# Binary Representation and Error-Correcting Codes

Andreas Boulios and Nakul Khambhati

October 29, 2023

## 1 Challenge

Imagine you're on a space mission and you want to send a message (a word, a video, a voice recording etc.) back to a friend of yours on Earth. It turns out that any message can be represented by a sequence of 1s and 0s which we will call a bit-string. For example, every natural number has a binary expansion. The first 7 natural numbers are $1, 10, 11, 100, 101, 110, 111$.

**Problem 1.1.**
Propose a way to represent any letter in the English alphabet (and therefore any word and any sentence) using only bit-strings.

In space, cosmic rays might interfere with your transmission and cause some bits of your message to flip! Indeed, even most methods of transferring data on Earth are prone to errros:

- Cellular data can get corrupted by random noise.
- Scratches in the back of a CD will alter the original message that was encoded on the CD.

However, despite the amount of scratches on a CD, the CD player can read the message with almost perfect accuracy.
**Goal:** Our goal will be to explore some methods for how a message can be encoded such that, even if some bits are lost in transmission, the receiver can deduce the original message.

## 2 Messages and Corruption

For simplicitly, assume that we are guarenteed that each time you send a bit-string to your friend, exactly one bit flips. We call this the corrupted message.

$$\begin{array}{c} 11001010 \\ \downarrow \\ 1\underline{0}001010 \end{array}$$

In general it is not hard to come up with a way to add some extra bits of redundancy to your message so that it can correct one bit flip:
**Brute force method:**
Suppose you want to send someone a 16-bit message, e.g.:

$$1011001110100101$$

Since you know that only one bit will flip, to help them recover it you can send three identical copies of the message, a total of 48 bits.

$$\begin{array}{c}1011001110100101\\1011001110100101\\1011001110100101\end{array}$$

**Problem 2.1.**

- How would you be able to recover the original 16 bit string from the corrupted 48 bit string?
- If two bits were corrupted would you always be able to recover the intended message?
- How many copies of the message would the sender have to send you to guarantee that you can recover the message if 2 bits were corrupted?

From now on, we will assume that through the transmission of our message **only one bit is corrupted**. Indeed the method we described above is inefficient, as it sends a message 3 times larger then the one we intended the receiver to get.

Can we find a more efficient method to add extra bits so that if any one of them is flipped, we can always locate the error and deduce the original message?

Before we delve into the more efficient methods, let's get a bit more familar with binary representations and why they are useful.

# 3 Binary Numbers and Yes or No questions

**Problem 3.1.**

Your classmate thinks of a number in the set $A = \{0, 1, \cdots, 14, 15\}$. You can ask a series of yes or no questions about the number. Your goal is to guess the number with the fewest possible questions. (*Hint: Each time, choose a subset of $A$ and ask whether your classmate's number is in given subset.*)

- Can you find a strategy that guarantees that you find it with at most 15 questions?
- Can you find a strategy with fewer questions?

**Problem 3.2.**

Once again your classmate thinks of a number between 0 and 15.[1] Each time you can ask a yes or no questions about the number's representation in binary: e.g. "Is the digit in the 4's place a 1?"[2]

- Can you find a strategy that guarantees to find the number with 4 questions? (Don't overthink it).
- Which set of numbers from 0 to 15 have a 1 in the 1's place?
- Which set of numbers from 0 to 15 have a 1 in the 2's place?
- Which set of numbers from 0 to 15 have a 1 in the 4's place?

[1]Recall that, just as in the decimal system, we can add extra 0s to the left of a number without changing its value. For example, 0 is the same as 0000 and 111 is the same as 0111.

[2]Recall that binary digits represent powers of two. The right-most digit is the 1's place, the next one is the 2's place, then the 4's place and so on.

- Which set of numbers from 0 to 15 have a 1 in the 8's place?
- Is there a difference between asking: "Is your number in the subset: $\{2, 3, 6, 7, 10, 11, 14, 15\}$?" and "Does your number when written in binary have a 1 at the 2's place?"
- Looking at the previous problem, can you now find which four questions can guarantee that you find your classmate's number?

**Problem 3.3.**
Suppose you ask your classmate the following questions:

1. Is your number in the subset $\{8, 9, 10, 11, 12, 13, 14, 15\}$?
2. Is your number in the subset $\{4, 5, 6, 7, 12, 13, 14, 15\}$?
3. Is your number in the subset $\{2, 3, 6, 7, 10, 11, 14, 15\}$?
4. Is your number in the subset $\{1, 3, 5, 7, 9, 11, 13, 15\}$?

Your classmate recently learned about the binary system and is excited to use it so instead of answering with a yes or a no, he responds 1 for yes and 0 for no. Suppose his responses in order are 1 1 0 0.

- What is the number he is thinking of?
- What is the binary representation of that number?
- Compare his responses with the number he was thinking of and explain their relationship.

**Problem 3.4.**
- If our classmate was thinking of a number between 0 and 31 (two times as many possible numbers), how many additional questions would we have to ask?
- If our classmate was thinking of a number between 0 and 2047, how many questions would we need then?
- Find a formula relating the possible numbers your classmate can choose from $N$ with the number of yes or no questions $Q$ we have to ask to find their number.

# 4 More puzzles

**Problem 4.1.**
Every binary string of length 8 is transmitted along with a 9th bit, which is the parity bit. This parity bit ensures that every string has an even number of 1s. If the number of 1s in the 8-bit string is odd, the parity bit is set to 1, otherwise, it's set to 0.
**Example:** If the 8-bit string is 11010101, then the parity bit is 0, and the transmitted string is 110101010. On the other hand if the message is 01001010 then the parity bit is 1 and the transmitted message is 010010101.
**Challenge:** If you receive the string 101110100, was there an error during transmission? Can you recover the original message? If yes, what is it? If not, why not?

**Problem 4.2.**
Two secret agents communicate using a series of 1s and 0s. To ensure their message isn't tampered with, they add two extra bits at the end of every 4-bit message. The first bit is a parity bit for the first and second bit, and the second is a parity bit for the first and third bit.
**Example**: If the message is 1011, then the transmitted string with extra bits is 101110.
**Challenge**: If the agents receive the string 101110 and are told that exactly one bit flipped, can they locate which one it is? If we also guarentee that none of the extra bits have been flipped (i.e. the corrupted bit is one of the first 4 bits), can we recover the original message?

**Problem 4.3.**
Given a binary string, your challenge is to turn all the bits to 0s by flipping one bit at a time. But there's a catch: every time you flip a bit, its adjacent bits (if they exist) are also flipped. Can you turn the string 1010 to 0000 in 2 moves? If yes, how? Can you convert the string 10010 to 11111?

**Problem 4.4.**
A Gray Code sequence is a binary sequence where two consecutive numbers differ in only one bit. A Gray Code sequence that covers all 2 bits is 00, 01, 11, 10. Can you find the Gray Code sequence for all 3 bits? Construct a path from 0000 to 1111 where each sequence in the path differs by only one bit from its predecessor.

# 5 Back to our problem

We are given a 16 bit string, e.g.

1011001110100101

For our own comfort lets visualize it in a 4 by 4 grid:

| | | | |
|---|---|---|---|
| 1 | 0 | 1 | 1 |
| 0 | 0 | 1 | 1 |
| 1 | 0 | 1 | 0 |
| 0 | 1 | 0 | 1 |

We label the cells of the grid from 0 to 15 starting from top left to bottom right.

**Problem 5.1.**
The grid is sent as a message and because of some noise, exactly one bit gets flipped. Suppose that along with the corrupted grid you are also given a magic error detection device. This device lets you plug in a subset of the indices of your grid (e.g. $\{0, 1, 5, 7, 14, 15\}$ and answers with a 1 if the error is located within those grid squares and with a 0 if it isn't. What questions are you going to ask to find the location of your error with only four questions?

Our final trick as a receiver will be to construct this "magic device". For this, we will need to figure out a clever way to choose our redundant bits.

**Problem 5.2.**
We receive an 7 bit message along with a secret note that says: "my message had an even number of 1s in it." If during the transmission of this message only a single bit was altered can you tell if there has been an error or not? (*Hint: You've seen this in the previous section.*)

**Problem 5.3.**
Devise a method that pads a 7 bit string with an 8th bit of your choosing, such that when your friend receives the message (who also knows your strategy) they can know whether or not exactly one bit has been flipped.

This 8th bit essentially functions as a "checker" or "parity" bit. With it's help the receiver can check whether any of the 8 bits had been corrupted or not. Similar to how we our "magic device" worked! In our example above we say that **this checker bit "was responsible" to check this 8 bit string.** Now we are ready to put everything together and finish our algorithm.

**Problem 5.4.**
You want to send the following 11 bit message: 101 0110 0001. Think of them in a square grid as below, again labeled from 0 to 15 but with 0 and all powers of two left empty.

| | | | |
|---|---|---|---|
| X | | | 1 |
| | 0 | 1 | 0 |
| | 1 | 1 | 0 |
| 0 | 0 | 0 | 1 |

Position 0 of the matrix will be empty for now and the other 4 positions, the powers of two, will be filled in with checker bits, each corresponding to a subset of 7 bits of your string.

- How should the transmitter choose those checker bits to give the receiver enough information to locate the error? Remember that with the help of each checker bit the receiver can know if there is an error in the collection of 8 bits the checker bit is respobsible for. (*Hint: The answer has something to do with binary numbers.*)
- Having agreed beforehand with the transmitter about the positions of the checker bits and the subsets each one is responsible for, can you now interpret the total of 15 bits to see if there is an error and where it is?

**Problem 5.5.**
For all the following problems you will receiver a 15 bit string in a 4x4 grid. Position 0 lf the grid will remain empty for now. The bits in positions 1,2, 4, and 8 will be checker bits. It helps to think of these numbers in their binary representation.

The checker bit **in position 8** is a 1 if there is an **odd number of 1s** in the grids whose number has a 1 in the **8's** position when written in binary. It is 0 otherwise. The bit in position **4** corresponds in the same way to the positions with a 1 in the **4's** place when written in binary. The bit in position **2** corresponds to the positions with a 1 in the **2's** place when written in binary. The bit in position **1** corresponds to the positions with a 1 in the **1's** place when written in binary. Find the message the sender wanted to send each time.

**5.5 a)**

| | | | |
|---|---|---|---|
| X | 1 | 0 | 1 |
| 0 | 0 | 0 | 1 |
| 1 | 1 | 0 | 0 |
| 0 | 0 | 0 | 1 |

**5.5 b)**

| | | | |
|---|---|---|---|
| X | 1 | 0 | 1 |
| 1 | 0 | 0 | 1 |
| 1 | 1 | 1 | 0 |
| 0 | 1 | 0 | 1 |

**Problem 5.6.**
Suppose now that during transmission of a 15 bit message you know 0 or 2 bits were corrupted. As the receiver will you be able to tell if the message has been corrupted or not?

**Problem 5.7.**
Using the same rule for the checker bits, someone sends you the following message:

| | | | |
|---|---|---|---|
| X | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 |
| 0 | 1 | 1 | 0 |
| 0 | 1 | 0 | 1 |

1. Check if the message has been corrupted.
2. Assuming at most 1 bit was corrupted, can you find it?
3. If you know 2 bits were corrupted which ones could they have been? (find possible pairs of bits).

It should be clear at this point that with our current method we can retrieve our original message if it has at most 1 corrupted bit. However, if it is possible for 0,1, or 2 bits to be corrupted, we can't tell if there are 1 or 2 corrupted bits just by looking at our message.

**Idea:** We want to send an 11 bit message. Assume that we have already filled our 4x4 grid with 15 bits using the previous method. This time we will also place a checker bit in position 0, according to the following rule: It is 1 if there is an odd number of 1's in the remaining 15 bits (positions 1 through 15) and 0 otherwise. If our 11 bit message is: 101 0110 0001 Then the 4x4 grid we send will be:

| | | | |
|---|---|---|---|
| 1 | 1 | 0 | 1 |
| 0 | 0 | 1 | 0 |
| 1 | 1 | 1 | 0 |
| 0 | 0 | 0 | 1 |

**Problem 5.8.**
During transmission of the message above **1 or 2** bits will be corrupted. Can the receiver tell the difference with the help of this new checker bit?

**Problem 5.9.**
A classmate (using the strategy in the previous problem) sends you messages using a device that can corrupt 0, 1, or 2 bits. For each of the following messages find the number of bits that were corrupted. If only 1 bit was corrupted, find the original 11-bit message.

**5.9 a)**

| | | | |
|---|---|---|---|
| 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 1 |
| 0 | 0 | 1 | 0 |
| 0 | 1 | 1 | 1 |

**5.9 b)**

| | | | |
|---|---|---|---|
| 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 |
| 0 | 0 | 1 | 1 |
| 0 | 0 | 0 | 1 |

**5.9 c)**

| | | | |
|---|---|---|---|
| 1 | 1 | 0 | 1 |
| 1 | 1 | 0 | 1 |
| 0 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 |

**5.9 d)**

| | | | |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 1 | 0 | 1 | 1 |
| 1 | 0 | 1 | 1 |
| 1 | 0 | 0 | 0 |

# 6 The Hat Problem

Imagine there are three prisoners who are going to be executed tomorrow morning. The king, being merciful, gives them a chance to earn their freedom.

The next day, each prisoner is given a hat to wear, either black or white. They can see the hats of the other two prisoners, but not their own. They are not allowed to communicate in any way.

The prisoners are told that at least one of them is wearing a white hat. They will simultaneously either guess the color of their hat or pass. If at least one of them guesses and all who guess are correct, they will be set free. If anyone who guesses is incorrect, or if they all pass, they will be executed.

**Problem 6.1.**
Find a strategy for the prisoners to maximize their chance of freedom.