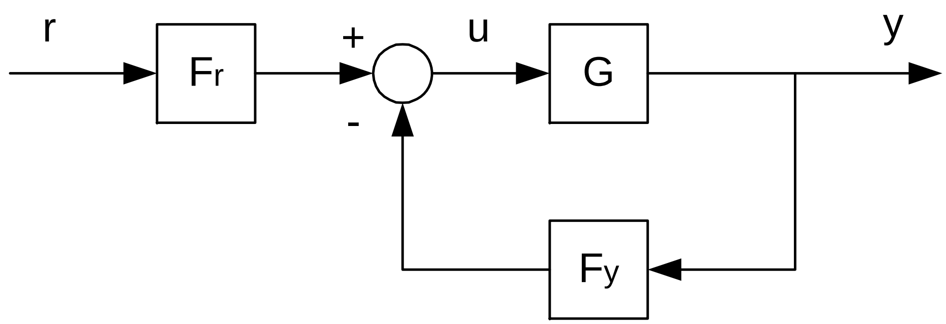
Controller Structures and Controller Design

- Design of multivariable controllers
- *RGA*-analysis
- Design specifications
- Internal model control (IMC)

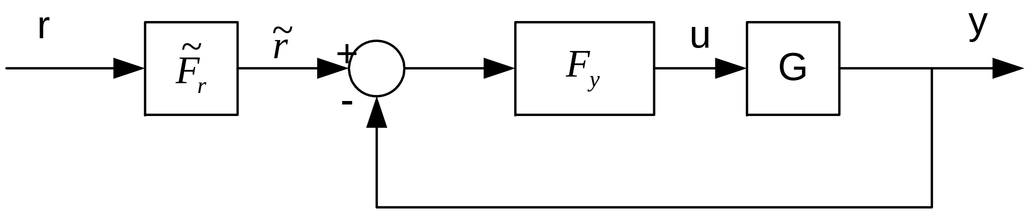
Aalto University
School of Electrical Engineering

Model-based controller structures



$$u = F_r r - F_y y$$ two-degree-of-freedom configuration

$$F_r = F_y$$
$$u = F_y (r - y)$$ one-degree-of-freedom configuration



$$u = F_y\left(\tilde{F}_r r - y\right) \qquad \tilde{r} = \tilde{F}_r r$$

A two-degrees-of-freedom structure can be interpreted as a one-degree-of-freedom structure, in which the reference signal has been filtered by a prerefilter.

1. Design first F_y such that S and T fulfil the desired specifications $\tilde{r} \rightarrow y$
2. If the servo-properties are inadequate, design the prefilter $\tilde{F}_r$

The prefilter can be used to smoothen the variations in the reference signal (lowpass filter).

On the other hand, the bandwidth (from r to z) can be increased by increasing the gain in those frequencies for which T has been designed low because of model uncertainty. But: larger bandwidth needs bigger control signals.

Design procedure

- Model the process and its uncertainty, specify the disturbances.
- Design the control schema, scale the equations if needed, choose appropriate scales for the actuators and measurement devices.
- Design the controller.
- Test by simulations, implement and validate the design.

Multivariable controllers

Main difficulty: cross-couplings, change in one input variable affects several output variables.

Ex. Control of the temperature and flow rate of tap water; the larger the cross-couplings are, the more difficult is control .

Measure for the cross-couplings: **RGA** (relative gain array)

For any square matrix *A*

$$\mathrm{RGA}(A) = A.*\left(A^{-1}\right)^{T}$$

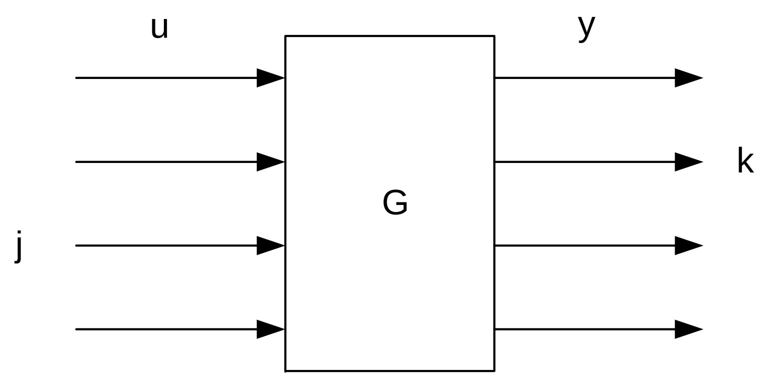
$$\mathrm{RGA}(A) = A.*\left(A^{-1}\right)^T$$

Ex. $A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}$ $\left(A^{-1}\right)^T = \begin{bmatrix} -2 & 1.5 \\ 1 & -0.5 \end{bmatrix}$

$$\mathrm{RGA}(A) = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} .* \begin{bmatrix} -2 & 1.5 \\ 1 & -0.5 \end{bmatrix} = \begin{bmatrix} -2 & 3 \\ 3 & -2 \end{bmatrix}$$

RGA properties:

1. Each row and column sum is unity.
2. RGA remains invariant in scalings with diagonal matrices $\mathrm{RGA}(A) = \mathrm{RGA}(D_1 A D_2)$



Interpretation of *RGA* from control viewpoint:

How does the *i*'th input affect the *j*:th output?

$$y_k = G_{kj}(p)u_j$$

Heuristically :
RGA element = 1, good

$$y_k = \frac{1}{\left(G^{-1}(p)\right)_{jk}} u_j$$

Decentralized control
(Pairing-problem)

Can the multivariable control problem be divided into, several SISO loops, which are controlled separately without bothering about other loops?

Determine control u_i solely based on measurement y_j

and reference r_j

$$u_i = F_r^i r_j - F_y^i y_j$$

But: in control design such combinations y_j and u_i

must be found, which have the strongest couplings.

This is the **pairing-problem**. (Which measurement is used to determine each control?)

To determine that the (partly heuristical) *RGA*-measure can be used.

It can be proven that

-If each SISO-loop in the decentralized control configuration is stable and further $RGA(G(i\omega)) = I, \quad \forall \omega$ then the closed loop system is stable.

-If one or more of the elements of $RGA(G(0))$ is negative, and a diagonal (decentralized) controller is used, the closed loop is unstable or at least goes unstable, if some of the SISO-loops is broken.

Design rule

1. Try to place the diagonal elements of $RGA(G(i\omega_c))$

   as close to unity as possible in the complex plane.
   (Near the cross-over frequency or bandwidth.)

2. Avoid negative elements in the main diagonal of
   $RGA(G(0))$

Note. These specifications are often hard to meet!

Ex. Consider again the system

$$G(s)=\begin{bmatrix}\frac{2}{s+1} & \frac{3}{s+2}\\ \frac{1}{s+1} & \frac{1}{s+1}\end{bmatrix}$$

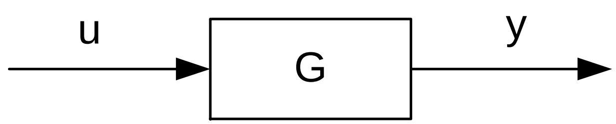
and $$G(0)=\begin{bmatrix}2 & 1.5\\ 1 & 1\end{bmatrix} \qquad \mathrm{RGA}\big(G(0)\big)=\begin{bmatrix}4 & -3\\ -3 & 4\end{bmatrix}$$

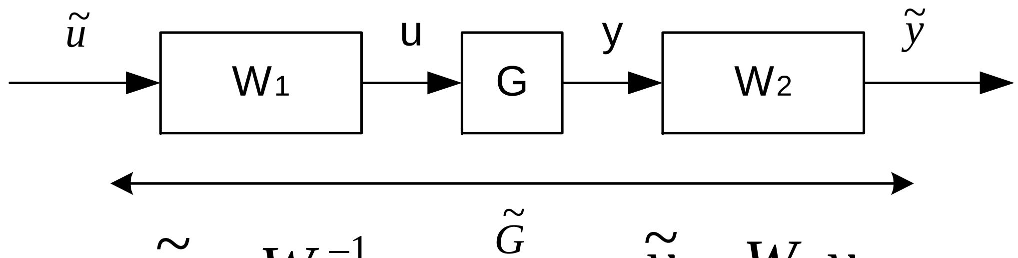
$$G(i5)=\begin{bmatrix}0.0769-0.3846i & 0.2069-0.5172i\\ 0.0385-0.1923i & 0.0385-0.1923i\end{bmatrix}$$

$$\mathrm{RGA}(G(i5))=\begin{bmatrix}-1.7692+1.1538i & 2.7692-1.1538i\\ 2.7692-1.1538i & -1.7692+1.1538i\end{bmatrix}$$

The difficulties in couplings $u_1 \leftrightarrow y_2, \quad u_2 \leftrightarrow y_1$

are already seen from the negative elements is zero frequency





$$\tilde{u} = W_1^{-1} u \qquad \tilde{y} = W_2 y$$

$$\tilde{y} = \tilde{G}\tilde{u} = \underbrace{W_2 G W_1}_{\tilde{G}\text{ (diag.)}} \tilde{u}$$

Design a controller for the modified (diagonal) model

$$\tilde{u} = -F_y^{\text{diag}}\tilde{y}$$

which in terms of the original variables is

$$u = -W_1 F_y^{\text{diag}} W_2 y$$

The control structure is *diagonal* (*decoupled*).

Diagonalization can be done in zero frequency, or a dynamical diagonalization generally or in a specified frequency can be tried.

Example.
$$G(s)=\begin{bmatrix}\frac{2}{s+1} & \frac{3}{s+2}\\ \frac{1}{s+1} & \frac{1}{s+1}\end{bmatrix}$$

Recall that this system has a *RHP*-zero in the point s = 1. But does it show in control?

$$F_y = F_r = \begin{bmatrix}K_1\frac{s+1}{s} & -\frac{3K_2(s+0.5)}{s(s+2)}\\ -K_1\frac{s+1}{s} & \frac{2K_2(s+0.5)}{s(s+1)}\end{bmatrix}$$ compensator

$$GF_y = \begin{bmatrix}\frac{K_1(-s+1)}{s(s+2)} & 0\\ 0 & \frac{K_2(s+0.5)(-s+1)}{s(s+1)^2(s+2)}\end{bmatrix}$$ open system

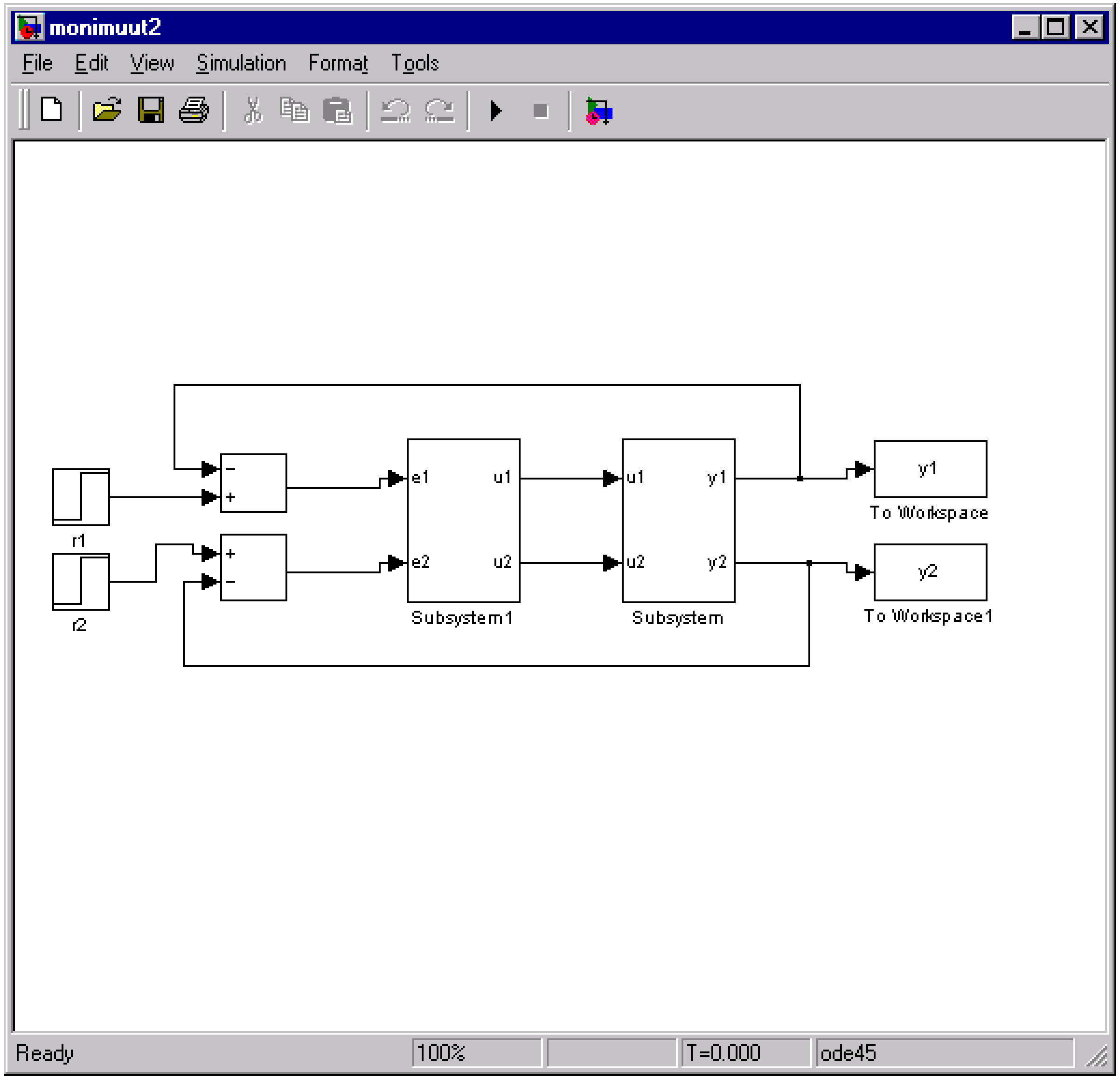
Controller 1:

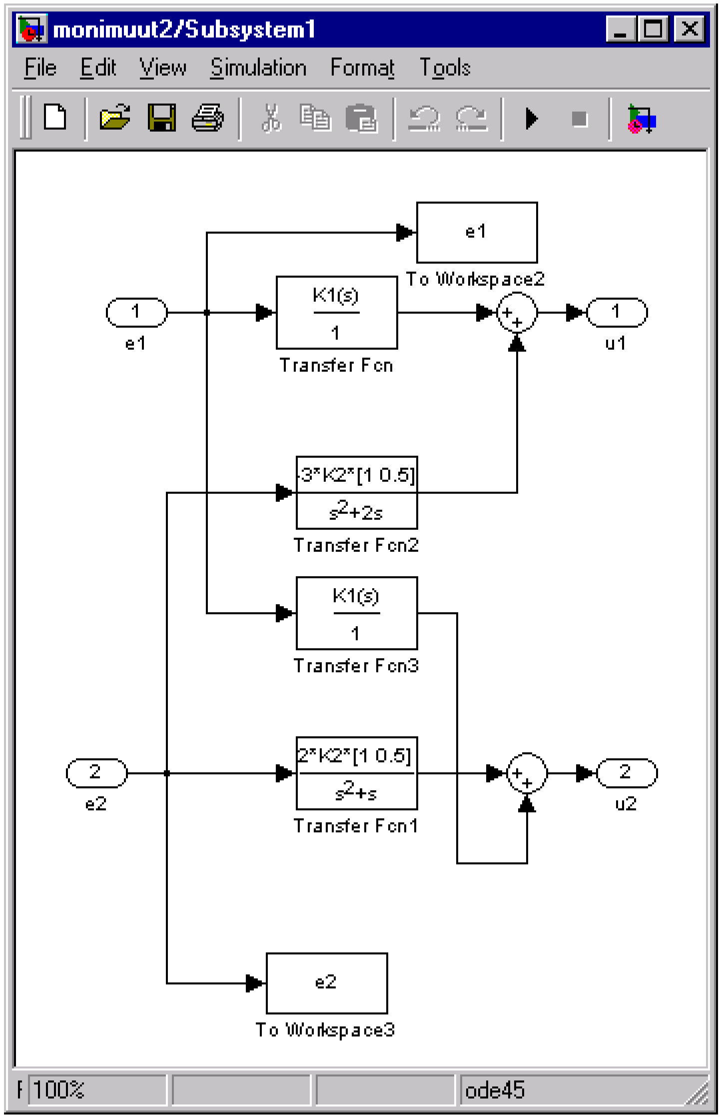
Diagonalizing compensator

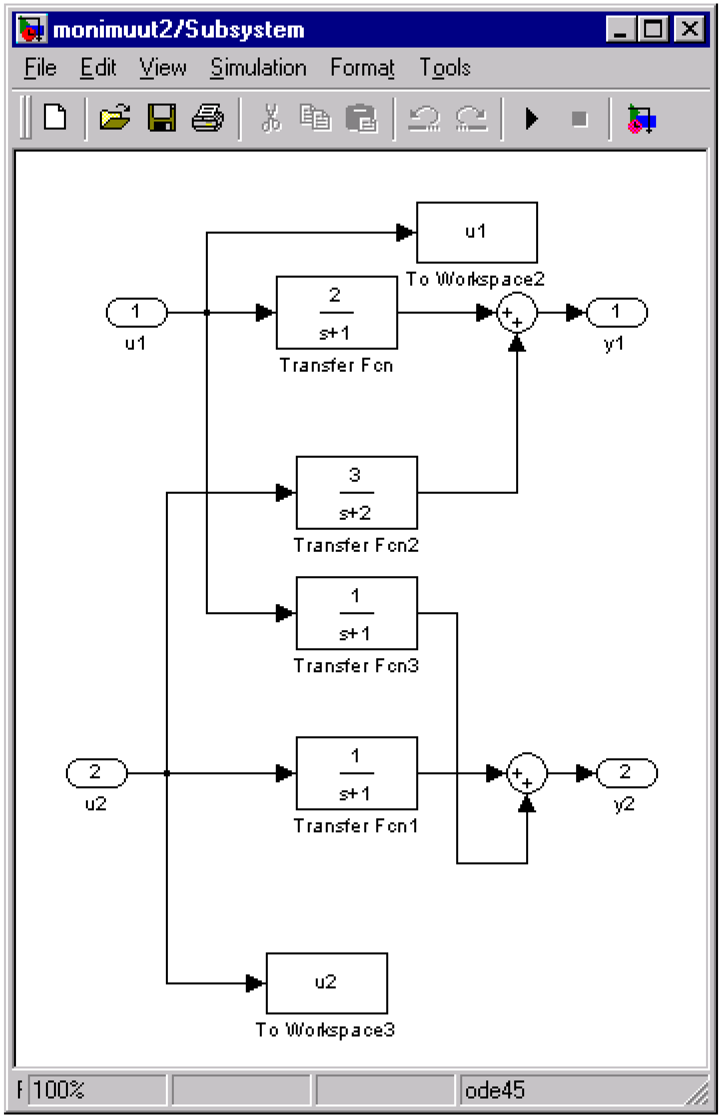
$$F_y = F_r = \begin{bmatrix} K_1 \dfrac{s+1}{s} & -3K_2 \dfrac{(s+0.5)}{s(s+2)} \\ -K_1 \dfrac{s+1}{s} & \dfrac{2K_2(s+0.5)}{s(s+1)} \end{bmatrix}$$

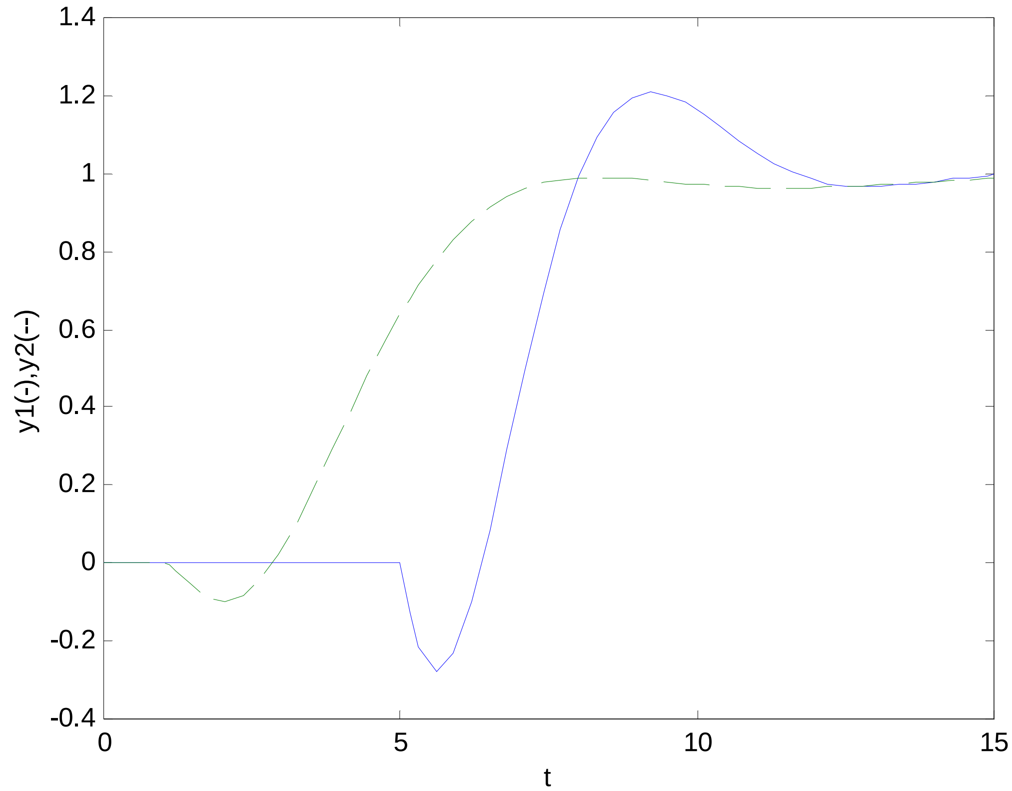
gives the loop transfer function (matrix)

$$GF_y = \begin{bmatrix} \dfrac{K_1(-s+1)}{s(s+2)} & 0 \\ 0 & \dfrac{K_2(s+0.5)(-s+1)}{s(s+1)^2(s+2)} \end{bmatrix}$$


monimuut2
File Edit View Simulation Format Tools
r1
r2
e1
u1
e2
u2
Subsystem1
u1
y1
u2
y2
Subsystem
y1
To Workspace
y2
To Workspace1
Ready
100%
T=0.000
ode45


monimuut2/Subsystem1
File Edit View Simulation Format Tools
e1
To Workspace2
1
e1
K1(s)
1
Transfer Fcn
1
u1
3*K2*[1 0.5]
s2+2s
Transfer Fcn2
K1(s)
1
Transfer Fcn3
2
e2
2*K2*[1 0.5]
s2+s
Transfer Fcn1
2
u2
e2
To Workspace3
100%
ode45


monimuut2/Subsystem
File Edit View Simulation Format Tools
u1
To Workspace2
1
u1
2
s+1
Transfer Fcn
1
y1
3
s+2
Transfer Fcn2
1
s+1
Transfer Fcn3
2
u2
1
s+1
Transfer Fcn1
2
y2
u2
To Workspace3
100%
ode45



Steps at $t = 1$ and $t = 5$.

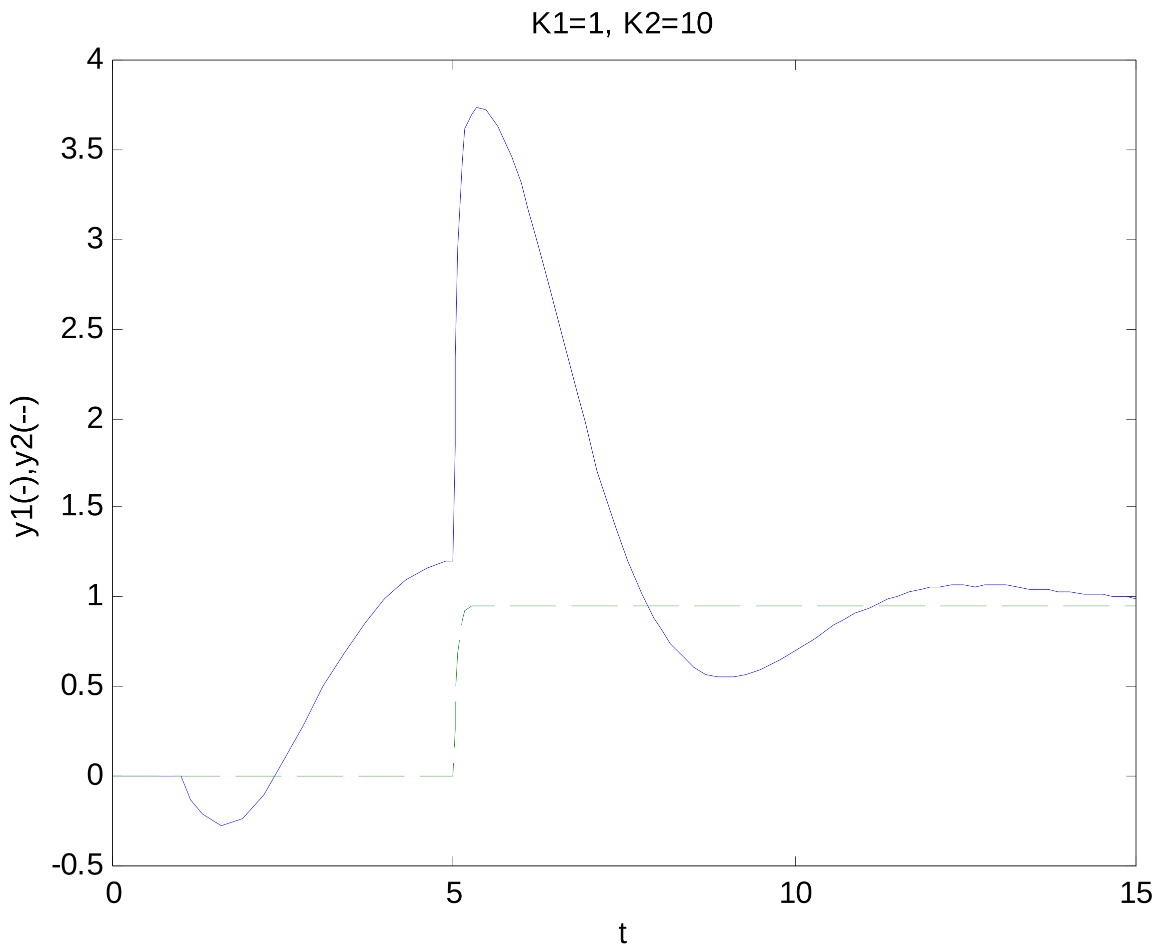
The bandwidth of neither output cannot exceed the (approx) value ½ rad/s= 0.5 rad/s.

Controller 2:

$$F_y = F_r = \begin{bmatrix} K_1 \dfrac{s+1}{s} & K_2 \\ -K_1 \dfrac{s+1}{s} & K_2 \end{bmatrix} \qquad GF_y = \begin{bmatrix} \dfrac{K_1(-s+1)}{s(s+2)} & \dfrac{K_2(5s+7)}{(s+2)(s+1)} \\ 0 & \dfrac{2K_2}{s+1} \end{bmatrix}$$

$$G_c = T = \begin{bmatrix} \dfrac{K_1(-s+1)}{s(s+2)} & g_{12}(s) \\ 0 & \dfrac{2K_2}{s+1+2K_2} \end{bmatrix}$$

It is seen that the RHP-zero has been removed from the term (2,2), but it still remains in term (1,1).

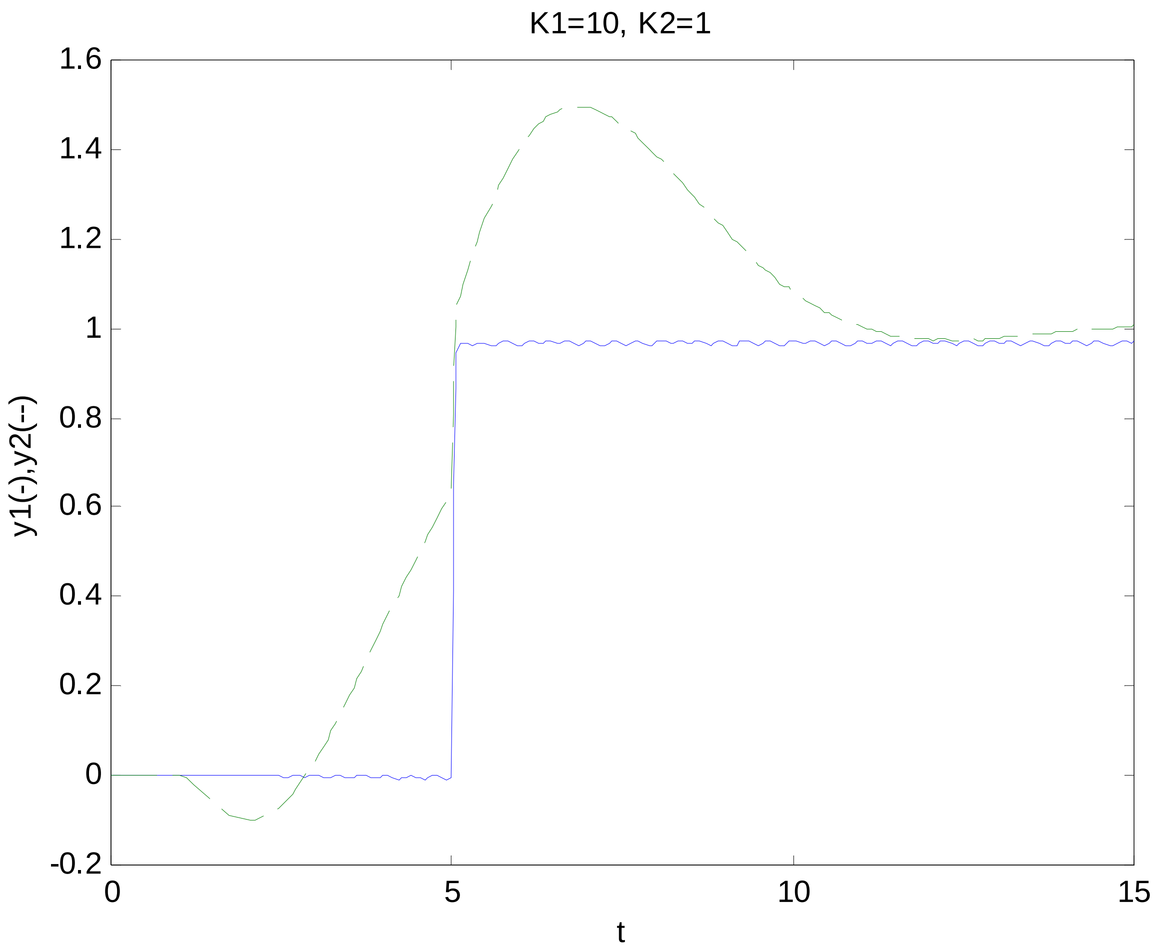

K1=1, K2=10
y1(-),y2(--)
t
4
3.5
3
2.5
2
1.5
1
0.5
0
-0.5
0
5
10
15

Controller 3:

$$F_y = F_r = \begin{bmatrix} K_1 & -\dfrac{3K_2(s+0.5)}{(s+2)(s+1)} \\ K_1 & \dfrac{2K_2(s+0.5)}{s(s+1)} \end{bmatrix}$$

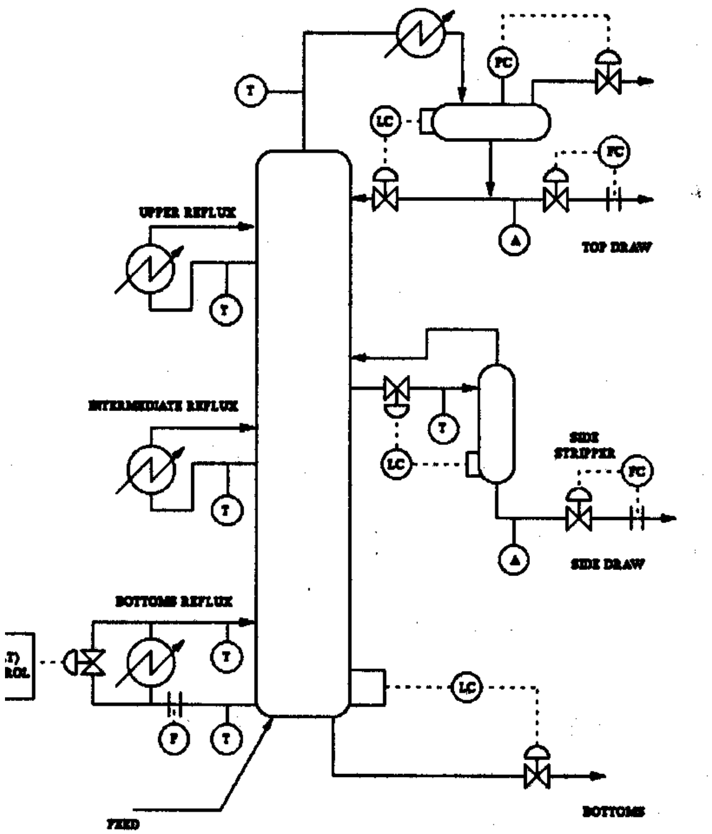
$$\begin{bmatrix} \dfrac{K_1(5s+7)}{(s+1)(s+2)} & 0 \\ \dfrac{2K_1}{s+1} & \dfrac{K_2(-1+s)(s+0.5)}{s(s+1)^2(s+2)} \end{bmatrix}$$

Loop gain



K1=10, K2=1
1.6
1.4
1.2
1
0.8
0.6
0.4
0.2
0
-0.2
y1(-),y2(--)
0
5
10
15
t

Example: Distillation column



$$G(s)=\begin{bmatrix} \dfrac{4.05e^{-27s}}{50s+1} & \dfrac{1.77e^{-28s}}{60s+1} & \dfrac{5.88e^{-27s}}{50s+1} \\ \dfrac{5.39e^{-18s}}{50s+1} & \dfrac{5.72e^{-14s}}{60s+1} & \dfrac{6.90e^{-15s}}{40s+1} \\ \dfrac{4.38e^{-20s}}{33s+1} & \dfrac{4.42e^{-22s}}{44s+1} & \dfrac{7.20}{19s+1} \end{bmatrix}$$

$$G_d(s)=\begin{bmatrix} \dfrac{1.44e^{-27s}}{40s+1} \\ \dfrac{1.83e^{-15s}}{20s+1} \\ \dfrac{1.26}{32s+1} \end{bmatrix}$$

the third output variable is not important in this process

calculate the RGA of two first rows

$$\text{RGA}(\tilde{G}(0)) = \begin{bmatrix} 0.3203 & -0.5946 & 1.2744 \\ -0.0170 & 1.5733 & -0.5563 \end{bmatrix}$$

$$\text{RGA}(\tilde{G}(i/50)) = \begin{bmatrix} 0.4794 - 0.3558i & -0.6325 - 0.0158i & 1.1532 + 0.3716i \\ -0.1256 + 0.3558i & 1.5763 + 0.0158i & -0.4707 - 0.3716i \end{bmatrix}$$

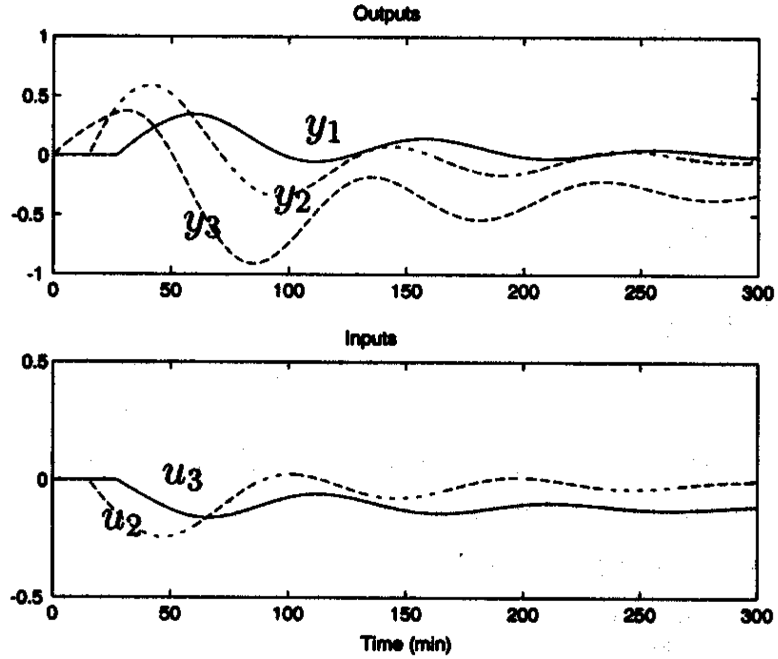
”Coupling”: output 1 – control 3; output 2 – control 2

Arrange the elements of the transfer function according to this coupling.

$$G(s) = \begin{bmatrix} \dfrac{5.88e^{-27s}}{50s+1} & \dfrac{1.77e^{-28s}}{60s+1} \\ \dfrac{6.90e^{-15s}}{40s+1} & \dfrac{5.72e^{-14s}}{60s+1} \end{bmatrix}$$

By ignoring the off-diagonal terms the PI-controller can be designed by classical methods

$$F_y^{dec}(s) = \begin{bmatrix} \dfrac{63s+1}{50s} & 0 \\ 0 & \dfrac{67s+1}{50s} \end{bmatrix}$$


Outputs
1
0.5
0
-0.5
-1
0
50
100
150
200
250
300
y1
y2
y3
Inputs
0.5
0
-0.5
0
50
100
150
200
250
300
u3
u2
Time (min)

Then do a stationary *decoupling*

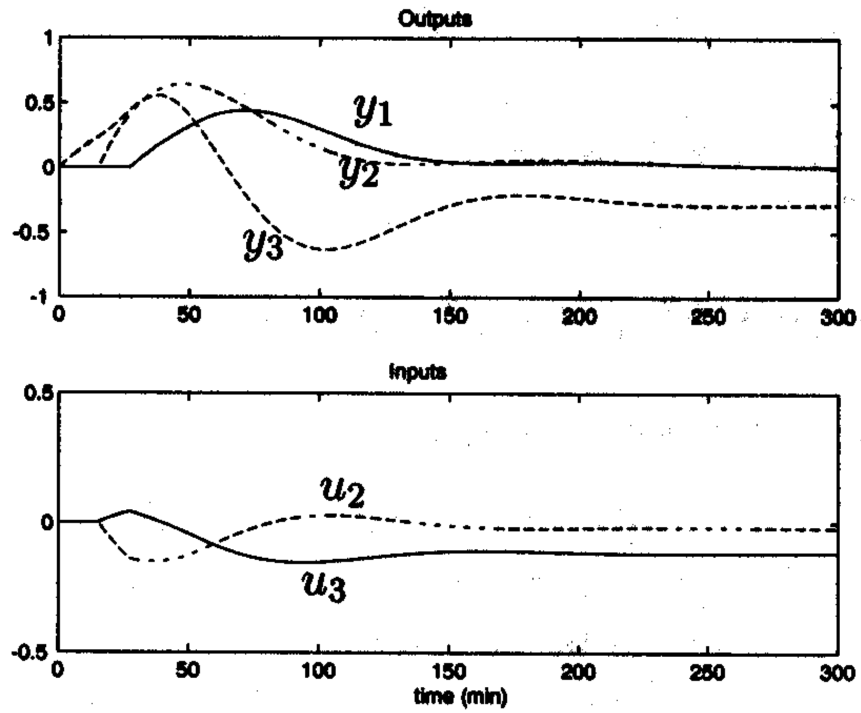
$$G(0)=\begin{bmatrix}5.88 & 1.77\\ 6.90 & 5.72\end{bmatrix} \qquad W_1 = G^{-1}(0) \qquad W_2 = I$$

$$\tilde{G}(s)=\begin{bmatrix}\dfrac{1.57e^{-27s}}{50s+1}-\dfrac{0.57e^{-28s}}{60s+1} & X\\ X & \dfrac{1.57e^{-14s}}{60s+1}-\dfrac{0.57e^{-15s}}{40s+1}\end{bmatrix}$$

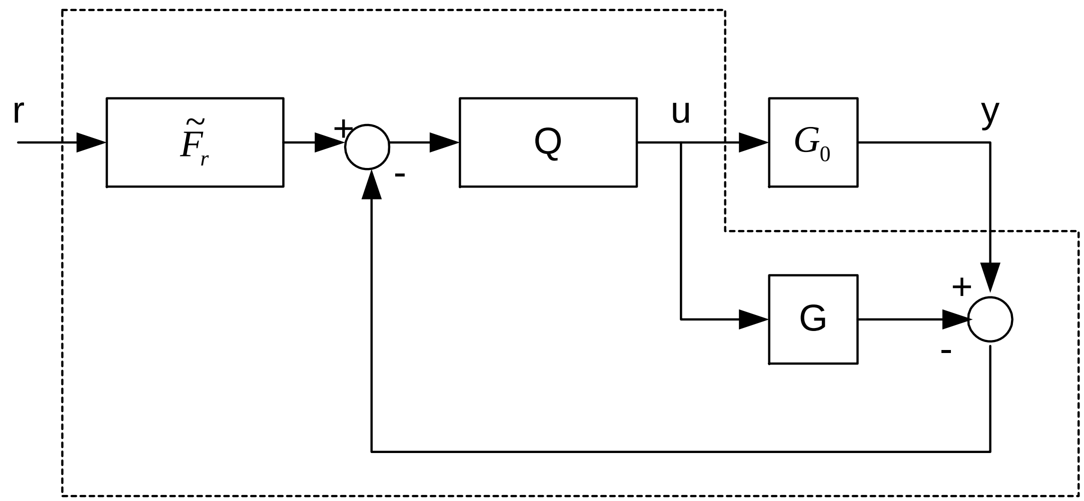
$$F_y^{dec}(s) = \begin{bmatrix} \frac{63s+1}{50s} & 0 \\ 0 & \frac{55s+1}{50s} \end{bmatrix}$$

and the controller is

$$F_y(s) = W_1 F_y^{diag}(s) = \begin{bmatrix} \frac{0.267(63s+1)}{50s} & -\frac{0.0826(55s+1)}{50s} \\ -\frac{0.3221(63s+1)}{50s} & \frac{0.2745(55s+1)}{50s} \end{bmatrix}$$


Outputs
1
0.5
0
-0.5
-1
0
50
100
150
200
250
300
y_1
y_2
y_3
Inputs
0.5
0
-0.5
u_2
u_3
time (min)

IMC-control (Internal Model Control)



There would be no need for feedback, if the model were accurate and there would be no disturbances. Why not use only ”new information” in the feedback loop?

$$y - Gu \qquad u = -Q(y - Gu)$$

Aalto University
School of Electrical
Engineering

$$u = -Q(y - Gu) + Q\tilde{r}, \quad \tilde{r} = \tilde{F}_r r$$

By IMC-control this control structure is meant, in which the matrix Q must be chosen and the process model G is an essential part.

The transfer function between y and u becomes

$$u = -(I - QG)^{-1} Qy \qquad F_y = (I - QG)^{-1} Q$$

In the nominal case $G_0 = G$ the closed loop is

$$G_c = \left(I + GF_y\right)^{-1} GF_y \tilde{F}_r = GQ\tilde{F}_r$$

Sensitivity functions

$$T = (I + GF_y)^{-1} GF_y = GQ$$

$$S = I - GQ$$

The transfer function from a disturbance at the process output *w* to the control *u* is

$$G_{wu} = -(I + F_y G)^{-1} F_y = -Q$$

That gives a natural interpretation to the ”tuning parameter” Q .

But this holds in the nominal case only $G_0 = G$

If there is a model error, the closed loop transfer function becomes

$$G_c = G_0 \left(I + Q \left(G_0 - G \right) \right)^{-1} Q \tilde{F}_r$$

When G is stable, all transfer functions are stable, *if and only if* Q is stable. (The system is internally stable). Note: nominal case only.

In fact: Q parameterizes all stabilizing controllers, the so-called *Youla-parameterization*, Q-*parameterization*.

But how to choose Q in practice?

It would be nice to choose

$$Q = G^{-1} \qquad S \equiv 0 \quad G_c \equiv I$$

which is however impossible $\quad F_y \equiv \infty$

but take this as a starting point (SISO-case)

1. If G has more poles than zeros $\quad G^{-1}$

   is not realizable

$$Q(s) = \frac{1}{(\lambda s + 1)^n} G^{-1}(s)$$

n and λ
tuning parameters

2. Process has a RHP-zero, which would cancel out in

GG^{-1} ; this would lead to an internally unstable system, compare to pole-zero cancellations discussed earlier.

If G has a term $(-\beta s + 1)$ in the numerator

a. Forget it in forming $Q \approx G^{-1}$

b. Replace it with $(\beta s + 1)$; error in phase only

3. G has a delay term $e^{-s\tau}$

The delay can be approximated by the Pade approximation

$$e^{-s\tau} \approx \frac{1 - s\tau/2}{1 + s\tau/2}$$

and then the normal design procedure.

Example.

$$G(s) = \frac{1}{\tau s + 1} \qquad Q(s) = \frac{\tau s + 1}{\lambda s + 1}$$

$$F_y(s) = \frac{\dfrac{\tau s + 1}{\lambda s + 1}}{1 - \dfrac{1}{\lambda s + 1}} = \frac{\tau s + 1}{\lambda s}$$

which is the PI-controller

$$F_y(s) = \frac{\tau}{\lambda}\left(1 + \frac{1}{s\tau}\right)$$