Course Project

Faculty of Engineering Ain Shams University.

Electronics & Communication Engineering department.

Wireless Communication Networks

ECE353s

Submitted to:

Dr. Michael Ibrahim

Eng. Nada Nader

Contents

Contributors:

- Islam Ahmed Mohammed Moukhtar 2000141
- Mohamed Hussein El-Bayoumy Fawzy 2001050
- Kebrianous Wagih Sadek 2000962
- Ahmed Abdou Abdallah Hassan 2000031

Part (A)

Introduction

It is required to design a simple planning tool for service provider that owns 340 channels over 900MHz band where the: **GOS, city area, user density, $\mathbf{SIR_{min}}$** and **sectorization method** are assigned then we get:

1) Cluster size (N).
2) Number of cells.
3) Cell radius (R).
4) Traffic intensity per cell and Traffic intensity per sector.
5) Base station transmitted power (P_{TX}).
6) A plot for the MS received power in dBm versus the receiver distance from the BS.

Procedures

1. We calculated cluster size given the SIR_{min} and the sectorization method.
2. To get cell radius, we must get the number of users per sector by dividing the traffic intensity per sector by the traffic intensity per user (A_u = 0.025 Erlang).
3. By dividing the traffic intensity per sector by the user density, which is assigned, we get the cell area and hence the cell radius (assuming the hexagonal shape for each cell).
4. Using the obtained cell area, we divide the city area by the cell area, and we finally get the number of cells.
5. By using the Hata model for propagation, we managed to calculate the transmitted power from the base station and plot the received power in **dBm** versus the distance from the base station in **Km**.

In order to implement this design, we divided it into several functions.

Implemented Functions

1) Cluster Size function

- It takes the SIR in dB and Sectorization method as input arguments.
- To calculate cluster size:

$$SIR_{ratio}= \frac{1}{i_o}\left(\sqrt{3N}-1\right)^4$$

$$\sqrt{3N}= \sqrt[4]{i_o SIR_{ratio}}+1$$

$$\boxed{N=\frac{1}{3}\left(\sqrt[4]{i_o SIR_{ratio}}+1\right)^2}$$

```
 function [N] = cluster_size_fn (SIR,sectorization)
 %input('please choose SIR in dB, & Choose sectorization
method:no_sectorization,120_sectorization, 60_sectorization')
 io=6;% intialize
 SIR_ratio=power(10,SIR/10); %find SIR in ratio
 if sectorization=="no_sectorization"
     io=6;
 elseif sectorization=="120_sectorization"
     io=2;
 elseif sectorization=="60_sectorization"
     io=1;
 else
     msgbox('invalid data ');
 end
 dummy=power(io*SIR_ratio,1/4); %dummy variable used in middle operations
 dummy=dummy + 1 ;
 dummy=power(dummy,2);
 N=dummy/3;
 N_vector=[];
 for i=0:ceil(N)
     for k=0:ceil(N)
         N_correct=power(i,2)+(i*k)+power(k,2);
         N_vector=[N_vector,N_correct];
     end
 end
 N_vector=sort(N_vector);
  for i=1:length(N_vector)
     if N_vector(i) >= N
         N=N_vector(i);
         break;
     end
  end
 end
```

2) Intensity Calculation function

- It takes GOS, cluster size (N) and sectorization as input arguments.
- Number of channels per cell is calculated as follows:

$$\text{No. of channels per cell}=\frac{\text{Total No. of channels}}{\text{N}}$$

Hence, $\text{No. of channels per sector}=\frac{\text{No. of channels per cell}}{i_o}$, where i_o depends on the sectorization method: **[1: for no sectorization, 3: for 120^o sectorization, 6: for 60^o sectorization].**

- Now we calculate the traffic intensity per sector by solving erlang-B equation:

$$\text{GOS}=\frac{\frac{A^C}{C!}}{\sum_{k=0}^{C}\frac{A^k}{k!}}$$

- Finally, we get the traffic intensity as follows:

$$\boxed{A_{cell}=A_{sector}*\text{No. of sectors}}$$

```
function [ACell,ASector]  = intesityCalculation(GOSGiven,N,sectorization)
       % S which is given
       totalNumberOfChannels = 340 ;

       % K which is calculated aftetr knowing the number of cells per cluster
       channelsPerCell = totalNumberOfChannels / N ;

       % Number of Channels per sector (sub cell)
       switch sectorization
           case 'no_sectorization'
               numberOfSectors = 1;
               c = floor(channelsPerCell);
           case '120_sectorization'
               numberOfSectors = 3;
               c = floor(channelsPerCell/3);
           case '60_sectorization'
               numberOfSectors = 6;
               c = floor(channelsPerCell/6);
       end
       % GOS Formula as a function of A
       GOSEquation = @(As,GOS) GOS - (As^c / factorial(c)) /  sum( As.^(0:c) ./
factorial( (0:c) ) )  ;

       % Solve for A
       options = optimoptions('fsolve', 'Diagnostics','off','Display', 'none');
       initial_guess = c ;
       ASector = fsolve(@(As) GOSEquation(As,GOSGiven) , initial_guess , options ) ;

       ACell = numberOfSectors*ASector ;
```

3) Radius Calculation function

- It takes user density, A_{sector} and sectorization method as input arguments.
- Given the traffic intensity per user A_u:

$$\text{No. of users per sector} = \frac{A_{sector}}{A_u}$$

$$Area_{sector} = \frac{\text{No.of users per sector (U)}}{\text{User Density}}$$

$$Area_{cell} = Area_{sector} * i_o$$

$$Area_{cell} = \frac{3\sqrt{3}}{2} R^2$$

$$\boxed{R = \sqrt{\frac{2*Area_{cell}}{3\sqrt{3}}}}$$

```
function cellRadius = radiusCalculation(userDensity,ASector,sectorization)

   % Based on sectorization method we can get the area of sector (sub cell)
   switch sectorization
        case 'no_sectorization'
            numberOfSectors = 1;
        case '120_sectorization'
            numberOfSectors = 3;
        case '60_sectorization'
            numberOfSectors = 6;
   end

    % Au is given (I assumed it per sector)
    Au = 0.025;

    % Calculate Number of users per sector
    numberOfUsersPerSector = floor(ASector / Au) ;

    % Sector area
    sectorArea = numberOfUsersPerSector / userDensity ;

    % Cell area
    cellArea = sectorArea * numberOfSectors ;

    % Radius of the cell in Km
    cellRadius = sqrt( 2*cellArea / (3*sqrt(3)) );

end
```

4) Number of cells function

- It takes cell radius and city area as input arguments.
- Using the obtained cell radius, we get the cell area then we get the number of cells:

$$\boxed{\text{No. of cells} = \frac{\text{City Area}}{\text{Area}_{\text{cell}}}}$$

- Note: In this function we assumed that there isn't any surrounding area that our given area is interfering on it.

```
function [number_of_Cells] = no_cells_fn(R,A) %R is the cell radius in km / A is the city area in km^2
cell_area=1.5*power(3,0.5)*power(R,2);
number_of_Cells=ceil(A/cell_area);
end
```

5) Power calculation function

- It takes cell radius and MS sensitivity as input arguments.
- Using Hata model for urban-city medium, we could get the path loss in dB as follows:

$$\boxed{C_H = 0.8 + (1.1 * log_{10}(f) - 0.7) * h_M - 1.56 * log_{10}(f)}$$

- C_H: The antenna correction factor
- f: The frequency in MHz

$$\boxed{\begin{gathered}P_{L,dB} = 69.55 + 26.16 * log_{10}(f) - 13.82 * log_{10}(h_B) - C_H \\ + (44.9 - 6.55 * log_{10}(h_B)) * log_{10}(R)\end{gathered}}$$

- Now after getting the path loss, we can get the transmitted power as follows:

$$\boxed{P_{TX,dBm} = P_{RX,dBm} + P_{L,dB}}$$

```
function transmittedPower = powerCalculation(prxmin,R)
% Given the transmission frequency is 900 MHZ
f = 900 ;

% BS height
hB = 20 ;

% MS height
hM = 1.5 ;

% Resource for the Hata model formulae : https://en.wikipedia.org/wiki/Hata_model
% Antenna height correction factor
CH = 0.8 + (1.1*log10(f)-0.7)*hM -1.56*log10(f);
% Path loss in urban areas. Unit: decibel (dB)
```

```
     LU = 69.55 + 26.16*log10(f) - 13.82*log10(hB) - CH + (44.9 -
6.55*log10(hB))*log10(R) ;
     transmittedPower = prxmin + LU ;
 end
```

6) Plot_PRX_function

- It takes the transmitted power in **dBm** and the reuse distance in **Km** as input arguments, then plots the received power versus the distance.

```
 function [] = plot_PRX(P_TX,D) %P_TX is in dBm
 Hm=1.5;%in meter
 Hb=20;%in meter
 f=900; %in MHZ
 d=0:D/1000:D;
 P_RX=zeros(length(d));
 CH=0.8 + (1.1 * log10(f) - 0.7) * Hm- 1.56 * log10(f);
 Lu=69.55 + 26.16*log10(f) - 13.82*log10(Hb) - CH + (44.9-6.55 * log10(Hb)) *
log10(d);
 % assume that anntena at TX and RX has unity gain
 P_RX=P_TX-Lu;% P_RX is in dBm
 plot(d,P_RX,'linewidth',1.5)
 xlabel('d in km');
 ylabel('PRX in dBm')
 grid on;
 end
```

The main Code

1) Cluster size

```
clear;
clc;
c=340; %# channels
f=900; %in MHZ
%GOS=input('Enter GOS: ');
%City_Area=input('Enter City area in Km^2: ');
%user_density=input('Enter user density in user/Km^2: ');
%SIR=input('Enter minimum SIR in dB: ');
%sectorization=input('Choose sectorization:no_sectorization,120_sectorization,
60_sectorization: ');
prompt = {'Enter GOS(ratio no percent): ','Enter City area in Km^2: ',...
    'Enter user density in user/Km^2: ','Enter minimum SIR in dB: ',...
    'Choose sectorization:no_sectorization,120_sectorization, 60_sectorization: '};
answer = inputdlg(prompt);
GOS=str2double(answer(1));
City_Area=str2double(answer(2));
user_density=str2double(answer(3));
SIR=str2double(answer(4));
sectorization=string(answer(5));
Hm=1.5; % in meters
Hb=20; % in meters
ms_sensitivity=-95; % in dBm
Au=0.025; %traffic intensity per user
N=cluster_size_fn(SIR,sectorization);
```

2) Number of cells

```
[ACell,ASector]  = intesityCalculation(GOS,N,sectorization);
R = radiusCalculation(user_density,ASector,sectorization);
number_of_Cells = no_cells_fn(R,City_Area);
```

3) Cell Radius

```
R = radiusCalculation(user_density,ASector,sectorization);
```

4) Traffic intensity per cell, and traffic intensity per sector

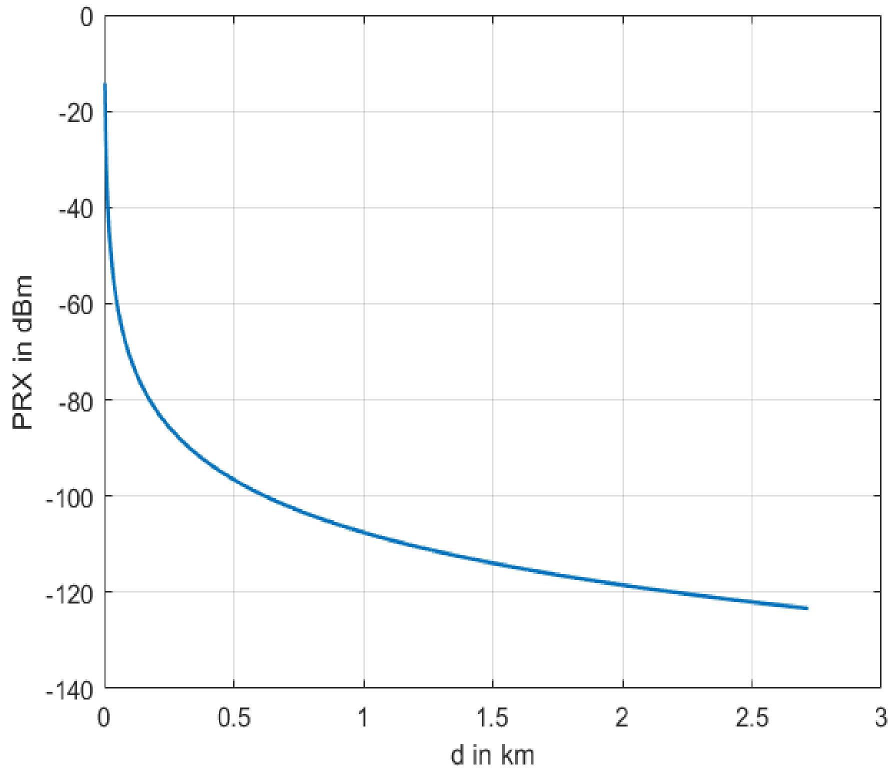
```
[ACell,ASector]  = intesityCalculation(GOS,N,sectorization);
```

5) Base station transmitted power

```
transmittedPower = powerCalculation(ms_sensitivity,R);
```

6) A plot for the MS received power in dBm versus the receiver distance from the BS

```
D=sqrt(3*N)*R; %reuse distance
plot_PRX(transmittedPower,D)
```



Printing results

```
msgbox({['cluster size = ',num2str(N)] ...
    ['number of cells = ',num2str(number_of_Cells)] ...
    ['Cell Radius  = ',num2str(R),' Km'] ...
    ['Traffic intensity per cell = ',num2str(ACell),' Erlang'] ...
    ['Traffic intensity per sector = ',num2str(ASector),' Erlang'] ...
    ['Base station transmitted power = ',num2str(transmittedPower),' dBm']})
```

We assigned the following parameters to our code as follows:

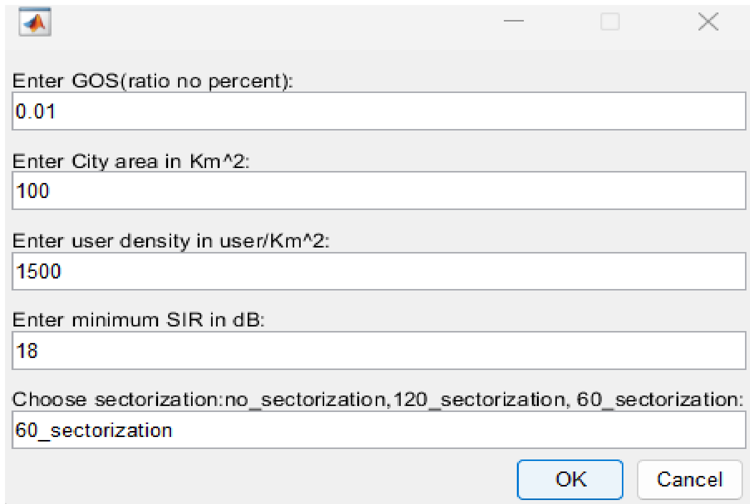
GOS = 1%

City area = 100 Km^2

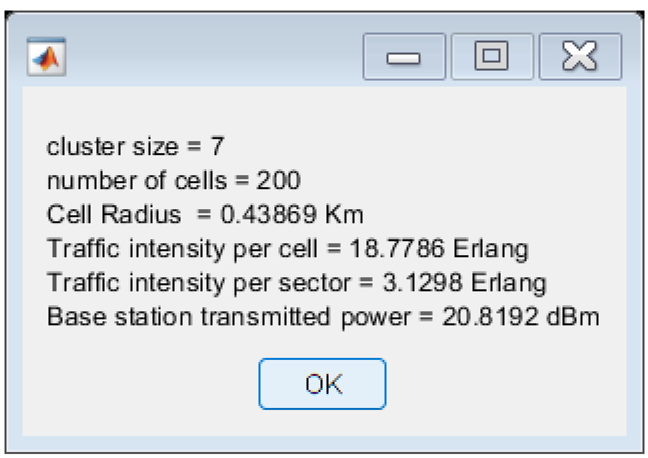
User Density = 1500 users/ Km^2

SIR = 18 dB

Sectorization method: 60_sectorization



Input to the tool



Output Results

Part (B)

The main Code

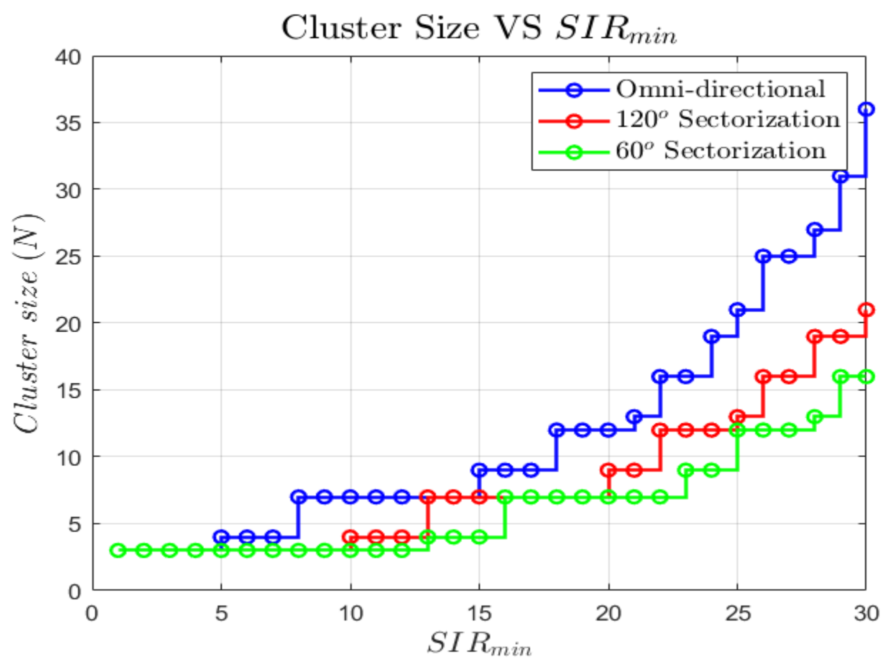
```
% %% %%%%%%%%%%%%%%%%%% Point (1) %%%%%%%%%%%%%%%%%% %%

% Given range of SIRmin
SIRmin = linspace(1,30,30);

% Initialize a vector for every sectorization method
N = zeros(size(SIRmin));

% Sectorization values
sectorization = ["no_sectorization","120_sectorization","60_sectorization"];

% Initiate a figure window
Nfig = figure('Name', 'N VS SIRmin', 'NumberTitle', 'off');
colors = ["bo-","ro-","go-"];

% Compute the vectors
for sectMethod = 1:length(sectorization)
    for SIRindex = 1 : length(SIRmin)
        N(SIRindex) =cluster_size_fn(SIRmin(SIRindex),sectorization(sectMethod));
    end
    % Plot using stairs
    stairs(SIRmin,N,colors(sectMethod),'LineWidth', 1.5);
    hold on
end
hold off

% Plot settings
xlabel('$SIR_{min}$', 'Interpreter', 'latex', 'FontSize', 14);
ylabel('$Cluster\ size\ (N)$', 'Interpreter', 'latex', 'FontSize', 14);
legend('Omni-directional', '$120^o$ Sectorization', '$60^o$ Sectorization', ...
       'Location', 'northeast', 'Interpreter', 'latex', 'FontSize', 12);
grid on;
title('Cluster Size VS $SIR_{min}$','Interpreter', 'latex', 'FontSize', 16);

 %%%%%%%%%%%%%%%%%% Point (2&3) %%%%%%%%%%%%%%%%%% %%
for SIR = [14,19]
    % Givens
    SIRmin = SIR ;
    userDesnity = 1400 ;
    cityArea = 100 ;
    % GOS range
```

```
     GOS = linspace(1,30,30) / 100;
    % Initialize a vector for A (cell traffic intensity)
    [ACell,ASector,NoCells] = deal(zeros(size(GOS)));

    % Sectorization values
    sectorization = ["no_sectorization","120_sectorization","60_sectorization"];

    % Initialize figures
    Afig = figure('Name', 'Cell traffic intensity VS GOS', 'NumberTitle', 'off');
    NumberOfCellsfig = figure('Name', 'Number of cells VS GOS', 'NumberTitle',
'off');

    % Plot colors
    colors = ["bd-","rp-","gh-"];

    for sectMethod = 1:length(sectorization)
        N = cluster_size_fn(SIRmin,sectorization(sectMethod));
        for GOSindex = 1 : length(GOS)
            [ACell(GOSindex),ASector(GOSindex)] =
intesityCalculation(GOS(GOSindex),N,sectorization(sectMethod));
            R =
radiusCalculation(userDesnity,ASector(GOSindex),sectorization(sectMethod));
            NoCells(GOSindex)= no_cells_fn(R,cityArea);
        end
        % Plot the traffic intensity
        figure(Afig)
        plot(GOS,ACell,colors(sectMethod), 'LineWidth', 1.5);
        hold on
        figure(NumberOfCellsfig)
        plot(GOS,NoCells,colors(sectMethod), 'LineWidth', 1.5);
        hold on
    end
    hold off
    hold off

    % Plots settings
    figure(Afig)
    xlabel('GOS', 'Interpreter', 'latex', 'FontSize', 14);
    ylabel('Traffic intensity per cell', 'Interpreter', 'latex', 'FontSize', 14);
    legend('Omni-directional', '$120^o$ Sectorization', '$60^o$ Sectorization', ...
           'Location', 'northeast', 'Interpreter', 'latex', 'FontSize', 12);
    grid on;
    title(sprintf('Traffic intensity per cell VS GOS for SIRmin = %d
dB',SIRmin),'Interpreter', 'latex', 'FontSize', 16);
     figure(NumberOfCellsfig)
    xlabel('GOS', 'Interpreter', 'latex', 'FontSize', 14);
    ylabel('Traffic intensity per cell', 'Interpreter', 'latex', 'FontSize', 14);
    legend('Omni-directional', '$120^o$ Sectorization', '$60^o$ Sectorization', ...
```

```
             'Location', 'northeast', 'Interpreter', 'latex', 'FontSize', 12);
      grid on;
      title(sprintf('Number of cells VS GOS for SIRmin = %d dB',
SIRmin),'Interpreter', 'latex', 'FontSize', 16);
 end

  %%%%%%%%%%%%%%%%%% Point (4&5) %%%%%%%%%%%%%%%%%% %%
 for SIR = [14,19]
     % Givens
     SIRmin = SIR ;
     GOS = 2/100 ;
     cityArea = 100 ;

     % GOS range
     userDesnity = linspace(100,2000,30);

     % Initialize a vector for A (cell traffic intensity)
     [ACell,ASector,NoCells,R] = deal(zeros(size(userDesnity)));

     % Sectorization values
     sectorization = ["no_sectorization","120_sectorization","60_sectorization"];

     % Initialize figures
     Rfig = figure('Name', 'Radius VS user density', 'NumberTitle', 'off');
     NumberOfCellsfig2 = figure('Name', 'Number of cells VS user density',
'NumberTitle', 'off');

     % Plot colors
     colors = ["bd-","rp-","gh-"];

     for sectMethod = 1:length(sectorization)
         N = cluster_size_fn(SIRmin,sectorization(sectMethod));
         for UDindex = 1 : length(userDesnity)
             [ACell(UDindex),ASector(UDindex)] =
intesityCalculation(GOS,N,sectorization(sectMethod));
             R(UDindex) =
radiusCalculation(userDesnity(UDindex),ASector(UDindex),sectorization(sectMethod));
             NoCells(UDindex)= no_cells_fn(R(UDindex),cityArea);
         end
         % Plot the traffic intensity
         figure(Rfig)


         plot(userDesnity,R,colors(sectMethod), 'LineWidth', 1.5);
         hold on
         figure(NumberOfCellsfig2)
         plot(userDesnity,NoCells,colors(sectMethod), 'LineWidth', 1.5);
         hold on
```

```
    end
    hold off

    % Plots settings
    figure(Rfig)
    xlabel('GOS', 'Interpreter', 'latex', 'FontSize', 14);
    ylabel('Traffic intensity per cell', 'Interpreter', 'latex', 'FontSize', 14);
    legend('Omni-directional', '$120^o$ Sectorization', '$60^o$ Sectorization', ...
           'Location', 'northeast', 'Interpreter', 'latex', 'FontSize', 12);
    grid on;
    title(sprintf('Cell Radius VS user density for SIRmin = %d
dB',SIRmin),'Interpreter', 'latex', 'FontSize', 16);


    figure(NumberOfCellsfig2)
    xlabel('GOS', 'Interpreter', 'latex', 'FontSize', 14);
    ylabel('Traffic intensity per cell', 'Interpreter', 'latex', 'FontSize', 14);
    legend('Omni-directional', '$120^o$ Sectorization', '$60^o$ Sectorization', ...
           'Location', 'northeast', 'Interpreter', 'latex', 'FontSize', 12);
    grid on;
    title(sprintf('Number of cells VS user density for SIRmin = %d dB',
SIRmin),'Interpreter', 'latex', 'FontSize', 16);
 end
```

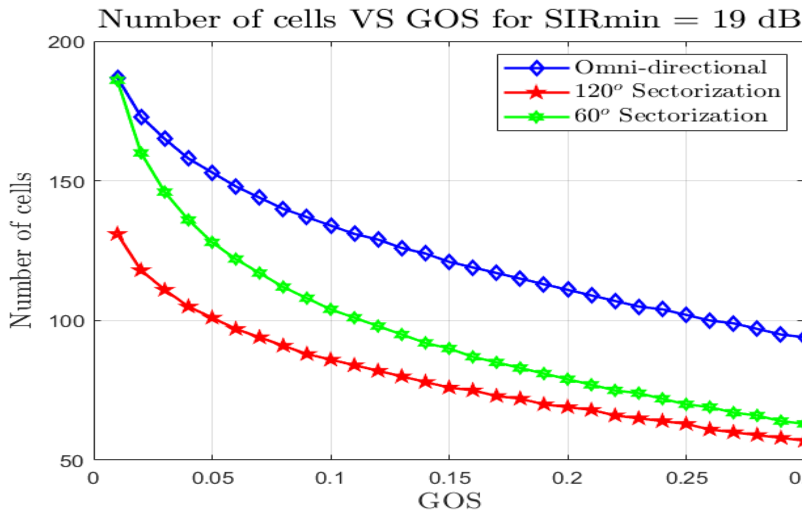
1. Plot the cluster size versus SIR_{min} with range from 1 dB to 30 dB.



- For constant cell radius, as we increase the SIR_{min} we increase the chance for interference which increases the cluster size.
- At constant SIR_{min}, as we have more sectors, we have less interferers so, the interference decreases for each cell so, we can accept smaller cluster which means smaller "N".

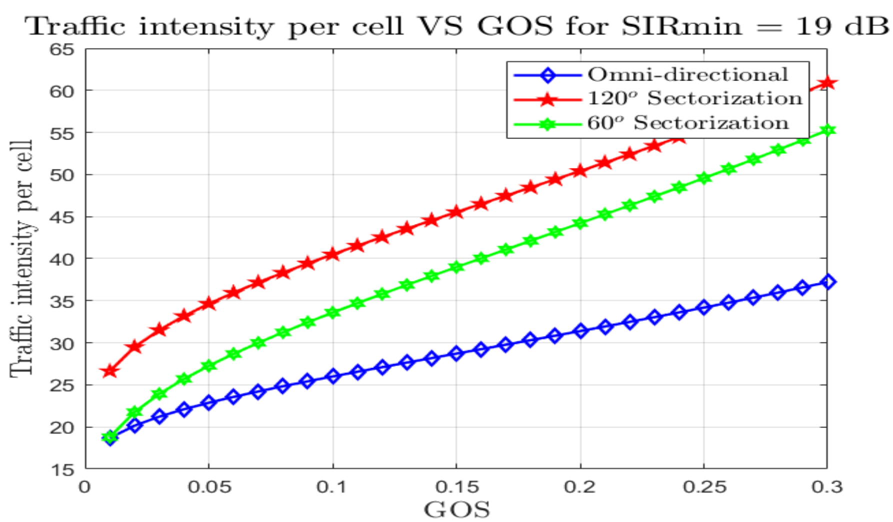
2. For SIR_{min}= 19 *dB* & user density= 1400 *users*/ km^2

(i) <u>Plot the **Number of cells** versus **GOS** (1% to 30%).</u>



- As the GOS increases, it means higher probability of blocking a call, so the capacity decreases hence the number of cells decreases by increasing the GOS.
- At a certain GOS, as we increase the sectors per cell, we effectively increase the capacity, so we need a smaller number of cells to achieve the same GOS.

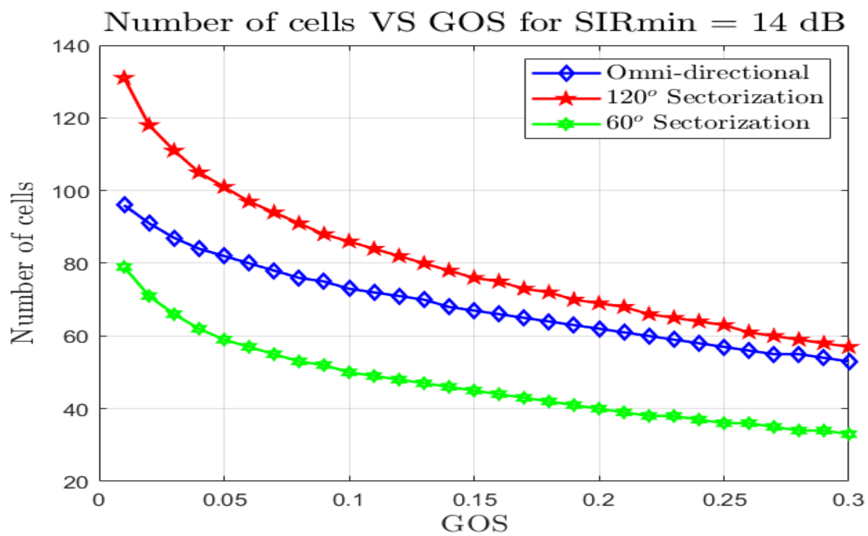
(ii) <u>Plot the **Traffic intensity per cell** versus **GOS** (1%to 30%).</u>



- As the GOS increases, it means higher probability for blocking a call which increases the traffic intensity per cell.
- As we increase the number of sectors, the capacity increases so, we allow more traffic having the same GOS.

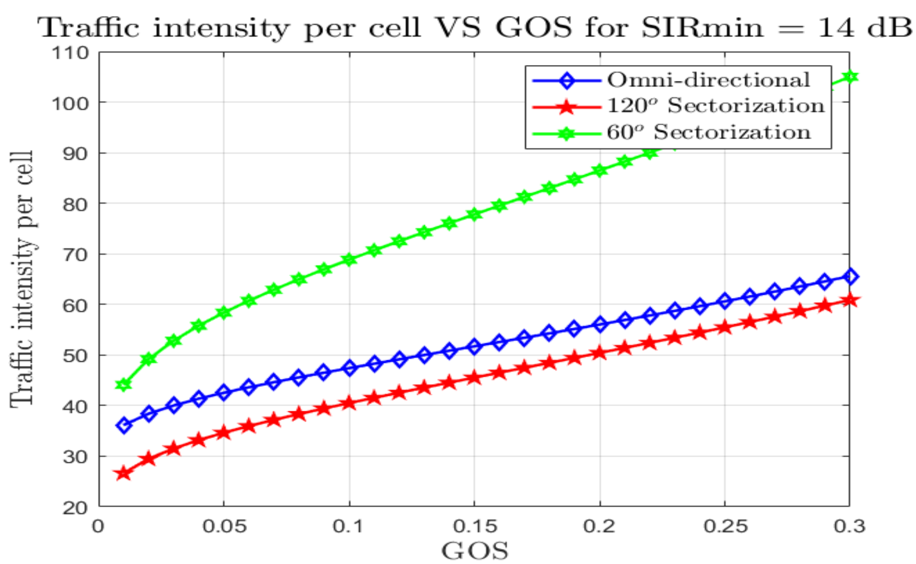
3. At SIR_{min}= 14 *dB* & User density= 1400 *users*/km^2

(i) Plot the **Number of cells** versus **GOS** (1%to 30%).



We notice that the number of cell decreases with the GOS with the same trend as in 2.(i) but we have a difference here which is, at the same GOS we have lower number of cells because by increasing the SIR_{min}, the cluster size increases for all sectorizations as shown in Figure 1 hence the number of cells increases.

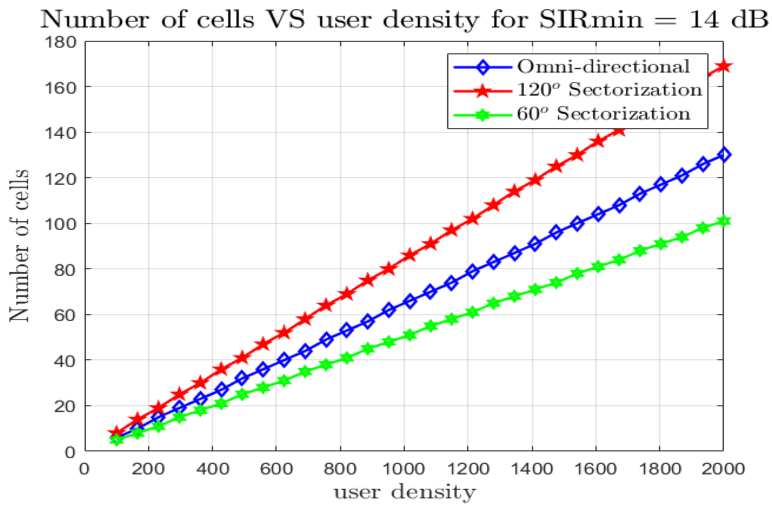
(ii) Plot the **Traffic intensity per cell** versus **GOS** (1%to 30%).



We notice here that the traffic intensity per cell increases with GOS with the same trend as in 2. (ii) but we have a difference which is, at the same GOS the traffic intensity increases as we decrease the SIR_{min} because this will subject the arriving call to more interference which increases the traffic intensity for all the sectorization methods.

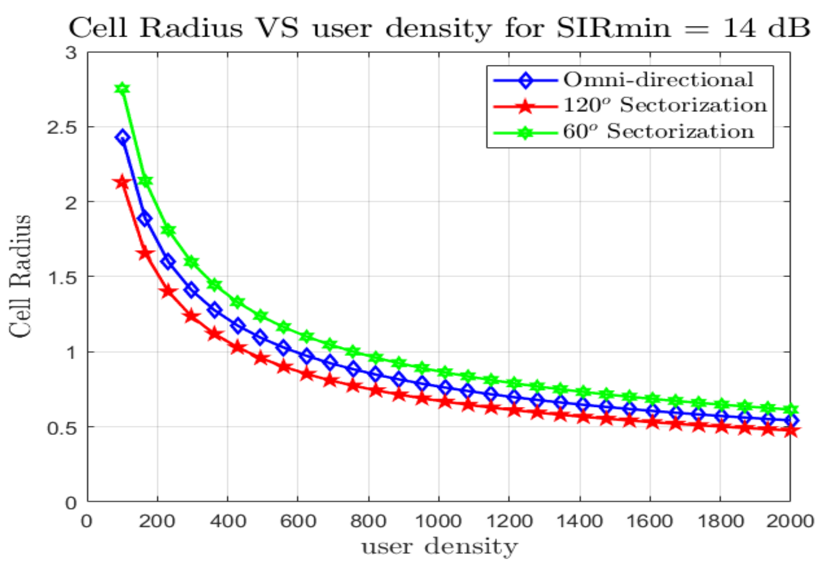
4. At $SIR_{min} = 14\ dB$ & GOS = 2%

(i) Plot the **Number of cells** versus **User density** (100 to 2000 $users/km^2$).



- We notice here that the number of cells increases with the user density because as the user density increases, the number of calls accessed to the same channel increases, so we increase the capacity by increasing the number of cells.
- As we increase the number of sectors, the capacity gets better at constant user density, hence the number of needed cells decreases.

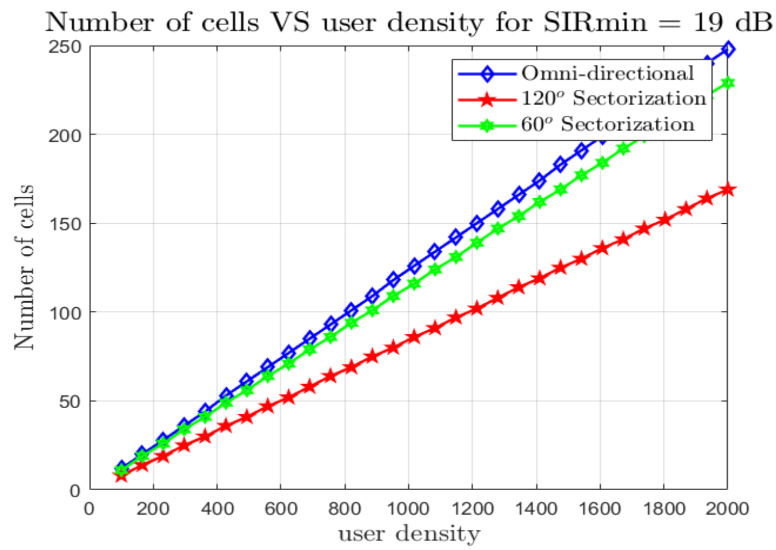
(ii) Plot the **Cell Radius** versus **User density** (100 to 2000 $users/km^2$).



- We notice that the cell radius decreases heavily with the user density because from the previous curve as we got that we need to increase the number of cells as the user density increases, and for the same area the cell radius must decrease.
- By increasing the number of sectors for the same user density, we get better capacity, so we need larger cells hence the cell radius increases.

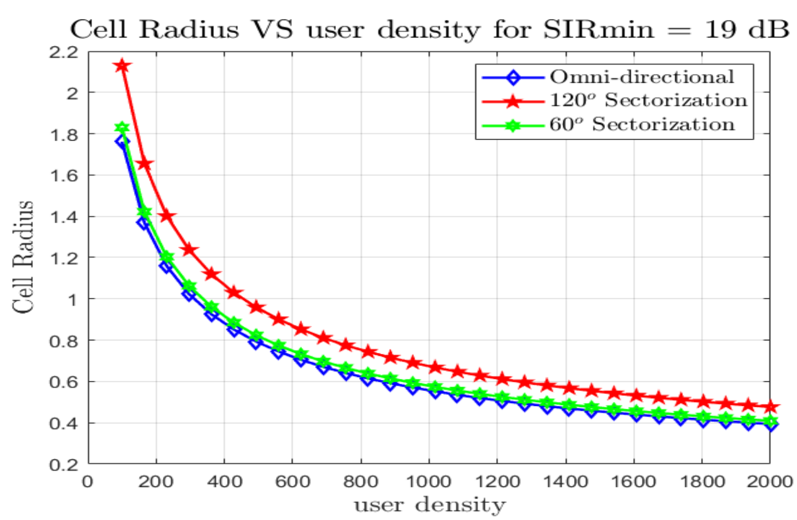
5. At SIR_{min}= 19 *dB* & GOS= 2%

(i) Plot the **Number of cells** versus **User density** (100 to 2000 *users*/km^2).



We notice that the number of cells increases with the user density with the same trend as in 4.(i) but there is a difference which is, at the same user density, the number of cells increases as we increase the SIR_{min} because as by increasing the SIR_{min}, we can add more cells without interference for all sectorization method.

(ii) Plot the **Cell Radius** versus **User density** (100 to 2000 *users*/km^2).



We notice that the cell radius decreases with the user density with the same trend as in 4.(ii) but there is a difference which is, at the same user density, the cell radius decreases by increasing the SIR_{min} since the number of cells increases as this case.