Aalto University
School of Electrical Engineering

ELEC-E8101: Digital and Optimal Control

Lecture 7
Discretization

Ville Kyrki
Office 2570, Maarintie 8 Building
Email: ville.kyrki@aalto.fi

Slides based on ELEC-E8101 material by Themistoklis Charalambous

Practicalities

Practical information for the rest of the course:

- Lecturer changes to Ville Kyrki, two weeks guest lecturer Gökhan Alcan.
    - Change may cause some overlap in topics and changes in notation.
    - For the rest of the course, we will not follow precisely Åström & Wittenmark.

- Homeworks will be more practice and programming-based.

- First quiz from this part in a week (Thu Nov 3 – Fri Nov 4).

- First exercise session related to this part on Wed Nov 2.

Syllabus

Topics for the rest of the course:

- Lecture 7 Discretization (28.10.)
- Lecture 8 Discrete PID (4.11.)
- Lecture 9 Disturbances (11.11.)
- Lecture 10 Optimal control in state space (18.11., Gökhan)
- Lecture 11 Introduction to stochastic optimal control (25.11., Gökhan)
- Lecture 12 Summary (2.12.)

Today

Let's review:

- Sampling and Zero-order hold (+ notation used by me)

Then we go to:

- Options of discretization in control systems

- Discretization methods for designing discrete-time systems

Recap: Sampling

The bridge from continuous to discrete

- The world surrounding us is *analog*!

- Signal analysis and control, for the reasons discussed in the introduction of this course, is done via digital computers

- As a result, analog signals and systems have to be transformed to discrete and vice-versa

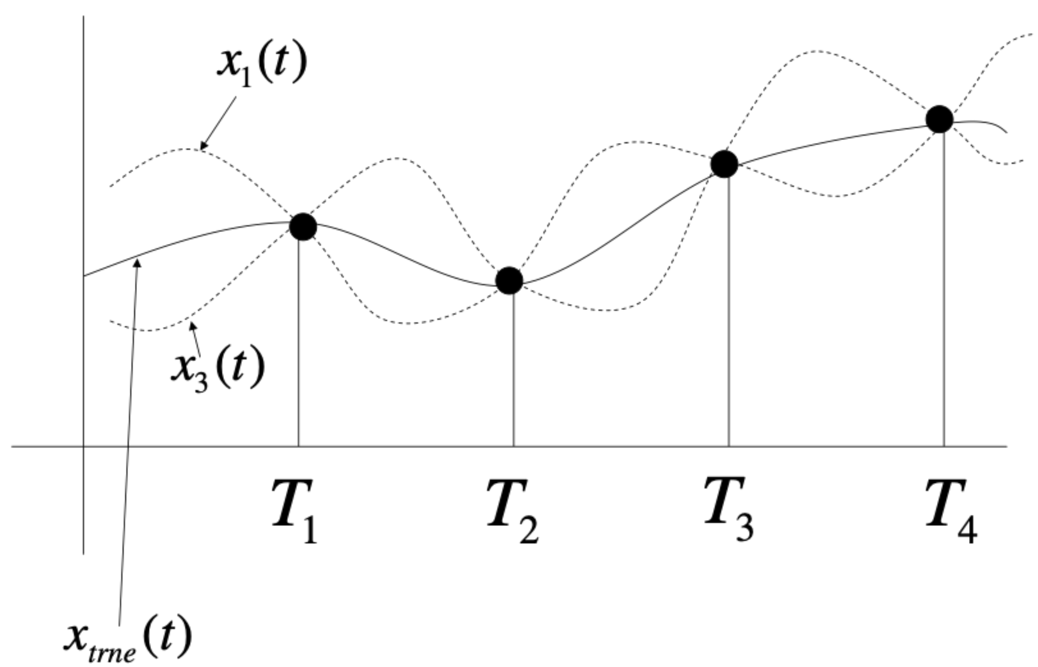
- This is done via ***sampling***! The measured entities are called *samples*. The sampling takes place in regular intervals, say every T_s seconds. Hence,

$$\boxed{x[k] \equiv x(kT_s)}$$

Recap: Sampling

The bridge from continuous to discrete

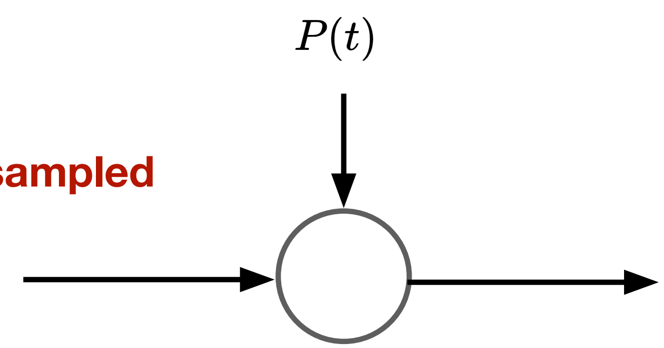
- Time interval T_s is known as the *sampling interval*
- The *sampling frequency* is $f_s = 1/T_s$ (Hz)
- The *sampling angular frequency* is $\omega_s = 2\pi/T_s = 2\pi f_s$ (rad/s)
- In general, a signal cannot be reconstructed uniquely by its samples



Recap: Sampling

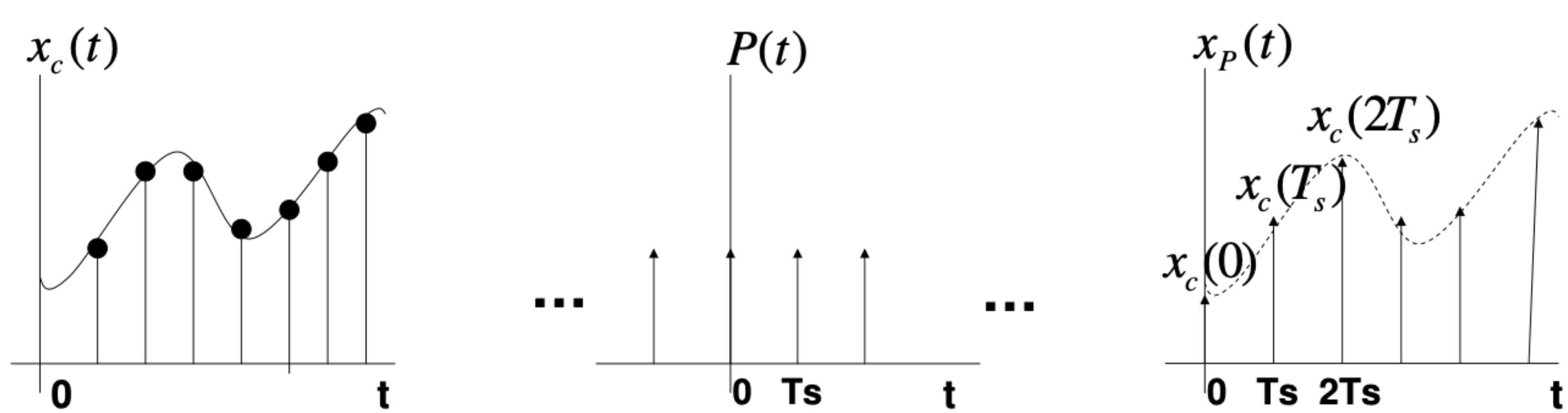
The bridge from continuous to discrete

Signal to be sampled



Sampled signal

where



Frequency range of a signal

Did you discuss this earlier?

- Suppose that the Fourier transform of $x_p(t)$ exists, i.e., $\mathcal{F}\{x_p(t)\} = X_p(j\omega)$

- Then, $X_p(j\omega) = \mathcal{F}\{x_p(t)\} = \mathcal{F}\{x_c(t)p(t)\} = \frac{1}{2\pi} X_c(j\omega) \star P(j\omega)$

- Since $p(t)$ is periodic

$$p(t) = \sum_{k=-\infty}^{\infty} a_k e^{jk\omega_s t} \text{ (Fourier series)}$$

$$a_n = \frac{1}{T_s} \int_{-\frac{T_s}{2}}^{\frac{T_s}{2}} p(t) e^{-jn\omega_s t} = \frac{1}{T_s} \int_{-\frac{T_s}{2}}^{\frac{T_s}{2}} \delta(t) e^{-jn\omega_s t} = \frac{1}{T_s}, \ \forall n$$

$$\Rightarrow p(t) = \frac{1}{T_s} \sum_{k=-\infty}^{\infty} e^{jk\omega_s t}$$

- Moreover, $\mathcal{F}\{e^{jk\omega_s t}\} = 2\pi\delta(\omega - k\omega_s) \Rightarrow P(j\omega) = \frac{2\pi}{T_s} \sum_{n=-\infty}^{\infty} \delta(\omega - n\omega_s), \ \omega_s = \frac{2\pi}{T_s}$

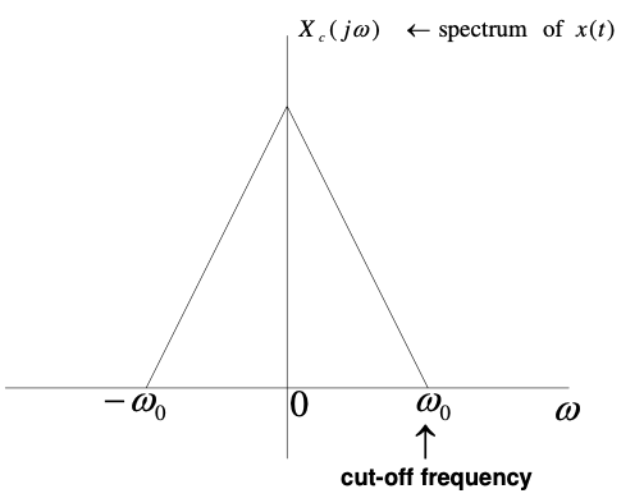
Frequency range

- As a result:

$$X_p(j\omega) = \frac{1}{T_s} \sum_{n=-\infty}^{\infty} X_c(j\omega) \star \delta(\omega - n\omega_s)$$

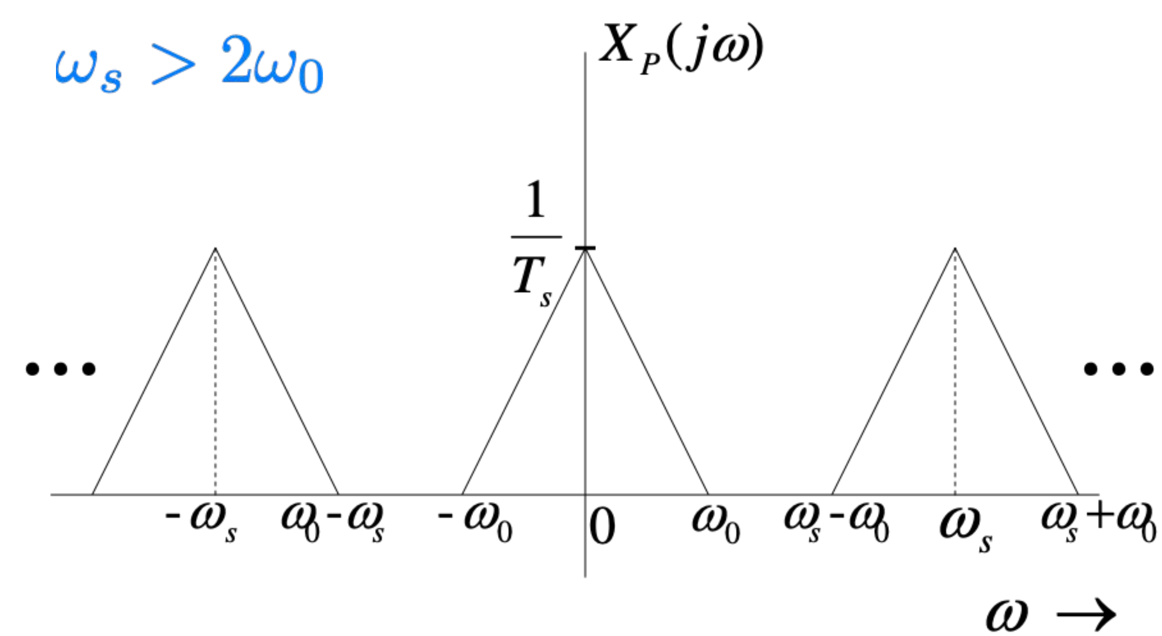
$$= \frac{1}{T_s} \sum_{n=-\infty}^{\infty} X_c(j(\omega - n\omega_s)) \star \delta(\omega) \quad \text{(property of convolution)}$$

$$\Rightarrow X_p(j\omega) = \frac{1}{T_s} \sum_{n=-\infty}^{\infty} X_c(j(\omega - n\omega_s))$$



(bandwidth)

$\omega_s > 2\omega_0$



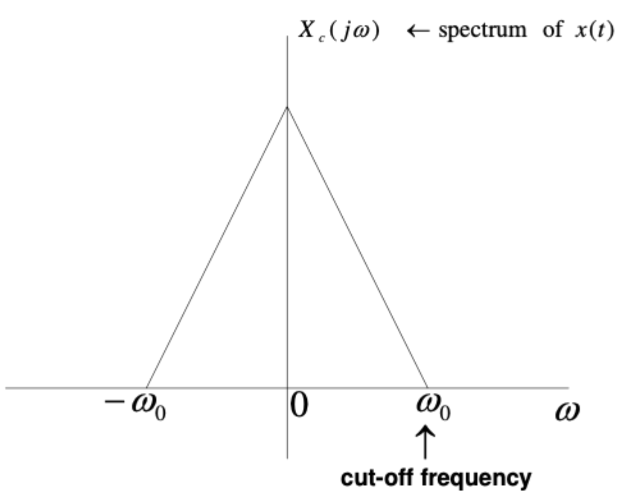
Frequency range

- As a result:

$$X_p(j\omega) = \frac{1}{T_s}\sum_{n=-\infty}^{\infty} X_c(j\omega) \star \delta(\omega - n\omega_s)$$

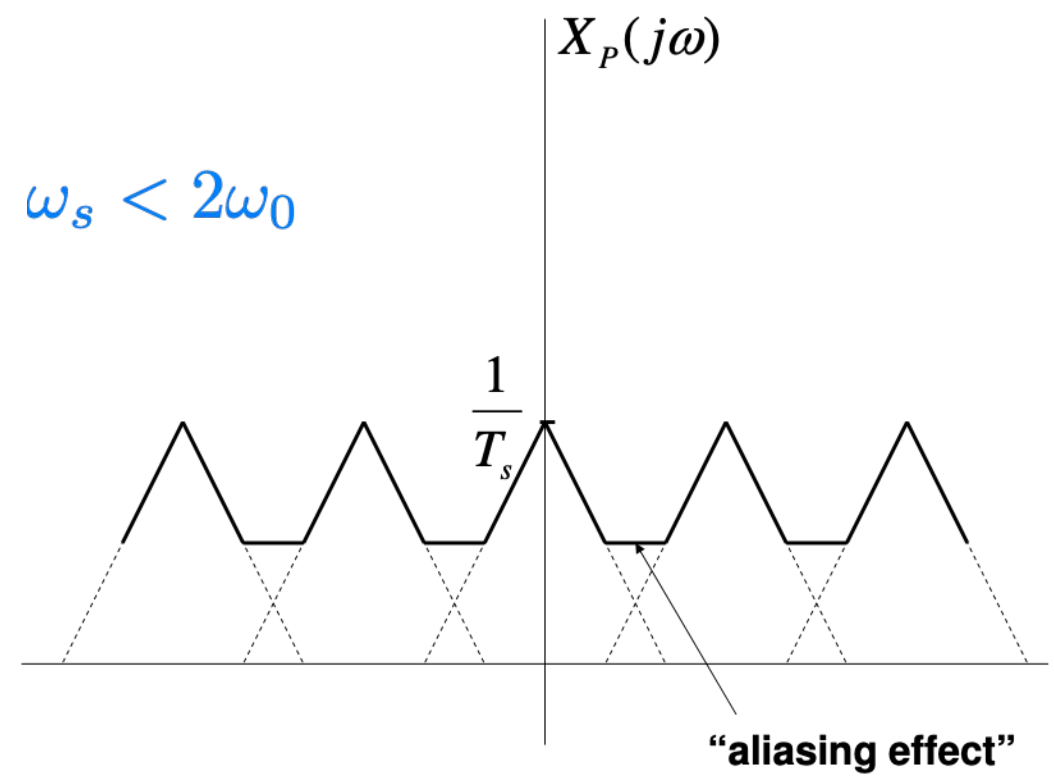
$$= \frac{1}{T_s}\sum_{n=-\infty}^{\infty} X_c(j(\omega - n\omega_s)) \star \delta(\omega) \quad \text{(property of convolution)}$$

$$\Rightarrow X_p(j\omega) = \frac{1}{T_s}\sum_{n=-\infty}^{\infty} X_c(j(\omega - n\omega_s))$$



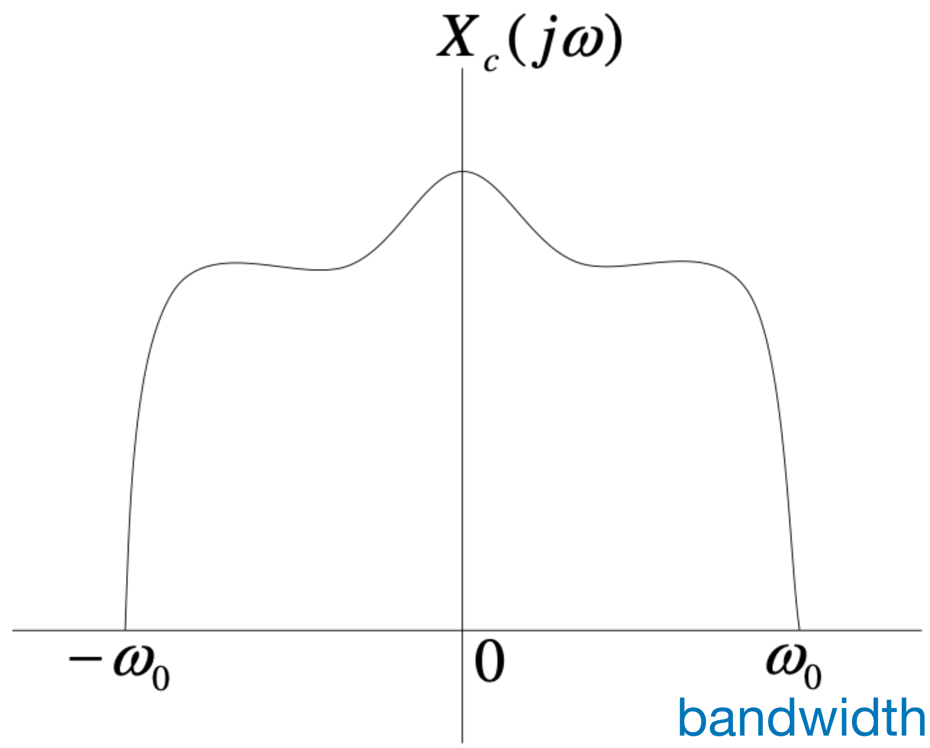
(bandwidth)

$\omega_s < 2\omega_0$



Sampling criterion/theorem

- Suppose $x_c(t)$ is a low-pass signal with $X_c(j\omega) = 0, \forall\ |\omega| > \omega_0$, e.g.,



- Then, $x_c(t)$ can be uniquely determined by its samples $x_c(nT_s), n = 0, \pm 1, \pm 2, \ldots$

if the sampling angular frequency is at least twice as big as ω_0, i.e.,

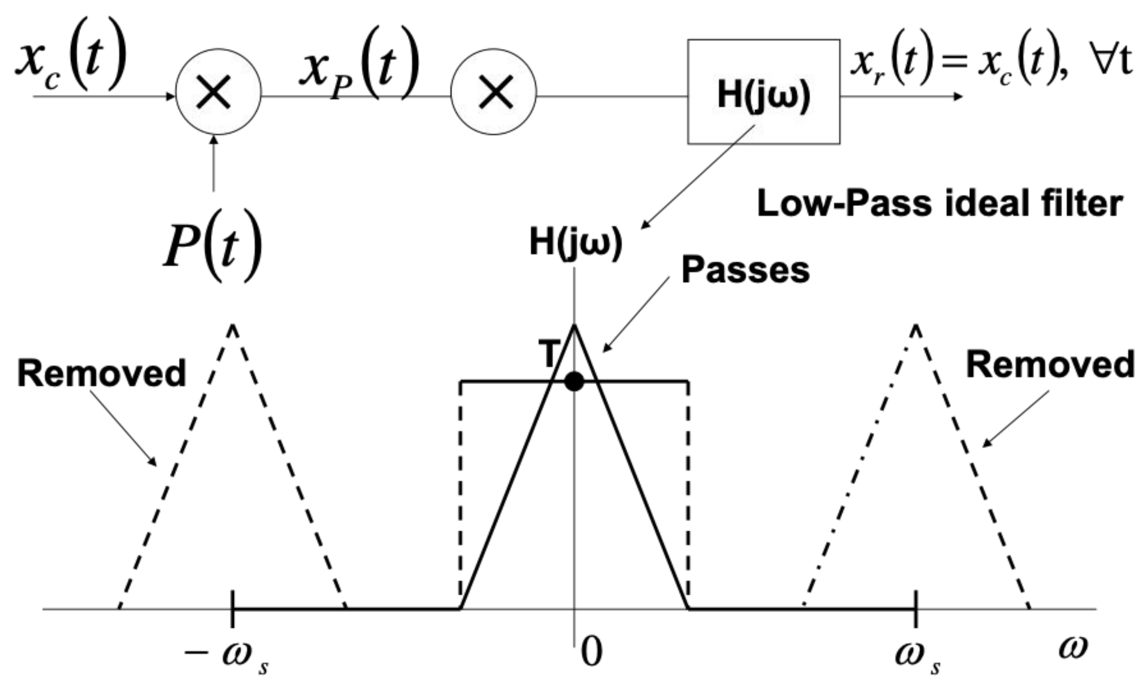
$$\omega_s = \frac{2\pi}{T_s} > 2\omega_0$$

- The minimum sampling angular frequency, for which the inequality holds, is called the *Nyquist angular frequency*

Remarks on sampling

- Reconstructing such a signal requires a low-pass filter with cut-off frequency:

$$\omega_0 < \omega_c < \omega_s - \omega_0$$



- From the natural meaning of frequency, we understand that the fast changes of a signal in time domain correspond to the existence of high frequencies with high energy

- The bigger the bandwidth of a signal, the faster the changes in the time domain

Remarks on sampling

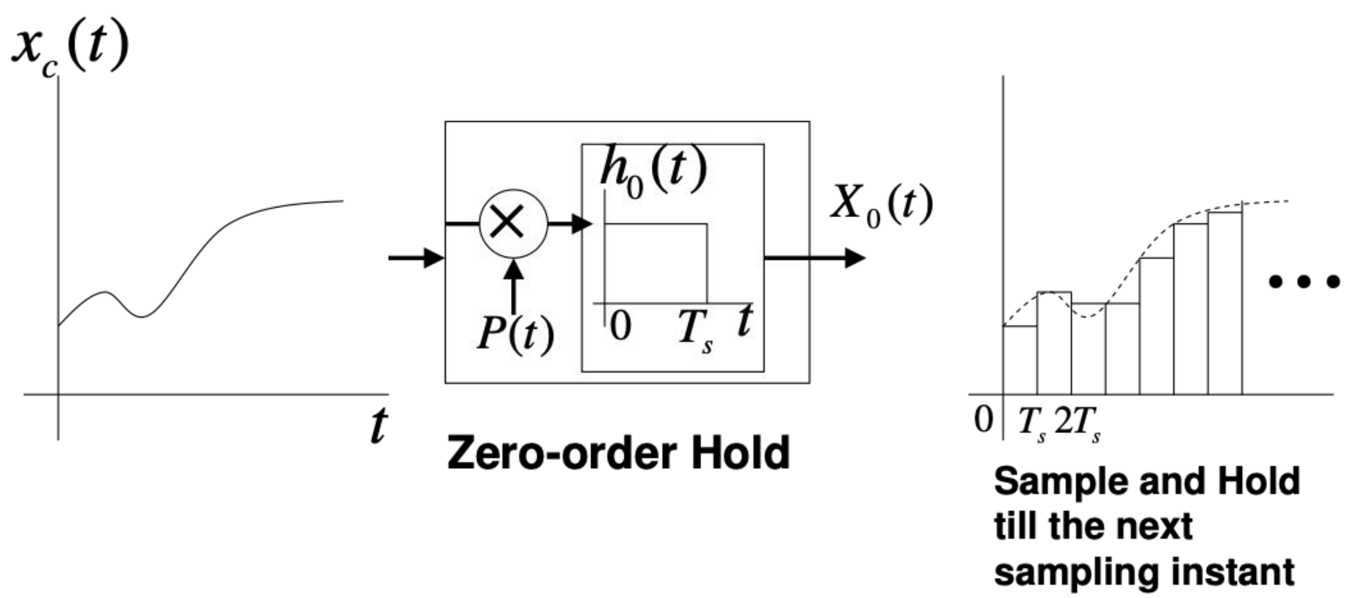
- Signals with a limited bandwidth are called band-limited. If a signal is not band-limited, then obviously its perfect reconstruction is not possible. One can only approximate it!

- However we should choose the sampling period to be:
  - small enough, so that information loss is small
  - high enough, so that the system does not run out of memory

- The reconstruction of a signal might be adequate if $X(j\omega)$ reduces to zero fast for $\omega \to \infty$, which is the case in most practical systems

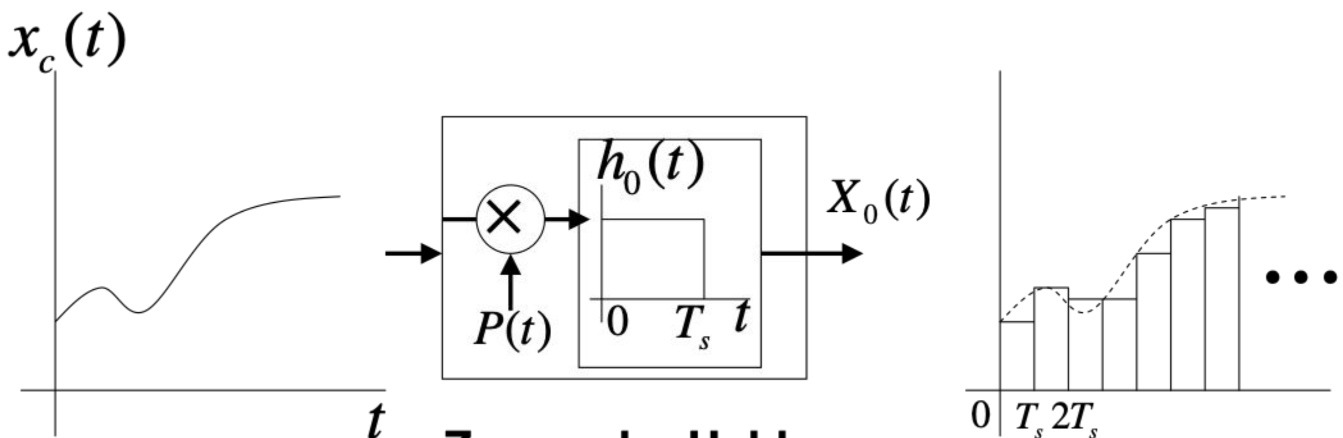
Sampling with Zero-Order Hold (ZOH)

- Sampling narrow and large-amplitude pulses which approximate impulses are in practice difficult to generate and transmit
- Therefore, it is often more convenient to generate the sampled signal in a form referred to as a zero-order hold (ZOH)
- Such a system samples x_c at a given instant and holds that value until the next instant



Recap: Zero-Order Hold (ZOH) sampling

- A simple sampling method is ZOH, in which a sample is held till the next sampling instant



Zero-order Hold

Sample and Hold till the next sampling instant

$$
\begin{aligned}
x_0(t) &= x_P(t) * h_0(t) \\
&= \sum_{n=-\infty}^{\infty} x_c(nT_s)\delta(t - nT_s) * h_0(t) \\
&= \sum_{n=-\infty}^{\infty} x_c(nT_s)\delta(t) * h_0(t - nT_s)
\end{aligned}
$$

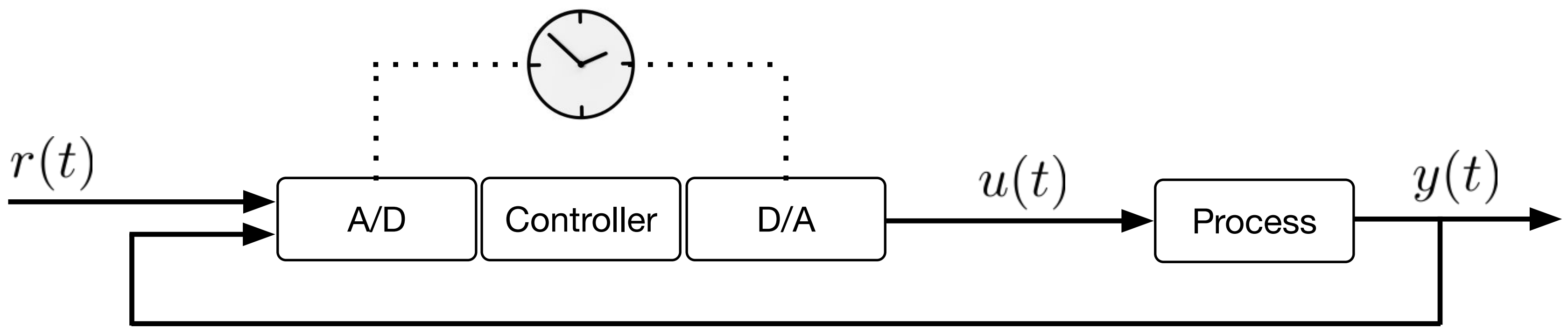
$$
\Rightarrow x_0(t) = \sum_{n=-\infty}^{\infty} x(nT_s)h_0(t - nT_s)
$$

Recap: Discretization

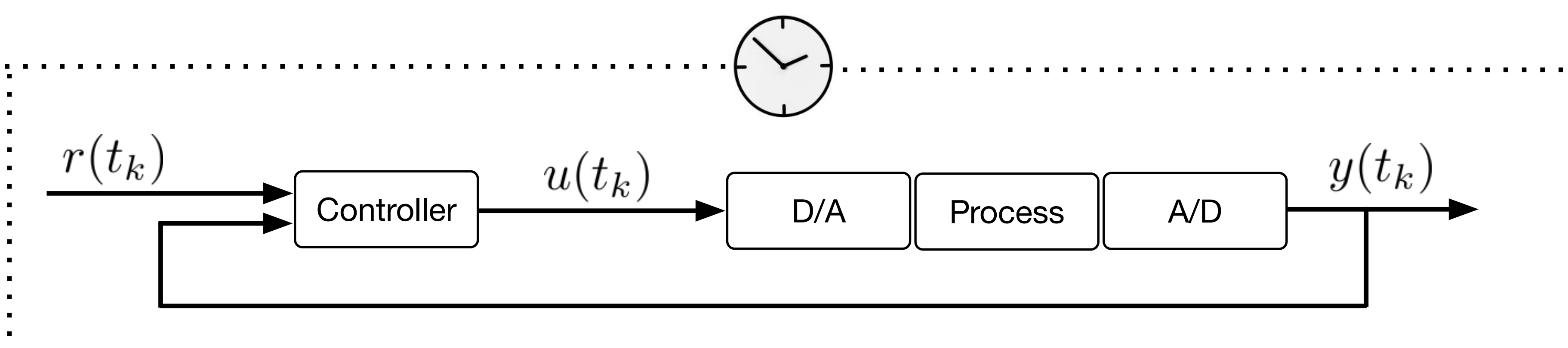
Do you remember this from an earlier lecture?

- **There are 2 main design approaches:**

(a) Discretize the analog controller



(b) Discretize the process and do the design totally in discrete time



Discretization methods

- We will see how we can transform the transfer function $G(s)$ of an analog system in the s-domain to an equivalent transfer function $G(z)$ of a discrete system in the z-domain.

No unique way!

- $G(s)$ can be transformed to $G(z)$ by 3 different approaches:
  - Rely on the use of numerical or analytical methods for solving differential equations describing the given system and for converting them to difference equations (earlier lecture)
  - Match the response of continuous-time systems to specific inputs (e.g., impulse, step and ramp functions) to those of discrete-time systems for the same inputs
  - Match the poles and zeros of $G(s)$ in s-domain with the corresponding poles and zeros of $G(z)$ in z-domain

	s-domain		z-domain
pole	$s = s_0$	⇒	$z = e^{s_0 T_s}$ ← not algebraic!

Aalto University
School of Electrical Engineering

Can we find simple approximations?

Backward difference method

Why do we want to compute a derivative in a practical controller?

- Approximate the derivative by using the difference between the current and the previous sample divided by the sampling period, i.e.,

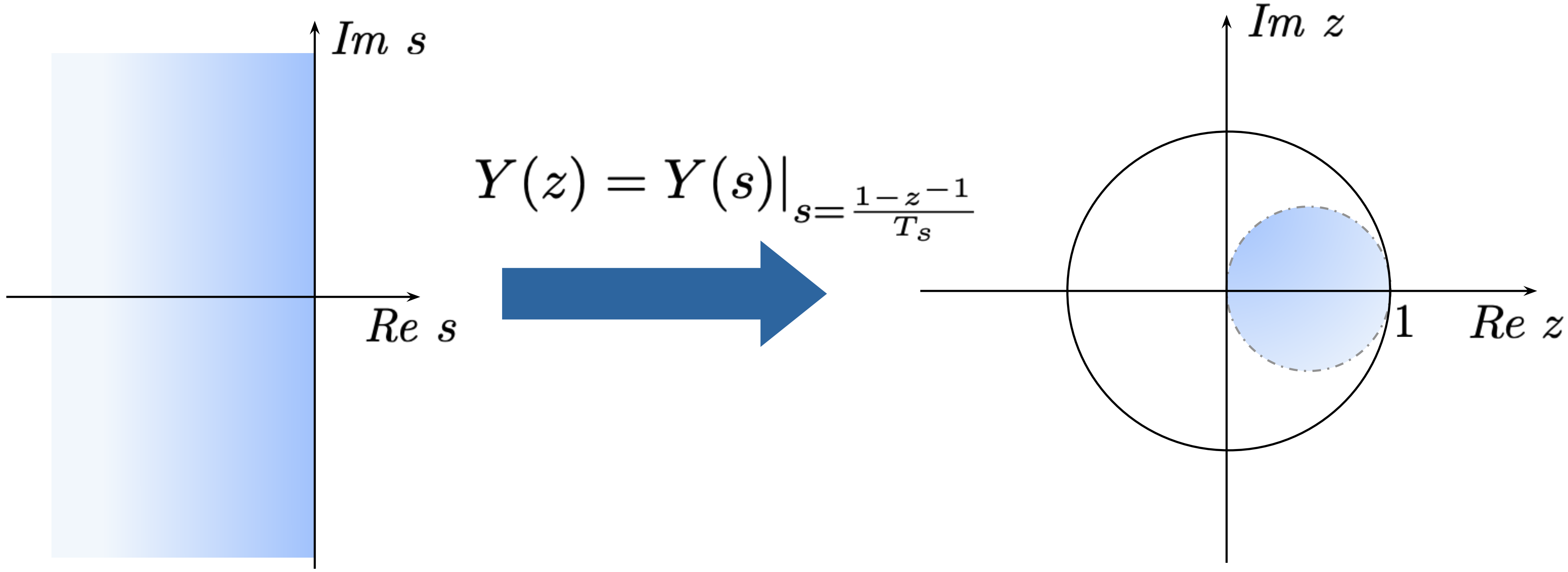
$$\frac{d}{dt}y(t)|_{t=kT_s} \approx \frac{y(kT_s) - y([k-1]T_s)}{T_s} = \frac{y[k] - y[k-1]}{T_s}$$

- The Laplace-transform of the derivative is $\mathcal{L}(\dot{y}(t)) = sY(s)$

- The z-transform of its approximation is

$$\mathcal{Z}\left(\frac{d}{dt}y(t)|_{t=kT_s}\right) = \mathcal{Z}\left(\frac{y[k] - y[k-1]}{T_s}\right) = \frac{1 - z^{-1}}{T_s}Y(z)$$

$$Y(z) = Y(s)\big|_{s=\frac{1-z^{-1}}{T_s}}$$



How would integral approximation look like here?

Forward difference method

- Approximate the derivative by using the difference between the next and the current sample divided by the sampling period, i.e.,

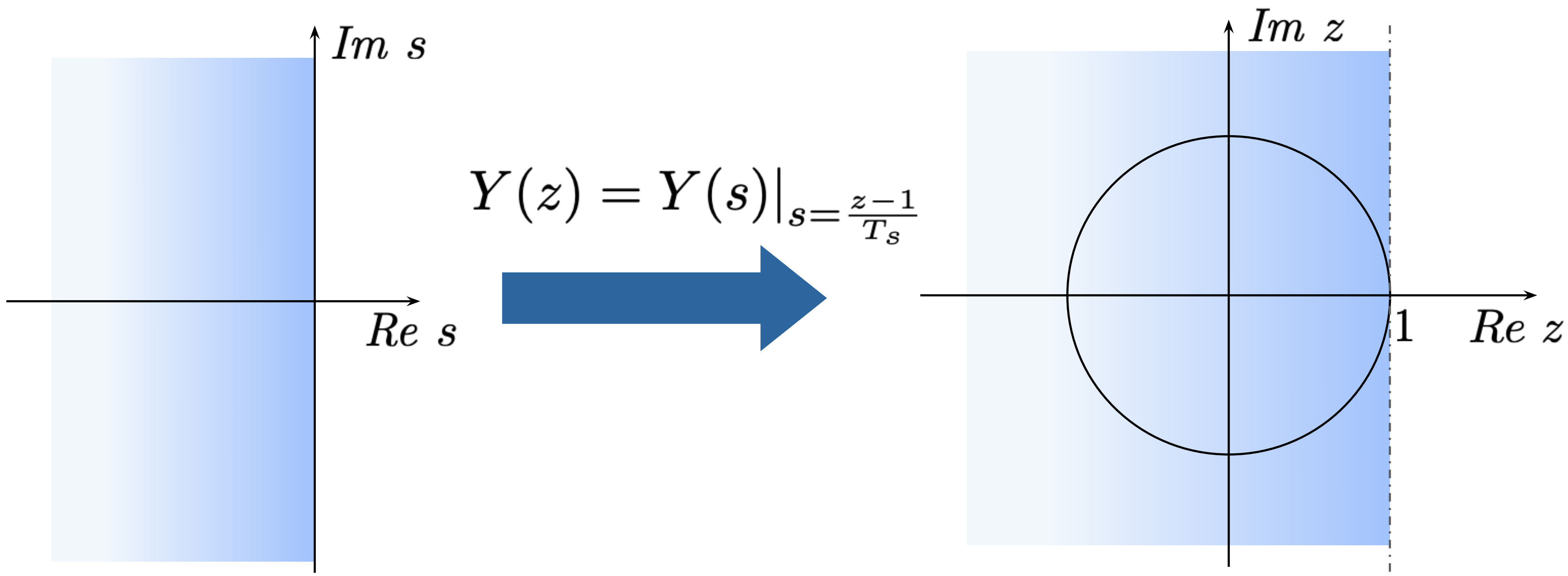
$$\frac{d}{dt}y(t)|_{t=kT_s} \approx \frac{y([k+1]T_s) - y(kT_s)}{T_s} = \frac{y[k+1]-y[k]}{T_s}$$

Causality?

- The Laplace-transform of the derivative is $\mathcal{L}(\dot{y}(t)) = sY(s)$

- The z-transform of its approximation is

$$\mathcal{Z}\left(\frac{d}{dt}y(t)|_{t=kT_s}\right) = \mathcal{Z}\left(\frac{y[k+1]-y[k]}{T_s}\right) = \frac{z-1}{T_s}Y(z)$$



How would integral approximation look like here?

Time-domain integrals

Why do we want to compute an integral in a practical controller?

- Let's consider the forward-difference approximation

$$\frac{d}{dt}y(t)|_{t=kT_s} \approx \frac{y([k+1]T_s) - y(kT_s)}{T_s} = \frac{y[k+1] - y[k]}{T_s}$$

- Solving for *y[k+1]* gives (incremental form)

$$y[k+1] \approx y[k] + T_s\dot{y}[k]$$

Note: this is the integral of d/dt y.

- This is Euler approximation of $\int_{kT_s}^{(k+1)T_s} \dot{y}(t)dt$
- Can also be written as a sum

$$\int_0^T \dot{y}(t)dt \approx \sum_{i=0}^{T/T_s-1} \dot{y}[i]$$

- Similarly, for *y*

$$\int_0^T y(t)dt \approx \sum_{i=0}^{T/T_s-1} y[i]$$

This is the integral of y.

- Incremental Euler approximation is commonly used in controller implementations, storing current value of the integral for the next iteration.

How about backward difference approximation?

Time-domain integrals

- Let's consider the backward-difference approximation

$$\frac{d}{dt}y(t)|_{t=kT_s} \approx \frac{y(kT_s) - y([k-1]T_s)}{T_s} = \frac{y[k] - y[k-1]}{T_s}$$

- Solving for *y[k+1]* gives (incremental form)

$$y[k+1] \approx y[k] + T_s \dot{y}[k+1]$$

Compare to fwd:

$$y[k+1] \approx y[k] + T_s \dot{y}[k]$$

- Called backward (or implicit) Euler approximation

- Can also be written as a sum

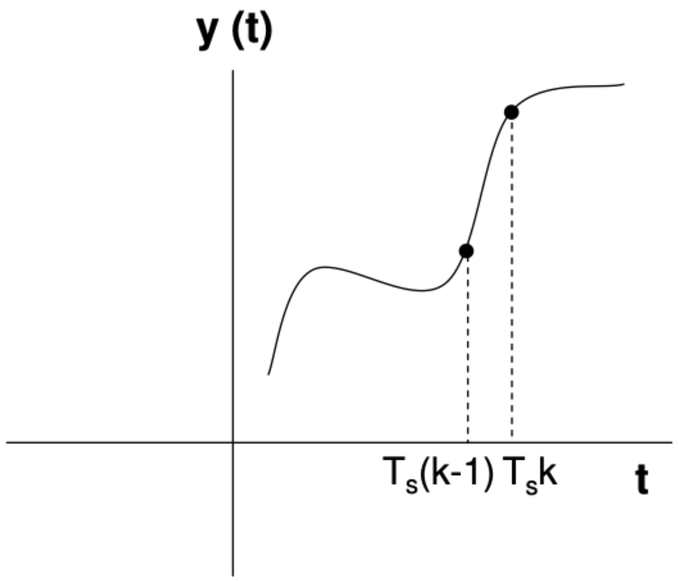
$$\int_0^T \dot{y}(t)dt \approx \sum_{i=1}^{T/T_s} \dot{y}[i]$$

- Computing backward/implicit Euler integral requires iterative solution of each time step (e.g. Newton method), because of the reliance on the future (k+1) value of the derivative.

How about higher-order differential equations?

Approximation of differential eqns by difference eqns

- Calculate the derivative by using the difference between the current and the previous sample divided by the sampling period, i.e.,



$$\frac{d}{dt}y(t)|_{t=kT_s} \approx \frac{y(kT_s) - y([k-1]T_s)}{T_s} = \frac{y[k] - y[k-1]}{T_s}$$

- The second derivative is therefore

$$\frac{d^2}{dt^2}y(t)|_{t=kT_s} = \frac{d}{dt}[\frac{d}{dt}y(t)]|_{t=kT_s} \approx \frac{\frac{d}{dt}y(t)|_{t=kT_s} - \frac{d}{dt}y(t)|_{t=(k-1)T_s}}{T_s}$$

$$= \frac{(y[k] - y[k-1])/T_s - (y[k-1] - y[k-2])/T_s}{T_s}$$

$$= \frac{1}{T_s{}^2}(y[k] - 2y[k-1] + y[k-2])$$

- In general…

$$\frac{d^n}{dt^n}y(t)|_{t=kT_s} \approx \frac{1}{T_s}^n \sum_{i=0}^{n}(-1)^i \binom{n}{i} y[k-i]$$

Impulse-invariance method

- The impulse response is given by

$$g(t) = \mathcal{L}^{-1}\left(G(s)\right)$$

- From $g(t)$, via discretization, we get $g(kT_s)$, where T_s is the sampling period

$$g[k] = g(kT_s)$$

- From $g(kT_s)$, using z-transform, we derive $G(z)$ in z-domain, i.e.,

$$G(z) = \mathcal{Z}\left(\mathcal{L}^{-1}\left(G(s)\right)_{t=kT_s}\right)$$

- Using this method, **frequency and step responses are not preserved**
- **The discrete system will be stable if the original analog system is stable**

Proof on next two slides

Impulse-invariant method: LTI systems

- The impulse response of an LTI system can be written as follows:

$$h(t) = \mathcal{L}^{-1}\left\{H(s)\right\} = \mathcal{L}^{-1}\left\{\frac{d_1}{s-\lambda_1} + \frac{d_2}{s-\lambda_2} + \ldots + \frac{d_n}{s-\lambda_n}\right\}$$
$$= d_1 e^{\lambda_1 t} + d_2 e^{\lambda_2 t} + \ldots + d_n e^{\lambda_n t}$$

- Each element's time response contains every mode of the system (although some coefficients may be negligible)

- After sampling: $h[k] = d_1 e^{\lambda_1 kh} + d_2 e^{\lambda_2 kh} + \ldots + d_n e^{\lambda_n kh}$

- Taking z-transforms

$$H(z) = \sum_{k=0}^{\infty} h[k] z^{-k} = \sum_{k=0}^{\infty}\left(\sum_{m=1} n d_m e^{\lambda_m kh}\right) z^{-k}$$
$$= \sum_{m=1}^{n} d_m \sum_{k=0}^{\infty}\left(e^{\lambda_m h} z^{-1}\right)^k$$
$$= \sum_{m=1}^{n} \frac{d_m}{1 - e^{\lambda_m h} z^{-1}}$$

Impulse-invariant method: LTI systems

- System $H(z)$ has n poles:

$$p_i = e^{\lambda_i h} = e^{(-\sigma_i + j\omega_i)h} = e^{-\sigma_i h} e^{j\omega_i h} \Rightarrow |p_i| = e^{-\sigma_i h} |e^{j\omega_i h}| = e^{-\sigma_i h} < 1$$

- Hence, the z-transform always projects a stable pole in the s-domain to a stable pole in the z-domain $\rightarrow$ the discrete system will be stable if the original analog system is stable

Skipped on lecture

Recap: Step-invariance method

- The step response is given by

$$g_{\text{step}}(t) = \mathcal{L}^{-1}\left(\frac{G(s)}{s}\right)$$

- The corresponding step response in discrete time is given by

$$g_{\text{step}}[k] = g_{\text{step}}(kT_s)$$

- The z-transform of $g_{\text{step}}[k]$ is given by

$$\mathcal{Z}\left(g_{\text{step}}[k]\right) = \frac{z}{z-1}G(z)$$

- The final transformation is thus given by

$$G(z) = \frac{z-1}{z}\mathcal{Z}\left(\mathcal{L}^{-1}\left(\frac{G(s)}{s}\right)_{t=nT_s}\right)$$

- **Neither the frequency nor the impulse responses are preserved**

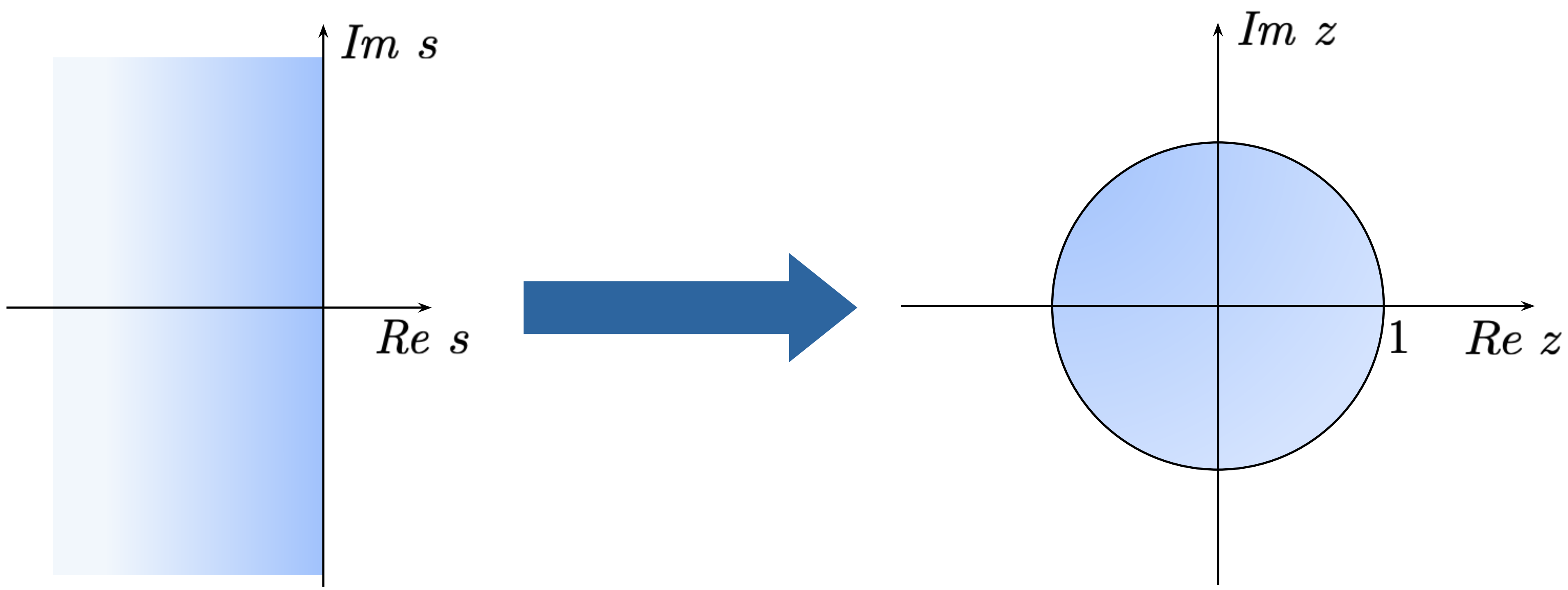
Bilinear or Tustin method

- Using this method, the conversion of $G(s)$ in z-domain is

$$Y(z) = Y(s)\big|_{s=\frac{2}{T_s}\frac{z-1}{z+1}}$$

- Represents one of the most popular methods because the stability of the analog system is preserved

- This transformation has a unique mapping between the s-domain and the z-domain, as shown below:



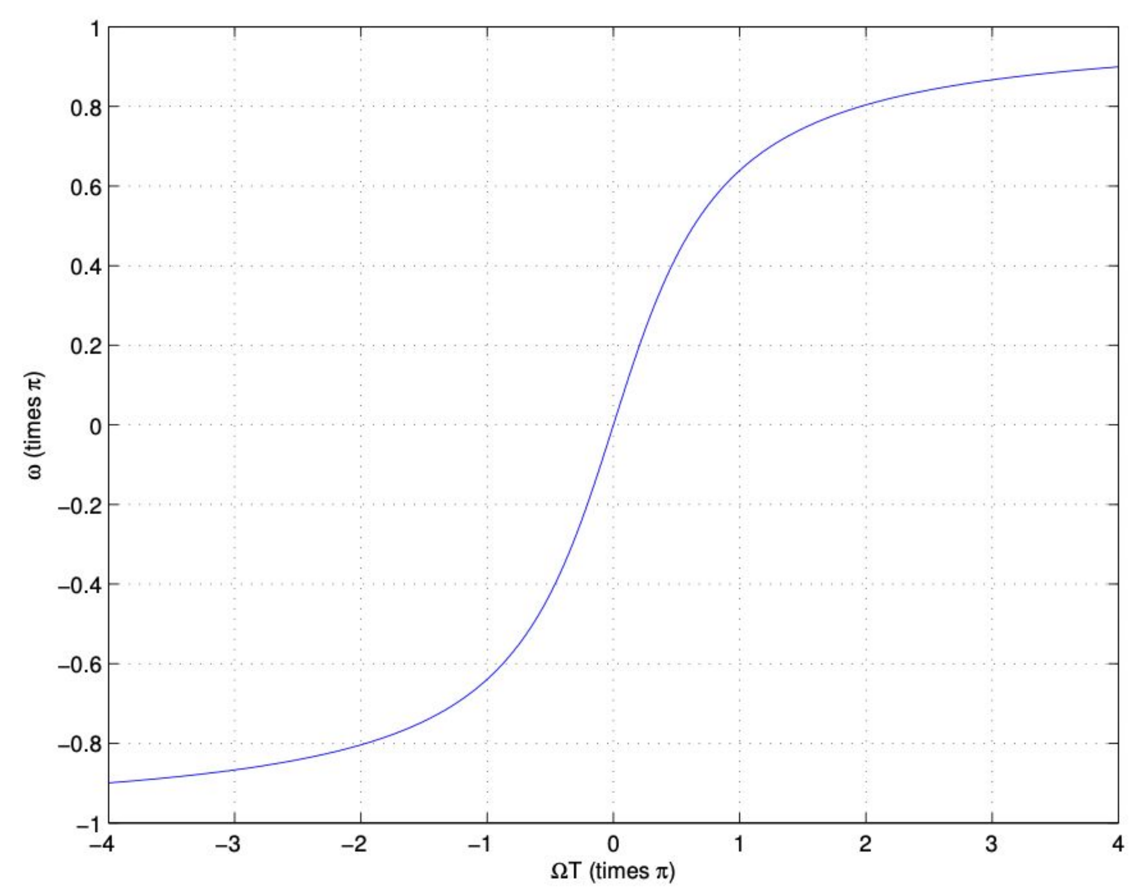
Corresponds to trapezoid integration

Bilinear or Tustin method: Frequency response

- What is the relationship between $H_c(j\Omega)$ and $H(e^{j\varpi})$?

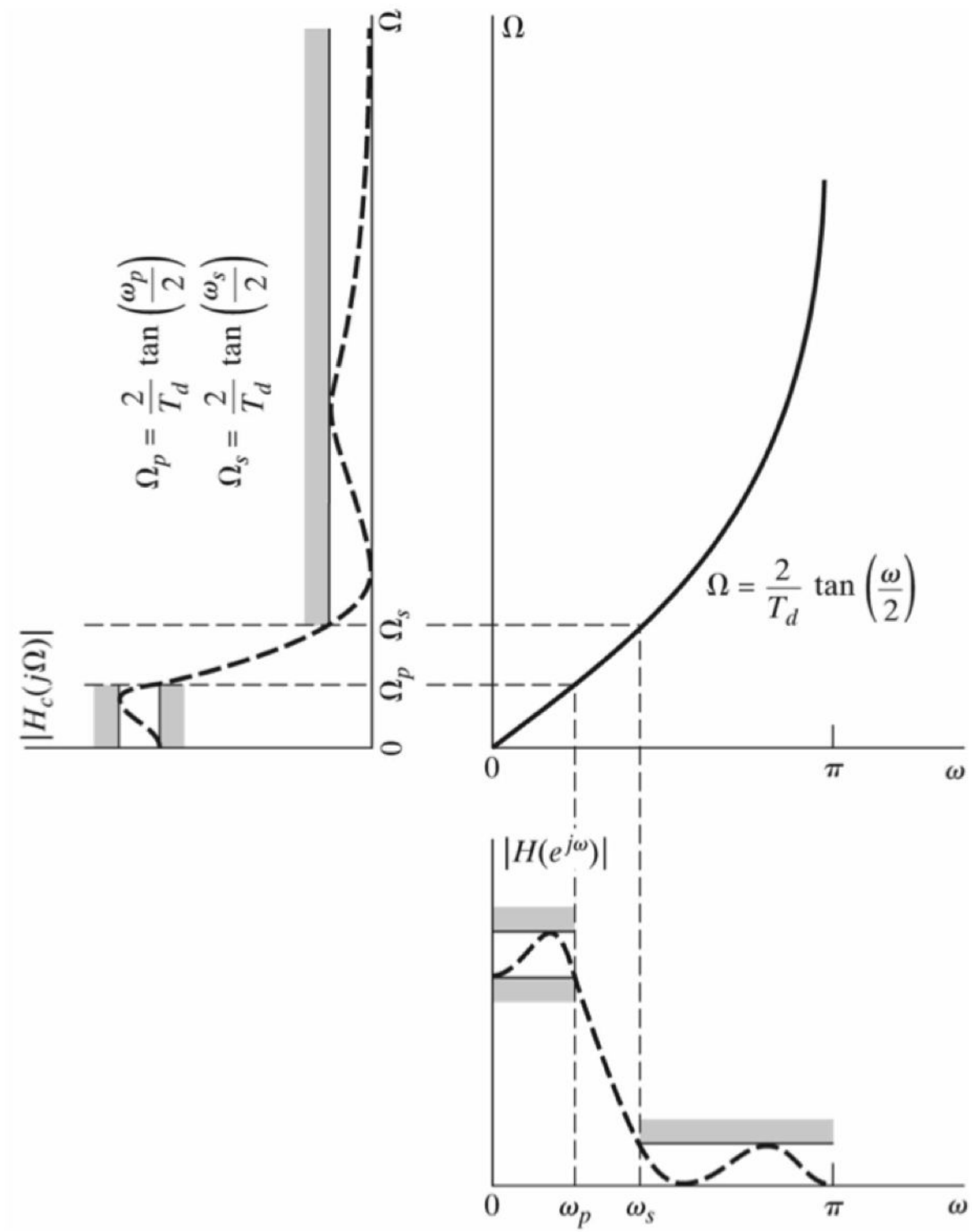
Recall that $H_c(j\Omega) = H_c(s)\big|_{s=j\Omega}$ and $H(e^{j\omega}) = H(z)\big|_{z=e^{j\omega}}$. Substituting these into the bilinear transform formula, we get:

$$
\begin{aligned}
j\Omega &= \frac{2}{T_s}\frac{e^{j\omega}-1}{e^{j\omega}+1} \\
&= \frac{2}{T_s}\frac{e^{j\omega/2}-e^{-j\omega/2}}{e^{j\omega/2}+e^{-j\omega/2}} \\
&= \frac{2j}{T_s}\frac{\frac{1}{2j}\left(e^{j\omega/2}-e^{-j\omega/2}\right)}{\frac{1}{2}\left(e^{j\omega/2}+e^{-j\omega/2}\right)} \\
&= \frac{2j}{T_s}\frac{\sin(\omega/2)}{\cos(\omega/2)} \Rightarrow \boxed{\Omega = \frac{2}{T_s}\tan(\omega/2)}
\end{aligned}
$$



This nonlinear relationship is called "frequency warping".

Bilinear or Tustin method: Frequency response



- The good news is that we don't have to worry about aliasing.

- The "bad" news is that we have to account for frequency warping when we start from a discrete-time filter specification.

Remark: direct design based on specifications

- A rational z-transform can be written as

$$H(z) = c\frac{(z-\mu_1)(z-\mu_2)\ldots(z-\mu_M)}{(z-p_1)(z-p_2)\ldots(z-p_N)}$$

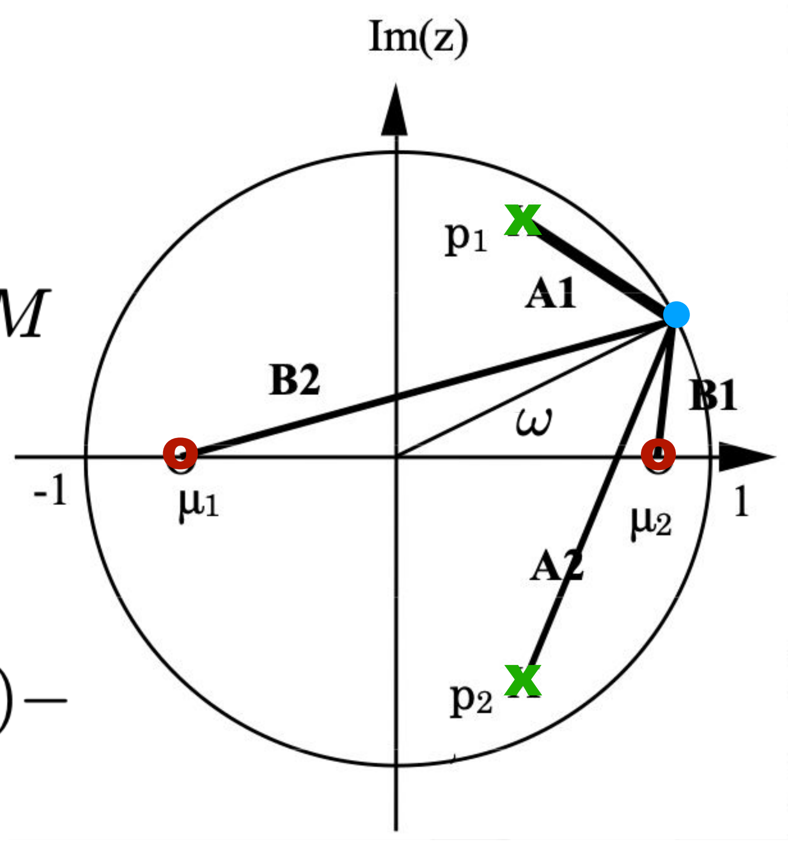
- To compute the frequency response of $H(e^{j\omega})$, we compute $H(z)$ at $z = e^{j\omega}$

- But $z = e^{j\omega}$ represents a point on the circle's perimeter

- Let $A_i = |e^{j\omega} - p_i|$ and $B_i = |e^{j\omega} - \mu_i|$

$$|H(e^{j\omega})| = |c|\frac{|e^{j\omega}-\mu_1|\ |e^{j\omega}-\mu_2|\ldots|e^{j\omega}-\mu_M|}{|e^{j\omega}-p_1|\ |e^{j\omega}-p_2|\ldots|e^{j\omega}-p_N|}, \qquad N \geq M$$

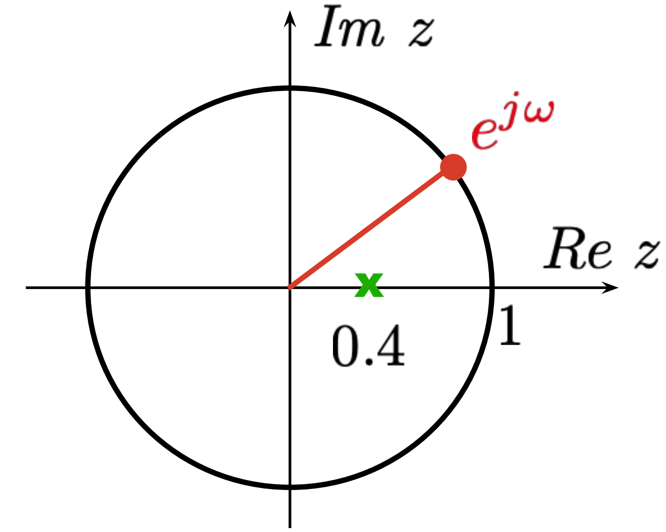
$$= |c|\frac{B_1 B_2 \ldots B_M}{A_1 A_2 \ldots A_N}$$

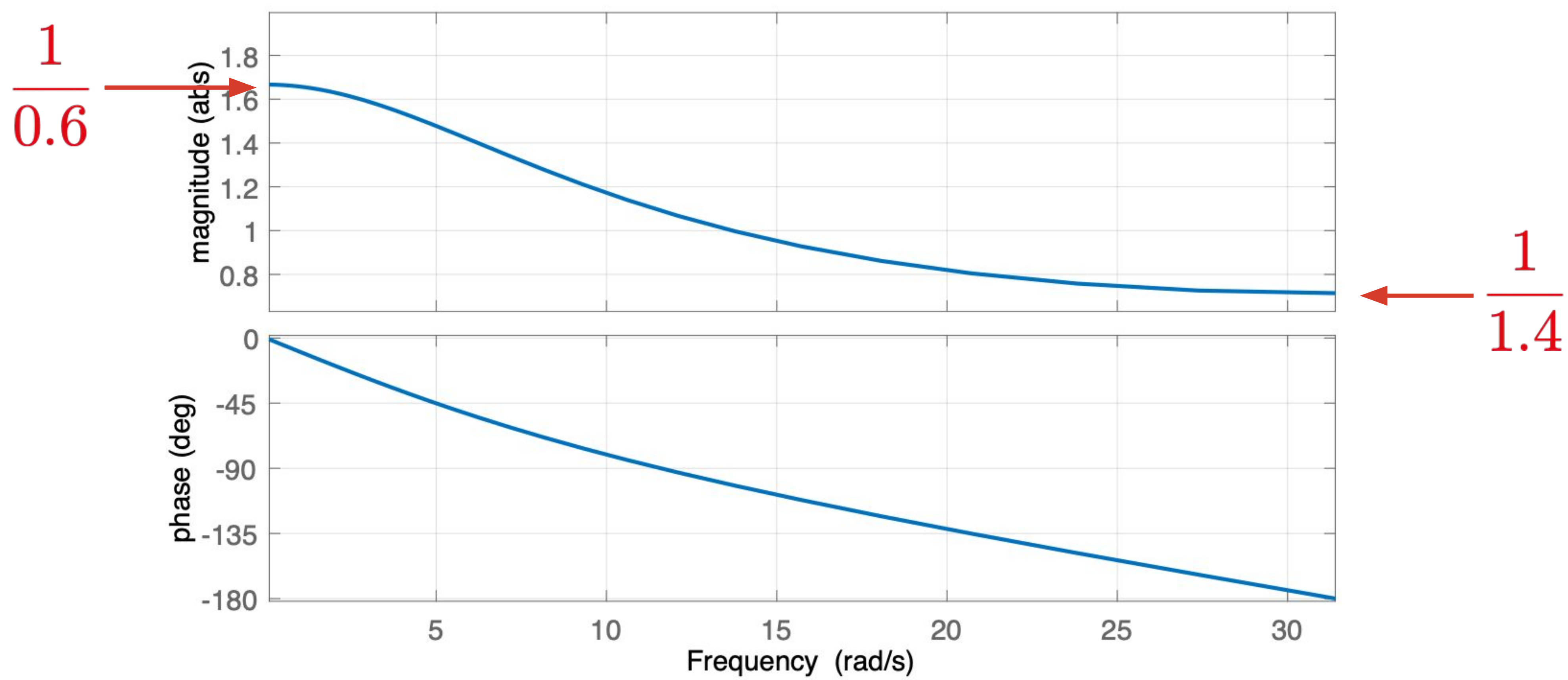


$$\angle H(e^{j\omega}) = \angle(e^{j\omega}-\mu_1) + \angle(e^{j\omega}-\mu_2) + \ldots + \angle(e^{j\omega}-\mu_M) - \\ - \angle(e^{j\omega}-p_1) - \angle(e^{j\omega}-p_2) - \ldots - \angle(e^{j\omega}-p_N)$$

Example

- Consider the low-pass filter $H(z) = \dfrac{1}{z - 0.4}$






```
num = 1;                % numerator
den = [ 1  -0.4 ];      % denominator
H = tf(num,den,0.1);    % discrete-time transfer function with sampling time 0.1s
bode(H)                 % plots the bode plot
```

Learning outcomes

By the end of *this* lecture, you should be able to:

- Understand what happens to the signal when sampling
- What is the sampling frequency so that one can reconstruct the signal
- Consider the options of discretization in control systems
- Use discretization methods for designing discrete-time systems