

## APPENDIX B: TUTORIALS

### B.1 SSDSA Open-loop tutorial: Exploring Steady State Behavior of an MSF Plant Case study

*Objective:* To study the steady state disturbance sensitivity analysis of an MSF in the open loop mode.

*Note:* In this tutorial, the MSF process is considered as an example. Similar procedure applies to other case studies.

#### Process description:

For detail description of the process, its main variables and schematic of the flow sheet refer to chapter 2.

Here we analyze the static behavior of the MSF plant to variation in the process inputs (disturbances). This shows how the measured variable responds at steady state to a range of changes in the feed conditions. This analysis is essential for understanding the disturbance detrimental, severity and nonlinearity effect on the process. Specifically, the steady state behavior of the distillate product,  $W_d$ , Blow Down flow ( $B_D$ ), Top brine temperature  $T_{B0}$ , last stage level and temperature ( $L_{B22}$ ,  $T_{B22}$ ) can be examined in the *simulink* module.

#### Launching the SSDSA simulation module

At the main menu of the PCCL, select the MSF case study. In the sub menu, select steady state disturbance analysis. A new simulink window, that shows the MSF in open-loop mode, pops up. One typical steady state operating condition for the input variables is shown in the corresponding boxes of the block diagram. Record these steady state values on the work sheet. To launch the SSDSA menu, simply click the *start* button.

#### Tutorial Procedure:

1. Select one of the available disturbances by marking its checkbox.
2. Select the open loop mode.
3. Select appropriate step size and number of steps.
4. Run the simulation by clicking the run button.
5. Examine the generated plots.
6. Discuss your results and make conclusions.

## B.2 SSDSA Closed-loop tutorial: Exploring Steady State Behavior of an MSF Plant

### Case study

*Objective:* To study the steady state disturbance sensitivity analysis of an MSF in the closed loop mode.

*Note:* In this tutorial, the MSF process is considered as an example. Similar procedure applies to other case studies.

#### **Process description:**

For detail description of the process, its main variables and schematic of the flow sheet refer to chapter 2.

Here we analyze the closed-loop static behavior of the MSF plant to variation in the process inputs (disturbances). This show how the measured variable responds at steady state to a range of changes in the feed conditions while certain inputs are used as manipulated variables, i.e. allowed to vary to compensate (counteract) for the disturbance effect. This analysis is essential for understanding the ability and efficiency of the chosen manipulated variable to control the process. Single loop and multiloops can be investigated.

#### **Launching the SSDSA simulation module**

At the main menu of the PCCL, select the MSF case study. In the sub menu, select steady state disturbance analysis. A new simulink window, that shows the MSF in open-loop mode, pops up. One typical steady state operating condition for the input variables is shown in the corresponding boxes of the block diagram. Record these steady state values on the work sheet. To launch the SSDSA menu, simply click the *start* button.

#### **Tutorial Procedure:**

1. Select one of the available disturbances by marking its checkbox.
2. Select the closed loop mode.
3. Select appropriate step size and number of steps.
4. Select appropriate manipulated and controlled variable pair.
5. Run the simulation by clicking the run button.
6. Examine the generated plots.
7. Discuss your results and make conclusions.

Additional steps:

8. Repeat the above procedure for the same controlled variable but with another manipulated variable.
9. Compare the results for the two different manipulated variables and make a conclusion.

10. The user can reexamine all possible manipulated variables for the same controlled variable and comes up with the best input-output pair.

Additional steps:

11. The user can study the multivariable framework by selecting two controlled variable and two manipulated variable simultaneously and run the simulation.

12. The user shall examine the results, discuss them and draw conclusions.

### B.3 Open-loop tutorial: Exploring Dynamics of an MSF Plant Case study

*Objective:* To study the open loop dynamic behavior of an MSF plant and to fit the step test data to a first order plus dead time model.

*Note:* In this tutorial, the MSF process is considered as an example. Similar procedure applies to other case studies.

#### Process description:

For detail description of the process, its main variables and schematic of the flow sheet refer to chapter 2.

Here we analyze the dynamic behavior of the MSF plant to variation in the process inputs. This show how the measured variable responds in time to changes in the feed conditions. This analysis is essential for understanding the dynamic characteristic of the process and for controller design. Specifically, the trend of the distillate product,  $W_d$ , Blow Down flow ( $B_D$ ), Top brine temperature  $T_{B0}$ , last stage level and temperature ( $L_{B22}$ ,  $T_{B22}$ ) can be examined in the *simulink* module.

#### Launching the open-loop simulation module

At the main menu of the PCCL, select the MSF case study. In the sub menu, select MSF or MSF with level control. A new simulink window, that shows the MSF in open-loop mode, pops up. One typical steady state operating condition for the input variables is shown in the corresponding boxes of the block diagram. Record these steady state values on the work sheet. To examine the steady state value of the outputs at this operating condition, simply click the *start* button. Watch the output trend by clicking the designated scope icon to monitor the measured process variables.

For clear view of the output trend, the plotting scale shown in the scope can be adjusted either by clicking the left-button of the mouse on the edge of the data curve or by clicking the auto-scale button in the toolbar menu.

#### Tutorial Procedure:

1. Step the steam flow,  $W_s$  by a specific amount, say  $\Delta W_s$ , click the start button and watch the temperature responds until it reaches a new steady state value. Record the step in the steam flow and the new Top Brine Temperature steady state value on the work sheet.

*Note:*

To step a specific input, simply double click the step function box that corresponds to the desired input. A new dialog box called "Block parameters" appears at which you can change the numerical value of the step input by entering a non-zero value for the "final value". You can also set the time at which the step change starts. Pressing ok will close the dialog box and save the changes.

Some of the case studies do not have a designated step block. In this case, one can still impose step changes by directly clicking the desired input block. As a result, a block parameter dialog box pops up allowing the user to alter the value of the specific input variable.

2. Compute the steady state gain, the dead time if applicable and the response time as described in the work sheet.

To estimate the final (steady state) value of the process output from the trend shown in the scope, simply draw a square around the end of the simulation using the left-button of the mouse. This will expand (zoom-in) that specific region. To get more precise value for the steady state, point at the curve and click the left button. This will increase the resolution of the y-axis. Similarly, to better estimate  $t_{63.2}$ , expand the region at which  $y_{63.2}$  occurs using the same technique just described. Pointing the mouse on the curve and clicking the left button once more, the resolution of y-axis and x-axis will increase allowing more accurate reading of  $t_{63.2}$ .

3. Repeat step 3 using exactly the same amount of step change,  $\Delta W_s$ , but with negative value. Record your results and compute the dynamic parameters as in the previous step.
4. The two steps in the steam flow were both the same in magnitude. Are the computed model parameters, i.e.  $K_p$  and  $\tau_p$ , the same for these two steps? How/why are they different? Does the process dynamic behave linearly?
5. Repeat steps 3 to 8 for the other process inputs, i.e.  $B_0$  (recycle flow rate),  $T_f$  (seawater feed temperature).
6. Based on the above analysis can you conclude upon which process input has the most effect on the process temperature? Can you determine which process input has the slowest impact on the process dynamics?
7. To further assess the linearity of the process dynamics, carry out two different but proportional step changes with specific ratio. Examine the resulted static gains and time constant. Are the calculated process parameters proportional to each others with the same given ratio?

### *Ideas*

Innovative user can test the process dynamics for input functions other than step changes.

The user can utilize the simulink library to includes, pulse, sine, ramp, etc signals.

*Work sheet (Effect of  $W_s$  on  $T_{B0}$ )*

Initial steady state:

$T_{fss} =$	$W_{sss} =$	$B_{0ss} =$	$T_{B0ss} =$
-------------	-------------	-------------	--------------

Positive step change

Negative step change

$\Delta W_s =$ New final value of $T_{B0} =$ $\Delta u = \Delta W_s$ $\Delta y = T_{B0} - T_{B0ss}$ $K_p = \Delta y / \Delta u =$	$\Delta W_s =$ New final value of $T_{B0} =$ $\Delta u = \Delta W_s$ $\Delta y = T_{B0} - T_{B0ss}$ $K_p = \Delta y / \Delta u =$
$t_{start} =$ $T_{B063.2} = T_{B0ss} + 0.632(\Delta y) =$ $t_{63.2} =$ $\tau = t_{63.2} - t_{start} =$	$t_{start} =$ $T_{B063.2} = T_{B0ss} + 0.632(\Delta y) =$ $t_{63.2} =$ $\tau = t_{63.2} - t_{start} =$
$t_{step} =$ $\theta = t_{step} - t_{start} =$	$t_{step} =$ $\theta = t_{step} - t_{start} =$

*Work sheet (Effect of  $T_f$  on  $T_{B0}$ )*

Initial steady state:

$T_{fss} =$	$W_{sss} =$	$B_{0ss} =$	$T_{B0ss} =$
-------------	-------------	-------------	--------------

Positive step change

Negative step change

$\Delta T_f =$ New final value of $T_{B0} =$ $\Delta u = \Delta T_f$ $\Delta y = T_{B0} - T_{B0ss}$ $K_p = \Delta y / \Delta u =$	$\Delta T_f =$ New final value of $T_{B0} =$ $\Delta u = \Delta T_f$ $\Delta y = T_{B0} - T_{B0ss}$ $K_p = \Delta y / \Delta u =$
$t_{start} =$ $T_{B063.2} = T_{B0ss} + 0.632(\Delta y) =$ $t_{63.2} =$ $\tau = t_{63.2} - t_{start} =$	$t_{start} =$ $T_{B063.2} = T_{B0ss} + 0.632(\Delta y) =$ $t_{63.2} =$ $\tau = t_{63.2} - t_{start} =$
$t_{step} =$ $\theta = t_{step} - t_{start} =$	$t_{step} =$ $\theta = t_{step} - t_{start} =$

*Work sheet (Effect of  $B_0$  on  $T_{B0}$ )*

Initial steady state:

$T_{fss} =$	$W_{sss} =$	$B_{0ss} =$	$T_{B0ss} =$
-------------	-------------	-------------	--------------

Positive step change

Negative step change

$\Delta B_0 =$ New final value of $T_{B0} =$ $\Delta u = \Delta B_0$ $\Delta y = T_{B0} - T_{B0ss}$ $K_p = \Delta y / \Delta u =$	$\Delta B_0 =$ New final value of $T_{B0} =$ $\Delta u = \Delta B_0$ $\Delta y = T_{B0} - T_{B0ss}$ $K_p = \Delta y / \Delta u =$
$t_{start} =$ $T_{B063.2} = T_{B0ss} + 0.632(\Delta y) =$ $t_{63.2} =$ $\tau = t_{63.2} - t_{start} =$	$t_{start} =$ $T_{B063.2} = T_{B0ss} + 0.632(\Delta y) =$ $t_{63.2} =$ $\tau = t_{63.2} - t_{start} =$
$t_{step} =$ $\theta = t_{step} - t_{start} =$	$t_{step} =$ $\theta = t_{step} - t_{start} =$

## B.4 Single-Loop PI Control Tutorial (1): PI Control of an MSF plant

*Objective:* To design and test a PI controller of an MSF plant for set point change and disturbance rejection.

*Note:* In this tutorial, the MSF process is considered as an example. Similar procedure applies to other case studies.

### Process description:

For detail description of the process, its main variables and schematic of the flow sheet refer to chapter 2. Here we analyze the dynamic behavior of the MSF plant under a single PI control loop.

All controllers should be designed for a specific level of operation, which include specifying an expected or desired value for set point and typical or baseline for important process parameters. Here we consider controlling the top brine temperature of an MSF plant. The main possible manipulated variable is the steam feed rate to brine heater.

A standard Proportional-Integral control algorithm will be used. It is therefore, necessary to determine the controller parameters, i.e.  $k_c$  (controller gain) and  $\tau_I$  (integral time). Next the performance of the controller can be examined through simulation for tracking different set point changes and to reject possible disturbances. To simulate real practice, the PI control is implemented in discrete time fashion with sampling time of 5 minutes.

There are several methods for determining the PI parameter values. One method is the reaction curve method. This method utilizes the process model parameters determined in the previous Open-loop Workshop tutorial. These values can be fitted in predefined tuning formula such as the Coon and Cohen formula. Record your result:

*Coon and Cohen formula:*

$$k_c = \frac{1}{k_p} \frac{\tau}{\theta} [0.9 + \theta/12\tau], \quad \tau_I = \frac{\theta[30 + 3(\theta/\tau)]}{9 + 20(\theta/\tau)}$$

$k_c =$

$\tau_I =$

If no dead time is observed, approximate the settings by:

$$k_c = \frac{1}{k_p}$$
$$\tau_I = \tau$$

## Tutorial Procedure

1. At the main menu of the PCCL, select the MSF case study. In the sub menu, select MSF with single-loop control. A new simulink window, that shows the MSF in closed-loop mode, pops up. Make sure that the controller parameters are set at zero. When the controller parameters are zero, the process will operate in open loop mode, i.e. without a controller. Click the start button to simulate the process. View the process response at the scope box and record the initial set point of the top brine temperature:

$$T_{B0 \text{ setpoint}} =$$

*If the simulation need adjustment follow the instruction under "fixing the simulation time" at the end of this handout.*

2. Enter the value of the controller gain and time integral computed in step 1 above in the designated boxes. Note that the reciprocal of the time integral multiplied by the sampling time should be entered in the  $KI$  box. Enter the initial set point value in the "set point" box. Rerun the simulation by clicking the start button. Watch the closed-loop response in the scope box. The *output* response shall remain at the recorded value above.

*Set point test:*

3. Explore the controller performance for set point change. Enter a new set point of 94 °C in the set point box, run the simulation and observe the temperature closed-loop response.

To change the set point value, simply double click the "set point" box. When a new dialog box appears, enter your desired value in the space called "constant value".

One can also examine the time response of the other outputs and the steam flow rate. At this given values for  $k_c$  and  $\tau_1$ , does the temperature reach its new set pint exactly? Note that the final value of the temperature should be higher than the set point by amount of 0.046. Is the response fast enough?

4. Try different larger values for  $k_c$  at fixed  $\tau_1$  and re-examine the closed-loop response. Try different larger values for  $\tau_1$  at fixed  $k_c$  and examine the closed-loop response. In which case, the response is faster?

*Disturbance test:*

5. Explore the controller performance for regulatory problem. Re-enter the temperature set point of step 2 (without the decimal digits) in the designated box. Click the step box of the feed temperature. Add +5 to the given value. Close the dialog box by clicking *ok*. This step change simulates a sudden increase in the feed temperature. Set the controller gain and time integral to zero. Simulate the process under disturbance without controller. Observe the temperature response.

6. Re-enter the PI controller parameter values found in step 1 above. Click the start button and watch the top brine temperature responds. Discuss how the controller improves the regulatory response.
7. Repeat step 7-8 using a disturbance of magnitude of -5 in the feed temperature.

In the case of disturbance rejection, does bringing the top brine temperature to its original set point, brings also the distillate product to its original set point? why not?

*Sampling time test:*

All the above simulation was carried out at a specific sampling time of 5 minute. The choice of the sampling time plays an important role in the controller performance. To examine such a role, change the simulation sampling time. Double click the "*time*" box and change the sampling time to, say 10. Double click the large MSF box. When a new *simulink* model appears, double click the "*signal generator*" box. Update the period to 10. After closing all dialog boxes, the new sampling time is activated. Run the closed-loop simulation (steps 4-9) again and observe the effect of the new sampling time on the control performance.

*Idea*

Innovative user can test different control structure, i.e. different output-input pairing.

*Fixing the simulation time:*

If the process response is fast, i.e. the simulation time is larger than the apparent dynamic of the process, change the simulation time. On the toolbar menu, open the "simulation" and select "parameters". A dialog box for *simulation parameter* comes up. Change the value of "Stop time" to your desired value.

## B.5 Single-Loop PI Control Tutorial (2): PI Control of an MSF plant

*Objective: To build intuition about PI tuning by trial and error method and experience the advantage and drawback of such a method.*

### Process description:

For detail description of the process, its main variables and schematic of the flow sheet refer to chapter 2. Here we analyze the dynamic behavior of the MSF plant under a single PI control loop.

In previous tutorial, the controller parameters, i.e.  $k_c$  (controller gain) and  $\tau_I$  (integral time) were determined by the reaction curve method. The reaction curve method requires building information about the process model. The reaction curve method requires stepping the process. Sometimes, it takes long time for the process to settle to another steady state value. During which disturbances may also occur. Moreover, some processes may not exhibit regular dynamic similar to that of a first-order system.

For this reason, PI controller can be tuned by the continuous cycling method as described here. The PI controller is applied in discrete time fashion with sampling time of 5 minute.

### Tutorial procedure

1. Use the same MSF module with single loop control. Set controller integral time to zero. Therefore, the integral action will be disabled. Select arbitrary value for  $k_c$ . Similarly, select an arbitrary value for the set point. Click the start button to simulate the process. View the top brine temperature response at the scope box.
2. Repeat the previous step repeatedly. Each time increase the value of  $k_c$  gradually. Observe the temperature closed-loop response at each new value of  $k_c$ . When you observe a continuous cycling of the temperature response with steady amplitude, record the following information:

The ultimate controller gain,  $k_u =$

*The ultimate gain is the controller gain at which the continuous cycling is observed.*

The ultimate period,  $T_u =$

*The ultimate period is the period of oscillation for the obtained continuous cycling.*

Using these ultimate values, compute the PI parameters as follows:

$$k_c = k_u/2.2 \qquad k_i = 1.2T_s/T_u$$

Where  $T_s$  is the sampling time. For clear capturing of the ultimate period of oscillation, you may need to zoom-in onto specific portion of the temperature

response shown in the scope box. To zoom-in, simply point on the temperature curve, press the left-button of the mouse, draw a rectangular around the region to be zoomed, release the mouse button. An expanded plot of the zoomed-in region will appear in the scope, from which you can estimate the cycling period. Clicking the mouse once more will increase the axis resolution even more allowing for better estimation of the numerical value.

3. Enter the computed values of the controller settings in their designated boxes. Explore the controller performance for set point change. Enter a value in the set point box, run the simulation and observe the temperature closed-loop response. Repeat the simulation for other set point values.
4. Explore the controller performance for regulatory problem. Re-enter the temperature set point of step 2 (obtained in previous Workshop (Tut\_msf2)) in the designated box. Click the step box of the feed temperature ( $T_f$ ). When the dialog box pops up, enter 5 for step time, 0 for initial value, +5 for final value and 5 for sample time. Close the dialog box by clicking *ok*. This step change simulates a sudden increase in the temperature of the seawater feed stream. Set the controller gain and time integral to zero. Simulate the process under disturbance without controller. Observe the process response.
5. Re-enter the PI controller parameter values found in step 2 above. Click the start button and watch the temperature responds. Compare the results with those obtained in the previous workshop, i.e. when the PI controller was tuned using reaction curve method.

The controller settings found in step 2 above are for set point change. Repeat the same procedure, but for disturbance rejection. To do so, set the set point at the initial value reported in the previous Workshop. Step the feed temperature as in step 5. Repeat step 2 till the ultimate values are obtained. Are the new ultimate values exactly the same as those found in step 2?

6. Repeat steps 3 to 6 using the new controller settings **only** if they were different than these computed previously.
7. What conclusion can you make about the control performance using the continuous cycling tuning method?

## B.6 Multivariable PI Control of Polyethylene Reactor

*Objective: To build intuition about designing multi-loop PI control and experience the effect of cross-loop interaction.*

### Process Description

For detail description of the process, its main variables and schematic of the flow sheet refer to chapter 2. Here we deal the control of at least two controlled variables. For the specific polyethylene example, the monomer and hydrogen concentrations are controlled by manipulating their corresponding flow rates.

### Tutorial procedure

6. At the Polyethylene sub menu, select multi-loop control. A *simulink* module, showing the closed-loop process in multi-loop mode, pops up. The first control-loop that links the monomer concentration with the monomer feed flow rate. The second control-loop links the hydrogen concentration with the hydrogen feed flow rate. These two control loops are chosen for demonstration purposes. The user can alter the control-loop structure as desired.
7. Compute the controller parameter values for each control loops using the same methods used in the single-loop control tutorials.
8. Insert the computed values of the controller settings in their designated boxes.
9. Explore the controller performance for regulatory problem. To do so, select an appropriate disturbance from the available list. Change the numerical of the chosen disturbance. Simulate the process under disturbance with the two controller loops in service. Observe the output response.
10. Discuss the performance of the multi loops control system. Do both controlled variable return back to their set points rapidly and smoothly? Try to increase the controller gain or time integral of the second loop and rerun the closed-loop simulation. What do you observe? And Why?
11. Change the upper and lower limit of the feed flow rates. To do so, point the mouse on the PID subsystem, press the mouse right button and select "look under mask". When a new *dialog box* appears, double click the "saturation" box and change the upper and lower limit values. Close the dialog box.
12. Rerun the simulation for a larger value for the disturbance and observe the closed loop response of both outputs. Do you observe any sign of input saturation? If yes, do you observe any offset in the output responses?
13. Now increase the integral time for the second loop 10 times and rerun the simulation. Do you observe any improvement in the control performance? Is the performance of the other loop get affected by retuning? What do you conclude.

(ideas)

14. Look in the literature and/or books for tuning formula that apply for multi-loop PID control and try to implement it on this process.
15. In the above tests, the control pairing was predefined. It is more rigorous to pair the control loops based on Relative Gain Array (RGA). To do so, follow the following steps"

First compute the steady gain matrix as follows:

$$G = \begin{bmatrix} k_{pT,Tf} & k_{pT,Qc} \\ k_{pC,Tf} & k_{pC,Qc} \end{bmatrix}$$

Compute the inverse of the transpose of the gain matrix:

$$G_T = (G^T)^{-1}$$

Compute the RGA as follows:

$$RGA = \begin{bmatrix} g_{11} \bullet g_{T11} & g_{12} \bullet g_{T12} \\ g_{21} \bullet g_{T21} & g_{22} \bullet g_{T22} \end{bmatrix}$$

If the off-diagonal elements of the RGA are negative or positive close to 0.5, then the process is highly interactive. Therefore, decentralized control may not perform well.

Pair the controlled variable and manipulated variable so that the corresponding relative gain (row element) is positive and as close to one as possible. According to this analysis does the pairing suggested by the RGA in consistent with the pairing used in the above tests?