Assignment 1 - Control Design

Saturday, February 3, 2024 2:27 PM

Remark: since we are told $q_2, \dots, q_5$ are directly actuated & not q_2, then

$$B(q) = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \tau = B\tau \quad \begin{matrix} \nearrow B \in \mathbb{R}^{5\times 4} \\ \rightarrow \tau \in \mathbb{R}^4 \end{matrix}$$

w/ τ_i being the actuat: for joint q_{i+1}.

GOAL: design τ

1. We need to pick our gains (K_p, K_d).

Input: poles $= \{-20, -20\}$

Analytically, we can solve for K_p, K_d.

$\lambda^2 + K_d\lambda + K_p = (\lambda - 20)^2 = \lambda^2 - 2\cdot 20\lambda + 20^2$

$\therefore \begin{bmatrix} K_p \\ K_d \end{bmatrix} = \begin{bmatrix} 400 \\ -40 \end{bmatrix}$

Alternatīvly, we can use MATLAB's acker(·) funct:

$[K_p \quad K_d] = \text{acker}\left(\begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \end{bmatrix}, [-20 \quad -20]\right)$

2. We want to show that $C = h^{-1}(\{0\})$ defn̄s a VHC.

First, we are gīvn the error

$$e = h(q) = \begin{bmatrix} q_2 - q_2^{ref} \\ q_3 - q_3^{ref} \\ q_4 - q_4^{ref} \\ q_5 - q_5^{ref} \end{bmatrix} = [0_{4\times 1} \quad I_{4\times 4}]\, q - q_{ref} = Hq - q_{ref}$$

Then we are told to compute $B^\perp D(\sigma(\theta))\sigma'(\theta)$

① $\sigma(\theta) = \begin{bmatrix} \theta \\ q^{ref} \end{bmatrix} \in \mathbb{R}^5 \Rightarrow \sigma'(\theta) = \begin{bmatrix} 1 \\ 0_{4\times 1} \end{bmatrix}$

② $B^\perp := [1 \quad 0_{1\times 4}]$, so notice that $(\sigma'(\theta))^T = B^\perp$

③ Since $D(q)$ is assumed positive definite $\forall q$, we have $v^T D(q) v > 0 \quad \forall v \neq 0$. In particular, letting $v = \sigma'(\theta)$ & $q = \sigma(\theta)$, we have:

$$0 < (\sigma'(\theta))^T D(\sigma(\theta))\sigma'(\theta) = B^\perp D(\sigma(\theta))\sigma'(\theta)$$

& so $B^\perp D(\sigma(\theta))\sigma'(\theta)$ is non-zero $\forall \theta \in [\mathbb{R}]_{2\pi}$.

Moreover, since $B^\perp = [1 \quad 0_{1\times 4}]$, we have that:

$$B^\perp D(\sigma(\theta))\sigma'(\theta) = D_{11}(\sigma(\theta)).$$

In code, we are given the inputs:

D ~ the symbolic version of the matrix $D(q)$

$B^\perp = [1 \quad 0_{1\times 4}]$ ~ the left annihilator of B (define this in your code)

q_{ref} ~ "data.qref"

Then in code we compute

```
sym θ real
σ = [θ ; qref]
σ' = (B⊥)^T
val = B⊥ * subs(D, σ) * σ' ;   ∀ θ ∈ [R]2π
%. check val is non-zero. If indeed val does not depend on θ
%. (it shouldn't based on the handout) then we should be able to
%. run the command:
val_num = double(val)
%. check val_num ≠ 0 & provide an appropriate print stmt.
```

3. Defn̄ the controller

From the "data" structure we need:

H ~ "data.H"

$D(\cdot)$ ~ "data.D(·)"

B ~ "data.B"

$C(\cdot)$ ~ "data.C(·)"

$\nabla_q P(\cdot)$ ~ "data.G(·)"

K_p ~ "data.Kp"

K_d ~ "data.Kd"

q_{ref} ~ "data.qref"

So that the control input is computed as

$$\tau = \left(H(D(q))^{-1}B\right)^{-1}\left[H(D(q))^{-1}\left(C(q,\dot q)\dot q + \nabla_q P(q)\right) - K_p \sin(Hq - q^{ref}) - K_d H\dot q\right]$$