horizontal line

Task #6 REPORT ON PID

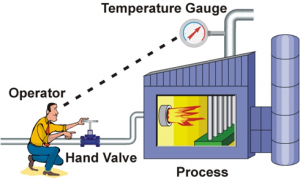
Tuesday, 11.09.2018

**─**

PID controllers are named after the Proportional, Integral and Derivative control modes they have. They are used in most automatic process control applications in industry. PID controllers can be used to regulate flow, temperature, pressure, level, and many other industrial process variables. This blog reviews the design of PID controllers and explains the P, I and D control modes used in them.

# Manual Control

Without automatic controllers, all regulation tasks will have to be done manually. For example: To keep constant the temperature of water discharged from an industrial gas-fired heater, an operator will have to watch a temperature gauge and adjust a fuel gas valve accordingly (Figure 1). If the water temperature becomes too high for some reason, the operator has to close the gas valve a bit – just enough to bring the temperature back to the desired value. If the water becomes too cold, he has to open the gas valve.



**Figure 1. An operator doing manual control.**

## **Feedback Control**

The control task done by the operator is called feedback control, because the operator changes the firing rate based on feedback that he gets from the process via the temperature gauge. Feedback control can be done manually as described here, but it is commonly done automatically, as will be explained in the next section.

## 

## **Control Loop**

The operator, valve, process, and temperature gauge forms a control loop. Any change the operator makes to the gas valve affects the temperature which is fed back to the operator, thereby closing the loop.

# 

# Automatic Control

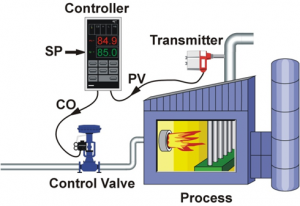
To relieve our operator from the tedious task of manual control, we should automate the control loop. This is done as follows:

* Install an electronic temperature measurement device.
* Automate the gas valve by adding an actuator (and perhaps a positioner) to it so that it can be driven electronically.
* Install a controller (in this case a PID controller), and connect it to the electronic temperature measurement and the automated control valve.

A PID controller has a Set Point (SP) that the operator can set to the desired temperature. The Controller’s Output (CO) sets the position of the control valve. And the temperature measurement, called the Process Variable (PV) gives the controller its much-needed feedback. The process variable and controller output are commonly transmitted via 4 – 20mA signals, or via digital commands on a Fieldbus.

When everything is up and running, the PID controller compares the process variable to its set point and calculates the difference between the two signals, also called the Error (E).

Then, based on the Error and the PID controller’s tuning constants, the controller calculates an appropriate controller output that opens the control valve to the right position for keeping the temperature at the set point. If the temperature should rise above its set point, the controller will reduce the valve position and vice versa.



**Figure 2. A PID controller doing automatic control.**

## **PID Control**

PID controllers have three control modes:

* Proportional Control.
* Integral Control.
* Derivative Control.

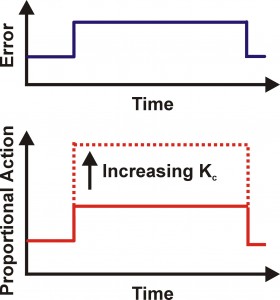
Each of the three modes reacts differently to the error. The amount of response produced by each control mode is adjustable by changing the controller’s tuning settings.

# Proportional Control Mode

The proportional control mode is in most cases the main driving force in a controller. It changes the controller output in proportion to the error (Figure 3). If the error gets bigger, the control action gets bigger. This makes a lot of sense, since more control action is needed to correct large errors.

The adjustable setting for proportional control is called the Controller Gain (Kc). A higher controller gain will increase the amount of proportional control action for a given error. If the controller gain is set too high the control loop will begin oscillating and become unstable. If the controller gain is set too low, it will not respond adequately to disturbances or set point changes.

Proportional Action



**Figure 3. Proportional control action.**

Adjusting the controller gain setting actually influences the integral and derivative control modes too. That is why this parameter is called *controller gain* and not *proportional gain*.

## **Proportional Band**

While most controllers use controller gain (Kc) as the proportional setting, some controllers use Proportional Band (PB), which is expressed in percent. Table 1 shows the relationship between Kc and PB.

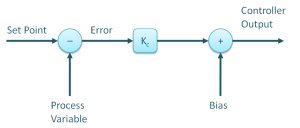
|  |  |
| --- | --- |
| **Controller Gain (Kc)** | **Proportional Band (PB) %** |
| 0.1 | 1000 |
| 0.2 | 500 |
| 0.5 | 200 |
| 1 | 100 |
| 2 | 50 |
| 5 | 20 |
| 10 | 10 |

**Table 1. Relationship between Kc and PB**

# Proportional-only Controller

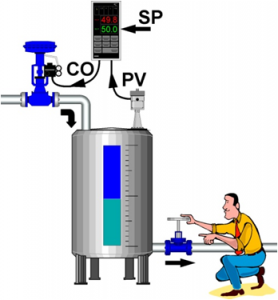
Proportional controllers are simple to understand and easy to tune. The controller output is simply the output of the proportional control mode, plus a bias. The bias is needed so that the controller can maintain an output (say at 50%) while there is no error (set point = process variable).

Proportional Controller



**Figure 4. A proportional-only controller algorithm.**

The use of proportional control alone has a large drawback – offset. Offset is a sustained error that cannot be eliminated by proportional control alone. For example, let’s consider controlling the water level in the tank in Figure 5 with a proportional-only controller. As long as the flow out of the tank remains constant, the level will remain at its set point.

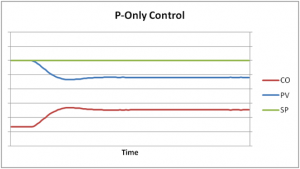


**Figure 5. Level control, with operator causing a disturbance.**

But, if the operator should increase the flow out of the tank, the tank level will begin to decrease due to the imbalance between inflow and outflow. While the tank level decreases, the error increases and our proportional controller increases the controller output proportional to this error. Consequently, the valve controlling the flow into the tank opens wider and more water flows into the tank.

As the level continues to decrease, the valve continues to open until it gets to a point where the inflow again matches the outflow. At this point the tank level (and error) will remain constant. Because the error remains constant our P-controller will keep its output constant and the control valve will hold its position. The system now remains at balance, but the tank level remains below its set point. This residual sustained error is called Offset.

Figure 6 shows the effect of a sudden decrease in fuel gas pressure to the process heater described earlier, and the response of a p-only controller. The decrease in fuel-gas pressure reduces the firing rate and the heater outlet temperature decreases. This creates and error to which the controller responds. However, a new balance-point between control action and error is found and the temperature offset is not eliminated by the proportional controller.



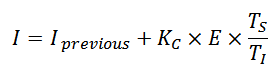
**Figure 6. A proportional controller’s response to a disturbance.**

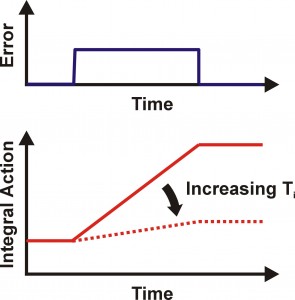
Under proportional-only control, the offset will remain until the operator manually changes the bias on the controller’s output to remove the offset. This is typically done by putting the controller in manual mode, changing its output manually until the error is zero, and then putting it back in automatic control. It is said that the operator manually “Resets” the controller.

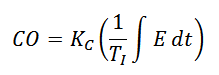
# Integral Control Mode

The need for manual reset as described above led to the development of automatic reset or the Integral Control Mode, as we know it today. As long as there is an error present (process variable not at set point), the integral control mode will continuously increment or decrement the controller’s output to reduce the error. Given enough time, integral action will drive the controller output far enough to reduce the error to zero.

If the error is large, the integral mode will increment/decrement the controller output fast, if the error is small, the changes will be slower. For a given error, the speed of the integral action is set by the controller’s integral time setting (TI). A large value of TI (long integral time) results in a slow integral action, and a small value of TI (short integral time) results in a fast integral action (Figure 7). If the integral time is set too long, the controller will be sluggish, if it is set too short, the control loop will oscillate and become unstable. In the figure, TS is the control algorithm’s execution interval, sometimes called sampling time or scan time.







**Figure 7. Integral control action and an integral-only controller’s equation.**

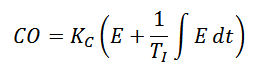
Most controllers use integral time in minutes as the unit of measure for integral control, but some others use integral time in seconds, integral gain in repeats per minute or repeats per second.

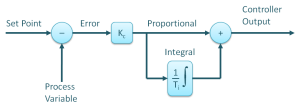
|  |  |  |  |
| --- | --- | --- | --- |
| **Integral Time** | | **Integral Gain** | |
| **Minutes** | **Seconds** | **Rep / Min** | **Rep / Sec** |
| 0.05 | 3 | 20 | 0.333 |
| 0.1 | 6 | 10 | 0.167 |
| 0.2 | 12 | 5 | 0.0833 |
| 0.5 | 30 | 2 | 0.0333 |
| 1 | 60 | 1 | 0.0167 |
| 2 | 120 | 0.5 | 0.00833 |
| 5 | 300 | 0.2 | 0.00333 |
| 10 | 600 | 0.1 | 0.00167 |
| 20 | 1200 | 0.05 | 0.00083 |

**Table 2. Units of the integral control mode.**

# Proportional + Integral Controller

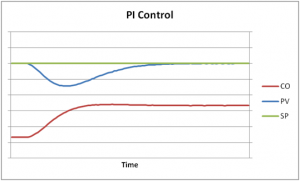
Commonly called the PI controller, its controller output is made up of the sum of the proportional and integral control actions (Figure 8).





**Figure 8. The PI controller algorithm.**

Figure 9 shows how the integral mode continues to increment the controller’s output to bring the heater outlet temperature back to its set point. Compared to Figure 6, it is clear how Integral control eliminates offset.



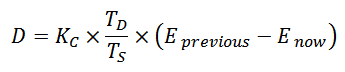
**Figure 9. A PI controller’s response to a disturbance.**

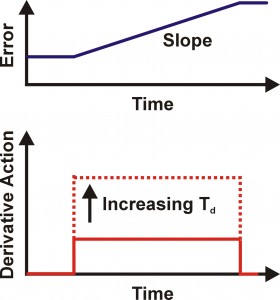
# 

# Derivative Control Mode

The third control mode in a PID controller is derivative. Derivative control is rarely used in controlling processes, but it is used often in motion control. For process control, it is not absolutely required, is very sensitive to measurement noise and it makes trial-and-error tuning more difficult. Nevertheless, using the derivative control mode of a controller can make a control loop respond a little faster than with PI control alone.

The derivative control mode produces an output based on the rate of change of the error (Figure 10). Derivative mode is sometimes called Rate. The derivative mode produces more control action if the error changes at a faster rate. If there is no change in the error, the derivative action is zero. The derivative mode has an adjustable setting called Derivative Time (TD). The larger the derivative time setting, the more derivative action is produced. A derivative time setting of zero effectively turns off this mode. If the derivative time is set too long, oscillations will occur and the control loop will run unstable. Again TS is the controller’s execution interval.





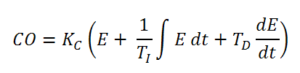
**Figure 10. Derivative control action.**

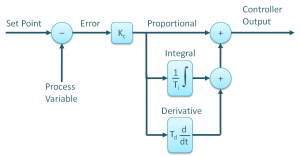
Two units of measure are used for the derivative setting of a controller: minutes and seconds.

# 

# Proportional + Integral + Derivative Controller

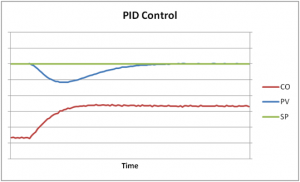
Commonly called the PID controller, its controller output is made up of the sum of the proportional, integral, and derivative control actions (Figure 11). There are other configurations too.



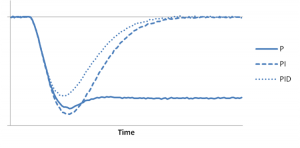


**Figure 11. The Standard (Non Interactive) PID controller algorithm.**

PID control provides more control action sooner than what is possible with P or PI control. This reduces the effect of a disturbance, and shortens the time it takes for the level to return to its set point.



**Figure 12. A PID controller’s response to a disturbance.**



**Figure 13. The result of a P, PI, and PID controllers’ response to a disturbance.**