Wireless System Performance - Miniproject

NDS 820

June 6, 2019

Chapter 1

Troels

1.1 MM6 - Cellular concept

A mobile telephone system with 100 channels uses a modulation scheme requiring a minimum S/I of at least 20 dB to achieve acceptable downlink performance. A symmetric hexagonal cell plan is used with the base stations located at the centre of hexagons (omni-directional antennas). All base stations are assumed to use the same transmitter power. At most how many channels per cell, k, can be offered by the system if the propagation loss increases with the fourth power of the distance? How many channels if the propagation loss increases with the third power of the distance?
Hint: For simplicity, assume there are 6 co-channel cells in the first tier and all of them are at the same distance from the mobile; evaluate both the simple and the more accurate worst-case expression for the first tier interference (inline Matlab function available from course Moodle).

To find out how many channels are needed per cell using the given variables:

- Channels: 100
- Loss power coefficient: 4

The signal to interference ratio can be calculated by:

$$\frac{S}{I} = \frac{R_c^{-n}}{\sum_{k=1}^{K} D_k^{-n}} =\approx \frac{Q^n}{K} \tag{1.1}$$

where n is the path loss exponent. K is the number of co-channel cells. Q is Co-channel reuse ratio.

Assuming 6 co-channels per cell, 20 dB S/I and path loss of power 4. Then we have:

$$20dB = \frac{Q^4}{6} \Rightarrow \tag{1.2}$$

$$10^{\frac{20}{10}} = \frac{Q^4}{6} \Rightarrow \tag{1.3}$$

$$Q = \sqrt[4]{600} = 4.94 \tag{1.4}$$

Using $\sqrt{3 \cdot N} > Q$ as a requirement, we can check the minimum amount of cells needed:

$$N > \left(\frac{Q}{\sqrt{3}}\right)^2 \Rightarrow \tag{1.5}$$

$$N > \frac{Q^2}{3} \Rightarrow \tag{1.6}$$

$$N = \frac{4.94^2}{3} = 8.15 \tag{1.7}$$

In this case, the minimum amount of cells is 8.15, but since we can't have part of a cell, the minimum amount of cells is 9. Check: $Q < \sqrt{3 \cdot 9} \Rightarrow \sqrt{3 \cdot 9} = 5.1962$ which is bigger than $Q = 4.94$ which means that 9 cells is good enough.

For figuring out how many channels there are per cell we calculate $\frac{channels}{cells}$ which in this case is 100 channels and 9 cells: $k = \frac{100}{9} = 11.11$. Since it is impossible to use more than 100 channels, we need round down instead which results in channels to spare. In this we get the amount of channels per cell to be 11.

Matlab code:

```
ch = 100; %Channels
n = 4; %Fourth power

syms N positive
N_r = ceil(double(solve(100 == (sqrt(3*N).^n/6), N)));
channel_per_cell = floor(ch / N_r);
```

Where N_r is the minimum number of cells that should be used. In this case, using the matlab script, we get the "channel_per_cell" to be 11.

For the third power:

$$20dB = \frac{Q^3}{6} \Rightarrow \tag{1.8}$$

$$10^{\frac{20}{10}} = \frac{Q^3}{6} \Rightarrow \tag{1.9}$$

$$Q = \sqrt[3]{600} = 8.43 \tag{1.10}$$

We do the check:

$$N > \left(\frac{Q}{\sqrt{3}}\right)^2 \Rightarrow \tag{1.11}$$

$$N > \frac{Q^2}{3} \Rightarrow \tag{1.12}$$

$$N = \frac{8.43^2}{3} = 23.68 \tag{1.13}$$

Again, we can't use part of a cell, so we need to increase it, by table look-up the closest amount of cells that can be used is 28. We can now calculate the amount of channels per cell for the third power:

$$\frac{channels}{cells} \Rightarrow \tag{1.14}$$

$$k = \frac{100}{28} = 3.57 \approx 3 \tag{1.15}$$

In this case we again have to round down.

1.2 MM7 - Cellular principle and channel allocation

1.2.1 A

Consider a cellular system in which calls have exponential holding time (duration) with an average of 2 minutes, and the probability of blocking is to be no more than 1% (block calls cleared system). Assume than every subscriber makes 2 calls per hour on average with Poission arrival. The total number of channels available is 100 and the required worst-case S/I is 20 dB. Assume that the propagation path loss exponent is n = 3.76.

- Find the number of users that can be supported in each cell with omni directional antennas; use both the simple and the more accurate formula for worst-case S/I calculation, assuming first-tier interference only (a Matlab inline function is available from the course material in Moodle for the latter)
- Compare the result to the case with 120 degrees sectoring, using the simple formula.

Hint: Use the Erlang B-curves included, and the relation between offered traffic intensity per user (in units of Erlang), Au , the mean call arrival rate, λ , and average call holding time, H.

$$PrBlocking = 0.01 \tag{1.16}$$
$$H = Holdingtime = 2minutes \tag{1.17}$$
$$callsPrHour = 2 \tag{1.18}$$
$$channels = 100 \tag{1.19}$$
$$SIdB = 20 \tag{1.20}$$
$$Pathloss, n = 3.76 \tag{1.21}$$

Using what we have, we find out the usage of each user:

$$A_u = H \cdot \lambda = 2 \cdot \frac{2}{60} = \frac{1}{15} = 0.067 \tag{1.22}$$

$$\frac{S}{I} = \frac{Q^n}{6} = \frac{Q^{3.76}}{6} \tag{1.23}$$

$$10^{\frac{20}{10}} = \frac{Q^{3.76}}{6} \tag{1.24}$$

$$Q = \sqrt[3.76]{600} = 5.48 \tag{1.25}$$

$$Nr > \left(\frac{Q}{\sqrt{3}}\right)^2 = \left(\frac{5.48}{\sqrt{3}}\right)^2 \Rightarrow \tag{1.26}$$

$$Nr > 10.01 \Rightarrow Nr = 12 \tag{1.27}$$

Now we find how many channels per cells there are:

$$k = \frac{channels}{cells} \Rightarrow \tag{1.28}$$

$$k = \frac{100}{12} = 8.33 \approx 8 \tag{1.29}$$

Looking on the figure (fig. 1.1) we can find out how many users there are...

$$A = 3.1 \tag{1.30}$$

$$\frac{A}{A_u} = \frac{3.1}{0.067} = 46,26 \approx 46users \tag{1.31}$$

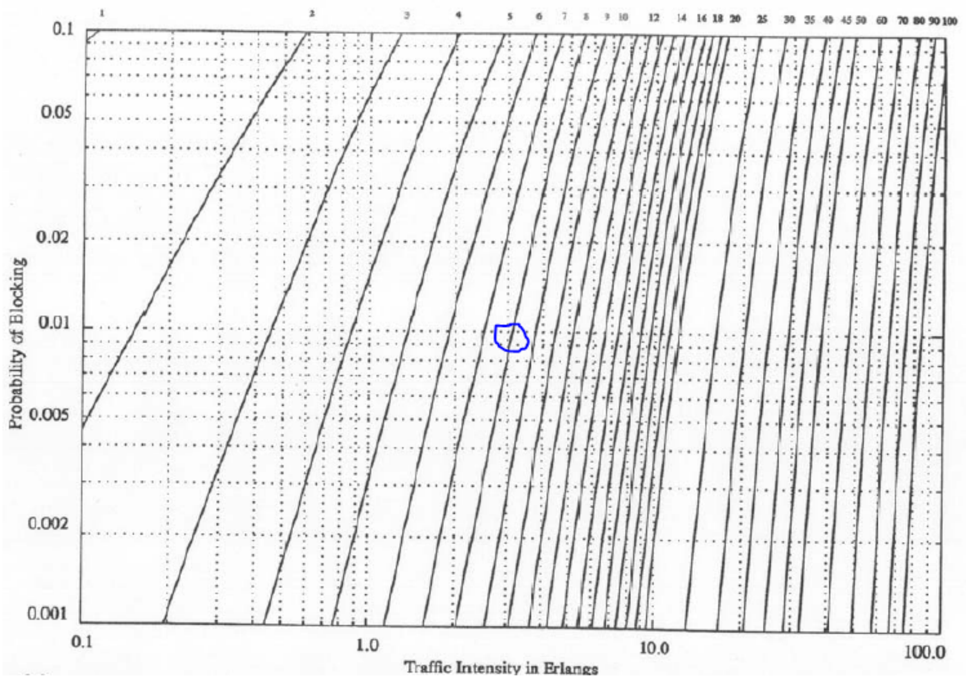


Figure Erlang B loss system – probability of blocking as a function of the total offered traffic intensity in Erlangs; the number of channels is given as a curve parameter.

Figure 1.1

Now if we sector the cells using a 120° plan, the number of interfering cells within the first tier will decrease. For N=7, we know that only two cells from the first tier will interfere (see course). So let's compute the required Nr using the simple formula with only two interfering cells and let's see if 7 fits or not and if it's the case, whether we can use a lower N or not:

$$\frac{S}{I} = \frac{Q^n}{6} = \frac{Q^{3.76}}{2} \tag{1.32}$$

$$10^{\frac{20}{10}} = \frac{Q^{3.76}}{2} \tag{1.33}$$

$$Q = \sqrt[3.76]{200} = 4.09 \tag{1.34}$$

$$Nr > \left(\frac{Q}{\sqrt{3}}\right)^2 = \left(\frac{4.09}{\sqrt{3}}\right)^2 \Rightarrow \tag{1.35}$$

$$Nr > 5.55 \Rightarrow Nr = 7 \tag{1.36}$$

So N=7 is a perfect fit : we can't use a lower N with only two interfering cells. Maybe with a lower number of cells in a cluster, only one cell of the first tier would interfere. So let's compute the minimum N needed for only one interfering cell. After doing exactly the same calculations, we can see that in this case, N must be equal to 4 at minimum. So we have to see how many cells in the first tier interfere in the case N=4:

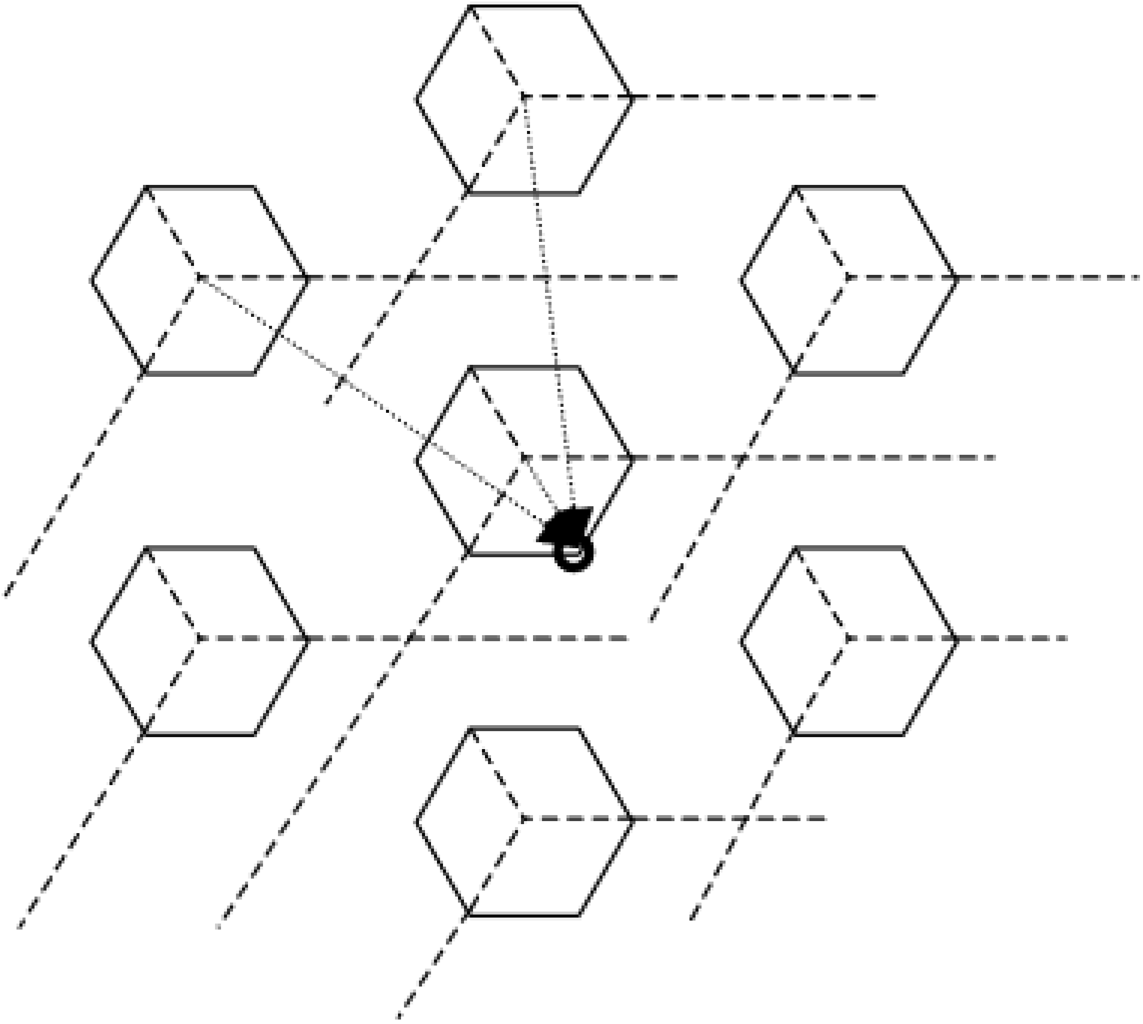


Figure 1.2: Worst case scenario for N=4 and 120° sectoring

We can see on the figure that two cells interfere in the first tier. As such, we cannot use N=4.

Let's do as before to find the number of channels in a cell:

$$k = \frac{channels}{cells} \Rightarrow \tag{1.37}$$

$$k = \frac{100}{7} = 14.29 \approx 14 \tag{1.38}$$

And now we can again look at the picture and see the numbers of users in a cell:

$$A = 7.5 \tag{1.39}$$

$$\frac{A}{A_u} = \frac{7.5}{0.067} = 46,26 \approx 111 users \tag{1.40}$$

So we can see that by using sectoring, we can put way more users in a cell. It brings some problems: first, it increases complexity of the system by a lot. And more importantly, it is important to keep in mind that by dividing the cell, which has a certain number of channels with a certain number of user supported, in 3 sectors with each one supporting 1/3 of the channels, then you will not have the same number of users that are supported by the network, like we assumed here, so the increase in the reuse factor will not be as big as the increase in system capacity. Still, sectoring helps a lot with system capacity in this case.

1.2.2 B

A terminal moves with a speed 70 km/h through a cellular network. Make a rough estimate on the average number of handovers per call if the average call duration is 1 minute and the cell radius is 1 km and 50 m, respectively. Assume that a handover is made each time a terminal crosses a cell boundary.

To find out how many handovers there are, we need to compute the distance travelled first, then we can calculate the handover over different cell radius:

$$Speed = 70\frac{km}{h} \tag{1.41}$$

$$CellDuration = 1minute \tag{1.42}$$

$$CellRadius1 = 1km \tag{1.43}$$

$$CellRadius2 = 50m \tag{1.44}$$

$$distanceOneMin = 70\frac{km}{h} \cdot 1min = 1.16667km = 1166.67m \tag{1.45}$$

$$Handover1kmMean = \frac{1166.67}{2 \cdot 1000} = 0.5833 \tag{1.46}$$

$$Handover5mmMean = \frac{1166.67}{2 \cdot 50} = 11.6667 \tag{1.47}$$

From the above it can be concluded that the *terminal* is moving 1166.67m over 1 minute
When we have a cell radius of 1km, the mean handover to be found is 0.5833 over 1 minute.
When we have a cell radius of 50m, the mean handover to be found is 11.6667 over 1 minute.

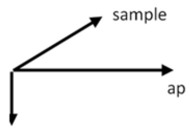
1.2.3 C

c) Compute the downlink interference compatibility matrix *I* for an in-building system comprising 6 access points (cells) distributed over 3 floors. The computation is to be based on calculation of the S/I ratio and a (mutual) “exclusion rule” which states

$$I_{ij} = \begin{cases} 1, \text{ if } \left(S/_I\right)_{5\%-tile} < 5\text{ dB} \\ 0, \text{ otherwise} \end{cases}, \quad I = \begin{bmatrix} & \vdots & \\ \cdots & I_{ij} & \cdots \\ & \vdots & \end{bmatrix}$$

The interference conditions for the in-building system is characterised by the path loss matrix PL_val (Matlab .mat file which can be downloaded from Moodle):

PL_val (dimension is $1 \times$ no. access points $\times$ no. of samples): The estimated path loss in dB for respective access points and sample positions over the three floors of the building (samples can be assumed to be uniformly distributed over the floor space). The path loss is defined between antenna ports.



Note: The matrix can be squeezed to a two dimensional matrix by using Matlab’s *squeeze* function.

The downlink transmit power is the same in all cells and it can be assumed that users always connect to the strongest cell.

Based on the computations, what can you say in general about the mutual interference between a couple of cells?

For this exercise we use Matlab since the data provided can be read there. We have the following to work with:

$$Floors = 3 \tag{1.48}$$
$$AccessPoints = 6 \tag{1.49}$$
$$Locations = 311 \tag{1.50}$$
$$minDBDiff = 5 \tag{1.51}$$

For the assignment, the power of each access point is assumed to be the same. The pathloss is estimated for each access point and based on that, it can be seen if the access points interfere with each other. To know if the access points interfere with each other, the pathloss between them should be less than 5dB for at least 5 percent of cases (from the equation above).

First, we load our data (we use the hint from above):

```
load('Access-Point-PL.mat');     %Load
PL_val = squeeze(PL_val);        %Remove extra stuff
```

Now we select the lowest values for each access point at every position. This will give us an idea of what access points are used at all positions:

```
[MinPLVal,MinPLAP] = min(PL_val); % Split up for min per column
```

With the data ordered, we can construct our interference matrix and check what the difference is between the different access points according to the predefined value of 5dB. With all the data stored in the interference matrix, we can see how many times each access point is visited and how much each access point interferes with the strongest access point.

```
iMatrix = zeros(size(PL_val,1)); % 6x6
```

```
for i = 1:size(PL_val,2)        % 311
    for j = 1:size(PL_val,1)    % 6
        difference = (PL_val(j,i)-MinPLVal(i)) <= minDBDiff; % Calc difference 1/0
        iMatrix(MinPLAP(i),j) =  iMatrix(MinPLAP(i),j) + difference; % Add to matrix
    end
end
```

The code above goes through several steps. First, the pathloss value is selected from PL_val, then an access point is selected. The pathloss valus is then compared to the value stored in the array stored earlier for the lowest pathloss value. If the difference is lower or equal to minDBDiff, 5dB, then that value is added to the interference matrix. The result looks as follows:

```
iMatrix =
    57     1     6     0     0     0
     6    54     0     3     0     0
     4     0    50     2     4     0
     0     8     2    48     0     1
     0     0     0     0    57     5
     0     0     1     1     5    45
```

Where the diagonal is the total pathloss columns from PL_var (311) spread out over all the strongest access points. Everything else in the column is the number of times that access point interferes with the strongest access point. To find out how much the interference's make up of the total to find the 95th percentile, the values are converted to percentages. We then check if those percentages are above the threshold of 5 percent. If it is, the access point is interfering too much, above 5 percent:

```
iMatrix = iMatrix./diag(iMatrix)' % Convert to percent
iMatrix =
    1.0000    0.0185    0.1200         0         0         0
    0.1053    1.0000         0    0.0625         0         0
    0.0702         0    1.0000    0.0417    0.0702         0
         0    0.1481    0.0400    1.0000         0    0.0222
         0         0         0         0    1.0000    0.1111
         0         0    0.0200    0.0208    0.0877    1.0000

iMatrix = iMatrix >= 0.05 % Check if the percent is over threshold of 5
```

The result looks as follows:

```
iMatrix =
    1   0   1   0   0   0
    1   1   0   1   0   0
    1   0   1   0   1   0
    0   1   0   1   0   0
    0   0   0   0   1   1
    0   0   0   0   1   1
```

Comparing the matrices we can see that some access points did manage to interfere but since they stayed below the threshold, they were discarded in the interference matrix. From the interference matrix it can be seen that the interference between the access points isn't symmetric. This is odd because you would expect that if something interferes one way, it would also interfere the other way in a symmetric fashion.

1.3 MM9 - Link adaptation

1.3.1 A

To conserve battery power in the mobile terminal, the user terminals in a wireless cellular system control the transmit power to achieve a constant received power at the base station. With this scheme, estimate the reduction in average transmitter power in the mobile terminal, compared to a system in which the mobiles transmit with constant power throughput the cell. Assume that the user terminals are uniformly distributed over circular cells of radius R, and that the propagation loss is proportional to the distance raised to the forth power. Hint: Figure out about the probability of a user being located at an arbitrary position within the cell, and using that, derive the average transmit power level for the two cases. Note that since only the reduction is asked for, it is sufficient to know the power in each case up to a proportionality factor.

In order to find the power reduction we will need to calculate the expected power for the fixed power and variable power transmissions.

What we need to find is:

$$E[x] = \int_A x \cdot f_x(x)dx \tag{1.52}$$

where $f_x(x)$ is the pdf of the transmitted power. This pdf can be expressed as:

$$f_x(x) = \frac{1}{\pi \cdot R^2} \tag{1.53}$$

From our perspective, we look at the distance without the angle, but to calculate it you need the uniform distributed angle and distance which is easier in the Im Polar plane.

`Constant (same power):`

$$P_t(R) = \int_0^R \int_0^\pi K \cdot R^4 \frac{1}{\pi \cdot r^2} \cdot r \;\; d\theta dr \Rightarrow K \cdot R^4 \tag{1.54}$$

`Variable (power control):`

$$P_t(r) = \int_0^R \int_0^{2\pi} K \cdot r^4 \frac{1}{\pi \cdot r^2} \cdot r \;\; d\theta dr \Rightarrow \frac{KR^4}{3} \tag{1.55}$$

And the factor between them:

$$\frac{P_t(R)}{P_t(r)} = \frac{K \cdot R^4}{\frac{K \cdot R^4}{3}} = 3 \tag{1.56}$$

3 times more power when power control is not present.

1.3.2 C

Consider the WCDMA HSDPA example from the slide set. HSDPA uses the CMDA principle with a fixed spreading factor of 16 (processing gain), and a chip rate of 3.84 Mcps (chips per s, or just bits per s). The spread signal occupies a channel bandwidth of approx. 5 MHz. Assume that the block error probability (BLEP) curves for different modulation and coding schemes (MCS) can be approximated by the following expression (including link level processing with HSDPA turbo coding, rate matching, and interleaving).

$$BLEP = 1 - \frac{1}{\pi}(\frac{\pi}{2} + t_g^{-1}(k(SINR + o - 10 \cdot log_{10}(n_c)))) \tag{1.57}$$

where 'o' is a modulation and coding scheme dependent offset (in dB) and 'n_c' is the number of orthogonal channels (codes) used for transmission (1 to 5 possible in this example). The parameter k is a constant gain multiplier, which can be assumed equal to 10, and 't_g^{-1}' is the inverse tangent function. The signal to interference plus noise ratio (SINR) is measured in dB in this expression!

1. Plot the ideal link adaptation curve assuming that the offset takes the following values:

First, verify the behaviour of the curves by plotting the different combinations,e.g. BLEP vs. SINR.

For each of the combinations you can calculate the maximum rate according to the formula shown in the slide set (maximum rate as a function of modulation order M, code rate r_c and number of orthogonal channels, or codes in WCDMA, n_c).

Let's first plot all the BLEP vs SINR curves:

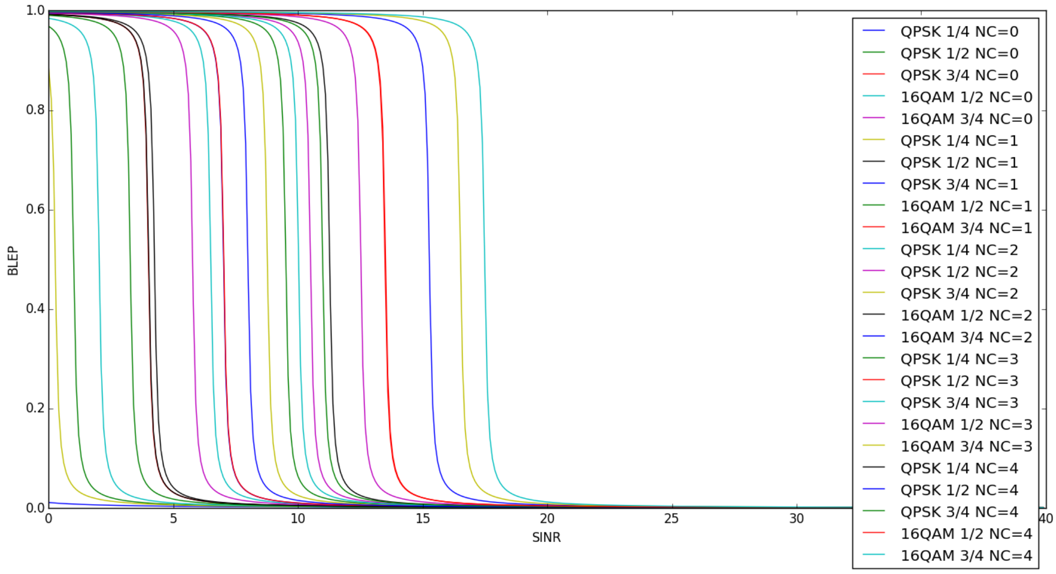


Figure 1.10: figure 1

Thanks to the BLEP values we have gathered, we can know calculate the averages emitting rates at a all SINR for all possibilities, thanks to this intuitive formula :

$AvgRate(SINR) = MaxRate * (1 - BLEP(SINR))$

Let's plot those curves for all combinations of parameters:

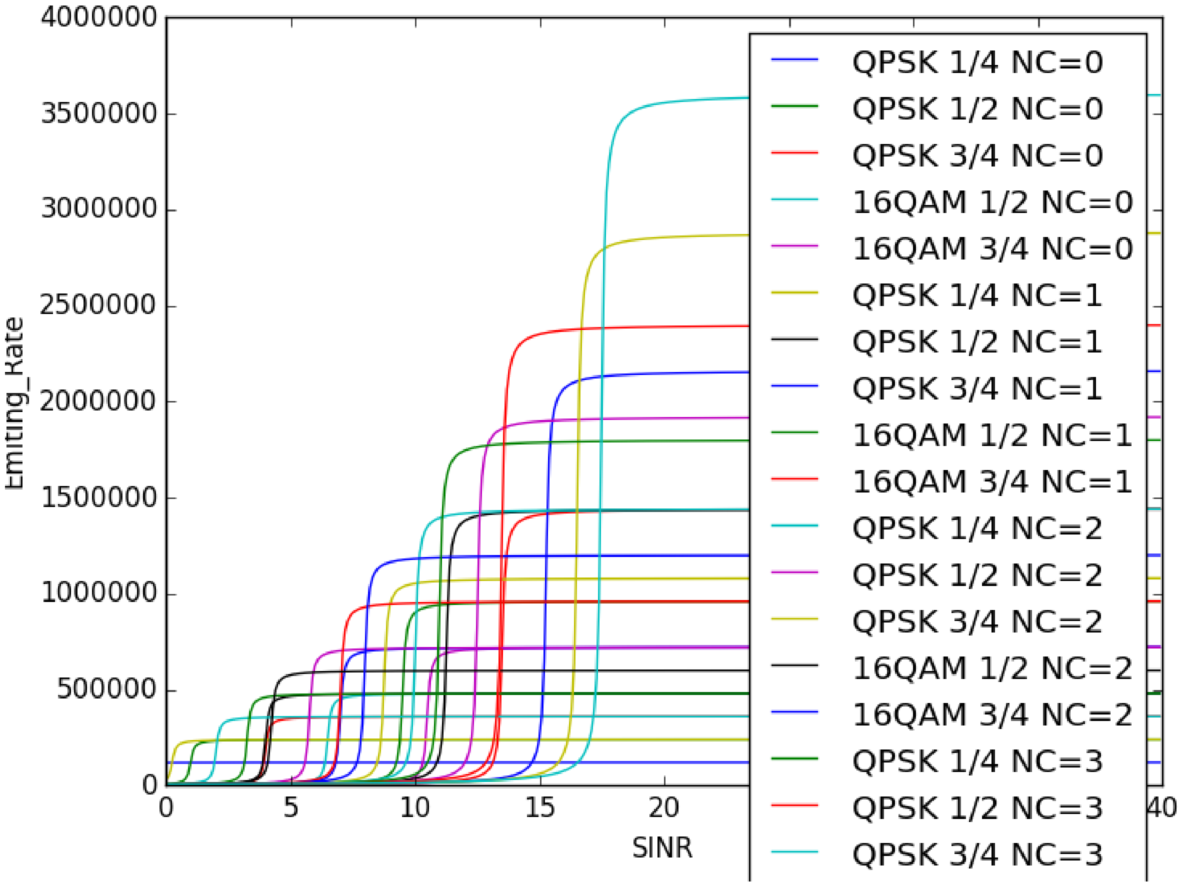


Figure 1.11: figure 2

2. How many of the possible MCS and code combinations are actually used for the ideal link adaptation curve?

In order to plot the ideal link adaptation curve, we only need the combinations that are the best at certain SINR. If a combination is worse than another one for each SINR, then it is never going to be used and as such it will not be used for our ideal link adaptation model.

We can see that there is only 14 such curves on our plot. Thus, we only use 14 combinations for the ideal link adaptation curve.

3. On the same plot, plot the channel capacity assuming the AWGN channel (Shannon bound); how "close" is the ideal adaptation to the Shannon bound?

Hint: Remember that Shannon states the capacity for a channel of given bandwidth and signal to (Gaussian) noise ratio in the channel.

In the particular case of AWGN channel we are considering, Shannon-Hartley theorem says that

$Capacity = Bandwith * (1 + SINR)$

We can plot that along the other link adaptation curves:

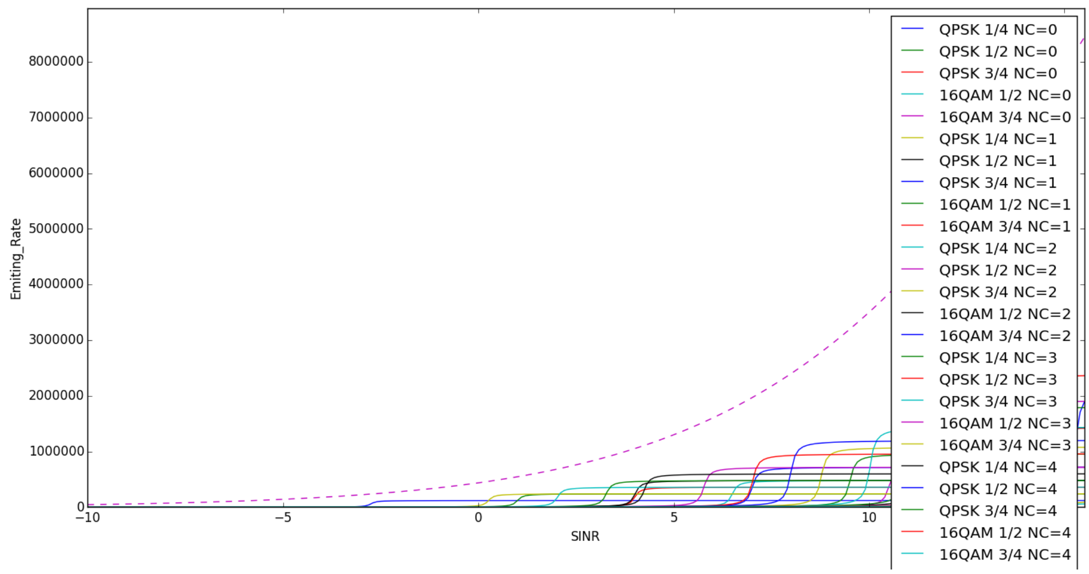


Figure 1.12: figure 3

We have to be cautious and take the real SNR in the channel as a parameter for the Shannon forumla. Since the SINR as seen by the transmitter effectively 16 times better after using the WCDMA modulation, we have to be cautious and translate it back to compute the actual Shannon bound in our case. At high SINR, difference between the best we can come up and the theoretical best tend to infinity, since we are limited in our MCS order : at best, we have access to 16QAM. Using 64 QAM or 128 QAM for example would help reduce the gap for a while at high SINR, but as long as we are limited in the order, however big it may be, difference will tend to infinity anyway.

Here is the curve of the difference between the best rate we can produce and the maximum theoretical rate:

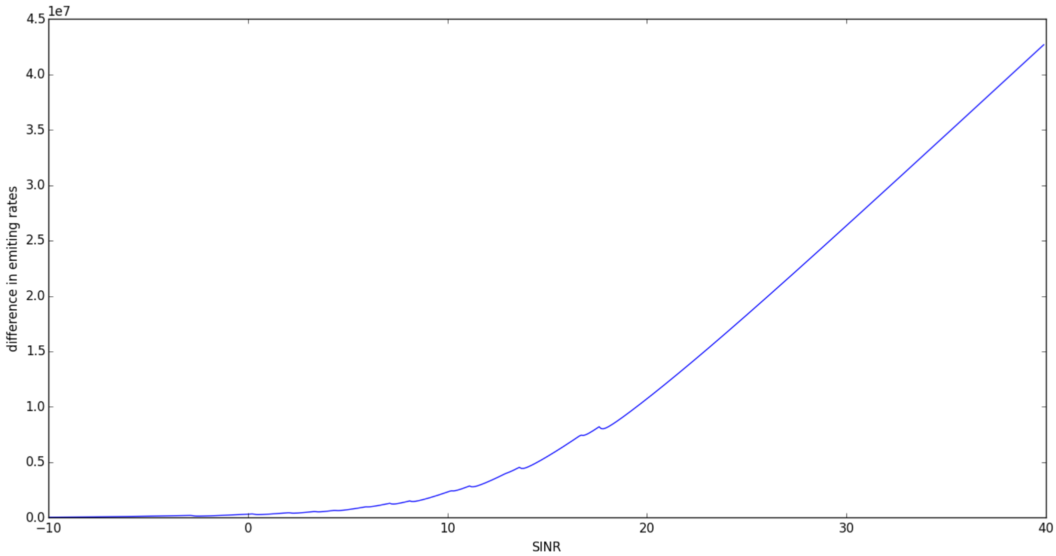


Figure 1.13: figure 4