Very simple control
We assume that everything is linear

• This creates huge mathematical simplifications
• Linear system:
  • accepts a signal \( x(t) \)
  • produces a signal \( y(t) = K x(t) \)
  • \( K (x(t) + y(t)) = K x(t) + K y(t) \)
  • \( K (a x(t)) = a K x(t) \)
  • (notice this means \( K 0 = 0 \))

\( K \) stands for a linear operator, so that (for example) we could have

\[ K x(t) = a x(t) \]

or

\[ K x(t) = \frac{dx}{dt} \]
In fact, study only the response to a step

- You can approximate any function with a lot of steps
- Step is \( u(t) \)
  - this is 0 for \( t \leq 0 \), 1 otherwise
  - so \( u(t) - u(t+dt) \) is a bar
- Approximate \( f(t) \) by

\[
\sum_i f(i\Delta t)(u(i\Delta t) - u(i\Delta t + \Delta t))
\]

- ex: simplify this expression
- ex: we know \( K \ u(t) \) - what is \( K \ f(t) \)?
Ideas: plant/process, control

• Plant/process is the thing we wish to control
  • assume: 1 input, 1 output, linear
  • for simple examples, I’ll write out the form of the plant
    • but very often, it isn’t known exactly
      • System Identification

• Control:
  • supply the plant with the input needed to produce the output you want
  • Q: why is this hard?
    • A1: Plant may not be exactly known
    • A2: Plant may have dynamics
    • A3: Desired output may change
The very simplest control

• Plant: \( K x(t) = c x(t) \)
  • here \( c \) is a known constant

• We’d like the output to be 1
  • feed plant with \( 1/c \)
    • and go home early

• Example of open loop control
  • compute a fixed input and supply to plant
    • whatever the plant

• Advantages:
  • simple, sometimes works

• Disadvantages:
  • what if your model is wrong?
History of feedback

Watt’s flyball governor, C19

These were still in use in late C20!
Closed loop control

• Derive an input to the plant from
  • setpoint (where you want the output to be)
  • current plant output

• The form we will discuss is:
We have

c(t) = G (i(t) - o(t))
o(t) = H c(t)

so

o(t) + H G o(t) = H G i(t)

which you should remember
Simple, worrying example

- \( H \ c(t) = a \ c(t) \)
- \( G \ x(t) = b \ x(t) \)
- \( o(t)+ab \ o(t)=ab \ i(t) \)
- \( o(t)= ab/(1+ab) \)
- which isn’t what we wanted
- (remember, \( i(t) \) is the output value we want)
- steady state error is \( \lim \ t->\infty \ (o(t)-i(t)) \)
Fix with integral term

- Idea:
  - if \((i(t)-o(t))\) is not zero, there should be some control input
  - magnitude increases until it is zero

\[
Gx(t) = bx(t) + c \int_0^t x(s) ds
\]
Fixing with integral term

\[ o(t) + abo(t) + ac \int_0^t o(s)ds = abi(t) + ac \int_0^t x(s)ds \]

Differentiate

\[ (1 + ab) \frac{do(t)}{dt} + aco(t) = ab \frac{di(t)}{dt} + aci(t) \]

BUT we’re interested in \( t > 0 \), and \( i(t) \) is a step at 0

\[ (1 + ab) \frac{do(t)}{dt} + aco(t) = aci(t) \]
Fixing with integral term

\[(1 + ab) \frac{do(t)}{dt} + aco(t) = ac\]

Assume that \(\frac{do}{dt} \to 0\) as \(t \to \infty\)
(we’ll see it does in a moment)

\[o(t) = 1\]

For large \(t\), which is what we wanted
Fixing with integral term

\[ \frac{do}{dt} + \frac{ac}{1 + ab} o(t) = 1 \quad \quad o(0) = 0 \]

\[ o(t) = (1 - e^{\frac{-ac}{1 + ab} t}) \]
Example

- is it a good idea to get a faster response by making c bigger?
A more interesting plant

\[ v(t) = v(0) + \int_0^t \frac{F(s)}{m} \, dt \]

- Apply a force to the car to control its velocity
  - eg braking

\[ v(t) = \int_0^t \frac{F(s)}{m} \, dt \]
Proportional control

\[ o(t) + H \ G \ o(t) = H \ G \ i(t) \]

\[ Gx(t) = bx(t) \]

\[ o(t) + H [bo(t)] = H [bi(t)] \]

\[ o(t) + \frac{b}{m} \int_{0}^{t} o(s)ds = \frac{b}{m} \int_{0}^{t} i(s)ds \]

\[ \frac{do}{dt} + \frac{b}{m} o(t) = \frac{b}{m} \]

Recall that \( t > 0, \ i(t) = 1 \)
Notice

\[ \frac{do}{dt} + \frac{b}{m} o(t) = \frac{b}{m} \]

\[ o(t) = (1 - e^{\frac{-bt}{m}}) \]

• steady state error is now zero
• larger b/m  -> faster response
  • BUT larger forces applied to car
• (obvious)  b/m <0  -> unstable behavior
• Example
Examples

Small b/m -> low rise time

- output
- proportional term
- demand
Examples

Bigger b/m -> faster rise time
Examples

Very big b/m -> fast rise time

- Output
- Proportional term
- Demand
Examples

Gigantic b/m -> integrator panics

output
proportional term
demand
Examples

Gigantic b/m, smarter integrator -> very fast rise time

output
proportional term
demand
Proportional - Integral (PI) control

\[ Gx(t) = bx(t) + c \int_0^t x(s)ds \]

\[ o(t) + H \left[ bo(t) + c \int_0^t o(s)ds \right] = H \left[ bi(t) + c \int_0^t i(s)ds \right] \]

\[ o(t) + \frac{1}{m} \int_0^t \left[ bo(u) + c \int_0^u o(s)ds \right] = \frac{1}{m} \int_0^t \left[ bi(u) + c \int_0^u i(s)ds \right] \]

\[ \frac{d^2 o}{dt^2} + \frac{b}{m} \frac{do}{dt} + \frac{c}{m} o(t) = \frac{c}{m} \quad \text{(recall } t>0, i(t)=1) \]
Assume derivatives $\rightarrow 0$ as $t \rightarrow \infty$ (we’ll see they do)
then $o(t) = 1$ for very large $t$, which is what we wanted

$$\frac{d^2 o}{dt^2} + \frac{b}{m} \frac{do}{dt} + \frac{c}{m} o(t) = \frac{c}{m}$$

$A_1 e^{zt} + A_2 t + A_3$

$$e^{zt} \left( A_1 z^2 + \frac{b}{m} [A_1 z + A_2] + A_1 \frac{c}{m} \right) + A_2 t \frac{c}{m} + A_3 \frac{c}{m} = \frac{c}{m}$$

$A_2 = 0$

$A_3 = 1$

$A_1 = -1 \quad (o(0) = 0)$

$$z^2 + \frac{b}{m} z + \frac{c}{m} = 0$$
$z = \frac{1}{2} \left[ -\frac{b}{m} \pm \sqrt{\frac{b^2}{m^2} - 4\frac{c}{m}} \right]$
Careful with b

- small c

\[ c = \epsilon \frac{b^2}{m} \]

- gives roots that are like

\[ z = \frac{1}{2} \left[ -\frac{b}{m} \pm \sqrt{\frac{b^2}{m^2} - 4\frac{c}{m}} \right] \]

\[ -\frac{b}{m} \left( 1 - \frac{\epsilon}{4} \right) \quad \text{Might be quite fast} \]

\[ -\frac{b}{m} \frac{\epsilon}{4} \quad \text{rather a lot slower} \]
Examples

PI control $m=1$, $b=10$, $c=25$
Examples

PI control $m=1$, $b=10$, $c=1$

- Output
- Integral term
- Proportional term
Examples

PI control $m=1$, $b=10$, $c=300$
Examples

PI control $m=1$, $b=10$, $c=300$ step waveform
Examples

PI control $m=1$, $b=1$, $c=300$

Output

Integral term

Proportional term
Examples

PI control $m=1$, $b=1$, $c=300$ step waveform
Critical damping occurs when there is a double root equivalent when \( \zeta = 1 \)
\( \zeta < 1 \) underdamped (soln. wobbles)
\( \zeta > 1 \) overdamped (slow rise time)
More on quadratic equations!

\[ z^2 + 2\zeta\omega z + \omega^2 = 0 \]

Damping

\[ z = -\omega \left( \zeta \pm i\sqrt{1 - \zeta^2} \right) \]

Natural frequency

Our equation

\[ z^2 + \frac{b}{m}z + \frac{c}{m} = 0 \]

\[ \omega = \sqrt{\frac{c}{m}} \]
\[ \zeta = \frac{1}{2} \frac{b}{\sqrt{cm}} \]

Critical damping:

\[ b = 2\sqrt{cm} \]
Examples

PI control critical damping $m=1, b=20, c=100$
A derivative term

• Issue:
  • may be hard to get fast rise time
    • big m requires big b for critical damping
  • this may be because we are feeding back the current error

• Idea:
  • predict future error
  • this is equivalent to feeding back some fraction of the derivative
The most important slide

- A very high fraction of all controllers in the real world are:

\[ Gx(t) = K_i \int_0^t x(u)du + K_p x(t) + K_d \frac{dx}{dt} \]

- PID controller
A more interesting plant

\[ v(t) = v(0) + \int_{0}^{t} \frac{F(s)}{m} \, dt \]

- Apply a force to the car to control its velocity
  - eg braking

\[ v(t) = \int_{0}^{t} \frac{F(s)}{m} \, dt \]
Example

PID control critical damping $m=1$, $kp=20$, $ki=100$, $kd=0$
Proportional-Integral-Derivative (PID) control

Thrash through math of PI slide, and end up with:

\[
\begin{align*}
\frac{d^2 o}{dt^2} + \frac{K_p}{m + K_d} \frac{do}{dt} + \frac{K_i}{m + K_d} o &= \frac{K_i}{m + K_d} \\
\end{align*}
\]

Compare to:

\[
\begin{align*}
\frac{d^2 o}{dt^2} + \frac{b}{m} \frac{do}{dt} + \frac{c}{m} o(t) &= \frac{c}{m} \\
\end{align*}
\]

Kd makes the mass look smaller!
Examples

PID control critical damping $m=100$, $kp=20$, $ki=100$, $kd=0$
Examples

PID control critical damping $m=100$, $kp=20$, $ki=100$, $kd=-99$
Yet more interesting plant

Apply a force to the mass, want to control its position.

\[ m \frac{d^2x}{dt^2} + b \frac{dx}{dt} + kx = F \]
Thrash through math of past slides, and end up with:

\[
\frac{d^2 o}{dt^2} + \frac{K_p + b}{m + K_d} \frac{dx}{dt} + \frac{K_i + k}{m + K_d} x = \frac{K_i + k}{m + K_d}
\]

Compare to:

\[
m \frac{d^2 x}{dt^2} + b \frac{dx}{dt} + k x = F
\]

Kd makes the mass look smaller! Kp changes the damping constant! Ki changes the spring constant!
Examples

PID control critical damping $m=1$, $b=0.01$, $c=0.01$, $kp=20$, $ki=300$, $kd=-0.9$
Examples

PID control $m=1$, $b=0.01$, $c=0.01$, $kp=20$, $ki=300$, $kd=-0.95$
Examples

PID control \( m=1, b=0.01, c=0.01, kp=20, ki=300, kd=-0.98 \)
Proportional-Integral-Derivative (PID) control

Thrash through math of past slides, and end up with:

\[
\frac{d^2 o}{dt^2} + \frac{K_p + b}{m + K_d} \frac{dx}{dt} + \frac{K_i + k}{m + K_d} x = \frac{K_i + k}{m + K_d}
\]

Compare to:

\[
m \frac{d^2 x}{dt^2} + b \frac{dx}{dt} + k x = F
\]

Kd makes the mass look smaller!  Kp changes the damping constant!  Ki changes the spring constant!
Examples

PID control \( m=10, b=0.01, c=0.01, kp=2001, ki=300001, kd=0 \)

PID control \( m=10, b=0.01, c=0.01, kp=20, ki=300, kd=-9.9 \)
Tuning

- Usually, you don’t know the plant and can’t do the math
- Powerful rule of thumb (manual tuning)

If the system must remain online, one tuning method is to first set $K_i$ and $K_d$ values to zero. Increase the $K_p$ until the output of the loop oscillates, then the $K_p$ should be set to approximately half of that value for a "quarter amplitude decay" type response. Then increase $K_i$ until any offset is corrected in sufficient time for the process. However, too much $K_i$ will cause instability. Finally, increase $K_d$, if required, until the loop is acceptably quick to reach its reference after a load disturbance. However, too much $K_d$ will cause excessive response and overshoot. A fast PID loop tuning usually overshoots slightly to reach the setpoint more quickly; however, some systems cannot accept overshoot, in which case an overdamped closed-loop system is required, which will require a $K_p$ setting significantly less than half that of the $K_p$ setting that was causing oscillation.
Tuning, II

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Rise time</th>
<th>Overshoot</th>
<th>Settling time</th>
<th>Steady-state error</th>
<th>Stability</th>
</tr>
</thead>
<tbody>
<tr>
<td>$K_p$</td>
<td>Decrease</td>
<td>Increase</td>
<td>Small change</td>
<td>Decrease</td>
<td>Degrade</td>
</tr>
<tr>
<td>$K_i$</td>
<td>Decrease</td>
<td>Increase</td>
<td>Increase</td>
<td>Eliminate</td>
<td>Degrade</td>
</tr>
<tr>
<td>$K_d$</td>
<td>Minor change</td>
<td>Decrease</td>
<td>Decrease</td>
<td>No effect in theory</td>
<td>Improve if $K_d$ small</td>
</tr>
</tbody>
</table>

Kd = 0 for about 75% of deployed systems
Stability and oscillation (rough)

- Linear systems can clearly oscillate
  - generally, too big a $K_p$ or $K_d$ can cause problems

- Nonlinearities can easily cause oscillations

- Delays cause oscillations
Examples

PI control around delay of 1e-4s, plant=1, kp=0.1, ki=5000
Examples

PI control around delay of 20e-4s, plant=1, kp=0.1, ki=5000
Examples

PI control around delay of 200e-4s, plant=1, kp=1, ki=1

- output
- i-term
- p-term
PI control around delay of 200e-4s, plant=1, kp=1, ki=10

PI control around delay of 200e-4s, plant=1, kp=1, ki=70
Ideas

- Plant/process
- control
- Open vs closed loop
- stability
- Linear vs non-linear
- Simplest linear feedback control
  - x constant
  - with derivative term
  - large gains can cause instability
  - steady state error is a problem
- Delay is a problem
- non-linearities can create excitement
Ideas

- PID control
  - standard procedure
    - (there are tons in the car software)
  - P controls; I reduces steady state error; D increases response speed
  - Straightforward tuning procedure
    - (see software example)