Lab Exercise 6: Report

Convolutional Coded Systems with

BPSK, 8PSK, and 16QAM

Group 3

許博凱、林柏亨、李柏希



Figure 1: The First Lady of the Internet

**1. Introduction**

Convolutional coded system can significantly reduce the bit error rate (BER) by inserting extra bits to protect the data. In addition, we implement trellis termination in this lab to obtain better performance. The generators used in the simulation are *g1­* = [101] and *g2* = [111]. Every 120 bits in the input sequence are collected to form a packet. The codeword’s length is therefore 240 bits--twice the packet size.

**2. Simulation**

The simulation is written using C++, with *N0* = 2 J. It is based largely upon the former results in Lab 2, 3, 4, and 5.

**3. Results**

With convolutional code

|  |  |  |  |
| --- | --- | --- | --- |
| SNR | BPSK | 8PSK | 16QAM |
| 0 | C:\Users\Brian\Downloads\Lab6\BPSK\code0.bmp | C:\Users\Brian\Downloads\Lab6\PSK8\code0.bmp | C:\Users\Brian\Downloads\Lab6\QAM16\code0.bmp |
| 1 | C:\Users\Brian\Downloads\Lab6\BPSK\code1.bmp | C:\Users\Brian\Downloads\Lab6\PSK8\code1.bmp | C:\Users\Brian\Downloads\Lab6\QAM16\code1.bmp |
| 2 | C:\Users\Brian\Downloads\Lab6\BPSK\code2.bmp | C:\Users\Brian\Downloads\Lab6\PSK8\code2.bmp | C:\Users\Brian\Downloads\Lab6\QAM16\code2.bmp |
| 3 | C:\Users\Brian\Downloads\Lab6\BPSK\code3.bmp | C:\Users\Brian\Downloads\Lab6\PSK8\code3.bmp | C:\Users\Brian\Downloads\Lab6\QAM16\code3.bmp |
| 4 | C:\Users\Brian\Downloads\Lab6\BPSK\code4.bmp | C:\Users\Brian\Downloads\Lab6\PSK8\code4.bmp | C:\Users\Brian\Downloads\Lab6\QAM16\code4.bmp |
| 5 | C:\Users\Brian\Downloads\Lab6\BPSK\code5.bmp | C:\Users\Brian\Downloads\Lab6\PSK8\code5.bmp | C:\Users\Brian\Downloads\Lab6\QAM16\code5.bmp |
| 6 | C:\Users\Brian\Downloads\Lab6\BPSK\code6.bmp | C:\Users\Brian\Downloads\Lab6\PSK8\code6.bmp | C:\Users\Brian\Downloads\Lab6\QAM16\code6.bmp |
| 7 | C:\Users\Brian\Downloads\Lab6\BPSK\code7.bmp | C:\Users\Brian\Downloads\Lab6\PSK8\code7.bmp | C:\Users\Brian\Downloads\Lab6\QAM16\code7.bmp |
| 8 | C:\Users\Brian\Downloads\Lab6\BPSK\code8.bmp | C:\Users\Brian\Downloads\Lab6\PSK8\code8.bmp | C:\Users\Brian\Downloads\Lab6\QAM16\code8.bmp |
| 9 | C:\Users\Brian\Downloads\Lab6\BPSK\code9.bmp | C:\Users\Brian\Downloads\Lab6\PSK8\code9.bmp | C:\Users\Brian\Downloads\Lab6\QAM16\code9.bmp |
| 10 | C:\Users\Brian\Downloads\Lab6\BPSK\code10.bmp | C:\Users\Brian\Downloads\Lab6\PSK8\code10.bmp | C:\Users\Brian\Downloads\Lab6\QAM16\code10.bmp |
| 11 |  | C:\Users\Brian\Downloads\Lab6\PSK8\code11.bmp | C:\Users\Brian\Downloads\Lab6\QAM16\code11.bmp |
| 12 |  | C:\Users\Brian\Downloads\Lab6\PSK8\code12.bmp | C:\Users\Brian\Downloads\Lab6\QAM16\code12.bmp |
| 13 |  | C:\Users\Brian\Downloads\Lab6\PSK8\code13.bmp | C:\Users\Brian\Downloads\Lab6\QAM16\code13.bmp |
| 14 |  | C:\Users\Brian\Downloads\Lab6\PSK8\code14.bmp | C:\Users\Brian\Downloads\Lab6\QAM16\code14.bmp |

Without convolutional code

|  |  |  |  |
| --- | --- | --- | --- |
| SNR | BPSK | 8PSK | 16QAM |
| 0 |  |  |  |
| 1 |  |  |  |
| 2 |  |  |  |
| 3 |  |  |  |
| 4 |  |  |  |
| 5 |  |  |  |
| 6 |  |  |  |
| 7 |  |  |  |
| 8 |  |  |  |
| 9 |  |  |  |

PSNR values: (setting PSNR between identical images to 100 dB)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| SNR | BPSK | | 8PSK | | 16QAM | |
| No CC | CC | No CC | CC | No CC | CC |
| 0 | 13.52 | 17.92 | 11.57 | 13.59 | 11.77 | 12.58 |
| 1 | 14.78 | 20.82 | 12.05 | 15.74 | 12.13 | 14.39 |
| 2 | 15.91 | 27.20 | 13.14 | 18.39 | 12.82 | 16.52 |
| 3 | 18.06 | 34.12 | 13.93 | 20.79 | 13.27 | 18.96 |
| 4 | 19.96 | 43.28 | 15.04 | 24.99 | 14.22 | 21.88 |
| 5 | 23.79 | 52.95 | 16.40 | 31.51 | 15.58 | 27.40 |
| 6 | 26.34 | 100 | 17.86 | 37.35 | 17.22 | 33.24 |
| 7 | 29.47 | 100 | 20.06 | 44.94 | 19.48 | 40.19 |
| 8 | 40.22 | 100 | 22.60 | 56.46 | 21.91 | 49.40 |
| 9 | 48.17 | 100 | 25.87 | 57.79 | 25.11 | 65.80 |
| 10 | 58.22 | 100 | 31.25 | 72.25 | 29.58 | 100 |
| 11 |  |  | 36.62 | 100 | 34.40 | 100 |
| 12 |  |  | 43.04 | 100 | 40.52 | 100 |
| 13 |  |  | 52.35 | 100 | 48.50 | 100 |
| 14 |  |  | 100 | 100 | 66.80 | 100 |

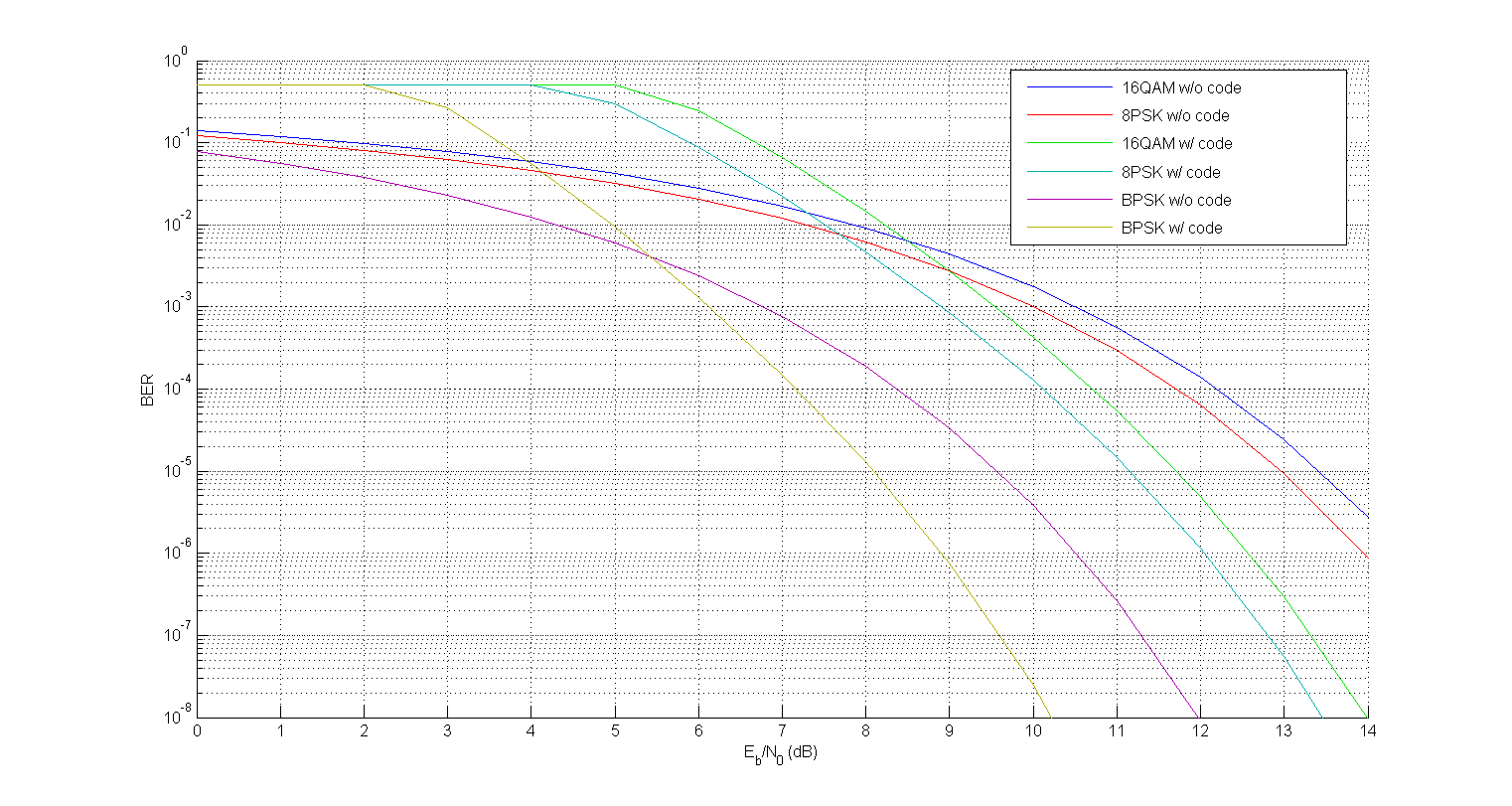
****

Figure 2: BER plotted using BERTool in MATLAB

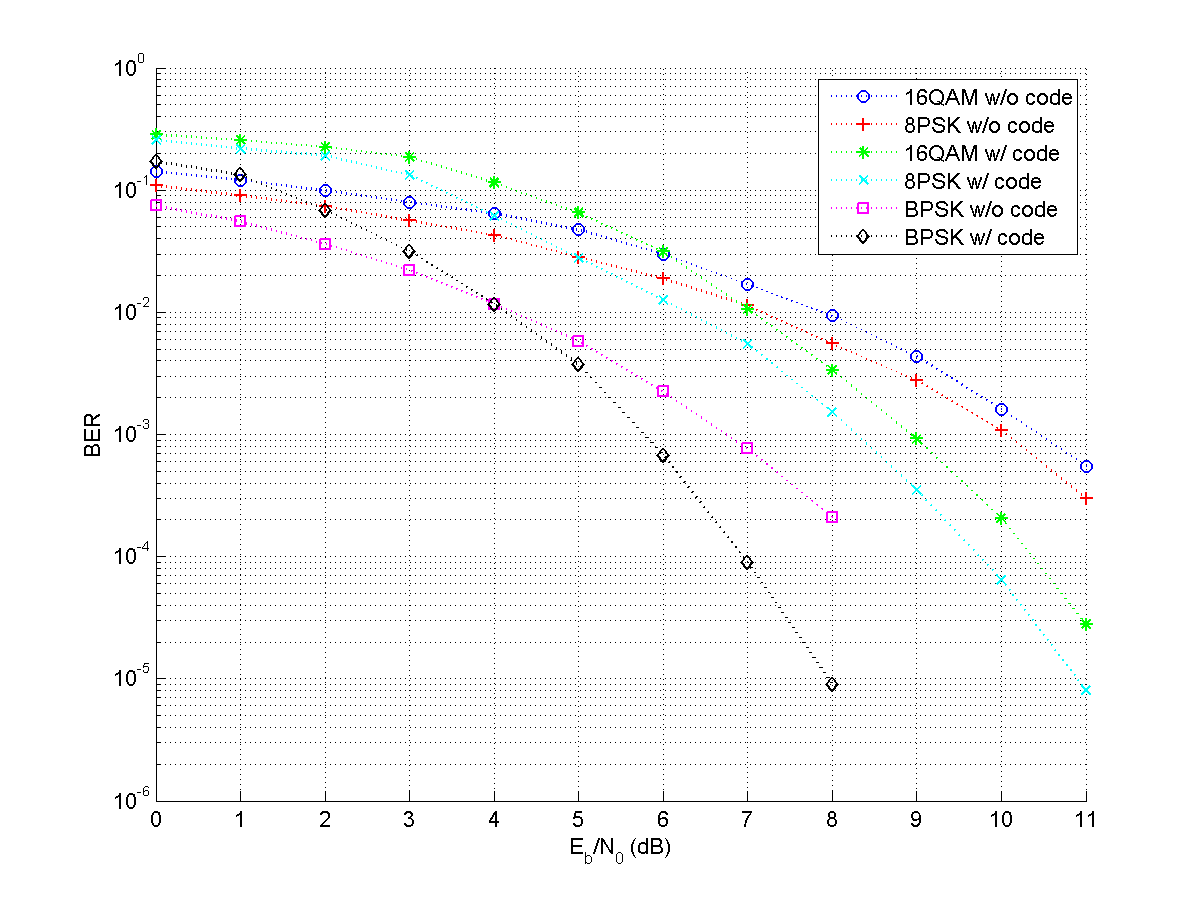


Figure 3: Simulated BER

**4. Conclusion**

From Figure 3 it can be observed that convolutional code (CC) indeed decrease the bit error rate when SNR is high. The simulated results also match the theoretical ones nicely. Identical images are even received with CC when *Eb/N0* is high (PSNR = 100).

The peak signal-to-noise ratio (PSNR) is defined as

,

where *MAX* is the maximum possible pixel value. The *MSE* (mean square error) between two *m**n* images *I1­* ­­and *I2* is calculated with

.

Typical values in lossy image compression are between 30 and 50 dB. Acceptable values for wireless transmission are about 20 to 25 dB. The PSNR in this lab are calculated by first converting all images to grayscale. Because the bmp files offered are indexed 16 bpp, some palette lookups need to be done before the images are transformed into grayscale. We can see from the simulation results that the PSNRs of bmp images increase monotonously, with the ones having convolutional code higher than those that do not.

Unfortunately, whether PSNR is an adequate indicator for transmission quality is not clear. The following images all consists of only two colors--black and white, and are all obtained by applying a particular dithering on the original image. The captions under the images show the different dithering algorithms. The number following “Bayer” indicates the size of the threshold map. Although they all have PSNR below the values of those transferred under BER = 10-1, they surely look better than their fuzzy counterparts presented in the simulation results, with higher contrast. Also, among the following white-and-black images, the one using average dithering has the highest PSNR. Yet one might argue that Stucki or Bayer 64 looks better.

|  |  |  |
| --- | --- | --- |
| E:\Dropbox\大三\大三下\通訊實驗\Lab5\lena_bayer4.png  Bayer 4  9.31 | E:\Dropbox\大三\大三下\通訊實驗\Lab5\lena_bayer9.png  Bayer 9  9.06 | E:\Dropbox\大三\大三下\通訊實驗\Lab5\lena_bayer16.png  Bayer 16  8.95 |
| E:\Dropbox\大三\大三下\通訊實驗\Lab5\lena_bayer64.png  Bayer 64  8.91 | E:\Dropbox\大三\大三下\通訊實驗\Lab5\lena_jarvis_judice_steinberg.png  Jarvis, Judis &Ninke  10.84 | E:\Dropbox\大三\大三下\通訊實驗\Lab5\lena_stkinson.png  Atkinson  10.88 |
| E:\Dropbox\大三\大三下\通訊實驗\Lab5\lena_stucki.png  Stucki  10.96 | E:\Dropbox\大三\大三下\通訊實驗\Lab5\floyd.png  Floyd-Steinberg  9.79 | E:\Dropbox\大三\大三下\通訊實驗\Lab5\lena_threshold.png  average dithering  11.74 |

**Appendix**

To analysis the quality of sound, we try to measure the sum of different square value, define as:

the sum of different square value :=







To listen the sound of handel, you need to use Matlab and follow the action below:

Change directory to CommLab6\_group3 and Type the following command in command line:

>> listen(modul, snr) // module == 2 or 8 or 16 snr == 0~9 or 13

For example : >> listen(2,9);

You will hear the sound of handel and feel the quality of sound.

To plot above figure to analysis the sound of handel, you need to use Matlab and follow the action below, and Type the following command in command line:

>> runSound(modul) // module == 2 or 8 or 16

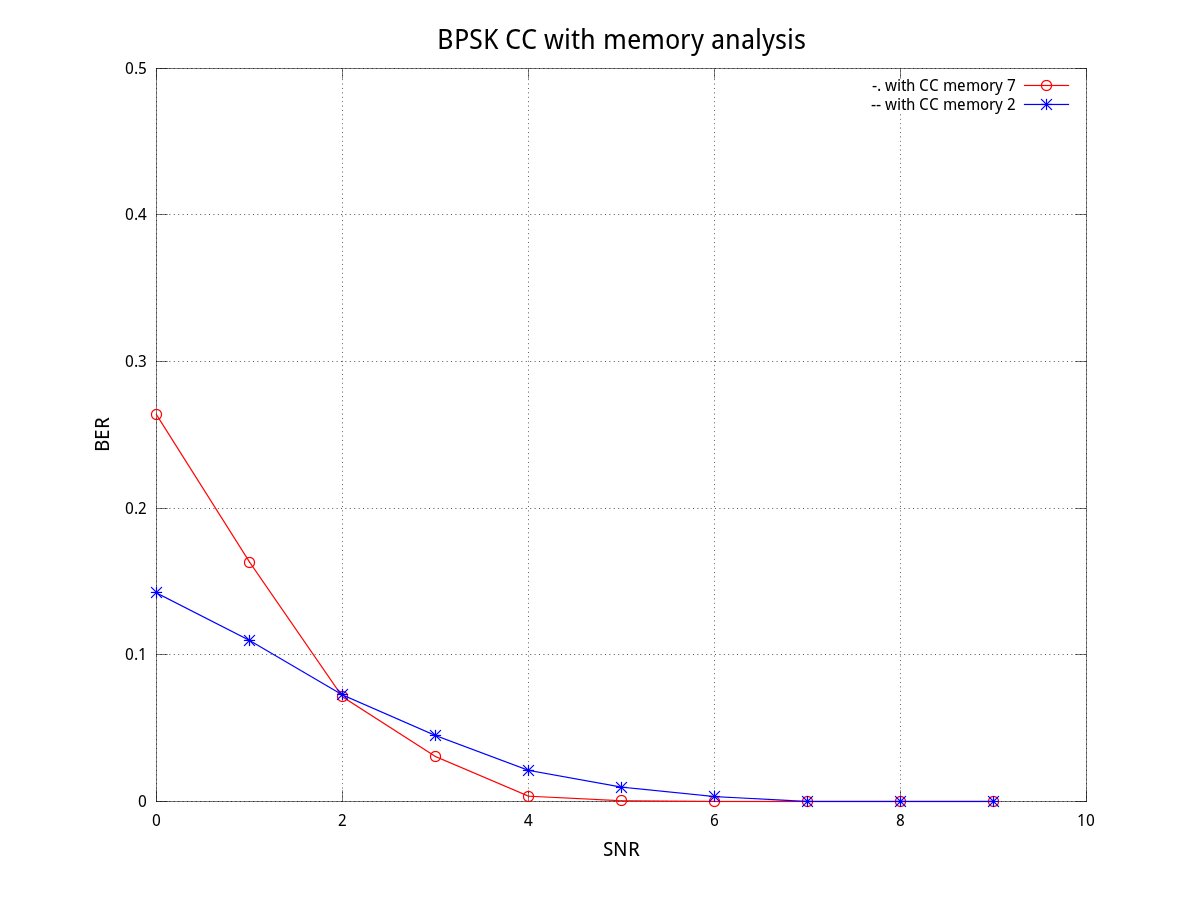
For example : >> runSound(2);

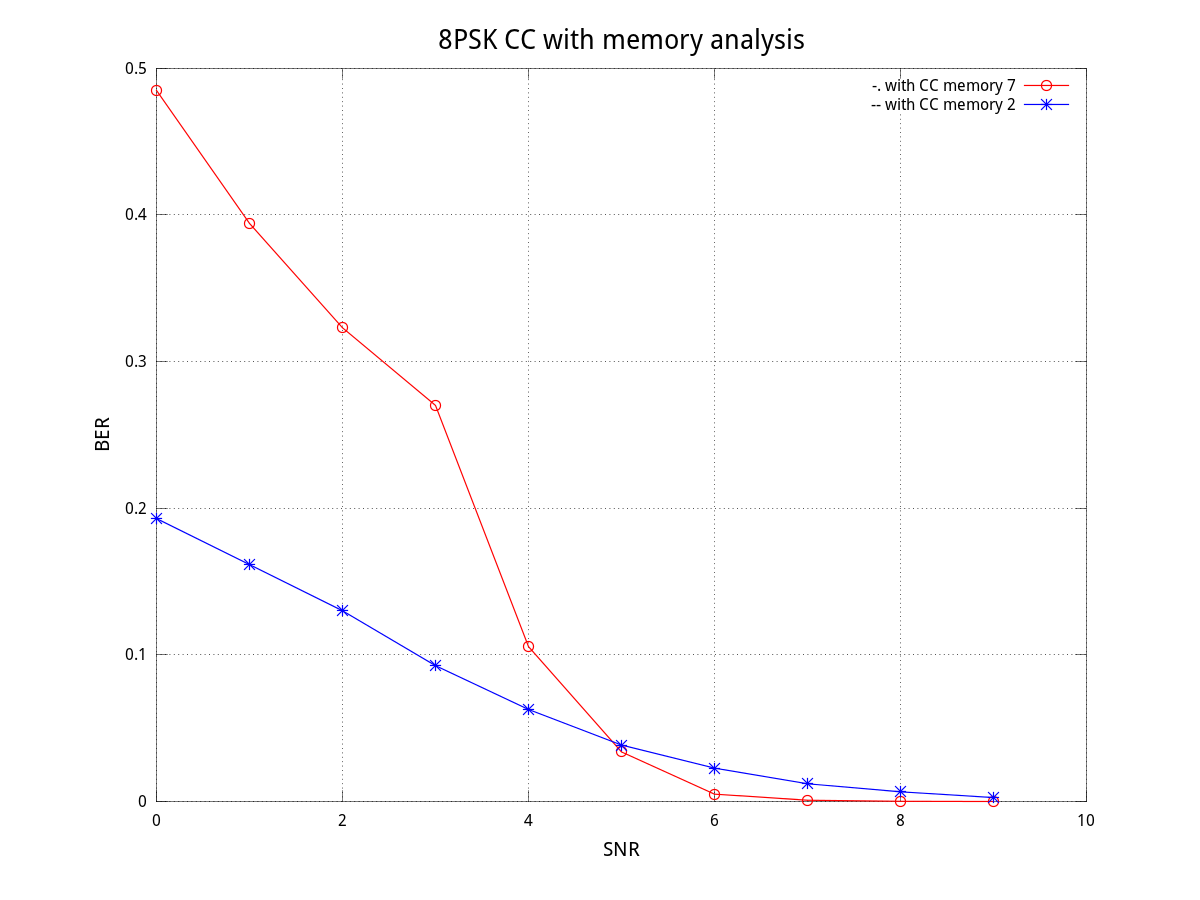
You will see the figure ‘BPSK handel CC analysis’

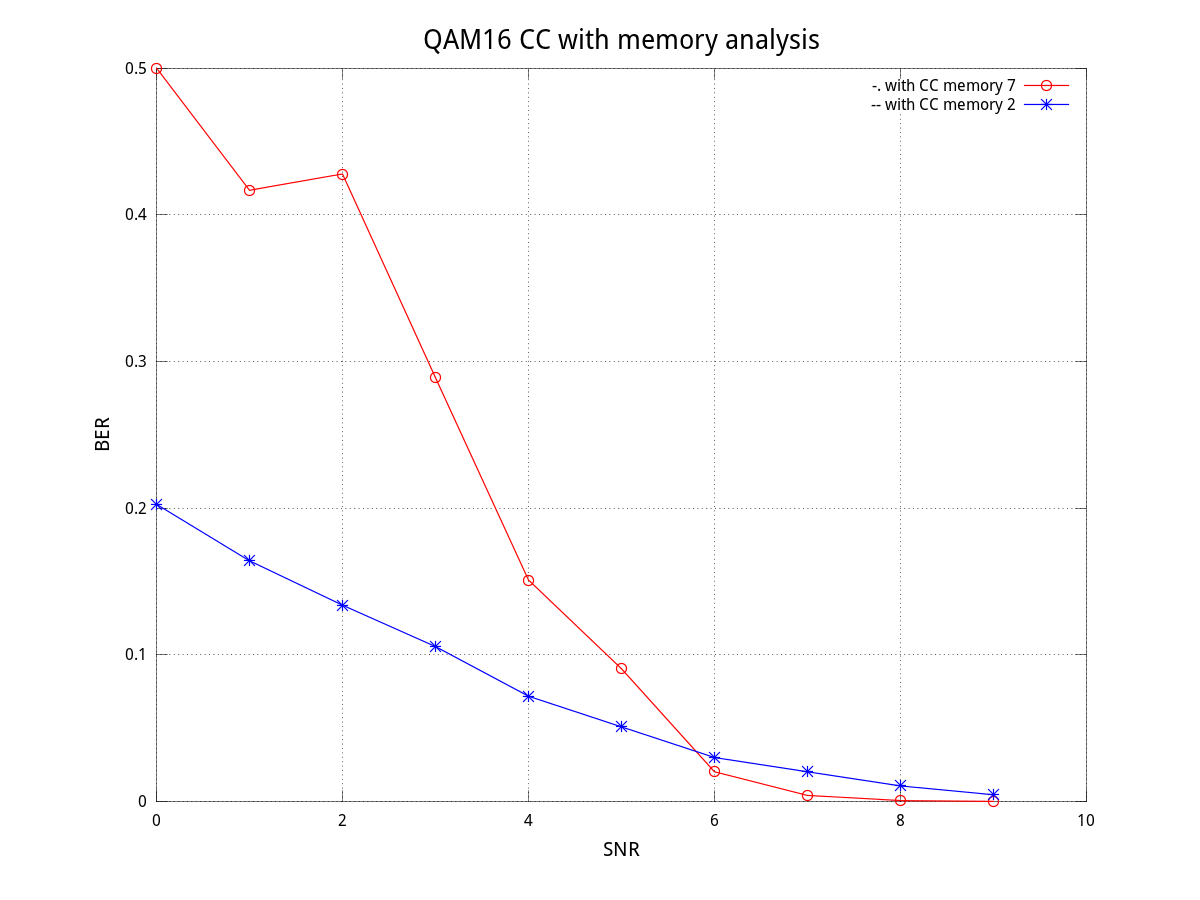
=======================================================================

Exercise Option:

CC with memory 7 : {g1 ; g2} = {133 ; 171} analysis







|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| BPSK\SNR | 0 | 1 | 2 | 3 | 4 |
| Memory 7 | 0.26363 | 0.163235 | 0.071296 | 0.030622 | 0.003628 |
| Memory 2 | 0.142130 | 0.109783 | 0.072619 | 0.044940 | 0.021186 |
| SNR | 5 | 6 | 7 | 8 | 9 |
| Memory 7 | 0.000491 | 0.000044 | x | x | X |
| Memory 2 | 0.009766 | 0.003329 | x | x | X |
| 8PSK/SNR | 0 | 1 | 2 | 3 | 4 |
| Memory 7 | 0.484722 | 0.394048 | 0.322917 | 0.270000 | 0.105556 |
| Memory 2 | 0.192857 | 0.161458 | 0.130000 | 0.092593 | 0.062708 |
| SNR | 5 | 6 | 7 | 8 | 9 |
| Memory 7 | 0.033784 | 0.005007 | 0.000866 | 0.000150 | 0.000012 |
| Memory 2 | 0.038462 | 0.022727 | 0.012063 | 0.006566 | 0.002646 |
| 16QAM/SNR | 0 | 1 | 2 | 3 | 4 |
| Memory 7 | 0.500000 | 0.416667 | 0.427778 | 0.288889 | 0.150980 |
| Memory 2 | 0.202564 | 0.164062 | 0.133772 | 0.105556 | 0.071759 |
| SNR | 5 | 6 | 7 | 8 | 9 |
| Memory 7 | 0.090476 | 0.020238 | 0.004127 | 0.000615 | 0.000070 |
| Memory 2 | 0.050833 | 0.029960 | 0.020200 | 0.010609 | 0.004561 |

When the SNR is large enough, the BER with memory 7 decrease lower than with memory 2.