Implementing digital IIR filters in the lattice form

Lab 6, SDP

Objective

The students should become familiar with *lattice*-type realization structure used for implementing IIR filters

Theoretical notions

Exercises

1. Consider the causal IIR system with poles and zeros, with the system function:

$$H(z) = \frac{1 + 2z^{-1} + 3z^{-2} + 2z^{-3}}{1 + \frac{2}{5}z^{-1} + \frac{7}{20}z^{-2} + \frac{1}{2}z^{-3}}$$

   Find and draw the equivalent *lattice* structure for the IIR filter.

2. Consider the causal IIR system, with no zeros, with the following system function:

$$H(z) = \frac{1}{1 + \frac{2}{5}z^{-1} + \frac{7}{20}z^{-2} + \frac{1}{2}z^{-3}}$$

   Find and draw the equivalent *lattice* structure for the IIR filter.

3. Using the Octave software, use the `ellip()` function to design one of the following elliptic IIR filters:

   a. A low-pass IIR filter of order 4, elliptic type, with cutoff frequency of 6kHz at a sampling frequency of 44.1kHz;
   b. A high-pass IIR filter of order 4, elliptic type, with cutoff frequency of 2.5kHz at a sampling frequency of 44.1kHz;

c. A band-pass IIR filter of order 4, elliptic type, with passband between 0.5kHz and 5.5kHz at a sampling frequency of 44.1kHz.

4. Create an Octave function to `tf2latc_iir()` to compute the coefficients of the lattice form of an IIR filter, starting from the coefficients of the Transfer Function.

   Call it like this: `[K, V] = tf2lact_iir(b, a)`

   Use it to convert the filter designed in the previous exercise.

5. Create an Octave script to filter an input signal `x` with an IIR filter in lattice form, given the coefficients K and V:

   `y = filter_latc_iir(K, V, x)`

6. Use the function above to load and low-pass the audio signal `Kalimba.mp3`.

   a) Load the file using `audioread()`
   b) Filter the signal using `filter_latc_iir()`, with the previously designed filter

Final questions

1. TBD