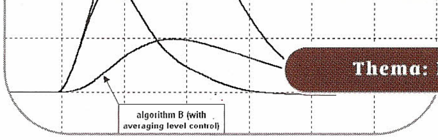


Better Process Control: An

Improved Training Of Process Control Engineers

Myke King

Mr. M. King is Director of Whitehouse Consulting, Redway House, East Lane, Merstone, Isle of Wight, PO30 3DJ, United Kingdom.
E-mail: mykeking@compuserve.com
tel.: +44 (0)1983 529931

Distributed control systems (DCS) have been in use for over 30 years. Yet the majority of the process industry does not properly understand many of their features. On almost every process there exists the opportunity to improve basic control and thus to increase process profitability. The key to this is to provide better training and design tools for the engineers that implement and support such systems. The problem is that most of the available training does not properly address practical issues and most of the design tools fail to deliver effective controllers.

Control engineers are of course aware that good basic controllers will improve process operation. The process will recover quickly from disturbances – maintaining product quality constant and enabling plant constraints to be approached more closely. Without good control it may be necessary to operate well within product specifications to ensure that disturbances do not cause off-grade production. This 'giveaway' incurs increased operating costs and reduced product yield. Similarly it may be necessary to operate well away from other limitations to ensure that critical equipment constraints are never violated, thus under-utilising plant capacity. It is common for improved control to increase process revenue by several per cent. While the process industry is generally good at ensuring that the *hardware* of the instrumentation is sound, problems arise in configuration both from using the wrong control algorithm and from implementing the wrong tuning.

Choice of Control Algorithm

There is not always an awareness that the DCS offers a wide range of control algorithms. Or, if the engineer has read the manual thoroughly, he or she may not understand why there are so many choices. Not fully appreciating the benefit of each algorithm the engineer will select the default or the one that most closely matches his or her understanding of proportional-integral-derivative (PID) control. In almost every circumstance this will lead to the wrong selection. The engineer will have missed the opportunity to significantly improve the response of the process to disturbances - needlessly extending (typically by a factor of three!) the time that the unit takes to recover.

Consider subjecting the process in Figure 1 to a reduction in feed rate caused by the operator reducing the set point of FC1. The two heater outlet temperature trends are in response to the same disturbance to feed rate. Each algorithm was tuned to give virtually the same response to temperature set-point changes. Algorithm A, chosen by most, is the conventional version of PID control. Care need be taken in understanding how the DCS vendor has converted the algorithm to the discrete form but, in continuous form, it can be represented as:

$$M = K_c\left[E + \frac{1}{T_i}\int E.dt + T_d\frac{dPV}{dt}\right]$$

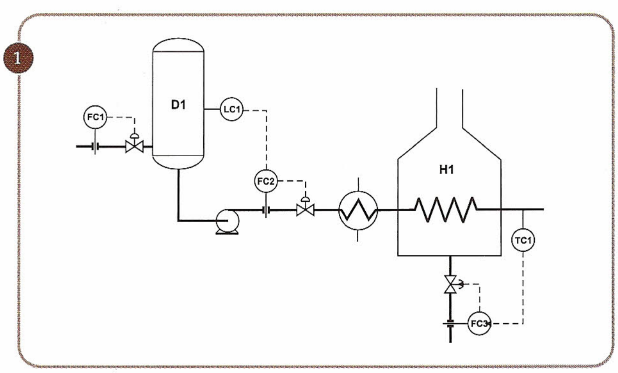


Fig. 1 Process flow diagram

Opportunity To Increase Profitability

and algorithm B as:

$$M = K_c\left[PV + \frac{1}{T_i}\int E.dt + T_d\frac{dPV}{dt}\right]$$

Controller Tuning

The next challenge is ensuring the controllers are properly tuned. Many are not. Probably the most common example is ‘averaging’ level control. Rather than the controller being tuned for a fast return to set point, this permits the vessel level to vary (within alarm limits), thus minimising changes to the downstream flow. There are many situations where level controllers can exploit surge capacity within the process. This reduces downstream disturbances - often giving remarkable improvements to process stability. Many control engineers appreciate this, but few properly calculate and implement the correct tuning.

Figure 2 shows the effect of changing from tight to averaging control. Tuning constants for tight control can be derived from:

$$K_c = \frac{0.8V}{F.ts} \qquad T_i = \frac{V}{12.5f}$$

and for averaging control from:

$$K_c = \frac{80f}{F.d} \qquad T_i = \frac{V.d}{12.5f}$$

Care is needed in ensuring that the engineering units used for each parameter are consistent with the way in which the PID controller is coded. In the example cited, the corrective action taken by the averaging level controller is almost 3000 times slower than that taken by the tight level controller.

Further, few understand the adaptive nature of the ‘error squared’ algorithm, i.e.

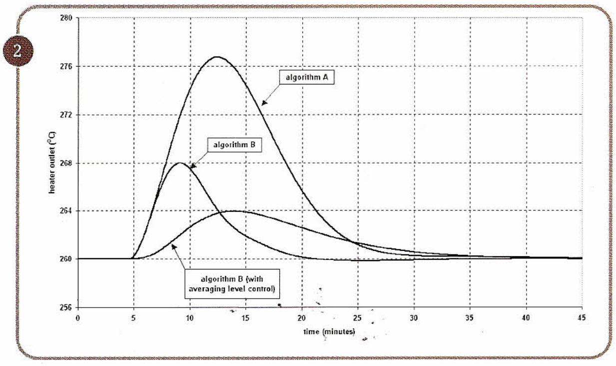
$$M = K_c|E|\left[PV + \frac{1}{T_i}\int E.dt + T_d\frac{dPV}{dt}\right]$$

Used normally only for averaging control, tuning can be similarly calculated. But the formulae can vary greatly – depending on how the vendor has coded the algorithm.

Another example is the use (or not!) of derivative action. It is seen as additional complexity, making the controller more difficult to tune. Many believe that derivative action should only be applied to temperature controllers. Derivative control is highly beneficial if the process dead time (θ) is large compared to the process lag (t). Its anticipatory nature helps the controller respond much more quickly to disturbances. Many temperature controllers have such dynamics, but not all. The use of derivative action here would bring little benefit and could cause stability problems. Similarly there will be many other controllers where θ/τ is much greater than unity; ignoring the benefit of derivative action will greatly extend the time taken by the process takes to recover from disturbances. Figure 3 is based on the use of the conventional PID controller on a process that has a lag of one minute. Tuning constants were determined to minimise ITAE (integral over time of absolute error) while avoiding excessive controller output overshoot. It illustrates the impact that dead time (θ) has on the choice of T_d.

Multivariable control (MVC) packages are becoming increasing common. These adjust the set points of the basic controllers to permit further increases in process profitability by more closely approaching operating constraints. As part of MVC design, a major ‘step-testing’ exercise must be undertaken to obtain the process dynamics. On complex processes this can take several weeks, often working shifts to cover round-the-clock testing. It is not something that the plant

2



3

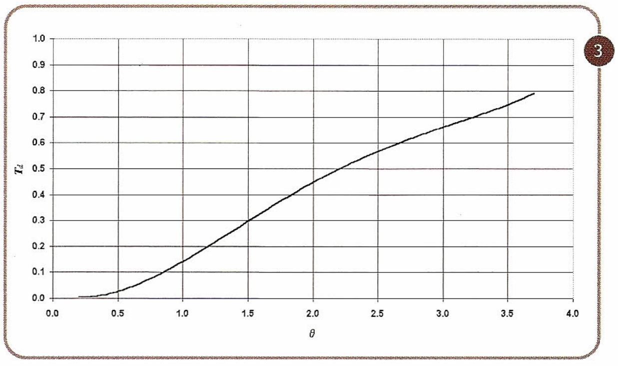
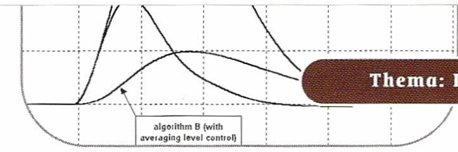


Fig. 2 Response to changes in feed rate

Fig. 3 Effect of process deadtime



owner would willingly undertake more often than absolutely necessary. Changing a basic control algorithm, or re-tuning one, changes the process dynamics. Thus, once step testing has started, the engineer has effectively committed the site to retaining poor basic controllers at least until the next major process modification, when step testing would have to be repeated in any case.

Conclusions

The first priority in addressing these problems is training. However care should be taken in selecting the best provider. DCS vendors are generally poor at explaining why their systems support so many different algorithms. MVC vendors provide good product-specific courses covering their technology but usually try not to become too involved with the *basic* controls. Academic institutions provide a broad range of courses but usually only address theoretical issues using mathematical techniques that have little relevance to the process industry.

The control engineer will need an effective tuning aid. There is a bewildering array of methods. Almost every month a journal publishes a new one or a new product is announced. With only a few exceptions these methods are usually flawed. Rarely do they account for the variety of algorithm types and usually they apply incorrect tuning criteria. In selecting a method, care should be taken in ensuring that it handles both self-regulating and non-self-regulating (integrating) processes. The version of PID algorithm it assumes should be identical to that configured by the DCS vendor. The method should be based on discrete, rather than continuous, control. And it should allow the user to specify the tuning criteria.

Finally the control engineer should be given time to address the basic control problems. Effort should be focussed on those controllers where a noticeable improvement is possible and adequate time should be allowed before beginning any installation of MVC.

Definition of Symbols

d	maximum deviation from set point
E	error between PV and set point
F	instrument range of manipulated flow controller
f	normally expected flow disturbance
K_c	controller gain
M	controller output
PV	process value (measurement)
t	time
T_d	derivative action time
T_i	integral action time
ts	controller scan interval
V	vessel volume between 0 and 100% of level indication