

COMP4097 Mobile Computing

Assignment 2

Jonatan Juhas (19502966)

November 20, 2019

1. a) As the hamming code (7 4) requires a message length of 4, we split it into 2 parts: 1000 and 1010.

The first part (assuming the pattern M7 M6 M5 P4 M3 P2 P1) is:

$$\begin{array}{rcl} 1 & 0 & 0 \quad - \quad 0 \quad - \quad - \\ 1 & 0 & 0 \quad - \quad 0 \quad - \quad 1 \\ 1 & 0 & 0 \quad - \quad 0 \quad 1 \quad 1 \\ 1 & 0 & 0 \quad 0 \quad 0 \quad 1 \quad 1 \end{array} \begin{array}{l} \downarrow P1: \\ \downarrow P2: \\ \downarrow P4: \end{array} \begin{array}{l} _001 \rightarrow 0 + 0 + 1 = 1 \text{ (odd)} \rightarrow 1001 \\ _001 \rightarrow 0 + 0 + 1 = 1 \text{ (odd)} \rightarrow 1001 \\ _101 \rightarrow 1 + 0 + 1 = 2 \text{ (even)} \rightarrow 0101 \end{array}$$

The second part is:

$$\begin{array}{rcl} 1 & 0 & 1 \quad - \quad 0 \quad - \quad - \\ 1 & 0 & 1 \quad - \quad 0 \quad - \quad 0 \\ 1 & 0 & 1 \quad - \quad 0 \quad 1 \quad 0 \\ 1 & 0 & 1 \quad 0 \quad 0 \quad 1 \quad 0 \end{array} \begin{array}{l} \downarrow P1: \\ \downarrow P2: \\ \downarrow P4: \end{array} \begin{array}{l} _001 \rightarrow 0 + 1 + 1 = 2 \text{ (even)} \rightarrow 0011 \\ _001 \rightarrow 0 + 0 + 1 = 1 \text{ (odd)} \rightarrow 1001 \\ _101 \rightarrow 1 + 0 + 1 = 2 \text{ (even)} \rightarrow 0101 \end{array}$$

So the message transmitted to Station B will be 1001011 1010010

1. b) We receive 2 messages: 1011011 and 1011010.

For the first message we check all parity bits:

$$P1: 1011 \rightarrow 1 + 0 + 1 + 1 = 3 \quad \text{✗}$$

$$P2: 1001 \rightarrow 1 + 0 + 0 + 1 = 2 \quad \text{✓}$$

$$P4: 1101 \rightarrow 1 + 1 + 0 + 1 = 3 \quad \text{✗}$$

Correcting the message with $P1 + P4 = 5$ to 1001011.

We check the parity bits in the second message:

P1: $0011 \rightarrow 0 + 0 + 1 + 1 = 2$ ✓

P2: $1001 \rightarrow 1 + 0 + 0 + 1 = 2$ ✓

P4: $1101 \rightarrow 1 + 1 + 0 + 1 = 3$ ✗

Correcting the message with $P4 = 4$ to 1010010.

1. c) We again receive 2 messages after the spike: 1100111 and 1011010.

The first message check results in:

P1: $1101 \rightarrow 1 + 1 + 0 + 1 = 3$ ✗

P2: $1111 \rightarrow 1 + 1 + 1 + 1 = 4$ ✓

P4: $0011 \rightarrow 0 + 0 + 1 + 1 = 2$ ✓

Correcting the message with $P1 = 1$ to 1100110.

The second message check:

P1: $0011 \rightarrow 0 + 0 + 1 + 1 = 2$ ✓

P2: $1001 \rightarrow 1 + 0 + 0 + 1 = 2$ ✓

P4: $1101 \rightarrow 1 + 1 + 0 + 1 = 3$ ✗

Correcting the message with $P4 = 4$ to 1010010.

The channel seems to be quite unreliable but we manage to fix all errors thanks to the hamming code encoding.

2. a) The equation for calculating the rate can be summarized as:

$$\frac{\text{payload bytes} * 8 * 1000}{\text{\#slots} * 0.625 * 1000}$$

An ACL Link (Asynchronous Connection-Less) using DM5 and maximum number of bytes can achieve a rate of $\frac{224*8*1000}{5*0.625*1000} \approx 477.8kb/s$ as indicated in the table under asymmetric rate.

The reverse connection rate uses DM1 so we adjust the number of payload bytes to 17 and number of slots to 1 so we get $\frac{17*8*1000}{1*0.625*1000} \approx 108.8kb/s$.

2. b) $2Mb = 2000kb$ so with a transfer speed of $477.8kbps$ we can transfer the image in $\approx 4.1s$.

2. c) Since the high quality stream requires a minimum transfer speed of $256kbps$, which is below our theoretical maximum of $477.8kbps$, the connection will be stable.

The "professional" grade mp3 transfer requires $512kbps$ transfer rate, which exceeds our limits. Certain packets will therefore come delayed or not at all which will result in stuttering in the audio.

2. d) An SCO HV3 link only supports a maximum connection speed of $64kbps$, which is way below our limit. Most packets will not arrive at the slave on time or at all and will be dropped. The listening experience will be drastically diminished.

3. a) Given the frequency reuse of 7 and a semi-random distribution of the initial cell frequencies (2-4-6-3-5-7) we construct a hexagonal lattice as shown in figure 1.

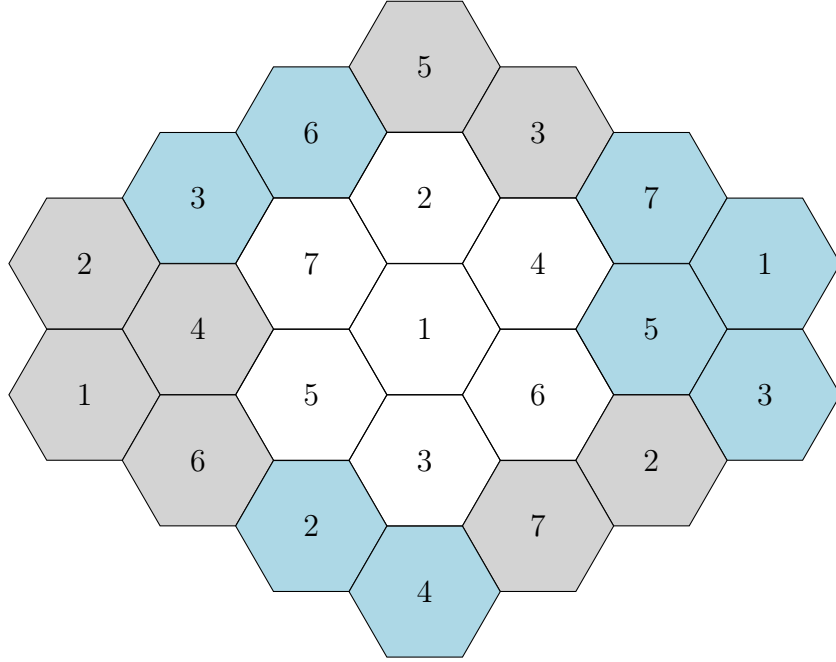


Figure 1: Suggested cell layout using frequency reuse of 7

The cell area can be calculated as $1.5\sqrt{3}R^2$. Since our radius is 1km for each cell, one cell covers the surface of $\approx 2.6km^2$. Our are consists of 23

segments which totals an area of $\approx 59.8km^2$.

3. b) In the past, before the cell architecture was introduced, a single antenna could only handle a specific amount of simultaneous connections - 25 channels on a high powered transmitter within an 80km reach.

By decreasing this reach and using more transmitters, that do not interfere with each other (such as 5G antennas), the effective amount of devices capable of connecting (let's say within a city) increases. This is very important especially for the future of IoT technology.

4. During paging the MTSO (Mobile Telecommunications Switching Office) tries to connect to a specific device and deliver a call. It sends a paging request to specific BS's within an area based on the number called, each transmitting it on their own channel.

The station to which the phone being called is connected then establishes connection.

5. a) Given the chipping codes:

A: -1 -1 -1 -1

B: -1 -1 1 1

C: -1 1 -1 1

Building a dot product we get $(-1 * -1 * -1) + (-1 * -1 * 1) + (-1 * 1 * -1) + (-1 * 1 * 1) = 0$. Thus the 3 chipping codes are orthogonal and can be used for CDMA.

5. b) For the first bit we generate the code for A (1), B (1) and C (0) by combining A and B's chipping codes and the inverse of C's chipping code (1 -1 1 -1):

$$(-1-1+1) (-1-1-1) (-1+1+1) (-1+1-1) = -1 -3 1 -1$$

The second bit pattern A (0) - inverse, B (1), C (1)

$$(1-1-1) (1-1+1) (1+1-1) (1+1+1) = -1 1 1 3$$

The third bit pattern A (1), B (1), C (1)

$$(-1-1-1) (-1-1+1) (-1+1-1) (-1+1+1) = -3 -1 -1 1$$

The receiver will receive the signal pattern D -1 -3 1 -1 | -1 1 1 3 | -3 -1 -1 1

5. c) We take the input received and multiply it by A's chipping code:
 $(-1 * -1) + (-3 * -1) + (1 * -1) + (-1 * -1) = 4$ so A must have sent a 1
 $(-1 * -1) + (1 * -1) + (1 * -1) + (3 * -1) = -4$ so the second bit is 0
 $(-3 * -1) + (-1 * -1) + (-1 * -1) + (1 * -1) = 4$ so the last bit was 1

The bit pattern sent by A was therefore 101 which is indeed correct.

5. d) We proceed as in the last part:
 $(-1 * -1) + (-3 * -1) + (1 * 1) + (-1 * 1) = 4 \rightarrow 1$
 $(-1 * -1) + (1 * -1) + (1 * 1) + (3 * 1) = 4 \rightarrow 1$
 $(-3 * -1) + (-1 * -1) + (-1 * 1) + (1 * 1) = 4 \rightarrow 1$

So the bit pattern sent by B was 111

6. a) We can convert the DMS (degrees, minutes, seconds) format into the decimal format by using the simple formula $d + \frac{m}{60} + \frac{s}{3600}$. The result will be
 $22 + \frac{15}{60} + \frac{16.9}{3600} = 22.254694$, $113 + \frac{54}{60} + \frac{15}{3600} = 113.90416$

22.254694, 113.90417 (aka Tian Tan Buddha)

6. b) The device can send a timestamp to the satellite and then wait for a response. The satellite sends the timestamp back and the device can then measure how long it took for the signal to travel to the satellite and back estimating the distance to it. If at least 3 satellites can be reached the location of the device can be triangulated by "drawing" a circle around each satellite with the radius of the calculated distance and finding the spot where they overlap.

Using more satellites leads to better positioning results as well as above sea height measurements.