Project Report
M6 SDR Receiver

Hochschule Rhein-Waal

Communication and Information Engineering B.Sc.

CI_5.02 Communication Systems

WS2020/2021

Prepared by

Amr Mostafa - 20733

Ahmed S.M. Ahmed - 25197

Supervised by

Prof. Dr. Volker Strumpen

Table of Content

1. Introduction 1
2. Project Structure 1
3. Impairments and Solutions 1
   3.1. Carrier's Frequency and Phase offsets 1
   3.2. Downsampler Timing and period Offset 1
   3.3. Multipath echoes and narrowband interferers 1
4. Implementation 2
   4.1. Determining the Assumed Carrier Frequency 2
   4.2. Carrier Recovery 2
   4.3. Downconversion 3
   4.4. Matching Filter 3
   4.5. Time Recovery 4
   4.6. Correlation 5
   4.7. Linear Equalization 6
   4.8. Decoding 6
5. Conclusion 7
6. References 7

1. Introduction

The purpose of this perm is to cover the work done on the implementation of the term project required for passing the module CI_5.02 Communication Systems for winter semester 2020/2021. The report covers the problems faced and the solution used to overcome these problems. Details of every design decision is included. The MATLAB source code for the project is attached with the report as a MATLAB project.

2. Project Structure

As advised in the beginning of the course, a study was formed of the two students writing the report. We decided to adapt Unite and Build strategy rather than Divide and Conquer Strategy. Hence, the contributions to the project and the code are equally distributed to both students. We both contributed in every step of the project and its implementation.

3. Impairments and Solutions

3.1. Carrier's Frequency and Phase offsets

Due to unsynchronized oscillators on the transmitter side and the receiver side there could be a frequency offset, phase offset or both, therefore, we cannot use the assumed carrier frequency directly to demodulate the received signal to baseband spectrum.

Solution: Carrier Recovery.

In order to demodulate correctly we need to use an adaptive algorithm to perform a carrier recovery.

Available options:

Dual Phase Locked Loop (dualpll)

Dual Costas Loop

Broadband channel noise

1. Matching Filter

2. Coding

3.2. Downsampler Timing and period Offset

If there is a timing offset in the Downsampler, the first pulse minimum or maximum is not detected properly, which causes all subsequent pulse sampling times to be shifted.

To counter this, we need to implement a time recovery adaptive algorithm to detect the right timing and period for downsampling.

Available options:

1. Decision-Directed Adaptation

2. Adaptive Power Maximization

3.3. Multipath echoes and narrowband interferers

Due to surrounding obstacles, the signal could be received with multiple echoes in the receiver, also a nearby user of the spectrum could cause a narrowband interference

To counter both issues we need to implement an equalizer adaptive algorithm.

Available Options:

1.Trained Adaptive Least-Mean-Square Equalization

2.Blind Adaptive Decision-Directed Equalization

3.Blind Adaptive Dispersion Minimizing Equalization

4. Implementation

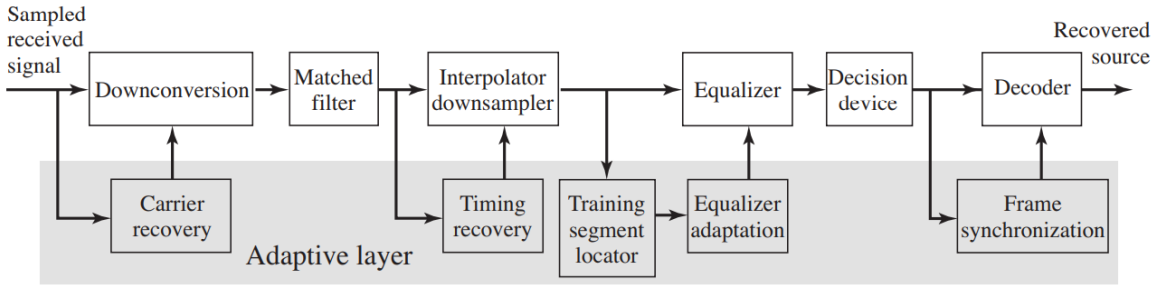

Figure 1: DSP Portion of SDR,

C. Richard Johnson, Jr, Cornell University, New York William A. Sethares, University of Wisconsin, Madison Andrew G. Klein,(2011). Software Receiver Design. Cambridge University Press. p. 345. ISBN 9781107007529

4.1. Determining the Assumed Carrier Frequency

Since Sub-Nyquist sampling is used by the ASP portion of the receiver, which replicates the passband at centre frequencies. Therefore, we need to calculate the current assumed frequency using the following formula:

$$f_c = min_k |f_{if} + kf_s|$$

4.2. Carrier Recovery

Initially, dual phase locked loop was considered for carrier recovery as it is efficient and widely used in communication systems. Due to required signal preprocessing before applying the dual loop as it requires the signal to be squared and applied to a narrow passband filter which extracts approximation of the carrier, However, the BPF causes phase shift (delay in time domain). This phase could be compensated by using filtfilt function in MATLAB instead of filter function.

MathWorks, *https://www.mathworks.com/help/signal/ug/compensate-for-delay-and-distortion-introduced-by-filters.html*

However, the bandwidth of the second and third signals is close to the Nyquist frequency, hence, the signals need more processing to handle the squaring process successfully.

Since a dual Costas Loop can achieve the same results without the need of preprocessing, it was the best choice as the carrier recovery algorithm.

Dual Costas Loop

The Costas loop performs frequency and phase tracking directly on the received signal, using a squarer on the feedback path. It reverses the order used in PLL to recover the carrier: first demodulation, LPF, then squaring the output.

The first loop tracks the frequency offset converging to a straight line as:

$$\theta_1[n] = 2\pi(fc - f0)nTs + \emptyset$$

And its algorithm update function is

$$\theta_1[k+1] = \theta_1[k] - \mu LPF\{r(kTs)\cos(2\pi f0kTs + \theta_1[k])\} \times LPF\{r(kTs)\sin(2\pi f0kTs + \theta_1[k])\}.$$

The second loop tracks the phase offset with respect to the first loop, hence, generating a synchronized sinusoid that can be used for downconversion.

$$\approx \frac{1}{4}s_{avg}^2\cos^2(\emptyset - \theta_1)$$

$$\theta_2[k+1] = \theta_2[k] - \mu LPF\{r(kTs)\cos(2\pi f0kTs + \theta_1[k] + \theta_2[k])\} \times LPF\{r(kTs)\sin(2\pi f0kTs + \theta_1[k] + \theta_2[k])\}$$

4.3. Downconversion

After the carrier recovery process, downconversion to baseband spectrum is now possible by demodulating the received signal with the recovered carrier.

x = r.*carest;

Figure 4: Demodulation (Mixing)

A LPF FIR filter is used after to remove the double frequency components and eliminate possible FDM frequencies or noise.

However, the correlating matching filter behaving as a LPF and having the capability of removing the double frequency components, it is not designed for this purpose, so a LPF is used.

4.4. Matching Filter

The matching filter uses SRRC pulse as it is used for the pulse shaping filter in the M6 transmitter, because its self-convolution is a Nyquist pulse and it has a smaller absolute bandwidth than other qualifying pulse shapes.

Correlating the SRRC pulse with the baseband signal enhances the peaks of the pulses.

Since the SRRC is a symmetric signal we can convolute the SRRC pulse with the demodulated signal which acts the same as correlation.

4.5. Time Recovery

As the upsampling factor 'M' is non integer number and due to the possibility of downsampling timing or period offset, a time recovery adaptive algorithm is needed to correctly downsample the received signal.

After inspecting the three received signals parameters, we found out that the SRRC Rolloff factor for the three signals respectively are 0.4, 0.9, and 0.5

While Direct decision time recovery algorithm would work on the first signal, it will not work on the second and the third signals. As the SRRC expresses local minima with Rolloff $\beta > 0.4$ with the DD algorithm as shown in figure 2.

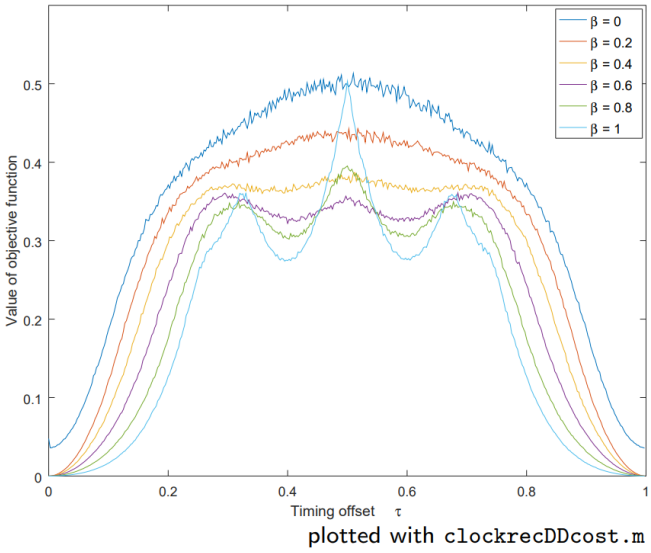

Figure 2

Which prevents the Decision directed algorithm steepest decent objective function from converging to the desired global minima.

The Output power maximizing adaptive algorithm is the obvious choice in our situation as it works fine with SRRC of any Rolloff factor $\beta > 0$ as shown in figure 3, and since the SRRC with Rolloff 0 is a sinc, which is not a good choice for communication systems anyway, this is not a problem.

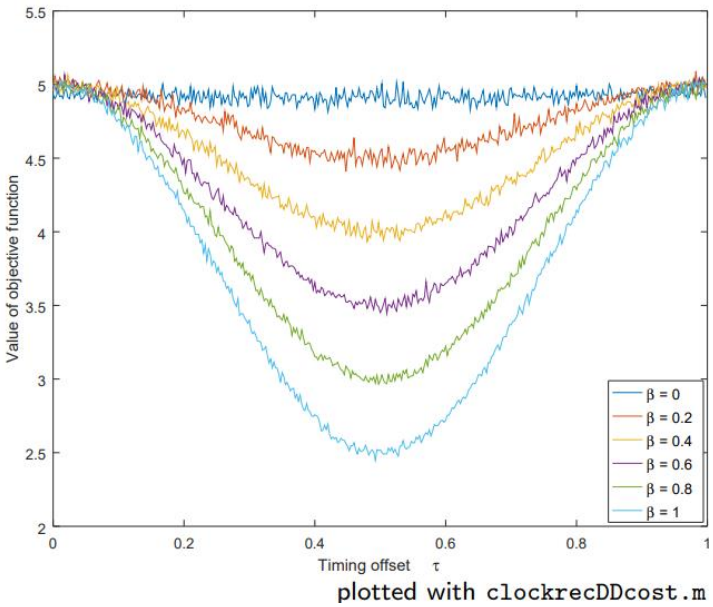

Figure 3

Also, the Output power maximizing algorithm converges twice as fast as the decision directed algorithm, by all means, the output power maximizing is the winner.

The algorithm follows steepest ascent to maximize the objective function:

$$\tau\,[k\,+\,1]\;=\;\tau\,[k]\;+\;\mu'\;\frac{dJ_{op}(\tau)}{d\tau}\bigg|_{\tau=\tau[k]}$$

The update function is:

$$\tau\,[k\,+\,1]=\;\tau\,[k]+\;\mu x\,[k]\,(x\left(k\,\frac{T}{M}\,+\,\tau\,[k]+\,\delta\right)-\,x\left(k\,\frac{T}{M}\,+\,\tau\,[k]-\delta\right))$$

4.6. Correlation

We wish to detect the beginning of the received message which is delayed by filters across the implementation of the receiver. Since the message is constructed of headers of a known preamble ‘A0Oh well whatever Nevermind’ followed by frames of 100 characters each.

We can use cross correlation to correlate the known preamble with the received message to detect the exact location of the header.

Using MATLAB corrx.m function achieves this purpose. Applying cross correlation between the preamble and the received signal and extract the maximum absolute value of the correlation would give the location of the preamble in the signal, but not necessarily the first one.

By analysing the three received signals and the characteristics of converting letters to 4PAM,

A formula to locate the first header is implemented.

By counting the modulus of the detected preamble location over the total frame length including the preamble yields the first preamble location.

A snippet of the MATLAB code used in implementing the correlator is shown in figure 4

4.7. Linear Equalization

Due to Multipath echoes caused by the channel and possible narrowband interferers there is a possibility of intersymbolic interference. To revert this, we need to implement an adaptive equalizer to determine the impulse response needed to revert the channel effect on the signal, hence removing the echoes.

Since the signal includes a preamble that is repeated periodically in the beginning of each message frame, an adaptive Least-Mean-Square equalizer (LMS) can be used to determine the appropriate frequency response needed to equalize the received message.

However, trying to integrate the mentioned adaptive equalizer into the system, our trial was not successful.

A decision of removing the equalizer for the benefit of the first two signals was made as will be discussed in the next section, as the receiver turned unfunctional when integrating the faulty algorithm design.

4.8. Decoding

After inspecting the Transmitter's specifications mentioned in chapter 15 of the SDR reference book. It is mentioned that the message is encoded by converting each letter to a seven bits sequence of ASCII table, then converted to (5,2) block coding then to 4-PAM. With n letters, 7n binary uncoded bis are generated, converted to $7n\left(\frac{5}{2}\right)$ coded bits and finally $7n\left(\frac{5}{2}\right)\left(\frac{1}{2}\right)$ 4-PAM signal is generated.

The decoding method is implemented as advised in the textbook; however, it did not work on the received messages. An isolated source code file nocode52.m provides the implemented decoding in the MATLAB project handed along with the report.

So after inspecting BigTransmitter source code, no such encoding was found, rather, letters2PAM function is used to generate the message with respect to the distribution of the data over frames of 100 characters each and preambles before each of them.

So PAM2letters.m function is used in our implementation to decode the hard decisions into a text message.

Then strrep.m function is used to strip out the preambles from the message.

5. Conclusion

The implemented SDR receiver is able to recover the first two mystery signals successfully with minimal errors without the need for an equalizer. The third signal however is highly distorted, the receiver managed to decode approximately half the message with minimal errors, the second half of the message is a little bit readable, however it needs an equalizer to fully recover the message correctly.

The implementation is working fine for low to medium distortions. To make the receiver even more efficient and capable of decoding high distorted signals, an adaptive equalizer should be added.

6. References

1. Prof. Dr. Volker Strumpen, Communication Systems course of Hochschule Rhein-Waal, 2020/2021
2. C. Richard Johnson, Jr, Cornell University, New York William A. Sethares, University of Wisconsin, Madison Andrew G. Klein, (2011). Software Receiver Design. Cambridge University Press. p. 345. ISBN 9781107007529
3. All the MATLAB codes used to implement the SRD receiver for the project are based on the codes provided with the textbook [2], an individual citation for each code used is included in the source code for our receiver.