JSPM’s
JAYAWANTRAO SAWANAT COLLEGE OF ENGINEERING
Sr. No. 58, Handewadi Road, Hadapsar, Pune, Maharashtra 411028
Department of Electronics and Telecommunication Engineering

CLASS: - T.E. (E&TC)

SUBJECT: - Cellular Network Lab

INSTRUCTION MANUAL

JSPM's
JAYAWANTRAO SAWANAT COLLEGE OF ENGINEERING
Sr. No. 58, Handewadi Road, Hadapsar, Pune, Maharashtra 411028
Department of Electronics and Telecommunication Engineering

Instruction Manual

Course: Cellular Network Lab

Experiment List			
Experiment Learning Outcomes	**Title of Experiment**	**CO mapped**	**PO mapped**
ELO-1: Students will be able to simulate performance parameters i.e Bit Error rate (BER), Path loss and RMS Delay spread for wireless channel with respect to provided SNR.	1) Compute and compare the median loss by employing Hata model for various distance for carrier frequencies of 2.1 GHz and 6 GHz. Assume transmit and receive antenna heights of 40 m and 2 m in a large city. Plot the graph of path loss vs distance. 2) Simulate BER performance over a wireline AWGN channel with BPSK transmission for SNR: 0 to 50 dB.	**CO 308.1**	PO1,PO2, PO12
ELO-2: Students will be able to calculate RMS delay spread obtained from the PDP and will know about need of ISI equalizer at receiver.	3) Compute the RMS delay spread for a given Power profile and plot the graph of Power vs Delay.	**CO 308.1**	PO1,PO2, PO12

ELO-3: Students will be able to perform Link budget analysis, BER performance of multi antenna channel and compute Doppler shift.	4) Perform a Link-Budget analysis for a wireless communication system.	**CO 308.3**	PO1,PO2,
ELO-3: Students will be able to perform BER performance of multi antenna channel and compute Doppler shift.	5) Simulate BER performance of multi-antenna Rayleigh channel for SNR varying from 0 to 60dB.	**CO 308.2**	PO1,PO2
ELO-3: : Students will be able to perform Link budget analysis, BER performance of multi antenna channel and compute Doppler shift.	6) Compute doppler shift of the received signal for different carrier frequency of mobile generations by considering vehicle is moving at 60 miles per hour at an angle of 30 degree with the line joining the base station.	**CO 308.3**	PO1,PO2
ELO-4: Students will be able to evaluate performance parameters using open source network simulator tool.	7) Study mobile environment to evaluate performance parameters using any open source Network Simulator tool.	**CO 308.6:**	PO1,PO2

ELO 2 : Students will be able to perform Link budget analysis, BER performance of multi antenna channel and compute Doppler shift.	8) Simulate the OFDM system and evaluate frame error rate against SNR	**CO 308.2**	PO1,PO2
ELO2 : Students will be able to perform Link budget analysis, BER performance of multi antenna channel and compute Doppler shift.	9)Simulate a cellular system with 48 channels per cell and blocking probability of 2%. Assume traffic per user is 0.04 E. What is the number of users that can be supported in a city of 603 km2 area if cell radios are changed in the steps of 500 m, 700m, 900 m, 1000 m 1200 m and 1500 m.	**CO 308.3**	PO1,PO2

Common Procedure to all Programs GNU Octave
1. Click on the GNU Octave (GUI) Icon on the desktop.
2. Octave window open.
3. Click on the 'FILE' Menu on menu bar.
4. Click on New Script from the file Menu.
5. An editor window open, start typing commands.
6. Then Click on Run from Menu bar and then click Save and Run option

EXPERIMENT NO.1

Title: Path Loss Calculation using Hata Model

Aim: Compute and compare the median loss by employing Hata model for various distance for carrier frequencies of 2.1 GHz and 6 GHz. Assume transmit and receive antenna heights of 40 m and 2 m in a large city. Plot the graph of path loss vs distance.

CO 308.1: Illustrate functioning of wireless propagation models and diversity in wireless communication

ELO-1: Students will be able to simulate performance parameters i.e Bit Error rate (BER), Path loss and RMS Delay spread for wireless channel with respect to provided SNR..

Objectives: After performing this experiment, the learner will be able to

1. Calculate median attenuation loss by using Hata model
2. Calculate median attenuation loss for different combinations of frequencies, distance and antenna heights

Pre-requisites: Basic Communication system.

Hardware & Software used: Octave, PC and Printer

Theory:

MATLAB is a matrix laboratory. It is a language used for technical computations. Whereas, Octave is programming language used for numerical computations. In order to plan the installation and deployment of a wireless network, one needs to characterize the performance of the communication system in terms of the transmitted power and total load in terms of users that can be supported by the network. Link-budget of wireless link is a systematic listing of power losses and gains of different intermediate components in the transceiver chain.

Hata Model

The Hata model is a popular model for signal strength prediction proposed initially by the Japanese engineer MasaharuHata in his 1980 paper titled "Empirical Formula for Propagation Loss in Land Mobile Radio Services". The Hata model presents an analytical approximation for the graphical-information based on another Okumura model. Parameters required for simulations:
Hbts= Height measured from the base of the BTS tower to the radiation centerline
Tbts = Terrain elevation at the location of the BTS
Htav= Height of the average terrain (from 3 Km to 15 km distance from the BTS)
Hm=Height of the mobile antenna in meters
f= Range of frequencies in MHz
d=Range of Tx-Rx separation distances in Kilometers

Pt = Power transmitted by the BTS antenna in Watts
Gt= BTS antenna gain in dBi
Hb= Hbts+ Tbts - Htav = Effective Height of the BTS antenna in meters
models =

1. Big City (Urban model

aHm=3.2*(log10(11.75*Hm))^2-4.97;
C=0

2. Small & Medium City (Urban model)

aHm = (1.1*log10(f)-0.7)*Hm-(1.56*log10(f)-0.8);

3. Sub-urban environment

aHm = (1.1*log10(f)-0.7)*Hm-(1.56*log10(f)-0.8);
C=-2*(log10(f/28))^2-5.4;

4. Open Rural environment

aHm = (1.1*log10(f)-0.7)*Hm-(1.56*log10(f)-0.8);
C=-4.78*(log10(f))^2+18.33*log10(f)-40.98;
A = 69.55 + 26.16*log10(f) - 13.82*log10(Hb)-aHm;
B = 44.9 - 6.55*log10(Hb);
Path Loss (dB)=PL=A+B*log10(d)+C;
Received Signal Level(dB) =Pr = 10*log10(Pt*1000)+Gt-PL

Algorithm: Compute and compare the median loss by employing Hata model for various distance for the given carrier frequency

---

Input: Transmit and Receive antenna heights

Output: Plot the graph of path loss vs distance

---

1. Initialize all the parameters with values
2. Compute Path Loss
3. Compute Received Signal Level
4. Plot Graph of Path loss vs distance

Program:

```
%Distance vs loss (dB) for small city for Hata Model'
clc;
clear all;
close all;
Hte=input('Enter Base Station Antena Hight (hte)');
Hre=input('Enter Mobile Station Antena Hight(hre)');
d= input('Enter Distance from Base Station');
```

```
f=input('Enter The Frequency:');
for i=1:length(d)
  CH=0.8+((1.1*log(d))-0.7)*Hre-1.56*log(d);
  LU=69.55+26.16*log(d)-132.82*log(Hte)-CH+(44.9-6.55*log(Hte))*log(f);
  end
figure(1)
plot(d,LU)
title('Distance vs loss (dB) for small city for hata Model');
xlabel ('Frequency(MHz)');
ylabel('Propogation Path loss(dB)');
grid on;
[200 300 400 500 600 700 800 900 1000 1100]
```

Output:

Conclusions:

JSPM's
JAYAWANTRAO SAWANAT COLLEGE OF ENGINEERING
Sr. No. 58, Handewadi Road, Hadapsar, Pune, Maharashtra 411028
Department of Electronics and Telecommunication Engineering

Que 1: Brief with mathematical equation a Free-Space Propagation Model?

Ans1 :

Que 2 :Compare Okumura Model and Hata Model in two points
Ans2 :

EXPERIMENT NO.2

Title: Simulate BER performance over a wireline AWGN channel with BPSK transmission for SNR: 0 to 50 dB.

Aim: Simulate BER performance over a wireline AWGN channel with BPSK transmission for SNR: 0 to 50 dB.

CO 308.1: Illustrate functioning of wireless propagation models and diversity in wireless communication.

ELO-1: Students will be able to simulate performance parameters i.e Bit Error rate (BER), Path loss and RMS Delay spread for wireless channel with respect to provided SNR.

Pre-requisites: Basic Communication system.

Hardware & Software used: Octave, PC and Printer

Theory:

Digital modulation schemes have greater capacity to convey large amounts of information than analog modulation schemes. Further, Phase shift keying (PSK) involves the switching of the instantaneous phase of the carrier between 2 or more levels according to the baseband digital data. The medium between the transmitting antenna and the receiving antenna is termed as channel. In wireless transmission, the characteristics of the signal change as it travels from the transmitter to the receiver. The signal characteristics are due to several phenomena:

1) Existence of line of sight path between the antennas
2) Reflection, refraction and diffraction of the signal due to the objects in between the antennas
3) The relative motion between the transmitter and receiver and the objects in between them
4) The signal attenuation as it travels through the medium
5) Noise

Algorithm: Simulate BER performance over a wireline AWGN channel with BPSK transmission for SNR: 0 to 50 dB.

Input: Value of SNR

Output: Plot the graph of Bit Error Rate vs SNR

---

1. Initialize all the parameters with values
2. Compute Bit error Rate
3. Plot Graph of Bit Error Rate vs SNR

Program: **Matlab Code :**

```
%Script to generate Binary PSK

clear all;
close all;
clc;
fc=1000;                        % Frequency for "0" bits
t=linspace(0,1/1000,50);

e0=cos(2*pi*fc*t);              % BPSK output for "1"
e1=-cos(2*pi*fc*t);             % BPSK output for "0"

b=mod(randperm(16),2);
bnot=1-b;
n=['The binary data is   ',num2str(b)];
bpsk1=[ ];bpsk2=[ ];bin=[ ];

for i=1:length(b)
   bpsk1=[bpsk1,b(i)*e0];
   bpsk2=[bpsk2,bnot(i)*e1];
   bin=[bin,b(i)*ones(1,50)];
end;

bpskout=bpsk1+bpsk2;
tm=[0:length(bpsk1)-1];

plot(tm,bin,'r--');
axis([0 length(bin) 0 1.5]);
hold on;

plot(tm,bpskout,'b');
axis([0 length(tm) -1.5 1.5]);
hold off;

xlabel('Time index');
ylabel('Amplitude');
```

```
legend('Random binary','BPSK output');
title('Simulation of Binary Phase Shift keying');

gtext(n);


% Script of Eb/N0 Vs BER for BPSK modulation scheme
clear;
clc;
N=10000000;                    %Number of input bits
EbN0dB = -5:1:27;
data=randn(1,N)>=0;            % Generating a uniformly distributed random 1s and 0s
bpskModulated = 2*data-1;  % Mapping 0->-1 and 1->1
M=2;                           % Number of Constellation points M=2^k for BPSK k=1
Rm=log2(M);                    % Rm=log2(M) for BPSK M=2
Rc=1;                          %Number of Constellation points M=2^k for BPSK k=1
BER = zeros(1,length(EbN0dB)); %Place holder for BER values for each Eb/N0
index=1;
for k=EbN0dB,
%Adding noise with variance according to the required Eb/N0
EbN0 = 10.^(k/10); %Converting Eb/N0 dB value to linear scale
noiseSigma = sqrt(1./(2*Rm*Rc*EbN0)); %Standard deviation for AWGN Noise
noise = noiseSigma*randn(1,length(bpskModulated));
received = bpskModulated + noise;
%Threshold Detector
estimatedBits=(received>=0);
%Bit Error rate Calculation
BER(index) = sum(xor(data,estimatedBits))/length(data);
index=index+1;
endfor
%Plot commands follows
plotHandle=plot(EbN0dB,log10(BER),'r--');
set(plotHandle,'LineWidth',1.5);
title('SNR per bit (Eb/N0) Vs BER Curve for BPSK Modulation Scheme');
xlabel('SNR per bit (Eb/N0) in dB');
ylabel('Bit Error Rate (BER) in dB');
grid on;
hold on;
theoreticalBER = 0.5*erfc(sqrt(10.^(EbN0dB/10)));
plotHandle=plot(EbN0dB,log10(theoreticalBER),'k*');
set(plotHandle,'LineWidth',1.5);
legend('Simulated','Theoretical');
grid on;
```

Output:

Conclusion:

Que 1: What is Bit Error rate and SNR ?Ans1 :

Que 2 : What is AWGN Channel.

Ans2 :

EXPERIMENT NO.3

Title: Compute the RMS delay spread for a given Power profile and plot the graph of Power vs Delay.

Aim: Compute the RMS delay spread for a given Power profile and plot the graph of Power vs Delay.

CO 308.1: Illustrate functioning of wireless propagation models and diversity in wireless communication

ELO-1: Students will be able to calculate RMS delay spread obtained from the PDP and will know about need of ISI equalizer at receiver.

Objectives: After performing this experiment, the learner will be able to

1. Calculate power received from the estimate of delay spread.
2. The RMS delay spread obtained from the PDP must be compared with the symbol duration, , when the symbol time period is greater than 10 times the RMS delay spread, no ISI equalizer is needed in the receiver

Pre-requisites: Basic Communication system.

Hardware & Software used: Octave, PC and Printer

Theory:

In radio systems with low antenna heights, there are often multiple indirect paths between the transmitter and receiver due to reflections from surrounding objects, in addition to the direct path when there is line-of-sight. Such multipath propagation is particularly significant in urban environments, where the sides of buildings and paved road surfaces provide strong reflections. As a result, the received signal consists of the summation of several components having various amplitudes, phase angles and directions of arrival.

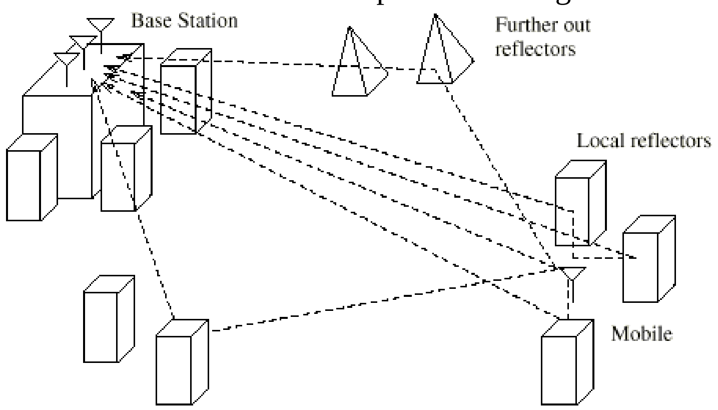


Figure: Illustration of reflections of various kinds

The resulting spatial variability of signal strength can be viewed as having two regimes:

a) Rapid fading which varies over distances of the order of a wavelength due primarily to changes in phase angles of different signal components.

b) Slow fading which varies over larger distances due primarily to changes in shadowing loss by surrounding objects.

In addition, the various signal components can be Doppler shifted by different amounts due to the movement of the mobile or of reflecting objects such as vehicles.

The multipath mobile channel can be characterized in terms of its impulse response which varies at a rate dependent on the speed of the mobile and/or the scatterers. Therefore, a receiver has to be able to cope with the signal distortion arising from echoes in the channel as well as the rapid changes in the nature of this distortion. Such characteristics of the mobile radio channel are described by the power delay profiles and the Doppler spectra which are obtained from wideband channel sounding measurements.

Delay Spread :

As an electromagnetic wave can travel from the transmitter to the receiver via multiple paths, the signal can reach the receiver with interference from its own echoes. Delay spread measures the effect of the time dispersion in multipath channels. Thus, the total power received in a multipath wireless channel occurs over a spread of time referred to as the delay spread.

Figure shows how a transmitted pulse its received at the receiver with different signal strength as it travels through a multipath channel with different propagation delays (τ, τ1, τ2).

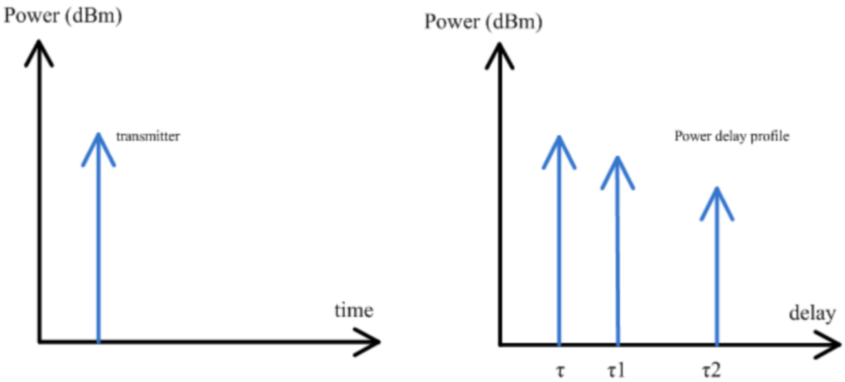


Figure: Power Delay Profile Example 1

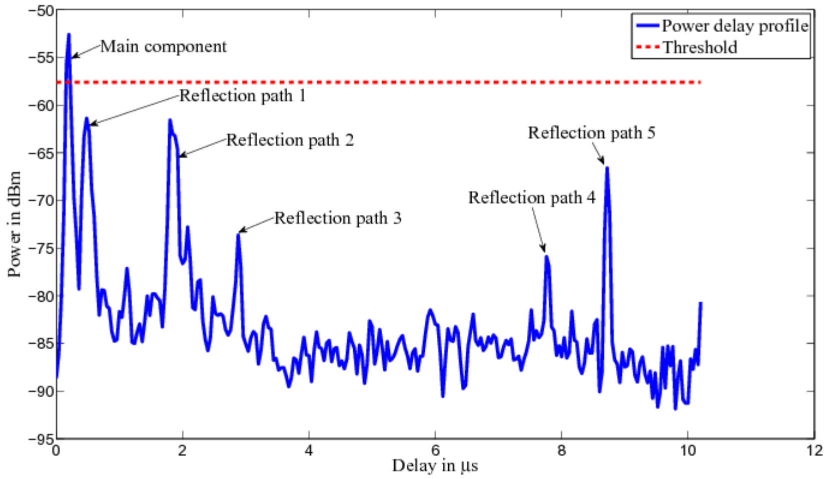


Figure: Power Delay Profile Example 2

Detection of the first arrival peak is not an easy task to do when generating a power delay profile. Generally, one might think that the first peak corresponds to the maximum peak of the signal, but this is not necessarily the case. It may be that the peak of maximum energy is not the first peak, as this can suffer greater attenuation than other arrivals in Non-Line of Sight (NLOS) situations.

Power Delay Profile is usually supplied as a table of values obtained from empirical data and it serves as a guidance to system design. Nevertheless, it is not an accurate representation of the real environment in which the mobile is destined to operate at.

Power delay profile is a good representation of the average "geometry" of the transmitter, the receiver, and the reflectors. To quantify "how spread-out" the arriving signals are, we use time dispersion parameters:

Excess delay: the delay with respect to the first arriving signal (τ)

Maximum excess delay: the excess delay of the latest arriving Multi Path Component (MPC)

Mean excess delay: the "mean" excess delay of all arriving MPC

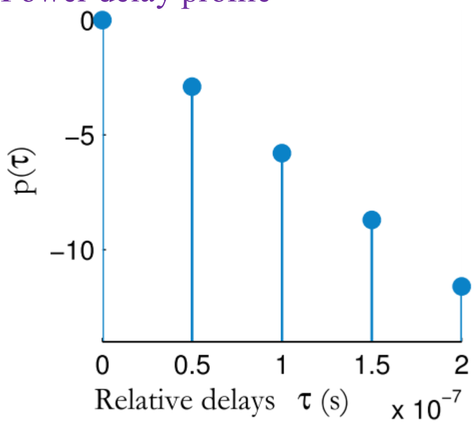
RMS delay spread: the "standard deviation" of the excess delay of all arriving MPC

In a nutshell, the RMS delay spread indicates the capability of the communication channel of supporting high data rate communications by implying the probability of performance degradation which may occur due to the ISI because of multipath signal propagation.

Even though a same signal is transmitted from single transmission antenna, the signal may go through various different path. Each of the different path may cause different travel distance if the signal get reflected by one or more obstacles (like buildings) and in some case different path may has different physical property of propagation media, so it is higly likely that the signal traveling through different path would arrive at the reciever antenna at different timing. So if you send a signal from a transmitter antenna and measure the arrival time at the reciever antenna which is a certain distance away from the transmitter antenna, you would get multiple different arrival timing.

If you plot those arrival timing on the axis of time, you would see a certain variation (spread) of those values. This spread is called 'Delay Spread'.

Power delay profile



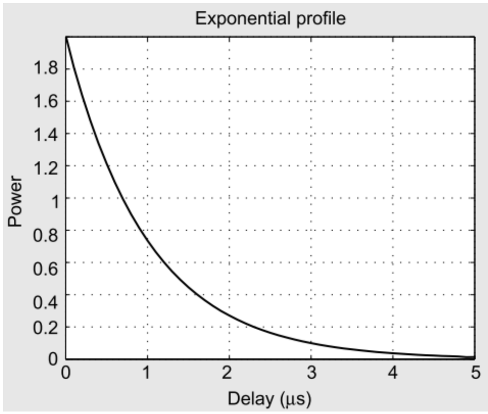


Figure: (1) Discrete PDP (2) Continuous/exponential PDP

For continuous PDP, the RMS delay spread (τrms) can be calculated as

$$\tau_{rms} = \sqrt{\frac{\int_{-\infty}^{\infty} S(\tau)(\tau-\bar{\tau})^2 d\tau}{\int_{-\infty}^{\infty} S(\tau) d\tau}}$$

where, the mean delay $\bar{\tau}$ is given by

$$\bar{\tau} = \sqrt{\frac{\int_{-\infty}^{\infty} S(\tau)\tau d\tau}{\int_{-\infty}^{\infty} S(\tau) d\tau}}$$

For discrete PDP (as in Figure 11.4), the RMS delay spread (τ_{rms}) can be calculated as

$$\tau_{rms} = \sqrt{\frac{\sum_i p_i(\tau_i-\bar{\tau})^2}{\sum_i p_i}}$$

where, p_i is the power of the ith path, τ_i is the delay of the ith path and the mean delay $\bar{\tau}$ is given by

$$\bar{\tau} = \sqrt{\frac{\sum_i p_i \tau_i}{\sum_i p_i}}$$

Knowledge of the delay spread is essential in system design for determining the trade-off between the symbol rate of the system and the complexity of the equalizers at the receiver. The ratio of RMS delay spread (τrms) and symbol time duration (Tsym) quantifies the strength of intersymbol interference (ISI). Typically, when the symbol time period is greater than 10 times the RMS delay spread, no ISI equalizer is needed in the receiver. The RMS delay spread obtained from the PDP must be compared with the symbol duration to arrive at this conclusion.

With the power delay profile, one can classify a multipath channel into frequency selective or frequency non-selective category. The derived parameter, namely, the maximum excess delay together with the symbol ~~time of each transmitted symbol, can be used to classify the channel into frequency~~ selective or non-selective channel.

PDP can be used to estimate the average power of a multipath channel, measured from the first signal that strikes the receiver to the last signal whose power level is above certain threshold. This threshold is chosen based on receiver design specification and is dependent on receiver sensitivity and noise floor at the receiver.

Maximum excess delay, also called maximum delay spread, denoted as (Tm), is the relative time difference between the first signal component arriving at the receiver to the last component whose power level is above some threshold. Maximum delay spread (Tm) and the symbol time period (Tsym) can be used to classify a channel into frequency selective or non-selective category. This classification can also be done using coherence bandwidth (a derived parameter from spaced frequency correlation function which in turn is the frequency domain representation of power delay profile).

A channel is classified as frequency selective, if the maximum excess delay is greater than the symbol time period, i.e, Tm >Tsym. This introduces intersymbol interference into the signal that is being transmitted, thereby distorting it. This occurs since the signal components (whose powers are above either a threshold or the maximum excess delay), due to multipath, extend beyond the symbol time. Intersymbol interference can be mitigated at the receiver by an equalizer.

On the other hand, if the maximum excess delay is less than the symbol time period, i.e, Tm <Tsym , the channel is classified as frequency non-selective or zero-mean channel. Here, all the scattered signal components (whose powers are above either a specified threshold or the maximum excess delay) due to the multipath, arrive at the receiver within the symbol time. This will not introduce any ISI, but the received signal is distorted due to inherent channel effects like SNR condition. Equalizers in the receiver are not needed.

Formulae:

$$Mean\ delay, \bar{\tau} = \frac{\int_{-\infty}^{\infty} \phi(\tau)\ \tau\ d\tau}{\int_{-\infty}^{\infty} \phi(\tau)\ \ d\tau}$$

$$RMS\ delay, \sigma_{\tau}^{RMS} = \sqrt{\frac{\int_{-\infty}^{\infty} \phi(\tau)(\tau-\bar{\tau})^2 d\tau}{\int_{-\infty}^{\infty} \phi(\tau) d\tau}}$$

$$maximum\ \ symbol\ \ rate\ \ to\ \ avoid\ \ ISI = \frac{1}{10 \times \sigma_{\tau}^{RMS}}$$

$$For\ \ 0.9\ \ correlation, Coherance\ BW = \frac{1}{50 \times \sigma_{\tau}^{RMS}}$$

Consider,

$$moment\_1 = \int_{-\infty}^{\infty} \phi(\tau)\ \tau\ d\tau$$

$$moment\_2\ = \int_{-\infty}^{\infty} \phi(\tau)(\tau-\bar{\tau})^2 d\tau$$

Program:

Define new function, File→New Function→meas_continuous_PDP

```
function [meanDelay,rmsDelay,symbolRate,coherenceBW] =
meas_continuous_PDP(fun,lowerLim,upperLim)
%Function to calculate mean Delay, RMS delay spread, maximum symbol
```

```
%rate that a signal can be transmitted without ISI and the coherence
%BW for the PDP equation specified as function handle(fun)
% example: fun = @(tau) exp(-tau/0.00001); %given PDP equation
%lowerLim - lower limit for integration
%upperLim - upper limit for integration
moment_1 = @(x) x.*fun(x);
meanDelay = integral(moment_1,lowerLim,upperLim)/integral(fun,lowerLim,upperLim);
moment_2 = @(y) ((y-meanDelay).^2).*fun(y);
rmsDelay = sqrt(integral(moment_2,lowerLim,upperLim)/integral(fun,lowerLim,upperLim));
symbolRate = 1/(10*rmsDelay); %maximum symbol rate to avoid ISI
coherenceBW = 1/(50*rmsDelay);%for 0.9 correlation
%coherenceBW = 1/(5*rmsDelay);%for 0.5 correlation
Endfunction
```

Run following script after saving above function in a file.

```
fun = @(tau) 2*exp(-tau/1e-6);
[meanDelay, rmsDelay, symbolRate, coherenceBW] = meas_continuous_PDP(fun,0,10e-6);
tau = [0:0.01e-6:5e-6];
fun1 = 2*exp(-tau/1e-6);
plot (tau, fun1, 'r', 'LineWidth', 2);
title ('Power vs Delay', 'Fontsize',20);
xlabel ('Delay', 'Fontsize',16);
ylabel('Power(dBm)', 'Fontsize',16);
```

Conclusions:

1.What is end result of Delay in Propagation?

2.How delay can be avoided?

EXPERIMENT NO.4

Title:
Perform a Link-Budget analysis for a wireless communication system.

Aim:
Perform a Link-Budget analysis for a wireless communication system.

CO 308.3: Analyze Cellular radio network architecture for mobile communication

ELO-1: Students will be able to perform Link budget analysis, BER performance of multi antenna channel and compute Doppler shift.

Pre-requisites: Basic Communication system.

Hardware & Software used: MATLAB/Octave, PC and Printer

Theory:

Link-budget of a wireless link is a systematic listing of power losses and gains of different intermediate components in the transceiver chain. The various additive and negative components for the net signal power at the receiver

The + sign denotes a component which enhances or adds to the received signal strength while the – sign denotes a component which subtract from the signal strength or SNR. The final = in the last row denotes the required SNR. Therefore, The link-budget expression for the SNR required is given as:

$$SNRreq = Pt(dB) + Gt(dB) - L50(dB) - M\ dB + Gr(dB) - Lc(dB) - (N + 1)(dB)$$

Therefore, the above expression can be recast to compute the required transmit power as

$$Pt(dB) = SNRreq - Gt(dB) + L50(dB) + M\ dB - Gr(dB) + Lc(dB) + (N + 1)\ dB$$

Components of a typical wireless-link budget

Additive(+)/Negative(−) Component Symbol
+ Transmitter power Pt
+ Transmit-antenna gain Gt
− Median-link-propagation loss L50
− Margin M dB
+ Mobile-receive antenna gain Gr
− Cabling losses Lc
− Receiver (noise + interference) N + I
= Required SNR SNRreq

Program:

```
clc;
clear all;
k=1.38*(10^(-23)); %Boltzmann Constant
T=273; % temperature
F=5; % Noise Figure

PSD=k*T*F; % Power Spectral Density

B=30*(10^3):(10^3):40*(10^3); % Bandwidth

NP=10*log10(PSD*B); %Noise Power in db
BER=10^(-4);  %Bit Error Rate


% Calculation of Signal To Noise Ratio
x=(1-(2*BER));
display(x);
SNR=((x^2)/(1-(x^2)))*2;
display(SNR);
SNRdB=10*log10(SNR);
display(SNRdB);

%Given Data
Gt=12;
Gr=5;
L=167;
M=10;
LC=3;

%Transmitted Power
Pt=SNRdB-Gt-Gr+L+M+LC+NP;
display(Pt);
plot(B,Pt,'r');
xlabel('Bandwidth');
ylabel('Transmitted Power');
```

Output:

Conclusion:

Que 1: What is Path loss exponent ? Write mathematical relation with respect to received power

Ans1 :

Que 2 : How to compute receiver noise power ?

Ans2 :

EXPERIMENT NO. 5

Title:

BER performance of multi-antenna Rayleigh channel for SNR varying from 0 to 60 dB.

Aim:

Simulate BER performance of multi-antenna Rayleigh channel for SNR varying from 0 to 60 dB.

CO 308.2: Discriminate OFDM and MIMO technologies used for wireless communication.

ELO-2: Students will be able to perform BER performance of multi antennachannel and compute Doppler shift.

Pre-requisites: Basic Communication system.

Hardware & Software used: MATLAB/Octave, PC and Printer

Theory:

BER in Wireless Communication Systems:

The instantaneous BER (a) for a particular value of the amplitude a of the Rayleigh fading channel coefficient h is given by the standard Gaussian Q function as

$$\mathrm{BER}\,(a) = Q\left(\sqrt{\frac{a^2 P}{\sigma_n^2}}\right) = Q\left(a\sqrt{\frac{P}{\sigma_n^2}}\right)$$
$$= \frac{1}{\sqrt{2\pi}}\int_{a\sqrt{\frac{P}{\sigma_n^2}}}^{\infty} e^{-x^2/2}dx = \frac{1}{\sqrt{2\pi}}\int_{a\sqrt{\mu}}^{\infty} e^{-x^2/2}dx \qquad (1)$$

where the quantity μ has been defined as the SNR, i.e., μ = Pσ2n
to simplify the notation. Substituting xa√μ = t, the above integral can be simplified using standard calculus as

$$\mathrm{BER}\,(a) = \int_{1}^{\infty} e^{-\mu a^2 t^2/2} a\sqrt{\mu}\,dt \qquad (2)$$

Observe that the above expression for BER in the wireless channel depends on the instantaneous amplitude a of the fading coefficient, which is a random quantity. Thus, the above BER itself is random in nature, i.e., it is low for high values of a and vice versa.

BER in Multi-Antenna Wireless Systems:

The final expression for the BER of this multi-antenna wireless system with MRC is given as

$$\mathrm{BER}_{\text{Multi-Antenna}} = \left(\frac{1-\lambda}{2}\right)^{L} \sum_{l=0}^{L-1} {}^{L+l-1}C_l \left(\frac{1+\lambda}{2}\right)^{l} \qquad (3.17)$$

where the quantity λ is defined as $\lambda = \sqrt{\frac{\mathrm{SNR}}{2+\mathrm{SNR}}}$, with SNR denoting the average receiver SNR for each link given by $\frac{P}{\sigma_n^2}$ and ${}^{n}C_k = \frac{n!}{k!(n-k)!}$. Notice that for $L = 1$, i.e., the case with a single receive antenna, the above expression reduces to

$$\begin{aligned}\mathrm{BER} &= \left(\frac{1-\lambda}{2}\right)^{1} \sum_{l=0}^{0} {}^{0}C_l \left(\frac{1+\lambda}{2}\right)^{l} = \frac{1}{2}(1-\lambda)\\ &= \frac{1}{2}\left(1-\sqrt{\frac{\mathrm{SNR}}{2+\mathrm{SNR}}}\right)\end{aligned}$$

Program:

```
clc;
clear all;
clc;
clear all;
N=1000000;
L=4;
for SNRdb=[0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32
33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60];
    SNR=10.^(SNRdb/10);
    display(SNR);
    p=(SNR/(2+SNR)).^0.5;
    display(p);
    q=((1-p)/2).^L;
    %r=((1+p)/2);

    for index=0:1:L-1;
      n=L+index-1;
      display(n);
      k=index;
      display(index);
      n_factorial=factorial(n);
      k_factorial=factorial(k);
      display(k_factorial);
```

```
    n_minus_k_factorial=factorial(n-k);
    combi=n_factorial/(k_factorial.* n_minus_k_factorial);
    %syms index
    f = combi.*(((1+p)/2).^index);
    V = subs(f, k, 0:3)
    BER = sum(V)
    display(BER);

    %syms index;
    %BER = symsum((combi.*(r.^index)),index,0,3);
    BERdb=10.*log10(BER);
    BERdb_perbit=BERdb/N;
    display(BERdb_perbit);
    %plot(SNRdb,BER,'r');

   end
   ezplot(BERdb_perbit,[0,60]);
   hold on;
end

 xlabel('SNR in Decibel');
 ylabel('BER in Decibel per bit');
 grid on;
```

Output:

Conclusion:

Que 1: Compare the BER over a Rayleigh fading wireless channel with that of wire line AWGN channel
ANS:

Que 2: What is diversity in wireless communication?
ANS:

EXPERIMENT NO.6

Title:

Compute doppler shift of the received signal for different carrier frequency of mobile generations by considering vehicle is moving at 60 miles per hour at an angle of 30 degree with the line joining the base station.

Aim:

Compute doppler shift of the received signal for different carrier frequency of mobile generations by considering vehicle is moving at 60 miles per hour at an angle of 30 degree with the line joining the base station.

CO 308.3: Analyze Cellular radio network architecture for mobile communication

ELO-2: Students will be able to perform Link budget analysis, BER performance of multi antenna channel and compute Doppler shift.

Pre-requisites: Basic Communication system.

Hardware & Software used: MATLAB/Octave, PC and Printer

Theory:

Doppler Fading in Wireless Systems:

Another unique aspect of communication over wireless channels is the Doppler fading nature of such channels. The Doppler shift is a fundamental principle related to the electromagnetic radio-wave propagation. In this context, the Doppler shift associated with an electromagnetic wave is defined as the perceived change in the frequency of the wave due to relative motion between the transmitter and receiver. This is schematically shown in Figure 1.

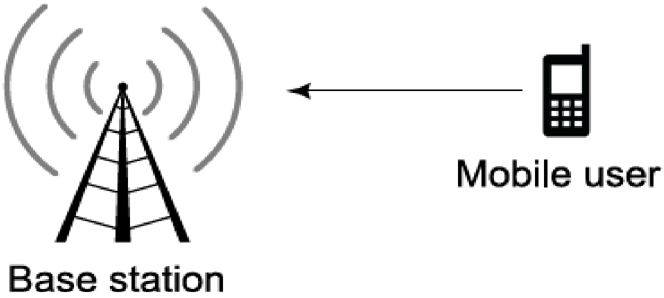


Figure 1: Doppler fading due to user mobility

The perceived frequency is higher than the true frequency if the transmitter is moving towards the receiver and lower otherwise. Doppler fading is inherent in wireless communications due to the untethered nature of mobile transceivers, which enables mobility in wireless systems, leading to relative motion between the transmitter and the receiver. This is different compared to the conventional wired communications, where the tethered nature of the fixed radio-access medium does not allow for mobility.

Doppler Shift Computation:

The mobile station is moving with a velocity v at an angle θ with the line joining the mobile and base station. Let the carrier frequency be fc. The Doppler shift for this scenario is given as

$$f_d = \left(\frac{v}{c}\cos\theta\right) f_c \tag{1}$$

where c = 3 × 108m/s is the velocity of light, i.e., velocity of an electromagnetic wave in free space. It can be clearly seen that the Doppler shift increases with the velocity v. Moreover, it depends critically on the angle θ between the direction of motion and the line joining the transmitter and receiver.

Program:

```
clc;
clear all;
speed_miles=60; % Given Speed in miles per hour
speed_km=60*1.6; % conversion to km per hour
speed_meter_per_sec=speed_km*(5/18); % conversion to m/s
c=3*(10^8);       %speed of light in m/s
fd=(speed_meter_per_sec/c)*cos(pi/6)*(1850*10^6); % Doppler Shift frequency
display(fd);
```

Conclusion:

Output:

Que 1 : What is Doppler shift ?

Que 2: Explain Doppler impact on wireless channel

EXPERIMENT NO. 7

Title: Study of Network Simulator Tools.

Aim: Study mobile environment to evaluate performance parameters using any open source Network Simulator tool.

CO 308.6: Specify different issues in wireless communication system for performance analysis

ELO 3: Students will be able to evaluate performance parameters using open source network simulator tool.

Pre-requisites: Basic Communication system.

Hardware & Software used: MATLAB/Octave, PC and Printer

Theory:

Simulation is a very important modern technology. It can be applied to different science, engineering, or other application fields for different purposes. Computer assisted simulation can model hypothetical and real-life objects or activities on a computer so that it can be studied to see how the system function. Different variables can be used to predict the behavior of the system. Computer simulation can be used to assist the modeling and analysis in many natural systems. Typical application areas include physics, chemistry, biology, and human-involved systems in economics, finance or even social science. Other important applications are in the engineering such as civil engineering, structural engineering, mechanical engineering, and computer engineering. Application of simulation technology into networking area such as network traffic simulation.

Basic concepts in network simulation

In the area of computer and communications networks, simulation is a useful technique since the behavior of a network can be modeled by calculating the interaction between the different network components (they can be end-host or network entities such as routers, physical links or packets) using mathematical formulas. They can also be modeled by actually or virtually capturing and playing back experimental observations from a real production networks. After we get the observation data from simulation experiments, the behavior of the network and protocols supported can then be observed and analyzed in a series of offline test experiments. All kinds of environmental attributes can also be modified in a controlled manner to assess how the network can behave under different parameters combinations or different configuration conditions. Another characteristic of network simulation that worth noticing is that the simulation program can be used together with different applications and services in order to observe end-to-end or other point-to-point performance in the networks.

Type of network simulators

For network protocol designers, it is often difficult to decide which simulator to choose for a particular task. Therefore, we conduct a survey to find a network simulator that provides a good balance between availability of ready to use models, scripting and language support, extendibility, graphical support, easiness of use, etc. The survey is based on a collection of a number of criteria including published results, interesting characteristics and features. From our survey results, we broadly categories network simulators as: "Widely Used" simulators and "Other" simulators. The network simulators taken into consideration as "Widely Used" are Ns-2, Ns-3, GloMoSim, J-Sim, OMNet++, OPNet, and QualNet.

1. OPNET (Optimized Network Evaluation Tool):

OPNET 's software environment is called Modeler, which is specialized for network research and development. It can be flexibly used to study communication networks, devices, protocols, and applications. Because of the fact of being a commercial software provider, OPNET offers relatively much powerful visual or graphical support for the users. The graphical editor interface can be used to build network topology and entities from the application layer to the physical layer. Object-oriented programming technique is used to create the mapping from the graphical design to the implementation of the real systems. An example of the graphical GUI of OPNET can be seen in Figure 1. We can see all the topology configuration and simulation results can be presented very intuitively and visually. The parameters can also be adjusted and the experiments can be repeated easily through easy operation through the GUI.

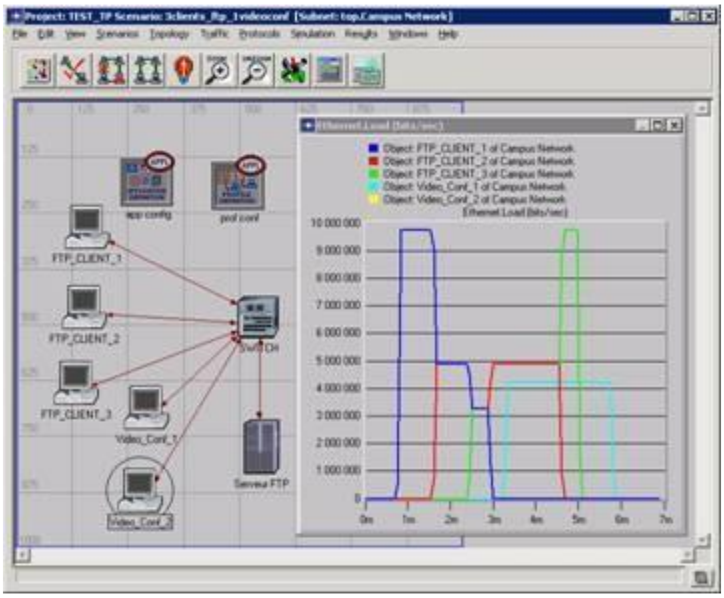


Figure 1. OPNET GUI

Main features:

OPNET inherently has three main functions: modeling, simulating and analysis. **For modeling**, it provides intuitive graphical environment to create all kinds of models of protocols. **For simulating**, it uses three different advanced simulations technologies and can be used to address a wide range of studies. **For analysis**, the simulation results and data can be analyzed and displayed very easily. User friendly graphs, charts, statistics, and even animation can be generated by OPNET for users ' convenience.

2.Network Simulator 2 (NS2)

NS2 is one of the most popular open source network simulators. The original NS is a discrete event simulator targeted at networking research. NS2 is the second version of NS (Network Simulator). NS is originally based on REAL network simulator. The first version of NS was developed in 1989 and evolved a lot over the past few years. The current NS project is supported through DARPA. The current second version NS2 is widely used in academic research and it has a lot of packages contributed by different non-benefit groups. An example of the graphical GUI of NS2 can be seen in Figure 2.

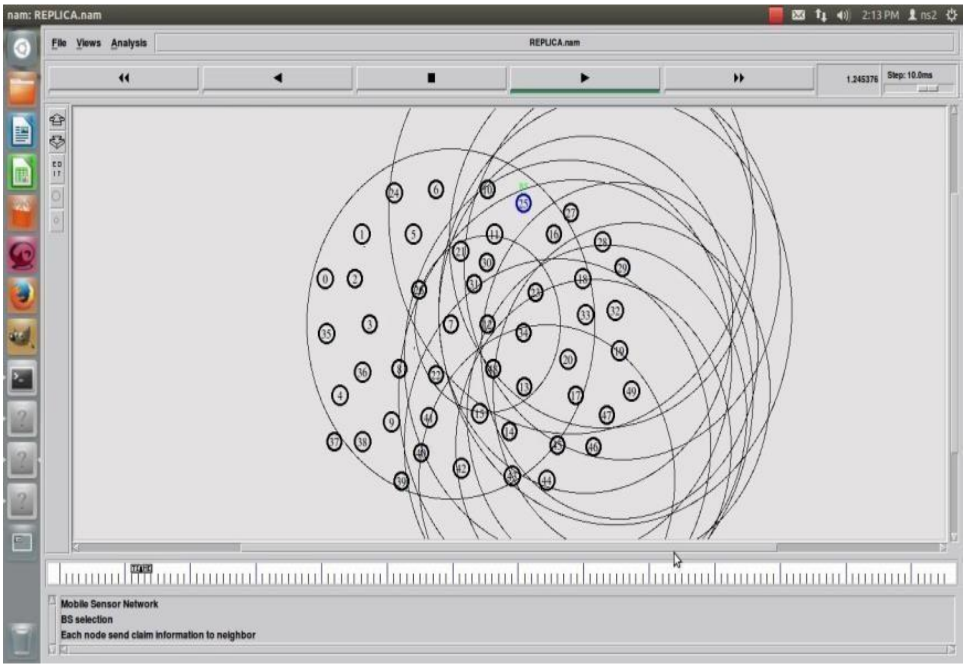


Figure 2. NS2 GUI

Main features:

First and foremost, NS2 is an object-oriented, discrete event driven network simulator which was originally developed at University of California-Berkely. The programming it uses is C++ and OTcl (Tcl script language with Object-oriented extensions developed at MIT). The usage of these two programming language has its reason. The biggest reason is due to the internal characteristics of these two languages. C++ is efficient to implement a design but it is not very easy to be visual and graphically shown. It's not easy to modify and assembly different components and to change different parameters without a very visual and easy-to-use descriptive language. Moreover, f or efficiency reason, NS2 separates control path implementations from the data path implementation. The event scheduler and the basic network component objects in the data path are written and compiled using C++ to reduce packet and event processing time. OTcl happens to have the feature that C++ lacks. So the combination of these two languages proves to be very effective. C++ is used to implement the detailed protocol and OTcl is used for users to control the simulation scenario and schedule the events. A simplified user's view of NS2 is shown in Figure 3. The OTcl script is used to initiate the event scheduler, set up the network topology, and tell traffic source when to start and stop sending packets through event scheduler. The scenes can be changed easily by programming in the OTcl script. When a user wants to make a new network object, he can either write the new object or assemble a compound object from the existing object library, and plumb the data path through the object. This plumbing makes NS2 very powerful.

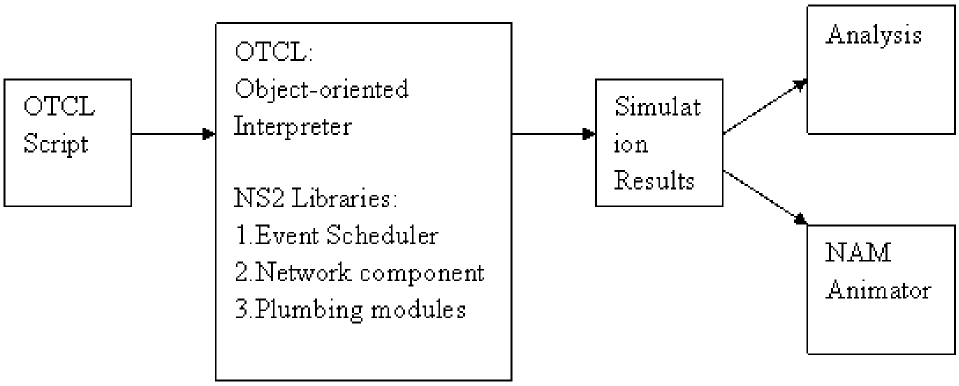


Figure 3. Simplified User's View o f NS2

3. <u>Network Simulator 3 (NS3)</u>

Similar to NS2, NS3 is also an open sourced discrete-event network simulator which targets primarily for research and educational use. NS3 is licensed under the GNU GPLv2 license, and is available for research and development. NS3 is designed to replace the current popular NS2. However, NS3 is not an updated version of NS2 since that NS3 is a new simulator and it is not backward-compatible with NS2.

4. <u>OMNeT++</u>

OMNeT++ has generic and flexible architecture which makes it successful also in other areas like the IT systems, queuing networks, hardware architectures, or even business processes as well. It is similar with NS2 and NS3, OMNeT++ is also a public-source, component-based network simulator with GUI support. Its primary application area is communication networks. Like NS2 and NS3, OMNeT++ is also a discrete event simulator. It is a component-based architecture. Components are also called modules and are programmed in C++. The components are then assembled into larger components and models by using a high-level language. Its function is similar to that of OTcl in NS2 and Python in NS3. OMNeT++ also provides GUI support, and due to its modular architecture, the simulation kernel can be embedded into all kinds of different user s' applications. Figure 5 is an OMNeT++ GUI screenshot.

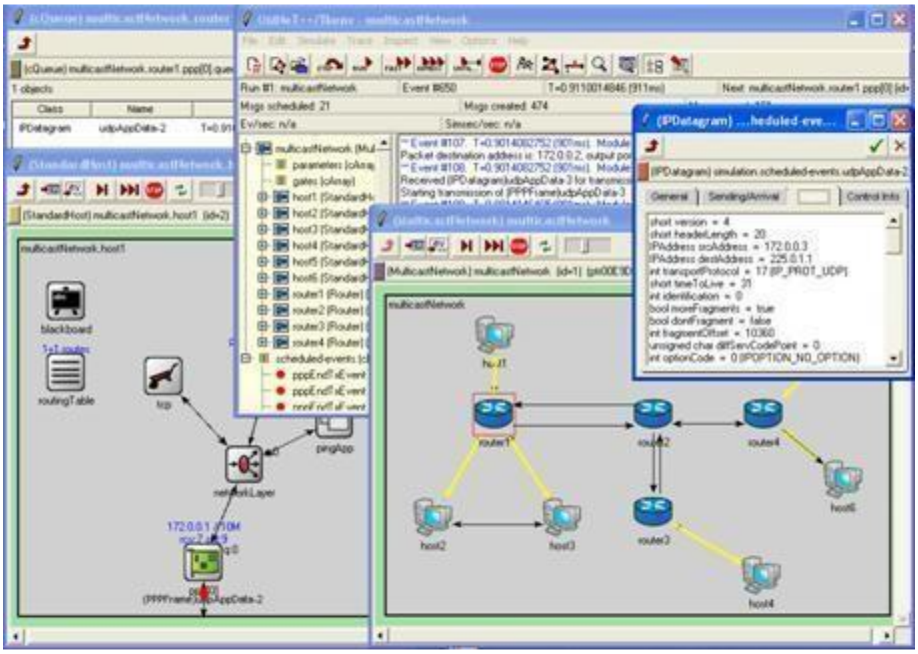


Figure 5. OMNeT++ GUI

Main features:

Since OMNeT++ is designed to provide a component-based architecture, the models or modules of OMNeT++ are assembled from reusable components. Modules are reusable and can be combined in various ways which is one of the main features of OMNeT++.

5. Global Mobile Information System Simulator (Glo-MoSim)

It is a parallel discrete event simulation soft-ware[4] that simulates wireless and wired network sys-tems. It is designed as a set of library modules, each of which simulates a specific wireless communication pro-tocol in the protocol stack. It assumes that the network is decomposed into a number of partitions and a single entity is defined to simulate a single layer of the com-plete protocol stack for all the network nodes that be-long to the partition. The parallel implementation of GloMoSim can be executed using variety of conservative synchronization protocols, which include the null mes-sage and conditional event algorithm. The library has been developed using PARSEC, a C based parallel simulation language. It uses the Parsec compiler to com-pile the simulation protocols. It has been designed to be extensible and comprehensible. GloMoSim aims to de-velop a modular simulation environment for protocol stack that is capable of scaling up networks with thou-sands of heterogeneous nodes. GloMoSim currently supports protocols for a purely wireless network.

Features:

- GloMoSim is a library-based sequential and parallel simulator for wireless networks.
- GloMoSim facilitates the ability to use in a parallel environment which distinguishes it from most other wireless network simulators.
- It allows the simulation Scalability to simulate net-works with a hundred and thousand of nodes.
- It supports various layers like Mobility, Radio Propagation, Radio Model, Packet reception models, Data Link, Network (Routing), Transport and Ap-plication. i.e. (It supports almost all the OSI layers with limited benefits).
- GloMoSim supports direct satellite communication, multi-hop wireless communication and most of the traditional internet protocols.
- It facilitates to build a library of parallelized mod-els that can be used for the evaluation of a varie-ty of wireless network protocols.

6. NetSim

NetSim is a stochastic discrete event network simula-tiontool used for network lab experimentation and research. Its leading network simulation software for protocol modeling and simulation, allowing us to ana-lyze computer networks with unmatched depth, power and flexibility. NetSim comes with an in-built develop-ment environment, which serves as the interface be-tween User's code and NetSim's protocol libraries and simulation kernel. It provides network performance metrics at various abstraction levels such as Network, sub-network, Node and a detailed packet trace. It has unique features and functionality. NetSimis available as Standard or Academic versions and is built on a com-mon design framework of high level architecture and code. In a word, NetSim is truly a fantastic product that is not only versatile, but also robust and provides those features that are hard to come with any simulators.

Features:

- NetSim modeling and simulation are supported for Aloha, Slotted Aloha, Token Ring/Bus, Ethernet CSMA/CD, Fast and Gigabit Ethernet, WLAN - IEEE 802.11 a/b/g/n and e, X.25, Frame Relay, TCP, UDP, IPv4 and IPv6, Routing - RIP, OSPF, BGP,MPLS, Wi-Max, MANET, GSM, CDMA, Wire-less Sensor Network, Zigbee, Cognitive radio.

- It simulates a wide variety of Cisco routers, including 2500 series, 2600 series, 2800 series, and 3600 series routers, as well as the Cisco Catalyst 1900 series, 2900 series, and 3500 series switches.
- Protocol libraries are available as open C code for user modification. This can help avoid the time consuming process of programming, customization and configuration commercial simulators to meet customer specific needs.
- Along with the Boson Virtual Packet Technology engine NetSim utilizes Boson's proprietary Network Simulator, Router Simulator, and EROUTER soft-ware technologies, to create individual packets. These packets are routed and switched through the simulated network, allowing NetSim to build an appropriate virtual routing table and simulate true networking. Other simulation products on the mar-ket do not support this level of functionality.
- It can be used to create a simulation of the topology of corporate network and help practice trouble-shooting without using devices on the production network.

Comparison of simulators based on General Information

Sr. No	Name	License Type	Language	Supported Operating	GUI port
1	**Ns2**	Open source	C++ and OTCL	GNU/Linux,FreeBS D, Mac OS X, Win-dows XP, Windows Vista and Win. 7.	Limited
2	**Ns3**	Open source	C++and Optional Python Bindings	GNU/Linux,FreeBS D, Mac OS X, Win-dows XP, WindowsVista and Win. 7.	Yes
3	**QualNet**	Commercial (Separate license for academicians and others)	C++	UNIX, Window, MAC, Linux	Yes
4	**GloMoSim**	Open source	C	Windows, Linux,Sun SPARC Solaris	Limited
5	**NetSim**	Proprietary	C and Java	Windows (7, Vista)and windows XP	Yes
6	**OMNET++**	Open source (for study an dresearch	C++	Windows XP or Later, Linux, Mac OS X,	Yes

Comparison of simulators based on the properties of simulators

Sr.No	Name	Simulation Event Type	Available Module	Scalability	Number of nodesupport	Paral-lelism
1	NS2	Discrete-event	Wired,Wireless, AdHoc and Wireless Sensor Networks	Limited	Up to 3000	No
2	NS3	Discrete-event	Wired,Wireless, Adhoc and Wireless Sensor Networks	Limited	Up to 3000	No
3	QualNet	Discrete- event	Wired &Wireless(like WiFi, Sensor net-work,MANET,WIMAX)network	Large	500-20000	Yes(S MP/Beowulf
4	GloMoSim	Discrete-event	Wired, Wireless & Ad-Hoc Networks. But currently pure wireless support	Large	Up to 10,000	Yes(S MP/Beowulf
5	NetSim	Stochastic Discrete-	Wired &Wireless, sensor network Event	Large Enough	-------	-------
6	OMNET++	Discrete-(wireless LAN, WiMAX)	Wired, Wireless, Ad-hoc and WSN.	Enough	--------	MPI/P

Installation of Network Simulator (NS2).

To install the NS2 software and test it with simple example on Ubuntu Operating System.

Steps for Installing Network Simulator 2 (NS2) on Ubuntu

Step 1: Download and Extract ns2

Download the all in one package for ns2 from internet. The package downloaded will be named "ns-allinone-2.35.tar.gz". Copy it to the home folder. Then in a terminal use the following two commands to extract the contents of the package:

tar -xvzf ns-allinone-2.35.tar.gz

All the files will be extracted into a folder called "ns-allinone-2.35".

Step 2: Building the dependencies

1. Ns2 requires a few packages to be pre installed. It also requires the GCC- version 4.3 to work correctly. So install all of them by using the following command:

```
sudo apt-get install build-essential autoconf automake libxmu-dev
```

2. One of the dependencies mentioned is the compiler GCC-4.3, which is no longer available, and thus we have to install GCC-4.4 version. The version 4.4 is the oldest we can get. To do that, use the follwoing command:

```
sudo apt-get install gcc-4.4
```

3. Once the installation is over, we have to make a change in the "ls.h" file. Use the following steps to make the changes: Navigate to the folder "linkstate", use the following command. Here it is assumed that the ns folder extracted is in the home folder of your system.

```
cd ~/ns-allinone-2.35/ns-2.35/linkstate
```

4. Now open the file named "ls.h" and scroll to the 137th line. In that change the word "error" to "this->error". The image below shows the line 137 (highlightedin the image below) after making the changes to the ls.h file. To open the file use the following command:

```
gedit ls.h
```

```
// this next typedef of iterator seems extraneous but is required by gcc-2.96
typedef typename map<Key, T, less<Key> >::iterator iterator;
typedef pair<iterator, bool> pair_iterator_bool;
iterator insert(const Key & key, const T & item) {
        typename baseMap::value_type v(key, item);
        pair_iterator_bool ib = baseMap::insert(v);
        return ib.second ? ib.first : baseMap::end();
}

void eraseAll() { this->erase(baseMap::begin(), baseMap::end()); }
T* findPtr(Key key) {
        iterator it = baseMap::find(key);
        return (it == baseMap::end()) ? (T *)NULL : &((*it).second);
}
```

5. Save that file and close it.
6. Now there is one more step that has to be done. We have to tell the ns which version of GCC will be used. To do so, go to your ns folder and type the following command:

sudo gedit ns-allinone-2.34/otcl-1.13/Makefile.in

7. In the file, change Change CC= @CC@ to CC=gcc-4.4, as shown in the image below:

```
*Makefile.in ×

#
# try ./configure first to fill in all the definitions corresponding
# to your system, but you always can edit the sections below manually.
#

CC=             gcc-4.4
CFLAGS=         @CFLAGS@
RANLIB=         @RANLIB@
INSTALL=        @INSTALL@

#
# how to compile, link, and name shared libraries
#

SHLIB_LD=       @SHLIB_LD@
SHLIB_CFLAGS=   @SHLIB_CFLAGS@
SHLIB_SUFFIX=   @SHLIB_SUFFIX@
SHLD_FLAGS=     @DL_LD_FLAGS@
DL_LIBS=        @DL_LIBS@
```

8. Save that file and close it.

Step 3: Installation

1. Now we are ready to install ns2. To do so we first require root privileges and then we can run the install script. Use the following two commands:

cd ns-allinone-2.35

./install

2. After successful installation following message will be shown on the screen. Ns-allinone package has been installed successfully.

Ns-allinone package has been installed successfully.

Here are the installation places:

tcl8.5.10: /home/sitams/ns-allinone-2.35/{bin,include,lib}

tk8.5.10: /home/sitams/ns-allinone-2.35/{bin,include,lib}

otcl: /home/sitams/ns-allinone-2.35/otcl-1.14

tclcl: /home/sitams/ns-allinone-2.35/tclcl-1.20

ns: /home/sitams/ns-allinone-2.35/ns-2.35/ns

nam: /home/sitams/ns-allinone-2.35/nam-1.15/nam

xgraph: /home/sitams/ns-allinone-2.35/xgraph-12.2

gt-itm: /home/sitams/ns-allinone-2.35/itm, edriver, sgb2alt, sgb2ns, sgb2comns, sgb2hierns

----

Please put /home/sitams/ns-allinone-2.35/bin:/home/sitams/ns-allinone-2.35/tcl8.5.10/unix:/home/sitams/ns-allinone-2.35/tk8.5.10/unix into your PATH environment; so that you'll be able to runitm/tclsh/wish/xgraph.

IMPORTANT NOTICES:

(1) You MUST put /home/sitams/ns-allinone-2.35/otcl-1.14, /home/sitams/ns-allinone-2.35/lib,
into your LD_LIBRARY_PATH environment variable.

If it complains about X libraries, add path to your X librariesinto LD_LIBRARY_PATH.

If you are using csh, you can set it like:

setenv LD_LIBRARY_PATH

If you are using sh, you can set it like:

export LD_LIBRARY_PATH=

(2) You MUST put /home/sitams/ns-allinone-2.35/tcl8.5.10/library into your TCL_LIBRARY environmental variable. Otherwise ns/nam will complain during startup.

After these steps, you can now run the ns validation suite withcd ns-2.35; ./validate

For trouble shooting, please first read ns problems page http://www.isi.edu/nsnam/ns/ns-problems.html. Also search the ns mailing list archive for related posts.

Step 4: Setting the Environment Path

1. The final step is to tell the system, where the files for ns2 are installed or present. To do that, we have to set the environment path using the ".bashrc" file. In that file, we need to add a few lines at the bottom. The things to be added are given below:

   sudo gedit ~/.bashrc

2. Lines to be added:

   # LD_LIBRARY_PATH

   OTCL_LIB=/home/sitams/ns-allinone-2.35/otcl-1.14

NS2_LIB=/home/sitams/ns-allinone-2.35/lib X11_LIB=/usr/X11R6/lib USR_LOCAL_LIB=/usr/local/lib

export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:$OTCL_LIB:$NS2_LIB:$X11_LIB:$USR_LOCAL_LIB

TCL_LIBRARY

TCL_LIB=/home/sitams/ns-allinone-2.35/tcl8.5.10/libraryUSR_LIB=/usr/lib

export TCL_LIBRARY=$TCL_LIB:$USR_LIB# PATH

XGRAPH=/home/sitams/ns-allinone-2.35/bin:/home/sitams/ns-allinone-2.35/tcl8.5.10/unix:/home/sitams/ns-allinone-2.35/tk8.5.10/unix

#the above two lines beginning from xgraph and ending with unix should come on the same line

NS=/home/sitams/ns-allinone-2.35/ns-2.35/
NAM=/home/sitams/ns-allinone-2.35/nam-1.15/
PATH=$PATH:$XGRAPH:$NS:$NAM

3. Once the changes have been made, save the file and restart the system.

Step 5: Running ns2

1. Once the system has restarted, open a terminal and start ns2 by using the following command:

ns

2. System returns

%

Step 6: Testing ns2

Test the addition of two integer numbers.

```
% set a
1010
% set b
2020
% expr
$a+$b30
%
```

Conclusion:

Que1 : list various network simulator tools ?

Que 2: State advantages of Network simulator

EXPERIMENT NO.8

Title: Simulate the OFDM system and evaluate frame error rate against SNR

Aim: Simulate the OFDM system and evaluate frame error rate against SNR

CO 308.2: Discriminate OFDM and MIMO technologies used for wireless communication

ELO 2 : Students will be able to perform Link budget analysis, BER performance of multi antenna channel and compute Doppler shift.

Pre-requisites: Basic Communication system.

Hardware & Software used: MATLAB/Octave, PC and Printer

Theory:

Orthogonal Frequency-Division Multiplexing (OFDM) forms the basis for 4G, i.e., Fourth Generation wireless communication systems. OFDM is used in 4G wireless cellular standards such as Long-Term Evolution (LTE) and WiMAX (Worldwide Interoperability for Microwave Access). OFDM is a key broadband wireless technology which supports data rates in excess of 100 Mbps. Similarly, the wireless local area (LAN) standards such as 802.11 a/g/n are based on OFDM.

An OFDM system is defined by IFFT/FFT length – N, the underlying modulation technique (BPSK/QPSK/QAM), supported data rate, etc. The FFT/IFFT length N defines the number of total subcarriers present in the OFDM system.

IEEE 802.11 standard [IEEE80211] specifies the following parameters for its OFDM physical layer. FFT/IFFT size = 64 (implies 64 subcarriers in total = used + unused = NFFT) Number of data subcarriers = 48 (Nd) Number of pilot subcarriers = 4 (Np) Derived parameters from the above specification are: Number of total USED subcarriers = 52 (Nu = Nd+ Np) Number of total UNUSED subcarriers = 12 (Nun = NFFT – Nu). According to the spec, these 52 used subcarriers are distributed in the following way. The 52 used subcarriers are named as 1, 2, 3… 26 and -1, -2, -3… -26. The used subcarriers 1 to 26 are mapped to 1 to 26 of IFFT inputs and the subcarriers -1,-2, ..., -26 are mapped to the IFFT inputs 38 to 63. The remaining IFFT inputs 27 to 37 and the input 0 (dc input) are set to "0". In this manner the 12 null subcarriers are mapped to IFFT inputs. The OFDM transmitter & receiver system which will be simulated in this experiment is as shown in fig 1

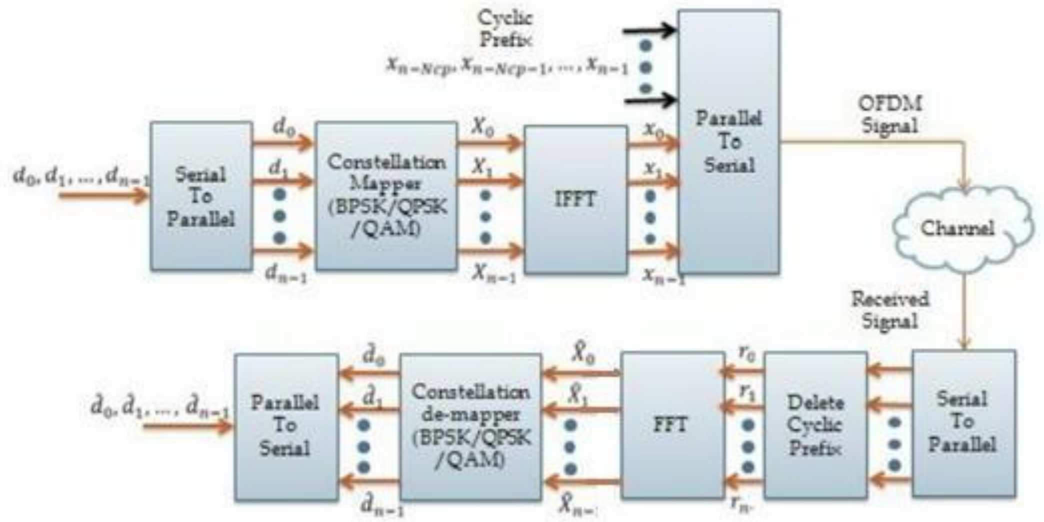


Fig. 1 : OFDM Transmitter & Receiver System

Arrangement of subcarriers: The IEEE 802.11 specification specifies how to arrange the given subcarriers. The 52 used subcarriers (data + pilot) are assigned numbers from -26,-25, …,-2,-1 and 1,2,…,25,26. The following figure illustrates the scheme of assigning these subcarriers to the IFFT inputs.

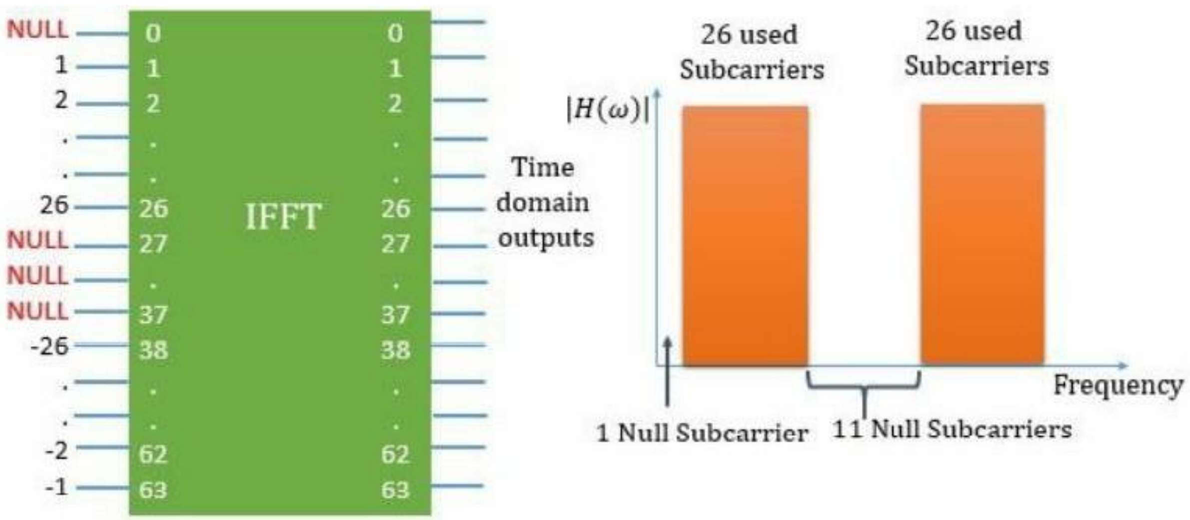


Fig: 2 Arrangement of subcarriers

For simulation consider IEEE 802.11 standard [IEEE80211] which specifies the following

parameters for its OFDM physical layer.

FFT/IFFT size = 64 (implies 64 subcarriers in total = used + unused =

NFFT)Number of data subcarriers = 48 (Nd)

Number of pilot subcarriers = 4 (Np)

Derived parameters from the above specification are:

Number of total USED subcarriers = 52 (Nu = Nd+

Np)

Number of total UNUSED subcarriers = 12 (Nun = NFFT – Nu).

Program:

Matlab Code:

```
%Simulation of OFDM system in an AWGN environment
clear; clc;
%--------Simulation parameters----------------
nSym=10^4; %Number of OFDM Symbols to transmit
EbN0dB = -20:2:8; % bit to noise ratio
%.....................................................................
%--------OFDM Parameters - Given in IEEE Spec--
N=64; %FFT size or total number of subcarriers (used + unused) 64
Nsd = 48; %Number of data subcarriers 48
Nsp = 4 ; %Number of pilot subcarriers 4
ofdmBW = 20 * 10^6 ; % OFDM bandwidth
%.......................................................................
%..............Derived Parameters..............................
deltaF = ofdmBW/N; %=20 MHz/64 = 0.3125 MHz
Tfft = 1/deltaF; % IFFT/FFT period = 3.2us
Tgi = Tfft/4;%Guard interval duration - duration of cyclic prefix
Tsignal = Tgi+Tfft; %duration of BPSK-OFDM symbol
Ncp = N*Tgi/Tfft; %Number of symbols allocated to cyclic prefix
Nst = Nsd + Nsp; %Number of total used subcarriers
nBitsPerSym=Nst; %For BPSK the number of Bits per Symbol is same as num of subcarriers
%.......................................................................

EsN0dB = EbN0dB + 10*log10(Nst/N) + 10*log10(N/(Ncp+N)); % converting to symbol to
noise ratio
errors= zeros(1,length(EsN0dB));
theoreticalBER = zeros(1,length(EsN0dB));
```

```
%Monte Carlo Simulation
for i=1:length(EsN0dB),
for j=1:nSym

%..........................Transmitter..............................
s=2*round(rand(1,Nst))-1; %Generating Random Data with BPSK modulation
%IFFT block
%Assigning subcarriers from 1 to 26 (mapped to 1-26 of IFFT input)
%and -26 to -1 (mapped to 38 to 63 of IFFT input); Nulls from 27 to 37
%and at 0 position
X_Freq=[zeros(1,1) s(1:Nst/2) zeros(1,11) s(Nst/2+1:end)];
% Pretending the data to be in frequency domain and converting to time domain
x_Time=N/sqrt(Nst)*ifft(X_Freq);
%Adding Cyclic Prefix
ofdm_signal=[x_Time(N-Ncp+1:N) x_Time];
%......................Channel Modeling ........................
noise=1/sqrt(2)*(randn(1,length(ofdm_signal))+1i*randn(1,length(ofdm_signal)));
r= sqrt((N+Ncp)/N)*ofdm_signal + 10^(-EsN0dB(i)/20)*noise;
%..........................Receiver.................................
%Removing cyclic prefix
r_Parallel=r(Ncp+1:(N+Ncp));
%FFT Block
r_Time=sqrt(Nst)/N*(fft(r_Parallel));
%Extracting the data carriers from the FFT output
R_Freq=r_Time([(2:Nst/2+1) (Nst/2+13:Nst+12)]);
%BPSK demodulation / Constellation Demapper.Force +ve value --> 1, -ve value --> -1
R_Freq(R_Freq>0) = +1;
R_Freq(R_Freq<0) = -1;
s_cap=R_Freq;
numErrors = sum(abs(s_cap-s)/2); %Count number of errors
%Accumulate bit errors for all symbols transmitted
errors(i)=errors(i)+numErrors;
end
theoreticalBER(i)=(1/2)*erfc(sqrt(10.^(EbN0dB(i)/10))); %Same as BER for BPSK over
AWGN
end
simulatedBER = errors/(nSym*Nst);
plot(EbN0dB,log10(simulatedBER),'r-o');
hold on;
plot(EbN0dB,log10(theoreticalBER),'k*');
grid on;
title('BER Vs EbNodB for OFDM with BPSK modulation over AWGN');

xlabel('Eb/N0 (dB)');ylabel('BER');legend('simulated','theoretical');
```

Output:

Conclusion:

Que 1: What if OFDM ?

Que 2: What is MIMO ?

Experiment No 9

Title: Simulate a cellular system with 48 channels per cell and blocking probability of 2%. Assume traffic per user is 0.04 E. What is the number of users that can be supported in a city of 603 km2 area if cell radios are changed in the steps of 500 m, 700m, 900 m, 1000 m 1200 m and 1500 m.

Aim: Simulate a cellular system with 48 channels per cell and blocking probability of 2%. Assume traffic per user is 0.04 E. What is the number of users that can be supported in a city of 603 km^2 area if cell radios are changed in the steps of 500 m, 700m, 900 m, 1000 m 1200 m and 1500 m.

CO 308.3: Analyze Cellular radio network architecture for mobile communication

ELO: Students will be able to perform Link budget analysis, BER performance of multi antenna channel and compute Doppler shift.

Pre-requisites: Basic Communication system.

Hardware & Software used: MATLAB/Octave, PC and Printer

Theory:

For N=48channels and blocking probability PB=0.02, it can be seen that the net offered load or supported traffic is given as A=38.4E.Also, since the total traffic A=NA0,wehave

$$N=A/A0=38.4/0.02=960$$

Further, from Figure 1, it can be seen that the area of a typical hexagonal cell of radius1km is given as

$$\text{Cell area}=6\times1\times1/\sqrt{3}=3.46km^2.$$

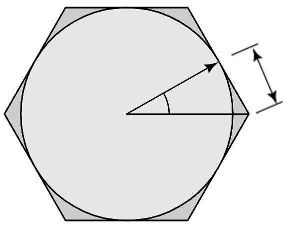


Figure1 *Hexagonal Area*

Therefore, the number of cells that are required to cover the entire city is given as

Number of cells=city area/cell area

=603/3.46=174

Therefore, the total number of users that can be supported by the cellular provider is given as

Number of users supported= Numberofcells×Numberofuserspercell≈174×960=167,040

Thus, approximately167,040 users can be supported by the cellular service provider in the given city for the desired blocking probability of 2%.

Program:

```
clc;
clear all;
Pb=0.02;
Ao=0.04;
r=0.5:0.2:1.5;
AreaCity=603;
A=38.4;
ang=pi/6;

NoUsersPerCell=A/Ao;
display(NoUsersPerCell);
len1=tan(ang)*r;
len=2*len1;
display(len);
AreaHexagon=3.*r.*len;
display(AreaHexagon);
NoCells=AreaCity./AreaHexagon;
display(NoCells);
NoUser=NoUsersPerCell.*NoCells;
display(NoUser);

plot(r,NoUser,'b');
xlabel('Radius of Hexagon');
ylabel('NoUser');
title('Graph for Estimating total number of Users In City ');
```

Output:

Conclusion:

Que 1: What is blocking probability ?

Que 2 : What id Erlang ?
