Multiloop Control Systems

1. Introduction
2. The relative gain array (RGA)
3. Pairing input-output variables
4. Dynamic considerations
5. Multiloop controller tuning
6. Reducing control loop interactions
Introduction

- Multiloop control approach
  - Pair input and output variables together
  - Design SISO controller for each input/output pair
  - Attempt to minimize interactions between controllers

- Challenges
  - Often not obvious how to pair variables to minimize control loop interactions
  - Number of possible pairings for $nxn$ system = $n!$
  - Detuning of SISO controllers often necessary due to interactions
**Distillation Column Example**

- Controller outputs – $x_D$, $x_B$, $P$, $h_D$, $h_B$
- Manipulated inputs – $D$, $B$, $R$, $Q_D$, $Q_B$
- Best pairings not obvious
- Number of possible pairings = $n! = 5! = 120$
- Need efficient method to screen possible pairings
The Relative Gain

- Quantifies the change in steady-state gain between an input-output pair that occurs when other control loops are closed.
- Provides a measure of steady-state process interactions from only gain information.
- Relative gain between input $j$ and output $i$

$$\lambda_{ij} = \frac{\left( \frac{\partial y_i}{\partial u_j} \right)_u}{\left( \frac{\partial y_i}{\partial u_j} \right)_y} \quad i, j \in [1, n]$$

- $\left( \frac{\partial y_i}{\partial u_j} \right)_u$ is the open-loop gain with all other control loop open.
- $\left( \frac{\partial y_i}{\partial u_j} \right)_y$ is the closed-loop gain with all other control loop closed.
Relative Gain Calculation for 2x2 System

- **Steady-state model**
  \[ y_1 = K_{11}u_1 + K_{12}u_2 \quad y_2 = K_{21}u_1 + K_{22}u_2 \]

- **Open-loop gain**
  \[ \left( \frac{\partial y_1}{\partial u_1} \right)_{u_2=0} = K_{11} \]

- **Closed-loop gain**
  \[ y_2 = 0 \quad \Rightarrow \quad u_2 = -\frac{K_{21}}{K_{22}}u_1 \]
  \[ y_1 = K_{11}\left(1 - \frac{K_{12}K_{21}}{K_{11}K_{22}}\right)u_1 \quad \Rightarrow \quad \left( \frac{\partial y_1}{\partial u_1} \right)_{y_2=0} = K_{11}\left(1 - \frac{K_{12}K_{21}}{K_{11}K_{22}}\right) \]

- **Relative gain**
  \[ \lambda_{11} = \frac{\left( \frac{\partial y_1}{\partial u_1} \right)_{u_2=0}}{\left( \frac{\partial y_1}{\partial u_1} \right)_{y_2=0}} = \frac{K_{11}}{K_{11}\left(1 - \frac{K_{12}K_{21}}{K_{11}K_{22}}\right)} = \frac{1}{1 - \frac{K_{12}K_{21}}{K_{11}K_{22}}} \]
General Relative Gain Calculation

- Steady-state model

\[ y = Ku \quad y = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} \quad u = \begin{bmatrix} u_1 \\ \vdots \\ u_n \end{bmatrix} \quad K = \begin{bmatrix} K_{11} & \cdots & K_{1n} \\ \vdots & \ddots & \vdots \\ K_{n1} & \cdots & K_{nn} \end{bmatrix} \]

- Calculation of H matrix

\[ H = (K^{-1})^T = \begin{bmatrix} H_{11} & \cdots & H_{1n} \\ \vdots & \ddots & \vdots \\ H_{n1} & \cdots & H_{nn} \end{bmatrix} \]

- Relative gain

\[ \lambda_{ij} = K_{ij} H_{ij} \quad \text{2x2 system} \quad \lambda_{11} = \frac{1}{1 - \frac{K_{12} K_{21}}{K_{11} K_{22}}} \]
The Relative Gain Array (RGA)

- **Definition**
  \[ \Lambda = \begin{bmatrix} \lambda_{11} & \cdots & \lambda_{1n} \\ \vdots & \ddots & \vdots \\ \lambda_{n1} & \cdots & \lambda_{nn} \end{bmatrix} \]

- Each row and column of the RGA must sum to unity
  - 2x2 system
    \[ \Lambda = \begin{bmatrix} \lambda_{11} & \lambda_{12} \\ \lambda_{21} & \lambda_{22} \end{bmatrix} \]
  
  - General \( nxn \) system – only need to calculate \( n-1 \) elements per row or column
Input-Output Pairings: 2x2 Case

- **RGA**
  \[ \Lambda = \begin{bmatrix} \lambda & 1-\lambda \\ 1-\lambda & \lambda \end{bmatrix} \]
  \[ \lambda = \left( \frac{\partial y_1/\partial u_1}{\partial y_1/\partial u_1} \right)_{u_2} \]

- **Five possible cases**
  - \( \lambda = 1 \): open-loop and closed-loop gains are identical; pair \( u_1 \) and \( y_1 \)
  - \( \lambda = 0 \): open-loop gain is zero; do not pair \( u_1 \) and \( y_1 \)
  - \( 0 < \lambda < 1 \): closed-loop gain is larger than the open-loop gain; pair \( u_1 \) and \( y_1 \) if \( \lambda \) is “close” to 1
  - \( \lambda > 1 \): closed-loop gain is smaller than the open-loop gain; pair \( u_1 \) and \( y_1 \) if \( \lambda \) is “close” to 1
  - \( \lambda < 0 \): open-loop and closed-loop gains have opposite signs; do not pair \( u_1 \) and \( y_1 \)
Input-Output Pairings: General Case

- **RGA**

  \[
  \Lambda = \begin{bmatrix}
  \lambda_{11} & \cdots & \lambda_{1n} \\
  \vdots & \ddots & \vdots \\
  \lambda_{n1} & \cdots & \lambda_{nn}
  \end{bmatrix}
  \quad \lambda_{ij} = \frac{\left(\frac{\partial y_i}{\partial u_j}\right)_u}{\left(\frac{\partial y_i}{\partial u_j}\right)_y}
  \]

- **Pairing rules**
  
  » \( \lambda_{ij} = 1 \): pair \( u_j \) and \( y_i \)
  
  » \( \lambda_{ij} < 0 \): do not pair \( u_j \) and \( y_i \)
  
  » \( \lambda_{ij} = 0 \): do not pair \( u_1 \) and \( y_1 \)
  
  » \( 0 < \lambda_{ij} < 1 \): pair \( u_1 \) and \( y_1 \) if \( \lambda \) is “close” to 1
  
  » \( \lambda_{ij} > 1 \): pair \( u_1 \) and \( y_1 \) if \( \lambda \) is “close” to 1

- **Caveat**

  » RGA is based only on steady-state information and neglects process dynamics
Blending System Example

- Use $w_A$ and $w_B$ to control $w$ and $x$

- Material balances

$$w = w_A + w_B \quad wx = w_A \quad \Rightarrow \quad x = \frac{w_A}{w_A + w_B}$$

- Open-loop gains

$$K_{11} = \left( \frac{\partial w}{\partial w_A} \right)_{w_B} = 1 \quad K_{12} = \left( \frac{\partial w}{\partial w_B} \right)_{w_A} = 1$$

$$K_{21} = \left( \frac{\partial x}{\partial w_A} \right)_{w_B} = \frac{w_B}{(w_A + w_B)^2} = \frac{1-x}{w} \quad K_{22} = \left( \frac{\partial x}{\partial w_B} \right)_{w_A} = \frac{-w_A}{(w_A + w_B)^2} = -\frac{x}{w}$$

- Relative gain

$$\lambda = \frac{1}{1 - \frac{K_{12}K_{21}}{K_{11}K_{22}}} = x$$
Blending System Example cont.

- **RGA**

\[
\Lambda = \begin{bmatrix}
\lambda & 1-\lambda \\
1-\lambda & \lambda \\
\end{bmatrix} = \begin{bmatrix}
x & 1-x \\
1-x & x \\
\end{bmatrix}
\]

- **Cases**

\[
x = 0.9 \quad \Lambda = \begin{bmatrix}
0.9 & 0.1 \\
0.1 & 0.9 \\
\end{bmatrix} \quad \text{Pair } u_1/y_1, u_2/y_2
\]

\[
x = 0.1 \quad \Lambda = \begin{bmatrix}
0.1 & 0.9 \\
0.9 & 0.1 \\
\end{bmatrix} \quad \text{Pair } u_1/y_2, u_1/y_2
\]

\[
x = 0.5 \quad \Lambda = \begin{bmatrix}
0.5 & 0.5 \\
0.5 & 0.5 \\
\end{bmatrix} \quad \text{No best pairing}
\]
4x4 RGA Example

- **RGA for distillation column**
  
  \[
  \Lambda = \begin{bmatrix}
  0.931 & 0.150 & 0.080 & -0.164 \\
  -0.011 & -0.429 & 0.286 & 1.154 \\
  -0.135 & 3.314 & -0.270 & -1.910 \\
  0.215 & -2.030 & 0.900 & 1.919 \\
  \end{bmatrix}
  \]

- **Pairings**
  
  » \( \lambda_{11} = 0.931 \rightarrow \text{pair } u_1/y_1 \)
  
  » \( \lambda_{24} = 1.154 \rightarrow \text{pair } u_4/y_2 \)
  
  » \( \lambda_{43} = 0.900 \rightarrow \text{pair } u_3/y_4 \)
  
  » Pair \( u_2/y_3 \rightarrow \lambda_{32} = 3.314 \)
Dynamic Considerations: Stability

- Assumptions
  - Each process transfer function $G_{pij}(s)$ is stable, rational and proper.
  - Each controller transfer function $G_{ci}(s)$ contains integral action.
  - Each individual control loop is stable when the other $n-1$ loops are open.

- Niederlinski stability theorem
  - Assume the inputs and outputs are paired as $u_1/y_1$, $u_2/y_2$, …, $u_n/y_n$
  - The multiloop control system is unstable if: $\left| \prod_{i=1}^{n} K_{ii} \right| < 0$
  - Can be applied to different pairing by rearranging $K$ such that the gains of the paired inputs and outputs are on the diagonal
  - The result also provides a sufficient condition for stability if $n = 2$
Dynamic Considerations: Performance

- Transfer function matrix

\[ G_p(s) = \begin{bmatrix} \frac{-2e^{-s}}{10s+1} & \frac{1.5e^{-s}}{s+1} \\ \frac{1.5e^{-s}}{s+1} & \frac{2e^{-s}}{10s+1} \end{bmatrix} \Rightarrow K = \begin{bmatrix} -2 & 1.5 \\ 1.5 & 2 \end{bmatrix} \]

- Pairings based on RGA

\[ \Lambda = \begin{bmatrix} \lambda & 1-\lambda \\ 1-\lambda & \lambda \end{bmatrix} = \begin{bmatrix} 0.64 & 0.36 \\ 0.36 & 0.64 \end{bmatrix} \Rightarrow \text{Pair } u_1/y_1, u_2/y_2 \]

- Pairings based on time constants: \( u_1/y_2, u_2/y_1 \)
Multiloop Controller Tuning

- **Control system design**
  - Use RGA to pair input and output variables
  - Design SISO controller for each input-output pair

- **Standard tuning procedure**
  - Tune each controller with other controllers in manual
  - Place all controllers in automatic and detune controllers as necessary to mitigate interactions

- **Special case**
  - One output is much more important than the other outputs (e.g. reactor temperature vs. reactor liquid level)
  - Only detune controller for least important outputs so important output remains well controlled
Reducing Control Loop Interactions

- **Alternative approaches**
  - Detune individual SISO controllers
  - Select different manipulated inputs and/or controller outputs
  - Utilize a multivariable controller (see Chapter 20 in text)

- **Blending system example**

\[
\begin{align*}
\mathbf{w} &= \mathbf{w}_A + \mathbf{w}_B \\
\mathbf{x} &= \frac{\mathbf{w}_A}{\mathbf{w}_A + \mathbf{w}_B}
\end{align*}
\]

\[
\begin{align*}
\mathbf{u}_1 &= \mathbf{w}_A + \mathbf{w}_B \\
\mathbf{u}_2 &= \mathbf{w}_A \\
\Rightarrow\quad \mathbf{w} &= \mathbf{u}_1 \\
\mathbf{x} &= \frac{\mathbf{u}_2}{\mathbf{u}_1}
\end{align*}
\]

\[
\mathbf{K} = \begin{bmatrix}
1 & 0 \\
-\frac{\mathbf{u}_2}{\mathbf{u}_1} & 1 \\
\end{bmatrix}
\quad \Rightarrow \quad \lambda = \frac{1}{1 - \frac{\mathbf{K}_{12}\mathbf{K}_{21}}{\mathbf{K}_{11}\mathbf{K}_{22}}} = 1
\quad \Rightarrow \quad \mathbf{\Lambda} = \begin{bmatrix}
1 & 0 \\
0 & 1
\end{bmatrix}
\]