Chapter 4: Discretization of continuous-time controllers

ECE 481 - Digital Control Systems

Yash Vardhan Pant

Based on notes by Prof. Chris Nielsen

Outline

[X] Introduction

[ ] Ideal sample and zero order hold
    [ ] Preserving linearity

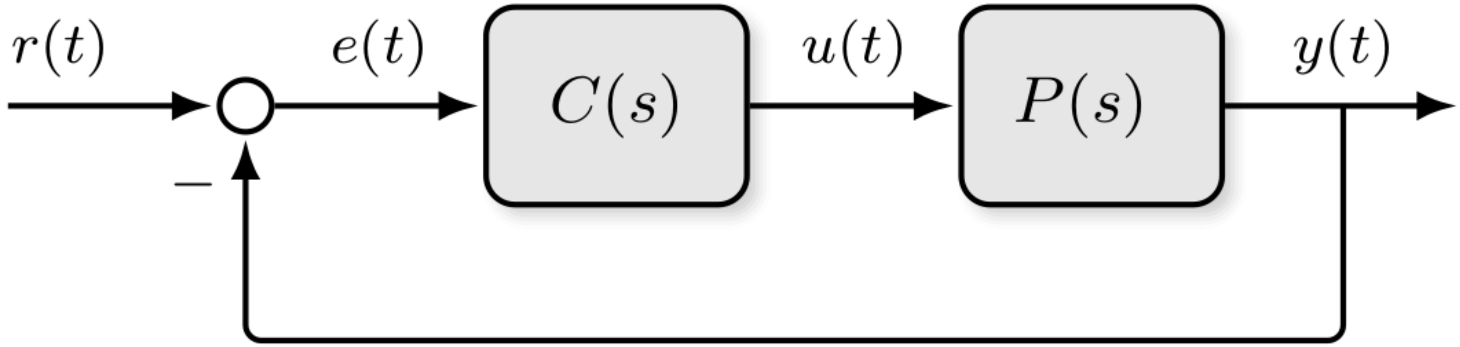
[ ] Discrete approximations
    [ ] Forward and backward Euler (left and right side rules)
    [ ] Trapezoidal approximation
    [ ] Exact (step-invariant) approximation

[ ] Controller design via approximation continuous-time controllers

X = The upcoming topic
- = Topic that has been covered

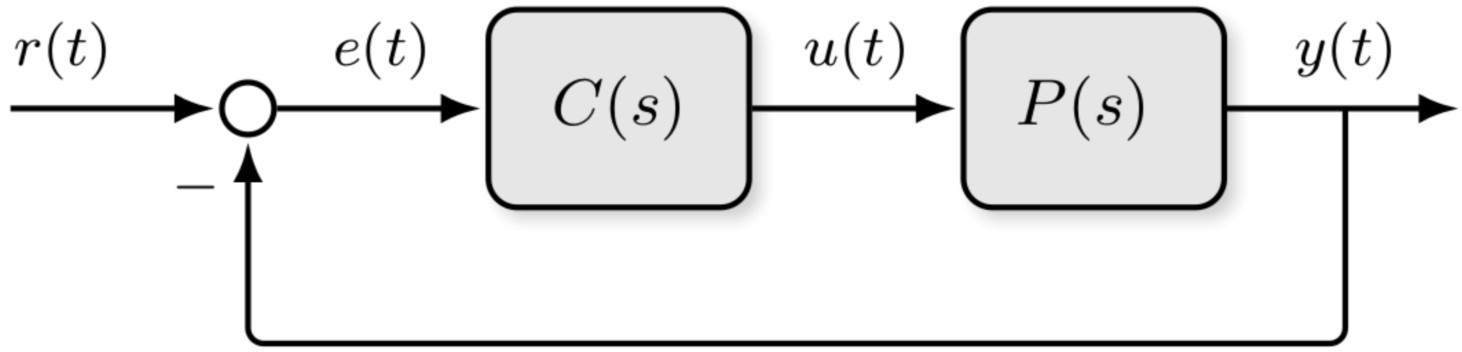
A first look at discretization of continuous-time controllers

Let us start with a continuous controller C(s) that works well for the plant P(s) , e.g., via pole-placement.

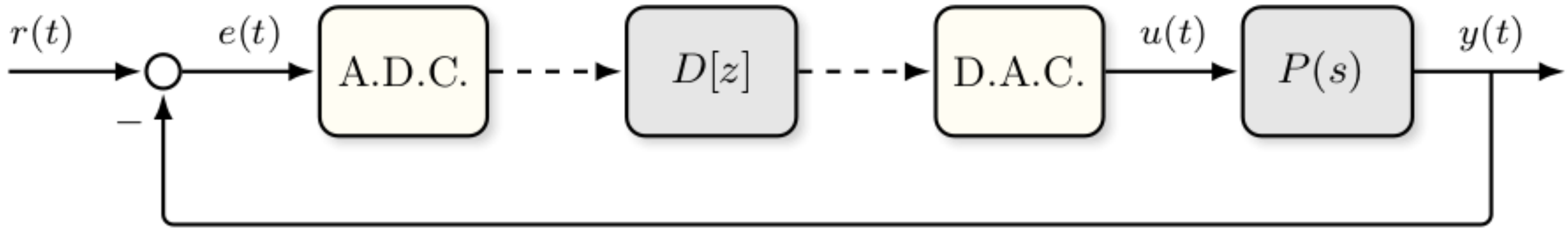


A first look at discretization of continuous-time controllers

Let us start with a continuous controller C(s) that works well for the plant P(s)



We want to design a discrete-time controller D(z) so that the system above is well-approximated.



Outline

[-] Introduction

[X] Ideal sample and zero order hold

  [ ] Preserving linearity

[ ] Discrete approximations

  [ ] Forward and backward Euler (left and right rule)

  [ ] Trapezoidal approximation

  [ ] Exact (step-invariant) approximation

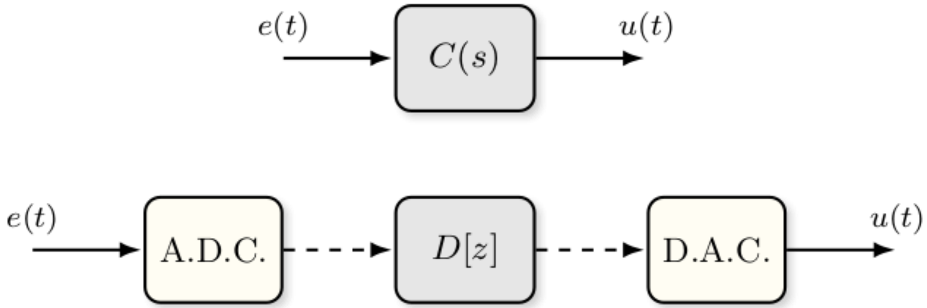
[ ] Controller design via approximation continuous-time controllers

X = The upcoming topic

- = Topic that has been covered

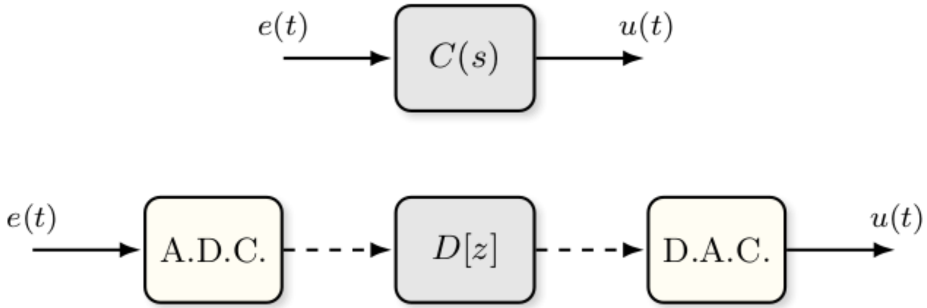
Ideal sample and zero-order hold

We want to approximate the continuous-time control law



Ideal sample and zero-order hold

We want to approximate the continuous-time control law



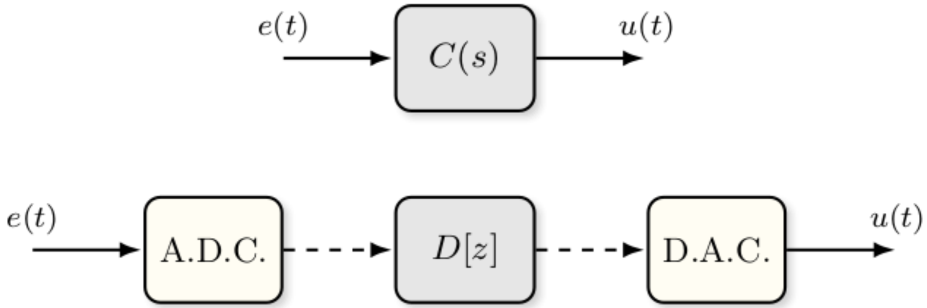
Approximation method:

- Model the A.D.C as an ideal sampler, no noise, no quantization.

- Model the D.A.C using zero-order hold

Ideal sample and zero-order hold

We want to approximate the continuous-time control law



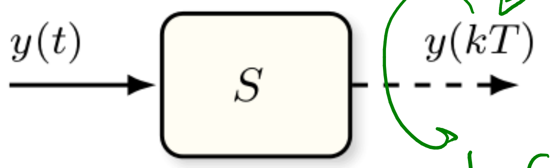
Approximation method:

- Model the A.D.C as an ideal sampler

- Model the D.A.C using zero-order hold

Ideal sampler

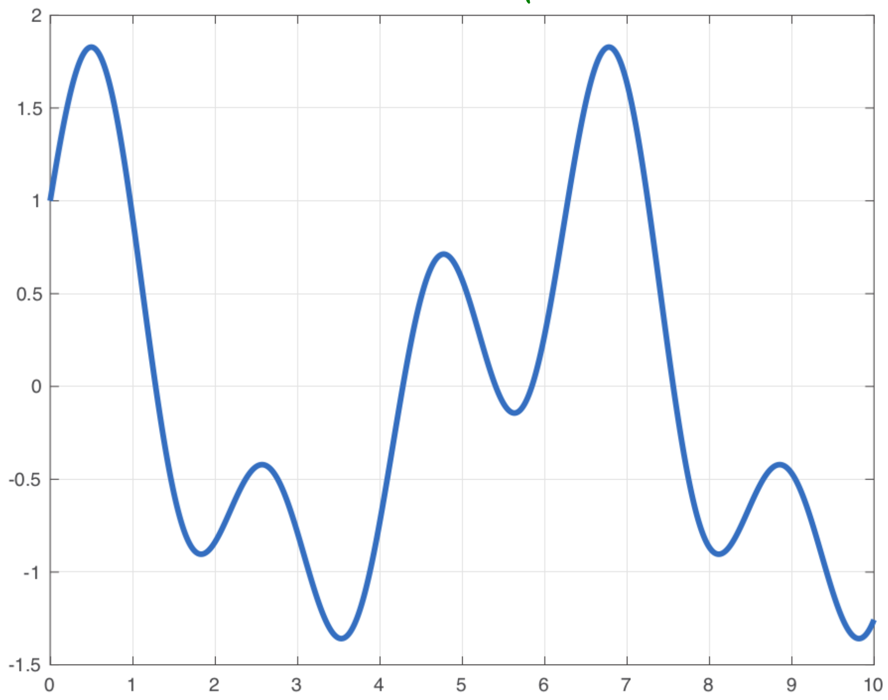
Get a discrete-time representation of a continuous time signal



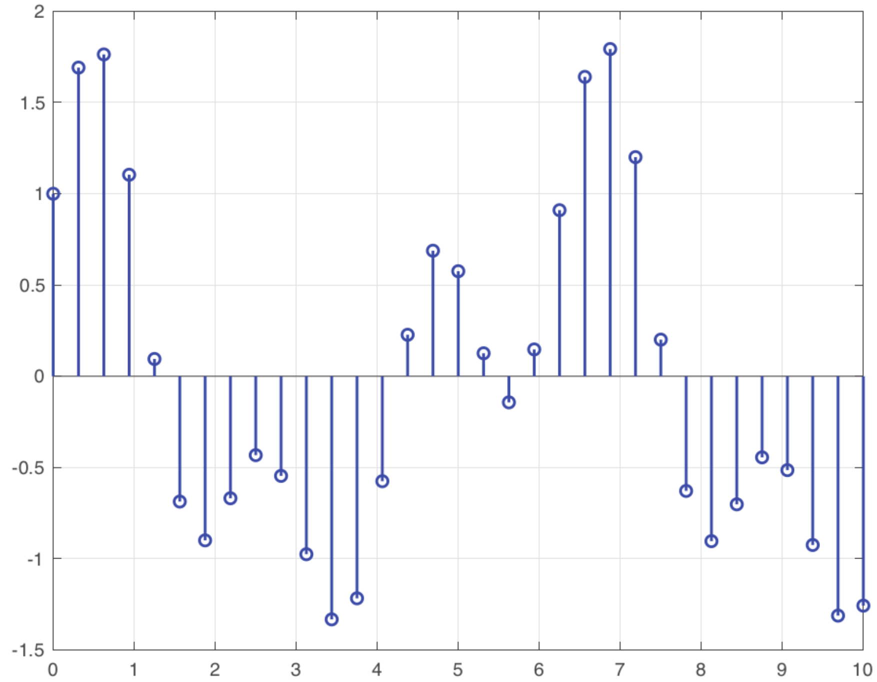
Sampling time (T)

$k \in \mathbb{Z}_{\geq 0}$ → Instant.

CT



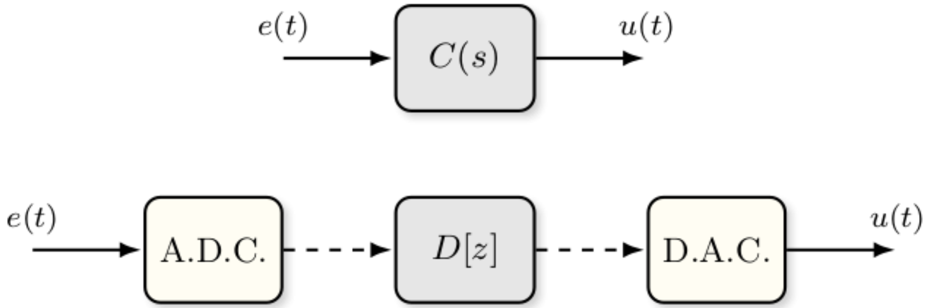
y(t): Input to S



y[k]: Output of S

Ideal sample and zero-order hold

We want to approximate the continuous-time control law



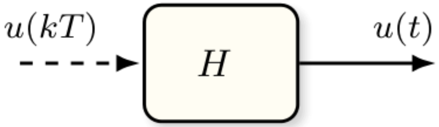
Approximation method:

- Model the A.D.C as an ideal sampler

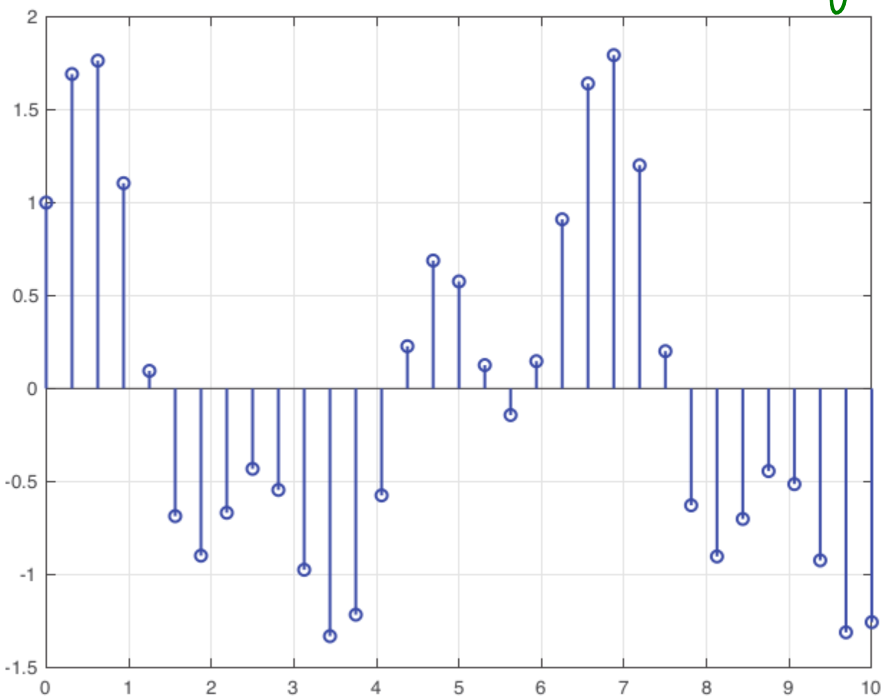
- Model the D.A.C using zero-order hold

Zero-order hold (DAC)

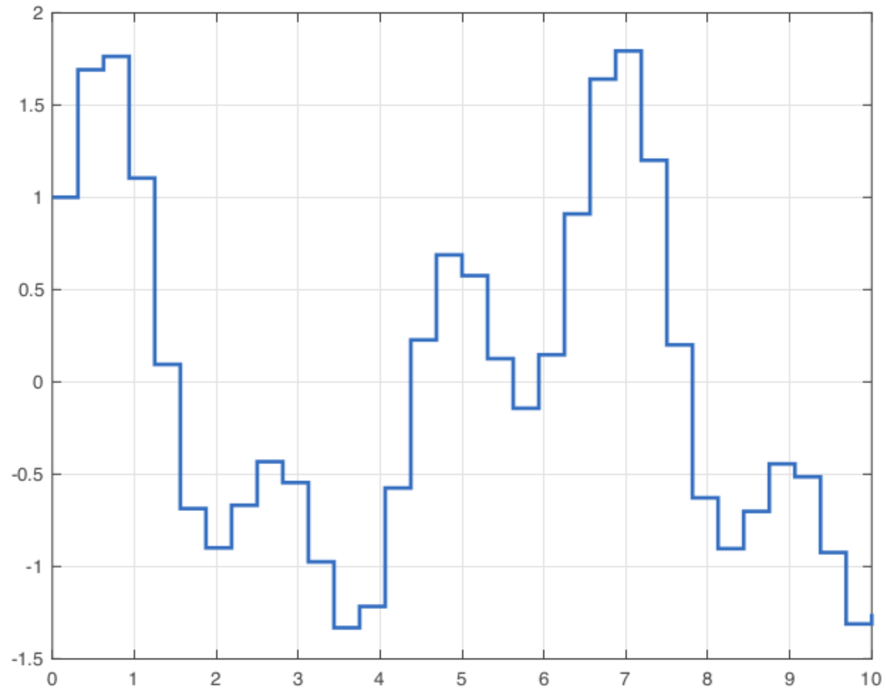
Get a continuous-time signal from a discrete-time signal



CONTROL INPUT (by micro-controller)



u[k]: Input to the ZOH



u(t): Output of the ZOH

Represent u(t) as a function of u[k]:

$$u(t) = H(u(kT))$$
$$= u[k], \ \forall t \in [kT, (k+1)T)$$

Outline

[-] Introduction

[-] Ideal sample and zero order hold

    [X] Preserving linearity

[ ] Discrete approximations

    [ ] Forward and backward Euler (left and right rule)

    [ ] Trapezoidal approximation

    [ ] Exact (step-invariant) approximation

[ ] Controller design via approximation continuous-time controllers

X = The upcoming topic

- = Topic that has been covered

Preserving linearity with ideal sampling and ZOH

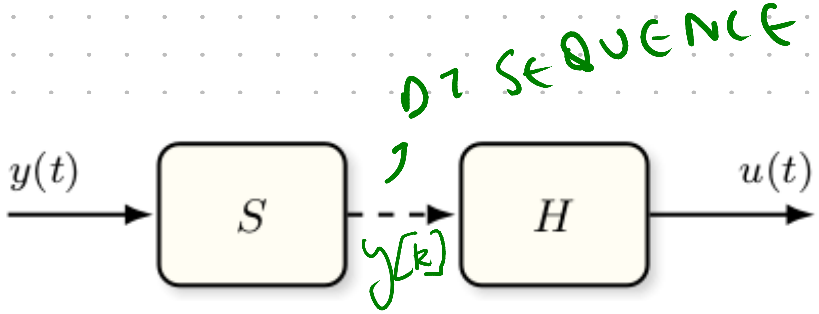
Exercise: Prove that the ideal sampler S, and zero-order hold H are both linear systems.

As a consequence, the compositions sample-and-hold and hold-and-sample are both linear systems (ignoring quantization).

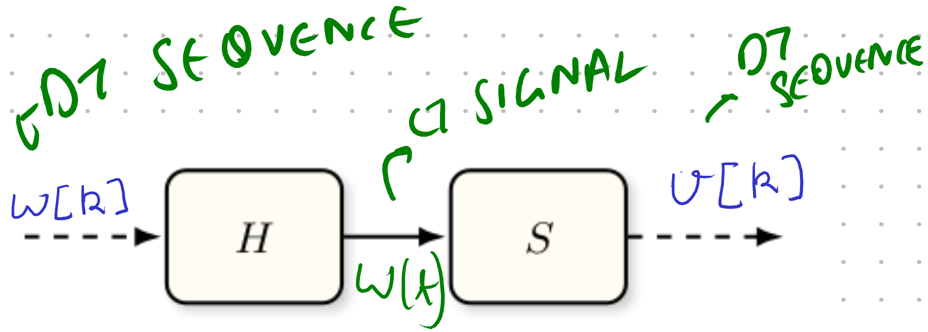
Preserving linearity with ideal sampling and ZOH

Exercise: Prove that the ideal sampler S, and zero-order hold H are both linear systems.

As a consequence, the compositions sample-and-hold and hold-and-sample are both linear systems (ignoring quantization).



Sample-then-hold: H o S



Hold-then-sample: S o H

$$u(t) = \underbrace{H(S(y(t)))}_{H \circ S}$$

COMPOSITION

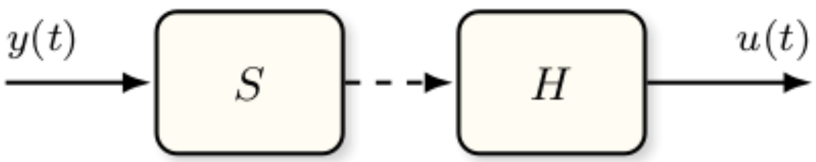
$$u(t) = y(t) \ \forall t \, ?$$

$$v[k] = \underbrace{S(H(w[k]))}_{S \circ H}$$

Preserving linearity with ideal sampling and ZOH

Exercise: Prove that the ideal sampler S, and zero-order hold H are both linear systems.

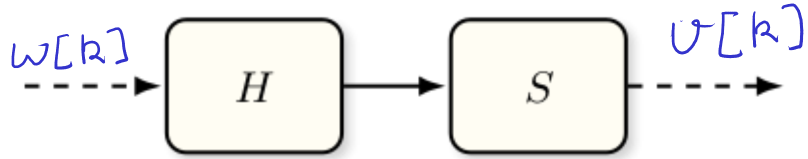
As a consequence, the compositions sample-and-hold and hold-and-sample are both linear systems (ignoring quantization).



Sample-then-hold: H o S

$$u(t) = \underbrace{H(S(y(t)))}_{H \circ S}$$

composition



Hold-then-sample: S o H

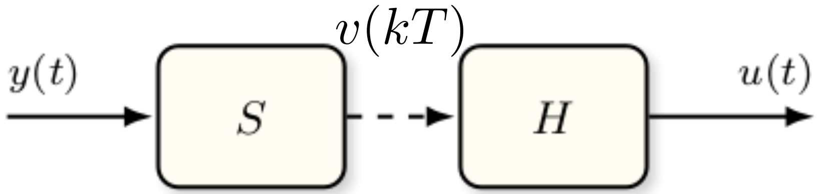
$$v[k] = \underbrace{S(H(w[k]))}_{S \circ H}$$

H o S: Not a Linear Time-Invariant System!
(No transfer function representation, see Section 4.2 in notes)

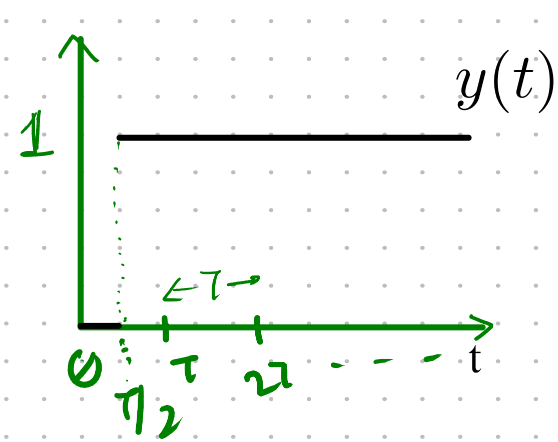
Sample-then-hold ($H \circ S$) is not LTI! An example:

$T \rightarrow$ Sampling time

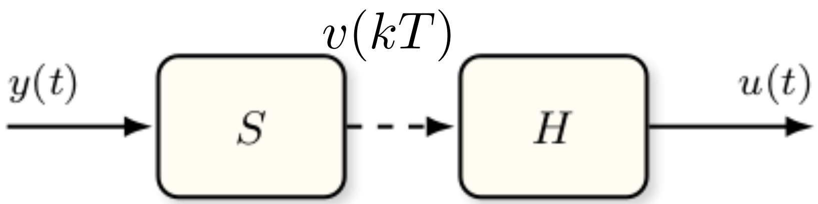
Unit step: $1(t)$



Let $y(t) = 1(t-T/2)$ UNIT STEP, DELAYED BY $T/2$

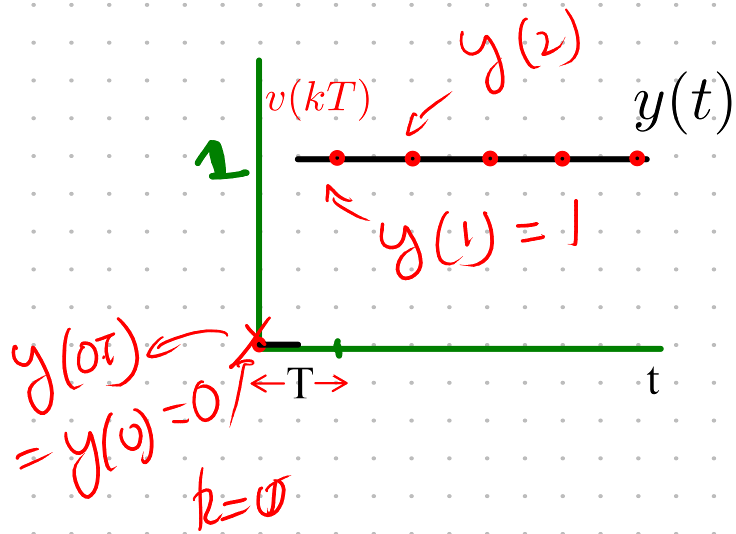


Sample-then-hold (H o S) is not LTI! An example:

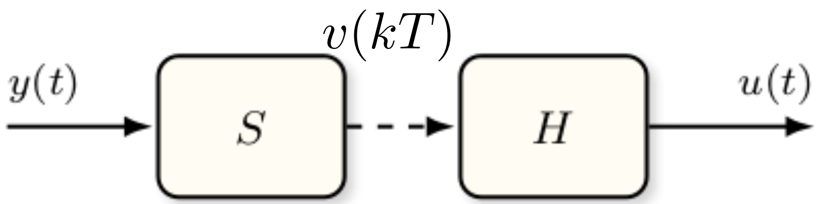


Let y(t) = 1(t−T/2)

$v(kT) = S(y(t))$



Sample-then-hold (H o S) is not LTI! An example:

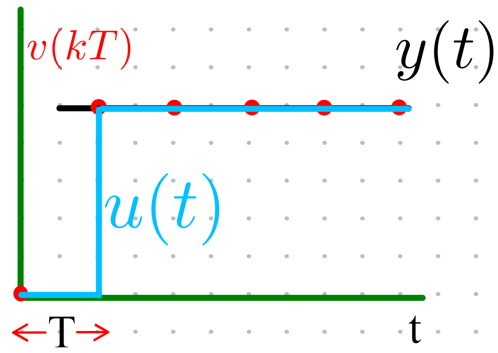


Let y(t) = 1(t-T/2)

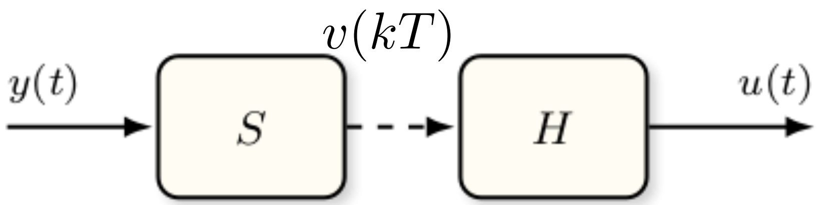
Output u(t) = 1(t-T)

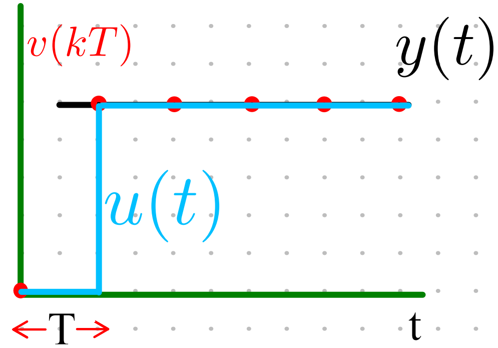
$\{ u(t) = H(v(kT)) \}$

i.e., the output is the unit-step delayed by one sampling period (T).



Sample-then-hold (H o S) is not LTI! An example:





Let y(t) = 1(t-T/2)

Output u(t) = 1(t-T)

i.e., the output is the unit-step delayed by one sampling period (T).

Now consider y(t) = 1(t-T), then u(t) = 1(t-T) <verify>

Therefore H o S is linear, but not time-invariant! No transfer function for this operation.

Preserving linearity with ideal sampling and ZOH

Exercise: Prove that the ideal sampler S, and zero-order hold H are both linear systems.

As a consequence, the compositions sample-and-hold and hold-and-sample are both linear systems (ignoring quantization).

$w(t) = H(w[k])$

$$\begin{tikzcd} {} \arrow[r, "y(t)"] & \boxed{S} \arrow[r, dashed] & \boxed{H} \arrow[r, "u(t)"] & {} \end{tikzcd}$$

Sample-then-hold: H o S

$$\begin{tikzcd} {} \arrow[r, dashed, "{w[k]}"] & \boxed{H} \arrow[r] & \boxed{\begin{matrix} S \\ (T) \end{matrix}} \arrow[r, dashed, "{v[k]}"] & {} \end{tikzcd}$$

Hold-then-sample: S o H

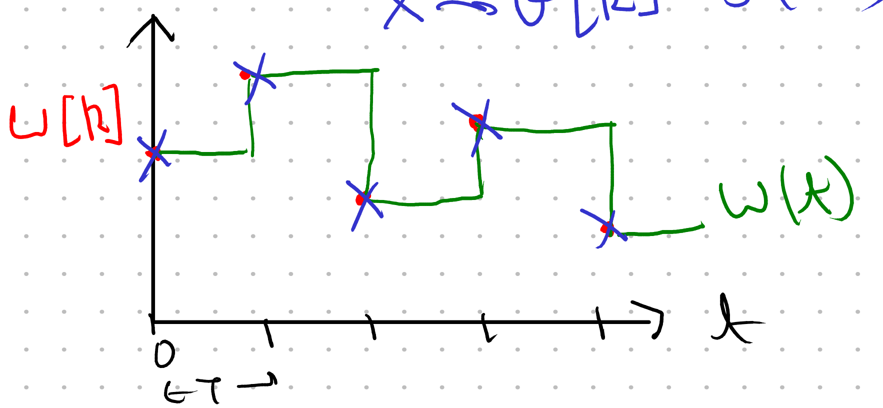
$$u(t) = \underbrace{H(S(y(t)))}_{H \circ S}$$

composition

$$v[k] = \underbrace{S(H(w[k]))}_{S \circ H}$$

$= ?\ w[k]$

$\times \rightsquigarrow v[k] = S(w(t))$



What is S o H ?

Identity system!

$v[k] = w[k] \quad \forall k$

Outline

[-] Introduction

[-] Ideal sample and zero order hold

    [-] Preserving linearity

[X] Discrete approximations

    [ ] Forward and backward Euler (left and right rule)

    [ ] Trapezoidal approximation

    [ ] Exact (step-invariant) approximation

[ ] Controller design via approximation of continuous-time controllers

X = The upcoming topic

- = Topic that has been covered

Discrete approximations

- Start with the assumption that the CT controller $C(s)$ achieves the desired specifications.

- We want to construct a DT controller $D[z]$ that preserves (or tries to) certain properties of $C(s)$.

- In this course, we will study two common types of discretization methods:

    1. Approximation using numerical integration (or differentiation), e.g., Euler's methods.

    2. Construct $D[z]$ such that it matches response of $C(s)$ at sample instants for certain inputs, e.g., impulse-invariant and step-invariant discretization methods.

Outline

[-] Introduction

[-] Ideal sample and zero order hold

    [-] Preserving linearity

[X] Discrete approximations

    [X] Forward and backward Euler (left and right rule)

    [ ] Trapezoidal approximation

    [ ] Exact (step-invariant) approximation

[ ] Controller design via approximation continuous-time controllers

X = The upcoming topic

- = Topic that has been covered

Numerical integration

$$e \rightarrow \boxed{C} \rightarrow u$$

(INTEGRATOR)

Approximating continuous time integration (differentation) for discrete-time signals:

Let $C(s) = 1/s$, or $U(s) = \frac{1}{s} E$, $sU = E$

Inv LT: $\dot{u}(t) = e(t)$, or $u(t) - u(t_0) = \int_{t_0}^{t} e(\tau)\,d\tau$ — (1)

Sample $u(t)$ at rate $1/T$ Hz, then the difference between two successive samples of $u(t)$ is:

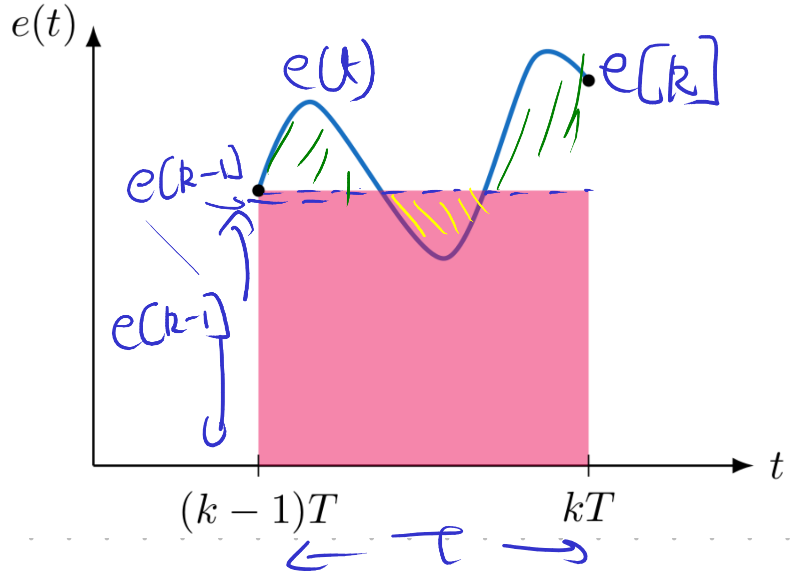
$$u(kT) - u((k-1)T) = \int_{(k-1)T}^{kT} e(\tau)\,d\tau$$

$\therefore$ at sampling instants:

$$u[k] = u[k-1] + \left(\text{Area under } e(t) \text{ over } t \in [(k-1)T, kT)\right).$$

→ Approximate this using samples of $e(t)$.
[Numerical integration].

Euler's approximation (forward), or the Left-side rule



LSR: Approximate by assuming $e(t)$ is constant in $[(k-1)T, kT]$, with value $e((k-1)T)$

$$\int_{(k-1)T}^{kT} e(\tau)\,d\tau \simeq T e((k-1)T)$$

$$\Rightarrow u(kT) = u((k-1)T) + T e((k-1)T)$$

$$u[k] = u[k-1] + T e[k-1]$$

$$\boxed{D_{LSR}[z] = C\left(\frac{z-1}{T}\right) \text{ (Summary)}}$$

Take Z-TF (w/ 0-initial conditions)

$$\Rightarrow U[z] = z^{-1} U[z] + T z^{-1} e[z]$$

$$\Rightarrow \frac{U[z]}{E[z]} = \frac{T}{z-1}$$

$$\left(\text{CT TF: } \frac{U}{E} = \frac{1}{s} \right)$$

$$\Rightarrow s \simeq \frac{z-1}{T} \quad \text{(LSR)}$$

C2D for state-space models using forward Euler (or LSR)

$u \rightarrow \boxed{P} \rightarrow Y$

$e \rightarrow \boxed{C} \rightarrow u$

$$\dot{x}_c = A_c x_c + B_c e \qquad x \in \mathbb{R}^{n}, \; u \in \mathbb{R}^{m},$$
$$u = C_c x_c + D_c e \qquad e \in \mathbb{R}^{\ell}$$

LT: $sX_c = AX_c + B_c E$, Let's apply LSR to this
$$U = C_c X_c + D_c E$$

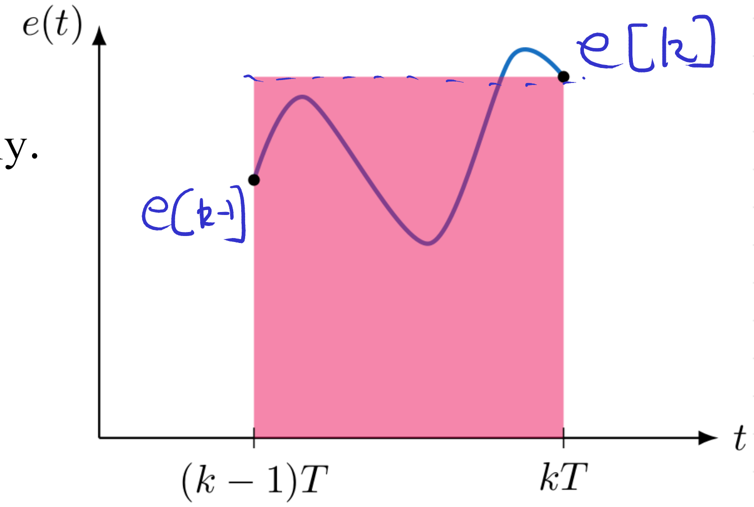
$$\Rightarrow \left(\frac{z-1}{T}\right) X_c = AX_c + B_c E, \quad U = C_c X_c + D_c E$$

Inv Z TF: $x_c[k+1] = I \underline{x_c}[k] + TA_c \underline{x}_c[k] + TB_c e[k]$
$$u[k] = C_c x_c[k] + D_c e[k]$$

$$\therefore \; (A_c, B_c, C_c, D_c) \xrightarrow[LSR]{} (I + TA_c, TB_c, C_c, D_c)$$

Euler's approximation (backward), or the right-side rule

We studied the corresponding approximation of time derivative previously.



Assume $e(t)$ is constant between $[(k-1)T, kT]$ with value $e(kT) = e[k]$

$\frac{U}{E} = \frac{1}{s}$, $u[k] = u[k-1] + \int \cdots$

$u[k] = u[k-1] + T e[k]$ ← Take Z-TF

$u[k]$

RSR!

Z-TF $\Rightarrow \frac{U[z]}{E[z]} = \frac{Tz}{z-1}$, approx for $\frac{U}{E} = \frac{1}{s}$

$$s \simeq \frac{z-1}{Tz}$$

$$\boxed{D_{RSR}[z] = C\left(\frac{z-1}{Tz}\right)}$$

Example: PID from CT to DT (Forward Euler)

PID controller: $U(s) = K_pE(s) + sK_dE(s) + (1/s)K_iE(s)$

$$u(t) = K_p e(t) + \frac{d}{dt} e_t \cdot K_D + K_I \int_0^t e(\tau)\,d\tau$$

LSR, $s \simeq \dfrac{z-1}{T}$

$$U[z] = K_p E[z] + K_d\left(\frac{z-1}{T}\right)E[z] + K_I\left(\frac{T}{z-1}\right)E[z]$$

$$T(z-1)U = \left(T(z-1)K_p + K_d(z-1)^2 + K_I T^2\right)E$$

$$= E\left[K_d z^2 - \underbrace{(2K_D + TK_p)}_{\gamma} z + \underbrace{(K_D + K_I T^2 - TK_p)}_{c}\right]$$

(You can get DT TF)

$$z^{-1} \Rightarrow Tu[k+1] - Tu[k] = K_d e[k+2] - \gamma e[k+1] + c e[k]$$

$$\Rightarrow u[k+1] = c_1 u[k] + c_2 e[k+2] + c_3 e[k+1] + c_4 e[k]$$

$e \to \boxed{C} \to u$

(Non-causal Difference eqn)

Example: PID from CT to DT (Backward Euler)

$$s \leftarrow \frac{z-1}{Tz}$$

PID controller: $U(s) = K_p E(s) + sK_d E(s) + (1/s)K_i E(s)$

$$U[z] = K_p E[z] + K_d\left(\frac{z-1}{Tz}\right)E[z] + \frac{K_i T z}{z-1}E[z]$$

$$= \left(\underbrace{K_p T}_{C_1} z(z-1) + \underbrace{K_d}_{C_2}(z-1)^2 + \underbrace{T^2 K_i}_{C_3} z^2\right)\frac{E[z]}{Tz(z-1)}$$

Collect terms

$$\Rightarrow \Big[z^2\big(\overbrace{C_1 + K_D + C_3}^{\gamma_1}\big) + z\big(\overbrace{-C_1 - 2C_2}^{\gamma_2}\big) + C_2\Big]E$$

$$= Tz^2 U - TzU$$

Inv Z-TF:

$$U[k+2] = \frac{1}{T}\Big(TU[k+1] - \gamma_1 e[k+2] - \gamma_2 e[k+1] - C_2 e[k]\Big)$$

LTI system, so shift

$$\Rightarrow u[k] = \frac{1}{T}\Big(Tu[k-1] - \gamma_1 e[k] - \gamma_2 e[k-1] - C_2 e[k-2]\Big)$$

Causal!

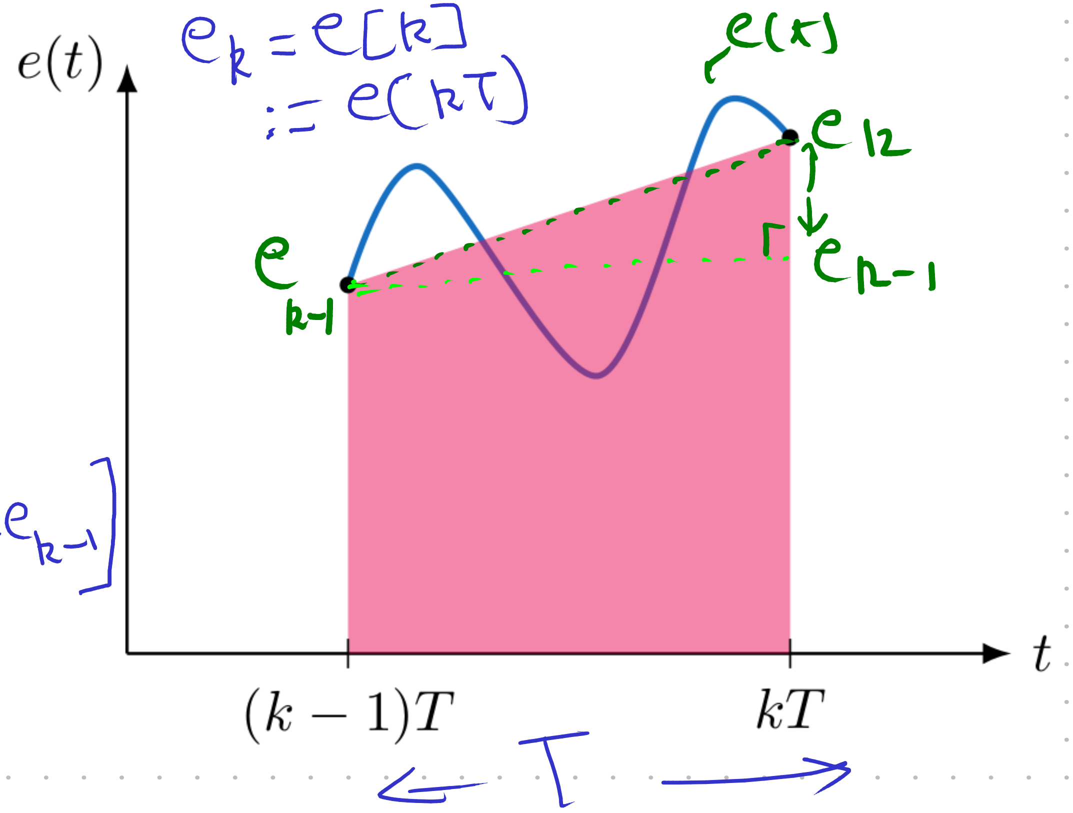
Tustin's (or Trapezoidal) approximation

Assume $e(t)$ is a line with slope $= (e_k - e_{k-1})/T$

Consider the area in the trapezoid:

$$\int_{(k-1)T}^{kT} e(\tau)\,d\tau \simeq T e_{k-1} + \frac{T}{2}[e_k - e_{k-1}]$$

$$= \frac{T}{2}(e_k + e_{k-1})$$



$C(s) = 1/s$, the approx makes it:

$$u[k] = u[k-1] + \frac{T}{2}(e_k + e_{k-1})$$

(approx. for $1/s$)

Z-TF: $\frac{U}{E} = \frac{T}{2}\frac{z+1}{z-1}$

$\boxed{s \simeq \frac{2}{T}\frac{z-1}{z+1}}$ $\Rightarrow$ $D_{TUSTIN}[z] = C\left(\frac{2}{T}\frac{z-1}{z+1}\right)$

Numerical integration: some notes

1. As T-> 0, all 3 approximations become perfect (i.e., DT system's behavior approaches the CT system's).

2. The CT controller C(s) is stable IFF its approximation D[z] is stable (preservation of stability).
Only Tustin's approximation preserves stability.

Note, this does not imply closed-loop system stability just yet.

Quick note: Stability of DT systems

A DT (LTI) system is BIBO stable IFF every pole of the G[z] has a magnitude less than one.

Consider, $\dfrac{Y[z]}{U[z]} = \dfrac{z}{z-a}$ Pole is at $z=a$

Corresponding difference equation: $\underset{Z\text{-}TF}{\text{Inv.}} \left\{ zY[z] - aY[z] = zU[z] \right\}$

$\Rightarrow y[k+1] = ay[k] + u[k+1]$ $u[k] = 0 \ \forall k = 0.$

That $u[k]$ is bounded. For ease,

Say, $y[0] = y_0 \neq 0$.

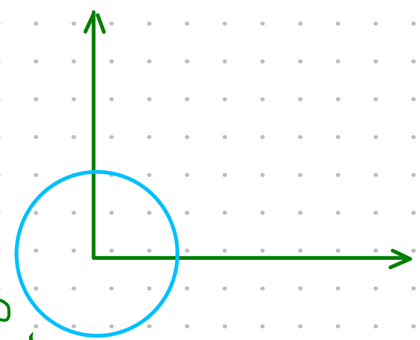
Then, $y[k+1] = ay[k]$ has the solution: $y[n] = a^n y_0$

$y_1 = a y_0, \ y_2 = a y_1 = a^2 y_0, \ \ldots$

What happens to $y[n]$ as $n \to \infty$?

<u>Case 1:</u> $|a| > 1$, $y[n]$ doesn't converge.

<u>Case 2:</u> $|a| < 1$, $y[n]$ converges to 0 as $n \to \infty$.



Tustin's approximation: Preserving stability

Let λ be a pole of the controller $C(s)$, what is the DT pole?

$$C(s) = \frac{1}{s-\lambda} \Rightarrow D_{TUSTIN}[z] = C\left(\frac{2}{T}\frac{z-1}{z+1}\right)$$

$$\Rightarrow D[z] = \frac{1}{\frac{2}{T}\left(\frac{z-1}{z+1}\right) - \lambda}$$

$$= \frac{T(z+1)}{2z - 2 - Tz\lambda - T\lambda} = \frac{T(z+1)}{z(2-T\lambda) - (2+T\lambda)}$$

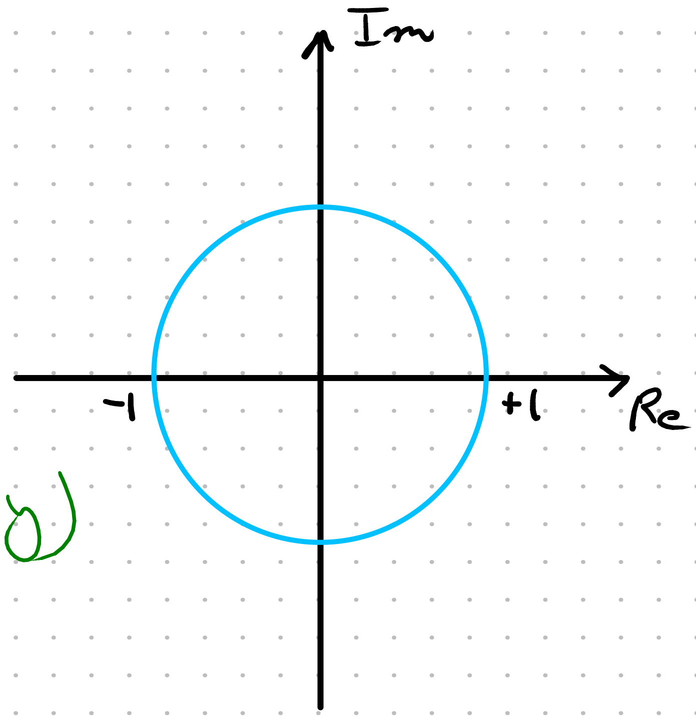
Solve for DT pole: $z = \dfrac{2+T\lambda}{2-T\lambda} = \dfrac{1+\frac{T}{2}\lambda}{1-\frac{T}{2}\lambda}$

$$\underset{(CT)}{\lambda} \longleftrightarrow \frac{1+\frac{T}{2}\lambda}{1-\frac{T}{2}\lambda} \quad (DT)$$

Tustin's approximation: Preserving stability

Tustin's approximation maps a CT pole (Lambda) to a DT pole at:

$$z = \frac{1 + \frac{T}{2}\lambda}{1 - \frac{T}{2}\lambda}$$



If $\underbrace{\mathrm{Re}(\lambda) < 0}_{\text{CT stability}}$, $\Rightarrow$ $\underbrace{|z| < 1}_{\text{DT stability}}$ (verify)

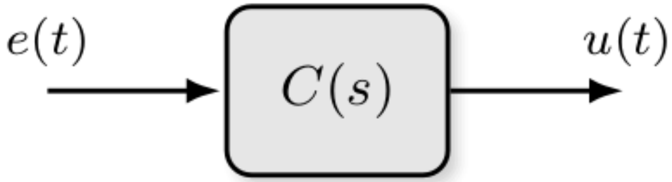
e.g., $\lambda = -2$, DT pole?

$(T = 1)$ at $z = 0$.

Step-invariant transformation

The default method in Matlab's c2d function. Details in chapter 7.

CT Controller:

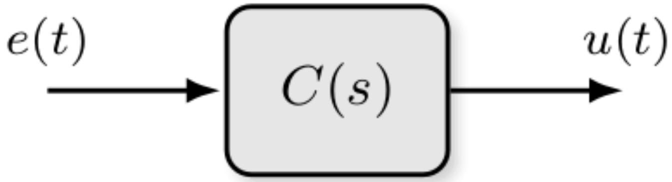


(only for LTI systems)

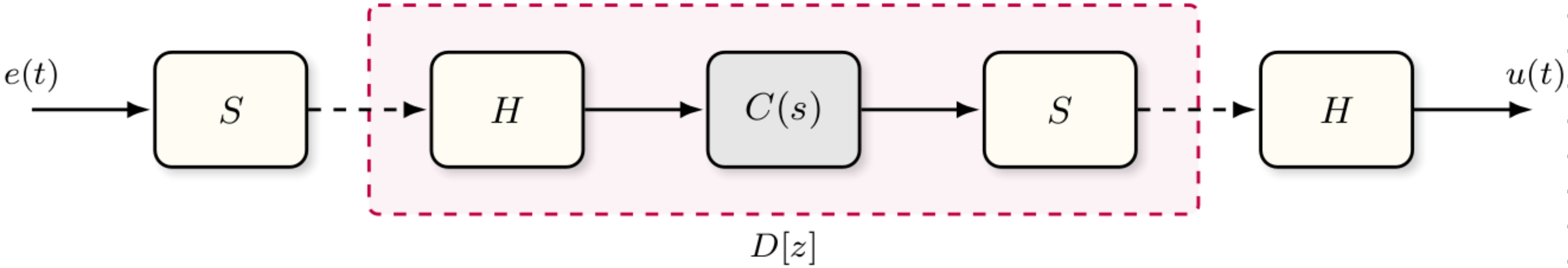
Step-invariant transformation

The default method in Matlab's c2d function. Details in chapter 7.

CT Controller:



Step-invariant DT controller:



As we will see later, this corresponds to mapping poles of C(s) to those of D[z] by the mapping:

$$z = e^{\lambda T}$$

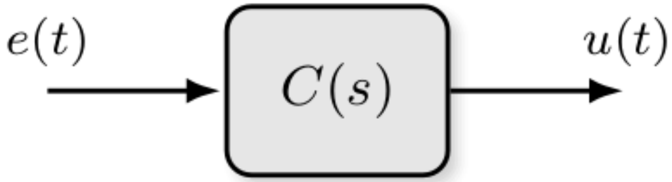
CT pole

DT pole.

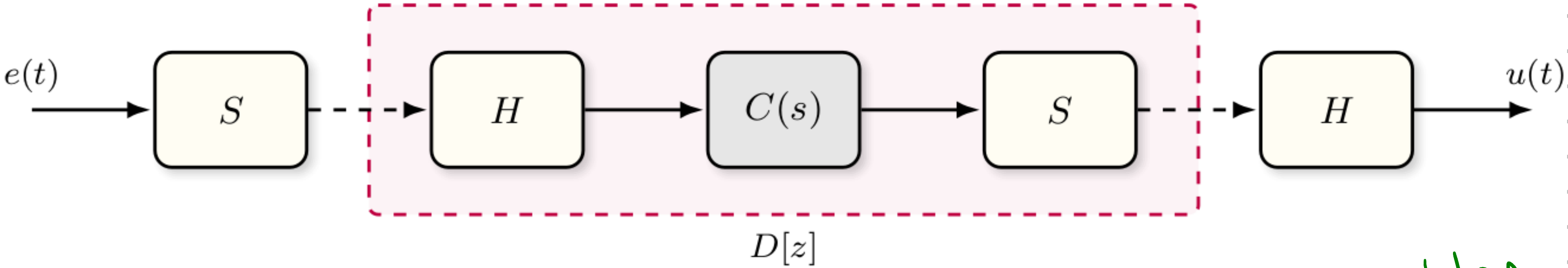
Step-invariant transformation

The default method in Matlab's c2d function. Details in chapter 7.

CT Controller:

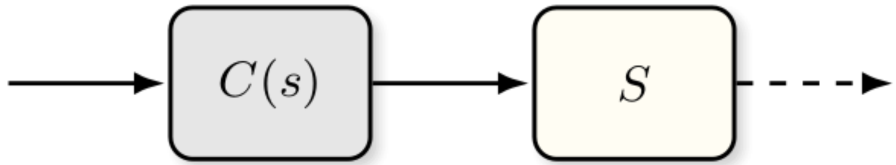


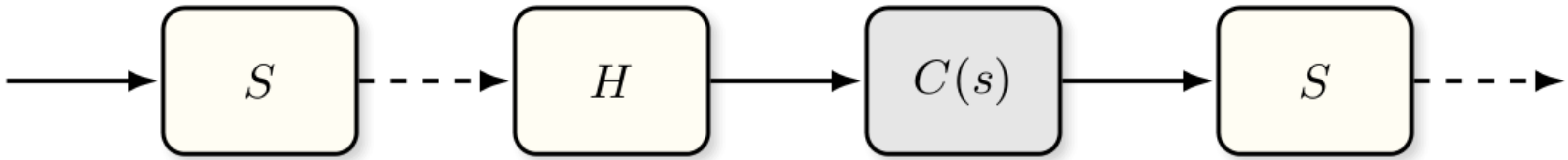
Step-invariant DT controller:



at DT samples.

The response (DT) to a step is the same for both systems:

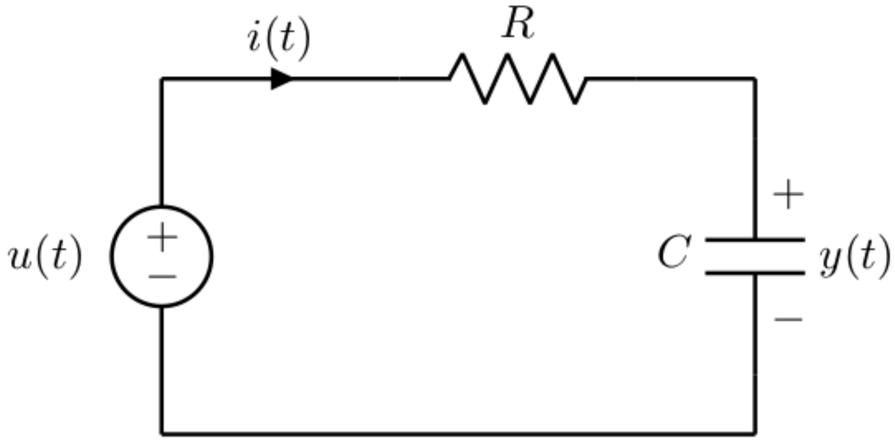




Step-invariant transformation: Example

Transfer function of the system is:

$$\frac{Y}{U} = \frac{1}{RCs + 1}, \text{ let } R = 10, C = 1$$



See matlab code. We will compare the different discretizations.

Outline

[-] Introduction

[-] Ideal sample and zero order hold
    [-] Preserving linearity

[-] Discrete approximations
    [-] Forward and backward Euler (left and right rule)
    [-] Trapezoidal approximation
    [-] Exact (step-invariant) approximation

[X] Controller design via approximation of continuous-time controllers

X = The upcoming topic
- = Topic that has been covered

Emulation: Controller design via approximating continuous-time controllers

1. Given $P(s)$ and closed-loop requirements, design $C(s)$ to satisfy requirements (e.g., pole-placement).

Emulation: Controller design via approximating continuous-time controllers

1. Given $P(s)$ and closed-loop requirements, design $C(s)$ to satisfy requirements (e.g., pole-placement).

2. Approximate $C(s)$ to get $D[z]$ via a discretization procedure (numerical approx. or exact transforms).

Question: Is this enough? Is the closed-loop sampled-data system with {A2D, $D[z]$, D2A} and $P(s)$ guaranteed to be stable or satisfy the performance requirements?

Emulation: Controller design via approximating continuous-time controllers

1. Given $P(s)$ and closed-loop requirements, design $C(s)$ to satisfy requirements (e.g., pole-placement).

2. Approximate $C(s)$ to get $D[z]$ via a discretization procedure (numerical approx. or exact transforms).

Question: Is this enough? Is the closed-loop sampled-data system with {A2D, $D[z]$, C2D} and $P(s)$ guaranteed to be stable or satisfy the performance requirements?

No! The approximations only preserve properties of $C(s)$ to an extent. Unlike direct DT design, there are no guarantees to be had here.

Emulation: Controller design via approximating continuous-time controllers

1. Given $P(s)$ and closed-loop requirements, design $C(s)$ to satisfy requirements (e.g., pole-placement).

2. Approximate $C(s)$ to get $D[z]$ via a discretization procedure (numerical approx. or exact transforms).

3. Simulate the closed-loop sampled-data system (CL composition of $D[z]$ and $P(s)$), extensively.

Emulation: Controller design via approximating continuous-time controllers

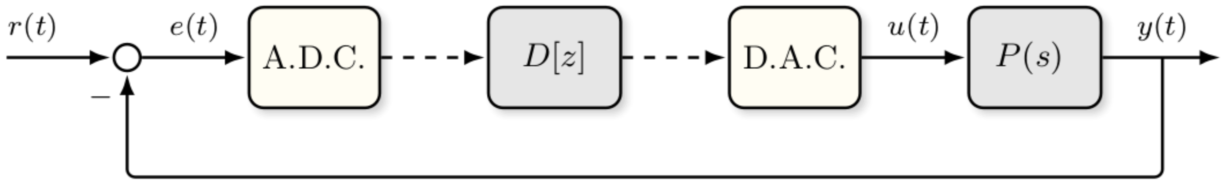
1. Given $P(s)$ and closed-loop requirements, design $C(s)$ to satisfy requirements (e.g., pole-placement).

2. Approximate $C(s)$ to get $D[z]$ via a discretization procedure (numerical approx. or exact transforms).

3. Simulate the closed-loop sampled-data system (CL composition of $D[z]$ and $P(s)$), extensively.

4. If simulation results are satisfactory, implement $D[z]$ as either a difference equation or state-space model.

Example 4.4.1 Lead controller design

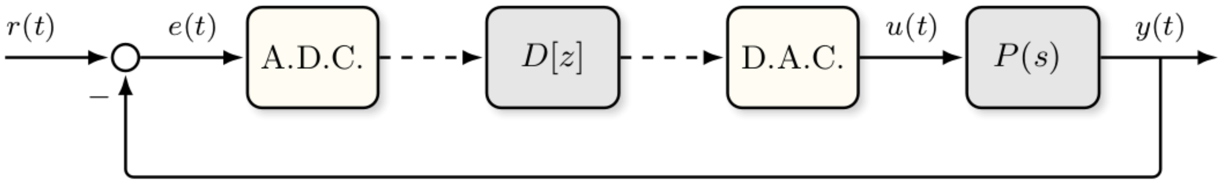


Consider the plant $P(s) = \dfrac{1}{s(s+2)}$

We want to design a discrete time controller (sampling time = 0.2s). Control specifications are:

0. Closed-loop Stability
1. Asymptotic step tracking (note: plant already has a pole at the origin)
2. Percentage overshoot %OS<=5%
3. Settling time Ts<=3 seconds

Example 4.4.1 Lead controller design



Consider the plant $P(s) = \dfrac{1}{s(s+2)}$

We want to design a discrete time controller (sampling time = 0.2s). Control specifications are:

0. Closed-loop Stability
1. Asymptotic step tracking (note: plant already has a pole at the origin)
2. Percentage overshoot %OS<=5%
3. Settling time Ts<=3 seconds

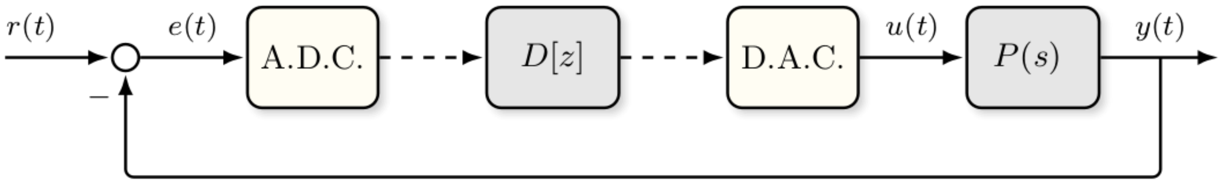
$\Rightarrow \phi_{pm} \geq 65^\circ, \ \omega_{BW} = 2 \text{ rad } s^{-1}$

CT design outline:

Design a lead controller (see appendix 2.B.): $C(s) = K\dfrac{\alpha T s + 1}{Ts + 1}, \qquad \alpha > 1, \ T > 0, \ K > 0$

Mapping the step requirements to frequency response: $\Phi_{\text{pm}}^{\text{des}} = 65^\circ$ (PM) and $\omega_{\text{BW}}^{\text{des}} = 2$ rad/s (bandwidth)

Example 4.4.1 Lead controller design



Consider the plant $P(s) = \dfrac{1}{s(s+2)}$

We want to design a discrete time controller (sampling time = 0.2s). Control specifications are:

0. Closed-loop Stability
1. Asymptotic step tracking (note: plant already has a pole at the origin)
2. Percentage overshoot %OS<=5%
3. Settling time Ts<=3 seconds

CT design outline:

Design a lead controller (see appendix 2.B.): $C(s) = K\dfrac{\alpha T s + 1}{T s + 1}, \qquad \alpha > 1,\ T > 0,\ K > 0$

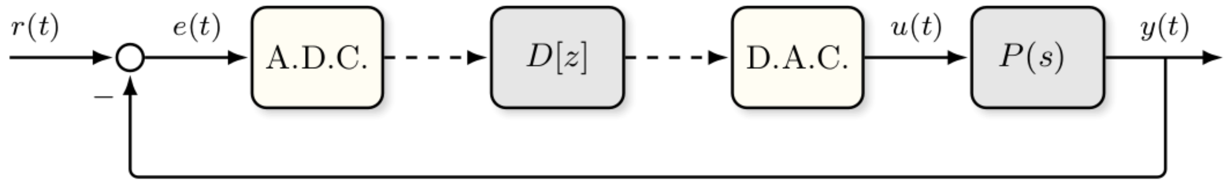
Mapping the step requirements to frequency response: $\Phi_{\text{pm}}^{\text{des}} = 65^\circ$ (PM) and $\omega_{\text{BW}}^{\text{des}} = 2$ rad/s (bandwidth)

To satisfy the bandwidth specification, set $\hat{K} = K\alpha^{1/2}$ so that GCF of loop gain is 2 rad/s

From Bode plot of $\hat{K}P$, we see that we need a phase lead of an addition 20 degrees (approx) at 2 rad/s

$$\phi_{\max} = \Phi_{\text{pm}}^{\text{des}} - \Phi_{\text{pm}} = 20^\circ$$

Example 4.4.1 Lead controller design



Consider the plant $P(s) = \frac{1}{s(s+2)}$

We want to design a discrete time controller (sampling time = 0.2s). Control specifications are:

0. Closed-loop Stability
1. Asymptotic step tracking (note: plant already has a pole at the origin)
2. Percentage overshoot %OS<=5%
3. Settling time Ts<=3 seconds

CT design outline:

Design a lead controller (see appendix 2.B.): $C(s) = K\frac{\alpha Ts+1}{Ts+1}, \qquad \alpha > 1,\ T > 0,\ K > 0$

Mapping the step requirements to frequency response: $\Phi_{\text{pm}}^{\text{des}} = 65°$ (PM) and $\omega_{\text{BW}}^{\text{des}} = 2$ rad/s (bandwidth)

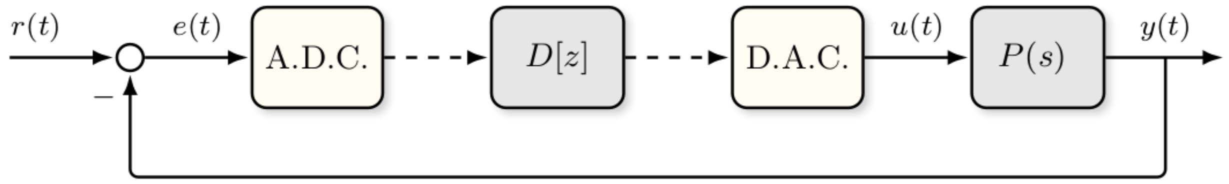
To satisfy the bandwidth specification, set $\hat{K} = K\alpha^{1/2}$ so that GCF of loop gain is 2 rad/s

From Bode plot of $\hat{K}P$, we see that we need a phase lead of an addition 20 degrees (approx) at 2 rad/s

$$\phi_{\max} = \Phi_{\text{pm}}^{\text{des}} - \Phi_{\text{pm}} = 20°$$

For the lead controller design, this implies $\alpha = \frac{1+\sin(\phi_{\max})}{1-\sin(\phi_{\max})} = 2.03 \Rightarrow K = \hat{K}\alpha^{-1/2} = 3.95$

Example 4.4.1 Lead controller design



Consider the plant $P(s) = \dfrac{1}{s(s+2)}$

We want to design a discrete time controller (sampling time = 0.2s). Control specifications are:

0. Closed-loop Stability
1. Asymptotic step tracking (note: plant already has a pole at the origin)
2. Percentage overshoot %OS<=5%
3. Settling time Ts<=3 seconds

CT design outline:

Design a lead controller (see appendix 2.B.): $C(s) = K\dfrac{\alpha T s + 1}{Ts+1}, \qquad \alpha > 1,\ T > 0,\ K > 0$

Mapping the step requirements to frequency response: $\Phi_{\text{pm}}^{\text{des}} = 65^\circ$ (PM) and $\omega_{\text{BW}}^{\text{des}} = 2$ rad/s (bandwidth)

To satisfy the bandwidth specification, set $\hat{K} = K\alpha^{1/2}$ so that GCF of loop gain is 2 rad/s

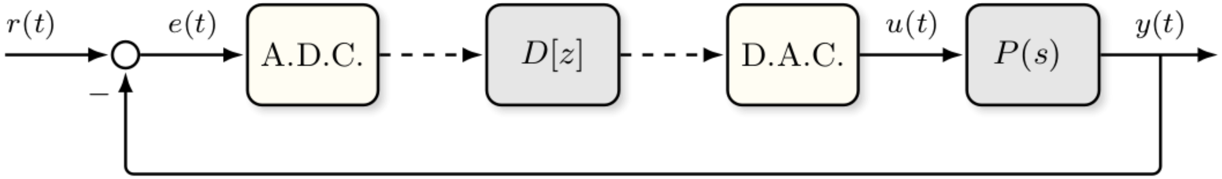
From Bode plot of $\hat{K}P$, we see that we need a phase lead of an addition 20 degrees (approx) at 2 rad/s

$$\phi_{\max} = \Phi_{\text{pm}}^{\text{des}} - \Phi_{\text{pm}} = 20^\circ$$

For the lead controller design, this implies $\alpha = \dfrac{1+\sin(\phi_{\max})}{1-\sin(\phi_{\max})} = 2.03 \Rightarrow K = \hat{K}\alpha^{-1/2} = 3.95$

Finally, to get the additional phase at the right frequency, set $T = \dfrac{1}{\omega_m\sqrt{\alpha}} = 0.3523$

Example 4.4.1 Lead controller design



Consider the plant $P(s) = \dfrac{1}{s(s+2)}$

We want to design a discrete time controller (sampling time = 0.2s). Control specifications are:

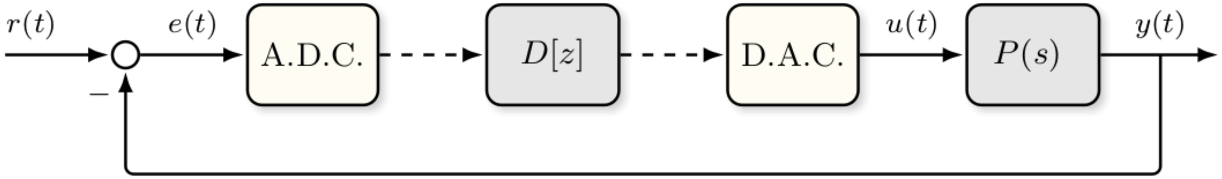
0. Closed-loop Stability
1. Asymptotic step tracking (note: plant already has a pole at the origin)
2. Percentage overshoot %OS<=5%
3. Settling time Ts<=3 seconds

CT controller obtained via lead compensator design: $C(s) = \dfrac{8.014s + 11.2}{s + 2.84}$

Check Bode plot of PC to see GCF and PM.

Step response Ts = 1.69s, %OS=3.74% (satisfied)

Example 4.4.1 Lead controller design



Consider the plant $P(s) = \dfrac{1}{s(s+2)}$

We want to design a discrete time controller (sampling time = 0.2s). Control specifications are:

0. Closed-loop Stability
1. Asymptotic step tracking (note: plant already has a pole at the origin)
2. Percentage overshoot %OS<=5%
3. Settling time Ts<=3 seconds

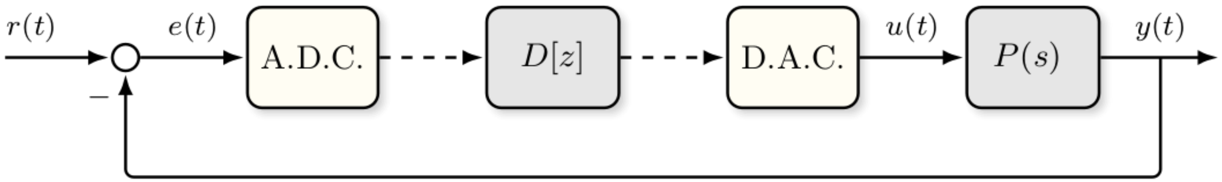
CT controller obtained via lead compensator design: $C(s) = \dfrac{8.014s + 11.2}{s + 2.84}$

Get DT controller (Tustin and step-invariant for comparison):

```
Dbt = c2d(C, 0.2, 'tustin'); % Trapezoidal approximation
Dsi = c2d(C, 0.2); % step-invariant method
```

Example 4.4.1 Lead controller design

Consider the plant $P(s) = \dfrac{1}{s(s+2)}$



We want to design a discrete time controller (sampling time = 0.2s). Control specifications are:

0. Closed-loop Stability
1. Asymptotic step tracking (note: plant already has a pole at the origin)
2. Percentage overshoot %OS<=5%
3. Settling time Ts<=3 seconds

CT controller obtained via lead compensator design: $C(s) = \dfrac{8.014s + 11.2}{s + 2.84}$

Get DT controller (Tustin and step-invariant for comparison):

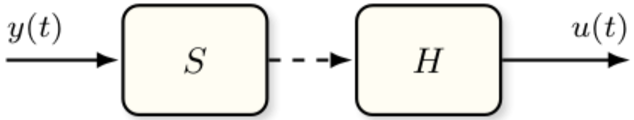
```
Dbt = c2d(C, 0.2, 'tustin'); % Trapezoidal approximation
Dsi = c2d(C, 0.2); % step-invariant method
```

The DT controllers did not meet the CL requirements!

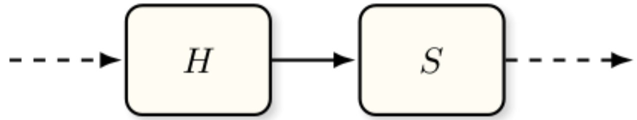
Sampling time (0.2s) is too slow. Close the loop faster.

Worsen the plant to improve the performance of digital controller

What if we cannot just arbitrarily decrease the sampling time ?
Let us stop ignoring the sample/hold operation and 'augment' the plant to approximate these operations.



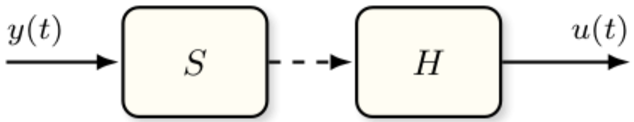
(a) Sample, then hold $H \circ S$.



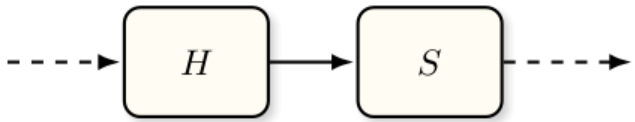
(b) Hold, then sample $S \circ H$.

Worsen the plant to improve the performance of digital controller

What if we cannot just arbitrarily decrease the sampling time?
Let us stop ignoring the sample/hold operation and 'augment' the plant to approximate these operations.



(a) Sample, then hold $H \circ S$.



(b) Hold, then sample $S \circ H$.

H o S is not LTI, but we can approximate it with a system that has the same impulse response:

$$r(t) = \frac{1}{T}\mathbf{1}(t) - \frac{1}{T}\mathbf{1}(t-T)$$

Worsen the plant to improve the performance of digital controller

What if we cannot just arbitrarily decrease the sampling time?

Let us stop ignoring the sample/hold operation and 'augment' the plant to approximate these operations.

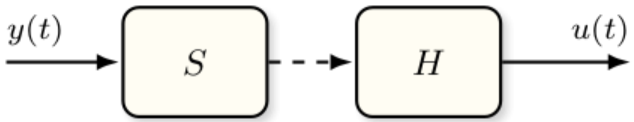
(a) Sample, then hold $H \circ S$. (b) Hold, then sample $S \circ H$.

H o S is not LTI, but we can approximate it with a system that has the same impulse response:

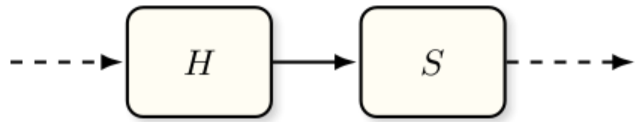
$r(t) = \frac{1}{T}\mathbf{1}(t) - \frac{1}{T}\mathbf{1}(t-T)$, which has a Laplace transform: $R(s) = \frac{1 - e^{-sT}}{sT}$ → Delay of T seconds.

Worsen the plant to improve the performance of digital controller

What if we cannot just arbitrarily decrease the sampling time ?
Let us stop ignoring the sample/hold operation and 'augment' the plant to approximate these operations.



(a) Sample, then hold $H \circ S$.



(b) Hold, then sample $S \circ H$.

H o S is not LTI, but we can approximate it with a system that has the same impulse response:

$r(t) = \frac{1}{T}\mathbf{1}(t) - \frac{1}{T}\mathbf{1}(t-T)$, which has a Laplace transform: $R(s) = \frac{1 - e^{-sT}}{sT}$

For frequency response of this approximation of sample/hold system, consider:

$$
\begin{aligned}
R(j\omega) &= \frac{1 - e^{-j\omega T}}{j\omega T} \\
&= e^{-j\omega \frac{T}{2}} \frac{e^{j\omega \frac{T}{2}} - e^{-j\omega \frac{T}{2}}}{j\omega T} \\
&= e^{-j\omega \frac{T}{2}} \frac{2\sin\left(\omega \frac{T}{2}\right)}{\omega T} \\
&= e^{-j\omega \frac{T}{2}} \frac{\sin\left(\omega \frac{T}{2}\right)}{\omega \frac{T}{2}}.
\end{aligned}
$$

Worsen the plant to improve the performance of digital controller

What if we cannot just arbitrarily decrease the sampling time ?
Let us stop ignoring the sample/hold operation and 'augment' the plant to approximate these operations.

(a) Sample, then hold $H \circ S$.

(b) Hold, then sample $S \circ H$.

H o S is not LTI, but we can approximate it with a system that the same impulse response:

$r(t) = \frac{1}{T}\mathbf{1}(t) - \frac{1}{T}\mathbf{1}(t-T)$, which has a Laplace transform: $R(s) = \frac{1 - \mathrm{e}^{-sT}}{sT}$

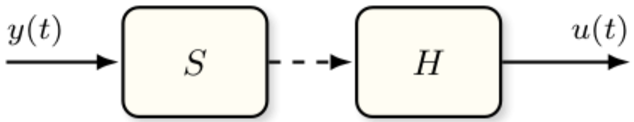
For frequency response of this approximation of sample/hold system, consider:

$$
\begin{aligned}
R(j\omega) &= \frac{1 - \mathrm{e}^{-j\omega T}}{j\omega T} \\
&= \mathrm{e}^{-j\omega \frac{T}{2}} \frac{\mathrm{e}^{j\omega \frac{T}{2}} - \mathrm{e}^{-j\omega \frac{T}{2}}}{j\omega T} \\
&= \mathrm{e}^{-j\omega \frac{T}{2}} \frac{2\sin\left(\omega \frac{T}{2}\right)}{\omega T} \\
&= \mathrm{e}^{-j\omega \frac{T}{2}} \frac{\sin\left(\omega \frac{T}{2}\right)}{\omega \frac{T}{2}}.
\end{aligned}
$$

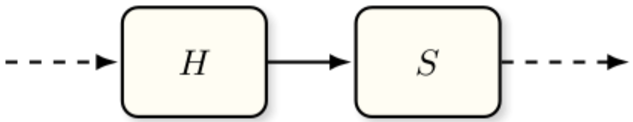
At low frequencies (i.e. w -> 0), we get $R(s) \approx \mathrm{e}^{-s\frac{T}{2}}$

Worsen the plant to improve the performance of digital controller

What if we cannot just arbitrarily decrease the sampling time ?
Let us stop ignoring the sample/hold operation and 'augment' the plant to approximate these operations.



(a) Sample, then hold $H \circ S$.



(b) Hold, then sample $S \circ H$.

H o S is not LTI, but we can approximate it with a system that the same impulse response:

$r(t) = \frac{1}{T}\mathbf{1}(t) - \frac{1}{T}\mathbf{1}(t-T)$, which has a Laplace transform: $R(s) = \frac{1-\mathrm{e}^{-sT}}{sT}$

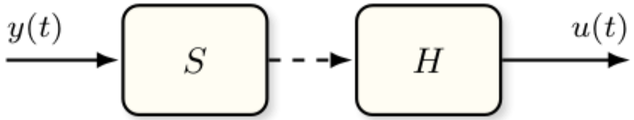
For frequency response of this approximation of sample/hold system, consider:

$$
\begin{aligned}
R(j\omega) &= \frac{1-\mathrm{e}^{-j\omega T}}{j\omega T} \\
&= \mathrm{e}^{-j\omega\frac{T}{2}} \frac{\mathrm{e}^{j\omega\frac{T}{2}} - \mathrm{e}^{-j\omega\frac{T}{2}}}{j\omega T} \\
&= \mathrm{e}^{-j\omega\frac{T}{2}} \frac{2\sin\left(\omega\frac{T}{2}\right)}{\omega T} \\
&= \mathrm{e}^{-j\omega\frac{T}{2}} \frac{\sin\left(\omega\frac{T}{2}\right)}{\omega\frac{T}{2}}.
\end{aligned}
$$

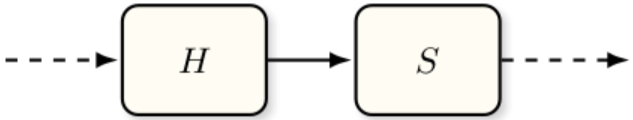
At low frequencies (i.e. w -> 0), we get $R(s) \approx \mathrm{e}^{-s\frac{T}{2}}$ (Time delay of T/2 seconds)

Worsen the plant to improve the performance of digital controller

What if we cannot just arbitrarily decrease the sampling time ?
Let us stop ignoring the sample/hold operation and 'augment' the plant to approximate these operations.



(a) Sample, then hold $H \circ S$.



(b) Hold, then sample $S \circ H$.

H o S is not LTI, but we can approximate it with a system that the same impulse response:

$r(t) = \frac{1}{T}\mathbf{1}(t) - \frac{1}{T}\mathbf{1}(t-T)$, which has a Laplace transform: $R(s) = \frac{1-e^{-sT}}{sT}$

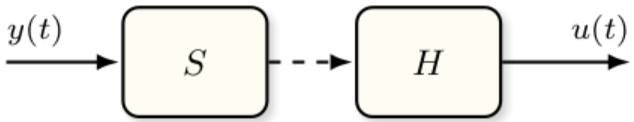
At low frequencies (i.e. w -> 0), we get $R(s) \approx e^{-s\frac{T}{2}}$ (Time delay of T/2 seconds)

Approach: Design a continuous-time controller for the delayed plant $\underbrace{e^{-s\frac{T}{2}} P(s)}$

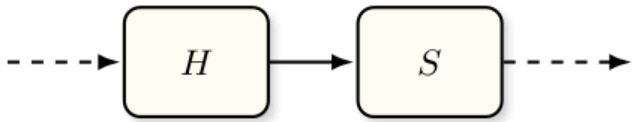
Augmented plant.

Worsen the plant to improve the performance of digital controller

What if we cannot just arbitrarily decrease the sampling time?
Let us stop ignoring the sample/hold operation and 'augment' the plant to approximate these operations.



(a) Sample, then hold $H \circ S$.



(b) Hold, then sample $S \circ H$.

H o S is not LTI, but we can approximate it with a system that the same impulse response:

$r(t) = \frac{1}{T}\mathbf{1}(t) - \frac{1}{T}\mathbf{1}(t-T)$, which has a Laplace transform: $R(s) = \frac{1 - e^{-sT}}{sT}$

At low frequencies (i.e. w -> 0), we get $R(s) \approx e^{-s\frac{T}{2}}$ (Time delay of T/2 seconds)

Approach: Design a continuous-time controller for the delayed plant $e^{-s\frac{T}{2}} P(s)$

Two ways to do this:

a) Using frequency domain methods; compensate for the phase lag introduced by the delay.
b) Linarize the delay (Pade approximation) and then work with the resulting plant.

We will explore the first method, continuing the previous example.

Example 4.4.2

Consider the augmented plant $P_{\mathrm{w}}(s) := \mathrm{e}^{-s\frac{T}{2}} \dfrac{1}{s(s+2)}$

We pick the same $\hat{K}$ (5.6) for the bandwidth specification, leading to a GCF of 2 rad/s

We now need to increase phase by $\phi_{\max} = 31.3^\circ$ at 2 rad/s to get a PM of 65 degrees as desired.

Example 4.4.2

Consider the augmented plant $P_{\mathrm{w}}(s) := \mathrm{e}^{-s\frac{T}{2}} \dfrac{1}{s(s+2)}$

We pick the same $\hat{K}$ (5.6) for the bandwidth specification, leading to a GCF of 2 rad/s

We now need to increase phase by $\phi_{\max} = 31.3^\circ$ at 2 rad/s to get a PM of 65 degrees as desired.

Like before, this gives us

$$\alpha = \frac{1 + \sin(\phi_{\max})}{1 - \sin(\phi_{\max})} = 3.1622 \Rightarrow K = \hat{K}/\alpha^{1/2} = 3.1623$$

To add the phase at the right frequency (2 rad/s), we set $T = \dfrac{1}{\omega_m \sqrt{\alpha}} = 0.2823$

Example 4.4.2

Consider the augmented plant $P_w(s) := e^{-s\frac{T}{2}} \frac{1}{s(s+2)}$

We pick the same $\hat{K}$ (5.6) for the bandwidth specification, leading to a GCF of 2 rad/s

We now need to increase phase by $\phi_{max} = 31.3^\circ$ at 2 rad/s to get a PM of 65 degrees as desired.

Like before, this gives us

$$\alpha = \frac{1+\sin(\phi_{max})}{1-\sin(\phi_{max})} = 3.1622 \Rightarrow K = \hat{K}/\alpha^{1/2} = 3.1623$$

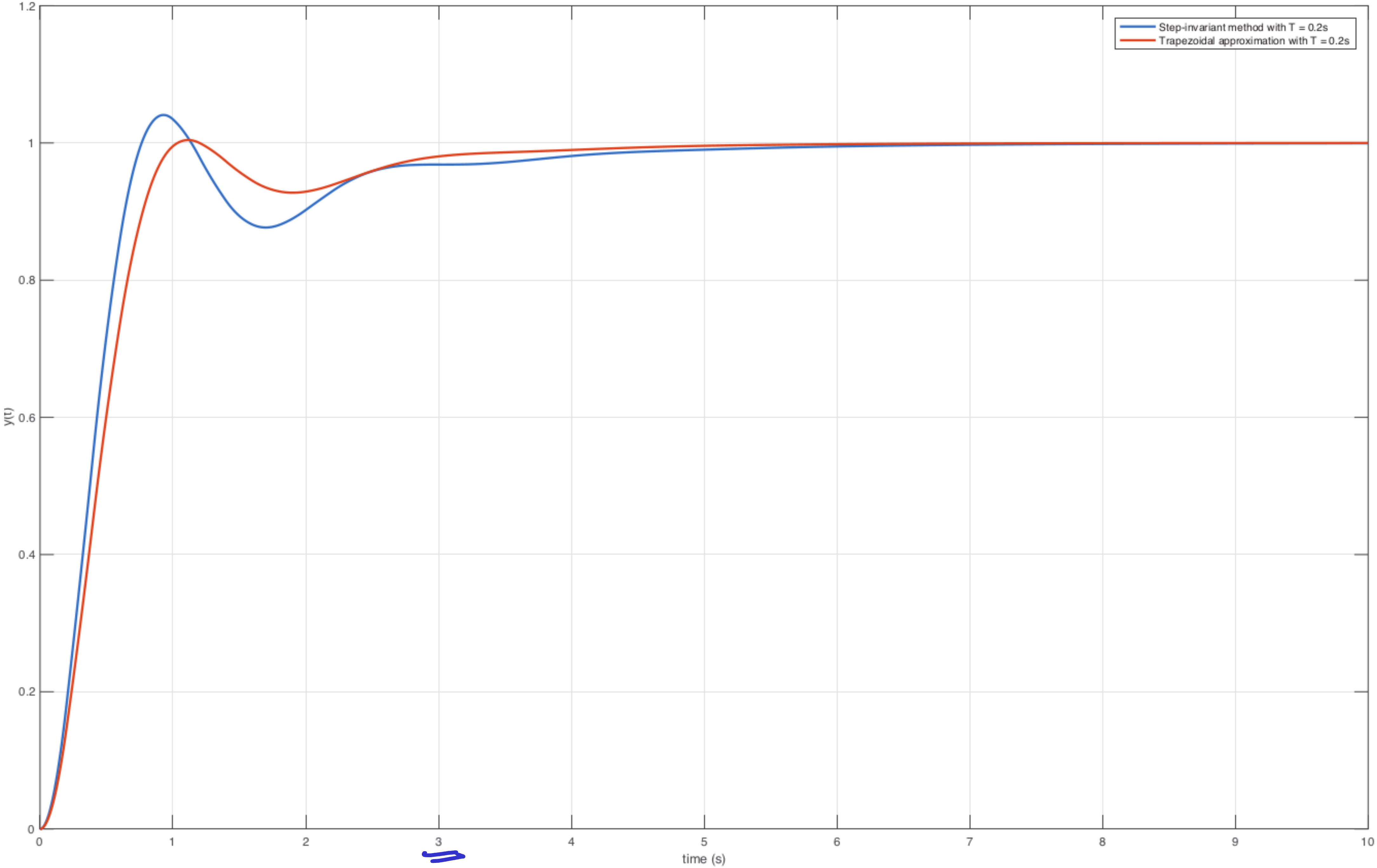
To add the phase at the right frequency (2 rad/s), we set $T = \frac{1}{\omega_m\sqrt{\alpha}} = 0.2823$

All this gives us the CT lead-controller:

$$C(s) = 10\frac{s+11.2}{s+3.542}$$

Get the DT controllers as before by discretizing C(s)

Example 4.4.2: Continued



Meet overshoot requirement without needing to sample faster, but the response is sluggish.

Approximating delay to work in the s-domain: Pade approximation

First order Pade approximation of delay: $e^{-sT} \approx \dfrac{1 - \frac{T}{2}s}{1 + \frac{T}{2}s}$

Now, the sample-then-hold (H o S) operation is approximated with the TF:

$$R(s) = \frac{1 - e^{-sT}}{sT} \approx \frac{1}{1 + s\frac{T}{2}}$$

The augmented plant is:

$$P_{\mathrm{w}}(s) := \frac{1}{1 + s\frac{T}{2}} P(s) \approx R(s)P(s)$$

We can now design a controller for this augmented plant, and then discretize it.

Outline

[-] Introduction

[-] Ideal sample and zero order hold
    [-] Preserving linearity

[-] Discrete approximations
    [-] Forward and backward Euler (left and right rule)
    [-] Trapezoidal approximation
    [-] Exact (step-invariant) approximation

[-] Controller design via approximation continuous-time controllers

X = The upcoming topic
- = Topic that has been covered