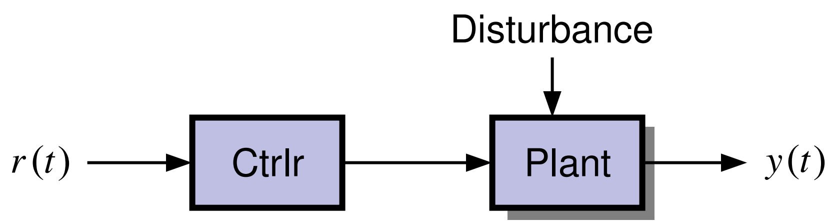
BASIC PROPERTIES OF FEEDBACK

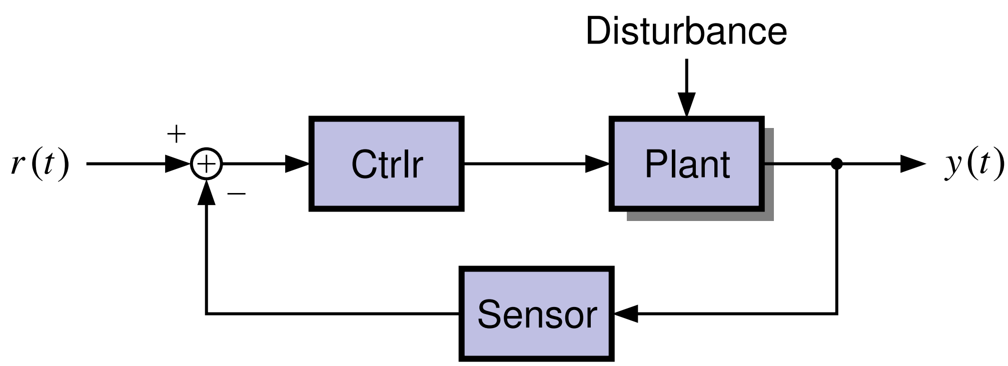
4.1: Setting up an example to benchmark controllers

- There are two basic types/categories of control systems:

OPEN LOOP:



CLOSED LOOP:

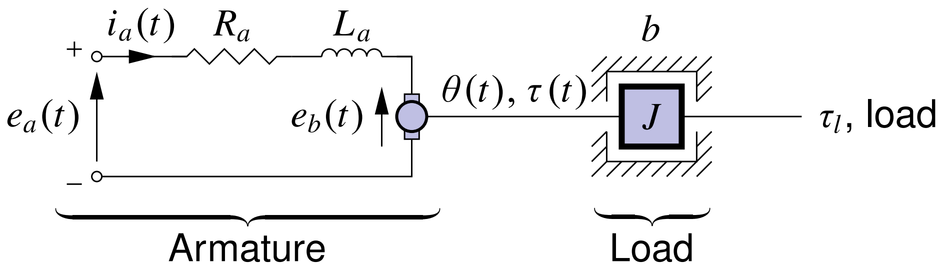


- This chapter of notes is concerned with comparing open-loop and closed-loop control, and showing the potential benefits (and some pitfalls) of closed-loop (*i.e.*, feedback) control.
- We evaluate these two categories of controller in a number of ways: disturbance rejection, sensitivity, dynamic tracking, steady-state error, and stability.

DC motor speed control

- In order to compare open- and closed-loop control, we will use an extended example.

- Recall equations of motion for a dc motor (pg. 2–19) but add a load torque.



- Assume that we are trying to control motor speed:

$$\left.\begin{aligned} J\ddot{\theta} + b\dot{\theta} &= k_\tau i_a + \tau_l \\ k_e\dot{\theta} + L_a\frac{\mathrm{d}i_a}{\mathrm{d}t} + R_a i_a &= e_a \end{aligned}\right\} \begin{aligned} &\text{let output } y = \dot{\theta}, \\ &\quad \text{disturbance } w \triangleq \tau_l. \end{aligned}$$

$$\left.\begin{aligned} J\dot{y} + by &= k_\tau i_a + w \\ k_e y + L_a\frac{\mathrm{d}i_a}{\mathrm{d}t} + R_a i_a &= e_a \end{aligned}\right\} \begin{aligned} &sJY(s) + bY(s) = k_\tau I_a(s) + W(s) \\ &k_e Y(s) + sL_a I_a(s) + R_a I_a(s) = E_a(s) \end{aligned}$$

- Solving the mechanical equation for $I_a(s)$ gives

$$sJY(s) + bY(s) = k_\tau I_a(s) + W(s)$$

$$\begin{aligned} k_\tau I_a(s) &= sJY(s) + bY(s) - W(s) \\ I_a(s) &= \frac{(sJ+b)\,Y(s) - W(s)}{k_\tau}. \end{aligned}$$

- Substituting into the electrical equation gives

$$\begin{aligned} k_e Y(s) + sL_a I_a(s) + R_a I_a(s) &= E_a(s) \\ k_e Y(s) + (sL_a + R_a)\,\frac{(sJ+b)\,Y(s) - W(s)}{k_\tau} &= E_a(s). \end{aligned}$$

- Some algebra then yields

$$k_e Y(s) + (sL_a + R_a)\,\frac{(sJ+b)\,Y(s) - W(s)}{k_\tau} = E_a(s)$$

$$k_\tau k_e Y(s) + (sL_a + R_a)\,(sJ+b)\,Y(s) = k_\tau E_a(s) + (R_a + L_a s)\,W(s)$$

$$\left(JL_a s^2 + (bL_a + JR_a)\,s + (bR_a + k_\tau k_e)\right)Y(s) = k_\tau E_a(s) + (R_a + L_a s)\,W(s).$$

- Dividing both sides by $bR_a + k_\tau k_e$ gives

$$\left(\frac{JL_a}{bR_a+k_\tau k_e}s^2 + \frac{bL_a+JR_a}{bR_a+k_\tau k_e}s + 1\right)Y(s) = \frac{k_\tau}{bR_a+k_\tau k_e}E_a(s) + \frac{R_a+L_a s}{bR_a+k_\tau k_e}W(s).$$

- The left-hand-side can be factored into two parts:

$$\left(\frac{JL_a}{bR_a + k_\tau k_e}s^2 + \frac{bL_a + JR_a}{bR_a + k_\tau k_e}s + 1\right) = (\tau_1 s + 1)\,(\tau_2 s + 1)\,.$$

  - Roughly, one of these time constants is mechanical; the other is electrical.

- If we assume that the mechanical time constant is much larger than the electrical, the right-hand-side can be approximated by

$$\frac{k_\tau}{bR_a + k_\tau k_e}E_a(s) + \frac{R_a + L_a s}{bR_a + k_\tau k_e}W(s) \approx AE_a(s) + BW(s),$$

where

$$A = k_\tau/(bR_a + k_\tau k_e)$$

$$B \approx R_a/(bR_a + k_\tau k_e).$$

- Then, we have overall relationship

$$(\tau_1 s + 1)(\tau_2 s + 1)Y(s) = AE_a(s) + BW(s).$$

- So,

$$Y(s) = \frac{A}{(\tau_1 s + 1)(\tau_2 s + 1)}E_a(s) + \frac{B}{(\tau_1 s + 1)(\tau_2 s + 1)}W(s).$$

4.2: Advantage of feedback: Disturbance rejection

- We look at how the open-loop and feedback systems respond to a step-like disturbance.
- If $e_a(t) = e_a \cdot 1(t)$ (constant) and $w(t) = w \cdot 1(t)$ (constant), What is steady state output?
- Recall Laplace-transform final value theorem:

> *If a signal has a constant final value, it may be found as*
>
> $$y_{ss} = \lim_{s\to 0} sY(s).$$
>
> *Note: A signal will have a constant final value iff all of the poles of $Y(s)$ are strictly in the left-half s-plane, except possibly for a single pole at $s = 0$.*

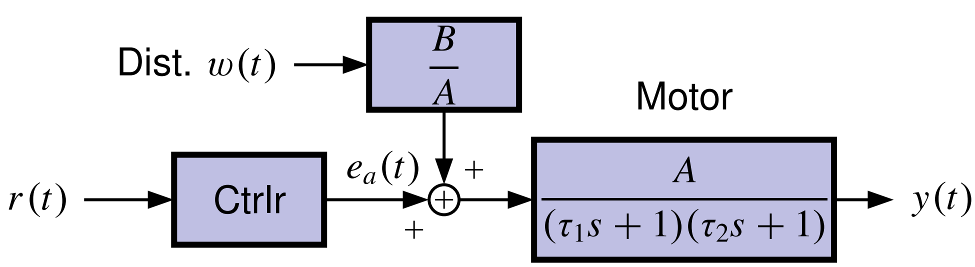
- For the input signals $e_a(t)$ and $w(t)$, we have

$$E_a(s) = \frac{e_a}{s}, \quad W(s) = \frac{w}{s}.$$

- So,

$$\begin{aligned} y_{ss} &= \lim_{s\to 0} s\left(\frac{A}{(\tau_1 s + 1)(\tau_2 s + 1)}\frac{e_a}{s} + \frac{B}{(\tau_1 s + 1)(\tau_2 s + 1)}\frac{w}{s}\right) \\ &= Ae_a + Bw. \end{aligned}$$

- This is the response of the open-loop system (without a controller).
- Let's make a simple controller for the open-loop system. The block diagram looks like:



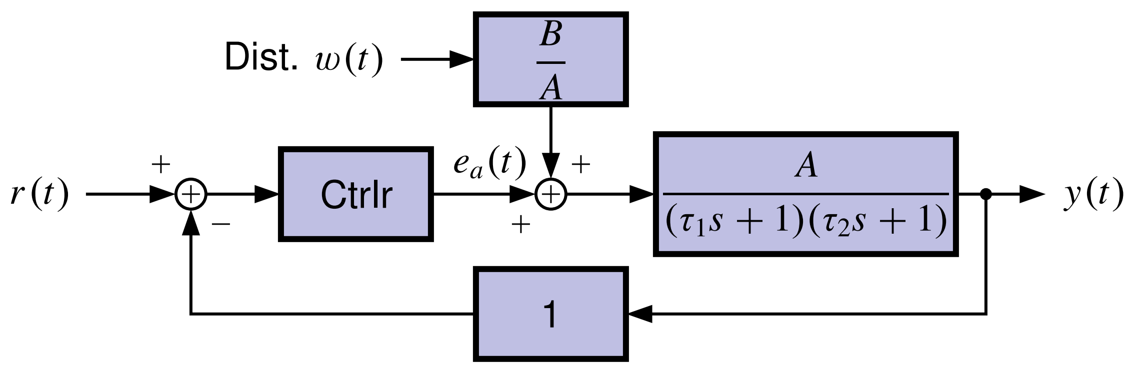
- We will design the controller to be a gain of K_{ol} such that

$$e_a(t) = K_{ol} r(t),$$

- Choose K_{ol} so that there is no steady-state error when $w = 0$.

$$\begin{aligned} y_{ss} &= AK_{ol} r_{ss} + Bw_{ss} \\ &\Rightarrow K_{ol} = 1/A. \end{aligned}$$

- Is closed-loop any better? The block diagram looks like:



- Let's make a similar controller for the closed-loop system (with the possibility of a different value of K.)

$$e_a(t) = K_{cl}(r(t) - y(t)).$$

- The transfer function for the closed-loop system is:

$$\begin{aligned} Y(s) &= \frac{AK_{cl}}{(\tau_1 s+1)(\tau_2 s+1)}(R(s) - Y(s)) + \frac{B}{(\tau_1 s+1)(\tau_2 s+1)} W(s) \\ &= \frac{AK_{cl}}{(\tau_1 s+1)(\tau_2 s+1)} R(s) - \frac{AK_{cl}}{(\tau_1 s+1)(\tau_2 s+1)} Y(s) \\ &\quad + \frac{B}{(\tau_1 s+1)(\tau_2 s+1)} W(s). \end{aligned}$$

- Combining $Y(s)$ terms

$$Y(s)\left(1+\frac{AK_{cl}}{(\tau_1 s+1)(\tau_2 s+1)}\right)=\frac{AK_{cl}}{(\tau_1 s+1)(\tau_2 s+1)}R(s)+\frac{B}{(\tau_1 s+1)(\tau_2 s+1)}W(s)$$

$$Y(s)\left(\frac{(\tau_1 s+1)(\tau_2 s+1)+AK_{cl}}{(\tau_1 s+1)(\tau_2 s+1)}\right)=\frac{AK_{cl}}{(\tau_1 s+1)(\tau_2 s+1)}R(s)+\frac{B}{(\tau_1 s+1)(\tau_2 s+1)}W(s).$$

- This gives

$$Y(s)=\frac{AK_{cl}}{(\tau_1 s+1)(\tau_2 s+1)+AK_{cl}}R(s)+\frac{B}{(\tau_1 s+1)(\tau_2 s+1)+AK_{cl}}W(s).$$

- Employing the final-value theorem for $w=0$ gives

$$\begin{aligned} y_{ss} &= \frac{AK_{cl}}{1+AK_{cl}}r_{ss} \\ &= \frac{1}{1+\frac{1}{AK_{cl}}}r_{ss}. \end{aligned}$$

- If $AK_{cl}\gg 1$, $y_{ss}\approx r_{ss}$.
- Open-loop with load:

$$\begin{aligned} y_{ss} &= AK_{ol}r_{ss}+Bw_{ss}=r_{ss}+Bw_{ss} \\ \delta y &= Bw_{ss}. \end{aligned}$$

- Closed-loop with load:

$$\begin{aligned} y_{ss} &= \frac{AK_{cl}}{1+AK_{cl}}r_{ss}+\frac{B}{1+AK_{cl}}w_{ss} \\ \delta y &\approx \frac{B}{1+AK_{cl}}w_{ss}. \end{aligned}$$

which is much better than open-loop since $AK_{cl}\gg 1$.

ADVANTAGE OF FEEDBACK: Better disturbance rejection (by factor of $1+AK_{cl}$).

4.3: Advantage of feedback: Sensitivity and dynamic tracking

- The steady-state gain of the open-loop system is: 1.0
- How does this change if the motor constant A changes?

$$A \to A + \delta A$$

$$\begin{aligned} G_{ol} + \delta G_{ol} &= K_{ol}(A + \delta A) \\ &= \frac{1}{A}(A + \delta A) \\ &= 1 + \underbrace{\frac{\delta A}{A}}_{\text{gain error}}. \end{aligned}$$

- In relative terms: $\dfrac{\delta G_{ol}}{G_{ol}} = \dfrac{\delta A}{A} = \underbrace{1.0}_{\text{sensitivity}} \dfrac{\delta A}{A}.$
- Therefore, a 10 % change in A results in a 10 % change in gain. Sensitivity=1.0.
- Steady-state gain of closed-loop system is: $\dfrac{AK_{cl}}{1 + AK_{cl}}.$

$$G_{cl} + \delta G_{cl} = \frac{(A + \delta A)K_{cl}}{1 + (A + \delta A)K_{cl}}.$$

- From calculus (law of total differential)

$$\delta G_{cl} = \frac{\mathrm{d}G_{cl}}{\mathrm{d}A}\delta A$$

or

$$\frac{\delta G_{cl}}{G_{cl}} = \underbrace{\left(\frac{A}{G_{cl}}\frac{\mathrm{d}G_{cl}}{\mathrm{d}A}\right)}_{\text{sensitivity } S_A^{G_{cl}}} \frac{\delta A}{A}.$$

- To calculate this, we first compute

$$\begin{aligned}\frac{\mathrm{d}G_{cl}}{\mathrm{d}A} &= \frac{\mathrm{d}}{\mathrm{d}A}\left(\frac{AK_{cl}}{1+AK_{cl}}\right)\\ &= \frac{(1+AK_{cl})K_{cl}-K_{cl}(AK_{cl})}{(1+AK_{cl})^2}\\ &= \frac{K_{cl}}{(1+AK_{cl})^2}.\end{aligned}$$

- Then,

$$\begin{aligned}\frac{\delta G_{cl}}{G_{cl}} &= \left(\frac{A}{G_{cl}}\frac{\mathrm{d}G_{cl}}{\mathrm{d}A}\right)\frac{\delta A}{A}\\ S_A^{G_{cl}} &= \frac{A}{AK_{cl}/(1+AK_{cl})}\frac{K_{cl}}{(1+AK_{cl})^2}\\ &= \frac{1}{1+AK_{cl}}.\end{aligned}$$

ADVANTAGE OF FEEDBACK: Lower sensitivity to modeling error (by a factor of $1+AK_{cl}$)

Dynamic Tracking

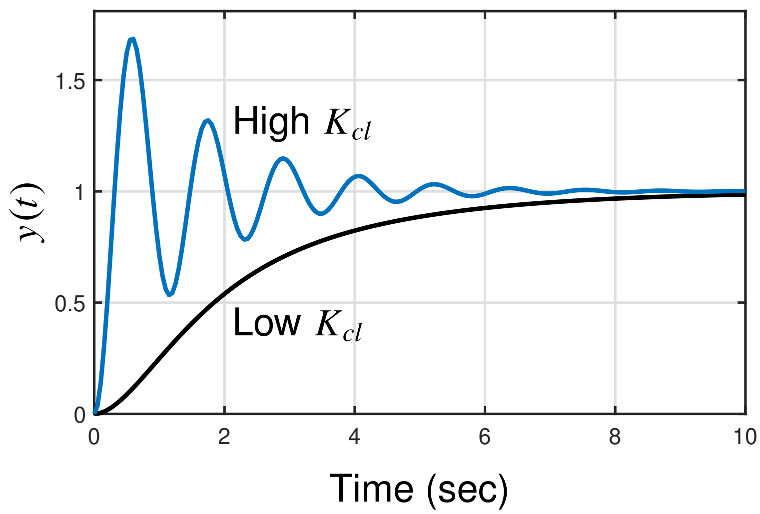
- Steady-state response of closed-loop better than open-loop: Better disturbance rejection, better (lower) sensitivity.
- What about transient response?
- Open-loop system: Poles at roots of $(\tau_1 s+1)(\tau_2 s+1)$

$$\Rightarrow \quad s=-1/\tau_1, \quad s=-1/\tau_2.$$

- Closed-loop system: Poles at roots of $(\tau_1 s+1)(\tau_2 s+1)+AK_{cl}$.

$$\Rightarrow \quad s=\frac{-(\tau_1+\tau_2)\pm\sqrt{(\tau_1+\tau_2)^2-4\tau_1\tau_2(1+AK_{cl})}}{2\tau_1\tau_2}.$$

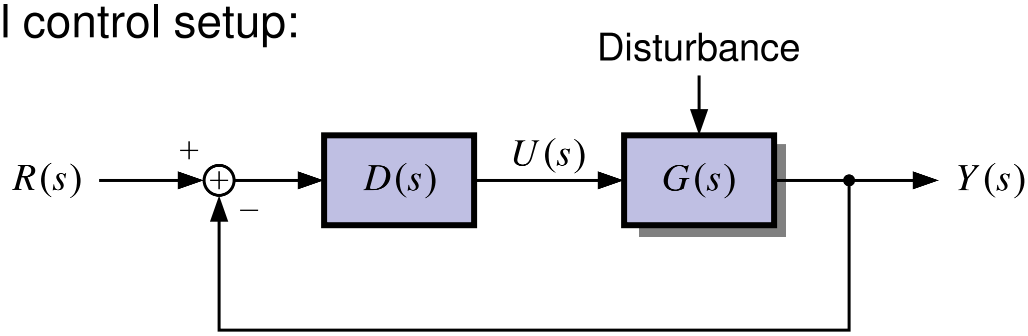
- ***FEEDBACK MOVES POLES***
  - System may have faster/slower response
  - System may be more/less damped
  - System may become unstable!!!
  - Often a high gain K_{cl} results in instability.
  - For this dc motor example, we can get step responses of the following form:



- So, we see the first potential downside of feedback—if the controller is not well designed, it may make the plant's response worse than it was to begin with.
- Designing controllers is a main focus of the rest of this course (and of follow-on courses). It's not a trivial task.
- But, we can get a really good start toward improving the dynamic response of the closed-loop system with a very simple controller
- We look next at the PID controller, then return to exploring (potential) advantages of feedback.

4.4: Proportional-integral-derivative (PID) control (a)

- General control setup:



- Need to design controller $D(s)$.
- One option is PID (Proportional Integral Derivative) control design.
  - Extremely popular. 90^+ % of all fielded controllers are PID.
  - Doesn't mean that they are great, just popular.
- We just saw proportional control where $u(t) = K_p e(t)$, or $D(s) = K_p$.
- Proportional control tends to increase speed of response, but:
  - Can allow non-zero steady-state error.
  - Can result in larger transient overshoot.
  - May not eliminate a constant disturbance.
- Integral control, where $D(s) = \dfrac{K_i}{s}$, can eliminate steady-state error,
  - But, transient response can get worse, and
  - Stability margins can get worse.
- Derivative control, where $D(s) = K_d s$, can reduce oscillations in dynamic response, but
  - Steady-state error can get worse.
- In the next sections, we look at each of these controllers separately, then consider how to use them together.

Proportional control

- Proportional controllers compute the control effort such that

$$u(t) = K_p(r(t) - y(t)) = K_p e(t) \qquad \dots \qquad D(s) = K_p.$$

IDEA: For plants with positive gain, if $e(t) = r(t) - y(t) > 0$, then I'm not "trying hard enough." Multiply error by (positive) K_p to "try harder."

- Also, if $e(t) < 0$, then I've tried too hard already. Multiply (negative) error by (positive) gain K_p to try to pull response back.

EXAMPLE: Determine behavior of closed-loop poles for the dc motor.

$$\frac{Y(s)}{R(s)} = \frac{AK_p}{(\tau_1 s + 1)(\tau_2 s + 1) + AK_p}.$$

- Poles are roots of $(\tau_1 s + 1)(\tau_2 s + 1) + AK_p$.
- Without feedback, $K_p \to 0$.

$$s_1 = -1/\tau_1, \qquad s_2 = -1/\tau_2.$$

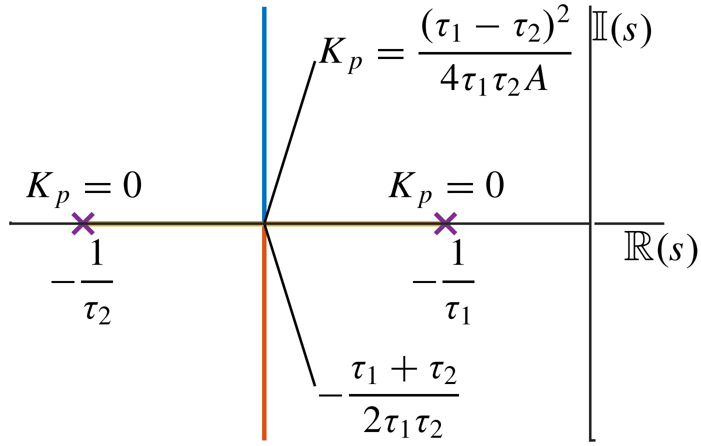
- With feedback,

$$\begin{aligned}\frac{Y(s)}{R(s)} &= \frac{AK_p}{(\tau_1 s + 1)(\tau_2 s + 1) + AK_p} \\ &= \frac{AK_p}{\tau_1\tau_2 s^2 + (\tau_1 + \tau_2)\, s + \left(1 + AK_p\right)}.\end{aligned}$$

- Solving for root locations gives

$$s_1,\ s_2 = \frac{-(\tau_1 + \tau_2) \pm \sqrt{(\tau_1 + \tau_2)^2 - 4\tau_1\tau_2(1 + AK_p)}}{2\tau_1\tau_2}.$$

- We can plot the locations of the poles (a "root locus" plot) parametrically as K_p changes



- For $0 < K_p < \dfrac{(\tau_1 - \tau_2)^2}{4\tau_1\tau_2 A}$, poles move horizontally toward each other along the real axis.
  - Rise time gets faster since dominant pole moves farther from origin, natural frequency increases.
  - Settling time gets faster since real part of dominant pole moves farther from origin.
  - Damping remains same (no overshoot).
- For $K_p > \dfrac{(\tau_1 - \tau_2)^2}{4\tau_1\tau_2 A}$, the poles gain imaginary part.
  - Settling time remains same since real part of pole locations is unchanged.
  - Rise time decreases since natural frequency increases.
  - Overshoot increases since damping ratio decreases.
- For systems having more poles than this example, increasing K_p often leads to instability.
- How do we improve accuracy, but keep stability?

4.5: Proportional-integral-derivative (PID) control (b)

Integral and proportional-integral control

- Pure integral controllers compute the control effort such that:

$$u(t) = \frac{K_p}{T_i}\int_0^t e(\tau)\,\mathrm{d}\tau, \qquad D(s) = \frac{K_p}{T_i s}.$$

  - T_i = "Integral time" = time for output = K_p with input $e(t) = 1(t)$.
  - An alternate formulation has

$$u(t) = K_i\int_0^t e(\tau)\,\mathrm{d}\tau, \qquad D(s) = \frac{K_i}{s}.$$

- Integral feedback can give nonzero control even at points of time when $e = 0$ because of "memory."
  - In many cases this can eliminate steady-state error to step-like reference inputs and step-like disturbances.

IDEA: To avoid instability or oscillations with proportional control, the proportional gain K_p must be kept "small."

- But, then when error gets small, we no longer try very hard to correct it—leads to finite steady-state error.
- Also, some nonlinearities (*e.g.*, coulombic friction) can cause output to get stuck even if control effort is nonzero.
- Integral control can help: If we integrate the error signal, the integrated value will grow over time if the error is "stuck".
- This increases the control signal $u(t)$ until the error starts decreasing—making the error converge to zero.

EXAMPLE: Substitute: $u(t) = \frac{K_p}{T_i}\int_0^t (r(\tau) - y(\tau))\,\mathrm{d}\tau$ into dc-motor eqs.

$$\tau_1\tau_2\ddot{y}(t) + (\tau_1+\tau_2)\dot{y}(t) + y(t) = A\left[\frac{K_p}{T_i}\int_0^t (r(\tau)-y(\tau))\,\mathrm{d}\tau\right] + Bw(t).$$

- Differentiate,

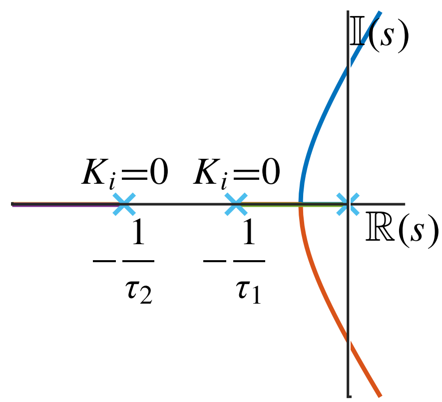
$$\tau_1\tau_2\dddot{y}(t) + (\tau_1+\tau_2)\ddot{y}(t) + \dot{y}(t) = \frac{AK_p}{T_i}(r(t)-y(t)) + B\dot{w}(t)$$

$$\tau_1\tau_2\dddot{y}(t) + (\tau_1+\tau_2)\ddot{y}(t) + \dot{y}(t) + \frac{AK_p}{T_i}y(t) = \frac{AK_p}{T_i}r(t) + B\dot{w}(t).$$

- If $r(t) = \text{cst}$, $w(t) = \text{cst}$, $\dot{w}(t) = 0$,

$$\frac{AK_p}{T_i}y_{ss} = \frac{AK_p}{T_i}r_{ss} \quad \Rightarrow \quad \text{no error.}$$

- Steady-state tracking improves, but dynamic response degrades, especially after poles leave real axis.
  - Very oscillatory; possibly unstable.



- Can be improved by adding proportional term to integral term.

$$u(t) = K_p e(t) + \frac{K_p}{T_I}\int_0^t e(\tau)\,\mathrm{d}\tau, \qquad D(s) = K_p\left(1+\frac{1}{T_i s}\right).$$

- Poles are at the roots of

$$\tau_1\tau_2 s^3 + (\tau_1+\tau_2)s^2 + (1+AK_p)s + \frac{AK_p}{T_i} = 0.$$

  Two degrees of freedom.

Derivative and proportional-derivative control

- Pure derivative controllers compute the control effort such that:

$$u(t) = K_p T_d \dot{e}(t), \qquad D(s) = K_p T_d s,$$

  where T_d = "derivative time".

- An alternate formulation has

$$u(t) = K_d \dot{e}(t), \qquad D(s) = K_d s.$$

IDEA: Would like to anticipate "momentum," which mechanically is proportional to velocity, and subtract out its predicted contribution.

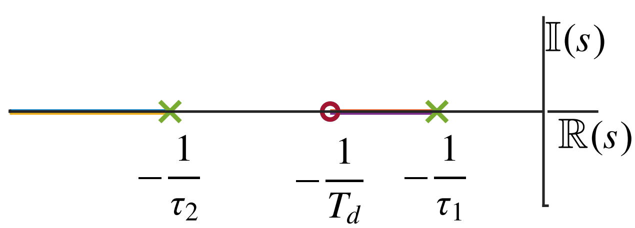
- Contribution to control effort acts as braking force when approaching reference value quickly.

WARNING: *PURE DERIVATIVE CONTROL IMPRACTICAL SINCE DERIVATIVE MAGNIFIES SENSOR NOISE!*

- Practical version = "lead control," which we will study later.
- Derivative control tends to stabilize a system.
- Does nothing to reduce constant error! If $\dot{e}(t) = 0$, then $u(t) = 0$, even if $e(t) \neq 0$.
- Motor control: Poles at roots of $\tau_1\tau_2 s^2 + (\tau_1 + \tau_2 + AK_pT_d)s + 1 = 0$.
  - T_d enters ζ term, can make damping better.
- PD = Proportional plus derivative control where

$$D(s) = K_p(1 + T_d s) \qquad \text{or} \qquad D(s) = K_p + K_d s.$$

- Root locus for dc motor, PD control.



- Great damping, possibly poor steady-state error.

4.6: Proportional-integral-derivative (PID) control (c)

Proportional Integral Derivative Control

- $D(s) = K_p\left(1 + \frac{1}{T_i s} + T_d s\right)$ or $D(s) = K_p + K_i \frac{1}{s} + K_d s.$
- Need ways to design parameters $\{K_p,\, T_i,\, T_d\}$ or $\{K_p,\, K_i,\, K_d\}$.
- In general (*i.e.,* not always),

$$K_p,\ T_i\ \uparrow \Longleftrightarrow \text{error}\ \downarrow,\ \text{stability}\ \downarrow$$

$$T_d\ \uparrow \Longleftrightarrow \text{stability}\ \uparrow$$

- For speed control problem,

$$u(t) = K_p\left[(r(t) - y(t)) + \frac{1}{T_i}\int_0^t (r(\tau) - y(\tau))\,\mathrm{d}\tau + T_d(\dot{r}(t) - \dot{y}(t))\right].$$

(math happens). Solve for poles

$$\tau_1\tau_2 T_i s^3 + T_i((\tau_1 + \tau_2) + AK_pT_d)s^2 + T_i(1 + AK_p)s + AK_p = 0$$

$$s^3 + \left[\frac{\tau_1 + \tau_2 + AK_pT_d}{\tau_1\tau_2}\right]s^2 + \left[\frac{1 + AK_p}{\tau_1\tau_2}\right]s + \frac{AK_p}{\tau_1\tau_2 T_i} = 0.$$

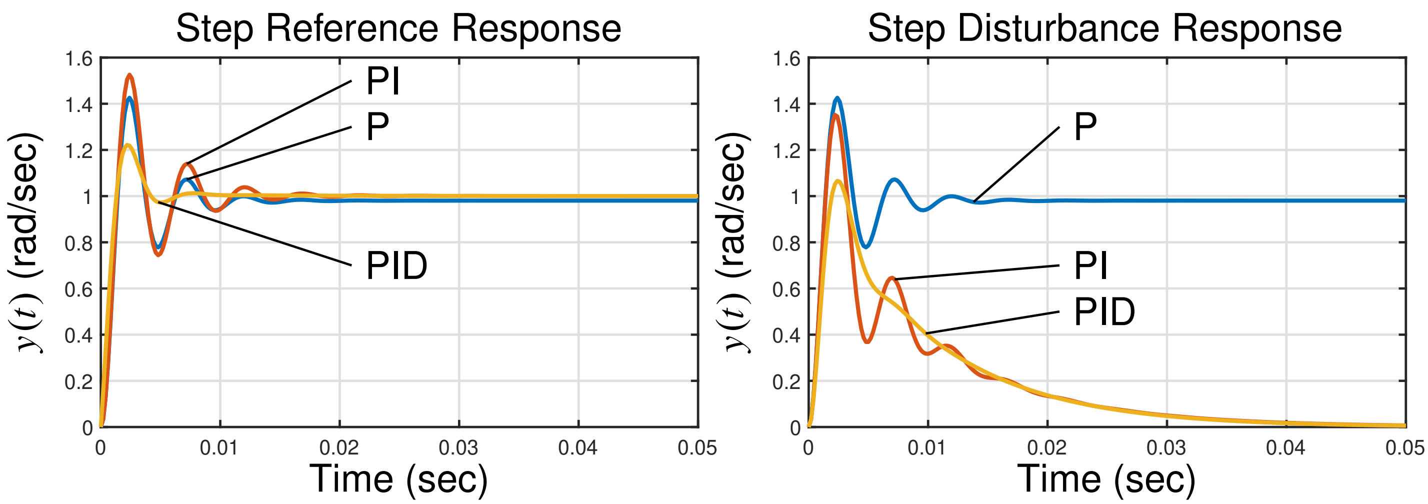
- Three coefficients, three parameters. We can put poles anywhere! Complete control of dynamics in this case.
- Entire transfer functions are:

$$\frac{Y(s)}{W(s)} = \frac{T_i B s}{T_i\tau_1\tau_2 s^3 + T_i(\tau_1 + \tau_2)s^2 + T_i(1 + AK_p)s + AK_p}.$$

$$\frac{Y(s)}{R(s)} = \frac{AK_p(T_i s + 1)}{T_i\tau_1\tau_2 s^3 + T_i(\tau_1 + \tau_2)s^2 + T_i(1 + AK_p)s + AK_p}.$$

- We can plot responses in MATLAB:

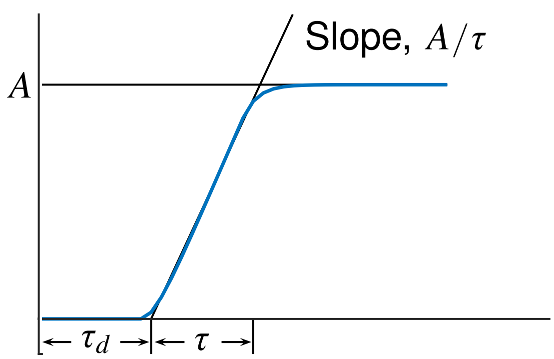
```
num = [TI*B 0];
den = [TI*TAU1*TAU2 TI*(TAU1+TAU2) TI*(1+A*KP) A*KP];
step(num, den)
```



Ziegler–Nichols tuning of PID controllers

- "Rules-of-thumb" for selecting K_p, T_i, T_d.
- Not optimal in any sense—but often provide good performance.

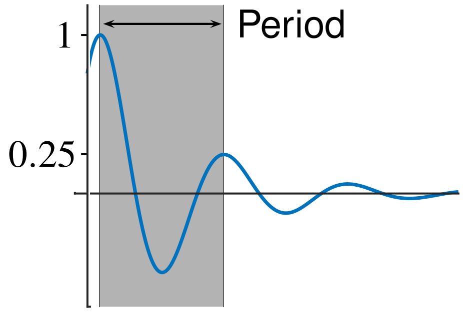
METHOD I: If system has step response like this,



$$\frac{Y(s)}{U(s)} = \frac{Ae^{-\tau_d s}}{\tau s + 1},$$

(first-order system plus delay)

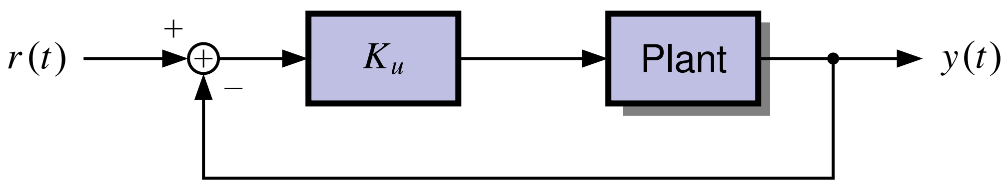
- We can easily identify A, τ_d, τ from this step response.
- Don't need complex model!
- Tuning criteria: Ripple in impulse response decays to 25 % of its value in one period of ripple



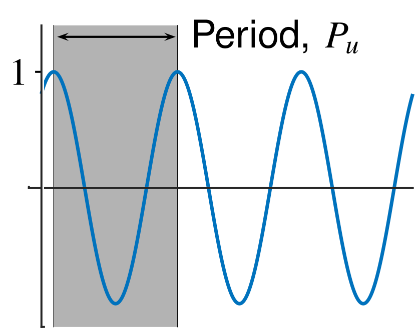
RESULTING TUNING RULES:

P	PI	PID
$K_p = \frac{\tau}{A\tau_d}$	$K_p = \frac{0.9\tau}{A\tau_d}$ $T_i = \frac{\tau_d}{0.3}$	$K_p = \frac{1.2\tau}{A\tau_d}$ $T_i = 2\tau_d$ $T_d = 0.5\tau_d$

METHOD II: Configure system as



- Turn up gain K_u until system produces oscillations (on stability boundary) K_u = "ultimate gain."



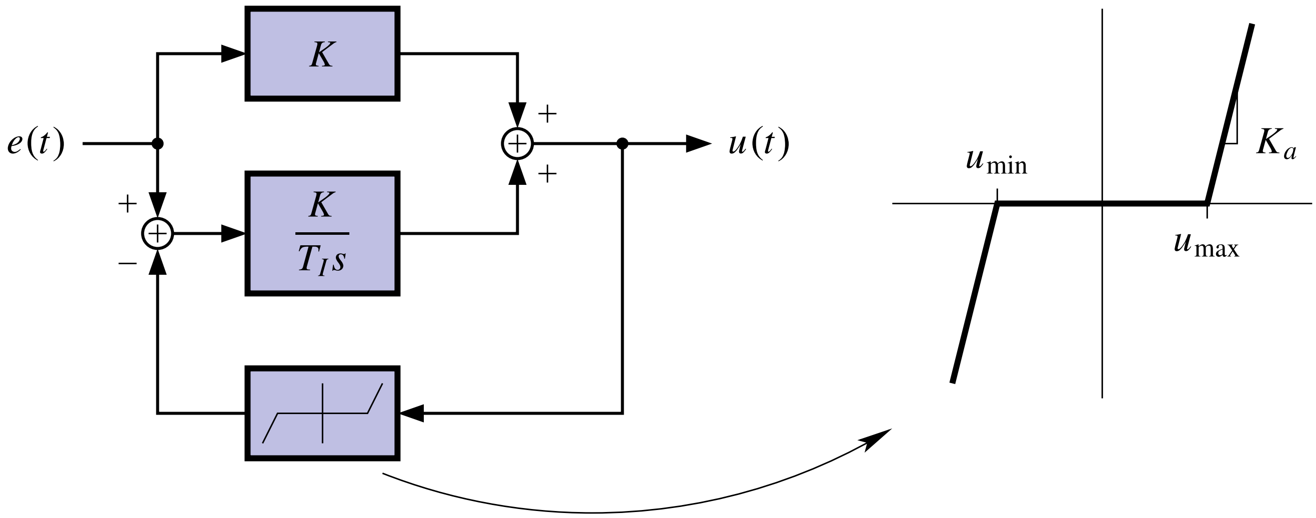
RESULTING TUNING RULES:

P	PI	PID
$K_p = 0.5K_u$	$K_p = 0.45K_u$ $T_i = \frac{1}{1.2}P_u$	$K_p = 0.6K_u$ $T_i = 0.5P_u$ $T_d = \frac{P_u}{8}$

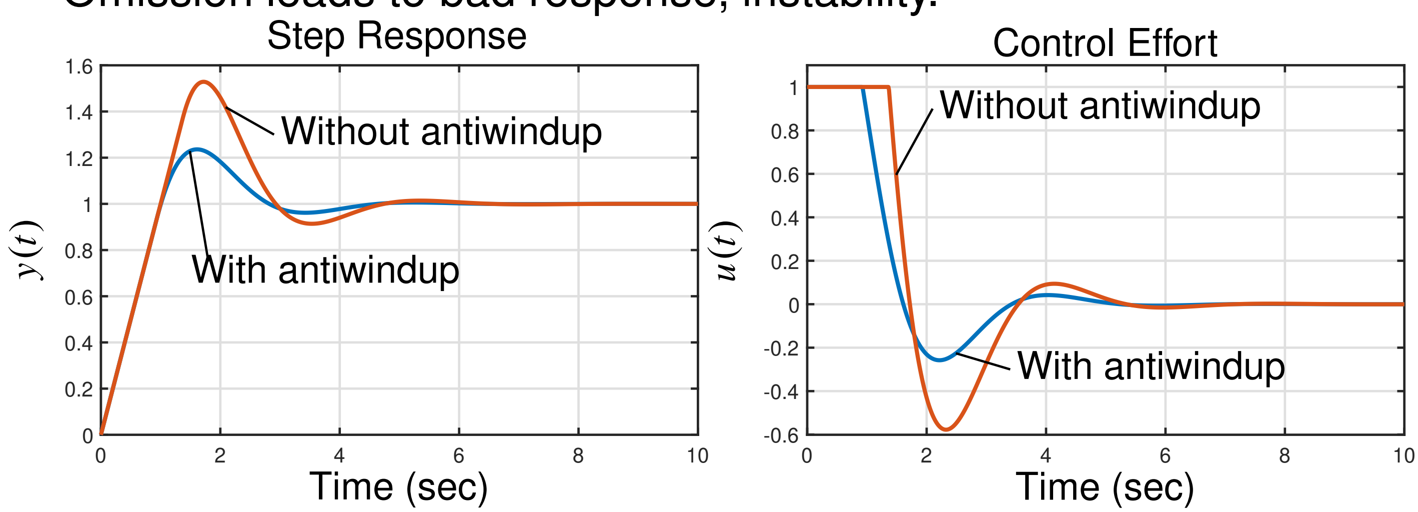
Practical Problem: Integrator Overload

- Integrator in PI or PID control can cause problems.
- For example, suppose there is saturation in the actuator.
  - Error will not decrease.
  - Integrator will integrate a constant error and its value will "blow up."

- Solution = "integrator anti-windup." Turn off integration when actuator saturates.



- Doing this is *NECESSARY* in any practical implementation.
- Omission leads to bad response, instability.

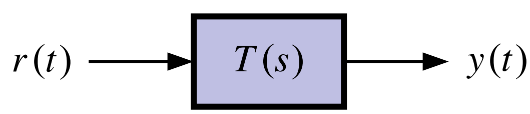


4.7: Steady-state error (a)

- System error is any difference between $r(t)$ and $y(t)$. Two sources:
  1. Imprecise tracking of $r(t)$.
  2. Disturbance $w(t)$ affecting the system output.

Steady-state error (w.r.t. reference input)

- We have already seen examples of CL systems that have some tracking error (proportional ctrl) or not (integral ctrl) to a step input. We will formalize this concept here.
- Start with very general control structure with "closed-loop" transfer function $T(s)$ that computes $Y(s)$ from $R(s)$:



- We don't care what is inside the box $T(s)$. It could be any block diagram, and we may need to compute $T(s)$ from the block diagram.
- The error is

$$\begin{aligned} E(s) &= R(s) - Y(s) \\ &= R(s) - T(s)R(s) \\ &= [1 - T(s)]\, R(s). \end{aligned}$$

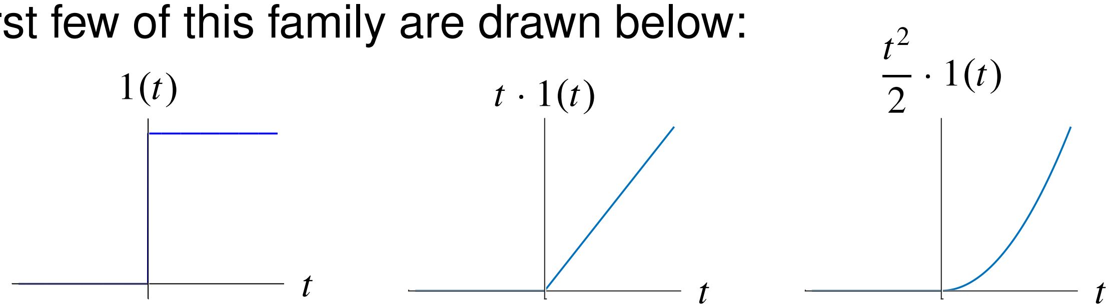
- Assume conditions of final value theorem are satisfied (*i.e.*, $[1 - T(s)]R(s)$ has poles only in LHP except perhaps for a single pole at the origin)

$$e_{ss} = \lim_{t\to\infty} e(t) = \lim_{s\to 0} s[1 - T(s)]R(s).$$

- When considering steady-state error to different reference inputs, we restrict ourselves to inputs of the type

$$r(t) = \frac{t^k}{k!}1(t); \quad R(s) = \frac{1}{s^{k+1}}.$$

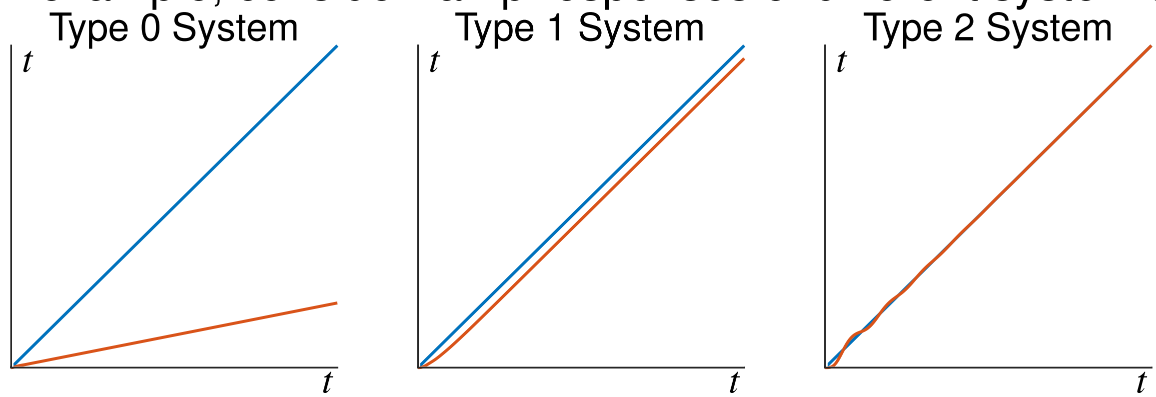
- The first few of this family are drawn below:



- As k increases, reference tracking is progressively harder. (It is easier to track a constant reference than it is to track a moving reference.)
- We define a concept called system type to describe the ability of the closed-loop system to track inputs of different kinds.
  - If system type = 0, constant steady-state error for step input, infinite s.s. error for ramp or parabolic input.
  - If system type = 1, no steady-state error for step input, constant s.s. error for ramp input, infinite s.s. error for parabolic input.
  - If system type = 2, no steady-state error for step or ramp inputs, constant s.s. error for parabolic inputs.
  - And so forth, for higher-order system types.
- To find the system type in general, we must compute the following equation for values of $k = 0, 1, \ldots$ until we calculate a finite nonzero value for e_{ss}:

$$e_{ss} = \lim_{s \to 0} \frac{1 - T(s)}{s^k} = \begin{cases} 0, & \text{type} > k; \\ \text{constant}, & \text{type} = k; \\ \infty, & \text{type} < k. \end{cases}$$

- As an example, consider ramp responses of different system types:



DANGER: Higher order sounds better but they are harder to stabilize and design. Transient response may be poor.

- Summary table for computing steady-state error for different system types (where the limits evaluate to finite nonzero values)

Steady-state tracking errors e_{ss} for generic closed-loop $T(s)$

Sys. Type	Step Input	Ramp Input	Parabola Input
Type 0	$\lim_{s\to 0}(1-T(s))$	∞	∞
Type 1	0	$\lim_{s\to 0}\dfrac{1-T(s)}{s}$	∞
Type 2	0	0	$\lim_{s\to 0}\dfrac{1-T(s)}{s^2}$

EXAMPLES:

(1) Consider $T(s) = \dfrac{s+1}{(s+2)(s+3)}$. What is the system type?

- Try $k=0$. Evaluate

$$\lim_{s\to 0}[1-T(s)] = \lim_{s\to 0}\frac{(s+2)(s+3)-(s+1)}{(s+2)(s+3)} = \frac{5}{6} \neq 0.$$

- Therefore, the system type is zero, $e_{ss} = 5/6$ to unit step input.

(2) Consider $T(s) = \dfrac{s+6}{(s+2)(s+3)}$. What is the system type?

- Try $k=0$. Evaluate

$$\lim_{s\to 0}[1-T(s)] = \lim_{s\to 0}\frac{(s+2)(s+3)-(s+6)}{(s+2)(s+3)} = \frac{0}{6} = 0.$$

- Therefore the system type must be greater than zero.

- Try $k=1$. Evaluate

$$\lim_{s\to 0}\frac{1-T(s)}{s} = \lim_{s\to 0}\frac{1}{s}\frac{(s+2)(s+3)-(s+6)}{(s+2)(s+3)}$$
$$= \lim_{s\to 0}\frac{1}{s}\frac{s^2+4s}{(s+2)(s+3)} = \frac{4}{6} \neq 0.$$

- Therefore, the system is type 1, $e_{ss}=2/3$ to unit ramp input.

(3) Consider $T(s)=\dfrac{5s+6}{(s+2)(s+3)}$. What is the system type?

- Try $k=0$. Evaluate

$$\lim_{s\to 0}[1-T(s)] = \lim_{s\to 0}\frac{(s+2)(s+3)-(5s+6)}{(s+2)(s+3)} = \frac{0}{6} = 0.$$

  Therefore, the system type must be greater than zero.

- Try $k=1$. Evaluate

$$\lim_{s\to 0}\frac{1-T(s)}{s} = \lim_{s\to 0}\frac{1}{s}\frac{(s+2)(s+3)-(5s+6)}{(s+2)(s+3)}$$
$$= \lim_{s\to 0}\frac{1}{s}\frac{s^2}{(s+2)(s+3)} = \frac{0}{6} = 0.$$

  Therefore, the system type must be greater than one.

- Try $k=2$. Evaluate

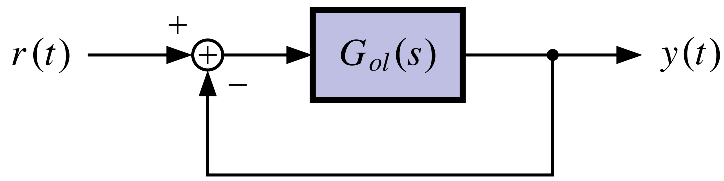
$$\lim_{s\to 0}\frac{1-T(s)}{s^2} = \lim_{s\to 0}\frac{1}{s^2}\frac{(s+2)(s+3)-(5s+6)}{(s+2)(s+3)}$$
$$= \lim_{s\to 0}\frac{1}{s^2}\frac{s^2}{(s+2)(s+3)} = \frac{1}{6} \neq 0.$$

  Therefore, the system type is two, $e_{ss}=1/6$ to unit parabola input.

4.8: Steady-state error w.r.t. reference input, unity feedback

WARNING: The following method is a special case of the above general method. Always use the appropriate method for the problem at hand!

- Unity-feedback is when the control system looks like:



- That is, the feedback loop has a gain of exactly one.
- If we are fortunate enough to be considering a unity-feedback scenario, the prior rules have a simpler solution.
- But, if there are any dynamics in the feedback loop, we DO NOT have a unity-feedback system, and must use the more general rules from Section 4.7.
- For unity-feedback systems, there are some important simplifications:

$$\begin{aligned} T(s) &= \frac{G_{ol}(s)}{1+G_{ol}(s)} \\ 1-T(s) &= \left[\frac{1+G_{ol}(s)}{1+G_{ol}(s)}\right] - \frac{G_{ol}(s)}{1+G_{ol}(s)} \\ &= \frac{1}{1+G_{ol}(s)}. \end{aligned}$$

So,

$$E(s) = \frac{1}{1+G_{ol}(s)}R(s).$$

- For test inputs of the type $R(s) = \dfrac{1}{s^{k+1}}$

$$e_{ss} = \lim_{s\to 0} sE(s) = \lim_{s\to 0} \frac{1}{[1+G_{ol}(s)]s^k}.$$

- For a system that is type 0,

$$e_{ss} = \lim_{s\to 0} \frac{1}{1+G_{ol}(s)} = \frac{1}{1+K_p}, \qquad K_p = \lim_{s\to 0} G_{ol}(s).$$

- For a system that is type 1,

$$e_{ss} = \lim_{s\to 0} \frac{1}{1+G_{ol}(s)}\frac{1}{s} = \lim_{s\to 0} \frac{1}{sG_{ol}(s)} = \frac{1}{K_v}, \qquad K_v = \lim_{s\to 0} sG_{ol}(s).$$

- For a system that is type 2,

$$e_{ss} = \lim_{s\to 0} \frac{1}{1+G_{ol}(s)}\frac{1}{s^2} = \lim_{s\to 0} \frac{1}{s^2G_{ol}(s)} = \frac{1}{K_a}, \qquad K_a = \lim_{s\to 0} s^2G_{ol}(s).$$

These formulas meaningful **only** for unity-feedback!

$K_p = \lim_{s\to 0} G_{ol}(s)$.	"position error constant"
$K_v = \lim_{s\to 0} sG_{ol}(s)$.	"velocity error constant"
$K_a = \lim_{s\to 0} s^2G_{ol}(s)$.	"acceleration error constant"

Steady-state tracking errors e_{ss} for unity-feedback case **only**.

Sys. Type	Step Input	Ramp Input	Parabola Input
Type 0	$\frac{1}{1+K_p}$	∞	∞
Type 1	0	$\frac{1}{K_v}$	∞
Type 2	0	0	$\frac{1}{K_a}$

EXAMPLES:

(1) Consider $G_{ol}(s) = \dfrac{s+1}{(s+2)(s+3)}$. What is the system type?

$$G_{ol}(0) = \frac{1}{2\cdot 3} = \frac{1}{6}.$$

- Therefore, system type$=0$, e_{ss} to unit step$=\dfrac{1}{1+1/6}=\dfrac{6}{7}$.

(2) Consider $G_{ol}(s)=\dfrac{(s+1)(s+10)(s-5)}{(s^2+3s)(s^4+s^2+1)}$. What is the system type?

$$G_{ol}(0)=\frac{1\cdot 10\cdot(-5)}{0\cdot 1}=\infty \qquad \Rightarrow \text{Type} > 0.$$

$$sG_{ol}(s)=\frac{(s+1)(s+10)(s-5)}{(s+3)(s^4+s^2+1)}$$

$$sG_{ol}(s)|_{s=0}=\frac{1\cdot 10\cdot(-5)}{3\cdot 1}=\frac{-50}{3}.$$

- Therefore, system type$=1$, e_{ss} to unit ramp$=\dfrac{-3}{50}$.

(3) Consider $G_{ol}(s)=\dfrac{s^2+2s+1}{s^4+3s^3+2s^2}$. What is the system type?

$$G_{ol}(0)=\frac{1}{0}=\infty \qquad \Rightarrow \text{Type} > 0$$

$$sG_{ol}(s)=\frac{s^2+2s+1}{s^3+3s^2+2s}$$

$$sG_{ol}(s)|_{s=0}=\frac{1}{0}=\infty \qquad \Rightarrow \text{Type} > 1$$

$$s^2G_{ol}(s)=\frac{s^2+2s+1}{s^2+3s+2}$$

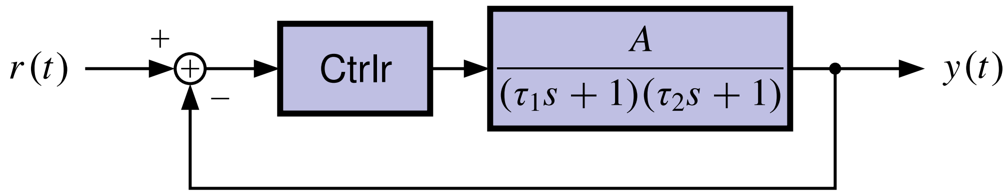
$$s^2G_{ol}(s)\Big|_{s=0}=\frac{1}{2} \qquad \Rightarrow \text{Type} = 2.$$

- Therefore, system type$=2$, e_{ss} to unit parabola$=2$.

KEY POINT: *Open-loop $G_{ol}(s)$ tells us about closed-loop s.s. response.*

KEY POINT: *For unity-feedback systems, number of poles of $G_{ol}(s)$ at $s=0$ is equal to the system type.*

EXAMPLE: DC-motor example with proportional control, $D(s) = K_p$.



$$D(s)G(s) = \frac{K_p A}{(\tau_1 s + 1)(\tau_2 s + 1)}, \qquad \lim_{s \to 0} D(s)G(s) = K_p A.$$

- So system is type 0, with s.s. error to step input of $\dfrac{1}{1 + K_p A}$.
- This agrees with prior results.

EXAMPLE: DC-motor example with PI control, $D(s) = K_p \left[1 + \dfrac{1}{T_i s} \right]$.

$$D(s)G(s) = \frac{K_p A + \frac{K_p A}{T_i s}}{(\tau_1 s + 1)(\tau_2 s + 1)}.$$

$$\lim_{s \to 0} D(s)G(s) = \infty$$

$$\lim_{s \to 0} s D(s)G(s) = \frac{K_p A}{T_i}.$$

- System is type 1, with s.s. error to ramp input of $\dfrac{T_i}{K_p A}$.

EXAMPLE: DC-motor with two-integrator controller,

$$D(s) = K_p \left[1 + \frac{1}{T_i s} + \frac{1}{T_i s^2} \right].$$

$$D(s)G(s) = \frac{K_p A + \frac{K_p A}{T_i s} + \frac{K_p A}{T_i s^2}}{(\tau_1 s + 1)(\tau_2 s + 1)}$$

$$\lim_{s \to 0} s^2 D(s)G(s) = \frac{K_p A}{T_i}.$$

- System is type 2, with s.s. error to parabolic input of $\dfrac{T_i}{K_p A}$.

4.9: Steady-state error w.r.t. disturbance

- Recall, system error is any difference between $r(t)$ and $y(t)$.
- We have just spent considerable time considering differences between $r(t)$ and $y(t)$ because the system is not capable of tracking $r(t)$ perfectly.
  - This is an issue with $T(s)$ for the general case, or $G_{ol}(s)$ for the unity-feedback case.
- But, another source of steady-state error can be uncontrolled inputs to the system—disturbances.

EXAMPLE: Consider a vehicle cruise-control system. We may set the reference speed $r(t) = 55$ mph, but we find that the steady-state vehicle speed y_{ss} is affected by wind and road grade (in addition to the cruise-control system's ability to track the reference input.)

- We can think of a system's overall response to both the reference input and the disturbance input as

$$Y(s) = T(s)R(s) + T_w(s)W(s).$$

- We find the system type with regard to the reference input by examining $T(s)$; similarly, we find the system type with respect to the disturbance input by examining $T_w(s)$.
- Due to linearity, we can consider these two problems separately.
  - When thinking about system type with respect to reference input, we consider $W(s) = 0$ and follow the procedures outlined earlier.
  - When thinking about system type with respect to disturbance input, we consider $R(s) = 0$ and follow the procedure below.

- We do not wish the output to have *ANY* disturbance term in it, so the output error due to the disturbance is equal to whatever output is caused by the disturbance.

$$e_{ss} = r_{ss} - y_{ss} = 0 - \lim_{s\to 0} sT_w(s)W(s).$$

- We say that the system is type 0 with respect to disturbance if it has nonzero steady-state error when the disturbance is $1/s$.
  - Type 0 system (w.r.t. disturbance) has constant $e_{ss} = -\lim_{s\to 0} T_w(s)$.
- We say that a system is type 1 with respect to disturbance if it has nonzero steady-state error when the disturbance is $1/s^2$.
  - Type 1 system (w.r.t. disturbance) has constant $e_{ss} = -\lim_{s\to 0} \dfrac{T_w(s)}{s}$.
- We say that a system is type 2 with respect to disturbance if it has nonzero steady-state error when the disturbance is $1/s^3$.
  - Type 2 system (w.r.t. disturbance) has constant $e_{ss} = -\lim_{s\to 0} \dfrac{T_w(s)}{s^2}$.
- Summary table for computing steady-state error for different system types with respect to disturbance (where the limits evaluate to finite nonzero values)

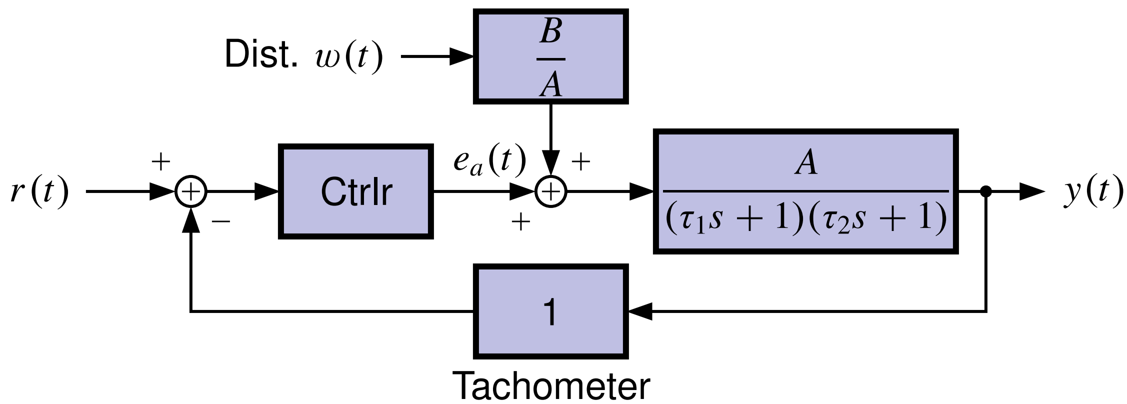
Steady-state errors e_{ss} due to disturbance for generic closed-loop $T_w(s)$

Sys. Type	Step Input	Ramp Input	Parabola Input
Type 0	$-\lim_{s\to 0} T_w(s)$	$-\infty$	$-\infty$
Type 1	0	$-\lim_{s\to 0} \dfrac{T_w(s)}{s}$	$-\infty$
Type 2	0	0	$-\lim_{s\to 0} \dfrac{T_w(s)}{s^2}$

NOTE: There are no special cases for unity feedback when computing system type with respect to disturbance.

- You must always compute the closed-loop transfer function $T_w(s) = Y(s)/W(s)$, and then perform the tests.

EXAMPLE: Consider the motor-control problem from before.



- We'll use a proportional controller with gain K_{cl}, so

$$e_a(t) = K_{cl}(r(t) - y(t)).$$

- When we looked at the example earlier, we found that

$$Y(s) = \frac{AK_{cl}}{(\tau_1 s+1)(\tau_2 s+1)+AK_{cl}}R(s) + \frac{B}{(\tau_1 s+1)(\tau_2 s+1)+AK_{cl}}W(s).$$

- From this equation, we gather

$$T(s) = \frac{AK_{cl}}{(\tau_1 s+1)(\tau_2 s+1)+AK_{cl}}$$

$$T_w(s) = \frac{B}{(\tau_1 s+1)(\tau_2 s+1)+AK_{cl}}.$$

- Starting with system type 0, test for finite nonzero steady-state error:

$$e_{ss} = -\lim_{s\to 0} T_w(s) = \frac{-B}{1+AK_{cl}} \neq 0.$$

- Therefore, the system is type 0 with respect to disturbance (it's also type 0 with respect to reference input, but the two system types ARE NOT the same in general).