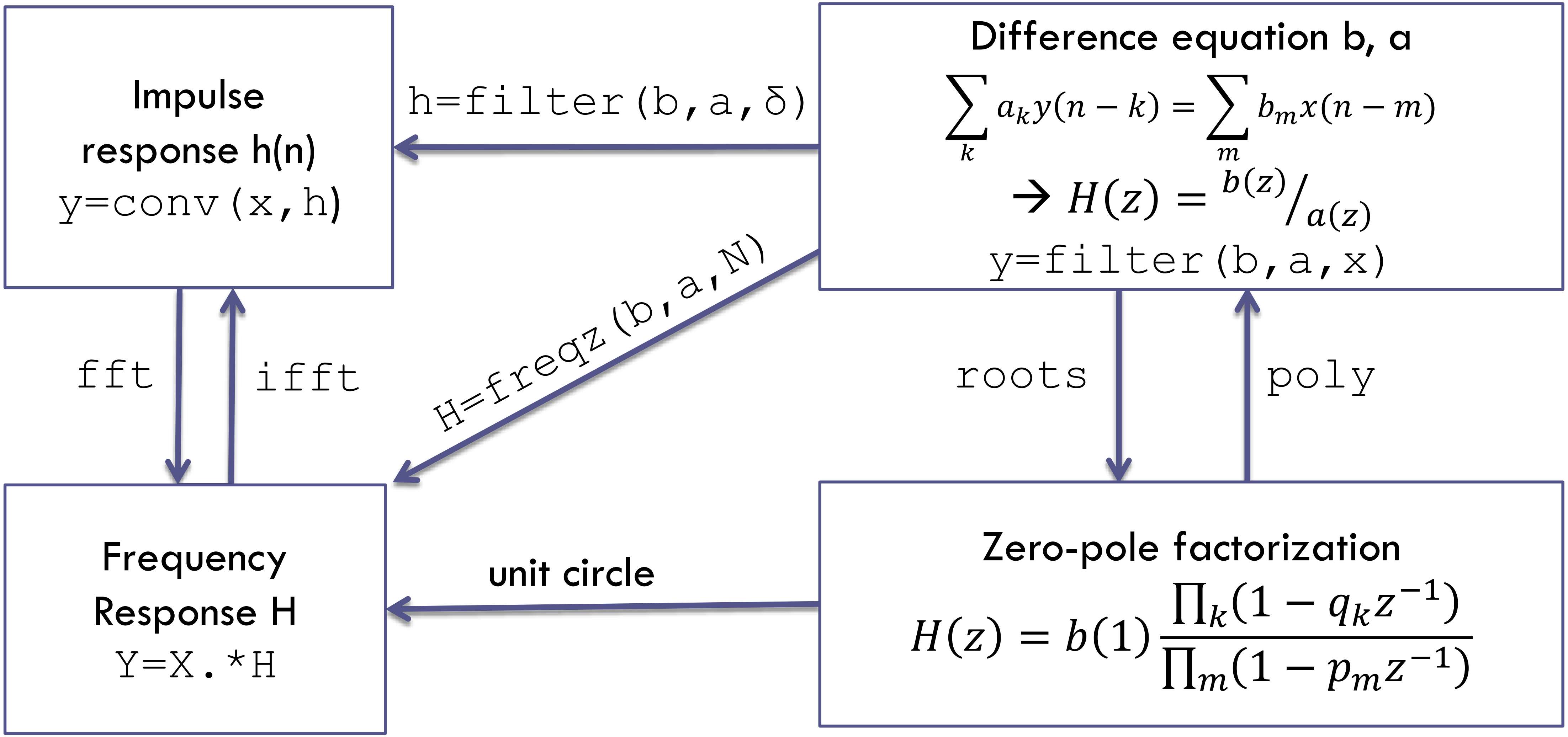
DIGITAL FILTERS

Michele Buccoli

Filter characterization



Filter characterization

- `h=filter(b,a,δ)` → length(h)= length(δ)
- `H=fft(h,N)` (`h=ifft(H,N)`) computes (requires) the frequency components from 0 to 2 π
- `[H,f]=freqz(b,a,N)` computes the frequency components from 0 to π → simmetry properties
- Estimate H on the unit circle:

```
f=linspace(0,pi,N+1); f=f(1:end-1);
z=roots(b); p=roots(a);
H=b(1)*prod((1-z*exp(-1i*f')),1)./…
       prod((1-p*exp(-1i*f')),1);
H=H.';
```

Filter characterization

- See MATLAB code "DigitalFilters".

Filters

- The amplitude of the response is **enhanced** in the frequency components near the **poles**
  - Poles are the infinite-impulse response (IIR) component of the filter
  - If not properly placed, they can make the system **unstable**
- The amplitude of the response is **reduced** in the frequency components near the **zeros**
  - Zeros are the finite-impulse response (FIR) component of the filter
- Let's write a function `show_filter(z,p)` that shows:
  - on the left subplot the zplane
  - on the right subplot the transfer function (see `plotyy`)

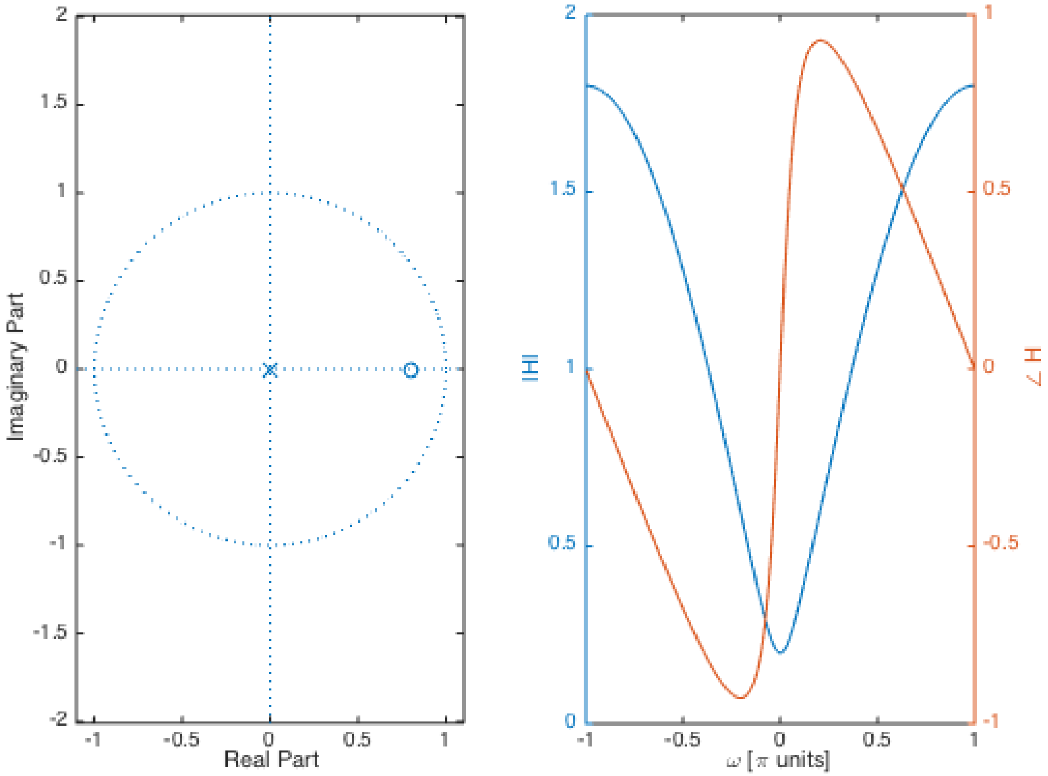
Filters

This function plots a filter, given its zeros and poles

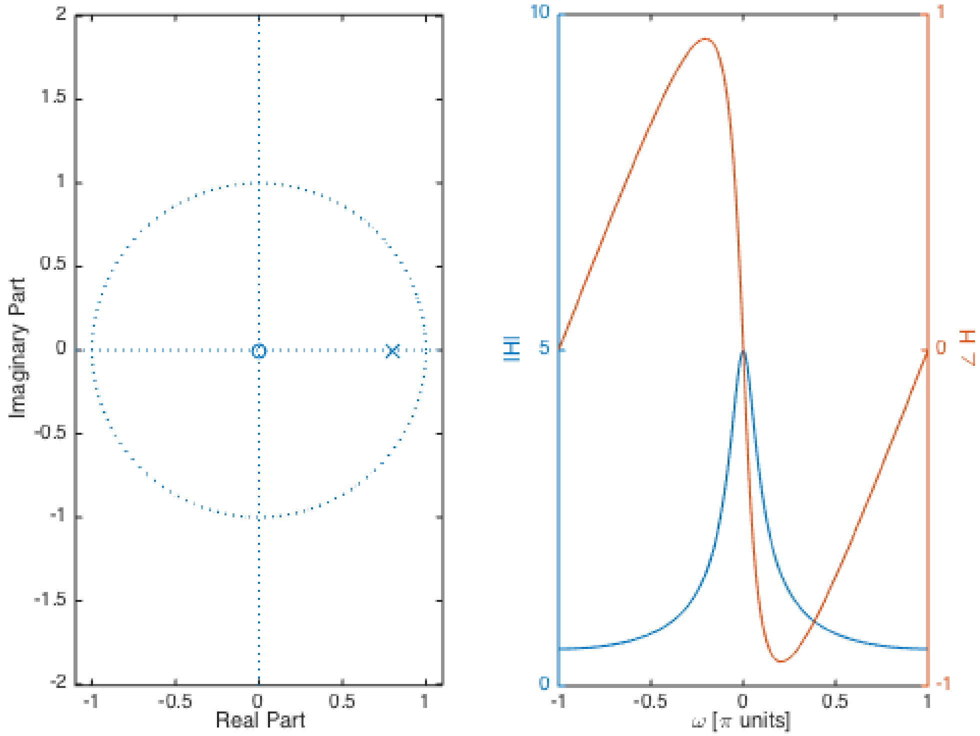
```
function [  ] = show_filter( z,p )
  a=poly(p); b=poly(z); [H, f]=freqz(b,a);
  f=[-flipud((f(2:end)));f]/pi;
  H=[flipud(conj(H(2:end)));H];
  figure; subplot(1,2,1); zplane(b,a);
  subplot(1,2,2);
  [ax,p1,p2] = plotyy(f,abs(H),f,angle(H));
  ylabel(ax(1),'|H|'); % label left y-axis
  ylabel(ax(2),'\angle H'); % right y-axis
  xlabel(ax(1),'\omega [\pi units]');
  xlabel(ax(2),'\omega [\pi units]');
end
```

Filters

One zero-filter with z_1=0.8
high-pass filter

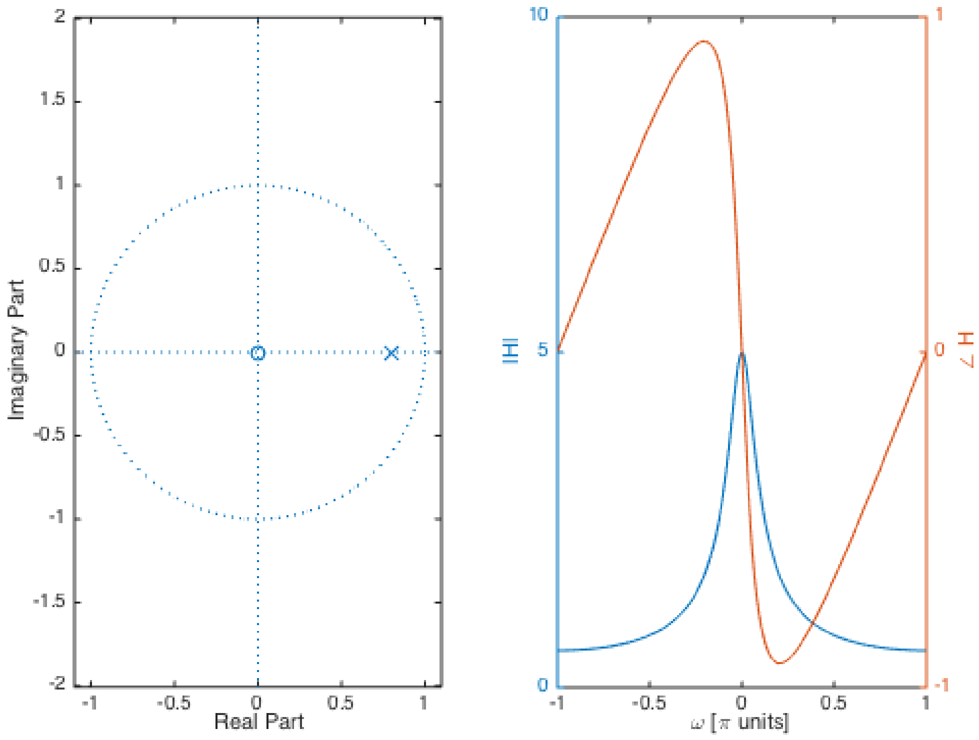


One pole-filter with p_1=0.8
low-pass filter



Filters

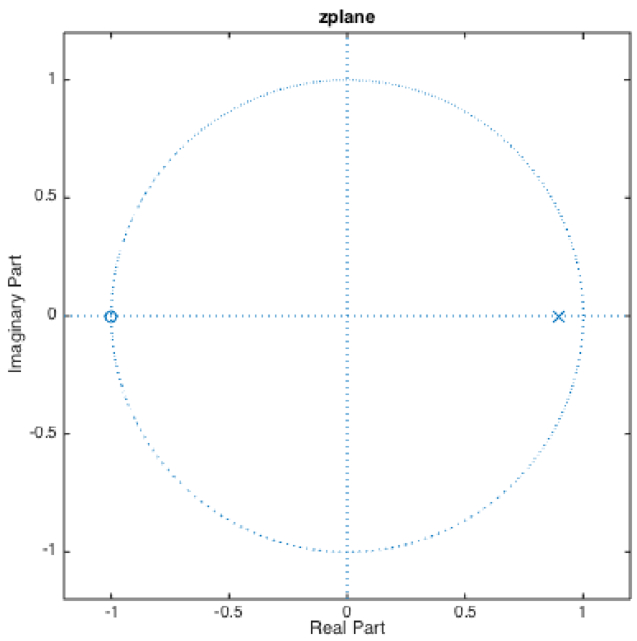
- One pole-filter with p_1=0.8: low-pass filter

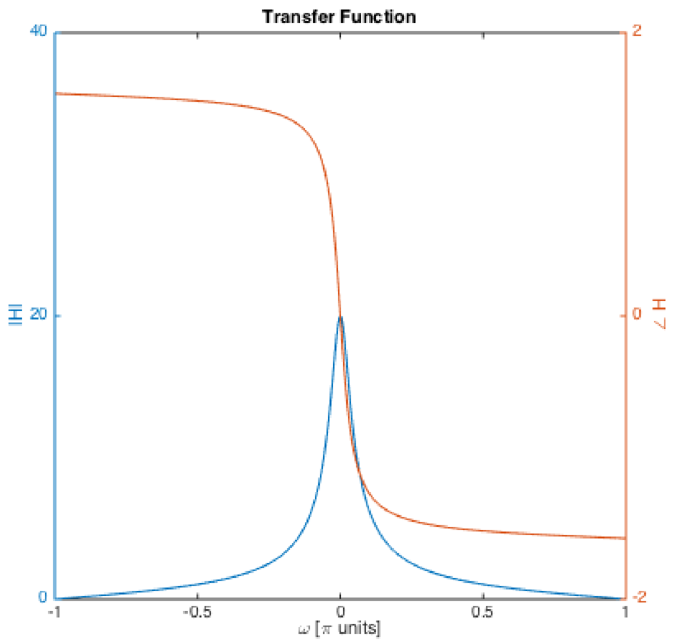


Filter design

Filters

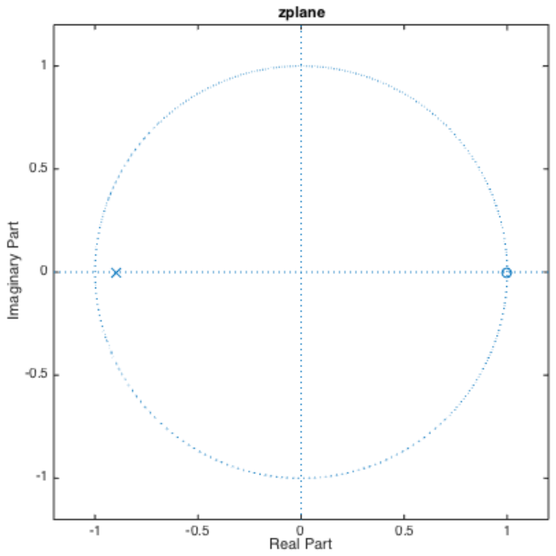
- Lowpass filter
  - Zeros close to $\omega=\pi$ → -1
  - Poles close to $\omega=0$ → 0.9 (inside the unit circle!)
  - $H(z)=(1+z^{-1})/(1-0.9z^{-1})$

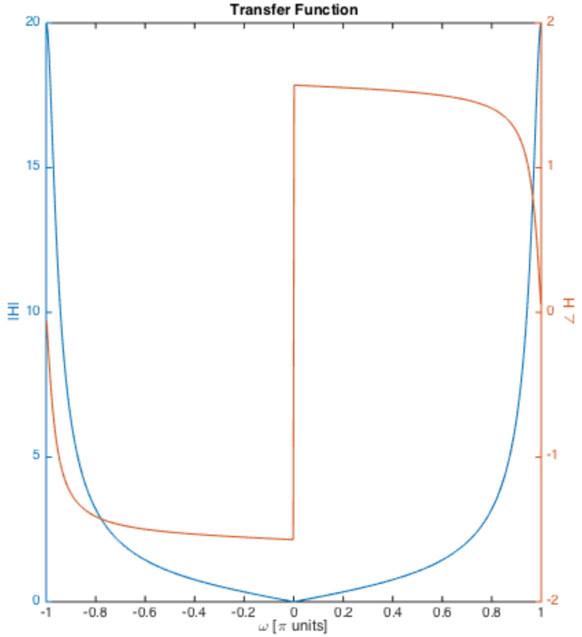




Filters

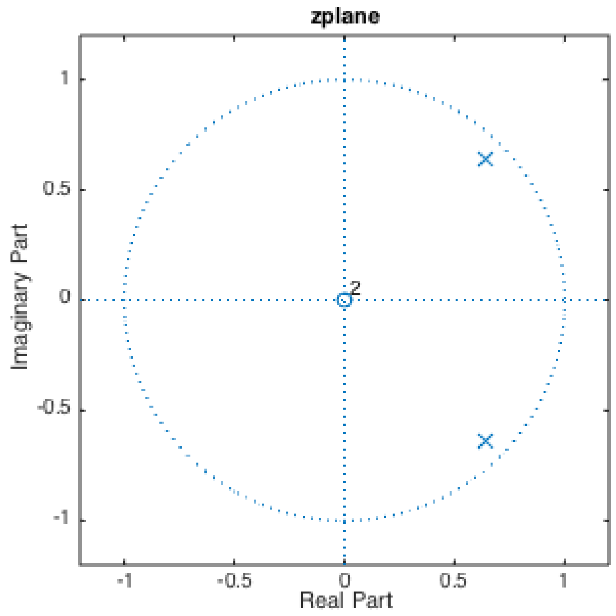
- Highpass filter
  - Zeros close to $\omega=0 \rightarrow 1$
  - Poles close to $\omega=\pi \rightarrow -0.9$ (inside the unit circle!)
  - $H(z)=(1-z^{-1})/(1+0.9\, z^{-1})$

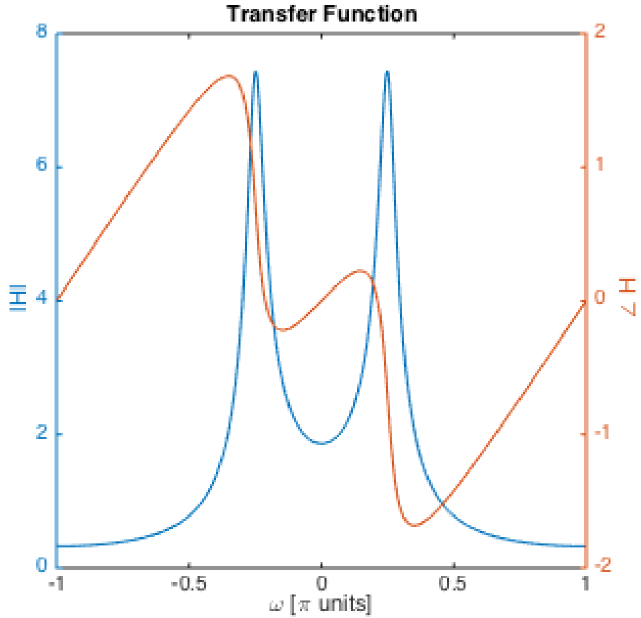




Filters

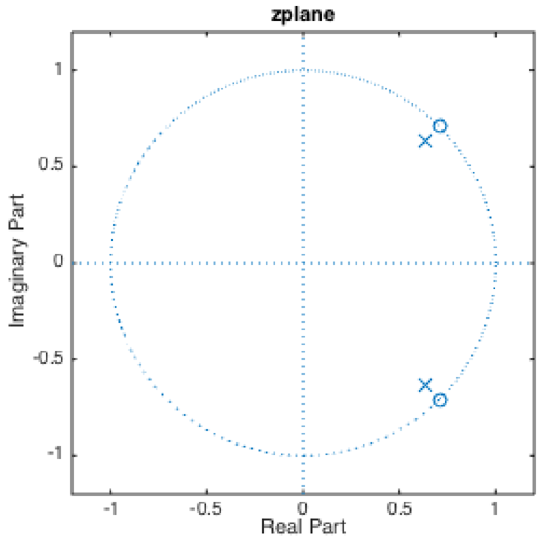
- Bandpass filter (on frequency ϕ)
  - No zeros
  - Poles close to $\omega=\phi \rightarrow \rho e^{\pm i\varphi}$ $(\rho<1)$
  - $H(z)=1/[(1-\rho e^{i\varphi} z^{-1})(1-\rho e^{-i\varphi} z^{-1})]$

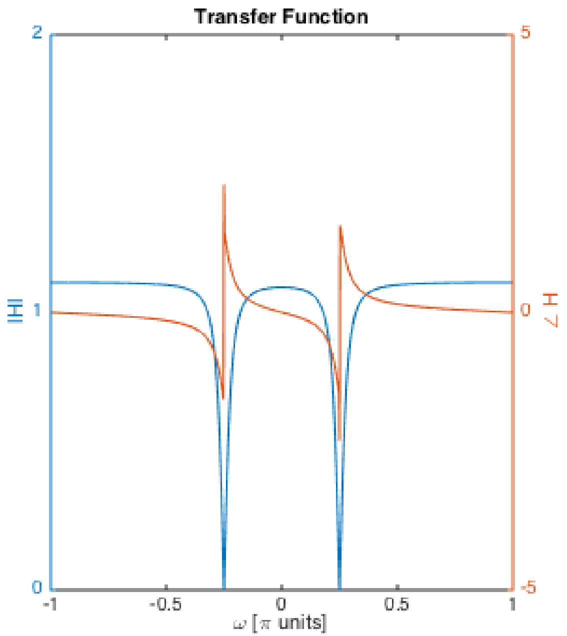




Filters

- Stopband (notch) filter (on frequency ϕ)
  - Zeros at $\omega=\phi \rightarrow e^{\pm i\varphi}$
  - Poles close to $\omega=\phi \rightarrow \rho e^{\pm i\varphi}$ $(\rho<1)$
  - $H(z)=[(1-e^{i\varphi}\, z^{-1})\,(1-e^{-i\varphi}\, z^{-1})]/[(1-\rho e^{i\varphi}\, z^{-1})(1-\rho e^{-i\varphi}\, z^{-1})]$





Minimum Phase Filters

Minimum phase filters

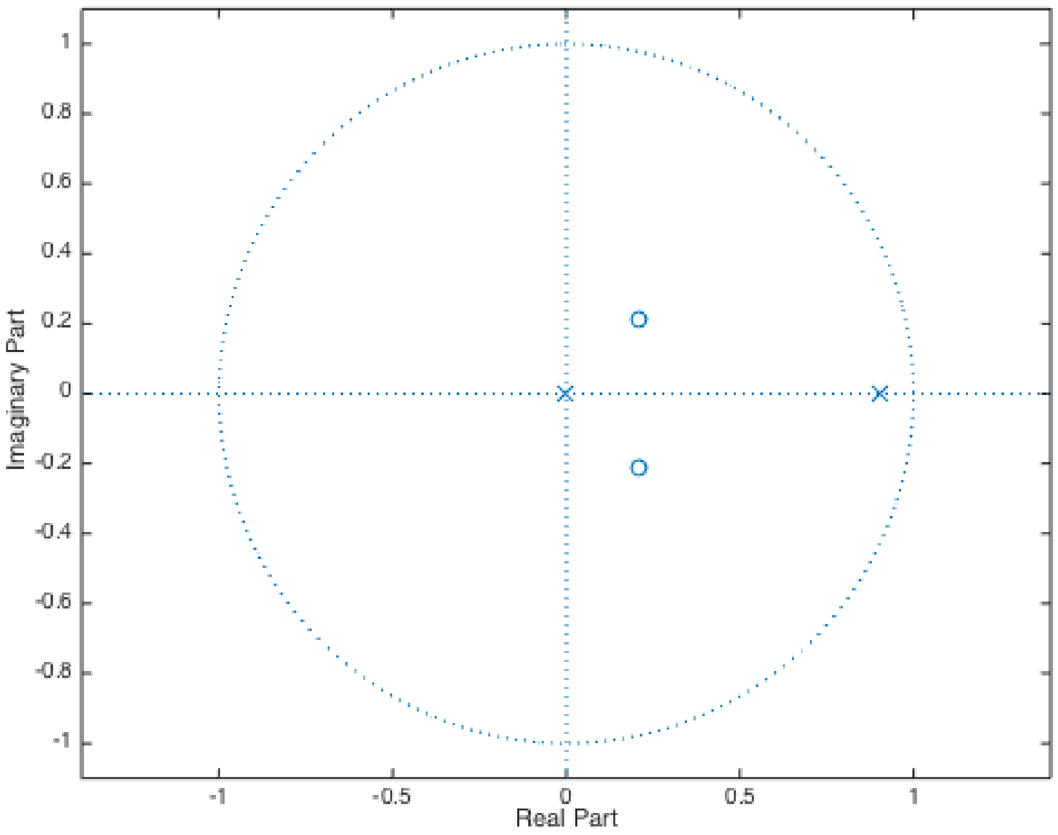
- A LTI filter H(z)=B(z)/A(z) is called "minimum phase filter" if all of its poles and zeros are inside the unit circle
  - poles → the filter is stable
  - zeros → the inverse filter H^{-1}=A(z)/B(z) is stable
- Given a second order LTI filter H(z)=B(z)/A(z)
  - B=[1,c]
  - A=[1 –a –b]
- Let's find a, b, c that implement a minimum phase filter

Minimum phase filters

- $-c$ is the zero → $|c| < 1$
- $1-az^{-1}-bz^{-2}=0$ → $(p_1z^{-1}-1)(p_2z^{-1}-1)=0$
  - $1-(p_1+p_2)z^{-1}+(p_1p_2)z^{-2}=0=1-az^{-1}-bz^{-2}$
  - $p_1+p_2=a$
  - $-p_1p_2=b$
- Choose $|p_1|<1, |p_2|<1$

```
z1=0.9; c=-z1;
p1=0.3.*exp(1i*pi/4); p2=conj(p1);
a=p1+p2; b=-p1*p2;
A=[1 c]; B=[1, -a, -b];
figure; zplane(B,A);
```

Minimum phase filters



Allpass Filter

Allpass filter

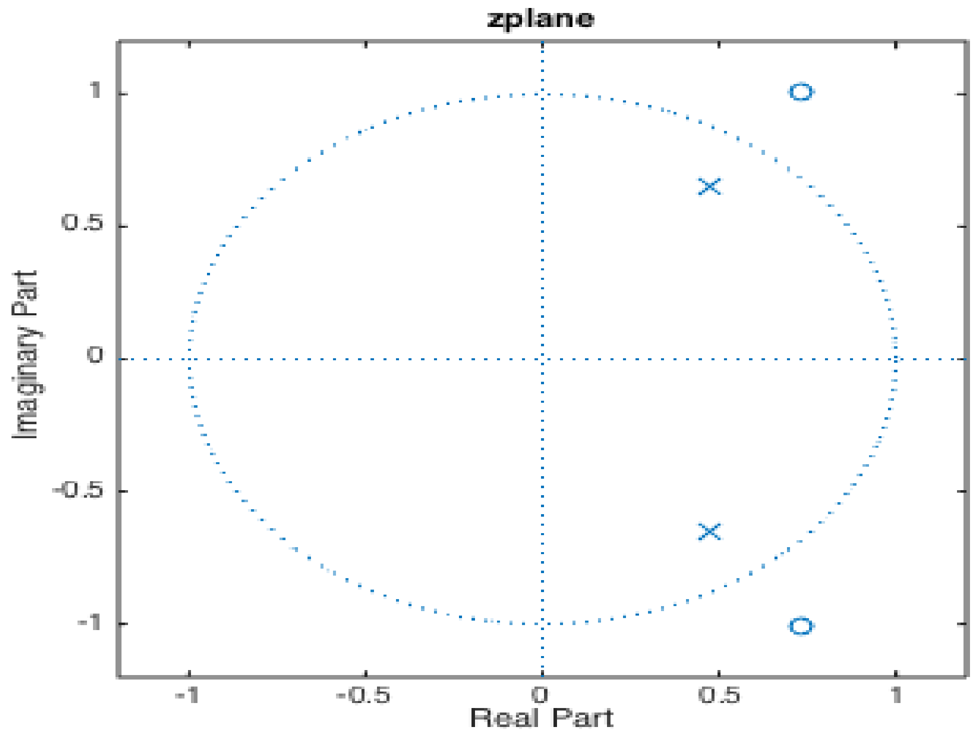
- An allpass filter passes all frequencies with equal gain: $|H(\omega)|$=constant
- Given the denominator difference equation a(n), with poles p_1, p_2, p_3, …
  - b(n) must have zeros at $z_b=1/p_b^*$

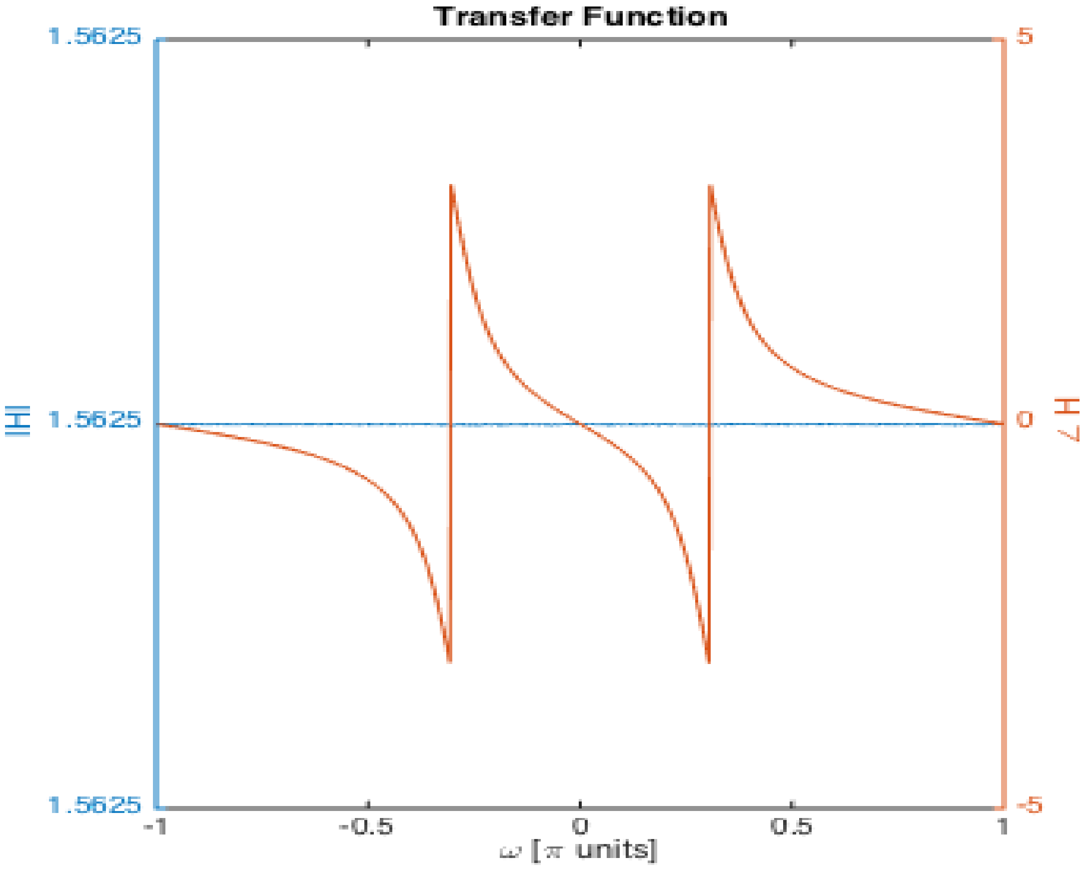
```
rho=0.8; phi=0.3*pi; p1=rho*exp(1i*phi);
p=[p1, conj(p1)]; z=1./conj(p);
```

  - $b_i(n)=a^*_{N-i}$ → conjugated and folded version of a(n)

```
a=poly(p); b=fliplr(conj(a));
```

Allpass filter

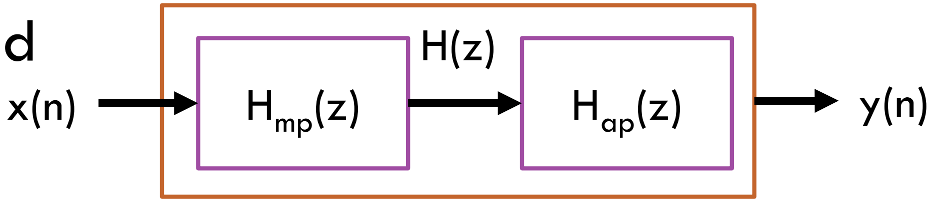




Minimum-phase and allpass decomposition

Minimum phase/allpass decomposition

- A causal **stable** filter H(z) can always be decomposed into its minimum phase $H_{mp}(z)$ and allpass $H_{ap}(z)$ components

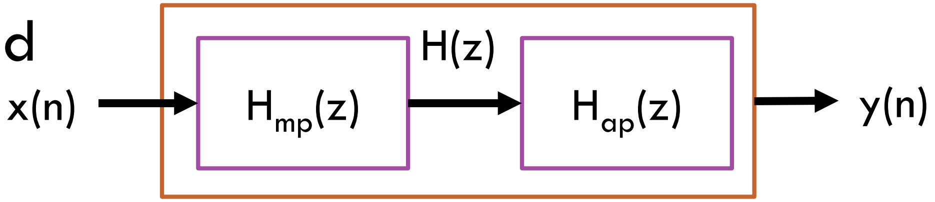


- Assumption: H(z) is **stable** → poles *p* are all in the unit circle

$$H(z) = \frac{\prod_k (1 - z_k z^{-1})}{\prod_i (1 - p_i z^{-1})}$$

Minimum phase/allpass decomposition

- A causal **stable** filter H(z) can always be decomposed into its minimum phase $H_{mp}(z)$ and allpass $H_{ap}(z)$ components

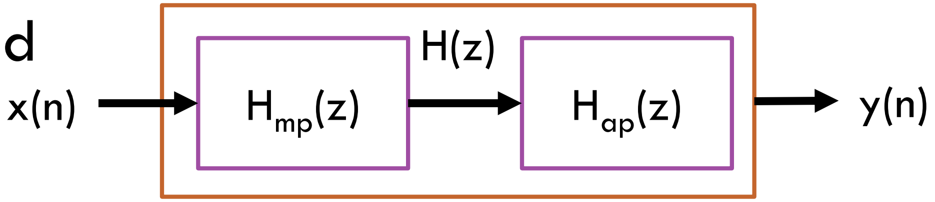


- Assumption: H(z) is **stable** → poles *p* are all in the unit circle
- Split the zeros **z** into
  - minimum phase $|z_{min}|<1$
  - maximum phase $|z_{max}|>1$

$$H(z) = \frac{\prod_{u \in z_{min}}(1 - z_u z^{-1}) \prod_{v \in z_{max}}(1 - z_v z^{-1})}{\prod_i (1 - p_i z^{-1})}$$

Minimum phase/allpass decomposition

- A causal **stable** filter H(z) can always be decomposed into its minimum phase $H_{mp}(z)$ and allpass $H_{ap}(z)$ components



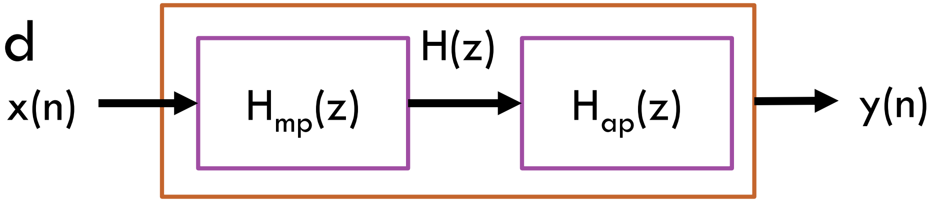
- Assumption: H(z) is **stable** → poles *p* are all in the unit circle
- Split the zeros **z** into
  - minimum phase $|z_{min}|<1$
  - maximum phase $|z_{max}|>1$

$$H(z) = \frac{\prod_{u \in z_{min}} (1 - z_u z^{-1}) \prod_{v \in z_{max}} (1 - z_v z^{-1})}{\prod_i (1 - p_i z^{-1})}$$

- $H_{ap}(z)$ is composed of $z_{ap}=z_{max}$ and $p_{ap}=1/z^*_{max}$
  - H_{ap} is still stable, but we need to compensate for the extra poles p_{ap}

Minimum phase/allpass decomposition

- A causal **stable** filter H(z) can always be decomposed into its minimum phase $H_{mp}(z)$ and allpass $H_{ap}(z)$ components



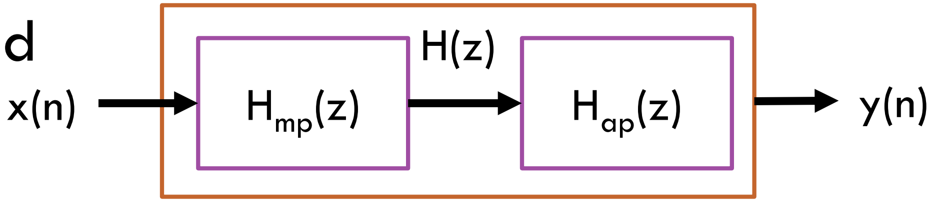
- Assumption: H(z) is **stable** → poles *p* are all in the unit circle
- Split the zeros **z** into

$$H(z) = \frac{\prod_{u\in z_{min}}(1-z_u z^{-1})}{\prod_i(1-p_i z^{-1})} \boxed{\frac{\prod_{v\in z_{max}}(1-z_v z^{-1})}{\prod_{v\in z_{max}}(1-\frac{1}{z_v^*} z^{-1})}} \prod_{v\in z_{max}}(1-\frac{1}{z_v^*} z^{-1})$$

- $H_{ap}(z)$ is composed of $z_{ap}=z_{max}$ and $p_{ap}=1/z^*_{max}$
  - H_{ap} is still stable, but we need to compensate for the extra poles p_{ap}

Minimum phase/allpass decomposition

- A causal **stable** filter H(z) can always be decomposed into its minimum phase $H_{mp}(z)$ and allpass $H_{ap}(z)$ components



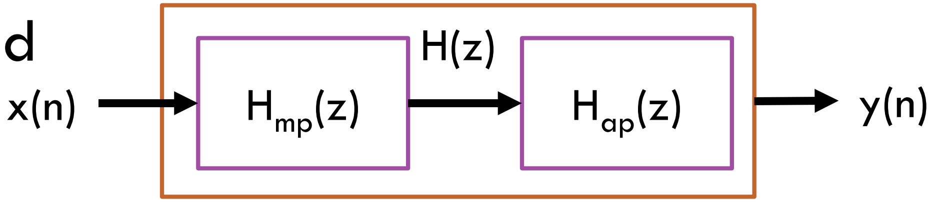
- Assumption: H(z) is **stable** → poles *p* are all in the unit circle
- Split the zeros **z** into

$$H(z) = \boxed{\frac{\prod_{u \in z_{min}} (1 - z_u z^{-1}) \prod_{v \in z_{max}} (1 - \frac{1}{z_v^*} z^{-1})}{\prod_i (1 - p_i z^{-1})}} H_{ap}(z)$$

- H_{mp} filter is composed of $z_{mp}=[z_{min}, 1/z^*_{max}]$; $p_{mp}=p$
  - H_{mp} is stable AND invertible

Minimum phase/allpass decomposition

- A causal **stable** filter H(z) can always be decomposed into its minimum phase $H_{mp}(z)$ and allpass $H_{ap}(z)$ components



$$H(z) = H_{mp}(z)H_{ap}(z)$$

$$|H^{-1}(z)| = \left|H_{mp}^{-1}(z)\right| \cdot \left|H_{ap}(z)\right|$$

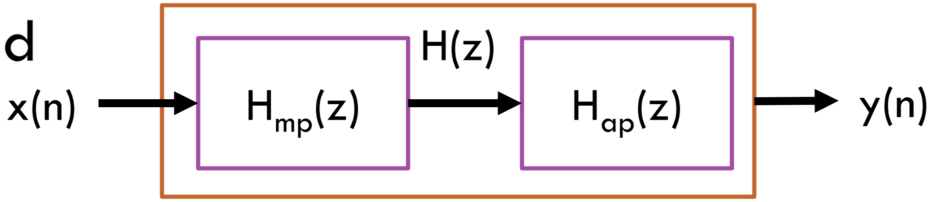
$$p_{max} \stackrel{\text{def}}{=} {}^{1}/_{z_{max}^{*}}$$

$$H_{mp}(z) = \frac{\prod_{u \in z_{min}}(1 - z_u z^{-1}) \prod_{v \in p_{max}}(1 - p_v z^{-1})}{\prod_i (1 - p_i z^{-1})}$$

$$H_{ap}(z) \frac{\prod_{v \in z_{max}}(1 - z_v z^{-1})}{\prod_{v \in p_{max}}(1 - p_v z^{-1})}$$

Minimum phase/allpass decomposition

- A causal **stable** filter H(z) can always be decomposed into its minimum phase $H_{mp}(z)$ and allpass $H_{ap}(z)$ components



$$H(z) = H_{mp}(z)H_{ap}(z)$$

$$|H^{-1}(z)| = \left|H_{mp}^{-1}(z)\right| \cdot \left|H_{ap}(z)\right|$$

$$p_{max} \stackrel{\text{def}}{=} {}^{1}\!/_{z_{max}^{*}}$$

$$H_{mp}(z) = \frac{\prod_{u \in z_{min}}(1 - z_u z^{-1}) \prod_{v \in p_{max}}(1 - p_v z^{-1})}{\prod_i (1 - p_i z^{-1})}$$

$$H_{ap}(z) \frac{\prod_{v \in z_{max}}(1 - z_v z^{-1})}{\prod_{v \in p_{max}}(1 - p_v z^{-1})}$$

Test with

$$p = \left\{0.9; 0.8e^{\pm i\frac{\pi}{2}}\right\}$$

$$z = \left\{2; 3e^{\pm i\frac{\pi}{8}}; 0.5e^{\pm i\frac{\pi}{4}}\right\}$$

Minimum phase/allpass decomposition


```
z=[2; 3*exp(1i*pi/8); 3*exp(-1i*pi/8);
   0.5*exp(1i*pi/4); 0.5*exp(-1i*pi/4)];
p=[0.9; 0.8*exp(1i*pi/2); 0.8*exp(1i*pi/2)];
b=poly(z); a=poly(p); [H,f]=freqz(b,a);

z_min=z(abs(z)<1); z_max=z(abs(z)>=1);

p_max=1./conj(z_max); b_ap=poly(z_max);
a_ap=poly(p_max); H_ap=freqz(b_ap,a_ap);

z_mp=[z_min; p_max]; b_mp=poly(z_mp); a_mp=a;
H_mp=freqz(b_mp,a);

H_=H_mp.*H_ap;
```

Minimum phase/allpass decomposition