Available online at www.sciencedirect.com
SCIENCE DIRECT®
ELSEVIER
Journal of Process Control 15 (2005) 259–271
JOURNAL OF PROCESS CONTROL
www.elsevier.com/locate/jprocont

Controller design for serial processes

Audun Faanes [1], Sigurd Skogestad *

Department of Chemical Engineering, Norwegian University of Science and Technology, N-7491 Trondheim, Norway

Received 2 May 2003; received in revised form 26 June 2004; accepted 11 July 2004

Abstract

In this paper, we aim at obtaining insight into how a multivariable feedback controller works, with special attention to serial processes. Serial processes are important in the process industry, and the structure of this process makes it simple to classify the different elements of the multivariable controller.

In particular, we focus on the difference between the feedforward and feedback parts of the controller. Feedforward control may improve the performance significantly, but is sensitive to uncertainty, especially at low frequencies. Feedback control is very effective at lower frequencies where high feedback gains are allowed.

An example of neutralization of an acid in a series of three tanks is used to illustrate the ideas.

Keywords: Control structure; Serial process; Multivariable control; Feedforward; Feedback

1. Introduction

Before designing and implementing a multivariable controller, there are some questions that are important to answer: What will the multivariable controller really attempt to do? Will a multivariable controller significantly improve the performance as compared to a simpler scheme? How accurate a model is needed?

Conceptually, a multivariable controller uses the two basic principles of "*feedforward*" action, based mainly on the model (for example the off-diagonal decoupling elements of the controllers), and *feedback* correction, based mainly on the measurements. There is a fundamental difference between feedforward and feedback controllers with respect to their sensitivity to uncertainty. Feedforward control is sensitive to *static* uncertainty, whereas feedback is not. On the other hand, aggressively tuned feedback controllers are very sensitive to uncertainty in the *high-frequency* (*crossover*) *frequency region*. Similar differences with respect to uncertainty can be found for multivariable controllers.

A multivariable controller often yields significant *nominal* improvements compared to local single-loop control, and this is largely because of the "feedforward" action. With *model error*, the feedforward effect may in fact lead to worse performance. The use of feedback from downstream measurements depends less on the model, as use of high feedback gains at low frequencies removes the steady-state error. However, at higher frequencies high feedback gains may lead to stability problems, and it is at these higher frequencies one may have the largest benefit of the model-based "feedforward" action of the multivariable controller.

In this paper, we discuss these issues for the important class of *serial processes*, in which the states in one process unit influence the states in the downstream unit, but *not* the other way round. This structure is very common in the process industry, where the outlet flow of one process enters into the next. One example, which will be studied in Section 4, is neutralization performed

* Corresponding author. Tel.: +47 73 59 41 54; fax: +47 73 59 40 80.

E-mail address: skoge@chemeng.ntnu.no (S. Skogestad).

[1] Also affiliated with Statoil ASA, TEK, Process Control, N-7005 Trondheim, Norway.

0959-1524/$ - see front matter
doi:10.1016/j.jprocont.2004.07.001

in several tanks in series. Examples of processes that are not serial are processes with some kind of recycle of material or energy. Even for such processes, however, parts of the process may be modelled as a serial process, if the outlet variations of the last unit are dampened through other process units before it is recycled.

Buckley [1] discusses control structure design for serial processes and distinguishes between material balance control (control of inventory or pressure by flow rate adjustments) and product quality control (control of quality parameters such as concentration). Shinskey [2] and McMillan [3] present methods for the design of pH neutralization processes. Mixing tanks are used to dampen disturbances, and they find that the total volume may be reduced by use of multiple stages with one control loop for each tank. Another advantage with multiple stages is that one may use successively smaller and smaller control valves, leading to a more precise manipulated variable in the last stage. McMillan and Shinskey both recommend different sized tanks to avoid equal resonance frequencies in the tanks, but this has later been questioned [4,5].

A discussion on the open-loop response of serial processes is given by Marlin [6, p. 156]. Morud and Skogestad [7] note that the poles and zeros of the transfer function of a serial process are the poles and zeros of the transfer functions of the individual units. Thus, the overall response may be predicted directly from the individual units, in contrast for example to processes with recycle. Many series connections of processing units are not really serial processes, as the response of each unit also depends on the downstream unit (for example if the outlet flow rate from a unit depends on the pressure in the subsequent unit) [6,8,9]. Morud and Skogestad denote the latter process structure *cascades*, whereas Marlin uses the terms *noninteracting* and *interacting* series, respectively, for the two structures.

In Section 2 we develop the model structure for serial processes and discuss some of their properties. In Section 3 the control of serial processes is discussed, and the division of the controller in feedforward, feedback and resetting blocks is presented. One popular multivariable controller is MPC, and to be able to use the theory for linear systems, we summarize in Appendix A how to express an unconstrained MPC combined with a state estimator on state space and transfer function form, see more details in [10]. The ideas of the paper are further illustrated through an example with pH neutralization in three stages (Section 4). The paper is concluded by a short discussion (Section 5) and conclusion (Section 6).

2. Model structure of serial processes

We define a serial process by the following (also see Fig. 1):

A serial process can be divided into a series of subprocesses or units, where the states in each unit depend on the states in the unit itself (x_i), the states in the upstream unit (x_{i-1}), and the exogenous variables (u_i, d_i) to the unit.

The model for unit no. i can then be expressed as

$$\frac{\mathrm{d}}{\mathrm{d}t}x_i = f_i(x_i, x_{i-1}, u_i, d_i) \tag{1}$$

where x_i and x_{i-1} are the state vectors for unit i and unit $i-1$, respectively, and the external input is divided into a vector of manipulated inputs, u_i, and disturbances, d_i. We further define the outputs from a unit as a function of the states of this unit

$$y_i = g_i(x_i) \tag{2}$$

It is easy to also include direct throughput terms, i.e., define $y_i = g_i(x_i, x_{i-1}, u_i, d_i)$, but it makes the expressions below slightly more complex.

We linearize (1) and (2) around a working point, introduce $A_{i,j} = \partial f_i/\partial x_j$; $j = i, i-1$, $B_i = \partial f_i/\partial u_i$, $C_i = \partial g_i/\partial x_i$, and $E_i = \partial f_i/\partial d_i$ and let the variables represent deviations from this working point. Applying Laplace transformation, and recursively inserting for variables from the upstream process unit, we obtain:

$$y(s) = G(s)u(s) + G_d(s)d(s) \tag{3}$$

where we have defined the total output vector, $y(s)$, as all the outputs, $u(s)$ as all the manipulated inputs, and $d(s)$ as all the disturbances. Defining

$$M_i = (sI - A_{i,i})^{-1} \tag{4}$$

we get

$$G(s) = \begin{bmatrix} C_1M_1B_1 & 0 & 0 & \cdots & 0 \\ C_2M_2A_{2,1}M_1B_1 & C_2M_2B_2 & 0 & \cdots & \vdots \\ \vdots & \vdots & & \ddots & 0 \\ C_nM_n\prod_{r=1}^{n-1}[A_{n-r+1,n-r}M_{n-r}]B_1 & C_nM_n\prod_{r=1}^{n-2}[A_{n-r+1,n-r}M_{n-r}]B_2 & \cdots & \cdots & C_nM_nB_n \end{bmatrix} = \begin{bmatrix} G_{1,1} & 0 & 0 & \cdots & 0 \\ G_{2,1} & G_{2,2} & 0 & \cdots & \vdots \\ \vdots & \vdots & \ddots & & 0 \\ G_{n,1} & G_{n,2} & \cdots & \cdots & G_{n,n} \end{bmatrix} \tag{5}$$