Basics of Feedback Control

Raj Rajkumar

Lecture 22

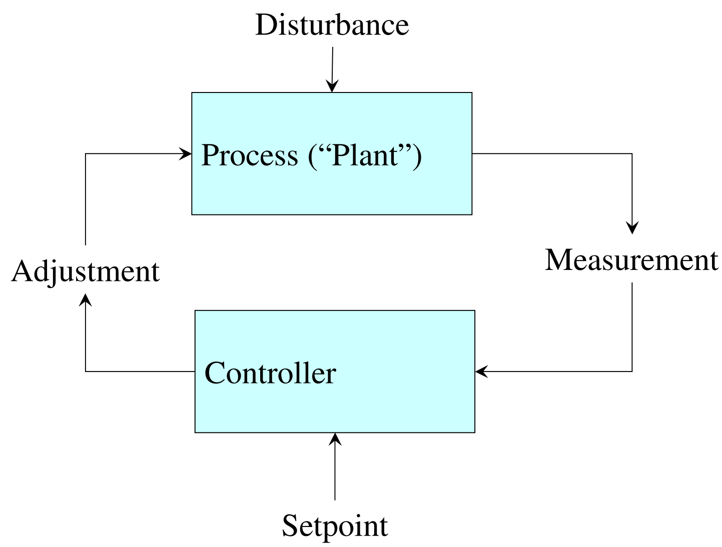
Outline of This Lecture

- The Feedback Control Problem
- Proportional Control
- Derivative Control
- Integral Control

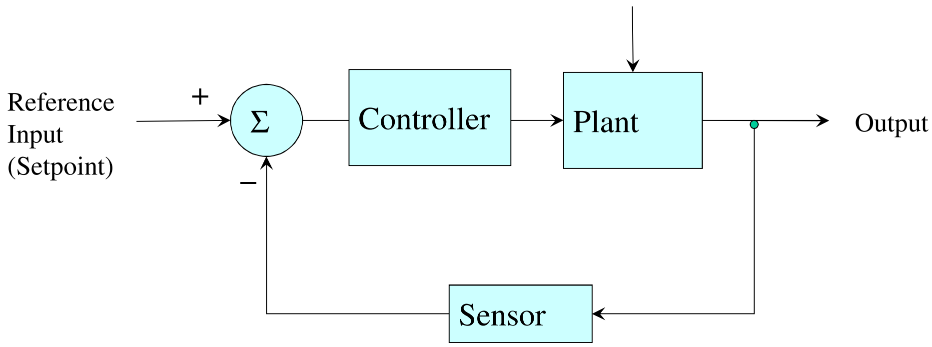
Feedback in Embedded Systems

- Many real-time embedded systems make **control decisions**
  - E.g. Elevators, automotive cruise control
- These decisions are usually made by software and based on feedback from the hardware under its control (termed the "**plant**")
  - Such feedback commonly takes the form of an analog sensor that can be read via an A/D converter
  - A sample from the sensor might represent position, voltage, temperature, or any other appropriate parameter
- Each sample provides the software with additional information upon which to base its control decisions

High-Level View of Feedback Control

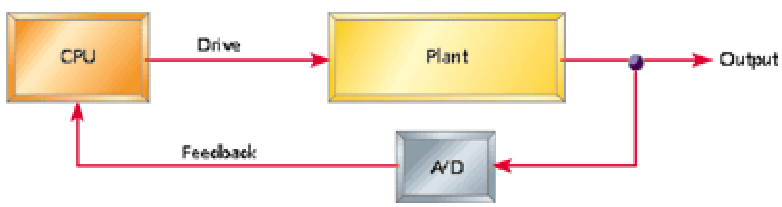


Actual Feedback Control Loop



Closed-Loop vs. Open-Loop

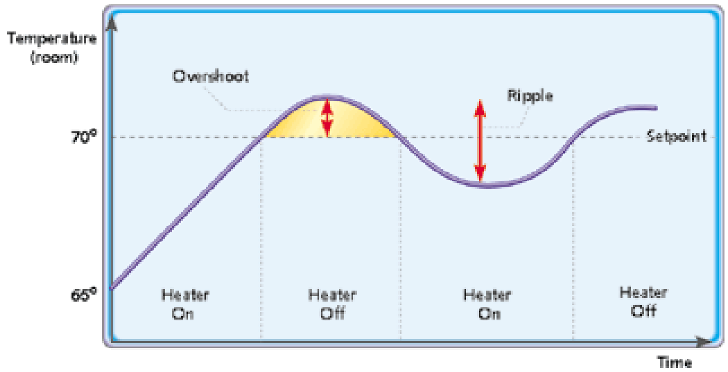
- Open-loop systems do not necessarily use feedback
- Closed-loop systems use feedback



- After reading each new sample from the sensor
  - Software reacts to the plant's changed state by recalculating and adjusting the drive signal
  - Plant responds to this change, another sample is taken, and the cycle repeats
- Eventually, the plant should reach (close to) the desired state and the software may cease making changes

On-Off Control – Your Thermostat

- How much should the software increase or decrease the drive signal?
  - E.g. Set the drive signal to its minimum value when you want the thermostat to decrease its temperature
  - Set the drive signal to its maximum value when you want the thermostat to



Proportional Control

- If the difference between the current output and its desired value (the current error) is large
  - Software should probably change the drive signal a lot
- If the error (difference) is small
  - Software should change the drive signal only a little
- We want the change to look like this
  - K_p * (desired – current)
  - K_p is called the **proportional gain**
- If K_p is chosen well,
  - Time taken to reach a new setpoint will be as short as possible, with overshoot (or undershoot) and oscillation minimized
- Is there a problem here?

Proportional-Derivative (PD) Control

- Problems with Proportional Control
  - You want to reach the desired value quickly (higher K_p)
  - You want to settle down once you are near the desired value (low K_p)
- We have (or can discover) information about the rate of change of the plant's output
  - Rate of change of a signal = its derivative
  - If the output is changing rapidly, an overshoot/undershoot may lie ahead
- We can then reduce the size of the change suggested by the proportional controller
  - Subtract a derivative-based term from the proportional term
- PD Controller characteristics
  - Slower response time
  - Far less overshoot and ripple than a proportional controller

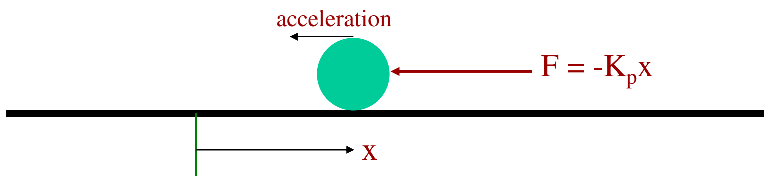
Proportional-Integral-Derivative (PID) Control

- Problems with PD Control
  - Change in drive signal triggered by error = difference in desired and current values of output
  - Derivative only tells you how two consecutive output values differ
  - What if the error is not sufficient to trigger any action?
- Something else is needed to drive the plant toward the setpoint
- **Integral term**
  - Sum of all past errors in the plant output
    - Or at least a significant recent window
  - A persistent error will eventually cause the sum to grow large enough
  - The integral term then forces a change in the drive signal
- **P**roportional, **I**ntegral and **D**erivative combination = **PID**

A Simple System with ***Proportional Control***

Consider a marble on a flat and perfectly level table.

- Any point can be an equilibrium point (just pick one)
- Motion can be described by Newton' s law **F = ma**, or **x´´ = F/m**
- Suppose that we want to keep the marble at **x = 0**,
  - By applying proportional control: $\mathbf{F = -}K_p\mathbf{x}$.
- The feedback is negative since if the marble position error is negative, it pushes with a positive force and vice versa.
- K_p is a positive integer known as ***proportional control constant***.



Concepts of Stability

- The objective of controlling the marble is to put it at rest at the origin (set point). i.e. **x = 0**, and **x' = 0**, from any initial condition.
  - Or for that matter, after the marble was disturbed by some force.
  - This intuition is formalized as the notion of **stability**
  - Or more precisely, *asymptotic stability*, i.e. the error will converge to zero over time.
- The opposite of stability is **instability**, meaning that the error will grow larger and larger without bound.
  - That is, the marble will leave the origin for good.
- In-between is **marginal stability**, the error stays within some bound.

Review

- From your calculus courses, we know that the solution of ***Linear Partial Differential Equations*** is a sum of exponentials:

  $y(t) = c_1\exp(\lambda_1 t) + c_2\exp(\lambda_2 t) + ...$

  - Exponents λ_1 , λ_1 ... are known as the **eigenvalues** of the system.
  - The eigenvalues can be complex, e.g., exp(Re(λ) t)*exp(Im(λ) jt)

- Solving LPDEs using the Laplace Transform
  - 3*y´´ + 4*y´ + 5*y = 0 (replace n^{th} derivative with s^n)
  - $3s^2 + 4s + 5 = 0$
  - Solve for *s* and you will get the eigenvalues

Stability and Eigenvalues

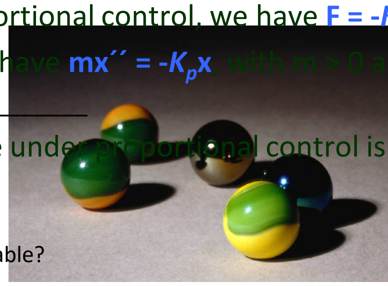
- What is the relationship between
  - The magnitude of eigenvalues and
  - The notion of (asymptotic) stability, marginal stability and instability?

- **Stability**
  - All real parts are negative
- **Instability**
  - At least one eigenvalue's real part is greater than zero.
- **Marginal stability**
  - None greater than zero but some eigenvalues equal to zero.

Back To The Marbles

- Now, let us analyze the stability of our marble under proportional control.
- From Newton's law, we have **$F = m\,x''$**
- From proportional control, we have **$F = -K_p\,x$**
- Hence, we have **$mx'' = -K_p x$** with $m > 0$ and $K_p > 0$,

  s = ______________



- The marble under proportional control is

  Stable?

  Unstable?

  Marginally stable?
- Why?

Marble under Proportional Control

- The system trajectories can be obtained by solving the differential equations by using Laplace transform

  L$(x'') = s^2$, **L**$(x') = s$, **L**$(x) = 1$

- Start with **$x'' = (-K_p/m)\,x$** ⇨

  $x'' + (K_p/m)\,x = 0$

  Apply Laplace transforms

  $s^2 + K_p/m = 0$

  $s = \pm j\,\mathrm{sqrt}(K_p/m)$ are the eigenvalues

- As you might have guessed, the marble will oscillate with frequency

  $\omega = \mathrm{sqrt}(K_p/m)$ radians/sec
- System is *marginally stable* since the real part of eigenvalue is zero

Marble Under P + D control

$F = m\,x''$ // where m is the mass of the marble

$F = -K_p x$ // proportional control

$F = -K_p x - K_d x'$ // **proportional + derivative control**

$mx'' + K_d x' + K_p x = 0$ // putting it together

$ms^2 + K_d s + K_p = 0$ // applying Laplace transforms

$s = (-K_d \pm \mathrm{sqrt}(K_d^2 - 4mK_p)) / 2m$

$m > 0$, $K_p > 0$ and $K_d > 0$

- Is this system stable? Why?
  - Since $sqrt(K_d^2 - 4m*K_p) < K_d$, the real part of s is always < 0
  - It follows that the system is stable.

Intuition

- When the position error *x* is positive, the proportional control force is negative.
  - It pushes the marble back to the origin (setpoint) at 0.
  - What will be the force and the speed when marble reaches the origin if we use only proportional control?
- When the marble is moving towards the setpoint, the velocity is negative. So the force due to derivative control $-K_d x'$ is positive.
  - This counters the proportional force and slows down the marble' s motion towards the origin.
- **Proportional control is like your car's gas pedal**
  - Moves the car towards the setpoint (a chosen origin)
- **Derivative control is like your brake**

Tuning a Simple PD Controller - I

Experimental tuning procedure

1. First set the derivative gain to zero
2. Slowly increase the proportional gain until the device oscillates around the set point with a speed that is satisfactory
   - ✓ At this point, the real part of the eigenvalue is _________
   - ✓ The imaginary part of the eigenvalue determines the __________ of oscillation
3. Slowly increase the derivative gain until the device settles down at the setpoint
   - ✓ At this point, the real part of the eigenvalue is ________

Tuning a Simple PD Controller - II

- **Fine-tuning**
  - ❖ If the motion towards the setpoint is too slow, we can __________ the proportional gain ***or*** ___________ the derivative gain
  - ❖ If the motion overshoots the setpoint too much, we can __________ the derivative gain ***or*** __________ the proportional gain
- <u>Avoid using too large a proportional gain and too large a derivative gain at the same time</u>
  - This will saturate the actuator
  - It is like slamming on your gas pedal and the brake together

Integral Control

- A system often has **friction** or **changing workloads** that you may not be able to model in advance.
- In auto-cruise control, we cannot know how many passengers will be in the car or what the current slope of the road is.
- Friction may change due to machine conditions (worn-out tires, loss of engine conditioning).
- Un-modeled heavy load often results in **steady-state error**, the system will settle "near", rather than at, the setpoint.

What Does ***Integral Control*** Do?

- Integral control adds up (integrates) the past errors and then adds a force that is ***proportional to the cumulative errors.***
- If the marble gets stuck near a setpoint due to some friction
  - The position error adds up over time
  - Use this to eventually generate a force large enough to help get the marble going.
- If the car has a heavier load and the velocity settles at a value lower than the setpoint for a while.
  - The error adds up and
  - The integral control leads to increased throttle

Using Integral Control

- Check the eigenvalue of the system to make sure all of them are sufficiently negative.
  - $X'' = F/m$, where $F = -K_p x - K_d x' - K_i * \int x$
  - $s^2 + (K_d/m) s + K_p/m + (K_i/m) 1/s = 0$ (because the Laplace transform of the integral is 1/s)
  - $s^3 + K_d/m\ s^2 + K_p/m\ s + K_i/m = 0$
- Effect of integral is to add an ***order*** to the system. A large value of K_i will lead to a positive eigenvalue (bad!)
- We can solve the equation to see if the real part of the eigenvalues are still negative (best done using tools like Matlab)

The *Dark Side* of Integral Control

- Integral control acts on *cumulative* errors
  - It takes a while to reach a large sum
  - It will also take time to reduce the sum
- Consider the following case
  - The marble gets stuck on the left side of the set-point
  - After 10 sec, the integral control is large enough to help get the marble moving
  - The integral will keep increasing until the marble crosses the origin
  - It will take a while to “wash out” the cumulative error
- An overdose of integral control is a common source of **overshoot**, **oscillation** and even **instability**.
  - What can you do to offset this?
- As a rule of thumb, start from zero and use it lightly

How to Tune a Simple PID Controller

Experimental tuning procedure

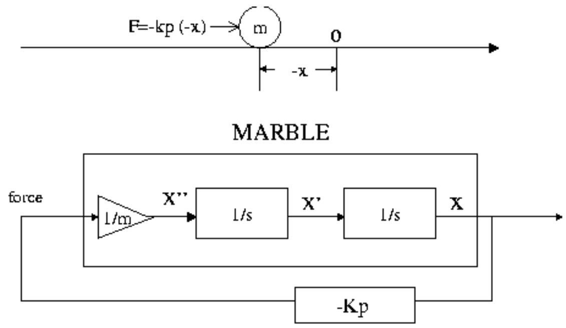
- First set derivative gain and integral gains to zero.
- Slowly increase the proportional gain until the device oscillates around the set point with a speed that is satisfactory.
  - At this point, the real part of the eigenvalue is _________
  - The imaginary part of the eigenvalue determines the __________ of oscillation
- Slowly increase the derivative gain until the device settles down at the setpoint
  - At this point, the real part of the eigenvalue is _____________
- If there is a steady-state error, slightly increase the integral gain until the steady state error is corrected and yet does not cause serious oscillation.
  - This means that the real part of the eigenvalues are still _______

How to Tune a Simple PID Controller

- **Fine-tuning**
  - If the motion towards the setpoint is too slow
    - We can ___________ the proportional gain or ____________ the derivative gain
    - Do *not* play with the integral gain
  - If there is a steady-state error
    - We can add a little ___________ gain
  - If the motion overshoots the setpoint and oscillates
    - We can ___________ the derivative gain or
    - We can reduce the ___________ gain and the ___________ gain

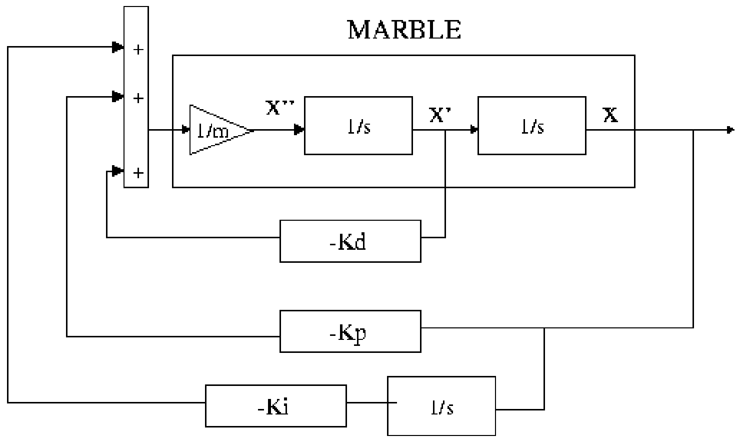
Basic Simulation Diagram Concepts

Proportional Control of Marble



Mass is a double-integrator with weight $1/m$, which transforms force to acceleration, acceleration to velocity, and then velocity to position. ($1/s$ denotes integration)

Marble Under PID Control



Cruise Control – PID In Action

- Cruise control controls the speed of the car by adjusting the throttle position (throttle valve controls engine power and speed)
- A good cruise control system would accelerate aggressively to the desired speed without overshooting, and then maintain that speed with little deviation, even if you go over a hill
- **Proportional control**
  - Difference in ideal and current speeds dictates proportional control
  - Cruise control set at 60mph; car traveling at 50mph has twice the throttle opening than when it gets to 55mph
- **Derivative control** *(time derivative of speed is acceleration)*
  - While driving downhill, proportional control can lead to significant overshoot
  - If speed changes too rapidly, decrease throttle opening correspondingly
- **Integral control** (*time integral of speed is distance*)
  - Helps the car deal with hills, and settle into the correct speed
  - If car goes up a hill, proportional control helps, but may still slow the car down; the longer the car stays at a slower speed, the more the integral control, and therefore the throttle opening increases

Summary of Lecture

- Feedback control
  - stability
  - instability
  - marginal stability
- Feedback controllers
  - proportional control
  - proportional + derivative control
  - proportional + derivative + integral control
- Simulation diagram concepts