Ch9: Centralized Control - Position Control

Introduction

Closed-loop Dynamic Equation

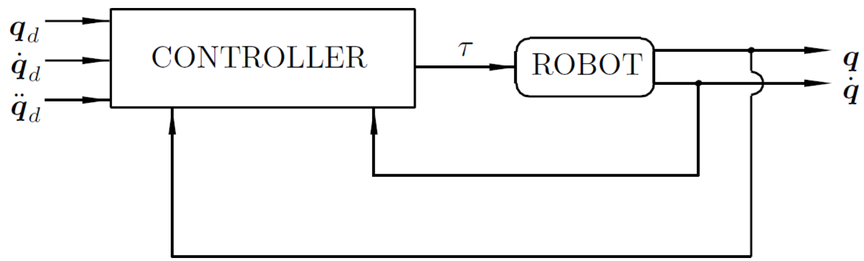
Consider the dynamic model of an n-DOF open-chain manipulator with no friction at the joints and no external force at the end-effector.

$$\boldsymbol{\tau} = \boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{g}(\boldsymbol{q}) \quad \text{or} \quad \frac{d}{dt}\begin{bmatrix}\boldsymbol{q}\\ \dot{\boldsymbol{q}}\end{bmatrix} = \begin{bmatrix}\dot{\boldsymbol{q}}\\ \boldsymbol{M}(\boldsymbol{q})^{-1}[\boldsymbol{\tau} - \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} - \boldsymbol{g}(\boldsymbol{q})]\end{bmatrix}$$

(state-space form)

In general, a position/motion **Control Law** (**Controller**) with desired joint position $\boldsymbol{q}_d(t) \in \mathbb{R}^n$, velocity $\dot{\boldsymbol{q}}_d(t) \in \mathbb{R}^n$, and acceleration $\ddot{\boldsymbol{q}}_d(t) \in \mathbb{R}^n$ can be expressed as a nonlinear function $\boldsymbol{\tau}_c$ as

$$\boldsymbol{\tau} = \boldsymbol{\tau}_c\big(\boldsymbol{q}, \dot{\boldsymbol{q}}, \boldsymbol{q}_d, \dot{\boldsymbol{q}}_d, \ddot{\boldsymbol{q}}_d, \boldsymbol{M}(\boldsymbol{q}), \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}}), \boldsymbol{g}(\boldsymbol{q})\big)$$



Note: For practical purposes, it is desirable that the controller $\boldsymbol{\tau}_c$ does not depend on the joint acceleration $\ddot{\boldsymbol{q}}$ since computing or measuring acceleration is usually highly sensitive to noise.

Closed-loop Dynamic Equation (cont.)

Thus, the **closed-loop dynamic equation** is derived as

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{g}(\boldsymbol{q}) = \boldsymbol{\tau}_c\big(\boldsymbol{q},\dot{\boldsymbol{q}},\boldsymbol{q}_d,\dot{\boldsymbol{q}}_d,\ddot{\boldsymbol{q}}_d,\boldsymbol{M}(\boldsymbol{q}),\boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}}),\boldsymbol{g}(\boldsymbol{q})\big)$$

or in the state-space form as

$$\frac{d}{dt}\begin{bmatrix}\boldsymbol{q}_d - \boldsymbol{q}\\ \dot{\boldsymbol{q}}_d - \dot{\boldsymbol{q}}\end{bmatrix} = \boldsymbol{f}\big(\boldsymbol{q},\dot{\boldsymbol{q}},\boldsymbol{q}_d,\dot{\boldsymbol{q}}_d,\ddot{\boldsymbol{q}}_d,\boldsymbol{M}(\boldsymbol{q}),\boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}}),\boldsymbol{g}(\boldsymbol{q})\big)$$

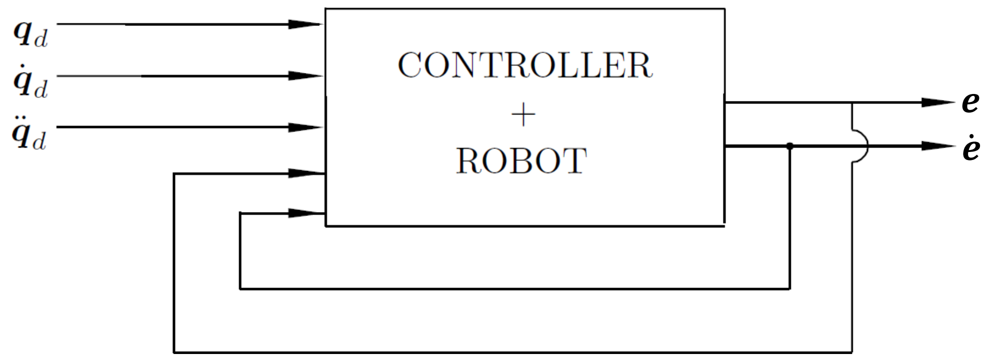
$\boldsymbol{e} = \boldsymbol{q}_d - \boldsymbol{q} \in \mathbb{R}^n,$

$\dot{\boldsymbol{e}} = \dot{\boldsymbol{q}}_d - \dot{\boldsymbol{q}} \in \mathbb{R}^n,$

and by replacing $\boldsymbol{q}$ with $\boldsymbol{q}_d(t) - \boldsymbol{e}$ and $\dot{\boldsymbol{q}}$ with $\dot{\boldsymbol{q}}_d(t) - \dot{\boldsymbol{e}}$ in $\boldsymbol{f}$:

$$\frac{d}{dt}\begin{bmatrix}\boldsymbol{e}\\ \dot{\boldsymbol{e}}\end{bmatrix} = \tilde{\boldsymbol{f}}(t,\boldsymbol{e},\dot{\boldsymbol{e}})$$

In general, a nonautonomous nonlinear ODE when $\boldsymbol{q}_d = \boldsymbol{q}_d(t)$.

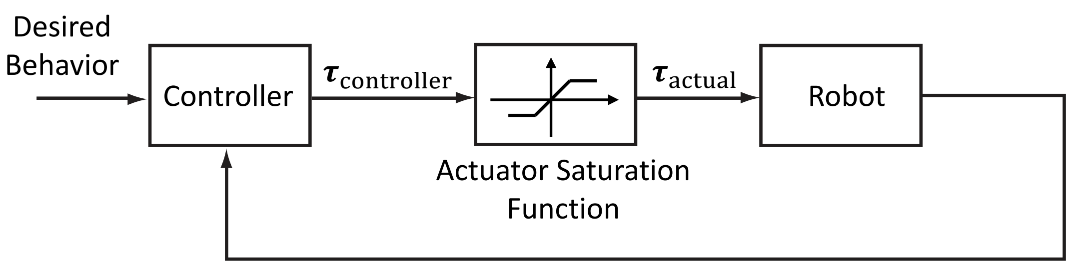


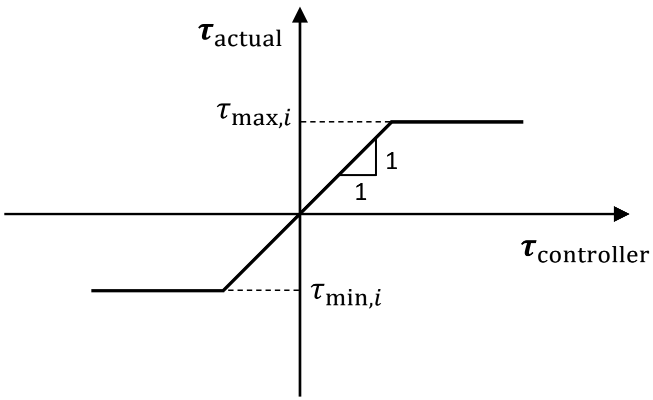
Actuator Saturation

In some controllers, choosing large values for the control parameters causes a large (initial) control torque which is beyond the robot actuators capacity which are limited by maximum and minimum allowable values $\boldsymbol{\tau}_{\text{max}}$, $\boldsymbol{\tau}_{\text{min}}$. Therefore, the control parameters should be chosen properly.

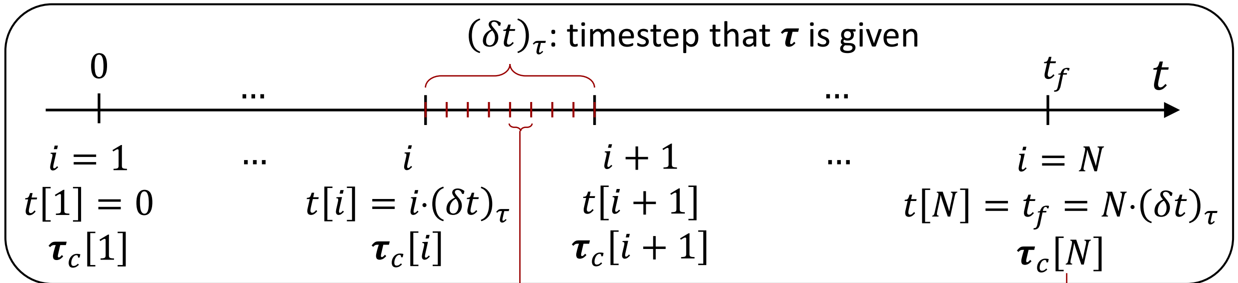
To consider the actuator saturation limits in the simulation, we add a saturation function as follows:

$$\boldsymbol{\tau}_{\text{actual}} = \text{sat}(\boldsymbol{\tau}_{\text{controller}})$$





Pseudocode for Controllers



$(\delta t)_{\ddot{\theta}} = (\delta t)_\tau / n_{\text{res}}$: timestep for motion simulation and computing $\ddot{\boldsymbol{\theta}}$.
n_{res}: Integration resolution.

N: Number of samples

$(\delta t)_\tau, (\delta t)_{\ddot{\theta}} \in \mathbb{R}^+$

Given $\boldsymbol{\mathcal{F}}_{\text{tip}}[i]$ $(i = 1, \dots, N)$
Set $\boldsymbol{\theta}[1] = \boldsymbol{\theta}(0)$, $\dot{\boldsymbol{\theta}}[1] = \dot{\boldsymbol{\theta}}(0)$, $\boldsymbol{e}_{\text{int}} = \boldsymbol{0}$
Set $\overline{\boldsymbol{\theta}} = \boldsymbol{\theta}[1]$, $\dot{\overline{\boldsymbol{\theta}}} = \dot{\boldsymbol{\theta}}[1]$
For $i = 1$ to $N - 1$
  $\left[\boldsymbol{\theta}_d[i], \dot{\boldsymbol{\theta}}_d[i]\right] = \text{trajectory}(t[i] = i \cdot (\delta t)_\tau, \dots)$
  $\boldsymbol{\tau}_c[i] = \text{controller}\left(\overline{\boldsymbol{\theta}}, \dot{\overline{\boldsymbol{\theta}}}, \boldsymbol{\theta}_d[i], \dot{\boldsymbol{\theta}}_d[i], \ddot{\boldsymbol{\theta}}_d[i], \boldsymbol{e}_{\text{int}}, \dots\right)$
  $\boldsymbol{\tau}_c[i] = \text{sat}(\boldsymbol{\tau}_c[i])$
  For $j = 1$ to n_{res}
    $\ddot{\boldsymbol{\theta}} = \text{ForwardDynamics}\left(\overline{\boldsymbol{\theta}}, \dot{\overline{\boldsymbol{\theta}}}, \boldsymbol{\tau}_c[i], \boldsymbol{\mathcal{F}}_{\text{tip}}[i]\right)$
    $\overline{\boldsymbol{\theta}} = \overline{\boldsymbol{\theta}} + \dot{\overline{\boldsymbol{\theta}}} \cdot (\delta t)_{\ddot{\theta}}$
    $\dot{\overline{\boldsymbol{\theta}}} = \dot{\overline{\boldsymbol{\theta}}} + \ddot{\boldsymbol{\theta}} \cdot (\delta t)_{\ddot{\theta}}$
  end
$\boldsymbol{e}_{\text{int}} = \boldsymbol{e}_{\text{int}} + (\delta t)_\tau (\boldsymbol{\theta}_d[i] - \overline{\boldsymbol{\theta}})$ % error integral
$\boldsymbol{\theta}[i+1] = \overline{\boldsymbol{\theta}}$
$\dot{\boldsymbol{\theta}}[i+1] = \dot{\overline{\boldsymbol{\theta}}}$
end

First-order Euler Integration
(we can also use any other ODE solver like **ode45** which is based on an explicit **Runge-Kutta** (4,5) formula)

Position Control

Position Control Objective

Given a desired constant joint position (set-point reference) $\boldsymbol{q}_d \in \mathbb{R}^n$, we wish to find joint torques/forces $\boldsymbol{\tau} \in \mathbb{R}^n$ such that the joint position $\boldsymbol{q}(t) \in \mathbb{R}^n$ tend to $\boldsymbol{q}_d$ accurately:

$$\lim_{t\to\infty} \boldsymbol{q}(t) = \boldsymbol{q}_d \quad \Rightarrow \quad \lim_{t\to\infty} \boldsymbol{e}(t) = \boldsymbol{0}$$

$$\boldsymbol{e}(t) = \boldsymbol{q}_d - \boldsymbol{q}(t) \in \mathbb{R}^n$$

position error

The most common position controllers:

- PD Control (or P Control Plus Velocity Feedback)
- PD Control with Gravity Compensation
- PD Control with Desired Gravity Compensation
- PID Control

PD Control

PD Control

(or P Control Plus Velocity Feedback)

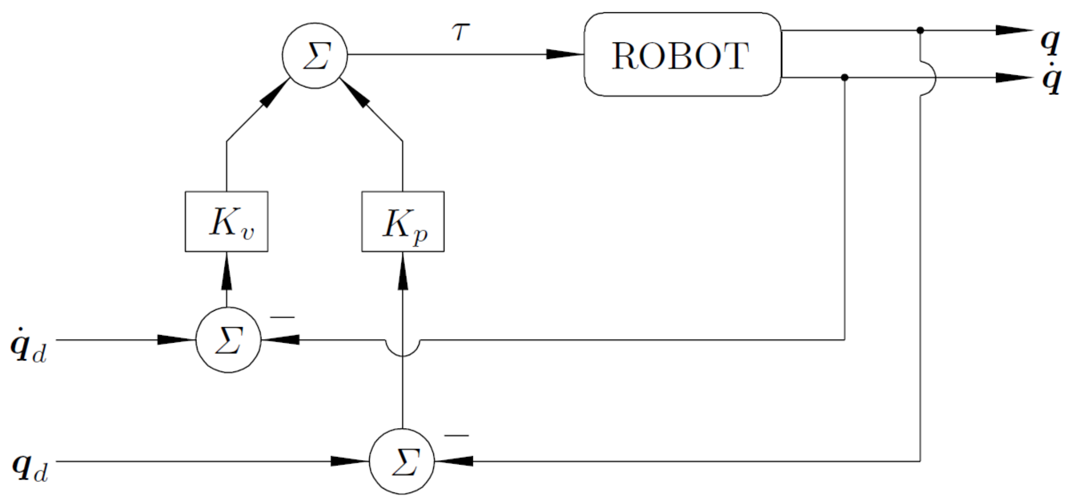
The PD (Proportional Derivative) control law is given by

$$\boldsymbol{\tau} = \boldsymbol{K}_p \boldsymbol{e} + \boldsymbol{K}_v \dot{\boldsymbol{e}} \xrightarrow[\dot{\boldsymbol{q}}_d = 0]{\text{Since } \boldsymbol{q}_d = \text{constant}} \boldsymbol{\tau} = \boldsymbol{K}_p \boldsymbol{e} - \boldsymbol{K}_v \dot{\boldsymbol{q}}$$

$$\boldsymbol{e} = \boldsymbol{q}_d - \boldsymbol{q}$$

$\boldsymbol{K}_p, \boldsymbol{K}_v \in \mathbb{R}^{n\times n}$ are symmetric positive definite matrices. If $\boldsymbol{K}_p = \mathrm{diag}\{K_{p,i}\}$, $\boldsymbol{K}_v = \mathrm{diag}\{K_{v,i}\}$, the controller is called PD Independent Joint Control.

This controller is the simplest (linear) controller that may be used to control robot manipulators.



PD Control

The closed-loop dynamic equation is derived as

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{g}(\boldsymbol{q}) = \boldsymbol{K}_p\boldsymbol{e} - \boldsymbol{K}_v\dot{\boldsymbol{q}}$$

$$\frac{d}{dt}\begin{bmatrix}\boldsymbol{e}\\ \dot{\boldsymbol{q}}\end{bmatrix} = \begin{bmatrix}-\dot{\boldsymbol{q}}\\ \boldsymbol{M}(\boldsymbol{q})^{-1}\left(\boldsymbol{K}_p\boldsymbol{e} - \boldsymbol{K}_v\dot{\boldsymbol{q}} - \boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} - \boldsymbol{g}(\boldsymbol{q})\right)\end{bmatrix} = \boldsymbol{f}(\boldsymbol{e},\dot{\boldsymbol{q}}) \quad (*) \qquad \boldsymbol{q} = \boldsymbol{q}_d - \boldsymbol{e}$$

The system is **autonomous** because $\boldsymbol{q}_d$ is constant.

Note: In general, this system may have several equilibrium points, and the origin $(\boldsymbol{e},\dot{\boldsymbol{q}}) = \boldsymbol{0} \in \mathbb{R}^{2n}$ is not necessarily an equilibrium point.

$$\boldsymbol{f}(\boldsymbol{e},\dot{\boldsymbol{q}}) = \boldsymbol{0} \quad \Rightarrow \quad \dot{\boldsymbol{q}} = \boldsymbol{0}, \qquad \boldsymbol{K}_p\boldsymbol{e} - \boldsymbol{g}(\boldsymbol{q}_d - \boldsymbol{e}) = \boldsymbol{0}$$

Note: If the manipulator model does not include the gravitational torques term $\boldsymbol{g}(\boldsymbol{q})$ (e.g., those which move only on the horizontal plane), then the only equilibrium is the origin $(\boldsymbol{e},\dot{\boldsymbol{q}}) = \boldsymbol{0} \in \mathbb{R}^{2n}$.

PD Control
(when $g(q) = 0$)

To study the stability of the equilibrium we can use Lyapunov's direct method and LaSalle's Theorem to show asymptotic stability of the origin $(\boldsymbol{e}, \dot{\boldsymbol{q}}) = \boldsymbol{0}$.

Consider a Lyapunov function candidate as $V(\boldsymbol{e}, \dot{\boldsymbol{q}}) = \frac{1}{2}\dot{\boldsymbol{q}}^T \boldsymbol{M}(\boldsymbol{q})\dot{\boldsymbol{q}} + \frac{1}{2}\boldsymbol{e}^T \boldsymbol{K}_p \boldsymbol{e} > 0$ (PD)

Kinetic energy of the arm

$\forall \boldsymbol{e}, \dot{\boldsymbol{q}} \neq \boldsymbol{0}$

$$\dot{V}(\boldsymbol{e}, \dot{\boldsymbol{q}}) = \dot{\boldsymbol{q}}^T \boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \frac{1}{2}\dot{\boldsymbol{q}}^T \dot{\boldsymbol{M}}(\boldsymbol{q})\dot{\boldsymbol{q}} + \boldsymbol{e}^T \boldsymbol{K}_p \dot{\boldsymbol{e}}$$

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} = \boldsymbol{K}_p \boldsymbol{e} - \boldsymbol{K}_v \dot{\boldsymbol{q}} - \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}}, \quad \dot{\boldsymbol{e}} = -\dot{\boldsymbol{q}} \qquad (*)$$

$$\dot{\boldsymbol{q}}^T \left[\frac{1}{2}\dot{\boldsymbol{M}} - \boldsymbol{C}\right]\dot{\boldsymbol{q}} = \boldsymbol{0}$$ (Property of dynamic model)

$$\dot{V}(\boldsymbol{e}, \dot{\boldsymbol{q}}) = -\dot{\boldsymbol{q}}^T \boldsymbol{K}_v \dot{\boldsymbol{q}} \leq 0 \quad \text{(NSD)}$$

Equilibrium Point Theorem → The origin $(\boldsymbol{e}, \dot{\boldsymbol{q}}) = \boldsymbol{0}$ is (globally) stable and the solutions $\boldsymbol{e}(t)$ and $\dot{\boldsymbol{q}}(t)$ are bounded.

PD Control
(when $g(q) = 0$)

Now, we use LaSalle (invariant set) theorem to analyze the global asymptotic stability of the origin.

$$R = \left\{(\boldsymbol{e}, \dot{\boldsymbol{q}}) \in \mathbb{R}^{2n} : \dot{V}(\boldsymbol{e}, \dot{\boldsymbol{q}}) = 0\right\}$$

$(\boldsymbol{e}, \dot{\boldsymbol{q}}) = \mathbf{0}$ is the largest invariant set in R

$\Rightarrow$ The origin $(\boldsymbol{e}, \dot{\boldsymbol{q}}) = \mathbf{0}$ is globally asymptotically stable for any initial condition $\boldsymbol{q}(0), \dot{\boldsymbol{q}}(0) \in \mathbb{R}^{\boldsymbol{n}}$:

$$\lim_{t \to \infty} \boldsymbol{e}(t) = \mathbf{0} \qquad \lim_{t \to \infty} \dot{\boldsymbol{q}}(t) = \mathbf{0}$$

$\Rightarrow$ Thus, the control objective is achieved.

Note: Friction at the joints may also affect the position error.

PD Control (when $\boldsymbol{g}(\boldsymbol{q}) \neq \mathbf{0}$)

The study of unicity of the equilibrium and boundedness of solutions for a control system under PD control when $\boldsymbol{g}(\boldsymbol{q}) \neq \mathbf{0}$ is somewhat more complex than when $\boldsymbol{g}(\boldsymbol{q}) = \mathbf{0}$.

For robots with only revolute joints, we can prove that

- For <u>any</u> $\boldsymbol{K}_p = \boldsymbol{K}_p^T > 0$, $\boldsymbol{K}_v = \boldsymbol{K}_v^T > 0$, it is guaranteed that $\boldsymbol{e}(t)$ and $\dot{\boldsymbol{q}}(t)$ are bounded for all initial conditions. Moreover, $\lim_{t\to\infty} \dot{\boldsymbol{q}}(t) = \mathbf{0}$ (it does not guarantee $\lim_{t\to\infty} \boldsymbol{q}(t) = \boldsymbol{q}_d$ or even $\lim_{t\to\infty} \boldsymbol{q}(t) = \text{constant}$).
- By choosing $\boldsymbol{K}_p$ sufficiently large, e.g., $\lambda_{\min}(\boldsymbol{K}_p) > n \cdot \left(\max_{i,j,q} \left|\frac{\partial g_i(\boldsymbol{q})}{\partial \boldsymbol{q}_j}\right|\right)$, then the closed-loop equation has a unique equilibrium (but not necessarily at origin).
- The error bound decreases, as $K_{v,i}$ become larger (in case $\boldsymbol{K}_v = \text{diag}\{K_{v,i}\}$), however, large $K_{v,i}$ can saturate the robot actuators.

$\Rightarrow$ Thus, the control objective <u>cannot</u> be achieved using PD control <u>unless</u> the desired position $\boldsymbol{q}_d$ is such that $\boldsymbol{g}(\boldsymbol{q}_d) = \mathbf{0}$ (i.e., the origin $(\boldsymbol{e}, \dot{\boldsymbol{q}}) = \mathbf{0}$ is an equilibrium).

Note: Friction at the joints may also affect the position error.

PD Control with Gravity Compensation

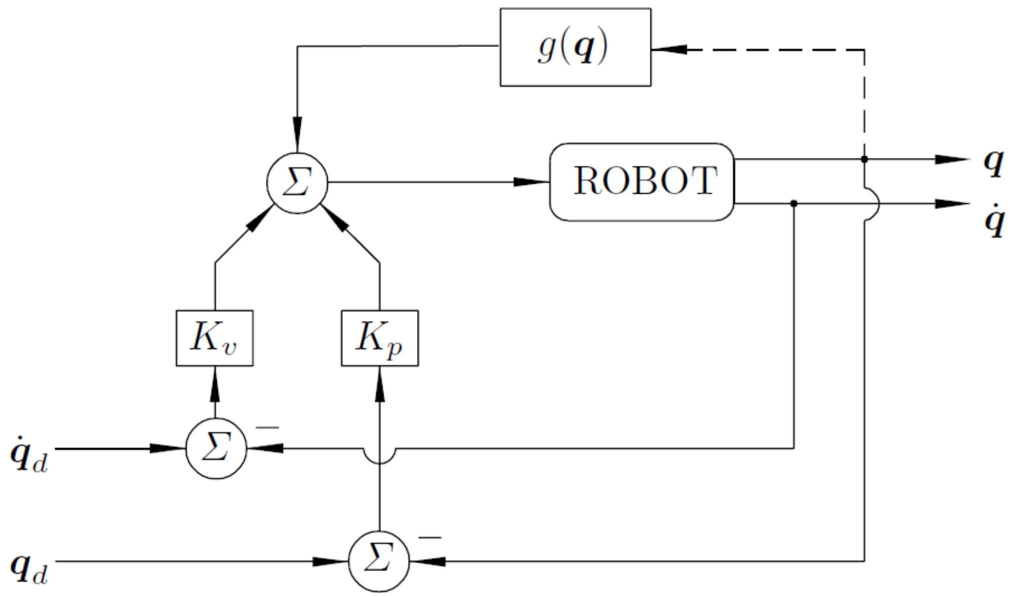
PD Control with Gravity Compensation

The PD control law with gravity compensation is given by

$$\boldsymbol{\tau} = \boldsymbol{K}_p\boldsymbol{e} + \boldsymbol{K}_v\dot{\boldsymbol{e}} + \boldsymbol{g}(\boldsymbol{q}) \xrightarrow[\dot{\boldsymbol{q}}_d = 0]{\text{Since } \boldsymbol{q}_d = \text{constant}} \boldsymbol{\tau} = \boldsymbol{K}_p\boldsymbol{e} - \boldsymbol{K}_v\dot{\boldsymbol{q}} + \boldsymbol{g}(\boldsymbol{q})$$

$\boldsymbol{K}_p, \boldsymbol{K}_v \in \mathbb{R}^{n\times n}$ are symmetric positive definite matrices.

$$\boldsymbol{e} = \boldsymbol{q}_d - \boldsymbol{q}$$



PD Control with Gravity Compensation

The closed-loop dynamic equation is derived as

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{g}(\boldsymbol{q}) = \boldsymbol{K}_p\boldsymbol{e} - \boldsymbol{K}_v\dot{\boldsymbol{q}} + \boldsymbol{g}(\boldsymbol{q})$$

$$\frac{d}{dt}\begin{bmatrix}\boldsymbol{e}\\ \dot{\boldsymbol{q}}\end{bmatrix} = \begin{bmatrix}-\dot{\boldsymbol{q}}\\ \boldsymbol{M}(\boldsymbol{q})^{-1}\left(\boldsymbol{K}_p\boldsymbol{e} - \boldsymbol{K}_v\dot{\boldsymbol{q}} - \boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}}\right)\end{bmatrix} = \boldsymbol{f}(\boldsymbol{e},\dot{\boldsymbol{q}}) \qquad \boldsymbol{q} = \boldsymbol{q}_d - \boldsymbol{e}$$

The system is **autonomous**, and the origin $(\boldsymbol{e},\dot{\boldsymbol{q}}) = \boldsymbol{0} \in \mathbb{R}^{2n}$ is the only equilibrium point.

Note: Using the same proof given for PD Control when $\boldsymbol{g}(\boldsymbol{q}) = \boldsymbol{0}$, we can show that the origin $(\boldsymbol{e},\dot{\boldsymbol{q}}) = \boldsymbol{0}$ is <u>globally asymptotically stable</u> for any initial condition $\boldsymbol{q}(0),\dot{\boldsymbol{q}}(0) \in \mathbb{R}^{\boldsymbol{n}}$:

$$\lim_{t\to\infty} \boldsymbol{e}(t) = \boldsymbol{0} \qquad \lim_{t\to\infty} \dot{\boldsymbol{q}}(t) = \boldsymbol{0}$$

$\Rightarrow$ Thus, the position control objective is achieved.

Note: Friction at the joints may also affect the position error.

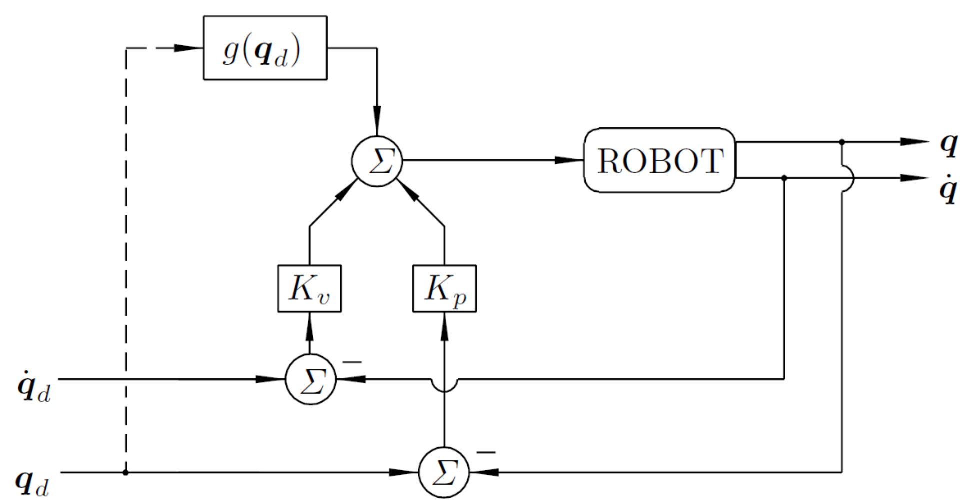
PD Control with Desired Gravity Compensation

PD Control with Desired Gravity Compensation

Implementation of the PD control with gravity compensation requires **on-line** computation of $\boldsymbol{g}(\boldsymbol{q})$. However, since the elements of $\boldsymbol{g}(\boldsymbol{q})$ involve trigonometric functions of $\boldsymbol{q}$, its real time computation take a longer time than the computation of the 'PD-part' of the control law, especially in high sampling frequency applications. A solution is using **PD Control with Desired Gravity Compensation** which requires only **off-line** computation of $\boldsymbol{g}(\boldsymbol{q}_d)$:

$$\boldsymbol{\tau} = \boldsymbol{K}_p\boldsymbol{e} + \boldsymbol{K}_v\dot{\boldsymbol{e}} + \boldsymbol{g}(\boldsymbol{q}_d) \qquad \boldsymbol{e} = \boldsymbol{q}_d - \boldsymbol{q}$$

$\boldsymbol{K}_p, \boldsymbol{K}_v \in \mathbb{R}^{n\times n}$ are symmetric positive definite matrices.



PD Control with Desired Gravity Compensation

The closed-loop dynamic equation is derived as

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{g}(\boldsymbol{q}) = \boldsymbol{K}_p\boldsymbol{e} - \boldsymbol{K}_v\dot{\boldsymbol{q}} + \boldsymbol{g}(\boldsymbol{q}_d)$$

$$\frac{d}{dt}\begin{bmatrix}\boldsymbol{e}\\ \dot{\boldsymbol{q}}\end{bmatrix} = \begin{bmatrix}-\dot{\boldsymbol{q}}\\ \boldsymbol{M}(\boldsymbol{q})^{-1}\left(\boldsymbol{K}_p\boldsymbol{e} - \boldsymbol{K}_v\dot{\boldsymbol{q}} - \boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} - \boldsymbol{g}(\boldsymbol{q}) + \boldsymbol{g}(\boldsymbol{q}_d)\right)\end{bmatrix} = \boldsymbol{f}(\boldsymbol{e},\dot{\boldsymbol{q}})$$

$$\boldsymbol{q} = \boldsymbol{q}_d - \boldsymbol{e}$$

The system is **autonomous** (since $\boldsymbol{q}_d$ is constant), and in general, may have multiple equilibria which the origin $(\boldsymbol{e},\dot{\boldsymbol{q}}) = \boldsymbol{0} \in \mathbb{R}^{2n}$ is always one of them:

$$\boldsymbol{f}(\boldsymbol{e},\dot{\boldsymbol{q}}) = \boldsymbol{0} \quad \Rightarrow \quad \dot{\boldsymbol{q}} = \boldsymbol{0}, \qquad \boldsymbol{K}_p\boldsymbol{e} - \boldsymbol{g}(\boldsymbol{q}_d - \boldsymbol{e}) + \boldsymbol{g}(\boldsymbol{q}_d) = \boldsymbol{0}$$

PD Control with Desired Gravity Compensation

For robots with only revolute joints, we can prove that

- For any $\boldsymbol{K}_p = \boldsymbol{K}_p^T > 0$, $\boldsymbol{K}_v = \boldsymbol{K}_v^T > 0$, it is guaranteed that $\boldsymbol{e}(t)$ and $\dot{\boldsymbol{q}}(t)$ are bounded for all initial conditions. Moreover, $\lim_{t\to\infty} \dot{\boldsymbol{q}}(t) = \mathbf{0}$ (it does not guarantee $\lim_{t\to\infty} \boldsymbol{q}(t) = \boldsymbol{q}_d$ or even $\lim_{t\to\infty} \boldsymbol{q}(t) = \text{constant}$).
- By choosing $\boldsymbol{K}_p$ sufficiently large, e.g., $\lambda_{\min}(\boldsymbol{K}_p) > n \cdot \left(\max_{i,j,q} \left| \frac{\partial g_i(\boldsymbol{q})}{\partial \boldsymbol{q}_j} \right| \right)$, then the closed-loop equation has a unique equilibrium at origin $(\boldsymbol{e}, \dot{\boldsymbol{q}}) = \mathbf{0}$ and it is globally asymptotically stable.

$$\lim_{t\to\infty} \boldsymbol{e}(t) = \mathbf{0} \qquad \lim_{t\to\infty} \dot{\boldsymbol{q}}(t) = \mathbf{0}$$

$\Rightarrow$ Thus, the position control objective is achieved.

Note: Friction at the joints may also affect the position error.

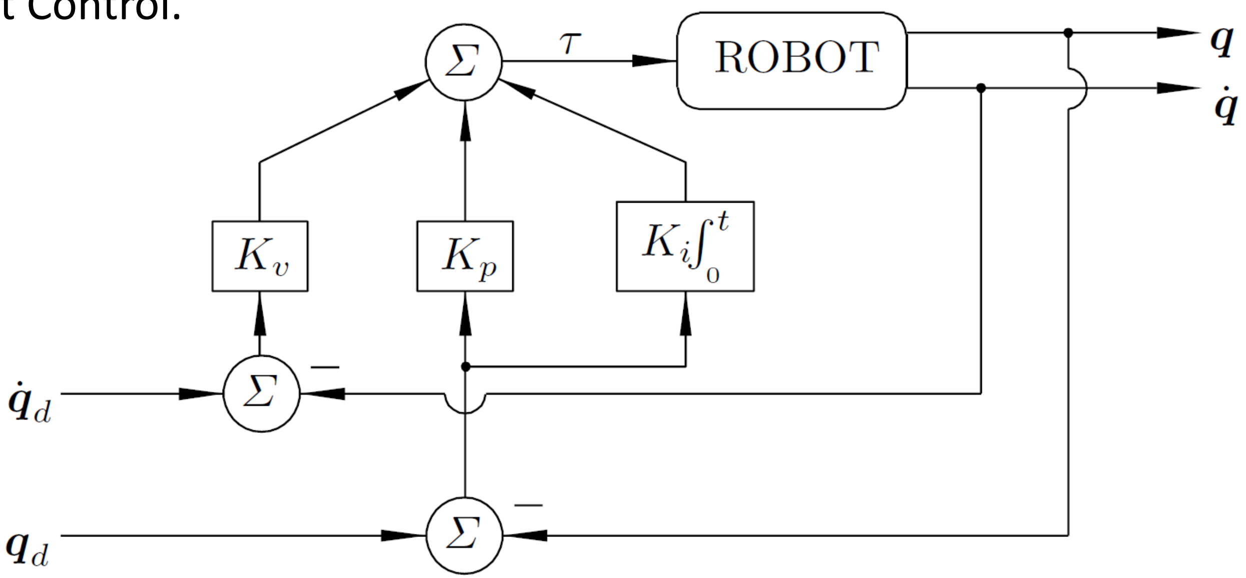
PID Control

PID Control

The PID (Proportional Integral Derivative) control law is given by

$$\boldsymbol{\tau} = \boldsymbol{K}_p \boldsymbol{e} + \boldsymbol{K}_v \dot{\boldsymbol{e}} + \boldsymbol{K}_i \int_0^t \boldsymbol{e}(\tau) d\tau \qquad \boldsymbol{e} = \boldsymbol{q}_d - \boldsymbol{q}$$

$\boldsymbol{K}_p, \boldsymbol{K}_v, \boldsymbol{K}_i \in \mathbb{R}^{n \times n}$ (position, velocity, and integral gains) are symmetric positive definite matrices. If $\boldsymbol{K}_p = \mathrm{diag}\{K_{p,i}\}$, $\boldsymbol{K}_v = \mathrm{diag}\{K_{v,i}\}$, $\boldsymbol{K}_i = \mathrm{diag}\{K_{i,i}\}$, the controller is called PID Independent Joint Control.



PID Control

The closed-loop dynamic equation is derived as

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{g}(\boldsymbol{q}) = \boldsymbol{K}_p\boldsymbol{e} + \boldsymbol{K}_v\dot{\boldsymbol{e}} + \boldsymbol{K}_i \overbrace{\int_0^t \boldsymbol{e}(\tau)d\tau}^{\boldsymbol{\xi}}$$

$$\Rightarrow \begin{cases} \boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{g}(\boldsymbol{q}) = \boldsymbol{K}_p\boldsymbol{e} + \boldsymbol{K}_v\dot{\boldsymbol{e}} + \boldsymbol{K}_i\boldsymbol{\xi} \\ \dot{\boldsymbol{\xi}} = \boldsymbol{e} \end{cases}$$

$$\boldsymbol{q} = \boldsymbol{q}_d - \boldsymbol{e}$$

$$\frac{d}{dt}\begin{bmatrix} \boldsymbol{\xi} \\ \boldsymbol{e} \\ \dot{\boldsymbol{q}} \end{bmatrix} = \begin{bmatrix} \boldsymbol{e} \\ -\dot{\boldsymbol{q}} \\ \boldsymbol{M}(\boldsymbol{q})^{-1}\left(\boldsymbol{K}_p\boldsymbol{e} - \boldsymbol{K}_v\dot{\boldsymbol{q}} + \boldsymbol{K}_i\boldsymbol{\xi} - \boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} - \boldsymbol{g}(\boldsymbol{q})\right) \end{bmatrix} \xrightarrow[\text{point}]{\text{equilibrium}} \begin{bmatrix} \boldsymbol{K}_i^{-1}\boldsymbol{g}(\boldsymbol{q}_d) \\ \boldsymbol{0} \\ \boldsymbol{0} \end{bmatrix}$$

Translating this equilibrium point to the origin via a suitable change of variable:

$$\boldsymbol{z} = \boldsymbol{\xi} - \boldsymbol{K}_i^{-1}\boldsymbol{g}(\boldsymbol{q}_d)$$

$$\frac{d}{dt}\begin{bmatrix} \boldsymbol{z} \\ \boldsymbol{e} \\ \dot{\boldsymbol{q}} \end{bmatrix} = \begin{bmatrix} \boldsymbol{e} \\ -\dot{\boldsymbol{q}} \\ \boldsymbol{M}(\boldsymbol{q})^{-1}\left(\boldsymbol{K}_p\boldsymbol{e} - \boldsymbol{K}_v\dot{\boldsymbol{q}} + \boldsymbol{K}_i\boldsymbol{z} + \boldsymbol{g}(\boldsymbol{q}_d) - \boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} - \boldsymbol{g}(\boldsymbol{q})\right) \end{bmatrix}$$

The system is **autonomous,** and its unique equilibrium is the origin $(\boldsymbol{z},\boldsymbol{e},\dot{\boldsymbol{q}}) = \boldsymbol{0} \in \mathbb{R}^{3n}$.

PID Control: Tuning Method

For robots with only revolute joints, we can prove that the symmetric positive definite matrices $\boldsymbol{K}_p, \boldsymbol{K}_i, \boldsymbol{K}_v$ which satisfy the following relations can only guarantee achievement of the position control objective by making the origin $(\boldsymbol{z}, \boldsymbol{e}, \dot{\boldsymbol{q}}) = \boldsymbol{0}$ locally asymptotically stable (i.e., if $\boldsymbol{e}(t)$, $\dot{\boldsymbol{q}}(t)$ are "sufficiently small", $\lim_{t \to \infty} \boldsymbol{e}(t) = \boldsymbol{0}$).

$$\lambda_{\max}\{\boldsymbol{K}_i\} \geq \lambda_{\min}\{\boldsymbol{K}_i\} > 0$$
$$\lambda_{\max}\{\boldsymbol{K}_p\} \geq \lambda_{\min}\{\boldsymbol{K}_p\} > n \cdot k_g$$
$$\lambda_{\max}\{\boldsymbol{K}_v\} \geq \lambda_{\min}\{\boldsymbol{K}_v\} > \frac{\lambda_{\max}\{\boldsymbol{K}_i\}}{\lambda_{\min}\{\boldsymbol{K}_p\} - k_g} \cdot \frac{\lambda_{\max}^2(\boldsymbol{M}(\boldsymbol{q}))}{\lambda_{\min}(\boldsymbol{M}(\boldsymbol{q}))}$$

$$k_g = \max_{i,j,q} \left| \frac{\partial g_i(\boldsymbol{q})}{\partial \boldsymbol{q}_j} \right|$$

Note: A system with $\boldsymbol{K}_p, \boldsymbol{K}_i, \boldsymbol{K}_v$ parameters which satisfy these relations does not necessarily achieve a proper settling time. It is possible to find a set of the symmetric PD matrices $\boldsymbol{K}_p, \boldsymbol{K}_i, \boldsymbol{K}_v$ which achieve a small settling time, while violating these relations.