# Lesson 21:

I. PD joint control (cont'd) (SHV 9.2)

II. Inverse Dynamics control (IDC)

A. Joint Space IDC (SHV 9.3.1)

B. Task Space IDC (SHV 9.3.2)

---

## I. PD Joint Control

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = u \qquad (1)$$

- What happens if we include gravity? i.e., $G(q) \neq 0$

  New eq. pt: $\bar{x}$ s.t. $G(q) = -K_p \tilde{q}$

  corresponds to point where control torques balance the gravitational torques, resulting in steady-state error.

  * Can reduce steady-state error by increasing $K_p$, but can never eliminate it w/ PD control.
  - Also possible to apply Lyapunov analysis to the eq. pt. to prove stability.
  - Modified PD control w/ gravity compensation term:

$$u = -K_p\tilde{q} - K_d\dot{q} + G(q) \implies \tilde{q} \to 0, \quad t \to \infty$$

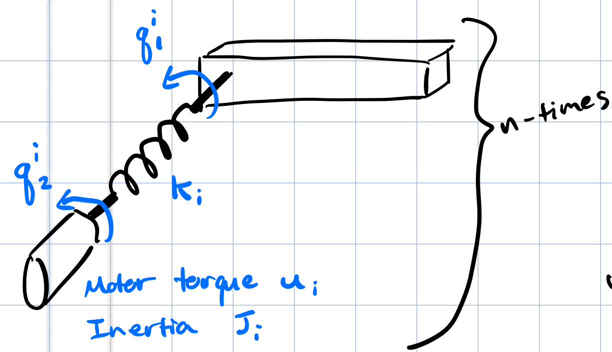
- What happens if we have flexible joints? (assume no gravity)

  Consider $q_1 \in \mathbb{R}^n$ as robot joint config.

  and $q_2 \in \mathbb{R}^n$ as vector of motor shaft angles, i.e.,

$$q = \begin{bmatrix} q_1 \\ q_2 \end{bmatrix} \in \mathbb{R}^{2n} \quad \text{and} \quad x = \begin{bmatrix} q_1 \\ q_2 \\ \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} \in \mathbb{R}^{4n}$$

Note that $(q_1 - q_2)$ is vector of elastic joint deflections.




$$KE = \frac{1}{2}\dot{q}_1^T M(q_1)\dot{q}_1 + \frac{1}{2}\dot{q}_2^T J \dot{q}_2$$

$$PE = \cancel{P(q_1)}^{\,0} + \frac{1}{2}(q_1 - q_2)^T K (q_1 - q_2)$$

where $J > 0$ is diag. mtx. of motor inertias

$K > 0$ is diag. mtx. of stiffnesses.

E-L: $$M(q_1)\ddot{q}_1 + C(q_1, \dot{q}_1)\dot{q}_1 + G(q_1) + K\cdot(q_1 - q_2) = 0 \quad \text{(links)}$$

$$J\ddot{q}_2 + K\cdot(q_2 - q_1) = u \quad \text{(motors)}$$

↑ 2n equations of motion.

Define PD control law:

Let $u = -K_p \tilde{q}_2 - K_d \dot{q}_2$, where $\tilde{q}_2 = q_2 - q_d$

for a const. set-point $q_d \in \mathbb{R}^n$, i.e. using motor feedback (collocated)

* <u>Note</u>: Collocated feedback is more stabilizing than non-collocated feedback (joint-angle feedback)

If $G(q_1) = 0$, then

$$V(x) = \frac{1}{2}\dot{q}_1^T M(q_1)\dot{q}_1 + \frac{1}{2}\dot{q}_2^T J\dot{q}_2 + \frac{1}{2}(q_1-q_2)^T K(q_1-q_2) + \underbrace{\frac{1}{2}\tilde{q}_2^T K_p \tilde{q}_2}_{\text{virtual spring energy from controller.}}$$

SHV Prob. 9-3 $\Rightarrow (q_1, q_2, \dot{q}_1, \dot{q}_2) \to (q_d, q_d, 0, 0)$

is G.A.S. by LaSalle.

because $V(x)$ is radially unbounded.

- Not easy with gravity, need to use "feedback linearization"

## II. Inverse Dynamics Control

- Special case of FB linearization w/ full actuation.
- Model-based (need to identify model first)
- Useful for dynamic behaviors like trajectory tracking or energy shaping.
- Joint-space or task-space.

### A. Joint Space IDC

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = u \quad \sim (1)$$

- What if we wanted to command joint accelerations?

$$\ddot{q} = a_q(t)$$

- Invert the torque-to-acceleration dynamics:

$$\cancel{M(q)}\,\underline{\ddot{q}} + \cancel{C(q,\dot{q})\dot{q}} + \cancel{G(q)} = \boxed{u := \cancel{M(q)}\,\underline{a_q(t)} + \cancel{C(q,\dot{q})\dot{q}} + \cancel{G(q)}}$$

(inner loop / torque controller)

$$\Rightarrow \ddot{q} = a_q \longleftarrow \text{decoupled joint dynamics (double integrator)}$$

Now we can define an outer-loop / acceleration controller to track a desired joint trajectory $q^d(t)$ by inserting linear dynamics into the double integrator.

Assume that $q^d(t)$ is twice differentiable.

Outer-loop / accel. controller:

$$a_q := \underbrace{\ddot{q}^d(t)}_{\text{Feedforward}} \underbrace{- K_p \underbrace{(q - q^d(t))}_{\tilde{q}} - K_d \underbrace{(\dot{q} - \dot{q}^d(t))}_{\dot{\tilde{q}}}}_{\text{Feedback}}$$

$$\ddot{q} = a_q \Rightarrow \ddot{q} = \ddot{q}^d(t) - K_p(q - q^d(t)) - K_d(\dot{q} - \dot{q}^d(t))$$

$$\ddot{\tilde{q}} + K_d \dot{\tilde{q}} + K_p \tilde{q} = 0$$

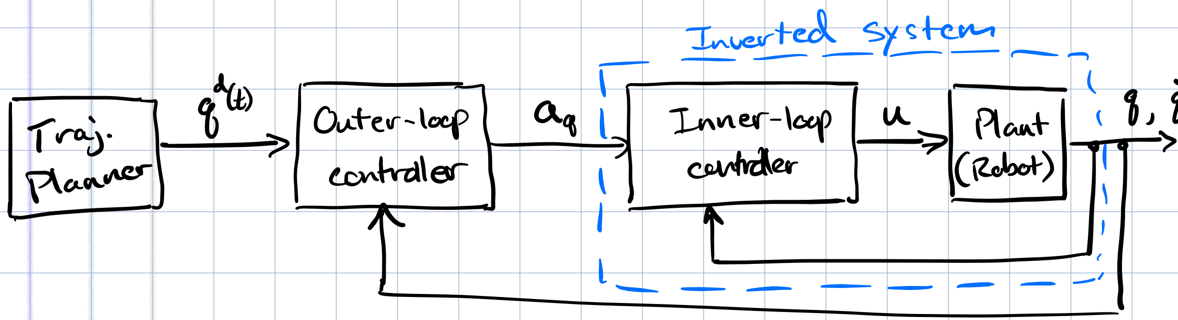
* Closed-loop error dynamics are linear (hence "FB linearization")

- Will be stable by choosing $K_p > 0$, $K_d > 0$

$$\Rightarrow (\tilde{q}, \dot{\tilde{q}}) = (0,0) \text{ is G.E.S} \Rightarrow q(t) \xrightarrow[t \to \infty]{} q^d(t)$$

Final control law:

$$u = M(q)\Big[\underbrace{\ddot{q}^d - K_p\tilde{q} - K_d\dot{\tilde{q}}}_{a_q}\Big] + C(q,\dot{q})\dot{q} + G(q)$$




- Outer loop could alternatively insert desired dynamics instead of tracking a reference trajectory:

$$a_q := \bar{M}(q)^{-1}\left[v - \bar{C}(q,\dot{q})\dot{q} - \bar{G}(q)\right]$$

where $\bar{M} \in \mathbb{R}^{n\times n}$, $\bar{C} \in \mathbb{R}^{n\times n}$, $\bar{G} \in \mathbb{R}^n$ are desired dynamics terms.

and $v$ is new input in the "shaped" dynamics.

$$\ddot{q} = a_q \implies \bar{M}\ddot{q} + \bar{C}\dot{q} + \bar{G} = v$$

This is called "energy shaping"