12 Z-transform

12.1 FT, DTFT and DFT: summary

1. There is a difference in amplitude of FT and DTFT for continuous and discrete function of

$$X_s(\Omega) = \frac{1}{T_s} \sum_{n=-\infty}^{\infty} X(\Omega - n\Omega_s)$$

where discrete frequency is connected to the "real" frequency of a unsamples signal Ω by: $\omega = \Omega T_s$. As a consequence one can calculate FT of unsampled signal from DTFT (or DFT) by noticing that

$$T_s X_s(\Omega) = \sum_{n=-\infty}^{\infty} X(\Omega - n\Omega_s)$$

and by selecting only one period of the DTFT we get

$$X(\Omega) = T_s X_s(\Omega)$$

2. DFT in matlab gives result with discrete frequency on the x-axis. Matlab gives results with $\omega \in (0, 2\pi)$ and one need to use `fftshift` to plot it for $\omega \in (-\pi, \pi)$. Discrete frequency is related to frequency Ω by

$$\Omega = \frac{\omega}{T_s}$$

Since $\omega \in [-\pi, \pi)$ corresponds to $\Omega \in [-\pi/T_s, \pi/T_s)$ or $\Omega \in [-\Omega_s/2, \Omega_s/2)$

3. DFT in matlab is a vector which has the same length as the length of the input vector, since DFT is defined by:

$$X[k] = \sum_{n=0}^{N-1} x[n]\, e^{-j2\pi nk/N} \quad 0 \leq k \leq N-1$$

Where input signal is defined for $0 \leq n < N$ and the transform is defined at frequency $\omega = 2\pi k/N$ with $0 \leq k < N$.

4. The difference equation is a formula for computing an output sample at time n based on past and present input samples (x[n], x[n-1], x[n-2],..) and past output samples (y[n-1], y[n-2], ...) in the time domain.

5. in input starts at particular time point n, then one can use the difference equation directly to calculate the output (see Example 37).

$$\sum_{n=0}^{N} a_n y[k-n] = \sum_{n=0}^{N} b_n x[k-n]$$

12.2 So far

- we have shown that time discrete systems must be described by difference and not differential equations
- summation/subtraction of time-delayed signal must be used to implement integration/differentiation
- digital signal and can be represented as a sum of time-shifted weighted delta function
- difference equation allows us to find impulse response and we can use convolution sum to find system response to any time discrete input signal
- DTFT is a Fourier transform of sampled signal and DFT is a frequency discrete version of DTFT calculated for signal with length $N < \infty$
- Frequency space for DTFT and DFT is $-\pi \leq \omega < \pi$ and the real frequency is conected with ω through sampling time: $\Omega = \frac{\omega}{T_s}$

◇ **Example 40.** Laplace Transform for time discrete system
We have a digital network with corresponding differential and difference equations

$$\begin{aligned} y(t) &= x(t) + 0.5y(t - t_s) \\ y[n] &= x[n] + 0.5y[n-1] \end{aligned}$$

From the differential equation we can find out the impulse response:

$$\begin{aligned} h(t) &= \delta(t) + 0.5h(t - t_s) \\ H(s) &= 1 + 0.5H(s)e^{-st_s} \\ H(s) = \frac{1}{1 - 0.5e^{-st_s}} &= \frac{1}{1 - 0.5e^{-\Delta t_s} e^{-j\omega t_s}} \\ s &= \Delta + j\omega \end{aligned}$$

Zeros and poles? $H(s)$ will be zero at $\Delta \to -\infty$. Poles:

$$\begin{aligned} 1 - 0.5e^{-st_s} &= 0 \\ 1 &= 0.5e^{-st_s} \\ \ln 1 &= \ln 0.5 - st_s \\ -\ln 0.5 &= -st_s \\ s &= \frac{-0.6931}{t_s} \\ H(s)|_{s=\frac{-0.6931}{t_s}} &= \frac{1}{1 - 0.5e^{t_s^{-1}(0.6931 \pm n2\pi j)}} \end{aligned}$$

So the poles are at:

$$s = \frac{-0.6931}{t_s} \pm \frac{n2\pi j}{t_s}$$

PLOT

Suppose we have $z = e^{st_s}$ and $\frac{1}{z} = e^{-st_s}$. Then

$$H(z) = \frac{1}{1 - 0.5z^{-1}} = \frac{z}{z - 0.5}$$

$H(z)$ has a zero at $z = 0$ instead of infinity, it has a pole at $z = 0.5$
Why is this useful? Assume that we have a digital signal:

$$\begin{aligned}
h(t) &= \delta(t) + h(t - t_s) + h(t - 2t_s) + h(t - 3t_s) \\
H(s) &= 1 + H(s)e^{-st_s} + H(s)e^{-2st_s} + H(s)e^{-3st_s} \\
H(z) &= 1 + H(z)\frac{1}{z} + H(z)\frac{1}{z^2} + H(z)\frac{1}{z^3}
\end{aligned}$$

Definition 29 Z-Transform
We can now define **z-transform** for digital signal

$$\mathcal{Z}_b\left\{f[n]\right\} = F_b(z) = \sum_{n=-\infty}^{\infty} f[n]\, z^{-n} = \tag{72}$$

$$= \ldots + f[-2]\, z^{2} + f[-1]\, z^{1} + f[0]\, z^{0} + +f[1]\, z^{-1} + f[2]\, z^{-2} + \ldots \tag{73}$$

And z is a complex variable $z = \Sigma + i\Omega$. Inverse operation is given by:

$$\mathcal{Z}_b^{-1}\left\{F_b(z)\right\} = f[n] = \frac{1}{2\pi j}\oint F_b(z) z^{n-1} \mathrm{d}z$$

where the integration is performed along a particular counter-clockwise closed path in the z-plane.

Definition 30 Unilateral Z-Transform We can also define unilateral transform:

$$\mathcal{Z}\left\{f[n]\right\} = F(z) = \sum_{n=0}^{N} f[n]\, z^{-n} = f[0]\, z^{0} + +f[1]\, z^{-1} + f[2]\, z^{-2} + \ldots \tag{74}$$

Definition 31 Z-Transform and DTFT

Z-transform is closely connected to Discrete Time Fourier Transform, as

$$\mathcal{Z}_b\left\{x[n]\right\} = X_b(z) = \sum_{n=-\infty}^{\infty} x[n]z^{-n} = \sum_{n=-\infty}^{\infty} x[n]e^{-st_s n} \tag{75}$$

$$t_s = \frac{\omega}{\Omega} \quad \text{and} \quad s = j\Omega \tag{76}$$

$$\mathcal{Z}_b\left\{x[n]\right\}\Big|_{s=j\Omega} = \sum_{n=-\infty}^{\infty} x[n]e^{-j\Omega t_s n} = \sum_{n=-\infty}^{\infty} x[n]e^{-j\omega n} = \mathcal{F}_*\left\{x[n]\right\} \tag{77}$$

◇ **Example 41.** z-transform of a signal
Signal is given by:

$$\begin{aligned}
y[n] &= \{\underline{0}, 3, 5, 6, 9, 2, 4\} \\
y(t) &= 0\delta(t) + 3\delta(t-t_s) + 5\delta(t-2t_s) + 6\delta(t-3t_s) + 9\delta(t-4t_s) + 2\delta(t-5t_s) + 4\delta(t-6t_s) \\
Y(s) &= 3e^{-st_s} + 5e^{-2st_s} + 6e^{-3st_s} + 9e^{-4st_s} + 2e^{-5st_s} + 4e^{-6st_s} \\
z &= e^{st} \\
Y(z) &= 3z^{-1} + 5z^{-2} + 6z^{-3} + 9z^{-4} + 2z^{-5} + 4z^{-6}
\end{aligned}$$

◇ **Example 42.** z-transform of a signal
Assume we have a digital network governed by:

$$H(z) = \frac{3z+2}{5z^2+4z+1}$$

Find difference equation describing the system

$$\begin{aligned}
H(z) = \frac{Y(z)}{X(s)} = \frac{3z+2}{5z^2+4z+1} &= \frac{3z^{-1}+2z^{-2}}{5+4z^{-1}+1z^{-2}} \\
Y(z)\left[5+4z^{-1}+1z^{-2}\right] &= X(z)\left[3z^{-1}+2z^{-2}\right] \\
5y[n]+4y[n-1]+y[n-2] &= 3x[n-1]+2x[n-2]
\end{aligned}$$

$$y[n] = -\frac{4}{5}y[n-1] - \frac{1}{5}y[n-2] + \frac{3}{5}x[n-1] + \frac{2}{5}x[n-2]$$

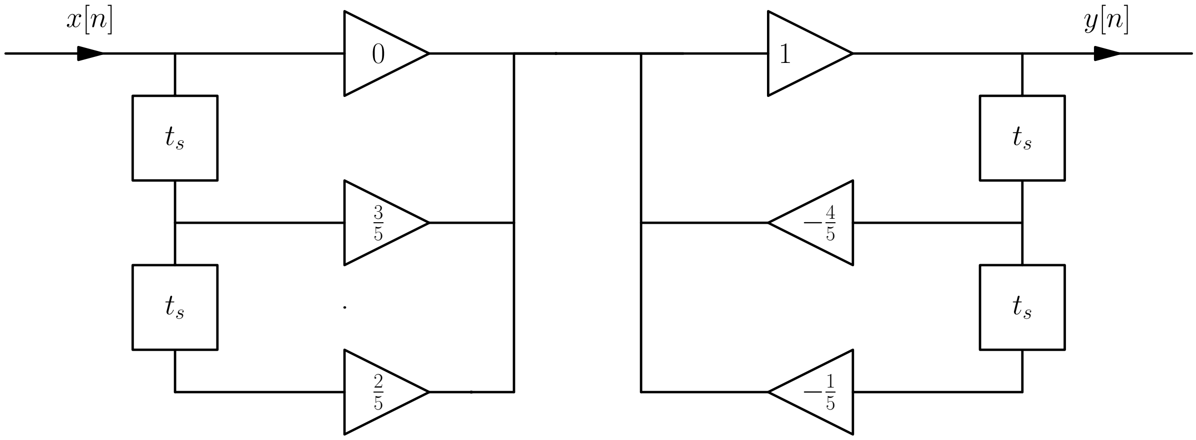


Figure 70:

◇ **Example 43.** convolution using z-transform

If we now go back to the example above and calculate the convolution using Z transform:

$$x[n] = \{2, \underline{1}, 1.5, 1\}$$
$$h[n] = \{\underline{0.5}, 1.5, 1\}$$

$$y[n] = \sum_{k=-\infty}^{\infty} x[k]\, h[n-k]$$

$$\begin{aligned}
X(z) &= 2z + 1 + 1.5z^{-1} + 1z^{-2} \\
H(z) &= 0.5 + 1.5z^{-1} + z^{-2} \\
Y(z) &= X(z)H(s) \\
Y(z) &= \left(2z + 1 + 1.5z^{-1} + 1z^{-2}\right)\left(0.5 + 1.5z^{-1} + z^{-2}\right) = \\
&= z + 0.5 + 0.75z^{-1} + 0.5z^{-2} + \\
&+ \ 3 + 1.5z^{-1} + 1.5^2 z^{-2} + 1.5z^{-3} + \\
&+ \ 2z^{-1} + z^{-2} + 1.5z^{-3} + z^{-4} = \\
&= z + 3.5 + 4.25z^{-1} + 3.75z^{-2} + 3z^{-3} + z^{-4} \\
y[n] &= \{1, \underline{3.5}, 4.25, 3.75, 3, 1\}
\end{aligned}$$

Same as before; OK! ♣

12.3 Important Z transforms

Unit step impulse:

$$u[k] = \begin{cases} 1 & k \geq 0 \\ 0 & k < 0 \end{cases}$$
$$U(z) = 1 + z^{-1} + z^{-2} + z^{-3} + z^{-4} + \ldots$$
$$U(z) = \frac{1}{1 - z^{-1}} = \frac{z}{z - 1} \quad \text{ROC: } |z| > 1$$

Ramp function:

$$R(z) = z^{-1} + 2z^{-2} + 3z^{-1} + ... = \frac{z}{(z-1)^2} \quad \text{ROC: } |z| > 1$$

◇ **Example 44.** Signal is given by: $x[k] = \{1, 2, \underline{3}, 2, 1\}$, find its z-transform.

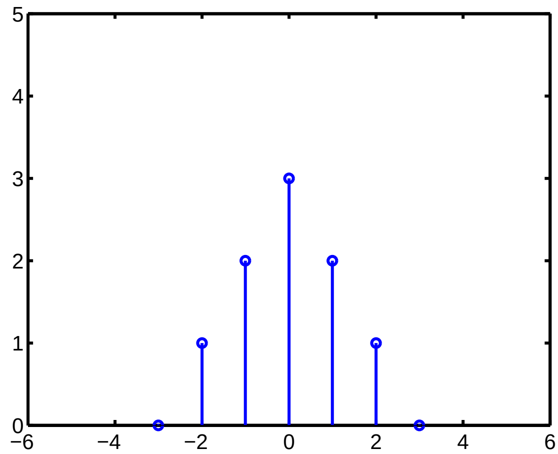


Figure 71: $x[k] = \{1, 2, \underline{3}, 2, 1\}$

$$\begin{aligned}
x[n] &= \{1, 2, \underline{3}, 2, 1\} \\
x(t) &= 1\delta(t + 2t_s) + 2\delta(t + t_s) + 3\delta(t) + 2\delta(t - t_s) + 1\delta(t - 2t_s) \\
X(s) &= 1e^{2st_s} + 2e^{st_s} + 3e^0 + 2e^{-st_s} + 1e^{-2st_s} \\
z &= e^{st} \\
X(z) &= z^2 + 2z^1 + 3z^0 + 2z^{-1} + z^{-2} = \frac{z^4 + 2z^3 + 3z^2 + 2z^1 + 1}{z^2}
\end{aligned}$$

This converges for all $z \neq 0$. ♣

12.4 Time-shift properties of Z-transform

$$\begin{aligned}
\mathcal{Z}\{f[n-n_0]\,u[n-n_0]\} &= \sum_{n=0}^{\infty} f[n-n_0]\,u[n-n_0]\,z^{-n} = \sum_{n=n_0}^{\infty} f[n-n_0]\,z^{-n} = \\
&= f[0]\,z^{-n_0} + f[1]\,z^{-n_0-1} + f[2]\,z^{-n_0-2} + f[3]\,z^{-n_0-3} + ... = \\
&= z^{-n_0}\left[f[0]\,z^0 + f[1]\,z^{-1} + f[2]\,z^{-2} + f[3]\,z^{-3} + ...\right] \\
\mathcal{Z}\{f[n-n_0]\,u[n-n_0]\} &= z^{-n_0}F(z)
\end{aligned}$$

for bilateral transform

$$\begin{aligned}
\mathcal{Z}\{f[n-n_0]\} &= \sum_{n=-\infty}^{\infty} f[n-n_0]\,z^{-n} = \sum_{m=-\infty}^{\infty} f[m]\,z^{-(m+n_0)} = \\
&= z^{-n_0}\sum_{m=-\infty}^{\infty} f[m]\,z^{-m} = z^{-n_0}\mathcal{Z}\{f[m]\} = z^{-n_0}F(z)
\end{aligned}$$

ROC: ROC{f}; z = 0 and z $\to \infty$ need to be considered separately

Definition 32 Time-shift properties of unilateral and bilateral Z-transform

$$\mathcal{Z}\{f[n-n_0]\,u[n-n_0]\} = z^{-n_0}F(z) \quad (78)$$

$$\mathcal{Z}\{f[n-n_0]\} = z^{-n_0}F(z) \quad (79)$$

ROC: ROC{f}; z = 0 and z $\to \infty$ need to be considered separately

Definition 33 Z-transform of the difference equation

We have shown that digital system can be represented by a difference equation

$$\sum_{n=0}^{N} a_n y[k-n] = \sum_{n=0}^{M} b_n x[k-n]$$

$$y[k] + \sum_{n=1}^{N} a_n y[k-n] = \sum_{n=0}^{M} b_n x[k-n]$$

$$Y(z) + Y(z)\sum_{n=1}^{N} a_n z^{-n} = X(z)\sum_{n=0}^{M} b_n z^{-n}$$

$$H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{n=0}^{M} b_n z^{-n}}{1 + \sum_{n=1}^{\infty} a_n z^{-n}}$$

$$H(z) = K\frac{(z-z_1)(z-z_2)...(z-z_M)}{(z-p_1)(z-p_2)...(z-p_N)}$$

NOTE: zeros and poles do NOT have the same interpretation as for the LT and FT as $z = e^{st_s}$.

◇ **Example 45.** Given difference equation:

$$y[n] - 0.6y[n-1] - 0.2y[n-2] = x[n] + 0.9x[n-1] \quad (80)$$

Find output for $x[n] = \{\underline{3}, 1, 2\}$.

Using time shifting properties of the z-transform

$$Y(z) - 0.6Y(z)z^{-1} - 0.2Y(z)z^{-2} = X(z)\left(1 + 0.9z^{-1}\right)$$

$$H(z) = \frac{Y(z)}{X(z)} = \frac{1 + 0.9z^{-1}}{1 - 0.6z^{-1} - 0.2z^{-2}} = \frac{z^2 + 0.9z}{z^2 - 0.6z - 0.2}$$

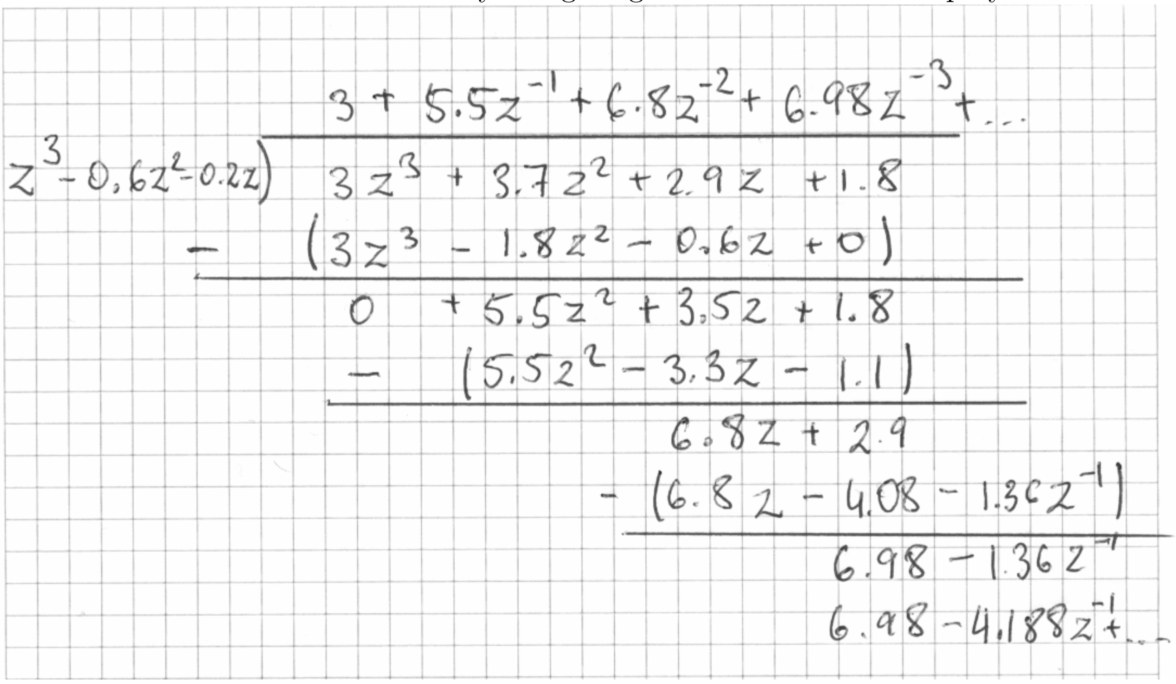
Z-transform of the input:

$$X(z) = 3 + z^{-1} + 2z^{-2}$$

Output:

$$Y(z) = X(z)H(z) = \frac{(z^2+0.9z)\left(3+z^{-1}+2z^{-2}\right)}{z^2-0.6z-0.2} = \frac{3z^3+3.7z^2+2.9z+1.8}{z^3-0.6z^2-0.2z}$$

We can now do inverse transform by using long division of those two polynomials:



Matlab command `ldiv(b,a)` can bu used to get the coefficients. For the example above, the first couple of coefficients will be:

$$y[n] = \{\underline{3.00}, 5.50, 6.80, 6.98, 5.548, 4.7248, 3.9445, 3.3116, ...\}$$

We can also use partial fraction expansion to get the inverse z-transform. ♣

12.5 connection between s- and z-planes

If we consider an analogue signal with defined cut-off frequency Ω_M which is sampled with $\Omega_s = 2\Omega_M$. This signal corresponds to a strip in the s-plane.

$$\begin{aligned} z = e^{sT_s} = e^{(\sigma+\Omega j)T_s} = e^{\sigma T_s}e^{j\Omega T_s} &= re^{j\omega} \\ r &= e^{T_s\sigma} \\ w &= \Omega T_s \end{aligned}$$

Where Ω is the frequency on the s-plane and $r > 0$. For $\sigma = 0$, $r = 1$, for $\sigma > 0$, $r > 1$ and for $\sigma < 0$, $r < 1$.

$$\begin{aligned} -\frac{\Omega_s}{2} < \Omega < \frac{\Omega_s}{2} \\ -\frac{2\pi}{2T_s} < \Omega < \frac{2\pi}{2T_s} \\ -\frac{2\pi}{2T_s} < \frac{\omega}{T_s} < \frac{2\pi}{2T_s} \\ -\pi < \omega < \pi \end{aligned}$$

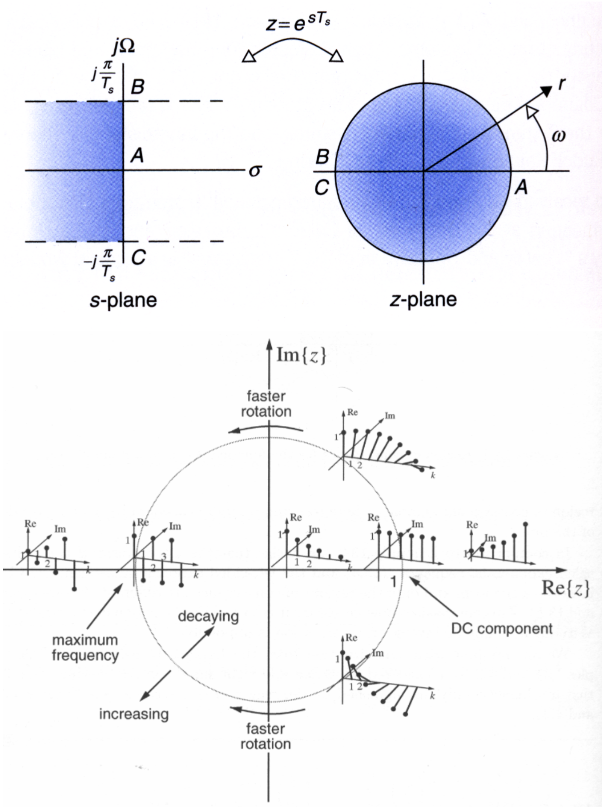


Figure 13.6: Illustration of the z-plane

Figure 72:

12.6 Discrete time convolution using z-transform

$$y[n] = \sum_{k=-\infty}^{\infty} x[k]\, h[n-k]$$

If we let our x[k] to be causal (starts at zero)

$$y[n] = x[0]\, h[n-0] + x[1]\, h[n-1] + x[2]\, h[n-2] +$$

Now we take z-transform:

$$\begin{aligned} Y(z) &= x[0]\, H(z) z^{0} + x[1]\, H(z) z^{-1} + x[2]\, H(z) z^{-2} + ... \\ &= H(z)\left[x[0]\, z^{0} + x[1]\, z^{-1} + x[2]\, z^{-2} + ...\right] \\ Y(z) &= H(z) X(z) \end{aligned}$$

And ROC for $Y(z)$ is at least the intersection between ROCs for $X(z)$ and $H(z)$.
In general:

$$\begin{aligned} Y(z) &= \sum_{n=-\infty}^{\infty} y[n] z^{-n} = \sum_{n=-\infty}^{\infty}\sum_{k=-\infty}^{\infty} x[k]\, h[n-k]\, z^{-n} = \sum_{k=-\infty}^{\infty}\sum_{n=-\infty}^{\infty} x[k]\, h[n-k]\, z^{k-n} z^{-k} = \\ &= \sum_{k=-\infty}^{\infty} x[k] \sum_{n=-\infty}^{\infty} h[n-k]\, z^{k-n} z^{-k} = \\ &= \sum_{k=-\infty}^{\infty} x[k]\, z^{-k} \sum_{n=-\infty}^{\infty} h[n-k]\, z^{n-k} = \sum_{k=-\infty}^{\infty} x[k]\, z^{-k} H(z) = X(z) H(z) \end{aligned}$$

Definition 34 Discrete time convolution using z-transform

$$y[n] = \sum_{k=-\infty}^{\infty} x[k]\, h[n-k] \quad (81)$$

$$Y(z) = X(z)H(z) \quad (82)$$

12.7 ROC for finite and infinite support signals

For a finite support signal x[n] defined for $n \in [N_0, N]$, the ROC of the z-transform is is the whole z-plane excluding $z = 0$ and/or $z = \pm\infty$, depending on the N and N_0.

Signals of infinite support can be causal, anti causal combination of both or noncauslal. For a causal signal:

$$X_c(z) = \sum_{n=0}^{\infty} x_c[n]\, z^{-n} = \sum_{n=0}^{\infty} x_c[n]\, r^{-n} e^{-jn\omega}$$

The frequency ω has no effect on the convergence. If R_1 is the radius of the farthest-out pole of $X_c(z)$ then the region of convergence will be befined by:

$$r = |z| > R_1$$

As $|x[n]\,| < MR_1^n$ and some value of $M > 0$.

For an anti-causal signal

$$r = |z| < R_2$$

A noncauslal signal has a ROC

$$R_1 < |z| < R_2$$

Example: Find ROC and Z-transforms of two following signals:

$$x_1[n] = \left(\frac{1}{2}\right)^n u[n]$$

$$x_2[n] = -\left(\frac{1}{2}\right)^n u[-n-1]$$

Signal x_1 is causal ane x_2 is anticauslat.

$$X_1(z) = \sum_{n=0}^{\infty} \left(\frac{1}{2}\right)^n z^{-n} = \frac{1}{1-0.5z^{-1}} = \frac{z}{z-0.5}$$

$$X_2(z) = -\sum_{n=-\infty}^{-1} \left(\frac{1}{2}\right)^n z^{-n} = -\sum_{m=0}^{\infty} \left(\frac{1}{2}\right)^{-m} z^m + 1 =$$

$$= -\sum_{m=0}^{\infty} (2)^m z^m + 1 = \frac{-1}{1-2z} + 1 = \frac{z}{z-0.5} \frac{1}{1-0.5z^{-1}} = \frac{z}{z-0.5}$$

For $X_1(z)$, the pole is at $R_1 = 0.5$ and $|z| > 0.5$. For $X_2(z)$, the pole is also at $R_1 = 0.5$ but now $|z| < 0.5$. So the z-transform is the same but ROC is different. Both are needed to describe the signals.

12.8 Signal behaviour and poles of the z-transform

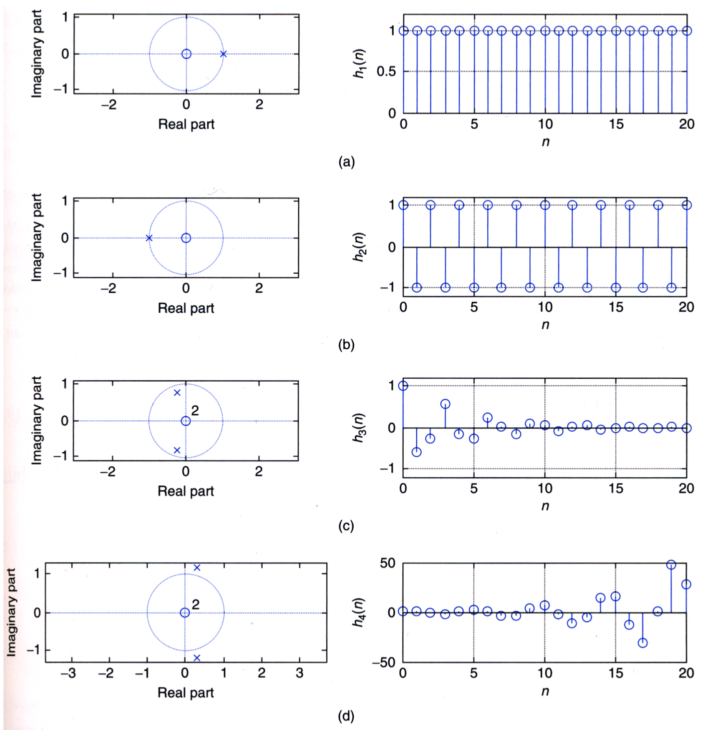


Figure 73:

The z-transform is a linear transformation, meaning that

$$\mathcal{Z}\left\{ax[n] + ay[n]\right\} = a\mathcal{Z}\left\{x[n]\right\} + b\mathcal{Z}\left\{y[n]\right\}$$

If we consider a signal with real or complex α:

$$x[n] = a^n u[n]$$

Z-transform of this signal can be used to compute z-transforms of for example:

$$x[n] = \cos(\omega_0 n + \theta)$$

The z-transform of causal signal $x[n] = a^n u[n]$ is:

$$x[n] = a^n = a^0 + a^1 + a^2 + ... = \frac{1}{1-a} \quad |a| < 1$$
$$\mathcal{Z}\left\{a^n\right\} = a^0 z^0 + a^1 z^{-1} + +a^2 z^{-2} + ... = \frac{1}{1 - az^{-1}} = \frac{z}{z-a}$$
$$\left|az^{-1}\right| < 1$$
$$|a| < |z|$$

Special case 1: a = 1, then

$$a^n = u[n]$$
$$\mathcal{Z}\left\{u[n]\right\} = \frac{z}{z-1}$$

This has a pole at $z = 1e^{i0}$, radius 1 and lowest discrete frequency $\omega = 0$.

If $a = -1$, then the z-transform has a pole at $z = -1 = 1e^{i\pi}$ (highest discrete frequency $\omega = \pi$). For $a \in \Re$, if we move the pole toward the centre of the z-plane, then the corresponding signal decays exponentially for $0 < a < 1$ and is a modulated exponential for $-1 < a < 0$, $x[n] = |a|^n \cos(\pi n)$.

Special case 2:

$$a^n = \left(e^{\beta}\right)^n = e^{\beta n}$$
$$\beta = \ln a$$
$$\mathcal{Z}\left\{e^{\beta n}\right\} = \frac{z}{z - e^{\beta}}$$

cos-function:

$$\cos bn = \frac{1}{2}\left[e^{jbn} + e^{-jbn}\right]$$
$$\mathcal{Z}\left\{\cos bn\right\} = \frac{1}{2}\mathcal{Z}\left\{e^{jbn}\right\} + \frac{1}{2}\mathcal{Z}\left\{e^{-jbn}\right\} = \frac{1}{2}\left(\frac{z}{z - e^{jb}}\right) + \frac{1}{2}\left(\frac{z}{z - e^{-jb}}\right) =$$
$$= \frac{z}{2}\left(\frac{z - e^{-jb} + z - e^{jb}}{(z - e^{jb})(z - e^{-jb})}\right) = \frac{z(z - \cos b)}{z^2 - 2z\cos b + 1}$$

Appendix B.5 Two-sided z-Transform Pairs

$x[k]$	$X(z) = \mathcal{Z}\{x[k]\}$	ROC
$\delta[k]$	1	$z \in \mathbb{C}$
$\varepsilon[k]$	$\frac{z}{z-1}$	$\lvert z\rvert > 1$
$a^k\varepsilon[k]$	$\frac{z}{z-a}$	$\lvert z\rvert > \lvert a\rvert$
$-a^k\varepsilon[-k-1]$	$\frac{z}{z-a}$	$\lvert z\rvert < \lvert a\rvert$
$k\varepsilon[k]$	$\frac{z}{(z-1)^2}$	$\lvert z\rvert > 1$
$ka^k\varepsilon[k]$	$\frac{az}{(z-a)^2}$	$\lvert z\rvert > \lvert a\rvert$
$\sin(\Omega_0 k)\varepsilon[k]$	$\frac{z\sin\Omega_0}{z^2 - 2z\cos\Omega_0 + 1}$	$\lvert z\rvert > 1$
$\cos(\Omega_0 k)\varepsilon[k]$	$\frac{z(z-\cos\Omega_0)}{z^2 - 2z\cos\Omega_0 + 1}$	$\lvert z\rvert > 1$

Figure 74:

13 Digital filters

We have discussed time continuous filters, and would like to find out how to make a DSP filters with similar properties. 1st order Butterworth filter contained 2 elements, R and C. For that filter:

$$V_o(s) = V_i(s)\frac{1}{1 + RCs}$$

We can use this to find a difference equation

$$\begin{aligned} V_o(s)(1+RCs) &= V_i(s) \\ V_i(s) - V_o(s) &= RCsV_o(s) \\ v_i(t) - v_o(t) &= RC\frac{\mathrm{d}v_o}{\mathrm{d}t} \end{aligned}$$

If we now discretize this equation and find output

$$\begin{aligned} x[n] - y[n] &= RC\frac{y[n]-y[n-1]}{t_s} \\ x[n]\frac{t_s}{RC} - y[n]\frac{t_s}{RC} &= y[n] - y[n-1] \\ x[n]\frac{t_s}{RC} + y[n-1] &= y[n] + y[n]\frac{t_s}{RC} \\ x[n]\frac{t_s}{RC} + y[n-1] &= y[n]\left(1+\frac{t_s}{RC}\right) \\ y[n] = x[n]\frac{t_s}{RC+t_s} &+ y[n-1]\frac{RC}{RC+t_s} \\ y[n] &= x[n]\alpha + y[n-1](1-\alpha) \end{aligned}$$

and

$$\alpha = \frac{t_s}{RC+ts}$$

And this will relate time continuous realisation of the filter with DSP filter.

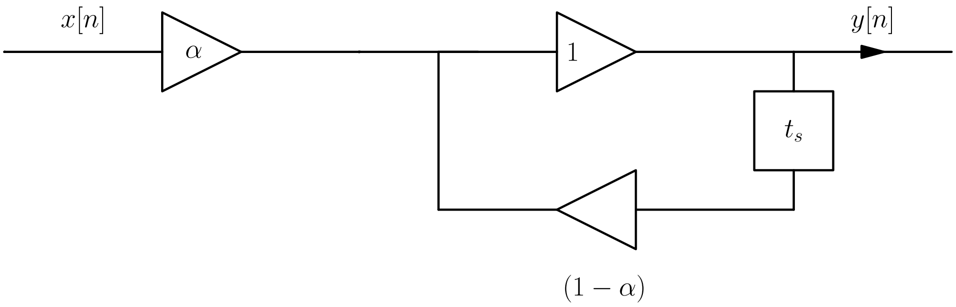


Figure 75:

$$\begin{aligned} y[n] &= \alpha x[n] + (1-\alpha)y[n-1] \\ Y(z) &= \alpha X(z) + (1-\alpha)Y(z)z^{-1} \\ H(z) = \frac{Y(z)}{X(s)} = \frac{\alpha}{1-(1-\alpha)z^{-1}} &= \frac{\alpha z}{z-(1-\alpha)} \end{aligned}$$

Now we let x[n] be the unit step response:

$$\begin{aligned} X(z) &= \frac{z}{z-1} \\ Y(z) &= H(z)X(z) = \frac{\alpha z}{z-(1-\alpha)}\frac{z}{z-1} = \\ &= \alpha z\left[\frac{1}{z-(1-\alpha)}\frac{z}{z-1}\right] = \alpha z\left[\frac{k_1}{z-(1-\alpha)} + \frac{k_2}{z-1}\right] \\ k_1 &= \frac{\alpha-1}{\alpha} \quad k_2 = \frac{1}{\alpha} \end{aligned}$$

$$\begin{aligned}
Y(z) &= \alpha z \left[\frac{\alpha - 1}{\alpha} \frac{1}{z-(1-\alpha)} + \frac{1}{\alpha} \frac{1}{z-1} \right] = \frac{-z(1-\alpha)}{z-(1-\alpha)} + \frac{z}{z-1} \\
\mathcal{Z}\left\{a^n\right\} &= \frac{z}{z-a} \\
y[n] &= -(1-\alpha)(1-\alpha)^n + u[n] \\
y[n] &= u[n] - (1-\alpha)^{n+1} = 1 - (1-\alpha)^{n+1} \quad n \geq 0
\end{aligned}$$

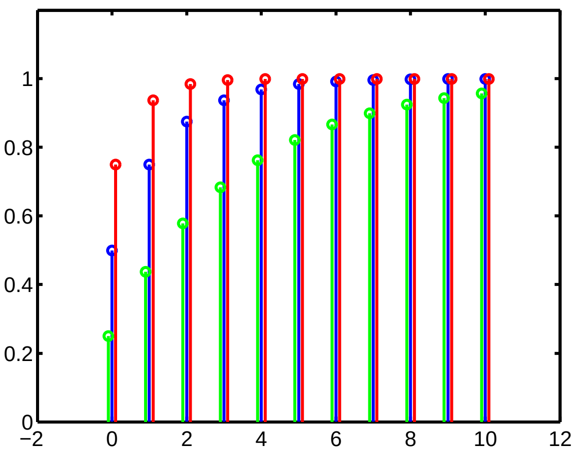
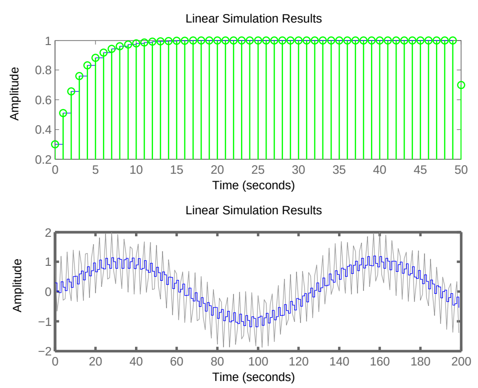


Figure 76:

◇ **Example 46.** DSP filter What will be spectral response of a filter above. First we can test that by supplying a combination of sin functions as an input.




```
close all;
% filter
alpha = 0.1;

b = [alpha 0];
a = [1 -(1-alpha)];

A = ldiv(b,a);
x = zeros(1,50)+1;
set(gca,'linewidth',2);
set(gca,'fontsize',14);
A = dimpulse(b,a);
y = conv(A,x);
subplot(2,1,1)
dlsim(b,a,x); hold on;
stem(y,'LineWidth',1,'Color','green');
hold off;

n = 1:200;
x = sin(2*n)+sin(n/20);
y = conv(A,x);
subplot(2,1,2)
dlsim(b,a,x); hold on;
%stem(y,'LineWidth',1,'Color','green');
hold off;

set(gcf,'PaperUnits','points');
set(gcf,'PaperSize',[400 400]);
set(gcf,'PaperPositionMode','auto')
set(gca,'linewidth',2);
set(gca,'fontsize',10);
print('-dpdf','-r300','digifilter3')
```

Figure 77: Digital low pass filter described by a difference equation $y[n] = \alpha x[n] + (1-\alpha)y[n-1]$ and $\alpha = 0.1$

So the filter works as a low pass filter. Magnitude response can be calculated using matlab command: `fvtool(b,a);` where `a` and `b` describes z-transform of filter transfer function.

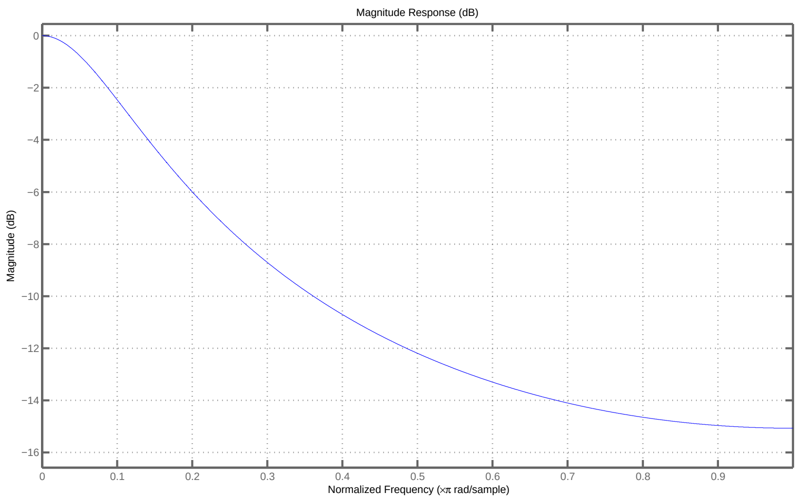
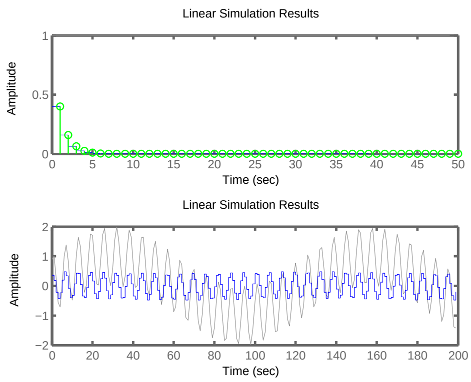


Figure 78: Magnitude response for a filter with difference equation: $y[n] = \alpha x[n] + (1 - \alpha)y[n-1]$ and $\alpha = 0.1$

◇ **Example 47.** High pass filter can be defined by a difference equation

$$
\begin{aligned}
y[n] &= \alpha y[n-1] + \alpha\left(x[n] - x[n-1]\right) \\
Y(z) &= \alpha X(z) z^{-1} + \alpha\left(X(z) - X(z)z^{-1}\right) \\
H(z) &= \frac{Y(z)}{X(s)} = \frac{\alpha(1-z^{-1})}{1-\alpha z^{-1}} = \frac{\alpha z - \alpha}{z - \alpha}
\end{aligned}
$$




```
close all;
% filter
alpha = 0.4;

b = [alpha -alpha];
a = [1 -alpha];

A = ldiv(b,a);
x = zeros(1,50)+1;
set(gca,'linewidth',2);
set(gca,'fontsize',14);
A = dimpulse(b,a);
y = conv(A,x);
subplot(2,1,1)
dlsim(b,a,x); hold on;
stem(y,'LineWidth',1,'Color','green');
hold off;

n = 1:200;
x = sin(n)+sin(n/20);
y = conv(A,x);
subplot(2,1,2)
dlsim(b,a,x); hold on;
%stem(y,'LineWidth',1,'Color','green');
hold off;

set(gcf,'PaperUnits','points');
set(gcf,'PaperSize',[400 400]);
set(gcf,'PaperPositionMode','auto')
set(gca,'linewidth',2);
set(gca,'fontsize',10);
print('-dpdf','-r300','digifilter3H')
```

Figure 79: Digital high pass filter described by a difference equation $y[n] = \alpha y[n-1] + \alpha\left(x[n] - x[n-1]\right)$ and $\alpha = 0.4$

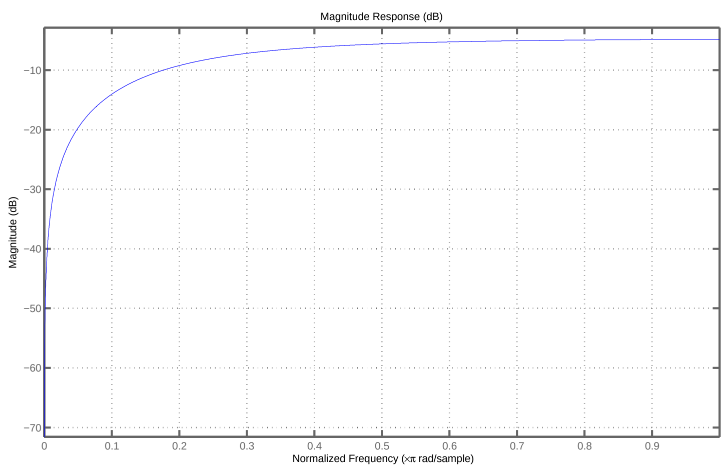


Figure 80: Magnitude response for a filter with difference equation: $y[n] = \alpha y[n-1] + \alpha\left(x[n] - x[n-1]\right)$ and $\alpha = 0.4$

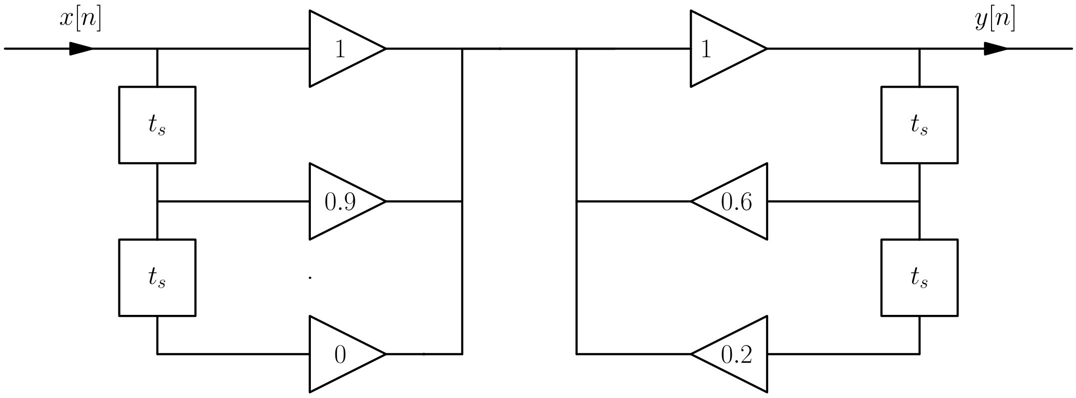


Figure 81: DSP system

◇ **Example 48.** Determine output for system shown in figure 81 and input signal:

$$x[n] = \{\underline{3}, 1, 2\}$$

We first need to find the difference equation from the diagram:

$$y[n] = x[n] + 0.9x[n-1] + 0.6y[n-1] + 0.2y[n-2]$$

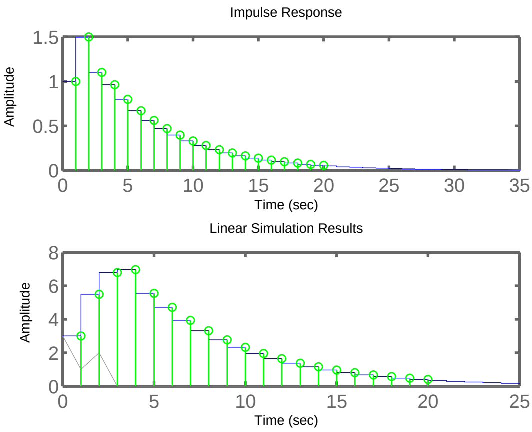
We can make table or use z-transform. If we make a table we have to tabulate: x[n], 0.9x[n-1], y[n], -0.6y[n-1] and -0.2y[n-2] and sum up. $n \in [0, \infty]$

$$\begin{aligned}
y[n] - 0.6y[n-1] - 0.2y[n-2] &= x[n] + 0.9x[n-1] \\
Y(z)(1 - 0.6z^{-1} - 0.2z^{-2}) &= X(z)(1 + 0.9z^{-1}) \\
H(z) = \frac{1 + 0.9z^{-1}}{1 - 0.6z^{-1} - 0.2z^{-2}} &= \frac{z^2 + 0.9z}{z^2 - 0.6z - 0.2} \\
X(z) &= 3 + z^{-1} + 2z^{-2} \\
Y(z) = X(z)H(z) = \frac{(z^2 + 0.9z)(3 + z^{-1} + 2z^{-2})}{z^2 - 0.6z - 0.2} &= \frac{3z^3 + 3.7z^2 + 2.9z + 1.8}{z^3 - 0.6z^2 - 0.2z}
\end{aligned}$$

We can use matlab or long division to find impulse response and output:

There are many ways to analyse digital networks:

- From difference equation directly using $x[n]$ and a table
- Z-transform of the system and the input; multiply and then do long division or tables
- find H(z), find impulse response and do convolution
- use Matlab to implement some of those methods




```
b1 =  [3 3.7 2.9 1.8];
a1 = [1 0.6 0.2 0];
b =  [1 .9 0];
a = [1 -0.6 -0.2];
A = ldiv(b,a);
x = [3 1 2 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]
subplot(2,1,1)
dimpulse(b,a); hold on;
stem(A,'LineWidth',1,'Color','green');
hold off;
set(gca,'linewidth',2);
set(gca,'fontsize',14);
A = dimpulse(b,a);
%output
y = conv(x, A);
%or
subplot(2,1,2)
dlsim(b,a,x); hold on;
b1 =  [3 3.7 2.9 1.8];
a1 = [1 -0.6 -0.2 0];
A = ldiv(b1,a1);
stem(A,'LineWidth',1,'Color','green');
hold off;
```

Figure 82:

13.1 IIR and FIR Filters

IIR stands for infinite impulse response and FIR stands for finite impulse response. The two different filter types differ by expansion describing

IIR

$$\sum_{n=0}^{N} a_n y[k-n] = \sum_{n=0}^{M} b_n x[k-n]$$

$$H(z) = \frac{Y(z)}{X(z)} = \frac{\sum\limits_{n=0}^{M} b_n z^{-n}}{1 + \sum\limits_{n=1}^{\infty} a_n z^{-n}}$$

$$H(z) = K\frac{(z-z_1)(z-z_2)...(z-z_M)}{(z-p_1)(z-p_2)...(z-p_N)}$$

– $H(z)$ has poles, so the filter might not be stable

– h[n] is described by a recursive equation and is typically decaying function

FIR

$$\begin{aligned} y[k] &= \sum_{n=0}^{M} b_n x[k-n] \\ H(z) &= \frac{Y(z)}{X(z)} = \sum_{n=0}^{M} b_n z^{-n} \\ H(z) &= K(z-z_1)(z-z_2)...(z-z_M) \end{aligned}$$

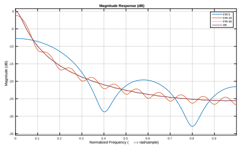
– $H(z)$ has no poles, so the filter is always stable

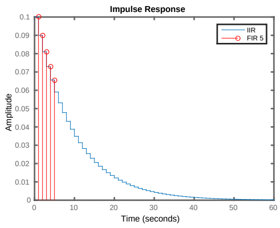
– To make a good filter one need large M (larger then fo IIR)

– $h[n] = b[n]$

Could we convert an IIR low pass filter discussed above to FIR version?

$$\begin{aligned} \beta &= (1-\alpha) \\ H(z) &= \frac{\alpha z}{z-\beta} = \alpha\frac{z}{z-\beta} = \alpha\frac{1}{1-\beta/z} = \alpha\left[(\beta/z)^0 + (\beta/z)^1 + (\beta/z)^2 + ...\right] \\ h[n] &= \alpha\beta^n \end{aligned}$$






```
clear all; close all;
alpha = 0.1;
b = [alpha 0];
a = [1 -(1-alpha)];

A2 = ldiv(b,a,2);
A5 = ldiv(b,a,5);
A20 = ldiv(b,a,20);
A40 = ldiv(b,a,40);
fvtool(A5,1,A20,1,A40,1,b,a);
set(gca,'linewidth',2);

legend('FIR 5','FIR 20','FIR 40','IIR')

set(gcf, 'PaperSize',[14 11]);
set(gcf, 'PaperPositionMode', 'auto');
set(gca,'linewidth',2);
set(gca,'fontsize',10);
print('-dpdf','-r300','FIR_IIR.pdf')
figure;
dimpulse(b,a)
hold on;
stem(1:5,A5,'r');
legend('IIR','FIR 5')
set(gcf, 'PaperSize',[6 5]);
set(gcf, 'PaperPositionMode', 'auto');
set(gca,'linewidth',2);
set(gca,'fontsize',10);

print('-dpdf','-r300','FIR_IIR_IR.pdf')
hold off;

figure
zplane(b,a);
figure
zplane(A40,1);
```

Figure 83: Comparison between magnitude response calculated for simplest IIR low pass filter and a corresponding FIR filter with 5,20 and 40 terms (b_n)

We need to test this in matlab! This really works, see Figure 83. Conversion between FIR and IIR is not always possible, and there exist a lot of methods to design filters with with desired characteristics which involve minimum number of terms. See for example matlab documentation for `fdesign.lowpass, fdesign.highpass`. In genereal one need to consider phase response in addition to amplitude response (depending on application). DEMO

◇ **Example 49.** Filter with 1 zero

$$y[n] = x[n] + x[n-1]$$
$$Y(z) = X(z)\left(1 + z^{-1}\right) = X(z)\frac{z+1}{z}$$

This has one zero at z = -1 and no poles. behaviour? For a signal with max frequency ($\omega = \pi$), $x[n] = -x[n-1]$ (this signal has a form $(-1)^n$). So the folter will remove max frequency for the input but will also effect frequencies near by. ♣

◇ **Example 50.** FIR convolution Use the table below to find $y[n]$ using discrete convolution, where $h[k] = \{\underline{10}, -2, 6, -1, 3, 0\}$ and $x[k] = \{\underline{1}, 2, 3, 4, 5\}$ as a input.

k=	-5	-4	-3	-2	-1	0	1	2	3	4	5	6	y[n]=
h[k]=													
n= 0													
n= 1													
n= 2													
n= 3													
n= 4													
n= 5													
n= 6													
n= 7													
n= 8													
n= 9													
n= 10													
n= 11													
n= 12													