PHYSICAL HUMAN-ROBOT INTERACTION

PID controllers and stability margins

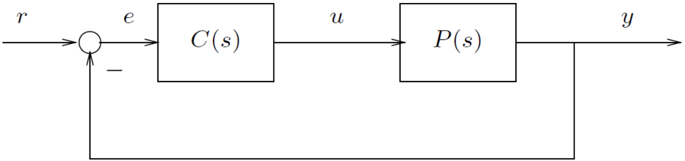
Riccardo Muradore

UNIVERSITÀ di VERONA
Dipartimento di INFORMATICA

Outline

- Continuous-time PID
- Stability margins
- Anti wind-up
- Nested Loops and Feedforward
- Discrete-time PID

Feedback Loop



- $C(s)$ controller
- $P(s)$ plant
- $r(t)$ reference signal
- $y(t)$ output (measurements)
- $e(t)$ error. $e(t) = r(t) - y(t)$
- $u(t)$ input (control commands)

Continuous-time PID

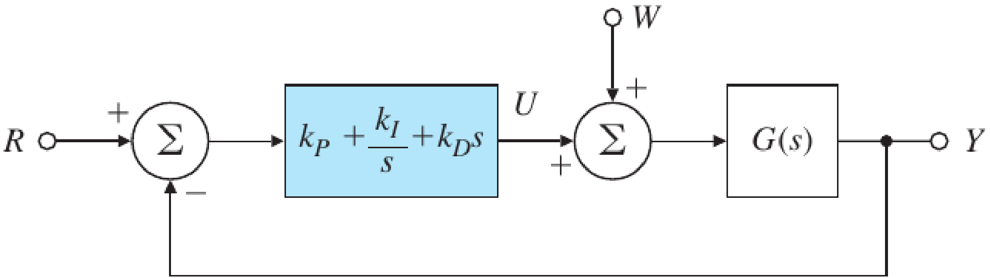
A PID (Proportional-Integrative-Derivative) is described by the integro-differential equation

$$u(t) = \underbrace{k_P e(t)}_{u_P(t)} + \underbrace{k_I \int_{t_0}^{t} e(\tau) d\tau}_{u_I(t)} + \underbrace{k_D \frac{de(t)}{dt}}_{u_D(t)} \tag{1}$$

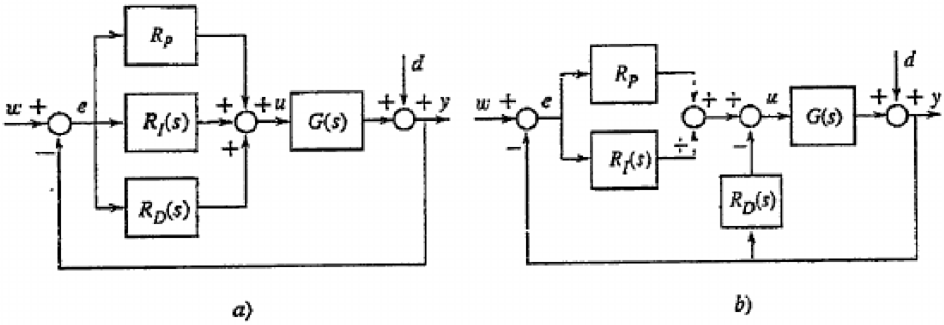
where

- $k_P \in \mathbb{R}^+$ is the proportional gain
- $k_I \in \mathbb{R}^+$ is the integrative gain
- $k_D \in \mathbb{R}^+$ is the derivative gain

Continuous-time PID



Continuous-time PID



Continuous-time PID

Applying the Laplace transform (with null i.c.) we get the transfer function

$$C(s) = k_P + \frac{k_I}{s} + k_D s \tag{2}$$

or equivalently

$$C(s) = k_P \left[1 + \frac{1}{T_I s} + T_D s \right] \tag{3}$$

where

- $T_I := \frac{k_P}{k_I}$ is the *reset rate* of the integral action ($\sim$ constant time of the integral action)
- $T_D := \frac{k_D}{k_P}$ is the *derivative rate* of the derivative action ($\sim$ constant time of the derivative action)

The transfer function $C(s)$, both in the form (2) and (3), is not causal for any $k_D \neq 0$ (i.e. $T_D \neq 0$).
The degree of the polynomial at numerator is larger than the degree of the polynomial at denominator.

Continuous-time PID

To overcome this realizability problem, an *high-frequency pole* is added in order to have

$$C(s) = \frac{N(s)}{D(s)}, \qquad \deg\{N(s)\} = \deg\{D(s)\}$$

Since the the pole is located in high frequency the effect on the behavior of the controller is negligible: the closed-loop system response is almost the same in the frequency range of interest.

The transfer function of the controller $C(s)$ becomes

$$C(s) = k_P + \frac{k_I}{s} + \frac{k_D s}{1 + \frac{K_D}{K_P N} s} = k_P \left[1 + \frac{1}{T_I s} + \frac{T_D s}{1 + \frac{T_D}{N} s} \right] \tag{4}$$

$$C(s) = k_P + \frac{k_I}{s} + \frac{k_D s}{1 + k'_D s} = k_P \left[1 + \frac{1}{T_I s} + \frac{T_D s}{1 + T'_D s} \right] \tag{5}$$

where the additional pole is located in $-\frac{1}{T'_D}$ with $T'_D \ll T_D$.

Stability margins

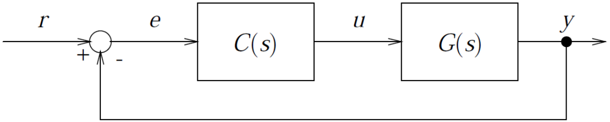
The choice of the parameters k_P, k_I, k_D (*tuning of the parameters*) is driven by different requirements/specs

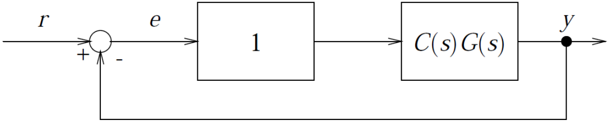
1. *asymptotic stability of the closed loop system*,
2. *stability margins*,
3. type of system: which canonical signal $r(t) = \delta_{-i}(t)$ the closed loop system can track with zero asymptotic error: $e(t) \to 0$, for $t \to \infty$,
4. static precision,
5. dynamic precision (fast tracking),
6. disturbance rejection.

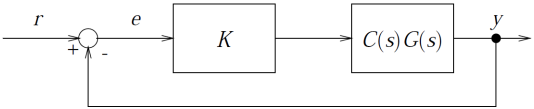
Since we can play with just 3 parameters, it is usually impossible to satisfy all the specifications (*trade-off*)

Stability margins

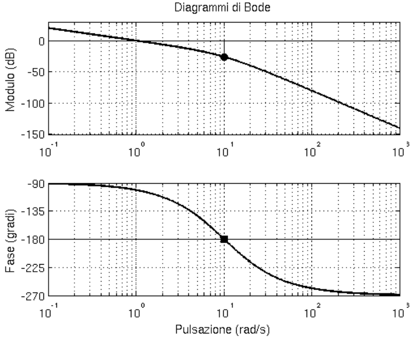
Equivalent block diagram with $L(s) = C(s)G(s)$ the open loop transfer function







Example



By increasing K, the transfer function $K\,L(s) = K\,C(s)G(s)$ assumes the critical value -1, i.e.

$$|K\,L(\jmath\omega)| = 1, \qquad \angle K\,L(\jmath\omega) = -\pi$$

for a certain frequency (e.g. $\omega = 10$rad/s in the example)

Why is it a problem? [Hint. *neutral stability*]

Definition

The gain margin (GM) m_g is the factor by which the gain K can be increased (or decreased in certain cases) before instability results.
It can be read directly from the Bode plot by measuring the vertical distance between the $|K\,L(\jmath\omega)|$ curve and the unitary magnitude line at the frequency where $\angle K\,L(\jmath\omega) = -\pi$.

Let ω_π be the frequency such that

$$\angle L(\jmath\omega_\pi) = -\pi$$

then

$$m_g \triangleq \frac{1}{|G(j\omega_\pi)|} \tag{6}$$

or

$$m_g^{dB} \triangleq -20\log_{10}|G(j\omega_\pi)|, \tag{7}$$

if GM is expressed in decibel.

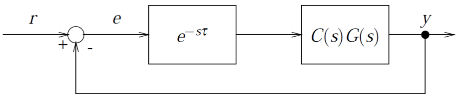
Gain margin

Observations:

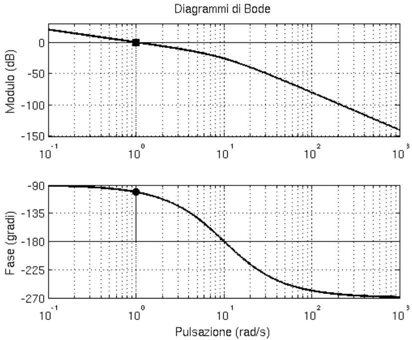
- the GM is the reciprocal of the gain $|G(j\omega_\pi)|$ at the frequency at which the phase angle reaches $-\pi$.
- The gain margin is the increase in the system gain when phase is equal to $-\pi$ that will result in a *marginally stable system* with intersection of the $-1 + \jmath 0$ point on the Nyquist diagram (some poles are on the imaginary axis $\jmath\mathbb{R}$ for $K = m_g$).
- The gain margin is the multiplicative factor that brings the closed loop system to instability:
  - for $K < m_g$ the closed-loop system is stable
  - for $K = m_g$ the closed-loop system has a pole on the origin or a pair of complex conjugate poles on the imaginary axis
  - for $K > m_g$ the closed-loop system is unstable

Remark. The conclusion *the closed-loop system is stable if and only if* $m_g^{dB} > 0$ holds true only for systems with open-loop transfer function $L(s)$ monotonically decreasing.

Phase margin



Example



By increasing the delay τ, the transfer function $e^{-s\tau}\, L(s) = K\, C(s)G(s)$ assumes the critical value $-1 + \jmath 0$, i.e.

$$|e^{-s\tau}\, L(\jmath\omega)| = 1, \qquad \angle e^{-s\tau}\, L(\jmath\omega) = -\pi$$

for a certain frequency (e.g. $\omega = 1$rad/s in the example)

Why is it a problem? [Hint. *neutral stability*]

Exercise

What is the relationship between τ and the phase in the Bode diagram? [Hint. $-\omega\tau$ *rad*]

Definition

The phase margin (PM) m_φ is the amount by which the phase of $L(\jmath\omega) = C(\jmath\omega)G(\jmath\omega)$ exceeds $-\pi$ when $|L(\jmath\omega)| = 1$.

Let ω_c be the crossover frequency (frequency at which the magnitude is unity, or 0 dB).

$$|L(\jmath\omega_c)| = 1$$

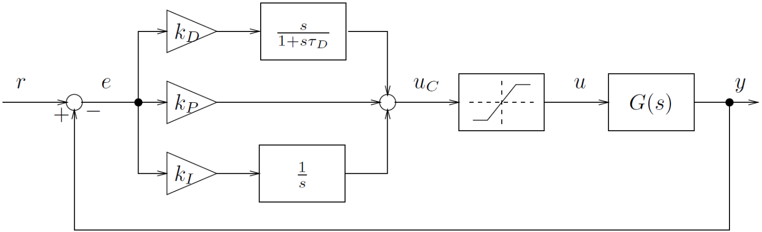
The phase margin is the angle we have to subtract to the phase at ω_c to obtain $-\pi$

$$m_\varphi \triangleq \pi + \angle L(j\omega_c). \tag{8}$$

Observations:

- The phase margin is the amount of phase shift of the $L(\jmath\omega)$ at unity magnitude that will result in a marginally stable system with intersection of the $-1 + \jmath 0$ point on the Nyquist diagram.
- The phase margin tells us the maximum angle we can subtract to the phase to bring the closed-loop system to instability.
- The conclusion *the closed-loop system is stable if and only if* $m_\varphi > 0$ holds true only for systems with open-loop transfer function $L(s)$ monotonically decreasing.

It is quite common that the command input has upper and lower bounds:



i.e.

$$u(t) = \begin{cases} u_C(t), & \text{if } |u_C(t)| < u_{MAX} \\ u_{MAX}, & \text{if } u_C(t) > u_{MAX} \\ -u_{MAX}, & \text{if } u_C(t) < -u_{MAX} \end{cases}$$

During saturation, the command to the plant stops changing and the feedback path is effectively opened. If the error signal continues to be applied to the integrator under these conditions, the integrator output will grow (*windup*) until the sign of the error changes and the integration turns around.

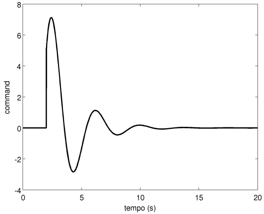
Command saturation

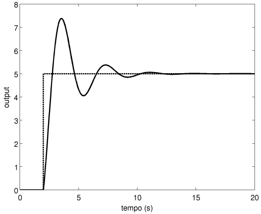
Due to the integral action, a saturation has an important effect on the response of the closed-loop system.
Let's start with a unit step as reference input

$$r(t) = \delta_{-1}(t-2)$$

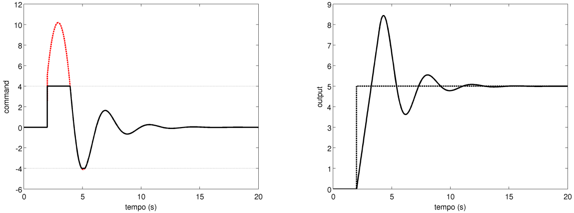
and no saturation.

The time evolution of the command $u(t)$ (plot of the left) and of the output $y(t)$ (plot on the right) are given below

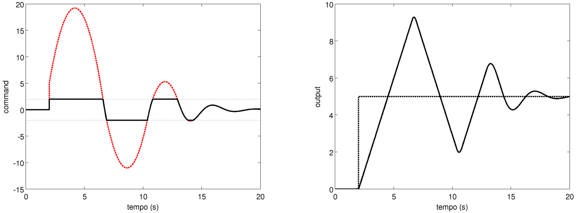




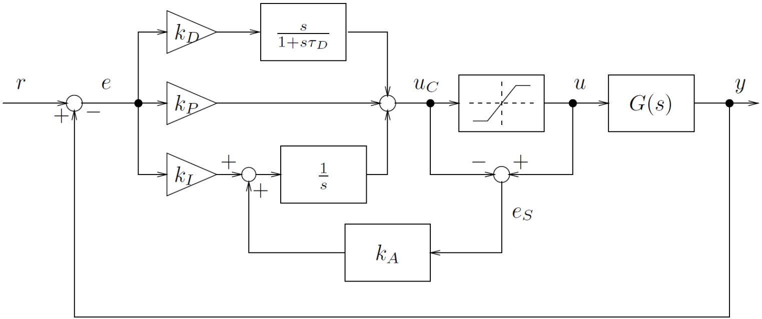
Time evolution with saturation $u_{MAX} = 4$:



Time evolution with saturation $u_{MAX} = 2$:



To mitigate such performance degradation, a simple loop can be added to the original feedback loop. This loop is called *anti wind-up*.

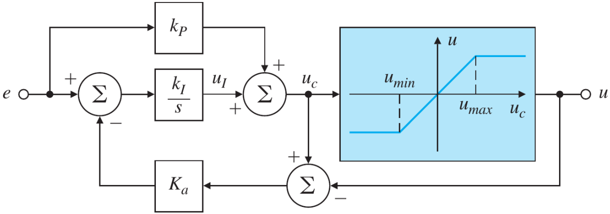


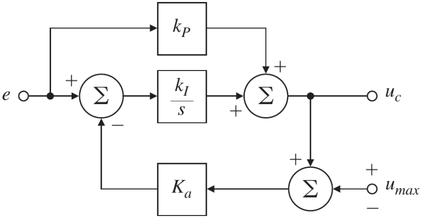
Anytime the command is saturated, the gain k_A *turns off* the integral action and so avoids large overshoot and poor transient response.

This circuit is called *integrator anti-windup*.

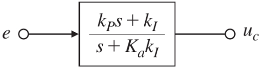
Anti wind-up

Transfer function of a P + Integrator anti-windup when the command is saturated



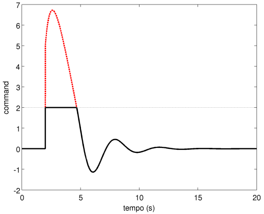


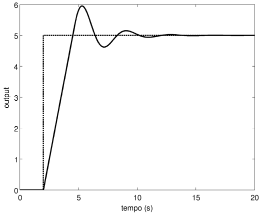
(c)



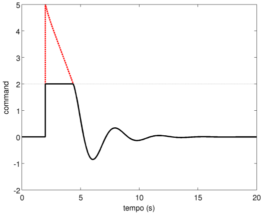
The purpose of the antiwindup is to provide local feedback to make the controller stable alone when the main loop is opened by signal saturation

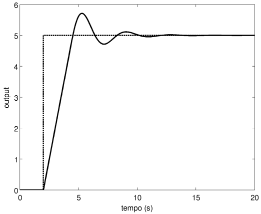
With anti wind-up circuit ($k_A = 2$):





With anti wind-up circuit ($k_A = 5$):



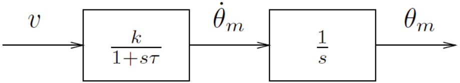


DC motor transfer function

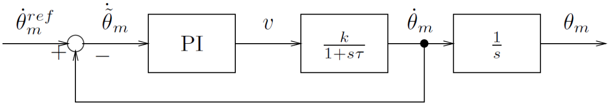
$$P(s) = \frac{\hat{\theta}_m(s)}{\hat{V}(s)} = \frac{K_t}{s\left[(Js+b)(Ls+R)+K_mK_e\right]}$$

Simplified DC motor transfer function ($L \simeq 0$)

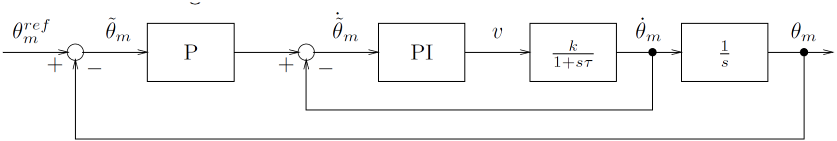
$$P(s) \simeq \frac{K_t/R}{s\left[Js+\left(b+\frac{K_mK_e}{R}\right)\right]} = \frac{k}{s\,(1+\tau s)}$$



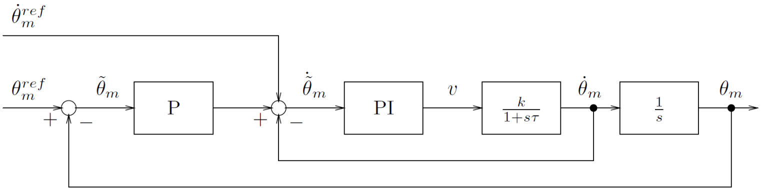
Speed feedback loop (PI)

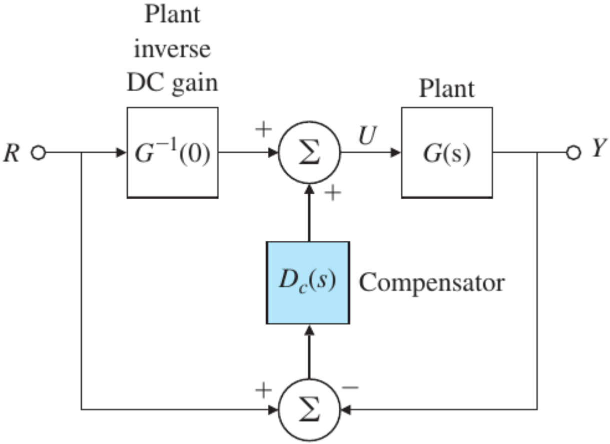


Nested Speed feedback loop (PI) and position feedback loop (P)



If also the reference velocity $\dot{\theta}$ is available, we can implement a *feedforward circuit* to improve response time




Plant
inverse
DC gain
$G^{-1}(0)$
Plant
$G(s)$
R
U
Y
$D_c(s)$
Compensator
+
+
+
−

Remark 1

A PI speed control loop is equivalent to a PD control loop.
(what about the initial conditions ???)

Remark 2

Any time an integral action is used, it is a good practice to implement the anti-windup circuit.

Remark 3

The nested loop position/velocity (P/PI) is equivalent to a PID position control loop.

State space representation

Sometimes it is convenient to exploit the state space representation instead of the transfer function representation.

The second-order system

$$\begin{aligned} m\ddot{x} + b\dot{x} + kx &= u \\ y &= x \end{aligned}$$

can be rewritten as

$$\begin{aligned} \ddot{x} + \frac{b}{m}\dot{x} + \frac{k}{m}x &= \frac{1}{m}u \\ y &= x \end{aligned}$$

State vector $X \in \mathbb{R}^2$

$$X := \begin{bmatrix} x \\ \dot{x} \end{bmatrix}$$

State space representation

$$\begin{aligned}\dot{X}(t) &= AX(t) + Bu \\ y(t) &= CX(t) + Du\end{aligned}$$

with

$$
\begin{aligned}
\dot{X}(t) &= AX(t) + Bu \\
y(t) &= CX(t) + Du
\end{aligned}
$$

with

$$
\begin{aligned}
A &= \begin{bmatrix} 0 & 1 \\ -\frac{k}{m} & -\frac{b}{m} \end{bmatrix} \\
B &= \begin{bmatrix} 0 \\ \frac{1}{m} \end{bmatrix} \\
C &= \begin{bmatrix} 1 & 0 \end{bmatrix} \\
D &= 0
\end{aligned}
$$

What happens if the model is the following one?

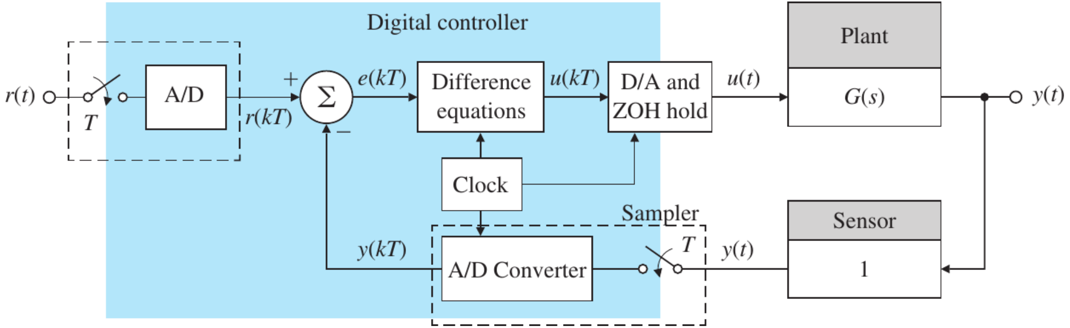
$$
\begin{aligned}
m\ddot{x} + b\dot{x} + k(x - x_e) &= u \\
y &= x
\end{aligned}
$$

Matlab/Simulink S-function

Define a continuous system in state-space format.
csfunc.m

Define a discrete system in state-space format.
dsfunc.m

To implement a controller, we need its discrete-time formulation



Discrete-time PID – Single actions

Using Euler approximation of the derivative ($\dot{x}(t) \simeq \frac{x(t)-x(t-T)}{T}$), we have

Proportional action

$$u(t) = k_P e(t) \qquad \rightarrow \qquad u(t_k) = k_P e(t_k)$$
$$\rightarrow \qquad u(k) = k_P e(k)$$

Integral action

$$u(t) = k_I \int_{t_0}^{t} e(\tau) d\tau \qquad \rightarrow \qquad u(k) = u(k-1) + k_I T e(k)$$

Derivative action

$$u(t) = k_D \frac{de(t)}{dt} \qquad \rightarrow \qquad u(k) = \frac{k_D}{T}[e(k) - e(k-1)]$$

A general continuous-time PID controller

$$u(t) = k_P e(t) + k_I \int_{t_0}^{t} e(\tau) d\tau + k_D \frac{de(t)}{dt}$$

can be re-written by derivation as

$$\dot{u}(t) = k_P \dot{e}(t) + k_I e(t) + k_D \ddot{e}(t)$$

and, finally, using twice the Euler approximation we get

$$u(k) \quad = \quad u(k-1) + \left(k_P + k_I T + \frac{k_D}{T}\right) e(k) - \left(k_P + 2\frac{k_D}{T}\right) e(k-1) + \frac{k_D}{T} e(k-2)$$

Exercise

Compute the discrete-time difference equations for PI and PD controllers.