Multivariable control using single loops

- Interactions
- Choice of pairings (RGA)

Multivariable process

Distillation column

“Increasing reflux L from 1.0 to 1.1 changes $y_D$ from 0.95 to 0.97, and $x_B$ from 0.02 to 0.03”

“Increasing boilup V from 1.5 to 1.6 changes $y_D$ from 0.95 to 0.94, and $x_B$ from 0.02 to 0.01”

Steady-State Gain Matrix

$$\begin{pmatrix}
\Delta y_D \\
\Delta x_B 
\end{pmatrix}
= G(0)
\begin{pmatrix}
\Delta L \\
\Delta V 
\end{pmatrix}$$

Effect of input 1 ($\Delta L$) on output 2 ($\Delta x_B$)

Can also include dynamics:

$$G(s) = \begin{pmatrix}
0.2 & -0.1 \\
\frac{1}{1 + 50s} & \frac{1}{0.1} \\
\frac{1}{1 + 40s} & \frac{1}{1 + 40s} 
\end{pmatrix}
\rightarrow \Delta y_D \quad \text{(Time constant 50 min for $y_D$)}$$

$$\rightarrow \Delta x_B \quad \text{(time constant 40 min for $x_B$)}$$
Analysis of Multivariable processes

Process Model 2x2

"Open-loop"
\[ y_1(t) = g_{11}(s)u_1(t) + g_{12}(s)u_2(t) \]
\[ y_2(t) = g_{21}(s)u_1(t) + g_{22}(s)u_2(t) \]

INTERACTIONS: Caused by nonzero \( g_{12} \) and/or \( g_{21} \)

RGA: Consider effect of \( u_1 \) on \( y_1 \)

1) "Open-loop" (\( C_2 = 0 \)): \[ y_1 = g_{11}(s)u_1 \]
2) "Closed-loop" (close loop 2, \( C_2 \neq 0 \)): \[ y_1 = \frac{g_{11}(s) - g_{12}g_{21}C_2}{1 + g_{22}C_2} u_1 \]

Derivation:

Close loop 2: \[ u_2 = -c_2(y_2 - y_2) \]
Here: \( y_2 = g_{21}u_1 + g_{22}u_2 \) and assume \( y_2 = 0 \):
\[ u_2 = -c_2(g_{21}u_1 + g_{22}u_2) \]
\[ u_2 = \frac{-c_2g_{21}}{1 + g_{22}C_2} u_1 \]

Effect of \( u_1 \) on \( y_1 \) with loop 2 closed is then:
\[ y_1 = g_{11}u_1 + g_{12}u_2 = g_{11} \left( 1 - \frac{g_{12}g_{21}C_2}{1 + g_{22}C_2} \right) u_1 \]
Limiting Case $C_2 \rightarrow \infty$ (perfect control of $y_2$) (steady state)

$$y_1 = \left[ g_{11}(s) - \begin{smallmatrix} g_{12} & g_{21} \end{smallmatrix} \right] u_1 = g_{11}(1/\lambda_{11}) u_1$$

How much has “gain” from $u_1$ to $y_1$ changed by closing loop 2 with perfect control?

$$\text{Relative Gain} = \frac{(y_1/u_1)_{CL}}{(y_1/u_1)_{OL}} = \frac{g_{11}}{g_{11} - \frac{g_{12} g_{21}}{g_{22}}} = \frac{1}{1 - \frac{g_{12} g_{21}}{g_{11} g_{22}}}$$

The relative Gain Array (RGA) is the matrix formed by considering all the relative gains

$$\text{RGA} = \mathbf{\Lambda} = \begin{bmatrix} \lambda_{11} & \lambda_{12} \\ \lambda_{21} & \lambda_{22} \end{bmatrix} = \begin{bmatrix} 
\frac{(y_1/u_1)_{CL}}{(y_1/u_1)_{OL}} & \frac{(y_1/u_2)_{CL}}{(y_1/u_2)_{OL}} \\
\frac{(y_2/u_1)_{CL}}{(y_2/u_1)_{OL}} & \frac{(y_2/u_2)_{CL}}{(y_2/u_2)_{OL}} 
\end{bmatrix}$$

Example from before

$$G = \begin{bmatrix} 0.2 & -0.1 \\ 0.1 & -0.1 \end{bmatrix} \quad \mathbf{\Lambda} = \begin{bmatrix} \lambda_{11} & \lambda_{12} \\ \lambda_{21} & \lambda_{22} \end{bmatrix} = \frac{1}{\begin{bmatrix} 0.1 & 0.1 \\ 0.2 & 0.1 \end{bmatrix}} = 2$$

$$\text{RGA} = \begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix}$$

Only acceptable pairings:

- $u_1 \rightarrow y_1$
- $u_2 \rightarrow y_2$

Not recommended:

- $u_1 \rightarrow y_2$
- $u_2 \rightarrow y_1$

With integral action:

Negative RGA $\Rightarrow$ individual loop unstable OR overall system unstable when individual loops saturates
Property of RGA:
- Columns and rows always sum to 1
- RGA independent of scaling (units) for u and y.

RGA for general case:

\[
[RGA]_{ij} = \frac{(g_{ij})_{OL}}{(g_{ij})_{CL}} = [G]_{ij} [G^{-1}]_{ji}
\]

= element-by-element multiplication of G and G^{-1}T.

Matlab: RGA = G.*pinv(G).

Example

\[
G = \begin{bmatrix} 5 & 10 & 1 \\ 20 & -10 & 0 \\ 18 & 0 & 2 \end{bmatrix}
\]

\[
G = \begin{bmatrix} 5 & 10 & 1 \\ 20 & -10 & 0 \\ 18 & 0 & 2 \end{bmatrix}
\]

\[
\text{rga} = \begin{bmatrix} 0.3125 & 1.2500 & -0.5625 \\ 1.2500 & -0.2500 & 0 \\ -0.5625 & 0 & 1.5625 \end{bmatrix}
\]

Conclusion: of the 6 possible pairings only one has positive RGA’s

Use of RGA:

1) **Interactions**
- RGA-element ($\lambda$) > 1: Smaller gain by closing other loops (“fighting loops” gives slower control)
- RGA-element ($\lambda$) < 1: Larger gain by closing other loops (can be dangerous)
- RGA-element ($\lambda$) negative: Gain reversal by closing other loops (Oops!)

**Rule 1. Avoid pairing on negative steady-state relative gain – otherwise you get instability if one of the loops become inactive (e.g. because of saturation)**

**Rule 2. Choose pairings corresponding to RGA-elements close to 1**

Traditional: Consider Steady-state

Better (improved Rule 2): Consider frequency corresponding to closed-loop time constant
Example

\[ G = \begin{pmatrix} 16.8 & 30.5 & 4.30 \\ -16.7 & 31.0 & -1.41 \\ 1.27 & 54.1 & 5.40 \end{pmatrix}, \quad RGA(G) = \begin{pmatrix} 1.50 & 0.99 & -1.48 \\ -0.41 & 0.97 & 0.45 \\ -0.08 & -0.95 & 2.03 \end{pmatrix} \]

Only diagonal pairings give positive steady-state RGA's!

Distillation

\[ y = \begin{pmatrix} y_D \\ x_B \end{pmatrix}, \quad u = \begin{pmatrix} L \\ V \end{pmatrix} \]

\[ G(0) = \begin{pmatrix} 87.8 & -86.4 \\ 108.2 & -109.6 \end{pmatrix}, \quad RGA(0) = \begin{pmatrix} 35 & -34 \\ -34 & 35 \end{pmatrix} \]
**Distillation**

\[ y = \begin{pmatrix} y_D \\ x_B \end{pmatrix}, \quad u = \begin{pmatrix} L \\ V \end{pmatrix} \]

\[ G(0) = \begin{pmatrix} 87.8 & -86.4 \\ 108.2 & -109.6 \end{pmatrix}, \quad RGA(0) = \begin{pmatrix} 35 & -34 \\ -34 & 35 \end{pmatrix} \]

Can break interactions with cascade:
Frequency-dependent RGA with TC

**Sometimes useful: Iterative RGA**

- For large processes, lots of pairing alternatives
- RGA evaluated iteratively is helpful for quick screening

\[ RGA(G) = \Lambda(G) = G \times (G^{-1})^T \]

\[ \Lambda^2(G) = \Lambda(\Lambda(G)) \]

\[ \Lambda^\infty = \lim_{k \to \infty} \Lambda^k(G) \]

- Converges to “Permuted Identity” matrix (correct pairings) for generalized diagonally dominant processes.
- Can converge to incorrect pairings, when no alternatives are dominant.
- Usually converges in 5-6 iterations
Example of Iterative RGA

\[ G = \begin{bmatrix} 1 & 2 \\ -1 & 1 \end{bmatrix} \]

\[ \Lambda = \begin{bmatrix} 0.33 & 0.67 \\ 0.67 & 0.33 \end{bmatrix} \quad \Lambda^2 = \begin{bmatrix} -0.33 & 1.33 \\ 1.33 & -0.33 \end{bmatrix} \]

\[ \Lambda^3 = \begin{bmatrix} -0.07 & 1.07 \\ 1.07 & -0.07 \end{bmatrix} \quad \Lambda^4 = \begin{bmatrix} 0.00 & 1.00 \\ 1.00 & 0.00 \end{bmatrix} \]

Correct pairing

Exercise. Blending process

- Mass balances (no dynamics)
  - Total: \( F_1 + F_2 = F \)
  - Sugar: \( F_1 = x F \)
(a) Linearize balances and introduce: \( u_1 = dF_1, u_2 = dF_2, y_1 = F_1, y_2 = x, \)
(b) Obtain gain matrix \( G \) \( y = G u \)
(c) Nominal values are \( x = 0.2 \) [kg/kg] and \( F = 2 \) [kg/s]. Find \( G \)
(d) Compute RGA and suggest pairings
(e) Does the pairing choice agree with “common sense”?
Equation.

(a) The balances “mass in = mass out” for total mass and sugar mass are

\[ F_1 + F_2 = F, \quad F_1 = xF \]

Note that the mixing process itself has no dynamics. Linearization yields

\[ dF_1 + dF_2 = dF, \quad F_1 = x^* dF + F^* dx \]

With \( u_1 = dF_1, u_2 = dF_2, \ y_1 = dF \) and \( y_2 = dx \) we then get the model

\[
\begin{align*}
y_1 &= u_1 + u_2 \\
y_2 &= \frac{1-x^*}{x^*} u_1 - \frac{x^*}{x^*} u_2
\end{align*}
\]

where \( x^* = 0.2 \) is the nominal steady-state sugar fraction and \( F^* = 2 \) kg/s is the nominal amount.

(b,c) The transfer matrix then becomes

\[ G(s) = \begin{pmatrix}
\frac{1}{F^*} & 1 \\
1-x^* & \frac{x^*}{F^*}
\end{pmatrix} = \begin{pmatrix}
\frac{1}{0.4} & 1 \\
0.2 & 0.1
\end{pmatrix} \]

(4) The corresponding RGA matrix is (at all frequencies)

\[ A = \begin{pmatrix}
x^* & 1-x^* \\
1-x^* & x^*
\end{pmatrix} = \begin{pmatrix}
0.2 & 0.8 \\
0.8 & 0.2
\end{pmatrix} \]

For decentralized control, it then follows from pairing rule 1 ("prefer pairing on RGA elements close to 1") that we should pair on the off-diagonal elements; that is, use \( u_1 \) to control \( y_2 \) and use \( u_2 \) to control \( y_1 \).

(c) This corresponds to using the largest stream (water, \( u_2 \)) to control the amount \( y_1 = F \), which is reasonable from a physical point of view. Also note that the RGA-elements are always between 0 and 1 for this process, and the RGA-elements are all 0.5, corresponding to "switching" the pairings, when \( x^* \leq 0.5 \), which is when the two feed streams are equal.