PID controllers

Author:
Modestino Carbone

*"Everything I'd hope to find in the system;
control, order, perfection.
None of it meant a thing."*

Kevin Flynn, *Tron: Legacy* (2010)

1 Introduction

Proportional-Integral-Derivative (*PID*) controllers are one of the most widely used control algorithms in engineering and industrial automation. Their popularity stems from their simplicity, robustness, and effectiveness in a wide range of applications, from temperature control systems to motor speed regulation and robotic motion.
A *PID* controller continuously calculates an error value as the difference between a desired setpoint and a measured process variable (Picture of a PID controller fig. 1). It then applies a correction based on three terms: the proportional (P), which reacts to the current error; the integral (I), which accounts for the accumulation of past errors; and the derivative (D), which predicts future errors based on the current rate of change. By combining these three components, *PID* controllers can achieve precise and stable control, minimizing overshoot and settling time.
Despite their straightforward concept, tuning a *PID* controller to achieve optimal performance can be complex and often requires empirical adjustments or advanced optimization techniques.

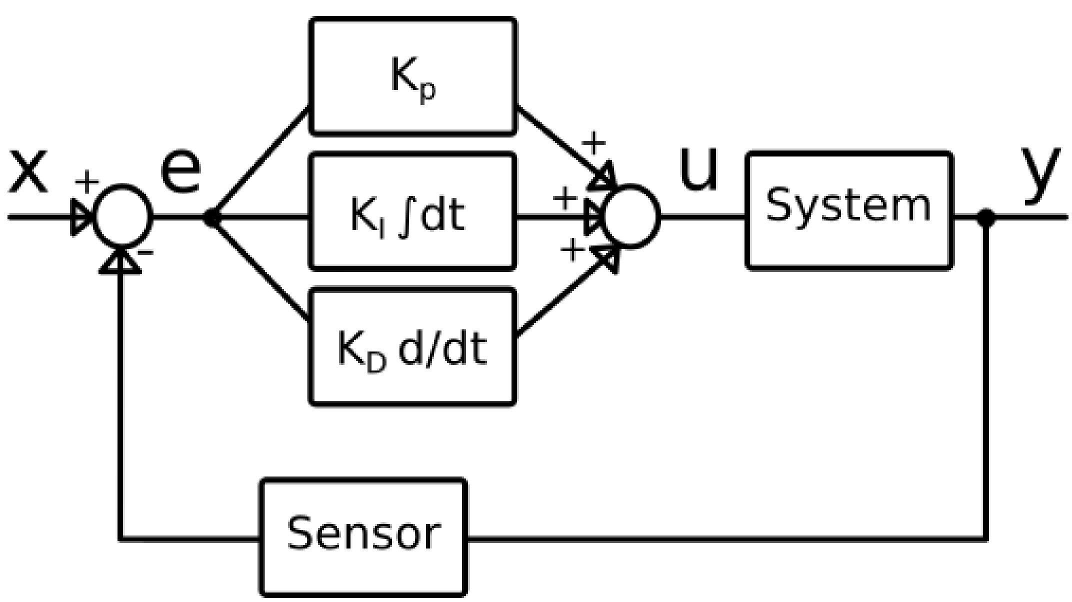

Figure 1: scheme of a PID controller

2 Foundamentals

A simple proportional controller is not always sufficient to guarantee that the setpoint is reached. When there is a non-zero error between the input and the output, the controller reacts proportionally to this error in an attempt to minimize it. Increasing the proportional gain can reduce the error more quickly, but it may also lead to oscillations in the system's output. To compensate for these oscillations and improve stability, the integral component is introduced.

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau \tag{1}$$

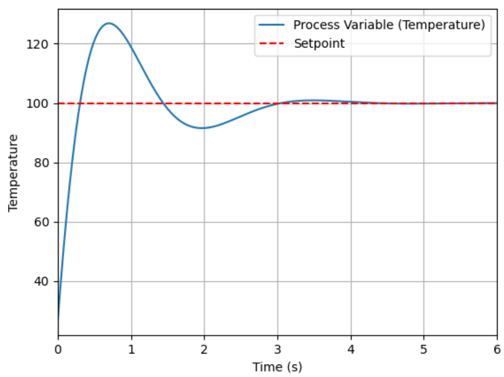

Figure 2: Start-up oscillation of the output of a controlled system (blue line), setpoint (dashed red line).

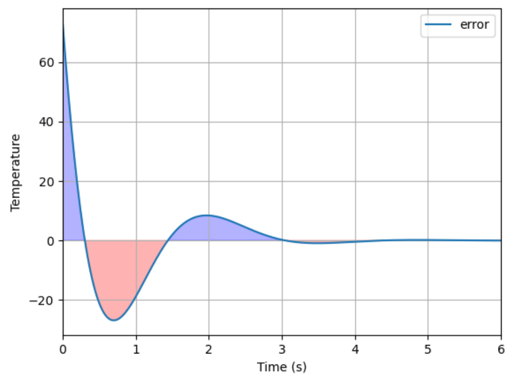

Figure 3: Error = setpoint - output of the controlled system.

The oscillatory behavior of the system depicted in Figure 2 enables the integral term to reduce the steady-state error by accumulating the persistent deviation from the setpoint.
As illustrated in Figure 3, the integral of the error can be either positive or negative, depending on the system's deviation from the setpoint over time.
The standard PID control formula, as discussed in the introduction, incorporates a derivative term that serves to anticipate the system's response, thereby improving stability and dynamic performance.

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{de(t)}{dt} \tag{2}$$

The figure below illustrates how the fundamental principle of differentiation can be used to predict the behavior of the system.

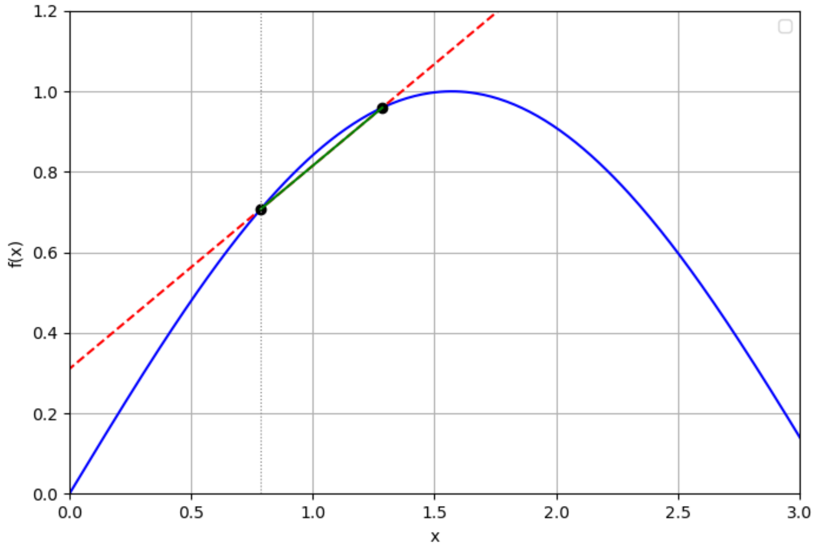

Figure 4: Differentiation principle

3 Ziegler-Nichols PID Tuning *for a Closed loop*

The Ziegler-Nichols method is a classic technique used to tune the parameters of a PID controller. Procedure (Closed-Loop Method):

- Set $K_i = 0$ and $K_d = 0$.
- Increase K_p gradually until the system output begins to oscillate with constant amplitude.
- The value of K_p at this point is called the ultimate gain: K_{cr}.
- Measure the oscillation period: T_{cr}.

The resulting closed loop used for the extarction of the parameters is showed below.

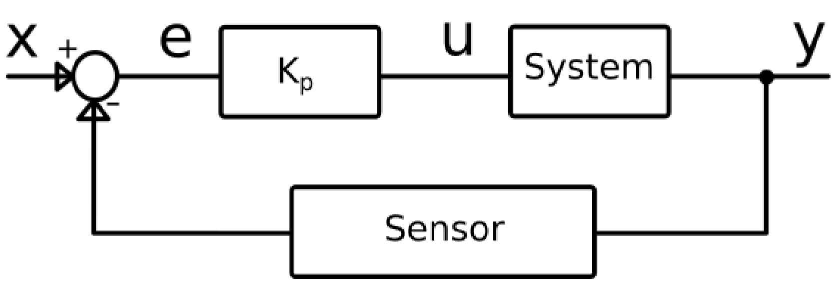

Figure 5: A single proportional controller

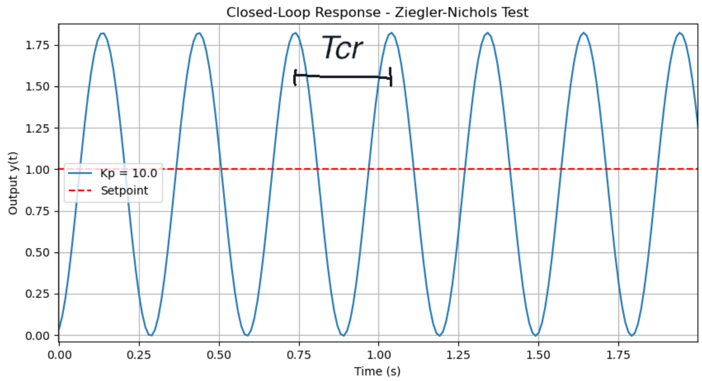

Figure 6: Oscillation of the output T_{cr} triggered by K_{cr}

This method assumes the system can reach sustained oscillations, and works best on stable or marginally stable systems. It's widely used in industry for initial tuning.

Controller Type	K_p	k_i	k_d
P	$0.5 \times K_{cr}$	0	0
PI	$0.45 \times K_{cr}$	$K_p / (0.83 \times T_{cr})$	0
PID	$0.6 \times K_{cr}$	$K_p / (0.5 \times T_{cr})$	$K_p \times (0.125 \times T_{cr})$