# Control Engineering II

## Practice 1:
## Calculation of Discretized PID Controllers

C.A. Monje, D. Copaci, M. Malfaz, J. Muñoz,
L. Mena, L. Moreno, S. Garrido

2026

uc3m | Universidad Carlos III de Madrid
OpenCourseWare

## 1. Digital control systems

Controllers in a control system can be implemented using several techniques and components, for example: mechanical systems, hydraulic systems, analog or digital electronic circuits, and also microprocessors. Compared with analog components, the latter have a relatively slow response; however, the continuous development of faster microprocessors means that, in practice, this limitation is becoming negligible, providing great versatility in the design and tuning of the controller parameters. In general, the typical topology to be considered in digital control systems is shown in Figure 1.

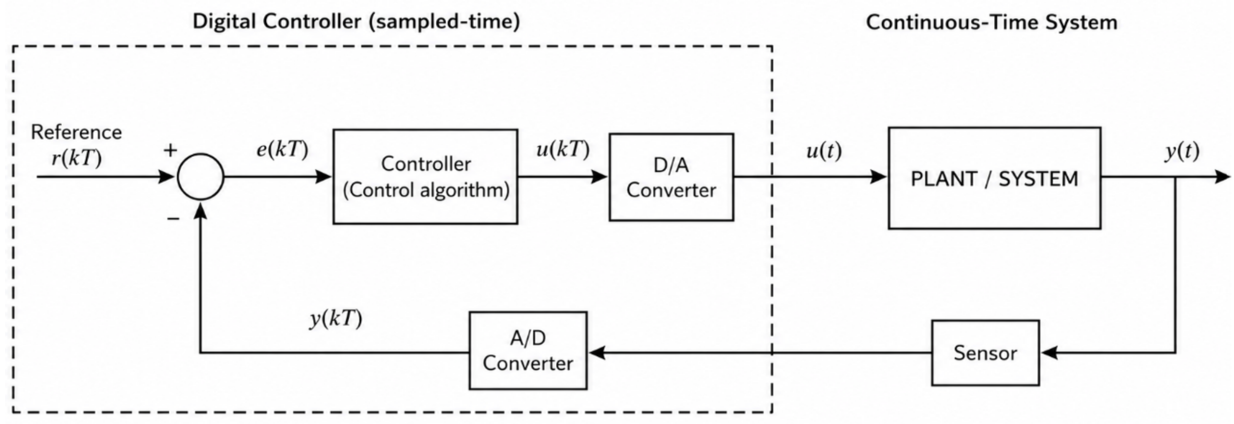


Figure 1: Digital control system with microprocessor

In general, digital control systems require the use of analog-to-digital (A/D) and digital-to-analog (D/A) converters, since the I/O signals of the computer are digital and the plant or process to be controlled usually uses analog signals (Figure 1). Both converters sample the signal every $T_s$ seconds, where $T_s$ is the sampling period and is one of the most important parameters to consider in the design of digital control systems.

In the digital control system shown in Figure 1, the controller is a high- or low-level program executed on a dedicated processor with the minimum hardware equipment required to perform the control tasks, or, as in our case, on a personal computer.

## 2. Design of digital control algorithms

*2.1 Discretization Principles*

*2.1.1. Approximation of differential equations by difference equations*

To be implemented on a digital computer, the dynamic equations of the system components must be written as difference equations that approximate the differential equations or continuous transfer functions of the system. Several techniques exist to approximate a differential equation by means of difference equations.

In the case of the derivative, there are three basic techniques: forward-difference approximation, backward-difference approximation, and trapezoidal approximation. The forward-difference approximation is expressed as follows:

$$\left.\frac{dy}{dt}\right|_{k=KT_s} = \frac{y(k+1)-y(k)}{T_s} \quad (1)$$

where $T_s$ is the sampling period.
If we use the previous expression to discretize a continuous transfer function $G(s)$ and obtain its Z-transform, we have:

$$G_1(z) = G(s)\Big|_{s=\frac{z-1}{T_s}} \quad (2)$$

On the other hand, the backward-difference approximation states that:

$$\left.\frac{dy}{dt}\right|_{k=KT_s} = \frac{y(k)-y(k-1)}{T_s} \quad (3)$$

When discretizing the transfer function $G(s)$ using the previous expression, we obtain:

$$G_1(z) = G(s)\Big|_{s=\frac{1-z^{-1}}{T_s}} \quad (4)$$

Finally, the trapezoidal approximation states that:

$$G_1(z) = G(s)\Big|_{s=\frac{T_s}{2}\left(\frac{1+z^{-1}}{1-z^{-1}}\right)} \quad (5)$$

To approximate an integral, there are equivalent methods that allow us to model a continuous-time system in discrete time. It is important to emphasize that, although the previous approximation techniques differ only slightly from one another, the dynamic characteristics of the resulting difference equations can vary significantly and may not correspond directly to those of their respective differential equations. In this regard, the trapezoidal method provides the best approximation accuracy.

*2.1.2. Discretization using the Z-transform of G(s)*

The effects of the discrete processing of signals in a digital control system, and therefore of the A/D and D/A converters, can be modeled by means of a zero-order sampler and hold, as shown in Figure 2.

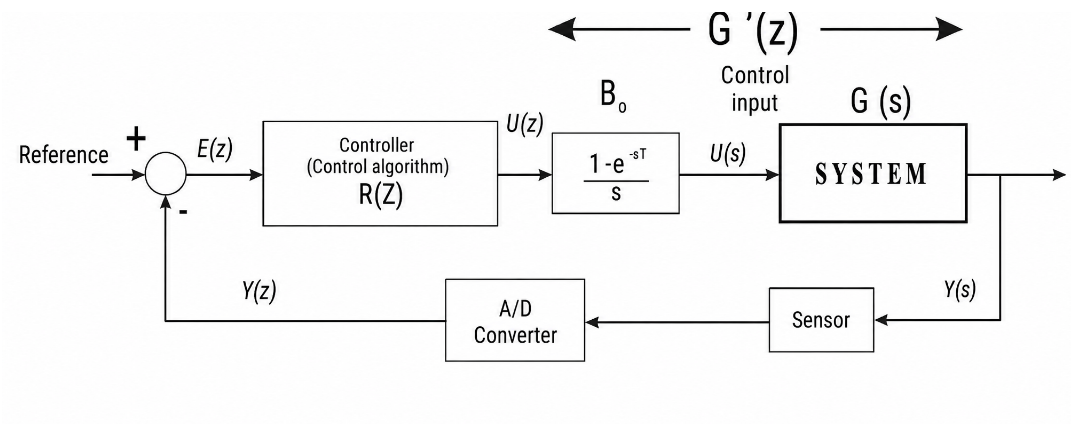


Figure 2: Modeling of a digital control system with a zero-order sampler and hold

In this case, the Z-domain transfer function of the process-hold system is given by:

$$G^{'}(z) = Z\left\{\frac{1-e^{-sT_s}}{s}G(s)\right\} = (1-z^{-1})Z\left\{\frac{G(s)}{s}\right\}. \tag{6}$$

*2.2 Design of digital controllers*

In practice, there are two approaches to designing digital controllers:

1. **Continuous design followed by discretization.** In this case, the controller is designed as if it were a continuous-time system, and the result is subsequently discretized.
2. **Direct digital design.** The plant model is discretized, and the controller is designed using discrete-time techniques based on the scheme shown in Figure 2.

*2.2.1. The digital PID controller*

The PID controller can be discretized by replacing the derivative with a backward difference:

$$\frac{d\varepsilon}{dt} \approx \frac{\varepsilon\left(kT_s\right) - \varepsilon((k-1)T_s)}{T_s}, \tag{7}$$

and the integral with a summation, in this case also using a backward rectangular approximation:

$$\int \varepsilon dt = T_s \sum_{i=0}^{k} \varepsilon(i), \tag{8}$$

where $T_s$ is the sampling period of the error signal.

Considering that the signals do not change during the sampling interval, and that $T_s$ is given by the control cycle time of our program, the expression for the digital PID controller can be written as follows:

$$u\left(k\right) = C\varepsilon\left(k\right) + \frac{T_d}{T_s}\left(\varepsilon\left(k\right) - \varepsilon\left(k-1\right)\right) + \frac{T_s}{T_i}\sum_{i=0}^{k} \varepsilon(i). \tag{9}$$

The previous expression is known as the "positional" form of the digital PID controller. The "incremental" form is obtained from the previous equation using a forward rectangular approximation, yielding:

$$\Delta u(k) = C(\varepsilon(k) - \varepsilon(k-1)) + \frac{T_d}{T_s}(\varepsilon(k) - 2\varepsilon(k-1) + \varepsilon(k-2)) + \frac{T_s}{T_i}\varepsilon(k) \quad (10)$$

The difference between both expressions is that, at each sampling instant, the incremental form provides an increment of the control signal, whereas the positional form gives an absolute value of $u(k)$.

*2.2.2. Sampling time*

As observed in the previous equations, in the design of digital controllers, the appropriate selection of the sampling period $T_s$ is important and depends on the system. A value of $T_s$ that is too large degrades the stability of the system, leading to loss of information about the signals if they change rapidly. On the other hand, if the value of $T_s$ is too small, faster A/D converters with higher resolution will be required.
In order to properly reconstruct the sampled signal, Shannon's theorem establishes a criterion for selecting $T_s$, that is:

$$\omega_s > 2\omega_b \quad (11)$$

where $\omega_s$ is the sampling frequency and $\omega_b$ is the bandwidth of the signal, both measured in rad/s.

There are also practical rules for determining the sampling time. One commonly used by control engineers is to satisfy the following relationship:

$$T_s < \frac{T_{dom}}{10} \quad (12)$$

where $T_s$ is the sampling period and $T_{dom}$ is the dominant time constant of the system. In this practice, we choose $T_s = T_{dom}/10$. Industrial processes are usually modeled with varying degrees of accuracy as first-order systems. In these systems, the time constant corresponds to the time at which the output reaches 0.63 of its final value.

In general, sampling intervals of 1 second are suitable for flow, pressure, level, and temperature processes, whereas electromechanical systems require sampling times on the order of milliseconds.

### 3. Position control of a DC motor

When different actuation methods and position and velocity measurements are considered, motors provide practical examples of characteristic situations in real-time control systems; furthermore, they are relatively inexpensive and provide a good learning environment. On the other hand, the software developed for motor control can be applied to other control problems by making minor changes to its configuration. Figure 3 shows a simplified control system for a direct current motor.

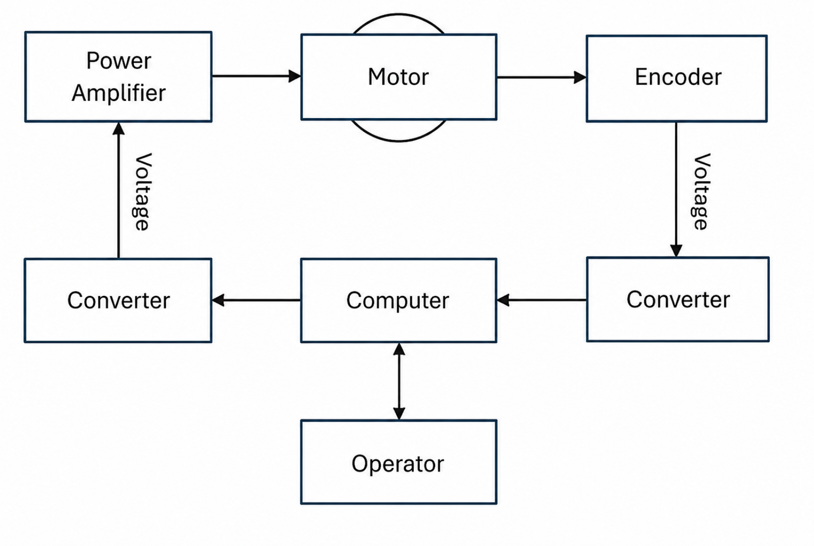


Figure 3: Digital control system of a DC motor

*3.1. The physical system*

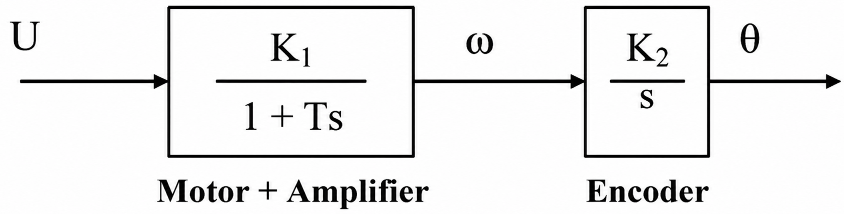


Figure 4: Motor system with position output

The open-loop transfer function of the motor with position output to be considered, as shown in Figure 4, is given by:

$$\frac{\theta(s)}{U(s)} = \frac{K_1}{(1+Ts)}\frac{K_2}{s} \tag{13}$$

where $K_1$, $K_2$ and $T$ are the gains of the motor and the encoder, and the motor time constant, respectively. The identification of the system parameters is performed by analyzing the time response of the system in open loop.

*3.2. PID controller design*

To control the motor position, a continuous PID controller is calculated for the transfer function (13). Since the motor has a pole at the origin, it is not necessary to compute the integral action of the PID controller; therefore, to control the system, it is sufficient to calculate a PD controller of the form $R(s) = K(s + a)$.

Once the continuous PID controller has been calculated, the system is discretized. For this purpose, backward rectangular approximate discretization is used, making the following change of variable:

$$s = \frac{1-z^{-1}}{T_s} \tag{14}$$

where $T_s$ is the sampling time. The resulting controller is:

$$R(z) = K(\frac{1-z^{-1}}{T_s} + a) \tag{15}$$

### *3.3. Sampling time*

The sampling time of the digital control system can be selected using the bandwidth of the closed-loop system, which is defined as the frequency at which the gain of the closed-loop system is 3 dB below the gain at zero frequency, or alternatively as 10 times smaller than the time constant of the system.

## 4. Simulation work

Given the direct current motor with the following parameters, according to the scheme in Figure 4: $K_1 K_2 = 3.60$ and $T = 0.12$. Design the ideal continuous PD controller required for the closed-loop system to satisfy the following specifications:

- $M_p$ = 12%
- $t_p$ = 0.44 seconds

*To check the results obtained in this section, complete the file $P1$.mlx and run it using the "Run" button. To do this, first set the current MATLAB directory to the folder of the practice session (see Figure 5) and then open the file $P1$.mlx from the current directory.*

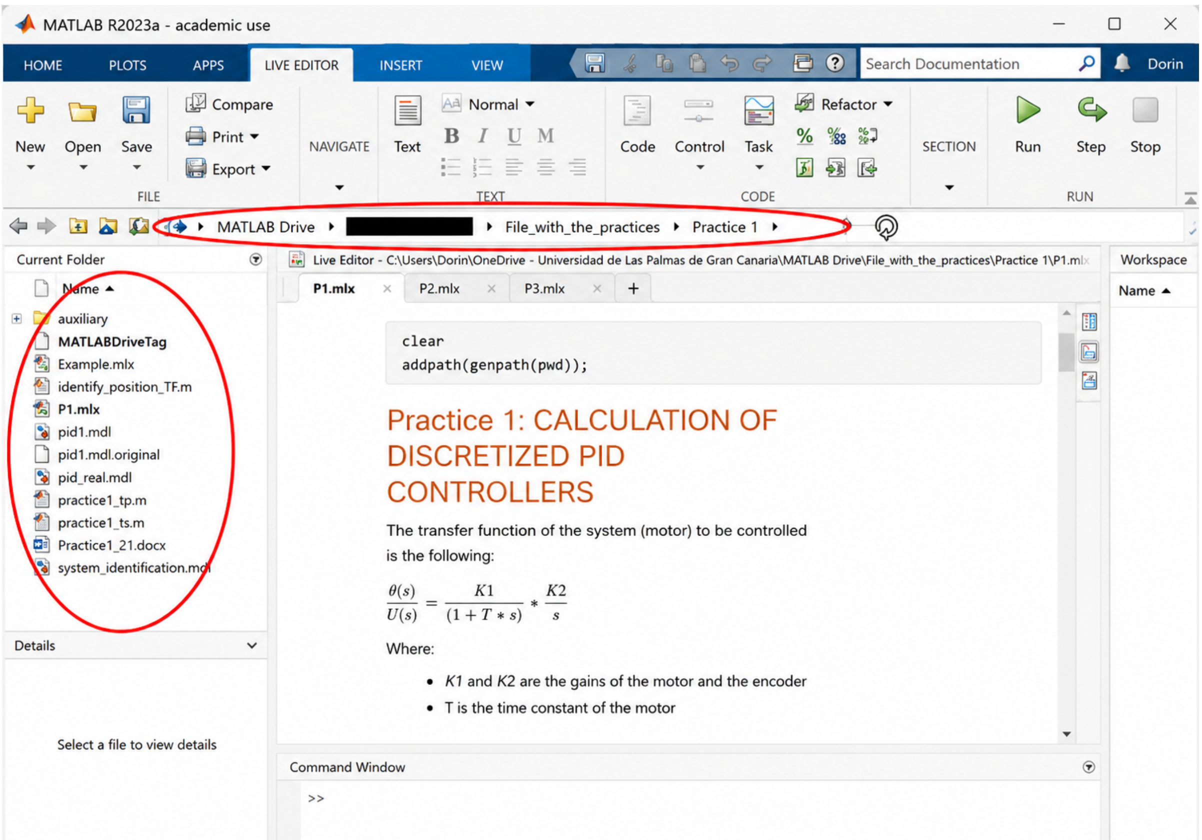


Figure 5: MATLAB interface

# P1- Calculation of Discretized PID Controllers

1. Once this has been done, discretize the controller using the backward Euler method and thus obtain the expression of the PD controller as a function of $z$. Use the sampling period considered most appropriate.
2. Launch the MATLAB Simulink tool and open the file "$pid1.mdl$". Once this has been done, become familiar with each of its blocks, identifying the equivalence between the system in Figure 4 and its feedback scheme represented in Simulink.
3. Assign the values to each of the blocks according to the given motor parameters and those obtained for the controller.
4. Using the file $pid1.mdl$, analyze the results obtained by comparing them with the problem specifications.
   - Measure the settling time, peak time, and overshoot of the closed-loop system from the obtained response.
   - Plot both the output and the control action.
5. Analyze the influence of the sampling period on the results. To do this, modify the sampling time and introduce the new discrete controller obtained.
6. Analyze the influence of the system parameters by studying what happens when these parameters are varied.