Development of Multiple Protocols in Novel Simulation Environment

by

Venkata Sai Karteek Rupakula

A Thesis Presented in Partial Fulfillment
of the Requirements for the Degree
Master of Science

Approved July 2017 by the
Graduate Supervisory Committee:

Daniel W. Bliss, Chair
Chaitali Chakrabarti
Tom McGiffen

ARIZONA STATE UNIVERSITY

August 2017

ABSTRACT

When one considers the current state of wireless communications, it becomes clear that it is both absolutely amazing and something of a mess. Present communications standards are the result of local optimizations over time that led to a confusing set of suboptimal and fragile wireless standards. Starting from a clean sheet of paper, Bliss Laboratory for Information, Signals, and Systems (BLISS) is considering a fluid set of communications standards co-optimized with flexible but power-efficient computational implementations that will enable the next revolution of wireless communications. The main aim is to enable much higher data rates and much lower data rates with corresponding lower power consumption as the needs of the users vary.

The thesis mainly looks at the different sections of the work done, to prime the development of the protocol development engine. It discusses channel modeling, and system integration of receiver and channel noise. It also proposes a Carrier-Sense Multiple Access (CSMA) Media Access Control (MAC) layer protocol implementation for (Wireless Fidelity) Wi-Fi protocol. This work also talks about the Graphical User Interface (GUI), which is a part of Protocol Development Kit (PDK) - a combination of the Protocol Recommendation Engine (PRE) and simulation package to aid the development of protocols. It also sheds light on the Automatic Dependent Surveillance - Broadcast (ADS-B) radio protocol, that will eventually replace radar as Air Traffic Control's (ATC) primary tool for separating aircraft.

All the algorithms used in this thesis, to define radio operation were in principle defined by mathematical descriptions; however, to test and implement these algorithms they had to be converted to a computer language. There were multiple phases of this conversion. In the first phase, the implementation of these algorithms was done in Matrix Laboratory (MATLAB). To aid this development, basic radio finite state machines and radio algorithmic tools were provided.

DEDICATION

To my family and many friends.

ACKNOWLEDGMENTS

I wish to express my gratitude to my advisor and mentor, Prof Daniel W. Bliss, for mentoring me and providing invaluable insight at every stage of my research. I am deeply indebted for his guidance, encouragement and support throughout my graduate study. I wish to thank Prof. Chaitali Chakrabarti, Dr. Tom McGiffen for serving on my thesis committee. I would like to thank all the collaborating researchers at Google for their help and support. It is the weekly meeting with them that kept me going. I would also like to thank Dr. Tom McGiffen for many useful discussions. Without these discussions, I would have needed many more years to graduate.
I am also grateful to the Google and the School of Electrical, Computer and Energy Engineering at ASU for partially funding my graduate study.

I wish to thank my colleagues in the Bliss Laboratory of Information, Signals, and Systems Lab. I have learnt a lot from them. It is because of them that the lab was such a great place to work in.

I also wish to thank all my friends and roommates in Arizona. Without them I would not have such colorful memories to cherish. Finally and most importantly, I would like to thank my parents. I am grateful to them for fostering my interests in science and engineering. Their unconditional love and sustained support has been, and will always be the force with me.

TABLE OF CONTENTS

CHAPTER Page

CHAPTER Page

LIST OF TABLES

LIST OF FIGURES

Chapter 1

INTRODUCTION

1.1 Motivation

Communications research requires representative simulations, which focus on various aspects of communications. For example, network level simulations often simulate a simplified bit error rate (BER) vs signal-to-noise ratio (SNR) curve, which is a representation of the physical level. As our research focusses more on how these levels affect each other (studying interference and network congestion together, cognitive radio (CR) scenarios, or receiver design vs network capacity), we came up with a unique simulation environment.

The simulation environment we use is based on time domain sampling, that allows simultaneous simulation of lower level and higher level details. As time domain sampling is used, simulating dissimilar radios, radios out of sync, or interference is straightforward. Additionally, modeling a network of radios is straightforward, as that just means managing a stream of samples from each radio at the physical layer, simulating channel processing, then managing a stream of appropriately processed samples to each receiver. Researchers often need to simulate a variety of protocols. Cognitive radio (CR) may require simulating multiple protocols simultaneously. Additionally, researchers often develop an environment to simulate one protocol, and then generate a new environment when they have work on another protocol. A single environment supporting multiple protocols addresses both of these issues. This facilitates studying a different radio protocol, as the existing simulation environment is reused, and software

development is limited to the different radio. We have also been able to develop a common software interface for radios such that a variety of radio protocols can just be "plugged in", allowing one to leverage off of the tools and experience associated with the existing environment. It supports study of multiple protocols operating simultaneously, which is of increasing importance in today's environment.

1.2 Google R2

The Google Radio Revolution (R2) program attempts to face the impending Cambrian explosion in communications by reinventing the thinking behind the usage of protocols. The main aim of google R2 is to open wireless development and enable a wide range of benefits that include economic growth, reduced wasted energy, and support of economically less advantaged populations. To accomplish all these goals, we came up with the idea of developing an ecosystem of protocol developers. To enable this ecosystem, we made use of tools that reduce the non-recurring costs dramatically and developed the Protocol Development Kit (PDK) as an attempt to address these needs. It addresses the underlying system needs to enable fluid protocols. There are two aspects to this effort:

- Improving the versatility of communications hardware both in terms of computational capabilities, and the RF transceiver center frequency and bandwidth flexibility
- Developing computational architectures that can efficiently satisfy the computations needed for a wide range of protocols and standards

Furthermore, it will also reduce the engineering overhead in developing these protocols by providing a set of tools that will allow easier interactions between environmental parameters, protocol parameter definition, simulation, initial experimental validation, and hardware implementation.

1.3 Thesis Contribution

In this thesis, we consider a fluid set of communications standards that are co-optimized with power-efficient computational implementations. The main aim was to enable much higher data rates and much lower data rates with corresponding lower power consumption as the needs of the users vary. It is a well known fact that the current communications do not take full advantage of spectral allocations, and they lack robustness. Hence, all the time and effort were put in to demonstrate a flexible wireless system, develop fluid protocols, and break rigidity of the current radio standards. The principle contributions of this thesis are:

- To provide a new approach for protocol selection and implementation using the PDK
- To provide a Graphical User Interface (GUI) for the whole system
- Analyze and develop realistic channel models for different environments
- Develop and deliver MATLAB code to implement ADS-B, Bluetooth radio protocols
- Perform system level integration of channel and receiver noise
- Architect and implement a code reuse solution to integrate different radios

Apart from the steps mentioned above, debugging was also a major step in successfully integrating the previously mentioned protocols. Debugging involved identifying various problems, isolating the source of the problems, and then correcting them or determining a way to work around them. Finally, all the implemented protocols were tested to verify that the results were as desired.
As a final remark, it is important to acknowledge that much of the inspiration and motivation for this work derived from the vision of the next generation of wireless communications. In the end, it is this vision that provided the guiding framework.

1.4 Thesis Organization

This thesis is organized as follows. Chapter 2 proposes a GUI for the entire simulation package. Chapter 3 talks about mobile radio propagation. It also provides a background on channel modeling. Chapter 4 explains the CSMA MAC layer protocol. Chapter 5 discuss the ADS-B and its implementation. Chapter 6 gives an insight into the synchronization in LTE and also talks about an efficient code reuse solution. Finally in Chapter 7, conclusions and extensions to future work are discussed.

Chapter 2

GRAPHIC USER INTERFACE

2.1 GUI - PDK User Guidance

The GUI is a part of the PDK - a combination of the PRE and simulation package, to aid the development of protocols. GUI is just a tool to suggest and explore protocol parameters. The whole process begins by setting environmental parameters in the GUI. The recommendation engine then provides initial protocol parameter suggestions, and lets the user modify these based upon other engineering insights. Various types of parameters that specify an environmental operating point are provided for the user to choose. The main aim of the GUI is to help the user navigate the various environmental operating points quickly. There are a variety of parameter specifications which the user can play with. Some of the parameters include topology, modulation and coding. They also include computational implementation choices that are dependent upon hardware details. Similarly, there are choices for transceiver settings as well, which will be discussed in detail in the following sections.

2.2 MATLAB GUI

Matlab is one of the most popular platforms for scientific and engineering computing. This is due to the inherently implemented plethora of tools for most engineering/scientific fields, and to its high computing level implementation which offers the advantage of a sharp learning curve to a beginner.

Matlab has a GUI designer "Guide", which is relatively easy to get familiar with and requires little knowledge of programming. Hence, the GUI was developed from scratch using MATLAB. The addition of elements such as plots, buttons and sliders is straightforward. It even provides an easy way to build GUIs with elements such as check boxes, radio buttons, list boxes, pushbuttons etc. and link them to built-in Matlab functions. The purpose of the GUI is to view and control the various user parameters graphically, quickly, easily and accurately. This was done by exploiting the powerful computational and graphical functions of Matlab. It is constructed using the embedded "GUIDE" Matlab tool for building GUIs [1].

2.3 Graphical User Interface

In this section we will have a look at the various GUI portions that were used for running the simulations. All the sections will be briefly discussed with the help of a figure for each. First let us have a look at the environment specifications part of the GUI. It is shown in figure 1.

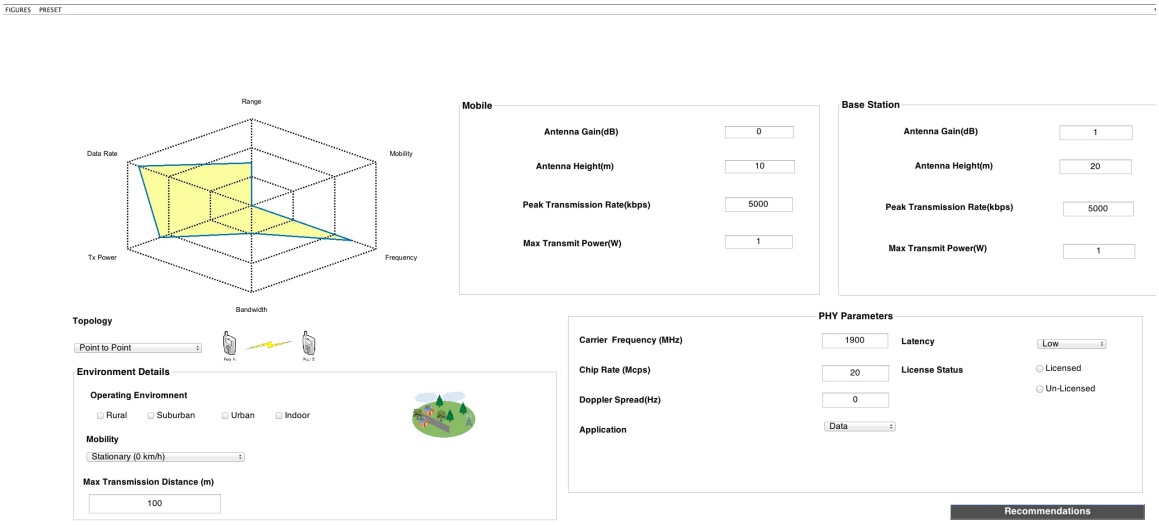


Figure 1: Initial environmental definition GUI

The protocol parameter recommendation engine provides guidance based upon the environmental parameters. With the help of the GUI, the user can select the desired parameters. At this point of time the user can play with the following parameters:

- Network Topology
  - Point to Point
  - Mesh
  - Star
- Operating Environment
  - Rural
  - Urban and Suburban
  - Indoor
- Mobility
- Transmission distance

The user can also specify the following PHY Parameters:

- Carrier frequency, Chip rate and Doppler spread

The user can also set the mobile and Base Station parameters such as

- Antenna gain
- Antenna height
- Transmit power
- Peak transmission rate

As we can see, the guidance provided by the PRE is a function of available network topology, user density, channel delay spread, channel doppler spread, and spectral allocation. It also depends on the available hardware (including number of antenna channels, computational tools, and transceiver flexibility), and goals for wakeup latency, data-rate, data latency, and power expenditures.

After the user inputs the desired parameters, the user clicks on the **RECOMMENDATIONS** push button which moves on to the next portion of the GUI, which is discussed below.

Now let us have a look at the Recommendations part of the GUI. It is shown in figure 2.

This has two main sections:

- Recommendations
- Implemented PHY and MAC Specifications

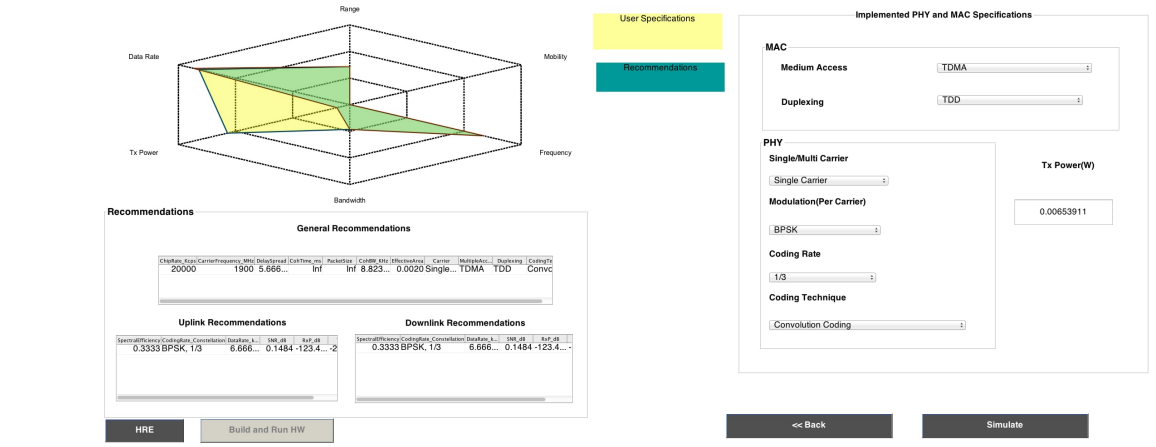


Figure 2: Recommendations Page

The recommendations portion is divided into three sections:

- General Recommendations: In this section the parameters such as delay spread, carrier technique to be used(single/multi), Multiple access method (TDMA/FDMA/CSMA), duplexing and coding technique to be used are specified. These are generated by the PRE.
- Uplink Recommendations and Downlink Recommendations: In this section, achievable spectral efficiency, coding rate, data rate, Signal-to-noise ratio (SNR) and the transmit and receive powers are displayed so that the user can look at these parameters and can go ahead with the recommended protocol, if he or she is satisfied.
- Implemented PHY and MAC Specifications

Now that the user has the recommendations generated by the PRE, the next step is to run the simulation. The user can either go ahead with these recommendations or can override them. This is done in the **Implemented PHY and MAC specifications** section.

Here the user can override the following:

- MAC parameters
  - Medium access (TDMA/FDMA)
  - Duplexing (TDD/FDD)
- PHY parameters
  - Single/Multi carrier
  - Modulation scheme (BPSK/QPSK/16 QAM/64 QAM)
  - Coding rate and technique

After the user inputs the desired parameters, the user clicks on the **SIMULATE** push button which moves on to the next portion of the GUI, which is discussed below. Now let us have a look at the simulation results part of the GUI. An example simulations result page is shown in figure 3.

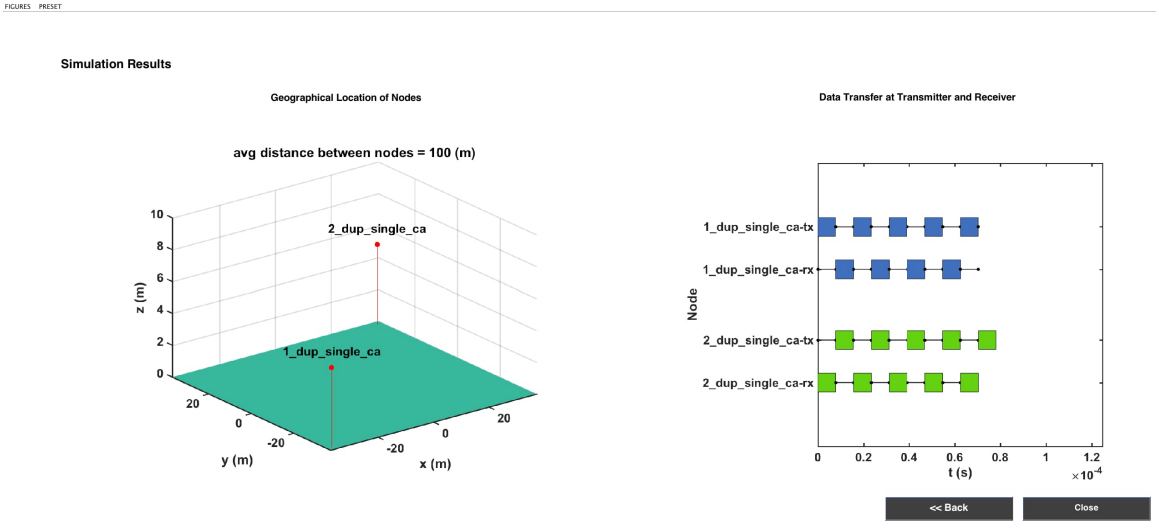


Figure 3: Simulation Results Page

After the user presses on the simulation push button, the simulation starts on MATLAB. Once the simulation is successfully completed, the simulation results are displayed on this page. Two plots are displayed as a part of the result.

- Graphical Location of the nodes: It is a 3D plot which tells the user the location of the nodes.
- Timing diagram: It shows the time at which the packets were transmitted and received by the nodes.

We have also included various plots which the user can have a look at and evaluate how the efficient the protocol is. The user can view various plots from the *Figures* drop-down menu such as

- SNR-Range
- Delay spread
- Tx and Rx spectrum
- Constellation diagram
- Bit Error rate (BER)

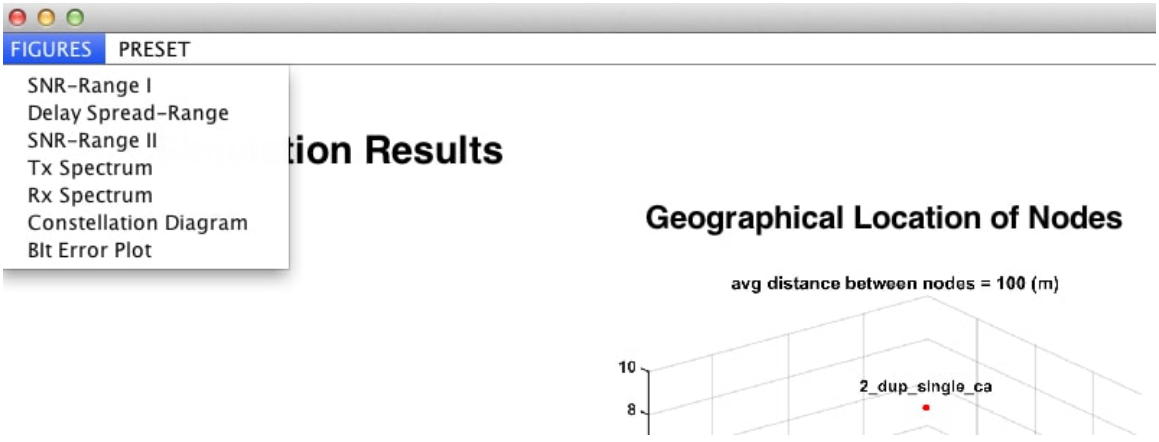


Figure 4: Figures: Drop down menu

There is also a *Hardware Recommendation Engine* (HRE) push button which the user will use, to find out regarding the the hardware specifications which will support the given inputs. HRE takes the parameters and develops an efficient implementation given a processor system. It determines how to take the protocol which is developed and translate that into an executable which runs across the processor. It develops an implementation which is then installed on the radios.

Next to the push button is the *Build and Run HRE* push button which will install the protocols on the processor and execute them on the radios to transmit signals over the air. It can be seen in the figure below.

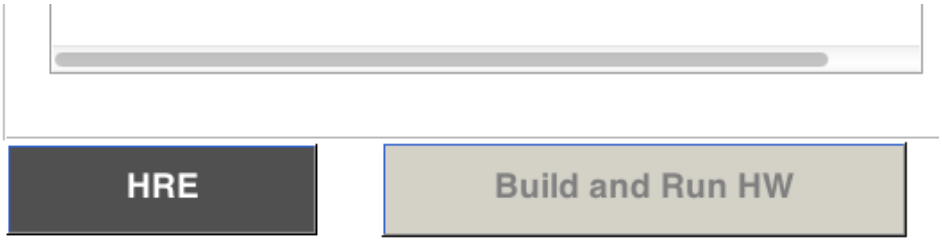


Figure 5: HRE Pushbutton

To make it more easy for the user while giving the inputs, we have a preset pull down menu for quick input selection. We have the all the inputs *preset* on the *environmental definition* page of the GUI. At the moment it is available for LTE, ADS-B, Bluetooth, and Wi-fi. It can be seen in figure 6.

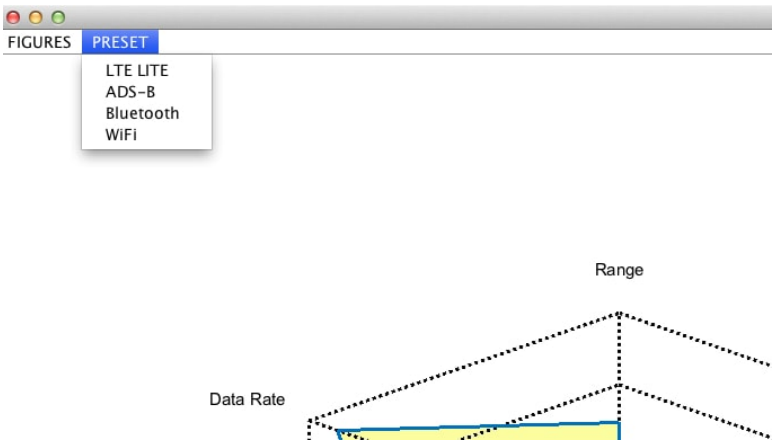


Figure 6: Preset: Drop down menu

To make it more easy for the user to simulate a few standard protocols, we have a preset pull down menu for quick simulation. It is provided on the *environmental definition* page of the GUI. At the moment, simulations are available for LTE-TDMA, LTE-FDMA, LTE-MBMS, ADS-B and Bluetooth.

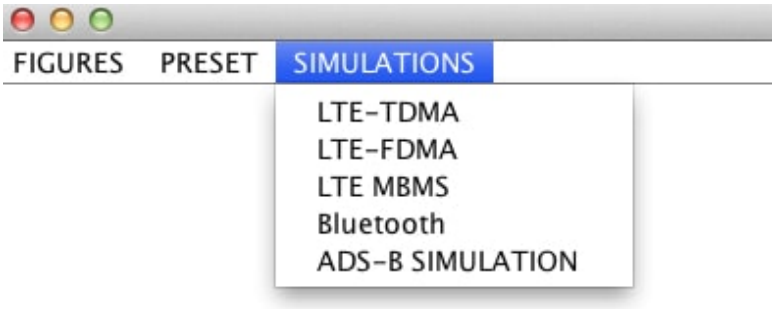


Figure 7: Simulations: Drop down menu

Chapter 3

MOBILE RADIO PROPAGATION

Until now the channel which we had used for our simulations, was a basic one. We wanted to make the simulations as realistic as possible, thus providing the user with a better platform to evaluate a given protocol. Hence, we considered developing a channel model which accounts for small-scale and large-scale fading effects.

The following sections talk about the different propagation models considered. The final section of this chapter will talk about the channel model which was modeled for simulations.

The mobile radio channel places fundamental limitations on the performance of wireless communication systems. The transmission path between the transmitter and the receiver can vary from simple line-of-sight to one that is severely obstructed by buildings, mountains, and foliage. The wireless channels are highly unpredictable and random in nature [2]. Modeling the wireless channel is, and has been the most difficult parts of a mobile radio system. In this chapter we discuss about large-scale and small-scale fading.

3.1 Large-Scale Path Loss

Path loss is caused by the dissipation of the power radiated by the transmitter as well as by the effects of the propagation channel [3]. It is the power loss involved in transmission between the base station (BTS) and the mobile station (MS). It depends on various factors like the antenna height, carrier frequency and transmission distance.

Large-scale fading is the result of signal attenuation due to signal propagation over large distances and signal being blocked by large objects in the propagation path, which is known as shadowing. These are typically distant objects in the environment such as mountains, hills, or large buildings. The resulting variations in received power, due to path loss over relatively large distances, are referred to as *large-scale propagation effects*.

3.2 Basic Propagation Mechanisms

The three propagation mechanisms that influence propagation in a mobile communication system are reflection, diffraction, and scattering.

Reflection occurs when a propagating electromagnetic wave impinges upon an object which has very large dimensions when compared to the wavelength of the propagating wave. They generally occur from the surface of the earth and from buildings and walls.

Diffraction occurs when a radio wave on meeting an obstacle bends around the obstacle, which results in change of direction of part of the wave energy from the

normal line-of-sight path. It depends on the geometry of the object, as well as the amplitude,phase, and polarization of the incident wave at the point of diffraction. *Scattering* occurs when a propagating electromagnetic wave impinges upon an object which has very small dimensions when compared to the wavelength of the propagating wave. Generally, lamps or street signs can induce scattering.

3.3 Outdoor Propagation Model

The radio propagation model is an empirical mathematical formulation that characterizes radio wave propagation as a function of distance between transmitter and receiver, function of frequency and function of other condition. A propagation model is usually developed to predict the behavior of propagation in different environments. Radio propagation models are empirical in nature and are developed based on large amounts of data collected for the specific scenario. These models can be broadly classified into three types - Empirical, Deterministic and Stochastic [3].

3.3.1 Hata Model

A model that is widely used for predicting path loss in mobile wireless system is the Hata model. It is simply the empirical formulation of the graphical path loss data produced by Okumura and is valid for a frequency range of 150MHz and 1500MHz. It predicted the median path loss for the distance, **d**, from transmitter to receiver antenna up to 20 km.

The transmitter and receiver antenna heights are considered to be in between 30m to 200m, and 1m to 10m respectively [4]. It is presented in the urban area propagation loss as a standard formula and equations are available for suburban areas as well.

The Hata model for urban areas is formulated as follows:

$$L_U = 69.55 + 26.16 \log_{10} f - 13.82 \log_{10} h_B - C_H + [44.9 - 6.55 \log_{10} h_B] \log_{10} d \tag{3.1}$$

For large cities,

$$C_H = 3.2(\log_{10}(11.75 h_M))^2 \; - \; 4.97 \tag{3.2}$$

The Hata model for suburban areas is formulated as following:

$$L_{SU} = L_U \; - \; \left(\log_{10} \frac{f}{28}\right)^2 \tag{3.3}$$

where

L_U= Path loss in urban areas (dB)

L_{SU}= Path loss in suburban areas (dB)

h_B= Height of base station antenna (m)

h_M= Height of mobile station antenna (m)

f= Frequency of transmission (MHz)

C_H= Antenna height correction factor

d= Distance between the base and mobile stations (km)

3.4 Indoor Propagation Model

There been considerable interest recently in propagation prediction for indoor environments. The reason is that indoor propagation prediction is becoming very

useful to mobile telephone operators that provide services in large cities, where many subscribers are pedestrians demanding that coverage be provided within buildings, shopping malls, and train stations [5]. Indoor radio propagation is also dominated by reflection, diffraction, and scattering but it also considers partition losses between different floors of a building, floor attenuation, antenna placement. This section outlines the ITU model.

3.4.1 ITU Model

The ITU indoor propagation model, also known as ITU model for indoor attenuation, is a radio propagation model that estimates the path loss inside a room or a closed area inside a building delimited by walls of any form. This model is applicable to only the indoor environments. The ITU indoor path loss model is formally expressed as:

$$L = 20\log(f) + N\log_d + P_f(n) - 28 \tag{3.4}$$

where

L= Path loss (dB)

f= Frequency of transmission (MHz)

d= Distance (m)

N= The distance power loss coefficient

n= Number of floors between the transmitter and receiver

$P_f(n)$= Floor loss penetration factor

3.5 Small-Scale Fading and Multipath

The motion in space of a wireless receiver operating in a multipath channel results in a communications link that experiences small-scale fading. The term small-scale fading describes the rapid fluctuations in received power level due to small sub-wavelength changes in receiver position. This effect is due to the constructive and destructive interference of the numerous multipath waves that impinge upon a wireless receiver [6]. Small scale fading is a characteristic of radio propagation resulting from the presence reflectors and scatterers that cause multiple versions of the transmitted signal to arrive at the receiver, each distorted in amplitude, phase and angle of arrival.

3.5.1 Small-Scale Multipath Propagation

Multipath in the radio channel creates small-scale fading effects. The three most important effects are:

- Rapid changes in the signal strength over a small travel distance or time interval
- Random frequency modulation due to varying Doppler shifts on different multipath signals
- Time dispersion caused by multipath propagation delays [2]

3.5.2 Parameters of Mobile Multipath Channels

The following are some of the multipath channel parameters:

- Delay spread and coherence bandwidth
- Doppler spread and coherence time

All these parameters are derived from the power delay profile. The power delay profile gives the distribution of signal power received over a multipath channel as a function of propagation delays. In a power delay profile plot, the signal power of each multipath is plotted against their respective propagation delays. A transmitted pulse gets received at the receiver with different signal strength as it travels through a multipath channel with different propagation delays(τ_0, τ_1, τ_2).
Delay spread and coherence bandwidth are the parameters which describe the time dispersive nature of the channel, whereas Doppler spread and coherence time describe the time varying nature of the channel. Delay spread can be interpreted as the difference between the time of arrival of the earliest significant multipath component (typically the line-of-sight component) and the time of arrival of the latest multipath components.

Coherence bandwidth is a statistical measure of the range of frequencies over which the channel can be considered flat (i.e., a channel which passes all spectral components with approximately equal gain and linear phase). In other words, coherence bandwidth is the range of frequencies over which two frequency components have a strong potential for amplitude correlation.

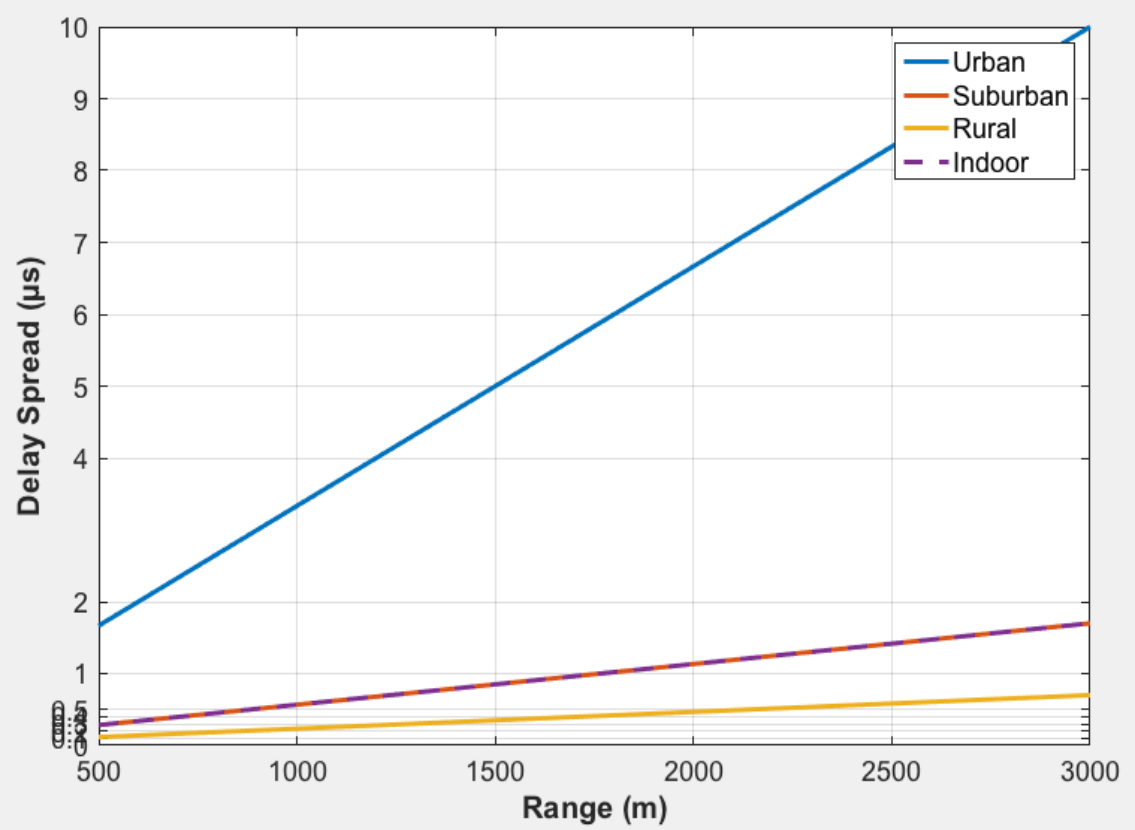


Figure 8: Delay Spread Plot

Delay spread plot for different environments

Delay spread and coherence bandwidth are inversely proportional to one another.

$$B_c \approx \frac{1}{50\sigma_\tau} \tag{3.5}$$

Doppler spread B_D is a measure of the spectral broadening caused by the time rate of change of the mobile radio channel and is defined as the range of frequencies over which the received Doppler spectrum is essentially non-zero. Coherence time T_c is the time domain dual of Doppler spread.

It is the time duration over which two received signals have a strong potential for amplitude correlation. Doppler spread and coherence time are inversely proportional to one another.

$$T_c \approx \frac{1}{f_m} \tag{3.6}$$

3.6 Frequency Selective Fading

It is a well known fact that multipath delay spread leads to time dispersion and frequency selective fading. This section talks about frequency selective fading. A channel is said to go frequency selective fading if:

- Bandwidth of signal > Bandwidth of channel
- Delay spread > symbol period

It is mainly due to time dispersion of the transmitted symbols within the channel. Thus the channel induces intersymbol interference (ISI). Viewed in the frequency domain, certain frequency components in the received signal spectrum have greater gains than others [2].

3.6.1 Ricean Distribution

The model behind Rician fading is similar to that for Rayleigh fading, except that in Rician fading a strong dominant component is present. This dominant component can for instance be the line-of-sight wave. The in-phase and quadrature phase component of the received signal are i.i.d. jointly Gaussian random variables. The Ricean K - factor is defined as the ratio of power in the Line-of-sight (LOS)

component to the power in the other (non-LOS) multipath components. For K=0, we have Rayleigh fading and, for K=∞ we have just only a LOS component.

3.7 Channel Model for Simulations

Until now for all the simulations that were being carried out, the channel was just a basic channel without taking into consideration large scale or small scale fading effects. We decided to implement a channel which was more realistic. For the channel implementation, multipath propagation and large scale fading were taken into consideration, thus making it similar to the real life wireless channel. The technical document *Universal Mobile Telecommunications System (UMTS); Spatial channel model for Multiple Input Multiple Output (MIMO) simulations* was referred to for various specifications. I had considered the following two environments:

- Suburban macrocell
- Urban macrocell

The characteristics of the macro cell environments assume that BS antennas are above rooftop height.

Table 1: Environment Parameters

Channel Scenario	Suburban Macro	Urban Macro
Number of paths (N)	9	9
Delay spread	$\mu_{DS} = -6.80,\ \epsilon_{DS} = 0.288$	$\mu_{DS} = -6.18,\ \epsilon_{DS} = 0.18$
$\sigma_{DS} = 10^{\epsilon_{DS}x+\mu_{DS}}, x \in \eta(0,1)$		
$r_{DS} = \sigma_{delays}/\sigma_{DS}$	1.4	1.7

For a given scenario and set of parameters given by a column of Table 1 [7], realizations of each user's parameters such as the path delays, powers were derived.

3.7.1 Generating user parameters for urban macrocell and suburban macrocell environments

- Urban macrocell or suburban macrocell environment was chosen.
- Random delays were determined for each of the N multipath components. For macrocell environments, N = 9 as given in Table 1. Generate random variables τ'_1 ,..., τ'_n according to

$$\tau'_n = -r_{DS}\,\sigma_{DS}\,ln\,z_n \quad n = 1, ..., N \tag{3.7}$$

  In the equation above, $z_n(n = 1, ..., N)$ are i.i.d. random variables with uniform distribution U(0,1). r_{DS} and σ_{DS} are given in Table 4. The delays were ordered in the decreasing order and the minimum was subtracted from all. The delay for the nth path τ_n which is the value of $\tau'_n - \ \tau'_1$ were quantized in time to the nearest 1/16th chip interval.
- Next, random average powers for each of the N multipath components. The unnormalized powers be given by

$$P'_n = e^{\frac{(1-r_{DS})(\tau'_n - \ \tau'_1)}{r_{DS}\,\sigma_{DS}}}.10^{\frac{-\xi}{10}} \quad n = 1, ..., 6 \tag{3.8}$$

  In the equation above ξ_n(n = 1,...6) are are i.i.d. random variables with standard deviation $\sigma_{RND} = 3dB$ which is a shadowing randomization effect on the per-path powers. Average powers were normalized so that the total average power for all six paths was equal to one.

Once the path delays and path gains were found out, they were given as the input parameters to The **RicianChannel** system object in MATLAB, which filters an input signal through a Rician multipath fading channel. The Rician K-factor was set to one and the direct path Doppler shift was set to zero.

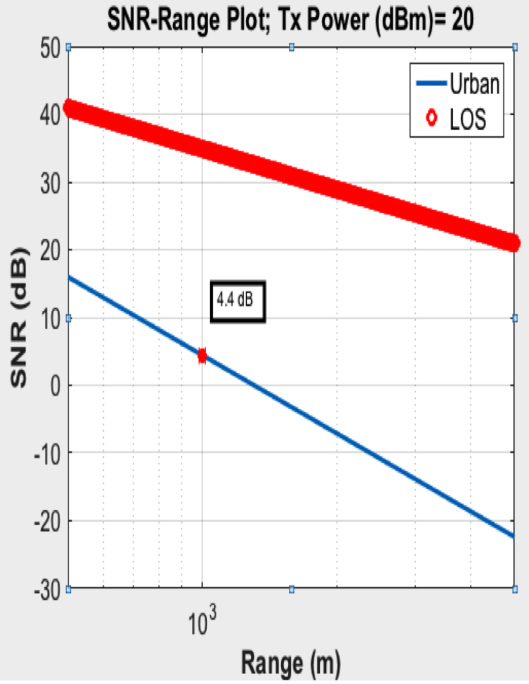


Figure 9: SNR plot I

SNR Plot for Urban environment, with transmit distance of 1000m, Antenna heights of 10m and 20m, Tx and Rx powers of 0.01 W

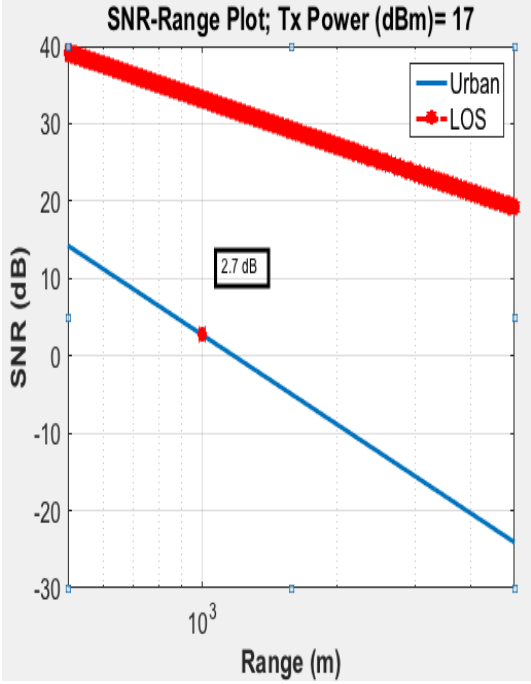


Figure 10: SNR Plot II

SNR Plot for Urban environment, with transmit power as suggested by the PRE

For indoor environments, the channel was modeled as two clusters each of which represented an independent propagation path between the transmitter and the receiver. A cluster is composed of subpaths or taps which share angular spreads, angles of arrival, and angles of departure. Delay and power level vary from tap to tap. The cluster parameter (path gain) for both the clusters was chosen as per the values specified. The path delays after the first were typically chosen to in between 1 ns and 100 ns.

3.8 System Integration : Noise

This section deals with the system integration of the channel and receiver noise. Systems integration commonly refers to "making sure the whole simulation package works". That along with receiver equalization, and transmitter rate adaptation, was what was targeted, but these latter two issues are not yet worked upon.

3.8.1 Noise

Noise is an unwanted electrical or electromagnetic energy that degrades the quality of signals and data that occurs in digital and analog systems, and can affect communications system of all types. It is generated by random vibrations of conducting electrons and holes in the material. It is often referred to as thermal noise. In a warm resistor, free electrons move in thermally excited motion. This gives rise to a noise voltage that appears across the resistor's terminals. This noise was first analyzed in 1927 by J.B Johnson of the Bell Telephone Laboratories, and it goes by the name *thermal noise*, *white gaussian noise*, *Johnson noise* and kTB noise. This chapter describes a simple model for thermal noise used for the simulations.

3.8.2 Noise Model

3.8.2.1 Thermal Noise

It is referred to as "Thermal Noise" because of the dependency on temperature. The total thermal noise power is a function of three quantities:

- Boltzmann's constant "**k**" in $Joules/^{\circ}K$
- Temperature in **Kelvin**
- Overall Bandwidth of the channel selective filtering in the receiver

The equation for thermal noise power is as below

$$\text{Noise Power} = kTB \tag{3.9}$$

where

k = Boltzmann's constant (1.38e-23)

T = Temperature (290°K)

The convention for the temperature of T is set by IEEE standard to be 290°K, which is close to ordinary room temperature. So we have assumed T = 290°K

B = Bandwidth(Hz)

Thermal noise is spread more or less uniformly over the entire frequency spectrum. Therefore, the amount of noise appearing in the output of an ideal receiver is proportional to the absolute temperature of the receiver input system (antenna etc) times the bandwidth of the receiver. The factor of proportionality seen in the above equation is the Boltzmann's Constant. The resulting noise is in Joules/Second or Watts. To convert the noise power to dBW, we have to use 10 times the log of the noise power in watts.

To get the total Noise power we need to sum the thermal noise power at the input of the system with Noise figure (NF). Let us have a look at what it is.

3.8.2.2 Noise Figure

The Noise figure is the amount of noise power added by the electronic circuitry in the receiver to the thermal noise power from the input of the receiver. The thermal noise at the input to the receiver passes through to the demodulator. This noise is present in the receive channel and cannot be removed. The noise figure of circuits in the receiver such as amplifiers and mixers, adds additional noise to the receive channel. This raises the noise floor at the demodulator.

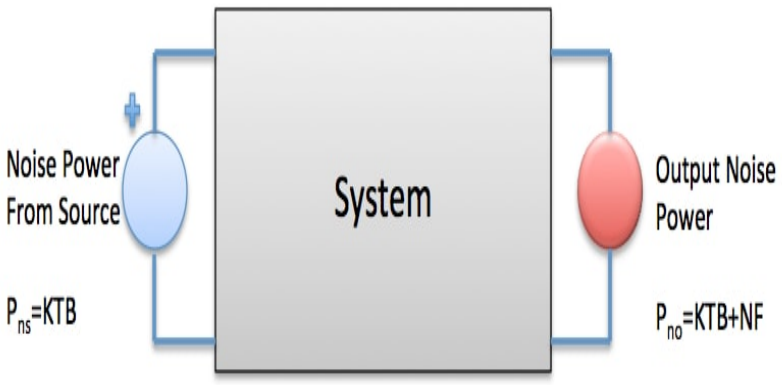


Figure 11: Block Diagram

Figure showing the basic receiver noise block diagram

Noise figure has nothing to do with modulation or demodulation. It is independent of the modulation format and of the fidelity of modulators and demodulators. It is more general concept than noise-quieting used to indicate the sensitivity of FM receivers or BER used in digital communications.

Now that we have the thermal noise at the input, we added the additional noise added by the system (the NF) to get the noise power at the output.

$$\text{Noise Power at output} = kTB + NF \tag{3.10}$$

We have considered a Noise Figure of 4 (linear) for the simulations. For the noise model implemented, we calculate the noise power at the input, noise power at the output. As we know a random variable's power equals its mean-squared value and as, the noise has zero mean, it's power equal to its variance. So now we add this variance to the generated samples. Thus our noise model is complete.

Chapter 4

MEDIUM ACCESS CONTROL PROTOCOL

Until now, our focus was on the Physical Layer (PHY). As most of the systems we see in our daily life use a common medium we wanted to implement a MAC layer protocol, as it is necessary by systems that share a common communications medium. Often viewed as the "brains" of the network, the MAC Layer uses PHY, to perform the tasks of carrier sensing, transmission, and receiving of frames. We chose to implement a protocol used by stations contending for access to a shared medium like an Ethernet cable or a radio channel. We found that there were multiple flavors of CSMA.

- CSMA/CD (Collision Detection) is used by Ethernet stations to detect frame collisions on 802.3 wired (Local Area Network) LANs.
- CSMA/CA (Collision Avoidance) is used by Wi-Fi stations to avoid frame collisions on 802.11 wireless LANs.

As our focus was on wireless networks we decided to go ahead with CSMA/CA used by Wi-Fi stations. This chapter talks about the need for such medium-access control, working of the CSMA protocol, and implementation of CSMA protocol in detail.

4.1 The need for medium-access control

In wireless and certain wired networks, multiple users share the same physical medium. Data communication rates in networks can often be improved by using medium-access control protocols, whereby multiple users share the medium in a

controlled manner such that the adverse effects of their interfering signals is reduced [8]. Multiple access techniques permit multiple access to a channel. They allow several terminals connected to the same multi-point transmission medium to transmit over it and to share its capacity. Examples of shared physical media are wireless networks, bus networks, ring networks and point-to-point links operating in half-duplex mode. It is based on a multiple access protocol and control mechanism, also known as MAC. It deals with issues such as addressing, assigning multiplex channels to different users, and avoiding collisions. Media access control is a sub-layer in Layer 2 (data link layer) of the Open Systems Interconnection (OSI) model and a component of the link layer of the Transmission Control Protocol/Internet Protocol (TCP/IP) model.

4.2 CSMA

Before CSMA was developed, systems made use of slotted aloha protocol. In such a system, a node decides whether or not to transmit in a passive manner, without trying detect the presence of a carrier signal from another node. A simple modification to the Aloha protocol is to have nodes determine whether another transmission is in progress before initiating a transmission. It makes sure that the transmission is only initiated if the transmitting nodes does not detect any ongoing transmission. This type of protocol is called CSMA.

4.2.1 Working of CSMA

The fundamental access method of the IEEE 802.11 MAC is a Distributed Coordination Function (DCF) known as CSMA/CA. The DCF shall be implemented in all Stations (STAs). For a STA to transmit, it shall sense the medium to determine if another STA is transmitting. If the medium is not determined to be busy, the transmission may proceed.

4.2.1.1 CS mechanism

Every Wi-Fi station must first determine the state of the medium as idle or busy through the process of carrier sense, prior to being allowed to perform pro-active collision avoidance and ultimately transmit a frame. Physical and virtual carrier sensing functions are used to determine the state of the medium. When either function indicates a busy medium, the medium shall be considered busy; otherwise, it shall be considered idle. A physical CS mechanism shall be provided by the PHY while a virtual CS mechanism shall be provided by the MAC.
Under the physical sense we have Clear Channel Assessment (CCA). It is physical carrier sense which listens to received energy on the radio interface. It indicates a busy medium for the current frame. CCA is defined in the IEEE 802.11-2007 standards as part of the Physical Medium Dependent (PMD) and Physical Layer Convergence Protocol (PLCP) layer. Clear Channel Assessment is composed of two related functions, Carrier Sense (CS) and Energy Detection (ED).

- *CS* refers the ability of the receiver to detect and decode an incoming Wi-Fi signal preamble. In addition, CCA must be reported as **busy** when another Wi-Fi signal preamble is detected, and must be held as **busy** for the length of the received frame as indicated in the frame's Physical Layer Convergence Protocol (PLCP) length field.
- *ED* refers to the ability of the receiver to detect the non-Wi-Fi energy level present on the current channel (frequency range) based on the noise floor, ambient energy, interference sources, and unidentifiable Wi-Fi transmissions that may have been corrupted but can no longer be decoded. Unlike carrier sense which can determine the exact length of time the medium will be busy with the current frame, energy detection must sample the medium every slot time to determine if the energy still exists. In addition, energy detection requires a pre-defined threshold which determines if the reported energy level is adequate to report the medium as busy or idle.

Now let us look at virtual carrier sensing. Network Allocation Vector (NAV) is the virtual carrier sense which is used by stations to reserve the medium for mandatory frames which must follow the current frame. NAV reserves the medium as busy for future frames that are required to be transmitted immediately following the current frame. This is important to reserve the medium as busy for these mandatory frames. The MAC layer frame headers contain a duration field that specifies the transmission time required for the frame, in which time the medium will be busy. The stations listening on the wireless medium read the Duration field and set their NAV, which is an indicator for a station on how long it must defer from accessing the medium.NAV may be thought of as a counter, which counts down to zero at a uniform rate. When the counter is zero, the virtual CS indication is that the medium is idle; when nonzero,

the indication is busy. The medium shall be determined to be busy when the STA is transmitting. In IEEE 802.11, the NAV represents the number of microseconds the sending STA intends to hold the medium busy (maximum of 32,767 microseconds). To briefly summarize, the sender first sends an RTS frame which includes the duration of time that it needs to occupy the channel. The stations that are affected by this transmission create the NAV timer that shows how much time must pass before these stations are allowed to check the channel for idleness. If the channel is free, the destination replies with a CTS. Only then does the actual DATA/ACK exchange happen. Neighboring nodes that receive either the RTS or CTS set their NAV so as to reserve the channel for the impending DATA/ACK transmission. In other words, each station, before sensing the physical medium to see if it is idle, first checks its NAV to see if it has expired.

4.3 CSMA Implementation

The following are the steps in the CSMA protocol implemented for a peer to peer topology. The same steps can be extended to star and/or mesh. Let us consider two nodes node A and node B. Initially it is assumed that both the nodes are in idle state. Let us have a look at what the transmitter does:

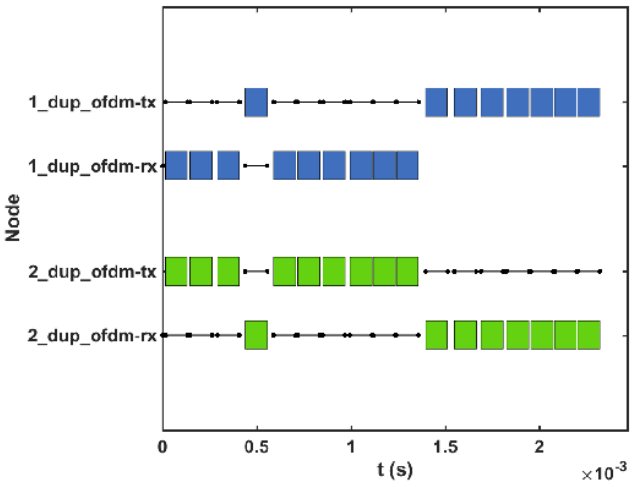
- From the idle state,the next state it enters the sense state. The random back-off timer is set to zero. The signal energy is also reset so that it can detect energy in the next state.
- In the sense state, initially it is going to compare the signal energy (the first 40 samples), with the threshold.
  - If the signal energy exceeds the threshold, it implies that the channel is busy, and thus it cannot transmit at this point of time. Hence it is going to jump back to idle state and from idle state to sense state and the entire process as mentioned above, repeats.
  - If the signal energy is less than the threshold, it implies that the channel is free, thus it can transmit at this point of time. But it does not transmit right away. It waits for a time period, which is called the random back-off period, before it starts sending out the packet.

It also listens to the channel for the entire back-off time period. If during this time period if it senses another transmission taking place, it resets its timer back to zero, and then it jumps back to idle state and from idle state to sense state and the entire process, as mentioned above repeats. If the channel is free for the entire time period it begins transmitting the packet.

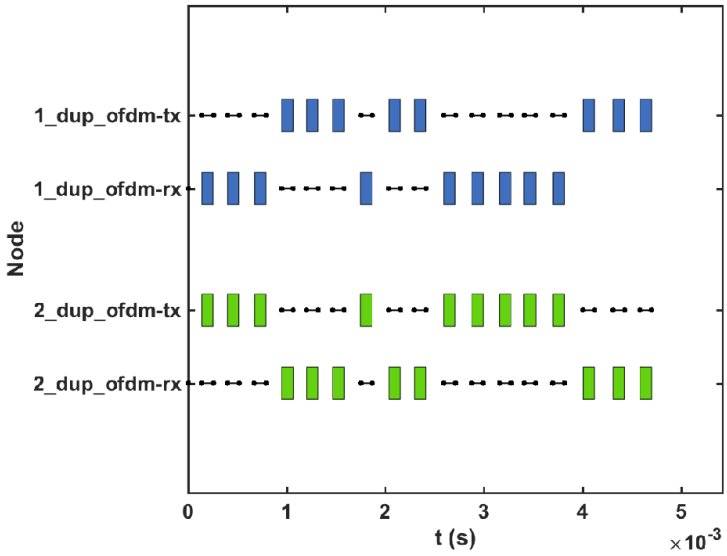
- Once the transmission of one packet is done, the entire chain repeats.

Let us have a look at what the receiver does:

- From the idle state,the next state it enters the sense state.
- In the sense state, initially it is going to compare the received signal energy (the first 40 samples), with the threshold. If the signal energy exceeds the threshold, it implies that the channel has some data. The receiver now receives the frame and starts decoding.
- If the signal energy is less than the threshold, it implies that the channel is free, thus it has no packets to receive. It then goes back to idle state and then to the sense state. The process repeats.
- Once the reception of one packet is done, the entire chain repeats.

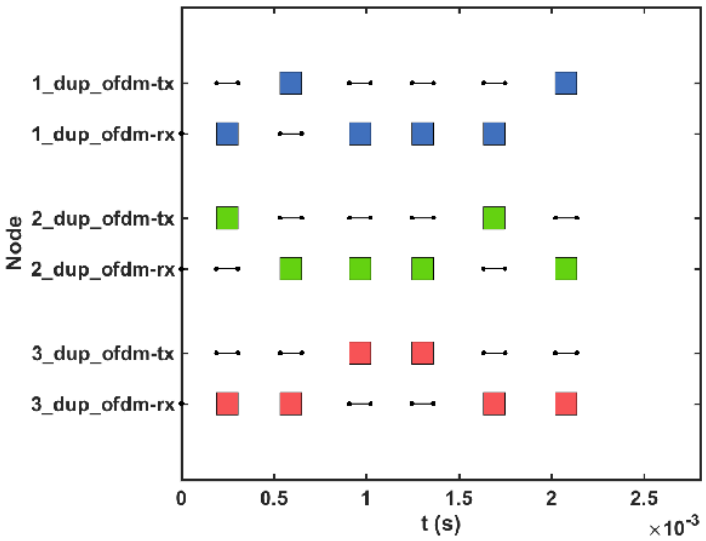


(a) Peer to Peer CSMA I

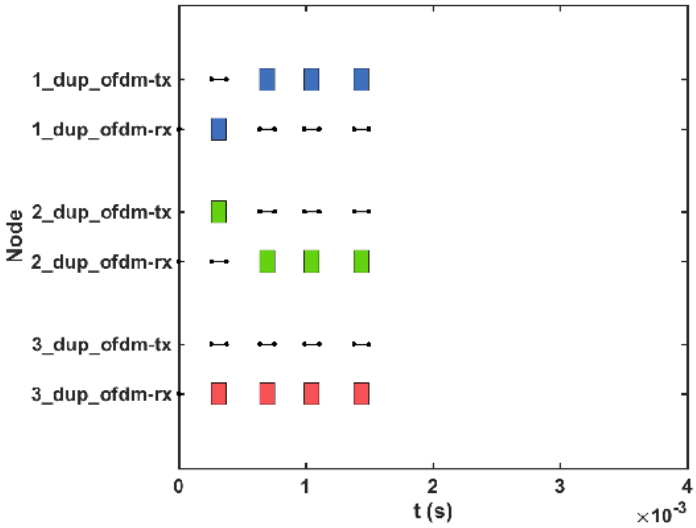


(b) Peer to Peer CSMA II

Figure 12: Peer to peer CSMA

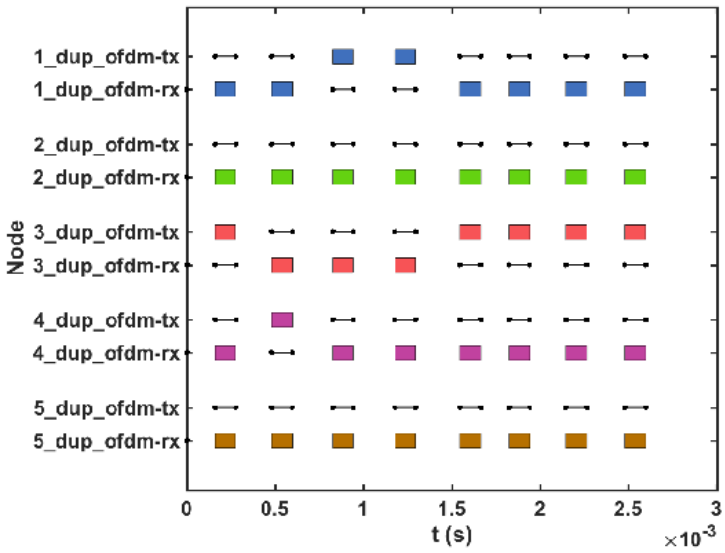


(a) CSMA Mesh I

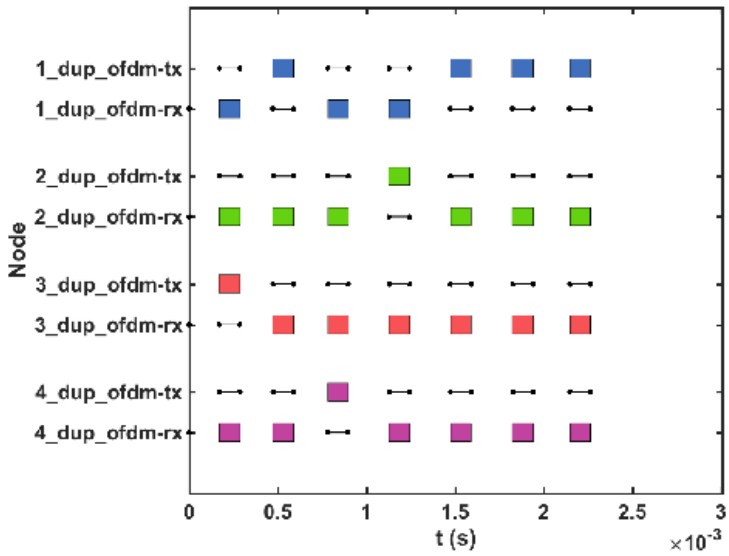


(b) CSMA Mesh II

Figure 13: CSMA - Mesh Topology



(a) CSMA multiple nodes I



(b) CSMA multiple nodes II

Figure 14: CSMA with Multiple nodes

4.5 Major Drawback

One drawback of the above implementation of the CSMA is the hidden node problem. As we can see from the picture given below, two nodes are trying to transmit at the same time. This will result in a collision, or as one can say, packet loss. Let us briefly talk about the hidden node problem and how it can be resolved to implement an error free CSMA mechanism.

4.5.1 Hidden node problem

A key problem in wireless networking is the design of MAC mechanisms that operate in a distributed fashion. Such MACs are needed for ad hoc wireless networks as well as for 'mesh' or 'grid' networks where packets take multiple wireless hops before reaching infrastructure nodes that connect to the wired network. The goal of any such MAC is to negotiate access to the physical channel such that transmissions can occur without collision. In such MACs, a basic problem is the hidden node problem, which occurs when the MAC allows some node to transmit despite that transmission causing a collision [9].

Let us consider a scenario of wireless networking with three wireless devices A, B and C. The transmission range of access point A reaches B, but not at access point C, similarly transmission range of access point C reaches B, but not A. These nodes are known as hidden terminals.

The problem occurs when nodes A and C start to send data packets simultaneously to the access point V. Because the access points A and C are out of range of each other, the result is that they cannot detect a collision while transmitting. CSMA/CD does not work and collisions occur, which corrupt the data received by the access point B due to the hidden node problem.

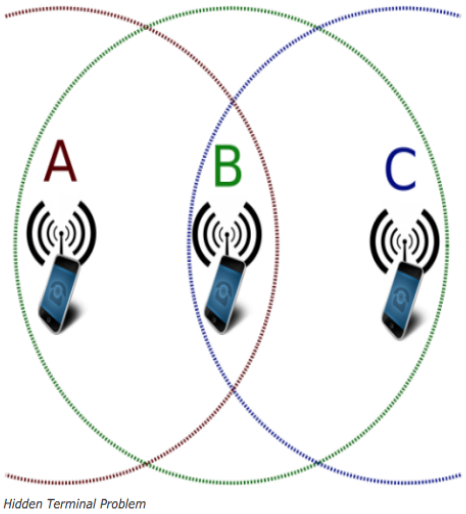
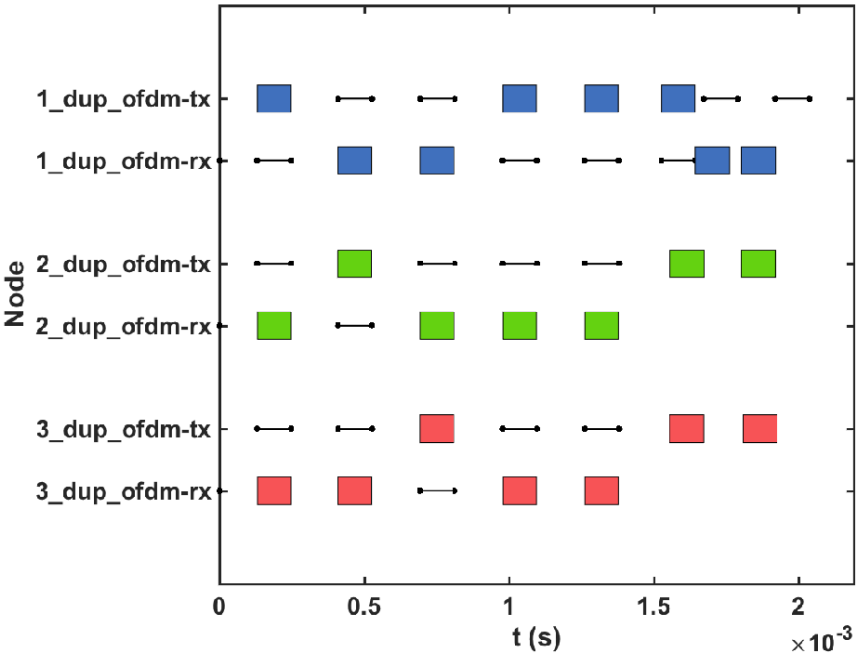


Figure 15: Hidden node Problem

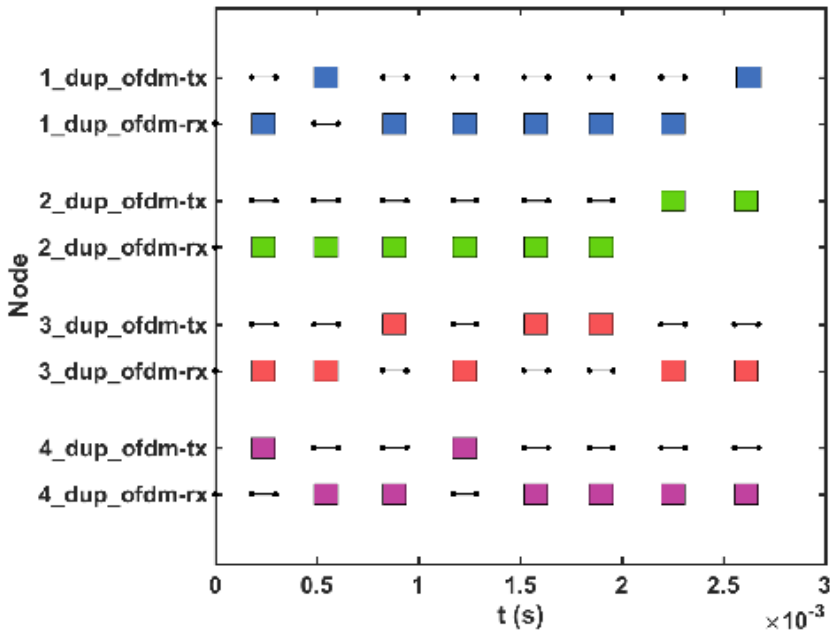
Scenario describing the hidden node problem

The hidden terminal analogy can be described as follows:

- Terminal A sends data to B, terminal C cannot hear A
- Terminal C wants to send data to B, terminal C senses a 'free' medium (CS fails) and starts transmitting
- Collision at B occurs, A cannot detect this collision (CD fails) and continues with it's transmission to B
- Terminal A is 'hidden' from C and vice-versa



(a) Scenario I



(b) Scenario II

Figure 16: Illustration of Collisions

4.5.2 Overcoming Hidden node problem

In wireless networks, interference is location based. Thus, the hidden terminal problem may happen frequently. Resolving hidden terminal problem becomes one of the major design considerations of MAC protocols. IEEE 802.11 DCF is the most popular MAC protocol used in both wireless LANs and Mobile Ad-hoc Networks (MANETs). It's RTS/CTS handshake is mainly designed for such a purpose [10]. RTS/CTS (Request to Send/Clear to Send) is the optional mechanism used by the 802.11 wireless networking protocol to reduce frame collisions. CSMA/CA is supplemented by the exchange of a RTS packet sent by the sender S, and a CTS packet sent by the intended receiver R. Thus alerting all nodes within range of the sender, receiver or both, to not transmit for the duration of the main transmission. The node also awaits receipt of an acknowledgement packet from the Access Point(AP) to indicate the packet was received and checksummed correctly. If such acknowledgement does not arrive in a timely manner, it assumes the packet collided with some other transmission, causing the node to enter a period of binary exponential back-off prior to attempting to re-transmit. It is particularly beneficial in a wireless LAN due to a common problem of multiple stations being able to see the Access Point, but not each other.

However, the RTS/CTS frames can cause a new problem called the exposed terminal problem in which a wireless node that is nearby, but is associated with another access point overhears the exchange and then is signaled to back-off and cease transmitting for the time specified in the RTS.

Chapter 5

AUTOMATIC DEPENDENT SURVEILLANCE - BROADCAST

5.1 Motivation

Automatic Dependent Surveillance Broadcast (ADS-B) is a category of technologies and applications that could fundamentally change the way aircraft are tracked in the National Airspace System (NAS). Instead of relying on costly radar technology, aircraft will broadcast their state vector and other information to ground receivers and other aircraft. ADS-B has the potential to increase capacity, improve efficiency, reduce costs, and improve safety in the NAS. Applications not possible with today's radar technology can be performed with ADS-B [11].

5.2 Introduction

ADS-B is a function on an aircraft or ground vehicle that periodically broadcasts its state vector (horizontal and vertical position, horizontal and vertical velocity) and other information. The broadcast ADS-B message provides surveillance information to other users, principally Air Traffic Control (ATC) and aircraft/vehicle operators. The applications for ADS-B include ATC display of traffic, runway incursion detection and alerting, and Cockpit Display of Traffic Information (CDTI).

It is a technique for obtaining surveillance data. An ADS-B equipped aircraft or airport vehicle determines its own position and velocity (and possibly other information such as planned route) and broadcasts it periodically. Any properly equipped user within range can receive and use this information. ADS-B is known as dependent and cooperative surveillance; dependent because it relies on the availability of onboard navigation and transmission equipment, and cooperative because information can only be obtained from properly equipped aircraft [12].
ADS-B has been touted as the most important technology to transform air traffic control away from radar based systems and to the more reliable satellite systems by the Federal Aviation Administration (FAA). The FAA envisions that the complete ADS-B system will enable pilots to see and avoid weather disturbances, air traffic, and terrain with the most up-to-date flight information, providing an unparalleled level of situational awareness. As a result of this significant overhaul, the FAA anticipates compulsory compliance in ADS-B to be complete by 2020 for the majority of aircraft in the NAS [13].

5.3 ADS-B Technologies

There are a number of technologies which are essential for ADS-B to function both in the air and on the ground. Ground stations will be needed on the ground to send and receive ADS-B information. The surveillance data must be transmitted to ATC facilities for use by controllers and traffic flow managers. In the air, an ADS-B transceiver is necessary for sending and receiving ADS-B data, along with a pilot interface for entering any data and CDTI for viewing the data [11].

5.3.1 ADS-B-Out

ADS-B data exchange can be broken down in to two categories: ADS-B Out and ADS-B In. ADS-B Out refers to an aircraft broadcasting its position and other information. ADS-B Out transmits information about altitude, airspeed, and location derived through Global Positioning System (GPS) from an equipped aircraft to ground stations and to other equipped aircraft in the vicinity. Air traffic controllers use the information to see participating aircraft in real time with the goal of improving traffic management. It may also contain the state vector, and intent information, along with other information relating to the source and accuracy of the data. The FAA has mandated that aircraft operating in airspace must be equipped with ADS-B Out by Jan. 1, 2020.

5.3.2 ADS-B-In

ADS-B In is the ability to receive information via an ADS-B transceiver. It is not part of the mandate and requires additional equipment, allows participating aircraft to receive traffic and weather information from ADS-B ground stations and nearby aircraft broadcasting their positions through ADS-B Out. This information can be displayed in the cockpit to improve situational awareness.

5.3.3 ADS-B Protocols

As, ADS-B is a digital radio data link technology, there must be a standard protocol for encoding and decoding the data. In the US there are two proposed data link protocols, 1090 MHz Extended Squitter (1090-ES) and Universal Access Transceiver (UAT). This chapter will focus on the 1090-ES as it was the one which was considered for implementation.

5.3.3.1 1090 MHz Extended Squitter (1090-ES)

1090-ES is an ADS-B protocol based on the Mode S transponder. By definition, the word *squitter* refers to a periodic burst or broadcast of aircraft-tracking data that is transmitted periodically by a Mode S transponder. Under the current Mode S setup, a standard transponder squit only sends the most basic aircraft identification, system status and pressure altitude information which ATC's ground computers must correlate with radar tracking information to derive aircraft position, direction of flight, airborne velocity, vertical climb/descent, and so on. The ES format is capable of carrying much more data than the basic *short squit* Mode S version. Around, 49 individual parameters can be sent over the extended squitter. 1090-ES suggests that it transmits and receives on 1090 MHz.
The ES provides five types of reports:

- Airborne Position
- Surface Position
- Aircraft identification and emitter category; and
- Event-driven [14].

Table 2: Mode ES Properties

Property	**Specification**
Transmit Frequency	1090 MHz
Modulation	Pulse Position Modulation
Data Rate	1 Mbit/s
Extended Squitter Length	112 microseconds

5.3.4 ADS-B Packet Structure

An ADS-B message is 112 bits long, and consist of 5 parts. All ADS-B messages start with the Downlink Format (DF) 17 (10001 in binary code) for the first 5 bits. Bits 6-8 are used as additional identifier, which has different meanings within different types of ADS-B message. The preamble, of 8μs long, is used to synchronize the transmitters and receivers, it consists of four pulses with a length of 0.5μs per pulse, with interspaces (to the first pulse) of 1, 3.5 and 4.5μs respectively.

Table 3: Key bits of a message

nBits	**Bits**	**Abbr.**	**Name**
5	1-5	DF	Downlink Format
3	6-8	CA	Capability
24	9-32	ICAO	ICAO aircraft address
56	33-88	DATA	Data
5	[33-37]	[TC]	Type code
24	89-112	PI	Parity/Interrogator ID

5.3.4.1 ADS-B Message Types

To identify what information is contained in a ADS-B message, we need to take a look at the Type Code (TC) of the message, indicated by bits 33 - 37 of the ADS-B message (or first 5 bits of the DATA segment)

Table 4: Type Code and information contained

TC	**Content**
1-4	Aircraft identification
5-8	Surface position
9-18	Airborne position (w/ Baro Altitude)
19	Airborne velocities
20-22	Airborne position (w/ GNSS Height)
23-31	Reserved for other uses

5.3.4.2 ADS-B Checksum

ADS-B uses cyclic redundancy check to validate the correctness of received message, where the last 24 bits are the parity bits.

5.3.5 Application:Aircraft Identification

For the simulation of ADS-B we considered a scenario where an aircraft is sending information, in this case, it's the flight number to the ground station. The data block is coded with the **Pulse Position Modulation (PPM)**. It defines two symbols. Each symbol has two chips, where one has a high value and the other has a low value.

If the first chip is high followed by low chip, this corresponds to the symbol being a 1. Alternatively, if the first chip is low followed by high chip, then the symbol is 0.

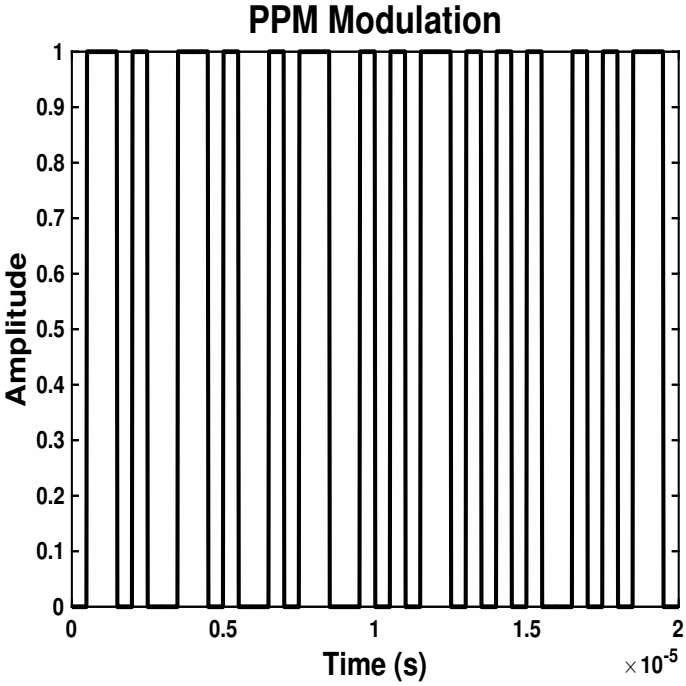


Figure 17: Pulse Position Modulation

The pulse position modulated signal

An aircraft identification message has DF: 17, and TC: 1 to 4, the 56-bit DATA field. An ADS-B packet was formed by generating bits randomly in MATLAB. The generated bits were modulated using PPM. Parity bits were generated and appended at the end of the packet. After all this, they were sent over the air and the decoding was done in the way described below:

For decoding characters, a lookup table is needed for mapping numbers to characters. It is defined as follows, where the # is not used, and _ represents a separation.

```
#ABCDEFGHIJKLMNOPQRSTUVWXYZ#####_###############0123456789######
```

In summary, characters and there decimal representations are:

A - Z : 1 - 26

0 - 9 : 48 - 57

_ : 32

Using the table given above, the aircraft number was successfully decoded without any errors.

```
flightID =

X7K

flightID =

X7K

flightID =

S33MN

flightID =

S33MN

flightID =

DS1

flightID =

DS1

flightID =

QBX

flightID =

QBX

All receivers turned off and simulation is ending.
>>
```

Figure 18: Decoding of Flight IDs

Decoded Flight IDs after simulation

Chapter 6

LTE : SYNCHRONIZATION AND CODE RE-USE

6.1 About LTE

LTE (Long Term Evolution) is the project name given to development of a high performance air interface for cellular mobile communication systems. It is the last step towards the 4th generation (4G) of radio technologies designed to increase the capacity and speed of mobile telephone networks. It is an evolution of the Universal Mobile Telecommunications Service (UMTS) / 3rd Generation Partnership Project (3GPP) 3rd Generation (3G) standards. LTE uses a different form of radio interface Orthogonal Frequency Division Multiple Access/Single Carrier Frequency Division Multiple Access (OFDMA / SC-FDMA) instead of Code Division Multiple Access (CDMA), but there are many similarities with the earlier forms of 3G architecture and opportunities for re-use of some elements of 3G network architecture. LTE can be seen as providing an evolution of functionality, increased speeds and general improved performance compared to 3G. LTE has introduced a number of new technologies when compared to previous cellular systems. They enable LTE to operate more efficiently with respect to the use of spectrum, and also provide the much higher data rates that are now required.

The motivation for LTE:

- Need to ensure the continuity of competitiveness of the 3G system for the future
- User demand for higher data rates and quality of service
- High spectral efficiency, high peak data rates, short round trip time as well as flexibility in frequency and bandwidth.

6.2 LTE Downlink Synchronization

In LTE, there are two downlink synchronization signals which are used by the User Equipment (UE) to obtain the cell identity and frame timing.

- Primary synchronization signal (PSS)
- Secondary synchronization signal (SSS)

The PSS is linked to the cell identity within the group. The SSS is linked to the cell identity group and the cell identity within the group. One can obtain the cell identity within the group by successfully demodulating the PSS. The SSS can then be demodulated and combined with knowledge of the cell identity within the group to obtain the cell identity group Once you establish both the above stated values, you can determine the cell identity.

6.2.1 Primary synchronization signal (PSS)

The primary synchronization signal (PSS) is based on a frequency-domain Zadoff-Chu sequence. Zadoff-Chu sequences are a construction of Frank-Zadoff sequences defined by D. C. Chu in [15]. These codes have the useful property of having zero cyclic

autocorrelation at all nonzero lags. When used as a synchronization code, the correlation between the ideal sequence and a received sequence is greatest when the lag is zero. When there is any lag between the two sequences, the correlation is zero. The PSS is mapped into the first 31 subcarriers either side of the DC subcarrier. Therefore, the PSS uses six resource blocks with five reserved subcarriers each side. As the DC subcarrier contains no information in LTE this corresponds to mapping onto the middle 62 subcarriers within an OFDM symbol in a resource grid. The PSS is mapped to different OFDM symbols depending on which frame type is used. Frame type 1 is Frequency Division Duplex (FDD), and frame type 2 is Time Division Duplex (TDD). If using FDD, the PSS is mapped to the last OFDM symbol in slots 0 and 10. If using TDD, the PSS is mapped to the third OFDM symbol in subframes 1 and 6.

6.2.2 Secondary synchronization signal (SSS)

SSS is based on maximum length sequences (m-sequences). An m-sequence is a pseudorandom binary sequence which can be created by cycling through every possible state of a shift register of length resulting in a sequence of length. Three m-sequences, each of length 31, are used to generate the synchronization signals. Two binary sequences, each of length 31, are used to generate the SSS [16]. The two sequences are scrambled with a binary scrambling code. The second SSS sequence used in each radio frame is scrambled with a binary scrambling code corresponding to the cyclic shift value of the first sequence transmitted in the radio frame.
The scrambled sequences are interleaved to alternate the sequence transmitted in the first and second SSS transmission in each radio frame that allows the receiver to

determine the frame timing from observing only one of the two sequences; if the first SSS signal observed is in subframe 0 or subframe 5, synchronization can be achieved when the SSS signal is observed in subframe 0 or subframe 5 of the next frame. The SSS is transmitted in the same subframe as the PSS but one OFDM symbol earlier. The SSS is mapped to the same subcarriers (middle 72 subcarriers) as the PSS. The SSS is constructed using different scrambling sequences when mapped to even and odd resource elements.

6.3 Code Reuse

This part of my research work, tested my object-oriented coding skills. I integrated "LTE SCFDMA" and "LTE OFDMA" radios. I architected and implemented a code reuse solution. In our code base there was one common SCFDMA transmitter/receiver node, and one OFDMA transmitter/receiver node. Our goal was to use the previously mentioned nodes for various configurations, such as base station-base station, mobile-mobile, base station-mobile, and mobile-base station communication, without creating the nodes again and again for each configuration. We observed that, by making use of code reuse, lots of computational complexity was avoided, thus making our simulations faster.

In our simulation package, the synchronization module was not included. Hence, by making use of the code written by my colleague, the synchronization module was integrated into our LTE simulation package.

Once the modules (PSS, SSS generation) were integrated, the delay was obtained as a function of frame number. Thus, the receive data had to be synchronized accordingly, and only then the next step in the receiving chain was carried out. After the synchronization module was incorporated, we observed that the BER was almost at 0.

Chapter 7

FUTURE WORK

Right now, the PDK considers the most minimal of the system requirements and specifications. Additional features can be added to make it more reliable, precise. The below mentioned are a few system functionalities that we intend to develop and implement in the near future.

- On the GUI side:
  - Displaying BER/PER curves
  - Extensive usage of MATLAB spectrum analysis tools to further aid the user in visualization
  - Automated FCC report generation
  - Quick documentation - software block diagram, and interface specifications to the level of data being passed
- On the WISCAcomm side:
  - Network level Simulations to back MAC layer choices
  - Implementation of Power Control and Rate Adaption on radios
  - Equalization, and Signal acquisition on radios
  - Unit-level and System-level testing for product robustness

BIBLIOGRAPHY

[1] Antonios S. Andreatos, and Anastasios D. Zagorianos, "Matlab GUI Application for Teaching Control Systems," 2009 6th WSEAS International Conference on Engineering Education, Greece, 2009.

[2] A. Goldsmith. *Wireless Communications*, 1st ed. Cambridge University Press, 2005.

[3] T. S. Rappaport. *Wireless Communications: Principles and Practice*, 2nd ed. Prentice Hall, 2002.

[4] Govind Sati, and Sonika Singh, "A Review On Outdoor Propogation Models In Radio Communication," *International Journal of Computer Engineering & Science*, March 2014.

[5] Kwok- Wai Cheung, Jonathan H.-M. Sau, and R. D. Murch, "A New Emperical Model for Indoor Propogation Prediction," *IEEE Transactions on Vehicular Technology*, vol. 47, no. 3, August 1998.

[6] Gregory D. Durgin, Theodore S. Rappaport, "Theory of Multipath Shape Factors for Small-Scale Fading Wireless Channels," *IEEE Transactions on Antennas and Propogation*, vol. 48, no. 5, May 2000.

[7] European Telecommunications Standards Institute (2016). *Universal Mobile Telecommunications System (UMTS); Spatial channel model for Multiple Input Multiple Output (MIMO) simulations*. 3GPP TR 25.996 version 13.0.0 Release 13. Retrieved from http://www.etsi.org/standards-search

[8] D. W. Bliss and S. Govindasamy. *Adaptive Wireless Communications: MIMO Channels and Networks*, Cambridge University Press, Cambridge, 2013.

[9] Yihong Zhou, and Scott M. Nettles, "Balancing the Hidden and Exposed Node Problems With Power Control In CSMA/CA-Based Wireless Networks," 2005 IEEE Wireless Communications and Networking Conference, Los Angeles, 2005.

[10] Kaixin Xu, M. Gerla, and Sang Bae, "How effective is the IEEE 802.11 RTS/CTS handshake in ad hoc networks," 2002 IEEE Global Telecommunications Conference, Taiwan, 2002.

[11] Lester, E.A. (2005). *Benefits and Incentives for ADS-B Equipage in the National Airspace System* (Thesis dissertation). Retrieved from https://dspace.mit.edu/handle/1721.1/38468

[12] Daniel S. Hicok, and D. Lee, "Application of ADS-B for airport surface surveillance," 1998 Digital Avionics Systems Conference , Washington, 1998.

[13] Brandon Stark, Brennan Stevenson, and YangQuan Chen, "ADS-B for Small Unmanned Aerial Systems: Case Study and Regulatory Practices," 2013 International Conference on Unmanned Aircraft Systems, Georgia, 2013.

[14] Lima Peru, "Guide On Technical and Operational Considerations for the Implementation of ADS-B in the SAM Region," ver. 1.2, May 2013.

[15] Chu, D. C. "Polyphase Codes with Good Periodic Correlation Properties." IEEE Trans. Inf. Theory. Vol. 18, Number 4, July 1972, pp. 531-532.

[16] European Telecommunications Standards Institute (2008). *LTE; Evolved Universal Terrestrial Radio Access (E- UTRA); Physical channels and modulation.* 3GPP TS 36.211 version 8.4.0 Release 8. Retrieved from http://www.etsi.org

APPENDIX A

ALGORITHM FOLLOWED IN ARBITER

As radio protocols proliferate, it is preferable to have a simulation environment that supports a wide variety of protocols and scenarios. To meet this need, the following simulation architecture was used:

Data: user input
initialization;
while *true* **do**
  for $n \leftarrow 1$ **to** $nNodes$ **do**
    `// Loop over transmitters`
    TxStep(n);
    event = GetTransmitterState(n);
    txEvents(n) = [txEvents(n) event];
  end
  for $k \leftarrow 1$ **to** $nNodes$ **do**
    `// Loop over receivers`
    request = rxRequests(n);
    if *request == 'receive'* **then**
      for $n \leftarrow 1$ **to** $nNodes$ **do**
        `// Get relevant txEvents`
        events = txEvents(n);
        - Identify where events and request overlap in time
        - Process these events over the wireless channel between nodes k and n
        - Add noise
      end
    end
  end
end

Algorithm 1: How to write algorithms

- The PHY layer class methods follows:
  - ClassName < BaseNode % constructor
  - GenerateFrame
  - ReceiveFrame

* A common class base class is used for all PHY layer implementations (class).
  - BaseNode
  - SetTx
  - SetRx
  - GetTransmitterState
  - GetReceiverState

- MAC Layer The MAC layer class methods follows:
  - ClassName
  - TxStep
  - RxStep

In brief, when Arbiter calls TxStep, the MAC transmit state machine is updated, and calls are made to the supporting PHY layer to complete what is needed in the present state. Similarly, when Arbiter calls RxStep, a state machine is updated, and PHY layer calls are made as needed.