Exercise 3:
Control structure design for a generic chemical process

TABLE OF CONTENTS

1. INTRODUCTION

The main purpose of this exercise is for the students to practice the application of the plant-wide control structure design procedure described in [1]. In this Exercise we consider the application of the algorithmic approach for the control structure design to the so-called reactor-separator-recycle process, which consists of a CSTR reactor coupled with a distillation column.

This Exercise intends to show that the design of failing or poorly operating control structures can be avoided if the design procedure is done in a structured way, like the algorithmic top-down/bottom up approach for the control structure design procedure suggested by Skogestad in [2]. However, only the first 3 steps of the top-down part of the procedure will be applied to the reactor-separator-recycle process:

Step 1. Define operational objectives (economics) and constraints.

Step 2. Identify DOFs (MVs) and optimize operation for important disturbances (offline analysis).

Step 3. Select primary (economic) controlled variables.

A detailed description of the tasks related to these steps above and also additional exercise requirements will be given in the Tasks section.

All the related Exercise code is written in MATLAB and is going to be provided to the students. Since the process model is written in MATLAB, the built-in function *fmincon* is used for the process optimization. The following files are needed for the successful computation of this exercise:

- *optScriptdetailed.m*
- *objfun.m*
- *nlcon.m*
- *CSTR_SS.m*
- *colamod_SS.m*

More detailed description of the code is given in the MATLAB code section.

2. PROCESS DESCRIPTION

Today most of the chemical plants are highly integrated and interconnected due to economic considerations. This may include reactant recycle, energy integration and/or connectivity of multiple processes in so-called “Verbund” sites. The dynamic and the steady state behavior of each unit operation are usually coupled with the rest of the process subunits and in general differ significantly from their individual unit counterparts. Processes with recycle streams are typical representatives of interconnected processes.

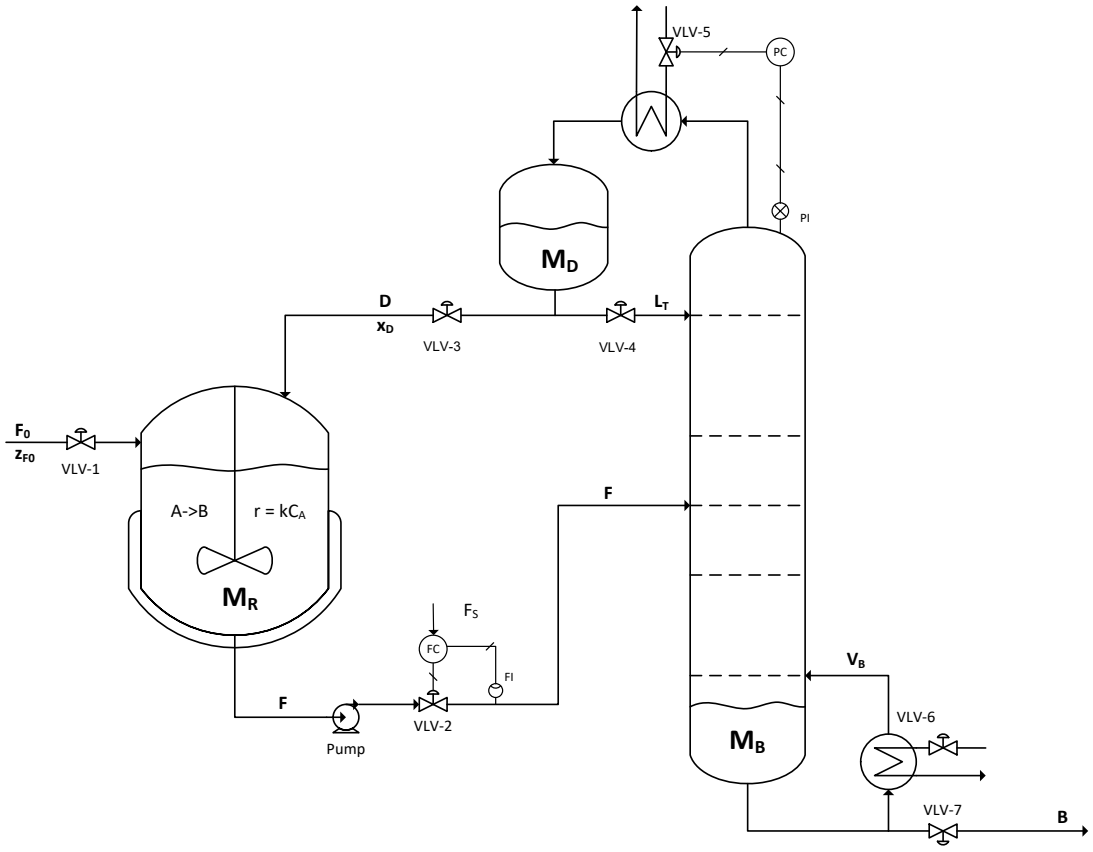

Figure 1: Reactor and separator process with recycle.

Most of the chemical processes consist of a reaction and a subsequent separation/purification section, from where unreacted chemicals are recycled and final products packed. Based on the importance of integrated plants, the simple process chosen for this exercise consist of a reactor, separator, and a recycle stream. The process flow diagram is illustrated in Figure 1. The reaction that occurs in the continuous stirred tank reactor (CSTR) is an irreversible first order elementary reaction:

$$A \rightarrow B$$

We assume that the reactor temperature is tightly controlled and it is constant, so the reaction rate constant k_1 can also be assumed to be constant as well. The output of the CSTR, which consist of a solution of A and B, is fed into the 13th stage of a fractional distillation column with 22 stages. The relative volatility α_{AB} is assumed to be 2. The products are separated and the distillate stream, which consists mostly of component A, is recycled back to the CSTR. Table 1 gives a set of possible operational conditions.

Table 1: Possible operational values for the reactor and separator with recycle process.

	Value	Units
Feed flow rate, F_0	$460/60$	kmol/min
Reactor outflow, F	$958/60$	kmol/min
Vapor boilup flow rate, V_B	$1276/60$	kmol/min
Reflux flow rate, L_T	$778/60$	kmol/min
Distillate (reflux) flow rate, D	$497/60$	kmol/min
Distillate composition, x_D	0.82	%
Condensate molar holdup, M_D	212.5	kmol
Bottom product flow rate, B	$460/60$	kmol/min
Bottom product composition, x_B	0.0105	%
Reboiler molar holdup, M_B	147.5	kmol
Reactor composition, $z_F = x_R$	0.43	%
Reactor holdup, M_r	2800	kmol

The steady state equations of the process are the following:

Reactor: Overall Mass Balance: $F_0 + D = F$

Component Mass Balance: $F_0 z_F + D x_D = F z + M_r k_1 z$

Column: Overall Mass Balance: $F = D + B$

Component Mass Balance: $F z = D x_D + B x_B$

Overall: $F_0 = B$

These can be used for analyzing the disturbance rejection capability of different control structures for this process.

Note: The reactor holdup has a steady state effect!

3. MATHEMATICAL MODEL

COLUMN

The assumptions for the column steady state model are the following:

- Constant relative volatility.
- Constant pressure.
- Constant molar flows.
- Total condenser.

This leads to the following tray balances
Component balances for a generic tray: $Vy_{i-1} + Lx_{i+1} = Vy_i + L_i x_i$
Component balances for the feed tray: $Fz_F + Vy_{i-1} + Lx_{i+1} = Vy_i + Lx_i$

The vapor-liquid equilibrium of component A can be described as:

$$y_{i,A} = \frac{\alpha_{AB} x_{i,A}}{1 + (\alpha_{AB} - 1) x_{i,A}}$$

As the mixture only includes components A and B, the concentration of B is directly given as $x_{i,B} = 1 - x_{i,A}$. The temperature in the column is defined as a linear function of the liquid concentration of component A:

$$T_i = 100 - 20 x_{i,A}$$

The mass balances for the bottom and distillate flow are given by:

$$B = F + L - V$$

$$D = V - L$$

REACTOR

The assumptions for the reactor are the following:

- CSTR.
- Constant temperature.
- 1st order reaction kinetics.

$$F_0 \cdot z_{F0} + D x_D = F \cdot z_F + k_1 M_{R,A}$$

$$F_0 + D = F$$

Note: The notation used in the derivation of the mathematical model matches the notations used in the process flow diagram depicted in Figure 1.

4. MATLAB CODE DESCRIPTION

As the focus of this course is control and not modelling, a MATLAB script is available for the optimization of the process model. The file is called *optScriptDetailed.m*. Due to the non-linear nature of the mathematical model, which can be defined as the equality constraints, we need to use a non-linear programming (NLP) solver. The name of one of the NLP solver in MATLAB is *fmincon.* Details can be found by typing *help fmincon* at the MATLAB command prompt. *fmincon* requires as an argument 2 MATLAB functions:

1. The objective function with the name *objfun.m*. It provides the objective function for our process that needs to be minimized by the NLP solver.
2. The nonlinear constraints function with the name *nlcon.m*. It provides the NLP solver with the nonlinear equality and inequality constrains evaluations at a specific point. In this process, we just need to keep all the derivative parts of the model equations at 0. It is important to mention, that *nlcon.m* only consists of variable reallocation and the call of the functions *colamod_SS.m* and *CSTR_SS.m*.

Note: All the information above is included for educational purposes. The students will receive all the above files configured and ready to run ("Press and Play" so to say ☺, no modelling required). However, we strongly recommend, that you familiarize yourself with the way of coding a NLP and how to apply SOC. Make sure you have all the files in the same folder and look at *optScriptDetailed.m*.

5. TASKS

STEP 1: DEFINE OPERATIONAL OBJECTIVES (ECONOMICS) AND CONSTRAINTS.

The objective for this case study is to minimize the energy cost for a given feed flow rate, hence the objective function is defined as

$$\text{Cost} = \text{Steam price} \times \text{total steam}$$

The process constraints for this case are:

- Reactor holdup $M_R \leq 2800$ mol
- Product quality $x_B \leq 0.0105$

STEP 2: IDENTIFY DOFS (MVS) AND OPTIMIZE OPERATION FOR IMPORTANT DISTURBANCES (OFFLINE ANALYSIS)

The main disturbances for this process are:

- Feed rate $F_0 \pm 20\%$
- Implementation error $\pm 20\%$

Task 1: Identify the degrees of freedom (DOFs) for this process. Suggest pairings for the outputs that need to be controlled but have no effect on the cost.

Task 2: Find the optimal operating point for the given feed F_0. Suggest pairings for the active constraints.

STEP 3: IDENTIFICATION OF CANDIDATE CONTROLLED VARIABLES.

Task 3: Evaluate the loss for the following list of measurements candidates to identify, whether these variables can be applied for self-optimizing control:

Variable	x_D	x_B	z_F	F	L	T_3	T_8	T_{13}	T_{18}
Loss									

There are two ways to evaluate the loss:

- Evaluate loss using the expression:

$$Loss = \frac{1}{2}\mathbf{z}^{\mathrm{T}}\mathbf{z} = \frac{1}{2}\left\|\mathbf{z}\right\|_2, \quad \mathbf{z} = \mathbf{J}_{uu}^{1/2}\left(\mathbf{H}\mathbf{G}^{y}\right)^{-1}\mathbf{H}\mathbf{Y}, \quad \mathbf{Y} = \left[\mathbf{F}\mathbf{W}_d \; \mathbf{W}_n\right]$$

- Evaluate the cost by simulating the disturbed model (the evaluated candidates are kept constant) and subtract it from the optimal cost for the disturbed process.

$$\text{Loss} = J(\mathbf{u}, \mathbf{d}) - J(\mathbf{u}_{opt}, \mathbf{d})$$

Task 4: Find a self-optimizing control structure using the Nullspace method and Exact Local Method.

Nullspace method:

Select the matrix **H** as a basis for the null space of $\mathbf{F}^T$ such that **HF**=0

Exact local method:

$$\mathbf{H}^T = (\mathbf{Y}\mathbf{Y}^T)^{-1}\mathbf{G}^y$$

Note: As a set of measurements you can use the two best candidates from the previous step.

Task 5: Show that the CV selected using the null space method is the best (self-optimizing CV) if you don't have implementation errors (measurement noise, control error).

Note: This task is purely algebraic. You do not have to use MATLAB for it.

6. BIBLIOGRAPHY

[1] S. Skogestad, "Control structure design for complete chemical plants," *Computers & Chemical Engineering*, 2004.

[2] J. J. Downs and S. Skogestad, "An industrial and academic perspective on plantwide control," *Annual Reviews in Control*, vol. 35, no. 1, pp. 99–110, Apr. 2011.