PV-15

PY-15

Control Loop Tuning

Back to Basics

Implement an Effective Loop Tuning Strategy

Tim Olsen
Norman Ito
Emerson Process Management

Use these loop tuning guidelines
to improve process control
and enhance product uniformity.

Many undergraduate chemical engineering programs teach the Ziegler-Nichols tuning methods, developed by John G. Ziegler and Nathaniel B. Nichols in the 1940s *(1)*. This tuning method has large controller gain and short integral time, and sometimes creates process oscillations, which are not good for most chemical engineering applications. As a result, many process control engineers revert to tune-by-feel techniques.

In tuning by feel, individual control loops are tuned as fast as possible without disrupting upstream and downstream control loops. However, by tuning only individual loops, the overall process performance and the ability to recover from disturbances are reduced. When a process has been tuned by feel, console operators often need to put controllers into manual operation to settle the process down after a major process disturbance.

Automatic process control attenuates disturbances and maintains control of the process variables to match desired setpoints, and appropriate tuning enables this capability. This article describes process controller basics, and details a step-by-step process for control loop tuning.

Process control basics

In a control loop, the controller manipulates the process in order to maintain conditions at setpoint (SP) based on changing information about one or more process variables (PVs) (Figure 1). The process is often affected by upstream and downstream disturbances.
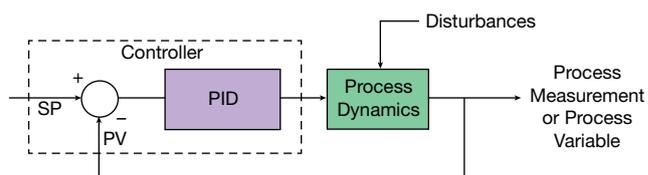
Before attempting any tuning, it is important to understand the process dynamics. The process dynamics are determined by performing step tests. The proportional-integral-

derivative (PID) equations that calculate tuning parameters incorporate both process dynamics and controller terms. A PID equation can be interpreted as follows:

• The proportional (P) term adjusts the controller output in proportion to the difference between the setpoint and the value of the process variable (referred to as setpoint error). Increasing the proportional gain increases the loop response, but too big a gain can cause the loop to oscillate.

• The integral (I) term sums the setpoint error over time and slowly adjusts the controller output until the error is zero. Integral action aims to eliminate any setpoint error, but too much integral action can cause the integral term to ramp too quickly relative to the process variable (windup), which can result in an oscillatory loop response.

• The derivative (D) term decreases the controller output when the rate of change in the process variable is increasing rapidly (or vice versa). The derivative term adds stability, but is sensitive to noise. It can make a loop unstable if too much derivative action is used in the wrong situations (*e.g.,* noisy process variable, slow controller execution

▲ **Figure 1.** In this control loop, a controller maintains conditions at setpoint (SP) based on changing information about the process variable (PV).

rate, pure deadtime process, etc.).

There are three forms of PID equations that may be implemented based on the host distributed control system (DCS) being used. The series (classical) form was widely used in pneumatic and electronic controllers, but many modern digital control systems today use either the standard form or include it as an option.

Standard PID equation. The standard (sometimes called ideal or ISA) form is often referred to as noninteracting, as the integral and derivative terms do not influence each other. The standard form is somewhat more complex to understand than the other forms, but it is also more flexible in that it can cancel oscillatory open-loop dynamics.

Series PID equation. In the series (classical, interacting) form, the integral and derivative terms interact. Tuning a second-order system is simple because the larger time constant sets the integral time and the smaller time constant determines the derivative time. If the derivative term is not used, the series and standard equation forms are identical.

Parallel PID equation. The structure of the parallel form is the easiest to understand mathematically, as all three tuning parameters are independent. It is often the form presented in academic control courses. The tuning parameters, however, have a more theoretical mathematical interpretation, which makes the terms somewhat more difficult to visualize (or interpret).

Achieving a desired controller response to a setpoint change or a disturbance may require that different tuning constants be installed depending on the form of the PID equation. The PI tuning rules presented in this article, for a first-order-plus-deadtime process, are based on the standard or series form of the PID algorithm.

Step 1. Understand the process and operating objectives

Before starting to tune, you need to have a full understanding of the process and its operating objectives.

In the simple process diagram shown in Figure 2, a flash drum at 150 psig feeds vapor through a 16-in. line into a large distillation column at less than 2 psig. Both vessels have pressure controllers.
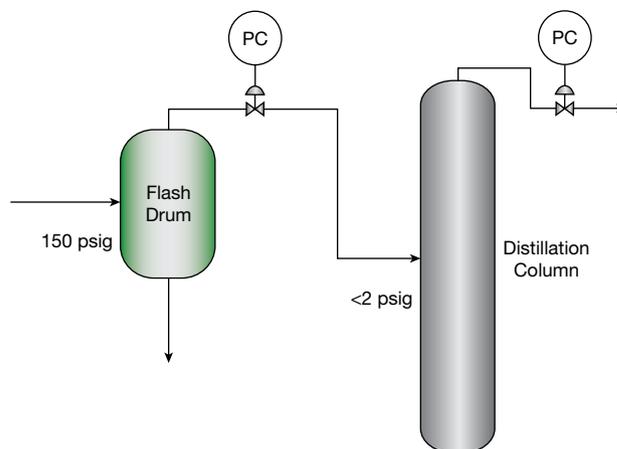
At this facility, each pressure controller was tuned individually, which resulted in process instability in the distillation column and reduced product quality. The flash drum was tuned very aggressively at a standard deviation of 0.1 psi from the 150-psig setpoint, and the flowrate of the flash drum vapors significantly disrupted the distillation column performance. In this scenario, any disturbance that causes deviation in the 150-psig setpoint is quickly adjusted back to setpoint — this means the feed drum disturbance goes right to the distillation column.

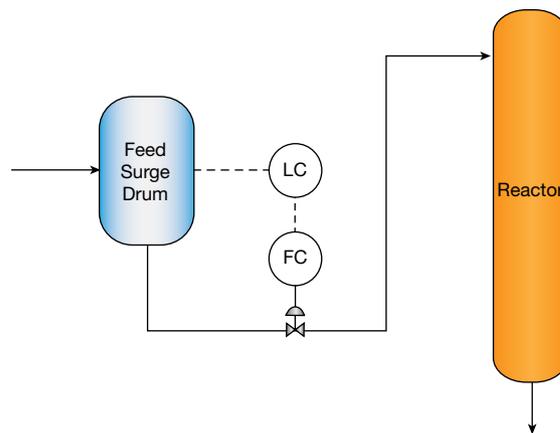The overall objective of this process is to effectively

control the distillation column to obtain the desired product quality and yield. The objective is not tight control of the flash drum pressure, especially if it detrimentally affects the downstream distillation performance. Therefore, the distillation column's pressure controller should be more aggressively tuned than the flash drum's.

In another example, a feed surge drum supplies a reactor (Figure 3). If loops were tuned individually, the feed surge drum level would have the same priority as the reactor inlet flow. Any disturbances in flow to the feed surge drum would quickly adjust the drum level (to maintain it at the setpoint), causing a sudden change in flow to the reactor. Tuning each loop individually gives equal priority to both drum level and reactor inlet flow. Equal priority in individual control loops does not necessarily result in the best process performance.

Article continues on next page

▲ **Figure 2.** A flash drum at 150 psig feeds vapor into a large distillation column at less than 2 psig. Both vessels have pressure controllers.

▲ **Figure 3.** A feed surge drum supplies a reactor. The surge drum's level influences the flowrate to the reactor.

Understanding the process interactions and control objectives is the first step to efficient loop tuning, and leads to the next step.

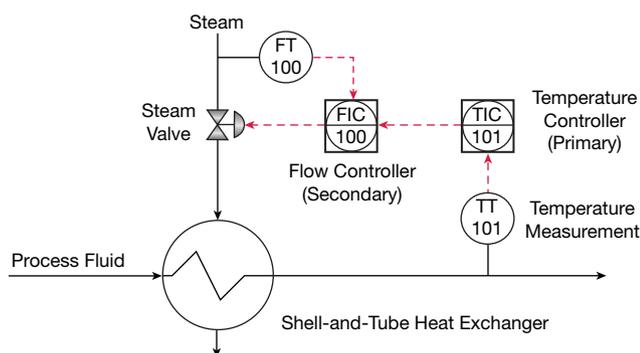### Step 2. Prioritize the control loops

Based on the information gathered in Step 1, prioritize the control loops to determine which should execute fast and which should execute slow. Process control loops that exhibit strong interactions will fight each other if given equal priority in tuning.

In the flash drum and distillation column example in Figure 2, maintaining the flash drum at 150 psig is not critical to the separation. The flash drum can be used as a variability buffer to absorb disturbances from the reactors. Lowering the priority of the drum pressure control loop minimizes disturbances to the distillation column.

In the feed surge drum and reactor example in Figure 3, priority should be given to the reactor flow and the feed surge drum level allowed to fluctuate to absorb upstream disturbances. The drum level would have a slower response speed than the reactor feed flow control. Although the reactor flow will still increase or decrease to ensure that the surge drum is not overfilled or dry, the change will be gradual, avoiding sudden changes in flow to the reactor and potential reduction in yield.

In another example, a cascade controller with a process fluid temperature setpoint has an inner loop that controls steam flow and an outer loop that controls process fluid temperature (Figure 4). The recommended practice is to tune the inner control loop so it responds 5–10 times faster than the outer control loop if possible, to avoid direct interacting conflict. It is difficult to accomplish this speed of response using Ziegler-Nichols or tune-by-feel tuning.

The solution for this dilemma is to implement a tuning methodology such as lambda tuning, in which the control loop speed of response can be selected as a tuning parameter.

The inner control loop can be tuned 5–10 times faster than the outer loop to quickly compensate for changes in steam flow. Because the setpoint of the inner loop is a function of the outer loop, they will always interact. If the inner and the outer loops have equal priority, tuning will cause instability or oscillations because the inner loop cannot track the setpoint of the outer loop.
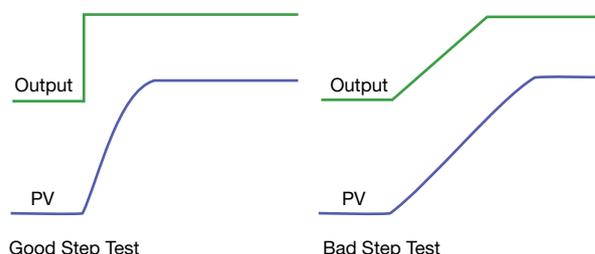
Most surge vessels can utilize their capacity as variability buffers to absorb upstream disturbances, thereby minimizing the impact to downstream processes. This can be a culture change for console operators who like to see flat-line displays of process variables matching the setpoint — *i.e.,* tuned aggressively to maintain tight control. The levels in surge drums, distillation column bottoms, and overhead drums can fluctuate from 40% to 60% — around the 50% setpoint — to maintain steady flow downstream, with gradual changes to rebalance the process flow. However, in some cases these vessels must have minimal variability, and it is important to identify these constraints in Step 1 before tuning.

Step 3. Perform step testing

Each control loop must be step tested. There are two objectives to step testing: To identify hardware issues (such as a sticky control valve), and to understand the process dynamics.

To perform a step test, place the controller in manual mode, and change the output by a small percentage. The size of the output step should be large enough to allow you to observe the process variable response (above any noise that may be present), yet small enough to minimize process upsets (Figure 5). In a good step test, the output is immediate, and the deadtime and time constant can be determined. A bad step test will produce a ramped output, and the process response will not be representative of the true process dynamics that are needed to calculate the tuning parameters.

Before you execute a step test, make sure that you understand the process and the expected changes to prevent unwanted process upsets. For example, when performing

▲ **Figure 4.** A shell-and-tube heat exchanger has a cascade controller with a process fluid temperature setpoint. The inner loop controls steam flow and the outer loop controls process fluid temperature.
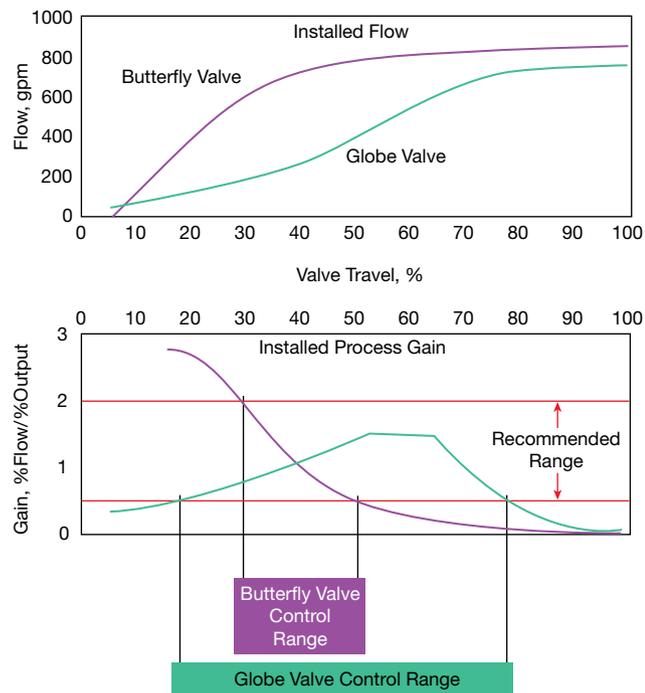

▲ **Figure 5.** When performing a step test, the size of the output step should be large enough to allow you to observe the process variable (PV) response, yet small enough to minimize process upsets.

a step test in a refinery, do not reduce the hydrogen quench flow because this will cause a temperature runaway in the hydrocracker reactor; start by increasing the hydrogen flow, and then reduce it back to its standard rate.

Begin with small steps, such as 0.25%, 0.50%, or 1%, depending on the sensitivity of the process. If you do not see a change in the process variable, continue making step changes until a response is evident. The process response, including the deadtime, process noise, and actuator non-linearities, will determine the size of the step and the timing between steps. It is not uncommon for certain control loops to require an output change in excess of 2% before a response from the control valve is seen.

Underperforming field devices add process constraints that cannot be tuned out or compensated for with advanced process control. During the step testing, be aware of the type of control valve and its typical operating point. A valve that is operating at or near its fully open or fully closed position will be difficult to control and will underperform no matter what the tuning parameters. If this is the case, investigate what has changed from the original design that has caused the valve to operate outside of its normal operating range.

Be aware of the process gain (%flow/%output) when performing a step test. For example, a globe valve has a wider range of recommended process gain than a butterfly valve (Figure 6). If a 1% change in the controller output is made, the valve position will change by 1%, but the flowrate may not change by 1%. At around 38% open for both types of valves, a 1% change in output results in a 1% change in flow (for an installed process gain of 1). However, at around 50% open for both types of valves, a 1% change in output results in a 0.5% change in flow for the butterfly valve but a 1.5% change in flow for the globe valve. Outside the recommended 0.5% to 2.0% output change, the installed process gain makes these loops much more difficult to tune. This is why it is necessary to select a control valve of the size and type appropriate for the system's operating flows. This also explains why different operating conditions (*e.g.*, temporary turndown conditions) can result in automatic control becoming unstable.
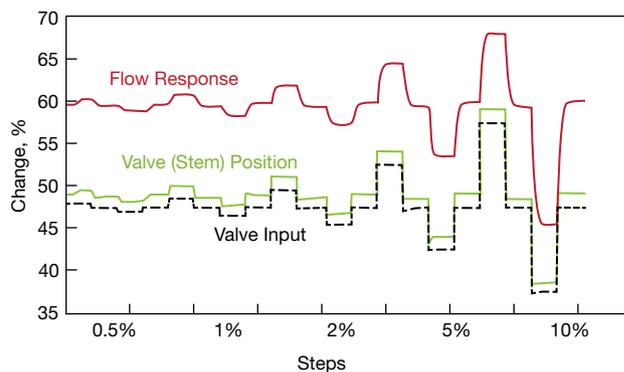
Step 4. Resolve hardware issues

As you identify hardware problems such as a sticky control valve, isolate the control valve and make repairs, or plan the repairs during the next turnaround. If you are unable to make the required hardware improvements, proceeding with tuning will be more challenging, with no guarantee of tuning success.

Smart positioners have made significant advances over the past decades and improve control valve performance for better response and positioning (Figure 7). Positioners are most commonly used to move a control valve to a specified position so that a process parameter (*e.g.,* flow, pressure, or temperature) remains at the desired value. They provide better resolution for small moves and changes.

Step 5. Repeat the step tests of loops that had hardware fixes

Now that the control valves are fully functioning, follow the same step test procedures in Step 3 to verify the process dynamics.
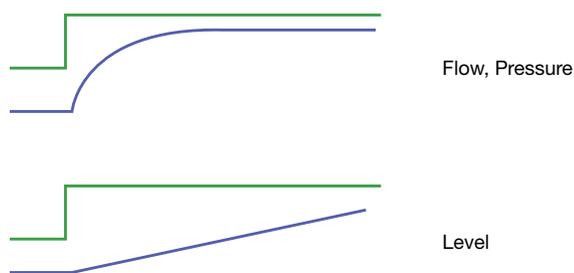
Article continues on next page

▲ **Figure 6.** A globe valve has a wider range of recommended process gain than a butterfly valve.

▲ **Figure 7.** Smart positioners improve valve performance for better response and positioning. They can more accurately control small changes.
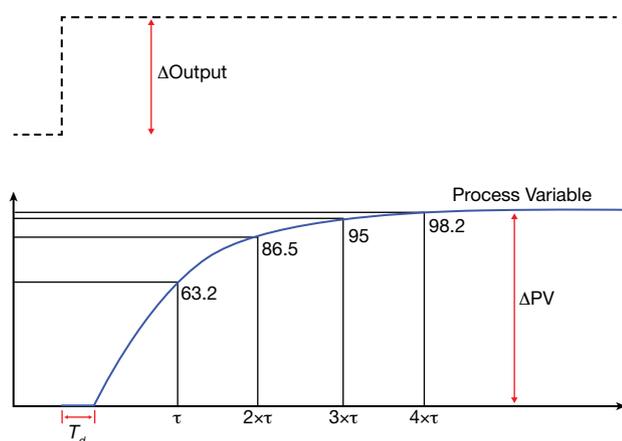
As shown in Figure 8, different processes respond differently to a step test. Flow and pressure controllers experience the most common type of response — first-order plus deadtime — whereas a level controller has an integrating response. There are several types of responses that could be observed, such as first-order plus deadtime, integrating, second-order, and inverse response, among others.

First-order-plus-deadtime response example. The process dynamics information obtained from a step test on a process with a first-order-plus-deadtime response includes the deadtime, steady-state process gain, and time constant. Typically these parameters are an average of several step tests around the normal operating range.

In Figure 9, the steady-state gain can be calculated by dividing the difference in the measured process variable before and after the step test by the output change. Normally, the gain is converted to units of %PV/%output because most controllers calculate tuning parameters based on a setpoint error in percent of span units (rather than engineering units), which results in a controller output in

▲ **Figure 8.** Flow and pressure controllers experience the most common type of step test response: first-order plus deadtime. A level controller has an integrating response.

▲ **Figure 9.** In a first-order-plus-deadtime response, the deadtime (T_d) is the amount of time after the step test is made until the process begins to respond. The time it takes the process to settle is approximately 4τ.

dimensionless units. The time constant tau (τ) is the time required to reach 63.2% of the final process change. The deadtime (T_d) is the amount of time after the step test is made until the process begins to respond.

Step 6. Calculate tuning parameters

Once you understand the process dynamics and control loop priorities, you can use model-based tuning, such as lambda tuning, to set the speed of response for each controller based on operating objectives and priorities.

Lambda tuning. The tuning examples in this article utilize lambda tuning. Lambda tuning refers to all tuning methods in which the control loop speed of response can be selected as a tuning parameter; the closed-loop time constant is referred to as lambda (λ).

Lambda tuning is used widely in the pulp and paper industry, where there is a strong connection between paper uniformity and manufacturing efficiency, and control-loop interactions with upstream hydraulics. Paper can be judged by its physical characteristics; therefore, all upstream variability is captured in the final product. When it was introduced by Dahlin in 1968 *(2),* lambda tuning offered a new way of coordinating the tuning of the paper mill loops to gain improved process stability along with a uniform product. This tuning technique is relatively new in other industries.

A recommended starting point for the value of lambda is three times the larger of tau or deadtime. That way, the time for the loop to reach the setpoint is approximately four times the selected value of lambda.

Tuning parameters established for normal operation may not work as effectively during periods of turndown. If turndown continues for a long time, consider re-evaluating the required tuning parameters for the new operating conditions.

Self-regulating process example. In this example, a step test was performed on a system with a first-order-plus-deadtime response and a standard PI controller. The test resulted in a process gain of 2%span/%output, a deadtime of 1.5 s, and a time constant of 4 s. The standard PI controller equation is *(3, 4):*

$$K_c = \left(\frac{1}{K_p} \right) \times \frac{T_r}{\lambda + T_d} \tag{1}$$

where T_r is the integral time (which is equal to τ, the time constant), K_c is the controller gain, T_d is the deadtime, K_p is the steady-state process gain, and λ is the user-defined closed-loop time constant.
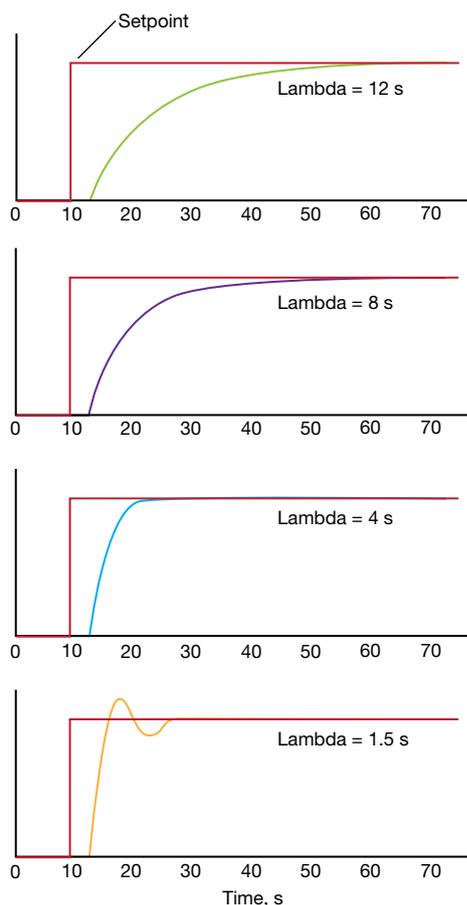
In this example, λ was initially chosen as 12 s, or three times the time constant. Using Eq. 1, K_c was initially calculated to be 0.148. As shown in Table 1, the value of λ was

decreased to achieve faster response times.

The graphs in Figure 10 show how fast the tuning will move the process variable back to its setpoint after a setpoint change. The orange curve illustrates the oscillation that typically occurs with a lambda value that is too low (*i.e.,* faster than the deadtime value).

Reactor ratio tuning example. When a facility implements a ratio control on the DCS, depicted in Figure 11, the assumption is that the ratio will be maintained consistently. However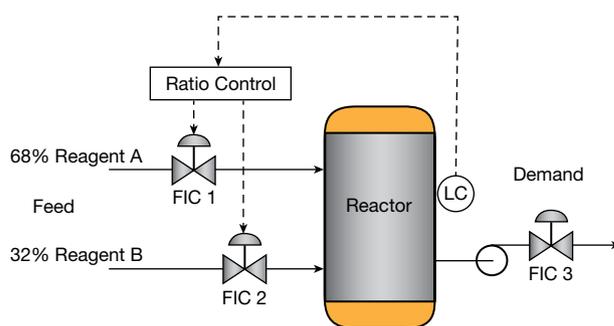, there are inherent differences in control valve characteristics based on valve type (*e.g.,* a globe valve has a wider range of process gain than a butterfly valve). Depending on where the control valve is operating, the initial response of each valve can be very different when the setpoint is changed.

In Figure 12, Ziegler-Nichols tuning was performed on two ratio-controlled flow loops. The loop for Component A uses a 2-in. equal-percentage control valve, and the loop for Component B has a 3-in. linear control valve. When the overall process flow changes, the change in ratio can

Table 1. Controller gain increases with decreasing lambda.		
Lambda	**Gain**	**Integral Time**
12 s	0.15	4 s/rep
8 s	0.21	4 s/rep
4 s	0.36	4 s/rep
1.5 s	0.67	4 s/rep

▲ **Figure 11.** A ratio control system with lambda tuning maintains consistent ratios despite process disturbances.
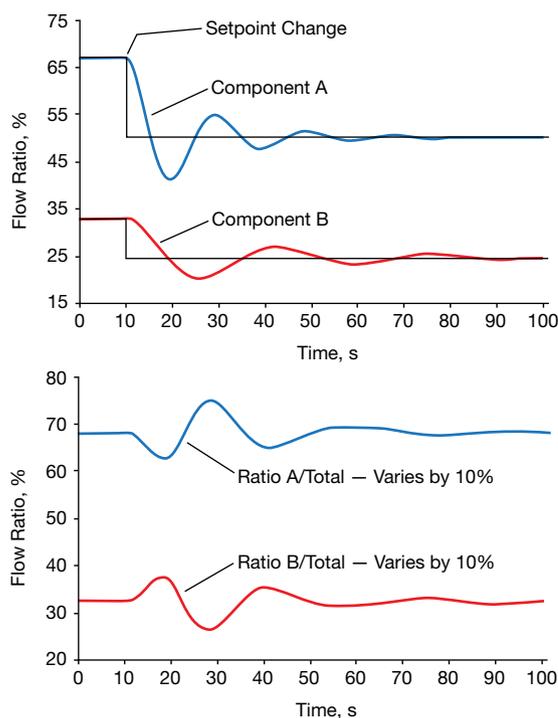
▲ **Figure 10.** A lower lambda value will move the process back to setpoint faster. However, oscillation will occur with a lambda value that is too low (*i.e.,* faster than the deadtime value).

▲ **Figure 12.** If Ziegler-Nichols tuning is performed on two ratio-controlled flow loops, when the overall process flow changes, the change in ratio can vary by as much as 10%.

vary by as much as 10%. This can result in lower yield of the desired product(s) and higher yield of undesired side reactions.

If both controllers are tuned with the same lambda value, any change in demand flow will result in both upstream reagent flows reaching new setpoints at the same time (Figure 13). The result is that the ratio of components remains the same regardless of process demand flow changes.

Step 7. Input new tuning parameters and observe

Keep a log of tuning parameter changes, including the time and date, along with who made the changes. Some modern systems will electronically log changes and the user who made them. Even if you have an electronic tuning log, it is always good to have a physical backup and to keep it updated. Include in the log your choice for lambda and why that value was selected (*e.g.*, to minimize interactions, prevent setpoint overshoot, improve load disturbance response, etc.).

If you make changes to the tuning, you will need to observe the controller's ability to minimize disturbances while maintaining the overall operating objectives effec-

tively. A common way to test the speed of response is to install the new tuning parameters and then perform a small setpoint change to confirm that the response is as expected.
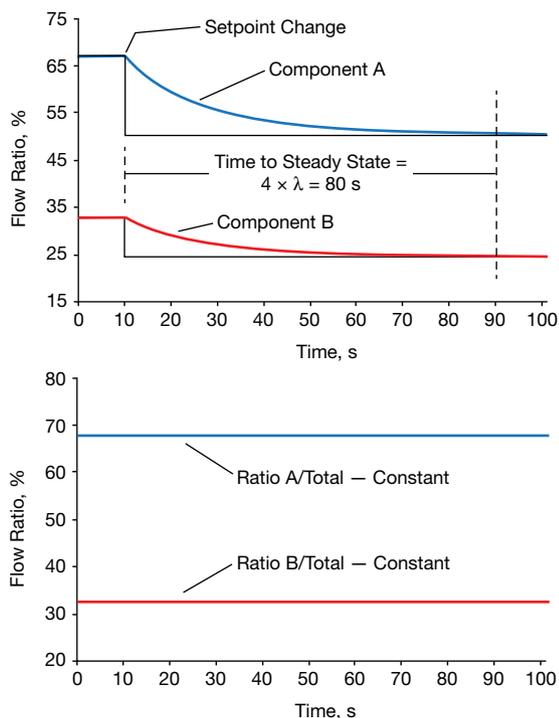
Step 8. As needed, follow up on tuning to ensure desired performance

Control valves have moving parts and degrade over time, and the performance of these control elements will impact the performance of tuning. A control loop that functioned fine in the past may not be working today due to hardware issues or a change in the operating parameters. If operating conditions change, and will continue for a while, a re-evaluation of the process and operating objectives as well as the tuning parameters may be in order.

Closing thoughts

Modern loop tuning is a simple and inexpensive way to troubleshoot control loop performance problems and improve overall process operations. Many vendors offer tools that capture process responses during step testing, making it easier to evaluate process dynamics and calculate tuning parameters. These tools are a convenience, especially if you are tuning on a frequent basis or have a large number of loops requiring attention. CEP

▲ **Figure 13.** If the flow controllers are tuned with the same lambda value, a disturbance in the process flow will result in the same change for both flow controllers. That way, both loops will reach their setpoint value at the same time.

LITERATURE CITED

1. **Bennett, S.,** "A History of Control Engineering 1930–1955," Peter Peregrinus Ltd., London, U.K. (1993).
2. **Dahlin, E. B.,** "Designing and Tuning Digital Controllers," *Instruments and Control Systems,* **41** (6), p. 77 (1968).
3. **Morari, M., and E. Zafiriou,** "Robust Process Control," Prentice Hall, Englewood Cliffs, NJ (1989).
4. **Chien, I. L. , and P. S. Fruehauf,** "Consider IMC Tuning to Improve Controller Performance," *Chem. Eng. Progress,* **86** (10), pp. 35–41 (Oct. 1990).
5. **Martin, J., et. al.,** "How to Select Controller Modes and Tuning Parameters from Simple Process Models," *ISA Transactions,* **15**, pp. 314–319 (Apr. 1976).

TIM OLSEN (Email: Tim.Olsen@emerson.com) has been with Emerson Process Management for 15 years. He is a refining consultant within the PlantWeb global refining industry solutions group, where he supports Emerson's technical and business strategy. He is currently the chair of AIChE's Fuels and Petrochemicals Div. He has a BS in chemical engineering from Iowa State Univ. with emphasis in industrial engineering and process control, and an MBA from the Univ. of Iowa.

NORMAN ITO (Email: Norman.Ito@emerson.com) is a senior consulting engineer at Emerson Process Management with 25 years of experience in process control design, process auditing, process simulation, and training. Much of his auditing experience covers process areas in the pulp and paper industry, steel industry (control of a basic oxygen furnace), pharmaceutical industry (batch reactor controls), power industry (boiler controls and cogeneration simulation), heavy oil operations, and liquid petroleum pipeline controls. He has a BS in electrical engineering from the Univ. of Toronto, and currently teaches the Emerson Education Services process control courses 9030–9034.