CDS 101/110a: Lecture 4.1
State Feedback

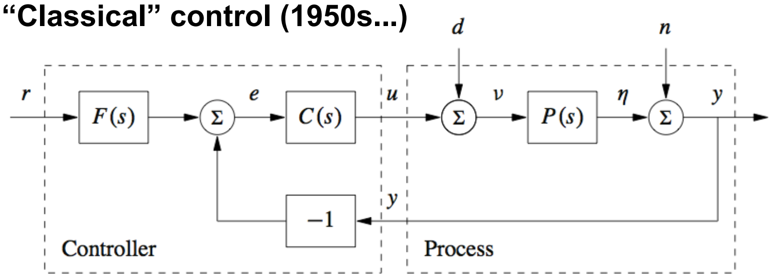
Richard M. Murray
19 October 2015

Goals:
- Introduce control design concepts and classical "design patterns"
- Describe the design of state feedback controllers for linear systems
- Define reachability of a control system and give tests for reachability

Reading:
- Åström and Murray, Feedback Systems 2e, Ch 7

Design Patterns for Control Systems

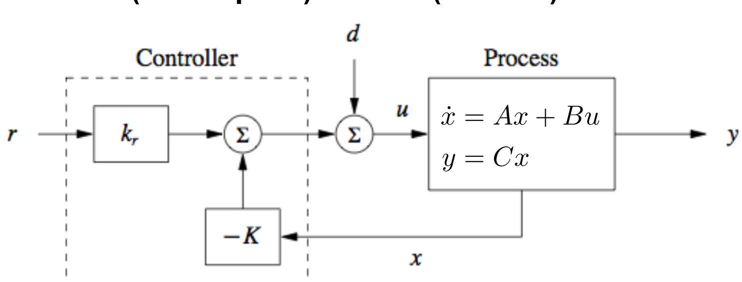
"Classical" control (1950s...)



- Reference input shaping
- Feedback on output error
- Compensator dynamics shape closed loop response
- *Uncertainty* in process dynamics $P(s)$ + external disturbances (d) & noise (n)

- Goal: output $y(t)$ should track reference trajectory $r(t)$
- Design typically done in "frequency domain" (second half of CDS 101/110a)

"Modern" (state space) control (1970s...)



- Assume dynamics are given by linear system, with known A, B, C matrices
- Measure the state of the system and use this to modify the input
- $u = -K\,x + k_r\,r$

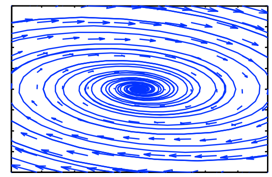
- Goal unchanged: output $y(t)$ should track reference trajectory $r(t)$ [often constant]

State Space Control Design Concepts

System description: single input, single output system (MIMO also OK)

$$\dot{x} = f(x, u) \qquad x \in \mathbb{R}^n,\ x(0) \text{ given}$$
$$y = h(x) \qquad u \in \mathbb{R},\ y \in \mathbb{R}$$

Stability: stabilize the system around an equilibrium point

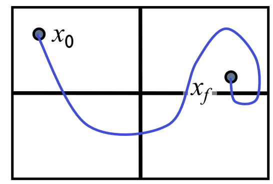


- Given equilibrium point $x_e \in \mathbb{R}^n$, find control "law" $u = \alpha(x)$ such that

$$\lim_{t \to \infty} x(t) = x_e \text{ for all } x(0) \in \mathbb{R}^n$$

- Often choose x_e so that $y_e = h(x_e)$ has desired value r (constant)

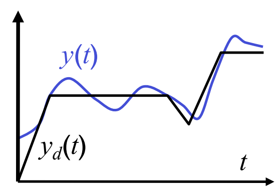
Reachability: steer the system between two points



- Given $x_o, x_f \in \mathbb{R}^n$, find an input $u(t)$ such that

$$\dot{x} = f(x, u(t)) \text{ takes } x(t_0) = x_0 \to x(T) = x_f$$

Tracking: track a given output trajectory

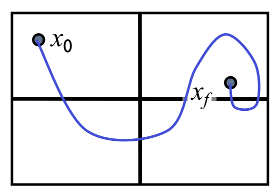


- Given $r = y_d(t)$, find $u = \alpha(x,t)$ such that

$$\lim_{t \to \infty} \big(y(t) - y_d(t)\big) = 0 \text{ for all } x(0) \in \mathbb{R}^n$$

Reachability of Input/Output Systems

$$\dot{x} = f(x, u) \qquad x \in \mathbb{R}^n,\ x(0) \text{ given}$$
$$y = h(x) \qquad u \in \mathbb{R},\ y \in \mathbb{R}$$



Defn An input/output system is *reachable* if for any $x_o, x_f \in \mathbb{R}^n$ and any time $T > 0$ there exists an input $u_{[0,T]} \in \mathbb{R}$ such that the solution of the dynamics starting from $x(0) = x_0$ and applying input $u(t)$ gives $x(T) = x_f$.

Note: the term "controllable" is also commonly used to describe this concept

Remarks

- In the definition, x_0 and x_f do not have to be equilibrium points ⇒ we don't necessarily stay at x_f after time T.
- Reachability is defined in terms of states ⇒ doesn't depend on output
- For *linear systems,* can characterize reachability by looking at the general solution:

$$\dot{x} = Ax + Bu \qquad y = Cx \qquad x(T) = e^{AT}x_0 + \int_{\tau=0}^{T} e^{A(T-\tau)}Bu(\tau)\,d\tau$$

If integral is "surjective" (as a linear operator), then we can find an input to achieve any desired final state.

Tests for Reachability

$$\dot{x} = Ax + Bu \qquad x \in \mathbb{R}^n,\ x(0) \text{ given}$$
$$y = Cx \qquad u \in \mathbb{R},\ y \in \mathbb{R}$$

$$x(T) = e^{AT}x_0 + \int_{\tau=0}^{T} e^{A(T-\tau)}Bu(\tau)d\tau$$

Thm A linear system is reachable if and only if the $n \times n$ *reachability matrix*

$$\begin{bmatrix} B & AB & A^2B & \cdots & A^{n-1}B \end{bmatrix}$$

is full rank.

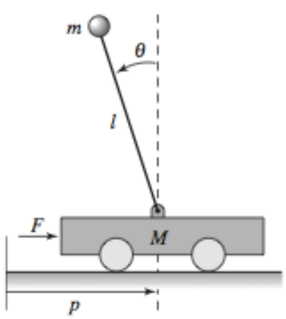
Note: also called "controllability" matrix

Remarks

- *Very* simple test to apply. In MATLAB, use ctrb(A,B) and check rank w/ det()
- If this test is satisfied, we say "the pair (A,B) is reachable"
- Some insight into the proof can be seen by expanding the matrix exponential

$$\begin{aligned} e^{A(T-\tau)}B &= \left(I + A(T-\tau) + \frac{1}{2}A^2(T-\tau)^2 + \cdots + \frac{1}{(n-1)!}A^{n-1}(T-\tau)^{n-1} + \cdots\right)B \\ &= B + AB(T-\tau) + \frac{1}{2}A^2B(T-\tau)^2 + \cdots + \frac{1}{(n-1)!}A^{n-1}B(T-\tau)^{n-1} + \cdots \end{aligned}$$

Example #1: Linearized pendulum on a cart



Question: can we locally control the position of the cart by proper choice of input?

- Simple case: move from one equilibrium point to another
- More generally: hit arbitrary position, angle and velocities (but near equilibrium point)

Approach: look at the linearization around upright position (good approximation to the full dynamics if θ remains small)

$$\frac{d}{dt}\begin{bmatrix} p \\ \theta \\ \dot{p} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & \frac{m^2l^2g}{M_tJ_t-m^2l^2} & \cancelto{0}{\frac{-cJ_t}{M_tJ_t-m^2l^2}} & \cancelto{0}{\frac{-\gamma lm}{M_tJ_t-m^2l^2}} \\ 0 & \frac{M_tmgl}{M_tJ_t-m^2l^2} & \cancelto{0}{\frac{-clm}{M_tJ_t-m^2l^2}} & \cancelto{0}{\frac{-\gamma M_t}{M_tJ_t-m^2l^2}} \end{bmatrix}\begin{bmatrix} p \\ \theta \\ \dot{p} \\ \dot{\theta} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \frac{J_t}{M_tJ_t-m^2l^2} \\ \frac{lm}{M_tJ_t-m^2l^2} \end{bmatrix}u$$

- Simplify by setting c, γ = 0
- Define $\mu = M_tJ_t - m^2l^2$

$$W_r = \begin{array}{c} \begin{bmatrix} 0 & \frac{J_t}{\mu} & 0 & \frac{gl^3m^3}{\mu^2} \\ 0 & \frac{lm}{\mu} & 0 & \frac{gl^2m^2(m+M)}{\mu^2} \\ \frac{J_t}{\mu} & 0 & \frac{gl^3m^3}{\mu^2} & 0 \\ \frac{lm}{\mu} & 0 & \frac{gl^2m^2(m+M)}{\mu^2} & 0 \end{bmatrix} \\ \begin{array}{cccc} B & AB & A^2B & A^3B \end{array} \end{array}$$

- Full rank as long as constants are such that columns 1 and 3 are not multiples of each other
- $\Rightarrow$ reachable as long as $\det(W_r) = \dfrac{g^2l^4m^4}{\mu^4} \neq 0$
- $\Rightarrow$ can "steer" (linearization) between any two points by proper choice of input

Trajectory Generation (and Tracking)

Given that a (linear) system is reachable, how do we compute the inputs??

- Method #1: formulate as an "optimal control problem" and solve numerically

$$\min_{u(\cdot)} \int_0^T L(x,u)\,dt \qquad \text{subject to} \qquad \dot x = f(x,u), \qquad x(0)=x_0,\ x(T)=x_f$$

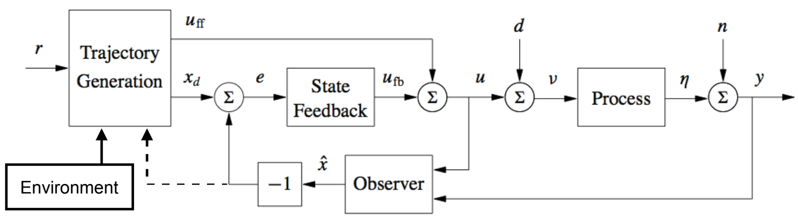
- Method #2: create a stabilizing control law to an equilibrium point: $u = u_e + \alpha(x\text{-}x_e)$

$$\lim_{t\to\infty} x(t) = x_e \text{ for all } x(0)\in\mathbb{R}^n \qquad \Longrightarrow \qquad x(0)=x_0 \ \to\ x(\infty)=x_e$$

- These methods *only* work if the system is reachable and almost always require that the linearization at a nearby equilibrium point be reachable (which we can check)

Given feasible input/state trajectory, use feedback to manage uncertainty

- General picture = trajectory generation (feedforward) + feedback compensation



Types of uncertainty:

- Process disturbances
- Sensor noise
- Unmodeled dynamics

More on trajectory generation in CDS 112

Control Design Concepts

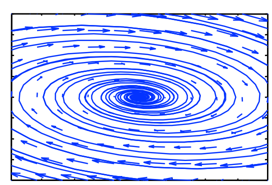
System description: single input, single output system (MIMO also OK)

$$\begin{aligned} \dot x &= f(x,u) \qquad & x\in\mathbb{R}^n,\ x(0) \text{ given} \\ y &= h(x,u) \qquad & u\in\mathbb{R},\ y\in\mathbb{R} \end{aligned}$$

Stability: stabilize the system around an equilibrium point

- Given equilibrium point $x_e \in \mathbb{R}^n$, find control "law" $u=\alpha(x)$ such that

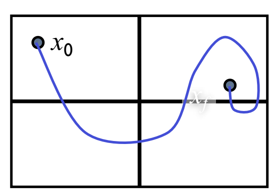
$$\lim_{t\to\infty} x(t) = x_e \text{ for all } x(0)\in\mathbb{R}^n$$



Reachability: steer the system between two points

- Given $x_o, x_f \in \mathbb{R}^n$, find an input $u(t)$ such that

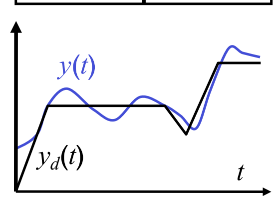
$$\dot x = Ax + Bu \ \text{ takes } x(t_0) = x_0 \to x(T) = x_f$$



Tracking: track a given output trajectory

- Given $y_d(t)$, find $u=\alpha(x,t)$ such that

$$\lim_{t\to\infty} \big(y(t) - y_d(t)\big) = 0 \text{ for all } x(0)\in\mathbb{R}^n$$



State space controller design for linear systems

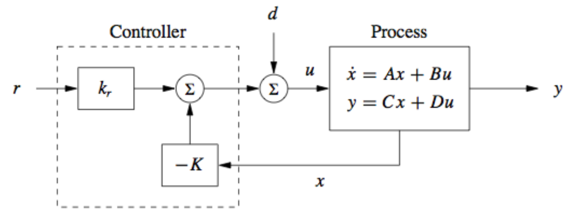
$$\dot{x} = Ax + Bu \qquad x \in \mathbb{R}^n,\ x(0) \text{ given}$$
$$y = Cx \qquad u \in \mathbb{R},\ y \in \mathbb{R}$$

$$x(T) = e^{AT}x_0 + \int_{\tau=0}^{T} e^{A(T-\tau)}Bu(\tau)d\tau$$

Goal: find a linear control law $u = -Kx + k_r r$ such that the closed loop system

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

is stable at equilibrium point x_e with $y_e = r$.



Remarks

- If $r = 0$, control law simplifies to $u = -Kx$ and system becomes $\dot{x} = (A - BK)x$
- Stability based on eigenvalues ⇒ use K to make eigenvalues of $(A - BK)$ stable
- Can also link eigenvalues to *performance* (eg, initial condition response)
- Question: when can we place the eigenvalues anyplace that we want?

Theorem The eigenvalues of $(A - BK)$ can be set to arbitrary values if and only if the pair (A, B) is reachable.

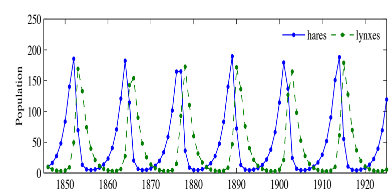
MATLAB/Python: K = place(A, B, eigs)

Python users: use python-control toolbox (available at python-control.org)

Example #2: Predator prey

System dynamics

$$\frac{dH}{dt} = (r+u)H\left(1 - \frac{H}{k}\right) - \frac{aHL}{c+H}, \qquad H \geq 0,$$
$$\frac{dL}{dt} = b\frac{aHL}{c+H} - dL, \qquad L \geq 0.$$



- Stable limit cycle with unstable equilibrium point at $H_e = 20.6$, $L_e = 29.5$
- Can we design the dynamics of the system by modulating the food supply ("u" in "$r + u$" term)

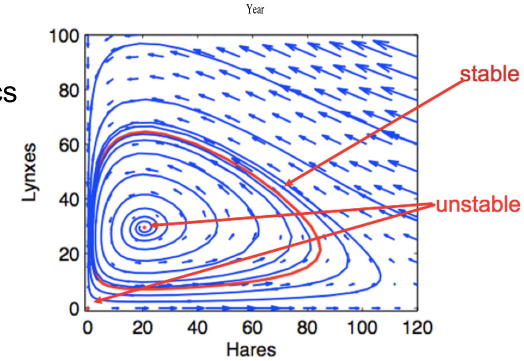
Q1: can we move from some initial population of foxes and rabbits to a specified one in time T by modulation of the food supply?

- Eg: need large amount of food for 1872 Olympics

Q2: can we stabilize the lynx population around a desired equilibrium point (eg, L_d = ~30)?

- Try to keep lynx and hare population in check

Approach: try to stabilize using state feedback law



Example #2: Problem setup

Equilibrium point calculation

$$\frac{dH}{dt} = (r+u)H\left(1-\frac{H}{k}\right) - \frac{aHL}{c+H}$$
$$\frac{dL}{dt} = b\frac{aHL}{c+H} - dL$$

- x_e = (20.6, 29.5), u_e = 0, L_e = 29.5

```
f = inline('predprey(0, x)', 'x');
xeq = fsolve(f, [20, 30])'; He = xeq(1); Le = xeq(2);

% Generate the linearization around the eq point
App = [
  -((a*c*k*Le + (c + He)^2*(2*He - k)*r)/((c + He)^2*
  (a*b*c*Le)/(c + He)^2, -d + (a*b*He)/(c + He)
];
Bpp = [He*(1 - He/k); 0];

% Check reachability
if (det(ctrb(App, Bpp)) ~= 0) disp "reachable"; end
```

Linearization

- Compute linearization around equilibrium point, x_e:

$$A = \left.\frac{\partial f}{\partial x}\right|_{(x_e,u_e)} \qquad B = \left.\frac{\partial f}{\partial u}\right|_{(x_e,u_e)} \qquad \frac{dx}{dt} \approx A(x-x_e) + B(u-u_e) + \begin{array}{c}\text{higher}\\ \text{order terms}\end{array}$$

- Redefine local variables: $z = x - x_e,\ v = u - u_e$

$$\frac{d}{dt}\begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} -\frac{acL_e}{(c+H_e)^2} - \frac{2H_e r}{k} + r & -\frac{aH_e}{c+H_e} \\ \frac{abcL_e}{(c+H_e)^2} & \frac{abH_e}{c+H_e} - d \end{bmatrix} \cdot \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} + \begin{bmatrix} H_e\left(1-\frac{H_e}{k}\right) \\ 0 \end{bmatrix} v$$

- Reachable? YES, if a, $b \neq 0$ (check [B AB]) ⇒ can locally steer to any point

Example #2: Stabilization via eigenvalue assignment

$$\frac{d}{dt}\begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} -\frac{acL_e}{(c+H_e)^2} - \frac{2H_e r}{k} + r & -\frac{aH_e}{c+H_e} \\ \frac{abcL_e}{(c+H_e)^2} & \frac{abH_e}{c+H_e} - d \end{bmatrix} \cdot \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} + \begin{bmatrix} H_e\left(1-\frac{H_e}{k}\right) \\ 0 \end{bmatrix} v$$

Control design:

$$v = -Kz = -k_1(H-H_e) - k_2(L-L_e)$$
$$u = u_e + K(x-x_e)$$

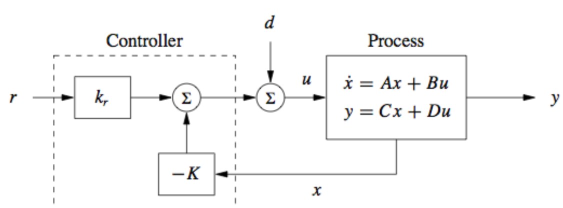
Place poles at stable values

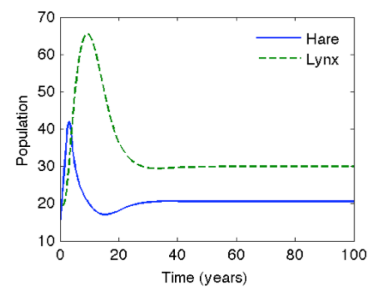
- Choose λ = -0.1, -0.2
- MATLAB: Kpp = place(App, Bpp, [-0.1; -0.2]);

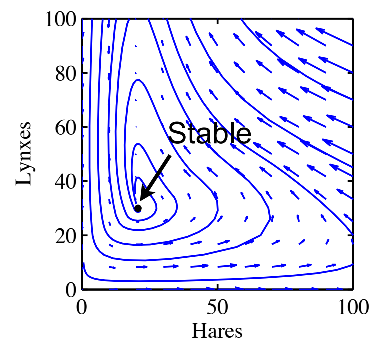
Key principle: *design of dynamics*

- Use feedback to create a stable equilibrium point

More advanced: control to desired value $r = L_d$ (Wed)







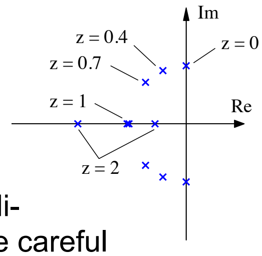
Implementation Details

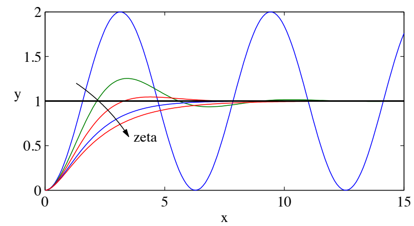
Eigenvalues determine performance

- For each eigenvalue $\lambda_i = \sigma_i + j\omega_i$, get a contribution of the form

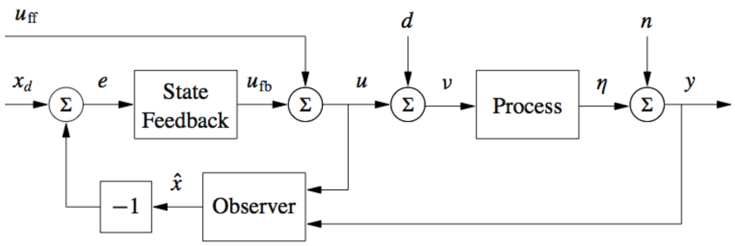
$$y_i(t) = e^{-\sigma_i t}\big(a\sin(\omega_i t) + b\cos(\omega_i t)\big)$$

- Repeated eigenvalues can give additional terms of the form $t^k e^{\sigma + j\omega}$ ⇒ be careful





Use *observer* to determine the current state if you can't measure it

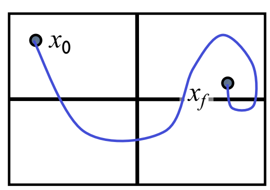


- Estimator looks at inputs and outputs of plant and estimates the current state
- Can show that if a system is *observable* then you can construct and estimator
- Use the *estimated* state as the feedback $u = K\hat{x}$

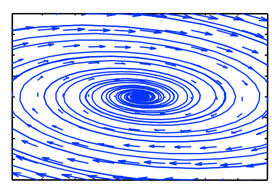
- Next week: basic theory of state estimation and observability
- CDS 110b: *Kalman filtering* = theory of optimal observers (and basis for particle filters, ...)

Summary: Reachability and State Space Feedback

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



$$\begin{bmatrix} B & AB & A^2B & \cdots & A^{n-1}B \end{bmatrix}$$



$$u = -Kx + k_r r$$

Key concepts

- Reachability: find u s.t. $x_0 \rightarrow x_f$
- Reachability rank test for linear systems
- State feedback to assign eigenvalues