



Summary

Mathematical tools.....	4
Signals definition and commonly used signals	4
Linear differential equations	5
Linear difference equations.....	7
Integral Transforms	7
Why integral transforms.....	7
Laplace transform: definition	8
Properties of the Laplace transform.....	10
Partial fraction expansion.....	10
Fourier transform: definition.....	14
Properties of the Fourier transform	14
Z-transform	16
Properties of Z transform and commonly used transformations.....	17
How to obtain the Z transform	18
Modelling of dynamical systems	19
Generalities on dynamical systems	19
Mathematical description and properties.....	22
State Space description	26
State space description for linear systems in continuous time.....	26
State Space description for linear systems in discrete time	30
Impulse response and transfer functions.....	30
Impulse Response and convolution.....	30
Continuous time transfer functions	31
Transfer functions formulations	35
Discrete time transfer function	36
From state space to transfer function.....	38
Behavioral Analysis.....	38
Free response and forced response	38
Free response from the state space.....	39
Computing the free evolution from the state space	39
Matrix exponential	40
Natural modes	41



Free response from the transfer function	45
Forced Response	46
Step response	46
Polynomial response	47
The transfer function in steady state: frequency response	47
Control Systems Analysis	50
Time Domain Characteristics	50
Systems Response Characteristics in time-domain	51
Qualitative Techniques (Poles – Zeros concept)	51
System Stability	53
Concept.....	53
Definition	54
Formal Definition.....	54
Internal Stability of CT Systems	55
Asymptotic Stability of CT Systems	55
BIBO Stability of CT Systems.....	55
Internal Stability of DT Systems.....	55
Asymptotic Stability of DT Systems	56
BIBO Stability of DT Systems	56
Can a system be BIBO stable and not internally stable?	56
Importance of Stability	57
Testing Techniques	58
Frequency Domain Methods	58
Representation and Plotting "Bode"	58
What is a Polar Plot?.....	61
Nyquist Analysis.....	63
Complex Functions	63
Derivative.....	64
Singular Points	65
Closed Contours.....	65
Enclosure	65
Clockwise direction.....	65
Encirclements	66
Properties of the Mapping $P(s)$	66



Systems Stability	67
Gain Margin and Phase Margin	72
Gain Margin	72
Phase Margin	72
Evaluating Gain and Phase Margins by Nyquist Diagram	74
Performance Specifications in the Frequency Domain	74
Constant Magnitude Circles (M-Circles)	77
Nichols Charts	78
Phase Margin and the Damping factor	79
Feedback Control and its Advantages	80
Examples of Control systems	81
Performance	83
First Order Systems	84
Second Order Systems	85
The General Second Order System	86
Natural Frequency	86
Damping Ratio	86
Under-damped Second Order System Response	86
Block Diagrams	87
Common block diagrams configurations	88
Cascade Form	88
Parallel Form	88
Feedback Form	89
Examples of Control Systems Design	89
State Feedback	90
Controllability	93
Observer Design	95
Observability	97
State Feedback Example	99
PID Control	104
Definition	104
Proportional Action	105
Integral Action	106
Derivative Action	108



All PID actions	109
Tuning Methods.....	110
MATLAB PID Tuner	110
Lead Compensators	113
Lag Compensator.....	117
Control design using Lead-Lag Compensation	120
Deriving steady state requirements for the tracking error	121
Deriving steady state requirements for the plant noise rejection	122
Translate transient requirements to a range for crossover frequency and phase margin	122
Plot the Bode diagram for the uncompensated system.....	123
Design the Lead part, to add the needed phase, at the desired frequency.....	124
Useful concepts in probability theory	131
Random Variables.....	131
Probability distribution function	131
Probability Density function	132
Expected values and momenta	133
Gaussian (normal) probability density function	134
Further arguments.....	134

MATHEMATICAL TOOLS

SIGNALS DEFINITION AND COMMONLY USED SIGNALS

We can define a signal as any **time-dependent** physical quantity; we use signal to transmit and receive information. Their natural mathematical representation is a function of time of the kind:

$$S: \mathbb{R} \rightarrow \mathbb{R}$$

For real-valued signals, or

$$S: \mathbb{R} \rightarrow \mathbb{C}$$

For complex-valued signals

This formulation is intended for signals that are defined for every instant of time, that is for **analog signals**. Analog signals can be discontinuous functions, but are defined over the real domain.

Another kind of signal is the **discrete time signal**, a function of time of the form:

$$S_d: \mathbb{Z} \rightarrow \mathbb{R}$$

For real-valued discrete time signals or:

$$S_d: \mathbb{Z} \rightarrow \mathbb{C}$$

For complex-valued discrete time signals.

A discrete time signal is a **sampled** representation of a continuous time signal, namely, a collection of values that the continuous time signals assumes at particular time instants; these time instants are commonly multiples of a quantity called **sampling time** T_s .

With the definition of the sampling time, a discrete time signal can be defined over instants of time that are not integers, but the resulting domain will always be a sequence of discrete values to which we can refer with integer numbers.

An example of analog signal can be the following:

$$S(t) = \sin(\omega t)$$

While the respective sampled version looks like:

$$S(nT_s) = \sin(\omega nT_s) \quad \text{with } T \in \mathbb{R}; \quad n = 1, 2, 3 \dots$$

In automatic control area, the most used analog signal are the following:

- **Dirac's Delta:** an impulsive signal denoted with $\delta(t)$ or $u_{-1}(t)$ that is non-zero only in the origin and that carries a finite quantity of energy (more about this in the links)
- **Heaviside step function:** a signal denoted with $u_0(t)$ defined as:

$$u_0(t) = \begin{cases} 1 & \text{for } t > 0 \\ \frac{1}{2} & \text{for } t = 0 \\ 0 & \text{for } t < 0 \end{cases}$$

- **Polynomial signals** of the form $u_n(t) = \frac{t^n}{n!}$, for which the factorial scaling is defined only for the aim of the common calculations to be applied at the signal.

Discrete-time equivalent versions of the above signals are always possible.

Useful links:

- About Dirac's Delta and step function:
<https://ocw.mit.edu/courses/mathematics/18-03sc-differential-equations-fall-2011/unit-iii-fourier-series-and-laplace-transform/step-and-delta-functions-integrals-and-generalized-derivatives/> from MIT university.
- About the sampling process and discretization theory:
https://ccrma.stanford.edu/~jos/pasp/Sampling_Theory.html from Stanford University

LINEAR DIFFERENTIAL EQUATIONS

We will deal with **linear differential equations** (LDE). In such equations, the unknown function and its derivatives are **linearly combined**. We will manage functions of one variable, so that the class of equations on which we will focus are the **Ordinary differential equations** (ODE); we will implicitly

assume that the independent variable for the unknown function is time, so that the form of the equations will be like the following:

$$\dot{y} - K(t)y = 0$$

$$a\ddot{y} - b\dot{y} = 0$$

$$a\ddot{y} - b\ddot{y} + c\dot{y} + dy = g(t)$$

We briefly explain some properties and definitions about these kind of equations:

- 1) The **order** of the differential equation is given by the higher order of derivative that appears in the equation; some degree of derivative can be missing, as well as the function itself, like in the second equation above.
- 2) The known term can be equal or different from zero ($g(t)$ in the third equation); in the first case we speak about **homogeneous** differential equation, **non-homogeneous** in the other case.
- 3) The coefficients of the unknown function can be **constant** or **time variant** ($K(t)$ in the first equation).
- 4) The solution of a generic linear differential equation is given by the **sum of two functions**: $z(t)$ is the solution of the homogeneous associated equation (that is the equation in which $g(t)$, if present, is set to zero); $\varphi(t)$, that is one particular solution of the non-homogeneous differential equation.
- 5) To solve the LDE in closed form, we need to know the **initial conditions**, that is, we need to solve the Cauchy problem. The number of initial conditions must be equal to the degree of the differential equation.
- 6) An n -th order linear differential equation can be decomposed in a **system** of n first order differential equations.
- 7) For the linear differential equations, the **principle of superposition** holds: if $y_1(t), y_2(t)$ are two solutions of a linear differential equation, the function $y_{tot}(t) = c_1y_1(t) + c_2y_2(t)$ with $c_1, c_2 \in \mathbb{R}$ is still a solution for the LDE.

Useful links:

-“Differential Equations and Dynamical Systems”, Lawrence Perko, Springer

-http://mathinsight.org/ordinary_differential_equation_introduction (from Math Insight)

-<https://www.youtube.com/watch?v=tF9eEGs8aao> (lesson by Dr Chris Tisdell)

-<http://tutorial.math.lamar.edu/Classes/DE/IntroBasic.aspx> (From Paul’ online Math notes)

-<http://www.math.psu.edu/tseng/class/Math251/Notes-LinearSystems.pdf> (From Penn State Univesity).

EXAMPLE: FROM HIGH ORDER DIFFERENTIAL EQUATION TO A SYSTEM OF FIRST ORDER DIFFERENTIAL EQUATIONS



Given the following differential equation:

$$6\ddot{y} - 4\dot{y} + y = 0$$

We can write the substitution:

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

From which we get:

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = \ddot{y} \end{cases}$$

And:

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = \frac{4x_2 - x_1}{6} \end{cases}$$

We obtained a first order system of LDE from a second order one.

LINEAR DIFFERENCE EQUATIONS

In the field of control, most of the time the algorithms and the data acquisition systems are digital; this leads to refer to the discrete counterpart of differential equations, that is, difference equations. Even in this formulation, the equations that we will treat are linear, and take the form:

$$y(k-1) - \alpha(k)y(k) = 0$$

$$ay(k-2) - by(k-1) = 0$$

$$ay(k-3) - by(k-2) + cy(k-1) + dy(k) = g(k)$$

Where the unknown discrete sequence $y(k)$ represents a discrete time signal that satisfies the relation imposed by the equation. Linear difference equations can be characterized with the same properties that we exposed for the linear differential equations.

INTEGRAL TRANSFORMS

WHY INTEGRAL TRANSFORMS

The study of dynamical systems can be tackled in two different domains: **time domain** and **complex variable** domains. This is possible due to three main integral transformations that allow us to

describe a system of differential (difference) equations in terms of a system of algebraic equations of complex variables. In the case of the linear systems, this operation is very useful since the inverse transformation of such a system can always be calculated in closed form.

The three transformations that we are going to use are:

- **Laplace transform:** transforms a function of the real variable t into a function of the complex variable $s = \sigma + j\omega$.
- **Fourier transform:** is a particular case of the Laplace transform: transforms a function of the real variable t into a function of the complex variable $j\omega$. The Fourier transform allows to operate a frequency domain analysis of signals and systems.
- **Z-Transform:** is an equivalent of the Laplace transform applied to discrete time signals. It the main tool in the area of digital control and signal processing.

LAPLACE TRANSFORM: DEFINITION

The (unilateral) Laplace transform of a function of real value t is defined as follows:

$$F(s) = \mathcal{L}\{f(t)\} = \int_0^{\infty} f(t)e^{-st} dt$$

And:

$$f(t) = \mathcal{L}^{-1}\{F(s)\} = \frac{1}{2\pi j} \lim_{T \rightarrow \infty} \int_{\gamma - iT}^{\gamma + iT} F(s)e^{st} ds$$

For practical purposes, the computation of the direct Laplace transform is made using **pre-compiled tables** of the transformations, while the inverse Laplace transform is made using the **partial fraction expansion**, that avoids the resolution of the integrals.

A list of the most commons Laplace transforms is the following:



$f(t) \quad (t \geq 0)$	$F(s)$
$\delta(t)$ (unit impulse)	1
$u(t)$ (unit step)	$\frac{1}{s}$
t (unit ramp)	$\frac{1}{s^2}$
$t^n \quad (n > -1)$	$\frac{n!}{s^{n+1}}$
e^{-at}	$\frac{1}{s+a}$
$\sin bt$	$\frac{b}{s^2 + b^2}$
$\cos bt$	$\frac{s}{s^2 + b^2}$
$e^{-at} \sin \omega t$	$\frac{\omega}{(s+a)^2 + \omega^2}$
$e^{-at} \cos \omega t$	$\frac{s+a}{(s+a)^2 + \omega^2}$
$\frac{\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2}$	$\frac{\omega_n}{\sqrt{1-\xi^2}} e^{-\xi\omega_n t} \sin \omega_n \sqrt{1-\xi^2} t \quad (\xi < 1)$
$\frac{\omega_n^2}{s(s^2 + 2\xi\omega_n s + \omega_n^2)}$	$1 - \frac{\omega_n}{\sqrt{1-\xi^2}} e^{-\xi\omega_n t} \sin(\omega_n \sqrt{1-\xi^2} t + \theta)$ where $\theta = \cos^{-1} \xi \quad (\xi < 1)$
$\frac{df(t)}{dt}$	$sF(s) - f(0)$
$\int_0^t f(t) dt$	$\frac{1}{s} F(s)$
$f(t-a)$	$e^{-as} F(s)$

Useful Links:

-“Control systems fundamentals” William S. Levine, CRC press LLC

-<https://www.youtube.com/watch?v=ZGPtPkTft8g> a graphical interpretation by Brian Douglas.

<https://www.youtube.com/watch?v=sZ2qulI6GEk> from MIT openCourseWare

PROPERTIES OF THE LAPLACE TRANSFORM

From the definition, the following properties of the Laplace transform derive:

	Property/Theorem	Time Domain	Complex Frequency Domain
1	Linearity	$c_1 f_1(t) + c_2 f_2(t) + \dots + c_n f_n(t)$	$c_1 F_1(s) + c_2 F_2(s) + \dots + c_n F_n(s)$
2	Time Shifting	$f(t-a)u_0(t-a)$	$e^{-as}F(s)$
3	Frequency Shifting	$e^{-as}f(t)$	$F(s+a)$
4	Time Scaling	$f(at)$	$\frac{1}{a}F\left(\frac{s}{a}\right)$
5	Time Differentiation See also (2.18) through (2.20)	$\frac{d}{dt}f(t)$	$sF(s) - f(0^-)$
6	Frequency Differentiation See also (2.22)	$tf(t)$	$-\frac{d}{ds}F(s)$
7	Time Integration	$\int_{-\infty}^t f(\tau)d\tau$	$\frac{F(s)}{s} + \frac{f(0^-)}{s}$
8	Frequency Integration	$\frac{f(t)}{t}$	$\int_s^{\infty} F(s)ds$
9	Time Periodicity	$f(t+nT)$	$\frac{\int_0^T f(t)e^{-st}dt}{1-e^{-sT}}$
10	Initial Value Theorem	$\lim_{t \rightarrow 0} f(t)$	$\lim_{s \rightarrow \infty} sF(s) = f(0^-)$
11	Final Value Theorem	$\lim_{t \rightarrow \infty} f(t)$	$\lim_{s \rightarrow 0} sF(s) = f(\infty)$
12	Time Convolution	$f_1(t)*f_2(t)$	$F_1(s)F_2(s)$
13	Frequency Convolution	$f_1(t)f_2(t)$	$\frac{1}{2\pi j} F_1(s)*F_2(s)$

The most used in practice are:

- Time differentiation
- Time integration
- Time convolution
- Final value theorem (Pay attention: this theorem can be applied if and only if the limit in the time domain exists)

That allow to get an algebraic equation starting from an integral-differential one, and to compute a **convolution as a simple multiplication**. Further theory about Laplace transform.

PARTIAL FRACTION EXPANSION

The transformation of the linear differential equations with constant coefficients generally leads to the calculus of inverse Laplace transform of a **rational fraction** of the type:

$$F(s) = \frac{b_m s^m + b_{m-1} s^{m-1} + \dots + b_0}{a_n s^n + a_{n-1} s^{n-1} + \dots + a_0}$$

The roots of the numerator are called **zeros**, while the roots of the denominator are called **poles**. Since the coefficients of the function are real numbers, the zeros and the poles will be real or complex

conjugate numbers.

Since a rational fraction can be decomposed in a linear combination of terms of the form:

$$F(s) = \sum_{i=1}^n \frac{r_i}{(s-p)^j}$$

Where p is a pole of the $F(s)$ function and j is its **multiplicity**. The coefficient r_i is called **residue** associated with the pole p and is a real or complex coefficient. Since this kind of term has an immediate anti-transformation that can be derived from the table above, the time behavior of $F(s)$ can be easily predicted once the residue is calculated. We will show some examples about partial expansion method.

Useful Links:

<http://faculty.atu.edu/mfinan/4243/Laplace.pdf> from Arkansas Tech University

<https://www.youtube.com/watch?v=D6Rd1K93nSA> from the channel of MIT openCourseWare

<https://www.youtube.com/watch?v=aX-t-xj5oaE> from the channel of Doers and Thinkers

https://www.khanacademy.org/math/algebra-home/alg-rational-expr-eq-func/alg-partial-fraction/e/partial_fraction_expansion_1 in this link you can find a tutored guide to solve exercises about partial fractions expansion and some video lectures.

EXAMPLE 1: LAPLACE DIRECT AND INVERSE TRANSFORMATION OF EXPONENTIAL FUNCTIONS

Given the following function of time:

$$f(t) = (-2 + 5e^{-2t} - 7e^{-5t})u_0(t)$$

Where we denote with $u_0(t)$ the unitary step function, defined as follows:

$$u_0(t) = \begin{cases} 1 & \text{for } t > 0 \\ \frac{1}{2} & \text{for } t = 0 \\ 0 & \text{for } t < 0 \end{cases}$$

We want to compute its Laplace transform. Since the function is a linear combination of exponentials, each term can be transformed alone, due to the linearity of the Laplace transform; we have from the tables:

$$\mathcal{L}\{-2u(t)\} = -\frac{2}{s}$$

$$\mathcal{L}\{5e^{-2t}u(t)\} = \frac{5}{s+2}$$

$$\mathcal{L}\{-7e^{-5t}u(t)\} = \frac{-7}{s+5}$$

And then:



$$\mathcal{L}\{f(t)\} = -\frac{2}{s} + \frac{5}{s+2} - \frac{7}{s+5}$$

Suppose now we have the complex variable expression:

$$F(S) = \frac{-4s^2 - 3s - 20}{s^3 + 7s^2 + 10s}$$

And we want to compute its inverse transformation; we can apply the partial fraction expansion since it is a rational fraction.

The procedure to do this is the following:

Find the poles of the ratio: the values for which the denominator gets null are

$$\begin{cases} p_1 = 0 \\ p_2 = -5 \\ p_3 = -2 \end{cases}$$

From which we can state that the ratio can be written as:

$$F(S) = \frac{r_1}{s} + \frac{r_2}{s+2} + \frac{r_3}{s+5}$$

The values of the residuals can be found multiplying the expression by the denominator to which the residual is associated and then substituting to the Laplace variable the value of the pole associated with the residual. In this case we get:

$$r_1 = [sF(s)] = s \frac{-4s^2 - 3s - 20}{s^3 + 7s^2 + 10s} = s \frac{-4s^2 - 3s - 20}{s(s+2)(s+5)} = -2 \quad \text{for } s = 0$$

$$r_2 = [(s+2)F(s)] = (s+2) \frac{-4s^2 - 3s - 20}{s^3 + 7s^2 + 10s} = (s+2) \frac{-4s^2 - 3s - 20}{s(s+2)(s+5)} = 5 \quad \text{for } s = -2$$

$$r_3 = [(s+5)F(s)] = (s+5) \frac{-4s^2 - 3s - 20}{s^3 + 7s^2 + 10s} = (s+5) \frac{-4s^2 - 3s - 20}{s(s+2)(s+5)} = -7 \quad \text{for } s = -5$$

So that our expression becomes:

$$F(S) = \frac{-2}{s} + \frac{5}{s+2} + \frac{-7}{s+5}$$

And our time domain function can be evaluated by means of the tables:

$$f(t) = (-2 + 5e^{-2t} - 7e^{-5t})u(t)$$

EXAMPLE 2

In the previous example the poles were real; if we have the following function:

$$F(s) = \frac{s}{s^2 + 4s + 13}$$

The poles of the denominator are:

$$\begin{cases} p_1 = -2 + j3 \\ p_2 = p_1^* = -2 - j3 \end{cases}$$

We have complex conjugate poles. The expression can be written as:

$$\frac{r_1}{s + 2 - j3} + \frac{r_1^*}{s + 2 + j3}$$

In this case also the residuals are complex conjugated one of the other, so we can calculate only one of them:

$$r_1 = [(s + 2 + j3)F(s)] = \frac{1}{2} + \frac{j}{3} \quad \text{for } s = -2 + j3$$

We have now the following expression:

$$F(s) = \frac{\frac{1}{2} + \frac{j}{3}}{s + 2 - j3} + \frac{\frac{1}{2} - \frac{j}{3}}{s + 2 + j3}$$

To find the time domain behavior we will need the following information:

$$|r_1| = \frac{\sqrt{13}}{6}$$

$$\text{angle}(r_1) = \varphi_1 = \arctan\left(\frac{2}{3}\right)$$

$$\alpha_1 = \text{Re}(p_1) = -2$$

$$\omega_1 = \text{Im}(p_1) = 3$$

If we proceed as we did in the previous example, where the poles and the residuals were real, we can derive the following expression:

$$f(t) = (r_1 e^{p_1 t} + r_1^* e^{p_1^* t})u(t)$$

In this case, we have complex coefficients and exponentials. By writing the precedent expression in polar form, we get:

$$\begin{aligned} f(t) &= |r_1| e^{j\varphi_1} e^{(\alpha_1 + j\omega_1)t} + |r_2| e^{j\varphi_2} e^{(\alpha_2 + j\omega_2)t} = \\ &= |r_1| e^{\alpha_1 t} (e^{j(\omega_1 t + \varphi_1)} + e^{-j(\omega_1 t + \varphi_1)}) u(t) = \\ &= 2|r_1| e^{\alpha_1 t} \cos(\omega_1 t + \varphi_1) u(t) \end{aligned}$$

Where the module and the phase of the two residuals are equal since they are complex conjugate. In our case:

$$f(t) = \frac{\sqrt{13}}{3} e^{-2t} \cos\left(3t + \arctan\left(\frac{2}{3}\right)\right) u(t)$$

We can note that the pulsation of the cosine and the coefficient of the exponential function are derived only on the poles characteristics. The residuals has no influence on these characteristics of the response.

FOURIER TRANSFORM: DEFINITION

The Fourier transform has an interpretation as a particular case of the Laplace transform. Its definition is the following:

$$F(j\omega) = \mathcal{F}\{f(t)\} = \int_{-\infty}^{\infty} f(t) e^{-j\omega t} dt$$

And:

$$f(t) = \mathcal{F}^{-1}\{F(j\omega)\} = \frac{1}{2\pi} \int_{-\infty}^{\infty} F(j\omega) e^{j\omega t} d\omega$$

This transformation can be obtained taking only the imaginary part of the Laplace variable $s = \sigma + j\omega$.

The meaning of the Fourier transform is that a function of time that respects some continuity conditions has a representation as a **superposition of infinite complex exponentials** with proper pulsation. In the real domain, it corresponds to the superposition of infinite sinusoidal functions with proper amplitude, phase and frequency. In particular, the **module** of the complex function $F(j\omega)$ represents the amplitude of the sinusoidal component for every ω , while the **phase** of $F(j\omega)$ represents the initial phase of every sinusoidal component.

The transformed function $F(j\omega)$ is the *spectrum* of the function $f(t)$. Since the combination of the sinusoidal components **is linear**, the Fourier analysis is useful to decompose a problem in sub-problems for which the solution is known.

Useful links:

- <https://www.youtube.com/watch?v=gZNm7L96pfY> Lectures about Fourier transform by Stanford University
- <https://www.youtube.com/watch?v=1JnayXHhjlq> An introduction about Fourier transform by Brian Douglas
- <https://ocw.mit.edu/courses/mathematics/18-03sc-differential-equations-fall-2011/unit-iii-fourier-series-and-laplace-transform/fourier-series-basics/> Useful material by MIT openCourseWare

PROPERTIES OF THE FOURIER TRANSFORM

The main properties of the Fourier transform are the following:

Function, $f(t)$	Fourier Transform, $F(\omega)$
<i>Definition of Inverse Fourier Transform</i>	<i>Definition of Fourier Transform</i>
$f(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} F(\omega) e^{j\omega t} d\omega$	$F(\omega) = \int_{-\infty}^{\infty} f(t) e^{-j\omega t} dt$
$f(t - t_0)$	$F(\omega) e^{-j\omega t_0}$
$f(t) e^{j\omega_0 t}$	$F(\omega - \omega_0)$
$f(\alpha t)$	$\frac{1}{ \alpha } F\left(\frac{\omega}{\alpha}\right)$
$F(t)$	$2\pi f(-\omega)$
$\frac{d^n f(t)}{dt^n}$	$(j\omega)^n F(\omega)$
$(-jt)^n f(t)$	$\frac{d^n F(\omega)}{d\omega^n}$
$\int_{-\infty}^t f(\tau) d\tau$	$\frac{F(\omega)}{j\omega} + \pi F(0) \delta(\omega)$
$\delta(t)$	1
$e^{j\omega_0 t}$	$2\pi \delta(\omega - \omega_0)$
$\text{sgn}(t)$	$\frac{2}{j\omega}$

And are related to the corresponding Laplace properties.

In the field of Automatic control, the inverse transform of the Fourier transform is not a common practice; in fact the spectral analysis is made to evaluate globally the frequency content of a signal before and after the transition through a LTI system, or to evaluate the amplitude and the phase of an output single frequency.

Useful links:

- <https://www.youtube.com/watch?v=D1WF9YKqf3o> About Fourier's transform properties

Z-TRANSFORM

The Z-transform is the main mathematical tool that is used in the field of **digital control**; it is used when we have to manage discrete time signals, or when we want to implement a digital controller. In its standard expression, given a sequence of samples $x[n]$, the Z-Transform works as follows:

$$X(z) = Z\{x[n]\} = \sum_{-\infty}^{\infty} x[n]z^{-n}$$

Or:

$$X(z) = Z\{x[n]\} = \sum_0^{\infty} x[n]z^{-n}$$

Used in practice to treat finite length sequences.

The z variable is a complex variable. In particular if we apply the substitution $z = e^{sT}$ where s is the Laplace variable and T is the sampling time of the discrete signal (that we will always consider constant), we get the Laplace transform of the discrete sequence of samples. This fact indicates how the z variable plane is obtained by a **transformation of the Laplace complex plane**.

The formal definition of the inverse Z-transform is rarely used in practice, since in LTI systems the expressions that we manage involving the z variable are rational fractions, that can be easily anti-transformed

PROPERTIES OF Z TRANSFORM AND COMMONLY USED TRANSFORMATIONS

Property / Theorem	Time Domain	Z transform
Linearity	$af_1[n] + bf_2[n] + \dots$	$aF_1(z) + bF_2(z) + \dots$
Shift of $x[n]u_0[n]$	$f[n-m]u_0[n-m]$	$z^{-m}F(z)$
Right Shift	$f[n-m]$	$z^{-m}F(z) + \sum_{n=0}^{m-1} f[n-m]z^{-n}$
Left Shift	$f[n+m]$	$z^mF(z) + \sum_{n=-m}^{-1} f[n+m]z^{-n}$
Multiplication by a^n	$a^n f[n]$	$F\left(\frac{z}{a}\right)$
Multiplication by e^{-naT}	$e^{-naT} f[n]$	$F(e^{aT}z)$
Multiplication by n	$nf[n]$	$-z \frac{d}{dz} F(z)$
Multiplication by n^2	$n^2 f[n]$	$z \frac{d}{dz} F(z) + z^2 \frac{d^2}{dz^2} F(z)$
Summation in Time	$\sum_{m=0}^n f[m]$	$\left(\frac{z}{z-1}\right)F(z)$
Time Convolution	$f_1[n] * f_2[n]$	$F_1(z) \cdot F_2(z)$
Frequency Convolution	$f_1[n] \cdot f_2[n]$	$\frac{1}{j2\pi} \oint x F_1(v) F_2\left(\frac{z}{v}\right) v^{-1} dv$
Initial Value Theorem	$f[0] = \lim_{z \rightarrow \infty} F(z)$	
Final Value Theorem	$\lim_{n \rightarrow \infty} f[n] = \lim_{z \rightarrow 1} (z-1)F(z)$	

A list of the main properties of the Z transform is shown in the table; the most important fact is that the z variable represents a **time-shift operator**; the power of z indicates how many steps of lead or lag the relative sample in the time suffers; for example:

$$\begin{aligned}
 Y(z) &\rightarrow y[n] \\
 Y(z)z^{-1} &\rightarrow y[n-1] \\
 Y(z)z^{-2} &\rightarrow y[n-2] \\
 Y(z)z &\rightarrow y[n+1]
 \end{aligned}$$

...

This property is useful to obtain time domain **differences equations** that represent the behavior of the system in time.

A list of the most common z -transforms related to the Laplace transforms follows:

Laplace transform	Corresponding z-transform
$\frac{1}{s}$	$\frac{z}{z-1}$
$\frac{1}{s^2}$	$\frac{Tz}{(z-1)^2}$
$\frac{1}{s^3}$	$\frac{T^2z(z+1)}{2(z-1)^3}$
$\frac{1}{s+a}$	$\frac{z}{z-e^{-aT}}$
$\frac{1}{(s+a)^2}$	$\frac{Tze^{-aT}}{(z-e^{-aT})^2}$
$\frac{a}{s(s+a)}$	$\frac{z(1-e^{-aT})}{(z-1)(z-e^{-aT})}$
$\frac{b-a}{(s+a)(s+b)}$	$\frac{z(e^{-aT}-e^{-bT})}{(z-e^{-aT})(z-e^{-bT})}$
$\frac{(b-a)s}{(s+a)(s+b)}$	$\frac{(b-a)z^2 - (be^{-aT} - ae^{-bT})z}{(z-e^{-aT})(z-e^{-bT})}$
$\frac{a}{s^2+a^2}$	$\frac{z \sin aT}{z^2 - 2z \cos aT + 1}$
$\frac{s}{s^2+a^2}$	$\frac{z^2 - z \cos aT}{z^2 - 2z \cos aT + 1}$
$\frac{s}{(s+a)^2}$	$\frac{z[z - e^{-aT}(1+aT)]}{(z-e^{-aT})^2}$

Where T is the sampling time at which the continuous time process is monitored.

HOW TO OBTAIN THE Z TRANSFORM

If we are dealing with sequences of samples, the Z transform can be easily calculated; anyway, often, **we start from a continuous time** physical description, suitable to be studied in the Laplace variable domain. It is possible to pass from the Laplace expression to the Z expression of the behavior of the system by fixing a sampling interval T_s for the continuous time data time. Is also possible to do the inverse operation.

In addition, the Z transform can be obtained after a system identification procedure, that is the estimation of the characteristics of the system by means of input/output pairs of samples.

EXAMPLE: Z-TRANSFORM FROM A LAPLACE EXPRESSION

Suppose we are given with the previous expression in the s domain:

$$F(s) = \frac{s}{s^2 + 4s + 13}$$

This expression has its continuous time domain transformation; if we want to consider only discrete time samples, we can use the Z-transform, obtaining it directly from the s domain function. Once one fixes the sampling time, the equivalent expression in Z-domain can be obtained by use of the tables

or, most often, by automated algorithms; for example in Matlab, the command *c2d* (continuous to discrete), given the sampling time $T_s = 0.001$, returns:

$$F(z) = \frac{0.000998z - 0.000998}{z^2 - 1.996z + 0.996}$$

If we change the sampling time, the expression changes consequently; with $T = 0.01$

$$F(z) = \frac{0.009801z - 0.009801}{z^2 - 1.96z + 0.9608}$$

RELATED MATLAB COMMANDS

A list of useful Matlab functions regarding integral transforms follows; notice that from the command window one can type '**help**' followed by the command name to have almost all the information about that function, in particular about outputs and necessary input arguments.

- *tf* (...) is able to define an output variable of the type 'tf'. It is useful to define a Laplace variable 's'
- *laplace* (...) calculates the laplace transform of a symbolic expression
- *fourier* (...) calculates the fourier transform of a symbolic expression
- *ztrans* (...) calculates the Z- transform of a symbolic expression
- *residue* (...) is used both to convert a polynomial into partial fraction form or vice versa

MODELLING OF DYNAMICAL SYSTEMS

GENERALITIES ON DYNAMICAL SYSTEMS

Automatic Control is the branch of the System Theory that studies the methodologies to be applied when we want a particular dynamic system to respond in a certain way, either when it is free to evolve or when it undergoes exogenous inputs.

Such a definition reflects one of the most powerful aspects of the subject, that is, its **generality**. We speak about dynamical system when the evolution of such a system is determined not only by the present input, but also by its previous history: the system reacts in different ways to the inputs with reference to its '**state**'.

On the contrary, a static system exhibits an instantaneous reaction to the input independently on its past.

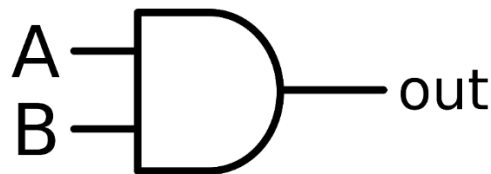
A practical example of a dynamical system is the process of the room heating; we all experienced that the reach of the desired temperature is a long-term process that evolves in function of the room temperature and of the surrounding conditions; a counterexample of static system is the turn on of a light bulb, that is (almost) instantaneous.

These two phenomena should suggest us that we could describe a static system with an algebraic language, while we cannot describe a dynamical system without seek help of differential equations

or differences equations. Indeed, all the descriptions of dynamical system are based upon differential (difference) equations, and it should be clear, since as we cannot say which the solution of a differential equation is without knowing the initial conditions, we cannot describe a system dynamic without knowing its initial state. The initial conditions (surrounding conditions) can be generally thought as the system 'state' even if, as we will say later, the concept of state is more abstract. Even the concept of 'output' of a system is quite general: a system can have a spread of possible outputs we don't care about. In the room heating example, we generally think at the room temperature as the classic output, but we would be interested in other quantities that changes during the process (humidity is just an intuitive example) and in that case, those would become our output. We could be interested to multiple outputs as well. In most of cases, our output of interest can be derived once we know the state of the system and the inputs.

EXAMPLE: STATIC AND DYNAMIC SYSTEMS.

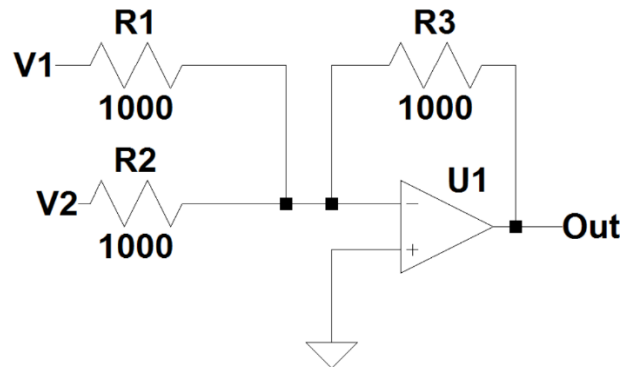
We now show the difference between a static and dynamic system.
Consider the logic component AND gate:



The symbol represents a component that exhibits the following behavior:

A	B	out
1	1	1
1	0	0
0	1	0
0	0	0

This table imposes an instantaneous relationship between the output and the two inputs. There is no reference to the history of the gate: when the input changes, the output changes at the same time. Another example of static system is the summing amplifier:

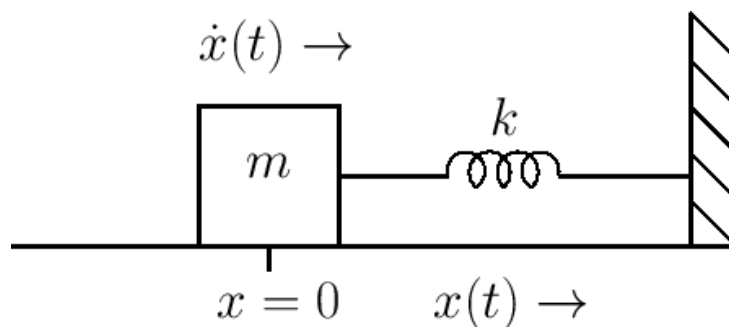


This circuit gives as output:

$$Out = -(V_1 + V_2)$$

Also in this case, the relationship between the inputs and the output is algebraically described and there's no memory in the system.

A classic example of a dynamical system is the mass-spring system:



The second law of motion leads to the following differential equation:

$$m\ddot{x} = -kx$$

This is a second order linear differential equation with constant coefficients; the evolution of the system cannot be described in closed form without the knowledge of the two initial conditions:

$$\begin{cases} x(0) = x_0 \\ \dot{x}(0) = v_0 \end{cases}$$

This fact imposes a dependency of the system from its past, and introduces the concept of state of the system.

All the three examples we showed are time-invariant: the input-output relationship of the static system doesn't change in time; the differential equation of the mass-spring system has constant coefficients. Note that neither the gate or the operational amplifier are actually instantaneous in their response: in our description we simply neglect their dynamics due to their speed. In some context, this kind of description would not be sufficient, so even this two simple system would become dynamical systems.

MATHEMATICAL DESCRIPTION AND PROPERTIES

A general mathematical formulation follows.

For a static system, we have:

$$\mathbf{y}(t) = f(\mathbf{u}(t)) \text{ for continuous time static system}$$

$$\mathbf{y}(k) = f(\mathbf{u}(k)) \text{ for discrete time dynamic system}$$

For a dynamical system:

$$\mathbf{y}(t) = f(\mathbf{u}(t), \mathbf{x}(t), t) \text{ for continuous time dynamic systems}$$

$$\mathbf{y}(k) = f(\mathbf{u}(k), \mathbf{x}(k), t) \text{ for discrete time dynamic systems}$$

Where we denoted the outputs with $\mathbf{y}(t)$, the inputs with $\mathbf{u}(t)$ and the states with $\mathbf{x}(t)$ (and relative discrete sequences for discrete time); in the general formulation, they are vectors.; t is the time. We distinguish the systems on the base of the number of their inputs and their outputs in:

- SISO (single input-single output): $\mathbf{y}(t), \mathbf{u}(t) \in \mathbb{R}$
- SIMO(single input-multiple output): $\mathbf{y}(t) \in \mathbb{R}^n, \mathbf{u}(t) \in \mathbb{R}$
- MISO(multiple input-single output): $\mathbf{y}(t) \in \mathbb{R}, \mathbf{u}(t) \in \mathbb{R}^n$
- MIMO(multiple input-multiple output): $\mathbf{y}(t) \in \mathbb{R}^n, \mathbf{u}(t) \in \mathbb{R}^m$

With obvious discrete time parallel.

We now focus on the time variable t that appears in the functional written above; this explicit dependence is there to denote that a system can change its behavior in time, for example, due to the variation of particular parameters that are time variant or due to other causes. A system that does not change its behavior in time is called **Time-Invariant**, that is to say, its response doesn't depend on absolute time: to stimulate this system now or tomorrow in the same way will return the same results. Another fundamental distinction for dynamical systems is based upon their **linearity**: a system is linear when all the relations among the quantities (inputs, outputs, states) are linear, that is when the differential equations that hold the system are linear. An interesting fact about linear differential equations is that a higher order linear differential equation can be arranged in a **system** of first order differential equation (which solution can be found easily). The linear systems theory is more complete and predictive than the non-linear systems one.

In the following, we will always deal with **causal** dynamic systems, for which the output is determined only by the past and the present values of the quantities that regulates the system (input, states).

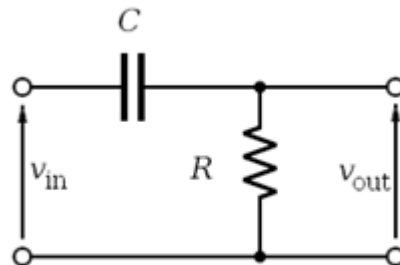
Useful links:

-“System Dynamics”, Dean C. Karnopp, Donald L. Margiolis, Ronald C. Rosenberg, John Wiley & Sons Inc.

-“Introduction to the Mathematical Theory of Systems and Control”, Jan Willem Polderman, Jan C. Willems

http://mathinsight.org/dynamical_system_idea (From Math insight)

EXAMPLE: ELECTRICAL NETWORK EXAMPLE



We analyze now the charge of a capacitor.

First, we can pinpoint inputs and output of this system; the figure suggests us that the output of the circuit is the voltage across the resistor, and we will proceed according to this choice, but we could be interested as well to the voltage across the capacitor, or to both the voltages. This means that this system can be a multiple output or a single output one, according to our will. The input is labeled as V_{in} , and that is the voltage that we impose to the network.

We chose to monitor only the Voltage across the resistor, even if we would be monitoring also the voltage across the capacitor (and then we would had a multiple output system).

To solve this system we must use three equations from the circuit theory:

$$\begin{cases} V_R = V_{out} = RI \\ I_c = C \frac{dV_c}{dt} \\ \sum_i^3 V_i = 0 \end{cases}$$

These represent the two constitutive law of the components and the Kirchhoff law for the voltages. Since the current flowing into the two components is the same, we can use the equations to write:

$$\begin{aligned} V_{in} &= V_C + V_R \\ V_{in} &= V_C + RC \frac{dV_C}{dt} \end{aligned}$$

We know from the Analysis that this equation has a unique solution and that, given an initial condition, we can fix this solution. It is not our goal to reach this solution now, but to emphasize the structure of this differential equation; it shows the following form:

$$\dot{y} + ay = g(t)$$

This equation is a linear differential equation, since the function to be found and its derivative are present in the form of a polynomial of the first order in every term.

We now focus on the other quality that we defined for a system, which is the *time-invariance*. We note that the system is completely defined by the differential equation above; the only things that can

change the properties of the solution are the resistance and the capacitance, two parameters. We assumed that these parameters are constant, so the solution of the differential equation will be the same always, given the same initial condition. This assumption could not be verisimilar; the parameters of the system could change in time, and in most of cases, they do. In this case the system is not time invariant. Even the simple example above is physically time variant, since the components undergo to the usage wear and change their characteristics in time; anyway, the process is slow enough for most of the application, so we can think at the parameters as constant.

Application of superposition principle (linearity):

Return to the simple electrical network example; suppose that the input of the circuit is:

$$V_{in(t)} = V_1(t) + V_2(t)$$

Suppose that we know the response of the system to the inputs $V_1(t)$ and $V_2(t)$ separately, but we don't know the response to $V_{in}(t)$.

The superposition principle assures us that:

$$V_{Ctot} = V_{C(V_1)} + V_{C(V_2)}$$

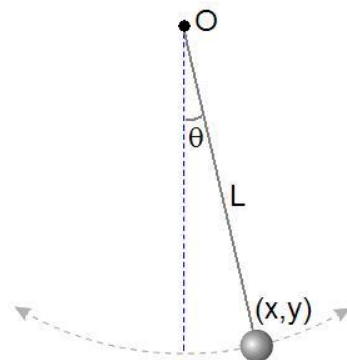
Where the two terms satisfies the differential equation:

$$V_1 = V_{C(V_1)} + RC \frac{dV_{C(V_1)}}{dt}$$

$$V_2 = V_{C(V_2)} + RC \frac{dV_{C(V_2)}}{dt}$$

EXAMPLE: THE PENDULUM

We now expose a second famous example, the simple pendulum. This example gives an idea of how important is linearity for our purposes.



The governing equation for the pendulum dynamics is the Newton's law. We can chose a reference system attached to the mass M, such that a principal axe is oriented as the wire (normal axe), the other perpendicular to the wire (tangent axe). With this assumption, the two equations for the forces equilibrium are

$$\sum F_n = 0$$

$$\sum F_t = 0$$

The normal forces that act on the system are the normal gravity component and the wire reaction T .

$$T - Mg \cos \theta = 0$$

Where the normal acceleration is set to zero, since the mass never changes its position with respect to the center O . The tangential forces, in absence of friction are the mass inertia and the gravitational component.

$$Mg \sin \theta + Ma_t = 0$$

This last equation is the one we need to foresee the system dynamic. We write:

$$a_t = g \sin \theta$$

We now know that the amplitude of the arc fixed by the mass position and the vertical is

$$x = L\theta$$

Therefore, we can turn back to the O reference system and write:

$$\begin{cases} x = L\theta \\ \dot{x} = L\dot{\theta} \\ \ddot{x} = L\ddot{\theta} \end{cases}$$

Where our equation becomes:

$$L\ddot{\theta} = g \sin \theta$$

So that the final differential equation for the pendulum angle becomes:

$$\ddot{\theta} = \frac{g}{L} \sin \theta$$

Looking at this equation, we notice that it is nonlinear: the unknown function $\theta(t)$ and its derivatives do not appear only in polynomial form. The primitive function $\theta(t)$ is argument for the sine function. This makes the pendulum equation much more difficult to be solved than a linear one. For this reason, we linearize it. In this case, we can make a strong hypothesis about the pendulum behavior, bounding its evolution to a small angle of deviation from the vertical. We so operate a ‘small angle approximation’ for the sine function:

$$\sin \theta \simeq \theta \quad \theta \simeq 0$$

Getting to the linear equation:

$$\theta'' = \frac{g}{L} \theta$$

That has got a sinusoidal solution for all three the quantities, position, velocity, acceleration.

The **linearization procedure** is fundamental for an immediate study of the phenomenon and its control; anyway, it cuts off a large part of the system dynamics, which is intrinsically nonlinear. Another big advantage of linearization is the possibility to decompose a higher order linear

differential equation in to a system of first order differential equations. In our case, we can impose a variable substitution and get:

$$x_1 = \theta$$

$$x_2 = \dot{\theta}$$

From which we decompose the second order differential equation into a system of first order equations:

$$\begin{cases} \dot{x}_2 = \frac{g}{L} x_1 \\ \dot{x}_1 = x_2 \end{cases}$$

With this decomposition, we get a differential *system*. The two variables x_1, x_2 are the *states* of the physical system and characterize its dynamic. They are two, since the DE is of the second order. This is the link between the system states and the DE that holds the phenomenon.

Notice that the substitution that we made is arbitrary. We chose position and velocity as the system states, but we could have chosen acceleration to be a state, being able to get the values of the third quantity. This emphasizes the fact that the **minimum number** of the states of a dynamical system is imposed by the physics, but we can use any set of states to describe the phenomenon properly.

STATE SPACE DESCRIPTION

STATE SPACE DESCRIPTION FOR LINEAR SYSTEMS IN CONTINUOUS TIME

We now focus on the mathematical modelling of linear systems.

Suppose that the functional:

$$y(t) = f(u(t), x(t), t)$$

Can be described by means of linear differential equations; then, a **State-Space representation** of the following form is always possible:

$$\begin{cases} \dot{x} = A(t)x + B(t)u \\ y = C(t)x + D(t)u \end{cases}$$

Where the terms of the system are vector **functions of time** with the following meanings:

- $x \in \mathbb{R}^n$ Is the vector of the n states that are present in the system
- $\dot{x} \in \mathbb{R}^n$ Is the vector of the derivatives of the n states
- $u \in \mathbb{R}^p$ Is the input vector
- $y \in \mathbb{R}^q$ Is the output vector
- $A(t) \in \mathbb{R}^{n \times n}$ is the **state matrix**: it relates the value of the states to the value of their derivatives

- $B(t) \in \mathbb{R}^{n \times p}$ is the **input matrix**: it relates the value of the input to the value of the derivative of the states
- $C(t) \in \mathbb{R}^{q \times n}$ is the **output matrix**: links the value of the state to the value of the output of interest
- $D(t) \in \mathbb{R}^{p \times q}$ is the **feedforward matrix**: is the direct link from the input to the output

In the above formulation, the four system matrices are functions of time; this is because the functional exhibits an explicit time-dependency, typical of time-variant systems. If the functional that describes the systems can be written as:

$$\mathbf{y}(t) = f(\mathbf{u}(t), \mathbf{x}(t))$$

Where $f(\cdot)$ is a linear vector function, then the state-space representation is:

$$\begin{cases} \dot{\mathbf{x}} = A\mathbf{x} + B\mathbf{u} \\ \mathbf{y} = C\mathbf{x} + D\mathbf{u} \end{cases}$$

With constant matrices A, B, C, D is always possible. This last representation is the most common for systems that are **linear and time-invariant**. Such systems are referred to as LTI systems.

Useful links:

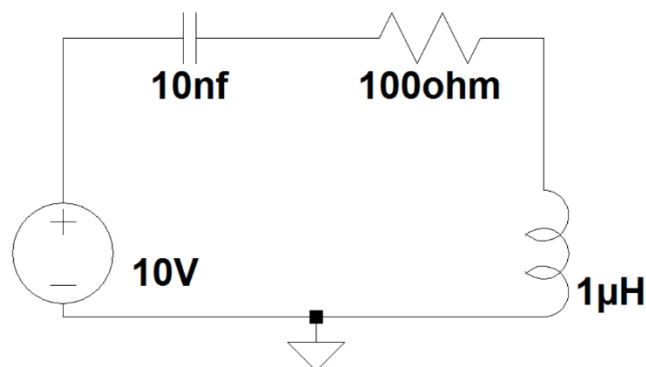
<http://web.mit.edu/2.14/www/Handouts/StateSpace.pdf> From MIT website

“Modern control systems” ninth edition, R.C. Dorf, R.H. Bishop, Prentice Hall (first chapters)

“System dynamics” K. Ogata, Prentice Hall (chapter 10)

EXAMPLE: BUILDING THE STATE SPACE REPRESENTATION OF A SYSTEM

The procedure to build the state space representation of a system is analogous to the decomposition of a higher order linear differential equation to a system of first order differential equations, once the states, the input and the outputs are defined. The first thing to do is to fix the states of the system, that are those quantities that are related to the others by means of an integral relation and thus take into account the memory of the system. Once the states are defined, their variation must be expressed in function of the input and of the states themselves. Consider the following RLC circuit:



The two components that adds a state to the system are the inductor and the capacitor, for which the following relations must be satisfied:

$$v_L = L \frac{di_L}{dt}$$
$$i_C = C \frac{dv_C}{dt}$$

From these, we understand that the two states of the system are the voltage across the capacitor and the current through the inductor, since their instantaneous value is find through the integration of the other electrical variable:

$$\mathbf{x} = \begin{pmatrix} i_L \\ v_C \end{pmatrix}$$

Where the time dependency is implied.

The input of the system in this case is the voltage source V_G .

We must express the variation of these quantities as a linear combination of themselves and of the inputs. To begin we can invert the relations:

$$\frac{di_L}{dt} = \frac{v_L}{L}$$
$$\frac{dv_C}{dt} = \frac{i_C}{C}$$

Now, the voltage across the inductor and the current through the capacitor must be expressed in terms of the states and the input.

From the Kirchhoff voltage law we can state:

$$v_L = V_G - v_C - V_R$$

The only term that must be further changed is the voltage across the resistor, since it's not a state. We express it as:

$$V_R = i_L R$$

And get:

$$v_L = V_G - v_C - i_L R$$

For the second equation, the things are easier since the current through the capacitor is the same as the current through the inductor, a state variable; this leads us to:

$$\begin{cases} \frac{di_L}{dt} = \frac{V_G - v_C - i_L R}{L} \\ \frac{dv_C}{dt} = \frac{i_L}{C} \end{cases}$$

From the system above we must, by inspection, define the A, B, C, D matrices; it is easy to find out that:

$$A = \begin{pmatrix} -\frac{R}{L} & -\frac{1}{L} \\ \frac{1}{C} & 0 \end{pmatrix}, B = \begin{pmatrix} \frac{1}{L} \\ 0 \end{pmatrix}$$

The above system can now be written by means of the vector equation:

$$\dot{\mathbf{x}} = A\mathbf{x} + Bu$$

To complete the state space we must define the output of interest. Suppose we want to know the voltage across the capacitor; then the output will be a state and the C matrix will be:

$$C = (0 \quad 1) \rightarrow y = C\mathbf{x} = v_c$$

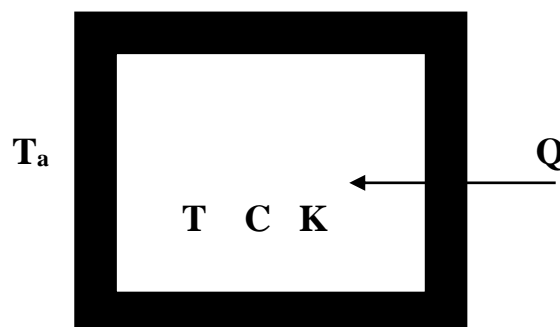
If we were interested to the inductor voltage, we had to define also the D matrix since:

$$v_L = V_G - v_c - i_L R \rightarrow C = (-R \quad -1) ; \quad D = 1 \rightarrow y = C\mathbf{x} + Du$$

If we were interested to both the states then we would have a SIMO system with the vector y of the output:

$$\mathbf{y} = C\mathbf{x} ; \quad C = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

EXAMPLE: FURNACE HEATING MODELLING



Consider the above schematization of a closed environment having thermal capacity C , homogeneous temperature T , Embedded in a box with thermic conductance K . The outer temperature is denoted with T_a .

The thermal balance for the system is:

$$C\dot{T} + K(T - T_a) = Q$$

If we are interested in the temperature T inside the furnace, it will be our output. We choose as input variables the heat flow Q and the outer temperature T_a . The state variable can be naturally defined as T , to get the following relation between the state and its derivative:

$$\dot{x} = -\frac{K}{C}x + \frac{K}{C}T_a + \frac{Q}{C}$$

From this, we can build the state space representation:

$$\begin{aligned}\dot{x} &= -\frac{K}{C}x + \begin{pmatrix} \frac{1}{C} & 0 \\ 0 & \frac{K}{C} \end{pmatrix} u \\ \mathbf{y} &= x\end{aligned}$$

Therefore, our matrices will be:

$$A = 1; \quad B = \begin{pmatrix} \frac{1}{C} & 0 \\ 0 & \frac{K}{C} \end{pmatrix}; \quad C = 1; \quad D = 0$$

STATE SPACE DESCRIPTION FOR LINEAR SYSTEMS IN DISCRETE TIME

When we manage with a linear, time-invariant discrete time system, the state space representation takes the form:

$$\begin{cases} \mathbf{x}(k+1) = A\mathbf{x}(k) + B\mathbf{u}(k) \\ \mathbf{y}(k) = C\mathbf{x}(k) + D\mathbf{u}(k) \end{cases}$$

The same definitions given for continuous linear systems hold. This representation is particularly suitable for the simulation and control purposes, since the evolution of the state and outputs sequences can be computed directly by means of matrix products.

Useful links:

-K.J. Astrom, B.Wittenmark, 'Computer-controlled Systems', Prentice-Hall

IMPULSE RESPONSE AND TRANSFER FUNCTIONS

IMPULSE RESPONSE AND CONVOLUTION

Another way to describe the behavior of a dynamical linear time-invariant (LTI) system is its **impulse response**. It describes how the system reacts to the Dirac Delta. Since the system that we are describing is linear and time invariant, every possible output can be calculated by summing all the (impulse) responses generated by any possible input (every input signal can be decomposed in an

infinite sequence of Dirac delta).

The impulse response **completely characterizes** the response of a dynamical system.

The operation by which the output to any arbitrary input signal is computed is called **convolution**, or integral sum.

We denote with $h(t)$ the function that describes the impulse response of the system and with $u(t)$ the input signal; then the output of the system is computed through the formula:

$$y(t) = h(t) * u(t) = \int_{-\infty}^{+\infty} u(\tau)h(t - \tau)d\tau$$

The convolution sums together all the impulse responses produced by the input, properly **translated** and **amplified** by the amplitude of the input itself in every instant of time (due to the properties of linearity and time invariance).

Unfortunately, the computation of the convolution is often very heavy, so that it is convenient to work in the Laplace domain, where this operation is reduced to a **simple product of complex functions**.

The impulse response is defined also in the **discrete time** domain by means of a proper sequence $h(k)$, while the convolution in the discrete is expressed by the following sum:

$$y(k) = h(k) * u(k) = \sum_{n=-\infty}^{\infty} u(n)h(k - n)$$

The above discussion is related to a SISO system; if the system is characterized by multiple inputs or multiple outputs, then the impulse response becomes a **matrix** of functions of proper dimension.

Useful links:

More on convolution and impulse response

<http://mathworld.wolfram.com/Convolution.html> from MathWorld

<http://www.dspguide.com/ch6/2.htm>

“Unified signal theory”, G. Cariolaro, Springer, 2011

CONTINUOUS TIME TRANSFER FUNCTIONS

For continuous time systems, a transfer function is the expression in the **Laplace domain** of the impulse response that characterizes a LTI system. It allows to calculate the output of a system in the complex domain, since its form is the following:

$$H(s) = \frac{Y(s)}{U(s)}$$

The $H(s)$ is always a rational function of s for a LTI system. The degree of $Y(s)$ is always greater or equal to the degree of $U(s)$ **for a causal system**. We can write:

$$Y(s) = H(s)U(s)$$

If we apply the inverse Laplace transform to this last expression we get:

$$\mathcal{L}^{-1}\{Y(s)\} = y(t) = \mathcal{L}^{-1}\{H(s)\} * u(t)$$

We notice that the time domain expression of $H(s)$ coincides with the impulse response $h(t)$, since the output of the system is calculated by means of the convolution between $h(t)$ itself and the input $u(t)$.

For this reason the transfer function is the **Laplace transform of the impulse response**.

There are several ways to obtain the transfer function of a system:

- 1) By direct transformation of the impulse response.
- 2) From the differential formulation of the system description
- 3) By an identification procedure that through experimentally collected data reconstructs the relationship that bonds input and outputs (this is the most used procedure in practice)

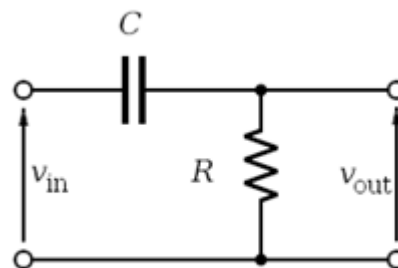
Useful links:

- <https://ocw.mit.edu/courses/mathematics/18-03sc-differential-equations-fall-2011/unit-iii-fourier-series-and-laplace-transform/transfer-system-and-weight-functions-greens-formula> From the MIT openCourseWare

- <https://ocw.mit.edu/courses/mathematics/18-03sc-differential-equations-fall-2011/unit-ii-second-order-constant-coefficient-linear-equations/frequency-response-and-practical-resonance> From the MIT openCourseWare

EXAMPLE: TRANSFER FUNCTION OF THE RC CIRCUIT

We now take again into account the RC circuit to find its transfer function:



We found that the governing equation of the circuit is the following:

$$v_{in} = v_c + RC \frac{dv_c}{dt}$$

If we operate the Laplace transform to the quantities of the equation, all time-dependent, we find:

$$V_{in}(s) = V_c(s) + (RC)V_c(s)s - v_c(0^-)$$

Setting the initial condition to zero we can obtain:

$$V_{in}(s) = V_C(s)(1 + RCs)$$

The transfer function from the input to the voltage across the capacitor is:

$$\frac{V_C(s)}{V_{in}(s)} = \frac{1}{(1 + RCs)}$$

From this, we can also find the voltage across the resistor since:

$$v_{in} = v_C + v_R$$

That in the Laplace domain becomes:

$$\begin{aligned} V_{in}(s) &= V_C(s) + V_R(s) \\ V_R(s) &= V_{in}(s) - V_C(s) \\ V_R(s) &= V_{in}(s) \left(1 - \frac{1}{(1 + RCs)} \right) \end{aligned}$$

And then:

$$\frac{V_R(s)}{V_{in}(s)} = \left(\frac{RCs}{1 + RCs} \right)$$

Suppose now that we want to evaluate the response of the system to a step input of amplitude α ; in particular we want to evaluate the behavior of the voltage over the capacitor.

In the Laplace domain we simply multiply the step by the transfer function and operate the inverse Laplace transform:

$$V_C(s) = \frac{V_C(s)}{V_{in}(s)} U(s) = \frac{1}{RC \left(\frac{1}{RC} + s \right)} \frac{\alpha}{s}$$

We can now apply the partial fraction expansion to get the time domain behavior. The poles of the system:

$$\begin{cases} p_1 = 0 \\ p_2 = -\frac{1}{RC} \end{cases}$$

$$V_C(s) = \frac{r_1}{s + \frac{1}{RC}} + \frac{r_2}{s}$$

The value of the two residuals are:

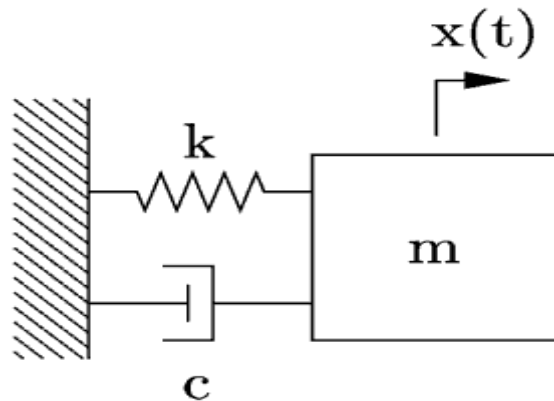
$$\begin{cases} r_1 = -\alpha \\ r_2 = \alpha \end{cases}$$

The time domain expression becomes:

$$f(t) = \alpha \left(-e^{-\left(\frac{t}{RC}\right)} + 1 \right) u(t)$$

That is the classical behavior of the capacitor charge.

EXAMPLE: SPRING MASS DAMPER SYSTEM



A classic example of a second order system is the damped mass-spring oscillator. The equilibrium equation for the system is the following:

$$m\ddot{x} + c\dot{x} + kx = g(t)$$

Where $g(t)$ is an input force and we are interested in the output $x(t)$. Now, if we impose $g(t) = \delta(t)$, an impulsive force as input, and all the initial conditions to zero, we can calculate the transfer function of the system:

$$\begin{aligned} mX(s)s^2 + cX(s)s + kX(s) &= G(s) \\ X(s)(ms^2 + cs + k) &= G(s) \\ F(s) = \frac{X(s)}{G(s)} &= \frac{1}{(ms^2 + cs + k)} = \frac{1}{\left(s^2 + \frac{c}{m}s + \frac{k}{m}\right)} \end{aligned}$$

Suppose we now apply an unitary sinusoidal signal as input to the system; the behavior of the system can be derived computing the inverse Laplace transform of the following expression:

$$X(s) = \frac{1}{\left(s^2 + \frac{c}{m}s + \frac{k}{m}\right)} \frac{\omega}{s^2 + \omega^2}$$

The behavior of the system in this case depends on the value of the parameters and can be found with the partial fraction expansion.

If we use the parameters:

$$\begin{cases} m = 10 \\ c = 10 \\ k = 5 \end{cases}$$

We get the transfer function:

$$X(s) = \frac{1}{s^2 + s + 0.5}$$

TRANSFER FUNCTIONS FORMULATIONS

We introduced a transfer function in the form of a rational function of s ; for these kind of functions the factorization of numerator and denominator results in the pole-zero form of the transfer function:

$$H(s) = \frac{\prod_{i=0}^Z (s - z_i)}{\prod_{i=0}^P (s - p_i)}$$

An alternative formulation, which highlights the value of the transfer function for $s = 0$ is the DC-Gain form of the transfer function, of the form:

$$H(s) = K \frac{\prod_{i=0}^Z (s/z_i - 1)}{\prod_{i=0}^P (s/p_i - 1)}$$

with particular attention to the denominator, assuming that the transfer functions are causal, we define two different important forms of the transfer functions, that will be deeper investigated in the following sections:

- First order transfer function: $\frac{N(s)}{(\tau s + 1)}$ the coefficient τ is called time constant and gives an indication of how fast a system reacts to the input; in particular, we know from the partial fraction expansion, that the time constant will characterize the associated time domain evolution of the system; the bigger the time constant, the slower the exponential growth (decay) of the system.
- Second order transfer function: $\frac{N(s)}{(s^2 + 2\omega\zeta s + \omega^2)}$ in the case of a second order transfer function, we define two different parameters: the natural frequency ω is the absolute value of each complex conjugate poles that generate the second order denominator, while the damping ratio ζ is an indication of how much the complex poles are ‘far’ from the imaginary axis. In fact, the angle θ between the poles and the real axis is equal to: $\theta = \cos^{-1} \zeta$. In terms of the quality of the system’s response, the natural frequency indicates at which frequency the system oscillates when it is free to evolve, while the damping ration is a measure of the attitude of the system to dissipate energy. A low damping factor means that the system dissipates very few energy for every oscillation and for this reason, it is able to oscillate for long time (theoretically forever if $\zeta = 0$) without external inputs.



Useful links:

About time constants:

<https://www.facstaff.bucknell.edu/mastascu/eLessonsHTML/SysDyn/SysDyn3TCBasic.htm>

About second order systems :

<https://www.facstaff.bucknell.edu/mastascu/eControlHTML/SysDyn/SysDyn2.html>

<http://www.ece.rutgers.edu/~gajic/psfiles/chap6.pdf> from Rutgers University

most of the previously suggested textbooks treats the argument

DISCRETE TIME TRANSFER FUNCTION

A discrete time transfer function is the Z-transform of a discrete time impulse response; it takes the form:

$$H(z) = \frac{Y(z)}{X(z)}$$

The $H(z)$ is always a rational function of z for a LTI system. The degree of $Y(s)$ is always greater or equal to the degree of $X(s)$ for a causal system. We can write:

$$Y(z) = H(z)X(z)$$

A discrete time transfer function can be calculated starting from a continuous time transfer function once a sampling time T is fixed. An example is given in the following.

EXAMPLE: FROM CONTINUOUS TIME TRANSFER FUNCTION TO DISCRETE TIME TRANSFER FUNCTION

Consider again the transfer function:

$$X(s) = \frac{1}{s^2 + s + 0.5}$$

Now we want to compute the discrete time transfer function related to the continuous one with the sampling time $T = 0.001$.

With the command `c2d` in matlab we get:

```

Command Window
>> s=tf('s');
>> X=1/(s^2+s+0.5)

X =

      1
-----
s^2 + s + 0.5

Continuous-time transfer function.

>> Xd=c2d(X,0.001,'zoh')

Xd =

4.998e-07 z + 4.997e-07
-----
z^2 - 1.999 z + 0.999

Sample time: 0.001 seconds
Discrete-time transfer function.

```

We can rewrite the transfer function as (using the usual symbols y for the output and u for the input):

$$F(z) = \frac{Y(z)}{U(z)} = \frac{(4.998)10^{-7} z^{-1} + (4.998)10^{-7} z^{-2}}{1 - 1.999z^{-1} + 0.999z^{-2}} = \frac{N(z)}{D(z)}$$

We can now derive a difference equation that represents the transfer function; we write:

$$Y(z)D(z) = U(z)N(z)$$

The discrete time equivalent of this expression is:

$$y(n) - 1.999y(n-1) + 0.999y(n-2) = (4.998)10^{-7} u(n-1) + (4.998)10^{-7} u(n-2)$$

And so we get the expression for the output at the time instant n :

$$y(n) = 1.999y(n-1) - 0.999y(n-2) + (4.998)10^{-7} u(n-1) + (4.998)10^{-7} u(n-2)$$

This expression shows that the behavior of the system is given not only by the present input, but also by the past inputs and outputs; if we want to know the output at a time n we must know the previous two inputs and outputs of the system.

Useful links:

http://techt teach.no/publications/discretetime_signals_systems/discrete.pdf from TechTeach about discrete time transfer functions and discrete systems

FROM STATE SPACE TO TRANSFER FUNCTION

We can obtain the transfer function of the system **starting from the state space** representation by taking the Laplace transform of the two equations ; from the first we get:

$$s\mathbf{X}(s) - \mathbf{x}(0) = A\mathbf{X}(s) + B\mathbf{U}(s) \rightarrow (sI - A)\mathbf{X}(s) = B\mathbf{U}(s) + \mathbf{x}(0)$$

From which we get the expression for the states in the s domain:

$$\mathbf{X}(s) = (sI - A)^{-1}B\mathbf{U}(s) + (sI - A)^{-1}\mathbf{x}(0)$$

By substituting this expression in the output expression for the system we get:

$$\mathbf{Y}(s) = C((sI - A)^{-1}B\mathbf{U}(s) + (sI - A)^{-1}\mathbf{x}(0)) + D\mathbf{U}(s)$$

If the **initial conditions vanish** and we divide the output by the input:

$$\mathbf{G}(s) = \frac{\mathbf{Y}(s)}{D\mathbf{U}(s)} = C(sI - A)^{-1}B + D$$

The derivation of the z domain transfer function is the same, taking into account that in this domain we have:

$$\mathbf{x}(k + 1) \rightarrow \mathbf{X}(k)z$$

BEHAVIORAL ANALYSIS

The behavioral analysis is the study of the characteristics of the response of a system when it is stimulated by an input signal or when it is free to evolve in time without external inputs.

FREE RESPONSE AND FORCED RESPONSE

In both the two main LTI systems descriptions, i.e. transfer functions and state space formulation, we can observe that the output of a system can be derived from the contributions of **two separated stimuli**, which are the inputs and the initial conditions (states values at $t = t_{(0)}$) of the system. Starting from this observation we can exploit the **linearity principle** to assert that the generic response of a system is given by the sum of two terms:

- **Free response:** is the output of the system when it evolves without external inputs
- **Forced response:** is the output of the system with a given input and with null initial conditions

In particular the free response is fundamental for the characterization of a system; the **natural modes** are the characteristics evolutions the *free response* of the system.

These modes will be real functions of time $m(t)$, and we are particularly to their asymptotic behavior; a mode can be:

- Convergent: $\lim_{t \rightarrow \infty} |m(t)| = 0$
- Bounded: $0 < |m(t)| < M$



- Divergent: $\lim_{t \rightarrow \infty} |m(t)| > M \quad \forall M \in \mathbb{R}$

The natural modes can be studied both through the Laplace domain representation of the system through the state space representation.

FREE RESPONSE FROM THE STATE SPACE

The free response can be evaluated from the state space both in continuous and discrete time; in the following the regarding basic concepts will be shown.

COMPUTING THE FREE EVOLUTION FROM THE STATE SPACE

Suppose we are dealing with a system with no input applied, of which we want to consider the **free evolution**; the equation that we are interested in is the following:

$$\dot{\mathbf{x}}(t) = A\mathbf{x}(t)$$

Taking the Laplace transform, we obtain:

$$s\mathbf{X}(s) - \mathbf{x}(0) = A\mathbf{X}(s) \rightarrow \mathbf{X}(s) = (sI - A)^{-1}\mathbf{x}(0)$$

The term $(sI - A)^{-1}$, already appeared in the previous, is called **resolvent**.

If we want to compute the value of the states in the time domain, we have to apply the inverse Laplace transform to the expression and get:

$$\mathbf{x}(t) = \mathcal{L}^{-1}\{(sI - A)^{-1}\}\mathbf{x}(0)$$

$$\mathbf{x}(t) = \Phi(t)\mathbf{x}(0)$$

Where we denoted:

$$\mathcal{L}^{-1}\{(sI - A)^{-1}\} = \Phi(t)$$

The matrix $\Phi(t)$ is called **state-transition matrix**, and maps the values of the states from the initial condition $\mathbf{x}_{(0)}$ to the value that they assume at the time t .

For the discrete time formulation, the calculation of the evolution of the state vector is simpler, since it only implies the **discrete iteration** of the state-space mapping until the desired time step is reached.

Useful links

More about the state transition matrix:

<http://web.mit.edu/2.14/www/Handouts/StateSpaceResponse.pdf> From MIT website

https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-241j-dynamic-systems-and-control-spring-2011/readings/MIT6_241JS11_chap11.pdf From MIT website

“Modern control systems” ninth edition, R.C. Dorf, R.H. Bishop, Prentice Hall

“System dynamics” K. Ogata, Prentice Hall

“Fundamentals of linear state space systems”, John Bay, McGraw Hill, 1998

MATRIX EXPONENTIAL

We can arrange the mathematical formulation of the state space system to get a notation that is similar to the one of a simple differential equation in the following way:

Starting from the identity (that holds if the series is convergent):

$$(I - M)^{-1} = I + \sum_{n=1}^{\infty} M^n$$

We can write the resolvent as:

$$(sI - A)^{-1} = \frac{1}{s} \left(I - \frac{A}{s} \right)^{-1} = \frac{I}{s} + \sum_{n=1}^{\infty} \frac{A^n}{s^{n+1}}$$

It leads us to the expression:

$$\Phi(t) = \mathcal{L}^{-1}\{(sI - A)^{-1}\} = I + \sum_{n=1}^{\infty} \frac{(tA)^n}{n!}$$

If we now recall the exponential **power series**:

$$e^{at} = 1 + \sum_{n=1}^{\infty} \frac{(ta)^n}{n!}$$

We note that it takes the same form of the $\Phi(t)$ expansion, with the difference that in this last case the terms are matrices instead of scalars.

So, given the definition of **matrix exponential**:

$$e^M = I + \sum_{n=1}^{\infty} \frac{M^n}{n!}$$

The state transition matrix can be written like follows:

$$\Phi(t) = e^{At}$$

With this result the solution of the vector equation:

$$\dot{\mathbf{x}}(t) = A\mathbf{x}(t)$$

Can be written in the following form, in analogy with the scalar case:

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

This result can be extended to discrete systems:

$$\mathbf{x}(k+1) = \mathbf{x}(k)e^{(t_{k+1}-t_k)A}$$

The difference between the two time instants equals the sampling time of the system T_s ; thus, we get:

$$x(k+1) = x(k)e^{T_s A}$$

NATURAL MODES

Given the state space formulation of a system without exogenous inputs:

$$\dot{x} = Ax$$

We know that, given an initial state $x(0_-)$, the value of the states at any time instant will be:

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

To introduce the concept we take the case for which A is a diagonal matrix with real distinct eigenvalues; in this particular case, the matrix exponential takes the form:

$$e^{At} = \begin{pmatrix} e^{\lambda_1 t} & 0 & \dots & 0 \\ 0 & e^{\lambda_2 t} & \dots & 0 \\ \dots & \dots & \dots & 0 \\ 0 & 0 & 0 & e^{\lambda_n t} \end{pmatrix}$$

In this case the evolution of the system will result to be:

$$x_{(t)} = \begin{pmatrix} x_1(0_-)e^{\lambda_1 t} \\ x_2(0_-)e^{\lambda_2 t} \\ \dots \\ x_n(0_-)e^{\lambda_n t} \end{pmatrix}$$

Every function $m_{i(t)} = e^{\lambda_i t}$ is called natural mode of the system.

In this case the natural modes are exponential functions, since we supposed that the eigenvalues of the A matrices were all real and distinct. Anyway, for a linear time invariant system the eigenvalues can be also complex conjugated pairs and can be not distinct (i.e. two eigenvalues can have the same value). Given the knowledge of the eigenvalues, we can find the following natural modes for the system:

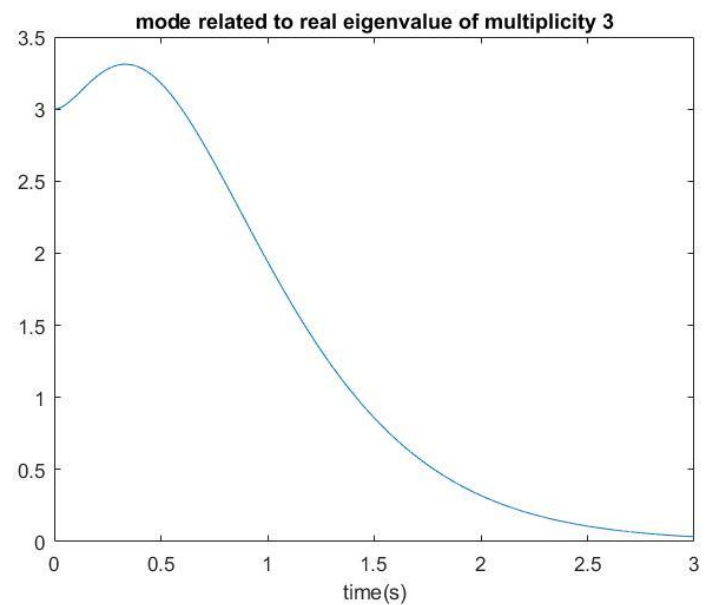
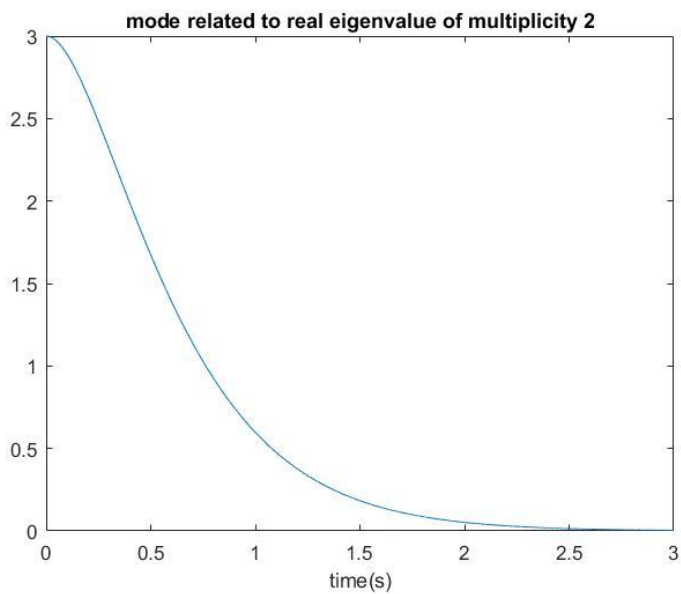
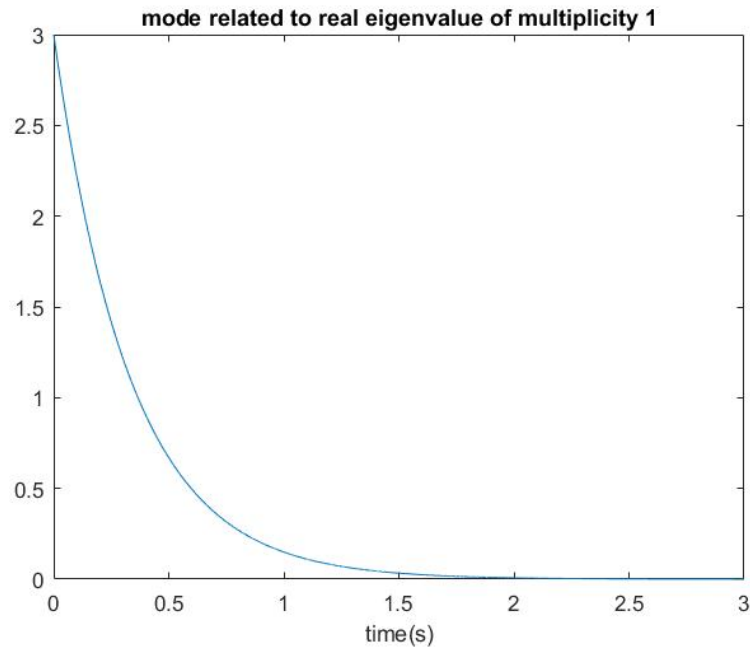
Aperiodic modes:

Are associated to real eigenvalues; given an eigenvalue λ of multiplicity n , there will be n associated natural modes of the following form:

$$m_k(t) = t^k e^{\lambda t}$$

With k going from 0 to $n - 1$.

This kind of modes have an asymptotic behavior that depends only on the value of the eigenvalue; if the eigenvalue is negative, the modes will all be convergent. They will diverge for positive eigenvalue and bounded (and equal to one) only if the eigenvalue is null and its multiplicity is one. In the following, typical shapes of natural modes of this type with negative λ are reported:



The inverse of λ is the time constant and represents the rate of decay (or amplification for positive λ) of the mode.

Pseudo-periodic modes:

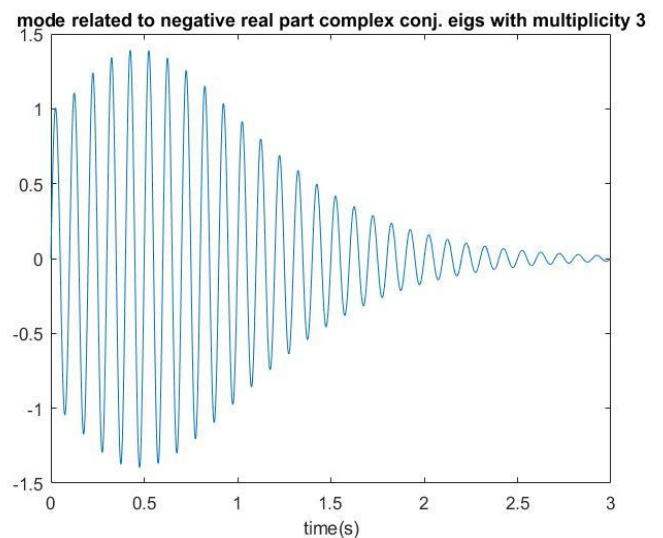
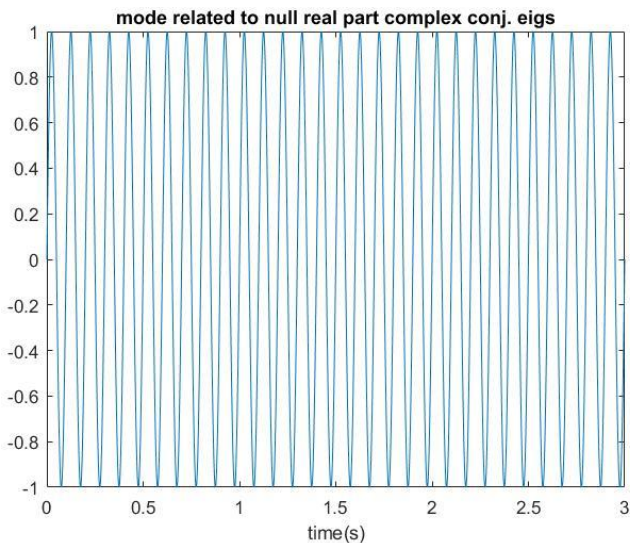
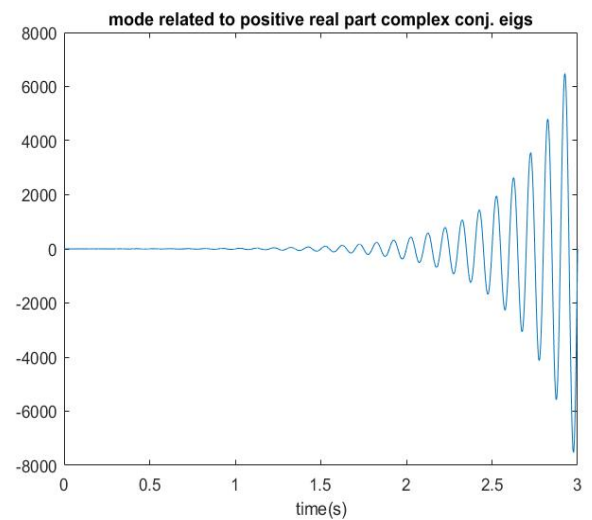
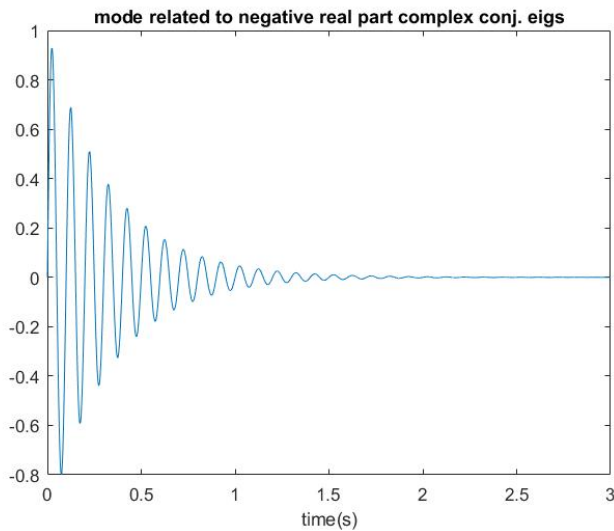
Are associated to complex conjugated eigenvalues of the form $\lambda = \sigma + j\omega$, with multiplicity n . There will be n modes of the kind:

$$m_k(t) = t^k e^{\sigma t} \cos(\omega t + \varphi)$$

With k going from 0 to $n - 1$.

This kind of mode shows an oscillation that can be amplified during the evolution if $\sigma > 0$, generating a divergent mode, of constant amplitude for $n = 1$ and $\sigma = 0$ (purely imaginary eigenvalues) or convergent to zero for $\sigma < 0$.

In the following some typical shapes of these modes are reported:



Natural modes for discrete time systems

When we handle a discrete time system, the shape of the natural modes are the same as the ones of the continuous time system, but they are obviously discretized. What changes is the relationship between the value of the eigenvalues and the behavior of the modes; for the discrete systems, we can distinguish the following cases based on the position of the eigenvalues in the Z domain:

- The eigenvalues inside the unitary circle in the complex plane generates convergent modes
- The eigenvalues outside the unitary circle in the complex plane generates divergent modes

- The eigenvalues on the unitary circle in the complex plane generates bounded modes
- The real eigenvalues generates exponential decay if $0 < \lambda < 1$
- Also the real eigenvalues can generate oscillations if $-1 < \lambda < 0$
- The modes are as fast as they are near to the origin of the plane
- Complex conjugate eigenvalues generates modes that are faster as the angle with the real axis increases
- Complex conjugate eigenvalues generates modes that are more resonant when the couple of eigenvalues is near to the unitary circle

Usefull links:

About natural modes:

https://ocw.mit.edu/courses/aeronautics-and-astronautics/16-07-dynamics-fall-2009/lecture-notes/MIT16_07F09_Lec19.pdf from MIT university

“Fundamentals of linear state space systems”, John Bay, McGraw Hill, 1998

EXAMPLE: NATURAL MODES FROM THE SYSTEM MATRIX

Consider now the ex4.1 system matrix for an RLC circuit:

$$A = \begin{pmatrix} -\frac{R}{L} & -\frac{1}{L} \\ \frac{1}{C} & 0 \end{pmatrix}$$

We want to find the eigenvalues of the system, but in this case, the matrix is not diagonal. The equation of interest is the following:

$$\dot{\mathbf{x}} = A\mathbf{x}$$

We can apply the transformation $T\mathbf{x}$, where T is the matrix that has for columns the eigenvectors of A . We get:

$$T\dot{\mathbf{x}} = AT\mathbf{x}$$

And apply now the inverse transformation:

$$T^{-1}T\dot{\mathbf{x}} = T^{-1}AT\mathbf{x}$$

that, for definition returns us the form:

$$\dot{\mathbf{x}} = U\mathbf{x}$$

Where L is the diagonal matrix that contains only the eigenvalues of the A matrix; in our case substituting the numerical values for R, L and C we get:

$$U = 10^7 \begin{pmatrix} -9.8990 & 0 \\ 0 & -0.1010 \end{pmatrix}$$

Now we can state that:

$$e^{Ut} = \begin{pmatrix} e^{-9.8990 \cdot 10^7 t} & 0 \\ 0 & e^{-0.1010 \cdot 10^7 t} \end{pmatrix}$$

So the natural modes of the system are:

$$\mathbf{x}(t) = \begin{pmatrix} \mathbf{x}_{1(0-)} e^{-9.8990 \cdot 10^7 t} \\ \mathbf{x}_{2(0-)} e^{-0.1010 \cdot 10^7 t} \end{pmatrix}$$

We see that the modes are both exponentially convergent.

FREE RESPONSE FROM THE TRANSFER FUNCTION

From the theory, we know that when we apply the inverse Laplace transform to a transfer function, the kind of time domain functions that we find are always combination of exponentials and sinusoidal functions; in fact, the poles and the relative residues of the transfer functions are always real or complex conjugated poles.

Consider now the expression $(sI - A)$; since it represents a square matrix, we can calculate its determinant, a polynomial of the s variable:

$$P(s) = \det(sI - A)$$

by imposing the value of the polynomial equal to zero, we can find its roots, or rather, the eigenvalues λ of the matrix A :

$$P(s)|_{s=\lambda} = 0$$

Since A is a real value matrix, its eigenvalues must be real or complex conjugated pairs. The values of the eigenvalues are **strictly related to the poles** of the transfer function that is obtained from the state space representation, since the denominator of the transfer function is related to the value of $(sI - A)$.

Therefore, the responses of system, given a transfer function, will show the same qualitative behavior of the natural modes.

The natural modes can be found also directly in the Laplace domain by taking into account the initial conditions of the transformed quantities when we manipulate the differential equation that holds the system and operating then the inverse Laplace transform.

All the considerations also hold for discrete time systems.

EXAMPLE: NATURAL MODES AND TRANSFER FUNCTION

Now we derive the transfer function for the RLC circuit with the input and the output specified in the ex. 4.1; we must use the formula:

$$G(s) = \frac{Y(s)}{U(s)} = C(sI - A)^{-1}B + D$$

Using the corresponding numerical values for the matrices, we obtain:

$$G(s) = \frac{10^{14}}{s^2 + 10^8 s + 10^{14}}$$

For which the poles are:

$$\begin{aligned} p_1 &= -9.8990 * 10^7 \\ p_2 &= -0.1010 * 10^7 \end{aligned}$$

Coincident with the state matrix eigenvalues.

FORCED RESPONSE

Among all the possible realizations of the forced response of a LTI system, in the following we expose some basic concepts about responses of particular interest; among them, one must include also the impulse response, already mentioned in the previous.

STEP RESPONSE

The step response is the way the system react to the application of a step signal while all the initial conditions are null.

Even if the step signal is constant for $t > t(0)$, this kind of response gives an almost **exhaustive information** about the system characteristics, since the **ideally infinite derivative of the input** for $t = t(0)$ stimulates the system with **an high frequency content**.

Even if the characteristics of the step response is useful to evaluate the performance of a feedback control system, they can already be defined, with the hypothesis that the system's output will be constant and proportional to the step amplitude after a certain time interval has passed (that is, a **steady state** is reached):

- **Overshoot:** is defined as the difference between the peak value of the impulse response and the amplitude of the input step, normalized with respect to the amplitude of the step:
$$\hat{s} = \frac{y_{res} - y_{step}}{y_{step}}$$
- **Rise time:** is a measure of how the system is able to track a fast input signal; it is defined as the time interval needed by the system to pass from 10% to 90% of the final steady state value. If the response oscillates around the final value before the steady state is reached, then the rise time is the time needed to pass from 0% to 100% of the final steady state value.



- **Settling time:** with reference to a percentage value α (5%, 10%) of the steady state value, is the time needed by the system to have its response bounded within a range of $\pm\alpha$ around the steady state value

POLYNOMIAL RESPONSE

The polynomial response of order k is the response of a system to inputs of the form:

$$u_k = \frac{t^k}{k!}$$

In particular the response to the **ramp** and to the **parabolic input** (first and second order respectively) are usually taken into account to evaluate the system's characteristics.

THE TRANSFER FUNCTION IN STEADY STATE: FREQUENCY RESPONSE

The transfer function makes possible to calculate the evolution of a system when an input signal is applied. This gives us information about the whole trend of the output.

From the ODE theory, we know that after a certain time, the response of the system will get a **steady state** (or will diverge), since every **sinusoidal component** of the input will be found in the output with the same frequency, but with **amplitude and phase modified** by the action of the system. The information about amplification and phase delay of each component can be found in the transfer function if it is evaluated in the frequency domain rather than in the Laplace domain. This kind of representation is useful since allow to evaluate the rejection/amplification characteristics of a system, in particular with the aid of the **Bode diagrams**, that will be discussed in the next sections. The expression of the transfer function in steady state is

$$H(j\omega) = \frac{Y(j\omega)}{X(j\omega)}$$

and represents the Fourier Transform of the impulse response. It is called **frequency response** of the system.

The rational function's absolute value and phase are respectively the **amplification** and the **phase delay** that an input sinusoidal function experiences when is transformed by the system.

Useful links:

About transient and steady state response:

<http://www.ece.rutgers.edu/~gajic/psfiles/chap6.pdf> from Rutgers University

About step response:

<http://lpsa.swarthmore.edu/Transient/TransInputs/TransStepTime.html> from Swarthmore College

About General system's responses:

http://www.mne.psu.edu/cimbala/me345/Lectures/Dynamic_systems.pdf From Penn state University

Most of previously suggested textbooks treat the argument.

EXAMPLE: FROM TRANSFER FUNCTION TO FREQUENCY RESPONSE:

We take the transfer function:

$$X(s) = \frac{1}{s^2 + s + 0.5}$$

We now apply to the system the input signal:

$$g(t) = \sin(2\pi t)$$

The input has pulsation $\omega = 1$.

We can evaluate the steady state response of the system to the input by substituting the Laplace variable with the $j\omega$ term and get:

$$X(j\omega) = \frac{1}{-\omega^2 + j\omega + 0.5}$$

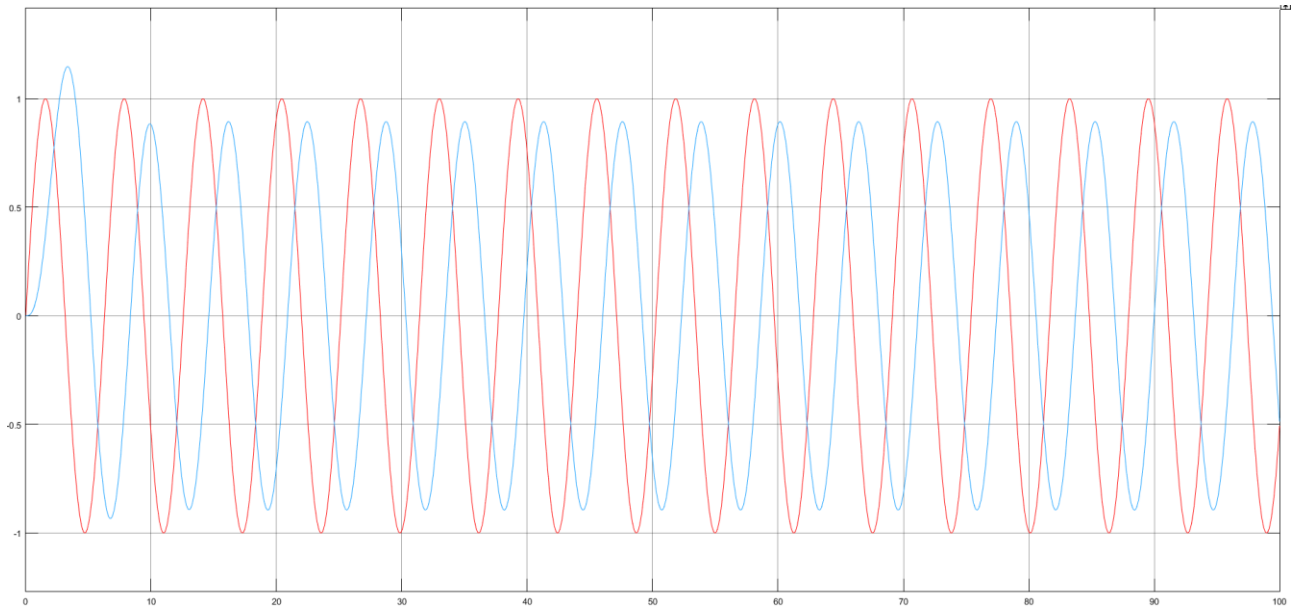
Once we fix a pulsation, we obtain a complex vector. We put $\omega = 1$ to evaluate the response of the system to our input:

$$X(j) = \frac{1}{j - 0.5} = 0.8944e^{-j 2.0344}$$

From this information about the absolute value and the phase of the frequency response we can get the output:

$$y(t) = |X(j)| \sin(2\pi t + \text{angle}(X(j))) = 0.8944 \sin(2\pi t - 2.0344)$$

A simulation shows the evolution of the system and confirms the results and the steady state condition that is reached after the transitory.



The input (red) and the output (blue)

RELATED MATLAB COMMANDS

A list of useful Matlab functions regarding integral transforms follows; notice that from the command window one can type '**help**' followed by the command name to have almost all the information about that function, in particular about outputs and necessary input arguments.

- `ss (...)` returns a system representation given the state space matrices
- `impz (...)` plots the behavior of the impulse response given a transfer function
- `conv (...)` performs the discrete convolution between two vectors
- `tf (...)` also generates a system representation given the transfer function coefficients
- `c2d (...)` transforms a Laplace domain transfer function into a Z-domain transfer function
- `d2c (...)` performs the inverse operation of `c2d`
- `zpk (...)` formulates a transfer function in form of zeros-poles-gain
- `pole (...)` returns the poles of a transfer function
- `ss2tf (...)` given a state space representation returns the transfer function
- `tf2ss (...)` performs the inverse operation of `ss2tf`
- `eig(...)` finds eigenvector and eigenvalues of a matrix
- `expm(...)` calculates the matrix exponential
- `step(...)` plots the step response of a given transfer function
- `mineral(...)` simplifies transfer functions with common factors within a prefixed tolerance

CONTROL SYSTEMS ANALYSIS

By analysis, we mean: observing and classifying the performance of the systems in response to changes in inputs and disturbance. Roughly speaking:

- How fast is it reaching/responding?
- What is the final value of its response.
- Is it oscillating?
- Is it stable?

Since the **objective** of control is to design systems that have the **performance we desire**, we need at this stage to define clear **performance characteristics** in response to known input signals. And to try to qualitatively describe systems.

Mainly, there are Analysis techniques in **Time Domain** and in **Frequency Domain**, and consequently, there are performance characteristics which were defined in Time Domain and in Frequency Domain.

TIME DOMAIN CHARACTERISTICS

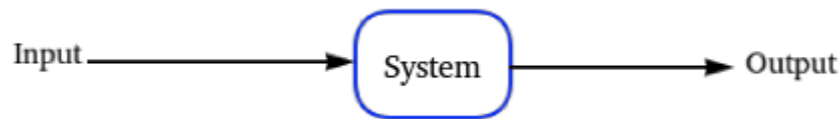


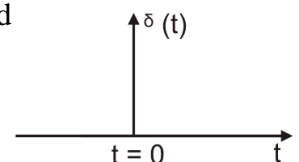
Figure 1- System representation

The above figure shows the main elements in the analysis procedure: the **Input**, the **System** and the **Output**:

For the input, usually known mathematical signals are used, they are, as already reported:

1. Impulse Signal:

The impulse signal has zero width, infinite height and finite area (integral). It is plotted with an arrow with a height showing the area of the impulse.

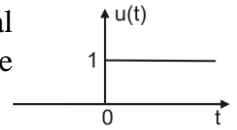


$$\delta(t) = 0 \quad t \neq 0$$

$$\int_{-\infty}^{\infty} \delta(t) dt = 1$$

2. Step Signal:

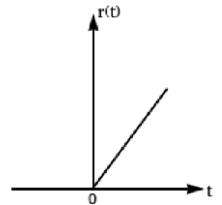
The Step signal is the integral of the impulse signal, it has a value of zero before the initial time, then a sudden change of value (theoretically at zero time), to the value of the amplitude.



$$u(t) = \begin{cases} 0 & \text{for } t < 0 \\ 1 & \text{for } t \geq 0 \end{cases} \quad U(s) = \frac{A}{s}$$

3. Ramp Signal:

The ramp signal is the integral of the step signal. Like the step, the ramp has a value of zero before the initial time, then it increases gradually and keeps increasing forever.



$$r(t) = \begin{cases} 0 & \text{for } t < 0 \\ t & \text{for } t \geq 0 \end{cases} \quad R(s) = \frac{A}{s^2}$$

4. Sinusoid Signal:

The Sinusoid signal is useful in frequency response techniques, which will be shown in the next section.

Useful links: Input types: <http://lpsa.swarthmore.edu/BackGround/ImpulseFunc/ImpFunc.html>

Output response is the sum of two responses: **forced response** and **natural response**.

SYSTEMS RESPONSE CHARACTERISTICS IN TIME-DOMAIN

In time-domain, performance of dynamical systems can be assessed **qualitatively** through analytical solution of differential equations or **quantitatively**.

QUALITATIVE TECHNIQUES (POLES – ZEROS CONCEPT)

A **pole** is a value of the complex frequency variable “s” that makes transfer function become infinity. Or can be described as a root of the denominator.

A **zero** is a value of “s” that makes the transfer function become zero. Or a root of the nominator.

EXAMPLE

Find the Poles and the zeros of the following transfer function:



$$\frac{4s^2 - 20s + 24}{s^2 + 3.3s + 6.4}$$

Solution: The roots of the nominator and the denominator can be calculated as follows:

$$\frac{4s^2 - 20s + 24}{s^2 + 3.3s + 6.4} = \frac{4(s - 3)(s - 2)}{(s + 1.65 - j1.9177)(s + 1.65 + j1.9177)}$$

Then it is easily to find the values of “s” that make the transfer function equals 0, which are: **z1=3**, and **z2=2**.

And the values that make the transfer function equals ∞ , which are: **p1=-1.65+j1.9177**, and **p2=-1.65-j1.9177**

MATLAB CODE

```
Command Window
>> s=tf('s');
>> G=(4*s^2-20*s+24)/(s^2+3.3*s+6.4)

G =

    4 s^2 - 20 s + 24
    -----
    s^2 + 3.3 s + 6.4

Continuous-time transfer function.

>> pole(G)

ans =

    -1.6500 + 1.9177i
    -1.6500 - 1.9177i

>> zero(G)

ans =

     3.0000
     2.0000

>> pzmap(G)
```

POLE-ZERO MAP

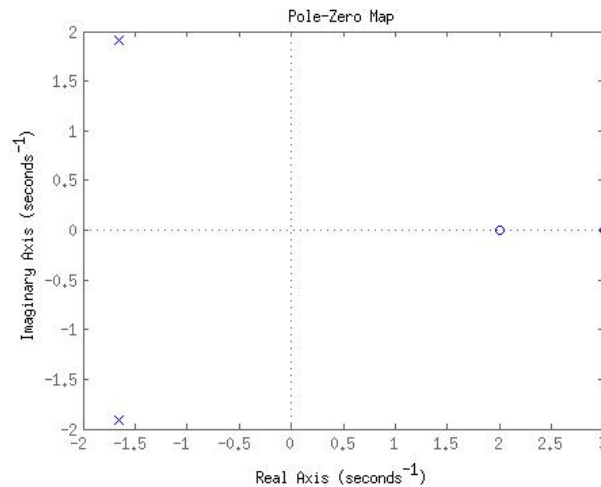


Figure 2- Pole-Zero Map (MATLAB function pzmap)

EXAMPLE

Evaluating the Response Using the Poles (Qualitative approach) [Nise]:

Given the System with the transfer function shown in the figure, with input $R(s)$ is a unit step. Write the output $c(t)$ in general terms, specifying the forced response and the natural response of the solution:

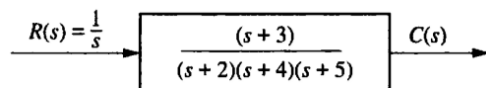


Figure 3- System in pole-zero form

SOLUTION

By inspection, each system pole generates an exponential as part of the natural response. The input's pole generates the forced response. Thus, by performing Partial Fraction analysis:

$C(s) = \frac{K}{s} + \frac{K_2}{s+2} + \frac{K_3}{s+4} + \frac{K_4}{s+5}$, where the first part $\frac{K}{s}$ is the forced response, and the other terms represent the natural response because they result from the poles of the system (system dynamics).

From the inverse Laplace transform, we get: $c(t) = K_1 + K_2 e^{-3t} + K_3 e^{-4t} + K_4 e^{-5t}$. Again, the first part K_1 is the forced response, and the other terms are due to the system dynamics (poles).

SYSTEM STABILITY

CONCEPT

Linear Systems response consists a sum of the Forced and Natural responses.

$$C_{total} = C_{forced} + C_{natural}$$

Since in steady-state the response consists only of the forced response. The **natural response** defines the stability of the system. From the Behavioural Analysis section about Natural Modes we know that they are of three types: **Convergent**: if it disappears after some amount of time, **Divergent**: if it tends to infinity, and **Bounded**: if it does not tend to infinity but it also does not disappear.

So, in general, the stability of the system is related to how its natural response will emerge with time. The following example demonstrate this property.

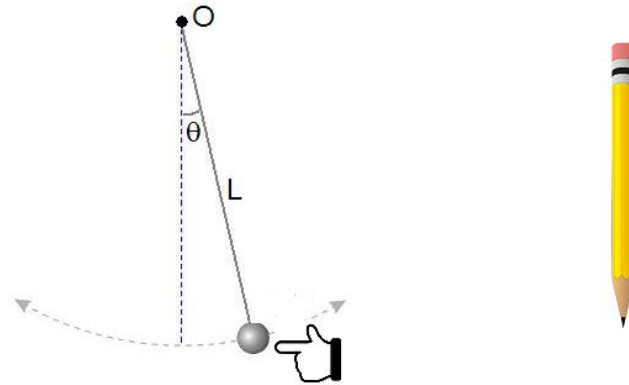


Figure 4- A simple Pendulum and a pencil standing on its tip

The pendulum returns to the rest position ($\theta=0$) even after it is moved by an external effect, therefore it can be described as a stable system. While in the other case the pen will fall if you try to place it on its tip regardless at what angle it is placed, in this case it is an unstable system.

DEFINITION

A simple definition to stability. Related only to **Linear Time-Invariant systems (LTI)** is the following, followed by a more general and detailed definition of stability:

A linear, time-invariant system is **stable** if the natural response approaches zero as time approaches infinity.

A linear, time-invariant system is **unstable** if the natural response grows without bound as time approaches infinity.

A linear, time-invariant system is **marginally stable** if the natural response neither decays nor grows but remains constant or oscillates as time approaches infinity.

FORMAL DEFINITION

In general the definition of stability is divided into two categories: **Internal Stability** and **Bounded Input Bounded Output (BIBO) Stability**.

From Part ("Behavioral Analysis"), we found that LTI systems responses are divided into two parts: one is due to the input $u(t)$ (**zero-state response** X_{zs}) and another due to initial state $x(t)$ (**zero-Input response** X_{zi})

The LTI system is represented by a state equation: $\dot{x}(t) = Ax(t) + Bu(t)$,

The zero-state response is obtained from the contribution $u(t)$ (no initial condition of the states $X_o = 0$).

While the zero-input response is obtained from the contribution of X_o (no input is applied $u(t)=0$).

Due to superposition, the system response is expressed as a combination of the two responses.

INTERNAL STABILITY OF CT SYSTEMS

Now we could say that the concept of internal stability is related to whether the zero-input response is bounded for any initial state X_o or not.

$$X_{zi}(s) = (sI - A)^{-1}X(0)$$

Therefore, the LTI system is internally stable if all the **eigenvalues of A have negative real parts**, and **at most one eigenvalue has zero real part**. This will guarantee that the zero-input response X_{zi} will **never grow unbounded** with time for any initial condition X_o .

ASYMPTOTIC STABILITY OF CT SYSTEMS

The LTI system is asymptotically stable if all the **eigenvalues of A have strictly negative real parts**, which means that all the natural modes are **convergent**.

BIBO STABILITY OF CT SYSTEMS

The BIBO stability is related to obtaining a bounded zero-state response for any bounded input $u(t)$.

from part ("Behavioural Analysis") we obtained an expression of X_{zs} and X_{zi} :

$$X_{zs}(s) = (C(sI - A)^{-1}B + D)u(s)$$

From Previously the relation between the natural modes of the system and the roots of the characteristic polynomial $\det(sI - A)$ we can relate them to the concept of stability.

The roots of the characteristic polynomial are the eigenvalues of A, or the **poles of the transfer function** $H(s) = C(sI - A)^{-1}B + D$.

For BIBO, X_{zs} contains the characteristic polynomial at the de-numerator, but it also contains additional terms in the numerator, which create **zeros that can be cancelled with the poles** "unstable roots" of the characteristic polynomial. $H(s) = C(sI - A)^{-1}B + D = \frac{C \cdot \text{adj}(sI - A) \cdot B}{\det(sI - A)} + D$

The possibility of **pole-zero cancellation** makes the internal stability condition more strict than the BIBO stability one.

INTERNAL STABILITY OF DT SYSTEMS

Similar to CT system case, the internal stability is related to the zero-input response.

$$X_{zi}(z) = (zI - A)^{-1}X(0)$$

Therefore, the LTI system is internally stable if all the **eigenvalues of A have magnitudes less than 1**, and **at most one eigenvalue has a magnitude of 1**. This will guarantee that the zero-input response X_{zi} will **never grow unbounded** with time for any initial condition X_o .

ASYMPTOTIC STABILITY OF DT SYSTEMS

The LTI system is asymptotically stable if all the **eigenvalues of A have strictly magnitudes less than 1**, which means that all the natural modes are **convergent**.

BIBO STABILITY OF DT SYSTEMS

The BIBO stability is related to obtaining a bounded zero-state response for any bounded input $u(k)$.

Similarly:

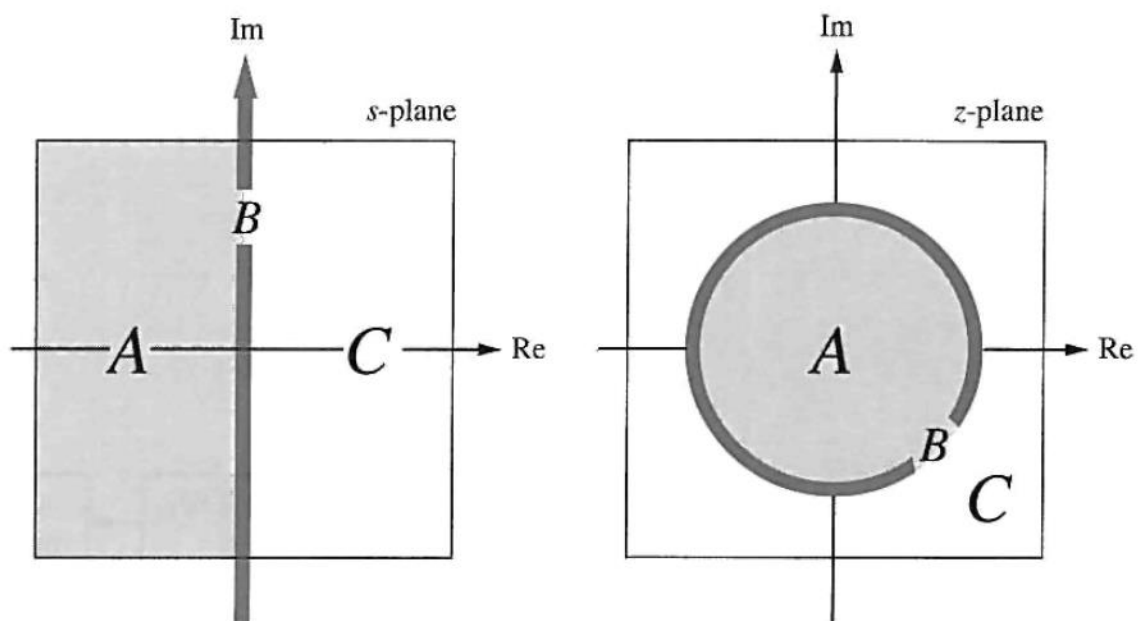
$$X_{zs}(z) = (C(zI - A)^{-1}B + D)u(z)$$

From Previously the relation between the natural modes of the system and the roots of the characteristic polynomial $\det(zI - A)$ we can relate them to the concept of stability.

The roots of the characteristic polynomial are the eigenvalues of A, or the **poles of the transfer function** $H(z) = C(zI - A)^{-1}B + D$.

Similarly to the CT case, there is a possibility of **pole-zero cancellation** that makes the internal stability condition more strict than the BIBO stability one.

The following figure shows the stability regions for CT and DT systems.



A is asymptotic stability region, **B** is stability region and **C** is instability region.

CAN A SYSTEM BE BIBO STABLE AND NOT INTERNALLY STABLE?

Consider the following the example:

EXAMPLE

$$\dot{X}(t) = \begin{bmatrix} -1 & 2 \\ 1 & 0 \end{bmatrix} X(t) + \begin{bmatrix} 1 \\ 0 \end{bmatrix} u(t)$$

$$y(t) = [3 \quad -3]X(t) + u(t)$$

The eigenvalues of A are -2 and 1, therefore the system is **internally unstable**.

But to analyze the BIBO stability, we need to compute the transfer function H(s) using the formula mentioned above.

In this case, the “unstable” pole (+1) is canceled with the zero, therefore the system is **BIBO stable**.

MATLAB COMMANDS:

```
Command Window
>> A=[-1 2; 1 0]; B=[1 ; 0]; C=[3 -3]; D=1;
>> eig(A)

ans =
    -2
     1

>> s=tf('s')

s =
    s

Continuous-time transfer function.

>> H=minreal(zpk(C*inv(s*eye(2)-A)*B+D))

H =
    (s+5) (s-1)
    -----
    (s+2) (s-1)

Continuous-time zero/pole/gain model.
fx >> |
```

P.S. : to cancel the unstable pole with the zero, use the MATLAB function “**minreal**” with an appropriate tolerance (max difference between the pole and the zero to cancel each other):

```
Command Window
>> minreal(H, 0.000001)

ans =
    (s+5)
    -----
    (s+2)

Continuous-time zero/pole/gain model.
fx >> |
```

IMPORTANCE OF STABILITY

When designing a feedback control system, stability is the first property to be considered. Other properties, such as controllability, observability, bandwidth etc. play an important role. But stability is the **most important one**.

Since the goal of control design is to obtain systems that have their outputs follow their inputs as closely as possible. Unstable systems cannot guarantee such behavior and are not useful in practice. Moreover, unstable systems have the amplitude of at least one of the state tends to infinity with time, this theoretical result leads in practice to undesirable situations like saturation, or to damage such as burning, breaking or explosion. Therefore, not satisfying stability makes it unreasonable to try to satisfy other system properties.

TESTING TECHNIQUES

Several techniques are found to evaluate the stability of the system other than checking the locations of the poles (or the roots of the characteristic polynomial), such as **Routh-Hurwitz** and frequency response techniques (see part("Frequency Domain Tools")).

Another tool is the **impulse response**, it can check the stability of the system since the impulse is a bounded signal. Routh-Hurwitz Criterion: Gives information about the number of closed-loop systems poles in the left-half plane, the right-half plane and at the $j\omega$ axis. Without the need to solve for the closed-loop system poles.

FREQUENCY DOMAIN METHODS

An Important property of LTI systems, is that for sinusoidal inputs, the output is always sinusoidal in the steady state case. Moreover, the output frequency is the same as the input.

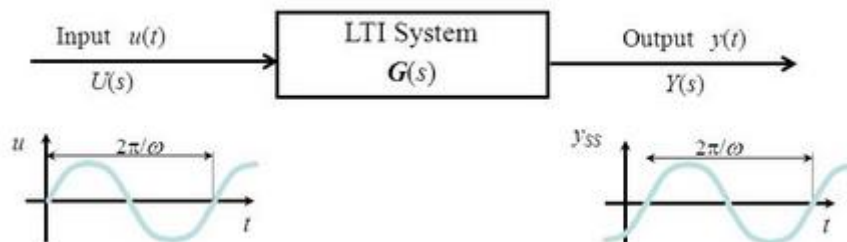


Figure 5- LTI systems response to sinusoid signals

The difference in the two signals is in the **amplitude**, and the **phase**.

Useful links: Refer to [Nise, Ch10] for mathematical representation of the frequency response of a system.

Video lecture with an introduction of frequency response:
<https://www.youtube.com/watch?v=eh1conN6YM>

REPRESENTATION AND PLOTTING "BODE"

The LTI system in this framework as a magnitude and phase:

$G(s)_{s=j\omega} = G(j\omega) = M_G(j\omega) \angle \phi_G(j\omega)$, where $M_G(j\omega)$ is the scaling in the amplitude of the input signal and $\phi_G(j\omega)$ is the shift in time.

Since they are both functions of the frequency, each can be plotted with the frequency, each in a separate figure.

EXAMPLE

Plot the Magnitude and phase frequency response for the system represented by the following transfer function: $G(s) = \frac{1}{s+2}$.

SOLUTION

$G(j\omega) = \frac{1}{j\omega+2} = \frac{2-j\omega}{\omega^2+4}$, this is a complex function that can be factorized into magnitude and phase:
 $|G(j\omega)| = M(\omega) = \frac{1}{\sqrt{\omega^2+4}}$, and the angle $\phi(\omega) = -\tan^{-1}(\omega/2)$

The two functions can be plotted as in the following figure (with semilog scale and the magnitude in Decibels).

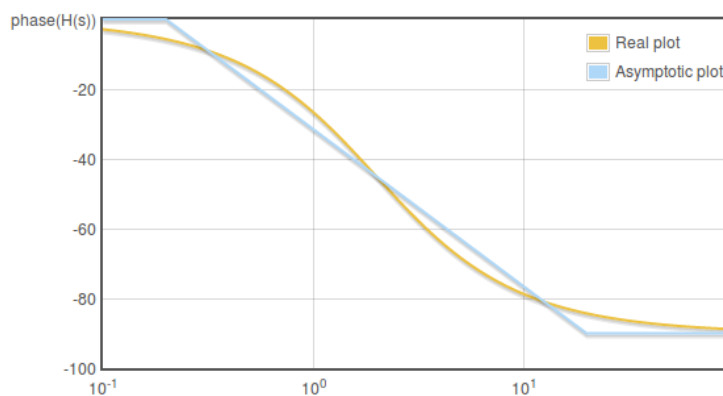


Figure 6- System phase as a function of frequency

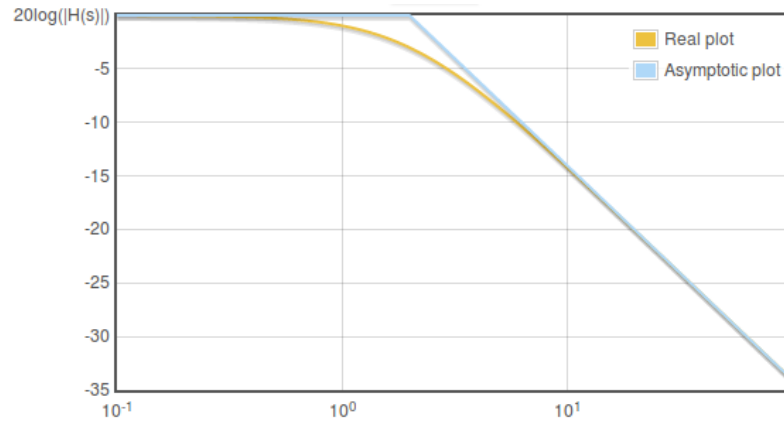


Figure 7- System magnitude as a function of frequency

However, the analytic computation of the frequency response is tedious and time consuming specially for large systems, it can be done easily by plotting the Asymptotic plot instead (refer to the link after), or by exploiting a software tool to perform the plot.

MATLAB CODE TO PLOT THE FREQUENCY RESPONSE

```
Command Window
>> G=tf([1],[1 2])

G =
    1
    ----
   s + 2
Continuous-time transfer function.

fx >> bode(G)
```

This produces the following figure.

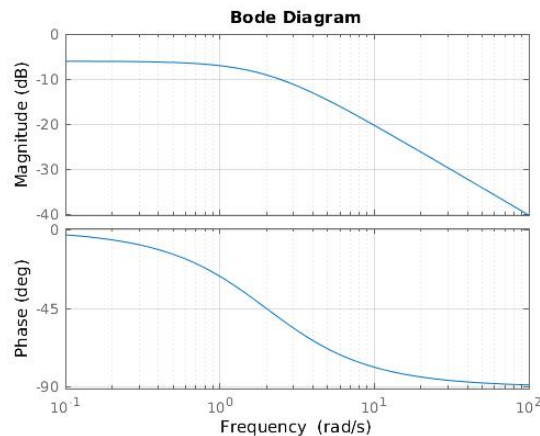


Figure 8- Bode Diagram for $G=1/(s+2)$

Useful links: How to plot Bode Diagram (sketch by hand), with a software plotter:
<http://www.onmyphd.com/?p=bode.plot>

Another way to represent the frequency response: the **Nyquist Plot**: which is a form of **Polar Plot**.

WHAT IS A POLAR PLOT?

The complex function of the LTI system $G(j\omega)$ can be also represented in real and imaginary form:
 $G(s)_{s=j\omega} = G(j\omega) = \Re G(j\omega) + j\Im G(j\omega)$ (**Complex Form**).

The Polar Plot is obtained by plotting the imaginary part $j\Im G(j\omega)$ versus the real part $\Re G(j\omega)$ (**could be also expressed in Polar Form too**) for each frequency value.

EXAMPLE

Given the open-loop continuous time transfer function: $GH(s) = \frac{1}{s+3}$

Letting $s = j\omega$ and rewriting in the polar form ($|magnitude| \angle phase$). We get:

$$GH(j\omega) = \frac{1}{j\omega + 3} = \frac{1}{\sqrt{\omega^2 + 9}} \angle -\tan^{-1} \frac{\omega}{3}$$

Then substitute for several frequency values, for $\omega = 0, \omega = 3, also \omega \rightarrow \infty$

$$GH(j0) = \frac{1}{\sqrt{9}} \angle 0^\circ = 0.333 \angle 0^\circ$$

$$GH(j3) = \frac{1}{\sqrt{18}} \angle -45^\circ = 0.2357 \angle -45^\circ$$

$$\lim_{\omega \rightarrow \infty} GH(j\omega) = 0 \angle -90^\circ$$

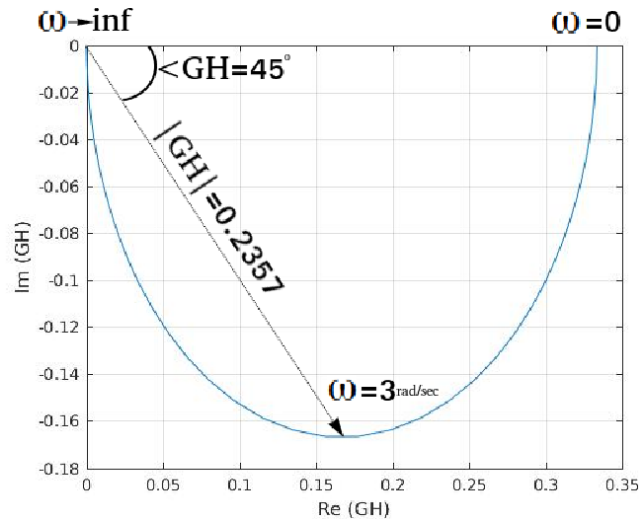


Figure 9- Polar plot for $GH(s)=1/(s+3)$

MATLAB CODE FOR POLAR PLOTTING

```
Command Window
>> s=tf('s')

s =
|
s

Continuous-time transfer function.

>> GH=1/(s+3)

GH =

    1
-----
s + 3

Continuous-time transfer function.

>> [re, im]=nyquist(GH);
fx >> plot(re(:),im(:))
xlabel('Re (GH)')
ylabel('Im (GH)')
```

Useful links: Useful expelation of Nyquist/Polar plots:

<http://www.facstaff.bucknell.edu/mastascu/econtrolhtml/freq/freq6.html>

Next, We will see how the Polar Plot and some properties of complex functions can be exploited to develop Nyquist Stability Theory.

NYQUIST ANALYSIS

The main purpose of Nyquist analysis, is to **determine the absolute and relative stability of closed-loop control systems**. This can be obtained from plotting the frequency response of the open-loop frequency response transfer function $GH(\omega)$, after putting the feedback system in the form:

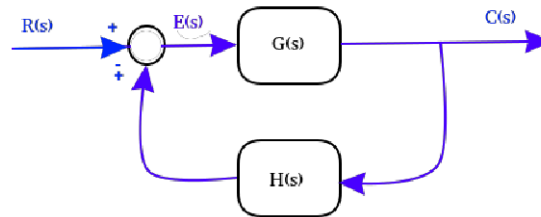


Figure 10- General feedback system

To understand Nyquist method, let us address the topic of **complex functions**.

COMPLEX FUNCTIONS

A real function of a real variable is easily graphed on a single set of coordinate axes. For example, the real function $f(x)$, x real, is easily plotted in rectangular coordinates with x as the abscissa and $f(x)$ as the ordinate. For example, the following is a plot of the function $f(x) = x^2 - x - 2$:

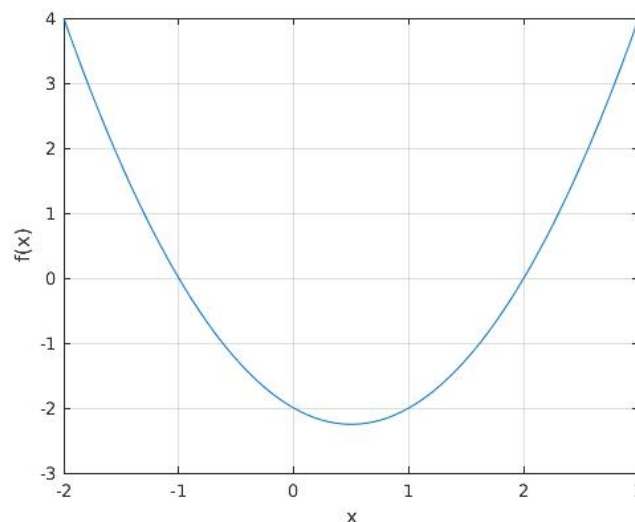


Figure 11- Plot of $f(x)=x^2-x-2$

On the other hand, a **complex function** of a complex variable, such as the transfer function $P(s)$, **cannot be plotted on a single set of coordinates**.

The complex variable $s = \sigma + j\omega$ depends on two independent quantities, the real and imaginary parts of s . Hence s cannot be represented by a line. The complex function $P(s)$ also has real and imaginary parts. It too cannot be graphed in a single dimension.

To plot $P(s)$ with $s = \sigma + j\omega$, **two two-dimensional graphs are required**. The first is a graph of $j\omega$ versus σ called the **s-plane**, the same set of coordinates as those used for plotting pole-zero maps in part (). The second is the imaginary part of $P(s)$ ($\text{Im}\{P\}$) versus the real part of

$P(s)$ ($\text{Re}\{P\}$) called the **P(s)-plane**. As the following:

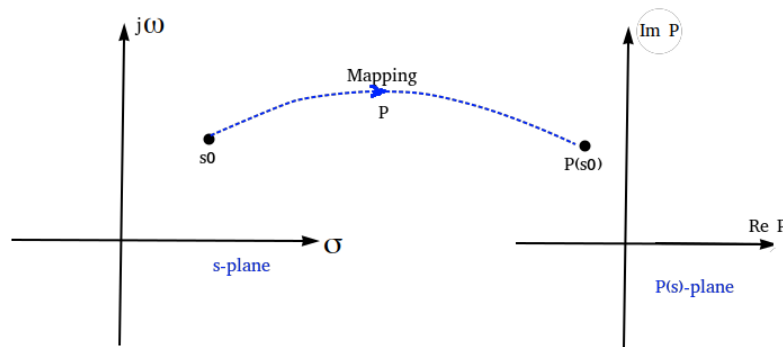


Figure 12- mapping from s-plane to P(s)-plane

This is referred to as **mapping**, as shown with the point s_0 in the s-plane is mapped into a point $P(s_0)$ in the P(s)-plane, by the function P .

EXAMPLE

For the function $P(s) = s^2 + 1$, the point $s_0 = 2 + j4$ is mapped into the point $P(s_0) = P(2 + j4) = (2 + j4)^2 + 1 = -11 + j16$

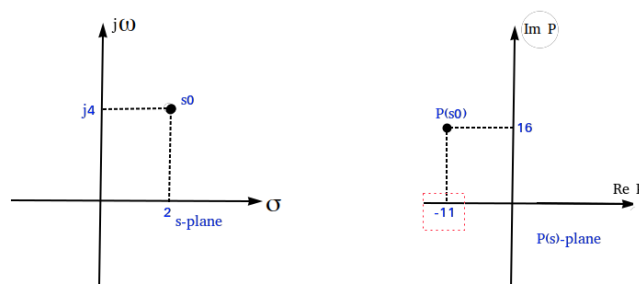


Figure 13- Mapping of the point s_0 into the P(s)-plane

To continue with the theory, let us define some concepts regarding the mapping.

DERIVATIVE

Defined by: $\left(\frac{dP}{ds}\right)_{(s=s_0)} = \lim_{s \rightarrow s_0} \left[\frac{P(s) - P(s_0)}{s - s_0} \right]$, if the **derivative exists** at all points in a region of the s-plane (the limit is finite and unique), then P is **analytic** in that region.

Transfer functions of practical **physical systems**, are **analytic** in the finite s -plane **except at the poles** of $P(s)$.

SINGULAR POINTS

A point at which $P(s)$ is not analytic is a singular point or singularity of $P(s)$. A pole of P is a singular point.

CLOSED CONTOURS

A closed contour in a complex plane is a continuous curve beginning and ending at the same point.

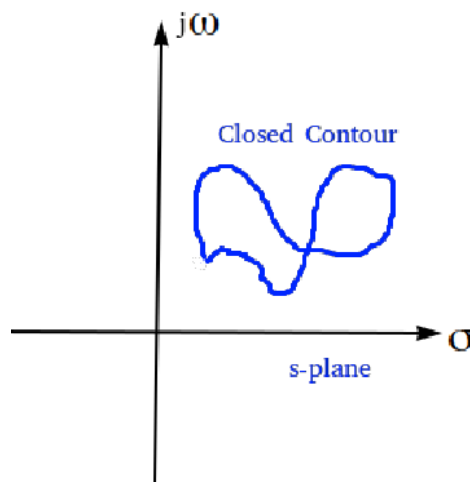


Figure 14- A closed contour

ENCLOSURE

All points to the right of a contour as it is traversed in a prescribed direction are said to be **enclosed** by it.

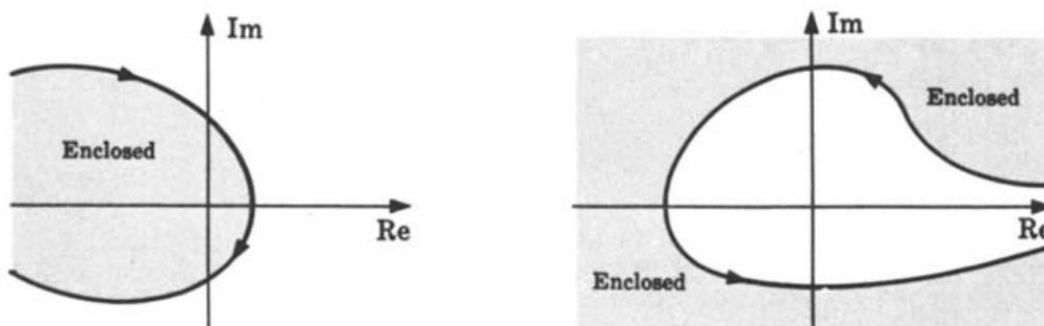


Figure 15- Conversion for Enclosure

CLOCKWISE DIRECTION

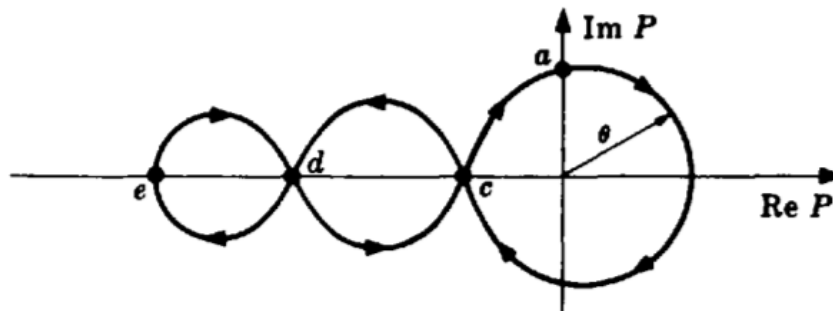
A clockwise (CW) traverse around a contour is defined as the **positive direction**.



Figure 16- CW and CCW directions

ENCIRCLEMENTS

A closed contour in the P -plane is said to make n **positive encirclements** of the origin if a radial line drawn from the origin to a point on the P curve rotates in a clockwise (CW) direction through 360° in completely traversing the closed path. If the path is traversed in a counterclockwise (CCW) direction, a **negative encirclement** is obtained. The total number of encirclements N_0 is equal to the CW minus the CCW encirclements.



PROPERTIES OF THE MAPPING $P(s)$

All mappings $P(s)$ considered here have the following properties:

1. P is a single-valued function. That is, every point in the s -plane maps into one and only one point in the P -plane.
2. s -plane contours avoid singular points of P .
3. P is analytic except possibly at a finite number of points (singularities) in the s -plane.
4. Every closed contour in the s -plane maps into a closed contour in the P -plane.
5. P is a **conformal mapping**. This means that the direction of and the angle between any two intersecting curves at their point of intersection in the s -plane are preserved by the mapping of these curves into the P -plane.
6. The mapping P obeys the **principle of arguments**. That is, the total number of encirclements N_0 of the origin made by a closed contour in the P -plane, mapped from a closed s -plane contour, is equal

to the number of zeros Z_0 minus the number of poles P_0 of P enclosed by the the s-plane (or z-plane) contour. That is,

$$N_0 = Z_0 - P_0$$

7. If the origin is enclosed by the P contour, then $N_0 > 0$. If the origin is not enclosed by the P contour, then $N_0 \leq 0$. That is,

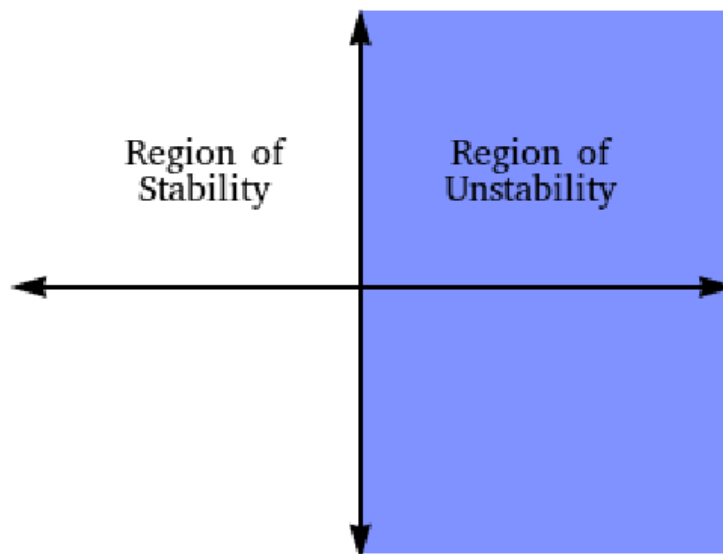
$$enclosed \rightarrow N_0 > 0$$

$$notenclosed \rightarrow N_0 \leq 0$$

The sign of N_0 is easily determined by shading the region to the right of the contour in the prescribed direction. If the origin falls in a shaded region, $N_0 > 0$; if not, $N_0 \leq 0$.

SYSTEMS STABILITY

From the previous part, we reached a conclusion that computing the poles of closed-loop systems gives information about its stability (BIBO stability). Shortly, if there is at least one pole at the right-half of the s-plane, the closed-loop system is considered unstable.



In the following part, we will see how the Polar plot of the open-loop system transfer function can have information about the stability of the closed loop system.

A closed-loop transfer function can be written in the form: $\frac{G(s)}{1+G(s)H(s)}$, see figure 7.

Where $G(s)H(s)$ is called the **open-loop function** (or the Loop function). We can express it as a ratio of polynomials, and its poles and zeros are assumed to be known to us. It can be open-loop stable or unstable, **we are interested in the stability after we close the loop.**

The denominator of the closed loop system can be expressed as: $1 + \frac{N(s)}{D(s)} = \frac{D(s)+N(s)}{D(s)}$, Now

We want to know: **How many zeros of $1 + G(s)H(s)$ are in the right-half of the s-plane?**

To find out, we apply the **Principle of Arguments**, we will perform a mapping through $1 + G(s)H(s)$ for a contour.

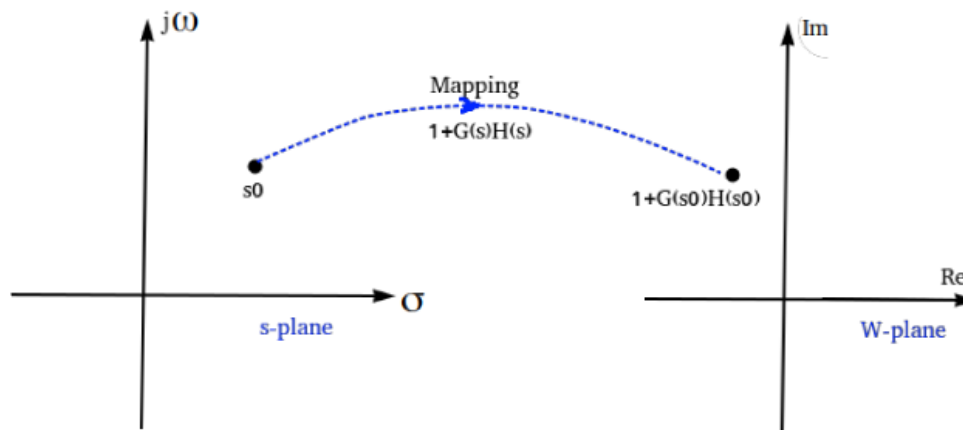
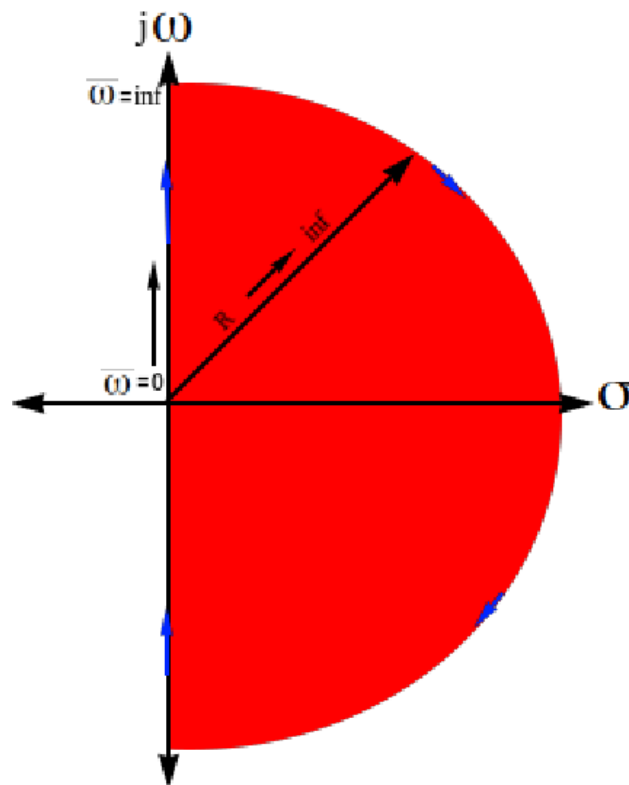


Figure 17- Mapping through $1+GH$

The mapping plane is called the W-plane.

What is the contour we want to map with $1 + G(s)H(s)$?

Of course the answer should be: a contour that encircles the entire right-half of the s-plan.



The contour starts from the origin, goes along the imaginary axis from $\omega = 0$ to $\omega \rightarrow \infty$, then forms half a circle of a radius $R \rightarrow \infty$ to the point $\omega \rightarrow -\infty$, then along the imaginary axis until it reaches the origin.

By applying the **principle of arguments**, we can determine the condition of stability of the closed-loop system. That is, the total number of encirclements N_0 of the **origin** made by the mapping or the contour shown above close in the W-plane, is equal to the number of zeros Z_0 minus the number of poles P_0 of P enclosed by the the s-plane (or z-plane) contour. That is,

$$N_0 = Z_0 - P_0$$

If we selected the open-loop function $G(s)H(s)$ instead of $1 + G(s)H(s)$, the critical point becomes **(-1,0)** instead of the **origin**, since the whole mapped contour is shifted to the left.

Since the contour cannot pass through the open-loop poles, it must **detour around** each open-loop pole lying on its path. As shown below:

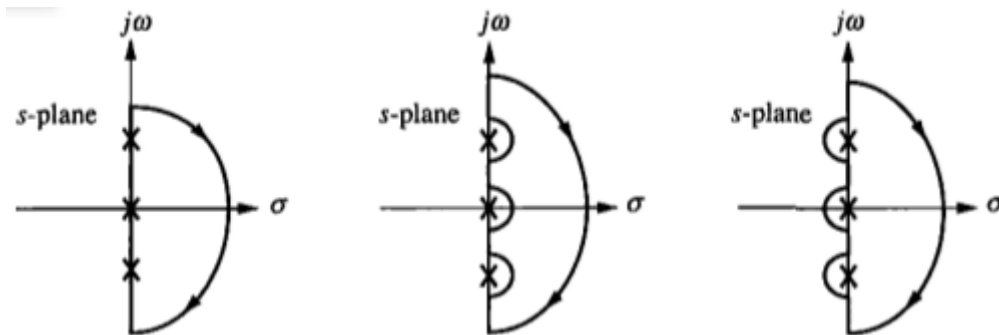


Figure 18- Nyquist contour with points on the path

For **sketching Nyquist plot by hand**, follow the following links and references:

EXAMPLE

For the following system. Plot the Nyquist diagram, and use the criteria to determine whether it is stable or not.

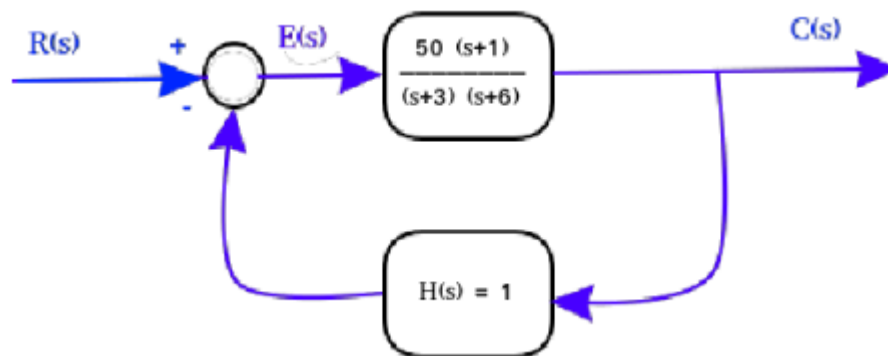


Figure 19- A feedback system

MATLAB CODE

```

Command Window
>> s=tf('s');
>> G=50*(s+1)/((s+3)*(s+6))

G =

    50 s + 50
    -----
    s^2 + 9 s + 18

Continuous-time transfer function.

>> H=tf([1],[1])

H =

    1

Static gain.

>> GH=feedback(G,H)

GH =

    50 s + 50
    -----
    s^2 + 59 s + 68

Continuous-time transfer function.

fx >> nyquist(GH)
  
```

Nyquist Plot for positive frequency values only:

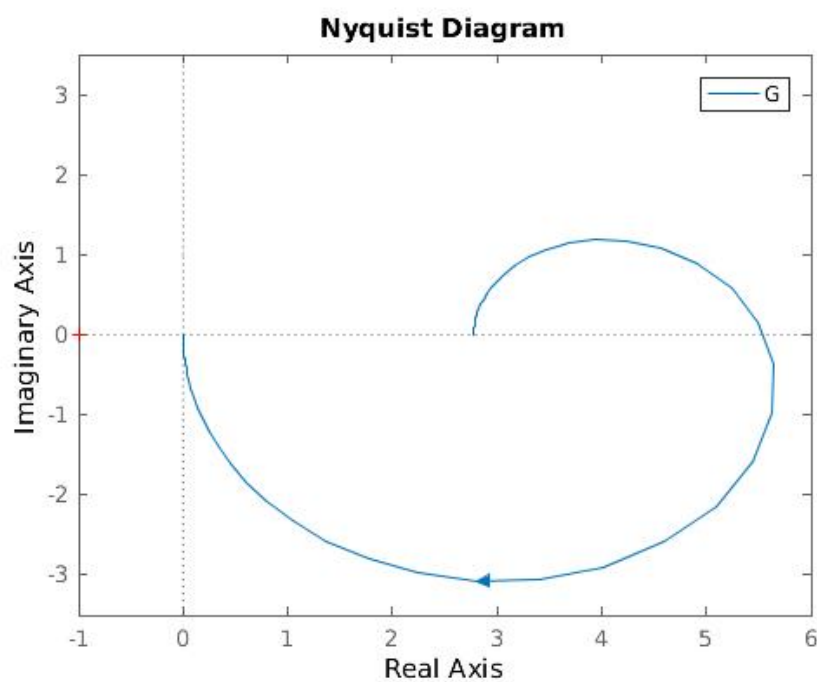


Figure 20- Nyquist diagram for the feedback system in figure 16 (positive frequencies only)

Nyquist Plot for all positive and negative frequency values:

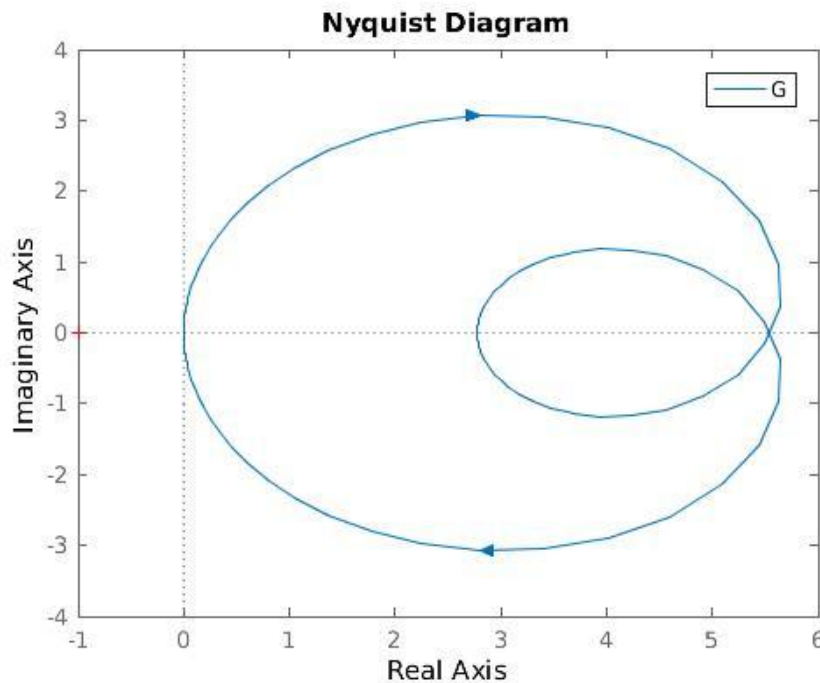


Figure 21- Nyquist diagram for the feedback system of figure 16, (negative and positive frequencies)

Useful links:

- Schaum's Outline of Theory and Problems of Feedback and Control Systems, Joseph J. Distefano, Allen R. Stubberud, Ivan J. Williams, 2nd edition, Chapter 11 Nyquist Analysis. **Contains Many Examples of Sketching Nyquist plot.**

- Control Engineering Lecture, Prof. Madan Gopal, Department of Electrical Engineering, IIT Delhi, NPTEL (India), https://www.youtube.com/watch?v=Rbvau5oXOk&list=PLghJObT_RyfLmKRT86TquJhG6QuiHZ6Pi&index=36

GAIN MARGIN AND PHASE MARGIN

Absolute **stability** means whether system is **stable** or unstable. But **Relative Stability** gives the **degree of stability** or how close it is to instability.

GAIN MARGIN

It is the factor by which the system gain can be increased before the system reaches to the verge of instability.

Gain margin is always calculated at the frequency at which the phase of the system is -180° , also known as phase cross over frequency.

PHASE MARGIN

It is the additional amount of phase lag which can be added to the system before the system reaches to the verge of instability.

Phase margin is always calculated at the frequency at which the gain of the system is 1 or 0 dB, also known as gain cross over frequency.

Evaluating Gain and Phase Margins by Bode plots is shown in figure 18.

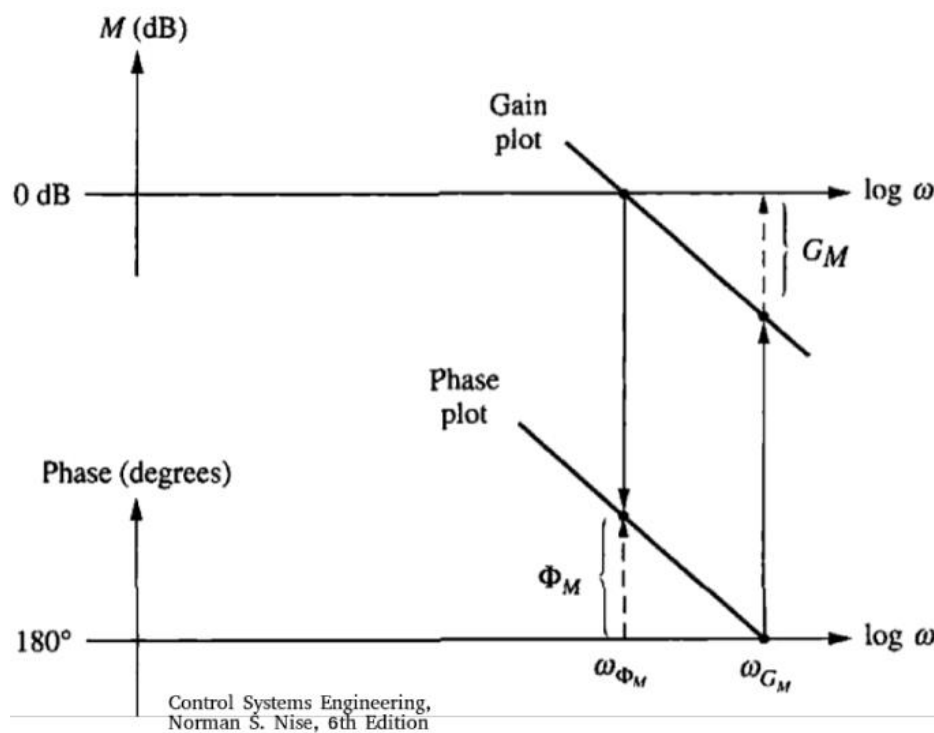


Figure 22- Evaluating Gain and Phase margins

The **gain margin** is found by using the phase plot to find the **frequency ω_{G_M} where the phase is 180°** . At this frequency, we look at the magnitude plot to determine the gain margin, G_M which is the gain required to raise the magnitude curve to 0 dB.

The **phase margin** is found by using the magnitude curve to find **the frequency ω_{Φ_M} , where the gain is 0 dB**. On the phase curve at that frequency, the phase margin ϕ_M is the difference between the phase value and 180° .

EVALUATING GAIN AND PHASE MARGINS BY NYQUIST DIAGRAM

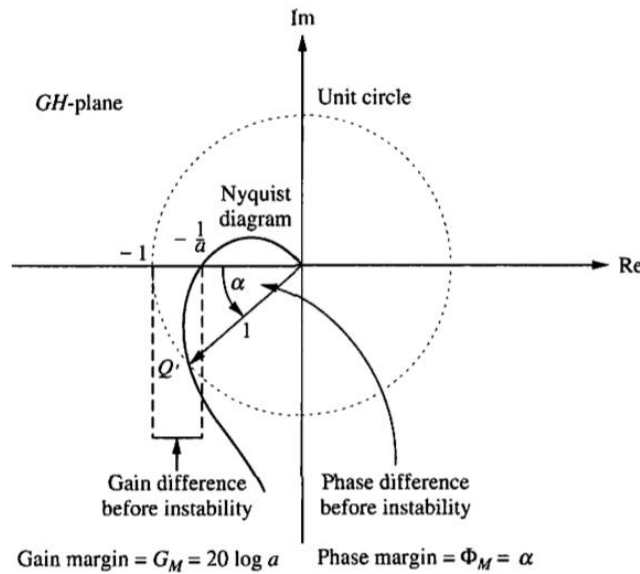


Figure 23- Gain and Phase margins in Nyquist diagram

Assuming the system is stable (no encirclements of -1), the **gain margin** is the difference between the Nyquist diagram's crossing of the **real axis at $-\frac{1}{a}$** and the -1 critical point determines the proximity of the system to instability. Thus, if the gain of the system were multiplied by **a** units, the Nyquist diagram would intersect the critical point. We then say that the gain margin is **a** units, or, expressed in dB, **$G_M = 20 \log a$** . Notice that the gain margin is the reciprocal of the real-axis crossing expressed in dB.

For the phase margin. At point **Q'** , where the gain is unity, a represents the system's proximity to instability. That is, at unity gain, if a phase shift of a degrees occurs, the system becomes unstable. Hence, the amount of phase margin is a .

PERFORMANCE SPECIFICATIONS IN THE FREQUENCY DOMAIN

Given a set of time-domain specifications, how do we specify the frequency response? Consider the 2nd order system which has the following transfer function:

, The corresponding loop function is: $L(s) = \frac{\omega_n^2}{s(s+2\zeta\omega_n)}$, which has the typical plot shown in figure 21.

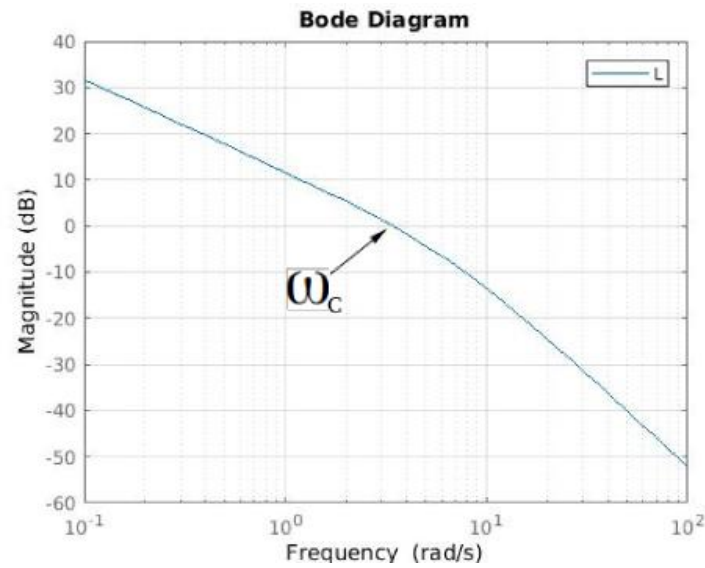


Figure 24- Magnitude of the frequency response of the loop function

The crossover frequency ω_c is expressed as a function of the natural frequency ω_n and the damping ratio ζ : $\omega_c = \omega_n * \sqrt{\sqrt{1 + 4\zeta^4} - 2\zeta^2}$.

Where the closed loop function (referred to as the Complementary Sensitivity Function), $T(s)$, is given by:

$$T(s) = \frac{L}{1+L} = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}, \text{ which has a typical bode plot as shown in figure 22.}$$

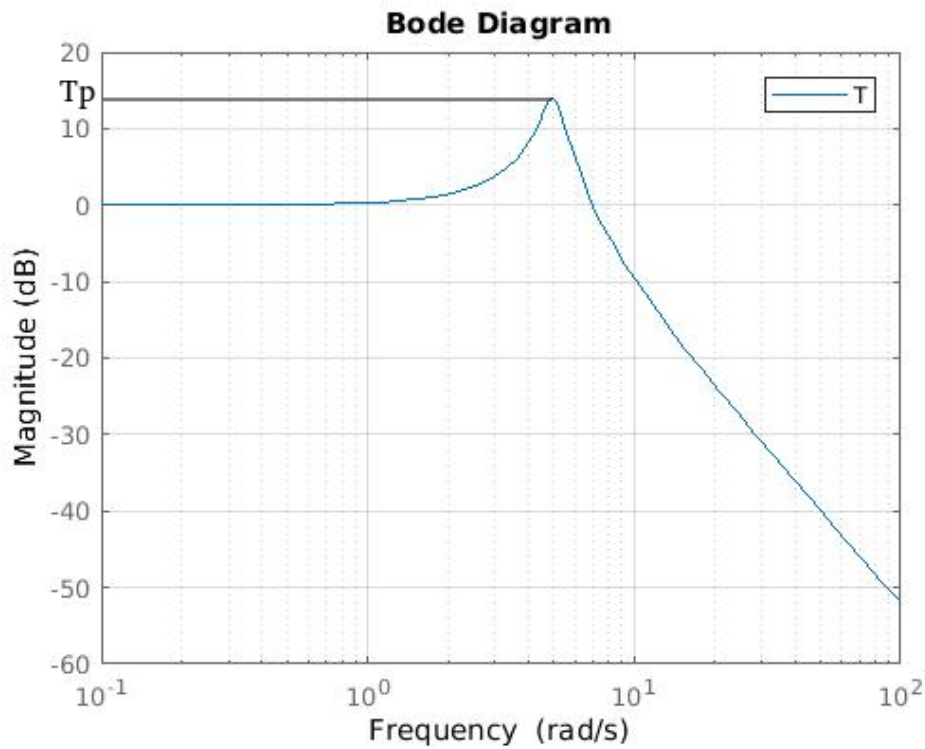


Figure 25- Magnitude of the frequency response of the closed-loop function

At high frequencies $|T(j\omega)| = \left| \frac{L(j\omega)}{1+L(j\omega)} \right| \approx |L(j\omega)|$, since $L(j\omega) \ll 1$.

At low frequencies $|T(j\omega)| = \left| \frac{L(j\omega)}{1+L(j\omega)} \right| \approx 1$, since $L(j\omega) \gg 1$.

And at the middle frequencies (around ω_c), the behaviour is determined by the damping ratio ζ , where the peak value of the sensitivity function is given by $T_p = \frac{1}{2\zeta\sqrt{1-\zeta^2}}$.

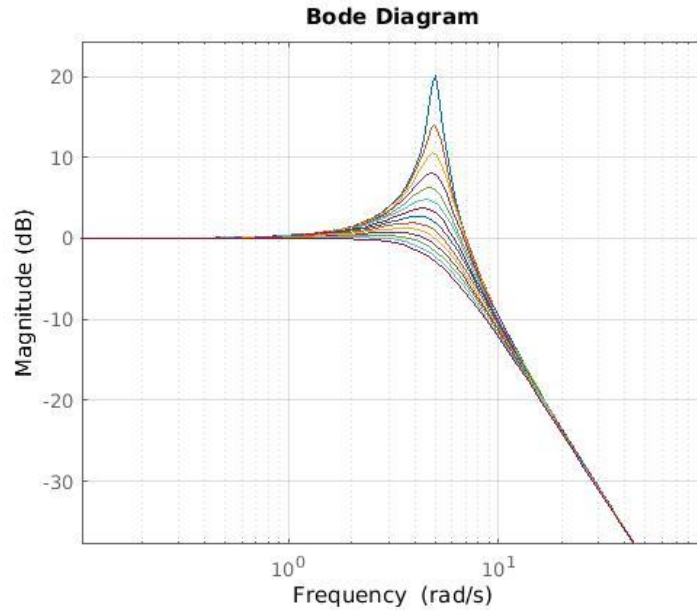


Figure 26- Magnitude of the frequency response of $T(s)$ for different values of the damping factor

The crossover frequency ω_c represents the speed of the system, depending on ω_n and ζ . Which is similar to the resonance frequency ω_R and the bandwidth ω_{BW} of the complimentary sensitivity function.

The bandwidth is given by: $\omega_{BW} = \omega_n \sqrt{1 - 2\zeta^2 + \sqrt{2 - 4\zeta^2 + 4\zeta^4}}$.

CONSTANT MAGNITUDE CIRCLES (M-CIRCLES)

The magnitude of the closed loop frequency response of a unity feedback system is given by:

$$|T(j\omega)| = \left| \frac{L(j\omega)}{1+L(j\omega)} \right|.$$

Plotting lines of constant magnitude in the Nyquist plane gives the M-circles. The intersection of the Polar Plot with a particular M-circle yields the value of M at the frequency of $G(j\omega)$ at the point of

intersection.

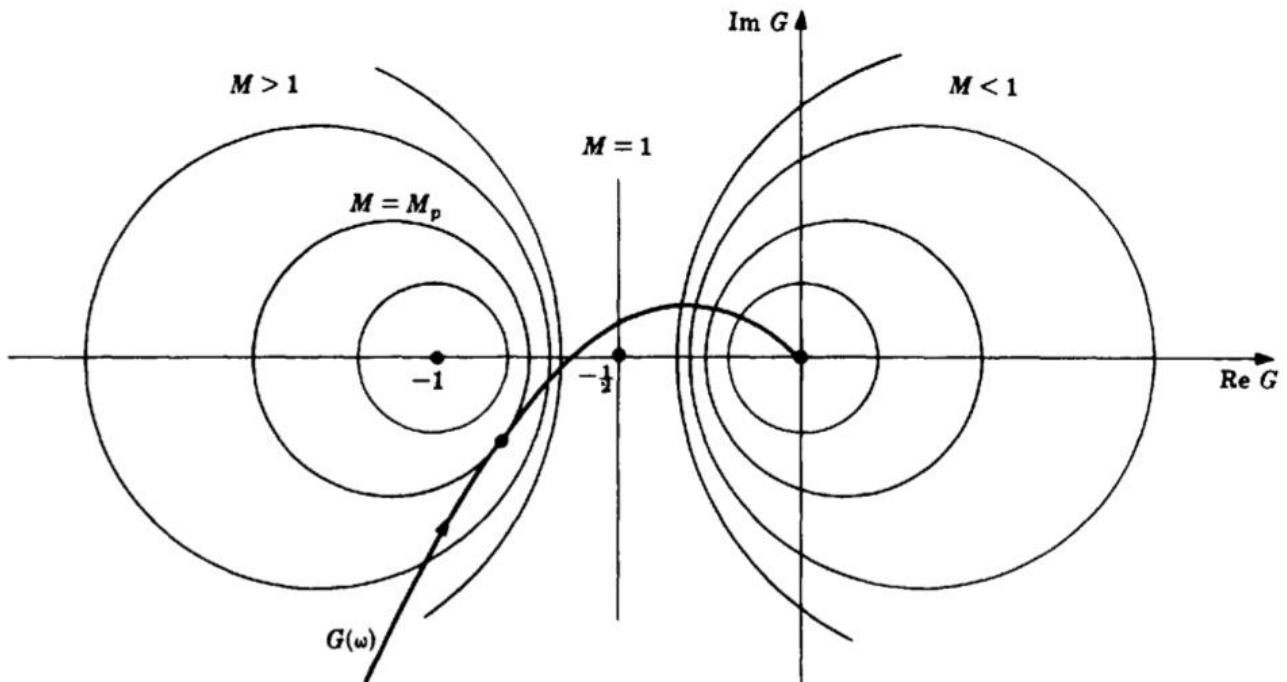


Figure 27- The M-Circles

The radius of an M-circle is given by: $radius = \left| \frac{M}{M^2 - 1} \right|$, and the center is: $center = \left(\frac{-M^2}{M^2 - 1}, 0 \right)$.

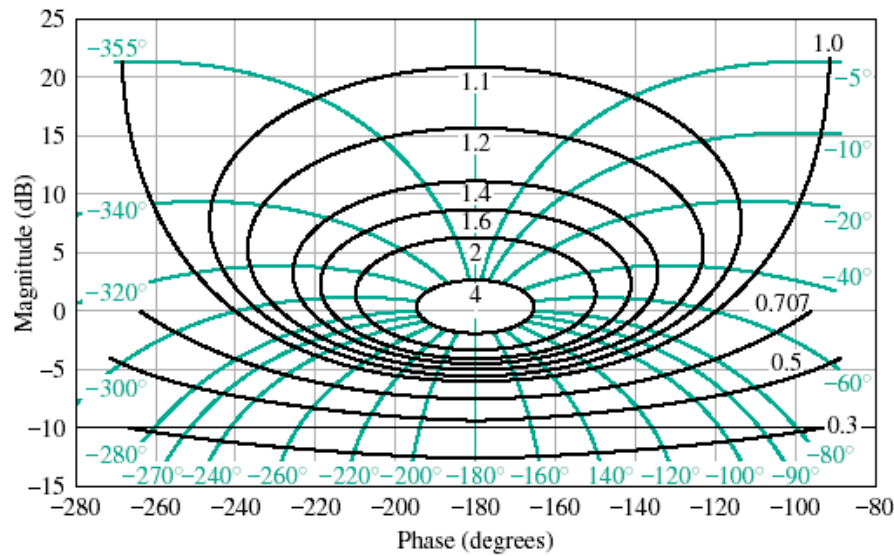
The resonance peak T_p is given by the largest value of M of the M-circles tangent to the polar plot.
 $T_p = M_{max} = M_p$.

NICHOLS CHARTS

It is a modification of the Nyquist and Bode methods. If $GH(\omega)$ represents the open-loop frequency response function of either a continuous-time or discrete-time system, then $GH(\omega)$ plotted on a Nichols chart is a contour on a **(dB) magnitude** versus **phase angle** plot in rectangular coordinates.

M-circles on the Nichols plane:

When the **M-circles** are plotted on the Nichols plane. They are not circles anymore. They become: **Closed curves for $M > 1$, and open curves for $M \leq 1$.**



The black curves are the constant magnitude curves.

PHASE MARGIN AND THE DAMPING FACTOR

For the same General form of the 2nd order system we can find an expression of the phase margin ϕ_M , by finding the frequency for $|G(j\omega)| = 1$, and substituting the phase angle at that frequency. We get: $\phi_M = \tan^{-1} \frac{2\zeta}{\sqrt{-2\zeta^2 + \sqrt{1+4\zeta^4}}}$. Note that the phase margin is a function of the damping ratio only.

The function is plotted in figure 25.

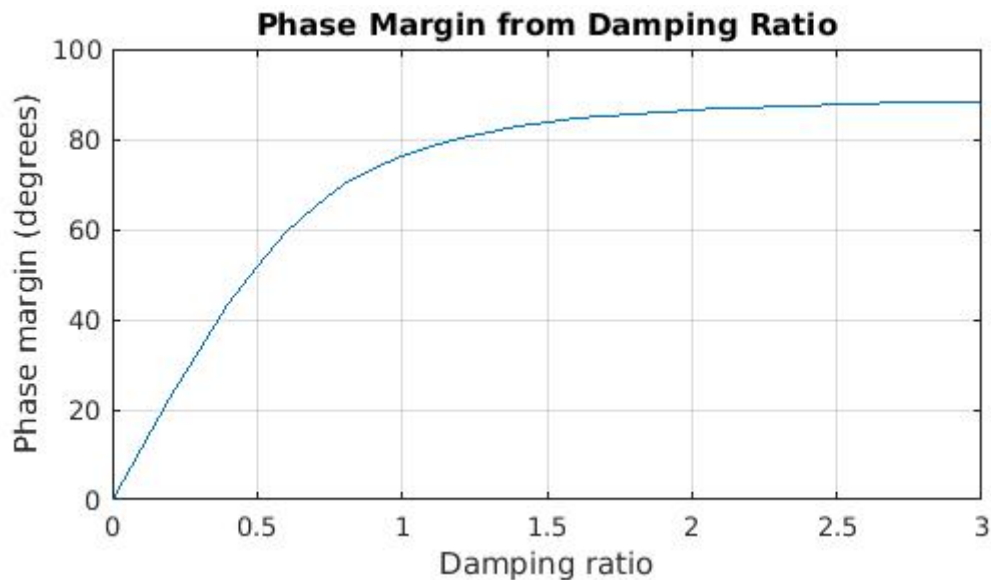


Figure 28- Phase Margin from Damping Ratio

The phase margin is used in control design in frequency domain.

FEEDBACK CONTROL AND ITS ADVANTAGES

A **control system** is an interconnection of components forming a system configuration that will provide a desired system response. It consists of the following parts:

Process: or a **plant**, is the system whose input is manipulated to have its output behave as desired.



Figure 29- The Process

Controller: A system that manipulates the input of the Process to get the output as desired.

Actuator: Usually a power amplifier that takes the small signal coming out of the controller.



Figure 30- Open-Loop Control System

In the above configuration, the controller is the solution to the problem: to get a certain output (**Controlled Variable**), what is the right input (called **Manipulated Variable**) to be produced?

It is easy to notice that to design such a system, knowledge of how the Process behaves is needed, here comes the importance of obtaining the **mathematical model** of the Process, and the Actuator as well.

It is also noted that after building the controller. It will work well if we guarantee that the “reality” is like our mathematical model. But, can we?

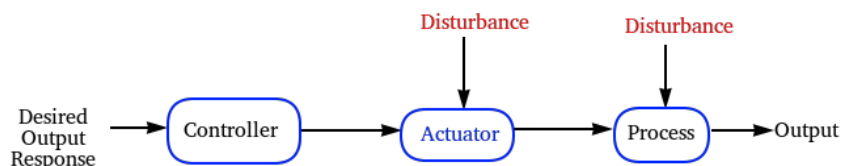


Figure 31- Open-Loop Control System with presence of disturbances

Actually we cannot make sure that our model represents the true system, there will always be deviations from the computed response due to **disturbances** by the surrounding environment. This

deviation can be very large, therefore we need to measure the **Process output** and **feed it back** to the controller.

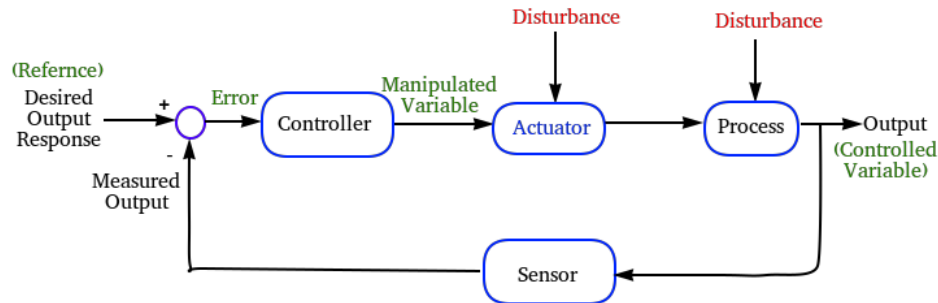


Figure 32- Closed-Loop Control System

The first configuration is called **Open-loop control system**, while the latter is called **Closed-loop control system**. As seen in block diagram, the measured signal is subtracted from the desired signal (the Reference signal), and the control is fed with the error between the two signals.

With the feedback, the control system is able to **reject** or “**attenuate**” the effect of the undesired environment conditions (called the Disturbance).

EXAMPLES OF CONTROL SYSTEMS

- For the following systems write down the general block diagram identifying the main blocks and signals of a control system.

EXAMPLE

Tank Level Control:

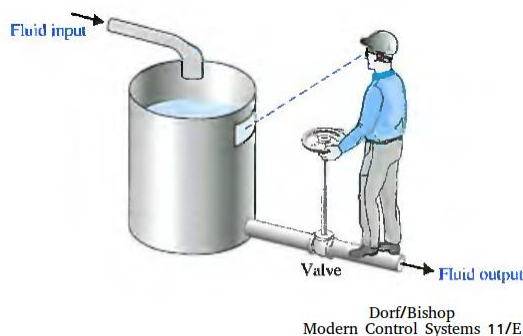


Figure 33- Tank level Control System

This system can be modeled by the following block diagram:

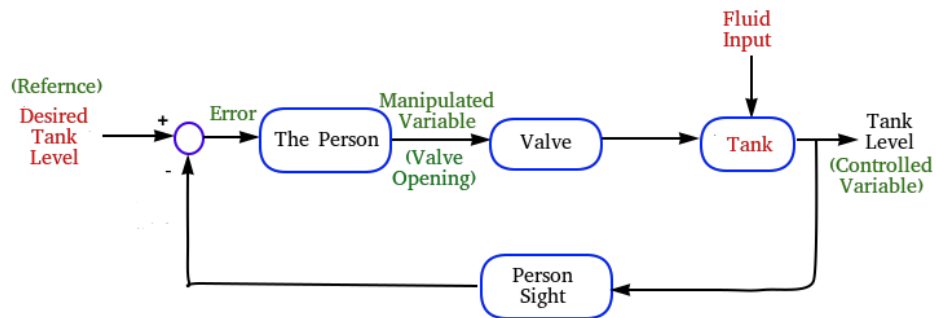


Figure 34- Block Diagram of a Tank Level Control System

In the case the controller is the person (the operator), who keeps monitoring the tank level through the gauge display and changes the valve opening accordingly to allow the right amount of the fluid to drain, in order to keep a certain tank level.

The fluid input is considered as a disturbance because it cannot be changed by the controller and can have any value.

It can be noted that the Open-loop version of this system is equivalent of the operator changes the valve opening without looking at the tank level gauge. Of course he can never tell if the inlet fluid flow increases or decreases.

EXAMPLE

Remote Antenna Positioning system

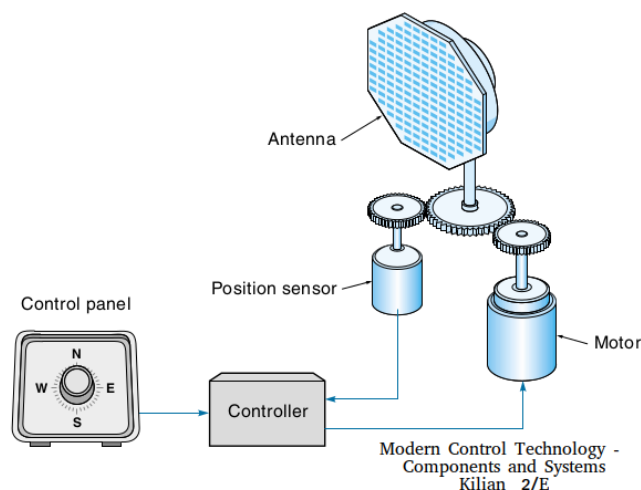


Figure 35- Remote Antenna Positioning system

This system can be modeled by the following block diagram:

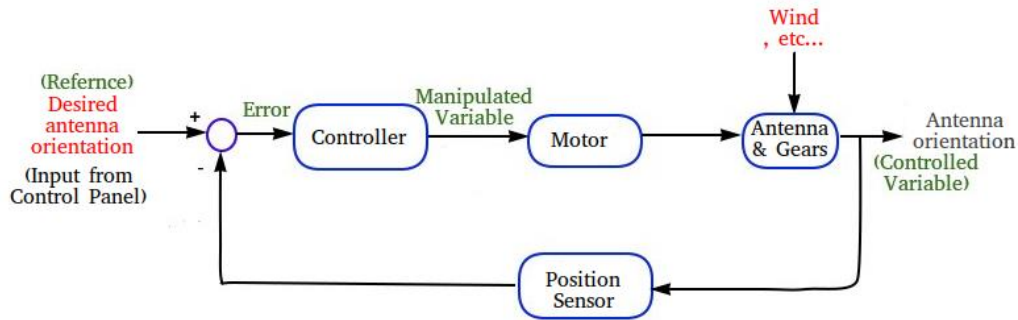


Figure 36- Block diagram of the Remote Antenna Positioning system

In this case the controller can be an electronic circuit, or a digital computer which drives the motor to eliminate “or reduce” any error between the desired and the measured antenna orientation.

The wind or any external effector represent the disturbance that change the antenna orientation undesirably. The position sensor converts continuously the actual orientation and it feeds it to the controller.

PERFORMANCE

The ability to adjust the transient and steady-state performance is a distinct **advantage of feedback control** systems.

Obtaining a system **response that follows the reference** input is not the only objective of introducing a controller. But roughly speaking: “**how fast**” and “**how accurate**” is the response?

Performance criteria are various measurable parameters that indicate how good (or bad) the control system is. These are divided into **transient** (moving) and **steady-state** (not changing) parameters.

The exact path the controlled variable takes when going from one position to the next is called its transient response. Consider the behavior of the antenna in the previous example whose possible response is shown in figure 37.

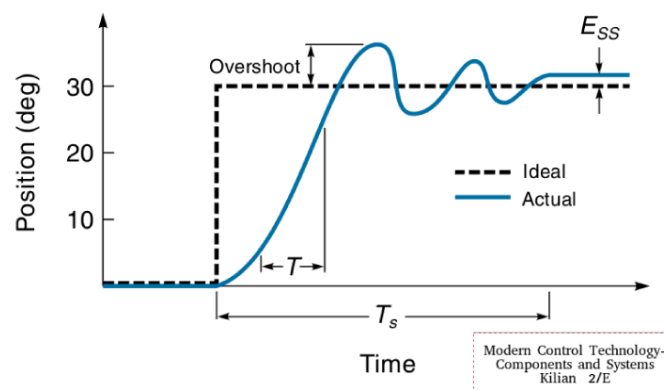


Figure 37- A typical step response

The antenna is directed to move from 0 to 30°, as shown by the dashed line. This type of command (changing instantaneously from one position to another) is called a **step change**. The actual response of the system is shown as a solid line. As you can see, there is a difference between the ideal path of the arm and the one it took. One major consideration is how fast the system picks up speed (called **rise time**). The real arm simply cannot move fast enough to follow the ideal path.

Rise time (T_r) is usually defined as the time it takes for the controlled variable to go from 10 to 90% of the way to its new position. Another transient parameter is **over-shoot**. Once the arm starts moving, its momentum will keep it going right on past where it was supposed to stop. Overshoot can be reduced by the controller but usually at the expense of a longer rise time. **Settling time** (T_s) refers to the time it takes for the response to settle down to within some small percentage (typically 2-5%) of its final value. In this case, it is the time it takes for the oscillations to die out. Rise time, settling time, and overshoot **are all related**; a change in one will cause a change in the others.

P.S.: The criteria mentioned above: Rise time, Settling time and Overshoot are the criteria for a class of linear systems called Second Order systems, which will be treated in part ().

Useful links: Modern Control Systems, Dorf & Bishop 11th edition, Chapter 1

Introduction to Control Systems. Modern Control Technology - Components and Systems, Kilian 2nd edition, Chapter 11 Feedback Control Principles.

FIRST ORDER SYSTEMS

The transfer function of a first order system can be written as: $G(s) = \frac{K}{\tau s + 1}$, where K is the steady-state gain of the system, and τ is the time constant of the system. $\tau = \frac{1}{a}$. The following figure is the step response for the case $K=1$.

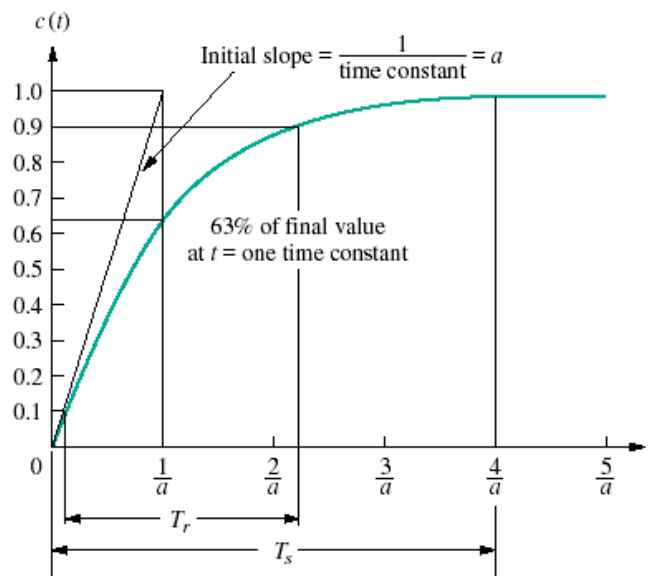


Figure 38- Step response of a 1st order system

As shown in the figure, **to compute the time constant**, draw a tangent of the output (at $t=0$), the time of the intersection with the steady state value (1.0 in this case) gives the value of the time constant. Or it is the time for the output response to reach 63% of the final value.

The time constant represents the speed of the response. And it is equal to the reciprocal of the pole location ($-a$).

SECOND ORDER SYSTEMS

From Linear Systems Theory, the response of a 2nd order system can be one of the following:

1- **Over-damped response**, with two real (distinct) poles $-\sigma_1, -\sigma_2$, and a natural response of two exponentials with time constants equal to the reciprocal of the pole locations.

$$c(t) = K_1 e^{-\sigma_1 t} + K_2 e^{-\sigma_2 t}$$

2- **Underdamped response**, with two complex poles $-\sigma_d \pm j\omega_d$, and a natural response of a damped sinusoid with an exponential envelope whose time constant is equal to the reciprocal of the pole's real part, and the radian frequency of the sinusoid (the damped frequency of oscillation) is equal to the imaginary part of the poles.

$$c(t) = A e^{-\sigma_d t} \cos(\omega_d t - \phi)$$

3- **Un-damped response**, with two imaginary poles at $\pm j\omega_1$, and a natural response of a sinusoid with radian frequency equal to the imaginary part of the poles.

$$c(t) = A \cos(\omega_1 t - \phi)$$

4- **Critically damped response**, with two identical poles at $-\sigma_1$, and a natural response of an exponential whose time constant is equal to the reciprocal of the pole location, and another term which is the product of time, t , and an exponential with time constant equal to the reciprocal of the pole location.

$$c(t) = K_1 e^{-\sigma_1 t} + K_2 t e^{-\sigma_2 t}$$

All the step responses are plotted below.

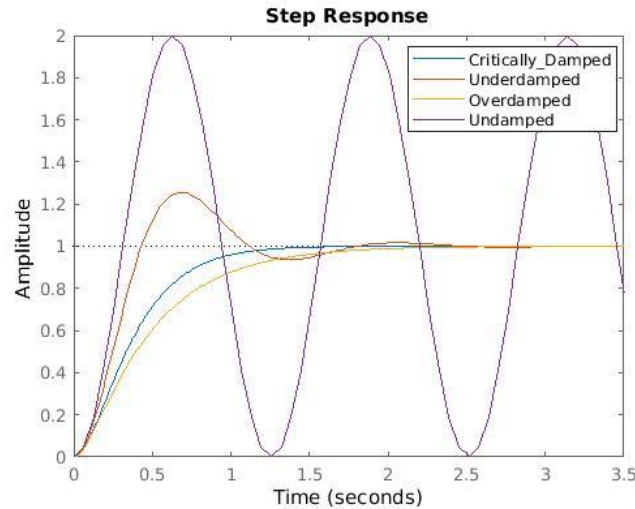


Figure 39- Types of responses of 2nd order systems

THE GENERAL SECOND ORDER SYSTEM

To find quantitative measures of the **Rise Time**, **Settling Time** and the **Percentage Overshoot**. We will describe the response of the 2nd order system using a general form.

The general form consists of two quantities that define the behavior: the **natural frequency** ω_n and the **damping ratio** ζ .

NATURAL FREQUENCY

It is the frequency of oscillation of the system without damping.

DAMPING RATIO

It is a measure of how much damped the response is. Specifically: Un-damped $\zeta = 0$, Underdamped $0 < \zeta < 1$, Critically damped $\zeta = 1$ and Overdamped $\zeta > 1$.

The general form is: $G(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}$, it has the poles $-\zeta\omega_n \pm \omega_n\sqrt{\zeta^2 - 1}$, analyzing the poles with respect to the value of ζ gives rise to the result stated previously.

UNDER-DAMPED SECOND ORDER SYSTEM RESPONSE

The under-damped second order system is a common model for physical problems.

The transient specifications are associated with underdamped responses.

Percentage Overshoot %OS : The amount that the waveform overshoots the steady-state, or final, value at the peak time, expressed as a percentage of the steady-state value

%OS is a function only of the damping ratio ζ , it is given by:

$$\%OS = e^{-\left(\frac{\zeta\pi}{\sqrt{1-\zeta^2}}\right)} \times 100, \text{ the inverse relation is given by:}$$

$$\zeta = \frac{-\ln(\%OS/100)}{\sqrt{\pi^2 + \ln^2(\%OS/100)}}$$

Rise Time T_r : The time required for the waveform to go from 0.1 of the final value to 0.9 of the final value (**or from 0 to the final value**)

T_r is a function of ζ and ω_n .

$$T_r(0 \rightarrow final) = \frac{\pi - \arccos(\zeta)}{\omega_n \sqrt{1 - \zeta^2}}$$

Settling Time T_s : The time required for the transient's damped oscillations to reach and stay within $\pm 2\%$ of the steady-state value (**or within $\pm 5\%$ of the steady-state value**)

T_s is a function of ζ and ω_n too.

$$T_s(\alpha\%) = -\frac{\ln \alpha}{\omega_n \zeta}$$

BLOCK DIAGRAMS

In part (3), it was shown that LTI systems modeled by **transfer functions**, and a transfer function is represented by a block with its input and output, usually systems are represented by an **interconnection of many subsystems**. To compute the response of this system, a single equivalent transfer function can be computed and then its response can be obtained.

There are some techniques that can be applied to reduce an interconnection of blocks into a single block.

First, we examine the main elements of block diagrams (in addition to blocks that represent systems!):

Signals: Either an input or an output, represented by arrows.



Figure 40- Signals representation

Summing Junctions: As the name implies, it takes the input signals and produce an output signal represents the arithmetic sum of the inputs.

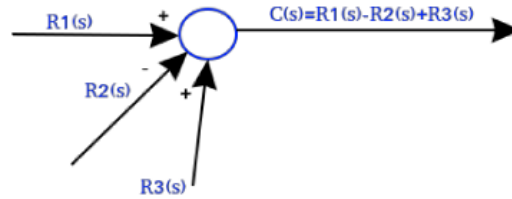


Figure 41- Summing junctions representation

Pickoff points: Which distributes one input signal to multiple output signals.

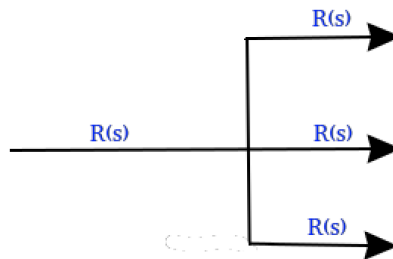


Figure 42- Pickoff point representation

COMMON BLOCK DIAGRAMS CONFIGURATIONS

CASCADE FORM

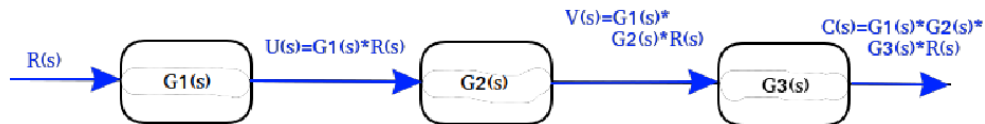


Figure 43- Cascade Form

And this is equivalent to:

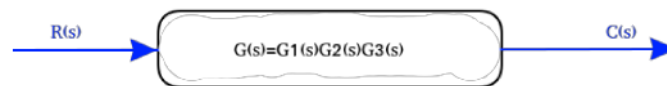


Figure 44- Equivalent block to cascaded blocks

PARALLEL FORM

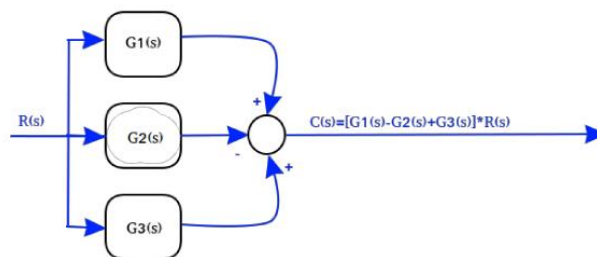


Figure 45- Parallel Form

And this is equivalent to:

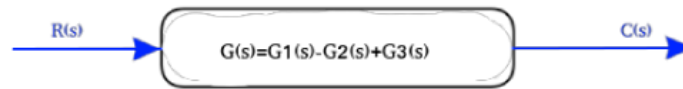


Figure 46- Equivalent Block to a parallel form

FEEDBACK FORM

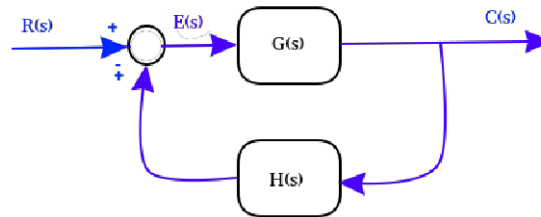


Figure 47- Feedback form

To compute the relation between $C(s)$ and $R(s)$, first we compute the relation between $C(s)$ and $E(s)$:

$$E(s) = R \mp H(s) * C(s)$$

$$C(s) = G(s) * [R(s) \mp H(s) * C(s)] = G(s) * R(s) \mp G(s) * H(s) * C(s)$$

Rearranging the equation:

$$(1 \pm G(s) * H(s)) * C(s) = G(s) * R(s)$$

$$C(s) = \frac{G(s)}{1 \pm G(s) * H(s)} * R(s)$$

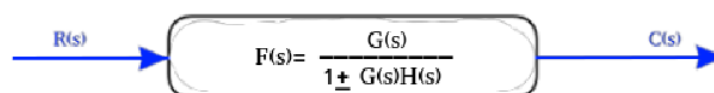


Figure 48- A block equivalent to a feedback form

Please refer to the links for examples of block diagram reduction of complex systems.

Useful links:

Control Systems Engineering, Norman S. Nise, 6th edition, Chapter 5 Reduction of Multiple Systems.
Further Explanation on: <https://www.electrical4u.com/block-diagrams-of-control-system/>
Examples on: <http://www.msubbu.in/sp/ctrl/BD-Quest.htm>

EXAMPLES OF CONTROL SYSTEMS DESIGN

In Controller Design, we must consider:

- Compensate the effect of load disturbances.
- Reduce the effect of measurement noise.
- Setpoint following (target tracking).
- Inaccuracies in the process model and parameter variations.

To do this we must consider:

- Purpose of the control system.
- Process model.
- Disturbance model.
- Model inaccuracies and changes.
- Applicable control strategies.
- Design parameters.

Many methods of control design were developed, in the following section, examples of “**State Feedback**”, “**PID**” and “**Lead-Lag Compensators**” control design methods are demonstrated.

STATE FEEDBACK

This method is also called “**Pole Placement**” design. The idea is to control the location of all closed-loop poles.

The Process to be controlled is described by a state-space model. The CT model is described in the form:

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

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

And the DT equivalent model is given by:

$$x(k+1) = Ax(k) + Bu(k)$$

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

The following figure shows the block diagram of the system:

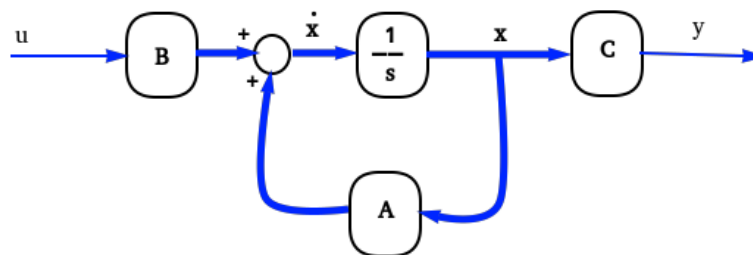


Figure 49- State-space representation

Where the light lines are scalars and the heavy lines are vectors. For simplicity: $u, y \in \mathbb{R}$ and $D = 0$.

In a typical feedback control system, the output y is fed back to the summing junction.

In “**State Feedback**” as the name implies, **all the state variables x are fed back to the control input u** , through a gain k_i , therefore the input becomes: $u = -Kx + r$ in this case \mathbf{K} is a vector $K \in \mathbb{R}^n$. Thus we obtain:

$$\dot{x} = Ax + Bu = Ax + B(-Kx + r) = (A - BK)x + Br$$

Now r is the new input, which in this case is considered the desired (or reference) input.

The result is shown the following diagram.

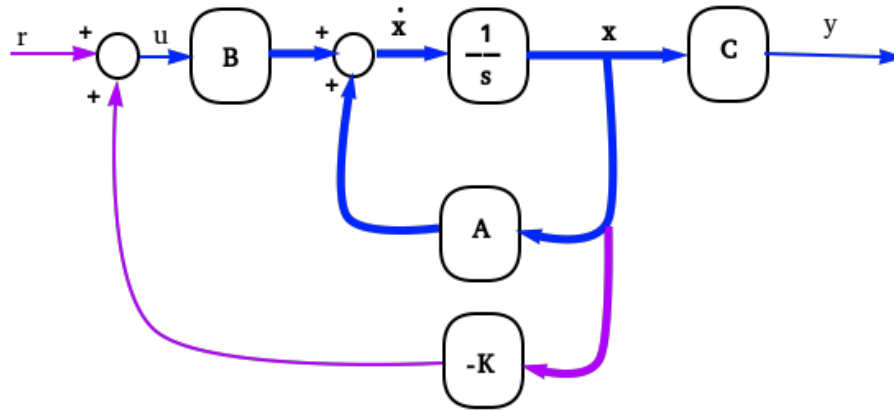


Figure 50- State-Feedback control

The closed loop system is a **state matrix** ($A-BK$) instead of just A . since its response is given by the location of its poles. Which are the roots of its characteristic equation, that is $\det[sI-(A-BK)]=0$

The values of K are selected to obtain the desired poles locations. The question is: **Is it always possible?** let's have some examples.

EXAMPLE

Find a state feedback gain K that modifies the following system:

$$\dot{x} = \begin{bmatrix} 1 & 2 & -1 \\ 0 & -1 & 0 \\ 1 & 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ -1 \\ 0 \end{bmatrix} u$$

To have its closed-loop poles at $(-5+j7, -5-j7, -7)$.

In this case $K = [k_1 \quad k_2 \quad k_3]$, So computing the closed-loop state matrix $A-BK$ gives:

$$A - BK = \begin{bmatrix} 1 & 2 & -1 \\ 0 & -1 & 0 \\ 1 & 0 & 0 \end{bmatrix} - \begin{bmatrix} 0 \\ -1 \\ 0 \end{bmatrix} [k_1 \quad k_2 \quad k_3] = \begin{bmatrix} 1 & 2 & -1 \\ k_1 & -1+k_2 & k_3 \\ 1 & 0 & 0 \end{bmatrix}$$

To obtain the characteristic equation: consider the following matrix:

$$sI - (A - BK) = \begin{bmatrix} s-1 & -2 & 1 \\ 0 & s+1 & 0 \\ -1+k_1 & k_2 & s+k_3 \end{bmatrix}$$

Then the determinant is given by:

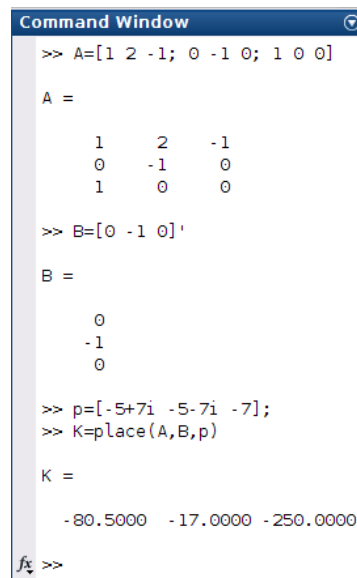
$$\det(sI - (A - BK)) = s^3 - k_2 s^2 + (k_2 - 2k_1)s + (1 + 2k_3 - k_2) = 0$$

The desired characteristic equation given the desired poles: $(s + 5 - j7)(s + 5 + j7)(s + 7) = 0$ Which gives:

$$s^3 + 17s^2 + 144s + 518 = 0$$

By equating the two left-hand sides we find that: $k_2 = -17, k_1 = \frac{-(144-17)}{2} = -80.5, k_3 = \frac{-(518+k_2-1)}{2} = -250$ enable us to have the characteristics equation similar to the desired one.

MATLAB CODE



```

Command Window
>> A=[1 2 -1; 0 -1 0; 1 0 0]

A =

     1     2    -1
     0    -1     0
     1     0     0

>> B=[0 -1 0]';

B =

     0
    -1
     0

>> p=[-5+7i -5-7i -7];
>> K=place(A,B,p)

K =

   -80.5000   -17.0000  -250.0000

fx >>
  
```

Now, Check this case where we slightly modify B.

EXMAPLE

Find a state feedback gain **K** that modifies the following system:

$$\dot{x} = \begin{bmatrix} 1 & 2 & -1 \\ 0 & -1 & 0 \\ 1 & 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} u$$

To have its closed-loop poles at **(-5+j7, -5-j7, -7)**.

In this case $K = [k_1 \ k_2 \ k_3]$, So computing the closed-loop state matrix **A-BK** gives:

$$A - BK = \begin{bmatrix} 1 & 2 & -1 \\ 0 & -1 & 0 \\ 1 & 0 & 0 \end{bmatrix} - \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} [k_1 \ k_2 \ k_3] = \begin{bmatrix} 1 & 2 & -1 \\ 0 & -1 & 0 \\ 1-k_1 & -k_2 & -k_3 \end{bmatrix}$$

To obtain the characteristic equation: consider the following matrix:

$$sI - (A - BK) = \begin{bmatrix} s-1 & -2 & 1 \\ 0 & s+1 & 0 \\ -1+k_1 & k_2 & s+k_3 \end{bmatrix}$$

Then the determinant is given by:

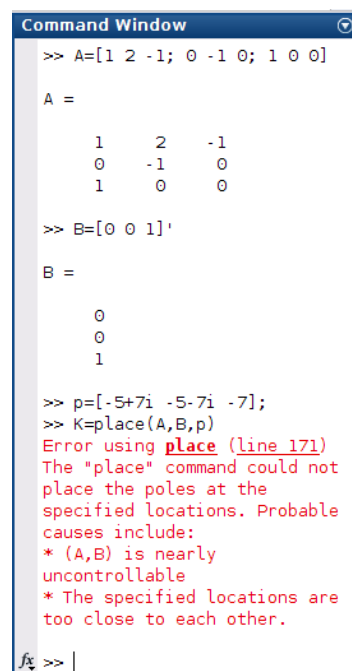
$$\det(sI - (A - BK)) = s^3 + k_3s^2 + (k_1 - 2)s + (k_1 - k_3 - 1) = 0$$

The desired characteristic equation given the desired poles: $(s + 5 - j7)(s + 5 + j7)(s + 7) = 0$ Which gives:

$$s^3 + 17s^2 + 144s + 518 = 0$$

By equating the two left-hand sides we obtain: $k_3 = 17, k_1 = 144 + 2 = 146$ but this enforces the last term to be $(146 - 17 - 1) = (128)$, which is not (518). Therefore, it is not possible to locate all the poles of this system in the desired locations.

MATLAB CODE



```

>> A=[1 2 -1; 0 -1 0; 1 0 0]

A =

     1     2    -1
     0    -1     0
     1     0     0

>> B=[0 0 1]';

B =

     0
     0
     1

>> p=[-5+7i -5-7i -7];
>> K=place(A,B,p)
Error using place (line 171)
The "place" command could not
place the poles at the
specified locations. Probable
causes include:
* (A,B) is nearly
uncontrollable
* The specified locations are
too close to each other.

```

The error message says that the system is “**nearly uncontrollable**”, so what does it mean to have a controllable system, and how can we test the controllability of the system?

CONTROLLABILITY

The problem of controllability is to find out whether or not we are **able to drive the system to a desired state with an input of finite duration** (a sequence of inputs of finite length for discrete time systems).

Since the problem of controllability is rather complicated, we introduce it by means of a discrete time system. Given an initial state $\mathbf{x}(0)$ and a scalar input sequence of length n , then the sequence of the states will be:

$$\begin{aligned}
 \mathbf{x}(1) &= \mathbf{A}\mathbf{x}(0) + \mathbf{B}\mathbf{u}(0) \\
 \mathbf{x}(2) &= \mathbf{A}\mathbf{x}(1) + \mathbf{B}\mathbf{u}(1) = \mathbf{A}^2\mathbf{x}(0) + \mathbf{A}\mathbf{B}\mathbf{u}(0) + \mathbf{B}\mathbf{u}(1) \\
 &\dots \\
 \mathbf{x}(n) &= \mathbf{A}^n\mathbf{x}(0) + \mathbf{A}^{n-1}\mathbf{B}\mathbf{u}(0) + \dots + \mathbf{B}\mathbf{u}(n-1)
 \end{aligned}$$

Taking the higher power term of A on the left side we can write:

$$x(n) - A^n x(0) = \mathcal{C} \begin{pmatrix} u(n-1) \\ u(n-2) \\ \dots \\ u(0) \end{pmatrix}$$

where the matrix \mathcal{C} is called **controllability matrix**, a square matrix having for each column one of the terms $A^k B$. If the controllability matrix is **nonsingular**, one can invert the system and find the input sequence such that the desired states are reached. This result can be extended to input vectors and continuous time systems.

EXAMPLE

Is the following system controllable?

$$\dot{x} = \begin{bmatrix} 1 & 2 & -1 \\ 0 & -1 & 0 \\ 1 & 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ -1 \\ 0 \end{bmatrix} u$$

We already know the result since we were able to find state feedback gain \mathbf{K} able to place the poles to the desired places. But let's verify that using the controllability matrix \mathbf{C} .

$C = [B \quad AB \quad A^2B] = \begin{bmatrix} 0 & -2 & 0 \\ -1 & 1 & -1 \\ 0 & 0 & -2 \end{bmatrix}$ the columns of this matrix are linearly independent, therefore it is a full-rank matrix. Thus, the system is fully controllable.

EXAMPLE

Is the following system controllable?

$$\dot{x} = \begin{bmatrix} 1 & 2 & -1 \\ 0 & -1 & 0 \\ 1 & 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ -1 \\ 0 \end{bmatrix} u$$

We already know the result since we were able to find state feedback gain \mathbf{K} able to place the poles to the desired places. But let's verify that using the controllability matrix \mathbf{C} .

$C = [B \quad AB \quad A^2B] = \begin{bmatrix} 0 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 0 & -1 \end{bmatrix}$ the columns of this matrix are linearly dependent, therefore it is a rank deficient matrix. Thus, the system is not fully controllable. Specifically, rank $\mathbf{C}=2$, this implies that **only two of the natural modes are controllable**.

Now that we defined the condition necessary to control all the natural modes the plant. One observation about this method arises. **Can we measure the states and feed them back to the controller?** Mostly no, since the states are internal variables, either their access is not possible, or it is expensive to physically measure them. The solution is within the following section.

OBSERVER DESIGN

Usually it is impractical to measure all the state, but it is possible to **estimate** them.

The **Observer**, or the **Estimator** is a dynamical system that calculates the state variables for the **input** and **output data** of the plant. And the **model** of the plant.

Block diagram of state observation:

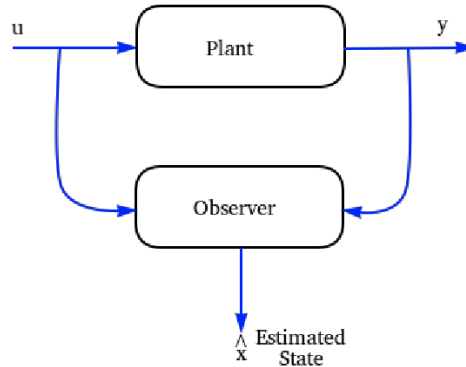


Figure 51- Observer topology

We will consider the method of **Luenberger** in observer design:

Consider a plant with the following state-space model:

$$\dot{x} = Ax + Bu$$

$$y = Cx$$

Writing the observer in the following form:

$$\dot{\hat{x}} = A\hat{x} + Bu + L(y - \hat{y})$$

$$\hat{y} = C\hat{x}$$

Now let us subtract the state equations of the plant and the observer:

$$(\dot{x} - \dot{\hat{x}}) = A(x - \hat{x}) - L(y - \hat{y})$$

$$(y - \hat{y}) = C(x - \hat{x})$$

Which describes the **error between the real and estimated states** and the outputs. We can substitute the output equation in the state equation:

$$(\dot{x} - \dot{\hat{x}}) = (A - LC)(x - \hat{x}) \rightarrow \dot{e}_x = (A - LC)e_x$$

$$(y - \hat{y}) = C(x - \hat{x}) = Ce_x$$

Now we got the **error dynamics**, we just need to make sure that it will converge to zero as the time evolves. For achieving this we can choose the values of design variable **L** so that the **eigenvalues** of the state matrix **(A-LC)** have **negative real parts**. In other words, to make it **asymptotically stable**.

We also need the observation error to converge **faster** than the closed-loop response.

As in the case in state feedback, **is it always possible to estimate the select L such that (A-LC) have eigenvalues of our choice?**

EXMAPLE

Design a suitable observer for the following plant:

$$\dot{x} = \begin{bmatrix} 1 & 2 & -1 \\ 0 & -1 & 0 \\ 1 & 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ -1 \\ 0 \end{bmatrix} u, \text{ and}$$

$$y = [0 \quad 0 \quad 1]x$$

Solution: let us assume that an observer with the following eigenvalues of (A-LC), is the one we look for, which corresponds to the following characteristic equation:

$$s^3 + 18s^2 + 107s + 210 = 0$$

let us compute the matrix (A-LC):

$$A - LC = \begin{bmatrix} 1 & 2 & -1 \\ 0 & -1 & 0 \\ 1 & 0 & 0 \end{bmatrix} - \begin{bmatrix} l_1 \\ l_2 \\ l_3 \end{bmatrix} [0 \quad 0 \quad 1] = \begin{bmatrix} 1 & 2 & -1 - l_1 \\ 0 & -1 & -l_2 \\ 1 & 0 & -l_3 \end{bmatrix}, \text{ then:}$$

$$sI - (A - LC) = \begin{bmatrix} s - 1 & -2 & 1 + l_1 \\ 0 & s + 1 & l_2 \\ -1 & 0 & s + l_3 \end{bmatrix}$$

The characterstic equation $\det[sI - (A - LC)] = s^3 + l_3s^2 + l_1s + (2l_2 - l_3 + l_1 + 1) = 0$

With this, we can easily substitute: $l_3 = 18, l_1 = 107, 2l_2 - l_3 + l_1 + 1 = 210 \rightarrow l_2 = 60$

$$L = [107 \quad 60 \quad 18]^T$$

MATLAB CODE

Now let us check this case.

EXAMPLE

Design a suitable observer for the following plant:

$$\dot{x} = \begin{bmatrix} 1 & 2 & -1 \\ 0 & -1 & 0 \\ 1 & 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ -1 \\ 0 \end{bmatrix} u, \text{ and}$$

$$y = [0 \quad 3 \quad 0]x$$

Solution: let us assume that the desired eigenvalues are same as the previous example, which gives the following characteristic equation:

$$s^3 + 18s^2 + 107s + 210 = 0$$

let us compute the matrix (A-LC):

$$A - LC = \begin{bmatrix} 1 & 2 & -1 \\ 0 & -1 & 0 \\ 1 & 0 & 0 \end{bmatrix} - \begin{bmatrix} l_1 \\ l_2 \\ l_3 \end{bmatrix} \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 2 & -1-l_1 \\ 0 & -1 & -l_2 \\ 1 & 0 & -l_3 \end{bmatrix}, \text{ then:}$$

$$sI - (A - LC) = \begin{bmatrix} s-1 & -2 & 1+l_1 \\ 0 & s+1 & l_2 \\ -1 & 0 & s+l_3 \end{bmatrix}$$

The characteristic equation $\det[sI - (A - LC)] = s^3 + 3l_2s^2 - 3l_2s + 3l_2 = 0$

With this, it is impossible to select a value of l_2 that makes all the 3 terms equal to the desired equation. It is up to the original matrices **A** and **C** to decide how the estimation error evolves with time.

OBSERVABILITY

The concept of Observability is related to our capabilities to **monitor** the evolution of a dynamical system **given the values of the output** that we get from the system itself.

Once we know the initial conditions for the states of a system, $\mathbf{x}(0)$ we can obtain the system state at every time instant t by means of the matrix exponential:

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

In few words, we can say that a system is observable if we can get the value of the initial conditions from the values of the output.

For a LTI system, the following relations hold:

$$\mathbf{y}(0) = C\mathbf{x}(0)$$

$$\dot{\mathbf{y}}(0) = C\dot{\mathbf{x}}(0) = CA\mathbf{x}(0)$$

$$\ddot{\mathbf{y}}(0) = C\ddot{\mathbf{x}}(0) = CA^2\mathbf{x}(0)$$

$$\ddot{\mathbf{y}}_{(t_0)} = C\ddot{\mathbf{x}}(0) = CA^3\mathbf{x}(0)$$

...

$$\mathbf{y}^{n-1}(0) = C\mathbf{x}^{n-1}(0) = CA^{n-1}\mathbf{x}(0)$$

Where n is the number of states of the system. The equations above can be put in a matrix form:

$$\begin{pmatrix} \mathbf{y}(0) \\ \dot{\mathbf{y}}(0) \\ \vdots \\ \mathbf{y}^{n-1}(0) \end{pmatrix} = \begin{pmatrix} C \\ CA \\ \vdots \\ CA^{n-1} \end{pmatrix} \mathbf{x}(0)$$



That is:

$$\mathcal{O}x(0) = Y(0)$$

The initial conditions of the system are found from the output values only if the **Observability matrix** \mathcal{O} is full rank:

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

The concept is maybe more intuitive for a discrete time system, for which the above relations are:

$$y_{(0)} = Cx_{(0)}$$

$$y_{(1)} = Cx_{(1)} = CAx_{(0)}$$

$$y_{(2)} = Cx_{(2)} = CA^2x_{(0)}$$

$$y_{(3)} = Cx_{(3)} = CA^3x_{(0)}$$

...

$$y_{(n-1)} = CA^{n-1}x_{(0)}$$

Since derivation in discrete time is equivalent to consider the successive element.

The Observability matrix is defined as before; in this case is clear that to obtain the n components of the initial state vector we need n equations, and that we need the Observability matrix to be full rank to solve the system.

Useful links:

More on controllability and Observability:

-“Linear state-space control systems” Robert L. Williams, II, Douglas A. Lawrence

-“Fundamentals of linear state space systems” John S. Bay, McGraw Hill, 1999

-<https://ocw.mit.edu/courses/aeronautics-and-astronautics/16-30-feedback-control-systems-fall-2010/lecture-notes> From MIT OpenCourseWare

EXAMPLE

Is the following system observable?

$$\dot{x} = \begin{bmatrix} 1 & 2 & -1 \\ 0 & -1 & 0 \\ 1 & 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ -1 \\ 0 \end{bmatrix} u \text{ and}$$

$$y = [0 \quad 0 \quad 1]x$$

Solution: constructing the observability matrix: $O = \begin{bmatrix} C \\ CA \\ CA^2 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 1 & 2 & -1 \end{bmatrix}$, and this matrix contains linearly independent columns (i.e. full rank). Therefore the system is fully observable.

EXAMPLE

Is the following system observable?

$$\dot{x} = \begin{bmatrix} 1 & 2 & -1 \\ 0 & -1 & 0 \\ 1 & 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ -1 \\ 0 \end{bmatrix} u$$

$$y = [0 \quad 3 \quad 0]x$$

Solution: constructing the observability matrix: $O = \begin{bmatrix} C \\ CA \\ CA^2 \end{bmatrix} = \begin{bmatrix} 0 & 3 & 0 \\ 0 & -3 & 0 \\ 0 & 3 & 0 \end{bmatrix}$, and this matrix contains linearly dependent columns (i.e. rank deficient). Therefore the system is not fully observable.

STATE FEEDBACK EXAMPLE

EXAMPLE

For the plant given by the following state-space representation:

$$\dot{x} = \begin{bmatrix} 1 & 2 & -1 \\ 0 & -1 & 0 \\ 1 & 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ -1 \\ 0 \end{bmatrix} u$$

$$y = [0 \quad 0 \quad 1]x$$

Use State-Feedback method to design a control system which has the following response:

Rise Time: $t_r(0 - 100) = 0.3 \text{ sec}$

Settling Time: $t_s(5\text{percent}) = 0.5 \text{ sec}$

% overshoot $\%OS = 0.07$

Solution:

1) This can be adopted by adopting the 2nd order prototype system general form.

We can compute $\omega_n = 12 \text{ rad/sec}$ and $\zeta = 0.65$, now we have 2 poles, we can **add another pole** to match the degree of the desired system with the degree of the plant. Let us make a **high frequency pole** such that it **decays quickly** having a **small effect** on the system response.

Therefore, our **desired closed loop** system must have the following **characteristic equation**:

$$(s^2 + 15.6s + 144) \left(\frac{s}{50} + 1 \right) = 0 \text{ to achieve the desired performance criteria.}$$

From the previous examples we already know that the plant is both controllable and observable.

Using the procedures shown in the previous examples or using the **MATLAB function “place”**, we get $K = [-494.8 \quad -65.6 \quad -3566.7]$

Now let us **design an observer** that makes the states of the system available.

Since we want the observation to be faster than the closed loop response. We must select the **eigenvalues** of the matrix **(A-LC)** to have **relatively high negative real parts**. Let them be **(-30, -40, -50)**. Which gives the following characteristic equation: $s^3 + 120s^2 + 4700s + 60000 = 0$

The characteristic equation of **(A-LC)** where $L = [l_1 \ l_2 \ l_3]^T$ is computed from a previous example as: $\det[sI - (A - LC)] = s^3 + l_3s^2 + l_1s + (2l_2 - l_3 + l_1 + 1) = 0$

We get: $l_3 = 120, l_1 = 4700, 2l_2 - l_3 + l_1 + 1 = 60000 \rightarrow l_2 = 27709.5$

$$L = [4700 \ 27709.5 \ 120]^T$$

Now Let us implement this result in **SIMULINK** to check the response.

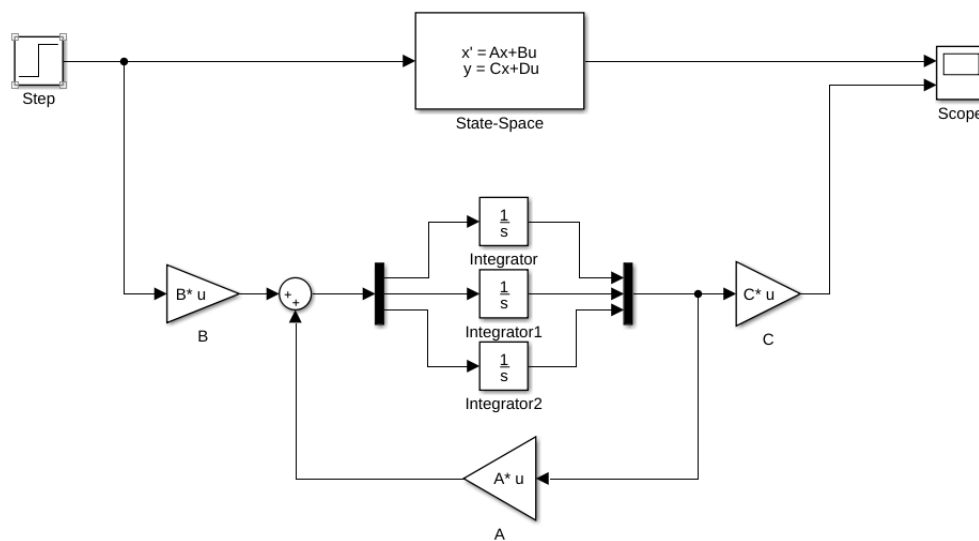


Figure 52- SIMULINK representation for the plant in state-space and observer with identical parameters, sharing the same input with the plant

The plant is represented by the SIMULINK block **State-Space**, and the observer is constructed from the blocks **Matrix Gain**, **Integrator**, **Sum**, **Mux** and **Demux**.

Since both the plant and the observer have exactly the same model, are excited by the same input, and starting from the same initial condition, we expect them to have the same output. As shown in the **Scope** graph in figure 53.

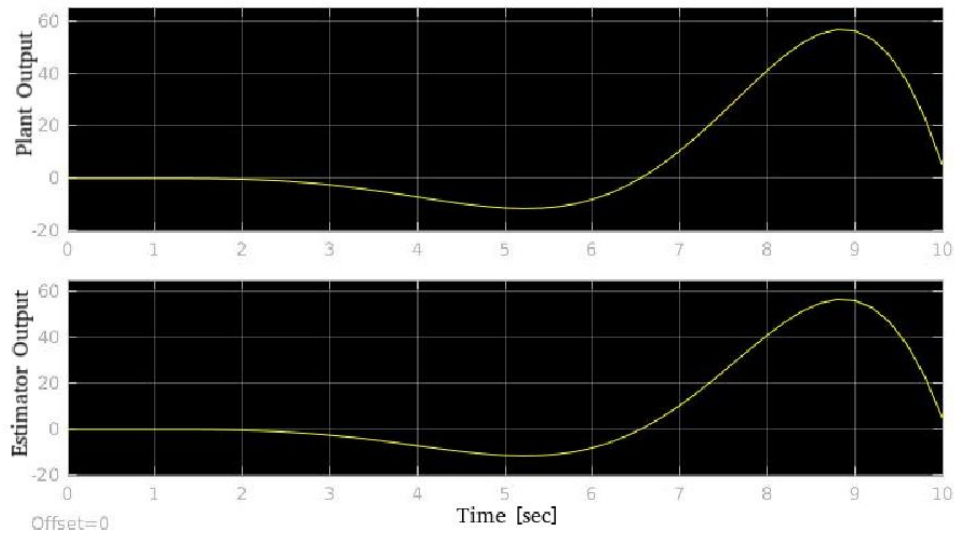


Figure 53- Output response of the system in figure 52

But if the initial condition was different, and this is a more realistic case, we can't expect the estimation error to converge to zero with this configuration. Below is the response of the plant and the observer, with the plant starting from initial condition $x_0 = [-0.1 \ 0.7 \ 0]^T$.

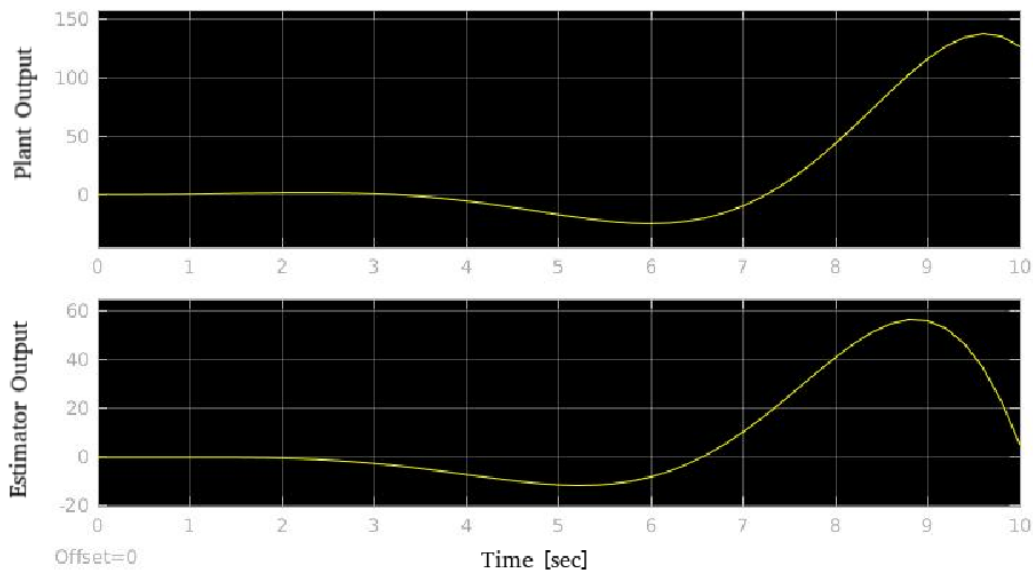


Figure 54- Output response of the system in figure 52, with the initial condition of the plant different from the observer initial condition

We will use Luenberger's configuration, as shown below:

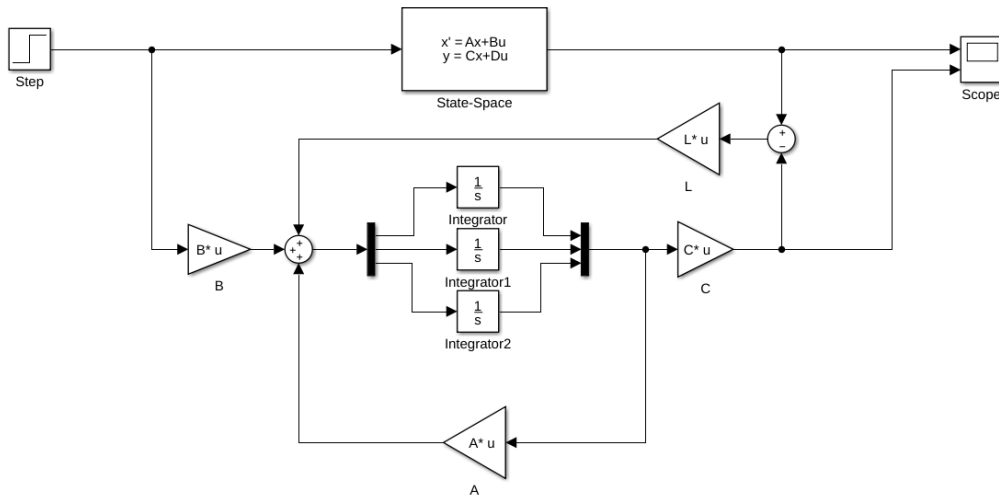


Figure 55- Luenberger observer

In this case, even with the difference in initial conditions, the output error converges very fast.

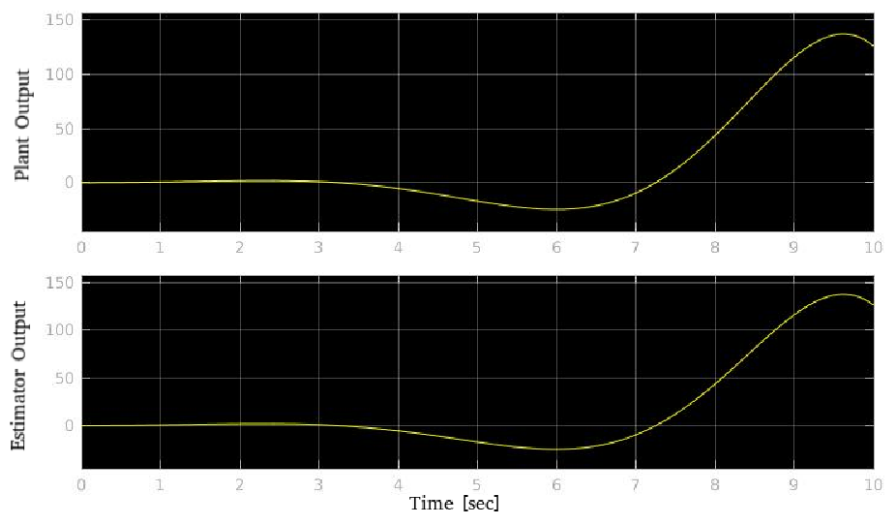


Figure 56- Output response of the system in figure 55

Now that the states are available, we can feed it back by a **Matrix Gain K**.

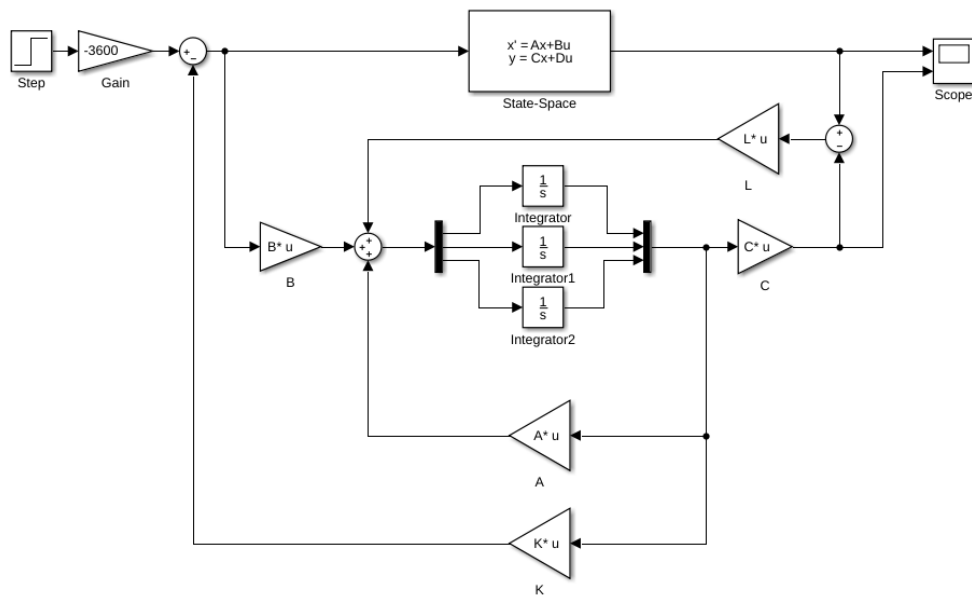


Figure 57- State feedback control

The response, as shown below, complies with the performance criteria.

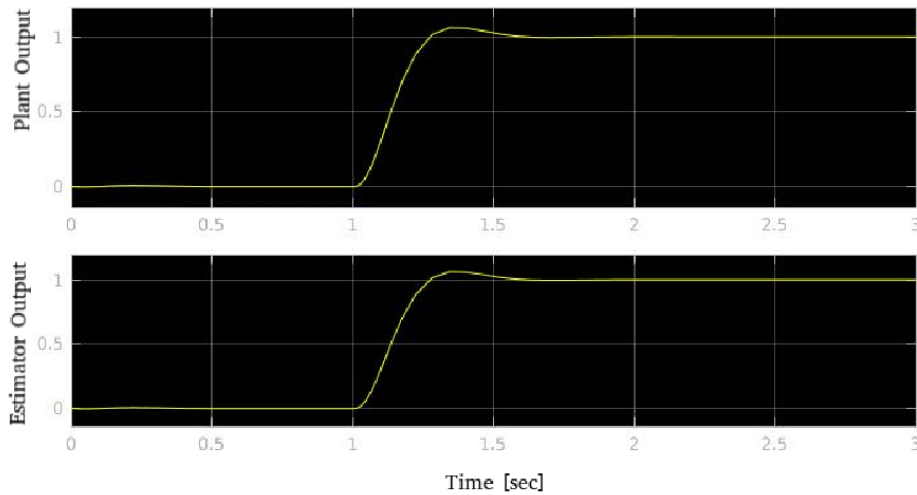


Figure 58- Output response of the system in figure 57

PID CONTROL

DEFINITION

The PID control is a control methodology in which the controller calculates continuously the error value $e(t)$ as the difference between the reference value (set point) and the measured value (process variable), and applies a correction based on the proportional, integral and derivative terms.

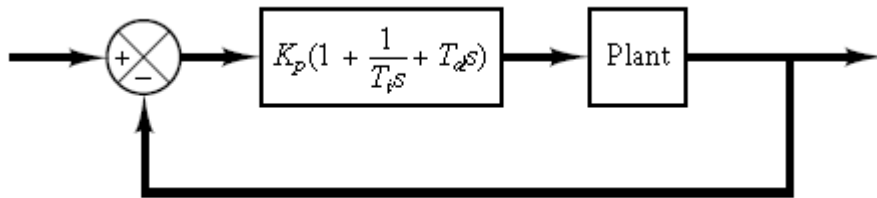


Figure 59- PID Control

The control variable $u(t)$ is a sum of three terms: **proportional**, **integral** and **derivative** terms.

$$u(t) = K_p \left(e(t) + \frac{1}{T_i} \int_0^t e(t) + T_d \frac{de(t)}{dt} \right).$$

When tuning the gains K_p , T_i and T_d , usually with no knowledge of the process model needed, the basic design requirements can be met.

The effects of each term is explained through the following example.

EXAMPLE

Consider a plant with a model given by: $G(s) = \frac{1}{(s+1)^3}$. Find the parameters of a PID controller to satisfy the following criteria for the response of a unitary step input.

Rise Time = 1 second



Settling Time= 6 seconds

%overshoot= 8%

And no Steady State Error

SOLUTION

We will examine the effect of increasing the value of each of the variables K_p , T_i and T_d .

PROPORTIONAL ACTION

$$u(t) = K_p e(t)$$

The proportional term produces an output value that is proportional to the current error value. The proportional response can be adjusted by multiplying the error by a constant K_p , called the proportional gain constant.

In general, increasing the proportional gain will increase the speed of the control system response. However, if the proportional gain is too large, the process variable will begin to oscillate. If K_p is increased further, the oscillations will become larger and the system will become unstable and may even oscillate out of control.

The Proportional term reduces the steady-state error, but does not eliminate it completely. The steady-state error can be corrected by adding an integral term.

By increasing K_p from 1 to 8, we obtain the following results:

Table 1- Performance for different values of K_p

K_p	Rise Time	Settling Time	Overshoot [%]	Steady Error	State
1	1.8723	8.3956	13.9071	0.5	
2	1.3498	10.0674	29.8582	0.333	
3	1.1082	13.5698	43.1090	0.25	
4	0.9634	18.6946	54.2671	0.2	
5	0.8656	26.0796	63.8735	0.167	
6	0.7936	40.5667	72.2968	0.143	
7	0.7378	85.8496	79.8040	0.125	
8	Not Stable	Not Stable	Not Stable	Not Stable	

The response is shown below for some values of K_p .

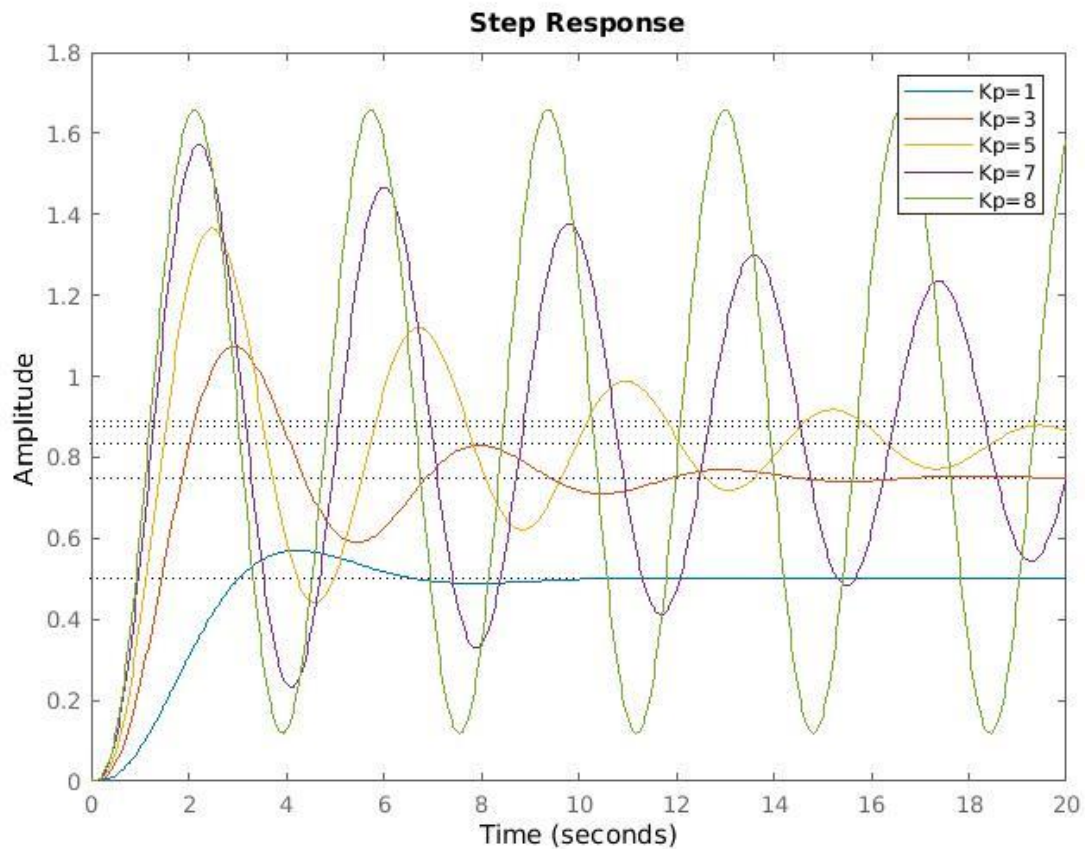


Figure 60- step response of a proportional control with different values of Kp

INTEGRAL ACTION

$$u(t) = \frac{1}{T_i} \int_0^t e(\tau) d\tau$$

The integral term accelerates the movement of the process towards set-point (reference) and **eliminates the residual steady-state error** that occurs with a pure proportional controller. However, since the integral term responds to accumulated errors from the past, it can cause the present value to **overshoot** the set-point value.

Using both the proportional and the integral actions, and by increasing T_i from 1 to 8, while keeping $K_p = 1$ we obtain the following results:

Table 2- performance of PI control for different values of T_i

T_i	Rise Time	Settling Time	Overshoot	Steady Error	State
1	1.7271	30.9388	54.3517	0	
1.5	2.0581	15.2556	28.5868	0	
2	2.3640	11.1555	13.5191	0	
2.5	2.6810	11.2633	3.5231	0	
3	3.0622	12.8774	0	0	
3.5	3.7284	17.0480	0	0	
4	9.2768	20.2509	0	0	
4.5	10.4974	23.5872	0	0	
5	12.0901	26.8732	0	0	

Note that T_i is in the **denominator** (increasing its value decreases the integral action). The response is shown for some values of T_i .

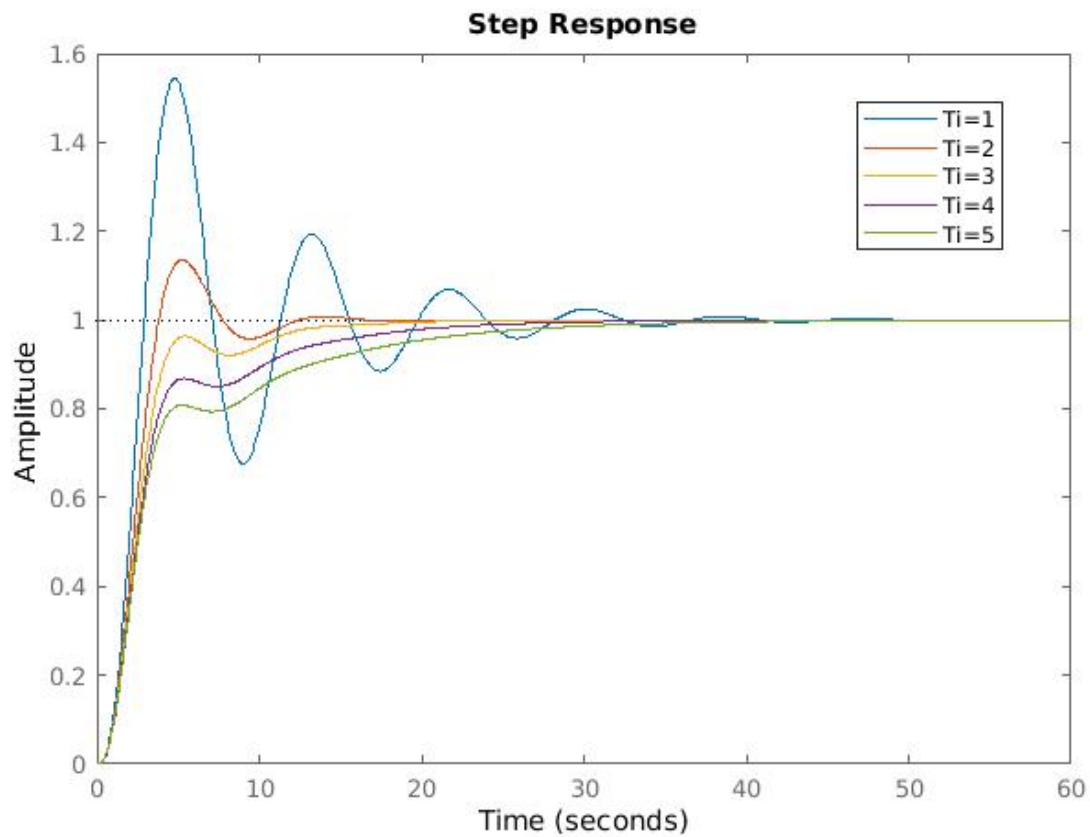


Figure 61- Step responses of PI control for different values of T_i

DERIVATIVE ACTION

$$u(t) = T_d \frac{de(t)}{dt}$$

Derivative action predicts system behavior and thus improves settling time and stability of the system.

Using all PID terms, and by increasing T_i from 0.1 to 4.1, while keeping $K_p = 1$ we obtain the following results:

Table 3- performance of PID control for different values of T_d

T_d	Rise Time	Settling Time	Overshoot	Steady Error	State
0.1	1.7956	27.3501	50.3298	0	
1.1	2.3298	17.9609	24.5506	0	

2.1	2.9310	16.0398	16.4991	0
3.1	3.5637	18.2269	13.7290	0
4.1	4.0142	20.1882	12.4257	0

The response is shown for some values of T_d .

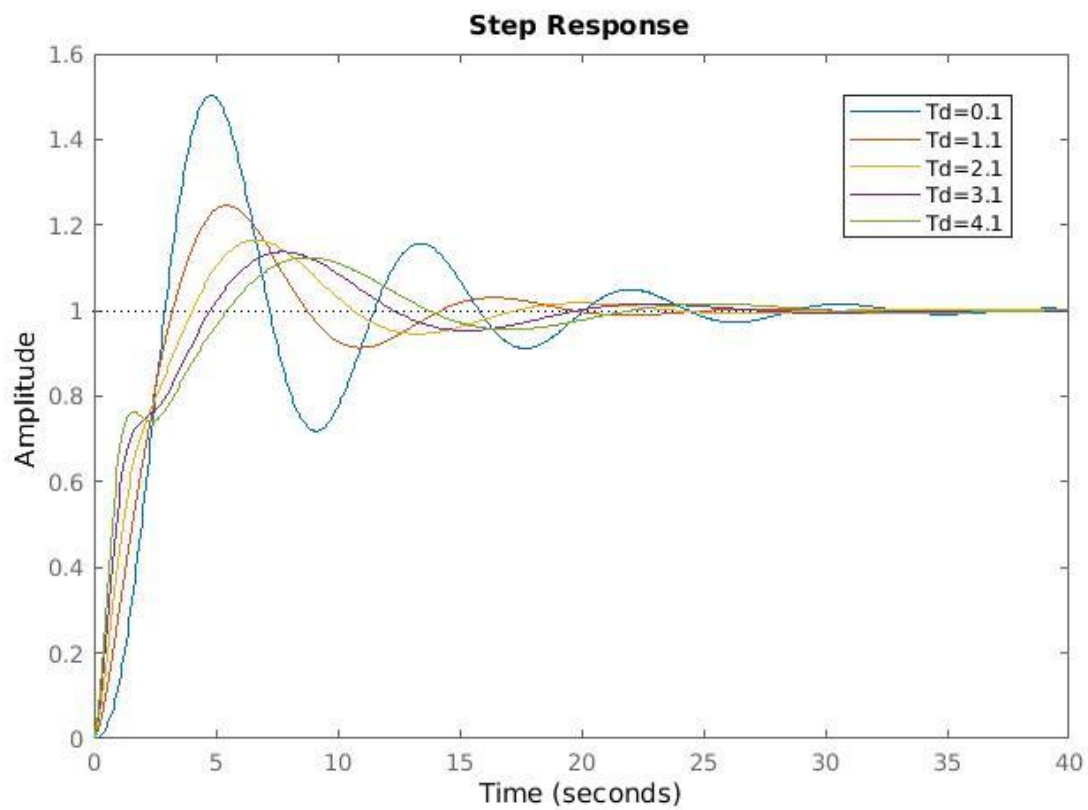


Figure 62- Step responses of PID control for different values of T_d

ALL PID ACTIONS

By selecting $K_p = 4.33$, $T_i = 3.62$ and $T_d = 0.894$ we get the following response.

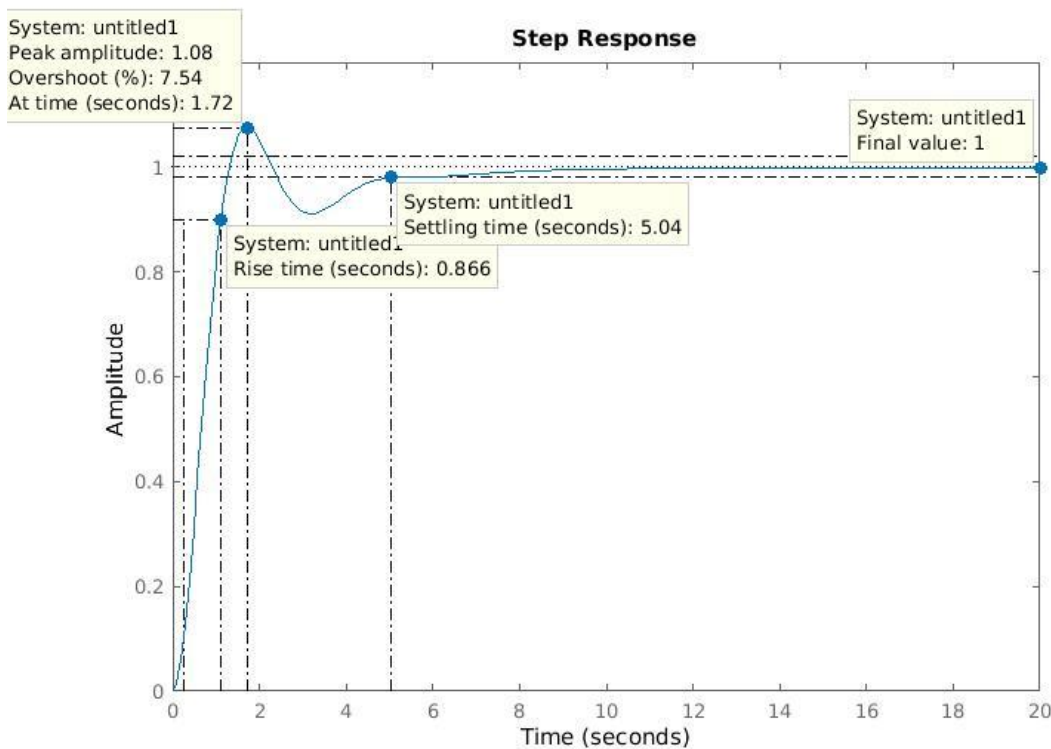


Figure 63- Step response of a tuned PID control

Which achieves the required specifications.

TUNING METHODS

Apart from **Manual** Tuning, several tuning methods can be exploited, some require few knowledge of the plant “process” model, other are empirical methods.

Links:

Astrom and Hagglund PID Automatic Tuning method

PID Controllers, Theory, Design, and Tuning, 2nd edition, by Karl J. Astrom and Tore Hagglund

Ziegler_Nichols PID Tuning method + Tuning via frequency response approach

Modern Control Engineering, 5th edition, by Katsuhiko Ogata, Chapter 8, PID Controllers and Modified PID Controllers

MATLAB PID TUNER

The PID Tuner tool in MATLAB can be used to automatically tune the PID parameters. One way is to use Simulink to connect the blocks as shown below.

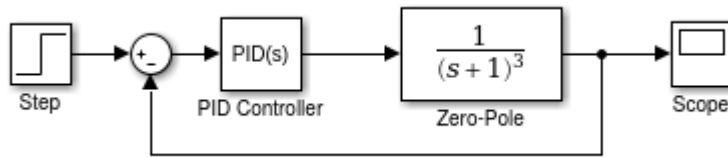


Figure 64- Simulink representation of a PID control

Selecting the “PID Controller” block opens the following dialog.

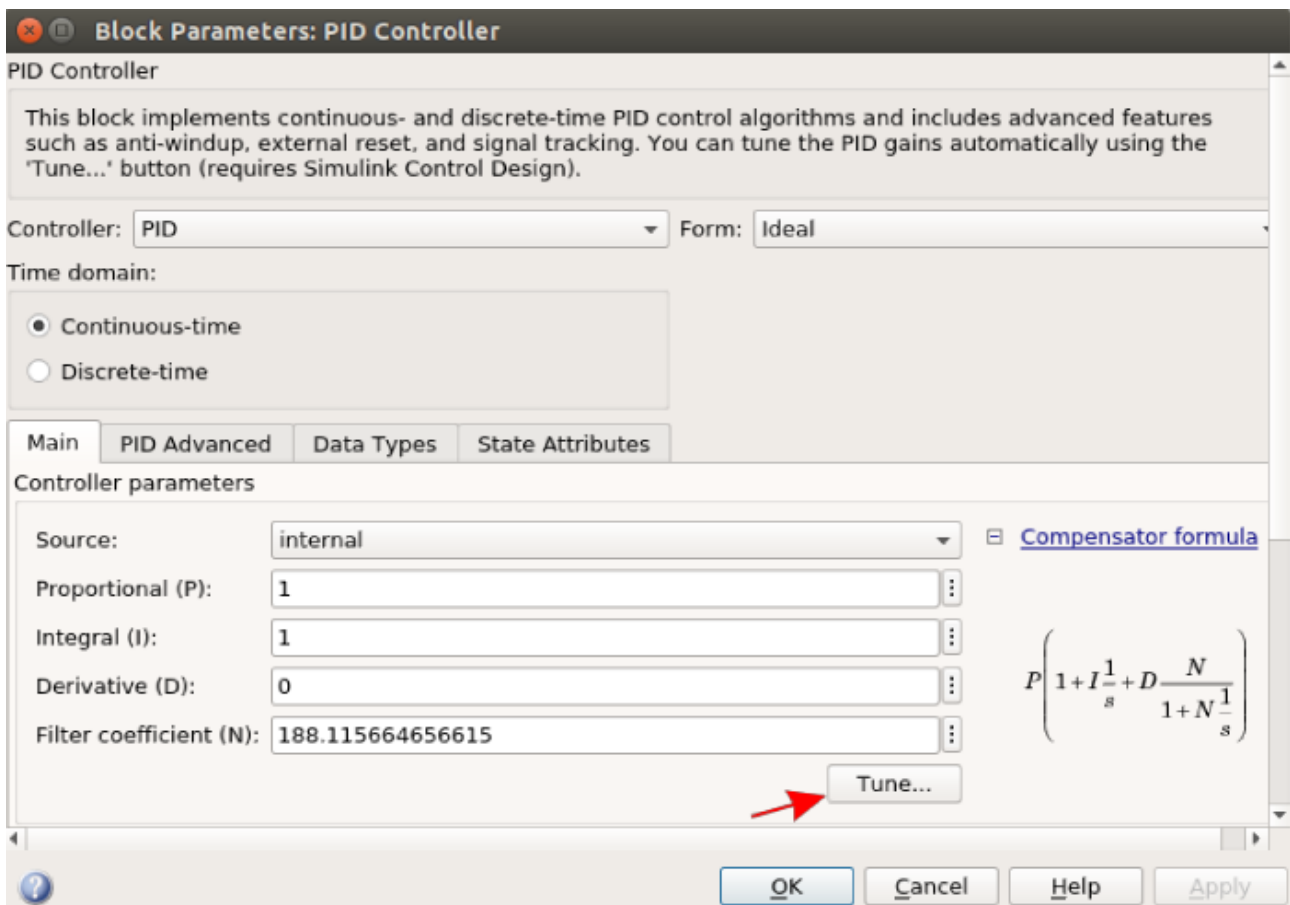


Figure 65- Simulink PID block

Then click on “Tune...” to open the following dialog.

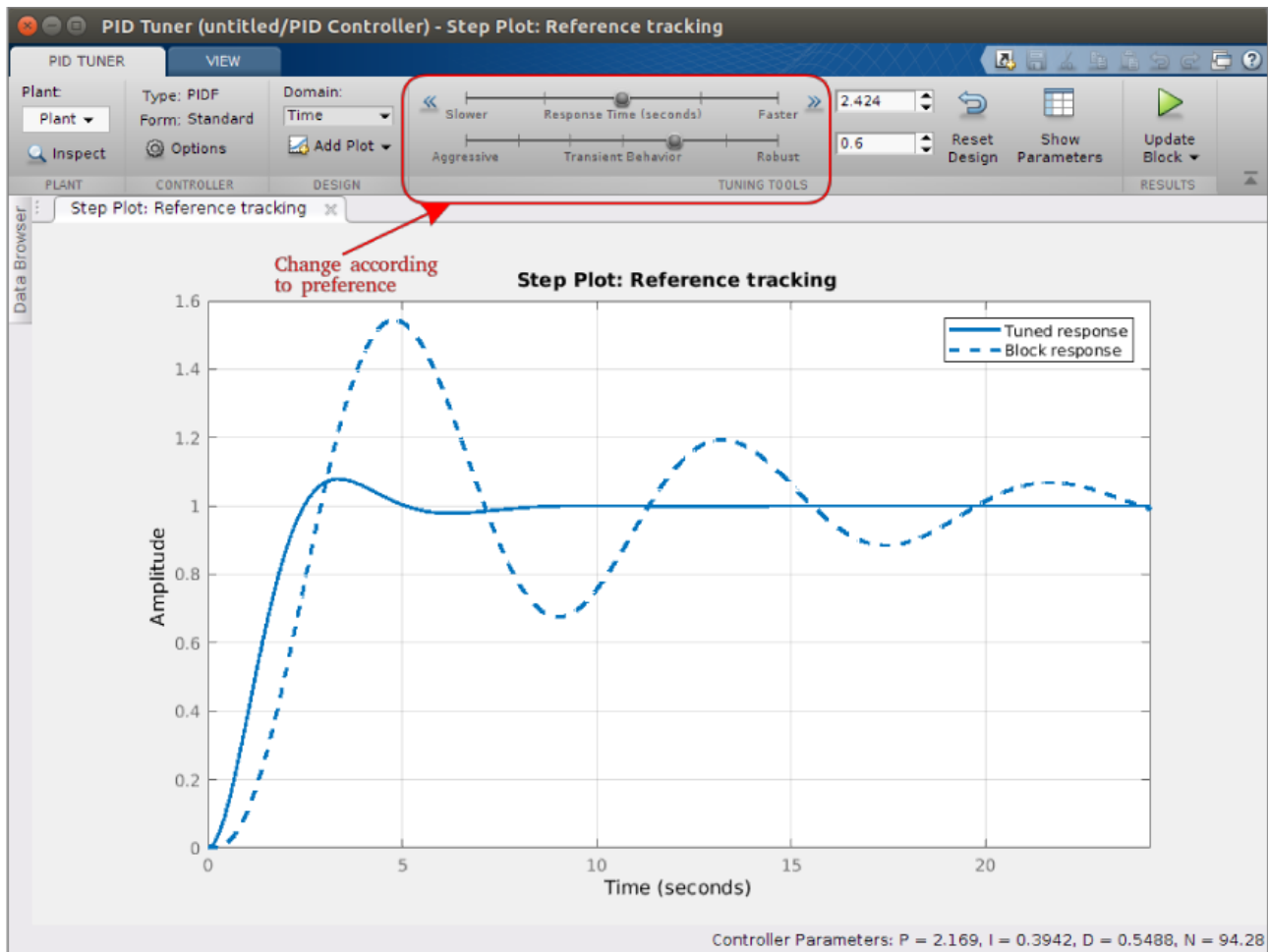


Figure 66- MATLAB PID Tuner, dialog

After finishing the tuning process, you can update the Simulink block by pressing “Update Block”. You can also export it to MATLAB workspace.

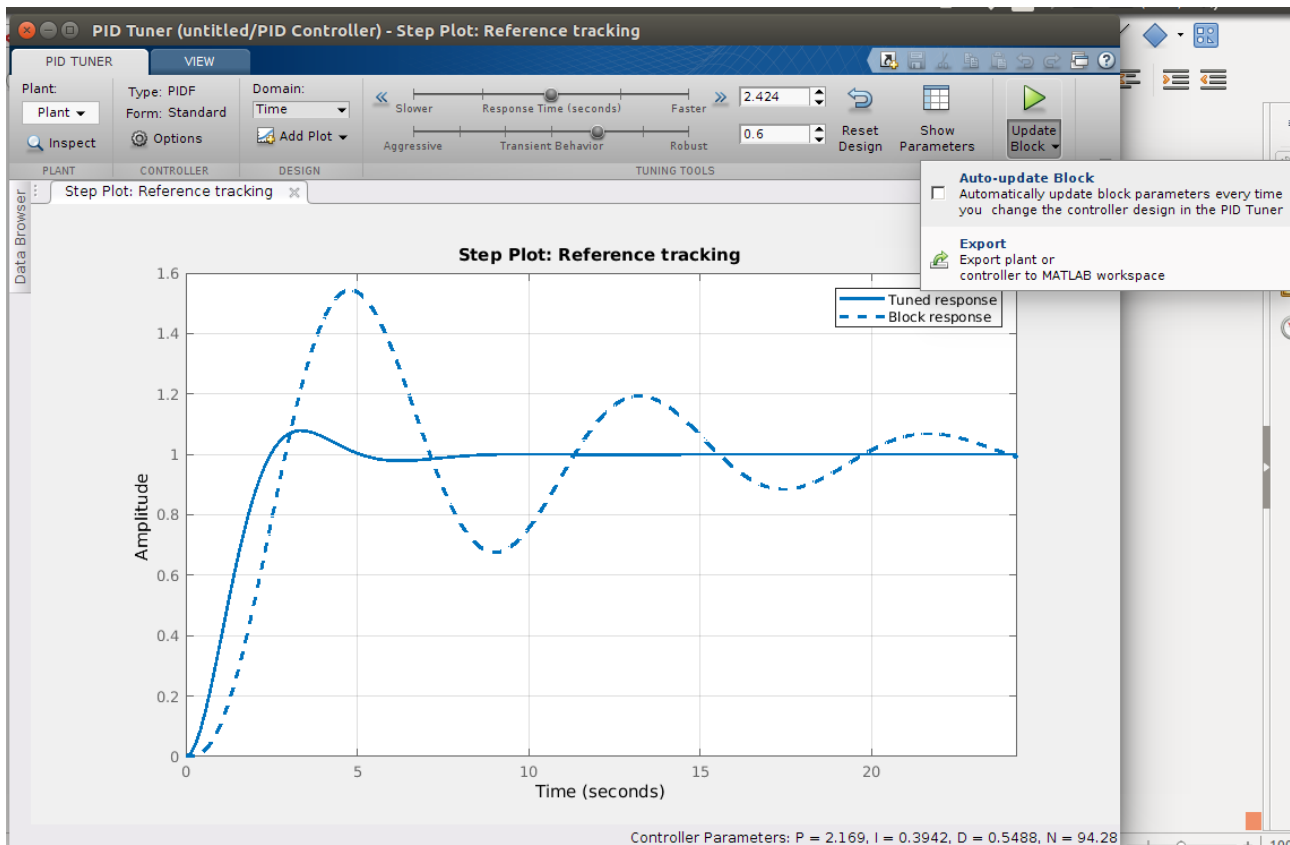


Figure 67- MATLAB PID Tuner, you can update the parameters in the Simulink block or export the controller transfer function to MATLAB workspace

Link

MATLAB PID Tuner: <https://it.mathworks.com/help/control/ref/pidtuner-app.html>

LEAD COMPENSATORS

The Lead Compensator transfer function is given by: $G_{lead} = \frac{1 + \frac{s}{z_d}}{1 + \frac{s}{m_d z_d}}$

The pole-zero location of the compensator is sketched below. Since $m_d > 1$, the pole is always located on the left-hand side of the zero. For some different m_d 's, the Bode diagram of the lead compensator with $z_d = 1$ are shown below.

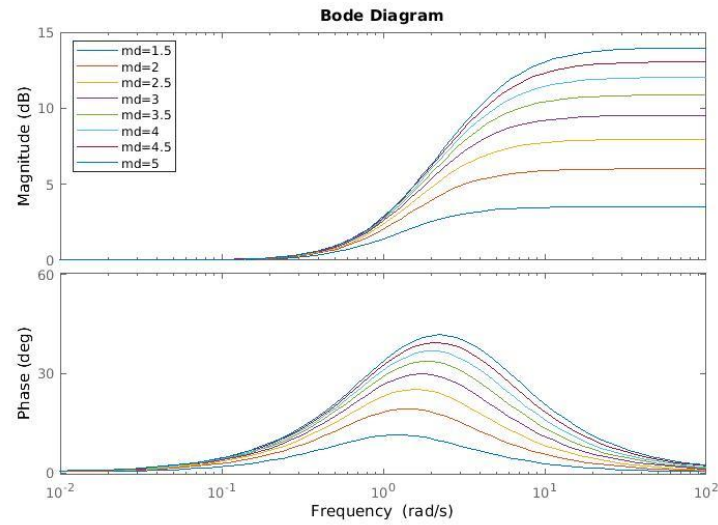


Figure 68- Bode diagram of a Lead compensator with differnt values of m_d and $z_d=1$

It can be observed that when m_d is large, the gain compensation is large, and the phase compensation as well.

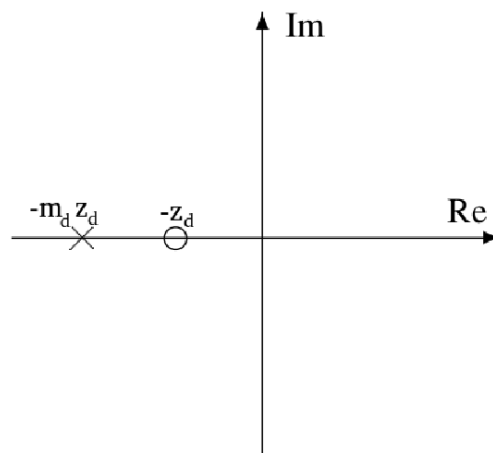


Figure 69- Pole-zero map of a Lead Compensator

To show the impact of the lead compensator on a process, consider the following example.

EXAMPLE

For the plant given by the transfer function: $G(s) = \frac{1}{s(s+1)^2}$

The step response and Bode plot are shown:

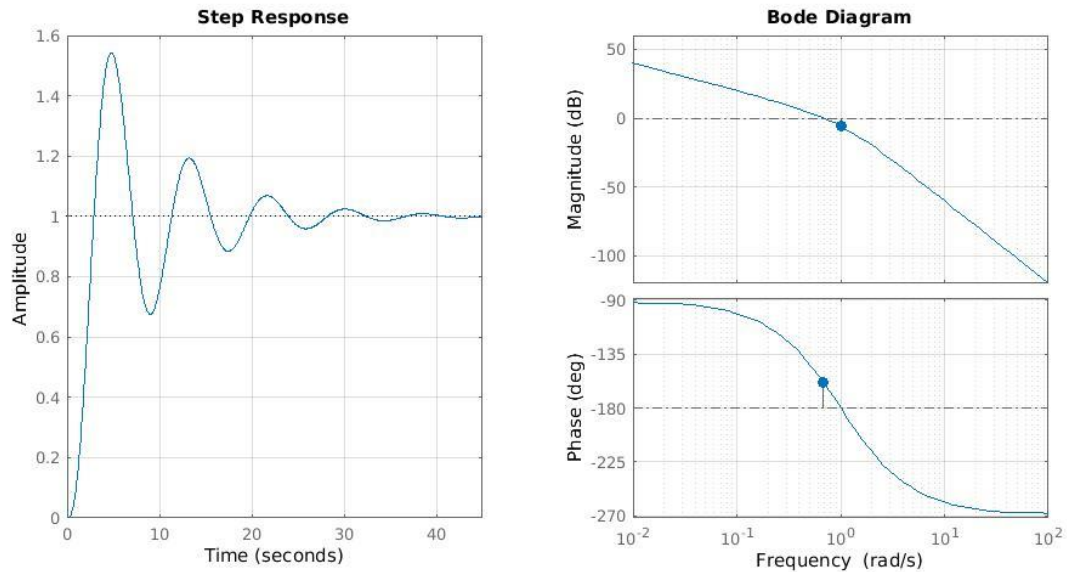


Figure 70- Bode diagram and the step response of $G(s)=1/(s(s+1)^2)$

Its phase margin is 21.39° at a frequency of 0.682 rad/sec, and a gain margin of 2 at a frequency of 1 rad/sec.

The objective is to increase **both the speed** and the **damping** of the feedback system. From the analysis part we know that **increasing the phase margin increases the damping** of the feedback system, and increasing the **crossover frequency (or the bandwidth) increases the speed of the system** in general (rise time and settling time).

That can be achieved by adding a Lead Compensator of a suitable frequency.

Selecting $z_d = 0.3846$ and $m_d = 16$, we get $G_{lead} = \frac{2.6s+1}{0.1625s+1}$, which has the following frequency response.

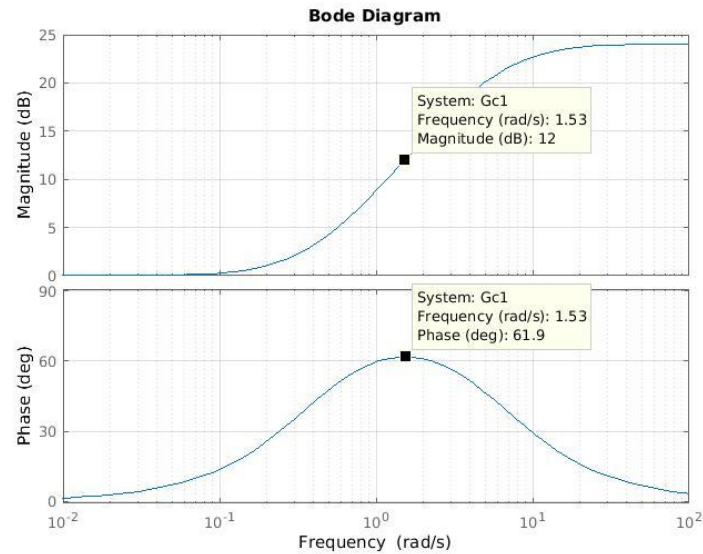


Figure 71- Bode diagram of the selected Lead Compensator

This Compensator adds gain of about 12 dB, and a phase of about 61.9° to the process at frequency of about 1.53 rad/sec. This will increase the crossover frequency ω_c because the gain plot will go up around this frequency, and will increase the phase around this frequency (the phase margin).

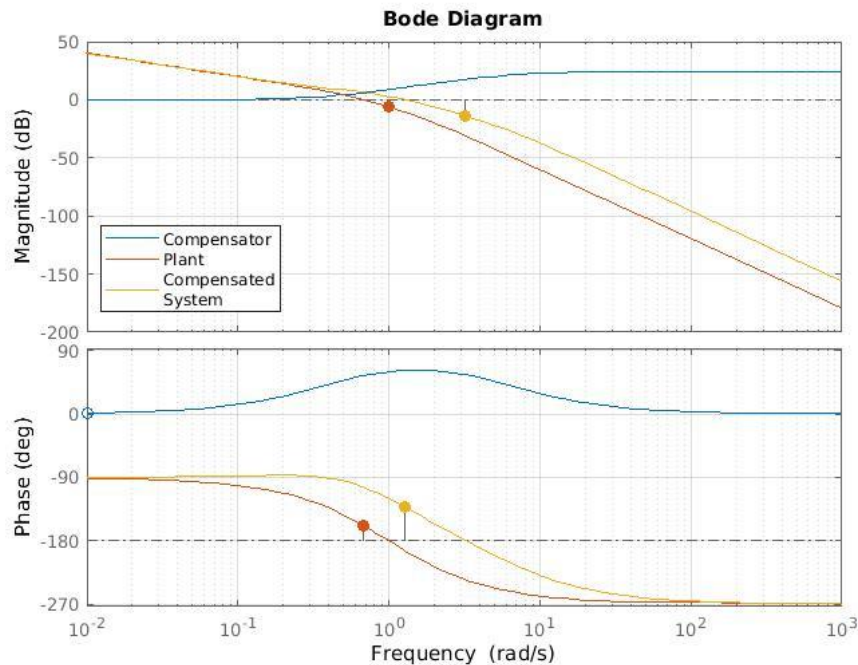


Figure 72- Bode diagram of the Lead Compensated und the uncompensated systems

The phase margin was increased from 21.39° to 47.25° , and the crossover frequency was increased from 0.682 to 1.287 rad/sec.

The following plot shows the improvement in the step response of the system.

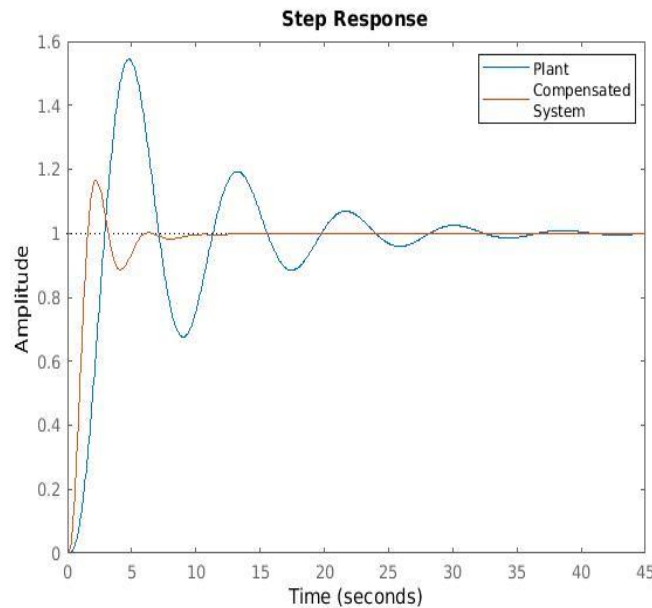


Figure 73- Step response of the compensated and the uncompensated systems

LAG COMPENSATOR

The Lag Compensator transfer function is given by: $G_{lag} = \frac{1 + \frac{s}{m_i z_i}}{1 + \frac{s}{z_i}}$

The pole-zero location of the compensator is sketched below. Since $m_i > 1$, the zero is always located on the left-hand side of the pole. For some different m_i 's, the Bode and Nyquist diagrams of the lead compensator with $z_i = 1$ are shown below.

It can be observed that when m_i is large, the gain attenuation is large, and the phase compensation as well.

To show the impact of the lag compensator on a process, consider the following example.

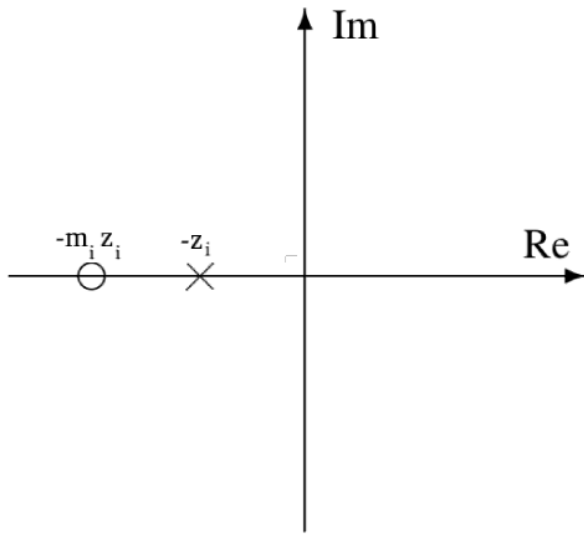


Figure 74- Pole-zero map of the Lag compensator

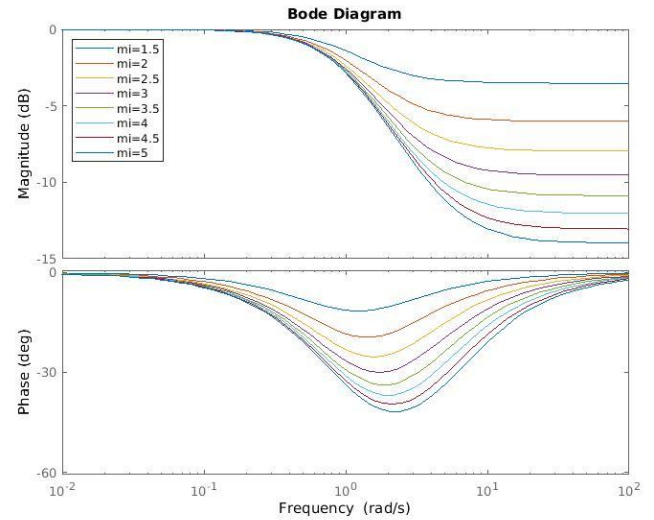


Figure 75- Bode diagram of the Lag compensator for different values of m_i and for $z_i=1$

EXAMPLE

For the same plant treated in the previous example: $G(s) = \frac{1}{s(s+1)^2}$

With lag compensation, increasing the damping by increasing the gain margin is possible by exploiting the attenuation in the gain. In this case **we are not interested** in the phase lag itself. Therefore, the region where the maximum phase lag occurs is usually placed at a **low frequency** where it **does not affect the transient behaviour**.

On the other hand, decreasing the gain reduces the cutoff frequency ω_c , which reduces the bandwidth and thus the speed of the system (rise time and settling time).

Selecting $z_i = \frac{0.001}{2.6} = 0.000385$ and $m_i = 3$, we get $G_{lead} = \frac{2.6s+1}{0.1625s+1}$, which has the following frequency response.

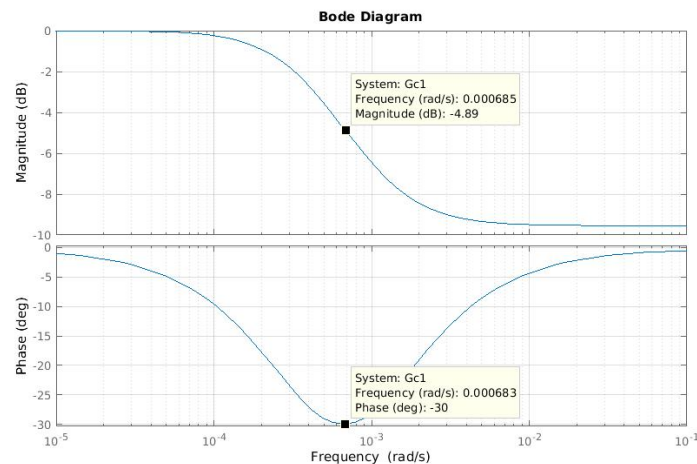


Figure 76- Bode diagram of the selected Lag compensator

This Compensator reduces the phase of about -30° at a low frequency (0.000683 rad/sec), so it does not actually affect the response.

And on the other hand it decreases the magnitude by -9.54 after that low frequency (middle and high frequencies), this will increase the phase margin of the system from 21.4° to 55.9° , which results in a more stable system with more damped response (less overshoot).

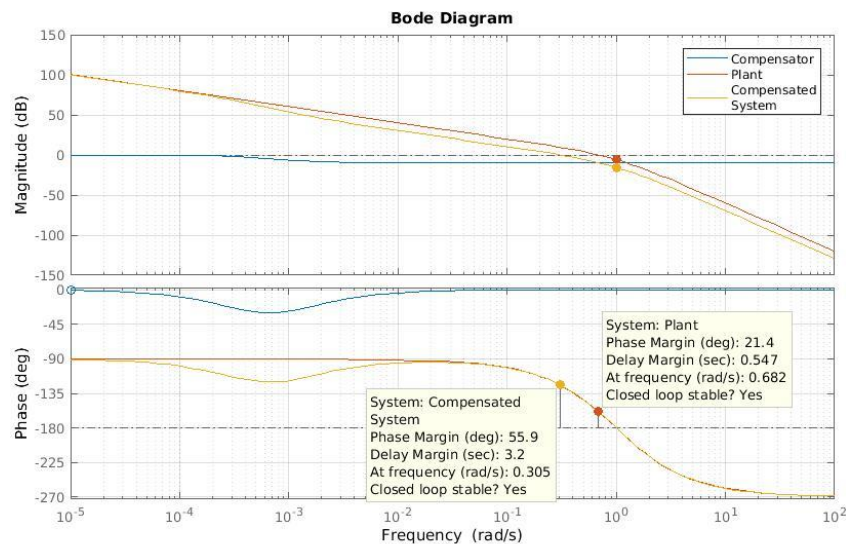


Figure 77- Bode diagram of the Lag compensated and the uncompensated systems

This gain attenuation results also in decreasing the crossover frequency ω_c from 0.682 to 0.305 rad/sec, which decreases the rise time and the settling time.

The following plot shows the improvement in the step response of the system.

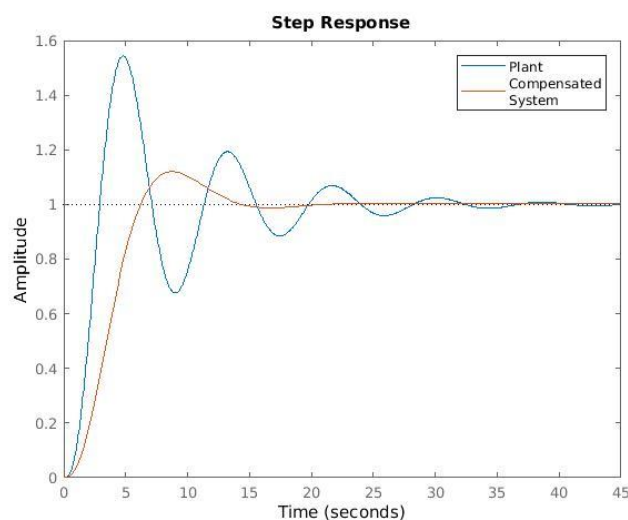


Figure 78-Step response of the Lag compensated and the uncompensated systems

Although we got a less oscillating response, less overshoot, the rise time is reduced a bit, but the settling time is still better due to the more damped response.

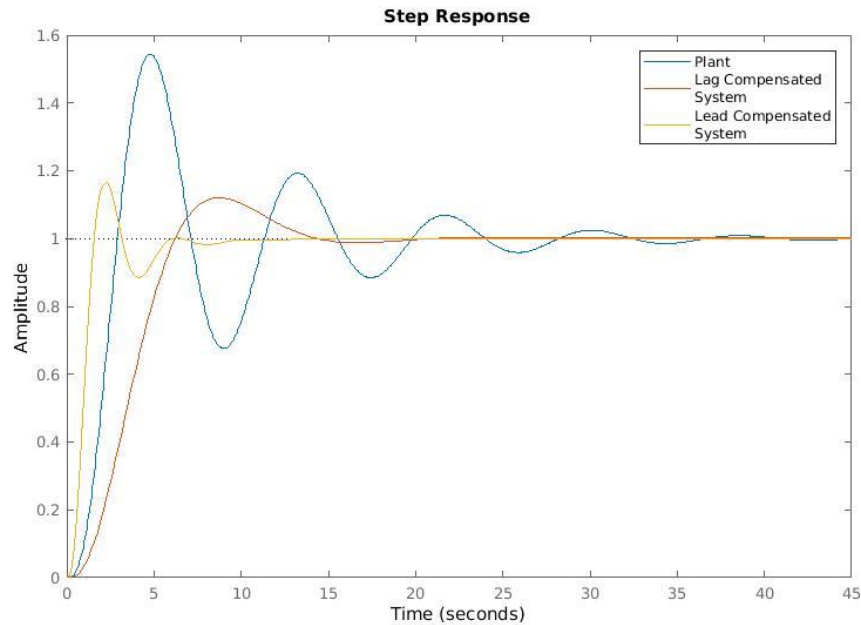


Figure 79- Step responses of the Lead Compensated, Lag Compensated and the uncompensated systems

The difference is more obvious plotting the responses of both compensation techniques together.

Both compensator got almost the same overshoot value. But the **Lead compensator** achieved considerably **faster** response.

CONTROL DESIGN USING LEAD-LAG COMPENSATION

After we examined the features of each compensator type. We can design a control system that meets specifications using a Lead-Lag compensator.

EXAMPLE

For the feedback system shown below.

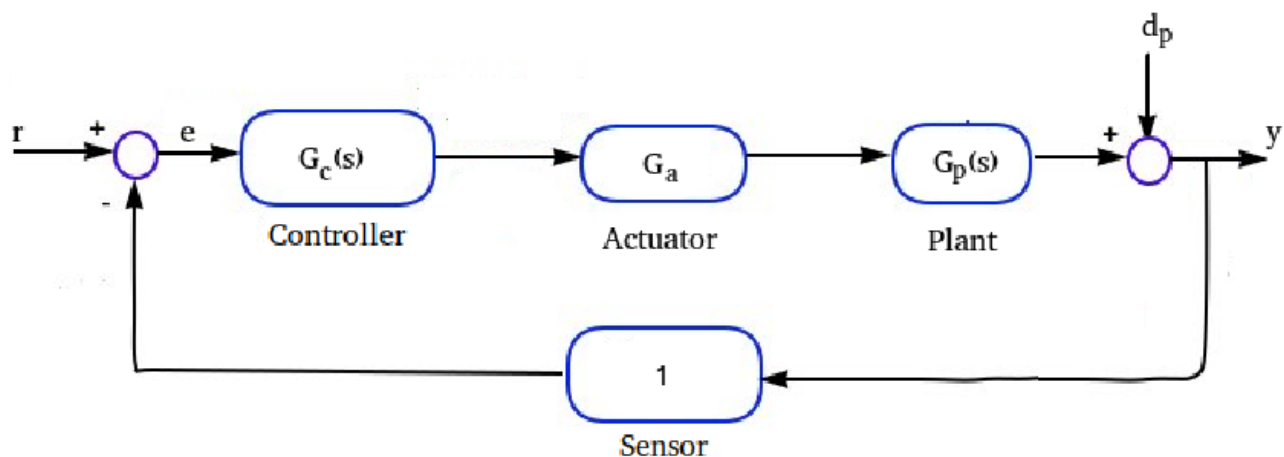


Figure 80- Block diagram of a control system



Where:

$$G_p(s) = \frac{5}{s(s+1)(0.25s+1)},$$

$$G_a = 1,$$

$$d_p(t) = D_{p0}t; |D_{a0}| \leq 5 * 10^{-3}$$

Design a suitable Lead-Lag compensator $G_c(s)$, in order to meet the following requirements:

Steady State error for a unitary step input reference “r” $|e_r^\infty| = 0$.

Steady State error in the presence of d_p : $|y_{d_a}^\infty| \leq 1.5 * 10^{-3}$

Rise time: $t_r \leq 0.5s$ (from 0 to 100%)

Settling time: $t_{s,5\%} \leq 3s$

Overshoot: $\%OS \leq 10\%$

SOLUTION

The Lead-Lag controller can be written in the form: $G_c(s) = \frac{K_c}{s^\nu} (G_{cLead})(G_{cLag}) =$

$$\frac{K_c}{s^\nu} \left(\frac{1 + \frac{s}{z_d}}{1 + \frac{s}{m_d z_d}} \right) \left(\frac{1 + \frac{s}{m_i z_i}}{1 + \frac{s}{z_i}} \right)$$

STEPS:

- Derive steady state requirements to define constraints on K_c and ν
- Translate transient requirements to a range for crossover frequency ω_c and phase margin ϕ_M .
- Plot Bode diagram for the uncompensated system.
- Design the Lead part, to add the needed phase, at the desired frequency ω_c .
- Design the Lag part, to reduce the gain, in order to adjust the frequency ω_c to become as required.

DERIVING STEADY STATE REQUIREMENTS FOR THE TRACKING ERROR

We will use the final value theorem to test the steady state output response.

Writing the expression for the output error “e” when a reference input “r” is applied:

$$e(s) = r(s) - y(s) = r(s) - G_c G_a G_p e(s)$$

$$(1 + G_c G_a G_p) e(s) = r(s)$$

$$e(s) = \frac{1}{1 + G_c G_a G_p} r(s)$$

The final value theorem is given by:

$$|e_r^\infty| = \lim_{s \rightarrow 0} s |e(s)|$$

$$|e_r^\infty| = \lim_{s \rightarrow 0} s \left| \frac{1}{1 + G_c G_a G_p} r(s) \right|$$

since we know that $r(s)$ is a step function, G_p has one pole in the origin and we also know the form of G_c

$$|e_r^\infty| = \lim_{s \rightarrow 0} s \left| \frac{1}{1 + \frac{K_c K_p}{s^\nu} s} \right| = \lim_{s \rightarrow 0} \left| \frac{s^{\nu+1}}{s^{\nu+1} + K_c K_p} \right| = 0, \text{ regardless the value of } \nu, \text{ the steady state error is 0}$$

thanks to the pole in-the-origin contained in the plant. Moreover, there is no constraint on K_c .

DERIVING STEADY STATE REQUIREMENTS FOR THE PLANT NOISE REJECTION

Writing the expression to the output error in the presence of d_p only:

$$y_{d_p}(s) = d_p(s) - G_c G_a G_p y_{d_p}(s)$$

$$(1 + G_c G_a G_p) y_{d_p}(s) = d_p(s)$$

$$y_{d_p}(s) = \frac{1}{1 + G_c G_a G_p} d_p(s)$$

and using the final value theorem: $|y_{d_p}^\infty| = \lim_{s \rightarrow 0} s |y_{d_p}(s)| = \lim_{s \rightarrow 0} s \left| \frac{1}{1 + G_c G_a G_p} d_p(s) \right| \leq 1.5 * 10^{-3}$

Knowing that $d_p(s) = \frac{D_{p0}}{s^2}$, we can write: $|y_{d_p}^\infty| = \lim_{s \rightarrow 0} s \left| \frac{1}{1 + \frac{K_c K_p}{s^\nu} s} \frac{D_{p0}}{s^2} \right| = \lim_{s \rightarrow 0} \left| \frac{D_{p0}}{s + \frac{K_c K_p}{s^\nu}} \right| \leq 1.5 * 10^{-3}$

by selecting $\nu = 0$ we still can meet the criteria. Note that the DC gain of the plant $K_p = 5$.

$$\left| \frac{D_{p0}}{K_c K_p} \right| \leq 1.5 * 10^{-3}$$

$$|K_c| \geq \frac{1.5 * 10^{-3} * 5}{5 * 10^{-3}} = 1.5$$

in summary: for the steady state requirements, no poles in the origin should be added to the controller, and the DC gain of the controller should be larger than 1.5.

TRANSLATE TRANSIENT REQUIREMENTS TO A RANGE FOR CROSSOVER FREQUENCY AND PHASE MARGIN

We shall adopt the General Form of 2nd order systems, because we have formulas that can relate the transient performance specifications t_r , t_s and %OS to the damping ratio ζ and the natural frequency ω_n . Recalling the general form:

The approach is to related ζ and ω_n to frequency response characteristics: ω_c and ϕ_M , then design compensators to achieve these characteristics in the frequency domain, with this we can obtain approximately the design specifications.

$$\%OS \rightarrow \zeta \rightarrow \phi_M$$

$$\zeta = \frac{|\ln(\%OS)|}{\sqrt{\pi^2 + \ln^2(\%OS)}} = \frac{\left| \ln\left(\frac{10}{100}\right) \right|}{\sqrt{\pi^2 + \ln^2\left(\frac{10}{100}\right)}} = 0.591$$

$$\phi = \tan^{-1} \frac{2\zeta}{\sqrt{-2\zeta^2 + \sqrt{1 + 4\zeta^4}}} = 58.5^\circ$$

$$t_r \rightarrow \omega_n \rightarrow (\text{with } \zeta) \omega_c$$

$$t_r \leq \frac{\pi - \cos^{-1}\zeta}{\omega_n \sqrt{1 - \zeta^2}}$$

$$\omega_c \geq \omega_n \sqrt{\sqrt{1 + 4\zeta^4} - 2\zeta^2} = \frac{\pi - \cos^{-1}\zeta}{t_r \sqrt{1 - \zeta^2}} \sqrt{\sqrt{1 + 4\zeta^4} - 2\zeta^2} = 3.94 \text{ rad/sec}$$

$$t_s \rightarrow \omega_n \rightarrow (\text{with } \zeta) \omega_c$$

$$t_{s,5\%} = \frac{-\ln\left(\frac{5}{100}\right)}{\omega_n \zeta}, \text{ and } \omega_c = \omega_n \sqrt{\sqrt{1 + 4\zeta^4} - 2\zeta^2}$$

$$\omega_c \geq \frac{-\ln\left(\frac{5}{100}\right)}{t_{s,5\%} \zeta} = \sqrt{\sqrt{1 + 4\zeta^4} - 2\zeta^2} = 2.439 \text{ rad/sec}$$

To meet rise time and settling time, we select $\omega_c = 4 \text{ rad/sec}$.

PLOT THE BODE DIAGRAM FOR THE UNCOMPENSATED SYSTEM

The uncompensated system is: $L = G_c G_a G_p = K_c G_p$, and selecting $K_c = 1.5$ we get: $L = \frac{7.5}{s(s+1)(0.25s+1)}$, which has the Bode diagram below.

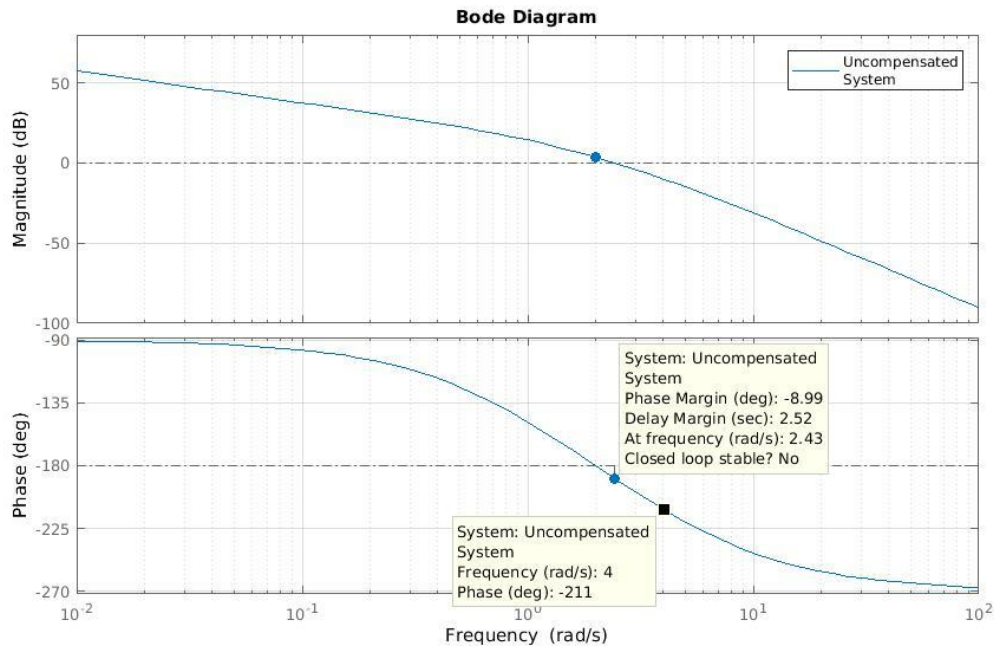


Figure 81- Bode diagram of the uncompensated system

The system has phase margin of -9° . Its crossover frequency $\omega_c = 2.43 \text{ rad/sec}$. Clearly the closed loop is unstable.

DESIGN THE LEAD PART, TO ADD THE NEEDED PHASE, AT THE DESIRED FREQUENCY

One other note on the uncompensated system, is that the phase at the desired frequency (4 rad/sec) is -211° , we need to adjust it to become about 58° (to become later the phase margin).

Thus, the needed phase addition is $(-211 - (-180 + 58)) = 89^\circ$, at the frequency 4 rad/sec.

Known that the lead compensator of 1 pole and 1 zero can provide phase lead up to 90° , but it is recommended to use multiple cascaded compensators to provide such high phase lead, each compensator provides 45° .

For the lead compensator of $G_{lead} = \frac{1 + \frac{s}{z_d}}{1 + \frac{s}{m_d z_d}}$, selecting $m_d = 6$ and $z_d = 1$, we get the following

Bode diagram.

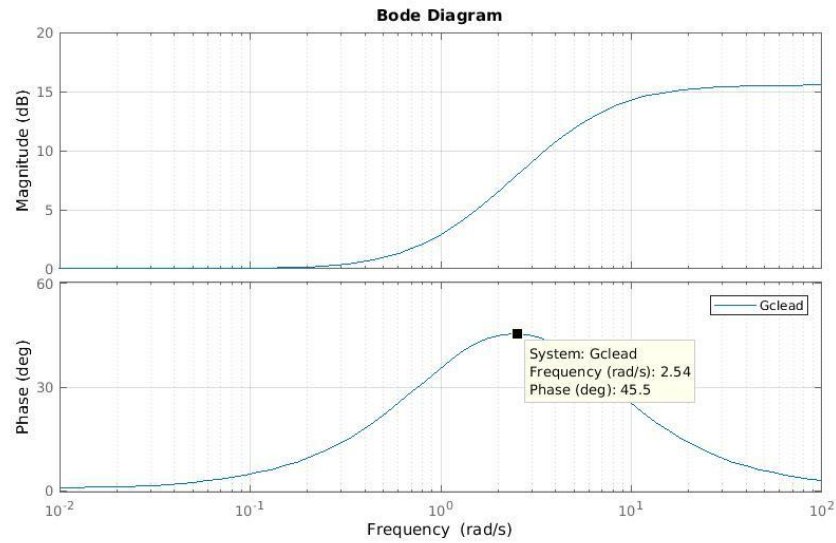


Figure 82- Bode diagram of a Lead compensator of $m_d=6$ and $z_d=1$

To adjust the frequency to by 4 rad/sec, we select z_d as follows: $z_d = \frac{\text{desired frequency}}{2.54} = \frac{4}{2.54} = 1.5748 \text{ rad/sec}$

For the selected values, The lead compensator becomes $G_{clead} = \frac{1 + \frac{s}{z_d}}{1 + \frac{s}{m_d z_d}}$, which has the following Bode plot.

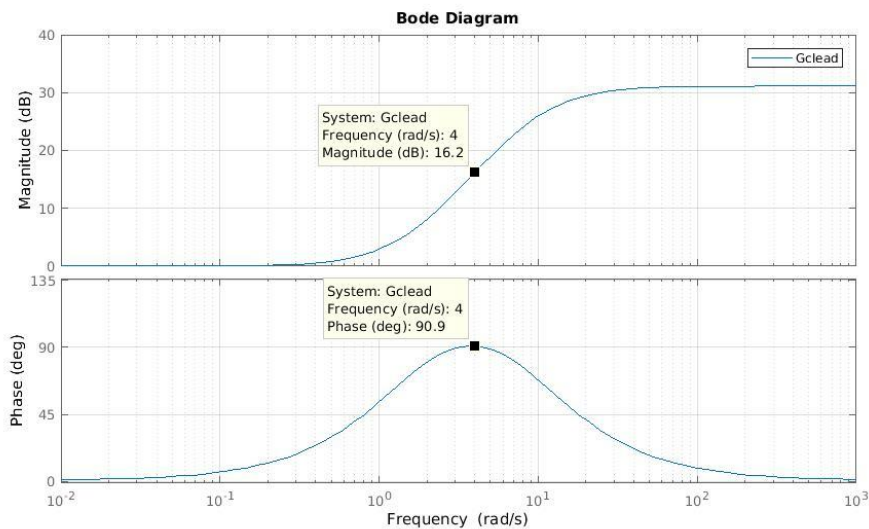


Figure 83- Bode diagram of the 2 cascaded Lead compensators

Applying this to the uncompensated system we get:

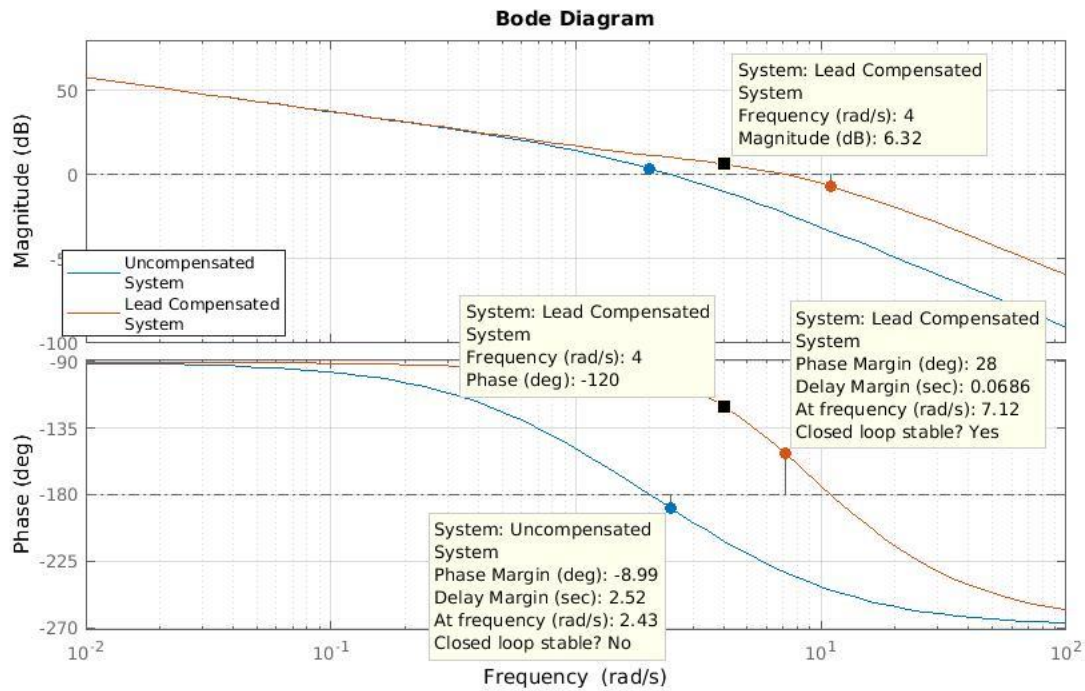


Figure 84- Bode plots of the system compensated with 2 cascaded lead compensators, and the uncompensated system

The phase of lead compensated system was improved, but the gain was increased too, which increased the crossover frequency to 7.12 rad/sec, which is a lot more that needed. Moreover, the phase at the desired frequency, is -120° , which is almost as needed.

The next step is to attenuate the gain to reduce the crossover frequency ω_c . This can not be done by reducing K_c , since it is already at its limit to meet steady state requirements.

- Design the Lag part, to reduce the gain, in order to adjust the frequency ω_c to become as required.

The Lag part is given by: $G_{lag} = \frac{1 + \frac{s}{m_i z_i}}{1 + \frac{s}{z_i}}$, selecting z_i to be at a small frequency, $z_i = 0.0003 \text{ rad/sec}$, a very small frequency since we are not interested in the phase lag, and $m_i = 2$, to obtain attenuation of about -6dB at frequencies higher that the selected, its Bode diagram is shown below.

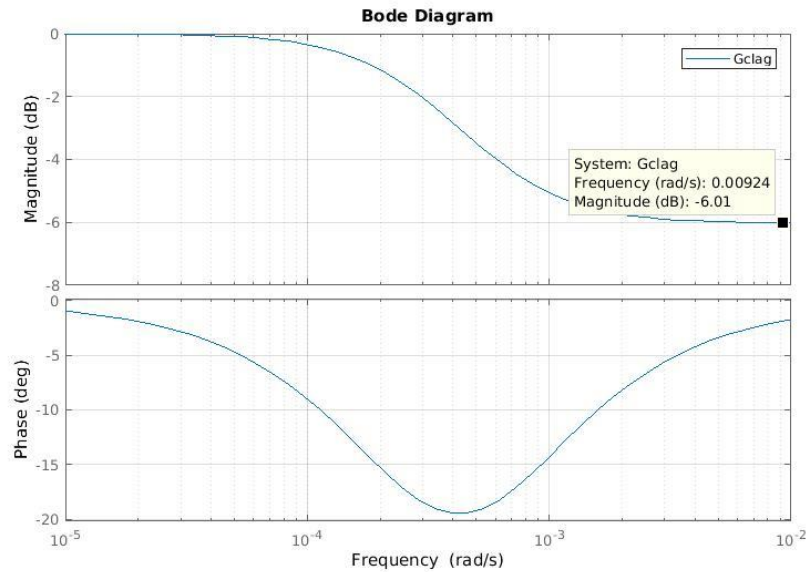


Figure 85- Bode diagram of the selected Lag compensator

By applying this compensator to the lead compensated system, we obtain a system with the following Bode diagram.

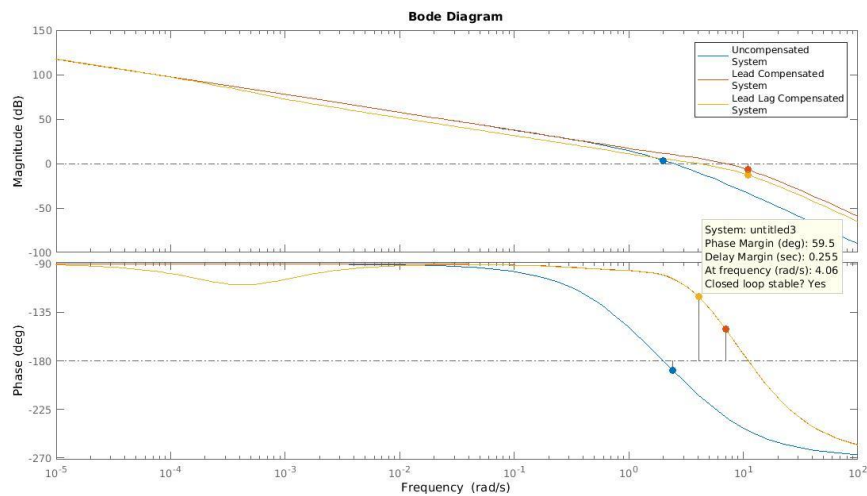


Figure 86- Bode diagram of the uncompensated system, the lead compensated system, and the Lead-Lag compensated system

This shows that the specifications in the frequency domain are satisfied. Recall that G_c is in the form:

$$G_c = K_c \frac{1 + \frac{s}{z_d}}{1 + \frac{s}{m_d z_d}} \frac{1 + \frac{s}{z_l}}{1 + \frac{s}{m_l z_l}} \frac{1 + \frac{s}{m_i z_i}}{1 + \frac{s}{z_i}}$$

To verify that the transient requirements are met, we plot the output response to a step input.

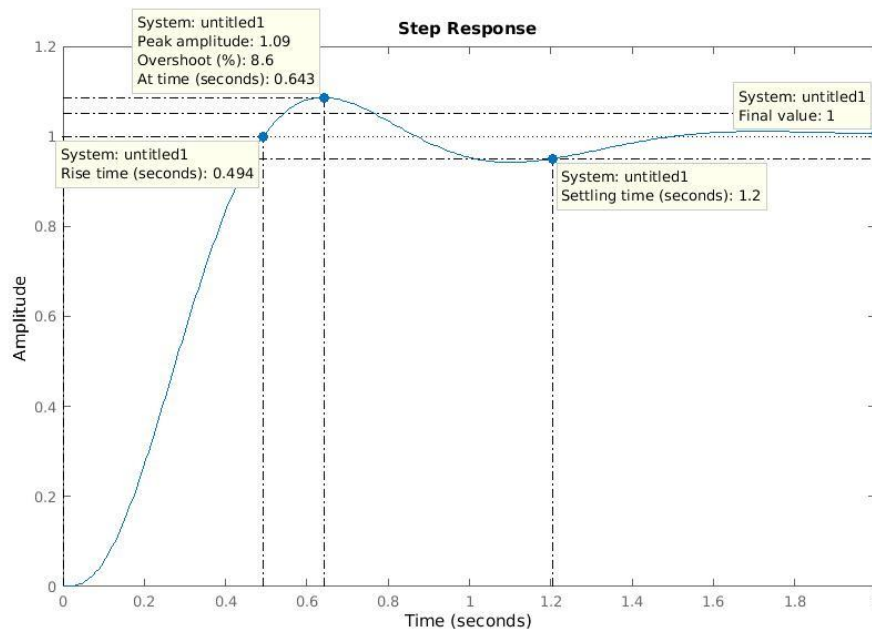


Figure 87- Step response of the Lead-Lag compensated System

Rise time: $t_r = 0.494s \leq 0.5s$ (from 0 to 100%)

Settling time: $t_{s,5\%} = 1.2s \leq 1.5s$

Overshoot: $\%OS = 8.6\% \leq 10\%$

Transient requirements are satisfied.

To check the steady state requirements, we can construct a Simulink model representing the feedback system, we apply the specified inputs and check the proper outputs.

Steady State error for a unitary step input reference “r” $|e_r^\infty| = 0$:

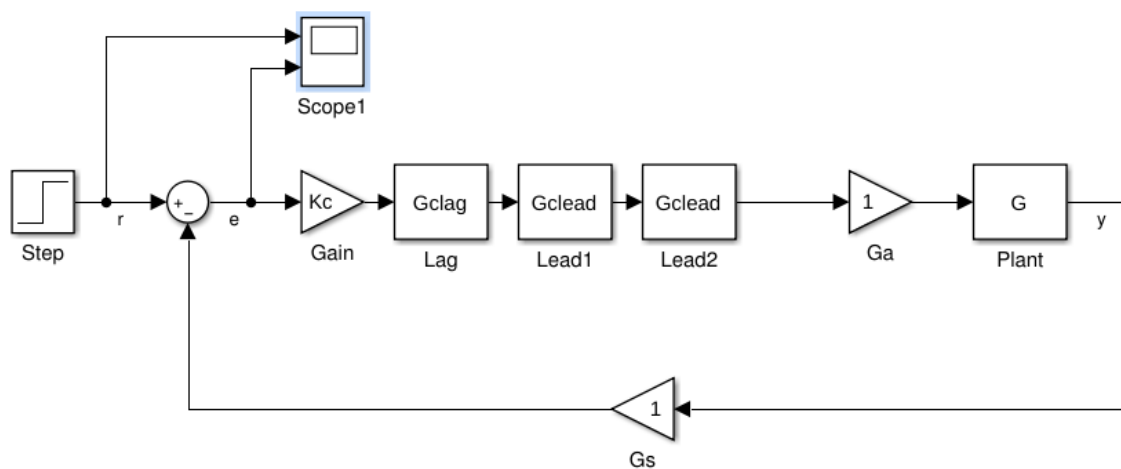


Figure 88- Simulink representation of the feedback system, evaluating output error for a step reference

The following figure shows the Simulink representation of the feedback system.

A “Step” block is connected to the reference input signal, and a “Scope” block is connected to the error signal. The output is plotted in the scope:

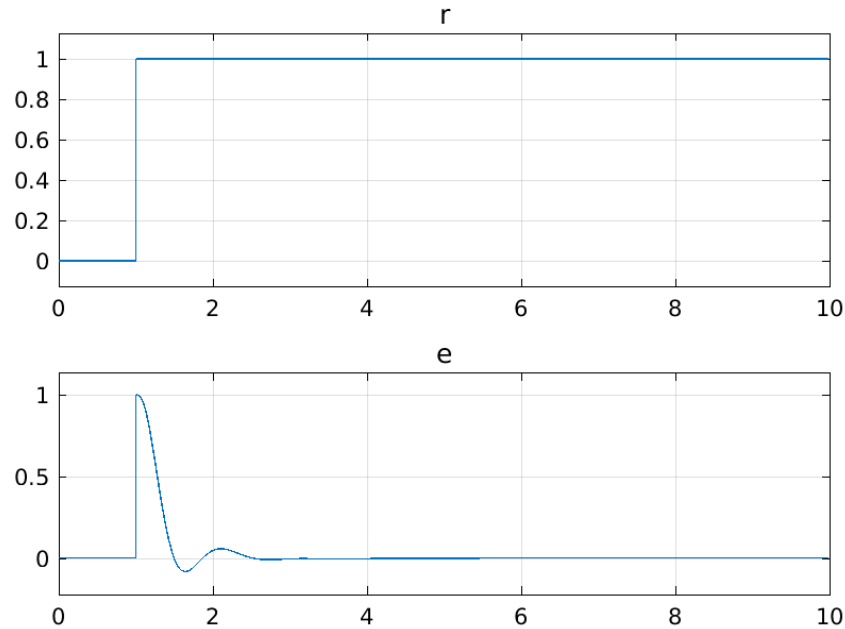


Figure 89- Output steady state error for a step reference

The 1st steady state requirement is satisfied.

Steady State error in the presence of d_p : $|y_{da}^{\infty}| \leq 1.5 \cdot 10^{-3}$, $d_p(t) = D_{p0}t$; $|D_{a0}| \leq 5 \cdot 10^{-3}$

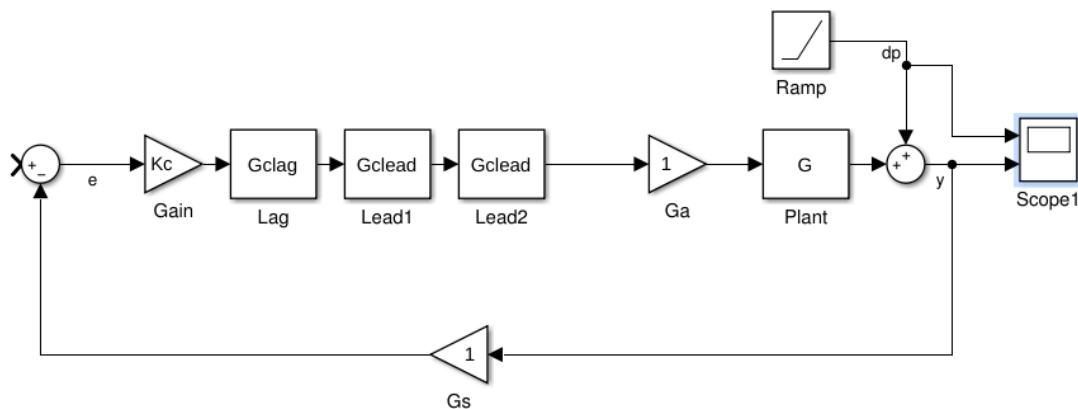


Figure 90- Simulink representation of the feedback system, disturbance error evaluation

A “Ramp” block is placed with the specified slope, and a “Scope” block is connected to the output signal. The reference signal was set to 0. The output is plotted in the scope block:

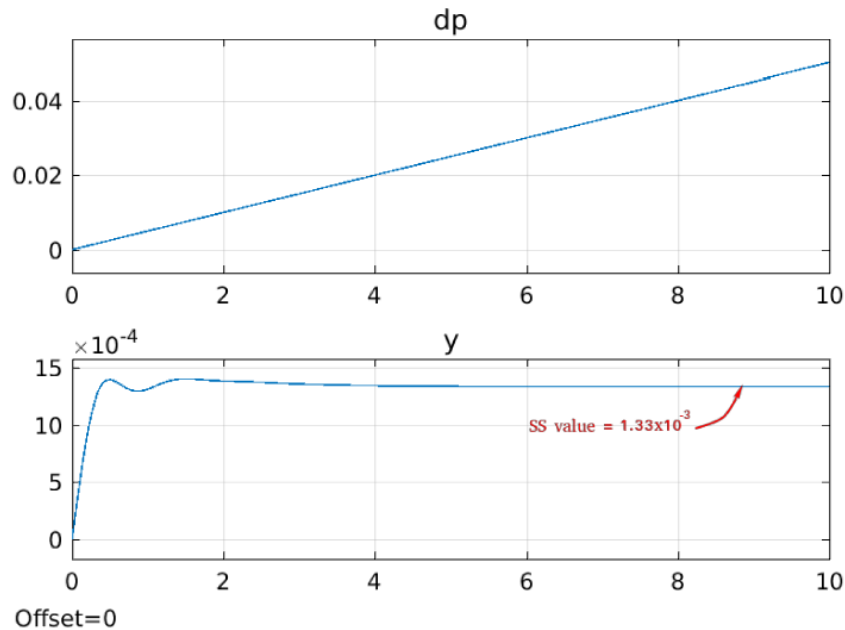


Figure 91- Output error for a ramp disturbance

$|y_{da}^{\infty}| = 1.33 \cdot 10^{-3} \leq 1.5 \cdot 10^{-3}$, the 2nd steady state requirement is also satisfied.

MATLAB Code

```

loopshapingdesign.m
1 - clear all, close all, clc
2 - %Defining The plant
3 - s=tf('s');
4 - G=5/(s*(s+1)*(0.25*s+1))
5 - Ga=1
6 - %Controller DC gain
7 - Kc=1.5
8 - %One stage of the Lead Compensator
9 - md=6
10 - zd=4/2.54
11 - Gclead=tf([1/zd 1],[1/(md*zd),1])
12 - %Bode plotting of the plant with 2 cascaded
13 - %Lead Compensators
14 - figure,
15 - bode(Kc*G, Kc*Gclead*Gclead*G)
16 - %One stage of the Lag Compensator
17 - mi=2
18 - zi=0.0003
19 - Gclag=tf([1/(mi*zi) 1],[1/zi,1])
20 - %Bode plotting of the compensated plant..
21 - figure,
22 - bode(Kc*G, Kc*Gclead*Gclead*G, Kc*Gclag*Gclead*Gclead*G)
23 - %Step response of the feedback system
24 - figure,
25 - step(feedback(Kc*Gclag*Gclead*Gclead*G,1))

```

USEFUL CONCEPTS IN PROBABILITY THEORY

The following chapter wants to introduce in a non-rigorous way some concepts that are useful in the framework of the **stochastic signals**, which are largely used to model **noise realizations** in a dynamical system.

RANDOM VARIABLES

Informally speaking, a random variable is a **function** that grants us the possibility to induce **measure of probability** on a measurable space. In other words, it allows us to have information about the probabilistic properties of a random quantity. For example, can tell us what is the probability that a particular signal takes some values at a given time. In general, a random variable can be a **one-dimensional or multi-dimensional**, depending on the aspects of the random phenomenon that we want to study in a probabilistic way.

We will manage only one-dimensional random variable, associated to the value that a one-dimensional random signal can assume at given time.

Furthermore, a random variable can be distinguished between:

- **Discrete** random variable
- **Continuous** random variable

In the first case, the values that the random variable can assume belongs to a numerable set, while in the second, the values belongs to a non-numerable set.

A random variable is completely described by means of its **probability distribution function** and its **probability density function**.

PROBABILITY DISTRIBUTION FUNCTION

The probability distribution function is a **real function** of real argument that measures the probability that occurs for the related random variable to assume values that are **smaller or equal to its argument**. For example, given a random variable X and the related distribution function $D_X(x^*)$, we have that:

$$D_X(x^*) = \gamma$$

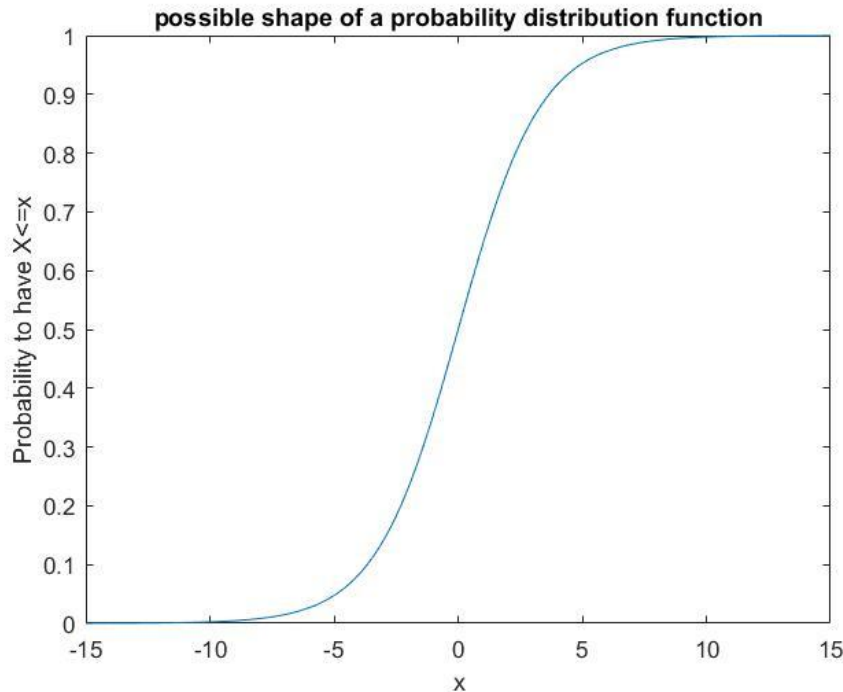
And the value γ is the probability that the random variable X assumes values x that are equal or less than x^* :

$$\gamma = \Pr\{X \leq x^*\}$$

Such a function must satisfy the following properties:

- $D_X(x) \geq 0$ for every value of x
- $D_X(x)$ is a monotonic non-decreasing function

- $\lim_{\varepsilon \rightarrow 0^+} D_X(x + \varepsilon) = D_X(x)$; that is, the distribution function is always a continuous function from the right
- $\lim_{x \rightarrow -\infty} D_X(x) = 0$; the probability must decrease to zero as the argument goes to minus infinity
- $\lim_{x \rightarrow \infty} D_X(x) = 1$; the probability for the random variable to assume values that are less than infinity must always be 1 (certain event)



This function represents a possible form for the distribution function; the independent variable x represents the values that the random variable X could assume; given a particular x^* , the value of the function is the probability to have $X \leq x^*$

PROBABILITY DENSITY FUNCTION

We also define the probability density function (PDF) as the **derivative of the probability distribution** function:

$$p_X(x) = \frac{dD_X(x)}{dx}$$

The reason for which the PDF is fundamental for the probabilistic characterization of the random variable is that its integral between two values gives back the **probability** for the random variable to assume the values that are **inside the integration interval**; this is true because the integral of the PDF is the probability distribution function, and then:

$$\int_a^b p_X(x) dx = D_X(b) - D_X(a)$$

The last expressions indicates the probability for X to be smaller than b and greater than a , that is:

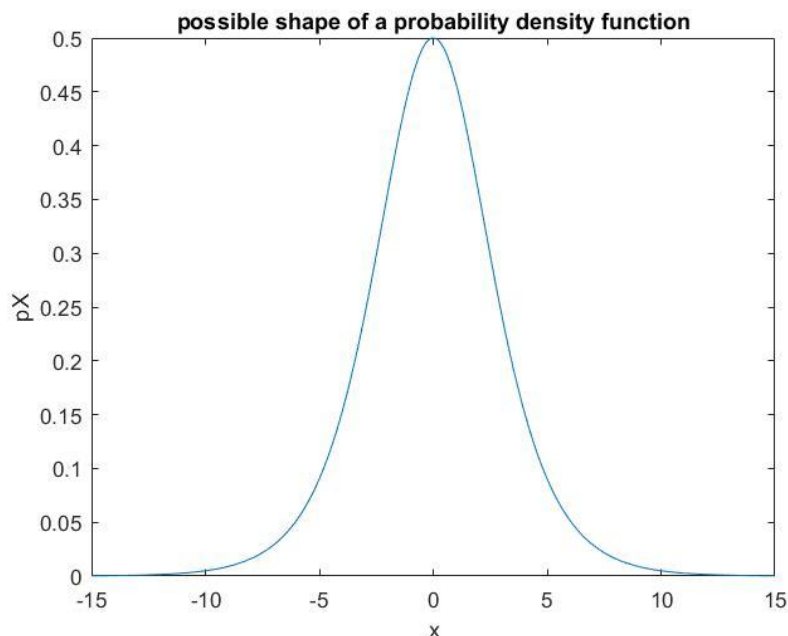
$$\int_a^b p_X(x) dx = \Pr\{a \leq X \leq b\}$$

The PDF must satisfy the following properties:

- $p_X(x) \geq 0$ for every possible x
- $\int_{-\infty}^{\infty} p_X(x) dx = 1$

The first property derives from the fact that the function represents the probability of an event and so **cannot be negative**; the second only reflects the fact that the probability that X assumes one value among all the possible values is equal to one, that is a certain event.

Note that the probability for a X variable to assume precisely a given real number is zero since it corresponds to the integral over a set of dimension zero.



In the graph is reported the PDF related to the distribution function of the previous paragraph.

EXPECTED VALUES AND MOMENTA

We now define some quantities that are useful to have a **characterization of a given random variable**.

Given a random variable X and the assigned PDF $p_X(x)$, we define:

Expected value with respect to $f(x)$.

Given the function $f(x)$, the expected value with respect to it is defined as:

$$E\{f(x)\} = \int_{-\infty}^{\infty} f(x)p_X(x) dx$$

It associates the function $f(x)$ to a number, given by the integral. The expected value is a **linear operator**, since it maintains the integral linear properties.

Among all the possible $f(x)$ the function class:

$$f(x) = x^n$$

Assumes particular importance and the relative expected values are called **momenta of order n**. In the following we list the most used expected values and their basilar interpretation:

- For $f(x) = x$ the expected value is simply called **expected value** and is the value that most probably the random variable will assume. It coincides with the ‘**barycenter**’ of the PDF and is indicated with the symbol m_x
- For $f(x) = x^2$ we have the **quadratic expected value**, $m_x^{(2)}$
- For $f(x) = (x - m_x)^2$ the expected value is called **variance** and it indicates how much the PDF is spread around its mean values, that is, how much the values that the random variable assumes are different from the expected value; is commonly denoted as σ^2 . Its square root is called **standard deviation**.

GAUSSIAN (NORMAL) PROBABILITY DENSITY FUNCTION

The Gaussian PDF is maybe the **most exploited** representation of random variables. Its expression is the following:

$$p_x(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\left(\frac{(x-m_x)^2}{2\sigma^2}\right)}$$

where the symbols represents the variance and the expected value of the PDF itself; since the Gaussian PDF is univocally determined when those two values are known, one can indicate it also with the expression $N(m_x, \sigma^2)$.

The Gaussian PDF has **many advantages**, that are maintained (or simply manageable) when it undergoes **linear time invariant transformations**. Its intensive use is also justified by important theoretical results, such as the Central limit theorem.

FURTHER ARGUMENTS

Many other concepts related to probability and statistics are useful in the dynamical system theory, in particular in the probabilistic framework identification of the systems the concepts of stochastic processes and multidimensional random variables are heavily exploited. Even if the main qualities related to the single random variable are still valid for a multidimensional random variable, their mathematical representations are not trivial (covariance matrix is an example of this fact).

Useful links:

‘Random signals and noise: a mathematical introduction’, S.Engelberg, CRC Press,2007

‘Probability, random signals and statistics’, X.R.Li, CRC Press 1999



<https://engineering.purdue.edu/~ipollak/ee438/FALL04/notes/Section2.1.pdf> from Purdue University