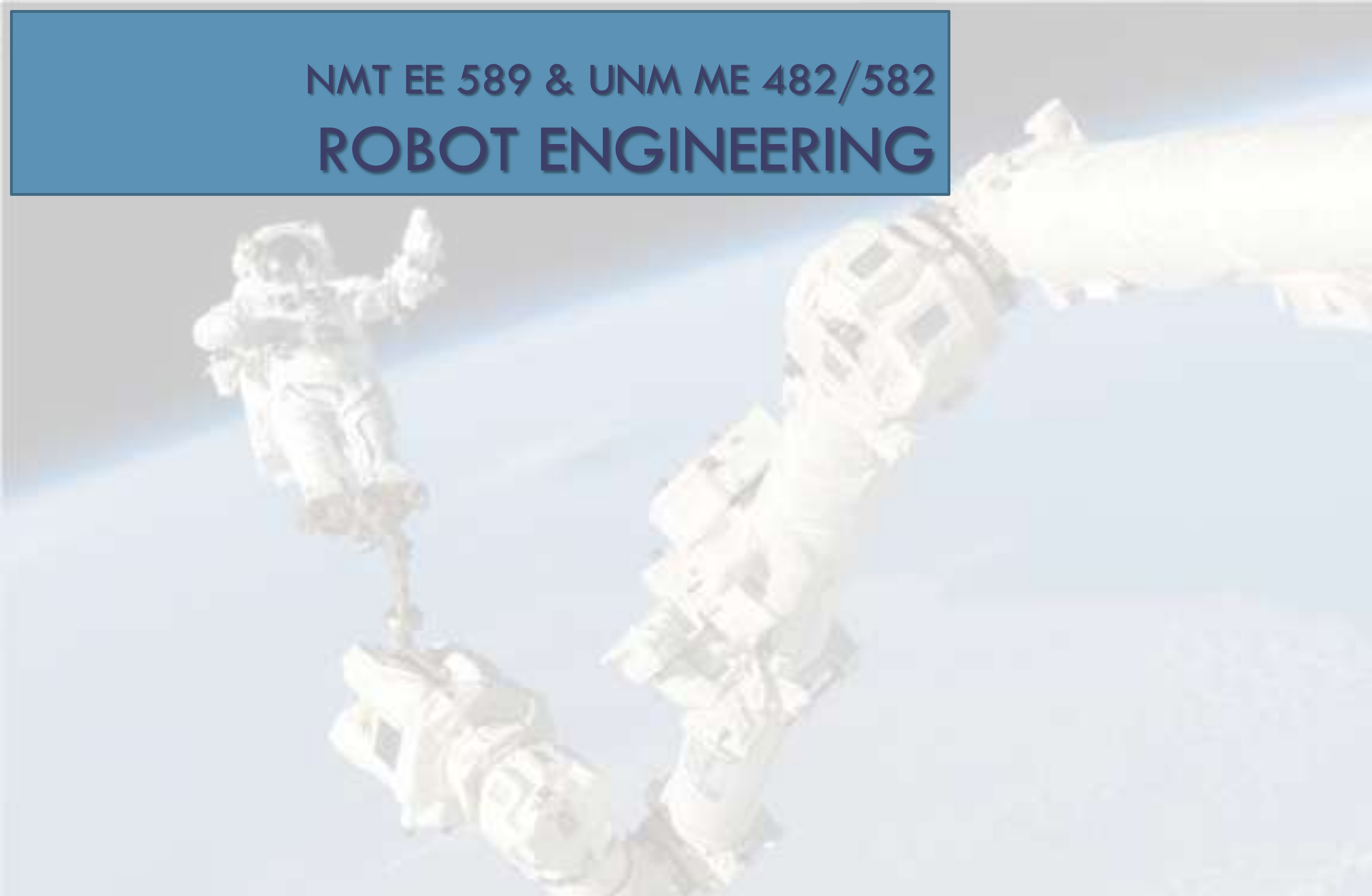


NMT EE 589 & UNM ME 482/582
ROBOT ENGINEERING

Dr. Stephen Bruder
NMT EE 589 & UNM ME 482/582

- **Moving beyond independent joint control**
- **As before we will develop the overall control strategy in two parts.**
  1. The first control law will serve to simplify our dynamic model.
  2. The second control law will serve to realize the trajectory following.

- Given the system (a robot) we wish to control as

$$D(\vec{q})\ddot{\vec{q}}(t)+C(\vec{q},\dot{\vec{q}})\dot{\vec{q}}(t)+\vec{G}(\vec{q})=\vec{\tau}(t) \qquad (8.15)$$

- Consider a control law of the form

$$\vec{\tau}(t)=\mathscr{D}(t)\vec{u}(t)+\mathscr{C}(t)\dot{\vec{q}}(t)+\vec{\mathscr{G}}(t) \qquad (8.16)$$

- Applying the controller of equation (8.16) to the system of equation (8.15) gives

$$D(\vec{q})\ddot{\vec{q}}(t)+[C(\vec{q},\dot{\vec{q}})-\mathscr{C}(t)]\dot{\vec{q}}(t)+[\vec{G}(\vec{q})-\vec{\mathscr{G}}(t)]=\mathscr{D}(t)\vec{u}(t) \qquad (8.17)$$

- Let us choose

$$\mathscr{C}(t)=C(\vec{q},\dot{\vec{q}})$$

$$\mathscr{D}(t)=D(\vec{q})$$

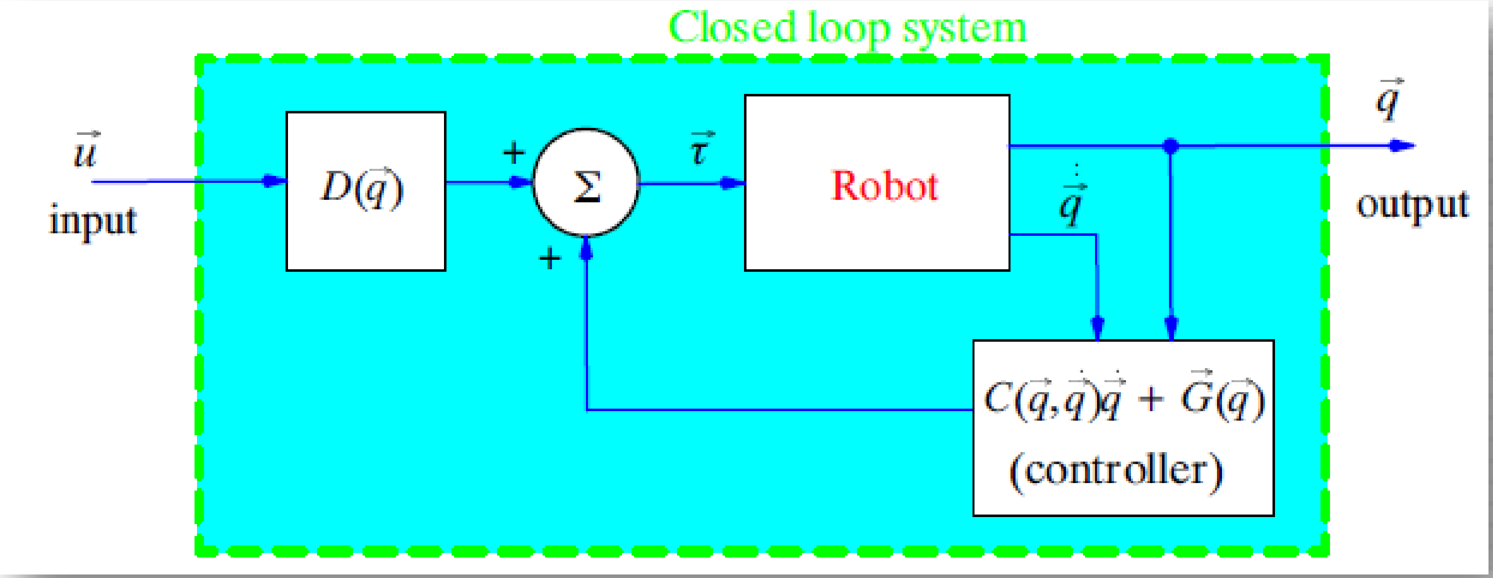
- and $\vec{\mathscr{G}}(t)=\vec{G}(\vec{q})$

- Hence the closed loop system becomes

$$D(\vec{q})\ddot{\vec{q}}(t)=D(\vec{q})\vec{u}(t)$$

$$\Rightarrow \qquad \ddot{\vec{q}}(t)=\vec{u}(t)$$

- **Since $D(\cdot)$ is a positive semi-definite matrix, it is always invertible!!**
  - We have linearized and decoupled the original MIMO, coupled, nonlinear system!!



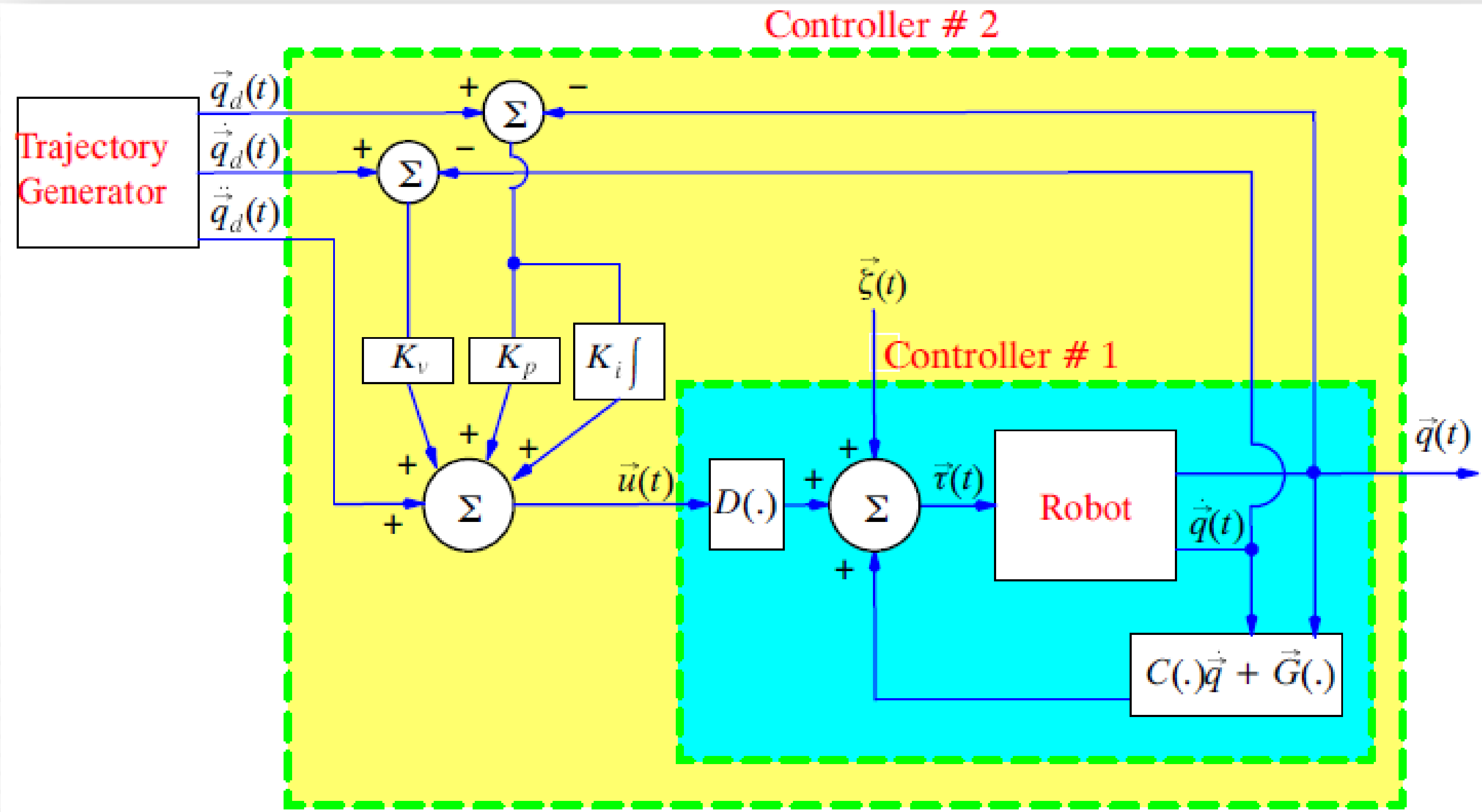
- **Note that, the trajectory following problem here is basically identical to the previous case of independent joint control, except that here we are working with vector quantities.**

- **Analogously, to achieve trajectory following choose**

$$\vec{u}(t) = \ddot{\vec{q}}_d(t) + K_v\dot{\vec{e}}(t) + K_p\vec{e}(t) + K_i\int_0^t \vec{e}(\sigma)d\sigma \qquad \textbf{(8.18)}$$

- **where the error vector is given by $\vec{e}(t) \doteq \vec{q}_d(t) - \vec{q}(t)$.**

- Thus, the overall controller is shown in the figure below.



- **In a realistic problem we will not know the dynamic model exactly, therefore when we formulate the controller as**

$$\vec{\tau}(t) = \mathcal{D}(t)\vec{u}(t) + \mathcal{C}(t)\dot{\vec{q}}(t) + \vec{\mathcal{G}}(t) \qquad \textbf{(8.19)}$$

- **We may actually choose**

$$\mathcal{C}(t) = \hat{C}(\vec{q}, \dot{\vec{q}})$$

$$\mathcal{D}(t) = \hat{D}(\vec{q})$$

and

$$\vec{\mathcal{G}}(t) = \hat{\vec{G}}(\vec{q})$$

- **where, hopefully** $\widehat{D}(\vec{q}), \hat{C}(\vec{q}, \dot{\vec{q}})$**, and** $\hat{\vec{G}}(\vec{q})$ **are "close to"** $D(\vec{q}), C(\vec{q}, \dot{\vec{q}})$**, and** $\vec{G}(\vec{q})$**, respectively.**

- Now, applying this controller to the robot's dynamic model gives

$$D(\vec{q})\ddot{\vec{q}}(t)+[C(\vec{q},\dot{\vec{q}})-\hat{C}(\vec{q},\dot{\vec{q}})]\dot{\vec{q}}(t)+[\vec{G}(\vec{q})-\hat{\vec{G}}(\vec{q})]=\hat{D}(\vec{q})\vec{u}(t)$$

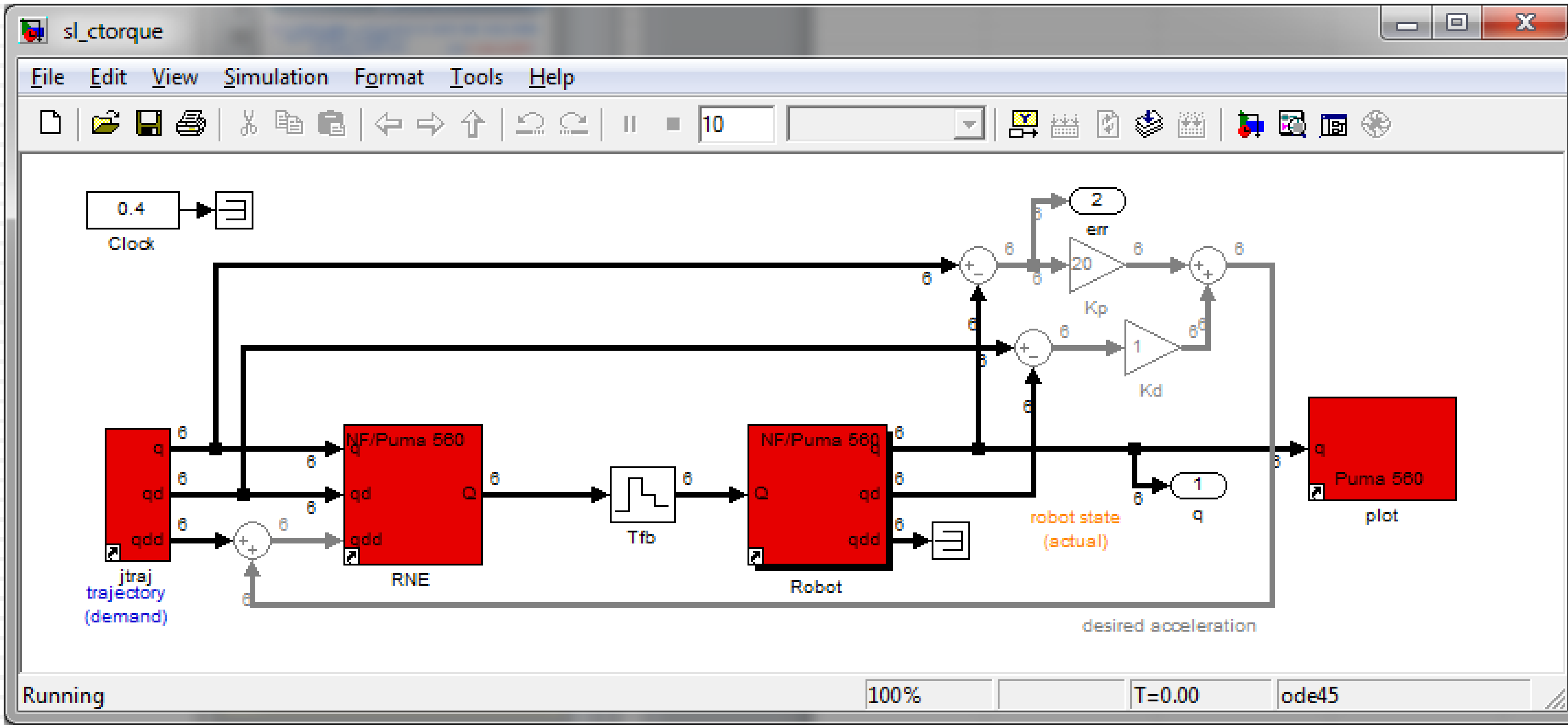
$$\Rightarrow \quad \vec{u}(t)=\hat{D}^{-1}(\vec{q})\left\{D(\vec{q})\ddot{\vec{q}}(t)+[C(\vec{q},\dot{\vec{q}})-\hat{C}(\vec{q},\dot{\vec{q}})]\dot{\vec{q}}(t)+[\vec{G}(\vec{q})-\hat{\vec{G}}(\vec{q})]\right\}$$

$$=\left[\hat{D}^{-1}(\vec{q})D(\vec{q})\right]\ddot{\vec{q}}(t)+\vec{\zeta}(t)$$

$$\approx \ddot{\vec{q}}(t)+\vec{\zeta}(t)$$

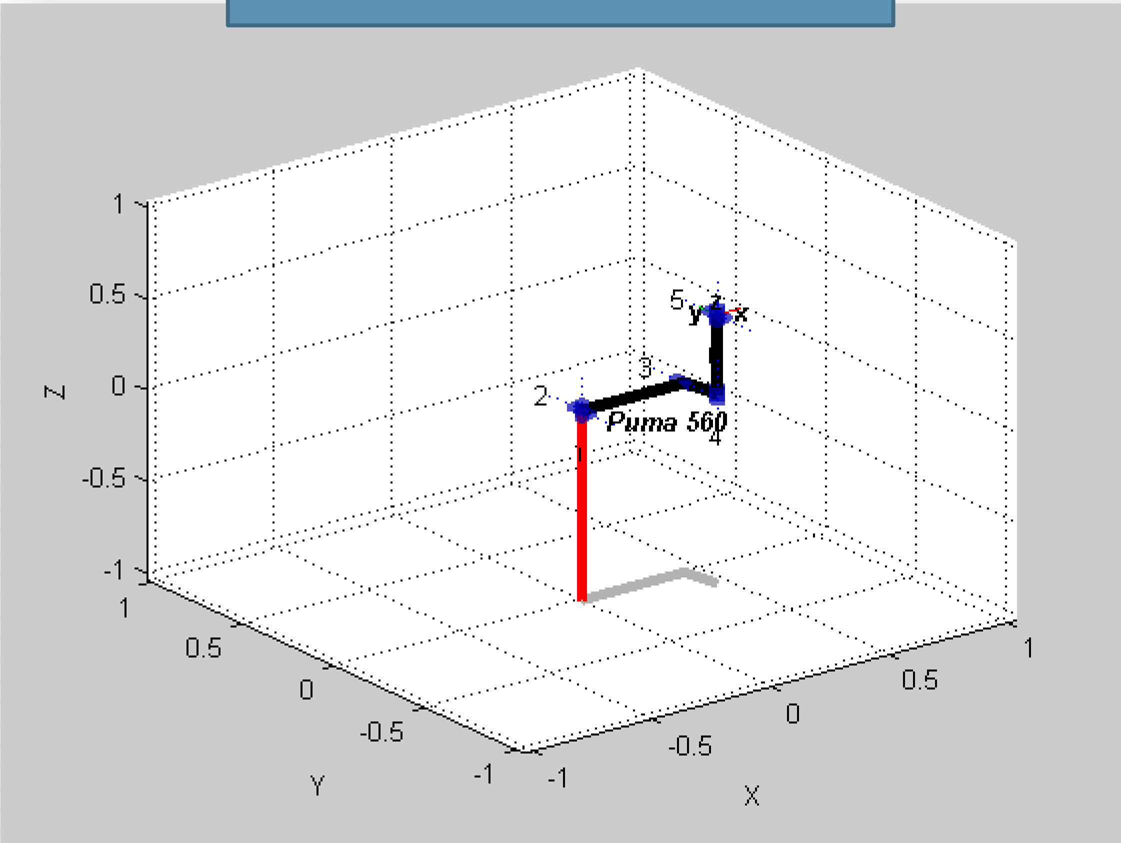
- where again $\vec{\zeta}(t)$ can be treated as a disturbance, and the previous argument can be employed.

❑ A Computed Torque Example – The PUMA 560 Robot



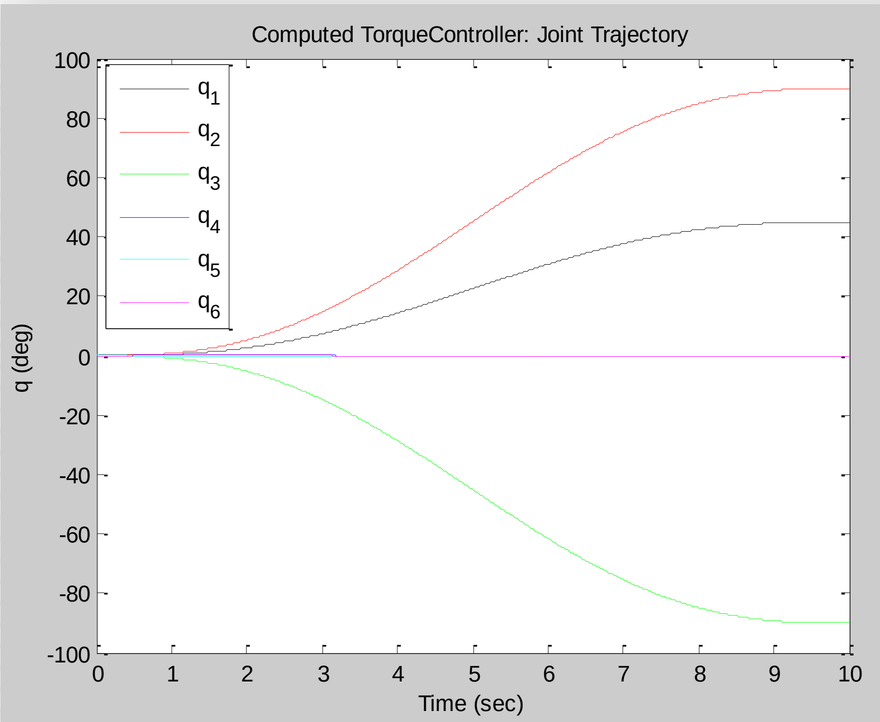
An Example – The PUMA 560 Robot

A MATLAB Animation



❑ An Example – The PUMA 560 Robot

Joint Angles (q)



Tracking Errors in q