

RHM 11 Feb 06
①

Lecture 6.1 - Trajectory Tracking

Trajectory tracking problem: Given a nonlinear control system

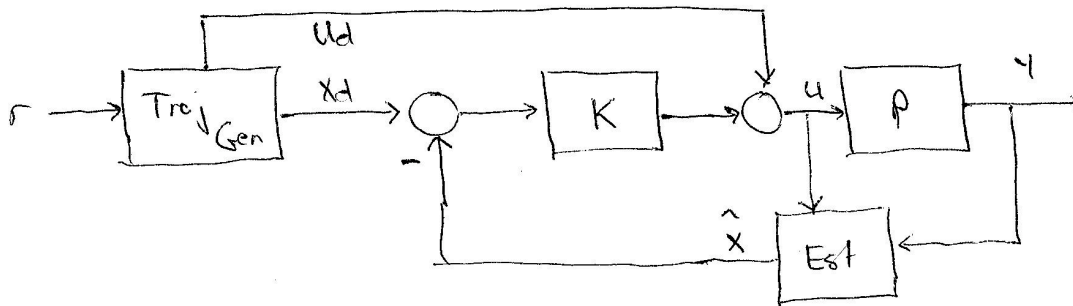
$$\dot{x} = f(x, u) \quad x \in \mathbb{R}^n, u \in \mathbb{R}^p$$

$$y = h(x) \quad y \in \mathbb{R}^q$$

and a reference trajectory $r(t) \in \mathbb{R}^q$, find a control law

$$u = \alpha(x, r) \text{ such that } \lim_{t \rightarrow \infty} (y(t) - r(t)) = 0.$$

Approach: two degree of freedom design



Trajectory generation: find feasible trajectory (satisfies dynamics)

$$\left. \begin{aligned} \dot{x}_d &= f(x_d, u_d) \\ r &= h(x_d) \end{aligned} \right\} \rightarrow x_d(r), u_d(r) \quad \left. \begin{array}{l} \text{will discuss} \\ \text{this problem} \\ \text{on Wed} \end{array} \right\}$$

Tracking: choose $u = \alpha(\hat{x}, x_d, u_d)$ such that closed loop system is stable, with desired performance

Estimation: determine $\hat{x}$ from y, u

RMM 11 Feb 06

(2)

Review: reference tracking for linear systems

$$\dot{x} = Ax + Bu$$

$$u = -Kx + k_r r$$

$$y = Cx$$

$$r = \text{constant}$$

1. Choose K to give desired closed loop dynamics (e.g. pole placement, LQR, etc)
2. Choose k_r to give desired output value at equilibrium

$$\dot{x} = Ax + Bu = (A - BK)x + Bk_r r$$

$$\text{Eq pt: } x_e = -(A - BK)^{-1} Bk_r r$$

$$y_e = -C(A - BK)^{-1} Bk_r r \stackrel{\text{desired}}{=} r$$

$$\Rightarrow k_r = -\frac{1}{C(A - BK)^{-1} B}$$

Remarks

1. This approach works by stabilizing the origin and then using $k_r r$ to "push" the system to x_e . Works because system is linear
2. Alternative formulation: solve directly for eq pt

$$\begin{bmatrix} 0 \\ r \end{bmatrix} = \begin{bmatrix} A & B \\ C & 0 \end{bmatrix} \begin{bmatrix} x_d \\ u_d \end{bmatrix} \rightarrow u = \underbrace{-K(x - x_d)}_{\text{stabilize eq pt}} + \underbrace{u_d}_{\text{nominal input}}$$

HW: show this is equiv to $k_r r$

3. Fundamental assumption: (x_d, u_d) is eq pt $\Rightarrow r$ is constant.

Gain scheduling

Nonlinear system with feasible trajectory:

$$\begin{array}{ll} \dot{x} = f(x, u) & \dot{x}_d = f(x_d, u_d) \\ y = h(x) & r(t) = h(x_d) \end{array} \left. \vphantom{\begin{array}{l} \dot{x} = f(x, u) \\ y = h(x) \end{array}} \right\} \begin{array}{l} \text{Solve via optimal} \\ \text{control or other} \\ \text{means (more on Wed)} \end{array}$$

To stabilize the reference trajectory, look at error $e = x - x_d$

$$\dot{x} = f(x, u)$$

$$e = x - x_d$$

$$\dot{x}_d = f(x_d, u_d)$$

$$v = u - u_d$$

$$\dot{e} = f(x, u) - f(x_d, u_d) =: F(e, v, x_d, u_d)$$

$$= f(e + x_d, v + u_d) - f(x_d, u_d)$$

Nonlinear,
time-varying
system

We can now treat (x_d, u_d) as parameters in the controller and linearize around $(e, v) = (0, 0)$

$$\dot{e} \approx A_d e + B_d v \quad A_d = \left. \frac{\partial F}{\partial e} \right|_{(0,0)} = \left. \frac{\partial f}{\partial x} \right|_{(x_d, u_d)}$$

$$B_d = \left. \frac{\partial F}{\partial v} \right|_{(0,0)} = \left. \frac{\partial f}{\partial u} \right|_{(x_d, u_d)}$$

Now stabilize $e=0$ by choosing K_d such that $(A_d - B_d K_d)$ stable.

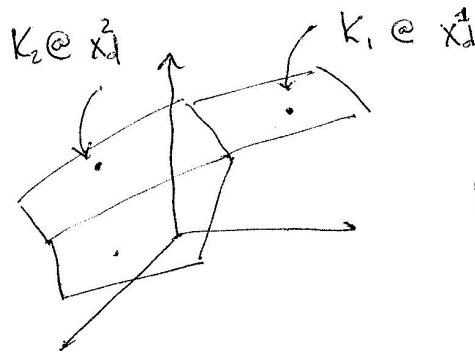
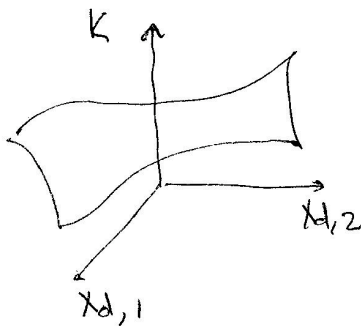
$$u = -K_d e + u_d = -K_d (x - x_d) + u_d$$

Note that K_d depends on (x_d, u_d) . Eg

$$u = -K_d(x_d, u_d) \cdot (x - x_d) + u_d$$

Remarks

1. Does not assume (x_d, u_d) are equilibrium values, just that they satisfy dynamics. Eq pt is a special case ($r = \text{constant}$)
2. More generally, can schedule gains on any parameter, or even the state: $u = -K(x, u) \cdot (x - x_d) + u_d$. Use with care (NOL)
3. Problem: linearizing about desired eq pt $\Rightarrow$ linearization may not work well if you are far from this point.
4. In practice, implement gain scheduling through interpolation



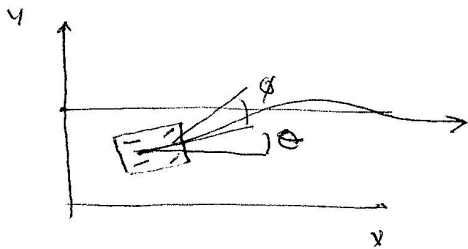
$$K_d = \sum \alpha_i(x) K_i$$

↑
interpolation fcn

Sample implementation: "falcon" library (RMM web page)

5. Theory: not much can be said about when this works
 - OK if (x_d, u_d) are varying sufficiently slowly
 - Works well in practice, even if they aren't

(5)

Example: steering control with velocity scheduling

$$\begin{aligned}\dot{x} &= \cos \theta v \\ \dot{y} &= \sin \theta v \\ \dot{\theta} &= \frac{v}{l} \tan \phi\end{aligned}$$

$$\left. \begin{aligned}\dot{x}_d &= v_r \\ y_d &= y_{ref} \\ \theta_d &= 0 \\ v_d &= v_r \\ \phi_d &= 0\end{aligned} \right\} \begin{array}{l} x_d \\ u_d \end{array}$$

Note that $(x_d, y_d, \theta_d; v_d, \phi_d)$ are not at equilibrium, but they do satisfy the equations of motion.

$$A_d = \left. \frac{\partial f}{\partial x} \right|_{(x_d, u_d)} = \begin{bmatrix} 0 & 0 & -\sin \theta \\ 0 & 0 & \cos \theta \\ 0 & 0 & 0 \end{bmatrix} \bigg|_{(x_d, u_d)} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}$$

$$B_d = \left. \frac{\partial f}{\partial u} \right|_{(x_d, u_d)} = \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & v_r/l \end{bmatrix} \leftarrow \text{depends on } u_d \text{ through } v_r$$

Error dynamics: $e = x - x_d$ $w = u - u_d$

$$\left. \begin{aligned}\dot{e}_x &= w_1 & w_1 &= -\lambda_1 e_x \\ \dot{e}_y &= e_\theta & w_2 &= -\frac{1}{v_r} (a_1 e_y + a_2 e_\theta) \\ \dot{e}_\theta &= \frac{v_r}{l} w_2\end{aligned} \right\} \begin{array}{l} \text{pole at } \lambda_1 \text{ and} \\ s^2 + a_1 s + a_2 = 0 \end{array}$$

Controller in original coordinates

$$\begin{bmatrix} v \\ \phi \end{bmatrix} = - \underbrace{\begin{bmatrix} \lambda_1 & 0 & 0 \\ 0 & \frac{a_1 l}{v_r} & \frac{a_2 l}{v_r} \end{bmatrix}}_{K_d} \underbrace{\begin{bmatrix} x - v_r t \\ y - y_r \\ \theta \end{bmatrix}}_e + \underbrace{\begin{bmatrix} v_r \\ 0 \end{bmatrix}}_{u_d}$$

Intuition: at slower speeds, turn wheel harder to get same time response. Note that gains blow up if $v_r \rightarrow 0$ (not recommended)