Digital PID

Kjartan Halvorsen

2021-07-12

Context

- Controller $F(s)$ obtained from a design in continuous time.
- Need discrete approxmation in order to implement on a computer

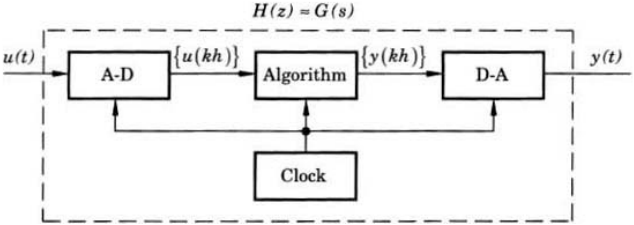
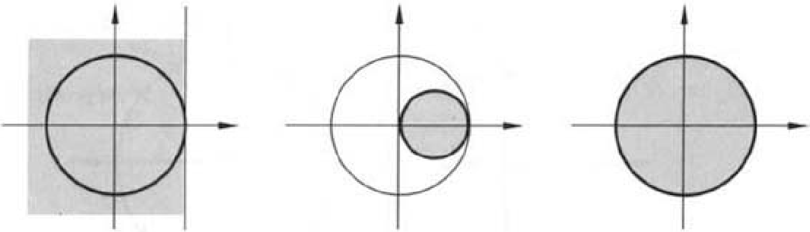


Figure 8.1 Approximating a continuous-time transfer function, $G(s)$, using a computer.

Åström and Wittenmark *Computer-controlled systems*

Mapping of the stable region of the s-plane



Forward differences Backward differences Tustin

Figure 8.2 Mapping of the stability region in the s-plane on the z-plane for the transformations (8.4), (8.5), and (8.6).

Åström and Wittenmark *Computer-controlled systems*

ISA form of the PID

ISA - International Society of Automation

$$F(s) = K_c \left(1 + \frac{1}{T_i s} + T_d s\right)$$

With low-pass filter for the derivative part

$$F(s) = K_c \left(1 + \frac{1}{T_i s} + \frac{T_d s}{\frac{T_d}{N} s + 1}\right), \quad N \approx 3 \ - \ 10$$

ISA form of the PID - derivative part

$$F(s) = K_c \left(1 + \frac{1}{T_i s} + \frac{T_d s}{\frac{T_d}{N} s + 1} \right), \quad N \approx 3 \; - \; 10$$

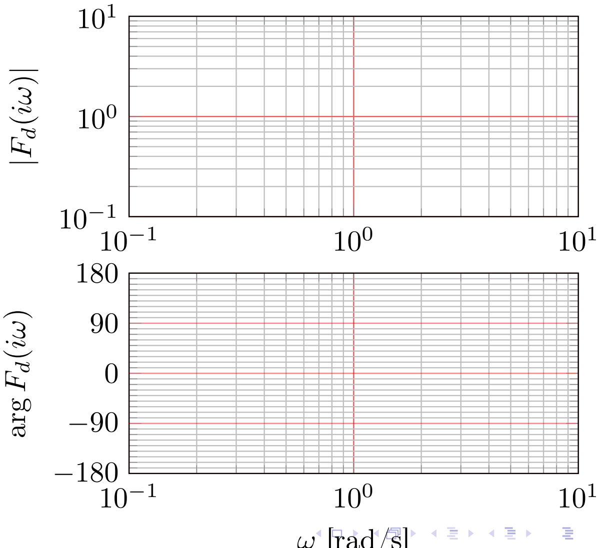
Activity Sketch the Bode plot for the derivative part ($T_d = 1$, $N = 10$)

$$F_d(s) = \frac{T_d s}{\frac{T_d}{N} s + 1}$$

using the low-frequency and high-frequency approximations.

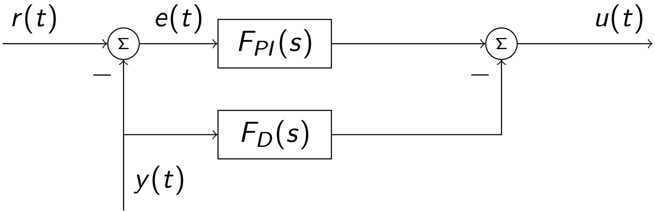
$$\omega \text{ small:} \quad F_d(i\omega) \approx T_d i\omega$$

$$\omega \text{ large:} \quad F_d(i\omega) \approx \frac{T_d i\omega}{\frac{T_d}{N} i\omega} = N$$



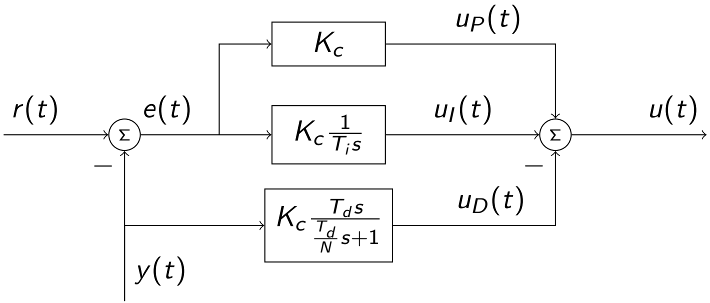
ISA form of the PID - derivative part

PID with derivative action only on the process variable



$$U(s) = \underbrace{K_c\left(1 + \frac{1}{T_i s}\right)}_{F_{PI}(s)} E(s) - \underbrace{\frac{T_d s}{\frac{T_d}{N}s + 1}}_{F_D} Y(s)$$

Common discretization of the PID



$$U(s) = U_P(s) + U_I(s) - U_D(s) = K_c E(s) + K_c \frac{1}{T_i s} E(s) - \frac{T_d s}{\frac{T_d}{N}s + 1} Y(s)$$

Activity 1) Use Euler's method $s \approx \frac{z-1}{h}$ for the integral part, and the backward difference $s \approx \frac{z-1}{zh}$ for the derivative part. 2) Apply the inverse z-transform to obtain the controller in the form of a difference equation.

The discrete PID algorithm

Dado: $y(kh-h),\ u_I(kh-h),\ u_D(kh-h)$

Sample signals: $r(kh),\ y(kh)$

$$e(kh) = r(kh) - y(kh)$$

$$u_P(kh) = K_c e(kh)$$

$$u_D(kh) = \frac{\frac{T_d}{N}}{\frac{T_d}{N}+h} u_D(kh-h) + K_c \frac{T_d}{\frac{T_d}{N}+h}\big(y(kh) - y(kh-h)\big)$$

$$u(kh) = u_P(kh) + u_I(kh-h) + u_D(kh), \qquad \text{Send to DAC}$$

$$u_I(kh) = u_I(kh-h) + K_c \frac{h}{T_i} e(kh)$$