Feedback Controllers

Alessandro Anzalone, Ph.D.
Hillsborough Community College, Brandon Campus

Agenda

1. Introduction
2. Action of Controllers
3. Types of Feedback Controllers
   1. Proportional Controller
   2. Proportional–Integral Controller
   3. Proportional–Integral–Derivative Controller
   4. Proportional–Derivative Controller
4. Reset Windup
5. Tuning Feedback Controllers
   1. Online Tuning: Ziegler–Nichols Technique
   2. Offline Tuning
6. Summary
7. References

Introduction

In this chapter we present the most important types of industrial controllers. These controllers are used in analog systems, in distributed control systems (DCSs), and in stand-alone controllers, also sometimes referred to as single-loop controllers, or simply loop controllers. DCSs and stand-alone controllers are computer based, and consequently, they do not process the signals on a continuous basis but rather, in a discrete fashion. However, the sampling time for these systems is rather fast, usually ranging from 10 times per second to about once per second. Thus, for all practical purposes, these controllers appear to be continuous.

Action of Controllers

The selection of controller action is critical. If the action is not selected correctly, the controller will not control. Let us see how to select the action, and what it means.

Action of Controllers

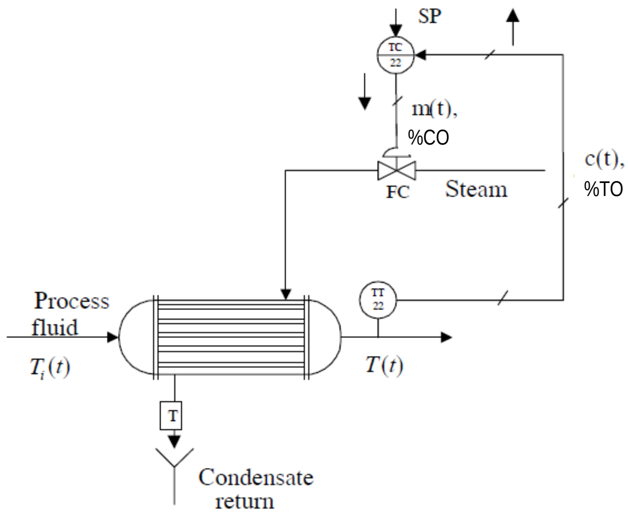


Figure 3-1.1 Heat exchanger control loop.

Action of Controllers

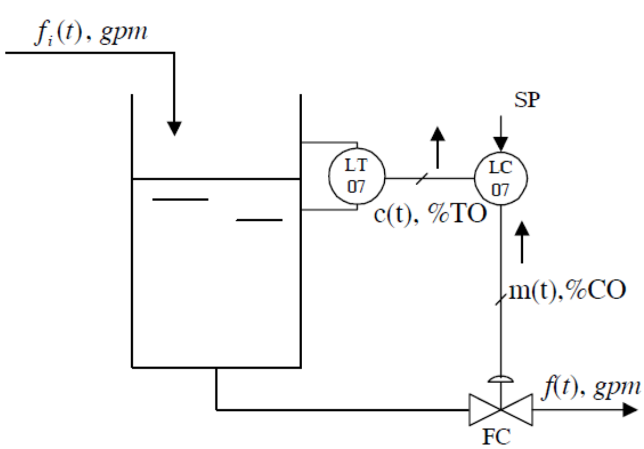


Figure 3-1.2 Liquid level control loop.

Types of Feedback Controllers

The way that feedback controllers make a decision is by solving an equation based on the difference between the set point and the controlled variable. In this section we look at the most common types of controllers by looking at the equations that describe their operation.

As mentioned, feedback controllers decide what to do to maintain the controlled variable at the set point by solving an equation based on the difference between the set point and the controlled variable. This difference, or error, is computed as

$$e(t) = c^{\text{set}}(t) - c(t)$$

Proportional Controller

The proportional (P) controller is the simplest type of controller we will discuss. The equation that describes its operation is

$$m(t) = \overline{m} + K_C e(t)$$

where m(t) is the controller output, %CO [the term m(t) is used to stress that as far as the controller is concerned, this output is the manipulated variable]; K_C is the controller gain, %CO/%TO; and $\overline{m}$ is the bias value, %CO (this is the output from the controller when the error is zero; $\overline{m}$ is a constant value and it is also the output when the controller is switched to manual; it is very often initially set at midscale, 50%CO).

Proportional Controller

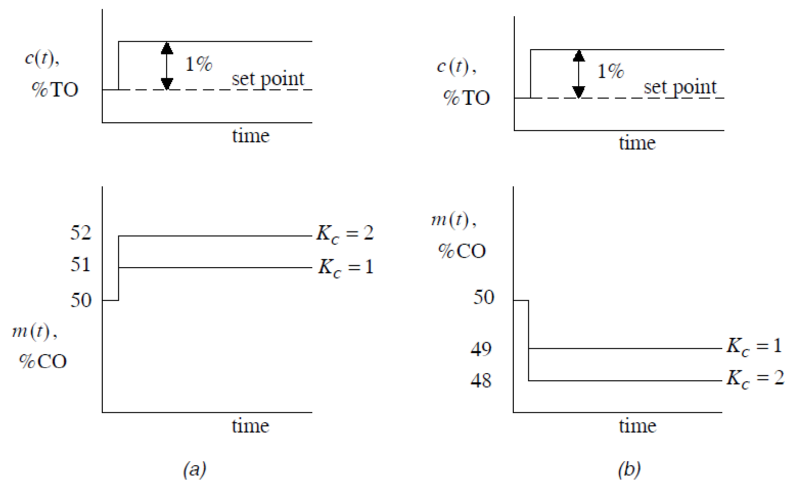


Figure 3-2.1 Effect of controller gain on output from controller: (*a*) direct-acting controller; (*b*) reverse-acting controller.

Proportional Controller

This equation shows that the output of the controller is proportional to the error. The proportionality is given by the controller gain, K_C. The significance of this gain is shown graphically in Fig. 3-2.1. The figure shows that the larger K_C value, the more the controller output changes for a given error. Thus K_C establishes the sensitivity of the controller to an error, that is, how much the controller output changes per unit change in error. In other words, K_C establishes the aggressiveness of the controller. The larger K_C is, the more aggressive the controller reacts to an error.

Proportional Controller

Proportional controllers have the advantage of only one tuning parameter, K_C. However, they suffer a major disadvantage—that of operating the controlled variable with an **offset**. Offset can be described as a steady-state deviation of the controlled variable from the set point, or just simply a steady-state error. To further explain the meaning of offset, consider the liquid-level control loop shown in Fig. 3-1.2.

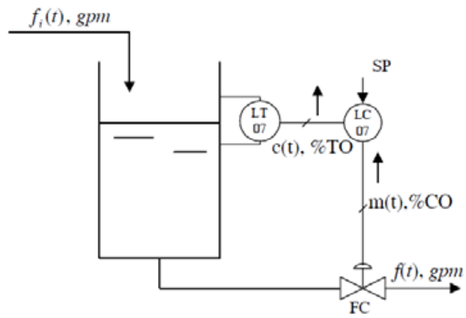


Figure 3-1.2 Liquid level control loop.

Proportional Controller

The design operating conditions are f_i = f = 150gpm and h = 4 ft. Let us also assume that for the outlet valve to deliver 150gpm, the signal to it must be 50%CO. If the inlet flow $f_i(t)$ increases, the response of the system with a proportional controller is shown in Fig. 3-2.2. The controller returns the controlled variable to a steady value but not to the set point required. The difference between the set point and the new steady state is the offset. The proportional controller is not "intelligent enough" to drive the controlled variable back to set point. The new steady-state value satisfies the controller.

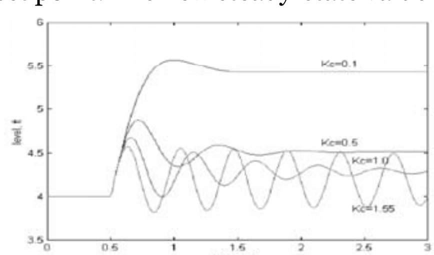


Figure 3-2.2 Response of level with different K_C values.

Proportional Controller

The obvious question is: Why does this offset occur?

$$m(t) = 50\% + K_C e(t) = 50\% + 10\% = 60\%$$

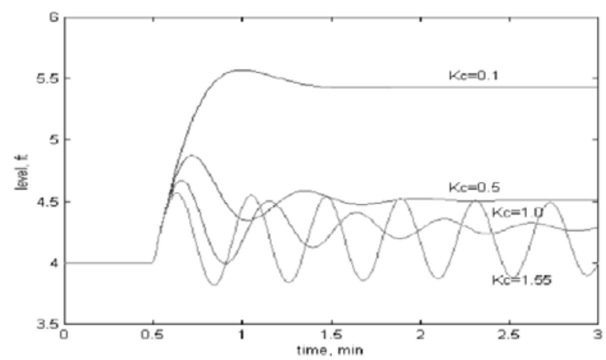


Figure 3-2.2 Response of level with different K_C values.

Proportional Controller

1. The magnitude of the offset depends on the value of the controller gain. Because the second term must have a value of +10%CO, the values are:

K_C	Offset, e(t),(%TO)
1	10
2	5.0
4	2.5

As mentioned previously, the larger the gain, the smaller the offset. The reader must remember that above a certain K_C, most processes go unstable. However, the controller equation does not show this.

2. It seems that all a proportional controller is doing is reaching a steady state operating condition. Once a steady state is reached, the controller is satisfied. The amount of deviation from the set point, or offset, depends on the controller gain.

Proportional Controller

Many controller manufacturers do not use the term K_C for the tuning parameter; they use the term proportional band, PB. The relationship between gain and proportional band is given by

$$\mathrm{PB} = \frac{100}{K_C}$$

In these cases the equation that describes the proportional controller is written as

$$m(t) = \overline{m} + \frac{100}{\mathrm{PB}} e(t)$$

PB is usually referred to as percent proportional band.

Proportional Controller

The transfer function of a proportional controller is

$$\frac{M(s)}{E(s)} = K_C$$

Proportional Controller

To summarize briefly, proportional controllers are the simplest controllers, with the advantage of only one tuning parameter, K_C or PB. The disadvantage of these controllers is operation with an offset in the controlled variable. In some processes, such as the level in a surge tank, the cruise control in a car, or a thermostat in a house, this may not be of any major consequence. For processes in which the process variable can be controlled within a band from set point, proportional controllers are sufficient. However, when the process variable must be controlled at the set point, not away from it, proportional controllers do not provide the required control.

Proportional-Integral Controller

Most processes cannot be controlled with an offset; that is, they must be controlled at the set point. In these instances an extra amount of "intelligence" must be added to the proportional controller to remove the offset. This new intelligence, or new mode of control, is the integral, or reset, action; consequently, the controller becomes a proportional–integral (PI) controller. The describing equation is

$$m(t) = \overline{m} + K_C e(t) + \frac{K_C}{\tau_I} \int e(t)\, dt$$

Proportional-Integral Controller

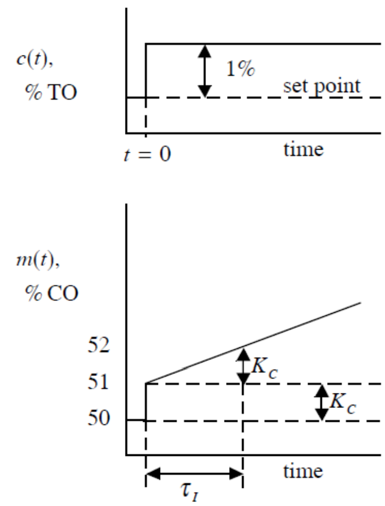


Figure 3-2.4 Response of a PI controller (direct acting) to a step change in error.

Proportional-Integral Controller

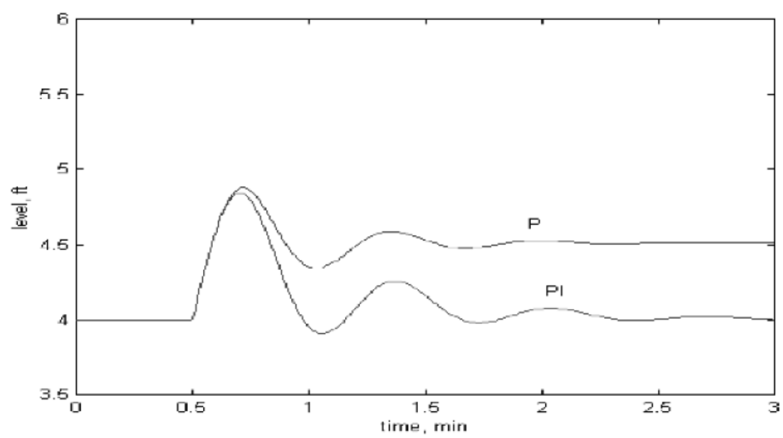


Figure 3-2.5 Response of level under P and PI control.

Proportional-Integral Controller

To explain why the PI controller removes the offset, consider the level control system used previously to explain the offset required by a P controller. Figure 3-2.5 shows the response of the level under P and PI controllers to a change in inlet flow from 150gpm to 170gpm. The response with a P controller shows the offset, while the response with a PI controller shows that the level returns to the set point, with no offset. Under PI control, as long as the error is present, the controller keeps changing its output (integrating the error). Once the error disappears, goes to zero, the controller does not change its output anymore (it integrates a function with a value of zero). Let us look at the PI equation at the moment the steady state is reached:

$$m(t) = 50\% + K_C(0) + \frac{K_C}{\tau_I}\int (0)\, dt$$
$$= 50\% + 0 + 10\% = 60\%$$

Proportional-Integral Controller

Some manufacturers do not use the reset time for their tuning parameter. They use the reciprocal of reset time, which we shall refer to as reset rate, t_I^R; that is,

$$\tau_I^R = \frac{1}{\tau_I}$$

Proportional-Integral Controller

The transfer function for the classical PI controller is

$$G_C(s) = \frac{M(s)}{E(s)} = K_C\left(1 + \frac{1}{\tau_I s}\right) = K_C \frac{\tau_I s + 1}{\tau_I s}$$

To summarize, proportional–integral controllers have two tuning parameters: the gain or proportional band, and the reset time or reset rate. The advantage is that the integration removes the offset. About 85% of all controllers in use are of this type. The disadvantage of the PI controller is related to the stability of the control loop.

Proportional-Integral Controller

Remembering that the ultimate gain, K_{CU}, is considered the limit of stability (maximum value of K_C before the system goes unstable), theory predicts, and practice confirms, that for a PI controller the K_{CU} is less than for a proportional controller. That is,

$$K_{CU}\big|_{\mathrm{P}} > K_{CU}\big|_{\mathrm{PI}}$$

The addition of integration adds some amount of instability to the system; this is presented in more detail in Chapter 5. Therefore, to counteract this effect, the controller must be tuned somewhat less aggressively (smaller K_C). The formulas we use to tune controllers will take care of this.

Proportional-Integral-Derivative Controller

Sometimes another mode of control is added to the PI controller. This new mode of control is the derivative action, also called the rate action, or pre-act. Its purpose is to anticipate where the process is heading by looking at the time rate of change of the error, its derivative. The describing equation is

$$m(t) = \overline{m} + K_C e(t) + \frac{K_C}{\tau_I} \int e(t)\,dt + K_C \tau_D \frac{de(t)}{dt}$$

Proportional-Integral-Derivative Controller

The transfer function of a PID controller is given by

$$G_C(s) = \frac{M(s)}{E(s)} = K_C\left(1 + \frac{1}{\tau_I s} + \tau_D s\right)$$

To summarize, PID controllers have three tuning parameters: the gain or proportional band, the reset time or reset rate, and the rate time. PID controllers should not be used in processes with noise. An advantage of the derivative mode is that it provides anticipation. Another advantage is related to the stability of the system. Theory predicts, and practice confirms, that the ultimate gain with a PID controller is larger than that of a PI controller. That is,

$$K_{CU}|_{\mathrm{PID}} > K_{CU}|_{\mathrm{PI}}$$

Proportional-Derivative Controller

The proportional–derivative (PD) controller is used in processes where a proportional controller can be used, where steady-state offset is acceptable but some amount of anticipation is desired, and no noise is present. The describing equation is

$$m(t) = \overline{m} + K_C e(t) + K_C \tau_D \frac{de(t)}{dt}$$

and the transfer function is

$$G_C(s) = \frac{M(s)}{E(s)} = K_C(1 + \tau_D s)$$

Reset Windup

The problem of reset windup is an important and realistic one in process control. It may occur whenever a controller contains integration. The heat exchanger control loop shown in Fig. 3-1.1 is again used at this time to explain the reset windup problem.

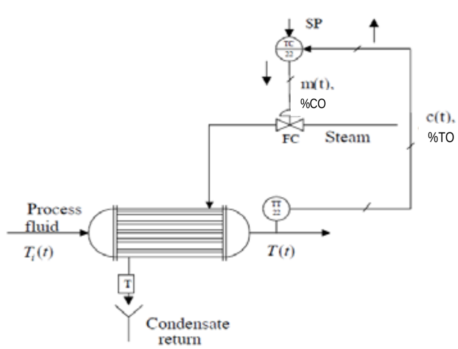
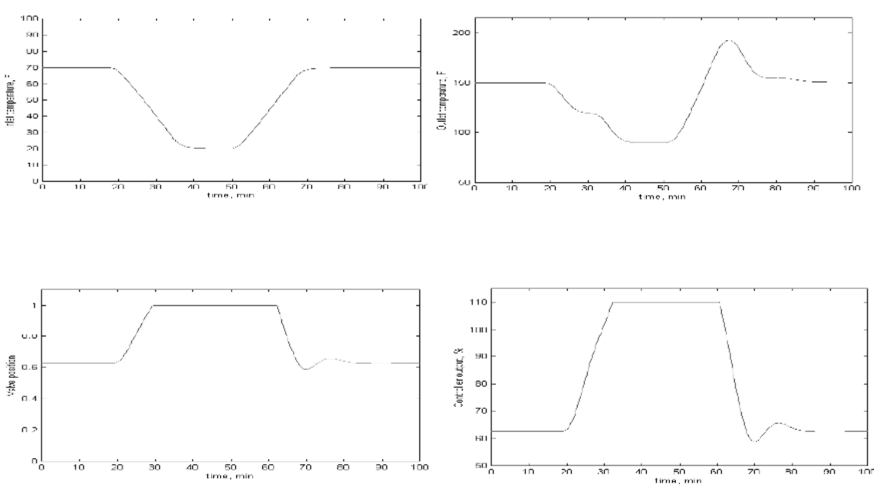


Figure 3-1.1 Heat exchanger control loop.

Reset Windup

Suppose that the process inlet temperature drops by an unusually large amount; this disturbance drops the outlet temperature. The controller (PI or PID) in turn asks the steam valve to open. Because the valve is fail-closed, the signal from the controller increases until, because of the reset action, the outlet temperature equals the desired set point. But suppose that in the effort of restoring the controlled variable to the set point, the controller integrates up to 100% because the drop in inlet temperature is too large. At this point the steam valve is wide open and therefore the control loop cannot do any more. Essentially, the process is out of control; this is shown in the following figure:

Reset Windup



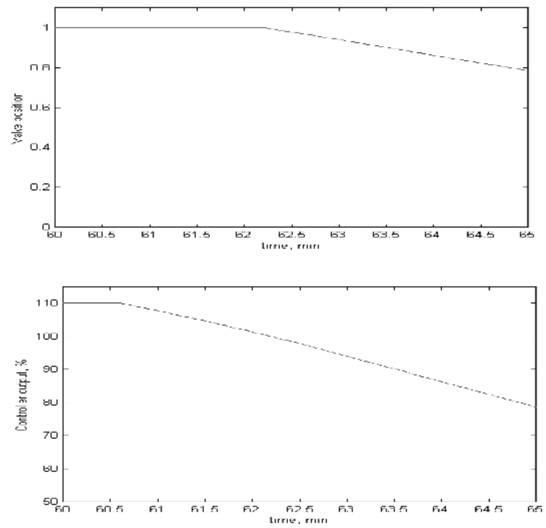
Reset Windup

The figure consists of four graphs: the inlet temperature, the outlet temperature, the valve position, and the controller's output. The figure shows that when the valve is fully open, the outlet temperature is not at set point. Since there is still an error, the controller will try to correct for it by further increasing (integrating the error) its output even though the valve will not open more after 100%. The output of the controller can in fact integrate above 100%. Some controllers can integrate between -15 and 115%, others between -7 and 107%, and still others between -5 and 105%. Analog controllers can also integrate outside their limits of 3 to 15 psig or 4 to 20mA. Let us suppose that the controller being used can integrate up to 110%; at this point the controller cannot increase its output anymore; its output has become saturated. This saturation is due to the reset action of the controller and is referred to as *reset windup*.

Reset Windup

Suppose now that the inlet temperature goes back up; the outlet process temperature will in turn start to increase. The outlet temperature reaches and passes the set point and the valve remains wide open when, in fact, it should be closing. The reason the valve is not closing is because the controller must now integrate from 110% down to 100% before it starts to close. Figure 3-3.2 shows an expanded view of how the controller's output starts to decrease from 110% and reaches 100% before the valve actually starts to close. The figure shows that it takes about 1.5 min for the controller to integrate down to 100%; all this time the valve is wide open. By the time the valve starts to close, the outlet temperature has overshot the set point by a significant amount, about 30°F in this case.

Reset Windup



Reset Windup

As mentioned earlier, this problem of reset windup may occur whenever integration is present in the controller. It can be avoided if the integration is limited to 100% (or 0%). Note that the prevention of reset windup requires us to limit the integration, not to limit the controller output when its value reaches 100% or 0%. While the output does not go beyond the limits, the controller may still be internally wound up, because it is the integral mode that winds up. Reset windup protection is an option that must be bought in analog controllers; however, it is a standard feature in DCS controller.

Reset windup occurs any time a controller is not in charge, such as when a manual bypass valve is open or when there is insufficient manipulated variable power. It also typically occurs in batch processes, in cascade control, and when a final control element is driven by more than one controller, as in override control schemes.

Tuning Feedback Controllers

Probably 80 to 90% of feedback controllers are tuned by instrument technicians or control engineers based on their previous experience. For the 10 to 20% of cases where no previous experience exists, or for personnel without previous experience, there exist several organized techniques to obtain a "good ballpark figure" close to the "optimum" settings.

Tuning Feedback Controllers

To use these organized procedures, we must first obtain the characteristics of the process. Then, using these characteristics, the tunings are calculated using simple formulas; Fig. 3-4.1 depicts this concept. There are two ways to obtain the process characteristics, and consequently, we divide the tuning procedures into two types: on-line and off-line.

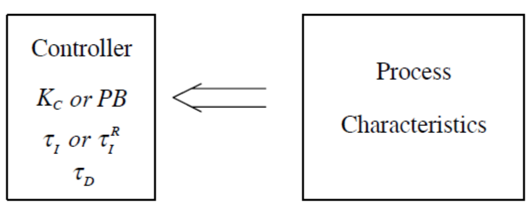


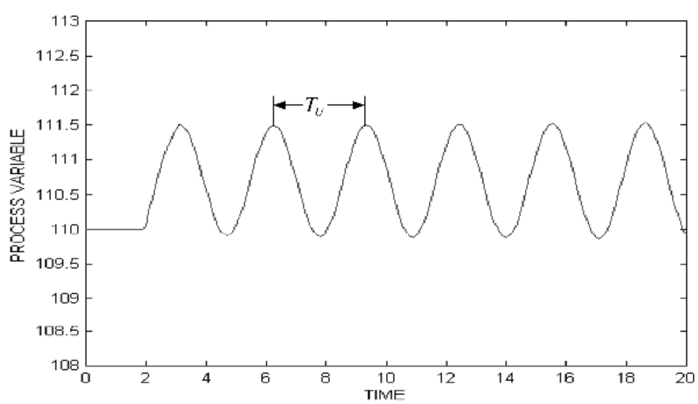
Figure 3-4.1 Tuning concept.

Online Tuning: Ziegler–Nichols Technique

The Ziegler–Nichols technique is the oldest method for online tuning. It gives approximate values of the tuning parameters K_C, τ_I, and τ_D to obtain approximately a one fourth (1/4) decay ratio response. The procedure is as follows:

1. With the controller online (in automatic), remove all the reset (τ_I = maximum or τ_I^R = minimum) and derivative (τ_D = 0) modes. Start with a small K_C value.
2. Make a small set point or load change and observe the response.
3. If the response is not continuously oscillatory, increase K_C, or decrease PB, and repeat step 2.
4. Repeat step 3 until a continuous oscillatory response is obtained.

Online Tuning: Ziegler–Nichols Technique



Online Tuning: Ziegler–Nichols Technique

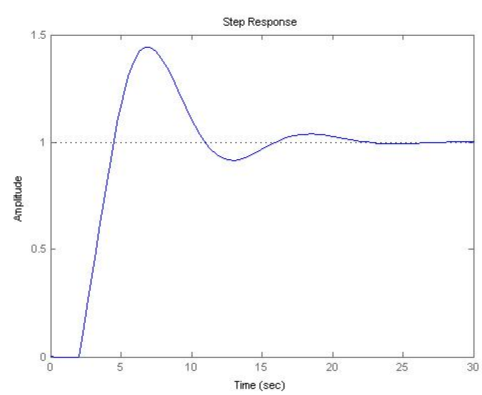
The gain that gives these continuous oscillations is the ultimate gain, K_{CU}. The period of the oscillations is called the ultimate period, T_U; this is shown in the previous figure.

The ultimate gain and the ultimate period are the characteristics of the process being tuned. The following formulas are then applied:

- ✓ For a P controller: $K_C = 0.5K_{CU}$
- ✓ For a PI controller: $K_C = 0.45K_{CU}$, $\tau_I = T_U/1.2$
- ✓ For a PID controller: $K_C = 0.65K_{CU}$, $\tau_I = T_U/2$, $\tau_D = T_U/8$

Online Tuning: Ziegler–Nichols Technique

This figure shows the response of a process with a PI controller tuned by the Ziegler–Nichols method. The figure also shows the meaning of a 1/4 decay ratio response.



http://www.mathworks.com/matlabcentral/fileexchange/18561

Offline Tuning

The data required for the offline tuning techniques are obtained from the step testing method presented in Chapter 2, that is, from K, τ, and to. Remember that K must be in %TO/%CO, and τ and to in time units consistent with those used in the controller to be tuned. These three terms describe the characteristics of the process. Once the data are obtained, any of the methods described next can be applied.

Offline Tuning

Ziegler–Nichols Method. The Ziegler–Nichols settings can also be obtained from the following formulas:

- ✓ For a P controller: $K_C = (1/K)(to/\tau)^{-1}$
- ✓ For a PI controller: $K_C = (0.9/K)(to/\tau)^{-1}$, $\tau_I = 3.33to$
- ✓ For a PID controller: $K_C = (1.2/K)(to/\tau)^{-1}$, $\tau_I = 2.0to$, $\tau_D = 0.5to$

The Ziegler–Nichols method was developed for to/τ<1.0. For ratios greater than 1.0, the tunings obtained by this method become very conservative.

Offline Tuning

Controller Synthesis Method. The controller synthesis method (CSM) was introduced by Martin, Corripio, and C. L. Smith. Several years later internal model control (IMC) was presented and the tunings from this method agree with those from the CSM. Some people also refer to the CSM as the lambda tuning method.

- ✓ For a P controller: $K_C = \tau/K(\lambda + to)$
- ✓ For a PI controller: $K_C = \tau/K(\lambda + to)$, $\tau_I = \tau$
- ✓ For a PID controller: $K_C = \tau/K(\lambda + to)$, $\tau_I = \tau$, $\tau_D = to/2$

Offline Tuning

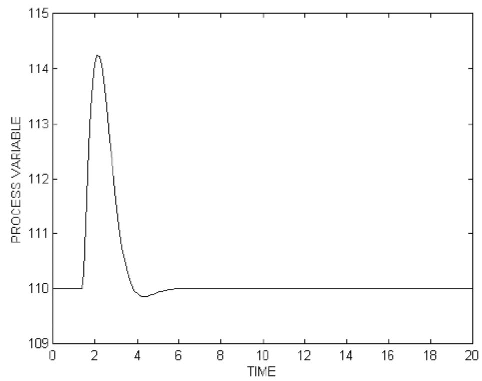
Looking at the formulas, it is clear that for each controller it comes down to only one tuning parameter, λ. As the formulas show, the smaller the λ value, the more aggressive (the larger the K_C) the controller becomes. We recommend the following values of λ as a first guess:

- ✓ For a P controller: $\lambda = 0$
- ✓ For a PI controller: $\lambda = to$
- ✓ For a PID controller: $\lambda = 0.2to$

The response obtained by this method tend to give a more overdamped (less oscillatory) response than the Ziegler–Nichols, depending on the value of λ used.

Offline Tuning

Figure 3-4.4 shows the response of the same process as in Fig. 3-4.3, but this time with a PI controller tuned by the CSM method, with the λ suggested. The CSM method is not limited by the value of to/τ as are the Ziegler–Nichols tunings.



Offline Tuning

Other Tunings. In this section we discuss the tuning of flow loops and level loops. Both loops are quite common, and present characteristics that make it difficult to tune them with the methods presented thus far.

Flow Loops. Flow loops are the most common loops in the process industries. Their dynamic response is rather fast. Consider the loop shown in Fig. 3-4.5. Assume that the controller is in manual and a step change in controller output is induced. The response of the flow is almost instantaneous; the only dynamic element is the control valve. The two-point method of Chapter 2, used to obtain a first-order-plus-deadtime approximation of the response, shows that the dead-time term is very close to zero, to ≈ 0min.

Offline Tuning

Flow Loops.

In every tuning equation for controller gain, the dead time appears in the denominator of the equation. Thus the results would show a need for an infinite controller gain. Analysis of these types of fast processes indicates that the controller needed is an integral only. Because pure integral controllers were not available when only analog instrumentation was available, a PI controller was used with very small proportional action and a large integral action. Today, this practice is still followed. The following is offered as a rule of thumb for flow loops:

- ✓ Conservative tuning: $K_C = 0.1$, $\tau_I = 0.1$min
- ✓ Aggressive tuning: $K_C = 0.2$, $\tau_I = 0.05$min

Offline Tuning

Level Loops. Level loops present two interesting characteristics. The first characteristic is that as presented in Chapter 2, very often levels are integrating processes. In this case it is impossible to obtain a response to approximate it with a first-order-plus-dead-time model. That is, it is impossible to obtain K, τ, and to, and therefore we cannot use any tuning equation presented thus far. Those levels processes that are not integrating processes but rather, self-regulating processes can be approximated by a first-order-plus-dead-time model, as shown in this chapter.

Offline Tuning

Level Loops.

The second characteristic of level loops is that there are two possible control objectives. To explain these control objectives, consider Fig. 3-4.6. If the input flow varies as shown in the figure, to control the level tightly at set point the output flow must also vary, as shown. We refer to this as *tight level control.*

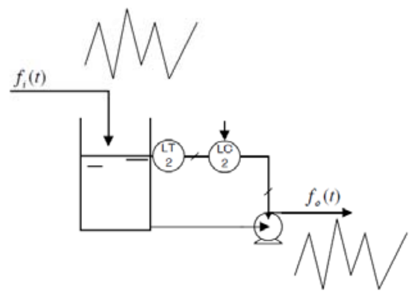


Figure 3-4.6 Level loop.

Offline Tuning

Level Loops.

However, the changes in output flow will act as disturbances to the downstream process unit. If this unit is a reactor, separation column, filter, and so on, the disturbance may have a major effect on its performance. Often, it is desired to smooth the flow feeding the downstream unit. To accomplish this objective, the level in the tank must be allowed to "float" between a high and a low level. Thus, the objective is not to control the level tightly but rather, to smooth the output flow with some consideration of the level. We refer to this objective as *average level control.* Let us look at how to tune the level controller for each objective.

Offline Tuning

Tight level control. If the level process happens to be self-regulated, that is, if it is possible to obtain K, τ, and to, the tuning techniques already presented in this chapter can be used. If the level process is integrating, the following equation for a proportional controller is proposed:

$$K_C = \frac{A}{4\tau_V K_V K_T}$$

where, A is the cross-sectional area of tank (length²), τ_V the time constant of the valve (time), K_V the valve's gain [length³/(time ·%CO)], and K_T the transmitter's gain (%TO/length). The valve's gain can be approximated by

$$K_V = \frac{\text{maximum volumetric flow provided by valve}}{100\%\text{CO}}$$

Offline Tuning

Tight level control.

The time constant of the valve depends on several things, such as the size of the actuator, whether a positioner is used or not, and so on. Anywhere between 3 and 10 seconds (0.05 to 0.17 minutes) could be used.

Offline Tuning

Average level control. To review what we had previously said, the objective of average level control is to smooth the output flow from the tank. To accomplish this objective, the level in the tank must be allowed to "float" between a high and a low level. Obviously, the larger the difference between the high and low levels, the more "capacitance" is provided, and the more smoothing of the flow is obtained.

There are two ways to tune a proportional controller for average level control. The first way is also discussed in Ref. 2 and says: *The ideal averaging level controller is a proportional controller with the set point at 50%TO, the output bias at 50%CO, and the gain set at 1%CO/%TO*. The tuning obtained in this case results in that the level in the tank will vary the full span of the transmitter as the valve goes from wide open to completely closed. Thus the full capacitance provided by the transmitter is used.

Offline Tuning

Average level control.

To explain the second way to tune the controller, consider Fig. 3-4.7. The figure shows two deviations, D1 and D2, not present in Fig. 3-4.6. D1 indicates the expected flow deviation from the average flow. D2 indicates the allowed level deviation from set point.

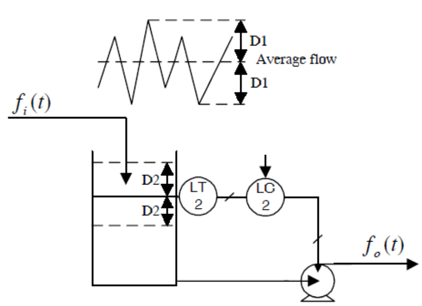


Figure 3-4.7 Level loop.

Offline Tuning

Average level control.
With this information we can now write the tuning equation:

$$K_C = (0.75)\frac{\dfrac{\text{expected flow deviation from the average input flow (D1)}}{\text{maximum flow given by final control element } (f_{o,\max})}}{\dfrac{\text{allowed level deviation from set point (D2)}}{\text{span of level transmitter}}}$$

The equation is composed of two ratios, and both ratios must be dimensionless. This equation allows you to (1) use less than the span of the transmitter if it is necessary for some reason, and (2) take into consideration the variations in input flow. For best results, the level should be allowed to vary as much as possible and D2 made as large as possible; this is a decision for the engineer. D1 depends on the process. The final control element shown in Fig. 3-4.3 is a pump; valves are also common. The equation was developed so that once the engineer decides on D2, this limit is not violated, while providing smoothing of the output flow.

Summary

In this chapter we have seen that the purpose of controllers is to make decisions on how to use the manipulated variable to maintain the controlled variable at set point. We have discussed the significance of the action of the controller, reverse or direct, and how to select the appropriate one. The carious types of controllers were also studied, stressing the significance of the tuning parameters, gain K_C or proportional band PB, reset time τ_I or reset rate τ_I^R, and derivative or rate time τ_D. The subject of reset windup was presented and its significance discussed. Finally, various tuning techniques were presented and discussed.

References

1. Automated Continuous Process Control, Carlos A. Smith, 2002, Wiley-Interscience, ISBN: 978-0471215783.
2. C. A. Smith and A. B. Corripio, Principles and Practice of Automatic Process Control, 3rd ed., Wiley, New York, 2006.