PID Introduction

Process controls are necessary for designing safe and productive plants. A variety of process controls are used to manipulate processes, however the most simple and often most effective is the PID controller.

Much more practical than the typical on/off controller, PID controllers allow for much better adjustments to be made in the system. While this is true, there are some advantages to using an on/off controller:

-relatively simple to design and execute

-binary sensors and actuators (such as an on/off controller) are generally more reliable and less expensive

Although there are some advantages, there are large disadvantages to using an on/off controller scheme:

-inefficient (using this control is like driving with full gas and full breaks)

-can generate noise when seeking stability (can dramatically overshoot or undershoot a set-point)

-physically wearing on valves and switches (continuously turning valves/switches fully on and fully off causes them to become worn out much quicker)

To allow for much better control and fine-tuning adjustments, most industrial processes use a PID controller scheme.

The controller attempts to correct the error between a measured process variable and desired setpoint by calculating the difference and then performing a corrective action to adjust the process accordingly. A PID controller controls a process through three parameters: Proportional (P), Integral (I), and Derivative (D). These parameters can be weighted, or tuned, to adjust their effect on the process. The following section will provide a brief introduction on PID controllers as well as methods to model a controlled system in Excel.

The **Process Gain(K)** is the ratio of change of the output variable(responding variable) to the change of the input variable(forcing function). It specifically defines the sensitivity of the output variable to a given change in the input variable.

$$K = \frac{\Delta Output}{\Delta input}$$

Gain can only be described as a steady state parameter and give no knowledge about the dynamics of the process and is independent of the design and operating variables. A gain has three components that include the sign, the value, the units. The sign indicates how the output responds to the process input. A positive sign shows that the output variable increases with an increase in the input variable and a

negative sign shows that the output variable decreases with an increase in the input variable. The units depend on the process considered that depend on the variables mentioned.

Example:

The pressure was increased from 21psi to 29psi. This change increased the valve position from 30%vp to 22%vp.

```
K = (29-21)psi / ((22-30)%vp) = -1.0psi/(%vp)
```

Dead Time(t0) is the between the change in an input variable and when the output variable begins. Dead time is important because it effects the controllability of the control system. A change in set point is not immediate because of this parameter. Dead time must be considered in tuning and modeling processes.

Types of Control

Process controls are instruments used to control a parameter, such as temperature, level, and pressure. PID controllers are a type of continuous controller because they continually adjust the output vs. an on/off controller, when looking at feed forward or feed backward conditions. An example of a temperature controller is shown in Figure 1.
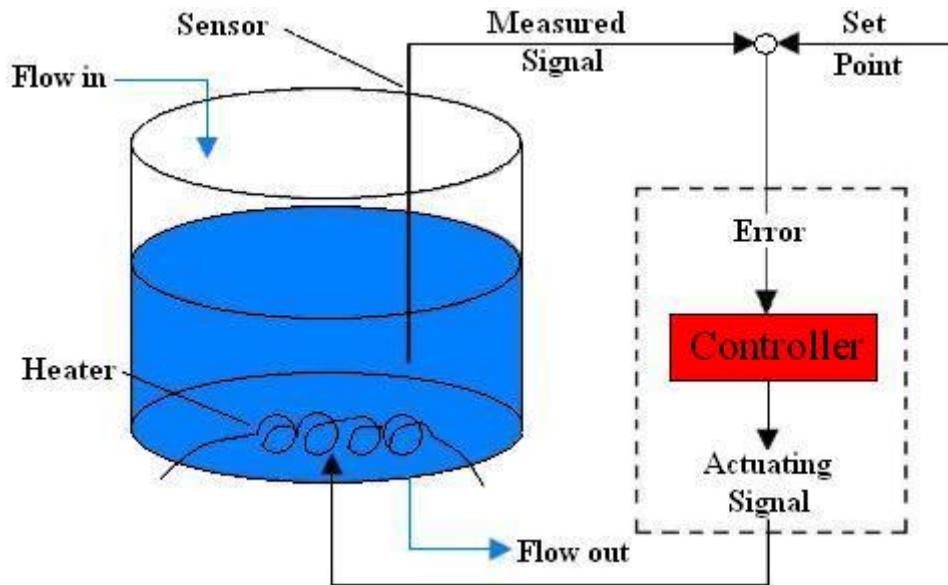
Figure 1. Temperature controller in a CSTR

As shown in Figure 1, the temperature controller controls the temperature of a fluid within a CSTR (Continuous Stirred Tank Reactor). A temperature sensor first measures the temperature of the fluid. This measurement produces a measurement signal. The measurement signal is then compared to the set point, or desired temperature setting, of the controller. The difference between the measured signal and set point is the error. Based on this error, the controller sends an actuating signal to the heating coil, which adjusts the temperature accordingly. This type of process control is known as error-based control

because the actuating signal is determined from the error between the actual and desired setting. The different types of error-based controls vary in the mathematical way they translate the error into an actuating signal, the most common of which are the PID controllers. Additionally, it is critical to understand feed-forward and feed-back control before exploring P, I, and D controls. [Feed Forward Control](#) [Feedback Control](#)

P, I, D, PI, PD, PID Control

As previously mentioned, controllers vary in the way they correlate the controller input (error) to the controller output (actuating signal). The most commonly used controllers are the proportional- integral- derivative (PID) controllers. PID controllers relate the error to the actuating signal either in a proportional (P), integral (I), or derivative (D) manner. PID controllers can also relate the error to the actuating signal using a combination of these controls.

Proportional (P) Control

One type of action used in PID controllers is the proportional control. Proportional control is a form of feedback control. It is the simplest form of continuous control that can be used in a closed-looped system. P-only control minimizes the fluctuation in the process variable, but it does not always bring the system to the desired set point. It provides a faster response than most other controllers, initially allowing the P-only controller to respond a few seconds faster. However, as the system becomes more complex (i.e. more complex algorithm) the response time difference could accumulate, allowing the P-controller to possibly respond even a few minutes faster. Athough the P-only controller does offer the advantage of faster response time, it produces deviation from the set point. This deviation is known as the offset, and it is usually not desired in a process. The existence of an offset implies that the system could not be maintained at the desired set point at steady state. It is analogous to the systematic error in a calibration curve, where there is always a set, constant error that prevents the line from crossing the origin. The offset can be minimized by combining P-only control with another form of control, such as I- or D- control. It is important to note, however, that it is impossible to completely eliminate the offset, which is implicitly included within each equation.

Mathematical Equations

P-control linearly correlates the controller output (actuating signal) to the error (diference between measured signal and set point). This P-control behavior is mathematically illustrated in Equation 1 (Scrcek, *et. al*).

$$c(t) = K_c e(t) + b$$

(1)
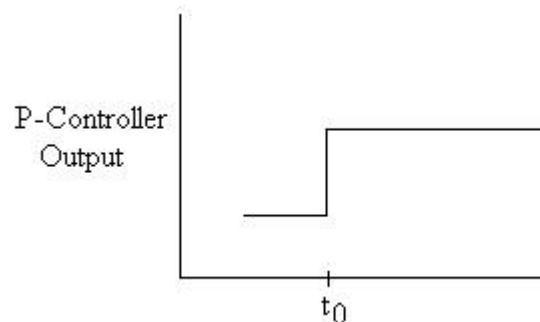
$c(t)$ = controller output

K_c = controller gain

$e(t)$ = error

b = bias

In this equation, the bias and controller gain are constants specific to each controller. The bias is simply the controller output when the error is zero. The controller gain is the change in the output of the

controller per change in the input to the controller. In PID controllers, where signals are usually electronically transmitted, controller gain relates the change in output voltage to the change in input voltage. These voltage changes are then directly related to the property being changed (i.e. temperature, pressure, level, etc.). Therefore, the gain ultimately relates the change in the input and output properties. If the output changes more than the input, K_c will be greater than 1. If the change in the input is greater than the change in the output, K_c will be less than 1. Ideally, if K_c is equal to infinity, the error can be reduced to zero. However, this infinitesimal nature of K_c increases the instability of the loop because zero error would imply that the the measured signal is *exactly* equal to the set point. As mentioned in lecture, exact equality is never achieved in control logic; instead, in control logic, error is allowed to vary within a certain range. Therefore, there are limits to the size of K_c, and these limits are defined by the system. Graphical representations of the effects of these variables on the system is shown in PID Tuning via Classical Methods.

As can be seen from the above equation, P-only control provides a linear relationship between the error of a system and the controller output of the system. This type of control provides a response, based on the signal that adjusts the system so that any oscillations are removed, and the system returns to steady-state. The inputs to the controller are the set point, the signal, and the bias. The controller calculates the difference between the set point and the signal, which is the error, and sends this value to an algorithm. Combined with the bias, this algorithm determines the action that the controller should take. A graphical representation of the P-controller output for a step increase in input at time t0 is shown below in Figure 2. This graph is exactly similar to the step input graph itself.

Figure 2. P-controller output for step input.

To illustrate this linear P-control relationship, consider the P-only control that controls the level of a fluid in a tank. Initially, the flow into the tank is equal to the flow out of the tank. However, if the flow out of the tank decreases, the level in the tank will increase because more fluid is entering than is leaving. The P-only control system will adjust the flow out of the tank so that it is again equal to the flow into the tank, and the level will once again be constant. However, this level is no longer equal to the initial level in the tank. The system is at steady-state, but there is a difference between the initial set point and the current position in the tank. This difference is the P-control offset.

Integral (I) Control

Another type of action used in PID controllers is the integral control. Integral control is a second form of feedback control. It is often used because it is able to remove any deviations that may exist. Thus, the system returns to both steady state and its original setting. A negative error will cause the signal to the system to decrease, while a positive error will cause the signal to increase. However, I-only controllers

are much slower in their response time than P-only controllers because they are dependent on more parameters. If it is essential to have no offset in the system, then an I-only controller should be used, but it will require a slower response time. This slower response time can be reduced by combining I-only control with another form, such as P or PD control. I-only controls are often used when measured variables need to remain within a very narrow range and require fine-tuning control. I controls affect the system by responding to accumulated past error. The philosophy behind the integral control is that deviations will be affected in proportion to the cumulative sum of their magnitude. The key advantage of adding a I-control to your controller is that it will eliminate the offset. The disadvantages are that it can destabilize the controller, and there is an integrator windup, which increases the time it takes for the controller to make changes.

Mathematical Equations

I-control correlates the controller output to the integral of the error. The integral of the error is taken with respect to time. It is the total error associated over a specified amount of time. This I-control behavior is mathematically illustrated in Equation 2 (Srcek, *et. al*).

$$c(t) = \frac{1}{T_i} \int e(t)dt + c(t_0)$$

(2)

c(t) = controller output

T_i = integral time

e(t) = error

$c(t_0)$ = controller output before integration

In this equation, the integral time is the amount of time that it takes for the controller to change its output by a value equal to the error. The controller output before integration is equal to either the initial output at time t=0, or the controller output at the time one step before the measurement. Graphical representations of the effects of these variables on the system is shown in PID Tuning via Classical Methods.

The rate of change in controller output for I-only control is determined by a number of parameters. While the P-only controller was determined by e, the rate of change for I-only depends on both e and T_i. Because of the inverse relationship between c(t) and T_i, this decreases the rate of change for an I-only controller.

The I-only controller operates in essentially the same way as a P-only controller. The inputs are again the set point, the signal, and the bias. Once again, the error is calculated, and this value is sent to the algorithm. However, instead of just using a linear relationship to calculate the response, the algorithm now uses an integral to determine the response that should be taken. Once the integral is evaluated, the response is sent and the system adjusts accordingly. Because of the dependence on T_i, it takes longer for the algorithm to determine the proper response. A graphical representation of the I-controller output for a step increase in input at time t0 is shown below in Figure 3. As expected, this graph represents the area under the step input graph.
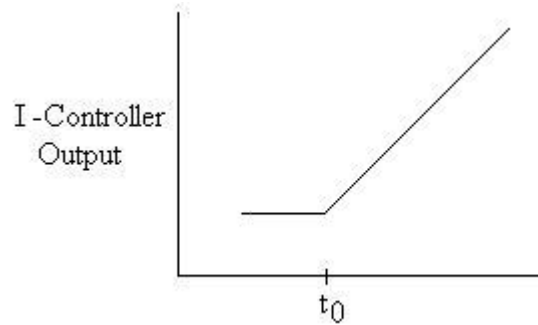
Figure 3. I-controller output for step input.

Derivative (D) Control

Another type of action used in PID controllers is the derivative control. Unlike P-only and I-only controls, D-control is a form of feed forward control. D-control anticipates the process conditions by analyzing the change in error. It functions to minimize the change of error, thus keeping the system at a consistent setting. The primary benefit of D controllers is to resist change in the system, the most important of these being oscillations. The control output is calculated based on the rate of change of the error with time. The larger the rate of the change in error, the more pronounced the controller response will be.

Unlike proportional and integral controllers, derivative controllers do not guide the system to a steady state. Because of this property, D controllers must be coupled with P, I or PI controllers to properly control the system.

Mathematical Equations

D-control correlates the controller output to the derivative of the error. The derivative of the error is taken with respect to time. It is the change in error associated with change in time. This D-control behavior is mathematically illustrated in Equation 3 (Scrcek, *et. al*).

$$c(t) = T_d \frac{de}{dt}$$

(3)

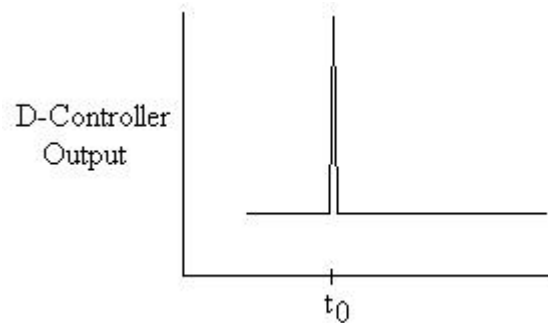$c(t)$ = controller output

T_d = derivative time constant

de = change in error

dt = change in time

Graphical representations of the effects of these variables on the system is shown in PID Tuning via Classical Methods.

Mathematically, derivative control is the opposite of integral control. Although I-only controls exist, D-only controls do not exist. D-controls measure only the change in error. D-controls do not know where the setpoint is, so it is usually used in conjunction with another method of control, such as P-only or a PI combination control. D-control is usually used for processes with rapidly changing process outputs. However, like the I-control, the D control is mathematically more complex than the P-control. Since it will

take a computer algorithm longer to calculate a derivative or an integral than to simply linearly relate the input and output variables, adding a D-control slows down the controller's response time. A graphical representation of the D-controller output for a step increase in input at time t0 is shown below in Figure 4. As expected, this graph represents the derivative of the step input graph.
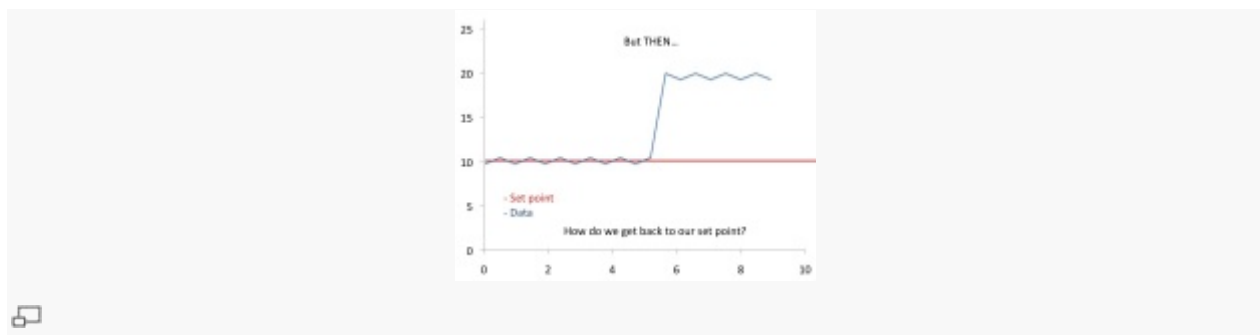

Figure 4. D-controller output for step input.

Controller Effects on a System

The following images are intended to give a visual representation of how P, I, and D controllers will affect a system.

Description
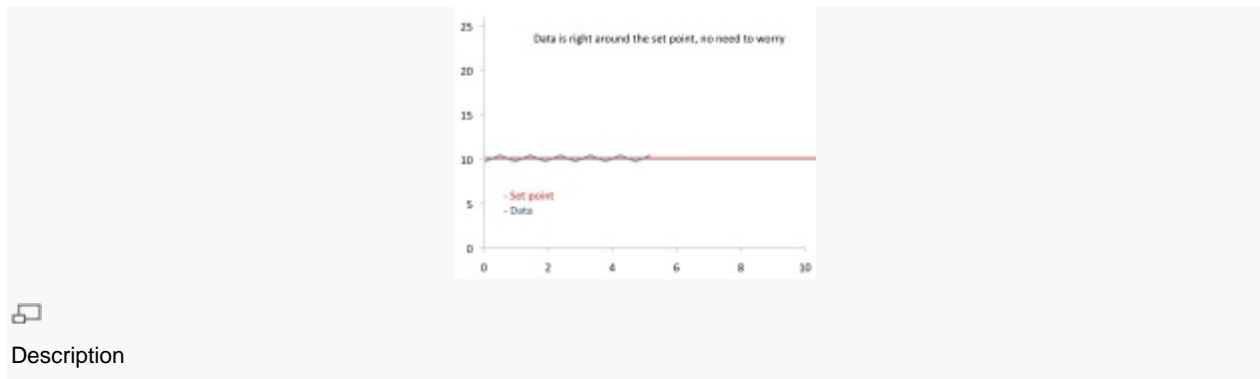
Figure 5. Stable data sample.
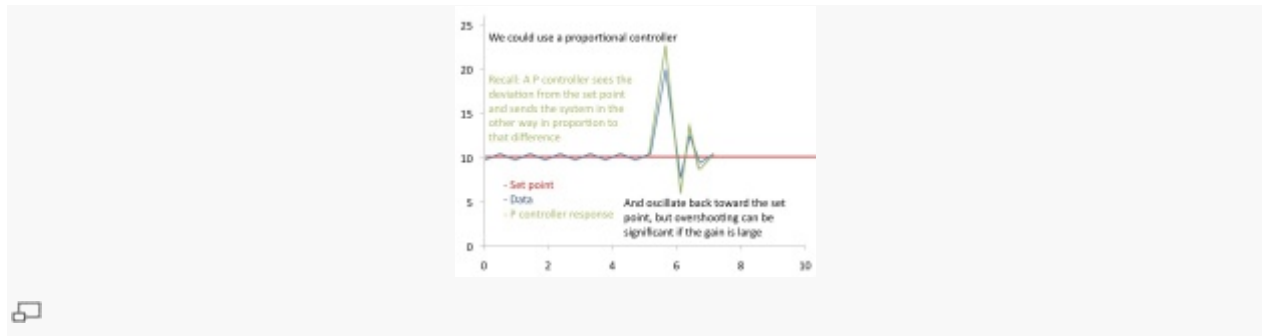

Figure 6. Data disturbance.

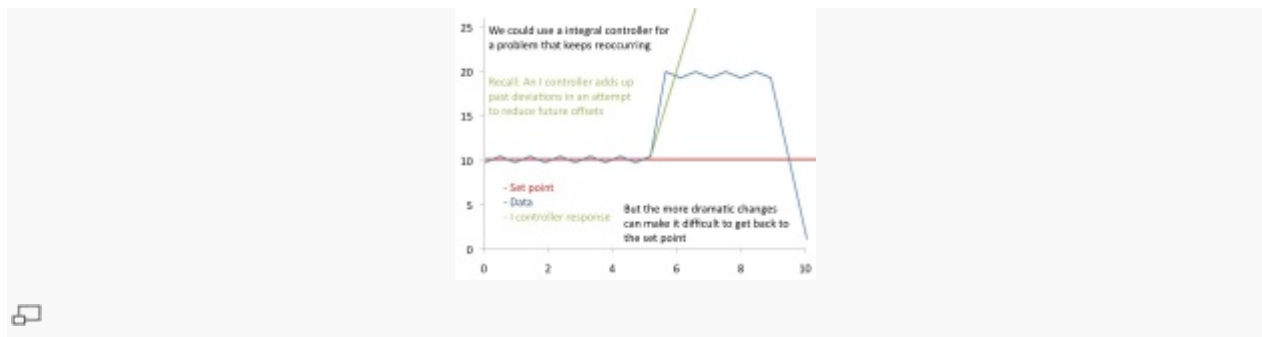**Figure 7.** P-controller effect on data.

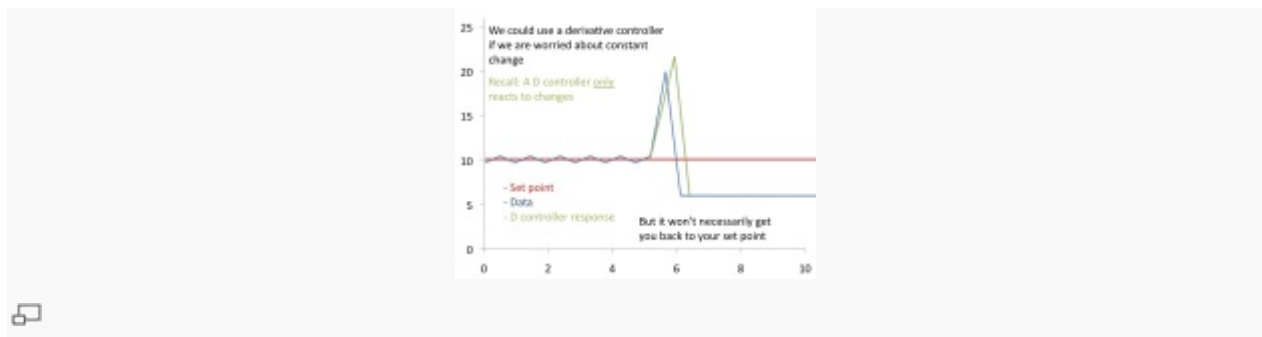**Figure 8.** I-controller effect on data.

Figure 9. D-controller effect on data.

Continue reading to see the results of combining controllers.

Proportional-Integral (PI) Control

One combination is the PI-control, which lacks the D-control of the PID system. PI control is a form of feedback control. It provides a faster response time than I-only control due to the addition of the proportional action. PI control stops the system from fluctuating, and it is also able to return the system to its set point. Although the response time for PI-control is faster than I-only control, it is still up to 50% slower than P-only control. Therefore, in order to increase response time, PI control is often combined with D-only control.

Mathematical Equations

PI-control correlates the controller output to the error and the integral of the error. This PI-control behavior is mathematically illustrated in Equation 4 (Scrcek, *et. al*).

$$c(t) = K_c \left(e(t) + \frac{1}{Ti} \int e(t)dt \right) + C$$

(4)

c(t) = controller output

K_c = controller gain

T_i = integral time

e(t) = error

C = initial value of controller

In this equation, the integral time is the time required for the I-only portion of the controller to match the control provided by the P-only part of the controller.

The equation indicates that the PI-controller operates like a simplified PID-controller with a zero derivative term. Alternatively, the PI-controller can also be seen as a combination of the P-only and I-only control equations. The bias term in the P-only control is equal to the integral action of the I-only control. The P-only control is only in action when the system is not at the set point. When the system is at the set point, the error is equal to zero, and the first term drops out of the equation. The system is then being controlled only by the I-only portion of the controller. Should the system deviate from the set point again, P-only control will be enacted. A graphical representation of the PI-controller output for a step increase in input at time t0 is shown below in Figure 5. As expected, this graph resembles the qualitatitive combination of the P-only and I-only graphs.
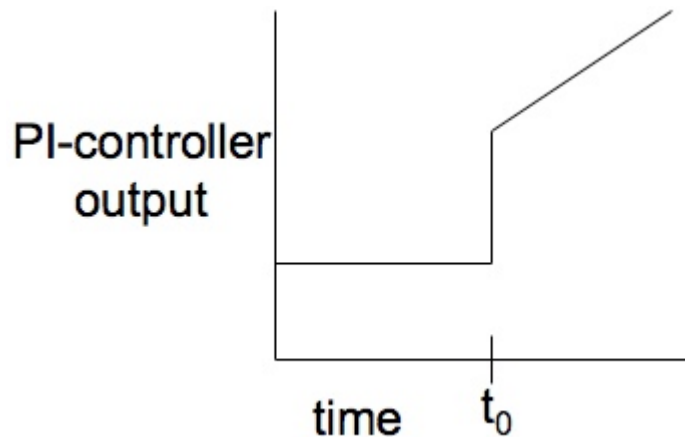
Figure 10. PI-controller output for step input.

Effects of K_c and T_i

With a PI control system, controller activity (aggressiveness) increases as K_c and T_i decreases, however they can act individually on the aggressiveness of a controller's response. Consider Figure 11 below with the center graph being a linear second order system base case.

Impact of Kc and T_i on Performance for PI Controller Form: $CO = CO_{bias} + Kc\,e(t) + \frac{Kc}{T_i}\int e(t)dt$
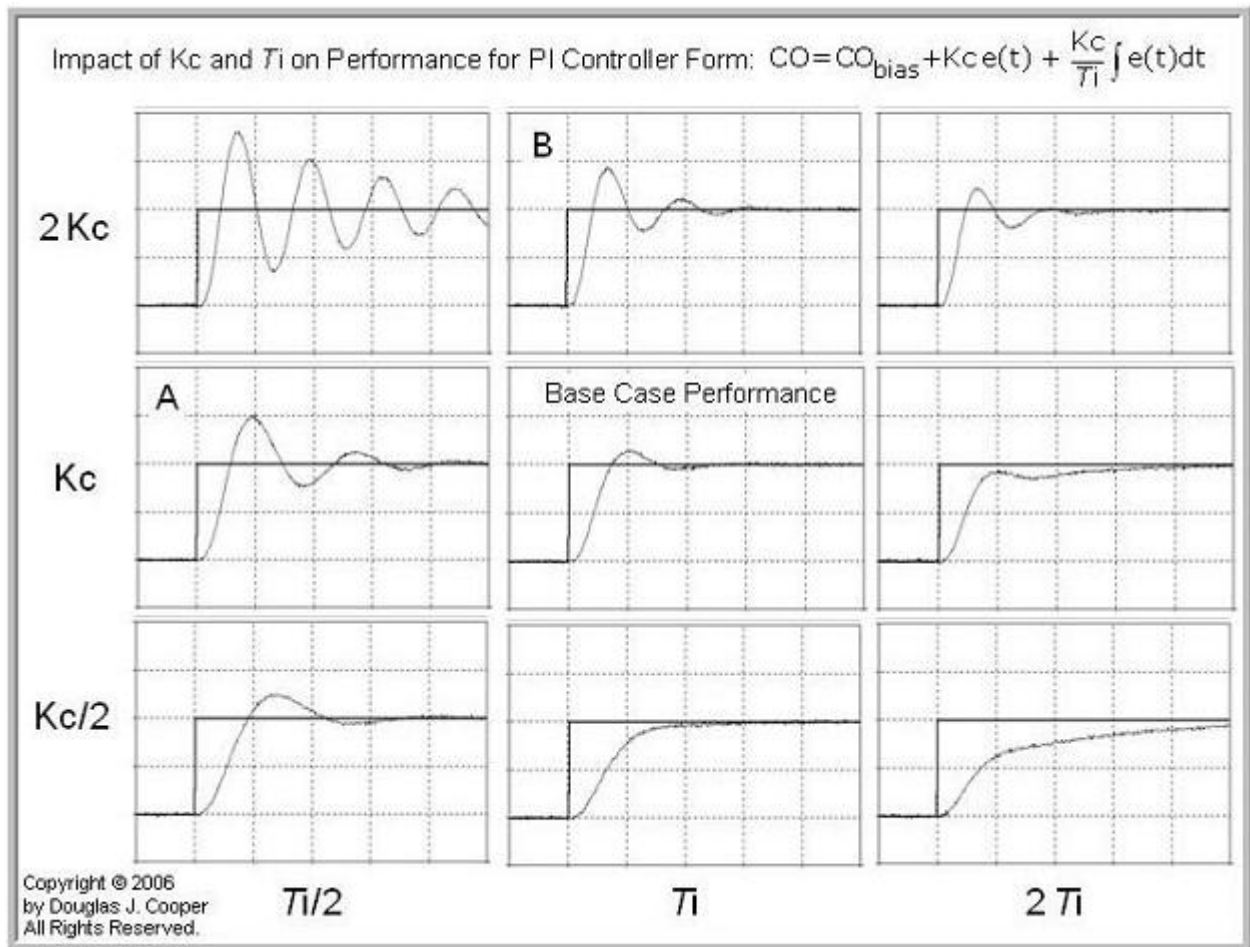
Figure 11. Effects of K_c and T_i [2]

The plot depicts how T_i and K_c both affect the performance of a system, whether they are both affecting it or each one is independently doing so. Regardless of integral time, increasing controller gain (moving form bottom to top on the plot) will increase controller activity. Similarly, decreasing integral time (moving right to left on the plot) will increase controller activity independent of controller gain. As expected, increasing K_c and decreasing T_i would compound sensitivity and create the most aggressive controller scenario.

With only two interacting parameters in PI control systems, similar performance plots can still cause confusion. For example, plots A and B from the figure both look very similar despite different parameters being affected in each of them. This could cause further problems and create a wildly aggressive system if the wrong parameter is being corrected. While trial and error may be feasible for a PI system, it

becomes cumbersome in PID where a third parameter is introduced and plots become increasingly similar.

Another noteworthy observation is the plot with a normal K_c and double T_i. The plot depicts how the proportional term is practical but the integral is not receiving enough weight initially, causing the slight oscillation before the integral term can finally catch up and help the system towards the set point.

Proportional-Derivative (PD) Control

Another combination of controls is the PD-control, which lacks the I-control of the PID system. PD-control is combination of feedforward and feedback control, because it operates on both the current process conditions and predicted process conditions. In PD-control, the control output is a linear combination of the error signal and its derivative. PD-control contains the proportional control's damping of the fluctuation and the derivative control's prediction of process error.

Mathematical Equations

As mentioned, PD-control correlates the controller output to the error and the derivative of the error. This PD-control behavior is mathematically illustrated in Equation 5 (Scrcek, *et. al*).

$$c(t) = K_c \left(e(t) + T_d \frac{de}{dt} \right) + C$$

(5)

c(t) = controller output

K_c = proportional gain

e = error

C = initial value of controller

The equation indicates that the PD-controller operates like a simplified PID-controller with a zero integral term. Alternatively, the PD-controller can also be seen as a combination of the P-only and D-only control equations. In this control, the purpose of the D-only control is to predict the error in order to increase stability of the closed loop system. P-D control is not commonly used because of the lack of the integral term. Without the integral term, the error in steady state operation is not minimized. P-D control is usually used in batch pH control loops, where error in steady state operation does not need to be minimized. In this application, the error is related to the actuating signal both through the proportional and derivative term. A graphical representation of the PD-controller output for a step increase in input at time t0 is shown below in Figure 6. Again, this graph is a combination of the P-only and D-only graphs, as expected.
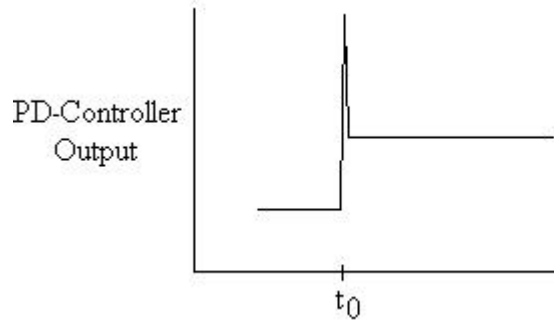
Figure 12. PD-controller output for step input.

Proportional-Integral-Derivative (PID) Control

Proportional-integral-derivative control is a combination of all three types of control methods. PID-control is most commonly used because it combines the advantages of each type of control. This includes a quicker response time because of the P-only control, along with the decreased/zero offset from the combined derivative and integral controllers. This offset was removed by additionally using the I-control. The addition of D-control greatly increases the controller's response when used in combination because it predicts disturbances to the system by measuring the change in error. On the contrary, as mentioned previously, when used individually, it has a slower response time compared to the quicker P-only control. However, although the PID controller seems to be the most adequate controller, it is also the most expensive controller. Therefore, it is not used unless the process requires the accuracy and stability provided by the PID controller.

Mathematical Equations

PID-control correlates the controller output to the error, integral of the error, and derivative of the error. This PID-control behavior is mathematically illustrated in Equation 6 (Scrcek, *et. al*).

$$c(t) = K_c \left(e(t) + \frac{1}{T_i} \int e(t)dt + T_d \frac{de}{dt} \right) + C$$

(6)

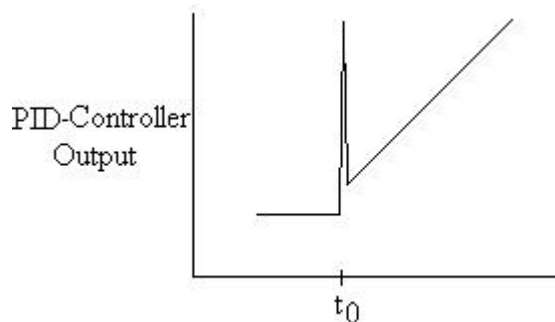c(t) = controller output

K_c = controller gain

e(t) = error

T_i = integral time

T_d = derivative time constant

C = intitial value of controller

As shown in the above equation, PID control is the combination of all three types of control. In this equation, the gain is multiplied with the integral and derivative terms, along with the proportional term, because in PID combination control, the gain affects the I and D actions as well. Because of the use of derivative control, PID control cannot be used in processes where there is a lot of noise, since the noise would interfere with the predictive, feedforward aspect. However, PID control is used when the process

requires no offset and a fast response time. A graphical representation of the PID-controller output for a step increase in input at time t0 is shown below in Figure 7. This graph resembles the qualitative combination of the P-only, I-only, and D-only graphs.

Figure 7. PID-controller output for step input.

In addition to PID-control, the P-, I-, and D- controls can be combined in other ways. These alternative combinations are simplifications of the PID-control.

Note: Order of e(t)

The order of the elements in the e(t) can vary depending on the situation. It could be the fixed element minus the varying element or the other way around. To better illustrate the concept let's go to an example. Let's say you are creating a PID control to control the fluid level in a tank by manipulating the outlet valve. When the fluid level in the tank exceeds your set value, you will want the valve to open up more to allow more flow out of the tank. You are looking for a positive response. Therefore your e(t) should give a positive value when the fluid level is higher than the set. In this case your e(t) will be (V-Vset). The same logic can be used for other systems to determine what the e(t) should be in the PID controls.

Modeling PID Controllers With Euler in Excel

As with many engineering systems, PID controllers can be modeled in Excel via numerical methods such as Euler's Method. First begin with the initial value for a given parameter. Determine the change in that parameter at a certain time-step by summing the three controllers P, I, and D at that step, which are found using the equations listed in the P, I, D, PI, PD, PID Control section above. Take this change, multiply it by the chosen time-step and add that to the previous value of the parameter of interest. For more detailed information see Numerical ODE Solving in Excel. An example of a chemical engineering problem that uses this method can be seen in Example 4 below.

Troubleshooting PID Modeling in Excel

When setting up an Excel spreadsheet to model a PID controller, you may receive an error message saying that you have created a circular reference. Say you are controlling the flow rate of one reactant (B) to a reactor which is dependent upon the concentration of another reactant (A) already inside the reactor. Your PID equations look as follows:

$$FB = 1 + K_c \left[(A - A_{\text{set}}) + \frac{1}{\tau_i} \int (A - A_{\text{set}}) dt + \tau_d \frac{d(A - A_{\text{set}})}{dt} \right]$$

$$x_i = \frac{d(A - A_{\text{set}})}{dt}$$

After you have set up your columns for A - Aset, d(A - Aset)/dt, xi, and the cells for your parameters like Kc, tau*i* and tau*d*, you will need to set up your PID column with your PID equation in it. After entering your equation into the first cell of the PID column, you may receive the Circular Reference error message when trying to drag the equation down to the other cells in your column.

There are two things you can do:

1. It is likely that you need to start your PID equation in the second or third cell of your PID column. Enter reasonable values into the first couple of cells before you start the PID equation, and you will find that these values shouldn't affect the end result of your controller.
2. You can also try decreasing the step size (Δt).

Summary Tables

A summary of the advantages and disadvantages of the three controls is shown below is shown in Table 1.

Table 1. Advantages and disadvantages of controls

	Proportional (P)	Integral (I)	Derivative (D)
Advantages	+ Fast response time	+ Contains small offset	+ Keeps system at consistent se
	+ Minimizes fluctuation	+ Returns system to steady state	+ Controls processes with rapid changing outputs
Disadvantages	- Contains large offset	- Slow response time	- Slow response time
	- Does not bring system to desired set point		- Requires combined use with another controller

A guide for the typical uses of the various controllers is shown below in Table 2.

Table 2. Typical uses of P, I, D, PI, and PID controllers

Controller	Estimates	When to use	Examples
P	Present	Systems with slow response, systems tolerant to offset	Float valves, thermostats, humidistat
I	Back	Not often used alone, as is too slow	Used for very noisy systems
D	Forward	Not used alone because it is too sensitive to noise and does not have set point	None
PI	Present & back	Often used	Thermostats, flow control, pressure control
PID	All time	Often used, most robust, but can be noise sensitive	Cases where the system has intertia that could get out of hand: i.e. temperature and concentration measurements on a reactor to avoid runaway.

A summary of definitions of the terms and symbols are shown below in Table 3.

Table 3. Definitions of terms and symbols.

Term or Symbol	Defintion
Bias	Controller output when the error is equal to zero
Change in Error, de	Differential change in error used in the the Derivative Action equation
Change in Time, dt	Differential change in time used in the Deriative Action equation
Controller	Instrument used to adjust a system property such as temperature, pressure, Controllers vary in three main aspects: (1) The way they correlate error and actuating signal (such as PID controller (2) The way they work relative to the data they gather (feedback Vs. feedfo (3) The way the adjust the system setting (on-off Vs. continuous)
Controller Gain, Kc	Ratio of the change in output to the change in input
Controller Output, c(t)	Voltage output from the controller, usually relating to the actuating signal
Controller Output before Integration, c(t0)	Constant used in the Integral Action equation
Derivative Time Constant, Td	Constant used in the Deriviative Action equation
Error, e(t)	The difference between the measured signal and the set point
Offset	Systematic error that prevents the system from being at the desired set point during steady state
Integral Time, Ti	The time it takes the controller to change the output by a value equal to the error

Example 1

Hypothetical Industries has just put you in charge of one of their batch reactors. Your task is to figure out a way to maintain a setpoint level inside of the reactor. Your boss wants to use some type regulator controller, but he is not quite sure which to use. Help your boss find the right type of controller. It is extremely important that the level inside the reactor is at the setpoint. Large fluctuation and error cannot be tolerated.

SOLUTION:
You would want to use a PID controller. Because of the action of P control, the system will respond to a change very quickly. Due to the action of I control, the system is able to be returned to the setpoint value.

Finally, because it is so critical for the system to remain at a constant setpoint, D control will measure the change in the error, and help to adjust the system accordingly.

Source : https://controls.engin.umich.edu/wiki/index.php/PIDIntro