

Ch8: Position Control

Robot Control Systems

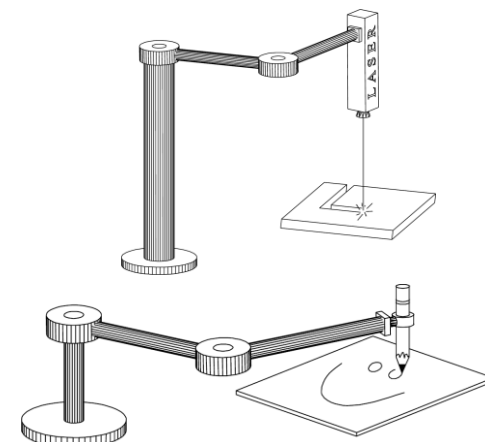
Robot Controllers

A robot controller generates the required torques/forces τ to perform a desired task ($\theta_d(t) \in \mathbb{R}^n$ in joint space or $x_d(t) \in \mathbb{R}^m$ or $T_d(t) \in SE(3)$ in task space) while satisfying given transient and steady-state requirements.

- Joint Space Control: Independent Joint Control and Model-Dependent Joint Control
- Task Space Control

Depending on the task and the robot environment, there are different feedback control strategies including

- Position Control (regulation or set-point control)
 - Motion Control (tracking control)
 - Force Control
 - Hybrid Motion–Force Control
 - Impedance Control
- when moving in free space**
- when making contact with environment**



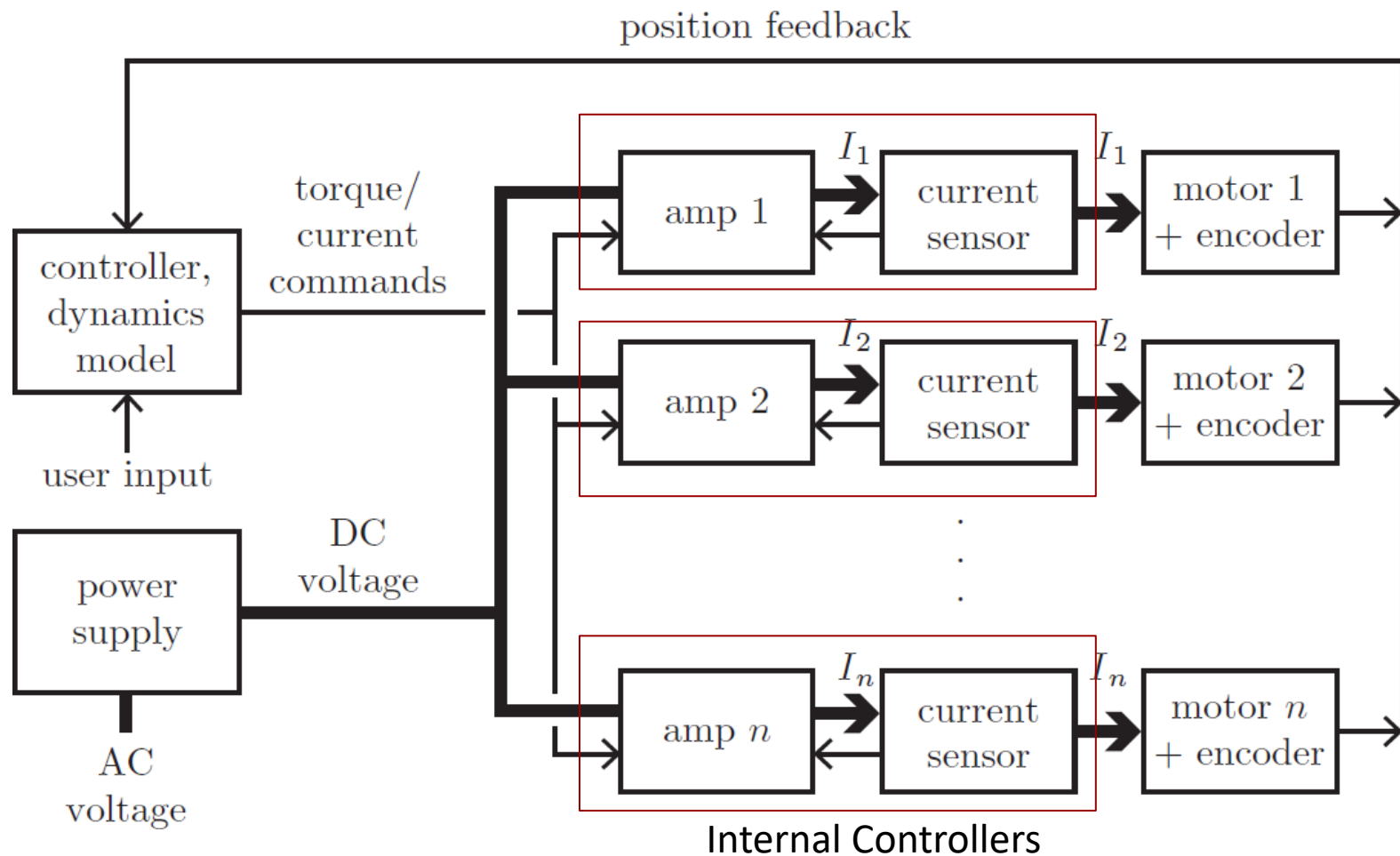
* A fundamental constraint imposed by the mechanics:

We cannot independently control the motion and force in the same direction.

Examples of Desired Tasks

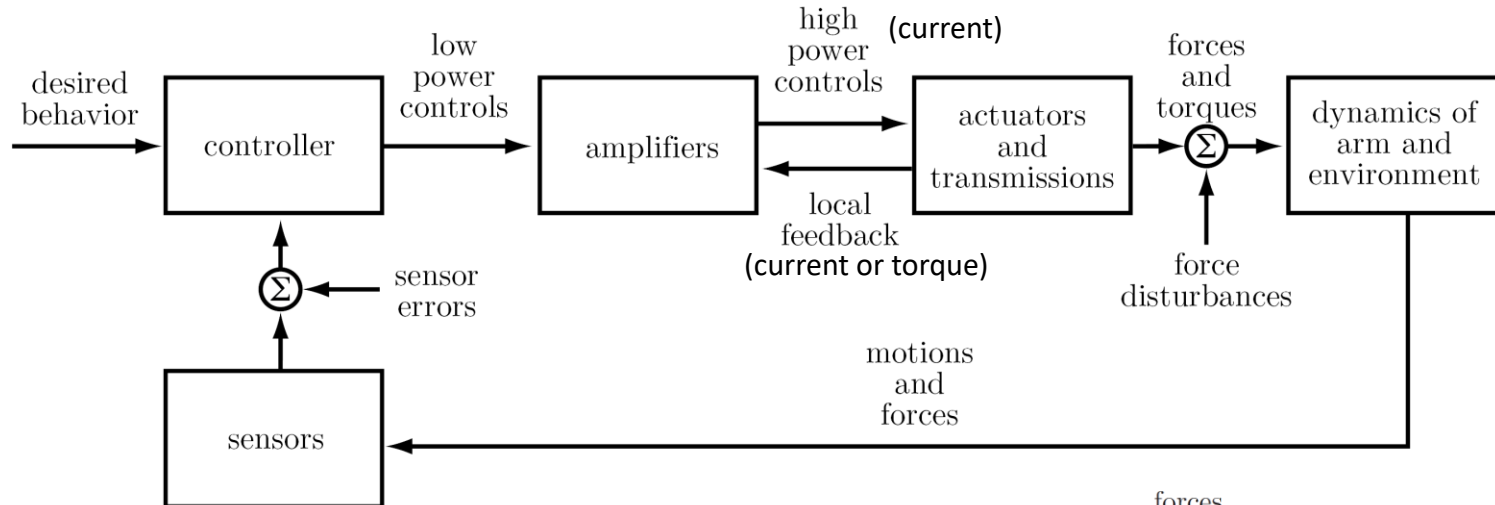
- Pick and place
- Spray painting or laser cutting
- Folding laundry
- Polishing with a polishing wheel
- Back massage
- Erasing a whiteboard
- Shaking hands with a human
- Inserting a peg in a hole
- Acting as a haptic display

A Typical Robot Driven by DC Electric Motors

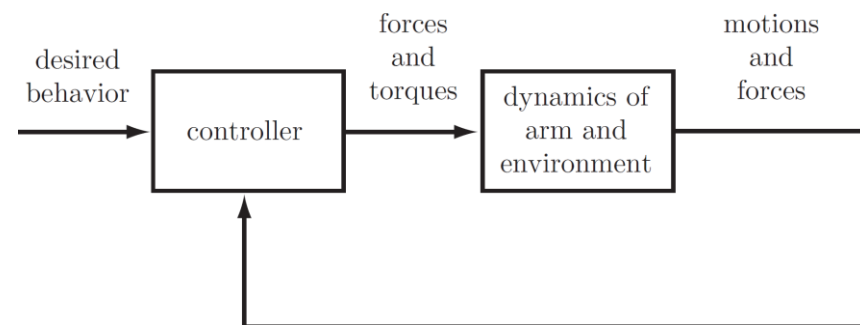


A Typical Robot Control System

Feedback control (or closed-loop control) uses position, velocity, and/or force sensors to measure the actual behavior of the robot, compares it with the desired behavior, and modulates the control signals sent to the actuators.



A simplified model with ideal sensors and ideal behavior of the amplifier and actuator blocks.



Closed-loop Dynamic Equation

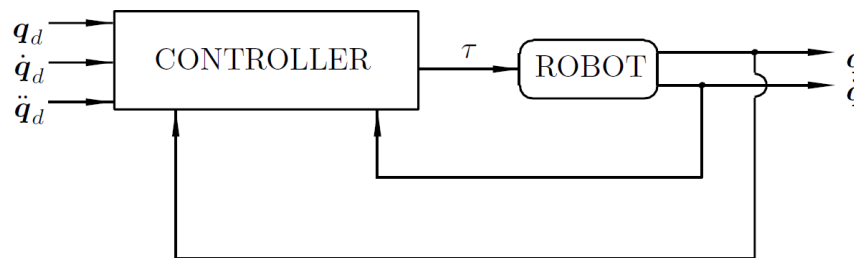
Consider the dynamic model of an n -DOF open-chain manipulator with no friction at the joints and no external force at the end-effector.

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

(state-space form)

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

$$\tau = \tau_c(q, \dot{q}, q_d, \dot{q}_d, \ddot{q}_d, M(q), C(q, \dot{q}), g(q))$$



Note: For practical purposes, it is desirable that the controller τ_c does not depend on the joint acceleration \ddot{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

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

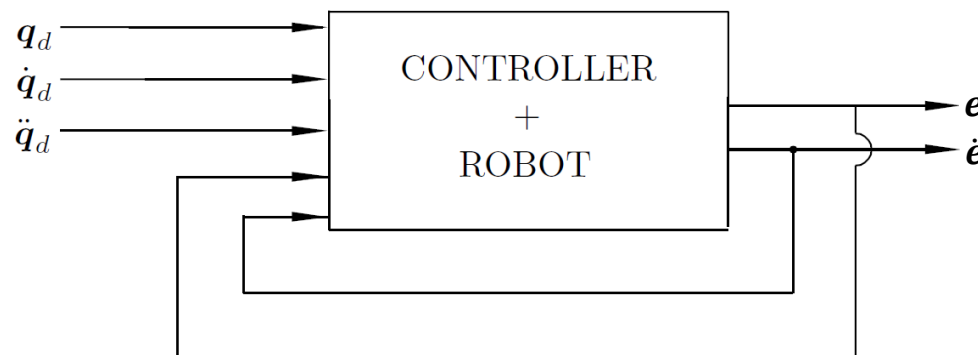
or in the state-space form as $\frac{d}{dt} \begin{bmatrix} \mathbf{q}_d - \mathbf{q} \\ \dot{\mathbf{q}}_d - \dot{\mathbf{q}} \end{bmatrix} = \mathbf{f}(\mathbf{q}, \dot{\mathbf{q}}, \mathbf{q}_d, \dot{\mathbf{q}}_d, \ddot{\mathbf{q}}_d, \mathbf{M}(\mathbf{q}), \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}), \mathbf{g}(\mathbf{q}))$

$$\begin{aligned} \mathbf{e} &= \mathbf{q}_d - \mathbf{q} \in \mathbb{R}^n, \\ \dot{\mathbf{e}} &= \dot{\mathbf{q}}_d - \dot{\mathbf{q}} \in \mathbb{R}^n, \end{aligned}$$

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

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

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

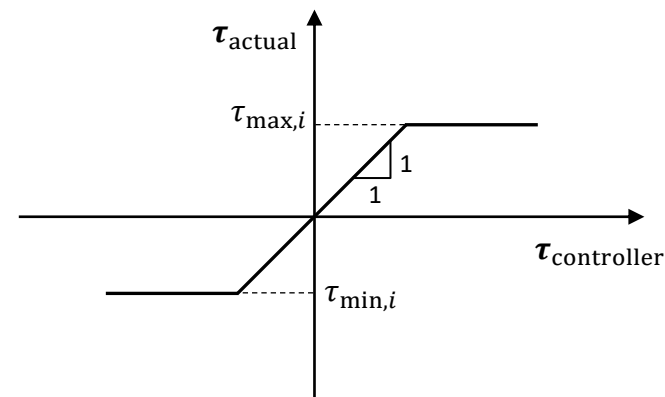
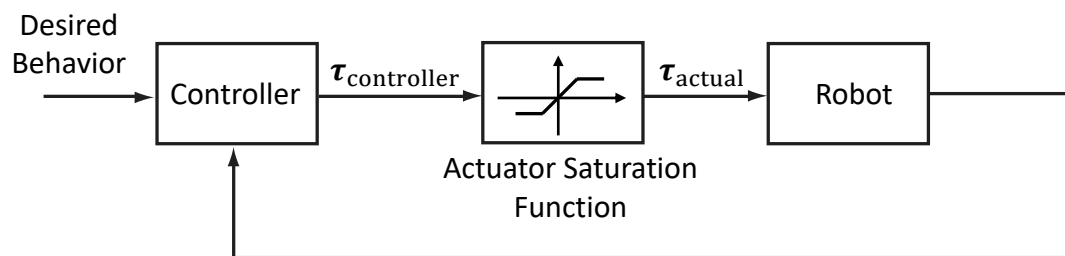


Actuator Saturation

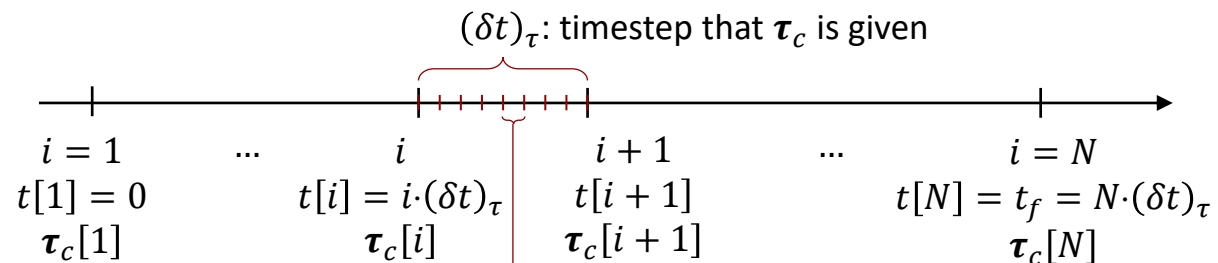
In some controllers, choosing large values for the control parameters causes a large (initial) control torque which is beyond the robot actuators capacity which are limited by maximum and minimum allowable values τ_{\max} , τ_{\min} . Therefore, the control parameters should be chosen properly.

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

$$\tau_{\text{actual}} = \text{sat}(\tau_{\text{controller}})$$



Pseudocode for Controllers



```

Set  $\theta[1] = \theta(0)$ ,  $\dot{\theta}[1] = \dot{\theta}(0)$ ,  $e_{\text{int}} = 0$ 
Set  $\bar{\theta} = \theta[1]$ ,  $\dot{\bar{\theta}} = \dot{\theta}[1]$ 
For  $i = 1$  to  $N - 1$ 
     $[\theta_d[i], \dot{\theta}_d[i]] = \text{trajectory}(t[i] = i \cdot (\delta t)_\tau, \dots)$ 
     $\tau_c[i] = \text{controller}(\bar{\theta}, \dot{\bar{\theta}}, \theta_d[i], \dot{\theta}_d[i], \ddot{\theta}_d[i], e_{\text{int}}, \dots)$ 
     $\tau_c[i] = \text{sat}(\tau_c[i])$ 
    For  $j = 1$  to  $n_{\text{res}}$ 
         $\ddot{\theta} = \text{ForwardDynamics}(\bar{\theta}, \dot{\bar{\theta}}, \tau_c[i], \mathcal{F}_{\text{tip}}[i])$ 
         $\bar{\theta} = \bar{\theta} + \dot{\bar{\theta}} \cdot (\delta t)_\theta$ 
         $\dot{\bar{\theta}} = \dot{\bar{\theta}} + \ddot{\theta} \cdot (\delta t)_\theta$ 
    end
     $e_{\text{int}} = e_{\text{int}} + (\delta t)_\tau (\theta_d[i] - \bar{\theta})$  % error integral
     $\theta[i+1] = \bar{\theta}$ 
     $\dot{\theta}[i+1] = \dot{\bar{\theta}}$ 
end
    
```

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

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

Position Control

Position Control Objective

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

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

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

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

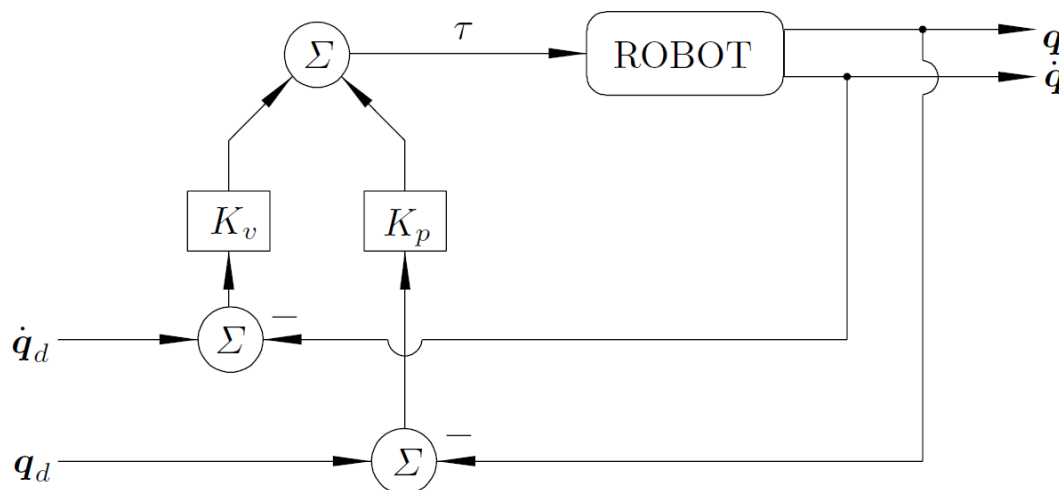
(or P Control Plus Velocity Feedback)

The PD (Proportional Derivative) control law is given by

$$\tau = K_p e + K_v \dot{e} \quad \xrightarrow[\dot{q}_d = 0]{\text{Since } q_d = \text{constant}} \quad \tau = K_p e - K_v \dot{q} \quad e = q_d - q$$

$K_p, K_v \in \mathbb{R}^{n \times n}$ are symmetric positive definite matrices. If $K_p = \text{diag}\{K_{p,i}\}$, $K_v = \text{diag}\{K_{v,i}\}$, 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

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

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

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

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

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

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

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 $(e, \dot{q}) = 0$.

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

$$\dot{V}(e, \dot{q}) = \dot{q}^T M(q) \ddot{q} + \frac{1}{2} \dot{q}^T \dot{M}(q) \dot{q} + e^T K_p \dot{e} \quad \forall e, \dot{q} \neq 0$$

Kinetic energy of the arm

$$M(q) \ddot{q} = K_p e - K_v \dot{q} - C(q, \dot{q}) \dot{q}, \quad e = -\dot{q} \quad (*)$$

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

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

Equilibrium Point Theorem →

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

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

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

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

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

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

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

\Rightarrow Thus, the control objective is achieved.

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

PD Control (when $g(q) \neq 0$)

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

For robots with only revolute joints, we can prove that

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

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

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

PD Control with Gravity Compensation

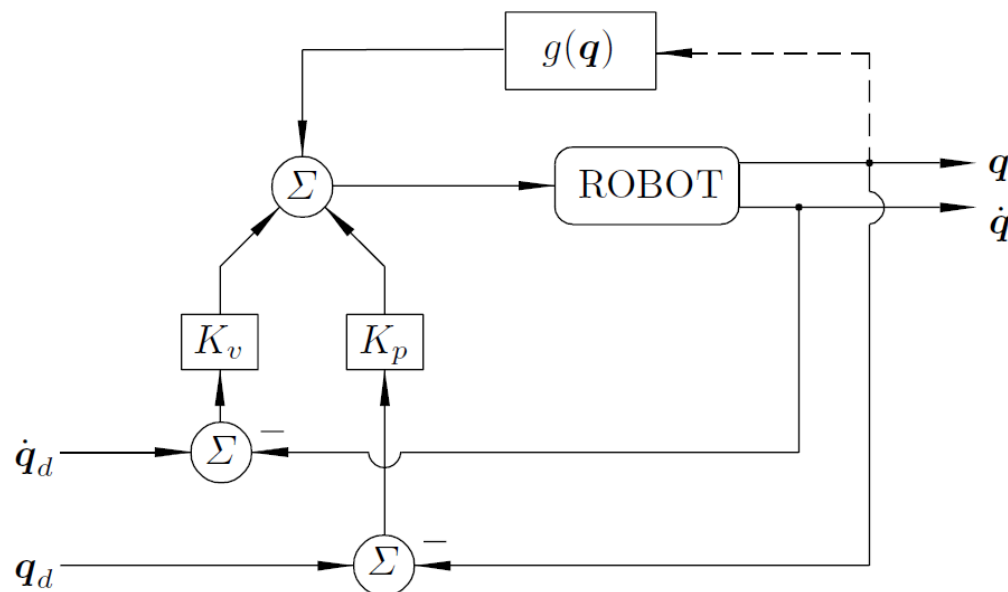
PD Control with Gravity Compensation

The PD control law with gravity compensation is given by

$$\tau = K_p e + K_v \dot{e} + g(q) \xrightarrow[\dot{q}_d = 0]{\text{Since } q_d = \text{constant}} \tau = K_p e - K_v \dot{q} + g(q)$$

$K_p, K_v \in \mathbb{R}^{n \times n}$ are symmetric positive definite matrices.

$$e = q_d - q$$



PD Control with Gravity Compensation

The closed-loop dynamic equation is derived as

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

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

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

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

$$\lim_{t \rightarrow \infty} \mathbf{e}(t) = \mathbf{0} \quad \lim_{t \rightarrow \infty} \dot{\mathbf{q}}(t) = \mathbf{0}$$

\Rightarrow Thus, the position control objective is achieved.

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

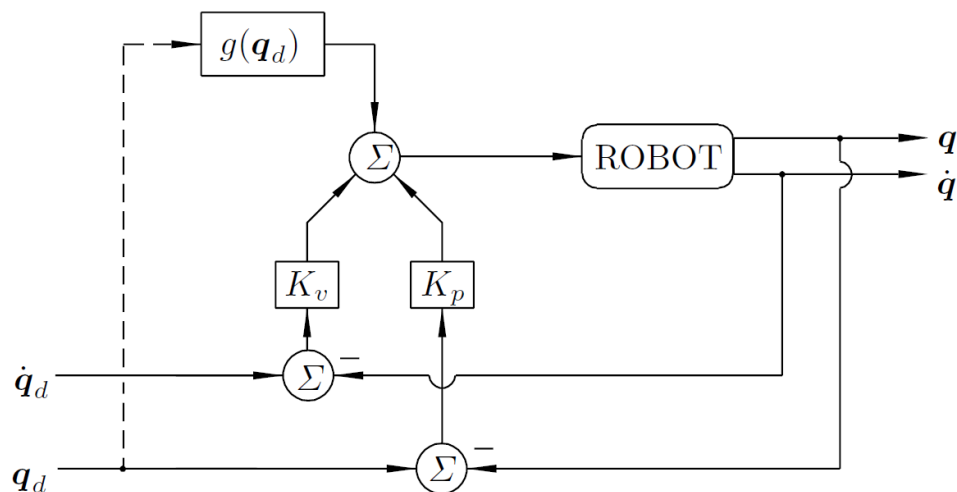
PD Control with Desired Gravity Compensation

PD Control with Desired Gravity Compensation

Implementation of the PD control with gravity compensation requires **on-line** computation of $\mathbf{g}(\mathbf{q})$. However, since the elements of $\mathbf{g}(\mathbf{q})$ involve trigonometric functions of \mathbf{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 $\mathbf{g}(\mathbf{q}_d)$:

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

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



PD Control with Desired Gravity Compensation

The closed-loop dynamic equation is derived as

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

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

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

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

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

PD Control with Desired Gravity Compensation

For robots with only revolute joints, we can prove that

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

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

\Rightarrow Thus, the position control objective is achieved.

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

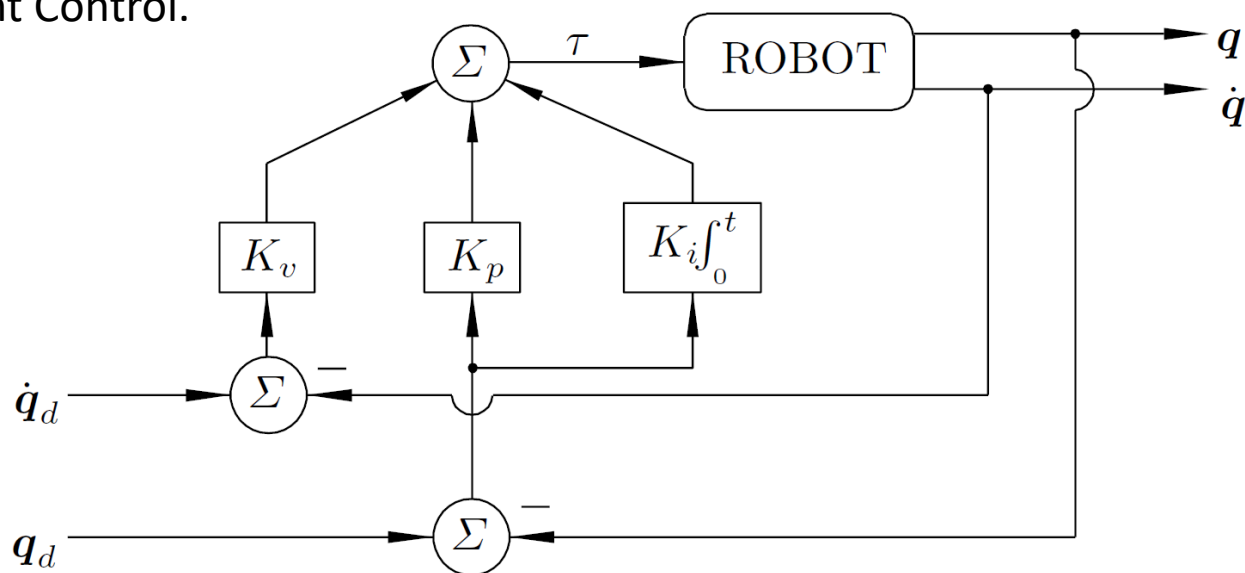
PID Control

PID Control

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

$$\tau = K_p e + K_v \dot{e} + K_i \int_0^t e(\tau) d\tau \quad e = q_d - q$$

$K_p, K_v, K_i \in \mathbb{R}^{n \times n}$ (position, velocity, and integral gains) are symmetric positive definite matrices. If $K_p = \text{diag}\{K_{p,i}\}$, $K_v = \text{diag}\{K_{v,i}\}$, $K_i = \text{diag}\{K_{i,i}\}$, the controller is called PID Independent Joint Control.



PID Control

The closed-loop dynamic equation is derived as

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + g(q) = K_p e + K_v \dot{e} + K_i \int_0^t e(\tau) d\tau$$

$$\Rightarrow \begin{cases} M(q)\ddot{q} + C(q, \dot{q})\dot{q} + g(q) = K_p e + K_v \dot{e} + K_i \xi \\ \dot{\xi} = e \end{cases} \quad q = q_d - e$$

$$\frac{d}{dt} \begin{bmatrix} \xi \\ e \\ \dot{q} \end{bmatrix} = \begin{bmatrix} e \\ -\dot{q} \\ M(q)^{-1} (K_p e - K_v \dot{q} + K_i \xi - C(q, \dot{q})\dot{q} - g(q)) \end{bmatrix} \xrightarrow[\text{point}]{\text{equilibrium}} \begin{bmatrix} K_i^{-1} g(q_d) \\ 0 \\ 0 \end{bmatrix}$$

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

$$z = \xi - K_i^{-1} g(q_d)$$

$$\frac{d}{dt} \begin{bmatrix} z \\ e \\ \dot{q} \end{bmatrix} = \begin{bmatrix} e \\ -\dot{q} \\ M(q)^{-1} (K_p e - K_v \dot{q} + K_i z + g(q_d) - C(q, \dot{q})\dot{q} - g(q)) \end{bmatrix}$$

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

PID Control: Tuning Method

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

$$\lambda_{\max}\{\mathbf{K}_i\} \geq \lambda_{\min}\{\mathbf{K}_i\} > 0$$

$$\lambda_{\max}\{\mathbf{K}_p\} \geq \lambda_{\min}\{\mathbf{K}_p\} > n \cdot \left(\max_{i,j,q} \left| \frac{\partial g_i(\mathbf{q})}{\partial \mathbf{q}_j} \right| \right)$$

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

Note: A system with \mathbf{K}_p , \mathbf{K}_i , \mathbf{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 \mathbf{K}_p , \mathbf{K}_i , \mathbf{K}_v which achieve a small settling time, while violating these relations.