Automatique

A. Hably[1]

[1]Gipsa-lab, Grenoble-INP
ahmad.hably@grenoble-inp.fr

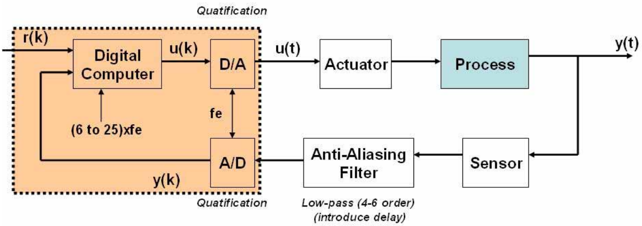
Commande d'un robot mobile

Outline

1. Digital implementation
   - Z-Transform
   - ZOH
   - Approximations
   - Extra

2. Application

Digital implementation of controllers



- Signals must be **sampled** and **quantized**
- A digital computer calculate the proper control signal value
- **Limits** on the speed or the bandwidth of the digital controller.

- One way to design a computer-controlled control system is to make a continuous-time design and then make a discrete-time approximation of this controller (Analog Design Digital Implementation).
- The computer-controlled system should now behave as the continuous-time system.

Definitions

- **Analog-to-digital converter (A/D)** : A device that samples a physical variable $y(t)$ (in V for example) to $y(kT)$ with T the sample period ($1/T$ sample rate in Hertz) and converts it to a binary number (bits).
- **Digital-to-analog converter (D/A)** : Converter changes the binary number to analog voltage

Z-Transform

Mathematical definition

The output of the ideal sampler, $x^*(t)$, is a series of impulses with values $x(kT_e)$, we have :

$$x^*(t) = \sum_{k=0}^{\infty} x(kT_e)\delta(t - kT_e)$$

Using the Laplace transform,

$$\mathcal{L}[x^*(t)] = \sum_{k=0}^{\infty} x(kT_e)e^{-ksT_e}$$

Noting $z = e^{sT_e}$, one can derive the so called Z-Transform

$$X(z) = Z[x(k)] = \sum_{k=0}^{\infty} x(k)z^{-k}$$

Z-Transform

Properties

$$X(z) = Z[x(k)] = \sum_{k=0}^{\infty} x(k) z^{-k}$$

Properties

$$\begin{aligned}
Z[\alpha x(k) + \beta y(k)] &= \alpha X(z) + \beta Y(z) \\
Z[x(k-n)] &= z^{-n} Z[x(k)] \\
Z[kx(k)] &= -z \frac{d}{dz} Z[x(k)] \\
Z[x(k) * y(k)] &= X(z).Y(z) \\
\lim_{k \to \infty} x(k) &= \lim_{1 \to z^{-1}} (z-1) X(z)
\end{aligned}$$

The z^{-1} can be interpreted as a pure delay operator

Z-Transform

Exercise

Determine the Z-Transform of the step function and of the ramp function

$$x_{step}(k) = 1 \text{ if } k \geq 0 \qquad\qquad x_{ramp}(k) = k \text{ if } k \geq 0$$
$$\phantom{x_{step}(k)} = 0 \text{ if } k < 0 \qquad\qquad \phantom{x_{ramp}(k)} = 0 \text{ if } k < 0$$

Z-Transform

Exercise

Determine the Z-Transform of the step function and of the ramp function

$$x_{step}(k) = 1 \text{ if } k \geq 0 \qquad x_{ramp}(k) = k \text{ if } k \geq 0$$
$$= 0 \text{ if } k < 0 \qquad = 0 \text{ if } k < 0$$

Solution

1) Step

$$X_{step}(z) = 1 + z^{-1} + z^{-2} + \cdots = \frac{1}{1 - z^{-1}} = \frac{z}{z-1}$$

2) Ramp (note that $x_{ramp}(k) = kx_{step}(k)$)

$$\begin{aligned} X_{ramp}(z) &= -z\frac{d}{dz}\left(\frac{z}{z-1}\right) \\ &= \frac{z}{(z-1)^2} \end{aligned}$$

Some useful transformations

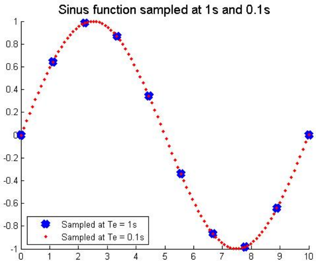
$x(t)$	$X(s)$	$X(z)$
$\delta(t)$	1	1
$\delta(t-kT_e)$	e^{-ksT_e}	z^{-k}
$u(t)$	$\frac{1}{s}$	$\frac{z}{z-1}$
t	$\frac{1}{s^2}$	$\frac{zT_e}{(z-1)^2}$
e^{-at}	$\frac{1}{s+a}$	$\frac{z}{z-e^{-aT_e}}$
$1-e^{-at}$	$\frac{1}{s(s+a)}$	$\frac{z(1-e^{-aT_e})}{(z-1)(z-e^{-aT_e})}$
$sin(\omega t)$	$\frac{\omega}{s^2+\omega^2}$	$\frac{zsin(\omega T_e)}{z^2-2zcos(\omega T_e)+1}$
$cos(\omega t)$	$\frac{s}{s^2+\omega^2}$	$\frac{z(z-cos(\omega T_e))}{z^2-2zcos(\omega T_e)+1}$

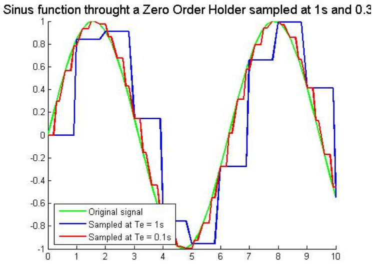
Sampler - Zero-order-hold (ZOH)

- **Sampler** : a switch that close every T_e seconds
- **Zero-order-hold (ZOH)** : Maintains the signal x throughout the sample period to get h as :

$$h(t + kT_e) = x(kT_e), \quad 0 \leq t < T_e$$

Exercise :





Zero-order-hold

Model of the Zero-order-hold

The transfer function of the zero-order-hold is given by :

$$G_{ZOH}(s) = \frac{1}{s} - \frac{e^{-sT_e}}{s} = \frac{1 - e^{-sT_e}}{s}$$

Exercise

Discretize (sampling time T_e) the system described by the Laplace function (using a Zero-order-hold) :

$$H(s) = \frac{Y(s)}{U(s)} = \frac{1}{s(s+1)}$$

Exercise

Discretize (sampling time T_e) the system described by the Laplace function (using a Zero-order-hold) :

$$H(s) = \frac{Y(s)}{U(s)} = \frac{1}{s(s+1)}$$

Adding the Zero order hold leads to :

$$\begin{aligned} G_{ZOH}(s)H(s) &= \frac{1-e^{-sT_e}}{s}\frac{1}{s(s+1)} \\ &= \frac{1-e^{-sT_e}}{s^2(s+1)} \\ &= (1-e^{-sT_e})\left(\frac{1}{s^2}-\frac{1}{s}+\frac{1}{s+1}\right) \end{aligned}$$

hence

$$Z[G_{ZOH}(s)H(s)] = (1-z^{-1})Z\left[\frac{1}{s^2}-\frac{1}{s}+\frac{1}{s+1}\right]$$

Exercise (cont'd)

Automatique

A.Hably

Digital implementation
- Z-Transform
- ZOH
- Approximations
- Extra

Application

$$
\begin{aligned}
Z[G_{ZOH}(s)H(s)] &= (1-z^{-1})Z\Big[\frac{1}{s^2}-\frac{1}{s}+\frac{1}{s+1}\Big] \\
&= (1-z^{-1})\Big[\frac{zT_e}{(z-1)^2}-\frac{z}{z-1}+\frac{z}{z-e^{-T_e}}\Big] \\
&= \frac{(ze^{-T_e}-z+zT_e)+(1-e^{-T_e}-T_e e^{-T_e})}{(z-1)(z-e^{-T_e})}
\end{aligned}
$$

if $T_e = 1$, we have

$$
\begin{aligned}
Z[G_{ZOH}(s)H(s)] &= \frac{(ze^{-T_e}-z+zT_e)+(1-e^{-T_e}-T_e e^{-T_e})}{(z-1)(z-e^{-T_e})} \\
&= \frac{ze^{-1}+1-2e^{-1}}{(z-1)(z-e^{-1})} \\
&= \frac{b_1 z + b_0}{z^2+a_1 z+a_0}
\end{aligned}
$$

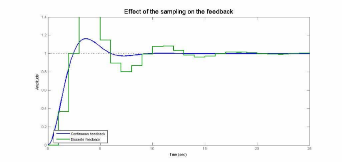
Exercise (cont'd)

Let us return back to sampled-time domain

$$
\begin{aligned}
& & \frac{Y(z)}{U(z)} &= \frac{b_1 z + b_0}{z^2 + a_1 z + a_0} \\
&\Leftrightarrow & Y(z) &= \frac{b_1 z + b_0}{z^2 + a_1 z + a_0} U(z) \\
&\Leftrightarrow & Y(z)(z^2 + a_1 z + a_0) &= (b_1 z + b_0) U(z) \\
&\Leftrightarrow & y(n+2) + a_1 y(n+1) + a_0 y(n) &= b_1 u(n+1) + b_0 u(n)
\end{aligned}
$$

With an unit feedback, the closed loop function is given by :

$$F_{cl}(z) = \frac{G(z)}{1 + G(z)}$$



Similar to the continuous systems. The stability boundary is the unit circle instead of the imaginary axis.

Representation of the discrete linear systems

The discrete output of a system can be expressed as :

$$y(k) = \sum_{n=0}^{\infty} h(k-n)u(n)$$

hence, applying the Z-transform leads to

$$Y(z) = Z[h(k)]U(z) = H(z)U(z)$$

$$H(z) = \frac{b_0 + b_1 z + \cdots + b_m z^m}{a_0 + a_1 z + \cdots + a_n z^n} = \frac{Y}{U}$$

where n ($\geq m$) is the order of the system
The corresponding **difference equation** :

$$\begin{aligned} y(k) &= \frac{1}{a_n}\Big[b_0 u(k-n) + b_1 u(k-n+1) + \cdots + b_m u(k-n+m) \\ &- a_0 y(k-n) - a_2 y(k-n+1) - \cdots - a_{n-1} y(k-1)\Big] \end{aligned}$$

Equivalence $\{p\} \leftrightarrow \{z\}$

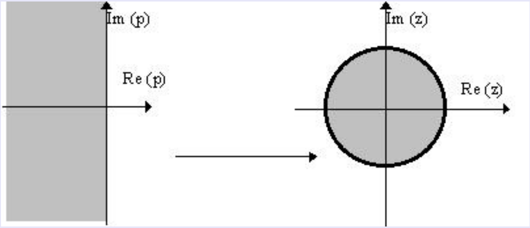
$\{p\} \longrightarrow \{z\}$

The equivalence between the Laplace domain and the Z domain is obtained by the following transformation :

$$z = e^{pT_e}$$

Two poles with a imaginary part which differs of $2\pi/T_e$ give the same pole in Z.

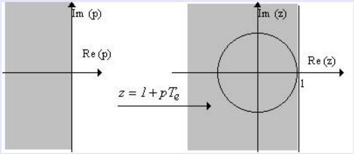
Stability domain



Approximations

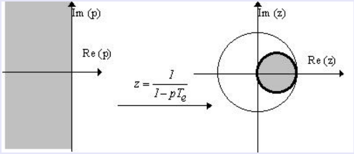
Forward difference (Rectangle inferior)

$$p = \frac{z-1}{T_e}$$



Backward difference (Rectangle superior)

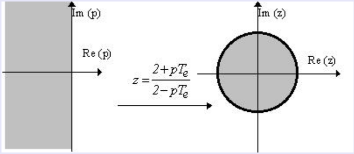
$$p = \frac{z-1}{zT_e}$$

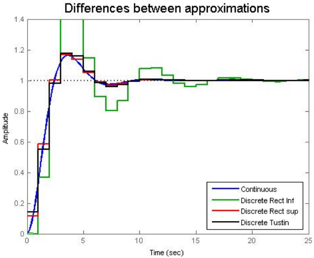


Approximations (cont'd)

Trapezoidal difference (Tustin)

$$p = \frac{2}{T_e}\frac{z-1}{z+1}$$





Stability Remarks

- Forward-difference approximation : it is possible that a stable continuous-time system is mapped into an unstable discrete-time system.
- Backward approximation : a stable continuous-time system will always give a stable discrete-time system.
- Tustin's approximation : has the advantage that the left half s-plane is transformed into the unit disc in the z-plane.

Definition

A **discrete-time** state space system is as follows :

$$\begin{cases} x((k+1)h) = A_d x(kh) + B_d u(kh), \quad x(0) = x_0 \\ y(kh) = C_d x(kh) + D_d u(kh) \end{cases}$$

where h is the sampling period.

`Matlab : ss(`A_d`,`B_d`,`C_d`,`D_d`,`h`) creates a SS object SYS representing a discrete-time state-space model`

Relation with transfer function

For discrete-time systems,

$$\begin{cases} x((k+1)h) = A_d x(kh) + B_d u(kh), \quad x(0) = x_0 \\ y(kh) = C_d x(kh) + D_d u(kh) \end{cases}$$

the discrete transfer function is given by

$$G(z) = C_d (zI_n - A_d)^{-1} B_d + D_d$$

where z is the shift operator, i.e. $zx(kh) = x((k+1)h)$

Recall Laplace & Z-transform

From Transfer Function to State Space

$H(s)$ to state space	$H(z)$ to state space
$\frac{X}{U} = den(s)$ $\frac{Y}{X} = num(s)$	$\frac{X}{U} = den(z)$ $\frac{Y}{X} = num(z)$
$\dot{X} = AX + BU$ $Y = CX + DU$	$X_{k+1} = A_d X_k + B_d U_k$ $Y_k = CX_k + DU_k$
$Y(s) = \underbrace{\left[C[sI - A]^{-1}B + D\right]}_{H(s)} U(s)$	$Y(z) = \underbrace{\left[C_d[zI - A_d]^{-1}B_d + D_d\right]}_{H(z)} U(z)$

Grenoble INP

Solution of state space equations

The state x_k, solution of system $x_{k+1} = A_d x_k$ with initial condition x_0, is given by

$$\begin{aligned} x_1 &= A_d x_0 \\ x_2 &= A_d^2 x_0 \\ x_n &= A_d^n x_0 \end{aligned}$$

The state x_k, solution of system $x_{k+1} = A_d x_k + B_d u_k$, is given by

$$\begin{aligned} x_1 &= A_d x_0 + B_d u_0 \\ x_2 &= A_d^2 x_0 + A_d B_d u_0 + B_d u_1 \\ x_n &= A_d^n x_0 + \sum_{i=0}^{n-1} A_d^{n-1-i} B_d u_i \end{aligned}$$

State space analysis (discrete-time systems)

Stability

A system (state space representation) is stable iff all the eigenvalues of the matrix A_d are inside the unit circle.

Controllability definition

Given two states x_0 and x_1, the system $x_{k+1} = A_d x_k + B_d u_k$ is controllable if there exist $K_1 > 0$ and a sequence of control samples $u_0, u_1, \ldots, u_{K_1}$, such that x_k takes the values x_0 for $k = 0$ and x_1 for $k = K_1$.
The system is **controllable** iff

$$\mathcal{C}_{\lceil (A_d, B_d)} = rg[B_d \quad A_d B_d \ldots A_d^{n-1} B_d] = n$$

State space analysis (2)

Observability definition

The system $x_{k+1} = A_d x_k + B_d u_k$ is said to be completely observable if every initial state $x(0)$ can be determined from the observation of $y(k)$ over a finite number of sampling periods.
The system is **observable** iff

$$\mathcal{O}_{(A_d, C_d)} = rg[C_d \quad C_d A_d \ldots C_d A_d^{n-1}]^T = n$$

Duality

Observability of $(C_d, A_d) \Leftrightarrow$ Controllability of (A_d^T, C_d^T).
(proof...)
Controllability of $(A_d, B_d) \Leftrightarrow$ Observability of (B_d^T, A_d^T).
(proof...)

Application to LEGO robots

Sampling period and Anti-aliasing filter

Sampling period :

- Compute the continuous-time transfer function of the open loop $F_{BO}(s)$
- Estimate the crossover frequency w_c of $F_{BO}(s)$ using the function "margin" in matlab.
- Choose sampling period T_s according to the rule : $w_c T$ is between 0.05 and 0.14.

Anti-aliasing filter :

- Fing the function of the filter for $g_n = 0.1$ and $\xi_f = 0.707$.
- Propose and tune the controller PID (hint use Matlab function "pidtune".
- Discretize the controllers using one of the approximation methods and the chosen sampling period.