Høgskolen i Buskerud. Finn Haugen (finn@techteach.no).

Løsning til deleksamen 2 i SEKY3322 Kybernetikk 3

Tid: 7. april 2008. Varighet 4 timer. Vekt i sluttkarakteren: 30%.

Hjelpemidler: Ingen trykte eller håndskrevne hjelpemidler. Kalkulator ikke tillatt.

---

1. (20% vekt)

$$u(k) = -Gx(k) \tag{1}$$
$$= -\left[G_{12}x_1(k) + G_{12}x_2(k) + G_{13}x_3(k) + G_{14}x_4(k)\right] \tag{2}$$

2. (5) Man får ikke beregnet noen stasjonær regulatorforsterkning (Riccatilikningen konvergerer ikke).

3. (15) Vekten W økes, dvs. at kostnaden på pådraget økes. Dermed vil regulatoren automatisk beregne mindre pådragsverdier, siden kriteriet jo skal minimeres.

4. (10) Separasjonsprinsippet betyr at man kan designe tilstandsestimatoren og regulatoren hver for seg, dvs. beregne Kalmanfilterforsterkningen og regulatorforsterkningen, uten hensyn til hverandre.

5. (25) Hensikten med dekopling er motvirke koplingene i prosessen slik at det ikke blir interaksjon mellom reguleringssløyfene. Sløyfene er med andre ord dekoplede, men prosesskoplingene er der fremdeles, selvsagt.

   Figur 1 viser et blokkdiagram for et multivariabelt reguleringssystem med dekopling.

   Vi skal nå utlede transferfunksjonene $D_1(s)$ og $D_2(s)$ i dekopleren. Vi starter med $D_1(s)$. Den skal gjøre at nettovirkningen som z_1 har på y_1 er null. Med andre ord (jf. blokkdiagrammet):

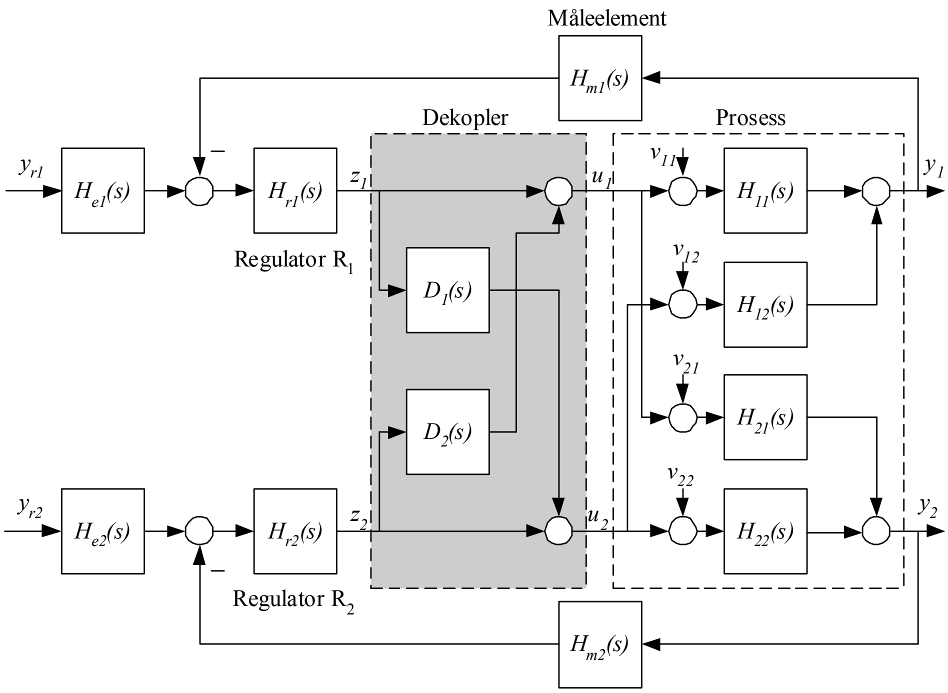
$$D_1(s)H_{22}(s)z_1(s) + H_{21}(s)z_1(s) = 0 \tag{3}$$

   for alle z_1. Dette er oppfylt med

$$D_1(s) = -\frac{H_{21}(s)}{H_{22}(s)} \tag{4}$$

   Tilsvarende blir

$$D_2(s) = -\frac{H_{12}(s)}{H_{11}(s)} \tag{5}$$



Figur 1: Multivariabelt reguleringssystem med dekopling og PID-regulatorer

6. (25) *The derivation below is taken from a text-book written by F. Haugen. It is not expected that the students write such a detailed answer, but the main points should be included.*

   It assumed that the process model is a state space model on the following form:

$$\dot{x} = f(x, v) + B(x, v) \cdot u \tag{6}$$

   or, simpler,

$$\dot{x} = f + Bu \tag{7}$$

   x is the state vector, v is the disturbance vector, and u is the control vector. f is a vector of scalar functions, and B is a matrix of scalar functions. Assume that the output vector is

$$y = x \tag{8}$$

   By taking the derivative of (8) and using (7) we obtain the following differential equation describing the process output vector:

$$\dot{y} = f + Bu \tag{9}$$

   Assume that r_y is the reference (or setpoint) of y. With the above assumptions, we derive the control function as follows: We start by defining the *transformed control vector* as

$$z \stackrel{\text{def}}{=} f + Bu \tag{10}$$

Then (9) can be written as

$$\dot{y} = z \tag{11}$$

which are n decoupled or independent *integrators* (n is the number of state variables), because $y(t) = \int_0^t z \, d\tau$. The transfer function from z to y is

$$\frac{y(s)}{z(s)} = \frac{1}{s} \tag{12}$$

We will now derive the control function for this integrator process, and thereafter derive the final control function. How can you control an integrator? With feedback and feedforward! A good choice for the *feedback controller* is a PI controller (proportional plus integral) because the controller should contain integral action to ensure zero steady-state control error in the presence of unmodelled disturbances (and there are such in a real system). The proportional action is necessary to get a stable control system (if a pure integral controller acts on an integration process the closed loop system becomes marginally stable, i.e. it is pure oscillatory). The multiloop feedback PI controller is

$$z_{\text{fb}} = K_p e + K_i \int_0^t e d\tau \tag{13}$$

where e is the control error:

$$e \stackrel{\text{def}}{=} r_y - y \tag{14}$$

In addition to the PI feedback action the controller should contain *feedforward* from the reference r_y to get fast reference tracking when needed (assuming the reference is varying). The feedforward control function can be derived by substituting the process output y in the process model (11) by r_y and then solving for y, giving

$$z_{\text{ff}} = \dot{r}_{y_f} \tag{15}$$

where index f indicates lowpass filter which may be of first order. A pure time differentiation should not be implemented because of noise amplification by the differentiation. Therefore the reference should be lowpass filtered before its time derivative is calculated.

The control function for the process (11) based on the sum of the feedback control function and the feedforward control function is as follows:

$$
\begin{aligned}
z &= z_{\text{fb}} + z_{\text{ff}} &(16)\\
&= \underbrace{K_p e + K_i \int_0^t e d\tau}_{z_{\text{fb}}} + \underbrace{\dot{r}_{y_f}}_{z_{\text{ff}}} &(17)
\end{aligned}
$$

Deriving the final control function, that is, the formula for the control vector u: From (10) we get

$$u = B^{-1}(z - f) \tag{18}$$

Here we use (17) to get the final control function:

$$\underline{u = B^{-1}\left(K_p e + K_i \int_0^t e d\tau + \dot{r}_{y_f} - f\right)} \tag{19}$$

How to tune the PI controller included in the control function (19)? In (19) K_p and K_i are diagonal matrices:

$$K_p = \begin{bmatrix} K_{p_1} & 0 & \cdots & 0 \\ 0 & K_{p_2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & K_{p_n} \end{bmatrix} = \text{diag}(K_{p_1}, K_{p_2}, \cdots, K_{p_n}) \tag{20}$$

$$K_i = \begin{bmatrix} K_{i_1} & 0 & \cdots & 0 \\ 0 & K_{i_2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & K_{i_n} \end{bmatrix} = \text{diag}(K_{i_1}, K_{i_2}, \cdots, K_{i_n}) \tag{21}$$

where the scalar values are

$$K_{i_j} = \frac{K_{p_j}}{T_{i_j}} \tag{22}$$

where K_{p_j} is the proportional gain and T_{i_j} is the integral time of control loop no. j. K_{p_j} and T_{i_j} can be calculated in several ways. *Skogestad's method* is one option[**?**]. Skogestad's method is reviewed below. The design principle of Skogestad's method is as follows. The control system *tracking transfer function* $T(s)$, which is the transfer function from the reference or setpoint to the (filtered) process measurement, is *specified* as a first order transfer function with time delay:

$$T(s) = \frac{y_{mf}(s)}{y_{m_{SP}}(s)} = \frac{1}{T_C s + 1} e^{-\tau s} \tag{23}$$

where T_C is the time constant of the control system which *the user must specify*, and τ is the process time delay which is *given* by the process model.

Skogestad's tuning formulas for an integrator with gain equal to one and time-delay equal to zero are as follows:

$$\underline{K_{p_j} = \frac{1}{T_{C_j}}} \tag{24}$$

and

$$\underline{T_{i_j} = k_j T_{C_j}} \tag{25}$$

where T_{C_j} is the specified time constant of feedback loop no. j, and k_j is a coefficient that can be set to 1.44.