Controller Algorithms and Tuning

The previous sections of this module described the purpose of control, defined individual elements within control loops, and demonstrated the symbology used to represent those elements in an engineering drawing. The examples of control loops used thus far have been very basic. In practice, control loops can be fairly complex. The strategies used to hold a process at setpoint are not always simple, and the interaction of numerous setpoints in an overall process control plan can be subtle and complex. In this section, you will be introduced to some of the strategies and methods used in complex process control loops.

Learning Objectives

After completing this section, you will be able to:

- Differentiate between discrete, multistep, and continuous controllers
- Describe the general goal of controller tuning.
- Describe the basic mechanism, advantages and disadvantages of the following mode of controller action:
  - Proportional action
  - Integral action
  - Derivative action
- Give examples of typical applications or situations in which each mode of controller action would be used.
- Identify the basic implementation of P, PI and PID control in the following types of loops:
  - Pressure loop
  - Flow loop
  - Level loop
  - Temperature loop

Note: To answer the activity questions the Hand Tool (H) should be activated.
Controller Algorithms

The actions of controllers can be divided into groups based upon the functions of their control mechanism. Each type of controller has advantages and disadvantages and will meet the needs of different applications. Grouped by control mechanism function, the three types of controllers are:

- Discrete controllers
- Multistep controllers
- Continuous controllers

**DISCRETE CONTROLLERS**

*Discrete controllers* are controllers that have only two modes or positions: on and off. A common example of a discrete controller is a home hot water heater. When the temperature of the water in the tank falls below setpoint, the burner turns on. When the water in the tank reaches setpoint, the burner turns off. Because the water starts cooling again when the burner turns off, it is only a matter of time before the cycle begins again. This type of control doesn’t actually hold the variable at setpoint, but keeps the variable within proximity of setpoint in what is known as a *dead zone* (Figure 7.15).

---

**Activities**

1. Which one of the following is an everyday example of a discrete controller?
   Select the options that apply.

   - Refrigerator
   - Electric iron
   - Air conditioner
   - Rice cooker
Controller Algorithms

**MULTISTEP CONTROLLERS**

*Multistep controllers* are controllers that have at least one other possible position in addition to on and off. Multistep controllers operate similarly to discrete controllers, but as setpoint is approached, the multistep controller takes intermediate steps. Therefore, the oscillation around setpoint can be less dramatic when multistep controllers are employed than when discrete controllers are used (Figure 7.16).

![Multistep Control Profile](image)

**CONTINUOUS CONTROLLERS**

*Controllers* automatically compare the value of the PV to the SP to determine if an error exists. If there is an error, the controller adjusts its output according to the parameters that have been set in the controller. The tuning parameters essentially determine:

- **How much** correction should be made? The *magnitude* of the correction (change in controller output) is determined by the proportional mode of the controller.

- **How long** should the correction be applied? The *duration* of the adjustment to the controller output is determined by the integral mode of the controller.

- **How fast** should the correction be applied? The *speed* at which a correction is made is determined by the derivative mode of the controller.

**Activities**

2. A controller with three or more set positions is called a continuous controller. Is this statement true or false?
Controller Algorithms

When there is an error, the controller makes a change in its output. It determines:
- How much? Proportional Mode
- How long? Integral Mode
- How fast? Derivative Mode

Automatic Feedback Control

Activities
Why Controllers Need Tuning?

Controllers are tuned in an effort to match the characteristics of the control equipment to the process so that two goals are achieved:

- The system responds quickly to errors.
- The system remains stable (PV does not oscillate around the SP).

**GAIN**

Controller tuning is performed to adjust the manner in which a control valve (or other final control element) responds to a change in error. In particular, we are interested in adjusting the gain of the controller such that a change in controller input will result in a change in controller output that will, in turn, cause sufficient change in valve position to eliminate error, but not so great a change as to cause instability or cycling.

Gain is defined simply as the change in output divided by the change in input.

**Examples:**

- Change in Input to Controller - 10%
- Change in Controller Output - 20%
  
  $$\text{Gain} = \frac{20\%}{10\%} = 2$$

- Change in Input to Controller - 10%
- Change in Controller Output - 5%
  
  $$\text{Gain} = \frac{5\%}{10\%} = 0.5$$

convey measurements and instructions to other instruments in a control loop to maintain the highest level of safety and efficiency.

The next three sections in this module discuss electricity, circuits, transmitters, and signals in greater detail so you can understand the importance of electricity in process control.

**Activities**

3. The change in the controller output divided by the change in the input to the controller is known as ________.
Why Controllers Need Tuning?

**Gain Plot** - The Figure below is simply another graphical way of representing the concept of gain.

\[
\text{Gain } K_c = \frac{\Delta \text{Output} \%}{\Delta \text{Input} \%}
\]

![Graphical Representation of Gain Concept](https://mywb.com)

**Examples** - The following examples help to illustrate the purpose of setting the controller gain to different values.

- **Controllers May be Tuned to Help Match the Valve to the Process**
  - *Fast Process May Require Less Gain To Achieve Stability*
    - [Diagram of Small Volume Liquid Process]
  - *Slow Process May Require Higher Gain To Achieve Responsiveness*
    - [Diagram of Large Volume Gas Process]

**Activities**

4. Fast or slow processes have no impact on controller gain settings. Is this statement true or false?
Proportional Mode

**PROPORTIONAL ACTION**

The proportional mode is used to set the basic gain value of the controller. The setting for the proportional mode may be expressed as either:
1. Proportional Gain
2. Proportional Band

**PROPORTIONAL GAIN**

In electronic controllers, proportional action is typically expressed as proportional gain. Proportional Gain (Kc) answers the question: "What is the percentage change of the controller output relative to the percentage change in controller input?"

Proportional Gain is expressed as:

\[
\text{Gain, (Kc)} = \frac{\Delta \text{Output}\%}{\Delta \text{Input}\%}
\]

**PROPORTIONAL BAND**

Proportional Band (PB) is another way of representing the same information and answers this question:
"What percentage of change of the controller input span will cause a 100% change in controller output?"

\[
\text{PB} = \frac{\Delta \text{Input} (\% \text{Span})}{\Delta \text{Output}} \text{ For 100}\%\Delta \text{Output}
\]

**Converting Between PB and Gain**

A simple equation converts gain to proportional Band:

\[
\text{PB} = \frac{100}{\text{Gain}}
\]

Also recall that:

\[
\text{Gain} = \frac{100}{\text{PB}}
\]

Proportional Gain, (Kc) = \(\frac{\Delta \text{Output}\%}{\Delta \text{Input}\%}\)

\[
\text{PB} = \frac{\Delta \text{Input}(\%\text{Span})}{\Delta \text{Output}} \text{ For 100}\%\Delta \text{Output}
\]

**Activities**

5. Identify the major disadvantage of proportional action.

1. Tends to leave an offset
2. Reset Windup during shutdown
3. Possible overshoot during startup
4. Can cause cycling in fast process by amplifying noisy signals
Proportional Mode

**LIMITS OF PROPORTIONAL ACTION**

**Responds Only to a Change in error** - Proportional action responds *only* to a *change in the magnitude* of the error.

**Does Not Return the PV to Setpoint** - Proportional action will *not* return the PV to setpoint. It will, however, return the PV to a value that is within a defined span (PB) around the PV.

**DETERMINING THE CONTROLLER OUTPUT**

**Controller Output** - In a proportional only controller, the output is a function of the change in error and controller gain.

\[
\text{Output Change, \%} = (\text{Error Change, \%}) \times \text{(Gain)}
\]

**Example**: If the setpoint is suddenly changed 10% with a proportional band setting of 50%, the output will change as follows:

**Calculating Controller Output**

\[
\Delta\text{Controller Output} = \Delta\text{Input, \%} \times \text{Gain}
\]

**EXAMPLE**

\[
\Delta\text{Input} = 10\%
\]

\[
\text{PB} = 50\%, \text{so Gain} = 100%/50% = 2
\]

**Activities**

6. If proportional gain is 0.5, and a level reading is 5% above setpoint, a proportional controller will signal the outflow control valve to open by <1 / 2.5 / 5> % of its full range.
Proportional Mode

ΔController Output = Δ Input X Gain
ΔController Output = 10% X 2 = 20%

Expressed in Units:
Controller Output Change = (0.2)(12 psi span) = 2.4 psi OR
(0.2)(16 mA span) = 3.2 mA

Proportional Action - Closed Loop

Loop Gain - Every loop has a critical or natural frequency. This is the frequency at which cycling may exist. This critical frequency is determined by all of the loop components. If the loop gain is too high at this frequency, the PV will cycle around the SP; i.e., the process will become unstable.

Low Gain Example - In the example below, the proportional band is high (gain is low). The loop is very stable, but an error remains between SP and PV.

High Gain Example - In the example, the proportional band is small resulting in high gain, which is causing instability. Notice that the process variable is still not on set point.
Proportional Mode

Proportional Summary - For the proportional mode, controller output is a function of a change in error. Proportional band is expressed in terms of the percentage change in error that will cause 100% change in controller output. Proportional gain is expressed as the percentage change in output divided by the percentage change in input.

\[ \text{PB} = \left( \frac{\Delta \text{Input}, \%}{\Delta \text{Output}, \%} \right) \times 100 = 100/\text{Gain} \]

\[ \text{Gain} = \Delta \text{Input} \% / \Delta \text{Output} \% \]

\[ \Delta \text{Controller Output} = (\text{Change in Error})(\text{Gain}) \]

1. Proportional Mode Responds only to a change in error
2. Proportional mode alone will not return the PV to SP.

Advantages - Simple

Disadvantages - Error

Settngs - PB settings have the following effects:

- Small PB (%) Minimize Offset
- High Gain (%) Possible cycling
- Large PB (%) Large Offset
- Low Gain Stable Loop

Tuning - reduce PB (increase gain) until the process cycles following a disturbance, then double the PB (reduce gain by 50%).

Activities

7. What will be the result if the proportional gain is set too high?
   Select all options that apply.
   1. Large offset
   2. Minimized offset
   3. Possible cycling
   4. Stable loop
**Integral Mode**

**INTEGRAL ACTION**

**Duration of Error and Integral Mode** - Another component of error is the *duration* of the error, i.e., how long has the error existed? The controller output from the integral or reset mode is a function of the duration of the error.

**Activities**

8. ___________ action is the type of control algorithm that eliminates offset.

**Open Loop Analysis**

**Purpose** - The purpose of integral action is to return the PV to SP. This is accomplished by repeating the action of the proportional mode as long as an error exists. With the exception of some electronic controllers, the integral or reset mode is always used with the proportional mode.

**Setting** - Integral, or reset action, may be expressed in terms of:

- **Repeats Per Minute** - How many times the proportional action is repeated each minute.
- **Minutes Per Repeat** - How many minutes are required for 1 repeat to occur.
Integral Mode

CLOSED LOOP ANALYSIS

Closed Loop With Reset - Adding reset to the controller adds one more gain component to the loop. The faster the reset action, the greater the gain.

Slow Reset Example - In this example the loop is stable because the total loop gain is not too high at the loop critical frequency. Notice that the process variable does reach set point due to the reset action.

Fast Reset Example - In the example the rest is too fast and the PV is cycling around the SP.

Activities

9. Which of the following are integral or reset actions expressed in terms of?
   Select all options that apply.
   1. Repeats per setting
   2. Repeats per minute
   3. Repeats per loop
   4. Minutes per repeat
**Integral Mode**

**RESET WINDUP**

**Defined** - Reset windup is described as a situation where the controller output is driven from a desired output level because of a large difference between the set point and the process variable.

**Shutdown** - Reset windup is common on shut down because the process variable may go to zero but the set point has not changed, therefore this large error will drive the output to one extreme.

**Startup** - At start up, large process variable overshoot may occur because the reset speed prevents the output from reaching its desired value fast enough.

**Anti Reset Windup** - Controllers can be modified with an anti-reset

---

**Activities**

10. Identify the major disadvantages of integral action.
   Select all options that apply.

1. Tends to leave an offset
2. Reset windup during shutdown
3. Possible overshoot during start up
4. Can cause cycling in fast process by amplifying noisy signals
Integral Mode

windup (ARW) device. The purpose of an anti-reset option is to allow the output to reach its desired value quicker, therefore minimizing the overshoot.

**SUMMARY**

Integral (Reset) Summary - Output is a repeat of the proportional action as long as error exists. The units are in terms of repeats per minute or minutes per repeat.

**Advantages** - Eliminates error

**Disadvantages** - Reset windup and possible overshoot

| Fast Reset | 1. High Gain |
| Large Repeats/Min., Small Min./Repeat | 2. Fast Return To Setpoint |
|  | 3. Possible Cycling |

| Slow Reset | 1. Low Gain |
| Small Repeats/Min., Large Min./Repeats | 2. Slow Return To Setpoint |
|  | 3. Stable Loop |

**Trailing and Error Tuning** - Increase repeats per minute until the PV cycles following a disturbance, then slow the reset action to a value that is 1/3 of the initial setting.
Derivative Mode

DERIVATIVE ACTION

**Derivative Mode Basics** - Some large and/or slow process do not respond well to small changes in controller output. For example, a large liquid level process or a large thermal process (a heat exchanger) may react very slowly to a small change in controller output. To improve response, a large initial change in controller output may be applied. This action is the role of the derivative mode.

The derivative action is initiated whenever there is a change in the rate of change of the error (the slope of the PV). The magnitude of the derivative action is determined by the setting of the derivative. The mode of a PID controller and the rate of change of the PV. The Derivative setting is expressed in terms of minutes. In operation, the controller first compares the current PV with the last value of the PV. If there is a change in the slope of the PV, the controller determines what its output would be at a future point in time (the future point in time is determined by the value of the derivative setting, in minutes). The derivative mode immediately increases the output by that amount.

![Graph showing derivative action](image)

**Activities**

11. ___________ action is a control algorithm that is tied to the rate of change in the error.

12. Which of the following are derivative or rate actions expressed in terms of?

1. Repeats per minute
2. Hours
3. Seconds
4. Minutes
5. Milliseconds
Derivative Mode

**Example** - Let's start a closed loop example by looking at a temperature control system. In this example, the time scale has been lengthened to help illustrate controller actions in a slow process. Assume a proportional band setting of 50%. There is no reset at this time. The proportional gain of 2 acting on a 10% change in set point results in a change in controller output of 20%. Because temperature is a slow process the setting time after a change in error is quite long. And, in this example, the PV never becomes equal to the SP because there is no reset.

**Rate Effect** - To illustrate the effect of rate action, we will add the rate mode with a setting of 1 minute. Notice the very large controller output at time 0. The output spike is the result of rate action. Recall that the change in output due to rate action is a function of the speed (rate) of change of error, which in a step is nearly infinite. The addition of rate alone will not cause the process variable to match the set point.

**Activities**

13. The addition of derivative or rate alone to a close loop control can cause the process variable to match the set point. Is this statement true or false?
**Derivative Mode**

**Effect of Fast Rate** - Let's now increase the rate setting to 10 minutes. The controller gain is now much higher. As a result, both the IVP (controller output) and the PV are cycling. The point here is that increasing the rate setting will not cause the PV to settle at the SP.

![Graph showing IVP and PV with PD control, high rate setting, closed loop analysis.](image)

**Activities**

- **Need for Reset Action** - It is now clear that reset must be added to bring process variable back to set point.

- **Applications** - Because this component of the controller output is dependent on the speed of change of the input or error, the output will be very erratic if rate is used on fast process or one with noisy signals. The controller output, as a result of rate, will have the greatest change when the input changes rapidly.

- **Controller Option to Ignore Change in SP** - Many controllers, especially digital types, are designed to respond to changes in the PV only, and to ignore changes in SP. This feature eliminates a major upset upset that would occur following a change in the setpoint.
## Derivative Mode

### SUMMARY

**Derivative (Rate) Summary** - Rate action is a function of the *speed of change* of the error. The units are *minutes*. The action is to apply an immediate response that is equal to the proportional plus reset action that would have occurred some number of minutes in the future.

**Advantages** - Rapid output reduces the time that is required to return PV to SP in slow process.

**Disadvantage** - Dramatically amplifies noisy signals; can cause cycling in fast processes.

### Settings

| Large (Minutes) | 1. High Gain  
|                | 2. Large Output Change  
|                | 3. Possible Cycling |
| Small (Minutes) | 1. Low Gain  
|                | 2. Small Output Change  
|                | 3. Stable Loop |

### Trial-and-Error Tuning

Increase the rate setting until the process cycles following a disturbance, then reduce the rate setting to one-third of the initial value.
Controller Algorithms

**Controller Algorithms**

**Proportional, PI, and PID Control**

By using all three control algorithms together, process operators can:

- Achieve rapid response to major disturbances with derivative control
- Hold the process near setpoint without major fluctuations with proportional control
- Eliminate offset with integral control

Not every process requires a full PID control strategy. If a small offset has no impact on the process, then proportional control alone may be sufficient.

PI control is used where no offset can be tolerated, where noise (temporary error readings that do not reflect the true process variable condition) may be present, and where excessive dead time (time after a disturbance before control action takes place) is not a problem.

In processes where no offset can be tolerated, no noise is present, and where dead time is an issue, customers can use full PID control. Table 7.2 shows common types of control loops and which types of control algorithms are typically used.

<table>
<thead>
<tr>
<th>Controlled Variable</th>
<th>Proportional Control</th>
<th>PI Control</th>
<th>PID Control</th>
</tr>
</thead>
<tbody>
<tr>
<td>Flow</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Level</td>
<td>Yes</td>
<td>Yes</td>
<td>Rare</td>
</tr>
<tr>
<td>Temperature</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Pressure</td>
<td>Yes</td>
<td>Yes</td>
<td>Rare</td>
</tr>
<tr>
<td>Analytical</td>
<td>Yes</td>
<td>Yes</td>
<td>Rare</td>
</tr>
</tbody>
</table>

Table 7.2: Control Loops and Control Algorithms

**Activities**

14. What type of control is used in an application where noise is present, but where no offset can be tolerated?

1. P only
2. PD
3. PI
4. PID