PID Basics

The purpose of this Interactive Learning Module (ILM) is to make you familiar with PID control. It is part of a series of modules that are designed as a complement to the book [1]. The other modules are *PID Loop Shaping* and *PID Windup*. The modules can be viewed as an attempt to make the key pictures in the book interactive. They give time and frequency domain views of the responses of a closed-loop system consisting of a PID controller and a process model. Many process models can be selected, controller parameters can be changed interactively, and the resulting responses are displayed instantaneously.

In the *PID Tuning* module, the basic view consists of responses to set-points, load disturbances, and high frequency measurement noise. Frequency responses and mixed time and frequency responses can also be shown.

There are two icons to access *Instructions* and *Theory*. *Instructions* gives access to this document which contains suggestions for exercises. *Theory* gives access to relevant theory via Internet.

The module can be used in many different ways, one extreme is a full-fledged exercise with serious analysis and reporting, another is simply free experimentation following your own ideas. Whatever you do it is good practice to record your observations. The modules can also be used in lectures. The following exercises are intended to give a good intuitive understanding of the properties of PID Control. The exercises are structured as follows:

1. Getting started
2. Basics
3. Load disturbance response
4. Set-point response
5. Measurement noise
6. Performance and robustness
7. Examples

1. Getting Started

Familiarize yourself with the interactive learning module by clicking on the different interactive elements. If you are reading this you are already familiar with the icon *Instructions*.

A default process transfer function is loaded when you start the system. A process transfer function appears on the screen together with buttons for selecting controller types and sliders to change the controller parameters. A number of parameters that quantifies robustness and performance of the closed-loop system are given on the screen. Two curves show the process output and the control signal when the system is subject to step changes in set-point, load disturbance, and high-frequency measurement noise. The buttons `save` and `delete` at the top right-hand corner of the screen makes it easy to store a simulation for comparison.

- Familiarize yourself with the system by observing what happens when you change the controller and its parameters.

- Change the dots and the vertical lines that controls the amplitudes of the set-point, the load disturbances and the noise.
- Click right and left, or above and below, the triangles that controls the scales.

Several options can be chosen from the *Settings* menu which has the following entries:

- Process transfer function
- Controller parameters
- Time/Frequency domain
- Load/Save
- Simulation
- Examples from [1]

Several process transfer functions can be selected, there is also an option to enter an arbitrary transfer function in the numerator (num) denominator (den) form used in Matlab. Controller parameters can be changed by sliders, but specific values can also be entered by choosing *Controller parameters* from the *Settings* menu.

The properties of the closed-loop system are normally shown as time responses of the Gang of Six, see [1] page 100. Frequency responses can also be shown from the *Time/Frequency domain* selection in the *Settings* menu. The loop transfer function $L = PC$, where P is the process transfer function and C is the controller transfer function, and the transfer functions

$$T = \frac{PC}{1+PC} \qquad PS = \frac{P}{1+PC} \qquad FT = \frac{FPC}{1+PC}$$
$$S = \frac{1}{1+PC} \qquad CS = \frac{C}{1+PC} \qquad FCS = \frac{FC}{1+PC}$$

can be shown, see Section 4.3 in [1].

Results can be stored and recalled using the *Load/Save* menu. Data can be saved and recalled using the options *Save design* and *Load design*. The option *Save report* can be used to save all essential data in ascii format. This is useful for documenting your results.

The menu selection *Simulation* makes in possible to choose the simulation time, and to activate the *Sweep* option which can be used to show the results for several controller parameters simultaneously. It is possible to load a number of figures from the book [1] using the selection *Examples* in the *Settings* menu. In this way you can explore what happens when you change the parameters in the examples.

Controllers of the type P, I, PI, PD and PID and their parameters can be chosen from the panel. The performance measures are integrated absolute error (`IAE`), `overshoot` for the set-point response, integral gain (`ki`), and maximum error (`emax`) for load disturbances. The `IAE` and the `emax` values are normalized to unit step changes in set-point and load disturbance. The response to noise is characterized by the standard deviations of the signals. The robustness measures are maximum sensitivity (`Ms`), maximum complementary sensitivity (`Mt`), gain margin (`gm`) and phase margin (`pm`).

2. Basics

A simple and intuitive way to familiarize yourself with PID control is to look at the responses of the closed-loop system in the time domain and to observe how the responses depend on the controller parameters. The module is set up for this configuration when you start it. The picture is similar to Figure 4.2 in [1]. Here are some simple exercises.

- Choose a proportional controller and observe what happens when the gain is changed. You can use the *Sweep* in the *Settings>Simulation* menu to display the results for three parameter values simultaneously. You can also use the `save` button to store a plot. Look at the process output and the control signal and try to explain intuitively what happens. Does the output reach the set-point? How does the steady-state error depend on the controller gain?
- Pick a value of the gain that gives nice-looking responses and save the results by clicking on the `save` box in the upper right corner. Then switch to integral control (I). Adjust integral time to get nice-looking curves. Use the *Save design* selection in the *Settings>Load/Save* menu to store data from settings you find interesting.
- Delete the results for proportional control by clicking on `delete` and on the (blue) curve that you want to delete. Click on `save` to save the results for integral control. Choose PI control and adjust the controller parameters. Look at the curves and make a qualitative comparison of I and PI controllers. Explore the effect of set-point weight (b) on the results. Use the *Save design* selection in the *Settings>Load/Save* menu to store interesting results.
- Summarize your comparisons of I and PI control. Use the *Load design* selection in the *Settings>Load/Save* menu to recall stored data.

The control action obtained with a PID controller has three components corresponding to the proportional, the integral and the derivative part. These components can be viewed separately by clicking on the boxes `P`, `I` and `D` in the window for the controller output. The following exercise gives you some insight.

- Load the default transfer function

  $$P(s) = \frac{1}{(s+1)(0.5s+1)(0.25s+1)(0.125s+1)}.$$

  Use the `Settings/Controller Parameters` menu to set set-point weight `b` to 1. Look at the proportional, integral and derivative terms of the control signal. Which are the dominating terms in the beginning of the step response? Which term reacts most rapidly to a load disturbance. What are the steady state values of the different terms? Which term gives the major contribution to the response due to measurement noise?

3. Load Disturbance Response

Load disturbances are disturbances that drive the system away from its desired behavior, they typically have low frequencies. The response to load disturbances is a key issue in process control, since most controllers attempt to keep process variables close to desired set-points. The purpose of this exercise is to investigate the effects of load disturbances and how they are influenced by controller type and parameter settings. In the simulation the load disturbance is a step at the process input.

To get the response to load disturbances you should set the set-point and the noise amplitudes to zero by dragging the bars in the display. With the default setting the amplitude of the load disturbance is 0.9. Investigate the load disturbance response by changing controllers and their parameters. Here are some suggestions for exercises.

Basics

- Choose a process with the transfer function

$$G(s) = \frac{1}{(1+s)(1+as)(1+a^2s^2)(1+a^3s^3)}$$

  from the settings menu and set $a = 1$.

- Choose a P-controller and observer the behavior of the process output and the control signal for different controller gains. What is the qualitative changes in the response when controller gain is changed? Save the results for a proportional gain that you think is reasonable by clicking on the `save` button in the upper right corner.

- Choose a PI controller and compare the response to load disturbance with the results obtained with proportional control. Compare the maximum errors (`emax`). Use the *Save design* selection in the *Settings>Load/Save* menu to store data from settings you find interesting. Can you estimate how the steady-state error depends on the controller gain? Find a value of the gain that gives a reasonable response, and note the numerical value for future use. The *Save report* option in the *Settings>Load/Save* manu can be used to store the results. Delete the curve for proportional control and save the response for PI control.

- Choose a PID controller and adjust the parameters to give a good response. Compare process output and control signal with the corresponding curves for PI control. What are the major differences? Use the *Save design* selection in the *Settings>Load/Save* menu to store data from settings you find interesting.

- Summarize performance and robustness measures for P, PI, and PID control in a table. Study the numerical values and make an assessment of the different controllers.

- Repeat the exercises when the process parameter is changed from $a = 1$ to $a = 0.2$. What can you say about the performances of P, PI, and PID controllers for the different values of a?

Ziegler-Nichols tuning

Several tuning methods for PID controllers are based on the information obtained when the system i brought to oscillation under pure proportional control. This exercise gives insight into such procedures.

- Choose the process with the transfer function

$$G(s) = \frac{1}{(1+s)(1+as)(1+a^2s^2)(1+a^3s^3)}$$

with $a = 1$. Choose proportional control with gain $K = 1$, and then set the gain equal to the ultimate gain K_u, which is equal to the gain margin. Explain why the system has a periodic oscillation. Note the period of the oscillation T_u. Save the periodic time responses.

- Use your notes of controller parameters from the exercise with load disturbances and compute the ratios K/K_u, T_i/T_u, and T_d/T_u. Make the same calculations for the system with $a = 0.2$.
- Compare your results with the Ziegler-Nichols tuning rules, see [1] page 159. What can you say about the Ziegler-Nichols method based on this exercise?

4. Set–Point Response

The response to set-points is important when making grade changes in process control. Following set-points is a key issue in motion control. The purpose of this exercise is to explore how the set-point response of the system is influenced by the controller parameters. Set the load disturbance and the noise amplitude to zero. Investigate the set-point response by changing controllers and their parameters. Here are some suggestions for exercises.

- Choose a process with the transfer function

$$G(s) = \frac{1}{(1+s)(1+as)(1+a^2s^2)(1+a^3s^3)}$$

from the settings menu and set $a = 1$.

- Choose a P-controller and observe the behavior of the output and the control signal for different controller gains. Use the *Sweep* option from the *Settings>Simulation* menu or the save button to see changes more clearly. Pick a gain that gives a reasonable response. Use the *Save design* selection in the *Settings>Load/Save* menu to store data from settings you find interesting.
- Use proportional control and pick a value of the gain that gives an oscillatory response. Save the response for proportional control and switch to PD control. Observe the behavior of the responses when the derivative time is increased.
- Delete the response with proportional control and save the response for PD control. Then investigate the effect of the parameter N on the response.
- Choose proportional control with the gain that gave good response, save the plots and switch to PI control. Find controller parameters that give good responses.
- Save a PI controller that gives good response. Investigate the effect of the parameter b on the response. Using the *Sweep* option you can directly see the effects of $b = 0$, 0.5 and 1.0.

- Repeat the exercise with $a = 0.2$.
- Summarize the results obtained for processes with $a = 1$ and $a = 0.2$. Can you find a simple way to characterize the differences between these processes?

5. Response to Measurement Noise

Measurement noise is a disturbance that distorts the information about the process that is obtained by the sensors. Measurement noise typically has high frequencies. Measurement noise is fed into the system by the feedback, it creates control actions and variations in the process output. It is important that measurement noise does not generate too large control actions.

- Choose a process with the transfer function

$$G(s) = \frac{1}{(1+s)(1+as)(1+a^2s^2)(1+a^3s^3)}$$

  with $a = 0.2$. Use the standard display that shows responses to set-point, load disturbance, and measurement noise. Choose PID control and adjust the parameters to give good responses. Save the plots by clicking on `save` button.

- Change parameter N and observe the changes in the responses. Take particular note of the fluctuations in the control signal. What conclusions can you draw?

6. Performance and Robustness

Performance can be expressed in many different ways. Integrated absolute error and maximum error are two useful performance measures for load disturbance response. Since process dynamics may change during operation it is important that the system is not too sensitive to process variations. The following exercises will give you an intuition of the trade-offs between performance and robustness.

- Consider a process with the transfer function

$$G(s) = \frac{1}{(s+1)^3}$$

  Choose a PI controller and adjust the controller gains to minimize the integrated absolute error for load disturbances. List the controller performance parameters `IAE`, emax, and the robustness parameters `Ms`, `Mt`, `gm`, and `pm`.

- Choose the option *Sweep* in *Settings>Simulation* and explore the effects of 50% variations in controller gain and integral time.
- Adjust controller parameter to give sensitivities smaller than 1.6. Look at the responses when the controller parameters are changed by 50%.

7. Examples

In this section we will give examples of different systems. The first example is a process with long time delay and the other examples are from the book [1].

Processes with Long Time Delays

It is sometimes claimed that systems with pure time delays cannot be well controlled using PID controllers. This exercise will give you intuition about control of processes with long time delays.

- Choose a process with the transfer $G(s) = \dfrac{1}{1+0.1s}e^{-s}$. This process is almost a pure time delay.
- Explore P and PI controllers. Determine parameters for a PI controller that gives a setpoint response without overshoot and a load disturbance response with low `IAE`.
- Save the curves for a PI controller. Switch to a PID controller and investigate the effect of introducing derivative action. Does derivative action give any improvements?
- Give the values of derivative time T_d and filter parameter N, that gives an unstable closed loop system. Can you explain the results analytically? Hint: Look at the frequency response of the loop transfer function.
- What happens to the control signal caused by measurement noise when derivative action is increased?
- Switch to the frequency domain. Investigate the transfer functions $CS = C/(1+PC)$ and $CP = P/(1+PC)$ for the PI and PID controllers.

Example 7.1 - MIGO and AMIGO design for lag-dominant processes

Load Example 7.1 from the *Examples* menu. The controller in this example was designed to maximize integral gain and to with a combined sensitivity less than 1.4.

- Investigate the performance that can be obtained for PI and PID controllers if the robustness requirement is relaxed to $M_s < 1.6$ and $M_t < 1.6$.
- Investigate the performance that is obtained if `IAE` is maximized and no sensitivity constraints are applied.

Example 7.2 - MIGO and AMIGO design for processes with balanced lag and delay

Load Example 7.2 from the *Examples* menu. The controller in this example was designed to give a combined sensitivity less than 1.4.

- Investigate the performance that can be obtained for PI and PID controllers if the robustness requirement is relaxed to $M_s < 1.6$ and $M_t < 1.6$.
- Investigate the performance that is obtained if `IAE` is maximized and no sensitivity constraints are applied.

Example 7.3 - MIGO and AMIGO design for delay-dominant processes

Load Example 7.3 from the *Examples* menu. The controller in this example was designed to give a combined sensitivity less than 1.4.

- Investigate the performance that can be obtained for PI and PID controllers if the robustness requirement is relaxed to $M_s < 1.6$ and $M_t < 1.6$.
- Investigate the performance that is obtained if `IAE` is maximized and no sensitivity constraints are applied.

Summary

Summarize your observations concerning PI and PID control of lag and delay dominant processes.

8. Reference

[1] Karl Johan Åström and Tore Hägglund. *Advanced PID Control*. ISA - The Instrumentation, Systems, and Automation Society, Research Triangle Park, NC 27709, 2005.