omega_\pi^2 + 1 + K_{cr} = 0 \\ \Rightarrow \omega_\pi &= \sqrt{\frac{1+K_{cr}}{26}} = \sqrt{\frac{9.75}{26}} \text{ rad/sec} \end{aligned} \quad (4.78)$$

■

4.8 Root locus

The root locus technique⁶ has been developed in 1948 by Walter R. Evans (1920-1999). This is a graphical method for sketching in the s -plane the locus of roots of the following polynomial when parameter K varies to 0 to infinity:

$$D(s) + KN(s) \quad (4.79)$$

Usually polynomial $D(s) + KN(s)$ represents the denominator of the closed-loop transfer function. The root locus technique enables to sketch in the s -plane the poles of the closed-loop transfer function shown in Figure 4.6 or in Figure 4.7. Indeed in both configurations the denominator of the closed-loop transfer function is the same.

It is worth noticing that the roots of $D(s) + KN(s)$ are also the roots of $1 + K \frac{N(s)}{D(s)}$:

$$D(s) + KN(s) = 0 \Leftrightarrow 1 + K \frac{N(s)}{D(s)} = 0 \Leftrightarrow L(s) := KF(s) = -1 \quad (4.80)$$

Without loss of generality let's define transfer function $F(s)$ as follows:

$$F(s) = \frac{N(s)}{D(s)} = a \frac{\prod_{j=1}^{m \leq n} (s - z_j)}{\prod_{i=1}^n (s - p_i)} \quad (4.81)$$

Transfer function $L(s) = KF(s)$ is called the loop transfer function. In the SISO case the numerator of the loop transfer function $L(s)$ is scalar as well as its denominator.

⁶Walter R. Evans, Graphical Analysis of Control Systems, Transactions of the American Institute of Electrical Engineers, vol. 67, pp. 547 - 551, 1948

Equation $L(s) = -1$ can be equivalently split into two equations:

$$\begin{cases} |L(s)| = 1 \\ \arg(L(s)) = (2k + 1)\pi, \quad k = 0, \pm 1, \dots \end{cases} \quad (4.82)$$

The magnitude condition can always be satisfied by a suitable choice of K . On the other hand the phase condition does not depend on the value of K but only on the sign of K . Thus we have to find all the points in the s -plane that satisfy the phase condition. When scalar gain K varies from zero to infinity (i.e. K is positive), the root locus technique is based on the following rules:

- The root locus is symmetrical with respect to the horizontal real axis (because roots are either real or complex conjugate);
- The number of branches is equal to the number of poles of the loop transfer function. Thus the root locus has n branches;
- The root locus starts at the n poles of the loop transfer function;
- The root locus ends at the zeros of the loop transfer function. Thus m branches of the root locus end on the m zeros of $F(s)$ and there are $(n - m)$ asymptotic branches;
- Assuming that coefficient a in $F(s)$ is positive, a point s^* on the real axis belongs to the root locus as soon as there is an odd number of poles and zeros on its right. Conversely assuming that coefficient a in $F(s)$ is negative, a point s^* on the real axis belongs to the root locus as soon as there is an even number of poles and zeros on its right. Be careful to take into account the multiplicity of poles and zeros in the counting process;
- The $(n - m)$ asymptotic branches of the root locus which diverge to ∞ are asymptotes.
 - The angle δ_k of each asymptote with the real axis is defined by:

$$\delta_k = \frac{\pi + \arg(a) + 2k\pi}{n - m} \quad \forall k = 0, \dots, n - m - 1 \quad (4.83)$$

- Denoting by p_i the n poles of the loop transfer function (that are the roots of $D(s)$) and by z_j the m zeros of the loop transfer function (that are the roots of $N(s)$), the asymptotes intersect the real axis at a point (called pivot or centroid) given by:

$$\sigma = \frac{\sum_{i=1}^n p_i - \sum_{j=1}^{m \leq n} z_j}{n - m} \quad (4.84)$$

- The breakaway / break-in points are located on the real axis and always have a vertical tangent. They are located at the roots s_b of the following equation as soon as there is an odd (if coefficient a in $F(s)$ is positive) or even (if coefficient a in $F(s)$ is negative) number of poles and zeros on its

right (Be careful to take into account the multiplicity of poles and zeros in the counting process):

$$\begin{aligned} \frac{d}{ds} \left(\frac{1}{F(s)} \right)_{s=s_b} &= \frac{d}{ds} \left(\frac{D(s)}{N(s)} \right)_{s=s_b} = 0 \\ \Leftrightarrow D'(s_b)N(s_b) - D(s_b)N'(s_b) &= 0 \end{aligned} \quad (4.85)$$

Indeed from the fact that breakaway / break-in points have always a vertical tangent we can write:

$$1 + K F(s) = 1 + K \frac{N(s)}{D(s)} = 0 \Rightarrow \frac{dK}{dp} = - \frac{D'(s)N(s) - D(s)N'(s)}{N^2(s)} = 0 \quad (4.86)$$

From this relation we get (4.85).

- On the imaginary axis we have $s = j\omega$. Thus the value of the (positive) critical gain beyond which the closed-loop system becomes unstable is the value of K ($K \geq 0$) such that the root locus of $F(s)$ crosses the imaginary axis. In that situation at least one pole of the closed-loop system is purely imaginary. As far as $D(s) + KN(s)$ represents the denominator of the closed-loop transfer function the critical gain can be obtained by replacing s by $j\omega$ and by solving:

$$1 + K F(j\omega) = 0 \Leftrightarrow D(j\omega) + KN(j\omega) = 0 \quad (4.87)$$

The previous equation is then split into its real and imaginary part and provides a system of 2 equations which lead to the value of the critical gain and the oscillation frequency at the critical gain. It is worth noticing that the Routh criterion can be used for the same purpose.

- Note that if the degree of polynomial $D(s)$ is greater than or equal to the degree of polynomial $N(s)$ plus 2, meaning that the relative degree of transfer function $F(s)$ is greater than or equal to 2 ($n - m \geq 2$), then the sum of the poles of the feedback system is independent of the value of parameter K , and therefore is equal to the sum of the poles of the open loop system when $K = 0$. This property is known as the *centroid theorem*. To get this result, we have simply to expand $D(s) + KN(s)$ taking into account $n - m \geq 2$:

$$\begin{aligned} F(s) = \frac{N(s)}{D(s)} &= a \frac{\prod_{j=1}^{m \leq n-2} (s - z_j)}{\prod_{i=1}^n (s - p_i)} \\ \Rightarrow D(s) + KN(s) &= \prod_{i=1}^n (s - p_i) + K a \prod_{j=1}^{m \leq n-2} (s - z_j) \\ &= s^n - (r_1 + r_2 + \dots + r_n) s^{n-1} + \dots \end{aligned} \quad (4.88)$$

Assuming that $n - m \geq 2$, the coefficient of the term s^{n-1} in polynomial $D(s) + KN(s)$ does not depend on parameter K . Because this coefficient is obtained has the opposite of the sum $r_1 + r_2 + \dots + r_n$ of the roots of polynomial $D(s) + KN(s)$, we conclude the sum of the poles of the feedback system is independent of the value of parameter K .

Example 4.12. *Let's consider the following transfer function:*

$$F(s) = \frac{s + 4}{s(s + 2)(s + 3)} \quad (4.89)$$

We wish to investigate where are located the poles of the closed-loop system represented either in Figure 4.6 or in Figure 4.7 when static gain K varies from 0 to ∞ .

Let's apply the previous rules to sketch the root locus:

- The loop transfer function $F(s)$ has $n = 3$ poles and $m = 1$ zero;
- The number of branches is equal to the number of poles of the loop transfer function. Thus the root locus has $n = 3$ branches;
- The root locus starts at the $n = 3$ poles of the loop transfer function, which are $(0, -2, -3)$. This leads to sketch 1 in Figure 4.1.
- The root locus ends at the zeros of the loop transfer function. Thus $m = 1$ branch of the root locus end on the $m = 1$ zero of $F(s)$, which is (-4) . This leads to sketch 2 in Figure 4.1.
- Transfer function $F(s)$ reads:

$$F(s) = \frac{s + 4}{s(s + 2)(s + 3)} = 1 \times \frac{s + 4}{s(s + 2)(s + 3)} \quad (4.90)$$

Thus coefficient $a = 1$ of $F(s)$ is positive. Then a point s^* on the real axis belongs to the root locus as soon as there is an odd number of poles and zeros on its right (be careful to take into account the multiplicity of poles and zeros). This leads to sketch 3 in Figure 4.1.

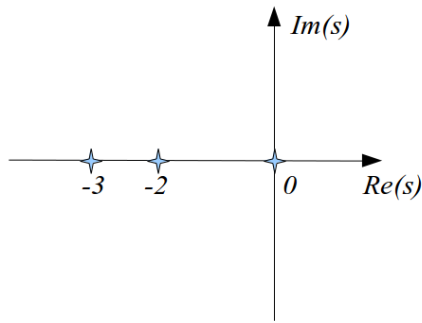
- There are $(n - m) = 2$ asymptotic branches. The $(n - m) = 2$ asymptotic branches of the root locus which diverge to ∞ are asymptotes.
- The angle δ_k of each asymptote with the real axis is defined by:

$$\delta_k = \frac{\pi + 2k\pi}{n - m} = \frac{\pi + 2k\pi}{2} \quad \forall k = 0, 1 \quad (4.91)$$

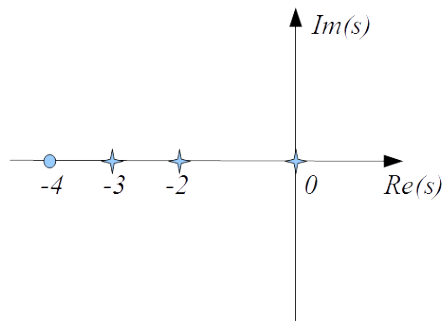
- Denoting by p_i the n poles of the loop transfer function (that are the roots of $D(s)$) and by z_j the m zeros of the loop transfer function (that are the roots of $N(s)$), the asymptotes intersect the real axis at a point (called pivot or centroid) given by:

$$\sigma = \frac{\sum_{i=1}^n p_i - \sum_{j=1}^{m \leq n} z_j}{n - m} = \frac{0 + (-2) + (-3) - (-4)}{2} = -0.5 \quad (4.92)$$

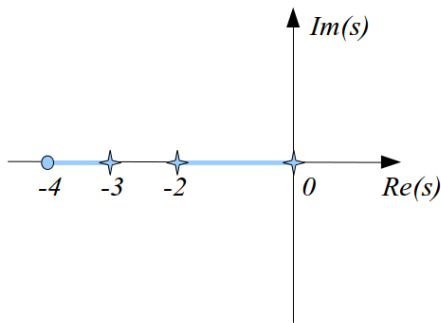
This leads to sketch 4 in Figure 4.1.



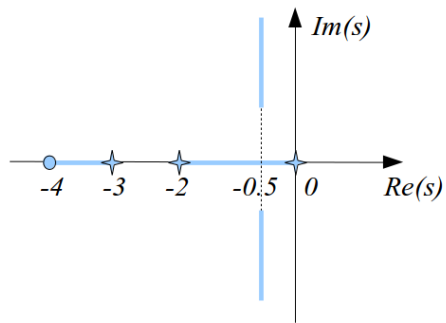
Sketch 1



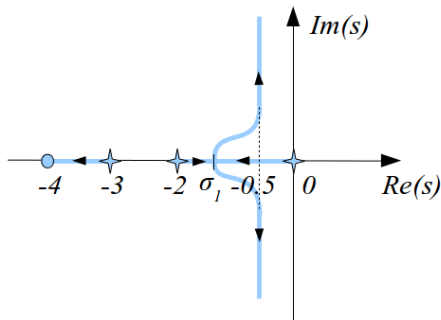
Sketch 2



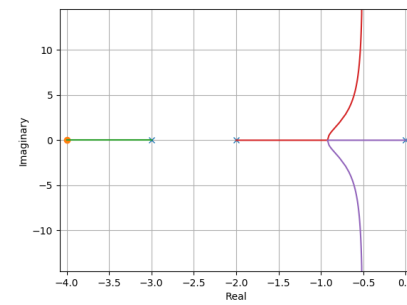
Sketch 3



Sketch 4



Sketch 5



Actual root locus (Python)

Table 4.1: Sketches of the root locus of $F(s)$

- The breakaway / break-in points are located at the roots s_b of the following equation as soon as there is an odd (because coefficient $a = 1$ of $F(s)$ is positive) number of poles and zeros on its right (be careful to take into account the multiplicity of poles and zeros);

$$\frac{d}{ds} \left(\frac{1}{F(s)} \right)_{s=s_b} = 0 \quad (4.93)$$

We get:

$$\begin{aligned} \frac{d}{ds} \left(\frac{s(s+2)(s+3)}{s+4} \right) &= \frac{d}{ds} \left(\frac{s^3+5s^2+6s}{s+4} \right) \\ &= \frac{(3s^2+10s+6)(s+4) - (s^3+5s^2+6s)}{(s+4)^2} \\ &= \frac{2s^3+17s^2+40s+24}{(s+4)^2} \\ \Rightarrow \frac{d}{ds} \left(\frac{1}{F(s)} \right)_{s=s_b} = 0 &\Rightarrow 2s^3 + 17s^2 + 40s + 24 = 0 \quad (4.94) \\ &\Rightarrow \begin{cases} s_{b1} \approx -0.92 \\ s_{b2} \approx -2.63 \\ s_{b3} \approx -4.95 \end{cases} \end{aligned}$$

Among the three potential breakaway / break-in points s_{bi} only $s_{b1} \approx -0.92$ is such that there is an odd (because coefficient $a = 1$ of $F(s)$ is positive) number of poles and zeros on its right. This leads to sketch 5 in Figure 4.1 where the arrows indicate how move the roots as gain K increases.

The actual root locus is represented in Figure 4.1. It has been obtained thanks to the following commands on Python:

```
#!/usr/bin/python
import control
import matplotlib.pyplot as plt
import numpy as np
F = control.tf([1, 4], np.polymul(np.polymul([1, 0], [1, 2]), [1, 3]))
control.rlocus(F)
plt.grid()
plt.show()
```

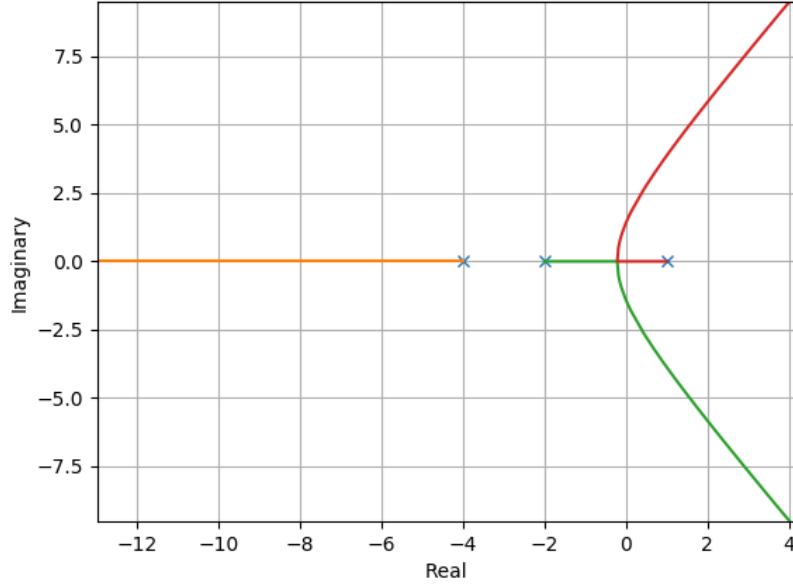
- Because $n - m \geq 2$, the centroid theorem is applicable and indicates that the sum of the poles of the feedback system is equal to the sum of the poles of the open loop system, that is -5 .

■

Example 4.13. Let's consider the following transfer function:

$$F(s) = \frac{1}{(s-1)(s+2)(s+4)} \quad (4.95)$$

The breakaway / break-in points are located at the roots s_b of the following equation as soon as there is an odd (because coefficient $a = 1$ of $F(s)$ is

Figure 4.12: Root locus of $F(s)$

positive) number of poles and zeros on its right (be careful to take into account the multiplicity of poles and zeros);

$$\begin{aligned}
 \frac{d}{ds} \left(\frac{1}{F(s)} \right)_{s=s_b} &= 0 \\
 \Rightarrow \frac{d}{ds} ((s-1)(s+2)(s+4)) &= 0 \\
 \Rightarrow \frac{d}{ds} (s^3 + 5s^2 + 2s - 8) &= 0 \\
 \Rightarrow 3s^2 + 10s + 2 &= 0 \\
 \Rightarrow \begin{cases} s_{b1} \approx -0.21 \\ s_{b2} \approx -3.12 \end{cases} &
 \end{aligned} \tag{4.96}$$

Among the two potential breakaway / break-in points s_{bi} only $s_{b1} \approx -0.21$ is such that there is an odd (because coefficient $a = 1$ of $F(s)$ is positive) number of poles and zeros on its right.

The root locus is represented in Figure 4.12.

It is worth noticing that the values of K such that the root locus of $F(s)$ intercepts the imaginary axis can be computed thanks to the Routh criterion. Indeed the denominator of the closed loop reads:

$$\begin{aligned}
 1 + K F(s) &= 1 + K \frac{1}{(s-1)(s+2)(s+4)} \\
 &= 1 + \frac{K}{s^3 + 5s^2 + 2s - 8} \\
 &= \frac{s^3 + 5s^2 + 2s - 8 + K}{s^3 + 5s^2 + 2s - 8}
 \end{aligned} \tag{4.97}$$

Applying the Routh criterion on polynomial $D(s) = s^3 + 5s^2 + 2s - 8 + K$ leads to the following Routh array:

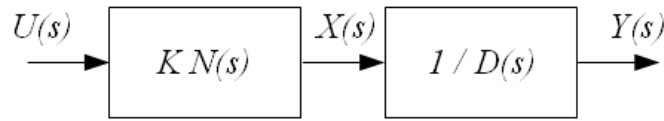


Figure 4.13: Effect of zeros of the system response

s^3	1	2	(4.98)
s^2	5	$K - 8$	
s^1	$\frac{18-K}{5}$		
s^0	$K - 8$		

There is no sign change in the first column of the Routh array as soon as the following inequality holds:

$$\begin{cases} 18 - K > 0 \\ K - 8 > 0 \end{cases} \Leftrightarrow 8 < K < 18 \quad (4.99)$$

As a consequence the values of K such that the root locus of $F(s)$ intercepts the imaginary axis are $K = 8$ (for which it can be checked that the roots of $D(s)$ are $0, -0.44$ and -4.56) and $K = 18$ (for which it can be checked that the roots of $D(s)$ are -5 and $\pm 1.414j$).

■

4.9 Effect of zeros on the system response

Consider a system with the following transfer function:

$$F(s) = K \frac{N(s)}{D(s)} = K \frac{s^m + b_{m-1}s^{m-1} + \dots + b_1s + b_0}{s^n + a_{n-1}s^{n-1} + \dots + a_1s + a_0} \quad (4.100)$$

Numerator $N(s)$ does not affect the homogeneous response of the system. Now consider Figure 4.13 where the all-pole system $1/D(s)$ is excited by signal $x(t)$:

$$x(t) = K \sum_{k=0}^m b_k \frac{d^k}{dt^k} u(t) \quad (4.101)$$

As far as $x(t)$ is the superposition of the derivatives of $u(t)$ we conclude that the zeros of the system shape the manner on how the poles of the system are excited.

4.10 Step response of non-minimum phase transfer functions

We consider a *stable* linear time invariant single-input single-output system whose transfer function $F(s)$ is proper (meaning $F(\infty) < \infty$) with finite static gain (meaning $F(0) < \infty$). If $F(s)$ has at least one real strictly positive zero, then $F(s)$ is a non-minimum phase transfer function and its step response exhibits *zero crossings*. Indeed the Laplace transform $Y_s(s)$ of the unit step response $y_s(t)$ reads:

$$Y_s(s) = F(s) \frac{1}{s} \quad (4.102)$$

Let z be a real strictly positive zero of $F(s)$. Setting $s = z$ yields $Y_s(s)|_{s=z} = F(z) \frac{1}{z}$. Since $F(z) = 0$ it follows that $Y_s(s)|_{s=z} = 0$. Coming back in the time domain we get:

$$\int_0^{\infty} y_s(t) e^{-st} dt \Big|_{s=z} = \int_0^{\infty} y_s(t) e^{-zt} dt = 0 \quad (4.103)$$

Since z is a real strictly positive zero, e^{-zt} is positive on $[0, \infty)$. It follows that $y_s(t)$ must cross zero on $[0, \infty)$ in order to achieve a null value for the integral. Note that (4.103) depends on z but does not depend on either the poles or the remaining zeros of $F(s)$.

It can be shown that undershoot in the step response of a strictly proper stable transfer function $F(s)$ occurs if and only if the plant has an odd number of real right-half plane zeros⁷. To get this result, the key relations are the following, where a denotes a constant, z_i the zeros of $F(s)$ and λ_j its poles :

$$\left\{ \begin{array}{l} F(s) = a \frac{\prod_{i=1}^m \left(1 - \frac{s}{z_i}\right)}{\prod_{j=1}^n \left(1 - \frac{s}{\lambda_j}\right)} \\ \left. \frac{d^r y(t)}{dt^r} \right|_{t=0} = \lim_{s \rightarrow \infty} s^r F(s) \end{array} \right. \quad (4.104)$$

More generally, the following properties hold for *stable* non-minimum phase transfer functions⁸:

- If $F(s) - F(0)$ has at least one strictly positive real zero, then the step response of $F(s)$ exhibits *overshoot*.
- If the transfer function $F(s) - F(\infty)$ has an odd number of strictly positive real zeros, then the step response exhibits initial *undershoot*.

Example 4.14. We consider the following stable transfer functions $F_1(s)$ and $F_2(s)$:

$$\left\{ \begin{array}{l} F_1(s) = \frac{1-s}{(s+1)^2} \\ F_2(s) = \frac{(0.5-s)(2-s)}{(s+1)^3} \end{array} \right. \quad (4.105)$$

⁷On Undershoot and Nonminimum Phase Zeros, M. Vidyasagar, IEEE Transactions on Automatic Control, vol. 31, pp. 440 - 440, Issue: 5, May 1986

⁸Nonminimum-phase zeros - much to do about nothing - classical control - revisited part ii, J.B. Hoagg and D.S. Bernstein, IEEE Control Systems, vol. 27, no. 3, pp. 45-57, June 2007

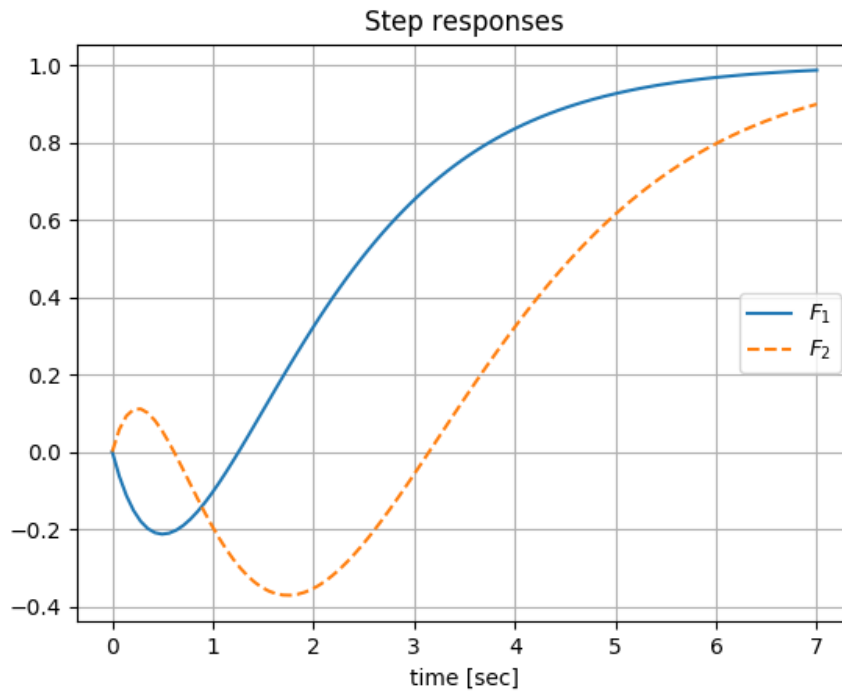


Figure 4.14: Step response of $F_1(s)$ and $F_2(s)$

- It is clear that $F_1(s)$ has one strictly positive real zero, $z_1 = 1$, whereas $F_2(s)$ has two strictly positive real zeros, $z_1 = 0.5$ and $z_2 = 2$. Thus step response of $F_1(s)$ and $F_2(s)$ exhibits zero crossings.
- $F_1(s) - F_1(0)$ and $F_2(s) - F_2(0)$ do not have strictly positive real zero. Thus both step responses does not exhibit overshoot.
- The zero of $F_1(s) - F_1(\infty)$ is 1, that is an odd number of strictly positive real zeros. Thus the step response of $F_1(s)$ exhibits initial undershoot.
- The zeros of $F_2(s) - F_2(\infty)$ are 2 and 0.5, that is an even number of strictly positive real zeros. Thus the step response of $F_1(s)$ does not exhibit initial undershoot.

Those facts can be seen on the step response of $F_1(s)$ and $F_2(s)$ shown in Figure 4.14. ■

4.11 Strong stabilizability and parity interlacing property

A plant, defined by a transfer matrix $\mathbf{F}(s)$, is said *strongly stabilizable* if there exists a stable stabilizing controller $\mathbf{C}(s)$ of $\mathbf{F}(s)$ or, in other words, if $\mathbf{C}(s)$ internally stabilizes $\mathbf{F}(s)$.

A necessary and sufficient condition for the existence of a strongly stabilizing controller is the parity interlacing property. The parity interlacing property states that *a dynamical plant $\mathbf{F}(s)$ is strongly stabilizable if and only if between any pair of zeros with positive real part (taking into account also the zeros at infinity) there is an even number of poles, the latter counted according to their multiplicities*⁹.

⁹Youla D. C., J. J. Bongiorno, Jr., and C. N. Lu, Single-loop feedback stabilization of linear multivariable plants, *Automatica*, 10 (1974), pp. 159-173.

Chapter 5

Basic feedback control design

5.1 Chapter overview

In this chapter we will see how to tune two types of well known controllers:

- Lead-lag controllers, which can be tuned in order to comply with specifications in terms of precision and phase margin;
- PID controller, which can be tuned in order to comply with specifications in terms of transient response specifications or dominant poles placement.

Furthermore prefiltering as well as algebraic controller design and anti-windup will also be developed.

5.2 Lead-lag controller

5.2.1 Steady state error

In Figure 5.1 we have specialized the corrector $C(s)$ of Figure 4.2 to be a constant value K . This kind of feedback control is called proportional control.

Usually reference input $r(t)$ is the signal we would want to obtain at the output and consequently such a control loop is currently designed such that the output $y(t)$ follows the reference input $r(t)$. In the case of perfect feedback control the error signal $\epsilon(t)$ is then zero at every instant. But in the case of real

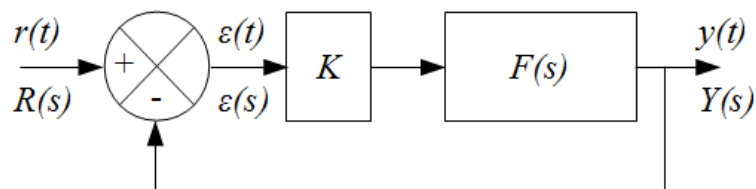


Figure 5.1: Proportional control loop

feedback control this difference is not zero at every instant mainly because of time constants. The steady-state error $\epsilon(\infty)$ represents the tracking error for a particular input command when time t tends to infinity:

$$\epsilon(\infty) = \lim_{t \rightarrow \infty} \epsilon(t) \quad (5.1)$$

This section presents some results on the value of the steady-state error $\epsilon(\infty)$ for different shape of the signal input $r(t)$.

Rather than computing the inverse Laplace transform of $\epsilon(s)$ back into the time domain and evaluating it as time t approaches infinity we will use the final value theorem:

$$\epsilon(\infty) = \lim_{s \rightarrow 0} s\epsilon(s) \quad (5.2)$$

It is worth noticing that the computation of the steady-state error thanks to the final value theorem is only meaningful for stable closed-loop system. Consequently the stability of the closed-loop system shall be checked when performing steady-state error analysis.

From Figure 5.1 the Laplace transform of the error $\epsilon(s)$ reads:

$$\epsilon(s) = R(s) - KF(s)\epsilon(s) \Leftrightarrow \epsilon(s) = \frac{1}{1 + KF(s)}R(s) \quad (5.3)$$

The steady-state error will depend on the input command $R(s)$ and on the loop transfer function $KF(s)$. In the following we will consider without loss of generality that the plant transfer function $F(s)$ is of the form:

$$F(s) = \frac{A}{s^\alpha} \cdot \frac{1 + b_1s + \dots + b_ms^m}{1 + a_1s + \dots + a_ns^n} \quad (5.4)$$

It is worth noticing that the coefficients which are in factor to $s^0 = 1$ at the numerator and denominator of transfer function $F(s)$ are equal to 1.

Integer α represents the number of integrations in $F(s)$, but also in the open-loop transfer function $KF(s)$. By definition the type number of a system is the number of integrations in its open-loop transfer function. Thus if $\alpha = 0$ then the system is said to be of type 0, if $\alpha = 1$ then type 1 and so on.

We first examine the steady-state error to a unit step input:

$$r(t) = \Gamma(t) \Rightarrow R(s) = \frac{1}{s} \quad (5.5)$$

When the input reference signal is a step the steady-state error is called the position error and is found to be:

$$\begin{aligned} \epsilon_p &= \lim_{s \rightarrow 0} s\epsilon(s) = \lim_{s \rightarrow 0} s \frac{1}{1 + KF(s)} \frac{1}{s} \\ &= \lim_{s \rightarrow 0} \frac{1}{1 + \frac{KA}{s^\alpha} \frac{1 + b_1s + \dots + b_ms^m}{1 + a_1s + \dots + a_ns^n}} \end{aligned} \quad (5.6)$$

The value of the preceding limit depends on the system type:

- If $\alpha = 0$ (system of type 0) then

$$\epsilon_p = \frac{1}{1 + KA} \quad (5.7)$$

– If $\alpha > 0$ then

$$\epsilon_p = 0 \quad (5.8)$$

Now we examine the steady-state error to a unit-ramp input:

$$r(t) = t \quad \forall t \geq 0 \Rightarrow R(s) = \frac{1}{s^2} \quad (5.9)$$

When the input reference signal is a unit-ramp the steady-state error is called the velocity error and is found to be:

$$\begin{aligned} \epsilon_v &= \lim_{s \rightarrow 0} s\epsilon(s) = \lim_{s \rightarrow 0} \frac{1}{s} \frac{1}{1 + \frac{A}{s^\alpha} \frac{1+b_1s+\dots+b_ms^m}{1+a_1s+\dots+a_ns^n}} \\ &= \lim_{s \rightarrow 0} \frac{1}{s + \frac{KA}{s^{\alpha-1}} \frac{1+b_1s+\dots+b_ms^m}{1+a_1s+\dots+a_ns^n}} \end{aligned} \quad (5.10)$$

The value of the preceding limit depends on the system type:

– If $\alpha = 0$ (system of type 0) then

$$\epsilon_v = \infty \quad (5.11)$$

– If $\alpha = 1$ (system of type 1) then

$$\epsilon_v = \frac{1}{KA} \quad (5.12)$$

– If $\alpha > 1$ then

$$\epsilon_v = 0 \quad (5.13)$$

Similarly when the input reference signal is a parabola the steady-state error is called the acceleration error and is found to be:

$$r(t) = t^2 \quad \forall t \geq 0 \Rightarrow R(s) = \frac{2}{s^3} \Rightarrow \epsilon_a = \lim_{s \rightarrow 0} \frac{2}{s^2 + \frac{KA}{s^{\alpha-2}} \frac{1+b_1s+\dots+b_ms^m}{1+a_1s+\dots+a_ns^n}} \quad (5.14)$$

As in the previous cases the value of the steady-state error depends on the system type:

– If $\alpha < 2$ then

$$\epsilon_a = \infty \quad (5.15)$$

– If $\alpha = 2$ (system of type 2) then

$$\epsilon_a = \frac{2}{KA} \quad (5.16)$$

– If $\alpha > 2$ then

$$\epsilon_a = 0 \quad (5.17)$$

System's type	$r(t) = \Gamma(t)$	$r(t) = t$	$r(t) = t^2$
0 ($\alpha = 0$)	$\frac{1}{1 + KA}$	∞	∞
1 ($\alpha = 1$)	0	$\frac{1}{KA}$	∞
2 ($\alpha = 2$)	0	0	$\frac{2}{KA}$

Table 5.1: Steady state errors

A summary of the steady-state error according to the system type and the input reference command $r(t)$ is provided in Table 5.1.

The steady state errors are classified in accordance with the number of integration in the open-loop transfer function. As $s \rightarrow 0$ the term which dominates in determining the steady-state error of the *stable* closed-loop system is $s^{\alpha-N}$. For a system of type α the steady-state error of the closed-loop system is constant when the input command is of the form t^α ; if the input command is of the form t^N where $N < \alpha$ then the steady-state error is 0 whereas when $N > \alpha$ the steady-state error tends to infinity.

It is worth noticing that to make the steady-state error small, that is to increase the precision of the closed-loop system, the term KA (which is the static gain for a type 0 system) in the open-loop transfer function shall be chosen sufficiently high. But on the other hand the higher the value of proportional control K is, the closer the closed-loop system will be from instability. Indeed we can see in Figure 4.9 that increasing the value of K will displace the $-1/K + j0$ critical point to the right; hence the closed-loop system is approaching to the verge of instability. This problem is usually known as the stability-precision dilemma.

Example 5.1. Consider a plant with the following transfer function:

$$F(s) = \frac{1}{(2s + 1)(3s + 1)(4s + 1)} \quad (5.18)$$

We wish to control this system thanks to a proportional control as in Figure 5.1. As far as this system is of class 0 the proportional gain K will only be used to set the position error. Assume that system specification requires the position error to be 1%. This leads to the following value of proportional gain K :

$$1\% = \frac{1}{100} = \frac{1}{1 + KA} \quad (5.19)$$

Here the transfer function $F(s)$ is such that $A = 1$ (cf. (5.4)) and consequently $K = 99$. Unfortunately we have seen in the previous section that the closed-loop system is stable when $K < 8.75$. As a consequence the system is

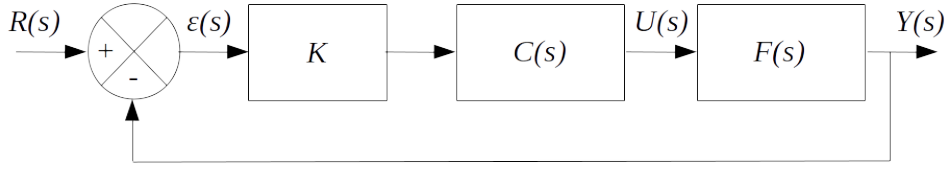


Figure 5.2: Lead-lag control

unstable with such a value of proportional gain K . This is an example of stability-precision dilemma. ■

When system specifications cannot be satisfied with the proportional control loop in Figure 5.1, other control loop schemes shall be envisioned. This is the purpose of the next chapter.

5.2.2 Controller presentation

Let consider Figure 5.2 where $C(s)$ is a lead-lag controller and K a proportional gain.

A lead-lag controller has the following transfer function:

$$C(s) = \frac{1 + \tau s}{1 + \alpha \tau s} \quad (5.20)$$

Where α and τ are positive constants. Parameter α enables to define the controller type:

- Lead controller if $0 < \alpha < 1$
- Lag controller if $\alpha > 1$

For a lead controller the pole $\frac{-1}{\alpha\tau}$ is situated on the left of the zero $\frac{-1}{\tau}$ and vice versa for a lag controller.

Let $\epsilon(\infty)$ be a specified steady-state error. We have seen in the previous chapter that the Laplace transform of the system error $\epsilon(s)$ reads:

$$\epsilon(s) = R(s) - KC(s)F(s)\epsilon(s) \Leftrightarrow \epsilon(s) = \frac{1}{1 + KC(s)F(s)}R(s) \quad (5.21)$$

Therefore applying the final value theorem the steady-state error $\epsilon(\infty)$ reads:

$$\epsilon(\infty) = \lim_{t \rightarrow \infty} \epsilon(t) = \lim_{s \rightarrow 0} s\epsilon(s) = \lim_{s \rightarrow 0} \frac{s}{1 + KC(s)F(s)}R(s) \quad (5.22)$$

As far $C(0) = 1 \forall \alpha, \tau$ we get:

$$C(0) = 1 \forall \alpha, \tau \Rightarrow \epsilon(\infty) = \lim_{s \rightarrow 0} \frac{s}{1 + KF(s)}R(s) \quad (5.23)$$

Consequently a lead-lag controller allows to modify the steady-state error of the closed-loop system by selecting appropriately the static gain K ,

independently of parameters α and τ . In other words, steady state error is improved by increasing static gain K (assuming that the closed-loop remains stable).

Phase contribution $\Phi(\omega)$ and gain contribution $|C(\omega)|$ of lead-lag controller $C(s)$ at frequency ω are obtained as follows:

$$\begin{aligned} C(j\omega) &= \frac{1+j\tau\omega}{1+j\alpha\tau\omega} \\ &= \frac{(1+j\tau\omega)(1-j\alpha\tau\omega)}{1+(\alpha\tau\omega)^2} \\ &= \frac{1+\alpha(\omega\tau)^2+j\omega\tau(1-\alpha)}{1+(\alpha\tau\omega)^2} \\ &= |C(\omega)| e^{j\Phi(\omega)} \end{aligned} \quad (5.24)$$

Thus:

– Phase contribution $\Phi(\omega)$ at frequency ω reads:

$$\begin{cases} \cos(\Phi(\omega)) = \frac{1+\alpha(\omega\tau)^2}{\sqrt{(1+\alpha(\omega\tau)^2)^2+(\omega\tau(1-\alpha))^2}} \\ \sin(\Phi(\omega)) = \frac{\omega\tau(1-\alpha)}{\sqrt{(1+\alpha(\omega\tau)^2)^2+(\omega\tau(1-\alpha))^2}} \end{cases} \Rightarrow \tan(\Phi(\omega)) = \frac{\omega\tau(1-\alpha)}{1+\alpha(\omega\tau)^2} \quad (5.25)$$

– Gain contribution $|C(\omega)|$ at frequency ω reads:

$$|C(\omega)| = \sqrt{\frac{1+(\tau\omega)^2}{1+(\alpha\tau\omega)^2}} \quad (5.26)$$

A lead compensator adds positive phase to the system over the frequency range $\frac{1}{\tau}$ to $\frac{1}{\alpha\tau}$ whereas a lag compensator adds negative phase to the system over the frequency range $\frac{1}{\alpha\tau}$ to $\frac{1}{\tau}$. The extremal phase shift Φ_m is achieved at frequency ω_m :

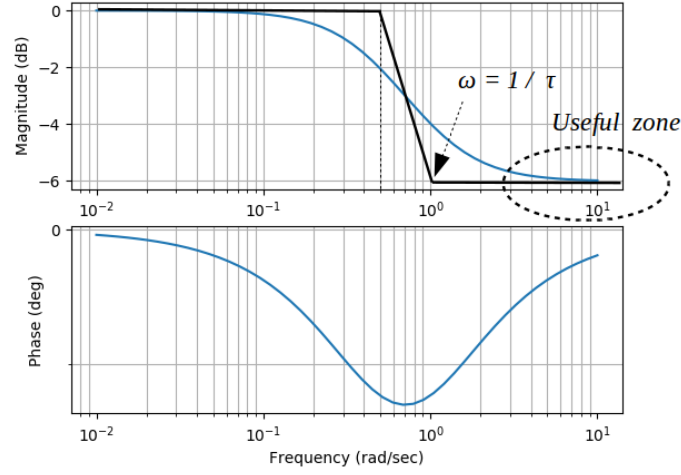
$$C(j\omega_m) = |C(\omega_m)| e^{j\Phi_m} \text{ where } \begin{cases} \omega_m = \frac{1}{\tau\sqrt{\alpha}} \\ |C(\omega_m)| = \frac{1}{\sqrt{\alpha}} \\ \sin(\Phi_m) = \frac{1-\alpha}{1+\alpha} \end{cases} \quad (5.27)$$

Let $g = |C(\omega)|^2$. From the preceding relations we get:

$$g = |C(\omega)|^2 = \frac{1+(\tau\omega)^2}{1+(\alpha\tau\omega)^2} \Rightarrow (\tau\omega)^2 = \frac{1-g}{g\alpha^2-1} \quad (5.28)$$

Let $p = \tan(\Phi(\omega))$. From the preceding relations it is clear that p and c are related as follows:

$$\begin{aligned} p = \tan(\Phi(\omega)) &\Rightarrow p^2 = \frac{(\omega\tau)^2(1-\alpha)^2}{(1+\alpha(\omega\tau)^2)^2} = \frac{\frac{1-g}{g\alpha^2-1}(1-\alpha)^2}{\left(1+\alpha\frac{1-g}{g\alpha^2-1}\right)^2} \\ &\Leftrightarrow p^2 \left(1 + \alpha\frac{1-g}{g\alpha^2-1}\right)^2 - \frac{1-g}{g\alpha^2-1}(1-\alpha)^2 = 0 \\ &\Leftrightarrow p^2 \left(\frac{g\alpha^2-1+\alpha(1-g)}{g\alpha^2-1}\right)^2 - \frac{1-g}{g\alpha^2-1}(1-\alpha)^2 = 0 \\ &\Leftrightarrow p^2 \left(\frac{g\alpha(\alpha-1)+\alpha-1}{g\alpha^2-1}\right)^2 - \frac{(1-g)(g\alpha^2-1)}{(g\alpha^2-1)^2}(1-\alpha)^2 = 0 \\ &\Leftrightarrow p^2 \left(\frac{g\alpha+1}{g\alpha^2-1}\right)^2 (\alpha-1)^2 - \frac{(1-g)(g\alpha^2-1)}{(g\alpha^2-1)^2}(1-\alpha)^2 = 0 \end{aligned} \quad (5.29)$$

Figure 5.3: Lag controller Bode plot ($\alpha = 2 > 1$ and $\tau = 1$)

Assuming that $\alpha \neq 1$ and $g\alpha^2 - 1 \neq 0$ we finally get:

$$\begin{aligned} p^2 (g\alpha + 1)^2 - (1 - g)(g\alpha^2 - 1) &= 0 \\ \Leftrightarrow \alpha^2 g(gp^2 + g - 1) + 2\alpha gp^2 + p^2 + 1 - g &= 0 \end{aligned} \quad (5.30)$$

Thus the values of parameters α and τ of a lead-lag controller which produce a gain g and a phase p at frequency ω are defined as follows:

$$\begin{cases} \tau = \frac{1}{\omega} \sqrt{\frac{1-g}{g\alpha^2-1}} \\ \alpha^2 g(gp^2 + g - 1) + 2\alpha gp^2 + p^2 + 1 - g = 0 \end{cases} \quad \text{where } \begin{cases} p = \tan(\Phi(\omega)) \\ g = |C(\omega)|^2 \end{cases} \quad (5.31)$$

- For a lag controller where $p < 0$ and $0 < g < 1$ it is easily verified that a sufficient condition for the solution of the quadratic equation in α to be positive is:

$$gp^2 + g - 1 < 0 \quad (5.32)$$

- Similarly for a lead controller where $p > 0$ and $g > 1$ it is easily verified that a sufficient condition for the solution of the quadratic equation in α to be positive is:

$$p^2 + 1 - g < 0 \quad (5.33)$$

5.2.3 Lag controller tuning: graphical method

Compensator design using a lag controller is based on the controller's attenuation at *high frequencies* as shown in Figure 5.3.

Assuming that steady state error and phase margin requirements are given, lag controller tuning can be done graphically as follows:

1. Using the final value theorem, determine gain $K > 0$ which satisfies the steady state error requirement;

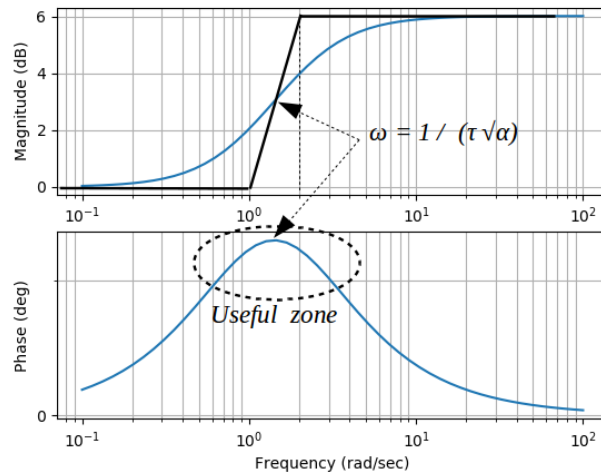


Figure 5.4: Lead controller Bode plot ($\alpha = 0.5 < 1$ and $\tau = 1$)

2. Draw the Bode plot of $K F(j\omega)$;
3. Find on the Bode phase plot of $K F(j\omega)$ the frequency ω_1 corresponding to the required phase margin M_Φ increased by 5° . The extra 5° is an overestimate (*forfait moustachu* in French) of the phase lag induced by the lag controller at *high frequency*;
4. Read on the Bode magnitude plot the value $20 \log_{10} |K F(j\omega_1)|$ at frequency ω_1 . This value will be shifted to 0 dB thanks to the attenuation of the lag controller at *high frequencies*:

$$-20 \log_{10} |K F(j\omega_1)| = -20 \log_{10} (\alpha) \Leftrightarrow \alpha = K F(j\omega_1) \quad (5.34)$$

5. Place the controller zero $1/\tau$ such that $1/\tau \ll \omega_1$, for example one decade to the left of the new gain crossover frequency ω_1 :

$$\frac{1}{\tau} \approx \frac{\omega_1}{10} \Leftrightarrow \tau \approx \frac{10}{\omega_1} \quad (5.35)$$

Finally draw the Bode plot of $K C(j\omega) F(j\omega)$ where $C(s) = \frac{1+\tau s}{1+\alpha\tau s}$ and check the value of the phase margin. If the achieved phase margin is not satisfactory increase the overestimate of the phase lag induced by the lag controller (try with 8° rather than 5° for example) and repeat the design process.

5.2.4 Lead controller tuning: graphical method

Compensator design using a lead controller is based on the controller's phase lead as shown in Figure 5.4.

Assuming that steady state error and phase margin requirements are given, lead controller tuning can be done graphically as follows:

1. Using the final value theorem, determine gain $K > 0$ which satisfies the steady state error requirement;
2. Draw the Bode plot of $K F(j\omega)$;
3. Find on the Bode phase plot of $K F(j\omega)$ the gain crossover frequency ω_2 such that $20 \log_{10} (K F(j\omega_2)) = 0 \text{ dB}$ and compute the phase difference $\Delta\phi$ between the actual and the required phase margin M_ϕ :

$$\Delta\phi = M_\phi - (\arg(K F(j\omega_2)) + 180^\circ) \quad (5.36)$$

4. Compute the maximum phase lead Φ_m of the lead controller as follows:

$$\Phi_m = \Delta\phi + 5^\circ \quad (5.37)$$

The increase of 5° is an overestimate (*forfait moustachu* in French) of the consequence of the amplification effect of the phase lead controller on the actual phase margin. Indeed, due to the amplification effect, the actual gain crossover frequency will be higher (that is shifted to the right on the Bode plot);

5. Compute parameter α of the lead controller as follows:

$$\alpha = \frac{1 - \sin(\Phi_m)}{1 + \sin(\Phi_m)} \quad (5.38)$$

6. Compute the lead controller amplification $1/\sqrt{\alpha}$ and read on the Bode magnitude plot the value of the frequency ω_3 such that:

$$20 \log_{10} (K F(j\omega_3)) = +10 \log_{10} (\alpha) \quad (5.39)$$

Frequency $\omega_3 > \omega_2$ will be the actual gain crossover frequency. Then compute parameter τ of the lead controller such that ω_3 corresponds to the frequency at which the maximum phase lead is achieved:

$$\omega_3 = \frac{1}{\tau\sqrt{\alpha}} \Leftrightarrow \tau = \frac{1}{\omega_3\sqrt{\alpha}} \quad (5.40)$$

Finally draw the Bode plot of $K C(j\omega) F(j\omega)$ where $C(s) = \frac{1+\tau s}{1+\alpha\tau s}$ and check the value of the phase margin. If the achieved phase margin is not satisfactory increase the overestimate of the effect of the lead controller on the phase margin (try with 8° rather than 5° for example) and repeat the design process.

5.2.5 Example

We consider a plant with the following transfer function:

$$F(s) = \frac{4}{s(s+2)} \quad (5.41)$$

We wish to control this plant such that the closed-loop match with the following specifications:

- Steady-state error of the closed-loop when the input is a unit-ramp input, that is the velocity error: $\epsilon_v = 0.05$
- Phase margin: $M_\phi = 50^\circ$

Error $\epsilon(t)$ between the reference signal $r(t)$ and the output $y(t)$ reads:

$$\epsilon(t) = r(t) - y(t) \quad (5.42)$$

The use of the final value theorem leads to the expression of velocity error:

$$\epsilon_v = \lim_{t \rightarrow \infty} \epsilon(t) = \lim_{s \rightarrow 0^+} s \epsilon(s) \quad (5.43)$$

Reading Figure 5.2 we get $Y(s) = \frac{KC(s)F(s)}{1+KC(s)F(s)}$. Furthermore as far as we focus on velocity error we have $r(t) = t \forall t \geq 0$. Thus $R(s) = \mathcal{L}[t] = \frac{1}{s^2}$ and:

$$\begin{aligned} \epsilon(s) &= R(s) - Y(s) \\ &= \frac{1}{1+KC(s)F(s)} R(s) \\ &= \frac{1}{1+KC(s)\frac{4}{s(s+2)}} \frac{1}{s^2} \end{aligned} \quad (5.44)$$

Using the fact that $C(0) = 1$ we finally get:

$$\begin{aligned} \epsilon_v &= \lim_{s \rightarrow 0^+} s \epsilon(s) \\ &= \lim_{s \rightarrow 0^+} s \frac{1}{1+KC(s)\frac{4}{s(s+2)}} \frac{1}{s^2} \\ &= \lim_{s \rightarrow 0^+} \frac{1}{1+KC(s)\frac{4}{s(s+2)}} \frac{1}{s} \\ &= \lim_{s \rightarrow 0^+} \frac{1}{s+KC(s)\frac{4}{s+2}} \\ &= \frac{1}{2K} \end{aligned} \quad (5.45)$$

Consequently we shall choose $K = 10$ to set ϵ_v to 0.05:

$$\epsilon_v = 0.05 = \frac{1}{2K} \Rightarrow K = 10 \quad (5.46)$$

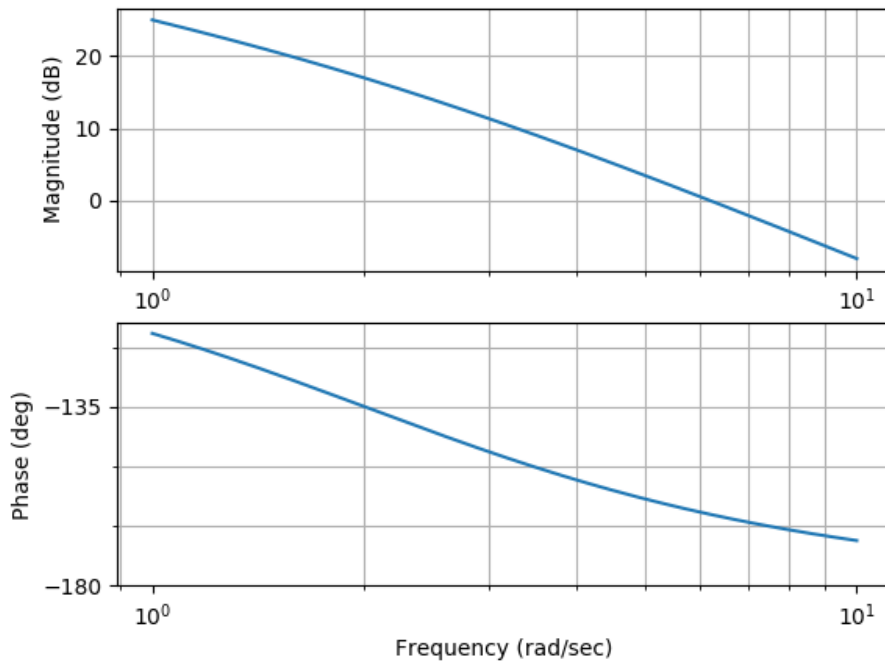
In order to apply the graphical method the Bode plot of $K F(j\omega)$ is sketched in Figure 5.5.

- For the lag controller we get:

$$\begin{cases} \omega_1 \approx 1.4 \text{ rad/sec} \\ 20 \log_{10} |K F(j\omega_1)| \approx 21 \text{ dB} \\ \alpha = 10^{21/20} \approx 11.2 \\ \tau \approx \frac{10}{\omega_1} \approx 7.1 \end{cases} \quad (5.47)$$

- For the lead controller we get:

$$\begin{cases} \omega_2 \approx 6 \text{ rad/sec} \\ \arg(K F(j\omega_2)) + 180^\circ \approx 17^\circ \\ \Phi_m = 50 - 17 + 5 = 38^\circ \\ \alpha = \frac{1 - \sin(\Phi_m)}{1 + \sin(\Phi_m)} \approx 0.24 \\ 10 \log_{10}(\alpha) \approx -6.2 \text{ dB} \\ \omega_3 \approx 9 \text{ rad/sec} \\ \tau = \frac{1}{\omega_3 \sqrt{\alpha}} \approx 0.23 \end{cases} \quad (5.48)$$

Figure 5.5: Bode plot of $K F(j\omega)$

We recall that the design process does not ensure the actual location of the dominant pole.

5.2.6 Lead-lag controller tuning: computational method

The method proposed by Phillips and Harbor² has the following steps:

- First get the required phase margin M_Φ as well as the settling time t_s from the design specifications;
- Find gain K such that the requirement on the steady-state error $\epsilon(\infty)$ of the closed-loop system is satisfied;
- Then choose the gain crossover frequency ω_c corresponding to the required phase margin M_Φ and the settling time t_s . Frequency ω_c is chosen by trial and error but shall satisfy:

$$\omega_c \geq \frac{8}{t_s \tan(M_\Phi)} \quad (5.49)$$

- Compute the phase angle φ of the controller $C(s)$ at frequency ω_c as:

$$\varphi := \arg(C(j\omega_c)) = -180^\circ + M_\Phi - \arg(KF(j\omega_c)) \quad (5.50)$$

It is worth noticing that angle φ shall be between -90° and $+90^\circ$: this corresponds to the minimum and maximum phase shift achievable by the controller.

- Set parameters α and τ of the lead-lag controller as follows:

$$C(s) = \frac{1 + \tau s}{1 + \alpha \tau s} \text{ where } \begin{cases} \tau = \frac{1 - K|F(j\omega_c)| \cos(\varphi)}{\omega_c K|F(j\omega_c)| \sin(\varphi)} \\ \alpha \tau = \frac{\cos(\varphi) - K|F(j\omega_c)|}{\omega_c \sin(\varphi)} \end{cases} \quad (5.51)$$

As in the PID tuning method, the preceding results may be obtained by equating the real part and the imaginary part of $K C(j\omega_c)$ to $\frac{\cos(\varphi)}{|F(j\omega_c)|}$ and $\frac{\sin(\varphi)}{|F(j\omega_c)|}$ respectively:

$$\begin{aligned} K C(j\omega_c) &= K \frac{1 + \alpha(\omega_c \tau)^2 + j\omega_c \tau(1 - \alpha)}{1 + (\alpha \tau \omega_c)^2} = \frac{\cos(\varphi)}{|F(j\omega_c)|} + j \frac{\sin(\varphi)}{|F(j\omega_c)|} \\ \Rightarrow \begin{cases} K \frac{1 + \alpha(\omega_c \tau)^2}{1 + (\alpha \tau \omega_c)^2} &= \frac{\cos(\varphi)}{|F(j\omega_c)|} \\ K \frac{\omega_c \tau(1 - \alpha)}{1 + (\alpha \tau \omega_c)^2} &= \frac{\sin(\varphi)}{|F(j\omega_c)|} \end{cases} \end{aligned} \quad (5.52)$$

If the achieved performances are not satisfactory then frequency ω_c has to be tuned by trial and error. However we recall that even then the closed-loop system may not achieved the required performance or can be unstable as far as the design process does not ensure the actual location of the dominant poles.

5.3 Transient response specifications

This section provides the links between time domain and frequency domain specifications. Indeed in one hand we may have time domain specifications, which are usually provided by the end-user of the system, whereas engineers rather prefer frequency domain specifications because they are more tractable in terms of transfer functions. Frequency domain specifications can be provided either in terms of dominant poles or in terms of phase margin M_Φ and corresponding gain crossover frequency ω_c .

Transient response specifications are usually based on second order underdamped model where the damping ratio m is in between 0 and 0.7 ($0 < m < 0.7$) and where the static gain K has been set to 1 reads:

$$G(s) = \frac{Y(s)}{R(s)} = \frac{\omega_0^2}{s^2 + 2m\omega_0 s + \omega_0^2} \quad (5.53)$$

It is worth noticing that model $G(s)$ should not be interpreted as the actual transfer function of the closed-loop system but as the specified reduced model of the actual closed-loop system.

Moreover transfer function $G(s)$ can be viewed as the closed-loop transfer function of the loop depicted in Figure 5.6 where the open-loop transfer function $F(s)$ is put within a unity feedback loop.

Open-loop transfer function $F(s)$ could be obtained thanks to the following identification:

$$\begin{cases} \frac{Y(s)}{R(s)} = \frac{\omega_0^2}{s^2 + 2m\omega_0 s + \omega_0^2} = \frac{1}{\frac{s^2 + 2m\omega_0 s}{\omega_0^2} + 1} \\ \frac{Y(s)}{R(s)} = \frac{F(s)}{1 + F(s)} = \frac{1}{\frac{1}{F(s)} + 1} \end{cases} \Rightarrow F(s) = \frac{\omega_0^2}{s(s + 2m\omega_0)} \quad (5.54)$$

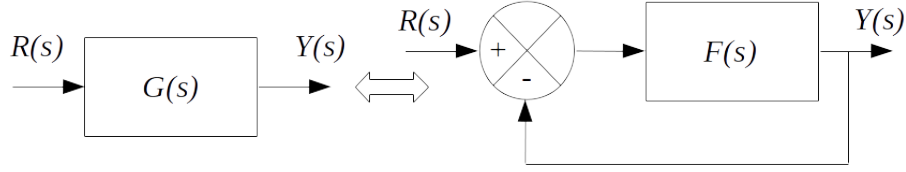


Figure 5.6: Unity feedback loop for transient response specifications

In the frequency domain we get:

$$F(j\omega) = \frac{\omega_0^2}{-\omega^2 + j2m\omega_0\omega} \quad (5.55)$$

The phase margin occurs at the gain crossover frequency ω_c which is the frequency such that $|F(j\omega_c)| = 1$. Thus frequency ω_c reads:

$$|F(j\omega_c)| = 1 \Rightarrow \frac{\omega_0^4}{\omega_c^4 + 4m^2\omega_0^2\omega_c^2} = 1 \Leftrightarrow \left(\frac{\omega_c}{\omega_0}\right)^4 + 4m^2\left(\frac{\omega_c}{\omega_0}\right)^2 - 1 = 0 \quad (5.56)$$

The roots of this quadratic equation are given by:

$$\left(\frac{\omega_c}{\omega_0}\right)^2 = -2m^2 \pm \sqrt{4m^4 + 1} \quad (5.57)$$

Since the root corresponding to the phase margin must be real positive we finally get:

$$\omega_c = \omega_0 \sqrt{\sqrt{4m^4 + 1} - 2m^2} \quad (5.58)$$

The phase margin M_Φ of the system is then defined as:

$$M_\Phi = 180^\circ + \arg(F(j\omega_c)) \quad (\text{modulo } 360^\circ) \quad (5.59)$$

According to the expression of $F(j\omega)$ we get:

$$\arg(F(j\omega_c)) = -180^\circ + \arctan\left(\frac{2m\omega_0\omega_c}{\omega_c^2}\right) \quad (5.60)$$

It is worth noticing that we add -180° (*modulo* 360°) because the real part of the denominator of $F(j\omega)$ is negative.

Thus:

$$\begin{aligned} M_\Phi &= 180^\circ + \arg(F(j\omega_c)) \quad (\text{modulo } 360^\circ) \\ &= 180^\circ - 180^\circ + \arctan\left(2m\frac{\omega_0\omega_c}{\omega_c^2}\right) \\ &= \arctan\left(2m\frac{\omega_0}{\omega_c}\right) \\ &= \arctan\left(\frac{2m}{\sqrt{\sqrt{4m^4+1}-2m^2}}\right) \end{aligned} \quad (5.61)$$

Assuming $0 < m < 0.7$ the damping ratio m and the phase margin M_Φ are related through the following approximate relation:

$$M_\Phi(\text{in degrees}) \approx 100m \quad (5.62)$$

Thus the phase margin M_Φ is related to the damping ratio m , that is to the overshoot of the closed-loop system response.

In addition under the same assumption on m the 95% settling time t_s of second order underdamped closed-loop model can be approximated by:

$$t_s \approx \frac{4}{m\omega_0} \quad (5.63)$$

Then we get the following relation between the phase margin M_Φ and the settling time t_s :

$$\begin{cases} \tan(M_\Phi) = 2m\frac{\omega_0}{\omega_c} \\ t_s \approx \frac{4}{m\omega_0} \end{cases} \Rightarrow \tan(M_\Phi) \approx \frac{8}{t_s\omega_c} \quad (5.64)$$

Thus for a given phase margin M_Φ , increasing the gain crossover frequency ω_c reduces the settling time t_s .

From the preceding results it is clear that there exists a close relation between time domain specifications, typically settling time t_s and relative overshoot D , and frequency domain specifications, typically the phase margin M_Φ and the corresponding gain crossover frequency ω_c :

$$\begin{cases} D = e^{-\frac{\pi m}{\sqrt{1-m^2}}} \\ t_s \approx \frac{4}{m\omega_0} \\ \tan(M_\Phi) = 2m\frac{\omega_0}{\omega_c} \end{cases} \Leftrightarrow \begin{cases} m = \frac{-\ln(D)}{\sqrt{\pi^2 + \ln^2(D)}} \\ M_\Phi (\text{in degrees}) \approx 100m \\ \omega_c \approx \frac{8}{t_s \tan(M_\Phi)} \end{cases} \quad (5.65)$$

Roughly speaking, the damping ratio is closely related to the phase margin M_Φ whereas the settling time t_s decreases when gain crossover frequency ω_c increases. In other words system stability is improved and overshoot D is reduced by increasing the phase margin M_Φ whereas settling time t_s is reduced by increasing system's bandwidth ω_c .

The preceding relations also lead to the expected location of the dominant poles λ_{dp} and $\bar{\lambda}_{dp}$ of the closed-loop system:

$$\begin{cases} \lambda_{dp} = -m\omega_0 + j\omega_0\sqrt{1-m^2} \\ \bar{\lambda}_{dp} = -m\omega_0 - j\omega_0\sqrt{1-m^2} \end{cases} \quad (5.66)$$

5.4 ITAE performance index

A control system is optimum when a selected performance index J is minimized. In 1953 Graham developed the Integral of Time-weighted Absolute Error (ITAE) performance index which reads as follows:

$$J_{ITAE} = \int_0^\infty t |r(t) - y(t)| dt \quad (5.67)$$

This performance index used in the ITAE optimization process is a kind of tradeoff between a short settling time (time t explicitly appears in the integral to be minimized) and a precise closed-loop system (error $|r(t) - y(t)|$ appears in the integral to be minimized)

$m = 1$	$s + \omega_0$	
$m = 2$	$a_1 = 1.505$	$J_{ITAE} = 1.93556$
$m = 3$	$a_2 = 1.783, a_1 = 2.172$	$J_{ITAE} = 3.11623$
$m = 4$	$a_3 = 1.953, a_2 = 3.347, a_1 = 2.648$	$J_{ITAE} = 4.56372$
$m = 5$	$a_4 = 2.068, a_3 = 4.499, a_2 = 4.675, a_1 = 3.257$	$J_{ITAE} = 6.28854$
$m = 6$	$a_5 = 2.152, a_4 = 5.629, a_3 = 6.934, a_2 = 6.7920, \dots$ $a_1 = 3.740$	$J_{ITAE} = 8.29536$
$m = 7$	$a_6 = 2.217, a_5 = 6.745, a_4 = 9.349, a_3 = 11.580, \dots$ $a_2 = 8.680, a_1 = 4.323$	$J_{ITAE} = 10.5852$
$m = 8$	$a_7 = 2.275, a_6 = 7.849, a_5 = 11.888, a_4 = 17.588, \dots$ $a_3 = 16.116, a_2 = 11.339, a_1 = 4.815$	$J_{ITAE} = 13.1553$

Table 5.2: Values of coefficients a_i of $q(s)$ based on the ITAE performance index

Graham studied the values of the coefficients of the following transfer function $G(s)$ to minimize performance index J_{ITAE} :

$$G(s) = \frac{Y(s)}{R(s)} = \frac{N_{cl}(s)}{q(s)} \quad (5.68)$$

Where:

$$\begin{cases} N_{cl}(s) = \omega_0^m \\ q(s) = s^m + a_{m-1} \omega_0 s^{m-1} + a_{m-2} \omega_0^2 s^{m-2} + \dots + a_1 \omega_0^{m-1} s + \omega_0^m \end{cases} \quad (5.69)$$

In the preceding equation natural frequency ω_0 is a free parameter which has to be selected by the designer. Table 5.2 provides the values of coefficients a_i of $q(s)$ based on the ITAE performance index for a step input. Those coefficients are actually those which have been recomputed in 1989 by Y. Cao¹

In the following sections we consider the problem of designing a controller that provides asymptotic tracking of a reference input with zero steady state error.

Example 5.2. Consider a plant with the following transfer function:

$$F(s) = \frac{1}{(2s + 1)^2} \quad (5.70)$$

We wish to design a controller which minimize the ITAE performance index and also which leads to a settling time t_s for a step input less than 1 sec.

We choose for example a PID controller. Then the closed-loop transfer function without prefiltering ($C_{pf}(s) = 1$) reads:

$$\begin{aligned} \frac{Y(s)}{R_{pf}(s)} &= \frac{(K_d s^2 + K_p s + K_i) F(s)}{s + (K_d s^2 + K_p s + K_i) F(s)} \\ &= \frac{K_d s^2 + K_p s + K_i}{s(2s+1)^2 + K_d s^2 + K_p s + K_i} \\ &= \frac{K_d s^2 + K_p s + K_i}{4s^3 + (4+K_d)s^2 + (1+K_p)s + K_i} \end{aligned} \quad (5.71)$$

¹Y. Cao, Correcting the minimum ITAE standard forms of zero-displacement-error systems, Journal of Zhejiang University (Natural Science) Vol. 23, No.4, pp. 550-559, 1989

That is:

$$\frac{Y(s)}{R_{pf}(s)} = \frac{\frac{K_d}{4}s^2 + \frac{K_p}{4}s + \frac{K_i}{4}}{s^3 + \frac{4+K_d}{4}s^2 + \frac{1+K_p}{4}s + \frac{K_i}{4}} \quad (5.72)$$

The coefficients of the denominator which minimize the ITAE performance index are obtained from Table 5.2:

$$s^3 + 1.783\omega_0 s^2 + 2.172\omega_0^2 s + \omega_0^3 \quad (5.73)$$

We need to select ω_0 in order to meet the settling time requirement. Since $t_s \approx \frac{4}{m\omega_0}$ where m is set in a first attempt to 0.8 we get:

$$\begin{cases} t_s = 1 \\ m = 0.8 \end{cases} \Rightarrow \omega_0 \approx \frac{4}{m t_s} = 5 \quad (5.74)$$

Then equating the denominator of (5.72) with (5.73) we obtain the three coefficients of the PID controller as:

$$\begin{aligned} s^3 + \frac{4+K_d}{4}s^2 + \frac{1+K_p}{4}s + \frac{K_i}{4} &= s^3 + 8.915s^2 + 54.3s + 125 \\ \Rightarrow \begin{cases} K_i = 500 \\ K_p = 216.2 \\ K_d = 31.66 \end{cases} & \end{aligned} \quad (5.75)$$

Then the closed-loop transfer function without prefiltering (5.72) becomes:

$$\frac{Y(s)}{R_{pf}(s)} = \frac{7.915s^2 + 54.05s + 125}{s^3 + 8.915s^2 + 54.3s + 125} \quad (5.76)$$

Then we select the prefilter $C_{pf}(s)$ in order to achieve the required ITAE transfer function. We check that all the roots of polynomial $7.915s^2 + 54.05s + 125$ have negative real parts. Therefore we obtain for the transfer function of the prefilter:

$$C_{pf}(s) = \frac{125}{7.915s^2 + 54.05s + 125} \quad (5.77)$$

We finally get the transfer function of the whole system:

$$\begin{aligned} \frac{Y(s)}{R(s)} &= C_{pf}(s) \frac{Y(s)}{R_{pf}(s)} \\ &= C_{pf}(s) \frac{7.915s^2 + 54.05s + 125}{s^3 + 8.915s^2 + 54.3s + 125} \\ &= \frac{125}{s^3 + 8.915s^2 + 54.3s + 125} \end{aligned} \quad (5.78)$$

Figure 5.7 shows the response of the whole system to a unit step input. As expected the settling time lower than 1 sec. Furthermore the whole system has a zero steady state error and an overshoot about 1.6%. ■

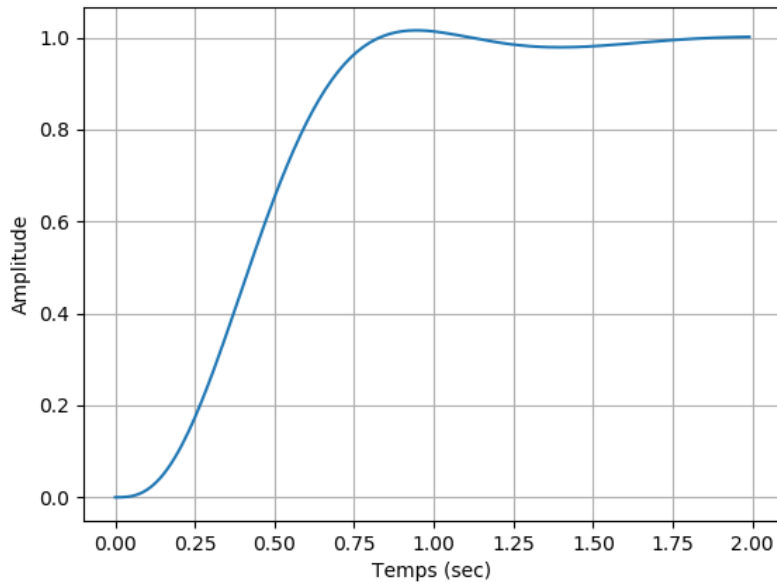


Figure 5.7: Unit step response of the whole system

5.5 PID controller

5.5.1 Controller presentation

The PID control is the most popular feedback controller used within the process industries. It was an essential element of early governors and has been successfully used for several years.

PID control consists of a controller based on three basic modes:

- the Proportional (P) mode,
- the Integral (I) mode and
- the Derivative (D) mode

The transfer function $C(s)$ of a PID controller is the following:

$$C(s) = K_p + \frac{K_i}{s} + K_d s \quad (5.79)$$

Note that if only the Proportional mode appears in the transfer function $C(s)$ of the controller, the controller is called a proportional controller:

$$C_P(s) = K_p \quad (5.80)$$

Similarly if only Proportional and Integral modes appears in the transfer function $C(s)$ of the controller, the controller is called a proportional and integral controller (PI controller):

$$C_{PI}(s) = K_p + \frac{K_i}{s} \quad (5.81)$$

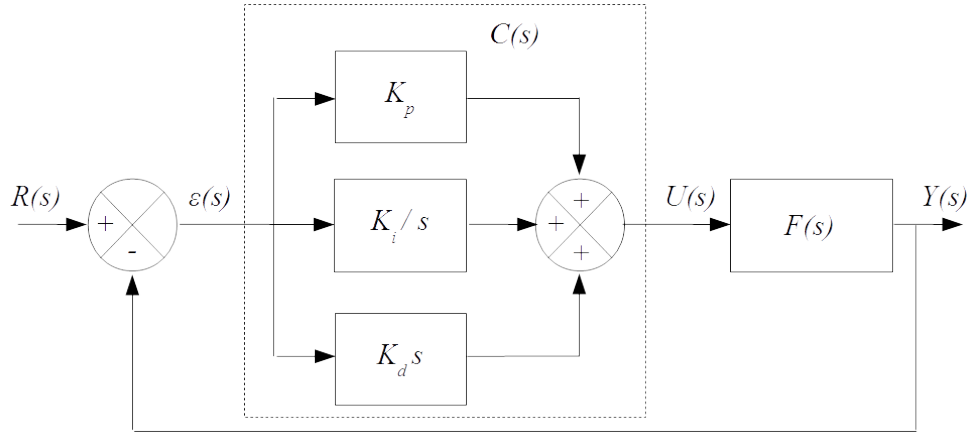


Figure 5.8: PID controlled system

Finally if only Proportional and Derivative modes appears in the transfer function $C(s)$ of the controller, the controller is called a proportional and derivative controller (PD controller):

$$C_{PD}(s) = K_p + K_d s \quad (5.82)$$

Figure 5.8 shows a PID controlled system. Note that the derivative term $K_d s$ is not a proper transfer function; as a consequence the derivative term is either measured by a dedicated sensor or is changed to be $\frac{K_d s}{1 + \tau_d s}$ where τ_d is a very small time constant (i.e. close to zero):

$$K_d s \rightarrow \frac{K_d s}{1 + \tau_d s} \quad \text{where } \tau_d \approx 0 \quad (5.83)$$

The previous approximation of the derivative term can be interpreted as the ideal derivative term $K_d s$ filtered by a first-order system $\frac{1}{1 + \tau_d s}$ with time constant τ_d . A typical expression of τ_d is the following where N is chosen between 10 and 20:

$$\tau_d = \frac{K_d}{K_p N} \quad (5.84)$$

As seen in the previous chapter the steady state error in proportional control will decrease with increasing gain K_p (assuming no integral action). As far as integral action is concerned the term $\frac{K_i}{s}$ usually decreases the overall bandwidth of the system thus increasing the overshoot and the settling time of the closed-loop system. On the other hand the derivative action $K_d s$ usually increases the overall bandwidth of the system, thus increasing the overshoot and the settling time of the closed-loop system.

The effect which is usually observed when increasing the PID gains K_p , K_i and K_d on the step response of the closed-loop system is summarized on Table 5.3.

One disadvantage of this ideal PID controller is that a sudden change in set point $r(t)$ (and hence in the error $\epsilon(t)$) will cause the derivative term to become very large and thus provide a *derivative kick* to the plant $F(s)$ which

PID gain	Overshoot	Settling time	Steady-state error
Increasing K_p ($K_p \nearrow$)	Increases \nearrow	Decreases \searrow	Decreases \searrow
Increasing K_i ($K_i \nearrow$)	Increases \nearrow	Increases \nearrow	Eliminates
Increasing K_d ($K_d \nearrow$)	Decreases \searrow	Decreases \searrow	Increases \nearrow

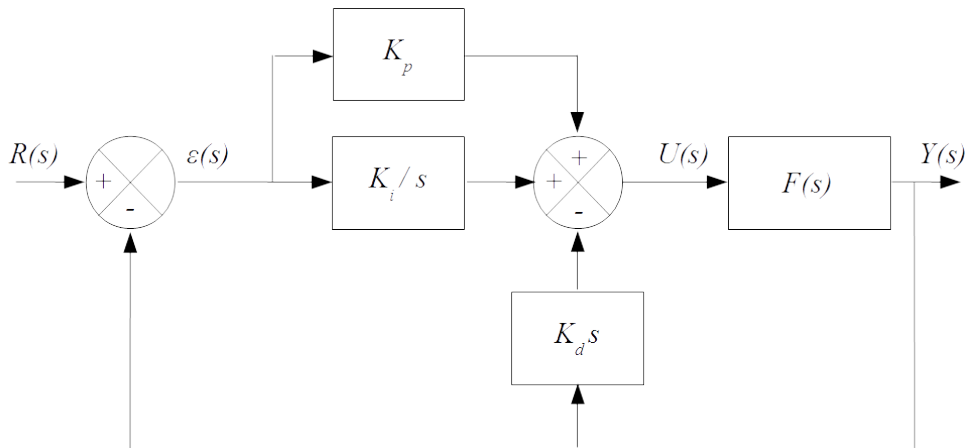
Table 5.3: Usual effect when increasing K_p , K_i and K_d gains

Figure 5.9: PI-D controlled system

is not desirable. An alternative implementation consists in putting the derivative action on the measurement $Y(s)$ and not the error $\epsilon(s)$. This leads to an alternative PID control scheme, known as PI-D control. Figure 5.9 shows a PI-D controlled system.

Finally it is worth noticing that whatever the location of the PID controller $C(s)$ in the loop, either after the comparator as in Figure 5.10, in the feedback loop as in Figure 5.11 or split in the PI-D configuration as in Figure 5.9, the denominator $q(s)$ of the closed-loop transfer function remains the same:

$$q(s) = 1 + F(s)C(s) = 1 + F(s) \left(K_p + \frac{K_i}{s} + K_d s \right) \quad (5.85)$$

Indeed:

- The closed-loop transfer function of the unity feedback loop in Figure 5.10

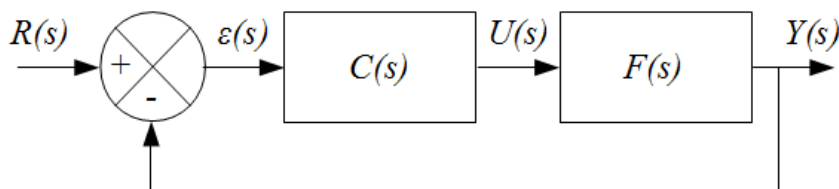
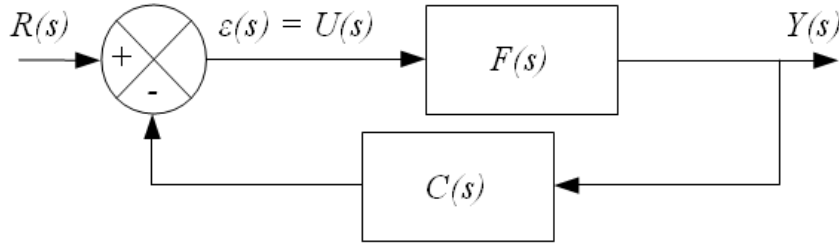


Figure 5.10: Unity feedback loop

Figure 5.11: Feedback loop with $C(s)$

reads:

$$\frac{Y(s)}{R(s)} = \frac{F(s)C(s)}{1 + F(s)C(s)} = \frac{F(s)C(s)}{q(s)} \quad (5.86)$$

- The closed-loop transfer function of the feedback loop in Figure 5.9 where the PID controller is split reads:

$$\begin{aligned} Y(s) &= F(s) \left(-K_d s Y(s) + \left(K_p + \frac{K_i}{s} \right) (R(s) - Y(s)) \right) \\ \Leftrightarrow Y(s) \left(1 + F(s) \left(K_p + \frac{K_i}{s} + K_d s \right) \right) &= F(s) \left(K_p + \frac{K_i}{s} \right) R(s) \\ \Leftrightarrow \frac{Y(s)}{R(s)} &= \frac{F(s) \left(K_p + \frac{K_i}{s} \right)}{1 + F(s) \left(K_p + \frac{K_i}{s} + K_d s \right)} = \frac{F(s) \left(K_p + \frac{K_i}{s} \right)}{q(s)} \end{aligned} \quad (5.87)$$

- The closed-loop transfer function of the feedback loop with $C(s)$ in Figure 5.11 reads:

$$\frac{Y(s)}{R(s)} = \frac{F(s)}{1 + F(s)C(s)} = \frac{F(s)}{q(s)} \quad (5.88)$$

As a conclusion the transfer function of the closed-loop has the same denominator whatever the PID location in the loop.

5.5.2 Ziegler and Nichols tuning method

In 1942 Ziegler and Nichols published a gain tuning method to achieve fast closed-loop step response without excessive oscillations with PID controller in the loop. The method they have developed is based on the minimization on the following criteria which is called the Integral of the Absolute Error (IAE) criteria:

$$J_{IAE} = \int_0^{\infty} |r(t) - y(t)| dt \quad (5.89)$$

The Ziegler and Nichols method first consider that gains K_i and K_d are set to zero whereas the proportional gain K_p is increased until the closed-loop system reaches the boundary of instability. The gain on the border of instability is called the critical gain and is denoted K_{cr} whereas the period of sustained oscillations is called the ultimate period of oscillation and is denoted T_{cr} .

At a gain just below the ultimate gain, the closed-loop oscillates with decaying amplitude while at a gain just above the ultimate gain the amplitude of oscillations increases with time. Critical gain K_{cr} and ultimate frequency

Controller type	K_p	K_i	K_d
Proportional (P)	$0.5K_{cr}$	0	0
Proportional-Integral (PI)	$0.45K_{cr}$	$\frac{0.54K_{cr}}{T_{cr}}$	0
Proportional-Integral-Derivative (PID)	$0.6K_{cr}$	$\frac{1.2K_{cr}}{T_{cr}}$	$0.075K_{cr}T_{cr}$

Table 5.4: Ziegler-Nichols PID controller gains

ω_{cr} can be obtained by noticing that at the verge of instability (where the system is said to be marginally stable) the characteristic equation must have at least an imaginary root which is noticed $j\omega_{cr}$. Thus by substituting s by $j\omega_{cr}$ in the characteristic equation of the closed-loop system (i.e. the denominator of the closed-loop system which is set to zero) we get a complex equation that can be converted into two simultaneous equations by noticing that the real part and the imaginary part shall be equal to zero. From these we can solve for the two unknowns: one is the ultimate frequency ω_{cr} and the other is the ultimate gain K_{cr} . The ultimate period of oscillation T_{cr} is related to the ultimate frequency ω_{cr} by:

$$T_{cr} = \frac{2\pi}{\omega_{cr}} \quad (5.90)$$

Notice that the ultimate gain K_{cr} and the ultimate frequency ω_{cr} can be computed thanks to the Routh array in the same fashion the gain margin is computed.

Table 5.4 provides the Ziegler-Nichols PID controller gains. It was obtained by Ziegler and Nichols who aimed to achieve an underdamped response to a unit step input for the following plant model:

$$F(s) = \frac{K_0}{1 + \tau s} e^{-Ts} \text{ where } \tau, T > 0 \quad (5.91)$$

It is worth noticing that if the critical gain K_{cr} cannot be computed then there is no way to set the PID controller gains following the Ziegler-Nichols method.

5.5.3 Phillips and Harbor tuning method

Setting K_p and K_d parameters

Others methods exist to set PID controllers. We present hereafter the method proposed by Phillips and Harbor ².

Assume that the controller is a PID controller. Its transfer function $C(s)$ reads:

$$C(s) = K_p + \frac{K_i}{s} + K_d s \quad (5.92)$$

As far as it is assumed that the open-loop transfer function $C(s)F(s)$ passes through the point $1 \times e^{j(-180^\circ + M_\Phi)}$ for the gain crossover frequency ω_c and the

²Charles L. Phillips and Royce D. Harbor. Feedback Control System. Prentice-Hall, 1991

phase margin M_Φ we get:

$$\begin{aligned} C(j\omega_c)F(j\omega_c) &= 1 \times e^{j(-180^\circ + M_\Phi)} \\ \Leftrightarrow \begin{cases} |C(j\omega_c)| &= \frac{1}{|F(j\omega_c)|} \\ \arg(C(j\omega_c)) + \arg(F(j\omega_c)) &= -180^\circ + M_\Phi \end{cases} \end{aligned} \quad (5.93)$$

Denoting by φ the angle of $C(j\omega_c)$ we get:

$$\varphi := \arg(C(j\omega_c)) = -180^\circ + M_\Phi - \arg(F(j\omega_c)) \quad (5.94)$$

Furthermore:

$$\begin{aligned} C(j\omega_c) &= K_p + j \left(K_d \omega_c - \frac{K_i}{\omega_c} \right) \\ &= |C(j\omega_c)| e^{j \arg(C(j\omega_c))} \\ &= |C(j\omega_c)| (\cos(\varphi) + j \sin(\varphi)) \end{aligned} \quad (5.95)$$

Equating the real part and the imaginary part yields:

$$\begin{cases} K_p &= |C(j\omega_c)| \cos(\varphi) = \frac{\cos(\varphi)}{|F(j\omega_c)|} \\ K_d \omega_c - \frac{K_i}{\omega_c} &= |C(j\omega_c)| \sin(\varphi) = \frac{\sin(\varphi)}{|F(j\omega_c)|} \end{cases} \quad (5.96)$$

Consequently the method proposed by Phillips and Harbor² has the following steps:

- First get the required phase margin M_Φ as well as the settling time t_s from the design specifications;
- Then choose the gain crossover frequency ω_c corresponding to the required phase margin M_Φ and the settling time t_s . Frequency ω_c is chosen by trial and error but shall satisfy:

$$\omega_c \geq \frac{8}{t_s \tan(M_\Phi)} \quad (5.97)$$

- Compute the phase angle φ of the controller $C(s)$ at frequency ω_c as:

$$\varphi := \arg(C(j\omega_c)) = -180^\circ + M_\Phi - \arg(F(j\omega_c)) \quad (5.98)$$

It is worth noticing that angle φ shall be between -90° and $+90^\circ$: this corresponds to the minimum and maximum phase shift achievable by the controller.

- Finally choose the integral gain $|K_i|$ (for example $|K_i| = 0.01$ but it could be $K_i = 0$ when integral action is not required); the proportional gain K_p and derivative gain K_d are then given by:

$$\begin{cases} K_p &= \frac{\cos(\varphi)}{|F(j\omega_c)|} \\ K_d &= \frac{\sin(\varphi)}{\omega_c |F(j\omega_c)|} + \frac{K_i}{\omega_c^2} \end{cases} \quad (5.99)$$

If the achieved performances are not satisfactory then frequency ω_c has to be tuned by trial and error. Usually ω_c is increased until the real part of some closed-loop poles is equal to the real part of the required dominant poles.

However remember that even then the closed-loop system may not achieved the required performance or can be unstable as far as the design process does not ensure the actual location of the dominant poles. Indeed the design process come from the assumption that the closed-loop system can be modelled by a second order model (which is obviously a reduced order model).

Tuning K_i through root locus technique

The influence of parameter K_i can be investigated thanks to the root locus technique. Indeed let's write transfer function $F(s)$ as follows where $N(s)$ and $D(s)$ are polynomials:

$$F(s) = \frac{N(s)}{D(s)} \quad (5.100)$$

Transfer function $F(s)$ is assumed to be a proper and co-prime rational fraction.

A PID controller is used to control the plant. The transfer function $C(s)$ of the PID controller is:

$$C(s) = K_p + \frac{K_i}{s} + K_d s \quad (5.101)$$

The controller $C(s)$ is put either in front the plant in a unity feedback loop as depicted in Figure 5.10 or in the feedback path as depicted in Figure 5.11 or in a PI-D configuration as depicted in Figure 5.9. For all configurations the denominator $q(s)$ of the closed-loop transfer function $H(s) = \frac{Y(s)}{R(s)}$ reads:

$$q(s) = sD(s) + N(s) (K_d s^2 + K_p s + K_i) \quad (5.102)$$

Using the expression of parameters K_p and K_d provided in (5.99) the closed-loop characteristic polynomial reads:

$$\begin{aligned} q(s) &= sD(s) + N(s) (K_d s^2 + K_p s + K_i) \\ &= sD(s) + N(s) \left(\left(\frac{\sin(\varphi)}{\omega_c |F(j\omega_c)|} + \frac{K_i}{\omega_c^2} \right) s^2 + \frac{\cos(\varphi)}{|F(j\omega_c)|} s + K_i \right) \\ &= sD(s) + N(s) \left(\frac{\sin(\varphi)}{\omega_c |F(j\omega_c)|} s^2 + \frac{\cos(\varphi)}{|F(j\omega_c)|} s \right) + K_i N(s) \left(\frac{s^2}{\omega_c^2} + 1 \right) \end{aligned} \quad (5.103)$$

Dividing both side of the closed-loop characteristic equation by the terms without K_i reads:

$$\begin{aligned} q(s) &= 0 \\ \Leftrightarrow sD(s) + N(s) \left(\frac{\sin(\varphi)}{\omega_c |F(j\omega_c)|} s^2 + \frac{\cos(\varphi)}{|F(j\omega_c)|} s \right) + K_i N(s) \left(\frac{s^2}{\omega_c^2} + 1 \right) &= 0 \\ \Leftrightarrow 1 + K_i L(s) &= 0 \end{aligned} \quad (5.104)$$

Where:

$$L(s) = \frac{N(s) \left(\frac{s^2}{\omega_c^2} + 1 \right)}{sD(s) + N(s) \left(\frac{\sin(\varphi)}{\omega_c |F(j\omega_c)|} s^2 + \frac{\cos(\varphi)}{|F(j\omega_c)|} s \right)} \quad (5.105)$$

The division by the terms without K_i doesn't change the roots of the closed-loop transfer function. $L(s)$ is a proper transfer function since the degrees of its numerator and denominator equal those of the closed-loop transfer function's numerator and denominator respectively.

The design procedure is the following:

- Draw the root locus of $L(s)$ as K_i varies;
- Choose the value of K_i to satisfy the specifications from the root locus;
- Once K_i is chosen set K_p and K_d according to (5.99).

5.5.4 Real dominant pole placement with PI or PD controller

Consider a plant described by its transfer function $F(s)$ where $N(s)$ and $D(s)$ are polynomials:

$$F(s) = \frac{N(s)}{D(s)} \quad (5.106)$$

Transfer function $F(s)$ is assumed to be a proper and co-prime rational fraction.

A Proportional-Integral (PI) controller is used to control the plant. The transfer function $C(s)$ of the PI controller is:

$$C(s) = K_p + \frac{K_i}{s} \quad (5.107)$$

The controller $C(s)$ is put either in front the plant in a unity feedback loop as depicted in Figure 5.10 or in the feedback path as depicted in Figure 5.11. Nevertheless, the integral action is usually put in the forward path in order to achieve an unity closed-loop static gain. For all configurations the closed-loop characteristic equation is:

$$1 + C(s)F(s) = 0 \Leftrightarrow C(s) = \frac{-1}{F(s)} \quad (5.108)$$

Furthermore the denominator $q(s)$ of the closed-loop transfer function $H(s) = \frac{Y(s)}{R(s)}$ reads as follows irrespective the chosen feedback loop configuration:

$$q(s) = sD(s) + N(s)(K_p s + K_i) \quad (5.109)$$

Suppose that the requirements of the closed-loop control performance in frequency or time domain are converted into a real dominant pole $\lambda_{dp} \in \mathbb{R}^-$. Substitute λ_{dp} into (5.108) yields:

$$K_p + \frac{K_i}{\lambda_{dp}} = \frac{-1}{F(\lambda_{dp})} \Leftrightarrow K_i = -\lambda_{dp} \left(\frac{1}{F(\lambda_{dp})} + K_p \right) \quad (5.110)$$

Substituting (5.110) into (5.108) yields:

$$\begin{aligned}
K_p + \frac{K_i}{s} &= \frac{-1}{F(s)} := -\frac{D(s)}{N(s)} \\
\Leftrightarrow 0 &= sD(s) + N(s)(K_p s + K_i) \\
&= sD(s) + N(s) \left(K_p s - \lambda_{dp} \left(\frac{1}{F(\lambda_{dp})} + K_p \right) \right) \\
&= sD(s) - \lambda_{dp} \frac{N(s)}{F(\lambda_{dp})} + K_p N(s) (s - \lambda_{dp}) \\
&= \left(sD(s) - \lambda_{dp} \frac{N(s)}{F(\lambda_{dp})} \right) \left(1 + K_p \frac{N(s)(s - \lambda_{dp})}{sD(s) - \lambda_{dp} \frac{N(s)}{F(\lambda_{dp})}} \right)
\end{aligned} \tag{5.111}$$

Simplifying by $\left(sD(s) - \lambda_{dp} \frac{N(s)}{F(\lambda_{dp})} \right)$ yields:

$$1 + K_p L(s) = 0 \tag{5.112}$$

Where the *fictitious* transfer function $L(s)$ is defined by:

$$L(s) = \frac{N(s)(s - \lambda_{dp})}{sD(s) - \lambda_{dp} \frac{N(s)}{F(\lambda_{dp})}} \tag{5.113}$$

Fictitious transfer function $L(s)$ is proper since the degrees of its numerator and denominator equal those of the closed-loop transfer function's numerator and denominator respectively.

The design procedure for the PI controller is then the following:

- Set the value of the dominant pole λ_{dp} according to the specifications;
- Draw the root locus of fictitious transfer function $L(s)$ given by (5.113) as $K_p \geq 0$ varies. The location of the pole λ_{dp} will not move despite the change of K_p . Note that negative values of K_p can be achieved by plotting the root locus of $-L(s)$;
- From the root locus of $L(s)$ choose the value of K_p which guarantee that λ_{dp} is the dominant pole of the closed-loop;
- Once K_p is chosen set K_i according to (5.110).

Note that the same kind of result can be obtained by a Proportional-Derivative (PD) controller:

$$C(s) = K_p + K_d s \tag{5.114}$$

Then:

$$\begin{cases} K_d = -\frac{1}{\lambda_{dp}} \left(\frac{1}{F(\lambda_{dp})} + K_p \right) \\ L(s) = \frac{N(s) \left(1 - \frac{s}{\lambda_{dp}} \right)}{D(s) - \frac{s N(s)}{\lambda_{dp} F(\lambda_{dp})}} \end{cases} \tag{5.115}$$

Finally, denoting by n the degree of $D(s)$, $n := \deg(D(s))$, it is worth noticing that the closed-loop transfer function has $n+1$ poles with a PI controller (the PI controller *adds* a pole) whereas the closed-loop transfer function still has n poles with a PD controller as soon as $F(s)$ is *strictly proper*, i.e. $\deg(N(s)) < \deg(D(s))$.

5.5.5 Complex conjugate dominant pole placement with PID controller

We present hereafter the method proposed by Wang & all³. This is an extension of the methods presented in section 5.5.4.

Consider a plant described by its transfer function $F(s)$ where $N(s)$ and $D(s)$ are polynomials:

$$F(s) = \frac{N(s)}{D(s)} \quad (5.116)$$

Transfer function $F(s)$ is assumed to be a proper and co-prime rational fraction.

A PID controller is used to control the plant. The transfer function $C(s)$ of the PID controller is:

$$C(s) = K_p + \frac{K_i}{s} + K_d s \quad (5.117)$$

The controller $C(s)$ is put either in front the plant in a unity feedback loop as depicted in Figure 5.10 or in the feedback path as depicted in Figure 5.11 or in a PI-D configuration as depicted in Figure 5.9. Nevertheless, the integral action is usually put in the forward path in order to achieve an unity closed-loop static gain. For all configurations the closed-loop characteristic equation is:

$$1 + C(s)F(s) = 0 \Leftrightarrow C(s) = \frac{-1}{F(s)} \quad (5.118)$$

Furthermore the denominator $q(s)$ of the closed-loop transfer function $H(s) = \frac{Y(s)}{R(s)}$ reads as follows irrespective the chosen feedback loop configuration:

$$q(s) = sD(s) + N(s)(K_d s^2 + K_p s + K_i) \quad (5.119)$$

Suppose that the requirements of the closed-loop control performance in frequency or time domain are converted into a pair of complex conjugate dominant poles $\{\lambda_{dp}, \bar{\lambda}_{dp}\}$:

$$\begin{cases} \lambda_{dp} = a + jb \\ \bar{\lambda}_{dp} = a - jb \end{cases} \Leftrightarrow \begin{cases} a = \text{Re}(\lambda_{dp}) \\ b = \text{Im}(\lambda_{dp}) \end{cases} \quad (5.120)$$

The dominance of poles λ_{dp} and $\bar{\lambda}_{dp}$ requires that the ratio of the real part of any of other poles to a exceeds r where r is usually greater than 3. Thus, we want all other poles to be located at the left of the line $s = r a$. The problem of the guaranteed dominant pole placement is to find the PID parameters such that all the closed-loop poles lie in the required region except the dominant poles λ_{dp} and $\bar{\lambda}_{dp}$.

Substitute λ_{dp} by $a + jb$ into (5.118) yields:

$$K_p + \frac{K_i}{a + jb} + K_d(a + jb) = \frac{-1}{F(\lambda_{dp})} \quad (5.121)$$

³Guaranteed dominant pole placement with PID controllers, Qing-Guo Wang, Zhiping Zhang, Karl Johan Astrom, Lee See Chek, Journal of Process Control 19 (2009) 349–35

This complex equations can be decomposed into two sub-equations, one from the real part and the other from the imaginary part. Solving the two equations for K_i and K_d in terms of K_p yields:

$$\begin{cases} K_i = -(a^2 + b^2) \left(\frac{K_p}{2a} + X_1 \right) \\ K_d = -\frac{K_p}{2a} + X_2 \end{cases} \quad (5.122)$$

Where:

$$\begin{cases} X_1 = \frac{1}{2b} \text{Im} \left(\frac{-1}{F(\lambda_{dp})} \right) - \frac{1}{2a} \text{Re} \left(\frac{-1}{F(\lambda_{dp})} \right) \\ X_2 = \frac{1}{2b} \text{Im} \left(\frac{-1}{F(\lambda_{dp})} \right) + \frac{1}{2a} \text{Re} \left(\frac{-1}{F(\lambda_{dp})} \right) \end{cases} \quad (5.123)$$

This simplifies the original problem to a one-parameter problem in K_p for which the root locus method will be applied. We plot the roots of the closed-loop characteristic equation for all positive values of K_p and determine the range of K_p such that the roots except the chosen dominant pair are the required region. Substituting (5.122) into (5.118) yields:

$$1 + sX_2 \frac{N(s)}{D(s)} - (a^2 + b^2) \frac{X_1 N(s)}{s D(s)} - K_p \frac{s^2 - 2as + (a^2 + b^2) N(s)}{2as D(s)} = 0 \quad (5.124)$$

Dividing both side by the terms independent of K_p gives after some manipulations:

$$1 + K_p L(s) = 0 \quad (5.125)$$

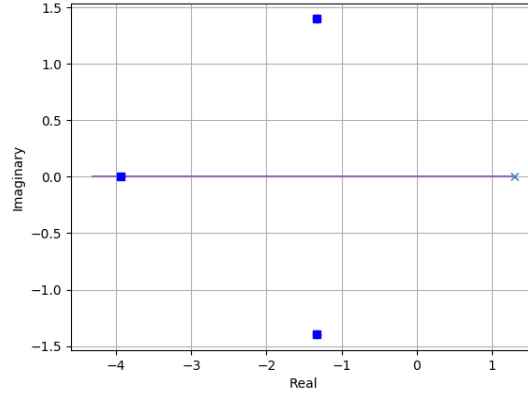
Where the *fictitious* transfer function $L(s)$ is defined by:

$$L(s) = \frac{-N(s) (s^2 - 2as + (a^2 + b^2))}{2asD(s) + N(s) (2aX_2s^2 - 2a(a^2 + b^2)X_1)} \quad (5.126)$$

The division by the terms independent of K_p doesn't change the roots of the closed-loop transfer function. Fictitious transfer function $L(s)$ is proper since the degrees of its numerator and denominator equal those of the closed-loop transfer function's numerator and denominator respectively.

The design procedure for the PID controller is then the following:

- Set the value of the dominant poles λ_{dp} and $\bar{\lambda}_{dp}$ according to the specifications;
- Draw the root locus of fictitious transfer function $L(s)$ given by (5.126) as $K_p \geq 0$ varies. The location of the poles λ_{dp} and $\bar{\lambda}_{dp}$ will not move despite the change of K_p . Note that negative values of K_p can be achieved by plotting the root locus of $-L(s)$;
- From the root locus of $L(s)$ choose the value of K_p which guarantee that λ_{dp} and $\bar{\lambda}_{dp}$ are the dominant poles of the closed-loop;
- Once K_p is chosen set K_i and K_d according to (5.122).

Figure 5.12: Root locus of $L(s)$ as K_p varies

Example 5.3. Consider a plant with the following transfer function:

$$F(s) = \frac{1}{(2s + 1)^2} \quad (5.127)$$

We wish to design a PID controller such that the closed-loop system exhibits a time response t_s of 3 seconds and an overshoot of 5%.

Using the results of section 5.3, dominant poles λ_{dp} and $\bar{\lambda}_{dp}$ of the closed-loop system are the following:

$$\begin{cases} a = -m \omega_0 \approx -\frac{4}{t_s} \\ m = \frac{-\ln(D)}{\sqrt{si^2 + \ln^2(D)}} \\ \omega_0 \approx \frac{4}{m t_s} \\ b = \omega_0 \sqrt{1 - m^2} \end{cases} \Rightarrow \begin{cases} \lambda_{dp} = a + j b = -1.33 + j1.4 \\ \bar{\lambda}_{dp} = a - j b = -1.33 - j1.4 \end{cases} \quad (5.128)$$

Root locus of transfer function $L(s)$ given by (5.126) is shown in Figure 5.12.

It is clear that whatever the value of K_p the required poles at $-1.33 \pm j1.4$ are invariant. The third pole of the closed-loop transfer function is located on the horizontal branch. The square around the value -4 corresponds to a value of K_p set to 56; it is indeed a fast pole comparatively to the two required dominant poles. The values of K_i and K_d obtained thanks to (5.122) are the following:

$$K_p = 56 \Rightarrow \begin{cases} K_i = 58.88 \\ K_d = 22.44 \end{cases} \quad (5.129)$$

The step response of the closed-loop system obtained thanks to the designed PID is shown in Figure 5.13.

Once the PID controller $C(s)$ is placed in the direct path, the closed-loop transfer function reads:

$$\frac{Y(s)}{R(s)} = \frac{F(s)C(s)}{1 + F(s)C(s)} = \frac{22.44s^2 + 56s + 58.89}{4s^3 + 26.44s^2 + 57s + 58.89} \quad (5.130)$$

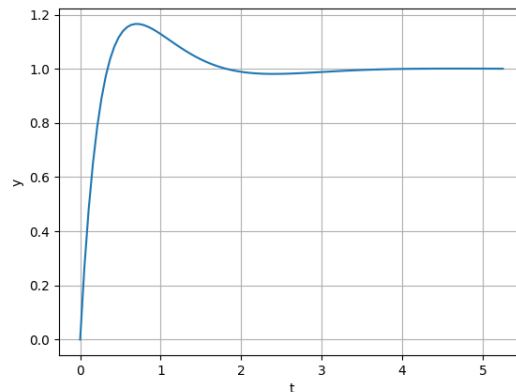


Figure 5.13: Closed loop unit step response with PID controller

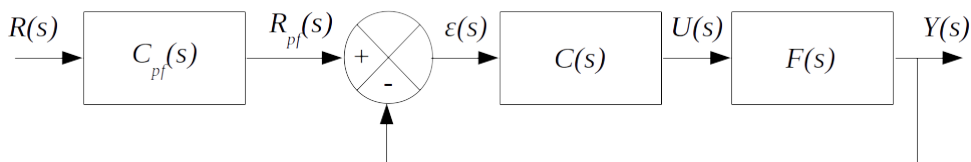


Figure 5.14: Unity feedback loop with prefilter

We can check that the poles of the closed-loop transfer function are those drawn on Figure 5.12, that are the two expected dominant poles $-1.33 \pm j1.4$ and the fast pole -4 .

■

5.6 Prefiltering

5.6.1 Purpose

Let's come back to the unit step response shown in Figure 5.13. It is clear that the achieved overshoot is greater than the specification at 5%. Why? Basically because the specifications based on dominant pole placement assume that the numerator of the closed-loop system is a constant, which is usually not the case because the numerator of the closed-loop system is simply a consequence of the designed controller $C(s)$. In order to change into a constant the numerator of the closed-loop system, a prefilter $C_{pf}(s)$ may be used.

As shown in Figure 5.14, the prefilter $C_{pf}(s)$ is a controller which is situated outside the feedback loop.

What is the purpose of the prefilter? Once the controller $C(s)$ is designed, the poles of the feedback loop transfer function $\frac{Y(s)}{R_{pf}(s)}$ are set, but not its zeros.

These zeros may cause undesirable overshoots in the transient response of the closed-loop system. The purpose of the prefilter $C_{pf}(s)$ is to reduce or eliminate such overshoots in the closed-loop system. Additionally the prefilter

may annihilate slow stable poles which sometimes cannot be shifted by the controller. Moreover in the case where the zeros are positive, the system without prefilter may exhibit undesirable non-minimum phase behaviour (see section 4.10).

The transfer function of a *basic* prefilter reads as follows where constant K_{pf} is set such that the static gain of $\frac{Y(s)}{R(s)}$ is unitary, meaning that the position error is zero, and where time constant λ_{pf} is set through a trial and error process until that the transient response becomes satisfactory:

$$\begin{cases} C_{pf}(s) = \frac{K_{pf}}{s+\lambda_{pf}} \\ \left. \frac{Y(s)}{R(s)} \right|_{s=0} = \frac{K_{pf}}{s+\lambda_{pf}} \frac{F(s)C(s)}{1+F(s)C(s)} \Big|_{s=0} := 1 \Rightarrow K_{pf} = \lambda_{pf} \frac{1+F(0)C(0)}{F(0)C(0)} \end{cases} \quad (5.131)$$

We will see in the following sections how to enhance the design of the prefilter.

5.6.2 Transfer function with negative real-parts zeros

We focus in Figure 5.14. Let $N_{cl}(s)$ be the numerator of transfer function $\frac{Y(s)}{R_{pf}(s)}$ and $q(s)$ its denominator:

$$\frac{Y(s)}{R_{pf}(s)} = \frac{F(s)C(s)}{1+F(s)C(s)} = \frac{N_{cl}(s)}{q(s)} \quad (5.132)$$

In this section we will assume that transfer function $\frac{Y(s)}{R_{pf}(s)}$ has all its zeros with negative real-parts, or equivalently that all the roots of $N_{cl}(s)$ are located in the left half plane.

Prefilter $C_{pf}(s)$ is designed such that its poles cancel the zeros of the closed-loop system (i.e. the roots of $N_{cl}(s)$). If there is no pole of the closed-loop system to cancel, the numerator of the prefilter is set to be a constant K_{pf} . In such a case the transfer function of the full system reads:

$$\frac{Y(s)}{R(s)} = \frac{K_{pf}}{q(s)} \quad (5.133)$$

As a consequence the transfer function of the prefilter reads:

$$C_{pf}(s) = \frac{K_{pf}}{N_{cl}(s)} \quad (5.134)$$

Note that this is only possible because the roots of $N_{cl}(s)$ have negative real-parts, meaning $C_{pf}(s)$ is stable.

Usually constant K_{pf} is set such that the static gain of $\frac{Y(s)}{R(s)}$ is unitary, meaning that the position error is zero:

$$\left. \frac{Y(s)}{R(s)} \right|_{s=0} = 1 \Rightarrow K_{pf} = D_{cl}(0) \quad (5.135)$$

Additionally the numerator of the prefilter may also cancel some *slow* stable poles (poles in the left plane) of the closed-loop system when they cannot be shifted by the controller $C(s)$. In this case, the numerator of the prefilter $C_{pf}(s)$ is no more a constant.

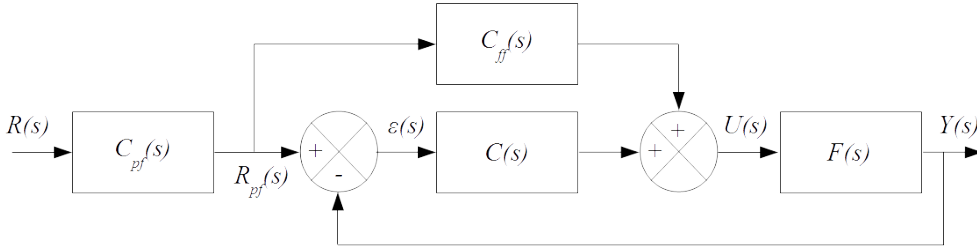


Figure 5.15: Feedback loop with feedforward compensation and prefilter

5.6.3 Transfer function with positive real-parts zeros

When feedback loop transfer function $\frac{Y(s)}{R_{pf}(s)} = \frac{N_{cl}(s)}{q(s)}$ has zeros with positive real parts the design technique presented in section 5.6.2 to design the prefilter $C_{pf}(s)$ is not applicable. Indeed in that situation $N_{cl}(s)$ has roots with positive real parts and thus prefilter $C_{pf}(s) = \frac{K_{pf}}{N_{cl}(s)}$ is unstable.

One way to tackle this problem is to use a feedforward compensator $C_{ff}(s)$ as shown in Figure 5.15.

Thanks to the feedforward compensator $C_{ff}(s)$, transfer function $\frac{Y(s)}{R_{pf}(s)}$ reads as follows:

$$\frac{Y(s)}{R_{pf}(s)} = \frac{F(s)(C(s) + C_{ff}(s))}{1 + F(s)C(s)} \quad (5.136)$$

Let $N_F(s)$ and $D_F(s)$ be the numerator and the denominator of $F(s)$, respectively. Similarly let $N_C(s)$ and $D_C(s)$ be the numerator and the denominator of $C(s)$ and $N_{ff}(s)$ and $D_{ff}(s)$ be the numerator and the denominator of $C_{ff}(s)$. Thus transfer function $\frac{Y(s)}{R_{pf}(s)}$ reads:

$$\begin{cases} F(s) = \frac{N_F(s)}{D_F(s)} \\ C(s) = \frac{N_C(s)}{D_C(s)} \\ C_{ff}(s) = \frac{N_{ff}(s)}{D_{ff}(s)} \end{cases} \Rightarrow \frac{Y(s)}{R_{pf}(s)} = \frac{N_F(s)(N_C(s)D_{ff}(s) + D_C(s)N_{ff}(s))}{D_{ff}(s)(N_F(s)N_C(s) + D_F(s)D_C(s)} \quad (5.137)$$

Choosing the feedforward compensator $C_{ff}(s)$ to be a constant gain K_{ff} leads to the following expression for the transfer function $\frac{Y(s)}{R_{pf}(s)}$:

$$C_{ff}(s) = K_{ff} \Rightarrow \frac{Y(s)}{R_{pf}(s)} = \frac{N_F(s)(N_C(s) + K_{ff}D_C(s))}{N_F(s)N_C(s) + D_F(s)D_C(s)} \quad (5.138)$$

We recognize in polynomial $N_F(s)N_C(s) + D_F(s)D_C(s)$ the polynomial $q(s)$ obtained when $C_{ff}(s) = 0$. Consequently the denominator of feedback loop transfer function $\frac{Y(s)}{R_{pf}(s)}$ remains the same when $C_{ff}(s) = K_{ff}$ while its numerator becomes $N_F(s)(N_C(s) + K_{ff}D_C(s))$ when feedforward is used. Assuming that $N_F(s)$ has no root in the right half plane, constant K_{ff} shall be chosen such that all the roots of $N_C(s) + K_{ff}D_C(s)$ have negative real parts to be able to compute a stable prefilter $C_{pf}(s)$. This can be achieved through the root locus of the *fictitious* transfer function $\frac{D_C(s)}{N_C(s)}$. Take care that

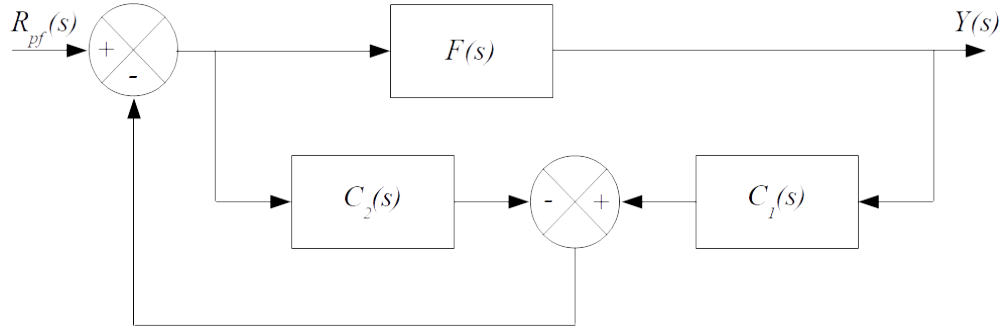


Figure 5.16: Feedback loop non-minimum phase plant

here $D_C(s)$ is the numerator of the *fictitious* transfer function whereas $N_C(s)$ is its denominator.

Once K_{ff} has been chosen such that all the roots of $N_C(s) + K_{ff} D_C(s)$ have negative real parts then the design technique presented in section 5.6.2 can be applied.

If the plant with transfer function $F(s)$ is non-minimum phase, meaning that $N_F(s)$ has roots with positive real part, then the situation is more complicated. $N_F(s)$ can then be split into two parts as follows, where polynomial $N_s(s)$ contains the roots of $N_F(s)$ with negative real part (*stable* zeros) and $N_u(s)$ contains the roots of $N_F(s)$ with positive real part (*unstable* zeros):

$$N_F(s) = N_s(s) N_u(s) \quad (5.139)$$

Following Xing & al.⁴, the prefilter $C_{pf}(s)$ can be built as a function of either $\frac{N_u(-s)}{N_s(s)}$ or $\frac{N_u(-s)}{D_1(s)}$, where the roots of $D_1(s)$ are chosen quite far from the desired dominant poles. Polynomial $N_u(-s)$ is the polynomial whose roots are the opposite of the roots of polynomial $N_u(s)$. Those techniques are called Zero Phase Error Compensator (ZPEC) and Modified Zero Phase Error Compensator (MZPEC), respectively.

Alternatively, consider the feedback loop shown in Figure 5.16. The transfer function $\frac{Y(s)}{R_{pf}(s)}$ reads:

$$\frac{Y(s)}{R_{pf}(s)} = \frac{F(s)}{1 + C_1(s)F(s) - C_2(s)} \quad (5.140)$$

In the following, $C_1(s)$ and $C_2(s)$ have the following expressions, where $Q(s)$ is a transfer function to be designed⁵:

$$\begin{cases} C_1(s) = F_n^{-1}(s)Q(s) \\ C_2(s) = Q(s) \end{cases} \quad (5.141)$$

⁴Yankai Xing, Mohammad Pourmahmood Aghababa, Bogdan Marinescu, Florent Xavier. Analysis and Control of Non-Minimum Phase Behavior in Nonlinear AC Grids Equipped with HVDC Links. 2019 IEEE PES Innovative Smart Grid Technologies Europe (ISGT-Europe), Sep 2019, Bucharest, Romania. pp.1-5, 10.1109/ISGTEurope.2019.8905703. hal-02510793

⁵Wang L, Su J. Disturbance rejection control for non-minimum phase systems with optimal disturbance observer. ISA Transactions (2015), <http://dx.doi.org/10.1016/j.isatra.2014.08.003>

Transfer function $F_n(s)$ has the following expression, where polynomial $N_u(-s)$ is the polynomial whose roots are the opposite of the roots of polynomial $N_u(s)$:

$$F(s) = \frac{N_s(s) N_u(s)}{D(s)} \Rightarrow F_n(s) = \frac{N_s(s) N_u(-s)}{D(s)} := F(s) A^{-1}(s) \quad (5.142)$$

All-pass filter $A(s)$ is stable and has the following expression:

$$A^{-1}(s) = \frac{N_u(-s)}{N_u(s)} \Rightarrow A(s) = \frac{N_u(s)}{N_u(-s)} \quad (5.143)$$

Then the transfer function $\frac{Y(s)}{R_{pf}(s)}$ reads:

$$\frac{Y(s)}{R_{pf}(s)} = \frac{F(s)}{1 + F_n^{-1}(s)Q(s)F(s) - Q(s)} := \frac{N_s(s) N_u(-s)}{D(s) T(s)} \quad (5.144)$$

where transfer function $T(s)$ reads as follows

$$T(s) := Q(s) + A^{-1}(s) (1 - Q(s)) \quad (5.145)$$

It is clear that if $T(s)$ is stable then all zeros of $\frac{Y(s)}{R_{pf}(s)}$ will have negative real part. Consequently $Q(s)$ shall be designed such that $F_n^{-1}(s)Q(s)$ is proper and $T(s)$ stable. Let's assume that $Q(s)$ reads as follows, where $N_Q(s)$ and $D_Q(s)$ are polynomials:

$$Q(s) = \frac{N_Q(s)}{D_Q(s)} \quad (5.146)$$

Then transfer function $T(s)$ reads as follows:

$$Q(s) = \frac{N_Q(s)}{D_Q(s)} \Rightarrow T(s) = \frac{N_Q(s) N_u(s) + N_u(-s) (D_Q(s) - N_Q(s))}{D_Q(s) N_u(s)} \quad (5.147)$$

Thus, if we choose $N_Q(s)$ and $D_Q(s)$ such that $D_Q(s) - N_Q(s) = \alpha(s) N_u(s)$, there is a simplification by $N_u(s)$ in $T(s)$ which becomes:

$$D_Q(s) - N_Q(s) = \alpha(s) N_u(s) \Rightarrow T(s) = \frac{N_Q(s) + \alpha(s) N_u(-s)}{D_Q(s)} \quad (5.148)$$

Moreover, if $D_Q(s)$ has all its roots with negative real part then $T(s)$ is stable.

Finally, Internal Model Control and/or Fractional Order Controller may also be used to handle this situation, but this is out of the scope of this lecture.

Example 5.4. We consider the following non-minimum phase transfer function:

$$F(s) = \frac{10s - 1}{s^2 + 1.4s + 1} = \frac{10(s - 0.1)}{s^2 + 1.4s + 1} \quad (5.149)$$

By identification we get:

$$\begin{cases} N_s(s) = 10 \\ N_u(s) = s - 0.1 \\ D(s) = s^2 + 1.4s + 1 \end{cases} \Rightarrow \begin{cases} A^{-1}(s) = \frac{N_u(-s)}{N_u(s)} = \frac{s+0.1}{s-0.1} \\ F_n(s) = F(s)A^{-1}(s) = \frac{-10(s+0.1)}{s^2+1.4s+1} \end{cases} \quad (5.150)$$

Then $Q(s)$ is designed such that $F_n^{-1}(s)Q(s)$ is proper and $T(s)$ stable. In order to render $F_n^{-1}(s)Q(s)$ proper we choose $Q(s)$ for example as follows where a is a design parameter:

$$Q(s) = \frac{1}{s+a} \Rightarrow \begin{cases} N_Q(s) = 1 \\ D_Q(s) = s+a \end{cases} \quad (5.151)$$

Then we choose $N_Q(s)$ and $D_Q(s)$ such that $D_Q(s) - N_Q(s) = \alpha(s)N_u(s)$:

$$D_Q(s) - N_Q(s) = \alpha(s)N_u(s) \Rightarrow s+a-1 = \alpha(s)(s-0.1) \quad (5.152)$$

We finally get:

$$\begin{cases} \alpha(s) = 1 \\ a = 0.9 \end{cases} \Rightarrow T(s) = \frac{N_Q(s) + \alpha(s)N_u(-s)}{D_Q(s)} = \frac{1 + (s+0.1)}{s+0.9} \quad (5.153)$$

and:

$$\frac{Y(s)}{R_{pf}(s)} = \frac{N_s(s)N_u(-s)}{D(s)T(s)} = \frac{10(s+0.1)(s+0.9)}{(s^2+1.4s+1)(s+1.1)} \quad (5.154)$$

We can see that $\frac{Y(s)}{R_{pf}(s)}$ has all its zeros with negative real part, which was the primary objective of the design. Furthermore, $\frac{Y(s)}{R_{pf}(s)}$ is also stable. ■

5.6.4 Design steps

The general scheme for the controlled system is provided in Figure 5.15 where $C(s)$ is the transfer function of the controller, $C_{pf}(s)$ the transfer function of the prefilter and $C_{ff}(s)$ the transfer function of the potential feedforward compensator.

The design philosophy is to set the transfer functions $C(s)$, $C_{pf}(s)$ and possibly $C_{ff}(s)$ in order to force the transfer function of the full system to have the following expression where K_{pf} is a constant gain and $q(s)$ a polynomial formed with the required closed-loop poles:

$$\frac{Y(s)}{R(s)} = \frac{K_{pf}}{q(s)} \quad (5.155)$$

The design steps of the control loop are then the following:

- Design the controller $C(s)$ such that transfer function of feedback loop without prefiltering ($C_{pf}(s) = 1$) and without feedforward compensator ($C_{ff}(s) = 0$) has the required denominator $q(s)$. In other words controller $C(s)$ is used to set the poles of the controlled system.
- In the case where the feedback loop transfer function $\frac{Y(s)}{R_{pf}(s)} = \frac{N_{cl}(s)}{q(s)}$ has zeros with positive real parts, and assuming that $N_F(s)$ has no root in the right half plane, then design the feedforward compensator $C_{ff}(s) = K_{ff}$ such that polynomial $N_C(s) + K_{ff}D_C(s)$ has all its roots with negative real parts.

- Design the prefilter $C_{pf}(s)$ such that transfer function of the full system does not have any zero:

$$\frac{Y(s)}{R(s)} = \frac{K_{pf}}{q(s)} \quad (5.156)$$

In other words prefilter $C_{pf}(s)$ is used to annihilate the zeros of the transfer function of the controlled system.

Example 5.5. Consider a plant with the following transfer function:

$$F(s) = \frac{1}{s(s-2)} \quad (5.157)$$

Obviously the plant is not stable, indeed there is one pole at +2. In order to stabilize the plant we decide to use the following PD controller (we do not use an integral action because the plant $F(s)$ has already an integral term):

$$C(s) = K_p + K_d s \quad (5.158)$$

Furthermore we set the targeted transfer function of the controlled system as follows (see Figure 5.14):

$$\frac{Y(s)}{R(s)} = \frac{2}{s^2 + 5s + 2} \quad (5.159)$$

The first step of the design is to set the values K_p and K_d of the PD controller such that the denominator of the targeted transfer function is achieved. Transfer function $\frac{Y(s)}{R_{pf}(s)}$ (without prefilter) reads:

$$\begin{aligned} \frac{Y(s)}{R_{pf}(s)} &= \frac{C(s)F(s)}{1+C(s)F(s)} \\ &= \frac{K_p + K_d s}{K_p + K_d s + s(s-2)} \\ &= \frac{K_p + K_d s}{s^2 + s(K_d - 2) + K_p} \end{aligned} \quad (5.160)$$

The actual denominator will be equal to the targeted denominator as soon as K_p and K_d are set as follows:

$$s^2 + s(K_d - 2) + K_p = s^2 + 5s + 2 \Rightarrow \begin{cases} K_d = 7 \\ K_p = 2 \end{cases} \quad (5.161)$$

Thus transfer function $\frac{Y(s)}{R_{pf}(s)}$ (without prefilter) reads:

$$\frac{Y(s)}{R_{pf}(s)} = \frac{K_p + K_d s}{s^2 + s(K_d - 2) + K_p} = \frac{2 + 7s}{s^2 + 5s + 2} \quad (5.162)$$

As soon as the zeros $-2/7$ of $\frac{Y(s)}{R_{pf}(s)}$ (without prefilter) have negative real parts there is no need of feedforward compensator $C_{ff}(s)$:

$$C_{ff}(s) = 0 \quad (5.163)$$

Taking now into account prefilter $C_{pf}(s)$ transfer function $\frac{Y(s)}{R(s)}$ reads:

$$\frac{Y(s)}{R(s)} = \frac{R_{pf}(s)}{R(s)} \frac{Y(s)}{R_{pf}(s)} = C_{pf}(s) \frac{2 + 7s}{s^2 + 5s + 2} \quad (5.164)$$

Thus transfer function of the controlled system will read $\frac{Y(s)}{R(s)} = \frac{2}{s^2 + 5s + 2}$ as soon as prefilter $C_{pf}(s)$ is set as follows:

$$\frac{Y(s)}{R(s)} = \frac{2}{s^2 + 5s + 2} \Rightarrow C_{pf}(s) = \frac{2}{2 + 7s} \quad (5.165)$$

■

5.7 Algebraic controller design

5.7.1 Polynomial product

Let $A(s)$ be a polynomial of degree n and $B(s)$ be a polynomial of degree m :

$$\begin{cases} A(s) = a_0 + a_1s + \dots + a_ns^n \\ B(s) = b_0 + b_1s + \dots + b_ms^m \end{cases} \quad (5.166)$$

Then the product $A(s)B(s)$ is a polynomial $C(s)$ of degree $n + m$:

$$C(s) = A(s)B(s) := c_0 + c_1s + \dots + c_{n+m}s^{n+m} \quad (5.167)$$

Coefficients c_0, c_1, \dots, c_{n+m} of polynomial $C(s)$ can be obtained as follows:

$$\begin{bmatrix} c_0 \\ c_1 \\ \vdots \\ c_{n+m} \end{bmatrix} = \mathbf{T}_B \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_n \end{bmatrix} \quad (5.168)$$

where Toeplitz matrix \mathbf{T}_B is the following $(n + m + 1) \times (n + 1)$ matrix which is obtained by repeatedly shifting to the bottom its first column:

$$\mathbf{T}_B = \begin{bmatrix} b_0 & 0 & \dots & 0 & 0 \\ b_1 & b_0 & & \vdots & \vdots \\ \vdots & \vdots & \ddots & & \\ b_m & b_{m-1} & & \ddots & \\ 0 & b_m & \ddots & & \\ \vdots & 0 & & \vdots & \vdots \\ \vdots & \vdots & & b_m & b_{m-1} \\ 0 & 0 & \dots & 0 & b_m \end{bmatrix} \quad (5.169)$$

This result can be achieved by matching the coefficients of like power of s in the development of $A(s)B(s)$.

Alternatively, coefficients c_0, c_1, \dots, c_{n+m} of polynomial $C(s)$ can be obtained as follows:

$$\begin{bmatrix} c_0 \\ c_1 \\ \vdots \\ c_{n+m} \end{bmatrix} = \mathbf{T}_A \begin{bmatrix} b_0 \\ b_1 \\ \vdots \\ b_m \end{bmatrix} \quad (5.170)$$

where Toeplitz matrix \mathbf{T}_A is the following $(n+m+1) \times (m+1)$ matrix which is obtained by repeatedly shifting to the bottom its first column:

$$\mathbf{T}_A = \begin{bmatrix} a_0 & 0 & \cdots & 0 & 0 \\ a_1 & a_0 & & \vdots & \vdots \\ \vdots & \vdots & \ddots & & \\ a_n & a_{n-1} & & & \\ 0 & a_n & \ddots & & \\ \vdots & 0 & & \vdots & \vdots \\ \vdots & \vdots & & a_n & a_{n-1} \\ 0 & 0 & \cdots & 0 & a_n \end{bmatrix} \quad (5.171)$$

Moreover, let $A(s)$ and $X(s)$ be polynomials of degree n_1 and m_1 , respectively, and $B(s)$ and $Y(s)$ be polynomials of degree n_2 and m_2 , respectively:

$$\begin{cases} A(s) = a_0 + a_1s + \cdots + a_{n_1}s^{n_1} \\ X(s) = x_0 + x_1s + \cdots + x_{m_1}s^{m_1} \\ B(s) = b_0 + b_1s + \cdots + b_{n_2}s^{n_2} \\ Y(s) = y_0 + y_1s + \cdots + y_{m_2}s^{m_2} \end{cases} \quad (5.172)$$

Furthermore assume that $n_1 + m_1 \geq n_2 + m_2$. Then the product $A(s)X(s) + B(s)Y(s)$ is a polynomial $C(s)$ of degree $n_1 + m_1$:

$$\begin{aligned} n_1 + m_1 &\geq n_2 + m_2 \\ \Rightarrow C(s) &= A(s)X(s) + B(s)Y(s) := q_0 + q_1s + \cdots + q_{n_1+m_1}s^{n_1+m_1} \end{aligned} \quad (5.173)$$

Coefficients $q_0, q_1, \dots, q_{n_1+m_1}$ of polynomial $C(s)$ can be obtained as follows:

$$\begin{bmatrix} q_0 \\ q_1 \\ \vdots \\ q_{n_1+m_1} \end{bmatrix} = \mathbf{S}_{XY} \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_{n_1} \\ \frac{a_{n_1}}{b_0} \\ b_1 \\ \vdots \\ b_{n_2} \end{bmatrix} \quad (5.174)$$

where Sylvester matrix \mathbf{S}_{XY} is the following $(n_1 + m_1 + 1) \times (n_1 + 1 + n_2 + 1)$ matrix which is obtained by shifting successively to the bottom coefficients x_i

and then coefficients y_j :

$$\mathbf{S}_{XY} = \left[\begin{array}{cccc|cccc} x_0 & 0 & \cdots & 0 & y_0 & 0 & \cdots & 0 \\ x_1 & x_0 & & \vdots & y_1 & y_0 & & \vdots \\ \vdots & \vdots & \ddots & & \vdots & \vdots & \ddots & \\ x_{m_1} & x_{m_1-1} & & x_0 & y_{m_2} & & & \\ 0 & x_{m_1} & \ddots & & 0 & y_{m_2} & \ddots & \\ \vdots & 0 & & \vdots & \vdots & 0 & & \vdots \\ \vdots & \vdots & & & \vdots & \vdots & y_{m_2} & y_{m_2-1} \\ 0 & 0 & \cdots & 0 & 0 & \cdots & 0 & y_{m_2} \end{array} \right] \left. \vphantom{\begin{array}{c} \\ \\ \\ \\ \\ \\ \\ \end{array}} \right\} n_1 + m_1 + 1 \text{ rows}$$

$\underbrace{\hspace{15em}}_{n_1+1 \text{ columns}} \quad \underbrace{\hspace{15em}}_{n_2+1 \text{ columns}}$

(5.175)

Alternatively, coefficients $q_0, q_1, \dots, q_{n_1+m_1}$ of polynomial $C(s)$ can be obtained as follows:

$$\begin{bmatrix} q_0 \\ q_1 \\ \vdots \\ q_{n_1+m_1} \end{bmatrix} = \mathbf{S}_{AB} \begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_{m_1} \\ y_0 \\ y_1 \\ \vdots \\ y_{m_2} \end{bmatrix} \quad (5.176)$$

where Sylvester matrix \mathbf{S}_{AB} is the following $(n_1 + m_1 + 1) \times (m_1 + 1 + m_2 + 1)$ matrix which is obtained by shifting successively to the bottom coefficients a_i and then coefficients x_j :

$$\mathbf{S}_{AB} = \left[\begin{array}{cccc|cccc} a_0 & 0 & \cdots & 0 & b_0 & 0 & \cdots & 0 \\ a_1 & a_0 & & \vdots & b_1 & b_0 & & \vdots \\ \vdots & \vdots & \ddots & & \vdots & \vdots & \ddots & \\ a_{n_1} & a_{n_1-1} & & a_0 & b_{n_2} & & & \\ 0 & a_{n_1} & \ddots & & 0 & b_{n_2} & \ddots & \\ \vdots & 0 & & \vdots & \vdots & 0 & & \vdots \\ \vdots & \vdots & & & \vdots & \vdots & b_{n_2} & b_{n_2-1} \\ 0 & 0 & \cdots & 0 & 0 & \cdots & 0 & b_{n_2} \end{array} \right] \left. \vphantom{\begin{array}{c} \\ \\ \\ \\ \\ \\ \\ \end{array}} \right\} n_1 + m_1 + 1 \text{ rows}$$

$\underbrace{\hspace{15em}}_{m_1+1 \text{ columns}} \quad \underbrace{\hspace{15em}}_{m_2+1 \text{ columns}}$

(5.177)

5.7.2 Sylvester's Theorem

Consider the two polynomials:

$$\begin{cases} A(s) = a_0 + a_1s + \cdots + a_{n-1}s^{n-1} + a_ns^n \\ B(s) = b_0 + b_1s + \cdots + b_{m-1}s^{m-1} + b_ms^m \end{cases} \quad (5.178)$$

Polynomials $A(s)$ and $B(s)$ are said to be *coprime* if they have no common root. In other words, there is no polynomial $R(s) \neq 1$ such that:

$$\begin{cases} A(s) = Y(s)R(s) \text{ s.t. } \deg(Y(s)) < n \\ B(s) = X(s)R(s) \text{ s.t. } \deg(X(s)) < m \end{cases} \quad (5.179)$$

The preceding equation can be rewritten as follows:

$$\frac{A(s)}{B(s)} = \frac{Y(s)}{X(s)} \text{ s.t. } \begin{cases} \deg(Y(s)) < \deg(A(s)) \\ \deg(X(s)) < \deg(B(s)) \end{cases} \quad (5.180)$$

or, equivalently:

$$A(s)X(s) + B(s)(-Y(s)) = 0 \text{ s.t. } \begin{cases} \deg(Y(s)) < \deg(A(s)) = n \\ \deg(X(s)) < \deg(B(s)) = m \end{cases} \quad (5.181)$$

Let \mathbf{S}_{AB} be the following square matrix, also called *Sylvester resultant* matrix:

$$\mathbf{S}_{AB} = \left[\begin{array}{cccccc} a_0 & 0 & \cdots & b_0 & 0 & \cdots \\ a_1 & a_0 & \ddots & b_1 & b_0 & \ddots \\ \vdots & a_1 & \ddots & \vdots & b_1 & \ddots \\ a_{n-1} & \vdots & & b_{m-1} & \vdots & \\ a_n & a_{n-1} & & b_m & b_{m-1} & \\ 0 & a_n & \ddots & 0 & b_m & \ddots \\ \vdots & \ddots & \ddots & \vdots & \ddots & \ddots \end{array} \right] \left. \vphantom{\begin{array}{cccccc} a_0 & 0 & \cdots & b_0 & 0 & \cdots \\ a_1 & a_0 & \ddots & b_1 & b_0 & \ddots \\ \vdots & a_1 & \ddots & \vdots & b_1 & \ddots \\ a_{n-1} & \vdots & & b_{m-1} & \vdots & \\ a_n & a_{n-1} & & b_m & b_{m-1} & \\ 0 & a_n & \ddots & 0 & b_m & \ddots \\ \vdots & \ddots & \ddots & \vdots & \ddots & \ddots \end{array}} \right\} \begin{array}{l} n+m \text{ rows} \\ m \text{ columns} \quad n \text{ columns} \end{array} \quad (5.182)$$

Then it can be shown that $A(s)$ and $B(s)$ are coprime (i.e. they have no common roots) if and only if:

$$\det(\mathbf{S}_{AB}) \neq 0 \quad (5.183)$$

This property indicates that polynomial equation $A(s)X(s) - B(s)Y(s) = 0$, or equivalently $A(s)X(s) + B(s)Y_1(s) = 0$ with $Y_1(s) := -Y(s)$, has no polynomial solution $X(s)$ and $Y(s)$ as soon as $A(s)$ and $B(s)$ are coprime.

Furthermore if $A(s)$ and $B(s)$ are coprime, then there exists unique polynomials $X(s)$ and $Y(s)$ with $\deg(X(s)) < \deg(B(s)) = m$ and $\deg(Y(s)) < \deg(A(s)) = n$ such that the following Bézout equation holds:

$$A(s)X(s) + B(s)Y(s) = 1 \text{ s.t. } \begin{cases} \deg(Y(s)) < \deg(A(s)) = n \\ \deg(X(s)) < \deg(B(s)) = m \end{cases} \quad (5.184)$$

One way to solve a Bézout equation is to turn it into a set of $n + m$ linear equations with $n + m$ unknowns. Let:

$$\begin{cases} X(s) = x_0 + x_1s + \cdots + x_{m-1}s^{m-1} \\ Y(s) = y_0 + y_1s + \cdots + y_{n-1}s^{n-1} \end{cases} \quad (5.185)$$

Then the set of linear equation reads:

$$\mathbf{S}_{AB} \begin{bmatrix} x_0 \\ \vdots \\ x_{m-1} \\ y_0 \\ \vdots \\ y_{n-1} \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \quad (5.186)$$

Alternatively, given a polynomial $q(s)$ of degree $n + m - 1$, assume that we are looking for polynomials $X(s)$ and $Y(s)$ such that:

$$A(s)X(s) + B(s)Y(s) = q(s) \quad (5.187)$$

Assuming that $A(s)$ and $B(s)$ are coprime, the set of linear equation to be solved reads as follows:

$$\begin{aligned} q(s) &= q_0 + q_1s + \cdots + q_{n+m-1}s^{n+m-1} \\ \Rightarrow \mathbf{S}_{AB} \begin{bmatrix} x_0 \\ \vdots \\ x_{m-1} \\ y_0 \\ \vdots \\ y_{n-1} \end{bmatrix} &= \begin{bmatrix} q_0 \\ q_1 \\ \vdots \\ q_{n+m-1} \end{bmatrix} \end{aligned} \quad (5.188)$$

Furthermore, it is worth noticing that the following relation also holds⁶:

$$A(s)X(s) + B(s)Y(s) = \begin{bmatrix} a_0 & a_1 & \cdots & a_n & b_0 & b_1 & \cdots & b_m \end{bmatrix} \begin{bmatrix} X(s) \\ sX(s) \\ \vdots \\ \frac{s^n X(s)}{Y(s)} \\ sY(s) \\ \vdots \\ s^m Y(s) \end{bmatrix} \quad (5.189)$$

⁶William A. Wolovich, Automatic Control Systems: Basic Analysis and Design, Saunders College Publishing, 1995

Example 5.6. Let $A(s)$ and $B(s)$ be the following polynomials:

$$\begin{cases} A(s) = s(s-1) = s^2 - s \\ B(s) = s - 2 \end{cases} \quad (5.190)$$

It is straightforward that $A(s)$ and $B(s)$ are coprime polynomials because they have no common root.

To check this result, we compute the Sylvester resultant matrix \mathbf{S}_{AB} , which is here a square 3×3 matrix:

$$\mathbf{S}_{AB} = \left[\begin{array}{ccc|ccc} a_0 & & & b_0 & & 0 \\ & a_1 & & & b_0 & \\ & & a_2 & & & b_0 \\ \hline & & & 0 & & b_1 \\ & & & & 0 & \\ & & & & & \end{array} \right] \left. \vphantom{\begin{array}{ccc|ccc} a_0 & & & b_0 & & 0 \\ & a_1 & & & b_0 & \\ & & a_2 & & & b_0 \\ \hline & & & 0 & & b_1 \\ & & & & 0 & \\ & & & & & \end{array}} \right\} \begin{array}{l} 2+1 \text{ rows} \\ \underbrace{\hspace{10em}}_{\substack{1 \text{ column} \quad 2 \text{ columns}}} \end{array} = \begin{bmatrix} 0 & -2 & 0 \\ -1 & 1 & -2 \\ 1 & 0 & 1 \end{bmatrix} \quad (5.191)$$

It can be seen that $\det(\mathbf{S}_{AB}) \neq 0$, thus $A(s)$ and $B(s)$ are coprime polynomials:

$$\det(\mathbf{S}_{AB}) = \det \left(\begin{bmatrix} 0 & -2 & 0 \\ -1 & 1 & -2 \\ 1 & 0 & 1 \end{bmatrix} \right) = 2 \neq 0 \quad (5.192)$$

As far as $A(s)$ and $B(s)$ are coprime, then there exists unique polynomials $X(s)$ and $Y(s)$ with $\deg(X(s)) < m$ and $\deg(Y(s)) < n$ such that the following Bézout equation holds:

$$A(s)X(s) + B(s)Y(s) = 1 \quad (5.193)$$

One way to solve a Bézout equation is to turn it into a set of $n+m = 3$ linear equations with $n+m = 3$ unknowns. Let:

$$\begin{cases} X(s) = x_0 \\ Y(s) = y_0 + y_1s \end{cases} \quad (5.194)$$

Then the set of linear equation reads:

$$\mathbf{S}_{AB} \begin{bmatrix} x_0 \\ y_0 \\ y_1 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \Leftrightarrow \begin{bmatrix} 0 & -2 & 0 \\ -1 & 1 & -2 \\ 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_0 \\ y_0 \\ y_1 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \quad (5.195)$$

We finally get:

$$\begin{bmatrix} x_0 \\ y_0 \\ y_1 \end{bmatrix} = \begin{bmatrix} 0.5 \\ -0.5 \\ -0.5 \end{bmatrix} \Rightarrow \begin{cases} X(s) = x_0 = 0.5 \\ Y(s) = y_0 + y_1s = -0.5 - 0.5s \end{cases} \quad (5.196)$$

■

5.7.3 One degree of freedom controller design

A general form of a one degree of freedom (abbreviated as 1DOF) control system with prefiltering is shown in Figure 5.14 where controller $C(s)$ is not a PID controller but a very general controller with a proper transfer function of *relative degree* $r \geq 0^7$ whose denominator $D_C(s)$ is a *monic polynomial*⁸ of degree m :

$$C(s) = \frac{N_C(s)}{D_C(s)} := \frac{y_0 + y_1s + \cdots + y_{m-r}s^{m-r}}{x_0 + x_1s + \cdots + x_{m-1}s^{m-1} + 1 \times s^m} \quad (5.197)$$

In the following the transfer function $F(s)$ of the plant is written in terms of the ratio between its numerator $N_F(s)$ and its denominator $D_F(s)$, which is also assumed to be a *monic polynomial*:

$$F(s) = \frac{N_F(s)}{D_F(s)} \quad \text{where} \quad \begin{cases} D_F(s) \text{ monic polynomial} \\ \deg(N_F(s)) \leq \deg(D_F(s)) = n \end{cases} \quad (5.198)$$

Thus the transfer function of the controlled system without prefiltering ($C_{pf}(s) = 1$) reads:

$$\frac{Y(s)}{R_{pf}(s)} = \frac{C(s)F(s)}{1 + C(s)F(s)} = \frac{N_C(s)N_F(s)}{N_C(s)N_F(s) + D_C(s)D_F(s)} \quad (5.199)$$

The coefficients of polynomials $N_C(s)$ and $D_C(s)$ are then set in order that the characteristic polynomial of the transfer function $\frac{Y(s)}{R_{pf}(s)}$ matches with some predefined polynomial $q(s)$: it could be either the ITAE polynomial defined in Table 5.2 or another polynomial which set the poles of the closed-loop system.

$$N_C(s)N_F(s) + D_C(s)D_F(s) := q(s) \quad (5.200)$$

This equation is known as a Diophantine equation. Denoting by m the degree of $D_C(s)$ and by n the degree of $D_F(s)$, and using the fact that the degree of $N_C(s)N_F(s)$ is lower or equal than the degree of $D_C(s)D_F(s)$ (because $F(s)$ and $C(s)$ are both proper transfer functions), we conclude that polynomial $q(s) := N_C(s)N_F(s) + D_C(s)D_F(s)$ is of degree $n + m$. Because $D_C(s)$ and $D_F(s)$ are *monic polynomials*, we conclude that polynomial $q(s)$ is also a *monic polynomial* with $n + m$ coefficients.

On the other hand controller $C(s)$ is formed with $m - r + 1$ unknown coefficients in its numerator (y_0, y_1, \dots, y_{m-r}) and m unknown coefficients in its denominator (x_0, x_1, \dots, x_{m-1}), which forms a set of $2m - r + 1$ unknown coefficients.

In order to be able to identify all the coefficients of s^i in (5.200) the following equation shall be satisfied:

$$n + m = 2m - r + 1 \Rightarrow m = n + r - 1 \quad (5.201)$$

⁷The *relative degree* is the difference between the degree of the denominator (number of poles) and degree of the numerator (number of zeros)

⁸A *monic polynomial* is a polynomial whose leading coefficient, that is the nonzero coefficient of highest degree, is equal to 1

Consequently as soon as the predefined *monic polynomial* $q(s)$ is a of degree $n + m = 2n + r - 1$, Diophantine equation (5.200) can be solved.

To summarize, the denominator $q(s)$ of the closed-loop system as well as the numerator $N_C(s)$ and the denominator $D_C(s)$ of the controller $C(s)$, whose *relative degree* is set to r , shall be set as follows:

$$\Rightarrow \begin{cases} D_F(s) \text{ monic polynomial of degree } n \\ C(s) \text{ controller of relative degree } r \\ q(s) = 1 \times s^{2n+r-1} + q_{2n+r-2}s^{2n+r-2} + \dots + q_1s + q_0 \\ N_C(s) = y_{n-1}s^{n-1} + \dots + y_1s + y_0 \\ D_C(s) = 1 \times s^{n+r-1} + x_{n+r-2}s^{n+r-2} + \dots + x_1s + x_0 \end{cases} \quad (5.202)$$

In addition coefficients x_i of $D_C(s)$ and y_j of $N_C(s)$ can be obtained by:

$$\mathbf{S}_{AB} \begin{bmatrix} x_0 \\ \vdots \\ x_{n+r-2} \\ 1 \\ y_0 \\ \vdots \\ y_{n-1} \end{bmatrix} = \begin{bmatrix} q_0 \\ q_1 \\ \vdots \\ q_{2n+r-2} \\ 1 \end{bmatrix} \quad (5.203)$$

Let:

$$\begin{cases} N_F(s) = b_0 + b_1s + \dots + b_ms^m \\ D_F(s) = a_0 + a_1s + \dots + a_{n-1}s^{n-1} + 1 \times s^n \end{cases} \quad (5.204)$$

Then \mathbf{S}_{AB} is the following Sylvester resultant (square) matrix:

$$\mathbf{S}_{AB} = \left[\begin{array}{ccc|ccc} a_0 & 0 & \dots & b_0 & 0 & \dots \\ a_1 & a_0 & \ddots & b_1 & b_0 & \ddots \\ \vdots & a_1 & \ddots & \vdots & b_1 & \ddots \\ a_{n-1} & \vdots & & b_{m-1} & \vdots & \\ 1 & a_{n-1} & & b_m & b_{m-1} & \\ 0 & 1 & \ddots & 0 & b_m & \ddots \\ \vdots & \ddots & \ddots & \vdots & \ddots & \ddots \end{array} \right] \left. \vphantom{\begin{array}{ccc|ccc} \right\} 2n + r \text{ rows} \quad (5.205)$$

$\underbrace{\hspace{10em}}_{n+r \text{ columns}} \quad \underbrace{\hspace{10em}}_{n \text{ columns}}$

Example 5.7. Consider the case where the numerator $N_F(s)$ and the denominator $D_F(s)$ of transfer function $F(s)$ read as follows:

$$\begin{cases} N_F(s) = b_2s^2 + b_1s + b_0 \\ D_F(s) = s^3 + a_2s^2 + a_1s + a_0 \end{cases} \quad (5.206)$$

We seek a controller $C(s)$ of relative degree $r = 0$. Because $D_F(s)$ is of degree $n = 3$, the algebraic controller $C(s)$ is composed of the ratio of 2 polynomials

$N_C(s)$ and $D_C(s)$ of degree $n - 1 = 2$ and $n + r - 1 = 2$, respectively:

$$C(s) = \frac{N_C(s)}{D_C(s)} \text{ where } \begin{cases} N_C(s) = y_2 s^2 + y_1 s + y_0 \\ D_C(s) = s^2 + x_1 s + x_0 \end{cases} \quad (5.207)$$

The denominator $N_F(s)N_C(s) + D_F(s)D_C(s)$ of the closed-loop transfer function is a polynomial of degree $2n + r - 1 = 5$. Using the results on the product of polynomials presented in 5.7.1 we get:

$$N_F(s)N_C(s) + D_F(s)D_C(s) = q_0 + q_1 s + q_2 s^2 + q_3 s^3 + q_4 s^4 + q_5 s^5 \quad (5.208)$$

where:

$$\begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \\ q_4 \\ q_5 \end{bmatrix} = \begin{bmatrix} a_0 & 0 & 0 & | & b_0 & 0 & 0 \\ a_1 & a_0 & 0 & | & b_1 & b_0 & 0 \\ a_2 & a_1 & a_0 & | & b_2 & b_1 & b_0 \\ 1 & a_2 & a_1 & | & 0 & b_2 & b_1 \\ 0 & 1 & a_2 & | & 0 & 0 & b_2 \\ 0 & 0 & 1 & | & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ \frac{1}{y_0} \\ y_1 \\ y_2 \end{bmatrix} \quad (5.209)$$

We finally get:

$$\begin{aligned} N_F(s)N_C(s) + D_F(s)D_C(s) &= (b_2 s^2 + b_1 s + b_0)(y_2 s^2 + y_1 s + y_0) \\ &\quad + (s^3 + a_2 s^2 + a_1 s + a_0)(s^2 + x_1 s + x_0) \\ &= 1 \times s^5 + (x_1 + a_2 + b_2 y_2) s^4 \\ &\quad + (x_0 + a_2 x_1 + a_1 + b_2 y_1 + b_1 y_2) s^3 \\ &\quad + (a_2 x_0 + a_1 x_1 + a_0 + b_2 y_0 + b_1 y_1 + b_0 y_2) s^2 \\ &\quad + (a_1 x_0 + a_0 x_1 + b_1 y_0 + b_0 y_1) s + a_0 x_0 + b_0 y_0 \end{aligned} \quad (5.210)$$

On the other hand, let λ_i , $i = 1, \dots, 2n + r - 1 = 5$ be the desired closed-loop poles. We can write:

$$\begin{aligned} q(s) &= \prod_{i=1}^{2n+r-1=5} (s - \lambda_i) \\ &= (s - \lambda_1)(s - \lambda_2)(s - \lambda_3)(s - \lambda_4)(s - \lambda_5) \\ &= s^5 + \sum_{p=0}^4 q_p s^p \end{aligned} \quad (5.211)$$

where:

$$q_{5-k} = (-1)^k \sum_{1 \leq i_1 < \dots < i_k \leq 5} \lambda_{i_1} \dots \lambda_{i_k} \text{ where } k = 1, \dots, 5 \quad (5.212)$$

Once the coefficients of s^k of $q(s)$ are identified, we get in a matrix form:

$$\begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \\ q_4 \\ 1 \end{bmatrix} = \begin{bmatrix} a_0 & 0 & 0 & | & b_0 & 0 & 0 \\ a_1 & a_0 & 0 & | & b_1 & b_0 & 0 \\ a_2 & a_1 & a_0 & | & b_2 & b_1 & b_0 \\ 1 & a_2 & a_1 & | & 0 & b_2 & b_1 \\ 0 & 1 & a_2 & | & 0 & 0 & b_2 \\ 0 & 0 & 1 & | & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ \frac{1}{y_0} \\ y_1 \\ y_2 \end{bmatrix} \quad (5.213)$$

Once this system inverted we get:

$$\begin{bmatrix} x_0 \\ x_1 \\ \frac{1}{y_0} \\ y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} a_0 & 0 & 0 & | & b_0 & 0 & 0 \\ a_1 & a_0 & 0 & | & b_1 & b_0 & 0 \\ a_2 & a_1 & a_0 & | & b_2 & b_1 & b_0 \\ 1 & a_2 & a_1 & | & 0 & b_2 & b_1 \\ 0 & 1 & a_2 & | & 0 & 0 & b_2 \\ 0 & 0 & 1 & | & 0 & 0 & 0 \end{bmatrix}^{-1} \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \\ q_4 \\ 1 \end{bmatrix} \quad (5.214)$$

■

5.7.4 Prefilter design

Once $C(s)$ is set then the transfer function $C_{pf}(s)$ of the prefilter can be computed. For example it can be computed in order to match with the required ITAE transfer function:

$$\frac{Y(s)}{R(s)} = \frac{N_{cl}(s)}{q(s)} \quad (5.215)$$

Thus from the following relations:

$$\begin{cases} \frac{Y(s)}{R(s)} = \frac{N_{cl}(s)}{q(s)} \\ \frac{Y(s)}{R(s)} = C_{pf}(s) \frac{Y(s)}{R_{pf}(s)} \\ \frac{Y(s)}{R_{pf}(s)} = \frac{N_C(s)N_F(s)}{N_C(s)N_F(s) + D_C(s)D_F(s)} = \frac{N_C(s)N_F(s)}{q(s)} \end{cases} \quad (5.216)$$

We get:

$$C_{pf}(s) = \frac{Y(s)}{R(s)} \cdot \frac{R_{pf}(s)}{Y(s)} = \frac{N_{cl}(s)}{q(s)} \cdot \frac{q(s)}{N_C(s)N_F(s)} = \frac{N_{cl}(s)}{N_C(s)N_F(s)} \quad (5.217)$$

It is worth noticing that the design is possible only if all the zeros of $F(s)C(s)$, that are the roots of $N_C(s)N_F(s)$, have negative real parts so that the prefilter $C_{pf}(s)$ is stable.

Consequently the design strategy to compute the transfer function $C(s)$ of the controller and the transfer function $C_{pf}(s)$ of the prefilter is the following:

- Find the transfer function $C(s)$ to give the characteristic polynomial of $\frac{Y(s)}{R_{pf}(s)}$ of the form given in Table 5.2:

$$\frac{Y(s)}{R_{pf}(s)} = \frac{F(s)C(s)}{1 + F(s)C(s)} = \frac{N_C(s)N_F(s)}{q(s)} \quad (5.218)$$

- Design the prefilter $C_{pf}(s)$ to cancel the unwanted terms in the numerator of the transfer function $\frac{Y(s)}{R_{pf}(s)}$ so that the closed-loop transfer function $\frac{Y(s)}{R(s)} = \frac{N_{cl}(s)}{q(s)}$ matches with the ITAE transfer function:

$$C_{pf}(s) = \frac{N_{cl}(s)}{N_C(s)N_F(s)} \quad (5.219)$$

The transfer function $C_{pf}(s)$ of prefilter may be reduced by using for example the technique presented in Section 3.6.

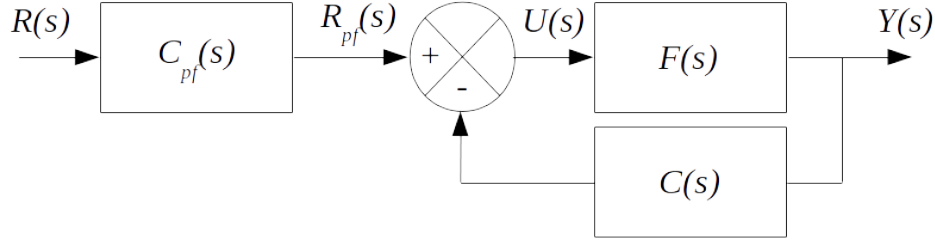


Figure 5.17: Controller $C(s)$ in the feedback loop and prefilter $C_{pf}(s)$

Example 5.8. Consider an unstable plant with the following transfer function:

$$F(s) = \frac{N_F(s)}{D_F(s)} = \frac{1}{(s-1)(s+2)} \quad (5.220)$$

The specifications are the following:

- Settling time t_s about 1 seconds
- Overshoot D of 5%
- The position error shall be zero

An algebraic controller $C(s)$ with relative degree $r = 1$ as well as a prefilter $C_{pf}(s)$ will be designed to achieve those specifications. We decide to put controller $C(s)$ in the feedback loop as shown in Figure 5.17.

From the specifications we compute the location of the dominant poles as follows:

$$\{\lambda_{dp}, \bar{\lambda}_{dp}\} = -m\omega_0 \pm j\omega_0\sqrt{1-m^2} \text{ where } \begin{cases} m = \frac{-\ln(D)}{\sqrt{\pi^2 + \ln^2(D)}} \\ t_s = \frac{4}{m\omega_0} \Leftrightarrow \omega_0 = \frac{4}{m t_s} \end{cases} \quad (5.221)$$

We get:

$$\begin{cases} m \approx 0.69 \\ \omega_0 \approx 5.8 \end{cases} \Rightarrow \{\lambda_{dp}, \bar{\lambda}_{dp}\} \approx -4 \pm j4.2 \quad (5.222)$$

The degree of the denominator of $F(s)$ is $n = 2$ and we impose the relative degree of controller $C(s)$, $r = 1$. Thus the degree of denominator of the closed-loop transfer function is $2n + r - 1 = 4$. So in addition to the 2 dominant poles $\{\lambda_{dp}, \bar{\lambda}_{dp}\}$ we have to choose two non-dominant poles faster than $\{\lambda_{dp}, \bar{\lambda}_{dp}\}$, for example $\lambda_f = -20 + j$ and $\bar{\lambda}_f = -20 - j$. So $q(s)$ reads:

$$\begin{aligned} \lambda_f = -20 \Rightarrow q(s) &= (s - \lambda_{dp})(s - \bar{\lambda}_{dp})(s - \lambda_f)(s - \bar{\lambda}_f) \\ &= (s^2 + 2m\omega_0 s + \omega_0^2)(s^2 + 40s + 401) \\ &\approx s^4 + 48s^3 + 754.64s^2 + 4553.6s + 13489.64 \end{aligned} \quad (5.223)$$

Let $C(s) = \frac{N_C(s)}{D_C(s)}$ where $N_C(s)$ is a polynomial of degree $n - 1 = 1$ and $D_C(s)$ is a monic polynomial of degree $n + r - 1 = 2$:

$$C(s) = \frac{N_C(s)}{D_C(s)} = \frac{y_1s + y_0}{s^2 + x_1s + x_0} \quad (5.224)$$

The denominator $N_C(s)N_F(s) + D_C(s)D_F(s)$ of the closed-loop transfer function $\frac{Y(s)}{R_{pf}(s)}$ reads:

$$N_C(s)N_F(s) + D_C(s)D_F(s) = q_0 + q_1s + q_2s^2 + q_3s^3 + s^4 \quad (5.225)$$

where:

$$\begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \\ 1 \end{bmatrix} = \begin{bmatrix} -2 & 0 & 0 & | & 1 & 0 \\ 1 & -2 & 0 & | & 0 & 1 \\ 1 & 1 & -2 & | & 0 & 0 \\ 0 & 1 & 1 & | & 0 & 0 \\ 0 & 0 & 1 & | & 0 & 0 \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ \frac{1}{y_0} \\ y_0 \\ y_1 \end{bmatrix} \quad (5.226)$$

We get:

$$\begin{aligned} N_C(s)N_F(s) + D_C(s)D_F(s) &= (y_1s + y_0) + (s^2 + x_1s + x_0)(s^2 + s - 2) \\ &= s^4 + s^3(1 + x_1) + s^2(x_0 + x_1 - 2) \\ &\quad + s(x_0 - 2x_1 + y_1) - 2x_0 + y_0 \end{aligned} \quad (5.227)$$

Identifying $N_C(s)N_F(s) + D_C(s)D_F(s)$ with $q(s)$ leads to the values of coefficients x_i and y_i :

$$\begin{cases} 1 + x_1 \approx 48 \\ x_0 + x_1 - 2 \approx 754.64 \\ x_0 - 2x_1 + y_1 \approx 4553.6 \\ -2x_0 + y_0 \approx 13489.64 \end{cases} \Rightarrow \begin{cases} x_1 \approx 47 \\ x_0 \approx 709.64 \\ y_1 \approx 3937.96 \\ y_0 \approx 14908.92 \end{cases} \quad (5.228)$$

Thus $C(s)$ has the following expression:

$$C(s) = \frac{y_1s + y_0}{s^2 + x_1s + x_0} \approx \frac{3937.96s + 14908.92}{s^2 + 47s + 709.64} \quad (5.229)$$

The unit step response of the closed-loop transfer function $\frac{Y(s)}{R_{pf}(s)}$ is drawn in Figure 5.18. We notice that the time response comply with the specifications but neither the overshoot nor the position error.

The prefilter $C_{pf}(s)$ is designed to overcome those drawbacks. Closed loop transfer function $Y(s)/R_{pf}(s)$ reads:

$$\begin{aligned} \frac{Y(s)}{R_{pf}(s)} &= \frac{N_F(s)}{(s - \lambda_{dp})(s - \bar{\lambda}_{dp})(s - \lambda_f)(s - \bar{\lambda}_f)} \\ &\approx \frac{s^2 + 47s + 709.64}{s^4 + 48s^3 + 754.64s^2 + 4553.6s + 13489.64} \end{aligned} \quad (5.230)$$

where $\{\lambda_{dp}, \bar{\lambda}_{dp}\}$ are the desired dominant poles and $\{\lambda_f, \bar{\lambda}_f\}$ is the fast poles of the loop.

As far as the roots of $N_F(s)$ lies in the left half plane, and in order to achieve a null position error, the transfer function of the prefilter $C_{pf}(s)$ is designed as follows:

$$C_{pf}(s) = \frac{\lambda_{dp}\bar{\lambda}_{dp}\lambda_f\bar{\lambda}_f}{N_F(s)} \approx \frac{13489.64}{s^2 + 47s + 709.64} \quad (5.231)$$

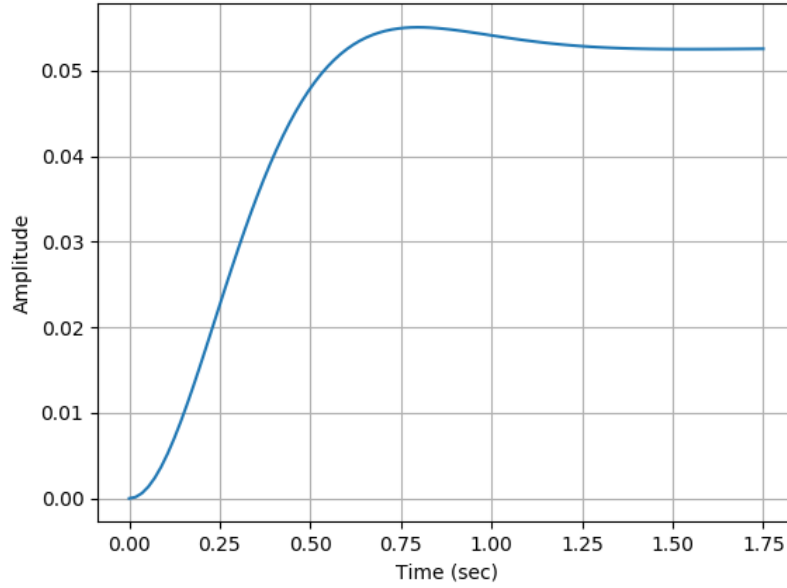


Figure 5.18: Unit step response without prefilter

The closed-loop transfer function finally reads:

$$\begin{aligned}
 \frac{Y(s)}{R(s)} &= C_{pf}(s) \times \frac{Y(s)}{R_{pf}(s)} \\
 &= \frac{\lambda_{dp} \bar{\lambda}_{dp} \lambda_f \bar{\lambda}_f}{(s - \lambda_{dp})(s - \bar{\lambda}_{dp})(s - \lambda_f)} \\
 &\approx \frac{13489.64}{s^4 + 48s^3 + 754.64s^2 + 4553.6s + 13489.64}
 \end{aligned} \tag{5.232}$$

The unit step response of the closed-loop system with the prefilter $C_{pf}(s)$ is drawn in Figure 5.19. We have also drawn on this figure the step response of $\frac{\omega_0^2}{s^2 + 2m\omega_0 s + \omega_0^2}$, which is the targeted closed-loop transfer function from which the specifications have been derived. We notice that now the unit step response fully comply with the specifications. ■

Example 5.9. Consider a plant with the following transfer function:

$$\begin{aligned}
 F(s) = \frac{N_F(s)}{D_F(s)} &= \frac{1}{(2s+1)(3s+1)(4s+1)} \\
 &= \frac{1/24}{s^3 + 26/24 s^2 + 9/24 s + 1/24}
 \end{aligned} \tag{5.233}$$

We wish to design a controller with relative degree $r = 0$ which minimize the ITAE performance index and also which leads to a settling time t_s for a step input less than 0.5 sec. We decide to put controller $C(s)$ in the direct path as shown in Figure 5.14.

The degree of the denominator of the plant transfer function is $n = 3$. Consequently we shall choose a predefined polynomial $q(s)$ whose degree is $2n + r - 1 = 5$. The coefficients of the closed-loop denominator are chosen in

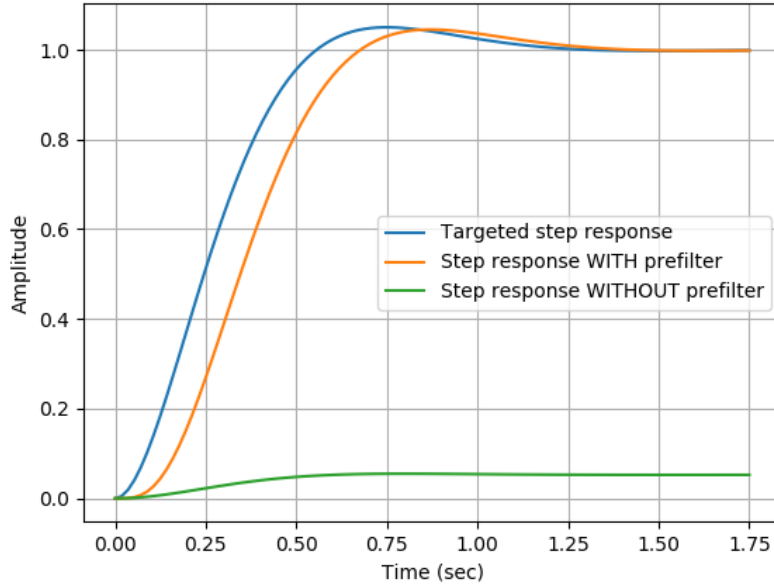


Figure 5.19: Unit step response without and with prefilter

order to minimize the ITAE performance index. From Table 5.2 we get the following polynomial of degree 5:

$$q(s) = s^5 + 2.068\omega_0 s^4 + 4.499\omega_0^2 s^3 + 4.675\omega_0^3 s^2 + 3.257\omega_0^4 s + \omega_0^5 \quad (5.234)$$

We need to select ω_0 in order to meet the settling time requirement. Since $t_s \approx \frac{4}{m\omega_0}$ where m is set in a first attempt to 0.8 we get:

$$\begin{cases} t_s = 0.5 \\ m = 0.8 \end{cases} \Rightarrow \omega_0 \approx \frac{4}{m t_s} = 10 \quad (5.235)$$

In addition the numerator and the denominator of the controller $C(s)$ are polynomial of degree $n + r - 1 = 2$:

$$\begin{cases} N_C(s) = y_2 s^2 + y_1 s + y_0 \\ D_C(s) = s^2 + x_1 s + x_0 \end{cases} \quad (5.236)$$

Consequently the closed-loop transfer function without prefiltering ($C_{pf}(s) = 1$) reads:

$$\begin{aligned} \frac{Y(s)}{R_{pf}(s)} &= \frac{N_C(s)N_F(s)}{N_C(s)N_F(s) + D_C(s)D_F(s)} \\ &= \frac{(y_2 s^2 + y_1 s + y_0)/24}{(y_2 s^2 + y_1 s + y_0)/24 + (s^2 + x_1 s + x_0)(s^3 + 26/24 s^2 + 9/24 s + 1/24)} \end{aligned} \quad (5.237)$$

That is:

$$\frac{Y(s)}{R_{pf}(s)} = \frac{(y_2 s^2 + y_1 s + y_0) / 24}{s^5 + q_4 s^4 + q_3 s^3 + q_2 s^2 + q_1 s + q_0} \quad (5.238)$$

Where:

$$\begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \\ q_4 \\ 1 \end{bmatrix} = \begin{bmatrix} 1/24 & 0 & 0 & | & 1/24 & 0 & 0 \\ 9/24 & 1/24 & 0 & | & 0 & 1/24 & 0 \\ 26/24 & 9/24 & 1/24 & | & 0 & 0 & 1/24 \\ 1 & 26/24 & 9/24 & | & 0 & 0 & 0 \\ 0 & 1 & 26/24 & | & 0 & 0 & 0 \\ 0 & 0 & 1 & | & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ \frac{1}{y_0} \\ y_1 \\ y_2 \end{bmatrix} \quad (5.239)$$

Equating those coefficients with the required coefficients of polynomial $q(s)$ given in (5.234) result in the following matrix equation:

$$\begin{bmatrix} \omega_0^5 \\ 3.257\omega_0^4 \\ 4.675\omega_0^3 \\ 4.499\omega_0^2 \\ 2.068\omega_0 \\ 1 \end{bmatrix} = \begin{bmatrix} 1/24 & 0 & 0 & | & 1/24 & 0 & 0 \\ 9/24 & 1/24 & 0 & | & 0 & 1/24 & 0 \\ 26/24 & 9/24 & 1/24 & | & 0 & 0 & 1/24 \\ 1 & 26/24 & 9/24 & | & 0 & 0 & 0 \\ 0 & 1 & 26/24 & | & 0 & 0 & 0 \\ 0 & 0 & 1 & | & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ \frac{1}{y_0} \\ y_1 \\ y_2 \end{bmatrix} \quad (5.240)$$

Solving this matrix equation for $\omega_0 = 10$ leads to the following values:

$$\begin{bmatrix} x_0 \\ x_1 \\ \frac{1}{y_0} \\ y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} 428.3 \\ 19.597 \\ \frac{1}{2.3996 \cdot 10^6} \\ 7.7781 \cdot 10^5 \\ 1.0089 \cdot 10^5 \end{bmatrix} \quad (5.241)$$

Thus the controller $C(s)$ has the following transfer function:

$$C(s) = \frac{2.3996 \cdot 10^6 + 7.7781 \cdot 10^5 s + 1.0089 \cdot 10^5 s^2}{428.3 + 19.597 s + 1 \times s^2} \quad (5.242)$$

This is, dividing the numerator and the denominator by 24:

$$C(s) = \frac{99982.154 + 32408.573s + 4203.623s^2}{17.845637 + 0.8165278s + 0.0416667s^2} \quad (5.243)$$

And the closed-loop transfer function without prefiltering (5.238) becomes:

$$\frac{Y(s)}{R_{pf}(s)} = \frac{99982.154 + 32408.573s + 4203.623s^2}{100000 + 32570s + 4675s^2 + 449.9s^3 + 20.68s^4 + s^5} \quad (5.244)$$

Then we select the prefilter $C_{pf}(s)$ in order to achieve the required ITAE transfer function. We check that all the roots of polynomial $99982.154 + 32408.573s + 4203.623s^2$ have negative real parts. Therefore we obtain for the transfer function of the prefilter:

$$C_{pf}(s) = \frac{100000}{99982.154 + 32408.573s + 4203.623s^2} \quad (5.245)$$

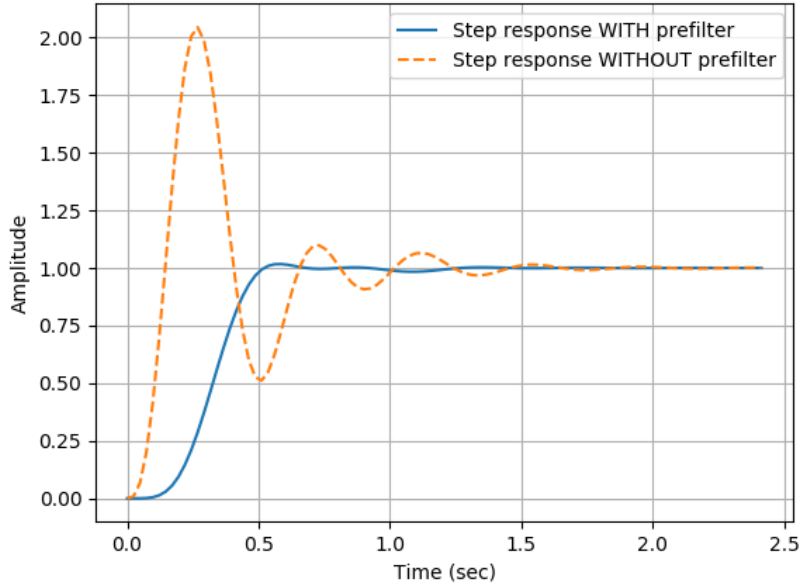


Figure 5.20: Unit step response of the whole system

We finally get the transfer function of the whole system:

$$\begin{aligned}
 \frac{Y(s)}{R(s)} &= C_{pf}(s) \frac{Y(s)}{R_{pf}(s)} \\
 &= C_{pf}(s) \frac{99982.154 + 32408.573s + 4203.623s^2}{100000 + 32570s + 4675s^2 + 449.9s^3 + 20.68s^4 + s^5} \\
 &= \frac{100000}{100000 + 32570s + 4675s^2 + 449.9s^3 + 20.68s^4 + s^5}
 \end{aligned} \tag{5.246}$$

Figure 5.20 shows the response of the whole system to a unit step input.

As expected the settling time lower than 0.5 sec. Furthermore the whole system has a zero steady state error and an overshoot about 1.7%. We can observed that the prefilter $C_{pf}(s)$ greatly decreases the overshoot. ■

5.7.5 Youla parameterization of all stabilizing controllers

We consider the feedback control loop with unity feedback shown in Figure 5.21 where $F(s)$ and $C(s)$ are the plant model and the controller transfer function, respectively.

We have seen that in a standard negative feedback configuration, the denominator of the closed-loop transfer function satisfies the following pole placement Diophantine equation where $q(s)$ is *Hurwitz*:

$$N_C(s)N_F(s) + D_C(s)D_F(s) = q(s) \tag{5.247}$$

Let split the stable right hand-side polynomial $q(s)$ as a factor of two *Hurwitz* polynomials $q_C(s)$ and $q_F(s)$:

$$q(s) = q_C(s)q_F(s) \tag{5.248}$$

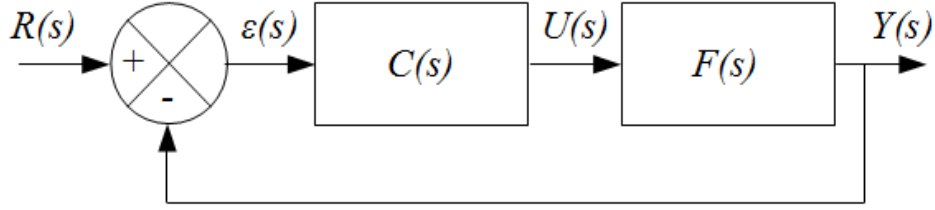


Figure 5.21: Feedback control loop with unity feedback

Then the Diophantine equation becomes the following Bézout equation:

$$\frac{N_C(s)}{q_C(s)} \frac{N_F(s)}{q_F(s)} + \frac{D_C(s)}{q_C(s)} \frac{D_F(s)}{q_F(s)} = 1 \quad (5.249)$$

Thus transfer function $F(s)$ can be divided into two coprime transfer functions $N(s)$ and $D(s)$ as follows:

$$F(s) = \frac{N_F(s)}{q_F(s)} \frac{q_F(s)}{D_F(s)} := \frac{N(s)}{D(s)} \quad \text{where} \quad \begin{cases} N(s) = \frac{N_F(s)}{q_F(s)} \\ D(s) = \frac{D_F(s)}{q_F(s)} \end{cases} \quad (5.250)$$

Because they are coprime, Bezout theorem indicates that *stable* transfer functions $N(s)$ and $D(s)$ are such that there exists two *stable* transfer functions $X(s)$ and $Y(s)$ such that the following relation holds:

$$Y(s)D(s) + X(s)N(s) = 1 \quad \forall s \quad (5.251)$$

Then Youla & al.⁹ have shown in 1976 that the class of controllers that make the closed-loop system internally stable is given as follows where $Q(s)$ is an arbitrary real *stable* rational transfer function:

$$C(s) = \frac{X(s) + Q(s)D(s)}{Y(s) - Q(s)N(s)} \quad (5.252)$$

It is worth noticing that if $F(s)$ is a *stable* transfer function, then we can make the following choice in the Bezout identity (5.251):

$$\begin{cases} Y(s) := 1 \\ D(s) := 1 \\ X(s) := 0 \\ N(s) := F(s) \end{cases} \quad (5.253)$$

With this choice, (5.252) becomes as follows where $Q(s)$ is still an arbitrary real *stable* rational transfer function:

$$C(s) = \frac{X(s) + Q(s)D(s)}{Y(s) - Q(s)N(s)} := \frac{Q(s)}{1 - Q(s)F(s)} \quad (5.254)$$

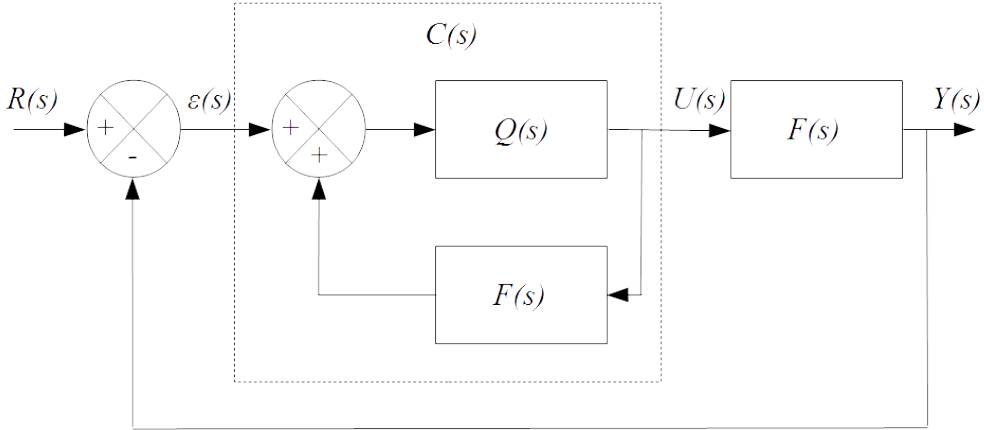


Figure 5.22: Feedback control loop with unity feedback and Youla parameterization

With this parameterization of controller $C(s)$, the feedback control loop with unity feedback shown in Figure 5.21 can be re-arranged as shown in Figure 5.22.

Furthermore, applying the parameterization of controller $C(s)$ expressed by (5.252) we obtain the following closed-loop transfer function for the feedback control loop with unity feedback shown in Figure 4.2:

$$G(s) = \frac{F(s)C(s)}{1 + F(s)C(s)} = \frac{F(s)\frac{Q(s)}{1-Q(s)F(s)}}{1 + F(s)\frac{Q(s)}{1-Q(s)F(s)}} = Q(s)F(s) \quad (5.255)$$

Because $G(s) := \frac{Y(s)}{R(s)}$ and $F(s) := \frac{Y(s)}{U(s)}$, we get that $Q(s)$ is the transfer function between $U(s)$ and $R(s)$

$$G(s) = \frac{Y(s)}{R(s)} = \frac{U(s)Y(s)}{R(s)U(s)} = \frac{U(s)}{R(s)}F(s) := Q(s)F(s) \Rightarrow Q(s) = \frac{U(s)}{R(s)} \quad (5.256)$$

More generally, for SISO plants, a necessary and sufficient condition for a factorization to be coprime is that $N(s)$ and $D(s)$ have no common zero neither finite nor infinite¹⁰.

Example 5.10. We consider the following transfer function:

$$F(s) = \frac{s+1}{(s-2)(s+3)} \quad (5.257)$$

Then, assuming any parameter $a > 0$ (indeed $N(s)$ and $D(s)$ shall be stable transfer functions), we can choose for example:

$$\begin{cases} N(s) = \frac{s+1}{(s+a)^2} \\ D(s) = \frac{(s-2)(s+3)}{(s+a)^2} \end{cases} \quad (5.258)$$

⁹Youla, D. C., Bongiorno, J. J., Jr., and Jabr, H. A., Modern Wiener-Hopf design of optimal controllers; Part 1: the single-input-output case, IEEE Transaction on Automatic Control, vol. AC-21, Feb. 1976, pp.3-13

¹⁰Fortuna L., Frasca M., Optimal and Robust Control, CRC Press, 2012

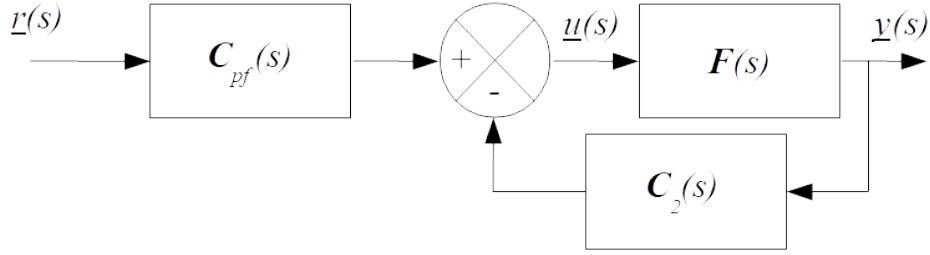


Figure 5.23: Two-degree-of-freedom control loop

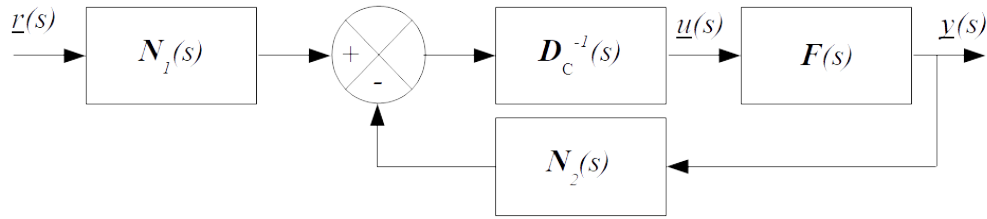


Figure 5.24: Polynomial realization of the two-degree-of-freedom control loop

The degree of polynomial $s+a$ is chosen to be 2 in order that $N(s)$ and $D(s)$ have no common zero when s is infinite. Furthermore, $a \notin \{-1, 2, -3\}$ in order that $N(s)$ and $D(s)$ are coprime.

Then stable transfer functions $X(s)$ and $Y(s)$ can be computed such that the following relation holds:

$$Y(s)D(s) + X(s)N(s) = 1 \quad \forall s \quad (5.259)$$

where, assuming any parameter $b > 0$:

$$\begin{cases} X(s) = \frac{s+c}{s+b} \\ Y(s) = \frac{s+d}{s+b} \end{cases} \quad (5.260)$$

■

5.7.6 Two degrees of freedom controller design

Consider the configuration shown in Figure 5.23 where $\mathbf{F}(s)$ is the plant transfer function, $\mathbf{C}_{pf}(s)$ a prefilter and $\mathbf{C}_2(s)$ a controller in the feedback loop. This feedback loop is called a two-degree-of-freedom control loop. From Figure 5.23 control $\underline{u}(s)$ reads:

$$\underline{u}(s) = \mathbf{C}_{pf}(s)\underline{r}(s) - \mathbf{C}_2(s)\underline{y}(s) \quad (5.261)$$

Now we select controllers $\mathbf{C}_{pf}(s)$ and $\mathbf{C}_2(s)$ with the same polynomial denominator $\mathbf{D}_C(s)$ and polynomial numerator $\mathbf{N}_1(s)$ and $\mathbf{N}_2(s)$, respectively¹¹:

$$\begin{cases} \mathbf{C}_{pf}(s) = \mathbf{D}_C^{-1}(s)\mathbf{N}_1(s) \\ \mathbf{C}_2(s) = \mathbf{D}_C^{-1}(s)\mathbf{N}_2(s) \end{cases} \quad (5.262)$$

¹¹Chi-Tsong Chen, Linear System Theory and Design, Oxford University Press, New York, 1999, 334pages, ISBN 0-19-511777-8.

Thus control $\underline{u}(s)$ can now be written as follows:

$$\begin{aligned}\underline{u}(s) &= \mathbf{D}_C^{-1}(s)\mathbf{N}_1(s)\underline{r}(s) - \mathbf{D}_C^{-1}(s)\mathbf{N}_2(s)\underline{y}(s) \\ &= \mathbf{D}_C^{-1}(s) (\mathbf{N}_1(s)\underline{r}(s) - \mathbf{N}_2(s)\underline{y}(s))\end{aligned}\quad (5.263)$$

This leads to the polynomial realization of the two-degree-of-freedom control loop shown in Figure 5.24.

Now, assume that the plant transfer function $\mathbf{F}(s)$ reads as follows:

$$\mathbf{F}(s) = \mathbf{N}_F(s)\mathbf{D}_F^{-1}(s) \quad (5.264)$$

Then from Figure 5.24 the relation between control $\underline{u}(s)$ and reference input $\underline{r}(s)$ reads as follows:

$$\begin{aligned}\underline{u}(s) &= \mathbf{D}_C^{-1}(s) (\mathbf{N}_1(s)\underline{r}(s) - \mathbf{N}_2(s)\mathbf{N}_F(s)\mathbf{D}_F^{-1}(s)\underline{u}(s)) \\ \Leftrightarrow (\mathbb{I} + \mathbf{D}_C^{-1}(s)\mathbf{N}_2(s)\mathbf{N}_F(s)\mathbf{D}_F^{-1}(s)) \underline{u}(s) &= \mathbf{D}_C^{-1}(s)\mathbf{N}_1(s)\underline{r}(s)\end{aligned}\quad (5.265)$$

We finally get:

$$\begin{aligned}\underline{u}(s) &= (\mathbb{I} + \mathbf{D}_C^{-1}(s)\mathbf{N}_2(s)\mathbf{N}_F(s)\mathbf{D}_F^{-1}(s))^{-1} \mathbf{D}_C^{-1}(s)\mathbf{N}_1(s)\underline{r}(s) \\ &= (\mathbf{D}_C(s) (\mathbb{I} + \mathbf{D}_C^{-1}(s)\mathbf{N}_2(s)\mathbf{N}_F(s)\mathbf{D}_F^{-1}(s)))^{-1} \mathbf{N}_1(s)\underline{r}(s) \\ &= (\mathbf{D}_C(s) + \mathbf{N}_2(s)\mathbf{N}_F(s)\mathbf{D}_F^{-1}(s))^{-1} \mathbf{N}_1(s)\underline{r}(s) \\ &= ((\mathbf{D}_C(s)\mathbf{D}_F(s) + \mathbf{N}_2(s)\mathbf{N}_F(s)) \mathbf{D}_F^{-1}(s))^{-1} \mathbf{N}_1(s)\underline{r}(s) \\ &= \mathbf{D}_F(s) (\mathbf{D}_C(s)\mathbf{D}_F(s) + \mathbf{N}_2(s)\mathbf{N}_F(s))^{-1} \mathbf{N}_1(s)\underline{r}(s)\end{aligned}\quad (5.266)$$

This leads to the following relation between output $\underline{y}(s)$ and reference $\underline{r}(s)$:

$$\begin{aligned}\underline{y}(s) &= \mathbf{F}(s)\underline{u}(s) \\ &= \mathbf{N}_F(s)\mathbf{D}_F^{-1}(s)\mathbf{D}_F(s) (\mathbf{D}_C(s)\mathbf{D}_F(s) + \mathbf{N}_2(s)\mathbf{N}_F(s))^{-1} \mathbf{N}_1(s)\underline{r}(s) \\ &= \mathbf{N}_F(s) (\mathbf{D}_C(s)\mathbf{D}_F(s) + \mathbf{N}_2(s)\mathbf{N}_F(s))^{-1} \mathbf{N}_1(s)\underline{r}(s)\end{aligned}\quad (5.267)$$

Thus the closed-loop transfer function reads:

$$\underline{y}(s) = \mathbf{G}(s)\underline{r}(s) \quad (5.268)$$

where:

$$\mathbf{G}(s) = \mathbf{N}_F(s) (\mathbf{D}_C(s)\mathbf{D}_F(s) + \mathbf{N}_2(s)\mathbf{N}_F(s))^{-1} \mathbf{N}_1(s) \quad (5.269)$$

In the SISO case, this simplifies as follows:

$$G(s) = \frac{N_1(s)N_F(s)}{D_C(s)D_F(s) + N_2(s)N_F(s)} \quad (5.270)$$

Then the model matching problem becomes solving $D_C(s)$, $N_1(s)$ and $N_2(s)$ from (5.270). More specifically, let $F(s) = \frac{N_F(s)}{D_F(s)}$ be a proper transfer function, where $N_F(s)$ and $D_F(s)$ are coprime. The problem statement for the two degrees of freedom controller design, or model matching, is the following: given any targeted closed-loop transfer function $T(s)$, find proper controllers $C_{pf}(s) = \frac{N_1(s)}{D_C(s)}$ and $C_2(s) = \frac{N_2(s)}{D_C(s)}$ in Figure 5.23 to implement $T(s)$. Following Chi-Tsong Chen¹², this problem is solved as follows:

¹²Shou-Yuan Zhang and Chi-Tsong Chen, Design of unity feedback systems to achieve arbitrary denominator matrix, in IEEE Transactions on Automatic Control, vol. 28, no. 4, pp. 518-521, April 1983, doi: 10.1109/TAC.1983.1103255

1. First check that the targeted closed-loop transfer function $T(s)$ is *implementable*, that is that $\frac{T(s)}{F(s)}$ is proper and BIBO stable;
2. Compute polynomials $E(s)$ and $H(s)$, where $E(s)$ and $H(s)$ are coprime, such that:

$$\frac{T(s)}{N_F(s)} = \frac{E(s)}{H(s)} \quad (5.271)$$

3. Let n be the degree of $D_F(s)$ and $r \geq 0$ be the *chosen* relative degree of $C_2(s)$. Introduce a *Hurwitz* polynomial $P(s)$ (polynomial whose roots have negative real part) such that $P(s)H(s)$ is a polynomial of degree $2n + r - 1$, :

$$T(s) = \frac{P(s)E(s)N_F(s)}{P(s)H(s)} \quad (5.272)$$

4. Make the identification:

$$T(s) = G(s) \Leftrightarrow \frac{P(s)E(s)N_F(s)}{P(s)H(s)} = \frac{N_1(s)N_F(s)}{D_C(s)D_F(s) + N_2(s)N_F(s)} \quad (5.273)$$

It is clear the achievement of the targeted transfer function $T(s)$ is based on the pole / zero cancellation of the roots of $P(s)$. This justifies why polynomial $P(s)$ shall be *Hurwitz*.

5. From the numerator of (5.273), we get the expression of $N_1(s)$:

$$N_1(s) = P(s)E(s) \quad (5.274)$$

6. Let m be the degree of *monic polynomial* $D_C(s)$ and $m - r$ be the degree of polynomial $N_2(s)$, such that $C_2(s) = \frac{N_2(s)}{D_C(s)}$ has a relative degree r . Using the fact that the degree of $N_2(s)N_F(s)$ is lower or equal than the degree of $D_C(s)D_F(s)$ (because $F(s)$ and $C_2(s)$ are both proper transfer functions), we conclude that polynomial $q(s) := N_2(s)N_F(s) + D_C(s)D_F(s)$ is of degree $n + m$. Because $D_C(s)$ and $D_F(s)$ are *monic polynomials*, we conclude that polynomial $q(s)$ is also a *monic polynomial* with $n + m$ coefficients.

On the other hand controller $C_2(s)$ is formed with $m - r + 1$ unknown coefficients in its numerator (y_0, y_1, \dots, y_{m-r}) and m unknown coefficients in its denominator (x_0, x_1, \dots, x_{m-1}), which forms a set of $2m - r + 1$ unknown coefficients.

In order to be able to identify all the coefficients of s^i in the polynomial which forms the denominator of (5.273) the following equation shall be satisfied:

$$n + m = 2m - r + 1 \Rightarrow m = n + r - 1 \quad (5.275)$$

Thus we get $C_2(s) = \frac{N_2(s)}{D_C(s)}$ where:

$$\begin{cases} D_C(s) = 1 \times s^{n+r-1} + x_{n+r-2}s^{n+r-2} + \dots + x_1s + x_0 \\ N_2(s) = y_{n-1}s^{n-1} + \dots + y_1s + y_0 \end{cases} \quad (5.276)$$

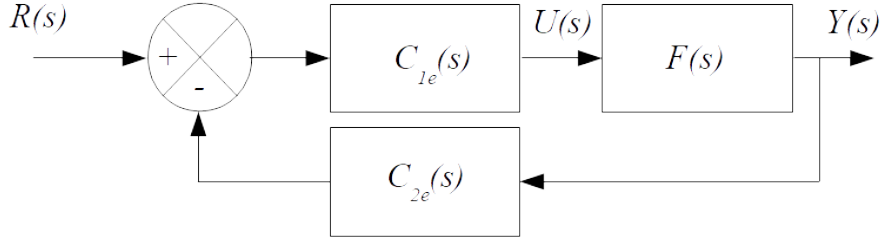


Figure 5.25: Equivalent two-degree-of-freedom control loop

As seen in section 5.7.3, coefficients x_i of $D_C(s)$ and y_j of $N_2(s)$ can be obtained by:

$$\mathbf{S}_{AB} \begin{bmatrix} x_0 \\ \vdots \\ x_{n+r-2} \\ 1 \\ \hline y_0 \\ \vdots \\ y_{n-1} \end{bmatrix} = \begin{bmatrix} q_0 \\ q_1 \\ \vdots \\ q_{2n+r-2} \\ 1 \end{bmatrix} \quad (5.277)$$

Let:

$$\begin{cases} N_F(s) = b_0 + b_1s + \dots + b_ms^m \\ D_F(s) = a_0 + a_1s + \dots + a_{n-1}s^{n-1} + 1 \times s^n \end{cases} \quad (5.278)$$

Then \mathbf{S}_{AB} is the following Sylvester resultant (square) matrix:

$$\mathbf{S}_{AB} = \left[\begin{array}{ccc|ccc} a_0 & 0 & \dots & b_0 & 0 & \dots \\ a_1 & a_0 & \ddots & b_1 & b_0 & \ddots \\ \vdots & a_1 & \ddots & \vdots & b_1 & \ddots \\ a_{n-1} & \vdots & & b_{m-1} & \vdots & \\ 1 & a_{n-1} & & b_m & b_{m-1} & \\ 0 & 1 & \ddots & 0 & b_m & \ddots \\ \vdots & \ddots & \ddots & \vdots & \ddots & \ddots \end{array} \right] \left. \vphantom{\begin{array}{ccc|ccc} \right\} 2n+r \text{ rows} \quad (5.279)$$

$\underbrace{\hspace{10em}}_{n+r \text{ columns}} \quad \underbrace{\hspace{10em}}_{n \text{ columns}}$

Then prefilter controller $C_{pf}(s) = \frac{N_1(s)}{D_C(s)}$ and feedback loop controller $C_2(s) = \frac{N_2(s)}{D_C(s)}$ in Figure 5.23 are obtained to achieve model matching $G(s) = T(s)$.

On the basis of the polynomial realization of the two-degree-of-freedom control loop in Figure 5.24, an equivalent two-degree-of-freedom control loop is

shown in Figure 5.25 where¹³:

$$\begin{cases} C_{1e}(s) = \frac{N_1(s)}{D_C(s)} \\ C_{2e}(s) = \frac{N_2(s)}{N_1(s)} \end{cases} \quad (5.280)$$

Then we get:

$$\begin{aligned} G(s) &= \frac{N_1(s)N_F(s)}{D_C(s)D_F(s)+N_2(s)N_F(s)} \\ &= \frac{\frac{N_1(s)N_F(s)}{D_C(s)D_F(s)}}{1+\frac{N_2(s)N_F(s)}{D_C(s)D_F(s)}} \\ &= \frac{C_{1e}(s)F(s)}{1+C_{1e}(s)C_{2e}(s)F(s)} \end{aligned} \quad (5.281)$$

Other design methods are presented by Wolovich¹⁴ and Febina Beevi & al.¹⁵.

Example 5.11. Consider a plant with the following transfer function:

$$\begin{aligned} F(s) = \frac{N_F(s)}{D_F(s)} &= \frac{1}{(2s+1)(3s+1)(4s+1)} \\ &= \frac{1}{1+9s+26s^2+24s^3} \\ &= \frac{1/24}{s^3+26/24s^2+9/24s+1/24} \end{aligned} \quad (5.282)$$

We wish to design a two degrees of freedom controller design which minimizes an ITAE performance index with a settling time t_s for a step input less than 0.5 sec.

1. Let $T(s)$ be the targeted closed-loop transfer function. The degree of the denominator of the plant transfer function is $n = 3$. The ratio $\frac{T(s)}{F(s)}$ will be proper and BIBO stable if $T(s)$ is chosen for example as follows, see Table 5.2:

$$T(s) = \frac{\omega_0^3}{s^3 + 1.783\omega_0 s^2 + 2.172\omega_0^2 s + \omega_0^3} \quad (5.283)$$

We need to select ω_0 in order to meet the settling time requirement. Since $t_s \approx \frac{4}{m\omega_0}$ where m is set in a first attempt to 0.8 we get:

$$\begin{cases} t_s = 0.5 \\ m = 0.8 \end{cases} \Rightarrow \omega_0 \approx \frac{4}{m t_s} = 10 \quad (5.284)$$

2. Polynomials $E(s)$ and $H(s)$, where $E(s)$ and $H(s)$ are coprime, are such that:

$$\frac{T(s)}{N_F(s)} = \frac{24\omega_0^3}{s^3 + 1.783\omega_0 s^2 + 2.172\omega_0^2 s + \omega_0^3} := \frac{E(s)}{H(s)} \quad (5.285)$$

¹³Pradin B., Garcia G., Modélisation, analyse et commande des systèmes linéaires, Presses Universitaires du Mirail, 2010

¹⁴William A. Wolovich, Automatic Control Systems: Basic Analysis and Design, Saunders College Publishing, 1995, ISBN-10 : 0030237734

¹⁵P. Febina Beevi, T. K. Sunil Kumar, Jeevamma Jacob, Novel Design Method for Two Degree of Freedom Controllers, International Review of Automatic Control (I.R.E.A.CO.), Vol. 8, N. 4, July 2015, ISSN 1974-6059

3. The degree of $H(s)$ is $n = 3$. We choose the relative degree of $C_2(s)$ to be $r = 0$. Then Hurwitz polynomial $P(s)$ (polynomial whose roots have negative real part) shall be chosen such that $P(s)H(s)$ is a polynomial of degree $2n + r - 1 = 5$. Consequently $P(s)$ is a polynomial of degree $5 - 3 = 2$ and we choose the 2 roots of $P(s)$ faster than the roots of the denominator $T(s)$ when $\omega_0 = 10$, for example -15 twice:

$$P(s) = (s + 15)^2 \quad (5.286)$$

4. Then we make the identification:

$$\begin{aligned} T(s) &= G(s) \\ \Leftrightarrow \frac{P(s)E(s)N_F(s)}{P(s)H(s)} &= \frac{N_1(s)N_F(s)}{D_C(s)D_F(s)+N_2(s)N_F(s)} \\ \frac{(s+15)^2 24\omega_0^3|_{\omega_0=10} (1/24)}{(s^3+1.783\omega_0 s^2+2.172\omega_0^2 s+\omega_0^3)|_{\omega_0=10}} &= \frac{N_1(s)N_F(s)}{D_C(s)D_F(s)+N_2(s)N_F(s)} \quad (5.287) \\ \Leftrightarrow \frac{(s+15)^2 \omega_0^3|_{\omega_0=10}}{q(s)} &= \frac{N_1(s)N_F(s)}{D_C(s)D_F(s)+N_2(s)N_F(s)} \end{aligned}$$

5. From the numerator of $T(s)$, we get the expression of $N_1(s)$:

$$N_1(s) = P(s)E(s) = (s + 15)^2 24\omega_0^3|_{\omega_0=10} = 24 \cdot 10^3 (s + 15)^2 \quad (5.288)$$

6. Let $q(s)$ be the denominator of $T(s)$. We have seen that:

$$\begin{aligned} q(s) &= (s^3 + 1.783\omega_0 s^2 + 2.172\omega_0^2 s + \omega_0^3)|_{\omega_0=10} (s + 15)^2 \\ &= (s^3 + 17.83 s^2 + 217.2 s + 10^3) (s + 15)^2 \\ &= s^5 + 47.83 s^4 + 977.1 s^3 + 11.5277 \cdot 10^3 s^2 + 78.87 \cdot 10^3 s + 225 \cdot 10^3 \\ &:= s^5 + q_4 s^4 + q_3 s^3 + q_2 s^2 + q_1 s + q_0 \end{aligned} \quad (5.289)$$

Diophantine equation $D_C(s)D_F(s) + N_2(s)N_F(s) = q(s)$ is then solved where monic polynomial $D_C(s)$ is of degree $n + r - 1 = 2$ and polynomial $N_2(s)$ is of degree $n - 1 = 2$:

$$\begin{cases} D_C(s) = s^2 + x_1 s + x_0 \\ N_2(s) = y_2 s^2 + y_1 s + y_0 \end{cases} \quad (5.290)$$

As seen in section 5.7.3, coefficients x_i of $D_C(s)$ and y_j of $N_2(s)$ can be obtained by:

$$\mathbf{S}_{AB} \begin{bmatrix} x_0 \\ x_1 \\ -\frac{1}{y_0} \\ y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \\ q_4 \\ 1 \end{bmatrix} \quad (5.291)$$

where \mathbf{S}_{AB} is the following Sylvester resultant (square) matrix:

$$\mathbf{S}_{AB} = \begin{bmatrix} 1/24 & 0 & 0 & | & 1/24 & 0 & 0 \\ 9/24 & 1/24 & 0 & | & 0 & 1/24 & 0 \\ 26/24 & 9/24 & 1/24 & | & 0 & 0 & 1/24 \\ 1 & 26/24 & 9/24 & | & 0 & 0 & 0 \\ 0 & 1 & 26/24 & | & 0 & 0 & 0 \\ 0 & 0 & 1 & | & 0 & 0 & 0 \end{bmatrix} \quad (5.292)$$

Solving this matrix equation leads to the following values:

$$\begin{bmatrix} x_0 \\ x_1 \\ \frac{1}{y_0} \\ y_1 \\ y_2 \end{bmatrix} = \mathbf{S}_{AB}^{-1} \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \\ q_4 \\ 1 \end{bmatrix} = \begin{bmatrix} 926.08 \\ 46.747 \\ \frac{1}{5.3991 \cdot 10^6} \\ 1.8845 \cdot 10^6 \\ 0.2521 \cdot 10^6 \end{bmatrix} \quad (5.293)$$

The knowledge of polynomials $D_C(s)$, $N_2(s)$ and $N_1(s)$ enable to form the controllers $C_{1e}(s)$ and $C_{2e}(s)$:

$$\begin{cases} C_{1e}(s) = \frac{N_1(s)}{D_C(s)} = \frac{24 \cdot 10^3 (s+15)^2}{s^2 + x_1 s + x_0} = \frac{24 \cdot 10^3 (s+15)^2}{s^2 + 46.747s + 926.08} \\ C_{2e}(s) = \frac{N_2(s)}{N_1(s)} = \frac{y_2 s^2 + y_1 s + y_0}{10^3 (s+15)^2} = \frac{0.2521 \cdot 10^6 s^2 + 1.8845 \cdot 10^6 s + 5.3991 \cdot 10^6}{10^3 (s+15)^2} \end{cases} \quad (5.294)$$

$$\begin{cases} C_1(s) = \frac{N_1(s)}{D_1(s)} = \frac{10^3 (s+15)^2}{y_2 s^2 + y_1 s + y_0} = \frac{10^3 (s+15)^2}{41.6667 \times 10^{-3} s^2 + 1.9478s + 38.5868} \\ C_2(s) = \frac{N_2(s)}{N_1(s)} = \frac{a_2 s^2 + a_1 s + a_0}{10^3 (s+15)^2} = \frac{10.5069 \times 10^3 s^2 + 78.5208 \times 10^3 s + 224.9614 \times 10^3}{10^3 (s+15)^2} \end{cases} \quad (5.295)$$

Figure 5.26 shows the response of the closed-loop with controllers $C_1(s)$ and $C_2(s)$ to a unit step input. The targeted step response is the response of the system with ITAE transfer function $T(s) = \frac{\omega_0^3}{s^3 + 1.783\omega_0 s^2 + 2.172\omega_0^2 s + \omega_0^3}$ where $\omega_0 = 10$. As expected the settling time lower than 0.5 sec. Furthermore the whole system has a zero steady state error and an overshoot about 1.6%. ■

5.7.7 Equivalent two-degree of freedom feedback loops

Following Taguchi & Araki¹⁶, the unity feedback loop with prefilter shown in Figure 5.27 is equivalent to either the *feedforward* two-degree of freedom control loop shown in Figure 5.28 or to the *feedback* two-degree of freedom control loop shown in Figure 5.29. Indeed:

¹⁶Hidefumi Taguchi, Mituhiko Araki, Two-Degree-of-Freedom PID Controllers - Their Functions and Optimal Tuning, IFAC Proceedings Volumes, Volume 33, Issue 4, 2000, Pages 91-96, ISSN 1474-6670, [https://doi.org/10.1016/S1474-6670\(17\)38226-5](https://doi.org/10.1016/S1474-6670(17)38226-5).

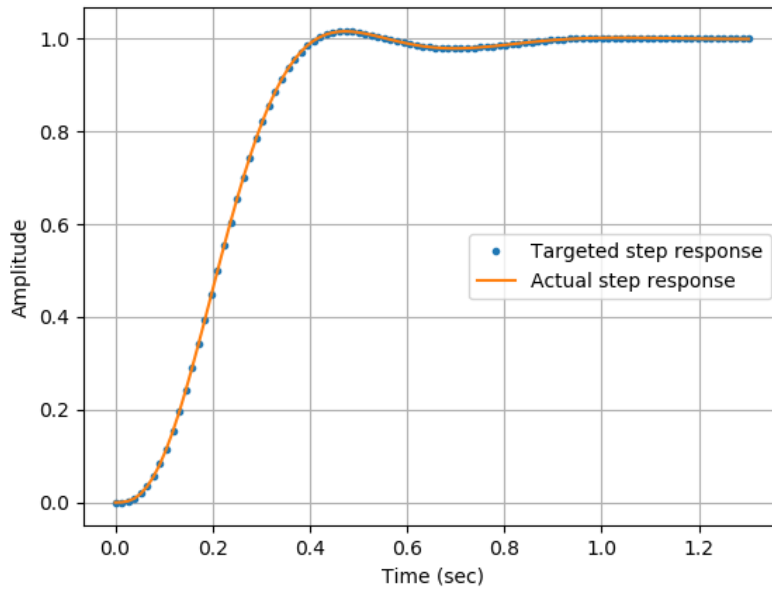


Figure 5.26: Unit step response of the closed-loop

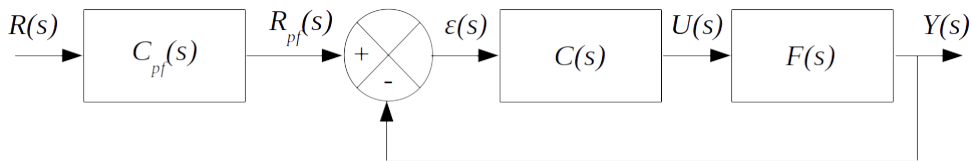


Figure 5.27: Unity feedback loop with prefilter

- The transfer function of the unity feedback loop with prefilter in Figure 5.27 reads:

$$\frac{Y(s)}{R(s)} = \frac{C_{pf}(s)F(s)C(s)}{1 + F(s)C(s)} \quad (5.296)$$

- The transfer function of the *feedforward* two-degree of freedom control loop shown in Figure 5.28 reads:

$$\frac{Y(s)}{R(s)} = \frac{F(s)(C_1(s) + C_{ff}(s))}{1 + F(s)C_1(s)} \quad (5.297)$$

- Finally, the transfer function of the *feedback* two-degree of freedom control loop shown in Figure 5.29 reads:

$$\frac{Y(s)}{R(s)} = \frac{F(s)C_2(s)}{1 + F(s)(C_2(s) + C_{fb}(s))} \quad (5.298)$$

By identifying the numerator and the denominator of each transfer function,

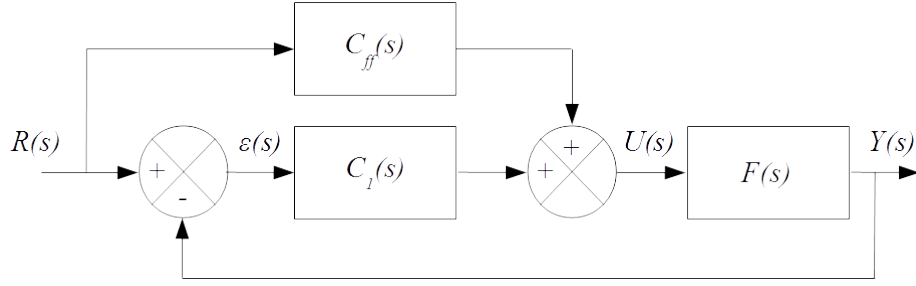


Figure 5.28: Feedforward two-degree of freedom control loop

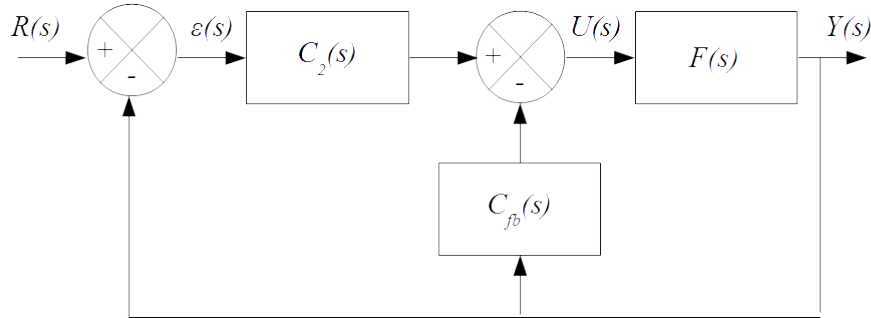


Figure 5.29: Feedback two-degree of freedom control loop

we get the following relations:

$$\begin{cases} C_{pf}(s)C(s) = C_1(s) + C_{ff}(s) = C_2(s) \\ C(s) = C_1(s) = C_2(s) + C_{fb}(s) \end{cases} \quad (5.299)$$

5.8 Usefulness of integral action

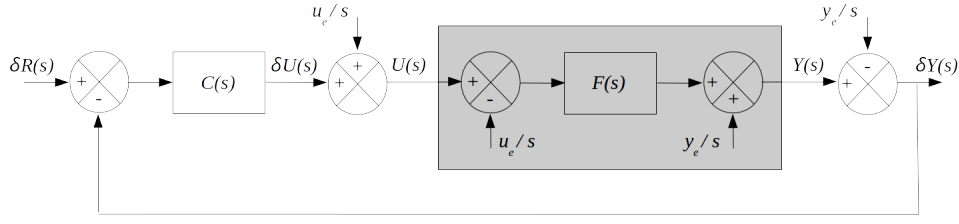
Usually controllers are designed on the basis on a linearized model of the nonlinear plant to be controlled around an equilibrium point whose value is not the origin: $(u_e, y_e) \neq (0, 0)$. More precisely we have seen that the linearization of a nonlinear plant around the equilibrium point (u_e, y_e) leads to the following input-output relation where $N(s)$ and $D(s)$ are polynomials in s and s is the variable corresponding to the derivation in the time domain ($s := \frac{d}{dt}$):

$$D(s) (y(t) - y_e) = N(s) (u(t) - u_e) \quad (5.300)$$

By taking the Laplace transform of the preceding equation (with initial conditions set to zero), and taking into account that because u_e and y_e are constants their Laplace transform is u_e/s and y_e/s respectively, we get:

$$\begin{aligned} D(s) (Y(s) - \frac{y_e}{s}) &= N(s) (U(s) - \frac{u_e}{s}) \\ \Leftrightarrow Y(s) &= \frac{N(s)}{D(s)} (U(s) - \frac{u_e}{s}) + \frac{y_e}{s} \end{aligned} \quad (5.301)$$

Fraction $\frac{N(s)}{D(s)} = F(s)$ represents the transfer function of the linearized plant around the equilibrium point (u_e, y_e) . In order to correctly control $F(s)$ we

Figure 5.30: Boucle de contrôle autour d'un point d'équilibre $(u_e, y_e) \neq (0, 0)$

shall add to the control δu provided by the controller $C(s)$ the value u_e of the control at equilibrium. Similarly we shall subtract to the output y of the actual plant the value y_e at equilibrium. This is shown in Figure 5.30 where the grey box represents the plant to be controlled. Notice that the commanded value δr which feeds the comparator is given with respect to the output at equilibrium: $\delta r(t) = r(t) - y_e$.

In order to avoid to add in the loop the values (u_e, y_e) of the equilibrium point, we will see that it is often interesting to put within the controller $C(s)$ an integral effect. To see that we first compute the expression of output $Y(s)$ as a function of inputs u_e and $\delta R(s)$ when the plant is controlled by the loop shown in Figure 5.30:

$$\begin{aligned} Y(s) &= F(s) \left(\frac{u_e}{s} + C(s) (\delta R(s) - (Y(s) - \frac{y_e}{s})) \right) \\ \Rightarrow Y(s) &= \frac{F(s)}{1+F(s)C(s)} \frac{u_e}{s} + \frac{F(s)C(s)}{1+F(s)C(s)} (\delta R(s) - \frac{y_e}{s}) \end{aligned} \quad (5.302)$$

Using the fact that $\delta r(t) = r(t) - y_e$, and taking the Laplace transform of this expression, we get $\delta R(s) = R(s) - \frac{y_e}{s}$. Thus we can substitute $\delta R(s) - \frac{y_e}{s}$ by $R(s)$. We get:

$$Y(s) = \frac{F(s)}{1+F(s)C(s)} \frac{u_e}{s} + \frac{F(s)C(s)}{1+F(s)C(s)} R(s) \quad (5.303)$$

For *high* values of time, the value of output $y(t)$ is obtained thanks to the final value theorem, assuming that the loop is stable:

$$\begin{aligned} \lim_{t \rightarrow \infty} y(t) &= \lim_{s \rightarrow 0} sY(s) \\ &= \lim_{s \rightarrow 0} \left(\frac{F(s)}{1+F(s)C(s)} u_e + \frac{F(s)C(s)}{1+F(s)C(s)} s R(s) \right) \end{aligned} \quad (5.304)$$

Assuming that $F(0) < \infty$ and that controller $C(s)$ exhibits an integral term (i.e. $C(s) = \frac{N_C(s)}{s D_{C1}(s)}$ where $\frac{N_C(0)}{D_{C1}(0)} < \infty$), we realize that the term which depends on u_e in $y(t)$ cancels for *high* values of time:

$$\left\{ \begin{array}{l} F(0) < \infty \\ C(s) = \frac{N_C(s)}{s D_{C1}(s)} \\ \frac{N_C(0)}{D_{C1}(0)} < \infty \end{array} \right. \Rightarrow \lim_{s \rightarrow 0} \frac{F(0)}{1 + F(0) \frac{N_C(0)}{s D_{C1}(0)}} u_e = 0 \quad (5.305)$$

Consequently, the integral term within the controller $C(s)$ enables to *forget* the value u_e of the control at equilibrium. Notice that a similar reasoning can

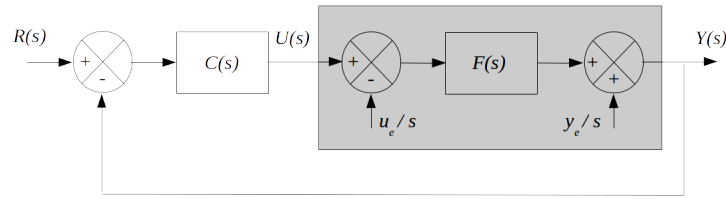


Figure 5.31: Boucle de contrôle autour d'un point d'équilibre $(u_e, y_e) \neq (0, 0)$ avec un correcteur $C(s)$ à effet intégral

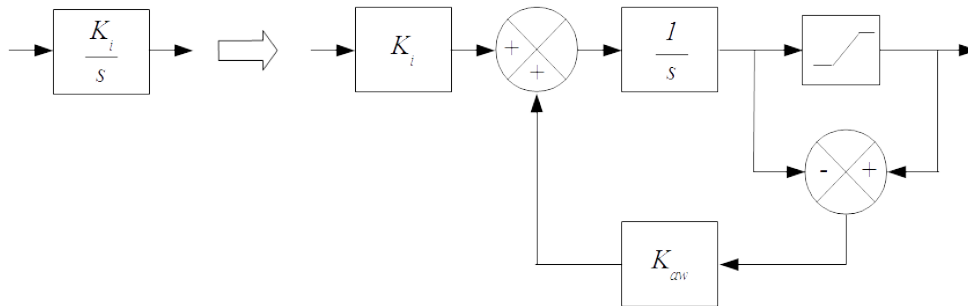


Figure 5.32: Current anti-windup structure dealing with integral term

be applied to check that the same effect is obtained when u_e represents an exogenous disturbance. Thus the control loop simplifies as shown in Figure 5.31 where the integrator within $C(s)$ shall be initialized with the value u_e of the control at equilibrium.

5.9 Anti-windup

Usually amplitude of input of actual plants is limited. When the controller exhibits an integral effect, saturation on the input signal of the plant may generate high overshoot or render the closed-loop unstable. This phenomenon is known as *windup* effect. The anti-windup controller design for saturated control consists in taking into account the saturated control in order to minimize *windup* effect and to enhance the domain where stability and performance of the closed-loop system are retained comparatively to the case where a saturation is simply placed after the controller to limit the input signal of the plant.

A usual anti-windup structure is to replace the integral term $\frac{K_i}{s}$ of a PID (or a PI-D) controller by the structure shown in Figure 5.32 where the gain K_{aw} is usually chosen between $\frac{K_i}{K_p}$ and $10\frac{K_i}{K_p}$.

Alternatively we present in Figure 5.33 an anti-windup structure proposed by R.H. Middleton¹⁷. The stability analysis of such a structure needs mathematical

¹⁷Dealing With Actuator Saturation, R.H. Middleton, The Control Handbook, A CRC Handbook Published in Cooperation with IEEE Press, 1996

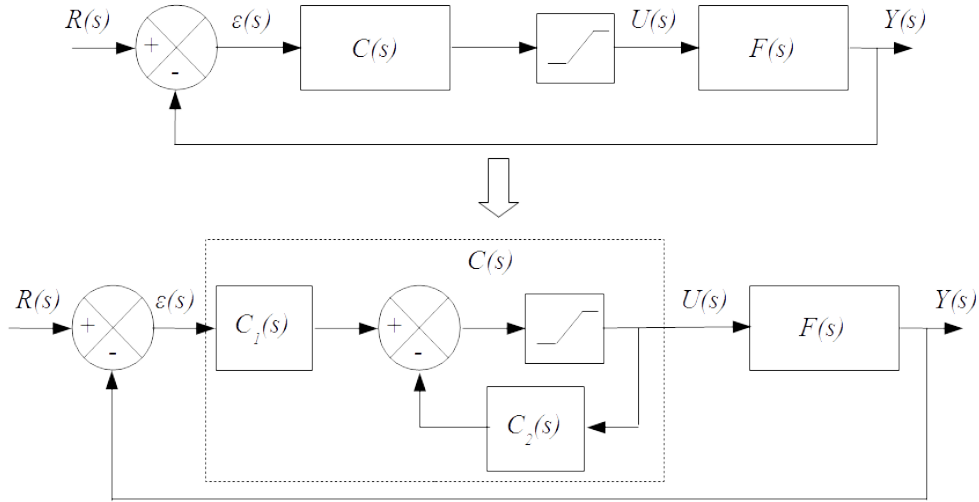


Figure 5.33: Anti-windup structure

tool such as LMI (Linear Matrix Inequalities) which are out of the scope of this lecture.

Controller $C(s)$ is written as $C(s) = N_C(s)/D_C(s)$, where polynomials $N_C(s)$ and $D_C(s)$ are respectively the numerator and the denominator of $C(s)$. In the following, we will denote $D_{aw}(s)$ the polynomial with the same degree than $D_C(s)$. Effectiveness of the anti-windup structure depends on the choice of polynomial $D_{aw}(s)$. Usually acceptable results are obtained by building $D_{aw}(s)$ from the n_f fastest poles of the closed-loop system obtained through the use of controller $C(s)$ and assuming no saturation:

$$\left\{ \begin{array}{l} C(s) = \frac{N_C(s)}{D_C(s)} \\ \deg(D_C(s)) = n_f \\ \frac{F(s)C(s)}{1+F(s)C(s)} = \frac{N_{cl}(s)}{(s-\lambda_1)\cdots(s-\lambda_{n_f})\cdots(s-\lambda_n)} \\ Re(\lambda_1) \leq \cdots \leq Re(\lambda_{n_f}) \leq \cdots \leq Re(\lambda_n) \end{array} \right. \Rightarrow D_{aw}(s) = (s - \lambda_1) \cdots (s - \lambda_{n_f}) \quad (5.306)$$

Then the anti-windup structure is composed of three parts as shown in Figure 5.33:

- A controller $C_1(s)$ whose transfer function reads:

$$C_1(s) = \frac{N_C(s)}{D_{aw}(s)} \quad (5.307)$$

- A controller $C_2(s)$ whose transfer function reads:

$$C_2(s) = \frac{D_C(s) - D_{aw}(s)}{D_{aw}(s)} \quad (5.308)$$

- A nonlinear bloc which reproduces the saturation on the input signal $u(t)$ of the plant.

It is worth noticing that assuming a non-saturated input signal, the antiwindup structure produces the same effect controller $C(s)$. Indeed:

$$\begin{cases} U(s) = C_1(s)\epsilon(s) - C_2(s)U(s) \\ C_1(s) = \frac{N_C(s)}{D_{aw}(s)} \\ C_2(s) = \frac{D_C(s) - D_{aw}(s)}{D_{aw}(s)} \end{cases} \Rightarrow \frac{U(s)}{\epsilon(s)} = \frac{C_1(s)}{1 + C_2(s)} = \frac{N_C(s)}{D_C(s)} \quad (5.309)$$

Example 5.12. Consider a plant with the following transfer function:

$$F(s) = \frac{1}{(2s + 1)^2} \quad (5.310)$$

The following PID controller is supposed to be used to control the plant. Notice that the transfer function of the PID controller has been rendered causal thanks to the denominator $1 + 0.01s$ which appear within the derivative term $K_d s$:

$$C(s) = 340 + \frac{1000}{s} + \frac{60s}{1 + 0.01s} = \frac{6340s^2 + 35000s + 100000}{s^2 + 100s} \quad (5.311)$$

Without any saturation the closed-loop transfer function reads:

$$\frac{Y(s)}{R(s)} = \frac{F(s)C(s)}{1 + F(s)C(s)} = \frac{6340s^2 + 35000s + 100000}{4s^4 + 404s^3 + 6741s^2 + 35100s + 100000} \quad (5.312)$$

The 4 poles of the closed-loop transfer function are -81.6 , -13.3 and $-3 \pm j3.7$.

As far as the degree of the denominator of the controller $C(s)$ is 2, polynomial $D_{aw}(s)$ is chosen to be of degree 2. It is built from the 2 fastest poles of the closed-loop system obtained through the use of controller $C(s)$ and assuming no saturation, that is from -81.6 and -13.3 :

$$\begin{aligned} D_{aw}(s) &= (s + 81.6)(s + 13.3) \\ \Rightarrow \begin{cases} C_1(s) = \frac{N_C(s)}{D_{aw}(s)} = \frac{6340s^2 + 35000s + 100000}{s^2 + 95s + 1085} \\ C_2(s) = \frac{D_C(s) - D_{aw}(s)}{D_{aw}(s)} = \frac{5s - 1085}{s^2 + 95s + 1085} \end{cases} \end{aligned} \quad (5.313)$$

Figure 5.34 compares the results which are achieved with and without an antiwindup structure in the situation where the amplitude of control $u(t)$ is limited to ± 10 and where reference signal $r(t)$ is obtained through the prefilter $C_{pf}(s) = \frac{100000}{6340s^2 + 35000s + 100000}$ feeds by the unit step function. Obviously without any antiwindup structure the closed-loop system becomes oscillatory whereas with the antiwindup structure acceptable result is achieved despite a small delay in the time response. ■

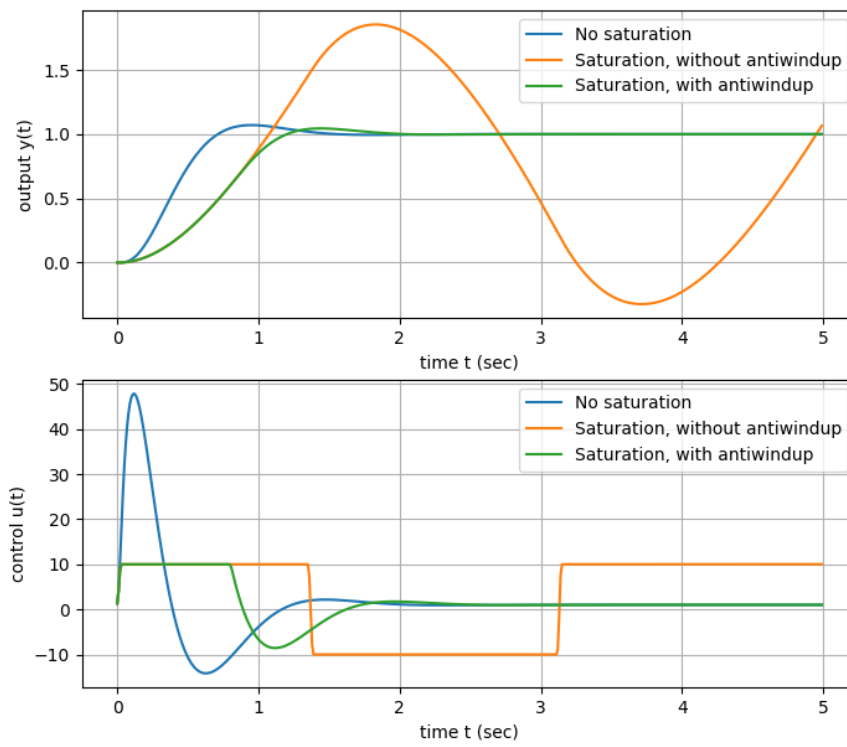


Figure 5.34: Comparison of different control strategies

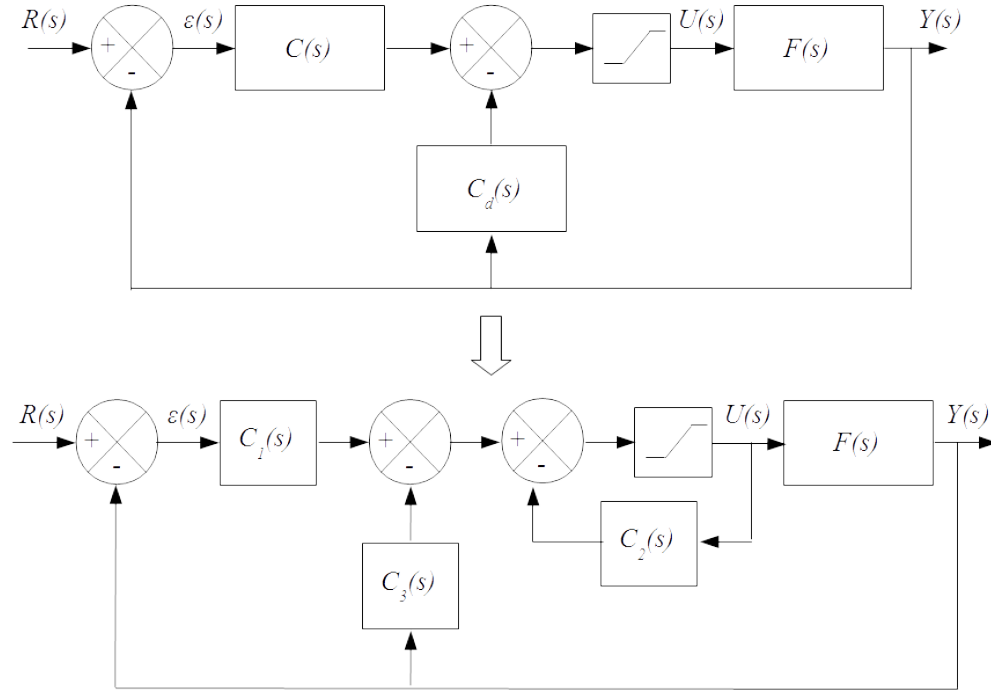


Figure 5.35: Anti-windup structure with two controllers

The same kind of structure may be extended when two controllers are used, as shown in Figure 5.35.

Assuming no saturation we have:

$$U(s) = C(s)\epsilon(s) - C_d(s)Y(s) \quad (5.314)$$

and:

$$\begin{aligned} U(s) &= -C_2(s)U(s) - C_3(s)Y(s) + C_1(s)\epsilon(s) \\ \Leftrightarrow U(s) &= \frac{C_1(s)}{1+C_2(s)}\epsilon(s) - \frac{C_3(s)}{1+C_2(s)}Y(s) \end{aligned} \quad (5.315)$$

Thus the two structures are equivalent assuming no saturation as soon as the following relations hold:

$$\begin{cases} C(s) = \frac{C_1(s)}{1+C_2(s)} \\ C_d(s) = \frac{C_3(s)}{1+C_2(s)} \end{cases} \quad (5.316)$$

Let $N_i(s)$ be the numerator of $C_i(s)$ and $D_i(s)$ be the denominator of $C_i(s)$ where $i = 1, 2, 3$. Similarly Let $N_C(s)$ be the numerator of $C(s)$ and $D_C(s)$ be the denominator of $C(s)$ and $N_d(s)$ be the numerator of $C_d(s)$ and $D_d(s)$ be the denominator of $C_d(s)$. From the preceding relations, we can make the following choices in order to get the transfer functions of the controllers in the

anti-windup structure with two controllers:

$$\begin{cases} D_1(s) = D_2(s) := D_{aw}(s) \\ N_1(s) = N_C(s) \\ N_2(s) = D_C(s) - D_1(s) \\ N_3(s) = N_d(s)(N_2(s) + D_2(s)) \\ D_3(s) = D_d(s)D_2(s) \end{cases} \quad (5.317)$$

Hurwitz polynomial $D_{aw}(s)$ is computed as shown in (5.306).

Chapter 6

Discrete time systems

6.1 Chapter overview

Thanks to the flexibility and the processing power of computers digital controllers are nowadays widely developed to control continuous time plants. When a digital controller is used the output of the digital controller is converted into an analog signal to control the actuator of the continuous time plant whereas the sensed output of the plant are converted into a digital sequence to be processed by the controller. Figure 6.1 illustrates how a digital controller control a continuous time plant.

Actually a digital controller is an algorithm running on a computer. It generates a digital control $u(k)$ at every time step t_k . The digital control $u(k)$ is then converted to an analog control signal $u(t)$ thanks to a Digital-to-Analog Converter (DAC). The analog signal $u(t)$ is applied to the continuous time plant whose transfer function is denoted $F(s)$. The behaviour of the plant is known thanks to the sensed output signal $y(t)$. This signal is then sampled by a Analog-to-Digital Converter (ADC) to produce the digital sequence $y(k)$ which is compared to the digital reference sequence $r(k)$. The error sequence $\epsilon(k)$ is then used to feed the digital controller $C(z)$. As shown in Figure 6.1, the Digital-to-Analog Converter (DAC), the Analog-to-Digital Converter (ADC) as

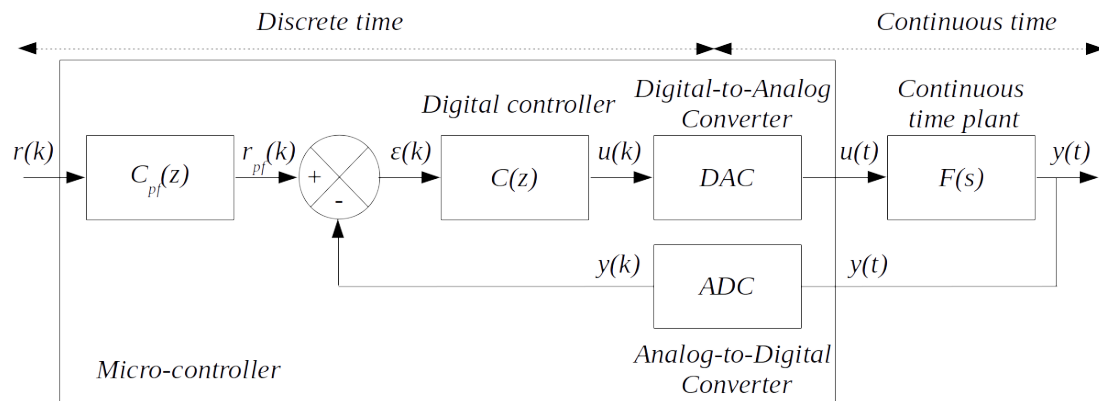


Figure 6.1: Digital control loop for continuous time plant

well as the digital controller $C(z)$ and the digital prefilter $C_{pf}(z)$ are usually embedded in a micro-controller.

Discrete time controllers for continuous time plants can be designed through designing first a continuous time controller for the plant and then deriving a discrete time equivalent controller that closely approximates the behaviour of the original continuous time controller. This approach is especially useful when an existing continuous time controller or a part of the controller is to be replaced with a discrete time controller.

From a practical point of view, the sampling frequency $F_s := 1/T$ is usually chosen to be between 5 and 25 greater than the expected closed loop bandwidth.

In that chapter we will introduce the \mathcal{Z} transform which is the dedicated tool to study discrete time system. Then we provide some tools to represent a continuous time plant with sampled input and output and to derive a discrete time controller that closely approximates the behaviour of the original continuous time controller. Performance analysis and stability of the digital control loop will also be tackled.

6.2 Nyquist–Shannon sampling theorem

Let $X(f)$ be the Fourier transform of a continuous time signal $x(t)$:

$$X(f) = \int_{-\infty}^{+\infty} x(t)e^{-j2\pi ft} dt \quad (6.1)$$

Let T be the sampling period. Its reciprocal is the sampling frequency F_s :

$$T = \frac{1}{F_s} \quad (6.2)$$

The Poisson summation formula indicates that the Fourier transform $X_s(f)$ of the samples $x(kT)$ is a periodic summation of $X(f)$ shifted by multiples of the sampling frequency $F_s = \frac{1}{T}$ and combined by addition:

$$X_s(f) = \sum_{k=-\infty}^{\infty} X\left(f - \frac{k}{T}\right) = \sum_{n=-\infty}^{\infty} T \cdot x(nT) e^{-i2\pi nTf} \quad (6.3)$$

The preceding relation indicates that $X_s(f)$ is a periodic spectrum which is also known as the discrete time Fourier transform (DTFT) of the sequence $T \cdot x(nT)$.

Frequency $\frac{F_s}{2}$ is called the Nyquist frequency.

The Nyquist–Shannon sampling theorem states that if a continuous time signal has a band limited spectrum within $[-B, +B]$ then its spectrum reconstruction from the spectrum $X_s(f)$ of the sampled signal is not altered as soon as the sampling rate F_s is larger than $2B$.

$$F_s \geq 2B \quad (6.4)$$

The Nyquist–Shannon sampling theorem is illustrated in Figure 6.2.

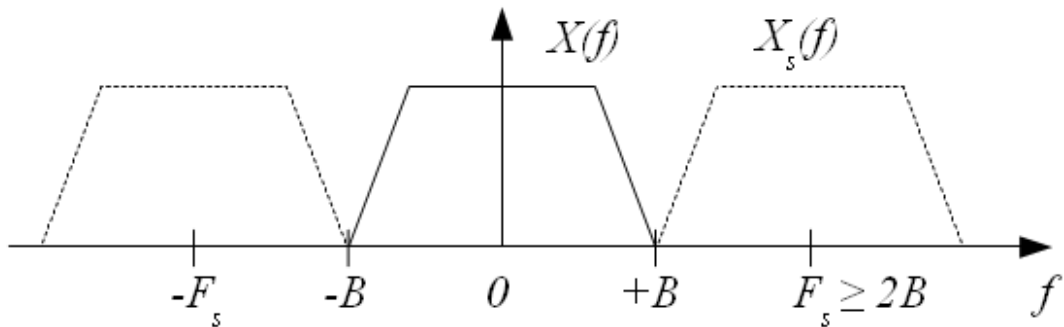


Figure 6.2: Nyquist–Shannon sampling theorem

Usually the sampling rate F_s is chosen between $6B$ and $24B^1$:

$$6B \leq F_s \leq 24B \quad (6.5)$$

When the spectrum of the continuous time signal is too wide (or when there is no bandlimit) the reconstruction from discrete to continuous time exhibits imperfections known as aliasing. To overcome the aliasing phenomenon a low pass anti-aliasing continuous time filter is introduced just before the sampler (Analog-to-Digital Converter) in order to reduce the higher frequency components in the analog signal.

6.3 Reference sequences

The aim of this section is to present important sequences which are currently use to study discrete time systems: more specifically we will focus on the unit impulse, the sampled unit step and the sampled exponential.

Discrete-time signals x_s are represented mathematically as sequences of numbers. Denoting by $x(k)$ the k^{th} number in the causal sequence x_s we get:

$$x_s = \{x(0), x(1), \dots, x(k), \dots\} \quad (6.6)$$

A sequence $x(k)$ is said to be causal when the values of the sequence are null $\forall k < 0$.

In a practical setting, such sequences arise from periodic sampling of a continuous time (or analog) signal $x(t)$. In that case, the numeric value of the k^{th} number in the sequence is equal to the value of the continuous time signal $x(t)$ at time kT :

$$x(k) := x(kT) \quad (6.7)$$

¹Peaucelle D., Systèmes à temps discret: Commande numérique des procédés, notes de cours, 2003

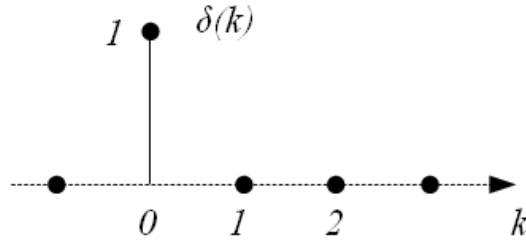


Figure 6.3: Unit impulse

6.3.1 Unit impulse

The unit impulse is denoted $\delta(k)$; this sequence is zero everywhere except at $k = 0$ where the value of the sequence is 1.

$$\delta(k) = \begin{cases} 1, & k = 0 \\ 0, & k \neq 0 \end{cases} \quad (6.8)$$

The schematic representation of the unit impulse is depicted on figure 6.3.

The following properties of the unit impulse are very close from the properties of the Dirac delta function in the continuous time domain:

- Multiplication between a sequence $x(k)$ and the unit impulse translated by $k_0 \in \mathbb{N}$:

$$x(k)\delta(k - k_0) = x(k_0)\delta(k - k_0) \quad (6.9)$$

- Convolution: the discrete time convolution $*$ between two sequences $x(k)$ and $y(k)$ is defined by:

$$x(k)*y(k) = \sum_{m=-\infty}^{\infty} x(m)y(k - m) \quad (6.10)$$

The discrete time convolution is a commutative product:

$$x(k)*y(k) = y(k)*x(k) \quad (6.11)$$

Indeed, changing m by $l := k - m$ and using the fact that a summation can be done either from the last term to the first term or vice-versa, i.e. $\sum_{l=+\infty}^{-\infty} a_l = \sum_{l=-\infty}^{+\infty} a_l$, we get:

$$\begin{cases} x(k)*y(k) = \sum_{m=-\infty}^{\infty} x(m)y(k - m) \\ l := k - m \end{cases} \quad (6.12)$$

$$\Rightarrow x(k)*y(k) = \sum_{l=+\infty}^{-\infty} x(k - l)y(l)$$

$$\Leftrightarrow x(k)*y(k) = \sum_{l=-\infty}^{+\infty} x(k - l)y(l) = y(k)*x(k)$$

When the sequences $x(k)$ and $y(k)$ are causal, we get:

$$\begin{cases} x(k) = 0 \quad \forall k < 0 \\ y(k) = 0 \quad \forall k < 0 \Leftrightarrow y(k - m) = 0 \quad \forall (k - m) < 0 \Leftrightarrow m > k \end{cases} \quad (6.13)$$

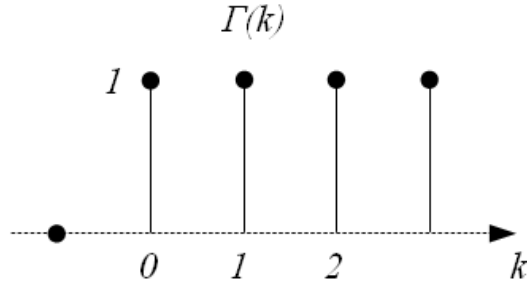


Figure 6.4: Sampled unit step

Thus the discrete time convolution reduces to:

$$x(k)*y(k) = \sum_{m=0}^k x(m)y(k-m) \quad (6.14)$$

As far as the discrete time convolution is concerned, the unit impulse satisfies the following property:

$$x(k)*\delta(k-k_0) = x(k-k_0) \quad \forall k_0 \in \mathbb{N} \quad (6.15)$$

Taking $k_0 = 0$ we get:

$$x(k)*\delta(k) = x(k) \quad (6.16)$$

From the preceding equation, it is clear that the unit impulse is the neutral element for discrete time convolution. The preceding equation can also be stated as follows for causal sequences:

$$x(k) = \sum_{m=0}^k x(m)\delta(k-m) \quad (6.17)$$

6.3.2 Sampled unit step

The sampled unit step which will be denoted by $\Gamma(k)$ is a causal sequence whose value is zero for negative values of k and one for positive values of k :

$$\Gamma(k) = \begin{cases} 1 & \forall k \geq 0 \\ 0 & \forall k < 0 \end{cases} \quad (6.18)$$

The sampled unit step is depicted on figure 6.4.

The sampled unit step is linked to the unit impulse by the following relation:

$$\Gamma(k) = \sum_{m=0}^{\infty} \delta(k-m) \quad (6.19)$$

6.3.3 Sampled exponential

The sampling of the complex exponential function $e^{j\omega t}$ reads:

$$e^{j\omega t} \Big|_{t=kT} = e^{j\alpha k} \quad \text{where } \alpha := \omega T \quad (6.20)$$

Setting $\omega = 2\pi f$ where f is the ordinary frequency (units of Hz or equivalently cycles per second) and using the fact that $-\pi \leq \alpha \leq +\pi$ we get:

$$-\pi \leq 2\pi fT \leq +\pi \quad (6.21)$$

In the following we will denote α the reduced frequency. This is linked to the ordinary frequency f and the natural frequency ω by the following relations where T is the sampling period:

$$\alpha = 2\pi fT = \omega T \quad (6.22)$$

Thus the sampled exponential sequence is defined by:

$$x(k) = e^{j\alpha k} \quad \text{where } -\pi \leq \alpha \leq +\pi \quad (6.23)$$

The preceding relation reads $-\pi \leq \alpha \leq +\pi$ and comes from the Nyquist–Shannon sampling theorem. Indeed setting $f = B$ and having in mind that the sampling period T represents the interval between samples and is the reciprocal of the sampling frequency F_s yields:

$$\begin{cases} 2\pi BT \leq \pi \\ T = \frac{1}{F_s} \end{cases} \Rightarrow F_s \geq 2B \quad (6.24)$$

We recognize in the preceding equation the Nyquist–Shannon sampling theorem illustrated in Figure 6.1.

6.4 \mathcal{Z} transform

6.4.1 Definition

Given the causal sequence $\{x(0), x(1), \dots, x(k), \dots\}$ we define its \mathcal{Z} transform by:

$$\mathcal{Z}[x(k)] = \sum_{k=0}^{\infty} x(k)z^{-k} \quad (6.25)$$

The region of convergence is the set of values of z in the complex plane for which the \mathcal{Z} transform summation converges. Similarly to the Laplace transform the region of convergence of a \mathcal{Z} transform is never computed from a practical point of view.

Example 6.1. Compute the \mathcal{Z} transform of the sequence $x(k) = a^k \quad \forall k \geq 0$.

We get:

$$\mathcal{Z}[a^k] = \sum_{k=0}^{\infty} a^k z^{-k} = \sum_{k=0}^{\infty} \left(\frac{a}{z}\right)^k \quad (6.26)$$

We recall that the sum of the first n terms of a geometric series where a_0 is the first term of the series, and q is the common ratio is:

$$a_0 + a_0q + a_0q^2 + a_0q^3 + \cdots + a_0q^{n-1} = \sum_{k=0}^{n-1} a_0q^k = a_0 \frac{1 - q^n}{1 - q} \quad (6.27)$$

As n goes to infinity, the modulus of the common ratio q must be less than one for the series to converge. Then the sum then becomes:

$$\sum_{k=0}^{\infty} a_0q^k = a_0 \frac{1}{1 - q} \quad \text{if } \|q\| < 1 \quad (6.28)$$

Applying the preceding result to the computation of the \mathcal{Z} transform of the sequence $x(k) = a^k \quad \forall k \geq 0$ reads:

$$\mathcal{Z} [a^k] = \frac{1}{1 - \frac{a}{z}} = \frac{z}{z - a} \quad (6.29)$$

Note that the region of convergence is defined by $\left\| \frac{a}{z} \right\| < 1$.

Taking $a = 1$ leads to the \mathcal{Z} transform of the sampled unit step $\Gamma(k)$:

$$\mathcal{Z} [\Gamma(k)] = \frac{z}{z - 1} \quad (6.30)$$

■

6.4.2 Properties

This section presents the main properties of the \mathcal{Z} transform without demonstration. In the following $x(k)$ is a causal sequence and $X(z)$ is its \mathcal{Z} transform.

Linearity

For any sequences $x_1(k)$ and $x_2(k)$ and any constants a_1 and a_2 in \mathbb{R} or \mathbb{C} :

$$\mathcal{Z} [a_1x_1(k) + a_2x_2(k)] = a_1X_1(z) + a_2X_2(z) \quad (6.31)$$

Time delay

As soon as $x(k)$ is a causal sequence we get:

$$\mathcal{Z} [x(k - 1)] = z^{-1}X(z) \quad \text{s.t. } x(k) = 0 \quad \forall k < 0 \quad (6.32)$$

Consequently z^{-1} is interpreted as the one sampling period delay operator. More generally we have:

$$\mathcal{Z} [x(k - n)] = z^{-n}X(z) \quad \forall n \in \mathbb{N}^+ \quad \text{s.t. } x(k) = 0 \quad \forall k < 0 \quad (6.33)$$

Time advance

$$\mathcal{Z}[x(k+1)] = z^1 X(z) - zx(0) \quad (6.34)$$

More generally we have:

$$\mathcal{Z}[x(k+n)] = z^n X(z) - z^n x(0) - z^{n-1} x(1) - \dots - zx(n-1) \quad \forall n \in \mathbb{N}^+ \quad (6.35)$$

Initial value theorem

$$x(0) = \lim_{z \rightarrow \infty} X(z) \quad (6.36)$$

Example 6.2. Knowing the \mathcal{Z} transform of $x(k) = a^k$ is $\frac{z}{z-a}$ check that $x(0) = 1$. We check that:

$$x(0) = \lim_{z \rightarrow \infty} X(z) = \lim_{z \rightarrow \infty} \frac{z}{z-a} = 1 \quad (6.37)$$

■

Final value theorem

Assuming that the value of $x(k)$ exists when $k \rightarrow \infty$ then

$$\lim_{k \rightarrow \infty} x(k) = \lim_{z \rightarrow 1} (1 - z^{-1}) X(z) \quad (6.38)$$

Example 6.3. Knowing that the \mathcal{Z} transform of the sampled unit step $\Gamma(k)$ is $\frac{z}{z-1}$ compute $\lim_{k \rightarrow \infty} \Gamma(k)$.

We check that:

$$\lim_{k \rightarrow \infty} \Gamma(k) = \lim_{z \rightarrow 1} (1 - z^{-1}) \frac{z}{z-1} = \lim_{z \rightarrow 1} \frac{z-1}{z} \frac{z}{z-1} = 1 \quad (6.39)$$

■

It is worth noticing that the final value theorem gives the correct limit as soon as the value of sequence $x(k)$ exists when $k \rightarrow \infty$.

Example 6.4. If we apply the final value theorem on the sequence a^k we get:

$$\lim_{z \rightarrow 1} (1 - z^{-1}) \frac{z}{z-a} = \lim_{z \rightarrow 1} \frac{z-1}{z} \frac{z}{z-a} = \begin{cases} 0 & \forall a \neq 1 \\ 1 & \text{if } a = 1 \end{cases} \quad (6.40)$$

As far as a^k has no limit as soon as $\|a\| > 1$ it is clear that result provided by the final value theorem is not correct when $\|a\| > 1$.

■

Multiplication by exponential

$$\mathcal{Z}[a^{-k}x(k)] = \mathcal{Z}[x(k)]|_{z \rightarrow az} = X(az) \quad (6.41)$$

Example 6.5. From the \mathcal{Z} transform of the sampled unit step $\Gamma(k)$ find the \mathcal{Z} transform of the exponential sequence $x(k) = e^{-akT} \quad \forall k \in \mathbb{N}^+$

$$\begin{aligned} x(k) &= (e^{aT})^{-k} \Gamma(k) \\ \Rightarrow X(z) &= \mathcal{Z}[\Gamma(k)]|_{z \rightarrow ze^{aT}} = \frac{z}{z-1} \Big|_{z \rightarrow ze^{aT}} = \frac{ze^{aT}}{ze^{aT}-1} = \frac{z}{z-e^{-aT}} \end{aligned} \quad (6.42)$$

■

Multiplication by k

$$\mathcal{Z}[kx(k)] = -z \frac{d}{dz} \mathcal{Z}[x(k)] = -z \frac{d}{dz} X(z) \quad (6.43)$$

Example 6.6. Find the \mathcal{Z} transform of the sequence $x(k) = kT \quad \forall k \in \mathbb{N}^+$

$$\begin{aligned} x(k) = Tk\Gamma(k) \Rightarrow X(z) &= -Tz \frac{d}{dz} \mathcal{Z}[\Gamma(k)] = -Tz \frac{d}{dz} \left(\frac{z}{z-1} \right) \\ &= -Tz \frac{z-1-z}{(z-1)^2} = \frac{Tz}{(z-1)^2} \end{aligned} \quad (6.44)$$

■

Example 6.7. From the preceding example find the \mathcal{Z} transform of the sequence $x(k) = (kT)^2 \quad \forall k \in \mathbb{N}^+$

$$\begin{aligned} x(k) = Tk \cdot kT \Rightarrow X(z) &= -Tz \frac{d}{dz} \mathcal{Z}[kT] = -Tz \frac{d}{dz} \left(\frac{Tz}{(z-1)^2} \right) \\ &= -T^2 z \frac{(z-1)^2 - 2z(z-1)}{(z-1)^4} = -T^2 z \frac{(z-1) - 2z}{(z-1)^3} \\ &= \frac{T^2 z(z+1)}{(z-1)^3} \end{aligned} \quad (6.45)$$

■

Convolution

We have seen that the discrete time convolution $*$ between two causal sequences $x(k)$ and $y(k)$ is defined by:

$$x(k)*y(k) = \sum_{m=0}^k x(m)y(k-m) \quad (6.46)$$

Then the \mathcal{Z} transform of the convolution between two causal sequences $x(k)$ and $y(k)$ is the product of the \mathcal{Z} transforms of $x(k)$ and $y(k)$:

$$\mathcal{Z}[x(k)*y(k)] = X(z)Y(z) \quad (6.47)$$

When $y(k)$ is the unit impulse, $y(k) = \delta(k)$, and using the fact that the unit impulse is the neutral element for discrete time convolution, we can write:

$$\mathcal{Z}[x(k)*\delta(k)] = \mathcal{Z}[x(k)] \Leftrightarrow X(z)\mathcal{Z}[\delta(k)] = X(z) \quad (6.48)$$

Simplifying by $X(z)$, we get the \mathcal{Z} transform of the unit impulse $\delta(k)$:

$$\mathcal{Z}[\delta(k)] = 1 \quad (6.49)$$

6.4.3 Usual \mathcal{Z} transform pairs

Usual \mathcal{Z} transform pairs are provided in Table 6.1.

$x(k) \forall k \in \mathbb{N}^+$	$X(z)$
$\delta(k)$	1
$\Gamma(k)$ (step function)	$\frac{z}{z-1}$
kT	$\frac{Tz}{(z-1)^2}$
$(kT)^2$	$\frac{T^2 z(z+1)}{(z-1)^3}$
a^k	$\frac{z}{z-a}$
e^{-akT}	$\frac{z}{z-e^{-aT}}$
$\sin(\omega_0 kT)$	$\frac{z \sin(\omega_0 T)}{z^2 - 2z \cos(\omega_0 T) + 1}$
$\cos(\omega_0 kT)$	$\frac{z(z - \cos(\omega_0 T))}{z^2 - 2z \cos(\omega_0 T) + 1}$

Table 6.1: Usual \mathcal{Z} transform pairs

6.4.4 Inverse \mathcal{Z} transform

Basically there exists three methods to compute inverse \mathcal{Z} transform:

- Method based on Cauchy's residue theorem;
- Method based on partial fraction expansion;
- Method based on long division.

Cauchy's residue theorem

From the residue theorem coming from complex analysis it can be demonstrated that as soon as $X(z)$ is a proper rational fraction (the degree of its denominator is greater *or equal* to the degree of its numerator) its inverse \mathcal{Z} transform reads:

$$x(k) = \mathcal{Z}^{-1}[X(z)] = \sum_i \text{Res}_{z=\lambda_i} [z^{k-1} X(z)] \quad \forall k \in \mathbb{N}^+ \quad (6.50)$$

The residue $\text{Res}_{z=\lambda_i} [z^{k-1} X(z)]$ shall be computed around each pole λ_i of $X(z)$. Assuming that λ_i is a pole of multiplicity n_i then the residue of $z^{k-1} X(z)$ around pole λ_i is given by:

$$\text{Res}_{z=\lambda_i} [z^{k-1} X(z)] = \frac{1}{(n_i - 1)!} \frac{d^{n_i-1}}{dz^{n_i-1}} \left((z - \lambda_i)^{n_i} z^{k-1} X(z) \right) \Big|_{z=\lambda_i} \quad (6.51)$$

Let $n_0(k)$ be the multiplicity of the root 0 of the polynomial corresponding to the numerator of $z^{k-1} X(z)$. Then $x(k)$ is defined $\forall n_0(k) \geq 0$.

Example 6.8. Find the inverse \mathcal{Z} transform of $X(z) = \frac{z}{z-\alpha}$

$X(z)$ has a unique pole at $\lambda_1 = \alpha$ with multiplicity $n_1 = 1$. The residue of $z^{k-1} X(z)$ around pole λ_1 of multiplicity $n_1 = 1$ reads:

$$\begin{aligned} x(k) = \text{Res}_{z=\alpha} [z^{k-1} X(z)] &= \frac{1}{(1-1)!} \frac{d^{1-1}}{dz^{1-1}} \left((z - \alpha)^1 z^{k-1} X(z) \right) \Big|_{z=\alpha} \\ &= (z - \alpha) z^{k-1} X(z) \Big|_{z=\alpha} \\ &= z^k \Big|_{z=\alpha} \\ \Leftrightarrow x(k) &= \alpha^k \quad \forall k \in \mathbb{N}^+ \end{aligned} \quad (6.52)$$

The numerator of $z^{k-1}X(z)$ is z^k , which has root 0 with multiplicity $n_0(k) = k$. Thus $x(k)$ is defined $\forall n_0(k) = k \geq 0$. ■

Example 6.9. Find the inverse \mathcal{Z} transform of $X(z) = \frac{Tz}{(z-1)^2}$

$X(z)$ has a unique pole at $\lambda_1 = 1$ with multiplicity $n_1 = 2$. The residue of $z^{k-1}X(z)$ around pole λ_1 of multiplicity $n_1 = 2$ reads:

$$\begin{aligned} x(k) = \text{Res}_{z=\alpha} [z^{k-1}X(z)] &= \frac{1}{(2-1)!} \frac{d^{2-1}}{dz^{2-1}} \left((z-1)^2 z^{k-1}X(z) \right) \Big|_{z=1} \\ &= \frac{d}{dz} Tz^k \Big|_{z=1} \\ &= Tkz^{k-1} \Big|_{z=1} \\ \Leftrightarrow x(k) &= kT \quad \forall k \in \mathbb{N}^+ \end{aligned} \quad (6.53)$$

The numerator of $z^{k-1}X(z)$ is z^k , which has root 0 with multiplicity $n_0(k) = k$. Thus $x(k)$ is defined $\forall n_0(k) = k \geq 0$. ■

Partial fraction expansion

The computation of the partial fraction expansion of $\frac{X(z)}{z}$. Then by multiplying by z each term of the partial fraction expansion we should recognize by inspection of Table 6.1 the associated pairs. It is worth noticing that factor z systemically appears on the numerator of the pairs in Table 6.1 (except for the \mathcal{Z} transform of the unit impulse). By making the partial fraction expansion of $\frac{X(z)}{z}$ (and not $X(z)$) and then multiplying each term of the partial fraction expansion by z we get a sum of factors where z systemically appears on the numerator and Table 6.1 can be used. You may also use the residue theorem to compute the inverse \mathcal{Z} transform of each term of the partial fraction expansion.

Long division

The polynomial long division of $X(z)$ written on the following form:

$$X(z) = \frac{b_n + b_{n-1}z^{-1} + \dots + b_0z^{-n}}{a_n + a_{n-1}z^{-1} + \dots + a_0z^{-n}} \quad (6.54)$$

Indeed the division according to the increasing power of z^{-1} allows to identify $x(k)$ with the quotient of the long division which reads $\sum_{k=0}^{\infty} x(k)z^{-k}$. It is worth noticing that as far as the \mathcal{Z} transform involves z^{-k} the long division has to be done according to the increasing power of z^{-1} and not of z . In addition the long-division approach does not yield a closed-form solution because the steps to compute each coefficient of the quotients shall be done indefinitely to get the complete sequence $x(k)$.

Example 6.10. Find the 3 first values of the sequence whose \mathcal{Z} transform is $X(z) = \frac{Tz}{(z-1)^2}$

We get:

$$X(z) = \frac{Tz}{z^2 - 2z + 1} = \frac{Tz^{-1}}{1 - 2z^{-1} + z^{-2}} \quad (6.55)$$

Then we compute the quotient of the polynomial long division (i.e. division according to the increasing power of z^{-1}) of $X(z)$:

$$\frac{Tz^{-1}}{-(Tz^{-1} - 2Tz^{-2} + Tz^{-3})} \left| \frac{1 - 2z^{-1} + z^{-2}}{Tz^{-1} + 2Tz^{-2} + \dots} \right. \quad (6.56)$$

Then we identify the quotient $Tz^{-1} + 2Tz^{-2} + \dots$ of the polynomial long division with $\sum_{k=0}^{\infty} x(k)z^{-k}$. In that example this leads to the first 3 values of $x(k)$:

$$Tz^{-1} + 2Tz^{-2} + \dots = \sum_{k=0}^{\infty} x(k)z^{-k} \Rightarrow \begin{cases} x(0) = 0 \\ x(1) = T \\ x(2) = 2T \\ \vdots \end{cases} \quad (6.57)$$

It is clear that the long-division approach does not yield a closed-form solution because the steps to compute each coefficient of the quotients shall be done indefinitely to get the complete sequence $x(k)$. ■

6.4.5 Relation with Laplace transform

Laplace transform of a sampled continuous time signal

The \mathcal{Z} transform can be seen as the Laplace transform of a sampled causal continuous time signal where $z = e^{sT}$. Indeed the sampling of a causal continuous time signal $x(t)$ to a discrete time sequence $x_s(k)$ is mathematically modelled thanks to the following product which involves the Dirac delta function $\delta(t)$ and the sampling interval T :

$$x_s(k) := x(t)|_{t=kT} = \sum_{k=0}^{\infty} x(t)\delta(t - kT) = \sum_{k=0}^{\infty} x(kT)\delta(t - kT) \quad (6.58)$$

By taking the Laplace transform of this equation we get:

$$\begin{aligned} \mathcal{L}[x_s(k)] &= \mathcal{L}[\sum_{k=0}^{\infty} x(kT)\delta(t - kT)] \\ &= \sum_{k=0}^{\infty} \mathcal{L}[x(kT)\delta(t - kT)] \\ &= \sum_{k=0}^{\infty} x(kT)\mathcal{L}[\delta(t - kT)] \\ &= \sum_{k=0}^{\infty} x(kT)e^{-skT} \\ &= \sum_{k=0}^{\infty} x(kT)(e^{sT})^{-k} \end{aligned} \quad (6.59)$$

Let's define:

$$z = e^{sT} \quad (6.60)$$

Consequently setting $z = e^{sT}$ in Laplace transform of a sampled causal continuous time signal $x(t)$ leads to the \mathcal{Z} transform of the causal sequence $\{x(0), x(1), \dots, x(k), \dots\}$

$$\mathcal{L}[x_s(k)]_{z=e^{sT}} = \sum_{k=0}^{\infty} x(kT)z^{-k} := \mathcal{Z}[x(k)] \quad (6.61)$$

Conversely let's consider a continuous time signal $x(t)$ whose Laplace transform exhibits distinct poles only. The partial fraction expansion of $X(s)$ reads:

$$X(s) = \frac{N(s)}{\prod_{i=1}^n (s - \lambda_i)} = \sum_{i=1}^n \frac{a_i}{s - \lambda_i} \quad (6.62)$$

Taking the inverse Laplace transform of $X(s)$ yields the expression of $x(t)$ as a sum of exponential functions:

$$\mathcal{L}^{-1}[X(s)] = x(t) = \sum_{i=1}^n a_i e^{\lambda_i t} \quad \forall t \geq 0 \quad (6.63)$$

Then the sampling of the continuous time signal $x(t)$ to the discrete time sequence $x_s(k)$ leads to the following expression:

$$x_s(k) := x(t)|_{t=kT} = \sum_{i=1}^n a_i e^{\lambda_i kT} \quad \forall k \in \mathbb{N}^+ \quad (6.64)$$

Knowing that $\mathcal{Z}[a^k] = \frac{z}{z-a}$ and using the fact that the \mathcal{Z} transform is a linear operator the \mathcal{Z} transform of $x_s(k)$ reads:

$$\begin{aligned} x_s(k) &= \sum_{i=1}^n a_i (e^{\lambda_i T})^k \\ \Rightarrow X(z) = \mathcal{Z}[x_s(k)] &= \sum_{i=1}^n a_i \mathcal{Z}[(e^{\lambda_i T})^k] = \sum_{i=1}^n a_i \frac{z}{z - e^{\lambda_i T}} \end{aligned} \quad (6.65)$$

As a consequence each simple fraction $\frac{1}{s - \lambda_i}$ in the Laplace transform domain is associated in the \mathcal{Z} transform domain to the term $\frac{z}{z - e^{\lambda_i T}}$:

$$\frac{1}{s - \lambda_i} \xrightarrow{\mathcal{Z}} \frac{z}{z - e^{\lambda_i T}} \quad (6.66)$$

In others words each pole λ_i in the Laplace transform domain is associated in the \mathcal{Z} transform domain to the pole $e^{\lambda_i T}$:

$$\lambda_i \xrightarrow{\mathcal{Z}} e^{\lambda_i T} \quad (6.67)$$

This relation can also be read as $z = e^{sT}$.

From Laplace transform to \mathcal{Z} transform

The purpose of this section is to provide the relation which enables to get the \mathcal{Z} transform of the discrete time sequence $x(k)$ from the Laplace transform of the related continuous time signal $x(t)$. Thus starting from $X(s)$, which is assumed to be a strictly proper rational fraction, we have to apply the inverse Laplace transform to get the continuous time signal $x(t)$. Then $x(t)$ is sampled at every time step kT to obtain the discrete time sequence $x(k)$ and the \mathcal{Z} transform is applied to get $X(z)$. This is denoted as follows:

$$X(z) = \mathcal{Z} [X(s)] \quad (6.68)$$

The residue theorem applied on this specific inversion problem leads to the following result:

$$\mathcal{Z} [X(s)] = \sum_i Res_{z=\lambda_i} \left[\frac{X(s)}{1 - z^{-1}e^{sT}} \right] \quad (6.69)$$

Where the residue $Res_{z=\lambda_i} \left[\frac{X(s)}{1 - z^{-1}e^{sT}} \right]$ shall be computed around each pole λ_i of $X(s)$. Assuming that λ_i is a pole of multiplicity n_i then the residue of $\frac{X(s)}{1 - z^{-1}e^{sT}}$ around pole λ_i is given by:

$$Res_{s=\lambda_i} \left[\frac{X(s)}{1 - z^{-1}e^{sT}} \right] = \frac{1}{(n_i - 1)!} \frac{d^{n_i-1}}{ds^{n_i-1}} \left((s - \lambda_i)^{n_i} \frac{X(s)}{1 - z^{-1}e^{sT}} \right) \Big|_{s=\lambda_i} \quad (6.70)$$

Example 6.11. Let's compute the \mathcal{Z} transform of $X(s) = \frac{1}{s^2}$.

It is clear that $X(s)$ has a single pole, $\lambda_1 = 0$, with multiplicity $n_1 = 2$. The residue of $\frac{X(s)}{1 - z^{-1}e^{sT}}$ around pole $\lambda_1 = 0$ is given by:

$$\begin{aligned} x(k) = Res_{s=0} \left[\frac{X(s)}{1 - z^{-1}e^{sT}} \right] &= \frac{1}{(n_1-1)!} \frac{d^{n_1-1}}{ds^{n_1-1}} \left(s^{n_1} \frac{X(s)}{1 - z^{-1}e^{sT}} \right) \Big|_{s=0} \\ &= \frac{d}{ds} \frac{1}{1 - z^{-1}e^{sT}} \Big|_{s=0} \\ &= \frac{z^{-1}T e^{sT}}{(1 - z^{-1}e^{sT})^2} \Big|_{s=0} \\ &= \frac{z^{-1}T}{(1 - z^{-1})^2} \end{aligned} \quad (6.71)$$

By multiplying the numerator and the denominator by z^2 we finally get:

$$x(k) = \frac{Tz}{(z - 1)^2} \quad \forall k \geq 0 \quad (6.72)$$

In order to check this result let's take the inverse Laplace transform of $X(s)$. We get:

$$x(t) = \mathcal{L}^{-1} [X(s)] = t \quad (6.73)$$

Then the sequence $x(k)$ is obtained by sampling the continuous time signal $x(t)$ at each time step $t = kT$. We get:

$$x(k) = x(t)|_{t=kT} = kT \quad (6.74)$$

Sequence $x(k)$ can be written as follows where $\Gamma(k)$ is the unit step sequence:

$$x(k) = kT = Tk\Gamma(k) \quad (6.75)$$

The property of the \mathcal{Z} transform with respect to the multiplication by k can be applied to get the \mathcal{Z} transform of sequence $x(k)$:

$$\begin{aligned} \mathcal{Z}[x(k)] &= \mathcal{Z}[Tk\Gamma(k)] \\ &= T\mathcal{Z}[k\Gamma(k)] \\ &= -Tz\frac{d}{dz}\Gamma(z) \end{aligned} \quad (6.76)$$

Where:

$$\Gamma(z) = \mathcal{Z}[\Gamma(k)] = \frac{z}{z-1} \quad (6.77)$$

We finally get:

$$\mathcal{Z}[x(k)] = -Tz\frac{d}{dz}\frac{z}{z-1} = \frac{Tz}{(z-1)^2} \quad (6.78)$$

We obviously get the same result than the result obtained thanks to the residue theorem. ■

6.5 Discrete linear time invariant systems

6.5.1 Discrete time transfer function

For systems that have only one input and one output it is frequently convenient to work with an input-output description of the system. For discrete linear time systems this consists of a single n^{th} order linear difference equation relating the sampled output $y(k)$ to the sampled input $u(k)$ of the system:

$$\begin{aligned} a_0y(k) + a_1y(k-1) + \dots + a_ny(k-n) \\ = b_0u(k) + b_1u(k-1) + \dots + b_mu(k-m) \end{aligned} \quad (6.79)$$

The constant coefficients a_i and b_i are defined by the system parameters. For causal systems we have:

$$m \leq n \quad (6.80)$$

When the difference equation reflects the input output relation of a digital controller, the algorithm implemented on the computer (sometimes a DSP: digital signal processor) computes at each time step the output $y(k)$ as a function of the previous values of the output and also current and past values of the input:

$$\begin{aligned} y(k) = \frac{1}{a_0} (b_0u(k) + b_1u(k-1) + \dots + b_mu(k-m)) \\ - \frac{1}{a_0} (a_1y(k-1) + \dots + a_ny(k-n)) \end{aligned} \quad (6.81)$$

Assuming that sequences $y(k)$ and $u(k)$ are causal (so that the time response of the discrete time system is invariant) and applying the time delay property of the \mathcal{Z} transform, the \mathcal{Z} transform of difference equation (6.79) reads:

$$\begin{aligned} a_0Y(z) + a_1z^{-1}Y(z) + \cdots + a_nz^{-n}Y(z) \\ = b_0U(z) + b_1z^{-1}U(z) + \cdots + b_mz^{-m}U(z) \end{aligned} \quad (6.82)$$

Analogously to the continuous time domain, the discrete time transfer function $H(z)$ of a discrete linear time invariant system is defined to be the ratio of the \mathcal{Z} transform of the output sequence $y(k)$ to the \mathcal{Z} transform of the input sequence $u(k)$:

$$H(z) = \frac{Y(z)}{U(z)} = \frac{b_0 + b_1z^{-1} + \cdots + b_m}{a_0 + a_1z^{-1} + \cdots + a_nz^{-n}} \quad (6.83)$$

This transfer function is usually rationalized by multiplying top and bottom by z^n :

$$H(z) = \frac{Y(z)}{U(z)} = \frac{b_0z^n + b_1z^{n-1} + \cdots + b_mz^{n-m}}{a_0z^n + a_1z^{n-1} + \cdots + a_n} \quad (6.84)$$

As far as the \mathcal{Z} transform of the unit impulse $\delta(k)$ is 1, it is worth noticing that the \mathcal{Z} transform of the impulse response of a discrete linear time invariant system is its transfer function (as in the continuous time case):

$$\begin{cases} Y(z) = H(z)U(z) \\ u(k) = \delta(k) \Rightarrow U(z) = 1 \end{cases} \Rightarrow Y(z) = H(z) \quad (6.85)$$

6.5.2 Stability analysis

Routh–Hurwitz based stability criterion

Analysis tools based on the Laplace domain also apply to \mathcal{Z} domain thanks to specific adjustments. As far as stability is concerned the symbolic relation $z = e^{sT}$ is quite useful to analyze discrete time systems. Indeed we know that a continuous time system is stable if and only if all the poles of its transfer function have negative real part. Applying relation $z = e^{sT}$ shows that a discrete time system is stable if and only if all the poles of its transfer function have a magnitude less than 1.

As an example consider a first order discrete time system defined by the following difference equation where a and b are real parameters:

$$y(k) + ay(k-1) = bu(k-1) \quad (6.86)$$

The transfer function $H(z)$ of this system is obtained by taking the \mathcal{Z} transform of the difference equation. Assuming that $y(k)$ and $u(k)$ are causal sequences this yields:

$$\begin{aligned} Y(z) + az^{-1}Y(z) &= bz^{-1}U(z) \\ \Rightarrow H(z) = \frac{Y(z)}{U(z)} &= \frac{bz^{-1}}{1+az^{-1}} = \frac{b}{z+a} \end{aligned} \quad (6.87)$$

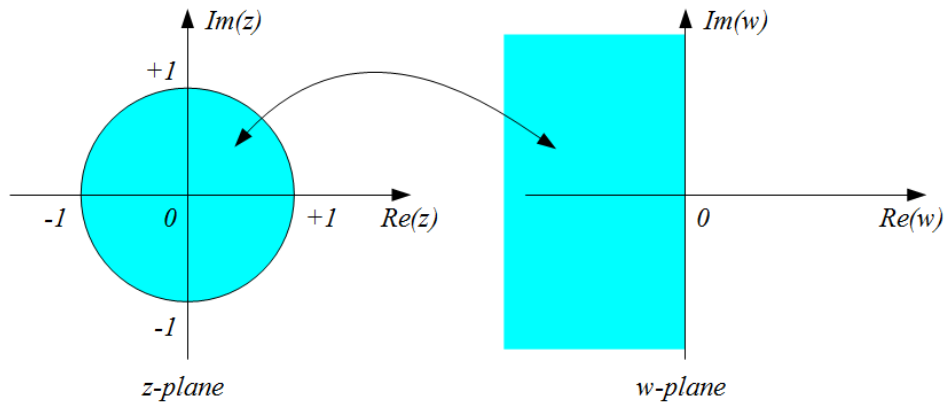


Figure 6.5: Homographic transformation

It is clear that all the poles of its transfer function have a magnitude less than 1 as soon as $\|a\| < 1$; as far as parameter a is a real this simply reads:

$$-1 < a < +1 \quad (6.88)$$

Analogously to the continuous time domain we may apply Routh criterion to analyze discrete time stability. Nevertheless some adjustment shall be made in order that the criterion shall indicate the number of poles outside the unity circle rather than the number of poles with positive real part. This adjustment is achieved thanks to the following homographic transformation which matches the imaginary axis of the complex w -plane to the unit circle of the complex z -plane as illustrated in Figure 6.5.

$$w = \frac{z-1}{z+1} \Leftrightarrow z = \frac{1+w}{1-w} \quad (6.89)$$

The following relation indicates that the unit circle in the z -plane is changed into the vertical imaginary line in the w -plane:

$$\begin{aligned} z = e^{j\theta} \Rightarrow w &= \frac{e^{j\theta}-1}{e^{j\theta}+1} = \frac{e^{j\frac{\theta}{2}}}{e^{j\frac{\theta}{2}}} \left(\frac{e^{j\frac{\theta}{2}}-e^{-j\frac{\theta}{2}}}{e^{j\frac{\theta}{2}}+e^{-j\frac{\theta}{2}}} \right) \\ \Leftrightarrow w &= \frac{2j \sin(\frac{\theta}{2})}{2 \cos(\frac{\theta}{2})} = j \tan\left(\frac{\theta}{2}\right) \end{aligned} \quad (6.90)$$

More generally every point within the unit circle in the z -plane is changed into a point with negative real part in the w -plane.

From a practical point of view the stability of a discrete time system described by its transfer function $H(z) = \frac{N(z)}{D(z)}$ can be assessed:

- either by computing the roots of the characteristic polynomial $D(z)$ and by checking that there are all encompassed in the unit circle,
- or by replacing z by $\frac{1+w}{1-w}$ in the characteristic polynomial $D(z)$ and by computing the Routh array of the polynomial in w which appears at the numerator of the fraction.

An alternative of the Routh–Hurwitz based stability criterion which has been presented is the Jury stability criterion² which will *not* be developed here.

Stability analysis of a second order discrete time system

Consider a second order discrete time system defined by the following difference equation where a_1 , a_0 and b_0 are real parameters:

$$y(k) + a_1y(k-1) + a_0y(k-2) = b_0u(k-2) \quad (6.91)$$

The transfer function $H(z)$ of this system is obtained by taking the \mathcal{Z} transform of the difference equation. Assuming that $y(k)$ and $u(k)$ are causal sequences this yields:

$$\begin{aligned} Y(z) + a_1z^{-1}Y(z) + a_0z^{-2}Y(z) &= b_0z^{-2}U(z) \\ \Rightarrow H(z) = \frac{Y(z)}{U(z)} &= \frac{b_0z^{-2}}{1+a_1z^{-1}+a_0z^{-2}} = \frac{b_0}{z^2+a_1z+a_0} \end{aligned} \quad (6.92)$$

In order to assess the stability of such a second order discrete time system we shall localize the roots of $D(z) = z^2 + a_1z + a_0$. The roots of $D(z)$ are complex if $a_1^2 - 4a_0 < 0$ and real otherwise. Rather than computing explicitly those roots and assess if they are inside the unit circle we will use the homographic transformation $z = \frac{1+w}{1-w}$. We get:

$$D(z)|_{z=\frac{1+w}{1-w}} = \left(\frac{1+w}{1-w}\right)^2 + a_1\left(\frac{1+w}{1-w}\right) + a_0 \quad (6.93)$$

And $D(z) = 0$ yields:

$$D(z) = 0 \Rightarrow (1+w)^2 + a_1(1+w)(1-w) + a_0(1-w)^2 = 0 \quad (6.94)$$

Expanding leads to a polynomial of degree 2:

$$(1 - a_1 + a_0)w^2 + 2(1 - a_0)w + 1 + a_1 + a_0 = 0 \quad (6.95)$$

All the coefficients the polynomial exist and have the same sign; so the necessary condition for the polynomial in w to have all its roots in the left half (complex) plane (LHP) are satisfied. The Routh array reads:

s^2	$1 - a_1 + a_0$	$1 + a_1 + a_0$	(6.96)
s^1	$2(1 - a_0)$		
s^0	$1 + a_1 + a_0$		

All the roots in w have negative real part meaning that all the roots in z are situated in the unit circle, if there is no sign change in the first column. Thus the following inequalities shall be satisfied:

$$\begin{cases} 1 - a_1 + a_0 > 0 \\ 1 - a_0 > 0 \\ 1 + a_1 + a_0 > 0 \end{cases} \Leftrightarrow \begin{cases} a_0 > a_1 - 1 \\ a_0 < 1 \\ a_0 > -a_1 - 1 \end{cases} \quad (6.97)$$

²https://en.wikipedia.org/wiki/Jury_stability_criterion

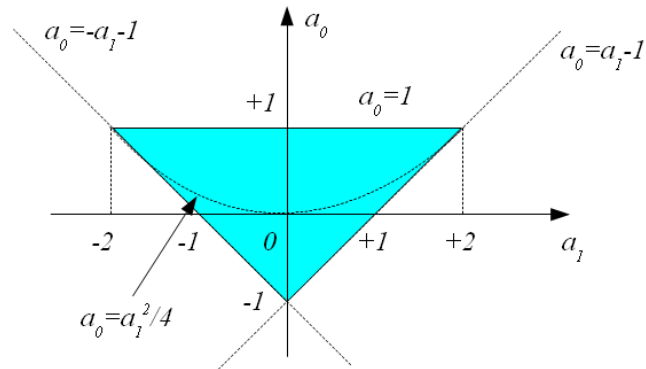


Figure 6.6: Region of coefficient values for second-order discrete time system stability

We have arbitrarily chosen a positive sign; we can also do the test assuming negative sign and check that there is no solution.

Conditions (6.97) are illustrated in Figure 6.6. This shows the resulting stability triangle in the $\{a_0, a_1\}$ plane. We conclude that a second order discrete time system is stable if, and only if, coefficients $\{a_0, a_1\}$ define a point that lies inside this triangle. As previously noted the poles of the characteristic polynomial $D(z)$ are complex if $a_0 > \frac{a_1^2}{4}$ and real otherwise.

f

6.5.3 Frequency response

In this section we will focus on *stable* discrete time linear invariant systems.

The frequency response of a discrete time system is the response of the system to the sampled exponential sequence $u(k) = e^{j\alpha k}$ where α is the reduced frequency. More specifically the frequency response consists in the magnitude and phase relation between the input and the output sequence when the input sequence $u(k)$ is the causal sampled exponential $e^{j\alpha k}$.

We have previously seen that the \mathcal{Z} transform of the output sequence $y(k)$ of a discrete linear time invariant system is obtained thanks to the product between its transfer function $H(z)$ and the \mathcal{Z} transform of the input sequence $u(k)$. Coming back to the discrete time domain, and having in mind that the inverse \mathcal{Z} transform of a product is the discrete time convolution we get:

$$Y(z) = H(z)U(z) \Rightarrow y(k) = h(k)*u(k) \quad (6.98)$$

As a consequence a discrete linear time invariant system is a convolutor. Specializing relation (6.98) to the case where $u(k)$ is the causal sampled exponential reads:

$$u(k) = e^{j\alpha k} \Rightarrow y(k) = \sum_{m=0}^{\infty} h(m)e^{j\alpha(k-m)} \quad (6.99)$$

Using the fact that the summation is a linear operation we get:

$$u(k) = e^{j\alpha k} \sum_{m=0}^{\infty} h(m)e^{-mj\alpha} \quad (6.100)$$

We recognize in the term $\sum_{m=0}^{\infty} h(m)e^{-mj\alpha}$ the \mathcal{Z} transform of the impulse response $h(k)$ where the \mathcal{Z} transform variable z is specialized to be $e^{j\alpha}$:

$$\sum_{m=0}^{\infty} h(m)e^{-j\alpha m} = \sum_{m=0}^{\infty} h(m)z^{-m} \Big|_{z=e^{j\alpha}} = H(z)|_{z=e^{j\alpha}} \quad (6.101)$$

Note that $H(z)|_{z=e^{j\alpha}} = H(e^{j\alpha})$ is called the frequency response of the discrete time system.

As a consequence the frequency response of a discrete linear time invariant system to the sampled exponential reads:

$$e^{j\alpha k} \Rightarrow y(k) = e^{j\alpha k} H(z)|_{z=e^{j\alpha}} = H(e^{j\alpha})e^{j\alpha k} \quad (6.102)$$

The preceding relation shows that the frequency response of a discrete linear time invariant system is closely linked to its transfer function. The frequency of the output sequence is the same than the frequency of the input sequence; only the amplitude and the phase of the input sequence are changed by the linear system. In addition the phase of the output sequence is independent of the amplitude of the input sequence. Denoting by $\|H(e^{j\alpha})\|$ the norm of $H(e^{j\alpha})$ and by $\Phi(\alpha)$ the phase of $H(e^{j\alpha})$ the preceding relation reads:

$$\begin{cases} u(k) = e^{j\alpha k} \\ H(e^{j\alpha}) = \|H(e^{j\alpha})\| e^{j\Phi(\alpha)} \end{cases} \quad (6.103)$$

$$\Rightarrow y(k) = \|H(e^{j\alpha})\| e^{j\alpha k} e^{j\Phi(\alpha)} = \|H(e^{j\alpha})\| e^{j(\alpha k + \Phi(\alpha))}$$

Where Re stands for real part and Im stands for imaginary part:

$$\begin{cases} \|H(e^{j\alpha})\| = \sqrt{(Re(H(e^{j\alpha})))^2 + (Im(H(e^{j\alpha})))^2} \\ \Phi(\alpha) = \angle H(e^{j\alpha}) = \arctan\left(\frac{Im(H(e^{j\alpha}))}{Re(H(e^{j\alpha}))}\right) \end{cases} \quad (6.104)$$

Thus the frequency response of LTI systems is essentially the same for continuous time and discrete time systems. However, an important distinction arises because the frequency response of discrete time LTI systems is always a periodic function of the reduced frequency α with period 2π .

Since $H(e^{j\alpha})$ is periodic with period 2π and since the frequencies α and $\alpha + 2\pi$ are indistinguishable, it follows that $H(e^{j\alpha})$ is only studied over the interval $-\pi < \alpha \leq +\pi$. With respect to this interval, the *low frequencies* are frequencies close to zero, whereas the *high frequencies* are frequencies close to $\pm\pi$. In addition as soon as the polynomials which appear on the numerator and denominator of $H(e^{j\alpha})$ have real coefficients we have:

$$\begin{cases} Re(H(e^{j\alpha})) = Re(H(e^{-j\alpha})) \\ Im(H(e^{j\alpha})) = -Im(H(e^{-j\alpha})) \end{cases} \Rightarrow \begin{cases} \|H(e^{j\alpha})\| = \|H(e^{-j\alpha})\| \\ \angle H(e^{j\alpha}) = -\angle H(e^{-j\alpha}) \end{cases} \quad (6.105)$$

Thus $\|H(e^{j\alpha})\|$ is an even function of α whereas $\angle H(e^{j\alpha})$ is an odd function of α . Consequently $\|H(e^{j\alpha})\|$ and $\angle H(e^{j\alpha})$ are only studied for positive values of α , which corresponds to physically achievable frequencies:

$$0 \leq \alpha < \pi \quad (6.106)$$

It is worth noticing that when input $u(k)$ is not the sampled exponential sequence but a real sinusoidal sequence then the output sequence $y(k)$ is obtained thanks to the real (or imaginary) part of the complex output sequence:

$$\begin{aligned} u(k) &= \cos(\alpha k) = \operatorname{Re}(e^{j\alpha k}) \\ \Rightarrow y(k) &= \operatorname{Re}(\|H(e^{j\alpha})\| e^{j(\alpha k + \Phi(\alpha))}) = \|H(e^{j\alpha})\| \cos(\alpha k + \Phi(\alpha)) \end{aligned} \quad (6.107)$$

And similarly:

$$\begin{aligned} u(k) &= \sin(\alpha k) = \operatorname{Im}(e^{j\alpha k}) \\ \Rightarrow y(k) &= \operatorname{Im}(\|H(e^{j\alpha})\| e^{j(\alpha k + \Phi(\alpha))}) = \|H(e^{j\alpha})\| \sin(\alpha k + \Phi(\alpha)) \end{aligned} \quad (6.108)$$

The static gain G_s of a discrete time system is the limit as $k \rightarrow \infty$ of the ratio between the output sequence $y(k)$ and the input sequence $u(k)$. Assuming that the limit exists and applying the final value theorem leads to the following expression of the static gain:

$$G_s = \lim_{k \rightarrow \infty} \frac{y(k)}{u(k)} = \lim_{z \rightarrow 1} \frac{(1 - z^{-1})Y(z)}{(1 - z^{-1})U(z)} = \lim_{z \rightarrow 1} \frac{Y(z)}{U(z)} = \lim_{z \rightarrow 1} H(z) \quad (6.109)$$

As an example consider again a first order discrete time system defined by the following difference equation where a and b are real parameters:

$$y(k) + ay(k - 1) = bu(k - 1) \quad (6.110)$$

The transfer function $H(z)$ of this system is obtained by taking the \mathcal{Z} transform of the difference equation. Assuming that $y(k)$ and $u(k)$ are causal sequences this yields:

$$\begin{aligned} Y(z) + az^{-1}Y(z) &= bz^{-1}U(z) \\ \Rightarrow H(z) &= \frac{Y(z)}{U(z)} = \frac{bz^{-1}}{1+az^{-1}} = \frac{b}{z+a} \end{aligned} \quad (6.111)$$

In order to assess the frequency response of the first order discrete time system we replace z by $e^{j\alpha}$ in the transfer function $H(z)$:

$$H(e^{j\alpha}) = \frac{b}{e^{j\alpha} + a} = \frac{b}{\cos(\alpha) + a + j \sin(\alpha)} \quad (6.112)$$

That is:

$$H(e^{j\alpha}) = \|H(e^{j\alpha})\| e^{j\Phi(\alpha)} \quad (6.113)$$

Where

$$\begin{cases} \|H(e^{j\alpha})\| = \frac{|b|}{\sqrt{1+2a \cos(\alpha)+a^2}} \\ \Phi(\alpha) = -\arctan\left(\frac{\sin(\alpha)}{\cos(\alpha)+a}\right) \quad (+\pi \text{ if } b < 0) \end{cases} \quad (6.114)$$

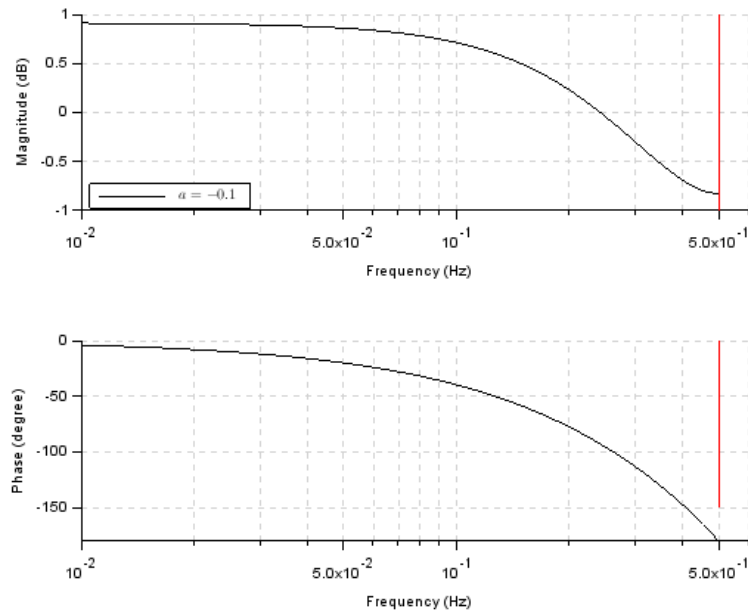


Figure 6.7: Frequency response of a low pass first order filter: $a = -0.1$ and $b = 1$

It can then be shown that a first order discrete time system is a low pass filter as soon as $-1 < a < 0$ and a high pass filter as soon as $0 < a < 1$. Figure 6.7 presents the frequency response of a low pass first order filter where $a = -0.1$ and $b = 1$, whereas Figure 6.8 presents the frequency response of a high pass first order filter where $a = +0.1$ and $b = 1$. It is worth noticing that the graduation on the abscissa is the product fT (which shall be lower than 0.5 so that $\alpha \leq \pi$) where f is the ordinary frequency of the input sequence and T the sampling period whereas the ordinate is graduated in dB . We can check that:

$$a = -0.1 \Rightarrow \begin{cases} 20 \log_{10} \|H(e^{j\alpha})\|_{\alpha=0} = +0.91 \text{ dB} \\ 20 \log_{10} \|H(e^{j\alpha})\|_{\alpha=\pi} = -0.83 \text{ dB} \end{cases} \quad (6.115)$$

And:

$$a = +0.1 \Rightarrow \begin{cases} 20 \log_{10} \|H(e^{j\alpha})\|_{\alpha=0} = -0.83 \text{ dB} \\ 20 \log_{10} \|H(e^{j\alpha})\|_{\alpha=\pi} = 0.91 \text{ dB} \end{cases} \quad (6.116)$$

6.6 Continuous time plant with Zero-Order Hold (ZOH) and sampler

The purpose of this section is to derive the transfer function of a continuous time plant with sampled output and input. More specifically consider the general configuration shown in Figure 6.9 where it is desired to find the equivalent discrete time transfer function $F(z)$ of the continuous time plant $F(s)$ with Digital-to-Analog Converter (DAC) and Analog-to-Digital

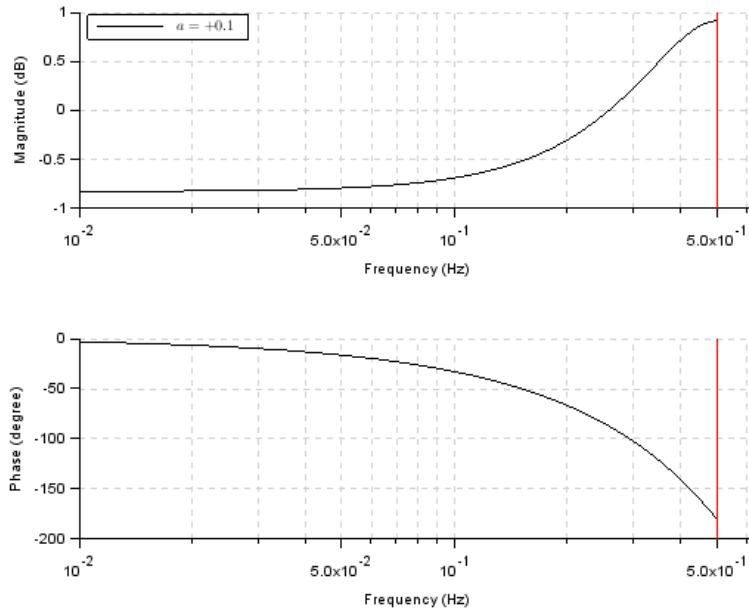


Figure 6.8: Frequency response of a high pass first order filter: $a = +0.1$ and $b = 1$

Converter (ADC). The equivalent discrete time transfer function $F(z)$ will be obtained by computing the impulse response of the block composed by the DAC, the plant $F(s)$ and the ADC.

The Digital-to-Analog Converter (DAC) converts the digital control $u(k)$ to the analog control signal $u(t)$. It is modelled by a Zero-Order Hold device with transfer function $B_0(s)$ as illustrated in Figure 6.10.

The Zero-Order Hold (ZOH) device generates the continuous time input signal $u(t)$ by holding each sample value $u(k)$ constant over one sample period T . Thus the impulse response of the Zero-Order Hold (ZOH) device is a rectangular pulse of duration T as illustrated in Figure 6.10. Therefore denoting by $\Gamma(t)$ the unit step function the transfer function of the ZOH device is given by:

$$u(k) = \delta(k) \Rightarrow g_0(t) = \Gamma(t) - \Gamma(t - T) \Rightarrow B_0(s) = \frac{1 - e^{-sT}}{s} \quad (6.117)$$

Let $u_0(t)$ be the response of the plant with transfer function $F(s)$ to the unit step function $\Gamma(t)$. Assuming that $F(s)$ is a linear and invariant plant, its response to a rectangular pulse of duration T reads:

$$u(t) = g_0(t) = \Gamma(t) - \Gamma(t - T) \Rightarrow y(t) = u_0(t) - u_0(t - T) \quad (6.118)$$

Finally the Analog-to-Digital Converter (ADC) is modelled by a sampler as illustrated in Figure 6.11.

Thus the impulse response of the block composed by the DAC, the plant $F(s)$ and the ADC is signal defined by (6.118) sampled at the sampling period

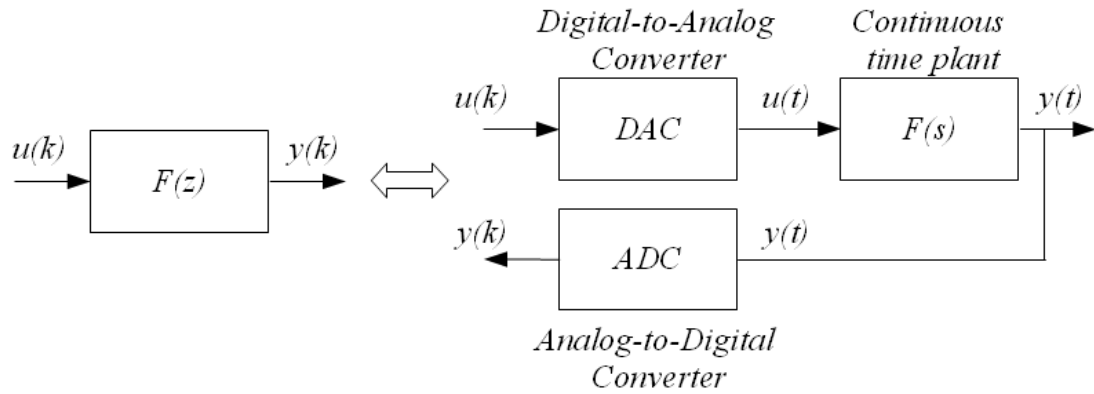


Figure 6.9: Continuous time plant $F(s)$ with Digital-to-Analog and Analog-to-Digital Converters

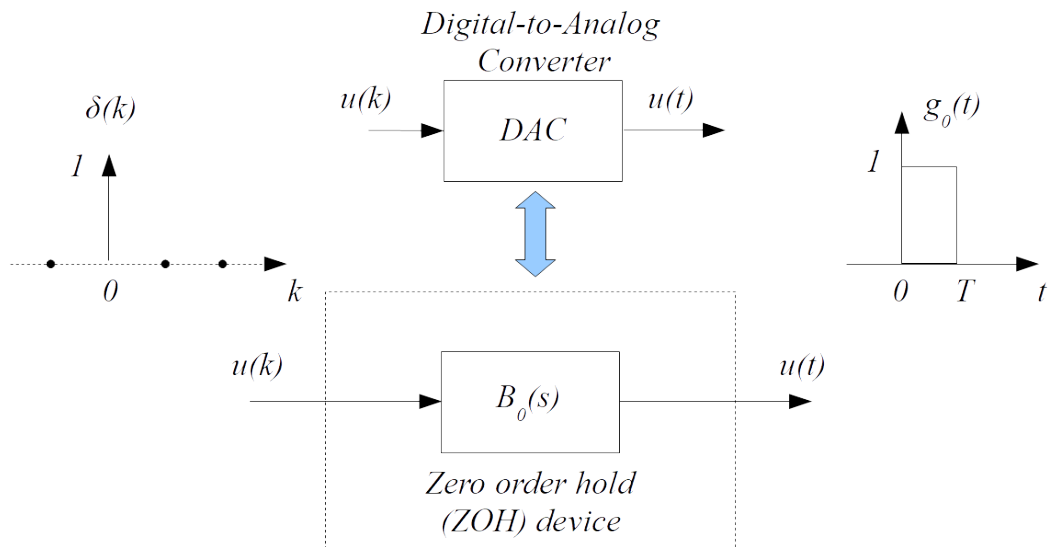


Figure 6.10: Zero-Order Hold (ZOH) device

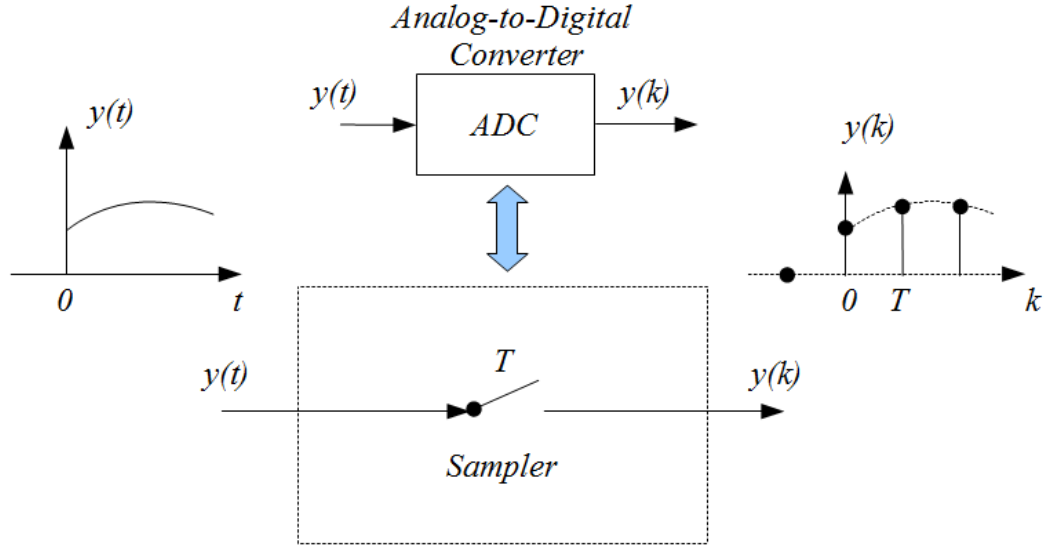


Figure 6.11: Model of analog-to-Digital Converter (ADC)

T :

$$t = kT \Rightarrow y(k) = u_0(kT) - u_0(kT - T) = u_0(kT) - u_0((k-1)T) \quad (6.119)$$

Having in mind that z^{-1} is the one sampling period delay operator, the \mathcal{Z} transform of the preceding equation yields:

$$\begin{aligned} F(z) &= \mathcal{Z}[u_0(kT) - u_0((k-1)T)] \\ &= \mathcal{Z}[u_0(kT)] - z^{-1} \mathcal{Z}[u_0(kT)] \\ &= (1 - z^{-1}) \mathcal{Z}[u_0(kT)] \end{aligned} \quad (6.120)$$

Here $u_0(kT)$ represents the sequence obtained by sampling continuous time signal $u_0(t)$ which is the unit step response of the continuous time system. As far as the Laplace transform of $u_0(t)$ is $\frac{F(s)}{s}$ the preceding relation is usually written as follows:

$$F(z) = \mathcal{Z}[B_0(s)F(s)] = (1 - z^{-1}) \mathcal{Z}\left[\frac{F(s)}{s}\right] \quad (6.121)$$

In the preceding relation the term $\mathcal{Z}\left[\frac{F(s)}{s}\right]$ shall be interpreted as the \mathcal{Z} transform of the sampled unit step response of the continuous time plant. Thus the computation of $\mathcal{Z}\left[\frac{F(s)}{s}\right]$ involves three steps: first to compute the inverse Laplace transform of $\frac{F(s)}{s}$, then to sample the obtained continuous time signal to obtain a sequence and finally to take its \mathcal{Z} transform. This is quite a long process but hopefully the residue theorem applied on this specific inversion problem leads to a straightforward result:

$$\mathcal{Z}\left[\frac{F(s)}{s}\right] = \sum_i \text{Res}_{z=\lambda_i} \left[\frac{1}{1 - z^{-1}e^{sT}} \cdot \frac{F(s)}{s} \right] \quad (6.122)$$

The residue $Res_{z=\lambda_i} \left[\frac{1}{1-z^{-1}e^{sT}} \cdot \frac{F(s)}{s} \right]$ shall be computed around each pole λ_i of $\frac{F(s)}{s}$. Assuming that λ_i is a pole of multiplicity n_i then the residue of $\frac{1}{1-z^{-1}e^{sT}} \cdot \frac{F(s)}{s}$ around pole λ_i is given by:

$$Res_{s=\lambda_i} \left[\frac{1}{1-z^{-1}e^{sT}} \cdot \frac{F(s)}{s} \right] = \frac{1}{(n_i-1)!} \frac{d^{n_i-1}}{ds^{n_i-1}} \left((s-\lambda_i)^{n_i} \frac{1}{1-z^{-1}e^{sT}} \cdot \frac{F(s)}{s} \right) \Big|_{s=\lambda_i} \quad (6.123)$$

Example 6.12. Consider the following continuous time plant whose transfer function is:

$$F(s) = \frac{1}{1+\tau s} \quad (6.124)$$

This plant is inserted in a discrete time environment as illustrated in Figure 6.9. We wish to find the equivalent discrete time transfer function $F(z)$ of the continuous time plant $F(s)$ with Digital-to-Analog Converter (DAC) and Analog-to-Digital Converter (ADC).

Applying (6.121) leads to the following expression of $F(s)$:

$$F(z) = (1-z^{-1}) \mathcal{Z} \left[\frac{1}{s(1+\tau s)} \right] \quad (6.125)$$

Where:

$$\mathcal{Z} \left[\frac{1}{s(1+\tau s)} \right] = \sum_i Res_{z=\lambda_i} \left[\frac{1}{1-z^{-1}e^{sT}} \cdot \frac{1}{s(1+\tau s)} \right] \quad (6.126)$$

Transfer function $\frac{1}{s(1+\tau s)}$ has two poles:

- $\lambda_1 = 0$, with multiplicity $n_1 = 1$; applying (6.123) the residue around that pole reads:

$$\begin{aligned} Res_{s=0} \left[\frac{1}{1-z^{-1}e^{sT}} \cdot \frac{1}{s(1+\tau s)} \right] &= \frac{1}{(1-1)!} \frac{d^{1-1}}{ds^{1-1}} \left(s \frac{1}{1-z^{-1}e^{sT}} \cdot \frac{F(s)}{s} \right) \Big|_{s=0} \\ &= \frac{1}{1-z^{-1}e^{sT}} \cdot \frac{1}{1+\tau s} \Big|_{s=0} \\ &= \frac{1}{1-z^{-1}} \end{aligned} \quad (6.127)$$

- $\lambda_2 = -\frac{1}{\tau}$, with multiplicity $n_2 = 1$; applying (6.123) the residue around that pole reads:

$$\begin{aligned} Res_{s=-\frac{1}{\tau}} \left[\frac{1}{1-z^{-1}e^{sT}} \cdot \frac{1}{s(1+\tau s)} \right] &= \frac{1}{(1-1)!} \frac{d^{1-1}}{ds^{1-1}} \left((s+\frac{1}{\tau}) \frac{1}{1-z^{-1}e^{sT}} \cdot \frac{F(s)}{s} \right) \Big|_{s=-\frac{1}{\tau}} \\ &= \frac{1}{1-z^{-1}e^{sT}} \cdot \frac{1}{\tau s} \Big|_{s=-\frac{1}{\tau}} \\ &= \frac{-1}{1-z^{-1}e^{-\frac{T}{\tau}}} \end{aligned} \quad (6.128)$$

Consequently $F(z)$ reads:

$$\begin{aligned}
 F(z) &= (1 - z^{-1}) \left(\frac{1}{1-z^{-1}} - \frac{1}{1-z^{-1}e^{-\frac{T}{\tau}}} \right) \\
 &= 1 - \frac{1-z^{-1}}{1-z^{-1}e^{-\frac{T}{\tau}}} = \frac{1-z^{-1}e^{-\frac{T}{\tau}} - 1 + z^{-1}}{1-z^{-1}e^{-\frac{T}{\tau}}} \\
 &= \frac{z^{-1} - z^{-1}e^{-\frac{T}{\tau}}}{1-z^{-1}e^{-\frac{T}{\tau}}}
 \end{aligned} \tag{6.129}$$

That is:

$$F(z) = \frac{1 - e^{-\frac{T}{\tau}}}{z - e^{-\frac{T}{\tau}}} \tag{6.130}$$

■

6.7 State-space realization

6.7.1 Discretization

We consider the following state-space realization of the plant:

$$\begin{cases} \dot{\underline{x}}(t) = \mathbf{A}\underline{x}(t) + \mathbf{B}\underline{u}(t) \\ \underline{y}(t) = \mathbf{C}\underline{x}(t) + \mathbf{D}\underline{u}(t) \end{cases} \tag{6.131}$$

The corresponding transfer function is obtained by taking the Laplace transform of the preceding relations and assuming $\underline{x}(0) = \mathbf{0}$:

$$\mathbf{F}(s) = \mathbf{C}(s\mathbb{I} - \mathbf{A})^{-1}\mathbf{B} + \mathbf{D} \tag{6.132}$$

The solution of the linear differential equation $\dot{\underline{x}}(t) = \mathbf{A}\underline{x}(t) + \mathbf{B}\underline{u}(t)$ where $\underline{x}(0) = \underline{x}_0$ reads as follows:

$$\underline{x}(t) = e^{\mathbf{A}t}\underline{x}_0 + \int_0^t e^{\mathbf{A}(t-\tau)}\mathbf{B}\underline{u}(\tau)d\tau \tag{6.133}$$

Assuming that the state vector $\underline{x}(t)$ is sampled with a Zero-Order Hold (ZOH) device with a sampling period T , we get at $t = kT$:

$$\underline{x}(kT) = e^{\mathbf{A}kT}\underline{x}_0 + \int_0^{kT} e^{\mathbf{A}(kT-\tau)}\mathbf{B}\underline{u}(\tau)d\tau \tag{6.134}$$

We seek the expression of $\underline{x}((k+1)T)$ as a function of $\underline{x}(kT)$. To do that, let's express $\underline{x}(t)$ at $t = (k+1)T$:

$$\begin{aligned}
 \underline{x}((k+1)T) &= e^{\mathbf{A}(k+1)T}\underline{x}_0 \\
 &\quad + \int_0^{(k+1)T} e^{\mathbf{A}((k+1)T-\tau)}\mathbf{B}\underline{u}(\tau)d\tau \\
 &= e^{\mathbf{A}(k+1)T}\underline{x}_0 \\
 &\quad + \int_0^{kT} e^{\mathbf{A}((k+1)T-\tau)}\mathbf{B}\underline{u}(\tau)d\tau \\
 &\quad + \int_{kT}^{(k+1)T} e^{\mathbf{A}((k+1)T-\tau)}\mathbf{B}\underline{u}(\tau)d\tau \\
 &= e^{\mathbf{A}T} \left(e^{\mathbf{A}kT}\underline{x}_0 + \int_0^{kT} e^{\mathbf{A}(kT-\tau)}\mathbf{B}\underline{u}(\tau)d\tau \right) \\
 &\quad + \int_{kT}^{(k+1)T} e^{\mathbf{A}((k+1)T-\tau)}\mathbf{B}\underline{u}(\tau)d\tau
 \end{aligned} \tag{6.135}$$

We recognize in the term into parentheses the expression of $\underline{x}(kT)$. Thus we can write:

$$\underline{x}((k+1)T) = e^{\mathbf{A}T} \underline{x}(kT) + \int_{kT}^{(k+1)T} e^{\mathbf{A}((k+1)T-\tau)} \mathbf{B} \underline{u}(\tau) d\tau \quad (6.136)$$

Assuming that the term $\underline{u}(\tau)$ which appears in the integral is constant during each time step (this is the case when a Zero-Order Hold (ZOH) device is used), we can replace it by $\underline{u}(kT)$ and put it out of the integral, as well as matrix \mathbf{B} . We get:

$$\begin{aligned} \underline{x}((k+1)T) &= e^{\mathbf{A}T} \underline{x}(kT) + \int_{kT}^{(k+1)T} e^{\mathbf{A}((k+1)T-\tau)} \mathbf{B} \underline{u}(kT) d\tau \\ &= e^{\mathbf{A}T} \underline{x}(kT) + \left(\int_{kT}^{(k+1)T} e^{\mathbf{A}((k+1)T-\tau)} d\tau \right) \mathbf{B} \underline{u}(kT) \\ &= e^{\mathbf{A}T} \underline{x}(kT) + \left(\int_0^T e^{\mathbf{A}(T-\tau)} d\tau \right) \mathbf{B} \underline{u}(kT) \\ &= e^{\mathbf{A}T} \underline{x}(kT) + \left(\int_0^T e^{\mathbf{A}t} dt \right) \mathbf{B} \underline{u}(kT) \end{aligned} \quad (6.137)$$

We finally get:

$$\underline{x}((k+1)T) = \mathbf{A}_d \underline{x}(kT) + \mathbf{B}_d \underline{u}(kT) \quad (6.138)$$

where:

$$\boxed{\begin{cases} \mathbf{A}_d = e^{\mathbf{A}T} \\ \mathbf{B}_d = \left(\int_0^T e^{\mathbf{A}t} dt \right) \mathbf{B} \end{cases}} \quad (6.139)$$

A trick to get \mathbf{A}_d and \mathbf{B}_d in one shot is to use the following property³:

$$\exp \left(\begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} T \right) = \begin{bmatrix} \mathbf{A}_d & \mathbf{B}_d \\ \mathbf{0} & \mathbb{I} \end{bmatrix} \quad (6.140)$$

Shortly, denoting $\underline{x}_{k+1} := \underline{x}((k+1)T)$ and $\underline{u}_k := \underline{u}(kT)$, we can write:

$$\begin{cases} \underline{x}_{k+1} := \underline{x}((k+1)T) \\ \underline{u}_k := \underline{u}(kT) \end{cases} \Rightarrow \underline{x}_{k+1} = \mathbf{A}_d \underline{x}_k + \mathbf{B}_d \underline{u}_k \quad (6.141)$$

Assuming that \mathbf{A}^{-1} exists and using the fact that $e^{\mathbf{A}T} \mathbf{A}^{-1} = \mathbf{A}^{-1} e^{\mathbf{A}T}$, we can compute the integral term which is needed to get \mathbf{B}_d :

$$\int_0^T e^{\mathbf{A}t} dt = \mathbf{A}^{-1} [e^{\mathbf{A}t}]_{t=0}^T = \mathbf{A}^{-1} (e^{\mathbf{A}T} - \mathbb{I}) \quad (6.142)$$

Thus, assuming that \mathbf{A}^{-1} exists, we get:

$$\underline{x}_{k+1} = e^{\mathbf{A}T} \underline{x}_k + \mathbf{A}^{-1} (e^{\mathbf{A}T} - \mathbb{I}) \mathbf{B} \underline{u}_k \quad (6.143)$$

As far as the output equation is concerned, it remains unchanged:

$$\underline{y}(t) = \mathbf{C} \underline{x}(t) + \mathbf{D} \underline{u}(t) \Rightarrow \underline{y}_k = \mathbf{C} \underline{x}_k + \mathbf{D} \underline{u}_k \quad (6.144)$$

Consequently, the state space representation of a discrete time system is the following where \mathbf{A}_d and \mathbf{B}_d are provided by (6.139):

$$\begin{cases} \underline{x}_{k+1} = \mathbf{A}_d \underline{x}_k + \mathbf{B}_d \underline{u}_k \\ \underline{y}_k = \mathbf{C} \underline{x}_k + \mathbf{D} \underline{u}_k \end{cases} \quad (6.145)$$

³<https://en.wikipedia.org/wiki/Discretization>

6.7.2 Linear system perturbed by white Gaussian noises

When the continuous time system is perturbed by white Gaussian noises, the state space representation reads:

$$\begin{cases} \dot{\underline{x}}(t) = \mathbf{A}\underline{x}(t) + \mathbf{B}\underline{u}(t) + \underline{w}(t) \\ \underline{y}(t) = \mathbf{C}\underline{x}(t) + \mathbf{D}\underline{u}(t) + \underline{v}(t) \end{cases} \quad \text{where} \quad \begin{cases} \underline{w}(t) \sim \mathcal{N}(\underline{0}, \mathbf{Q}) \\ \underline{v}(t) \sim \mathcal{N}(\underline{0}, \mathbf{R}) \end{cases} \quad (6.146)$$

Then the corresponding discrete time system is the following³:

$$\begin{cases} \underline{x}_{k+1} = \mathbf{A}_d \underline{x}_k + \mathbf{B}_d \underline{u}_k + \underline{w}_k \\ \underline{y}_k = \mathbf{C} \underline{x}_k + \mathbf{D} \underline{u}_k + \underline{v}_k \end{cases} \quad \text{where} \quad \begin{cases} \underline{w}_k \sim \mathcal{N}(\underline{0}, \mathbf{Q}_d = \int_0^T e^{\mathbf{A}\tau} \mathbf{Q} e^{\mathbf{A}^T \tau} d\tau) \\ \underline{v}_k \sim \mathcal{N}(\underline{0}, \mathbf{R}_d = \frac{\mathbf{R}}{T}) \end{cases} \quad (6.147)$$

A trick to get \mathbf{Q}_d is to use the following property³:

$$\exp\left(\begin{bmatrix} -\mathbf{A} & \mathbf{Q} \\ \mathbf{0} & \mathbf{A}^T \end{bmatrix} T\right) = \begin{bmatrix} * & \mathbf{A}_d^{-1} \mathbf{Q}_d \\ \mathbf{0} & \mathbf{A}_d^T \end{bmatrix} \Rightarrow \mathbf{Q}_d = (\mathbf{A}_d^T)^T (\mathbf{A}_d^{-1} \mathbf{Q}_d) \quad (6.148)$$

6.7.3 State space realization of $(1 - z^{-1}) \mathcal{Z} \left[\frac{F(s)}{s} \right]$

The purpose of this section is to show that (6.145) corresponds to the state space realization of (6.121). Indeed, from relation (6.121) we have:

$$F(z) = (1 - z^{-1}) \mathcal{Z} \left[\frac{F(s)}{s} \right] \quad (6.149)$$

The expression $\mathcal{Z} \left[\frac{F(s)}{s} \right]$ represents the \mathcal{Z} transform of the unit step response of a plant with transfer function $F(s)$. Let $y_s(t)$ be the unit step response of the plant. The unit step response $y_s(t)$ can be computed as the inverse Laplace transform of $\frac{F(s)}{s}$:

$$y_s(t) = \mathcal{L}^{-1} \left[\frac{F(s)}{s} \right] \quad (6.150)$$

Consider a SISO plant with the following state-space realization:

$$\begin{cases} \dot{\underline{x}}(t) = \mathbf{A}\underline{x}(t) + \mathbf{B}u(t) \\ y(t) = \mathbf{C}\underline{x}(t) \end{cases} \quad (6.151)$$

Transfer function $F(s)$ is obtained assuming no initial condition ($\underline{x}(0) = \underline{0}$). By taking the Laplace transform of the state-space realization we get:

$$F(s) = \frac{\mathcal{L}[y(t)]}{\mathcal{L}[u(t)]} = \frac{Y(s)}{U(s)} = \mathbf{C} (s\mathbb{I} - \mathbf{A})^{-1} \mathbf{B} \quad (6.152)$$

We have seen in (6.133) that, assuming to initial condition, the trajectory of the state vector of a linear system reads as follows:

$$\underline{x}(0) = \underline{0} \Rightarrow \underline{x}(t) = \int_0^t e^{\mathbf{A}(t-\tau)} \mathbf{B}u(\tau) d\tau \quad (6.153)$$

When $u(t) = 1 \forall t \geq 0$, the expression of the state vector $\underline{x}_s(t)$ corresponding an unit step input reads:

$$u(t) = 1 \forall t \geq 0 \Rightarrow \underline{x}_s(t) = \int_0^t e^{\mathbf{A}(t-\tau)} \mathbf{B} d\tau \quad (6.154)$$

Assuming that \mathbf{A}^{-1} exists, and because \mathbf{A}^{-1} and $e^{\mathbf{A}t}$ commute, the unit step response $\underline{x}_s(t)$ can be computed as follows:

$$\begin{aligned} \underline{x}_s(t) &= \int_0^t e^{\mathbf{A}(t-\tau)} \mathbf{B} d\tau = e^{\mathbf{A}t} \int_0^t e^{-\mathbf{A}\tau} \mathbf{B} d\tau \\ &= -\mathbf{A}^{-1} e^{\mathbf{A}t} \left(e^{-\mathbf{A}\tau} \Big|_{\tau=0}^{\tau=t} \right) \mathbf{B} \\ &= \mathbf{A}^{-1} (e^{\mathbf{A}t} - \mathbb{I}) \mathbf{B} \\ &= (e^{\mathbf{A}t} - \mathbb{I}) \mathbf{A}^{-1} \mathbf{B} \end{aligned} \quad (6.155)$$

Thus the unit step response $y_s(t)$ of the plant reads:

$$y_s(t) = \mathbf{C} \underline{x}_s(t) = \mathbf{C} (e^{\mathbf{A}t} - \mathbb{I}) \mathbf{A}^{-1} \mathbf{B} \quad (6.156)$$

Now we have to compute the \mathcal{Z} transform of the sampled unit step response $y_s(kT)$:

$$y_s(kT) = \mathbf{C} (e^{\mathbf{A}kT} - \mathbb{I}) \mathbf{A}^{-1} \mathbf{B} \quad (6.157)$$

The \mathcal{Z} transform of $e^{\mathbf{A}kT}$ and \mathbb{I} are the following:

$$\begin{cases} \mathcal{Z} [e^{\mathbf{A}kT}] = (\mathbb{I} - z^{-1} e^{\mathbf{A}T})^{-1} \\ \mathcal{Z} [\mathbb{I}] = \mathcal{Z} [e^{\mathbf{A}kT}]_{\mathbf{A}=\mathbf{0}} = (1 - z^{-1})^{-1} \mathbb{I} \end{cases} \quad (6.158)$$

From the facts that the \mathcal{Z} transform is linear we get:

$$\begin{aligned} \mathcal{Z} [y_s(kT)] &= \mathbf{C} (\mathcal{Z} [e^{\mathbf{A}kT}] - \mathcal{Z} [\mathbb{I}]) \mathbf{A}^{-1} \mathbf{B} \\ &= \mathbf{C} \left((\mathbb{I} - z^{-1} e^{\mathbf{A}T})^{-1} - (1 - z^{-1})^{-1} \mathbb{I} \right) \mathbf{A}^{-1} \mathbf{B} \end{aligned} \quad (6.159)$$

Furthermore we can check that for any invertible matrices \mathbf{E} and \mathbf{F} we have:

$$\mathbf{E}^{-1} - \mathbf{F}^{-1} = \mathbf{E}^{-1} (\mathbf{E} - \mathbf{F}) \mathbf{F}^{-1} \quad (6.160)$$

Setting $\mathbf{E} = \mathbb{I} - z^{-1} e^{\mathbf{A}T}$ and $\mathbf{F} = (1 - z^{-1}) \mathbb{I}$ we get:

$$\begin{aligned} \mathbf{E}^{-1} - \mathbf{F}^{-1} &= (\mathbb{I} - z^{-1} e^{\mathbf{A}T})^{-1} (\mathbb{I} - z^{-1} e^{\mathbf{A}T} - (1 - z^{-1}) \mathbb{I}) (1 - z^{-1})^{-1} \mathbb{I} \\ &= \frac{1}{1 - z^{-1}} (\mathbb{I} - z^{-1} e^{\mathbf{A}T})^{-1} (z^{-1} \mathbb{I} - z^{-1} e^{\mathbf{A}T}) \\ &= \frac{1}{1 - z^{-1}} (\mathbb{I} - z^{-1} e^{\mathbf{A}T})^{-1} z^{-1} (\mathbb{I} - e^{\mathbf{A}T}) \\ &= \frac{1}{1 - z^{-1}} (z \mathbb{I} - e^{\mathbf{A}T})^{-1} (\mathbb{I} - e^{\mathbf{A}T}) \end{aligned} \quad (6.161)$$

Thus:

$$\begin{aligned} \mathcal{Z} \left[\frac{F(s)}{s} \right] := \mathcal{Z} [y_s(kT)] &= \mathbf{C} (\mathbf{E}^{-1} - \mathbf{F}^{-1}) \mathbf{A}^{-1} \mathbf{B} \\ &= \frac{1}{1 - z^{-1}} \mathbf{C} (z \mathbb{I} - e^{\mathbf{A}T})^{-1} (\mathbb{I} - e^{\mathbf{A}T}) \mathbf{A}^{-1} \mathbf{B} \\ &= \frac{1}{1 - z^{-1}} \mathbf{C} (z \mathbb{I} - e^{\mathbf{A}T})^{-1} \mathbf{A}^{-1} (\mathbb{I} - e^{\mathbf{A}T}) \mathbf{B} \end{aligned} \quad (6.162)$$

We finally get:

$$(1 - z^{-1}) \mathcal{Z} \left[\frac{F(s)}{s} \right] = \mathbf{C} (z\mathbb{I} - e^{\mathbf{A}T})^{-1} \mathbf{A}^{-1} (\mathbb{I} - e^{\mathbf{A}T}) \mathbf{B} \quad (6.163)$$

The preceding equation indicates that the state space realization of $(1 - z^{-1}) \mathcal{Z} \left[\frac{F(s)}{s} \right]$ is the following:

$$(1 - z^{-1}) \mathcal{Z} \left[\frac{F(s)}{s} \right] = \left(\begin{array}{c|c} \mathbf{A}_d & \mathbf{B}_d \\ \hline \mathbf{C} & \mathbf{0} \end{array} \right) := \mathbf{C} (z\mathbb{I} - \mathbf{A}_d)^{-1} \mathbf{B}_d \quad (6.164)$$

We retrieve for \mathbf{A}_d and \mathbf{B}_d the expression given in (6.139):

$$\begin{cases} \mathbf{A}_d = e^{\mathbf{A}T} \\ \mathbf{B}_d = \mathbf{A}^{-1} (\mathbb{I} - e^{\mathbf{A}T}) \mathbf{B} = \left(\int_0^T e^{\mathbf{A}t} dt \right) \mathbf{B} \end{cases} \quad (6.165)$$

6.7.4 First order approximation

When the sampling period T is small enough, the following approximations hold:

$$e^{\mathbf{A}T} \approx \mathbb{I} + \mathbf{A}T \Rightarrow \begin{cases} \mathbf{A}_d \approx \mathbb{I} + \mathbf{A}T \\ \mathbf{B}_d \approx \mathbf{B}T \end{cases} \quad (6.166)$$

Those approximations can also be obtained by discretizing the derivative of $\underline{\dot{x}}(t)$ using the Euler's method:

$$\begin{cases} \underline{\dot{x}}(kT) = \mathbf{A}\underline{x}(kT) + \mathbf{B}\underline{u}(kT) \\ \underline{\dot{x}}(kT) \approx \frac{\underline{x}((k+1)T) - \underline{x}(kT)}{T} \\ \Rightarrow \underline{x}((k+1)T) \approx (\mathbb{I} + \mathbf{A}T) \underline{x}(kT) + \mathbf{B}T \underline{u}(kT) \end{cases} \quad (6.167)$$

Another approximation of $e^{\mathbf{A}T}$ is the following:

$$e^{\mathbf{A}T} = e^{\mathbf{A}T/2} \left(e^{\mathbf{A}^{-T/2}} \right)^{-1} \approx \left(\mathbb{I} + \frac{1}{2} \mathbf{A}T \right) \left(\mathbb{I} - \frac{1}{2} \mathbf{A}T \right)^{-1} \quad (6.168)$$

6.7.5 Miscellaneous

Furthermore, it is worth noticing that if $z_k = a_k + jb_k$ is an eigenvalue of the state matrix \mathbf{A}_d of the discrete time system, the corresponding eigenvalue s_k of the state matrix \mathbf{A} of the continuous time system can be obtained as follows:

$$s_k = \frac{1}{T} \ln \left(\sqrt{a_k^2 + b_k^2} \right) + j \frac{1}{T} \operatorname{atan} \left(\frac{b_k}{a_k} \right) \Leftrightarrow z_k = a_k + jb_k = e^{s_k T} \quad (6.169)$$

6.8 Discretization using Adomian series

6.8.1 Differential equations

First we recall the main principles of Adomian's method⁴. We consider the following differential equation where \underline{x} is an unknown (vector) function, L a

⁴Y. Cherruault, G. Adomian, Decomposition methods: a new proof of convergence, Math. Comp. Model. 18 (1993) 103–106

linear differential operator, $N(\underline{x}(t))$ a nonlinear operator acting upon \underline{x} and $\underline{h}(t)$ a known (vector) function independent of \underline{x} :

$$L(\underline{x}) = N(\underline{x}) + \underline{h} \quad (6.170)$$

After integration, and using the linearity of L , we get:

$$\underline{x}(t) = \underline{x}(0) + L^{-1}(N(\underline{x}) + \underline{h}) = (\underline{x}(0) + L^{-1}(\underline{h})) + L^{-1}(N(\underline{x})) \quad (6.171)$$

Then $\underline{x}(t)$ and the nonlinear operator $N(\underline{x})$, which is assumed to be analytic (meaning infinitely differentiable), are represented by the following series:

$$\begin{cases} \underline{x}(t) = \sum_{i=0}^{\infty} \underline{x}_i(t) \\ N(\underline{x}) = \sum_{i=0}^{\infty} A_i(\underline{x}_0, \dots, \underline{x}_i) \end{cases} \quad (6.172)$$

The so-called Adomian polynomials $A_m(\underline{x}_0, \dots, \underline{x}_m)$ are computed as follows:

$$A_m(\underline{x}_0, \dots, \underline{x}_m) = \frac{1}{m!} \left[\frac{d^m}{d\lambda^m} N \left(\sum_{k=0}^m \lambda^k \underline{x}_k \right) \right]_{\lambda=0} \quad m = 0, 1, 2, \dots \quad (6.173)$$

Adomian polynomials $A_m(\underline{x}_0, \dots, \underline{x}_m)$ can equivalently be computed as follows⁵:

$$A_m(\underline{x}_0, \dots, \underline{x}_m) = \frac{1}{2\pi} \int_{-\pi}^{+\pi} N \left(\sum_{k=0}^m \underline{x}_k e^{jk\lambda} \right) e^{-jm\lambda} d\lambda \quad m = 0, 1, 2, \dots \quad (6.174)$$

Using series expansion (6.172) within (6.171) yields:

$$\sum_{i=0}^{\infty} \underline{x}_i(t) = (\underline{x}(0) + L^{-1}(\underline{h})) + L^{-1} \left(\sum_{i=0}^{\infty} A_i(\underline{x}_0, \dots, \underline{x}_i) \right) \quad (6.175)$$

Equating terms with the same order on both sides leads to the approximation of order n of $\underline{x}(t)$:

$$\underline{x}(t) \approx \sum_{i=0}^n \underline{x}_i(t) \text{ where } \begin{cases} \underline{x}_0(t) = \underline{x}(0) + L^{-1}(\underline{h}) \\ \underline{x}_1(t) = L^{-1}(A_0(\underline{x}_0)) \\ \underline{x}_2(t) = L^{-1}(A_1(\underline{x}_0, \underline{x}_1)) \\ \vdots \\ \underline{x}_n(t) = L^{-1}(A_{n-1}(\underline{x}_0, \dots, \underline{x}_{n-1})) \end{cases} \quad (6.176)$$

Example 6.13. Let's consider the following differential equation⁶:

$$\begin{cases} \ddot{x}(t) = \sin(x) \\ x(0) = \theta_0 \\ \dot{x}(0) = 0 \end{cases} \quad (6.177)$$

⁵K. K. Kataria, P. Vellaisamy, Simple Parametrization Methods For Generating Adomian Polynomials, available online at <http://pefmath.etf.rs>

⁶G Adomian, A review of the decomposition method in applied mathematics, Journal of Mathematical Analysis and Applications, Volume 135, Issue 2, 1988, Pages 501-544, [https://doi.org/10.1016/0022-247X\(88\)90170-9](https://doi.org/10.1016/0022-247X(88)90170-9).

Here the linear differential operator L is the double integration d^2/dt^2 . Consequently L^{-1} is the twofold definite integration $\int_0^t (\int_0^{\tau_1} d\tau_2) d\tau_1$. Furthermore, for this example \underline{h} does not exist and $N(x) = \sin(x)$. Using (6.173), the Adomian series expansion of $\sin(x)$ reads:

$$\begin{cases} A_0(x_0) = \sin(x_0) \\ A_1(x_0, x_1) = x_1 \cos(x_0) \\ A_2(x_0, x_1, x_2) = x_2 \cos(x_0) - \frac{x_1^2}{2} \sin(x_0) \\ A_3(x_0, x_1, x_2, x_3) = x_3 \cos(x_0) - \frac{1}{6} x_1^3 \cos(x_0) - x_1 x_2 \sin(x_0) \\ \vdots \end{cases} \quad (6.178)$$

Thus, using (6.176), we get the following approximation of $\underline{x}(t)$:

$$\underline{x}(t) \approx \sum_{i=0}^n \underline{x}_i(t) \quad (6.179)$$

where:

$$\begin{aligned} \underline{x}_0(t) &= \underline{x}(0) + L^{-1}(\underline{h}) = \theta_0 \\ \underline{x}_1(t) &= L^{-1}(A_0(\underline{x}_0)) = \int_0^t \int_0^{\tau_1} \sin(x_0) d\tau_2 d\tau_1 \\ &= \int_0^t \int_0^{\tau_1} \sin(\theta_0) d\tau_2 d\tau_1 = \frac{t^2}{2} \sin(\theta_0) \\ \underline{x}_2(t) &= L^{-1}(A_1(\underline{x}_0, \underline{x}_1)) = \int_0^t \int_0^{\tau_1} x_1 \cos(x_0) d\tau_2 d\tau_1 \\ &= \int_0^t \int_0^{\tau_1} \frac{\tau_2^2}{2} \sin(\theta_0) \cos(\theta_0) d\tau_2 d\tau_1 = \frac{t^4}{4!} \sin(\theta_0) \cos(\theta_0) \\ \underline{x}_3(t) &= L^{-1}(A_2(\underline{x}_0, \underline{x}_1, \underline{x}_2)) \\ &= \int_0^t \int_0^{\tau_1} \left(x_2 \cos(x_0) - \frac{x_1^2}{2} \sin(x_0) \right) d\tau_2 d\tau_1 \\ &= \frac{t^6}{6!} (\sin(\theta_0) \cos^2(\theta_0) - 3 \sin^3(\theta_0)) \\ &\vdots \end{aligned} \quad (6.180)$$

■

6.8.2 Nonlinear dynamical systems

Now we consider the following nonlinear state space representation where $f(\underline{x}(t), \underline{u}(t))$ is an analytic (meaning infinitely differentiable) vector field on \mathbb{R}^n :

$$\dot{\underline{x}}(t) = f(\underline{x}(t), \underline{u}(t)) \quad (6.181)$$

The solution $\underline{x}(t)$ of this differential equation reads as follows where L^{-1} is the onefold definite integration $\int_0^t d\tau$:

$$\begin{aligned} \underline{x}(t) &= \underline{x}(0) + L^{-1}(f(\underline{x}(t), \underline{u}(t))) \\ &= \underline{x}(0) + \int_0^t (f(\underline{x}(\tau), \underline{u}(\tau))) d\tau \end{aligned} \quad (6.182)$$

Assuming a Zero-Order Hold (ZOH) digital-to-analog converter (DAC), the control vector $\underline{u}(t)$ is a known constant over the sampling interval T :

$$\underline{u}(t) = \underline{u}(kT) := \underline{u}_k = \text{constant} \forall t \in [t_k, t_{k+1}) = [kT, (k+1)T) \quad (6.183)$$

Thus, when discretization the nonlinear dynamical system (6.182), $\underline{u}(\tau)$ is replaced by $\underline{u}_k = \text{constant } \forall t \in [kT, (k+1)T)$. We get:

$$\begin{aligned}\underline{x}_{k+1} &= \underline{x}(0) + \int_0^{(k+1)T} (f(\underline{x}(\tau), \underline{u}(\tau))) d\tau \\ &= \underline{x}_k + \int_{kT}^{(k+1)T} (f(\underline{x}(\tau), \underline{u}(\tau))) d\tau \\ &= \underline{x}_k + \int_{kT}^{(k+1)T} (f(\underline{x}(\tau), \underline{u}_k)) d\tau\end{aligned}\quad (6.184)$$

More generally, $\forall t \in [kT, (k+1)T)$, we can write:

$$\underline{x}(t) = \underline{x}_k + \int_{kT}^t (f(\underline{x}(\tau), \underline{u}_k)) d\tau \quad \forall t \in [kT, (k+1)T) \quad (6.185)$$

Following the Adomian decomposition method, $\underline{x}(t)$ and $f(\underline{x}(t), \underline{u}_k)$ are presented by the following series:

$$\begin{cases} \underline{x}(t) = \sum_{i=0}^{\infty} \underline{x}_i(t) \\ f(\underline{x}(t), \underline{u}_k) = \sum_{i=0}^{\infty} A_i(\underline{x}_0, \dots, \underline{x}_i) \end{cases} \quad (6.186)$$

The so-called Adomian polynomials $A_i(\underline{x}_0, \dots, \underline{x}_i)$ are computed as in (6.173):

$$A_i(\underline{x}_0, \dots, \underline{x}_i) = \frac{1}{i!} \left[\frac{d^i}{d\lambda^i} f \left(\sum_{j=0}^i \lambda^j \underline{x}_j, \underline{u}_k \right) \right]_{\lambda=0} \quad i = 0, 1, 2, \dots \quad (6.187)$$

Then equating terms with the same order on both sides of (6.184) leads to the approximation of order n of $\underline{x}(t)$. Denoting now L^{-1} the onefold definite integration $\int_{kT}^t d\tau$, we get:

$$\underline{x}(t) \approx \sum_{i=0}^n \underline{x}_i(t) \text{ where } \begin{cases} \underline{x}_0(t) = \underline{x}_k \\ \underline{x}_1(t) = L^{-1}(A_0(\underline{x}_0)) := \int_{kT}^t A_0(\underline{x}_0) d\tau \\ \underline{x}_2(t) = \int_{kT}^t A_1(\underline{x}_0, \underline{x}_1) d\tau \\ \vdots \\ \underline{x}_n(t) = \int_{kT}^t A_{n-1}(\underline{x}_0, \dots, \underline{x}_{n-1}) d\tau \end{cases} \quad (6.188)$$

Finally, setting t to $(k+1)T$ in $\underline{x}(t)$, we get the approximation of order n of the dynamics of the discretized plant:

$$\underline{x}((k+1)T) \approx \sum_{i=0}^n \underline{x}_i((k+1)T) \quad (6.189)$$

When $n = 1$, this corresponds to the Euler's method.

Example 6.14. We consider the following linear time invariant plant:

$$\dot{\underline{x}} = \mathbf{A}\underline{x} + \mathbf{B}\underline{u} \quad (6.190)$$

It is clear that we get from identification with (6.181):

$$f(\underline{x}, \underline{u}) = \mathbf{A}\underline{x} + \mathbf{B}\underline{u} \quad (6.191)$$

Following Adomian's procedure we get:

$$\begin{cases} A_0(\underline{x}_0) = \mathbf{A}\underline{x}_0 + \mathbf{B}\underline{u}_k \\ A_1(\underline{x}_0, \underline{x}_1) = \left[\frac{d}{d\lambda} (\mathbf{A}(\underline{x}_0 + \lambda \underline{x}_1) + \mathbf{B}\underline{u}_k) \right]_{\lambda=0} = \mathbf{A}\underline{x}_1 \\ A_2(\underline{x}_0, \underline{x}_1, \underline{x}_2) = \frac{1}{2!} \left[\frac{d^2}{d\lambda^2} (\mathbf{A}(\underline{x}_0 + \lambda \underline{x}_1 + \lambda^2 \underline{x}_2) + \mathbf{B}\underline{u}_k) \right]_{\lambda=0} = \mathbf{A}\underline{x}_2 \\ \vdots \\ A_n(\underline{x}_0, \dots, \underline{x}_n) = \mathbf{A}\underline{x}_n \end{cases} \quad (6.192)$$

Thus the approximation of order n of $\underline{x}(t)$ reads $\underline{x}(t) \approx \sum_{i=0}^n \underline{x}_i(t)$ where:

$$\begin{cases} \underline{x}_0(t) = \underline{x}_k \\ \underline{x}_1(t) = \int_{kT}^t A_0(\underline{x}_0) d\tau = (\mathbf{A}\underline{x}_k + \mathbf{B}\underline{u}_k) (t - kT) \\ \underline{x}_2(t) = \int_{kT}^t A_1(\underline{x}_0, \underline{x}_1) d\tau = \mathbf{A}(\mathbf{A}\underline{x}_k + \mathbf{B}\underline{u}_k) \frac{(t - kT)^2}{2} \\ \vdots \\ \underline{x}_n(t) = \int_{kT}^t A_{n-1}(\underline{x}_0, \dots, \underline{x}_{n-1}) d\tau = \mathbf{A}^{n-1} (\mathbf{A}\underline{x}_k + \mathbf{B}\underline{u}_k) \frac{(t - kT)^n}{n!} \end{cases} \quad (6.193)$$

Finally, setting t to $(k+1)T$ in $\underline{x}(t)$, we get the approximation of order n of the dynamics of the discretized plant:

$$\underline{x}((k+1)T) \approx \sum_{i=0}^n \underline{x}_i((k+1)T) \quad (6.194)$$

where:

$$\begin{cases} \underline{x}_0((k+1)T) = \underline{x}_k \\ \underline{x}_1((k+1)T) = (\mathbf{A}\underline{x}_k + \mathbf{B}\underline{u}_k) T \\ \underline{x}_2((k+1)T) = \mathbf{A}(\mathbf{A}\underline{x}_k + \mathbf{B}\underline{u}_k) \frac{T^2}{2} \\ \vdots \\ \underline{x}_n((k+1)T) = \mathbf{A}^{n-1} (\mathbf{A}\underline{x}_k + \mathbf{B}\underline{u}_k) \frac{T^n}{n!} \end{cases} \quad (6.195)$$

When $n \rightarrow \infty$, and assuming that \mathbf{A}^{-1} exists, we get the following series:

$$\begin{aligned} \underline{x}((k+1)T) &= \sum_{i=0}^{\infty} \underline{x}_i((k+1)T) \\ &= \left(\sum_{i=0}^{\infty} \frac{(\mathbf{A}T)^i}{i!} \right) \underline{x}_k + \mathbf{A}^{-1} \left(\sum_{i=1}^{\infty} \frac{(\mathbf{A}T)^i}{i!} \right) \mathbf{B}\underline{u}_k \\ &= \left(\sum_{i=0}^{\infty} \frac{(\mathbf{A}T)^i}{i!} \right) \underline{x}_k + \mathbf{A}^{-1} \left(\sum_{i=0}^{\infty} \frac{(\mathbf{A}T)^i}{i!} - \mathbb{I} \right) \mathbf{B}\underline{u}_k \\ &= e^{\mathbf{A}T} \underline{x}_k + \mathbf{A}^{-1} (e^{\mathbf{A}T} - \mathbb{I}) \mathbf{B}\underline{u}_k \end{aligned} \quad (6.196)$$

We clearly retrieve result (6.143). ■

Example 6.15. We consider the following differential equation where $a \neq 0$, b and c are constant:

$$\begin{cases} a\ddot{\theta}(t) + b\dot{\theta}(t) + c\sin(\theta(t)) = u(t) \\ \theta(0) = \theta_0 \\ \dot{\theta}(0) = \dot{\theta}_0 \end{cases} \quad (6.197)$$

Let $x_1 := \theta$ and $x_2 := \dot{\theta}$. Thus the state space representation of this differential equation reads:

$$\begin{cases} x_1 := \theta \\ x_2 := \dot{\theta} \end{cases} \Rightarrow \begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = \frac{1}{a}(u - b x_2 - c \sin(x_1)) \end{cases} \quad (6.198)$$

Or, in vectorial form:

$$\underline{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \Rightarrow \dot{\underline{x}} = \begin{bmatrix} x_2 \\ \frac{1}{a}(u - b x_2 - c \sin(x_1)) \end{bmatrix} := f(\underline{x}, \underline{u}) \quad (6.199)$$

Following Adomian's procedure we get:

$$\left\{ \begin{array}{l} A_0(\underline{x}_0) = f(\underline{x}_0, \underline{u}_k) \\ = \begin{bmatrix} x_{0,2} \\ \frac{1}{a}(u_k - b x_{0,2} - c \sin(x_{0,1})) \end{bmatrix} \\ A_1(\underline{x}_0, \underline{x}_1) = \left[\frac{d}{d\lambda} (f(\underline{x}_0 + \lambda \underline{x}_1, \underline{u}_k)) \right]_{\lambda=0} \\ = \frac{d}{d\lambda} \begin{bmatrix} x_{0,2} + \lambda x_{1,2} \\ \frac{1}{a}(u_k - b(x_{0,2} + \lambda x_{1,2}) - c \sin(x_{0,1} + \lambda x_{1,1})) \end{bmatrix}_{\lambda=0} \\ = \begin{bmatrix} x_{1,2} \\ \frac{1}{a}(-b x_{1,2} - c x_{1,1} \cos(x_{0,1})) \end{bmatrix} \\ A_2(\underline{x}_0, \underline{x}_1, \underline{x}_2) = \frac{1}{2!} \left[\frac{d^2}{d\lambda^2} (f(\underline{x}_0 + \lambda \underline{x}_1 + \lambda^2 \underline{x}_2, \underline{u}_k)) \right]_{\lambda=0} \\ = \begin{bmatrix} x_{2,2} \\ \frac{1}{a} \left(-b x_{2,2} - c \left(x_{2,1} \cos(x_{0,1}) - \frac{x_{1,1}^2}{2} \sin(x_{0,1}) \right) \right) \end{bmatrix} \\ \vdots \end{array} \right. \quad (6.200)$$

Thus the approximation of order n of $\underline{x}(t)$ reads $\underline{x}(t) \approx \sum_{i=0}^n \underline{x}_i(t)$ where:

$$\left\{ \begin{array}{l} \underline{x}_0(t) = \underline{x}_k := \begin{bmatrix} x_{0,1} \\ x_{0,2} \end{bmatrix} \\ \underline{x}_1(t) = \int_{kT}^t A_0(\underline{x}_0) d\tau \\ = \begin{bmatrix} x_{0,2} \\ \frac{1}{a}(u_k - b x_{0,2} - c \sin(x_{0,1})) \end{bmatrix} (t - kT) \\ \underline{x}_2(t) = \int_{kT}^t A_1(\underline{x}_0, \underline{x}_1) d\tau \\ = \int_{kT}^t \begin{bmatrix} x_{1,2} \\ \frac{1}{a}(-b x_{1,2} - c x_{1,1} \cos(x_{0,1})) \end{bmatrix} d\tau \\ = \begin{bmatrix} \frac{1}{a}(u_k - b x_{0,2} - c \sin(x_{0,1})) \\ -\frac{b}{a^2}(u_k - b x_{0,2} - c \sin(x_{0,1})) - \frac{c}{a} x_{0,2} \cos(x_{0,1}) \end{bmatrix} \frac{(t - kT)^2}{2} \\ \vdots \end{array} \right. \quad (6.201)$$

Finally, setting t to $(k+1)T$ in $\underline{x}(t)$, we get the approximation of order n of the dynamics of the discretized plant:

$$\underline{x}((k+1)T) \approx \sum_{i=0}^n \underline{x}_i((k+1)T) \quad (6.202)$$

where:

$$\begin{cases} \underline{x}_0((k+1)T) = \underline{x}_k := \begin{bmatrix} x_{0,1} \\ x_{0,2} \end{bmatrix} \\ \underline{x}_1((k+1)T) = \begin{bmatrix} x_{0,2} \\ \frac{1}{a}(u_k - b x_{0,2} - c \sin(x_{0,1})) \end{bmatrix} T \\ \underline{x}_2((k+1)T) = \begin{bmatrix} \frac{1}{a}(u_k - b x_{0,2} - c \sin(x_{0,1})) \\ -\frac{b}{a^2}(u_k - b x_{0,2} - c \sin(x_{0,1})) - \frac{c}{a} x_{0,2} \cos(x_{0,1}) \end{bmatrix} \frac{T^2}{2} \\ \vdots \end{cases} \quad (6.203)$$

6.9 Derivation of discrete time controller from continuous time controller

6.9.1 Tustin's approximation

Tustin's approximation (or bilinear transformation) is frequently used to derive a discrete time controller that closely approximates the behaviour of the original continuous time controller.

Tustin's approximation comes from the first order approximation of $e^{\frac{sT}{2}}$. Indeed:

$$z = e^{sT} = \frac{e^{\frac{sT}{2}}}{e^{-\frac{sT}{2}}} \approx \frac{1 + \frac{sT}{2}}{1 - \frac{sT}{2}} \leftrightarrow s = \frac{2}{T} \frac{z-1}{z+1} \quad (6.204)$$

Consequently the discrete time controller $C_d(z)$ approximated from continuous time controller $C(s)$ is obtained by replacing s by $\frac{2}{T} \frac{z-1}{z+1}$:

$$C_d(z) = C(s)|_{s=\frac{2}{T} \frac{z-1}{z+1}} \quad (6.205)$$

Tustin's approximation has the advantage that the left half s-plane is transformed into the unit disc in the z-plane; indeed Tustin's approximation is an homographic transformation as illustrated in Figure 6.5.

Example 6.16. Consider the following PI continuous time controller whose transfer function is:

$$C_{PI}(s) = K_p + \frac{K_i}{s} \quad (6.206)$$

Applying the Tustin's approximation leads to the following discrete time controller:

$$C_d(z) = C(s)|_{s=\frac{2}{T} \frac{z-1}{z+1}} = K_p + K_i \frac{T}{2} \frac{z+1}{z-1} \quad (6.207)$$

One of the problems with the Tustin's approximation method is that the frequency scale is distorted during the approximation. Recall that to evaluate the frequency response the transfer function is evaluated along the imaginary axis $s = j\omega$ for continuous time systems whereas for discrete time systems the

transfer function is evaluated along the unit circle $z = e^{j\alpha} = e^{j\omega T}$. When the discrete time transfer function is obtained by using the Tustin's approximation on a continuous transfer function, the continuous frequency ω_c is mapped to the discrete frequency ω_d according to the following relation:

$$s = j\omega_c \Rightarrow \frac{2}{T} \frac{z-1}{z+1} = \frac{2}{T} \frac{e^{j\omega_c T} - 1}{e^{j\omega_c T} + 1} = \frac{2}{T} \frac{e^{\frac{j\omega_c T}{2}} - e^{-\frac{j\omega_c T}{2}}}{e^{\frac{j\omega_c T}{2}} + e^{-\frac{j\omega_c T}{2}}} = j \frac{2}{T} \tan\left(\frac{\omega_c T}{2}\right) \quad (6.208)$$

That is:

$$\omega_c \rightarrow \omega_d = \frac{2}{T} \tan\left(\frac{\omega_c T}{2}\right) \quad (6.209)$$

Consequently the frequency scale is distorted. This will create mismatch in the system's response if the controller design is based on precise frequency requirements. At low frequency the distortion is small. However close to the Nyquist frequency the distortion becomes significant. In order to match the continuous-time design at a specific frequency ω_c the Tustin's approximation can be modified by using a pre-warping frequency:

$$s \rightarrow \frac{\omega_c}{\tan\left(\frac{\omega_c T}{2}\right)} \frac{z-1}{z+1} \quad (6.210)$$

There are, however, still distortions at other frequencies.

Example 6.17. Consider the following continuous time controller whose transfer function is:

$$C(s) = \frac{1}{1 + \tau s} \quad (6.211)$$

Applying the Tustin's approximation with ω_c as the pre-warping frequency leads to the following discrete time controller:

$$\begin{aligned} C_d(z) &= C(s) \Big|_{s = \frac{\omega_c}{\tan\left(\frac{\omega_c T}{2}\right)} \frac{z-1}{z+1}} \\ &= \frac{1}{1 + \tau \frac{\omega_c}{\tan\left(\frac{\omega_c T}{2}\right)} \frac{z-1}{z+1}} \\ &= \frac{z+1}{z+1 + (z-1) \frac{\tau \omega_c}{\tan\left(\frac{\omega_c T}{2}\right)}} \\ &= \frac{1}{1 + \frac{\tau \omega_c}{\tan\left(\frac{\omega_c T}{2}\right)} \frac{z+1}{z+1}} \end{aligned} \quad (6.212)$$

Where:

$$a = \frac{1 - \frac{\tau \omega_c}{\tan\left(\frac{\omega_c T}{2}\right)}}{1 + \frac{\tau \omega_c}{\tan\left(\frac{\omega_c T}{2}\right)}} \quad (6.213)$$

Setting ω_c to $\frac{1}{\tau}$ leads to the following discrete time controller:

$$\omega_c = \frac{1}{\tau} \Rightarrow C_d(z) = \frac{1}{1 + \frac{1}{\tan\left(\frac{T}{2\tau}\right)}} \frac{z+1}{z+a} \quad (6.214)$$

Where:

$$a = \frac{1 - \frac{\tau \omega_c}{\tan\left(\frac{\omega_c T}{2}\right)}}{1 + \frac{\tau \omega_c}{\tan\left(\frac{\omega_c T}{2}\right)}} = \frac{1 - \frac{1}{\tan\left(\frac{T}{2\tau}\right)}}{1 + \frac{1}{\tan\left(\frac{T}{2\tau}\right)}} \quad (6.215)$$

■

6.9.2 State-space realization of bilinear conformal mapping

We consider the following *bilinear map* on the Laplace transform variable s where a and $b \neq -\frac{1}{a}$ are real numbers:

$$s \mapsto z = \frac{s+a}{1+bs} \Leftrightarrow z \mapsto s = \frac{z-a}{1-bz} \quad (6.216)$$

The function $s \mapsto z$ maps the circle in the complex s -plane centred on $(\frac{-a-1/b}{2}, 0)$ and having diameter $|-a+1/b|$ onto the $j\omega$ -axis in the s -plane and it maps the $j\omega$ -axis into a similar circle in the Right-Half-Plane RHP⁷.

Furthermore assume that transfer function $\mathbf{F}(s)$ as the following realization:

$$\mathbf{F}(s) = \mathbf{C}(s\mathbb{I} - \mathbf{A})^{-1}\mathbf{B} + \mathbf{D} := \left(\begin{array}{c|c} \mathbf{A} & \mathbf{B} \\ \hline \mathbf{C} & \mathbf{D} \end{array} \right) \quad (6.217)$$

Then the conformal map $s \mapsto z$ yields⁷:

$$\tilde{F}(z) = F(s)|_{s=\frac{z-a}{1-bz}} = \tilde{\mathbf{C}}(z\mathbb{I} - \tilde{\mathbf{A}})^{-1}\tilde{\mathbf{B}} + \tilde{\mathbf{D}} := \left(\begin{array}{c|c} \tilde{\mathbf{A}} & \tilde{\mathbf{B}} \\ \hline \tilde{\mathbf{C}} & \tilde{\mathbf{D}} \end{array} \right) \quad (6.218)$$

where:

$$\begin{cases} \tilde{\mathbf{A}} = (\mathbf{A} + a\mathbb{I})(\mathbb{I} + b\mathbf{A})^{-1} \\ \tilde{\mathbf{B}} = (1 - ab)(\mathbb{I} + b\mathbf{A})^{-1}\mathbf{B} \\ \tilde{\mathbf{C}} = \mathbf{C}(\mathbb{I} + b\mathbf{A})^{-1} \\ \tilde{\mathbf{D}} = \mathbf{D} - b\mathbf{C}(\mathbb{I} + b\mathbf{A})^{-1}\mathbf{B} \end{cases} \quad (6.219)$$

The proof of the preceding property follows by inspection: we check that the following property holds:

$$\tilde{F}(z)|_{z=\frac{s+a}{1+bs}} = \tilde{\mathbf{C}}\left(\frac{s+a}{1+bs}\mathbb{I} - \tilde{\mathbf{A}}\right)^{-1}\tilde{\mathbf{B}} + \tilde{\mathbf{D}} = \mathbf{C}(s\mathbb{I} - \mathbf{A})^{-1}\mathbf{B} + \mathbf{D} = F(s) \quad (6.220)$$

Similarly, the inverse transformation $s = \frac{z-a}{1-bz}$ produces the same effect on the state-space matrices, except that a and b are replaced by $-a$ and $-b$, respectively.

Furthermore, if ω_0 is not an eigenvalue of \mathbf{A} , then the Tustin's map $s \mapsto z = \frac{\omega_0+s}{\omega_0-s}$ changes transfer function $F(s)$ as follows⁸:

$$\tilde{F}(z) = F(s)|_{s=\omega_0\frac{z-1}{z+1}} = \tilde{\mathbf{C}}(z\mathbb{I} - \tilde{\mathbf{A}})^{-1}\tilde{\mathbf{B}} + \tilde{\mathbf{D}} := \left(\begin{array}{c|c} \tilde{\mathbf{A}} & \tilde{\mathbf{B}} \\ \hline \tilde{\mathbf{C}} & \tilde{\mathbf{D}} \end{array} \right) \quad (6.221)$$

where:

$$\begin{cases} \tilde{\mathbf{A}} = (\mathbf{A} + \omega_0\mathbb{I})(\omega_0\mathbb{I} - \mathbf{A})^{-1} \\ \tilde{\mathbf{B}} = \sqrt{2\omega_0}(\omega_0\mathbb{I} - \mathbf{A})^{-1}\mathbf{B} \\ \tilde{\mathbf{C}} = \sqrt{2\omega_0}\mathbf{C}(\omega_0\mathbb{I} - \mathbf{A})^{-1} \\ \tilde{\mathbf{D}} = F(\omega_0) = \mathbf{D} + \mathbf{C}(\omega_0\mathbb{I} - \mathbf{A})^{-1}\mathbf{B} \end{cases} \quad (6.222)$$

⁷M. G. Safonov, Imaginary-axis zeros in multivariable H-optimal control, Modelling, Robustness and Sensitivity Reduction in Control Systems, pp. 71-81, 1987

⁸Jarmo Malinen, The Cayley Transform as a Time Discretization Scheme, Numerical Functional Analysis and Optimization 28(7):825-851, August 2007, DOI: 10.1080/01630560701493321

Similarly, if -1 is not an eigenvalue of \tilde{A} , then the inverse transformation $z \mapsto s = \omega_0 \frac{z-1}{z+1}$ produces the following effect on the state-space realization of $\tilde{F}(z)$ ⁸:

$$\tilde{F}(z) \Big|_{z=\frac{\omega_0+s}{\omega_0-s}} = \mathbf{C} (s\mathbb{I} - \mathbf{A})^{-1} \mathbf{B} + \mathbf{D} := \left(\begin{array}{c|c} \mathbf{A} & \mathbf{B} \\ \hline \mathbf{C} & \mathbf{D} \end{array} \right) = F(s) \quad (6.223)$$

where:

$$\begin{cases} \mathbf{A} = \omega_0 (\tilde{A} - \mathbb{I}) (\mathbb{I} + \tilde{A})^{-1} \\ \mathbf{B} = \sqrt{2\omega_0} (\mathbb{I} + \tilde{A})^{-1} \tilde{\mathbf{B}} \\ \mathbf{C} = \sqrt{2\omega_0} \tilde{\mathbf{C}} (\mathbb{I} + \tilde{A})^{-1} \\ \mathbf{D} = \tilde{\mathbf{D}} - \tilde{\mathbf{C}} (\mathbb{I} + \tilde{A})^{-1} \tilde{\mathbf{B}} \end{cases} \quad (6.224)$$

6.9.3 Matched Pole-Zero (MPZ) mapping method

The purpose of this method is to map the poles and the zeros of the continuous time controller with the poles and the zeros of the discrete time controller according to the relation $z = e^{pT}$ where p is either a pole or a zero of the continuous time controller and to preserve the static gain of the continuous time controller in the discrete time domain.

The steps of the Matched Pole-Zero (MPZ) mapping method are the following:

- Map the poles and the zeros of the continuous time controller with the poles and the zeros of the discrete time controller according to the following relation where s is either a pole or a zero of the continuous time controller:

$$z = e^{sT} \quad (6.225)$$

- If the degree of the numerator of the discrete time controller is lower than the degree of its denominator then multiply by $z + 1$ the numerator until that the degree of the numerator of the discrete time controller becomes equal to the degree of its denominator.

The purpose of this step is to map the high frequency behaviours of the discrete time and continuous time and controllers.

- Multiply the transfer function of the discrete time controller by a gain K_n and set it to identify the static gain of the discrete time controller with the static gain of the continuous time controller:

$$\lim_{s \rightarrow 0} C(s) = \lim_{z \rightarrow 1} C_d(z) \quad (6.226)$$

Example 6.18. Consider the following continuous time controller whose transfer function is:

$$C(s) = \frac{1}{1 + \tau s} \quad (6.227)$$

It is clear that the pole of $C(s)$ is $-\frac{1}{\tau}$ whereas $C(s)$ has no zero.

6.9. Derivation of discrete time controller from continuous time controller 207

- Map the poles and the zeros of the continuous time controller with the poles and the zeros of the discrete time controller according to the following relation where s is either a pole or a zero of the continuous time controller:

$$C_{d1}(z) = \frac{1}{z - e^{-\frac{T}{\tau}}} \quad (6.228)$$

- If the degree of the numerator of the discrete time controller is lower than the degree of its denominator then multiply by $z + 1$ the numerator until that the degree of the numerator of the discrete time controller becomes equal to the degree of its denominator. We get:

$$C_{d2}(z) = \frac{z + 1}{z - e^{-\frac{T}{\tau}}} \quad (6.229)$$

- Multiply the transfer function of the discrete time controller by a gain K_n and set it to identify the static gain of the discrete time controller with the static gain of the continuous time controller. We get:

$$C_d(z) = K_n \frac{z + 1}{z - e^{-\frac{T}{\tau}}} \quad (6.230)$$

Gain K_n is set such that:

$$\lim_{s \rightarrow 0} C(s) = \lim_{z \rightarrow 1} C_d(z) \Leftrightarrow 1 = K_n \frac{2}{1 - e^{-\frac{T}{\tau}}} \Rightarrow K_n = \frac{1 - e^{-\frac{T}{\tau}}}{2} \quad (6.231)$$

Thus the Matched Pole-Zero (MPZ) mapping method leads to the following discrete time controller:

$$C_d(z) = \frac{1 - e^{-\frac{T}{\tau}}}{2} \frac{z + 1}{z - e^{-\frac{T}{\tau}}} \quad (6.232)$$

■

