CHEN4011
Advanced Modelling and COntrol

Curtin University

Dr. Ranjeet Utikar (RU)

Curtin University
Malaysia

Dr. Jobrun Nandong (JN)

Decentralized Control Systems

Outline

- Multivariable process plants
- Methods of controlling process plant
- Decentralized Control vs. Centralized Control Systems
- Issues in decentralized control system
- Decentralized PID Control Design Methods
- Decoupling controllers

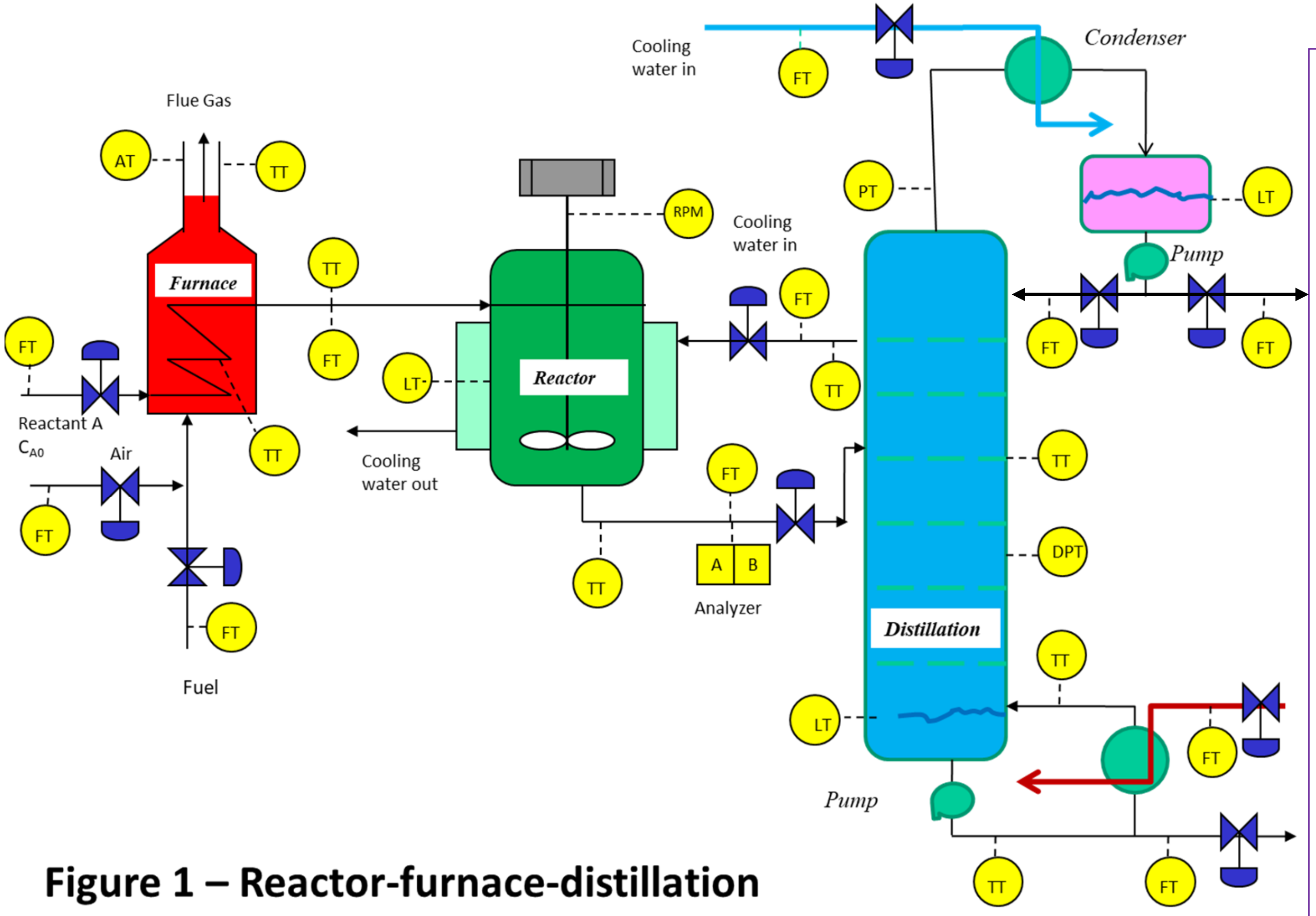
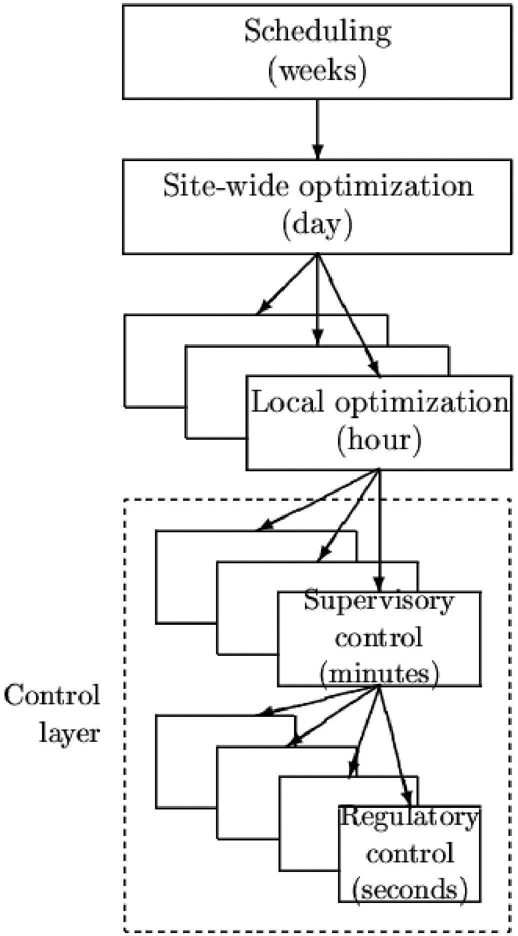


Figure 1 – Reactor-furnace-distillation system

- ✓ **Real Process often have more than one input and one outputs.**
- ✓ **Real process has multi-input and multi-output (MIMO)**
- ✓ **Engineers attempt to select a set of controlled variables from a set of measurement.**
- ✓ **This selection is not a trivial task**
- ✓ **Refer to the Figure 1 – how many measurements are there?**

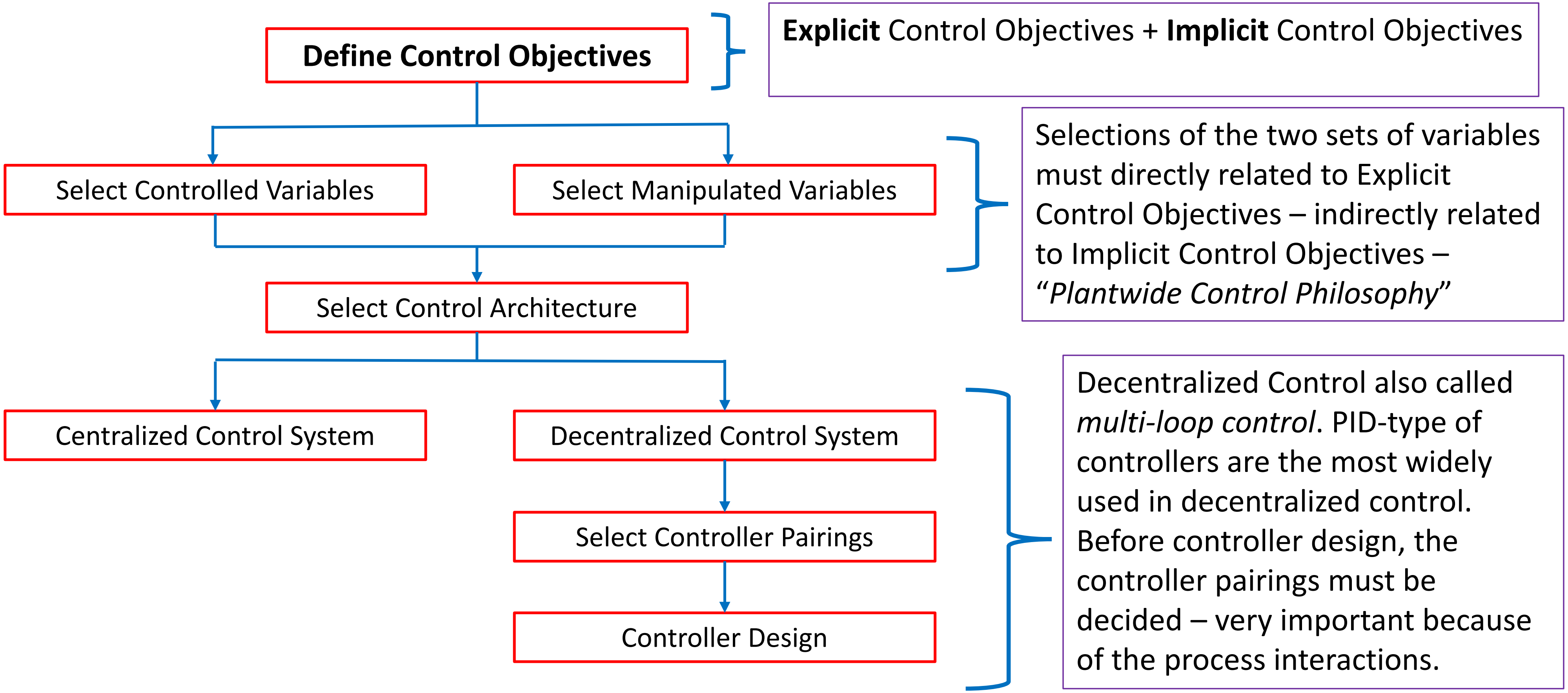


Plantwide Control Hierarchy

- Plantwide control system can be divided into several layers (shown in Figure)
- Regulatory control layer often employs multi-loop PID controllers to control liquid level, temperature, flow rate, pressure, etc. This layer is very important because it provide the stability for the given plant.
- Above the regulatory control, is the supervisory control layer, which provide setpoints to the regulatory layer. Centralized control systems might be adopted in this layer.
- In some advanced plants, real-time monitoring and optimization are implemented. This layer provide optimized conditions to the supervisory layers
- Overall plant might be optimized but the action rate is slower than the local optimization.
- Some plants also include inventory and production scheduling layer which might be performed offline.

Ref: Larsson, T., & Skogestad, S. (2000). Plantwide control-A review and a new design procedure.

Plantwide Control Design Flowchart



Define Control Objectives

Select Controlled Variables

Select Manipulated Variables

Select Control Architecture

Centralized Control System

Decentralized Control System

Select Controller Pairings

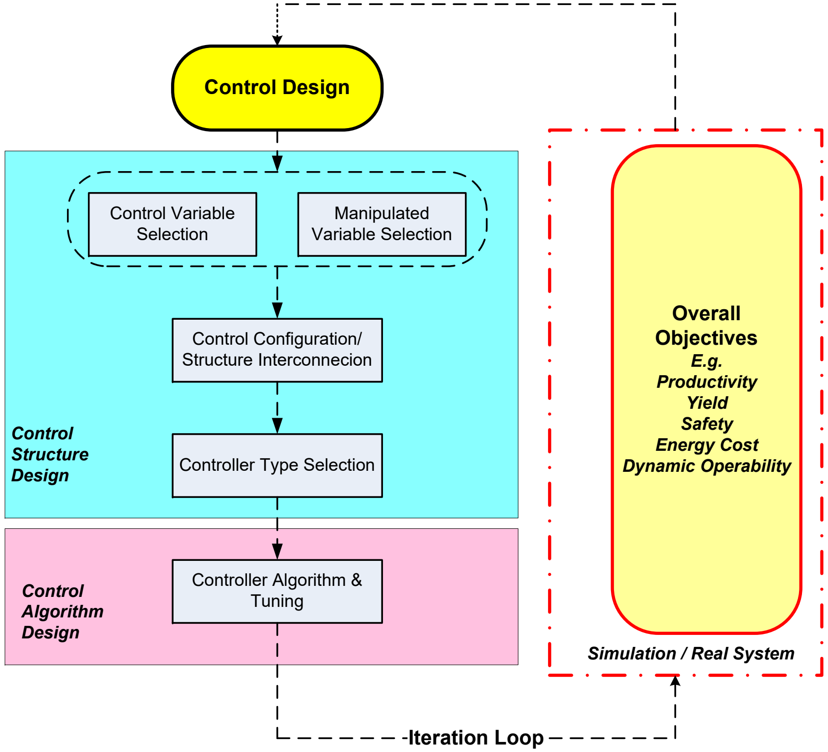
Controller Design

Explicit Control Objectives + **Implicit** Control Objectives

Selections of the two sets of variables must directly related to Explicit Control Objectives – indirectly related to Implicit Control Objectives – *"Plantwide Control Philosophy"*

Decentralized Control also called *multi-loop control*. PID-type of controllers are the most widely used in decentralized control. Before controller design, the controller pairings must be decided – very important because of the process interactions.

Plantwide Control Design Tasks

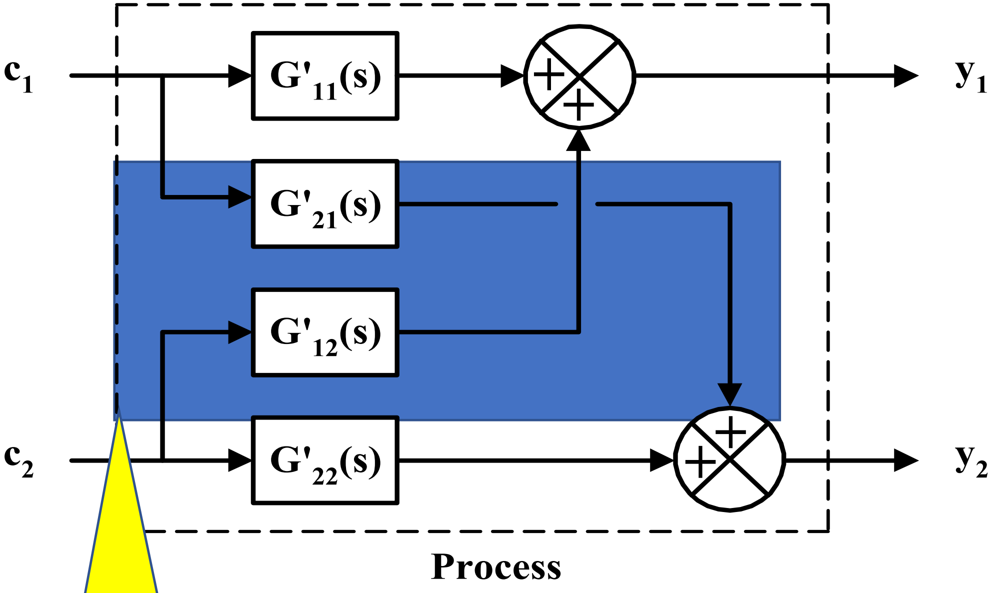


- Plantwide control design might involve several iterations. For example, the controller pairings selected in the first iteration might not perform up to expectation. So, another iteration required to choose different set of pairings.
- Different pairings often require re-design of the individual controllers. Once the last design step is completed, simulation is performed to assess the effectiveness of the complete control design.
- If the desired performance is achieved, the design is completed, else another iteration is required, and so on.

Operating Objectives

- Some typical examples…
  - Maintain smooth operation in distillation i.e. no flooding and no weeping.
  - Equipment protection e.g. no cavitation of pump.
  - Maximum yield of desired product e.g. maximum yield of B.
  - Safe operation e.g. reactor temperature < threshold value
  - Minimum operating cost e.g. minimum steam consumption.
  - Meet production specification e.g. purity meet customer demand.
  - Optimum profit
- **Control Strategy must achieve the entire objectives**

2x2 MIMO Process System



Create process interactions

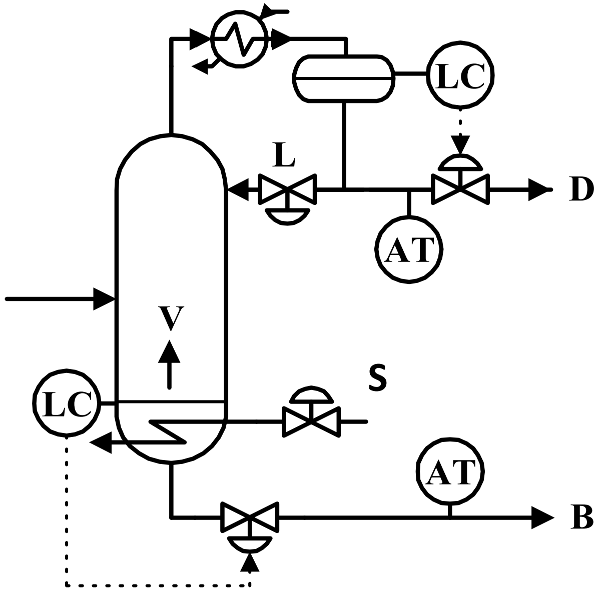
- 2 inputs c_1 and c_2
- 2 outputs y_1 and y_2

Plant transfer function matrix

$$\boldsymbol{G} = \begin{bmatrix} \boldsymbol{G'}_{11} & \boldsymbol{G'}_{12} \\ \boldsymbol{G'}_{21} & \boldsymbol{G'}_{22} \end{bmatrix}$$

- E.g., G'_{12} denotes transfer function from input c_2 to output y_1
- Process interactions – a change in c_1 or c_2 will affect both y_1 and y_2
- Process interaction leads to coupling effect between the first and second control loops.
- Process interactions – one of the major factor affecting the decentralized control performance

2x2 real process example – distillation column



- 2 inputs: (1) Steam Flow S, and (2) Reflux Flow L
- 2 outputs: (1) Distillate composition, and (2) Bottom composition
- Interactions exist between the top composition controller and bottom composition controller
- A change in the reflux flow will affect both top and bottom compositions
- A change in the steam flow will affect both top and bottom compositions
- Note that, the level controller often has weak coupling effects with the top and bottom compositions controllers
- Main consideration for the composition controllers

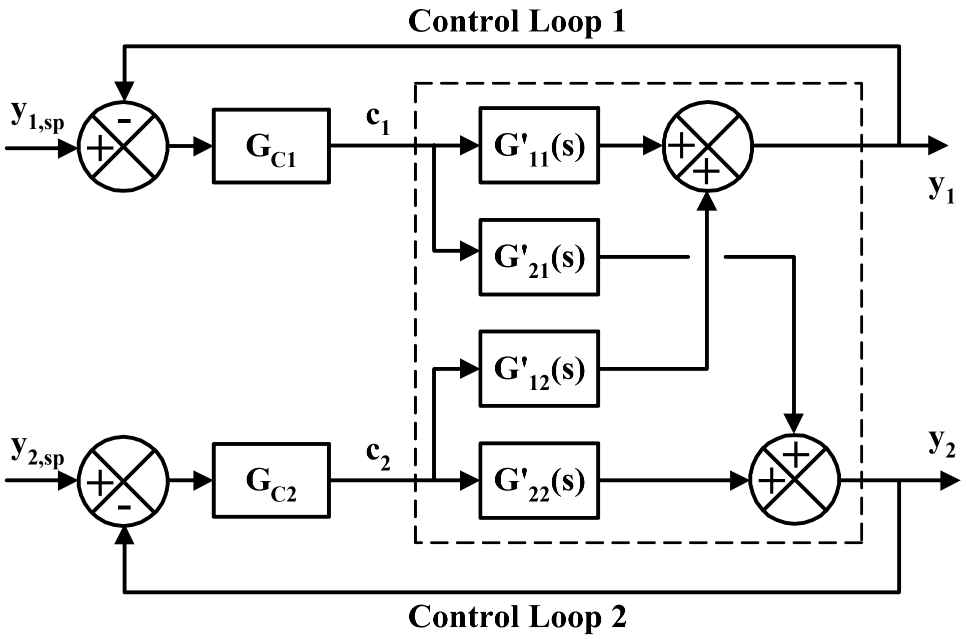
Controller Pairings

- One major task in decentralized control system design – to select controller pairings
- Controller pairings are chosen based on 3 main factors:

1) Process Interactions
2) Dynamic responses
3) Sensitivity to disturbances

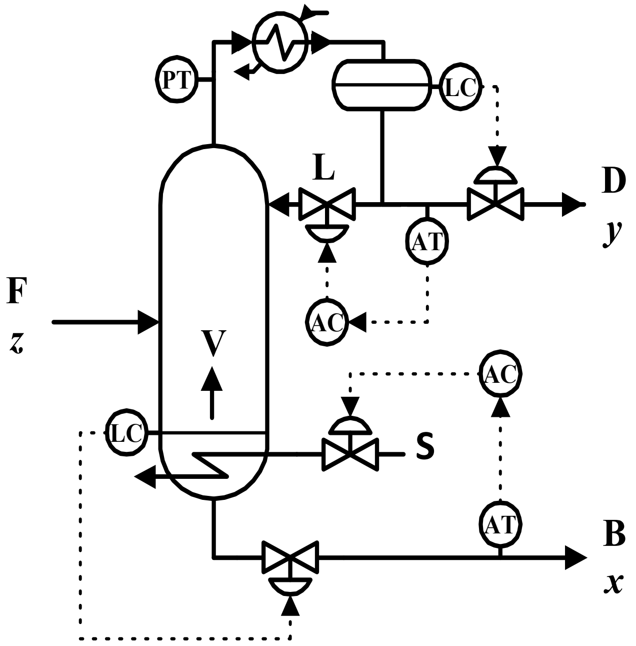
- Improper controller pairings can lead to severe process interactions – causes poor control performance
- Factors can be conflicting with each other – e.g., pairings that lead to minimum process interactions may exhibit slow dynamic responses – it is desirable to have fast dynamic responses.

Multi-Loop Controllers (Decentralized Control)



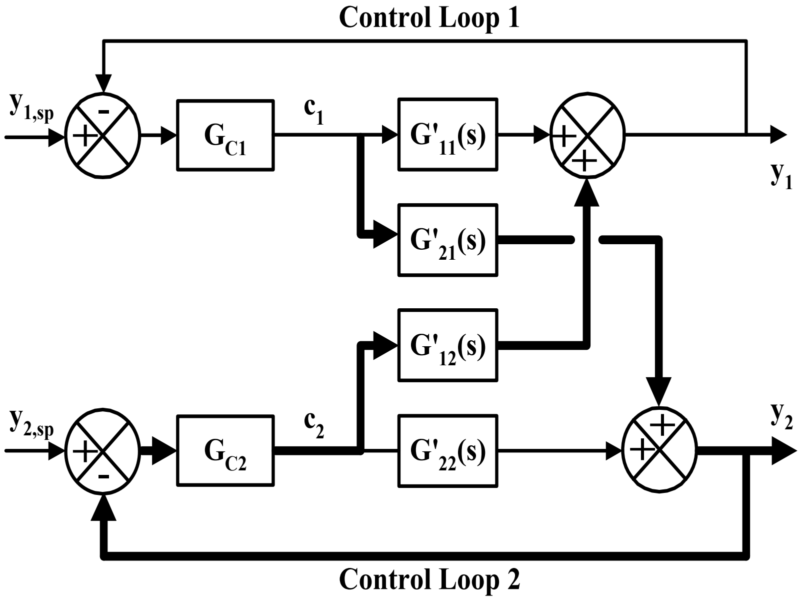
- Two single-input single-output (SISO) control loops – two controllers – multi-loop controllers
- Can we design each controller independently of another?
- What is the implication of the process interactions due to G'_{21} and G'_{12} on the controller design?
- Figure shows direct controller pairings, i.e., $c_1 \sim y_1 /\ c_2 \sim y_2$ pairings.
- Alternatively, indirect pairings - $c_1 \sim y_2 /$ $c_2 \sim y_1$
- Both types of pairings experience different severity of process interactions

Multi-Loop Controllers – Distillation Column



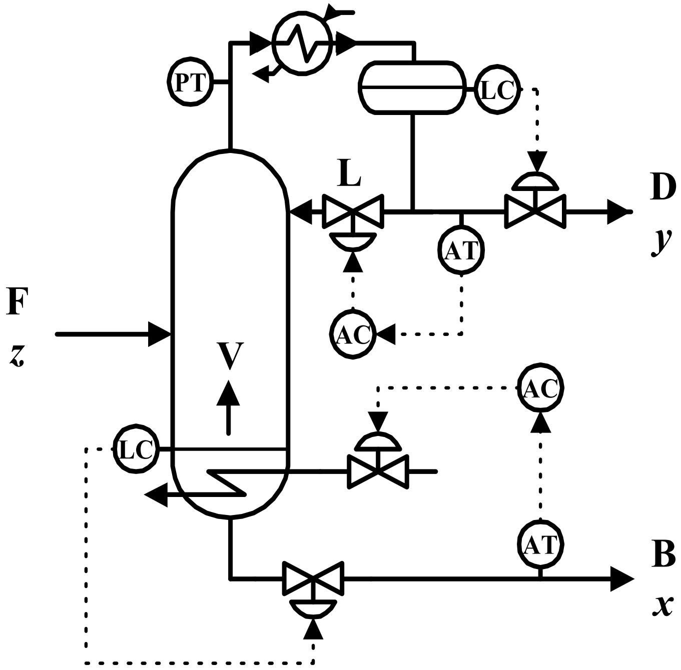
- Reflux flow L is used to control the top composition y
- Steam flow S is used to control the bottom composition x
- Bottom flow B is used to control bottom liquid level
- Generally, severe interactions occur between the top and bottom composition controllers; relatively weak interaction with the level controller
- Thus, pairings are considered for the compositions controllers
- Can we control the top composition using the steam flow, while the bottom composition using the reflux flow? *Why ?*

Coupling Effect of Loop 2 on y_1



- Let say the setpoint of G_{c1} is changed.
- G_{c1} will adjust the manipulated variable c_1
- Change in c_1 will affect y_1 via G'_{11} (main effect) and y_2 via G_{21} (coupling effect)
- Loop 2 will consider the change as disturbance and reacts by adjusting manipulated variable c_2
- Change in c_2 will affect y_2 via G'_{22} and y_1 via G'_{12}
- Loop 1 will consider the change in c_2 as disturbance and reacts by adjusting c_1, which will further affect both y_1 and y_2
- The cycle continues until both y_1 and y_2 settle at their setpoints – assuming no further external change occur (e.g., no setpoint change, or no external disturbance)

Multi-Loop Controllers – Distillation Column



- **L** - maintain composition of **D** - **<u>causes changes</u>** - composition of **<u>B</u>**.

- Bottom loop reacts and changes the **steam** flow rate to bring the bottom composition to its setpoint
  - To correct the effect - **<u>reflux changes</u>** - causes changes – **<u>composition of D moves to its setpoint.</u>**

Stead-State Coupling – Relative Gain Array (RGA)

$$RGA = \begin{pmatrix} \lambda_{11} & \lambda_{12} \\ \lambda_{21} & \lambda_{22} \end{pmatrix}$$

$$\lambda_{11} = \lambda_{22} \qquad \lambda_{11} + \lambda_{12} = 1 \qquad \lambda_{21} + \lambda_{22} = 1$$

Therefore, only one element requires evaluation:

$$\lambda_{11} = \frac{\left(\dfrac{\Delta y_1}{\Delta c_1}\right)_{c_2}}{\left(\dfrac{\Delta y_1}{\Delta c_1}\right)_{y_2}}$$

Change of y1 over change of c1, when the second loop is opened. This is called main effect.

Change of y1 over change of c1, when the second loop is closed. This is main effect plus coupling effect.

- Relative Gain Array (RGA) analysis is often used to evaluate process interaction
- For 2x2 MIMO, the RGA diagonal elements are the same $\lambda_{11} = \lambda_{22}$.
- Note that for 3x3, 4x4 and larger MIMO systems, these diagonal elements often take different values.
- What RGA really means? See the equation for explanation

Relative Gain Array – Coupling Effect

- $\lambda_{11} = 1$: => **no coupling** exists.
- $\lambda_{11} > 1$: => coupling effect in the **opposite direction** to the primary effect.
- $\lambda_{11} < 1$: => coupling effect in the **same direction** as the primary effect.
- $\lambda_{11} = 0.5$: **very severe coupling**.
- $0 < \lambda_{11} < 1$: **severity increases as λ_{11} decreases.**
- $\lambda_{11} > 1$: severity **increases as λ_{11} increases.**
- $\lambda_{11} < 0$: negative value indicates **very severe coupling** effects which can cause **closed-loop instability** and integrity issue.

RGA Calculation

- Given a 2x2 transfer function matrix

$$G = \begin{bmatrix} \frac{k_{11}e^{-\theta_{11}s}}{\tau_{11}s+1} & \frac{k_{12}e^{-\theta_{12}s}}{\tau_{12}s+1} \\ \frac{k_{21}e^{-\theta_{21}s}}{\tau_{21}s+1} & \frac{k_{22}e^{-\theta_{22}s}}{\tau_{22}s+1} \end{bmatrix}$$

- Steady-state gain matrix

$$K = \begin{bmatrix} k_{11} & k_{12} \\ k_{21} & k_{22} \end{bmatrix}$$

- RGA

$$\Lambda = \begin{bmatrix} \lambda_{11} & \lambda_{12} \\ \lambda_{21} & \lambda_{22} \end{bmatrix}$$

- Diagonal element can be calculated as

$$\lambda_{11} = \frac{k_{11}}{\underbrace{k_{11}}_{\substack{main \\ effect}} - \underbrace{\frac{k_{12}k_{21}}{k_{22}}}_{\substack{coupling \\ effect}}} = \frac{1}{1 - \frac{k_{12}k_{21}}{k_{11}k_{22}}}$$

- For MIMO system greater than 2, we can use the following Matlab Command to calculate the RGA:

```
>> RGA = K.*inv(K)’
```

Example of RGA calculation and pairings – 2x2 MIMO

- Consider a process

$$G = \begin{bmatrix} \frac{1.7e^{-2s}}{10s+1} & \frac{1.1e^{-3s}}{15s+1} \\ \frac{-1.2e^{-2.5s}}{11s+1} & \frac{1.4e^{-s}}{8s+1} \end{bmatrix}$$

$$K = \begin{bmatrix} 1.7 & 1.1 \\ -1.2 & 1.4 \end{bmatrix}$$

$$\lambda_{11} = \frac{1}{1 - \frac{k_{12}k_{21}}{k_{11}k_{22}}} = \frac{1}{1 - \frac{(1.1)(-1.2)}{(1.7)(1.4)}}$$

$$= 0.6432$$

$$\Lambda = \begin{bmatrix} 0.6432 & 0.3568 \\ 0.3568 & 0.6432 \end{bmatrix}$$

- Important criteria of pairing based on RGA:

1. Choose pairing with RGA element closest to unity
2. Do not choose pairing with RGA element is negative

- If pairing is with negative RGA element, then this leads to integrity issue where if one of the loop fails, another loop will become unstable.
- Taking consideration of two criteria above: **choose $\boldsymbol{u_1 \sim y_1 / u_2 \sim y_2}$ pairings.**

Example of RGA calculation and pairing – 3x3 MIMO

- Let the steady-state gain matrix

$$K = \begin{bmatrix} 0.7 & 0.3 & -0.4 \\ 1.1 & -0.6 & -0.2 \\ 0.2 & 0.5 & -0.9 \end{bmatrix}$$

- Type in Matlab Command Window

```
>> K = [0.7  0.3  -0.4; 1.1  -0.6  -0.2; 0.2  0.5  -0.9];

>> RGA = K.*inv(K)’
```

- RGA is

$$\Lambda = \begin{bmatrix} 0.9634 & 0.6129 & -0.5763 \\ 0.1656 & 0.7097 & 0.1247 \\ -0.1290 & -0.3226 & 1.4516 \end{bmatrix}$$

- Based on the RGA elements – for the first row, 0.9634 is the closest to unity, therefore, choose u_1 to control y_1.
- For the second row, 0.7097 is the closest to unity, and this input u_2 is not yet used – so use u_2 to control y_2.
- Finally, for the third row, the only input left is u_3 and its RGA element is positive – so use u_3 to control y_3.
- Controller pairings are: $u_1 \sim y_1 / u_2 \sim y_2 / u_3 \sim y_3$

RGA Analysis

- RGA – good measure for:
  1. Steady-state coupling effect of a configuration
  2. Input/output relationships have the same **general dynamic behaviors**.
- However, RGA can be **misleading** if the transfer function dynamics are significantly different.
- **RGA is a Steady-State measure of process interaction.**

Example of Impact of Different Dynamic Behaviors

$$\mathbf{G}_{11}(\mathbf{s}) = \frac{1.0}{100\mathbf{s}+1} \qquad \mathbf{G}_{12}(\mathbf{s}) = \frac{0.3}{10\mathbf{s}+1}$$

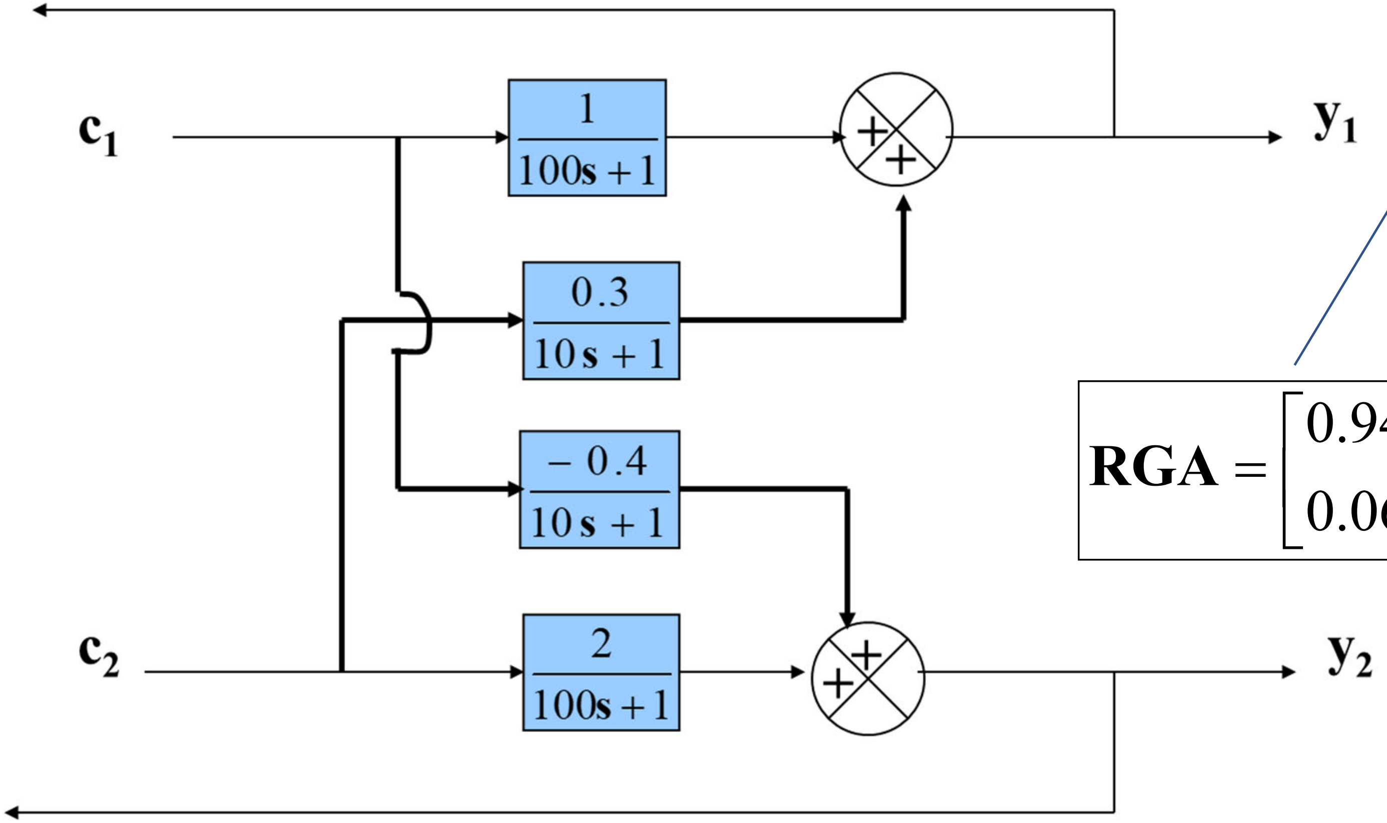
$$\mathbf{G}_{21}(\mathbf{s}) = \frac{-0.4}{10\mathbf{s}+1} \qquad \mathbf{G}_{22}(\mathbf{s}) = \frac{2.0}{100\mathbf{s}+1}$$

$$\boxed{\begin{pmatrix} \mathbf{K}_{11} & \mathbf{K}_{12} \\ \mathbf{K}_{21} & \mathbf{K}_{22} \end{pmatrix} = \begin{pmatrix} 1.0 & 0.3 \\ -0.4 & 2.0 \end{pmatrix}}$$

$$\boxed{\lambda_{11} = \frac{1}{1 - \dfrac{\mathbf{K}_{12}\,\mathbf{K}_{21}}{\mathbf{K}_{11}\,\mathbf{K}_{22}}}}$$

$$\textbf{\textit{Steady State}}\ \mathbf{RGA}, \lambda_{11} = \frac{1}{1 - \dfrac{-0.4(0.3)}{1(2)}} = 0.94$$

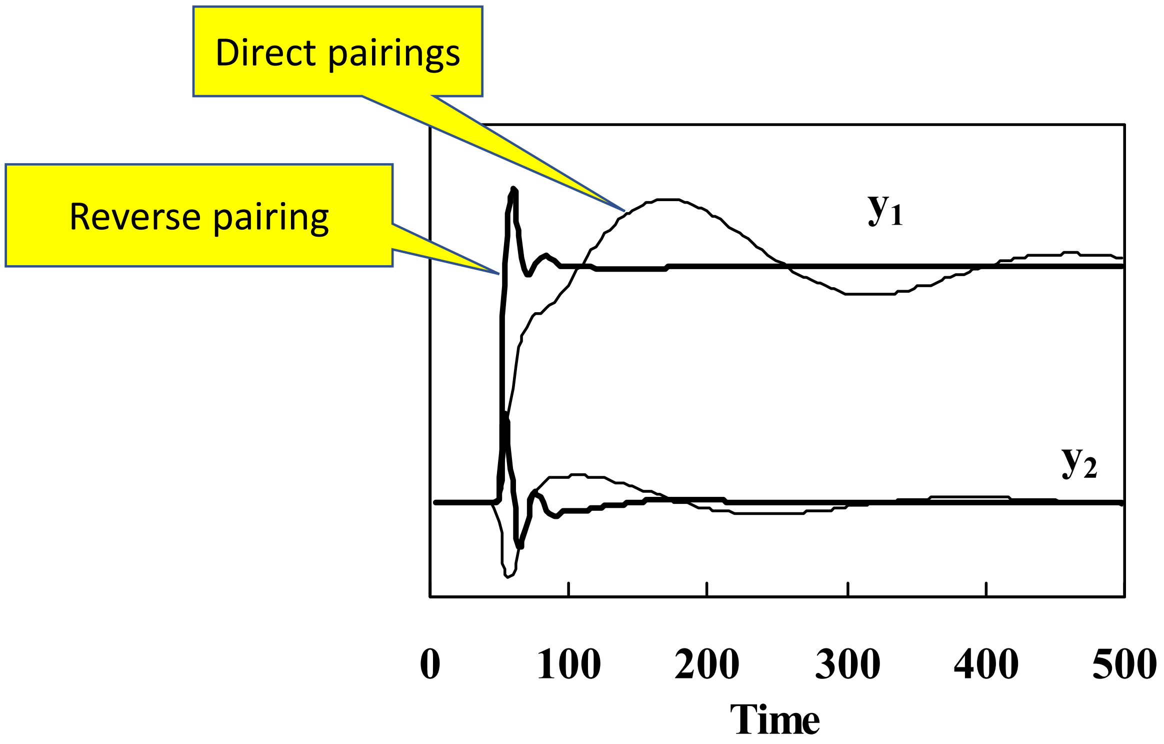
$$\boxed{\mathbf{RGA} = \begin{bmatrix} 0.94 & 0.06 \\ 0.06 & 0.94 \end{bmatrix}}$$



$$\textbf{RGA} = \begin{bmatrix} 0.94 & 0.06 \\ 0.06 & 0.94 \end{bmatrix}$$

RGA suggests direct pairings: $u_1 \sim y_1 / u_2 \sim y_2$

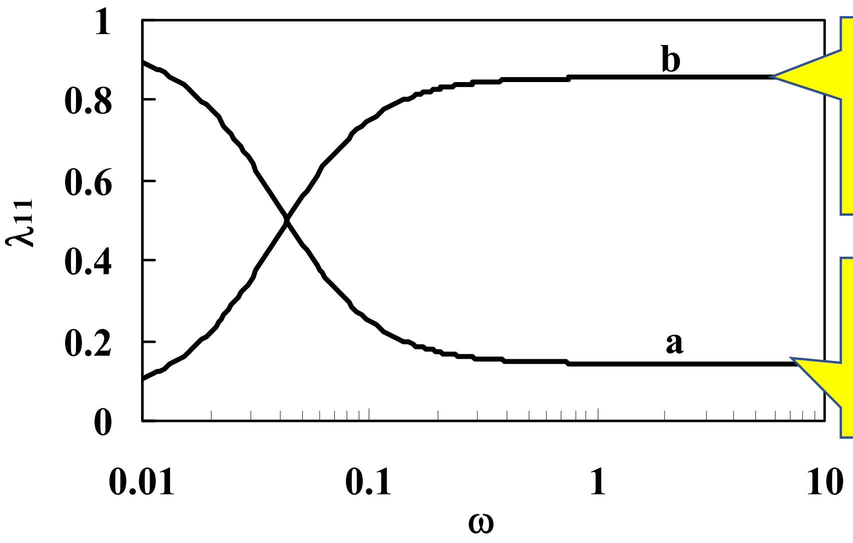
Direct Pairing (Thin Line) and Reverse Pairing (Thick Line)



- Figure shows that reverse pairings perform better than direct pairings ($u_1 \sim y_2 / u_2 \sim y_1$)
- Obviously, the steady-state RGA gives ***misleading*** suggestion. **Why?**
- Notice the transfer function dynamics are **significantly different** from each other.

Dynamic RGA

- When the transfer function dynamics vary significantly from each other, it is recommended to use Dynamic RGA instead of the steady-state RGA
- Dynamic RGA measures the process interactions as the frequency ω varies.



Reverse pairings – low interactions at higher frequency

Direct pairings – high interactions at higher frequency

$$\lambda_{11}(\omega) = \frac{1}{1 - \frac{|G_{12}(i\omega)|\,|G_{21}(i\omega)|}{|G_{11}(i\omega)|\,|G_{22}(i\omega)|}}$$

For a first order process:

$$|G(i\omega)| = \frac{K_p}{\sqrt{\tau_p^2\,\omega^2 + 1}}$$

For this example:

$$\lambda_{11} = \frac{1}{1 + \frac{16.7(10^2\omega^2 + 1)}{100^2\omega^2 + 1}}$$

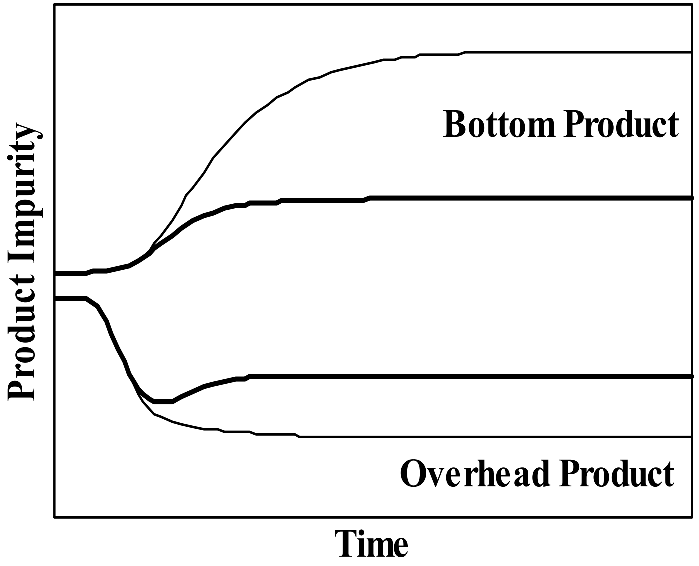
Use reverse pairing if operating at high frequency but use direct pairings if operating at low frequency.

Pairing Considerations

1. Choose pairing between manipulated and controlled variables, which results in the **least process interactions** – use steady-state RGA for comparable dynamics, or use dynamic RGA for dissimilar dynamics
2. Choose pairing between manipulated and controlled variables, which leads to **quick response** of controlled variable to manipulated variable (fast dynamic criterion)
   - E.g., $\frac{Y_1}{U_1} = \frac{2\exp(-s)}{10s+1}, \frac{Y_1}{U_2} = \frac{2\exp(-3s)}{12s+1}$; choose $U_1 \sim Y_1$ pairing because U_1 has faster dynamic than U_2; furthermore the input U_2 has longer deadtime which is not desirable as long deadtime imposes lower achievable control performance – sluggish response.
3. Choose pairing between manipulated and control variables, which is **least sensitive to disturbance**

Sensitivity to Disturbances

- Each configuration has a **different sensitivity to a disturbance**.
- Thick and thin line represent the results of different configurations.
- Notice that, the configuration with thick line is less sensitive to disturbance than the one with thin line.
- The one that is less sensitive to disturbance is a more efficient configuration (or pairings).
- Less sensitive to disturbance is also good because it could lead to lower control action required.



Example - Configuration Selection for a C3 Splitter

Configuration	RGA(λ_{11})
(L, B)	0.94
(L, V)	25.3
$(L/D, V/B)$	1.70
(D, V)	0.06

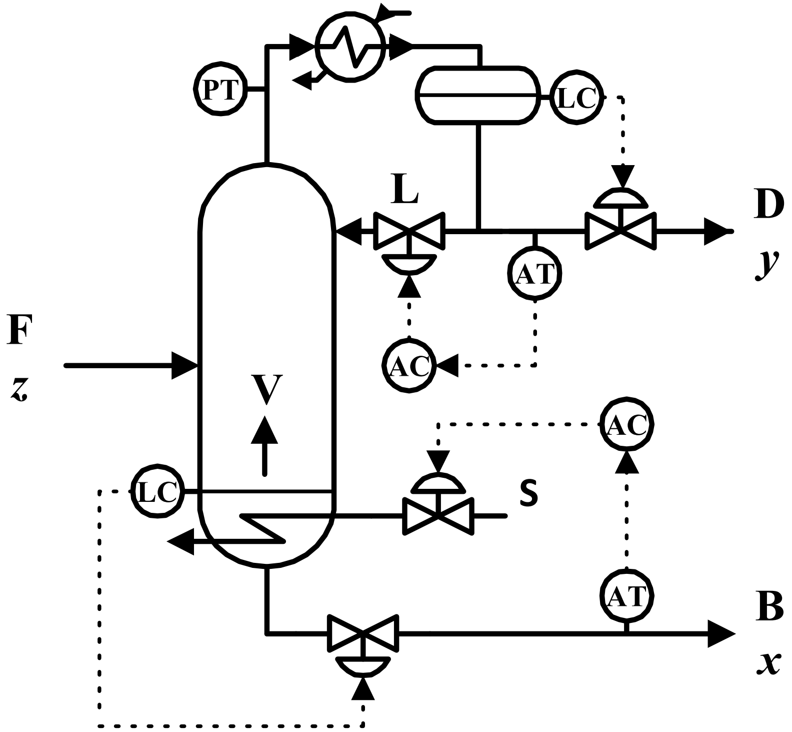
Least interactions

Severe interactions – coupling effect opposite to main effect

Mild interactions

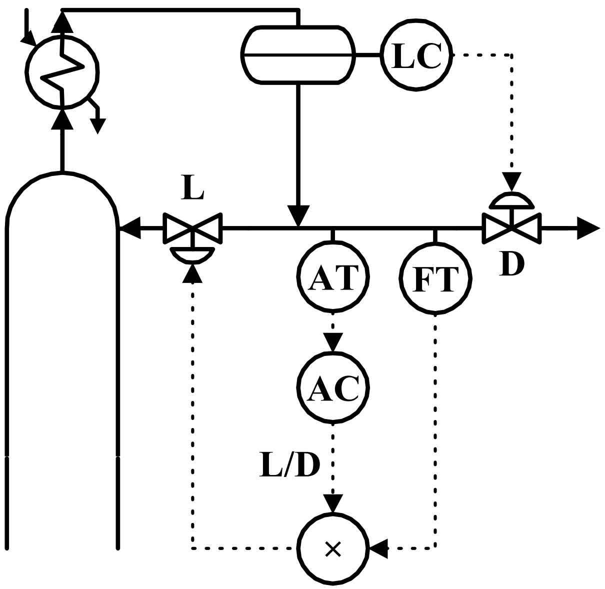
Mild interactions – coupling effect in the same direction as main effect Main effect is weaker than coupling effect

(L,V) Configuration Applied to the C3 Splitter



- Reflux flow L used to control top composition
- Steam flow S is used to control bottom composition
- Steam flow directly affects the vapor flow V in the column, so this means V is used to control the bottom composition
- Hence known as (L,V) configuration

Reflux Ratio Applied to the Overhead of the C3 Splitter



- Ratio controller is used where the wild stream is the distillate flow, while the reflux stream is the manipulated flow
- Thus, reflux ratio (L/D) is used as manipulated variable by the top composition controller
- For the bottom composition controller, the boilup ratio (V/B) can be used as manipulated variable, i.e. bottom flow B is wild stream while steam flow S is the manipulated stream (S is directly related to vapor flow V)
- Thus, this is called (L/D, V/B) configuration

Other Configurations

- (L,B) configuration
  - Reflux flow L is used to control top composition
  - Bottom flow B is used to control bottom composition
- (D,V) configuration
  - Distillate flow D is used to control top composition
  - Steam flow (related directly to V) is used to control bottom composition

Configuration Selection of C3 splitter

- **L, L/D** and **V** are the **least sensitive** to feed composition disturbances.
- **L and V** have the **most immediate effect** on the product compositions followed by
- **L/D and V/B** and in between,
- **D and B** yielding the **slowest** response.
- Therefore, each configuration involves conflicting factors, e.g., (L,V) is the least sensitive to disturbance and has fast dynamic response, BUT it exhibits the **most severe process interaction (**$\lambda_{11} = 25.3$)
- We need to simulate the configurations to evaluate their performances

Control Performance

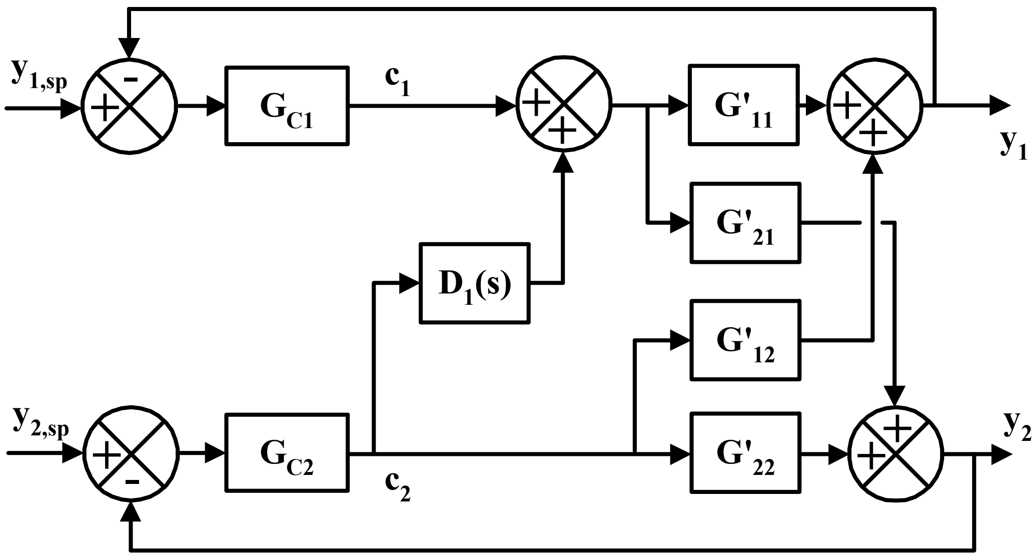
Configuration	IAE for Overhead	IAE for Bottoms
(L,B)	0.067	1.49
(L,V)	0.250	13.3
$(L/D,V/B)$	0.095	2.00
(D,V)	0.098	1.91

Smallest IAE – best top controller Configuration with least interactions

Analysis of Configuration Selection Example

- Note that **(L,V) is the worst configuration**
  - Although it is the **least susceptible to disturbances** and the **fastest acting configuration**, but it is the **most coupled** ($\boldsymbol{\lambda_{11} = 25.3}$)
- Although (D,V) has an RGA of 0.06, it shows decent control performance.
  - D has slow dynamic, but V has fast dynamic – if both have fast or slow dynamics, this configuration might show bad performance
- **(L,B) is the best** since it has
  - **Good decoupling** and the overhead product is most important.

One-Way Decoupler



$$D_1(s) = \frac{-G'_{12}(s)}{G'_{11}(s)}$$

- Decoupler can be used to **reduce the coupling effect** between two loops
- One-way decoupler often used
- Two-way decouplers might be use for very severe interactions but might lead to **very high control action** required
- High control action may lead to **higher rate of wear and tear** of control valve

Multi-Loop (Decentralized) PID Controller Design

- There are 6 broad categories of methods to design multi-loop PID control system as follows:
  1) **Detuning**
  2) **Sequential loop closing**
  3) **Independent tuning**
  4) **Simultaneous tuning**
  5) **Optimization**
  6) **Relay auto-tuning**

1) Detuning Method

- Step 1: An individual controller is tuned according to an existing single-input and single-output tuning formula, e.g., classical Ziegler-Nichols and Skogestad IMC.
    - E.g., consider 2 multi-loop PID controllers

$$G_{c_1} = K_{c_1}\left(1 + \frac{1}{\tau_{I_1} s} + \tau_{D_1} s\right), \qquad G_{c_2} = K_{c_2}\left(1 + \frac{1}{\tau_{I_2} s} + \tau_{D_2} s\right)$$

- Step 2: Detune each controller by a factor F
    - Detuned controllers

$$G'_{c_1} = \frac{K_{c_1}}{F}\left(1 + \frac{1}{\tau_{I_1} s} + \tau_{D_1} s\right), \qquad G'_{c_2} = \frac{K_{c_2}}{F}\left(1 + \frac{1}{\tau_{I_2} s} + \tau_{D_2} s\right)$$

- Step 3: Evaluate the closed-loop responses – if not satisfied then readjust F

Example 2x2 MIMO – Wood and Berry (WB) Column

- Wood and Berry Column is represented by transfer function matrix

$$G = \begin{bmatrix} \frac{12.8\exp(-s)}{16.7s+1} & \frac{-18.9\exp(-3s)}{21s+1} \\ \frac{6.6\exp(-7s)}{10.9s+1} & \frac{-19.4\exp(-3s)}{14.4s+1} \end{bmatrix}$$

- RGA Analysis

$$\Lambda = \begin{bmatrix} 2.0094 & -1.0094 \\ -1.0094 & 2.0094 \end{bmatrix}$$

- Recommended pairings are $U_1 \sim Y_1$ and $U_2 \sim Y_2$ (direct pairings)
- Direct pairings ensure RGA elements involved are positive – always avoid negative RGA pairings

Multi-loop PID Controllers for WB Column – Detuning Method

- Let us apply Ziegler-Nichols tuning (Matlab Control System Designer) to design 2 PID controllers
- Design PID 1 based on $g_{11} = \frac{12.8\exp(-s)}{16.7s+1}$

$$G_{c_1} = 1.2895\left(1 + \frac{1}{2s} + 0.4602s\right)$$

$$GM = 5.02dB, PM = 34.9\ deg$$

- Design PID 2 based on $g_{22} = \frac{-19.4\exp(-3s)}{14.4s+1}$

$$G_{c_2} = -0.2548\left(1 + \frac{1}{5.6s} + 1.4s\right)$$

$$GM = 4.88\ dB, PM = 40.1\ deg$$

- If $F = 2$

$$G_{c_1} = 0.6448\left(1 + \frac{1}{2s} + 0.4602s\right)$$

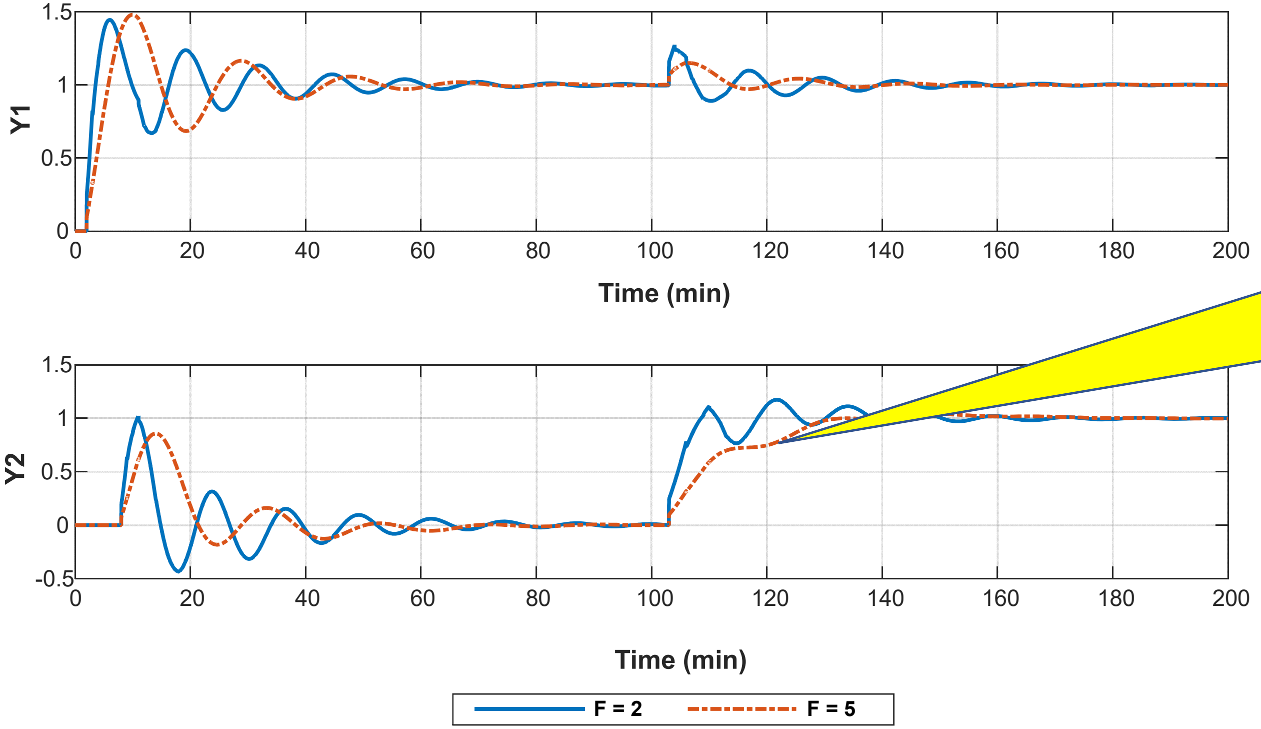
$$G_{c_2} = -0.1274\left(1 + \frac{1}{5.6s} + 1.4s\right)$$

- If $F = 5$

$$G_{c_1} = 0.2579\left(1 + \frac{1}{2s} + 0.4602s\right)$$

$$G_{c_2} = -0.0510\left(1 + \frac{1}{5.6s} + 1.4s\right)$$

Detuning method example



Increase in detuning factor F, reduces the oscillation, but more sluggish response

2) Sequential Loop Closing

- Step 1: Choose the fast loop first over the slower one.
- Step 2: Design the controller based on the faster transfer function (loop)
- Step 3: Close the fast loop (controller is activated)
- Step 4: Find the linearized model for the slower loop with the fast loop already closed.
- Step 5: Design the second controller using the linearized model obtained in Step 4.
- Step 6: Close the second (slower loop)

Repeat Step 4 to 6 for the remaining loops.

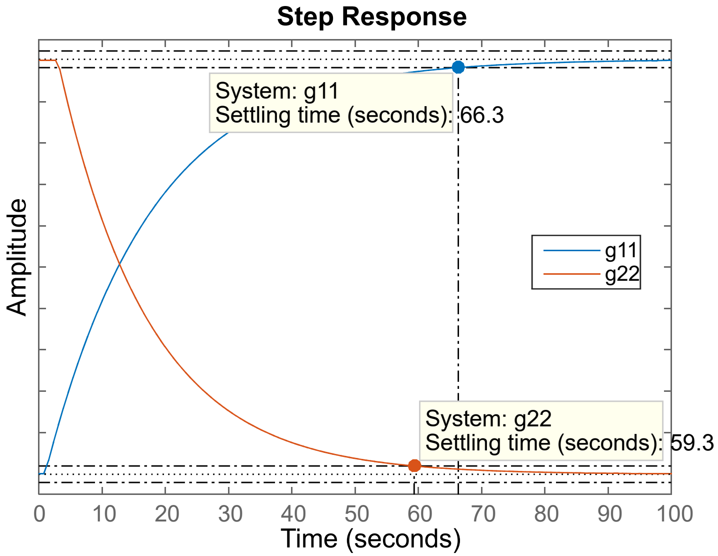
The order of loop closing is important as it substantially affects the overall control performance.

Multi-loop PID Controllers for WB Column – Sequential Loop Closing Method

- Choose which loop is the fastest – check the open-loop step responses of g_{11} and g_{22}. Use Matlab Command:

```
>> s = tf('s');
>>g11 = 12.8*exp(-s)/(16.7*s + 1);
>>g22 = -19.4*exp(-3*s)/(14.4*s + 1);
>> step(g11,g22);
```

From the step responses, we notice that g22 has shorter settling time – loop 2 is faster than loop 1



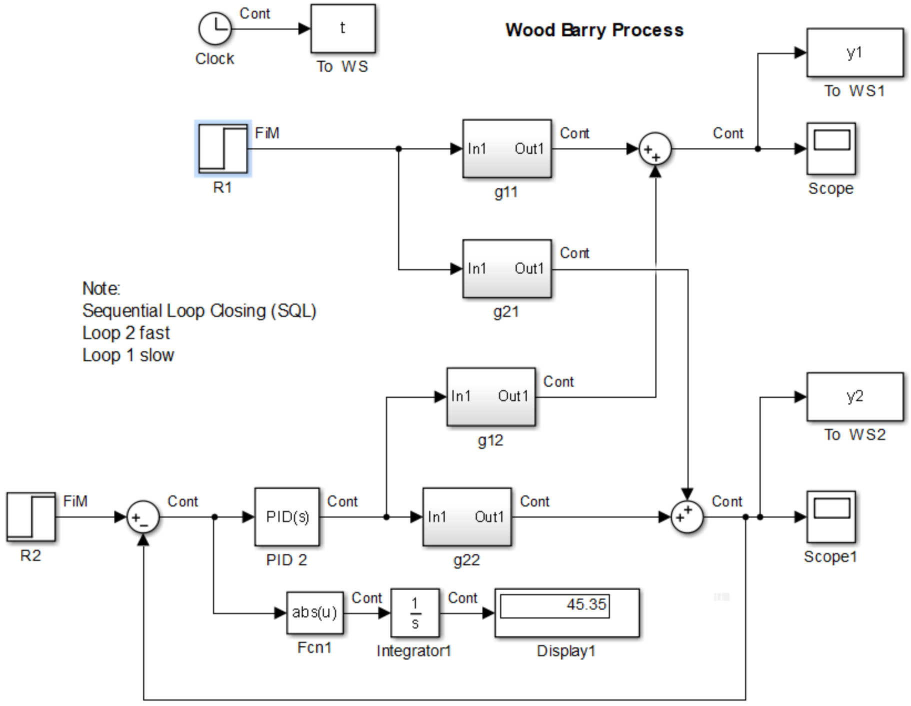
Sequential Loop Closing example

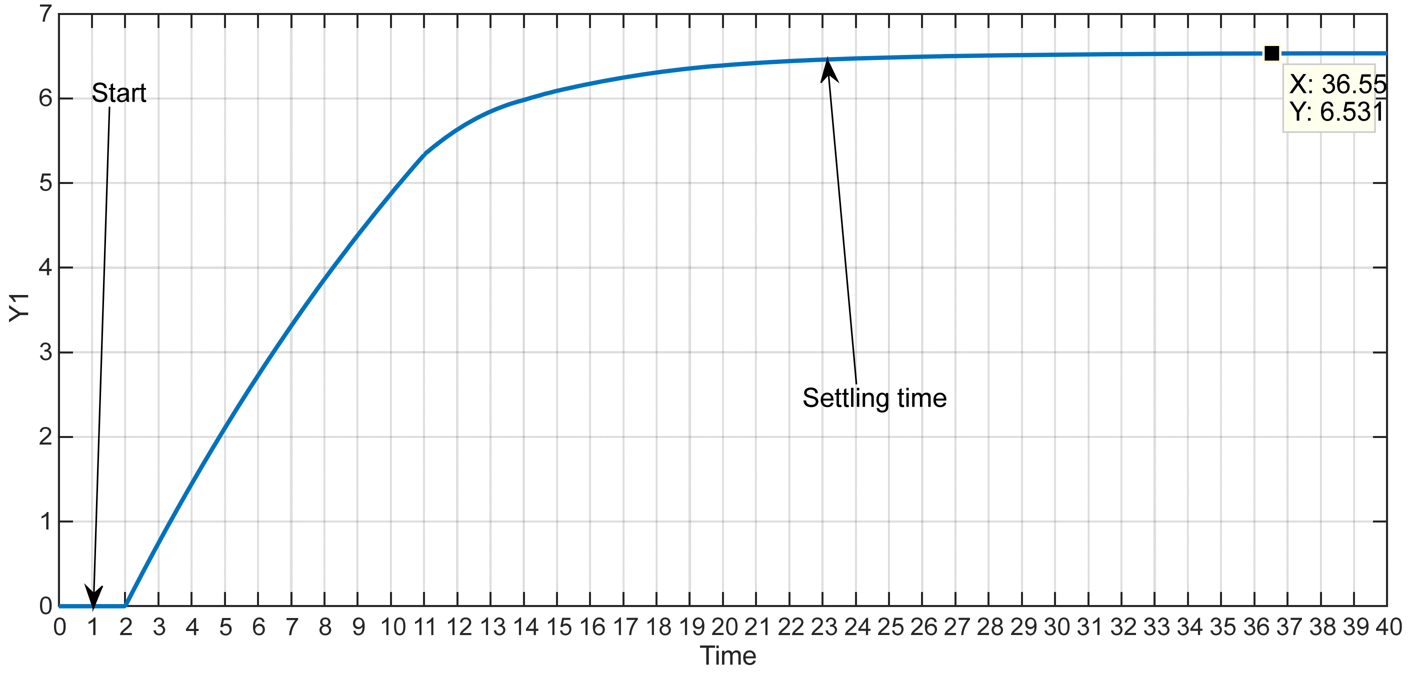
- Let us use Skogestad IMC (SIMC) tuning in Matlab Control System Designer
- Design PID 2 using g_{22}

$$G_{c_2} = -0.1423\left(1 + \frac{1}{16.5s} + 1.3636s\right)$$

$$GM = 9.71\ dB, PM = 74.8\ deg$$

- Apply 1 unit step change in R1 and then plot t against y1
- From this step response plot, obtain the FOPDT parameters – see next slide





- Delay $\theta = 1$
- Time constant $\tau_p = \frac{23 - 2}{4} = 5.25$
- Gain $K_p = 6.53$

- Linearized model for loop 1

$$\boldsymbol{g'_{11} = \frac{6.53\exp(-s)}{5.25s + 1}}$$

Sequential Loop Closing (SQL) example…

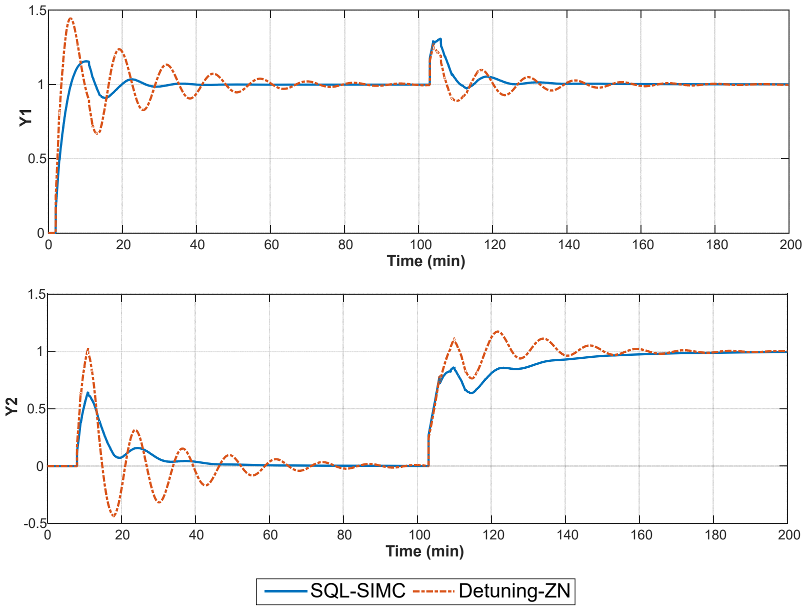
- Design PID1 using $\boldsymbol{g'_{11}} = \frac{\mathbf{6.53exp(-s)}}{\mathbf{5.25s+1}}$

$$G_{c_1} = 0.4405\left(1 + \frac{1}{5.7s} + 0.456s\right)$$

$$GM = 9.92\ dB, PM = 74.6\ deg$$

- Figure shows the comparison between the SQL tuning and previous Detuning method.

- SQL demonstrates substantial performance improvement over Detuning method.



3) Independent Tuning Method

- Independent tuning method relies on **Effective Open-Loop Transfer Function (EOTF)**
- Consider a 2x2 MIMO transfer function matrix

$$G = \begin{bmatrix} g_{11} & g_{12} \\ g_{21} & g_{22} \end{bmatrix}$$

- Reduce to decentralized form

$$G = \begin{bmatrix} g_{e11} & 0 \\ 0 & g_{e22} \end{bmatrix}$$

- EOTFs

$$g_{e11} = g_{11} - \frac{g_{12}g_{21}}{g_{22}}$$

$$g_{e22} = g_{22} - \frac{g_{12}g_{21}}{g_{11}}$$

- The controllers G_{c_1} and G_{c_2} are independently designed based on the EOTFs g_{e11} and g_{e22} respectively.
- Existing SISO tuning formulas can be used.
- Let us illustrate the application of Independent tuning in the next slides for the wood and Barry column example

Multi-loop PID Controllers for WB Column – Independent Tuning Method

- Recall WB column

$$G = \begin{bmatrix} \dfrac{12.8\exp(-s)}{16.7s+1} & \dfrac{-18.9\exp(-3s)}{21s+1} \\ \dfrac{6.6\exp(-7s)}{10.9s+1} & \dfrac{-19.4\exp(-3s)}{14.4s+1} \end{bmatrix}$$

- EOTFs

$$g_{e11} = \frac{12.8\exp(-s)}{16.7s+1} - \left\{ \frac{\left[\dfrac{-18.9\exp(-3s)}{21s+1}\right]\left[\dfrac{6.6\exp(-7s)}{10.9s+1}\right]}{\dfrac{-19.4\exp(-3s)}{14.4s+1}} \right\}$$

$$g_{e11} = \underbrace{\frac{12.8\exp(-s)}{16.7s+1}}_{\substack{main\ transfer \\ function}} - \underbrace{\frac{6.43(14.4s+1)e^{-7s}}{(21s+1)(10.9s+1)}}_{coupling\ transfer\ function}$$

- Approximation required to simplify the EOTF
- Let us equalize the delay of both main and coupling transfer functions:
- Assume a general formula as follows

$$\boldsymbol{g = \frac{k_p exp(-(\theta_1+\theta_2)s)}{(\tau_1 s+1)(\tau_2 s+1)} \cong \frac{k_p exp(-\theta_1 s)}{(\tau_1 s+1)(\tau_2 s+1)(\theta_2 s+1)}}$$

- Applying the formula to the coupling transfer function

$$g_{i11} = \frac{6.43(14.4s+1)e^{-7s}}{(21s+1)(10.9s+1)} \cong \frac{6.43(14.4s+1)e^{-s}}{(21s+1)(10.9s+1)(6s+1)}$$

Independent Tuning Method example

- Overall EOTF now becomes

$$g_{e11} = \left[\frac{12.8}{16.7s+1} - \frac{6.43(14.4s+1)}{(21s+1)(10.9s+1)(6s+1)}\right] e^{-s}$$

- The above transfer function can be used in Matlab Control System Designer – only single delay term
- The second EOTF can be derived the same way

$$g_{e22} = \frac{-19.4e^{-3s}}{14.4s+1} - \left\{\frac{\left[\frac{-18.9e^{-3s}}{21s+1}\right]\left[\frac{6.6e^{-7s}}{10.9s+1}\right]}{\frac{12.8e^{-s}}{16.7s+1}}\right\}$$

$$g_{e22} = \left[\frac{-19.4}{14.4s+1} + \frac{9.745(16.7s+1)}{(21s+1)(10.9*s+1)(6s+1)}\right] e^{-3s}$$

- Let us apply once again the Skogestad IMC tuning
- Design G_{c_1} based on g_{e11}:

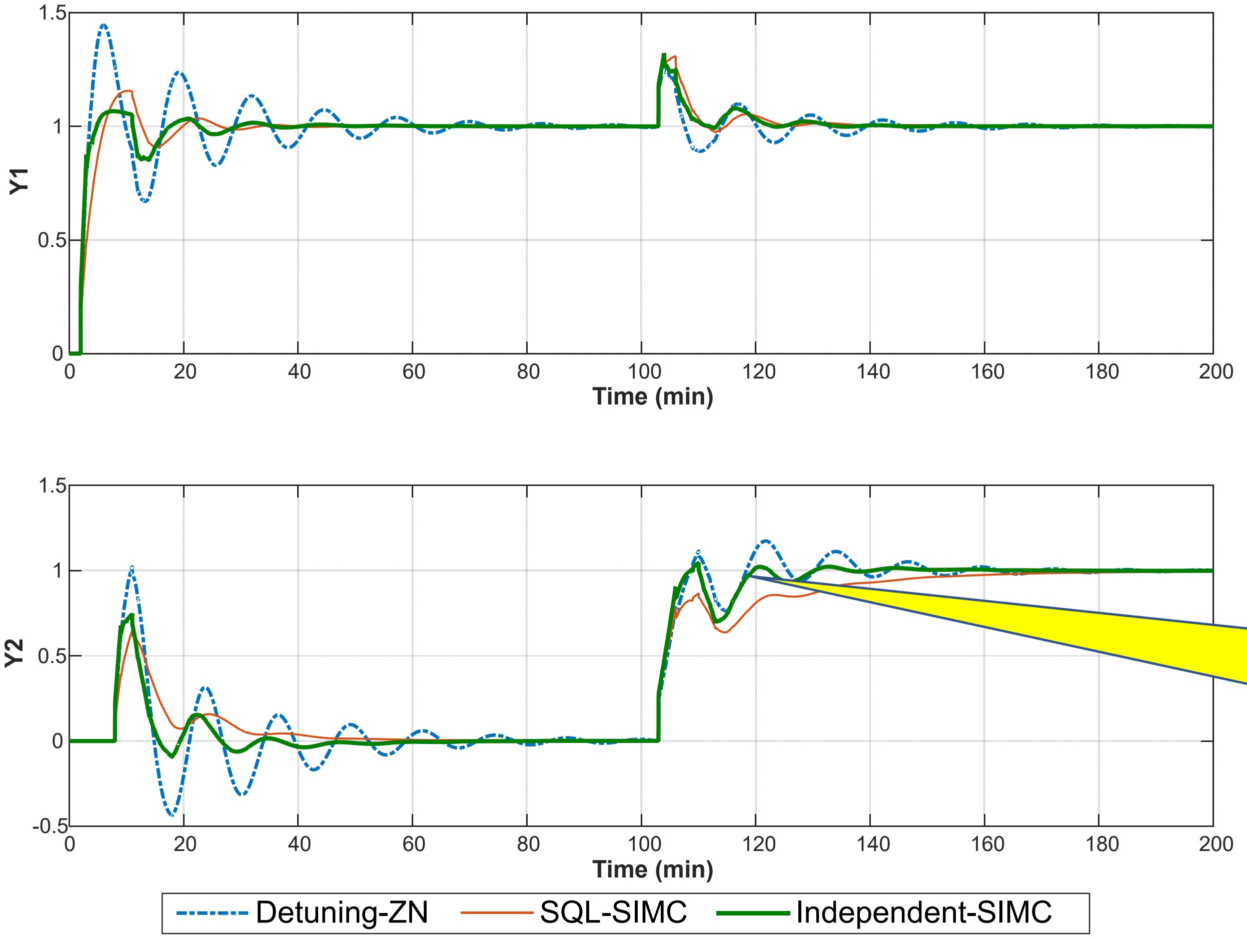
$$G_{c_1} = 0.775\left(1 + \frac{1}{8.5s} + 0.471s\right)$$

$$GM = 8.96\ dB, PM = 75.2\ deg$$

- Design G_{c_2} based on g_{e22}:

$$G_{c_2} = -0.16\left(1 + \frac{1}{9.3s} + 1.258s\right)$$

$$GM = 9.27\ dB, PM = 78.7\ deg$$



Independent tuning leads to substantial improvement over SQL, which outperforms Detuning

4) Simultaneous Dimensionless Parameter (SDP) Tuning

- A relatively new method which uses 3 dimensionless parameters: r_p, r_i and r_d
- These parameters take the following ranges: $0 < r_p < 1$, $r_i > 1$ and $r_d > 1$
- Details can be found in Mohd, N., & Nandong, J. (2018).
- Ideal PID Tuning formula is as follows

$$\tau_D = r_d\left(\frac{\theta}{2}\right), \qquad r_d > 1$$

$$K_c = \frac{r_p}{K_p}\left(\frac{\tau_p}{\tau_D}\right), \qquad 0 < r_p < 1$$

$$\tau_I = r_i \tau_{I_{min}}\left(\frac{\tau_p}{\theta}\right), \qquad r_i > 1$$

Where maximum lower limit $\tau_{I_{min}}$ on the reset time is given by

$$\tau_{I_{min}} = max\left\{\frac{\theta}{2}, \underline{I}_{ll}\right\}$$

$$\underline{I}_{ll} = \frac{K_L\theta}{2(1+K_L)}\left[1 + \frac{2(\tau_p - K_L\tau_D)}{\theta + 2\tau_p + K_L(2\tau_D - \theta)}\right]$$

Loop gain $K_L = K_c K_p$

The PID formula is programmed in Matlab:

[Kc,Ti,Td] = multiPItune(Kp,Tp,Dp,rp,ri,rd)

SDP Tuning Example – WB C

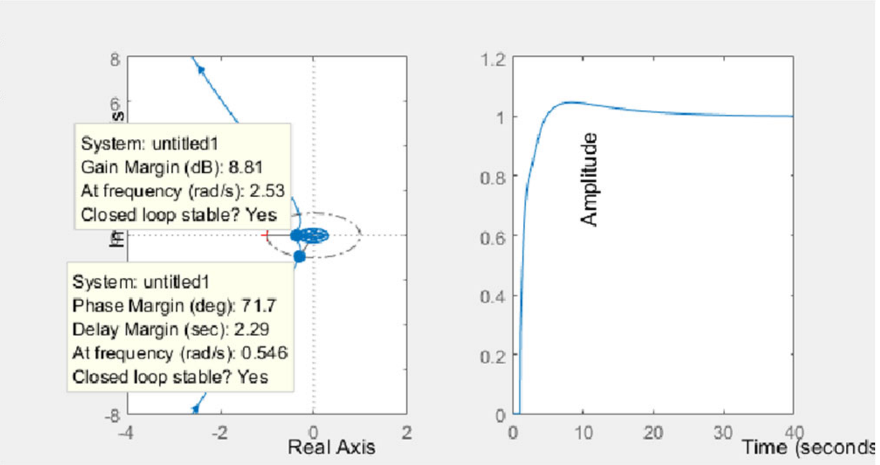
- Write on Command Window:

```
>> K = [12.8  -18.9; 6.6  -19.4];
>> T = [16.7  21;  10.9  14.4];
>>D = [1  3;  7  3];
```

- Let set the dimensionless parameters as follows: $r_p = 0.3, r_i = 1.2, r_d = 1.1$
- Type on Command Window:

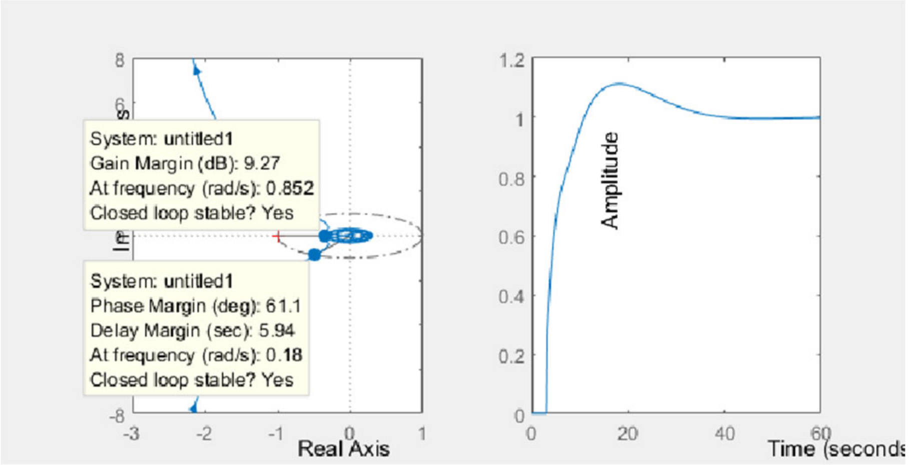
```
>> [Kc,Ti,Td] = multiPItune(K,T,D,rp,ri,rd);
```

- 2 figures – check the gain and phase margins are adequate (GM > 8 dB, PM > 60 deg)



System: untitled1
Gain Margin (dB): 8.81
At frequency (rad/s): 2.53
Closed loop stable? Yes

System: untitled1
Phase Margin (deg): 71.7
Delay Margin (sec): 2.29
At frequency (rad/s): 0.546
Closed loop stable? Yes



System: untitled1
Gain Margin (dB): 9.27
At frequency (rad/s): 0.852
Closed loop stable? Yes

System: untitled1
Phase Margin (deg): 61.1
Delay Margin (sec): 5.94
At frequency (rad/s): 0.18
Closed loop stable? Yes

SDP Tuning Results

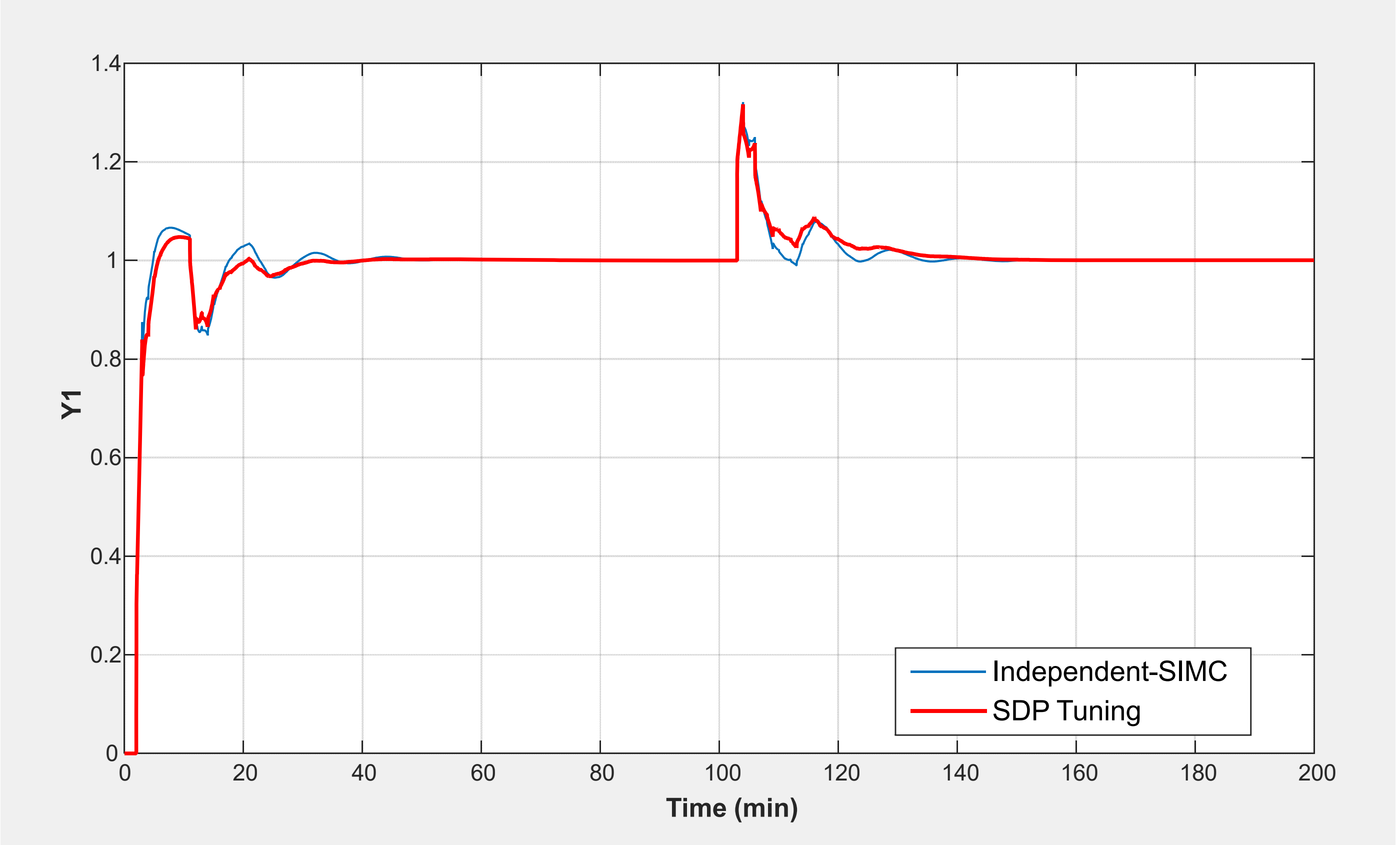
$$G_{c_1} = 0.7118\left(1 + \frac{1}{10s} + 0.55s\right)$$

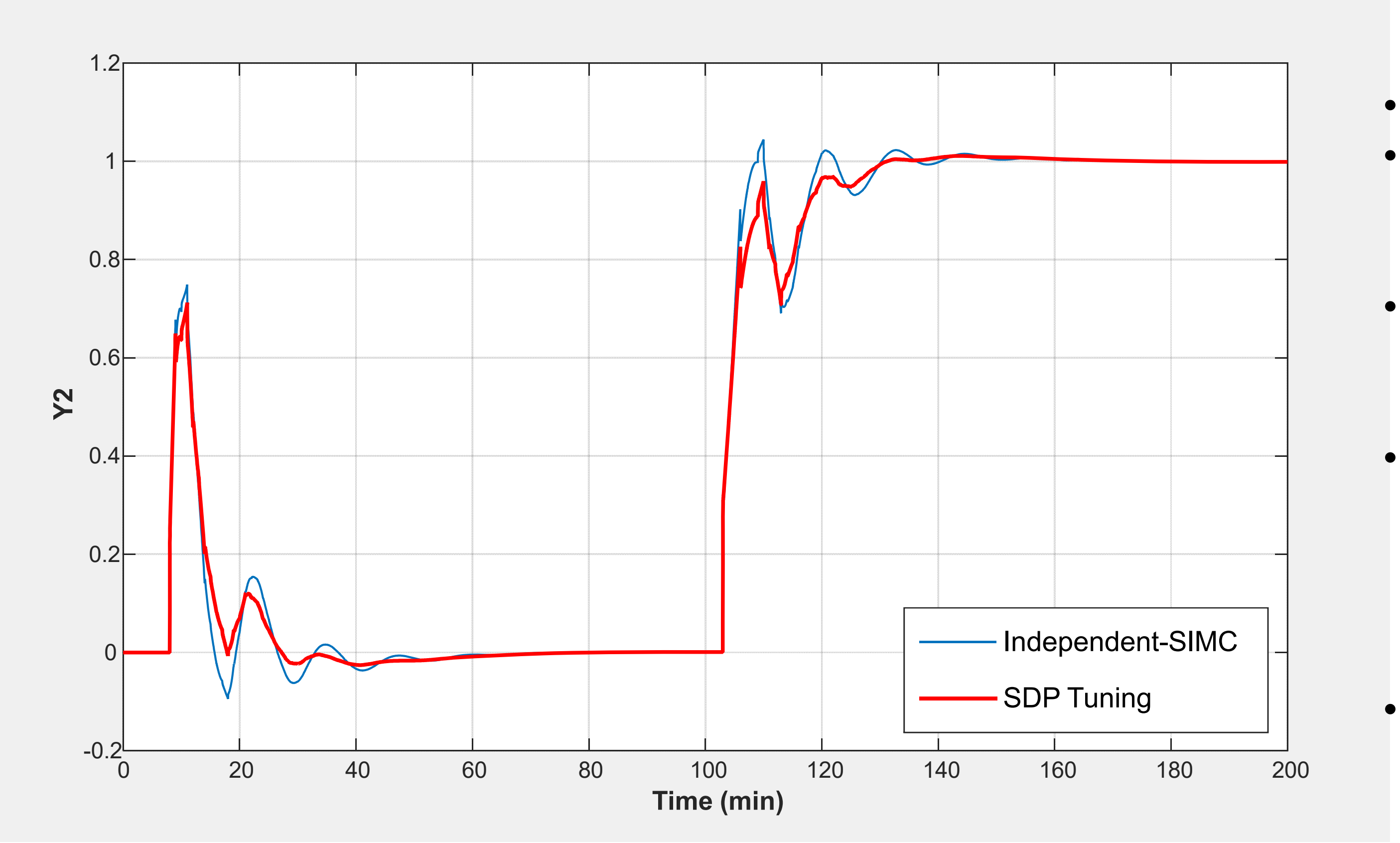
$$GM = 8.81\,dB, PM = 71.7\,deg$$

$$G_{c_2} = -0.135\left(1 + \frac{1}{8.64s} + 1.65s\right)$$

$$GM = 9.27\,dB, PM = 61.1\,deg$$

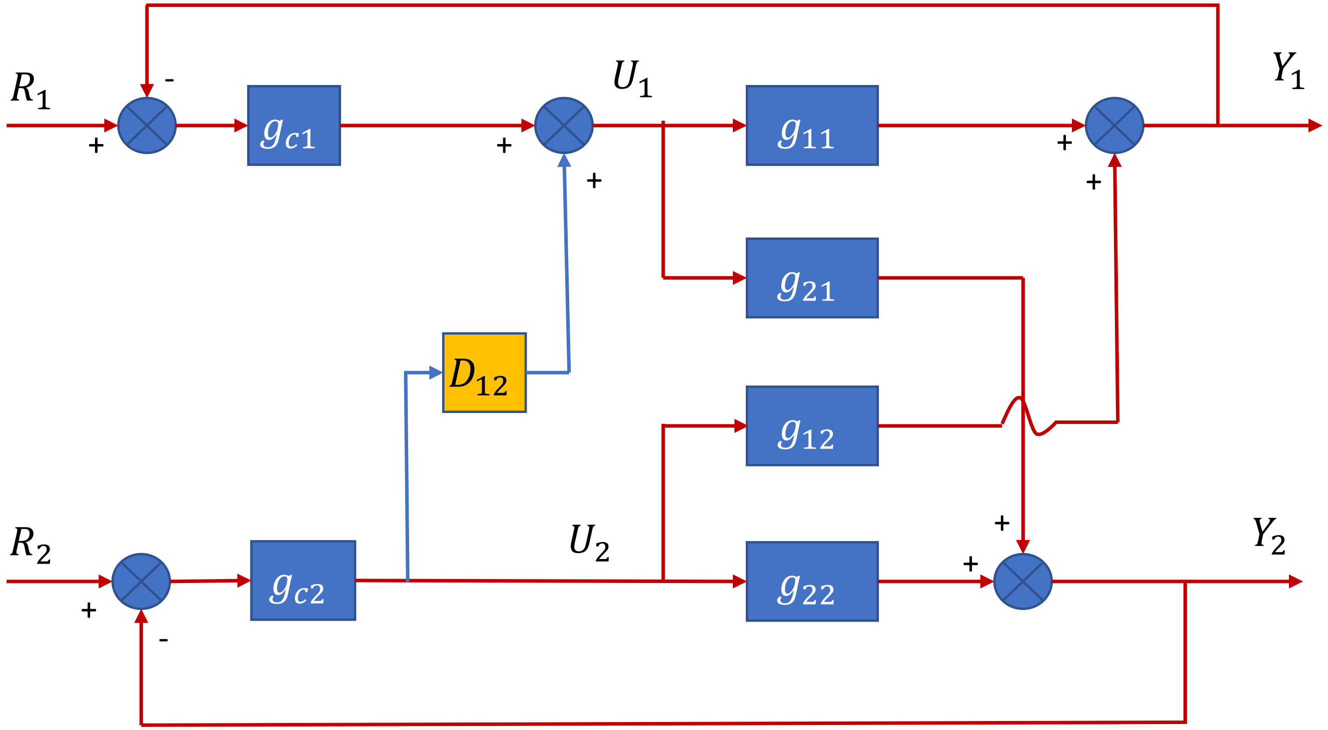
- Both Gain Margins and Phase Margins are adequate.
- Figure shows the comparison between SDP and Independent Tunings for Y1 response





- Wood & Barry Column
- Figure show response of Y2 for SDP and Independent Tunings
- SDP shows marked improvement over Independent Tuning
- Advantage of SDP – **one time tuning** for both control loops via the **3 common dimensionless** parameter values
- SDP **reduces the complexity** of multi-loop PID tuning tasks

Decoupling Control: One-Way Decoupling



- ✓ To remove the coupling effect in one direction only
- ✓ D_{12} decoupling the effect of control loop 2 on control loop 1
- ✓ Decoupler improves mainly only the control loop 1
- ✓ Decoupler can be viewed as a feedforward control
- ✓ Disturbance is the coupling effect from control loop 2

Decoupler Design

- Assume **perfect cancellation** of coupling effect from loop 2 to loop 1

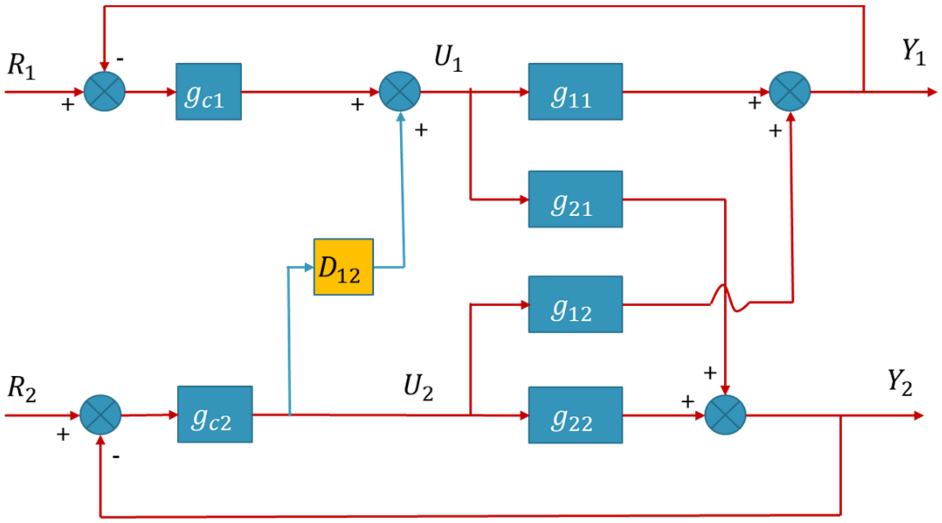
$$U_2 D_{12} g_{11} + U_2 g_{12} = 0$$

$$\therefore \ \boldsymbol{D_{12} = -\frac{g_{12}}{g_{11}}}$$

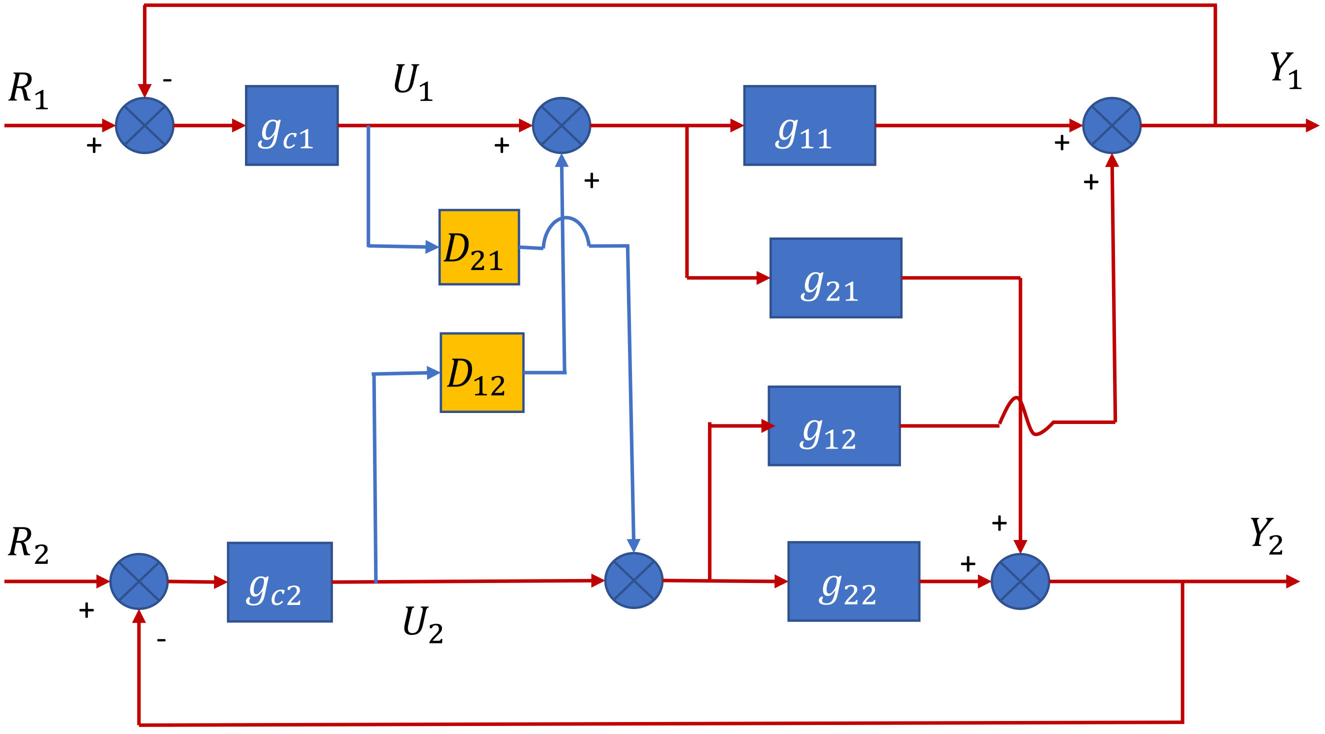
- General equation for decoupler D_{ij}

$$\therefore \ \boldsymbol{D_{ij} = -\frac{g_{ij}}{g_{ii}}}$$

- $\boldsymbol{D_{ij}}$ is to remove coupling effect from loop j to loop i



Two-Way Decoupling System



- ✓ D_{12} removes the coupling effect from loop 2 to loop 1
- ✓ D_{21} removes the coupling effect from loop 1 to loop 2
- ✓ This is a complete decoupling control system
- ✓ For $n \times n$ MIMO system, there will be $n(n-1)$ decouplers are required in a complete decoupling system
- ✓ Complete decoupling system may not be practical for a large MIMO
- ✓ Partial decoupling is often adopted.

Examples of Decoupling Control

- H{dpsoh#4

$$g_{11} = \frac{2\exp(-2s)}{5s+1}; \quad g_{12} = \frac{\exp(-3s)}{8s+1}$$

- Ilqg#wkh#ghfrxsohu# D_{12}

$$D_{12} = -\frac{g_{12}}{g_{11}} = \frac{0.5(5s+1)e^{-s}}{8s+1}$$

- Frpphqw
  - Ghfrxsohu#lv#sk|vlfdoo|#uhdol}deoh#vlqfh#lw#lv#d#fdxvdo#v|vwhp
  - Qxphudwru#ghdg#wlph#lv#juhdwhu#wkdq#ghqrplqdwru#vr#wkhuh# lv#qr#suhglfwlyh#whup

- Example 2

$$g_{11} = \frac{\exp(-5s)}{5s+1}; \quad g_{12} = \frac{\exp(-3s)}{8s+1}$$

- Find the decoupler D_{12}

$$D_{12} = -\frac{g_{12}}{g_{11}} = \frac{(5s+1)e^{+2s}}{8s+1}$$

- Comment
  - Decoupler is physically not realizable
  - It has a predictive term, i.e., e^{+2s}
  - The decoupler is a non causal system

Simulation Example

- Consider a 2x2 MIMO process

$$G(s) = \begin{bmatrix} \frac{2\exp(-3s)}{12s+1} & \frac{1.2\exp(-7s)}{19s+1} \\ \frac{1.9\exp(-6s)}{17s+1} & \frac{2.4\exp(-2s)}{14s+1} \end{bmatrix}$$

- Determine the pairings using steady-state RGA

$$\lambda_{11} = \frac{1}{1 - \frac{(1.2)(1.9)}{(2)(2.4)}} = 1.9048$$

- RGA

$$\Lambda = \begin{bmatrix} 1.9048 & -0.9048 \\ -0.9048 & 1.9048 \end{bmatrix}$$

Use direct pairings: $U_1 \sim Y_1 / U_2 \sim Y_2$

- Design multi-loop PID controllers using Simultaneous Dimensionless Parameter (SDP) tuning method

Simulation Example cont….

- Type on Matlab Command Window:

```
>> Kp = [2  1.2; 1.9  2.4];
>> Tp = [12  19; 17  14];
>> Dp = [3  7; 6  2];
```

- Invoke the 'multiPItune.m' Matlab program
- After some trial-and-error setting:

$$R_p = 0.3, R_i = 1.2, R_d = 1.02$$

```
>> [Kc,Ti,Td] = multiPItune(Kp,Tp,Dp,0.3,1.1,1.02);
```

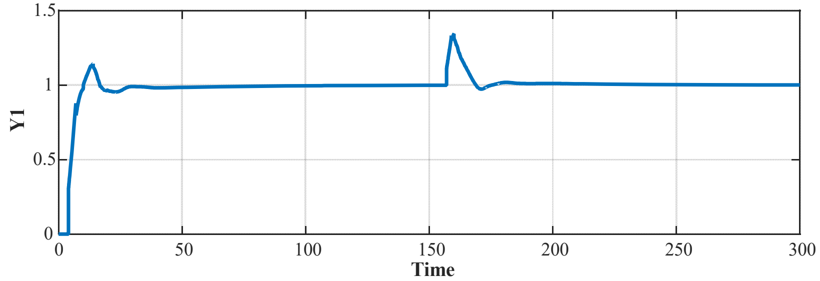
- PID controllers

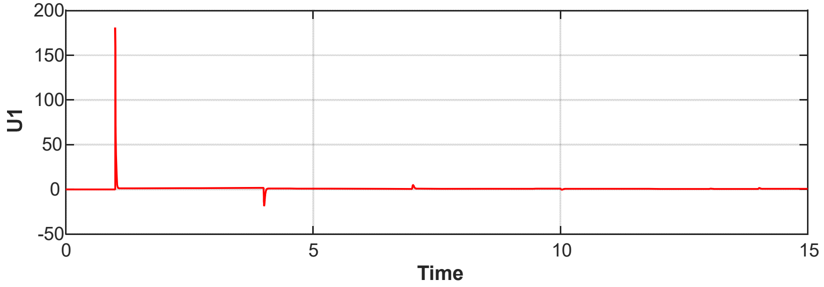
$$G_{c1} = 1.1765\left(1 + \frac{1}{6.6s} + 1.53s\right)$$

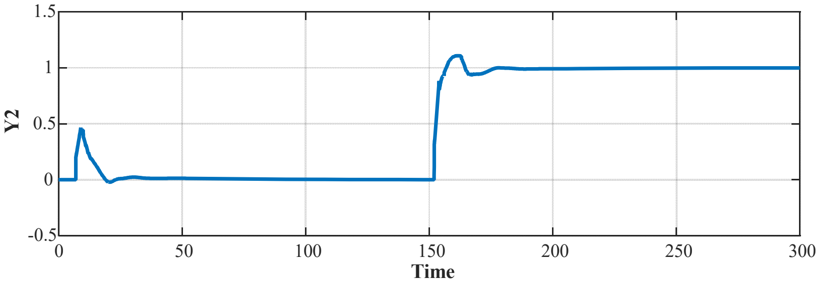
$$G_{c2} = 1.7157\left(1 + \frac{1}{7.7s} + 1.02s\right)$$

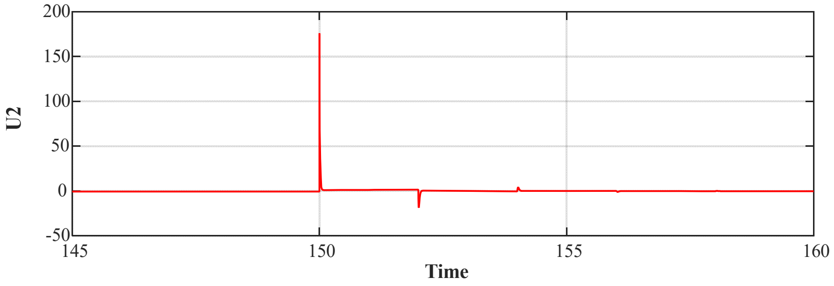
- $GM_1 = 9.2dB, PM_1 = 54deg, GM_2 = 8.8dB, PM_2 = 62deg$
- Apply sequential step changes to the setpoints of Y1 and Y2

Simulation Example cont….









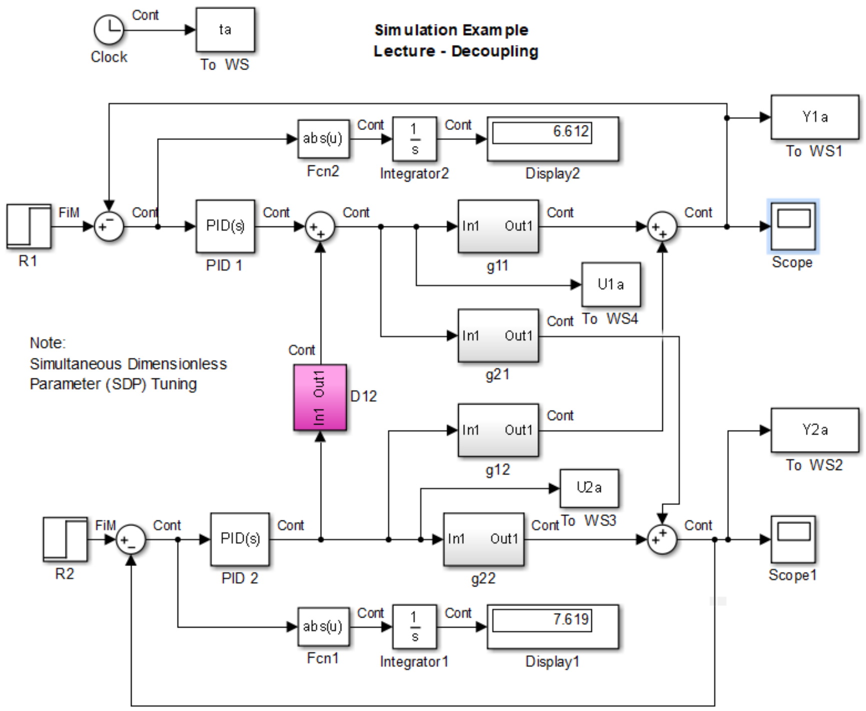
Simulation Example: 1-way decoupler $\boldsymbol{D_{12}}$

- Design decoupler D_{12}

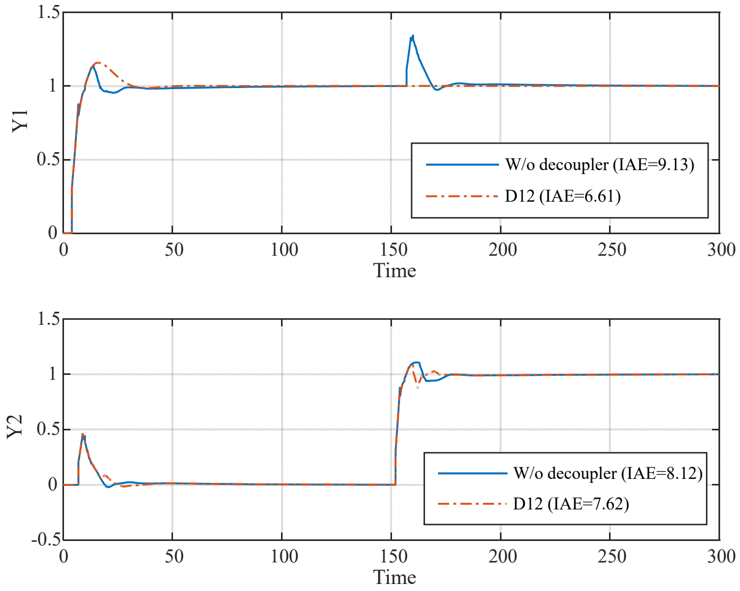
$$D_{12} = -\frac{g_{12}}{g_{11}} = -\frac{\dfrac{1.2\exp(-7s)}{19s+1}}{\dfrac{2\exp(-3s)}{12s+1}}$$

$$\therefore \quad D_{12} = -\frac{0.6(12s+1)e^{-4s}}{19s+1}$$

- **Physically realizable decoupler**



Simulation Example: 1-way decoupler $\boldsymbol{D_{12}}$



- Improvement of control performance with D_{12} decoupler
- IAE values with decoupler are smaller for both loops 1 and 2
- Notice that, when setpoint of Y_2 is applied,
- No change in the response of Y_1 when D_{12} decoupler is applied
- Decoupler effectively remove the coupling effect from the loop 2 to loop 1

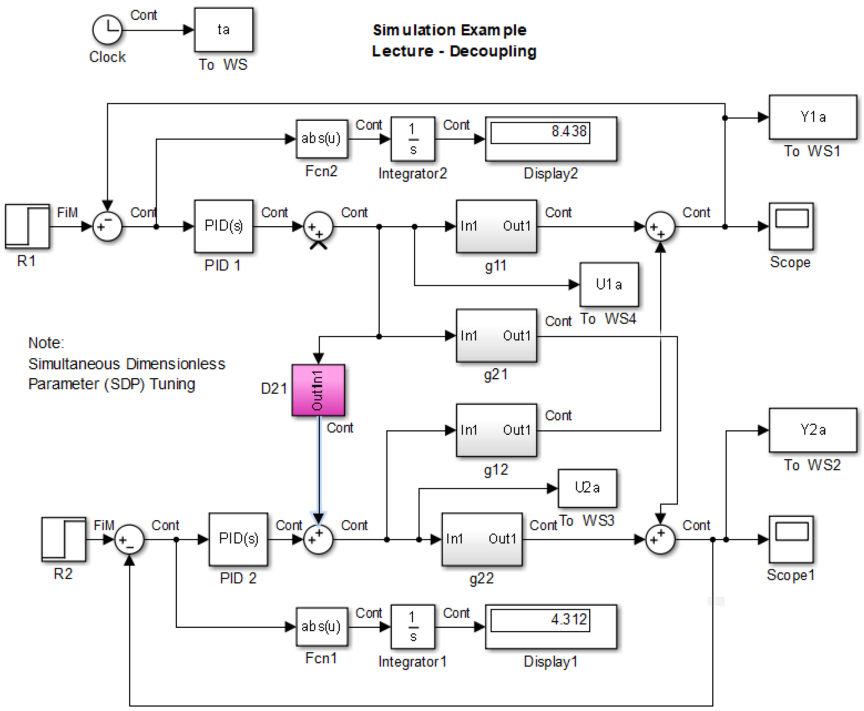
Simulation Example: 1-way decoupler $\boldsymbol{D_{21}}$

- Design decoupler D_{12}

$$D_{21} = -\frac{g_{21}}{g_{22}} = -\frac{\dfrac{1.9\exp(-6s)}{17s+1}}{\dfrac{2.4\exp(-2s)}{14s+1}}$$

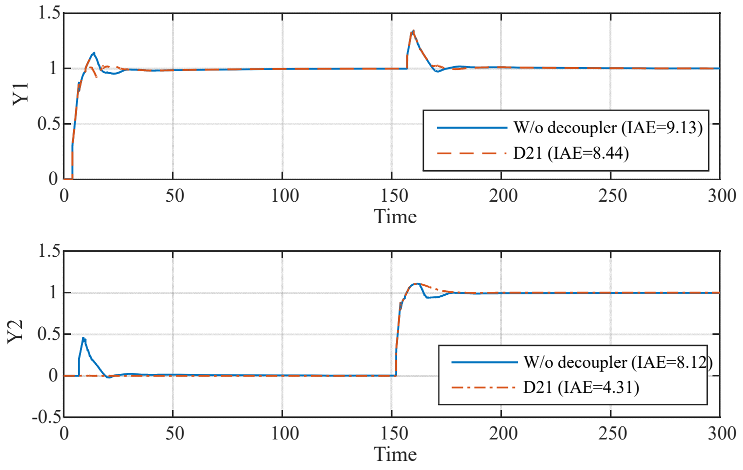
$$\therefore \quad D_{21} = -\frac{0.79(14s+1)e^{-4s}}{17s+1}$$

- **Physically realizable decoupler** D_{21}



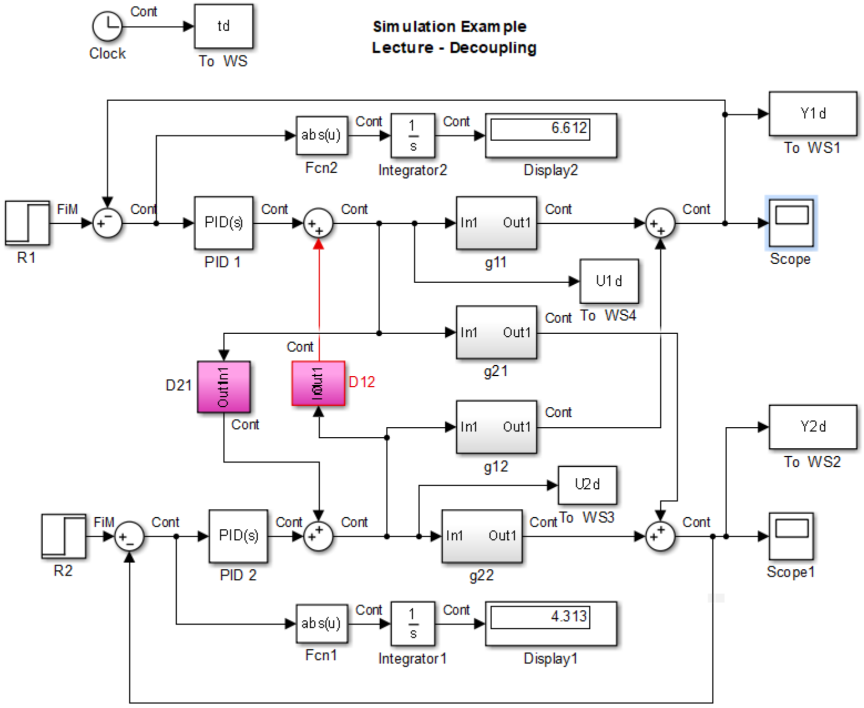
Simulation Example: 1-way decoupler $\boldsymbol{D_{21}}$

- Implementation of decoupler D_{21} also leads to improvement
- Decoupler removes the coupling effect from loop 1 to loop 2
- Notice that, when the setpoint of Y_1 is introduced,
- No change in the response of Y_2 when D_{21} is applied

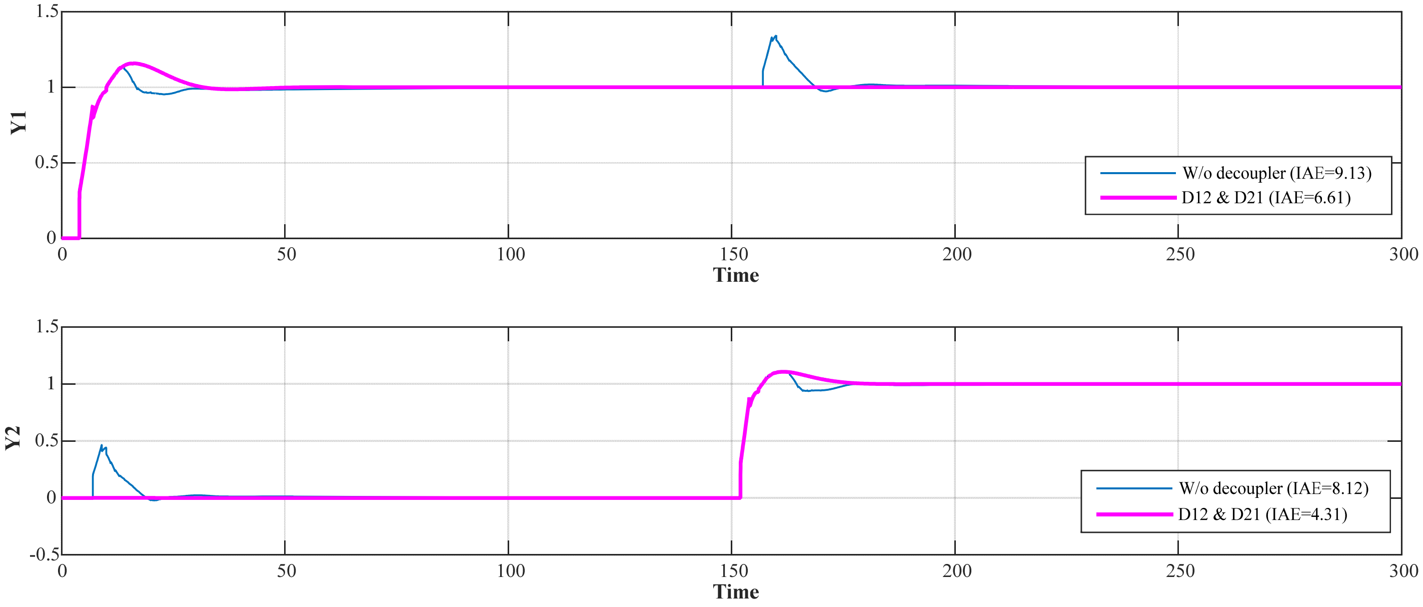


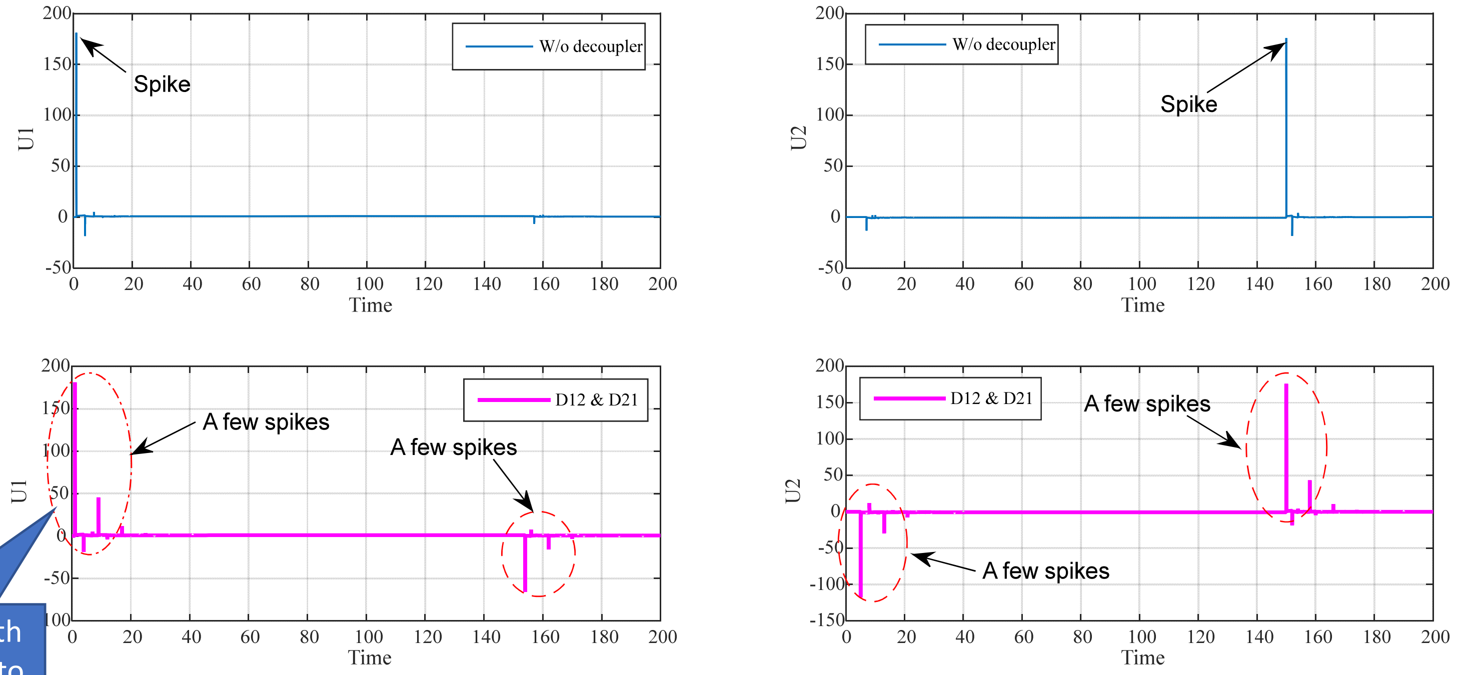
Simulation Example: 2-way decoupling control ($\boldsymbol{D_{12}}$ and $\boldsymbol{D_{21}}$)

- Apply both $\boldsymbol{D_{12}}$ and $\boldsymbol{D_{21}}$
- Simulink Model for application of both decouplers in next figure
- In this example, application of both decouplers – further improvement over 1-way decoupler
- From the plots of control actions – application of both decouplers,
- Leads to more large spikes
- Spikes can damage the instruments



Simulation Example: 2-way decoupling control ($\boldsymbol{D_{12}}$ and $\boldsymbol{D_{21}}$)





Application of both decouplers leads to larger control energy & more spikes

Summary

- Decentralized or multi-loop PID control system is widely used in process industry.
- Multi-loop control often adopted at the regulatory control layer which is crucial to achieve stability.
- Decentralized control design requires a proper controller pairings (configuration issue) due to process interactions.
- Relative Gain Array (RGA) analysis is used to solve for the configuration issue.
- Use steady-state RGA where transfer function dynamics of the plant are quite comparable.
- Use dynamic RGA when the dynamics are quite different among transfer functions
- Decouplers can be used to reduce process interactions => improve performance

References

- Riggs, J. B., & Karim, M. N. (2006). Chemical and Bio-process Control: James B. Riggs, M. Nazmul Karim. Prentice Hall.
- Mohd, N., & Nandong, J. (2018, December). Routh-Hurwitz tuning method for stable/unstable time-delay MIMO processes. IOP Conference Series: Materials Science and Engineering (Vol. 458, No. 1, p. 012051).
- Hovd, M., & Skogestad, S. (1994). Sequential design of decentralized controllers. Automatica, 30(10), 1601-1607.
- Lee, J., Cho, W., & Edgar, T. F. (1998). Multiloop PI controller tuning for interacting multivariable processes. Computers & chemical engineering, 22(11), 1711-1723.
- Vu, T. N. L., & Lee, M. (2010). Independent design of multi-loop PI/PID controllers for interacting multivariable processes. Journal of Process control, 20(8), 922-933.
- Nandong, J. (2015). Heuristic-based multi-scale control procedure of simultaneous multi-loop PID tuning for multivariable processes. Journal of Process Control, 35, 101-112.
- Larsson, T., & Skogestad, S. (2000). Plantwide control-A review and a new design procedure.