Aeronautical Engineering Program - Control of Dynamical Systems

D. Alazard[1]

[1] Institut Supérieur de l'Aéronautique et de l'Espace

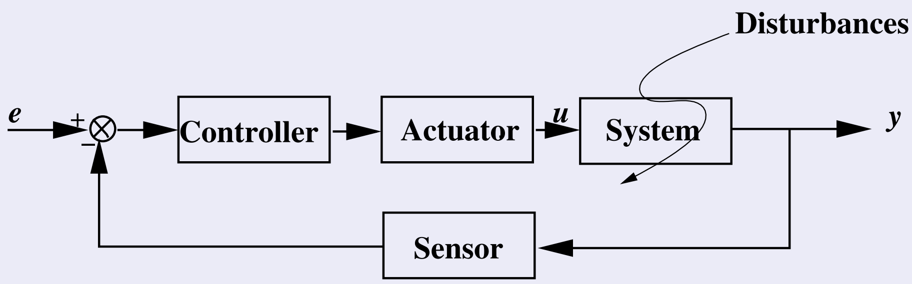
Outline

Outline

Closed-loop feedback control principle.

Compare the measurement of the output y with the desired value (input reference e) and apply a corrected input signal u on the system in order to :

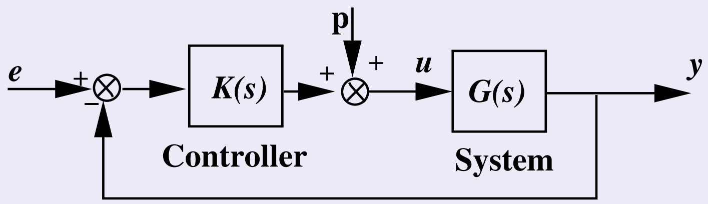
- stabilize the closed-loop system,
- track the reference input $e(t)$,
- reject eventual unknown disturbances (regulation).



General servo-loop set-up.

Closed-loop tranfers.

A simplified case (perfect actuator and sensor, disturbance only on the system input):



Closed-loop transfers:

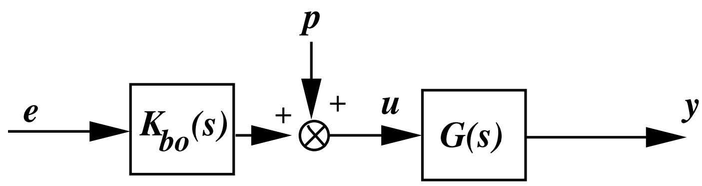
$$Y(s) = G(s)[P(s) + K(s)(E(s) - Y(s))]$$
$$Y(s) = \frac{K(s)G(s)}{1 + K(s)G(s)}E(s) + \frac{G(s)}{1 + K(s)G(s)}P(s)$$

- Transfer from input reference to system output: $\frac{Y}{E}(s) = \frac{K(s)G(s)}{1+K(s)G(s)}$,
- Transfer from disturbance signal to system output: $\frac{Y}{P}(s) = \frac{G(s)}{1+K(s)G(s)}$.

Outline

Performance objectives.

Ideally, we want: $Y(s) = 1.E(s) + 0.P(s)$ (i.e. perfect reference input tracking and perfect disturbance rejection)
$\Rightarrow L(s) = K(s)G(s)$ must be $>> 1$!!
($L(s)$: open-loop transfer function = the product of all transfers which are in the loop).
$G(s)$ is given, then $K(s)$ must be great.
Comparison with an open-loop control:



$$Y(s) = K_{bo}(s)G(s)E(s) + G(s)P(s)$$

$$\Rightarrow \widehat{K_{bo(s)}} = G(s)^{-1} \text{ (perfect reference input tracking)},$$

but this control cannot improve the natural capability of the system to reject disturbances.

Closed-loop feedback control principle | Performance and robustness objectives | Closed-loop stability - stability margins

Robustness against model uncertainties.

In all cases, the model is an approximation (linear approximation, some parameters are badly known, neglected sensor and actuator dynamics,...). Thus the direct inversion of the system in the controller ($\widehat{K_{bo(s)}} = G(s)^{-1}$) is very dangerous ! $\Rightarrow$ a closed-loop control is more robust (insensitive) to model uncertainties.
Indeed, let us consider a pure reference input tracking problem ($p(t) = 0$) and an open-loop controller $K_{bo}(s)$ which ensures the same transfer between the input reference and the output than the closed-loop controller $K(s)$ designed on the nominal model $G_0(s)$ of the system:

$$K_{bo}(s) = \frac{K}{1 + KG_0} \; .$$

Robustness against model uncertainties.

Let us assume that the real system is $G(s) = G_0(s) + \Delta G$: this model uncertainty ΔG will result in an output error:

- ΔY_{bo} on the output of the open-loop controlled system:

$$\Delta Y_{bo}(s) = K_{bo}(G - G_0)E(s) = \frac{K}{1 + KG_0}\Delta G\, E(s)$$

- ΔY_{bf} on the output of the closed-loop controlled system:

$$\begin{aligned}
\Delta Y_{bf}(s) &= \left[\frac{KG}{1+KG} - \frac{KG_0}{1+KG_0}\right]E(s) \\
&= \frac{K\,\Delta G}{(1+KG)(1+KG_0)}E(s) \qquad (1)\\
&= \frac{\Delta Y_{bo}(s)}{1+KG}\,. \qquad (2)
\end{aligned}$$

One more time, if $L = KG$ is great, the closed-loop controlled system will be more insensitive to the uncertainty ΔG than open-loop controlled system (ΔY_{bf} will be smaller than ΔY_{bo}).

Closed-loop feedback control principle | Performance and robustness objectives | Closed-loop stability - stability margins

Outline

Closed-loop stability.

From a practical view, it not possible to increase K as much as we want because the closed-loop system must be stable !!

Let us note: $L(s) = \frac{N(s)}{D(s)}$. It is easy to verify that all closed-loop transfers (from $E(s)$ to $Y(s)$, from $P(s)$ to $Y(s)$ et others...) have the same common denominator:

$$D_{bf}(s) = N(s) + D(s) \quad \text{closed-loop denominator} .$$

Thus $D_{bf}(s)$ must not have a pole in the right-hand half s-plane or along the imaginary axis.

if $\exists\, \omega \ / \ N(j\omega) + D(j\omega) = 0$, i.e. $1 + \frac{N(j\omega)}{D(j\omega)} = 1 + L(j\omega) = 0$,

then closed-loop system is on the stability boundary. Thus closed-loop stability is also entirely defined by the open-loop transfer $L(s)$.

In practice, we want that this stability be robust against uncertainties on nominal modal G_0.

$\Rightarrow$ stability margins notion.

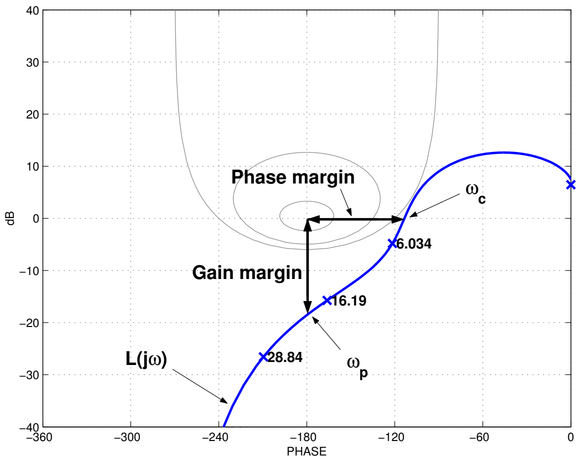
Stability margins.

Prevention against model errors: we seek a closed loop system whose stability supports:

- variations on the steady-state gain of G: $G \rightarrow k\,G$ (for instance: uncertainty on the actuator gain)
  $\Rightarrow$ Gain margin,
- variations on the phase of G: $G \rightarrow e^{-j\phi}G$ (for instance: phase lag due to actuator dynamics, discretization , ...)
  $\Rightarrow$ Phase margin,
- a parasite delay T in the servo-loop: $G \rightarrow e^{-Ts}G$
  $\Rightarrow$ Delay margin.

These stability margin notions give a qualitative information on the distance to the unstability and have a graphical interpretation on the frequency response of the open loop transfer $L(j\omega)$: they indicate the distance of $L(j\omega)$ to the critical point (i.e. the “point: -1”: magnitude$=1$ and phase$=-180\,deg$).

Interpretation in Nichols plot.



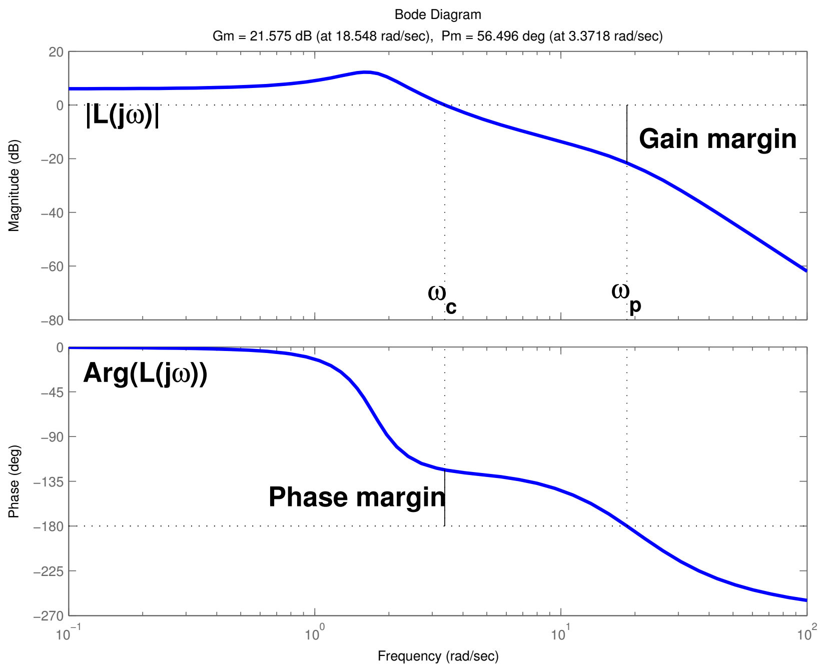
$$\text{gain margin} = 20\log_{10}|L(j\omega_p)|$$

$$\text{phase margin} = 180 + Arg[L(j\omega_c)]$$

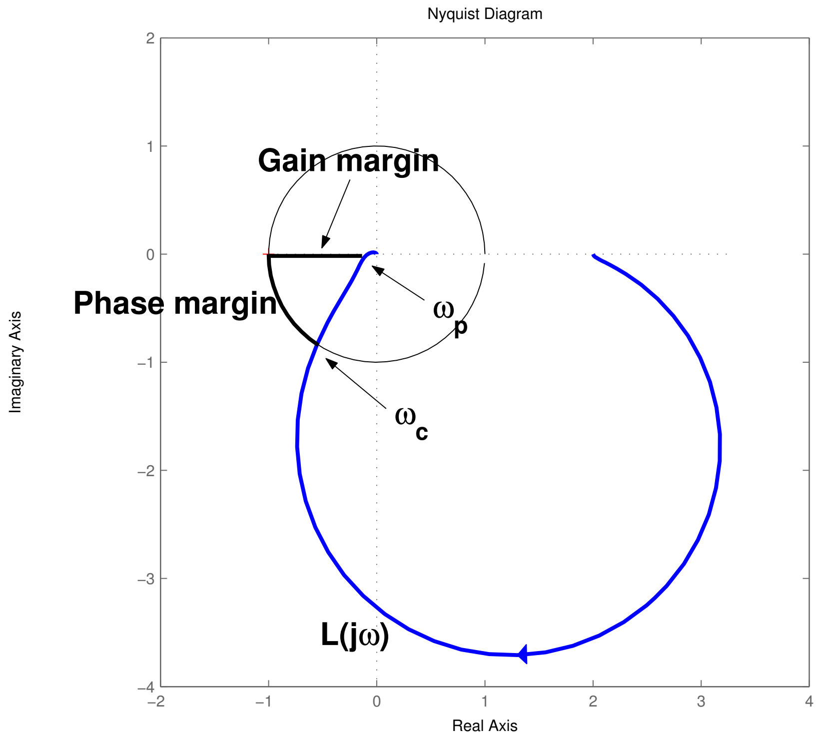
$$\text{delay margin} = \text{phase margin}\frac{\pi}{180\omega_c}$$

Closed-loop feedback control principle Performance and robustness objectives Closed-loop stability - stability margins

Interpretation in Bode plot.



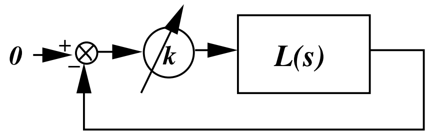
Interpretation in Nyquist plot.



Outline

Root locus (Evans locus).

We saw previously that the dynamic behavior of a system depends strongly on its poles (modal analysis). The root locus allows to qualitatively appreciate, in the complex s-plane (Im(s) vs. Re(s)), the path of the closed-loop poles when the loop gain k varies from 0 to ∞ according to figure below:



(to study k variations from $-\infty$ to 0, one can change the sign of $L(s)$).
Qualitative approach: construction rules are based only on the poles and zeros of the open-loop transfer function $L(s) = K(s)G(s)$.
Control design based on root locus: the principle consists in adding some poles and zeros (in $K(s)$) in order to drive poles of $G(s)$ far in left-hand half s-plane.
The root locus is symmetrical with respect to the horizontal real axis.

Root locus - construction rules.

$$L(s) = G_e \frac{\Pi_{i=1}^{m}(s-z_i)}{\Pi_{i=1}^{n}(s-p_i)} \quad \text{then: } D_{bf}(s) = \Pi_{i=1}^{n}(s-p_i) + k\, G_e \Pi_{i=1}^{m}(s-z_i)\ .$$

When $k = 0=$, the n closed-loop poles are the n open-loop poles (p_i).
When $k \to \infty$, the n closed-loop poles are attracted towards:

- the m open-loop zeros,
- $n-m$ asymptotes:
  - whose angles with real axis are $\frac{(2l-1)\pi}{n-m}$ $l = 1, \cdots, n-m$ if $G_e > 0$,
  - whose angles with real axis are $\frac{2l\pi}{n-m}$ $l = 1, \cdots, n-m$ if $G_e < 0$,
  - which are centered at a point on the real axis given by: $\frac{\Sigma_1^n p_i - \Sigma_1^m z_i}{n-m}$.

rk:

- the root locus on the real axis lies in a section of the real axis to the left of an odd number of poles and zeros if $G_e > 0$,
- the root locus on the real axis lies in a section of the real axis to the left of an even number of poles and zeros if $G_e < 0$.

Root locus - construction rules (cont').

These rules can be proved using the **angle condition**.
Indeed, the complex s belongs to the root locus if:

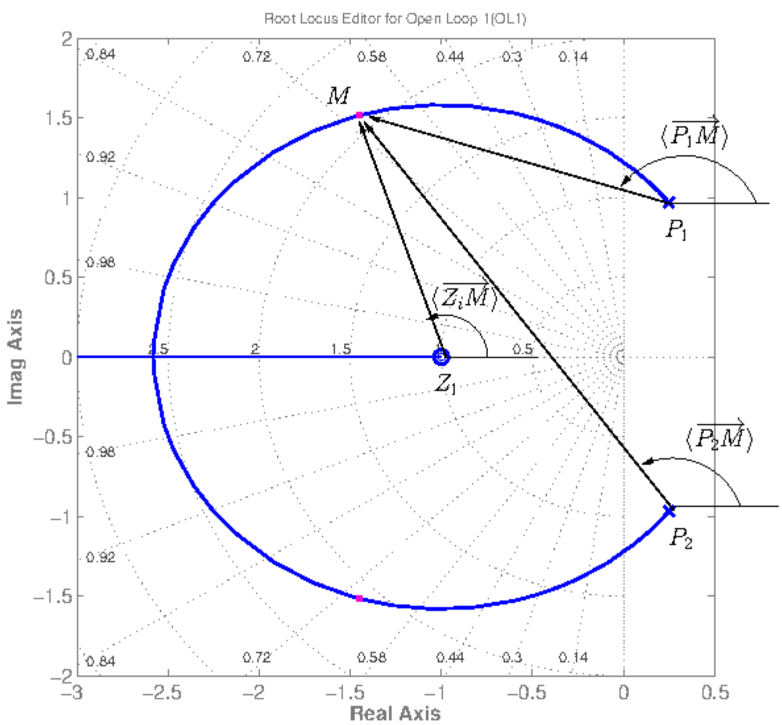
$$\exists k \quad / \quad G_e \frac{\Pi_{i=1}^{m}(s-z_i)}{\Pi_{i=1}^{n}(s-p_i)} = -1/k \,.$$

$$\begin{aligned}\Rightarrow \sum_{i=1}^{m} \text{Arg}(s-z_i) - \sum_{i=1}^{n} \text{Arg}(s-p_i) &= 180\,(\pm 360 l)\, deg \quad \text{if } kG_e > 0 \\ &= 0\,(\pm 360 l)\, deg \quad \text{if } kG_e < 0\end{aligned}$$

Considering the points M, P_i and Z_i in the complex plane associated with complex numbers s, p_i and z_i respectively, M belongs to the roots locus if:

$$\begin{aligned}\Rightarrow \sum_{i=1}^{m} \langle \overrightarrow{Z_i M} \rangle - \sum_{i=1}^{n} \langle \overrightarrow{P_i M} \rangle &= 180\,(\pm 360 l)\, deg \quad \text{if } kG_e > 0 \\ &= 0\,(\pm 360 l)\, deg \quad \text{if } kG_e < 0\end{aligned}$$

Root locus - angle condition illustration.



Root locus - construction rules (cont').

Other rules:

- let α_j the angle of the tangente of the branch at point P_j, then:

$$
\begin{aligned}
\alpha_i &= \sum_{i=1}^{m} \langle \overrightarrow{Z_i P_i} \rangle - \sum_{i=1, i\neq j}^{n} \langle \overrightarrow{P_i M} \rangle - 180\,(deg) \quad \text{if } kG_e > 0 \\
&= \sum_{i=1}^{m} \langle \overrightarrow{Z_i P_i} \rangle - \sum_{i=1, i\neq j}^{n} \langle \overrightarrow{P_i M} \rangle \quad \text{if } kG_e < 0
\end{aligned}
$$

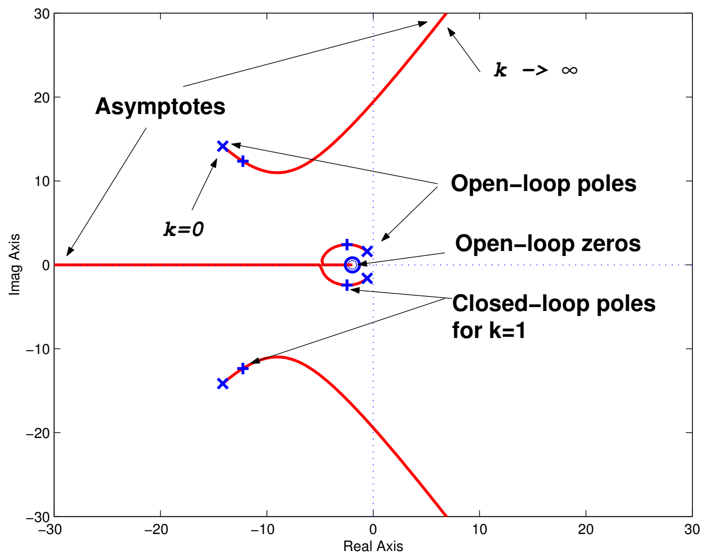
- let β_j the angle of the tangente of the branch at point Z_j, then:

$$
\begin{aligned}
\beta_i &= -\sum_{i=1, i\neq j}^{m} \langle \overrightarrow{Z_i P_i} \rangle + \sum_{i=1}^{n} \langle \overrightarrow{P_i M} \rangle + 180\,(deg) \quad \text{if } kG_e > 0 \\
&= -\sum_{i=1, i\neq j}^{m} \langle \overrightarrow{Z_i P_i} \rangle + \sum_{i=1}^{n} \langle \overrightarrow{P_i M} \rangle \quad \text{if } kG_e < 0
\end{aligned}
$$

- if 2 (auto-conjugate) branches converge to the same point on the real axis (or start from the same point on the real axis), the tangentes of these branches at the intesection are perpendicular to the real axis.

Root locus - example.

$$
L(s) = 1000 \frac{s+2}{(s^2 + \sqrt{2}\,20\,s + 20^2)(s^2 + s + 3)}
$$



Routh stability criterion.

Considering the polynome:

$$P(s) = a_n s^n + a_{n-1} s^{n-1} + a_{n-2} s^{n-2} + \cdots + a_2 s^2 + a_1 s + a_0$$

The **Routh stability criterion** is a test on the coefficients a_i to check that all the roots of $P(s)$ have a negative real part. This test is: **all the coefficient $c(i,1)$ ($i = 1, \cdots, n+1$, that is the first colmun) in the following table must have the same sign.**

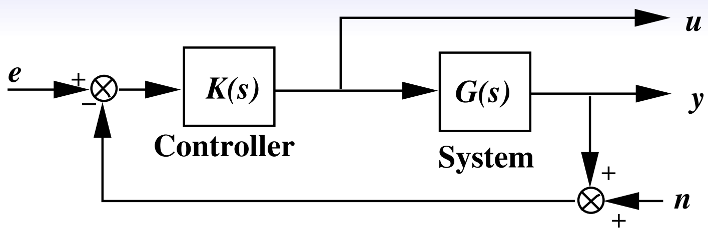
s^n	$c(1,1) = a_n$	$c(1,2) = a_{n-2}$	$c(1,3) = a_{n-4}$	$\cdots$	0
s^{n-1}	$c(2,1) = a_{n-1}$	$c(2,2) = a_{n-3}$	$c(2,3) = a_{n-5}$	$\cdots$	0
$\vdots$	$\dots$	$\dots$	$\dots$	$\dots$	$\dots$
s^{n+1-i}	$c(i,1)$	$\cdots$	$\cdots$	$c(i,j)$	$\cdots$
$\vdots$	$\dots$	$\dots$	$\dots$	$\dots$	$\dots$
s^0	$c(n+1,1)$	0	0	0	0

with: for $i = 3, \cdots, n+1$, $c(i,j) = c(i-2, j+1) - \frac{c(i-2,j)c(i-1,j+1)}{c(i-1,j)}$ if $c(i-1,j) \neq 0$, $c(i,j) = 0$ otherwise.

In the root locus method, ROUGH criterion can be used to find the condition on the loop gain k for the closed system to be stable.

Outline

Influence of measurement noise n.



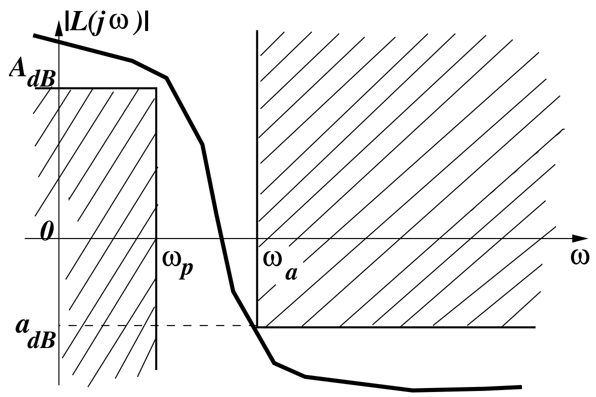
Transfer from n to y and from n to u:

$$\frac{Y(s)}{N(s)} = \frac{-KG}{1+KG} \quad \text{et} \quad \frac{U(s)}{N(s)} = \frac{-K}{1+KG}$$

Actuator health: $\Rightarrow K$ must be small!!
Great output to noise ratio: $\Rightarrow L = KG$ must be small!!
Trade-off: performance $\Rightarrow K \nearrow$; noise attenuation $\Rightarrow K \searrow$.
This trade-off must be balanced (and tuned) according to frequency:

- performances: low-frequency behavior,
- measurement noise: high frequency behavior.

Frequency-domain design on the open-loop transfer: trade-off.



A typical frequency-domain template on $|L(j\omega)|$ to manage trade-off between low frequency performance and high frequency noise rejection:

$$\forall\, \omega < \omega_p, \quad |L(j\omega)| > A >> 1 \quad \Rightarrow \quad S = (I+L)^{-1} << 1 \;\Rightarrow \text{performance,}$$

$$\forall\, \omega > \omega_a, \quad |L(j\omega)| < a << 1 \quad \Rightarrow \quad T = L(I+L)^{-1} << 1 \;\Rightarrow \text{noise rejection}$$

Frequency-domain design.

We shape with $K(s)$ the frequency response of $L(s) = K(s)G(s)$ using:

- phase lead filters: $K_1(s) = \frac{1+\tau s}{1+a\tau s}$ (with $a < 1$),
- phase lag filters: $K_2(s) = \frac{1+a\tau s}{1+\tau s}$ (with $a < 1$),
- low-pass filters, notch filters,...
- integrators (to perfectly reject very low frequency disturbances)
- ...

But warning: there are some intrinsic limitations:
<u>Bode law</u>: a stable filter whose gain decreases when ω goes from 0 to ∞ ($|G(0)| > |G(j\infty)|$) while its phase increases ($Arg(G(0)) < Arg(G(j\infty))$) does not exit...

Closed-loop feedback control principle Performance and robustness objectives Closed-loop stability - stability margins

Outline

Servo-loop accuracy.

If we want a null steady state error on the step response in spite of uncertainties on the steady state $G(0)$ of the system $G(s)$. $\Rightarrow L(s)$ must have at least one integrator:

$$L(s) = K(s)G(s) = \frac{N(s)}{D(s)} \quad \text{avec } D(s) = sD'(s)$$

then $\frac{Y(s)}{E(s)} = \frac{N(s)}{sD'(s)+N(s)}$ whose steady state [1] is equal to 1 $(\forall\, G(0) \neq 0)$.

If $G(s)$ reveals one integrator then no problem, else, we have to include it in the controller $K(s)$.

Exercise: Show that $L(s)$ must have 2 integrators to cancel the steady state error on a ramp response.

[1] Rk: the steady state of a transfer $G(s)$ is $G(s)|_{s=0}$ and is equal to its step response final value $y(\infty)$.

Outline

State feedback design - Single input case.

Principle: Considering a n-th order linear system given by its state space representation and assuming that all the state variables are measured:

$$\begin{cases} \dot{x} &= Ax + Bu, \quad x \in \mathbb{R}^n,\ u \in \mathbb{R} \\ y &= x \end{cases}$$

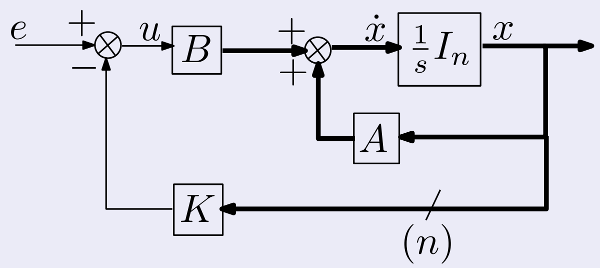
the objective is to compute a state feedback gain $K_{1\times n}$ such that the control law $u = e - Kx$ assigns the n closed-loop eigenvalues to prescribed values λ_i, $i = 1, \dots, n$.

λ_i are the tuning parameters, chosen such that the closed loop system is:

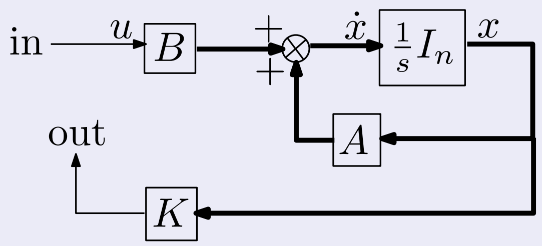
- stable: $\text{Re}(\lambda_i) < 0\ \forall\, i$,
- correctly damped: $-\text{Re}(\lambda_i)/\sqrt{\lambda_i \bar{\lambda}_i} \approx 0.7$,
- fast enough (for the dominant dynamics): $\text{Re}(\lambda_i) < -1/\tau$,
- not too far from the open-loop dynamics: $\text{Spec}(A)$,
- $\dots$

State feedback design - Single input case.

Closed-loop system representation: $\dot{x} = (A - BK)x + Be$



Open-loop transfer function $L(s)$ for a state feedback:
$L(s) = K(sI_n - A)^{-1}B$



Stability margins (in MATLAB): `margin(ss(A,B,K,0))`.

State feedback - Compution of the gain K.

Assumption: the pair (A, B) is controllable.
$\Rightarrow$ 2 methods
Method # 1: Direct identification of the closed-loop characteristic polynome.
Let $K = [k_0,\ k_1,\ \cdots,\ k_{n-1}]$ then

$$\det(sI_n - A + BK) = \Pi_{i=1}^{n}(s - \lambda_i) = s^n + a_{n-1}s^{n-1} + a_{n-2}s^{n-2} + \cdots + a_1 s + a_0\,.$$

The identification of polynome coefficients leads to a set of n equations with n unknows $(k_i,\ i = 0, \cdots, n-1)$ to be solved.

Example:

$$\dot{x} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 10 \\ 0 & 0 & -10 \end{bmatrix} x + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} u$$

Compute a state feedback K such that the closed-loop dynamics is governed by a second order with a pulsation ω (> 0) and a damping ratio ξ (> 0).

State feedback - Compution of the gain K.

Method # 2: assumption: λ_i are disctinct.

$$\lambda_i \in \text{Spec}(A - BK) \Leftrightarrow \exists\ v_i(\text{eigenvector})/(A - BK)v_i = \lambda_i v_i$$

$$\Leftrightarrow \begin{bmatrix} v_i \\ w_i \end{bmatrix} \in \text{Ker}\left([A - \lambda_i I_n \quad B]\right) \quad \text{with} \quad w_i = -Kv_i\,.$$

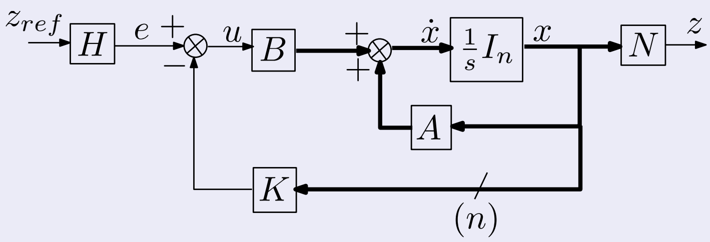
The direction $[v_i^T\ w_i^T]^T$ (up to a factor) must be computed for each λ_i.
Then K must satisfy: $w_i = -Kv_i,\ \forall i$

$$\Rightarrow K = -[w_1\ w_2\ \cdots\ w_n][v_1\ v_2\ \cdots\ v_n]^{-1} = -WV^{-1}\,.$$

Example: solve the previous example (slide # 33) using MATLAB and the function `null`.

State feedback and feedforward

Considering a single output $z = Nx$ to be "servo-looped" on the reference input z_{ref} using a state-feedback gain K and a feedforward gain H: $\Rightarrow u = Hz_{ref} - Kx$.



$$\Rightarrow \frac{Z}{Z_{ref}}(s) = N(sI_n - A + BK)^{-1}BH.$$

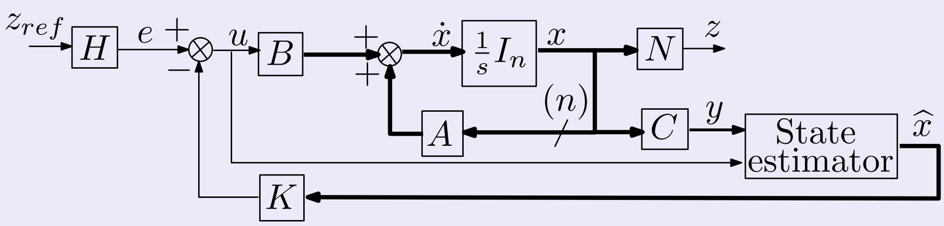
H can be computed to impose an unitary DC-gain between z_{ref} and z:

$$H = -\left(N(A - BK)^{-1}B\right)^{-1}.$$

Example: Considering the example of slide # 33, compute H to servo-loop the first state $x(1)$.

State feedback control: limitations

From a practical point of view, all the state variables are not always measured. Let $y = C_{p\times n}x$ the vector of the p measured outputs. Then, the state feedback gain K can be implemented using a state estimator: $u = Hz_{ref} - K\widehat{x}$.



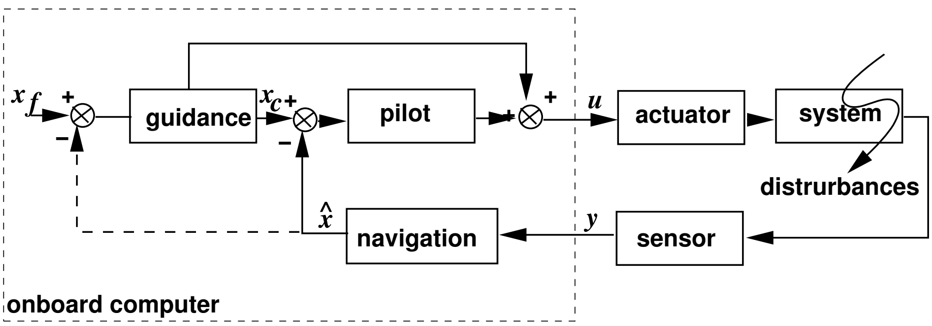
In the field of Aerospace vehicle control, such a state estimator is also called a **Navigation Filter** (see course on KALMAN filter).
Alternative: one can also used a static output feedback $K_y \in \mathbb{R}^{1\times p}$such that the control law $u = Hz_{ref} - K_y y$ assigns **only** p closed-loop eigenvalues to prescribed values λ_i, $i = 1, \cdots, p$.
Then (using the method # 2):

$$K_y = -[w_1 \;\cdots\; w_p]\left(C[v_1 \;\cdots\; v_p]\right)^{-1}; H = -\left(N(A - BK_yC)^{-1}B\right)^{-1}.$$

Outline

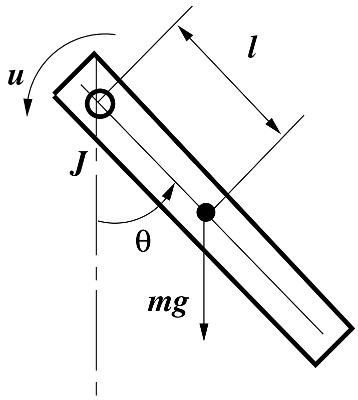
Multi-loop control architecture (GNC).

Several loops to control several dynamic behaviors.



- Guidance: (of low frequency behavior) trajectory computation (open loop or closed-loop), optimal approach (ex: minimal total time), simple non-linear model (ex: $|u(t)| < u_{max} \; \forall t$),
- Control (pilot): (of high frequency behavior) servo-loop on the trajectory to reject disturbances, full linear model.
- Navigation: state estimation from measurement and model.

Example: the pendulum.



Dynamic principle:

$$J\ddot{\theta}(t) + mlg\sin\theta(t) = u(t)$$

Linear approximation around an equilibrium position θ_e:

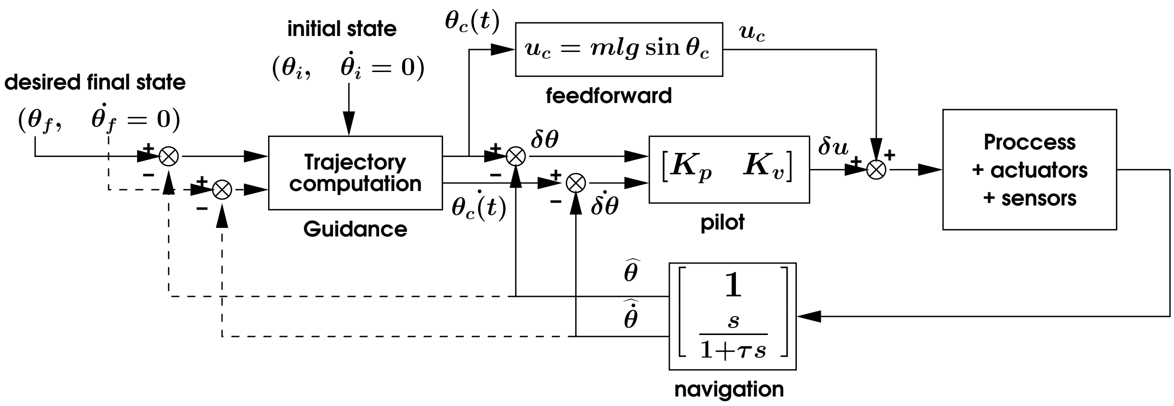
$$\theta = \theta_e + \delta\theta, \quad u = mlg\sin\theta_e + \delta u$$

Assumptions: small movings

$$\begin{bmatrix} \delta\dot{\theta} \\ \delta\ddot{\theta} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{mlg\cos\theta_e}{J} & 0 \end{bmatrix} \begin{bmatrix} \delta\theta \\ \delta\dot{\theta} \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{J} \end{bmatrix} \delta u$$

$\mathcal{L}$aplace : $\quad G(s) = \frac{\delta\theta}{\delta u}(s) = \frac{1}{Js^2 + mlg\cos\theta_e}$: $\theta_e = 0 \rightarrow$ oscillating mode; $\theta_e = \pi \rightarrow$ 1 unstable real m

Example: the pendulum (cont.).



Pilot inner loop dynamics: $D_{bf}(s) \approx (Js^2 + K_v s + K_p + mlg\cos\theta_c)$
Pilot design: tune K_p and K_v to damp and accelerate roots of D_{bf} $\forall\ \theta_c$ (if it is not possible, then gain-scheduling of K_p and K_v according to θ_c: $K_p = K_p(\theta_c)$ et $K_v = K_v(\theta_c)$).

Exercices, Problems and Matlab labworks on Dynamic Systems Modeling, Analysis and Control

D.Alazard

May 2011 - version 0.0

EMPTY PAGE

Contents

Institut Supérieur de l'Aéronautique et de l'Espace

EMPTY PAGE

Introduction

This document is a set of exercices, problems on dynamics systems modeling, analysis and control:

- chapter: 1 Dynamic system modeling and analysis,
- chapter: 2 Dynamic system control,
- chapter: 3 MATLAB labworks.

EMPTY PAGE

Chapter 1

Dynamic system modeling and analysis

1.1 Electrical cicruit # 1

Give a state space representation of the electrical circuit depicted in Figure 1.1 (u and y are the system input and ouput, respectively).

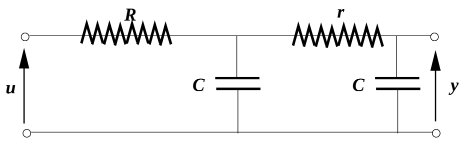


Figure 1.1: System $\mathcal{S}$ (C, r and R denote 1 capacitor and 2 resistors, respectively).

Give the transfer function of system $\mathcal{S}$.

1.2 Electrical cicruit # 2

Give a state space representation of the electrical circuit depicted in Figure 1.2 (u and y are the system input and ouput, respectively).

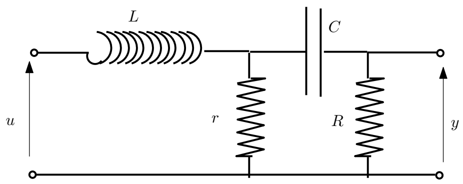


Figure 1.2: System $\mathcal{S}$ (C, r, R and L denote 1 capacitor, 2 resistors and 1 inductance, respectively).

Give the transfer function of system $\mathcal{S}$.

1.3 Transfer to state-space conversion #1

Give a state space representation for the following transfer (u and y are the system input and ouput, respectively) :

$$\frac{Y}{U}(s) = \frac{4s^2 - 1}{2s^2 + 3s + 2} \ .$$

1.4 State-space equation integration (continuous-time)

Let us consider the system described by the following input-ouput differential equation:

$$\ddot{y}(t) + 2\dot{y}(t) = u(t) \ .$$

- a) find the transfer function of this model and a state-space representation considering that this system can be seen as the series connection of two first order system,
- b) compute the system impulse response,
- c) compute the response of the system to initial conditions $y(0) = y_0$ and $\dot{y}(0) = \dot{y}_0$ (with $t_0 = 0$).

Remark to solve questions b) and c) one will use the most suitable representation.

1.5 Discrete-time system

Let us consider the system depicted in Figure 1.5 where the analog $PLANT$ is defined by a transfer function: $\frac{Y}{U}(s) = F(s) = \frac{1}{s^2-1}$, $B_o(s)$ is a zero-order hold and dt is the sampling period.

a) give a state space representation of $F(s)$,

b) give a state space representation of the discrete-time transfer between u_n and y_n,

c) give the z-transfer function between u_n and y_n.

1.6 z-transform

Let us consider the discrete-time system (between the input $u(n)$ and the output $y(n)$) defined by the z-transfer function: $F(z) = \frac{Y(z)}{U(z)} = \frac{z^2+1}{z^2-1}$:

a) compute and draw its impulse response (only the 10 first samples),

b) compute and draw its step response (only the 10 first samples),

c) give the time-domain recurrent equation associated with this filter,

d) using this recurrent equation, find again the impulse and the step responses.

EMPTY PAGE

Chapter 2

Dynamic system control

2.1 Oven temperature regulation

The transfer function of an electrical heating oven from the input tension (u) to the inside temperature (output: $y = \theta$) can be approximated by a first order model:

$$\frac{Y}{U}(s) = F(s) = \frac{G}{1 + Ts}$$

where parameters G and T depend on the thermal capacitance of the oven (fixed) and its load (unknown). Thus $G = G_0 + \Delta G$ and $T = T_0 + \Delta T$ where G_0 and T_0 are the nominal values of these parameters.

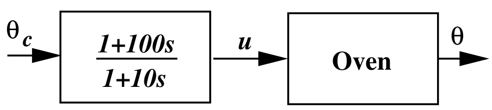
N.A.: $T_0 = 100\,s$, $G_0 = 1\,°C/Volt$, $\Delta T \in\ [-30\ \ +30]\,(s)$ and $\Delta G \in\ [-0.2\ \ +0.2]\,(°C/Volt)$.

Tracking performance on the nominal model

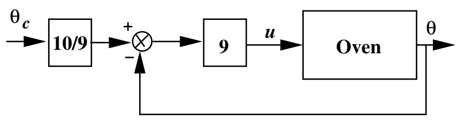
We want θ to track a desired profile constituted of several steps (see Figure below). The natural settling time of the oven ($300\,s$) is too slow to follow such a profile (assume for instance that $t_1 = 1mn$).

Two control strategies are proposed to boost the system dynamics:

- open loop strategy:



- closed-loop-strategy:



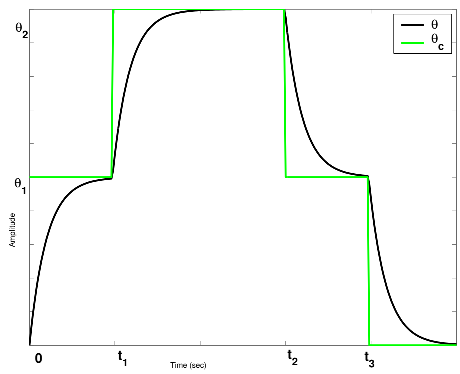


Figure 2.1: Oven typical response.

For the nominal model ($F_0(s) = F(s)|_{T=T_0,G=G_0}$) only and for both control strategies, compute the input/ouput transfer of the controlled system: $\theta/\theta_c(s)$. Check that they are the same and plot the step response of this transfer. Compare it with the system step response $F_0(s)$.

Disturbance rejection on the nominal model

During the heating phase, it must be possible to open the oven door (to add a new load for instance). The regulation must hold the temperature θ to the reference value θ_c as much as possible. Such a perturbation is modeled as an impulse signal $p(t)$ on the input of the system (see Figure below).

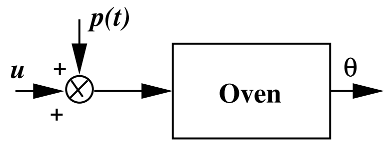


Figure 2.2: Perturbation on the system input.

Plot the impulse response of the transfer $\theta/P(s)$ for both control strategies and comment the result.

Robustness to uncertainties

For both control strategies: compute, in steady state of the step response of transfer $\theta/\theta_c(s)$, the maximal temperature variation $\Delta\theta$ due to the model variation ΔT. Do the same thing with variation ΔG and comment the results.

Integral control for robust tracking performance

Consider the new closed-loop control depicted in Figure below.

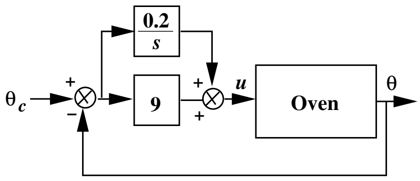


Figure 2.3: Perturbation on the system input.

Compute the closed-loop transfer $\theta/\theta_c(s)$ and verify that the steady state error on the step response is null whatever ΔT and ΔG.

Actuator dynamics

In fact, the heating resistor has its own dynamics $R(s)$. Assuming that $R(s) = \frac{\omega}{s+\omega}$, analyse qualitatively (using roots loci for various positive values of ω) and comment the influence of the resistor dynamics on the closed-loop dynamics. Do the same thing assuming that $R(s) = \frac{\omega^2}{s^2+1.4\omega s+\omega^2}$.

EMPTY PAGE

Chapter 3

MATLAB labworks

3.1 General remark:

Each of exercices of chapters 1 and 2 can be solved using MATLAB/SIMULINK (except for pure modeling steps). The main MATLAB functions to be used for each exercices are given in the following list. You are advised to read the `help` of these functions before to use them:

- exercices 1.1, 1.2 : `ss`, `ss2tf` and `zpk`,
- exercice 1.3 : `tf`, `tf2ss`,
- ex. 1.4: `impulse`, `initial`,

3.2 Pendulum

Let us consider the pendulum depicted in Figure 3.1. The dynamic behaviour of this pendulum can be described by following input-ouput differential equation:

$$J\ddot{\theta}(t) = u(t) - f\dot{\theta}(t) - mlg\sin\theta(t)$$

where $u(t)$ (Nm) is the torque applied by the actuator (electrical motor) and f (Nms) is the coefficient of the viscous friction inside the pivot joint.

- Describe directly this differential equation by a SIMULINK block diagram (N.A.: $m = 1\,Kg$, $l = 0.2\,m$, $J = ml^2$ (case of a concentrated mass at the pendulum tip), $g = 9.81\,m/s^2$ and $f = 0.004\,Nms$).
- Determine:
  - equilibrium conditions (control signal u_e and internal state vector x_e) imposing $\theta = \theta_e$ (macro-function `trim`),

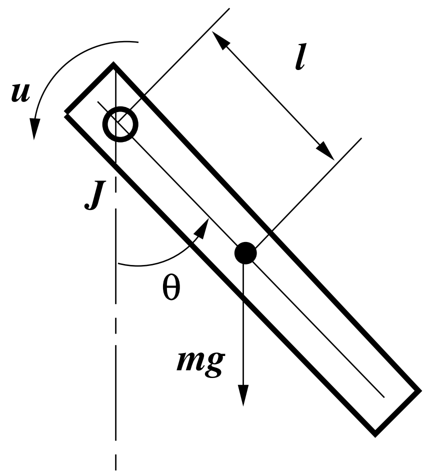


Figure 3.1: The pendulum.

– the linear model around these equilibrium conditions (macro-function `linmod`)
– the eigenvalues of this linear model,

for different values of θ_e (0, $\pi/2$, π).

- Simulate the response to initial conditions on angular rate ($\theta(t=0)=0$, $\dot{\theta}(t=0)=\dot{\theta}_0$) for different values $\dot{\theta}_0 = 1\,rd/s$, $\dot{\theta}_0 = 15\,rd/s$; from the linear model (macro-fonction `initial`) and from the SIMULINK file describing the non-linear model and compare both responses.

3.3 Bamoss: Flexible structure modeling, analysis and control

The experimental testbed (developped in ISAE/DMIA) studied in this practical labwork is depicted in the Figure 3.2. A video on the dynamic (open-loop and closed-loop) behavior of this testbed can be downloaded at `http://personnel.supaero.fr/alazard-daniel/demos/film_bamoss_ve.mpg`.

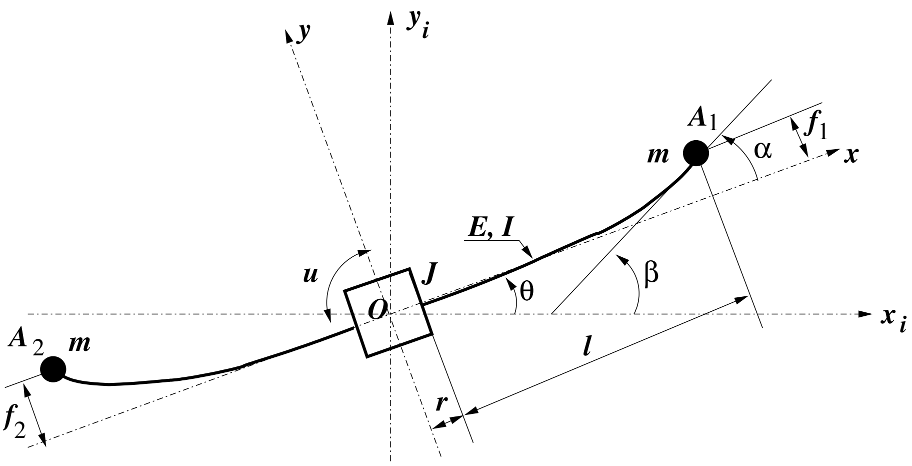


Figure 3.2: Bamoss simplified sketch.

This testbed works in the horizontal plane $R_i = (0,\ x_i,\ y_i)$ (r_i is an inertial frame) and is composed of:

- a hub with a square shape ($R = (0,\ x,\ y)$ is the hub-body frame), articulated w.r.t. the inertial frame by a pin-join around a vertical axis $(0,\ z_i)$. The half-side and the inertia (around $(0,\ z_i)$) of this hub are denoted r and J, respectively,
- a torque motor, driving the hub in rotation around the vertical axis $(0,\ z_i)$,
- 2 identical flexible beams (in the horizontal plane) cantilevered on the hub. The sizes of each beam are l (length), b (width) and h (thickness in the horizontal plane) and the young modulus of the beam material is denoted E,
- 2 local masses m fitted at the tip of each beam,
- an optical encoder to measure the angular position θ of the hub w.r.t. the inertial frame,
- a tachometer to measure the angular rate $\dot{\theta}$ of the hub w.r.t. the inertial frame,

- a gyrometer to measure the inertial angular rate $\dot{\beta}$ at one tip mass (at point a_1).

Let us denote :

- $u\,(Nm)$: the driving torque (control signal),
- $\theta\,(rd)$: angular position of the hub,
- f_1 and $f_2\,(m)$: lateral deflections at the free end of each beam,
- $\alpha\,(rd)$: the angular deviation (slope, w.r.t. to equilibrium position) at the free-end of the beam fitted with the gyrometer,
- $\beta = \theta + \alpha$,
- $S(s)$ the LAPLACE transformation of a signal $s(t)$.

The objective of the lab-work is to model the dynamic behavior of this system (applying structural dynamics theory) and to use MATLAB tools (applying automatic control theory) to analyse this model and to design a controller for the regulation of this system (servo-loop) around the position $\theta = 0$. the servo-loop performances will be analyzed by comparison of the open-loop and closed-loop responses to initial conditions. a main goal consists in simulating (using MATLAB) the experimental response presented in figure 3.3, obtained in the following conditions:

- at time $t = 0$, the following initial conditions are manually prescribed (see video):

$$\theta(t=0) = 0.18 \text{ deg}, \quad f_1(t=0) = 4\,mm, \quad f_2(t=0) = 0.5\,mm, \ldots$$

$$\ldots\dot{\theta}(t=0) = 20 \text{ deg}/s, \quad \dot{f}_1(t=0) = -7\,cm/s, \quad \dot{f}_2(t=0) = 18\,cm/s\ ,$$

- at time $t = 5\,s$, the feedback-loop is closed (on a proportional-derivative controller).

Assumptions:

- only motions in the horizontal plane are considered,
- masses and inertia of beams are neglected (inertia of local masses at each beam tip are also neglected),
- only small motions are considered.

Numerical application: $J = 0.015\,kg\,m^2$, $l = 0.286\,m$, $E = 200\,10^9\,N/m^2$, $m = 0.30\,kg$, $r = 0.05\,m$, $h = 0.64\,mm$, $b = 4\,cm$.

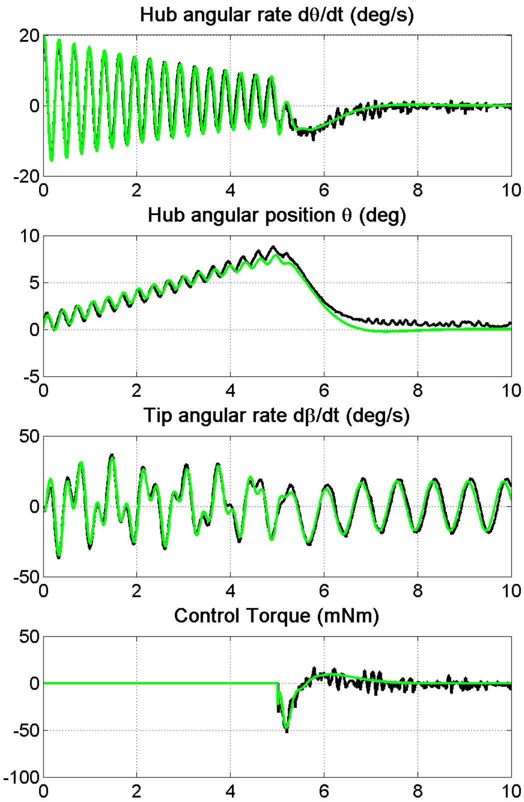


Figure 3.3: Experimental (black) and simulated (green) responses to initial conditions.

3.3.1 Dynamic modeling

Static deflection of a beam

Let us consider a single cantilever beam with a tip load P at its free-end (see following figure 3.4).

Question A.1: compute the lateral deflection f at the free-end and deduce the equivalent stiffness k of the beam (i.e. such that : $P = kf$) as a function of EI (I is the section quadratic moment) and l. Express the slope α at the free-end as a function of f and l.

The flexible beam can be now considered as a string with stiffness k acting between the mass m and the tip of a fictitious rigid beam of the hub body (see Figure 3.5. In the sequel, it is assumed that strings and iniertials forces of masses (m) work on ly the normal direction y of the beam (i.e. Coriolis and centrifugal

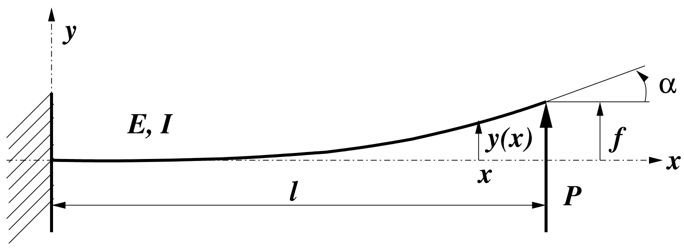


Figure 3.4: Static deformation of a cantilever beam with a tip load.

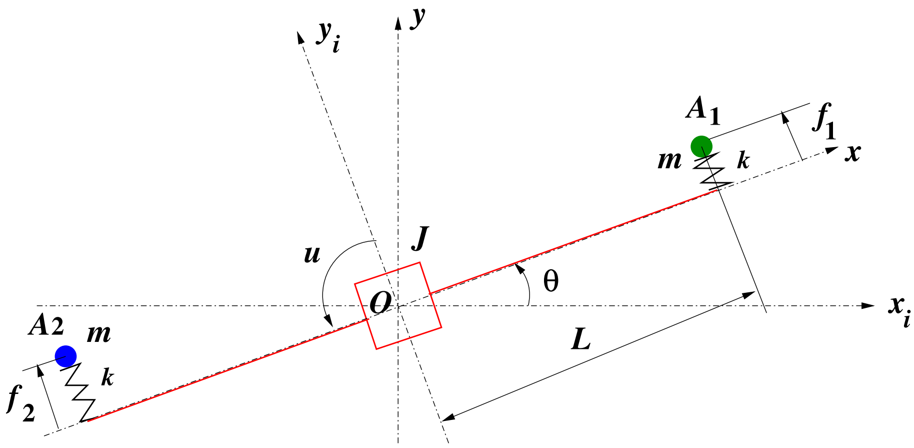


Figure 3.5: BAMOSS very simplified sketch.

accelerations are negleted. To derive the dynamic model, 2 options are proposed:

- Option 1: dynamic modeleling and analysis using LAGRANGE derivation. Questions A.2, A.3, A.4, A.5, ...
- Option 2: dynamic modeling using EULER, NEWTON approach. Questions A.2bis, A.3bis, A.5, ...

Option 1: dynamic modeling and analysis using Lagrange derivation

Let us denote : $q = [\theta \quad f_1 \quad f_2]^T$ the vector of generalized coordinates (that is the set of variables allowing the geometric configuration of the system to be entirely defined).

Question A.2: express the kinetic energy $\mathcal{T}$, the potential energy $\mathcal{V}$ and the dynamic model of this system under the generalized second-order form:

$$M\ddot{q} + Kq = Fu \ . \qquad (3.1)$$

Open-loop modal analysis

Question A.3:

- compute eigen-pulsation ω_i : $i = 1, \cdots, 3$ of the free system ($u = 0$),
- draw qualitatively the modal shape of each eigen-mode,
- study the system controllability directly from the second-order form.

Closed-loop modal analysis

Question A.4: compute the transmission zeros of the transfer function $\frac{\Theta}{U}(s)$ using a closed-loop modal analysis of system (3.1) with $u = -K_\theta \theta$ and $K_\theta \to \infty$. Do the same question with the transmission zeros of the transfer function $\frac{\beta}{U}(s)$.

State-space representation

An arbitrary damping matrix is introduced in the dynamic model:

$$D = \begin{bmatrix} 0.007 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} ,$$

and the dynamic model reads now:

$$M\ddot{q} + D\dot{q} + Kq = Fu \, . \tag{3.2}$$

Option 2: dynamic modeling using Euler, Newton approach

Question A.2bis: applying the Dynamic fundamental principle on the 3 bodies of Figure 3.5 (the green mass, the blue mass and the red main body), show that the dynamic behavior is described by the following set of 2-nd order differential equations:

$$J\ddot{\theta} = u + kLf_1 - kLf_2 - f_v\dot{\theta} \tag{3.3}$$

$$m(\ddot{f}_1 + L\ddot{\theta}) = -kf_1 \tag{3.4}$$

$$m(\ddot{f}_2 - L\ddot{\theta}) = -kf_2 \tag{3.5}$$

where $f_v = 0.007\, Ns/m$ is the viscous friction coefficient acting inside the hub pivot joint.

Question A.3bis: describe this set of equations by a functional block-diagram involving only integrators, sum and gains. Build the associated Simulink file.

Question A.5: create, using Matlab macro-function `ss` (option 1) or `linmod` (option 2), the model of the system between the input u and the 3 outputs θ, $\dot{\theta}$ and $\dot{\beta}$ and analyze:

- system eigen-values (macro-function `eig` or `damp`),
- system controllability (macro-function `ctrbf`),
- poles and zeros of the 3 transfers using function `zpk` (pole/zero cancelations and non-minimum phase response on the transfer between u and $\dot{\beta}$ should be highlighted and must match analytical results obtained in questions A.3 and A.4).

Question A.6: plot and comment the frequency-domain responses of the 3 transfers (macro-function `bode`).

Question A.7: plot output time-domain responses to initial conditions x_0 (macro-function `initial`) and comment observability property of each mode from each output.

3.3.2 Control design

Question B.1: using root loci and NICHOLS plots (macro-function `rltool`), analyze the control law :

$$u = K_p(\theta_{ref} - \theta) - K_d\dot{\theta}$$

(the loop must be " opened " on the input u). θ_{ref} is the hub position input reference and is null during regulation. Proportional and derivative gains K_p and K_d are tuned on the rigid-assumed model (which can be obtained when the YOUNG modulus E tends toward infinity) in order to have a second order closed-loop behavior with a pulsation $\omega = 2\,rd/s$ and a damping ratio $\xi = 0.707$.

Question B.2: simulate the closed-loop time-domain responses to initial conditions (macro functions `feedback`, `initial`) and the step response (macro function `step`). Comment these responses.

Question B.3: convert the continuous-time model to a discrete time-model taking into account a zero-order hold on the input u (macro-function `c2d`, sampling period $T_s = 0.1\,s$) and analyze, on root locus and NICHOLS plots, the previous proportional-derivative control law. Choose a sampling period to have at least a 6 dB gain margin.

Question B.4: let us consider one more time the continuous-time domain model and assume that only θ et $\dot{\beta}$ are measured: analyze, on root locus and NICHOLS plot, the control law:

$$u = K_p(\theta_{ref} - \theta) - K_d\dot{\beta}\ .$$

Design a filter $F(s)$ such that the new control law :

$$u = F(s)\Big(K_p(\theta_{ref} - \theta) - K_d\dot{\beta}\Big) \,.$$

is stabilizing.

3.4 Longitudinal flight control and auto-pilot

We consider the longitudinal flight model of a Concorde in the following conditions: altitude: 5000 feet, Mach=0.7 :

$$\begin{bmatrix} \dot{V} \\ \dot{\gamma} \\ \dot{\alpha} \\ \dot{q} \\ \dot{H} \end{bmatrix} = \begin{bmatrix} -1.202e-2 & -9.804 & -1.488e+1 & 0 & 0 \\ 3.579e-4 & 0 & 8.524e-1 & 0 & -4.163e-6 \\ -3.579e-4 & 0 & -8.524e-1 & 1 & 4.163e-6 \\ 0 & 0 & -2.665 & -2.783e-1 & 0 \\ 0 & 2.341e+2 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} V \\ \gamma \\ \alpha \\ q \\ H \end{bmatrix} + \begin{bmatrix} 4.958 & 0 \\ 0 & 3.113e-1 \\ 0 & -3.113e-1 \\ 0 & -4.936 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \delta_x \\ \delta_m \end{bmatrix} .$$

This linear model involves:

- 5 states (small variations around the equilibrium state):
  - V: aerodynamic speed (m/s). $V_{true} = V_{eq} + V$ with $V_{eq} = 234\,m/s$,
  - γ: climb angle (rd),
  - α: angle of attack (rd). $\alpha_{true} = \alpha_{eq} + \alpha$,
  - q: pitch rate (rd/s),
  - H: altitude (m). $H_{true} = H_{eq} + H$,
- 2 control inputs (small variations around trimmed conditions):
  - δ_x: thrust control (% of the maximal thrust),
  - δ_m: elevator deflection (rd, positive to pitch down).

Part 1: Measurement equation

All states are measured except α. Instead of α, the vertical acceleration is measured (this measure is more reliable than α for the design of the flight control loop). Moreover, pilot orders (joystick input) are in fact interpreted as load factor reference inputs (the load factor n_z is the vertical acceleration expressed in $g = 9.81\ m/s^2$) and is positive upwards. The distance l along x-axis from the A/C centre of gravity to the accelerometer is $l = 25\,m$ (frontwards).

- Under the small angle assumption ($\alpha_{eq} \approx 0$) and first order approximation (non-linear terms in q^2, $\dot{V}\gamma$, $\cdots$ are neglected), show that the load factor n_z can be expressed as a function of state derivatives $\dot{\gamma}$ and $\dot{q}$ and parameters l, $g = 9.91\,(m/s^2)$ and $V_{eq} = 234\,m/s$ (Mach 0.7); that is:

$$n_z = (V_{eq}\dot{\gamma} + l\,\dot{q})/g\ ,$$

- create on MATLAB (macro-function `ss`) the **full** model G_f (5 states: $[V,\ \gamma,\ \alpha,\ q,\ H]^T$, 2 control inputs: $[\delta_x,\ \delta_m]^T$, 5 outputs: $[V,\ \gamma,\ n_z,\ q,\ H]^T$)

Part 2: Short term model: flight control design

We seek here to improve A/C flying qualities in response to pilot orders by the mean of a flight control (see Figure 1). This flight control (i.e. this controller) will be designed on the **short term** model (short term corresponds approximatively to 10 seconds). This **short term** model (also called: model "α, q") is obtained assuming that slow state variables (V, γ, H) stay to zero during this short term.

- Analyse the step response of the full model over 10 seconds (macro-function `step`). Conclusions ?
- Create on MATLAB the **short term** model G_s (2 states: $[\alpha,\ q]^T$, 1 control input: δ_m, 2 outputs $[n_z,\ q]^T$),
- Compare the full model G_f and the **short term** model G_s from the point of view of i) their dynamics (macro-function `damp`), ii) their short term step responses (macro-function `step`) and iii) their frequency-domain responses (macro-function `bode`).
- On the **short term** model, analyse using a root locus (macro-function `rltool`) the influence on the A/C dynamics of a proportional control law: $\delta_m = -K_p n_z$.
- Analyse using root locus the influence on the A/C dynamics of a derivative control law: $\delta_m = -K_v q$.
- We consider now Figure 3.6 where A/C is the short term model G_s and the flight control law is :
$$\delta_m = K_p(n_{z_c} - n_z) - K_v q \ .$$
Tune K_p, K_v in order to meet the following **flight control specifications** [1]:
  - a 5% settling time on a step response lower or equal to $1.5\,s$,
  - a damping ratio of the angle of attack oscillation greater or equal to 0.65,
  - a 60 degrees phase margin (at least).
- Tune the integral gain K_i to cancel the previous steady state error:
$$\delta_m = (K_p + K_i \frac{1}{s})(n_{z_c} - n_z) - K_v q \ .$$
- We want now to take into account the elevator servo-control dynamics which is modeled as a second order transfer $F_a(s)$ with a natural frequency equal to $10\,rd/s$ and a damping ratio equal to 0.7. Are the previous performances still fulfilled?

[1] This tuning is a try and error procedure involving:

- analysis of the open-loop transfer function (the loop is opened on δ_m, that is `rltool([Kp Kv]*Gs)` to analyze closed-loop dynamics and phase margin,
- time-domain simulations using a `SIMULINK` file corresponding to Figure 3.6 (to be built) to analyse step response and settling time.

Highlight the trade-off between settling time and phase margin (try for instance a 1.s settling time).

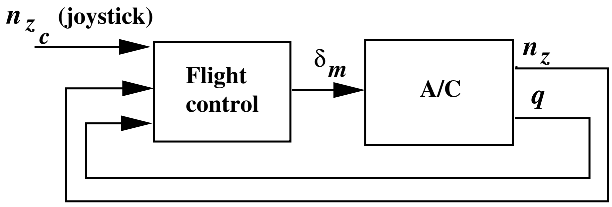


Figure 3.6: Longitudinal flight control.

Part 3: Auto-pilot design

We consider now the auto-pilot design. The auto-pilot must be able to track a trajectory (delivered by the upper guidance loop) according to the speed V_c or to climb angle γ_c (see Figure 2).

- Plug the previous control law ($\delta_m = (K_p + K_i \frac{1}{s})(n_{z_c} - n_z) - K_v q$) on the full model G_f using a `SIMULINK` file and taking into account elevator servo-control $F_a(s)$ on the input δ_m and the engine thrust servo-control $F_x(s) = \frac{1}{s+1}$ on the input δ_x. Check that the **flight control specifications** are still met.
- **Climb angle servo-loop:** design a controller $K_{pente}(s)$ $n_{z_c} = K_{pente}(s)(\gamma_c - \gamma)$ such that the step response be without steady state error and with a rise time equal to 10 seconds (step magnitude: $\gamma_c = 1\,deg$).
- **Speed servo-loop:** design a controller $K_{vit}(s)$ $\delta_x = K_{vit}(s)(V_c - V)$ such that the step response be without steady state error and with a settling time equal to 10 seconds (step magnitude: $V_c = 10\,m/s$).
- **Dynamic decoupling:** Check that the climb angle variation does not exceed 0.4 degree ($|\Delta\gamma| < 0.4\,°$ that is $|\Delta V_z| < 1.6\,m/s$ on the vertical speed) for a longitudinal speed step V_c equal to $10\,m/s$. On the other hand, tune the decoupling gain K_{dec} between the climb angle and the thrust control (see Figure (2)) such that the longitudinal speed variation does not exceed $0.2\,m/s$ ($|\Delta V| < 0.2\,m/s$) for a climb angle step γ_c equal to $1\,°$ (that is a vertical speed V_z equal to $4\,m/s$).
- **Guidance loop:** Design a guidance law $\gamma_c = f(H_c)$ allowing the flight condition to be taken into account (Speed V_c, Altitude H_c) specified by the flight planning and simulate the **long term** step response (1000 seconds) to an altitude step equal to $1000\,m$ [2].

[2]Keep in mind that the model varies according to the flying conditions (altitude, speed). In practice, the guidance loop must be validated on the non-linear model (great movings). Thus, various flight control and auto-pilot laws must be designed in various operating points distributed along the whole flight envelope. These laws are then gain-scheduled according to V_{eq} and H_{eq}.

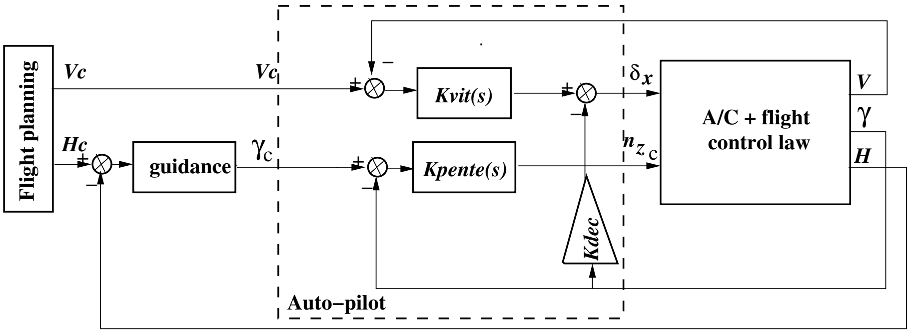


Figure 3.7: Guidance and auto-pilot loops.

EMPTY PAGE

Modelling, analysis and control of aerospace vehicles
Matlab lab-work

D. Alazard

1 Modeling and control of a pendulum

1.1 Modeling

Let us consider the pendulum depicted in Figure 1. The dynamic behaviour of this pendulum can be described by following input-ouput differential equation:

$$J\ddot{\theta}(t) = u(t) - f\dot{\theta}(t) - mlg \sin\theta(t)$$

where $u(t)$ (Nm) is the torque applied by the actuator (electrical motor) and f (Nms) is the coefficient of the viscous friction inside the pivot joint.

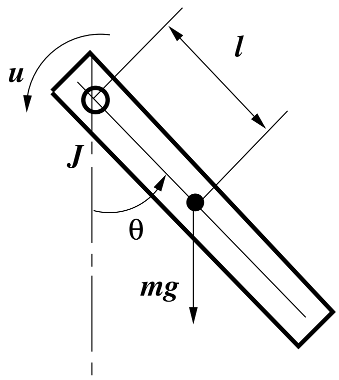


Figure 1: The pendulum.

- Describe directly this differential equation by a SIMULINK block diagram (N.A.: $m = 1\,Kg$, $l = 0.2\,m$, $J = ml^2$ (case of a concentrated mass at the pendulum tip), $g = 9.81\,m/s^2$ and $f = 0.004\,Nms$).
- Determine (hand-made computation):

- equilibrium conditions (i.e. control signal u_e) imposing $\theta = \theta_e$ (θ_e is a given constant value),
- the linear model (transfer function: $F(s) = \frac{\Delta\theta}{\Delta u}(s)$) around this equilibrium condition,
- the poles of this linear model,
- the BODE response,

for different values of θ_e (0, $\pi/2$, π).

- do the same thing using macro-functions `trim`, `linmod`, `ss`, `tf`, `damp` and `bode` on the non-linear `SIMULINK` model. The syntax to compute the linear model $F(s)$ is:

```
[xe,ue]=trim('SIMULINK_filename',[],[],theta_e,[],[],1)}
[A,B,C,D]=trim('SIMULINK_filename',xe,ue);
F=ss(A,B,C,D);
F=tf(F)
```

- Simulate the step response for different step magnitudes ($0.15\,Nm$, $1.5\,Nm$) using the linear model $F(s)$ (macro-fonction `step`) and using the SIMULINK file describing the non-linear model and compare both responses.

1.2 Control

Let us consider the linear model $F(s)$ around $\theta_e = \pi$.

1.2.1 Stabilization and disturbance rejection

- using MATLAB function `sisotool`, design a first order controller $K_1(s) = \frac{K_p + K_v s}{1+\tau s}$ to meet the following specifications:

  S1: the closed-loop dominant dynamics is governed by a second order dynamics $s^2 + 2\xi\omega s + \omega^2$ with $\omega > 12\,rd/s$,

  S2: the damping ratio of closed-loop poles must be greater than 0.6,

  S3: $|\text{gain margin}| > 6\,dB$,

  S4: phase margin $> 30\,deg$,

S5: $|K(j\omega)| < 40$ (i.e.: $32\,dB$) $\quad \forall\,\omega > 10\,rd/s$.

- plug the controller $K(s)$ in the `SIMULINK` file involving the non-linear model of the pendulum and check that the pendulum is stabilized around $\theta_e = \pi$ and that this controller works even if external disturbances (initial condition on $\dot{\theta}_0$) creates large angular motions (it is also recommended to take into account actuator saturation: $\pm 3\,Nm$).

1.2.2 Reference input tracking

- plot the step response of the previous controller on the linear model (macro-functions `feedback` and `step`) and tune a new second order controller $C_i(s)$ including one integrator to cancel the static error and meeting specs [S2] to [S5],

- comment the transcient response on the step response obtained with this new controller. Then, identify the gains K_p, K_v, K_i and the time-constant τ of the controller $C_i(s)$ expressed as a PID controller:

$$C_i(s) = K_p + K_v \frac{s}{1+\tau s} + K_i \frac{1}{s} \ ,$$

  and show that the transient response can be improved implementing the control law with the integral term in the direct path and the proportional and derivative terms in the feedback path according to Figure 2.

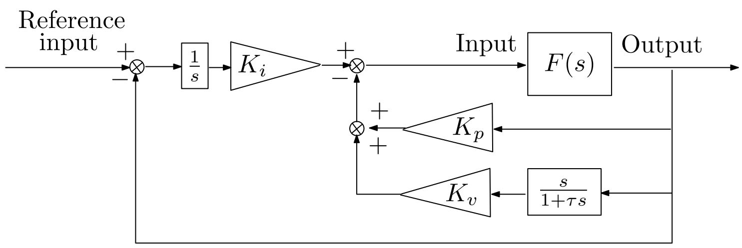


Figure 2: A possible implementation of the PID control.

- check the behavior of the new controller on the non-linear `SIMULINK` model.

1.2.3 Discrete time control

The previous control is implemented on a digital computer. The sampling rate is $50\,Hz$. A zero order hold is fitted on the input u.

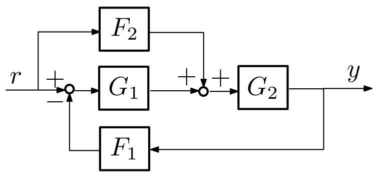
- Discretize the linear model (taking into account the zero order hold) and the controller using TUSTIN transformation (macro-function `c2d`) and analyze the servo-loop properties of the discrete-time servo-loop system (on both the linear model $F(s)$ using function `sisotool` and the non-linear `SIMULINK` model).
- Considering the PID controller $C_i(s)$ implemented in Figure 2, propose a simple way to implement it in discrete-time.

SIAE - Tiangin
Semester 3 (2013/11)
EL31: Servo-loop System
Duration: 2 h 30, No document allowed

D. Alazard

1 Block diagram to transfer function

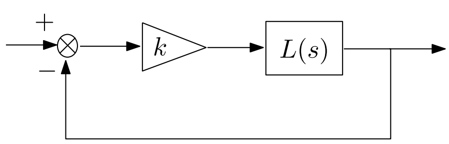
Let us consider the following system:



Compute the transfer function $\frac{Y(s)}{R(s)}$

2 Root-locus and stability margins

Let-us consider the servo-loop system described in the following Figure where $L(s) = \frac{s+1}{s^2-s}$ and k is a gain to be tuned.



a) draw qualitatively the root locus of $L(s)$,

b) compute the closed loop transfer function as a function of k,

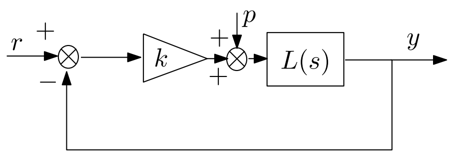
c) what condition must satisfy k to make the closed-loop system stable?,

d) what is the value of k such that the closed-loop system is a second order stable system with a damping ratio of $\sqrt{2}/2$? What is the corresponding pulsation?,

e) we consider that $k = 2$. Draw qualtitavely the BODE plot of $2L(s)$. What are the gain margin and the phase margin of such a tuning?

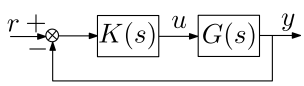
f) for $k = 2$, draw qualitatively the step response of the closed-loop system and comment it,

g) we consider a disturbance p acting on the input of $L(s)$ (see following Figure). Compute (for $k = 2$) the steady state error for a null reference input r and a step on the disturbance p. Propose a new controller to cancel this steady state error.



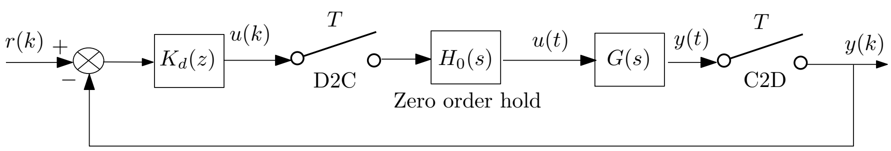
3 Discrete-time control

We consider the servo loop system described in the follwing Figure where $G(s) = \frac{1}{s^2-s}$ and $K(s) = 12 + \frac{4}{s} + 8s$ (P.I.D. controller).



a) Compute the closed-loop transfert $Y/R(s)$ and the closed-loop poles (one pole is obvious).

b) This controller must be implemented on a digital computer: propose a sampling period T and a discrete-time controller $K_d(z)$ to implement the servo-loop according to the following Figure.

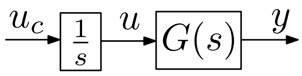


c) Compute the discrete-time transfer fonction $G_d(z)$ between $u(k)$ and $y(k)$ as a fonction of the sampling period T and using table given in appendix.

d) Give the time domain recurent equations required to implement $K_d(z)$ on the digital computer.

4 State feedback

We consider the plant $G(s) = \frac{Y}{U}(s) = \frac{1}{s^2-s}$ and the system depicted in the following block diagram:



a) give the state space representation of the system between u_c and y associated with the state vector $x = [y,\ \dot{y},\ u]^T$,

b) compute a state feedback $u = e - Kx$ such that the dominant dynamic of the closed-loop system corresponds to a second order system with a pulsation $\omega = 2\,rd/s$ and a damping ratio $\xi = 1/\sqrt{2}$.

A Apendix: Z-transform table

continuous-time signal		discrete-time signal	
$h(t)$	$H(s)$	$h(kT) = h(k)$	$H(z)$
$\delta(t)$ (DIRAC "function")	1		
		$\delta(k)$	1
$\mathbf{e}(t)$ (step function*)	$\frac{1}{s}$	$\mathbf{e}(k)$	$\dfrac{z}{z-1}$
$t\mathbf{e}(t)$ (ramp function)	$\frac{1}{s^2}$	$kT\mathbf{e}(k)$	$\dfrac{Tz}{(z-1)^2}$
$t^2\mathbf{e}(t)$	$\frac{2}{s^3}$	$k^2T^2\mathbf{e}(k)$	$\dfrac{T^2z(z+1)}{(z-1)^3}$
$e^{-at}\mathbf{e}(t)$	$\dfrac{1}{s+a}$	$e^{-knT}\mathbf{e}(k)$	$\dfrac{z}{z-e^{-aT}}$
$t\,e^{-at}\mathbf{e}(t)$	$\dfrac{1}{(s+a)^2}$	$kT\,e^{-akT}\mathbf{e}(k)$	$\dfrac{Tze^{-aT}}{\left(z-e^{-aT}\right)^2}$