Digital Signal Processing
All-Pass Filter Design Example

D. Richard Brown III

Problem Setup

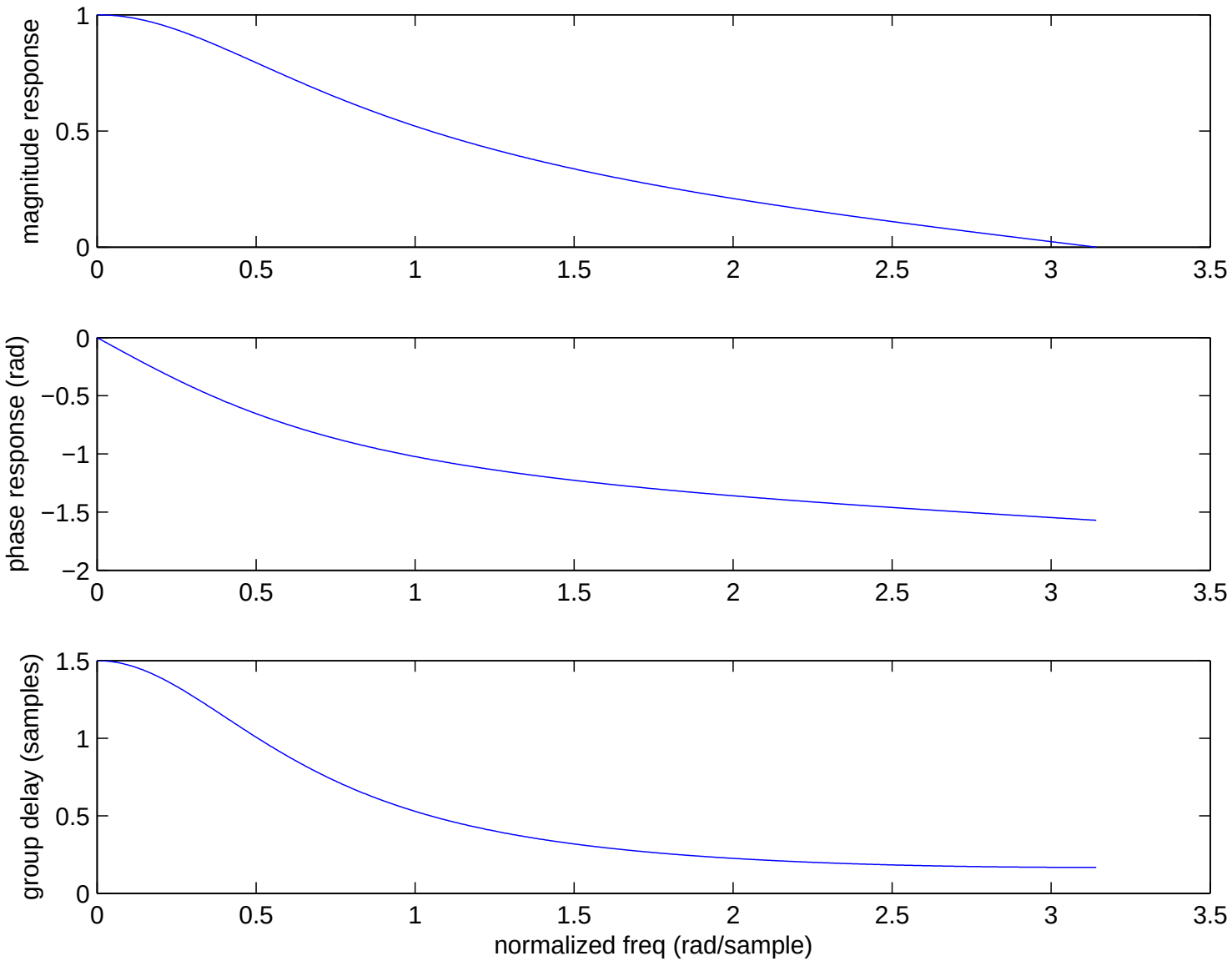
We are given a causal stable IIR low-pass filter

$$H(z) = \frac{\frac{1-\alpha}{2}(1 + z^{-1})}{1 - \alpha z^{-1}}$$

with $\alpha = 0.5$. We can realize and analyze this filter in Matlab with

```
w = pi*[0:0.001:1];          % normalized frequencies
alpha = 0.5;                 % LPF parameter
blp = (1-alpha)/2*[1 1];     % numerator coefficients
alp = [1 -alpha];            % denominator coefficients
hlp = freqz(blp,alp,w);      % compute DTFT
glp = grpdelay(blp,alp,w);   % compute group delay
```

Low-Pass Filter Frequency Response

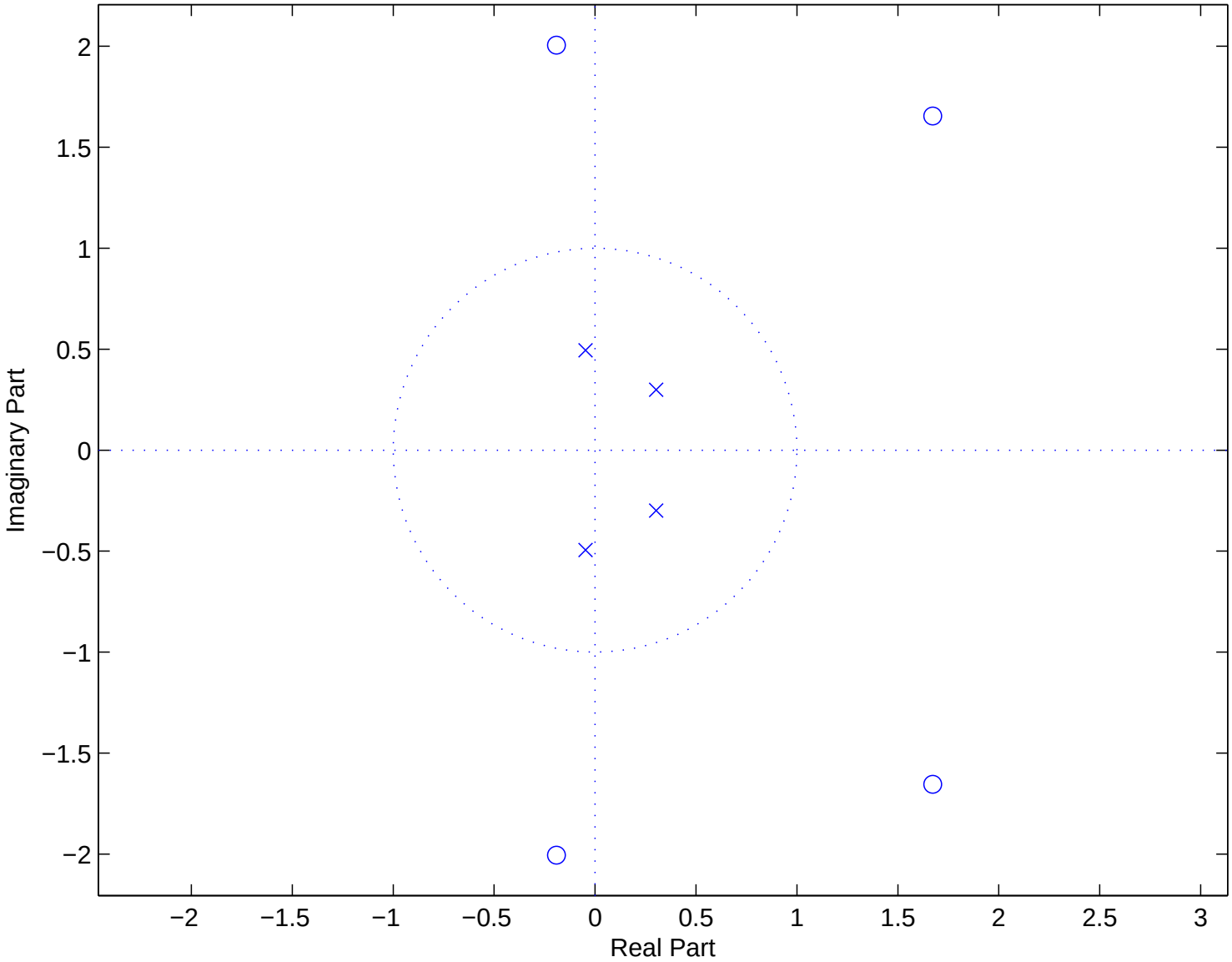


All-Pass Filter Design to Linearize Phase Response

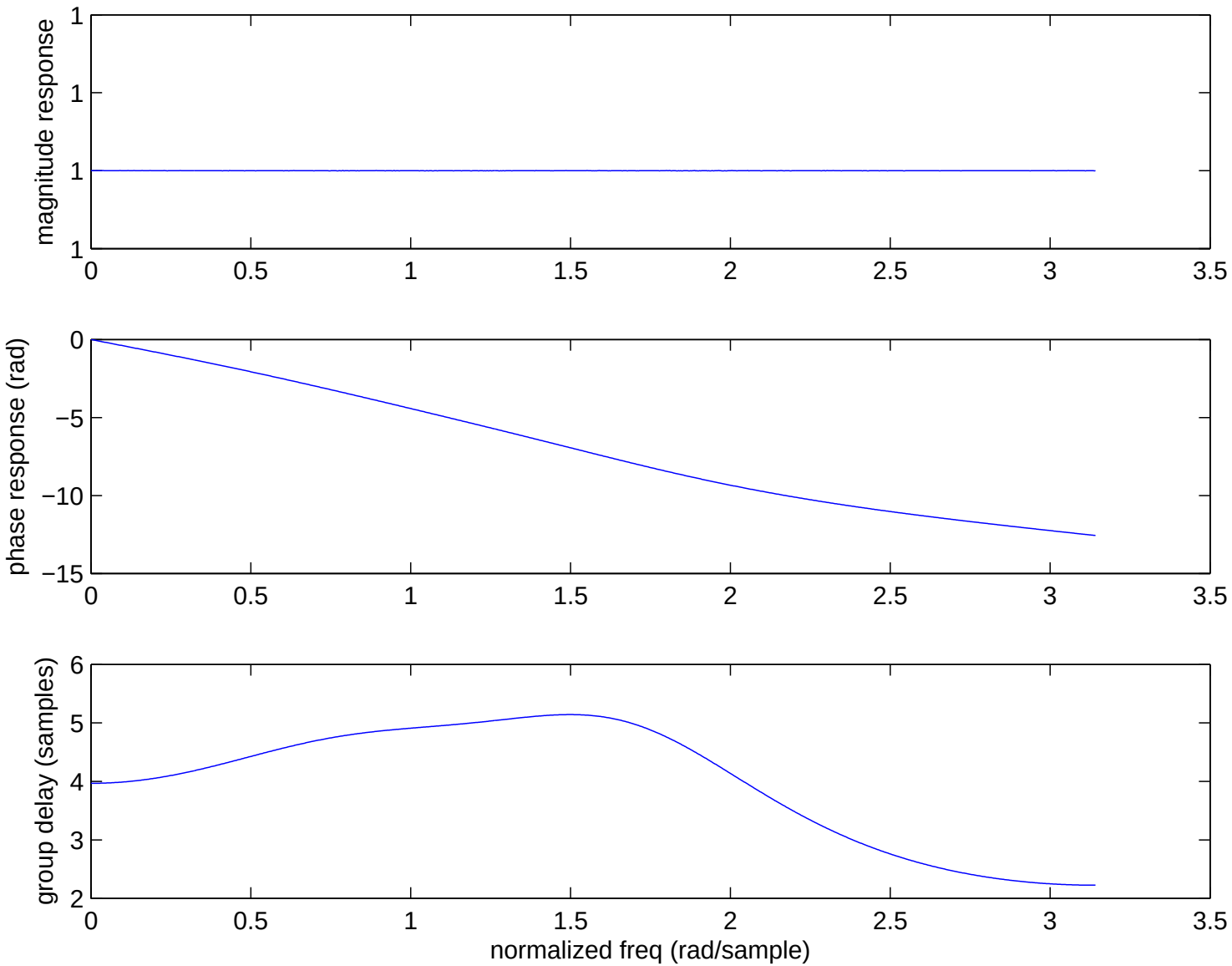
We will cascade a causal all-pass filter with the LPF to linearize the phase response and equalize the group delay in the passband. We can do this in Matlab with

```
N = 4;                          % all-pass filter order
F = w(1:501)/pi;                % normalized frequencies
edges = [0 1/2];                % band-edge frequencies
Gd = max(glp)-glp(1:501);  % desired group-delays of APF (>0)
[bap,aap] = iirgrpdelay(N,F,edges,Gd); % make all-pass filter
hap = freqz(bap,aap,w);         % compute DTFT
gap = grpdelay(bap,aap,w); % compute group-delay
```

All-Pass Filter z-plane



All-Pass Filter Frequency Response



Compute and Plot Cascaded Response

We can do this in Matlab with

```
b = conv(blp,bap);     % product of numerators
a = conv(alp,aap);     % product of denominators
h = freqz(b,a,w);      % compute DTFT
g = grpdelay(b,a,w);   % compute group delay
subplot(3,1,1)
plot(w,abs(h));
ylabel('magnitude response');
subplot(3,1,2);
plot(w,unwrap(angle(h)));
ylabel('phase response (rad)');
subplot(3,1,3);
plot(w,g);
xlabel('normalized freq (rad/sample)');
ylabel('group delay (samples)');
```

Cascaded LPF-AP Frequency Response