IIR Filter Design

This lecture describes the design of IIR filters using the bilinear z-transform.
After this lecture you should be able to: (1) identify the amplitude response characteristics of the four basic s-domain analog low-pass filters, (2) use the bilinear z-transform to convert an s-domain transfer function to the z-domain, (3) "prewarp" digital filter frequency specifications to obtain the design frequencies for analog filter design, and (4) use Matlab to design any of the four standard digital IIR filters.
This material is covered is sections 7.4.6 to 7.4.9 of the text.

IIR Filters

IIR filters can achieve sharper transitions between passband and stopband than FIR filters of the same order because of the "peaked" responses created by poles.

The most common technique for designing IIR filters is to transform an existing analog filter design into a digital IIR filter. Closed-form expressions for the transfer functions of analog filter designs with the more commonly required amplitude responses have been derived.

Standard Analog Filters

The four basic analog filter types are: Butterworth, Chebyschev type I and II, and Elliptic. The Butterworth filter has a monotonically decreasing (no ripple) amplitude response in both the passband and the stopband. The Chebyshev (type I) filter has ripples in the passband but not in the stopband. The Chebyshev type II filter has ripple in the stopband but not in the passband. The Elliptic filter has ripple in both passband and stopband. The steepness of the amplitude response in the transition region is least for the Butterworth and highest for the Elliptic.

The Bilinear z-Transform

We often need to design digital filters that have the same useful characteristics as one of the above analog filters (i.e. bounded ripple or monotonic response in the stopband or passband).

The most effective way to do this is to use a mapping from the s-domain to the z-domain that retains the important amplitude response characteristics of the analog filter's transfer function. The most common mapping is the *bilinear z-transform*[1], which involves substituting:

$$s = \frac{2}{T}\frac{z-1}{z+1}$$

.

Exercise: Use the bilinear z-transform to convert an analog filter with a transfer function $H(s) = \frac{1}{2+s}$ into an IIR filter with a similar amplitude response.

The most important characteristic of the bilinear z-transform mapping is that it is a *conformal* mapping which means that the shape of the amplitude response (e.g. passband ripple, stopband attenuation, etc.) is not altered.

A second effect is that the frequency response is "warped." The effect is that the response of the *digital* filter at a frequency ω maps to the response of the *analog* filter at a frequency[2]:

$$\Omega = \frac{2}{T}\tan\left(\omega\frac{T}{2}\right)$$

This means the analog frequency range $\Omega = -\infty \ldots \infty$ is maps to the range $\omega = -\pi \ldots \pi$.

Exercise: If we want to design a digital IIR filter using the bilinear z-transform method with a -3dB frequency of 30 Hz and the sampling rate is 150 Hz, what would the -3dB frequency of the original analog filter have to be?

The bilinear z-transform also maps the left hand s-plane into the inside of the z-domain unit circle. This ensures that stable analog filters are transformed into stable digital filters.

[1]The bilinear z-transform is not really a z-transform.

[2]Note that the text uses the variable ω' instead of the conventional Ω.

IIR Design with Matlab

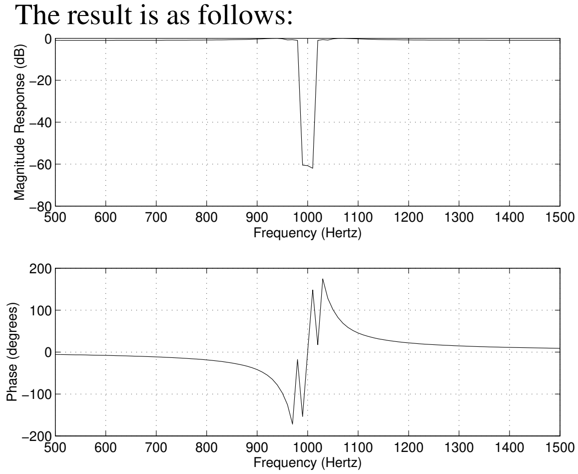
Functions are available in Matlab to design the common IIR filter types using the bilinear z-transform (`butter()`, `cheby1()`, `cheby2()`, `ellip()`). These functions pre-warp the filter frequency specifications, compute the poles and zeros of the appropriate analog filter and use the bilinear z-transform to map these to the z-plane.

For example, the following commands compute the coefficients for a 6-th order bandstop filter:

```
>> [b a]=ellip(6,1,60,[980 1020]/4000,'stop');
>> freqz(b,a,[500:10:1500],8000);
>> print -deps iirstop.eps
```

The function takes as arguments the order (6), the passband ripple (1 dB) and stopband attenuation (60 dB), the filter cutoff frequencies (normalized to the Nyquist frequency, $F_s/2$), and an optional parameter that specifies the type of filter (bandstop in this case).

The result is as follows:



Comparing this with the FIR filter with the same specifications designed in the previous lecture shows the advantages and disadvantages of of IIR filters: the 6-th order elliptic IIR filter has a better amplitude response than the 80-th order FIR filter at the expense of a distinctly non-linear phase response.