Sensing, Actuation, Control

ENES 100, Fall 2006

Prof. Bruce Jacob
Electrical & Computer Engineering
(with enormous thanks to Prof. Bill Levine)

OUTLINE:

• Some example control systems
• Feedback: Open loop vs. closed loop
• Proportional, Integral, Derivative (PID) controllers
• Hovercraft control issues
The Toilet

- 2000 year old control system
- System not used for present purpose until 19th century (cholera epidemics)

- Sensor: float
- Actuator: valve
- Power: water level
- Failsafe: overflow tube
The Toilet

- 2000 year old control system
- System not used for present purpose until 19th century (cholera epidemics)

• Sensor: float
• Actuator: valve
• Power: water level
• Failsafe: overflow tube
Power Brakes (e.g. disk)

- **Manual activation**
- **Separate hydraulic networks** (per brake or per opposite pair)
- **Additional failsafe (optional):** power needed to hold brake open (fails closed)

- Sensor: foot pedal
- Actuator: brake calipers
- Power: hydraulic
- Failsafe: dual system
Power Brakes (e.g. disk)

- Manual activation
- Separate hydraulic networks (per brake or per opposite pair)
- Additional failsafe (optional): power needed to hold brake open (fails closed)

- Sensor: foot pedal
- Actuator: brake calipers
- Power: hydraulic
- Failsafe: dual system
Antilock Brakes

- Each wheel monitored separately for significant deviation in wheel speed
- Each wheel controlled/pulsed separately
- Problem: contaminated sensors
- Add’l sensors: wheel angle & gyroscope

- Sensor: wheel speed
- Actuator: pulse emitter
- Power: hydraulic
- Failsafe: manual, sensors
Antilock Brakes

- Each wheel monitored separately for significant deviation in wheel speed
- Each wheel controlled/pulsed separately
- Problem: contaminated sensors
- Add’l sensors: wheel angle & gyroscope

- Sensor: wheel speed
- Actuator: pulse emitter
- Power: hydraulic
- Failsafe: manual, sensors
Fletched Arrow

- Bare shaft: completely unstable
- Weighted tip: slightly more stable
- Fletching acts as control mechanism (correction proportional to deviation)

- Sensor: fletching
- Actuator: fletching
- Power: pressure
- Failsafe: n/a
Fletched Arrow

- Bare shaft: completely unstable
- Weighted tip: slightly more stable
- Fletching acts as control mechanism (correction proportional to deviation)

- Sensor: fletching
- Actuator: fletching
- Power: pressure
- Failsafe: n/a
Steam Valve

- Sensor: spring-loaded piston
- Actuator: valve
- Power: (steam) pressure
- Failsafe: backup/none

- Plug/spring acts as control mechanism (correction proportional to deviation: higher pressure => valve opens more)
Steam Valve

- Plug/spring acts as control mechanism (correction proportional to deviation: higher pressure => valve opens more)

- Sensor: spring-loaded piston
- Actuator: valve
- Power: (steam) pressure
- Failsafe: backup/none
Centrifugal Governor

- Sensor: centrifugal pendulum
- Actuator: valve
- Power: torque on shaft
- Failsafe: backup/none
Centrifugal Governor

- Also called the “flyball” governor
- Proportional control: the faster the rotation, the more the valve closes
- On nearly every steam engine made

- Sensor: centrifugal pendulum
- Actuator: valve
- Power: torque on shaft
- Failsafe: backup/none
Feedback Control

OPEN LOOP
- Power Brakes
- Power-Assist Steering
- Manual Throttle

CLOSED LOOP
- Anti-Lock Brakes
- Compass-Assisted Steering
- Governor-Controlled Throttle
Power Brakes

OPEN LOOP

Input to System → Controller (Software/Hardware) → Plant (Thing being Controlled) → System Output

Foot on Pedal → Magnify Force → Brake System → Car Slows
Anti-Lock Brakes

CLOSED LOOP

- Input to System: Foot on Pedal
- Controller: Even or Pulsed Force
- Plant: Individual Brakes per Wheel
- System Output: Car Slows

Feedback: Wheel rotation speed
Are any wheels slipping? Are any wheels rotating much more slowly than the others? If so, slipping brakes are pulsed to try to recover traction.
Power-Assist Steering

OPEN LOOP

Input to System → Controller (Software/Hardware) → Plant (Thing being Controlled) → System Output

Torque on Steering Wheel → Magnify Force (gain is variable, rel. to veh. speed) → Hydraulic Steering System → Car Turns

Vehicle Speed
Compass-Assisted Steering

CLOSED LOOP

Input to System → Controller → Plant → System Output

Software/Hardware

Thing being Controlled

Feedback

Compass Direction (desired) → Appropriate Degree of Turning → Steering System → Vehicle Turns

Feedback: Current vehicle direction of travel
If direction of vehicle is not equal to the desired compass point, control system adjusts steering appropriately (note: vehicle can point one way and go another)
Manual Throttle

OPEN LOOP

Input to System → Controller
Software/Hardware

Plant
Thing being Controlled

System Output

Set Throttle → Open or Close Valve

Engine Fuel Line
Engine Speeds Up (Slows Down)
Governor-Controlled Throttle

CLOSED LOOP

Feedback: Engine shaft rotation speed
If load on engine increases, the rotation slows, causing the governor to open the throttle.
PID Controller

Proportional, Integral, Derivative

- **Proportional** term ensures the system reacts as soon as there is a change in the system: change in new output follows the error.
- **Integral** term provides hysteresis, tracks effectiveness of control system: measures delta between output and input to date.
- **Derivative** term anticipates future behavior: reacts to quick changes in plant output vs. input.

\[
u(t) = K_P e(t) + K_I \int_0^t e(t) \, dt + K_D \frac{de}{dt}
\]
Example System

Thermostat — A Popular Controls Example

- Water heater: controlled by voltage
- Sensor: temperature (V representing T)
Example System

Thermostat — A Popular Controls Example

- Response of system to step input
Proportional Controller

\[
\text{while (1)} \\
\quad \text{error} = \text{desired()} - \text{reading()}; \\
\quad \text{increase\_temp( error * pGain );}
\]
Integral Controller

while (1)
    cum_E += [desired() - reading()];
    cum_E = bound_cumulative_error( cum_E );
    increase_temp( cum_E * iGain );
while (1)
    error = desired() - reading();
    cum_E += error;    // and then bound it
    increase_temp(   error * pGain +
                     cum_E * iGain );
PID Controller (predictive)

\[
\text{error} = \text{desired()} - \text{reading()}; \\
\text{cum}_E += \text{error}; \quad \text{// and then bound it} \\
\text{delta} = \text{prev}_\text{reading} - \text{this}_\text{reading}; \\
\text{increase}_\text{temp}( \quad \text{error} \times \text{pGain} + \\
\text{cum}_E \times \text{iGain} + \\
\text{delta} \times \text{dGain} );
\]
Hovercraft Control Issues

Issues you will have to address:

- Sensing location
- Sensing speed/direction
- Changing location/speed/direction
- Making informed decisions
Sensing Location

- Echolocation (distance from walls)?
- Dark/light sensor (following tape)?
- Magnetic sensor (following tape)?
Sensing Speed/Direction

- Is following tape/walls sufficient?
- What about angular momentum?
Changing Orientation

Turning is obvious ... or is it?

How do you stop turning?

Forward thrust is obvious ... or is it?

Are your fans perfect?
Some Things to Think About

Which is likely to be easiest?

When you drive, do you look ahead to turn?
Some Things to Think About

How do you tell the difference?

And does it matter?
Some Things to Think About

How do you tell the difference?

And does it matter?
Some Things to Think About

How do you tell the difference?

And does it matter?
Bottom Line

The control problem is likely to be your biggest headache when designing your hovercraft.

Give it a lot of thought.

(unless, of course, you simply bounce your way down the course and take a minor time penalty … :)