EL2620 Nonlinear Control

Lecture 7

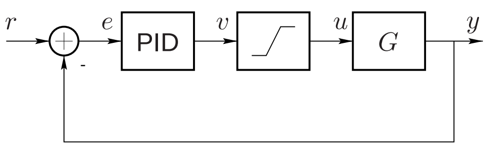
- Compensation for saturation (anti-windup)
- Friction models
- Compensation for friction

Today's Goal

You should be able to analyze and design

- Anti-windup for PID and state-space controllers
- Compensation for friction

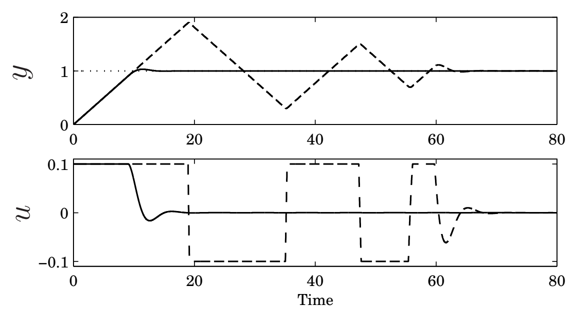
The Problem with Saturating Actuator



- The feedback path is **broken** when u saturates $\Rightarrow$ Open loop behavior!
- Leads to problem when system and/or the *controller* are unstable
  - Example: I-part in PID

Recall: $C_{\text{PID}}(s) = K\left(1 + \frac{1}{T_i s} + T_d s\right)$

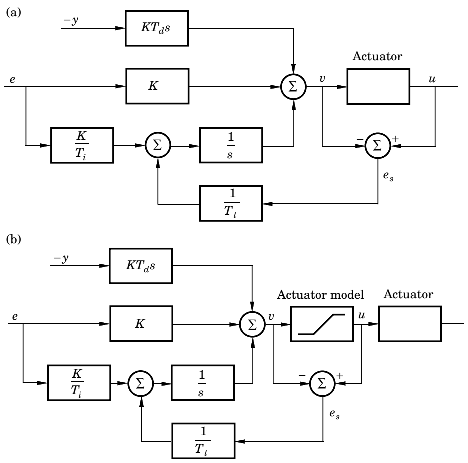
Example—Windup in PID Controller



PID controller without (dashed) and with (solid) anti-windup

Anti-Windup for PID Controller

Anti-windup (a) with actuator output available and (b) without



Anti-Windup is Based on Tracking

When the control signal saturates, the integration state in the controller *tracks* the proper state

The tracking time T_t is the design parameter of the anti-windup

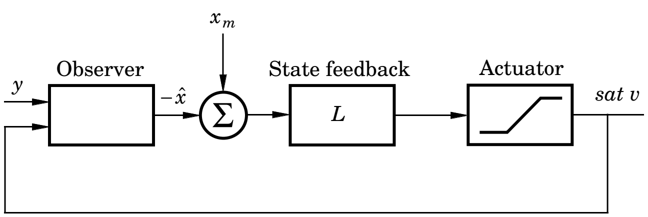
Common choices of T_t:

- $T_t = T_i$
- $T_t = \sqrt{T_i T_d}$

Remark: If $0 < T_t \ll T_i$, then the integrator state becomes sensitive to the instances when $e_s \neq 0$:

$$I(t) = \int_0^t \left[\frac{Ke(\tau)}{T_i} + \frac{e_s(\tau)}{T_t} \right] d\tau \approx \frac{1}{T_t} \int_0^t e_s(\tau)\, d\tau$$

Anti-Windup for Observer-Based State Feedback Controller



$$\dot{\hat{x}} = A\hat{x} + B\,\mathrm{sat}\,v + K(y - C\hat{x})$$
$$v = L(x_m - \hat{x})$$

$\hat{x}$ is estimate of process state, x_m desired (model) state

Need actuator model if $\mathrm{sat}\,v$ is not measurable

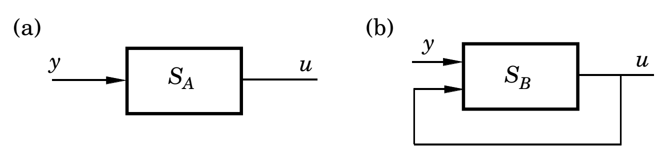
Anti-Windup for General State-Space Controller

State-space controller:

$$\dot{x}_c = Fx_c + Gy$$
$$u = Cx_c + Dy$$

Windup possible if F unstable and u saturates

Idea: Rewrite representation of control law from (a) to (b) with the same input–output relation, but where the unstable S_A is replaced by a stable S_B. If u saturates, then (b) behaves better than (a).



Mimic the observer-based controller:

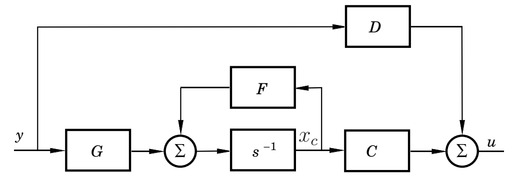
$$\begin{aligned}\dot{x}_c &= Fx_c + Gy + K(u - Cx_c - Dy)\\ &= (F - KC)x_c + (G - KD)y + Ku\end{aligned}$$

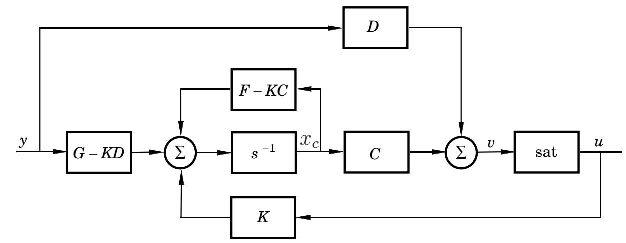
Choose K such that $F_0 = F - KC$ has desired (stable) eigenvalues. Then use controller

$$\begin{aligned}\dot{x}_c &= F_0 x_c + G_0 y + Ku\\ u &= \mathrm{sat}(Cx_c + Dy)\end{aligned}$$

where $G_0 = G - KD$.

State-space controller without and with anti-windup:





Controllers with ”Stable” Zeros

Most controllers are minimum phase, i.e. have zeros strictly in LHP

$$\begin{aligned}\dot{x}_c &= Fx_c + Gy \\ u &= Cx_c + Dy\end{aligned} \quad \Rightarrow_{u=0} \quad \begin{aligned}\dot{x}_c &= \overbrace{(F - GC/D)}^{\texttt{zero dynamics}} x_c \\ y &= -C/Dx_c\end{aligned}$$

Thus, choose ”observer” gain

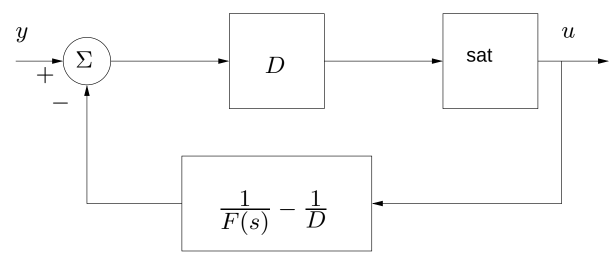
$$K = G/D \quad \Rightarrow \quad F - KC = F - GC/D$$

and the eigenvalues of the ”observer” based controller becomes equal to the zeros of $F(s)$ when u saturates

Note that this implies $G - KD = 0$ in the figure on the previous slide, and we thus obtain P-feedback with gain D under saturation.

Controller $F(s)$ with ”Stable” Zeros

Let $D = \lim_{s\to\infty} F(s)$ and consider the feedback implementation

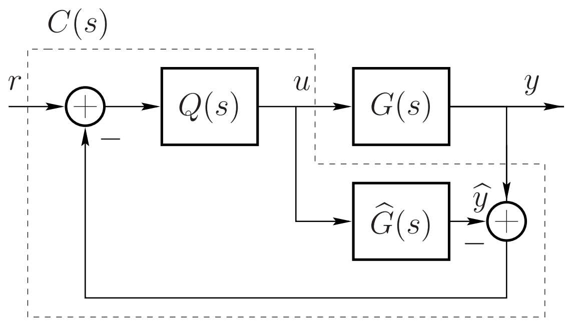


It is easy to show that transfer function from y to u with no saturation equals $F(s)$!

If the transfer function $(1/F(s) - 1/D)$ in the feedback loop is stable (stable zeros) $\Rightarrow$ No stability problems in case of saturation

Internal Model Control (IMC)

IMC: apply feedback only when system G and model $\hat{G}$ differ!



Assume G stable. Note: feedback from the model error $y - \widehat{y}$.

Design: assume $\widehat{G} \approx G$ and choose Q stable with $Q \approx G^{-1}$.

Example

$$\widehat{G}(s) = \frac{1}{T_1 s + 1}$$

Choose

$$Q = \frac{T_1 s + 1}{\tau s + 1}, \quad \tau < T_1$$

Gives the controller

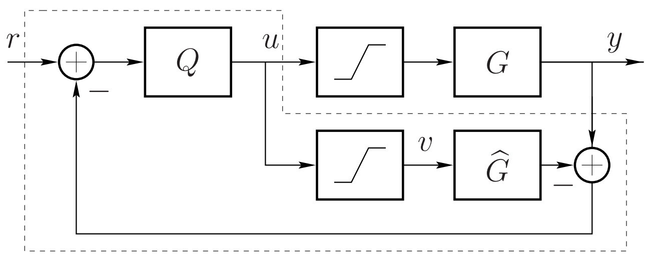
$$F = \frac{Q}{1 - Q\widehat{G}} \quad \Rightarrow$$

$$F = \frac{T_1 s + 1}{\tau s} = \frac{T_1}{\tau}\left(1 + \frac{1}{T_1 s}\right)$$

PI-controller!

IMC with Static Nonlinearity

Include nonlinearity in model



Choose $Q \approx G^{-1}$.

Example (cont'd)

Assume $r = 0$ and abuse of Laplace transform notation

$$u = -Q(y - \widehat{G}v) = -\frac{T_1 s + 1}{\tau s + 1} y + \frac{1}{\tau s + 1} v$$

if $|u| < u_{\text{max}}$ ($v = u$): PI controller $u = \frac{-(T_1 s + 1)}{\tau s} y$

If $|u| > u_{\text{max}}$ ($v = \pm u_{\text{max}}$):

$$u = -\frac{T_1 s + 1}{\tau s + 1} y \pm \frac{u_{\text{max}}}{\tau s + 1}$$

No integration.

An alternative way to implement anti-windup!

Other Anti-Windup Solutions

Solutions above are all based on tracking.

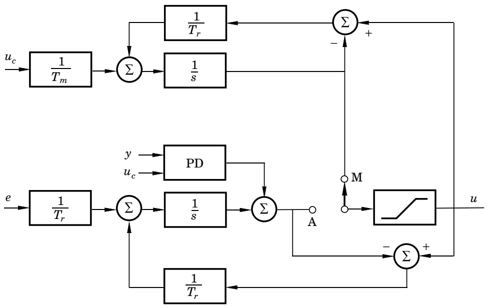
Other solutions include:

- Tune controller to avoid saturation
- Don't update controller states at saturation
- Conditionally reset integration state to zero
- Apply optimal control theory (Lecture 12)

Bumpless Transfer

Another application of the tracking idea is in the switching between automatic and manual control modes.

PID with anti-windup and bumpless transfer:



Note the incremental form of the manual control mode ($\dot{u} \approx u_c/T_m$)

Friction

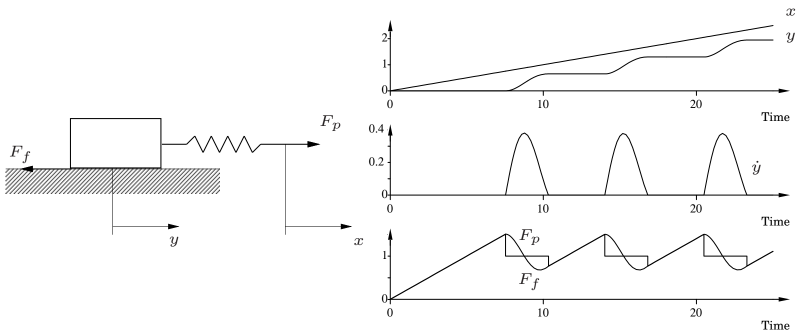
Friction is present almost everywhere

- Often bad:
  - Friction in valves and other actuators
- Sometimes good:
  - Friction in brakes
- Sometimes too small:
  - Earthquakes

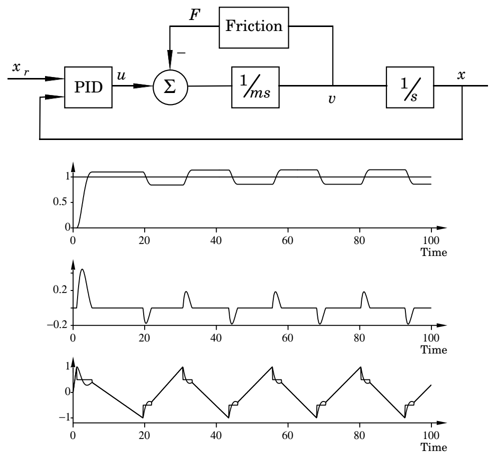
Problems:

- How to model friction?
- How to compensate for friction?
- How to detect friction in control loops?

Stick-Slip Motion

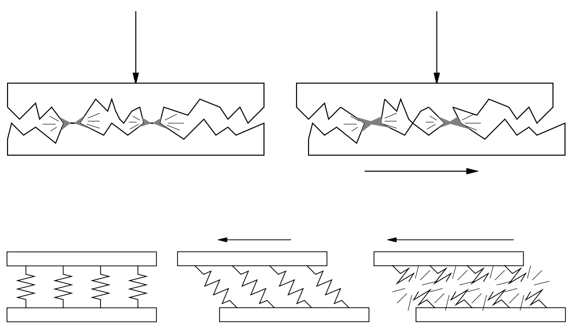


Position Control of Servo with Friction



5 minute exercise: *Which are the signals in the previous plots?*

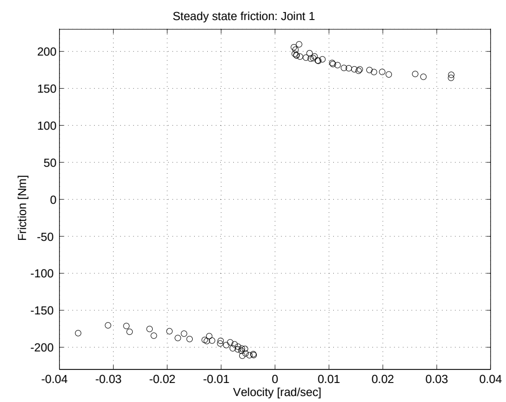
Friction Modeling



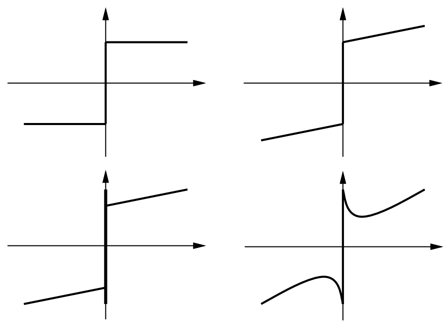
Stribeck Effect

Friction increases with decreasing velocity (for low velocities)

Stribeck (1902)



Classical Friction Models



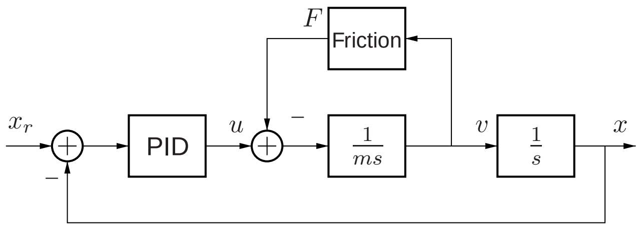
Advanced models capture various friction phenomena better

Friction Compensation

- Lubrication
- Integral action
- Dither signal
- Model-based friction compensation
- Adaptive friction compensation
- The Knocker

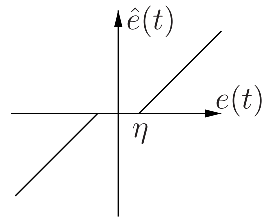
Integral Action

- Integral action compensates for any external disturbance
- Works if friction force changes slowly ($v(t) \approx$ const)
- If friction force changes quickly, then large integral action (small T_i) necessary. May lead to stability problem



Modified Integral Action

Modify the integral part to $I = \frac{K}{T_i} \int^t \hat{e}(\tau) d\tau$ where

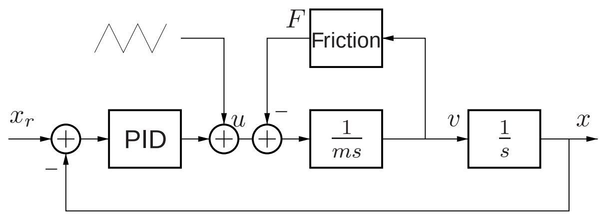


Advantage: Avoid that small static error introduces oscillation

Disadvantage: Error won't go to zero

Dither Signal

Avoid sticking at $v = 0$ (where there is high friction)
by adding high-frequency mechanical vibration (*dither*)



Cf., mechanical maze puzzle
(*labyrintspel*)

Model-Based Friction Compensation

For process with friction F:

$$m\ddot{x} = u - F$$

use control signal

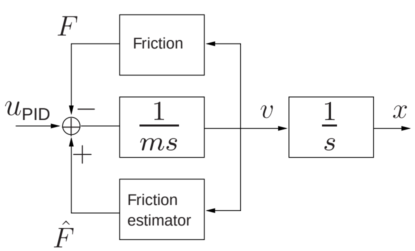
$$u = u_{\text{PID}} + \hat{F}$$

where u_{PID} is the regular control signal and $\hat{F}$ an estimate of F.

Possible if:

- An estimate $\hat{F} \approx F$ is available
- u and F does apply at the same point

Adaptive Friction Compensation



Coulomb friction model: $F = a \operatorname{sgn} v$

Friction estimator:

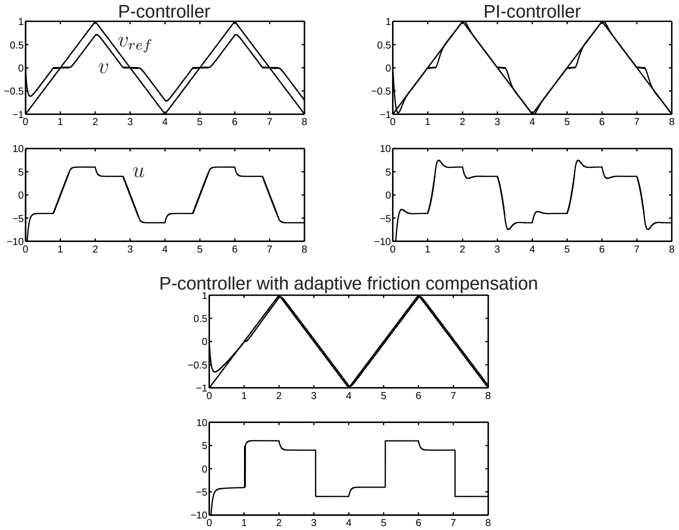
$$\dot{z} = k u_{\text{PID}} \operatorname{sgn} v$$
$$\hat{a} = z - km|v|$$
$$\hat{F} = \hat{a} \operatorname{sgn} v$$

Adaptation converges: $e = a - \hat{a} \to 0$ as $t \to \infty$

Proof:

$$\begin{aligned}\frac{de}{dt} &= -\frac{d\hat{a}}{dt} = -\frac{dz}{dt} + km\frac{d}{dt}|v| \\ &= -k u_{\text{PID}} \operatorname{sgn} v + km\dot{v} \operatorname{sgn} v \\ &= -k \operatorname{sgn} v (u_{\text{PID}} - m\dot{v}) \\ &= -k \operatorname{sgn} v (F - \hat{F}) \\ &= -k(a - \hat{a}) \\ &= -ke\end{aligned}$$

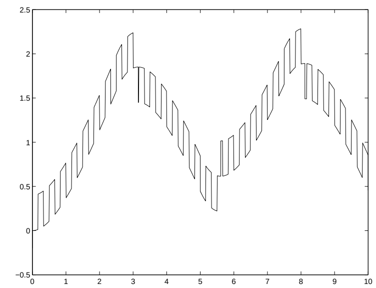
Remark: Careful with $\frac{d}{dt}|v|$ at $v = 0$.



The Knocker

Coulomb friction compensation and square wave dither

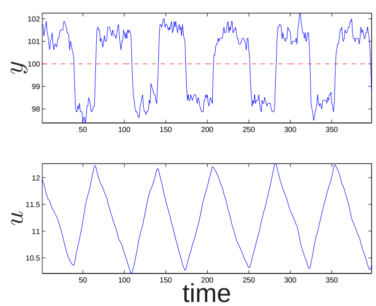
Typical control signal u



Hägglund: Patent and Innovation Cup winner

Detection of Friction in Control Loops

- Friction is due to wear and increases with time
- **Q:** When should valves be maintained?
- **Idea:** Monitor loops automatically and estimate friction



Horch: PhD thesis (2000) and patent

Today's Goal

You should be able to analyze and design

- Anti-windup for PID, state-space, and polynomial controllers
- Compensation for friction

Next Lecture

- Backlash
- Quantization