Amme 3500: System Dynamics and Control

Root Locus Design

Dr Ian R. Manchester
# Course Outline

<table>
<thead>
<tr>
<th>Week</th>
<th>Content</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Introduction</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Frequency Domain Modelling</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Transient Performance and the s-plane</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Block Diagrams</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Feedback System Characteristics</td>
<td>Assign 1 Due</td>
</tr>
<tr>
<td>6</td>
<td>Root Locus</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>Root Locus 2</td>
<td>Assign 2 Due</td>
</tr>
<tr>
<td>8</td>
<td>Bode Plots</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>Bode Plots 2</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>State Space Modeling</td>
<td>Assign 3 Due</td>
</tr>
<tr>
<td>11</td>
<td>State Space Design Techniques</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>Advanced Control Topics</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>Review</td>
<td>Assign 4 Due</td>
</tr>
</tbody>
</table>
Cascade compensation: PID Controller

- In lecture 5 we had a brief introduction to the classic PID controller.
- We will now re-examine the design of these controllers in light of the root locus techniques we have studied.
- The complete three-term controller is described by

\[ u(t) = K_p e + K_I \int_0^t e(\tau) d\tau + K_d \dot{e} \]
Designing Control Systems

• We have seen how we can use the Root Locus for designing systems to meet performance specifications
• The root locations are important in determining the nature of the system response
• By manipulating the System Gain K we showed how we could change the closed loop transient response
• What if our desired performance specification doesn’t exist on the Root Locus?
Reminder: Time Response

2nd Order System

<table>
<thead>
<tr>
<th>$\zeta$</th>
<th>Poles</th>
<th>Step response</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>$s$-plane $+ j\omega_n$, $-j\omega_n$</td>
<td>$c(t)$</td>
</tr>
<tr>
<td>$0 &lt; \zeta &lt; 1$</td>
<td>$s$-plane $+ j\omega_n \sqrt{1 - \zeta^2}$, $-j\omega_n \sqrt{1 - \zeta^2}$</td>
<td>$c(t)$</td>
</tr>
<tr>
<td>$\zeta = 1$</td>
<td>$s$-plane $- j\omega_n$</td>
<td>$c(t)$</td>
</tr>
<tr>
<td>$\zeta &gt; 1$</td>
<td>$s$-plane $- j\omega_n - \omega_n \sqrt{\zeta^2 - 1}$, $-\zeta\omega_n + \omega_n \sqrt{\zeta^2 - 1}$</td>
<td>$c(t)$</td>
</tr>
</tbody>
</table>
Specify what the transient system specifications can be formulated in the s-plane.

\[ T_r \leq 0.6 \text{ sec} \quad \rightarrow \quad \omega_n \geq \frac{1.8}{t_r} = 3.0 \text{ rad/sec} \]

\[ M_p \leq 10\% \quad \rightarrow \quad \zeta \geq 0.6 \]

\[ T_s \leq 3 \text{ sec} \quad \rightarrow \quad \sigma \geq \frac{4}{3} = 1.5 \]
Specifications in the s-plane

• Imagine a situation in which we require a particular overshoot and settling time
• For the situation shown here, we can achieve response A by adjusting the system gain
• Response B cannot be achieved with a simple change in gain

We will now look at the derivative term which can be used to change the transient response:

\[ u(t) = K_p e + K_d \dot{e} \]

This is called an **Ideal Derivative Compensator**.
PD Controller

• The ideal derivative compensator adds a pure differentiator to the forward path of the control system

• This is effectively equivalent to an additional zero

\[ U(s) = K(s + z_c) \]

• As you should by now be aware, the location of the open loop poles and zeros affects the root locus and hence the transient response of the closed loop system
Consider a simple second order system whose root locus looks like this (roots -1, -2).

Adding a zero to this system drastically changes the shape of the root locus.

The position of the zero will also change the shape and hence the nature of the transient response.
PD Controller

Uncompensated system

Zero at -2

PD Controller


Zero at -3
Zero at -4
PD Controller

- As the zero moves to the left in the s-plane, it has less effect on the transient response of the system.
- In the limit as the zero moves left, the response will be identical to the uncompensated system.

PD Controller Design

• It’s clear that the nature of the response will change as a function of the zero location
• How can we use this information to help us design for a particular specification?
• We can select the location of the additional zero based on the angle criteria to meet our design goal
PD Controller Design

• Given the system shown on the right, we’d like to design a compensator to yield a 16% overshoot with a 3 fold reduction in settling time.

• The settling time is

\[ T_s = \frac{4}{\sigma} = \frac{4}{1.205} = 3.320s \]
PD Controller Design

• Based on this root location, we can find the desired location for the compensated poles

• We’d like to reduce settling time by 1/3 so

$$\sigma = \frac{4}{T_s} = \frac{4}{1.107} = 3.613$$
PD Controller Design

- By the angle criteria
  \[ \theta_c - 120.26 - \theta_1 - \theta_2 = (2k + 1)180^\circ \]
  \[ \theta_c = 95.6^\circ \]

- Based on this we can compute the required zero location, \( z_c \)

\[
\tan^{-1}\left( \frac{6.193}{3.613 - z_c} \right) = 180^\circ - 95.6^\circ
\]

- so \( z_c = 3.006 \)

PD Controller Design

• What value of K will put the roots at the desired location?
• Recall that the RL is the location of the closed loop poles but is based on the open loop locations

\[ 1 + KG = 0 \]

• We can solve this for K for the compensated system given the desired root location
• So

\[ K = \frac{1}{|G|} \]
For our system, we have

\[ K = \frac{1}{s + 3.006} \frac{s(s + 4)(s + 6)}{s + 3.006} \mid_{s = -3.613 + 6.193 j} \]

\[ = \frac{s(s + 4)(s + 6)}{s + 3.006} \mid_{s = -3.613 + 6.193 j} \]

\[ = \frac{(-3.613 + 6.193 j)(0.387 + 6.193 j)(2.387 + 6.193 j)}{(-0.608 + 6.193 j)} \]

\[ = 47.45 \]
• The resulting root locus takes on the form shown here
• We need to verify the second order assumption on which this design was made
• It appears that the third pole is not far enough to the left, although it is close to the zero

Dr Ian R. Manchester
Amme 3500 : Root Locus Design

PD Controller Design

- A simulation of the resulting step response is required to verify the design.
- The resulting step response demonstrates a threefold improvement in settling time with a 3% difference in overshoot.

Lead Compensation

• In practice, we often don’t want to implement a pure differentiator
• A pure PD controller requires active components to realize the differentiation
• The differentiation will also tend to amplify high frequency noise
Lead Compensation

• For compensation using passive components, a pole and zero will result

\[ U(s) = \frac{K(s + z_c)}{(s + p_c)} z_c < p_c \]

• If the pole position is selected such that it is to the left of the zero, the resulting compensator will behave like an ideal derivative compensator

• The name Lead Compensation reflects the fact that this compensator imparts a phase lead
Lead Compensation

• In this case there won’t be a unique zero location that yields the desired system response
• Selecting the exact values of the pole and zero location is usually an iterative process
• Each iteration considers the performance of the resulting system
• The choice of pole location is a compromise between the conflicting effects of noise suppression and compensation effectiveness
Lead Compensation Example

- Find a compensation for $$G(s) = \frac{1}{s(s+1)}$$
- That will provide overshoot of no more than 20% and a rise time no more than 0.25s
- Noise suppression requirements require that the lead pole be no larger than 20
Lead Compensation Example

- First find a point in the s-plane that we’d like to have on the root locus: $-3.5 + j3.5\sqrt{3}$
- Place the lead pole at -20 and solve for the position of the zero:

$$D(s) = \frac{s+5.4}{s+20}$$

θ₁=120
θ₂=112.4
θ₃=20.2
θₙ=72.6
Lead Compensation Example

Dr Ian R. Manchester Amme 3500 : Root Locus Design Slide 27
Lead Compensation Example

- We could also have placed the pole at -15, although this is likely to affect the 2\textsuperscript{nd} order assumption
  \[-3.5 + j3.5\sqrt{3}\]
- Place the lead pole at -15 and solve for the position of the zero
  \[D(s) = \frac{s+4.54}{s+15}\]

\[\theta_1 = 120\]
\[\theta_2 = 112.4\]
\[\theta_3 = 27.8\]
\[\theta_c = 80.2\]
Lead Compensation Example
Steady State Error

• We also looked at how the system type is related to the steady state error of the system for a set of input types
• Systems may not meet our desired steady state error requirements based purely on a proportional controller
• Additional poles at the origin may therefore be required to change the nature of the steady state system response
Steady State Error

\[ -\theta_1 - \theta_2 - \theta_3 = (2k + 1)180^\circ \]

\( (a) \)

\[ -\theta_1 - \theta_2 - \theta_3 - \theta_c \neq (2k + 1)180^\circ \]

\( (b) \)
Steady State Error

- If we add a zero close to the pole at the origin the root locus reverts to approximately the same as the uncompensated case.
- The system type has been increased without appreciably affecting the transient response.

• We will now look at the integral term used to eliminate steady state error

\[ u(t) = K_p e + K_i \int_0^t e(\tau) \, d\tau \]

• This is called an \textit{Ideal Integral Compensator}
PI Controller

• If we rewrite the transfer function for the integral compensator we find

\[ U(s) = K_p + \frac{K_I}{s} = \frac{K(s + z_c)}{s} \]

• This is simply a pole at the origin and a zero at some other position

• The zero can be selected based on our design requirements – normally close to the origin to minimize the angular contribution
• Given the system shown here we’d like to add a PI controller to eliminate the steady state error

• We find that

\[ e(\infty) = \frac{1}{1 + K_p} = \frac{1}{1 + 8.23} = 0.108 \]
PI Controller Example


Dr Ian R. Manchester
Amme 3500 : Root Locus Design
Slide 36
Lag Compensation

• As with the lead compensation, using passive components results in a pole and zero

\[ U(s) = \frac{K(s+z_c)}{(s+p_c)}, \quad z_c > p_c \]

• If the pole position is selected such that it is to the right of the zero near the origin, the resulting compensator will behave like an ideal integral compensator although it will not increase the system type

• The name Lag Compensation reflects the fact that this compensator imparts a phase lag
Lag Compensation

• How does the lag compensator affect the steady state performance?

• If we have a plant of the form

$$G(s) = \frac{K(s + z_1)(s + z_2)\cdots}{(s + p_1)(s + p_2)\cdots}$$

• The static error constant will be

$$K_p = \lim_{s \to 0} G(s) = \frac{Kz_1z_2\cdots}{p_1p_2\cdots}$$
Lag Compensation

- Adding the lag compensator will yield the following static error constant
  \[ K_{pc} = \frac{(Kz_1z_2 \cdots)z_c}{(p_1p_2 \cdots)p_c} \]

- The improvement in the steady state error will be approximately
  \[ K_{pc} = K_p \frac{z_c}{p_c} \]

- In order for this ratio to be significant with a close pole and zero, these should be close to the origin
Lag Controller Example

- Given the system shown here
- We’d like to reduce the steady error by a factor of 10 (remember that the steady state error won’t be eliminated in this case)


\[
\begin{align*}
R(s) & \rightarrow E(s) \rightarrow K \rightarrow \frac{1}{(s+1)(s+2)(s+10)} \rightarrow C(s)
\end{align*}
\]
Lag Controller Example

• Previously we found the system error to be 0.108 with $K_p=8.23$. For a 10 fold improvement

$$e(\infty) = \frac{0.108}{10}$$

• This requires $K_{pc}=91.6$. This implies that

$$\frac{z_c}{p_c} = \frac{K_{pc}}{K_p} = \frac{91.6}{8.53} = 11.1$$

• We therefore require a ratio of approximately 11 between the pole and zero location

• Selecting $p_c=0.01$, $z_c$ will be 0.111
Lag Controller Example

Controller Design

• The design of a PID or Lead/Lag controller consists of the following 8 steps
  – Evaluate the performance of the uncompensated system to determine what improvement in transient response is required
  – Design the PD/Lead controller to meet the transient response specifications
  – Simulate the system to be sure all requirements are met
  – Redesign if simulation shows that requirements have not been met
PID Controller Design

- Design the PI/Lag controller to yield the required steady-state error
- Determine the gains required to achieve the desired specification
- Simulate the system to be sure that all requirements have been met
- Redesign if simulation shows that requirements have not been met
MATLAB Case Studies

\[ J\ddot{\theta} = u \]

\[ G(s) = \frac{1}{Js^2} \]
MATLAB Case Studies

• Aluminium extraction from Bauxite
• slow multi-compartment dynamics

\[ G(s) = \frac{1}{(s + 1)(s + 0.2)(s + 0.1)} \]
MATLAB Case Studies

- F/A-18 on landing approach, 140 knots.

\[
G(s) = \frac{\alpha(s)}{\delta_e(s)} = 0.072 \frac{(s + 23)(s^2 + 0.05s + 0.04)}{(s - 0.7)(s + 1.7)(s^2 + 0.08s + 0.04)}
\]
Design Example

• The University of Michigan has a very nice set of design examples for a variety of systems
• Have a look at the following URL for details
• http://www.engin.umich.edu/group/ctm/index.html
Conclusions

• We have presented design methods based on the root locus for changing the characteristics of the system response to meet our specifications
• There are many cases in which simple gain adjustment will not be sufficient to meet the specifications
• In these cases, we must introduce additional dynamics into the system to meet our performance requirements
Further Reading

• Nise
  – Sections 9.1-9.6

• Franklin & Powell
  – Section 5.5-5.6