2E1252
Control Theory and Practice

Lecture 11:
Actuator saturation and anti wind-up

Mikael Johansson
School of Electrical Engineering
KTH, Stockholm, Sweden

Mikael Johansson mikaelj@ee.kth.se

Learning aims

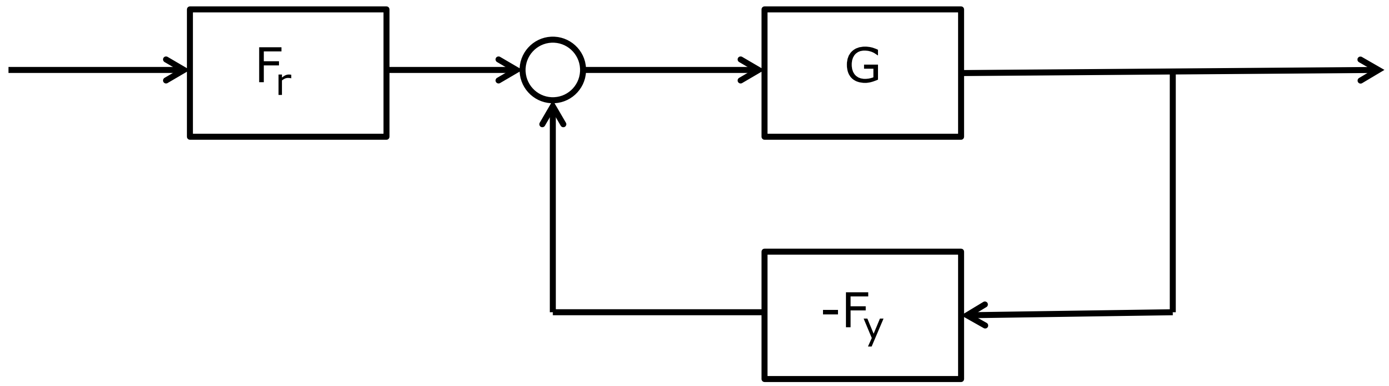
After this lecture, you should

- understand how saturation can cause controller states to “wind up”
- know how to modify a linear observer to account for saturation
- be able to interpret the observer modification in a block diagram
- be able to analyze closed-loop stability using the small-gain theorem
- know how to tune the anti-windup gain for a PID controller

Mikael Johansson mikaelj@ee.kth.se

So far…

…we have designed **linear controllers** for **linear systems**

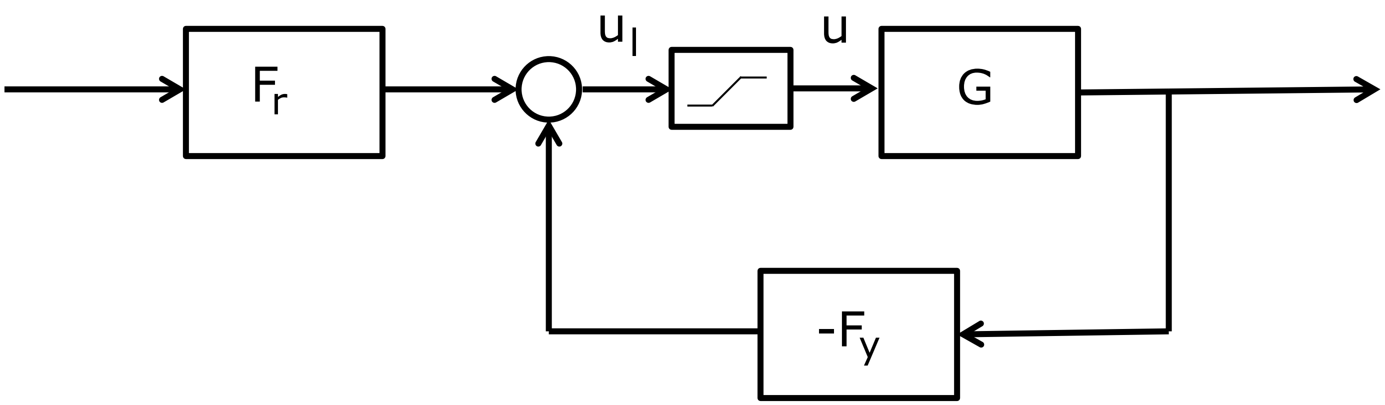


(uncertainty models allow us to account for some nonlinearities)

Mikael Johansson mikaelj@ee.kth.se

Actuation is typically limited

In practically all control systems, the actuation is limited



The new block represents a **saturation**

$$u = \text{sat}(u_l) = \begin{cases} u_{\min} & \text{if } u_l \leq u_{\min} \\ u_{\max} & \text{if } u_l \geq u_{\max} \\ u_l & \text{otherwise} \end{cases}$$

Mikael Johansson mikaelj@ee.kth.se

The impact of limited actuation

Consider the simple servo model:

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

A controller is

$$F_y(s) = \frac{439s^2 + 710.5s + 316.2}{s^3 + 26.47s^2 + 349.8s - 7.13}$$

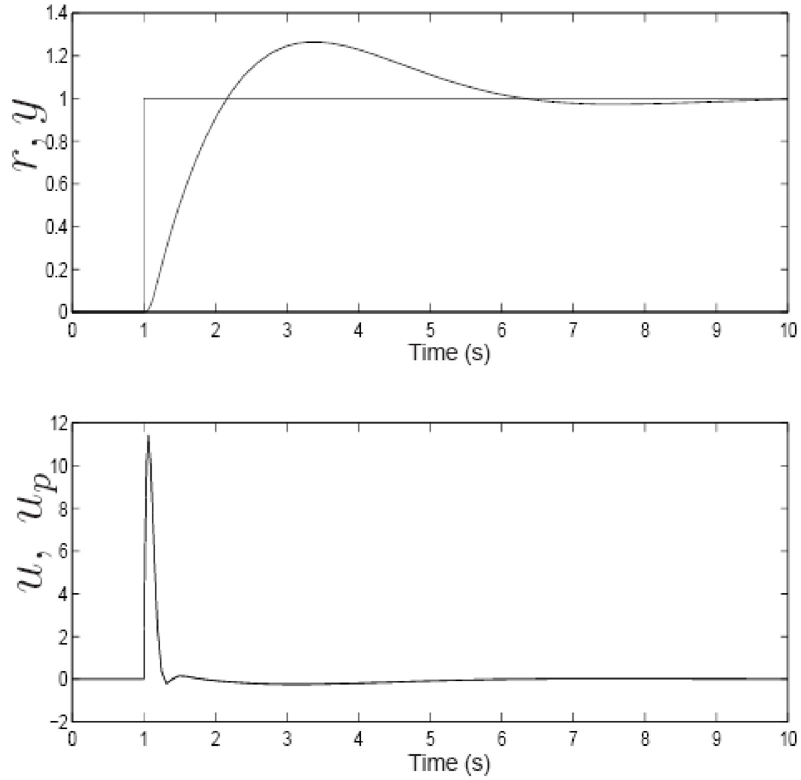
which has poles in -13.2444$\pm$ 13.2255i, and 0.0204

Note: controller is unstable.

Mikael Johansson mikaelj@ee.kth.se

Step response of linear system

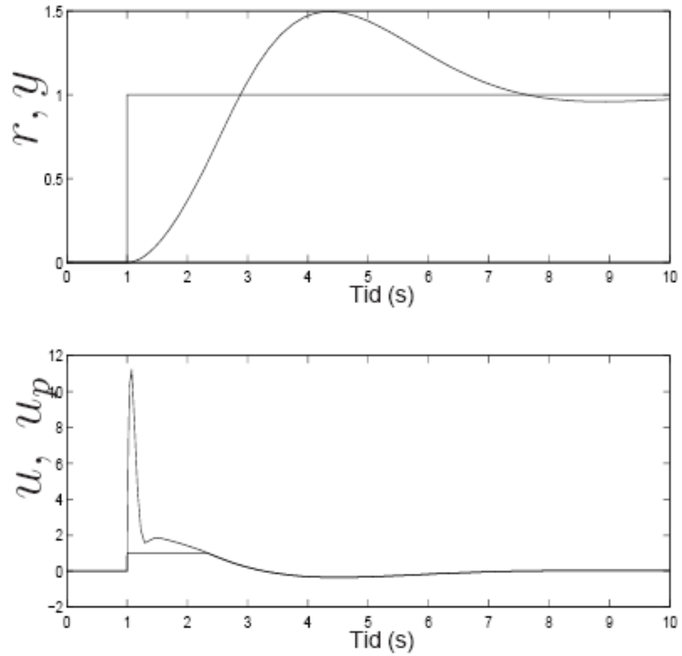
A unit step response for the linear closed-loop (assuming no saturation)



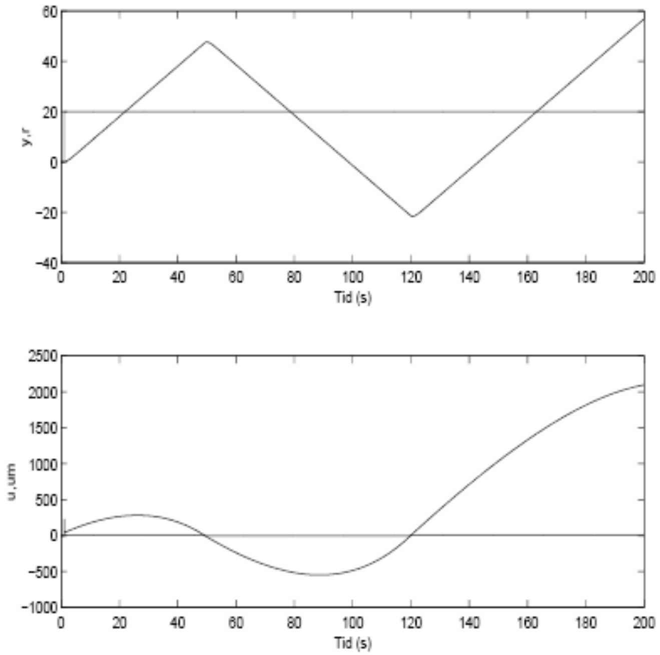
Mikael Johansson mikaelj@ee.kth.se

Step response with saturated control

Step responses with actuator saturation for small and large reference change



Slower, larger overshoot

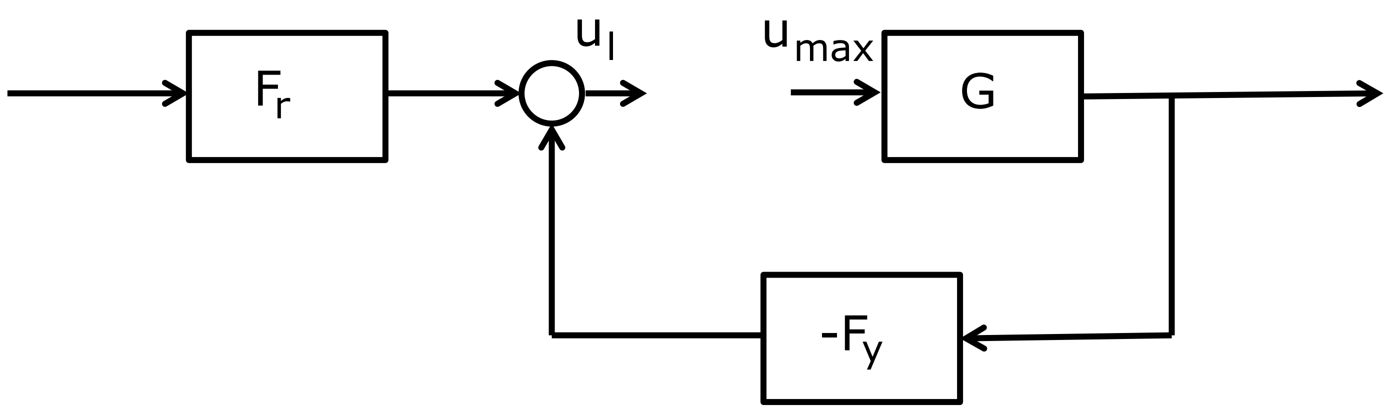


unstable

Mikael Johansson mikaelj@ee.kth.se

What is wrong?

When actuator is saturated, the system operates in open loop
(changes in u_l do not affect G or controller states while saturated)



Constant input make plant and controller states to grow large (“wind up”)

– particularly critical when plant or controller is unstable or integrating

Mikael Johansson mikaelj@ee.kth.se

Understanding what is wrong

Consider observer-based controllers (e.g. LQG, H-infinity, H-2, etc.)

$$\frac{d}{dt}\hat{x}(t) = A\hat{x}(t) + Bu_l(t) + K(y(t) - \hat{y}(t))$$
$$u_l(t) = -L\hat{x}(t)$$

Controller transfer function:

$$U_l(s) = -F_y(s)Y(s) = -L(sI - A + BL + KC)^{-1}K\,Y(s)$$

Basic idea of observer: "simulate system" and correct when $\hat{y}(t) \neq y(t)$

- simulation part does not reflect reality when input is saturated!

An improvement

Make sure that observer reflects actual system dynamics

$$\begin{aligned}
\frac{d}{dt}\hat{x}(t) &= A\hat{x}(t) + Bu(t) + K(y(t) - \hat{y}(t)) \\
u(t) &= \mathrm{sat}(u_l(t)) \\
u_l(t) &= -L\hat{x}(t)
\end{aligned}$$

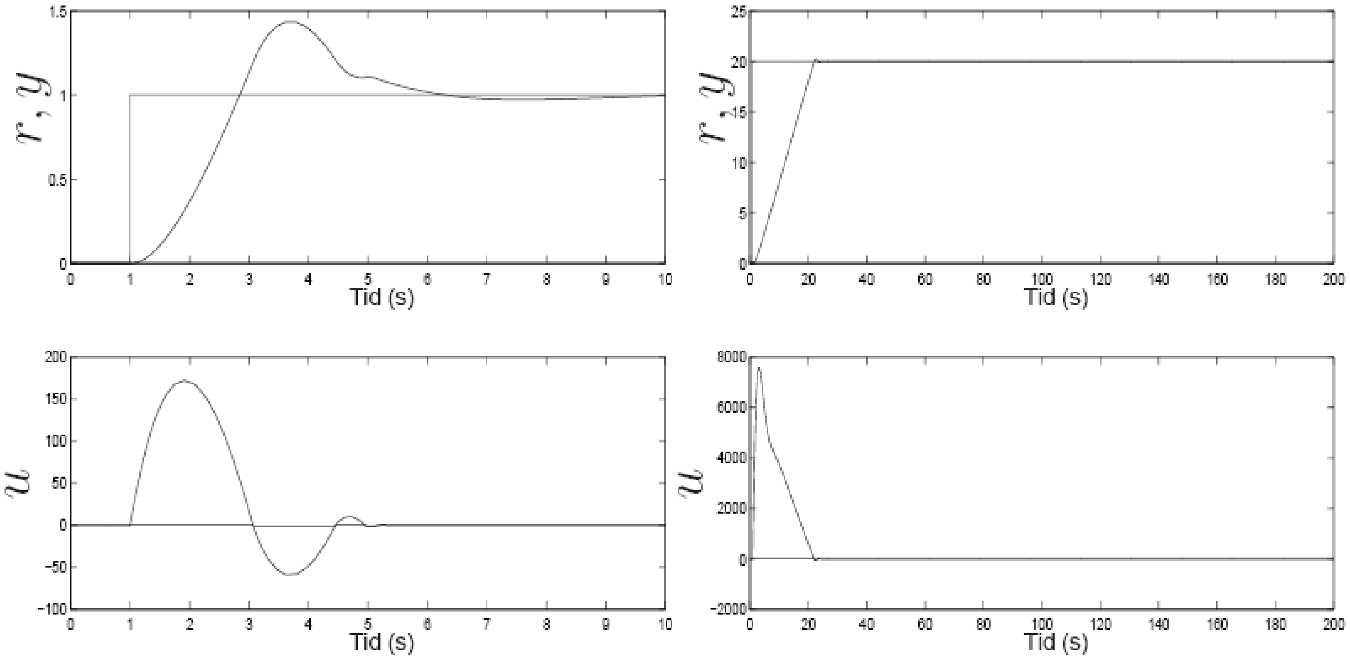
Modification known as **observer-based anti-windup**

- avoids that controller states wind up
- often enough to get reasonable performance

Mikael Johansson mikaelj@ee.kth.se

Step responses with modified observer

Step responses with actuator saturation for small and large reference change



Smaller overshoot, no longer unstable for large reference changes.

Analysis: no wind-up in saturation

When in saturation, $u(t) = u_{\text{lim}}$ is constant and controller dynamics is

$$\frac{d}{dt}\hat{x}(t) = A\hat{x}(t) + Bu(t) + K(y(t) - \hat{y}(t)) =$$
$$= (A - KC)\hat{x}(t) + Bu_{\text{lim}} + Ky(t)$$

Stability properties given by A-KC, which is typically stable

Taking Laplace transforms, we find

$$U_l(s) = -L(sI - A + KC)^{-1}KY(s) - K(sI - A + KC)^{-1}B\frac{u_{\text{lim}}}{s}$$

Mikael Johansson mikaelj@ee.kth.se

Interpretation: feedback from u-u_l

To relate modified controller to original, re-write observer as

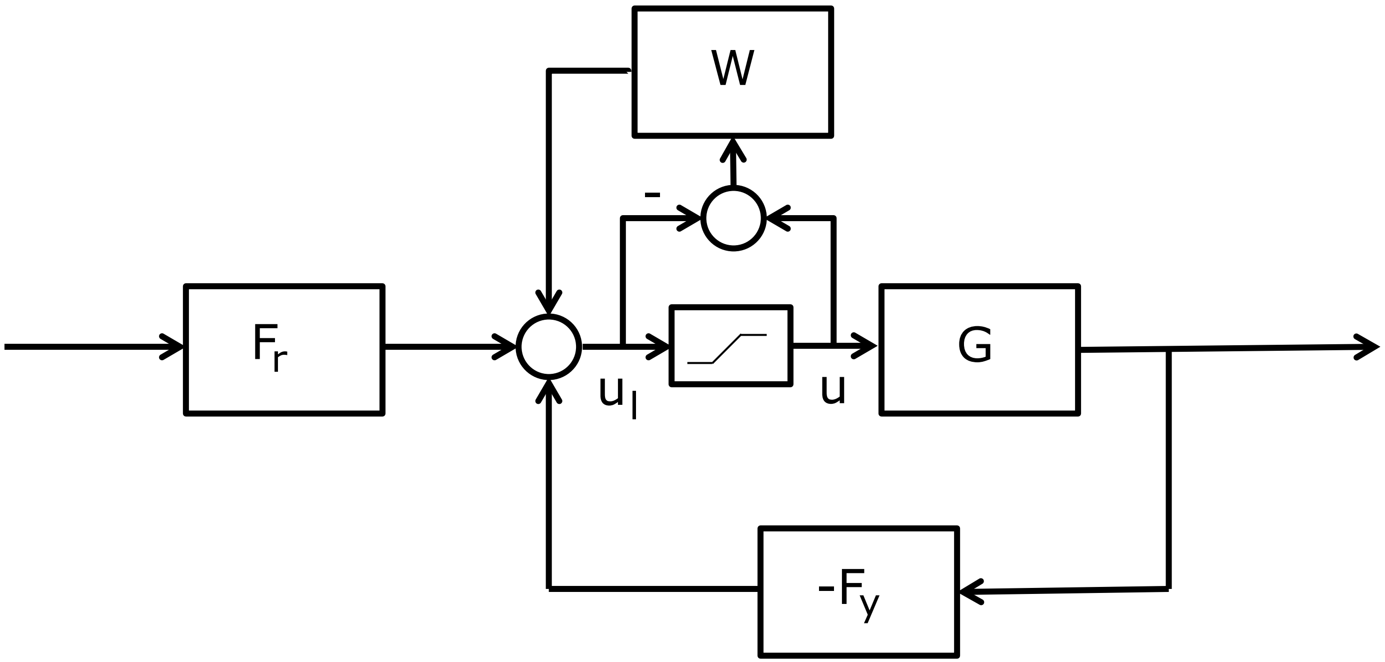
$$\begin{aligned}\frac{d}{dt}\hat{x}(t) &= A\hat{x}(t) + Bu(t) + K(y(t) - \hat{y}(t)) = \\ &= (A - KC)\hat{x}(t) + B(u_l(t) + u(t) - u_l(t)) + K(y(t) - \hat{y}(t)) = \\ &= (A - BL - KC)\hat{x}(t) + Ky(t) + B(u(t) - u_l(t))\end{aligned}$$

Taking Laplace transforms, we find

$$\begin{aligned}U_l(s) &= -L(sI - A + BL + KC)^{-1}KY(s) \\ &\quad - L(sI - A + BL + KC)^{-1}B(U(s) - U_l(s)) = \\ &= -F_y(s)Y(s) + W(s)(U(s) - U_l(s))\end{aligned}$$

The linear control law plus compensation from $u(t) - u_l(t)$

Interpretation in block diagram



Basis for many heuristic techniques for anti-windup (more later…)

Mikael Johansson mikaelj@ee.kth.se

What about stability?

We have shown that

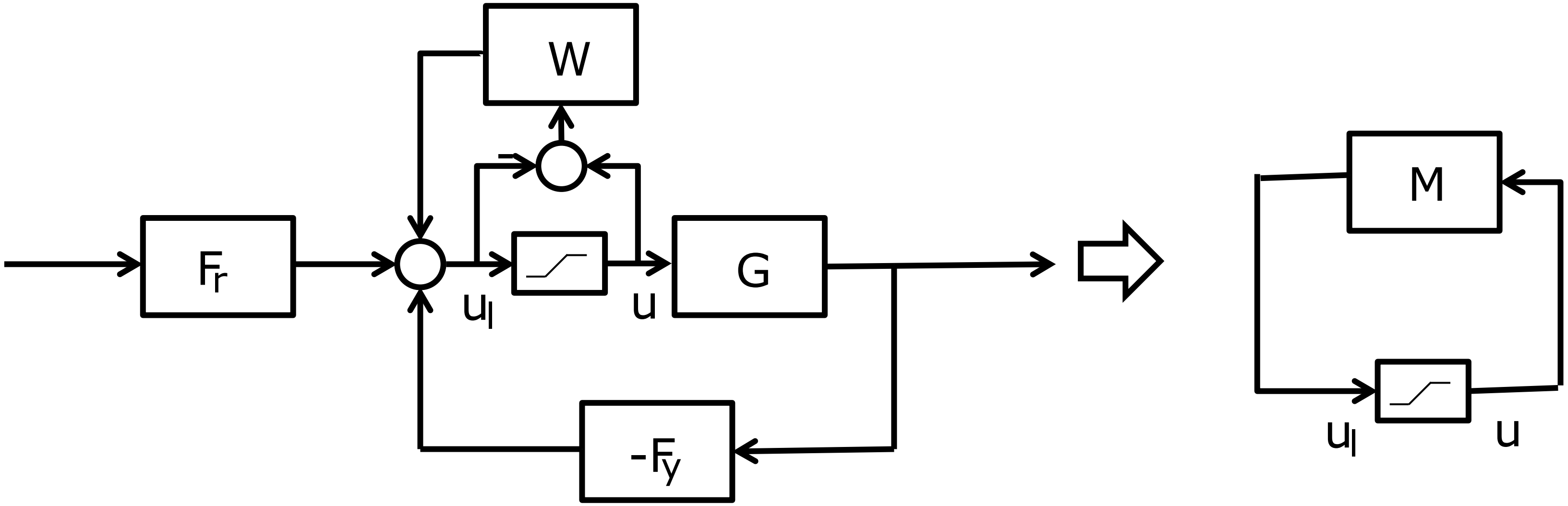
- in absence of saturation, the closed-loop system is stable
- when input remains in saturation, the controller is stable

but **no** stability guarantees when control moves in and out of saturation!

Global stability can sometimes be ensured using small-gain theorem

Mikael Johansson mikaelj@ee.kth.se

A small-gain analysis



To find the linear system M, note that

$$u_l = W(u - u_l) - GF_y u \Rightarrow (I + W)u_l = (W - GF_y)u := Mu$$

Since gain of saturation nonlinearity is one (cf. Lecture 1), so if

$$\|M\|_\infty < 1$$

closed-loop stability is ensured

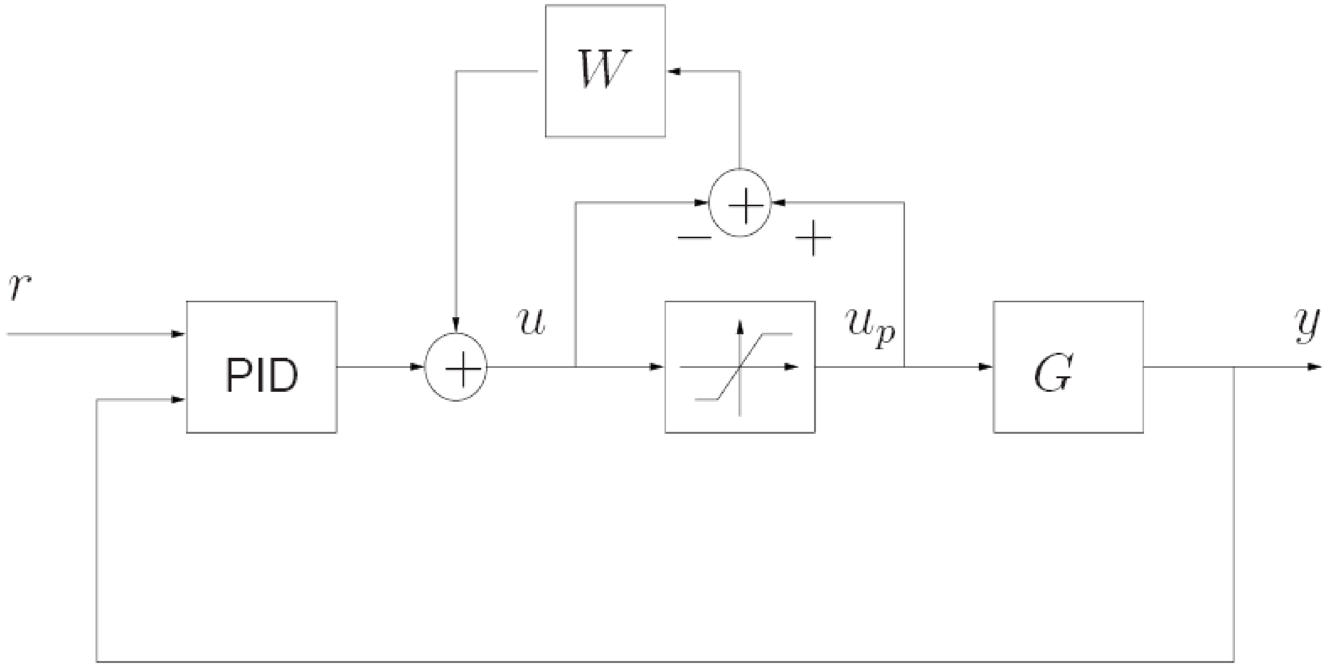
Mikael Johansson mikaelj@ee.kth.se

Design guidelines

1. Design observer-based controller using technique of choice
2. Modify observer to reflect the presence of saturation nonlinearity
3. Attempt to establish stability using small gain theorem, simulate
4. If unsatisfactory, re-design controller with higher penalty on control

Mikael Johansson mikaelj@ee.kth.se

Application to PID control



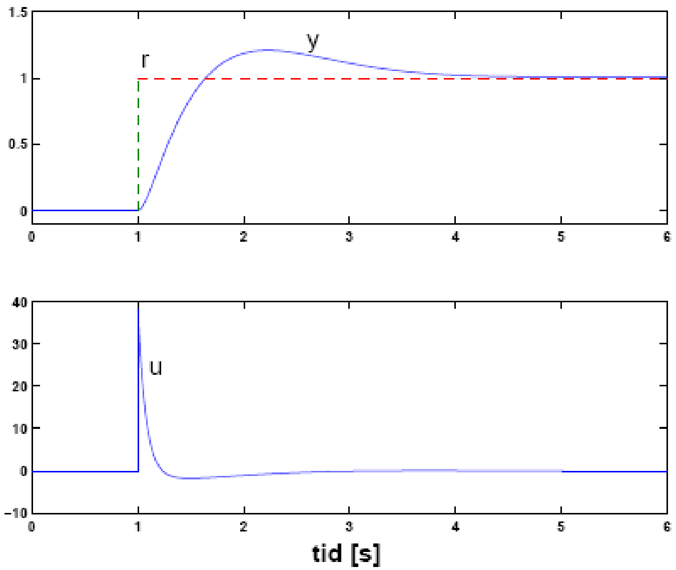
Common choice: $W(s) = \dfrac{1}{sT_t}$ (tracking anti-reset windup)

Rule-of-thumb: $T_t = \sqrt{T_i T_d}$ (T_i integral time, T_d derivative time of PID)

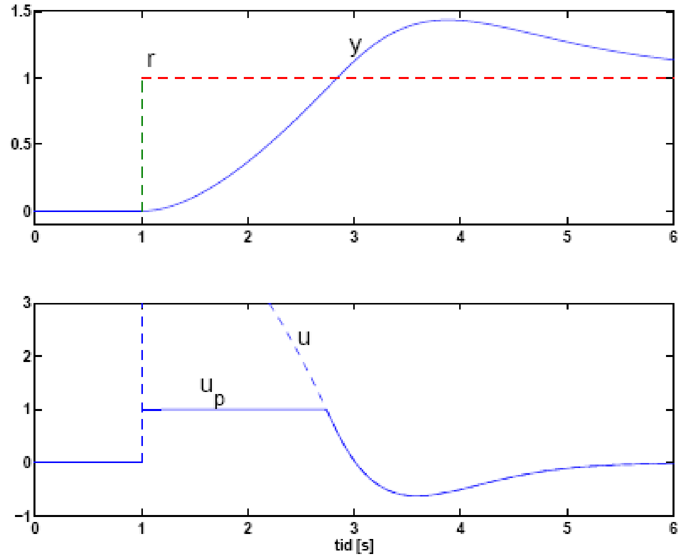
Mikael Johansson mikaelj@ee.kth.se

DC Servo under PID control

No constraint

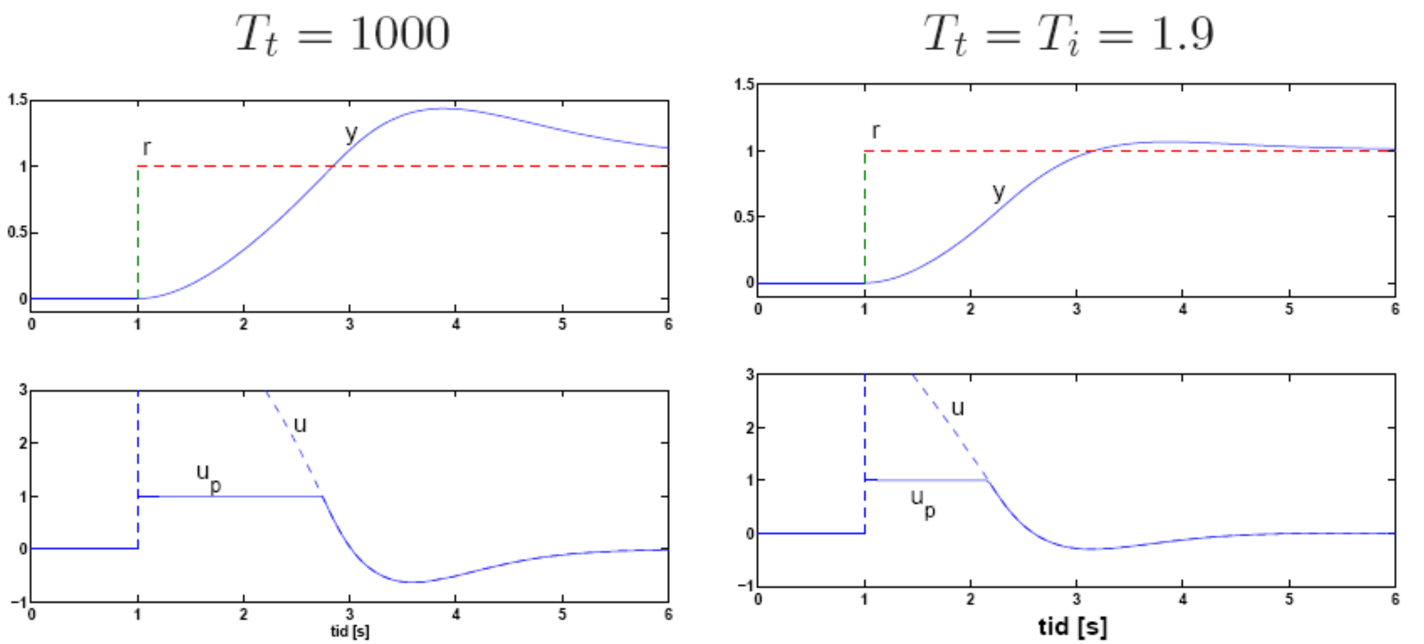


$u_p \in [-1, 1]$, no compensation



Mikael Johansson mikaelj@ee.kth.se

Servo: PID+anti-windup



Summary

The problem with limited actuation:

- New phenomena, not predicted by linear control theory
  - controller, plant states “wind up” (grow large), feedback loop open
- Observer-based anti-windup
  - make estimator reflect actual process dynamics
  - ensures that controller states are stable in saturation
  - interpretation as feedback from “saturation error”
  - global stability analysis via small-gain theorem
- Anti-windup for PID control
  - same structure, heuristic compensator, rules-of-thumbs