NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

A COMPARISION OF FREQUENCY OFFSET ESTIMATION METHODS IN ORTHOGONAL FREQUENCY DIVISION MULTIPLEXING (OFDM) SYSTEMS

by

Bulent KARAOGLU

December 2004

Thesis Advisor:	Roberto Cristi
Thesis Co-Advisor:	Murali Tummala

Approved for public release; distribution is unlimited

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE			Form Approved OMB No. 0704-0188
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503.			
1. AGENCY USE ONLY *(Leave blank)*	**2. REPORT DATE** December 2004	**3. REPORT TYPE AND DATES COVERED** Master's Thesis	
4. TITLE AND SUBTITLE: A comparision of frequency Offset Methods in Orthogonal Frequency Division Multiplexing (OFDM) Systems			**5. FUNDING NUMBERS**
6. AUTHOR(S) Bulent KARAOGLU			
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000			**8. PERFORMING ORGANIZATION REPORT NUMBER**
9. SPONSORING /MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A			**10. SPONSORING/MONITORING AGENCY REPORT NUMBER**
11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.			
12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution is unlimited			**12b. DISTRIBUTION CODE**

13. ABSTRACT (maximum 200 words)

OFDM is a modulation technique that achieves high data rates, increased bandwidth efficiency and robustness in multipath environments. However, OFDM has some disadvantages, such as sensitivity to channel fading, large peak to average ratio and sensitivity to frequency offset. The latter causes intercarrier interference (ICI) and a reduction in the amplitude of the desired subcarrier which results in loss of orthogonality. In this thesis, the effects of frequency offset are studied in terms of loss of orthogonality. A number of techniques for frequency offset estimation are presented and tested in computer simulations.

14. SUBJECT TERMS. OFDM (Orthogonal Frequency Division Multiplexing), Frequency Offset , Coarse Frequency Offset Estimation, Fine Frequency Offset Estimation, IFFT(Inverse Fast Fourier Transform), FFT, Cyclic Prefix, MATLAB , Preamble, OFDM Symbol			**15. NUMBER OF PAGES** 65
			16. PRICE CODE
17. SECURITY CLASSIFICATION OF REPORT Unclassified	**18. SECURITY CLASSIFICATION OF THIS PAGE** Unclassified	**19. SECURITY CLASSIFICATION OF ABSTRACT** Unclassified	**20. LIMITATION OF ABSTRACT** UL

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release; distribution is unlimited

A COMPARISION OF FREQUENCY OFFSET ESTIMATION METHODS IN ORTHOGONAL FREQUENCY DIVISION MULTIPLEXING (OFDM) SYSTEMS

Bulent KARAOGLU
Lieutenant Junior Grade, Turkish Navy
B.S., Turkish Naval Academy, 1999

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN ELECTRICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
December 2004

Author: Bulent KARAOGLU

Approved by: Roberto Cristi
Thesis Advisor

Murali Tummala
Co-Advisor

John P. Powers
Chairman, Department of Electrical and Computer Engineering

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

OFDM is a modulation technique that achieves high data rates, increased bandwidth efficiency and robustness in multipath environments. However, OFDM has some disadvantages, such as sensitivity to channel fading, large peak to average ratio and sensitivity to frequency offset. The latter causes intercarrier interference (ICI) and a reduction in the amplitude of the desired subcarrier that results in loss of orthogonality. In this thesis, the effects of frequency offset are studied in terms of loss of orthogonality. A number of techniques for frequency offset estimation are presented and tested in computer simulations.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF FIGURES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF TABLES

THIS PAGE INTENTIONALLY LEFT BLANK

ACKNOWLEDGMENTS

First of all, I want to thank my thesis advisors Prof. Roberto Cristi and Prof. Murali Tummala for their help and understanding. I also want to thank to my family and my big brothers Erol Kucuk and Ozer Eroglu, my friends Hakan Erdagi, Bahadir Sengun, Tekin Kirimsal, Ergun Karamik, Enis Mula, Omer Aktas, Yilmaz Bayar, Serdar Yilmaz, Alicia Reade, Maggie Valdes, Juliana Hernandez, Alvaro Naverreta and my hero Fernando Carlos Redondo for their endless support to me. This thesis is dedicated to Miss Shirley Laverada and Fenerbahce SK, Istanbul football club.

THIS PAGE INTENTIONALLY LEFT BLANK

EXECUTIVE SUMMARY

High data rate transmission is one of the major challenges in modern communications, and it is very important for both in military and commercial applications. OFDM is used as part of the IEEE 802.11a standard in wireless local area networks to provide high data rate transmission. Because it is reliable and has improved bandwidth efficiency with respect to conventional Frequency Division Multiplexing (FDM) systems, OFDM is seen as the future technology for communications in wireless local area systems.

The main disadvantages of OFDM are sensitivity to Intersymbol Interference (ISI) and Intercarrier Interference (ICI). One of the main reasons for ICI is loss of synchronization caused by frequency offset between oscillators at the transmitter and the receiver. This causes the carriers to lose orthogonality, so they cannot be completely separated at the receiver. As a consequence, ICI lowers the signal-to-noise ratio (SNR) and increases the error probability.

This thesis investigated the effects of frequency offset in OFDM demodulation and how we can estimate it so that we can compensate for its effects.

In particular, we studied the effects of ICI through computer simulations. We show that an error in frequency synchronization as small as 0.5% of the frequency spacing can cause significant ICI, thereby making it impossible to demodulate the signal. In order to cope with this problem, data-driven, blind and semiblind techniques were analyzed.

The data driven technique introduced by Moose relies on a particular data structure, and it is the most effective. A blind technique introduced by Cimini is less effective, but it does not require any particular data structure. Finally, a semiblind technique based on the cyclic prefix seems to be very effective in tracking frequency drifts during data transmission. Computer simulations under different noise conditions support our arguments.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

Orthogonal frequency division multiplexing (OFDM) is a transmission method that can achieve high data rates by multicarrier modulation. The wireless local area network standard, IEEE 802.11a, issued by the Institute of Electrical and Electronics Engineers (IEEE), is based on OFDM. Furthermore, OFDM exhibits much better bandwidth efficiency than classical frequency division multiplexing (FDM) provided that the orthogonality of the carriers is preserved. One of the biggest problems with OFDM systems is their sensitivity to frequency offset between oscillators at the transmitter and the receiver which causes ICI due to loss of orthogonality.

A. OBJECTIVE

A very important aspect in OFDM is time and frequency synchronization. In particular, frequency synchronization is the basis of the orthogonality between frequencies. Loss in frequency synchronization is caused by a number of issues. It can be caused by Doppler shift due to relative motion between the transmitter and the receiver. This is particularly severe when each OFDM frame has a large number of frequencies closely spaced next to each other.

In systems based on the IEEE 802.11a standard, the Doppler effects are negligible when compared to the frequency spacing of more than 300 kHz. What is more important in this situation is the frequency error caused by imperfections in oscillators at the modulator and the demodulator. These frequency errors cause a frequency offset comparable to the frequency spacing, thus lowering the overall SNR.

The main goals of this thesis were to understand the effects of frequency offset on OFDM systems and to present frequency offset estimation techniques so that we can correct for their effects. The performance of each technique is compared under various conditions.

B. RELATED WORK

References [1-4] are fundamentaled to the developments presented in this thesis, the discussions on OFDM basics and the effects of multipath channels on OFDM signals. The MATLAB code in [1] provided a starting point in developing the frequency offset

estimation techniques. Timing synchronization has been addressed in [4] while [5] and [6] present techniques for frequency offset estimations.

C. THESIS ORGANIZATION

Chapter II presents a general description and the basics of OFDM . Also, the OFDM block diagram is explained briefly in Chapter II. In Chapter III, the frequency offset is defined. Several simulations are conducted, and the effects of frequency offset are examined under different channel conditions. Chapter IV investigates two frequency offset methods: a data-driven (Moose) method and a blind (Cimini) method. The data-driven method were chosen as a data-aided frequency offset estimation technique, and the Cimini method were chosen as a blind-estimation frequency-offset estimation technique. Both of these methods are analyzed, and a comparison of the two techniques is provided. In Chapter V, the effects of the preamble and the cyclic prefix are examined, and related simulation results are shown. Finally, Chapter VI presents the conclusions reached in this thesis. Appendix A presents the Matlab code used for simulating the implementation of signal-to-noise ratio (SNR) degradation caused by frequency offset. Appendix B and C presents the Matlab code used for simulating the effects of the frequency offset. Appendix D presents the Matlab code used for simulating the use of the cyclic prefixes and preambles.

II. BASICS OF ORTHOGONAL FREQUENCY DIVISION MULTIPLEXING (OFDM)

In this chapter, the basics of orthogonal frequency division multiplexing (OFDM) are discussed. In particular, a general description and a brief history of OFDM are presented. Also, the OFDM block diagram is explained briefly.

A. ORTHOGONAL FREQUENCY DIVISION MULTIPLEXING (OFDM)

The rapid growth in the requirements for information in every branch of life necessitates improvements in high data rate transmission. This can be accomplished by using Orthogonal Frequency Division Multiplexing (OFDM).

OFDM is a multicarrier transmission technique used for wireless communication in which a high-rate data sequence is split into multiple low rate blocks, each one modulated onto separate sub carriers. The IEEE 802.11a standard incorporates OFDM as a standard because of its robustness under multipath fading conditions [1].

A brief survey of the history of OFDM reveals that it was first introduced in the 1950s [3]. In the 1960s, OFDM techniques were used in several high-frequency military systems. In the 1980s, it was studied for high-speed modems, digital mobile communications and high-density recording. Finally in the 1990s, OFDM was exploited for wideband data communication over mobile radio FM channels, high-bit-rate digital subscriber lines (HDSL at 1.6 Mbps), asymmetric digital subscriber lines (ADSL up to 6 Mbps), very-high-speed digital subscriber lines (VDSL at 100 Mbps), digital audio broadcasting (DAB), and high-definition television (HDTV) terrestrial broadcasting [1].

The efficient use of the Fast Fourier Transform (FFT) in OFDM systems was one of the main factors that contributed to its rapid development and widespread use. Consequently, OFDM was accepted as a wireless local area network (WLAN) standard in September 1999 by the IEEE. The modulation parameters of OFDM for IEEE 802.11a are listed in Table 1, and the rate dependent parameters are listed in Table 2 [1].

PARAMETER	VALUE
Number of data subcarriers	48
Number of pilot subcarriers	4
Total number of subcarriers	52
Subcarrier frequency spacing	0.3125 MHz
IFFT/FFT period	$3.2 \left(1/\Delta f\right) \mu s$
Preamble duration	$16 \mu s$
Signal duration BPSK_OFDM symbol	$4 \left(T_{GI} + T_{FFT}\right) \mu s$
Guard interval (GI) duration	$0.8 \left(T_{FFT}/4\right) \mu s$
Training symbol GI duration	$0.4 \left(T_{FFT}/2\right) \mu s$
Symbol interval	$4 \left(T_{GI} + T_{FFT}\right) \mu s$
Short training duration	$8 \left(10T_{FFT}/4\right) \mu s$
Long training sequence duration	$8 \left(T_{GI} + 2T_{FFT}\right) \mu s$

Table 1. OFDM Timing Related Parameters

Data rate (Mbps)	Modulation	Coding Rate (R)	Coded bits per subcarrier $\left(N_{BPSC}\right)$	Coded bits per OFDM symbol $\left(N_{CBPS}\right)$	Data bits per OFDM symbol $\left(N_{DBPS}\right)$
TYN6	BPSK	1/2	1	48	24
9	BPSK	3/4	1	48	36
12	QPSK	1/2	2	96	48
18	QPSK	3/4	2	96	72
24	16-QAM	1/2	4	192	96
36	16-QAM	3/4	4	192	144
48	64-QAM	2/3	6	288	192
54	64-QAM	3/4	6	288	216

Table 2. OFDM Rate Dependent Parameters

In OFDM, a frequency-selective channel is subdivided into narrower flat fading channels. Although the frequency responses of the channels overlap with each other as shown in Figure 1, the impulse responses are orthogonal at the carriers, because the nulls of the each impulse response coincides with the maximum values of another impulse response and thus the channels can be separated [1].

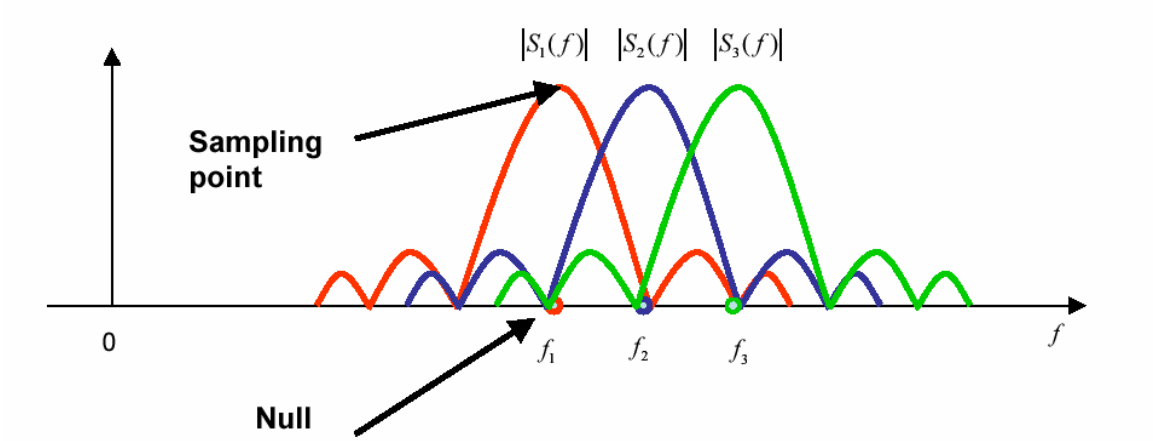


Figure 1. OFDM and the orthogonality principle (From Ref. 4.)

B. OFDM IMPLEMENTATION

This section explains the basic OFDM implementation. In OFDM the data are transmitted in blocks of length N. The $n-$th data block $\{X_n[0]....X_n[N-1]\}$ is transformed into the signal block $\{x_n[0]....x_n[N-1]\}$ by the IFFT as given by

$$x_n[l] = \frac{1}{N}\sum_{k=o}^{N-1} X_n[k] e^{jk\frac{2\pi}{N}l}, \qquad l = 0,...,N-1. \tag{1}$$

Each frequency $2\pi k/N$, $k = 0,...,N-1$ represents a carrier. In IEEE 802.11a, only 52 subcarriers are used. The data come from a constellation as depicted in Figure 2, which accommodates a number of bits in each frequency symbol transmitted $X_n[k]$.

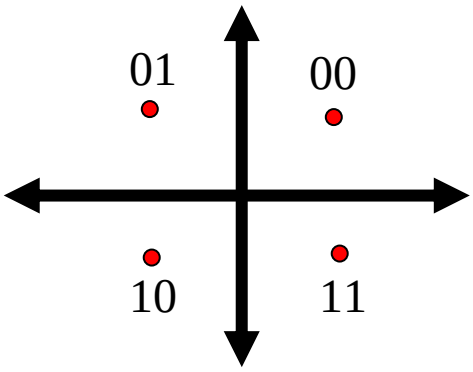


Figure 2. A typical QPSK Constellation

A basic OFDM implementation scheme is shown in Figure 3. Data at each sub-carrier (X_m) are input into the inverse fast Fourier transform (IFFT) to be converted to time-domain data (x_m) and after parallel-to-serial conversion (P/S), a cyclic prefix is added to prevent intersymbol interference (ISI). At the receiver, the cyclic prefix is removed, because it contains no information symbols. After the serial-to-parallel (S/P) conversion, the received data in the time domain (y_m) are converted to the frequency domain (Y_m) using the fast Fourier transform (FFT) algorithm.

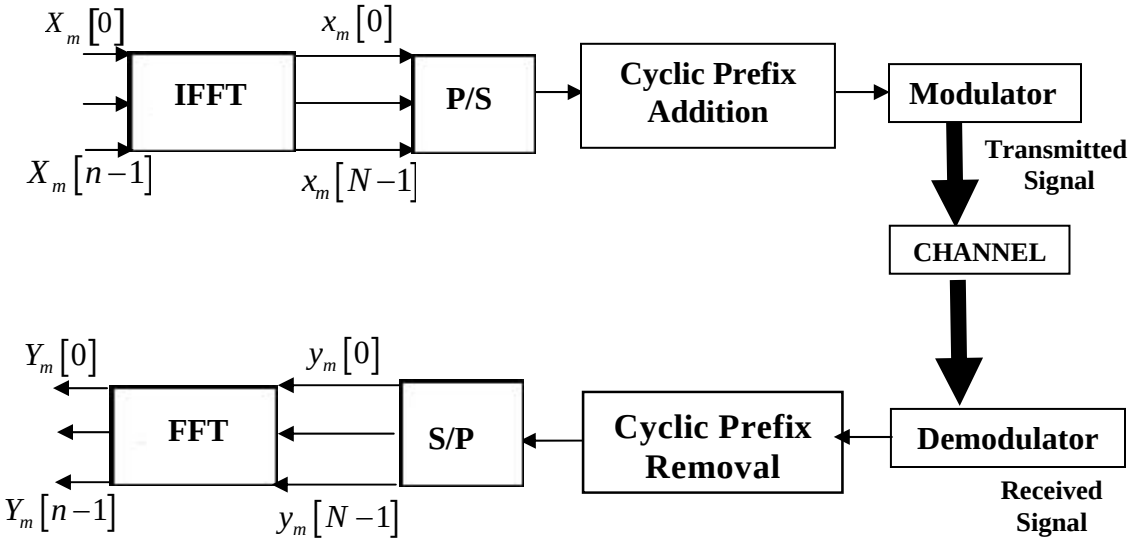
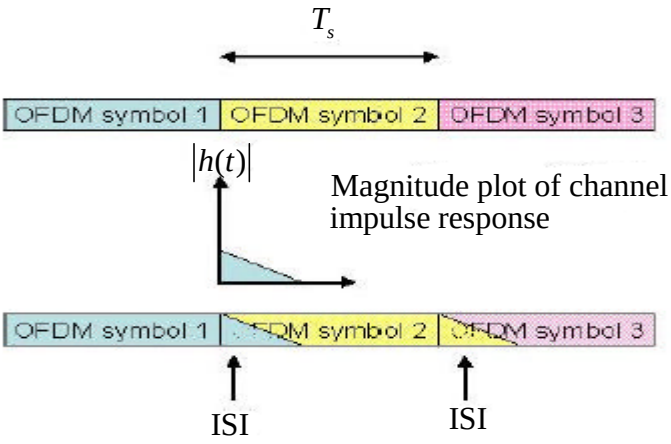
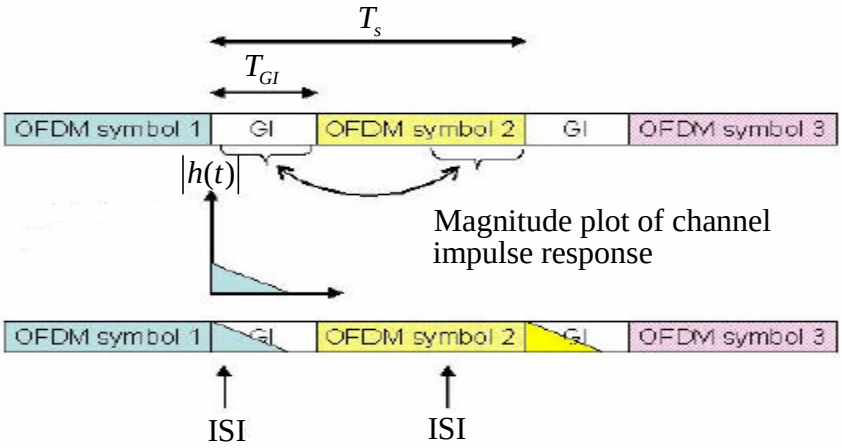


Figure 3. OFDM Implementation

The addition of a guard interval at the beginning of each frame avoids Inter Block Interference (IBI). The effect is shown in Figure 4 in which the channel impulse response *h(t)* causes the data from one frame to interfere with the following frame. In Figure 4a, intersymbol interference (ISI) between OFDM symbols causes the data loss because there is no guard interval (GI). When we use the GI, ISI between OFDM symbols does not cause the data loss, because the GI contains no information symbols; this case is shown in Figure 4b. Here, T_s represents the symbol duration and T_{GI} represents the guard interval duration in Figure 4.



(a) OFDM Symbols without Guard Interval



(b) OFDM Symbols with Guard Interval

Figure 4. Effects of Guard Interval (From Ref. 1.)

The guard interval is discarded at the receiver before demodulation. Also, the repetition of the tail of the data, the cyclic prefix, helps in time and frequency synchronization and will be discussed in Chapter V.

At the OFDM receiver, the inverse of the process that was used at the transmitter is performed. After removing the guard interval, the time-domain data are converted to the frequency domain using the FFT.

The characteristics of the channel significantly affect the data transmitted. The noise and the other distorting parameters that exist in the channel make it difficult to correct and decode the received data.

The basics of the OFDM, and the OFDM block diagram have been explained briefly in this chapter. The following chapter examines the frequency offset and synchronization.

III. FREQUENCY OFFSET AND FREQUENCY SYNCHRONIZATION

In this chapter, the degradation due to frequency offset and some of the techniques to obtain frequency synchronization are discussed. In particular, the effect of an error in frequency synchronization on the demodulated signal is considered, and an expression for degradation in terms of Signal-to-Noise Ratio (SNR) is described.

A. EFFECTS OF FREQUENCY OFFSET ON OFDM SIGNALS

Frequency offset comes from a number of sources, such as Doppler shift or frequency drifts in the modulator and the demodulator oscillators. The first source of error arises when there is relative motion between transmitter and receiver. In this case, the frequency shift is given by

$$\Delta f = \frac{v}{c} f_C \qquad (2)$$

where v is the relative velocity, c is the speed of light, and f_c is the carrier frequency. Compared to the frequency spacing, this shift is negligible. For example, with a carrier frequency of $f_C = 5$ GHz and a velocity of 100 km/h, the offset value is $\Delta f = 1.6$ kHz, which is relatively insignificant compared to the carrier spacing of 312.5 kHz.

The other source of frequency offset is due to frequency errors in the oscillators. The IEEE 802.11a standard requires the oscillators to have frequency errors within 20 ppm (or $\pm 20 \times 10^{-6}$). For a carrier of 5 GHz, this means a maximum frequency error of

$$|\Delta f_{MAX}| = 2 \times 20 \times 10^{-6} \times 5 \times 10^{9} = 200 \text{ kHz} \qquad (3)$$

where the factor "2" accounts for the transmitter and receiver having errors with opposite signs. This error is relatively large compared to the frequency spacing of the carrier.

OFDM systems are more sensitive to frequency errors than single-carrier modulations (SCM) systems. In OFDM systems, a frequency offset destroys orthogonality between carriers and introduces inter carrier interference (ICI), which is not the case in SCM systems. Conversely, single-carrier systems are more sensitive to timing offset errors while OFDM generally exhibits good performance in the presence of timing errors.

Before an OFDM receiver demodulates the subcarriers, it performs two synchronization tasks. First, it determines the symbol boundaries and the optimal timing instants in order to minimize both the ICI and the intersybol interference (ISI). Second, it tries to estimate and correct frequency errors [1].

In practice, the frequency, which is the time derivative of the phase, is never perfectly constant, thereby causing ICI in OFDM receivers [1]. One of the destructive effects of frequency offset is loss of orthogonality. The loss of orthogonality causes the ICI as shown in Figure 5. The areas, colored with yellow, show the ICI. When the centers of adjacent subcarriers are shifted because of the frequency offset, the adjacent subcarriers nulls are also shifted from the center of the other subcarrier. The received signal contains samples from this shifted subcarrier, leading to ICI [4].

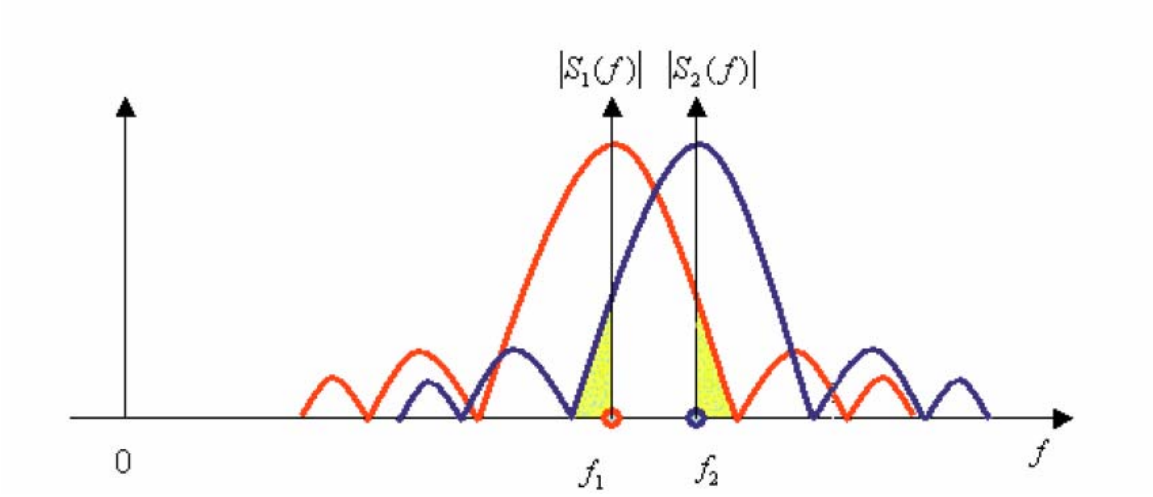


Figure 5. Illustration of ICI (From Ref. 4.)

The destructive effects of the frequency offset can be corrected by estimating the frequency offset itself and applying proper correction. This calls for the development of a frequency synchronization algorithm. Due to the increasing importance of OFDM demodulation in wireless and uncertain channels, a number of techniques have been developed.

Typically, three types of algorithms are used for frequency synchronization: algorithms that use pilot tones for estimation (data-aided), algorithms that process the data at

the receiver (blind), and algorithms that use the cyclic prefix for estimation [5]. Among these algorithms, blind techniques are attractive because they do not waste bandwidth to transmit pilot tones [4]. However, they use less information at the expense of added complexity and degraded performance [4].

The general approaches to the problem of synchronization consist of a number of steps, including frame detection, carrier frequency offset and sampling error correction [7]. Frame detection is used to determine the symbol boundary needed for correct demodulation. Within each frame, the carrier frequency offset between the transmitter and the receiver causes an unknown phase shift factor [7].

B. FREQUENCY OFFSET AND INTERCARRIER INTERFERENCE

According to [8], the degradation of the SNR, D_{freq}, caused by the frequency offset, is approximated as

$$D_{freq} \cong \frac{10}{3\ln 10}(\pi \Delta f T)^2 \frac{E_b}{N_0} \tag{4}$$

where Δf is the frequency offset, T is the symbol duration in seconds , E_b is the energy per bit of the OFDM signal and N_0 is the one-sided noise power spectrum density (PSD) . The frequency offset has an effect like noise and it degrades the signal-to-noise ratio (SNR), where SNR is the E_b/N_0 ratio. Figure 6 shows the calculated degradation of the SNR due to the frequency offset. For smaller SNR values, the degradation is less than for bigger SNR values as shown in Figure 6. The Matlab code, used to implement (4) is presented in Appendix A.

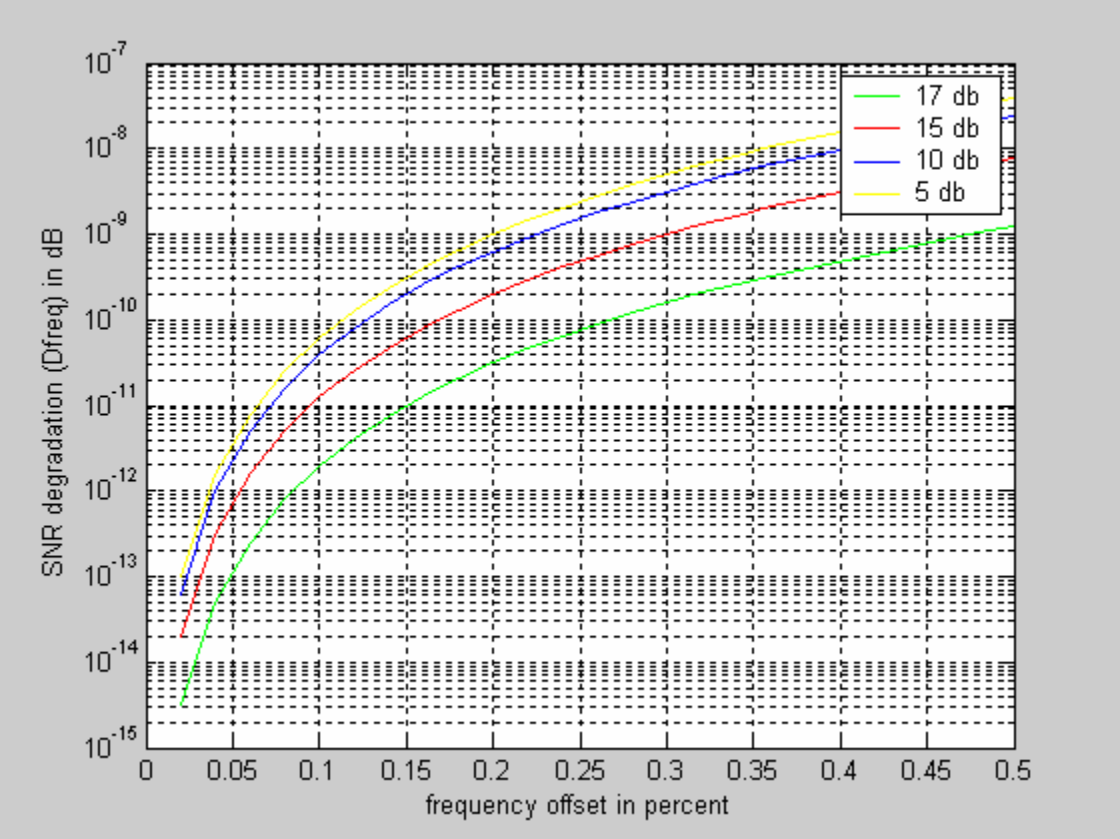


Figure 6. SNR degradation of frequency offset for different $\frac{E_b}{N_o}$ values (5, 10, 15, 17 dB)

In order to study the degradation in OFDM systems and its effects on the received signal, a number of computer simulations were run, and the results are shown below. In the first simulation, the received signal with no frequency offset is examined. In this case, the data were sent by two of the carriers as depicted in Figure 7. In Figure 7, arrows represent the subcarriers which have data and the dots represent the subcarriers which are set to zero (no data) . We generate 512 random QPSK signals as data. We send data using only two of the subcarriers, and the other subcarriers have no data.

Figure 7. Data sent using two of the subcarriers with no frequency offset

The Matlab code, used for simulating the case shown in Figure 7 is presented in Appendix B. Figure 8 shows, under these ideal conditions, the received signal constellation in the absence of frequency offset. Since there is no frequency offset or noise, it is seen that there is no ICI and no interference between the data and the other zeros

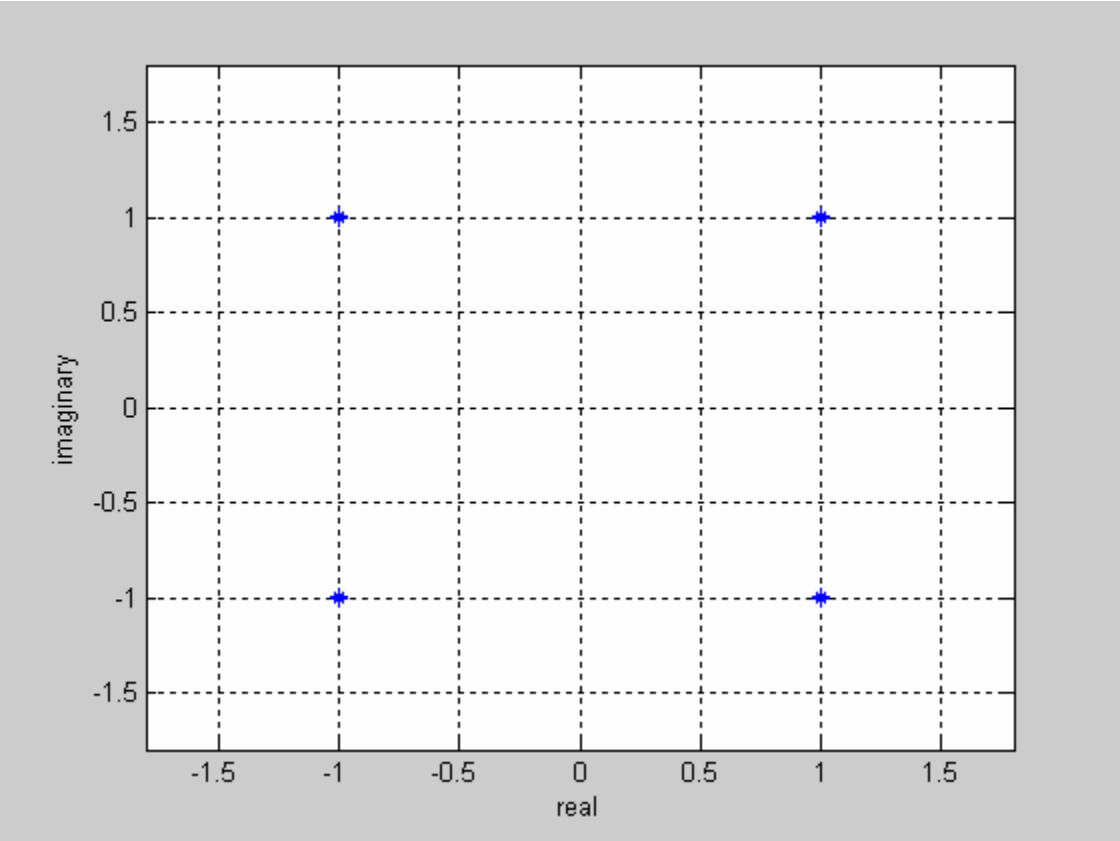


Figure 8. Received signal constellation with 0% frequency offset

When frequency offset is introduced in the carrier, its effects are observed in terms of ICI. The result with 0.3% frequency offset is shown in Figure 9. In particular, we can see that the signal from neighboring carriers causes interference and we have a distorted signal constellation at the receiver.

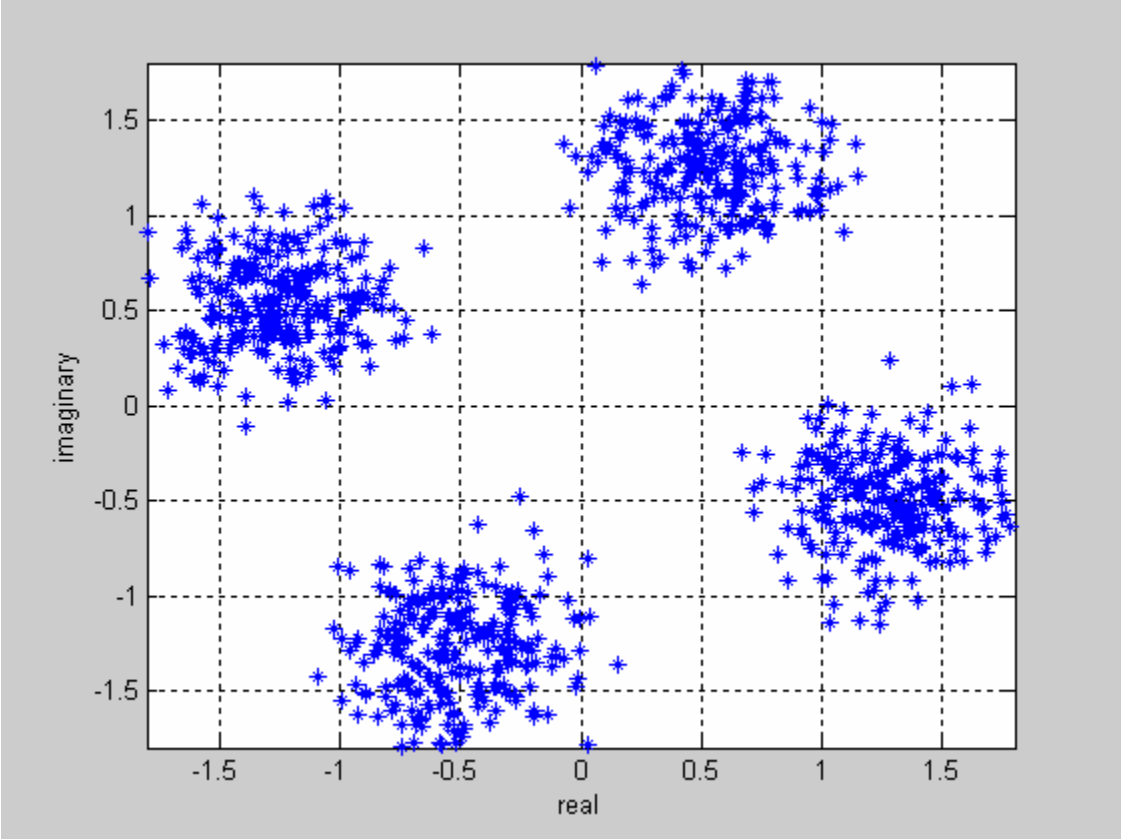


Figure 9. Received signal constellation with 0.3% frequency offset

When the frequency offset is increased, the simulation results reveal that the distortion in the received signal is increased. In Figure 10, the received signal for 0.5% frequency offset value is shown. When compared to Figure 9, it can be seen that the received signal with 0.5% frequency offset value is more distorted than the received signal with 0.3% frequency offset value.

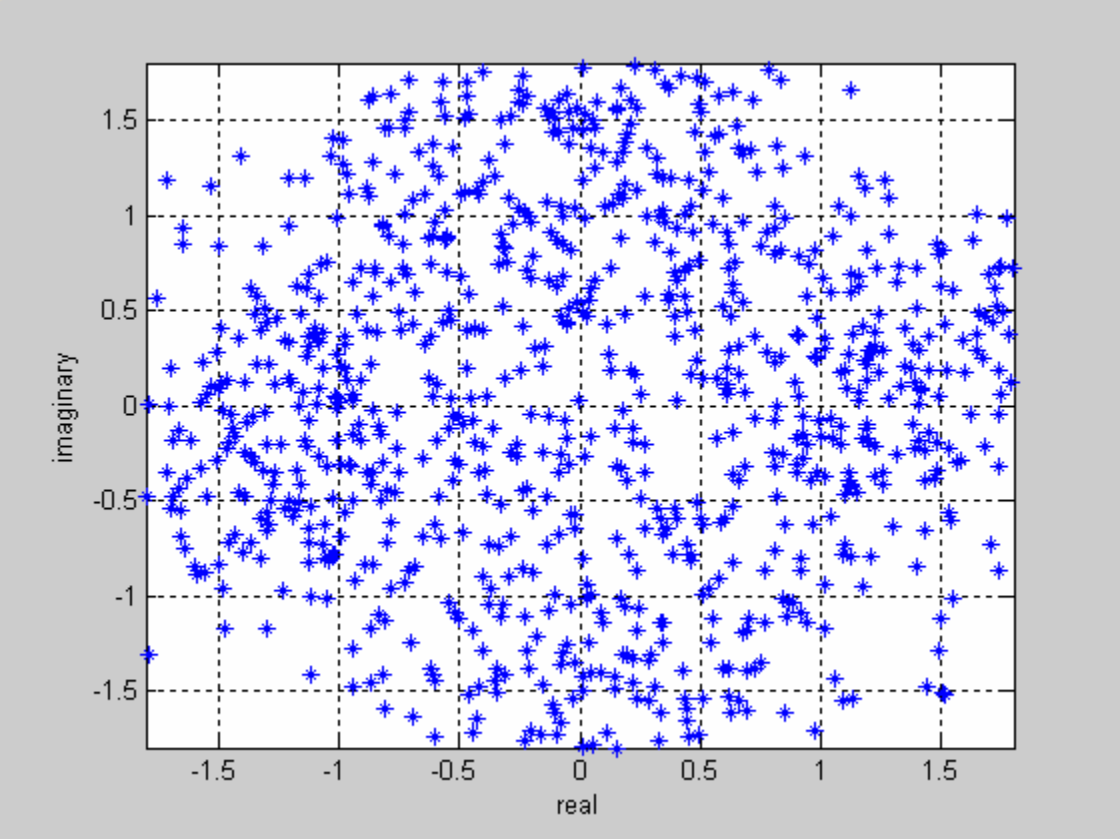


Figure 10. Received signal constellation with 0.5% frequency offset

The effects of the frequency offset can also be observed in a model that is depicted in Figure 11. In this model, data are sent with every subcarrier, except one, which is set to zero. In Figure 11 all the arrows represent data input and the dot represents the subcarrier that is set to zero.

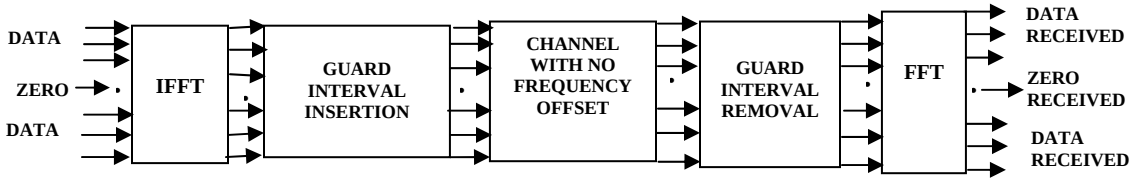


Figure 11. Data sent using every subcarrier, except one, with no frequency offset

The effects of frequency offset using the model depicted in Figure 11 are shown in Figure 12 and 13. The Matlab code used for simulating the case shown in Figure 11 is presented in Appendix C. If we have the frequency offset in the channel, we cannot re-

ceive a zero (no data) at the subcarrier that was set to zero shown in Figure 11. Figure 12 shows the effect of ICI due to frequency offset on the subcarrier with zero data from all other subcarriers. In the ideal case of no frequency offset, the demodulated value should be zero for the whole time. When frequency offset is present, the effect is like random noise which increases with the frequency offset. As shown in Figure 13, the effect of ICI increases considerably when the frequency offset is on the order of 0.4% - 0.6%.

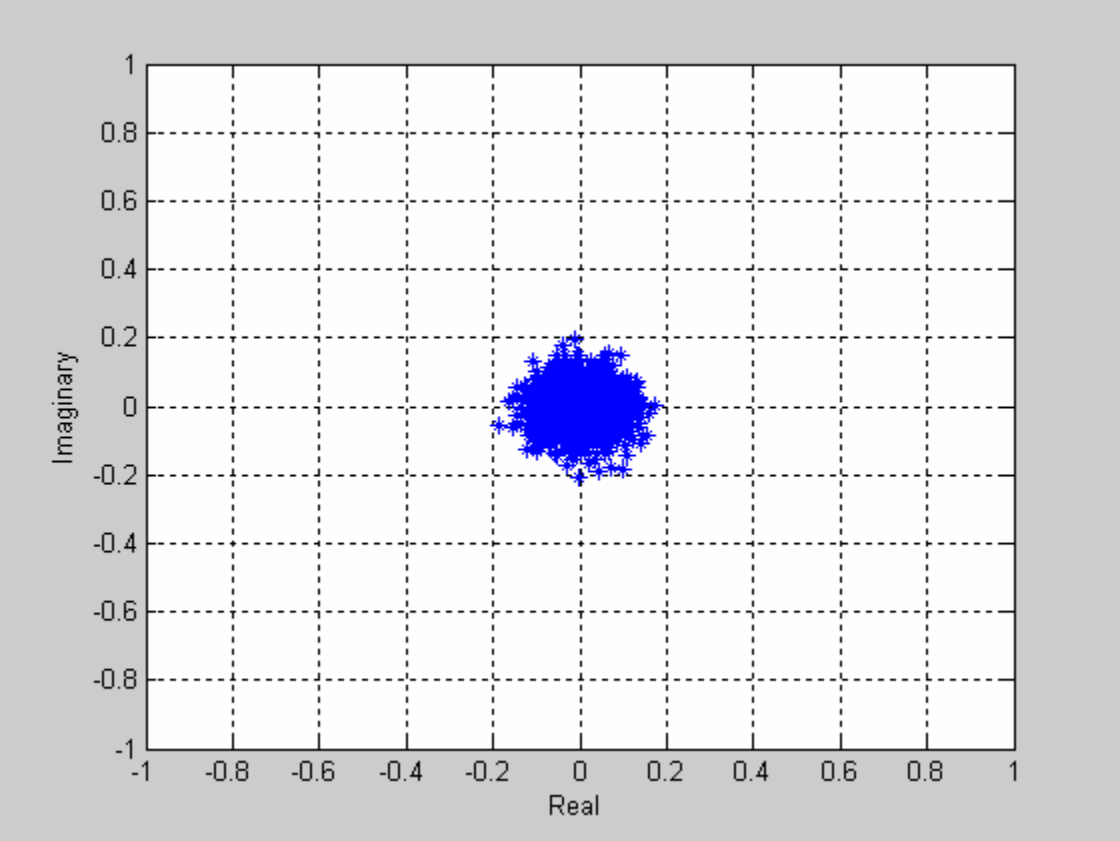


Figure 12. Received signal at the zero subcarrier with 0.1% frequency offset

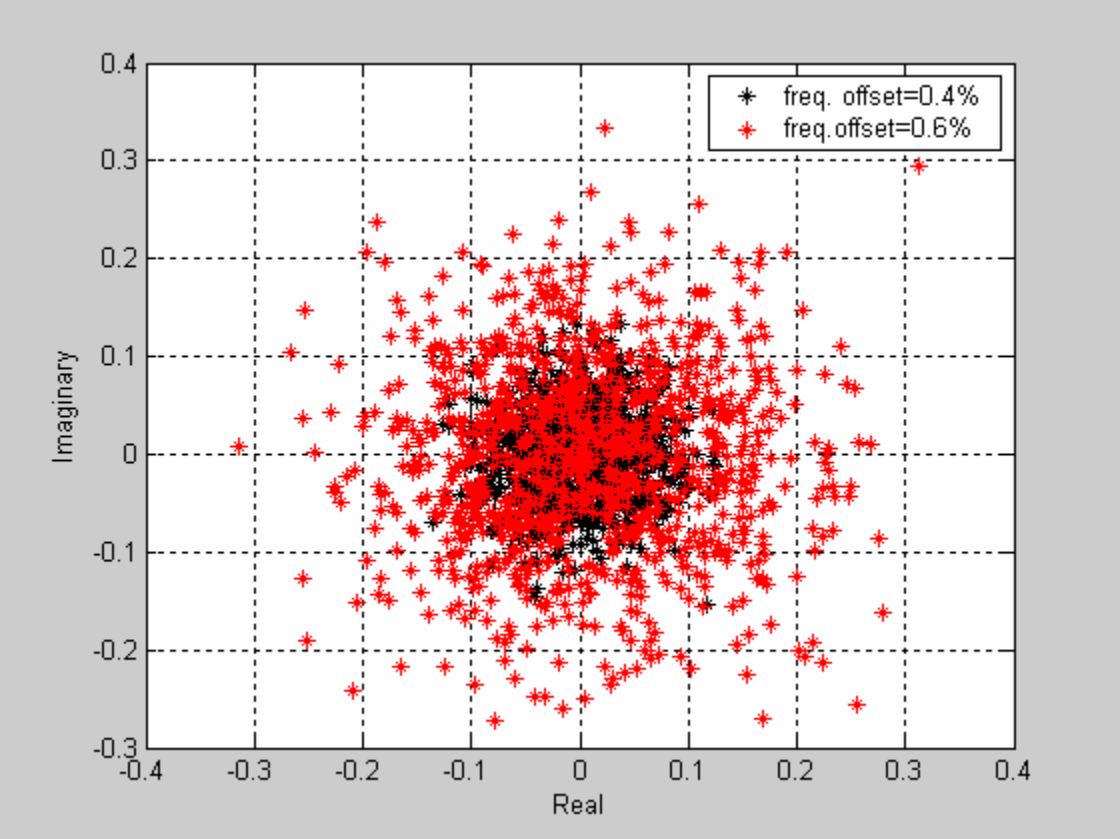


Figure 13. Received signal at the zero subcarrier with 0.4% and 0.6% frequency offset

When we compare the results in Figures 12 and 13, we can see that when we increase the frequency offset value, the received signal is distorted more and for the frequency offset values bigger than 0.6%, the received data are unreadable.

The frequency offset effects have been studied in this chapter. In the following chapters, the frequency offset estimation techniques are examined.

THIS PAGE INTENTIONALLY LEFT BLANK

IV. DATA-DRIVEN AND BLIND FREQUENCY OFFSET ESTIMATION TECHNIQUES

In this chapter, data-driven and blind frequency offset estimation techniques are examined and compared to one another. The first technique considered is a data-driven technique that was proposed by Moose [5]. The second technique examined is a blind estimation technique presented by Cimini [6]. Both methods are analyzed and a comparison is made between them.

A. DATA-DRIVEN TECHNIQUE

Moose [5] considered the effects of frequency offset on the performance of OFDM systems. According to [5], a frequency offset of more than 0.4% of the frequency spacing yields unacceptable system performance.

The data-driven technique estimates the frequency offset from the repetition of the same data frame by maximum likelihood. The OFDM signal at the receiver is given by [8]

$$r_n = (1/N)\left[\sum_{k=-K}^{K} X_k H_k e^{2\pi jn(k+\varepsilon)/N}\right] \quad \text{for } n=0,1,....,N-1 \tag{5}$$

where X_k is the signal transmitted, H_k is the transfer function of the channel at the k^{th} carrier, and ε is the frequency offset (in the notation used in [8]). In order to determine the frequency offset, two received data frames are compared at each frequency given by [8]

$$R_{1k} = \sum_{n=0}^{N-1} r_n e^{-2\pi jnk/N}, \qquad k = 0,1,2,...,N-1 \tag{6}$$

where R_{1k} is the k-th element of the discrete Fourier transform (DFT) of first frame and

$$R_{2k} = \sum_{n=0}^{N-1} r_{n+N} e^{-2\pi jnk/N}, \qquad k = 0,1,2,...,N-1 \tag{7}$$

where the R_{2k} is k-th element of the DFT of the second frame. From (5), we can see that

$$r_{n+N} = r_n e^{2\pi j\epsilon} \rightarrow R_{2k} = R_{1k} e^{2\pi j\epsilon}. \tag{8}$$

If noise (W_{1k} and W_{2k}) is added, from (6) and (8), the signals one and two at the receiver become

$$\begin{aligned} Y_{1k} &= R_{1k} + W_{1k} \\ Y_{2k} &= R_{1k} e^{2\pi j\varepsilon} + W_{2k}, \qquad k = 0,1,2,...,N-1. \end{aligned} \tag{9}$$

A schematic of the data-driven technique is shown in Figure 14. The data vector X is sent twice, and the received signals Y_1 and Y_2 are compared with each other. Because of the frequency offset, Y_1 and Y_2 are not equal. In order to determine the relative frequency offset ε,we can use a maximum likelihood approach which yields [8]

$$\hat{\varepsilon} = \left(\frac{1}{2\pi}\right) \tan^{-1} \left\{ \frac{\left(\sum_{k=-K}^{K} \mathrm{Im}\left[Y_{2k} Y_{1k}^{*} \right] \right)}{\left(\sum_{k=-K}^{K} \mathrm{Re}\left[Y_{2k} Y_{1k}^{*} \right] \right)} \right\} \tag{10}$$

where $\hat{\varepsilon}$ is the maximum likelihood estimate of the relative frequency offset defined as $\varepsilon = N\Delta f / B$, where B is bandwidth, N is the number of subcarriers and Δf is the frequency offset in Hz.

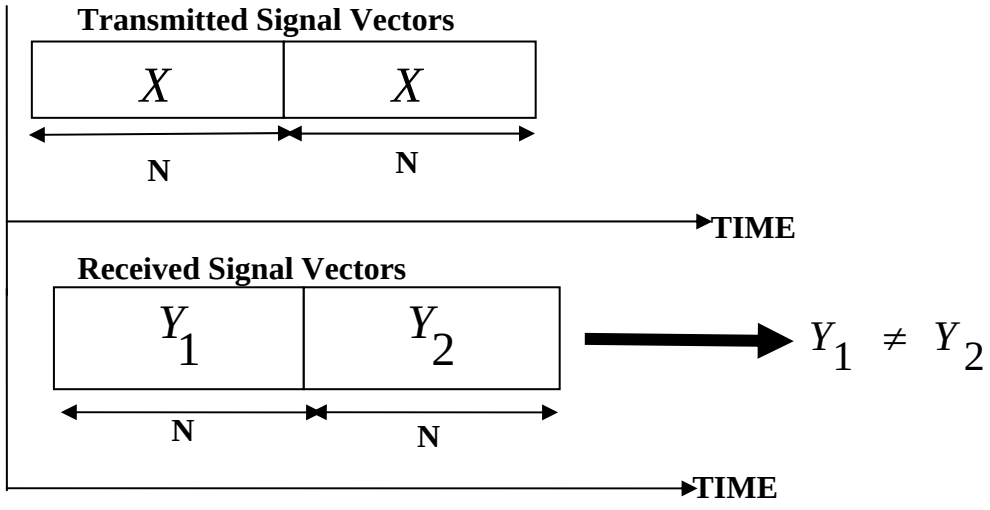


Figure 14. Schematic reprasentation of the data-driven (Moose) approach

Using the data-driven estimation technique, a series of simulations were run, and the results are depicted below. The simulations were done by modifying the Matlab code in [1-4]. The modified Matlab code is available from thesis advisor, Prof. Roberto Cristi, upon request. The maximum likelihood estimation of the frequency offset is obtained for an OFDM system using a block length of 256 symbols. The simulation results for different E_b/N_0 values are shown in Figure 15. It can be seen from Figure 15 that we have satisfactory frequency offset estimations as a percentage of the frequency spacing.

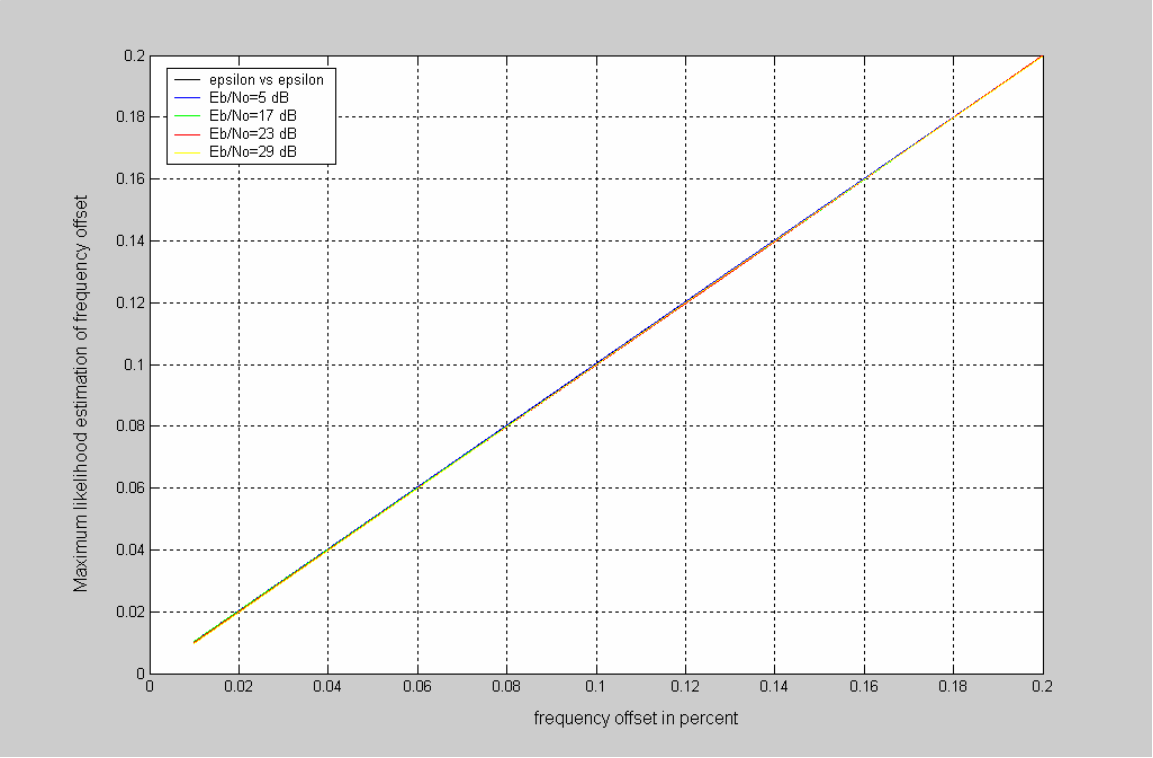


Figure 15. Maximum likelihood frequency offset estimation for 256 block length versus relative frequency offset as percentage of the frequency spacing

The block length of the OFDM frame has an effect on the frequency offset estimation. The results derived from simulation using the data-driven technique show an improvement with longer frames. The result obtained for block lengths of 64 is shown in Figure 16. From Figures 15 and 16, it can be seen that if we increase the block length of the OFDM frame, we can have better frequency offset estimation results by using the data-driven technique. The data-driven technique has been shown [8] to be successful for frequency offset values up to 50% of the frequency spacing.

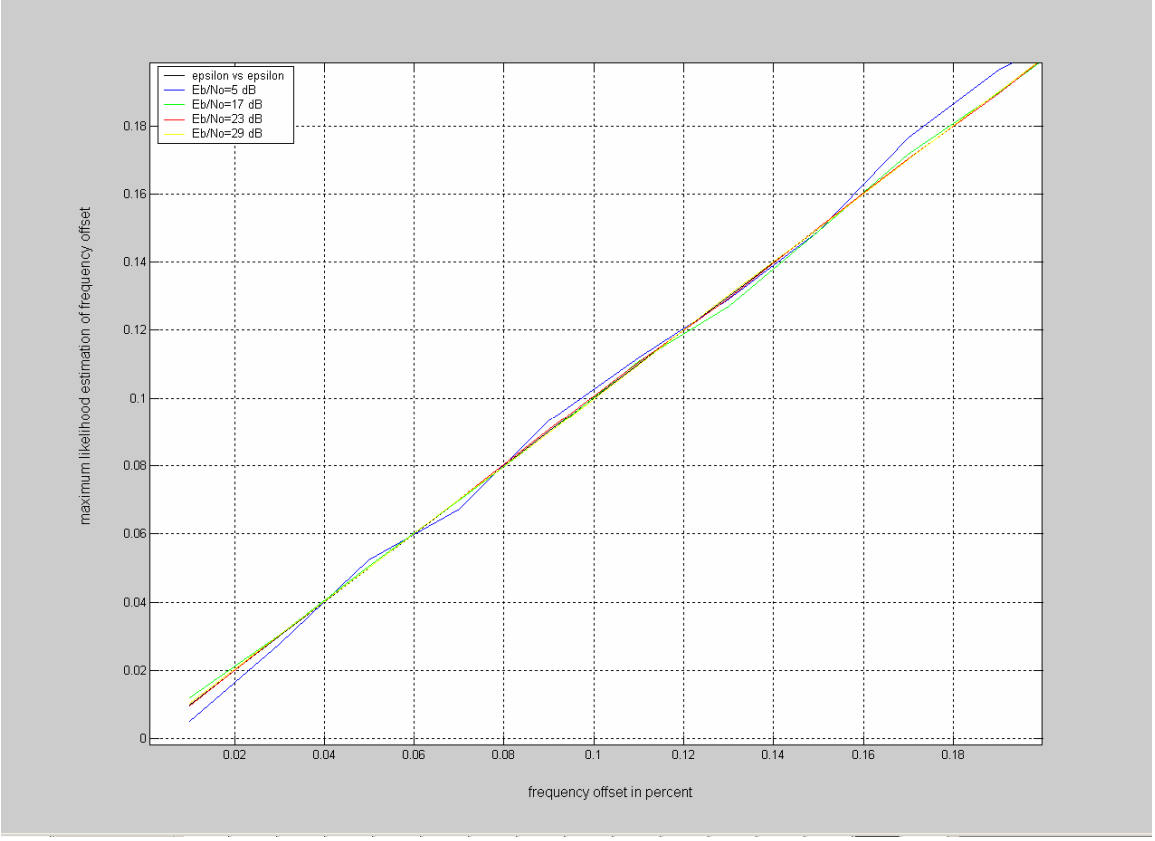


Figure 16. Maximum likelihood frequency offset estimation for 64 block length versus relative frequency offset as percentage of the frequency spacing

B. BLIND ESTIMATION TECHNIQUE

The blind technique is focused on determining both frequency and timing offset. The timing offset is another significant issue for OFDM systems, and a detailed discussion of this can be found in [14].

The blind technique estimates the frequency offset by analyzing the signal at the receiver. For a QPSK signal, if there is no offset and no noise, the exact QPSK signal is received at the receiver. If there is a frequency offset in the channel, the constellation is rotated by a phase proportional to the offset. The phase rotation at the constellation is shown in Figure 17 in which it can be seen that the constellation points are all rotated slightly. Frequency offset causes a uniform rotation of all tone phases because of the ICI. Frequency offset causes shifts in the clusters but does not change the centroids of the clusters.

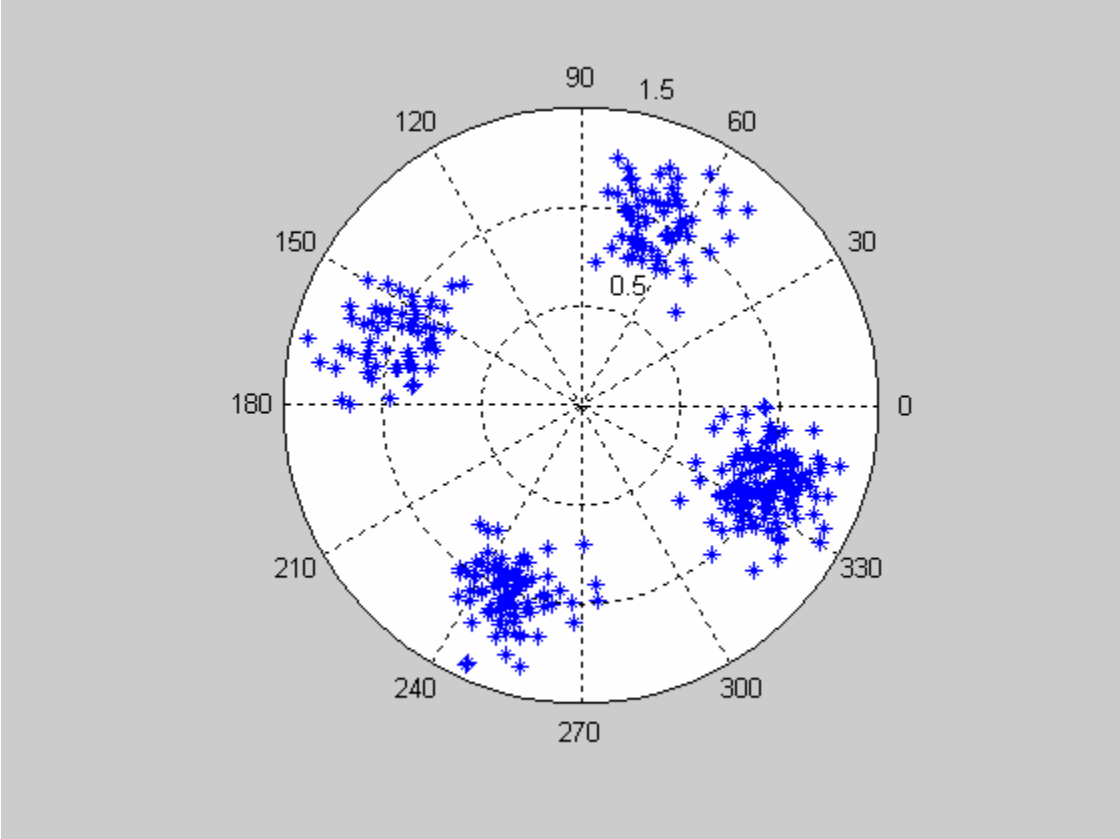


Figure 17. QPSK signal constellation at the receiver with frequency offset

In order to design an algorithm, which is not data dependent, in the case of BPSK and QPSK, we can use a very simple feature of the constellation. In particular, if

$$x(m) = \pm 1 \tag{11}$$

as in BPSK, we see that

$$x(m)^2 = 1 \tag{12}$$

independent of data. In the case of QPSK, where

$$x(m) = e^{j\left(\frac{\pi}{4} + k\frac{\pi}{2}\right)}, \qquad k = 0,...,3 \tag{13}$$

then

$$\left(x(m)\right)^4 = e^{j\pi} = -1, \tag{14}$$

independent of the data. In this way, given a received signal

$$y(m) = Ae^{j\phi}x(m) \tag{15}$$

when $x(m)$ is QPSK, we can write

$$y(m)^4 = -A^4 e^{j4\phi} \tag{16}$$

since

$$x(m)^4 = -1. \tag{17}$$

This shows that regardless of the transmitted data, the amplitude A and phase ϕ of the channel can be estimated from $y(m)^4$, provided the data is QPSK. Similarly, it can be estimated from $y(m)^2$, provided the data is BPSK. Figure 18 shows this fact where we assume the data to be QPSK. In order to estimate A and ϕ of the channel, we use magnitude and phase of the center mass of the data. It has been shown in [6] that a small frequency offset results in a shift in phase of the channel, which can be measured from the data as shown.

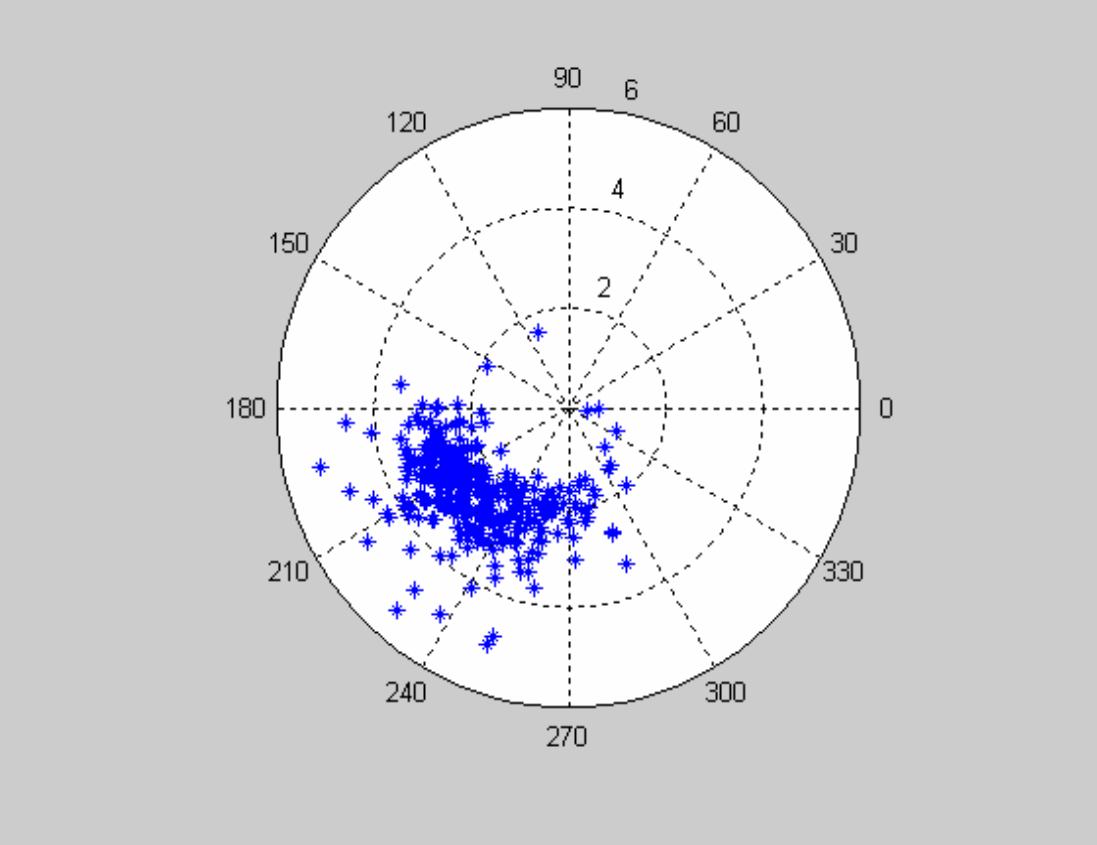


Figure 18. 4^{th} power of the received QPSK signal

C. COMPARISION OF DATA-DRIVEN AND BLIND TECHNIQUES

In this chapter, the two frequency offset estimation techniques are examined and compared. As previously discussed, these two methods are the data-driven technique, which uses a data-aided algorithm, and the blind technique, which uses a non-data-aided (blind) method.

Recall that the data-driven technique is superior to the blind estimation technique in frequency offset estimation under conditions of varying noise power. However, the blind technique is easier to apply than the data-driven technique, because data from the transmitter are not required.

Both the data-driven and blind techniques gave satisfactory estimations under 0.5% frequency offset values. Under different noise values, the blind technique gave worse estimations than the data-driven technique. It was observed that both techniques are affected equally by the increase in noise level.

Figure 19 provides a comparison of the two frequency offset estimation techniques, data-driven and blind, at 10 dB. It can be seen from Figure 19 that the data-driven technique provides a more accurate frequency offset estimate than the blind technique.

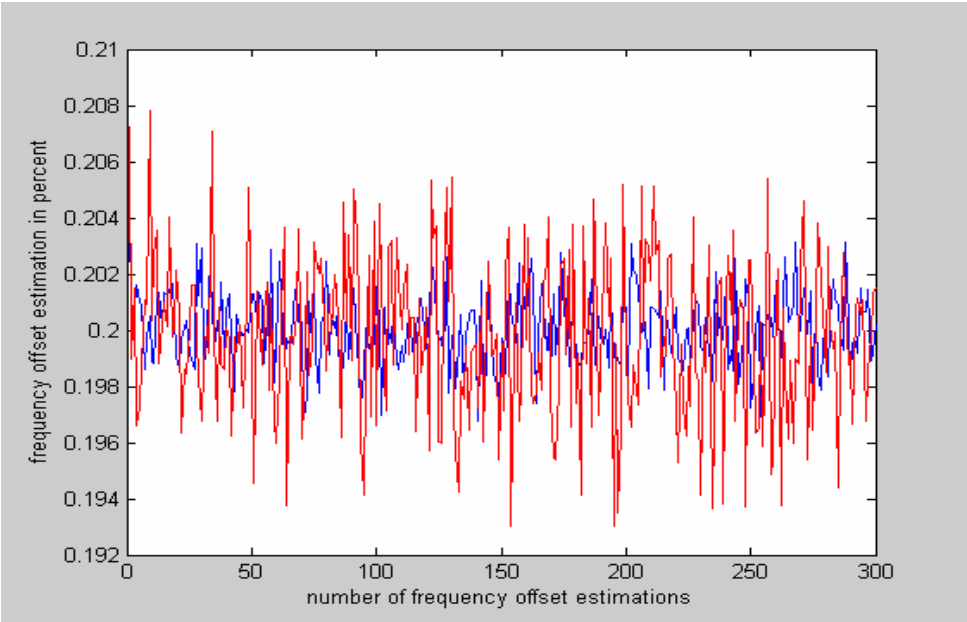


Figure 19. Comparision of data-driven (blue) and blind (red) techniques at 10 dB

In this chapter, two frequency estimation techniques, data-driven and blind, were studied. We have examined the data-driven technique proposed by Moose [5] and the blind technique proposed by Cimini [6]. We have seen that both techniques give satisfactory results for frequency offset estimations and that the data-driven technique is more robust than the blind technique; however the blind technique is easier to implement than the data-driven technique.

The following chapter examines the use of cyclic prefixes and preambles for frequency offset estimation.

V. PREAMBLE AND CYCLIC PREFIX USAGE FOR FREQUENCY OFFSET ESTIMATION

In this chapter, the use of cyclic prefixes and preambles for frequency offset estimation is examined. A preamble is used for both channel and frequency offset estimation in OFDM systems. Only the data-driven estimation techniques are used to estimate the frequency offset with a preamble, since pilot tones are used in this method. In this study, the data-driven algorithm described in Chapter 4 was used.

A. PREAMBLE STRUCTURE FOR FREQUENCY OFFSET ESTIMATION

In a communication system, the preamble is designed to establish frequency and time synchronization. The IEEE 802.11a standard gives guidelines on the use of various preamble structures [8]. The preamble structure, according to the IEEE 802.11a standard, as shown in Figure 20 is made of ten short symbols A1-A10 (16 samples each) for packet detect, AGC, diversity selection, coarse frequency offset estimation, and symbol timing. Also, two long symbols (64 symbols each) C1 and C2 are used for channel estimation and fine frequency offset estimation. The cyclic prefix (CP) is 32 samples long, longer than the CP during the data transmission. This is to accommodate the longer timing uncertainty during synchronization.

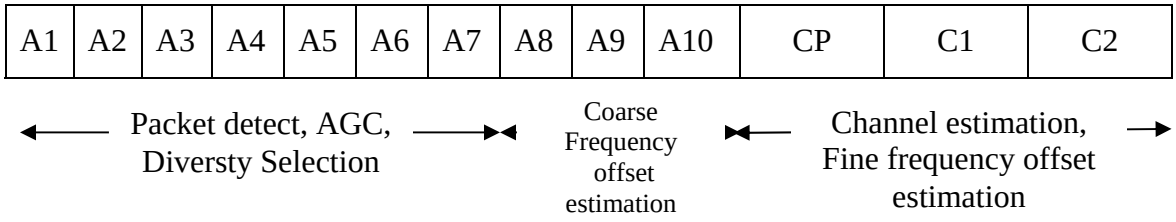


Figure 20. The IEEE 802.11a standard preamble (From Ref. 8.)

Preamble usage is very helpful for frequency estimation in OFDM systems. This is because the use of pilot tones at the receiver helps to analyze the received signal. If there is enough information about the signal at the receiver, this information can be used to estimate the frequency offset. If the data with known OFDM symbols are sent in the preamble, these symbols can be used for frequency offset estimation. These symbols are

called pilot tones, and only a few pilot tones are required for frequency offset estimation, since frequency offset has the same effect on each subcarrier.

B. FREQUENCY OFFSET ESTIMATION BY USING THE PREAMBLE

In IEEE 802.11a, the frequency offset is estimated using the preamble. In particular, the first 160 samples constitute the short preamble, which yields a coarse frequency estimate. The second part of the preamble is constituted of the same frame of 64 samples repeated twice. This yields fine frequency estimate. After estimating the frequency offset in the preamble, the estimate is updated at every frame to keep track of frequency drifts in the oscillator.

In order to track the frequency changes, the cyclic prefix is used in conjunction with the data-driven method. This is made possible by the fact that the CP is the repetition of the last 16 data samples of the data frame so that we can apply Moose's algorithm. The Matlab code used for simulating the frequency offset estimation with preambles at the beginning of the data is presented in Appendix D.

Figure 21 shows the estimate based on CP only. It can be seen from Figure 21 that the estimation has a larger initial error, which decays to zero with time.

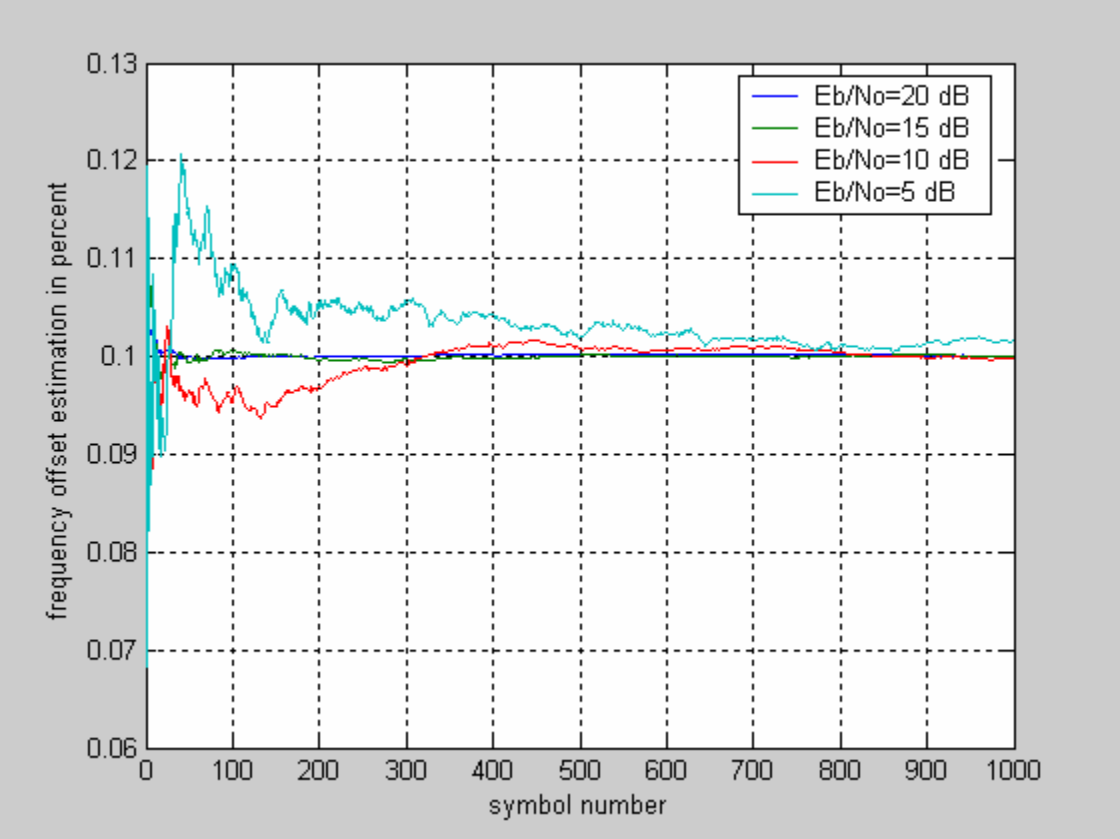


Figure 21. Frequency offset estimation using a preamble at the beginning of the data

Frequency offset estimation with a preamble at the beginning of the data for different offset values is depicted in Figure 22. It is seen that it is harder to make robust frequnecy offset estimation when there is noise in the system and that estimation using the data-driven technique with preamble yields similar results to data-driven and blind techniques, as demonstrated in the previous chapter.

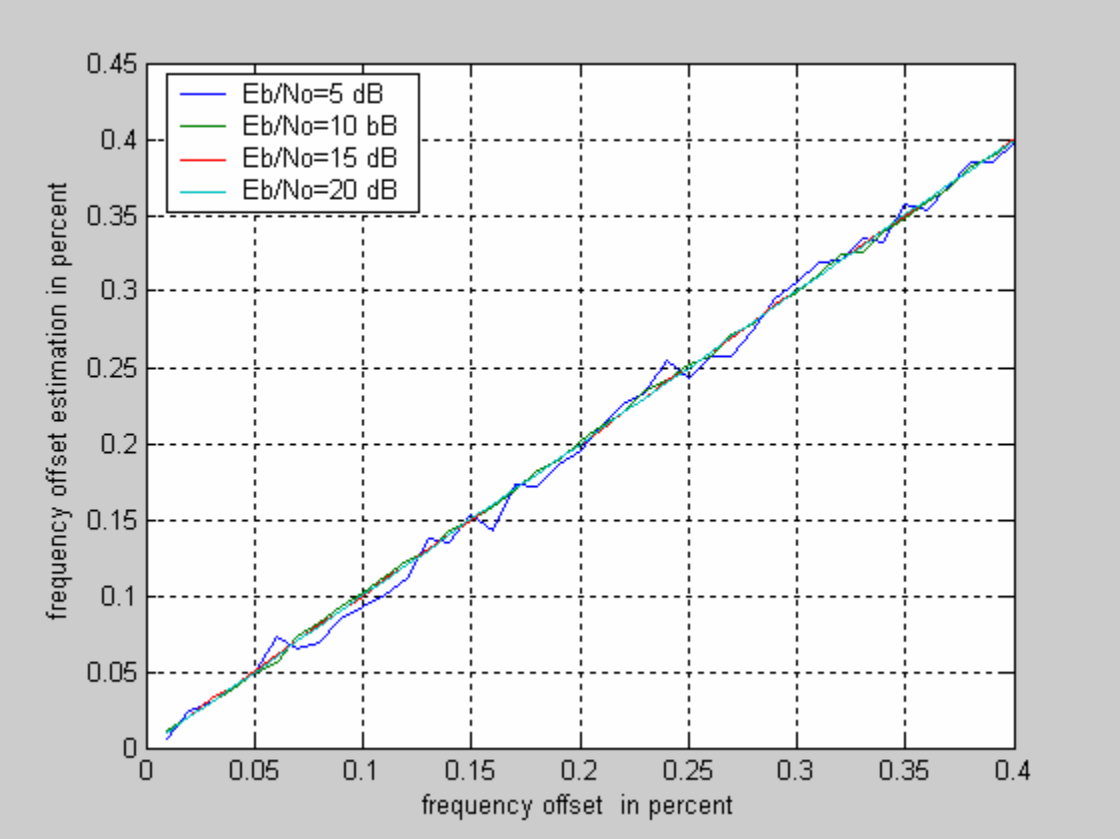


Figure 22. Frequency offset estimation versus frequency offset using a preamble at the beginning of the data

In this chapter, preamble data at the receiver have been used to estimate frequency offset. From this analysis, it has been shown that a preamble helps estimate the frequency offset and correct estimation results. However, the use of pilot tones reduces the bandwidth efficiency because some of the data frames are used for pilot tones instead of user data.

The following chapter presents a summary of the work performed in prior chapters and the significant results.

VI. CONCLUSION

The objective of this thesis was to investigate the effects of frequency offset and the performance of three different frequency offset estimation techniques. The objective was accomplished by investigating the effects of frequency offset on OFDM systems, the performance of the data-driven and blind estimation techniques, and preamble usage in frequency offset estimation.

A. SUMMARY OF WORK PERFORMED

This thesis presented three methods for frequency offset estimation: data-driven, blind and semi-blind. The data-driven and semi-blind rely on the repetition of data, while the blind technique determines the frequency offset from the QPSK data.

The MATLAB codes presented in Appendix A, B, C, D and in [1-4] were used as the basic simulation programs for this research and were modified as appropriate. Effects of the data-driven and blind estimation methods and the usage of the preamble in frequency offset estimations were simulated by modifying the MATLAB simulation in [1].

B. SIGNIFICANT RESULTS AND CONCLUSIONS

Simulation results demonstrated the distortive effects of frequency offset on OFDM signals; frequency offset affects symbol groups equally. Additionally, it was seen that an increase in frequency offset resulted in a corresponding increase in these distortive effects and caused degradation in the SNR of individual OFDM symbols.

The data-driven method performed best in terms of frequency offset estimate. However, it requires the data to be repeated twice, which affects the data rate. The blind method is sensitive to noise, but it performs satisfactorily with high SNR.

The use of preambles in frequency offset estimation was also examined. It was seen that the use of preambles and cyclic prefixes increase the robustness of the frequency offset estimation on that user data which comes after the preamble.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX A. MATLAB CODE OF IMPLEMENTATION OF SNR DEGRADATION CAUSED BY FREQUENCY OFFSET

This appendix presents the Matlab code used for simulating (4). The result of the implementation of (4) is shown in Figure 6.

```
%code----------------------------------------------------------------------------------
 % SNR degradation caused by frequency offset
%bulent karaoglu,december 2004,naval post graduate school
clear,clc,close all;
offset=[0.02 0.04 0.06 0.08 0.10 0.12 0.14 0.16 0.18 0.20 0.22 0.24 0.26 0.28 0.30
0.32 0.34...
      0.36 0.48 0.50];%frequnecy offset aqs percentage of the frequency spac-
ing,we enter directly
ebno=[1.5849 10 31.6228 50.11872] %signal-to-noise ratio(inverse of the loga-
rithmic (dB) value

for i=1:20
   tfft(i)=3.2*offset(i)/1000000; %ifft-fft period according to the timing related
parameters in table 1
   tgi(i)=0.8*tfft(i)/4000000; %guard interval duration according to the timing re-
lated parameters in table 1
   t(i)=4*(tfft(i)+tgi(i)); %symbol duration according to the timing related pa-
rameters in table 1
   %implementation of the formula which was given in
   %page 11,(formula number (4))

   debes(i)=(10*(pi*offset(1,i)*t(1,i))^2*ebno(1,1))/3*log(10);

   deon(i)=(10*(pi*offset(1,i)*t(1,i))^2*ebno(1,2))/3*log(10)
  deonbes(i)=(10*(pi*offset(1,i)*t(1,i))^2*ebno(1,3))/3*log(10);
  deonyedi(i)=(10*(pi*offset(1,i)*t(1,i))^2*ebno(1,4))/3*log(10);
  xekseni(i)=(offset(1,i));
  end
  figure(1);
 semilogy(xekseni,debes,'green');
  hold on;;
  semilogy(xekseni,deon,'red');
  hold on;
  semilogy(xekseni,deonbes);
  hold on;
  semilogy(xekseni,deonyedi,'yellow');
  grid ;
  xlabel('frequency offset in percent');
  ylabel(' SNR degradation (Dfreq) in dB');
  legend('17 db','15 db','10 db','5 db');
```

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX B. MATLAB CODE OF THE EFFECTS OF THE FREQUENCY OFFSET CASE 1 SIMULATIONS

This appendix presents the Matlab code used for simulating the model in Figure 7. The results of the simulation are shown in Figure 8, 9 and 10.

```
%code----------------------------------------------------------------------------------------
%author:bulent karaoglu,December 2004, naval post graduate school
%thesis of frequency offset effects -december 2004
clear
clc
close all
df=312500;  % FREQUENCY SPACING
% dw=input('GIVE THE FREQUEMCY OFFSET PLEASE:       ');
VAR=input('GIVE THE NOISE VARIANCE PLEASE:       ');
Z1=[];
for dw=312:312:5*312  % DIFFERENT FREQUENCY OFFSETS
CORRESPONDING TO 0.1, 0.2 , 0.3 , 0.4 , 0.5 %
% GENERATING THE DATA FOR CASE 2
for k=1:31
   data(k,1:1000)=1*sign(randn(1,1000))+i*sign(randn(1,1000));
end
data(32,1:1000)=0;
for k=33:64
   data(k,1:1000)=1*sign(randn(1,1000))+i*sign(randn(1,1000));
end
% TAKE THE IFFT OF THE DATA
iftdata=ifft(data);
% ADD CYCLIC PREFIX
iftdatacp=[iftdata(49:64,:;iftdata];
% MAKE THE DATA SERIAL
datatx=reshape(iftdatacp,1,80000);
% CHANNEL SIMULATION - ADD FREQUENCY OFFSET AND NOISE
n=0:(length(datatx)-1);
dataoffset=datatx.*exp(i*dw/df*n)+sqrt(VAR)*randn(1,80000)
% DEMODULATION
datarv=reshape(dataoffset,80,1000);
datarmvrv=datarv(17:80,:);
datafin=fft(datarmvrv);
Z=var(datafin(32,:));
Z1=[Z1 Z];
CASES(dw,:)=datafin(32,:);
end
% plotting the data
% figure(1)
```

```
% plot(CASES(312,:),'b*')
% xlabel('Real axis');ylabel('Imaginel axis');
% axis([-1 1 -1 1])
% grid on
%
% figure(2)
% plot(CASES(312*2,:),'r*');
% xlabel('Real axis');ylabel('Imaginel axis');
% axis([-1 1 -1 1])
% grid on
%
% figure(3)
% plot(CASES(312*3,:),'g*')
% xlabel('Real axis');ylabel('Imaginel axis');
% axis([-1 1 -1 1])
% grid on
%
% figure(4)
% plot(CASES(312*4,:),'b*')
% xlabel('Real axis');ylabel('Imaginel axis');
% axis([-1 1 -1 1])
% grid on
%
% figure(5)
% plot(CASES(312*5,:),'b*')
% xlabel('Real axis');ylabel('Imaginel axis');
% axis([-1 1 -1 1])
% grid on

figure(6)
plot(CASES(312,:),'k*')
hold on
plot(CASES(312*5,:),'r*')
xlabel('Real');ylabel('Imaginary');
legend('freq. offset=0.4%','freq.offset=0.6%');
grid on
% N=1:1000;
% figure(7)
% plot(N,abs(CASES(312,:)),'b',N,abs(CASES(312*5,:)),'r')
% xlabel('SYMBOL NUM');ylabel('VARIANCE');
% grid on
%
% figure(8)
% plot(3/300:3/300:15/300,Z1)
% xlabel('offset');ylabel('VARIANCE')
```

APPENDIX C. MATLAB CODE OF THE EFFECTS OF THE FREQUENCY OFFSET CASE 2 SIMULATIONS

This appendix presents the Matlab code used for simulating the model in Figure 11. The results of the simulation are shown in Figure 12, and 13.

```
%code----------------------------------------------------
%bulent karaoglu,December 2004.naval postgraduate school
%freq effects on qpsk
clear
clc
close all
df=312500;
dw=input('GIVE THE FREQUEMCY OFFSET PLEASE:       ');
VAR=input('GIVE THE NOISE VARIANCE PLEASE:       ');
Z1=[];
%for dw=0.5:10
% GENERATING THE DATA
data(31,1000)=0;
data(32,1:1000)=1*sign(randn(1,1000))+i*sign(randn(1,1000));
data(33,1:1000)=1*sign(randn(1,1000))+i*sign(randn(1,1000));
data(34:64,1:1000)=0;
% TAKE THE IFFT OF THE DATA
iftdata=ifft(data,[],2);
% ADD CYCLIC PREFIX

iftdatacp=[iftdata(49:64,:);iftdata];
% MAKE THE DATA SERIAL
datatx=reshape(iftdatacp,1,80000);
% CHANNEL SIMULATION - ADD FREQUENCY OFFSET AND NOISE
n=0:(length(datatx)-1);
dataoffset=datatx.*exp(i*dw*n)+sqrt(VAR)*randn(1,80000);
% DEMODULATION
datarv=reshape(dataoffset,80,1000);
datarmvrv=datarv(17:80,:);
datafin=fft(datarmvrv,[],2);
datavar=reshape(datafin,1,64000);
Z=var(datavar);
Z1=[Z1 Z]
%end
% plotting the data
figure(1)
plot(datafin(32,:),'*')
xlabel('real');ylabel('imaginel')
grid on
```

```
axis([-1.8 1.8 -1.8 1.8])
figure(2)
plot(datafin(33,:),'*')
title(' ..... ')
grid on axis([-1.8 1.8 -1.8 1.8])

figure(3)
plot(real(datafin(34,:)),imag(datafin(34,:)),'*')
xlabel('real');ylabel('imaginel')
grid on
axis([-1.8 1.8 -1.8 1.8])
% ESTIMATING THE VARIANCE OF THE DATA RECEIVED
% Z2=var([datarv(1),datarv3(1)]);
% Z3=var([datarv(1),datarv4(1)]);
% Z4=var([datarv(1),datarv5(1)]);
% Z5=var([datarv(1),datarv6(1)]);
% Z=[Z1,Z2,Z3,Z4,Z5];
%
% figure(7)
% i=[1:5];
% plot([0 0.05*i],[0 Z])
```

APPENDIX D. MATLAB CODE OF THE USE OF CYCLIC PREFIXES AND PREAMBLES SIMULATIONS

This appendix presents the Matlab codes used for simulating the use of the cyclic prefixes and preambles. The result of Code 1 is shown in Figure 21, and the result of Code 2 is shown in Figure 22.

```
%code1--------------------------------------------------------------------------
%the use of cyclic prefixes and preambles in OFDM frequnecy offset
%estimation
%bulent karaoglu,dec 2004,nps
close all
clc
clear
VAR=0.00002;  % CASE 1
d=0;
eps=0.13
 for w=1:1000       % AVERAGING
      S=zeros(1,64);
      k=1:6;
      S(1+4*k)=[-1-j,-1-j,1+j,1+j,1+j,1+j];     % positive freq
      S(1+64-4*k)=[1+j,-1-j,-1-j,1+j,-1-j,1+j];          % negative freq
      S=sqrt(13/6)
      s=ifft(S);
      r_S=[s(1:32),s,s];
      Lp=[0,1,-1,-1,1,1,-1,1,-1,1,-1,-1,-1,-1,-1,1,1,1,-1,-1,1,-1,1,-1,1,1,1,1,1];
      Lm=[1,1,1,1,-1,1,-1,1,1,1,-1,-1,1,1,1,1,1,1,1,-1,1,-1,1,1,1,-1,-1,1,1];
      L=[Lp,zeros(1,11),Lm];
      ell=ifft(L);
      r_L=[ell(33:64),ell,ell];           % Long Preamble
      pr=[r_S, r_L];
      % MULTIPLY WITH THE OFFSET FREQUENCY
      q=0:(length(pr)-1);
      profst=pr.*exp(i*eps*q/160)+sqrt(VAR)*randn(1,320);   % source
      profst1=profst(1:160);
      R1=fft(profst1);
      profst2=profst(161:320);
      R2=fft(profst2);
      epsest1(w)=phase(sum(R2.*conj(R1)));
      d=1+d;
      estimate1(d)=sum(epsest1)/w;
   end
   VAR=0.0002;  % CASE 2
d=0;
```

```
eps=0.13
   for w=1:1000

      S=zeros(1,64);
      k=1:6;
      S(1+4*k)=[-1-j,-1-j,1+j,1+j,1+j,1+j];     % positive freq
      S(1+64-4*k)=[1+j,-1-j,-1-j,1+j,-1-j,1+j];          % negative freq
      S=sqrt(13/6)*S
      s=ifft(S);
      r_S=[s,s,s(1:32)];
      Lp=[0,1,-1,-1,1,1,-1,1,-1,1,-1,-1,-1,-1,-1,1,1,1,-1,-1,1,-1,1,-1,1,1,1,1,1];
      Lm=[1,1,1,1,-1,1,-1,1,1,1,-1,-1,1,1,1,1,1,1,1,-1,1,-1,1,1,1,-1,-1,1,1];
      L=[Lp,zeros(1,11),Lm]
      ell=ifft(L);
      r_L=[ell(33:64),ell,ell];           % Long Preamble
      pr=[r_S, r_L];
      % MULTIPLY WITH THE OFFSET FREQUENCY
      q=0:(length(pr)-1)
      profst=pr.*exp(i*eps*q/160)+sqrt(VAR)*randn(1,320);   % source
      R1=fft(profst1);
      profst2=profst(161:320);
      R2=fft(profst2);
      epsest2(w)=phase(sum(R2.*conj(R1)));
      d=1+d;
      estimate2(d)=sum(epsest2)/w;
   end
   VAR=0.0001;  % CASE 3
d=0;
eps=0.13
   for w=1:1000
      S=zeros(1,64);
      k=1:6;
      S(1+4*k)=[-1-j,-1-j,1+j,1+j,1+j,1+j];     % positive freq
      S(1+64-4*k)=[1+j,-1-j,-1-j,1+j,-1-j,1+j];          % negative fre
      S=sqrt(13/6)*S;
      s=ifft(S);
      r_S=[s,s,s(1:32)];
      Lp=[0,1,-1,-1,1,1,-1,1,-1,1,-1,-1,-1,-1,-1,1,1,1,-1,-1,1,-1,1,-1,1,1,1,1,1];
      Lm=[1,1,1,1,-1,1,-1,1,1,1,-1,-1,1,1,1,1,1,1,1,-1,1,-1,1,1,1,-1,-1,1,1];
      L=[Lp,zeros(1,11),Lm];
      ell=ifft(L);
      r_L=[ell(33:64),ell,ell];           % Long Preamble
      pr=[r_S, r_L];
      % MULTIPLY WITH THE OFFSET FREQUENCY
      q=0:(length(pr)-1);
      profst=pr.*exp(i*eps*q/160)+sqrt(VAR)*randn(1,320);   % source
```

```
        profst1=profst(1:160);

        R1=fft(profst1);
        profst2=profst(161:320);
        R2=fft(profst2);
       epsest3(w)=phase(sum(R2.*conj(R1)));
        d=1+d;
        estimate3(d)=sum(epsest3)/w;
    end
    VAR=0.0005;  % CASE 4
d=0;
eps=0.13
    for w=1:1000
        S=zeros(1,64);
        k=1:6;
        S(1+4*k)=[-1-j,-1-j,1+j,1+j,1+j,1+j];     % positive freq
        S(1+64-4*k)=[1+j,-1-j,-1-j,1+j,-1-j,1+j];          % negative freq
        S=sqrt(13/6)*S;
        s=ifft(S);
        r_S=[s,s,s(1:32)];
        Lp=[0,1,-1,-1,1,1,-1,1,-1,1,-1,-1,-1,-1,-1,1,1,1,-1,-1,1,-1,1,-1,1,1,1,1,1];
        Lm=[1,1,1,1,-1,1,-1,1,1,1,-1,-1,1,1,1,1,1,1,1,-1,1,-1,1,1,1,-1,-1,1,1];
        L=[Lp,zeros(1,11),Lm]
        ell=ifft(L);
        r_L=[ell(33:64),ell,ell];            % Long Preamble

        pr=[r_S, r_L];
        % MULTIPLY WITH THE OFFSET FREQUENCY
        q=0:(length(pr)-1);
        profst=pr.*exp(i*eps*q/160)+sqrt(VAR)*randn(1,320);   % source
        profst1=profst(1:160);
        R1=fft(profst1);
        profst2=profst(161:320)
        R2=fft(profst2);
        epsest4(w)=phase(sum(R2.*conj(R1)));
        d=1+d;
        estimate4(d)=sum(epsest4)/w;
    end
number_in_avg=1:1000; % AVERAGE
figure(1)
plot(number_in_avg,estimate1,number_in_avg,estimate2,number_in_avg,estimate
 3,number_in_avg,estimate4)
legend('EB/NO=20 dB','EB/NO=15dB','EB/NO=10dB','EB/NO=5dB')
xlabel('symbol number');ylabel('offset estimations');
grid
%code 2-----------------------------------------------------------
```

```
%the use of cyclic prefixes and preambles part 2
%bulent karaoglu,dec 2004,nps
close all
clc
clear
VAR=0.01;  % CASE 1
for eps=1:1:40        % POINTS IN THE GRAPH
   for w=1:100

      S=zeros(1,64);
      k=1:6;
      S(1+4*k)=[-1-j,-1-j,1+j,1+j,1+j,1+j];    % positive freq
      S(1+64-4*k)=[1+j,-1-j,-1-j,1+j,-1-j,1+j];          % negative freq
      S=sqrt(13/6)*S
      s=ifft(S);
      r_S=[s,s,s(1:32)];
      pr=[r_S, r_S];
      % MULTIPLY WITH THE OFFSET FREQUENCY
      q=0:(length(pr)-1);
      profst=pr.*exp(i*eps*0.01*q/160)+sqrt(VAR)*randn(1,320);   % source
      profst1=profst(1:160);
      R1=fft(profst1);
      profst2=profst(161:320);
      R2=fft(profst2);
      epsest(w)=phase(sum(R2.*conj(R1)));
   end
   estimate1(eps)=sum(epsest)/100;
end
VAR=0.001;  % CASE 2
for eps=1:1:40
   for w=1:100

      S=zeros(1,64);
      k=1:6;
      S(1+4*k)=[-1-j,-1-j,1+j,1+j,1+j,1+j];    % positive freq
      S(1+64-4*k)=[1+j,-1-j,-1-j,1+j,-1-j,1+j];          % negative freq
      S=sqrt(13/6)*S;
      s=ifft(S);
      r_S=[s,s,s(1:32)];
      pr=[r_S, r_S];
      % MULTIPLY WITH THE OFFSET FREQUENCY
      q=0:(length(pr)-1);
      profst=pr.*exp(i*eps*0.01*q/160)+sqrt(VAR)*randn(1,320);   % source
      profst1=profst(1:160);
      R1=fft(profst1);
      profst2=profst(161:320);
```

```
      R2=fft(profst2);
      epsest(w)=phase(sum(R2.*conj(R1)));
   end
   estimate2(eps)=sum(epsest)/100;
end
VAR=0.0001; % CASE 3
for eps=1:1:40
   for w=1:100
      S=zeros(1,64);
      k=1:6;
      S(1+4*k)=[-1-j,-1-j,1+j,1+j,1+j,1+j];    % positive freq
      S(1+64-4*k)=[1+j,-1-j,-1-j,1+j,-1-j,1+j];          % negative freq
      S=sqrt(13/6)*S;
      s=ifft(S);
      r_S=[s,s,s(1:32)];
      pr=[r_S, r_S];
      % MULTIPLY WITH THE OFFSET FREQUENCY
      q=0:(length(pr)-1)
      profst=pr.*exp(i*eps*0.01*q/160)+sqrt(VAR)*randn(1,320);   % source
      profst1=profst(1:160);
      R1=fft(profst1);
      profst2=profst(161:320);
      R2=fft(profst2)
      epsest(w)=phase(sum(R2.*conj(R1)));
   end
   estimate3(eps)=sum(epsest)/100;
end
VAR=0;  % CASE 4
for eps=1:1:40
   for w=1:100
      S=zeros(1,64);
      k=1:6;
      S(1+4*k)=[-1-j,-1-j,1+j,1+j,1+j,1+j];    % positive freq
      S(1+64-4*k)=[1+j,-1-j,-1-j,1+j,-1-j,1+j];          % negative freq

      S=sqrt(13/6)*S
      s=ifft(S);
      r_S=[s,s,s(1:32)];
      pr=[r_S, r_S]
      % MULTIPLY WITH THE OFFSET FREQUENCY
      q=0:(length(pr)-1);
      profst=pr.*exp(i*eps*0.01*q/160)+sqrt(VAR)*randn(1,320);   % source
      profst1=profst(1:160);
      R1=fft(profst1);
      profst2=profst(161:320);
      R2=fft(profst2)
```

```
      epsest(w)=phase(sum(R2.*conj(R1)));
   end
   estimate4(eps)=sum(epsest)/100;
end
eps=0.01:0.01:0.4;
figure(1)
plot(eps,estimate1,eps,estimate2,eps,estimate3,eps,estimate4)
legend('EB/NO=5DB','EB/NO=10DB','EB/NO=15DB','EB/NO=20DB')
xlabel('offset');ylabel('offset estimation');
grid
```

LIST OF REFERENCES

[1] A. Y. Erdogan, "Analysis of the Effects of Frequency Offset in OFDM Systems," Master's Thesis, Naval Postgraduate School, Monterey, California, 2004.

[2] K. C. Tan, "Development, Simulation and Evaluation of the IEEE 802.11a Physical Layer in a Multipath Environment," Master's Thesis, Naval Postgraduate School, Monterey, California, 2001.

[3] Serdar Umit Tezeren, "Reed-Muller codes in error correction in wireless adhoc networks," Master's thesis, Naval Postgraduate School, Monterey, California, 2004.

[4ghkhu] Ersoy Oz, "A Comparision of Timing Methods In Orthogonal Frequency Division Multiplexing (OFDM) Systems," Master's thesis, Naval Postgraduate School, Monterey, California, 2004.

[5] P.H. Moose, "A technique for orthogonal frequency division multiplexing frequency-offset correction," *IEEE Trans. on Commun*., vol. 42, no. 10, pp. 2908-2914, Oct. 1994.

[6] B. Mcnair, L.J. Cimini and N. Sollenerber, "A Robust and Frequency Offset Estimation Scheme for OFDM Systems," AT&T Labs- Research, New Jersey, 2000; http://www.novidesic.com/pubs/vtc99-513a.pdf, last accessed December 2004.

[7] Yun Chiu and Dejan Markovic and Haiyun Tang, Ning Zhang, "A Report on OFDM Receiver Design," University of Berkeley, California, 2000; http://bwrc.eecs.berkeley.edu/People/Grad_Students/dejan/ee225c/ofdm.pdf, last accessed December 2004.

[8] Richard Van Nee and Ramjee Prasad, *OFDM for Wireless Multimedia Communications*, The Artech House Universal Personal Communications, Norwood, MA, 2000.

[9] Juha Heiskala and John Terry, *OFDM Wireless LANs: A Theoretical and Practical Guide*, Sams Publishing, Indianapolis, 2002.

[10] T. Schmidl, and D. Cox, ''Robust frequency and timing synchronizatin for OFDM,'' *IEEE Trans on Consumer Elect*, vol. 43, no. 3, pp. 776-783, August 1997.

[11] J. J. Van de beek, M. Sandell, Borjenson, P-O., “ML Estimation of Timing and Frequency Offset in OFDM Systems,’’ *IEEE Trans. on Consumer Elect*, vol. 42, no. 10, pp. 2908-2914, Oct. 1994.

[12] Clark Robertson, Notes for EC4580 (Error Correction Coding), Naval Postgraduate School, 2003 (Unpublished).

[13] A.Y. Erdogan, Notes for OFDM , Naval Postgraduate School, 2004 (Unpublished).

[14] Mounir Ghogho and Ananthram Swami, *A Highly Accurate Blind Frequency-Offset Synchronization for OFDM Communication*, Army Research Laboratory, Adelphi, MD, 1995.

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California

3. Chairman, Code EC
   Department of Electrical and Computer Engineering
   Naval Postgraduate School
   Monterey, California

4. Professor Murali Tummala, Code EC/Tu
   Department of Electrical and Computer Engineering
   Naval Postgraduate School
   Monterey, California

5. Professor Roberto Cristi, Code EC/Cx
   Department of Electrical and Computer Engineering
   Naval Postgraduate School
   Monterey, California

6. Bulent Karaoglu
   Kurucular Sitesi C blok Daire 16
   81040
   Icerenkoy, Istanbul, TURKEY