Uriel Escobar
uesco001

EE114 Coding HW #2

1) Code shown in part1.py file. The output of can be shown in Figure 1.1

```
root@DESKTOP-2N5UBQD:~/EE114/HW2# python3 part1.py
Input probability p  .1
Input number of bits N   10
Input number encoding R   9
transmission success!
Input probability p
```

Figure 1.1 terminal output for part1.py

2) $P_{success}$ = 0.993

The output of part2 can be shown in figure 1.2

```
root@DESKTOP-2N5UBQD:~/EE114/HW2# python3 part2.py -j8
probabilty for
N = 10
R = 9
p = 0.1
is  0.993
```

Figure 1.2 terminal output for part2.py

3)

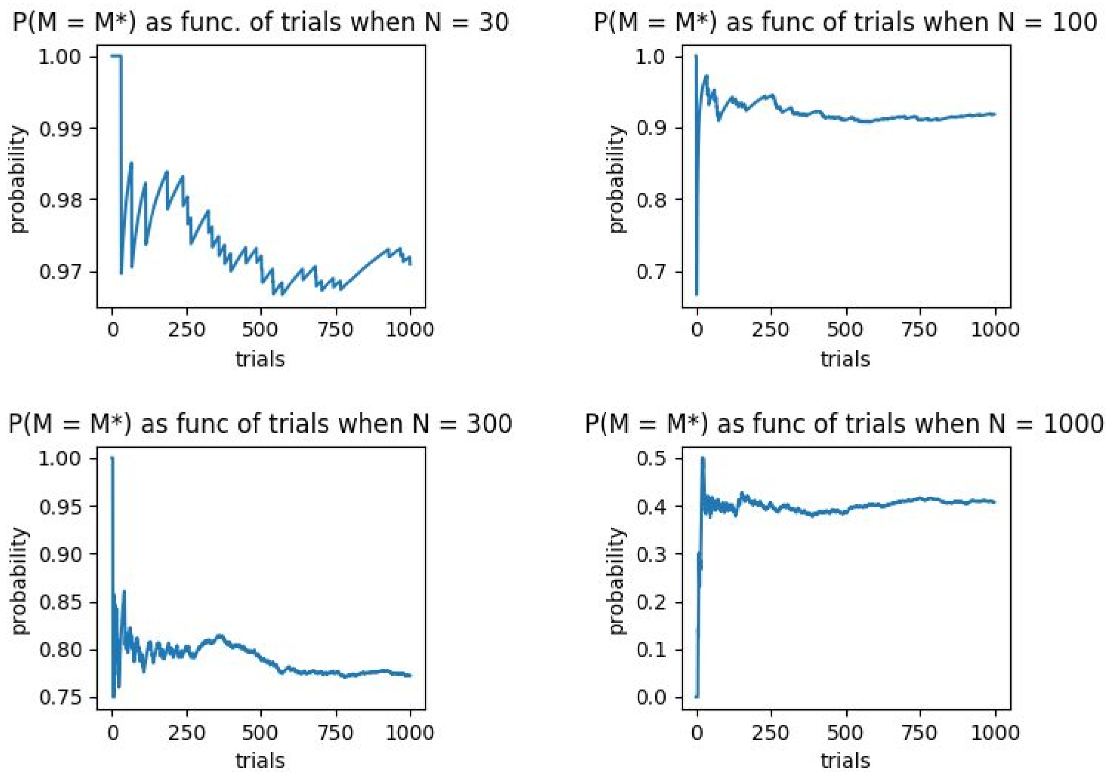

Figure 1.3 Graphs of Psucc as a function of trials for different N

Figure 1.3 shows how the probability of P(M=M*) changes as the trials increase. As the trials increase we see that the probability converges to one point. We can see that as N increases the probability changes. Figure 1.4 shows how P(M=M*) changes as N changes. P(M=M*) is inversely proportional to N.

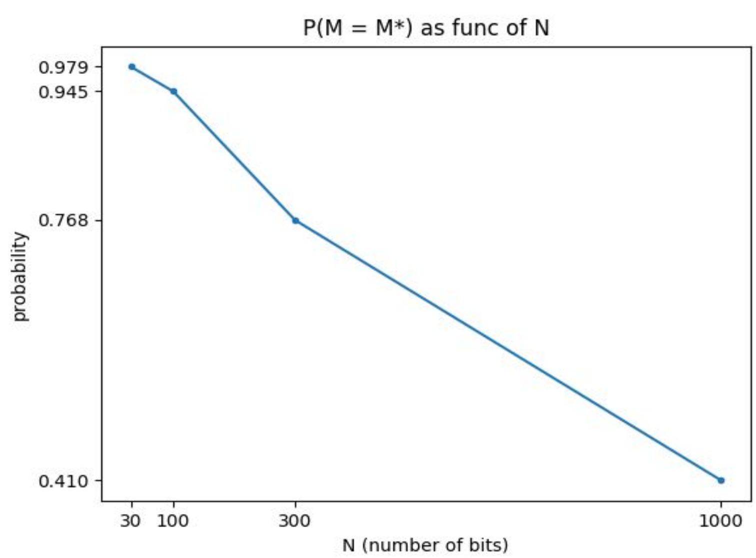

Figure 1.4 Graph of P(M=M) as a function of N*

4) & 5)
When p = .2 and N = 100 having a Psucces > 0.9 means R should change.

```
R must be  22 so Psuccess is >= .9
R must be  34 so Psuccess is >= .99
R must be  40 so Psuccess is >= .999
R must be  40 so Psuccess is >= .9999
root@DESKTOP-2N5UBQD:~/EE114/HW2# python3 part5.py
R must be  22 so Psuccess is >= .9
R must be  30 so Psuccess is >= .99
R must be  38 so Psuccess is >= .999
R must be  38 so Psuccess is >= .9999
root@DESKTOP-2N5UBQD:~/EE114/HW2# python3 part5.py
R must be  22 so Psuccess is >= .9
R must be  34 so Psuccess is >= .99
R must be  38 so Psuccess is >= .999
R must be  40 so Psuccess is >= .9999
root@DESKTOP-2N5UBQD:~/EE114/HW2# python3 part5.py
R must be  22 so Psuccess is >= .9
R must be  32 so Psuccess is >= .99
R must be  38 so Psuccess is >= .999
R must be  38 so Psuccess is >= .9999
root@DESKTOP-2N5UBQD:~/EE114/HW2# python3 part5.py
R must be  24 so Psuccess is >= .9
R must be  32 so Psuccess is >= .99
R must be  39 so Psuccess is >= .999
R must be  40 so Psuccess is >= .9999
root@DESKTOP-2N5UBQD:~/EE114/HW2# python3 part5.py
R must be  22 so Psuccess is >= .9
R must be  32 so Psuccess is >= .99
R must be  38 so Psuccess is >= .999
R must be  42 so Psuccess is >= .9999
```

Figure 1.5 Terminal output of Part 4&5, multiple times with 1000 trials

```
root@DESKTOP-2N5UBQD:~/EE114/HW2# python3 part5.py
R must be  22 so Psuccess is >= .9
R must be  32 so Psuccess is >= .99
R must be  42 so Psuccess is >= .999
R must be  52 so Psuccess is >= .9999
```

Figure 1.6 Terminal output of Part 4&5, with one trial of 1000000

As you can see in figure 1.5 R increases as the need for $P_{success}$ needs to increase. I ran the code 6 times and found that R changed for the same $P_{success}$. I figured this has to do with the number of trials being only 1000. So instead of using 1000 trials, I used 1 million trials. The results are shown in figure 1.6. As the need for $P_{success}$ to increases, so does R.

6) N, R, and p all change $P_{success}$. We find that N is inversely proportional to $P_{success}$ and R is directionally proportional. N increasing means that there are more bits to transmit so there is a higher chance a bit would flip. Increasing R better increases the approximation of bit flipping as shown in figure 1.3. $P_{success}$ is inversely proportional to p. A higher probability to flip a bit decreases the transmission succession.