

Winter 2007

Statistical characterization of correlation-based time/frequency synchronizers for OFDM

Anupama Sankaran

University of New Hampshire, Durham

Follow this and additional works at: <https://scholars.unh.edu/thesis>

Recommended Citation

Sankaran, Anupama, "Statistical characterization of correlation-based time/frequency synchronizers for OFDM" (2007). *Master's Theses and Capstones*. 346.

<https://scholars.unh.edu/thesis/346>

This Thesis is brought to you for free and open access by the Student Scholarship at University of New Hampshire Scholars' Repository. It has been accepted for inclusion in Master's Theses and Capstones by an authorized administrator of University of New Hampshire Scholars' Repository. For more information, please contact nicole.hentz@unh.edu.

**STATISTICAL CHARACTERIZATION OF CORRELATION-
BASED TIME/FREQUENCY SYNCHRONIZERS FOR
OFDM**

BY

ANUPAMA SANKARAN

Bachelor of Engineering, University of Madras, India, May 2003

THESIS

Submitted to the University of New Hampshire
in Partial Fulfillment of
the Requirements for the Degree of

Master of Science
in
Electrical Engineering

December, 2007

UMI Number: 1449605

UMI[®]

UMI Microform 1449605

Copyright 2008 by ProQuest Information and Learning Company.
All rights reserved. This microform edition is protected against
unauthorized copying under Title 17, United States Code.

ProQuest Information and Learning Company
300 North Zeeb Road
P.O. Box 1346
Ann Arbor, MI 48106-1346

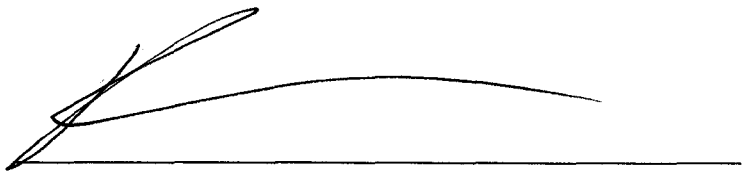
This thesis has been examined and approved.



Thesis Director
Prof. Michael J. Carter
Professor of Electrical and Computer Engineering



Prof. Kent Chamberlin
Professor of Electrical and Computer Engineering



Prof. Richard Messner
Professor of Electrical and Computer Engineering

29 November 2007

Date

ACKNOWLEDGEMENTS

I express my sincere gratitude to my advisor, Prof. Michael J. Carter, for offering his continuous help, support, suggestions, ideas, guidance and advice during the development of this thesis. I thank Dr. John LaCourse, Prof. Neda Pekaric-Nad and our wonderful administrative assistant, Kathy Reynolds for their endless backup throughout the course of my Masters studies at the University of New Hampshire. I extend my sincere thanks to Prof. Kent Chamberlin and Prof. Richard Messner for taking the time off from their busy schedules to serve on the thesis committee.

Last but not the least, many thanks to IEEE for giving permission to reproduce figures from their journal publications.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS.....	iii
LIST OF TABLES.....	ix
LIST OF FIGURES.....	x
ABSTRACT.....	xv
1 INTRODUCTION AND MOTIVATION.....	1
1.1 Introduction.....	1
1.2 Carrier Frequency Offsets and Symbol Timing Errors.....	3
1.3 False Lock.....	4
1.4 Motivation.....	4
1.5 Thesis Overview.....	7
2 BASIC PRINCIPLES OF OFDM.....	8
2.1 Introduction.....	8
2.2 OFDM Low Pass System Architecture.....	9
2.3 Role of Cyclic Extension.....	13
3 FADING CHANNEL MODELS.....	14
3.1 Introduction.....	14
3.1.1 Effect of Doppler.....	16
3.1.2 Inter-Symbol Interference (ISI).....	17
3.1.3 Coherence Bandwidth of the Channel.....	17

3.1.4	Time-Dispersion (Echoes).....	18
3.2	Channel Models for Indoor Wireless Communications.....	18
3.3	Types of Fading Channels.....	19
3.4	Flat Fading.....	20
3.5	Frequency Selective Fading.....	21
3.6	Fast Fading.....	21
3.7	Slow Fading.....	22
3.8	Rayleigh Distribution.....	23
3.9	Effects of Fading on Multicarrier Systems.....	23
3.10	Fading Channel Models Used for Simulation.....	24
3.10.1	A Time-Dispersive Rayleigh WATM Fading Channel Model.....	25
3.10.2	SISO WLAN Models from ETSI/BRAN.....	26
3.10.2.1	Model B (with five taps).....	27
3.10.2.2	Model C (with six taps).....	28
3.10.2.3	Model D (with seven taps).....	29
4	CORRELATION-BASED SYNCHRONIZATION FOR OFDM.....	30
4.1	Introduction.....	30
4.2	Effects of Carrier Frequency and Time Synchronization Errors.....	32
4.3	OFDM Synchronization by Autocorrelation.....	33
4.3.1	OFDM Frame Structure.....	34
4.3.2	Frequency Tracking and OFDM Symbol Synchronization....	37
4.3.3	Frequency Acquisition and OFDM Frame Synchronization..	38

4.4	Estimation of Energy in the Tail of OFDM Symbols.....	39
4.5	Real-Time Versus Non-Real Time Estimators.....	40
4.5.1	Delayed Decision Versus Sequential Tests.....	40
4.5.1.1	Latency.....	41
4.5.1.2	Averaging.....	41
4.5.1.3	Peak Detection Threshold.....	41
4.5.1.4	Simple Peak-Picking Algorithm.....	42
5	CORRELATION STATISTICS EXPERIMENTS.....	44
5.1	Introduction.....	44
5.2	Raw Correlator Magnitudes.....	45
5.3	Ensemble Statistics.....	50
6	IMPROVING CORRELATOR OUTPUT USEFULNESS.....	52
6.1	Introduction.....	52
6.2	Interpeak Intervals Above Threshold.....	52
6.3	Averaging with Variable Window Sizes.....	56
6.3.1	Averaging with a Window Size of 4.....	57
6.3.2	Averaging with a Window Size of 6.....	59
6.3.3	Averaging with a Window Size of 8.....	60
6.3.4	Averaging with a Window Size of 10.....	62
6.3.5	Comparison of Results from 7.3.1-7.3.4.....	63
7	CHARACTERIZATION OF CORRELATOR PERFORMANCE IN FADING CHANNELS.....	65
7.1	Introduction.....	65

7.2 Keller Channel Model [1].....	65
7.2.1 Raw Correlator Outputs for Keller Channel Model [1].....	66
7.2.2 Interpeak Intervals for Keller Channel Model [1].....	68
7.2.3 Averaging with Variable Window Sizes for Keller Channel Model [1].....	70
7.2.3.1 Averaging with a Window Size of 4.....	70
7.2.3.2 Averaging with a Window Size of 6.....	71
7.2.3.3 Averaging with a Window Size of 8.....	72
7.2.3.4 Averaging with a Window Size of 10.....	73
7.3 Model B [2].....	74
7.3.1 Raw Correlator Outputs for Model B [2].....	74
7.3.2 Interpeak Intervals for Model B [2].....	76
7.3.3 Averaging with Variable Window Sizes for Model B [2].....	77
7.3.3.1 Averaging with a Window Size of 4.....	77
7.3.3.2 Averaging with a Window Size of 6.....	78
7.3.3.3 Averaging with a Window Size of 8.....	79
7.3.3.4 Averaging with a Window Size of 10.....	80
7.4 Model C [2].....	81
7.4.1 Raw Correlator Outputs for Model C [2].....	82
7.4.2 Interpeak Intervals for Model C [2].....	83
7.4.3 Averaging with Variable Window Sizes for Model C [2].....	85
7.4.3.1 Averaging with a Window Size of 4.....	85
7.4.3.2 Averaging with a Window Size of 6.....	86

7.4.3.3	Averaging with a Window Size of 8.....	87
7.4.3.4	Averaging with a Window Size of 10.....	88
7.5	Model D [2].....	89
7.5.1	Raw Correlator Outputs for Model D [2].....	89
7.5.2	Interpeak Intervals for Model D [2].....	91
7.5.3	Averaging with Variable Window Sizes for Model D [2].....	93
7.5.3.1	Averaging with a Window Size of 4.....	93
7.5.3.2	Averaging with a Window Size of 6.....	94
7.5.3.3	Averaging with a Window Size of 8.....	95
7.5.3.4	Averaging with a Window Size of 10.....	96
8	CONCLUSIONS AND FUTURE WORK.....	98
8.1	Conclusions.....	98
8.2	Suggestions for Future Work.....	101
8.2.1	Timing Offset Estimation.....	101
8.2.2	Other Investigations.....	102
	REFERENCES.....	103
	APPENDICES.....	105
	APPENDIX A MATLAB CODE.....	107
	APPENDIX B IEEE COPYRIGHT PERMISSION LETTER.....	114

LIST OF TABLES

Table 3.1 Types of Fading.....	20
Table 3.2 Model B with five taps.....	27
Table 3.3 Model C with six taps.....	28
Table 3.4 Model D with seven taps.....	29

LIST OF FIGURES

Figure 1.1 (a) Time synchronization acquisition error histogram for a time-dispersive Rayleigh fading environment.....	5
Figure 1.1 (b) Frequency synchronization acquisition error histogram for a time – dispersive Rayleigh fading environment.....	6
Figure 2.1 (a) OFDM transmitter.....	9
Figure 2.1 (b) OFDM Receiver.....	10
Figure 3.1 Multipath in an indoor wireless environment.....	15
Figure 3.2 (a) Impulse response.....	25
Figure 3.2 (b) Frequency domain channel transfer function.....	26
Figure 3.3 Model B impulse response.....	27
Figure 3.4 Model C impulse response.....	28
Figure 3.5 Model D impulse response.....	29
Figure 4.1 Histogram of energy in the cyclic extension of each OFDM symbol...39	39
Figure 5.1 Scatter plot of the signal constellation.....	45
Figure 5.2 (a) Continuous plot of the output of $G(j)$	46
Figure 5.2 (b) $G(j)$ correlator output expanded.....	46
Figure 5.3 (a) $G(j)$ output near the null symbol.....	47
Figure 5.3 (b) Expanded $G(j)$ output with markers.....	47
Figure 5.4 (a) Continuous plot of the frame synchronizer output $R(j)$	48
Figure 5.4 (b) Expanded $R(j)$ output around the first frame boundary.....	49

Figure 5.5 Ensemble averaging of 20 $G(j)$ sequences.....	51
Figure 6.1 (a) Histogram of interpeak intervals above threshold = 0.01.....	53
Figure 6.1 (b) Histogram of interpeak intervals above threshold = 0.1.....	54
Figure 6.1 (c) Histogram of interpeak intervals above threshold = 0.2.....	54
Figure 6.1 (d) Histogram of interpeak intervals above threshold = 0.3.....	55
Figure 6.2 (a) Histogram of windowed average for 4 symbols.....	57
Figure 6.2 (b) Plot of windowed average for 4 symbols.....	58
Figure 6.2 (c) Sum of windowed average with 4 symbols.....	58
Figure 6.3 (a) Histogram of windowed average for 6 symbols.....	59
Figure 6.3 (b) Plot of windowed average for 6 symbols.....	59
Figure 6.3 (c) Sum of windowed average with 6 symbols.....	60
Figure 6.4 (a) Histogram of windowed average for 8 symbols.....	60
Figure 6.4 (b) Plot of windowed average for 8 symbols.....	61
Figure 6.4 (c) Sum of windowed average with 8 symbols.....	61
Figure 6.5 (a) Histogram of windowed average for 10 symbols.....	62
Figure 6.5 (b) Plot of windowed average for 10 symbols.....	62
Figure 6.5 (c) Sum of windowed average with 10 symbols.....	63
Figure 7.1 (a) Keller channel model [1] - $G(j)$ correlator output.....	66
Figure 7.1 (b) Keller channel model [1] - $G(j)$ output with timing offsets.....	66
Figure 7.1 (c) Keller channel model [1] - $R(j)$ output with markers.....	67
Figure 7.2 (a) Keller channel model [1] - Interpeak intervals above 0.01.....	68
Figure 7.2 (b) Keller channel model [1] - Interpeak intervals above 0.1.....	68
Figure 7.2 (c) Keller channel model [1] - Interpeak intervals above 0.3.....	69

Figure 7.3 (a) Keller channel model [1] – Windowed average with 4 symbols....	70
Figure 7.3 (b) Keller channel model [1] – Sum of windowed average for 4 symbols.....	70
Figure 7.4 (a) Keller channel model [1] – Windowed average with 6 symbols....	71
Figure 7.4 (b) Keller channel model [1] – Sum of windowed average for 6 symbols.....	71
Figure 7.5 (a) Keller channel model [1] – Windowed average with 8 symbols....	72
Figure 7.5 (b) Keller channel model [1] – Sum of windowed average for 8 symbols.....	72
Figure 7.6 (a) Keller channel model [1] – Windowed average with 10 symbols..	73
Figure 7.6 (b) Keller channel model [1] – Sum of windowed average for 10 symbols.....	73
Figure 7.7 (a) Model B [2] - correlator output.....	74
Figure 7.7 (b) Model B [2] - correlator output with timing offsets.....	75
Figure 7.7 (c) Model B [2] - output with markers.....	75
Figure 7.8 (a) Model B [2] – Interpeak intervals above 0.01.....	76
Figure 7.8 (b) Model B [2] – Interpeak intervals above 0.1.....	76
Figure 7.8 (c) Model B [2] – Interpeak intervals above 0.3.....	77
Figure 7.9 (a) Model B [2] – Windowed average with 4 symbols.....	77
Figure 7.9 (b) Model B [2] – Sum of windowed average for 4 symbols.....	78
Figure 7.10 (a) Model B [2] – Windowed average with 6 symbols.....	78
Figure 7.10 (b) Model B [2] – Sum of windowed average for 6 symbols.....	79
Figure 7.11 (a) Model B [2] – Windowed average with 8 symbols.....	79

Figure 7.11 (b) Model B [2] – Sum of windowed average for 8 symbols.....	80
Figure 7.12 (a) Model B [2] – Windowed average with 10 symbols.....	80
Figure 7.12 (b) Model B [2] – Sum of windowed average for 10 symbols.....	81
Figure 7.13 (a) Model C [2] - correlator output.....	82
Figure 7.13 (b) Model C [2] – Expanded output.....	82
Figure 7.13 (c) Model C [2] - output with markers.....	83
Figure 7.14 (a) Model C [2] – Interpeak intervals above 0.01.....	83
Figure 7.14 (b) Model C [2] – Interpeak intervals above 0.1.....	84
Figure 7.14 (c) Model C [2] – Interpeak intervals above 0.3.....	84
Figure 7.15 (a) Model C [2] – Windowed average with 4 symbols.....	85
Figure 7.15 (b) Model C [2] – Sum of windowed average for 4 symbols.....	85
Figure 7.16 (a) Model C [2] – Windowed average with 6 symbols.....	86
Figure 7.16 (b) Model C [2] – Sum of windowed average for 6 symbols.....	86
Figure 7.17 (a) Model C [2] – Windowed average with 8 symbols.....	87
Figure 7.17 (b) Model C [2] – Sum of windowed average for 8 symbols.....	87
Figure 7.18 (a) Model C [2] – Windowed average with 10 symbols.....	88
Figure 7.18 (b) Model C [2] – Sum of windowed average for 10 symbols.....	88
Figure 7.19 (a) Model D [2] - correlator output.....	89
Figure 7.19 (b) Model D [2] – Expanded output.....	90
Figure 7.19 (c) Model D [2] - output with markers.....	90
Figure 7.20 (a) Model D [2] – Interpeak intervals above 0.01.....	91
Figure 7.20 (b) Model D [2] – Interpeak intervals above 0.1.....	91
Figure 7.20 (c) Model D [2] – Interpeak intervals above 0.3.....	92

Figure 7.21 (a) Model D [2] – Windowed average with 4 symbols.....	93
Figure 7.21 (b) Model D [2] – Sum of windowed average for 4 symbols.....	93
Figure 7.22 (a) Model D [2] – Windowed average with 6 symbols.....	94
Figure 7.22 (b) Model D [2] – Sum of windowed average for 6 symbols.....	94
Figure 7.23 (a) Model D [2] – Windowed average with 8 symbols.....	95
Figure 7.23 (b) Model D [2] – Sum of windowed average for 8 symbols.....	95
Figure 7.24 (a) Model D [2] – Windowed average with 10 symbols.....	96
Figure 7.24 (b) Model D [2] – Sum of windowed average for 10 symbols.....	96

ABSTRACT

STATISTICAL CHARACTERIZATION OF CORRELATION-BASED TIME/FREQUENCY SYNCHRONIZERS FOR OFDM

by

Anupama Sankaran

University of New Hampshire, December, 2007

Orthogonal Frequency Division Multiplexing (OFDM) has been widely adopted as a modulation format for reliable digital communication over multipath fading channels, e.g. IEEE 802.11g “WiFi” networks, as well as broadband wireline channels, e.g. DSL modems. However, its robustness to channel impairments comes at the cost of increased sensitivity to symbol timing and carrier frequency offset errors, and thus requires more complex synchronization methods than conventional single-carrier modulation formats.

In this thesis, a class of synchronization methods based upon the intrinsic autocorrelation structure of the OFDM signal is studied from a statistical perspective. In particular, the reasons for the existence of irreducible time and frequency offset estimation errors in the limit of increasing *signal-to-noise ratio* (SNR) are investigated for correlator-based synchronizers for the non-fading channel case and several fading channel models. It is demonstrated that the primary source of irreducible synchronization errors at high SNR is the natural random distribution of signal energy in the cyclic prefix of the OFDM symbol.

Comparisons of the distribution of correlator output magnitude between the non-fading and fading channel cases demonstrates that fading skews the distribution with respect to the non-fading case. A potential mechanism for reducing the effect of innate signal energy variability, correlator output windowed averaging, is studied from the perspective of its influence on the distribution of interpeak intervals in the temporal correlator output signal. While improved performance is realized through averaging for the non-fading channel case, this technique is not as effective for fading channels. In either instance, the windowed averaging method increases the latency of the synchronization process and thus introduces delay in the overall demodulation process.

CHAPTER 1

INTRODUCTION AND MOTIVATION

1.1 Introduction

Indoor environments, such as homes and offices, are difficult environments for radio communication due to the phenomenon of *multipath propagation*. Multiple distorted copies of the transmitted signal collectively reach the receiver at slightly different times after being reflected by the walls, ceilings, floors, furniture, people etc. The principal effect of multipath is signal *fading* where the transmitted signal is attenuated at certain times due to destructive interference, and thus communication is disrupted.

Orthogonal Frequency Division Multiplexing (OFDM) is a bandwidth-efficient multicarrier modulation scheme which can cope with the above mentioned phenomena and at the same time make efficient utilization of the spectrum to significantly improve the performance of the system. However, as a parallel channel transmission technique, OFDM is extremely sensitive to carrier frequency offsets arising from the inherent instabilities of the oscillators in the transmitter and the receiver, and Doppler effect in the channel. Due to the sensitivity of OFDM to carrier frequency offsets and symbol timing errors, proper

synchronization between the transmitter and the receiver is of critical importance for a robust system.

OFDM exhibits high spectral efficiency, low multipath distortion and resiliency to time-dispersive channels, which is the reason why the performance of OFDM is often exemplary in a fading environment. Without the addition of a cyclic extension, OFDM is much more bandwidth efficient, and has lower sidelobes, than other lower level modulation schemes. As a consequence of these desirable attributes, the use of OFDM has greatly increased in the last few years. Apart from finding its primary use in Wireless Local Area Networks (WLANs), OFDM is also being used as the transmission technique for various other applications; some of these include:

- Legacy voice Digital Subscriber Line (ADSL) modems where it co-exists with the telephone signals (POTS) [1].
- Digital Audio Broadcasting (DAB) in Europe such as the Eureka 147 [16-19].
- Digital Video Broadcasting (DVB) [20].
- Radio broadcasting such as Digital Radio Mondiale (DRM) [22].
- HIPERLAN/2 [23-24].
- Wireless LANs (WLAN) [21].

1.2 Carrier Frequency Offsets and Symbol Timing Errors

In OFDM, the subdivision of a symbol into parallel low rate channels gives rise to a longer symbol duration, which causes the system to be less sensitive to timing offsets. The sub-carriers are said to be orthogonal only if they are spaced in such a way that at a particular sub-channel frequency where the received signal is evaluated, all other sub-channel signals are zero. In order to help preserve this orthogonality, the following conditions should be met:

- The transmitter and the receiver should be synchronized in both time and frequency.
- The physical components that constitute the transmitter and the receiver should be stable with respect to temperature, vibration, and mechanical shock.

The effect of carrier frequency offsets manifests as a loss of orthogonality between the sub-carriers. Symbol timing errors cause overlap between adjacent symbols leading to *Inter Symbol Interference (ISI)*, which eventually leads to interference between sub-carriers called *Inter Carrier Interference (ICI)*. Among the factors that lead to carrier frequency offsets, are the Doppler shift and the oscillator instabilities. On the other hand, for ISI-free detection, precise timing information identifying the beginning of an OFDM symbol is required so that the uncorrupted portion of the received OFDM symbol can be processed using the *Fast Fourier Transform (FFT)*. In order to estimate and compensate the errors due to synchronization, and also to improve system performance, reliable synchronization algorithms need to be used for these systems.

1.3 False Lock

In addition to the guard interval or the cyclic extension of the OFDM symbol, which acts as a rough symbol boundary and absorbs some of the effect of multipath and ISI, OFDM frames are typically embedded with multiple copies of a random sequence, also called a *synchronization pattern*, that the receiver uses for symbol boundary detection and coarse frequency offset estimation. A synchronization detection algorithm called a *correlator*, which is designed to match the random sequence at the receiver to the incoming signal, is employed for frame synchronization. Owing to the fact that the OFDM symbols are packed with random data, a false detection i.e., false lock may be caused by a partial match between the random data samples and the synchronization pattern. The presence of random noise and channel interference also potentially increases the probability of false lock and leads to loss of orthogonality between the sub-carriers. In cases where deep fades occur, there could even be a misdetection where the correlation peak suffers significant degradation in amplitude and hence is left undetected. Also, whenever a false lock occurs, there is a possibility that the receiver will continue to process the false lock and thereby fail to process a legitimate OFDM symbol.

1.4 Motivation

The correlation-based OFDM symbol timing and carrier frequency offset estimation algorithms proposed by Keller et al. [1] perform with good estimation accuracy for timing and frequency tracking and acquisition in an AWGN channel,

but the performance of the timing and frequency acquisition algorithms in a Rayleigh fading environment show larger variations. Figure 1.1 (a) and Figure 1.1 (b) shows the timing and frequency acquisition error histograms in a time-dispersive Rayleigh fading channel.

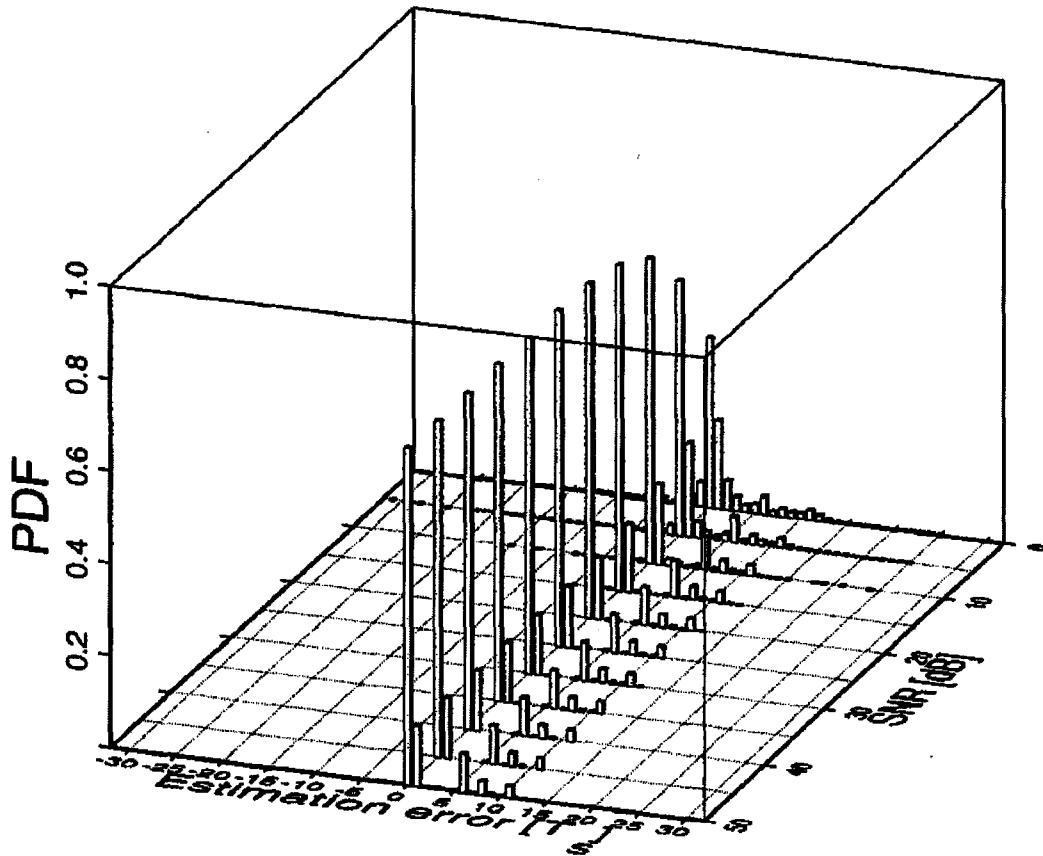


Figure 1.1 (a) Time synchronization acquisition error histogram for a time-dispersive Rayleigh fading environment

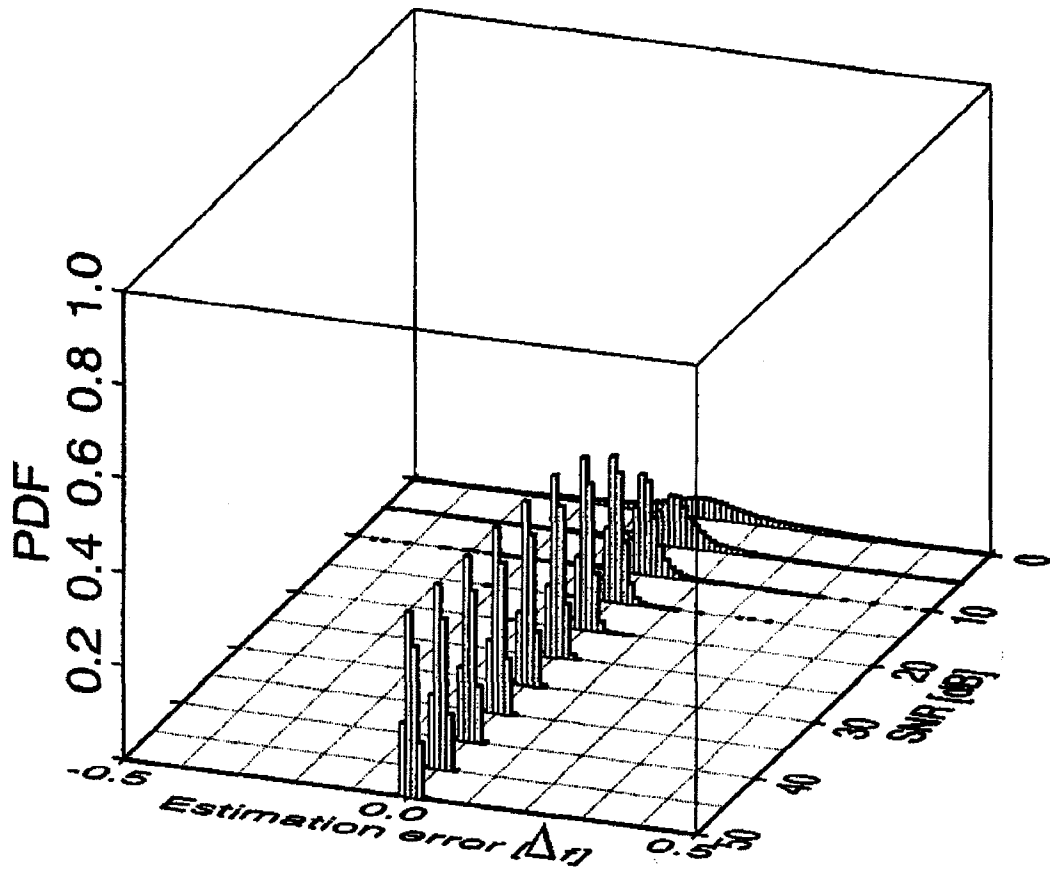


Figure 1.1 (b) Frequency synchronization acquisition error histogram for a time-dispersive Rayleigh fading environment

(Figures excerpted from [1], © [2001] IEEE, used with permission)

The timing estimation error histogram is spread over 11 sampling periods of the impulse response even at very high signal-to-noise ratios (SNR) of the order of 50 dB. The algorithm seeks to synchronize with the path with the highest amplitude. The frequency estimation errors, on the other hand, are below 5% of the sub-carrier spacing even for very high SNR values. This phenomenon of persistent false lock probability even at high SNR values served as motivation for further investigation of the performance of the correlation metrics in [1] in different fading environments. The choice of averaging

technique, detection threshold selection, and the choice of synchronization patterns strongly influence the false lock probability, yet little research has been done on these issues for multicarrier modulations in fading channels. The subsequent chapters discuss in detail the performance of the algorithms and suggestions for improvements.

1.5 Thesis Overview

Chapter 2 presents the basic principles of OFDM communication, the construction of OFDM symbols and the importance of the cyclic extension in OFDM modulation.

Chapter 3 introduces the reader to the concept of *fading* in wireless communications, giving more insights on the different types of fading and the fading channel models used for simulation purposes in this thesis.

Chapter 4 focuses on the correlation-based synchronizers for OFDM and the estimation of timing and frequency offsets from the correlator metrics. They also present various other synchronization techniques, their advantages and disadvantages, and a comparative study of the real-time and non-real time estimators for OFDM.

Chapters 5 through 7 give in detail the results of various experiments that were run with the correlation-based synchronization algorithms from [1] employing the various fading channel models from Chapter 3.

CHAPTER 2

BASIC PRINCIPLES OF OFDM

2.1 Introduction

Multicarrier modulation is currently used in numerous wireless systems. OFDM is a special form of multicarrier modulation and is especially suitable for transmission over dispersive channels. The essential goal of OFDM is to divide the input binary serial data stream into a large number of parallel data streams and transmit them simultaneously over different sub-channels, which are constructed to be orthogonal under ideal propagation conditions. The total number of sub-channels is chosen such that each sub-channel has a bandwidth less than the coherence bandwidth of the channel so that they undergo relatively *flat fading*, which denotes uniform attenuation across a range of frequencies. This offers two main advantages:

- The increase in the symbol duration of the sub-channel decreases the probability of the sub-channel symbols getting corrupted in a frequency selective fade.
- The decrease in sub-channel bandwidth eliminates the need for a channel equalizer.

The modulation schemes employed for OFDM sub-carriers are generally the Quadrature Amplitude Modulation (QAM) or Phase Shift Keying (PSK) with

both coherent and non-coherent detection. However, *Quadrature Phase Shift Keying (QPSK)* is used in our simulations.

2.2 OFDM Low Pass Equivalent System Architecture (Complex Envelope Representation)

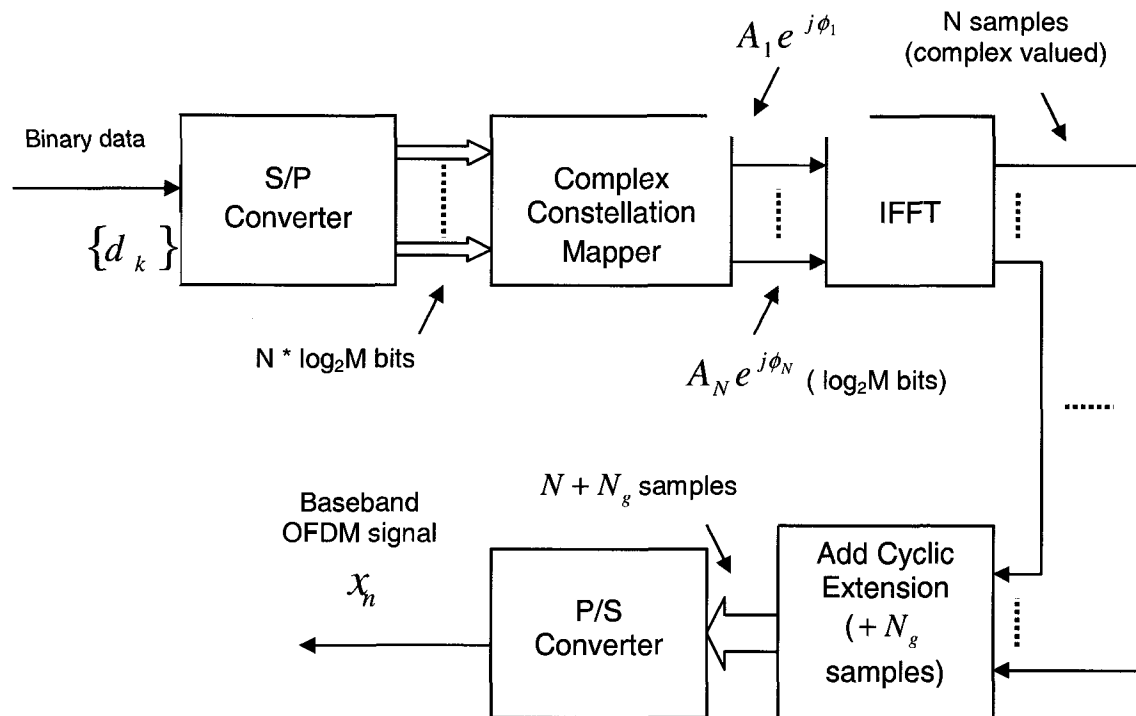


Figure 2.1 (a) OFDM Transmitter (Low Pass Equivalent)

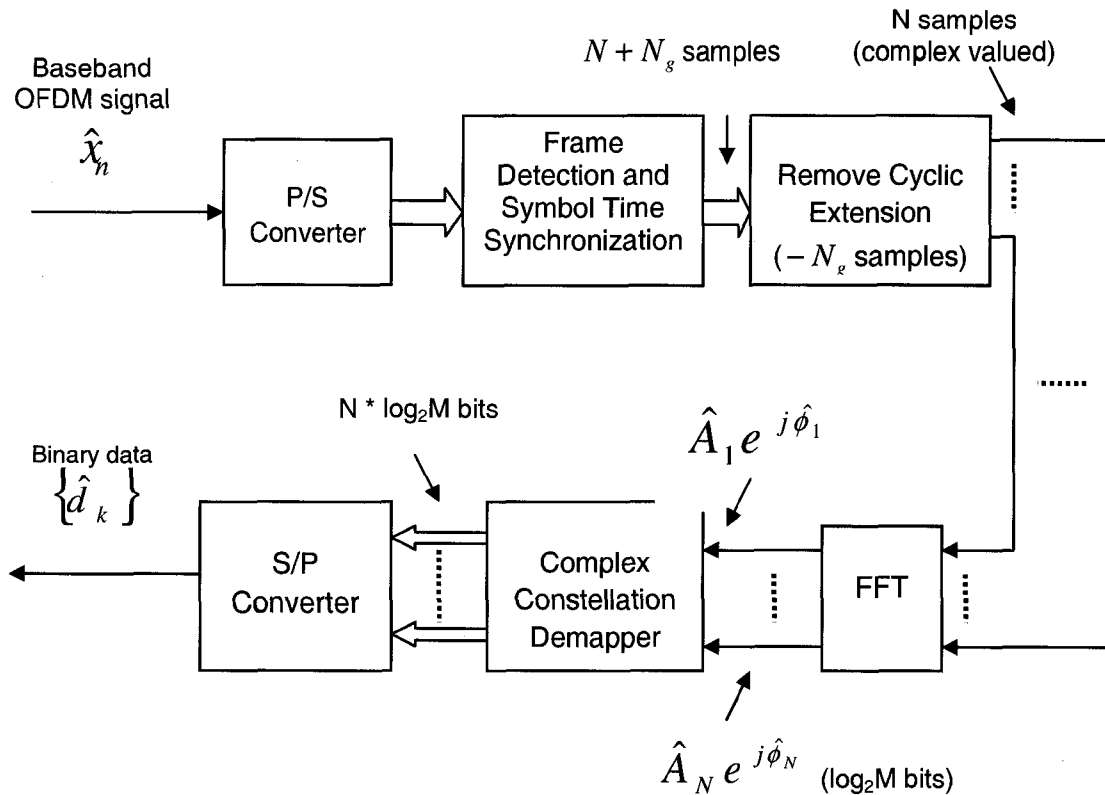


Figure 2.1 (b) OFDM Receiver (Low Pass Equivalent)

The OFDM modulation technique involves the simultaneous transmission of N sub-carriers, where N is an integer, that are orthogonal to each other in the time domain. If B is the bandwidth of the available spectrum in *Hertz*, the frequency separation between the N sub-carriers is $\Delta f = B/N$ and the sub-carriers are mutually orthogonal over a time $T_s = 1/\Delta f$.

Figure 2.1 shows the OFDM Low Pass Equivalent System Architecture wherein all the operation is carried out using low pass or baseband models including the transmission channel model. Figure 2.1 (a) shows the block diagram of an OFDM transmitter (low pass equivalent). The input binary data stream is divided into N parallel streams by the serial-to-parallel converter. This

process assigns $\log_2 M$ bits/word to each sub-carrier, where $M = 2^k$, k being an integer. The required amplitude and phase on each of the N parallel streams is then calculated based on the modulation scheme by passing the data on each stream through the complex constellation mapper, the output of which is a complex envelope, $A_k e^{j\phi_k}$, for the k^{th} sub-carrier. Since QPSK modulation is used in the work addressed here, each sub-channel is assigned 2 bits to represent one of the four states/constellation points associated with QPSK modulation, and $A_k = A$ for each k value. The four constellation points for QPSK are identified by $-1+i$, $-1-i$, $1+i$ and $1-i$, which have the same amplitude but differ in their phase. The spectra of N modulated parallel streams coming out of the QPSK modulator is converted to the time-domain by taking the *Inverse Fast Fourier Transform (IFFT)* of the N complex-valued samples. The length of the IFFT is chosen to be an integer power of 2 that is closest to the number of sub-carriers, N . The IFFT takes each complex-valued sample, which represent the *In-Phase (I)* and *Quadrature (Q)* components i.e., amplitude and phase of the modulation on each sub-channel and multiplies that by the carrier at the corresponding sub-channel frequency from its orthogonal basis set of sinusoids. Thus, the output of the IFFT is a linear combination of sinusoids which collectively form an OFDM symbol wherein each sample is the result of QPSK modulation on every sub-carrier from the basis set. The discrete baseband OFDM signal is given by,

$$x_n = \frac{1}{N} \sum_{k=0}^{N-1} a_k e^{\frac{j 2 \pi kn}{N}} \quad (2.1)$$

where a_k is the modulated complex signal on the k^{th} sub-carrier.

In order to effectively handle the impairments due to multipath delay spread, OFDM symbols are cyclically extended. This cyclic extension serves as the *Guard Interval (GI)* between adjacent OFDM symbols and also makes the system less sensitive to timing errors. The output of the IFFT operation is run through a parallel-to-serial converter and the cyclic extension of length N_g , comprising copies of the last N_g samples of each OFDM symbol, is attached to the start of each symbol, thereby making the effective length of the transmitted OFDM symbol to be $N + N_g$. Finally, the complex valued baseband signal is up-converted to *Radio Frequency (RF)* via a quadrature modulator and transmitted through the channel.

As shown in Figure 2.1 (b), after quadrature down-conversion and filtering of the received signal, the symbol timing and frequency synchronization algorithms are used to estimate the timing and frequency errors. The corrections are applied and the resulting signal is subjected to the FFT operation with the corrected symbol time offset, after removing the cyclic extension N_g .

Here, for simulation purposes, a total of 128 sub-channels were used and hence a 128 point IFFT was performed at the transmitter. The resulting OFDM symbol is cyclically extended by copying the last 13 samples (approximately 10% of the OFDM symbol length) to the start of each OFDM symbol. The OFDM

symbols are then grouped into frames with each frame containing a null symbol and a reference symbol followed by 62 OFDM symbols. The reference symbol is formed with repetitions of a unique synchronization pattern generated by the algorithms proposed in [3].

2.3 Role of Cyclic Extension

A widely accepted means of avoiding InterSymbol Interference (ISI) and preserving orthogonality of the sub-carriers is to add a cyclic extension to each OFDM symbol to induce circular convolution between the transmitted signal and the channel impulse response. To achieve this, the last N_g samples of each OFDM symbol are appended to the beginning of the corresponding symbol to form the complete OFDM symbol to be transmitted. At the receiver, the first N_g samples affected by ISI can be removed without much loss of information and the FFT can be performed on the remaining samples of the received sequence. Regardless of the loss of transmission power and bandwidth associated with the cyclic extension, the insertion of a cyclic extension results in a parallel orthogonal structure which allows for simple channel estimation. The length of the cyclic extension, N_g , is chosen to be longer than the maximum delay of the channel impulse response to preserve the sub-channel orthogonality under fading conditions. Normally, N_g is selected to be not more than $N/5$ ($N/10$ in our simulation), which can be interpreted as a 1 dB signal-to-noise ratio (SNR) loss introduced by the cyclic extension [14].

CHAPTER 3

FADING CHANNEL MODELS

3.1 Introduction

The design of wireless networks differs greatly from that of wired networks owing to the nature of the wireless channel, which is an unpredictable and difficult communications medium. In a typical indoor environment, a radio signal that is transmitted from a fixed source will traverse multiple different paths before reaching the receiver. This happens because the transmitter does not know precisely the location of the receiver, and the finite antenna directivity causes some radiation in all directions. So, in addition to the direct path of the signal from the transmitter to the receiver, also called the Line of Sight (LOS) path, the transmitted signal encounters reflectors such as walls, cabinets and other objects in the environment that produce reflected, diffracted and/or scattered copies of the transmitted signal. The relative delay and amplitude of each of these paths depends on the length of the path, the amount of energy absorbed by the reflectors and also the movement of the reflectors. These additional copies of the transmitted signal can be attenuated in power, delayed in time and shifted in phase and/or frequency with respect to the LOS signal path at the receiver. Figure 3.1 below shows the multiple paths in a fading environment.

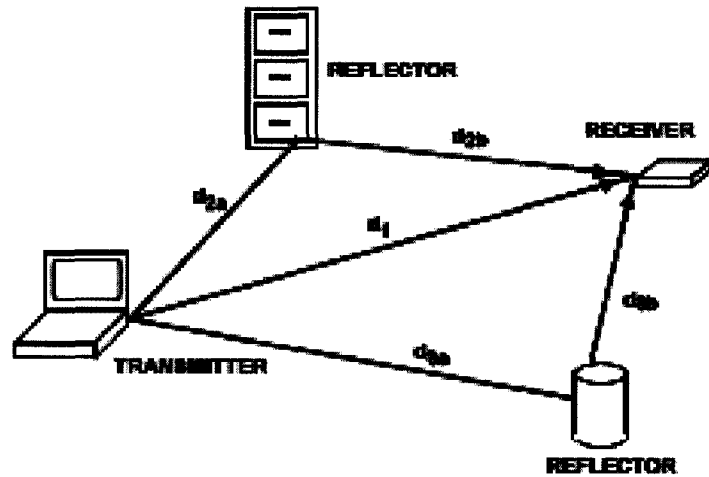


Figure 3.1 Multipath in an indoor wireless environment

(Figure excerpted from [15])

Fading occurs because two or more versions of these transmitted signals arrive at the receiver at different times. These versions of the transmitted signal, also called multipath waves, are summed together at the receiver to give a resultant signal which, most of the time, is distorted because of the nature of the transmission channel. The term *path loss* is used here to denote the local averaged received signal power relative to the transmit power. The rapid fluctuation of the amplitude of a radio signal over a short period of time or travel distance, for which the variation in large-scale path loss is negligible, is described as small-scale fading or simply fading. In small-scale fading, the power of the received signal could vary by 30 to 40 dB when the receiver is moved even by a fraction of a wavelength. These amplitude variations with spatial position are deterministic if the number, location and characteristics of the reflectors are known over time. In practice one cannot know exactly the positions and reflector

characteristics precisely enough to accurately estimate signal amplitude. Some of the effects of small-scale fading are discussed below.

3.1.1 Effect of Doppler

A receiver moving at a high speed can pass through several fade zones in a small period of time due to the constructive and destructive effects of multipath waves summing at various points in space. In fact, in some cases, the receiver may even stop at a specific location at which the received signal suffers a deep fade. In such cases, maintaining good communication characteristics can become very difficult, although passing vehicles or walking people can disturb the field pattern, thereby reducing the likelihood of the received signal remaining in a deep null for a long period of time. Each multipath wave experiences an apparent shift in frequency due to the relative motion between the transmitter and the receiver. This shift in the received signal frequency due to relative motion is called Doppler shift, which is directly proportional to the velocity and depends on the direction of motion of the receiver with respect to the direction of arrival of the received multipath waves. The Doppler shift is either positive or negative depending on whether the receiver is moving toward or away from the transmitter.

In communication channels that exhibit Rayleigh fading characteristics, the channel impulse response fluctuates with a high Doppler frequency, causing Inter-Carrier Interference (ICI). The orthogonality of the sinc-shaped sub-channel spectra of OFDM is lost as a result of these fluctuations, which changes the corresponding frequency domain transfer function during that time interval. A

channel is said to be non-stationary if the Rayleigh fading paths vary during an OFDM symbol period due to the high Doppler. On the other hand, a channel is said to be stationary if the Rayleigh fading channel model parameters do not change significantly during the OFDM symbol duration, but do vary over longer periods of time.

3.1.2 Inter-Symbol Interference (ISI)

Multipath propagation spreads the time over which the transmitted signal arrives at the receiver. This can cause ISI, where the received symbol in any given symbol period experiences interference from other symbols that have been delayed by multipath. Since increasing signal power also increases the power of ISI, this interference gives rise to an irreducible error floor that is independent of the signal power. It is difficult to analyze as it depends on the modulation format and the ISI characteristics, which in turn depends on the characteristics of the channel and the sequence of transmitted symbols.

3.1.3 Coherence Bandwidth of the Channel

If the bandwidth of the multipath channel is smaller than the bandwidth of the transmitted signal, the received signal suffers distortion, but its signal strength will not vary much over a local area. The bandwidth of the multipath channel can be quantified by the *coherence bandwidth* related to the specific multipath structure of the channel. Coherence bandwidth is defined as the range of frequencies over which the channel is considered *flat*, i.e., a channel that passes all the spectral components with equal gain and linear phase.

3.1.4 Time-Dispersion (Echoes)

Echoes arise due to the replicas of the transmitted signal reaching the receiver with different propagation delays, and are dependent on the geometric relationship between the transmitter, receiver and the surrounding environment. Some important statistics that can be used to quantify the time-dispersion such as *mean excess delay*, *maximum excess delay* and *RMS delay spread* can be derived from the channel path delay profiles [6]. The mean excess delay is the average excess delay of all the paths. The maximum excess delay is the value of excess time delay for which the signal amplitude is typically 20 dB below its peak amplitude. The RMS delay spread is a measure of the spread of the signal power about the mean delay time.

3.2 Channel Models for Indoor Wireless Communications

In order to generate an exact model for a specific fading environment, one needs to know the attributes of the reflectors in that environment at each instance in time, which is not very reasonable. For this purpose, channel models that closely replicate selected environments have been developed over the years. The most suitable model for our case, which exhibits Rayleigh fading characteristics, is described here.

An ensemble of n propagation paths that constitute the impulse response of a multipath Rayleigh fading channel is given by

$$h(t, \tau) = \sum_{i=1}^n A_i(t) * \delta(\tau - \tau_i) = \sum_{i=1}^n a_i * g_i(t) * \delta(\tau - \tau_i) \quad (3.1)$$

The above equation characterizes each propagation path i by a fixed delay τ_i and time varying amplitude $A_i(t) = a_i * g_i(t)$, the product of the complex amplitude a_i and the Rayleigh fading process $g_i(t)$.

3.3 Types of Fading Channels

Different transmitted signals undergo different types of fading depending on the relation between the signal parameters like bandwidth, symbol period, etc., and the channel parameters like RMS delay, Doppler spread, etc. The time delay spread of a multipath channel can result in significant distortion of the received signal and is equal to the delay spread between the arrival of the first received signal component and the last received signal component associated with a single transmitted pulse. If the delay spread is small when compared to the inverse of the signal bandwidth, then there is little time-spreading in the received signal. However, if the delay spread is large, then there is significant time-spreading of the received signal, which leads to substantial signal distortion. Having said that, multipath delays lead to time dispersion and frequency selective fading and Doppler spread leads to frequency dispersion and time selective fading. The four different types of fading are summarized in Table 3.1.

Types of Fading	Characteristics
Flat fading	<ol style="list-style-type: none"> 1. Bandwidth of signal < Bandwidth of channel 2. Delay spread < Symbol period
Frequency Selective fading	<ol style="list-style-type: none"> 1. Bandwidth of signal > Bandwidth of channel 2. Delay spread > Symbol period
Fast fading	<ol style="list-style-type: none"> 1. High Doppler spread 2. Coherence time < Symbol period
Slow fading	<ol style="list-style-type: none"> 1. Low Doppler spread 2. Coherence time > Symbol period

Table 3.1 Types of Fading

3.4 Flat Fading

The received signal undergoes *flat fading* if the communication channel has a constant gain and a linear phase response over a bandwidth which is greater than the bandwidth of the transmitted signal. The spectral characteristics of the transmitted signal are preserved at the receiver in a flat fading channel, but the signal strength changes with time due to variations in channel gain caused by multipath. Flat fading channels are also referred to as *amplitude varying channels* or *narrowband channels*. The transmitter requires about 20 – 30 dB more power to achieve low bit error rates for these channels as compared to other systems operating over non-fading channels. The most common amplitude distribution of the instantaneous gain of the flat fading channel is the *Rayleigh*

distribution in which the channel induced amplitude varies with time according to the distribution [cf. Section 3.8].

3.5 Frequency Selective Fading

A channel is said to be *frequency selective* if the channel has a constant gain and linear phase response over a bandwidth that is smaller than the bandwidth of the transmitted signal. The impulse response of the channel has a multipath delay that is greater than the reciprocal bandwidth of the transmitted message waveform. The transmitted symbols are dispersed in time within the channel and the channel induces ISI. In the frequency domain, some frequency components in the received signal will experience greater gains than the others.

As each multipath component has to be modeled and the channel considered as a linear filter, frequency selective fading channels are more difficult to model than flat fading channels. The bandwidth of the transmitted signal spectrum is greater than the coherence bandwidth of the channel. These channels are also known as *wideband channels* since the signal bandwidth is wider than the bandwidth of the channel impulse response.

3.6 Fast Fading

A channel can be classified as fast fading or slow fading based on how rapidly the baseband signal changes compared to the rate of change of the channel. Based on that, a channel is said to be *fast fading* if the channel impulse response changes rapidly within the symbol duration, i.e., the coherence time of

the channel is less than the symbol period of the baseband signal. This results in frequency dispersion due to Doppler, which leads to distortion of the signal. In the frequency domain, the distortion spreading due to fast fading increases with increasing Doppler spread relative to the bandwidth of the transmitted signal.

A fast fading channel can also be flat or frequency selective. In a flat-fast fading channel, the amplitude factor of the delta function that approximates the channel impulse response varies faster than the rate of change of the baseband signal. On the other hand, in a frequency selective-fast fading case, the amplitudes, phases and time delays of any one of the multipaths vary rapidly compared to the rate of the baseband signal. Fast fading may be the most appropriate model for very low data rate signaling.

3.7 Slow Fading

A channel is said to be a *slow fading* channel if the channel impulse response changes at a much slower rate than the baseband signal. The channel can be assumed to be static over one of the several reciprocal bandwidth intervals. On the other hand, in the frequency domain, the Doppler spread of the channel is much less than the bandwidth of the transmitted signal.

The velocity of the receiver and/or the velocity of the objects in the channel and the baseband signaling scheme together determine the fast or slow fading characteristics of a communications channel.

3.8 Rayleigh Distribution

The statistical time-varying nature of the received envelope of a flat fading signal, or the envelope of an individual multipath component is described by the Rayleigh distribution in communication channels. The envelope of the sum of two quadrature Gaussian noise signals follows the Rayleigh distribution which is given by

$$p(r) = \begin{cases} \frac{r}{\sigma^2} \exp\left(-\frac{r^2}{2\sigma^2}\right) & (0 \leq r \leq \infty) \\ 0 & (r < 0) \end{cases} \quad (3.2)$$

where σ is the RMS value of the received signal before envelope detection, and σ^2 is the time-average power of the received signal before envelope detection.

The corresponding cumulative distribution function (CDF) is given by

$$P(R) = \Pr(r \leq R) = \int_0^R p(r) dr = 1 - \exp\left(-\frac{R^2}{2\sigma^2}\right) \quad (3.3)$$

and the mean of the Rayleigh distribution is

$$r_{mean} = E[r] = \int_0^{\infty} r p(r) dr = \sigma \sqrt{\frac{\pi}{2}} = 1.2533\sigma \quad (3.4)$$

The variance of the Rayleigh distribution is

$$\sigma_r^2 = E[r^2] - E^2[r] = \int_0^{\infty} r^2 p(r) dr - \frac{\sigma^2 \pi}{2} = \sigma^2 \left[2 - \frac{\pi}{2}\right] = 0.4292\sigma^2 \quad (3.5)$$

3.9 Effects of Fading on Multicarrier Systems

The advantage of multicarrier modulation is that each sub-channel is relatively narrowband, which mitigates the frequency selective effect of delay

spread. However, since each sub-channel experiences flat fading, which can cause large bit error rates on some sub-channels, it is important to compensate for flat fading in sub-channels. In addition to sub-channel fading, multicarrier systems also experience additive white Gaussian noise (AWGN) and Doppler spreading. Symbol synchronization and carrier recovery, in particular, can be very challenging in such cases because multipath and channel time variations can make it extremely complex to estimate signal parameters accurately prior to demodulation. As data rates increase and channels become more and more complex (e.g., Multi Input, Multi Output (MIMO) schemes), the duties of the receiver in performing timing synchronization and carrier phase recovery become even more difficult and challenging.

3.10 Fading Channel Models Used for Simulation

Several statistical fading channel models for propagation effects have been proposed over the years. The most general one is the *ray tracing* model which includes all attenuated, diffracted and scattered multipath components. Computer programs based on ray tracing that use computer graphics, combined with aerial photographs or architectural drawings of the indoor or outdoor channel to obtain a three-dimensional geometric picture of the environment, are being widely employed for system planning in both indoor and outdoor environments.

The channel models considered here are the set of Single Input, Single Output (SISO) WLAN channels developed by Medbo et al. [2] for HIPERLAN/2 from ETSI/BRAN, and a time-dispersive Rayleigh channel model used in Keller

et al. [1] for testing the synchronization algorithms they developed for OFDM systems.

3.10.1 A Time-Dispersive Rayleigh WATM Fading Channel Model

The time-dispersive Rayleigh fading channel used here for simulation purposes is an indoor channel that depicts a warehouse-type environment for a Wireless Asynchronous Transmission Mode (WATM) system operating at 60 GHz with a sample rate of 225 MHz [2]. Figure 3.2 shows the impulse response consisting of three discrete delays and the corresponding frequency domain transfer function for the channel. The greatest path delay for this channel is 48.9 ns. It is assumed that there is minimum Doppler frequency shift associated with this model.

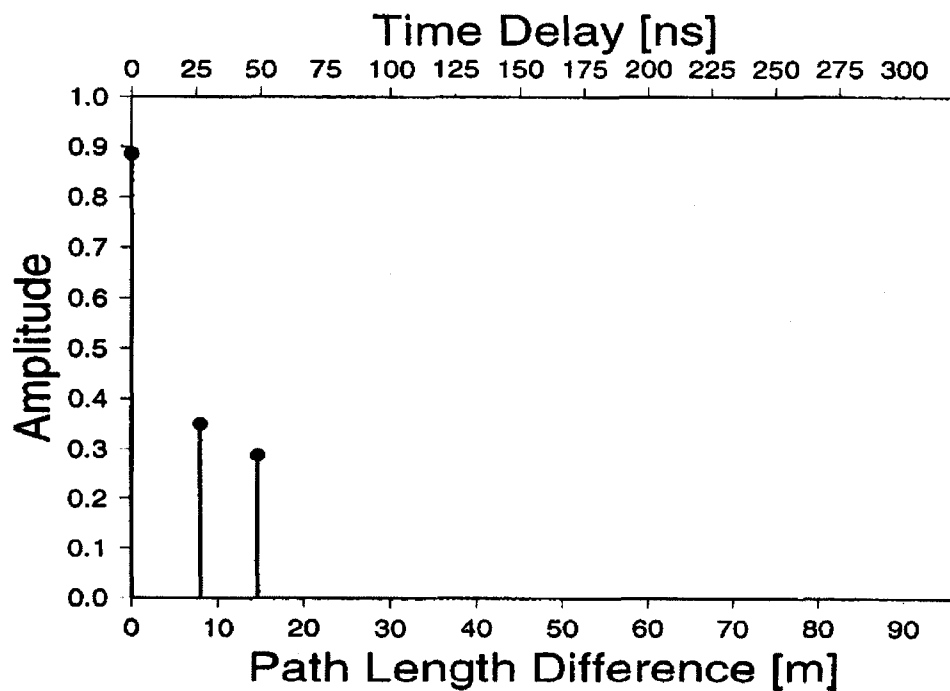


Figure 3.2 (a) Impulse response

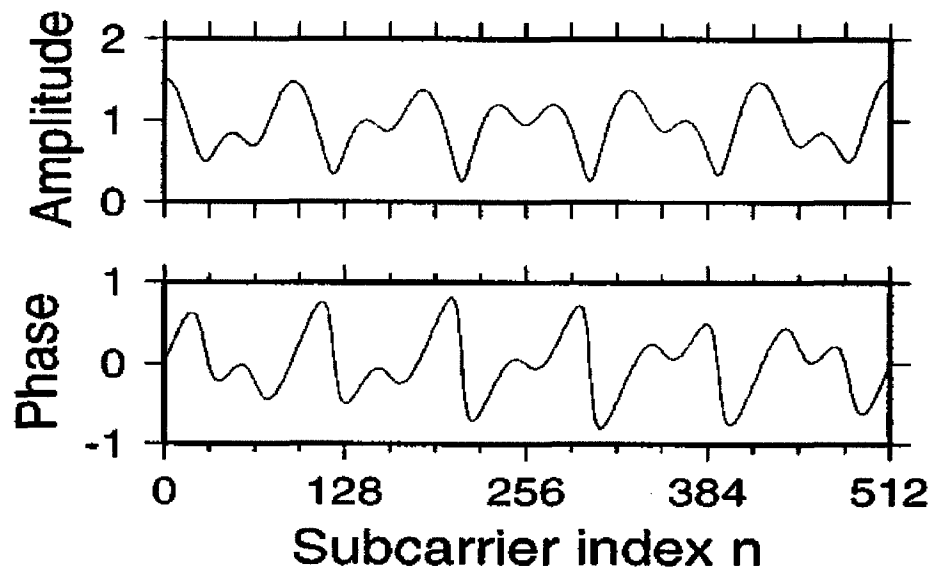


Figure 3.2 (b) Frequency domain channel transfer function

(Figures excerpted from [1], © [2001] IEEE, used with permission)

3.10.2 SISO WLAN Models from ETSI/BRAN

Medbo et al. [2] proposed five SISO WLAN models for different environments.

- *Model A* for typical office environment, non-line-of-sight (NLOS) conditions and 50 ns RMS delay spread
- *Model B* for a typical large open space and office environments, NLOS conditions and 100 ns RMS delay spread
- *Model C* for a large open space (indoor and outdoor), NLOS conditions and 150 ns RMS delay spread
- *Model D*, same as model C, LOS conditions and 140 ns RMS delay spread (This is the only Rician-like fading model used in this thesis)
- *Model E* for a typical large open space (indoor and outdoor), NLOS conditions and 250 ns RMS delay spread

Of the above mentioned five profiles, models B, C and D with their fewer numbers of taps were chosen for our simulation purposes. Table 3.2, Table 3.3

and Table 3.4 show the number of the taps, delays and the corresponding power associated with each of them for models B, C and D respectively. Figures 3.3, 3.4 and 3.5 show the power delay profiles of Models B, C and D respectively.

3.10.2.1 Model B (with five taps).

Tap Index	Excess Delay (ns)	Power (dB)
1	0	0
2	10	-5.4
3	20	-10.8
4	30	-16.2
5	40	-21.7

Table 3.2 Model B with five taps

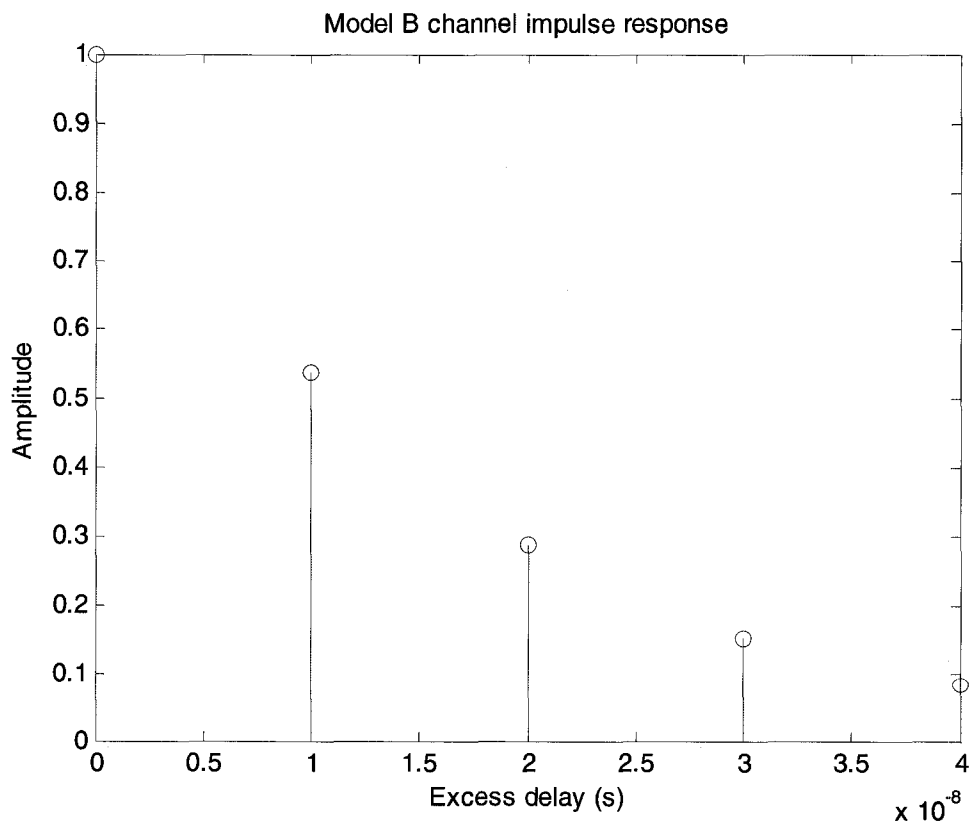


Figure 3.3 Model B impulse response

3.10.2.2 Model C (with six taps).

Tap Index	Excess Delay (ns)	Power (dB)
1	0	0
2	10	-2.1
3	20	-4.3
4	30	-6.5
5	40	-8.6
6	50	-10.8

Table 3.3 Model C with six taps

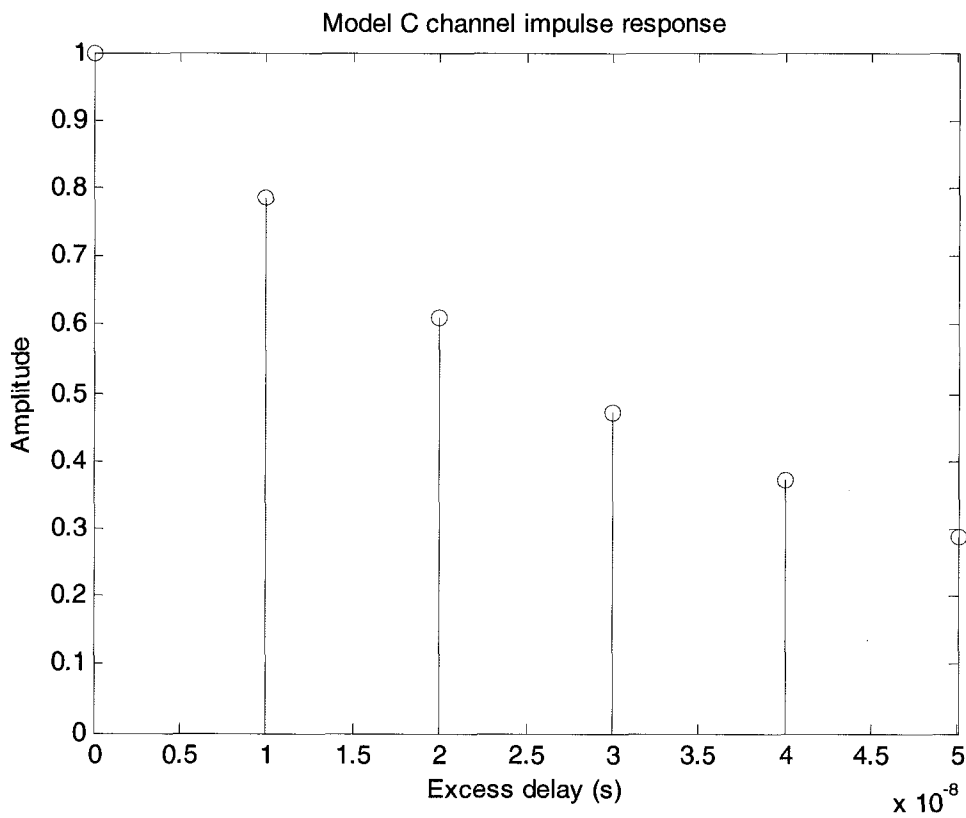


Figure 3.4 Model C impulse response

3.10.2.3 Model D (with seven taps).

Tap Index	Excess Delay (ns)	Power (dB)
1	0	0
2	10	-0.9
3	20	-1.7
4	30	-2.6
5	40	-3.5
6	50	-4.3
7	60	-5.2

Table 3.4 Model D with seven taps

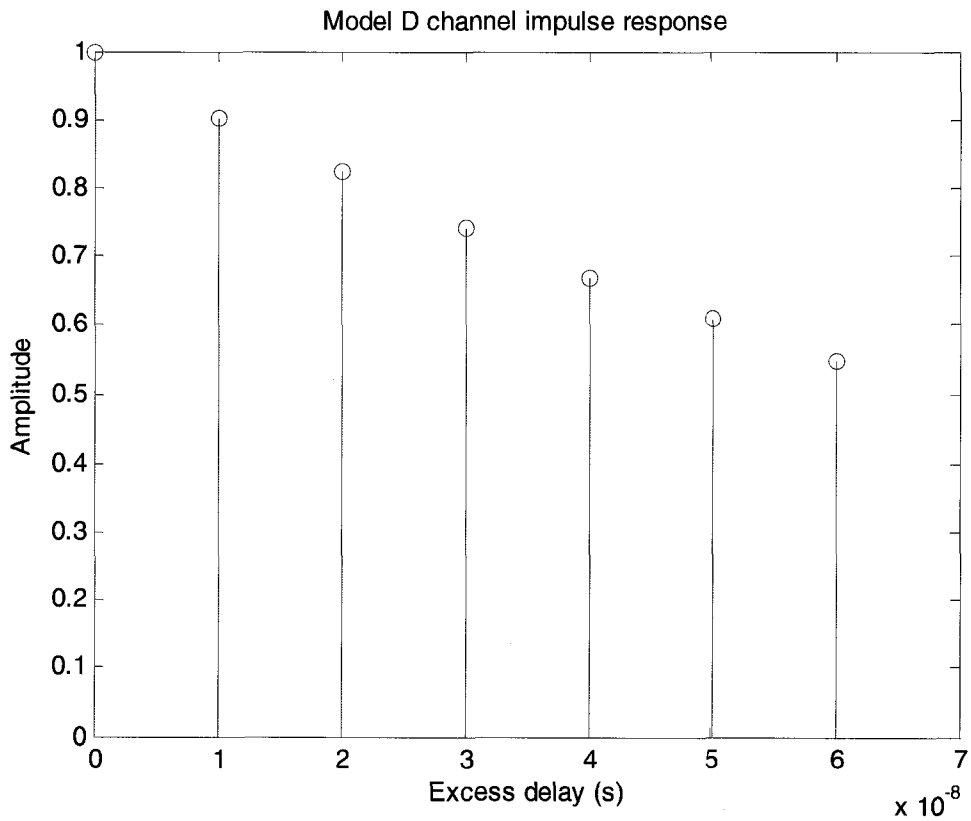


Figure 3.5 Model D impulse response

CHAPTER 4

CORRELATION-BASED SYNCHRONIZATION FOR OFDM

4.1 Introduction

As discussed in the previous chapters, timing and frequency synchronization between the transmitter and the receiver are of utmost importance for the performance of an OFDM system. Several authors have addressed this issue and have proposed a variety of techniques for the estimation and correction of time and frequency synchronization errors [7-13]. Carrier frequency synchronization and time synchronization estimation algorithms for data-aided OFDM schemes use pilot symbols or pilot tones or training sequences that are embedded into the OFDM symbols. Algorithms for the non-data aided schemes, on the other hand, make use of the redundancy introduced in the OFDM symbols by the cyclic extension.

In [7], Tureli *et al.* discuss how ESPRIT-like blind estimation algorithms make use of the information in the structure of the OFDM signal with low fixed complexity and also offer excellent performance over cyclic prefix (CP) based methods, which they have demonstrated by computing the Mean Squared Error (MSE) between the two. Hsu *et al.* [8] and Zhang *et al.* [9] exploit the pilot carriers or the pilot tones and compute the Maximum Likelihood (ML) estimate of the frequency offset. They show that the orthogonality of the carriers is maintained by using a window function which, on the other hand, also reduces

efficient utilization of the spectrum and increases the influence of additive noise. Speth *et al.* [10] explain a frame synchronization technique using interpolation, and the analysis of the results show that interpolation for coherent OFDM systems is very sensitive to frame misalignment. In [11], Meng-Han Hsieh *et al.* use a set of four smoothing algorithms based on the guard interval which maximize the log-likelihood function. Pacheco *et al.* in [13] discuss a frame synchronization method using the periodic preamble which computes the Bayesian estimate, a Maximum a Posteriori (MAP) estimator, using basic principles from Bayesian changepoint detection. This method is highly complex, though very accurate, and also has lower latency and complexity than some proposed correlation-based methods.

Although a variety of possible implementations of synchronizers other than correlation-based techniques have been proposed for OFDM, correlation-based synchronizers have their own advantages and disadvantages. Since correlation-based techniques fall under the category of non-data aided schemes, they are not quite suitable for the initial frequency acquisition because an accurate estimation usually needs an averaging over a large number of OFDM symbols, and in very high data rate transmissions, the time for initial frequency synchronization needs to be as short as possible [9]. On the other hand, the correlation-based techniques have a lot of advantages:

- Less computational complexity leading to less computational time.
- Low bandwidth, as they don't use any additional data other than the cyclic partial periodicity of the OFDM symbols as such.

4.2 Effects of Carrier Frequency and Time Synchronization Errors

In the frequency domain, the received signal is shifted as a result of carrier frequency synchronization error δf . If this error is an integral multiple M of the sub-carrier spacing Δf , then the sub-carriers are shifted by this factor M . The orthogonality is still preserved, but the mapping of the received data symbols to the sub-channels does not line up with that of the transmitted signal. In other words, the data symbols are mapped to wrong time instances, resulting in the worst case bit error rate (BER) of 0.5. On the other hand, if the frequency error δf is not an integral multiple of Δf , the sub-carriers are not mutually orthogonal as a result of the leakage of energy between them. In this case, the amount of ICI experienced by any single sub-carrier is equal to the sum of the interference amplitude contributions from all the other sub-carriers in the OFDM symbol. Additionally, the probability density of the interference amplitude of a single sub-carrier can be mathematically approximated by a Gaussian process as the OFDM symbols are also random variables. This interference amplitude decreases with the increase in sub-carrier spacing as only interferences from closely spaced sub-carriers contribute significantly to the total effect.

Unlike errors caused by frequency synchronization, time synchronization errors result in ISI as it involves misalignment of the receiver's FFT window, which gives rise to a progressive phase shift in the frequency domain symbols leading to poor BER performance of the system.

The time-shift property of the Fourier transform is given by,

$$f(t) \leftrightarrow F(\omega) \quad (4.1)$$

$$f(t - \tau) \leftrightarrow e^{-j\omega\tau} F(\omega) \quad (4.2)$$

Equation 4.2 describes the effect of time shifts in the FFT window of the receiver. Any shift τ will thereby introduce a progressive phase error $2\pi * \Delta f * \tau / T_s$ between any two adjacent sub-carriers. Now, if this time shift is an integral multiple I of the sampling time T_s , the phase shift introduced is $\delta\phi = 2\pi * I / N$. In cases where timing errors are very high, the FFT window of the receiver could even overlap the adjacent OFDM symbol to include samples from outside the current OFDM symbol, which then causes the subsequent symbols to interfere, thereby severely degrading the performance of the system. However, the phase error introduced by the time synchronization errors also changes depending on the modulation scheme employed.

4.3 OFDM Synchronization by Autocorrelation

Due to the effects outlined in section 4.1, both the time and frequency synchronization errors should be kept to a minimum for best performance of the system. In order to achieve this goal and also to reduce the redundancy in the OFDM symbols, the synchronization process itself is divided into two stages: an acquisition stage and a tracking stage. The initial synchronization error estimation is done in the acquisition phase using algorithms that utilize most of the synchronization information embedded in the OFDM symbol and compute the autocorrelation of the received samples. The tracking algorithms only have to

correct short-term variations. Here is where we take advantage of the cyclic time domain extension of the OFDM symbols.

For frame synchronization, a unique reference symbol is used. In this approach, there is no added redundancy in the data symbols and no *a priori* knowledge of the sequences used in the reference symbol. All processing is carried out in the time domain, thus eliminating the need for FFT-based demodulation prior to time and frequency offset estimation.

The algorithms that are investigated here were proposed by Keller et al. [1] and they evaluate the autocorrelation of the received complex baseband signal samples $z(j)$, with j being the index of the most recent input sample. The two correlation algorithms are given below:

$$G(j) = \sum_{m=0}^{N_g-1} z(j-m) \cdot z(j-m-N)^* \quad (4.3)$$

$$R(j) = \sum_{m=0}^{N+N_g-N_s-1} z(j-m) \cdot z(j-m-N_s)^* \quad (4.4)$$

Here, the correlation metric $G(j)$ is used for OFDM symbol synchronization and frequency tracking and $R(j)$ is used for frequency acquisition and frame synchronization.

4.3.1 OFDM Frame Structure

Periodic pseudo-random sequences with perfect or minimum autocorrelation/cross-correlation properties are widely used in spread spectrum, radar and multicarrier communication systems for synchronization purposes. In

OFDM systems, these sequences may be transmitted as part of the frame structure and can be used to determine the frame boundaries precisely. A correlation peak occurs when repeated copies of the synchronization symbol are aligned with one another in the correlation computation.

A unique algorithm for generating periodic, complex-valued, pseudo-random sequences with perfect autocorrelation properties was proposed by Alltop *et al.* [3] in 1980, according to which, for any X , an odd integer greater than 2, the λ^{th} quadratic phase sequence a_λ is given by,

$$a_\lambda(k) = X^{(-1/2)} \omega_X^{\lambda k^2} \quad (4.5)$$

where λ represents the number of users in a multiuser environment and ω_X is the X^{th} root of unity.

According to [3], for odd $X \geq 3$, P_X is a family of $p-1$ quadratic phase sequences with $\max(P_X) = X^{(-1/2)}$, where p is the smallest prime divisor of X . In particular, the correlation coefficients $A_{\lambda\mu}(\tau)$, $0 \leq \tau \leq X^{-1}$, $1 \leq \lambda, \mu \leq p-1$, satisfy

$$|A_{\lambda\mu}(\tau)| = \begin{cases} 1, & \text{for } \lambda = \mu, \tau = 0 \\ 0, & \text{for } \lambda = \mu, \tau \neq 0 \\ X^{-1/2}, & \text{otherwise} \end{cases} \quad (4.6)$$

The OFDM frame is built by putting together a null symbol followed by a reference symbol that consists of repetitive patterns of a unique periodic complex pseudo-random sequence generated by equation 4.5 for $\lambda=1$ (single user), and $X=31$. The reference symbol is then followed by the OFDM data symbols. In

our case, we make use of a frame length of 64 consisting of a null symbol, a reference symbol with synchronization patterns that are 31 samples in length (less than that of the OFDM length which is 128 samples), and a cyclic extension of 17 samples as opposed to 13 samples for the regular data symbols, and 62 OFDM data symbols. All in all, the time and frequency synchronization algorithms studied here make use of 3 hierarchical periodic time-domain structures, namely: the short-term periodicity in the reference symbol, the medium-term periodicity in the cyclic extension, and the long-term periodicity in the frame structure. The cyclic extension is used for OFDM symbol synchronization and frequency tracking while the reference symbol is used for frame synchronization and frequency acquisition.

The correlation metric $G(j)$ works on sequences that are spaced one OFDM symbol apart and computes the partial autocorrelation for two sequences that are of the length of the cyclic extension for the data samples, N_g . As a result, $|G(j)|$ peaks at the OFDM symbol boundaries when the cyclic extension coincides with its replica in the tail of the OFDM symbol. The second correlation metric $R(j)$ works on sequences that are spaced by the length of the synchronization pattern, N_s and computes the partial autocorrelation between two sequences of length $(N + N_g - N_s - 1)$. The peaks of this metric occur at the frame boundaries as the reference symbol coincides with itself at these points.

4.3.2 Frequency Tracking and OFDM Symbol Synchronization

The magnitude of $G(j_{\max})$ is maximum if j_{\max} is the index of the last sample $z(j_{\max})$ of the current OFDM symbol. This would show that the cyclic extension samples and their copies in the current OFDM symbol are perfectly aligned in the autocorrelation summation window. However, the magnitude of $G(j)$, $|G(j)|$, is not constant as the samples in the cyclic extension differ on a symbol to symbol basis, and hence is dependent on the energy in the cyclic extension.

Since even a slight misalignment in the time-domain FFT windows causes phase errors in the frequency domain, the phase error $\psi(j)$ associated with a received sample $z(j)$ for a carrier frequency offset of δf is given by,

$$\psi(j) = 2\pi \cdot \delta f \cdot j \cdot T_s = 2\pi \frac{\delta f \cdot j}{N \cdot \Delta f} \quad (4.8)$$

From equation 4.8, it is clear that the phase error between any two received samples is a function of the frequency error and their relative time spacing. So, if the original phase difference between two symbols is known and assuming that all other sources of phase difference are negligible, the frequency error δf could be determined unambiguously.

The phase of $G(j_{\max})$ is equal to the average of the phase shifts between the cyclic prefix (guard interval) samples and their copies in the current OFDM symbol. Since the copies of the cyclic prefix samples in the data are N samples apart, the fine frequency tracking error estimate is given by

$$\delta f_t = \frac{\Delta f}{2\pi} \angle G(j_{\max}) \quad (4.9)$$

and this error δf_t should be smaller than half of the sub-carrier spacing, $\Delta f/2$. This requires the frequency acquisition algorithm to produce an initial estimate that is accurate to $\Delta f/2$ or better.

4.3.3 Frequency Acquisition and OFDM Frame Synchronization

As noted earlier, the autocorrelation metric $R(j)$ in equation 4.4 uses the periodicity in the reference symbol, the unique complex synchronization pattern of length N_s , to improve the acquisition range. Similar to the symbol timing synchronization based on $|G(j)|$, the magnitude of $R(j)$ is maximum when the synchronization patterns in the reference symbol perfectly align with one another. Along with the peaks that come out of the correlator $G(j)$, the peaks of $R(j)$ can also be used to assist in OFDM timing and frequency synchronization.

Similar to the frequency tracking estimate described in section 4.2.2, the frequency acquisition process also uses the same principle of using the phase information of $R(j)$ at the last sample of the reference symbol with index j_{\max} to compute the frequency acquisition error estimate as follows:

$$\angle R(j_{\max}) = 2\pi \cdot \delta f_a \cdot N_s \cdot T_s = 2\pi \cdot \delta f_a \frac{N_s}{N \cdot \Delta f} \quad (4.10)$$

$$\therefore \delta f_a = \frac{N}{N_s} \frac{\Delta f}{2\pi} \angle R(j_{\max}) \quad (4.11)$$

The frequency acquisition error estimate given by equation 4.11 can then be used by the frequency tracking algorithm to support fine tracking. Also, it

should be noted here that the maximum frequency error that can be unambiguously detected is increased by a factor N/N_s as the spacing between the samples that go into the computation of $R(j)$ is smaller than that of $G(j)$.

4.4 Estimation of Energy in the Tail of the OFDM Symbols

Given that the incoming binary data stream is random in nature, an estimation of the energy in the cyclic extension, also called *tail energy* in our terminology, was done to account for the variability in the estimated frequency and timing offsets each time the simulations were run in MATLAB. A histogram of the result for 512 symbols was obtained as shown in Figure 4.1.

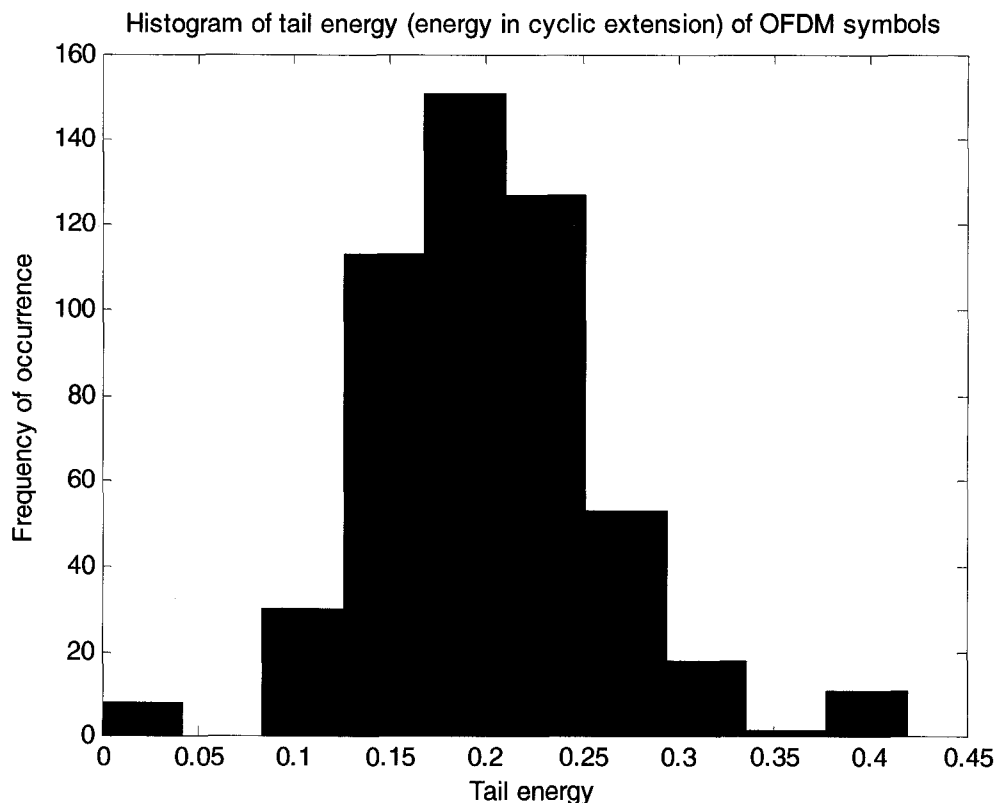


Figure 4.1 Histogram of energy in the cyclic extension for each OFDM symbol

From the above histogram, it is clearly seen that the energy in the cyclic extension is not the same for every OFDM symbol. Out of the 512 symbols that were taken into consideration, the 8 null symbols do not have any energy, the 8 reference symbols should have the same energy in their cyclic extension as the reference symbols were the same for every frame, but the remaining 492 data symbols have different energies going into their cyclic extension. This fact was verified by analyzing the actual data file of the output and it does not make the peaks at the symbol boundaries stand out from the other spurious peaks that show as a result of partial correlations. It becomes difficult and challenging to use the correlation metrics described in the previous sections, estimate the timing and frequency errors, and confidently decide to go with a peak index. A few techniques are suggested to help in the decision making process and they are outlined in section 4.5.

4.5 Real-Time Versus Non-Real Time Estimators

This section discusses the issues working with real-time and non-real time estimators i.e., estimators that run sequentially and those that involve making a delayed decision. The advantage of making a delayed decision over running sequential tests is greater accuracy in determining the correlation peak location.

4.5.1 Delayed Decision Versus Sequential Tests

The experiments run for the purpose of this thesis are all based on delayed decision as they seemed to give a better understanding of the structure of the correlator outputs when discriminating a peak.

4.5.1.1 Latency

A simple peak detector when used for peak discrimination has lower latency, but greater probability of false lock. The decision on locating the correct peak cannot be made until we have received a sufficient number of OFDM symbols, segmented and averaged the correlator outputs, and applied the corresponding offsets. A buffer must be filled with sufficient number of OFDM symbols before starting to analyze the data. Once again, this process involves a considerable delay before collecting the sufficient number of OFDM symbols which could cause the channel characteristics to change completely during that wait time.

4.5.1.2 Averaging

Averaging, a form of low-pass filtering, is a technique that is commonly used with time and frequency synchronizers to reduce the magnitude of the unwanted correlation peaks. The received OFDM symbols collected in the buffer can be subjected to several averaging techniques to sharpen the correlation peaks around the symbol and frame boundaries. This will not be possible when running sequential tests except for when a windowed average is used which will not involve huge wait times before the data gets collected. Chapter 5 describes in detail some of the averaging techniques that were used during the development of this thesis.

4.5.1.3 Peak Detection Threshold

A threshold could be set to eliminate the spurious peaks from the real peaks to simplify the decision making process. Setting a threshold is very

important as a very low threshold would bring in all the unwanted peaks as opposed to a very high threshold which would eliminate many useful peaks. An optimum value of threshold has to be set which would reasonably eliminate all the spurious peaks as well as serve to include most, if not all, of the real peaks that occur at the symbol and frame boundaries. Owing to the randomness associated with the input binary data stream, it is impossible to predict the incoming pattern and a fixed threshold would never work well under these circumstances as the magnitudes of the correlator peaks would keep changing depending on the distribution of the binary data in each sub-channel. The only workaround for this would be to implement a dynamic threshold setting algorithm which automatically sets an optimum threshold value based on the magnitudes of the correlator peaks. This process of peak detection using a threshold is only possible with a delay in processing the received symbols.

4.5.1.4 Simple Peak-Picking Algorithm

An alternative to the threshold detection process would be to use a simple peak-picking algorithm which sequentially scans each correlator peak and stores the peaks with the highest magnitude along with the corresponding peak index to be analyzed later during the decision process. The following steps explain the algorithm in detail.

1. Initialize a $n \times 2$ matrix A .
2. Pass $|G(j)|$, $|R(j)|$ through the peak-picking algorithm.
3. Find $\max |G(j)|$, $\max |R(j)|$.

4. Store $\max |G(j)|$, index of $\max |G(j)|$, $\max |R(j)|$ and index of $\max |R(j)|$ in A . Increase the size of A dynamically depending on the number of peaks.
5. Repeat Steps 3 and 4 until the entire sequence of $|G(j)|$ and $|R(j)|$ has been scanned for max values.
6. Analyze A to estimate the offsets.

Though this algorithm could work while running sequential tests, it could also be used with delayed decisions to pick the peak with maximum height from the set of peaks filtered using the detection threshold.

CHAPTER 5

CORRELATION STATISTICS EXPERIMENTS

5.1 Introduction

In order to collect various statistics to study the behavior of false lock probability at high SNR as discussed in [1], simulation experiments were run using MATLAB. A total of 128 sub-carriers were taken into account with QPSK modulation (2 bits per channel) on each sub-channel. The resulting OFDM symbols were cyclically extended and grouped into frames as explained in Chapter 2. As a result of employing a 10 % (approximately 13 samples) cyclic extension, the OFDM symbol length increased to 141 samples. The simulation experiments were carried out for a total of 496 OFDM symbols, which in turn were grouped into 8 frames by inserting a null symbol and a reference symbol at the start of each frame. Figure 5.1 below shows the scatter plot of the signal constellation with the in-phase and quadrature components.

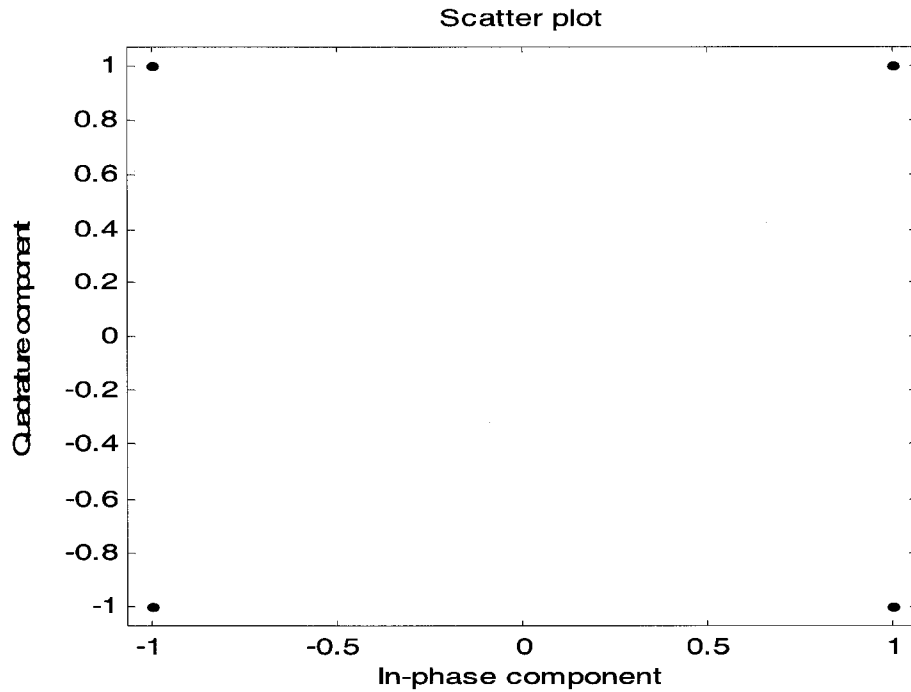


Figure 5.1 Scatter plot of the signal constellation

5.2 Raw Correlator Magnitudes

The frames generated by the OFDM sequence generation process were passed through a correlator algorithm $G(j)$ from [1] represented by equation 4.4. The fine frequency and timing offset estimates are derived from $G(j)$ at selected times corresponding to peak magnitude correlator outputs [1]. The $G(j)$ correlator takes advantage of the cyclic time domain extension of the OFDM symbols and hence it peaks in magnitude when the sliding window content coincides with the tail of the OFDM symbol. The magnitudes in the following graphs represent the corresponding a_k values from Equation 2.1. Figure 5.2 shows the continuous plot of correlator algorithm $G(j)$ output (magnitude only) for 8 OFDM frames, i.e., 496 OFDM symbols.

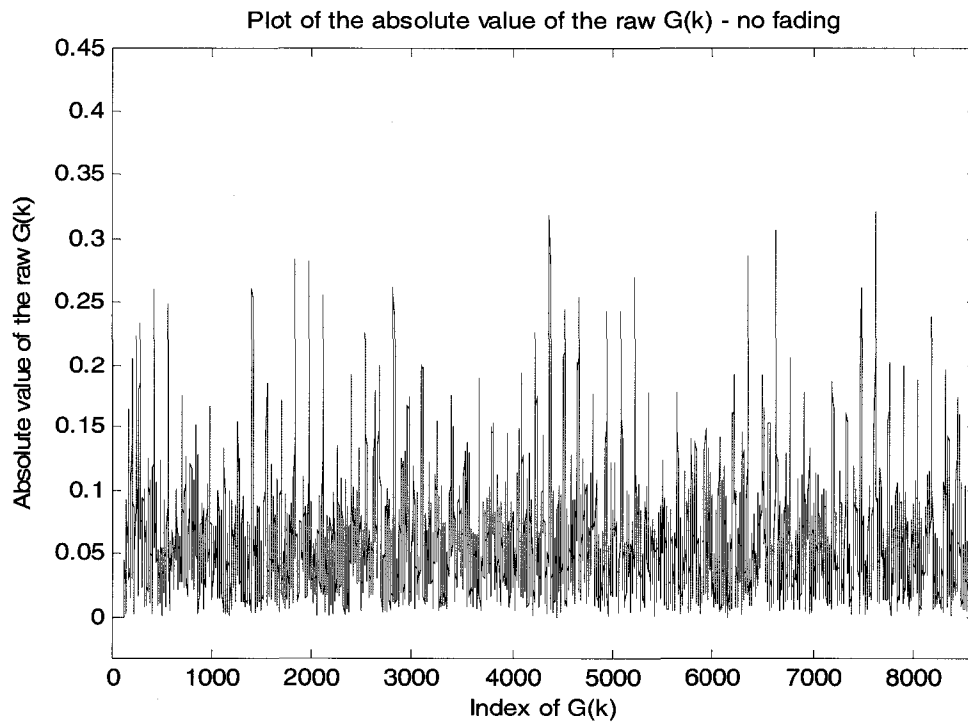


Figure 5.2 (a) Continuous plot of the output of $G(j)$

Figure 5.2 shows the structure of the raw symbol correlator output and the output around the first symbol (null symbol) of the OFDM frame sequence.

Plot of the absolute value of the raw $G(k)$ - no fading (expanded to show the correlation peaks)

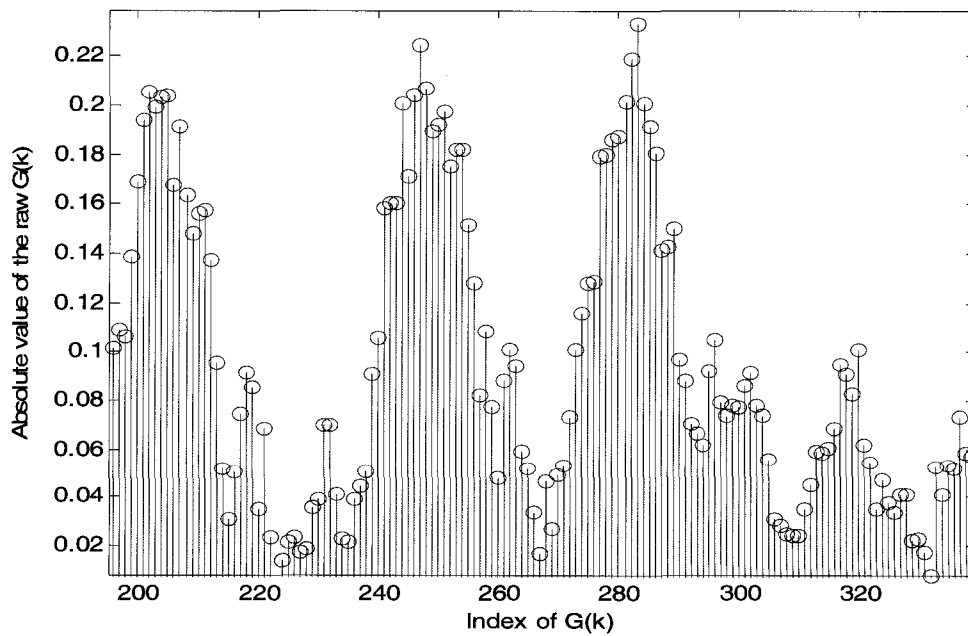


Figure 5.2 (b) $G(j)$ correlator output expanded

Plot of the absolute value of the raw $G(k)$ (no fading) - samples around the 1st and 2nd symbol

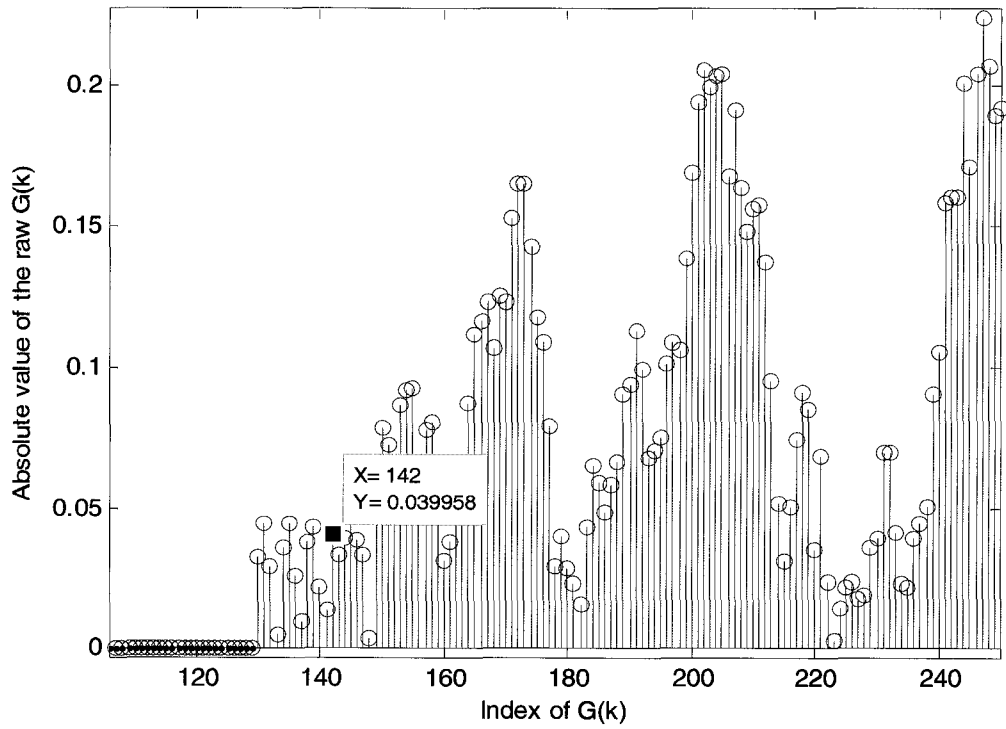


Figure 5.3 (a) $G(j)$ output near the null symbol

Plot of the absolute value of the raw $G(k)$ (no fading) with markers at the symbol boundaries

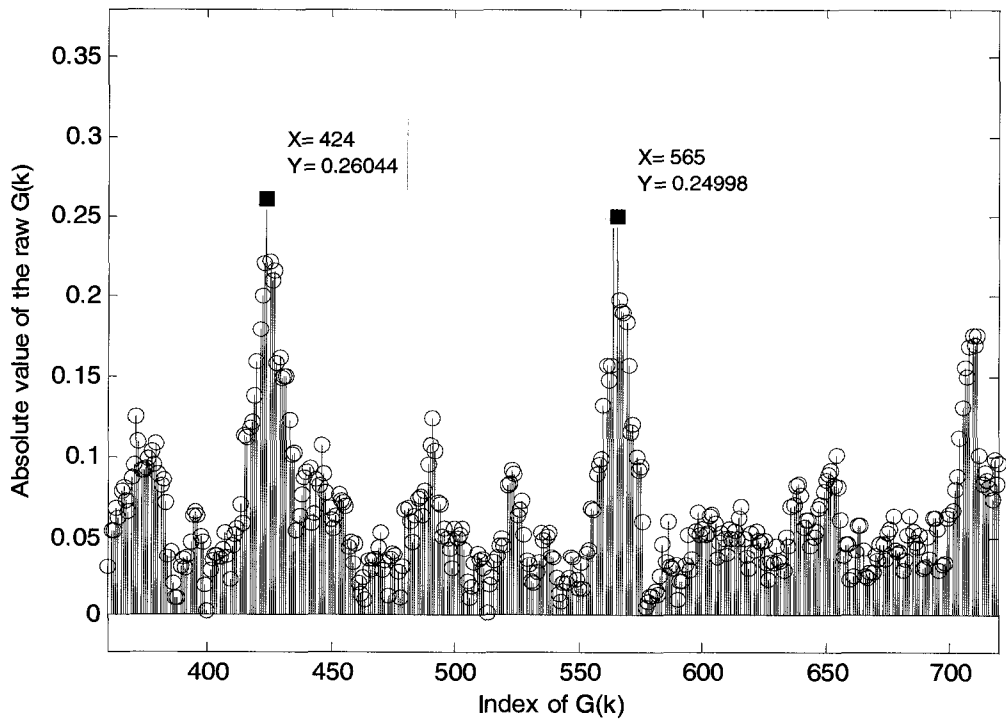


Figure 5.3 (b) Expanded $G(j)$ output with markers

As observed from figure 5.3, the relative energies of the correlator peaks around the frame and symbol boundaries are not very high to be able to distinguish them from the peaks due to partial correlation. This is because there is only one chance for the correlator to perfectly align with its copy from the symbol boundary. Additionally, the absolute value of the floor of the correlator output is quite high, and this fact makes it difficult to set a threshold and make a peak location decision in real time. Decision making posed to be a serious issue at this point as setting a high threshold would eliminate the useful peaks required to determine the OFDM symbol boundaries. A new idea of computing the ensemble average of several correlator output sequences was proposed, which significantly reduced the noise floor.

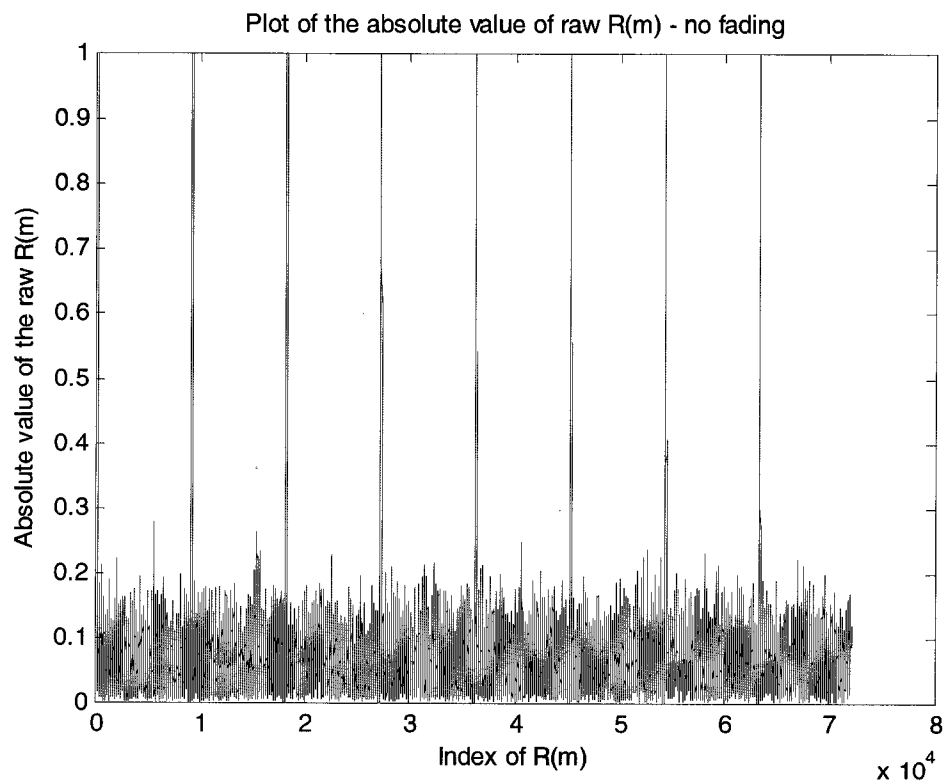


Figure 5.4 (a) Continuous plot of the frame synchronizer output $R(j)$

Plot of the absolute value of the raw $R(m)$ - no fading with a marker at first frame boundary

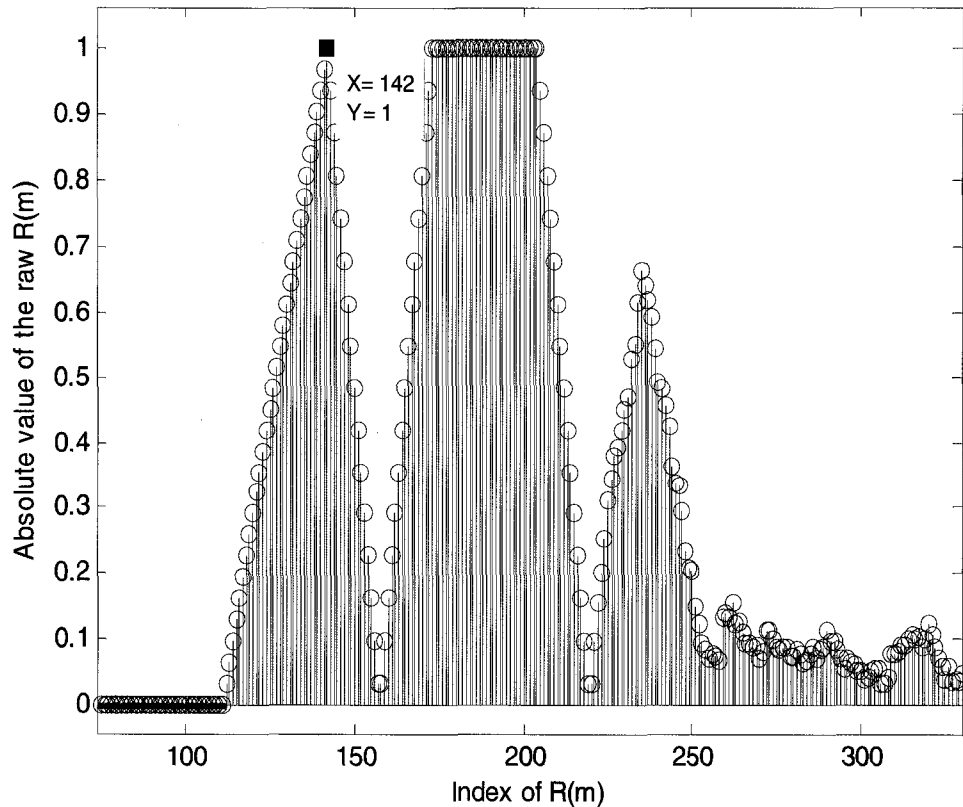


Figure 5.4 (b) Expanded $R(j)$ output around the first frame boundary

Figure 5.4 shows the output of the frame-level correlator algorithm $R(j)$.

Since the frame synchronizer is designed to look for the synchronization pattern at the beginning of each frame, the first highest peak appears at the very beginning of every reference symbol with a magnitude of unity as the synchronization pattern is bound to unity and is not affected by the threshold or averaging. Nevertheless, this highest peak at the frame boundary is not a distinct peak by itself but has a sidelobe structure to it. Also, an equal amplitude plateau with a height of 1 follows this shortly thereafter with the amplitude of the plateau being equal to the amplitude of the peak at the frame boundary. It might be possible to detect the peak at the frame boundary by a simple peak-picking

algorithm [Chapter 4] or by detecting the first peak with an amplitude of unity, but the situation might not hold good when fading is introduced.

5.3 Ensemble Statistics

Though the raw correlator outputs gave some valuable insights into the actual structure of the peak magnitudes, it was difficult to discriminate the real peaks at the symbol and frame boundaries from the spurious nearby peaks. Most importantly, a sample autocorrelation from a single realization of the random process is not an accurate reflection of the average across many realizations. This urged the use of some averaging technique to suppress the undesired peaks in order that the structure becomes more understandable for detailed analysis.

A total of 20 different OFDM sequences were generated in order to estimate an ensemble average. The average of the 20 correlator output sequences was then computed and plotted to determine the performance of ensemble averaging of correlator output magnitudes in reducing correlator sidelobe noise. The setting of a threshold depends on discriminating the true correlator peaks from spurious nearby peaks. Figure 5.4 below shows the output after computing the ensemble average of 20 different $G(j)$ output sequences. Equation 5.1 represents the ensemble average of M different $G(j)$ output sequences.

$$\text{Avg } (G(j)) = \frac{1}{M} \sum_{n=0}^{M-1} G(j + nl) \quad (5.1)$$

where l is the symbol length and M is the averaging factor.

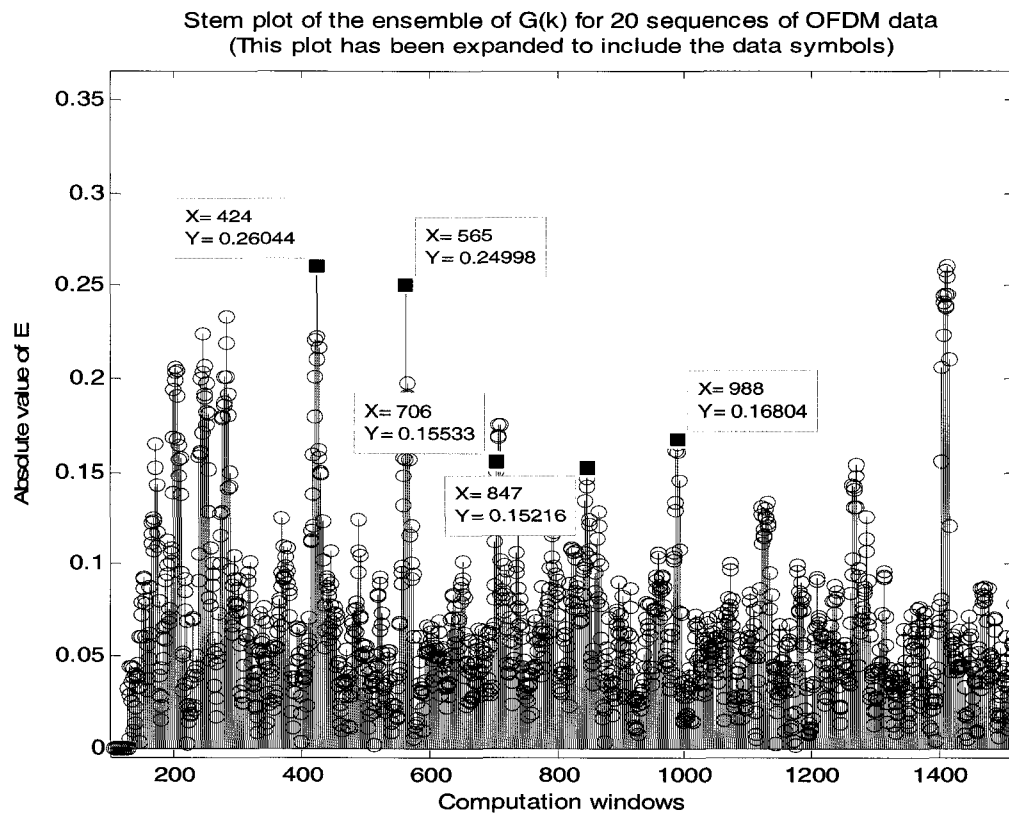


Figure 5.5 Ensemble average of 20 $G(j)$ sequences

From Figure 5.5, it is seen that ensemble averaging has considerably evened out the floor of $G(j)$ and has also sharpened the peak indices to be consistent with the symbol boundaries. The threshold selection process could be made easier and efficient by implementing the ensemble averaging as it is a filtering process in its own way. However, the penalty for easier peak discrimination against a fixed threshold detector is latency. The decision on peak location cannot be made until a sufficient number of OFDM symbols have been received, the correlator outputs are segmented and averaged, and the appropriate offsets applied. We just have to keep in mind that the situation will not be the same once fading is introduced.

CHAPTER 6

IMPROVING CORRELATOR OUTPUT USEFULNESS

6.1 Introduction

In order to get the maximum information about the channel characteristics from the correlator outputs, a series of experiments were conducted on the output of the symbol synchronizer $G(j)$. This chapter explains the experiments themselves and the interpretation of the experimental results.

6.2 Interpeak Intervals Above Threshold

From Figure 5.5, it can be seen that the floor of $G(j)$ has been made more uniform due to ensemble averaging. The output of the ensemble averaging was taken to set various thresholds and determine the peaks above the set threshold. Using a simple peak-picking algorithm [Chapter 5] which keeps track of the number of peaks above the desired threshold and the interpeak interval in terms of the number of samples between any two adjacent peaks above the threshold, the histogram of the interpeak interval versus the frequency of occurrence of the peaks corresponding to each interpeak interval was plotted as shown in Figure 6.1. By repeating the same process for different numbers of trials and different threshold values, it was concluded, as said before in Chapter 4, that the randomness in the incoming serial data causes the magnitude of the

ensemble average of the $G(j)$ correlator to vary significantly. This caused the magnitude of the peaks at the symbol boundaries to go below the threshold and hence go undetected by the peak-picking algorithm. Some useful peaks were effectively eliminated in this way. Setting a very low threshold did not work either as the spurious peaks produced by the peak detector at non-symbol boundary conditions yield faulty results. Though this experiment involved some difficulty in setting a threshold and picking the right peaks, it gave good insights about the behavior of the correlator $G(j)$ and the effect of the randomness in the incoming binary data stream.

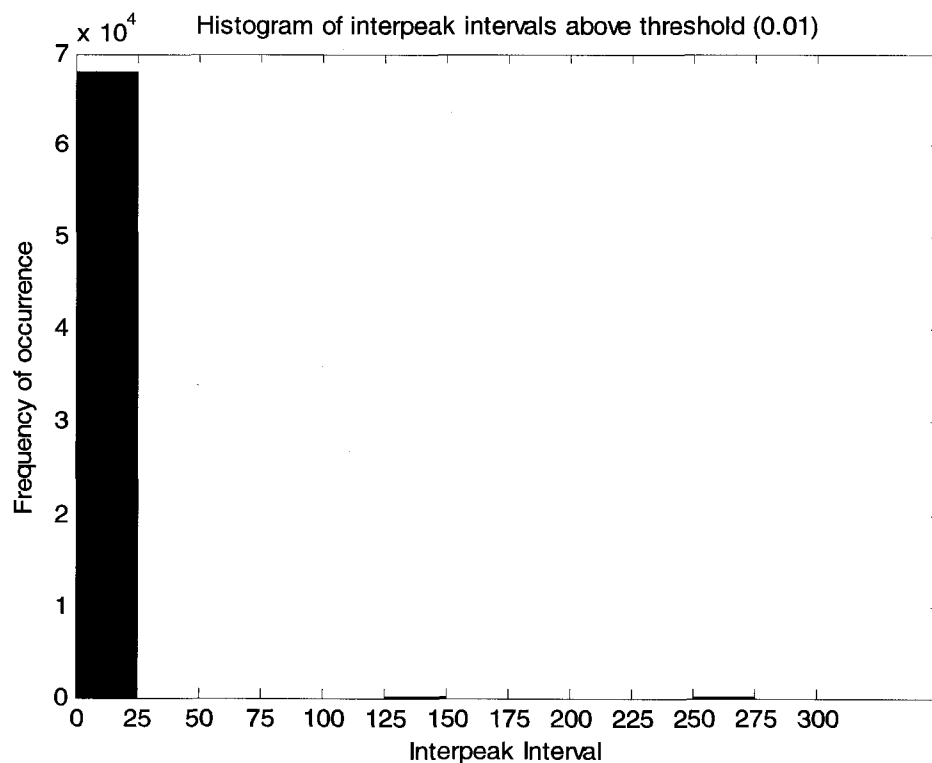


Figure 6.1 (a) Histogram of interpeak intervals above threshold = 0.01

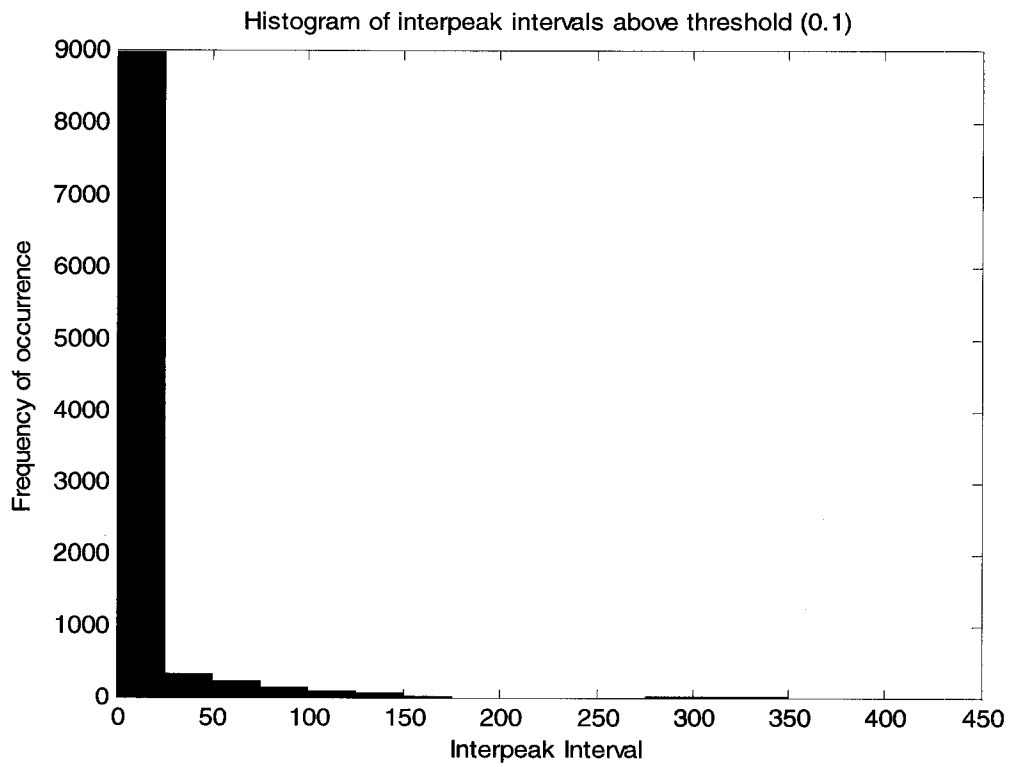


Figure 6.1 (b) Histogram of interpeak intervals above threshold = 0.1

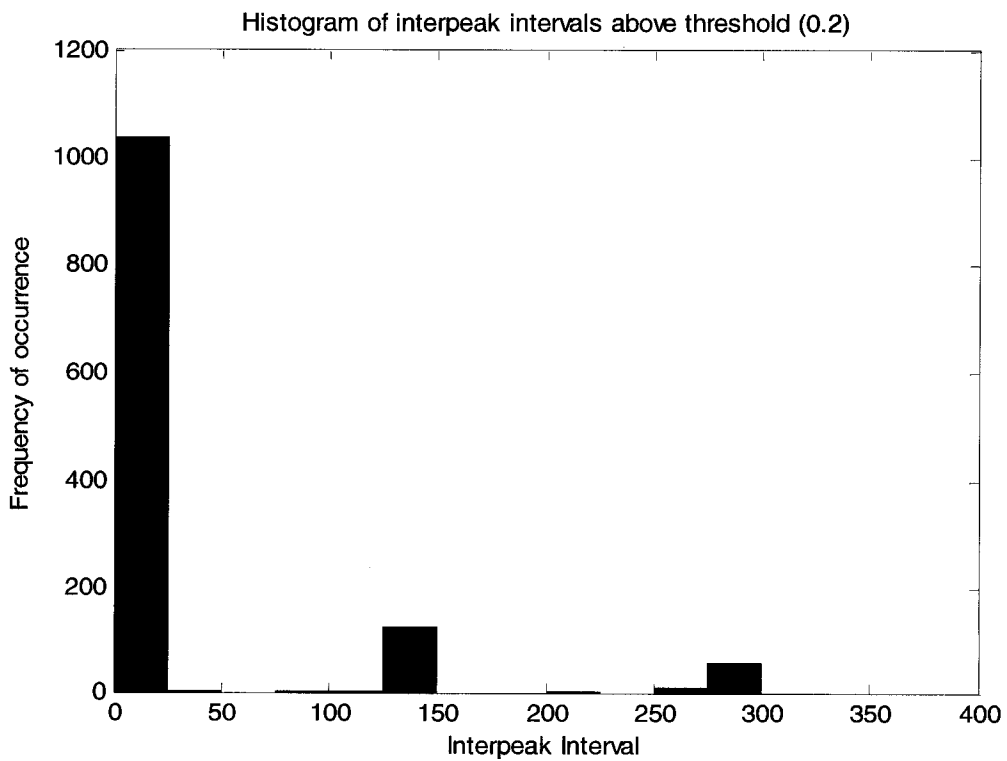


Figure 6.1 (c) Histogram of interpeak intervals above threshold = 0.2

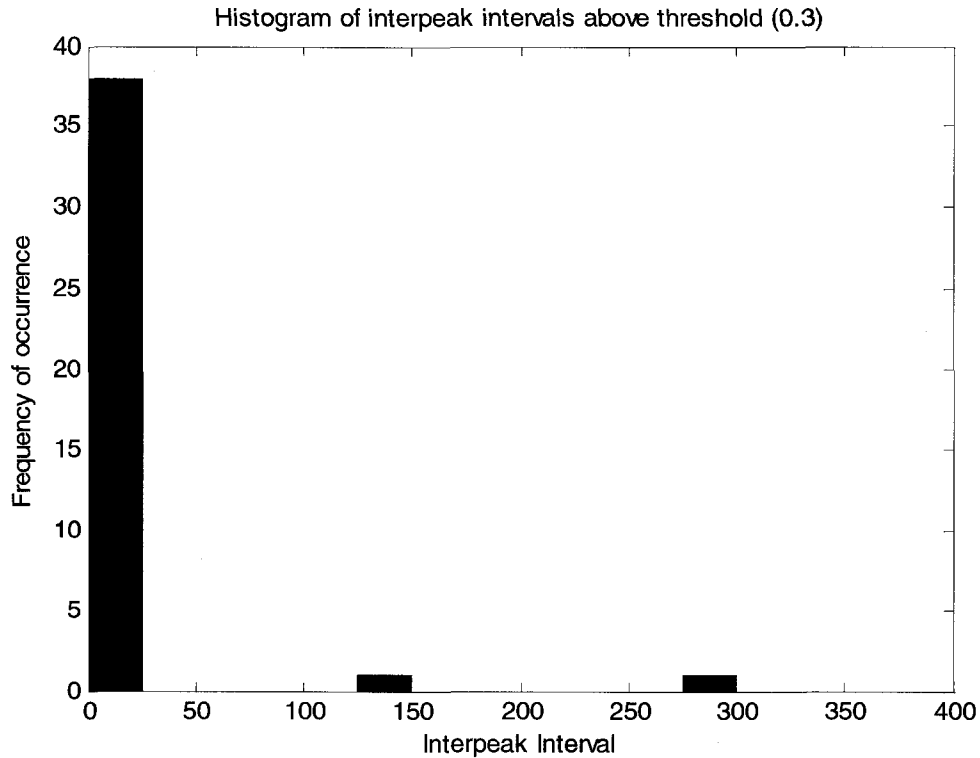


Figure 6.1 (d) Histogram of interpeak intervals above threshold = 0.3

From Figure 6.1 (a), it is clear that there are not many peaks that are one symbol length apart for a threshold as low as 0.01. The maximum number of peaks is for an interpeak interval of 25 which clearly not the distance between peaks separated by the OFDM symbol length. When the threshold is increased, the interpeak interval also increases as seen from Figures 6.1 (b), 6.1 (c) and 6.1 (d), but still does not get to the point of getting the maximum number of peaks for an interpeak interval of 141. For a threshold of 0.1, the frequency of peaks for an interpeak interval close to 150 is more than for any other demonstrated values of set thresholds. When the threshold is as high as 0.3, even most of the useful peaks are eliminated from the detection process. Under ideal conditions one would expect a histogram with a maximum frequency of occurrence equal to the

total number of OFDM symbols taken into account for a corresponding interpeak interval equal to the OFDM symbol length.

6.3 Averaging with Variable Window Sizes

Another simulation experiment performed to improve the usefulness of the raw $G(j)$ correlator outputs involved computing a windowed average of such outputs by taking into account windows of variable lengths (4 OFDM symbol lengths, 6 OFDM symbol lengths, 8 OFDM symbol lengths etc.).

Let A be the window size i.e., the number of symbols over which the windowed average is to be computed. A buffer is first loaded with samples that correspond to the first $(A*(N+N_g))$ samples of $G(j)$. The sum of the first sample of every symbol in the buffer is calculated and is averaged over A . The buffer is then shifted left by one sample to include the next $(A*(N+N_g))$ samples and the sum of the second sample of every symbol is calculated. This process is repeated for A number of times and the result is plotted to see the effect of windowed averaging. From the figures below it would be seen that this averaging technique evens out the peaks to a greater extent which was directly proportional to the window size. The larger the number of symbols taken into account for computing the windowed average, the more even the output was.

The samples that resulted from windowed averaging were subjected to more averaging over one OFDM symbol length by just computing the sum of the respective samples of the symbols in the windowed average. The plot resulting

from computing the average of the averaged $G(j)$ correlator output still shows significant sidelobe structure near the OFDM symbol boundaries with peak heights very close to the desired peak. Equation 6.1 below represents the windowed average of $G(j)$ correlator output for a window size of K .

$$T(j) = \sum_{m=0}^{K-1} G(j - m(N + N_g)) \quad (6.1)$$

6.3.1 Averaging with a Window Size of 4

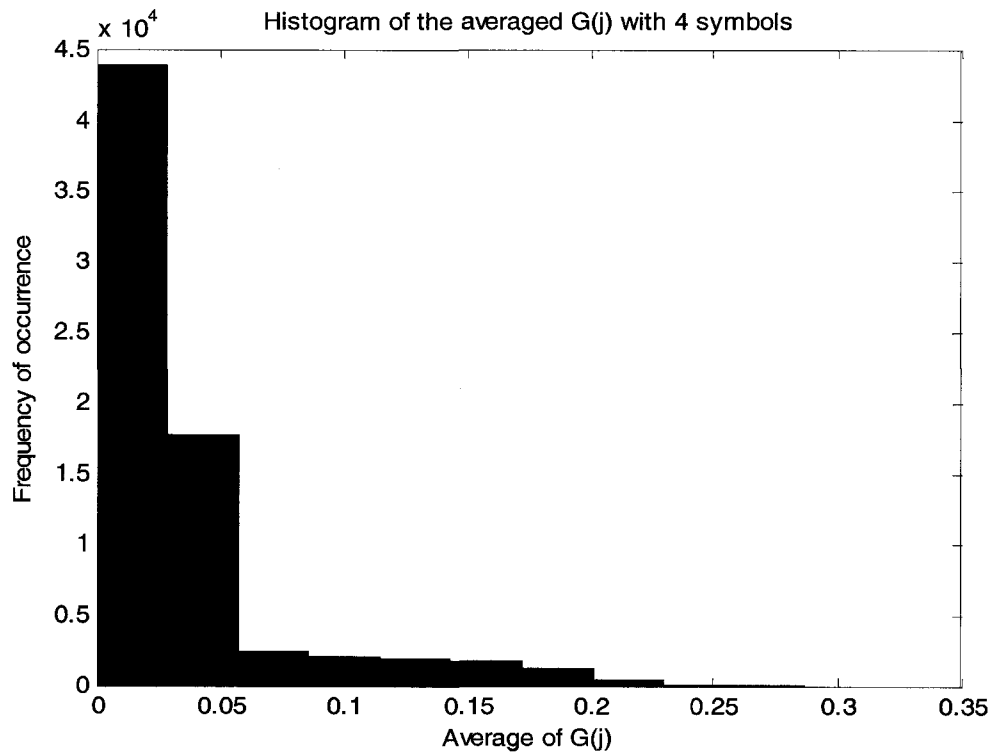


Figure 6.2 (a) Histogram of windowed average for 4 symbols

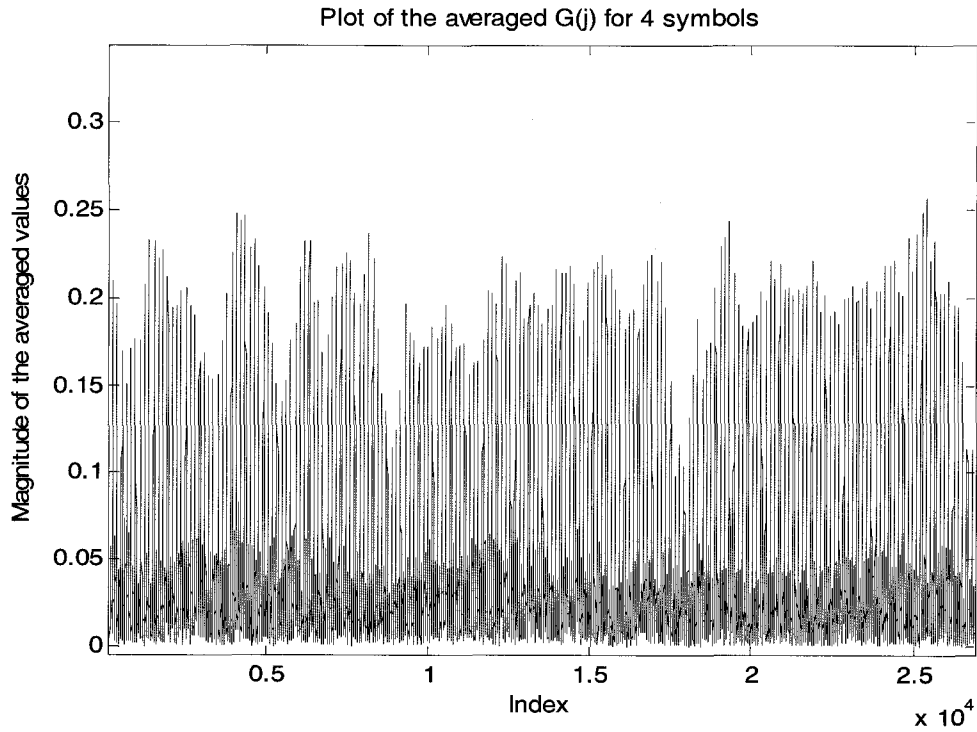


Figure 6.2 (b) Plot of windowed average for 4 symbols

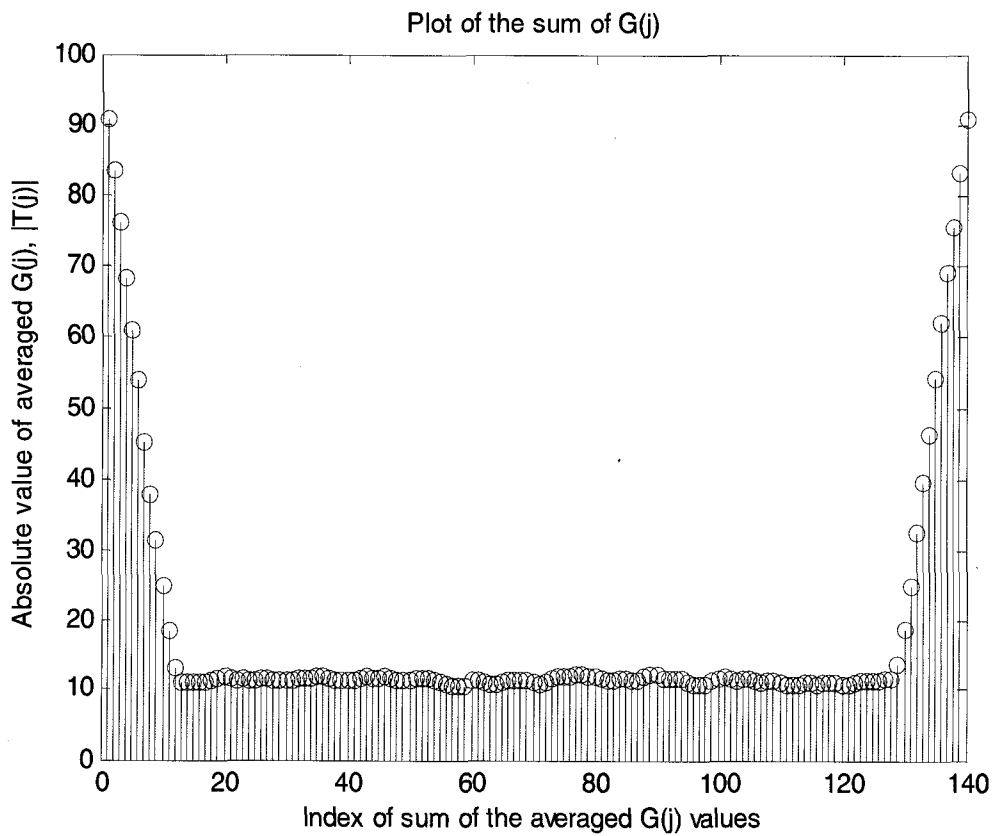


Figure 6.2 (c) Sum of windowed average with 4 symbols

6.3.2 Averaging with a Window Size of 6

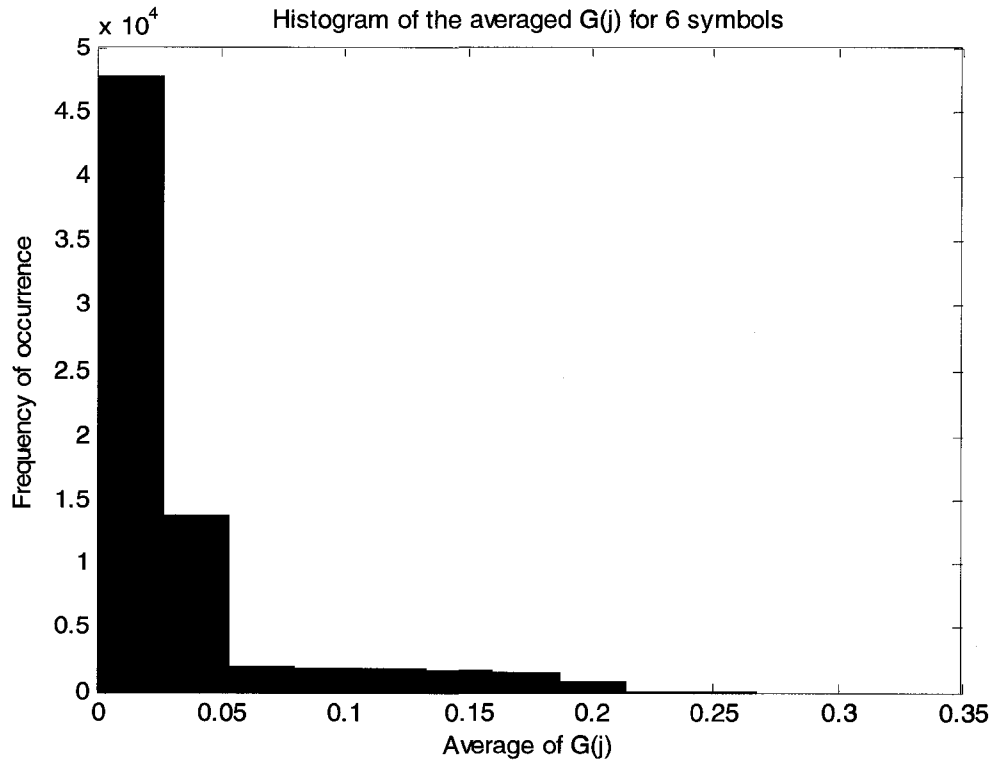


Figure 6.3 (a) Histogram of windowed average for 6 symbols

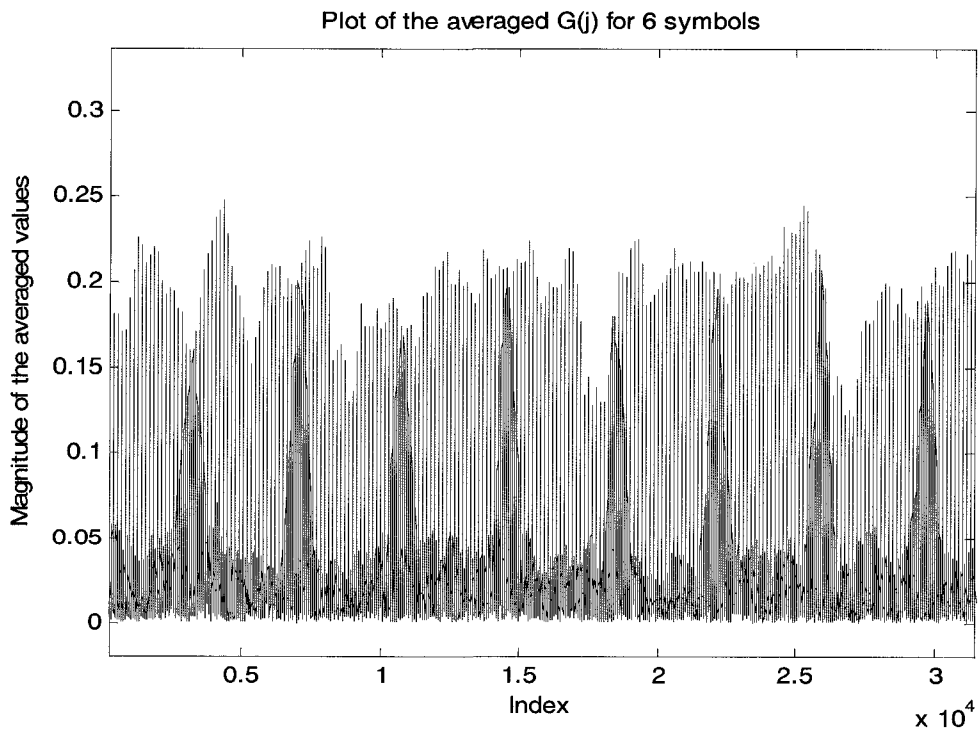


Figure 6.3 (b) Plot of windowed average for 6 symbols

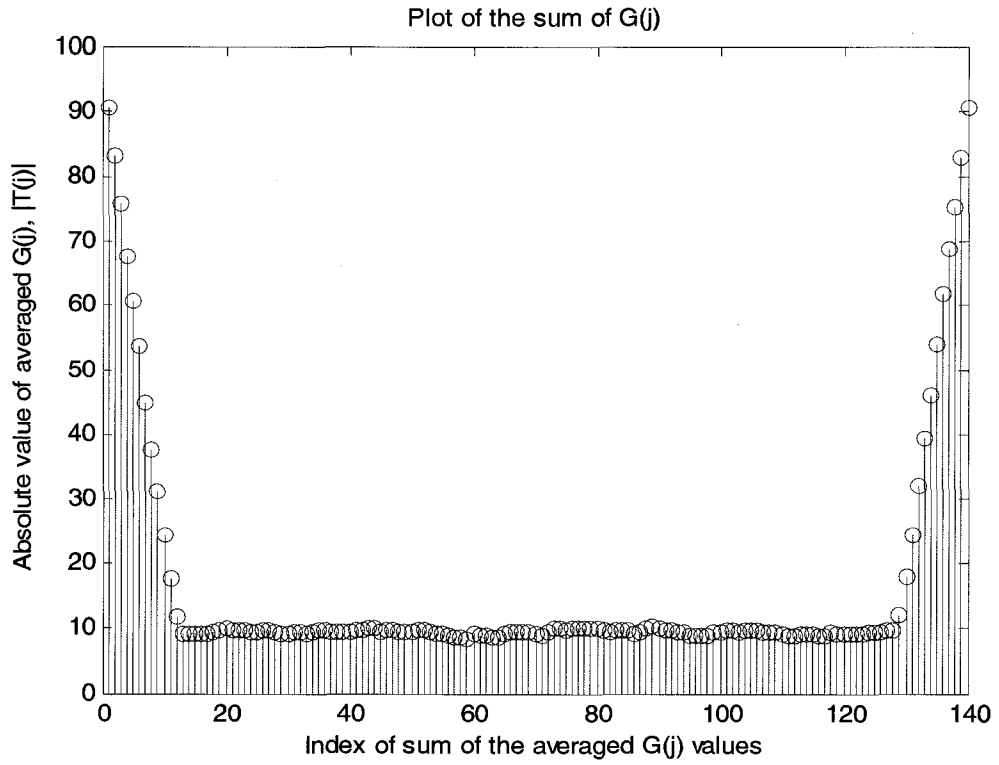


Figure 6.3 (c) Sum of windowed average with 6 symbols

6.3.3 Averaging with a Window Size of 8

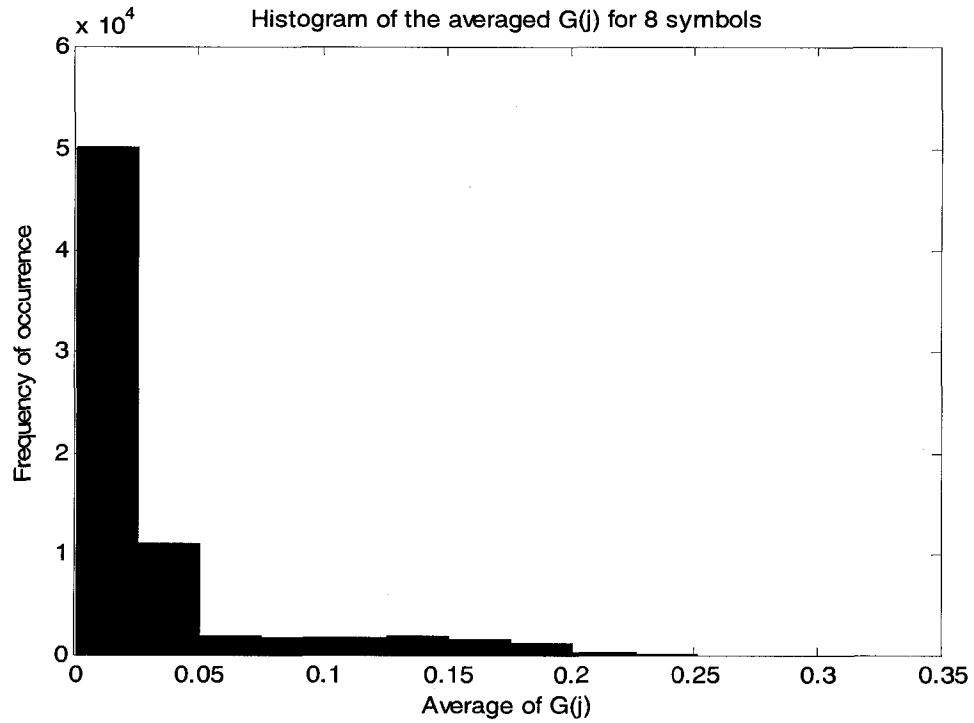


Figure 6.4 (a) Histogram of windowed average for 8 symbols

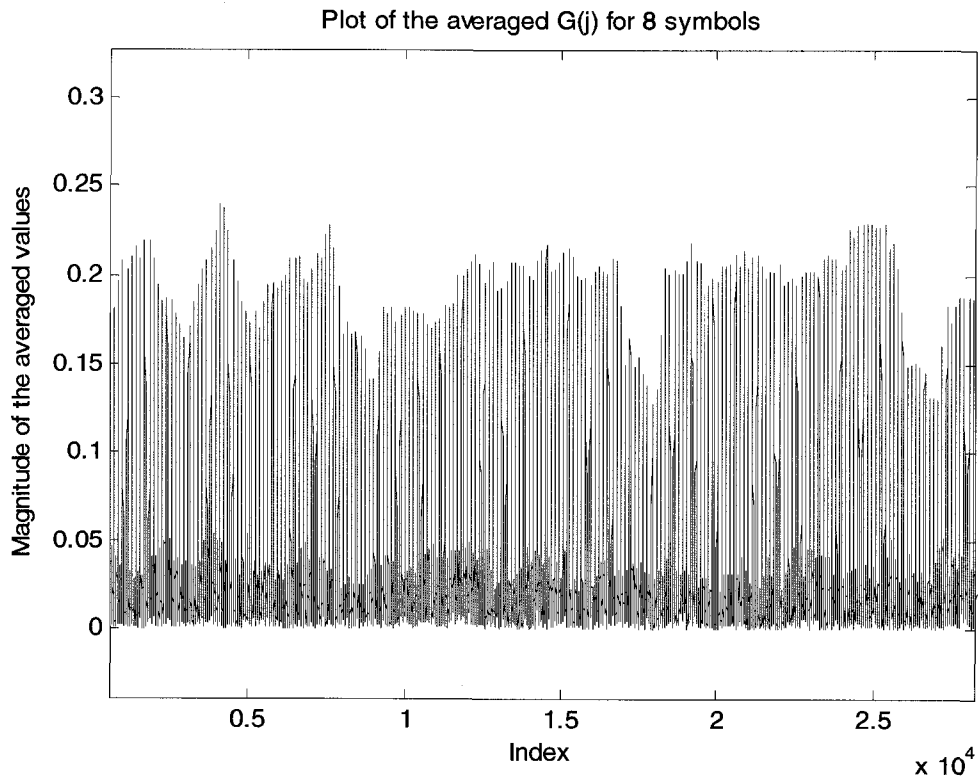


Figure 6.4 (b) Plot of windowed average for 8 symbols

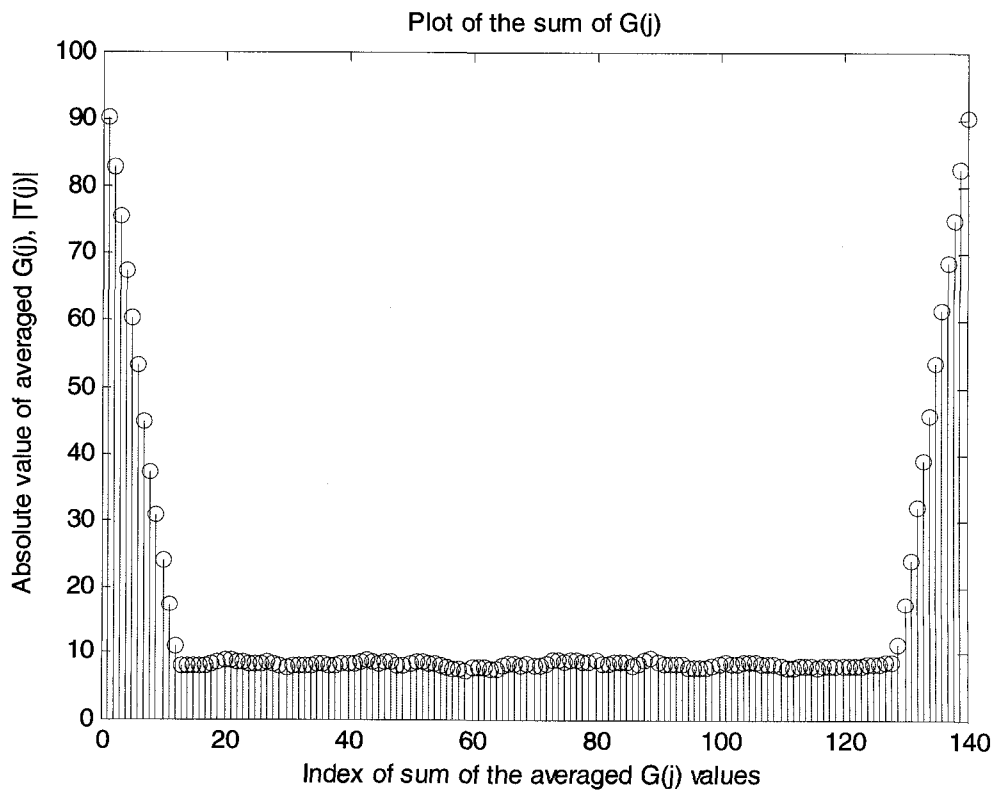


Figure 6.4 (c) Sum of windowed average with 8 symbols

6.3.4 Averaging with a Window Size of 10

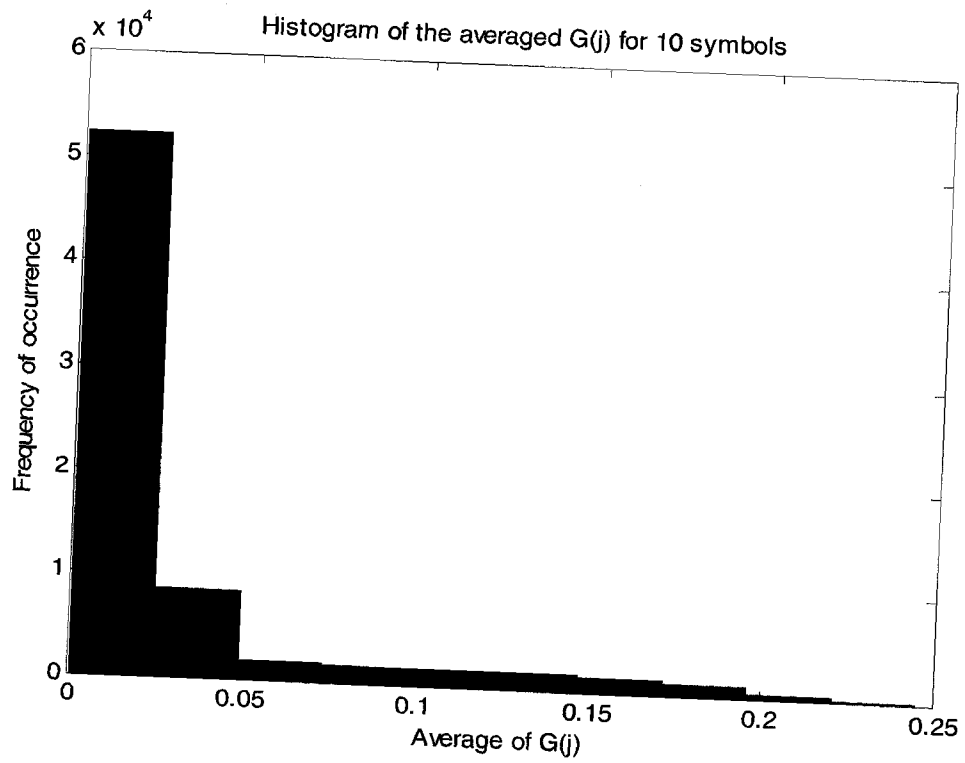


Figure 6.5 (a) Histogram of windowed average for 10 symbols

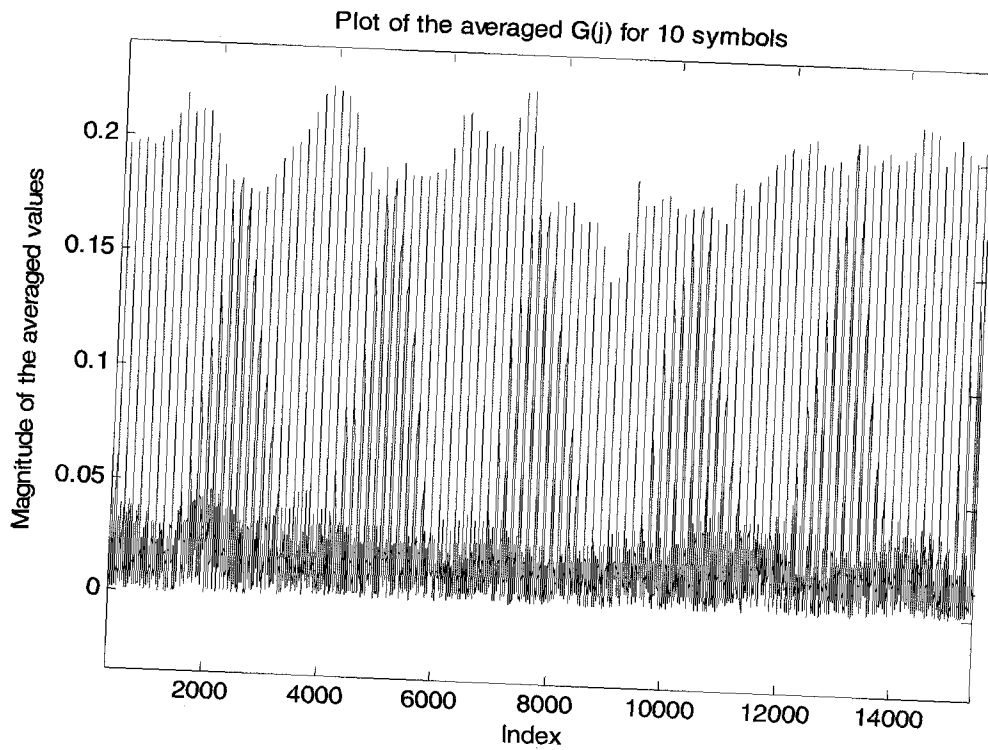


Figure 6.5 (b) Plot of windowed average for 10 symbols

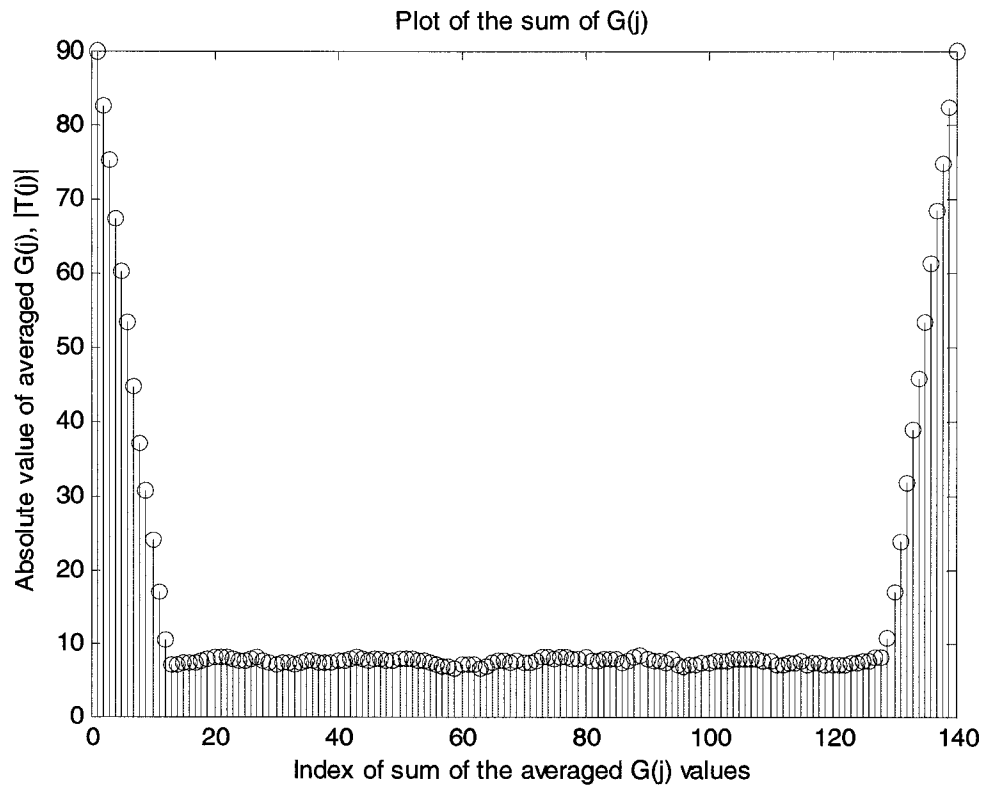


Figure 6.5 (c) Sum of windowed average with 10 symbols

6.3.5 Comparison of Results from 6.3.1 – 6.3.4

From 6.2a, 6.3a, 6.4a and 6.5a, it is seen that the number of peaks corresponding to an averaged magnitude between 0-1 is the maximum but the peaks slowly start appearing for other magnitudes as well as the window size is increased. Figure 6.2 (a) more or less does not have any peaks corresponding to magnitudes above 2.

Figures 6.2 (b), 6.3 (b), 6.4 (b) and 6.5 (b) all show the plot of the windowed average. It is clear from those figures that as the window size is increased, the plot smooths, thereby reducing the spurious bumps that occur at non-symbol boundaries.

The sum of windowed averages for various window sizes is shown in figures 6.2 (c), 6.3 (c), 6.4 (c) and 6.5 (c). These figures show that the process of summing the windowed average not only adds to the smoothing effect, but also reduces the floor of the average considerably with increase in window size. Unfortunately, the side lobe structure near the symbol boundary still looks very high even after using a number as big as 10 for window size. This still poses problems during the decision making process where one has to decide the frequency offset based on a decision made as a result of a peak-picking mechanism.

CHAPTER 7

CHARACTERIZATION OF CORRELATOR PERFORMANCE IN

FADING CHANNELS

7.1 Introduction

Chapters 5 and 6 discussed the results of various experiments carried out with the temporal correlator outputs without considering the effect of fading. In this chapter, we discuss the results of performing the same tests from Chapter 5 and 6 on the correlator outputs using four different fading channel models for the indoor wireless LAN environments [Chapter 3]. We will analyze each channel model in the subsequent sections of this chapter.

7.2 Keller Channel Model [1]

Figure 3.2 (a) shows the impulse response for this channel model with three taps. The figures below are indicative of the response of this channel to the synchronization algorithms.

7.2.1 Raw Correlator Outputs for Keller Channel Model [1]

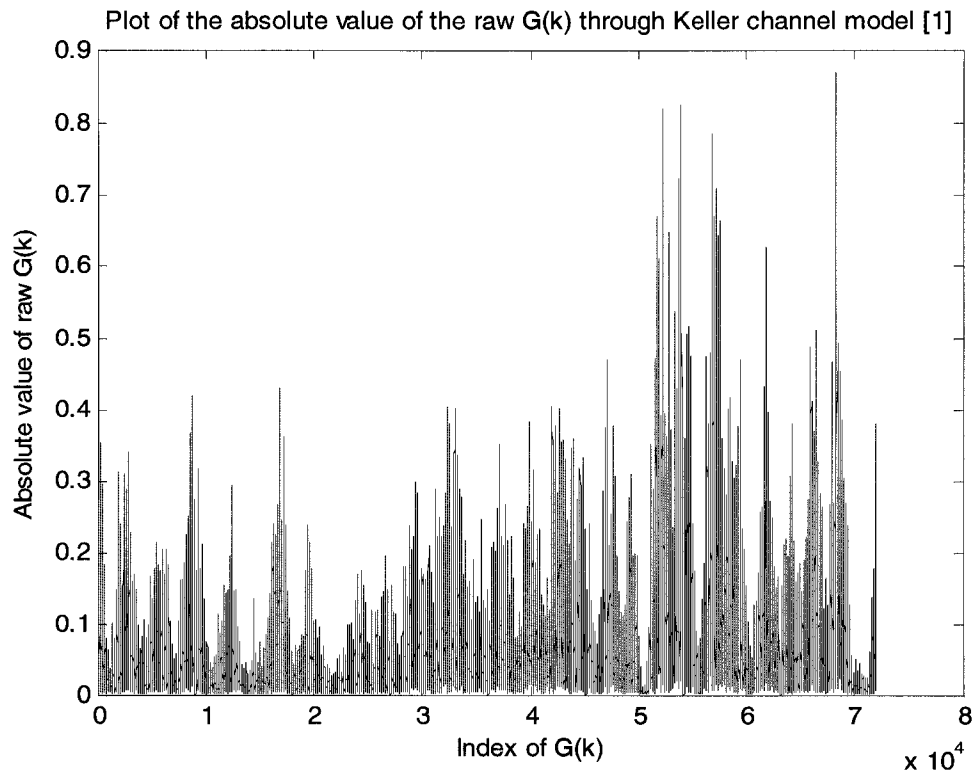


Figure 7.1 (a) Keller channel model [1] - $G(j)$ correlator output

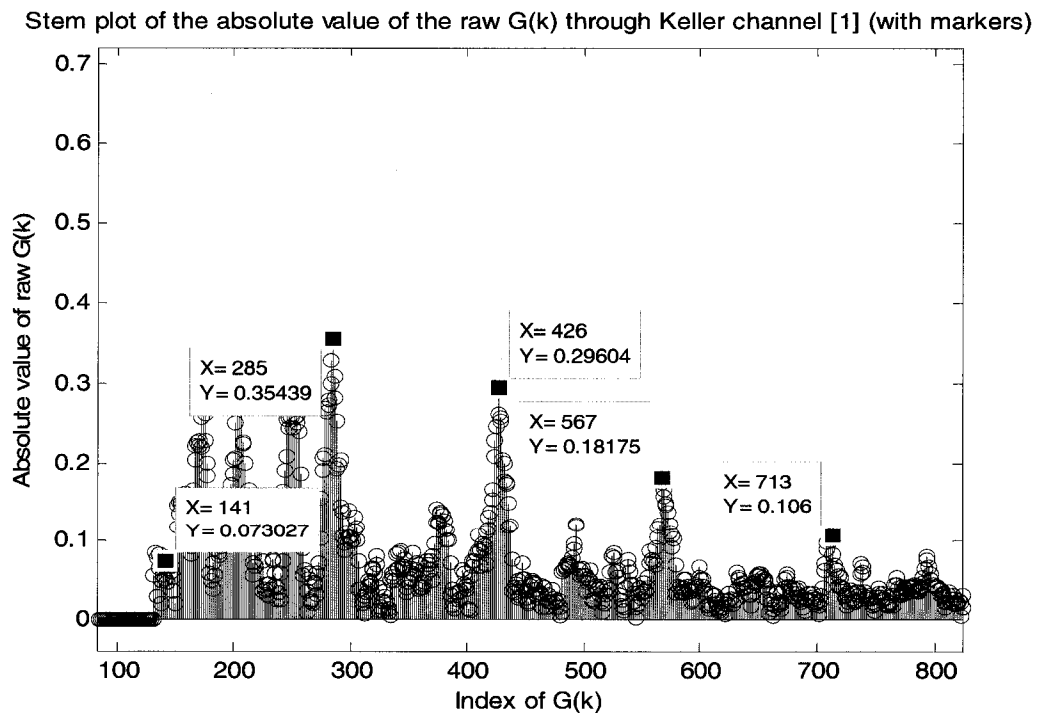


Figure 7.1 (b) Keller channel model [1] - $G(j)$ output with timing offsets

Figure 7.1 shows the raw $G(j)$ correlator output for the Keller channel model [1]. It is clear that the correlator peak heights are not the same as they were when fading was not taken into account. More importantly, the symbol boundaries as shown by the markers in figure 7.1(b) do not correspond to the actual symbol boundaries but instead are non-uniformly shifted as a result of time-dispersive fading.

Stem plot of the absolute value of the raw $R(m)$ through Keller channel [1] with markers at the frame boundaries

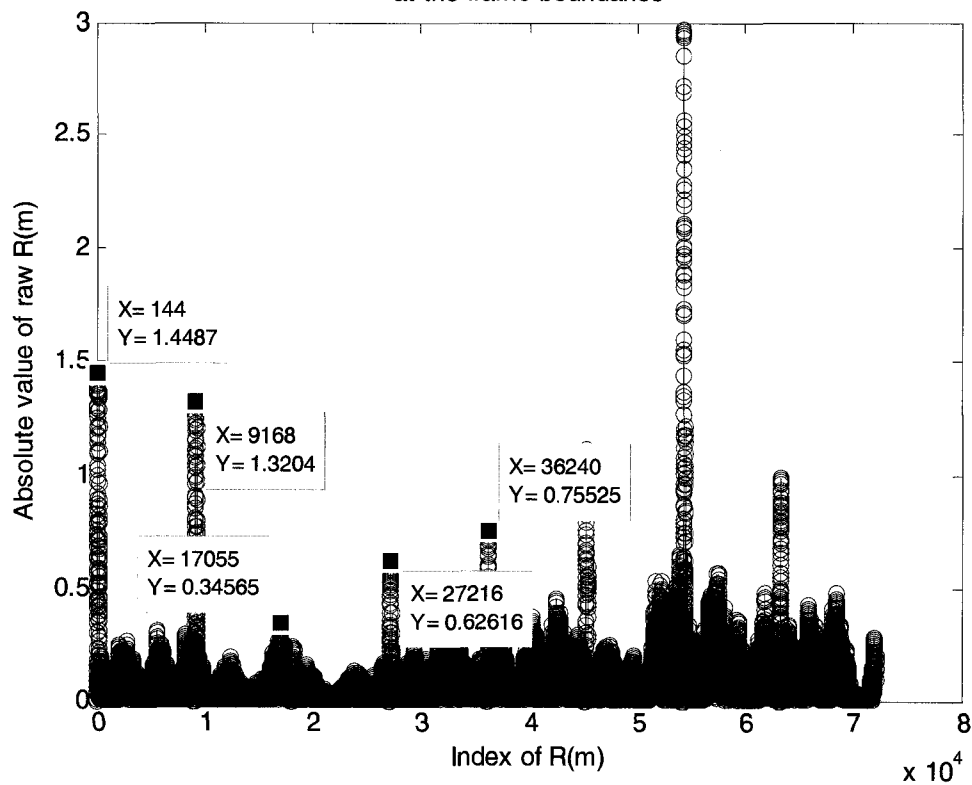


Figure 7.1 (c) Keller channel model [1] - $R(j)$ output with markers

7.2.2 Interpeak Intervals for Keller Channel Model [1]

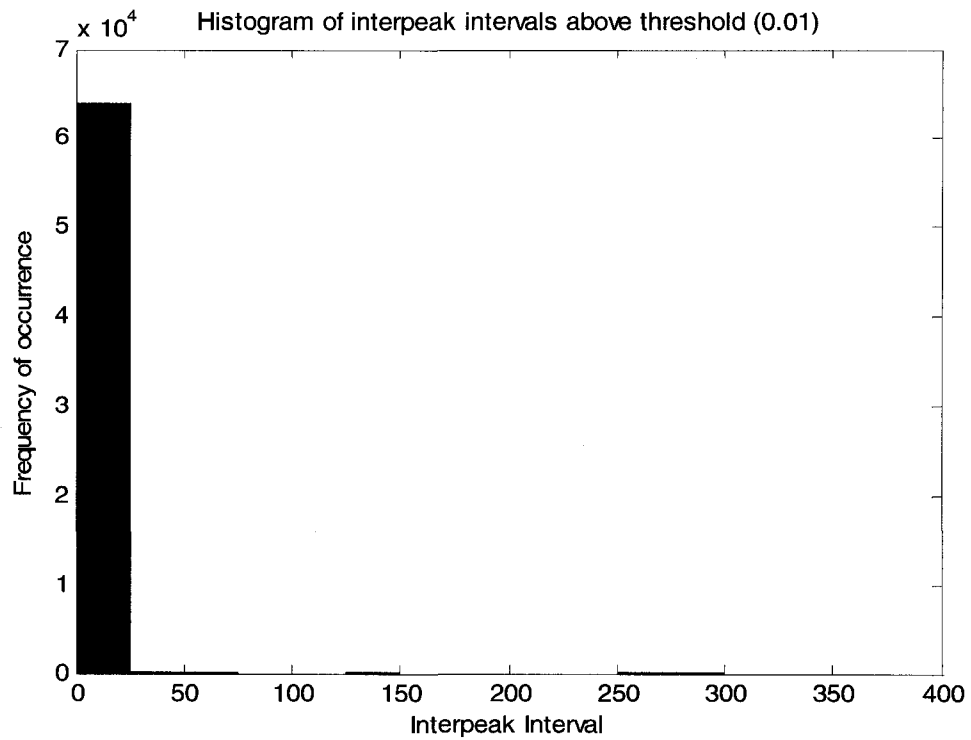


Figure 7.2 (a) Keller channel model [1] - Interpeak intervals above 0.01

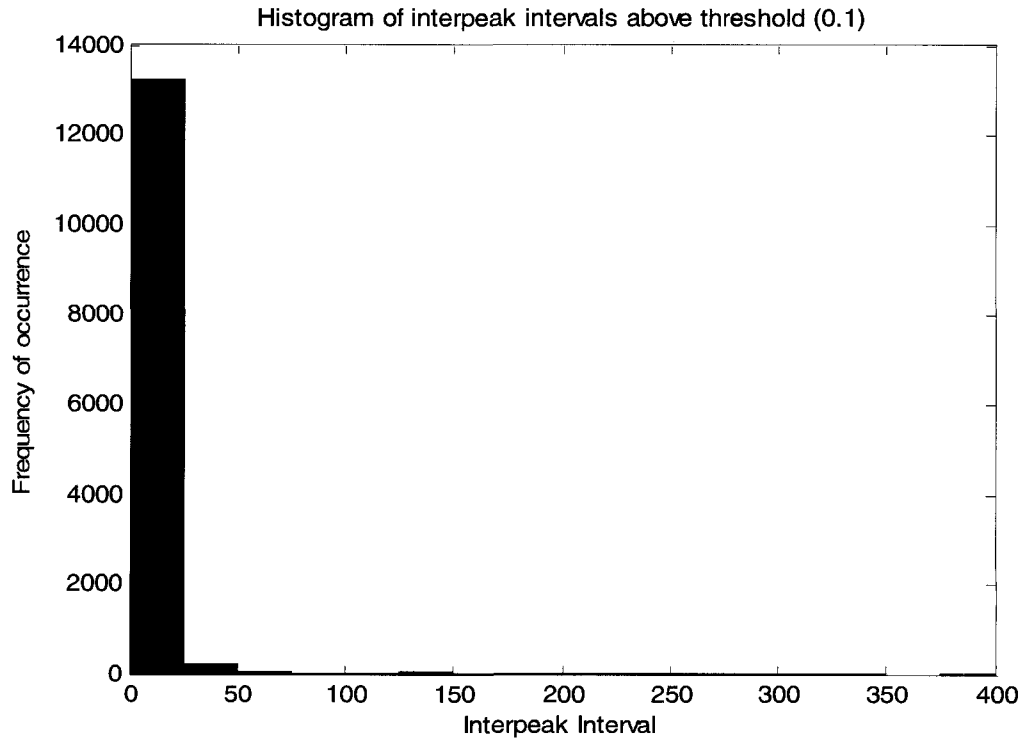


Figure 7.2 (b) Keller channel model [1] - Interpeak intervals above 0.1

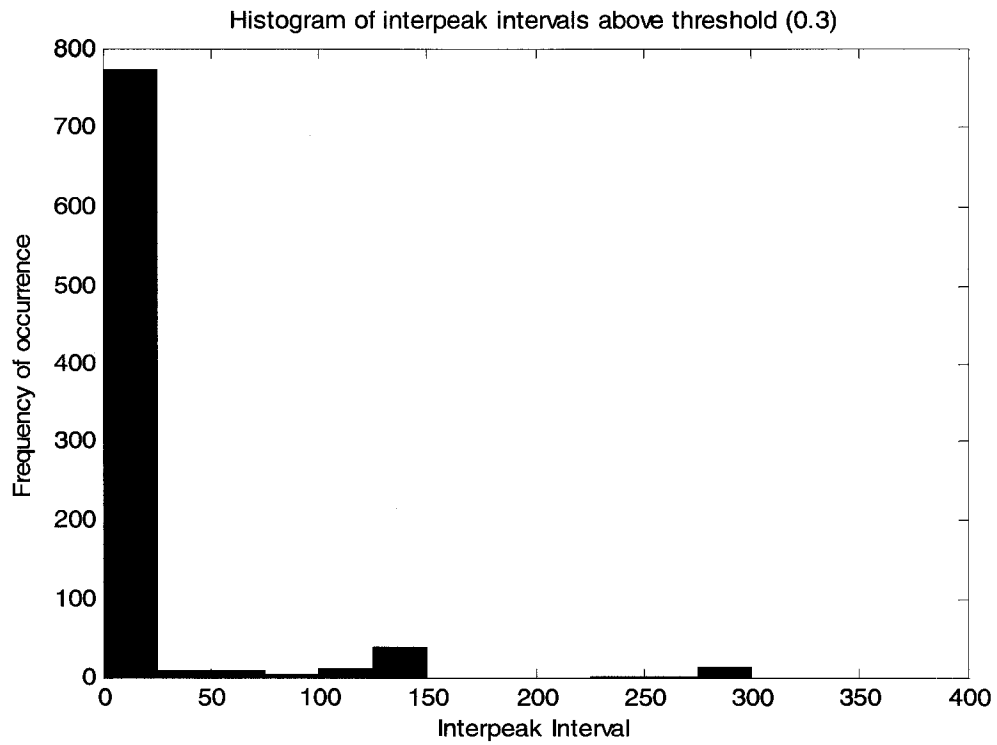


Figure 7.2 (c) Keller channel model [1] – Interpeak intervals above 0.3

A comparison of the histograms from 7.2.2 with the corresponding non-fading cases shows a general *increase* in the number of peaks above threshold at the same interpeak interval values. The reason for this is not fully clear, but may relate to the channel gain structure.

7.2.3 Averaging with Variable Window Sizes for Keller Channel Model [1]

7.2.3.1 Averaging with a Window Size of 4

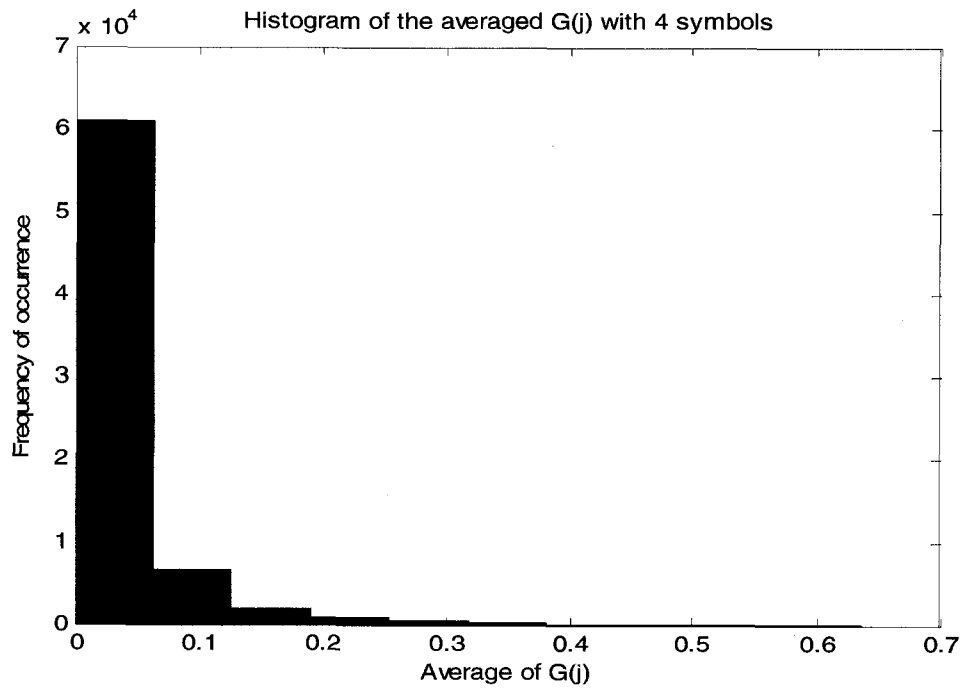


Figure 7.3 (a) Keller channel model [1] – Windowed average with 4 symbols

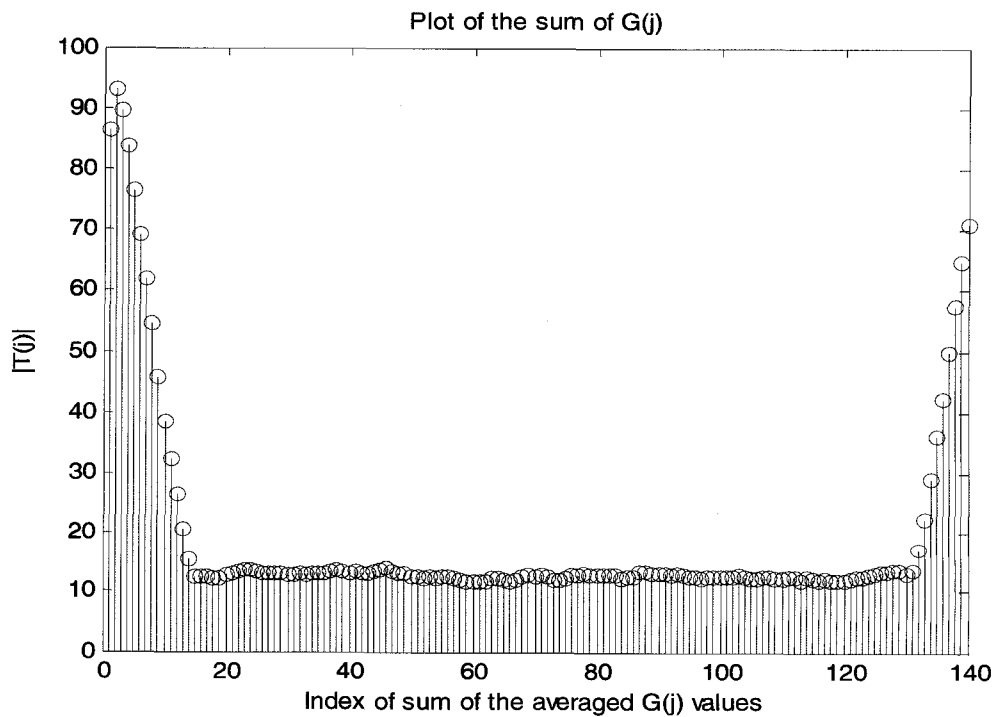


Figure 7.3 (b) Keller channel model [1] – Sum of windowed average for 4 symbols

7.2.3.2 Averaging with a Window Size of 6

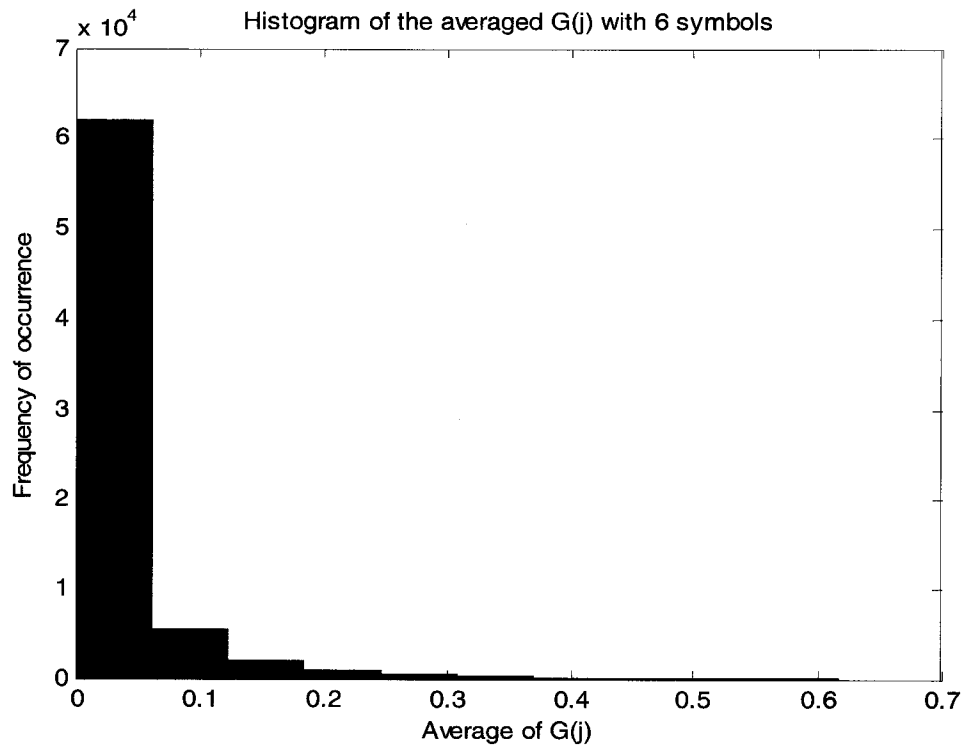


Figure 7.4 (a) Keller channel model [1] – Windowed average with 6 symbols

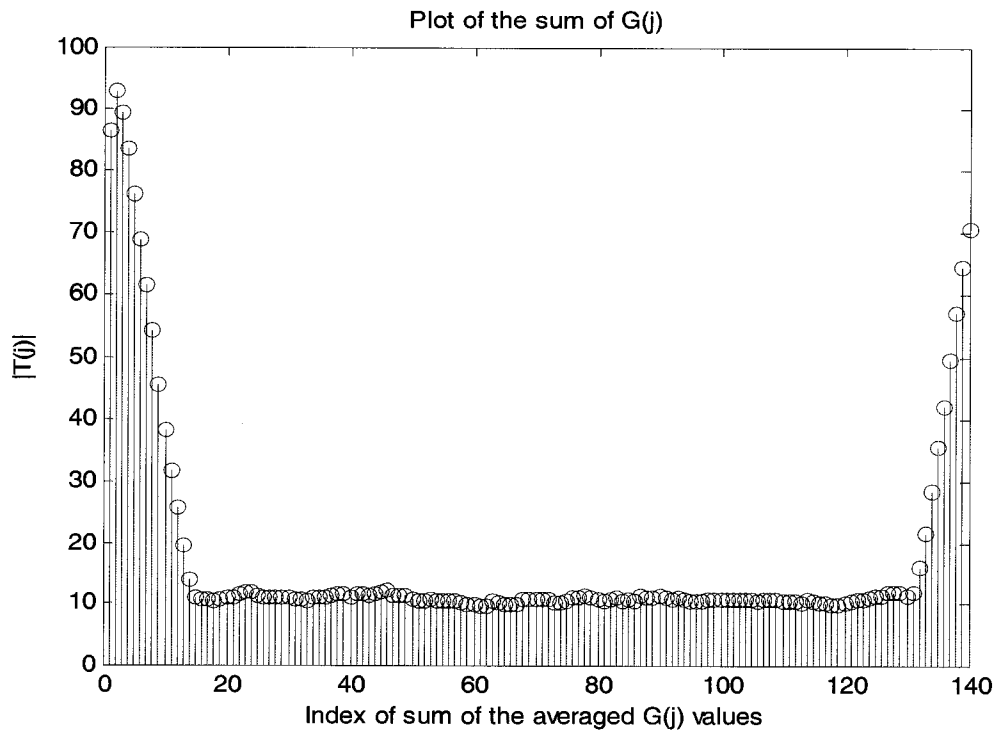


Figure 7.4 (b) Keller channel model [1] – Sum of windowed average for 6 symbols

7.2.3.3 Averaging with a Window Size of 8

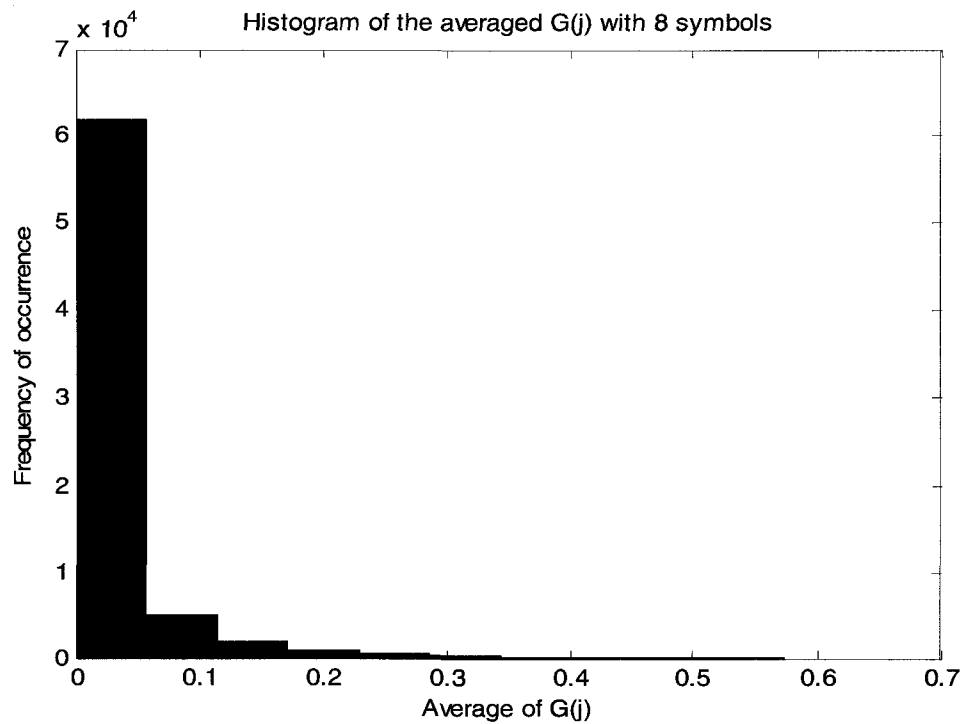


Figure 7.5 (a) Keller channel model [1] – Windowed average with 8 symbols

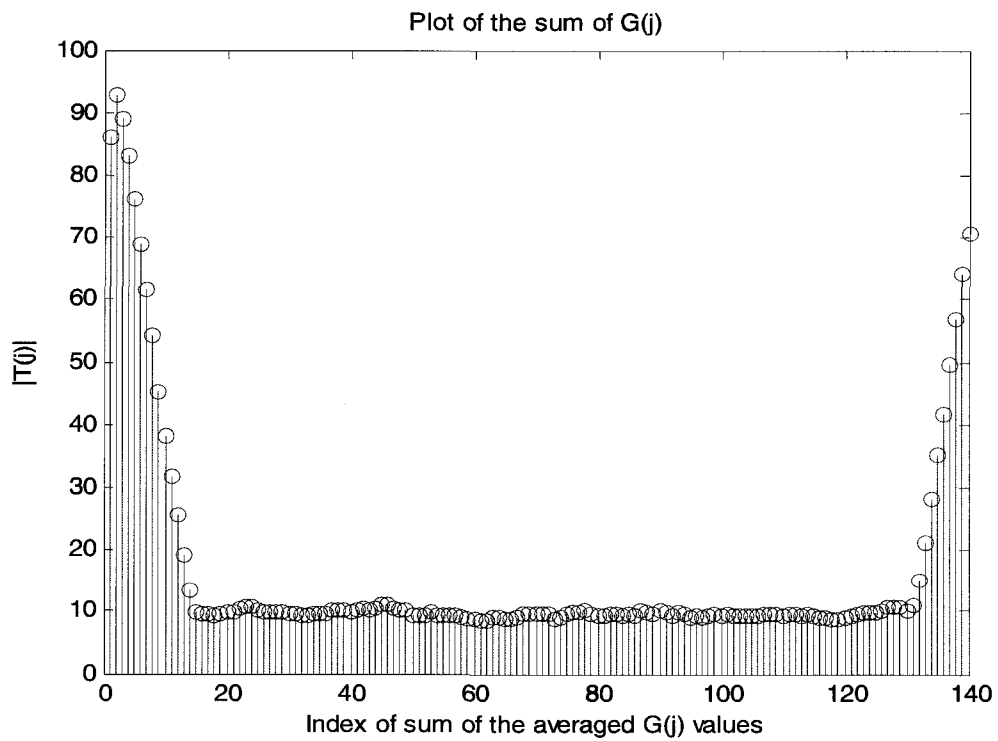


Figure 7.5 (b) Keller channel model [1] – Sum of windowed average for 8 symbols

7.2.3.4 Averaging with a Window Size of 10

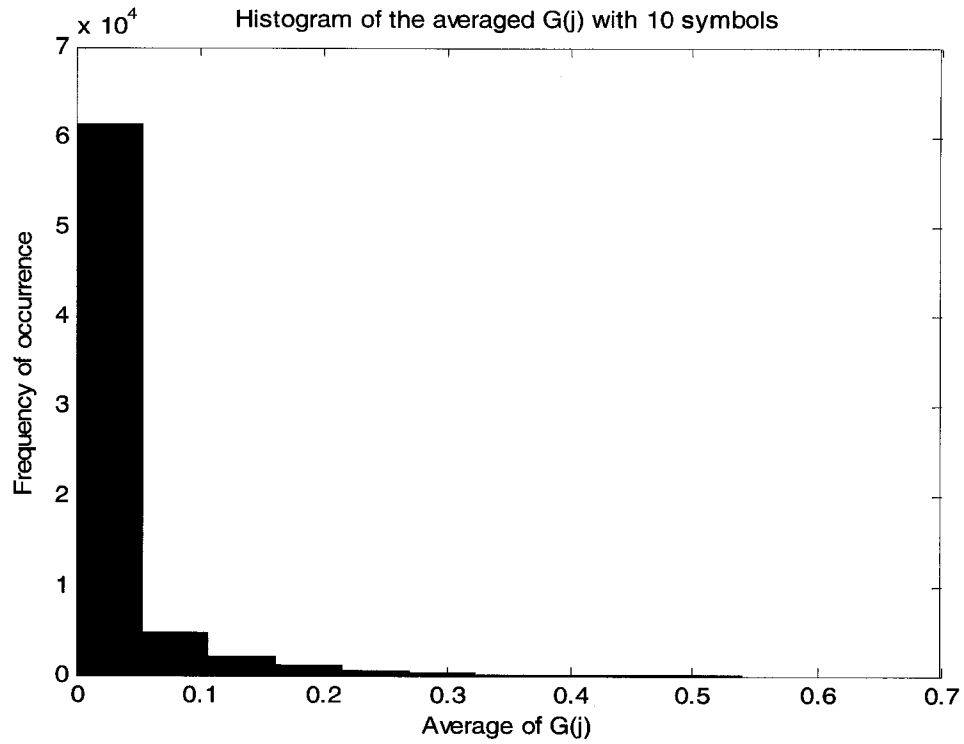


Figure 7.6 (a) Keller channel model [1] – Windowed average with 10 symbols

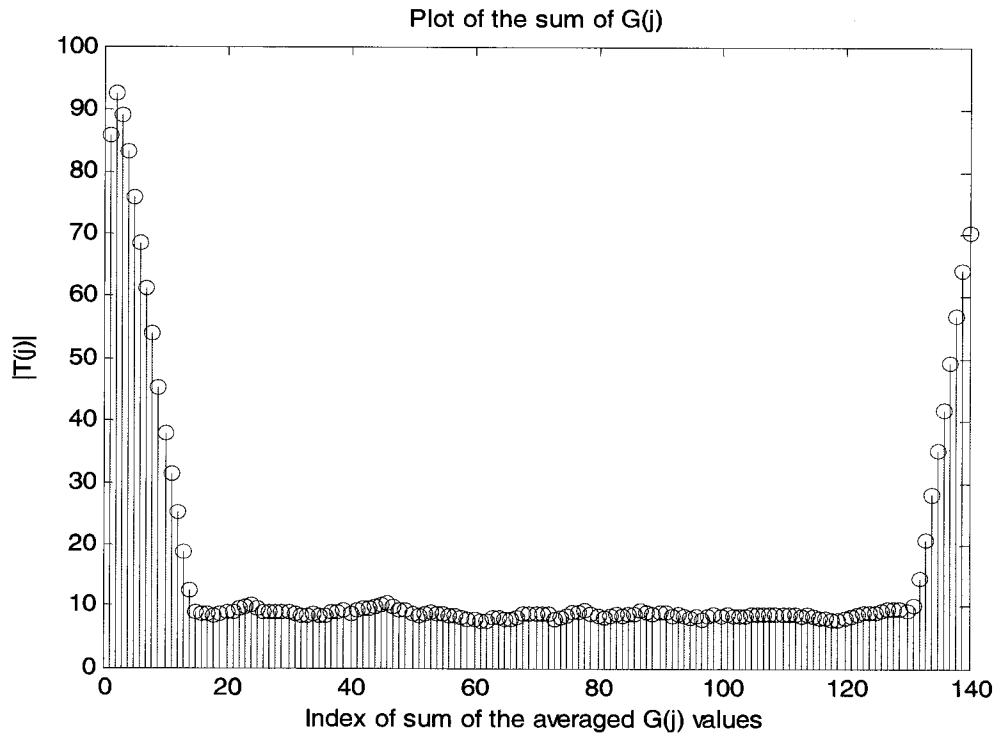


Figure 7.6 (b) Keller channel model [1] – Sum of windowed average for 10 symbols

After careful examination of the results of the experiments conducted on Keller channel model [1], it is clear that there is a slight shift in the symbol and frame boundaries and the peak amplitudes due to the effect of fading, though the windowed average with variable window lengths does not seem to show significant improvement in sharpening the peaks at the boundaries. Hence, the effect of correlator output averaging is *not* as apparent as it was for the non-fading channel case.

7.3 Model B [2]

Figure 3.3 shows the impulse response of this channel model with 5 taps. The following figures are indicative of the response of this channel to the correlation algorithms $G(j)$ and $R(j)$.

7.3.1 Raw Correlator Outputs for Model B [2]

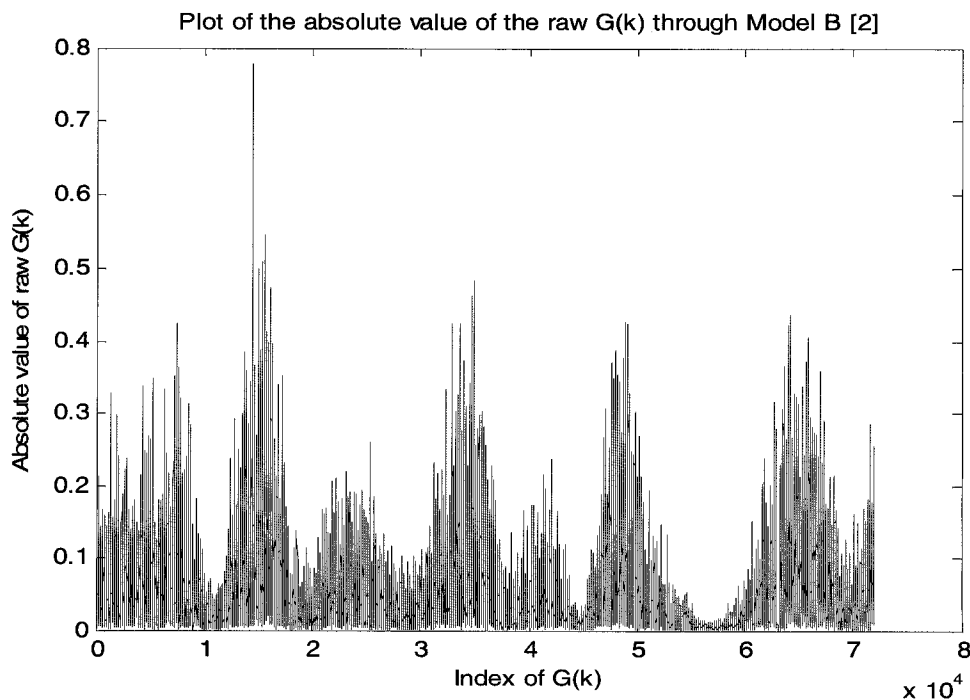


Figure 7.7 (a) Model B [2] - $G(j)$ correlator output

Stem plot of the absolute value of the raw $G(k)$ through Model B [2] (with markers)

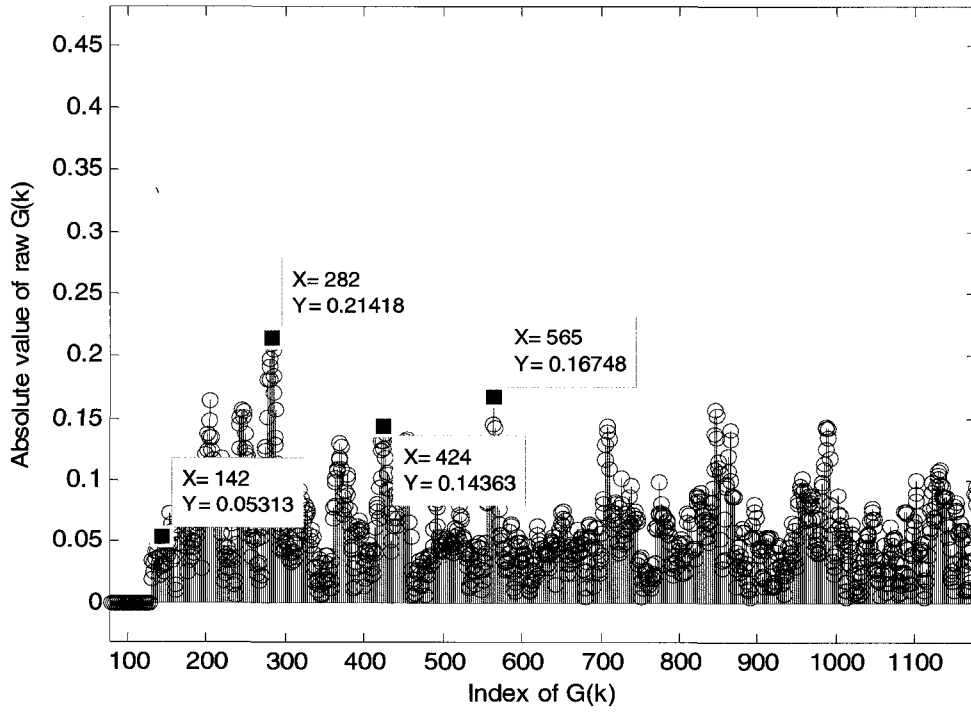


Figure 7.7 (b) Model B [2] - $G(j)$ correlator output with timing offsets

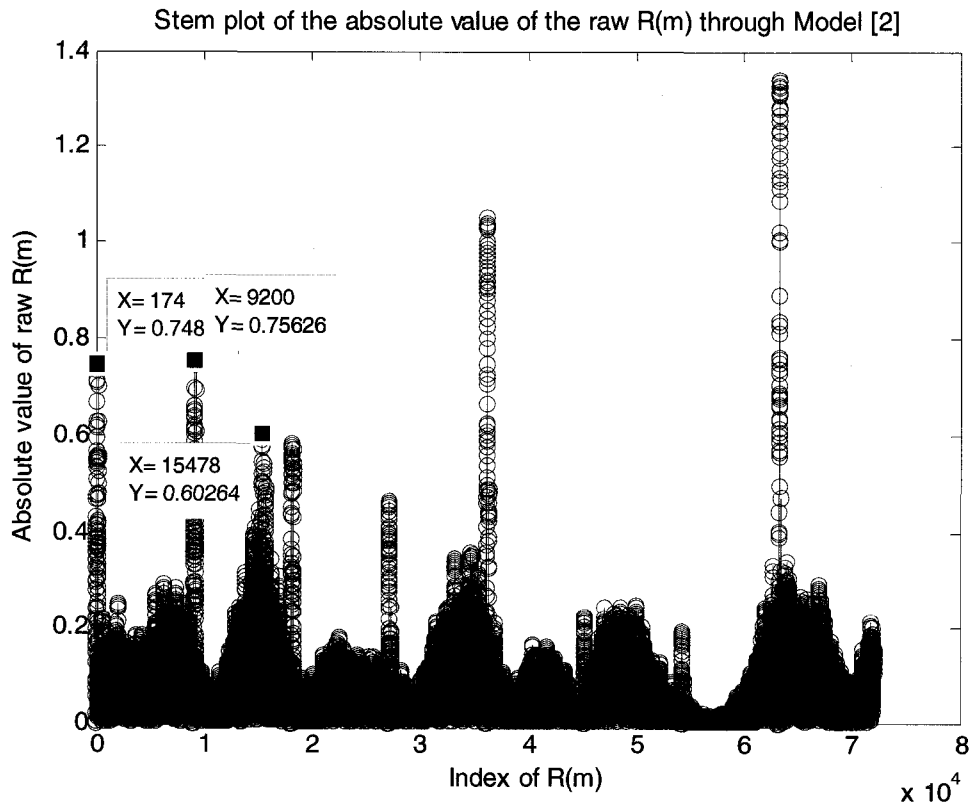


Figure 7.7 (c) Model B [2] - $R(j)$ output with markers

7.3.2 Interpeak Intervals for Model B [2]

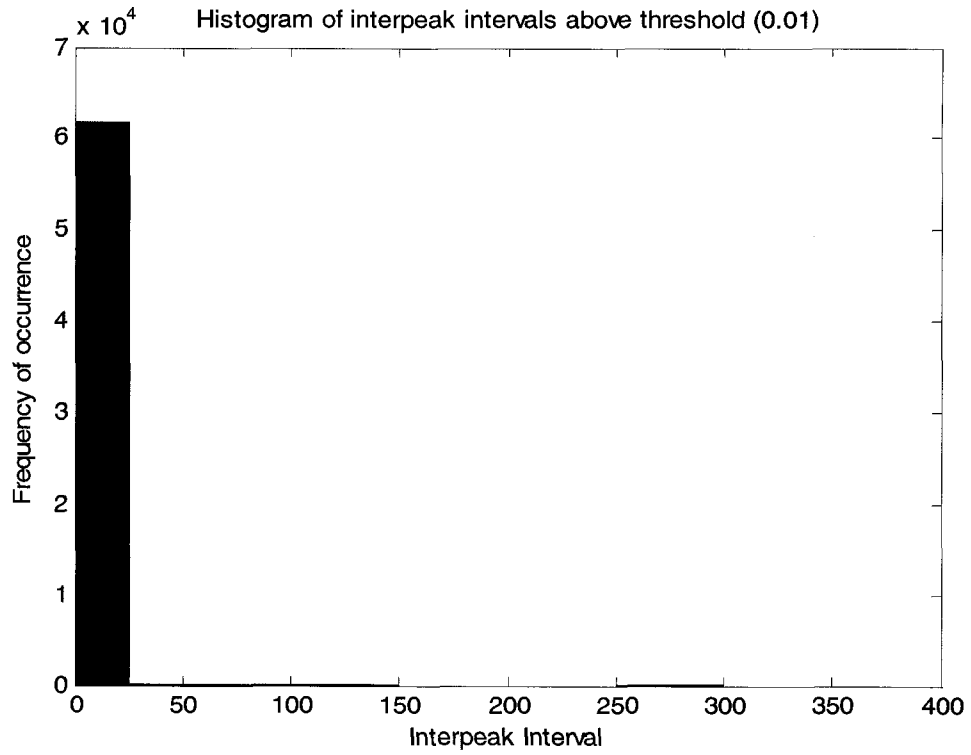


Figure 7.8 (a) Model B [2] – Interpeak intervals above 0.01

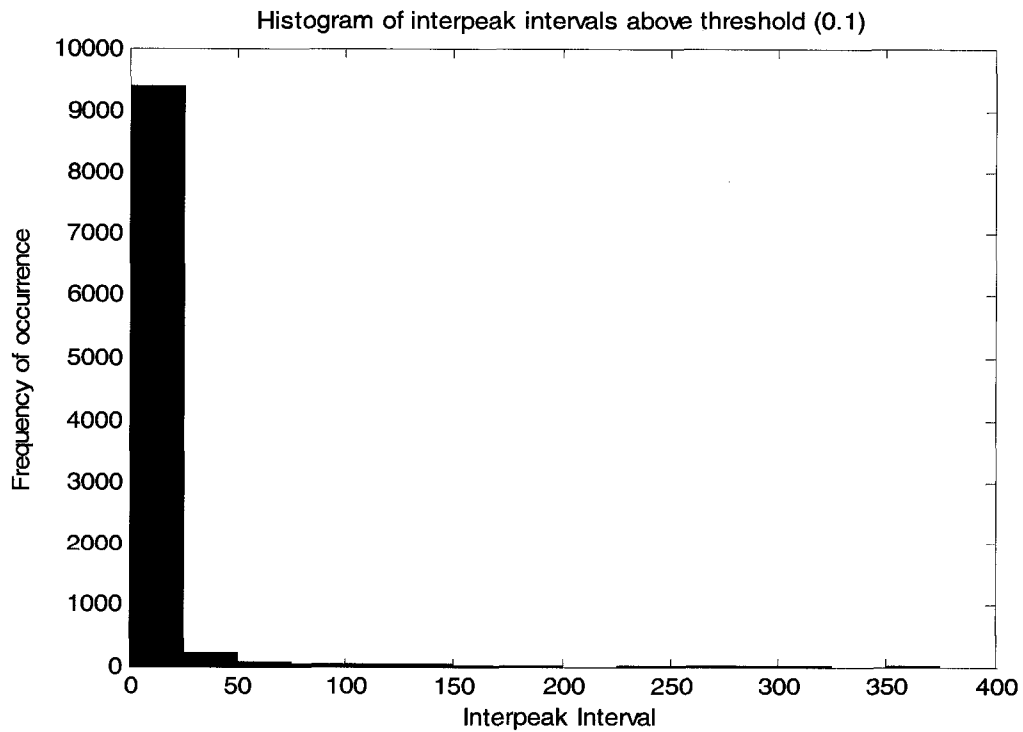


Figure 7.8 (b) Model B [2] – Interpeak intervals above 0.1

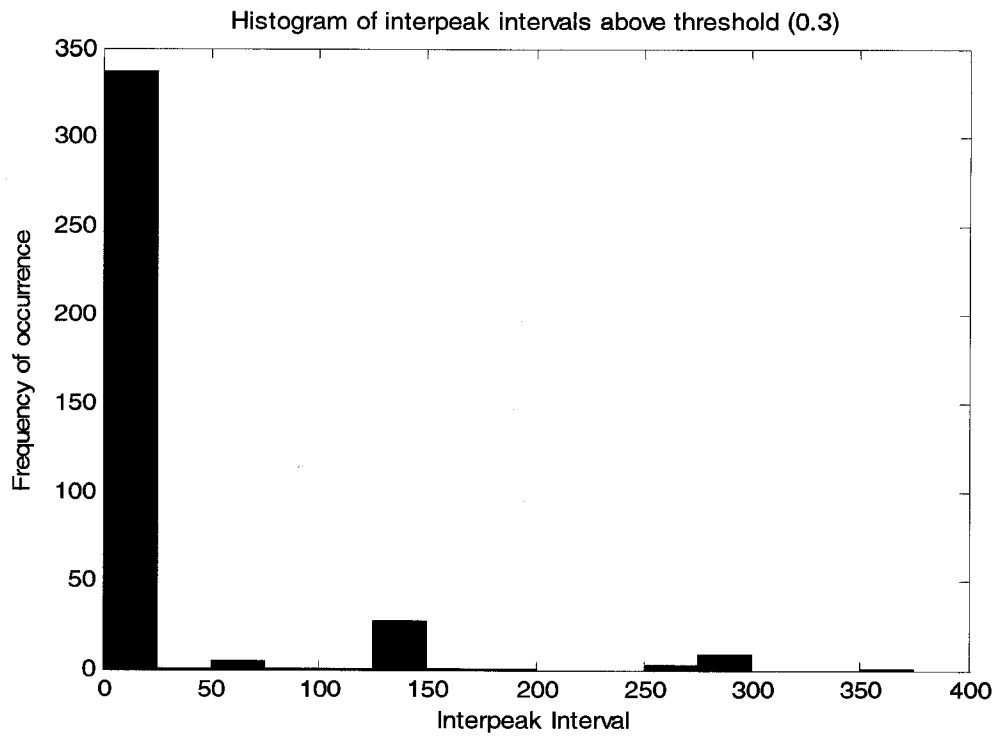


Figure 7.8 (c) Model B [2] – Interpeak intervals above 0.3

7.3.3 Averaging with Variable Window Sizes for Model B [2]

7.3.3.1 Averaging with a Window Size of 4

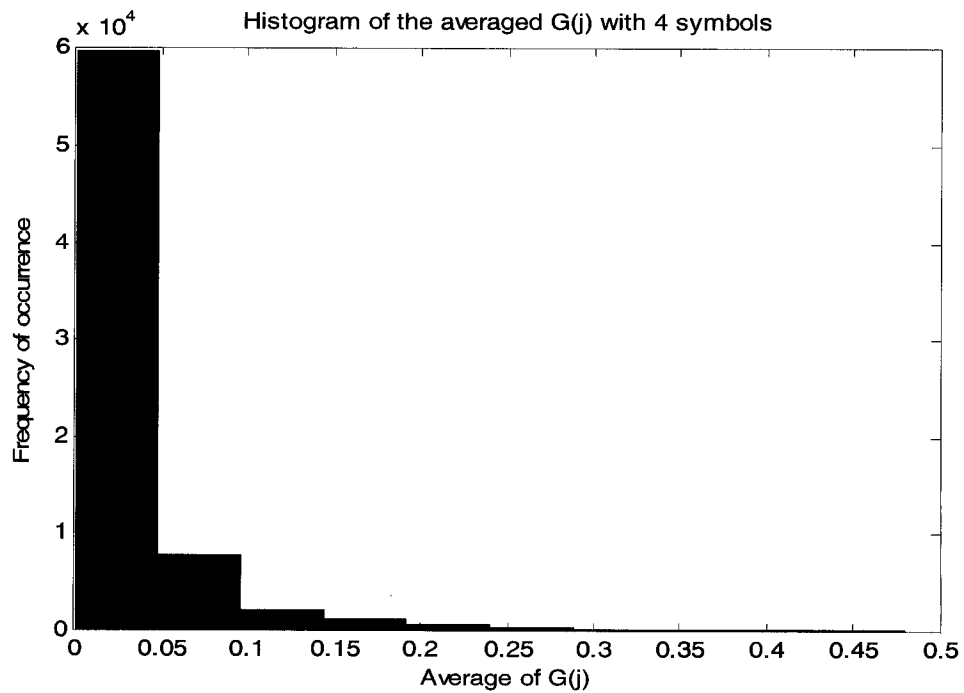


Figure 7.9 (a) Model B [2] – Windowed average with 4 symbols

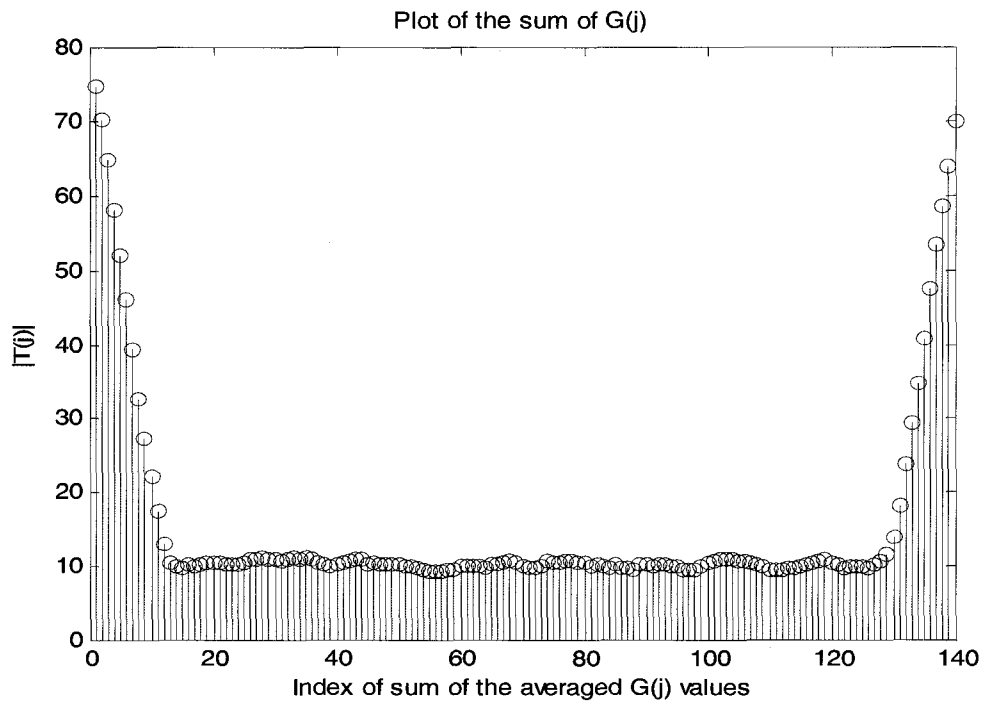


Figure 7.9 (b) Model B [2] – Sum of windowed average for 4 symbols

7.3.3.2 Averaging with a Window Size of 6

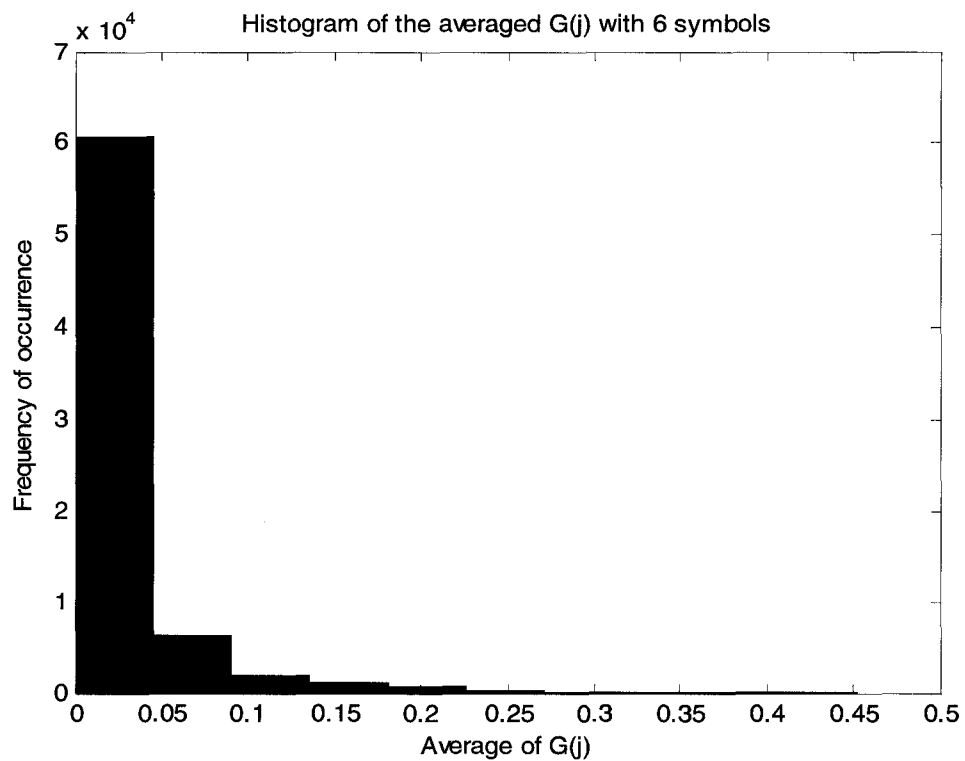


Figure 7.10 (a) Model B [2] – Windowed average with 6 symbols

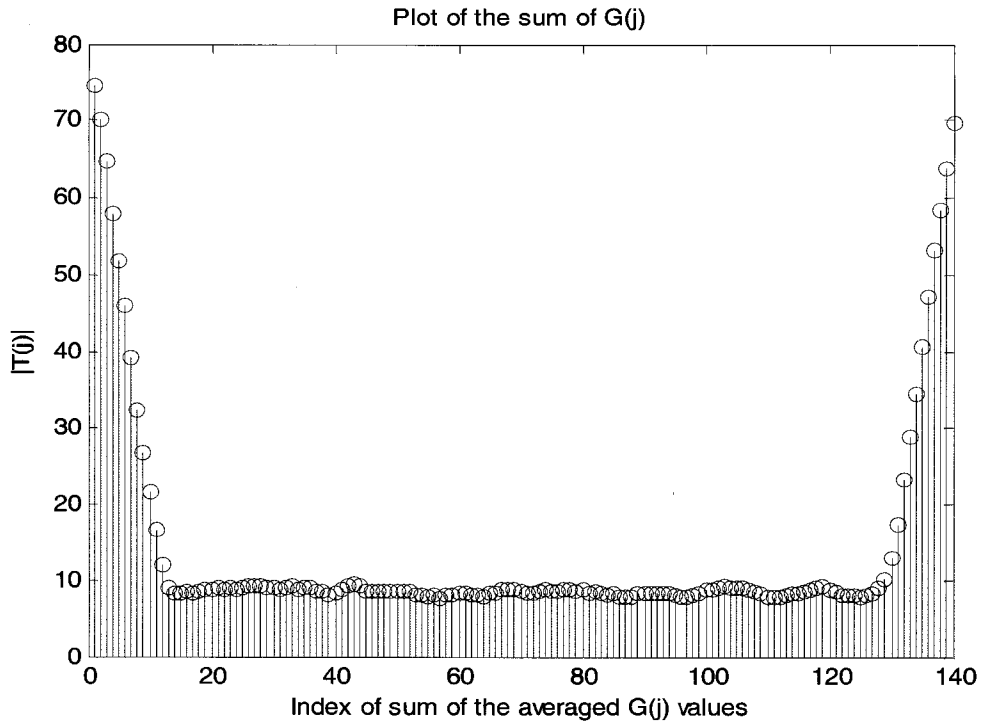


Figure 7.10 (b) Model B [2] – Sum of windowed average for 6 symbols

7.3.3.3 Averaging with a Window Size of 8

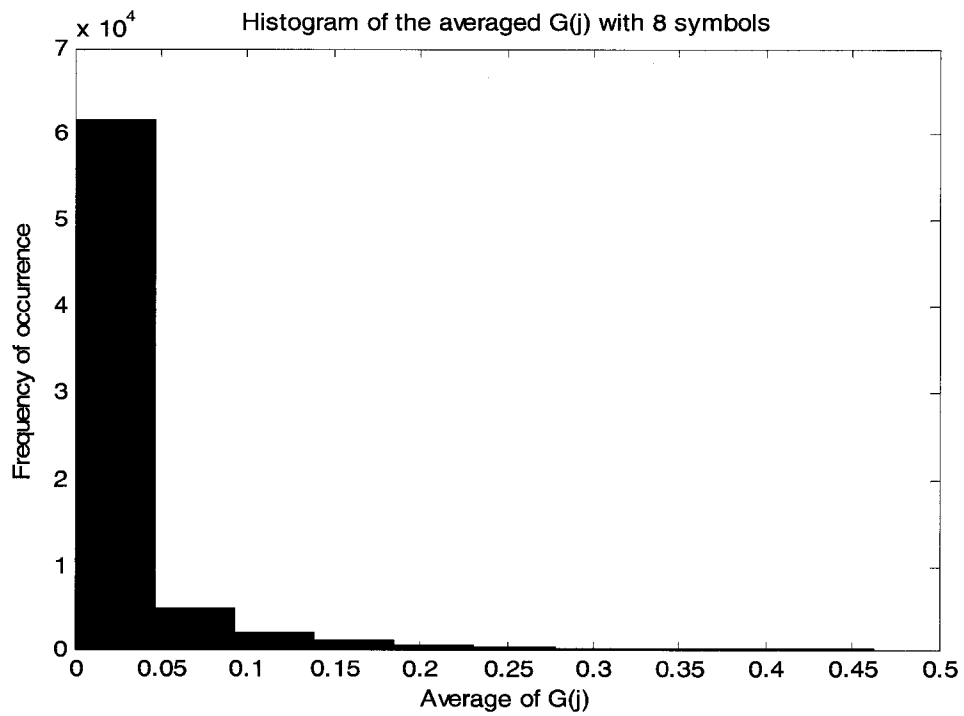


Figure 7.11 (a) Model B [2] – Windowed average with 8 symbols

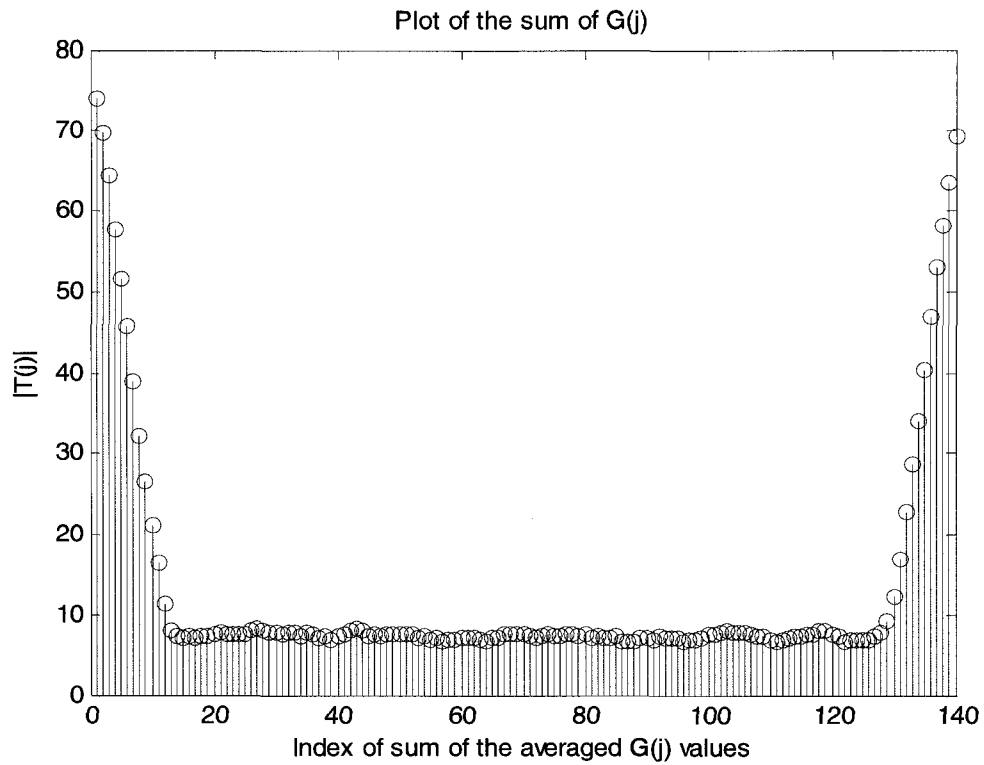


Figure 7.11 (b) Model B [2] – Sum of windowed average for 8 symbols

7.3.3.4 Averaging with a Window Size of 10

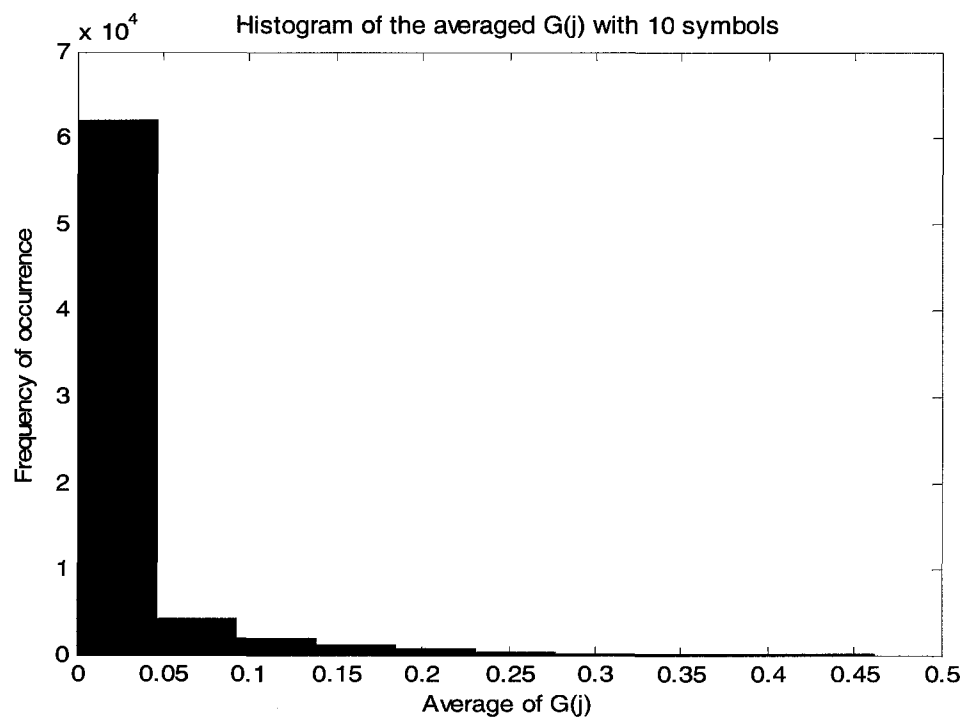


Figure 7.12 (a) Model B [2] – Windowed average with 10 symbols

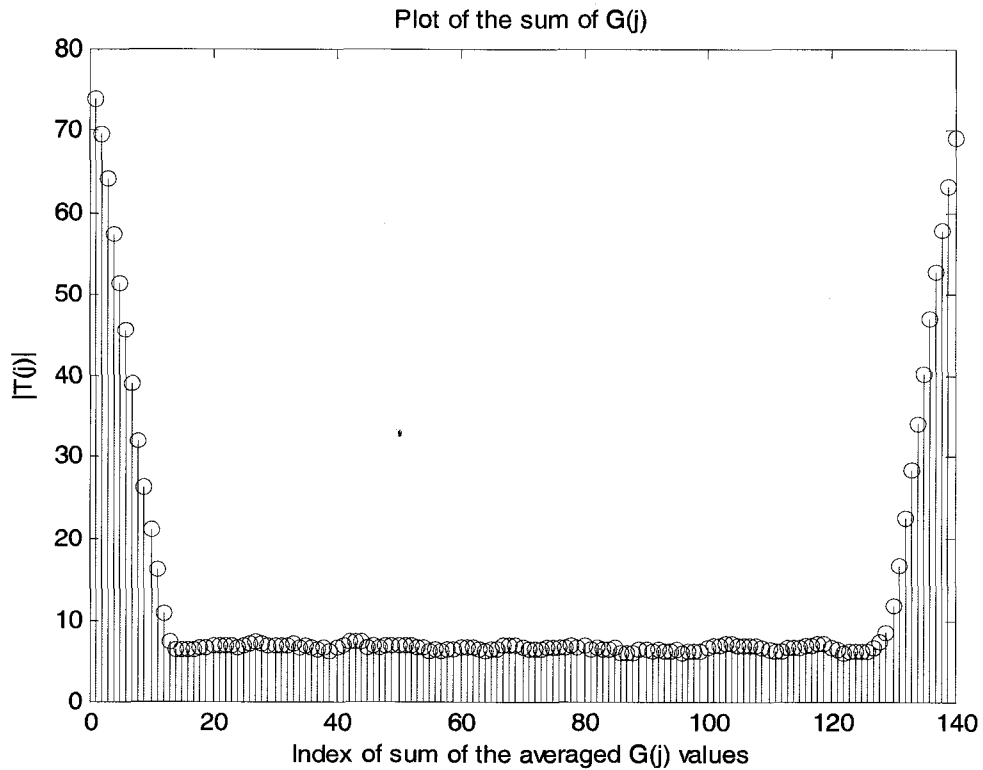


Figure 7.12 (b) Model B [2] – Sum of windowed average for 10 symbols

Once again, correlator output averaging sharpens the $|G(j)|_{avg}$ distribution by shifting more peaks into the lowest value bin.

7.4 Model C [2]

Figure 3.4 shows the impulse response of this channel model with 6 taps. The following figures are indicative of the response of this channel to the correlation algorithms $G(j)$ and $R(j)$.

7.4.1 Raw Correlator Outputs for Model C [2]

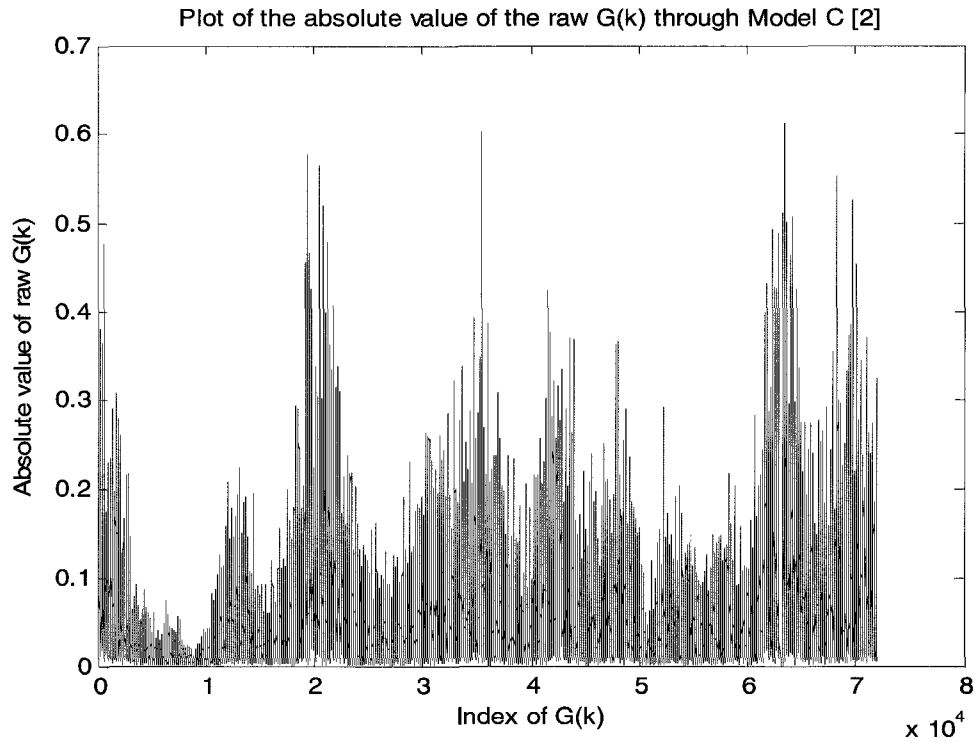


Figure 7.13 (a) Model C [2] - $G(j)$ correlator output

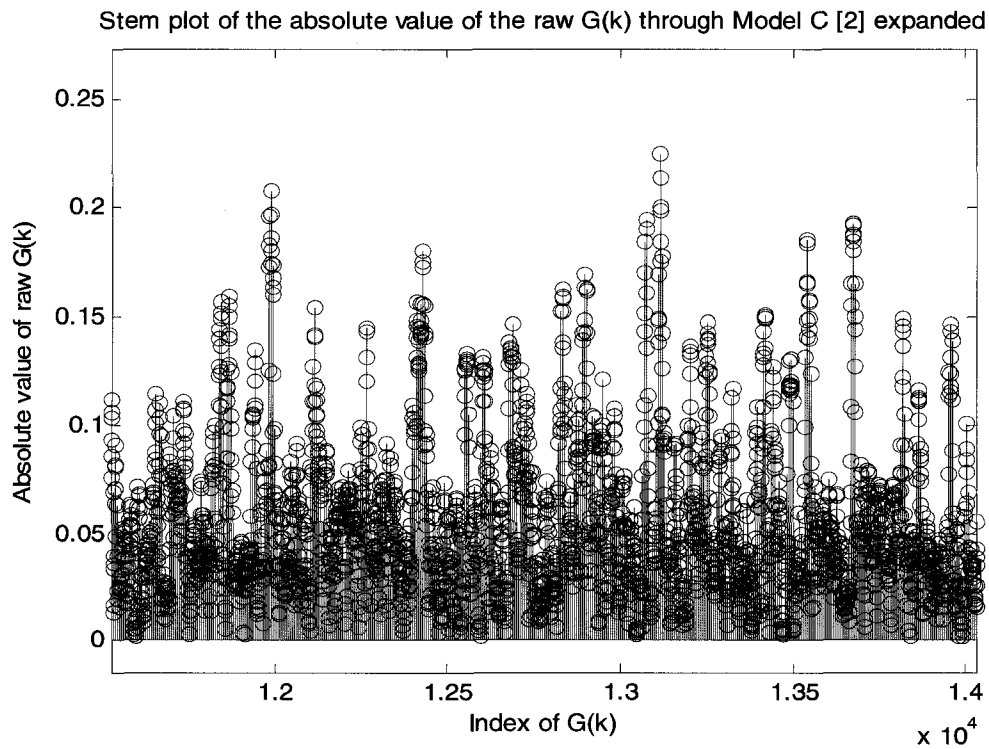
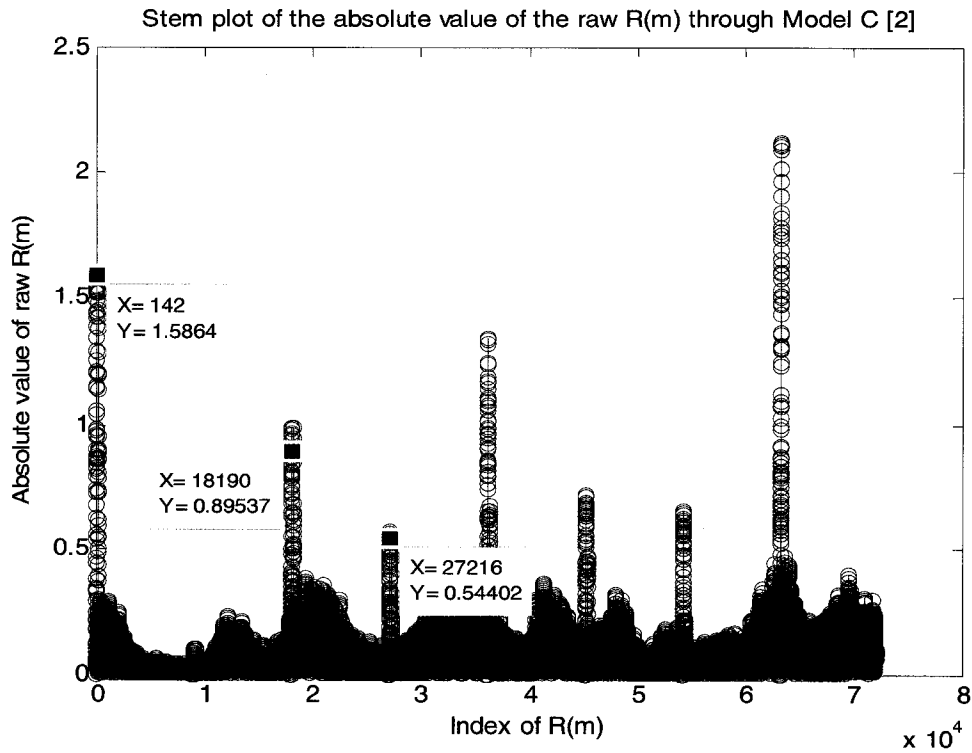
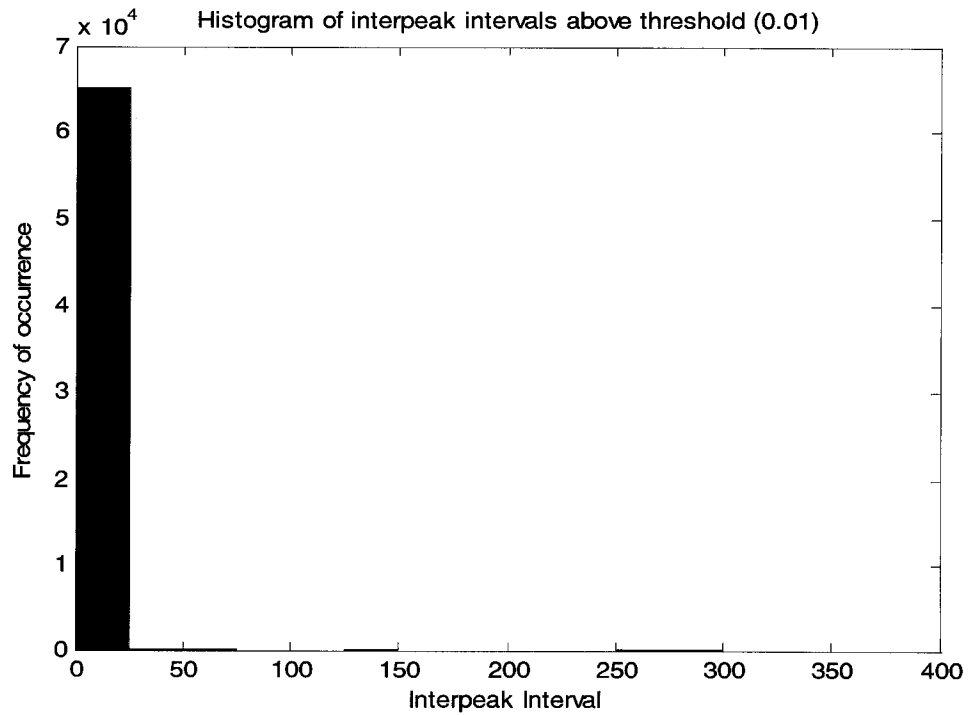


Figure 7.13 (b) Model C [2] - Expanded $G(j)$ output



7.4.2 Interpeak Intervals for Model C [2]



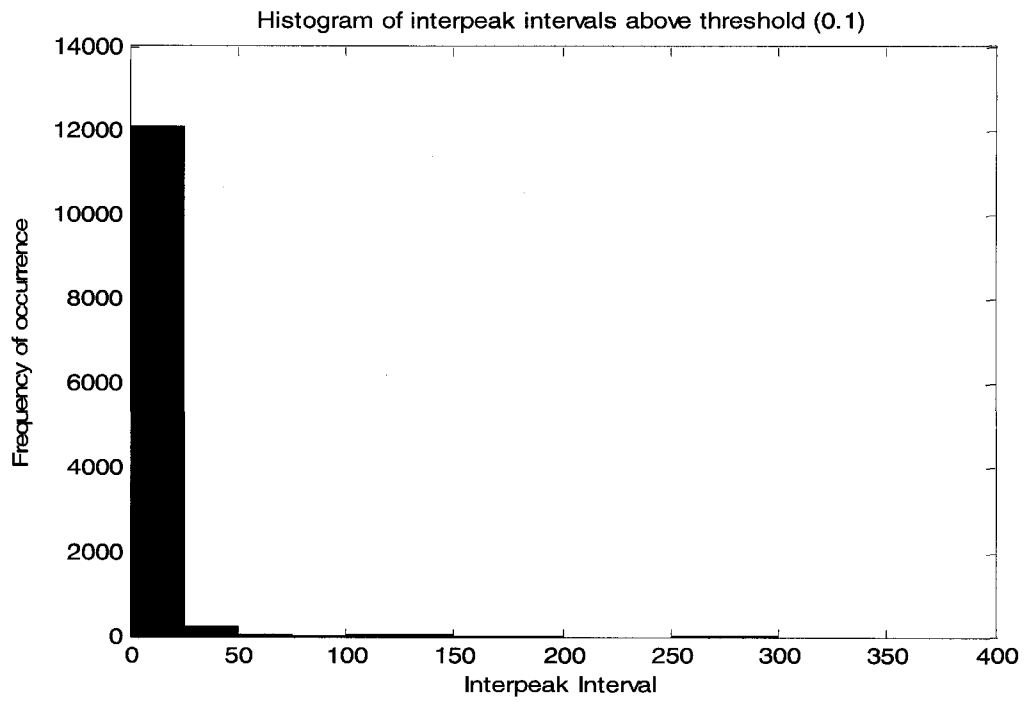


Figure 7.14 (b) Model C [2] – Interpeak intervals above 0.1

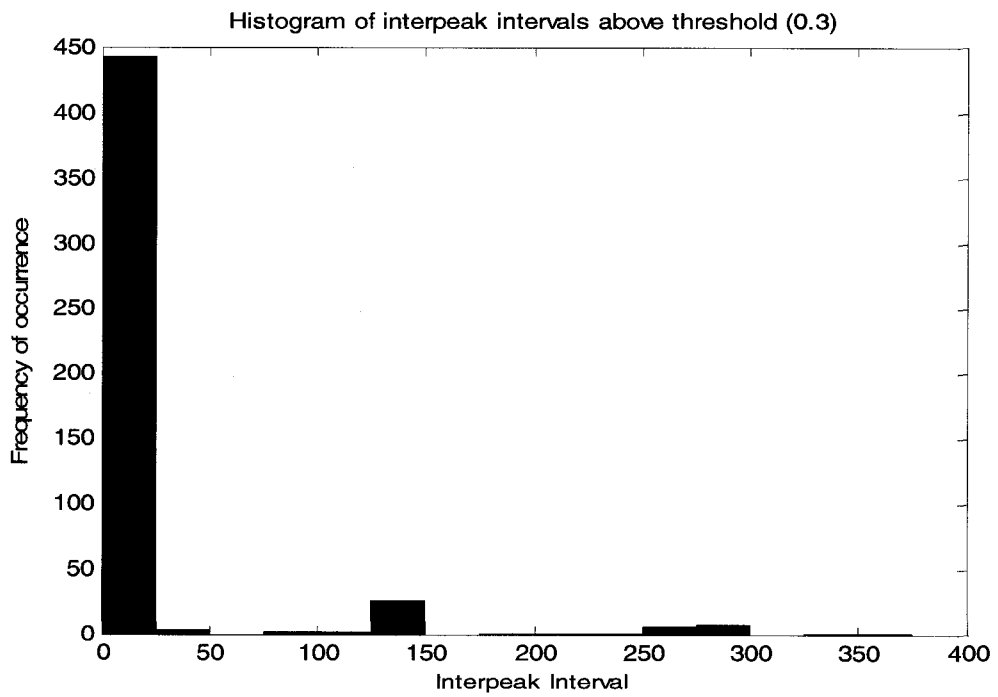


Figure 7.14 (c) Model C [2] – Interpeak intervals above 0.3

7.4.3 Averaging with Variable Window Sizes for Model C [2]

7.4.3.1 Averaging with a Window Size of 4

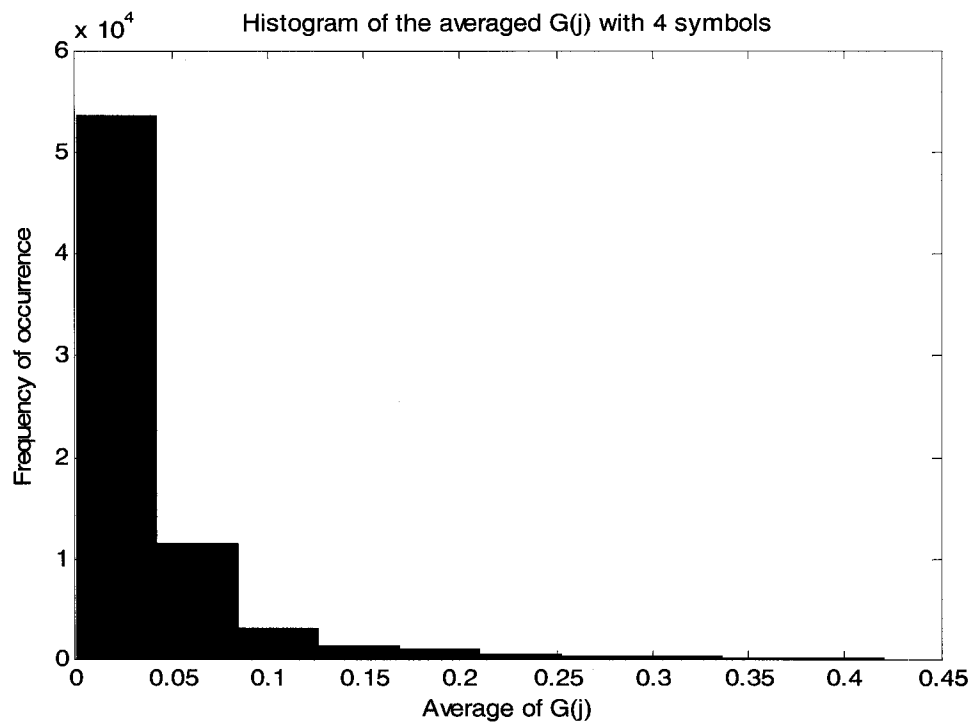


Figure 7.15 (a) Model C [2] – Windowed average with 4 symbols

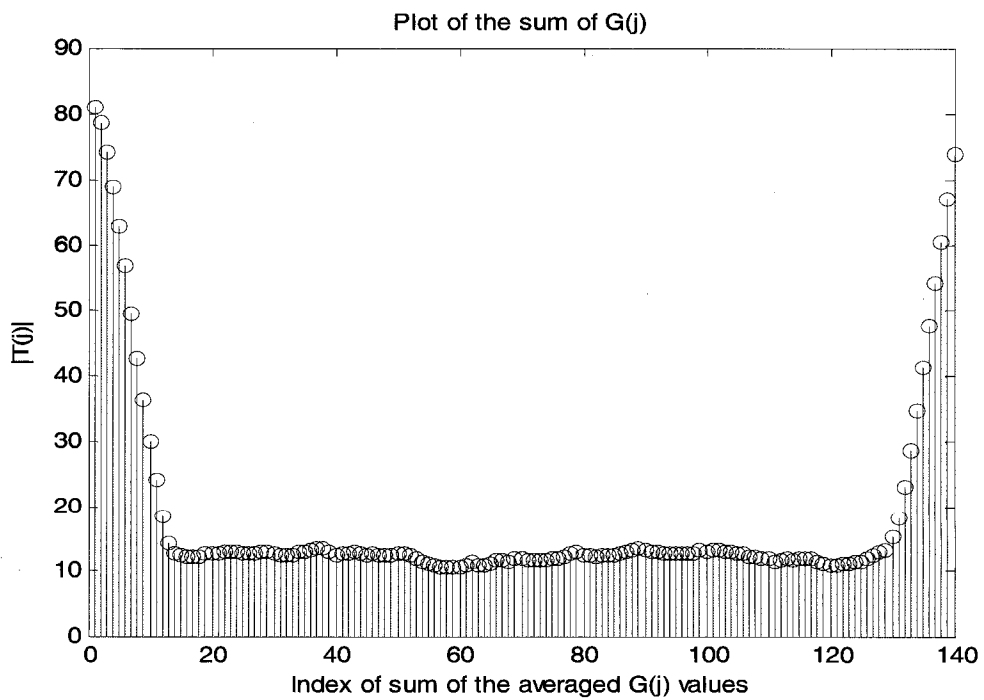


Figure 7.15 (b) Model C [2] – Sum of windowed average for 4 symbols

7.4.3.2 Averaging with a Window Size of 6

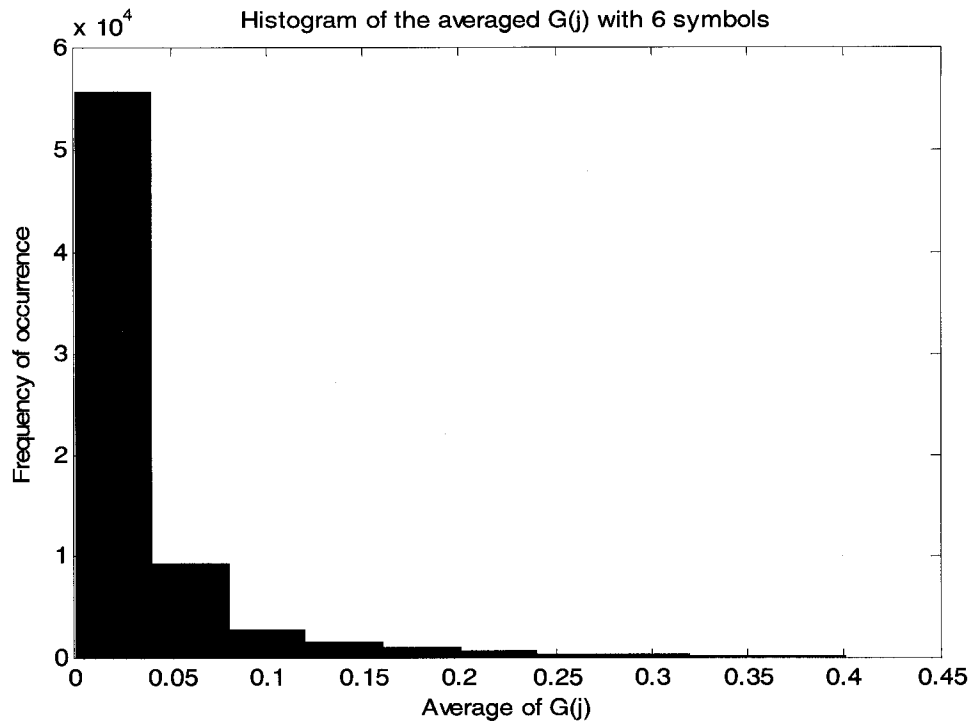


Figure 7.16 (a) Model C [2] – Windowed average with 6 symbols

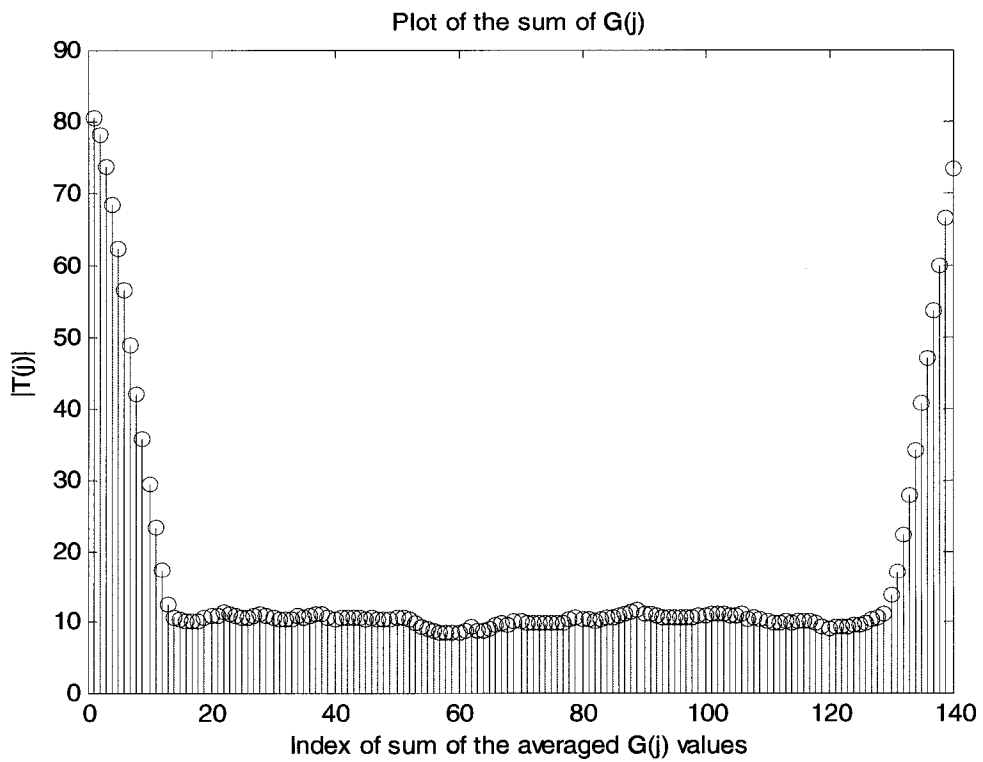


Figure 7.16 (b) Model C [2] – Sum of windowed average for 6 symbols

7.4.3.3 Averaging with a Window Size of 8

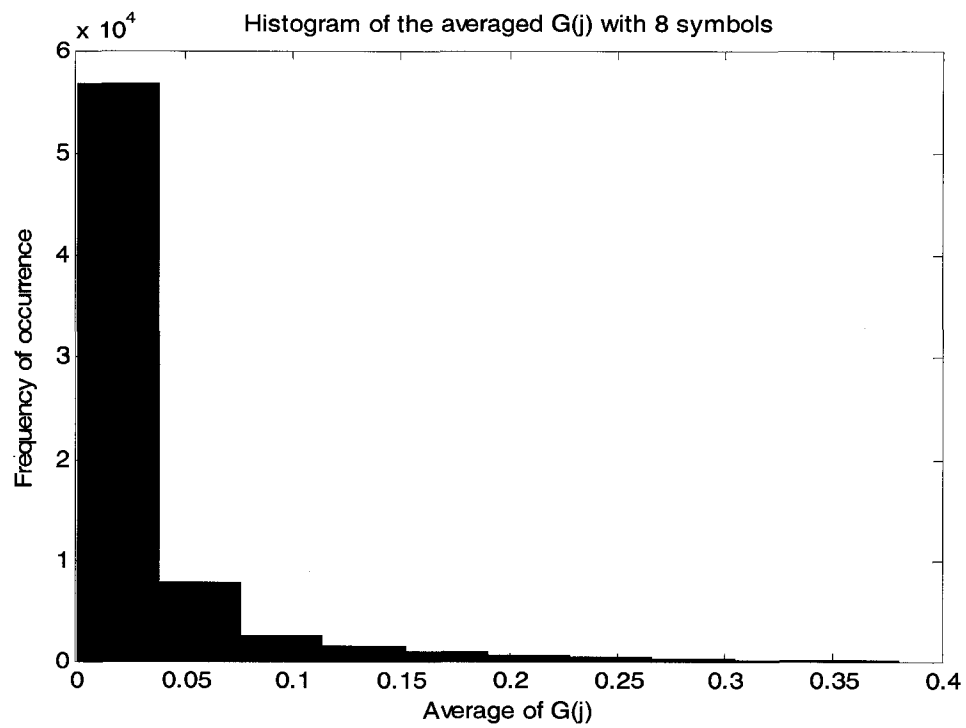


Figure 7.17 (a) Model C [2] – Windowed average with 8 symbols

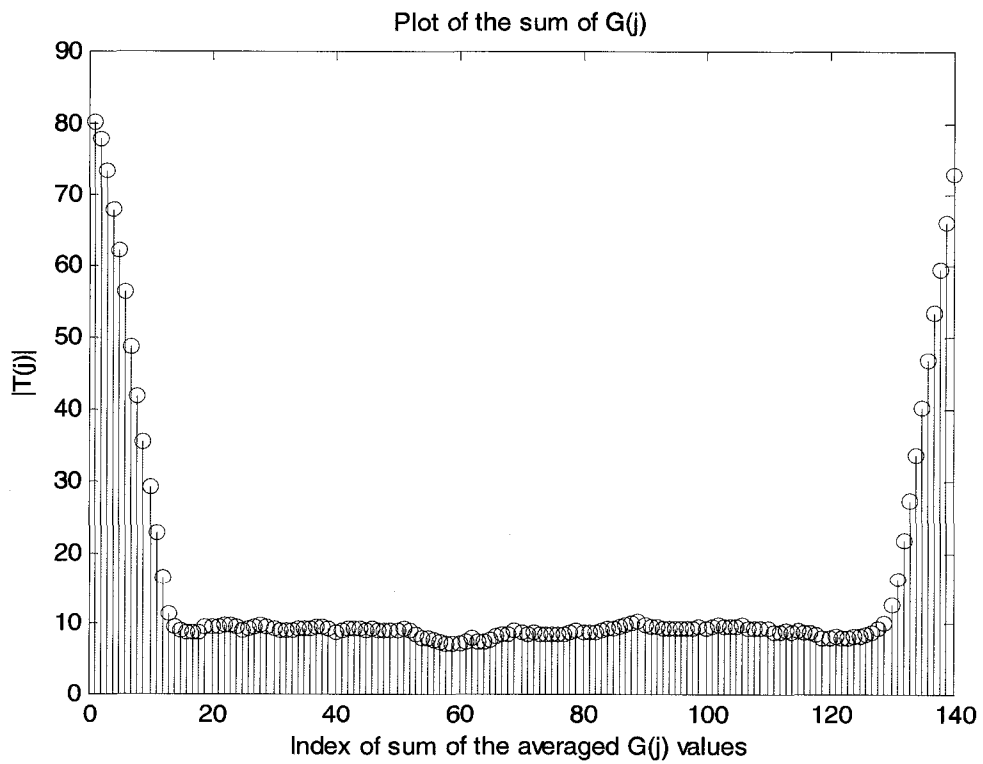


Figure 7.17 (b) Model C [2] – Sum of windowed average for 8 symbols

7.4.3.4 Averaging with a Window Size of 10

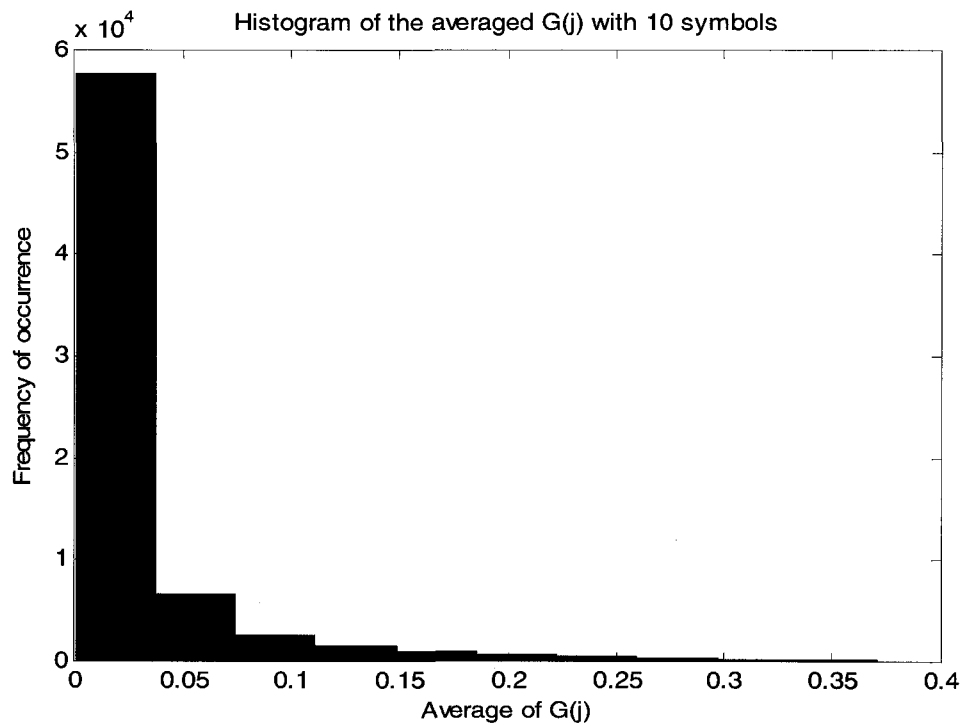


Figure 7.18 (a) Model C [2] – Windowed average with 10 symbols

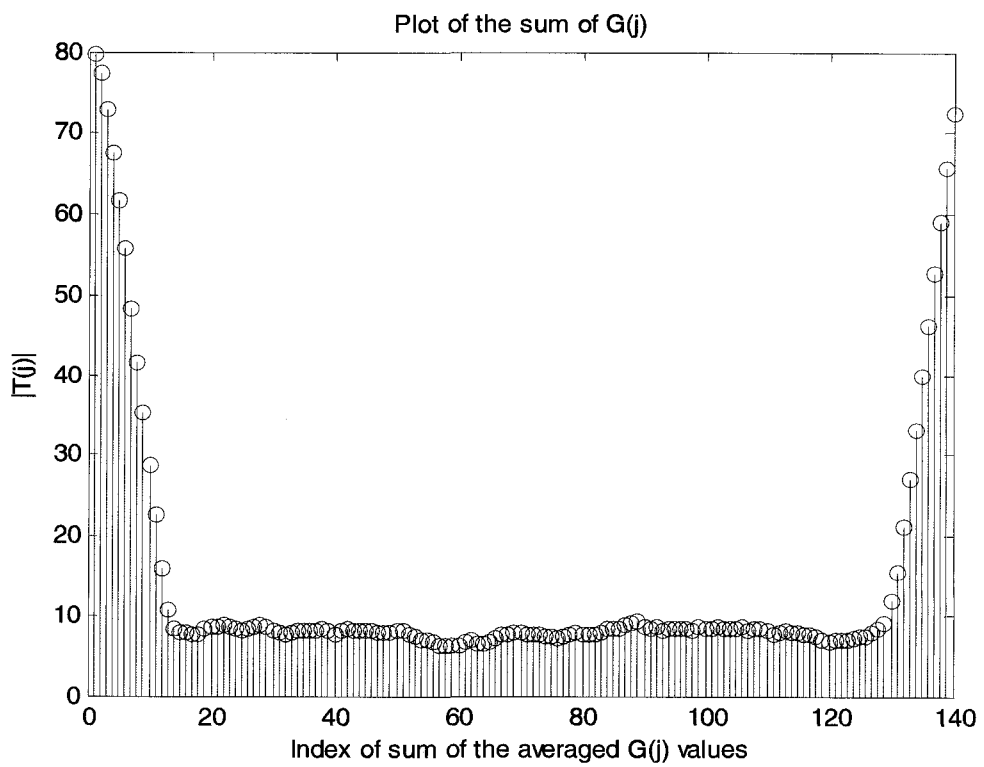


Figure 7.18 (b) Model C [2] – Sum of windowed average for 10 symbols

7.5 Model D [2]

Figure 3.5 shows the impulse response of this channel model with 7 taps. The following figures are indicative of the response of this channel to the correlation algorithms $G(j)$ and $R(j)$.

7.5.1 Raw Correlator Outputs for Model D [2]

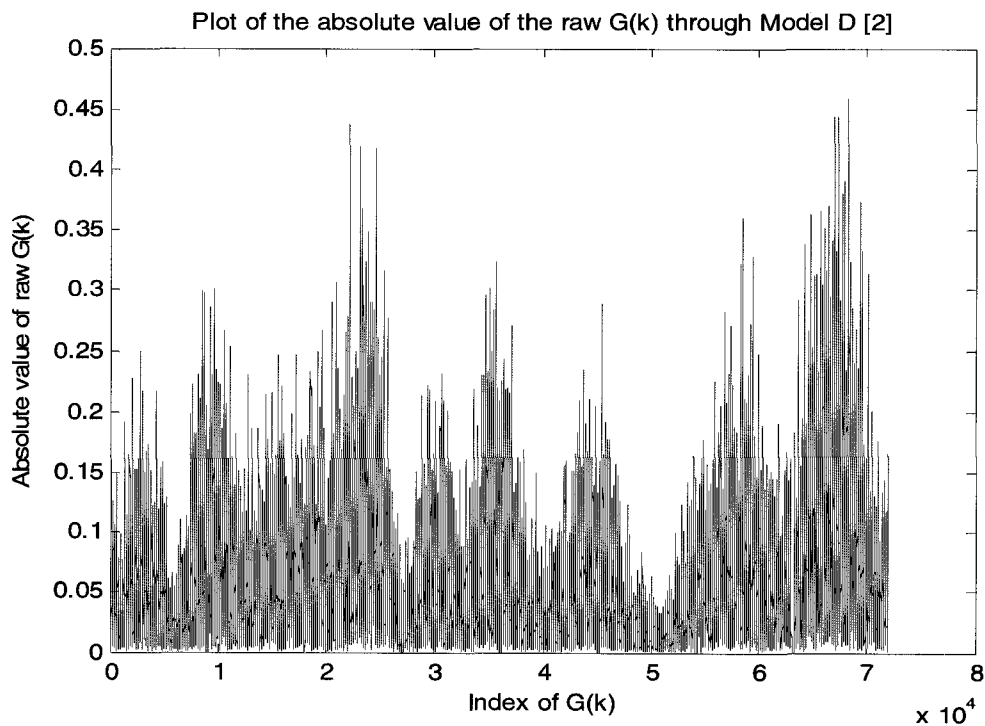


Figure 7.19 (a) Model D [2] - $G(j)$ correlator output

Stem plot of the absolute value of the raw $G(k)$ through Model D [2] (expanded)

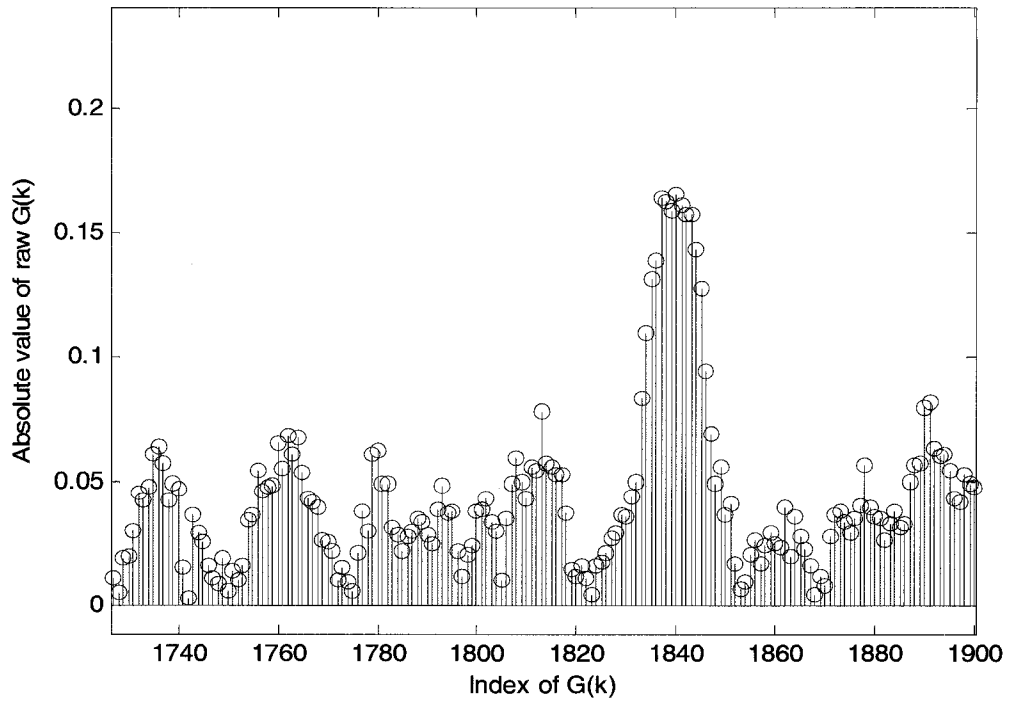


Figure 7.19 (b) Model D [2] – Expanded $G(j)$ output

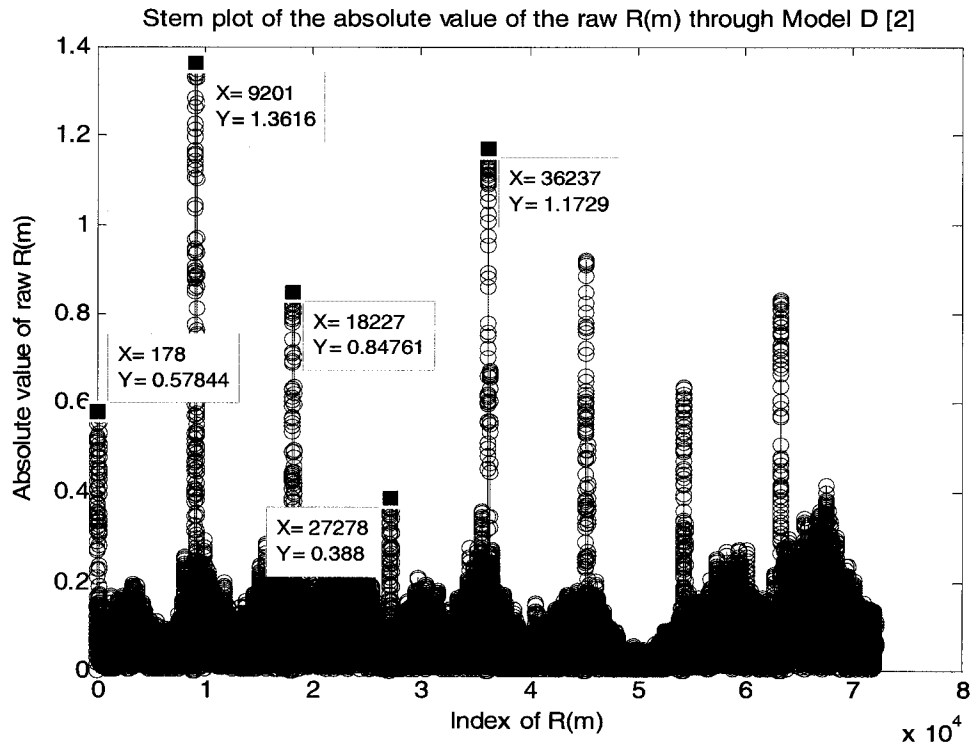


Figure 7.19 (c) Model D [2] - $R(j)$ output with markers

7.5.2 Interpeak Intervals Model D [2]

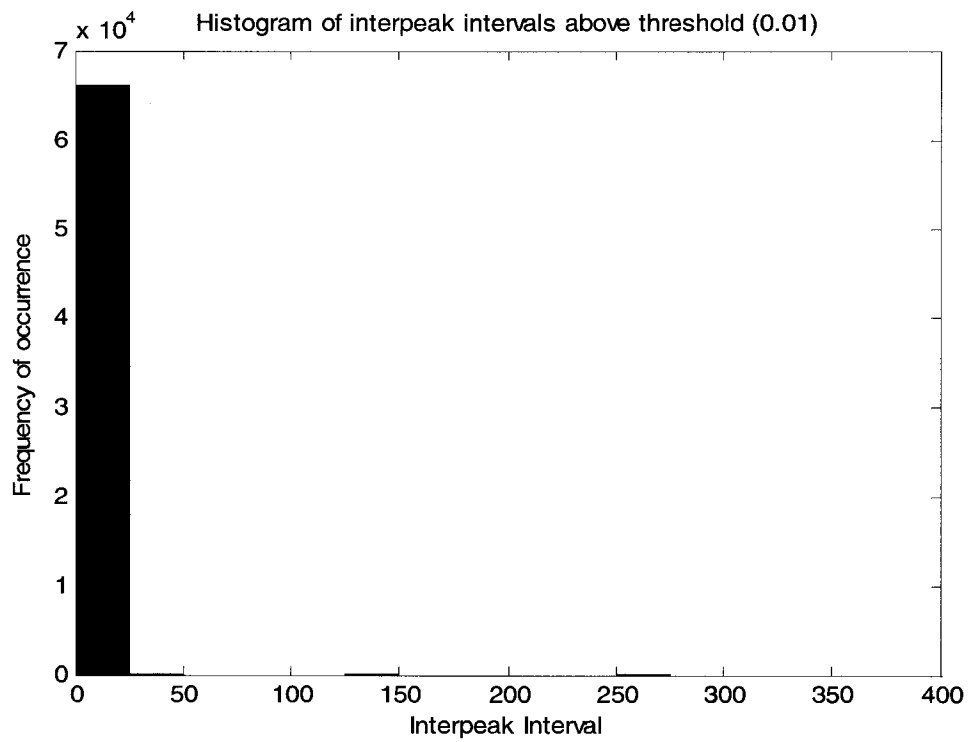


Figure 7.20 (a) Model D [2] – Interpeak intervals above 0.01

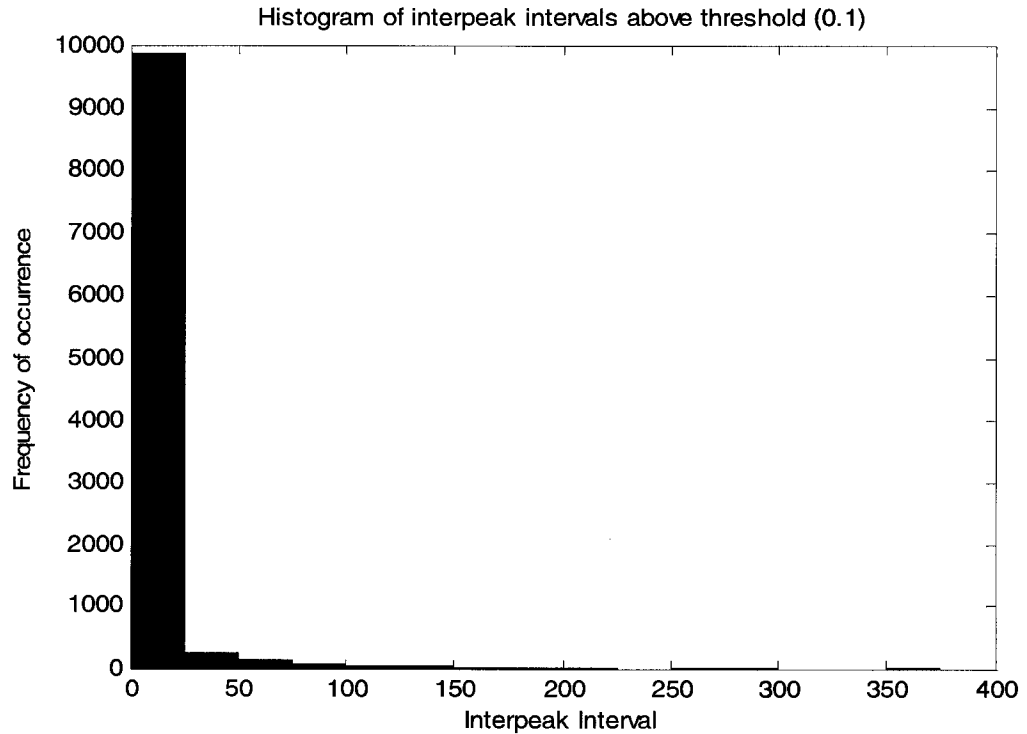


Figure 7.20 (b) Model D [2] – Interpeak intervals above 0.1

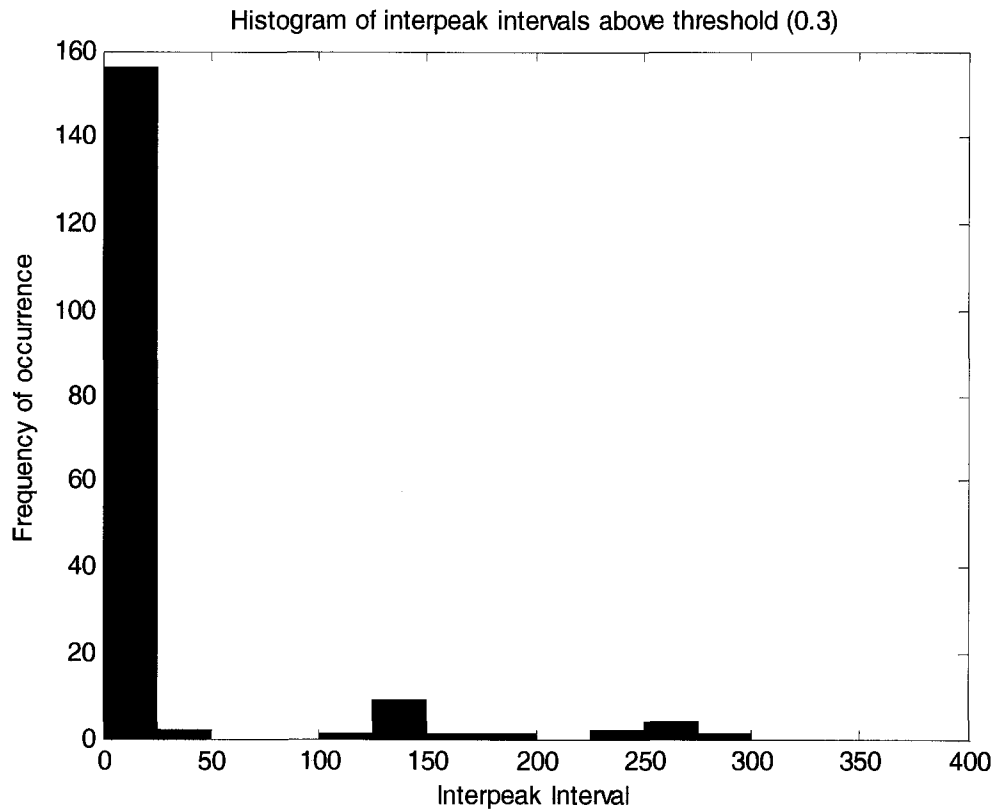


Figure 7.20 (c) Model D [2] – Interpeak intervals above 0.3

Observe in Figure 7.20 (c) that the spreading of interpeak interval distribution is about correct and double period peak was *not* seen for other channels.

7.5.3 Averaging with Variable Window Sizes for Model D [2]

7.5.3.1 Averaging with Window Size of 4

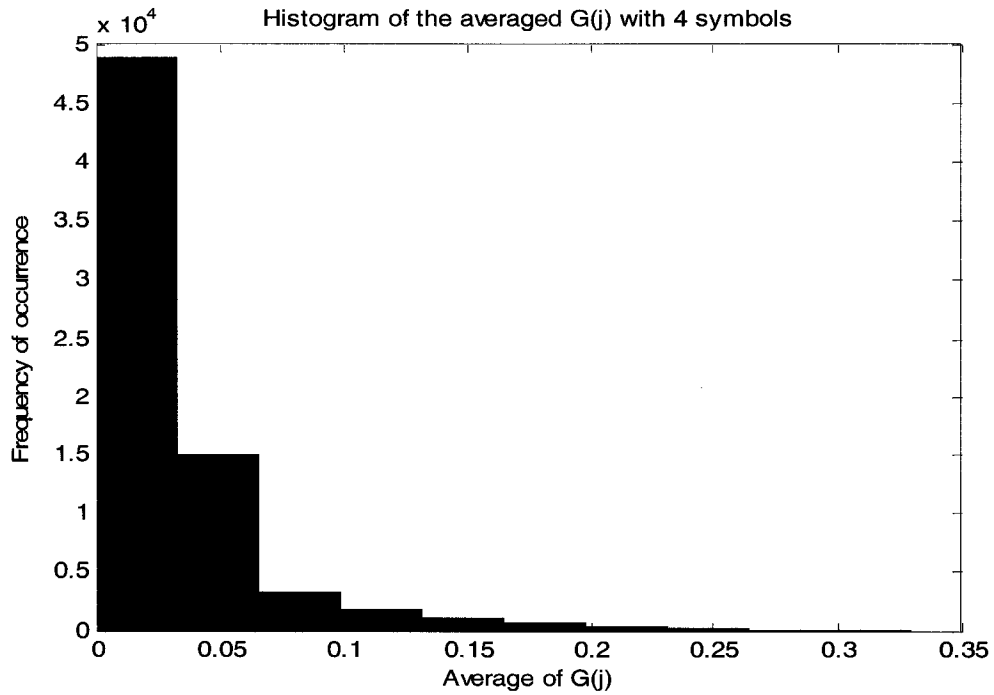


Figure 7.21 (a) Model D [2] – Windowed average with 4 symbols

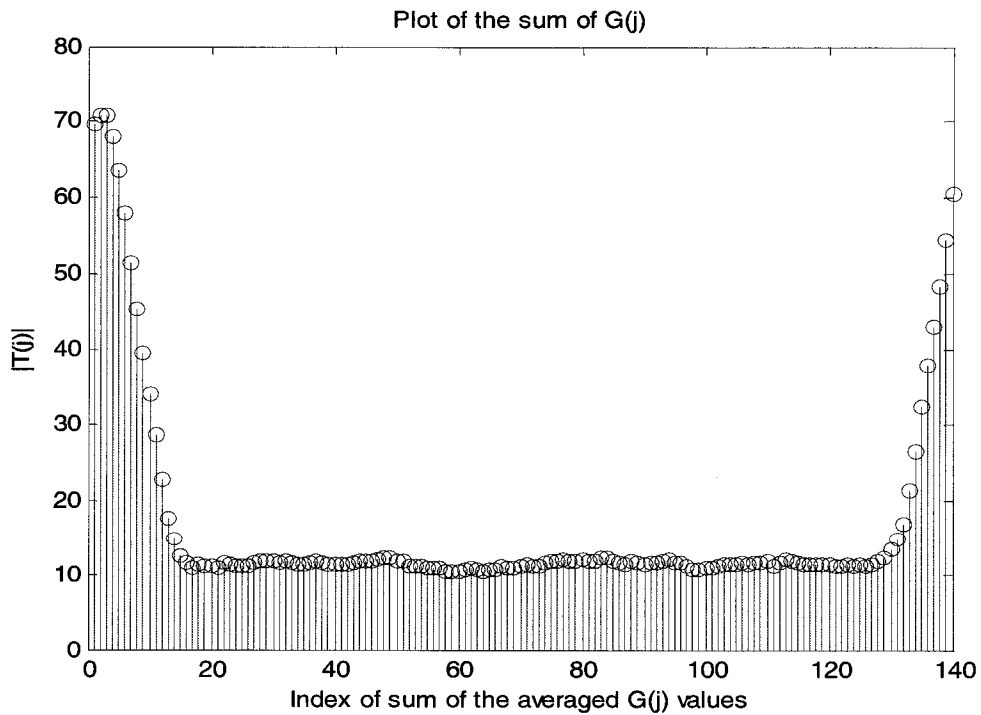


Figure 7.21 (b) Model D [2] – Sum of windowed average for 4 symbols

7.5.3.2 Averaging with a Window Size of 6

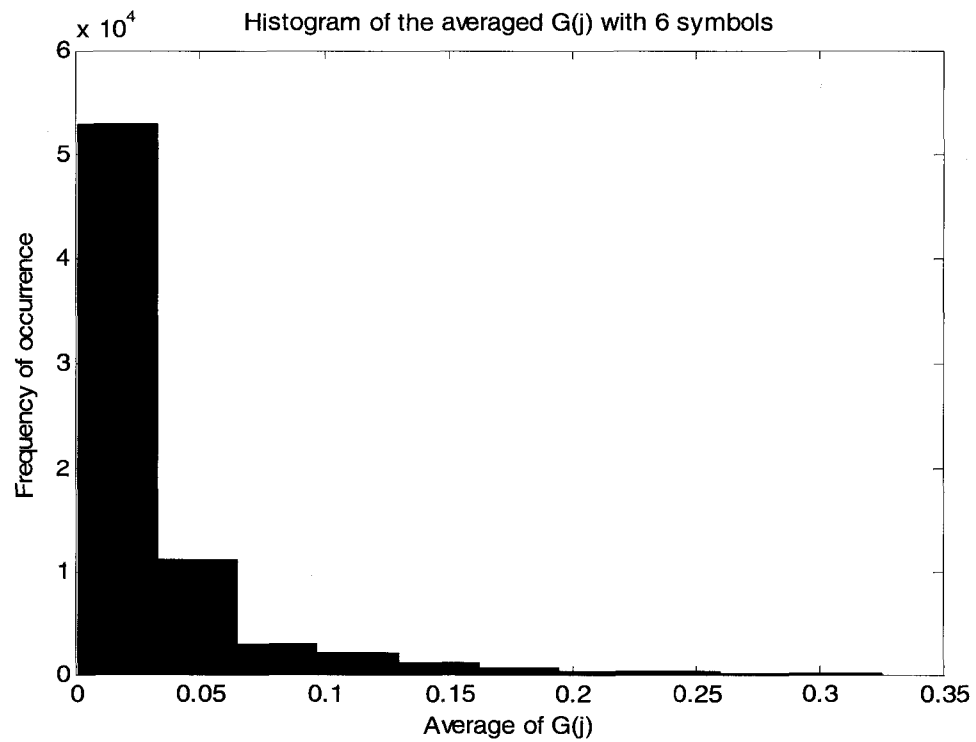


Figure 7.22 (a) Model D [2] – Windowed average with 6 symbols

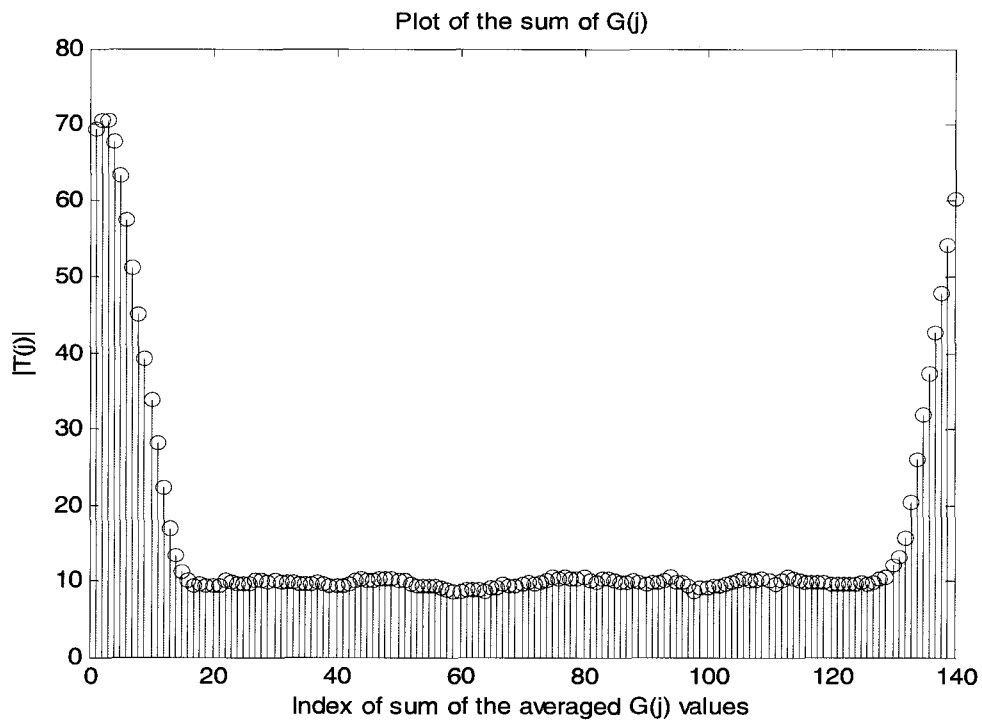


Figure 7.22 (b) Model D [2] – Sum of windowed average for 6 symbols

7.5.3.3 Averaging with a Window Size of 8

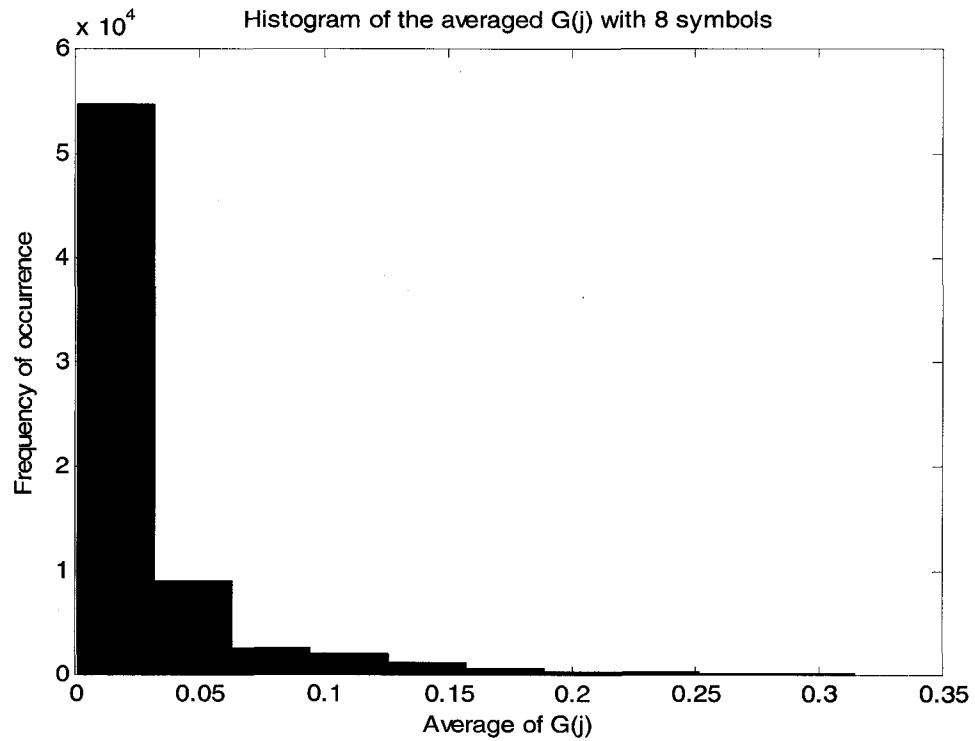


Figure 7.23 (a) Model D [2] – Windowed average with 8 symbols

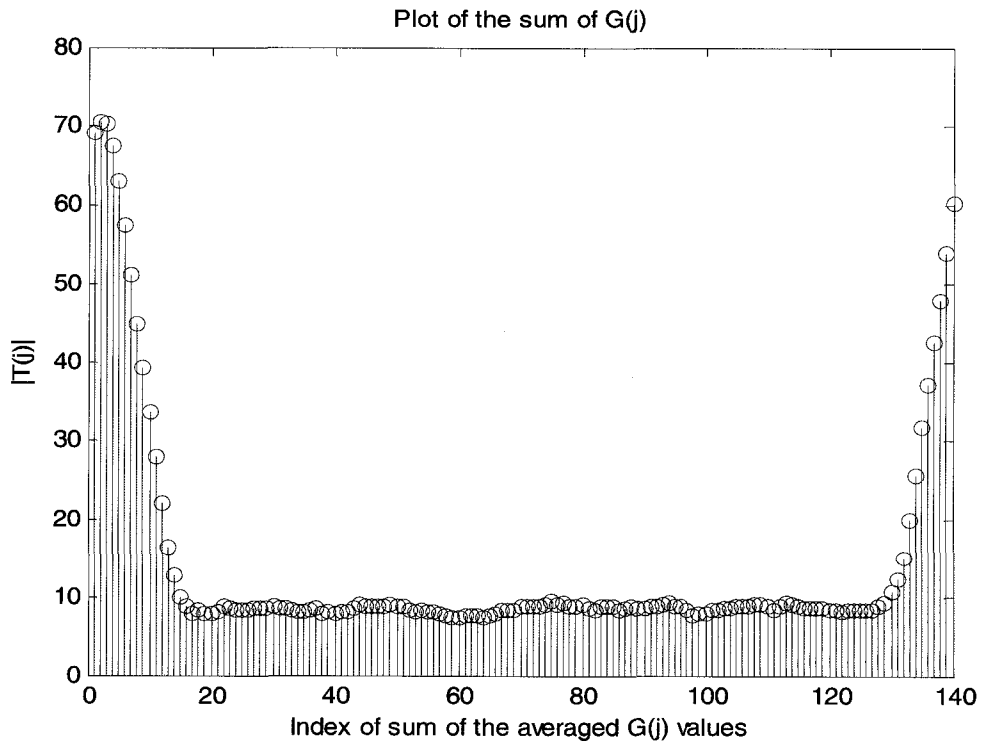


Figure 7.23 (b) Model D [2] – Sum of windowed average for 8 symbols

7.5.3.4 Averaging with a Window Size of 10

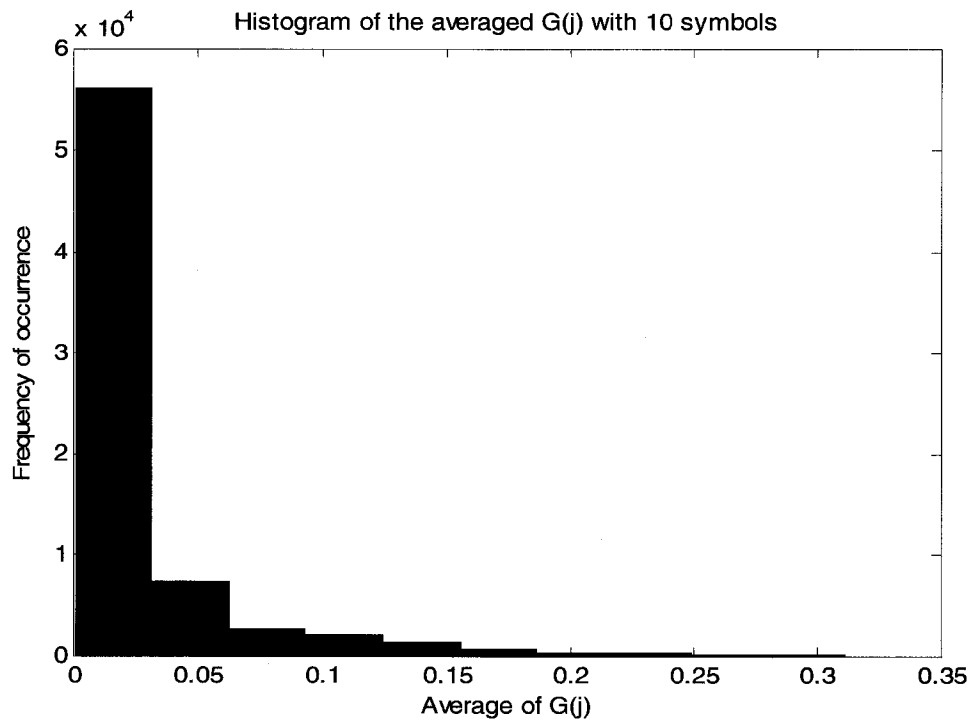


Figure 7.24 (a) Model D [2] – Windowed average with 10 symbols

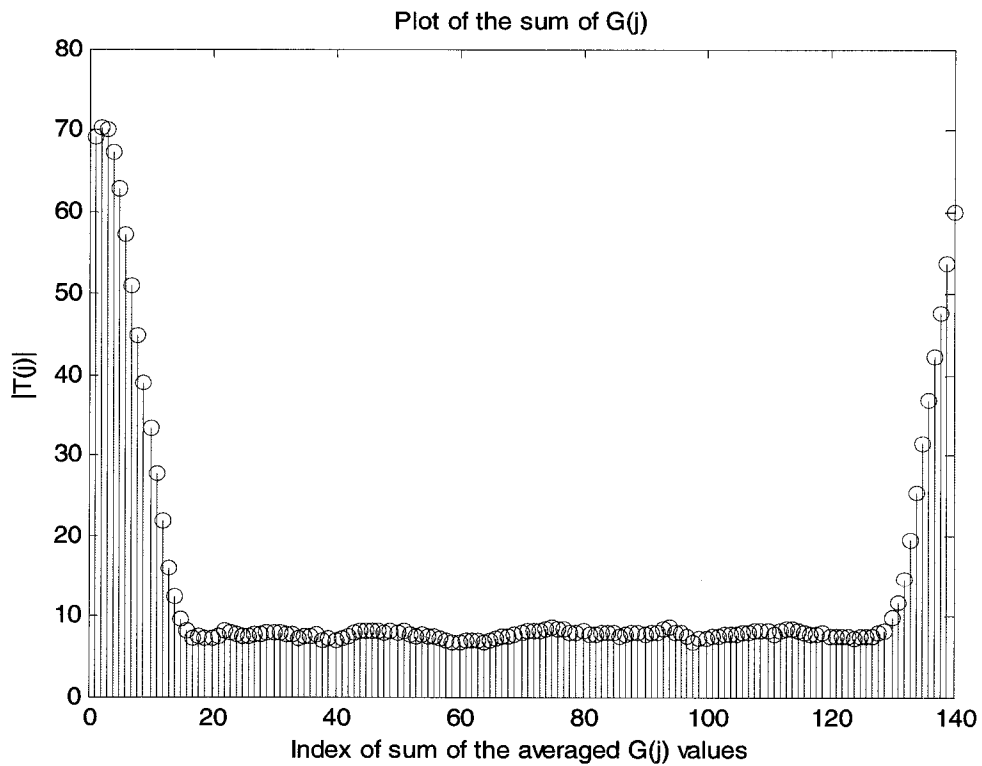


Figure 7.24 (b) Model D [2] – Sum of windowed average for 10 symbols

The general trend, although not uniformly and consistently obvious, is that averaging *sharpens* the $|G(j)|_{avg}$ distribution.

CHAPTER 8

CONCLUSIONS AND FUTURE WORK

8.1 Conclusions

The original motivation for this thesis was the existence of irreducible time and frequency offset error probability limits as the signal-to-noise ratio is increased. The question arose, “What are the causes of time and frequency offset errors using correlation-based synchronization algorithms in the limit of high SNR?” The establishment of symbol timing and tone frequency estimates with non-zero offset errors is referred to as *false lock*.

The probability of false lock in both time and frequency is strongly dependent on the correct choice of the correlator output peak location. While in principle one can delay the decision about the maximum correlator output location for a full OFDM symbol duration, it is not practical to implement a sliding window-based peak detection algorithm without specifying a detection threshold. It is this realization that led to the examination of the correlator output distributions.

It was determined that the correlator output distribution at the true peak locations is spread significantly for random data when using the QPSK modulated OFDM format. The spread of the distribution is due to the fluctuation of energies in the cyclic prefixes of the OFDM symbols. This greatly complicates the choice of peak detection threshold.

To better understand the structure of the correlator output statistics, the ensemble average of symbol-aligned correlator outputs was examined over a large number of test sequences. The resulting autocorrelation function estimate was well-behaved in the sense that the sidelobes decreased monotonically to the correlation “floor”, but the amplitudes of the closest sidelobes to the true correlation peak were very close to that of the peak itself. This close knit sidelobe structure makes it difficult to set an optimum peak detection threshold since false detection and missed detection are probable at any threshold setting.

In an effort to reduce the effective amplitude of the autocorrelation sidelobes, a windowed averaging technique was proposed and investigated. It was successful in reducing both the variance of the true correlation peak at the symbol/frame boundaries and the variance of the sidelobes. However, this technique did not render the task of setting a peak detection threshold any easier because of the very slight differences in the amplitudes of the true peak and its immediate sidelobes.

An additional finding was that the symbol and frame correlator outputs exhibited a constant high amplitude plateau around the reference symbol at the beginning of every frame. Further investigation revealed that this behavior was caused due to the presence of multiple copies of the synchronization pattern embedded inside the reference symbol. These copies align perfectly more than once, thus resulting in a very high constant amplitude correlation plateau. This in turn made it difficult to locate the single true peak exactly at the frame boundary.

The influence of fading channels on the correlator output statistics was examined, using the same methods as in the non-fading channel case, for four different Rayleigh and Rician channel models that are typical of indoor propagation environments. As some of the useful correlation peaks are lost in deep fades, one would expect the distribution of correlator outputs at the true peak location to be skewed with respect to that of the non-fading channel, and this was confirmed by experiment. The constant amplitude correlation plateau at the frame boundaries, which is due to the presence of the reference symbol, still existed, but its amplitude was frequently reduced with respect to that observed in the non-fading channel case. The technique of windowed averaging of correlator outputs was also examined for the selected channel models, and it was found that it was not as effective in reducing the autocorrelation sidelobe variance as it was for the non-fading channel. It should be noted that if deep fades occur while still receiving a sufficient number of correlation peaks, it would still be possible to process the data and make a decision on symbol and frame boundary locations in the data stream. However, in cases where there is a deep fade when reception begins, it is difficult to decide where to start counting the samples or what criteria to use in discriminating the real peaks.

While we faced some challenging issues and conducted various experiments to work our way towards the motivating question of persistent false lock at high SNR in fading channels, we were not able to get to that point and explore further the options available to us.

8.2 Suggestions for Future Work

8.2.1 Timing Offset Estimation

The focus of this thesis was on the statistical characterization of the synchronization correlator outputs rather than actually performing the complete synchronization algorithm. The present study presumed the use of simple threshold exceedance tests as a means of identifying correlator output peak locations in time. It should be possible to improve synchronization performance by using multiple correlation peaks to estimate a single value of sample offset between the current symbol boundary estimate and a corrected estimate. The approach suggested below uses a minimum mean-squared error formulation of the offset estimate.

Let P_1, P_2, \dots, P_k be the interpeak intervals between the peaks picked by the peak-picking algorithm [Chapter 5] with respect to the first peak above a decision threshold. Then the sum,

$$\sum_{k=1}^M [((N + N_g)k + Offset) - P_k]^2 = E \quad (9.1)$$

where k is the peak number and $Offset$ is the timing offset, gives the equation for the error E in choosing the wrong peaks.

The goal here is to choose the $Offset$ such that the error E is minimum i.e.,

$$\frac{\partial E}{\partial Offset} = 0$$

Hence, we have,

$$\begin{aligned}
\sum_{k=1}^M \left((N + N_g)k + \text{Offset} \right) * P_k &= 0 \\
\sum_{k=1}^M \left((N + N_g)k + \text{Offset} \right) &= \sum_{k=1}^M P_k \\
M * \text{Offset} &= \sum_{k=1}^M P_k - \sum_{k=1}^M \left(k (N + N_g) \right) \\
\text{Offset} &= \frac{1}{M} \sum_{k=1}^M P_k - \frac{(N + N_g)}{M} \sum_{k=1}^M k \\
\text{Offset} &= \frac{1}{M} \sum_{k=1}^M P_k - \frac{(N + N_g)}{M} \frac{M(M + 1)}{2} \\
\text{Offset} &= \frac{1}{M} \sum_{k=1}^M P_k - \frac{(N + N_g)(M + 1)}{2}
\end{aligned}$$

8.2.2 Other Investigations

The simulation experiments in this thesis were carried out with QPSK modulation on each subcarrier. These experiments could be repeated for higher order modulations to do a comparative study of synchronization correlator output statistics. One could examine for any modulation format the effect on the correlator output statistics of varying the length of the cyclic prefix. A dynamic threshold detector could also be implemented, which would adapt to the varying correlation peak magnitudes by maintaining a running estimate of the median peak magnitude and setting the threshold at some fixed percentage above the median value. One might develop an analytical model of the fixed threshold peak detector synchronizer and estimate the false lock probability distribution at different offsets from the true correlation peak location. This analytical study could be extended to the use of windowed averaging technique as well.

REFERENCES

- [1] Thomas Keller, Lorenzo Piazzo, Paolo Mandarini, and Lajos Hanzo, "Orthogonal Frequency Division Multiplex Synchronization Techniques for Frequency-Selective Fading Channels", *IEEE Journal on Selected Areas in Communications*, Vol. 19, Issue. 6, June 2001, pp 999-1008.
- [2] Jonas Medbo, and Peter Schramm, "Channel Models for HIPERLAN/2 in Different Indoor Scenarios", ETSI EP BRAN, 30th March 1998, Document #. 3ERI085B.
- [3] W. O. Alltop, "Complex Sequences with Low Periodic Correlations", *IEEE Transactions on Information Theory*, Vol. IT-26, Issue. 3, May 1980, pp 350-354.
- [4] Thomas Keller, and Lajos Hanzo, "Adaptive Multicarrier Modulation: A Convenient Framework for Time-Frequency Processing in Wireless Communications", *Proc. IEEE*, Vol. 88, Issue. 5, May 2000, pp 611-642.
- [5] T. Keller, L. Hanzo, "Orthogonal Frequency Division Multiplex Synchronization Techniques for Wireless Local Area Networks", *Proc. 7th Conference on Personal Indoor and Mobile Radio Communications*, Taipei, Taiwan, R.O.C, October 1996, pp 963-967.
- [6] Jorgen Bach Anderson, Theodore S. Rappaport, and Susumu Yoshida, "Propagation Measurements and Models for Wireless Communications Channels", *IEEE Communications Magazine*, January 1995, pp 42-49.
- [7] Ufuk Tureli, Hui Liu, and Michael D. Zoltowski, "OFDM Blind Carrier Offset Estimation: ESPRIT", *IEEE Transactions on Communications*, Vol. 48, Issue. 9, September 2000, pp 1459-1461.
- [8] Chien-Fang Hsu, Yuan-Hao Huang, Tzi-Dar Chiueh, "Design of an OFDM Receiver for High-Speed Wireless LAN", *IEEE International Symposium on Circuits and Systems (2001, Sydney, N. S. W)*, Vol. 4, May 2001, pp 558-561.
- [9] R. Zhang, T. T. Tjhung, H. J. Hu, and P. He, "Window Function and Interpolation Algorithm for OFDM Frequency-Offset Estimation", *IEEE Transactions on Vehicular Technology*, Vol. 52, Issue. 3, May 2003, pp 654-670.

- [10] Michael Speth, Ferdinand Classen, Heinrich Meyr, "Frame Synchronization of OFDM Systems for Frequency Selective Fading Channels", IEEE 47th Vehicular Technology Conference, Vol. 3, Issue. 4, May 1997, pp 1807-1811.
- [11] Meng-Han Hsieh, and Che-Ho Wei, "A Low-Complexity Frame Synchronization and Frequency Offset Compensation Scheme for OFDM Systems Over Fading Channels", IEEE Transactions on Vehicular Technology, Vol. 48, Issue. 5, September 1999, pp 1596-1609.
- [12] Timo Weiss, Albert Krohn, Fatih Capar, Ihan Martoyo, and Friedrich K. Jondral, "Synchronization Algorithms and Preamble Concepts for Spectrum Pooling Systems", IST Mobile and Wireless Telecommunications Summit, 2003-teco.edu, pp 788-792.
- [13] Ryan A. Pacheco, Oktay Ureten, Dimitrios Hatzinakos, and Nur Serinken, "Bayesian Frame Synchronization Using Periodic Preamble for OFDM-Based WLANs", IEEE Signal Processing Letters, Vol. 12, Issue. 7, July 2005, pp 524-527.
- [14] Kai Shi, and Erchin Serpedin, "Coarse Frame and Carrier Synchronization of OFDM Systems: A New Metric and Comparison", IEEE Transactions on Wireless Communications, Vol. 3, Issue. 4, July 2004, pp 1271-1284.
- [15] Karen Halford, and Mark Webster, "Multipath Measurement in Wireless LANs", Intersil Application Note AN9895, May 2000.
- [16] F. Mueller-Roemer, "Directions in Audio Broadcasting", J. Audio Eng. Soc., Vol. 41, Issue. 3, March 1993, pp 158-173.
- [17] G. Plenge, "DAB-A New Radio Broadcasting System-State of Development and Ways for its Introduction", Rundfunktech. Mitt, Vol. 35, Issue. 2, 1991, pp 45.
- [18] M. Alard, and R. Lassalle, "Principles of Modulation and Channel Coding for Digital Broadcasting for Mobile Receivers", EBU Review, August 1997 Tech. 224, pp 47-69.
- [19] "DAB", in Proc. 1st Int. Symp., Montreux, Switzerland, June 1992.
- [20] "Digital Video Broadcasting (DVB); Framing Structure, Channel Coding and Modulation for Digital Terrestrial Television", ETSI EN 300 744 V1.1.2, August 1997.
- [21] J. Borowski, S. Zeisberg, J. Hubner, K. Koora, E. Bogenfeld, and B. Kull, "Performance of OFDM and Comparable Single Carrier System in

MEDIAN Demonstrator 60 GHz Channel”, in Proc. ACTS Summit, Aalborg, Denmark, October 1997, pp 653-658.

- [22] “Digital Radio Mondiale (DRM); System Specification”, ETSI ES 201 980 V2.1.1, October 2005.
- [23] “Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications”, Standard, IEEE, New York, November 1997.
- [24] “High Performance Radio Local Area Network (HIPERLAN) Type 2; Requirements and Architectures for Wireless Broadband Access”, Broadband Radio Access Networks (BRAN) TR 101 031, ETSI, April 1997.

APPENDICES

APPENDIX A

MATLAB CODE

Reference Symbol Generation

```
clear all;
close all;

N = 31;

for i = 1:N;
    a(i) = (N^(-1/2))*(cos(2*pi*((i-1)^2)/N) + j*sin(2*pi*((i-1)^2)/N));
end
```

Experiments with the Generated OFDM Symbols

```
%%%%%%%%% Assumptions %%%%%%%%%%
% 1. N number of channels
% 2. QPSK modulation on each channel (2-bits per channel (the same
modulation is performed on all the N carriers))
% 3. 128-point IFFT

% This code plots the histogram of the interpeak intervals above a given
% threshold, histogram of the power in the tail of the OFDM symbols and
% averaging to improve OFDM symbol timing synchronization.

clear all;
close all;

% %%%%%%%%%% Initializing the Rayleigh fading channels %%%%%%%%%%
% % Keller channel model [1]
% chan1 = rayleighchan((25*10^(-9)),10000,[0 25*10^(-9) 48.9*10^(-
9)],[(20*log10(0.9)) (20*log10(0.35)) (20*log10(0.3))]);
% chan1.StoreHistory = 1;
%
% % ETSI paper - Model B
% chan2 = rayleighchan((10*10^(-9)),10000,[0 10*10^(-9) 20*10^(-9) 30*10^(-9)
40*10^(-9)],[0 -5.4 -10.8 -16.2 -21.7]);
```

```

% chan2.StoreHistory = 1;
%
% % ETSI paper - Model C
% chan3 = rayleighchan((10*10^(-9)),10000,[0 10*10^(-9) 20*10^(-9) 30*10^(-9)
40*10^(-9) 50*10^(-9)],[0 -2.1 -4.3 -6.5 -8.6 -10.8]);
% chan3.StoreHistory = 1;
%
% ETSI paper - Model D
chan4 = rayleighchan((10*10^(-9)),10000,[0 10*10^(-9) 20*10^(-9) 30*10^(-9)
40*10^(-9) 50*10^(-9) 60*10^(-9)],[0 -0.9 -1.7 -2.6 -3.5 -4.3 -5.2]);
chan4.StoreHistory = 1;

% y = 20; % 'y' indicates the number of correlated sequences that have to be
generated to find the ensemble
% for z = 1:y; % Generating y different correlated sequences for finding the
ensemble

N = 128; % OFDM symnol size i.e., total number of channels
CP = 13; % Guard interval size (cyclic prefix)
NS = 31; % Length of sync pattern
n = 512; % Total number of symbols including the null and reference symbols

% Load the OFDM data file
OFDMdata = load('OFDMdataFor8Frames.mat');
OFDMframes = OFDMdata.OFDMdata1;

% Filtering the OFDM symbols through the Rayleigh fading channel
h = filter(chan4,OFDMframes);
pathgain = chan4.PathGains; % Accessing the channel path gains
% plot(chan);
%
% % % % % Frequency tracking and OFDM symbol synchronization % % % % %
sfp = (length(OFDMframes)-(N+CP)); % Samples for plot ('sfp' i.e., the total
number of samples taken for the plot of the correlation function G(j))
for k = 1:sfp; % Index 'k' of the correlation function G(k) is set to the end of the (n-
1)th symbol
    G(k) = ((h(k+N))*conj(h(k)));
    for Ng = 1:(CP-1);
        G(k) = G(k) + ((h(k+Ng+N))*conj(h(k+Ng))); % Computing the correlation
function G(k)
        Gabs(k) = abs(G(k)); % Absolute value of G(k)
    end
end
end

% % % % % Frequency acquisition and OFDM frame synchronization % % % % %

```

```

% numofsamples = (length(OFDMframes)-(N+CP-NS)); % Number of samples
for computing the correlation function R(m)
% for m = 1:numofsamples;
%   R(m) = ((h(m+NS))*conj(h(m)));
%   for interval = 1:(NS-1);
%       R(m) = R(m) + ((h(m+interval+NS))*conj(h(m+interval))); % Computing
the correlation function R(m)
%       Rabs(m) = abs(R(m)); % Absolute value of R(m)
%   end
% end
%
% %%% Plot of the absolute value of the raw correlation peaks %%%
% figure(1);
% plot([1:sfp],Gabs); % Plots the absolute value of raw G(k)
% xlabel('Index of G(k)');
% ylabel('Absolute value of the raw G(k)');
% title('Plot of the absolute value of the raw G(k)');
% figure(2);
% stem([1:sfp],Gabs); % Stem plot of the absolute value of raw G(k)
% xlabel('Index of G(k)');
% ylabel('Absolute value of the raw G(k)');
% title('Stem plot of the absolute value of the raw G(k)');
% figure(3);
% plot([1:numofsamples],Rabs); % Plots the absolute value of raw R(m)
% xlabel('Index of R(m)');
% ylabel('Absolute value of the raw R(m)');
% title('Plot of the absolute value of raw R(m)');
% figure(4);
% stem([1:numofsamples],Rabs); % Stem plot of the absolute value of raw R(m)
% xlabel('Index of R(m)');
% ylabel('Absolute value of the raw R(m)');
% title('Stem plot of the absolute value of the raw R(m)');

% %%% Histogram of interpeak intervals %%%
% threshold = 0.3; % Setting a threshold value for detecting the peaks and to find
the interpeak interval
% count2 = 0; % Initializing the variable 'count2' to 0
% for p = 1:sfp; % p varies from 1 to sfp (the number of samples taken for the
plot of the correlation function)
%   if (abs(G(p)))>threshold % Checking if the absolute value of G(p) is greater
than the threshold
%       count2 = count2+1; % Incrementing 'count2' by 1
%       index(:,count2) = p; % The variable 'index' will have the indices of the
G(p)'s greater than the threshold
%   end
% end

```

```

% % Finding the index difference to get the interpeak interval
% diff(1) = index(1);
% for q = 1:(length(index)-1); % q varies from 1 to the length of the variable
'index' (which has the indices of G(p) greater than the threshold) minus 1
%   diff(q+1) = index(q+1) - index(q); % Calculating the difference in the index of
two peaks greater than threshold to find the interpeak interval
% end
% maxdiff = max(diff); % Getting the maximum value of the interpeak interval;
% % Plot of the histogram of the interpeak interval
% figure(13);
% Edges = [0 25 50 75 100 125 150 175 200 225 250 275 300 325 350 375
400];
% bins = histc(diff, Edges); % Histogram of the interpeak interval
% bar(Edges, bins, 'histc');
% xlabel('Interpeak Interval');
% ylabel('Frequency of occurrence');
% title('Histogram of interpeak intervals above threshold');
% xlim([0 400]);

% % % % % Histogram of power in the tail of the OFDM symbol % % % % %
% y = []; % Initializing the vector 'y'
% count3 = 0; % Initializing the variable 'count3' to 0
% for i2 = 1:n; % i2 varies from 1 to the number of OFDM symbols 'n'
%   count3 = count3+1; % Incrementing 'count3' by 1
%   y = OFDMframes(count3:count3+12); % Getting the 13 samples that
account for the cyclic prefix (tail of OFDM symbol)
% %   tailpower(i2) = sum(xcorr(y)); % Finding the tailpower by computing the
autocorrelation function of the samples in the CP
% % Variation in finding the tail power
%   tailpower1(i2) = 0; % Initializing the vector 'tailpower1' to 0
%   for j1 = 1:length(y); % j1 operates on the vector 'y'
%       tailpower1(i2) = tailpower1(i2)+(y(j1)*conj(y(j1))); % Computing the
tailpower for each case
%   end
%   count3 = count3+140; % Incrementing 'count3' by 140 to get the tail
information of the next OFDM symbol
% end
% % Plot of the histogram of the tail power
% % figure(14);
% % hist(tailpower); % Histogram of the tail power
% % xlabel('Bins');
% % ylabel('Tail power');
% % title('Histogram of the power only at the known end of OFDM symbols');
% figure(15);
% hist(tailpower1); % Histogram of the tail power
% xlabel('Tail power');

```



```

% ylabel('Frequency of occurrence');
% title('Histogram of the power only at the known end of OFDM symbols');
%%%%%%%% Averaging to improve G(j) %%%%%%%%%
K = 11; % Setting the window size for filtering/averaging (actual window size is
10 OFDM symbols)
noftemp = ((K-1)*(N+CP))+1; % Number of values of G(k) corresponding to a
window size of i3
i3 = sfp-(noftemp); % Calculating the number of times the following loop has to
be executed
temp = [G(1:noftemp)]; % Loading the first i3 values of the correlation function
G(k) into 'temp'
count4 = 1; % Initializing the variable 'count4' to 1
while (count4 <= i3);
    T(count4) = temp(end); % Calculating the first expression in the summation
    for i4 = 1:(K-1);
        T(count4) = T(count4) + temp(end-(i4*(N+CP))); % Calculating the sum over
the window size
    end
    TavG(count4) = T(count4)/K; % Averaged value of 'T'
    magT(count4) = abs(TavG(count4)); % Calculating the magnitude of the sum
'T(count4)'
    temp = [G((count4+1):(noftemp+count4))]; % Left shift and loading the next i3
values of G(k) into 'temp'
    count4 = count4 + 1; % Incrementing the variable 'count4' by 1
end
% Plot histogram of the average of G(j) over a window size of K OFDM symbols
figure(9);
hist(magT); % Histogram of the averaged G(j)
xlabel('Average of G(j)');
ylabel('Frequency of occurrence');
title('Histogram of the averaged G(j)');

% Plot of the averaged G(j)
figure(10);
plot([1:i3],magT); % Plots the index of magnitude of 'T' versus 'magT'
xlabel('Index');
ylabel('Magnitude of the averaged values');
title('Plot of the averaged G(j)');
xlim([1 i3]);

%%%%%%%% Averaging the averaged G(j) %%%%%%%%%
count5 = 1; % Initializing the variable 'count5' to 1
symformagT = (length(magT)+1)/(N+CP); % Computes the total number of
symbols that correspond to the length of 'magT'
for j2 = 1:(N+CP-1);

```

```

    sumTavg(j2) = magT(count5); % Assigning the first value of 'Tavg' that is to be
summed to 'sumTavg'
    for i5 = 1:(symformagT-2);
        sumTavg(j2) = sumTavg(j2) + magT(count5+(i5*(N+CP)+1)); % Computing
the sum for the length of Tavg
    end
    count5 = count5 + 1; % Incrementing the variable 'count5' by 1
end
% Plot of the average of averaged G(j)
figure(11);
plot([1:(N+CP-1)],sumTavg); % Plots the index of 'sumTavg' versus its value
xlabel('Index of sum of the averaged G(j)');
ylabel('|T(j)|');
title('Plot of the sum of G(j)');
figure(12);
stem([1:(N+CP-1)],sumTavg); % Stem plot of the index of 'sumTavg' versus its
value
xlabel('Index of sum of the averaged G(j) values');
ylabel('|T(j)|');
title('Plot of the sum of G(j)');
%
%
%
% H(z,:) = G; % Putting the correlated output G(k) in a single matrix for finding
the ensemble of the sequences
% P(z,:) = R; % Putting the correlated output R(m) in a single matrix for finding
the ensemble of the sequences
% end
% %%% Ensemble of z correlated sequences %%%
% E = mean(H,1); % Computes the ensemble of the matrix 'H' along each
column
% F = mean(P,1); % Computes the ensemble of the matrix 'P' along each column
% figure(5);
% plot([1:sfp], abs(E));
% title('Ensemble of G(k) for z number of sequences');
% xlabel('Computation windows');
% ylabel('Absolute value of E');
% xlim([1 sfp]); % Setting the limit of X-axis
% % ylim([0 0.25]); % Setting the limit of Y-axis
% figure(6);
% stem([1:sfp], abs(E)); % Stem plot of the ensemble
% title('Stem plot of the ensemble of G(k) for z number of sequences');
% xlabel('Computation windows');
% ylabel('Absolute value of E');
% xlim([1 sfp]); % Setting the limit of X-axis
% % ylim([0 0.25]); % Setting the limit of Y-axis

```

```
% figure(7);
% plot([1:numofsamples], abs(F));
% title('Ensemble of R(m) for z number of sequences');
% xlabel('Computation windows');
% ylabel('Absolute value of F');
% xlim([1 numofsamples]); % Setting the limit of X-axis
% % ylim([0 0.25]); % Setting the limit of Y-axis
% figure(8);
% stem([1:numofsamples], abs(F)); % Stem plot of the ensemble
% title('Stem plot of the ensemble of R(m) for z number of sequences');
% xlabel('Computation windows');
% ylabel('Absolute value of F');
% xlim([1 numofsamples]); % Setting the limit of X-axis
% % ylim([0 0.25]); % Setting the limit of Y-axis
```

APPENDIX B

IEEE COPYRIGHT PERMISSION LETTER

Yes

Sincerely,

Jacqueline Hansson
IEEE Intellectual Property Rights Office
445 Hoes Lane
Piscataway, NJ 08855-1331 USA
+1 732 562 3828 (phone)
+1 732 562 1746 (fax)
e-mail: j.hansson@ieee.org

Hi Jacqueline,

If that is the case, can I go ahead and include a hard copy of this email (if required) along with my thesis? Thanks again for your response.

Regards,
Anupama.

Dear Anupama Sankaran,

Thank you for sending the information I requested. This office receives a large amount of permission requests and considers its email replies to be official, as our email bears the IEEE Master Brand and the office information. Most publishers find email permission grants acceptable, as we do. It enables all parties to handle the process in a time-saving and efficient manner, particularly in a business where such requests are numerous and deadlines occur continuously. We hope that the University of New Hampshire will concur.

Sincerely,

Jacqueline Hansson.
IEEE Intellectual Property Rights Office
445 Hoes Lane
Piscataway, NJ 08855-1331 USA
+1 732 562 3828 (phone)

+1 732 562 1746 (fax)
e-mail: j.hansson@ieee.org

Hi Jacqueline,

Thanks for your response. Below is the information that you had asked for:

1. Thomas Keller, Lorenzo Piazzo, Paolo Mandarini and Lajos Hanzo, "Orthogonal Frequency Division Multiplex Synchronization Techniques for Frequency-Selective Fading Channels", IEEE Journal on Selected Areas in Communications, Vol. 19, No. 6, June 2001, pp 999-1008, Figures 6, 8, 9, 10 and Equations 1, 2, 3, 4, 5, 6, 7.
2. W. O. Alltop, "Complex Sequences with Low Periodic Correlations", IEEE Transactions on Information Theory, Vol. IT-26, No. 3, May 1980, pp 350-354, all equations from this paper.

I've already credited the source for figures that I've included so far but didn't know about the IEEE copyright until before I got your reply. I will include that as well. Will you be sending a formal letter or something that I can include along with my thesis as proof that I've taken your permission?

Thanks,
Anupama.

Dear Anupama Sankaran,

This is in response to your letter below, in which you have requested permission to reprint, in your upcoming thesis/dissertation, IEEE copyrighted figures. We are happy to grant this permission.*

Our only requirements are that you credit the original source (author, paper, and publication), and that the IEEE copyright line (© [Year] IEEE) appears prominently with each reprinted figure.

* Please be advised that in regards to photos, you must get permission from the owner of the photo, as IEEE does not hold copyright on most photos.

To complete the permission process, please use the reply function and send me the list of the IEEE publication titles, paper titles, author names and figure numbers which you'll be using. Thank you.

Sincerely yours,

Jacqueline Hansson
IEEE Intellectual Property Rights Office
445 Hoes Lane
Piscataway, NJ 08855-1331 USA

+1 732 562 3828 (phone)
+1 732 562 1746 (fax)
e-mail: j.hansson@ieee.org

On Thu Jun 21 17:43:28 2007,

The following information was submitted:

Name = Anupama Sankaran

Member = Member

MemberNumber = 41618460

Subscriber = Subscriber

Institution = University of New Hampshire, IEEE chapter

Author =

Publication =

PubTitle =

Other =

OtherName =

Submit_to = copyrights@ieee.org

Comments = I'm a graduate student at the University of New Hampshire, Durham, New Hampshire, working towards the completion of my Masters thesis and would like to get permission for citing figures from IEEE journals and conference papers in my Masters thesis document.