Digital Communications - HW2

Jacopo Pegoraro, Edoardo Vanin

24/04/2018

Exercise 1

We want to estimate the impulse response of an IIR real system affected by white noise and with different sampling period between the input and the output, putting ourselves at the receiver so knowing only the output of the system, the input and a bound on the order of the system (approximating the IIR with an FIR). In particular we call the input of the system $x(kT_x)$ with $T_x = 1$, that undergoes upsampling of a factor 2 becoming $y(mT_y)$ with $T_y = \frac{T_x}{2}$ and is then filtered with an impulse response h defined on T_y that is what we want to estimate. To the output of the filter $z(mT_y)$ is added a white Gaussian noise $w(mT_y)$ with variance $\sigma_w^2 = -8$ dB. We call the overall output after all these operations $r(mT_y)$. The difference equation of the filter is known at the transmitter and is, neglecting the dependance on T_y for an easier notation:

$$z(m) = -a_1 z(m-1) - a_2 z(m-2) + y(m) \tag{1}$$

where $a_1 = -0.9635$ and $a_2 = 0.4642$. To do the estimation we need to use as input a PN sequence and approximate the IIR filter with an FIR of order $N_h \leq 20$. However, we only know methods that can work if the input and the output of the filter are defined on the same sampling period, while in this case h is defined on T_y. To solve the problem we exploit the polyphase representation of the system, dividing h in $h^{(0)}$ and $h^{(1)}$ that act respectively only on the even and odd samples of the input (we are using the *first noble identity* to transform the cascade of the upsampler and the filter in the separate filtering by $h^{(0)}$ and $h^{(1)}$ followed by a parallel to serial converter [1]). Once the equivalent system is derived, we can perform separately the estimates of $h^{(0)}$ and $h^{(1)}$ that are now defined on the same sampling time as their input (T_x). Two different methods will be used on both the polyphase components: the correlation method and the least squares method (LS) [1]. In the following sections we give a brief description of the structure of the PN sequences and of the two methods.

PN sequences

PN sequences are deterministic and periodic binary sequences that, despite their non-random nature, present characteristics that make them suitable to simulate a white noise while being much more simple to generate [1]. In this work we will use a type of PN sequences called Maximal Length sequences (ML) that can be obtained with recursive equations and the use of a shift register and the xor operator. ML sequences are defined giving their length L that has to be

an integer that precedes a power of 2 (here we will use sequences of length 7, 15, 31, 63, 127, 255). They can be generated starting with an arbitrary binary vector of length r (where r is the integer that satisfies $2^r - 1 = L$) and applying a difference equation that changes for every L (we do not report her all the equations used and we refer to [1]). The initial vector must not be the zero vector. To estimate the channel we will use a ML sequence repeated 2 times to be able to eliminate the transient on the output when we need to, without computing the exact number of samples affected by transient behavior. The correlation and the mean of a ML sequence (using -1 and 1 as binary values to eliminate biases) are:

$$mean = \frac{1}{L}$$

$$r_{ML} = \begin{cases} 1 & \text{for } (n)_{modL} = 0 \\ -\frac{1}{L} & \text{otherwise} \end{cases}$$

that get closer to the ones of a white noise as L increases. It's important to recall that to have good results in channel estimation using as input ML sequences we need to stick to the situation where L is much bigger than N, otherwise the fact that mean and variance are not the same as a white noise will become more and more influent and affect the results.

Correlation method

The correlation method is based on the schema represented in figure 1, but in our analysis we used $r(k)$ instead of $d(k)$ to be consistent with the notation.
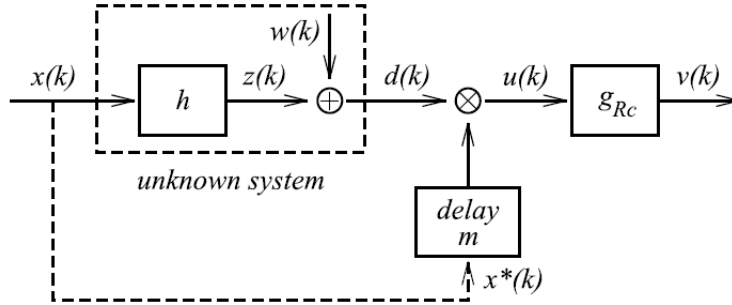

Figure 1: Schema of the correlation method.

The impulse response to be estimated is approximately equal to the correlation between the input sequence and the output of the unknown system (after the addition of the noise). The formula is:

$$\hat{h}_{corr} \simeq \hat{r}_{rx}(m) = \frac{1}{L} \sum_{k=L-1}^{2L-2} r(k)x^*(k-m) \tag{2}$$

where we use the samples of $r(m)$ starting from $L-1$ to be sure we skip the transient. To adapt this procedure to our specific case we computed the output

of the real system using its polyphase decomposition [1]. The result are the even/odd samples of the total output of the given system $r^{(0)}$ and $r^{(1)}$ that are the target outputs of the polyphase components, $h^{(0)}$ and $h^{(1)}$, to be used in 2. For each component then we carried out the computation of the correlation. The result is an estimate of the polyphase components from which we built the estimate $\hat{h}_{corr}$ of the complete impulse response by arranging the coefficients of the components as even and odd coefficients of $\hat{h}_{corr}$. As a measure of the error we make in estimating the coefficients we used the same functional as in the LS method (see the next section): $\mathcal{E}_{corr} = \sum_{k=L-1}^{2L-2} |r(k) - \hat{r}(k)|^2$, where $\hat{r}(k)$ is the output computed using the estimated impulse response. An estimate of the noise variance can be obtained with $\hat{\sigma}_{w,corr}^2 = \frac{1}{L}\mathcal{E}_{corr}$, where the result is equal to the true variance of the added noise only if the estimate coincides with the real impulse response.

LS method

The LS method aims to minimize the functional $\mathcal{E} = \sum_{k=L-1}^{2L-2} |r(k) - \hat{r}(k)|^2$ where the expressions of the target output and the estimated output are:

$$r(k) = \boldsymbol{h}^T \boldsymbol{x}(k) + w(k) \tag{3}$$

$$\hat{r}(k) = \hat{\boldsymbol{h}}^T \boldsymbol{x}(k) \tag{4}$$

and we introduce the matrix $\boldsymbol{\Phi}$ and the vector $\boldsymbol{\theta}$:

$$\Phi(i,n) = \sum_{k=L-1}^{2L-2} x^*(k-i)x(k-n) \tag{5}$$

$$\theta(n) = \sum_{k=L-1}^{2L-2} r(k)x^*(k-n) \tag{6}$$

that have, respectively, dimensions $(L-1) \times (L-1)$ and $(L-1) \times 1$. To simplify the computation of the two quantities we introduce the data matrix $\boldsymbol{\mathcal{I}}$, defined as:

$$\boldsymbol{\mathcal{I}} = \begin{bmatrix} x(L-1) & \cdots & x(0) \\ \vdots & \vdots & \vdots \\ x(2L-2) & \cdots & x(L-1) \end{bmatrix} \tag{7}$$

and the desired sample vector:

$$\boldsymbol{o}^T = [r(L-1) \quad r(L-2) \quad \ldots \quad r(2L-2)] \tag{8}$$

from which we can obtain $\boldsymbol{\Phi}$ and $\boldsymbol{\theta}$ as $\boldsymbol{\Phi} = \boldsymbol{\mathcal{I}}^H\boldsymbol{\mathcal{I}}$, $\boldsymbol{\theta} = \boldsymbol{\mathcal{I}}^H\boldsymbol{o}$. The solution of the LS estimate is given by $\hat{\boldsymbol{h}}_{ls} = \boldsymbol{\Phi}^{-1}\boldsymbol{\theta}$. As with the correlation method, here

---

[1] In both the correlation method and the LS method the way we compute the target outputs is the same. Note that even if the given filter is IIR, a polyphase decomposition can be obtained anyway by exploiting the fact that Matlab approximates to 0 the taps of the impulse response after a certain value. The resulting outputs we obtain could have been equivalently derived by first computing the global output by upsampling $x(k)$, filtering with h and in the end separating the even/odd sampler of $r(m)$. We chose the polyphase approach to avoid useless products by 0 in the filtering step.

too the estimate is done separately for the 2 polyphase components by setting as target the output of the real system components $r^{(0)}$ and $r^{(1)}$.

An estimate of the noise variance can be obtained with $\hat{\sigma}^2_{w,ls} = \frac{1}{L}\mathcal{E}_{ls}$ where $\mathcal{E}_{ls}$ is the mean squared error functional computed with the output obtained by filtering $x(k)$ with $\hat{h}_{ls}$.

Numerical Results

To conduct a good estimation of the impulse response h we need to carefully choose the 2 free parameters L, the length of the input PN sequence (that will be repeated 2 times), and N, the order of the filter that we know has to be assumed ≤ 20. To do this we need to consider the resulting estimate of the noise variance derived from computing $\hat{h}$ with many different values of N and L. The best choice will be the one that minimizes the variance (because that is related to minimizing the error, $\sigma^2_w \simeq \frac{\mathcal{E}_{min}}{L}$), without going below the performance bound of -8 dB, because in that case the estimate is affected by overfitting and is including the noise in the coefficients. In figure 2 we plotted the values of the variance of the noise for different N and L, in dotted line for the correlation method and in solid line for the LS. We also added indicators of the points to better understand where the estimate lies exactly with respect to the bound. We can see the for $L = 7$ and $L = 15$ the estimate does not show
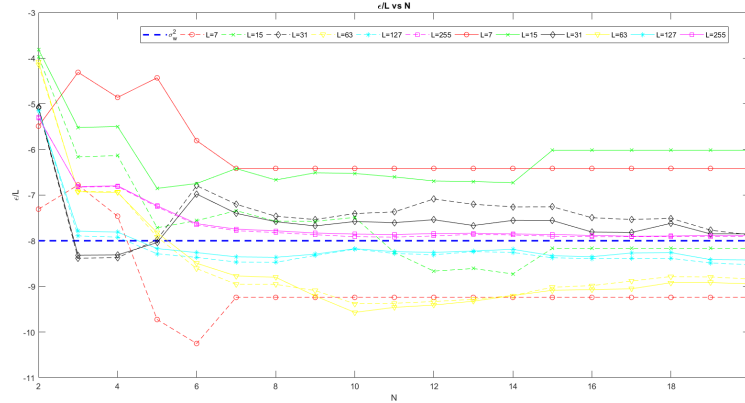

Figure 2: Plot of the estimated variance of the noise varying L and N.

a stable behavior, that is because trying to estimate the coefficients with a ML sequence leads to poor performances if $N \simeq L$. Indeed we should stick to the situation where L is a lot bigger than N. Good values of L and N can be $L = 31$ and $N = 5$ because in this point the estimated variance lies almost exactly on the bound. If instead we want to have a good estimate of more coefficients, $N = 15$, $L = 255$ could be another good choice. As a general rule if we want to estimate correctly more coefficients we should also increase L accordingly. To keep the order of the filter as low as possible and to not have to send a long sequence as input we decided for the first choice. In table 1 we confront the estimated coefficients and the variance of the noise with their real values.

4

nT_y	h	$\hat{h}_{corr}$	$\hat{h}_{ls}$
0	1.0000	1.0054	0.9991
1	0.9635	0.9247	0.9613
2	0.4641	0.5002	0.5062
3	-0.0001	0.0202	0.0255
4	-0.2155	-0.2549	-0.2253
	Real	Corr	LS
σ_w^2 [dB]	-8	-7.9776	-8.0404

Table 1: Estimated coefficients with the two methods for $N = 5$ and $L = 31$. Here are also reported the values of the estimated noise variance $\hat{\sigma}_w^2$.

We note that both methods perform well, and that the LS method is including a little noise into the estimate ($\frac{\mathcal{E}_{min}}{L} < \sigma_w^2$) but it's a very low value so we consider it not significant.
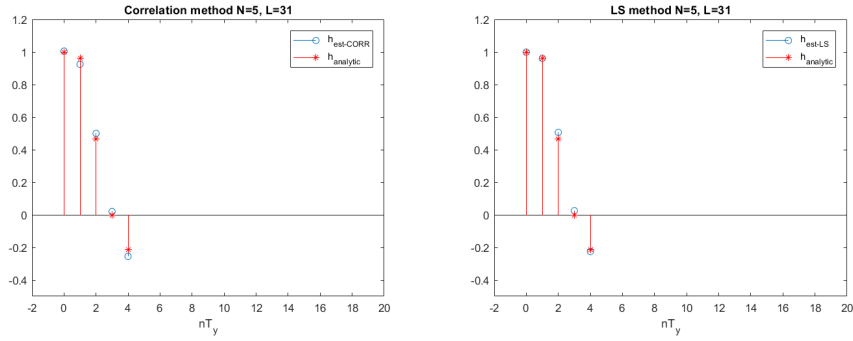In figure 3 we represent the analytical value of h and the estimates.

Figure 3: h and $\hat{h}$ with the correlation and LS method for the final choice of the parameters L=31 and N=5 .

Exercise 2

We want to simulate a flat fading channel with impulse response given by just one tap $h_0(nT_c)$, and classical Doppler spectrum determined by $f_d T_c = 40 \cdot 10^{-5}$. The Doppler spectrum model id defined as:

$$\mathcal{D}(f) = \begin{cases} \frac{1}{\pi f_d} \frac{1}{\sqrt{1-(f/f_d)^2}} & |f| \leq f_d \\ 0 & \text{otherwise} \end{cases}$$

If the impulse response has a single tap this means we have only one component in the power delay profile, that is the first component made of a LOS real constant C and a random part $\tilde{h}_1$ that we assume to be a complex Gaussian
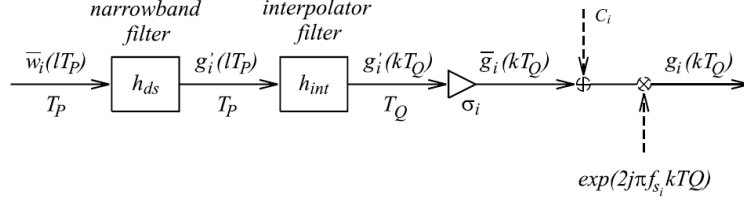
Figure 4: Schema for the simulation of the channel impulse response.

random process. We know that in this case the probability distribution function of $|\bar{h}_0| = |h_0/\sqrt{E[|h_0|^2]}|$ should be a Rice distribution, defined as [1]:

$$p_{|\bar{h}_0|}(a) = 2(1+K)ae^{-K-(1+K)a^2}I_0[2a\sqrt{K(1+K)}]\mathbb{1}(a) \qquad (9)$$

where I_0 is the modified Bessel function of the first type and order 0 and K a parameter called Rice factor that is given $K_{dB} = 2$ dB. The Rice factor is equal, in our case, to the ratio between the power of the direct component and the power of the reflected/scattered component: $K = \frac{C^2}{E[|\bar{h}_0|^2]}$. If we assume we will normalize the power delay profile to 1, from the knowledge of K we can derive $C = \sqrt{K/(K+1)} = 0.783$. Then the power of the non direct component must be set equal to $M_d = 1 - C^2$ for normalization.

Simulation

We start by assuming $T_c = 1$. Following the schema in figure 4, we generate a white noise with variance equal to 1 $\bar{w}_i(lT_p)$, defined of a sampling period T_p bigger than T_c such that $f_dT_p = 0.1$ so $T_p = 250$. This is necessary for the second step of the simulation, where we need to filter the white r.p. with an impulse response h_{ds} given by the coefficients reported in [2] to give it the shape of a classical Doppler spectrum (see figure 5). Indeed a convolution between h_{ds} that has frequency content in the interval $[-f_d, f_d]$ and a process defined on T_c would be practically non-feasible. It's important to recall that before filtering the coefficients must be normalized by the energy of the filter itself, to avoid unwanted modifications to the power of the r.p.. To go back to the original sampling time T_c we used an interpolator filter for which we choose to use the spline interpolation method by a factor equal to $\frac{T_p}{T_c} = 250$.

After the interpolation we scaled the r.p. by a factor $\sqrt{Md} = 0.622$, to give it the desired power level derived by the previous normalization. To complete the simulation we need to add the LOS component C to each sample of the resulting r.p.. What we obtained is a simulation of $h_0(nT_q)$ that however contains a transient due to the Doppler spectrum filter that has to be dropped in order to have a stationary random process, and to do this we have 2 choices. Being the Doppler filter an IIR filter, we can compute the transient on the output as $N_{transient} \simeq 5N_{eq}\frac{T_p}{T_q}$ where $N_{eq} = \lceil\frac{1}{\ln(|p_{max}|)}\rceil = 29$ is the equivalent time constant of the system (the number of samples needed to the strongest pole of the system to decade to $\frac{1}{e}$). We could also have made a more conservative choice by considering the truncation of the IIR impulse response made by Matlab and
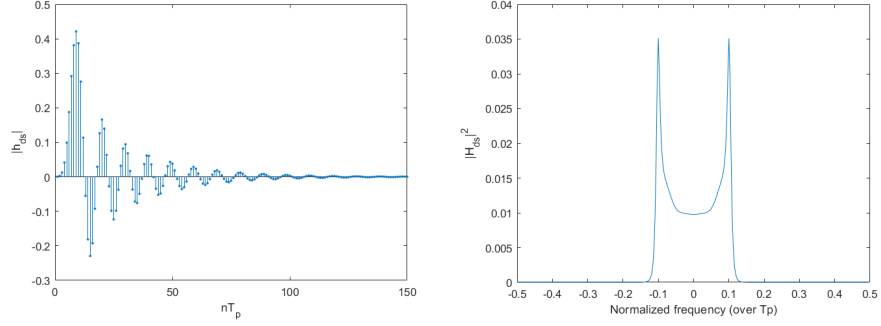
6

Figure 5: h_{ds} and $|H_{ds}|^2$.

counting as transient its length, as we would do with an FIR filter. The results in the two cases are basically the same so we go for the first method. In figure 6 we show a plot of the resulting $|h_0(nT_c)|$ for 7500 samples once dropped the transient.
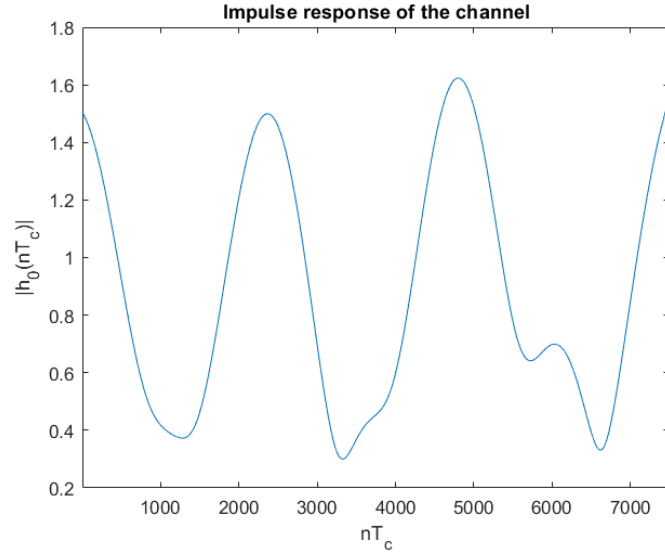

Figure 6: Impulse response of the channel, plot for 7500 samples.

Estimation of the pdf

For the estimation of the pdf we used a vector of 80000 samples, removing an initial transient as described above. The resulting distribution should be a Rice distribution with parameter $K = 1.5849$ in linear scale. In figure 7 on the left we can see the results of an estimation with a histogram with 17 bins over the 80000 samples, plotted along the theoretical curve of the distribution. The estimate

is enough accurate and follows quite well the asymptotic distribution. To give more quantitative results we also tried an estimate of the parameter K using a Matlab function for the estimation of Rice distributions that uses a simple moment-matching approach [3]. The result on the same 80000 samples as in the histogram proves that the realization is very close to the theoretical result because the vale of K obtained is $\hat{K} = 1.5838$ that is almost equal to the real one as we can see again from figure 7 on the right.
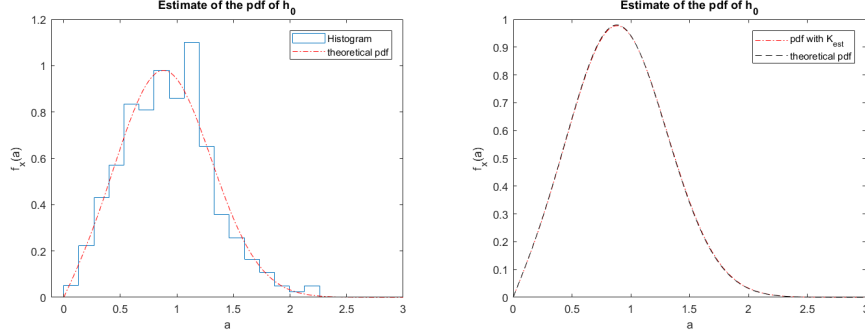

Figure 7: Estimate of the pdf of $|h_0|$ by an histogram (left) and with the use of an estimator for the parameter K.

Estimation of the PSD

For the spectrum estimation we used the Welch periodogram method. This method consists in dividing the input sequence into subsequences of consecutive D samples. Notice that two following subsequences, $h_0^{(s)}$ and $h_0^{(s+1)}$, overlap by S samples. For our case we used $D = \dfrac{\text{\# of samples}}{4} = 20000$ and $S = \dfrac{D}{2} = 10000$.

The number of subsequences is

$$N_s = \left\lfloor \frac{K - D}{D - S} + 1 \right\rfloor \tag{10}$$

The Welch periodogram is computed as

$$\mathcal{P}_{WE}(f) = \frac{1}{N_s} \sum_{s=0}^{N_s - 1} \mathcal{P}_{PER}^{(s)}(f) \tag{11}$$

where $\mathcal{P}_{PER}^{(s)}(f)$ is the periodogram computed for $h_0^{(s)}$ using a window of length D. We chose an Hamming window. The result of the estimate can be seen in figure 8, plotted along the theoretical curve obtained by considering a classical Doppler spectrum shape scaled by the power of h_0 with an added peak at frequency 0 due to the transform of the autocorrelation of the constant component C. Note that as we normalized the power of h_0 at the beginning, we don't need the normalization now to plot the results. The estimate seems to be pretty
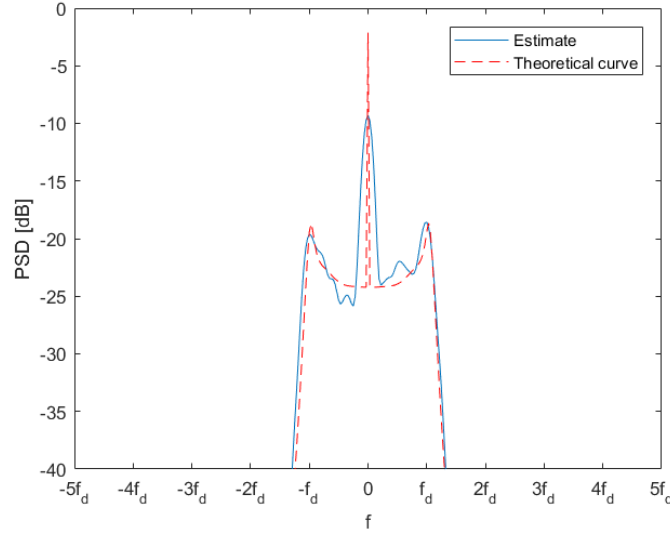
8

Figure 8: Estimate of the PSD of h_0 with the Welch periodogram (scaled by the number of samples), $D = 20000$ ad $S = 10000$ with 80000 samples.

accurate, aside from some variance in the central region of the curve and from the amplitude of the peak due to the LOS component C that is underestimated. Nevertheless the shape of the PSD can be correctly guessed as we clearly see the Doppler spectrum effect.

References

[1] Nevio Benvenuto, Giovanni Cherubini, *Algorithms for Communication Systems and their Applications*. Wiley, 2002.

[2] A. Anastasopoulos and K. M. Chugg, *An efficient method for simulation of frequency selective isotropic Rayleigh fading*, IEEE 47th Vehicular Technology Conference. Technology in Motion, Phoenix, AZ, pp. 2084-2088 vol.3., 1997.

[3] Ged Ridgway, *Rice/Rician distribution*, https://it.mathworks.com/matlabcentral/fileexchange/14237-rice-rician-distribution