Pabna University Of Science and Technology

DEPARTMENT OF

INFORMATION AND COMMUNICATION ENGINEERING

FACULTY OF ENGINEERING AND TECHNOLOGY

Lab Report

COURSE TITLE: Cellular and Mobile Communication Sessional

COURSE CODE: ICE-4104

SUBMITTED BY:	**SUBMITED TO:**
Name: Md. Naim Hossain **Roll: 190616** **Session: 2018-19** **4 th Year 1st Semester** **Department of ICE** **Pabna University of Science and Technology** **Submission Date:**	**Dr. Md. Imran Hossain** **Associate Professor** **Department of ICE** **Pabna University of Science and Technology** Signature:

INDEX

Experiment No : 01

Name of the Experiment :

If a total of 33 MHz of bandwidth is allocated to a particular FDD cellular telephone system which uses two 25 kHz simplex channels to provide full duplex voice and control channels, compute the number of channels available per cell.

if a system uses (a) 4-cell reuse, (b) 7-cell reuse (c) 12-cell reuse. If 1 MHz of the allocated spectrum is dedicated to control channels, determine an equitable distribution of control channels and voice channels in each cell for each of the three system.

Mathematical solution :

Given:

Thtal bandwidth =33 MHz

Channel bandwidth = 25 k}{z x 2 simplex channels = 50 kHz/duplex channel

Thtal available channels = 33,000/50 = 660 channels

(a) For N= 4,

total number of channels available per cell = 660/4 165 channels.

(b)ForN=7,

total number of channels available per cell = 660/7 95 channels.

(c) For N = 12,

total number of channels available per cell = 660/12 t 55 channels.

A 1 MHz spectrum for control channels implies that there are 1000/50 = 20

control channels out of the 660 channels available. 'ft evenly distribute the

control and voice channels, simply allocate the same number of channels in

each cell wherever possible. Here, the 660 channels must be evenly distributed

to each cell within the cluster. In practice, only the 640 voice channels would be

allocated, since the control channels are allocated separately as 1 per cell. (a) For N = 4, we can have 5 control channels and 160 voice channels per cell.

In practice, however, each cell only needs a single control cha channels have a greater reuse distance than the voice channels). Thus, one

control channel and 160 voice channels would be assigned to each cell.

(b) For N = 7, 4 cells with 3 control channels and 92 voice channels, 2 cells

with 3 control channels and 90 voice channels, and 1 cell with 2 control channels and 92 voice channels could be allocated. In practice, however, each cell

would have one control channel, four cells-4vould have 91 voice channels, and

three cells would have 92 voice channels.

(c) For N = 12, we can have 8 cells with 2 control channels and 53 voice channels, and 4 cells with 1 control channel and 54 voice channels each. In an

actual system, each cell would have 1 control channel, 8 cells would have 53

voice channels, and 4 cells would have 54 voice channels.nnel (the controlems.

Python Code :

```
bw = 33000

schannel_bw = 25

print('Channel Bandwidth..')

dup_ch_bw = 2 * schannel_bw

t_ch = bw / dup_ch_bw

print(dup_ch_bw)

print('Total available channel')

print(t_ch)

cc_bw = 1000

t_cc = cc_bw / dup_ch_bw

print('Total control channel')

print(t_cc)

N = [4, 7, 12]

for i in range(3):

    ch = t_ch / N[i]

    ch_per_cell = round(ch)

    print('Channel per cell')

    print(N[i])

    print(ch_per_cell)


    c = t_cc / N[i]

    cc = round(c)

    v = (t_ch - t_cc) / N[i]

    vc = round(v)

    print('Control channel and voice channel are..')
```

```
print(cc)
print(vc)
```

Output :

```
Channel Bandwidth..
50
Total available channel
660.0
Total control channel
20.0
Channel per cell
4
165
Control channel and voice channel are..
5
160
Channel per cell
7
94
Control channel and voice channel are..
3
91
Channel per cell
12
55
Control channel and voice channel are..2
53
```

Experiment No : 02

Name of the Experiment :

If a signal to interference ratio of 15 dB is required for satisfactory forward channel performance of a cellular system, what is the frequency reuse factor and cluster size that should be used for maximum capacity if the path loss exponent is (a) it = 4 , (b) it = 3? Assume that there are 6 co-channels cells in the first tier, and all of them are at the same distance from the mobile. Use suitable approximations.

Mathematical Solution :

(a) n = 4

First, let us consider a N= 7-cell reuse pattern.

co-channel reuse ratio q=D/R = √3N =√3x7 = 4.583.

sipal-to-noise interference ratio is s/I=q^n / co-channel = (1/6)x(4.583)^4 = 75.3 = 18.G6 dB. Using equation (2.9), the

b) n = 3

First, let us consider a 7-cell reuse pattern.

Using equation (2.9), the sipal-to-interference ratio is given by

S/I = (l,6)x(4.583) = 16.04 = 12.05 dB.

N.

If N is 12, (I = j = 2).

q=D/R = 6.0.

S/I = (1/6) 6^3x = 36 = 15.56 dB.

Since this is greater than the minimum required S/I, N = 12 can be used.

Python Code :

```
import math

R_SI = 15

io = 6

n = [4, 3]

for a in range(2):

    N = 7

    Q = math.sqrt(3 * N)


    print('n: ', n[a])

    print('Frequency reuse factor: ', Q)

    SI = 10 * (math.log10((1 / io) * (Q**n[a])))

    print('Signal to interference ratio: ', SI)


    if SI < R_SI:

        i, j = 2, 2
```

```
N = i**2 + i * j + j**2
Q = math.sqrt(3 * N)


print('n: ', n[a])
print('Frequency reuse factor: ', Q)


SI1 = 10 * (math.log10((1 / io) * (Q**n[a])))
print('Signal to interference ratio: ', SI1)
```

Output :

```
n:  4
Frequency reuse factor:  4.58257569495584
Signal to interference ratio:  18.66287339084195
n:  3
Frequency reuse factor:  4.58257569495584
Signal to interference ratio:  12.051776917172353
n:  3
Frequency reuse factor:  6.0
Signal to interference ratio:  15.563025007672874
```

Experiment No : 03

Name of the Experiment :

How many users can be supported for 0.5% blocking probability for the following number of trunked channels in a blocked calls cleared system? (a) 1, (b) 5,

(c) 10, (d) 20, (e) 100. Assume each user generates 0.1 Erlangs of traffic.

Solution to Example 2tion A = UAU, we can obtain

the total number of users that can be supported in the system.

Mathematical Solution :

A = UAU, we can obtain

the total number of users that can be supported in the system.

(a) Giv1n C = I , = 0.1, GOS =re 2.6, we obtain A erefore, total number of users, U = A/AU = 0.005/0.1 = 0.05 users.

But, actually one user could be supported on one channel.

So, U = 1.

(b) Given C = 5, AU = 0.1, GOS = 0.005

From Figure 2.6, we obtain A = 1.13.

Therefore, total number of users, U = A/A=11= 1.13/

0.1 II users.

(c) Given C = 10, AU = 0.1 , GOS = 0.005

From Figure 2.6, we obtain A = 3.96.

Therefore, total number of users, U 1 =/AU = 3.9

6/0.! t39 users.

(d) Given C = 20, AU = 0.1 , GOS = 0.005

From Figure 2.6, we obtain A = 11.10.

Therefore, total number of users, U = A/AU = 11

.1/0.1 = 110 users.

(e)Given C 0.005

we obtain A = 80.9.

Therefore, total number of users, U = A/AU = 80.9/0.1 = 809 users.= 100, = 0.1 , GOS =

Python Code :

```
Gos = 0.5 / 100

Au = 0.1

A = [0.005, 1.13, 3.96, 11.1, 80.9]

c = [1, 5, 10, 20, 100]


print('Blocking probability : ',Gos)

print('Traffic intensity per user :  ',Au)

print('Traffic intensity : ',A)

print('Channel : ',c)


U = [a / Au for a in A]

u = [round(user) for user in U]

print('Number of users : ',u)
```

Output :

```
Blocking probability :  0.005
Traffic intensity per user :   0.1
Traffic intensity :  [0.005, 1.13, 3.96, 11.1, 80.9]
Channel :  [1, 5, 10, 20, 100]
Number of users :  [0, 11, 40, 111, 809]
```

Experiment No : 04

Name of the Experiment :

An urban area has a population of 2 million residents. Three conijieting trunked mobile networks (systems A, B, and C) provide cellular service in this area. System A has 394 cells with 19 channels each, system B has 98 cells with 57 channels each, and system C has 49 cells, each with 100 channels. Find the number of users that can be supported at 2% blocking if each user averages 2 calls per hour at an average call duration of 3 minutes. Assuming that all three trunked systems are operated at maximum capacity, compute the percentage market penetration of each cellular provider.

Mathematical Solution :

Probability of blocking = 2% = 0.02 = GOS number of average call λ =2 call duration 3 min = (3/60) =0.05 Traffic intensity per user Au = λH = 2 x 0.05 = 0.1 Erlangs Number of channels per cell used in the system, C from the Erlang B chart, the total carried traffic A = 12 Erlangs. when C= 19 traffic A = 45 Erlangs. when C= 57 traffic A = 88 Erlangs. when C= 100

System A Therefore, the number of users that can be supported per cell is U = A/Au = 12/0.1 = 120. Since there are 394 cells, the total number of' subscribers that can be supported by System A is equalto 120 x 394 = 47280.

System B traffic A = 45 Erlangs. Therefore, the number of users that can be supported per cell is U = A/Au = 45/0.1 = 450. Since there are 98 cells, the total number of subscribers that can be supported by System B is equal to 450 x 98 = 44100.

System C traffic A= 88 Erlangs. Therefore, the number of users that can be supported per cell is U = A/Au = 88/0.1 = 880. Since there are 49 cells, the total number of subscribers that can be supported by System C is equal to 880 x 49 = 43120

total number of cellular subscribers that can be supported by these three systems are 47280 + 44100 + 43120 = 134500 users.

Since there are 2 million residents in the given urban area and the total number of cellular subscribers in System A is equal to 47280, the percentage market penetration A is equal to = 47280/2000000 = 2.36% market penetration of System B is equal to = 44100/2000000 = 2.205% market penetration of System C is equal to = 43120/2000000 = 2.156% The market penetration of the three systems combined is equal to = 134500/2000000 = 6.725%

Python Code :

```
blocking_p = 2 / 100

lamda = 2

H = 3 / 60

Au = lamda * H

print('For system A')

c1 = 19

cell_A = 394

A1 = 12

Ua = A1 / Au

print('Number of users in System A = ',Ua)

subscriber_A = Ua * cell_A

print('Total number of subscribers in system A = ',subscriber_A)

pmp= (subscriber_A / 2000000) * 100

print('percentage_market_penetration_for_A = ',pmp)

print('\t')

print('For system B')

channel_b = 57

cell_B = 98

Ab = 45

Ub = Ab / Au

print('Number of users in System B = ',Ub)

subscriber_B = Ub * cell_B

print('Total number of subscribers in system B = ',subscriber_B)

pmpb = (subscriber_B / 2000000) * 100

print('percentage_market_penetration_for_B = ',pmpb)

print('\t')

print('For system C')

channel_c = 100

cell_C = 49
```

```
Ac = 88
Uc = Ac / Au
print('Number of users in System C = ',Uc)
subscriber_C = Uc * cell_C
print('Total number of subscribers in system C = ',subscriber_C)
pmpc = (subscriber_C / 2000000) * 100
print('percentage_market_penetration_for_C = ',pmpc)
print('\t')
Total_number_of_subscribers = subscriber_A + subscriber_B + subscriber_C
Market_penetration_for_three_system = (Total_number_of_subscribers / 2000000) * 100
print('Total number of subscribers:', Total_number_of_subscribers)
print('Market penetration for three systems:', Market_penetration_for_three_system)
```

Output :

```
For system A
Number of users in System A =  120.0
Total number of subscribers in system A =  47280.0
percentage_market_penetration_for_A =  2.3640000000000003

For system B
Number of users in System B =  450.0
Total number of subscribers in system B =  44100.0
percentage_market_penetration_for_B =  2.205

For system C
Number of users in System C =  880.0
Total number of subscribers in system C =  43120.0
percentage_market_penetration_for_C =  2.156

Total number of subscribers: 134500.0
Market penetration for three systems: 6.7250000000000005
```

Experiment No : 05

Name of the Experiment :

A certain city has an area of 1,300 square miles and is covered by a cellular system using a 7-cell reuse pattern. Each cell has a radius of 4 miles and the city is allocated 40 MHz of spectrum with a full duplex channe] bandwidth of 60 kHz. Assume a GOS of 2% for an Erlang B system is specified. If the offered traffic per user is 0.03 Erlangs, compute (a) the number of cells in the service

area, (b) the number of channels per cell, (c) traffic intensity of each cell, (d) the

maximum carried traffic; (e) the total number of users that can be served for

2% GOS, (f') the number of mobiles per channel, and (g) the theoretical maximum number of users that could be served at one time by the system.

Mathematical Solution :

(a) Given:

Thtal coverage area = 1300 miles

Cell radius = 4 miles

The area of a cell (hexagon) can be shown to be 2.5981R2, thus each cell covers

2.5981 x (4)2 = 41.57 sqmi.

Hence, the total number of cells are = 1300/41.57 = 31 cells.

(b) total available channel(allocated spectrum) = (channel BW x frequency resuse factor x number of channel per cell)

The total number of channels per cell (C)

= allocated spectrum / (channel width x frequency reuse factor)

= 40, 000,000/ (60,000 x 7) = 95 channels/cell

(c) Given:

C = 95, and GOS = 0.02

From the Erlang B chart, we have

traffic intensity per cell A = 84 Erlangs/cell

(d) Maximum carried traffic = number of cells x traffic intensity per cell

= 31 x 84 = 2604 Erlangs.

(e) Given traffic per user = 0.03 Erlangs

¶Ibtal number of users = Thtal traffic / traffic per user = 2604 / 0.02 = 86,800 users.

(1) Number of mobiles per channel = number of users/number of channels = 86,800 / 666 = 130 mobiles/ehannel.

(g) The theoretical maximum number of served mobiles is the number of available channels in the system (all channels occupied) = C x Nc = 95 x 31 = 2945 users, which is 3.4% of the customer base.

Python Code :

```
import math

area = 1300
radius = 4
each_cell_covers = math.floor(2.5981 * radius**2)

# (a)
print('(a)')
number_of_cells = math.floor(area / each_cell_covers)
print('Number of cells:', number_of_cells)

# (b)
allocated_spectrum = 40000
channel_width = 60
frequency_reuse_factor = 7
print('(b)')
number_of_channel_per_cell = math.floor(allocated_spectrum / (channel_width *
frequency_reuse_factor))
print('Number of channels per cell:', number_of_channel_per_cell)

# (c)
print('(c)')
traffic_intensity_per_cell = 84
print('traffic_intensity_per_cell =',traffic_intensity_per_cell)

# (d)
print('(d)')
maximum_carried_traffic = number_of_cells * traffic_intensity_per_cell
```

```
print('Maximum carried traffic:', maximum_carried_traffic)


# (e)

traffic_per_user = 0.03

print('(e)')

total_number_of_user = maximum_carried_traffic / traffic_per_user

print('Total number of users:', total_number_of_user)


# (f)

number_of_channels = number_of_channel_per_cell * frequency_reuse_factor

print('(f)')

number_of_mobile_per_channel = math.floor(total_number_of_user / number_of_channels)

print('Number of mobile users per channel:', number_of_mobile_per_channel)


# (g)

print('(g)')

theoretical_maximum_number_of_user_that_could_be_served = number_of_cells *
number_of_channel_per_cell

print('Theoretical maximum number of users that could be served:',
theoretical_maximum_number_of_user_that_could_be_served)
```

Output :

```
(a)
Number of cells: 31
(b)
Number of channels per cell: 95
(c)
traffic_intensity_per_cell = 84
(d)
Maximum carried traffic: 2604
(e)
Total number of users: 86800.0
(f)
Number of mobile users per channel: 130
(g)
Theoretical maximum number of users that could be served: 2945
```

Experiment No : 06

Name of the Experiment :

If a transmitter produces 50 watts of power, express the transmit power in units of (a) dBm, and (b) dBW. If 50 watts is applied to a unity gain antenna with a 900 MHz carrier frequency, find the received power in dBm at a free space distance of 100 m from the antenna, What is P (10 km) 2 Assume unity gain for the receiver antenna.

Mathematical Solution :

Transmitter power, = 50 W

Carrier frequency, = 900 MHz

Using equation (3.9),

(a) Transmitter power, = mW)1

= lDlogL5() x = 47.0 dBm.

(b) Transmitter power, = W)j = lolog[50J = 17.0 dBW.

The received power can be determined using equation (3.1).

PG G X 5Ø1J\! j\ = t t r = / —3 = 3.5 x 10 W = 3.5 tO mW

(4n) dL (4it) (I00)(J)

Pr(dRm) = = x mW) = —24.5 dBm.

The received power at 10 km can be expressed in terms of dBm using equation (3.9), where d0 = 100 m and d = 10km

Pr(IOkm) =—24.5dBni—4OdB

= — 64.5 dBm.

Python Code :

```
import math

# pt = Transmitted power, fc = carrier frequency in MHz
pt = 50
```

```
fc = 900
gt = 1
gr = 1
d = 100
do = 10 * 10**3

# (a)
print('(a)')
Transmitted_power_in_dBm = math.ceil(10 * math.log10(50 * 10**3))
print('Transmitted power in dBm:', Transmitted_power_in_dBm)

# (b)
print('(b)')
Transmitted_power_in_dbW = math.ceil(10 * math.log10(50))
print('Transmitted power in dBW:', Transmitted_power_in_dbW)

lamda = (3 * 10**8) / (fc * 10**6)
pr_mw = (pt * gt * gr * (lamda**2)) / (((4 * math.pi)**2) * (d**2) * 1) * 1000
received_power_in_dbm = 10 * math.log10(pr_mw)
print('received_power_in_dbm : ',received_power_in_dbm)
# Power received at 10 km
pr_10km = received_power_in_dbm + (20 * math.log10(d / do))

print('Received power at 10 km:', pr_10km)
```

<u>Output :</u>

```
(a)
Transmitted power in dBm: 47
(b)
Transmitted power in dBW: 17
received_power_in_dbm :  -24.536922331474983
Received power at 10 km: -64.53692233147498
```

Experiment No : 07

Name of the Experiment :

Determine the path loss of a 900MHz cellular system in a large city from a base station with the height of 100m and mobile station installed in a vehicle with antenna height of 2m. The distance between mobile and base station is 4Km.

Mathematical Solution :

antenna height = 2

base station antenna height = 100

frequency = 900

distance = 4

a_hre = 3.2 * (math.log10(11.75 * hre))**2 - 4.97

Path loss = 69.55 + 26.16 * 10log(fc) - 13.82 * 10log(hte) - a_hre + (44.9 - 6.55 * 10log(hte)) * 10log(d)

Path loss = 69.55 + 26.16 * 10log(900) - 13.82 * 10log(100) – a_hre + (44.9 - 6.55 * 10log(100)) * 10log(4)

Python Code :

```
import math

hre = 2

hte = 100

fc = 900

d = 4


a_hre = 3.2 * (math.log10(11.75 * hre))**2 - 4.97

Lp = 69.55 + 26.16 * math.log10(fc) - 13.82 * math.log10(hte) - a_hre + (44.9 - 6.55 * math.log10(hte)) * math.log10(d)


print('Loss path : ',Lp)
```

Output :

Loss path : 137.29304510570788

Experiment No : 08

Name of the Experiment :

Determine the path loss between base station (BS) and mobile station (MS) of a 1.8GHz PCS system operating in a high-rise urban area. The MS is located in a perpendicular street to the location of the BS. The distances of the BS and MS to the corner of the street are 20 and 30 meters, respectively. The base station height is 20m.

Mathematical Solution :

Given value

Frequency(fc) =1.8 GHz

Base station height(hb) =20 m
Mobile station height(hm) = 1.5

Distance = $\sqrt{20^2 + 30^2}$

Correlation factor (a_hm) = $(1.1 \times \log(fc) - 0.7) \times hm - (1.56 \times \log(fc) - 0.8)$

$= -0.227$

Pathloss : = $(46.3 + 33.9 \times \log(fc) - 13.28 * \times \log(hb) - a\_hm +$

$(44.9 - 6.55 \times \log(hb)) \times \log(d))$

$= 94.54$ dB

Python Code :

```
import math


def calculate_path_loss(fc, hb, hm, d):
    a_hm = (1.1 * math.log10(fc) - 0.7) * hm - (1.56 * math.log10(fc) - 0.8) #corelation factor
    Lp = (46.3 + 33.9 * math.log10(fc) - 13.28 * math.log10(hb) - a_hm +
          (44.9 - 6.55 * math.log10(hb)) * math.log10(d))


    return Lp
```

```
fc = 1.8
hb = 20
hm = 1.5  #dhore neoya hoise 1-10
d = math.sqrt(20**2 + 30**2)  # Distance
path_loss = calculate_path_loss(fc, hb, hm, d)
print(f"The estimated path loss is: {path_loss:.2f} dB")
```

Output :

```
The estimated path loss is: 94.54 dB
```

Experiment No : 09

Name of the Experiment :

A mobile is located 5 km away from a base station and uses a vertical 2. 4 monopole antenna with a gain of 2.55 dB to receive cellular3radio signals. The E-field at 1 km from the transmitter is measured to be V/rn. The carrier frequency used for this system is 900 MHz. (a) Find the length and the gain of the receiving antenna. (b) Find the received power at the mobile using the 2-ray ground reflection model assuming the height of the transmitting antenna is 50 m and the receiving antenna is 1.5 m above ground.

Mathematical Solution :

Given: T-R separation distance = 5 km E-field at a distance of 1 km = I VIm Frequency of operation, f = 900 MHz c A = - = = I Length of the antenna, L, = X/4 = 0.333/4 = 0.0833 m = 8.33 cm. Gain of A /4 monopole antenna can be obtained using equation (3.2). Gain of antenna =1.8 = 2.55 dB. (b) Since the electric field is given by 2Eodo2lthrhr k d Ad 2* xl x 2ir(50) (Ii) = 113.1* The received power at a distanced can be obtained using equation (315) I P (d) = 113.1 *10) [1.8 (0.333)2 r 377 L P(d = 5 km) = 5.4 W = —122.68 dBW or —9268 dBm.

Python Code :

```
import math


T_R_field = 5
E_field = 10**(-3)
f = 900
do = 1000
lamda = (3 * 10**8) / (900 * 10**6)
```

```
# (a)
print('(a)')
length_of_antenna = lamda / 4
print('Length of antenna:', length_of_antenna,'m')

gain = 10**(2.55 / 10)
print('gain : ',gain)
# (b)
print('(b)')
ht = 50
hr = 1.5
d = 5 * 10**3

Er_d = (2 * E_field * do * 2 * math.pi * ht * hr) / (lamda * d**2)
Ae = (gain * lamda**2) / (4 * math.pi)
pr_d = (Er_d**2) / (120 * math.pi) * Ae
received_power_at_5km_distance = 10 * math.log10(pr_d)
print('Er(d) : ',Er_d)
print('Pr(d) : ',pr_d)
print('Received power at ( d=5 km ) distance:', received_power_at_5km_distance,'dBW')
```

Output :

```
(a)
Length of antenna: 0.08333333333333333 m
gain :  1.7988709151287878
(b)
Er(d) :  0.00011309733552923255
Pr(d) :  5.396612745386364e-13
Received power at ( d=5 km ) distance: -122.67878745280339 dBW
```

Experiment No : 10

Name of the Experiment :

A hexagonal cell within a 4-cell system has a radius of 1.387 km. A total of 60 channels are used within the entire system. If the load per user is 0.029 Erlangs, and X = callihour, compute the following for an Erlang C system that has a 5% probability of a delayed call: (a) How many users per square kilometer will this system support? (a) What is the probability that a delayed call will have to wait for more than lOs? (c) What is the probability that a call will be delayed for more than 10 seconds?

Mathematical Solution :

Given, Cell radius, R = 1.387 km Area covered per cell is 2.598 x (1.387)^2 = 5 sq kin Number of cells per cluster = 4 Total number of channels = 60 number of channels per cell = 60 / 4 = 15 channels.

(a) From Erlang C chart, for 5% probability of delay with C = 15, traffic intensity = 9.0 Erlangs. number of users = total traffic intensity / traffic per user =9.0/0.029 = 310 users = 310 users/5 sq km = 62 users/sq km

(b) Given λ = 1 , holding time H = Au/λ = 0.029 hour = 104.4 seconds. The probability that a delayed call will have to wait for more than 10 s is Pr[delay] = exp(.—(C-A)t/H) = exp(—(15—9.0)10/l04.4) = 56.29 % (c) Given Pr[delay >0] = 5% = 0.05 Probability that a call is delayed more than 10 seconds, Pr[delay> tO] =Pr[delay> 0]Pr[delay> t|delay] = 0.05 x 0.5629 = 2.81 %

Python Code :

```
import math


# R = cell radius, N = total number of channels, n = number of cells per cluster

R = 1.387

n = 4

N = 60

# A = Area covered per cell

A = 2.598 * R**2

# c = number of channels per cell

c = N / 4
```

```
# (a)
print('(a)')
traffic_intensity = 9
load_per_user = 0.029
number_of_users = math.floor((traffic_intensity / load_per_user) / A)
print('Number of users:', number_of_users)

# (b)
print('(b)')
lamda = 1
t = 10
H = (load_per_user / lamda) * 3600
the_probability_to_wait = math.exp(-(c - traffic_intensity) * t / H) * 100
print('The probability to wait:', the_probability_to_wait)

# (c)
print('(c)')
p = 5 / 100
probability_of_delay_more_than_10sec = p * the_probability_to_wait
print('Probability of delay more than 10 seconds:', probability_of_delay_more_than_10sec)
```

Output :

```
(a)
Number of users: 62
(b)
The probability to wait: 56.286658884286155
(c)
Probability of delay more than 10 seconds: 2.814332944214308
```