

Simulating a Binary Symmetric Channel

Information Theory Lab 12

Objective

Understand the model of a Binary Symmetric Channel, and simulate a BSC by randomly introducing bit errors in a file.

Theoretical notions

A Binary Symmetric Channel has the following representation:

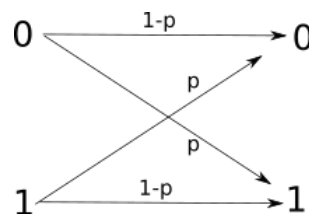


Figure 1: Binary symmetric channel (BSC)

With probability p , a bit will undergo an error, and with probability $1 - p$ it remains the same. Thus, p is known as *the probability of error*.

Practical issues

A data file is a sequence of bits (0/1).

Transmitting a data file over a BSC means that every bit in the original file has a chance p of undergoing an error.

Exercises

1. Write a C program to simulate a BSC for a given file. The program shall be called as follows:

```
BSC.exe 0.01 input.txt output.txt
```

- The arguments are:
 - 0.01: the error probability p of the channel
 - `input.txt`: the input file
 - `output.txt`: the output file
- The program will follow the following steps:
 - declare one large vector of `unsigned char` for input bits
 - open the input file and read everything into the input vector
 - for every bit in the input vector
 - * generate a random number x , and based on x do the following:
 - * toggle the bit, with probability p
 - * leave the bit unchanged, with probability $1 - p$
 - write the vector to the output data file

Implementation hints

- For randomly deciding when to make an error, with error probability p :
 - use `srand()` once, at the beginning of the program, to seed the random number generator
 - use `rand()` to generate a random number x in the range $[0 \dots RAND_MAX]$
 - x has $p\%$ chances to be smaller than $p \cdot RAND_MAX$
 - therefore: if $r < p \cdot RAND_MAX$, then change bit; otherwise, leave bit unchanged

Final questions

1. TBD
2. TBD