Telecom Network Systems

Cyclic Redundancy Check

P. Stallinga

MIEET 4º ano

By adding redundancy to data the size of the data is increased, without changing the amount of information. Note that that means that some combinations of bits are theoretically not possible. It is exactly for that reason that redundancy is added to data, in order so that the receiver can detect if something has gone wrong in the communication.
The most famous and widely used redundancy is parity. It adds a '0' or a '1' to a data word to make the total number of 1s even (odd). If now the receiver does not see an even (odd) number of 1s in the word, it knows something went wrong. It does not where (it can even be the parity bit itself), or why.
This technique is used in serial communication (RS232).

Example of even parity adding to 8-bit words:

Source word (without redundancy)	Channel word (with redundancy)
00000000	00000000**0**
00000001	00000001**1**
00000010	00000010**1**
|	|
11111111	11111111**0**

In case a word with wrong parity is received, the receiver can ask the sender to send it again. The receiver cannot correct the error itself. (As such, parity check only makes sense in duplex communication).

Note that if a word has two errors, they are undetected.

For the calculations we introduce the concept of bit-error rate (BER), which is the probability of 1 bit arriving wrong on the other side. We name this parameter *f*.

Exercise 1)
If we have 10,000 bits of data on a disk, with a BER of $f = 0.1$, what is the expectation value, standard deviation and probability distribution of number of flipped bits? (Draw a figure with the probability vs. number of bit flips).

Exercise 2)
With a disk of 1 GB of data, what should the BER be so that there is less than 0.1% error (999 out of 1000 disks contain no error)?

Exercise 3)
If we have no parity checking and send 1000 bytes of data, what is the probability of an (undetected) error in the communication (given a certain BER)? Add parity checking (1 bit per byte), what is now the probability of an undetected error in the communication?

More advanced forms of parity checking is cyclic redundancy check (CRC). “The basic idea behind CRCs is to treat the message string as a single binary word *M*, and divide it by a key word *k* that is known to both the transmitter and the receiver. The remainder *r* left after dividing *M* by *k* constitutes the 'check word' for the given message. The transmitter sends both the message string *M* and the check word *r*, and the receiver can then check the data by repeating the calculation, dividing *M* by the key word *k*, and verifying that the remainder is *r*.” (mathpages. Attached).

Exercise 4)
Calculate the CRC of message

```
M = 01110010001111100100
```

with key

```
k = 100101
```

Exercise 5)
Given a certain BER, what is the probability of an undetected error in a 20-bit message using a 6-bit CRC key?