Theses   Theses and Dissertations

Fall 2006

Channel-predictive link layer ARQ protocols in wireless networks

Huseyin Dogukan Cavdar
New Jersey Institute of Technology

ABSTRACT

CHANNEL-PREDICTIVE LINK LAYER ARQ PROTOCOLS IN WIRELESS NETWORKS

by
Huseyin Dogukan Cavdar

Communication performance over a wireless channel should be considered according to two main parameters: energy and throughput. The reliable data transfer is a key to these goals. The reliable node-to-node data transfer is performed by link layer protocols. One prominent approach is Automatic Repeat Request (ARQ) protocol. The traditional ARQ protocols attempt to recover the erroneously transmitted frames by retransmitting those frames, regardless of the channel state. Since this channel state unaware behaviour may cause unnecessary retransmissions, traditional ARQ protocols are expected to be energy inefficient. Some ideas have been proposed such as stochastic learning automaton based ARQ, and channel probing based ARQ. However, these algorithms do not attempt to estimate the channel's existing condition. Instead, the retransmission decision is made according to a simple feedback, on whether the previous frame was successful.

This thesis presents four proposed algorithms, which incorporates the channel state estimate in the feedback process to judiciously select a frame (re)transmission timing instant. Algorithms have been applied on Stop-and-Wait (S-W) ARQ, and the performance have been compared with respect to simple S-W ARQ, and probing based S-W ARQ. In probing based ARQ, when the channel deteriorates, transmitter starts probing channel periodically, but the periodicity is chosen arbitrarily, regardless of the fading state. In contrast, the proposed algorithms estimate the channel's existing condition by using feedbacks, and the probing interval is chosen according to the Average Fading Duration (AFD) of received signal. Simulations are performed with Rayleigh Fading Channel. The performance results show that at the cost of some additional delay, significant gain on energy saving and throughput performance can be achieved when AFD based intelligent probing is done.

CHANNEL-PREDICTIVE LINK LAYER ARQ PROTOCOLS IN WIRELESS NETWORKS

by
Huseyin Dogukan Cavdar

A Thesis
Submitted to the Faculty of
New Jersey Institute of Technology
in Partial Fulfillment of the Requirements for the Degree of
Master of Science in Electrical Engineering

Department of Electrical and Computer Engineering

January 2007

Blank Page

APPROVAL PAGE

CHANNEL-AWARE LINK LAYER ARQ PROTOCOLS IN WIRELESS NETWORKS

Huseyin Dogukan Cavdar

Dr. Swades De, Thesis Advisor   Date
Assistant Professor of Electrical and Computer Engineering, NJIT

Dr. Ali Abdi, Committee Member   Date
Assistant Professor of Electrical and Computer Engineering, NJIT

Dr. Hongya Ge, Committee Member   Date
Associate Professor of Electrical and Computer Engineering, NJIT

BIOGRAPHICAL SKETCH

Author: Huseyin Dogukan Cavdar

Degree: Master of Science

Date: January 2007

Undergraduate and Graduate Education

• Master of Science in Electrical Engineering,
  New Jersey Insitute of Technology, Newark, NJ, 2006

• Bachelor of Science in Electrical and Electronics Engineering,
  Bilkent University, Ankara, Turkey, 2005

Major: Computer Networking

To My Parents, My Brother and My Grandmother,

ACKNOWLEDGEMENT

TABLE OF CONTENTS

TABLE OF CONTENTS
(Continued)

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1   Overview of ARQ Protocols

Reliable reception and transmission of data is an essential point in communications between communicating devices [12]. Basically, there are two fundamental techniques for maintaining reliable and efficient communication over noisy channels: *forward-error-correction (FEC) schemes* and *automatic-repeat-request (ARQ)* [6]. An example of telephone conversation from real life can be considered, as noted in [11]. The listener says "OK", if the message is understood, or the listener asks to repeat the sentence, if the message is somehow not understood correctly(i.e, the message is corrupted). This feedback based mechanism assures that sender will know whether the message is transmitted correctly or not. In computer networking, the protocols that provides the reliable reception and transmission of data under the mechanism mentioned above are called *Automatic Repeat Request (ARQ) protocols*. In ARQ protocols, the feedback messages are called *Acknowledgements*. There are two types of possible acknowledgements: The ACK (positive) refers to successful transmission, and the NAK (negative) refers to corruption at transmission. There are three ways of ARQ protocols, which will be discussed in detail in the following chapter:

- Stop-and Wait (S-W) ARQ,

- Go-Back-N (GB-N) ARQ,

- Selective-Repeat (S-R) ARQ.

Basically, ARQ protocols includes three features to detect the erroneously packets [11]. There should be an error detection mechanism so that the receiver can detect the occured errors in messages. Besides, the receiver should inform the sender about the status of the received packet by acknowledging the sender. Moreover, erroneously received packet

1

should be retransmitted. That is, ARQ protocol with error detection and retransmission can provide reliable data transmission over a communication channel [4].

FEC provides the detection of the errors that is followed by process of error correction [12]. FEC is generally preferred when there is no return channel and retransmission to correct a few number of errors is inefficient [12]. When there is an availability of a return channel, ARQ is preferred.

There is also another form of ARQ known as *hybrid ARQ* [19]. In this form of ARQ, FEC is used to improve the efficiency of ARQ mechanisms, especially when the channel conditions are poor [10]. The receiver first attempts to correct any errors, and if it determines the error sequence is one that can not be corrected, it sends a NAK to the transmitter [19]. There are two kinds of hybrid ARQ schemes [19]:

- Type-1 Hybrid ARQ : Upon the receipt of the NAK, the transmitter resends the same codeword as on the original one [19],

- Type-2 Hybrid ARQ : Upon the receipt of the NAK, the transmitter resends additonal redundancy [19].

However, in our discussion, it is assumed that if any message is not recovered correctly, a retranmission request is made from receiver to transmitter.

There are also another mechanism used by transport protocols to detect the packet loss [5]. All the transmitted data is acknowledged by the receiver, and a lost packet is signaled once the timeout period is reached before the feedback arrives [5]. This end-to-end error recovery mechanism is called *Transport Layer Retransmissions* [5]. TCP and TP4(Transport Protocol 4) are example transport layer protocols operating in this manner [5]. However, in this study, node-to-node link-layer ARQ protocols is the focus mechanism.

1.2 Overview of Propagation Models

Modeling the mobile radio channel has always been questioned by mobile radio system designers because the radio channels are not stationary [17]. Propagation models have been

developed, which mainly aim to estimate the average signal strength at the receiver. There are two types of propagation models:

- Large-Scale Propagation Models: Propagation models that predicts the signal strength at receiver over large distances.

- Small-Scale Propagation Models: Propagation models that predicts the signal strength at receiver over short distances. They characterize the rapid fluctuations of the signal at receiver.

The types of large-scale and small-scale propagation models are discussed in CHAPTER 3. Besides, only the models, used in simulation , are discussed in detail. Log-normal shadowing is chosen as a path loss model and rayleigh fading model is chosen as a mobile radio channel.

1.3   Motivation and Related Work

The transmission media used by wireless data networks shows varying behaviour due to multipath propagation, the position wireless users relative to the radio server, user mobility, and non-stationary clutter [8]. The result of this situation is that different users sharing a common radio server are liable to experience different channel conditions at the same time [9]. This is referred to as *multiuser diversity*, the basis of opportunistic scheduling(OS) [9]. For this reason, OS is referred as to multiuser diversity scheduling, or channel-aware scheduling [9]. Efficient functioning of OS requires the feedback mechanism that provides control of scheduling [9]. Usually, maximum SINR based MAC scheduling is used, where the schedular picks up the user among all active users on the system, which has the best signal-to-noise-ratio, or equivalently, the best feasible data rate [3]. In thesis, it is assumed that the best user was already chosen before.

In wireless communication networks, the demand for high data rates and quality of service (QoS) is getting bigger [13]. But, the performance of wireless links is degraded due to channel fading [13]. In order to increase the throughput, adaptive modulation and coding (AMC) schemes have been studied at the physical layer [13]. This provides transmission rates to match with the varying channel conditions [13]. An alternative way to decrease the

effect of fading is rely on ARQ protocol at the data link layer, which request retransmissions for the erroneously received packets [13]. Retransmissions are activated only at necessary situations so ARQ is pretty effective in enhancing the system throughput [14]. One of ARQ protocols, Stop-and-Wait ARQ protocol is studied in this thesis.

The main purpose of this thesis is to figure out what solutions can be proposed to estimate the communication channel's existing situation while transfering data so that the proposed solution maintains energy efficiency. That is, the transmitter should not send data if it senses that the channel is not convenient for communication. Up to now, some opinions have been proposed, but all of them were based on the past conditions, which give only intuition about the channel's state. Besides, it is not sufficient information because it does not guarantee that the channel is going to pursue in that manner. A detailed information about what has been done on predicting channel's state is discussed in the last part of CHAPTER 2. In this thesis, an algorithm is proposed such that it provides better efficiency and consumes less energy. The estimation of channel's existing condition is approximated so that there is more reliable reason to do transmission or not.

1.4   Organization

Remaining part of thesis is organized in four chapters: Chapter 2 presents traditional ARQ and surveys the prior channel dependent ARQ protocols. Chapter 3 surveys the channel propagation models and also contains the channel simulation details that is used in the thesis. Chapter 4 contains the proposed channel-aware ARQ protocols. The protocol details are firstly described, and then the protocol performance is compared in terms of throughput, energy efficiency, and delay efficiency with respect to traditional S-W ARQ and probing based ARQ. In Chapter 5, a few remarks on the relative performance results are made and possible future directions of work are outlined. Finally, an appendix is included, that contains MATLAB source code of the simulation.

CHAPTER 2

ARQ PROTOCOLS

An ARQ protocol has the capabilities of error detection and retransmission, and it provides a means of reliable data transfer over an error-prone communication channel. There are three types of basic ARQ protocols, Stop-and-Wait, Go-Back-N, and Selective Repeat. A brief overview of these basic ARQ schemes are presented in the following sections.

2.1 Stop-and-Wait

The simplest type of ARQ protocols is Stop-and-Wait Protocol. The scenario of this protocol is as follows [11]: The transmitter waits for a packet from an upper layer. Once it gets a packet, a packet is transmitted to the receiver. Now, there are two ways. If a packet is received correctly, the ACK (positive acknowledgement) is sent to the transmitter to inform about the successful reception of a packet. In contrast, if a packet is somehow corrupted, the NAK (negative acknowledgement) is sent to the transmitter to request a retransmission of the lost packet. This is the basic principle of the protocol.
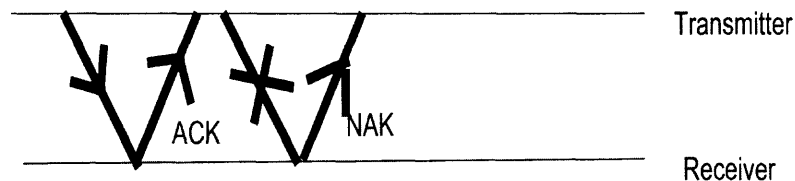
Figure 2.1 The simple scenario at Stop-and-Wait ARQ Protocol.

Unfortunately, this mechanism does not work in a perfect manner [11]. For instance; if the feedback message is corrupted, the transmitter normally does retransmission after some time, which is called "Time-out Period". However, after retransmission, the receiver

will have had two packets, and since there is no identification paramter, the receiver becomes undecided about whether the second packet is the same as the first packet or it is completely different packet. In order to overcome this problem, "Sequence Numbers" are added to the packets so that the receiver can identify the packets.
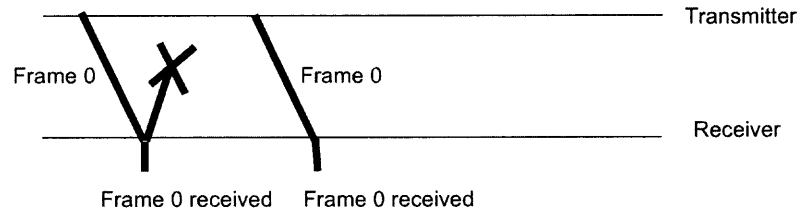
Figure 2.2 Sequence numbers are added to the header of the packet for identification.

As a second scenario, the corruption of a data packet is considered [11]. When a corrupted packet is received, the receiver sends the NAK to the transmitter or an ACK for the last correctly received packet is sent. However, the transmitter will have had two ACKs, and since there is no identification parameter on ACK packets, it becomes confused. The solution is same as the previous proposed solution, to append a "Sequence Number". Another
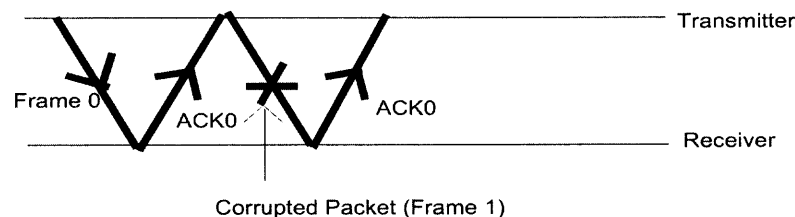
Figure 2.3 Sequence numbers are added to the header of the ACK packet for identification.

important issue is how long the transmitter should wait when a packet in either direction has been lost. This duration should be at least as the "Round-Trip Time", which is the time for complete cycle of transmission of a data packet and reception of a feedback packet.
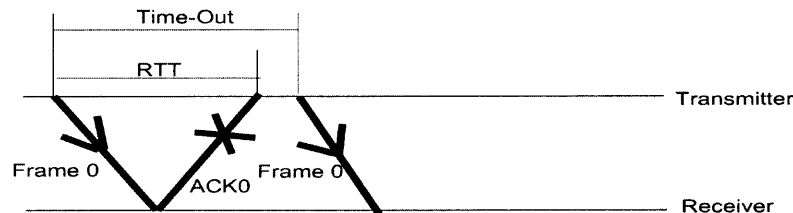
Figure 2.4 Time-Out and Round-Trip Time Relation.

2.2 Go-Back-N

Stop-and-Wait ARQ protocol with sequence numbers and well-determined RTT functionally works well. However, if efficiency aspect is considered, the protocol is inefficient. To understand the problem and to create a solution, it is enough to look at the figure below:
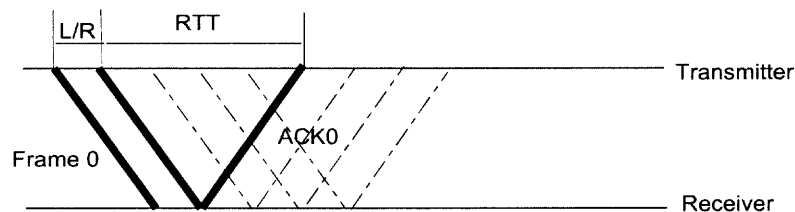
Figure 2.5 Pipelining.

The transmitter is busy only during the period of L/R, transmission time, and waits for feedback without doing anything until the time duration "L/R + RTT" is reached. It is obvious that if the transmitter sends more packets until that specific time is reached, the efficiency will enhance. This property is called "Pipelining". Go-Back-N protocol is based on pipelining. However, there is a limit at the number of packets that can be sent until the transmitter changes its mode from transmission to reception. This number is called "Window Size", and is generally represented by N. In the following figure, the example of a GB-N process is given with N=7 [4]. Round-trip time and timeout periods are given in terms of number of blocks, where RTT = 4 blocks and timeout = 7 blocks. The forward channel transmissions are not shown other than the first one, but the erroneous received

packets are marked as "error" /citechang. Besides, when a timeout occurs, the packets that have been sent, but not ackknowledged, are retransmitted.
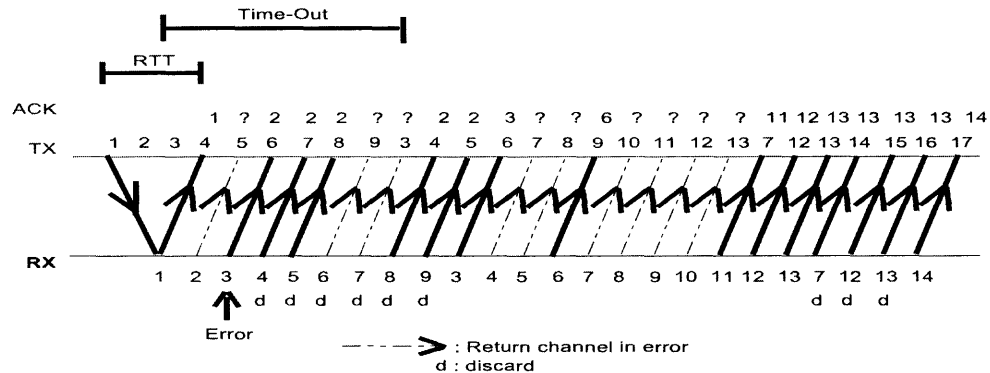
Figure 2.6 Go-Back-7. N is given by 7 in this case.

As the process of data transfer continues, the window slides so that Go-Back-N ARQ protocol is also called "Sliding-Window Protocol".

When compared to Stop-and-Wait, Go-Back-N is much more efficient since the transmitter keeps transmitting the packets until the reception of the feedbacks. However, the receiver discards out-of-order received packets. That is, eventhough a packet is received correctly, it is discarded if it is out-of-order. This disadvantage is overcome by the next ARQ protocol, called Selective-Repeat ARQ.

2.3 Selective-Repeat

As mentioned in the previous section, there are some cases that Go-Back-N ARQ protocol does not maintain good performance. Especially, when the window size, N, is too large, this deficiency gains more importance. At that scenario, even if only one packet fails to be sent, all the previously transmitted (but not acknowledged) packets will be retransmitted. However, Selective-Repeat (S-R) avoids this behaviour. In this protocol, only the corrupted packets are retransmitted within the number of window size. If there occurs an error at the reception of a packet, and the next packet is received correctly, the next packet is acknowl-

edged. That is, the order of packets is not important anymore in this protocol [11]. The following figure is an example of Selective-Repeat ARQ protocol with window size, N= 7, RTT= 4 packets, and timeout = 7 packets [4].
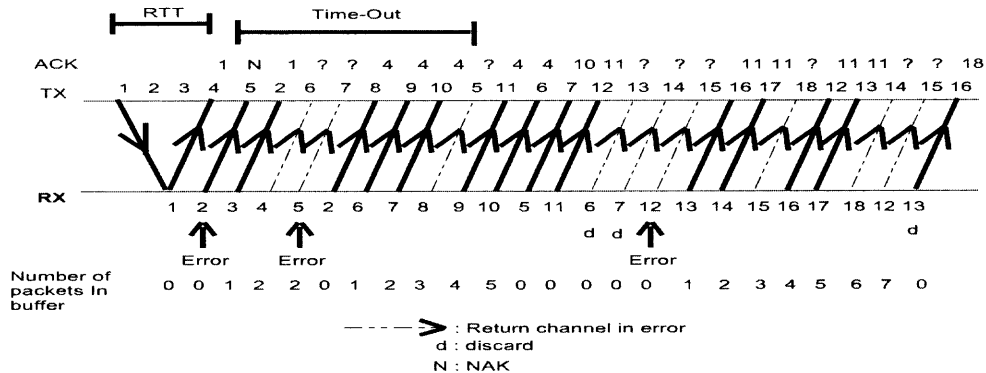
Figure 2.7 Selective-Repeat.

2.4 Prior Channel Predictive ARQ Protocols

Recently, there have been some attention drawn on channel-aware link-layer retransmission approach [21, 18]. This is very important in terms of energy efficiency because traditional ARQ protocols try to recover erroneous packets by retransmission of those packets regardless of the state of a communication channel [21]. This fact shows that traditional ARQ protocols are channel unaware and energy inefficient [18]. Especially, since wireless networks, which have randomly time-varying link characteristics, are subject to bursts of error prone behaviour, protocols that are adaptive to this randomly time varying channel conditions are intended to apply [18]. The main principle of those protocols is to predict the channel conditions at the transmission or reception time instant in order to decide to allow transfer or wait for better channel conditions. Absolutely, instead of transfering data at the time instant, when the channel conditions are not convenient for communication, it is much more energy efficient and more intelligent to wait for suitable time instant. There have been some approaches to this field, which can be considered "energy-efficient protocols".

Prior channel dependent protocols try to estimate the channel's condition based on binary feedback (via ACK or NAK), and adjust the data packet transmission accordingly.

Approach 1: One approach is to count the number of successive ACKs or NAKs [15]. Adaptive modulation based on immediate feedback is proposed in [7]. Besides, if some number of successive ACKs are received during the "BAD" state of channel (high error rate mode), it shows that the channel condition is getting better so it is the time to change the mode to "GOOD" state (low error rate mode). In same manner, if some number of successive NAKs are received during the "GOOD" state of channel (low error rate mode), it shows that the channel condition is impairing so it is the time to change the mode to "BAD" state (high error rate mode). The numbers are found by trial. However, finding the optimum values in a time-varying fading channel is unfeasible. Moreover, this approach, the count of successive feedbacks, gives the information about past, and accordingly it is said that the channel conditions might show similar characteristics at the transmission time instant, which is not true.

Approach 2: As an another approach, a link layer protocol is proposed such that it predicts the channel state by using the given past condition of a channel [18]. A finite state Markov (FSM) model is used to model a wireless channel. There are two states: "BAD" which means the channel is in deep fade, and "GOOD" which means it is convenient time to transmit a packet. Two state Markov model is called "Gilbert-Elliot Model". The principle of this protocol is as follows: There are two action options, transmit a packet or do not transmit a packet. According to the feedback from receiver, if the transfer is successful, the probability of transmitting the next packet increases, otherwise it decreases. Also, there is an algorithm, which updates the probalities of the actions, which is called "Stochastic Learning Control Algorithm based on Variable Structure Learning Automaton (VSLA)". At first glance, there are two questionable parts in this algorithm. The first questionable part is that it still relies on the past conditions, which does not reflect the existing condition of the channel. It only gives some intuation. Secondly, when you look at the algorithm,

there are some constants such as a, b, A or B, determining the updating process of the probabilities of the actions, which are arbitrary numbers. That is, they are not the numbers based on any claim.

Approach 3: Zorzi and Rao [21] proposed a probing ARQ protocol. Two independent first-order Markov models are used to characterize the forward and reverse channels. Both channel flow directions are subject to erroneous transfer. The acknowledgements are used to track the condition of the channel. This protocol operates as follows: The transmitter keeps sending packets as long as both channels are in "GOOD" state. Once either channel's condition impairs, the transmitter changes it mode from "Normal Mode" to "Probing Mode". In probing mode, the transmitter probes the channel periodically by sending probing packets, which are small-sized compared to data packets. This mode continues until ACK is received, which shows that the channel conditions is convenient for transmission. Up to now, this protocol seems to be the best one. However, it has some deficiencies. First of all, the decision is still based on acknowledgements, which give information about the condition of fading channel in the past, not now. Besides, the probing period is arbitrarily chosen, that is, it does not depend on any parameter.

CHAPTER 3

PROPAGATION MODELS

It is not easy to analyze and estimate the characteristics of wireless channels, and this issue has been one of the most difficult part in designing and modeling mobile radio system [17]. Some propagation models have been proposed and they have mainly tried to estimate the average received signal strength at a given distance. Propagation models can be categorized into two groups:

- Large-Scale Propagation Models,

- Small-Scale Propagation Models.

3.1   Large-Scale Propagation Models

Large-Scale Propagation Models try to estimate the average signal strength over large Transmitter-Receiver seperation distances. The first and most basic model is "Free-Space Propagation Model" [17]. As understood from its name, this propagation model is valid when there is a clear Line-of-Sight(LOS) between transmitter and receiver. This model is genereally used in satellite communication systems and microwave radio links [17]. The equation for this model is given by Friis Free Space Equation:

$$P_r(d) = \frac{P_t G_t G_r \lambda^2}{4\pi^2 d^2 L} \tag{3.1}$$

where P_r is the received power in terms of distance, P_t is the transmitted power, G_t and G_r are the antenna gains of transmitter and receiver respectively, d is the distance between transmitter and receiver, L is the loss factor, and λ is the wavelength in meters.

$$\lambda = \frac{c}{f} \tag{3.2}$$

where f is the carrier frequency and c is the speed of light.

The signal attenuation in decibel(dB), the difference between the transmitted power and the received power in dB, is called *"Path Loss"*. For Free- Space model, it is given as [17]:

$$PL(dB) = 10log_{10}\frac{P_t}{P_r} \tag{3.3}$$

There are three parameters that affect propagation in a mobile communication system, which I will give only the brief definitions [17].

- Reflection : When a propagating wave hits on a surface that has bigger dimensions compared to the wavelength of a wave, reflection occurs.

- Diffraction: When a path on a communication link is obstructed by an object that has sharp edges, diffraction occurs.

- Scattering : When there are many smaller objects compared to the wavelength of a wave on the path, scattering occurs.

There are some path loss models that are used to predict the signal-to-noise ratio (SNR) in mobile communication systems, which is an important parameter [17].

- Log-Distance Path Loss Model: According to this model, received average power decreases logarithmically with the distance. The formula is given as:

$$PL_{avg}(d) = PL_{avg}(d_o) + 10nlog_{10}\frac{d}{d_o} \tag{3.4}$$

where both path losses are average values, n is the path loss component that varies according to the environment, and d_o is the close-in reference distance, which is 1 km in large coverage cellular systems, and 100 or 1 m in microcellular systems. The value of n is 2 in free space, between 2.7 and 3.5 in urban area, and between 3 and 5 in shadowed urban areas [17].

- Log-normal Shadowing Model: The previous model works well. However, for instance, two points, which have same T-R seperation distances, might have different characteristics, and the log-distance path loss model does not mention about this fact.

According to the measurements, the path loss at a particular point shows a feature of log-normal distribution [17]. The formula is given as:

$$PL(d)[dB] = PL_{avg}(d) + X_\Theta = PL_{avg}(d_o) + 10nlog_{10}\frac{d}{d_o} + X_\Theta \qquad (3.5)$$

where X_Θ is a zero-mean (with Θ^2 variance) Gaussion distributed random variable (in dB). Besides, the received power is calculated as:

$$P_r(d)[dBm] = P_t(d)[dBm] - PL(d)[dB] \qquad (3.6)$$

The following figure is the plot of "Path Loss versus distance", where distance varies from 1 m to 100 m. n, the path loss component, is chosen as 3, which is the value used for urban areas. d_0 (reference distance) is chosen as 1 m, which is common for in-door environments. f_c (carrier frequency) is chosen 2.4 Ghz, which is the case in IEEE 802.11b standards. The Gaussian variable is generated randomly by MATLAB, and is converted into dBm. c, the speed of light, is 3×10^8 m/sec. L, the loss factor is chosen as 1 that shows there is no loss in the system hardware [17]. And finally, G_t and G_r, the antenna gains are chosen as 1, which refers to unity gain.
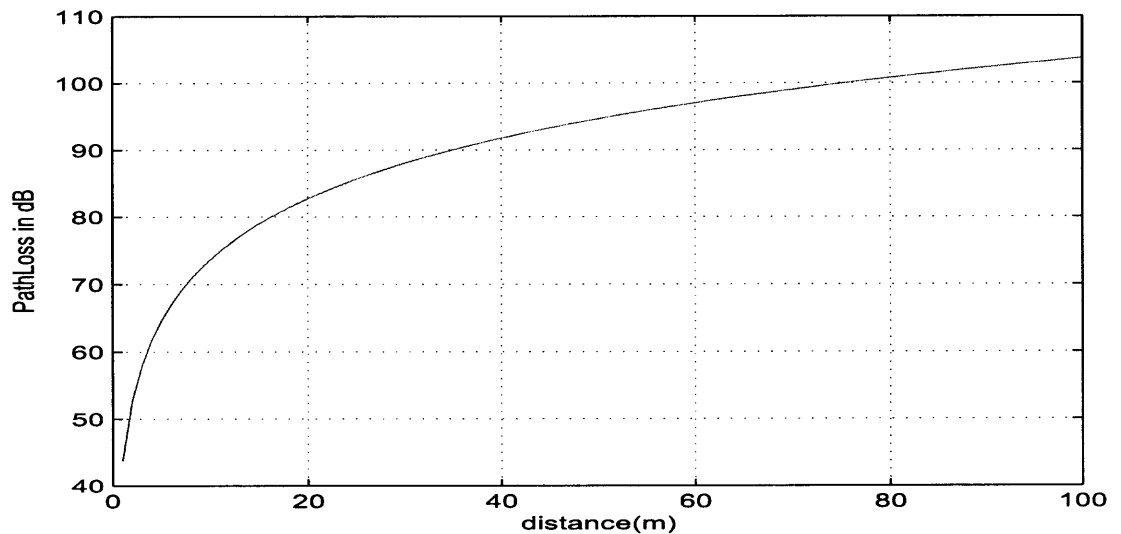
Figure 3.1 Path Loss using Log-normal Shadowing with the given parameters.

There is no always Line-of-Sight(LOS) during mobile communication, that is, there are obstacles such as trees or buildings on the communication path so propagation models should consider the presence of this irregular terrain [17]. Some of commonly used outdoor propagation models are as follows [17]:

- Longley-Rice Model: Used between 40 MHz and 100 GHz frequency range in point-to-point communications.

- Durkin's Model: Same as the Longley-Rice model, but the only difference is the assumption that the propagation is modeled by only LOS and diffraction from edges.

- Okumura Model: One of the most used models especially in urban areas between 150 MHz and 1920 MHz frequency range.

- Hata Model: Empirical formulation of Okumura Model between 150 MHz and 1500 MHz frequency range.

- PCS Extension to Hata Model: Since Hata model does not consider Personal Communications Systems(PCS), it was extended so that it covers PCS between 150 MHz and 2 GHz frequency range.

- Walfisch and Bertoni Model: This model considers the effects of rooftops.

- Wideband PCS Microcell Model: This model is developed for extensive measurements in LOS and obstructed areas.

The introduction and advancement of PCSs requires the modeling of radio propagation inside buildings [17]. The indoor radio channel has two main differences compared to an ordinary mobile radio channel. Firstly, the channel environment features vary much more. Secondly, the covered distances are smaller [17]. There are two commonly used indoor propagation models:

- Log-Distance Path Loss Model

- Ericsson Multiple Breakpoint Model

3.2 Small-Scale Propagation Models

Before giving the small-scale propagation models, it is better to give some definitions that are necessary to understand the small-scale fading. Small-fading is to characterize the rapid fluctuations at the received signal over a short time or short distance. Small-scale fading

can be also called only "fading" [17]. The interference between the versions of a signal at the receiver causes fading, and these different versions are called "multipath waves" [17]. There are some observed effects such as rapid changes in signal strength over a small time interval or distance, or variation at Doppler frequency on different multipath waves, which are the results of multipath [17].

There are some factors affecting small-scale fading [17]:

- Multipath Propagation,

- Speed of the Mobile,

- Speed of surrounding Objects,

- Transmission Bandwidth of the signal.

There is a very important parameter, called "Doppler Shift". From the following figure, there is a car(including mobile) moving from point A to point B with a speed of V, and receiving a signal from a point C, base station. Due to the relative motion between the mobile and the base station, each multipath wave experiences a shift in frequency [17]. This shift in frequency is called **Doppler Shift** and the formula to calculate is given as follows:

$$f_d = \frac{v}{\lambda} cos\Theta \tag{3.7}$$

Now, let discuss the types of small-scale fading. Small-scale fading can be categorized according to the transmitted signal with the characteristics of a channel [17]. Based on this categorization, there are two main categories [17]:

- Small-scale fading based on multipath time delay spread,

- Small-scale fading based on Doppler spread.

Moreover, small-scale fading based on multipath time delay spread can be grouped into two groups: If the mobile channel has a constant gain, and the bandwidth of the signal is
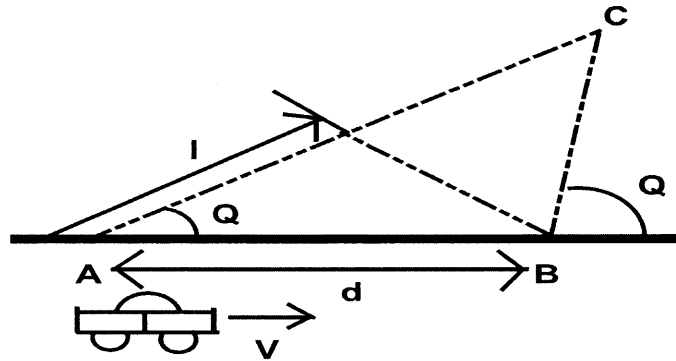
Figure 3.2 Doppler Shift.

smaller than the bandwidth of the channel, the signal undergoes *flat fading*. In flat fading, the characteristics of transmitted signal does not change when it reaches to the receiver through the channel. On the contrary, if the bandwidth of the signal is greater than the bandwidth of the channel, this type of fasing is called *Frequency Selective Fading*. In frequency selective fading, the transmitted signal can not preserve its characteristics so there exists multiple versions of that signal Small-scale fading based on Doppler spread is also grouped into two groups: If the channel impulse response varies much faster than the transmitted signal, then this channel's characteristics show *Fast Fading*. However, If the channel impulse response varies much slower than the transmitted signal, then this channel's characteristics show *Slow Fading*.

In mobile radio channels, there are two commonly used fading distributions in order to characterize the statistically time-varying nature of a fading channel [17]:

- Rayleigh Fading Channel Distribution,

- Ricean Fading Channel Distribution.

3.3   Rayleigh and Ricean Fading Channel

Rayleigh fading distribution is the most commonly used statistical model in order to characterize the statistical time varying nature of the received signal [17]. According to this

model, variation of the power of a signal shows Rayleigh distribution characteristics. It is the fact that the sum of two quadrature Gaussian shows the feature of Rayleigh fading disribution [17]. The probability density function of Rayleigh distribution is as follows:

$$p(r) = \frac{r}{\sigma^2} e^{\frac{-r^2}{2\sigma^2}}, r \geq 0 \qquad (3.8)$$

where σ is the rms value of the received voltage before envelope detection, and σ^2 is the variance of the original Gaussian signal before envelope detection [17]. The cumulative distribution function of Rayleigh signal, that is, the probability that a signal is below some threshold value, R, is [17]:

$$Pr(r \leq R) = 1 - e^{\frac{-R^2}{2\sigma^2}} \qquad (3.9)$$

The mean and the variance of a Rayleigh distributed signal are [17]:

$$E(r) = 1.2533\sigma \qquad (3.10)$$

$$\sigma_r^2 = 0.4292\sigma^2 \qquad (3.11)$$

An example of a probability density function of a Rayleigh distribution with the standard deviation "1" is given in the following figure:
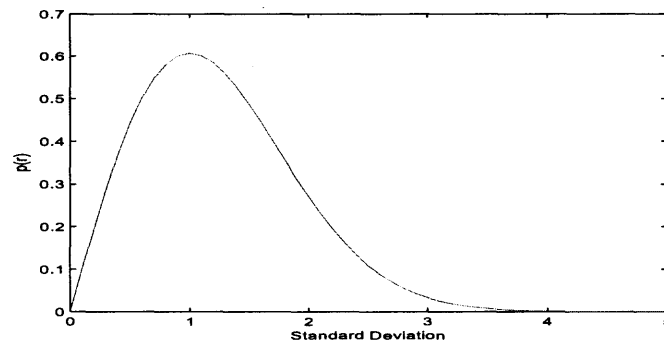
Figure 3.3 Pdf of a Rayleigh Fading Distribution with the standard deviation 1.

If there is a line-of-sight(LOS) propagation path between transmitter and receiver, then the small-scale fading envelope is said to have a Ricean distribution [17]. In Ricean distributed channel, there is a dominant signal among all the versions of the signal. As this

dominant signal component starts disappearing, Ricean channel characteristics changes to Rayleigh distribution so it can be said that Rayleigh distribution is a special case of Ricean distribution. The probability density function of Ricean distribution is as follows [17]:

$$p(r) = \frac{r}{\sigma^2} e^{-\frac{r^2+A^2}{2\sigma^2}} I_o(\frac{Ar}{\sigma^2}), A \geq 0, r \geq 0 \tag{3.12}$$

where A refers to peak amplitude of the dominant signal, and $I_o(.)$ refers to the modified Bessel function of the 1st kind and 0-order. Generally, the Ricean distribution is defined in terms of K, which is equal to $A^2/(2\sigma^2)$, and is called **Ricean Factor** [17]. When K is equal to 0, it is easy to see from the pdf equation that Ricean changes to Rayleigh [16].

3.4   Simulation of Rayleigh Fading Channel

Since it is needed to use the Rayleigh distributed fading channel in this thesis, the simulation of fading channel is implemented by using *Smith's Simulation Methodology*. Actually, this is the implementation of *Clarke and Gans Fading Model* [17]. The frequency domain implementation is shown step by step at the following figure from [17]. In order to realize this implementation, the following parameters should be specified and calculated:

- N, the number of frequency points, generally a power of two, should be specified,

- f_m, the maximum Doppler frequency shift, should be specified,

- $\triangle f$, the frequency space, which is equal to $2f_m/(N-1)$, should be calculated,

- T, time duration of a fading waweform, which is equal to $1/\triangle f$, should be calculated.
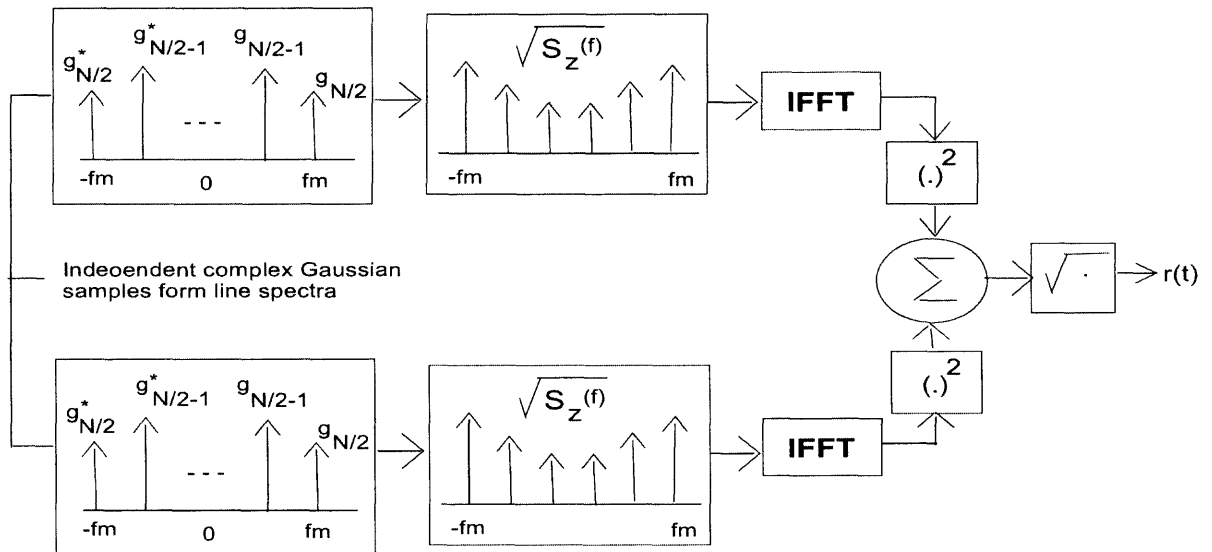
Figure 3.4 Frequency Domain Implementation of a Rayleigh Fading Simulator.

where square root of $S_{EZ}(f)$ is the fading spectrum. From the figure, the components of the Gaussian samples are symmetrical. The final signal, $r(t)$, is the simulated Rayleigh fading signal [17].

3.5   Level Crossing Rate and Average Fading Duration

The Level Crossing Rate(LCR) and Average Fading Duration(AFD) are two very important statictical parameters used for error control codes and diversity schemes [17]. Especially, you will see in the next chapter that these statistics play an important role in this thesis.

The Level Crossing Rate is the expected rate at which the normalized fading envelope crosses a specific threshold value either in a positive going way or a negative going way [17]. *Average Fading Duration* is the average duration such that the received envelope signal is below a certain threshold value [17]. The relation between these two statistics is as follows:

$$AFD(R_{Th}) = \frac{P_r[r \leq R_{Th}]}{LCR(R_{Th})} \qquad (3.13)$$

3.6 Simulation Results for Rayleigh Fading Channel

First of all, path loss is considered. To calculate the path loss, log-normal shadowing model is used, whose formula is given in equation 3.5. Same parameters are used that are defined in previous chapter. Besides, the specific path loss value at $d = 50$ m is recorded since that value is chosen as the T-R seperation distance. Next step is to simulate the fading channel, which is Rayleigh distributed. The Smith's algorithm explained in chapter 3.4 is followed for simulation. N, the number of frequency domain points, is chosen as 128, which must be a power of 2. f_m, the maximum Doppler frequency is chosen as 200 Hertz. The following graph shows the Doppler Fading Power Spectrum in terms of frequency. The formula to compute the output spectrum is given as follows:

$$S_{Ez} = \frac{1.5}{\pi f_m((1 - (\frac{f-f_c}{f_m})^2)^{1/2}} \tag{3.14}$$
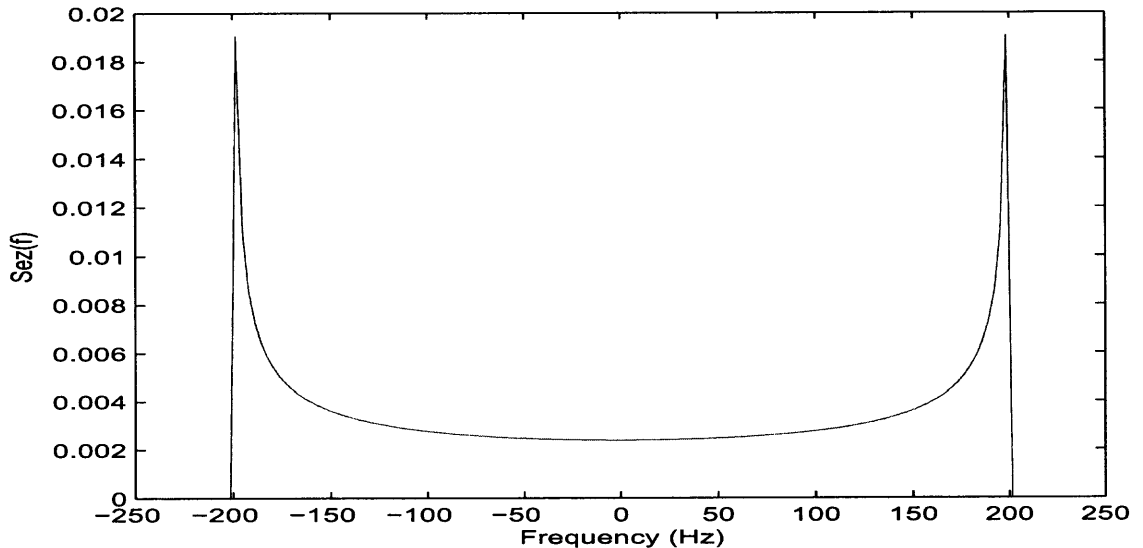
Figure 3.5 Doppler Spectrum with the maximum Doppler frequency, 200 Hz.

The following graph is the result of the Smith's algorithm, that is, the simulated Rayleigh Fading signal, $r(t)$. The total time that the signal stands for is 0.3175 seconds.

The unit of the signal in converted to decibel (dB) by firstly taking the logarithm of the signal and then multiplying by 10. The graph shows the power of the normalized fading signal.
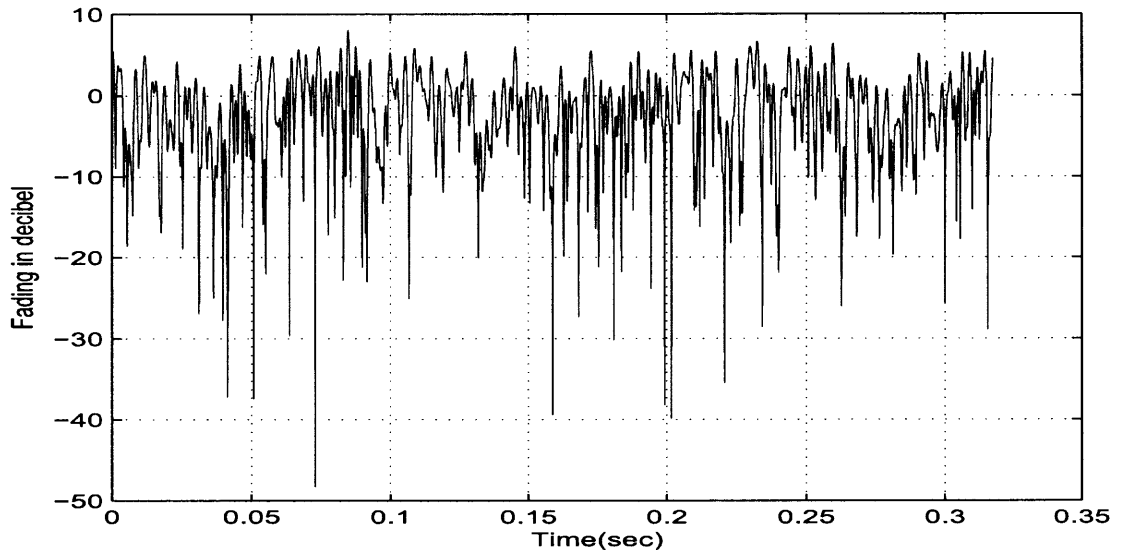
Figure 3.6 Power of the Normalized Rayleigh Fading Signal.

The previous figures seem to be a Rayleigh distributed signal. However, we have to be certain about that figure so that the specific MATLAB function is used for Rayleigh distribution. Firstly, the simulated signal (amplitude) is normalized and the histogram of this normalized signal is plotted. Then, its mean and variance is computed and these parameters are put into the MATLAB function of $raylpdf$. Now, the comparison can be performed between the probability density function (pdf) of the theoritical Rayleigh signal and the histogram of the simulated Rayleigh signal. As seen from the following figure, they match. Besides, the mean of the simulated signal is plotted as a line.
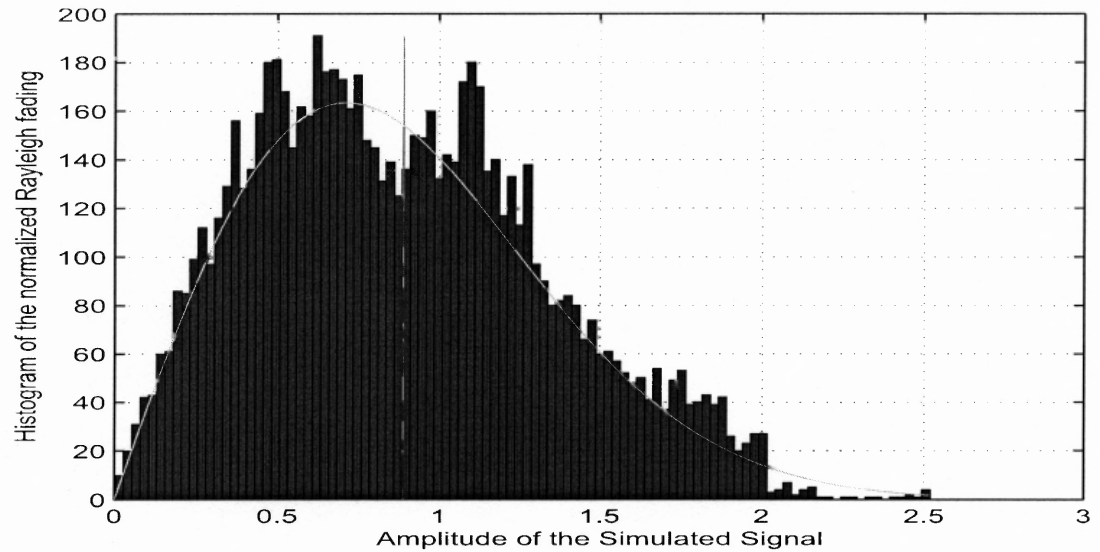
Figure 3.7 Pdf of simulated Rayleigh Fading channel with respect to its theoritical approximate within 8192 samples.

The transmitted power is chosen as 0 dBm. Since path loss and fading channel power were already computed, the received power can be calculated in dBm as follows:

$$P_{rx} = P_{tx} - \text{Fading Channel Power Loss} - \text{PathLoss};$$

(3.15)

The next graph is the plot of the received power in dBm versus time.
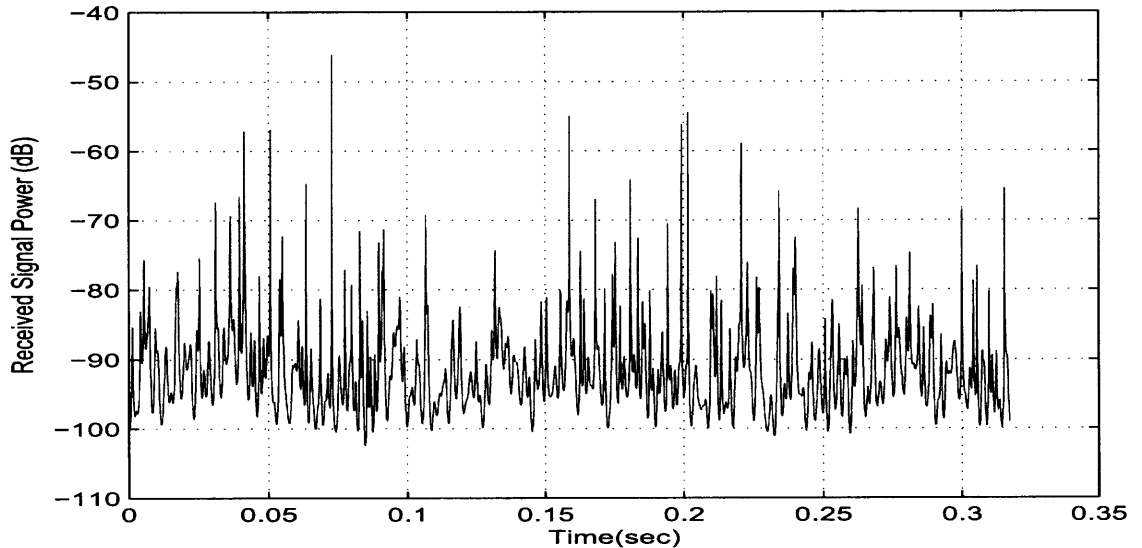
Figure 3.8 Received Power when the transmitted power is 0 dBm.

The minimum and the maximum value of the received power is recorded in order to use in the proposed algorithm.

Besides, the rate of change in the received power signal is computed for the future use. Since there are already some number of samples of the received signal, the power difference between each two time-neighbour is computed until all the samples are completed. After adding all of these to each other, the sum is divided into the number that is one less than the number of samples. This can be explained mathematically as follows:

$$\text{difference(t)} = \text{Power(t)} - \text{Power(t-1)}, \text{ for } 2 \leq t \leq \text{samples} \qquad (3.16)$$

$$\text{summation} = \Sigma\text{difference(t)}, \text{ for } 2 \leq t \leq \text{samples} \qquad (3.17)$$

$$\text{rate of change}_{\text{average}} = \text{summation/samples} - 1 \qquad (3.18)$$

Next, the parameters, Level Crossing Rate (LCR) and Average Fading Duration (AFD), are calculated. Since the minimum and maximum value of the received power signal is already recorded, the parameter values for each power level between the minimum and

maximum points can be easily computed. For Level Crossing Rate, the positive-going level crossings are computed. For a given threshold power level value, if power level at time t is less than this threshold, and power level at time t+1 is above threshold, the number of Level Crossing for that threshold value increases by 1. If power level of signal is represented as *P*, and power level of threshold is represented as *PTh*, this relation can be written mathematically as follows:

$$\text{if } (P(t) < PTh) \text{ and } (P(t+1) \geq PTh), \tag{3.19}$$

then the number of Level Crossing for that threshold increases by 1.

The following figure shows the Level Crossing Rate versus power level (threshold).
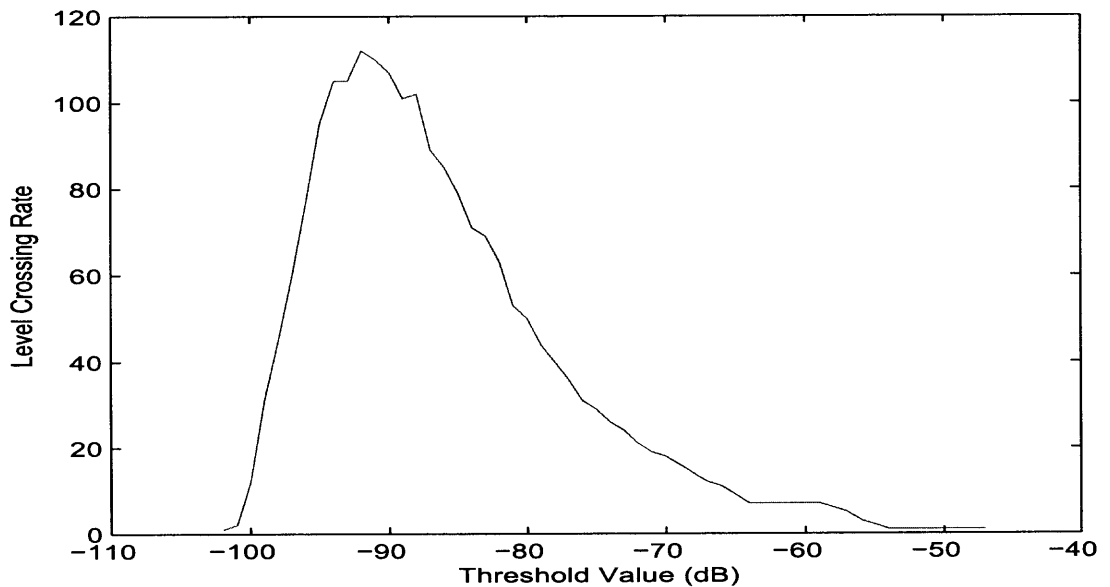
Figure 3.9 Level Crossing Rate of the Received signal shown at the Figure 4.4.

The Average Fading Duration (AFD) is computed from the given equation 3.13. According to the formula, the cumulative distribution functions should be calculated in addition to the LCR in order to compute AFD. This paramater is used to determine the probing period when the transmitter enters probing mode so that it plays an important role in this thesis.
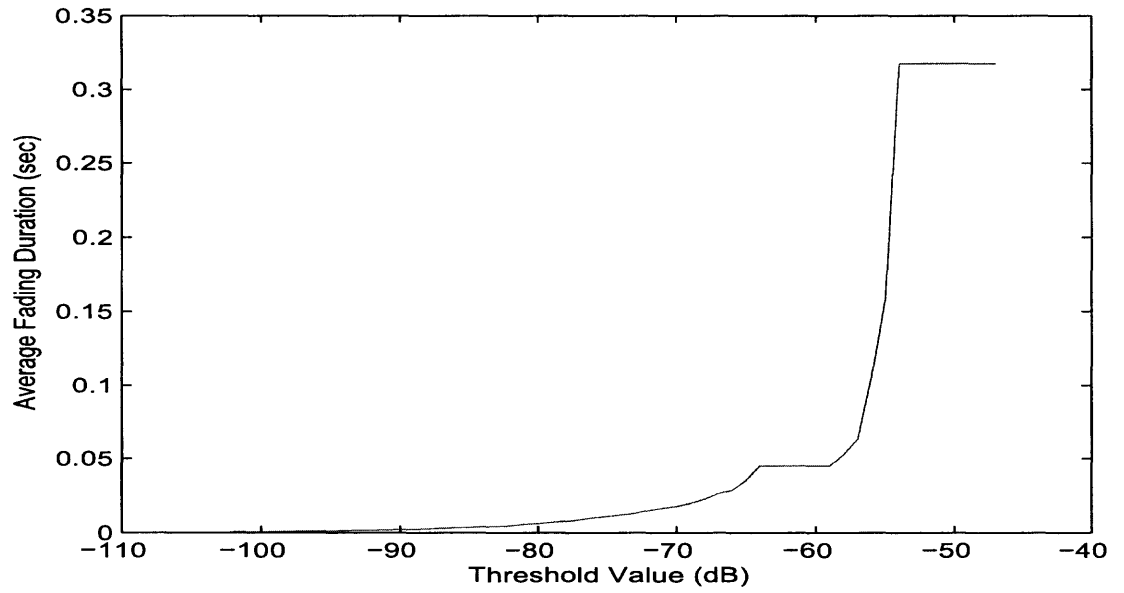
Figure 3.10 Average Fading Duration of the Received signal shown at the Figure 4.4.

CHAPTER 4

PERFORMANCE OF CHANNEL AWARE STOP-and-WAIT ARQ PROTOCOL

This chapter presents the newly proposed channel-aware ARQ protocol variants. Relative performance of the proposed approach has also been compared with respect to the basic S-W ARQ and the most relevant other channel-predictive ARQ protocol, namely probing based approach [21]. Before going into proposed channel-aware protocols, performance of the prior state of the art is analyzed.

4.1 Performance Analysis of Prior ARQ Protocols

In this thesis, Stop-and-Wait ARQ protocol is used to evaluate the channel awareness performance. As in [21], throughput and energy efficiency are considered as the performance yardsticks of a different S-W ARQ variants. In addition, delay efficiency is also considered. Before going into the channel-aware ARQ approaches that are proposed in the thesis, the throughput, energy efficiency and delay efficiency performances of the basic S-W and the probing based approach [21] are analyzed.

Traditional Stop-and-Wait ARQ protocol: Since there is no channel-checking mechanism, the transmission is always done. If the power level during any instant of RTT period is below the threshold value, then after time-out period is over, the transmission continues. Besides, the number of transmitted packets increases by 1. However, if the power level during RTT period is above the threshold value, then both the number of successfully transmitted packets and the number transmitted packets increase by 1. This process continues until the end of the fading waveform is reached. Mathematically, if the number of successfullly transmitted packets is shown as S and the power at that instant is represented as P, this process can be expressed as:

$$S = 1, \text{ if } P \geq \text{Threshold during every instant of RTT} \tag{4.20}$$

$$S = 0, \text{ if } P < \text{Threshold during any instant of RTT}$$

This whole process repeats for each fading margin value until the fading margin reaches to its maximum value. The parameter, Energy-Efficiency, is evaluated in terms of the number of transmissions per successfully transmitted packet, and they are inversely proportional with the throughput. The parameter, Delay Efficiency, is evaluated in terms of the number of successfully transmissions during a given run time. If the number of data packet transmissions is represented as T, and the number of successfully transmitted data packets is shown as S, then the throughput, delay-efficiency, and energy-efficiency are computed as follows:

$$\text{Throughput} = S \,/\, T \tag{4.21}$$

$$\text{Energy Efficiency} = T \,/\, S$$

$$\text{Delay Efficiency} = S \text{ during a given run time}$$

Probing based Stop-and-Wait ARQ Protocol [21]: In this scheme, there is a control mechanism such that if a NAK, negative acknowledgement, is sent to the transmitter, then transmission of data packet is suspended. NAK feedback is sent when the power level at that time instant is lower than the threshold value. Once this scenario is realized, the transmitter changes its mode from normal mode to probing mode. During probing mode, the probing packets are sent periodically, whose payload is much less than a data packet. Again, during the probing mode, if ACK, positive acknowledgement, is taken, the transmitter continues to send data packets. When the parameter, efficiency, is computed, the probing packets are counted as much as their size, that is, 10 probing packets refer to one data packet [21]. The probing period is chosen as 2 time slots. In addition to the calculation of throughput and efficiency of traditional S-W ARQ, let's refer the number of probing packets as P, and the proportion of the size of probing packet to data packet as W so the equations for throughput, delay-efficiency, and energy-efficiency are as follows:

$$\text{Throughout} = S \,/\, (T + P) \tag{4.22}$$

$$\text{Efficiency} = (T \,/\, S) + (W * P \,/\, S)$$

Delay Efficiency = S during a given run time

If the probing period increases, then it is observed that both throughput and efficiency increase, and delay efficiency decreases.

4.2 Proposed Channel-aware ARQ Protocol

The Round-Trip time (RTT) for data packets is $2 \times (d/v) + (L/R)$, and for probing packets, it is $2 \times (d/v) + (L_p/R)$. The time-out duration is chosen as $RTT + 1e - 005$ sec. From the simulated channel, it is found that the time parameters, RTT for data packets, RTT for probing packets, and time-out refers to how many time-slots. Receiver threshold(denoted as P_{Th}) is chosen as P_{Th}(min) plus some noise margin. P_{Th}(min) is obtained from the channel simulation data, and noise margin is considered to be 20 dB. And then, the analysis in terms of throughput, delay-efficiency and energy-efficiency can be performed. These performance parameters are computed as in equation 4.22.

After the environment has been set for the transfer of data, the analysis of the proposed algorithm can be performed. Firstly, the parameters are defined and assigned to the values. d, T-X seperation distance, is chosen as 50 m. c, the speed of light, is 3×10^8 m/sec. v, the speed of the signals, is $2 \times c/3$. So the propagation delay for one direction is d / v. R, the channel rate, is chosen as 11 Mbps ,which is generally the case in wireless environment. L, the data packet length, is chosen as 100 bytes, and L_p, probing packet length is chosen as 10 bytes.

Now, our proposal is discussed. First of all, it should be mentioned that from the simulated Rayleigh Fading channel, rate of change is known. Besides, AFD value for the predefined threshold value is redefined in terms of time slots. After these initiations, the main algorithm can be described. This algorithm includes four modifications over the proposed algorithm by [21]:

Algorithm 1: *Probing based on only receiver Threshold dependent AFD* − If the power

level of received signal at the receiver is lower than the threshold value, the transmitter enters to probing mode. Then, it immediately sends the probing packet, but then the question here is how frequent the transmitter should probe the channel. According to the last proposed opinion, the probing period is arbitrarily chosen. Instead of choosing a parameter that does not rely on any information, the probing interval is determined according to the Average Fading Duration (AFD) of threshold signal. That is, the transmitter waits along AFD of threshold signal.

The throughput, energy-efficiency and delay-efficiency graphs of first algorithm compared with the two previously proposed algorithms are given as follows:

Figure 4.1 Throughput vs. Fading Margin (Transmitted Power) when probing period is determined according to AFD value of the threshold value, if the received signal is below the threshold value
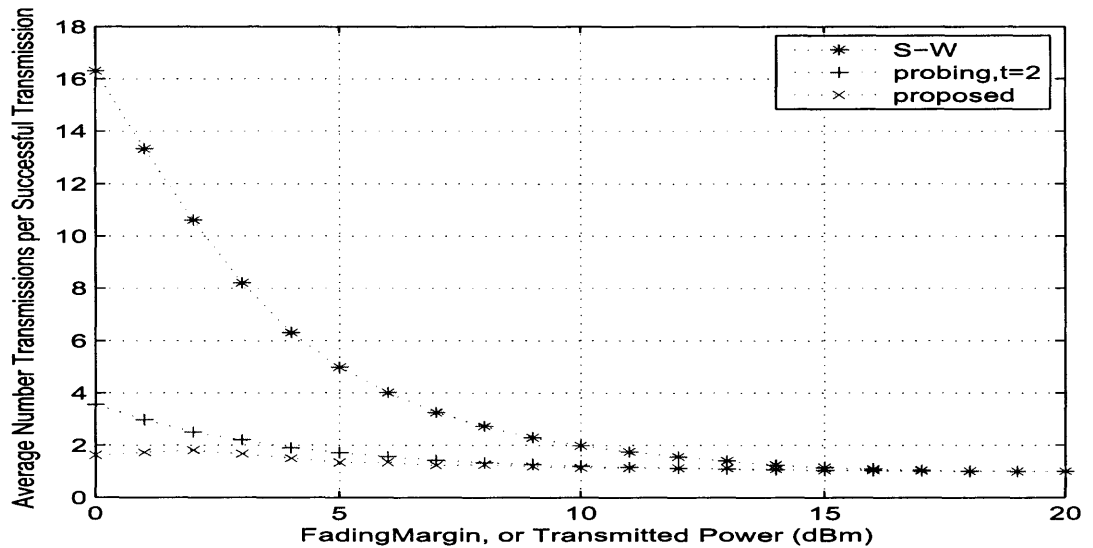
Figure 4.2 Energy Analysis when probing period is determined according to AFD value of the threshold value, if the received signal is below the threshold value

Figure 4.3 Delay-Efficiency Analysis when probing period is determined according to AFD value of the threshold value, if the received signal is below the threshold value

Algorithm 2: *Probing based on AFD that is a function of a current signal level* – If you look at the following figure, point A refers to the received signal level. Assume that AFD of the received signal is represented as t_1 and AFD of the threshold value is represented as

t_2. The AFD value in terms of time slots are computed for the received signal level. If the signal level is at point "A", then the transmitter waits for:

$$Time = t_1 + (t_2 - t_1)/2 \tag{4.23}$$
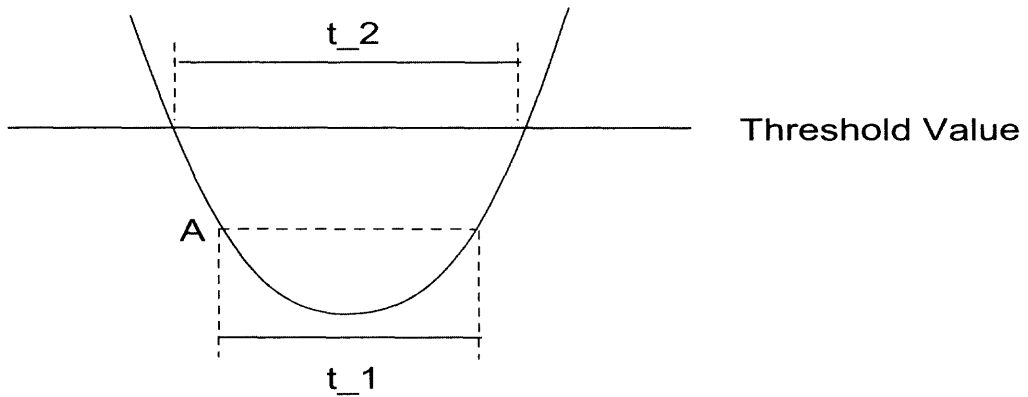
Figure 4.4 Illustration of how to use AFD for the received signals whose power levels are below the threshold.

The throughput, energy-efficiency and delay-efficiency graphs of second algorithm compared with the two previously proposed algorithms are given as follows:
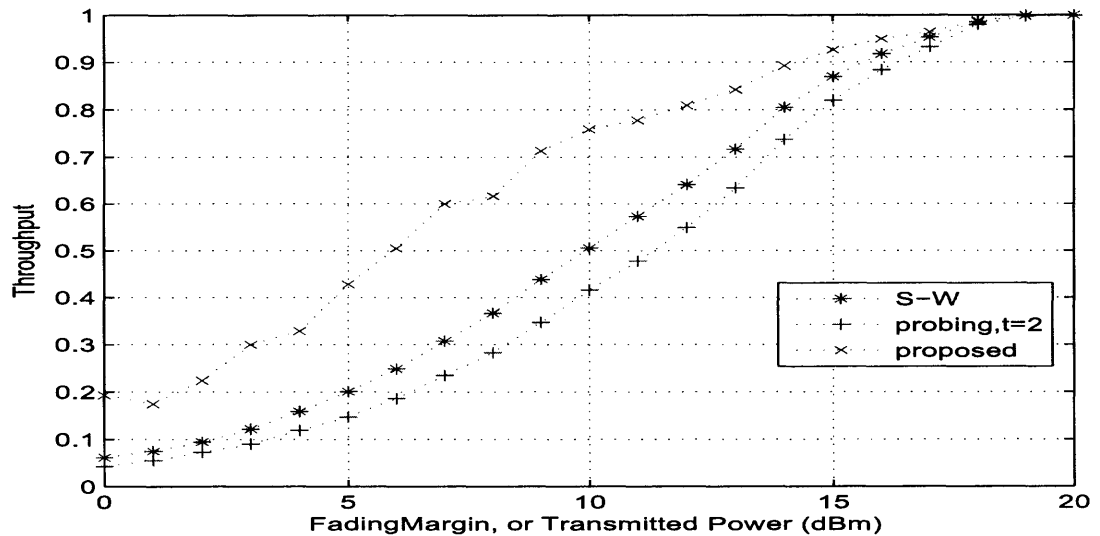

Figure 4.5 Throughput vs. Fading Margin (Transmitted Power) when probing period is determined according to AFD value of the received signal value and threshold value, if the received signal is below the threshold value

Algorithm 3: *Probing based on AFD that is a function of current signal level and its slope of change–* If you look at the figure 4.8, there are two points, A and B, which have the same signal levels. These points refer to the signal level below the threshold value. The transmitter should not wait for the same long of time for these two points. What recommended is to check the close history of the channel, that is, when the signal values at the last two time instants are checked, it can be figured out whether it has been keeping decreasing or increasing. If it is observed that the signal has been decreasing, so the point should be point "A", otherwise "B". The time durations that the transmitter should wait for sending the next probing packet are given as follows:

Again, assume that AFD of the received signal is represented as t_1 and AFD of the threshold value is represented as t_2. If the signal level is at point "A", then the transmitter

Figure 4.6 Energy Analysis when probing period is determined according to AFD value of the received signal value and threshold value, if the received signal is below the threshold value
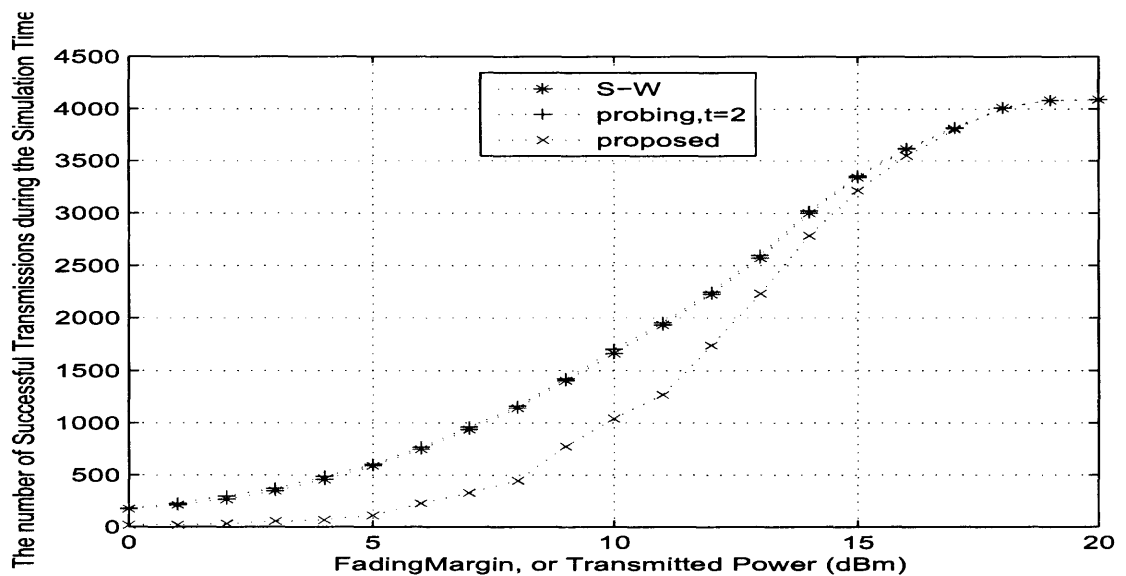

Figure 4.7 Delay-efficiency Analysis when probing period is determined according to AFD value of the received signal value and threshold value, if the received signal is below the threshold value

waits for:

$$Time = t_1 + (t_2 - t_1)/2 \qquad (4.24)$$

If the signal level is at point "B", then the transmitter waits for:

$$Time = (t_2 - t_1)/2 \qquad (4.25)$$
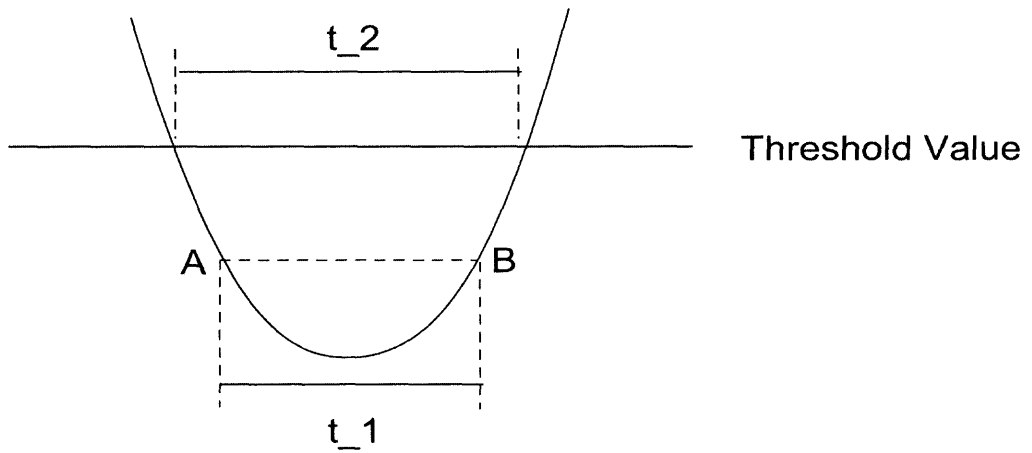
Figure 4.8 Increasing or decreasing.

The throughput, energy-efficiency and delay-efficiency graphs of third algorithm compared with the two previously proposed algorithms are given as follows:
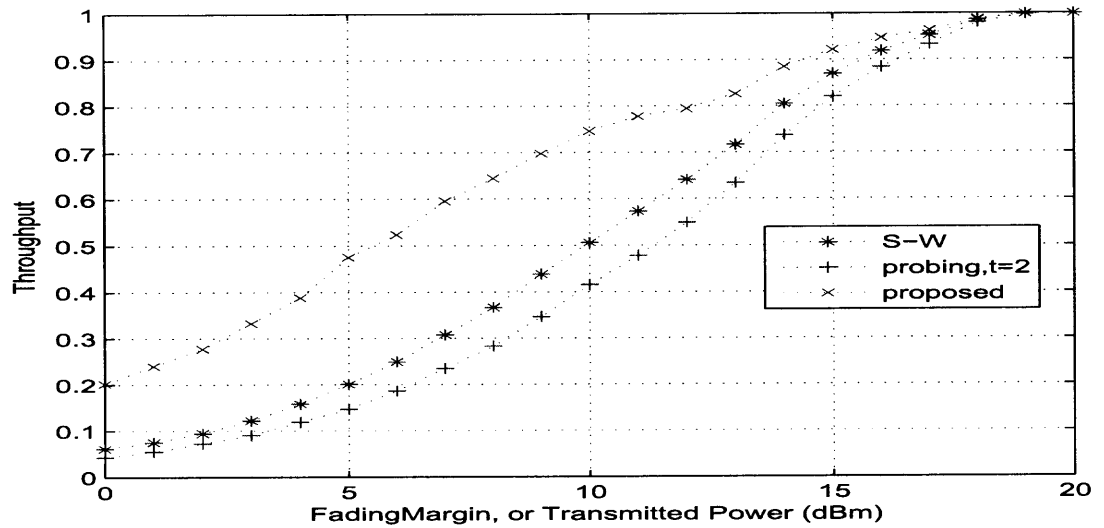
Figure 4.9 Throughput vs. Fading Margin (Transmitted Power) when probing period is determined according to AFD value of the received signal value and threshold value, and the history of the channel, if the recieved signal is below the threshold value
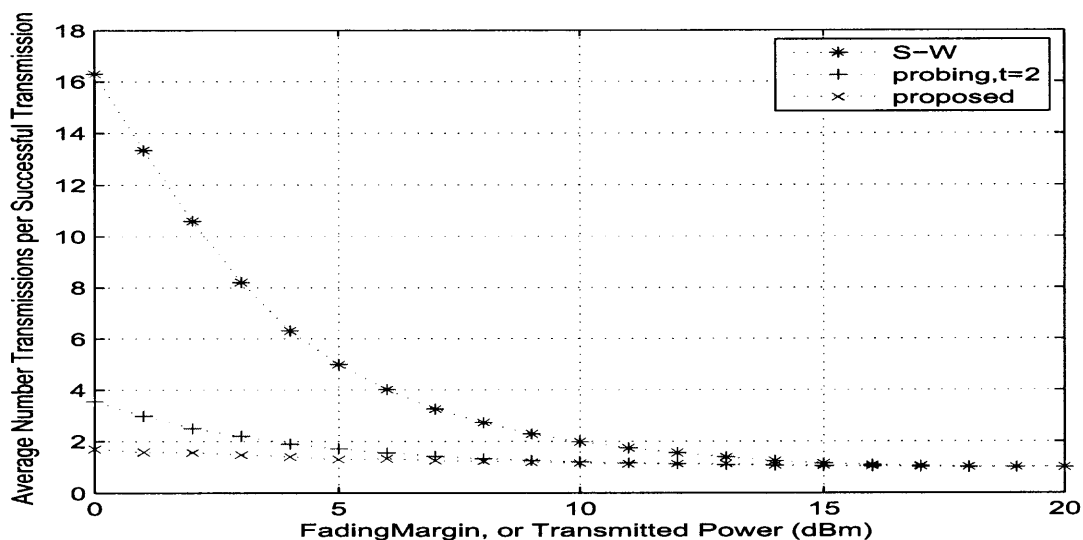
Figure 4.10 Energy Analysis when probing period is determined according to AFD value of the received signal value and threshold value, and the history of the channel, if the received signal is below the threshold value
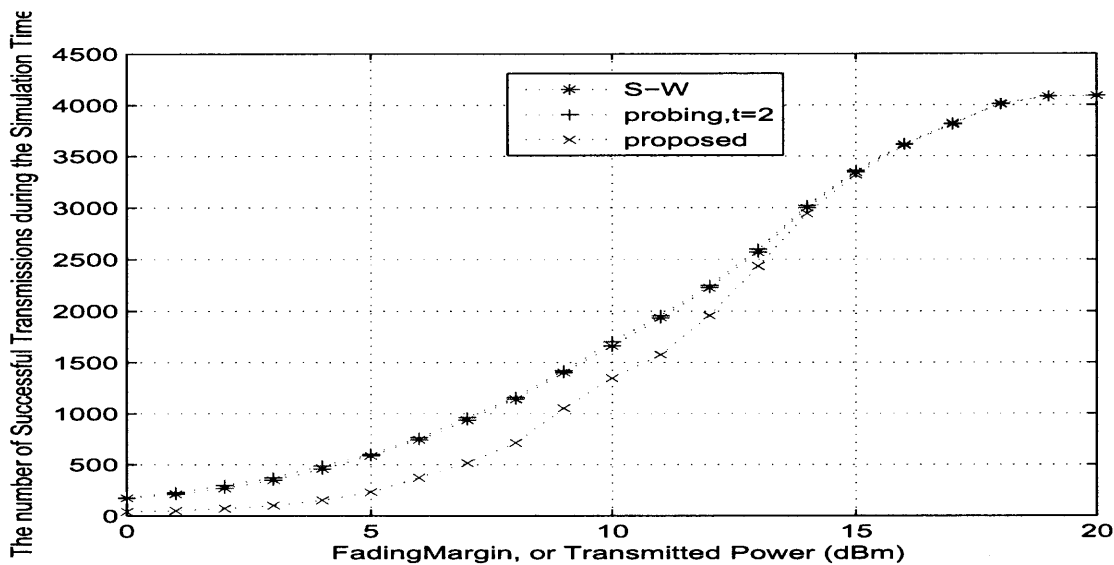
Figure 4.11 Delay-efficiency Analysis when probing period is determined according to AFD value of the received signal value and threshold value, and the history of the channel, if the received signal is below the threshold value

Algorithm 4: *Probing based on AFD that is a function of current signal level(which could be at or even above the threshold), and its slope of change* – Assume that the power level of the received signal is equal to the receiver threshold value, and it is observed from the history of the channel that the signal values have been decreasing. This observation shows us that the signal level will go down, so the transmitter should change its mode to probing mode. Besides, if the power level is greater than the receiver threshold value, since the rate of change in simulated channel and the RTT is already known parameters, the power levels of fading channel can be closely approximated during the communication period for the worst condition. If it is expected that the power level will drop under the threshold level at any instant during RTT, the transmitter enters probing mode.It will wait for the duration of equation 4.24.

The throughput, energy-efficiency and delay-efficiency graphs of last algorithm compared with the two previously proposed algorithms are given as follows:
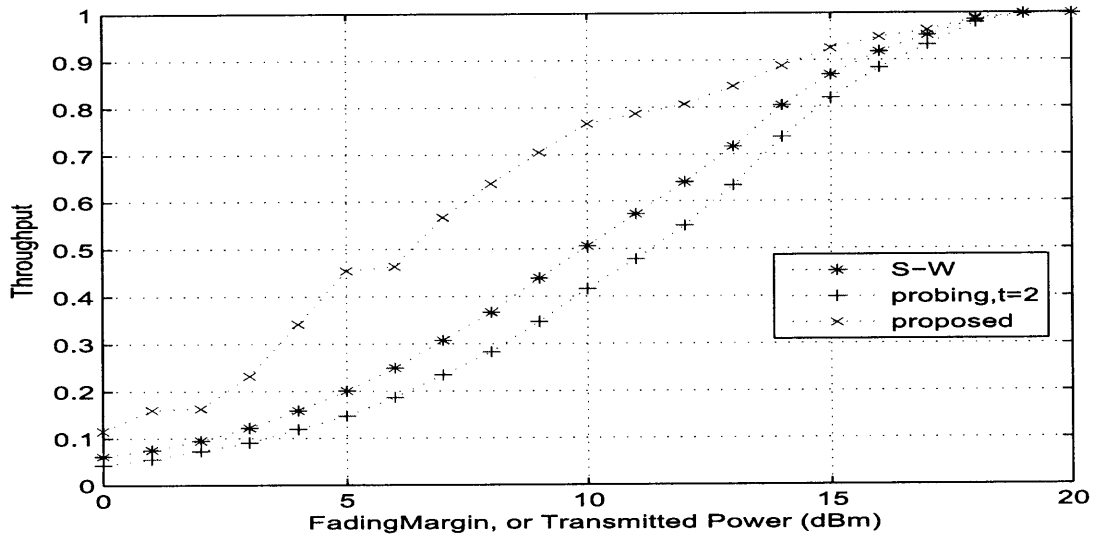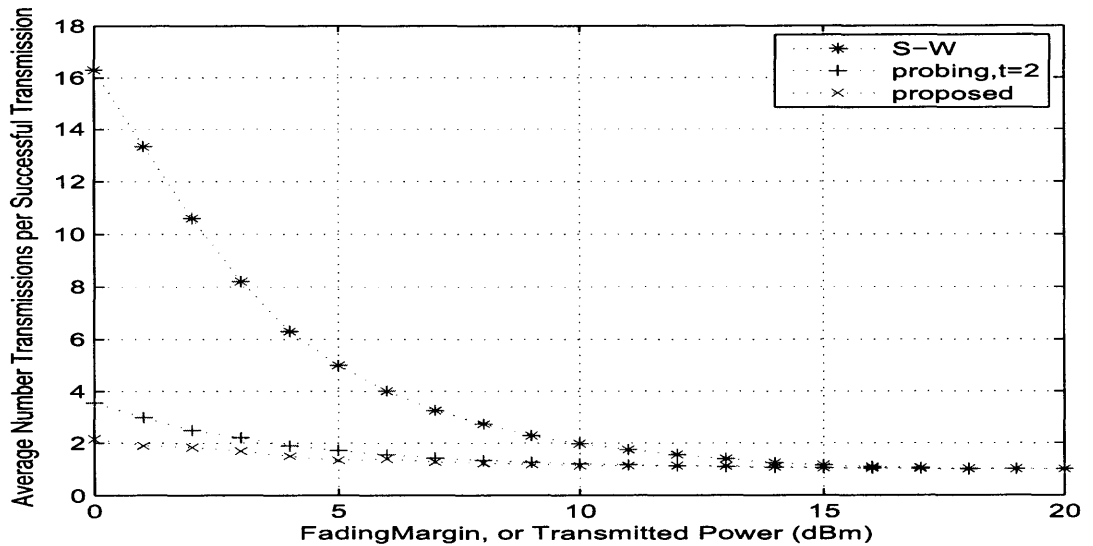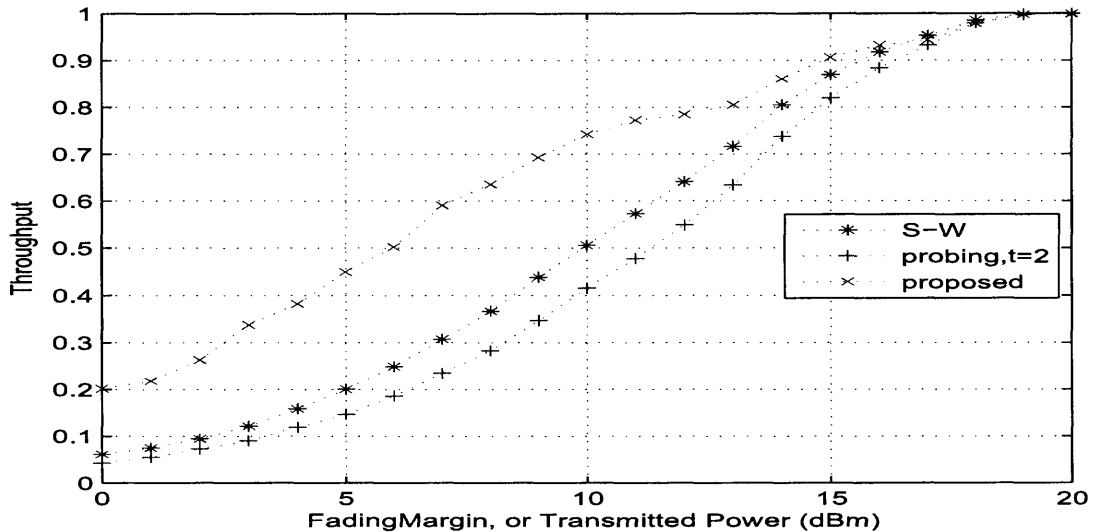

Figure 4.12 Throughput vs. Fading Margin (Transmitted Power) when probing period is determined according to AFD value of threshold value, if the received signal is above threshold value, but for the worst condition, it is likely to go below threshold value by using rate of channel
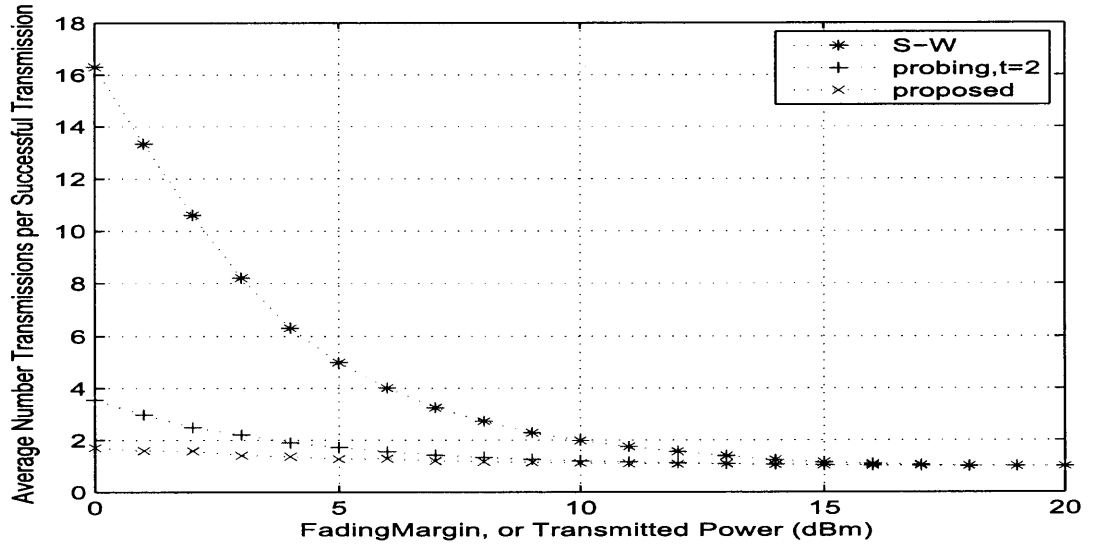
Figure 4.13 Energy Analysis when probing period is determined according to AFD value of threshold value, if the received signal is above threshold value, but for the worst condition, it is likely to go below threshold value by using rate of channel
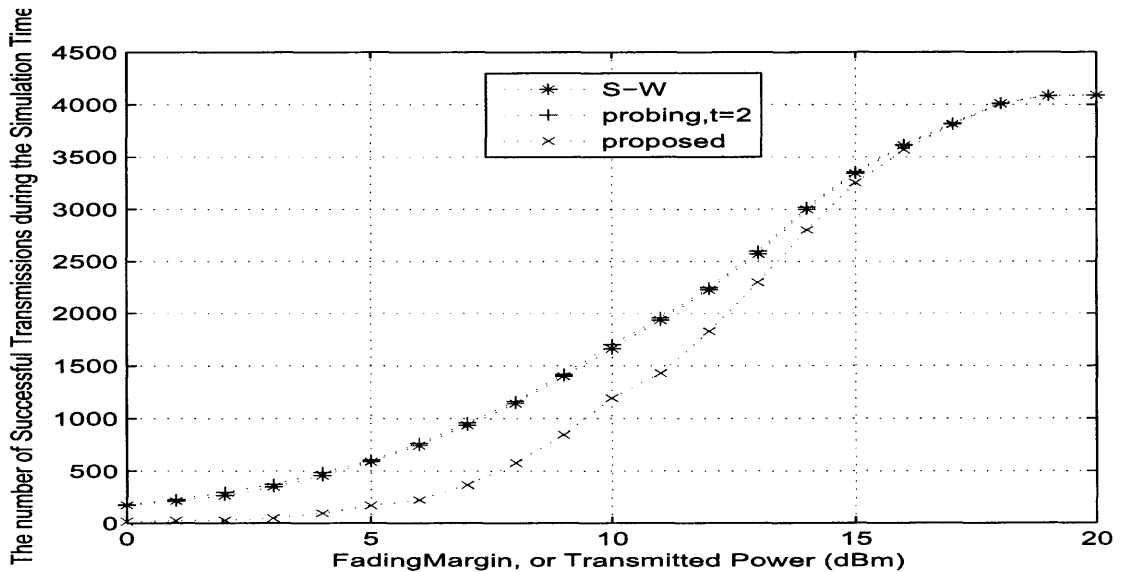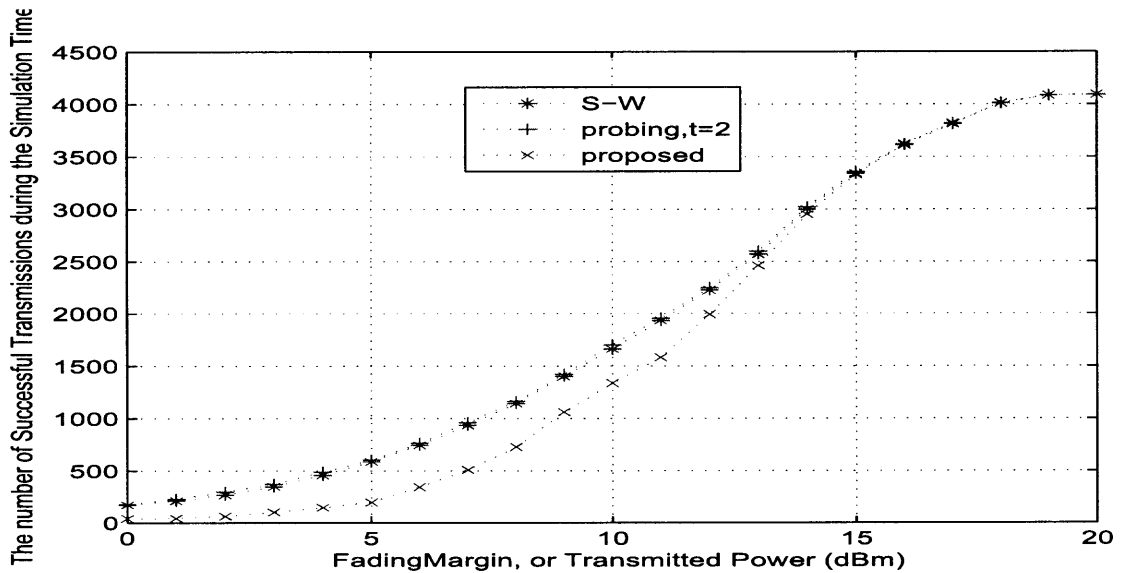

Figure 4.14 Delay-efficiency Analysis when probing period is determined according to AFD value of threshold value, if the received signal is above threshold value, but for the worst condition, it is likely to go below threshold value by using rate of channel

From the graphs, values for throughput, delay-efficiency (number of successful trans-missions during the fixed simulation period) and enery-efficiency (average number trans-missions per successfully received packet) for each step at fading margin=4 can be put into a table in order to see the algorithms:

Table 4.1 Throughput and Energy Efficiency Comparison for Different Algorithms at Fading Margin = 4dBm and 8dBm

|	S-W ARQ	Probing Method [21]	Algorithm 1	Algorithm 2	Algorithm 3	Algorithm 4
At 4dBm Throughput	0.1585	0.1189	0.3283	0.3418	0.3882	0.3822
At 4dBm Energy Efficiency	6.3085	1.8982	1.4954	1.4989	1.4079	1.3712
At 8dBm Throughput	0.3668	0.2832	0.6166	0.6382	0.6447	0.6357
At 8dBm Energy Efficiency	2.7266	1.3361	1.2575	1.2336	1.2305	1.1738

It is observed that the proposed algorithms works well in terms of higher throughput and consuming less energy. However, at fading margin = 4dBm, first algorithm seems to be slightly more energy-efficient than the the second algorithm. But, if you look at the next table, which shows the number of successful transmissions during the simulation period, it is observed that approximately, the twice number of successful tranmissions are done at the second algorithm compared to the first algorithm.

Table 4.2 Delay Efficiency Comparison for Different Algorithms at Fading Margin = 4dBm and 8dBm

	S-W ARQ	Probing Method [21]	Algorithm 1	Algorithm 2	Algorithm 3	Algorithm 4
At 4dBm Delay-Efficiency	457	487	65	94	151	146
At 8dBm Delay-Efficiency	1141	1160	447	575	713	726

Besides, according to the table, it is observed that the number of successful tranmissions at third algorithm is slightly more than the fourth algorithm. However, there is a reasonable decrease in the energy consumption at the fourth algorithm, Table 4.1.

Moreover, power consumption of this wireless system can be analyzed. For this analysis, the data of Cisco Aironet 350 Client Adapter, which is a 802.11b card, is used [20]. The typical power consumption at 100 mW(20 dBm) transmitter power by the transmitter

is 450 mA, and by the receiver is 270 mA [1]. The energy consumption graph is given as follows [20]:
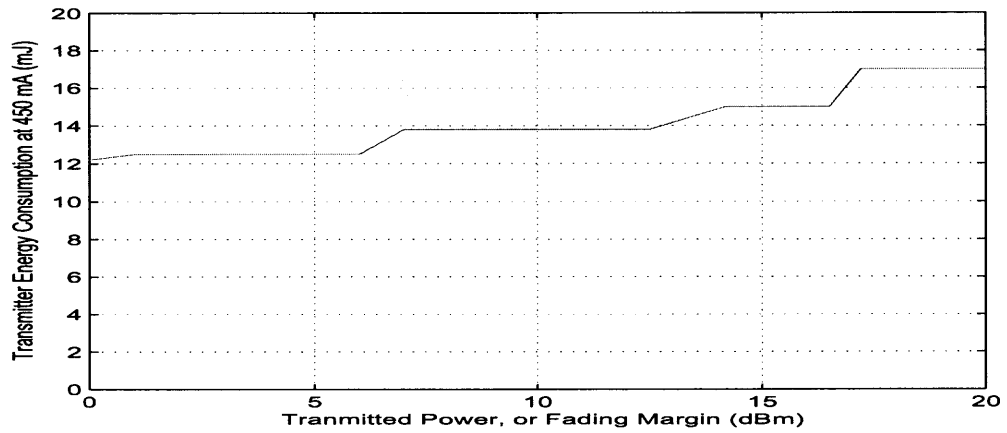
Figure 4.15 Energy Consumption at different transmit power levels

Table 4.3 Transmit Power and the corresponding value for the average number of transmissions per successful transmission

Fading Margin (Transmit Power)	Avg. number of transmissions per successful transmission
0	1.6949
1	1.5884
2	1.5750
3	1.4099
4	1.3712
5	1.2740
6	1.2854
7	1.2024
8	1.1738
9	1.1464
10	1.1183
11	1.0954
12	1.0906
13	1.0878
14	1.0595
15	1.0390
16	1.0259
17	1.0237
18	1.0087
19	1.0011
20	1.0006

By using the table above and the figure 4.15, the total power consumption can be computed. The approximate energy values from the figure 4.15 are found, and the corresponding power values can be computed easily as follows:

$$Power(watts) = Energy(joule)/Transmission\ Time\ (sec) \qquad (4.26)$$

$$Power(dBm) = 10log\ (Power\ (watts))$$

where transmission time is packet (data+probing) length / transmission rate.

The following graph shows the total power consumption of the wireless system. Since the proposed algorithms provide better performance at all cases, only the best algorithm(4th algorithm) is analysed in terms of power consumption.

Although with increased tranmission energy consumption per packet increases, the
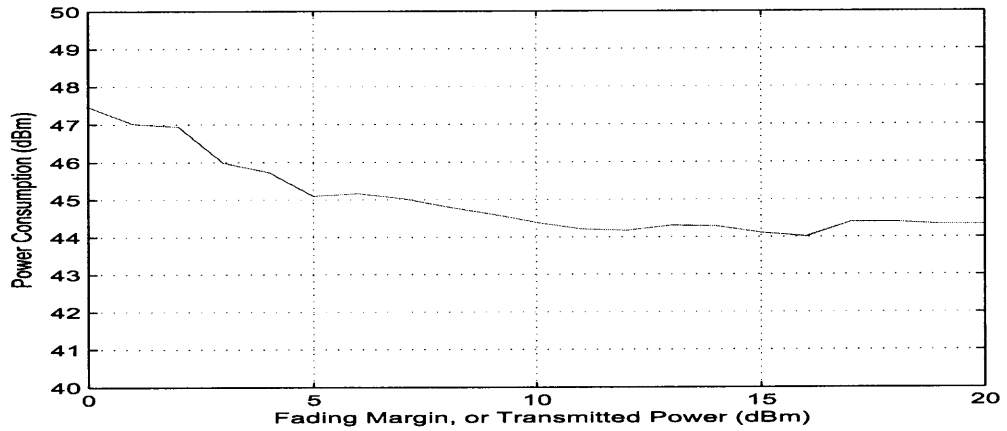
Figure 4.16 Power Consumption at different transmit power levels

average power consumption per successfully transmitted packet decreases due to the de-crease in the average number of tranmissions per successfully tranmission Table 4.3.

Also, it may be noted that there is an optimum value of transmit power, or fading margin (16 dBm), at which the average power consumption is minimum (44 dBm).

CHAPTER 5

CONCLUSION and FUTURE WORK

Energy efficient networking performance is a major issue in wireless communications. Traditional link layer ARQ protocols do not use physical layer intellegence in retransmission strategies, and therefore these blind ARQ approaches may not guarantee energy efficient performance. In this thesis, the four algorithms have been proposed that provides the channel state estimation in order to determine a frame transmission/retransmission timing instant. Then, these proposed algorithms have been compared with traditional S-W ARQ protocol, and the predictive S-W ARQ protocol proposed in [21]. The performance analysis of these algorithms has been evaluated in terms of three paramaters, which are throughput, energy-efficiency, and delay-efficiency. In terms of energy efficiency, it has been observed that the proposed algorithms operated as expected. That is, they provide more energy efficient solutions. However, in terms of delay efficiency, it has been observed that there was a decrease at the number of successful packets. This shows the trade-off between two efficiencies. Besides, the delay-efficiency performance makes the analysis more comfortable in terms of reliability of the analysis.

The proposed algorithms are generic, and can be applied to other ARQ protocols, which are Go-Back-N and Selective-Repeat ARQ protocols step by step and see whether the results on these more developed ARQ protocols works as well.

The MATLAB Code

This appendix includes the code of Path Loss, Rayleigh Fading Simulator, and the main algorithm in MATLAB.

Path Loss

```
clear all;
close all;
%Path Loss according to the log-normal shadowing
%firstly, let me define the variables and assign values to them
%d0 : close-in(reference) distance. Typically chosen 1 m in indoor
%environments and 100 m or 1 km in outdoor environments
d0 = 1; % in m
%d : T-R distance seperation
%n = path loss exponent
%w = wavelength
%f = carrier frequency
%c = speed of light
c = 3*(10^8);
f = 2.4*(10^9);
w = c/f;
%Pt : Transmitter power in dBm
Pt = 0; %in dBm
%X : zero-mean Gaussian distributed RV in dB (variation as 1)
X1 = real(normrnd(0,1));
X = real(10*log10(X1));

% PLd0 : Path Loss at distance d0 in dB
%Assumption is unity gains at antennas
PLd0 = -10*log10((w^2)/(((4*pi)^2)*d0));

n=3;

d=0:1:100;
    %PL : Path Loss at distance d
    PL = PLd0 + 10*n*log10(d/d0) - X;
    plot(d,PL)
    grid on;
    xlabel('distance(m)');
    ylabel('PathLoss in dB');

    PL(50)
```

Rayleigh Fading Simulator

```
clear all;
close all;
PathLoss;
%Rayleigh fading simulator according to Smith's algorithm
%Steps are written explicitly
%STEP1
%Assign N : The number of frequency domain points (usually power of 2)
N = 128;
%Assign fm : The max. Doppler frequency shift
fm = 200; %in Hertz
%STEP2
%Af : Frequency spacing between adjacent spectral lines
Af = 2*fm/(N-1);
Af
%T = time duration of a fading waveform
T = 1/Af;
T % in sec
%STEP3&STEP4
%Generate complex Gaussian RV's for each of the N/2 positive frequency
%components of the noise source
%two-dimensional Arrays are created so that we can follow the signal
%Besides, negative frequency components are assigned
%1st Gaussian
g1 = linspace(-(N*Af/2),(N*Af/2), N+1);
for f=(N/2)+1:N
    g1(f)=g1(f+1);
end
g1=g1(1:N);

for j=(N/2)+1:N
    a=randn(1);
    b=randn(1);
    g1(2,j) = a + i*b;
    g1(2,N-j+1) = g1(2,j);
end
%2nd Gaussian
g2 = linspace(-(N*Af/2), (N*Af/2), N+1);
for f=(N/2)+1:N
    g2(f)=g2(f+1);
end
g2=g2(1:N);

for l=(N/2)+1:N
    c=randn(1);
```

```
    d=randn(1);
    g2(2,l) = c + i*d;
    g2(2,N-l+1) = g2(2,l);
end
%STEP5
%Firstly, I will create Doppler fading spectrum
S = linspace(-(N*Af/2), (N*Af/2), N+1);
for f=(N/2)+1:N
    S(f)=S(f+1);
end
S=S(1:N);

for f=65:128
    S(2,f) = 1.5/(pi*fm*(1-(S(1,f)/fm)^2)^0.5);
    S(2,N-f+1) = S(2,f);
end

figure;
j=1:N;
plot(S(1,j),S(2,j))

% for j=1:N
    S(2,:) = sqrt(S(2,:));
% end
%Secondly, multiplication of fading spectrum and noises
Tc1(1,:) = S(1,:);
Tc2(1,:) = S(1,:);
Tc1(2,:) = S(2,:).* g1(2,:);
Tc2(2,:) = S(2,:).* g2(2,:);
%STEP6
%Firstly, Perform IFFT
    samples=8192;
    At = T/samples;
    %t=0:At:T-At;
    t=(0:samples-1)*At;
    Tc1t = ifft(Tc1(2,:),samples,'symmetric');
    Tc2t = ifft(Tc2(2,:),samples,'symmetric');

    sum = sqrt((Tc1t.^2) + (Tc2t.^2));
    sumdB = 10*log10(sum);
    rms=sqrt(mean(sum.^2));

    figure;
    plot(t,sumdB)
    grid on
    title('Rayleigh');
```

```
xlabel('Time(sec)')
ylabel('Fading in decibel')

figure;
hold off
hist(sum/rms,100)
grid on
title('Histogram of the normalized Rayleigh fading')

mn = mean(sum/rms)
hold on;
plot([mn mn],[0 max(hist(sum/rms,100))],'-r')


r=0:0.001:max(sum/rms);
sigma = mn/(sqrt(pi/2));
hold on;
plot(r,max(hist(sum/rms,100))*raylpdf(r,sigma),'-g')

figure;
plot(t,10*log10((sum/rms).^2))
grid on
title('Rayleigh Power');
xlabel('Time(sec)')
ylabel('Fading in decibel')

%Pr: received power
Pr=Pt-10*log10((sum/rms).^2)-PL(50);

figure;
plot(t,Pr)
grid on
title('Received Power in dB');
xlabel('Time(sec)')
ylabel('Fading in decibel')

figure;
hold off
hist(Pr,100)
grid on
title('Histogram of the Power')

['Maximum value of the Received Power is: ', num2str(max(Pr))]
['Minimum value of the Received Power is: ', num2str(min(Pr))]

%    sqrt(var(sum/rms)*2/(4-pi))
```

```
%Now, let's compute the rate of change of the fading
toplam = 0;
for i=2:samples
    diff(i-1) = abs(Pr(i) - Pr(i-1));
    %rate(i-1) = diff(i-1)/(At); % '1000' in order to see rate of change per msec
    toplam = toplam + diff(i-1);
end

    ['Average Channel variation rate in dB per one time-slot is : ',
num2str(toplam/(samples-1))]
    %Now, let's compute the Level Crossing Rate (LCR) and Average Fading
    %duration (AFD) respectively for some values of threshold
    %based upon positive-going level crosssing
    k=1;
    for threshold=ceil(min(Pr)):floor(max(Pr))
        LCR(k)=0;
        cum = 0;
        cum1(k) = 0;
        for i=1:samples
            if(Pr(i)<=threshold)
                cum = cum +1;
            end
        end
        cum1(k) = cum/samples;
        for j=2:samples
            if((Pr(j)>=threshold)&&(Pr(j-1)<threshold))
                LCR(k) = LCR(k)+1;
            end
            continue
        end

        AFD(k)= cum1(k)/(LCR(k)/T);
        k=k+1;
    end

    figure;
    plot(ceil(min(Pr)):floor(max(Pr)),AFD)
    title('Average Fading Duration vs. Threshold');
    xlabel('Threshold')
    ylabel('AFD')

    figure;
    plot(ceil(min(Pr)):floor(max(Pr)),LCR)
    title('Level Crossing Rate vs. Threshold');
    xlabel('Threshold')
    ylabel('LCR')
```

Proposed Algorithms

```
%clear all;
%close all;
%RayleighFadingSimulator;
%Let's define the parameters for transmission duration and perform the
%actual transmission process

d = 50;
c = 3*10^8; % in m/sec
v = 2*c/3;
Tp = d/v; % one-round trip time
R = 11*10^6; %Channel rate is 11Mbps
L = 100*8; %Packet length is 100 byte=800 bits
L_p = 10*8; %Probing packet length is 10 byte = 80 bits
RTT = 2*Tp + (L/R); %round-trip time
RTT_p = 2*Tp + (L_p/R); %round-trip time during probing
n = 1; %window size
num_succ_pac = 0;
num_tx = 1;
timeout = RTT +  1e-005;
%Firstly, let's analyze the classical S-W in terms of efficiency and energy
%consumption
%Let's find RTT refers to how many time slots
for i=1:8192
    if(t(i)<RTT)
        continue;
    end
    if(t(i)>=RTT)
        o = i
        break
    end
end
for i=1:8192
    if(t(i)<timeout)
        continue;
    end
    if(t(i)>=timeout)
        r = i
        break
    end
end
for i=1:8192
    if(t(i)<RTT_p)
        continue;
    end
```

```
    if(t(i)>=RTT_p)
        q = i
        break
    end
end
%RTT is 'o' time-slots
%timeout is 'r' time-slots
%RTT_p during probing is 'q' time slots
m=1;
threshold=ceil(min(Pr))+20;
Pow = Pr;
for fadingmargin=1:21
  num_succ_pac(m) = 0;
  num_tx(m) =0;
  k=1;
  Pr = Pow + fadingmargin -1;
  while(k<=8190)
    for j=k:(k+o-1)
      if(Pr(j)<threshold)
        num_tx(m) = num_tx(m) +1;
        k=k+r-o;%since time out occurs and we observed that timeout is always one more
slot than the RTT, retx occurs one more
        break;
      end
      if((j==k+o-1)&&(Pr(j)>=threshold))
        num_succ_pac(m) = num_succ_pac(m) +1;
        num_tx(m) = num_tx(m) +1;
      end
      if(Pr(j)>=threshold)
        continue;
      end
    end
    k=k+o-1;
  end
  throughput(m)=num_succ_pac(m)/num_tx(m);
  avg_num_tx_per_suc(m)=num_tx(m)/num_succ_pac(m);
  m=m+1;
end
%Secondly, let's analyze the probing S-W in terms of efficiency and energy
%consumption proposed at Zorzi's paper
m=1;
weight = 0.1;
for fadingmargin=1:21
  znum_succ_pac(m) = 0;
  znum_tx(m) =0;
  num_probing_pac(m) = 0;
```

```
   k=1;
   prb = 0;
   Pr = Pow + fadingmargin -1;
   while(k<=8190)
      %update of classical ARQ: if the SNR value is less than the threshold value, enter
probing mode and probe every t slots
      if(Pr(k)<threshold)
         if(prb~=0)
            num_probing_pac(m) = num_probing_pac(m) +1;
            k=k+2;
            continue;
         end
         prb = prb +1;
         num_probing_pac(m) = num_probing_pac(m) +1;
         k=k+q-1;
         continue;
      end
      prb = 0;
%    if((k>6)&&(Pr(k-5)<threshold))
%       num_probing_pac(m) = num_probing_pac(m) +1;
%    end
%
      for j=k+1:(k+o-1)
        if(Pr(j)<threshold)
           znum_tx(m) = znum_tx(m) +1;
           k=k+r-o;%since time out occurs and we observed that timeout is always one more
slot than the RTT, retx occurs one more
           break;
        end
        if((j==k+o-1)&&(Pr(j)>=threshold))
          znum_succ_pac(m) = znum_succ_pac(m) +1;
          znum_tx(m) = znum_tx(m) +1;
        end
        if(Pr(j)>=threshold)
           continue;
        end
      end
      k=k+o-1;
   end
   zthroughput(m)=znum_succ_pac(m)/(znum_tx(m)+num_probing_pac(m));
   zavg_num_tx_per_suc(m)=znum_tx(m)/znum_succ_pac(m);
   zavg_num_probe_per_suc(m)=weight*num_probing_pac(m)/znum_succ_pac(m);
   m=m+1;
end
```

1st Algorithm

```
%Thirdly, let's analyze the probing S-W in terms of efficiency and energy
%consumption based on our proposal
m=1;
weight = 0.1;
rateofchange=toplam/(samples-1);%from the RayleighFadingSimulator.m
for fadingmargin=1:21
  onum_succ_pac(m) = 0;
  onum_tx(m) =0;
  onum_probing_pac(m) = 0;
  k=3;
  prb1 = 0;
  Pr = Pow + fadingmargin -1;
  %update:adapting AFD for probing period
  ss = ceil(threshold)-ceil(min(Pr));
  AFD_THR = AFD(ss+1);
  AFD_THR_slot = ceil(AFD_THR/At);
  while(k<=8190)
    %update of classical ARQ: if the SNR value is less than the threshold value, enter
probing mode and probe every t slots
    %update of previous method:check the SNR during RTT for the worst case
    %by using the rate of channel per time slot
    %update2: how to use AFD?
    %Let's measure the AFD of the current signal
    if(ceil(Pr(k))<ceil(threshold))
      if(prb1~=0)
        s = ceil(Pr(k))-ceil(min(Pr));
        AFD_SNR = AFD(s+1);
        AFD_SNR_slot = floor(AFD_SNR/At);
        onum_probing_pac(m) = onum_probing_pac(m) +1;
        k=k+AFD_THR_slot;
        continue;
      end
      prb1 = prb1 +1;
      onum_probing_pac(m) = onum_probing_pac(m) +1;
      k=k+q-1;
      continue;
    end
    prb1 = 0;

    for j=k+1:(k+o-1)
      if(Pr(j)<threshold)
        onum_tx(m) = onum_tx(m) +1;
        k=k+r-o;%since time out occurs and we observed that timeout is always one more
slot than the RTT, retx occurs one more
```

```
      break;
    end
  if((j==k+o-1)&&(Pr(j)>=threshold))
    onum_succ_pac(m) = onum_succ_pac(m) +1;
    onum_tx(m) = onum_tx(m) +1;
  end
  if(Pr(j)>=threshold)
    continue;
  end
 end
 k=k+o-1;
end
 othroughput(m)=onum_succ_pac(m)/(onum_tx(m)+onum_probing_pac(m));
 oavg_num_tx_per_suc(m)=onum_tx(m)/onum_succ_pac(m);

oavg_num_probe_per_suc(m)=weight*onum_probing_pac(m)/onum_succ_pac(m);%Zor
zi page 285
 m=m+1;
end


figure;
plot(0:20,throughput,'b:*')
grid on;
%title('Throughput vs. FadingMargin')
xlabel('FadingMargin, or Transmitted Power (dBm)')
ylabel('Throughput')

hold on;
plot(0:20,zthroughput,'b:+')

hold on;
plot(0:20,othroughput,'b:x')
legend('S-W','probing,t=2','proposed');

figure;
plot(0:20,avg_num_tx_per_suc,'b:*')
grid on;
%title('Average Number Transmissions per Successful Transmission vs. FadingMargin')
xlabel('FadingMargin, or Transmitted Power (dBm)')
ylabel('Average Number Transmissions per Successful Transmission')

hold on;
plot(0:20,(zavg_num_tx_per_suc+zavg_num_probe_per_suc),'b:+')
% hold on;
% plot(ceil(min(Pr)):floor(max(Pr)),zavg_num_probe_per_suc,'b')
hold on;
```

```
plot(0:20,(oavg_num_tx_per_suc+oavg_num_probe_per_suc),'b:x')
legend('S-W','probing,t=2','proposed');

figure;
plot(0:20,num_succ_pac,'b:*')
grid on;
%title('Throughput vs. FadingMargin')
xlabel('FadingMargin, or Transmitted Power (dBm)')
ylabel('The number of Successful Transmissions during the Simulation Time')

hold on;
plot(0:20,znum_succ_pac,'b:+')

hold on;
plot(0:20,onum_succ_pac,'b:x')
legend('S-W','probing,t=2','proposed');
```

2nd Algorithm

```
%Thirdly, let's analyze the probing S-W in terms of efficiency and energy
%consumption based on our proposal
m=1;
weight = 0.1;
rateofchange=toplam/(samples-1);%from the RayleighFadingSimulator.m
for fadingmargin=1:21
  onum_succ_pac(m) = 0;
  onum_tx(m) =0;
  onum_probing_pac(m) = 0;
  k=3;
  prb1 = 0;
  Pr = Pow + fadingmargin -1;
  %update:adapting AFD for probing period
  ss = ceil(threshold)-ceil(min(Pr));
  AFD_THR = AFD(ss+1);
  AFD_THR_slot = ceil(AFD_THR/At);
  while(k<=8190)
    %update of classical ARQ: if the SNR value is less than the threshold value, enter
probing mode and probe every t slots
    %update of previous method:check the SNR during RTT for the worst case
    %by using the rate of channel per time slot
    %update2: how to use AFD?
    %Let's measure the AFD of the current signal
    if(ceil(Pr(k))<ceil(threshold))
      if(prb1~=0)
        s = ceil(Pr(k))-ceil(min(Pr));
        AFD_SNR = AFD(s+1);
```

```
        AFD_SNR_slot = floor(AFD_SNR/At);
        onum_probing_pac(m) = onum_probing_pac(m) +1;
        k=k+AFD_SNR_slot+ceil(((AFD_THR-AFD_SNR)/2)/At);
        continue;
    end
    prb1 = prb1 +1;
    onum_probing_pac(m) = onum_probing_pac(m) +1;
    k=k+q-1;
    continue;
  end
  prb1 = 0;

  for j=k+1:(k+o-1)
    if(Pr(j)<threshold)
       onum_tx(m) = onum_tx(m) +1;
       k=k+r-o;%since time out occurs and we observed that timeout is always one more
slot than the RTT, retx occurs one more
       break;
     end
    if((j==k+o-1)&&(Pr(j)>=threshold))
       onum_succ_pac(m) = onum_succ_pac(m) +1;
       onum_tx(m) = onum_tx(m) +1;
     end
    if(Pr(j)>=threshold)
       continue;
     end
    end
    k=k+o-1;
  end
  othroughput(m)=onum_succ_pac(m)/(onum_tx(m)+onum_probing_pac(m));
  oavg_num_tx_per_suc(m)=onum_tx(m)/onum_succ_pac(m);

oavg_num_probe_per_suc(m)=weight*onum_probing_pac(m)/onum_succ_pac(m);%Zor
zi page 285
  m=m+1;
end


figure;
plot(0:20,throughput,'b:*')
grid on;
%title('Throughput vs. FadingMargin')
xlabel('FadingMargin, or Transmitted Power (dBm)')
ylabel('Throughput')

hold on;
```

```
plot(0:20,zthroughput,'b:+')

hold on;
plot(0:20,othroughput,'b:x')
legend('S-W','probing,t=2','proposed');

figure;
plot(0:20,avg_num_tx_per_suc,'b:*')
grid on;
%title('Average Number Transmissions per Successful Transmission vs. FadingMargin')
xlabel('FadingMargin, or Transmitted Power (dBm)')
ylabel('Average Number Transmissions per Successful Transmission')

hold on;
plot(0:20,(zavg_num_tx_per_suc+zavg_num_probe_per_suc),'b:+')
% hold on;
% plot(ceil(min(Pr)):floor(max(Pr)),zavg_num_probe_per_suc,'b')
hold on;
plot(0:20,(oavg_num_tx_per_suc+oavg_num_probe_per_suc),'b:x')
legend('S-W','probing,t=2','proposed');

figure;
plot(0:20,num_succ_pac,'b:*')
grid on;
%title('Throughput vs. FadingMargin')
xlabel('FadingMargin, or Transmitted Power (dBm)')
ylabel('The number of Successful Transmissions during the Simulation Time')

hold on;
plot(0:20,znum_succ_pac,'b:+')

hold on;
plot(0:20,onum_succ_pac,'b:x')
legend('S-W','probing,t=2','proposed');
```

3rd Algorithm

```
%Thirdly, let's analyze the probing S-W in terms of efficiency and energy
%consumption based on our proposal
m=1;
weight = 0.1;
rateofchange=toplam/(samples-1);%from the RayleighFadingSimulator.m
for fadingmargin=1:21
  onum_succ_pac(m) = 0;
  onum_tx(m) =0;
  onum_probing_pac(m) = 0;
```

```
k=3;
prb1 = 0;
Pr = Pow + fadingmargin -1;
%update:adapting AFD for probing period
ss = ceil(threshold)-ceil(min(Pr));
AFD_THR = AFD(ss+1);
AFD_THR_slot = ceil(AFD_THR/At);
while(k<=8190)
    %update of classical ARQ: if the SNR value is less than the threshold value, enter
probing mode and probe every t slots
    %update of previous method:check the SNR during RTT for the worst case
    %by using the rate of channel per time slot
    %update2: how to use AFD?
    %Let's measure the AFD of the current signal
    if(ceil(Pr(k))<ceil(threshold))
        if(prb1~=0)
            s = ceil(Pr(k))-ceil(min(Pr));
            AFD_SNR = AFD(s+1);
            AFD_SNR_slot = floor(AFD_SNR/At);
            if((Pr(k)-Pr(k-1)<0)&&(Pr(k-1)-Pr(k-2)<0))
                onum_probing_pac(m) = onum_probing_pac(m) +1;
                k=k+AFD_SNR_slot+ceil(((AFD_THR-AFD_SNR)/2)/At);
                continue;
            end
            if((Pr(k)-Pr(k-1)>0)&&(Pr(k-1)-Pr(k-2)>0))
                onum_probing_pac(m) = onum_probing_pac(m) +1;
                k=k+ceil(((AFD_THR-AFD_SNR)/2)/At);
                continue;
            end
        end
        prb1 = prb1 +1;
        onum_probing_pac(m) = onum_probing_pac(m) +1;
        k=k+q-1;
        continue;
    end
    prb1 = 0;

    for j=k+1:(k+o-1)
        if(Pr(j)<threshold)
            onum_tx(m) = onum_tx(m) +1;
            k=k+r-o;%since time out occurs and we observed that timeout is always one more
slot than the RTT, retx occurs one more
            break;
        end
        if((j==k+o-1)&&(Pr(j)>=threshold))
            onum_succ_pac(m) = onum_succ_pac(m) +1;
```

```
      onum_tx(m) = onum_tx(m) +1;
    end
    if(Pr(j)>=threshold)
      continue;
    end
   end
  end
  k=k+o-1;
 end
  othroughput(m)=onum_succ_pac(m)/(onum_tx(m)+onum_probing_pac(m));
  oavg_num_tx_per_suc(m)=onum_tx(m)/onum_succ_pac(m);

oavg_num_probe_per_suc(m)=weight*onum_probing_pac(m)/onum_succ_pac(m);%Zor
zi page 285
  m=m+1;
end


figure;
plot(0:20,throughput,'b:*')
grid on;
%title('Throughput vs. FadingMargin')
xlabel('FadingMargin, or Transmitted Power (dBm)')
ylabel('Throughput')

hold on;
plot(0:20,zthroughput,'b:+')

hold on;
plot(0:20,othroughput,'b:x')
legend('S-W','probing,t=2','proposed');

figure;
plot(0:20,avg_num_tx_per_suc,'b:*')
grid on;
%title('Average Number Transmissions per Successful Transmission vs. FadingMargin')
xlabel('FadingMargin, or Transmitted Power (dBm)')
ylabel('Average Number Transmissions per Successful Transmission')

hold on;
plot(0:20,(zavg_num_tx_per_suc+zavg_num_probe_per_suc),'b:+')
% hold on;
% plot(ceil(min(Pr)):floor(max(Pr)),zavg_num_probe_per_suc,'b')
hold on;
plot(0:20,(oavg_num_tx_per_suc+oavg_num_probe_per_suc),'b:x')
legend('S-W','probing,t=2','proposed');
```

```
figure;
plot(0:20,num_succ_pac,'b:*')
grid on;
%title('Throughput vs. FadingMargin')
xlabel('FadingMargin, or Transmitted Power (dBm)')
ylabel('The number of Successful Transmissions during the Simulation Time')

hold on;
plot(0:20,znum_succ_pac,'b:+')

hold on;
plot(0:20,onum_succ_pac,'b:x')
legend('S-W','probing,t=2','proposed');
```

4th Algorithm

```
%Thirdly, let's analyze the probing S-W in terms of efficiency and energy
%consumption based on our proposal
m=1;
weight = 0.1;
rateofchange=toplam/(samples-1);%from the RayleighFadingSimulator.m
for fadingmargin=1:21
  onum_succ_pac(m) = 0;
  onum_tx(m) =0;
  onum_probing_pac(m) = 0;
  k=3;
  prb1 = 0;
  Pr = Pow + fadingmargin -1;
  %update:adapting AFD for probing period
  ss = ceil(threshold)-ceil(min(Pr));
  AFD_THR = AFD(ss+1);
  AFD_THR_slot = ceil(AFD_THR/At);
  while(k<=8190)
    %update of classical ARQ: if the SNR value is less than the threshold value, enter
probing mode and probe every t slots
    %update of previous method:check the SNR during RTT for the worst case
    %by using the rate of channel per time slot
    %update2: how to use AFD?
    %Let's measure the AFD of the current signal
    if(ceil(Pr(k))<ceil(threshold))
      if(prb1~=0)
        s = ceil(Pr(k))-ceil(min(Pr));
        AFD_SNR = AFD(s+1);
        AFD_SNR_slot = floor(AFD_SNR/At);
        if((Pr(k)-Pr(k-1)<0)&&(Pr(k-1)-Pr(k-2)<0))
          onum_probing_pac(m) = onum_probing_pac(m) +1;
```

```
        k=k+AFD_SNR_slot+ceil(((AFD_THR-AFD_SNR)/2)/At);
        continue;
      end
      if((Pr(k)-Pr(k-1)>0)&&(Pr(k-1)-Pr(k-2)>0))
        onum_probing_pac(m) = onum_probing_pac(m) +1;
        k=k+ceil(((AFD_THR-AFD_SNR)/2)/At);
        continue;
      end
    end
    prb1 = prb1 +1;
    onum_probing_pac(m) = onum_probing_pac(m) +1;
    k=k+q-1;
    continue;
  end
  prb1 = 0;

  if((ceil(Pr(k))==ceil(threshold))&&((Pr(k)-Pr(k-1)<0)&&(Pr(k-1)-Pr(k-2)<0)))
    if(prb1~=0)
      s = ceil(Pr(k))-ceil(min(Pr));
      AFD_SNR = AFD(s+1);
      AFD_SNR_slot = floor(AFD_SNR/At);
      onum_probing_pac(m) = onum_probing_pac(m) +1;
      k=k+AFD_SNR_slot+ceil(((AFD_THR-AFD_SNR)/2)/At);
      continue;
    end
    prb1 = prb1 +1;
    onum_probing_pac(m) = onum_probing_pac(m) +1;
    k=k+q-1;
    continue;
  end
  prb1=0;

  if((ceil(Pr(k))>ceil(threshold))&&((Pr(k)-Pr(k-1)<0)&&(Pr(k-1)-Pr(k-
2)<0))&&(ceil(Pr(k)-2*rateofchange)<ceil(threshold)))
    if(prb1~=0)
      s = ceil(Pr(k))-ceil(min(Pr));
      AFD_SNR = AFD(s+1);
      AFD_SNR_slot = floor(AFD_SNR/At);
      onum_probing_pac(m) = onum_probing_pac(m) +1;
      k=k+AFD_SNR_slot+ceil(((AFD_THR-AFD_SNR)/2)/At);
      continue;
    end
    prb1 = prb1 +1;
    onum_probing_pac(m) = onum_probing_pac(m) +1;
    k=k+q-1;
    continue;
```

```
    end
    prb1=0;

    for j=k+1:(k+o-1)
      if(Pr(j)<threshold)
        onum_tx(m) = onum_tx(m) +1;
        k=k+r-o;%since time out occurs and we observed that timeout is always one more
slot than the RTT, retx occurs one more
        break;
      end
      if((j==k+o-1)&&(Pr(j)>=threshold))
        onum_succ_pac(m) = onum_succ_pac(m) +1;
        onum_tx(m) = onum_tx(m) +1;
      end
      if(Pr(j)>=threshold)
        continue;
      end
    end
    k=k+o-1;
  end
  othroughput(m)=onum_succ_pac(m)/(onum_tx(m)+onum_probing_pac(m));
  oavg_num_tx_per_suc(m)=onum_tx(m)/onum_succ_pac(m);

oavg_num_probe_per_suc(m)=weight*onum_probing_pac(m)/onum_succ_pac(m);%Zor
zi page 285
  m=m+1;
end



figure;
plot(0:20,throughput,'b:*')
grid on;
%title('Throughput vs. FadingMargin')
xlabel('FadingMargin, or Transmitted Power (dBm)')
ylabel('Throughput')

hold on;
plot(0:20,zthroughput,'b:+')

hold on;
plot(0:20,othroughput,'b:x')
legend('S-W','probing,t=2','proposed');

figure;
plot(0:20,avg_num_tx_per_suc,'b:*')
grid on;
```

```
%title('Average Number Transmissions per Successful Transmission vs. FadingMargin')
xlabel('FadingMargin, or Transmitted Power (dBm)')
ylabel('Average Number Transmissions per Successful Transmission')

hold on;
plot(0:20,(zavg_num_tx_per_suc+zavg_num_probe_per_suc),'b:+')
% hold on;
% plot(ceil(min(Pr)):floor(max(Pr)),zavg_num_probe_per_suc,'b')
hold on;
plot(0:20,(oavg_num_tx_per_suc+oavg_num_probe_per_suc),'b:x')
legend('S-W','probing,t=2','proposed');

figure;
plot(0:20,num_succ_pac,'b:*')
grid on;
%title('Throughput vs. FadingMargin')
xlabel('FadingMargin, or Transmitted Power (dBm)')
ylabel('The number of Successful Transmissions during the Simulation Time')

hold on;
plot(0:20,znum_succ_pac,'b:+')

hold on;
plot(0:20,onum_succ_pac,'b:x')
legend('S-W','probing,t=2','proposed');
```

REFERENCES

[1] Data Sheet of Cisco Aironet 350 Series Client Adapters

[2] Matlab Reference, "Matlab Help", 2006.

[3] T. Bonald, "Flow-level performance analysis of some opportunistic scheduling algorithms", *Euro. Trans. Telecomm.*, vol. 16, no. 1, Jan. 2005, pp. 65-75.

[4] L. F. Chang,"Throughput Estimation of ARQ Protocols for a Rayleigh Fading Channel Using Fade- and Interfade-Duration Statistic" *IEEE Transactions on Veh. Tech.*, vol. 40, no.3, Feb. 1991, pp. 223-229.

[5] A. DeSomine, M. C. Chuah, and O. Yue, "Throughput Performance of Transport-Layer Protocols overWireless LANs" *Global Telecommunications Conference*, vol.1, 29 Nov. - 2 Dec. 1993, pp. 542-549.

[6] H. Djandji, "An Efficient Hybrid ARQ Protocol for Point-to-Multipoint Communication and Its Throughput Performance" *IEEE Transactions on Veh. Tech.*, vol. 48, no. 5, September 1999, pp. 1688-1698.

[7] D. L. Goeckel, "Adaptive coding for time-varying channels using outdated fading estimates" *IEEE Trans. Commun.*, vol. 47, no. 6, June 1999, pp. 844855

[8] A. Gyasi-Agyei, "Multiuser diversity Based Opportunistic Scheduling forWireless Data Networks" *IEEE Commun. Lett.*, vol. 9, no. 7, July 2005, pp. 670-72.

[9] A. Gyasi-Agyei, and S. Kim, "Cross-Layer Multiservice Opportunistic Scheduling for Wireless Networks" *IEEE Telecommunications Mag.*, vol. 44, June 2006, pp. 50-57.

[10] S. Kallel, "Efficient Hybrid ARQ Protocols with Adaptive Forward Error Correction" *IEEE Transactions on Com.*, vol. 42, no. 2/3/4, Feb/March/April 1994, pp. 281-189.

[11] J. F. Kurose, and Keith W. Ross, *Computer Networking: A Top-Down Approach Featuring the Internet.* USA: Addison-Wesley, 2003.

[12] A. Leon-Garcia, and I. Widjaja, *Communication Networks: Fundamental Concepts and KeyArchitectures.* New York, NY: McGraw Hill, 2004.

[13] Q. Liu, S. Zhou, and G. B. Giannakis, "Cross-Layer Combining of AdaptiveModulation and Coding With Truncated ARQ Over Wireless Links" *IEEE Transactions on Wireless Comm.*, vol. 3, no. 5, Sep. 2004, pp.1746-1755.

[14] E. Malkamaki, and H. Leib, "Performance of truncated type-II hibrid ARQ schemes with noisy feedback over block fading channels" *IEEE Trans. Commun.*, vol. 48, Sept. 2000, pp. 1477-1487.

[15] H. Minn, M. Zeng, and V. K. Bhargava, "On ARQ Scheme With Adaptive Error Control" *IEEE Transactions On Veh. Tech.*, vol. 50, no. 6, November 2001, pp. 1426-1436.

[16] R. J. Punnoose, P. V. Nikitin, and D. D. Stancil, "Efficient simulation of Ricean fading within a packet simulator", *Veh. Tech. Conf.*, vol. 2, 2000, pp. 764-67.

[17] T. Rappaport, *Wireless Communications: Principles and Practice.* Upper Saddle River, NJ: Prentice Hall, 2002.

[18] K. Sampath, *A Channel-Aware ARQ Protocol for Energy Efficient Wireless Networking*,M.S. Thesis, Stevens Institute of Technology, Hoboken, NJ, May 2004.

[19] S. D. Sandberg, and M. B. Pursley, "Retransmission Schemes for Meteor-Burst Communications" *Ninth Annual International Phoenix Conference on Comp. and Com.*, 21-23 March 1990, pp. 246-253.

[20] A. Sheth, and R. Han, "An Implementation of Transmit Power Control in 802.11b Wireless Networks" *Technical Report*, CU-CS-934-02, August 2002.

[21] M. Zorzi, and R. R. Rao, "Error Control and Energy Consumption in Communications for Nomadic Computing" *IEEE Transactions On Computers*, vol. 46, no. 3, March 1997, pp. 279-289.