KTH, INFORMATION SCIENCE AND ENGINEERING

SCHOOL OF ELECTRICAL ENGINEERING AND COMPUTER SCIENCE

Digital Signal Processing EQ2300 / 2E1340

Final Examination 2020–01–11, 09:00-14:00

Literature:

- Jaldén: *Summary notes for EQ2300* (30 pages printed material). Shorter handwritten notes in the booklet are allowed.
- *Beta – Mathematics Handbook*
- *Collection of Formulas in Signal Processing, KTH.*
- *One A4 of your own notes.* You may write on both sides, and it does not have to be hand written, but cannot contain full solutions to tutorial problems or previous exam problems.
- An unprogrammed pocket calculator.

Notice:

- Answer in English or Swedish.
- At most one problem should be treated per page.
- Answers without motivation/justification carry no rewards.
- Write your name and *personnummer* on each page.
- Write the number of solution pages on the cover page.

The exam consists of five problems with a maximum of 10 points each. 24 points is required for a passing grade (E).

Contact: Joakim Jaldén, Dept. of Information Science and Engineering, 08-790 77 88

Results: Will be reported within three working weeks on “My pages”.

Solutions: Will be available on the course homepage after the exam.

Good luck!

1. We begin with some mixed problems. Remember to motivate your answers.

a) Let $x[n] = \{1, 2, 3, 0\}$ and $h[n] = \{ 1, 1, 1, 0\}$, and compute the 4-point circular convolution $y[n] = h[n] \, \textcircled{4} \, x[n]$. (2p)

b) Let $x[n] = \{ 1, -1, 1, -1, 1, -1, 1, -1\}$, and compute the 8-point discrete Fourier transform (DFT) $X_8[k]$ of $x[n]$. (2p)

c) Ler $x[n]$ be an AR1 stochastic process with autocorrelation $r_x[k]$, where $r_x[0] = 1$ and $r_x[1] = \frac{1}{2}$. Obtain the power spectrum $P_x(\nu)$ of $x[n]$. (3p)

d) Show that the convolution $h[n] = h_1[n] * h_2[n]$ of two Type I linear phase FIR filters $h_1[n]$ and $h_2[n]$ is always another Type I linear phase FIR filter. (3p)

2. It has been noted during the lectures that Blackman-Tukey's method is the spectrum estimation method which makes the most efficient use of the available data out of the non-parametric methods covered. In this problem, we will look into this with an explicit example. To this end, assume that we have $N = 256$ samples $x[n]$ from a wide sense stationary stochastic process and that we wish to estimate the power spectral density $P_x(\nu)$ of $x[n]$. We will compare the performance of Bartlett's method and Blackman-Tukey's method.

a) Assume that we divide the $N = 256$ samples into blocks of length $L = 16$ in Bartlett's method.

   i. Compute the variance of the Bartlett's method, expressed in terms of the power spectrum $P_x(\nu)$. (1p)

   ii. Compute the (3 dB) spectral resolution $\Delta\nu$ of Bartlett's method. (2p)

b) Assume that we use a Bartlett (triangular) window of max-lag M in Blackman-Tukey's method. Compute a value of M such that the bias of Blackman-Tukey's method is (approximately) the same as that for Bartlett's method, i.e., such that

$$\mathrm{E}\{\hat{P}_x^{\mathrm{BT}}(\nu)\} \approx \mathrm{E}\{\hat{P}_x^{\mathrm{B}}(\nu)\}$$

where $\hat{P}_x^{\mathrm{BT}}(\nu)$ is Blackman-Tukey's spectrum estimate and where $\hat{P}_x^{\mathrm{B}}(\nu)$ is Bartlett's spectrum estimate. (3p)

c) Obtain the variance of Blackman-Tukey's method for this value of M and compare it to the value obtained for Bartlett's method. By doing so you should be able conclude that the variance of Blackman-Tukey's method is approximately $\frac{2}{3}$ times that of Bartlett's metod at the same spectral resolution. (2p)

d) Let $\hat{r}_x^{\mathrm{B}}[k]$ and $\hat{r}_x^{\mathrm{BT}}[k]$ as the autocorrelation estimates obtained from the inverse discrete Fourier transform of the corresponding spectrum estimates $\hat{P}_x^{\mathrm{B}}(\nu)$ and $\hat{P}_x^{\mathrm{BT}}(\nu)$. Show that the estimates of the autocorrelation $r_x[k]$ will be the same for both estimators at time lag $k = 0$, i.e., $\hat{r}_x^{\mathrm{B}}[0] = \hat{r}_x^{\mathrm{BT}}[0]$. From this it can be concluded that the improved performance of Blackman-Tukey's metod must be that it obtains better estimates of $r_x[k]$ for larger values of k. (2p)

3. Assume that we are designing a real time implementation of an FIR filter on a hardware platform such as the Arduino Due used in the lab. The signal we are filtering is sampled at 1 kHz. Our hardware platform can compute 10^5 real valued multiplications per second, and the multiplications are the limiting factor of the platform. If the filter is implemented as a direct convolution in the time domain, i.e.,

$$y[n] = \sum_{m=0}^{M-1} h[m]x[n-m]\,,$$

the maximum FIR filter length that can be used is therefore $M = 100 = 10^5/10^3$, as we need to finish the computation of one sample of $y[n]$ before the next sample of $x[n]$ arrives 1 ms later. We can however implement much longer filters if we filter in the frequency domain using overlap save or overlap add.

a) Show that it will be possible to implement more than 10 times longer filters using overlap save or overlap add, i.e., FIR filters with $M = 1000$, even if we assume that each complex valued multiplication requires 4 real valued multiplications. Propose a suitable FFT length N for the implementation. (2p)

b) Assume that we due to memory and latency constraints cannot use an FFT longer than $N = 1024$. What is then the longest filter than can be implemented if we again assume that each complex valued multiplication requires 4 real valued computations. (3p)

c) How many FFT operations (FFTs and and inverse FFTs) will be computed per second if we implement a length $M = 525$ filter using overlap save with an FFT length of $N = 1024$. (2p)

d) For the values of $M = 525$ and $N = 1024$ used in part c), and assuming that the platform can only perform one multiplication at a time, what will the latency of an overlap save implementation be? (3p)

Note for part d): By latency we mean the maximum time that will pass between a sample $x[n]$ becoming available as input to the filter until the corresponding sample $y[n]$ has been computed. The direct implementation of a filter of length $M = 100$ will have a latency of 1 ms. We need $x[n]$ to compute $y[n]$ in the direct implementation, and after $x[n]$ is available the system requires 100 multiplications to compute $y[n]$. If the system is capable of performing 10^5 multiplications per second this takes $100/10^5\,\mathrm{s} = 10^{-3}\,\mathrm{s}$ or $1\,\mathrm{ms}$. We will be done just as the next sample of $x[n]$ becomes available. For the latency of the overlap save you need to take into account both the time required to collect all samples needed for the computation, as well as the time required for the computation itself.

4. The transfer function of a general N-th order IIR filter with real valued coefficients is given by

$$H(z) = \frac{B(z)}{A(z)} = \frac{b_0 + b_1 z^{-1} + \dots b_N z^{-N}}{a_0 + a_1 z^{-1} + \dots a_N z^{-N}}.$$

In case the numerator and denominator coefficients have a mirror symmetry, i.e., if $b_n = a_{N-n}$ for $n = 0, \dots, N$, the filter is a so-called all-pass filter. In this case we can write the transfer function as

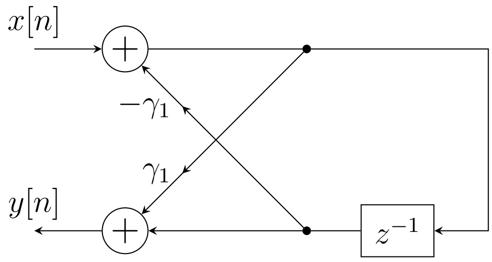
$$H(z) = \frac{z^{-N} A(z^{-1})}{A(z)}$$

where $A(z) = 1 + a_1 z^{-1} + \dots + a_N z^{-N}$, from which it can be shown that

$$\left| H(z = e^{j2\pi\nu}) \right| = 1$$

for all ν. In other words, for a sinusoidal input the filter will only affect the phase of the signal and not its amplitude. In this problem, we consider all-pass filters implemented using fixed point arithmetics with a $B+1$ signed magnitude representation of the range $(-1, 1)$.

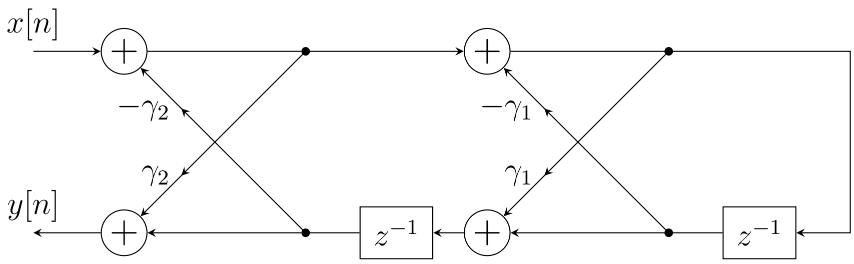
a) The circuit below implements a first order all-pass filter from $x[n]$ to $y[n]$ for any real valued constant γ_1, where $|\gamma_1| < 1$.



Obtain the impulse response and transfer function of the filter above, and show that it is an all-pass filter. (3p)

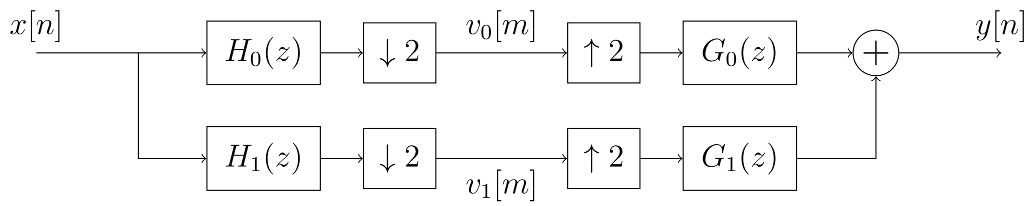
b) Compute the total power of the fixed point quantization noise at the output $y[n]$ for the first order all-pass filter above. (5p)

c) Compute the total power of the fixed point quantization caused by the multiplication with $-\gamma_2$ at the output $y[n]$ for the second order all-pass filter below, where γ_n is real valued and satisfy $|\gamma_n| < 1$ for $n = 1, 2$. You do not have to prove that it is an all-pass filter in this case: It is. You also do not have to compute the power of noise caused by any other multiplications than that of $-\gamma_2$ (2p)

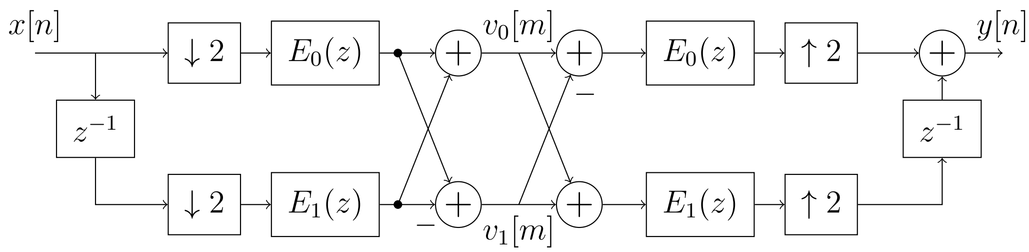


5. It is difficult to construct higher order QMF filterbanks with perfect reconstruction and good separation of low and high frequencies. However, if we only require that the overall filter-bank acts as an all-pass filter, where we do not control the phase, the design will become easier. We can in this case use IIR filters with a rapid transition from pass to stop band and good suppression in the stop band.

To this end, we will study the filter-bank given by



and the efficient polyphase implementation given by



a) Obtain expressions for $H_0(z)$, $H_1(z)$, $G_0(z)$, and $G_1(z)$ in terms of $E_0(z)$ and $E_1(z)$ such that the two implementations are equivalent. (3p)

b) Show that the z-transform of $y[n]$ is given by

$$Y(z) = T(z)X(z)$$

where $T(z) = 2z^{-1}E_0(z^2)E_1(z^2)$. In other words, if $E_0(z)$ and $E_1(z)$ are all-pass filters (with gain $\frac{1}{\sqrt{2}}$), then $T(z)$ will be an all-pass filter with unit gain. (4p)

c) Let $E_0(z)$ and $E_1(z)$ be first order all-pass IIR filters with transfer functions

$$E_0(z) = \frac{1}{\sqrt{2}} \times \frac{c_0 + z^{-1}}{1 + c_0 z^{-1}} \quad \text{and} \quad E_1(z) = \frac{1}{\sqrt{2}} \times \frac{c_1 + z^{-1}}{1 + c_1 z^{-1}} \ .$$

Obtain the filter coefficients a_n and b_n, and the system order N, of the low pass filter $H_0(z)$, such that (3p)

$$H_0(z) = \frac{B(z)}{A(z)} = \frac{\sum_{n=0}^{N} b_n z^{-n}}{\sum_{n=0}^{N} a_n z^{-n}} \ .$$

Example with $c_0 = 0.237$ and $c_1 = 0.715$