Modeling, Identification and Control, Vol. 30, No. 3, 2009, pp. 149–155, ISSN 1890–1328

Feedback: Still the Simplest and Best Solution

Sigurd Skogestad

Department of Chemical Engineering, Norwegian University of Science and Technology (NTNU), NO-7491 Trondheim, Norway. E-mail: sigurd.skogestad@chemeng.ntnu.no

Abstract

Most engineers are (indirectly) trained to be "feedforward thinkers" and they immediately think of "model inversion" when it comes to doing control. Thus, they prefer to rely on models instead of data, although feedback solutions in most cases are much simpler and more robust.[*]

Keywords: Classical control, stabilizing control, fundamental limitations, self-optimizing control, robustness

1 Introduction

In this paper, we will discuss and compare the following two approaches to control

1. Feedback control (output measurement based)

2. Feedforward control (model-based)

Feedback is a very powerful tool, but, maybe because of its simplicity, it often gets overlooked and forgotten, and it seems that its advantages need to be rediscovered every 20 years or so. Simple feedback loops, often using simple PID (proportional-integral-derivative) control algorithms, became the main tools for control engineers in the 1930's or so. The theory behind this is known as "*classical control*" and is documented in the works of Bode, Nichols and others from the 1940's. The main lesson of classical feedback control is: "*Pair close and crank up the gain*". More specifically, by "*pair close*" it is meant that one should use an input-output pair with a small phase lag (or, equivalently, small "effective time delay"). The effective time delay is the sum of the apparent time delay caused by dead time, inverse responses (unstable RHP zeros) and high-order lags. By "*crank up the gain*" it is meant that the gain around the feedback loop should be large to get good performance and robustness against uncertainty (changes in the system components). However, the loop gain cannot be too large, otherwise we get instability because of the presence of phase lag (effective time delay).

About 20 years later, in the era of "*optimal control*" in the 1960's, this lesson was forgotten. The slogan of optimal control was that anything could be controlled provided one had a model and the states were observable and controllable.

It took another 20 years until we reached the 1980's and negative feedback ("*crank up the gain*") was rediscovered with the introduction of "*robust control*" by John Doyle and others. Bode's stability criterion was replaced by the "small gain theorem" and powerful new robustness results were derived, including the structured singular value. I was lucky enough to be part of this development during my PhD study days at Caltech from 1983 to 1987 and this led to the publication of the book "Multivariable feedback control" in 1996, Skogestad and Postlethwaite (2005).

However, now, about 20 years later, it seems that the feedback lesson is again being forgotten. To some extent, this is caused by the extensive use of model predictive control (MPC), which is an improved version of optimal control from the 1960's. MPC is model based and feedback is only introduced indirectly. However, also MPC indirectly uses feedback as its main mean of

[*] This is an updated version of a paper originally presented at the IEEE conference ICIEA, Xi'an, China, May 2009.

doi:10.4173/mic.2009.3.5

dealing with uncertainty, and is subject to the same advantages and limitations as derived in classical and robust control.

Feedback is sometimes claimed to be outdated, and the argument is: "Feedback is based on what has happened and therefore comes too late. The future is to put focus on getting models and using model-based control". This statement is right in saying that one could use models to improve robustness and performance of feedback control, but it is wrong in saying that feedback is not part of the future. So it is time to rediscover the power of feedback control, and the objective of this paper is to point out the following *three fundamental advantages of feedback control*:

1. **Simplicity**. With feedback one can get tight control with only a very crude model.
2. **Robustness**. Feedback is required for making a system adapt to new conditions.
3. **Stabilization**. Feedback is the only way to fundamentally change the dynamics of a system.

We here concentrate on man-made systems, but **feedback is even more prevalent in nature**. Actually, because of the three fundamental advantages of feedback just mentioned and because control systems in nature have to be simple and robust, we can make this bold statement without knowing anything about the subject. The insight about the fundamental importance of feedback was the basis for the field of cybernetics, Wiener (1948). Today, with the great leaps forward in systems biology one is discovering the details of how nature uses (negative) feedback, but the insight that nature must rely on feedback was clear more than 50 years ago, Wiener (1948). (Nature also uses positive feedback, typically, to generate instability and quick changes, but this paper deals with the virtues of negative feedback).

2 Simplicity and Robustness of Feedback: The Feedback Amplifier

One important step in the development of feedback control theory was the introduction of Harold Black's feedback amplifier for telephone communication in 1927, Kline (1993). The engineers at the Bell laboratory in New York were trying to make an accurate amplifier, but were facing problems with the variation and uncertainty of the amplifier. The engineers suggested various "ingenious" feedforward ideas to correct for this. Then came Harold Black and proposed to wrap a simple high gain feedback loop around the amplifier, and suddenly there was no need to build an accurate amplifier. The solution was so simple that people found it difficult to believe.

To understand the feedback amplifier consider Figure 1. The objective is to amplify the signal $r(t)$ by a factor a such that $y_{ideal}(t) = ar(t)$. The original feedforward approach was to build a very accurate adjustable amplifier (Figure 1). The problem is that this is very difficult and also that the amplification will vary with the frequency of the signal $r(t)$. The corresponding feedback solution of Black is shown in Figure 2. The closed-loop response becomes

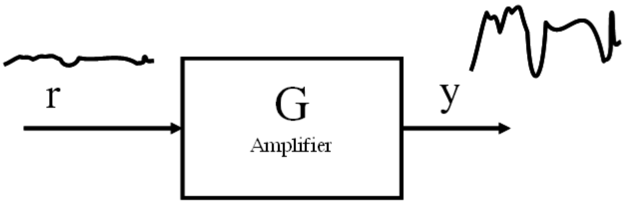

Figure 1: *Solution 1 (feedforward): $G = k = a$ (adjustable).*

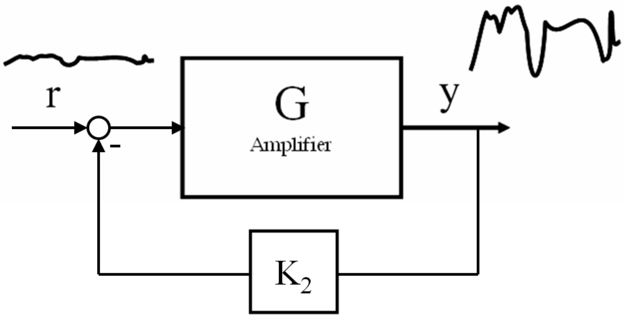

Figure 2: *Solution 2 (feedback): $G = k$ (large constant gain $k > a$). $K_2 = 1/a$ (adjustable).*

$$\frac{y}{r} = \frac{G}{1 + GK_2} \approx \frac{1}{K_2} = a \qquad (1)$$

where the approximation holds provided we have high-gain feedback with $|GK_2| \gg 1$. Thus, we have the (seemingly incredible) fact that the overall amplification is independent of the amplifier G. Even a lousy amplifier will work well, provided it has a sufficiently high gain and we can measure the result and compare it with the desired signal. Really, the only thing one needs to know is the sign of G.

However, except for first- or second-order systems with no effective delay, high-gain feedback has its price, and to avoid instability we know from Bode's stability criterion that the loop gain must be less than 1 at the critical frequency where the phase lag around the loop is -180 degrees (-360 degrees including the negative gain in the feedback loop). Otherwise, signals at this frequency will increase in magnitude for each pass through the loop and we have instability.

A comparison of the robustness of feedforward and feedback control is shown in Figures 3 and 4. The plant is $y = Gu + G_d d$ (this is not the amplifier case but the results are similar). The plant is $G = k/(10s+1)$ (s is the Laplace variable) and nominally $k = 10$. The response is shown for a step disturbance d where $G_d = 10/(10s+1)$. Since nominally $G = G_d$, the ideal feedforward control law is $u = -G^{-1}G_d d = -d$ (d is a measured disturbance). In Figure 3 we compare the response with that of a feedback control law $u = -K(s)y$ (y is a measured output) where $K(s)$ is a PI feedback controller with integral time 10 [s] and gain $1/\tau_c = 1$, corresponding to a closed-loop time constant $\tau_c = 1$ [s] (selected according to the IMC tuning rule, Skogestad (2004b)).

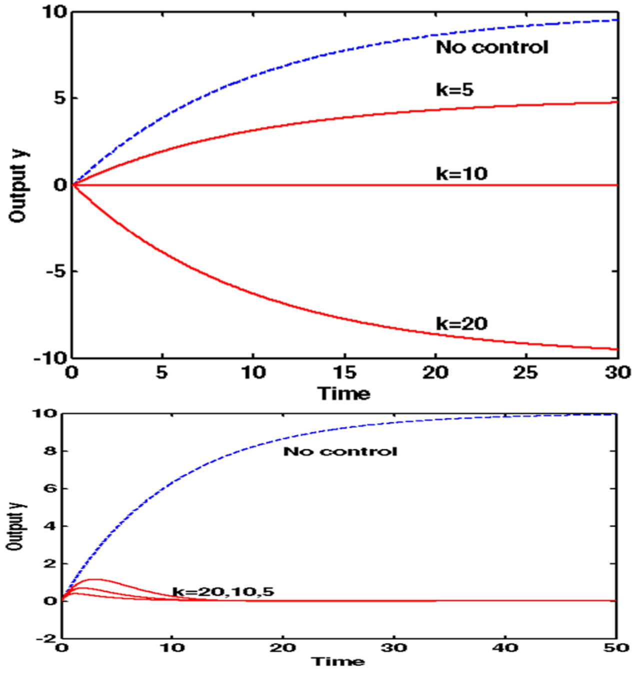

Figure 3: *The magic of feedback!* ☺ *Comparison of feedforward and feedback control with respect to gain error. Nominal gain is* $k = 10$. *Top: Feedforward control. Bottom: Feedback control.*

Nominally, $k = 10$ and the feedforward law gives perfect control. However, if the plant gain is $k = 5$ (which is only half of the disturbance gain) then we get only half of the correction, and if $k = 20$ we overcompensate such that feedforward is as bad as no control, but in the opposite direction. This should be compared with the excellent robustness of feedback control

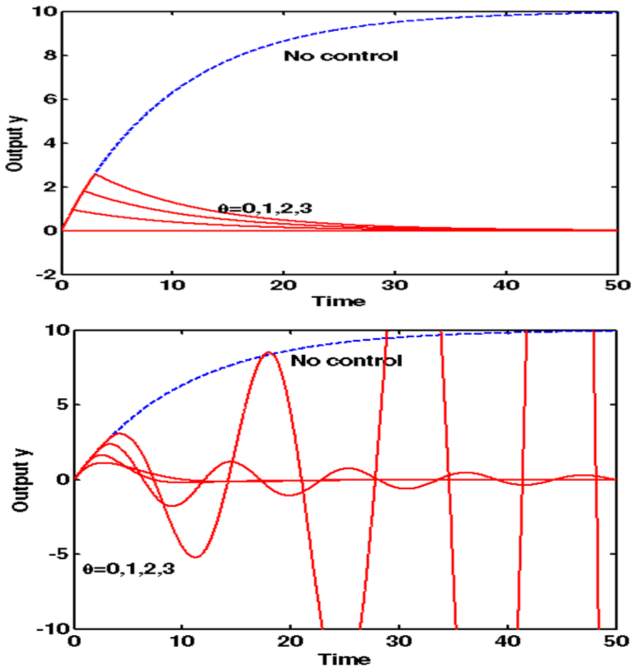

Figure 4: *The problem with feedback...* ☹ *Comparison of feedforward and feedback control with respect to time delay error. Nominal delay* $\theta = 0$. *Top: Feedforward control. Bottom: Feedback control.*

(bottom in Figure 3), which is hardly affected by the variation in the plant gain k.

As mentioned, the main problem with feedback control is its sensitivity to phase lag (effective time delay) in the loop as illustrated in Figure 4. The feedback strategy handles well a time delay increase from $\theta = 0$ to $\theta = 1$ [s]. However, when the time delay increases to $\theta = 2$ [s] (which is twice the closed-loop response time) the system starts oscillating, and for to $\theta = 3$ [s] the system is unstable. On the other hand, feedforward control is only weakly affected by the time delay error.

The conclusion is that feedback control is very robust to plant variations as long as the phase lag (effective time delay) is relatively small such that we can use a large feedback gain.

3 Simplicity and Robustness of Feedback: Self-Optimizing Control

Let us now look at a quite different issue: How should we link optimal operation and control, or in terms of feedback control: What should we control? The idea of self-optimizing control is to turn open-loop (“feedforward") optimization (strategy shown at right in Figure 5) into a setpoint feedback control problem (left strategy in Figure 5) (Morari et al. (1980); Skogestad (2000)). The trick is to find the right "magic" variable c to control. The term "magic" is here used for two reasons. First, finding it may not be easy, and second, if one can find a good variable then a simple feedback scheme may work so well that it almost seems like magic.

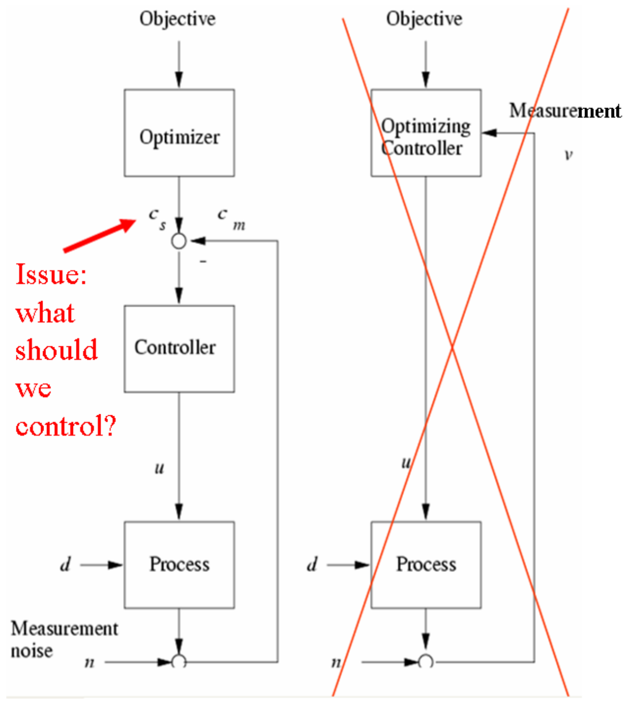

Figure 5: *Implementation of optimal operation: We want to replace open-loop "feedforward" optimization (right figure) by a self-optimizing feedback scheme (left figure) with constant setpoint c_s (such that the role of the "optimizer" can be minimized).*

In terms of selecting controlled variables for feedback control, there are two main cases:

1. Constrained optimization problems. Here the choice is easy: Control the active constraints! (Maarleveld and Rijnsdrop (1970)).

2. Unconstrained optimization problems. Here the choice is not so obvious. If possible, we would like to find some "self-optimizing" variable, which when kept constant indirectly gives optimal operation of the process.

An example of a constrained optimization problem is optimal operation of a 100m runner where the objective function is to minimize time, i.e., the cost is $J = T$. Here, optimal operation is limited by the maximum power (input constraint). Thus, we do not need much thinking to decide on a control policy: Just run as fast as possible (maximum input). Actually, keeping a maximum input does not require any feedback, but it does if there is an active output constraint, for example, if we want to minimize the driving time and are limited by the maximum speed limit.

An example of an unconstrained optimization problem is **marathon running**. Again the objective function is to minimize time, i.e., the cost is $J = T$, but clearly one should not simply just run as fast as possible. Thus, some more thinking is required to find a good control policy. One extreme, which requires a lot of systematic thinking, is to use a "feedforward" strategy where one designs a model-based optimizing controller for the marathon runner. However, this would probably require several PhD theses only to get the model, and it would require a computer and complex control system to implement the on-line optimizing controller.

A feedback solution is far simpler and in practice probably more effective. One candidate "magic" controlled variable (CV) is the speed, $c = v$. Indeed, running at constant speed is probably close to optimal for cases where the track is even and there is little wind. However, deciding on the appropriate setpoint for the speed may be difficult. An even better "magic" CV is probably the heart rate, $c = hr$. The optimal setpoint of the heart rate is almost constant, independent of the terrain and wind conditions, and one may even use similar optimal heart rates for different persons. Thus, heart rate seems to be a "self-optimizing" variable for marathon running.

Other "applications" of self-optimizing control include process control, biological processes, pizza baking, and running the central bank of a country, Skogestad (2004a).

How does one find self-optimizing variables in a systematic manner? This has been one of my main research interests over the last 10 years, and several methods have been developed.

In terms of analysis of a given choice of CVs (c), the crudest and most general approach is the "brute force approach", Skogestad (2000), where one evaluates the cost J when keeping constant CV setpoints for expected disturbances (d) and implementation errors (n). If one is also able to optimize the system and find the optimum (J_{opt}) then one can evaluate the loss $L = J - J_{opt}$, which gives insight into whether it is worth looking for a better policy. The problems with the "brute force approach" are that it gives no insight into what a good CV might be and that the number of possible CV combinations grows exponentially with the problem size (even worse, if one allows for variable combinations as CVs there are infinitely many possible strategies).

So what is a good self-optimizing variable (c)? To identify good candidates we may use the following four requirements (Skogestad (2000)):

Requirement 1. Its optimal value is insensitive to disturbances (so that the optimal variation Δc_{opt} is small).

Requirement 2. It is easy to measure and control accurately (so that the implementation error n is small).

Requirement 3. Its value is sensitive to changes in the manipulated variable u; that is, the gain G from u to c is large (so that even a large error in controlled variable c results in only a small error in u). Equivalently, the optimum should be "flat" with respect to the variable c.

Requirement 4. For cases with two or more controlled variables, the selected variables should not be closely correlated.

All four requirements should be satisfied. For example, for the marathon runner, the optimal heart rate is weakly dependent on the disturbances (requirement 1) and the heart rate is easy to measure (requirement 2). The heart rate is relatively sensitive to changes in power input (requirement 3). Requirement 4 does not apply since this is a problem with only one unconstrained input (the power).

For many problems the cost J depends mainly on the steady-state behavior and the four requirements can be combined into the maximum gain rule. According to Halvorsen et al. (2003), for a given choice of controlled variables (CVs, outputs) c, the worst-case loss for (small) disturbances and implementation errors can be estimated as:

$$L_{wc} = \frac{1}{2}\frac{1}{[\underline{\sigma}(\mathbf{G}')]^2} \tag{2}$$

Here $\mathbf{G}^{'} = \left(S\mathbf{G}\mathbf{J}_{uu}^{'-1/2}\right)$ is the scaled gain matrix, where $G = dc/du$ is the unscaled gain matrix, $S = diag\{1/span(c_i)\}$ is the output scaling, $\text{span}(c_i) = \Delta c_{opt,i+n_i}$ is the expected variation in c_i caused by its optimal variation due to disturbances ($\Delta c_{opt,i}$) plus its implementation error (n_i), and finally the input "scaling" $\mathbf{J}_{uu}$ is the Hessian matrix (second derivative) for the effect on the cost J of the unconstrained degrees of freedom u. So, we want to select controlled variables c that give the largest value of the minimum singular value $\underline{\sigma}(\mathbf{G}^{'})$ of the scaled gain matrix. In the scalar case this is simply the magnitude of the gain matrix, $\underline{\sigma}(\mathbf{G}^{'}) = |\mathbf{G}^{'}|$. Compared to the "brute force method", this method has the advantage of not requiring that one evaluates the cost for each choice of CVs. Instead, one needs to evaluate the gain matrix, G, and the scaling, span(c), which does not depend on the number of candidate CVs.

The use of the maximum gain rule requires a model, and the model may also be obtained experimentally. For example, for the marathon runner one would first need to analyze some optimal runs to find the expected variation $\Delta c_{opt,i}$ in the candidate CVs (speed, heart rate, etc.) with respect to the expected disturbances (wind, inclination). Next, one would need to change the power to find the (unscaled) gain for the same candidate CVs. The scaled gain could be evaluated, and one would prefer a CV with a high scaled gain.

For unconstrained quadratic optimization problems, which provide a local approximation of any real optimization problem, we have developed systematic methods for finding optimal *linear measurement combinations.* The simplest is the "*nullspace method*" which gives no loss for the case with no implementation error, provided we have as many measurements y as there are disturbances (d) plus inputs (u). The extension to the case with any number of measurements and to implementation error is given by Alstad et al. (2009).

We are also working on extending the method to nonlinear polynomial systems and to dynamic systems. One approach here is to make use of some property of the solution, such that the gradient with respect to the unconstrained degrees of freedom (u) should be zero at the optimal solution, $J_u = 0$ (e.g., Srinivasan et al. (2003)). Indeed, most of the methods mentioned above can be interpreted as variants of using the measurements for estimating the gradient J_u and then using feedback control to keep J_u close to zero. (However, to include implementation error, which is important for many problems, one may need to go back to the cost J, rather than working with the gradient J_u).

In summary, many real-life optimization problems can be turned into feedback control problems. The key is to find the right "magic" self-optimizing variable to control. The most obvious self-optimizing variable is the gradient J_u, because keeping $c = J_u$ at zero maintains operation optimal, irrespective of disturbances.

4 Stabilization Using Feedback: Anti-Slug Control of Multiphase Flow

Sometimes feedback control can really do magic, meaning that it can do things that could not be done otherwise. In particular, this applies to stabilization, which is only possible with feedback (model-based feedforward inversion will yield an unstable pole-zero cancellation and internal instability which eventually will make the system blow up). One example of stabilization by feedback is for multiphase flow in pipelines with a vertical section (riser). Here, the flow regime can often turn into an undesired sputtering flow regime known as riser-induced slug flow. For example, this happens quite frequently in the production of oil and gas. One can model this system for years and try all kinds of advanced estimation and model-based estimation schemes, and still get only limited improvement. However, as was first shown by Havre et al. (2000) a simple P or PI controller, based on using an exit valve to control the pressure at the bottom of the riser (Figure 6), can "magically" give steady non-slug flow at the same boundary conditions that otherwise gave slug flow (Figure 7).

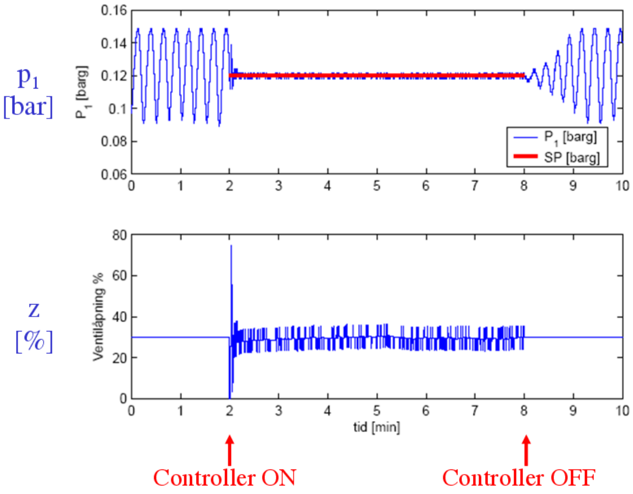

Figure 7: *Anti-slug control: Response with feedback strategy in Figure 6.*

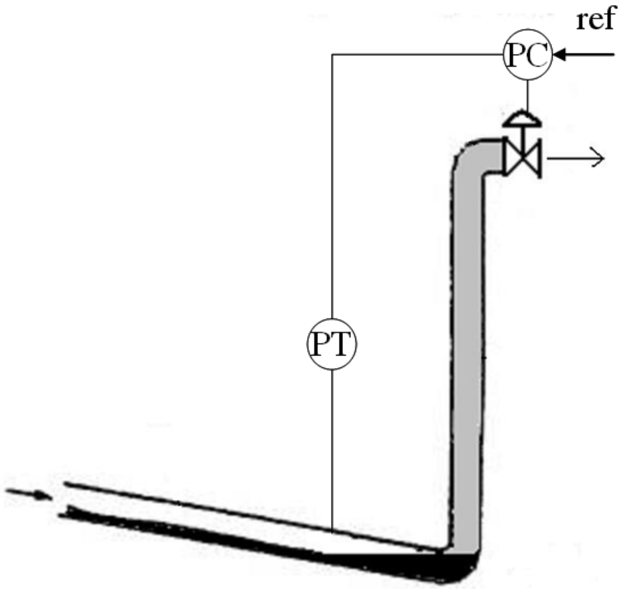

Figure 6: *Anti-slug control: Feedback control of pressure at bottom of riser.*

What is happening is that by feedback control we are able to stabilize a "naturally" unstable flow regime, that is, we are using feedback to extend the stability boundary of the desired solution. Actually, fluid flow experts find it hard to believe that this is possible. It would be a bit similar to using feedback to extend the stability boundary of laminar flow and thus avoid turbulence (although this is in practice very difficult because of the extremely short time and length scales involved in the switch from laminar to turbulent flow; one the other hand the time and length scales for the transition to slug flow are in the order of minutes and meters, respectively).

5 The Enemies of Feedback: Unstable (RHP) Zeros and "Effective Time Delay"

Let us continue the previous example. Controlling the pressure at the bottom of the riser works great and stabilizes the desired non-slug regime. However, the pressure at the bottom is often not measured, so it is tempting to switch and control the pressure at the top of the riser instead. However, this does not work because of the presence of unstable (RHP) zeros in the response from the valve position (input u) to the top pressure (output y), Storkaas and Skogestad (2007). This is due to the main fundamental limitation of feedback control: High feedback gains cannot be applied in the presence of time delays (Figure 3) and RHP-zeros because it results in closed-loop instability; see also Chapters 5 and 6 in Skogestad and Postlethwaite (2005).

Importantly, these fundamental limitations apply for any controller. Thus, the problem cannot be avoided with sophisticated model-based control. For example, it will not help to use a state estimator to estimate the bottom pressure from a top pressure measurement. When analyzing the estimated bottom pressure it may seem that it works, because the state estimator (Kalman filter) can be tuned to have fast response, but when we couple everything together there will be a hidden RHP pole-zero cancellation between the controller and the plant. The only option is to "change the system", for example, by introducing additional measurements or additional MVs (inputs).

6 Summary

High-gain feedback control is an extremely powerful tool.

1. For first-or second-order systems the phase lag can never exceed -180 degrees, so we can “*crank up the gain*” as much as we want. However, more generally the gain must be such that the closed-loop time constant is less than the effective time delay in the system, approximately.

2. Complex systems can be controlled by hierarchies (cascades) of single-input-single-output (SISO) control loops.

3. To achieve optimal operation (economically), it is important to select the right controlled variable (CV). First, any active constraints should be controlled. Second, for remaining unconstrained degrees of freedom one should search for variables that achieve “self-optimizing control”.

4. Stabilizing feedback control can make new things possible (for example, anti-slug control).

References

Alstad, V., Skogestad, S., and Hori, E. Optimal measurement combinations as controlled variables. *Journal of Process Control*, 2009. 19:138–148. doi:10.1016/j.jprocont.2008.01.002.

Halvorsen, I., Skogestad, S., Morud, J., and Alstad, V. Optimal selection of controlled variables. *Ind. Eng. Chem. Res.*, 2003. 42(14):3273–3284. doi:10.1021/ie020833t.

Havre, K., Stornes, K., and Stray, H. Taming slug flow in pipelines. *ABB Review*, 2000. 4:55–63.

Kline, R. Harold Black and the negative-feedback amplifier. *IEEE Control Systems Magazine*, 1993. 13(4):82–85. doi:10.1109/37.229565.

Maarleveld, A. and Rijnsdrop, J. Constraint control of distillation columns. *Automatica*, 1970. 6(1):51–58. doi:10.1016/0005-1098(70)90074-9.

Morari, M., Stephanopoulos, G., and Arkun, Y. Studies in the synthesis of control structures for chemical processes, part i. *AIChE J.*, 1980. 26(2):220–232.

Skogestad, S. Plantwide control: the search for the self-optimizing control structure. *J. Proc. Control*, 2000. 10:487–507. doi:10.1016/S0959-1524(00)00023-8.

Skogestad, S. Near-optimal operation by self-optimizing control: From process control to marathon running and business systems. *Computers and Chemical Engineering*, 2004a. 29(1):127–137. doi:10.1016/S0959-1524(00)00023-8.

Skogestad, S. Simple rules for model reduction and PID controller tuning. *Modeling, Identification and Control*, 2004b. 25(2):85–120. doi:10.4173/mic.2004.2.2.

Skogestad, S. and Postlethwaite, I. *Multivariable feedback control - analysis and design*. Wiley, Chichester, 2005. 572 pp (1996); 2nd Edition 2005.

Srinivasan, B., Bonvin, D., and Visser, E. Dynamic optimization of batch processes - iii. role of measurements in handling uncertainty. *Comp. Chem. Eng.*, 2003. 27(1):27–44. doi:10.1016/S0098-1354(02)00117-5.

Storkaas, E. and Skogestad, S. Controllability analysis of two-phase pipeline-riser systems at riser slugging conditions. *Control Engineering Practice*, 2007. 15:567–581. doi:10.1016/j.conengprac.2006.10.007.

Wiener, N. *Cybernetics : or Control and communication in the animal and the machine.* Technology Press, Cambridge, MA, 1948. 194 pp.