ANKARA UNIVERSITY

DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING

GRADUATION THESIS

ANALYSIS AND PERFORMANCE EVALUATION OF A BLIND IQ IMBALANCE COMPENSATION ALGORITHM

Göktuğ GÖKMEN

Yusuf Emre BAYSAL

ETHIC

We hereby declare that all information contained in this thesis, which was prepared in accordance with the thesis writing rules of Ankara University, is accurate and complete. We further declare that scientific ethics were adhered to in generating the information and all sources utilized in the preparation of this work have been cited appropriately.

Date

Signature

Students Name and Surname

Date

Signature

Students Name and Surname

ÖZET

Kablosuz haberleşme sistemleri, yaygınlaşan IoT uygulamaları nedeniyle son yıllarda büyük ölçüde büyümektedir. Daha büyük ağlar ve daha yüksek bit hızına olan ihtiyaç, daha iyi alıcı-verici tasarımlarını gerektirmektedir. Alıcı-verici cihazların tasarım ve üretim aşaması sırasında meydana gelen tutarsızlıklar ve kusurlar sistem performansını büyük ölçüde etkiler. Alıcı-vericilerde demodülasyon sürecinde sinyal bütünlüğünü bozan sorunlardan biri yaşanır. Alıcılardaki dik sinyaller farklı faz ve genliklere sahipse, demodüle edilmiş sinyal "IQ Dengesizliği" ile etkilenir. Bu etki, bit hatalarına ve bazı durumlarda paketlerin tamamen reddedilmesine yol açar.

Bu raporda, frekans bağımsız IQ dengesizliği modelleri ve bu etkinin nasıl telafi edileceği tartışılmaktadır. "Blind Compensation" algoritması analiz edilmiştir ve daha geleneksel olan "Adaptive Compensation" algoritmasıyla karşılaştırılmıştır. Her iki algoritma da IEEE 802.11 simülasyonlarında test edilmiş ve iki algoritmanın benzer sonuçlar verdiği gözlenmiştir. Bu nedenle, "Blind Compensation" algoritmasının IQ dengesizliğini düzeltmek için alternatif bir çözüm olabileceği söylenebilir.

ABSTRACT

Wireless communication systems boomed in size in the recent years because of widespread IoT applications. Bigger networks and the need for higher bitrate require better transceiver designs. Inconsistencies and defects during the design and production phase of the transceivers affect the performance of the system. One of the biggest problems happen during the frequency down conversion process in the transceiver. If the quadrature channels on the receivers differ in phase and amplitude in down conversion, the demodulated signal becomes affected with “IQ Imbalance”. This effect results in bit errors and in some cases complete rejection of packets.

In this report, frequency independent IQ imbalance models are discussed also the ways to compensate it. A blind compensation algorithm is analyzed and compared against the more conventional adaptive compensation algorithm. Both algorithms are tested in IEEE 802.11 simulations and the newly implemented blind algorithm gives similar results to the adaptive algorithm. So, it can be said that the blind compensation algorithm can be an alternative solution for IQ imbalance.

ACKNOWLEDGEMENT

We would like to express our sincere gratitude to out thesis advisor, Selçuk TASCIOĞLU, for his invaluable support, guidance, and encouragement throughout the course of this research. His expertise, insights, and feedback have been instrumental in shaping the scope and quality of this thesis, and We are deeply grateful for his mentorship.

Additionally, we wish to acknowledge the Department of Electrical-Electronics Engineering at Ankara University for providing us with the resources and environment necessary for conducting this research.

Thank you all for your contributions, guidance, and support.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

1 INTRODUCTION

Modern communication systems allow data transactions at increased rates. Since the bandwidth is limited, more and more information is sent with higher degree modulation schemes. This method, however, makes the encoded message more error prone because of the channel effects and defects on the hardware. A common example would be the phenomenon called "IQ Imbalance". There had been methods to counteract this issue [2,3] but the blind approach is not researched in real communication systems a lot.

This work starts with the occurrence of IQ imbalance and the methods used to describe it. Then takes a deep dive into Moseley's Blind Compensation algorithm, implements it into IEEE 802.11 standard and observes its performance against Valkama's "Adaptive Compensation Algorithm". Our aim for this work is to show that the Blind Algorithm gives comparable results to Adaptive Algorithm and can be an alternative method for IQ Imbalance Compensation.

2 IQ IMBALANCE MODELLING

This chapter focuses on general methods to describe IQ Imbalance in a system. Different approaches to this phenomenon give different perspectives to solve it, nevertheless these methods can be interchanged since the underlying event does not change. This work only focuses on frequency independent models however frequency dependent models are also commonly described and used in other works [3].

2.1 Quadrature Signals

Quadrature signals are method commonly used in digital signal processing fields. Also called as "complex signals", these signals are used in a variety of engineering applications. Digital communication systems, which are further explored in this thesis, is their most common use since all the today's popular wireless communication systems (Wi-Fi, BLE, LTE etc.) use quadrature signals in their packet structures. Fundamentals of quadrature signals lies within complex algebra.

A complex number can be visualized in a two-dimensional plane with two axes, real and imaginary. Sum of the real and imaginary components represents a point in the complex plane, and it's shown as:

$$c = a + jb \tag{1}$$

This representation is also called rectangular form. Showing complex number trigonometrically makes it clearer to see advantages of complex numbers. The same point before now can be represented as:

$$c = k(\cos(\theta) + \mathrm{j}\sin(\theta)) \tag{2}$$

Where k is the magnitude of the vector described by the point selected on the complex plane and it can be found with the equation:

$$k = \sqrt{a^2 + b^2} \tag{3}$$

And angle θ is the angle between the vector and the real axis.

Analyzing eq. (2) within signal processing domain, we can come up with the idea of representing a signal as the sum of two sinusoidal with 90 degrees of phase between them. The term quadrature signal comes that 90 degrees of phase angle, which is in place to represent two different signals with just one value.

This idea is essential in digital communication systems since it makes it possible to increase the bit throughput in a channel without needing more bandwidth within the physical layer. A fitting example would be comparing BPSK (Binary Phase Shift Keying) and QPSK (Quadrature Phase Shift Keying) baseband digital modulation methods. Let a bipolar NRZ (Non-Return to Zero) bitstream where the low signal value is -1 and the high signal value is 1. Its presentation on a constellation diagram would be at the right side of the diagram if the sample is high, and left side of the diagram if the sample is low. If we introduce a second bitstream where the signal only has complex components, we can transmit both bitstreams without the need of extra bandwidth and no interferences between the bitstreams.

2.2 Quadrature Mixing

To take advantage of the properties of quadrature signal, quadrature mixers or modulators were developed. Two baseband signals are defined before the modulation process begin, these are called In-Phase(I) and Quadrature-Phased(Q) signals. These two signals are mixed with carrier signals which have 90 degrees of phase angle between them. A practical example would be sine and cosine carrier signals. The modulated signals are then added to create the passband signal that is broadcasted.

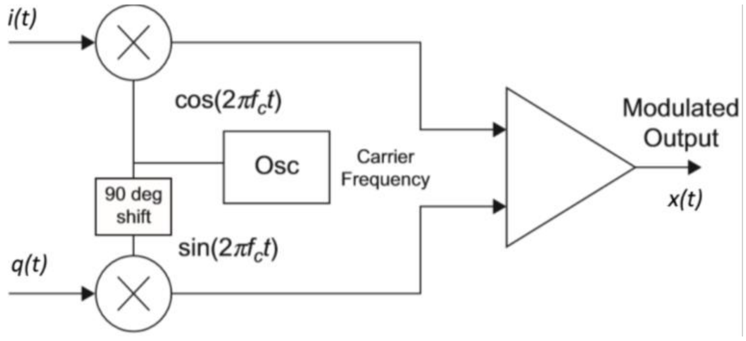

Figure 1 Quadrature Modulator

If we define input signals as $x_I(t)$, $x_Q(t)$ the modulated signal becomes:

$$x_{RF}(t) = \cos(\omega_c t)\, x_I(t) + \sin(\omega_c t)\, x_Q(t) \tag{4}$$

The effect of quadrature modulation can be seen during the demodulation phase. During the demodulation the same $X_{RF}(t)$ signal is mixed with sine and cosine carries in different paths. The resulting signals are shown as:

$$x_{II}(t) = x_{RF}(t)\cos(\omega_c t) = \cos(\omega_c t)^2\, x_I(t) + \cos(\omega_c t)\sin(\omega_c t)\, x_Q(t) \tag{5}$$

$$x_{QQ}(t) = x_{RF}(t)\sin(\omega_c t) = \sin(\omega_c t)\cos(\omega_c t)\;\; x_I(t) + \sin(\omega_c t)^2\, x_Q(t) \tag{6}$$

By using trigonometric identities, the equations can be rewritten as:

$$x_{II}(t) = \frac{1}{2}[\cos(2\omega_c t) \ + x_I(t)] + \frac{1}{2}[\sin(2\omega_c t)\, x_Q(t)] = \frac{1}{2}\left[x_I(t) + \cos(2\omega_c t) + \sin(2\omega_c t)\, x_Q(t)\right] \tag{7}$$

$$x_{QQ}(t) = \frac{1}{2}[\sin(2\omega_c t) \ x_I(t)] + \frac{1}{2}\left[x_Q(t) - \cos(2\omega_c t)\right] = \frac{1}{2}\left[x_Q(t) - \cos(2\omega_c t) + \sin(2\omega_c t)\, x_I(t)\right] \tag{8}$$

It is assumed that the system in question is a linear and time invariant (LTI) system, so it can be seen that if $x_{II}(t)$ and $x_{QQ}(t)$ signals were to pass through lowpass filters that have cutoff frequency selected smaller than $2\omega_c$ the resulting signals $x_{ILP}(t)$ and $x_{QLP}(t)$ would match the starting $x_I(t)$ and $x_Q(t)$ signals.

$$x_{ILP}(t) = LP\{x_{II}(t)\} = \frac{1}{2}x_I(t) \tag{9}$$

$$x_{QLP}(t) = LP\{x_{QQ}(t)\} = \frac{1}{2}x_Q(t) \tag{10}$$

2.3 IQ Imbalance

In a proper quadrature modulated RF signal, I and Q signals should be balanced in such a way that their amplitude should be equal, and their phase angles should be separated by 90 degrees. Any imperfections occurred in the modulation or demodulation process would affect the signal waveform and quantization would result in bit errors. Even though RF frontend design in integrated circuits have come a long way, some irregularities are bound the happen during the production phase of the ICs (Integrated Circuits). Especially local oscillator used in RF design are error prone with clock shift happening in the internal crystal-oscillator circuitry. These errors are device specific and does not change with channel noise, which makes it viable to use IQ imbalance as a classification method in RF fingerprint technologies.

IQ imbalance is categorized in two parts, phase imbalance and amplitude imbalance. Phase imbalance generally is caused by inaccurate timing with the effect of mentioned clock shift and amplitude imbalance is present when amplifier gains of each of I and Q paths does not match. There are two fundamental ways to describe IQ imbalance, single branch(asymmetric) IQ imbalance model and double branch(symmetric) IQ imbalance model.

In single branched IQ imbalance model, one of the signal branches (generally the I branch) is set as a reference and the imbalances are described on the other branch. As an example, for the quadrature modulated signal in eq. (4). If the $x_{RF}(t)$ signal is demodulated with phase imbalance of φ and amplitude imbalance of ε , the resulting signals are now shown as:

$$\dot{x}_{II}(t) = x_{RF}(t)\cos(\omega_c t) = x_{II}(t)$$

(11)

$$\dot{x}_{QQ}(t) = x_{RF}(t)\,\varepsilon\sin(\omega_c t + \varphi) = \varepsilon\cos(\varphi)\sin(\omega_c t)\,x_{RF}(t) + \varepsilon\sin(\varphi)\cos(\omega_c t)\,x_{RF}(t)$$

(12)

$$\dot{x}_{QQ}(t) = \varepsilon\cos(\varphi)\,x_{QQ}(t) + \varepsilon\sin(\varphi)\,x_{II}(t)$$

(13)

The relationship between ideal demodulation and demodulation with IQ Imbalance present can be shown in matrix form as following:

$$\begin{bmatrix} \dot{x}_{II}(t) \\ \dot{x}_{QQ}(t) \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ \varepsilon\cos(\varphi) & \varepsilon\sin(\varphi) \end{bmatrix} \begin{bmatrix} x_{II}(t) \\ x_{QQ}(t) \end{bmatrix}$$

(14)

Another way to describe IQ imbalance is to use double branched IQ imbalance model. In this model the IQ imbalance is shown as a symmetric effect in both of the I and Q channels. The phase imbalance φ is now shown as $\pm\psi$ in I and Q waveforms where:

$$\varphi = 2\psi$$

(15)

Amplitude imbalance is also shown in both channels, amplitude imbalance ε is now shown as $1-\epsilon$ in the I path and $1+\epsilon$ in the Q path where the conversion between the single and double branched models can be shown as:

$$\varepsilon = \frac{1+\epsilon}{1-\epsilon}$$

(16)

Double branch IQ imbalanced demodulation of $x_{RF}(t)$ signal with both of the demodulation paths is imbalanced now becomes:

$$\dot{x}_{II}(t) = (1-\epsilon)x_{RF}(t)\cos(\omega_c t - \psi) = (1-\epsilon)x_{RF}(t)[\cos(\omega_c t)\cos(\psi) - \sin(\omega_c t)\sin(\psi)] \tag{17}$$

$$\begin{aligned}\dot{x}_{II}(t) &= (1-\epsilon)\cos(\psi)\, x_{RF}(t)\cos(\omega_c t) - (1-\epsilon)\sin(\psi)x_{RF}(t)\sin(\omega_c t)\\ &= (1-\epsilon)\cos(\psi)x_{II}(t) - (1-\epsilon)\sin(\psi)x_{QQ}(t)\end{aligned} \tag{18}$$

$$\dot{x}_{QQ}(t) = (1+\epsilon)x_{RF}(t)\sin(\omega_c t + \psi) = (1+\epsilon)x_{RF}(t)[\sin(\omega_c t)\cos(\psi) + \cos(\omega_c t)\sin(\psi)] \tag{19}$$

$$\begin{aligned}\dot{x}_{QQ}(t) &= (1+\epsilon)\cos(\psi)\, x_{RF}(t)\sin(\omega_c t) + (1+\epsilon)\sin(\psi)x_{RF}(t)\cos(\omega_c t)\\ &= (1+\epsilon)\cos(\psi)x_{QQ}(t) + (1+\epsilon)\sin(\psi)x_{II}(t)\end{aligned} \tag{20}$$

If we put eq. (18,20) into matrix form the result will look like:

$$\begin{bmatrix}\dot{x}_{II}(t)\\ \dot{x}_{QQ}(t)\end{bmatrix} = \begin{bmatrix}(1-\epsilon)\cos(\psi) & -(1-\epsilon)\sin(\psi)\\ (1+\epsilon)\sin(\psi) & (1+\epsilon)\cos(\psi)\end{bmatrix}\begin{bmatrix}x_{II}(t)\\ x_{QQ}(t)\end{bmatrix} \tag{21}$$

3 ESTIMATION AND COMPENSATION ALGORITHMS

This section provides a description of Moseley's Blind IQ Imbalance Compensation Algorithm, which is the subject of the thesis, and Antilla and Valkama's Blind IQ Imbalance Compensation Algorithm, which is used as a comparison algorithm to calculate the performance of the former.

3.1 Adaptive IQ Imbalance Compensation Algorithm

In this section, we present an adaptive algorithm designed to mitigate mirror-frequency interference, leveraging the assumption that the ideal baseband signal exhibits properness. By assuming the properness of the ideal baseband signal, compensating for image interference entails restoring properness in the observed signal. Furthermore, practical aspects of compensating for I/Q imbalance are also addressed in a general context [2].

The mentioned algorithm, Antilla's Adaptive IQ Imbalance Compensation Algorithm, is utilized by MATLAB for IQ imbalance compensation in radio receivers. In our research topic of blind IQ imbalance compensation, we will use this algorithm as a comparative benchmark to predict the performance [3].

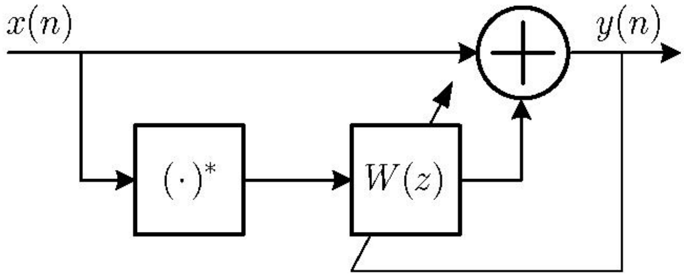

Figure 2 Proposed WL I/Q imbalance compensator structure. [3]

3.2 Blind IQ Imbalance Compensation Algorithm

Blind algorithms have gained popularity in recent years due to their simplicity, low complexity, and ability to estimate the statistical properties of a signal without requiring data training. Among these algorithms, Blind IQ Imbalance Compensation Algorithm is widely used for the estimation and compensation of IQ imbalance. The block diagram of this algorithm, as introduced in [1], is shown in the figure below. In this thesis, the Blind IQ Imbalance Compensation Algorithm is investigated in detail, with a focus on its performance in comparison to the Adaptive IQ Imbalance Compensation Algorithm, which is used as a comparison algorithm in this thesis. To correct the imbalance, I and Q channels can be amplified by some weights to correct the diagram to a circular one. The compensation algorithm estimates three parameters θ_1, θ_2, and θ_3. These three parameters are used to find weights c1 and c2 which will be used as coefficients of the I and Q signals.

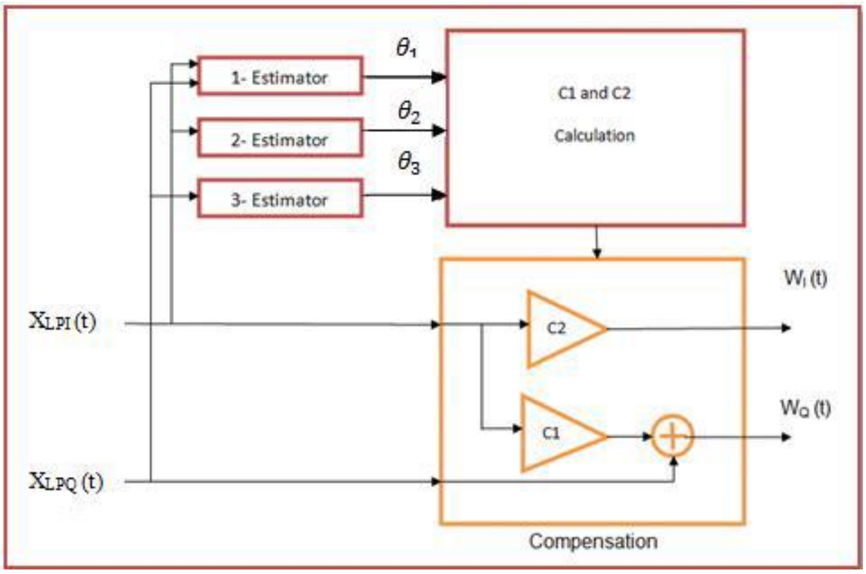

Figure 3 Implemented IQ Imbalance Compensation According to the Algorithm in [2]

The θ parameters are estimated from the IQ imbalanced signal after low pass filter.

$$x_{LP}(t) = x_{LPI}(t) + jx_{LPQ}(t) \tag{22}$$

$$\theta_1 = -1 * mean(sign(x_I(t))x_Q(t)) \tag{23}$$

$$\theta_2 = mean(|x_I(t)|) \tag{24}$$

$$\theta_3 = mean(|x_Q(t)|) \tag{25}$$

Weights c_1 and c_2 are calculated by using these parameters.

$$c_1 = \frac{\theta_1}{\theta_2} \tag{26}$$

$$c_2 = \sqrt{\left(({\theta_3}^2 - {\theta_1}^2)/{\theta_2}^2\right)} \tag{27}$$

After finding the weights c_1 and c_2, IQ compensated wave $w(t)$ calculated as following:

$$w_I(t) = c_2 * x_I(t) \tag{28}$$

$$w_Q(t) = c_1 * x_I(t) + x_Q(t) \tag{29}$$

$$w(t) = \left[w_I(t) + j\, w_Q(t)\right]/c_2 \tag{30}$$

4 SIMULATION RESULTS AND COMPARISONS

This chapter of the thesis presents an extensive analysis of the Adaptive IQ Imbalance Compensation Algorithm and Blind IQ Imbalance Compensation Algorithm through numerous and diverse simulations. The simulation processes, including the methodology and the parameters, are explained in detail, followed by a thorough examination of the results. The simulation outcomes are presented in terms of the Bit Error Rate (BER) of each algorithm for various packet counts. In addition, the performance of the algorithms under different levels of amplitude and phase imbalance is discussed.

4.1 Simulink Digital Modulation Simulation

In this study, a simple transceiver model was created using MATLAB Simulink, and a simulation of data transmission and reception was conducted. Binary data was initially modulated, and the I and Q channels were passed through a mixer and an AWGN channel. Then, on the receiver side, the signal was attempted to be demodulated to obtain the original signal. It was observed that data transmission between the transmitter and receiver was successfully achieved by the Simulink model. Asymmetric IQ imbalance was subsequently added to both the local oscillator on the transmitter side and the local oscillator on the receiver side for conducting simulation studies.

In the simulation, the performance metrics of the Adaptive IQ Imbalance Compensation Algorithm and the Blind IQ imbalance compensation algorithm studied in the thesis were compared. The correction algorithm was implemented using the Adaptive IQ Imbalance

Compensation block available in Simulink. For the Blind IQ imbalance correction algorithm studied in the thesis, it was implemented in Simulink.

A signal with 7.5 degrees of phase imbalance and 13dB amplitude imbalance on the transmitter side, and 30 degrees of phase imbalance and 7dB amplitude imbalance on the receiver side, was passed through a channel with a 20dB SNR. The resulting constellation diagram is shown with blue dots. The signal was then compensated using the Adaptive IQ Imbalance Compensation Algorithm and the Blind Compensation Algorithm.

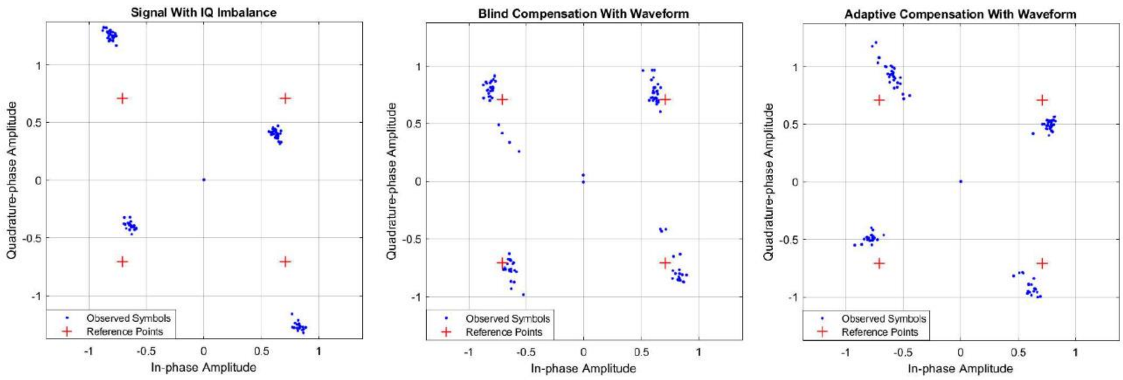

Figure 4 Compensation Algorithms Comparison

Both algorithms were applied on whole waveform of the input signals. Preliminary results show that blind algorithm is able to correct the imbalance presented in simulation. It can be seen that the blind algorithm has better performance then adaptive algorithm in this system. To show the quantitative results, bit error rate calculation is conducted on both of the compensated signals. The results justify discovering blind algorithm further in real digital communication systems.

Table 1 Simulink BER Comparison

	Blind Algorithm	**Adaptive Algorithm**
Bits Sent	2000	2000
Correct Bits	1993	1891
Incorrect Bits	7	109
BER(%)	0,35	5,45

4.1.1 Comparison of IQ Imbalance Compensation using Waveforms and Constellation Symbols

One question while implementing the blind algorithm was whether to apply the algorithm on the whole waveform or the sampled constellation points. To test the difference between the two methods of application, blind algorithm was applied by using all the whole waveform, and then using the constellation points. These methods were also compared on the adaptive algorithm.

There was not much difference observed between the two methods in both algorithms. The only difference was the need of adjusting the step size variable on the adaptive algorithm for it to respond swiftly with reduced number of data points.

We believe that the reason why two methods of implementation does not differ in this system is because of the modulation type used. In simple modulation schemes such as QPSK, QAM, the signals generated after demodulation are NRZ square waves, so there is not much difference between the average value of the waveform in a given sample range. This difference may be larger in modulation types that require additional processing after being reduced to baseband, such as OFDM. In addition, the system's sampling rate is also of great importance.

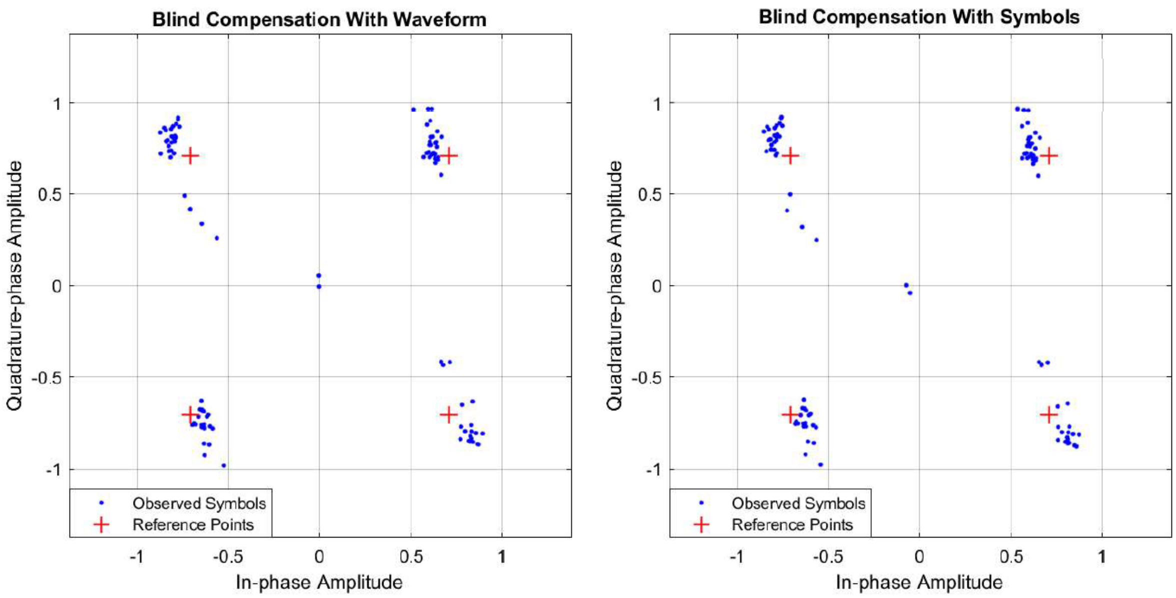

Figure 5 Two Methods of Implementation for Blind Algorithm

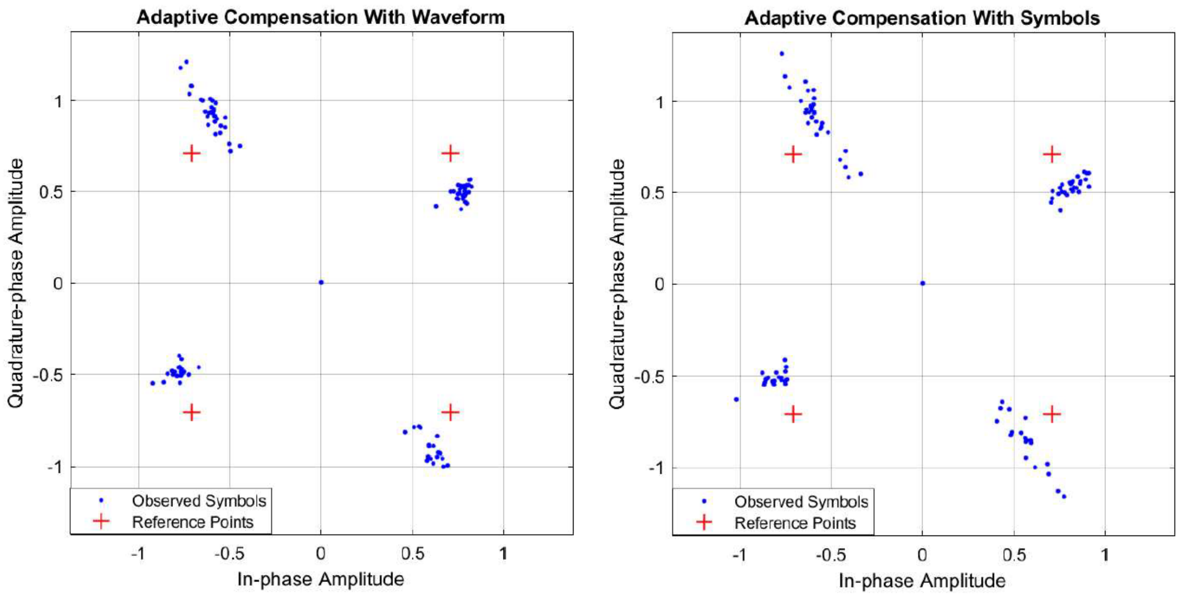

Figure 6 Two Methods of Implementation for Adaptive Algorithm

4.2 IEEE 802.11a MATLAB Image Transmission Simulation

In this section, instead of creating a simulation in the Simulink environment to progress over WiFi 802.11a/802.11n signals, which is our main area of work in IQ Imbalance Compensation for the thesis, we decided to create a simulation environment through the MATLAB WLAN Toolbox and continue our work from there.

In the transmitter part, a picture file is converted into PHY Layer Service Data Units (PSDUs) using WLAN Toolbox functions. Each PSDU in the simulation uses a single non-high-throughput (non-HT), 802.11a WLAN packet for transmission. Then, the generated waveform passes through an additive white Gaussian noise (AWGN) channel to simulate an over-the-air transmission, and IQ Imbalance is added after passing the AWGN channel, to simulate receiver IQ Imbalance.

In the receiver part, noisy and IQ Imbalanced signal is decoded, MAC addresses are extracted and checked for compatibility. If the MAC address matches, the MPDU signal packet is extracted from the signal, the data symbols decoded in 64QAM format are interpreted and converted into bits, and the transmitted data (the image transmitted in this scenario) is reconstructed and displayed on the screen.

The following two figures show the spectrum analysis and packet timing in the time domain of a signal that has passed through a 20dB AWGN channel and has been affected by 5dB amplitude imbalance and 10 degrees phase imbalance. The IQ Imbalance is added to the source signal by using MATLAB WLAN Toolbox's IQ Imbalance adding function.

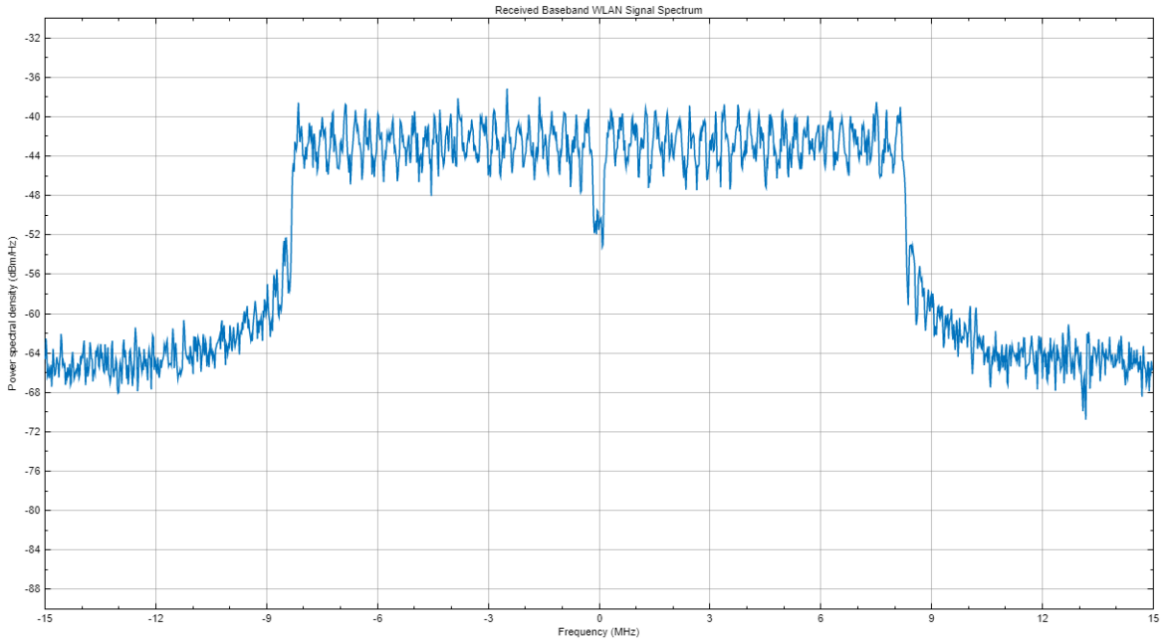

Figure 7 Spectrum analysis of the signal received after passing through a 20 dB AWGN channel, with 10 degrees of phase imbalance and 5 degrees of amplitude imbalance.

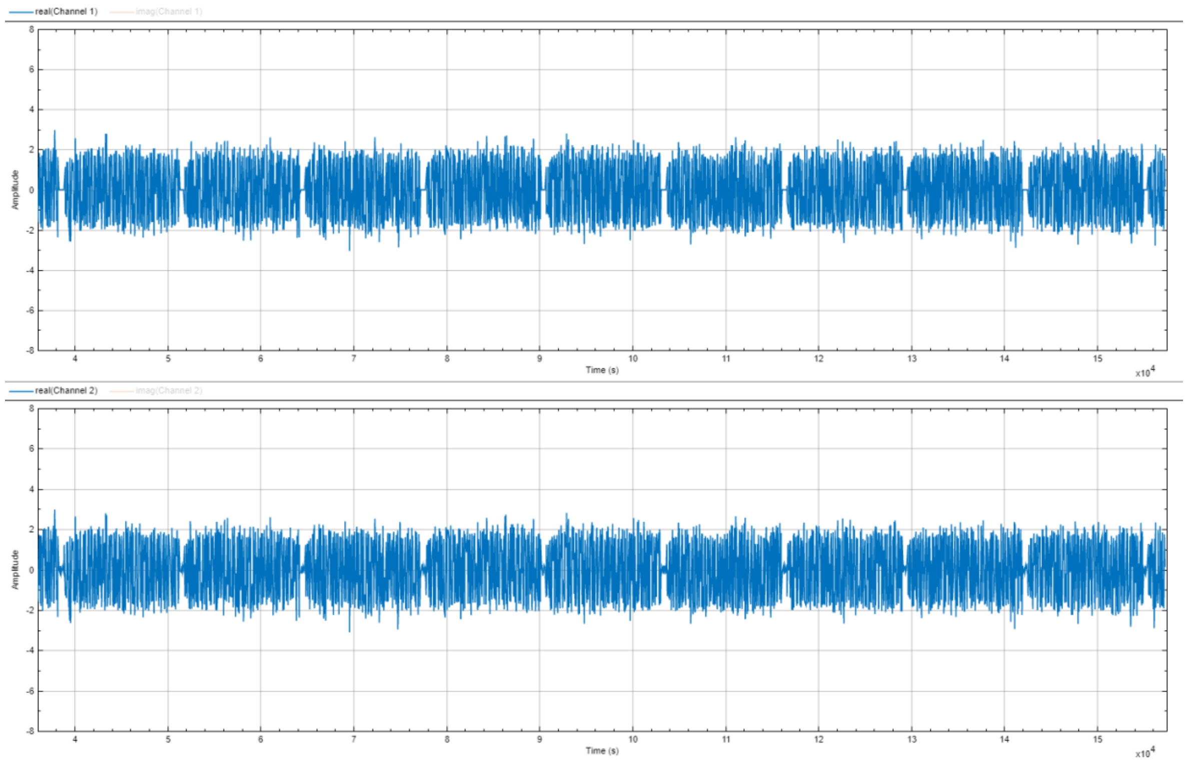

Figure 8 Packets Sent (Top) and Received (Bottom)

The following two figures show the constellation diagram of the signal with only 1dB Amplitude Imbalance, and 2 degrees phase imbalance added, as well as the transmitted and received image.

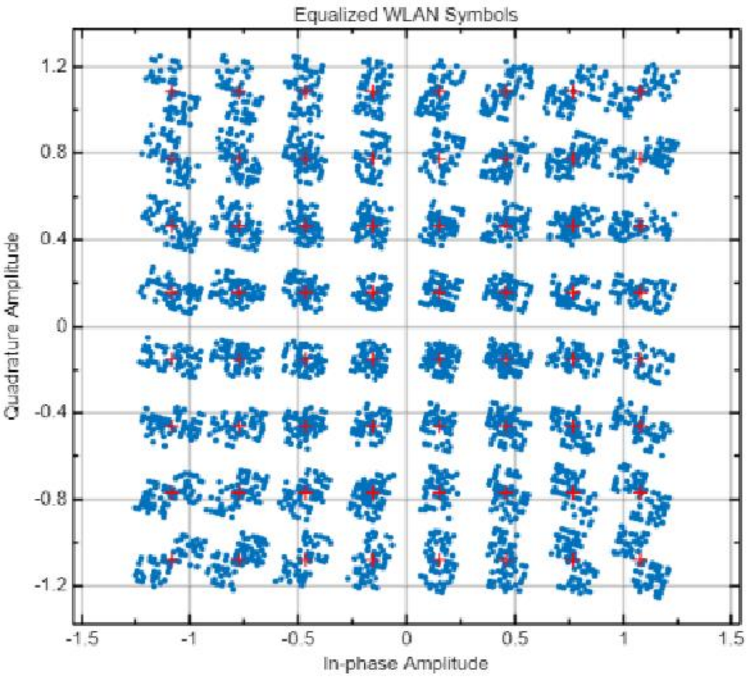

Figure 9 Constellation Diagram with 1dB Amplitude Imbalance and 2 degrees Phase Imbalance

Figure 10 Transmitted and Received Image with 1dB Amplitude Imbalance and 2 degrees Phase Imbalance

The following two figures show the constellation diagram and transmitted/received image of a signal with 5dB Amplitude Imbalance, and 10-degree phase imbalance added.

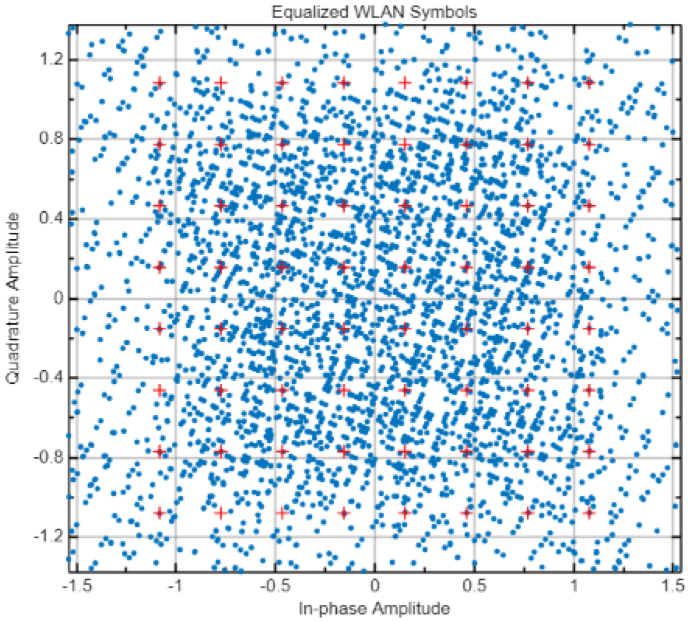

Figure 11 Constellation Diagram with 5dB Amplitude Imbalance and 10 degrees Phase Imbalance

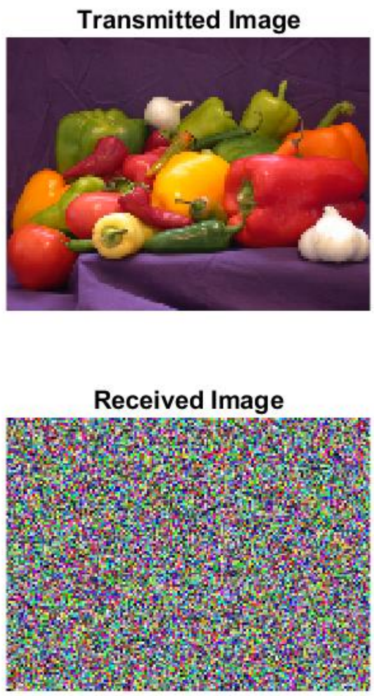

Figure 12 Transmitted and Received Image with 5dB Amplitude Imbalance and 10 degrees Phase Imbalance

The following two figures show the constellation diagram and the transmitted and received image of a signal with 5dB amplitude imbalance and 10-degree phase imbalance, followed by passing through a 30dB AWGN channel.

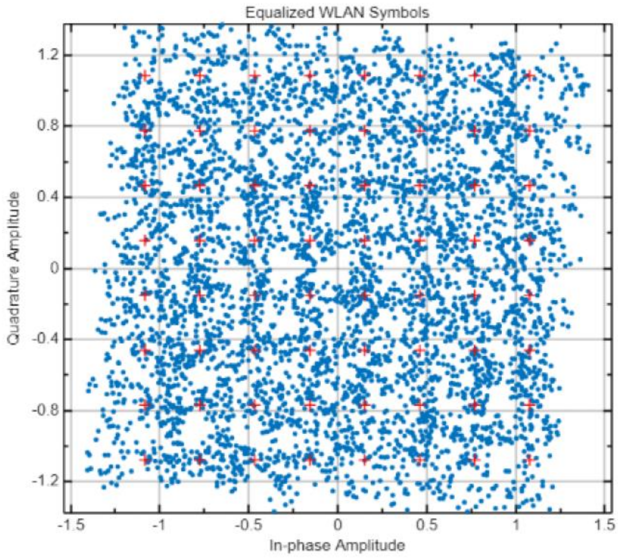

Figure 13 Constellation Diagram with Signal Passed 30db AWGN Channel, 5dB Amplitude Imbalance and 10 degrees Phase Imbalance

The following images show how IQ imbalance is compensated packet by packet when MATLAB's Adaptive IQ Imbalance Compensation algorithm function is used. The bit error ratios (BER) are as follows. As can be seen in the images, the bits carrying the information of the red color in the initially transmitted packets are significantly erroneous until the coefficients of the algorithm are calculated with high accuracy. This situation can also be observed by looking at the left side of the received image (the first transmitted bits appear in red color on the left side of the image and are incorrect).

Bit Error Rate (BER) = 0.02002

Number of bit errors = 8486

Number of transmitted bits = 423936

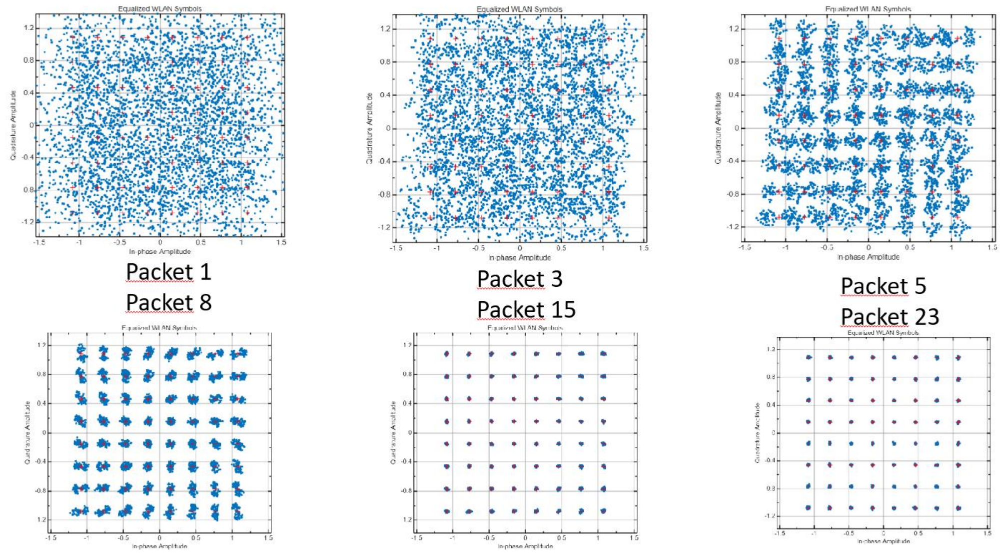

Figure 14 Constellation Diagrams of Adaptive IQ Imbalance Compensation Algorithm with 30db SNR, 2dB Amplitude Imbalance and 10 degrees Phase Imbalance

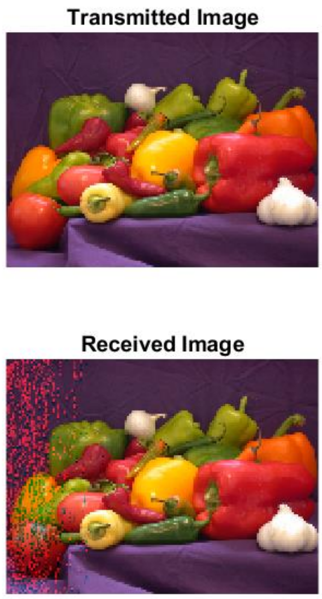

Figure 15 Transmitted and Received Image IQ Imbalance Compensation with Adaptive Algorithm

When the success of the Blind IQ Imbalance Compensation Algorithm in IQ Imbalance Compensation was studied and a simulation was conducted, it was observed that the results were quite successful. It should be noted that, as stated in the source paper, the Blind IQ Imbalance Compensation Algorithm was directly implemented into MATLAB. However, in [1], the average of the entire signal was taken when calculating the coefficients needed for IQ Imbalance Compensation by the algorithm. Although this method is very successful for calculating coefficients for IQ Imbalance Compensation for a signal that is already available from beginning to end, it cannot be used in a real-time processing system because it disrupts the causality of the system. To overcome this situation, the Blind IQ Imbalance Compensation Algorithm was modified by the team using a moving averaging filter, and it was implemented into MATLAB in its new form. Different window sizes were tried to determine the optimal window size for the moving averaging filter.

The following constellation diagram shows the result obtained after implementing the Blind IQ Imbalance Compensation Algorithm in MATLAB using the same values as in the Adaptive IQ Imbalance Compensation Algorithm function. In this simulation, a Window Size of 500 is used for the Averaging Filter.

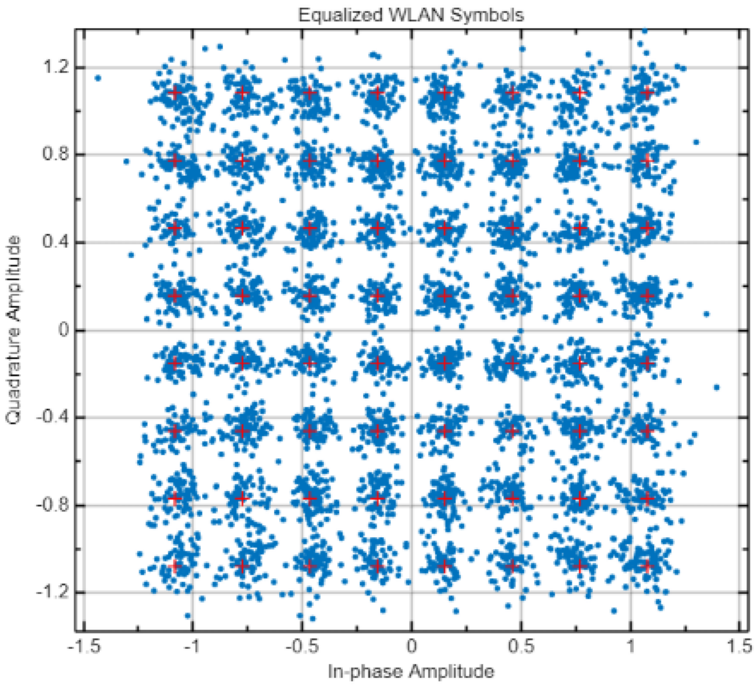

Figure 16 Constellation Diagrams of Blind Algorithm with 30db SNR, 2dB Amplitude Imbalance and 10 degrees Phase Imbalance

Figure 17 TX and RX Image IQ Imbalance Compensation with Blind Algorithm

4.3 IQ Imbalance Compensation Simulation in IEEE 802.11

In this section, Monte Carlo simulations were performed to obtain more reliable and robust data for a scientific thesis based on the IEEE 802.11a MATLAB Image Transmission Simulation section. Monte Carlo simulations are commonly used when a specific system or event is characterized by random variables. These simulations allow for probabilistic estimation of results by performing random sampling multiple times. In the thesis simulation, Monte Carlo simulations were employed to generate repeated iterations of both amplitude imbalance and imbalance within specified ranges while keeping the signal-to-noise ratio (SNR) level constant. The results obtained at different SNR levels were visualized and interpreted on a single graph. These simulations aimed to enhance the reliability and validity of the data obtained for the thesis.

The following settings were used for Monte Carlo simulations:

- SNR at between 15dB and 25dB, increasing 1dB by 1dB.
- Amplitude Imbalance between 0.5dB and 2.5dB randomly selected,
- Phase Imbalance between 8° and 12° randomly selected,
- 1000 iterations are performed for each SNR level,
- 23 packets sent in each iteration,

- For Blind Algorithm, window size is set as 500.

At each SNR level, a randomly selected value within the specified amplitude imbalance and phase imbalance ranges was used for the simulation. The simulation involved comparing results for two different values.

Firstly, the Packet Decode Rate (PDR) was examined, which refers to the receiver's ability to correctly decode the MAC address and sequence number of the received packet. This analysis determined whether the packet could be successfully interpreted or not, independent of the error rate, in order to visualize the reconstructed image. The PDR values obtained at each SNR level were combined and visualized.

Secondly, the Bit Error Rate (BER) was calculated. While reconstructing the image at the receiver, a comparison was made between the original bits generated at the transmitter and the decoded bits at the receiver. The differences between them were recorded as the Bit Error Rate. The BER values obtained at each SNR level were combined and visualized.

PDR and BER values were obtained and recorded 1000 times at each SNR level, and the average BER and PDR were calculated based on the results of these 1000 simulations. The calculated average BER and PDR values were used to generate the graphs described in the subsections below this section.

4.3.1 Packet Detection Rate Results

This section includes the Packet Decode Rate - Signal to Noise Ratio (PDR - SNR) graphs obtained from the simulations conducted as described in the previous section. To enhance the clarity of the results and show the variations more clearly, interpolation has been applied to the PDR values.

The figure below illustrates the Packet Decode Rate values at different SNR levels after passing the signal through only the AWGN channel. When the SNR level is 15dB, the PDR is almost %0. As the SNR level increases to 17dB, the PDR rises to around 70%. For SNR levels of 18dB and above, the PDR surpasses 95% and converges towards 100%.

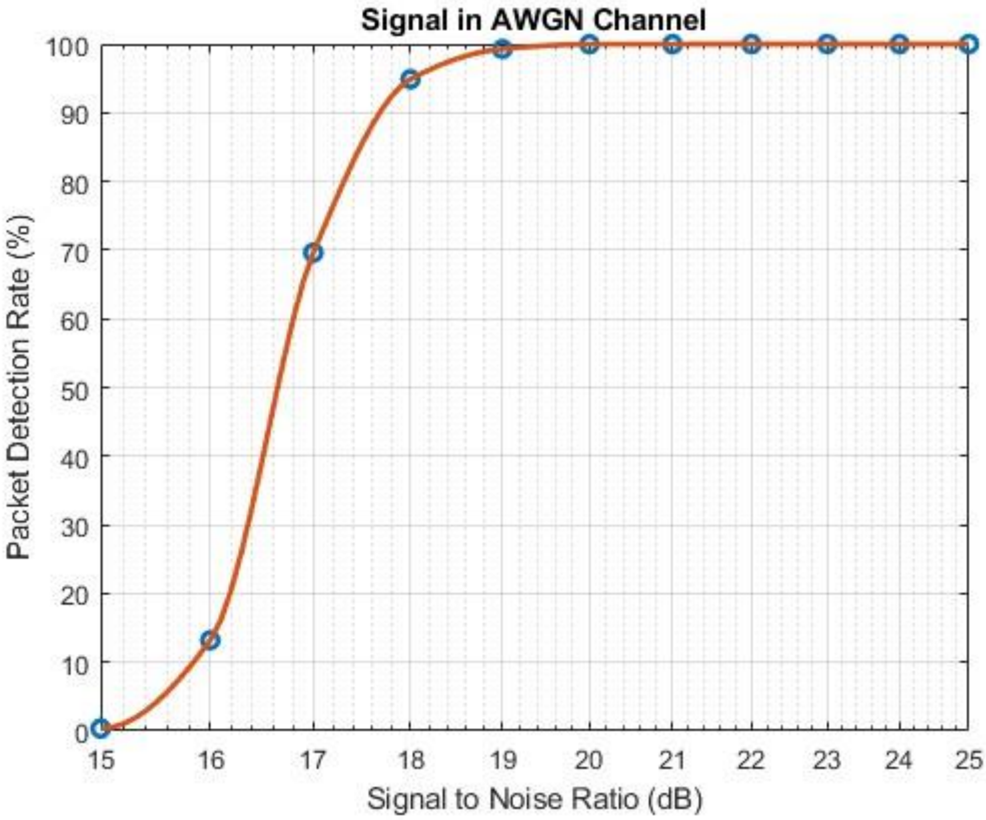

Figure 18 PDR-SNR in AWGN

- The figure below represents the Packet Decode Rate values at different SNR levels after passing the signal through the AWGN channel and adding IQ Imbalance. Until the SNR level reaches 19dB, the PDR remains at nearly 0%. When the SNR level exceeds 30dB, the PDR gradually increase to around 35% and 40% and reaches equilibrium. To observe the equilibrium point at the graph, SNR is set between 15dB and 40dB, increasing 1dB by 1dB.

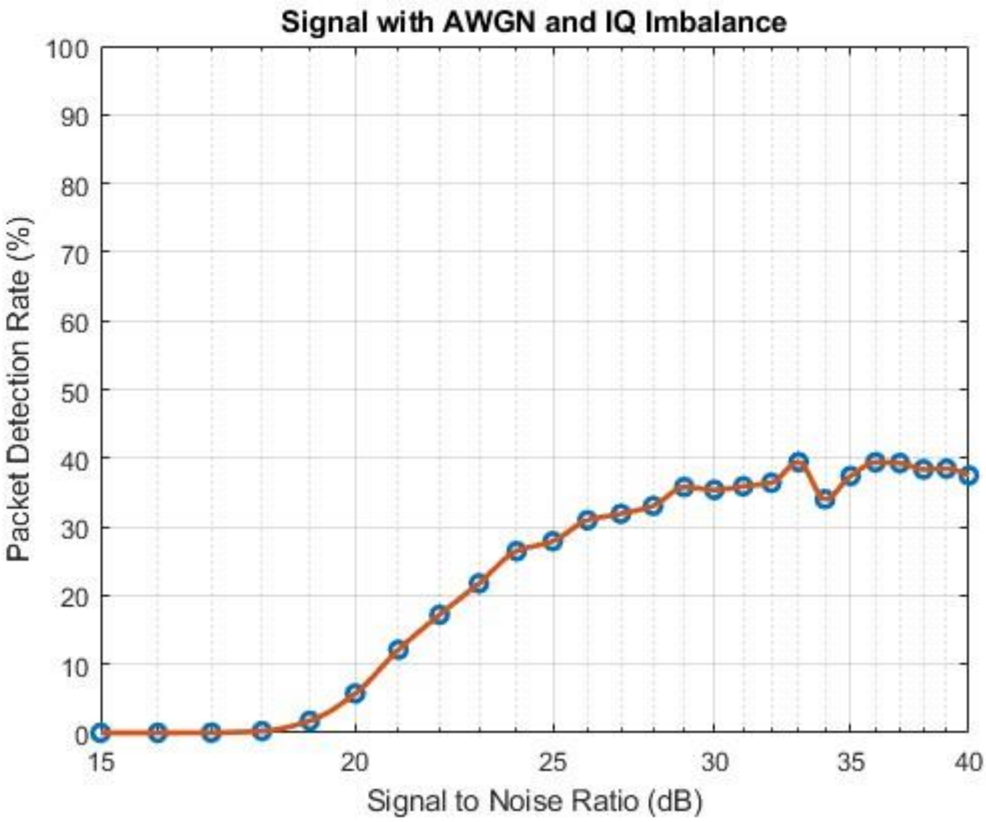

Figure 19 PDR-SNR in AWGN and IQ Imbalance

The figure below depicts the Packet Decode Rate values at different SNR levels after passing the signal through the AWGN channel and adding IQ Imbalance. After applying the Adaptive IQ Imbalance Compensation Algorithm to compensate the signal, the PDR starts at 0% when the SNR level is 15dB. Then, it quickly surpasses 60% by the time the SNR level reaches 19dB. Subsequently, the PDR exhibits a slower increase, reaching over 90% at 24dB and 25dB SNR levels. When compared to the previous graph without IQ Imbalance Compensation, it can be observed that the PDR has significantly improved.

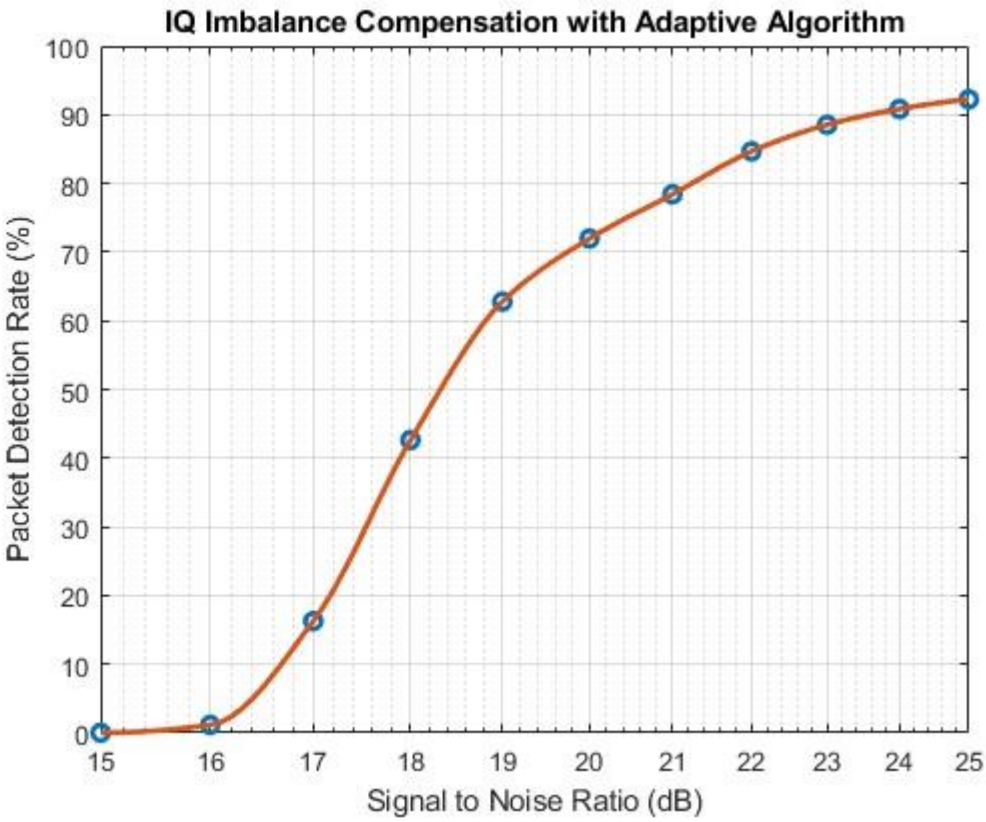

Figure 20 PDR-SNR in AWGN and IQ Imbalance

The figure below illustrates the Packet Decode Rate values at different SNR levels after passing the signal through the AWGN channel and introducing IQ Imbalance. After applying the Blind IQ Imbalance Compensation Algorithm to compensate the signal, the PDR starts at 0% when the SNR level is 15dB. It rapidly exceeds 70% by the time the SNR level reaches 20dB and then exhibits a slower increase, surpassing 90% at 23dB SNR level. When compared to the two previous graphs without IQ Imbalance Compensation, it can be concluded that the PDR has significantly improved.

Comparing it to the Adaptive Algorithm, it can be observed that the Adaptive Algorithm performs better at relatively low SNR levels (below 20dB), while the Blind Algorithm yields higher results after the 20dB SNR level, reaching over 90% PDR more quickly. The table below illustrates the differences at each SNR level:

Table 2 Comparison of Blind vs Adaptive Algorithm at PDR-SNR

	15dB	16dB	17dB	18dB	19dB	20dB	21dB	22dB	23dB	24dB	25dB
Blind	0.0%	0.4%	3.6%	14.4%	44.6%	69.7%	82.8%	88.1%	91.6%	93.2%	94.0%
Adaptive	0.0%	1.2%	16.3%	42.6%	62.8%	72.0%	78.4%	84.7%	88.5%	90.8%	92.2%

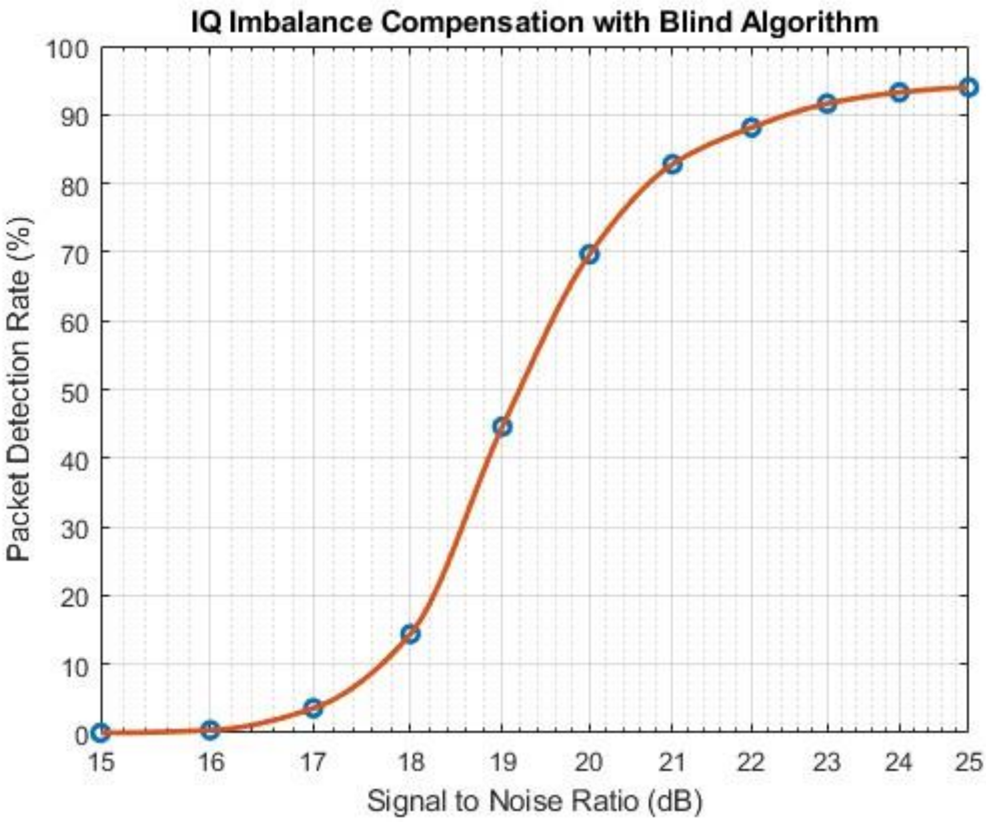

Figure 21 PDR-SNR IQ Imbalance Compensation with Blind Algorithm

4.3.2 Bit Error Rate Results

This section includes the Bit Error Rate - Signal to Noise Ratio (BER - SNR) graphs obtained from the simulations conducted as described in the previous section. To enhance the clarity of the results and show the variations more clearly, interpolation has been applied to the BER values.

The figure below illustrates the Bit Error Rate values at different SNR levels after passing the signal through only the AWGN channel. When the SNR level is 15dB, the BER is around 43%. As the SNR level increases to 18dB, the BER decreases to around 0% and remains around 0%. From this graph, it can be observed that the AWGN channel has a limited impact on the Bit Error Rate (BER) at the investigated SNR levels in the simulation.

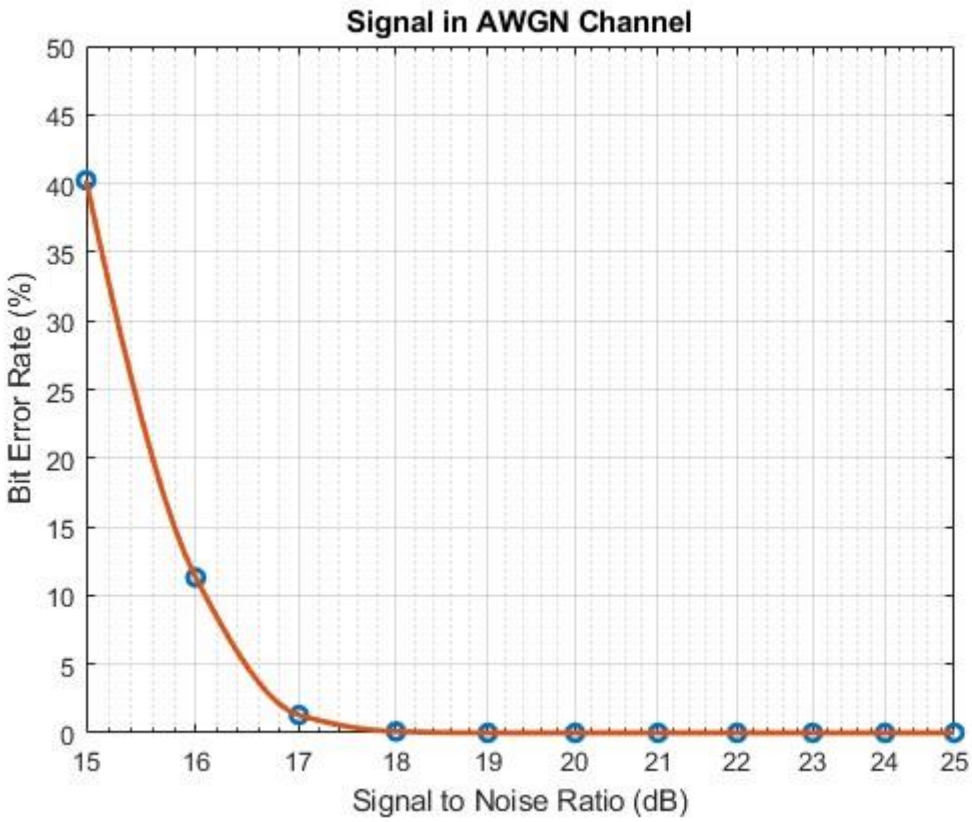

Figure 22 BER-SNR in AWGN

The figure below represents the Packet Decode Rate values at different SNR levels after passing the signal through the AWGN channel and adding IQ Imbalance. Unlike passing the signal only through the AWGN channel, it has been observed that adding IQ Imbalance has much more adverse effects on the Bit Error Rate (BER). At the 15dB SNR level, the BER is around 45% (it should be noted that 50% represents pure randomness, so 45% is significantly higher than an acceptable BER). The BER gradually decreases but only reaches around 28% at the 25dB SNR level and continues decreasing until about 35dB, where it reaches the equilibrium point that is 23%. To observe the equilibrium point at the graph, SNR is set between 15dB and 40dB, increasing 1dB by 1dB.

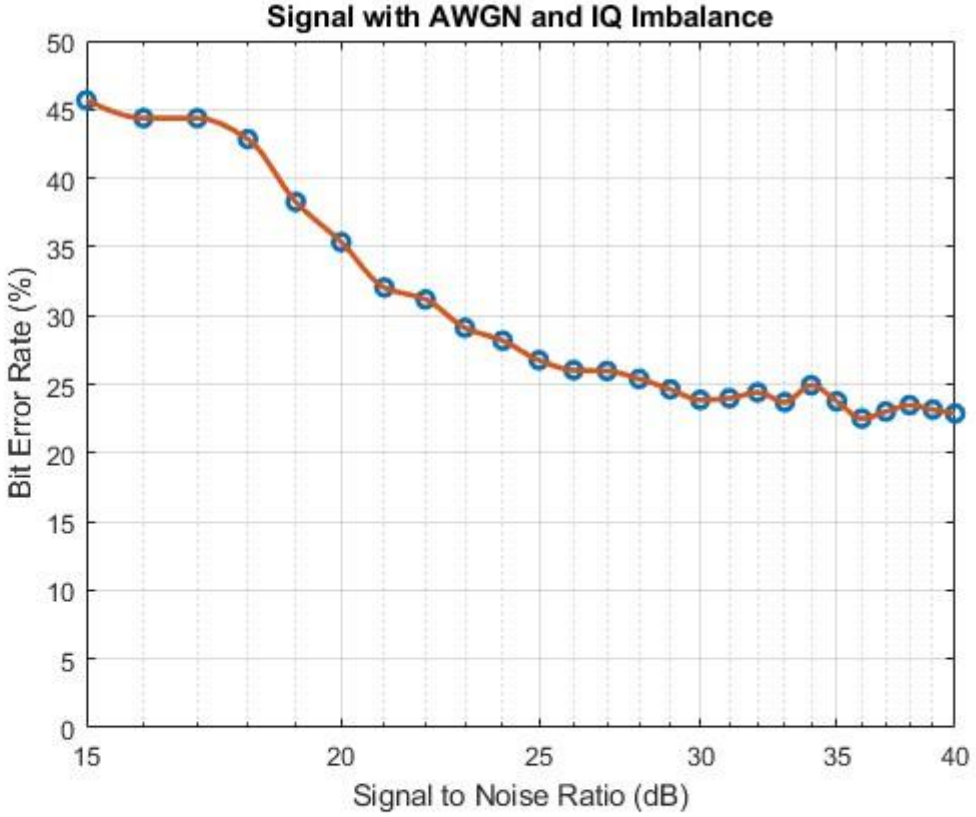

Figure 23 BER-SNR in AWGN and IQ Imbalance

The figure below shows the transmitted and received images of the signal after passing through the channel at a 20dB SNR level and being added to 2.5dB Amplitude Imbalance and 10-degree Phase Imbalance. As can be seen in the figure, the image is quite distorted.

Figure 24 TX and RX Image at 20dB SNR Level with 2.5dB Amplitude Imbalance and 10 degrees Phase Imbalance

The figure below depicts the Packet Decode Rate values at different SNR levels after passing the signal through the AWGN channel and adding IQ Imbalance. After applying the Adaptive IQ Imbalance Compensation Algorithm to compensate the signal, the BER, which is around 43% at the 15dB SNR level, rapidly drops below 10% until reaching 20dB, and then decreases gradually to around 2% at the 25dB level. At relatively low SNR levels like 16dB and 17dB, the compensation rate is lower due to the dominant effect of noise. However, at higher SNR levels, the adaptive algorithm achieves successful results. It is worth mentioning an important aspect of the Adaptive Algorithm: to calculate the required coefficients for compensation, a certain amount of data needs to be received by the receiver. This results in the inability to compensate for the initial packets, but after the first few packets, reliable correction can be performed. Considering that 23 packets were transmitted in the simulation, the initial failures contribute to the increase in the average BER. The impact of the number of packets on the Adaptive Algorithm will be further examined in the later stages of this chapter.

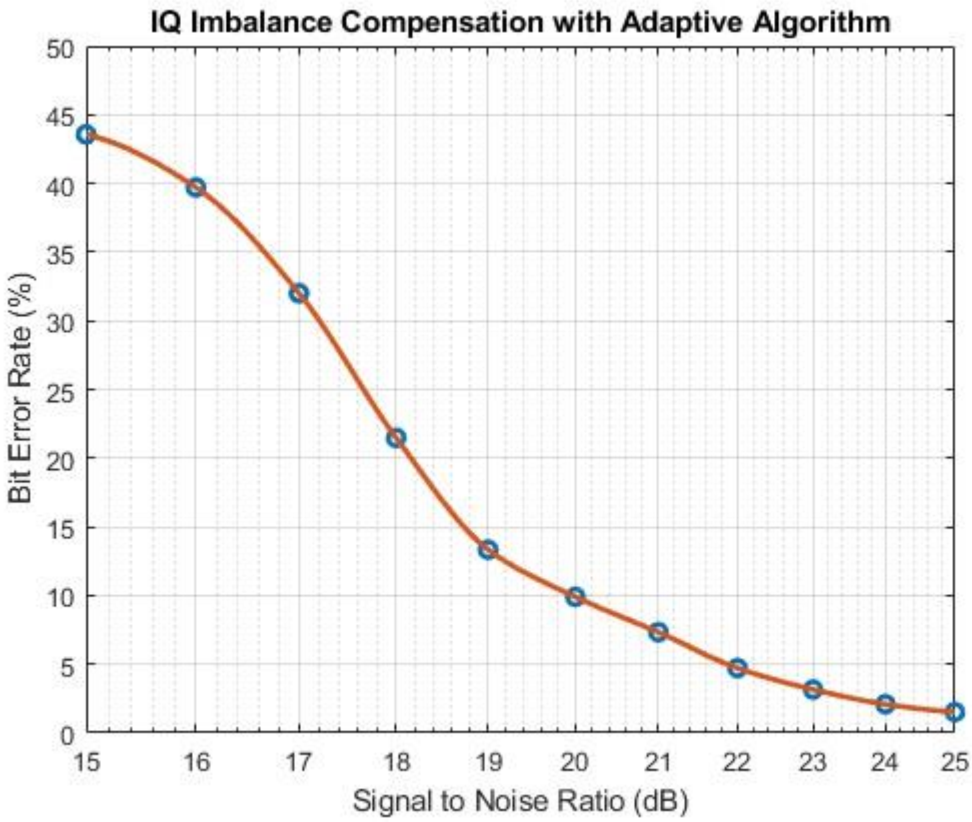

Figure 25 BER-SNR IQ Imbalance Compensation with Adaptive Algorithm

The figure below shows the transmitted and received images of the signal compensated with Adaptive Algorithm after passing through the channel at a 20dB SNR level and being added to 2.5dB Amplitude Imbalance and 10-degree Phase Imbalance. The corruption caused by the inability to calculate the coefficients in the initial packets can be observed on the left side of the figure.

Figure 26 TX and RX Image Compensated with Adaptive Algorithm.

The figure below illustrates the Packet Decode Rate values at different SNR levels after passing the signal through the AWGN channel and introducing IQ Imbalance. After applying the Blind IQ Imbalance Compensation Algorithm to compensate the signal, At the 15dB SNR level, the BER, like the Adaptive Algorithm, is around 43%. However, it rapidly decreases to below 5% at the 18dB SNR level and reaches around 1% at the 19dB SNR level. Compared to the Adaptive Algorithm, it shows significantly better at almost all SNR levels. The comparison of BER for each SNR value with the Adaptive Algorithm can be seen in the table below.

Table 3 Comparison of Blind vs Adaptive Algorithm of BER-SNR

	15dB	16dB	17dB	18dB	19dB	20dB	21dB	22dB	23dB	24dB	25dB
Blind	43.6%	40.3%	18.2%	3.6%	0.5%	0.3%	0.1%	0.1%	0.1%	0.1%	0.0%
Adaptive	43.6%	39.7%	32.0%	21.5%	13.3%	9.9%	7.3%	4.7%	3.2%	2.1%	1.5%

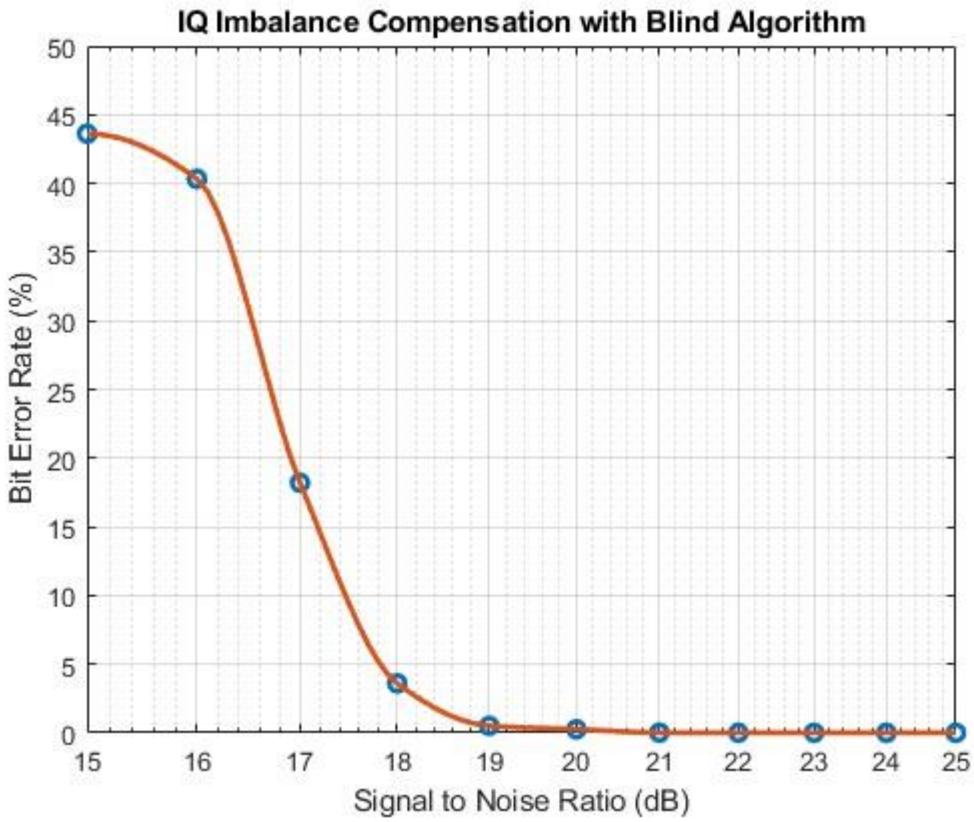

Figure 27 BER-SNR IQ Imbalance Compensation with Blind Algorithm

The figure below shows the transmitted and received images of the signal compensated with Blind Algorithm after passing through the channel at a 20dB SNR level and being added to 2.5dB Amplitude Imbalance and 10-degree Phase Imbalance.

Figure 28 TX and RX Image Compensated with Blind Algorithm

4.4 Adaptive IQ Imbalance Correction Algorithm: Packet Count Dependency

In previous topics, it is mentioned that the Adaptive IQ Imbalance correction algorithm failed to correct IQ imbalance in the initial packages since it could not calculate the necessary coefficients accurately. However, after calculating the coefficients, the success rate increased significantly. In this case, the Blind IQ imbalance correction algorithm has a higher success rate when the number of packages is low. Still, if longer packages are sent using the Adaptive IQ Imbalance correction algorithm, will the success rate exceed that of the Blind IQ imbalance correction algorithm? To answer this question, a new simulation environment is created and tested it as follows:

- SNR level: 40dB (in a low noise environment, with only amplitude and phase imbalances)
- Amplitude imbalance: 10dB level,
- Phase imbalance: 10 degrees,
- 23 packages will be sent,

As can be seen from the graph below, by performing Monte Carlo simulations and increasing the number of packets, it is observed that BER decreased significantly. Different numbers of packages, ranging from 23 to 1583, were selected for simulation (due to the large data size after approximately 1600 packages, the program was unable to complete the simulation). For each selected number of packages, the BER was calculated, and the results were observed.

BER, which was around 10% for 23 packages, dropped to about 0.1% for 1583 packages. The Blind IQ imbalance correction algorithm had a 0% BER rate at 40dB SNR, 10dB amplitude imbalance, and 10-degree phase imbalance levels. Thus, it can be concluded that the Adaptive IQ Imbalance correction algorithm is quite successful for long package transmissions.

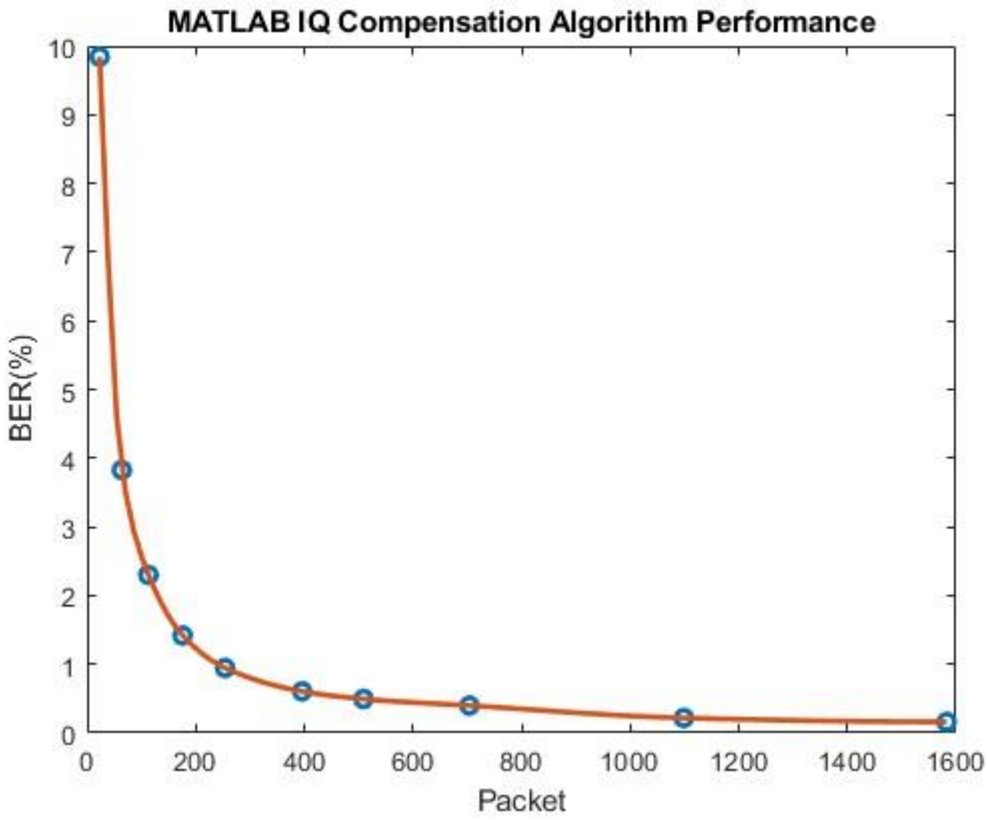

Figure 29 Adaptive Algorithm Performance Across Packet Count

Table 4 Imbalance Compensation Algorithm Performance Across Packet Count Table

Packet Count	**BER**
23	0.098452
64	0.038254
113	0.022993
176	0.014177
254	0.009455
396	0.006041
509	0.004927
704	0.003974
1099	0.002166
1583	0.001601

5 CONCLUSION AND RECOMMENDATIONS

This chapter of the thesis includes a comparison of the Monte Carlo simulations performed on the Adaptive IQ Imbalance Compensation Algorithm and the Blind IQ Imbalance Compensation Algorithm, which are the subject of the thesis, in the MATLAB environment. The opinions of the thesis team about the results, the future work that can be carried out regarding the topic, and the conclusion section are also included in this chapter.

5.1 Comparison of IQ Imbalance Compensation Algorithms

In this section, a comparison will be made between the Adaptive IQ Imbalance Compensation Algorithm and Blind IQ Imbalance Compensation Algorithm studied throughout the thesis and simulated using MATLAB.

Based on the work of the thesis team and the evaluation of the results, it has been observed that the Blind IQ Imbalance Compensation Algorithm produces similar results to the Adaptive IQ Imbalance Compensation Algorithm in terms of both BER and PDR comparisons. In fact, it has been observed that the Blind algorithm outperforms the Adaptive algorithm in terms of BER, primarily due to the absence of issues related to the initial packets where the coefficients cannot be corrected. In terms of PDR, the results are also comparable to the Adaptive Algorithm. Therefore, the Blind IQ Imbalance Compensation Algorithm is considered a successful algorithm for IQ Imbalance Compensation implementations.

5.2 Recommendations

This section includes the thoughts and recommendations of the thesis team on how the work carried out within the scope of the thesis can be advanced.

The aim of the thesis was to evaluate the usability of Moseley's Blind IQ Imbalance Compensation Algorithm in real systems and academic research, by comparing it with Antilla's Adaptive IQ Imbalance Compensation Algorithm, which is widely used in various applications including MATLAB. The results showed similarities between the two algorithms in the MATLAB simulation environment, indicating that Moseley's Blind Algorithm is a successful algorithm. However, since simulations conducted on a computer may not always reflect reality accurately, it would be beneficial to test this algorithm in a laboratory environment as well.

5.3 Conclusion

This section contains the thesis conclusion, which provides a summary of the work conducted throughout the research.

Throughout the thesis, frequency independent IQ Imbalance models and how this effect can be effectively compensated have been examined. Moseley's Blind IQ Imbalance Compensation Algorithm has been analyzed to demonstrate its usability in scientific research and real-world applications, and it has been tested through various simulations.

The algorithm's performance has been measured by comparing it with Antilla's Adaptive IQ Imbalance Compensation algorithm, which is widely used in various applications, including MATLAB. Both algorithms have been subjected to various simulations, and it has been observed that the results of the two algorithms are similar. As a result of these studies, it has been shown that Moseley's Blind IQ Imbalance Compensation algorithm can be confidently used for IQ Imbalance Compensation.

6 APPENDIX

6.1 Monte Carlo Simulations

Monte Carlo simulation is a computational technique that uses random sampling to simulate complex systems or processes that are difficult or impossible to analyze mathematically. It is named after the famous Monte Carlo Casino in Monaco, where gambling involves randomness and chance.

The basic idea behind Monte Carlo simulation is to generate random samples or events that represent the behavior of the system being studied, and then use statistical analysis to estimate the overall behavior of the system. The technique is widely used in various fields such as finance, engineering, physics, and chemistry.

7 References

[1] N. Moseley and C. Slump, "A low-complexity feed-forward I/Q imbalance compensation algorithm," *Computational Statistics & Data Analysis,* 2006.

[2] S. Al-Majmaie, "IQ Imbalance Compensation : Blind versus Pilot-Based Algorithms , using Different IQ Imbalance Models," 2014.

[3] L. Anttila, M. Valkama and M. Renfors, "Blind Compensation of Frequency-Selective I/Q Imbalances in Quadrature Radio Receivers: Circularity -Based Approach," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, Honolulu, 2007.

[4] A. Kiayani, L. Anttila, Y. Zou and M. Valkama, "Advanced Receiver Design for Mitigating Multiple RF Impairments in OFDM Systems: Algorithms and RF Measurements," *Journal of Electrical and Computer Engineering,* 2012.

[5] A. Schoonen, "IQ imbalance in OFDM Wireless," 2006.

[6] A. Aslan, "Düşük Maliyetli Alıcılarda RF Parmak İzi Oluşturma Yöntemine Dayalı Kablosuz Vericilerin Kimlik Tespiti," 2020.

[7] G. E. Sağır and S. Taşcıoğlu, "Evaluation of the impact of I/Q imbalance compensation on communication performance," *Communications Faculty of Sciences University of Ankara Series A2-A3 Physical Sciences and Engineering,* vol. 64, no. 2, pp. 66-80, 2022.