University of Indonesia

Performance comparison of channel estimation method for multiple antenna system in wireless communication

UNDERGRADUATE THESIS

An Undergraduate thesis submitted for the requirements of

Bachelor Engineering Degree

Fajar Adi Prabowo

0606142616

Faculty of Engineering

Electrical Engineering Department

Depok

January 2012

STATEMENT OF AUTHORSHIP

This thesis is the result of my own work, and all sources quoted or referenced have been stated correctly as they should be

Name: Fajar Adi Prabowo

Student Number: 0606142616

Sign:

Date: January 2012

APPROVAL PAGE

This thesis which is submitted by:

Name: Fajar Adi Prabowo

Student Number: 0606142616

Course: Bachelor of Engineering (Electrical)

Thesis title: Performance Comparison of Channel Estimation Method for Multiple Antenna System in Wireless communication

Has been successfully defended in front of the Board of Examiners and accepted as part of the requirements necessary to obtain an Engineering Degree in Electrical Engineering, Faculty of Engineering, University of Indonesia

BOARD OF EXAMINERS

Supervisor: Ir. Gunawan Wibisono, M.Sc., Ph.D (..................................)

Examiner: Dr. Muhammad Suryanegara, ST, M.Sc (..................................)

Examiner: Dr. Fitri Yuli Zulkifli, ST. M.Sc (..................................)

Approved in: Depok

Date: January 2012

ACKNOWLEDGEMENTS

I am greatly thankful to my supervisor Dr. Karla Ziri-Castro and Ir. Gunawan Wibisono who kindly also constantly guiding and supporting me from the initial of this undergraduate theories into the final stages which eventually led me to develop my skill and gain a better understanding in this research

Lastly, I would say my gratitude and blessing to my family and people who always supports me in any conditions. Furthermore, I do have a lot of respect for people that helped me in completing this thesis that I could not mention one by one.

I declare that the work presented in this research is my own work except where due reference or acknowledgement is given to the work of others.

Depok, January 2012

Fajar Adi Prabowo

DECLARATION PAGE
OF
THESIS PUBLICATION FOR ACADEMIC PURPOSES

As a student of University of Indonesia, I the undersigned below:

Name: Fajar Adi Prabowo

Student Number: 0606142616

Course: Bachelor of Engineering

Department: Electrical Engineering

Faculty: Engineering

Thesis type: Undergraduate Thesis

For the development of science, agreed to provide University of Indonesia a **Non-exclusive royalty – Free Right** on my scientific work, entitled:

Performance Comparison of Channel Estimation Method for Multiple Antenna System in Wireless Communication

Along with existing devices (if needed). **Non-exclusive royalty-Free Right** allows University of Indonesia to keep, convert/format, manage in the form of a database, and publish my theories as long as putting my name as author/creator and copyright owner.

I do solemnly and sincerely declare this statement and I make this solemn declaration conscientiously believing the same to be true.

Made in: Depok

Date: January 2012

Declarer,

(Fajar Adi Prabowo)

ABSTRACT
Name: Fajar Adi Prabowo
Course: Bachelor of Engineering (Electrical)
Thesis title: Performance Comparison of Channel Estimation Methods for Multiple Antenna System in Wireless Communication

Channel estimation has a definition to estimate the filter coefficient through received signal and other known information. There are many methods that channel estimation can be used such as minimum mean square error, least square, zero-force, maximum likelihood. In this research, least square and zero force will be discussed further. These two types are commonly known as the simplest and easiest type to implement in the wireless communication. Even though other methods like minimum mean square error or maximum likelihood have been widely known to estimate channel, such methods are proven very complex to implement. The aimed of this research is to understand the effect and compare the performance of the channel estimation method with the capacity of multiple antennas such as Single-Input Single-Output (SISO), Multiple-Input Single-Output (MISO), Single-Input Multiple-Output (SIMO) and Multiple-Input Multiple-Output (MIMO).

Least square and zero force estimation method were being used with BPSK modulation, this is due to BPSK is the simplest phase shift keying. First, the parameters need to be configured in simulation software. Then, the data within pilots system transmitted to receiver using BPSK modulation in each of the wireless antenna systems with different wireless channels. Space time block coding of Alamouti encoder were executed for MIMO and MISO system. After that, least square and zero force method were implemented and bit error rate (BER) performance can be achieved by equalizing the transmitted signal. Analysing the result by comparing the performance of least square and zero force were accomplished. In the recent result, it was shown that the least square performs better than zero force estimation method for the performance of bit error rate with signal to noise ratio.

Keywords

Wireless communication, least square, zero force, channel estimation, multiple antennas.

ABSTRACT
Name: Fajar Adi Prabowo
Program studi: Teknik Elektro Internasional
Judul: Perbandingan performa metode channel estimation pada sistem multi antenna di komunikasi nirkabel

Channel estimation memiliki definisi untuk memperkirakan koefisien filter melalui sinyal yang diterima dan informasi lain yang dikenal. Ada beberapa metode channel estimation yang dapat diguanakan seperti minimum mean square error, least square, zero-force, maximum likelihood. Dalam penelitian ini, least square dan zero-force akan dibahas lebih lanjut. Kedua jenis ini dikenal umum sebagai jenis yang paling sederhana dan paling mudah untuk diterapkan di dalam komunikasi nirkabel. Meskipun metode lain seperti minimum mean square error atau maximum likelihood telah dikenal secara luas untuk memperkirakan kanal, metode tersebut terbukti sangat kompleks untuk diimplementasikan. Penelitian ini bertujuan untuk memahami efek dan membandingkan kinerja channel estimation dengan menggunakan kapasitas multi antenna seperti Single-Input Single Output (SISO), Multiple-Input Single-Output (MISO), Single-Input Multiple-Output (SIMO) dan Multiple-Input Multiple-Output (MIMO).

Least square dan metode estimasi zero-force telah digunakan dengan memakai modulasi Binary Phase Shift Keying (BPSK), hal ini dikarenakan BPSK adalah metode phase shift keying yang paling sederhana untuk di implementasikan di dalam penelitian ini. Pertama-tama, segala parameter perlu dikonfigurasi terlebih dahulu untuk digunakan di dalam simulasi. Kemudian, data dalam sistem pilot dikirim ke penerima menggunakan modulasi BPSK di setiap sistem multi antenna dengan menggunakan saluran nirkabel yang berbeda. Lalu, Alamouti encoder dieksekusi khusus untuk sistem MIMO dan MISO. Setelah itu, least square dan zero-force diterapkan dan tingkat kesalahan bit atau dengan nama lain bit error rate (BER) dapat dicapai dengan menyamakan sinyal yang ditransmisikan. Kemudian, analisa hasil data dengan membandingkan kinerja least square dan zero-force dapat dicapai. Pada akhirnya, hasil yang telah ditemukan menunjukkan bahwa least square melakukan performa lebih baik daripada metode estimasi zero force yang berlaku pada kinerja tingkat kesalahan

bit dengan sinyal terhadap noise atau dengan nama lain signal to noise ratio (SNR)

Keywords

Komunikasi nirkabel, least square, zero force, channel estimation, multi antenna.

TABLE OF CONTENTS

TABLE OF FIGURES

CHAPTER I

INTRODUCTION

I.1. Background

Telecommunication is one of the basic important things in human life. As today, telecommunication plays the important roles of any human activities. In fact, it cannot be avoid that one of the basic thing needed by mankind is communication. There are different kinds of communication that used by people such as wireless communication or wire lined communication. Wireless telecommunications is the transfer information between two or more points that are physically not connected. Distances can be short, as a few meters as in television remote control; or long ranging from thousands to millions of kilometres for deep-space radio communications.

Wireless communication uses radio channel to transmit data or the signal. The radio channels in mobile radio systems are usually multipath fading channels, which are causing inter symbol interference (ISI) in the received signal. To remove ISI from signal, usually equalizer is used. The equalizer is using the estimated channel to get the received signal from the transmitter. This method is called channel estimation. By implementing channel estimation method, the signal that had been transmitted can be configured and the desired signal can be obtained by combining the equalizer with the transmitted signal.

Channel estimation or equalization is the process of reducing amplitude, frequency and phase distortion in a radio channel with the intent of improving transmission performance [1]. Before doing the channel estimation, first the radio channel used the modulation to modulate the signal in radio space. Then the receiver will demodulate the signal that had been transmitted before. However, when the receiver demodulate the signal by doing normal transmission, the outcomes will get an unclear signal because of some interference that has occurred in radio space such as climate or maybe object that had blocked the signal. Therefore, the signal need to be equalize using the channel estimation method as to reduce the inter symbol interference, hence would get maximum probability of correct decisions. This channel estimation will estimate the signal that had been loss caused of some interference in the way of the signal transmitted.

Least square and zero force are one of many methods for channel estimation. Those other methods can be like least mean square, minimum mean square error, and maximum likelihood. The least square and zero force were expected to be the simplest methods rather than minimum mean square error or any other methods. This is due to least square and zero force is less computational also less complex to implement than any other types. Least square can be found by minimizing the following squared quantity, while zero force inverts the folded frequency response of the channel. These two types have been used for the purpose to understand the basic algorithm of channel estimation in wireless communication.

This research will discussed about the least square and zero force estimation method in multiple antenna system for distribution like Gaussian, Rayleigh, and Rician channel. The least square and zero force were expected to be the simplest methods rather than minimum mean square error or any other methods. Furthermore, it also includes the research, analysis and MATLAB simulation that have been done about the channel estimation and comparing their performance. Space time block coding of Alamouti encoder implemented in this wireless channel for transmitting the data especially for MISO and MIMO. Pilot symbols also being used for the purpose of enabling measurement of the channel at the receiver. The processes include the capacity multiple antenna system occurred in terms of bit error rate and signal to noise ratio. MATLAB simulations have been done to look up the effect of channel estimation in multiple antennas system also the distribution channel.

I.2. Objectives and Scopes

The goal of this research is to compare the methods of channel estimation for wireless systems like SISO, SIMO, MISO and MIMO also for different type of channel distribution such as Rayleigh, Rician and Gaussian to assess the optimum channel estimation method for each wireless system.

The objectives will contain:

- Simulate the channel estimation method in wireless communication systems
- Comparing the performance of least square and zero force estimation method for multiple antenna systems
- Implement channel estimation methods
- Assess performance for each multiple antenna system (MIMO, MISO, SIMO and SISO) considering different channels such as Rayleigh, Rician, Gaussian

As the scopes will include:

- Simulation platform for multiple antenna systems such as MIMO, MISO, SIMO and SISO
- Performing BPSK modulation
- Implement Alamouti encoder to transmit multiple data for multiple antenna system such as MISO and MIMO
- Analyse and compare the performance of least square and zero force channel estimation in wireless antenna systems
- Performing wireless communication system using channels like Gaussian, Rayleigh and Rician including each of the parameters.

CHAPTER II

WIRELESS COMMUNICATION SYSTEM

II.1. Multiple Antenna Systems

In communication systems, antennas consist of different type such as SISO, SIMO, MISO and MIMO. Therefore, this research will describe four of wireless antenna which are Single-Input Single-Output (SISO), Multiple-Input Single-Output (MISO), Single-Input Multiple-Output (SIMO), and Multiple-Input Multiple-Output (MIMO).

II.1.1. Single-Input Single-Output (SISO)

Single-Input Single-Output which we called SISO is the most basic radio channel access mode and uses single antenna at transmitter and single antenna at the receiver. Both the transmitter and the receiver have one RF chain that means coder and modulator. SISO is relatively simple and cheap to implement also it has been used age long since the birth of radio technology. It is mostly used in radio and TV broadcast and our personal wireless technologies such as Wi-Fi and Bluetooth. The system of SISO can be looked up at figure (1).

Figure 1: SISO system

SISO system as explained in the figure (1) explained that it only use both single in transmitter and receiver which employs no diversity technique. SISO systems are sometimes troubled by multipath effects. Electromagnetic wavefronts are dispersed when they encounter signal-path obstructions like buildings, hills, tunnels, valleys and utility wires. In such cases, the scattered electromagnetic waves take many paths to reach the destination. That causing problems like cut-out (cliff effect), fading, and intermittent reception (picket fencing). In a digital communications system, it can cause a reduction in data speed and an increase in the number of errors.

In typical communication system, transmitter, receiver and channels can be shown with mathematical form of

$$y = h * t + n \qquad (2.1)$$

Where y is the received signal and in this system it only used one receive antenna signal. h is the matrix channel impulse response which this matrix has n_t (columns and n_r (number of received signal) rows. In this case of SISO channel the number of received and transmitted is one. t is the n_t dimensional transmitted signal. The notation n is the complex n_r dimensional of noise which called AWGN. And the notation of * means convolution. This is time discrete system model where noise is drawn from a Gaussian distribution of zero mean and variance [2]. The receiver receives the transmitted symbol perfectly if the noise variance is zero. However, if the noise variance is not zero and there is no constraint on the input, the infinite subset of inputs can be choose arbitrarily far apart, so it will distinguishable at the output with arbitrarily small probability of the error [2]. Therefore, the capacity of the channel is infinite if the noise variance is zero or the input is unconstrained. By using theoretic point of view the SISO channel capacity can be obtained. The capacity of Gaussian channel with power constraint P is

$$C = \max_{p(x):EX^2 \leq P} I(X;Y) \qquad (2.2)$$

By expanding *I(X;Y)* the information capacity can be calculated. Then, it will obtained the capacity formula for information capacity of Gaussian channel as

$$C = \max_{EX^2 \leq P} I(X;Y) = \frac{1}{2} log\left(1 + \frac{P}{N}\right) \quad (2.3)$$

Then, it is known that the mean capacity of SISO system with a random complex channel gain h_{11} is given as

$$C = E_H[log_2(1 + \rho.|h_{11}|^2)] \quad (2.4)$$

Where ρ is the average signal to noise ratio (SNR) at the receiver. So if $|h_{11}|^2$ is Rayleigh, $|h_{11}|^2$ follows a chi squared distribution with two degrees of freedom. Therefore, the equation above can be written like

$$C = E_H[log_2(1 + \rho.\chi_2^2)] \quad (2.5)$$

Where χ_2^2 is a chi-square distributed random variable with two degrees of freedom.

II.1.2. Multiple-Input Single-Output (MISO)

MISO which stands for Multiple-Input Single-Output is an antenna technology for wireless communications in which multiple antennas are used at the transmitter. The antennas combined which the purpose to optimize the data speed and minimize the error [3]. The receiver for this system has only one antenna. This type of system is one kind forms of smart antenna technology which other kinds are MIMO and SIMO. This MISO system is used to prevent the problems with multipath effects that occurred in SISO system or in other words it is used for improving signal robustness. For example, when the transmitter transmit signal or electromagnetic field and encountered with obstacle like buildings, hills and others, the waves are spread out and it will take many paths and reflection to reach the receiver antenna [3]. This will be problem for digital communication system like wireless internet which caused a reduction in data speed and an increase in the number of errors. This type of problem can be overcome with using two antennas or more at the transmitter which is MISO system along with the transmission of multiple signals at the transmitter that reduce the trouble caused of multipath wave propagation [3]. The application for this system is digital television (DTV), wireless local area networks (WLANs), metropolitan area networks (MANs), and mobile communications.

The process of MISO system can be explained by looking at the figure (2).

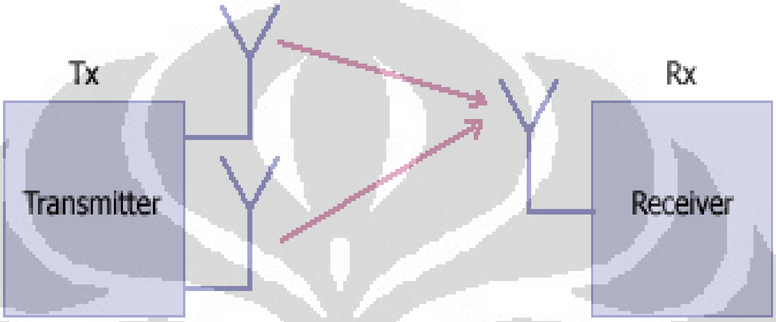

Figure 2: MISO system

This system employs a transmit diversity technique for improving the reliability of a message signal using two or more antennas at the transmitter with different characteristic. Technique of Alamouti Space Time Coding (STC) is employed at the transmitter with two antennas [4]. The transmitter uses the STC to allow transmit signal both in time and space, meaning the information is transmitted by two or more antennas at their different times consecutively.

The way of doing the channel model for MISO is by using the n_t as transmitter antennas at one end and a single receiver antenna at the other end. Therefore, the MISO channel model can be shown using mathematical form such as

$$y = hx + v \tag{2.6}$$

Equation above referred as the discrete-time MISO channel [5]. Where v is the Gaussian noise and the channel matrix h is the 1 x n_t matrix shown as

$$h = [h_1, h_2, \dots\dots \ , h_{nt} \tag{2.7}$$

and

$$x = [x_1, x_2, \dots\dots \ , x_{nt}]^T \quad (2.8)$$

The constant gain of the channel between the multiple transmitter antennas and the single receiver antenna over a symbol period represent by element h_i, for i = 1, 2, ……, n_t [5].

Therefore the channel capacity of the discrete time MISO channel model can be obtained

$$C = log_2 \left(1 + \frac{\rho}{n_T} \sum_{i=1}^{n_T} |hi|\right) \quad (2.9)$$

(bps/Hz)

The constant gain is notated same as equation before which is h_i where *ith* transmitter antenna, I = 1, 2, ……, n_T, and by using the number of n_T transmitter antennas the total transmitter power is normalized [5]. It is known that MISO does not have receiver diversity, only transmitter diversity. Moreover, it will show that increasing the number of n_T antennas at the transmitter end also results in a logarithmic increase relationship of n_T in average capacity [5].

In addition, this transmission strategy maximizes the received SNR by having the received signals from the various transmit antennas add up in-phase (coherently) and by allocating more power to the transmit antenna with the better gain [6]. This strategy, "aligning the transmit signal in the direction of the transmit antenna array pattern", is called transmit beamforming [6]. Using this beamforming, the MISO channel is changed into scalar AWGN channel and other any code which make optimal for the AWGN channel can be used directly.

II.1.3. Single-Input Multiple-Output (SIMO)

Single-Input Multiple-Output is an antenna technology in wireless communication which consists of one antenna at the transmitter and two or more antennas at the receiver. The purpose for this system is similar like MISO systems, the antennas are combined to minimize the error and optimizing the data speed. However for this type of system it used receiver diversity rather than transmitter diversity for MISO type. It can also overcome the problem of multipath effect in SISO. It almost same as MISO, for example when electromagnetic field or signal transmit from the transmitter and these signal encountered with obstacles like buildings or others, the signal will be scattered as having reflection from the obstacles and they take many path to reach to the receiver. These signal will causing problems such as fading or cut off. In wireless communication, it will have some reduction of the data and the number of errors will increased. Which in this case, SIMO can minimize the problem caused by multipath wave propagation with using two or more antennas at the receiver. The SIMO technology has widespread applications in digital television (DTV), wireless local area networks (WLANs), metropolitan area networks (MANs), and mobile communications [7]. Diversity reception which is known as an early form of SIMO has been used by military, commercial, amateur, and shortwave radio operators at frequencies below 30 MHz since the First World War [7].

Figure (3) expressed the process of SIMO antenna technique

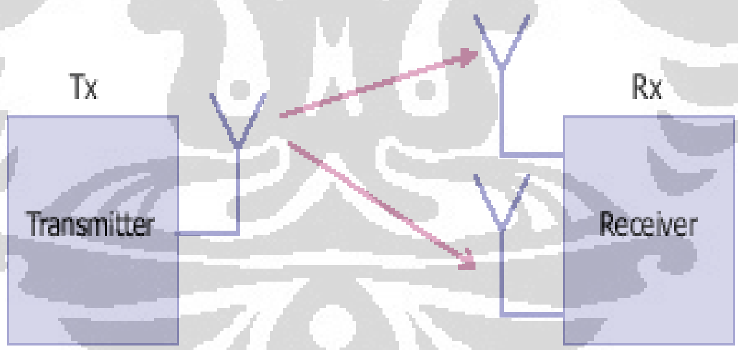

Figure 3: SIMO system

It employs receive diversity technique because SIMO has multiple antennas at the receiver. This configuration eliminates the need for a duplexer and can protect sensitive receiver components from the high power used in transmit. The receiver of this system can either choose the best antenna to receive a stronger signal, or combine from all antennas in such way to maximize the SNR (Signal to Noise Ratio) [7]. The choosing antenna technique called switched diversity or selection diversity [7]. The second techniques that combine the antennas are known as maximal ratio combining (MRC) [7].

The SIMO channel can be modelled using its kind of type which is using single transmitter at one end and n_T receiver antennas at the other end. These things can be written in mathematical form like

$$y = hx + n \tag{2.10}$$

Equation above is known as the discrete-time SIMO channel [5]. Where the h is channel matrix of n_R x 1 given as

$$h = [h_1, h_2, \dots \dots, h_{nR}]^T \tag{2.11}$$

The elements of *hi*, for i = 1, 2, ……, n_R, represent as the constant gain of the channel between the single transmitter antenna and ith receiver antenna over a symbol period [5].

Thus, equation below expressed as the channel capacity of the discrete-time SIMO channel model

$$C = log_2\left(1 + \rho \sum_{i=1}^{n_R} |hi| \right) \tag{2.12}$$

(bps/Hz)

Where *hi* is the same as equation before it is constant gain for *ith* receiver antenna, for I = 1, 2, ……, n_R. The SIMO channel is the opposite of MISO channel which SIMO only has receiver diversity that only used the multiple antennas at the receiver end, unlike MISO that has transmitter diversity. In addition, if the number of receive antenna increase it will only results in a logarithmic increase in average capacity [5].

II.1.4. Multiple-Input Multiple-Output (MIMO)

MIMO stands for Multiple-Input Multiple-Output is one of the smart antenna technologies. The multiple antennas put on both transmitter and receiver antennas to multiply throughput of a radio link. MIMO system that has similar count of antennas at transmitter and receiver in a point to point link is able to multiply the system throughput linearly with every additional antenna [4]. For instance, MIMO system that has 2x2 antennas will double the throughput. Figure (4) described the MIMO configurations.

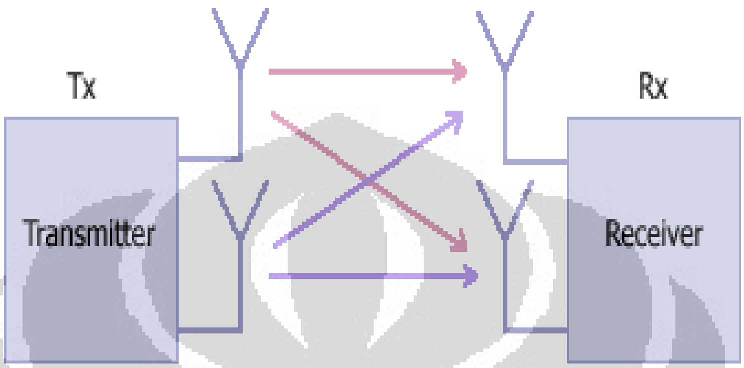

Figure 4: MIMO system

The difference of MIMO system than others it employs Spatial Multiplexing (SM) for enabling signal that has been coded and modulated data stream to be transmitted across different spatial domains. Multiple MIMO modes have been supported by Mobile WiMAX that uses SM or STC or even the both to maximize spectral efficiency to increase throughput without shrinking the coverage area. The dynamic switching between these two modes based on conditions of channel is known as Adaptive MIMO Switching (AMS) [4]. If the AMS combined with Adaptive Antenna System (AAS) which it can focus the transmit energy to the direction of a receiver and vice versa for the receiver, the MIMO system can further boost the performance of WiMAX [4]. Nowadays, MIMO system has been implemented for any wireless technologies such as Local Area Network (LAN), Wide Area Network (WAN), Personal Area Networks (PAN) and others trying to add it to increase the data rate multiple times to satisfy the bandwidth of the broadband users.

The MIMO system works by dividing a data stream into multiple unique streams, and each of which is modulated and transmitted through a different radio-antenna chain at the same time in the same frequency channel [8]. This system is a revolutionary technique that reverses of all people thinking about transmitting radio signal. MIMO leverages the structures of environmental and the multipath signal reflection taken as an advantage to improve the performance of radio transmission [8].

Within the used of multipath, each MIMO receive antenna-radio chain is a linear combination of the multiple transmitted data streams. It separated of the data streams at the receiver using MIMO algorithms that rely on estimates of all channels between each receiver and each transmitter. The route of the multipath treated as discrete channel creating multiple “virtual wires” over which to transmit the signals [8]. Moreover, to multiplying throughput, because of an antenna diversity advantage range is increasing, since each transmitted data stream has measurement from each of the receive antenna.

The MIMO systems are able to provide spatial diversity, time diversity and frequency diversity coherently by combining the use of the transmitter antennas also the receiver antennas [5]. Therefore, it improves the channel capacity and quality of bit error rate (BER) by enhancing wireless transmission over the MIMO channel. By considering a continuous time MIMO channel with n_T transmitter antennas and n_R receiver antennas it will

$$x(t) = [x_1(t), x_2(t), \dots, x_{n_T}(t)]^T \tag{2.13}$$

n_T x 1 vector of transmitted signals, where $[.]^T$ means vector transpose, and H (τ, t) be channel impulse response. Then by doing convolution of the channel impulse response H (τ, t) and the transmitted signals x(t) the received signal can be obtained like equation below [5]

$$y(t) = \int_{-\infty}^{\infty} \boldsymbol{H}(\tau, t)\mathbf{x}(\tau - t)d\tau + \boldsymbol{n}(t) \tag{2.14}$$

Where n (t) is assumed to be the n_R x 1 Gaussian noise vector, and the received signal y (t) expressed as

$$y(t) = [y_1(t), y_2(t), \dots, y_{n_R}(t)]^T \tag{2.15}$$

The equation convolution of the channel impulse response with the transmitted signals above can be written in discrete time representation by sampling the received signal y(t) at t = nT, where T is sampling interval can be expressed as [5]

$$y(t) = \sum_{K=-\infty}^{\infty} \boldsymbol{H}[k,n]\boldsymbol{x}[n-k] + \boldsymbol{n}[n] \quad (2.16)$$

For the equation above it can be simplified by doing the matrix form such as

$$y = Hx + n \quad (2.17)$$

Where H as the channel matrix of n_R x n_T given by

$$H = \begin{bmatrix} h_{11} & h_{12} & \cdots & h_{1n_T} \\ h_{21} & h_{22} & \cdots & h_{2n_T} \\ \vdots & \vdots & \ddots & \vdots \\ h_{n_R1} & h_{n_R2} & \cdots & h_{n_Rn_T} \end{bmatrix} \quad (2.18)$$

Where elements h_{ij}, i = 1, 2, …, n_R and j = 1, 2, …, n_T, denote as the constant gain of the transmitter antenna (jth) and receiver antenna (ith) over a symbol period.

The MIMO system can be expressed by assuming that Ω is a covariance matrix of the transmitter vector x, with n_T transmitter antennas and n_R receiver antennas [5]

$$C = log_2\left[\det\left(I_{n_R} + H\Omega H^H\right)\right] \quad (2.19)$$

(bps/Hz)

The det means determinant, I_{n_R} is the n_R x n_T identity matrix, $(.)^H$ means Hermitian transpose or transpose conjugate, and tr (Ω) ≤ ρ to provide a power constraint, which ρ is signal to noise ratio (SNR) at receiver antennas. If covariance matrix is equal to

$$\Omega = \frac{\rho}{N} I_{n_R} \quad (2.20)$$

N is used to normalize total transmitter power. Therefore, the channel capacity for MIMO with n to n transmitter and receiver respectively, can be shown on equation below

$$C = log_2\left[\det\left(I_{n_R} + \frac{\rho}{N} HH^H\right)\right] \quad (2.21)$$

II.2. Space time block coding (STBC)

Space time block coding is a technique to transmit multiple copies of a data stream across a number of antennas and to exploit the various received versions of the data to improve the reliability of data transfer. This technique used in wireless communication for multiple antenna systems. The fact that the transmitted signal must traverse a potentially difficult environment with scattering, reflection, refraction and so on and may then be further corrupted by thermal noise in the receiver means that some of the received copies of the data will be 'better' than others. This redundancy results in a higher chance of being able to use one or more of the received copies to correctly decode the received signal. In fact, space–time coding combines *all* the copies of the received signal in an optimal way to extract as much information from each of them as possible.

II.2.1. Alamouti Encoder

Alamouti is the simple of space time code which offers a simple method for achieving spatial diversity with two transmit antennas. Consider that the wireless system have transmission sequence such as (x_1, x_2, …, x_n). Usually, for normal transmission the transmitter will send the x_1 to the first time slot, x_2 to the second slot and so on. However, Alamouti advise that the transmitted signal grouped the symbols into groups of two. It means that for the first time slot, it will get x_1 and x_2 from the first and second antenna. In second time slot it will send conjugate of $-x_2$ and conjugate of x_1 also from the first and second antenna and so on. Figure (5) explain clearly about the scheme for Alamouti

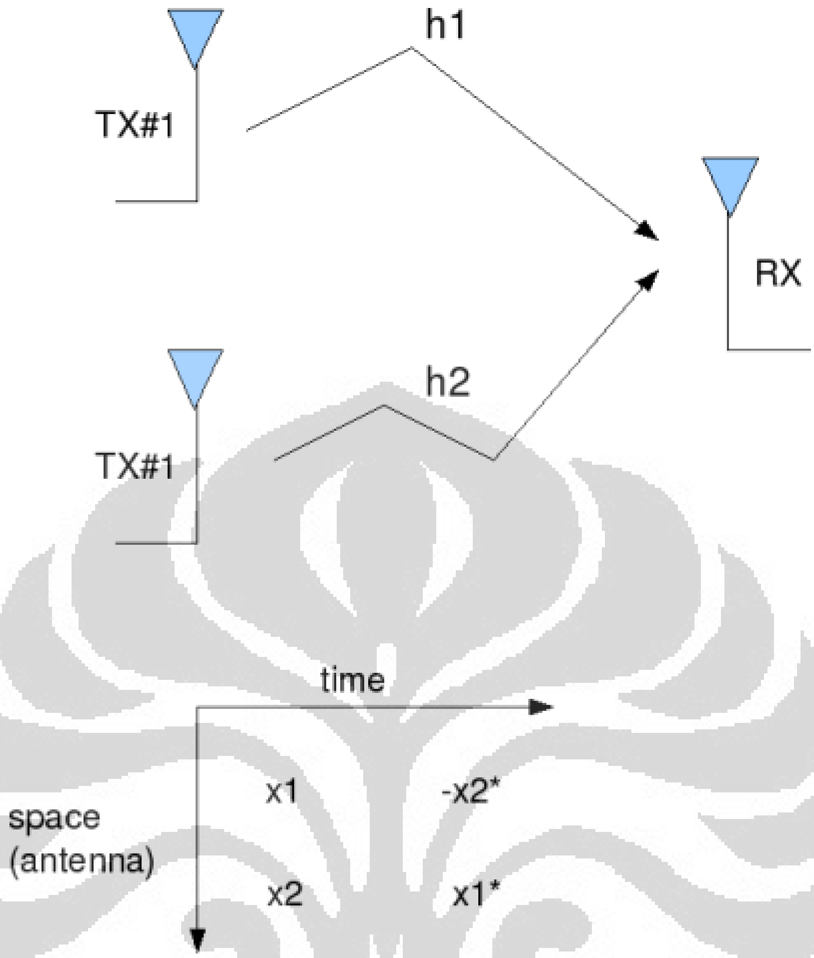

Figure 5: 2 transmit antenna, 1 receive antenna with Alamouti STBC coding

In the first time slot the received signal is,

$$y_1 = h_1 x_1 + h_2 x_2 + n_1 = [h_1 \; h_2]\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + n_1 \tag{2.22}$$

In the second time slot the received signal will be,

$$y_2 = -h_1 x_2^* + h_2 x_1^* + n_2 = [h_1 \; h_2]\begin{bmatrix} -x_2^* \\ x_1^* \end{bmatrix} + n_2 \tag{2.23}$$

Where y_1 and y_2 are the receive symbol on the first and second time slot respectively. The h_1 is the channel from first transmit antenna to receive antenna, also the same as h_2 from second transmit antenna. The x_1 and x_2 are transmitted symbols also n_1 and n_2 are the noise on first and second time slot respectively. Since the two noise terms are independent and identically distributed so,

$$E\left\{\begin{bmatrix} n_1 \\ n_2^* \end{bmatrix}\begin{bmatrix} n_1^* & n_2 \end{bmatrix}\right\} = \begin{bmatrix} |n_1|^2 & 0 \\ 0 & |n_2|^2 \end{bmatrix}. \tag{2.24}$$

Equation (2.24) can be presented in matrix notation like

$$\begin{bmatrix} y_1 \\ y_2^* \end{bmatrix} = \underbrace{\begin{bmatrix} h_1 & h_2 \\ h_2^* & -h_1^* \end{bmatrix}}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} n_1 \\ n_2^* \end{bmatrix}. \tag{2.25}$$

Where H = $\begin{bmatrix} h_1 & h_2 \\ h_2^* & -h_1^* \end{bmatrix}$. The $\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$, can be solved by finding the inverse of H. It is known that for general m x n matrix, the pseudo inverse is defined as, $H = (H^H H)^{-1} H^H$. Since this is a diagonal matrix, the inverse is just the inverse of the diagonal elements,

$$(H^H H)^{-1} = \begin{bmatrix} \frac{1}{|h_1|^2+|h_2|^2} & 0 \\ 0 & \frac{1}{|h_1|^2+|h_2|^2} \end{bmatrix} \tag{2.26}$$

So the estimate of the transmitted symbol is [9]

$$\begin{aligned} \widehat{\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}} &= (H^H H)^{-1} H^H \begin{bmatrix} y_1 \\ y_2^* \end{bmatrix} \\ &= (H^H H)^{-1} H^H \left(H \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} n_1 \\ n_2^* \end{bmatrix} \right) \\ &= \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + (H^H H)^{-1} H^H \begin{bmatrix} n_1 \\ n_2^* \end{bmatrix} \end{aligned} \tag{2.27}$$

Furthermore, figure (6) will illustrate about Alamouti STBC for MIMO system.

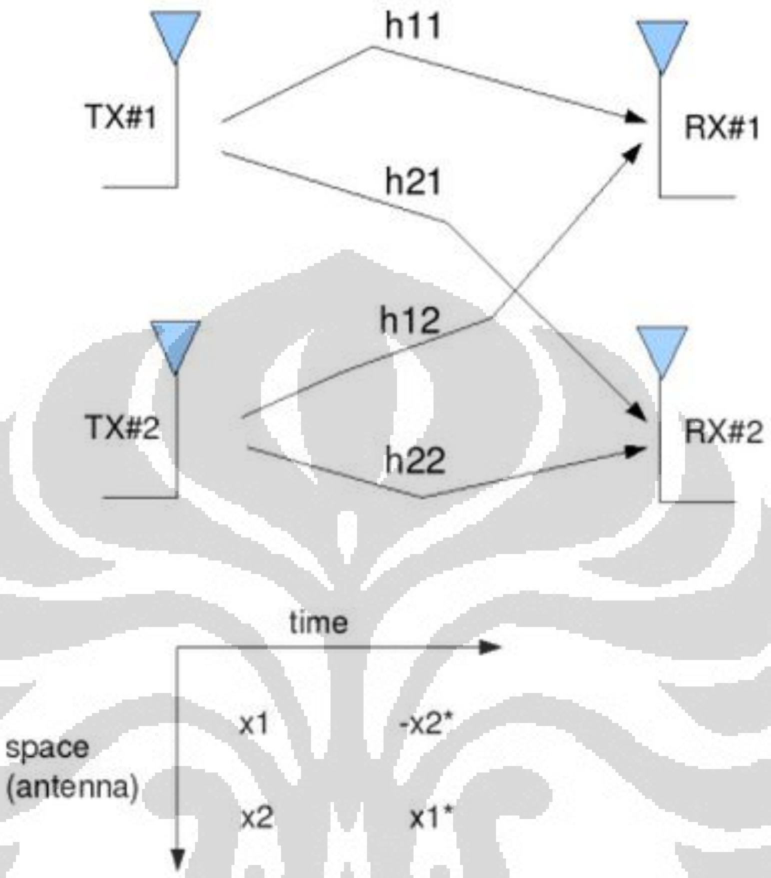

Figure 6: Alamouti STBC MIMO systems

The received signal in the first time slot for MIMO is,

$$\begin{bmatrix} y_1^1 \\ y_2^1 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} n_1^1 \\ n_2^1 \end{bmatrix} \quad (2.28)$$

In here, assuming that the channel remains constant for the second time slot, the received signal in the second time slot will be [10],

$$\begin{bmatrix} y_1^2 \\ y_2^2 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{bmatrix} \begin{bmatrix} -x_2^* \\ x_1^* \end{bmatrix} + \begin{bmatrix} n_1^2 \\ n_2^2 \end{bmatrix} \quad (2.29)$$

Where $\begin{bmatrix} y_1^1 \\ y_2^1 \end{bmatrix}$ are the received information at time slot 1 on receive antenna 1 and 2 respectively. Then, $\begin{bmatrix} y_1^2 \\ y_2^2 \end{bmatrix}$ are the received information at time slot 2 on receive antenna 1 and 2. The x_1 and x_2 are presented of the transmitted symbols. $\begin{bmatrix} n_1^1 \\ n_2^1 \end{bmatrix}$ are the noise at time slot 1 on receive 1 and 2 also $\begin{bmatrix} n_1^2 \\ n_2^2 \end{bmatrix}$ are the noise at time slot 2 on receive antenna 1 and 2 respectively. After that, by combining the equation at time slot 1 and 2, there will be,

$$\begin{bmatrix} y_1^1 \\ y_2^1 \\ y_1^{2*} \\ y_2^{2*} \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \\ h_{12}^* & -h_{11}^* \\ h_{22}^* & -h_{21}^* \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} n_1^1 \\ n_2^1 \\ n_1^{2*} \\ n_2^{2*} \end{bmatrix} \quad (2.30)$$

Where H = $\begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \\ h_{12}^* & -h_{11}^* \\ h_{22}^* & -h_{21}^* \end{bmatrix}$, and by finding the inverse of H, $\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$ can be solved. For general m x n matrix the pseudo inverse defined as, $H = (H^H H)^{-1} H^H$ the term will be,

$$(H^H H) = \begin{bmatrix} |h_{11}|^2 + |h_{21}|^2 + |h_{12}|^2 + |h_{22}|^2 & 0 \\ 0 & |h_{11}|^2 + |h_{21}|^2 + |h_{12}|^2 + |h_{22}|^2 \end{bmatrix} \quad (2.31)$$

So, the inverse of diagonal matrix in equation (2.31) is going to be,

$$(H^H H)^{-1} = \begin{bmatrix} \frac{1}{|h_{11}|^2+|h_{21}|^2+|h_{12}|^2+|h_{22}|^2} & 0 \\ 0 & \frac{1}{|h_{11}|^2+|h_{21}|^2+|h_{12}|^2+|h_{22}|^2} \end{bmatrix} \quad (2.32)$$

From equation (2.32), the estimate of the transmitted symbol would be shown as,

$$\widehat{\begin{bmatrix} x_1 \\ x_2^* \end{bmatrix}} = (H^H H)^{-1} H^H \begin{bmatrix} y_1^1 \\ y_2^1 \\ y_1^{2*} \\ y_2^{2*} \end{bmatrix} \quad (2.33)$$

II.3. Modulation

Modulation is the process of conveying message signal for example a digital bit stream or an analog audio signal, inside another signal that can be physically transmitted. Modulation of a sine waveform is used to transform a baseband message signal into a passband signal, for example low frequency audio signal into a radio frequency signal (RF signal). The purpose of modulation is usually enabling the carrier signal to transport the information in the modulation signal to the receiver. At the receiver, it performs a process of demodulation to extracts the modulation signal from the modulated carrier. PSK (Phase Shift Keying) modulation is digital modulation scheme that conveys data by changing, or modulating, the phase of a reference signal (the carrier wave).

II.3.1. BPSK Modulation

BPSK modulation stands for Binary phase shift keying modulation is the simplest form of PSK modulation type. In binary phase shift keying, the phase of a constant amplitude carrier signal is switched between two values according to the two possible signals m_1 and m_2 corresponding to binary 1 and 0, respectively. Usually, the two phases are separated by 180^0. For example, if the sinusoidal carrier has an amplitude A_C and energy per bit $E_b = \frac{1}{2} A_c^2 T_b$, then the transmitted signal using BPSK modulation is either [11]

$$S_{BPSK}(t) = \sqrt{\frac{2E_b}{T_b}} \cos(2\pi f_c t + \theta_c) \qquad 0 \leq t \leq T_b \text{ (binary 1)} \tag{2.34}$$

Or

$$S_{BPSK}(t) = \sqrt{\frac{2E_b}{T_b}} \cos(2\pi f_c t + \theta_c)$$

$$= -\sqrt{\frac{2E_b}{T_b}} \cos(2\pi f_c t + \theta_c) \qquad 0 \leq t \leq T_b \text{ (binary 0)} \tag{2.35}$$

When m_1 and m_2 presented as binary data signal m (t), then it takes on one of two possible pulse shapes. Then the transmitted signal can be represented,

$$S_{BPSK}(t) = m(t)\sqrt{\frac{2E_b}{T_b}} \cos(2\pi f_c t + \theta_c) \tag{2.36}$$

The BPSK signal is equivalent to a double sideband suppressed carrier amplitude modulated waveform, where $\cos(2\pi f_c t)$ is applied as the carrier, and the data signal m (t) is applied as the modulating waveform [11].

The bit error rate of BPSK in AWGN (Additive White Gaussian Noise) can be calculated as

$$P_b = Q\left(\sqrt{\frac{2E_b}{N_0}}\right) \text{ or } P_b = \frac{1}{2}\operatorname{erfc}\left(\sqrt{\frac{E_b}{N_0}}\right) \tag{2.37}$$

Since there is only one symbol for BPSK, so this is also symbol error rate.

Furthermore, if no multipath impairments are induced by the channel, the received BPSK signal can be expressed as

$$S_{BPSK}(t) = m(t)\sqrt{\frac{2E_b}{T_b}} \cos(2\pi f_c t + \theta_c + \theta_{ch})$$

$$= m(t)\sqrt{\frac{2E_b}{T_b}} \cos(2\pi f_c t + \theta) \quad (2.38)$$

Where θ_{ch} is the phase shift corresponding to the time delay in the channel. The coherent or synchronous demodulation used by BPSK which requires that information about the phase and frequency of the carrier be available at the receiver. The carrier phase and frequency may be recovered at the receiver using a phase locked loop (PLL) if a low level pilot carrier signal is transmitted along with the BPSK signal. Costas loop or squaring loop may be used to synthesize the carrier phase and frequency from the received the BPSK signal if no pilot carrier is transmitted [11]. Figure (7) will explained the block diagram of a BPSK receiver along with the carrier recovery circuits

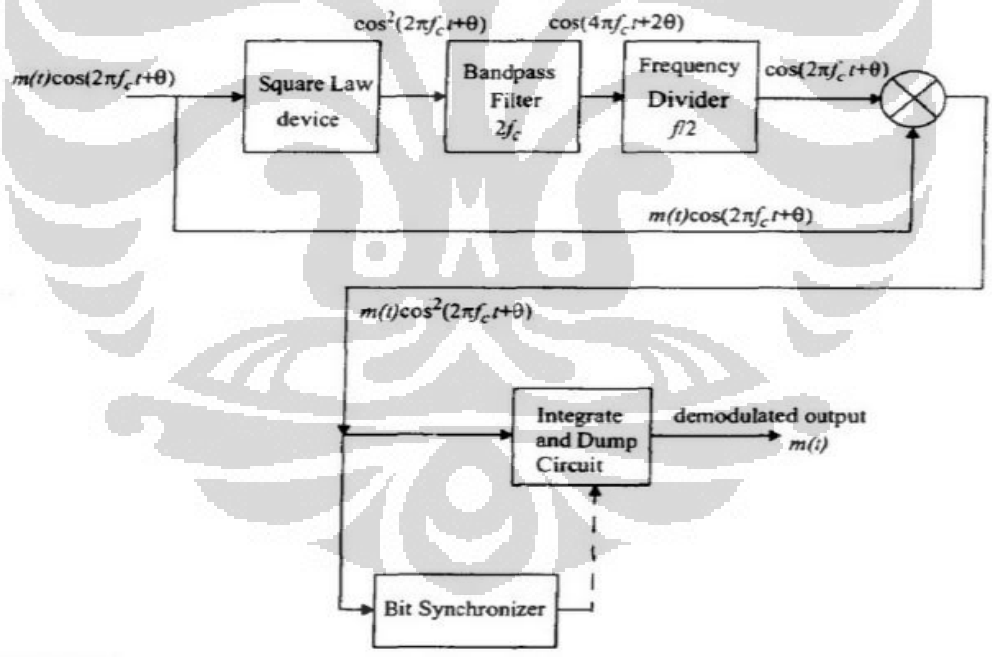

Figure 7 BPSK receiver with carrier recovery circuits

The received signal $\cos(2\pi f_c t + \theta)$ is squared to generate a dc signal and an amplitude varying sinusoid at twice the carrier frequency. By using bandpass filter with centre frequency tuned to $2f_c$ the dc signal is filtered. Then recreate the waveform cos ($2\pi f_c t$ + ɵ) using frequency divider. After the frequency divider, the output of the multiplier can be found which expressed as

$$m(t)\sqrt{\frac{2E_b}{T_b}}\cos^2(2\pi f_c t + \theta) = m(t)\sqrt{\frac{2E_b}{T_b}}\left[\frac{1}{2} + \frac{1}{2}\cos(2\pi f_c t + \theta)\right] \tag{2.39}$$

This signal applied to integrate and Dump Circuit that forms as the low pass filter segment of a BPSK detector. The detection will be optimum if the transmitter and receiver pulse shapes are matched. A bit synchronizer is used to facilitate sampling of the integrator output precisely at the end of each bit period. The switch at the output of the integrator closes to dump the output signal to the decision circuit at the end of each bit period. The decision circuit decides that the received signal corresponds to a binary 1 or 0 depending on whether the integrator output is above or below a certain threshold.

II.3.2. QPSK Modulation

Quadrature Phase Shift Keying (QPSK) is another way of the digital modulation technique. Quadrature Phase Shift Keying (QPSK) is a form of Phase Shift Keying in which two bits are modulated at once, selecting one of four possible carrier phase shifts (0, Π/2, Π, and 3Π/2).

QPSK perform by changing the phase of the In-phase (I) carrier from 0° to 180° and the Quadrature-phase (Q) carrier between 90° and 270°. This is used to indicate the four states of a 2-bit binary code. Each state of these carriers is referred to as a Symbol. Quadrature Phase-shift Keying (QPSK) is a widely used method of transferring digital data by changing or modulating the phase of a carrier signal. In QPSK digital data is represented by 4 points around a circle which correspond to 4 phases of the carrier signal. These points are called symbols

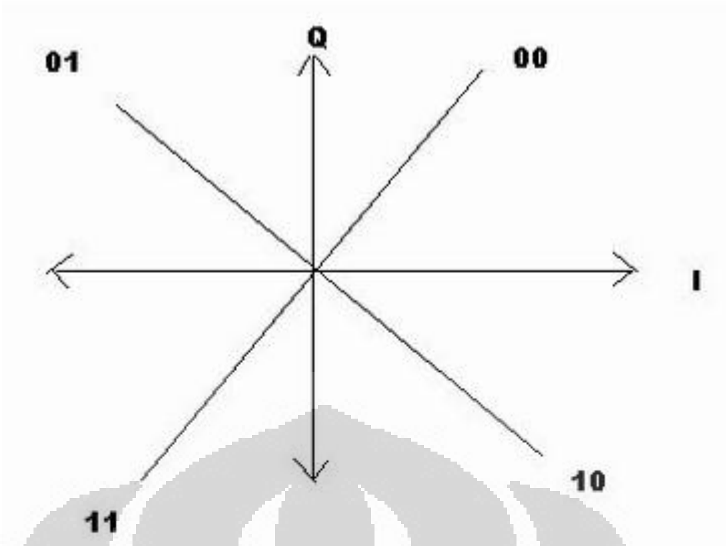

Figure 8: QPSK diagram showing how four different binary codes can be transmitted.

The unipolar binary message (data) first converted into a bipolar non-return-to-zero (NRZ) sequence using a unipolar to bipolar converter. The bit stream is then split into two bit streams I (in-phase) and Q (Quadrature) .The bit stream in-phase (I) is called the “even” stream and quadrature (Q) is called “Odd” stream.

The input data go to the Serial to Parallel Converter then it is split up into two. The two bit stream fed to the Low pass filter (LPF).Then the two bit stream after filtering fed to the modulator. The filter at the output of the modulator confines the power spectrum of the QPSK signal within the allocated band. The two modulator bit stream are summed and fed to the band pass filter (BPF) and produce the QPSK output.

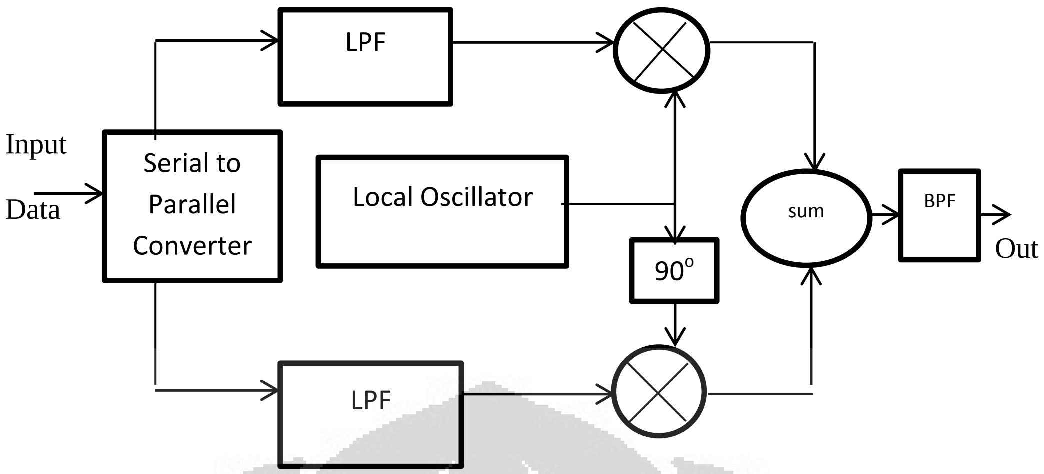

Figure 9: Block diagram of QPSK transmitter

QPSK is used extensively and it is widely known for some application such as:

1. Iridium (voice/data) Satellite Communication System
2. Digital Video Broadcasting Satellite
3. Cable modems
4. Videoconferencing
5. CDMA Systems
6. Cellular phone systems and other form of digital communication over an RF carrier.

II.4. Wireless Channel

There are two distinct environmental conditions that occur in multiple antenna systems.

1. **Line of Sight (LoS)**

   Line-of sight propagation refers to electro-magnetic radiation or electro-magnetic waves travelling in a straight line. The waves are deviated or reflected by obstructions and cannot travel over the horizon or behind obstacles. Above that, material disperses the rays respectively the energy of the waves. Radio signals usually travel at straight lines, however at low frequencies (under 2 MHz) the ground effect transmission causes greatly diffraction, giving photons to partially follow the earth's curvature along multiply deflected straight lines, therefore allowing AM radio signals in low noise environments to be received well after the transmitting antenna has dropped below the horizon. Moreover, frequencies approximately between 1 to 30 MHz can be reflected. Thus, giving radio transmissions in these range frequencies a potentially global reach, again along multiply deflected straight lines. However, at higher frequencies and in lower atmosphere any obstruction between transmitter and receiver will block the signal, it similar like light that senses by eye. Therefore, as the ability to visually sight a transmitting antenna roughly corresponds with the ability to receive signal propagation characteristic of high frequency radio is called line-of-sight. In practice, the propagation characteristics of these radio waves vary substantially depends on the exact frequencies and transmitted signal strength. Figure (10) will explained the line of sight propagation

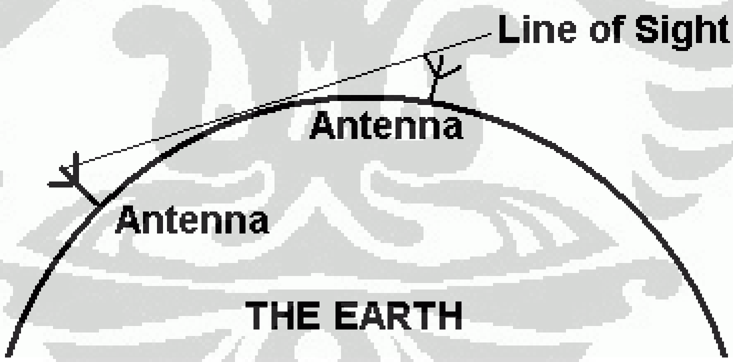

Figure 10: Line-of-Sight propagation

In this condition, the signal propagation for MIMO or even SISO would not encounter any problem with the physical interference along the link path. Therefore, it will disregard for the multipath advantage by MIMO technology and multipath disadvantage happen in SISO technology. However, even though it eliminates the advantage of multipath by MIMO technology, MIMO technology still got some distinct advantage because it uses spatial multiplexing [8]. For example, when the transmitter supplying signal with an identical data stream for MIMO and SISO, the data rate for the MIMO will be X times the data rate of the SISO system where X is the number of transmit/receive antennas.

2. **Non-Line of Sight (NLoS)**

Non-line of Sight is a term used to describe when usually radio waves through path of partially obstructed by a physical object in Fresnel zone (one of infinite number of concentric ellipsoids of revolution which define volumes in the radiation pattern of a circular aperture). The physical object can be many conditions such as buildings, trees, hills, mountains, and something like high voltage electric power lines. Some of the obstacles reflecting radio frequencies while some of it absorb the signals. In both of these cases they both may limit the use of many types of radio transmissions. However, the problem of the NLOS condition has been overcome today. For example, for wireless networks people have been dealing the NLOS conditions by placing relays at additional locations, sending the content of the radio signal around the obstructions. As for today, people also use the multipath signal propagation with bouncing the signal off others nearby objects to get to the receiver for some advance NLOS transmission.

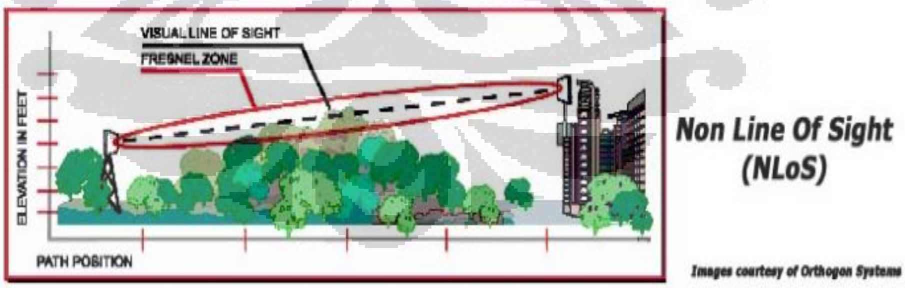

Figure 11: Non-Line of Sight

In this condition, the signal propagation encounters major physical interference along the link path and only altered the signals at the receiving antenna. These reformed signals have the propensity to interfere with each other which mostly results in multipath fading that SISO technology encountered big error. However, by using advanced digital signal processing hardware and very sophisticated algorithms that deal with space time coding in MIMO technology, it becomes possible to decode the multipath differentiated signals which benefit for the users even though they are all on same frequency [8].

II.5. Statistical Model

II.5.1. Rayleigh Fading Distribution

Rayleigh distribution is usually used to describe statistical time varying nature of a flat fading signal or the envelope of an individual multipath component in mobile radio channels. Rayleigh distribution is known as the envelope of the sum of two quadrature Gaussian noise signals. Furthermore, the probability density function (pdf) of Rayleigh distribution is

$$p\,(r) = \begin{cases} \frac{r}{\sigma^2}\left(-\frac{r^2}{2\sigma^2}\right) & (0 \le r \le \infty) \\ 0 & (r < 0) \end{cases} \qquad (2.40)$$

σ is the rms value of received voltage signal before envelope detection, and time-average power of the received signal before envelope detection is noted by σ^2 [11]. The probability envelope of the received signal does not exceed a specified value R is given by the corresponding cumulative distribution function (CDF)

$$P\,(R) = \Pr(r \le R) = \int_0^R p\,(r)dr = 1 - \exp\left(-\frac{R^2}{2\sigma^2}\right) \qquad (2.41)$$

The mean value r_{mean} of the Rayleigh distribution is given by

$$r_{mean} = E[r] = \int_0^\infty rp\,(r)dr = \sigma\sqrt{\frac{\pi}{2}} = 1.2533\sigma \qquad (2.42)$$

And the variance of the Rayleigh distribution is given by σ^2, which represents the ac power in the signal envelope

$$\sigma_r^2 = E[\,r^2] - E^2[r] = \int_0^\infty r^2 p\,(r)dr - \frac{\sigma^2\pi}{2}$$

$$= \sigma^2 \left(2 - \frac{\pi}{2} \right) = 0.4292\sigma^2 \qquad (2.43)$$

The square root of the mean square is the rms value of the envelope, or $\sqrt{2}\,\sigma$ where the standard deviation of the original complex Gaussian signals prior to envelope detection noted by σ.

The median value of r is found by solving

$$\frac{1}{2} = \int_0^{r_{median}} p(r)dr \qquad (2.44)$$

And r_{median} becomes

$$r_{median} = 1.177\sigma \qquad (2.45)$$

Therefore, the difference between mean and median is only 0.55 dB in Rayleigh fading signal. In practice, median is often used, since fading data are usually measured in the field and a particular distribution cannot be assumed. It is easy to compare different fading distribution which may have widely varying the mean by using median better than the mean.

II.5.2. Ricean Fading Distribution

If there is dominant non fading signal component present, like line of sight propagation path, the small scale fading envelope distribution is Ricean. In this condition, random multipath component landing on different angles are superimposed on a stationary dominant signal. These will effect of adding dc component to random multipath at the output of an envelope detector.

For example, detection of a sine wave in thermal noise will effect of a dominant signal arriving with many weaker multipath signals gives rise to the

Ricean distribution. As the dominant becomes weaker, the composite signal resembles a noise signal which has an envelope that is Rayleigh, therefore, when dominant component fades away, the Ricean distribution degenerates to Rayleigh distribution

The Ricean probability density function is expressed by

$$p(r) = \begin{cases} \frac{r}{\sigma^2} e^{-\frac{(r^2 + A^2)}{2\sigma^2}} I_0\left(\frac{Ar}{\sigma^2}\right) & for\ (A \geq 0, r \geq 0 \\ 0 & for\ (\ r \leq 0\ ) \end{cases} \quad (2.46)$$

The peak amplitude of the dominant signal is noted by A and I_0 (.) is the modified Bessel function of the first kind and zero order. Parameter K is often described of Ricean distribution which is defined as the ratio between the deterministic signal power and the variance of the multipath. It is given by K = A^2 / ($2\sigma^2$). In terms of dB it becomes

$$K\ (dB) = 10 \log \frac{A^2}{2\sigma^2}\ dB \quad (2.47)$$

Ricean factor is described as the parameter K and fully specifies the Ricean distribution. As A → 0, K → -∞ dB, and as the dominant path decreases in amplitude, the Ricean distribution degenerates to a Rayleigh distribution.

II.6. Channel Estimation

Channel estimation or equalization is the process of reducing amplitude, frequency and phase distortion in a radio channel with the intent of improving transmission performance [1]. As ISI (inter symbol interference) that caused by multipath in band limited (frequency selective) time dispersive channels distorts the transmitted signal, causing bit errors at the receiver. Therefore, the purpose of equalizer is to reduce the inter symbol interference (ISI) as much as possible to maximize the probability of correct decisions. This is only mathematical estimation of what is occurs in nature. Also by using channel estimation it can allows the receiver to approximate the effect of the channel on the signal. Block figure (12) illustrate the general procedure of the channel estimation.

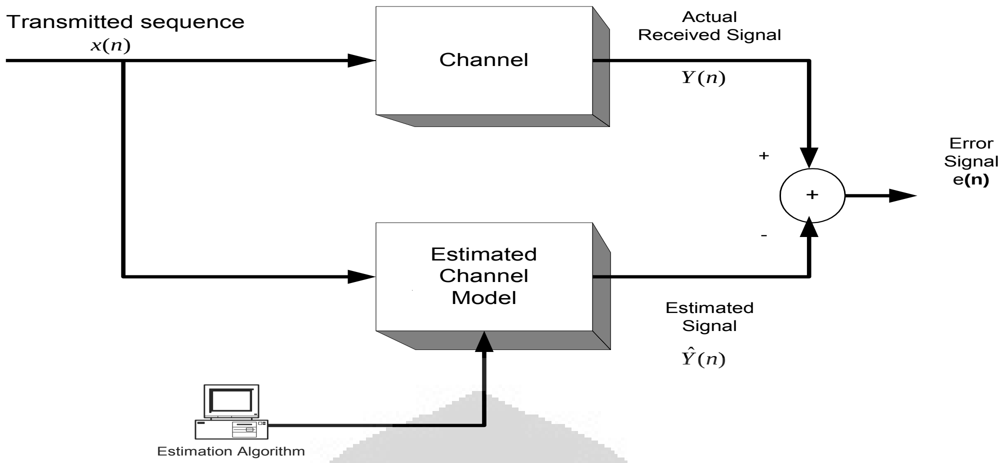

Figure 12: Procedure of Channel estimation

Furthermore, some theories of channel estimation with different methods will be discussed.

II.6.1. Least Square

Considering that communication system is corrupted by noise. Digital signal which is transmitted over fading multipath channel, after which the signal has memory of L symbols. At receiver thermal noise is generated and it is modelled by additive white Gaussian noise. The problem of demodulation here is to detect the transmitted bits from the received signal. Moreover, the received signal the detector needs also the channel estimates, which are provided by specific channel estimator device. Figure (13) explained all the process of transmitter to receiver with channel estimator and noise.

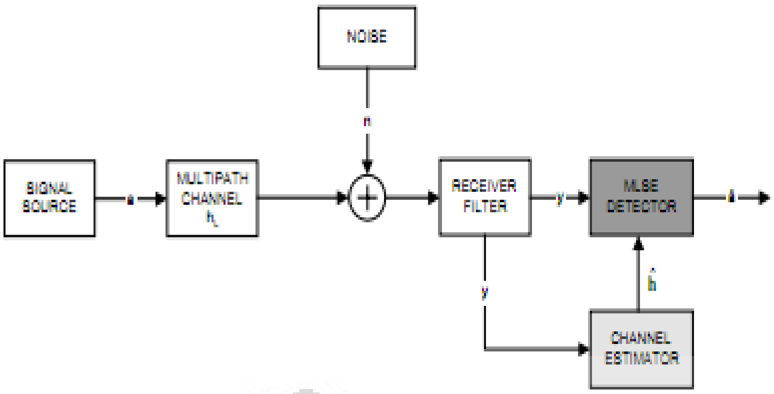

Figure 13: Block diagram of noise corrupted system.

The received signal is expressed as

$$\mathbf{y} = \mathbf{Mh} + \mathbf{n} \tag{2.48}$$

Where the complex channel impulse response of h of the wanted signal is

$$\mathbf{h} = [h_0 \quad h_1 \quad \cdots \quad h_L]^T \tag{2.49}$$

And noise samples is denoted by n. within each transmission burst the transmitter sends a unique training sequence, which is divided into a reference length of P and guard period of L bits and expressed like

$$\mathbf{m} = [m_0 \quad m_1 \quad \cdots \quad m_{P+L-1}]^T \tag{2.50}$$

Having bipolar elements the circulant training sequence matrix M is formed

$$\mathbf{M} = \begin{bmatrix} m_L & \cdots & m_1 & m_0 \\ m_{L+1} & \cdots & m_2 & m_1 \\ \vdots & & \vdots & \vdots \\ m_{L+P-1} & \cdots & m_P & m_{P-1} \end{bmatrix}$$

(2.51)

The LS estimates are found by minimising the following squared error quantity [12]

$$\hat{\mathbf{h}} = \arg\min_{\mathbf{h}} \|\mathbf{y} - \mathbf{M}\mathbf{h}\|^2$$

(2.52)

With the white Gaussian noise the solution will be

$$\hat{\mathbf{h}}_{LS} = \left(\mathbf{M}^H\mathbf{M}\right)^{-1}\mathbf{M}^H\mathbf{y}$$

(2.53)

Where $(\ )^H$ and $(\ )^{-1}$ explained as the Hermitian and inverse matrices, respectively [12]. The solution of $\hat{\mathbf{h}} = \arg\min_{\mathbf{h}} \|\mathbf{y} - \mathbf{M}\mathbf{h}\|^2$ is also the best linear unbiased estimate for the channel coefficients. Then the solution is simplified to

$$\hat{\mathbf{h}} = \frac{1}{P}\mathbf{M}^H\mathbf{y}$$

(2.54)

Provided that the periodic autocorrelation function of the training sequence is ideal with the small delays from 1 to L, because the correlation matrix $M^H M$ becomes diagonal. This last estimates equation is the scaled correlations between the received signal and training sequence.

II.6.2. Zero-force

For zero forcing equalizer, the equalizer coefficients c_n are chosen to force the samples of the combined channel and equalizer impulse response to zero at all but one of the NT spaced sampled points in the tapped delay line filter

[11]. An infinite length equalizer with zero ISI at the output can be obtained by letting the number of coefficient increase without bound. Frequency response $H_{eq}(f)$ of the equalizer is periodic with a period equal to the symbol rate 1/T when each of the delay elements provide a time delay equal to the symbol duration T. the combined response of the channel with the equalizer must satisfy Nyquist's first criterion (filter with an impulse response achieve ISI cancellation).

$$H_{ch}\,(f)H_{eq}\,(f) = 1, |f| < 1/2T \tag{2.55}$$

$H_{ch}(f)$ is the folded frequency response of the channel. Therefore, zero, infinite length, ISI equalizer is an inverse filter which inverts the folded frequency response of the channel [11]. By using a truncated length version the infinite length equalizer is usually implemented.

The zero forcing algorithms have the disadvantage that the inverse filter may excessively amplify noise at frequencies where the folded channel spectrum has high attenuation [11]. This zero force equalizer is not often used for wireless links in practical because it neglects the effect of noise altogether. On the other hand, this algorithm performs well for static channels with high SNR like wired telephone lines.

II.6.3. Minimum Mean Square Error (MMSE)

If the channel and noise distributions are known, then this a priori information can be exploited to decrease the estimation error. This approach is known as Bayesian estimation and for Rayleigh fading channels it exploits that

$$\mathrm{vec}(\mathbf{H}) \sim \mathcal{CN}(0,\, \mathbf{R}), \quad \mathrm{vec}(\mathbf{N}) \sim \mathcal{CN}(0,\, \mathbf{S}). \tag{2.56}$$

The MMSE estimator is the Bayesian counterpart to the least-square estimator and becomes [13]

$$\mathrm{H}_{\text{MMSE-estimate}} = \left(\mathbf{R}^{-1} + (\mathbf{P}^T \otimes \mathbf{I})^H \mathbf{S}^{-1} (\mathbf{P}^T \otimes \mathbf{I})\right)^{-1} (\mathbf{P}^T \otimes \mathbf{I})^H \mathbf{S}^{-1} \mathrm{vec}(\mathbf{Y}) \tag{2.57}$$

where $\otimes$ denotes the Kronecker product and the identity matrix $\mathbf{I}$ has the dimension of the number of receive antennas. The estimation Mean Square Error (MSE) is

$$\mathrm{tr}\left(\mathbf{R}^{-1} + (\mathbf{P}^T \otimes \mathbf{I})^H \mathbf{S}^{-1} (\mathbf{P}^T \otimes \mathbf{I})\right)^{-1} \tag{2.58}$$

and it will minimized by a training matrix $\mathbf{P}$ that in general can only be derived through numerical optimization. But there exist heuristic solutions with good performance based on waterfilling. As opposed to least-square estimation, the error estimation for spatially correlated channels can be minimized even if N is smaller than the number of transmitted antennas [13]. Thus, MMSE estimation can both decrease the estimation error and shorten the required training sequence. It needs however additionally the knowledge of the channel correlation matrix $\mathbf{R}$ and noise correlation matrix $\mathbf{S}$. In absence of an accurate knowledge of these correlation matrices, robust choices need to be made to avoid MSE degradation.

II.6.4. Maximum likelihood sequence estimation

Maximum-likelihood sequence estimation (MLSE) avoids the problem of noise enhancement since it does not use an equalizing filter instead it estimates the sequence of transmitted symbols. Given the channel response *h (t)*, the MLSE algorithm chooses the input sequence $\{d_k\}$ that maximizes the likelihood of the received signal *w (t)*.

Using a Gram-Schmidt orthonormalization procedure, it can express *w (t)* on a time interval $[0, LT_s]$ as [14]

$$w(t) = \sum_{n=1}^{N} w_n \phi_n(t), \tag{2.59}$$

Where

$$h_{nk} = \int_0^{LT_s} h(t - kT_s)\phi_n^*(t)dt \tag{2.60}$$

And

$$\nu_n = \int_0^{LT_s} n(t)\phi_n^*(t)dt. \tag{2.61}$$

The v_n are complex Gaussian random variables with mean zero and covariance

$$.5E[\nu_n^* \nu_m] = N_0 \delta[n - m]. \tag{2.62}$$

Thus, $w^N = (w_1, \ldots, w_N)$ has a multivariate Gaussian distribution

$$p(\mathrm{w}^N | d^L, h(t)) = \prod_{n=1}^{N} \left[\frac{1}{\pi N_o} \exp\left[-\frac{1}{N_0} \left| w_n - \sum_{k=0}^{L} d_k h_{nk} \right|^2 \right] \right] \tag{2.63}$$

Given a received signal *w(t)* or, equivalently, w^N, the MLSE decodes this as symbol sequence d^L that maximizes the likelihood function $p(w^N|d^L, h(t))$. The MLSE outputs the sequence

$$\begin{aligned}
\hat{d}^L &= \arg\max \left[\log p(\mathrm{w}^N | d^L, h(t))\right] \\
&= \arg\max \left[-\sum_{n=1}^{N} |w_n - \sum_k d_k h_{nk}|^2 \right] \\
&= \arg\max \left[-\sum_{n=1}^{N} |w_n|^2 + \sum_{n=1}^{N} \left(w_n^* \sum_k d_k h_{nk} + w_n \sum_k d_k^* h_{nk}^* \right) - \sum_{n=1}^{N} \left(\sum_k d_k h_{nk} \right) \left(\sum_m d_m^* h_{nm}^* \right) \right] \\
&= \arg\max \left[2\Re \left\{ \sum_k d_k^* \sum_{n=1}^{N} w_n h_{nk}^* \right\} - \sum_k \sum_m d_k d_m^* \sum_{n=1}^{N} h_{nk} h_{nm}^* \right].
\end{aligned} \tag{2.64}$$

Note that

$$\sum_{n=1}^{N} w_n h_{nk}^* = \int_{-\infty}^{\infty} w(\tau) h^*(\tau - nT_s) d\tau = y[n] \tag{2.65}$$

And

$$\sum_{n=1}^{N} h_{nk}h_{nm}^* = \int_{-\infty}^{\infty} h(\tau - kT_s)h^*(\tau - mT_s)d\tau = f[k-m] \tag{2.67}$$

Combining three equations above it concludes that

$$\hat{d}^L = \arg\max \left[2\Re\left\{ \sum_k d_k^* y[k] \right\} - \sum_k \sum_m d_k d_m^* f[k-m] \right] \tag{2.68}$$

It can be seen from this equation that MLSE output depends only on the sampler output {y[k]} and the channel parameters $f[n-k] = f(nT_s - kT_s)$ where $f(t) = h(t) * h^*(-t)$ [14].

CHAPTER III

SYSTEM MODEL

In digital communication, there are several things that must be accounted for in transmitting message from transmitter to receiver. For example, after generating the data in transmitter other things like the type of modulation, and channel coding need to be considered. Furthermore, the data also need to be specified with type of wireless channel and noise addition. Also in receiver, demodulating the data and checking the error that has occurred in system need to be carefully measured by using the detection type. Additionally, channel estimation was being used to estimate the data that has been transmitted and remove the interference symbol that occurred in wireless propagation. Block diagram in figure (14) will illustrate as a model for the methodology.

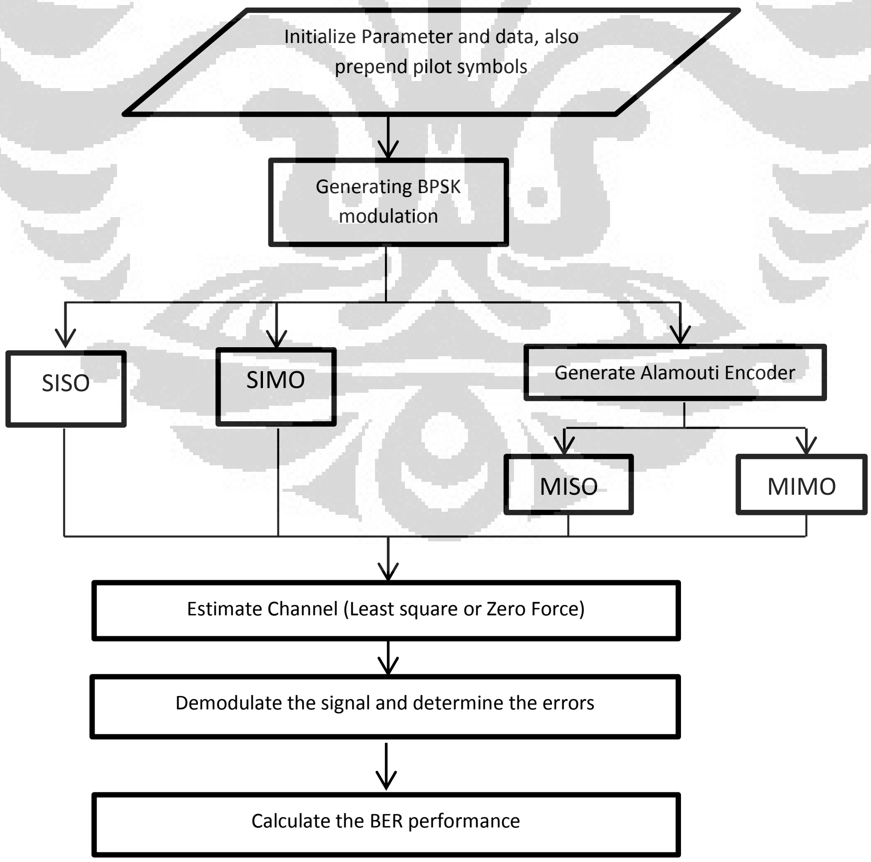

Figure 14: Flow chart of methodology

III.1. SISO model

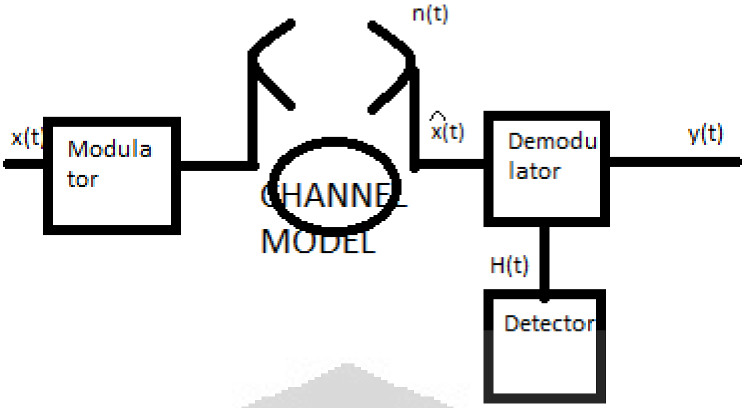

Figure 15: SISO model

Figure (15) illustrate as SISO model in wireless communication with x as the transmitted signal, H as the channel estimation or equalization, n as the noise occurs in system which in this type is Gaussian noise and y as the received signal. These notations can be compute as

$$y(t) = H(t)\hat{x}(t) + n(t) \tag{3.1}$$

Firstly, the transmitted signal sent data through the modulator. Then, BPSK modulation was generated. Therefore, the transmitted signal will be

$$x(t) = \sqrt{\frac{2Eb}{Tb}}\cos(2\pi f_c t + \theta_c) \tag{3.2}$$

Then, 8 pilot symbols per frame were prepends using the hadamard matrices. A hadamard matrices of order n is a solution to Hadamard's maximum determinant problem, i.e., has the maximum possible determinant (in absolute value) of any $n \times n$ complex matrix with elements $|a_{i,j}| \le 1$ (Brenner and Cummings 1972), namely $n^{n/2}$. An equivalent definition of the Hadamard matrices is given by

$$h_n h_n^T = n\, I_n \tag{3.3}$$

where I_n is the $n \times n$ identity matrix.

In SISO model, the hadamard matrices only used the first row vector frame in the order of 8. This is because SISO only used single input. Therefore, the matrices can be shown in (3.3)

$$h = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix} \quad (3.4)$$

After that, the signal going through the channel model or distribution channel such as Rayleigh or Rician fading distribution. These types of distribution are the one that has been used in this research. For Rayleigh distribution, the probability of density function can be obtained by using

$$p\,(h) = \begin{cases} \frac{h}{\sigma^2}\left(-\frac{h^2}{2\sigma^2}\right) & (0 \leq h \leq \infty) \\ 0 & (h < 0) \end{cases} \quad (3.5)$$

σ is the rms value of received voltage signal before envelope detection, and time-average power of the received signal before envelope detection is noted by σ^2 [11]. And Rician distribution probability density function can be solved by using

$$p(h) = \begin{cases} \frac{h}{\sigma^2} e^{-\frac{(h^2+A^2)}{2\sigma^2}} I_0\left(\frac{Ah}{\sigma^2}\right) & for\ (A \geq 0, r \geq 0 \\ 0 & for\ (h \leq 0) \end{cases} \quad (3.6)$$

Then the received signal can be configured with

$$\hat{x}(t) = x(t)h(t) + n(t)$$

Next, the channel estimation methods were configured. Least square and zero force have performed in this simulation. Least square estimation is way to estimate the signal that being transmit through the modulation and distribution by minimizes the parameter pilot symbols with the received signal.

$$H(t) = H_{LS}\,(\mathrm{t}) \quad (3.7)$$

$$H_{LS}(t) = \frac{X^T \hat{x}(t)}{pilot\ length} \quad (3.8)$$

Where X is the pilot symbol and $\hat{x}$ is the received signal and in this simulation the pilot length is 8. By using this method, the estimated signal that has been transmitted can be achieved.

For zero force, the method for cancelling the ISI is the inverse of the frequency response of the channel or the received signal. Therefore, the method will be

$$H(t) = H_{ZF}(t) \tag{3.9}$$

$$\hat{x}(f) H_{ZF}(f) = 1, |f| < 1/2T \tag{3.10}$$

And it can be simplify as

$$H_{ZF}(t) = \hat{x}^{-1}(t) \tag{3.11}$$

After that, combining the signal using the estimated channel and demodulation was performed. In SISO, the way to combine the signal is using the received signal with the conjugate of estimated signal.

$$y(t) = \hat{x}(t)(H(t)^{*}) \tag{3.12}$$

Finally, Bit error rate can be compute by comparing the received signal with the transmitted signal.

$$BER_{SISO} = y(t) - x(t) \tag{3.13}$$

III.2. SIMO model

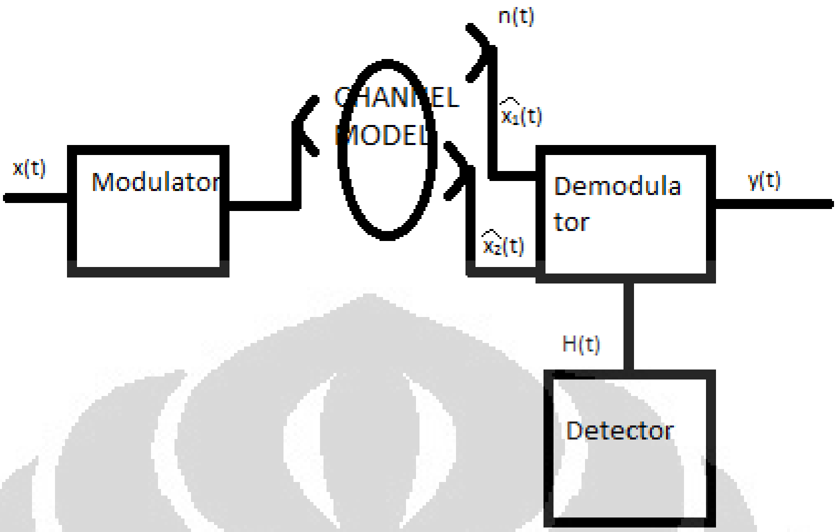

Figure 16: SIMO model

Figure (16) illustrate as model of wireless communication with x as the transmitted signal, H as the channel estimation or equalization, n as the noise occurs in system which in this type is Gaussian noise and y as the received signal. The equation can follow (3.1).

The signal then transmitted through BPSK modulation which the equation follow (3.2). Due to SIMO has two antennas at the receiver, the signal transmit to two different antennas which give more signal power at the end.

Then, 8 pilot symbols per frame were prepends using the hadamard matrices. SIMO model has the same antenna input with SISO which is only single antenna input. Therefore, hadamard matrices only used the first row vector frame in the order of 8. The matrices can be shown in (3.4)

After that the signal going through the distribution channel which is Rayleigh or Rician. The probability density function of Rayleigh and Rician are follow equation (3.5) and (3.6), respectively. The signal that has been transmitted can be expressed as

$$\hat{x}_1(t) = x(t)h(t) + n(t) \qquad (3.14)$$

$$\hat{x}_2(t) = x(t)h(t) + n(t) \tag{3.15}$$

Then, the channel estimation needs to be processed. Least square and zero force are performing in this simulation. Least square estimation can be expressed as

$$H_1(t) = H_{LS1}(t) \tag{3.16}$$

$$H_2(t) = H_{LS2}(t) \tag{3.17}$$

$$H_{LS1}(t) = \frac{X^T \hat{x}_1(t)}{pilot\ length} \tag{3.18}$$

$$H_{LS2}(t) = \frac{X^T \hat{x}_2(t)}{pilot\ length} \tag{3.19}$$

Where X is the pilot symbol and $\hat{x}$ is the received signal and in this simulation the pilot length is 8. By using this method, the estimated signal that has been transmitted can be achieved.

While zero force can be found by

$$H_1(t) = H_{ZF1}(t) \tag{3.20}$$

$$H_2(t) = H_{ZF2}(t) \tag{3.21}$$

$$H_{ZF1}(t) = \hat{x}_1{}^{-1}(t) \tag{3.22}$$

$$H_{ZF2}(t) = \hat{x}_2{}^{-1}(t) \tag{3.23}$$

After that, combining the signal using the estimated channel and demodulation was performed. Due to SIMO has two or more antennas at the receiver, therefore combining the signal can be found by summing or total of one antenna with another antenna.

$$y(t) = sum(\hat{x}_1(t)(H_1\ (t)^*\ and\ \hat{x}_2(t)(H_2\ (t)^*\ ) \quad (3.24)$$

Finally, Bit error rate can be compute by comparing the received signal with the transmitted signal.

$$BER_{SIMO} = y(t) - x(t) \quad (3.25)$$

III.3. MISO model

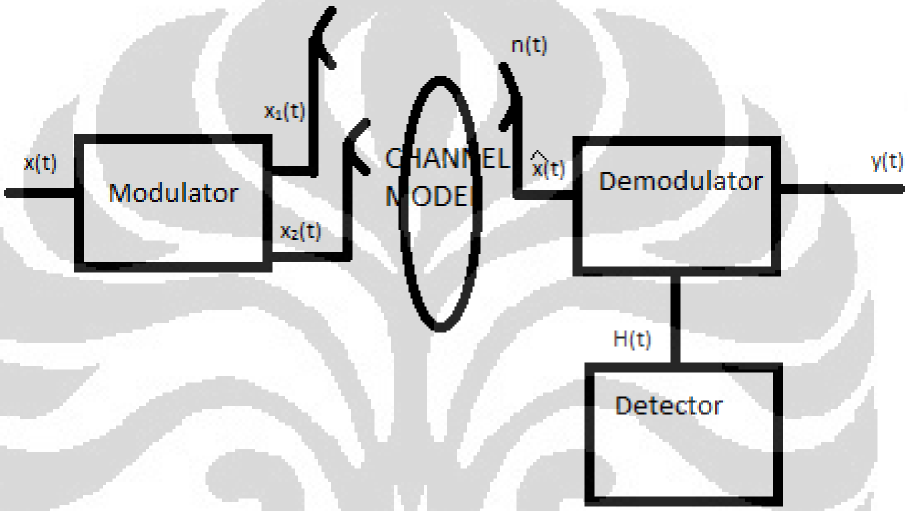

Figure 17: MISO model

Figure (17) illustrate as a MISO model in wireless communication with x as the transmitted signal, H as the channel estimation or equalization, n as the noise occurs in system which in this type is Gaussian noise and y as the received signal. The equation can follow (3.1).

Firstly, the transmitted signal sent data through the modulator. Then, BPSK modulation was generated to the multiple antennas input. Therefore, the transmitted signal will be

$$x_1(t) = \sqrt{\frac{2Eb}{Tb}}\cos(2\pi f_c t + \theta_c) \quad (3.26)$$

$$x_2(t) = \sqrt{\frac{2Eb}{Tb}}\cos(2\pi f_c t + \theta_c) \quad (3.27)$$

Then, 8 pilot symbols per frame were prepends using the hadamard matrices. Due to MISO has multiple antennas input, therefore the Hadamard matrices used first and second vector which is given by

$$h = \begin{bmatrix} 1 & 1 \\ 1 & -1 \\ 1 & 1 \\ 1 & -1 \\ 1 & 1 \\ 1 & -1 \\ 1 & 1 \\ 1 & -1 \end{bmatrix} \quad (3.28)$$

After that the signal going through the distribution channel which is Rayleigh or Rician. The probability density function of Rayleigh and Rician are follow equation (3.5) and (3.6), respectively.

However, Alamouti encoder was used for helping the MISO system configured further signal. This is because MISO using transmits diversity. Alamouti advised that the transmitted signal grouped the symbols into groups of two which means for MISO it will be

$$\widehat{x_1}(t) = h(t)x_1(t) + h(t)x_2(t) + n_1(t) \quad (3.29)$$

And

$$\widehat{x_2}(t) = -h(t)x_2^*(t) + \; h_2(t)x_1^*(t) + n(t) \quad (3.30)$$

Least square and zero force also have been used as the channel estimation method for MISO system. The equation for least square in MISO system can be follow with equation (3.16) until (3.19).For zero force, the equation will follow (3.20) until (3.23)

Then combining the signal was playing part also demodulation need to be processed afterwards. The equation for combining the signal will be

$$H_1(t) = half\ length\ of H(t) \quad (3.31)$$

$$H_2(t) = another\ half\ length\ of H(t) \quad (3.32)$$

$$y_1(t) = \widehat{x_1}(t)(H_1(t)^*) + \widehat{x_2}(t)^*((H_2(t)^*) \quad (3.33)$$

And

$$y_2(t) = \widehat{x_1}(t)(H_2(t)^*) - \widehat{x_2}(t)^*((H_1(t)^*) \quad (3.34)$$

Then the total combiner will be

$$y(t) = \ combining\ the\ y_1(t)\ as\ the\ first\ frame\ and\ y_2(t)\ second\ frame \quad (3.35)$$

Finally, the bit error rate for MISO system can be found

$$BER_{MISO} = y(t) - x(t) \quad (3.36)$$

III.4. MIMO model

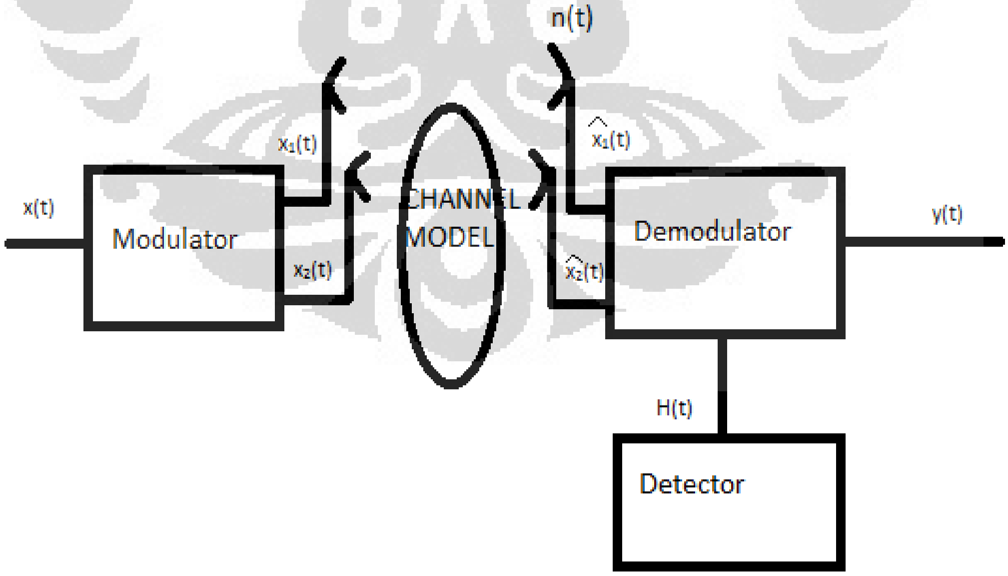

Figure 18: MIMO model

Figure (18) illustrate as MIMO model in wireless communication with x as the transmitted signal, H as the channel estimation or equalization, n as the noise occurs in system which in this type is Gaussian noise and y as the received signal. The equation can follow (3.1).

Firstly, the transmitted signal sent data through the modulator. Then, BPSK modulation was generated. Therefore, the transmitted signal will be like equation (3.26) and (3.27)

Then, 8 pilot symbols per frame were prepends using the hadamard matrices. MIMO has the similar processes when using pilot symbols which is using first and second vector like equation (3.28).

After that the signal going through the distribution channel which is Rayleigh or Rician. The probability density function of Rayleigh and Rician are follow the equation (3.5) and (3.6), respectively.

MIMO also has similar process when transmitting the data. However, it will be a bit complex than MISO as MIMO has two or more antennas at the receiver. Therefore, using Alamouti encoder the equation can be expressed as

$$\begin{bmatrix} \widehat{x_1^1} \\ \widehat{x_2^1} \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{bmatrix} \begin{bmatrix} x_{bpsk1} \\ x_{bpsk2} \end{bmatrix} + \begin{bmatrix} n_1^1 \\ n_2^1 \end{bmatrix} \tag{3.37}$$

And the second slot will be

$$\begin{bmatrix} \widehat{x_1^2} \\ \widehat{x_2^2} \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{bmatrix} \begin{bmatrix} -x_{bpsk2}^* \\ x_{bpsk1}^1 \end{bmatrix} + \begin{bmatrix} n_1^2 \\ n_2^2 \end{bmatrix} \tag{3.38}$$

Next, the channel estimation methods need to be configured. Least square and zero force have performed in this simulation. Least square and zero force also have been used as the channel estimation method for MISO system. The equation for least square in MISO system can be follow with equation (3.16) until (3.19).For zero force, the equation will follow (3.20) until (3.23)

MIMO has similar method with MISO for combining the estimated channel. It adapted with equation (3.31), (3.32), (3.33) and (3.34). However, in the last part, MIMO need to perform the calculation of the sum of the signal received from one antenna with another, which is the same way as SIMO due to MIMO have two or more receiver. So for the first equation it follows the MISO but the final part is going to be

$$y(t) = sum(y_1(t) and\ y_2\ (t)) \qquad (3.39)$$

After doing the demodulation part, determining the error playing its role. To determine the error, the process only compares the bit that already sent to the receiver with the bits that were being transmitted to the receiver. The notation can be seen as

$$BER_{MIMO} = \ y(t) - x(t) \qquad (3.40)$$

CHAPTER IV

RESULT AND ANALYSIS

In this research, two channel estimation methods have been completed in each of antenna systems and wireless channels. Least square and zero force estimation methods were being used in this simulation type. Therefore, the result of the simulation between the differences of two channel estimation methods will be compared and analysed.

First, the least square and zero force estimation with Rayleigh channel of single path and additive white Gaussian noise will be discussed. Figure (16) is the simulation result of these types of method.

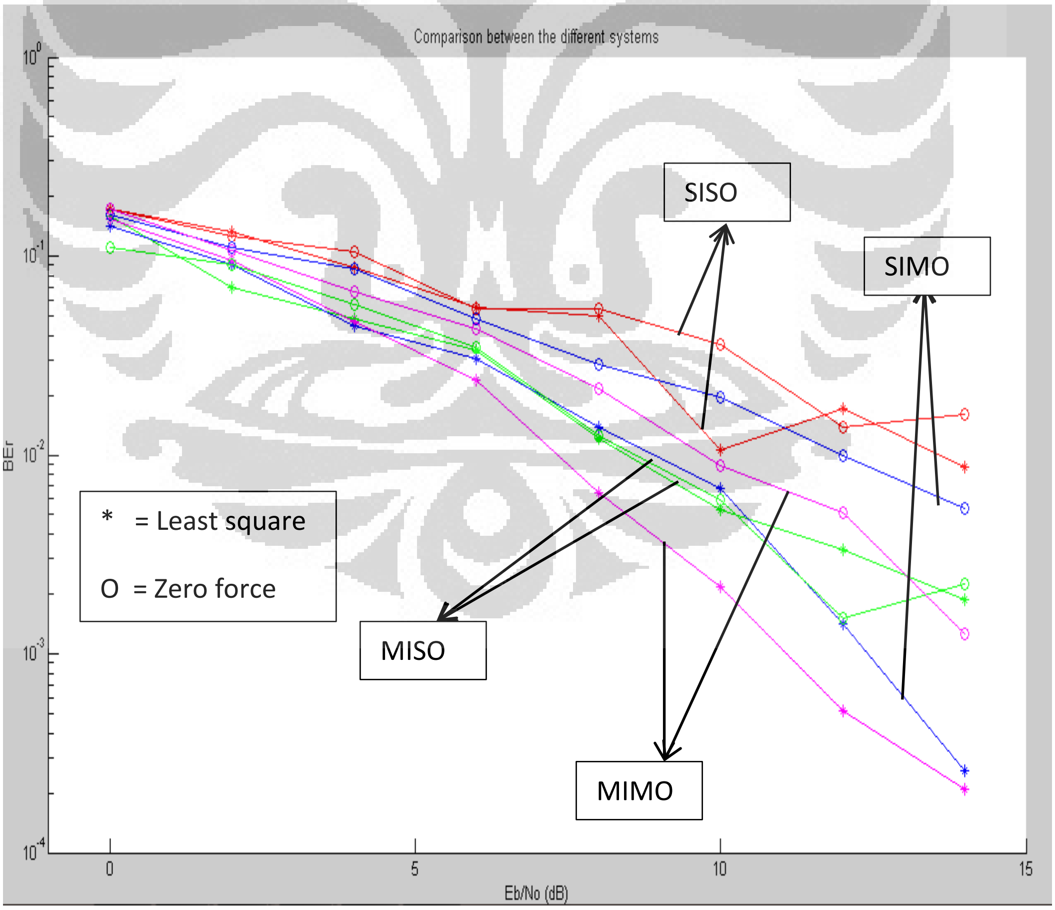

Figure 19: BEr performance of least square and zero force estimation in Rayleigh single path

It can be seen in least square that MIMO performs better than others and SISO was at the least performance of all. Also in high SNR, MIMO system performs the best of all. It is refers to the theory of MIMO performs better than others as it has multiple antenna at the inputs and multiple antenna at the output which gives better signal by using the MIMO estimated error techniques. Least square estimation aided with pilot symbols is used in this simulation for minimizing error that calculated as well as zero force.

With these two results compare, the bit error performance for SISO is better when using least square estimation. Similarly, MISO system for least square is better than MISO in zero force estimation even though it is only slightly better. In addition, bit error performance when using least square is lower than using zero force in all of the antenna systems which resulted in better performance and less error. This is because zero force neglected the noise in the system.

Then, the wireless channel model changed to Rayleigh with comparing least square and zero force estimation. The parameters that used were vector of path delays [0 2.5e-8 3e-8] and vector of average path gain [0 -10 -20]. Figure (17) expressed the simulation of this distribution.

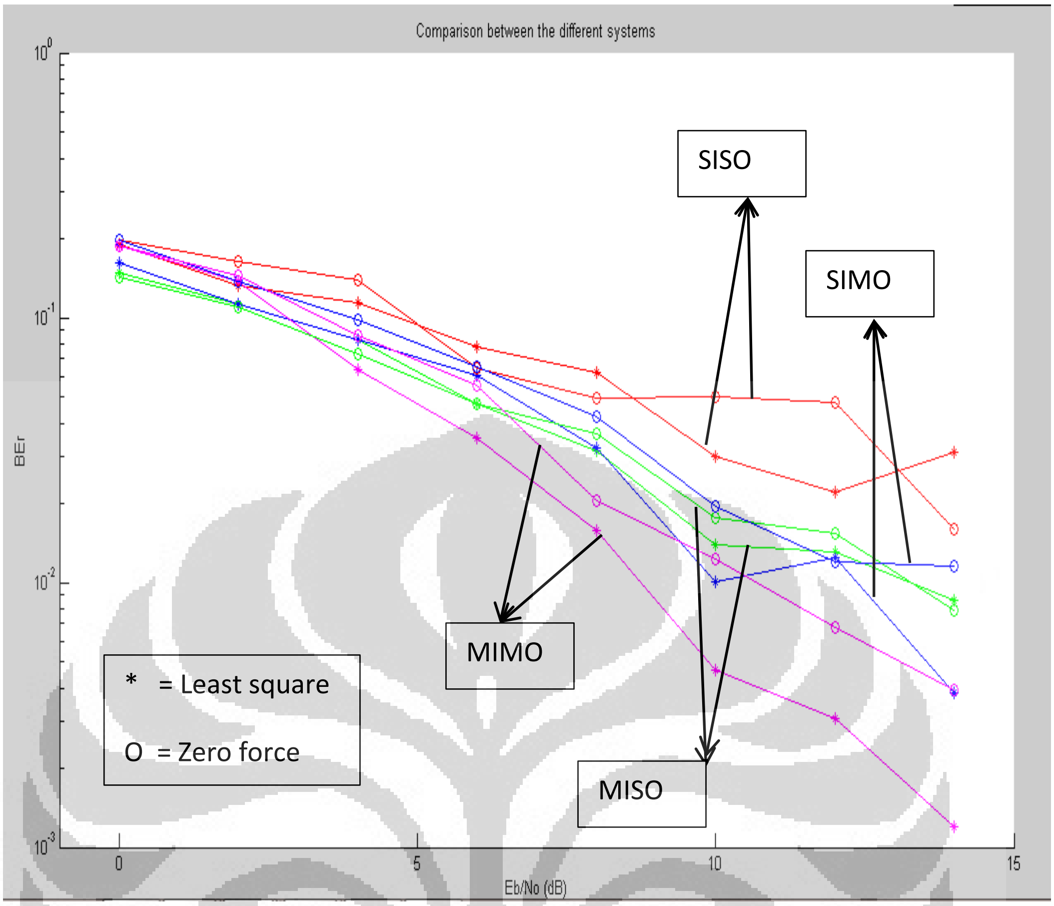

Figure 20: BEr performance of least square and zero force estimation in Rayleigh multipath

Figure (17) illustrate that MIMO performs the best of all with followed by SIMO, MISO, and SISO respectively. Because with multipath channel, MIMO will perform the best as the process will be optimal rather than others.

When the simulation changed to Rayleigh multipath, the performance for least square estimation is still slightly better than zero force, except the SISO and MISO systems. Due to SISO performs worst in multipath channel, it will also affecting the performance of least square. But this kind of situation was not happened in zero force, because zero force can cancelled the ISI that occurs in the system. Overall, the bit error rate for Rayleigh single path is preferable rather than Rayleigh multipath which has higher error ratio. This is possibly because of multipath parameter that being defined at the initial part.

Next, the simulation changed to Ricean channel with Kfactor of 2 and additive white Gaussian noise. The simulation with Ricean channel expressed in figure (18).

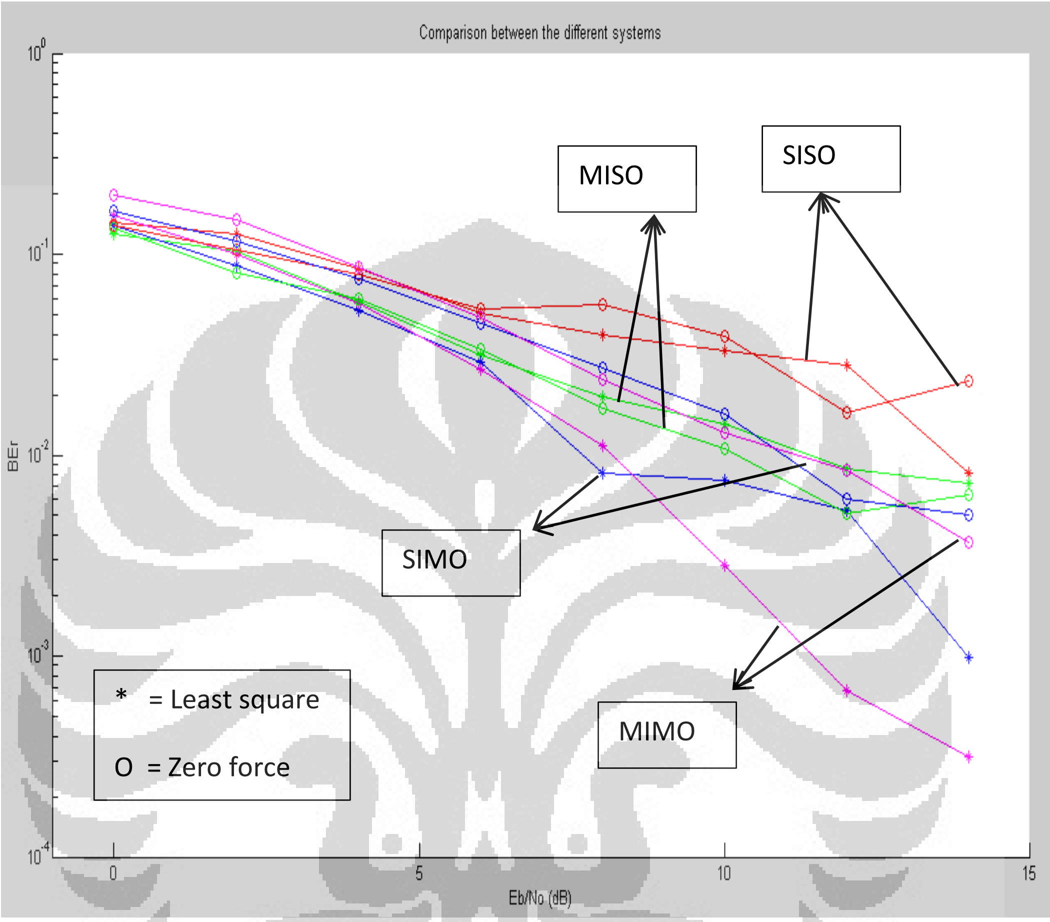

Figure 21: BEr performance of least square and zero force estimation in Ricean channel

As shown at the figure (18), Ricean channel performs less error for MIMO than with Rayleigh of multipath channel at high SNR. The MIMO performed better than others which are correct referring to theory. It also gives better view on the graph that differentiates with each of antenna systems.

From figure (18), almost all antenna types in least square show better bit error performance, except in MISO which is slightly poor than zero force estimation method. This will be the second times that zero force performs better than least square in MISO system, thus considering zero force is better with MISO system.

As it seen in figure (17) to (19), it can be concluded that least square estimation is much better than zero force in terms of the error performance. Also it can be seen that by using Rayleigh single path and Rician channel, the performance of bit error rate were a lot better than using Rayleigh multipath. This could be because of Rayleigh multipath is non-line of sight which means there might be some loss when transmitting the data. Based on simulation results, it seems that the performance of least square estimation for MIMO and SIMO in all three different channels were better than using zero force. However, it occurs that zero force performs a little better in Rayleigh multipath and Rician except for Rayleigh single path in MISO system. In addition, SISO also has the same condition when performing with Rayleigh multipath. These situations can conclude that zero force is performing better with MISO. But, overall least square were preferable. It is due to zero force algorithms have the disadvantage that it might excessively amplify noise at frequencies. Therefore, least square mostly performs much better with having many probability of correct decision than zero force method when it comes to bit error rate.

CHAPTER V

CONCLUSION

In conclusion, this research illustrates the degree of knowledge and comprehends the channel estimation method as well as the wireless communication that have been obtained through exploration and simulations. The performances of least square and zero force estimation methods for multiple antenna systems also have been attained.

Finally, the results in this research showed that the least square were performed better than the zero force. When the least square performed, it displayed that the bit error rate were more stable and better than zero force, hence lead to achieve almost perfect estimation. This is due to zero force usually neglect the noise altogether which might excessively amplify noise at frequencies. In addition, it also proved the theory which zero force is usually worst at wireless communication. In the end, least square estimation was proven better when doing the channel estimation rather than zero force.

BIBLIOGRAPHY

[1] Moghe, S.; Upadhyay, R.; , "Single input single output technology in 802.11n," *Emerging Trends in Electronic and Photonic Devices & Systems, 2009. ELECTRO '09. International Conference on* , vol., no., pp.247-249, 22-24 Dec. 2009 URL: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5441124&isnumber=5441033

[2] Khan, A. Vesilo, R. "A Tutorial on SISO and MIMO Channel Capacities." Accessed June 23th 2011. http://read.pudn.com/downloads110/doc/454417/capacity.pdf

[3] Searchmobilecomputing. "MISO (multiple input, single output)." Last modified November 2004. http://searchmobilecomputing.techtarget.com/definition/MISO

[4] Conniq. "Introduction to Multiple Antenna Systems: SIMO, MISO, MIMO." Accessed June 24th 2011. http://www.conniq.com/WiMAX/mimo-02.htm

[5] Miao, George J. 2007. Signal Processing in Digital Communications. 1st ed. Norwood : ARTECH HOUSE, INC

[6] Tse, D. Viswanath, P. "Fundamental of Wireless Communications." Last modified August 13rd 2004. http://www.eecs.berkeley.edu/~dtse/Chapters_PDF/Fundamentals_Wireless_Communication_chapter5.pdf

[7] Searchmobilecomputing. "SIMO (single input, multiple output)." Last modified November 2004. http://searchmobilecomputing.techtarget.com/definition/MISO

[8] Kassner, M. "802.11n, MIMO, and multipath environments." Last modified November 12th 2007. http://www.techrepublic.com/blog/wireless/80211n-mimo-and-multipath-environments/156

[9] Sankar, K."Alamouti STBC." Last modified October 16th 2008. http://www.dsplog.com/2008/10/16/alamouti-stbc/

[10] Sankar, K."Alamouti STBC with 2 receive antenna." Last modified March 15th 2009. http://www.dsplog.com/2009/03/15/alamouti-stbc-2-receive-antenna/

[11] Rappaport, Theodore S. 2002. *Wireless communications principles and practice*. 2nd ed. New Jersey: Prenctice Hall PTR.

[12]Pukkila, M. 2000. "Channel Estimation Modeling" In Postgraduate Course in Radiocommunications by Markku Pukkila. 1-15. Nokia Research Center. Accessed March 30th, 2011. http://www.comlab.hut.fi/opetus/260/chan_est.pdf

[13]E. Björnson, B. Ottersten, A Framework for Training-Based Estimation in Arbitrarily Correlated Rician MIMO Channels with Rician Disturbance, IEEE Transactions on Signal Processing, vol 58, pp. 1807-1820, 2010.

[14] Andrea Goldsmith. 2005. "Maximum Likelihood Sequence Estimation". *Wireless Communications*. Cambridge University Press

APPENDIXES

Least square with Rayleigh single path

```
clc;
clear;
frmLen = 192; % frame length
maxNumPackets = 100; % maximum number of packets
EbNo = 0:2:14; % Eb/No varying to 12 dB
N = 2; % number of Tx antennas
M = 2; % number of Rx antennas
pLen = 8; % number of pilot symbols per frame
W = hadamard(pLen);
pilots1 = W(:, 1);
pilots2 = W(:, 1:N); % orthogonal set per transmit antenna
h = gcf;
P = 2; % Modulation Order
T = 0.000000001;
FD= 100; % Doppler Shift

% Create BPSK mod-demod objects
bpskmod = modem.pskmod('M', P, 'SymbolOrder', 'Gray');
bpskdemod = modem.pskdemod(bpskmod);

% Pre-allocate variables for speed
tx1=zeros(frmLen,1) ; s1=zeros(frmLen/2,1) ; s2=s1 ; tx2=zeros(frmLen,2);
trans1=zeros(pLen+frmLen,1) ; trans2=zeros(pLen+frmLen,2) ;
r1by1=zeros(pLen+frmLen,1) ; r1by1_noisy=r1by1 ; H1by1=zeros(frmLen,1);
r1by2=zeros(pLen+frmLen,2) ; r1by2_noisy=r1by2 ; H1by2=zeros(frmLen,2);
z12=zeros(frmLen,2) ; z12_2=zeros(frmLen,1);
r2by1=zeros(pLen+frmLen,1) ; r2by1_noisy=r2by1 ; H2by1=zeros(frmLen,2);
z21_1=zeros(frmLen/2,1) ; z21_2=z21_1 ; z21=zeros(frmLen,1);
r2by2=zeros(pLen+frmLen,2) ; r2by2_noisy=r2by2 ; H2by2=zeros(frmLen,2,2);
z22_1=zeros(frmLen/2,2) ; z22_2=z22_1 ; z22=zeros(frmLen,2);
BER1by1=zeros(1,length(EbNo)); BER2by1 = BER1by1 ; BER1by2 = BER1by1 ; BER2by2 =
BER1by1;

% Set up a figure for visualizing BEr results
clf(h);
grid on;
hold on;
%set(gca,'yscale','log','xlim',[EbNo(1)-1 , EbNo(end)+1],'ylim',[1e-7 1]);
xlabel('Eb/No (dB)');
ylabel('BEr');
set(h,'NumberTitle','off');
set(h,'Name','Orthogonal Space-Time Block Coding');
set(h, 'renderer', 'zbuffer'); title('Comparison between the different systems');

% Loop over several EbNo points
for idx = 1:length(EbNo)
SNR = EbNo(idx)

% Loop over the number of packets
for packetIdx = 1:maxNumPackets
data = randint(frmLen, 1, P); % Generating Random Data (0 or 1)
```

```
bpskmod = modem.pskmod('M', P, 'SymbolOrder', 'Gray');
tx1 = modulate(bpskmod, data); % BPSK modulation

% Alamouti Encoder, G2
% X = [s1 s2; -s2* s1*]
s1 = tx1(1:N:end); s2 = tx1(2:N:end);
tx2(1:2:end, :) = [s1 s2];
tx2(2:2:end, :) = [-conj(s2) conj(s1)];

% Prepend pilot symbols for each frame
trans1 = [pilots1; tx1];
trans2 = [pilots2; tx2];

% For SISO System (1x1):
chan1by1 = mimochan (1,1,T,FD);
r1by1 = filter(chan1by1, trans1);
r1by1_noisy = awgn(r1by1, EbNo(idx));

% Channel Estimation
H1by1(1, 1) = (r1by1_noisy(1:pLen,1).' * pilots1(:, 1)./pLen); % Least Square Channel
Estimation

% assume held constant for the whole frame
H1by1 = H1by1(ones(frmLen, 1),1);

% Determine errors
demod1by1 = demodulate(bpskdemod, (r1by1_noisy(pLen+1:end,:).*conj(H1by1)));
error1by1(packetIdx) = biterr(demod1by1, data);

% For SIMO System (1x2):
chan1by2 = mimochan (1,2,T,FD);
r1by2 = filter (chan1by2, trans1);
r1by2_noisy = awgn(r1by2,EbNo(idx));

% Channel Estimation
for n = 1:2
H1by2(1, n) = (r1by2_noisy(1:pLen, n).' * pilots1(:, :) ./pLen); % Least Square Channel
Estimation
end

% assume held constant for the whole frame
H1by2 = H1by2(ones(frmLen, 1),:);

% Determine Errrors
z12=r1by2_noisy((pLen+1:end),:).* conj(H1by2);
z12_2= sum(z12,2);
demod1by2 = demodulate(bpskdemod,z12_2);
error1by2(packetIdx) = biterr(demod1by2, data);

% For MISO System (2x1):
chan2by1 = mimochan (2,1,T,FD);
r2by1 = filter (chan2by1 , (trans2./sqrt(2)));
r2by1_noisy = awgn(r2by1,EbNo(idx));

% Channel Estimation
```

```
for n = 1:2
H2by1(1, n) = (r2by1_noisy(1:pLen, :).' * pilots2(:, n) ./pLen); % Least Square Channel
Estimation
end

% assume held constant for the whole frame
H2by1 = H2by1(ones(frmLen, 1), :);

hidx = 1:2:length(H2by1);
z21_1 = r2by1_noisy(pLen+1:2:end).*conj(H2by1(hidx,1))+ ....
conj(r2by1_noisy(pLen+2:2:end)).*H2by1(hidx,2);
z21_2 = r2by1_noisy(pLen+1:2:end).*conj(H2by1(hidx,2))- ....
conj(r2by1(pLen+2:2:end)).*H2by1(hidx,1);
z21(1:2:end)= z21_1;
z21(2:2:end) = z21_2;

% Determine Errors
demod2by1 = demodulate(bpskdemod, z21);
error2by1 (packetIdx) = biterr(demod2by1, data);

% For MIMO System (2x2)
chan2by2 = mimochan (2,2,T,FD);
r2by2 = filter (chan2by2 , (trans2./sqrt(2)));
r2by2_noisy = awgn(r2by2,EbNo(idx));

% Channel Estimation
for n = 1:2
H2by2(1, n, :) = (r2by2_noisy(1:pLen, :).' * pilots2(:, n) ./pLen); % Least Square Channel
Estimation
end

% assume held constant for the whole frame
H2by2 = H2by2(ones(frmLen, 1), :, :);

% Combiner using estimated channel
heidx = 1:2:length(H2by2);
for i = 1:2
z22_1(:, i) = r2by2_noisy(pLen+1:2:end, i).* conj(H2by2(heidx, 1, i)) + ...
conj(r2by2_noisy(pLen+2:2:end, i)).* H2by2(heidx, 2, i);

z22_2(:, i) = r2by2_noisy(pLen+1:2:end, i).* conj(H2by2(heidx, 2, i)) - ...
conj(r2by2_noisy(pLen+2:N:end, i)).* H2by2(heidx, 1, i);
end
z22(1:2:end, :) = z22_1 ; z22(2:2:end, :) = z22_2;

% Detemine Errors
demod2by2 = demodulate(bpskdemod, sum(z22, 2));
error2by2(packetIdx) = biterr(demod2by2, data);

end

% Calculate BER for current idx
BER1by1(idx) = sum(error1by1)/(maxNumPackets*frmLen);
BER1x1 = BER1by1(idx)
```

```
BER1by2(idx) = sum(error1by2)/(maxNumPackets*frmLen);
BER1x2 = BER1by2(idx)

BER2by1(idx) = sum(error2by1)/(maxNumPackets*frmLen);
BER2x1 = BER2by1(idx)

BER2by2(idx) = sum(error2by2)/(maxNumPackets*frmLen);
BER2x2 = BER2by2(idx)

% Plot the results
semilogy(EbNo(1:idx), BER1by1(1:idx), 'r*');
% drawnow;
semilogy(EbNo(1:idx), BER2by1(1:idx), 'go');
%drawnow;
semilogy(EbNo(1:idx), BER1by2(1:idx), 'b+');
%drawnow;
semilogy(EbNo(1:idx), BER2by2(1:idx), 'm');
legend('SISO System (1Tx, 1Rx)','MISO System (2Tx, 1Rx)','SIMO System (1Tx, 2Rx)','MIMO
System (2Tx,2Rx)');
drawnow;
end % end of for loop for EbNo

semilogy(EbNo,BER1by1,'r', EbNo,BER2by1,'g', EbNo,BER1by2,'b', EbNo,BER2by2,'m');
hold off;
```

Least square with Rayleigh multipath channel

```
clc;
clear;
frmLen = 192; % frame length
maxNumPackets = 100; % maximum number of packets
EbNo = 0:2:14; % Eb/No varying to 12 dB
N = 2; % number of Tx antennas
M = 2; % number of Rx antennas
pLen = 8; % number of pilot symbols per frame
W = hadamard(pLen);
pilots1 = W(:, 1);
pilots2 = W(:, 1:N); % orthogonal set per transmit antenna
h = gcf;
P = 2; % Modulation Order
T = 0.000000001;
FD= 100; % Doppler Shift
tau = [0 2.5e-8 3e-8];
pdb = [0 -10 -20];
% Create BPSK mod-demod objects
bpskmod = modem.pskmod('M', P, 'SymbolOrder', 'Gray');
bpskdemod = modem.pskdemod(bpskmod);

% Pre-allocate variables for speed
tx1=zeros(frmLen,1) ; s1=zeros(frmLen/2,1) ; s2=s1 ; tx2=zeros(frmLen,2);
trans1=zeros(pLen+frmLen,1) ; trans2=zeros(pLen+frmLen,2) ;
r1by1=zeros(pLen+frmLen,1) ; r1by1_noisy=r1by1 ; H1by1=zeros(frmLen,1);
r1by2=zeros(pLen+frmLen,2) ; r1by2_noisy=r1by2 ; H1by2=zeros(frmLen,2);
z12=zeros(frmLen,2) ; z12_2=zeros(frmLen,1);
```

```
r2by1=zeros(pLen+frmLen,1) ; r2by1_noisy=r2by1 ; H2by1=zeros(frmLen,2);
z21_1=zeros(frmLen/2,1) ; z21_2=z21_1 ; z21=zeros(frmLen,1);
r2by2=zeros(pLen+frmLen,2) ; r2by2_noisy=r2by2 ; H2by2=zeros(frmLen,2,2);
z22_1=zeros(frmLen/2,2) ; z22_2=z22_1 ; z22=zeros(frmLen,2);
BER1by1=zeros(1,length(EbNo)); BER2by1 = BER1by1 ; BER1by2 = BER1by1 ; BER2by2 =
BER1by1;

% Set up a figure for visualizing BEr results
clf(h);
grid on;
hold on;
% set(gca,'yscale','log','xlim',[EbNo(1)-1 , EbNo(end)+1],'ylim',[1e-7 1]);
xlabel('Eb/No (dB)');
ylabel('BEr');
set(h,'NumberTitle','off');
set(h,'Name','Orthogonal Space-Time Block Coding');
set(h, 'renderer', 'zbuffer'); title('Comparison between the different systems');

% Loop over several EbNo points
for idx = 1:length(EbNo)
SNR = EbNo(idx)

% Loop over the number of packets
for packetIdx = 1:maxNumPackets
data = randint(frmLen, 1, P); % Generating Random Data (0 or 1)
bpskmod = modem.pskmod('M', P, 'SymbolOrder', 'Gray');
tx1 = modulate(bpskmod, data); % BPSK modulation

% Alamouti Encoder, G2
% X = [s1 s2; -s2* s1*]
s1 = tx1(1:N:end); s2 = tx1(2:N:end);
tx2(1:2:end, :) = [s1 s2];
tx2(2:2:end, :) = [-conj(s2) conj(s1)];

% Prepend pilot symbols for each frame
trans1 = [pilots1; tx1];
trans2 = [pilots2; tx2];

% For SISO System (1x1):
chan1by1 = mimochan (1,1,T,FD,tau,pdb);
r1by1 = filter(chan1by1, trans1);
r1by1_noisy = awgn(r1by1, EbNo(idx));

% Channel Estimation
H1by1(1, 1) = (r1by1_noisy(1:pLen,1).' * pilots1(:, 1) ./pLen); % Least Square Channel
Estimation

% assume held constant for the whole frame
H1by1 = H1by1(ones(frmLen, 1),1);

% Determine errors
demod1by1 = demodulate(bpskdemod, (r1by1_noisy(pLen+1:end,:).*conj(H1by1)));
error1by1(packetIdx) = biterr(demod1by1, data);

% For SIMO System (1x2):
chan1by2 = mimochan (1,2,T,FD,tau,pdb);
```

```
r1by2 = filter (chan1by2, trans1);
r1by2_noisy = awgn(r1by2,EbNo(idx));

% Channel Estimation
for n = 1:2
H1by2(1, n) = (r1by2_noisy(1:pLen, n).' * pilots1(:, :) ./pLen); % Least Square Channel
Estimation
end

% assume held constant for the whole frame
H1by2 = H1by2(ones(frmLen, 1),:);

% Determine Errrors
z12=r1by2_noisy((pLen+1:end),:).* conj(H1by2);
z12_2= sum(z12,2);
demod1by2 = demodulate(bpskdemod,z12_2);
error1by2(packetIdx) = biterr(demod1by2, data);

% For MISO System (2x1):
chan2by1 = mimochan (2,1,T,FD,tau,pdb);
r2by1 = filter (chan2by1 , (trans2./sqrt(2)));
r2by1_noisy = awgn(r2by1,EbNo(idx));

% Channel Estimation
for n = 1:2
H2by1(1, n) = (r2by1_noisy(1:pLen, :).' * pilots2(:, n) ./pLen); % Least Square Channel
Estimation
end

% assume held constant for the whole frame
H2by1 = H2by1(ones(frmLen, 1), :);

hidx = 1:2:length(H2by1);
z21_1 = r2by1_noisy(pLen+1:2:end).*conj(H2by1(hidx,1))+ ....
conj(r2by1_noisy(pLen+2:2:end)).*H2by1(hidx,2);
z21_2 = r2by1_noisy(pLen+1:2:end).*conj(H2by1(hidx,2))- ....
conj(r2by1(pLen+2:2:end)).*H2by1(hidx,1);
z21(1:2:end)= z21_1;
z21(2:2:end) = z21_2;

% Determine Errors
demod2by1 = demodulate(bpskdemod, z21);
error2by1 (packetIdx) = biterr(demod2by1, data);

% For MIMO System (2x2)
chan2by2 = mimochan (2,2,T,FD,tau,pdb);
r2by2 = filter (chan2by2 , (trans2./sqrt(2)));
r2by2_noisy = awgn(r2by2,EbNo(idx));

% Channel Estimation
for n = 1:2
H2by2(1, n, :) = (r2by2_noisy(1:pLen, :).' * pilots2(:, n) ./pLen); % Least Square Channel
Estimation
end
```

```
% assume held constant for the whole frame
H2by2 = H2by2(ones(frmLen, 1), :, :);

% Combiner using estimated channel
heidx = 1:2:length(H2by2);
for i = 1:2
z22_1(:, i) = r2by2_noisy(pLen+1:2:end, i).* conj(H2by2(heidx, 1, i)) + ...
conj(r2by2_noisy(pLen+2:2:end, i)).* H2by2(heidx, 2, i);

z22_2(:, i) = r2by2_noisy(pLen+1:2:end, i).* conj(H2by2(heidx, 2, i)) - ...
conj(r2by2_noisy(pLen+2:N:end, i)).* H2by2(heidx, 1, i);
end
z22(1:2:end, :) = z22_1 ; z22(2:2:end, :) = z22_2;

% Detemine Errors
demod2by2 = demodulate(bpskdemod, sum(z22, 2));
error2by2(packetIdx) = biterr(demod2by2, data);

end

% Calculate BER for current idx
BER1by1(idx) = sum(error1by1)/(maxNumPackets*frmLen);
BER1x1 = BER1by1(idx)

BER1by2(idx) = sum(error1by2)/(maxNumPackets*frmLen);
BER1x2 = BER1by2(idx)

BER2by1(idx) = sum(error2by1)/(maxNumPackets*frmLen);
BER2x1 = BER2by1(idx)

BER2by2(idx) = sum(error2by2)/(maxNumPackets*frmLen);
BER2x2 = BER2by2(idx)

% Plot the results
semilogy(EbNo(1:idx), BER1by1(1:idx), 'r*');
% drawnow;
semilogy(EbNo(1:idx), BER2by1(1:idx), 'go');
%drawnow;
semilogy(EbNo(1:idx), BER1by2(1:idx), 'b+');
%drawnow;
semilogy(EbNo(1:idx), BER2by2(1:idx), 'm');
legend('SISO System (1Tx, 1Rx)','MISO System (2Tx, 1Rx)','SIMO System (1Tx, 2Rx)','MIMO
System (2Tx,2Rx)');
drawnow;
end % end of for loop for EbNo

semilogy(EbNo,BER1by1,'r', EbNo,BER2by1,'g', EbNo,BER1by2,'b', EbNo,BER2by2,'m');
hold off;
```

Least square with Ricean fading channel

```
clc;
clear;
```

```
frmLen = 192; % frame length
maxNumPackets = 100; % maximum number of packets
EbNo = 0:2:14; % Eb/No varying to 12 dB
N = 2; % number of Tx antennas
M = 2; % number of Rx antennas
pLen = 8; % number of pilot symbols per frame
W = hadamard(pLen);
pilots1 = W(:, 1);
pilots2 = W(:, 1:N); % orthogonal set per transmit antenna
h = gcf;
P = 2; % Modulation Order
T = 0.000000001;
FD= 100; % Doppler Shift
tau = [0 2.5e-8 3e-8];
pdb = [0 -10 -20];
% Create BPSK mod-demod objects
bpskmod = modem.pskmod('M', P, 'SymbolOrder', 'Gray');
bpskdemod = modem.pskdemod(bpskmod);

% Pre-allocate variables for speed
tx1=zeros(frmLen,1) ; s1=zeros(frmLen/2,1) ; s2=s1 ; tx2=zeros(frmLen,2);
trans1=zeros(pLen+frmLen,1) ; trans2=zeros(pLen+frmLen,2) ;
r1by1=zeros(pLen+frmLen,1) ; r1by1_noisy=r1by1 ; H1by1=zeros(frmLen,1);
r1by2=zeros(pLen+frmLen,2) ; r1by2_noisy=r1by2 ; H1by2=zeros(frmLen,2);
z12=zeros(frmLen,2) ; z12_2=zeros(frmLen,1);
r2by1=zeros(pLen+frmLen,1) ; r2by1_noisy=r2by1 ; H2by1=zeros(frmLen,2);
z21_1=zeros(frmLen/2,1) ; z21_2=z21_1 ; z21=zeros(frmLen,1);
r2by2=zeros(pLen+frmLen,2) ; r2by2_noisy=r2by2 ; H2by2=zeros(frmLen,2,2);
z22_1=zeros(frmLen/2,2) ; z22_2=z22_1 ; z22=zeros(frmLen,2);
BER1by1=zeros(1,length(EbNo)); BER2by1 = BER1by1 ; BER1by2 = BER1by1 ; BER2by2 =
BER1by1;

% Set up a figure for visualizing BEr results
clf(h);
grid on;
hold on;
% set(gca,'yscale','log','xlim',[EbNo(1)-1 , EbNo(end)+1],'ylim',[1e-7 1]);
xlabel('Eb/No (dB)');
ylabel('BEr');
set(h,'NumberTitle','off');
set(h,'Name','Orthogonal Space-Time Block Coding');
set(h, 'renderer', 'zbuffer'); title('Comparison between the different systems');

% Loop over several EbNo points
for idx = 1:length(EbNo)
SNR = EbNo(idx)

% Loop over the number of packets
for packetIdx = 1:maxNumPackets
data = randint(frmLen, 1, P); % Generating Random Data (0 or 1)
bpskmod = modem.pskmod('M', P, 'SymbolOrder', 'Gray');
tx1 = modulate(bpskmod, data); % BPSK modulation

% Alamouti Encoder, G2
% X = [s1 s2; -s2* s1*]
s1 = tx1(1:N:end); s2 = tx1(2:N:end);
tx2(1:2:end, :) = [s1 s2];
```

```
tx2(2:2:end, :) = [-conj(s2) conj(s1)];

% Prepend pilot symbols for each frame
trans1 = [pilots1; tx1];
trans2 = [pilots2; tx2];

% For SISO System (1x1):
chan1by1 = mimochan (1,1,T,FD,tau,pdb);
chan1by1.KFactor = 2;
r1by1 = filter(chan1by1, trans1);
r1by1_noisy = awgn(r1by1, EbNo(idx));

% Channel Estimation
H1by1(1, 1) = (r1by1_noisy(1:pLen,1).' * pilots1(:, 1) ./pLen); % Least Square Channel
Estimation

% assume held constant for the whole frame
H1by1 = H1by1(ones(frmLen, 1),1);

% Determine errors
demod1by1 = demodulate(bpskdemod, (r1by1_noisy(pLen+1:end,:).*conj(H1by1)));
error1by1(packetIdx) = biterr(demod1by1, data);

% For SIMO System (1x2):
chan1by2 = mimochan (1,2,T,FD,tau,pdb);
chan1by2.KFactor = 2;
r1by2 = filter (chan1by2, trans1);
r1by2_noisy = awgn(r1by2,EbNo(idx));

% Channel Estimation
for n = 1:2
H1by2(1, n) = (r1by2_noisy(1:pLen, n).' * pilots1(:, :) ./pLen); % Least Square Channel
Estimation
end

% assume held constant for the whole frame
H1by2 = H1by2(ones(frmLen, 1),:);

% Determine Errrors
z12=r1by2_noisy((pLen+1:end),:).* conj(H1by2);
z12_2= sum(z12,2);
demod1by2 = demodulate(bpskdemod,z12_2);
error1by2(packetIdx) = biterr(demod1by2, data);

% For MISO System (2x1):
chan2by1 = mimochan (2,1,T,FD,tau,pdb);
chan2by1.KFactor = 2;
r2by1 = filter (chan2by1 , (trans2./sqrt(2)));
r2by1_noisy = awgn(r2by1,EbNo(idx));

% Channel Estimation
for n = 1:2
H2by1(1, n) = (r2by1_noisy(1:pLen, :).' * pilots2(:, n) ./pLen); % Least Square Channel
Estimation
end
```

```
% assume held constant for the whole frame
H2by1 = H2by1(ones(frmLen, 1), :);

hidx = 1:2:length(H2by1);
z21_1 = r2by1_noisy(pLen+1:2:end).*conj(H2by1(hidx,1))+ ....
conj(r2by1_noisy(pLen+2:2:end)).*H2by1(hidx,2);
z21_2 = r2by1_noisy(pLen+1:2:end).*conj(H2by1(hidx,2))- ....
conj(r2by1(pLen+2:2:end)).*H2by1(hidx,1);
z21(1:2:end)= z21_1;
z21(2:2:end) = z21_2;

% Determine Errors
demod2by1 = demodulate(bpskdemod, z21);
error2by1 (packetIdx) = biterr(demod2by1, data);

% For MIMO System (2x2)
chan2by2 = mimochan (2,2,T,FD,tau,pdb);
chan2by2.KFactor = 2;
r2by2 = filter (chan2by2 , (trans2./sqrt(2)));
r2by2_noisy = awgn(r2by2,EbNo(idx));

% Channel Estimation
for n = 1:2
H2by2(1, n, :) = (r2by2_noisy(1:pLen, :).' * pilots2(:, n) ./pLen); % Least Square Channel
Estimation
end

% assume held constant for the whole frame
H2by2 = H2by2(ones(frmLen, 1), :, :);

% Combiner using estimated channel
heidx = 1:2:length(H2by2);
for i = 1:2
z22_1(:, i) = r2by2_noisy(pLen+1:2:end, i).* conj(H2by2(heidx, 1, i)) + ...
conj(r2by2_noisy(pLen+2:2:end, i)).* H2by2(heidx, 2, i);

z22_2(:, i) = r2by2_noisy(pLen+1:2:end, i).* conj(H2by2(heidx, 2, i)) - ...
conj(r2by2_noisy(pLen+2:N:end, i)).* H2by2(heidx, 1, i);
end
z22(1:2:end, :) = z22_1 ; z22(2:2:end, :) = z22_2;

% Detemine Errors
demod2by2 = demodulate(bpskdemod, sum(z22, 2));
error2by2(packetIdx) = biterr(demod2by2, data);

end


% Calculate BER for current idx
BER1by1(idx) = sum(error1by1)/(maxNumPackets*frmLen);
BER1x1 = BER1by1(idx)

BER1by2(idx) = sum(error1by2)/(maxNumPackets*frmLen);
BER1x2 = BER1by2(idx)
```

```
BER2by1(idx) = sum(error2by1)/(maxNumPackets*frmLen);
BER2x1 = BER2by1(idx)

BER2by2(idx) = sum(error2by2)/(maxNumPackets*frmLen);
BER2x2 = BER2by2(idx)

% Plot the results
semilogy(EbNo(1:idx), BER1by1(1:idx), 'r*');
% drawnow;
semilogy(EbNo(1:idx), BER2by1(1:idx), 'go');
%drawnow;
semilogy(EbNo(1:idx), BER1by2(1:idx), 'b+');
%drawnow;
semilogy(EbNo(1:idx), BER2by2(1:idx), 'm');
legend('SISO System (1Tx, 1Rx)','MISO System (2Tx, 1Rx)','SIMO System (1Tx, 2Rx)','MIMO
System (2Tx,2Rx)');
drawnow;
end % end of for loop for EbNo

semilogy(EbNo,BER1by1,'r', EbNo,BER2by1,'g', EbNo,BER1by2,'b', EbNo,BER2by2,'m');
hold off;
```

Zero-Forcing with Rayleigh single path

```
clc;
clear;
frmLen = 192; % frame length
maxNumPackets = 100; % maximum number of packets
EbNo = 0:2:14; % Eb/No varying to 12 dB
N = 2; % number of Tx antennas
M = 2; % number of Rx antennas
pLen = 8; % number of pilot symbols per frame
W = hadamard(pLen);
pilots1 = W(:, 1);
pilots2 = W(:, 1:N); % orthogonal set per transmit antenna
h = gcf;
P = 2; % Modulation Order
T = 0.000000001;
FD= 100; % Doppler Shift
% Create BPSK mod-demod objects
bpskmod = modem.pskmod('M', P, 'SymbolOrder', 'Gray');
bpskdemod = modem.pskdemod(bpskmod);

% Pre-allocate variables for speed
tx1=zeros(frmLen,1) ; s1=zeros(frmLen/2,1) ; s2=s1 ; tx2=zeros(frmLen,2);
trans1=zeros(pLen+frmLen,1) ; trans2=zeros(pLen+frmLen,2) ;
r1by1=zeros(pLen+frmLen,1) ; r1by1_noisy=r1by1 ; H1by1=zeros(frmLen,1);
r1by2=zeros(pLen+frmLen,2) ; r1by2_noisy=r1by2 ; H1by2=zeros(frmLen,2);
z12=zeros(frmLen,2) ; z12_2=zeros(frmLen,1);
r2by1=zeros(pLen+frmLen,1) ; r2by1_noisy=r2by1 ; H2by1=zeros(frmLen,2);
z21_1=zeros(frmLen/2,1) ; z21_2=z21_1 ; z21=zeros(frmLen,1);
r2by2=zeros(pLen+frmLen,2) ; r2by2_noisy=r2by2 ; H2by2=zeros(frmLen,2,2);
z22_1=zeros(frmLen/2,2) ; z22_2=z22_1 ; z22=zeros(frmLen,2);
```

```
BER1by1=zeros(1,length(EbNo)); BER2by1 = BER1by1 ; BER1by2 = BER1by1 ; BER2by2 =
BER1by1;

% Set up a figure for visualizing BEr results
clf(h);
grid on;
hold on;
%set(gca,'yscale','log','xlim',[EbNo(1)-1 , EbNo(end)+1],'ylim',[1e-7 1]);
xlabel('Eb/No (dB)');
ylabel('BEr');
set(h,'NumberTitle','off');
set(h,'Name','Orthogonal Space-Time Block Coding');
set(h, 'renderer', 'zbuffer'); title('Comparision between the diffirent systems');

% Loop over several EbNo points
for idx = 1:length(EbNo)
SNR = EbNo(idx)

% Loop over the number of packets
for packetIdx = 1:maxNumPackets
data = randint(frmLen, 1, P); % Generating Random Data (0 or 1)
bpskmod = modem.pskmod('M', P, 'SymbolOrder', 'Gray');
tx1 = modulate(bpskmod, data); % BPSK modulation

% Alamouti Encoder, G2
% X = [s1 s2; -s2* s1*]
s1 = tx1(1:N:end); s2 = tx1(2:N:end);
tx2(1:2:end, :) = [s1 s2];
tx2(2:2:end, :) = [-conj(s2) conj(s1)];

% Prepend pilot symbols for each frame
trans1 = [pilots1; tx1];
trans2 = [pilots2; tx2];

% For SISO System (1x1):
chan1by1 = mimochan (1,1,T,FD);
r1by1 = filter(chan1by1, trans1);
r1by1_noisy = awgn(r1by1, EbNo(idx));
Z1 = pinv(r1by1_noisy);
r7 = Z1';
% Channel Estimation
H1by1(1, 1) = (Z1(1:pLen). ' * pilots1(:, 1) ./pLen); % Least Square Channel Estimation

% assume held constant for the whole frame
H1by1 = H1by1(ones(frmLen, 1),1);

% Determine errors
demod1by1 = demodulate(bpskdemod, (r1by1_noisy(pLen+1:end,:).*conj(H1by1)));
error1by1(packetIdx) = biterr(demod1by1, data);

% For SIMO System (1x2):
chan1by2 = mimochan (1,2,T,FD);
r1by2 = filter (chan1by2, trans1);
r1by2_noisy = awgn(r1by2,EbNo(idx));
Z2 = pinv(r1by2_noisy);
r1 = Z2';
```

```
% Channel Estimation
for n = 1:2
H1by2(1, n) = (r1(1:pLen, n).' * pilots1(:, :)./pLen); % Least Square Channel Estimation
end

% assume held constant for the whole frame
H1by2 = H1by2(ones(frmLen, 1),:);

% Determine Errrors
z12=r1by2_noisy((pLen+1:end),:).* conj(H1by2);
z12_2= sum(z12,2);
demod1by2 = demodulate(bpskdemod,z12_2);
error1by2(packetIdx) = biterr(demod1by2, data);

% For MISO System (2x1):
chan2by1 = mimochan (2,1,T,FD);
r2by1 = filter (chan2by1 , (trans2./sqrt(2)));
r2by1_noisy = awgn(r2by1,EbNo(idx));
Z3 = pinv(r2by1_noisy);
r2 = Z3';
% Channel Estimation
for n = 1:2
H2by1(1, n) = (r2(1:pLen, :).' * pilots2(:, n)./pLen); % Least Square Channel Estimation
end

% assume held constant for the whole frame
H2by1 = H2by1(ones(frmLen, 1), :);

hidx = 1:2:length(H2by1);
z21_1 = r2by1_noisy(pLen+1:2:end).*conj(H2by1(hidx,1))+ ....
conj(r2by1_noisy(pLen+2:2:end)).*H2by1(hidx,2);
z21_2 = r2by1_noisy(pLen+1:2:end).*conj(H2by1(hidx,2))- ....
conj(r2by1(pLen+2:2:end)).*H2by1(hidx,1);
z21(1:2:end)= z21_1;
z21(2:2:end) = z21_2;

% Determine Errors
demod2by1 = demodulate(bpskdemod, z21);
error2by1 (packetIdx) = biterr(demod2by1, data);

% For MIMO System (2x2)
chan2by2 = mimochan (2,2,T,FD);
r2by2 = filter (chan2by2 , (trans2./sqrt(2)));
r2by2_noisy = awgn(r2by2,EbNo(idx));
Z4 = pinv(r2by2_noisy);
r3 = Z4';

% Channel Estimation
for n = 1:2
H2by2(1, n, :) = (r3(1:pLen, :).' * pilots2(:, n)./pLen); % Least Square Channel Estimation
end

% assume held constant for the whole frame
H2by2 = H2by2(ones(frmLen, 1), :, :);

% Combiner using estimated channel
```

```
heidx = 1:2:length(H2by2);
for i = 1:2
z22_1(:, i) = r2by2_noisy(pLen+1:2:end, i).* conj(H2by2(heidx, 1, i)) + ...
conj(r2by2_noisy(pLen+2:2:end, i)).* H2by2(heidx, 2, i);

z22_2(:, i) = r2by2_noisy(pLen+1:2:end, i).* conj(H2by2(heidx, 2, i)) - ...
conj(r2by2_noisy(pLen+2:N:end, i)).* H2by2(heidx, 1, i);
end
z22(1:2:end, :) = z22_1 ; z22(2:2:end, :) = z22_2;

% Detemine Errors
demod2by2 = demodulate(bpskdemod, sum(z22, 2));
error2by2(packetIdx) = biterr(demod2by2, data);

end

% Calculate BER for current idx
BER1by1(idx) = sum(error1by1)/(maxNumPackets*frmLen);
BER1x1 = BER1by1(idx)

BER1by2(idx) = sum(error1by2)/(maxNumPackets*frmLen);
BER1x2 = BER1by2(idx)

BER2by1(idx) = sum(error2by1)/(maxNumPackets*frmLen);
BER2x1 = BER2by1(idx)

BER2by2(idx) = sum(error2by2)/(maxNumPackets*frmLen);
BER2x2 = BER2by2(idx)

% Plot the results
semilogy(EbNo(1:idx), BER1by1(1:idx), 'r*');
% drawnow;
semilogy(EbNo(1:idx), BER2by1(1:idx), 'go');
%drawnow;
semilogy(EbNo(1:idx), BER1by2(1:idx), 'b+');
%drawnow;
semilogy(EbNo(1:idx), BER2by2(1:idx), 'm');
legend('SISO System (1Tx, 1Rx)','MISO System (2Tx, 1Rx)','SIMO System (1Tx, 2Rx)','MIMO
System (2Tx,2Rx)');
drawnow;
end % end of for loop for EbNo

semilogy(EbNo,BER1by1,'r', EbNo,BER2by1,'g', EbNo,BER1by2,'b', EbNo,BER2by2,'m');
hold off;
```

Zero-Forcing with Rayleigh multipath channel

```
clc;
clear;
frmLen = 192; % frame length
maxNumPackets = 100; % maximum number of packets
```

```
EbNo = 0:2:14; % Eb/No varying to 12 dB
N = 2; % number of Tx antennas
M = 2; % number of Rx antennas
pLen = 8; % number of pilot symbols per frame
W = hadamard(pLen);
pilots1 = W(:, 1);
pilots2 = W(:, 1:N); % orthogonal set per transmit antenna
h = gcf;
P = 2; % Modulation Order
T = 0.000000001;
FD= 100; % Doppler Shift
tau = [0 2.5e-8 3e-8]; % path delay
pdb = [0 -10 -20]; % average path gain
% Create BPSK mod-demod objects
bpskmod = modem.pskmod('M', P, 'SymbolOrder', 'Gray');
bpskdemod = modem.pskdemod(bpskmod);

% Pre-allocate variables for speed
tx1=zeros(frmLen,1) ; s1=zeros(frmLen/2,1) ; s2=s1 ; tx2=zeros(frmLen,2);
trans1=zeros(pLen+frmLen,1) ; trans2=zeros(pLen+frmLen,2) ;
r1by1=zeros(pLen+frmLen,1) ; r1by1_noisy=r1by1 ; H1by1=zeros(frmLen,1);
r1by2=zeros(pLen+frmLen,2) ; r1by2_noisy=r1by2 ; H1by2=zeros(frmLen,2);
z12=zeros(frmLen,2) ; z12_2=zeros(frmLen,1);
r2by1=zeros(pLen+frmLen,1) ; r2by1_noisy=r2by1 ; H2by1=zeros(frmLen,2);
z21_1=zeros(frmLen/2,1) ; z21_2=z21_1 ; z21=zeros(frmLen,1);
r2by2=zeros(pLen+frmLen,2) ; r2by2_noisy=r2by2 ; H2by2=zeros(frmLen,2,2);
z22_1=zeros(frmLen/2,2) ; z22_2=z22_1 ; z22=zeros(frmLen,2);
BER1by1=zeros(1,length(EbNo)); BER2by1 = BER1by1 ; BER1by2 = BER1by1 ; BER2by2 =
BER1by1;

% Set up a figure for visualizing BEr results
clf(h);
grid on;
hold on;
%set(gca,'yscale','log','xlim',[EbNo(1)-1 , EbNo(end)+1],'ylim',[1e-7 1]);
xlabel('Eb/No (dB)');
ylabel('BEr');
set(h,'NumberTitle','off');
set(h,'Name','Orthogonal Space-Time Block Coding');
set(h, 'renderer', 'zbuffer'); title('Comparison between the different systems');

% Loop over several EbNo points
for idx = 1:length(EbNo)
SNR = EbNo(idx)

% Loop over the number of packets
for packetIdx = 1:maxNumPackets
data = randint(frmLen, 1, P); % Generating Random Data (0 or 1)
bpskmod = modem.pskmod('M', P, 'SymbolOrder', 'Gray');
tx1 = modulate(bpskmod, data); % BPSK modulation

% Alamouti Encoder, G2
% X = [s1 s2; -s2* s1*]
s1 = tx1(1:N:end); s2 = tx1(2:N:end);
tx2(1:2:end, :) = [s1 s2];
tx2(2:2:end, :) = [-conj(s2) conj(s1)];
```

```
% Prepend pilot symbols for each frame
trans1 = [pilots1; tx1];
trans2 = [pilots2; tx2];

% For SISO System (1x1):
chan1by1 = mimochan (1,1,T,FD,tau,pdb);
r1by1 = filter(chan1by1, trans1);
r1by1_noisy = awgn(r1by1, EbNo(idx));
Z1 = pinv(r1by1_noisy);
r7 = Z1';
% Channel Estimation
H1by1(1, 1) = (Z1(1:pLen). ' * pilots1(:, 1)./pLen); % Least Square Channel Estimation

% assume held constant for the whole frame
H1by1 = H1by1(ones(frmLen, 1),1);

% Determine errors
demod1by1 = demodulate(bpskdemod, (r1by1_noisy(pLen+1:end,:).*conj(H1by1)));
error1by1(packetIdx) = biterr(demod1by1, data);

% For SIMO System (1x2):
chan1by2 = mimochan (1,2,T,FD,tau,pdb);
r1by2 = filter (chan1by2, trans1);
r1by2_noisy = awgn(r1by2,EbNo(idx));
Z2 = pinv(r1by2_noisy);
r1 = Z2';
% Channel Estimation
for n = 1:2
H1by2(1, n) = (r1(1:pLen, n).' * pilots1(:, :)./pLen); % Least Square Channel Estimation
end

% assume held constant for the whole frame
H1by2 = H1by2(ones(frmLen, 1),:);

% Determine Errrors
z12=r1by2_noisy((pLen+1:end),:).* conj(H1by2);
z12_2= sum(z12,2);
demod1by2 = demodulate(bpskdemod,z12_2);
error1by2(packetIdx) = biterr(demod1by2, data);

% For MISO System (2x1):
chan2by1 = mimochan (2,1,T,FD,tau,pdb);
r2by1 = filter (chan2by1 , (trans2./sqrt(2)));
r2by1_noisy = awgn(r2by1,EbNo(idx));
Z3 = pinv(r2by1_noisy);
r2 = Z3';
% Channel Estimation
for n = 1:2
H2by1(1, n) = (r2(1:pLen, :).' * pilots2(:, n)./pLen); % Least Square Channel Estimation
end

% assume held constant for the whole frame
H2by1 = H2by1(ones(frmLen, 1), :);

hidx = 1:2:length(H2by1);
```

```
z21_1 = r2by1_noisy(pLen+1:2:end).*conj(H2by1(hidx,1))+ ....
conj(r2by1_noisy(pLen+2:2:end)).*H2by1(hidx,2);
z21_2 = r2by1_noisy(pLen+1:2:end).*conj(H2by1(hidx,2))- ....
conj(r2by1(pLen+2:2:end)).*H2by1(hidx,1);
z21(1:2:end)= z21_1;
z21(2:2:end) = z21_2;

% Determine Errors
demod2by1 = demodulate(bpskdemod, z21);
error2by1 (packetIdx) = biterr(demod2by1, data);

% For MIMO System (2x2)
chan2by2 = mimochan (2,2,T,FD,tau,pdb);
r2by2 = filter (chan2by2 , (trans2./sqrt(2)));
r2by2_noisy = awgn(r2by2,EbNo(idx));
Z4 = pinv(r2by2_noisy);
r3 = Z4';

% Channel Estimation
for n = 1:2
H2by2(1, n, :) = (r3(1:pLen, :).' * pilots2(:, n)./pLen); % Least Square Channel Estimation
end

% assume held constant for the whole frame
H2by2 = H2by2(ones(frmLen, 1), :, :);

% Combiner using estimated channel
heidx = 1:2:length(H2by2);
for i = 1:2
z22_1(:, i) = r2by2_noisy(pLen+1:2:end, i).* conj(H2by2(heidx, 1, i)) + ...
conj(r2by2_noisy(pLen+2:2:end, i)).* H2by2(heidx, 2, i);

z22_2(:, i) = r2by2_noisy(pLen+1:2:end, i).* conj(H2by2(heidx, 2, i)) - ...
conj(r2by2_noisy(pLen+2:N:end, i)).* H2by2(heidx, 1, i);
end
z22(1:2:end, :) = z22_1 ; z22(2:2:end, :) = z22_2;

% Detemine Errors
demod2by2 = demodulate(bpskdemod, sum(z22, 2));
error2by2(packetIdx) = biterr(demod2by2, data);

end

% Calculate BER for current idx
BER1by1(idx) = sum(error1by1)/(maxNumPackets*frmLen);
BER1x1 = BER1by1(idx)

BER1by2(idx) = sum(error1by2)/(maxNumPackets*frmLen);
BER1x2 = BER1by2(idx)

BER2by1(idx) = sum(error2by1)/(maxNumPackets*frmLen);
BER2x1 = BER2by1(idx)

BER2by2(idx) = sum(error2by2)/(maxNumPackets*frmLen);
BER2x2 = BER2by2(idx)
```

```
% Plot the results
semilogy(EbNo(1:idx), BER1by1(1:idx), 'r*');
% drawnow;
semilogy(EbNo(1:idx), BER2by1(1:idx), 'go');
%drawnow;
semilogy(EbNo(1:idx), BER1by2(1:idx), 'b+');
%drawnow;
semilogy(EbNo(1:idx), BER2by2(1:idx), 'm');
legend('SISO System (1Tx, 1Rx)','MISO System (2Tx, 1Rx)','SIMO System (1Tx, 2Rx)','MIMO
System (2Tx,2Rx)');
drawnow;
end % end of for loop for EbNo

semilogy(EbNo,BER1by1,'r', EbNo,BER2by1,'g', EbNo,BER1by2,'b', EbNo,BER2by2,'m');
hold off;
```

Zero-Forcing with Ricean fading channel

```
clc;
clear;
frmLen = 192; % frame length
maxNumPackets = 100; % maximum number of packets
EbNo = 0:2:14; % Eb/No varying to 12 dB
N = 2; % number of Tx antennas
M = 2; % number of Rx antennas
pLen = 8; % number of pilot symbols per frame
W = hadamard(pLen);
pilots1 = W(:, 1);
pilots2 = W(:, 1:N); % orthogonal set per transmit antenna
h = gcf;
P = 2; % Modulation Order
T = 0.000000001;
FD= 100; % Doppler Shift
tau = [0 2.5e-8 3e-8]; % path delay
pdb = [0 -10 -20]; % average path gain
% Create BPSK mod-demod objects
bpskmod = modem.pskmod('M', P, 'SymbolOrder', 'Gray');
bpskdemod = modem.pskdemod(bpskmod);

% Pre-allocate variables for speed
tx1=zeros(frmLen,1) ; s1=zeros(frmLen/2,1) ; s2=s1 ; tx2=zeros(frmLen,2);
trans1=zeros(pLen+frmLen,1) ; trans2=zeros(pLen+frmLen,2) ;
r1by1=zeros(pLen+frmLen,1) ; r1by1_noisy=r1by1 ; H1by1=zeros(frmLen,1);
r1by2=zeros(pLen+frmLen,2) ; r1by2_noisy=r1by2 ; H1by2=zeros(frmLen,2);
z12=zeros(frmLen,2) ; z12_2=zeros(frmLen,1);
r2by1=zeros(pLen+frmLen,1) ; r2by1_noisy=r2by1 ; H2by1=zeros(frmLen,2);
z21_1=zeros(frmLen/2,1) ; z21_2=z21_1 ; z21=zeros(frmLen,1);
r2by2=zeros(pLen+frmLen,2) ; r2by2_noisy=r2by2 ; H2by2=zeros(frmLen,2,2);
z22_1=zeros(frmLen/2,2) ; z22_2=z22_1 ; z22=zeros(frmLen,2);
```

```
BER1by1=zeros(1,length(EbNo)); BER2by1 = BER1by1 ; BER1by2 = BER1by1 ; BER2by2 =
BER1by1;

% Set up a figure for visualizing BEr results
clf(h);
grid on;
hold on;
%set(gca,'yscale','log','xlim',[EbNo(1)-1 , EbNo(end)+1],'ylim',[1e-7 1]);
xlabel('Eb/No (dB)');
ylabel('BEr');
set(h,'NumberTitle','off');
set(h,'Name','Orthogonal Space-Time Block Coding');
set(h, 'renderer', 'zbuffer'); title('Comparison between the different systems');

% Loop over several EbNo points
for idx = 1:length(EbNo)
SNR = EbNo(idx)

% Loop over the number of packets
for packetIdx = 1:maxNumPackets
data = randint(frmLen, 1, P); % Generating Random Data (0 or 1)
bpskmod = modem.pskmod('M', P, 'SymbolOrder', 'Gray');
tx1 = modulate(bpskmod, data); % BPSK modulation

% Alamouti Encoder, G2
% X = [s1 s2; -s2* s1*]
s1 = tx1(1:N:end); s2 = tx1(2:N:end);
tx2(1:2:end, :) = [s1 s2];
tx2(2:2:end, :) = [-conj(s2) conj(s1)];

% Prepend pilot symbols for each frame
trans1 = [pilots1; tx1];
trans2 = [pilots2; tx2];

% For SISO System (1x1):
chan1by1 = mimochan (1,1,T,FD,tau,pdb);
chan1by1.KFactor = 2;
r1by1 = filter(chan1by1, trans1);
r1by1_noisy = awgn(r1by1, EbNo(idx));
Z1 = pinv(r1by1_noisy);
r7 = Z1';
% Channel Estimation
H1by1(1, 1) = (Z1(1:pLen). ' * pilots1(:, 1)./pLen); % Least Square Channel Estimation

% assume held constant for the whole frame
H1by1 = H1by1(ones(frmLen, 1),1);

% Determine errors
demod1by1 = demodulate(bpskdemod, (r1by1_noisy(pLen+1:end,:).*conj(H1by1)));
error1by1(packetIdx) = biterr(demod1by1, data);

% For SIMO System (1x2):
chan1by2 = mimochan (1,2,T,FD,tau,pdb);
chan1by2.KFactor = 2;
r1by2 = filter (chan1by2, trans1);
r1by2_noisy = awgn(r1by2,EbNo(idx));
```

```
Z2 = pinv(r1by2_noisy);
r1 = Z2';
% Channel Estimation
for n = 1:2
H1by2(1, n) = (r1(1:pLen, n).' * pilots1(:, :)./pLen); % Least Square Channel Estimation
end

% assume held constant for the whole frame
H1by2 = H1by2(ones(frmLen, 1),:);

% Determine Errrors
z12=r1by2_noisy((pLen+1:end),:).* conj(H1by2);
z12_2= sum(z12,2);
demod1by2 = demodulate(bpskdemod,z12_2);
error1by2(packetIdx) = biterr(demod1by2, data);

% For MISO System (2x1):
chan2by1 = mimochan (2,1,T,FD,tau,pdb);
chan2by1.KFactor = 2;
r2by1 = filter (chan2by1 , (trans2./sqrt(2)));
r2by1_noisy = awgn(r2by1,EbNo(idx));
Z3 = pinv(r2by1_noisy);
r2 = Z3';
% Channel Estimation
for n = 1:2
H2by1(1, n) = (r2(1:pLen, :).' * pilots2(:, n)./pLen); % Least Square Channel Estimation
end

% assume held constant for the whole frame
H2by1 = H2by1(ones(frmLen, 1), :);

hidx = 1:2:length(H2by1);
z21_1 = r2by1_noisy(pLen+1:2:end).*conj(H2by1(hidx,1))+ ....
conj(r2by1_noisy(pLen+2:2:end)).*H2by1(hidx,2);
z21_2 = r2by1_noisy(pLen+1:2:end).*conj(H2by1(hidx,2))- ....
conj(r2by1(pLen+2:2:end)).*H2by1(hidx,1);
z21(1:2:end)= z21_1;
z21(2:2:end) = z21_2;

% Determine Errors
demod2by1 = demodulate(bpskdemod, z21);
error2by1 (packetIdx) = biterr(demod2by1, data);

% For MIMO System (2x2)
chan2by2 = mimochan (2,2,T,FD,tau,pdb);
chan2by2.KFactor = 2;
r2by2 = filter (chan2by2 , (trans2./sqrt(2)));
r2by2_noisy = awgn(r2by2,EbNo(idx));
Z4 = pinv(r2by2_noisy);
r3 = Z4';

% Channel Estimation
for n = 1:2
H2by2(1, n, :) = (r3(1:pLen, :).' * pilots2(:, n)./pLen); % Least Square Channel Estimation
end
```

```
% assume held constant for the whole frame
H2by2 = H2by2(ones(frmLen, 1), :, :);

% Combiner using estimated channel
heidx = 1:2:length(H2by2);
for i = 1:2
z22_1(:, i) = r2by2_noisy(pLen+1:2:end, i).* conj(H2by2(heidx, 1, i)) + ...
conj(r2by2_noisy(pLen+2:2:end, i)).* H2by2(heidx, 2, i);

z22_2(:, i) = r2by2_noisy(pLen+1:2:end, i).* conj(H2by2(heidx, 2, i)) - ...
conj(r2by2_noisy(pLen+2:N:end, i)).* H2by2(heidx, 1, i);
end
z22(1:2:end, :) = z22_1 ; z22(2:2:end, :) = z22_2;

% Detemine Errors
demod2by2 = demodulate(bpskdemod, sum(z22, 2));
error2by2(packetIdx) = biterr(demod2by2, data);

end

% Calculate BER for current idx
BER1by1(idx) = sum(error1by1)/(maxNumPackets*frmLen);
BER1x1 = BER1by1(idx)

BER1by2(idx) = sum(error1by2)/(maxNumPackets*frmLen);
BER1x2 = BER1by2(idx)

BER2by1(idx) = sum(error2by1)/(maxNumPackets*frmLen);
BER2x1 = BER2by1(idx)

BER2by2(idx) = sum(error2by2)/(maxNumPackets*frmLen);
BER2x2 = BER2by2(idx)

% Plot the results
semilogy(EbNo(1:idx), BER1by1(1:idx), 'r*');
% drawnow;
semilogy(EbNo(1:idx), BER2by1(1:idx), 'go');
%drawnow;
semilogy(EbNo(1:idx), BER1by2(1:idx), 'b+');
%drawnow;
semilogy(EbNo(1:idx), BER2by2(1:idx), 'm');
legend('SISO System (1Tx, 1Rx)','MISO System (2Tx, 1Rx)','SIMO System (1Tx, 2Rx)','MIMO
System (2Tx,2Rx)');
drawnow;
end % end of for loop for EbNo

semilogy(EbNo,BER1by1,'r', EbNo,BER2by1,'g', EbNo,BER1by2,'b', EbNo,BER2by2,'m');
hold off;
```