EECS 192: Mechatronics Design Lab

Discussion 6: Control Responses & Tuning

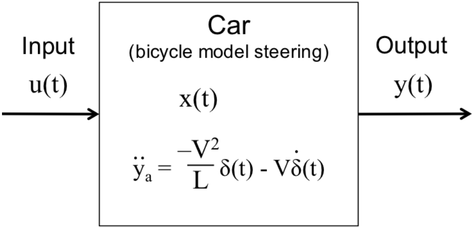
GSI: Andrew Barkan

24 Feb 2021 (Week 6)

Dynamical Systems Review

Dynamical Systems

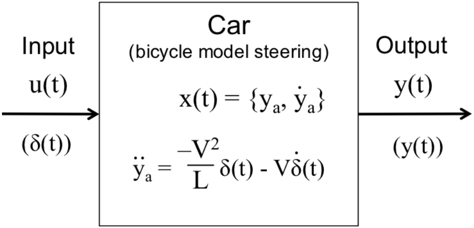
- The car's behavior is described by **state variables** (e.g. position, velocity)
- Actuators accept **inputs** and sensors read **outputs** (e.g. PWM, camera line)



Car System

Dynamical Systems

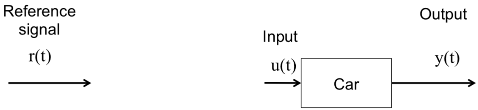
- The car's behavior is described by **state variables** (e.g. position, velocity)
- Actuators accept **inputs** and sensors read **outputs** (e.g. PWM, camera line)



Car System

Dynamical Systems

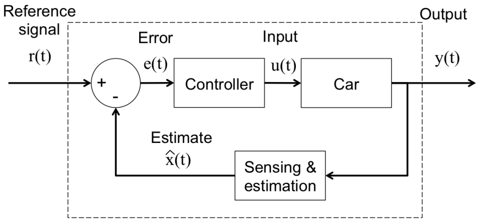
- The car's behavior is described by **state variables** (e.g. position, velocity)
- Actuators accept **inputs** and sensors read **outputs** (e.g. PWM, camera line)
- It should follow a **reference** (lateral displacement & track, velocity & setpoint)



Reference

Dynamical Systems

- The car's behavior is described by **state variables** (e.g. position, velocity)
- Actuators accept **inputs** and sensors read **outputs** (e.g. PWM, camera line)
- It should follow a **reference** (lateral displacement & track, velocity & setpoint)
- Our controller feeds **inputs** to the system to achieve this (servo angle, motor PWM)



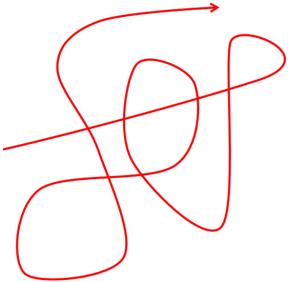
Closed Loop System

Dynamical Systems

- It is often nice to work with **Linear Time-Invariant Systems**
  $\dot{x}(t) = Ax(t) + Bu(t)$
  $y(t) = Cx(t) + Du(t)$
- When systems are nonlinear, we can create linear approximations about operating points
  (what is $\ddot{y} = \frac{-V^2}{L}\delta(t) - V\dot{\delta}(t)$?)
- Keep in mind the approximations get worse as we get further from our operating point

Control Tuning

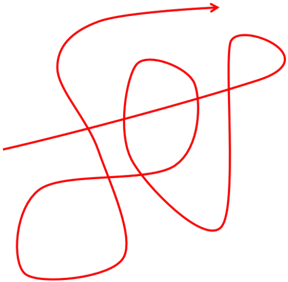
- We'd like to tune our controller to keep the car close to the track while it runs as *fast as possible*.
- How do we systematically examine how it's doing?



What to do about my crazy car?

Control Tuning

- We'd like to tune our controller to keep the car close to the track while it runs as *fast as possible*.
- How do we systematically examine how it's doing?
  - Frequency response
  - Impulse response
  - Step response

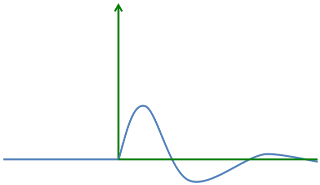


What to do about my crazy car?

Control Response

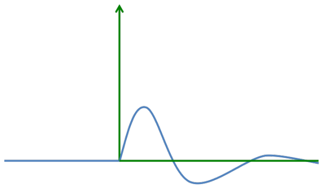
Responses

- Impulse response is very useful for analyzing the system



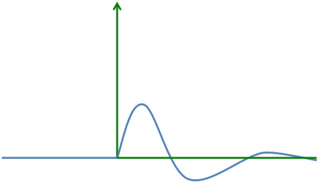
Responses

- Impulse response is very useful for analyzing the system
  - Time trace after Dirac delta input
  - Time-domain equivalent to Laplace domain transfer function



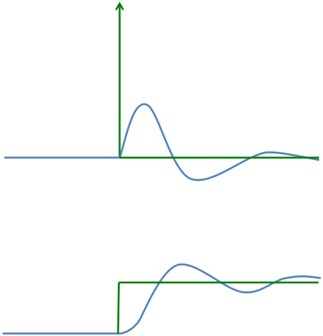
Responses

- Impulse response is very useful for analyzing the system
  - Time trace after Dirac delta input
  - Time-domain equivalent to Laplace domain transfer function
  - But we can't give our car an infinite input (and if we could we wouldn't want to)



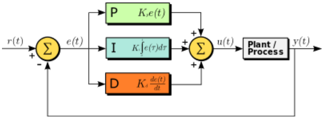
Responses

- Impulse response is very useful for analyzing the system
  - Time trace after Dirac delta input
  - Time-domain equivalent to Laplace domain transfer function
  - But we can't give our car an infinite input (and if we could we wouldn't want to)
- Instead, we often use step responses

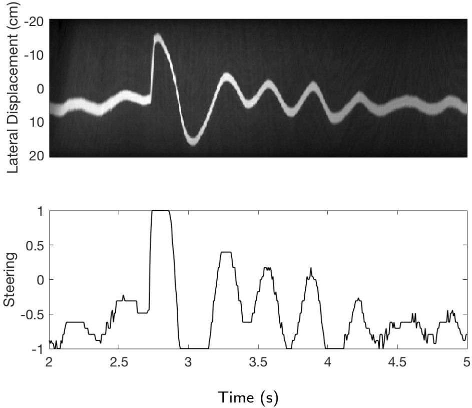


Step Response

- We can use the step response to tell how we should adjust PID gains
- **General tips**:
  - Often easiest to begin with only proportional feedback, then add other terms later
  - Higher gains improve tracking, but ...
  - Extremely high gains cause jittering and shaking (or even instability)
  - Time delay causes instability

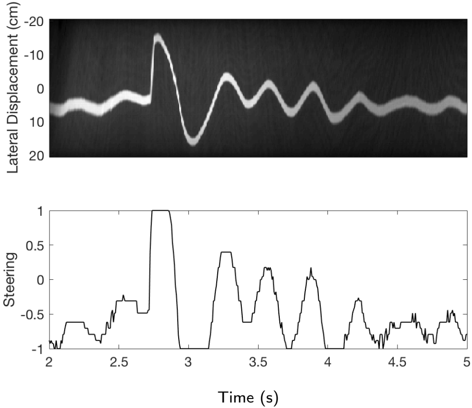


Step Response



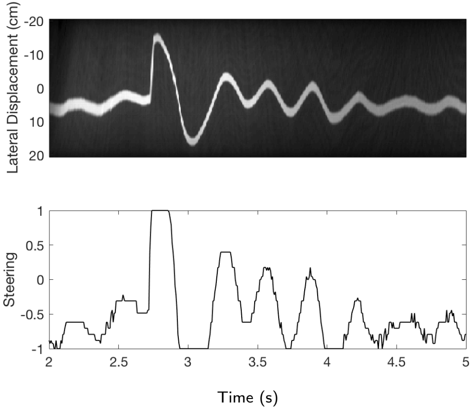
Step Response

- Note the steering saturation. Is the linear approximation good?

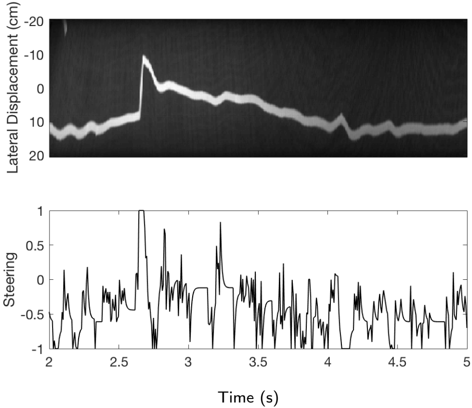


Step Response

- Note the steering saturation. Is the linear approximation good?
- P is too high

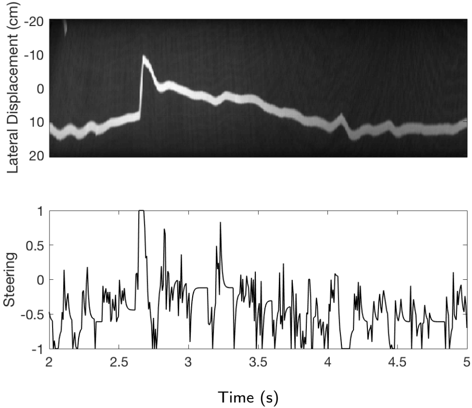


Step Response

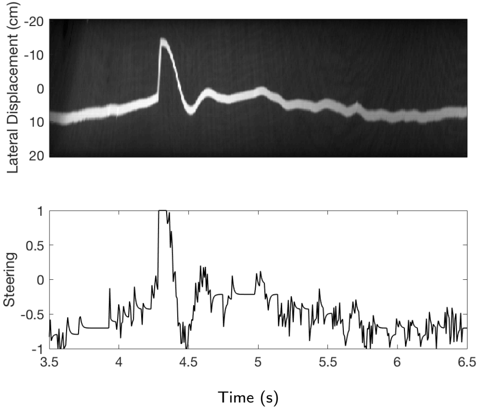


Step Response

- Note the jittery steering input
- P 1/2 of before, but D is too high

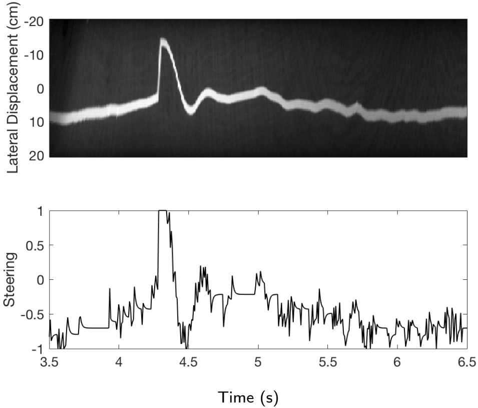


Step Response



Step Response

- Looking better
- P 3/4 of original, D 1/3 of earlier



Summary

- Save telemetry data!
- Look at step responses and control signals

Step Response and Telemetry Demo

Let's take a look at a demo where we implement simple model-free steering control and accompanying telemetry implementation!

BTW... it's not a very good controller.