Software Radio Fall 2023 Final Project Submission
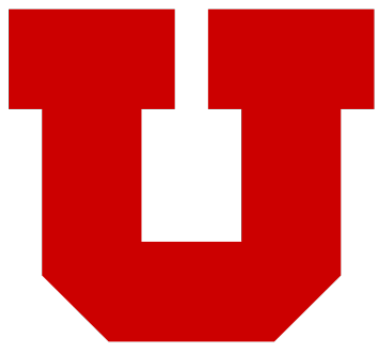
A Project Report

Submitted to the Professor Farhang

∘

by

Grant Brown

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

COLLEGE OF ENGINEERING UNIVERSITY OF UTAH

December 2023

0.1 Abstract

This work aims to develop the receiver portion of a software radio. It must take into account real-world issues encountered in software radio's, such as: carrier offset and timing phase offset due to crystal oscillator imperfections, as well as distortion introduced by various transmission mediums. In part I of the project the students must demodulate a undisturbed transmitted signal to baseband, acquire the optimal timing phase associated with the transmitted symbols, and extract the specified data. Part II and III of the project uses a channel that introduces distortion and the students must create a symbol-spaced equalizer and fractionally-spaced equalized to restore the Nyquist pulse. Part IV of the project introduces a carrier offset and phase drift into the transmitted data, requiring a carrier acquisition and tracking method to restore the symbols. Part V introduces a non-zero timing phase into the design and requires a symbol rate tracker to be implemented. Part VI is optional and utilizes an equalizer to perform tracking for carrier and timing phase offsets.

0.2 Section 1 - Introduction

As demonstrated in Figure 1, transmission of signals in digital systems requires the use of: a transmitter, a channel, a receiver, and a equalizer. The job of the transmitter is to generate data symbols which are free of ISI that can be transmitted over the desired channel medium. This is accomplished by first encoding bits to data symbols, which are then passed to a Nyquist pulse shaper, and then modulated to the desired carrier frequency. The receiver must then demodulate the desired signal to the baseband, perform matched filtering to remove unwanted spectrum's, and then sample at the correct timing phase to obtain the transmitted symbols. Physical devices, such as crystal oscillators are used in this process which bring physical limits to the precision of demodulation and sampling precision. These limits bring arise two major issues. One issue is carrier offset, which occurs because the transmitted signal is frequency shifted by a carrier $cos(2\pi f_c + \Theta(t))$. To perform exact demodulation we must know the timing varying phase as well as the exact frequency at which the signal was modulated. Any inaccuracies can cause a slight frequency offset where out signal is not centered around 0 Hz and can result in a constellation rotation shown in Figure 2. The other issue is timing recovery, which occurs because exactly recovering the signal after demodulation and filtering requires knowing the exact timings at which the Nyquist pulse was generated. These timings are referred to as the optimal timing phase. Any delta between the transmission timing phase and receiving timing phase can introduce ISI and disturb the Nyquist pulses. In addition, desired channel mediums typically will introduce some level of distortion to the received signals spectrum through the addition of unwanted noise and auxiliary signals. It is imperative to remove this noise to restore the Nyquist pulse properties to ensure no ISI and correct symbol detection. Equalizers are designed to perform this deconvolution operation which restores the initial spectrum by correctly weighting the irregularities through trainable weights, similar to a Neural Network.

The project explores these issues and potential solutions and implementations of these solutions through a 6-sectioned project. Most of the solutions introduced used in this project are pilot-aided or decision-directed methods. The transmitted signals consist of a preamble and payload as shown in Figure 7. The preamble consists of training symbols, which are called pilots, which are used to fine tune the receiver parameters to ensure the correct signals are received at the output of the subsystem. Part I focuses on the basics of recovering the timing phase from an ideal channel. Part II and III introduces channel irregulaitires into the design and requires symbol-spaced and fractionally-spaced equalizers to be implemented to restore the Nyquist pulses which removes ISI. Part IV explores carrier and phase offset and the rotation it causes in constellations. Part V requires students to utilize decision-directed methods to restore a drifting optimal timing phase. Part VI is optional and uses fractionally-spaced equalizers to perform all functions of the receiver.

0.3 Section 2 - Demodulation, Timing Acquisition, and Data Extraction

Part I loads the transmission data from an ideal channel. We may assume there is zero phase error and zero frequency offset. The transmitter performs modulation with a carrier wave of the form $cos(2\pi f_c t)$ where $f_c = 10^5 Hz$. This is determined by looking at the spectral content of the signal *xRF1.mat* shown in Figure 8, which mimics the shape of a square-root raised cosine frequency response around $f_c = 10^5 Hz$ as expected. To increase the received signal to have highest SNR ratio we utilize matched filtering. This match filter also operates as a low-pass filter which can remove all unwanted spectral content outside the baseband. The transmitter utilizes expansion to reduce the bandwidth of the transmitted signal, therefore we needed to decimate the signal at the

correct timing phase to ensure the signal samples are correctly identified at the receiver. According to chapter 10, for pulses with less then 100% excess bandwidth 3 spectral coefficients exist in the Fourier series of the ensemble power $E[y(t)^2]$ of the received signal. This leads to the closed-form solution for the ensemble average $E[y(\tau)^2] = \rho_0 + 2|\rho_1|cos(\frac{2\pi}{T_b}\tau + \angle\rho_1)$ where ρ_0 is the dc component and ρ_1 is the first harmonic. The ensemble average is a dc-shifted sinusoid, Figure 3, which was verified through the Matlab code. Since there is not channel distortion the optimal timing phase is the location of the peak of the sinusoidal graph. After some time to reach steady state, the peaks occur every 100 samples as expected due to expansion by 100. We find the first sample that correlates to the peaks which is sample 1 and then decimate by every 100 samples from there. To verify the preamble and payload are correctly deciphered we may look at the constellation shown in Figure 4. As shown we see that the data symbols are at the correct symbol mapping spots with the preamble being scattered around these four points.

It is stated that the preamble begins consists of four pilot sequences each of length 32. Therefore, if we cross-correlate the decimated baseband signal with the pilot sequence we obtain four peaks, as shown in Figure 5. These peaks indicated strong correlation between the baseband signal and the pilot, which means that the two sequences are the same. Therefore if we know the pilot is N=32 symbols then 32 symbols after the last peak the payload begins. Since there is no channel distortion we should expect a perfect QPSK constellation as shown in Figure 6. Passing this data to the text file we are able to correctly decipher xRF1.mat which tells a joke.

0.4 Section 3 - Preamble Detection and Symbol-Spaced Equalizer

In Part II we explore how channel distortion can cause problems with ISI and how the use of equalizers can be used to solve non-ideal channel variance. To explore the mathematical model of the channels impulse response we note that the channel can be modeled as $c(t) = \alpha_i \delta(t - \tau_i)$ assuming a single-path channel. When using Nyquist pulses the equivalent baseband impulse response between the transmitter and receiver is $c_{BB}(t) = \beta_i p(t - \tau_i)$, where $p(t)$ is the convolution between the matched receiver and transmitter filters, and $\beta_i = \alpha_i e^{-j2\pi f_c \tau_i}$ is a complex gain factor. If we assume that the delay incurred by the channel is insignificant the received signal at the demodulator gives $y(t) = \sum_{n=-\infty}^{\infty} s[n]c_{BB}(t - nT_b) + v(t)$ where $v(t)$ is noise added by a non-ideal channel. If we include the complex gain factor β_i the complex baseband equivalent impulse response of the channel simplifies to $y(t) = \beta_i \sum_{n=-\infty}^{\infty} s[n]c_{BB}(t - nT_b) + v(t)$. In the absence of channel noise we have perfect Nyquist pulses spaced by T_b at the output and sampling at these portions we can recover the transmitted symbols. When there is a significant delay in the channel or the channel has additive noise our assumptions fail and we must revert to the more general impulse response of $y(t) = \sum_{n=-\infty}^{\infty} s[n]c_{BB}(t - nT_b) + v(t)$. Here $c_{BB}(t)$ is not a perfect Nyquist pulse and ISI will occur. To restore the Nyquist pulse after the receiver we pass $y(t)$ through an equalizer, $w(t)$ whose cascade formation with the baseband channel impulse response, $c_{BB}(t) \star w(t)$, restores the Nyquist pulse shape to an acceptable level of ISI. The addition of multi-path channels follows the above discussion with an emphasis on the delay of the paths must be equal to ensure no ISI, otherwise an equalizer is needed.

In Part II we implement a symbol-spaced equalizer, as shown in Figure 9. By transmitting a sequence of pilot symbols we can utilize the cyclic equalization method shown in Figure 10. The goal is to minimize the mean squared error between the equalized symbols and the pilot, $|e[i]^2|$. To correctly utilize the cyclic equalizer we need to grab the preamble from the received signal. This can be done by performing an auto-correlation of the received and decimated signal. Theory says that when the transmitted signal is auto-correlated through the equation $r_{yy}(n, N+1) = \sum_{k=0}^{N} y[n-k]y^\star[n-N-1-k]$ we will obtain a flat top or constant auto-correlation, as shown in Figure 12, when transversing over the repeated pilot sequence. Grabbing any 32 symbol sequence from this flat section results in single period of the pilot sequence which is then used in the cyclic equalizer. The code to implement the cyclic equalizer is shown in Figure 11. The output of the weights may be shifted, as shown in Figure 13, due to an offset of where you grabbed the preamble and the at which symbol you start the pilot sequences. Since equalizer tap weights peak in the middle of the sequence and the channel impulse response also peaks in the middle we circularly shift the tap weights to be center aligned as shown in Figure 14. The effect of the equalizer can be shown through the a comparison of the constellation of the payload before and after equalization in Figure 15.

Correct timing phase is critical for performance because the symbol spaced equalizer does not correct timing phase errors. Figure 16 shows that the constellation of the decimated signal after equalization becomes less clear when compared to the optimal timing phase decimated sequence due to increased ISI. We will defer more detailed timing phase discussion until Part IV.

0.5 Section 4 - Fractionally Spaced Equalizer

Part III of the project requires us to implement a half-symbol spaced equalizer shown in Figure 17. Fractionally-spaced equalizers are more complex than symbol-eportspaced since they operate at a higher data rate then the symbol-spaced equalizers, however hold benefits that warrant the complexity increase. One key benefit that fractionally-spaced equalizers bring is that they can deal with timing phase imperfections without the need for an explicit timing recovery method. The core theory behind this benefit is that timing phase imperfections are brought about by incorrect phase in spectral aliasing. When the data rate is high enough to avoid the spectral aliasing the timing phase will not depend on the phase of the overlap anymore as long as the fractionally-spaced equalizer is used correctly.

To correctly train the tap-weights we once again utilize the cyclic method shown in Figure 18. It is important to note that because of the expansion by L we have to grab L*32 samples of the preamble to get all symbols in the pilot. In the case of the half-symbol spaced equalizer L = 2 and M=1 where M is the decimation factor.To ensure the correct tap weights are associated with pilot symbols we must shift the preamble sequence by 2. Since the pilot is still 32 symbols we must perform this cyclic equalization over a two clock cycle span for each iteration. The general conclusions of the symbol-spaced equalizers are applicable to the fractionally spaced equalizer as well.

The eye pattern of the half-symbol spaced equalizer is shown for xRF3.mat in Figure 19. As shown, the constellation becomes closer to ideal. Mathematically we can enumerate the performance of the equalizers by looking at the mean-square error of the two signals. Using xRF3.mat as our reference signal for comparison we see that the MSE of the symbol-spaced equalizer is $1.764204e - 02$ while the half symbol-spaced equalizer is $8.723901e - 03$, showing that on average the half symbol-spaced equalizer will perform a better equalization scheme, however it is marginal with optimal timing phase. When timing phase deviations are added the fractionally spaced equalizer starts to shine. Using xRF5.mat with a timing phase offset of 50 samples we see that the MSE of the symbol-spaced equalizer is $5.863084e - 02$ while the half symbol-spaced equalizer is $1.635847e - 02$, which shows that the half symbol-spaced equalizer has 5x better performance then the symbol-spaced equalizer when timing phase is not optimal.

0.6 Section 5 - Carrier Acquisition and Tracking

Part IV explores having carrier and phase offset in the receiver and channel and how that affects the constellations. The carrier $cos(2\pi f_c t + \Theta(t))$ is modulated by the data symbols at the transmitter. $\Theta(t)$ is a time varying phase that must be accounted for to avoid constellation rotation shown in Figure 20. In addition to a time-varying phase the receiver must perform demodulation to move the transmitted data symbols back to baseband. If the carrier frequency and the demodulation sinusoid frequency are not exactly equal we will obtain resulting in the signal $y(t) = e^{j2\pi\Delta f_c t} \sum_{n=-\infty}^{\infty} s[n]h_0(t - nT_b) + v(t)$ where $h_0(t)$ is the impulse response of the channel to the first data symbol. To obtain the ideal baseband signal of $y(t) = \sum_{n=-\infty}^{\infty} s[n]h_0(t - nT_b) + v(t)$ we must determine the carrier offset Δf_c. Utilizing the pilot we can utilize the auto-correlation of $J = \sum_{n=N_1}^{N_2} y[n + N]y^{\star}[n]$. Expanding this signal we know $x[n] = x[n + N]$ which will give us $J = e^{j2\pi\Delta f_c t} \sum_{n=N_1}^{N_2} |x[n]^2|$, which holds due to the uncorrelated nature of the noise $v(t)$ and the transmitted symbols $x[n]$. Solving for Δf_c we get: $\Delta f_c = \dfrac{1}{2\pi N T_b} \angle J$. Note that the length of the periodic signal N determines the precision to what we can determine in carrier offset.

After the equalizer we may still be dealing with some residual phase offset of the signals. To alleviate this problem before detecting the final data we pass it through a data-directed phase tracking method shown in Figure 21. The theory behind this method is a predictive phase detector, which compares the slicer input and output. If the phase error is small enough then we can approximate the error as $\epsilon[n] = \dfrac{I\{\tilde{s}[n]\hat{s}[n]\}}{R\{\tilde{s}[n]\hat{s}[n]\}}$ where $\tilde{s}[n]$ and $\hat{s}[n]$ are the slicer inputs and outputs. This error is then passed to a tune-able loop filter which updates the phase for the next symbol.

Utilizing this procedure on xRF7.mat we can see the effect of the carrier phase offset in the constellation in Figure 22. Applying the decision directed method to equalize the phase we get the constellation in Figure 23, showing the need for the carrier phase offset removal.

0.7 Section 6 - Symbol Rate Tracking

In Part V of the project we must account for a time-varying timing phase which was inserted in the transmitted signal. In this case the optimal timing phase may not be every L samples from the initial transmitted symbol. This can arise due to non-ideal crystal oscillators and we therefore must determine at which decimation points we should grab the transmitted signal to extract the payload. In this part we must do an initial coarse timing recovery based on the theory discussed on Section II on the ensemble average power. Once that is done we find the maximum power peak and treat that as the beginning of decimation sequence on our transmitted packed. From this transmitted packet we use the previously discussed cross-correlation peak analysis, shown in Figure 5. This allows us to detect the beginning of our payload. Once the payload has been detected we utilize a decision-directed timing recovery algorithm. The decision-directed method uses basic differential calculus to minimize the expectation of the error signal, $\xi = E[|e[n]|]^2$ where $e[n] = s[n] - y(nT_b + \tau)$. Taking the derivative of the error to find a local minimum solution depending on our initialization point gives (unless function is convex) gives $\frac{d\epsilon}{d\tau} = -2R\{(e^\star[n]\frac{dy(nT_b + \tau)}{d\tau})\}$. This gradient descent method then utilizes an approximation of this derivative to slowly move towards the optimal phase in each iteration.

Utilizing this algorithm on xRF9.mat shows the need to timing phase restoration. Without timing phase restoration, the QPSK constellation is of the form in Figure 24. When timing phase and slicing is applied to latch to the closest symbol in the desired constellation our QPSK constellation is of the form in Figure 25 with the correct text output.

0.8 Section 7 - Use of equalizer for tracking of carrier offset and timing phase

In a receiver three functions operate concurrently: timing recovery, carrier recovery, and channel equalization. If the channel equalizer tap-weight adaptation algorithm is fast enough one may automatically compensate for an residual carrier offset Δf_c. However, if the carrier offset it too large then the channel equalizer cannot resolve this issue, therefore a coarse carrier recovery is needed at the beginning. As mentioned earlier, a fractionally-spaced equalizer can perform timing recovery, therefore the timing and carrier recovery blocks can be removed. This leads to a receiver system shown in Figure 21 without the PLL, since the equalizer can solve for the phase rotation. The current implementation of the LMS algorithm has a convergence rate that is too low to track carrier offset issues with the constellation. The use of a faster adapting algorithm is needed for carrier offset tracking, such as the recursive least squares algorithm (RLS). The RLS algorithm projects the under-determined system to the solution with the smallest error vector within the systems vector space. The convergence characteristics shown in Figure 26 and Figure 27 show that the RLS convergence is an order of magnitude better then the LMS algorithm.

0.9 Conclusion

In this 6-part final project we have demonstrated understanding of the basics of software radio systems. We have explored the architectures for transmitter-channel-receiver systems, as well as common issues that may occur in these systems. We have equalized channel distortion using symbol and fractionally spaced equalizers. Results show that while symbol-spaced equalizers bare minimum complexity the advantages of the fractionally-spaced equalizer out-weight its added complexity. To solve non-ideal transmitter and receiver structures, we have utilized pilot and decision-directed methods to fix timing phase, constellation rotation, and carrier recovery. This has allowed for us to overcome common issues such as crystal oscillator imperfections to extract transmitted data at the output of the equalizer.

Professor Fahrang, thank you for the great semester and patience as we learn the material. I greatly enjoyed the class and I am looking forward to learning more on this topic over time with your guidance.
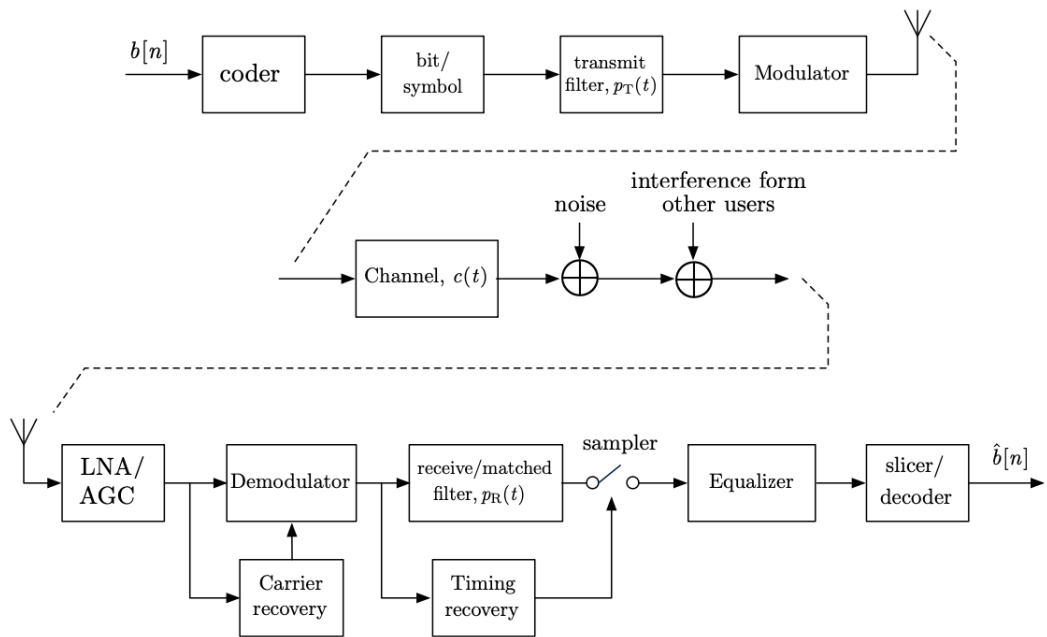
0.10 Figures

Figure 6.1: Block diagram of a digital communication system.

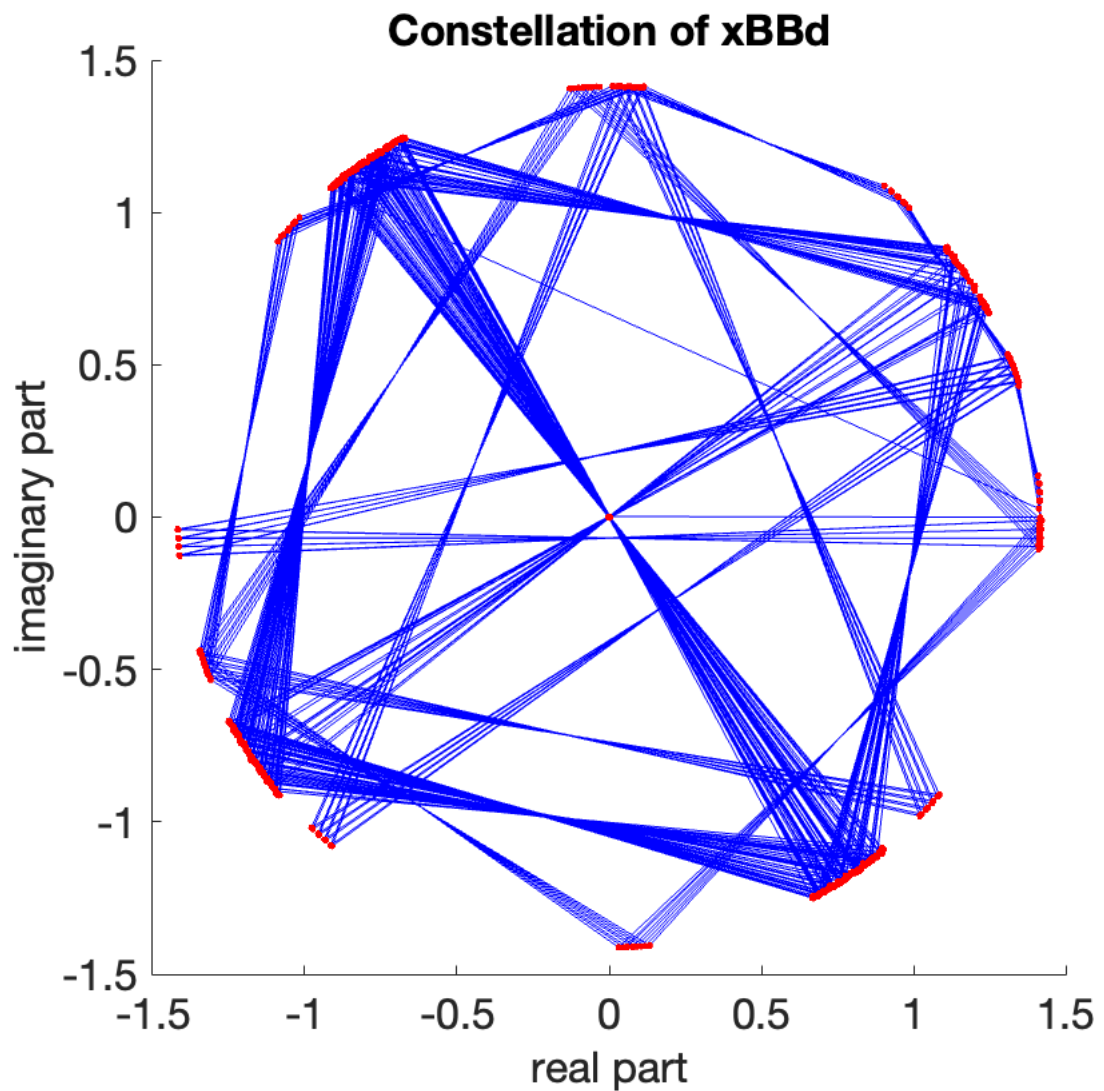Figure 1: Block Diagram of A Communications System

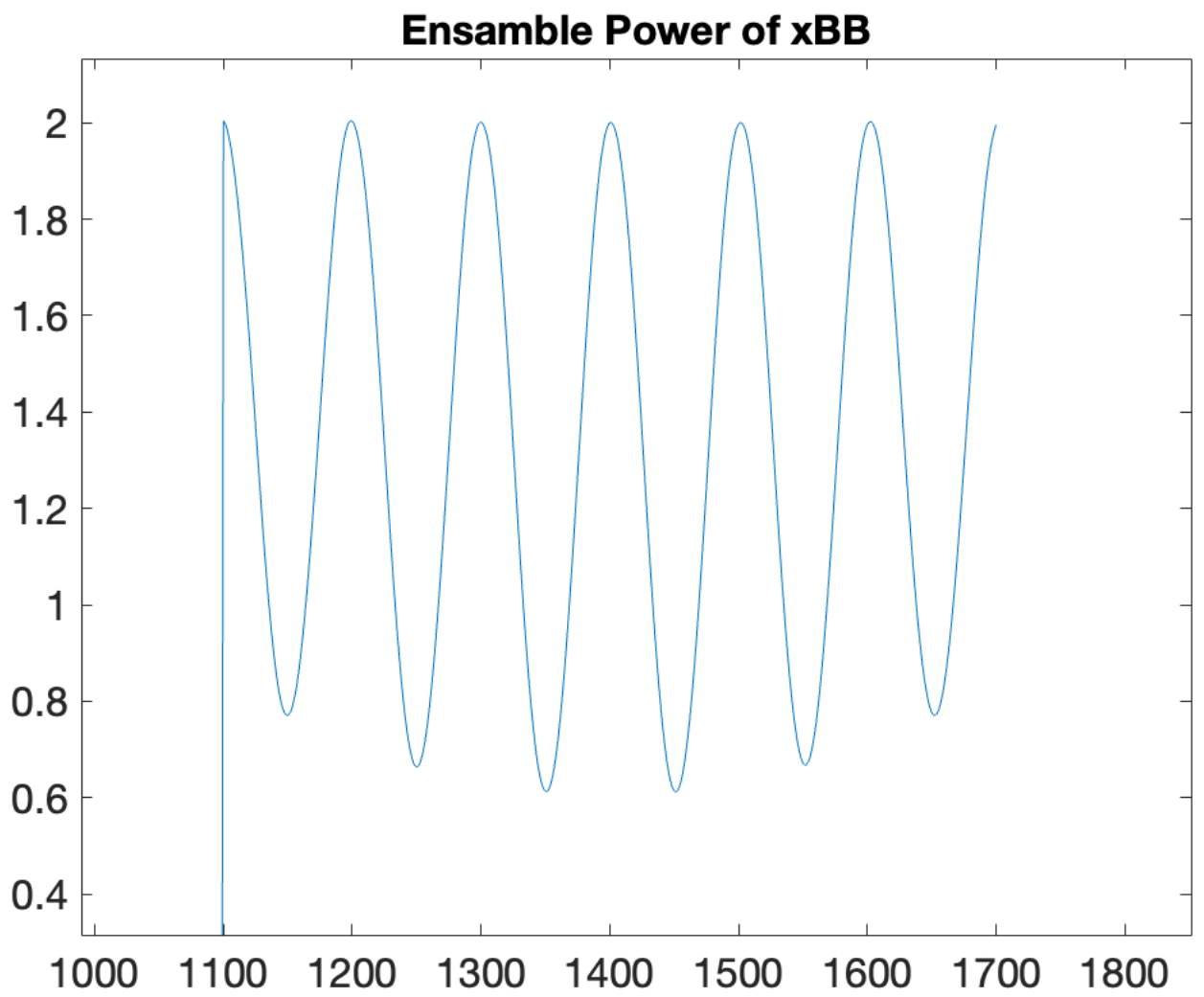Figure 2: Result of Frequency Offset in QPSK Constellation
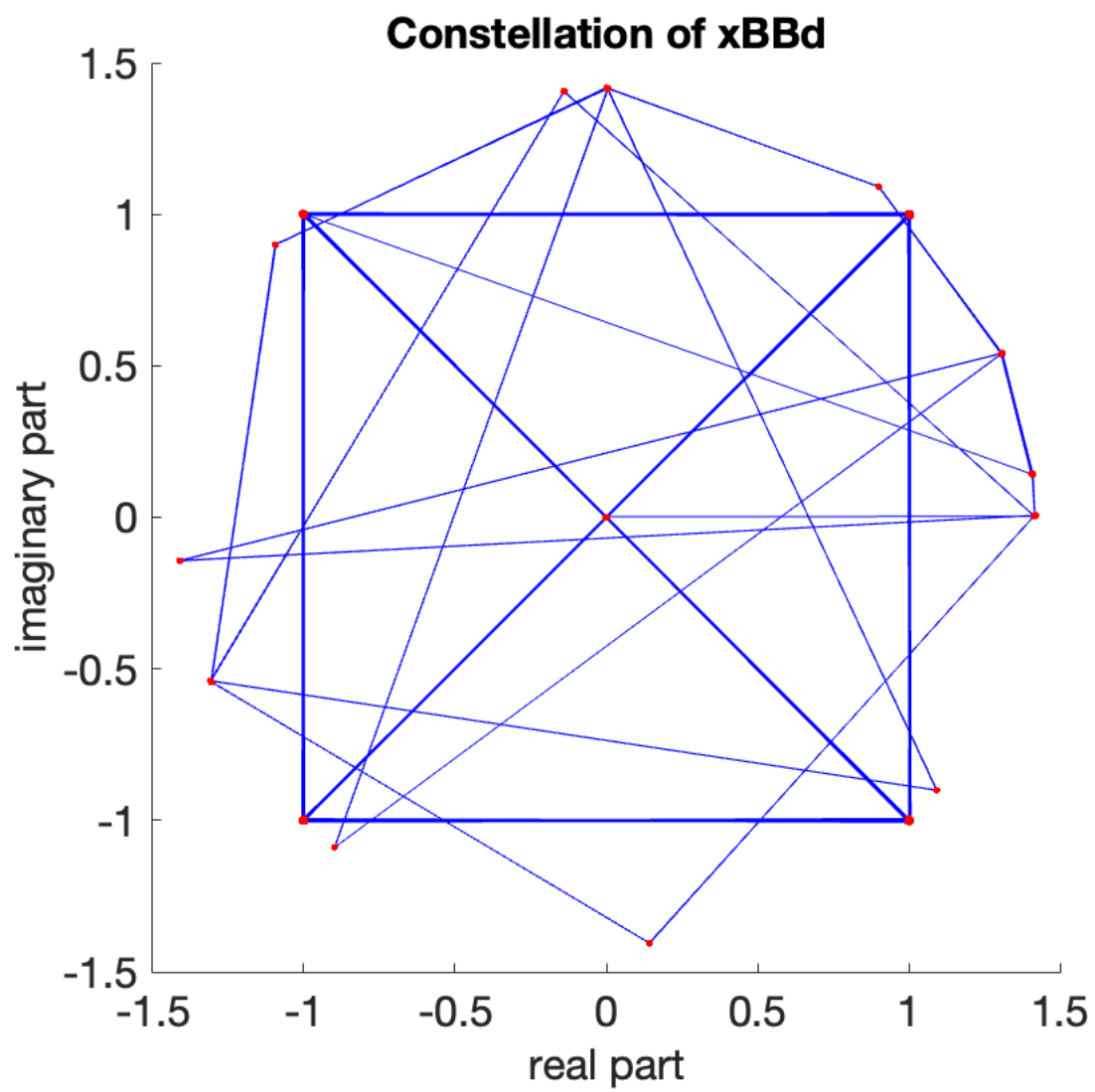
Figure 3: Ensemble Power of xRF1.mat
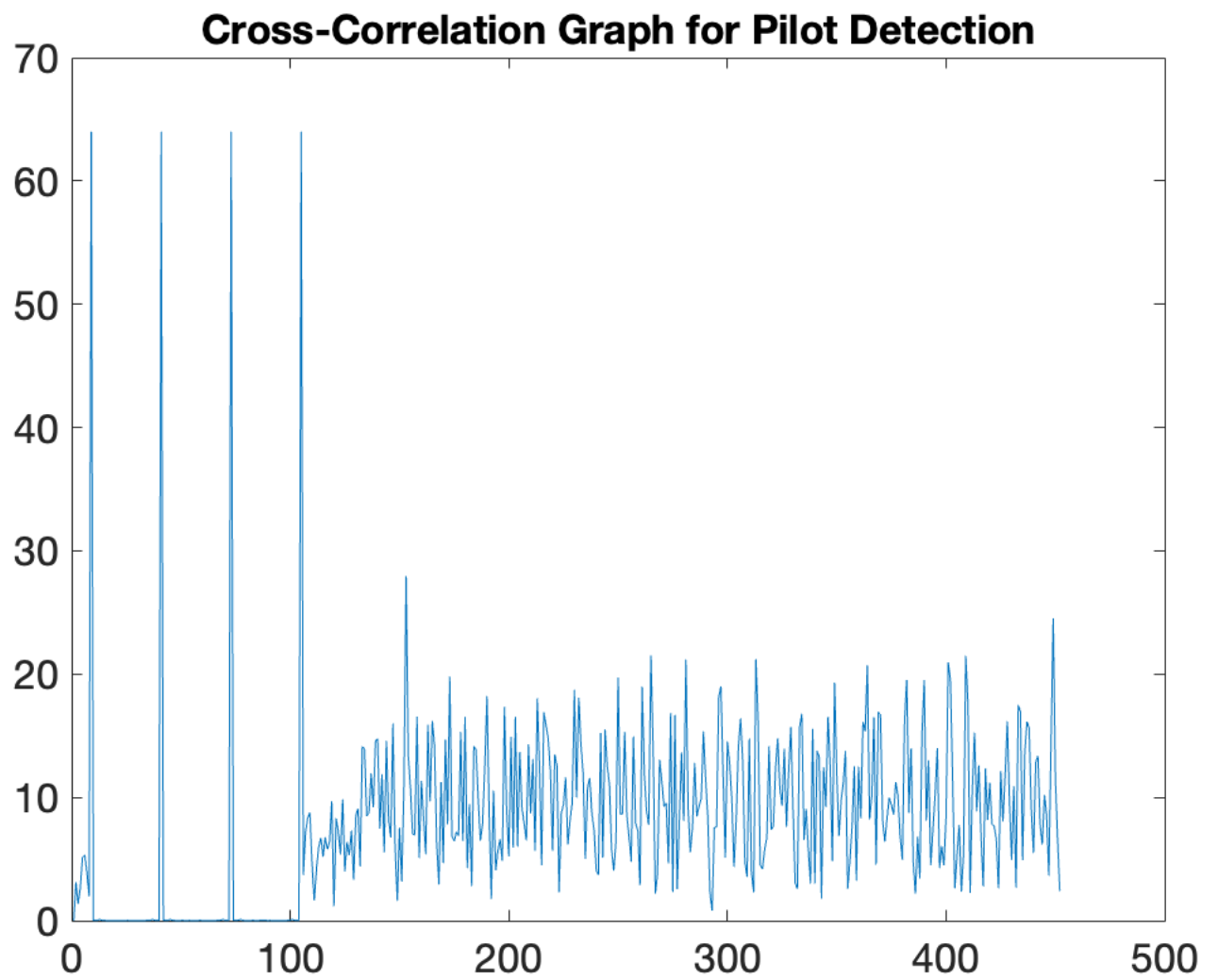
Figure 4: Packet QPSK Constellation of xRF1.mat

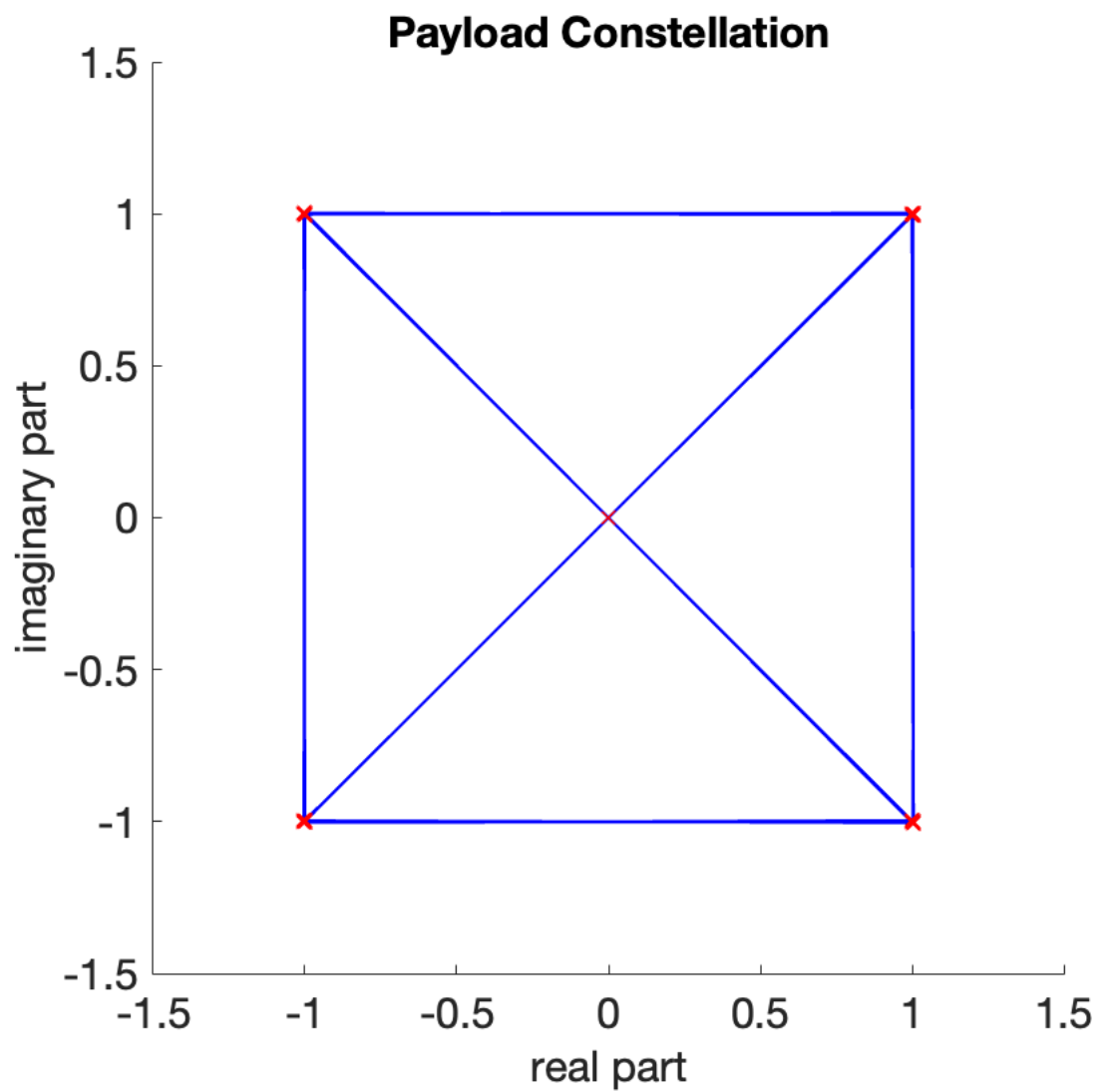Figure 5: Cross-Correlation of Premable with Pilot of xRF1.mat

Figure 6: Payload QPSK Constellation of xRF1.mat

Cyclic preamble	Payload

Figure 7: Structure of transmitted packets in project
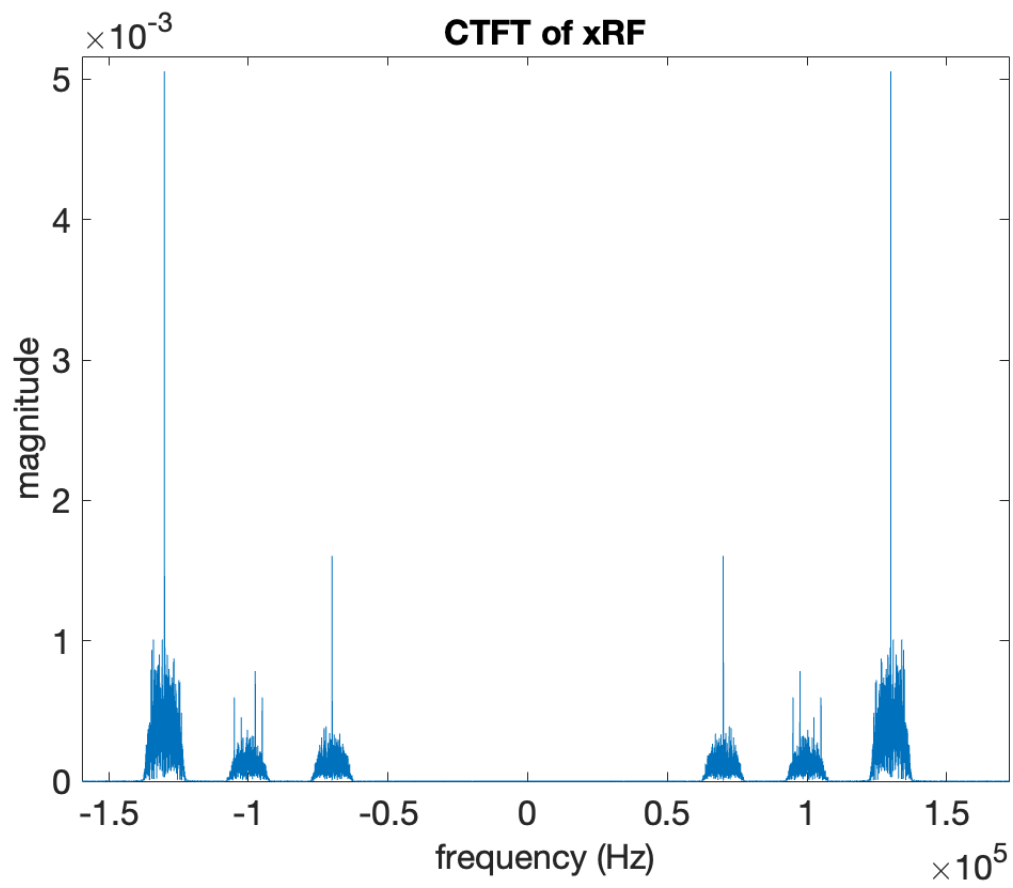
Figure 8: Spectrum of xBB signal in Part I
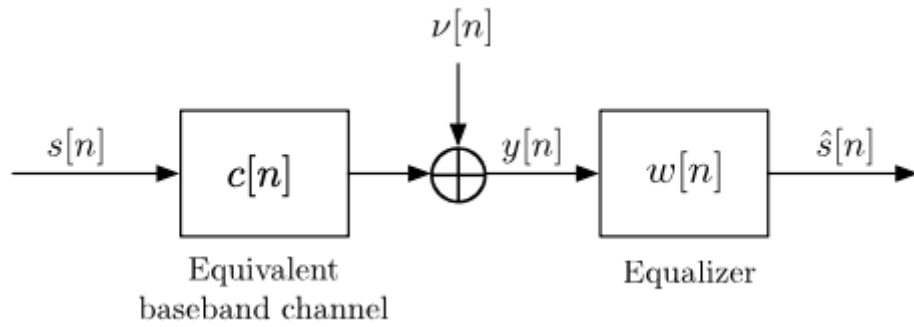
Figure 9: Discrete Time Symbol-Spaced Equalizer
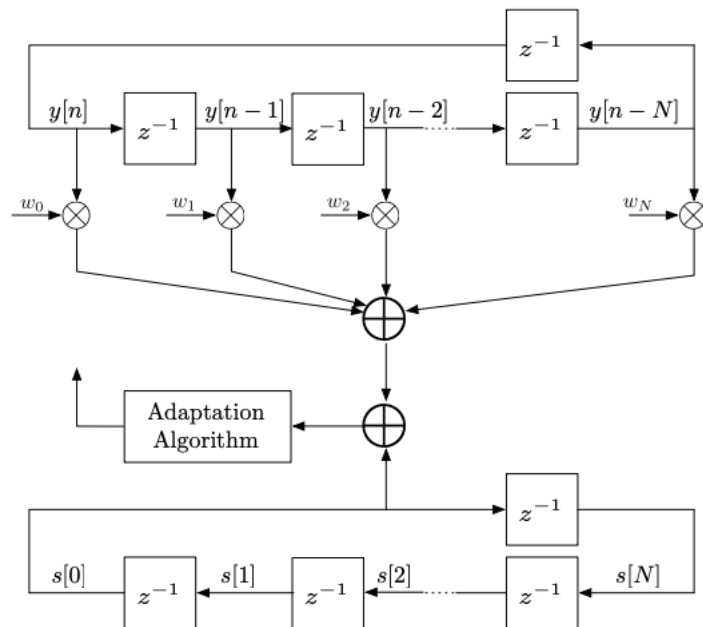
Figure 10: Symbol-Spaced Cyclic Equalizer Training

$$\text{for } i = 0, 1, 2, \cdots$$
$$e[i] = s[i \bmod N + 1] - \mathbf{w}^{\mathrm{H}}[i]\mathbf{y}_i$$
$$\mathbf{w}[i+1] = \mathbf{w}[i] + 2\mu e^*[i]\mathbf{y}_i$$
$$\text{end}$$

Figure 11: Cyclic Symbol-Spaced Equalizer Code
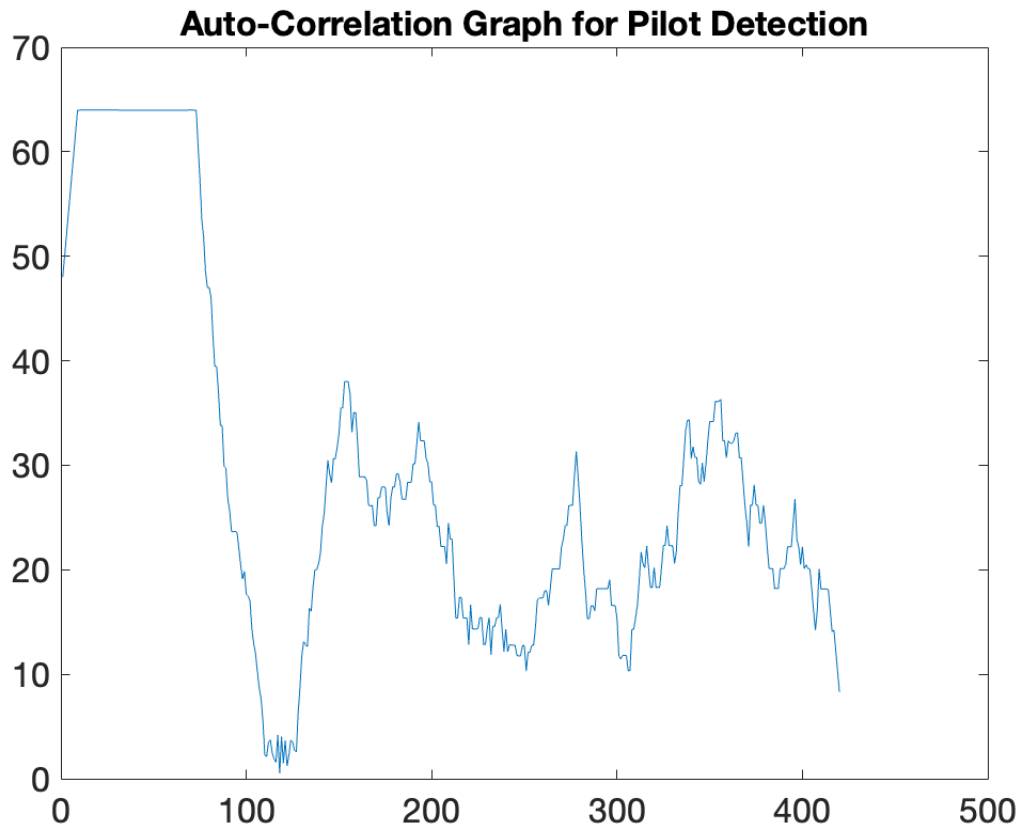
Figure 12: Auto-correlation to Find Preamble on xRF1.mat
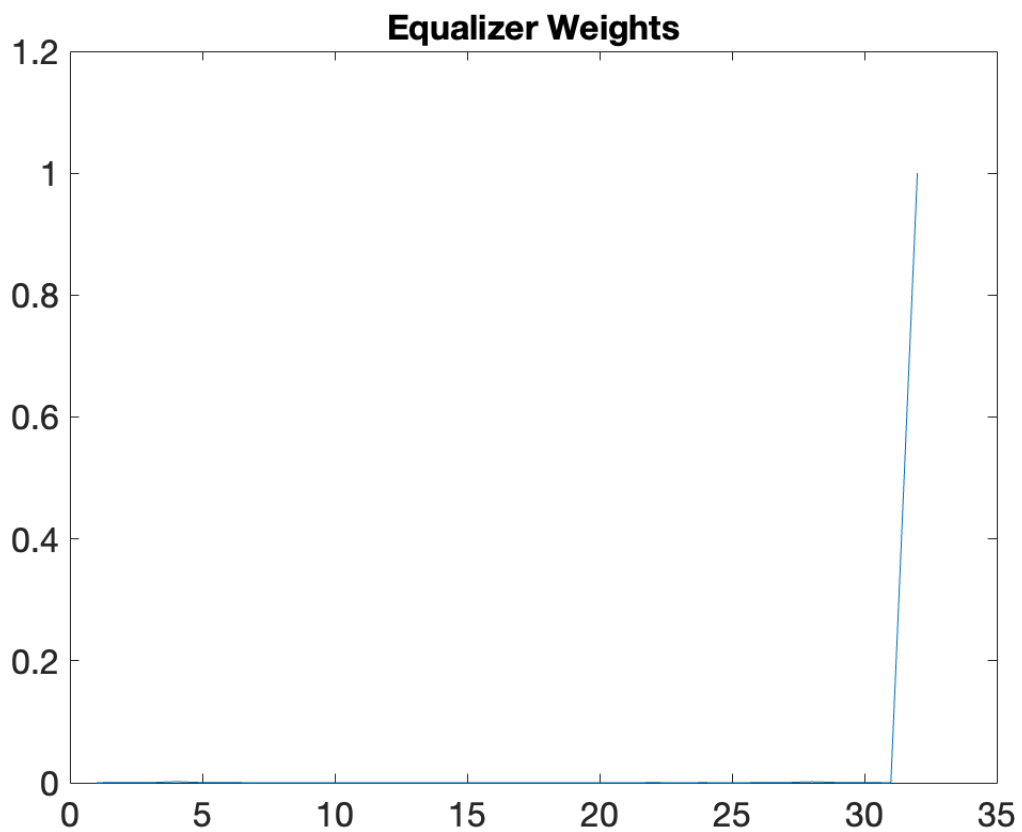
Figure 13: Non-Shifted Tap-Weights of Symbol-Spaced Equalizer on xRF1.mat
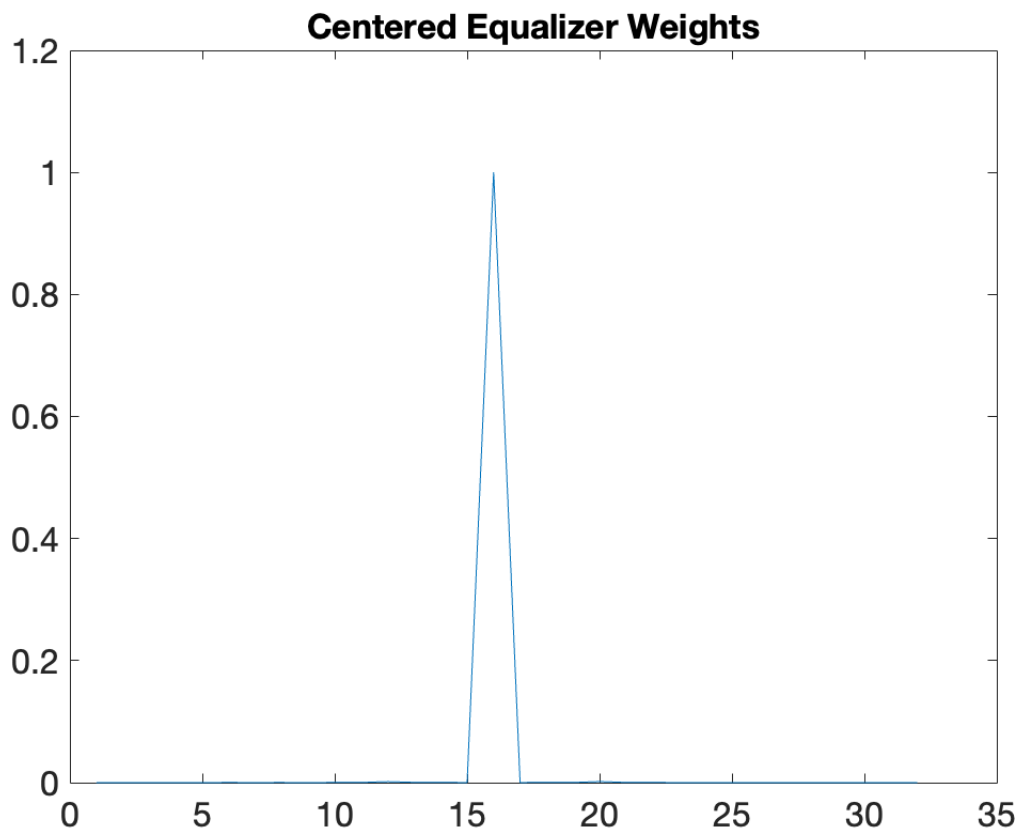
Figure 14: Shifted Tap-Weights of Symbol-Spaced Equalizer on xRF1.mat
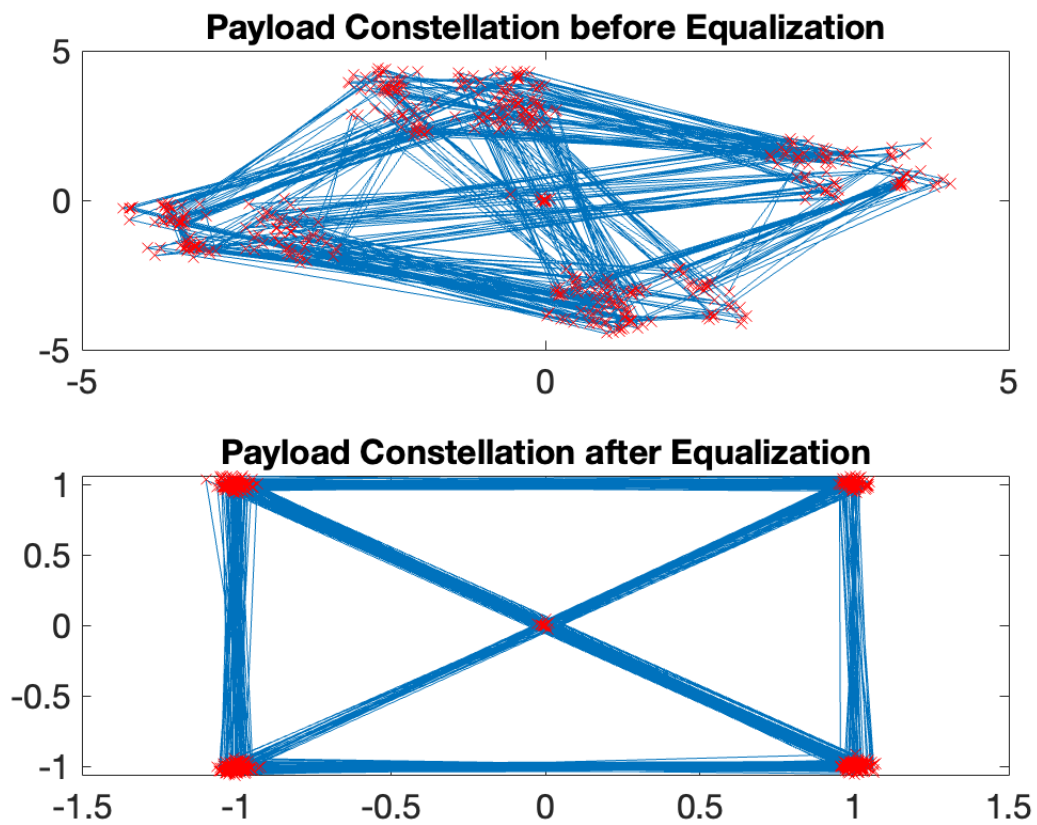
Figure 15: Payload Constellation of xRF3.mat before and after Symbol-Spaced Equalization
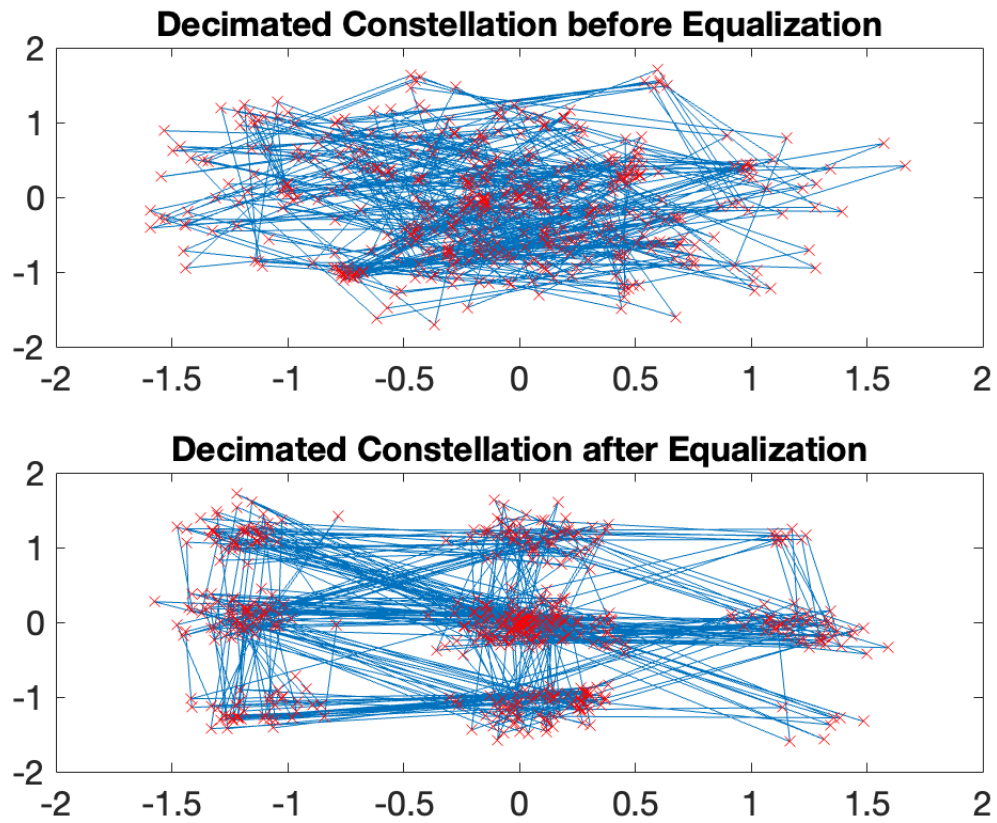
Figure 16: Payload Constellation of xRF5.mat before and after Symbol-Spaced Equalization with Wrong Timing Phase
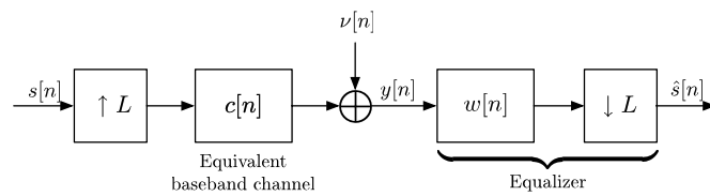
Figure 17: Fractionally-Spaced Equalizer
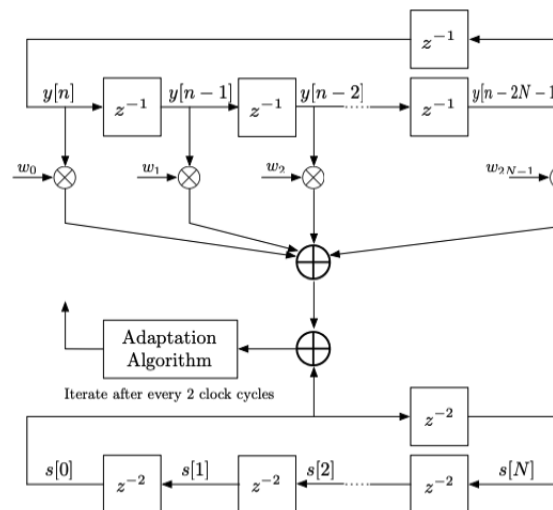
Figure 18: Fractionally-Spaced Cyclic Equalizer
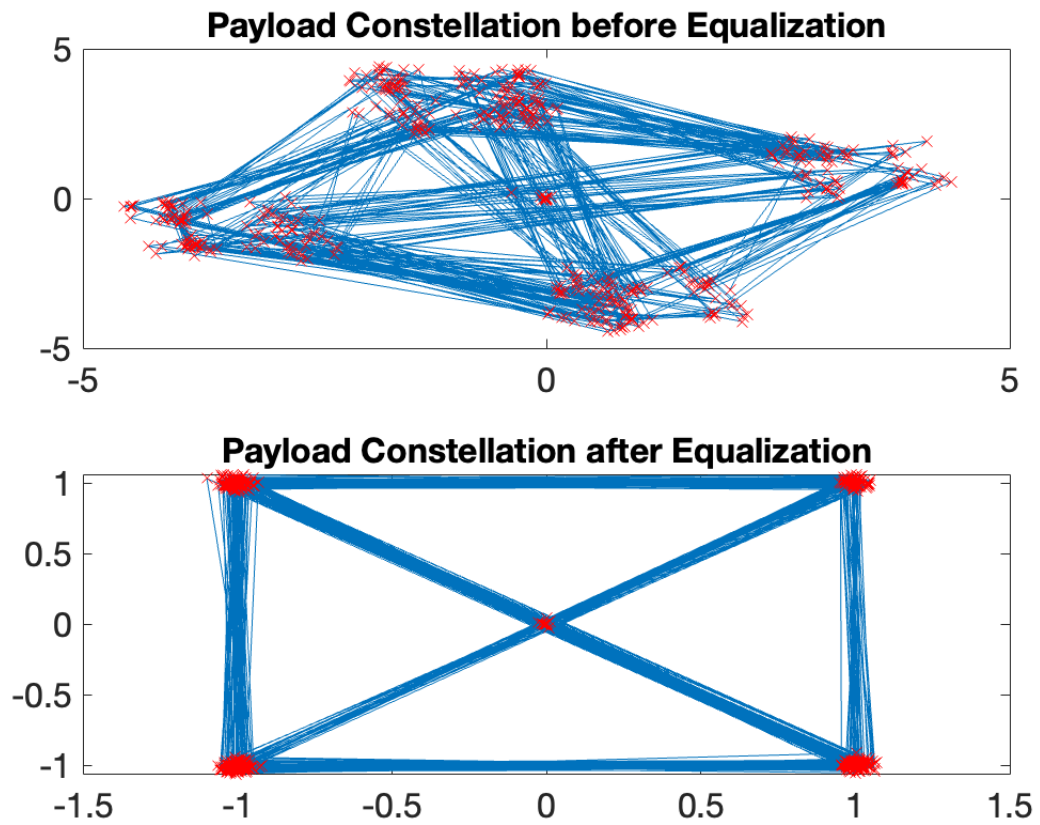
Figure 19: Payload Constellation of xRF3.mat before and after Half Symbol-Spaced Equalization
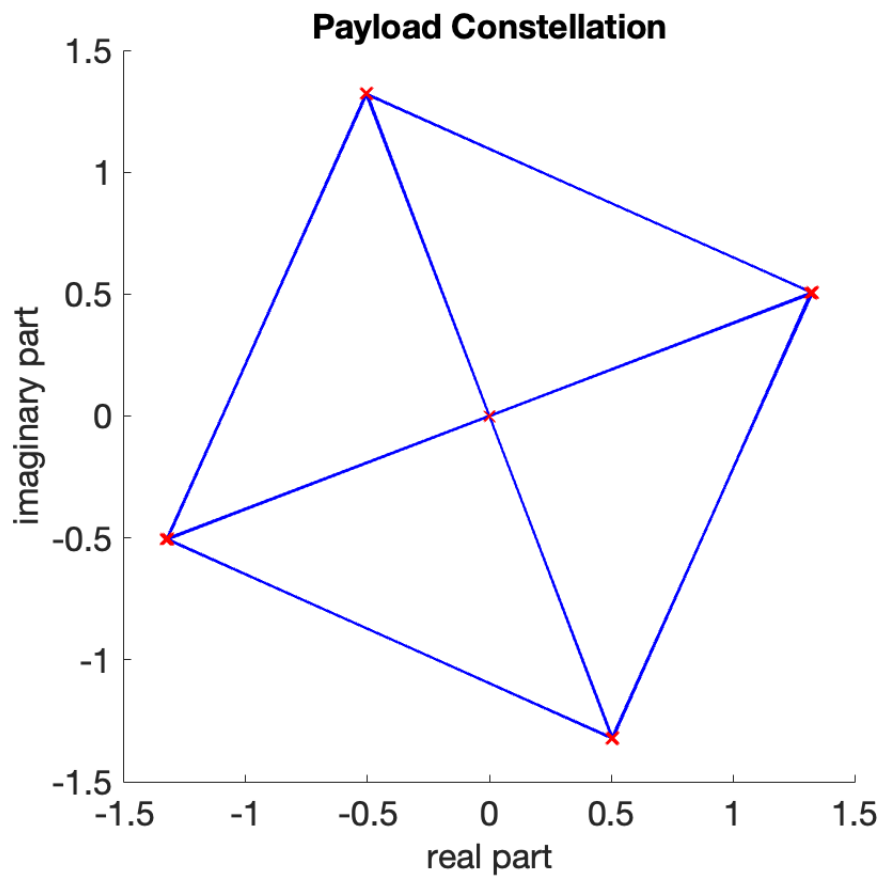
Figure 20: Ideal Channel xRF1.mat with Carrier Phase Offset
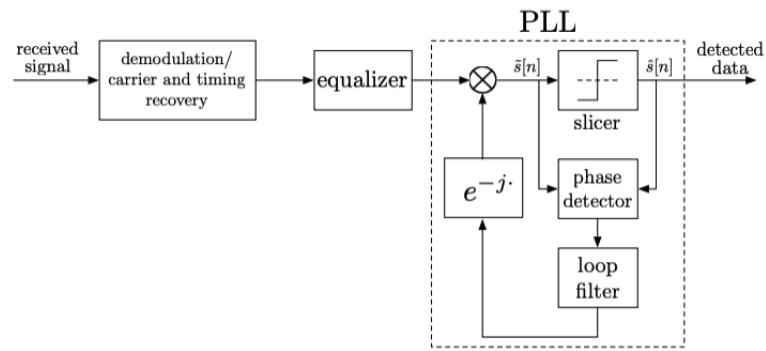
Figure 21: Data-Aided Phase PLL
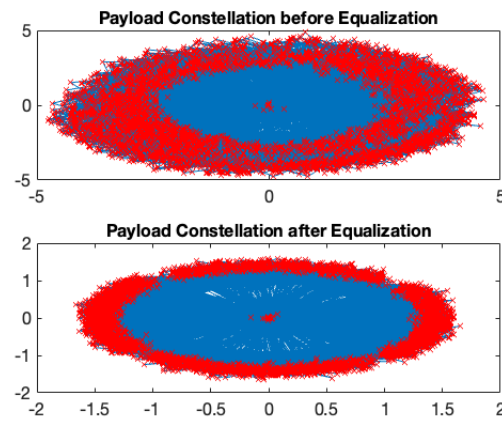
Figure 22: Payload of xRF7.mat Before Decision Directed Carrier Phase Removal
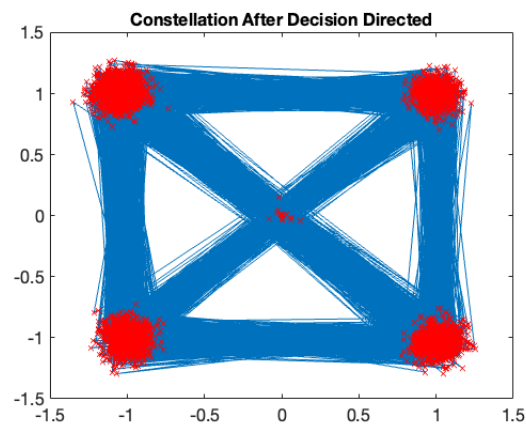
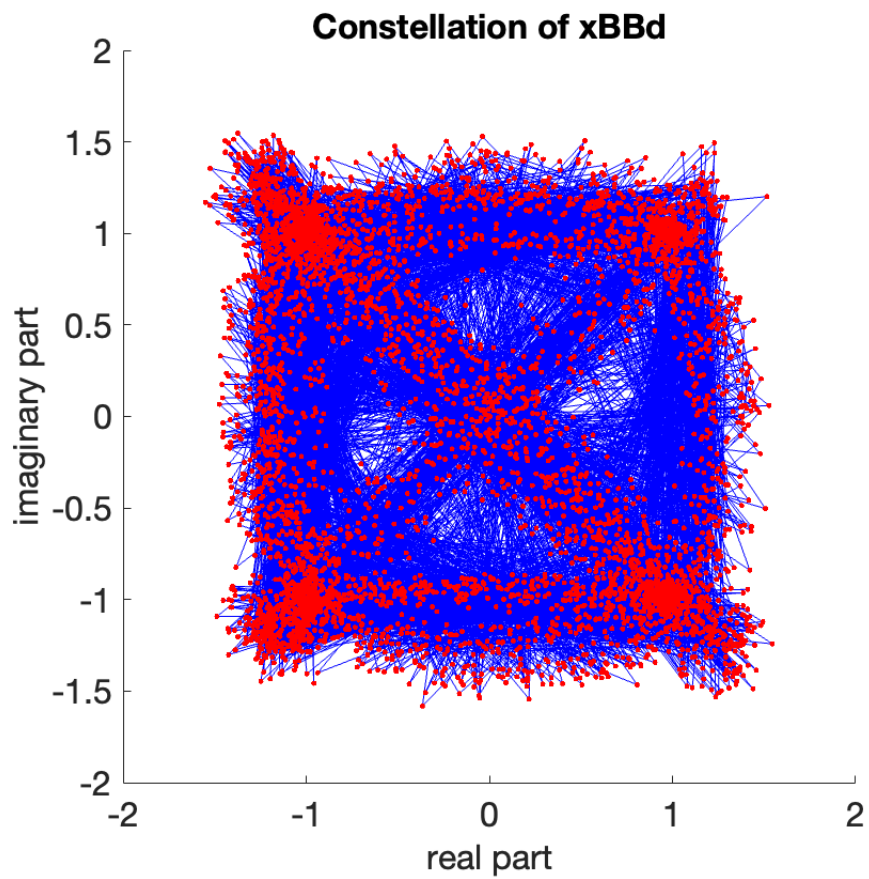Figure 23: Payload of xRF7.mat After Decision Directed Carrier Phase Removal

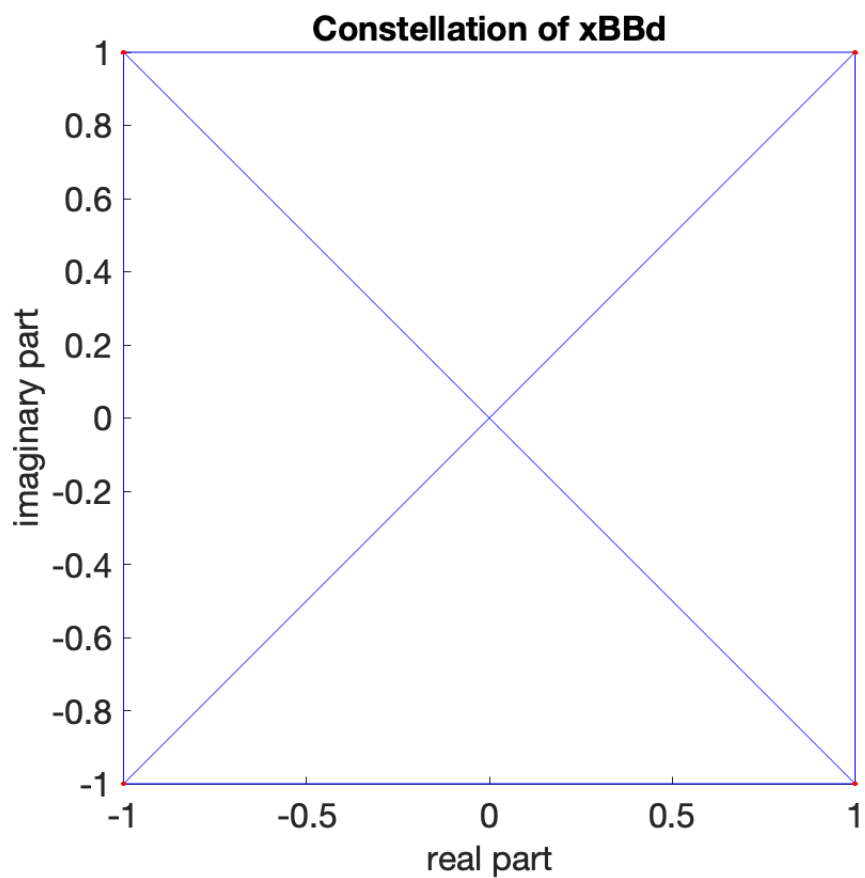Figure 24: Payload of xRF9.mat No Decision Directed Timing Phase Recovery

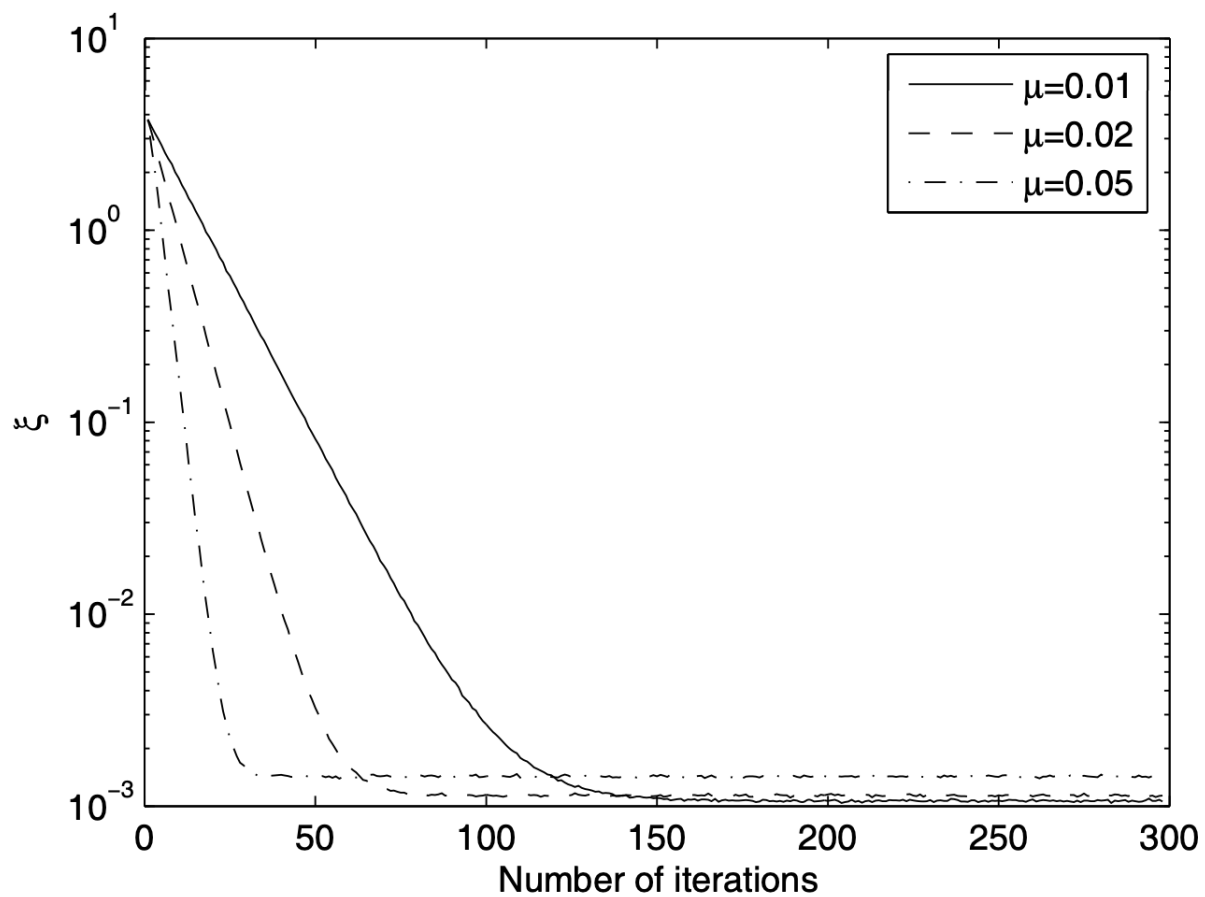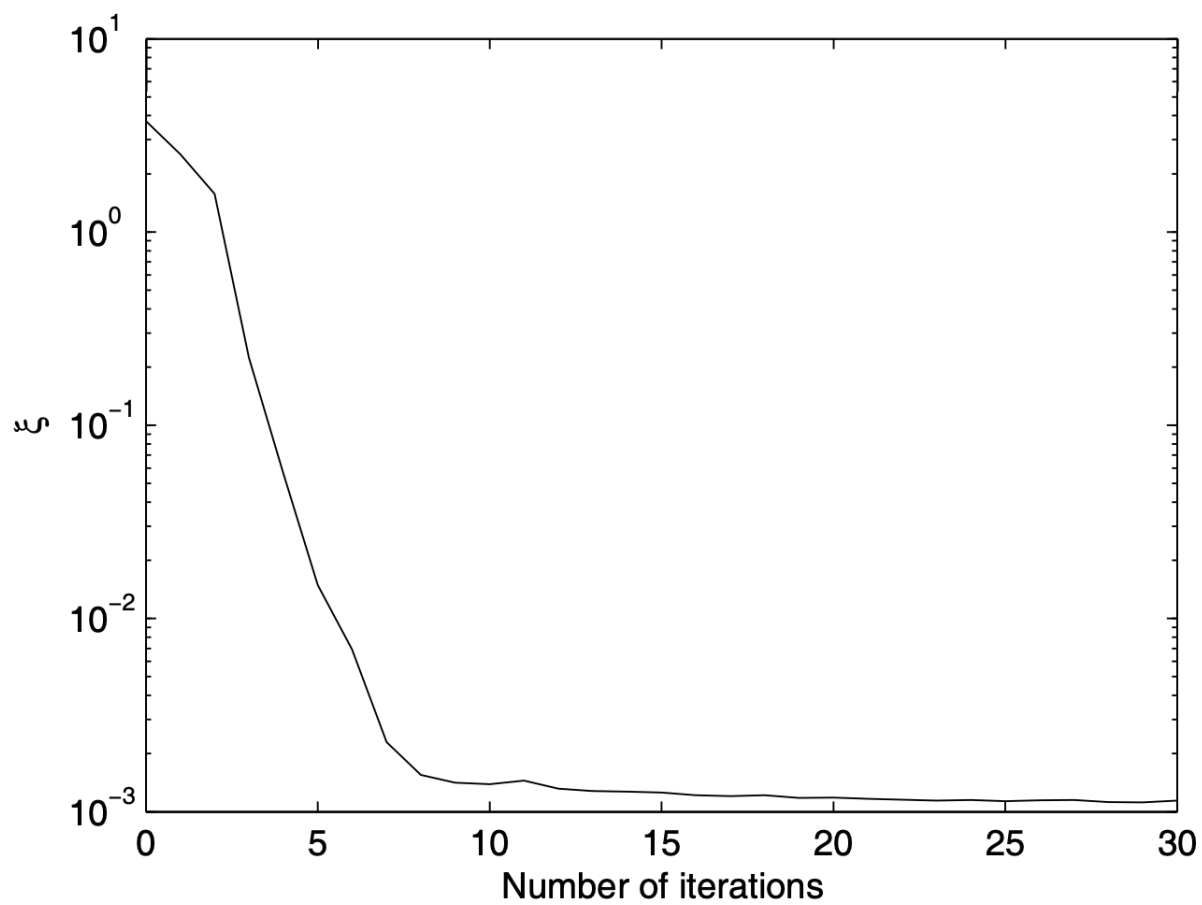Figure 25: Payload of xRF9.mat with Decision Directed Timing Phase Recovery

Figure 26: LMS Convergence Rates

Figure 27: RLS Convergence Rates