Feedback Control Systems

Control Systems

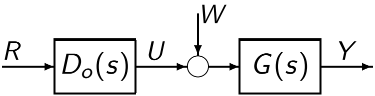
Fall 2023

Outline

- Open-loop and closed-loop control systems
- Stability of closed-loop systems
- Sensitivity of control systems to parameter variations
- Tracking and Regulation
- Steady-state error for tracking and regulation
- PID controllers
  - Proportional, Integral and Derivative (PID) Controller
  - PID controller tuning by Ziegler-Nichols method
  - PID controller tuning by model reference control method
  - PID controller implementation
  - Anti-windup
- Feedforward Control
- Cascade Control

Open-loop versus Closed-loop Control

Open-loop control :



$$Y(s) = G(s)D_o(s)R(s) + G(s)W(s)$$

Tracking Error Equation :

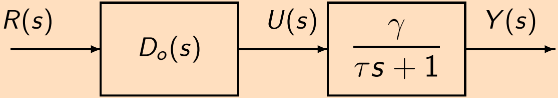
$$E = R - Y = R - [GD_oR + GW] = [1 - GD_o]R - GW$$

Controller design : Take $D_o \approx G^{-1}$. Note that this approximation should lead to a stable and implementable D_o.

- Simple, no sensor is required.
- Controller design needs full knowledge of the model.
- Sensitive to parameter variation or parameter uncertainty (variation with temperature, etc.),
- Sensitive to external disturbances W (load torque, parasite voltage).
- Cannot be applied to *unstable* systems.

Open-loop Control

Question



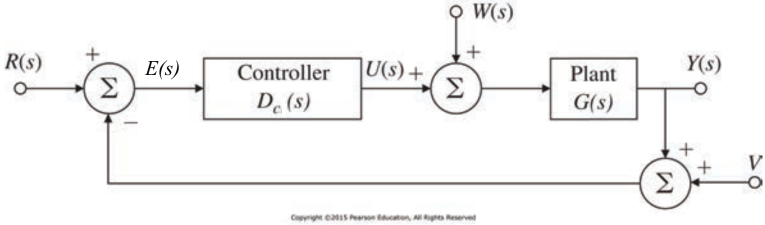
Let's design the controller $D_o(s)$ as the inverse of the plant model :

$$D_o(s) = \frac{\tau s + 1}{\gamma}$$

Then, $Y(s)$ will be equal to $R(s)$ and there will be no tracking error.
Is this controller realizable ?

Open-loop versus Closed-loop Control

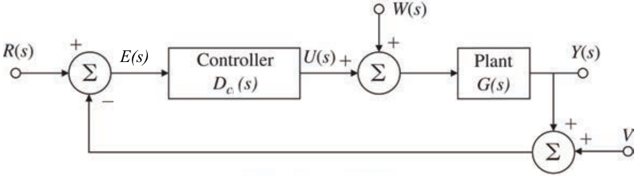
Closed-loop Block Diagram



Analysis of the closed-loop control system

- Is the closed-loop system stable ?
- Is it sensitive to model parameter variation and uncertainties ?
- Does it give good performance (small error) in **tracking** and disturbance rejection (**regulation**) ?

Open-loop versus Closed-loop Control



Closed-loop Transfer Functions :

$$Y = \frac{GD_c}{1+GD_c}R + \frac{G}{1+GD_c}W - \frac{GD_c}{1+GD_c}V$$
$$U = \frac{D_c}{1+GD_c}R - \frac{GD_c}{1+GD_c}W - \frac{D_c}{1+GD_c}V$$
$$E = \frac{1}{1+GD_c}R - \frac{G}{1+GD_c}W - \frac{1}{1+GD_c}V$$

Gang of Four : There are four closed-loop transfer functions :

$$\mathcal{S} = \frac{1}{1+GD_c} \quad ; \quad \mathcal{T} = \frac{GD_c}{1+GD_c} \quad ; \quad \mathcal{U} = \frac{D_c}{1+GD_c} \quad ; \quad \mathcal{V} = \frac{G}{1+GD_c}$$

Closed-Loop Stability

Closed-loop poles

- The closed-loop poles are the roots of the characteristic polynomial defined by : $p(s) = a(s)d(s) + b(s)c(s)$, where :

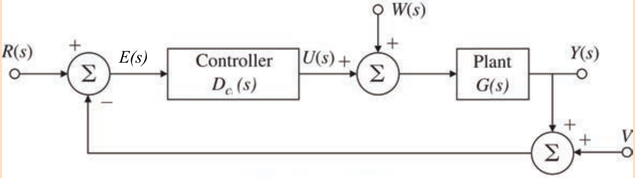
$$G(s) = \frac{b(s)}{a(s)} \quad \text{and} \quad D_c(s) = \frac{c(s)}{d(s)}$$

- The zeros of the transfer function $1 + GD_c$ are the closed loop poles if no zero and pole are cancelled when forming GD_c.
- The closed-loop poles are the union of the poles of all closed-loop transfer functions (the union of poles of the gang of four).

Internal Stability

- A closed-loop system is internally stable if and only if all the closed-loop poles are located in the LHP (the gang of four are stable).

Internal Stability



Suppose that $$G(s) = \frac{2s-1}{s^2+2}$$

- Is the closed loop system internally stable with $D_c(s) = 1$?
  (A) No (B) Yes
- Is the closed-loop system internally stable with $D_c(s) = \dfrac{4s+2}{2s-1}$?
  (A) No (B) Yes

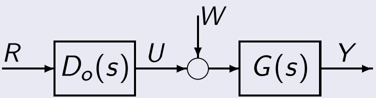
Sensitivity

Sensitivity

The sensitivity of a function $f : \mathbb{R} \to \mathbb{R}$ to the parameter x is the relative variation of f to the relative variation of x :

$$\mathcal{S}_x^f = \lim_{\Delta x \to 0} \frac{\frac{f(x+\Delta x)-f(x)}{f(x)}}{\frac{\Delta x}{x}} = \frac{\frac{df(x)}{dx}}{\frac{f(x)}{x}}$$

Open-Loop Sensitivity

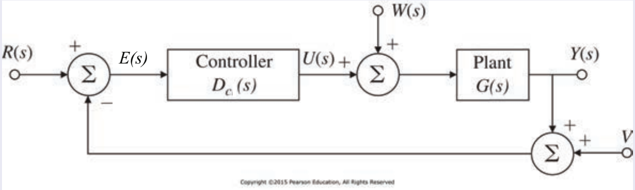


The sensitivity of GD_o to the plant model G : $\quad \mathcal{S}_G = \frac{\frac{dGD_o}{dG}}{\frac{GD_o}{G}} = 1.$

It means that 10% variation in G leads to 10% variation in the transfer function between R and Y.

Sensitivity

Closed-Loop Sensitivity



The sensitivity of $\mathcal{T} = GD_c(1 + GD_c)^{-1}$ to the plant model G :

$$\mathcal{S}_G^{\mathcal{T}} = \frac{\frac{d\mathcal{T}}{dG}}{\frac{\mathcal{T}}{G}} = \frac{G}{GD_c/(1 + GD_c)} \frac{(1 + GD_c)D_c - D_c(GD_c)}{(1 + GD_c)^2} = \frac{1}{1 + GD_c}$$

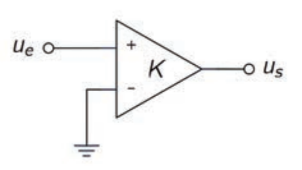
Remark : $\mathcal{S} = (1 + GD_c)^{-1}$ is called the *sensitivity function* and $\mathcal{T} = GD_c(1 + GD_c)^{-1}$ the *complementary sensitivity function*.

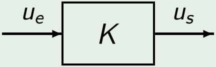
Closed-loop system is less sensitive to modelling error and variation than the open-loop system by a factor of $\mathcal{S}$.

Sensitivity

Example (Operational Amplifier)

Open-loop operation :





$$u_s = Ku_e$$

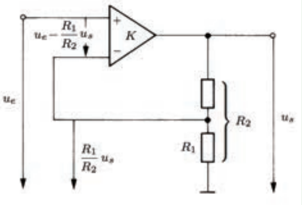
The sensitivity of the open-loop Op-Amp gain with respect to the parameter K :

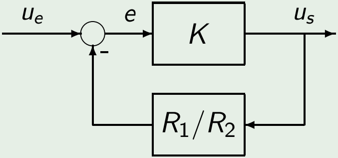
$$\mathcal{S}_K^K = 1$$

Open-loop systems are highly sensitive to modelling error and variation.

Example (Operational Amplifier)

Closed-loop operation :





$$u_s = K(u_e - \frac{R_1}{R_2} u_s) \Rightarrow u_s = \frac{K}{1 + K\frac{R_1}{R_2}} u_e$$

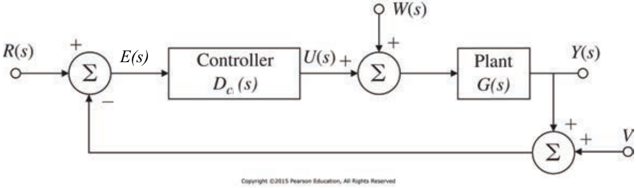
The sensitivity of the closed-loop Op-Amp gain with respect to the parameter K :

$$\text{Closed-loop gain } \mathcal{T} = \frac{K}{1 + K\frac{R_1}{R_2}} \Rightarrow \mathcal{S}_K^{\mathcal{T}} = \frac{\frac{d\mathcal{T}}{dK}}{\frac{\mathcal{T}}{K}} = \frac{\frac{(1+KR_1/R_2 - KR_1/R_2)}{(1+KR_1/R_2)^2}}{\frac{K}{1+KR_1/R_2}\frac{1}{K}} = \frac{1}{1 + K\frac{R_1}{R_2}}$$

Take $K = 1000$ and $R_1/R_2 = 1$ then the sensitivity is 0.001.

Closed-loop systems are less sensitive to modelling error and variation.

Tracking and Regulation



Tracking Performance

In some control systems the output should track a reference signal (desired output) in the absence of disturbance. The transfer function of interest is the one between R and the tracking error E : $\quad \mathcal{S} = \dfrac{E}{R} = \dfrac{1}{1 + GD_c}$

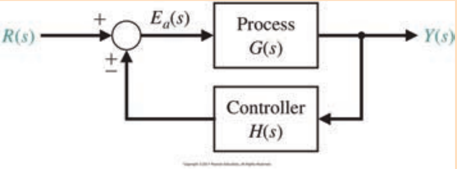
Regulation Performance

In some control systems the output should be constant (regulated on a desired value) in the presence of disturbance. The transfer function of interest is the one between W and E : $\quad \dfrac{E}{W} = -\dfrac{G}{1 + GD_c} = -\mathcal{V}$

Tracking

Question

Sometimes instead of a unity feedback system, the controller is placed in the feedback loop :



Find the transfer function between the reference signal and the tracking error.

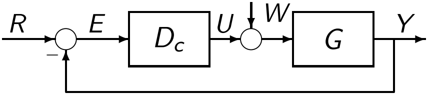
A : $\dfrac{1}{1 + H(s)G(s)}$ B : $\dfrac{G(s)}{1 + H(s)G(s)}$

C : $\dfrac{H(s)}{1 + H(s)G(s)}$ D : $\dfrac{1 + H(s)G(s) - G(s)}{1 + H(s)G(s)}$

Remark : The tracking error is always defined as $e(t) = r(t) - y(t)$.

Steady-State Error (Tracking performance)

Tracking Performance : Take $w(t) = 0, t > 0$.



$$E(s) = \frac{1}{1 + G(s)D_c(s)} R(s) \quad \Rightarrow \quad e(t) = \mathcal{L}^{-1}\{E(s)\}$$

Steady-state error : We use the final value theorem,

$$\lim_{t \to \infty} e(t) = \lim_{s \to 0} sE(s) = \lim_{s \to 0} s \frac{1}{1 + G(s)D_c(s)} R(s)$$

The number of poles of $G(s)D_c(s)$ at zero (or the number of integrators) is important for tracking performance

Steady-State Error (Tracking performance)

No integrator in $G(s)D_c(s)$

Steady-state tracking error for a unit step reference :

- If $r(t) = \mathbf{1}(t)$ is a unit step signal, i.e. $R(s) = \dfrac{1}{s}$

$$\lim_{t\to\infty} e(t) = \lim_{s\to 0} sE(s) = \lim_{s\to 0} s\frac{1}{1+G(s)D_c(s)}\frac{1}{s} = \frac{1}{1+K_p}$$

where $\boxed{K_p = \lim_{s\to 0} G(s)D_c(s)}$ is called the "position error constant".

- if $r(t) = t\mathbf{1}(t)$ is a unit ramp signal i.e. $R(s) = \dfrac{1}{s^2}$

$$\lim_{t\to\infty} e(t) = \lim_{s\to 0} sE(s) = \lim_{s\to 0} s\frac{1}{1+G(s)D_c(s)}\frac{1}{s^2} = \infty$$

- if $r(t) = \frac{1}{2}t^2\mathbf{1}(t)$ is a unit parabolic signal i.e. $R(s) = \dfrac{1}{s^3}$

$$\lim_{t\to\infty} e(t) = \lim_{s\to 0} sE(s) = \lim_{s\to 0} s\frac{1}{1+G(s)D_c(s)}\frac{1}{s^3} = \infty$$

Steady-State Error (Tracking performance)

One integrator in $G(s)D_c(s)$

Steady-state tracking error for a unit step reference :

- If $r(t) = \mathbf{1}(t)$ is a unit step signal, i.e. $R(s) = \dfrac{1}{s}$

$$\lim_{t\to\infty} e(t) = \lim_{s\to 0} sE(s) = \lim_{s\to 0} s\frac{1}{1+G(s)D_c(s)}\frac{1}{s} = \frac{1}{1+\infty} = 0$$

- if $r(t) = t\mathbf{1}(t)$ is a unit ramp signal i.e. $R(s) = \dfrac{1}{s^2}$

$$\lim_{t\to\infty} e(t) = \lim_{s\to 0} sE(s) = \lim_{s\to 0} s\frac{1}{1+G(s)D_c(s)}\frac{1}{s^2} = \lim_{s\to 0}\frac{1}{s+sG(s)D_c(s)} = \frac{1}{K_v}$$

where $\boxed{K_v = \lim_{s\to 0} sG(s)D_c(s)}$ is called the "velocity error constant".

- if $r(t) = \frac{1}{2}t^2\mathbf{1}(t)$ is a unit parabolic signal i.e. $R(s) = \dfrac{1}{s^3}$

$$\lim_{t\to\infty} e(t) = \lim_{s\to 0} sE(s) = \lim_{s\to 0} s\frac{1}{1+G(s)D_c(s)}\frac{1}{s^3} = \infty$$

Steady-State Error (Tracking performance)

Two integrators in $G(s)D_c(s)$

Steady-state tracking error for a unit step reference :

- If $r(t) = \mathbf{1}(t)$ is a unit step signal, i.e. $R(s) = \dfrac{1}{s}$

$$\lim_{t\to\infty} e(t) = \lim_{s\to 0} sE(s) = \lim_{s\to 0} s\frac{1}{1+G(s)D_c(s)}\frac{1}{s} = \frac{1}{1+\infty} = 0$$

- if $r(t) = t\mathbf{1}(t)$ is a unit ramp signal i.e. $R(s) = \dfrac{1}{s^2}$

$$\lim_{t\to\infty} e(t) = \lim_{s\to 0} sE(s) = \lim_{s\to 0} s\frac{1}{1+G(s)D_c(s)}\frac{1}{s^2} = \lim_{s\to 0}\frac{1}{s+sG(s)D_c(s)} = 0$$

- if $r(t) = \frac{1}{2}t^2\mathbf{1}(t)$ is a unit parabolic signal i.e. $R(s) = \dfrac{1}{s^3}$

$$\lim_{t\to\infty} e(t) = \lim_{s\to 0} sE(s) = \lim_{s\to 0} s\frac{1}{1+G(s)D_c(s)}\frac{1}{s^3} = \lim_{s\to 0}\frac{1}{s^2+s^2G(s)D_c(s)} = \frac{1}{K_a}$$

where $\boxed{K_a = \lim_{s\to 0} s^2 G(s)D_c(s)}$ is called the "acceleration error constant".

Steady-State Error (Tracking performance)

Tracking steady-state error vs number of integrators in $G(s)D_c(s)$

Reference	$r(t) = \mathbf{1}(t)$	$r(t) = t\mathbf{1}(t)$	$t(t) = \frac{1}{2}t^2\mathbf{1}(t)$
No integrator	$\dfrac{1}{1+K_p}$	∞	∞
One integrator	0	$\dfrac{1}{K_v}$	∞
Two integrators	0	0	$\dfrac{1}{K_a}$

- More integrator in the open-loop transfer function gives better tracking performance.
- Higher position, velocity and acceleration constants give smaller tracking steady-state error.

Questions

- Do we have the same property for disturbance rejection ?
- Can we add many integrators into the controller to be able to follow all signals ?

Steady-State Errors

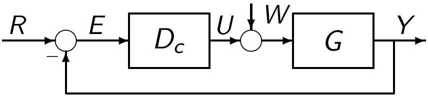
Question : Exam 2015

Consider $G(s) = \dfrac{s + 100}{s(s+2)(s+30)}$, with a proportional controller $D_c(s) = 10$.

1. Compute the steady-state errors for tracking a unit step **reference** signal.
   (A) 0 **(B)** ∞ **(C)** 0.06 **(D)** 16.67
2. Compute the steady-state errors for tracking a unit ramp **reference** signal.
   (A) 0 **(B)** ∞ **(C)** 0.06 **(D)** 16.67

Steady-State Error (Regulation Performance)

Regulation Performance : Take $r(t) = 0, k > 0$



$$E(s) = \frac{-G(s)}{1 + G(s)D_c(s)} W(s) \quad \Rightarrow \quad e(t) = \mathcal{L}^{-1} E(s)$$

The steady-state error for regulation is :

$$\lim_{t \to \infty} e(t) = \lim_{s \to 0} sE(s) = \lim_{s \to 0} s \frac{-G(s)}{1 + G(s)D_c(s)} W(s)$$

The number of poles of the controller $D_c(s)$ at zero (the number of the integrators in the controller) is important for regulation performance

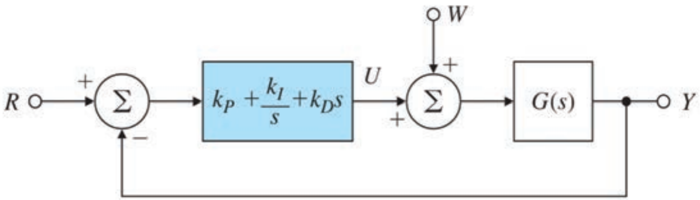
Steady-state Error (Regulation Performance)

Disturbance rejection

Given $D_c(s) = 10$ in closed-loop with $G(s) = \dfrac{(s+100)}{s(s+2)}$:

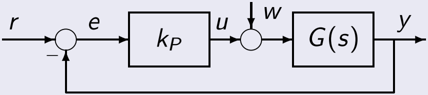
1. Compute the steady-state error for a unit step **disturbance** signal.
   (A) -0.002 **(B)** -0.1 **(C)** 0 **(D)** $-\infty$
2. Compute the steady-state error for a unit ramp **disturbance** signal.
   (A) -0.002 **(B)** -0.1 **(C)** 0 **(D)** $-\infty$

PID Controllers



- Proportional Control (P)
- Proportional Integral Control (PI)
- Proportional Derivative Control (PD)
- PID Control
- Ziegler-Nichols Tuning Method
- Model Reference Control Tuning Method
- Digital Implementation of PID Controllers
- Windup Problem

Proportional Controller



Control signal is proportional to the error signal.

$$u(t) = k_P e(t) \quad \Rightarrow U(s) = k_P E(s)$$

In order to have a non-zero control signal we SHOULD have a tracking error !

Tracking a unit step signal

Assume that $r(t) = 1$ and $R(s) = 1/s$ (no disturbance $w(t) = 0$). Find the steady-state tracking error :

A : 0 B : $\dfrac{k_P G(0)}{1 + k_P G(0)}$

C : $\dfrac{1}{1 + k_P G(0)}$ D : $\dfrac{k_P}{1 + k_P G(0)}$

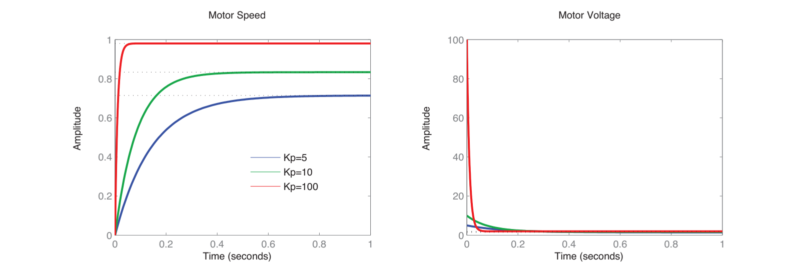
Proportional Controller

Example (Speed Control of a DC Motor)

The transfer function between the angular speed and the armature voltage in a DC motor is :

$$G(s) = \frac{\Omega_m(s)}{V_a(s)} = \frac{K}{\tau s + 1}$$

Take $K = \tau = 1$ and compute the speed $\omega_m(t)$ and the control voltage $v_a(t)$ of the motor for $r(t) = \mathbf{1}(t)$ in closed-loop operation with a P controller k_P.



- Higher k_P leads to faster response and less steady state error.
- Higher k_P leads to higher required voltage (stronger amplifier).

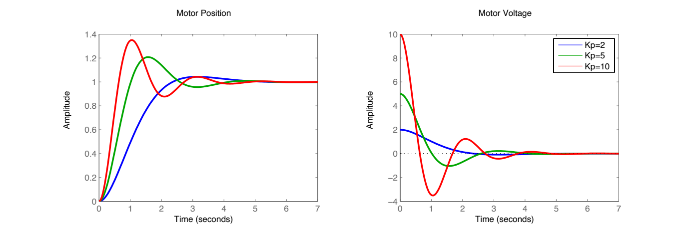
Proportional Controller

Example (Position Control of a DC Motor)

The transfer function between the angular position and the armature voltage in a DC motor is :

$$G(s) = \frac{\Theta_m(s)}{V_a(s)} = \frac{K}{s(\tau s + 1)}$$

Take $K = \tau = 1$ and compute the speed $\theta_m(t)$ and the control voltage $v_a(t)$ of the motor for $r(t) = \mathbf{1}(t)$ in closed-loop operation with a P controller k_P.



- The steady-state tracking error is zero because $G(s)$ has one integrator.
- Higher k_P leads to faster response and oscillation and higher required voltage (stronger amplifier).

Proportional-Integral Controller

PI Controller : The control input is given by : $u(t) = k_P e(t) + k_I \int_0^t e(\tau) d\tau$

$$\Rightarrow \quad U(s) = \left(k_P + \frac{k_I}{s}\right) E(s) \quad \Rightarrow \quad \boxed{D_c(s) = \frac{k_P s + k_I}{s}}$$

- The controller includes one integrator, so the steady-state error for constant disturbance is zero (the step disturbance is rejected).

$$E(s) = \frac{-G}{1 + GD_c} W(s) \quad \Rightarrow \quad e(\infty) = \lim_{s \to 0} s \frac{-G}{1 + GD_c} \frac{1}{s} = \frac{-G(0)}{1 + G(0)D_c(0)} = 0$$

- The open-loop transfer function GD_c includes at least one integrator so the steady-state error for tracking step reference signals is zero.

$$E(s) = \frac{1}{1 + GD_c} R(s) \quad \Rightarrow \quad e(\infty) = \lim_{s \to 0} s \frac{1}{1 + GD_c} \frac{1}{s} = \frac{1}{1 + G(0)D_c(0)} = 0$$

PI controller is the most used feedback controller in the world !

Proportional Integral Controller (Transient Response)

Example (Speed Control of a DC Motor with PI Controller)

Consider the closed-loop speed control of the DC motor with a PI controller :

$$G(s) = \frac{\Omega_m(s)}{V_a(s)} = \frac{K}{\tau s + 1} \qquad ; \qquad D_c(s) = k_P + \frac{k_I}{s}$$

Closed-loop transfer function : between the motor speed $\omega_m(t)$ and the reference $r(t)$ is :

$$\frac{\Omega_m(s)}{R(s)} = \frac{GD_c}{1 + GD_c} = \frac{\frac{K(k_P s + k_I)}{s(\tau s+1)}}{1 + \frac{K(k_P s + k_I)}{s(\tau s+1)}} = \frac{K(k_P s + k_I)}{\tau s^2 + (1 + Kk_P)s + Kk_I}$$

Transient response : depends on the value of ζ and ω_n, where

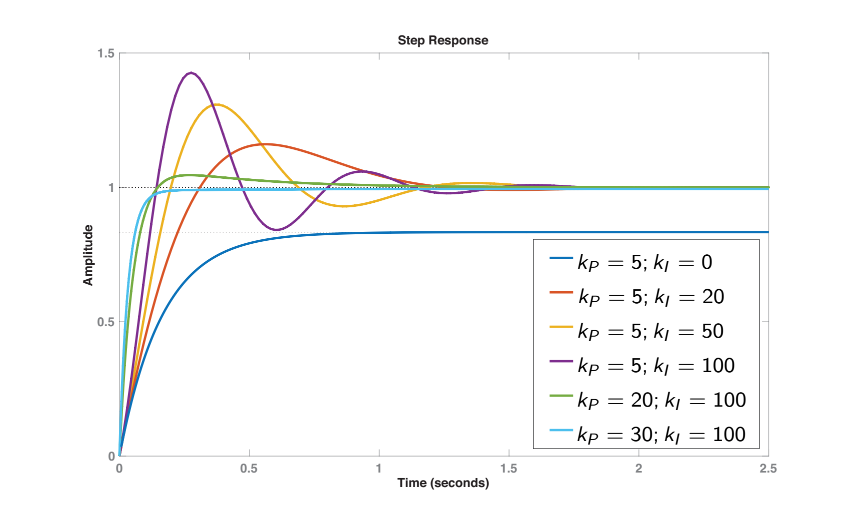
$$2\zeta\omega_n = \frac{Kk_P + 1}{\tau} \qquad \omega_n = \sqrt{\frac{Kk_I}{\tau}}$$

Remark : For a fixed k_P, increasing k_I will increase the natural frequency (faster response) and decrease the damping factor (more oscillation). For a fixed k_I, increasing k_P will increase the damping factor.

Proportional Integral Controller (Transient Response)

Example (Tracking performance of a DC motor speed control)

For $K = \tau = 1$



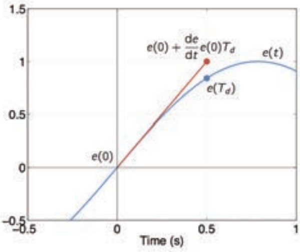
PD Controller : The control signal is proportional to the sum of the error signal and its derivative. The idea is to act proportionally to the future error, i.e. $u(t) = k_P e(t + T_d)$. A linear estimate of the future error is given by :

$$\frac{de(t)}{dt} \approx \frac{e(t + T_d) - e(t)}{T_d} \quad \Rightarrow e(t + T_d) \approx e(t) + T_d \frac{de(t)}{dt}$$

$$u(t) = k_P \left(e(t) + T_d \frac{de(t)}{dt} \right) \quad \Rightarrow \quad \boxed{U(s) = (k_P + k_D s) E(s)}$$

$$\boxed{D_c(s) = k_P + k_D s}$$

Remark : PD controller does not change the number of integrators, so it does not change the steady-state error but the transient response.



Proportional Derivative Controller

Example (Position Control of a DC Motor)

Consider the closed-loop position control of the DC motor with a PD controller :

$$G(s) = \frac{\Theta_m(s)}{V_a(s)} = \frac{K}{s(\tau s + 1)} \qquad ; \qquad D_c(s) = k_P + k_D s$$

Closed-loop transfer function : between the angular position $\theta_m(t)$ and the reference $r(t)$ is :

$$\frac{\Theta_m(s)}{R(s)} = \frac{GD_c}{1 + GD_c} = \frac{\frac{K(k_P + k_D s)}{s(\tau s+1)}}{1 + \frac{K(k_P + k_D s)}{s(\tau s+1)}} = \frac{K(k_P + k_D s)}{\tau s^2 + (1 + K k_D)s + K k_P}$$

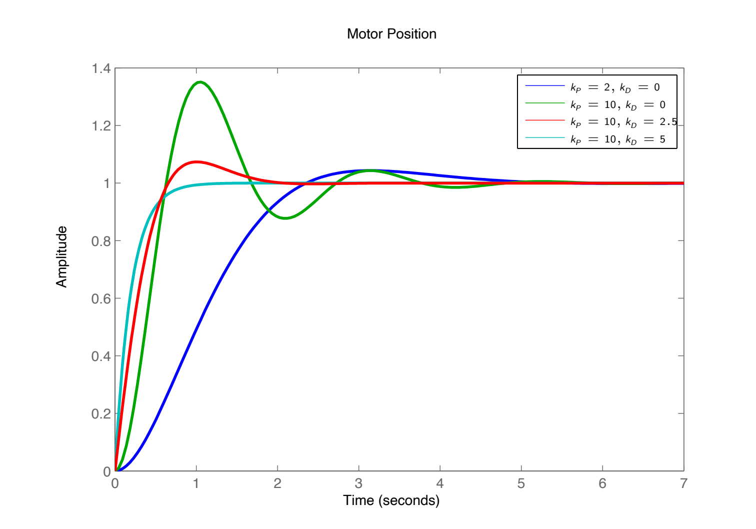
Transient response : depends on the value of ζ and ω_n, where

$$2\zeta\omega_n = \frac{1 + K k_D}{\tau} \qquad \omega_n = \sqrt{\frac{K k_P}{\tau}}$$

Remark : For a fixed k_D, increasing k_P will increase the natural frequency (faster response) and decrease the damping factor (more oscillation). For a fixed k_P, increasing k_D will increase the damping factor. So k_D has a stabilizing effect.

Proportional Derivative Controller (Transient Response)

Example (Tracking performance of a DC motor position control)



Proportional Derivative Controller

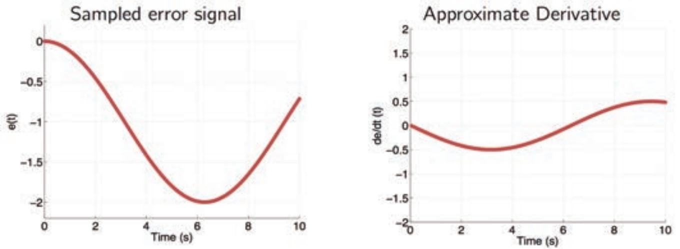
PD controller implementation : PD controller has a very large gain at high frequencies so it amplifies noise and abrupt change of the reference (note that the derivative of a step is a Dirac impulse).

$$|D_c(j\omega)| = |k_P + jk_D\omega| \quad \text{is large for large } \omega$$

The practical solution is to use a filtered version (low pass) :

$$D_c(s) = k_P + \frac{k_D s}{\tau s + 1} \qquad \text{where } \tau \ll k_D/k_P$$

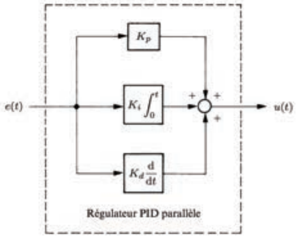
Example



Proportional Integral Derivative Controller

PID controller : Control signal is proportional to the error signal (current error), its integral (past errors) and its derivative (future errors).

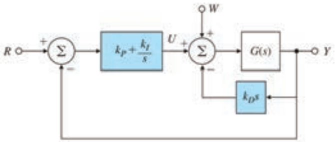
Parallel Structure :



$$u(t) = k_P e(t) + k_I \int_0^t e(\tau)d\tau + k_D \frac{de(t)}{dt}$$

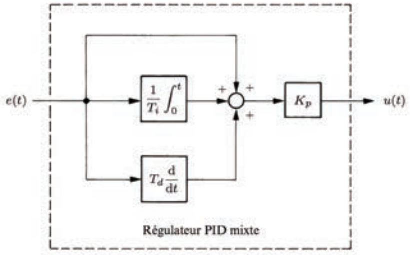
$$U(s) = \left(k_P + k_I \frac{1}{s} + k_D s \right) E(s)$$

Derivative in feedback : To not derivate the step changes in the error signal.



PID Controller Structures

Mixed Structure : In industry usually a mixed structure used for PID controllers.

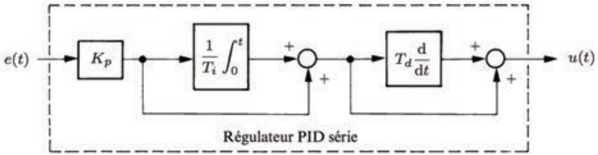


$$U(s) = K_p\left(1 + \frac{1}{T_i s} + T_d s\right) E(s)$$

with

$$K_p = k_P \quad T_i = k_P/k_I \quad T_d = k_D/k_P$$

Series Structure : with $D_c(s) = K_p\left(1 + \frac{1}{T_i s}\right)(1 + T_d s)$



Summary of PID Controllers

Proportional : Acts on the current error.

- Higher value gives better performance (faster response) in tracking and disturbance rejection.
- Higher value of gain leads to stronger control signal and may even destabilize the system.

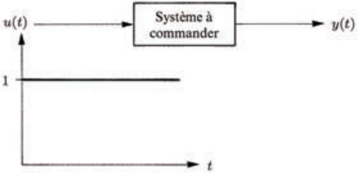
Integral : Acts on the past errors.

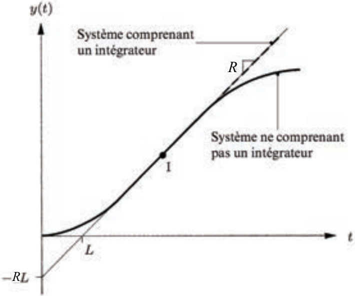
- Eliminates the steady state errors for step reference and step disturbance.
- Can make the system oscillatory or even unstable.

Derivative : Acts on the future errors.

- Increases the damping of the system, reduces the overshoot and improves the stability.
- Amplifies the measurement noise.
- Not often used in industry (difficult to be tuned manually).

First method of ZN : This method is based on the step response of the system in open-loop operation.





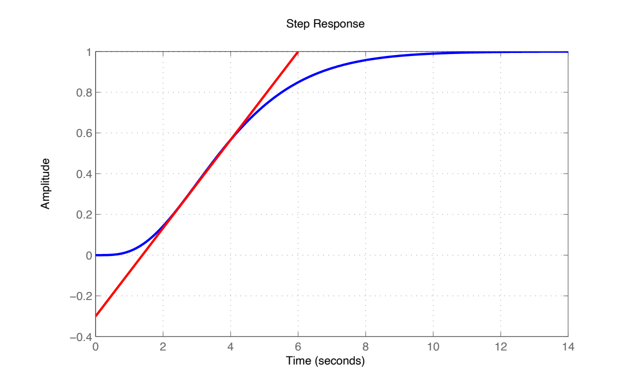
The PID controller parameters of a **Mixed Structure** are obtained as a function of L and R :

Type	K_p	T_i	T_d
P	$1/RL$		
PI	$0.9/RL$	$L/0.3$	
PID	$1.2/RL$	$2L$	$0.5L$

Ziegler-Nichols Tuning Methods

Example (**First method of ZN :**)

Consider the following 4th order system : $$G(s) = \frac{1}{(s+1)^4}$$



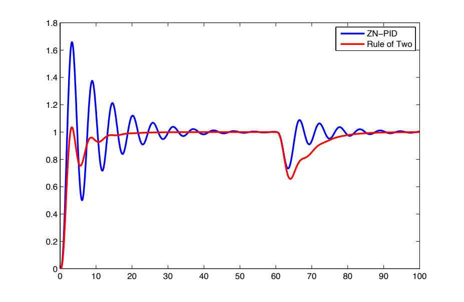
$$\Rightarrow RL = 0.3 \qquad \text{and} \qquad L = 1.3615$$

Ziegler-Nichols Tuning Methods

Example (**First method of ZN :)**

$$K_p = 1.2/RL = 4 \qquad T_i = 2 * L = 2.72 \qquad T_d = 0.5 * L = 0.68$$

$$\Rightarrow D_c(s) = 4\left(1 + \frac{1}{2.72s} + \frac{0.68s}{0.068s + 1}\right)$$



Rule of two : To obtain more damped results K_p is divided by two but T_i and T_d are multiplied by two.

Ziegler-Nichols Tuning Methods

Example

Use the first method of Ziegler-Nichols to design P, PI and PID controllers for the following system

$$G(s) = e^{-2s}\frac{0.5}{s}$$

Solution : The Laplace transform of the step response of the system is :

$$Y(s) = G(s)\frac{1}{s} = e^{-2s}\frac{0.5}{s^2}$$

In the time domain, $y(t)$ is the ramp signal $0.5t$ delayed by 2 second, so we get the parameters $R = 0.5$, $L = 2$, Therefore :

P controller : $K_p = 1/RL = 1$

PI controller : $K_p = 0.9/RL = 0.9, \quad T_i = 3.3L = 6.6$

PID controller : $K_p = 1.2/RL = 1.2, \quad T_i = 2L = 4, \quad T_d = 0.5L = 1$

Ziegler-Nichols Tuning Methods

Exercise

Consider the following dynamic system

$$G(s) = \frac{e^{-30s}}{100s+1}$$

Design by the first method of Ziegler-Nichols a PID controller.

1. Compute the step response :
   (A) $y(t) = 100(1 - e^{-30t})\mathbf{1}(t)$ **(B)** $y(t) = e^{-100(t-30)}\mathbf{1}(t-30)$
   (C) $y(t) = e^{-0.01t}\mathbf{1}(t-30)$ **(D)** $y(t) = (1 - e^{-0.01(t-30)})\mathbf{1}(t-30)$
2. Compute L and R.
   (A) R=30, L=100 **(B)** R=100, L=30
   (C) R=0.03, L=30 **(D)** R=0.01, L=30
3. Compute the PID controller.
   (A) $D_c(s) = 4(1 + \frac{60}{s} + 15s)$ **(B)** $D_c(s) = 4 + \frac{1}{60s} + 15s$
   (C) $D_c(s) = 4 + \frac{1}{15s} + 60s$ **(D)** $D_c(s) = 4(1 + \frac{15}{s} + 60s)$

Ziegler-Nichols Tuning Methods

Exercise

La relation entre la température du liquide $y(t)$ dans une cuve du mélange et la température du liquide d'alimentation $u(t)$ est donnée par :

$$\frac{dy(t)}{dt} + 2y(t) = 2u(t - 0.3)$$

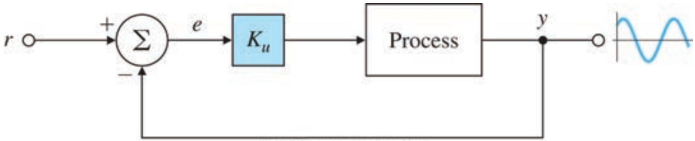
A partir de la réponse indicielle du système, dimensionner un régulateur PI par la première méthode de Ziegler-Nichols.

A : $K_p = 0.5,\ T_i = 0.5$ B : $K_p = 0.66,\ T_i = 1$

C : $K_p = 1.5,\ T_i = 1$ D : None of the above

Ziegler-Nichols Tuning Methods

Second method of ZN (Ultimate Sensitivity Method) : This method is based on increasing the gain of a proportional controller to bring the closed-loop system at the limit of stability.



Then the PID controller parameters for a **Mixed Structure** are functions of K_u and P_u (the period of oscillation) :

Type	K_p	T_i	T_d
P	$0.5K_u$		
PI	$0.45K_u$	$P_u/1.2$	
PID	$0.6K_u$	$0.5P_u$	$0.125P_u$

Ziegler-Nichols Tuning Methods

Example (**Second method of ZN :)**)

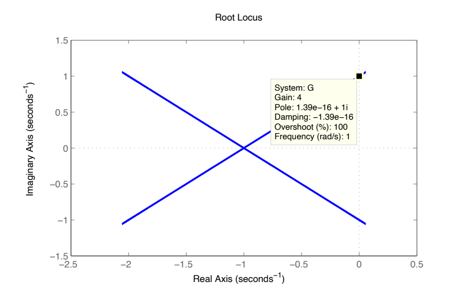
Consider the following 4th order system : $$G(s) = \frac{1}{(s+1)^4}$$

K_u and P_u can be computed by :

- Using a simulation program (e.g. Simulink)
- Finding the closed-loop poles :

$$(s+1)^4 + K = 0$$

When K goes from 0 to infinity. For all systems of order > 2 some of the closed-loop poles become unstable. The `rlocus` in Matlab can help finding K_u and P_u



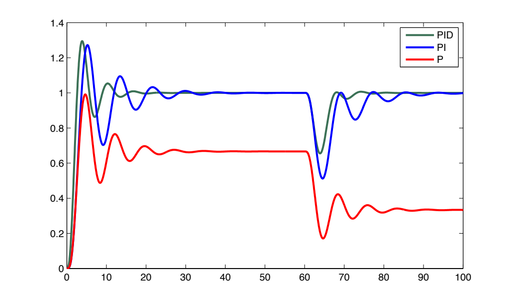
$$\Rightarrow K_u = 4 \quad \text{and} \quad P_u = 2\pi$$

Ziegler-Nichols Tuning Methods

Example (**Second method of ZN :)**

$$K_p = 0.6K_u = 2.4 \qquad T_i = 0.5 * P_u = 3.14 \qquad T_d = 0.125 * P_u = 0.785$$

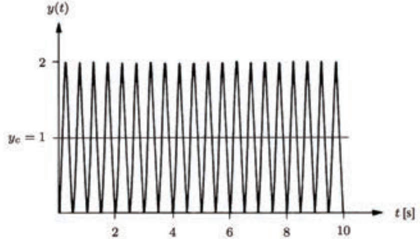
$$\Rightarrow D_c(s) = 2.4\left(1 + \frac{1}{3.14s} + \frac{0.785s}{0.078s + 1}\right)$$



Ziegler-Nichols Tuning Methods

Exercise

1.11.26 Un processus est commandé en boucle fermée par un régulateur proportionnel. Avec un gain valant 40,4, la réponse indicielle obtenue apparaît dans la figure 1.96.



Compute a PID controller based on the second method of ZN.

A) $K_p = 40.4$, $T_i = 0.5$, $T_d = 0.125$

B) $K_p = 24.24$, $T_i = 0.25$, $T_d = 0.0625$

C) $K_p = 24.24$, $T_i = 0.5$, $T_d = 0.125$

D) None of the above.

Ziegler-Nichols Tuning Methods

Summary of the ZN tuning methods :

- The ZN method is an **experimental** approach.
- It does not guarantee even the stability of the closed-loop system.
- The disturbance rejection transient is usually *acceptable* for many industrial systems.
- The tracking response is in general too aggressive. The rule of two can help improving the transients.
- The second method usually gives better response but it is more difficult to find K_u and P_u. In practice a relay used instead of the proportional gain.
- The ZN method can be used for initialization of the parameters that can be tuned manually afterward.

Online Manual Tuning

In practice the PID controllers are usually tuned online with no modeling ! Here, you have a heuristic method for tuning **parallel PID** controllers :

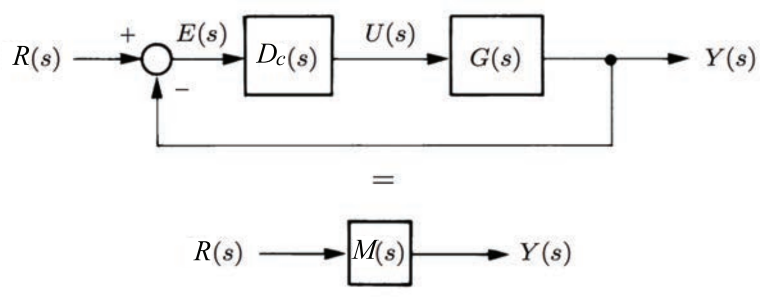
1. Put all parameters k_P, k_I and k_D to zero.
2. Increase k_P to obtain oscillation, then set k_P to half of the value.
3. Increase k_I to eliminate the steady-state error fast enough (too fast gives large overshoot).
4. Increase k_D (not too much), if needed, to reduce the overshoot.

Effect of increasing the parameters independently

	Rise time	**Overshoot**	**Settling time**	**Steady-state error**	**Stability**
k_P	Decrease	Increase	Minor	Decrease	Degrade
k_I	Decrease	Increase	Increase	Eliminate	Degrade
k_D	Minor	Decrease	Decrease	No effect	Improve for small k_D

Model Reference Control (MRC)

Principle : Given a stable and minimum phase model $G(s)$ of the plant, compute a controller $D_c(s)$ such that the closed-loop transfer function is equal to a reference model $M(s)$.



$$\frac{G(s)D_c(s)}{1+G(s)D_c(s)} = M(s) \qquad \Rightarrow \qquad D_c(s) = \frac{M(s)}{G(s)(1-M(s))}$$

Model Reference Control (MRC)

How to choose the reference model M?

- M should be stable with a steady-state gain equal to 1, i.e., $M(0) = 1$ (to track a step signal with zero steady-state error).
- M should satisfy the desired transient performance (rise-time, settling-time, overshoot, bandwidth).
- A simple first order choice is :

$$M(s) = \frac{1}{\tau_m s + 1} \qquad \begin{cases} \text{rise-time} = 2.2\tau_m, & \text{settling-time} = 3.9\tau_m \\ \text{overshoot\%} = 0, & \text{bandwidth} = 1/\tau_m \end{cases}$$

- A simple second-order choice is :

$$M(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \qquad \begin{cases} \text{rise-time} \approx \frac{1.8}{\omega_n} \\ \text{settling-time} \approx 3.9/\zeta\omega_n \\ \text{bandwidth} \approx \omega_n(1.85 - 1.19\zeta) \text{ for } 0.3 \leq \zeta \leq 0.8 \\ \text{overshoot} = e^{-\zeta\pi/\sqrt{1-\zeta^2}} \end{cases}$$

Model Reference Control

Case 1 (PI Controller)

Process model : $G(s) = \dfrac{\gamma}{\tau s + 1}$ **Reference model :** $M(s) = \dfrac{1}{\tau_m s + 1}$

Remarks :

- This reference model has a steady-state gain of 1, which leads to zero steady-state tracking error. Why ?
- Its step response has no overshoot, a rise-time of $2.2\tau_m$ and a settling time (2%) of $3.9\tau_m$. The bandwidth of reference model is $1/\tau_m$.
- τ_m is chosen based on performance specifications.

$$D_c(s) = \frac{M(s)}{G(s)(1 - M(s))} = \frac{\frac{1}{\tau_m s + 1}}{\frac{\gamma}{\tau s + 1}\left(1 - \frac{1}{\tau_m s + 1}\right)} = \frac{\tau}{\gamma \tau_m}\left(1 + \frac{1}{\tau s}\right)$$

$$D_c(s) = k_P + \frac{k_I}{s} \quad \Rightarrow \quad k_P = \frac{\tau}{\gamma \tau_m} \quad ; \quad k_I = \frac{1}{\gamma \tau_m}$$

Model Reference Control

Case 2 (PD Controller)

Process model : $G(s) = \dfrac{\gamma}{s(\tau s + 1)}$ **Reference model :** $M(s) = \dfrac{1}{\tau_m s + 1}$

$$D_c(s) = \frac{M(s)}{G(s)(1 - M(s))} = \frac{\frac{1}{\tau_m s+1}}{\frac{\gamma}{s(\tau s+1)}\left(1 - \frac{1}{\tau_m s+1}\right)} = \frac{1}{\gamma \tau_m}(1 + \tau s)$$

$$D_c(s) = k_P + k_D s \quad \Rightarrow \quad k_p = \frac{1}{\gamma \tau_m} \quad ; \quad k_D = \frac{\tau}{\gamma \tau_m}$$

Case 3 (PID controller)

$$G(s) = \frac{\gamma}{(\tau_1 s + 1)(\tau_2 s + 1)} \quad \text{and} \quad M(s) = \frac{1}{\tau_m s + 1}$$

$$D_c(s) = \frac{M(s)}{G(s)(1 - M(s))} = \frac{1}{G(s)\tau_m s} = \frac{(\tau_1 s + 1)(\tau_2 s + 1)}{\gamma \tau_m s}$$

$$D_c(s) = \frac{\tau_1 + \tau_2}{\gamma \tau_m} + \frac{1}{\gamma \tau_m s} + \frac{\tau_1 \tau_2}{\gamma \tau_m} s$$

Model Reference Control

Case 4 (PID Controller)

$$G(s) = \frac{\gamma\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \quad \text{and} \quad M(s) = \frac{1}{\tau_m s + 1}$$

$$D_c(s) = \frac{M(s)}{G(s)(1 - M(s))} = \frac{1}{G(s)\tau_m s} = \frac{s^2 + 2\zeta\omega_n s + \omega_n^2}{\gamma\omega_n^2\tau_m s}$$

$$\Rightarrow k_P = \frac{2\zeta}{\gamma\omega_n\tau_m} \quad ; \quad k_I = \frac{1}{\gamma\tau_m} \quad ; \quad k_D = \frac{1}{\gamma\omega_n^2\tau_m}$$

Remarks :

- The resonance mode of the plant are cancelled by the controller.
- If the resonance modes are not well damped ($\zeta << 1$), the disturbance response remains oscillatory.

$$\frac{Y(s)}{W(s)} = \frac{G(s)}{1 + G(s)D_c(s)} = \frac{\gamma\omega_n^2\tau_m s}{(\tau_m s + 1)(s^2 + 2\zeta\omega_n s + \omega_n^2)}$$

Model Reference Control

Case 5 (Systems with delay)

$$G(s) = \frac{\gamma e^{-\theta s}}{\tau s + 1} \quad \text{and} \quad M(s) = \frac{e^{-\theta s}}{\tau_m s + 1}$$

$$D_c(s) = \frac{M(s)}{G(s)(1 - M(s))} = \frac{e^{-\theta s}}{G(s)(1 + \tau_m s - e^{-\theta s})} = \frac{1 + \tau s}{\gamma(\tau_m s + 1 - e^{-\theta s})}$$

Let's approximate $e^{-\theta s} \approx 1 - \theta s$, which gives :

$$D_c(s) = \frac{1 + \tau s}{\gamma s(\tau_m + \theta)} \quad \Rightarrow \quad k_P = \frac{\tau}{\gamma(\tau_m + \theta)} \quad ; \quad k_I = \frac{1}{\gamma(\tau_m + \theta)}$$

If we take $e^{-\theta s} \approx \dfrac{1 - \theta s/2}{1 + \theta s/2}$, we have : $D_c(s) = \dfrac{(1 + \tau s)(1 + \theta s/2)}{\gamma(\tau_m + \theta)s\left(1 + \frac{\tau_m \theta}{2(\tau_m + \theta)} s\right)}$

Ignoring the high frequency pole of $D_c(s)$, a PID controller is obtained :

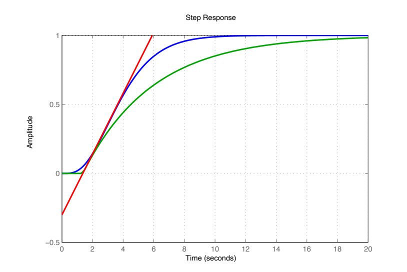
$$k_P = \frac{\theta/2 + \tau}{\gamma(\tau_m + \theta)} \quad ; \quad k_I = \frac{1}{\gamma(\tau_m + \theta)} \quad ; \quad k_D = \frac{\tau\theta}{2\gamma(\tau_m + \theta)}$$

Model Reference Control

Example

Consider the following 4th order system :

$$G(s) = \frac{1}{(s+1)^4} \approx \frac{e^{-1.36s}}{4.54s+1}$$



We have $\tau = 4.54$ and $\theta = 1.36$. For a desired closed-loop bandwidth of 0.2 rad/s, we choose $\tau_m = 5$:

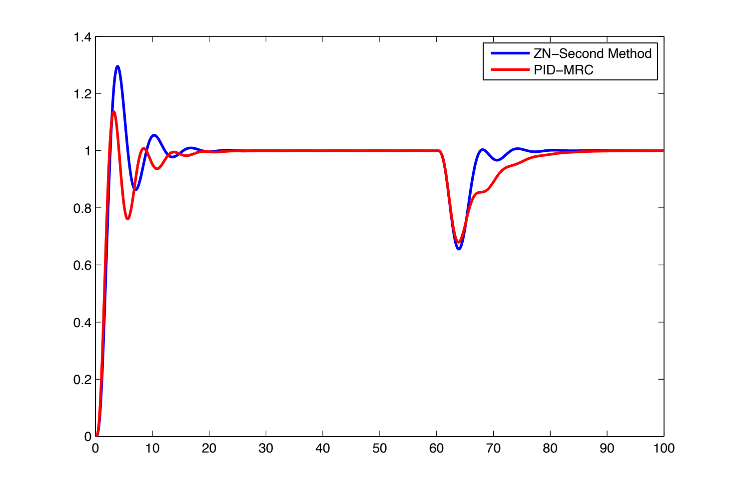
$$M(s) = \frac{e^{-1.36s}}{5s+1}$$

$$\Rightarrow \text{ PI-MRC } : \quad D_c(s) = \frac{1+\tau s}{\gamma s(\tau_m + \theta)} = \frac{4.54}{6.36} + \frac{1}{6.36s}$$

$$\Rightarrow \text{ PID-MRC} : k_P = \frac{1.36/2 + 4.54}{6.36}; k_I = \frac{1}{6.36}; k_D = \frac{4.54 \times 1.36}{2 \times 6.36}$$

Example

Comparison with second method of ZN



Model Reference Control

Example

Consider the following dynamic system

$$G(s) = \frac{e^{-30s}}{100s+1}$$

Compute PI and PID controllers using the MRC method to have a bandwidth of 0.02 rad/s.

Solution :

$$M(s) = \frac{e^{-30s}}{50s+1} \qquad \tau_m = 1/0.02 = 50, \quad \gamma = 1, \quad \tau = 100, \quad \theta = 30$$

PI controller : $\quad k_P = \frac{\tau}{\gamma(\tau_m+\theta)} = \frac{100}{80} = 1.25, \quad k_I = \frac{1}{80}$

PID controller : $\quad k_P = \frac{\theta/2+\tau}{\gamma(\tau_m+\theta)} = 1.437, \quad k_I = \frac{1}{\gamma(\tau_m+\theta)} = \frac{1}{80}$

$k_D = \frac{\tau\theta}{2\gamma(\tau_m+\theta)} = \frac{3000}{160}$

Model Reference Control

Exercise (Exam-2015)

La relation entre la température du liquide $y(t)$ dans une cuve du mélange et la température du liquide d'alimentation $u(t)$ est donnée par :

$$\frac{dy(t)}{dt} + 2y(t) = 2u(t - 0.3)$$

Dimensionner un régulateur PI par la méthode du modèle à poursuivre (*Model Reference Control*) pour obtenir une bande passante de 2.5 rad/s en boucle fermée.

A : $k_P = 0.357, k_I = 0.5$ B : $k_P = 1.4, k_I = 0.7$

C : $k_P = 0.714, k_I = 1.428$ D : None of the above

PID Control Implementation

The output of the PID controller (control law) is computed as :

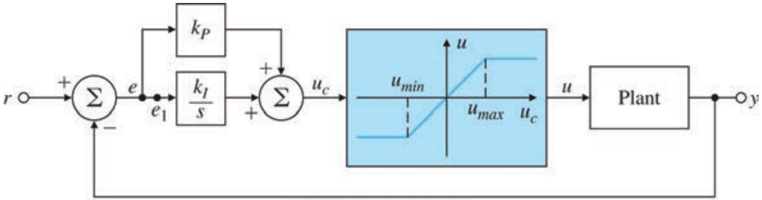
$$u(t) = k_P e(t) + k_I \int_0^t e(\tau) d\tau + k_D \frac{de(t)}{dt}$$

In practice, the control input is computed in a microcontroller using the sampled data with a very small sampling period ($h \ll$ dominant time constant of the plant model) :

1. The output $y(t)$ is sampled at instant kh to obtain $y(kh)$.
2. The tracking error is computed $e(kh) = r(kh) - y(kh)$.
3. The proportional part is computed $u_p(kh) = k_P e(kh)$.
4. The integral part is computed recursively $u_i(kh) = u_i(kh - h) + k_I h e(kh)$.
5. The derivative part is computed $u_d(kh) = k_D(e(kh) - e(kh - h))/h$.
6. The control input is : $u(kh) = u_p(kh) + u_i(kh) + u_d(kh)$.

Integrator Windup

Windup effect : If a controller has an integrator (almost always) and the actuator has some input constraints (almost always), the closed-loop system will face the windup problem.



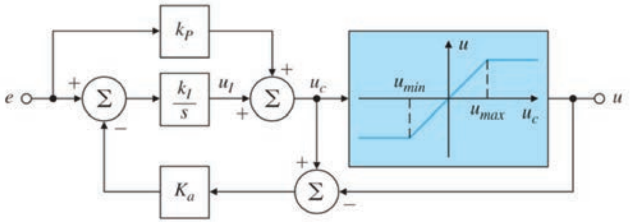
- For large errors, the output of the controller u_c will be saturated.
- Thus, the output y will grow slower than what was supposed to in linear simulation (without the saturation block).
- The error signal decays slowly and so its integral continues to grow.
- When finally y attains r and the error signal becomes negative, u_c is still large and positive and so y continues to grow.
- This creates large overshoot and oscillation at the output.

Anti-Windup Methods

- **Method 1 :** Turn off the integrator if the input is saturated.

$$
\begin{aligned}
& u_i(kh) = u_i(kh-h) + k_I h e(kh) \\
& \text{if } |u_p(kh) + u_i(kh)| > u_{\max}, \\
& \qquad u_i(kh) = u_i(kh-h) \\
& \text{end} \\
& u_c(kh) = u_p(kh) + u_i(kh)
\end{aligned}
$$

- **Method 2 :** When the input is saturated reduce the error signal.

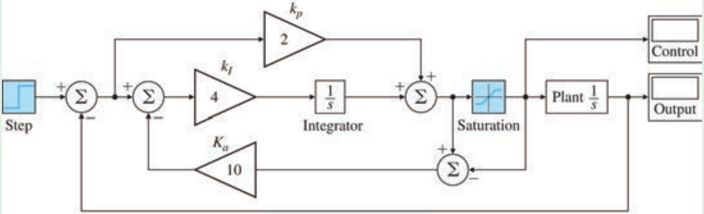


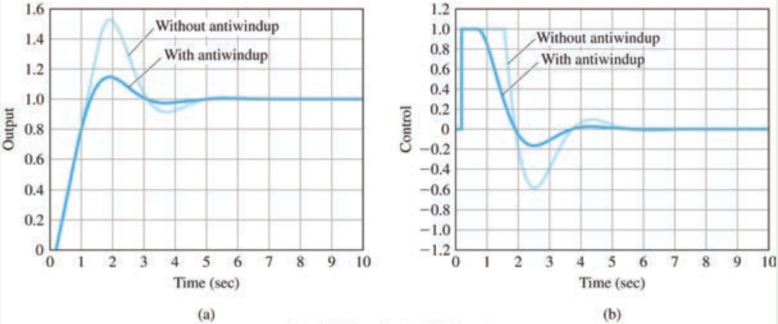
PID Control Implementation Code

```
err= ref - mes;
up = err * kp;
ud = (err-err_1) * kp *Td/h;
// err_1: err(k-1)
ui = ui_1 + kp/Ti*(err + err_1)*h/2;

if (Ti_On==0) ui=0;
// de/activate Ti part
if (Td_On==0) ud=0;
// de/activate Td part

tmp = up + ud + ui + u0;
// u0: feedforward cmd

if  (tmp>umax) cmd = max;
// saturations
     else if  (tmp<umin) cmd = umin;
                 else cmd = tmp;

tui1=ui;
tui2=tmp-cmd;
tmp_ui=ui;
if ((ARW!=0) && (tmp!=cmd)) {
// if ARW is ON and if there is a saturation
     if(abs(ui) > abs(ui_1)) {
// ARW with ui(k) = ui(k-1)
           ui = ui_1;
           }
    }
```

Anti-Windup Methods

Example





Feedforward Control

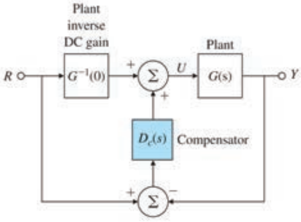
Feedforward controller : is a controller that leads, in open-loop operation, to perfect tracking of a reference signal $r(t)$. If the plant model is $G(s)$ then :

$$D_f(s) = G^{-1}(s)$$

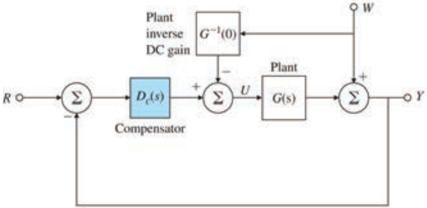
- This controller will not reject the disturbances.
- This scheme is very sensitive to the modelling errors.
- If $G(s)$ is unstable, the system becomes unstable.
- If $G(s)$ has RHP zeros, the feedforward controller becomes unstable.
- For stable systems, in general $D_f(s) = G^{-1}(0)$ is used that removes the steady state error.
- Usually $G(0)$ is known so with no integrator the zero steady-state error can be achieved.
- The feedforward controller is usually used with a feedback controller who takes care of stability, modelling error and disturbances.

Feedforward Control

Feedforward+Feedback control : A feedback control can be added to a feedforward control to reduce the sensitivity to the modelling error and reject the disturbances :



If the disturbance is measurable, it can be rejected with a similar scheme :



Feedforward Control

Example (Speed Control of a DC Motor)

The transfer function between the angular speed and the armature voltage in a DC motor is :

$$G(s) = \frac{\Omega_m(s)}{V_a(s)} = \frac{\gamma}{\tau s + 1}$$

Consider a proportional controller k_P and a feedforward controller $D_f(s) = G^{-1}(0)$. Compute the final value of the output $y(t)$:

$$Y(s) = \frac{k_P G(s)}{1 + k_P G(s)} R(s) + \frac{D_f(s) G(s)}{1 + k_P G(s)} R(s) = \frac{k_P G(s) + G^{-1}(0) G(s)}{1 + k_P G(s)} R(s)$$

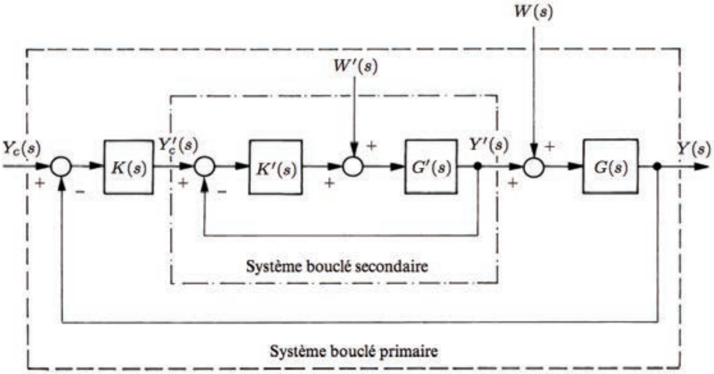
Therefore with $R(s) = 1/s$:

$$\lim_{t\to\infty} y(t) = \lim_{s\to 0} sY(s) = \frac{k_P G(0) + 1}{1 + k_P G(0)} = 1$$

So the output will track the unit step with no steady-state error.

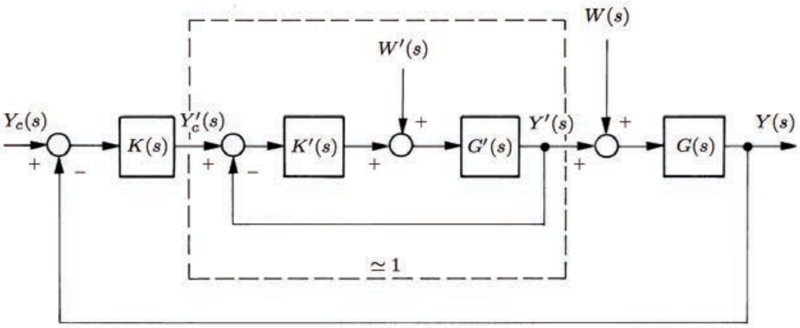
Cascade Controller

Cascade Control : Complicated systems can be divided to simple cascade systems and controlled by simple PID controllers.



Internal loop : Has a faster dynamic than the external loop (it has a larger bandwidth) and is designed first.
External loop : The controller of this loop is designed in the second step by ignoring the dynamic of the internal loop.

Cascade Controller



Advantages :

- PID controllers can be tuned, manually, for complex systems.
- Internal disturbances $W'(s)$ are taken into account more efficiently (faster).
- The internal signals can be limited by a saturation block (for system protection).

Cascade Controller

Example

Position control : The angular position of a DC motor can be controlled by a cascade controller with three loops : current loop, speed loop, position loop (current loop $\gg$ speed loop $\gg$ position loop)