Cambridge University Engineering Department

DESIGN PROJECT GF1: CONTROL SYSTEM

Week 4: Group Activity

1 Introduction

By now you should have a complete control system working on the evaporator process.

You should now combine with another pair to form a group of 4. As a group you have to execute the following tasks:

1. Evaluate whether the controllers you have designed over the previous 3 weeks perform sufficiently well at a number of operating points — see section 3 for background and an explanation of this.
2. If your conclusion to task 1 is negative, investigate how much improvement you can obtain by using one of the two following options:

Option 1 : a *gain-scheduled* controller design.

Option 2: Design a multivariable controller using state-feedback — see section 4 for an explanation of this.

3. Make a proposal on the basis of the investigations above — either for a constant-gain state-feedback design, or for a gain-scheduled controller, which works sufficiently well over the range of operating points.
4. Justify your final recommendation for a control system design for the evaporator, and demonstrate your design. Choose a set of relevant scenarios that illustrate the performance and robustness of your design.

Background material for section 4 has been presented in module 3F2. Those who opt for the state-feedback design should have taken module 3F2.

2 Planning and Execution

It is up to you as a pair to decide how to accomplish these tasks, but you should start by planning how to do it. Your final report should show the plan (diagrammatically if appropriate).

You are free to choose any plan that you agree upon as a pair. For example, you may wish to split the tasks between you, or to allocate roles such as ‘manager’, ‘tester’, ‘model builder’, etc.

There is no single best way to do this. An effective plan will take into account the personalities and capabilities of the group, as well as the tasks that have to be accomplished.

But *time is short.* Planning is essential, but don't take too long over it.

3 Change of Operating Point

It is increasingly important, for economic reasons, to operate processes at more than one operating point (product specification). For example, it may be necessary to change the product composition $X2$ from 25% to any value in the range 15% – 35%.

Investigate how well your existing controllers perform over this range of operating points. There is no precise performance specification, so you will have to use your judgement (as is quite common in practice). However, as a guideline, the behaviour of the previously designed controllers at 25% may be regarded as satisfactory, and the behaviour over the entire range of $X2$ is required to not be significantly worse than this (in terms of speed of response, overshoot, oscillations, settling time etc).

If not satisfactory, the performance of the controllers can be improved by making them 'adapt' to the operating point in some way. The simplest way of doing this is to design several controllers, each one for a different operating point — for instance, one for $X2 = 15\%$, one for $X2 = 25\%$, and one for $X2 = 35\%$. Then either switch between them, or interpolate smoothly between them, choosing the controller on the basis of either the current $X2$ value, or its current set point. This is known as *Gain Scheduling.*

For more information see Chapter 10 of [3].

4 State Feedback

There is considerable interaction between the three controlled variables $X2$, $P2$ and $L2$. Better control may therefore be possible by implementing a *multivariable* controller which controls all three variables in a coordinated way, rather than three separate single-loop controllers. You can do this by implementing *state feedback.*

4.1 Basic state feedback

State feedback theory was covered in module 3F2. Here is a very brief reminder. Suppose the nonlinear plant

$$\dot{x} = f(x, u)$$

has an equilibrium (x_0, u_0), namely $f(x_0, u_0) = 0$, where x is the state vector and u is the input vector. Consider small deviations away from this equilibrium condition:

$$\tilde{x} = x - x_0, \qquad \tilde{u} = u - u_0.$$

Then

$$\dot{\tilde{x}} \approx A\tilde{x} + B\tilde{u} \text{ if } A = \left[\frac{\partial f}{\partial x}\right]_{x=x_0} \text{ and } B = \left[\frac{\partial f}{\partial u}\right]_{u=u_0}.$$

If state feedback of the form

$$\tilde{u} = -K\tilde{x}$$

is applied, then the resulting closed-loop system has the equation

$$\dot{\tilde{x}} = [A - BK]\tilde{x}.$$

The eigenvalues of the matrix $A - BK$ are the closed-loop poles of the system. The behaviour of the system will be governed by the location of these eigenvalues in the complex plane. If the closed-loop system is stable, then $\tilde{x} \to 0$, or $x \to x_0$, which is what we want if x is to be held as close as possible to the set-point x_0 despite disturbances.

Before starting on this, remove the actuator lags in your plant model. *This is for the design only, you will need to put them back for testing.* This will simplify the design a lot, as the state vector will then be $x = [L2, X2, P2]^T$, and x_0 will be the vector of set-points for the controlled variables. So all the state variables will also be measured outputs. If this were not the case then you would have to use an *observer* to estimate those states which are not outputs.

To design a state-feedback scheme of this kind, you first need to find a linear state-space approximate model of the process at the operating point. Use `linmod` to do this, as before. (Remember to remove all previously implemented controllers before you do this.) Then design, implement and test a state feedback controller which will place the closed-loop poles at locations chosen by you. The MATLAB function `place` will help you to design a suitable state-feedback matrix K.

When implementing the state-feedback, remember that after you have formed the signal $\tilde{u}$, you need to add the 'steady-state' or 'trim' value u_0, in order to get the correct input signal u.

Note that when you put the actuator lags back, the actual closed loop poles will not be precisely where you have attempted to place them. How does the difference depend on where you have attempted to place the poles? Observe also that if you try to make the response too fast (the poles too far to the left) then the feedback gains will become too large and the control signals will hit their constraints, resulting in very nonlinear behaviour.

4.2 Integral action

The state-feedback scheme outlined so far will only work for one particular set-point vector. If x_0 changes, then you have to know the new value of u_0, and change it appropriately. Furthermore, this needs accurate knowledge of the 'DC' gains (zero-frequency gains) in the plant, which is usually not available. Using an inaccurate value of u_0 has the same steady-state effect as a constant disturbance acting on the states. But the standard way to counteract constant disturbances is to use integral action; this has the advantage that the correct value of u_0 is in effect generated automatically, even if the set-points change and the DC gains are not known accurately.

Integral action can be introduced as follows — the 3F2 notes have a slightly different scheme. Introduce a new variable v, where

$$\dot{v} = \tilde{x}.$$

Then, for small deviations from equilibrium, the complete system behaviour is described approximately by the state equation

$$\begin{bmatrix} \dot{\tilde{x}} \\ \dot{v} \end{bmatrix} = \begin{bmatrix} A & 0 \\ I & 0 \end{bmatrix} \begin{bmatrix} \tilde{x} \\ v \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} \tilde{u}.$$

Then, apply a control signal of the form

$$\tilde{u} = -[K, L] \begin{bmatrix} \tilde{x} \\ v \end{bmatrix}.$$

So a suitable matrix $[K, L]$ can again be found by using the function `place`, but using $\begin{bmatrix} A & 0 \\ I & 0 \end{bmatrix}$ instead of A, and $\begin{bmatrix} B \\ 0 \end{bmatrix}$ instead of B.

4.3 Optimal Control

An alternative way to choose the state feedback matrix K is to make it optimal in some sense. Most commonly it is chosen to minimise a quadratic cost of the form

$$J = \int_0^\infty \left[\tilde{x}(t)^T Q \tilde{x}(t) + \tilde{u}(t)^T R \tilde{u}(t)\right] dt \tag{1}$$

where Q and R are weighting matrices, and the system starts in some initial state $\tilde{x}(0)$ and is to be brought to a final state $\tilde{x}(\infty) = 0$. The MATLAB functions `lqr`, `reg` and `lqgreg` will help with this. For more information see Chapter 7 of [3], Chapter 5 of [2], or Chapter 7.4.3 of [1].

Note: The MATLAB functions `lqr`, `reg`, `kalman`, etc are in the MATLAB *Control Systems Toolbox* [4]. You can run the demo `ctrldemo` to get a better idea of how to use this Toolbox effectively. Alternatively you can type `demo`, then select `Toolboxes`, then `Control System`. Type `help control` to see all the functions available in this Toolbox.

5 Demonstration

During the last Tuesday morning session (9–11am) your pair will be asked to demonstrate its design to me and the three other pairs. I may want to check details of your model, of your controller designs, and the performance you obtain with them. Your work should be substantially complete by this time, but there will be time to make some corrections should I find errors.

Make sure you come to this meeting with specific simulations that illustrate the salient points of your design.

6 Final report

Your final report should cover the whole project, not just the final week. Your interim reports should be attached as appendices, and material in them need not be repeated; you should refer to them as appropriate.

The report should not be a blow-by-blow account of what you did. Imagine it is a report to a plant manager, recommending a control system to be installed on the process. You need to show that you have examined some alternatives, how they compare, and why (s)he should have confidence in your recommendations. Assume that your reader is hard-nosed: you will want to show off how clever you have been, but (s)he will want to know whether your control scheme will really work if implemented on the real process. Will it work when the process parameters change? Does it keep the process safely away from its constraints? Does it keep the product composition close to its set-point when typical disturbances come along? Will it do a good job when the operating point changes? These are some of the questions you might address.

The part of your report covering the last week's group activity should be your own work. It is permissible to have the same graphs, tables, etc as the other reports in your group, and you can cross-refer to each other's reports, for example when referring to work done by other members of the group. It is also permissible to have a common 'executive summary' of a paragraph or so, summarising your overall conclusions and recommendations.

References

[1] Franklin, G. F., Powell, J. D., and Emami-Naeini, A., *Feedback Control of Dynamic Systems, 5th edition*, Reading MA: Prentice-Hall, 2006. (CUED Shelfmark: QC 261)

[2] Maciejowski, J. M., *Multivariable Feedback Design*, Addison-Wesley, 1989. (CUED Shelfmark: QR 25)

[3] Newell, R. B., and Lee, P. L., *Applied Process Control, A Case Study*, New York: Prentice-Hall, 1989. (CUED Shelfmark: QL 30)

[4] *Matlab Control Toolbox User's Guide.*

R. Sepulchre May 2022