CDS 101: Lecture 9.1
PID and Root Locus

Richard M. Murray
25 November 2002

Goals:

- Define PID controllers and describe how to use them
- Introduce the root locus technique and describe how to use it to choose loop gain
- Show some of the limitations of feedback due to RHP poles and zeros

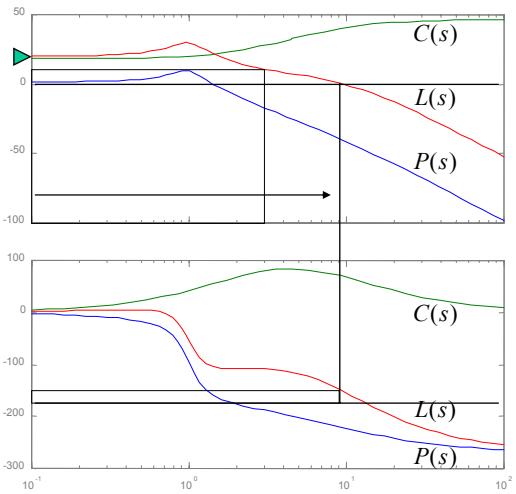
Reading:

- Astrom, Sec 6.1-6.4, 6.6
- *Optional*: PPH, Sec 13
- *Advanced*: Lewis, Chapter 12 + Sec 13.1

Lecture 8.1: Frequency Domain Design

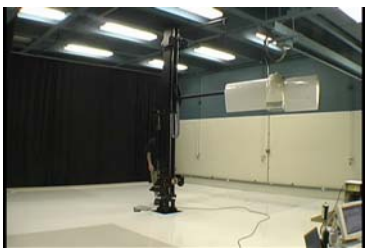
Loop Shaping for Stability and Performance

- Steady state error, bandwidth, tracking

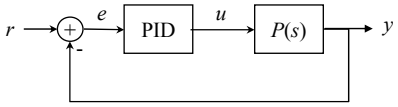


Main ideas

- Performance specifications give bounds on loop transfer function
- Use controller to shape response
- Gain/phase relationships constrain design approach
- Standard compensators: proportional, lead, PI



Overview: PID control



$$u = K_p e + K_I \int e + K_D \dot{e}$$

Intuition

- Proportional term: provides inputs that correct for "current" errors
- Integral term: insures that *steady state* error goes to zero (if not, control gets bigger)
- Derivative term: provides "anticipation" of upcoming changes

A bit of history on "three term control"

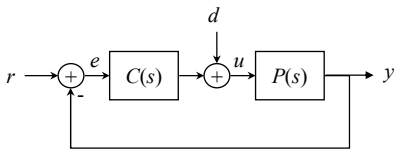
- First appeared in 1922 paper by Minorsky: "Directional stability of automatically steered bodies" under the name "three term control"
- Also realized that "small deviations" (linearization) could be used to understand the (nonlinear) system dynamics under control

Utility of PID

- PID control is most common feedback structure in engineering systems
- For many systems, only need PI or PD (special case)
- Many tools for tuning PID loops and designing gains (see reading)

25 Nov 02 R. M. Murray, Caltech CDS 3

Frequency domain compensation with PID

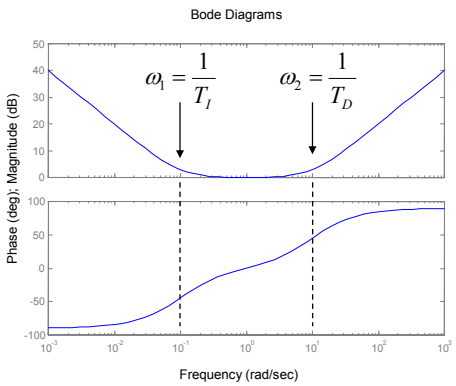


$$C(s) = K_p + K_I \cdot \frac{1}{s} + K_D s$$
$$= k(1 + \frac{1}{T_I s} + T_D s)$$
$$= \frac{kT_D}{T_I} \frac{(s+1/T_I)(s+1/T_D)}{s}$$

Transfer function for PID controller

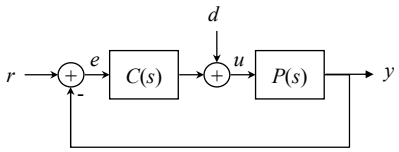
$$u = K_p e + K_I \int e + K_D \dot{e}$$
$$\downarrow$$
$$H_{ue}(s) = K_p + K_I \cdot \frac{1}{s} + K_D s$$

- Roughly equivalent to a PI controller with lead compensation
- Idea: gives high gain at low frequency plus phase lead at high frequency
- Place below desired crossover freq



25 Nov 02 R. M. Murray, Caltech CDS 4

Tools for Designing PID controllers



$$C(s) = K\left(1 + \frac{1}{T_I s} + T_D s\right)$$

Zeigler-Nichols tuning

- Design PID gains based on step response
- Works OK for many plants (a bit underdamped)
- Good way to get a first cut controller
- Frequency domain version also exists

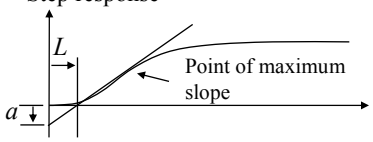
Caution: PID amplifies high frequency noise

- Sol'n: pole at high frequency

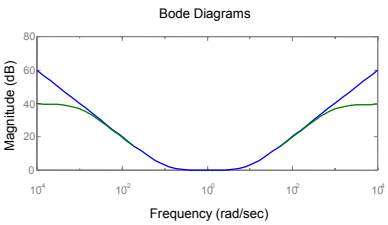
Caution: Integrator windup

- Prolonged error causes large integrated error
- Effect: get large undershoot (to reset integrator)
- Sol'n: move pole at zero to very small value
- Fancier sol'n: anti-windup compensation

Step response

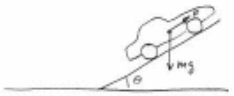


$K = 1.2/a \quad T_I = 2*L \quad T_D = L/2$

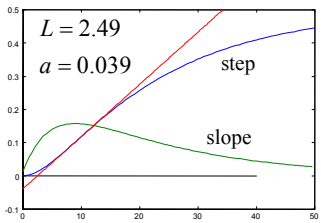


25 Nov 02 R. M. Murray, Caltech CDS 5

Example: PID cruise control

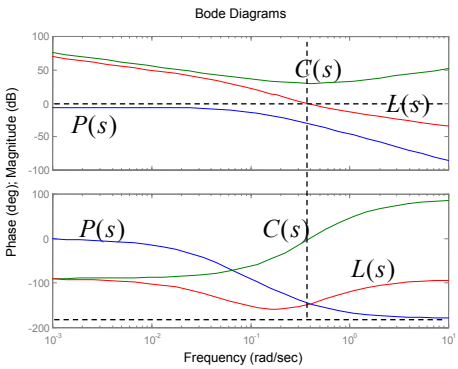


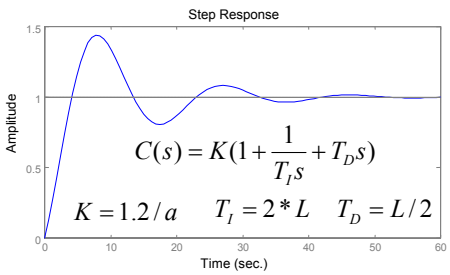
$$P(s) = \frac{1/m}{s + b/m} \cdot \frac{r}{s + a}$$



Ziegler-Nichols design for cruise controller

- Plot step response, extract L and a, compute gains





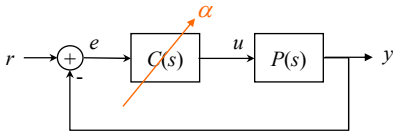
$$C(s) = K\left(1 + \frac{1}{T_I s} + T_D s\right)$$

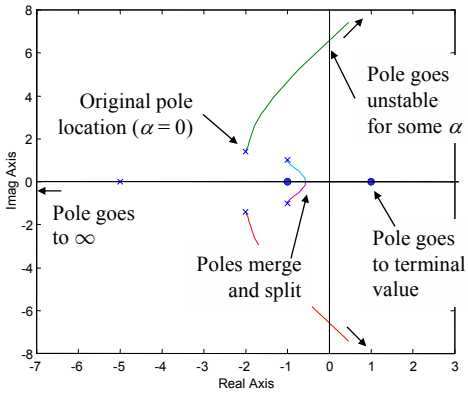
$K = 1.2/a \quad T_I = 2*L \quad T_D = L/2$

- Result: *sluggish* ⇒ increase loop gain

25 Nov 02 R. M. Murray, Caltech CDS 6

Pole Zero Diagrams and Root Locus Plots





Pole zero diagram verifies stability

- Roots of $1 + PC$ give closed loop poles
- Can *trace* the poles as a parameter is changed:

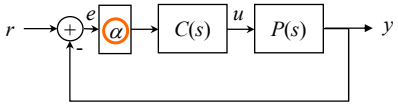
$$C(s) = K(1 + \frac{1}{\alpha T_I s} + T_D s)$$

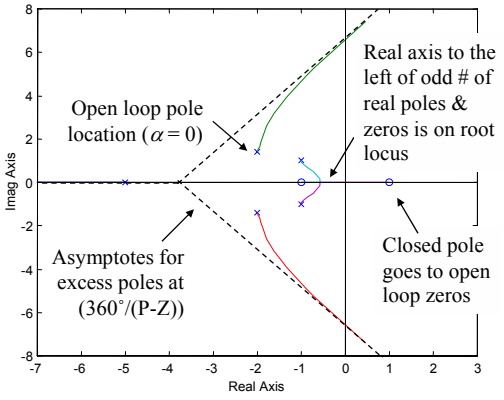
Root locus = locus of roots as parameter value is changed

- Can plot pole location for *any* single parameter; just repeatedly solve for roots
- Common choice in control is to vary the loop gain (K)

25 Nov 02 R. M. Murray, Caltech CDS 7

Root Locus for *Loop Gain*





Loop gain as root locus parameter

- Common choice for control design
- Special properties for loop gain
  - Roots go from poles of PC to zeros of PC
  - Excess poles go to infinity
  - Can compute asymptotes, break points, etc
- *Very* useful tool for control design
- MATLAB: rlocus

Additional comments

- Although loop gain is the most common parameter, *don't forget* that you can plot roots versus *any* parameter
- Need to link root location to performance…

25 Nov 02 R. M. Murray, Caltech CDS 8

Second Order System Response

Second order system response

- Spring mass dynamics, written in canonical form

Guidelines for pole placement

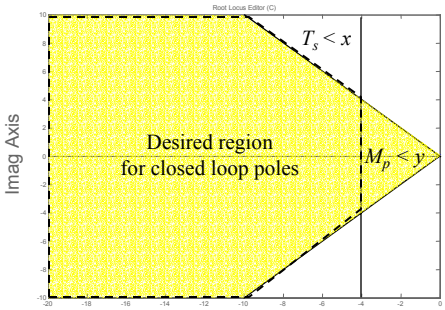
- Damping ratio gives Re/Im ratio
- Setting time determined by –Re(λ)

$$H(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} = \frac{\omega_n^2}{(s + \zeta\omega_n + j\omega_d)(s + \zeta\omega_n - j\omega_d)} \qquad \omega_d = \omega_n\sqrt{1-\zeta^2}$$

- Performance specifications

$$T_r \approx 1.8/\omega_n \qquad M_p \approx e^{-\pi\zeta/\sqrt{1-\zeta^2}}$$
$$T_s \approx 3.9/\zeta\omega_n \qquad e_{SS} = 0$$

ζ	M_p	Slope
0.707	4%	-1
0.5	16%	-1.7
0.25	44%	-3.9



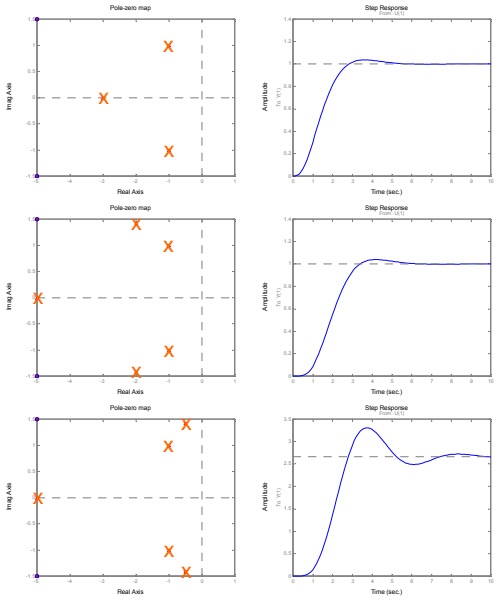
Effect of pole location on performance

Idea: look at "dominant poles"

- Poles nearest the imaginary axis (nearest to instability)
- Analyze using analogy to second order system

PZmap complements information on Bode/Nyquist plots

- Similar to gain and phase calculations
- Shows performance in terms of the *closed* loop poles
- Particularly useful for choosing system gain
- Also useful for deciding where to put controller poles and zeros (with practice)



Example: PID cruise control

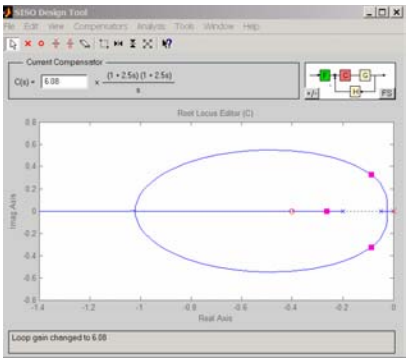
Start with PID control design:

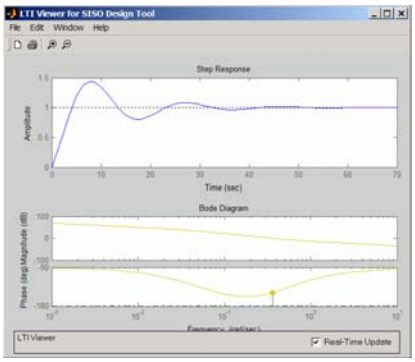
$$P(s) = \frac{1/m}{s + b/m} \cdot \frac{r}{s + a}$$

$$C(s) = K\left(1 + \frac{1}{T_I s} + T_D s\right)$$

Modify gain to improve performance

- Use MATLAB sisotool
- Adjust loop gain (K) to reduce overshoot and decrease settling time
  - $\zeta \approx 1 \Rightarrow$ less than 5% overshoot
  - Re(p) < -0.5 $\Rightarrow$ T_s less than 2 sec





Performance limits due to RHP poles and zeros

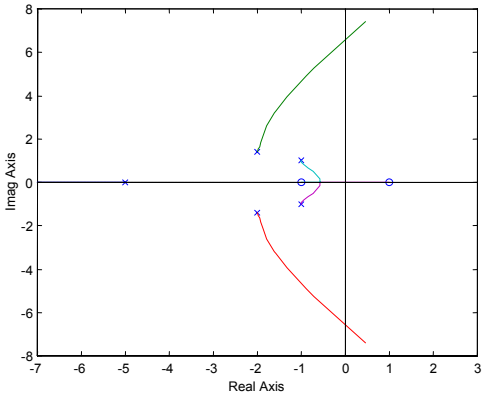
Loop gain limitations

- RHP poles require minimum value of loop gain to move unstable poles to LHP
- RHP zeros limit maximum value of loop gain before instability occurs
  - Serious design constraint!

Waterbed effect

$$M_1 = \max_{\omega_1 \le \omega \le \omega_2} |H_{er}(j\omega)|$$

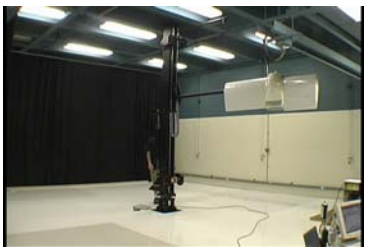
$$M_2 = \max_{0 \le \omega \le \infty} |H_{er}(j\omega)|$$

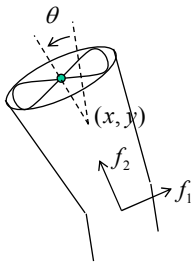


Thm: Suppose that P has a RHP zero at z. Then there exist constants c_1 and c_2 (depending on ω_1, ω_2, z) such that $c_1 \log M_1 + c_2 \log M_2 \ge 0$.

- M_1 is typically much less than 1 $\Rightarrow$ M_2 must be larger than 1 (since sum is positive)
- If we increase performance in active range (make M_1 and H_{er} smaller), we must lose performance (H_{er} increases) some place else [waterbed effect]

Example: Ducted fan lateral position control

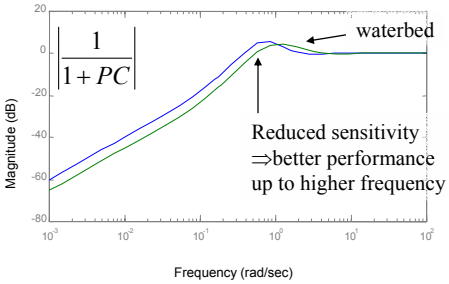


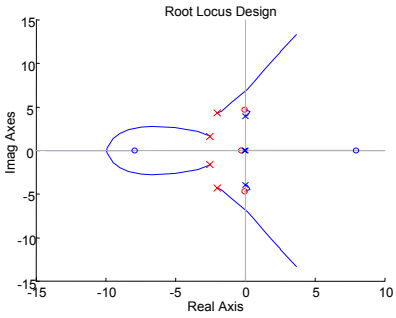


$$P(s) = \frac{(s^2 - mgl)}{s^2(Js^2 + ds + mgl)}$$

Lateral control (*x*)

- Right half plane zero makes design *very* tricky using y as output





25 Nov 02 R. M. Murray, Caltech CDS 13

Summary: PID and Root Locus

PID control design

- Very common (and classical) control technique
- Good tools for choosing gains

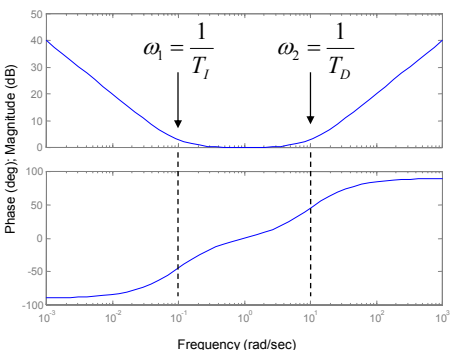
$$u = K_p e + K_I \int e + K_D \dot{e}$$

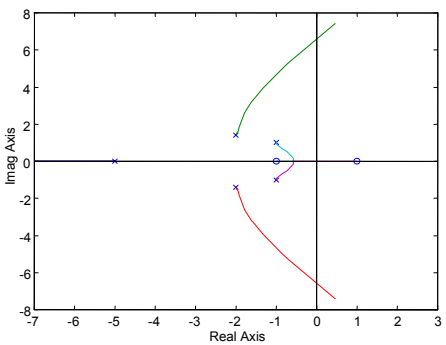
Root locus

- Show closed loop poles as function of a free parameter

Performance limits

- RHP poles and zeros place limits on achievable performance
- Waterbed effect





25 Nov 02 R. M. Murray, Caltech CDS 14