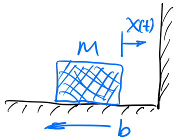
C. Control of Second-Order System



Model: $m\ddot{x} + B\dot{x} = F$

Objective: maintain a fixed position

$$\Rightarrow X_d(t) = \text{Constant} \Rightarrow \dot{X}_d(t) = \ddot{X}_d(t) = 0$$

Control: $F(t) = k_v(\underbrace{\dot{x}_d}_{0} - \dot{x}) + k_p(x_d - x) + m\underbrace{\ddot{x}_d}_{0}$

Apply the control algorithm to the system (model) to get the error dynamics:

$$m\ddot{x} + B\dot{x} = k_v(\dot{x}_d - \dot{x}) + k_p(x_d - x) + m\ddot{x}_d$$

$$\Rightarrow m(\ddot{x}_d - \ddot{x}) + k_v(\dot{x}_d - \dot{x}) + k_p(x_d - x) = B\dot{x} \; \underbrace{-B\dot{x}_d}_{0}$$

$$\Rightarrow m(\ddot{x}_d - \ddot{x}) + k_v(\dot{x}_d - \dot{x}) + B(\dot{x}_d - \dot{x}) + k_p(x_d - x) = 0$$

$$\Rightarrow m(\ddot{x}_d - \ddot{x}) + (k_v + B)(\dot{x}_d - \dot{x}) + k_p(x_d - x) = 0$$

Define $e(t) = x_d(t) - x(t)$, $\dot{e}(t) = \dot{x}_d(t) - \dot{x}(t)$, $\ddot{e}(t) = \ddot{x}_d - \ddot{x}$

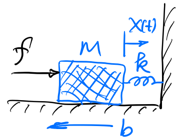
$$\Rightarrow m\ddot{e}(t) + (k_v + B)\dot{e}(t) + k_p e = 0$$ error dyn.

$$\Rightarrow \ddot{e}(t) + \underbrace{\frac{k_v + B}{m}}_{\hat{k}_v}\dot{e}(t) + \underbrace{\frac{k_p}{m}}_{\hat{k}_p} e(t) = 0 \Rightarrow S_{1,2}$$

We can choose k_v and k_p so that the pole positions are all located on the left-half of the S-plane, then $e(t) \xrightarrow{t \to \infty} 0 \Rightarrow X(t) \xrightarrow{t \to \infty} X_d(t)$

where $S_{1,2} = -\frac{\hat{k}}{2} \pm \frac{\sqrt{\hat{k}_v^2 - 4\hat{k}_p}}{2} = f(k_v, k_p)$, $\hat{k}_v \triangleq \frac{k_v + B}{m}$, $\hat{k}_p \triangleq \frac{k_p}{m}$

D. Control Law Partition and Trajectory Following Control



The mass-spring-friction system model:

$$m\ddot{x} + b\dot{x} + kx = f$$

The control law partition method partition the control into two parts.

The first part is model based, which is

$$f = \alpha f' + \beta$$

with $\alpha = m$ and $\beta = b\dot{x} + kx$

$\Rightarrow$ The error dynamics is

$$\cancel{m}\ddot{x} + \cancel{b\dot{x} + kx} = \alpha f' + \beta = \cancel{m} f' + \cancel{b\dot{x} + kx}$$

$$\Rightarrow \ddot{x} = f'$$

The 2nd-part of the control is servo part,

$$f' = \ddot{x}_d + k_v(\dot{x}_d - \dot{x}) + k_p(x_d - x)$$

$\Rightarrow$ The error dynamics now becomes

$$\ddot{x} = f' = \ddot{x}_d + k_v(\dot{x}_d - \dot{x}) + k_p(x_d - x)$$

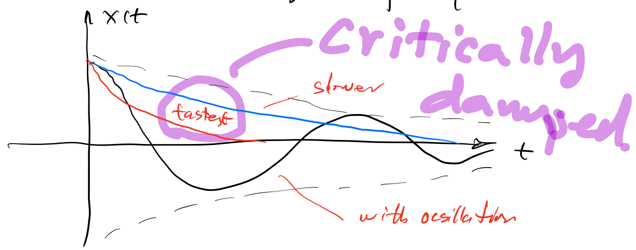
$$\Rightarrow (\ddot{x}_d - \ddot{x}) + k_v(\dot{x}_d - \dot{x}) + k_p(x_d - x) = 0$$

Define $e(t) = x_d(t) - x(t)$, $\dot{e}(t) = \dot{x}_d - \dot{x}$, $\ddot{e}(t) = \ddot{x}_d - \ddot{x}$.

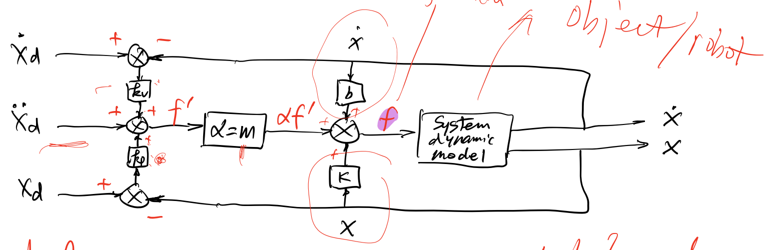
$$\Rightarrow \ddot{e}(t) + k_v\dot{e}(t) + k_p e(t) = 0$$ Standard

Let $\boxed{k_v = 2\sqrt{k_p}}$, then the error dynamics is going to behave in critically damped fashion for the response $\Rightarrow e(t) \xrightarrow[\text{critically damped}]{t\to\infty} 0 \Rightarrow x(t) \xrightarrow[\text{critically damped}]{t\to\infty} x_d(t)$

If $m=1$, $b=1$, $k_p=16$, $k_v = 2\sqrt{k_p} = 8$, then $e(t) \xrightarrow{t\to\infty} 0$ in critically damped fashion



In block diagram:

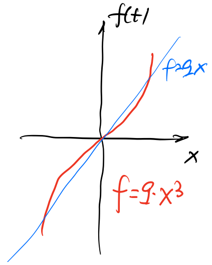


Control law partition
$$f = \alpha f' + \beta \quad \text{— model based}$$
$$= m\underbrace{[\ddot{x}_d + k_v(\dot{x}_d - \dot{x}) + k_p(x_d - x)]}_{f'} + b\dot{x} + kx$$
Servo-part

E Nonlinear Control of Robot Manipulators

A time-varying nonlinear system example.

mass-spring (nonlinear)-damper system:
$$m\ddot{x} + \underbrace{b\dot{x} + 9x^3}_{\text{nonlinear}} = f$$



Using the control law partition method:
$$f = \alpha f' + \beta = mf' + b\dot{x} + 9x^3 \quad \text{Exact Dyn. model}$$
with $f' = \ddot{x}_d + k_v(\dot{x}_d - \dot{x}) + k_p(x_d - x)$
$$\xrightarrow{e = x_d - x} \ddot{e} + k_v\dot{e} + k_p e = 0$$

choose k_v and k_p to make the system response in critically damped fashion $\Rightarrow e(t) \xrightarrow{t\to\infty} 0$, $x(t) \xrightarrow{t\to\infty} x_d(t)$

$\Rightarrow$ The control law partition works for both linear and nonlinear system but it has a significant flaw: It requires the exact and precise dynamic model to work! But in reality that is impossible!

F. MIMO Control Problem for Robot manipulators:

Computed Torque Control Method (control law partition)

The robot manipulator dynamic model is
$$M(\Theta)\ddot{\Theta} + V(\Theta, \dot{\Theta}) + G(\Theta) = \tau$$

Using the control law partition method, we have

$$\tau = \alpha \tau' + \beta \quad \text{where } \tau = R^{n\times 1}$$

and $\alpha = M(\Theta) \in R^{n\times n}$

$$\beta = V(\Theta, \dot{\Theta}) + G(\Theta) \in R^{n\times 1}$$

with the Servo-part as

$$\tau' = \ddot{\Theta}_d + K_v(\dot{\Theta}_d - \dot{\Theta}) + K_p(\Theta_d - \Theta)$$

Define error as : $E(t) = \Theta_d - \Theta$

$$\dot{E}(t) = \dot{\Theta}_d - \dot{\Theta}$$

$$\ddot{E}(t) = \ddot{\Theta}_d - \ddot{\Theta}$$

Then the error dynamics is

$$\cancel{M(\Theta)}\ddot{\Theta} + \cancel{V(\Theta,\dot{\Theta}) + G(\Theta)} = \cancel{M(\Theta)}\tau' + \cancel{V(\Theta,\dot{\Theta}) + G(\Theta)}$$

$$\Rightarrow \ddot{\Theta} = \tau' = \ddot{\Theta}_d + K_v(\dot{\Theta}_d - \dot{\Theta}) + K_p(\Theta_d - \Theta)$$

$$\Rightarrow \ddot{E} + K_v\dot{E} + K_pE = 0$$

where K_v and K_p are matrices. Choose K_v and K_p as diagonal matrices, then for j-th row:

$$\ddot{e}_j + k_{vj}\dot{e}_j + k_{pj}e_j = 0$$

Choose $k_{vj} = 2\sqrt{k_{pj}} \Rightarrow e_j(t) \xrightarrow[\text{Critically damped}]{t\to\infty} 0$, $\Theta \xrightarrow[\text{critically damped}]{t\to\infty} \Theta_d$

For $j = 1, 2, \cdots n$, then $\Theta(t) \xrightarrow[\text{Critically damped}]{t\to\infty} \Theta_d(t)$

Then the control law becomes:

$$\tau = \alpha\tau' + \beta = M\tau' + V + G$$

If let $\tau' \triangleq \ddot{\Theta}^*$, then the above control law becomes

$$\tau = M(\Theta)\cdot\ddot{\Theta}^* + V(\Theta,\dot{\Theta}) + G(\Theta)$$

Control law

$$\tau = M(\Theta)\ddot{\Theta} + V(\Theta,\dot{\Theta}) + G(\Theta)$$

dynamic model

This control algorithm has the same form as the robot dynamic model except that $\ddot{\Theta}$ is replaced by $\ddot{\Theta}^*$. So this control law is called "Computed Torque Method",

where

$$\ddot{\Theta}^* = \tau' = \ddot{\Theta}_d + K_v(\dot{\Theta}_d - \dot{\Theta}) + K_p(\Theta_d - \Theta)$$

Servo-Part.

II. Present Industrial Robot Control System

a. Individual Joint PID Control

Using control law partition method,

Let $\alpha = I_n$ identity matrix, no dynamic model

$\beta = 0 \implies$ no model needed.

and $\tau' = \ddot{\Theta}_d + \underbrace{K_v(\dot{\Theta}_d - \dot{\Theta})}_{\text{Derivative term}} + \underbrace{K_p(\Theta_d - \Theta)}_{\text{Proportional term}}$

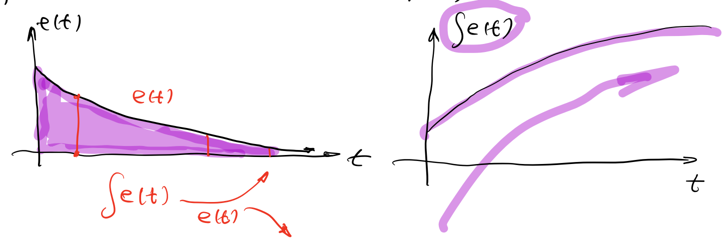
Servo:

If we add an integral term:

$$\tau' = \ddot{\Theta}_d + \underbrace{K_p(\Theta_d - \Theta)}_{P} + \underbrace{\cancel{K_I \int (\Theta_d - \Theta) dt}}_{I} + \underbrace{K_v(\dot{\Theta}_d - \dot{\Theta})}_{D}$$

If no integral term, it is called PD control.

Why do we need the I (integral) term in PID?



$$\tau' = \ddot{\Theta}_d + K_p E + K_I \int E + K_v \dot{E}$$

as $E \to 0$: $\tau' \to 0$, $K_p E \to 0$, $K_I \int E$ grows as $t \to \infty$, $K_v \dot{E} \to 0$

Small error is eliminated by the integral term?

The integral term works on the error signal, when the error signal becomes very small, the integral term becomes larger and larger so it drives the steady-state error to zero, eventually.

Features of Industrial PID/PD control:

① No dynamic model is needed.

② Choose K_v and K_p and sometimes K_I as diagonal matrices to decouple the effect, then this becomes individual joint control

$$\tau_j = k_{v_j} \dot{e}_j + k_{p_j} e_j + k_{I_j} \int e_j \, dt, \quad j = 1, 2, \dots n$$

③ The performance of the control depends on the values of the gains (K_v, K_p, K_I).
Normally better performance needs higher gain values $\implies$ more powerful motors, which are limited by physical actuator/motor capacities $\implies$ limited performance of the robot manipulator, especially in steady-state.