SOLUTIONS

E 94 **Digital Signalbehandling**, 2E1340

Final Examination 2004–08–27, 14.00–19.00

1. Note that the difference frequency, f_{diff} shows up as the frequency of the envelop (the outer shape) of the curves, as shown by the solid line in Figure 1.

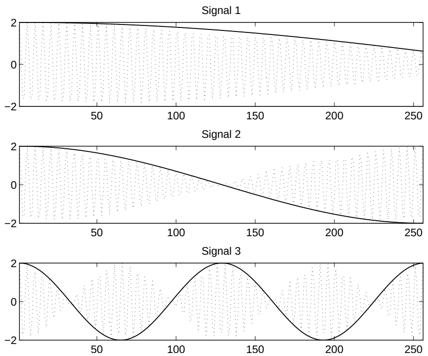

Figure 1: The envelop, $\cos(2\pi f_{\text{diff}} n)$, of the three signals.

a) The resolution limit for the periodogram is of the order of $|f_1 - f_2| \gtrsim 1/N$, corresponding to $f_{\text{diff}} \gtrsim 1/(2N)$. This means that the N signal samples should cover at least half a period of $\cos(2\pi f_{\text{diff}} n)$. In Signal 3, we can see 2 full periods of the envelope, so there should be no problems to resolve the two frequencies. In Signal 2, we see half a period of the envelope, so we are just at the resolution limit (since the more exact resolution limit is 0.89/M, we are on the safe side). In Signal 1, finally, we have less than a quarter of a period, so it should not be possible to resolve the two frequencies.

b) The resolution is reduced by 16 if we use Bartlett's method with 16 segments, so we would need to see at least 16 half periods = 8 full periods of the envelope to be able to resolve the frequencies. So, it is not possible to resolve the two frequencies in any of the three signals.

c) As described above, when we plot the signal, the envelope should look like at least half a period of a sinusoid, preferably much more. The intuitive explanation is that we have to be able to determine the difference frequency from the available data. In Signal 1, for example, it is clearly very difficult to determine that frequency since the envelope is almost constant over all the samples.

This is closely related to how a piano tuner determines if two strings are in tune. The amplitude variations (the envelop) caused by the factor $\cos(2\pi f_{\text{diff}} n)$ is heard as a "beating". For strings that are should be unison or one octave apart, there should be no beats, meaning that the tuner does not experience any fluctuations in amplitude. For other intervals like thirds, fifths and sixths, he tries to get a certain number of beats per second (at least in the equal tempered scale used for pianos). Of course, the tuner has to listen long enough to be able to determine the beat rate, corresponding to that you need a sufficient number of samples for the periodogram method.

2. a) Let $H_1(z) = 1 + bz^{-1}$ and $H_2(z) = \frac{1}{1-az^{-1}}$, with $b = 0.7$ and $a = 0.23$. Now $H(z)$ can be written either as $H(z) = H_2(z)H_1(z)$ or $H(z) = H_1(z)H_2(z)$. Mathematically these are equivalent, but numerically, switching the order makes a difference. Figure 2 contains block diagrams of the two different implementations. Note that both block diagrams can be written more compactly, using only one delay element. This, however, does *not* change the numerical properties.

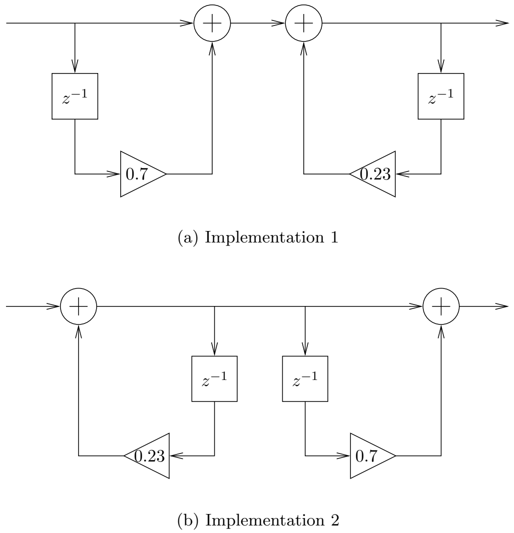

Figure 2: Two different implementations of the filter in problem 2

b) Assume that each multiplication acts as an independent white noise source. With B bits plus a sign bit, the power of each such noise source is $\sigma_e^2 = \frac{2^{-2B}}{12}$.

In implementation 1, both noise sources enter the system at the second filter stage. The noise power at the output is then (see complementary reading)

$$\sigma_{N_1}^2 = 2\sigma_e^2 \sum_{n=0}^{\infty} h_2^2(n).$$

$$H_2(z) = \frac{1}{1 - az^{-1}} \quad \Rightarrow \quad h_2(n) = a^n u(n)$$

$$\sum_{n=0}^{\infty} h_2^2(n) = \sum_{n=0}^{\infty} a^{2n} = \frac{1}{1-a^2}$$

$$\sigma_{N_1}^2 = 2\sigma_e^2 \frac{1}{1-a^2} \approx 2.11 \cdot \frac{2^{-2B}}{12}$$

In implementation 2, one noise source enters at the input of the system and the other enters directly at the output. The noise power at the output can then be written

$$\sigma_{N_2}^2 = \sigma_e^2(1 + \sum_{n=0}^{\infty} h^2(n)).$$

$$H(z) = \frac{1 + bz^{-1}}{1 - az^{-1}} \quad \Rightarrow \quad h(n) = a^n u(n) + ba^{n-1}u(n-1) = a^n(1 + \frac{b}{a})u(n) - \frac{b}{a}\delta(n)$$

$$h^2(n) = a^{2n}(1 + \frac{b}{a})^2 u(n) - \frac{2ab + b^2}{a^2}\delta(n)$$

$$\sum_{n=0}^{\infty} h^2(n) = \frac{(1 + \frac{b}{a})^2}{1 - a^2} - \frac{2ab + b^2}{a^2} = \frac{1 + 2ab + b^2}{1 - a^2}$$

$$\sigma_{N_2}^2 = \sigma_e^2(1 + \frac{1 + 2ab + b^2}{1 - a^2}) \approx 2.91 \cdot \frac{2^{-2B}}{12}$$

Implementation 1 gives the smallest round-off noise.

3. Use the overlap-save or overlap-add method, implemented with an N point DFT. If the input buffer has length M, we need $N \geq M + 1000 - 1$. The processing of one buffer will require 2 FFTs plus N complex valued multiplications, i.e. a total of $4N(1 + \log_2(N))$ real valued multiplications. So, the total processing time per buffer, including the buffer switching overhead is $T_{\text{calc}} = 4N(1 + \log_2(N))/10^8 + 10^{-7}$s, which has to be less than the time to fill the buffer, $T_{\text{fill}} = M/10^6 = (N - 999)/10^6$s.

The total delay from input to output corresponds to 2 buffer lengths, i.e. $T_{\text{in-out}} = 2 * M/10^6 = 2 * (N - 999)/10^6$s. Different values of N give

N	T_{calc}	T_{fill}	$T_{\text{in-out}}$
1024	0.451 ms	25 μs	50 μs
2048	0.983 ms	1.05 ms	2.10 ms
4096	2.13 ms	3.10 ms	6.20 ms
8192	4.59 ms	7.19 ms	14.4 ms
16384	9.83 ms	15.4 ms	30.8 ms

This shows that $N \geq 2048$ is necessary for the processor to have time to complete the calculations in time and $N \leq 4096$ to fulfill the real-time requirements, so either $N = 2048$ or $N = 4096$ will work. This corresponds to a buffer size of $M = 1049$ or $M = 3097$ samples, respectively. The shorter option is mostly to prefer, since it gives a lower system delay.

4. a) Combining

$$V_i(f) = \frac{1}{D}\sum_{k=1}^{D-1} X\Big(\frac{f-k}{D}\Big) H_i\Big(\frac{f-k}{D}\Big)$$

and

$$Y(f) = \sum_{i=0}^{D-1} F_i(f) V_i(Df) ,$$

we get

$$Y(f) = \sum_{i=0}^{D-1} F_i(f) \frac{1}{D} \sum_{k=0}^{D-1} H_i\Big(f - \frac{k}{D}\Big) X\Big(f - \frac{k}{D}\Big) .$$

To simplify the next question, it makes sense to reorder the summations to see what terms involve $X(f)$ and what involve the aliasing $X(f - k/D)$, $k \neq 0$:

$$Y(f) = \frac{1}{D}\left(\sum_{i=0}^{D-1} F_i(f) H_i(f)\right) X(f) + \frac{1}{D}\sum_{k=1}^{D-1}\left(\sum_{i=0}^{D-1} F_i(f) H_i\Big(f - \frac{k}{D}\Big)\right) X\Big(f - \frac{k}{D}\Big)$$

b) $y(n) = x(n - L)$ corresponds to $Y(f) = e^{-j2\pi Lf}X(f)$, so the factor in front of $X(f)$ should be $e^{-j2\pi Lf}$ and those in front of $X(f - k/D)$ should be zero. This gives the following set of requirements,

$$\begin{cases} \sum_{i=0}^{D-1} F_i(f) H_i(f) = De^{j2\pi fL} \\ \sum_{i=0}^{D-1} F_i(f) H_i\Big(f - \frac{k}{D}\Big) = 0, \quad k = 1, \ldots, D-1 \end{cases}$$

5. a) If we let Δ denote the time delay and S the scaling, we can write

$$x(n) = p\big((n - \Delta)S\big) + \nu(n) ,$$

where $\nu(n)$ denotes measurement noise. (An alternative parameterization is $x(n) = p(Sn - \Delta) + \nu(n)$, but then $x(\Delta/S)$ will correspond to the point $p(0)$, whereas $x(\Delta)$ corresponds to $p(0)$ in the equation above.)

b) Assume that we have N samples of $x(n)$, $\{x(n_0), x(n_0 + 1), \ldots, x(n_0 + N - 1)\}$. The most common estimation method is Least Squares (which is equivalent to Maximum Likelihood if the noise $\nu(n)$ is Gaussian),

$$\{\hat{S}, \hat{\Delta}\} = \arg\min_{S,\Delta} \sum_{n=n_0}^{n_0+N-1} |x(n) - p\big((n - \Delta)S\big)|^2$$

In general, non-linear numerical optimization methods will be needed to find the minimum of the cost function. If the pulse shape $p(n)$ is only known for integer n, some interpolation method is needed to handle non-integer scalings.