[2024-1 Digital Control]
Chapter 4. Open-loop Discrete-time Systems

Gyunghoon Park

School of ECE, University of Seoul

Compute $E^{\mathsf{d}}(z)$ from $E^*(s)$

The starred transform $E^*(s)$ of $e(t)$:

$$E^*(s) = e(0) + e(T)\mathsf{e}^{-Ts} + e(2T)\mathsf{e}^{-2Ts} + \cdots$$

The z-transform $E(z)$ of $e^{\mathsf{d}}(k)$ generated by sampling $e(t)$.:

$$E^{\mathsf{d}}(z) = e^{\mathsf{d}}(0) + z^{-1}e^{\mathsf{d}}(1) + z^{-2}e^{\mathsf{d}}(2) + \cdots$$

with $e^{\mathsf{d}}(k) = e(kT)$.

$\therefore$ They are closely related to each other in the sense that

$$\begin{aligned} E^{\mathsf{d}}(z) &= E^*(s)\big|_{\mathsf{e}^{Ts}=z} \\ &= \sum_{\text{at poles of } E(\lambda)} \left[\text{residues of } E(\lambda)\frac{1}{1 - z^{-1}\mathsf{e}^{T\lambda}}\right] \end{aligned}$$

where the latter follows from the residue theorem.

Example 4.1

Compute the starred transform $E^*(s)$ of

$$E(s) = \frac{1}{(s+1)(s+2)}$$

via the z-transform.

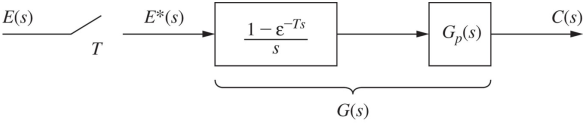
Open-loop sampled-data system

4.3. The Pulse Transfer Function

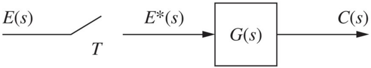
Open-loop sampled-data system = Sampler + Zero-order hold + Plant

Questions:

- Is there any TF $\hat{G}(s)$ satisfying $C(s) = \hat{G}(s)E(s)$? I'm NOT sure...
- How about $C^*(s) = \hat{G}(s)E^*(s)$? We ALWAYS have!



(a)



(b)

- Anyway, what does the open-loop mean?

Useful formula

4.3. The Pulse Transfer Function

Suppose that $A(s) = \mathcal{L}(a(t))$ can be expressed as

$$A(s) = B(s)F^*(s), \quad F^*(s) = f(0) + f(1)\mathrm{e}^{-Ts} + \cdots .$$

Then its starred transform $A^*(s)$ is computed as follows:

$$\begin{aligned}
A^*(s) &= \frac{1}{T}\sum_{n=-\infty}^{\infty} A(s + jn\omega_s) = \frac{1}{T}\sum_{n=-\infty}^{\infty} B(s + jn\omega_s)F^*(s + jn\omega_s) \\
&= \frac{1}{T}\left(\sum_{n=-\infty}^{\infty} B(s + jn\omega_s)\right)F^*(s) \qquad (\because\ F^*(s + jn\omega_s) = F^*(s)) \\
&= B^*(s)F^*(s).
\end{aligned}$$

$\therefore$ We thus have

$$A^*(s) = B^*(s)F^*(s) \quad \Rightarrow \quad A^{\mathsf{d}}(z) = B^{\mathsf{d}}(z)F^{\mathsf{d}}(z)$$

Definition of the "pulse" transfer function

4.3. The Pulse Transfer Function

Return to our sampled-data system whose output $c(t)$ satisfies

$$C(s) = G(s)E^*(s), \quad G(s) = G_p(s)\frac{1 - \mathsf{e}^{-Ts}}{s}.$$

The formula above leads to

$$C^{\mathsf{d}}(z) = G^{\mathsf{d}}(z)E^{\mathsf{d}}(z)$$

where

$$G^{\mathsf{d}}(z) := G^*(s)|_{\mathsf{e}^{Ts}=z} = \mathcal{Z}\left[G_p(s)\frac{1 - \mathsf{e}^{-Ts}}{s}\right]$$

is called the pulse transfer function.

So very roughly speaking, the pulse transfer function $G^{\mathsf{d}}(z)$

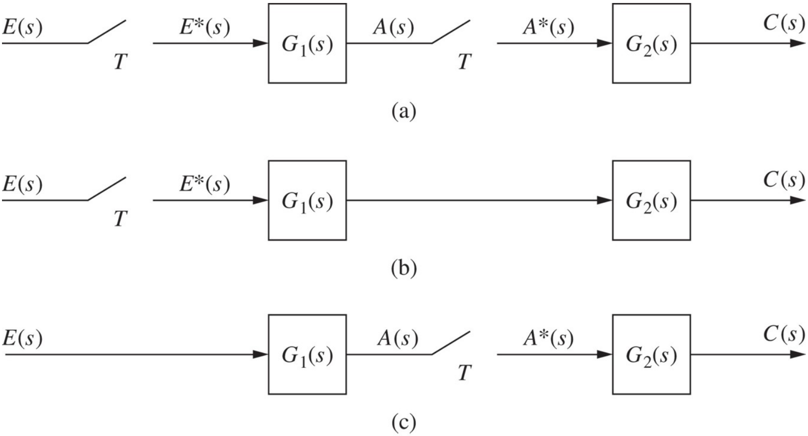
represents dynamical behavior of open-loop sampled-data system in z-domain.

Example 4.3

Compute the pulse transfer function $G^{\mathsf{d}}(z)$ of the sampled-data system with $G_p(s) = \dfrac{1}{s+1}$.

Other types of open-loop sampled-data systems

Note: Not all the systems can be represented as the form $C^*(s) = (\blacksquare)E^*(s)$...



(Cont'd)

4.3. The Pulse Transfer Function

(a) $C^{\mathsf{d}}(z) = G_1^{\mathsf{d}}(z)G_2^{\mathsf{d}}(z)E^{\mathsf{d}}(z)$

$\because A^*(s) = G_1^*(s)E^*(s)$ and $C^*(s) = G_2^*(s)A^*(s)$.

(b) $C^{\mathsf{d}}(z) = \overline{G_1 G_2}^{\mathsf{d}}(z)E^{\mathsf{d}}(z)$

where we use the overline $\overline{(\cdot)}$ to define

$$\overline{G_1 G_2}^{\mathsf{d}}(z) := \mathcal{Z}[G_1(s)G_2(s)] \neq G_1^{\mathsf{d}}(z)G_2^{\mathsf{d}}(z)$$

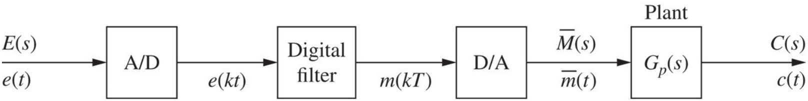
(c) $C^{\mathsf{d}}(z) = G_2^{\mathsf{d}}(z)\overline{G_1 E}^{\mathsf{d}}(z)$

$\because C(s) = G_2(s)A^*(s) = G_2(s)\overline{G_1 E}^*(s)$.

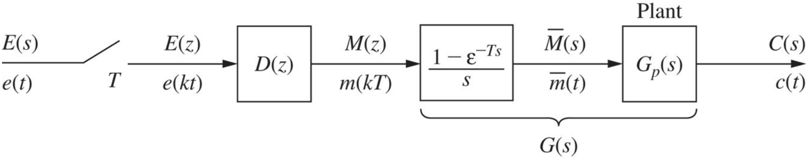
Open-loop system with a digital filter

4.4. Open-loop Systems Containing Digital Filters

Now we consider the following open-loop system



where $M^{\mathsf{d}}(z) = D^{\mathsf{d}}(z)E^{\mathsf{d}}(z)$: The output of the digital filter (or controller).



We claim that $C^{\mathsf{d}}(z) = G^{\mathsf{d}}(z)D^{\mathsf{d}}(z)E^{\mathsf{d}}(z)$.

Proof of claim

By the relation between the z- and starred transforms, one has

$$M^{\mathsf{d}}(z) = D^{\mathsf{d}}(z)E^{\mathsf{d}}(z) \quad \Rightarrow \quad M^*(s) = D^*(s)E^*(s).$$

The output $C(s)$ is then computed in s-domain by

$$\begin{aligned} C(s) &= G_p(s)\frac{1-\mathsf{e}^{-Ts}}{s}M^*(s) = G_p(s)\frac{1-\mathsf{e}^{-Ts}}{s}D^*(s)E^*(s) \\ &= G_p(s)\frac{1-\mathsf{e}^{-Ts}}{s}D^{\mathsf{d}}(z)|_{z=\mathsf{e}^{Ts}}E^*(s) \end{aligned}$$

and thus,

$$C^{\mathsf{d}}(z) = \mathcal{Z}\left[G_p(s)\frac{1-\mathsf{e}^{-Ts}}{s}\right]D^{\mathsf{d}}(z)E^{\mathsf{d}}(z) = G^{\mathsf{d}}(z)D^{\mathsf{d}}(z)E^{\mathsf{d}}(z).$$

Example 4.4

4.4. Open-loop Systems Containing Digital Filters

Compute the output $c^{\mathrm{d}}(k)$ of the sampled-data system with

- The continuous-time plant $G_p(s) = 1/(s+1)$
- Digital filter governed by the difference equation

$$m(kT) = 2e(kT) - e((k-1)T).$$

Discretization with zero-order hold preserves the DC gain.

We now show that

$$\lim_{s\to 0} G_p(s) = \lim_{z\to 1} G^{\mathsf{d}}(z).$$

- The steady-state step response of $G_p(s)$ = DC gain of $G_p(s)$:

$$c_{ss} = \lim_{s\to 0} sG_p(s)\frac{1}{s} = G_p(0) \quad (\because \text{ final-value theorem})$$

- The steady-state step response of $G^{\mathsf{d}}(z)$ = DC gain of $G^{\mathsf{d}}(z)$:

$$c^{\mathsf{d}}_{ss} = \lim_{z\to 1}(z-1)G^{\mathsf{d}}(z)\frac{z}{z-1} = \lim_{z\to 1} G^{\mathsf{d}}(z) = G^{\mathsf{d}}(1).$$

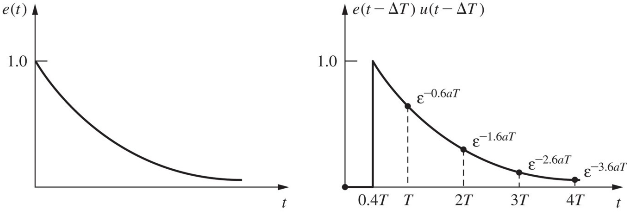
- On the other hand, both c_{ss} and c^{d}_{ss} must be the same (why?).

The z-transform of a delayed signal

4.5. Modified z-Transform

Consider a ΔT-delayed signal of $e(t)$ in continuous time (with $0 < \Delta \leq 1$)

$$e(t - \Delta T)u(t - \Delta T).$$



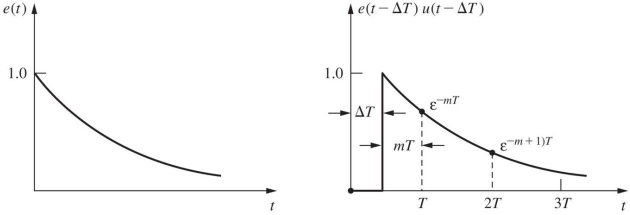
Then, the z-transform of the delayed signal is computed by

$$\begin{aligned} E^{\mathsf{d}}(z, \Delta) &:= \mathcal{Z}[e(t - \Delta T)u(t - \Delta T)] \\ &= \mathcal{Z}[E(s)\mathrm{e}^{-\Delta T s}] = \sum_{n=1}^{\infty} e(nT - \Delta T)z^{-n} \end{aligned}$$

- A generalization of the z-transform to deal with the delay in z-domain
- For each $0 \le m < 1$, it is defined as

$$
\begin{aligned}
E^{\mathsf{d}}_{\mathrm{mod}}(z, m) &:= E^{\mathsf{d}}(z, \Delta)|_{\Delta = 1 - m} \\
&= \mathcal{Z}[E(s)\mathrm{e}^{-\Delta T s}]|_{\Delta = 1-m} \\
&= e(T - \Delta T)z^{-1} + e(2T - \Delta T)z^{-2} + \cdots \Big|_{\Delta = 1-m} \\
&= e(mT)z^{-1} + e((1+m)T)z^{-2} + \cdots
\end{aligned}
$$

where we use the subscript mod to avoid confusion...



(Cont'd)

4.5. Modified z-Transform

We can also represent the modified z-transform $E^{\mathsf{d}}_{\mathrm{mod}}(z,m)$ as

$$\begin{aligned}
E^{\mathsf{d}}_{\mathrm{mod}}(z,m) &= \mathcal{Z}[E(s)\mathrm{e}^{-\Delta T s}]|_{\Delta=1-m} \\
&= \mathcal{Z}[E(s)\mathrm{e}^{-(1-m)Ts}] \\
&= z^{-1}\mathcal{Z}[E(s)\mathrm{e}^{mTs}] \\
&= z^{-1}\left(\sum_{\text{at poles of } E(\lambda)}\left(\text{residues of } E(\lambda)\mathrm{e}^{mT\lambda}\frac{1}{1-z^{-1}\mathrm{e}^{T\lambda}}\right)\right).
\end{aligned}$$

Sometimes we use $\mathcal{Z}_m$ to represent the modified z-transform: that is,

$$\mathcal{Z}_m[E(s)] := E^{\mathsf{d}}_{\mathrm{mod}}(z,m) = \mathcal{Z}[\mathrm{e}^{-\Delta T s}E(s)]|_{\Delta=1-m}.$$

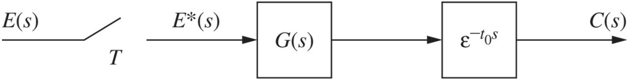
A useful property is, if the signal is $(k+\Delta)T$-delayed, then

$$\mathcal{Z}_m[\mathrm{e}^{-kTs}E(s)] = z^{-k}\mathcal{Z}_m[E(s)] = z^{-k}E^{\mathsf{d}}_{\mathrm{mod}}(z,m)$$

Systems with delay

4.6. Systems with Time Delays

Consider a system with a delayed time $t_0 = k + \Delta$:



Then the output $C(s)$ is represented by

$$C(s) = G(s)\mathrm{e}^{-t_0 s} E^*(s)$$

and thus, in z-domain we have

$$\begin{aligned} C^{\mathsf{d}}(z) &= \mathcal{Z}[G(s)\mathrm{e}^{-t_0 s}]E^{\mathsf{d}}(z) \\ &= z^{-k}\mathcal{Z}[G(s)\mathrm{e}^{-\Delta T s}]E^{\mathsf{d}}(z) \\ &= \big(z^{-k}G^{\mathsf{d}}_{\mathrm{mod}}(z,m)\big)E^{\mathsf{d}}(z). \end{aligned}$$

Exercise: Solve Example 4.8.

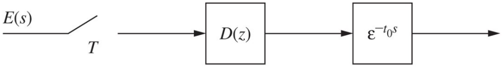
Systems + controller with delay

4.6. Systems with Time Delays

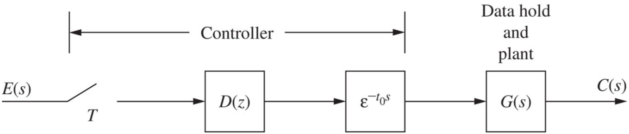
Handling delay is important in many control systems.

Ex. Teleoperation of a robot

- Naver Labs: `https://www.youtube.com/watch?v=JclqXwgx0yE`
- ETH Zürich: `https://www.youtube.com/watch?v=GF9QpFal8fQ`



(a)



(b)

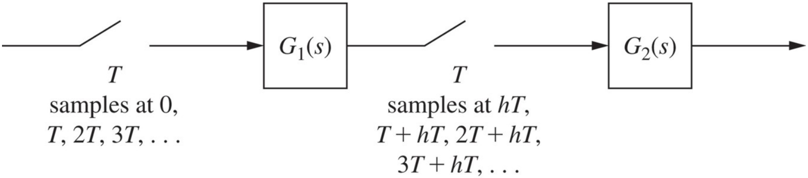
The same procedure brings $C^{\mathsf{d}}(z) = \left(z^{-k} G^{\mathsf{d}}_{\text{mod}}(z,m) D^{\mathsf{d}}\right) E^{\mathsf{d}}(z)$.

Synchronous vs. Nonsynchronous sampling

4.7. Nonsynchronous Sampling

Nonsynchronous sampling (in this class):

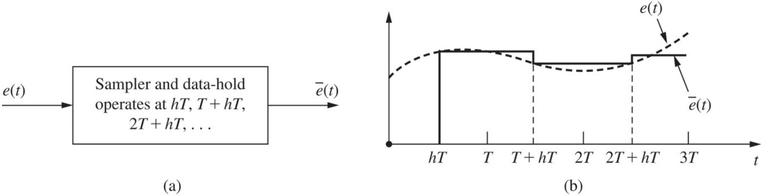
- The sampling rate or period is still T, but
- The sampling instants are not synchronous



This might happens when each device has each (nonsynchronized) clock.

Understanding a nonsynchronous sampler/hold

4.7. Nonsynchronous Sampling



If $0 < h < 1$, then one represents the output $\overline{e}(t)$ as

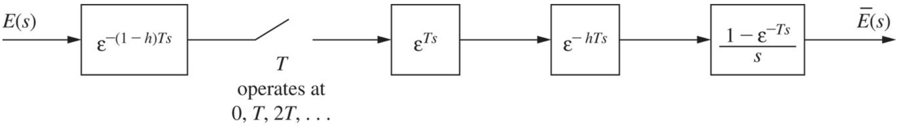
$$\begin{aligned}\overline{e}(t) = e(hT)&[u(t-hT) - u(t-h-hT)] \\ &+ e(T+hT)[u(t-T-hT) - u(t-2T-hT)] + \cdots\end{aligned}$$

which gives

$$\overline{E}(s) = \frac{1-\mathsf{e}^{-Ts}}{s}\mathsf{e}^{Ts}\mathsf{e}^{-hTs}E^{\mathsf{d}}_{\mathrm{mod}}(z,m)|_{m=h,z=\mathsf{e}^{Ts}}$$

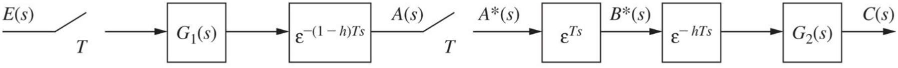
Lesson: Nonsynchronous case = Sync. case + Time-delay

The above computation allows to model nonsynchronous sampler/hold as



which is in fact the synchronous sampler/hold + time delays.

Similarly, the system in page 19 can be represented as



Both samplers operate at 0, T, $2T$, . . .

$$C^{\mathsf{d}}(z) = zG^{\mathsf{d}}_{1,\mathrm{mod}}(z,m)|_{m=h}G^{\mathsf{d}}_{2,\mathrm{mod}}(z,m)|_{m=1-h}E^{\mathsf{d}}(z)$$

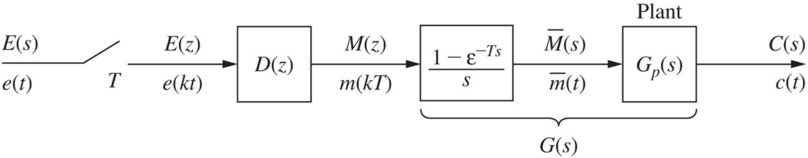
Our interest in the rest of the chapter

4.9. Review of Continuous-time State Variables

We want to find a discrete-time state-space equation

$$\begin{aligned} \mathbf{x}^{\mathsf{d}}(k+1) &= \mathbf{A}^{\mathsf{d}}\mathbf{x}^{\mathsf{d}}(k) + \mathbf{B}^{\mathsf{d}}u^{\mathsf{d}}(k), \\ y^{\mathsf{d}}(k) &= \mathbf{C}^{\mathsf{d}}\mathbf{x}^{\mathsf{d}}(k) + D^{\mathsf{d}}u^{\mathsf{d}}(k) \end{aligned}$$

that represents the sampled-data system from $u^{\mathsf{d}}(k)$ to $y^{\mathsf{d}}(k)$:



Note that, this satisfies

$$\mathbf{C}^{\mathsf{d}}(z\mathbf{I} - \mathbf{A}^{\mathsf{d}})^{-1}\mathbf{B}^{\mathsf{d}} + D^{\mathsf{d}} = P^{\mathsf{d}}(z) = \mathcal{Z}\left[G_p(s)\frac{1 - \mathsf{e}^{-Ts}}{s}\right]$$

State-variable model of a continuous-time system

4.9. Review of Continuous-time State Variables

We begin with the state-space representation of $G_p(s)$:

$$\dot{\mathbf{x}}^{\mathsf{c}}(t) = \mathbf{A}^{\mathsf{c}}\mathbf{x}^{\mathsf{c}}(t) + \mathbf{B}^{\mathsf{c}}u^{\mathsf{c}}(t), \qquad \text{(state equation)}$$

$$y^{\mathsf{c}}(t) = \mathbf{C}^{\mathsf{c}}\mathbf{x}^{\mathsf{c}}(t) + D^{\mathsf{c}}u^{\mathsf{c}}(t). \qquad \text{(output equation)}$$

where

- the superscript c is used to denote the "continuous-time" parts.
- $y^{\mathsf{c}}(t)$ and $y^{\mathsf{d}}(k)$ are related as

$$y^{\mathsf{d}}(k) = y^{\mathsf{c}}(kT), \quad k = 0, 1, 2, \dots$$

- and $u^{\mathsf{c}}(t)$ and $u^{\mathsf{d}}(k)$ are as

$$u^{\mathsf{c}}(t) = u^{\mathsf{d}}(k), \quad kT \le t < (k+1)T$$

(for the zero-order hold case)

Solution of continuous-time state-space equation

For given $t_0 \leq t$,
the solution of the (continuous-time) state equation is computed by

$$\mathbf{x}^{\mathsf{c}}(t) = \mathbf{\Phi}^{\mathsf{c}}(t - t_0)\mathbf{x}^{\mathsf{c}}(t_0) + \int_{t_0}^{t} \mathbf{\Phi}^{\mathsf{c}}(t - \tau)\mathbf{B}^{\mathsf{c}} u^{\mathsf{c}}(\tau)\mathrm{d}\tau.$$

where $\mathbf{\Phi}^{\mathsf{c}}(t)$ is called the continuous-time state transition matrix

$$\mathbf{\Phi}^{\mathsf{c}}(t) = \mathrm{e}^{\mathbf{A}^{\mathsf{c}} t} = \mathbf{I} + \mathbf{A}^{\mathsf{c}} t + \frac{1}{2!}\mathbf{A}^{\mathsf{c}} t^2 + \cdots \in \mathbb{R}^{n \times n}$$

Note that

- $\frac{d}{dt}\mathbf{\Phi}^{\mathsf{c}}(t) = \mathbf{A}^{\mathsf{c}}\mathbf{\Phi}^{\mathsf{c}}(t)$ and
- $G_p(s) = \mathbf{C}^{\mathsf{c}}(sI - \mathbf{A}^{\mathsf{c}})^{-1}\mathbf{B}^{\mathsf{c}} + D^{\mathsf{c}}$

DT state equations of sampled-data system (with ZOH)

Define the state variable of the discrete-time state-variable model:

$$\mathbf{x}^{\mathsf{d}}(k) := \mathbf{x}^{\mathsf{c}}(kT)$$

We then have the discrete-time state equation,
by replacing t and t_0 with $(k+1)T$ and kT, as follows:

$$\begin{aligned}
\mathbf{x}^{\mathsf{d}}(k+1) &= \mathbf{x}^{\mathsf{c}}((k+1)T) \\
&= \mathbf{\Phi}^{\mathsf{c}}(T)\mathbf{x}^{\mathsf{c}}(kT) + \int_{kT}^{(k+1)T} \mathbf{\Phi}^{\mathsf{c}}((k+1)T-\tau)\mathbf{B}^{\mathsf{c}}u^{\mathsf{c}}(\tau)\mathrm{d}\tau \\
&= \cdots \\
&= \mathbf{\Phi}^{\mathsf{c}}(T)\mathbf{x}^{\mathsf{d}}(k) + \left(\int_{0}^{T} \mathbf{\Phi}^{\mathsf{c}}(\sigma)\mathbf{B}^{\mathsf{c}}\mathrm{d}\sigma\right) u^{\mathsf{d}}(k) \\
&=: \mathbf{A}^{\mathsf{d}}\mathbf{x}^{\mathsf{d}}(k) + \mathbf{B}^{\mathsf{d}}u^{\mathsf{d}}(k)
\end{aligned}$$

(Cont'd)

4.10. Discrete-time State Equations

In summary, a sampled-data system associated with

- continuous-time plant whose state-space representation is

$$\dot{\mathbf{x}}^{\mathsf{c}}(t) = \mathbf{A}^{\mathsf{c}}\mathbf{x}^{\mathsf{c}}(t) + \mathbf{B}^{\mathsf{c}}u^{\mathsf{c}}(t), \quad y^{\mathsf{c}}(t) = \mathbf{C}^{\mathsf{c}}\mathbf{x}^{\mathsf{c}}(t) + D^{\mathsf{c}}u^{\mathsf{c}}(t)$$

- sampler
- zero-order hold

can be expressed in the state space as

$$\begin{aligned} \mathbf{x}^{\mathsf{d}}(k+1) &= \mathbf{A}^{\mathsf{d}}\mathbf{x}^{\mathsf{d}}(k) + \mathbf{B}^{\mathsf{d}}u^{\mathsf{d}}(k), \\ y^{\mathsf{d}}(k) &= \mathbf{C}^{\mathsf{d}}\mathbf{x}^{\mathsf{d}}(k) + D^{\mathsf{d}}u^{\mathsf{d}}(k) = \mathbf{C}^{\mathsf{c}}\mathbf{x}^{\mathsf{d}}(k) + D^{\mathsf{c}}u^{\mathsf{d}}(k) \end{aligned}$$

where the matrices are defined as the above.

Further remarks

4.10. Discrete-time State Equations

- The faster the sampling/holding is, the smaller T is.
- As $T \to 0$,

$$\mathbf{A}^{\mathsf{d}}(T) \to \mathbf{I}, \quad \mathbf{B}^{\mathsf{d}}(T) \to \mathbf{0}$$

  (Physical meaning of this phenomenon?)
- The above equation is valid only for ZOH case.
  If you use another type of hold, then $\mathbf{B}^{\mathsf{d}}$ needs to be modified.
- $\mathbf{\Phi}^{\mathsf{c}}(t)$ can be computed numerically, or use

$$\mathbf{\Phi}^{\mathsf{c}}(t) = \mathcal{L}^{-1}[(s\mathbf{I} - \mathbf{A}^{\mathsf{c}})]^{-1}$$