VRIJE
UNIVERSITEIT
BRUSSEL

PING PONG BALL

PID controller excercised

Péter Zoltán Csurcsia,
Nellie Van Eeckhaute, Cloë Theys
Pujan Bhandari

2024

1 Introduction

The goal of this session is to control the position of a ball in a transparent tube using a PID controller. For detection of the ball position, a camera is used. Once the actual position of the ball (PV, process variable) and the desired position of the ball (setpoint, SP) are known, the error signal (E) is calculated and used as the input to the controller. There are various options available to activate the different paths of the PID controller. An Arduino Uno transfers the desired PWM values (manipulated variable, MV) to the fans. The fan is utilized to regulate the height of the ball, facilitating its upward and downward movement within the tube. The fan operates (nearly) linearly, maintaining a(n almost) constant RPM input to achieve a new steady position for the ball, accounting for minor fluctuations or noise.
Performing different measurements and analyzing measured data in this session will help to understand the various aspects of the practical implementation of a PID controller.

2 PID-Controller

2.1 Overview

For convenience, we consider a parallel PID controller network. In the proportional branch of a PID controller, the error signal is multiplied by a gain (K). The integral controller continuously averages the accumulated error with a time constant (T_i). In the derivative branch, the rate of change of the error signal is multiplied by the gain and multiplied by the derivative time constant (T_d). The continuous-time PID equations are converted to the discrete-time domain using the so-called Tustin formula.
In closed-loop mode, the output of the system, also known as the process variable (PV), is measured by a sensor and compared with the desired value, known as the setpoint (SP). The error (E) is then calculated as the difference between the SP and PV and will be minimized by the closed-loop controller.

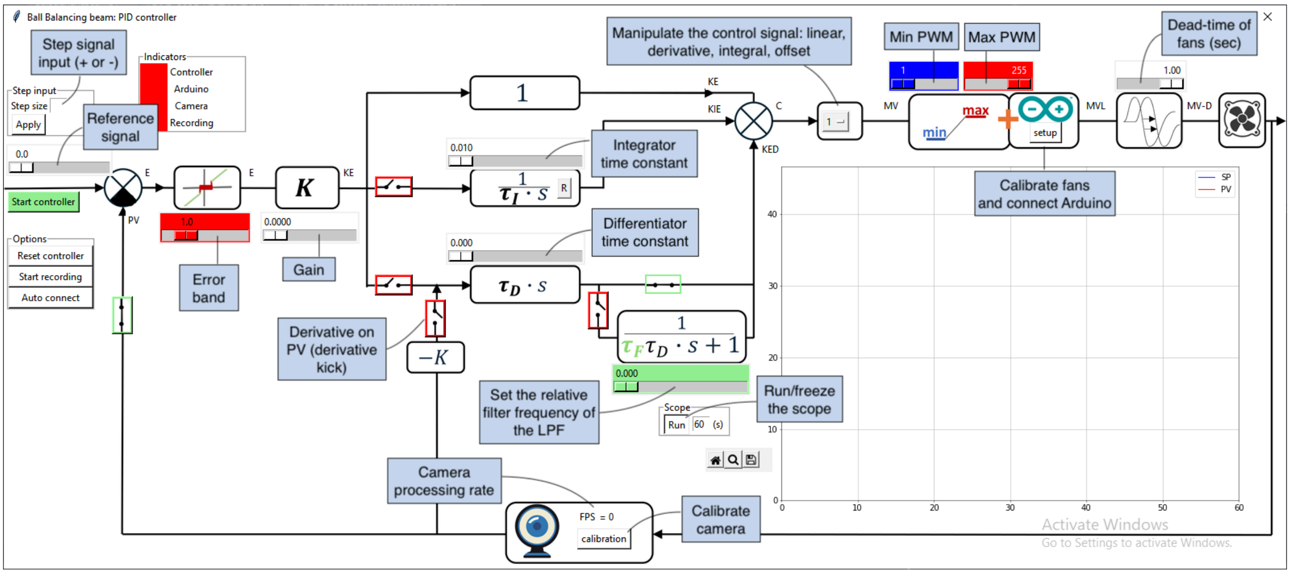


Figure 2.1: Overview of the closed-loop system and GUI.

2.2 Error deadband

When the error of the ball falls within the deadband, the error signal is set to zero. This prevents frequent unnecessary adjustments of the fans' speed when you know that the error cannot be smaller than a certain value.

2.3 Issues related to the D controller

In your previous lab exercises, the measurements (simulations) were assumed to be noise-free. However, in real life, there is measurement noise and disturbances. The ideal D controller amplifies high-frequency components that are in typical applications only noise (most systems are slow, meaning that we operate in the lower frequency band). The additional Low-pass filter helps to reduce high-frequency noise.

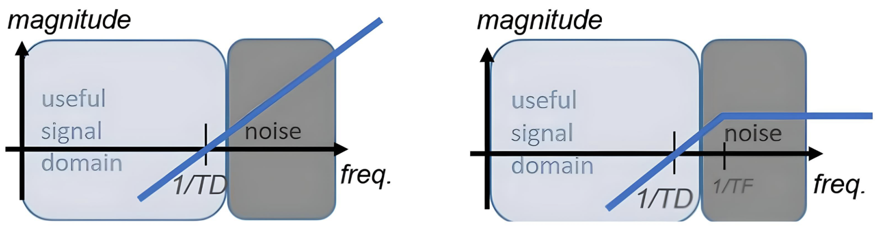


Figure 2.2: Derivative controller without and with a low-pass filter. Left: Transfer characteristic of an ideal D controller. Right: Transfer characteristic of an ideal D controller with a low-pass filter.

2.4 Derivative kick

When the setpoint (SP, the desired value) remains constant, it does not influence the derivative controller. However, when SP is suddenly changed (think of a step signal), it can cause an abrupt change that triggers an abnormal response in the derivative path. This phenomenon is referred to as "derivative kick". To reduce the derivative kick, the derivative path can be connected to the process variable (PV, the measured output signal). It is usually better because the ball position (PV) cannot change as volatilely as the setpoint resulting in better behaving derivatives.

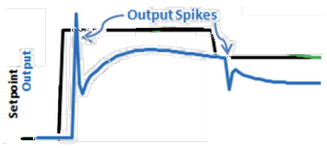


Figure 2.3: Presence of derivative kick

2.5 Manipulators

Signal manipulators are crucial for adjusting the output signal of a controller. In this software, there are four possibilities. The first one, labeled as linear (1), maintains the output signal unchanged. The second is the "integrator" ($\sum$) that accumulates (sums up) the control signal. The control signal (i.e., the PWM) will be larger and larger when the error is positive, smaller and smaller when the error is negative, and remains at the same level when the error is (nearly) zero. The derivative manipulator (Δ) calculates the difference between the current control value (PWM) and the previous one. The offset option (O) adds the minimum PWM value to the control signal.

3 First steps

Your lab instructor will provide a printed template and further instructions for completing these exercises. The block diagram that you see in the GUI represents the Ping Pong Ball demo setup that needs to be tuned. Follow these steps to start the controller successfully.

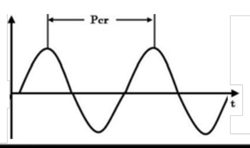
3.1 Calibration

1. Click on the button *setup* under the Arduino icon. This opens the *System calibration* window.
2. Click on the button *Refresh com ports*. Choose the COM-port connected to the Arduino from the list. If you are not sure which port is assigned to Arduino, open the device manager (or run the command: devmgmt.msc), go to the Ports (COM & LPT) list and look for the item USB-SERIAL CH340.
3. Click on the button *Connect arduino* and wait until the message "Arduino connected." appears.
4. Try to get the ball to a stable position using *Test PWM value* input. From here you can find the maximum PWM value by gradually increasing the PWM value. The maximum position PWM value is the value for which the ball is at the top of the tube, but still floating entirely inside the tube - cannot hover over the tube. If the correct setting is found use *set max position*. The minimum position and PWM value is the point where the ball begins to float - so it is not on the bottom of the tube. If the setting for the minimum is found use *set min position*. The usable (useful) height of the tube can automatically be read from the screen.

The basic calibration is ready, you can start the exercises. The goal is to find the tuning parameters by using different methods. There are nine sliders available to change the reference point, deadband for error, K, τ_I, τ_D, τ_F, minimum PWM, maximum PWM value, and deadtime respectively from left to right. The deadtime of the fans is the time delay between two signals sent to the fans.

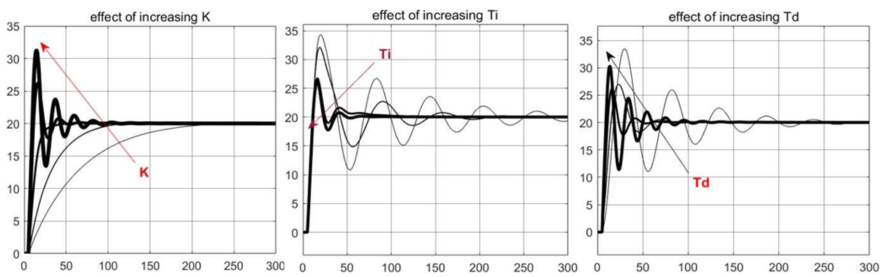
3.2 Ziegler-Nichols closed-loop method

1. Turn off any integrator and differentiator. Set K to a low value and turn the error band to zero.
2. Measure the step response of the closed loop system.
3. Gradually increase K until the system steadily oscillates. This value is the K_{Cr} – the critical gain.
4. Estimate the critical period time P_{Cr}.
5. Apply the following controller parameters from the table below. For fine tuning see the figure below on the right.



	K	τ_i	τ_D
PID	0.6Kcr	Pcr/2	Pcr/8

(a) Z-N closed loop tuning.



(b) Effect of the PID parameters.