Creative Computing II

Finite Impulse Response filters

9th March 2010

This lab sheet introduces the *Octave* `fir1` function for filter design, and covers how to use the filters it produces.

1. This part covers the interface of the `fir1` operator, and how to visualize the filter it produces.

   (a) construct the 100th-order low-pass FIR filter with a cut-off of 475Hz for a sample rate of 44.1kHz; plot the resulting filter's characteristics using `freqz`.

   The second argument to `fir1` *is a band edge frequency as a proportion of the Nyquist frequency for the input signal. For a 44.1kHz signal, the Nyquist frequency is 22050Hz, so this should be called as something like*

   ```
   a = fir1(100,475/22050);
   freqz(a)
   ```

   (b) construct the 100th-order high-pass FIR filter with a cut-off of 425Hz for a sample rate of 44.1kHz; plot the resulting filter's characteristics using `freqz`.

   To get a high-pass FIR filter from `fir1`*, the third argument should be the string 'high', so*

   ```
   b = fir1(100,425/22050);
   freqz(b)
   ```

   Note in both this part and the previous one the gain is about 1 (0dB) in the pass band, and decreases rapidly to a low value (large negative value in decibels) in the stop band. Also note the linear phase change over the pass band.

   (c) convolve the two filters' (from parts 1a and 1b) responses together; plot the resulting filter's characteristics using `freqz`.

   Because of the associativity of convolution, you can think of applying two systems one after the other as the same as applying a single system made up of the convolution of the impulse responses of the original two. Thus, convolving the two IRs from the parts above gives the filter resulting from the successive application of the previous two: `freqz(conv(a,b))`*.*

   (d) construct the appropriate-order band-pass FIR filter, with the pass-band being 425Hz-475Hz for a sample rate of 44.1kHz. What are the differences between this filter and the one in part 1c?

   The convolution of the two above systems produces a new, 200th-order system, which (if the original filters were ideal) would be a band-pass filter between 425Hz and 475Hz. However, because FIR filters cannot be ideal, the convolution produces a filter with some drawbacks; in particular, the gain in the passband is substantially less than 0dB, so even the frequencies passed will be attenuated. Using `fir1` *to create the band-pass filter directly will construct a system which has better pass-band properties:*

   ```
   d = fir1(200,[425 475]/22050)
   ```

2. This part covers the use of the filters you have produced in part 1.

   (a) Read in the sound data from the provided `scale.wav` file.

       The musical scale in this file is a major scale starting at middle C (with a fundamendal frequency of 262Hz) and going up to the C an octave above (a fundamental frequency of 523Hz).

   (b) Using the `filter` function, apply the four filters produced in part 1 to the sound data, and play the resulting filtered sounds using `sound`. Check that you understand the effects of the filters, in the light of your investigations about the frequencies present in the audio file at different times. If you don't understand the effects, check that your filters are correct.

       With the low- and high-pass filters, you should find that corresponding portions of the scale are attenuated: the low-pass filter should attenuate the last three notes or so, while the high-pass filter should attentutate the first few.

   (c) Verify that you get the same results using `conv` to apply the filters to the sound data.

   (d) Perform the same filtering operation using `fft` and its inverse. Can you explain any discrepancy between the results using this method and the previous two?

3. This part applies a filter you have produced to a synthetic tone, in a similar way to the method of operation of Moog synthesizers.

   (a) Construct a frequency spectrum of 44100 elements, with ones in the 56th bin and every 55 thereafter up to the Nyquist frequency, and the high half of the spectrum being the reverse of the low half.

   (b) Construct the time-domain signal corresponding to the spectrum in part 3a.

   (c) Apply the band-pass filter you constructed in part 1d to this signal, and listen to the result.

   (d) Apply the 1000th-order FIR filter with the same parameters, and listen to the results. Describe the differences in the sound as best you can.