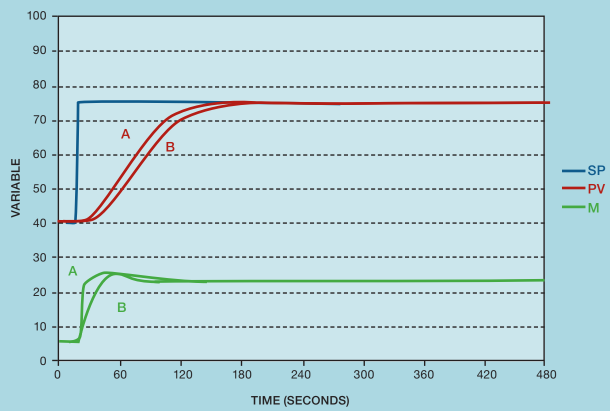
FEATURE SERIES: PRACTICAL PROCESS CONTROL

5: The I-PD Algorithm

Myke King continues his detailed series on process control, seeking to inspire chemical engineers to exploit untapped opportunities for improvement

PROBABLY the most misunderstood and underused control algorithm is the I-PD version. Recalling the last issue, it differs from the PI-D in that proportional action is based on the PV (process value) rather than the error. Almost universally, control system vendors, if they do document its purpose, suggest it should be used when only a slow response to set-point (SP) changes is required. However, this can be delivered by any control algorithm, simply by adjusting the tuning. Figure 1 shows, as curve A, a well-tuned controller responding to an SP change. Switching from PI-D to I-PD results in the response shown as curve B. It does, indeed respond more slowly; the algorithm no longer generates the proportional kick, relying solely on the integral action. One can see why the algorithm has the reputation for being slow. But the key issue is that different algorithms require different tuning parameters. It is unreasonable to expect the I-PD to perform well, using tuning designed for the PI-D algorithm.

Figure 2 – Retuning I-PD for SP changes

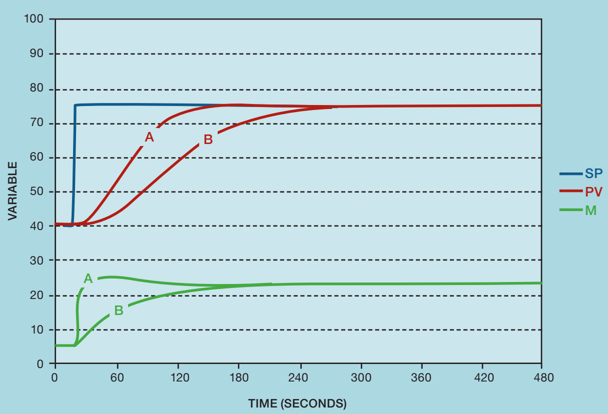


BENEFIT OF RE-TUNING

Figure 2 again compares the two algorithms but with the I-PD properly tuned. The lack of proportional kick has been largely compensated for by increasing the controller gain. This hasn't been achieved by overly aggressive tuning; the manipulated variable (MV) overshoot is the same for both controllers. While the PI-D slightly outperforms the I-PD, it is unlikely this would be noticeable on a real process.

Given the comparable performance still begs the question as to the benefit of the modified algorithm. To understand this, instead of testing the controller with a SP change, we subject it to a *load change*. This is simply a process disturbance. In the example of our fired heater (*Issue 981*), this might be an increase in the feed rate. Figure 3 shows the open loop response. With no controller in place, the fuel flow would remain constant and the outlet temperature will reduce to a new steady state. Curve A shows the response of the PI-D controller. Had we not also displayed curve B, the engineer would have been quite content with its behaviour. But we can now see that the I-PD controller returns the process to SP in about half the time. It also reduces the maximum deviation by more than 50%. If the temperature were the main influence on finished product quality, then we

Figure 1 -Switching algorithm without retuning



Probably the most misunderstood and underused control algorithm is the I-PD version

Figure 3 – Impact of tuning on load change

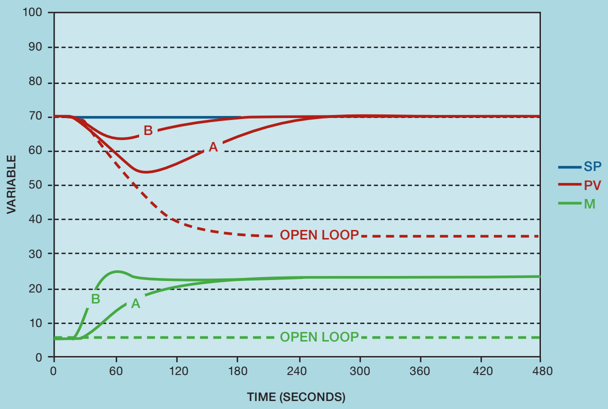
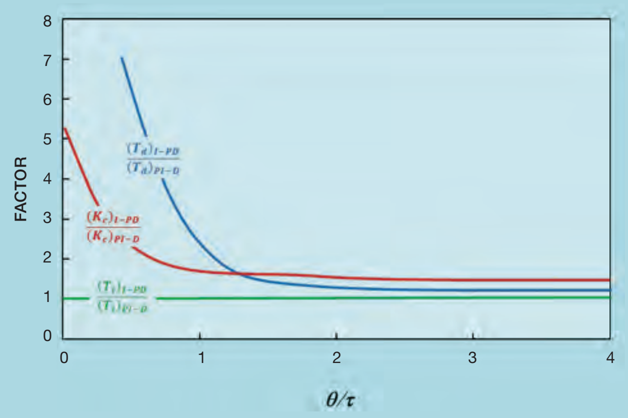


Figure 5 – Effect on controller tuning



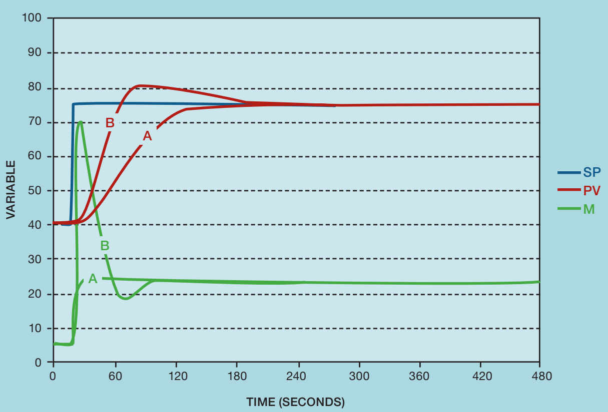
will have reduced off-grade production by about 80%. Again, this has not been achieved with aggressive tuning; we stay within the MV overshoot limit.

We should emphasise at this point that it is not the change of algorithm that has brought about the improvement. If the SP remains constant, the I-PD and P-ID algorithms perform identically. What has brought about the improvement is the change of tuning. We could have implemented the new tuning in the original controller and seen the same improvement. The change of algorithm is necessary to handle SP changes. This is illustrated in Figure 4. Retaining the P-ID algorithm results in a dangerously aggressive response to the SP change, with MV overshoot approaching 300%.

To further emphasise the improvement we can examine precisely just how big the changes to controller tuning can be. Figure 5 shows the factor by which the PID tuning parameters can be increased. At the very least, if the deadtime-to-lag ratio (θ/τ) is large, there is potential to increase controller gain (K_c) by 50%. But θ/τ is usually less than 0.5, so it is common that it can be increased threefold. Integral time (T_i) changes little but derivative action (T_d) can also be substantially increased.

It is difficult to overstate the importance of switching to the I-PD algorithm. But firstly we need to consider *cascade* controllers. In our example, temperature is cascaded to the flow controller; it adjusts the flow SP. (In a future article we will cover the advantages of this approach over the alternative of the temperature controller directly manipulating the fuel valve.) The temperature is the *primary* (or *master*) controller, while the flow is the *secondary* (or *slave*) controller. The temperature controller will largely need to deal with load changes. We've seen that one source of such a disturbance is a change in feed rate. But there are many others – including

Figure 4 – Why we need the I-PD algorithm



changes in heater inlet temperature, disturbances to the fuel system (such as header pressure and fuel gas heating value) and variation in heater efficiency (caused by changes to the air-to-fuel ratio). Load disturbances are likely to be far more frequent than SP changes. In many processes the heater temperature SP stays constant for many days.

AUTOMATIC ALGORITHM SWITCHING

The flow controller, however will experience far more SP changes. Any disturbance affecting the temperature will be corrected by the controller adjusting the fuel flow SP. Potentially the SP could be changed at every scan interval. We've seen that the PI-D algorithm marginally outperforms the I-PD algorithm for SP changes. Logically it would seem that it should be used for the secondary of a cascade, while the I-PD be chosen for the primary. However, this raises a couple of issues. Firstly the secondary will, on occasions, operate as a standalone controller. This will be particularly during starting up and shutting down the process, and at any other time when the temperature controller can't be used. Under these circumstances we might want to switch to the I-PD algorithm. Indeed, two control system vendors (ABB and Yokogawa) offer this as an automatic feature. We recall (*Issue 982*) that, for the ABB system, the parameter BETA is set by the engineer to select the algorithm. In fact there are two such parameters (BETA CONT and BETA DISC). BETA CONT is used by the secondary when in cascade; BETA DISC when not. By setting BETA DISC to 0 and BETA CONT to 1, the algorithm will switch from P-ID to I-PD. However it is important that this feature is **not** configured. It reflects the system vendor's lack of understanding that the algorithms require very different tuning. While the change of tuning could be automated, maintaining an infrequently used second set of tuning constants would be troublesome. So, in the case of ABB, both BETA parameters should be set to the same value – preferably 0. Similarly, in the Yokogawa system, the option described as *automatic determination* should **not** be selected.

A secondary issue is one of standardisation. The more consistency there is, the less the potential for an engineer to make mistakes. The I-PD should be used from both the primary and the secondary. As we saw in Figure 2, this will cause negligible degradation in the response to SP changes.

This issue also applies to the implementation of multivariable predictive control (MPC). When commissioned, what were primary controllers will become the secondaries of the MPC. If automatic switching is in place then step-testing to design the MPC will have been performed with the I-PD algorithm in place but, when commissioned, the MPC will be manipulating controllers with the P-ID control. Without the necessary change of tuning, the dynamic behavior of the process will be different from that used for MPC design. Indeed, as Figure 4 shows, the moves made by the MPC are likely to cause excessive MV overshoot.

The more consistency there is, the less the potential for an engineer to make mistakes

IMPLEMENTING TUNING CHANGES

The likelihood is that the process has been commissioned with all the controllers chosen as PI or PI-D. We've shown that switching to I-PD requires a substantial increase in controller gain. Properly engineered, this should not cause any problems, but a cautious approach is always wise. One might therefore consider implementing the new tuning incrementally, taking two or three steps.

To switch the algorithm in all the controllers on a process would be a major exercise. Properly engineering them from the dynamics obtained from step-testing would be impractically time-consuming. Instead, we modify only those controllers where the improvement would be noticeable. For example, if re-engineering a flow controller halves its recovery time from 10 seconds to five seconds, this would likely go unnoticed. However, doing the same for a heater outlet temperature, halving a 20 minute recovery would have a noticeably beneficial impact. But remember, if the process has MPC in place, changing the regulatory control algorithm and re-tuning will affect the overall process dynamics. The change will certainly degrade the performance of the MPC, potentially making it unstable. Changes to the regulatory controls should be part of the work completed before step-testing for MPC. And of course, for new processes, all controllers should be configured as I-PD. ■

NEXT ISSUE

The majority of controllers are configured as PI rather than PID. There are a variety of reasons for this. One is that derivative action amplifies noise. The other is that a three-dimensional search by trial and error is substantially more difficult than a two-dimensional one. In much the same way as engineers are missing opportunities by not using the I-PD algorithm, the same is true of derivative action. The next article makes the case for its inclusion and shows how to resolve the problems it can create.

Myke King CEng FIChemE is director of Whitehouse Consulting, an independent advisor covering all aspects of process control. The topics featured in this series are covered in greater detail in his book Process Control – A Practical Approach, published by Wiley in 2016.