ECE 1675/2570: Robotic Control (Spring 2022)

Module I: Robotics and Controls Primer

Lectures 3 and 4: Robot Control Architectures; Time Responses of Dynamical Systems; PID Control

January 26 and February 2, 2022

Zhi-Hong Mao
Professor of ECE and Bioengineering
University of Pittsburgh, Pittsburgh, PA

Outline

- Homework 2 (due Feb. 9)
- Robot control architectures
  - Deliberative control
  - Reactive control
  - "Hybrid" control
  - Behavior-based control
- Time responses of dynamical systems
  - First-order systems
  - Second-order systems
  - Time response specifications of design
  - Frequency responses
- PID control

Robot control architectures

- Why does a robot need a control architecture?
  - How would you put multiple feedback controllers together?
  - What if you need more than feedback control?
  - How would you decide what is needed, which part of the control system to use in a given situation and for how long, and what priority to assign to it?

Might have multiple subtasks

Robot control architectures

- Why does a robot need a control architecture?
- What is control architecture?
  - A robot *control architecture* provides guiding principles and constraints for organizing a robot's control system (its brain)
  - Robot control can take place in hardware and in software, but the more complex the controller, the more likely it is to be implemented in software

Robot control architectures

- Why does a robot need a control architecture?
- What is control architecture?
  - A robot *control architecture* provides guiding principles and constraints for organizing a robot's control system (its brain)
  - Robot control can take place in hardware and in software, but the more complex the controller, the more likely it is to be implemented in software
  - Robot controllers can be implemented in various languages: *C/C++*, *Python*, *C#*, *Matlab*, *Java*, *Assembly*, *Hardware Description Languages* (HDLs), *LISP*, *industrial robot languages*, *BASIC*, *Pascal*, etc.

Note: There is *no "best" language*; as robotics grows and matures, there are more and more specialized programming languages and tools.

Robot control architectures

- Why does a robot need a control architecture?
- What is control architecture?
- Types of control architectures
  - Deliberative control
  - Reactive control
  - "Hybrid" control
  - Behavior-based control

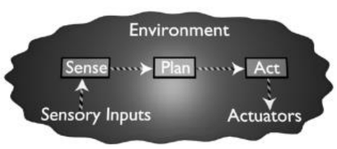
Deliberative control

- Definition
  - *Deliberation* refers to thinking hard ("thoughtfulness in decision and action")
    - Deliberative control grew out of early AI (e.g. Shakey)
    - Deliberative control looks into the future, and it works on a *long time-scale*

7

Deliberative control

- Definition
  - *Deliberation* refers to thinking hard
  - Deliberative control involves three steps: SPA
    - *Sensing (S)*
    - *Planning (P)*, the process of determining possible outcomes of actions and searching for the best sequence of actions to achieve a goal
    - *Acting (A)*



8

Deliberative control

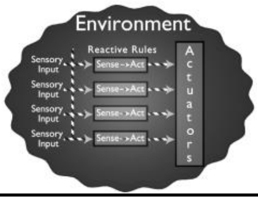
- Definition
- Drawbacks
  - *Time-scale:* it can be very slow (due to large state space)
  - *Space:* it can be very memory-intensive (state space representation requires significant storage)
  - *Information:* sometimes information is outdated
  - *Execution:* executing a plan can be difficult

Note: Since the 1980s, purely deliberative architectures are no longer used for the majority of physical robots

9

Reactive control

- Definition
  - *Reactive control* is control that tightly couples sensing and acting
    - It does not plan ahead (does not use any internal representations of the environment)
    - It is very fast
    - It is the most common control method in robotics

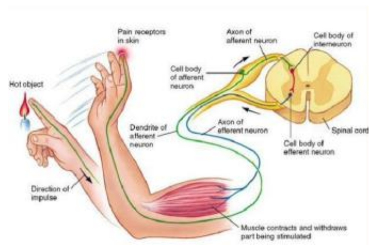


10

Reactive control

- Definition
  - *Reactive control* is control that tightly couples sensing and acting
  - Reactive control is similar to neural reflexes

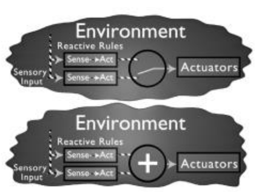
spinal cord



11

Reactive control

- Definition
- Action selection
  - *Action selection* is the process of deciding among multiple possible actions or behaviors
  - Two basic types
    - *Arbitration:* select one candidate
    - *Fusion:* combine multiple candidate actions into a single action



12

Reactive control

- Definition
- Action selection
  - *Action selection* is the process of deciding among multiple possible actions or behaviors
  - Two basic types
  - *Multitasking:* reactive systems must be able to support parallelism, the ability to monitor and execute multiple rules at once

13

Reactive control

- Definition
- Action selection
- Subsumption architecture (SA)
  - SA is the best known architecture for reactive control (introduced by Prof. Rodney Brooks at MIT in 1985)

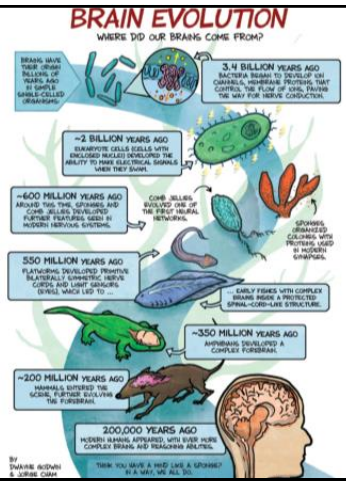
Herbert the robot

14

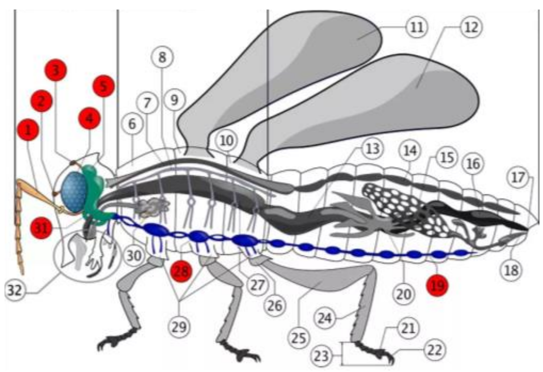
Reactive control

- Definition
- Action selection
- Subsumption architecture (SA)
  - SA is the best known architecture for reactive control
  - The *basic idea* is to build systems incrementally, from the simple parts to the more complex, using the already existing components as much as possible in the newly added stuff
    - This is called *bottom-up* design
    - This idea mimics our models of *evolutionary biology*

15

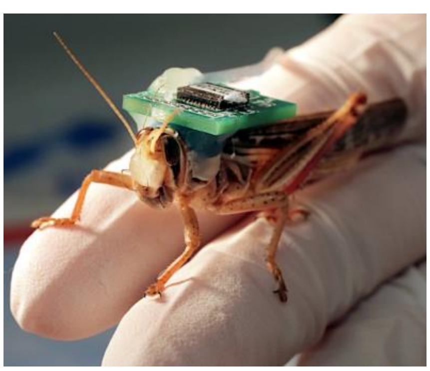


16



Nervous system of an insect

17

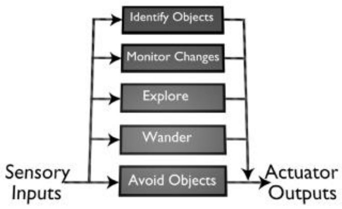


Study of insect brains to enhance robot control

18

Reactive control

- Definition
- Action selection
- **Subsumption architecture (SA)**
  - SA is the best known architecture for reactive control
  - The basic idea
  - SA is modular, with a hierarchy among the modules, which are suggestively called layers



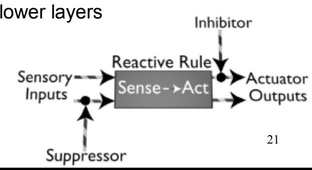
Reactive control

- Definition
- Action selection
- **Subsumption architecture (SA)**
  - SA is the best known architecture for reactive control
  - The basic idea
  - SA is modular, with a hierarchy among the modules, which are suggestively called layers
    - Each layer performs some task (e.g. avoiding objects)
    - Each layer is largely independent of other layers (allowing each to be designed and debugged separately)

Reactive control

- Definition
- Action selection
- **Subsumption architecture (SA)**
  - SA is the best known architecture for reactive control
  - The basic idea
  - SA is modular, with a hierarchy among the modules, which are suggestively called layers
    - Each layer performs some task
    - Each layer is largely independent of other layers
    - *Subsumption:* higher layers can, under certain conditions, "subsume" aspects of lower layers

The subsumption can occur by either suppressing the inputs or inhibiting the outputs



Reactive control

- Definition
- Action selection
- Subsumption architecture (SA)
  - SA is the best known architecture for reactive control
  - The basic idea
  - SA is modular, with a hierarchy among the modules, which are suggestively called layers
    - Each layer performs some task
    - Each layer is largely independent of other layers
    - *Subsumption*
    - "*The world is its own best model*," so no internal model is used
    - No sequencing of tasks between layers is used; they are all running in parallel all the time

22

Reactive control

- Definition
- Action selection
- Subsumption architecture (SA)
- Drawbacks of reactive control
  - No (or minimal) state
  - No internal representations of the world
  - No memory
  - No (or minimal) learning

23

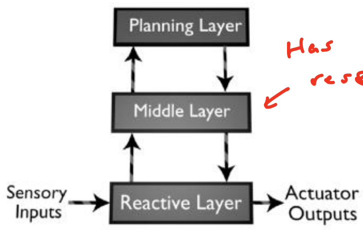
"Hybrid" control

- Definition
  - *"Hybrid" control* involves the combination of reactive and deliberative control within a single robot control system (it roughly means to think and act independently and concurrently)
  - This concept is different from the concept of hybrid control used in controls literature

24

“Hybrid” control

- Definition
- Three-layer architecture
  - Three layers: a *reactive layer*, a *planning layer*, and a *middle layer* linking the two



Has a lot of responsibility

25

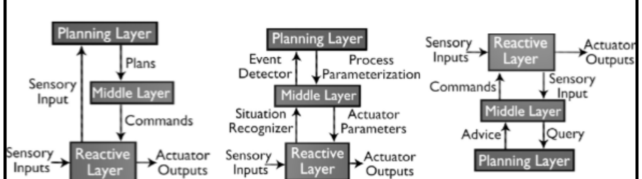
“Hybrid” control

- Definition
- Three-layer architecture
  - Three layers: a *reactive layer*, a *planning layer*, and a *middle layer* linking the two
  - The “magic middle”
    - compensates for the limitations of the other two
    - reconciles their disparate time-scales
    - reconciles their different representations
    - reconciles contradictory commands

26

“Hybrid” control

- Definition
- Three-layer architecture
  - Three layers: a *reactive layer*, a *planning layer*, and a *middle layer* linking the two
  - The “magic middle”
  - Various ways of managing layer interaction



27

"Hybrid" control

- Definition
- Three-layer architecture
- Drawbacks of "hybrid" control
  - The middle layer is difficult to design and build
  - The middle layer is specialized to a specific problem/robot
  - Sometimes the reactive and deliberative layers work to the detriment of each other

28

Behavior-based control

- Definition
  - *Behavior-based control* involves the use of "behaviors" as modules for control
  - About *behaviors*
    - Behaviors achieve and/or maintain particular goals
    - Behaviors are time-extended, not instantaneous
    - Behaviors can take inputs from sensors and also from other behaviors, and can send outputs to effectors and to other behaviors—we can create network of behaviors
    - Behaviors are more complex than actions

29

Behavior-based control

- Definition
- Connections with the other control architectures
  - Behavior-based control is closer to reactive control than to hybrid control, and farthest from deliberative control
  - Behavior based systems have reactive components, just as hybrid systems do, but they do not have traditional deliberative components

Note: Reactive control is too inflexible (incapable of representation or learning); deliberative control is too slow and cumbersome; and hybrid systems require complex interaction among components

30

Behavior-based control

- Definition
- Connections with the other control architectures
- Principles of design
  - Behaviors are typically executed in parallel
  - Networks of behaviors are used to store state and construct world models/representations
  - Behaviors operate on compatible time-scales

31

Behavior-based control

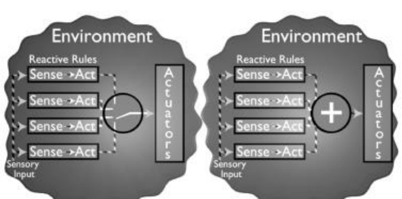
- Definition
- Connections with the other control architectures
- Principles of good design
- Key properties
  - The ability to react in real-time
  - The ability to use representations to generate (not only reactive) behavior
  - The ability to use a uniform structure and representation throughout the system (with no intermediate layers)

32

Behavior-based control

- Definition
- Connections with the other control architectures
- Principles of good design
- Key properties
- Behavior coordination (or action selection)

Combining behaviors



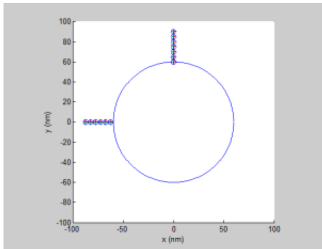
Fixed Heirarchy
Dynamic Heirarchy
Decision Theory
Probability

Potential Fields
Fuzzy Logic
Dynamical Systems
Motor Schemas

33

Behavior-based control

- Definition
- Connections with the other control architectures
- Principles of good design
- Key properties
- Behavior coordination
- Emergent behavior
  - Definition: *Emergent behavior* is structured (patterned, meaningful) behavior that is apparent from the observer's viewpoint, but not from the controller's /robot's view point



34

Parallel and distributed

Behavior-based control

- Definition
- Connections with the other control architectures
- Principles of good design
- Key properties
- Behavior coordination
- Emergent behavior
  - Definition
  - Architectures and emergence
    - *Reactive* and *behavior-based* systems employ parallel rules and behaviors, respectively, which interact with each other and the environment, thus providing the perfect foundation for exploiting emergent behavior by design
    - *Deliberative* systems are sequential (with no parallel interactions between the components) and thus would require environment structure to have any behavior emerge over time
    - *Hybrid* systems follow the deliberative model in attempting to produce a coherent, uniform output of the system, minimizing interactions and thus minimizing emergence

35

Time responses of first-order systems

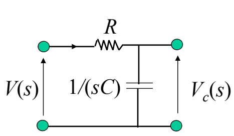
- First-order systems

dc gain

$$G(s)=\frac{C(s)}{R(s)}=\frac{b_0}{s+a_0}=\frac{K}{\tau s+1}$$

Time constant

  - Examples:



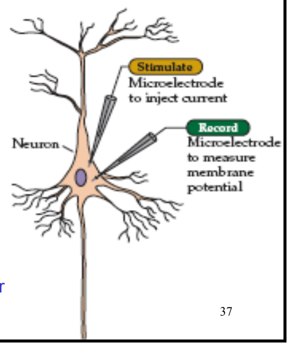
$$G(s)=\frac{V_C(s)}{V(s)}=\frac{1/(Cs)}{R+1/(Cs)}=\frac{1}{RCs+1}$$

Question: What does this circuit often used for?

36

Time responses of first-order systems

- First-order systems
  - Examples
    - Cruise control model
    - Leaky water tank model
    - Eye movement control model

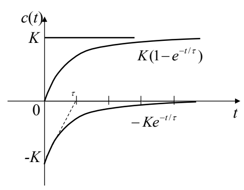


Neuron as leaky-integrator

Time responses of first-order systems

- First-order systems $G(s)=\frac{C(s)}{R(s)}=\frac{K}{\tau s+1}$
- Step response

$$R(s)=1/s,$$
$$C(s)=\frac{1}{s}\frac{K/\tau}{s+1/\tau}=\frac{K}{s}-\frac{K}{s+1/\tau},$$
$$c(t)=K(1-e^{-t/\tau}),\quad t>0$$



The limit of $c(t)$ as t goes to infinity is called the final value, or steady-state value of the response.

The parameter τ is called time constant; we may consider an exponential term to be zero after four time constants.

Time responses of first-order systems

- First-order systems $G(s)=\frac{C(s)}{R(s)}=\frac{K}{\tau s+1}$
- Step response

$$R(s)=1/s,$$
$$C(s)=\frac{1}{s}\frac{K/\tau}{s+1/\tau}=\frac{K}{s}-\frac{K}{s+1/\tau},$$
$$c(t)=K-Ke^{-t/\tau},\quad t>0$$

Forced response or steady-state response — Input signal

Natural response or transient response — System

Leaky Tank model

$A\dot{y}=f \rightarrow AsY(s)=F(s)$

$$\frac{Y(s)}{F(s)}=\frac{1}{As}$$

$A\dot{y}=f-cy$ (leaky tank)

$cy(s)+Asy(s)=F(s)$

$$\frac{y(s)}{F(s)}=\frac{1}{As+c}\rightarrow\frac{1/c}{\frac{A}{c}s+1}$$

first order system

DC Gain $= K = 1/c$

Time Constant $= A/c$

Do this problem

Closed loop system

DC Gain $= \frac{K}{1+K}$

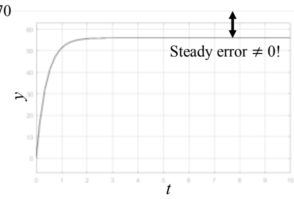
Ask about transfer function for this problem

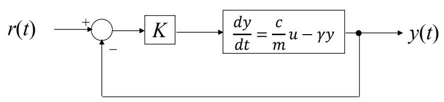
Time responses of first-order systems

- First-order systems
- Step response

$$c(t) = K(1 - e^{-t/\tau}), \quad t > 0$$

  – Example: using P control in cruise control





40

Time responses of first-order systems

- First-order systems
- Step response

$$G(s) = \frac{C(s)}{R(s)} = \frac{K}{\tau s + 1}$$

- System dc gain
  – The system dc gain is the steady-state gain to a constant input for the case the output has a final value, and it is equal to the system transfer function evaluated at $s = 0$ (why?)

41

Time responses of second-order systems

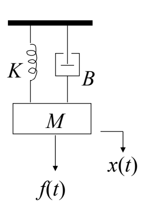
- Second-order systems

$$G(s) = \frac{C(s)}{R(s)} = \frac{b_0}{s^2 + a_1 s + a_0} = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}$$

Natural frequency

Damping ratio

  – Examples:



$$M\frac{d^2x}{dt^2} = f(t) - B\frac{dx}{dt} - Kx$$

$$G(s) = \frac{X(s)}{F(s)} = \frac{1}{Ms^2 + Bs + K}$$

42

Time responses of second-order systems

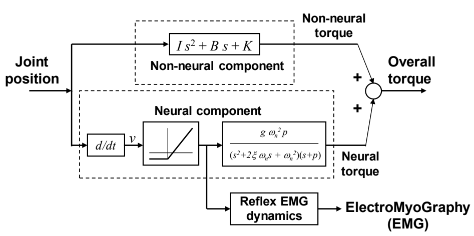
- Second-order systems
  - Examples:

An example from Dr. Ruiping Xia's research project (I am a collaborator)

43

Time responses of second-order systems

- Second-order systems
  - Examples:



44

Time responses of second-order systems

- Second-order systems $G(s) = \dfrac{\omega_n^2}{s^2 + 2\varsigma\omega_n s + \omega_n^2}$
- Step response
  - Case 1: $\zeta < 1$ (underdamped), including $\zeta = 0$ (undamped)

$$\boxed{c(t) = 1 - \frac{1}{\beta} e^{-\varsigma\omega_n t} \sin(\beta\omega_n t + \theta),}$$ where $\beta = \sqrt{1-\varsigma^2}$ and $\theta = \tan^{-1}(\beta/\varsigma)$

  - Case 2: $\zeta > 1$ (overdamped)

$$\boxed{c(t) = 1 + k_1 e^{-t/\tau_1} + k_2 e^{-t/\tau_2},}$$ where $\tau_{1,2} = 1/\left(\varsigma\omega_n \pm \omega_n\sqrt{\varsigma^2 - 1}\right)$

  - Case 3: $\zeta = 1$ (critically damped)

$$\boxed{c(t) = 1 + k_1 e^{-t/\tau} + k_2 t e^{-t/\tau},}$$ where $\tau = 1/\omega_n$

45

Time responses of second-order systems

- Second-order systems $G(s)=\dfrac{\omega_n^2}{s^2+2\zeta\omega_n s+\omega_n^2}$
- Step response

Case 1: $\zeta<1$ (underdamped)

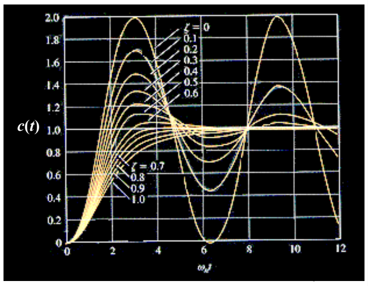
$$c(t)=1-\frac{1}{\beta}e^{-\zeta\omega_n t}\sin(\beta\omega_n t+\theta),$$

Case 2: $\zeta>1$ (overdamped)

$$c(t)=1+k_1e^{-t/\tau_1}+k_2e^{-t/\tau_2},$$

Case 3: $\zeta=1$ (critically damped)

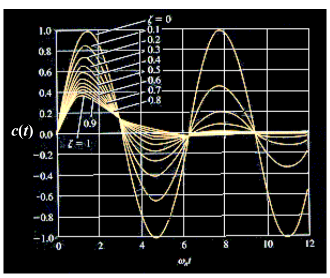
$$c(t)=1+k_1e^{-t/\tau}+k_2te^{-t/\tau},$$



Time responses of second-order systems

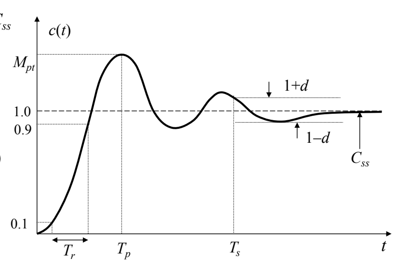
- Second-order systems $G(s)=\dfrac{\omega_n^2}{s^2+2\zeta\omega_n s+\omega_n^2}$
- Step response
  - Case 1
  - Case 2
  - Case 3
  - Initial condition and impulse response

The initial condition excitation of higher-order systems cannot be modeled as simply as that of the first-order system; however, the impulse response of any system does give an indication of the nature of the initial-condition response, and thus the transient response



Time response specifications in design

- Some parameters
  - Rise time, T_r
  - Peak value of the step response, M_{pt} ; time to reach it, T_p (how to calculate T_p?)
  - Steady state value, C_{ss}
  - Percent overshoot, $\dfrac{M_{pt}-C_{ss}}{C_{ss}}\times 100$
  - Settling time, T_s (how to calculate T_s?)

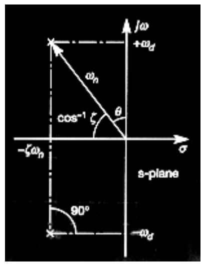


Time response specifications in design

- Some parameters
- Time response and pole locations
  - The settling time is inversely related to the real part of the poles (the speed of response is increased by moving the poles to the left in the s-plane)

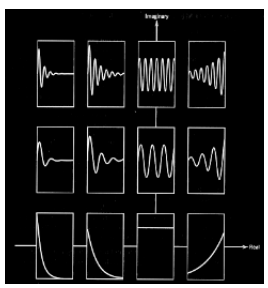
$$\boxed{T_s = k\tau = \frac{k}{\varsigma\omega_n}}$$

  - Decreasing the angle $\cos^{-1}\zeta$ (increasing ζ) reduces the percent overshoot



Time response specifications in design

- Some parameters
- Time response and pole locations
  - The settling time is inversely related to the real part of the poles (the speed of response is increased by moving the poles to the left in the s-plane)
  - Decreasing the angle $\cos^{-1}\zeta$ (increasing ζ) reduces the percent overshoot



This picture shows how changing pole locations in the s-plane affects responses

Frequency response of systems

- Frequency response: steady-state response of systems to sinusoidal inputs

$$r(t) = A\cos\omega_1 t, \quad R(s) = \frac{As}{s^2+\omega_1^2},$$

$$\lim_{t\to\infty} c_g(t) = 0$$

$$C(s) = G(s)R(s) = \frac{k_1}{s-j\omega_1} + \frac{k_2}{s+j\omega_1} + C_g(s)$$

$$k_1 = \frac{1}{2}AG(j\omega_1),\ k_2 = \frac{1}{2}AG(-j\omega_1),\ \boxed{G(j\omega_1) = |G(j\omega_1)|e^{j\phi(\omega_1)}}$$

$$c_{ss}(t) = k_1 e^{j\omega_1 t} + k_2 e^{-j\omega_1 t} = A|G(j\omega_1)|\frac{e^{j(\omega_1 t+\phi(\omega_1))} + e^{-j(\omega_1 t+\phi(\omega_1))}}{2}$$

$$= A|G(j\omega_1)|\cos(\omega_1 t + \phi(\omega_1))$$

What can we learn from tjw

→ magnitude and phase

magnitude: shape of the "envelope" —

ratio of $\frac{\text{output}}{\text{input}}$ in SS

Frequency response of systems

- Frequency response: steady-state response of systems to sinusoidal inputs

$$r(t) = A\cos\omega_1 t, \quad G(j\omega_1) = |G(j\omega_1)|e^{j\phi(\omega_1)}$$

$$\boxed{c_{ss}(t) = A|G(j\omega_1)|\cos(\omega_1 t + \phi(\omega_1))}$$

  – The steady-state gain of a system for a sinusoidal input is the magnitude of the transfer function evaluation at $s = j\omega_1$, and the phase shift of the output sinusoid relative to the input sinusoid is the angle of $G(j\omega_1)$

52

System has to be stable

Frequency response of systems

- Frequency response: steady-state response of systems to sinusoidal inputs
  – The steady-state gain of a system for a sinusoidal input is the *magnitude* of the transfer function evaluation at $s = j\omega_1$, and the *phase shift* of the output sinusoid relative to the input sinusoid is the angle of $G(j\omega_1)$
  – $G(j\omega)$ is defined as the frequency response function

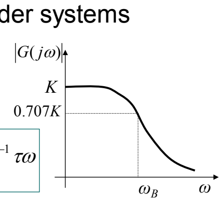
$$G(j\omega) = |G(j\omega)|e^{j\phi(\omega)}$$

53

Frequency response of systems

- Frequency response: steady-state response of systems to sinusoidal inputs
- Frequency response of first-order systems

$$G(s) = \frac{K}{\tau s + 1}$$



$$\boxed{|G(j\omega)| = \frac{K}{(1+\tau^2\omega^2)^{1/2}}, \quad \phi(\omega) = -\tan^{-1}\tau\omega}$$

  – System bandwidth, ω_B: The frequency at which the gain is equal to $1/\mathrm{sqrt}(2)$ (approximately 0.707) times the gain at very low frequencies

54

Frequency response of systems

- Frequency response: steady-state response of systems to sinusoidal inputs
- Frequency response of first-order systems
- Frequency response of second-order systems

$$G(s) = \frac{\omega_n^2}{s^2 + 2\varsigma\omega_n s + \omega_n^2} = \frac{1}{(s/\omega_n)^2 + 2\varsigma(s/\omega_n) + 1}$$

$$G(j\omega) = \frac{1}{[1-(\omega/\omega_n)^2] + j2\varsigma(\omega/\omega_n)}$$

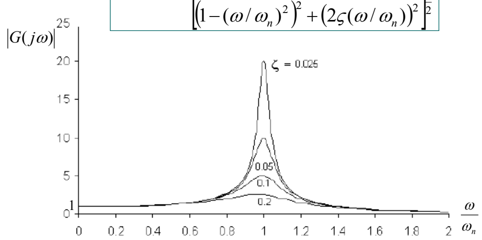
$$\boxed{|G(j\omega)| = \frac{1}{\left[\left(1-(\omega/\omega_n)^2\right)^2 + \left(2\varsigma(\omega/\omega_n)\right)^2\right]^{\frac{1}{2}}}}$$

Question: What will happen if $\zeta = 0$ and $\omega = \omega_n$?

Frequency response of systems

- Frequency response: steady-state response of systems to sinusoidal inputs
- Frequency response of first-order systems
- Frequency response of second-order systems

$$\boxed{|G(j\omega)| = \frac{1}{\left[\left(1-(\omega/\omega_n)^2\right)^2 + \left(2\varsigma(\omega/\omega_n)\right)^2\right]^{\frac{1}{2}}}}$$



PID control

- Proportional control
  - In proportional control, steady-state error tends to depend inversely upon proportional gain
  - Proportional control has a tendency to make a system faster
  - Proportional control does not change the order of the system

58

PID control

- Proportional control
- Integral control
  - In integral control, steady-state error should be zero (prerequisite: the closed loop system has to be stable)
  - Integral control has a tendency to make a system slower and may even sacrifice stability
  - Integral control changes the order of the system

59

PID control

- Proportional control
- Integral control
- Derivative control
  - Derivative control tends to increase the stability of the system
  - Derivative control tends to reduce the overshoot and improve the transient response
  - Derivative control changes the order of the system

60

PID control

- Proportional control
- Integral control
- Derivative control

Closed-loop response	Rise time	Overshoot	Settling time	Steady-state error
K_P	Decrease	Increase	Small change	Decrease
K_I	Decrease	Increase	Increase	Eliminate
K_D	Small change	Decrease	Decrease	Small change

61

PID control

- Proportional control
- Integral control
- Derivative control
- Another view on PID control
  - The proportional term gives the controller output a component that is a function of the present state of the system
  - The integrator output is determined by the past state of the system
  - The differentiator is a function of the slope of its input and thus can be considered to be a predictor of the future state of the system
  - The PID controller can viewed as giving control that is a function of the past, the present, and the predicted future

62

References

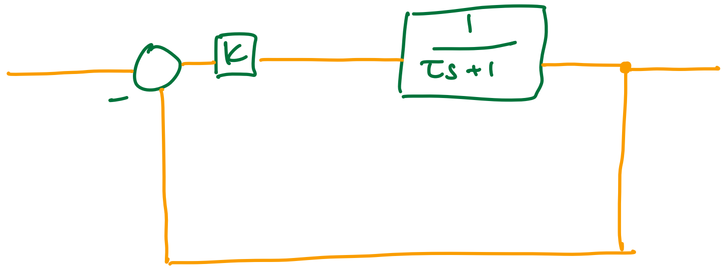
- K. J. Astrom and R. M. Murray, Feedback Systems: An Introduction for Scientists and Engineers, 2010.
- M. Mataric, The Robotics Primer, MIT Press, 2007.
- C. L. Phillips and R. D. Harbor, Feedback Control Systems, 4th Edition, Prentice Hall, 2000.
- R. A. R. Picone, Lecture Notes for ME 454/554 Robotics and Automation, Saint Martin's University, 2017.
- J. W. Roberts, R. Cory, and R. Tedrake, "On the controllability of fixed-wing perching," in Proceedings of the American Control Conference, 2009.
- R. Tedrake, Lecture Notes for 6.832 Underactuated Robotics, MIT, 2018.
- http://physiologyplus.com/withdrawal-reflex/
- https://www.newsmax.com/newsfront/artificial-intelligence-defense-contractor-insect/2019/01/10/id/897745/
- http://www.penguinslab.com/Pictures/Glassshatter/Pilot_shatter.jpg
- https://www.scientificamerican.com/article/your-brain-evolved-from-bacteria/
- https://www.thoughtco.com/internal-anatomy-of-an-insect-1968483

63

PID Control

Focus on 1st and 2nd order systems because they represent higher order systems

1. P (Proportional) Control → "Pure gain"



Original Pole of $\frac{1}{\tau s+1}$

$= -\frac{1}{\tau}$

With K, closed loop transfer function is $\dfrac{K \cdot \frac{1}{\tau s+1}}{1 + K\frac{1}{\tau s+1}}$

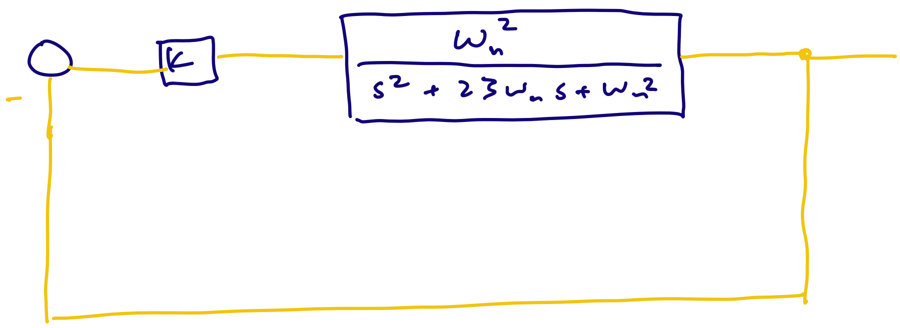
$= \dfrac{K}{K + \tau s + 1}$

Pole $= -\dfrac{1+K}{\tau}$ Pole moves to left as K increases

Time constant gets smaller as K increases, system is faster

$= \dfrac{1}{\left|1 - \frac{K+1}{\tau}\right|} = \dfrac{\tau}{K+1}$

Don't have to worry about stability since it's a first-order system



$\zeta < 1$ (underdamped)
Focus on this case

Two complex poles: $-\zeta\omega_n \pm \left(\sqrt{1-\zeta^2}\right)\omega_n j$

Closed loop transfer function:

$$\frac{K\dfrac{\omega_n^2}{s^2+2\zeta\omega_n s+\omega_n^2}}{1+K\dfrac{\omega_n^2}{s^2+2\zeta\omega_n s+\omega_n^2}} \longrightarrow \frac{K\omega_n^2}{s^2+2\zeta\omega_n s+(1+K)\omega_n^2}$$

Compare to nominal form in denom. only

DC Gain $= \dfrac{K\omega_n^2}{(1+K)\omega_n^2} = \dfrac{K}{1+K}$ Increase $K \rightarrow$ steady state error will go to 0

$s = 0$

steady state error $= 1 - \dfrac{K}{1+K} = \dfrac{1}{1+K}$ for unit step input

poles: $p_{1,2} = \dfrac{-2\zeta\omega_n \pm \sqrt{(2\zeta\omega_n)^2 - 4(1+K)}}{2}$

$$= -\zeta\omega_n \pm \sqrt{\zeta^2 - (1+K)}\,\omega_n$$

$$= -\zeta\omega_n \pm j\sqrt{1+K-\zeta^2}\,\omega_n$$

Poles move away from the real axis as K grows (real part does not change)

Time constant for underdamped 2nd order system is

$$\tau = \frac{1}{\zeta\omega_n} \quad \text{(real part)}$$

Does not change as K changes

Transfer function: $\dfrac{K\omega_n^2}{s^2 + 2\zeta\omega_n s + (1+K)\omega_n^2}$

(in denom.) Nominal form: $s^2 + 2\zeta_{new}\omega_{new}s + \omega_{new}^2$

$$\omega_{new} = \sqrt{(1+K)\omega_n^2} = \sqrt{1+K}\,\omega_n$$

$$2\zeta_{new}\omega_{new} = 2\zeta\omega_n$$

$$\zeta_{new} = \frac{\zeta\omega_n}{\omega_{new}}$$

$$= \frac{\zeta\omega_n}{\sqrt{1+K}\,\omega_n} = \frac{\zeta_n}{\sqrt{1+K}} = \zeta_{new}$$

As K increases, natural frequency increases (this is a problem - system is stiffer)

As K increases, damping ratio decreases

(this depends - if original system is overdamped, this is good; BUT, we are starting from an underdamped system, and we don't like decreasing ζ_n in that case because it results in larger overshoot)

damped frequency increases

rise time (influenced by time constant and frequency) 10% - 90%

time constant remains the same, frequency increases

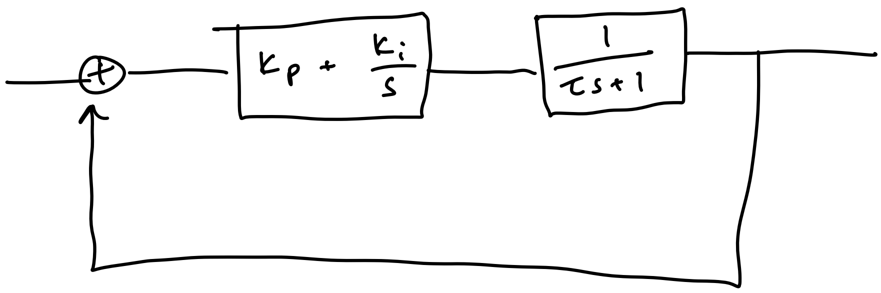
$\longrightarrow$ rise time is reduced

Increasing K makes system less stable for higher order system (overshoot)

2. Integral control

Start with First Order System

$C(s) = K_p + \frac{K_i}{s}$ (K_p is fixed)



$$T(s) = \frac{\left(K_p + \frac{K_i}{s}\right)\left(\frac{1}{\tau s+1}\right)}{1 + \left(K_p + \frac{K_i}{s}\right)\left(\frac{1}{\tau s+1}\right)} = \frac{K_p s + K_i}{\tau s^2 + s + K_p s + K_i}$$

$$= \frac{\frac{K_p s}{\tau} + \frac{K_i}{\tau}}{s^2 + \frac{K_p + 1}{\tau} s + \frac{K_i}{\tau}}$$

- DC Gain, e_{ss} System is stable → replace s with 0
DC Gain = 1 ; $T(0) = 1$

e_{ss} = 1 − output_{ss} if input is unit step
= 1 − 1 since gain = 1 (output = input)
= 0

- Intuitions about e_{ss}
(i) If the closed-loop system is stable, then DC-bias of the input to an integrator must be 0.
→ e_{ss} is input to integrator; since system is stable, e_{ss} must be 0

(ii) DC Gain of $\frac{K_i}{s} = \infty$ (replace s with 0); because of this, $e_{ss} \to 0$

- Transient response

Denom. is $s^2 + \frac{K_p+1}{\tau} s + \frac{K_i}{\tau}$

Nominal form of 2nd order:
$s^2 + 2\zeta\omega_n s + \omega_n^2$

$$\omega_n = \sqrt{\frac{K_i}{\tau}}$$

$$\zeta = \frac{K_p+1}{2\tau\omega_n} = \frac{K_p+1}{2\sqrt{K_i\tau}}$$

Increase K_I → ω_n increases
→ ζ decreases

Only want this if we have an overdamped system

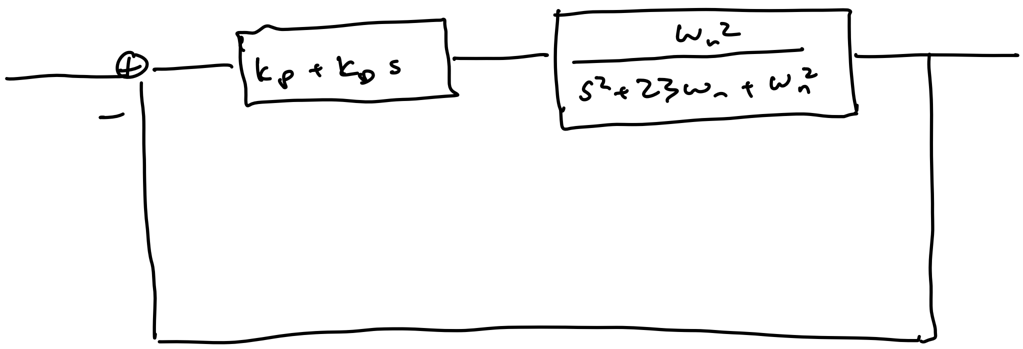
In underdamped situation, ζ decreasing will increase overshoot

Settling time does not change much on 1st order system

Major Benefit of PI control is removing e_{ss},
Price is increased oscillation

3. Derivative control (aka Damping Control)

$$C(s) = k_P + k_D s$$



$k_D s$ has no influence on steady-state error because DC Gain of $k_D s = 0$

$$T(s) = \frac{(k_P + k_D s) \cdot \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}}{1 + (k_P + k_D s) \cdot \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}}$$

$$= \frac{(k_P + k_D s)\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2 + (k_P + k_D s)\omega_n^2} \longrightarrow s^2 + (2\zeta\omega_n + k_D\omega_n^2)s + (k_P + 1)\omega_n^2$$

compare to $s^2 + 2\zeta_{new}\omega_{new} s + \omega_{new}^2$

$$\omega_{new} = \sqrt{k_P + 1}\,\omega_n$$

$$\zeta_{new} = \frac{2\zeta\omega_n + k_D\omega_n^2}{2\sqrt{k_P + 1}\,\omega_n} = \frac{2\zeta + k_D\omega_n}{2\sqrt{k_P + 1}}$$

k_D increases $\longrightarrow$

ω_{new} stays constant

ζ_{new} increases

What does this mean?

Already said no influence on e_{ss}

Reduces overshoot (Both k_I and k_P decrease ζ_n which increases overshoot)