Assignment 3

TTK4210 - Advanced Control of Industrial Processes

Sondre Myrberg

February 15, 2018

1. Anti-windup (and simulation) for the LV-model

Using the provided solution from assignment 2.

a. Implementation

In the PI-controller we get the transfer functions

$$k_1 = \frac{s + \frac{1}{2}}{2s} = \frac{1}{2}\frac{2s+1}{2s}$$
$$k_2 = \frac{-s - \frac{1}{2}}{2s} = -\frac{1}{2}\frac{2s+1}{2s}$$

which yields the parameters

j	1	2
K_{p_j}	$\frac{1}{2}$	$-\frac{1}{2}$
T_{i_j}	2	2

Using the anti-windup scheme for PI-controllers given in 6.3.1 in the course notes we can implement anti-windup as shown in Figure 1 and Figure 2. I don't have time and energy for the decoupled controller now.

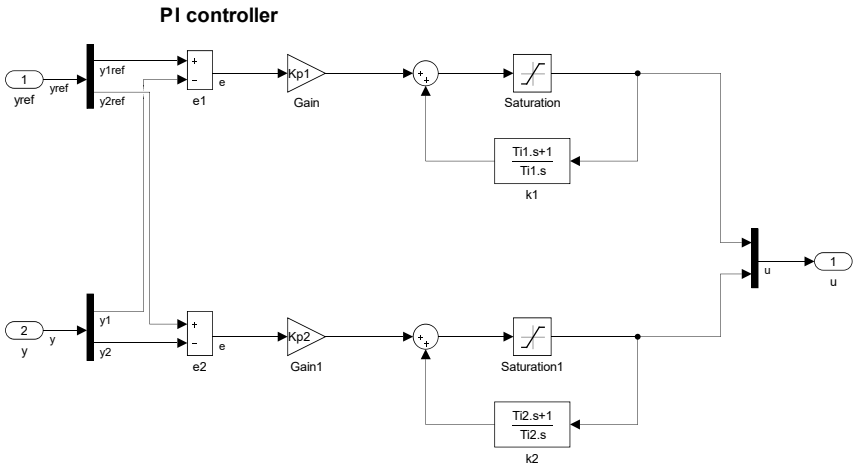


Figure 1: Anti-windup for PI controller

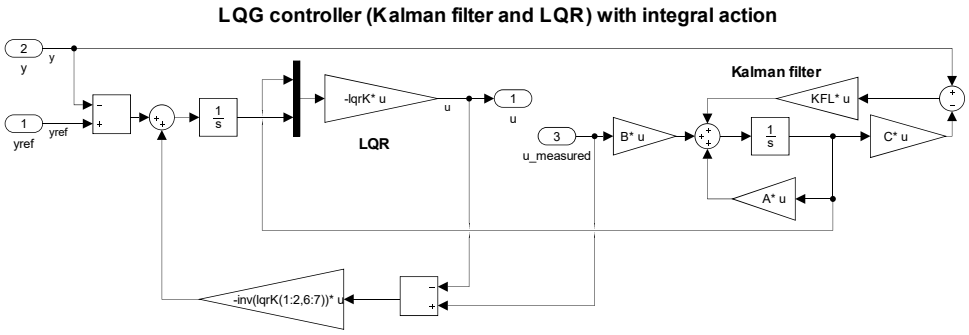
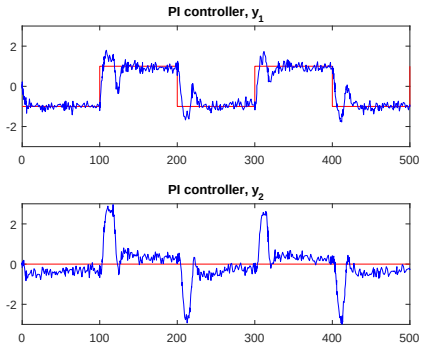


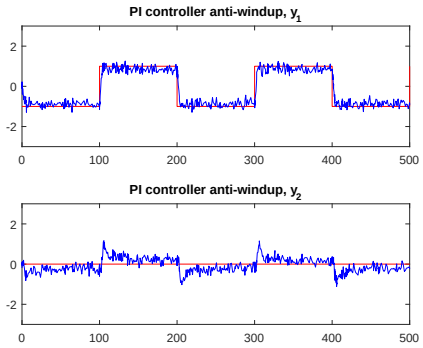
Figure 2: Anti-windup for LGQ-controller with integral action

b. Simulation with anti-windup

When simulating with and without anti-windup we clearly see differences in performance. The controllers with anti-windup implemented have smaller deviations from the setpoints than the controllers without anti-windup. For the PI-controller we see this in Figure 3 and for the LQG-controller this is shown in Figure 4. For both cases we see that the performance has increased significantly.

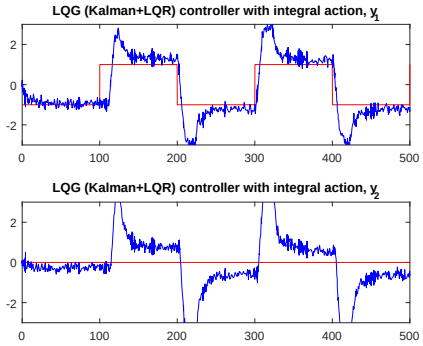


(a) Simulation of the PI-controller without anti-windup

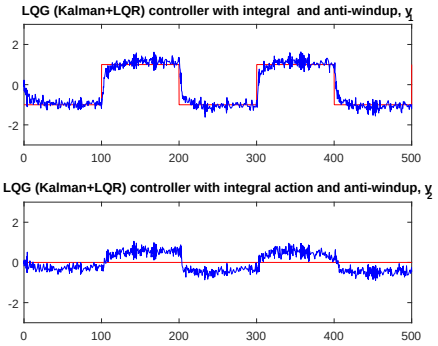


(b) Simulation of the PI-controller with anti-windup

Figure 3: Comparison of the response of the PI-controller with and without anti-windup



(a) Simulation of the LQG-controller without anti-windup



(b) Simulation of the LQG-controller with anti-windup

Figure 4: Comparison of the response of the LQG-controller with and without anti-windup

2. Anti-windup with PI controllers and selectors

a. Model implementation

The plant can be modelled as shown in Figure 5 with $y(s) = G(s)u(s) + G_d(s)d(s)$.

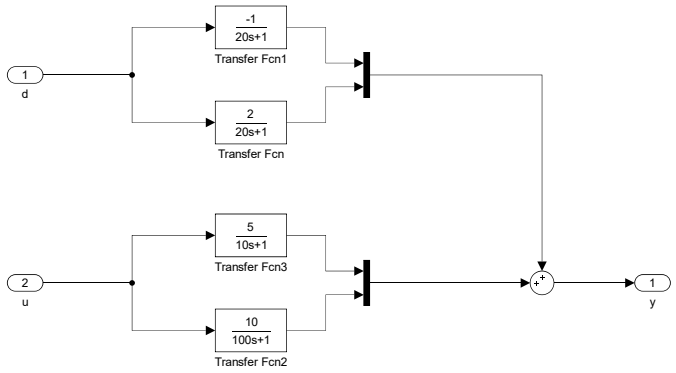


Figure 5: Implementation of the system given the transfer functions from input and disturbance to output

b. Controller tuning

Using Ziegler-Nichols method of controlling a PI(D)-controller on each of the two control loops individually we get the following table of parameters

Controller	K_p	T_i
Flow	3.04	4.63
Pressure	18	3.45

This was obtained by using a unit step function as disturbance and observing the corresponding outputs, as per Ziegler-Nichols method.

c. Simulation without anti-windup

The PI-controllers are implemented on the form shown in Figure 6. Here we can manually choose between the regular PI-controller and the anti-windup PI-controller. The

complete system is then shown in Figure 7. Here the selector chooses the output with lowest value. Simulating this yields the plot shown in Figure 8. Here we see that only the pressure is controlled as the flow controller winds up and never gets low enough value to get chosen.

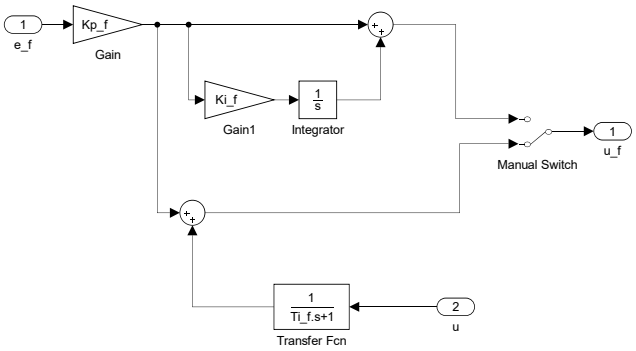


Figure 6: Diagram of the flow controller both with and without anti-windup. The controller for pressure have the same exact form, but different values for K and T.

d. Simulation with anti-windup

Simulating the same references and disturbances as abov, but choosing the anti-windup controller insted we get the response shown in Figure 9. Here we see that the selector choses the controller with the lowest output, thus both flow and pressure gets controlled.

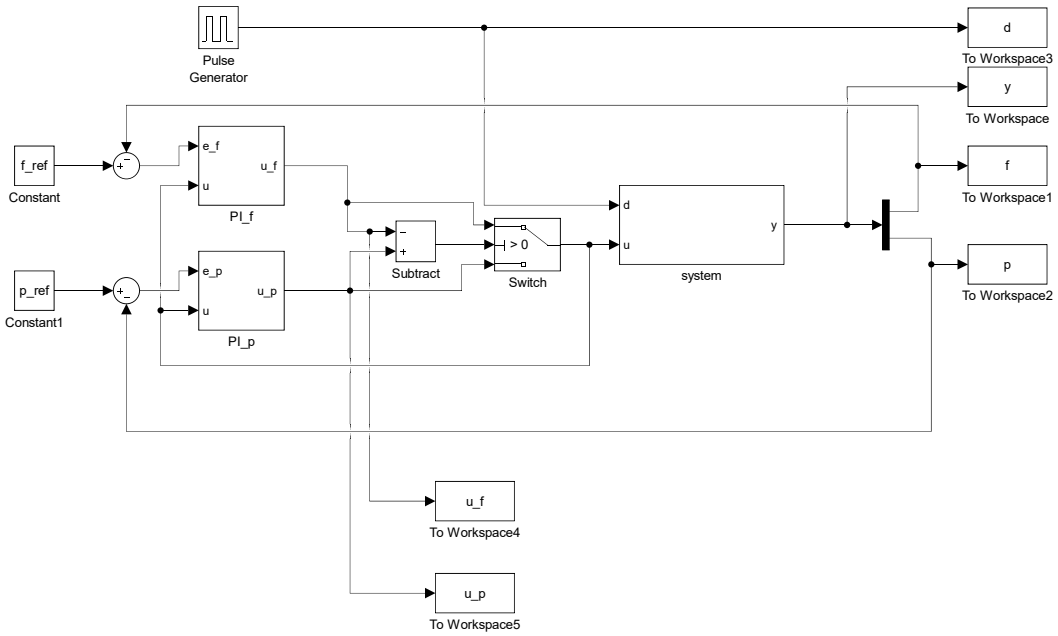


Figure 7: Simulink diagram of the complete system with selector of lowest controller output implemented.

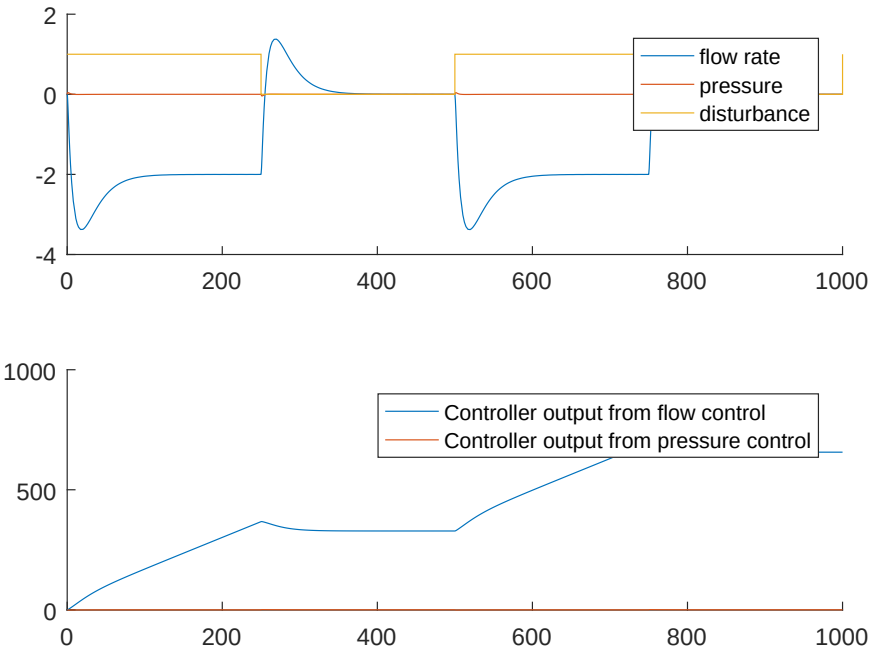


Figure 8: System and controller output without anti-windup

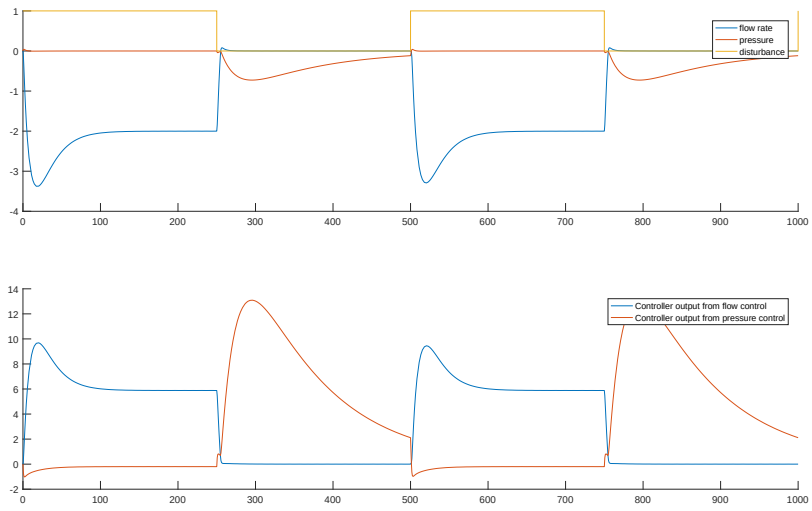


Figure 9: System and controller output with anti-windup