USER MANUAL

PITOPS-PID VER. 6.1

(PROCESS IDENTIFICATION & CONTROLLER TUNING OPTIMIZER SIMULATOR)

PID

PID AND ADVANCED CONTROL SOFTWARE

INDUSTRIAL PROCESS CONTROL SOFTWARE FOR DCS/PLC PID TUNING AND ADVANCED PROCESS CONTROL DESIGN & OPTIMIZATION

PICONTROL SOLUTIONS COMPANY

PART A: **SOFTWARE INSTALLATION AND GETTING STARTED**

CONTENTS

PART A: **SOFTWARE INSTALLATION AND GETTING STARTED**

1.0 INTRODUCTION

Part A specifies the hardware required to run Pitops™-PID software. It also explains software installation procedure and how to start and use Pitops™-PID.

2.0 SOFTWARE REQUIREMENTS AND INSTALLATION

Pitops™-PID runs on Windows XP, Windows NT, and Windows 98/2000 operating systems. Microsoft® Excel software is essential for running Pitops™-PID.

To install Pitops™-PID, download program installation setup exe file from picontrolsolutions website.

To start Installation, double-click **PITOPS_PID_SETUP.exe** file. To complete installation, follow all the step by step installation instructions on the screen. All Pitops™-PID files will be installed to the specified program folder. A program Group “PITOPS-PID” will be created with program icons “PITOPS PID, Readme First, Help and License Agreement.

3.0 SIMPLICITY OF PITOPS AND RELATED PRODUCTS

Pitops™ is very simple to use for any plant operator, control engineer, DCS/PLC technician or researcher. Pitops™ works entirely in the time-domain (seconds, minutes, etc.) It does not use the more complicated "***s***" (Laplace) or the "***Z***" (discrete) domains.

Use of Pitops™ does not require deep academic knowledge of process control theory. This manual does not cover details of process control theory and fundamentals. Users of Pitops™ who would like to learn about process control theory in detail are referred to PiControl's Process Control CBT (computer-based training) module, which covers both primary process control and advanced process control theory and fundamentals.

The users are also referred to Simcet^{TM}, Apromon^{TM} and Tadpole^{TM} products, all designed to help you improve process control at your plant in many new and innovative ways.

Simcet^{TM} is a real-time dynamic simulator that provides a real-plant like tuning environment.

Apromon^{TM} is a process and controller health monitoring and diagnostics software that helps to identify poorly performing controllers and then provides a structured methodology to improve the controller performance.

Tadpole ^{TM} is equipped with revolutionary, novel mathematics to reliably detect process oscillations.

For more information on all these products, visit the website www.picontrolsolutions.com.

4.0 GETTING STARTED ON USING PITOPS^{TM}

Pitops^{TM} software consists of two modules- Pitops-PID and Pitops-TFI.

PID stands for *PID Control Tuning and Design*. PID module simulates PID controllers, cascade PIDs, feedforward loops and Dead Time Compensator. Various other features are provided for primary and advanced control tuning and design. This document covers the PID module only.

TFI stands for *Transfer Function Identification*. TFI module identifies transfer functions using time-series plant data. Another separate document and software covers the TFI module.

PART B: **EXAMPLES ON PID AND FEEDFORWARD TUNING**

CONTENTS

PART B: **EXAMPLES ON PID AND FEEDFORWARD TUNING**

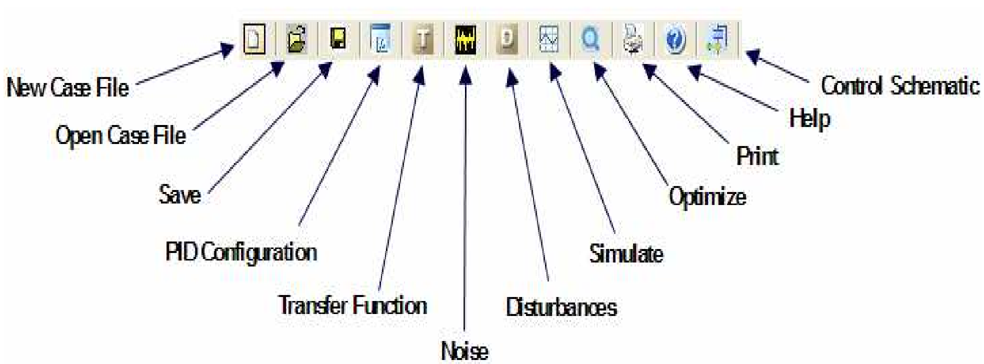
1.0 INTRODUCTION

This part helps the user to get started on how to tune PIDs using Pitops-PID. Illustrative examples are provided for tuning FC, TC, LC and cascade PID. Examples are provided also on feedforward and model-based controller strategies. Each example consists of detailed step-by-step procedure on how to configure Pitops.

Pull-down menu bars are located near the top left section of the screen and are labeled as: **| File | Inputs | Run | Help |**.

Click on the above pull-down menus and you will see the various pull-down menu options.

Under the pull-down menu bar is located the icon tool bar. Everything from the pull-down menus can be also conveniently accessed using the icon tool bar. The icon tool bar labels are shown below. After starting Pitops, locate the cursor on each icon to see its label.

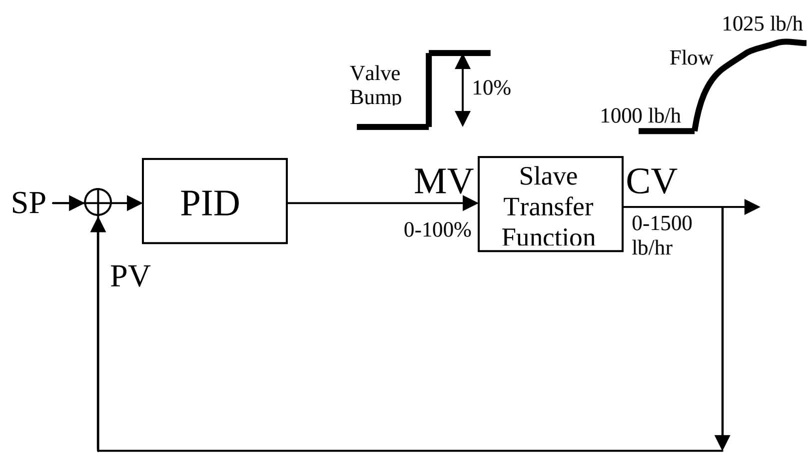


The examples below are designed to help you quickly start using the Pitops-PID program and put it to practical use. Detailed explanation on each of the icons, tool-bar options and all user-entered screen fields are explained in Part C.

2.0 CONFIGURE A TRANSFER FUNCTION

This section helps the user to configure a transfer function and watch its open loop response. A typical flow PID (FC) is used for study. The FC is in manual mode. Make 10% change in the PID output (valve position). Initial value of flow is 1000 lb/h. After a dead time of 30 seconds, flow rises to 1025 lb/h. Time constant is 40 seconds. The FC flowmeter range is 0-1500 lb/h. See Figure 1 below:

Figure 1. Transfer Function Configuration



The procedure to configure a transfer function is given below:

1. Start Pitops-PID by double-clicking on the Pitops-PID icon.

2. Click on File and then click on New Case File. Click Yes in response to the pop-up message. This clears all data from a previous Pitops run, initializes all variables to some default values and prepares for a new Pitops run. This is recommended before the start of every new Pitops run.

3. In the Process Transfer Function window on right hand side of

the screen, set Delay = 30, Gain = 2.5, Tau1 = 40. (Gain is calculated as (1025 -1000 lb/h) / 10%.

4. Click on Inputs, then click on PID Configuration and then click on Single PID, set CV Range High = 1500 (lb/hr), CV Range Low = 0. MV Range 0-100% represents valve range, leave default values.

5. On right side of the screen, set both SP Old and SP New = 1000 (lb/hr).

6. Also on right side of the screen, set Valve Bump = 10 (%).

7. Under Tuning, set P = 0. Proportional Gain is set to zero (no PID action); PID is now in Manual mode. PID equation may be modified by clicking on Inputs, PID Configuration, Single PID and PID Equation and then selecting appropriate equation.

8. Near the bottom of the screen, set Simulation Time = 200. This means we want to see the simulation for 200 time units.

9. Click on Run and then click on Simulate. The ***top window*** shows the PID PV (red) and SP (blue). The ***middle window*** shows PID OP (output).

   See the transfer function response. At time = 0 (start time of simulation), the valve position is increased from 50 to 60%. After a 30 second delay, the flow starts rising from 1000 lb/hr. Notice that 40 seconds are required to reach 63% of the total change from 1000 to 1025 lb/hr (1016).

   The above curve is called the open loop response. Examine the plot and check the transfer function parameters based on the shape of the curve.

10. **Zoom** and **Drag** feature: Inside any plot window, click with ***LEFT mouse button*** on top left corner of the area to be zoomed. Keeping the button pressed, move the mouse to specify the bottom right corner of the area to be zoomed. A

rectangle marks the selected area. Release the left mouse button. The section is zoomed.

Double click to **restore** the full original plot after zooming.

Locate cursor arrow anywhere on the plot and press the ***RIGHT mouse button***. Move the mouse up, down, right or left. The entire plot can be scrolled. This feature works also on a zoomed section.

Double click to **restore** the original plot after scrolling.

11. Notice that PID OP (valve position) was 50% at time = 0 and after the 10% bump increases to 60%. The 50% initial position is a default parameter and rarely needs to be changed. If one desires to change this, click on <u>Inputs</u>, <u>PID Configuration</u>, <u>Single PID</u>, and <u>Initial PID Output</u> and type in a new value.

12. On right side of the screen, type in different transfer function parameters and study the different open loop responses. Click on <u>Run</u> and then click on <u>Simulate</u> after making any data changes to see the new simulation.

13. To study a second order response, type in <u>Tau2</u> = 5 keeping other parameters the same). Click <u>Simulate</u>. Notice the second order wavy transfer function shape.

14. The PV and OP trends can be exported to an Excel data file. Click on <u>File</u> and then click on <u>Generate Vector Files</u>. Specify a filename in which to export the screen trends.

    Inside the Excel file, the first column contains time of file creation. The time samples are all constant. This first column containing time stamps is not useful as is, but is created to keep the file structure the same as the files used by the Pitops-TFI module that identifies transfer functions using plant data files.

    The second column is labeled <u>CV</u> (Controlled Variable) and contains the <u>PV (Process Variable</u>) data.

The third column is labeled MV1 (Manipulated Variable #1) and contains the PID Raw Output data.

The fourth column is labeled MV2 (Manipulated Variable #2) and contains the Disturbance PV data.

The Excel file consists of a three line header at the top followed by 200 data points (Simulation Time specified in Step 8 was 200), at unit time sample intervals. The number of data points can be changed by changing the simulation time. The three line header at the top can be changed to add more or less lines (rows); refer to the procedure explained in Part C, Section 13.4.

The Excel file generation feature can be used to improve model predictions in MPC (model predictive control) systems. First, the user will identify transfer functions using the Transfer Function Identification option provided in the sister module (Pitops-TFI) and then generate an Excel file using the procedure described in this section.

The complete simulation can be restored with the case file Ex1.pid (supplied with Pitops-PID software). To restore the case file, click on File, Open Case File, navigate to the desired subdirectory where Pitops software was installed and then select the file Ex1.pid and click on Open to complete reading the file data.

3.0 TUNE A PID WITH A SETPOINT CHANGE

In this section, we will configure a PID (FC) and then simulate a Setpoint (SP) change with various PID tuning parameters. Pitops-PID will determine optimum tuning parameters using the IAE (**I**ntegral **A**bsolute **E**rror) criterion). You can try different sets of PID tuning parameters and see their effect in the Pitops simulation.

1. Continue from Section 1 above. Make sure that Delay = 30, Gain = 2.5, Tau1 = 40, Tau2 = 0. Change Valve Bump to 0.0 (now we wish to observe the PID action on the valve; the valve

bump is used only to see open loop response, never when PID is active).

2. Change SP New to 1030 lb/hr. SP Old should be 1000. We wish to simulate a setpoint change from 1000 to 1030 with the PID in automatic mode.

3. Under Tuning, set P = 0.5, set I = 15. Note that the time unit assumed here is seconds. All transfer function parameters (dead time and time constant), simulation time and Integral constant (I) are all in seconds.

4. Near bottom of the screen, change Simulation Time to 500 (seconds). We wish to watch the PID control action for 500 seconds after the setpoint change from 1000 to 1030 lb/hr.

5. Click on the Simulate icon  in the top-bar. Examine the plot.

   The ***top window*** shows the SP and PV trends. The blue trend in the top window shows the setpoint change. The red trend in top window shows the actual change in flow (flow PV).

   The ***middle window*** shows the PID Output (OP) going to the control valve. The PID output manipulates the valve to achieve the flow change.

   The ***bottom window*** shows the individual P (Proportional), I (Integral) and D (Derivative) contributions. The sum of these three contributions comprise the total PID controller's control action. Note that if you do not see "PID Contribution" in the bottom window, then click on the button labeled "PID Contrib" near the bottom right corner of the screen. This button toggles from displaying PID Contribution and PID Raw Output. Both these trends are explained in a later section.

   Note the Error value shown near the bottom right corner of screen. This value is the integrated error between PV and SP calculated over the simulation time. The default error criterion is the IAE (integral absolute error).

6. The PID control action appears rather slow (sluggish). Let's try more aggressive tuning. Change P from 0.5 to 1.0. Click Simulate. A small **overshoot** can be seen now.

   Click the Optimize icon to run the optimizer. The Pitops optimizer determines new tuning parameters (this may take several seconds). Notice that now the Proportional Gain (P) is changed to around 5.5 and the Integral (I) is changed to about 50.

   The Pitops optimizer determines new and improved tuning parameters based on the selected error criterion. This example uses the IAE criterion. To see or change the error criterion, click on the PID Configuration icon and then click on Error Criteria. You will see four options – IAE, ISE, ITAE and Reduced Overshoot (these are all explained in detail in Section C).

   Generally, the IAE criterion works well but may produce rather aggressive control action. The ISE and ITAE criteria may also produce aggressive tuning. Use the Reduced Overshoot error criterion for generating less aggressive and more practical tuning parameters, suitable for use in the real plant.

   In the Tuning box, if D (Derivative is zero), optimizer will keep D at zero (a new value will not be calculated).

7. Note the value of Current ROC (Rate of Change) located near the bottom right corner of the screen. The current ROC is the maximum change in the PID output during two consecutive PID executions (scan times). Excessive valve changes may be undesirable in a given process. The Current ROC is a measure of how fast the valve is moved by the PID control action. Based on process knowledge, user can deduce that valve movement is excessive. PID tuning then may be modified to reduce the Current ROC.

   Notice the large change in PID OP at time = 0. This is called the **Proportional Kick** and is entirely due to the proportional contribution. Higher the P, larger will be the proportional kick.

Notice that the ROC value is equal to the proportional kick as seen from the plot.

8. Is the proportional kick too large here? It depends on the process characteristics. Let's say, based on our process knowledge, we want to limit the ROC from 12 to 6. Change the Max ROC from 10000 to 6. We do not want the PID OP to change by more than 6% during any two consecutive cycles.

   Now we can search for new PID values with help from the optimizer. Set P = 1.0, I = 20, D = 0. Click on Simulate. Notice that now the Current ROC is well inside the 6.0 Max ROC limit. Click on Optimize. After completion of optimizer, note that now Current ROC = Max ROC = 6.0. The new tuning parameters are: P = 2.9, I = 34. Notice that these parameters are less aggressive than before. Note also that in the middle plot, near time = 0, the PID OP jumps from 50% to about 56%. This difference (56 – 50 = 6) is the current ROC of 6.0. The Pitops-PID optimizer thus satisfied the 6% Max ROC constraint while determining the optimum PI tuning parameters.

9. Try different PI values to see if you can further minimize the Error value. Try Derivative from 0-3 and watch the effect on the setpoint change.

10. The IAE minimum value determined by Pitops gave a lower error, but moved the valve too quickly. P = 3 and I = 30 are more reasonable and practical, both smooth and stable.

    This complete simulation can be restored by clicking on the Open Case File icon and then selecting Ex2.pid.

4.0 ADD RANDOM NOISE

Random noise is always present in plant signals. Flows are particularly very noisy. We add random noise in this section to make a more realistic simulation. Also, we study derivative action here; this is meaningful only when the right amount of noise level is present in the simulation.

1. Continue from end of Section 3 above. Set P = 3, I = 30. Click on the Noise icon . Click Up-Arrow for Slave PV several times until the value is about 0.5. This means that noise band is 0.5%. Click OK on Noise menu, and click Simulate .

   Notice now that the flow (CV) looks more realistic with the random superimposed noise.

2. Inside the Tuning box, set D = 5.0 (Derivative). Click Simulate . Examine the plots. Notice the large fluctuations in the valve (PID OP), the middle plot. These are caused by the high derivative action and the noise. Try different sets of derivative constant from 0 to 20.

   Note: Generally, derivative on FCs is set to zero. In this example, derivative is used for illustrative reasons only. Generally speaking, derivative action is not used for FCs.

3. Set P = 3, I = 30 and D = 1. Click Simulate . Note the Error value and the shape of PV, OP trends.

   Make sure the ROC constraint is disabled (to disable, set Max ROC = 10000).

   Cancel noise by clicking on Noise , Yes for Cancel All Noise. Click OK.

   Click Optimize . Notice that now a D (Derivative) value is also calculated. In the absence of noise and with just a setpoint change, D can be too high as will be seen in this case.

   Derivative action can add to the stability of the loop, allowing a higher gain (P), provided that the PV signal is not excessively noisy.

Add different amount of noise again and see the impact of derivative action.

This complete simulation can be restored by clicking on the Open Case File icon and then selecting Ex3.pid.

5.0 CONFIGURE EXTERNAL DISTURBANCE

In this section, we configure an external disturbance and then tune the PID to respond well to both a disturbance and a setpoint change. PID tuning should always be determined based on both setpoint change and disturbances. Tuning that is determined for only one case may be too sluggish or unstable in the other case. The control configuration for the external disturbance is shown in Figure 2.

1. Continue from end of Section 3 or 4 or start a fresh run by clicking on the New Case File icon and click Yes.

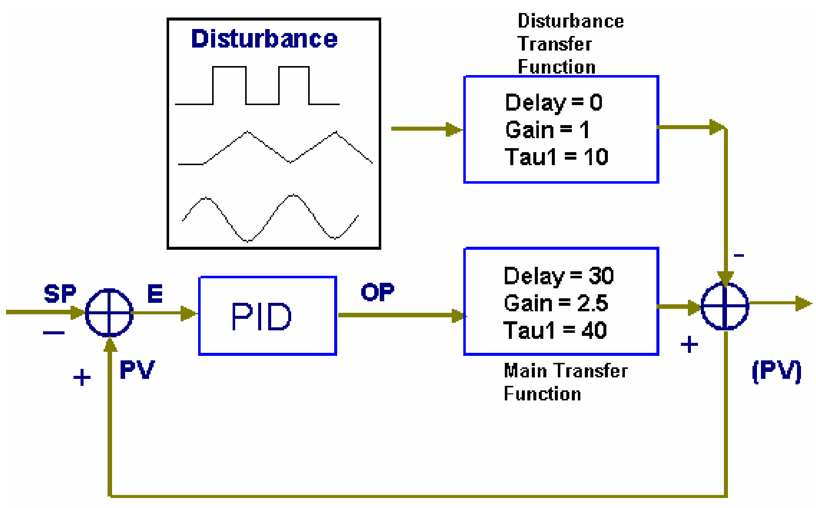
2. Change P = 3.5, I = 40, D = 0, Max ROC = 10000, Valve Bump = 0. Under Process Transfer Function on right side of the screen, change Delay = 30, Gain = 2.5 and Tau1 = 40. Click on the Noise icon, click Yes under Cancel all Noise, click OK to cancel noise, and then click Simulate.

3. Click on the Transfer Function icon in the top icon tool bar. Click on Disturbance and Set Gain = 1.0 and Time Constant = 10**.** For disturbance to work correctly, Time Constant must be greater than or equal to 1.0.

4. Now we will add a pulse disturbance. Click on the Disturbances icon located in the top-bar. Click ON to activate Pulse.

   Set Start Time = 300, End Time = 500, Width = 200, and Change = 20.

Figure 2. External Disturbance



5. Now we will add a ramp disturbance. Click again on the Disturbances icon located in the top-bar. Click Ramp. Click ON to activate Ramp.

   Set Start Time = 300, End Time = 1000 and Rate = 0.03 (this is the ramp rate). Set Change = 1000. This means keep ramping in the same direction until the ramp value is 1000. Since 1000 is a high value, the effect is for the ramp signal to increase only in this case. A small value will cause the ramp signal to ramp up and down creating a triangular saw-tooth pattern.

6. Now we will add a sinusoidal disturbance. Click again on the Disturbances icon located in the top-bar. Click Sine. Click ON to activate Sine.

   Set Start Time = 300, End Time = 1000 and Frequency = 0.05 (this is in radians/time unit). Set Amplitude = 3 (this is the amplitude of the sine wave).

7. Click OK to return to the main screen. Near the bottom right side of the screen, set Simulation Time to 800. Click Simulate. Examine the PV, SP and OP trends. See the configured disturbance signal on the top right window. See the PID control action (changes in OP) in the left middle window as the PID tries to reject the disturbance.

   Note that the disturbance signal is the combined effect of the three components - pulse, ramp and sine. This combined signal is passed through the disturbance transfer function with unit gain and time constant = 10.

   The disturbance signal is subtracted from the PV. An upward change in the disturbance signal causes a downward change in the PV.

8. Click Optimize to run the optimizer. The optimizer now will determine the optimum PI parameters corresponding to minimum IAE Error.

9. Determine optimum PID parameters with the current SP change and disturbance. Set P = 3, I = 30, D = 1. Click Simulate. Note Error. Click Optimize to run the optimizer. The optimizer now will determine the optimum PID parameters. Notice again that at minimum error, the calculated optimum D is rather high. Notice also that use of derivative action allows increasing the Proportional gain. In the real plant, using derivative beyond 2-3 is not desirable because of noise in the signals and restrictions on valve movements. The optimizer solution should be used only as an advisory starting point. The user must add appropriate noise band and watch the PID output ROC before finalizing the PID tuning constants to be input into the DCS. Good tuning for this case is: P = 5, I = 50, D = 2.

   This complete simulation can be restored by clicking on the Open Case File icon and then selecting Ex4.pid.

6.0 TUNE A TEMPERATURE CONTROL PID

The following exercise is designed to train the user on how to tune purity control PID loops:

Temperature at Tray#27 is controlled by manipulating the reboiler flow (FC). See Figures 3A and 3B below.

The FC flow range is 0-1400 lb/h. The FC has been well tuned already. The temperature (CV) range is 400 –1000°C. Temperature typically runs at 685°C. Dynamics between FC setpoint and temperature are as follows: dead time = 4 minutes, time constant = 25 minutes. If the FC setpoint is changed by 5.0 lb/h, the total change in temperature at the new steady state is 12°C. Find the optimum PI and PID tuning parameters for TC.

Figure 3A. TC-FC Cascade Control Strategy

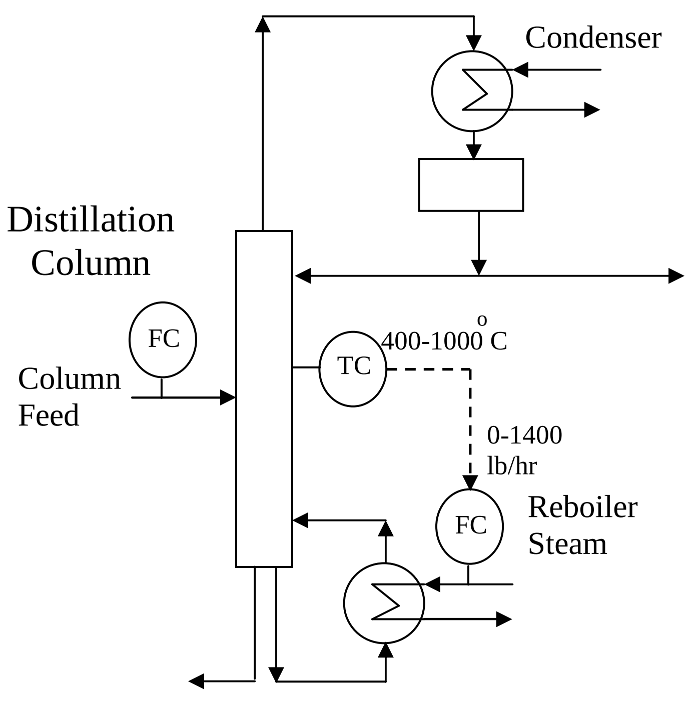
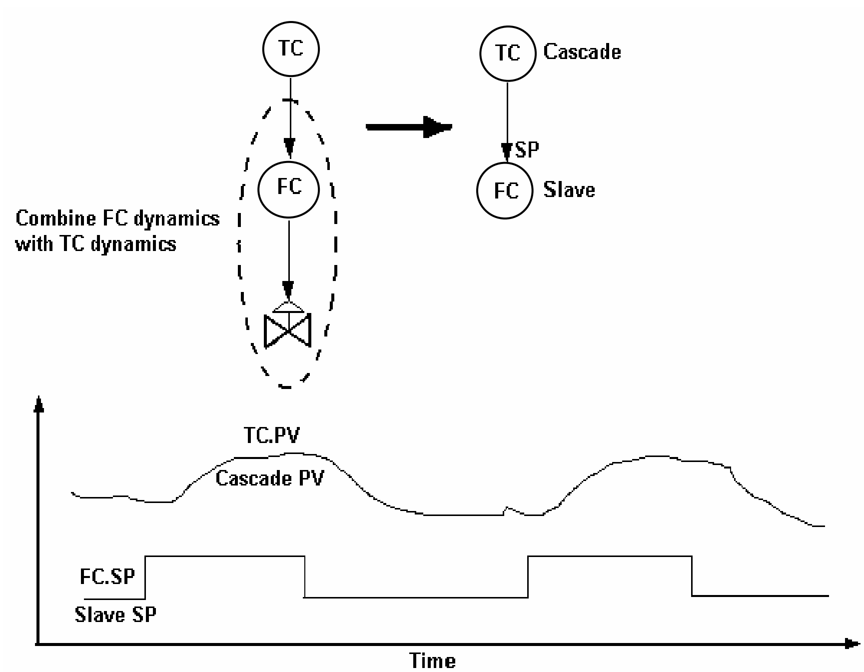


Figure 3B. TC-FC Cascade Loop Dynamic Relationship



Since you have worked through step-by-step procedures and examples above thus far, this tuning problem is for you to work on your own. Good tuning parameters are given below. Compare them against your own best values. Good luck!!

Recommended PID Tuning Parameters:

Optimum (aggressive) PI (IAE): P= 0.58, I = 27.
Recommended DCS tuning with PI only: P=0.2, I=25.
Optimum (aggressive) PID(IAE): P= 0.57, I = 30, D=1.5
Recommended DCS tuning with PID: P=0.2, I=20, D=0.25

This complete simulation can be restored by clicking on the Open Case File icon  and then selecting Ex5.pid.

7.0 TUNE A LEVEL CONTROL PID

This section simulates a level control PID loop. To simulate LC loop, first we need the ***Ramp Rate***. Ramp rate can be determined from a short plant test or may be calculated from the level sump geometry. The plant test procedure is as follows:

Put the LC in manual mode when conditions are relatively stable. Make a step change in the LC valve position by 2-5%. Wait for about 5-10 minutes. ***Ramp Rate*** is calculated as ***change in level / change in valve / time***.

Consider a LC that controls level in a distillation column sump by manipulating flow out of the column sump. The LC output manipulates the valve directly. The CV is the level with range of 0-100%. The MV is the valve, with range also 0-100%. The ramp rate is 0.05 (%level / %valve / minute). See Figure 4 below showing how the ramp rate is calculated. Dead time is 2 minutes. The procedure to determine the tuning constants for the LC is given below.

1. Click on the New Case File icon located near the top left corner of the screen. This clears the screen values and prepares for a new Pitops-PID example.

2. We will first configure a zero order transfer function and examine the open-loop level response. Note that ***zero order*** transfer functions are also called ***integrating*** transfer functions or ***ramp-type*** transfer functions.

   Under Process Transfer Function, on right side of the screen, set Delay to 0 minutes, set Gain = -0.05 (% level/ % valve position /minute). Tau1 and Tau2 must be both zero. Set SP Old and SP New to 55% (initial level).

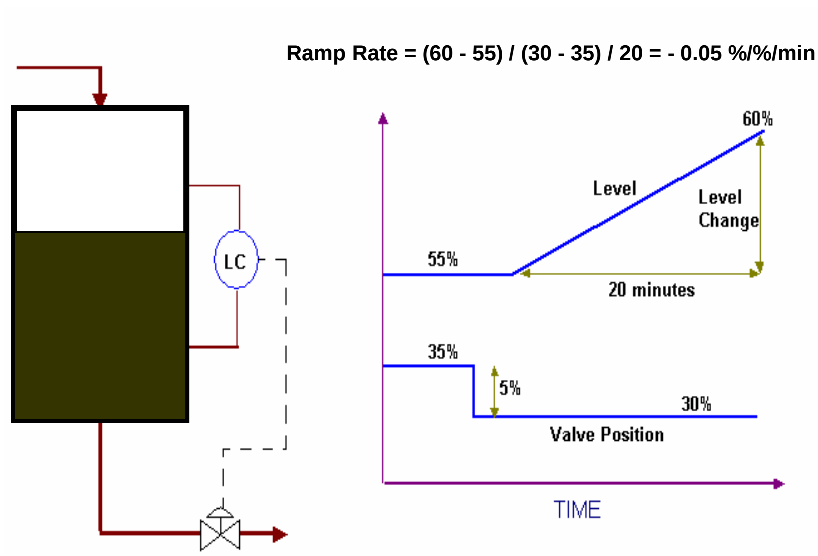
   Under Tuning on right side of the screen, set P = 0 (no PID action- manual mode). At this stage we want to demonstrate the open loop response with no PID action.

Set Valve Bump = -5% (this means that at time = 0, the valve is bumped down by 5%, going from 50% to 45%).

Click Simulate. Note the PV (indicating the tank level) in the top left plot. The level changes from 55% to 80% in 100 minutes due to 5% change in valve position.

The ramp rate = (80 - 55% level)/(45-50% valve position)/(100 min) = -0.05 (%/%/min). This visual check matches with the entered ramp rate of -0.05.

Figure 4. Level Control Loop



3. Now we will make a SP change in automatic mode. Set Valve Bump to zero. Change SP New to 60%. Set Delay to 2 minutes. Under Tuning, set P = 0.5, I = 15 minutes.

Click Simulate. Examine the level PV, SP and OP. The setpoint is changed from 55% to 60%. Control response is rather oscillatory.

Click PID Configuration, Error Criteria. Select option **D**, Reduced Overshoot criterion. This criterion is recommended for ramp type loops.

Click Optimize . Notice that the new SP is achieved very crisply. Notice also that the optimizer has moved the Integral tuning constant to a very high value (no integral action). This is because for SP changes only with no disturbances, mathematically, a zero order transfer function can be well tuned with proportional only control action. In practice though, to control well in the presence of disturbances, some integral control action is necessary.

Notice that the valve position before and after a setpoint change is the same (in this case 50%). This is an important characteristic of zero order transfer functions – for SP change, the initial and final valve positions will be the same.

Try P = 2, I = 50. Click Simulate . Now the control action looks smooth, fast and stable. These are acceptable tuning values.

4. Now add a pulse and a ramp disturbance. Make sure that the tuning is good for both setpoint changes and disturbances. For LCs, handling disturbances is more important than setpoint changes.

   Click Disturbances . Click ON for Pulse. Set Start Time = 300 min, End Time = 500 min, Width = 200 min, Change = 5%.

   Click ON for Ramp. Set Start Time =200 min, End Time = 2000 min, Rate = 0.2 %/min, Change = 1000.

5. Click Transfer Function . Under Disturbance, Set Gain = 1.0 and Time Constant = 1.0.

6. On the main screen, set Simulation Time to 800 minutes. Click Simulate . Examine the PV, SP and OP trends for the PID level controller. Notice how the PID tries to reject the

disturbance. See the Disturbance Signal in the top right window showing the pulse and ramp signals.

Set P = 0 (PID disabled with zero gain). Click Simulate . Examine the level PV and the disturbance signal. Notice that with the PID in manual, the level is directly affected by the disturbance.

Set P = 2, I = 10. Click Simulate . Observe control action. Click Optimize . See the PV, SP and OP trends with the new PI parameters.

Now add derivative action. Set P = 2, I =10, D = 0.5. Click Simulate . Examine control action. Click Optimize . Note the new PID parameters and remember the control response.

Note that the PID parameters provide god control. In a real process, we could use P = 4, I = 16 and D = 2 (or less).

Now we can add some noise to the PV signal to make it look like in a real plant. Click Noise . Increase Slave PV Noise to about 0.2. Click Simulate . Notice that now the valve flickers around in response to the noise.

Try different PID values in the presence of typical noise and determine the best values for downloading into the DCS.

This complete simulation can be restored by clicking on the Open Case File icon and then selecting Ex6.pid.

8.0 TUNE A CASCADE PID

This section simulates a cascade PID loop. The example selected is a distillation column, where the bottoms impurity is controlled by an online analyzer PID (AC, for analysis control) cascaded to a TC which in turn is cascaded to the reboil steam FC. See Figure 5 below. It is assumed that the fast FC is well tuned already. Tuning parameters for the TC and AC are to be determined.

To tune the TC and the AC, first the transfer function between the FC.SP and the temperature must be known. This can be identified by conducting a few pulse tests on the setpoint of the FC. Once the transfer function is known, the TC can be tuned using the procedure provided for tuning a single PID loop. After satisfactory tuning of the TC, the transfer function between the TC setpoint and the online analysis may be identified by conducting pulse tests on the TC setpoint.

Consider an example where the transfer function data are as follows:

Transfer Function	**Delay**	**Gain**	**Time Constant**
FC.SP - TC.PV	7 min	1.3 °C/(t/h)	25 min
TC.SP - AC.PV	20 min	0.4 ppm/°C	110 min

Instrument Ranges:

FC 0-500 t/hr
TC 0-200 °C
AC 0-50 ppm

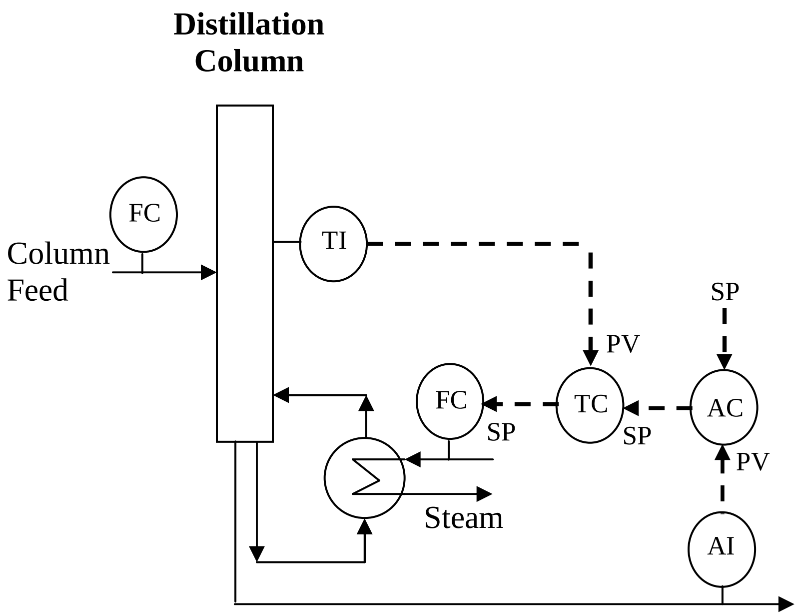
Based on the above information, we can build a cascade simulation and determine the PID tuning constants for the TC and AC. Procedure is as follows:

1. Click on the New Case File icon  located near the top left to start a new Pitops run.

2. On right side of the screen, enter Process Transfer Function Parameters for the TC (7, 1.3, 25). Click PID Configuration , CV Range and MV Range. The MV range for the TC is the FC range, 0-500 t/hr. The CV range for the TC is 0-200 °C. On right side of the screen, set SP Old = 110 °C and SP

New = 115 °C (select reasonable setpoint change typical in the plant based on process knowledge).

Figure 5. Cascade Control Loop



3. Click Simulate  with default P, I values for the TC. Click Optimize  to determine PI values based on IAE criterion. Use derivative action also (non-zero D). Based on process knowledge de-tune the tuning to avoid excessive changes in the reboiler steam flow. Add disturbances and noise if necessary. Finalize the best TC tuning parameters. Now the TC tuning may be considered complete.

4. Now click on Cascade PID button located near top right corner of screen. A new column for the cascade PID appears. Enter Transfer Function Parameters for the cascade PID AC (20, 0.4, 110). Click on PID Configuration , Cascade PID Configuration. Set the CV Range and MV Range as follows:

   The MV range for the AC is the TC range, 0-200 °C. The CV range for the AC is 0-50 ppm.

Under the Cascade column on the far right side of the screen, set SP Old = 20 ppm and SP New = 25 ppm (select reasonable setpoint change typical in the plant based on process knowledge).

5. Now we are ready to watch the simulation and the control action. Set Cascade P = 0.5, I = 25, D = 0. Set Simulation Time to 800 minutes. Click Simulate. Notice that now the cascade PV and SP are displayed on the screen. See the TC and the cascade AC trends.

6. Now we want to optimize the cascade PID tuning parameters. Click Optimize to run the optimizer to determine the cascade PID parameters. Notice the improved control of the cascade loop after completion of the optimizer.

7. Try different PID values for the slave and cascade PIDs. Add noise and external disturbances. Compare different Error and Max ROC values (these are located near bottom right corner of the screen). Try increasing delay and gains in the loops to account for nonlinearities or possible changes in dynamics. Finalize tuning parameters after running various different scenarios, typically seen in the real plant.

   This complete simulation can be restored by clicking on the Open Case File icon and then selecting Ex7.pid.

9.0 FEEDFORWARD TRANSFER FUNCTION

This section illustrates how to use the feedforward transfer function. Click on Control Schematic to see the location of the feedforward transfer function and its input and output signals.

1. Click on the New Case File icon located near the top left to start a new Pitops run.

2. Let's assume that the setpoint at start time of simulation was 50. Set SP Old and SP new to 50. We will leave CV range and MV range at their default values of 0-100 (to change the ranges, click on PID Configuration and set the ranges).

3. Assume the following transfer function parameters for the main process: On the right side of the screen, under the Process Transfer Function box, set Delay = 15 minutes, Gain = 2.0 and Tau1 = 50. Note that the time unit in this example is minutes. All time dependent variables must be specified in minutes.

4. Now define the disturbance transfer function parameters. Click on Transfer Function. The transfer function parameters for the Disturbance transfer function are seen. Set the Disturbance transfer function parameters to be: Delay = 20 minutes, Gain = 1.0 and Time Constant = 100.

5. Calculate the feedforward transfer function parameters based on the values of the main process and disturbance transfer function parameters.

Transfer Function	**Delay**	**Gain**	**Time Constant**
Main	15 min	2.0	50 min
Disturbance	20 min	1.0	100 min

The equations to calculate the feedforward parameters are given below:

Feedforward parameters:

$Delay_{FF} = Delay_{DIST} - Delay_{MAIN} = 20 - 15 = 5$

$Gain_{FF} = Gain_{DIST} / Gain_{MAIN} = 1.0 / 2.0 = 0.5$

$LAG_{FF} = Time\ Constant_{DIST} = 100$

$LEAD_{FF}$ = Time $Constant_{MAIN}$ = 50

where:

FF = Feedforward transfer function
DIST = Disturbance transfer function
MAIN = Main transfer function

6. Now enter the feedforward transfer function parameters as calculated above into Pitops-PID. Click on Transfer Function and click on Feedforward. Set the feedforward transfer function parameters to be: Delay = 5 minutes, Gain = 0.5 and Lag1 = 100 and Lead = 50.

7. Click on Control Schematic to view the block diagram of the control schematic. This schematic shows the location of all transfer functions and signals. Use this diagram as a guide while configuring the simulation.

   Now let's activate a disturbance signal which will be input to both the disturbance and feedforward transfer functions. Click on Disturbances. This brings up the Pulse Signal configuration fields. Click ON to enable the pulse signal. Set Start Time = 100, set End Time = 400. The pulse will start at time = 100 and end at time = 400. Set Width = 300, and Change = 10. The change specifies the amplitude (height) of the pulse. The width specified here (300) appears to be the end time minus start time (400 - 100 = 300). However, if the width is set to 50, then several pulses will be generated. Here we want a single pulse only. Click OK.

8. Now all configuration data required to simulate the feedforward transfer function have been entered. Now, we can run the simulation and examine the plots. Under Tuning, on the right side of the screen, set P to zero (we want to disable the PID so that we can watch the feedforward action only without any PID contribution). Set Simulation Time = 800. Click on Simulate. Notice that the PV trend (upper left plot) is flat, this is because the feedforward signal completely compensates (cancels) the disturbance signal. Notice that the

OP trend (middle left plot) is the feedforward signal contribution.

See the Disturbance signal in the top right window. Even though a disturbance signal is present, there is no apparent effect on the PV (top left plot). This is because the disturbance signal is completely compensated (cancelled) by the feedforward signal.

See the Feedforward signal in the middle right plot. This signal is the output signal from the main process transfer function. The input signal to the main process transfer function is the output signal from the feedforward transfer function (PID output is fixed since PID proportional gain was set to zero earlier, no PID contribution). The net effect of the feedforward and the disturbance is zero showing complete compensation.

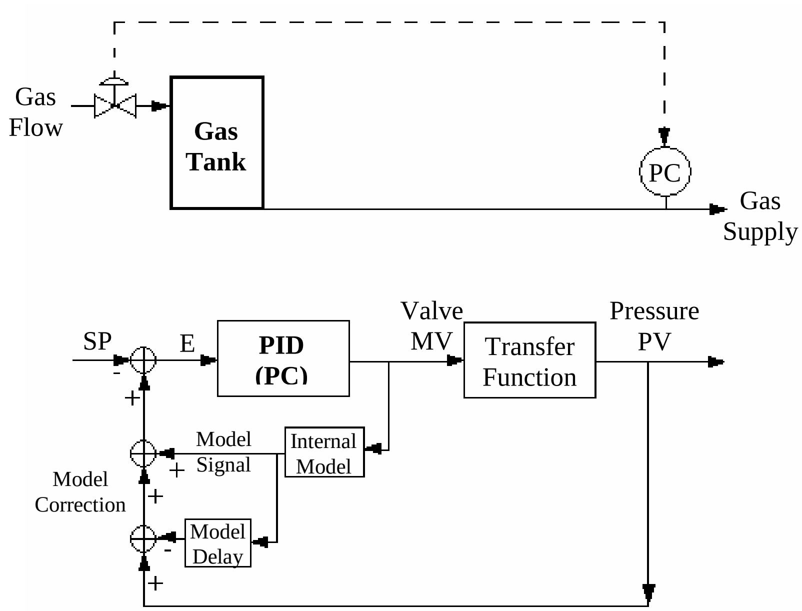
9. Change the transfer function parameter (try different values) and examine the simulation. Note that if the feedforward parameters are calculated based on the above equations, the disturbance is completely compensated and the main PV is a flat line (unchanged). If parameters do not satisfy the equations, then the disturbance does affect the main PV signal.

   This complete simulation can be restored by clicking on the Open Case File icon  and then selecting Ex8.pid.

10.0 MODEL-BASED CONTROLLER

This section illustrates how to use the model-based controller feature. This feature can be used not only for model-based control simulations but also to simulate IMC (internal model control) schemes and for dead-time compensation on loops with long dead times (particularly when the ratio of dead time to time constant is large). Figure 6 illustrates a dead time compensation application using the model-based controller scheme:

Figure 6. Model-based Controller Schematic



Pressure in a gas pipeline is controlled by adjusting the valve position of the supply pipeline as shown above. Distance between the pressure measurement and the gas tank is very long resulting in long time delay. With the PC in manual mode, if gas supply valve position is changed, the pressure downstream changes after a long delay. The delay can be compensated by using the model algorithm shown in the control block diagram above. The following procedure is used to configure the model-based control algorithm in Pitops-PID:

1. Click on the <u>New Case File</u> icon located near the top left to start a new Pitops run.

2. Let's assume that the setpoint at start time of simulation was 45 and we want to simulate a setpoint change to 50. Set <u>SP Old</u> to 45 and <u>SP new</u> to 50. We will leave CV range and MV range at their default values of 0-100 (to change the ranges, click on <u>PID</u> and set the appropriate CV and MV ranges).

3. Assume the following transfer function parameters for the main process: In the right side Process Transfer Function box, set Delay = 40 minutes, Gain = 2.0 and Tau1 = 20. Note that the time unit in this example is minutes. All time dependent variables must be specified in minutes.

4. Set Simulation Time to 500 (minutes). Click Simulate to watch the setpoint change. Control action appears rather oscillatory. Click on Optimize . The oscillations are eliminated due to better tuning determined by Pitops, but the setpoint change looks rather sluggish. Try different PI or PID parameters to improve the setpoint change control action. Because of the long delay (40 minutes) it is not possible to tune the PID aggressively. Long delay causes unfavorable dynamics in the loop. Proportional gain must be reduced in order for the loop to be stable. However, the effect of the dead time can be compensated using the model-based algorithm. The algorithm allows increasing the proportional gain of the PID thus enabling tighter control.

5. Configure the Model transfer function. Click on Transfer Function and then click on Model. Set Delay to 40, Gain to 2.0 and Time Constant to 20 (same parameters as the process transfer function entered in Step 3 above). Click OK.

6. Click on Simulate . Notice that the control action for the setpoint change now looks different as compared to before activating the Model transfer function. Click Control Schematic . The Gain of 2.0 and Time Constant of 20 entered in Step 5 above constitute the box labeled Internal Model. The PID output is fed to the Internal Model transfer function which has a gain of 2.0 and Time Constant of 20 and zero dead time. The Model Delay box is a pure delay function with delay of 40 minutes (specified in the Model transfer function).

   See the bottom right plot showing the model PV after dead time (the delayed model signal).

   The response appears rather sluggish. It is obvious that the tuning could be made more aggressive. Click on Optimize

. Notice that the new tuning is rather aggressive. This is possible because the model-based strategy has completely eliminated the effect of the dead time. The 40 minutes dead time in the process transfer function has been effectively "reduced" to 0. The smaller the dead time, the more aggressive a PID can be tuned. This is why, after adding the dead time compensation, the optimum IAE tuning parameters happen to be so aggressive. In practice, such aggressive tuning cannot be used in the DCS on the real plant. Try different tuning parameters and run the simulation. Examine the PID output rate of change and make sure that it would be acceptable on the real process. Proportional gain = 1.0 and Integral = 20 appear to be good stable PI tuning parameters. In this example, because of the model-based controller, the proportional gain can be increased by about four times compared to without the compensation.

7. Note that the model parameters are set to be exactly the same as the process transfer function (dead time = 40, process gain = 2.0 and time constant = 20). If the model parameters differ from the main process transfer function parameters, then the control loop quality deteriorates. In case of very large differences, the loop could even become unstable.

   To illustrate this, change the Model function parameters. Click on Transfer Function, click on Model transfer function and set the dead time to 20 and the process gain to 1.0. Click OK. Now there is a large difference (error) between the model parameters and the main process transfer function parameters. Click Simulate to run the simulation. Now the control loop is unstable- the PV oscillations keep growing larger and larger. The error between the two transfer functions causes out-of-phase dynamics, thus causing the loop to become unstable. For the model-based strategy to be useful, the transfer function parameters must be accurately known, at least within 15 - 25% accuracy.

   This complete simulation can be restored by clicking on the Open Case File icon and then selecting Ex9.pid.

11.0 VALVE STICTION SIMULATION AND PID OPTIMIZATION

Control valves can be defective and exhibit stiction problems. Valve Stiction is explained in Part C, Section 22 below. Please read that section before proceeding further here.

This section illustrates how to optimize PID tuning in the presence of valve stiction.

1. Click on the Open Case File, select the file Ex10.pid and then click on Open (the file Ex10.pid is provided with the Pitops-PID software). This example shows a FC (Flow Control) PID simulation with a SP change from 70 to 90 kg/hr. A pulse and a ramp disturbance are also active as can be seen in the disturbance plot (top right window).

   On the main screen, under Process Transfer Function, notice that Delay = 5, Gain = 3.0 and Tau 1 = 50. Note that the time unit in this example is **seconds** and the flow unit is kg/hr flow rate. The FC output goes to a control valve (0-100% range).

   Click on Disturbances and then Pulse and Ramp to see the configured disturbances.

2. PID equation is Equation B0; the PID scan time is 1 second. Initial tuning is P = 1.0, I = 15 and D = 0. Click on Optimize. Optimum PI tuning is P = 3.6 and I = 27.

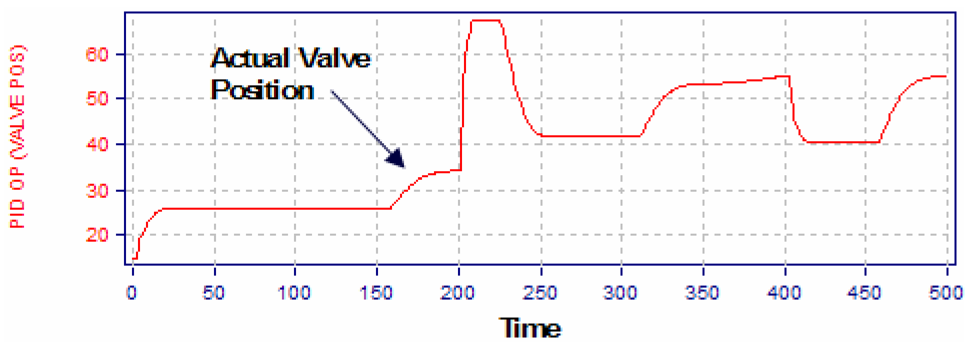
   Note 1: The ***integral*** is in ***seconds*** since time unit is seconds in this example).

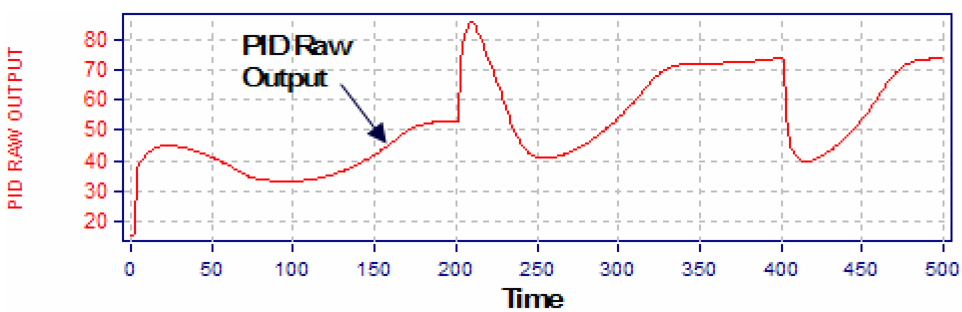
   Note 2: The ***Reduced Overshoot*** is the error criterion for the tuning optimization.

3. Near the bottom right corner of the screen, set Valve Stiction = 20. Click on PID Raw Output button (next to Valve Stiction). Now the bottom left plot window will display PID Raw Output. The button near the bottom right screen corner now will display PID Contribution. You can toggle from PID Raw Output to PID Contribution by clicking (toggling) this button.

4. Click Simulate . Notice the difference between the left middle plot and the left bottom plot. The left bottom plot shows the raw PID Output (OP) going to the valve. The left middle plot shows the actual valve position. See Figure 7 below.

Figure 7. PID Raw Output and Actual Valve Position with Valve Stiction





As explained in Part C, Section 22, valve stiction is caused by sticking, loose play or poor linkage in the control valve hardware. When the PID OP changes, until a certain minimum change has taken place, the valve does not move. Then, if the PID OP changes direction, then again, a certain minimum change in the PID OP must occur before the valve sees a change. This is why you see the “flat lines” in the PID OP (Valve Pos) Plot. During the time the Valve Pos is “flat”, the PID OP is changing to cover the Stiction Value.

5. Click Optimize . Notice that now optimum parameters in the presence of valve stiction are P = 4.4 and I = 38. *In general, a valve with stiction needs more proportional action and less integral action*.

   In a plant, you can estimate valve stiction by conducting “bump” tests on the valve until you see the PV change (this is proof that the valve finally has moved), you can attach temporary valve feedback position indicators and compare thePID OP with the valve position, or you can also use Pitops-TFI (Transfer Function Identification) software. Pitops-TFI is a separate software product that accompanies Pitops-PID.

   This final optimized simulation with valve stiction can be restored by clicking on the Open Case File icon and then selecting Ex11.pid.

PART C: **PITOPS^{TM}-PID REFERENCE MANUAL**

CONTENTS

PART C: **PITOPS^{TM}-PID REFERENCE MANUAL (Contd.)**

CONTENTS

PART C: **PITOPS^{TM}-PID REFERENCE MANUAL (Contd.)**

CONTENTS

PART C: **PITOPS-PID PARAMETER REFERENCE MANUAL**

1.0 MAIN SCREEN PARAMETERS

During a PID simulation, certain parameters (e.g., PID tuning parameters, transfer function parameters and others) are changed frequently. For convenience, these parameters that need frequent changes are displayed on the right side of the screen, from where they can be modified conveniently. Detailed explanation on each of these variables is provided below:

2.0 SINGLE PID / CASCADE PID

Near the top right corner of the screen, two buttons labeled Single PID and Cascade PID are located. Click on Cascade PID to activate a cascade PID. Click on Slave PID to disable the cascade loop and revert back to a single PID. The two buttons are used to toggle between a Single and Cascade loop. Click on the Cascade PID button to see the Cascade PID tuning parameters and transfer function parameters. When Slave PID button is clicked, the cascade column disappears.

When the Cascade PID button is clicked, the cascade PID is placed in automatic mode and the slave PID is placed in cascade (remote) mode. Now the slave PID can accept setpoints from the cascade PID.

When Slave PID is clicked, the mode of the slave PID is changed from cascade (remote) to automatic (local). This disables the cascade PID and causes the output (OP) of the cascade PID to track the setpoint of the slave PID.

Click on Control Schematic  to see the cascade and slave PID blocks.

3.0 TUNING (P, I, D PARAMETERS)

A box labeled Tuning is located near the top right corner of the screen, just below the Single PID, Cascade PID buttons.

This box displays PID (Proportional, Integral and Derivative tuning parameters). The user may change these parameters and then click on Simulate to run the simulation with the new parameters and refresh the screen plots.

3.1 P (PROPORTIONAL GAIN)

This is the Proportional Gain term in the PID equation. See the "P" parameter in the PID equation described later below.

3.2 I (INTEGRAL CONSTANT)

This is the Integral Tuning Constant in the PID equation. This is also called Reset. See the "I" parameter in the PID equation explained in a later section. Units are (time units)/repeat. If time unit is selected to be seconds, the integral will be seconds/repeat, if time unit is minutes, the integral will be minutes/repeat.

3.3 D (DERIVATIVE CONSTANT)

This is the Derivative Tuning Constant in the PID equation. See the "D" parameter in the PID equation explained in a later section.

If the user clicks on Optimize, then Pitops determines new PID parameters based on the error criterion. Click on PID Configuration, click on Error Criteria to see error criteria options. The IAE (Integral Absolute Error) is the most common criterion. This criterion minimizes the difference between the SP and PV while searching for improved PID parameters. Pitops runs a minimization algorithm that minimizes the error (IAE) while searching for new PID parameters.

If Derivative is set to zero, then the Optimize function will not search for a new Derivative constant, but set it fixed at zero.

The PID parameters inside the Tuning box correspond to the PID equation type specified by clicking on PID Configuration icon in the top bar and then clicking on PID Equation. Note that some equations use the Proportional Band (defined as 100/Proportional Gain) instead of Proportional Gain as the numerator.

4.0 MANUAL AND AUTOMATIC PID MODE

To put the PID in manual mode, set the proportional gain P to zero. This disables the PID controller. An explicit Auto/Manual flag is not provided.

5.0 PROCESS TRANSFER FUNCTION

The Process Transfer Function box is located below the Tuning box. It displays the transfer function parameters for the main process. These are the Slave Transfer Function and the Cascade Transfer Function parameters as displayed in the control schematic (click on Control Schematic to see the transfer function boxes).

The user must specify the transfer function parameters (Delay, Gain, Tau1 and Tau2).

The Gain is the process gain calculated as $\Delta CV/\Delta MV$ in open-loop mode (with the PID controller in manual mode).

The transfer function parameters can be determined by three methods:

- By conducting some step tests in the real plant and then examining the DCS trends.
- By analyzing the plant data trends with the accompanying Pitops-TFI (Transfer Function Identification) module.
- Inferring process gain based on control room operator experience and information.

6.0 SP OLD AND SP NEW

SP Old means setpoint old and SP New means setpoint new. The old setpoint specifies the PID's setpoint at the start time of the simulation (at time = 0). The new setpoint is used to simulate a setpoint change. The setpoint old and new values entered must lie within the CV (controlled variable) limits defined in a later section. If the value entered falls outside the CV limits, it will be clamped to the nearest (high/low) CV limit.

If setpoint old and new are the same, the Pitops simulation may show straight lines with no dynamics unless a disturbance or noise signal is active. Generally, PID tuning is based on two cases: tuning for setpoint changes and tuning for disturbances. Pitops simulates both setpoint change and/or disturbance to determine optimum PID tuning parameters.

SP Old should be set to typical setpoint for that PID in the real plant. This will make the simulation look very realistic. SP New will be based on how much the setpoint is typically changed for that PID. By setting plant-like values, the Pitops simulation can be made to look very much like the real plant DCS PID trends.

7.0 VALVE BUMP

Valve Bump is used to observe the open-loop shapes of transfer functions. The process delay time, the 63% rise time in a first order transfer function can be visually seen in the simulation plots. This field represents the change in the PID controller's output at the start time of the simulation (at time = 0). A value of 1 for this field means that a 1% change in the PID's output (control valve) was made at the start time of the simulation.

Set Valve Bump to zero while simulating PID action in closed loop mode. (Closed loop mode is when the PID output is changing based on the PID algorithm, in response to a setpoint change or disturbance). This mode is also called the automatic (Auto) mode. Open loop mode is when the PID output is not changing according to the PID algorithm, but is either fixed or is changed by the operator. This mode is also called manual (Man) mode.

See Section 2 in Part B of this manual for a procedure on how to configure an open-loop transfer function response using the Valve Bump.

8.0 MAX ROC AND CURRENT ROC (RATE OF CHANGE)

The ROC (Rate of Change) is the largest change in the PID's output between any two consecutive PID executions calculated during the simulation window. The Pitops optimizer determines tuning parameters based on the minimum error criterion (IAE, ISE, ITAE, Reduced Overshoot), explained in a later section below. The control action based on these criteria could be rather aggressive, causing excessive movement in the PID output (valve position). With the Max ROC feature, we can impose a limit on how fast the PID's output can change.

The Max ROC and Current ROC fields are displayed near the bottom right corner of the screen. The Current ROC is the largest change in the PID's output between any two consecutive PID executions with the current set of PID tuning parameters. The Max ROC is the maximum tolerable ROC. The user can examine the OP trend and the Current ROC displayed and then select a Max ROC based on his knowledge about the control loop needs.

For first order transfer functions with high ratio of Time Constant to Delay, the PID control action may be both tight and stable, but the output may be changing too fast for the real plant. In such cases, it is important to reduce the proportional contribution of the PID and reduce the aggressive action. The Max ROC feature helps in such cases.

Refer to Section 3 in Part B of this manual for a complete procedure on how to use the Max ROC function.

The following steps summarize how to use the Max ROC function:

Step #1:
Configure a PID loop (specify transfer function parameters, instrument ranges- CV, MV ranges).

Step #2:
Enter some reasonable starting PID tuning parameters (PID action should not be unstable).

Step #3:
Simulate a setpoint change; optimize tuning using the Optimize feature. Observe the trend of PID OP (PID Output) displayed in the bottom trend window. Note the Current ROC value displayed near the bottom right corner of the screen.

Step #4:
If the rate of change in the PID output appears to be aggressive, then set Max ROC equal to half the Current ROC value noted in step #3.

Step #5:
Run optimizer (click Optimize). The optimizer will determine the optimum PID tuning constants based on the imposed Max ROC. The PID tuning constants will not be as aggressive as before. Note that the Current ROC now will match the Max ROC.

Note: If the optimizer fails to improve the PID tuning constants, try different, preferably less aggressive P, I or P, I, D tuning parameters and rerun the optimizer. Make the PID tuning stable and non-oscillatory before clicking on Optimize .

9.0 ERROR

Error is displayed near the bottom right corner of the screen. Pitops allows selecting four types of error criteria. Click on PID Configuration , Error Criteria to see the four choices (IAE, ISE, ITAE, Reduced Overshoot). If IAE is selected, then the Error value displayed is the error calculated using the IAE equation (see explanation on IAE and other error criteria in a later section below).

When Optimize is clicked, Pitops reads the PID parameters specified by the user and searches for improved parameters while trying to minimize the Error value. The final PID parameters from the optimizer may be different depending on the selected criterion.

10.0 SIMULATION TIME

Simulation Time is the total time to run the simulation. Maximum simulation time is 10,000 time units (seconds, minutes, hours or other units). The longer the simulation, the longer it will take for Pitops-PID to run. Simulation Time should be longer for slow processes in order to observe the complete dynamics after a simulated setpoint change or disturbance. Typically, set Simulation Time no less than the Delay plus three times the first order Time Constant. Good values of Simulation Time are 100 – 1000.

11.0 EXIT

To exit Pitops-PID, either click on File and Exit, or click on the Windows “X” icon on the top right corner of the screen. Exit is used to exit the Pitops-PID module. You will see a "Yes" or "No" confirmation prompt. If you select No then the exit is cancelled. If you select Yes then the Pitops-PID module exits. Before exiting, all configuration data at time of exit are saved to a configuration file. If you restart Pitops-PID again later, then this file is opened and all data are restored. This allows the user to conveniently continue the last run after exiting the program.

12.0 TRANSFER FUNCTION EQUATION

On the right side of the screen, the Process Transfer Function equation format is displayed, as shown below:

$$G(s) = \frac{\text{Gain}\ e^{-\theta s}}{\tau_2 s^2 + \tau_1 s + 1}$$

Note that the Tau1 and Tau2 (time constants) displayed under the Process Transfer Function box are the denominator τ_1 and τ_2 shown in the above equation. Note that Pitops-PID uses the polynomial form and not the factorial form for the denominator of the above transfer function equation.

13.0 FILE

The File pull-down menu button is located near the top left corner of the screen. It is used to read and save Pitops-PID simulation case files. After a simulation is configured in Pitops, the whole data set can be saved conveniently to a file for later use. The File pull-down menu button provides the following file operations for opening and saving files. It also is used to generate vector files (explained below) and also to print plots and reports:

13.1 NEW CASE FILE

New Case File erases all current data and initializes with default values (mostly zeros). This is a convenient way to start a new run. Data from the previous run is erased and the simulator is ready for a new case. The icon  is also denotes New Case File.

13.2 OPEN CASE FILE

Open Case File function allows reading a previously saved case file (case files are described in a later section below). The icon  also denotes Open Case Files. The case file contains all configured data: PID tuning parameters, transfer function parameters, disturbance, noise, and all other configuration data. This is a convenient means of restarting the case at a later time.

At the time of exiting Pitops, the complete configuration is automatically saved to a special start-up case file. This file is automatically read after Pitops is rerun at a later time. Thus, exiting a session and restarting from the previous case are convenient.

To open an existing case file, click on Open Case File or the icon  and then select the appropriate file from the window and click on Open.

It is recommended that case files in the PID module should be given filename extension ".pid". This is not a requirement, but a good naming convention to follow. The Open Case File function expects case filename extensions to be PID. If different extensions are used, then *File Type* should be changed to *All Types* (*.*) to view the case files.

13.3 SAVE, SAVE AS

The Save function saves all current information to the existing file name on the hard disk for future use. All PID parameters, transfer function parameters, noise data, disturbance parameters and all other configuration data are saved to the current simulation case file. The icon  is also denotes Save.

This case file can be restored at a later time by using the Open Case File option described above.

To save the current configuration data to a file with a different name, click on Save As and then specify the new case file name as desired.

13.4 GENERATE VECTOR FILES

Pitops screen plots can be exported to Excel spreadsheet files. To generate data files, click on Generate Vector Files. Specify the filename to write the data to in Excel format and click on Save. An Excel file with the specified name will be created.

The file format is as follows: The first column contains time sample. The data begins in the second column. Also, the top three rows do not contain plant data, they display some tag labels or are blank. Data begins from fourth row onwards.

The file structure (column and row number where data are saved) can be changed as necessary. To change the file structure to suit your file format, go to the Pitops-PID directory and locate the file called ***Default.ini***. Using the Notepad editor (or any text editor), modify the Default.ini file and modify the Row and Col as needed and save the file.

All the rows in the file consist of data spaced apart at unit sample time.

The second column is labeled "CV" and contains the PV (Process Variable) data, as seen on the top left plot red-colored trend.

The third column is labeled "MV1" and contains the OP (PID Raw Output) data.

The fourth column is labeled "MV2" and contains the Disturbance Signal data.

To change the number of rows (number of data points in the above files), set the Simulation Time (on the main screen) equal to the number of data points desired. And then click on Simulate to rerun the simulation. Note that a unit simulation time corresponds to one data point in each file. Thus, to generate a file with 240 data points, set the simulation time to 240.

13.5 PRINT (PLOTS OR REPORTS)

Click on File / Print or click on the icon  to print the plots or a report on the complete configuration. Click Plots to print the plot or Report to print the report.

14.0 CONTROL SCHEMATIC

Click on Help / Control Schematic or the icon to see the complete control schematic block diagram of functions supported by Pitops. This diagram is useful as a convenient reference while configuring a Pitops simulation.

15.0 PID CONFIGURATION

Most of the frequently changed PID attributes (tuning parameters, setpoints) are changed directly from the data box on the right hand side of the screen, as explained above. All other PID configuration parameters are specified by clicking on the Inputs and PID Configuration button located in the top bar located near the top left corner of the screen (or you can also click on the icon).

If the Cascade PID button is toggled near the top right side of the screen to activate a cascade loop, then if the PID Configuration button is clicked, two choices can be seen: Slave PID Configuration and Cascade PID Configuration.

If cascade is not active, then only the Single PID option is available to access the single PID's parameters.

The slave and cascade configuration options are identical and are explained below:

15.1 INSTRUMENT RANGES

Click on Inputs / PID Configuration or the icon . The first option seen is the Instrument Ranges.

In case of a single flow controller (FC), the CV range will be the flowmeter range and the MV range will be the valve range, which is normally 0 - 100%.

Consider a TC - FC (temperature - flow) cascade loop where reactor temperature is controlled by manipulating steam flow to a heater. The CV range for the cascade TC will be the range of the temperature transmitter and the MV range will be the range of the FC flowmeter. For the slave FC, the CV range will be the flowmeter range and the MV range will be 0-100% (valve range).

The default ranges are 0 - 100%. By specifying the correct CV and MV ranges, the PID tuning parameters in Pitops will be fully compatible with those in the DCS or PLC. If the 0-100% default range is used, then the PID tuning in the DCS will be different.

15.2 PID EQUATION

Click on Inputs / PID Configuration or the icon  and then click on PID Equation to select the type of PID equation. Twenty-four PID equations are provided:

A0. P (dE + E dt / I + D d(dE / dt))
B0. P (dE + E dt / I + D d(dPV / dt))
C0. P (dPV + E dt / I + D d(dPV / dt))
A1. 100/ PB (dE + E dt / I + D d(dE / dt))
B1. 100 / PB (dE + E dt / I + D d(dPV / dt))
C1. 100 / PB (dPV + E dt / I + D d(dPV / dt))
A2. P (dE + E dt * I + D d(dE / dt))
B2. P (dE + E dt * I + D d(dPV / dt))
C2. P (dPV + E dt * I + D d(dPV / dt))
A3. 100/ PB (dE + E dt * I + D d(dE / dt))
B3. 100 / PB (dE + E dt * I + D d(dPV / dt))
C3. 100 / PB (dPV + E dt * I + D d(dPV / dt))
A4. P * dE + E dt / I + D d(dE / dt)
B4. P * dE + E dt / I + D d(dPV / dt)
C4. P * dPV + E dt / I + D d(dPV / dt)
A5. P * dE + E dt * I + D d(dE / dt)
B5. P * dE + E dt * I + D d(dPV / dt)
C5. P * dPV + E dt * I + D d(dPV / dt)
A6. 100/PB * dE + E dt / I + D d(dE / dt)
B6. 100/PB * dE + E dt / I + D d(dPV / dt)
C6. 100/PB * dPV + E dt / I + D d(dPV / dt)
A7. 100/PB * dE + E dt * I + D d(dE / dt)
B7. 100/PB * dE + E dt * I + D d(dPV / dt)
C7. 100/PB * dPV + E dt * I + D d(dPV / dt)

The parameters P, I and D are the Proportional gain, Integral constant and the Derivative constant in the PID equation. These are typical PID algorithms offered in most DCS and PLC systems. Additional new algorithms may be easily added if desired, contact Pitops customer service for details - info@picontrolsolutions.com or www.picontrolsolutions.com.

The term **E** is the error, defined as (**PV** - **SP**), the current deviation of the process variable from the setpoint.

The term **dt** is the execution time (scan time) of the PID algorithm. The term **dE** represents the delta **E**. This is the change in the error (**E**) between two consecutive PID executions.

Equations of type **B0** and **B1** are most common. In these equations, the derivative acts on the deviation in the process variable, not the error.

Equations of type **A** can cause a large derivative "kick" during a setpoint change because their derivative terms act on the error instead of the process variable. This kick may be unnecessary and even undesirable.

Proportional contribution in equations of type **C** is based on **delta PV**, and not **delta Error**. These equations are not recommended on slave loops where the effect of proportional kick is valuable and important for good setpoint control. Equations of type **C** are common on LCs (level controllers) where setpoint changes are rare.

15.3 PV FILTER CONSTANT

Click on Inputs / PID Configuration or the icon  and click on PV Filter Constant to specify the filter constant. This simulates filtering action provided in most DCS and PLC systems. A noisy signal could cause excessive jittery movement in the PID's output. This could result in undesirable control action including excessive wear on the valve. By adding some filter action, a noisy PV can be dampened.

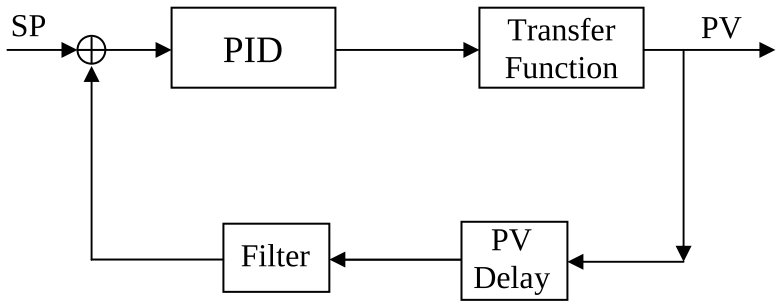
A value of zero means there is no filtering. To add filtering, type in a positive value equal to or more than 0.05.

The user must select the unit of time, and then all time dependent parameters must be specified in the selected time unit. For fast loops like FCs, the time unit will be typically seconds. The PV Filter Constant then must be specified in seconds, with values like 3 or 5 seconds. For very noisy signals, a filter constant of 10-12 seconds may be appropriate. More than 15 seconds is rather high degree of filtering and must be evaluated carefully. Excessive filtering adds delay in the loop and will harm control quality.

The location of the filter time constant block is shown in Figure 8 below.

PV filter constants can be specified on both slave and cascade loops.

Figure 8. Filter and PV Delay Function Blocks



15.4 PID EXECUTION PERIOD

Click on Inputs / PID Configuration, or the icon and then click on PID Execution Period. This parameter defines the execution frequency (scan time) of the PID controller. The default value is 1 time unit. This means that the PID controller executes every one time unit (second, minute,...) in the control system (DCS or PLC). Select any of the other specified options as necessary for slower or faster scan rates. For most applications, the default value 1 time unit will be adequate.

Note that numerous advanced control loops (cascade PIDs, feedforwards) encountered in the industrial or chemical plants are typically loops with slow dynamics. In most cases, the dead times are about 1 - 20 minutes and time constants are 10 - 300 minutes. Because the dynamics are slow, the execution period of 1 minute is appropriate.
If the process dynamics are fast, where the dead times and time constants are of the order of seconds or even milliseconds, then the time unit can be considered to be seconds or milliseconds. The X axis in the simulation plots then will represent seconds or milliseconds or any other (user selected) time basis. The PID integral tuning parameter and all other parameters (time delays, time constants, simulation time,...) will then correspond to the seconds, milliseconds or other selected time unit.

15.5 PV SAMPLE DELAY

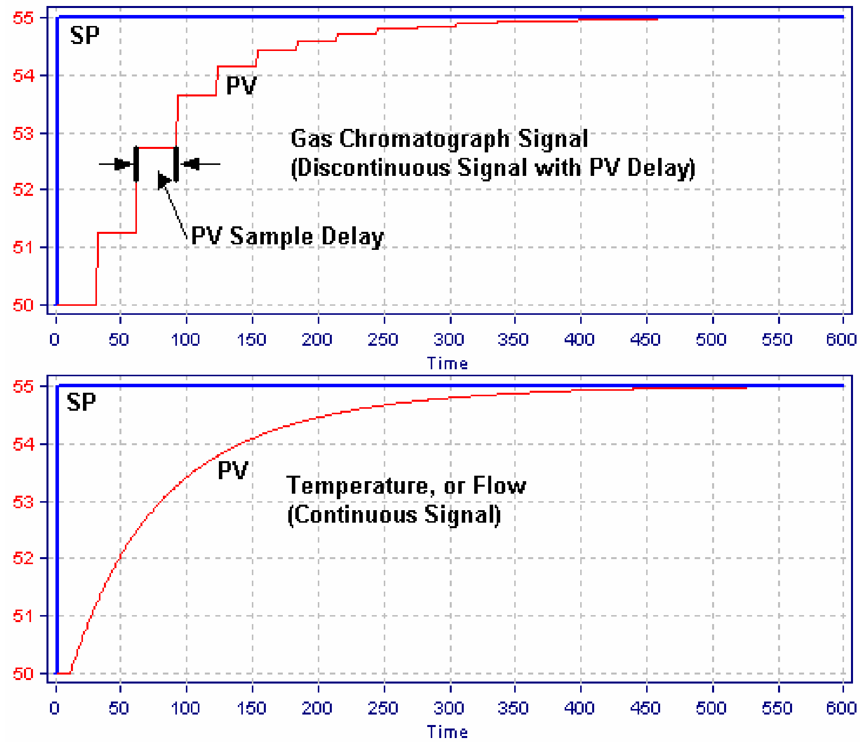
Click on Inputs / PID Configuration or the icon  and then click on PV Sample Delay to specify this parameter. The PV sample delay time is used to simulate a discontinuous signal, as in the case of gas chromatographs (GC).

A GC is an online analyzer that measures the content of specific components in fluids. An injection system injects a metered volume of fluid into the GC. The GC analyzes the fluid and outputs the fluid component analysis. The process of injection, fluid analysis and signal output typically takes 3-30 minutes.

A typical GC signal is shown in Figure 9.

The PV Sample Delay time is used to simulate the delay in a process signal. The default value is zero, as in the case of most instruments, e.g., temperature, flow, pressure and level signals. In the case of the GCs or similar types of discontinuous signals with delays, set the PV Sample Delay equal to the delay time based on the GC or instrument specifications.

Figure 9. PV Sample Delay in a Gas Chromatograph Signal



15.6 GAP ACTION

The purpose of gap action is to dampen unnecessary PID control action when the deviation of the process variable from its setpoint is small. The amount of the dampening is specified by the gap gain and the amount of deviation by the gap low and gap high.

Click on Inputs / PID Configuration or click on the icon  and then click on Gap Action to specify the gap parameters Gap Gain, Gap Low and Gap High. The default value for Gap Gain is 1.0 and for Gap Low and Gap High are 0.0.

Consider the following illustration of a Gap PID: Assume the following PID parameters: Setpoint = 50, Gap Gain = 0.5, Gap

Low = 3 and Gap High = 2. Now, if the process variable (PV) is between 47 and 52, then the Proportional Gain will be multiplied by the Gap Gain (0.5 in this example). The limits 47 and 51 are equal to the Setpoint minus the Gap Low (50 – 3 = 47) and Setpoint plus Gap High (50 + 2 = 52). The PID gain is thus dampened by a factor of 0.5 (gap gain) when the process variable is within the gap bands. Values of 0.0 (no PID action within the gap band) to 0.5 are typical values for Gap Gain.

15.7 ERROR CRITERIA

Click on Inputs / PID Configuration or click on the icon  and then click on Error Criteria to select from the following four options:

A. Integral Absolute Error (IAE)
B. Integral Squared Error (ISE)
C. Integral Time Absolute Error (ITAE)
D. Reduced Overshoot

The error criteria help to quantify the control quality of a PID. The Error, which is calculated as Process Variable (PV) minus Setpoint (SP) calculated over a specified time period, gives an indication of the quality of the PIDs tuning constants.

If a setpoint is changed and the time taken to reach the new setpoint is too long, the PID parameters are sluggish. In this case, the deviation between the PV and the SP will be large, and subsequently the Error will be large also.

The Integral Absolute Error (IAE) is the absolute Error (PV - SP) summed (integrated) over the simulation period. See Figures 10 a, b, c below. The integrated absolute error (Error value) is the shaded portion bounded by the SP and PV signals as shown in the figures.

With slow tuning (Figure 10 a), the IAE value will be larger than with optimal tuning (corresponding to minimum IAE, Figure 10 b). As tuning is made more aggressive, the PV becomes oscillatory and results in increased shaded area (larger error, Figure 10 c).

Figure 10 a. IAE Criterion with Slow Tuning

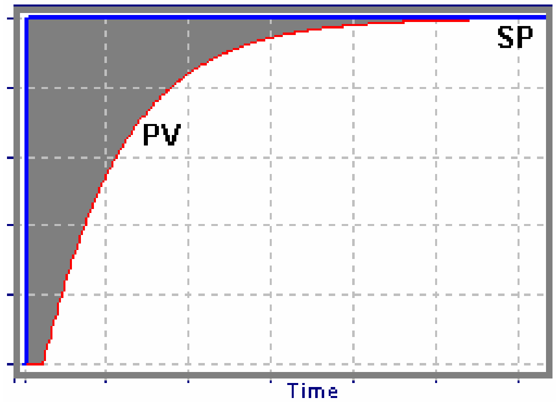


Figure 10 b. IAE Criterion with Optimal Tuning

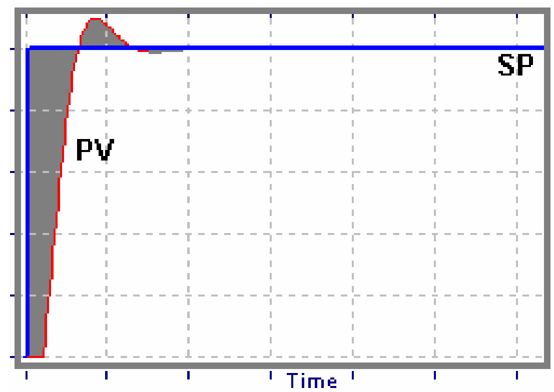
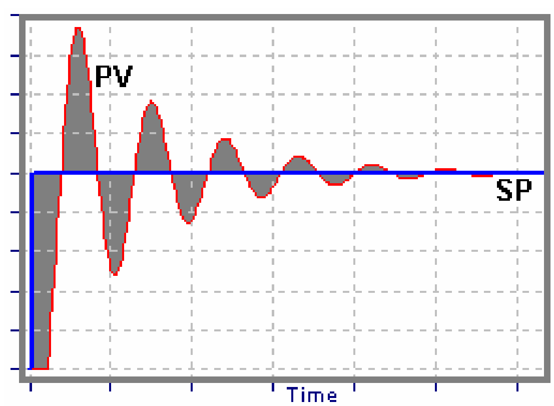


Figure 10 c. IAE Criterion with Aggressive Tuning

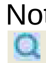


The Integral Squared Error (ISE) is the square of the error summed continuously over the simulation period.

The Integral Time Absolute Error (ITAE) is the absolute error multiplied by the elapsed simulation time summed continuously over the simulation period. The ITAE criterion penalizes the error more severely with elapsed time.

The fourth option Reduced Overshoot will generate PID tuning constants with little or no overshoot.

When the Optimize  button is clicked, the Pitops optimizer determines new PID parameters corresponding to minimum error based on the selected criterion. The Error value is displayed near the bottom right corner of the screen. Notice that the value of Error changes if the PID tuning is changed or some other parameters are changed.

For most cases, the IAE or the Reduced Overshoot criteria are recommended while running the Pitops optimizer (by clicking on Optimize  button). Note that PI or PID tuning generated by clicking on Optimize  button will be generally on the aggressive side. The IAE tuning is close to the ultimate upper limit on the aggressiveness of the PI or PID controller. For the real plant, detune the IAE optimized tuning or use the Reduced Overshoot criteria.

15.8 INITIAL PID OUTPUT

Initial PID Output is the PID output at the start time of the simulation (at time = 0). Initial PID output is the same as the initial control valve position. The default value is 50%. This means the valve is 50% open at the start time of the simulation.

Any value for the initial PID output (30%, 75%, ...) will not affect the PID tuning parameters. However, if the initial PID output is specified as 90%, this means that the control valve is open 90% at start time of the simulation. In this case, the valve has only 10% room to move upwards, and a "wind-up" situation can be simulated as illustrated in the example below:

Consider a TC-FC cascade example. A TC controls the reactor temperature. The TC cascades to a steam FC on a heater. Temperature setpoint is 150°C, the steam flow is 4800 kg/hour and the steam flow control valve is 97% open (Initial PID Output = 97%). Now, if the temperature must be raised from 150 to 160°C, the TC will increase the steam FC's setpoint. Because the valve is already 97% open, while increasing the steam flow, the valve will be wide open at 100%. The final desired temperature of 160°C may not be achieved because of the control valve limitation. Such situations can arise because of loss of supply steam pressure, abnormally high loading or inadequate sizing of the control valve.

15.9 NONLINEAR GAINS

Click on Inputs / PID Configuration or the icon  and then Nonlinear Gains to specify up to ten nonlinear gains. The default value of each of the ten gains is 1.0. The transfer function gain is multiplied with these gains depending on the PID's output (valve position). These gains help to simulate process nonlinearity commonly encountered in many practical examples.

An example is the valve position to flow relationship. After a valve is more than about 70% open, any incremental increase in flow per unit change in valve position can drop of steeply.

E.g., delta flow/delta valve position is 1.0 from 0-70% valve position. From 70-80%, delta flow/delta valve position is 0.5. From 80-90%, delta flow/delta valve position is 0.25. And finally, from 90-100%, delta flow/delta valve position is 0.1. Note that the process gain in the 90-100% valve position range is one-tenth of the process gain when the valve is under 70% open. In this example, the first seven nonlinear gains from 0-70% will be left at their default value of 1.0. The nonlinear gain for the 70-80% range is set to 0.5; nonlinear gain for the 80-90% range is set to 0.25, and nonlinear gain for the 90-100% range is set to 0.1.

This specification makes the transfer function nonlinear and allows simulation of a nonlinear transfer function amidst setpoint changes and disturbances.

15.10 TRANSFORMS

Click on Inputs / PID Configuration or the icon  and then click on Transform to see four options:

- No Transform
- Natural Logarithm
- Squared
- Square Root

The transform applies to both the PV and SP before the PID error is calculated. With no transform, the raw PV and raw SP are used to calculate error.

If Natural Logarithm is selected, then PV = ln(PV) and SP = ln(SP) and PID Error = ln(PV) - ln(SP).

If Squared is selected, then PV = PV^2 and SP = SP^2 and PID Error = PV^2 - SP^2

If Square Root is selected, then PV = sqrt(PV) and SP = sqrt(SP) and PID Error = sqrt(PV) - sqrt(SP)

These transforms are very useful in linearizing nonlinear processes.

16.0 TRANSFER FUNCTIONS

Click on Inputs / Transfer Function or the icon  to specify transfer function parameters for the Disturbance, Feedforward and Model transfer functions.

Click on Control Schematic  to see the Disturbance and Feedforward transfer functions. The Model transfer function consists of the Internal Model and Model Delay blocks as shown in the control schematic.

16.1 DISTURBANCE TRANSFER FUNCTION

Click on Inputs / Transfer Function or the icon and then click on Disturbance. This brings the Disturbance transfer function. Specify the Delay, Gain and Time Constant for the disturbance transfer function.

16.2 FEEDFORWARD TRANSFER FUNCTION

Click on Inputs / Transfer Function or the icon and then click on Feedforward. This brings the Feedforward transfer function. Specify the Delay, Gain, Lead, Lag1 and Lag2 for the feedforward transfer function.

16.3 MODEL TRANSFER FUNCTION

Click on Inputs / Transfer Function or the icon and then click on Model. This brings the Model transfer function. Specify the Delay, Gain, and Time Constant.

The Delay corresponds to the Model Delay as shown in the Control Schematic. The Internal Model also shown on the Control Schematic consists of the Gain and the Time Constant. For the Model to be perfect, the Gain and Tau1 of the main process transfer function (slave transfer function) should match the Gain and Time Constant for the Model. Also, the Delay in the slave transfer function should match the delay in the Model. If the match is poor, control action can become unstable.

17.0 NOISE

All instrument signals in industrial processes are noisy. Flows, temperatures, pressures, levels and online analyzers all exhibit noise of various magnitudes. The noise is also called **random noise** or **white noise**.

The noise is a high frequency disturbance that can affect control quality. Pitops can make the simulation look realistic by adding noise to the smooth PV signal.

With Pitops, noise can be injected at three locations as shown on the control schematic (click on Control Schematic to see the noise injection points). These are explained below:

17.1 SLAVE PV NOISE

Click on Inputs / Noise Configuration or the icon and then click on up arrow inside the Slave PV Noise box. Up arrow increases noise, down arrow reduces noise. The number displayed inside the box is fractional percent noise band. Typical noise bands encountered in the plant correspond to numbers of 0.1 - 2.0.

You can estimate the noise band by examining DCS trends and then add noise in Pitops to match the real process. Adjust the noise level by clicking on the up and down arrows under the Slave PV Noise box until the simulation matches closely with the DCS. You have to click on Simulate to see the noise effect.

17.2 CASCADE PV NOISE

Click on Inputs / Noise Configuration or the icon and click on the up and down arrows inside the Cascade PV Noise box to add or reduce noise to the cascade PV signal. Functionality is similar to the slave PV noise.

17.3 DISTURBANCE PV NOISE

Click on Inputs / Noise Configuration or the icon and click on the up and down arrows inside the Disturbance PV Noise box to add or reduce noise to the disturbance PV signal.

17.4 CANCEL ALL NOISE

To cancel all noise, click on Inputs / Noise Configuration or the icon and click Yes inside the box labeled Cancel All Noise. Click OK and then click Simulate to rerun the simulation and refresh the screen. This action cancels all noise.

18.0 DISTURBANCES

Click on Inputs / Disturbance Signals Configuration or the icon to generate a pulse, ramp or sinusoidal disturbance. The disturbance signals help in simulating a process disturbance. This helps to check how the PID handles disturbances. The PID should be tuned such that it handles both setpoint changes and disturbances well.

See Section 5 in Part B of this manual for a detailed example on how to configure a disturbance.

Click on Control Schematic or the icon to see the location of the disturbance signals (pulse, ramp, sine signals). Notice that these signals are input to the disturbance and feedforward transfer functions.

Note that after configuring the signals, the Disturbance transfer function gain must be changed to nonzero in order for the disturbance signals to pass through the function and have some effect on the PV. Typically, the Disturbance transfer function Gain is set to 1.0 and Time Constant to 1.0. This results in the disturbance signal passing through the disturbance transfer function unchanged. If Time Constant for the disturbance transfer function is set to zero, Pitops interprets this as a zero order transfer function (ramp) and will produce wrong results. The time constant should be either 1.0 for no lag effect, or something larger like 5 or 10. This will cause the disturbance signal to be lagged (time constant effect).

The following sections explain how to configure and activate the various disturbance signals.

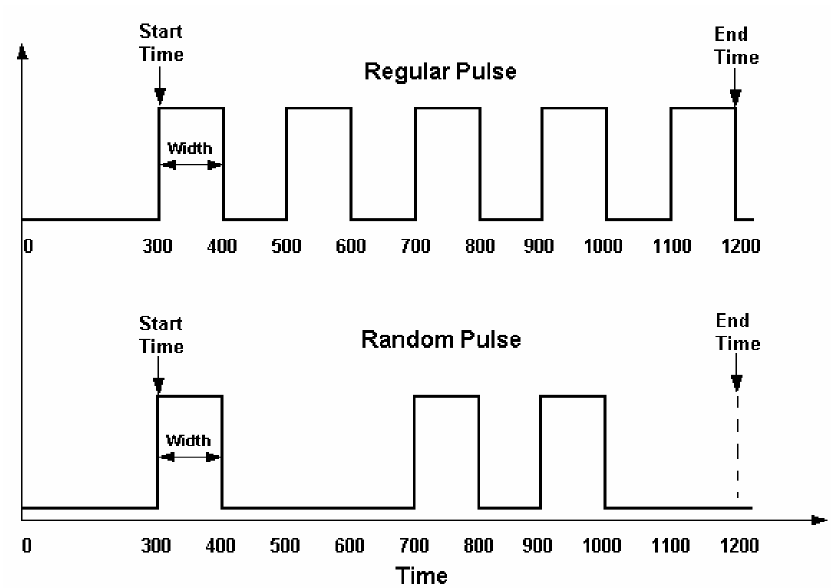
18.1 PULSE

Click on Inputs / Disturbance Signals Configuration or the icon and click on Pulse to see the Pulse signal configuration screen.

Click On to activate the pulse, or Off to deactivate the pulse.

See Figure 11 for a pulse signal illustration.

Figure 11. Pulse Disturbance Signal



Start Time is the start time of the pulse.

End Time is the end time of the pulse.

Width specifies the pulse width. If width is smaller than Start Time minus End Time, then multiple pulses can be generated.

A Regular pulse repeats regularly based on the pulse width and a Random pulse repeats itself randomly.

For the pulse shown in Figure 11, start time is 300, end time is 1200, pulse width is 100. Note the difference between a regular and random pulse. New pulses are randomly generated with a random pulse.

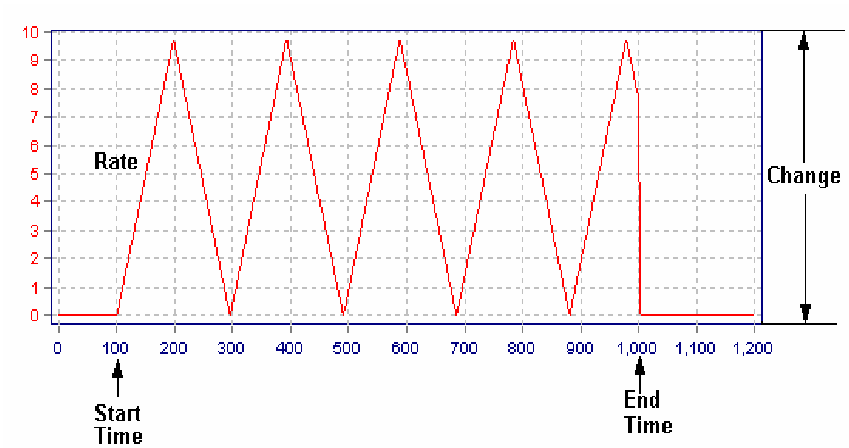
18.2 RAMP

Click on Inputs / Disturbance Signals Configuration or the icon  and then click on Ramp to see the ramp signal configuration screen.

Click On to activate the ramp, or Off to deactivate the ramp.

See Figure 12 for a ramp signal illustration.

Figure 12. Ramp Disturbance Signal



Start Time is the start time of the ramp.

End Time is the end time of the ramp.

Rate specifies the ramp rate (slope of the ramp).

The ramp signal starts increasing at the specified start time. After the value of the ramp equals Change, then the ramp signal starts reducing until it is zero. After zero, it starts increasing again till its value once again equals Change. The ramp can thus be made to increase and decrease.

For the ramp shown in Figure 12, start time is 100, end time is 1000, ramp rate is 0.1 and change is 10.

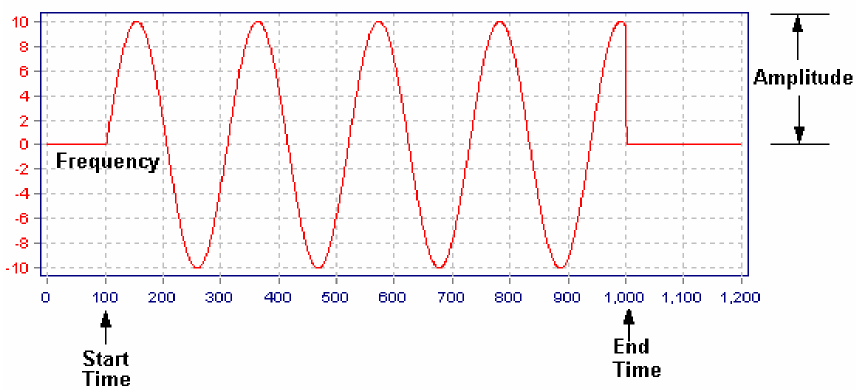
18.3 SINE

Click on Inputs / Disturbance Signals Configuration or the icon  and then click on Sine to see the sinusoidal signal configuration screen.

Click On to activate the sine signal, or Off to deactivate the signal.

See Figure 13 for a sinusoidal signal illustration.

Figure 13. Sinusoidal Disturbance Signal



Start Time is the start time of the sinusoidal signal.

End Time is the end time of the sinusoidal signal.

Frequency specifies the frequency of the sinusoidal signal.

Amplitude specifies the amplitude of the sinusoidal signal.

For the sine signal shown in Figure 13, start time is 100, end time is 1000, frequency is 0.03 and amplitude is 10.

One or more of the disturbance signals (pulse, ramp and sine) can be set active at any time to generate a complex signal representative of the real disturbance in the plant.

19.0 SIMULATE (RUN SIMULATOR)

Click on Run / Simulate or click on the icon to run the simulator and refresh the screen plots. If any configuration parameters (e.g., PID tuning, transfer function, ranges,...) are changed, the Simulate icon must be clicked to see the effect of these changes.

20.0 OPTIMIZE (OPTIMIZE TUNING)

Click on Run / Optimize or click on the icon to run the Pitops optimizer. The optimizer reads the specified (initial) PID parameters and searches for new improved parameters while trying to minimize the error. The error is defined based on the Error Criteria explained in an earlier section above. For the optimizer to work well, the following conditions are desirable. They will improve the chances of obtaining improved tuning parameters:

1. Configure a typical setpoint change (Required).
2. Configure a pulse disturbance for first or second order transfer function (Recommended).
3. Configure a ramp disturbance for a zero order transfer function (ramp, level control LC) (Recommended).
4. Initial PID parameters should not be unstable or oscillatory (Recommended).

PID tuning parameters before starting the optimization must be reasonable (stable). Optimization may fail if the initial PID tuning parameters are very aggressive. To determine whether your initial PID tuning parameters are very aggressive, see the simulation plots. If the response is very oscillatory, or the OP (PID OP – Valve Position) shown in the middle plot window) is bouncing between its high and low range limits, then the PID tuning parameters are too aggressive. In this case, try reducing the proportional action by a factor of 2-3. If control is still oscillatory, then try reducing the integral action by a factor of 2-3. Now rerun the simulation and examine the simulation plots.

Either a setpoint change or a disturbance or both must be specified in the configuration for the optimization to be successful. If a disturbance signal is not configured and if the setpoint old and new are the same, the simulation plots will display straight lines. In this case, the simulation is completely at steady state and the optimization algorithm has no information to successfully determine the optimum PID tuning constants.

The longer the simulation time, the longer will be the time required to complete the optimization. In most cases, simulation time of 300-800 time units is adequate.

21.0 OP LIMITS

Near the bottom right corner of the screen, you will see the OP Limits fields. There are two values: OP Low Limit and OP High Limit. These values allow clamping of the PID Output values. The PID OP gets clamped to these limits. Note that these limits are in % of the PID's MV Range. To illustrate this, click on the PID Configuration  icon.

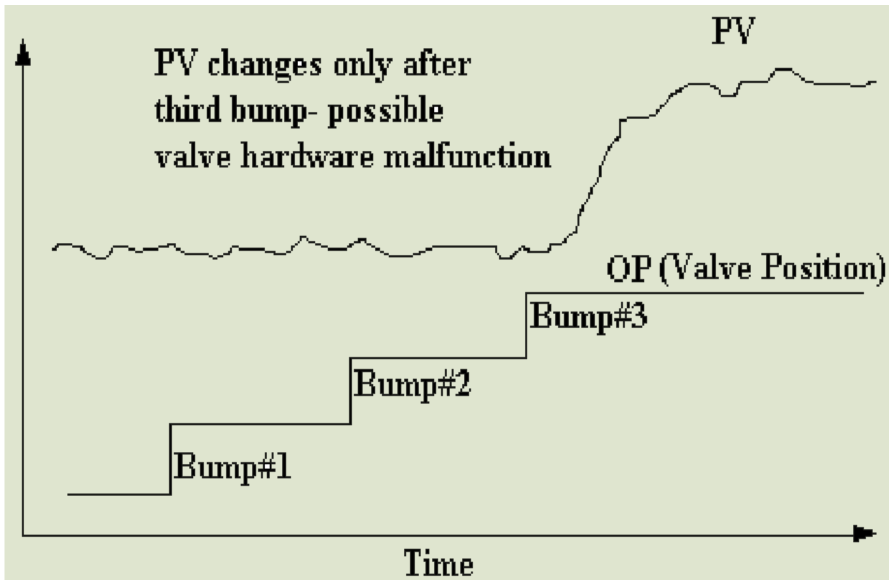
Set MV Range High = 200. Now, if OP High Limit is set to 75%, then the PID's OP will not be allowed to exceed 75% of 200, which is equal to 150. Note that the PID Low Limit and PID High Limit are in % units of the MV Range.

For a single PID going to a control valve, the MV Range is typically 0-100% and the PID Low Limit and PID High Limit are in % units.

22.0 VALVE STICTION AND PID RAW OP

Control valves can stick, have "loose-play" or "deadband" in their hardware linkages. As the PID OP changes, the valve may not move at all until the PID OP has changed by certain minimum threshold limit. This limit is called ***Stiction***. The stiction problem can be because of valve sticking or loose linkages. Stiction is often seen in compressor turbine nozzles, old control valves not maintained in a long time, valves servicing corrosive, abrasive materials or with poor valve hardware. Figure 14 illustrates a control valve problem.

Figure 14. Pulse Disturbance Signal



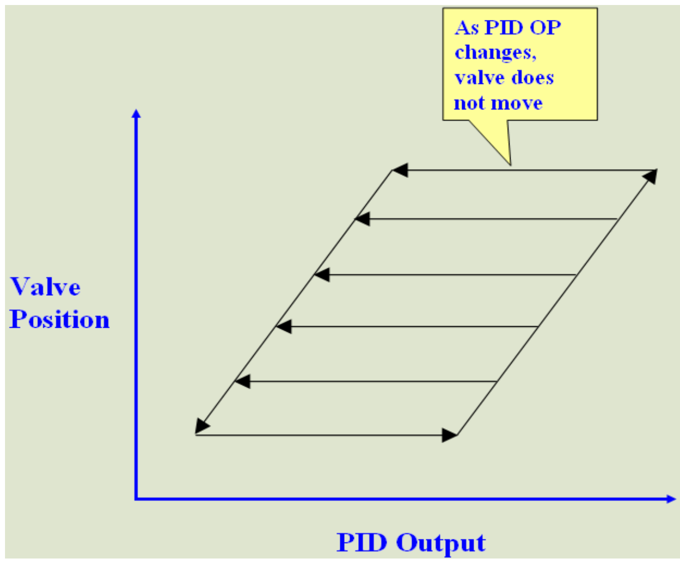
Only when the valve is bumped up three times, a noticeable change in flow is finally seen. This is clear proof of valve stiction. If there was no stiction, then the flow (PV) should have increased for the first two valve bumps also.

A typical relationship between the PID output and the actual valve position is shown in Figure 15.

Notice that once the PID OP has moved enough to overcome the stiction deadband, then the valve moves in accordance to the PID OP.

If the direction of the PID OP changes, then the valve will not move until the stiction deadband is traversed again.

Figure 15. Stiction Plot showing
PID OP versus Valve Position



Pitops-PID software simulates valve stiction. See the field labeled Valve Stiction located near the bottom right corner of the screen. Typical values of valve stiction are 0.5% to 25%. Valve stiction of over 10% is on the high side. Control quality will be poor when stiction is high. Stiction can generate unavoidable oscillations and will make the PID OP trend show a characteristic saw-tooth pattern.

Pitops-PID provides a powerful feature of optimizing PID tuning in the presence of valve stiction. Refer to a complete step-by-step example on PID simulation and optimization in the presence of valve stiction described in Part B, Section 11.

23.0 HELP

Click on Help / Help Topics in the top tool bar or click on the icon to open the Pitops-PID User Manual.

Click on Help / About in the top tool bar to see information on the Pitops-PID software product.

PART D: **PRACTICAL TIPS AND SPECIAL NOTES**

1.0 Time Unit
  1.1 How to Select Time Unit
  1.2 Effect of Fast Dynamics on Time Unit

2.0 Zoom and Scroll

3.0 Simulation Trend Box Notes
  3.1 Simulation Draws Flat Lines
  3.2 Simulation Shows Large Changes

4.0 Technical Help and Contact Information

PART D: **PRACTICAL TIPS AND SPECIAL NOTES**

A few important details related to the use of Pitops^{TM}-PID are summarized below:

1.0 TIME UNITS

Pitops does not require the time units to be explicitly defined. The time unit must be selected by the user. All variables must be specified in the selected time units. For example, if seconds are selected as the time unit, then all time dependent parameters must be specified in seconds. Similarly, if the time unit is selected to be minutes, then all time dependent parameters will be expected to be in minutes. The time dependent parameters are listed below as reference:

- Simulation Time
- Integral Time Constant
- Derivative Time Constant
- PID Execution Period (Scan Time)
- Transfer Function Time Constants (Tau1, Tau2)
- Delay
- Filter Time Constant
- PV Sample Delay
- Start, End Times, Pulse Width, Ramp Rate for Disturbance Signals

1.1 HOW TO SELECT TIME UNIT

The time unit needs to be selected based on dynamics of the process. E.g., compressor surge dynamics are fast and milli-second is an appropriate time unit. Distillation product impurity dynamics are slow and a minute is an appropriate time unit.

Once the time unit has been selected, all values for time dependent variables listed above must be in the same selected units.

This approach provides maximum flexibility. For example, simulation could be configured in the microsecond time frame without confusion on time units.

1.2 EFFECT OF FAST DYNAMICS ON TIME UNIT

Let's say you are simulating and tuning a PID for a process that has first order time constant (Tau1) of 2 minutes. In such a case, instead of selecting ***minutes*** as the time unit, it is better to select ***seconds*** as the time unit. Instead of 2 minutes time constant (Tau1), specify 120 seconds. In this case, you will see better trend plot resolution. In general, if the process is fast and the Tau1 value is under 3, it is better to simulate in a faster time unit.

Example 1: If Tau1 = 1.2 hour, use minutes as the time unit and use Tau1 = 1.2 * 60 = 72 minutes.

Example 2: If Tau1 = 1.0 minute, use seconds as the time unit and use Tau1 = 1.0 * 60 = 60 seconds.

Example 3: If Tau1 = 0.5 second, use milliseconds as the time unit and use Tau1 = 0.5 * 1000 = 500 milliseconds.

2.0 ZOOM AND SCROLL

To zoom on any section of the plot, click the left mouse button on the top left corner of the section to be zoomed and then move the mouse to the bottom right corner of the section, while keeping the left button continuously pressed. Then release the left mouse button. The section will be zoomed. To cancel the zoom and go back to the full screen, double click anywhere on the plot with the left mouse button.

The plot can be scrolled left to right or up and down by clicking with the right mouse button and then moving the mouse left-right or up-down. The right button must be kept continuously pressed during the scrolling. To go back to original display, double-click anywhere on the plot with the left mouse button.

3.0 SIMULATION TREND BOX NOTES

3.1 SIMULATION DRAWS FLAT LINES

If the simulation displays all flat lines with no dynamics, the problem may be due to the absence of any dynamic element specified in the configuration. To see some changes in the PV or OP signals, one or more of the following conditions must be true:

- SP change is specified (SP Old and SP New are different)
- PID gain must be non-zero
- Transfer function gains (disturbance, main) must be non-zero
- Disturbance signal must be specified
- Disturbance signal must be *on* (active)

Some form of dynamic element must be active to see a meaningful simulation.

In some cases, if the change in the PV or any signal is very small (change is smaller than 0.1%), Pitops may display a flat line on the simulation plots. If the size of the change is increased then the dynamics will be clearly visible.

3.2 SIMULATION SHOWS LARGE CHANGES

Pitops automatically scales the Y axes. It sets the Y axes extremities equal to the maximum and minimum in the simulation time frame. Sometimes a simulation signal may display a large change when little or none may be expected. In this case, the Y axes labels should be checked to see the maximum and minimum values. These may be almost equal (e.g., 11.98 - 12.02), causing the simulation to look as though a large change in the signal has taken place. To solve the problem, try a bigger SP change by increasing the difference between <u>SP Old</u> and <u>SP New</u> and/or activate one or more disturbance changes.

4.0 TECHNICAL HELP AND CONTACT INFORMATION

For free technical help and support, please contact the PiControl Solutions technical team at info@picontrolsolutions.com or visit our website at www.picontrolsolutions.com.