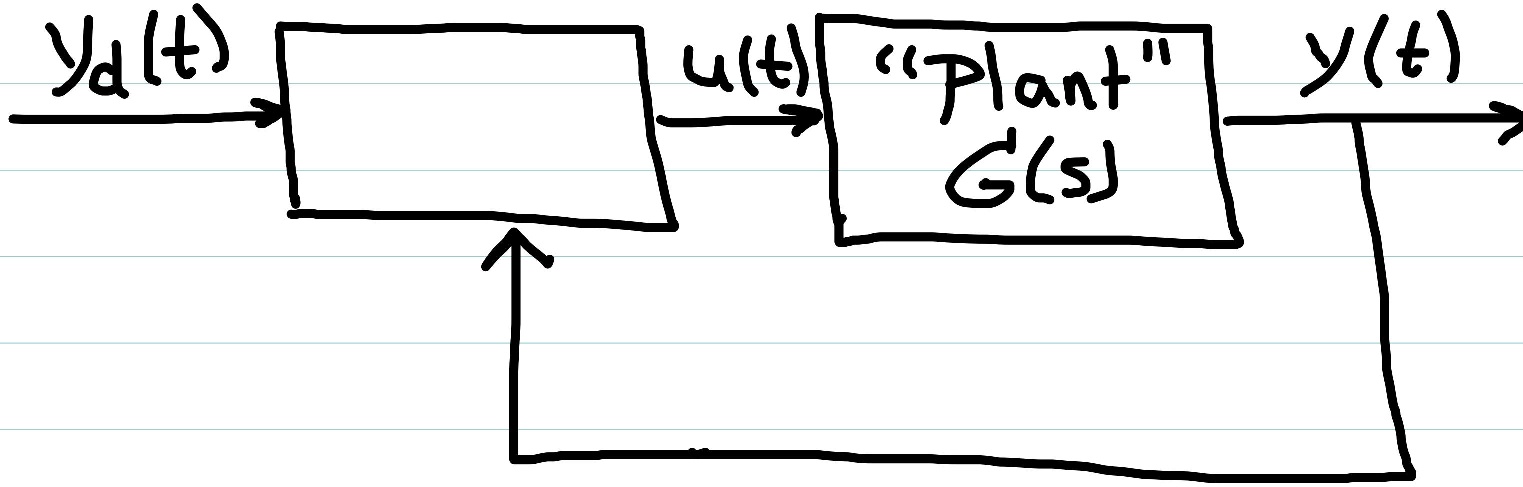
Feedback Control (finally!)

⇒ **Automatically** generate inputs $u(t)$ so that output $y(t)$ tracks "desired output" $y_d(t)$ as closely as possible

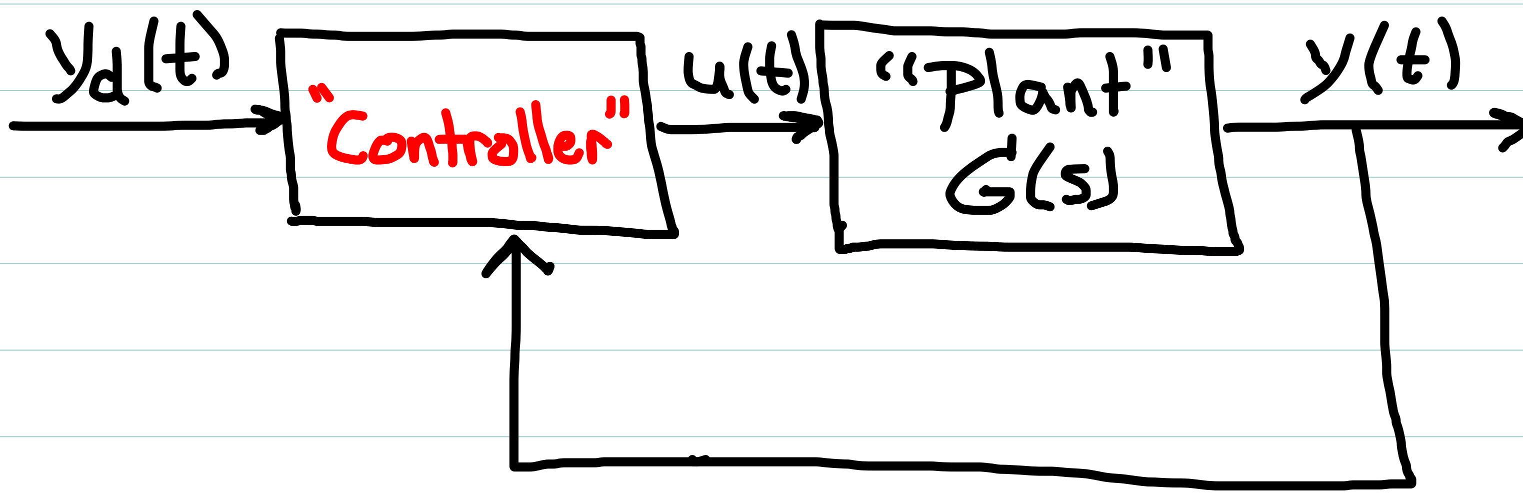
⇒ Input determined in real-time by continually comparing $y(t)$ with $y_d(t)$



Feedback Control (finally!)

⇒ **Automatically** generate inputs $u(t)$ so that output $y(t)$ tracks "desired output" $y_d(t)$ as closely as possible

⇒ Input determined in real-time by continually comparing $y(t)$ with $y_d(t)$



Feedback Controllers

=> The controller is a device that we design to compute $u(t)$ from $y_d(t)$ and $y(t)$, to satisfy specified constraints.

=> The relationship between $y_d(t)$, $y(t)$ and $u(t)$ is known as the "control law". This is a mathematical algorithm for computing $u(t)$.

=> For example:

$$u(t) = K\left[y_d(t) - y(t)\right]$$

In this control law, $u(t)$ is proportional to the difference between $y_d(t)$ and $y(t)$.

=> Controllers are implemented as programs (usually in C/C++) on a digital computer onboard the vehicle.

Control Laws

⇒ Control laws can be any mathematical function of $y(t)$ and $y_d(t)$, including differential equations

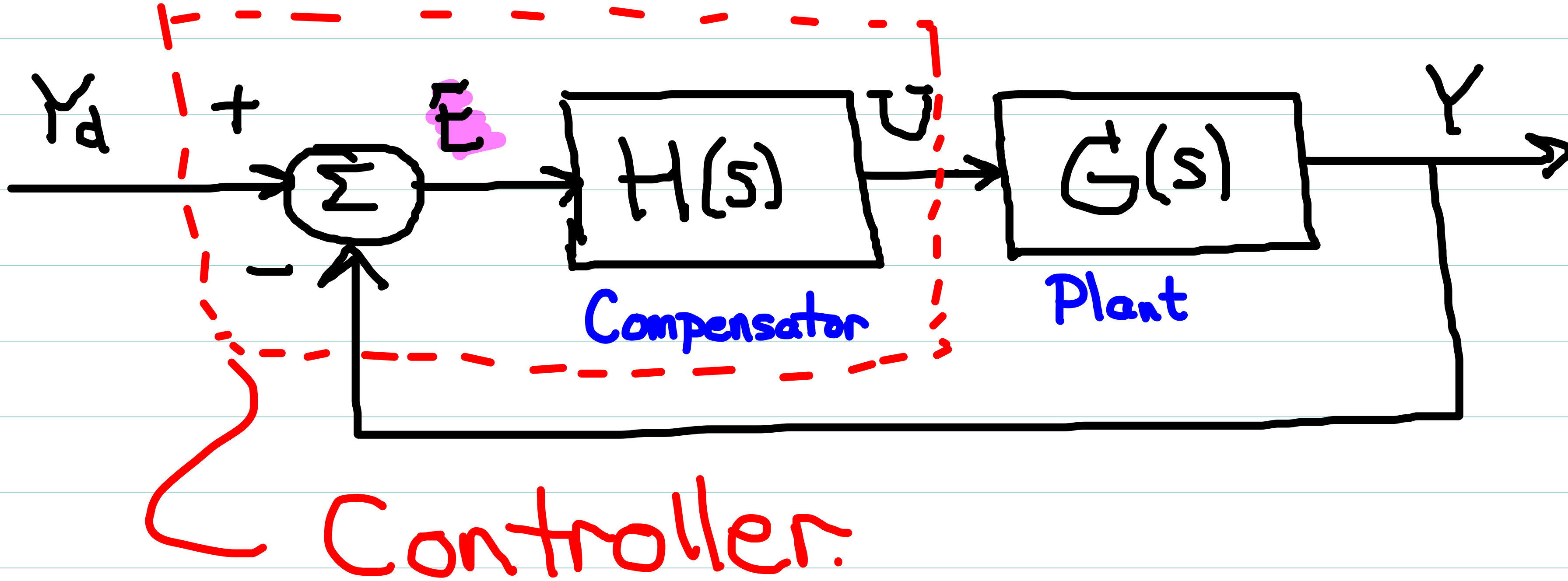
⇒ For example: $H(s) = \dfrac{\beta_1 s + \beta_0}{s + \alpha_0}$

$$\dot{u}(t) + \alpha_0 u(t) = \beta_1 \frac{d}{dt}[\overbrace{y_d(t) - y(t)}^{e(t)}] + \beta_0 [\overbrace{y_d(t) - y(t)}^{e(t)}]$$

⇒ In such cases, we can model the operation of the controller in the same transfer function framework used to model the physical system being controlled.

⇒ The "standard servo loop" is a systematic framework for analyzing these control strategies.

Standard Servo Loop



Action of controller is:

$$U(s) = H(s)E(s)$$

"Tracking error"

where $E(s) = Y_d(s) - Y(s) \Rightarrow \boxed{e(t) = y_d(t) - y(t)}$

however finding u(t) from e(t) requires solving differential equation corresponding to H(s).

Controller Design

$$U(s) = H(s)E(s)$$

$H(s)$ is a new transfer function that we design

It has no physical basis, we create it to solve the control problem for a particular physical system $G(s)$.

There is no unique specification of $H(s)$ for a specific $G(s)$. Many different design tradeoffs which do not have a unique sol'n.

Guiding principle: Use the simplest $H(s)$ (fewest poles + zeros) which will provide desired performance.

Servo Loop Analysis

$$U(s) = H(s)\left[Y_d(s) - Y(s)\right]$$

$$Y(s) = G(s)\,U(s)$$

Circular! Y depends on U, but U depends on Y.

Very tricky to "untangle" the circularity using the governing diff'l eq'ns for G, H.

Laplace makes it easy!

$$Y = GU = GHE = GH(Y_d - Y)$$

$$\Rightarrow (1+GH)Y = GHY_d$$

or

$$Y(s) = \underbrace{\boxed{\left[\frac{G(s)H(s)}{1+G(s)H(s)}\right]}}_{T(s)}\,Y_d(s)$$

T(s) "closed-loop" TF

Loop Transfer Functions

Define

$$L(s) = G(s)H(s)$$ "open-loop" TF

then

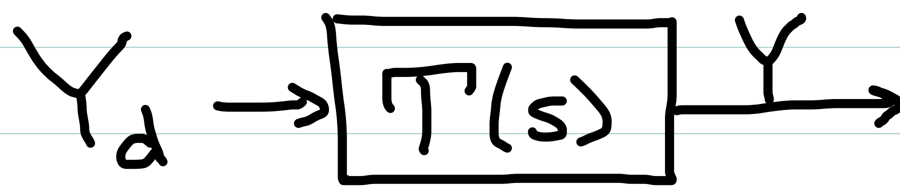
$$\boxed{T(s) = \frac{L(s)}{1+L(s)}}$$ "closed-loop" TF

and

$$Y(s) = L(s)E(s)$$ open-loop dynamics

$$Y(s) = T(s)Y_d(s)$$ closed-loop dynamics

$T(s)$ gives us direct information about system performance



$L(s)$ is an important intermediate quantity in analysis + design

Another Useful relationship

$$E(s) = Y_d(s) - Y(s) = Y_d(s) - T(s)\,Y_d(s)$$

$$= \underbrace{[1 - T(s)]}_{S(s)\text{ : "Sensitivity" TF}} Y_d(s)$$

Note that:

$$S(s) = 1 - T(s) = 1 - \frac{L(s)}{1+L(s)}$$

So:

$$S(s) = \frac{1}{1+L(s)}$$

Thus:

$$S(s) = 1 - T(s) = \frac{1}{1+L(s)}$$

are equivalent, although we will primarily work with the second form.

Final Important Relationship

$U(s) = H(s)E(s)$ [TF model of control law]

$$= \underbrace{\left[H(s)\mathcal{S}(s)\right]}_{R(s)} Y_d(s)$$

$$R(s) = H(s)\mathcal{S}(s) = \frac{H(s)}{1+L(s)}$$

Used to predict control signals which will be generated under ideal conditions

$\Rightarrow$ $y_d(t)$ known perfectly for all $t \geq 0$
$\Rightarrow$ perfect model of system (no errors in model, no disturbances)

$R(s)$ used only theoretically. $H(s)$ is used for actual implementation.

Example:

Suppose $G(s) = \frac{2(s+1)}{s+3}$ $\quad H(s) = \frac{K}{s}$

Then $L = GH = \frac{2K(s+1)}{s(s+3)}$

$$T = \frac{L}{1+L} = \frac{2K(s+1)}{s(s+3)+2K(s+1)} = \frac{2K(s+1)}{s^2+(3+2K)s+2K}$$

$$S = \frac{1}{1+L} = \frac{s(s+3)}{s^2+(3+2K)s+2K}$$

$$R = \frac{H}{1+L} = \frac{K(s+3)}{s^2+(3+2K)s+2K}$$

Three Derived TFs for Feedback Loops

Given $G(s)$ and $H(s)$, we can derive $R(s)$, $S(s)$, $T(s)$ so that:

$$Y_d(s) \longrightarrow \boxed{T(s)} \xrightarrow{Y(s)} \qquad T(s) = \frac{L(s)}{1+L(s)}$$

$$Y_d(s) \longrightarrow \boxed{S(s)} \xrightarrow{E(s)} \qquad S(s) = \frac{1}{1+L(s)}$$

$$Y_d(s) \longrightarrow \boxed{R(s)} \xrightarrow{U(s)} \qquad R(s) = \frac{H(s)}{1+L(s)}$$

$\Rightarrow$ Each of these derived TFs can be analyzed using the same tools developed for $G(s)$.

Uses of derived TF:

⇒ $T(s)$ tells us about actual response of controlled system for specific $y_d(t)$

$$\boxed{Y(s) = T(s)\,Y_d(s)}$$

⇒ $S(s)$ tells us about tracking accuracy for specific $y_d(t)$

$$\boxed{E(s) = S(s)\,Y_d(s)}$$

⇒ $R(s)$ tells us about required input for specific $y_d(t)$:

$$\boxed{U(s) = R(s)\,Y_d(s)}$$

Note: all 3 of these TF have the same denominator, hence same poles!!!

Example use of loop TF:

Suppose $y_d(t) = A\,\mathbb{1}(t)$ (step of magnitude A)

Then:

$$y(t) = A \times \{\text{step response of } T(s)\}$$

$$u(t) = A \times \{\text{step response of } R(s)\}$$

$$e(t) = A \times \{\text{step response of } S(s)\}$$

Note in particular here that:

$$e_{ss}(t) =$$

Example use of loop TF:

Suppose $y_d(t) = A\mathbb{1}(t)$ (step of magnitude A)

Then:

$$y(t) = A \times \{\text{step response of } T(s)\}$$

$$u(t) = A \times \{\text{step response of } R(s)\}$$

$$e(t) = A \times \{\text{step response of } S(s)\}$$

Note in particular here that:

$$e_{ss}(t) = A\,S(0) \qquad \text{(constant)}$$

Thus generally we'd like to make sure $S(0) = 0$ (or at least very small).

Example Application: Tracking Ability

A good feedback loop needs to ensure $|e_{ss}(t)|$ small for a wide variety of $y_d(t)$.

Suppose $y_d(t) = A$ (constant)

Then (assuming all poles of $\mathcal{S}(s)$ at least stable)

$$e_{ss}(t) = A\,\mathcal{S}(\emptyset)$$

So good tracking requires $|\mathcal{S}(\emptyset)|$ small.

Ideally, $\mathcal{S}(\emptyset) = \emptyset \Rightarrow e_{ss}(t) = \emptyset$ "perfect tracking"

and this is often a basic design requirement.

Tracking (cont)

Suppose more generally $y_d(t) = A\cos\omega t$

then $e_{ss}(t) = A|S(j\omega)|\cos(\omega t + \angle S(j\omega))$

and in particular $|e_{ss}(t)| \leq A|S(j\omega)|$

So we want $|S(j\omega)| \ll 1$ for a wide range of frequencies ω (including $\omega = \emptyset$)

$\Rightarrow$ Want Bode magnitude diagram $|S(j\omega)| \ll \emptyset\,dB$ for a large range of ω (including $\emptyset$).

$\Rightarrow$ We will show feedback loops with good tracking properties place constraints on design process, which often conflict with other requirements (stability + performance).

Bandwidth

Define ω_B to be largest ω for which

$$|S(j\omega)| \leq -3\text{dB} \quad \text{for all } \omega \in [0, \omega_B]$$

this is the (tracking) <u>bandwidth</u> of the system.

$\Rightarrow$ We want designs with <u>high bandwidth</u>.

<u>Note</u>: -3dB is an arbitrary boundary between acceptable and poor tracking. Realistic performance constraints are typically much tighter:

$$|S(j\omega)| \leq -20\text{dB} \quad (\leq 10\% \text{ worst case error})$$

or

$$|S(j\omega)| \leq -40\text{dB} \quad (\leq 1\% \text{ worst case error})$$

Example Application: Utility of $R(s)$

$\Rightarrow$ $R(s)$ lets us theoretically predict the $u(t)$ which will be generated under ideal circumstances given a specified $y_d(t)$.

$$u(t) = \mathcal{L}^{-1}\{R(s)Y_d(s)\}$$

$\Rightarrow$ Primary quantity of interest is $\max_{t \geq 0} |u(t)|$

$\Rightarrow$ Quantifies maximum control effort required.

$\Rightarrow$ Real actuators have limits $|u(t)| \leq U_{max}$

$\Rightarrow$ Must ensure our control strategy does not "saturate" the actuators, i.e. $\max_t |u(t)| \leq U_{max}$

Saturation

Saturation of actuators, i.e. $|u(t)| = u_{max}$ for some $t \geq 0$, may produce performance degradation or even <u>instability</u> even when the poles of $R(s)$ are "good."

Unfortunately, no simple design guidelines for $H(s)$ which ensure saturation does not occur.

Some degree of design iteration typically required

Advanced (graduate level) techniques do exist to incorporate actuator limits into the design process.

Closed-loop poles

Transient

=> Performance of controlled system (settling time, steady-state, overshoot, etc) depends on poles of $T(s)$

=> ($R(s)$ and $S(s)$ have same poles!!)

=> Where are these poles??

=> Determined by denominator of $T(s)$

=> ($R(s)$ and $S(s)$ have same denominator)

=> Denom of all 3 derived TF is:

$$1+L(s)$$

Charactistic Equation

Poles of $T(s)$, $R(s)$, $S(s)$ are at values of $s \in \mathbb{C}$ such that

(CE) $\boxed{1 + L(s) = 0}$ "Chaderistic equation" of feedback system

We need sol'ns of this equation to be in "good" locations of complex plane.

Will identify required properties for $L(s)$ so this is true, then work backwards to determine required properties of $H(s)$.

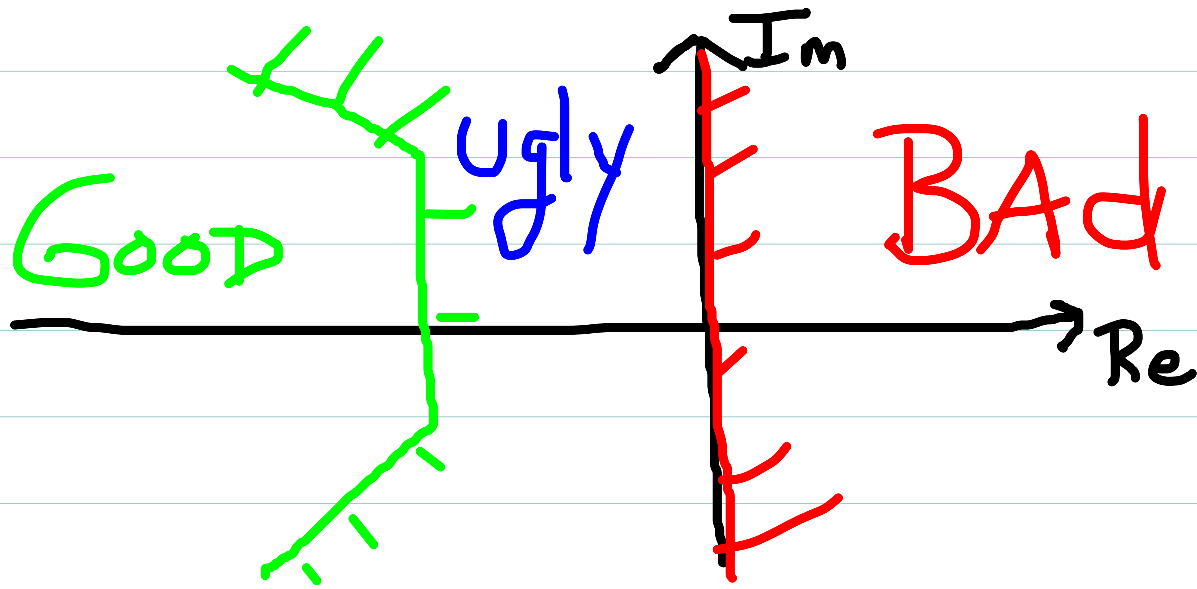
(recall: $L(s) = G(s)H(s)$).

Fundamental Consideration: Closed-loop Stability

Most basic design consideration: !!

Closed-loop poles should be "good", and certainly <u>must be stable</u>.

Thus, sol'ns of $\boxed{CE: \quad 1+L(s)=\phi}$ must be in left half of complex plane, preferably in "good region" (far from imag Axis, relatively close to or on the real Axis).



Note: $1 + L(s) = 0$ is a polynomial equation

Suppose

$$G(s) = \frac{2}{s^2} \qquad H(s) = \frac{K(s-z)}{(s-p)}$$

K, z, p design choices.

Then

$$L(s) = G(s)H(s) = \frac{2K(s-z)}{s^2(s-p)}$$

and

$$\boxed{1 + L(s) = 0 = 1 + \frac{2K(s-z)}{s^2(s-p)}}$$

SAME

Equivalently:

$$s^2(s-p) + 2K(s-z) = 0$$

or

$$\boxed{s^3 - ps^2 + 2Ks - 2Kz = 0}$$