Digital Signal Processing

Introduction of IIR filter

Zhuoqi Cheng
zch@mmmi.sdu.dk
SDU Robotics

SDU

Index

Two categories of discrete-time filters

Digital filters are divided into two categories according to their impulse response:

Infinite Impulse Response filters (IIR filters)	Finite Impulse Response filters (FIR filters)
$y(n) = \sum_{i=0}^{N} a_i\, x(n-i) - \sum_{i=1}^{N} b_i\, y(n-i)$	$y(n) = \sum_{i=0}^{N} a_i\, x(n-i)$
$H(z) = \frac{\sum_{i=0}^{N} a_i\, z^{-i}}{1 + \sum_{i=1}^{N} b_i\, z^{-i}}$	$H(z) = \sum_{i=0}^{N} a_i\, z^{-i}$

→ An FIR filter has 5 to 10 times larger realization structure than a corresponding IIR filter.
→ An FIR filter is always stable.
→ An FIR filter is called a **non-recursive structure**, while an IIR filter is called a **recursive structure**.
→ An FIR filter is **less sensitive** to coefficient changes and rounding errors than an IIR filter.

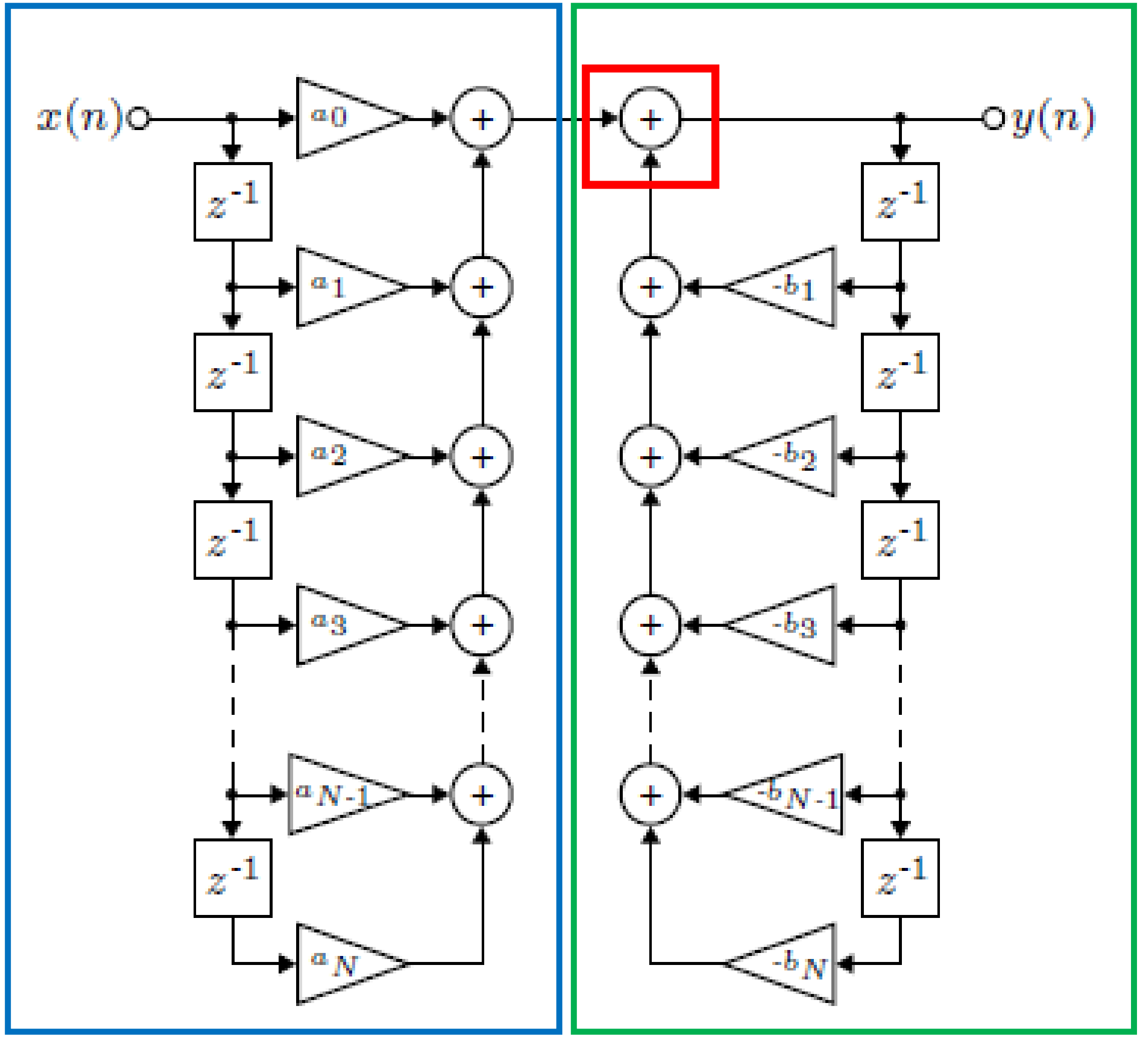
Recap: realization

Transfer function

$$H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{i=0}^{N} a_i\, z^{-i}}{1 + \sum_{i=1}^{N} b_i\, z^{-i}}$$

Difference equation

$$y(n) = \sum_{i=0}^{N} a_i x(n-i) - \sum_{i=1}^{N} b_i y(n-i)$$



SDU

Design of digital IIR filter

Digital IIR filters can be designed by transforming prototype filters in the s-domain using the following methods

→ **Matched z-transformation**
→ **Impulse invariant z-transformation**
→ Bilinear z-transformation (more popular, next lecture)

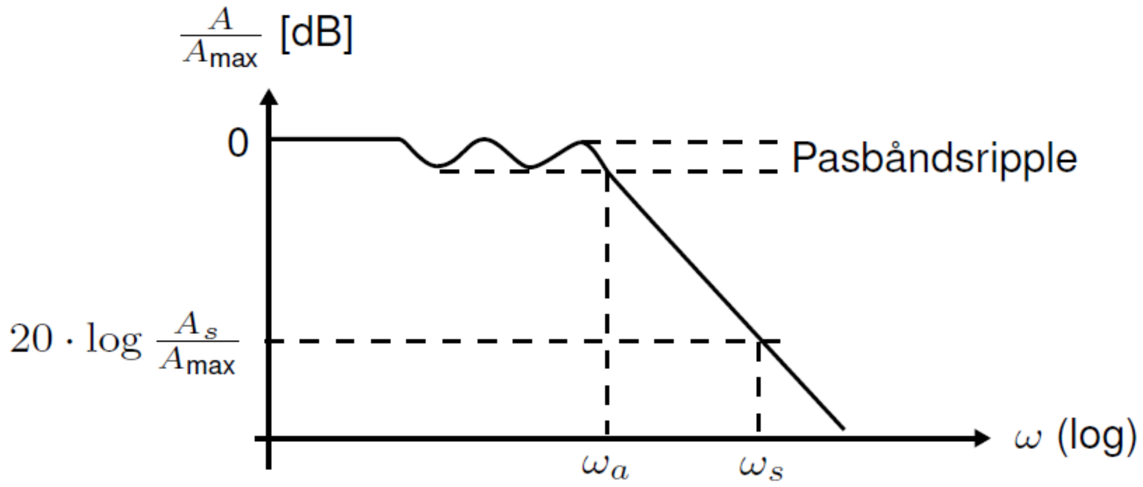
Design of digital IIR filter

An IIR filter is designed by following the procedure

1. The filter's specifications are drawn up (analog filter)
2. Convert the analog filter to digital filter: z-domain transfer function is made
3. Choose a realization structure
4. Implement the design through program or hardware

Recap - Filter specification

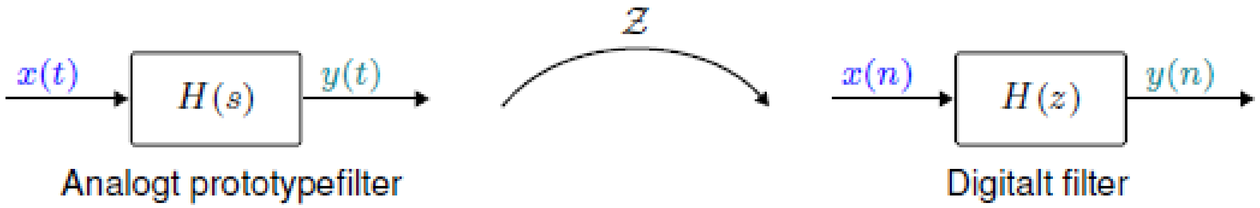
1. A **cutoff frequency ω_a** which indicates the upper limit of the passband.
2. A **stopband frequency ω_s** at which a **stopband attenuation A_s** is specified.
3. Information of passband ripple.



Transformation from an analog prototype filter

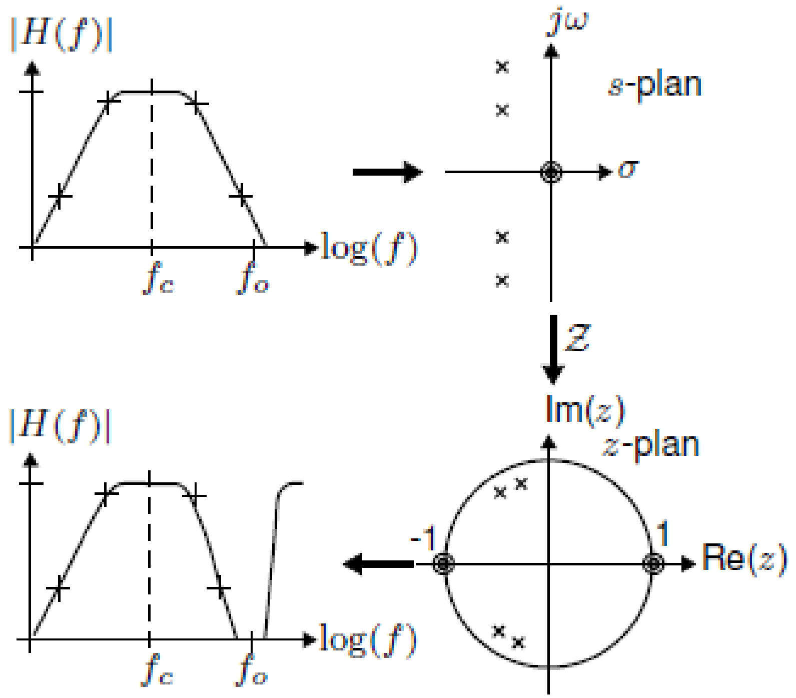
$$H(s) = \frac{\sum_{i=0}^{M} A_i s^i}{\sum_{i=0}^{N} B_i\, s^i}$$

$$H(z) = \frac{\sum_{i=0}^{M} a_i z^{-i}}{\sum_{i=0}^{N} b_i\, z^{-i}}$$



Transformation from an analog prototype filter

The transformation from s-domain to z-domain:
Transfer the **poles** and **zeros** from the s-plane to the z-plane. This will give a similar filter response.



Matched z-transform

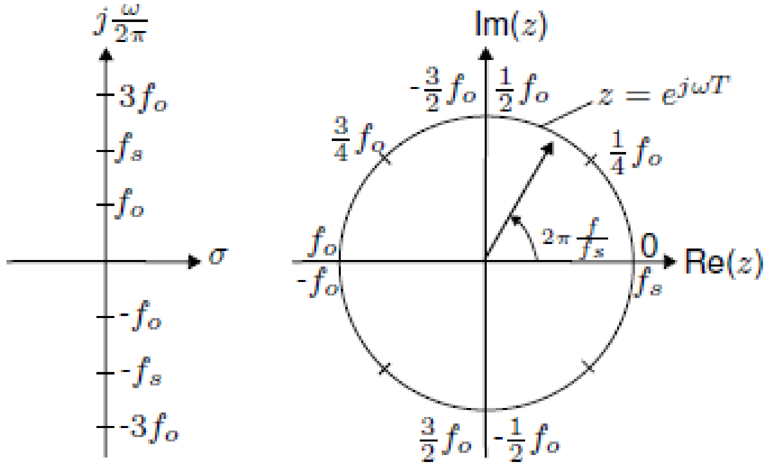
With matched z-transform, the **poles and zeros** of the prototype filter are transferred directly to the z-domain using the following

$$X_s(s) = X(z) \ \text{ when } \boxed{z = e^{sT}}$$

Same equation used previously to relate the z-plane and the s-plane.

Laplace transform of **sequence $\boldsymbol{x(n)}$**

$$X_s(s) = \sum_{n=0}^{\infty} x(n) e^{-snT}$$



z transform of **sequence $\boldsymbol{x(n)}$**

$$X(z) = \sum_{n=0}^{\infty} x(n) z^{-n}$$

s domain

z domain

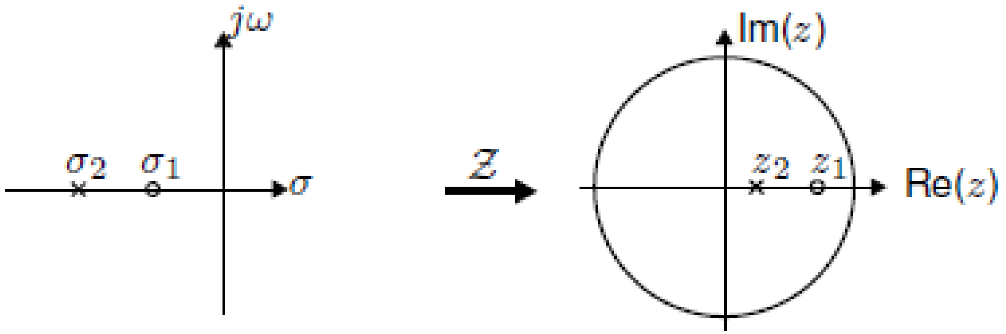
Matched z-transform for 1st order system

A 1st order transfer function with 1 zero is given by

$$H(s) = \frac{s + A_0}{s + B_0} = \frac{s - \sigma_1}{s - \sigma_2}$$

Where $\sigma_1 = -A_0$ is a real **zero** and $\sigma_2 = -B_0$ is a real **pole**

When using matched z-transformation, the **zero** and the **pole** can be calculated as
$z_1 = e^{s_1 T} = e^{\sigma_1 T}$ and $z_2 = e^{s_2 T} = e^{\sigma_2 T}$



Matched z-transform for 1st order system

A digital first-order filter based on matched z-transform has the form

$$H(z) = \frac{z - e^{\sigma_1 T}}{z - e^{\sigma_2 T}}$$

$$H(z) = \frac{1 - e^{\sigma_1 T} z^{-1}}{1 - e^{\sigma_2 T} z^{-1}}$$

Matched z-transform for 1st order system

Let's consider a 1st order system with no zero

$$H(s) = \frac{\omega_a}{s + \omega_a}$$

The above has only 1 pole, and it can be converted to the following

$$H(z) = \frac{1}{1 - e^{\sigma_1 T} z^{-1}}$$

where σ_1 is the pole of $H(s)$ and T is the sampling interval.

There is a problem above:
Try to substitute $z = 1$

$$H(1) = \frac{1}{1 - e^{\sigma_1 T}} \neq 1$$

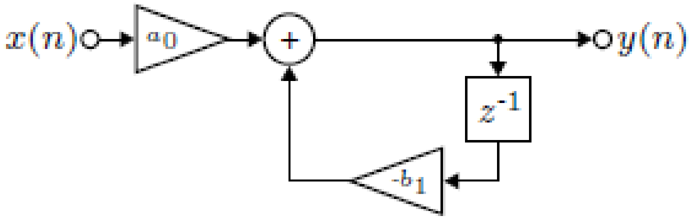
What does $z = 1$ mean?
$z = 1\, e^{j\omega T}$ where $\omega = 0$
It represents the DC gain!
It corresponds to the origin of the s-plane.
$H(s = 0) = 1$.

Matched z-transform for 1st order system

To solve this problem, we introduce an **amplification factor**

$$H(z) = \frac{1 - e^{\sigma_1 T}}{1 - e^{\sigma_1 T} z^{-1}}$$

A digital realization structure for the first order system is shown here.



Example

Given an analog 1st order lowpass filter

$$H(s) = \frac{\omega_a}{s + \omega_a} = \frac{1885}{s + 1885}$$

Whose cutoff frequency $\omega_a = 2\pi f_a$ and $f_a = 300$Hz.

It has **1 pole**: $s_1 = \sigma_1 = -1885$

We want to determine a digital filter by **matched z-transformation** with a sampling frequency of f_s =16 kHz.
The coefficients of the digital filter become

The original s-transfer function has no zeros, so we use the following

$$H(z) = \frac{1 - e^{\sigma_1 T}}{1 - e^{\sigma_1 T} z^{-1}}$$

$$1 - e^{\sigma_1 T} = 0.1111$$
$$e^{\sigma_1 T} = 0.8889$$

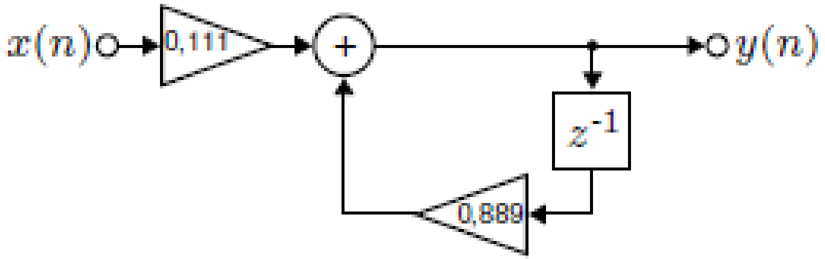
The digital low-pass filter transfer function is

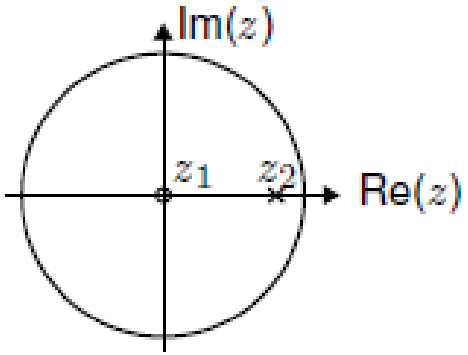
$$H(z) = \frac{0.1111}{1 - 0.8889\, z^{-1}}$$

SDU

Example

The digital filter $H(z)$ has zero $z = 0$ and pole $z = 0.8889$
The direct type II realization structure is shown below.

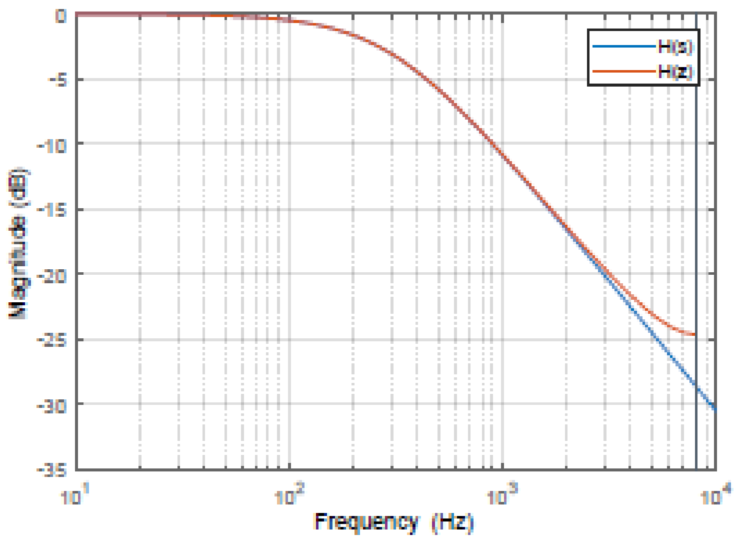




Although the transfer function in s-domain has no zeros, the converted transfer function in z-domain using matched z-transform has 1 zero.

Example

When comparing the amplitude characteristics of the two filters, you can find that they differ at high frequencies (close to the **Nyquist frequency** f_o).



Due to this inherent limitation, the matched z-transform method normally is used when $f_a \ll f_o$

SDU

Design procedure (matched z-transform)

1. Determine the analog filter's transfer function $H(s)$.
2. Determine the analog filter’s poles and zeros.
3. Convert the **poles** in s-domain to z-domain
4. Determine the coefficients of the digital transfer function. The numerator may be modified to have $H(z = 1) = 1$
5. Implement the transfer function as a **cascade structure**.

Matched z-transform for higher order system

For a higher order system with complex conjugate pole and zero pair, the transfer function can be written after factorization

$$H(s) = \frac{(s-z_1)(s-z_1^*)}{(s-p_1)(s-p_1^*)} \cdot \frac{(s-z_2)(s-z_2^*)}{(s-p_2)(s-p_2^*)} \cdot \dots \cdot \frac{(s-z_N)}{(s-p_N)}$$

By converting ‘s’ -> ‘z’, and replacing s_i with $\boxed{z = e^{sT}}$

$$H(z) = \frac{\quad}{(z-e^{p_1 T})(z-e^{p_1^* T})} \cdot \frac{\quad}{(z-e^{p_2 T})(z-e^{p_2^* T})} \cdot \dots \cdot \frac{\quad}{z-e^{p_N T}}$$

Determine the zeros for $H(z)$ so that $H(z=1)=1$

Normally, a **cascade structure** is used for the realization when matched z transform method is used.

Example: 2nd order system

For a 2nd order system with complex conjugate pole and zero pair, the transfer function can be written

$$H(s) = \frac{s^2 + A_1 s + A_0}{s^2 + B_1 s + B_0}$$

Which have **zeros** $s_1 = \sigma_1 + j\omega_1,\ s_1^* = \sigma_1 - j\omega_1,$ and **poles** $s_2 = \sigma_2 + j\omega_2,\ s_2^* = \sigma_2 - j\omega_2$

$$H(s) = \frac{(s - s_1)(s - s_1^*)}{(s - s_2)(s - s_2^*)}$$

By converting 's' -> 'z', and replacing s_1, s_2 with $\boxed{z = e^{sT}}$

$$H(z) = \frac{(z - e^{\sigma_1 T} e^{j\omega_1 T})(z - e^{\sigma_1 T} e^{-j\omega_1 T})}{(z - e^{\sigma_2 T} e^{j\omega_2 T})(z - e^{\sigma_2 T} e^{-j\omega_2 T})}$$

Using Euler formula $\cos(x) = (e^{jx} + e^{-jx})/2$, we have

$$H(z) = \frac{z^2 - (2e^{\sigma_1 T}\cos(\omega_1 T))z + e^{2\sigma_1 T}}{z^2 - (2e^{\sigma_2 T}\cos(\omega_2 T))z + e^{2\sigma_2 T}} = \frac{1 - (2e^{\sigma_1 T}\cos(\omega_1 T))z^{-1} + e^{2\sigma_1 T} z^{-2}}{1 - (2e^{\sigma_2 T}\cos(\omega_2 T))z^{-1} + e^{2\sigma_2 T} z^{-2}}$$

Example: 2nd order system

A 2nd order low-pass filter is given as

$$H(z) = \frac{a_0}{1 + b_1 z^{-1} + b_2 z^{-2}}$$

From the previous slide we know

$$\boxed{H(z) = \frac{1 - (2e^{\sigma_1 T}\cos(\omega_1 T))z^{-1} + e^{2\sigma_1 T} z^{-2}}{1 - (2e^{\sigma_2 T}\cos(\omega_2 T))z^{-1} + e^{2\sigma_2 T} z^{-2}}}$$

In this case,

$$b_1 = -2e^{\sigma_2 T}\cos(\omega_2 T) \quad \text{and} \quad b_2 = e^{2\sigma_2 T}$$

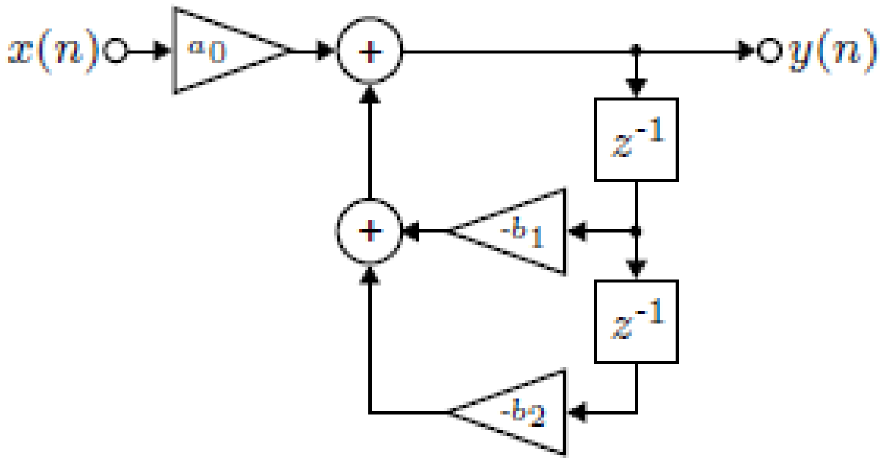
If the **DC gain** for the low-pass filter is to be 0 dB then it applies that

$$a_0 = 1 + b_1 + b_2$$

We can change the zeros positions during the conversion.
We need to force $H(z = 1) = 1$

Example: 2nd order system

The realization structure for the low-pass filter is given as shown below.



Exercise

Design a digital 2nd order Butterworth low pass filter with cutoff frequency $f_a = 800$Hz using Matched z-transform with sampling rate of 8 kHz.

A frequency-normalized transfer function for a 2nd order Butterworth low-pass filter with cutoff frequency $f_a = 800$ Hz is

$$H(s) = \frac{2.527e07}{s^2 + 7109\, s + 2.527e07}$$

Matlab:
[b,a] = butter(2,2*pi*fa,'s');
H_s = tf(b,a)

It has two poles $s = -3554.5 \pm 3554.6i$

The filter coefficients can now be calculated using the formulas

$b_1 = -2e^{\sigma_2 T}\cos(\omega_2 T)$ and $b_2 = e^{2\sigma_2 T}$ and $a_0 = 1 + b_1 + b_2$

Matlab: to find zeros and poles
- roots()
- pzmap()

Can you calculate the final discrete-time transfer function**?**

$$H(z) = \frac{a_0}{1 + b_1 z^{-1} + b_2 z^{-2}}$$

Matlab function: Hz=c2d(Hs, Ts, method)

→ Create the required continuous-time transfer function with required specs: Hs
→ Determine the sampling period Ts
→ Method can be
  → **'matched' — Zero-pole matching method**
  → 'impulse' — Impulse invariant discretization
  → 'tustin' — Bilinear (Tustin) method.

```
fa = 800;
[b,a] = butter(2,2*pi*fa,'s');
H_s = tf(b,a)

fs = 8000;
H_z = c2d(H_s, 1/fs, 'matched')

>> H_z =
    0.1266 z + 0.1266
  ----------------------
  z^2 - 1.158 z + 0.4112
```

$H(z=1)=1$ applies as well!

Example

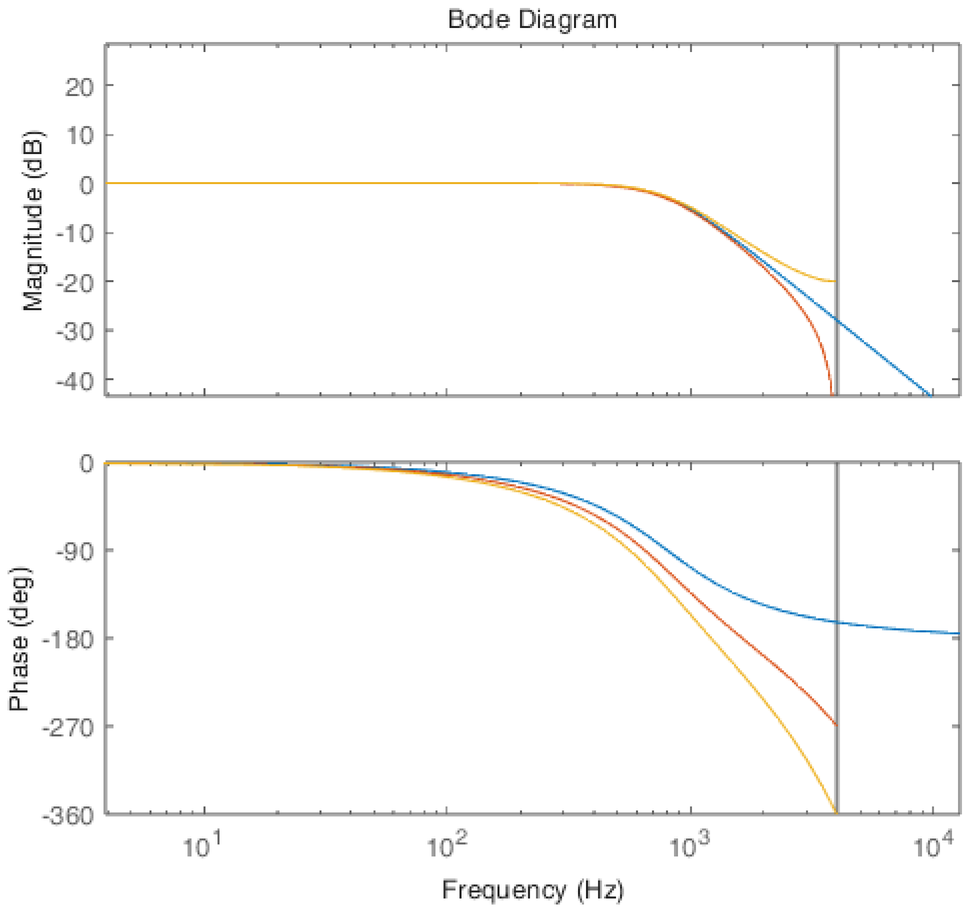
You can see that the digital filter is a very aliased version of the original analog filter $H(s) = \frac{2.527e07}{s^2 + 7109\,s + 2.527e07}$ (blue).

Matlab calculated (red)

$$H_z = 0.1266\,\frac{z+1}{z^2 - 1.158\,z\; +\; 0.4112}$$

Manual calculated (yellow)

$$H_z = \frac{0.2532}{z^2 - 1.158\,z\; +\; 0.4112}$$



Getting worse when f closer to f_0

SDU

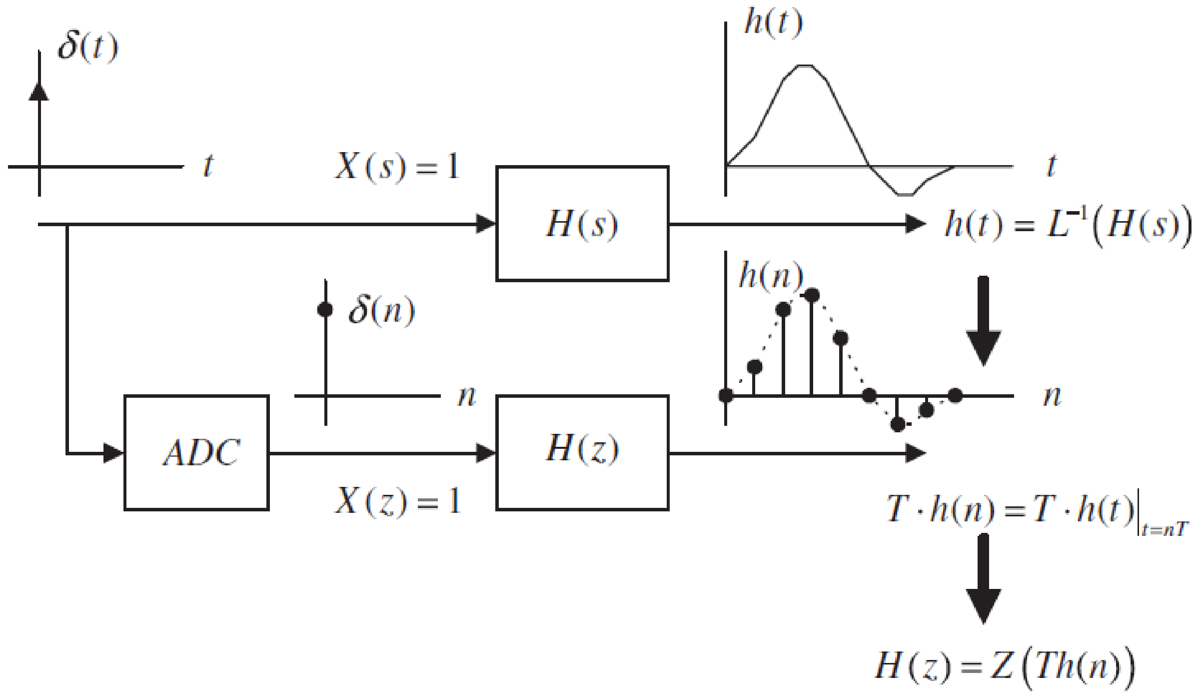
Impulse invariant z-transform

The analog impulse response can be achieved by using inverse Laplace transform of analog filter $H(s)$.

$$h(t) = \mathcal{L}^{-1}[H(s)]$$

Then we sample this analog impulse response with a sampling interval of T. Also, T is used as a scale factor.

$$T \cdot h(n) = T \cdot h(nT), \quad n \geq 0$$



Impulse invariant z-transform

Given a Nth order filter

$$H(s) = \frac{\sum_{i=0}^{M} A_i s^i}{\sum_{i=0}^{N} B_i s^i} \qquad M \leq N$$

Using partial fraction solution

$$H(s) = \sum_{i=1}^{N} \frac{k_i}{s - s_i}$$

Matlab function:
residue()

where s_i is the analog filter's pole

Impulse invariant z-transform

The transfer function

$$H(s) = \sum_{i=1}^{N} \frac{k_i}{s - s_i}$$

Using inverse Laplace transformed

$$h(t) = \mathcal{L}^{-1}[H(s)] = \sum_{i=1}^{N} k_i e^{s_i t}$$

The impulse response of the digital filter thus becomes

$$h(n) = h(nT) = \sum_{i=1}^{N} k_i e^{s_i nT}$$

By z-transform of $h(n)$

$$\boxed{H(z) = T\sum_{i=1}^{N} k_i \frac{z}{z - e^{s_i T}} = T\sum_{i=1}^{N} \frac{k_i}{1 - e^{s_i T} z^{-1}}}$$

ZT5	$e^{s_0 nT}$	$\frac{z}{z-e^{s_0 T}}$

Example

A transfer function for a 2nd order Butterworth low-pass filter with cutoff frequency $f_a = 800$ Hz

$$H(s) = \frac{2.527e07}{s^2 + \ 7109\, s + 2.527e07}$$

It has two poles $s = -3554.5 \pm 3554.6i$

Using partial fraction decomposition (Matlab residue()), $H(s)$ can be written

$$H(s) = \sum_{i=1}^{N} \frac{k_i}{s - s_i} = -\frac{j3554.5}{s + 3554.5 - 3554.6} + \frac{j3554.5}{s + 3554.5 + 3554.6j}$$

According to

$$\boxed{H(z) = T \sum_{i=1}^{N} \frac{k_i}{1 - e^{s_i T} z^{-1}}}$$

Sampling interval $T = 1/8k$

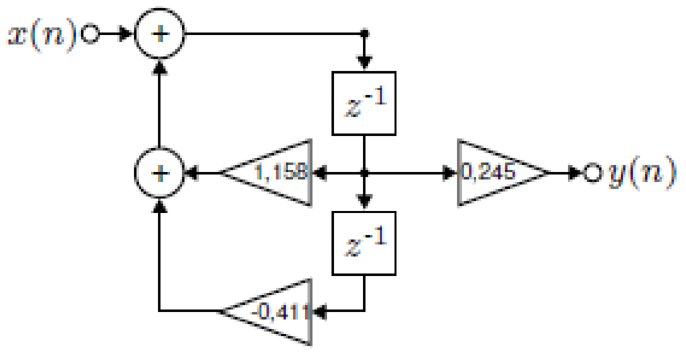
$$H(z) = \frac{-0.4443j}{1 - (0.579 + 0.2757j)z^{-1}} + \frac{0.4443j}{1 - (0.579 - 0.2757j)z^{-1}}$$

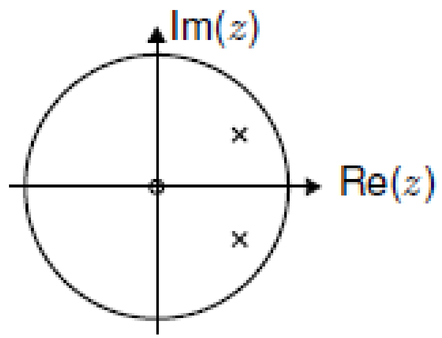
Realization

For IIR using Impulse Invariant z transform method, normally **parallel realization** is used.

For instance,

$$H(z) = \frac{-0.4443j}{1-(0.579+0.2757j)z^{-1}} + \frac{0.4443j}{1-(0.579-0.2757j)z^{-1}}$$



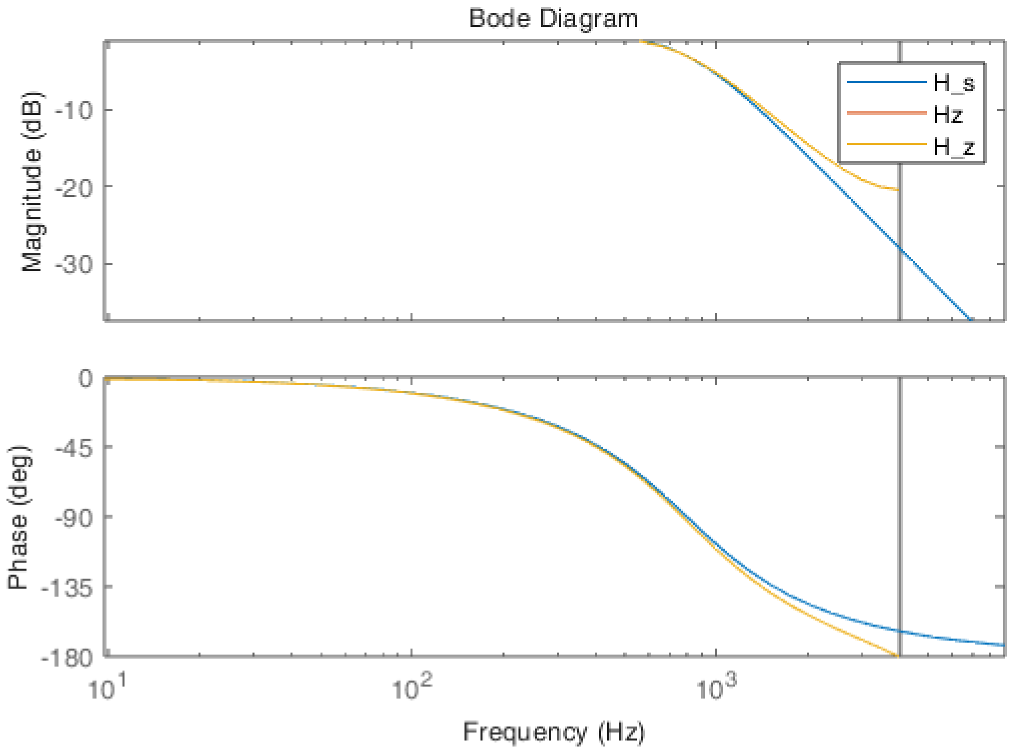


Matlab

```
fa = 800;
[b,a] = butter(2,2*pi*fa,'s');
H_s = tf(b,a)

fs = 8000;
H_z = c2d(H_s, 1/fs, ‘impulse’)

>> H_z =
        0.2449 z
  ----------------------
  z^2 - 1.158 z + 0.4112
```



Blue: H(s)
Red: H(z) using equation
Yellow: H(z) using Matlab c2d()

Exercise

Use Impulse invariant z transform method to derive a 4th order Butterworth bandpass filter, cutoff frequency $f_{a1} = 300Hz$, $f_{a2} = 800Hz$, sampling frequency $f_s = 8000Hz$.

1. Find the analog filter transfer function
   → [b,a] = butter(4,[2*pi*300, 2*pi*800],'bandpass','s')
   → H_s = tf(b, a)
2. Partial fraction decomposition
   → [r,p,k]= residue(b,a);
3. Using $H(z) = T\sum_{i=1}^{N}\frac{k_i}{1-e^{s_i T}z^{-1}}$, find the transfer function
4. Do a bode() plot to check if all the specs are satisfied
5. Try to draw the block diagram realization

(option) Try using function c2d() in place of step 2 to 3, and compare results.