TTK4210 Advanced Control of Industrial Processes
Department of Engineering Cybernetics
Norwegian University of Science and Technology
Spring 2018 - Assignment 3
Due date: Wednesday 14 February at 16:00.

1. **Anti-windup (and simulation) for the LV model**
   Consider the LV model and controllers from assignment 2. If you did not finish assignment 2, you may use the solution (which will be published Wednesday 7 February). Use the same reference signals and saturation limits. If you do not see results of the saturation, you might increase the disturbances or decrease the saturation limits.

   (a) **Implementation**
   Implement anti-windup for all three controllers.

   (b) **Simulation with anti-windup**
   Simulate with the large disturbances given in assignment 2. Plot the results and compare to the performance without anti-windup (make comparable plots with and without anti-windup implemented). Comment on the results.

2. **Anti-windup with PI controllers and selectors**
   Selectors and 'overrides' are often used in plants when operational objectives change with operating conditions. Clearly, selectors makes the control system non-linear, and mistakes are easily made when designing such control systems, leading to sustained oscillations or even instability. It may reasonably be argued that more advanced control techniques, specifically MPC, can be a good alternative to regulatory control systems in which standard single-loop controllers (i.e., PI) are combined with selectors, since changes in the set of active constraints are handled relatively easily with MPC.

   This assignment will address a particular problem of windup when using selectors. This windup is not caused by constraints in the manipulated variable, but occurs for inactive ('deselected') controllers, due to the absence of feedback when the controller is not selected. In this assignment a simple remedy to this problem is tested.

   Introduction to selective control

   Selective control is sometimes used when there are more than one candidate controlled variable for a manipulated variable. For each of the candidate controlled variables a separate controller is then used, and the value of the manipulated variable that is implemented is selected among the controller outputs. A simple example of selective control with pressure control on one side and flow control on the other side of a valve is shown in Fig. 1. Normally one selects simply the highest or lowest value. A few points should be made about this control structure:

   - Clearly, a single manipulated variable can control only one controlled variable at the time, i.e., the only variable that is controlled at any instant is the variable for which the corresponding controller output is implemented.

It might appear strange to point out such a triviality, but discussions with several otherwise sensible engineers show that many have difficulty comprehending this. Thus, one should consider with some care how such a control structure will work.

- The selection of the active controller is usually based on the controller outputs, not the controller inputs. Nevertheless the local operators and engineers often believe that the selection is based on the controller inputs, or that ”the control switches when a measurement passes its setpoint”. In principle, the selection of the active controller may also be based on the controller inputs[1]. Some type of scaling will then often be necessary, in order to compare different types of physical quantities (e.g., comparing flowrates and pressures).
- If the controllers contain integral action, a severe problem that is similar to ”reset windup” can occur unless special precautions are taken. The controllers that are *not* selected, should be reset (for normal PID controller this is done by adjusting the value of the controller integral) such that for the present controller measurement, the presently selected manipulated variable value is obtained. Commonly used terms for this type of functionality are ”putting the inactive controllers in tracking mode” or ”using a feedback relay”. This functionality should be implemented with some care, as faulty implementations which permanently lock the inactive controllers are known to have been used. On a digital control system, the controllers should do the following for each sample interval:
  - Read in the process measurement.
  - Calculate new controller output.
  - The selector now selects the controller output to be implemented on the manipulated variable.
  - The controllers read in the implemented manipulated variable value.
  - If the implemented manipulated variable value is different from the controller output, the internal variables in the controller (typically the integral value) should be adjusted to obtain the currently implemented manipulated variable value as controller output, for the current process measurement.

For PI controllers, the simple PI anti-windup scheme from Chapter 6.3.1 of the Course Notes may be used.

[1]Provided appropriate scaling of variables is used, the auctioneering control structure may be a better alternative to using selective control with the selection based on controller inputs.

The control problem

Consider the small plant depicted in Fig. 1. In normal operation, the flow controller should be active, but the pressure controller should take over if the downstream pressure becomes too high. This is achieved by selecting the lower of the two controller outputs, and applying that as the valve opening. A downstream disturbance affects both pressure and flow, and can cause the pressure control to take priority over the flow control.

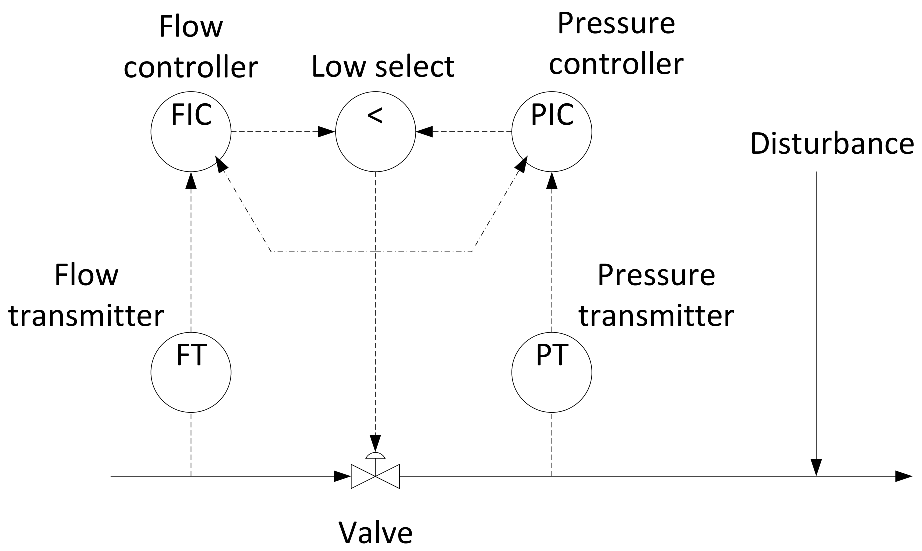

Figure 1: Schematic of simple plant with two controllers and a selector. Note that the applied control signal is fed back to the controllers.

In Fig. 1, solid arrows indicate process flows (in pipelines), dashed lines indicate signal transmission, and the dash-dotted lines indicate 'feedback' from the signal actually applied to the valve back to the controllers, for avoiding windup. In this assignment we will only study the effects of the selector, but one should note that there is little additional complexity involved in also having limited range of actuation of the valve, e.g., to the range ± 1.

A simple model of the plant is given as

$$\begin{bmatrix} f(s) \\ p(s) \end{bmatrix} = \begin{bmatrix} \frac{5}{10s+1} \\ \frac{10}{100s+1} \end{bmatrix} u(s) + \begin{bmatrix} \frac{-1}{20s+1} \\ \frac{2}{20s+1} \end{bmatrix} d(s) \tag{1}$$

where f is the flowrate, p is the pressure, u is the valve position, and d is the disturbance. Note that these variables, as always when using transfer function models, are expressed in *deviation variables*, and thus both negative valve positions, pressures and flowrates *do* make sense[2].

[2] In addition, the variables are scaled, so not too much emphasis should be placed on the magnitude of the variables

(a) **Model implementation**
Implement the model of the plant in Simulink. The two controllers may both be given the setpoint zero. The disturbance should be modelled as a square wave with period 500.

(b) **Controller tuning**
Tune each of the PI controllers, without taking saturation or the selector into account. Any tuning methodology could be used (this should not be very difficult anyway, since the plant in each of the loops is Strictly Positive Real). However, completely unrealistic tunings should be avoided (i.e., for asymptotically stable plants like in this example, the closed loop time constant should not be orders of magnitude faster than the open loop time constant).

(c) **Simulation without anti-windup**
Implement the resulting PI controllers in Simulink - without accounting for windup, and simulate with the disturbance active and both controllers operating (i.e., with the selector).
Comment on the results of the simulation.

(d) **Simulation with anti-windup**
Implement the PI controllers with anti-windup, and redo the simulation.
Comment on the results and compare to the simulation results without anti-windup.