The Fascinating Art
of the
Coding Theory

Extended Essay

Mathematics

International Baccalaureate Programme

Word Count: 4000

2021-2022

To what extent is error detection helpful when sending data from a satellite to the earth?

Table of Contents

Introduction

An Overview

Since the first uses of technology[1], there has always been a keen interest in transferring data quickly and precisely, avoiding "errors" that might distort the message[2]. This paper will focus on deep-spatial communications[3], and the useful applications of Error Correction Codes[4] (ECC) for the exploration of outer space. Space programmes such as Voyager [5] or the Hubble telescope[6] have introduced revolutionary ECC's which changed the way we conceive computational errors. Exploring these topics has been a dream of mine and this paper was the perfect opportunity to produce something useful from my researches by posing a research question that will lead me in the branches of sophisticated ECC technology.

If we only think how challenging it can be to send the correct message through people, it's difficult to conceive the challenges faced by mathematicians and scientists that developed systems to correct the immense errors occurring in space[7]. Throughout this paper we will discuss the history that brought humanity to understand the methods to correct and detect complicated errors, ensuring effective communication over immense light years or even parsec distances[8]; developing a sample algorithm that will correct errors sent from a satellite sending messages to space.

[1] "Compuational Systems." *Bartleby*, 28 Dec. 2021,www.bartleby.com/subject/engineering/computer-science/concepts/computational-systems.

[2] Techopedia. "What Is Error Correction? - Definition from Techopedia." *Techopedia.com*, Techopedia, 19 Sept. 2011, www.techopedia.com/definition/821/error-correction.

[3] Madhavapeddy, Anil, et al. An Architecture for Interspatial Communication.

[4] "Error Correcting Codes | Brilliant Math & Science Wiki." *Brilliant.org*, brilliant.org/wiki/error-correcting-codes/.

[5] "Voyager - Mission Timeline." *Nasa.gov*, 2019, voyager.jpl.nasa.gov

[6] NASA. "Home." *STScI*, Q Starter Kit, 2019, hubblesite.org.

[7] Baird, Daniel. "Space Communications: 7 Things You Need to Know." *NASA*, NASA, 5 Oct. 2020, https://www.nasa.gov/feature/goddard/2020/space-communications-7-things-you-need-to-know.

[8] "What Is a Parsec?" *BBC Science Focus Magazine*, www.sciencefocus.com/space/what-is-a-parsec/.

The Problem

The transistor, is in the list of the most innovative and useful invention of the past century[9]. It created an easier path to the development of technology, introducing binary arithmetic[10].

In computers any data is stored and processed with digits also named as bits, which can be either a 0 or a 1[11], the goal of the coding theory[12] is to develop efficient systems that will be able to detect and correct errors by adding redundancy[13]. These algorithms have found applications in different fields from simple technologies such as CDs and DVDs to deep space communication technologies (the real scope of the paper).

Exploring outer space has been and still is one of the priorities of humanity[14], understanding what's beyond our solar system and universe is a challenge that will still be faced for possibly hundreds of years. Progress, though, has been made as satellites are now able to travel immense distances and still be able to effectively communicate with our planet[15] (Voyager programmes reached interstellar space). This is only possible with error correction algorithms as they are an essential aspect for the functioning of these satellites.

[9] Ganapati, Priya. "Dec. 23, 1947: Transistor Opens Door to Digital Future." *Wired*, Conde Nast, 23 Dec. 2009, https://www.wired.com/2009/12/1223shockley-bardeen-brattain-transistor/#:~:text=Hill%2C%20New%20Jersey.-,It's%20been%20called%20the%20most%20important%20invention%20of%20the%2020th,developed%20to%20replace%20vacuum%20tubes.

[10] "Arithmetic Operations of Binary Numbers." *GeeksforGeeks*, 31 July 2021, https://www.geeksforgeeks.org/arithmetic-operations-of-binary-numbers/#:~:text=Binary%20arithmetic%20is%20an%20essential,two%20digits%3A%200%20and%201.

[11] "Binary Number System." *Mathsisfun.com*, 2016, www.mathsisfun.com/binary-number-system.html.

[12] Weisstein, Eric W. "Coding Theory." *Mathworld.wolfram.com*, mathworld.wolfram.com/CodingTheory.html.

[13] look at "redundancy section"

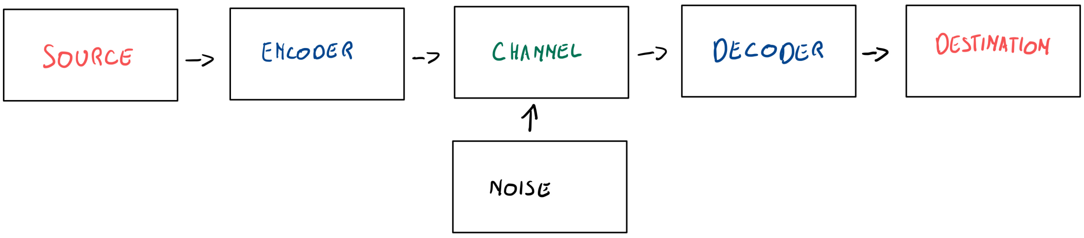
[14] Logsdon, John M. "Space Exploration | History, Definition, & Facts." *Encyclopædia Britannica*, 30 Jan. 2019, www.britannica.com/science/space-exploration.

[15] "Webb's Launch GSFC/NASA." *Webb.nasa.gov*, webb.nasa.gov/.

Communication Systems

Noisy Channel Coding Theorem

Figure 1



Author: Hand drawn by me

Before learning about ECC in deep space communication, we must understand the concept of channel and noise in technological terms. These are fundamental aspects for the algorithms we will later develop, as understanding these concepts should be considered a prerequisite before learning about the codes.

Messages are sent through a channel which must be imagined as long tube, in this channel there might be some holes that lead to errors, the tube will also have a fixed cross sectional area meaning it has a limited carrying capacity later named as "Channel Capacity"[16].

The concept of channel capacity was first introduced by C.E. Shannon[17] in the paper "A mathematical theory of communication"[18] in which he introduced the noisy channel coding theory and pointed its relevance for ECC. Even though channel capacity is beyond the goal of this paper it will be useful to define it.

A channel is a set of inputs each which gives you a probability distribution of outputs. The way you get the capacity of the channel is by inventing a probability distribution on inputs

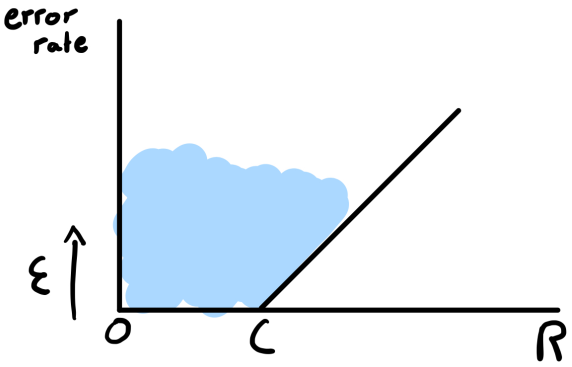
[16] "Channel Capacity - an Overview | ScienceDirect Topics." *Www.sciencedirect.com*, www.sciencedirect.com/topics/physics-and-astronomy/channel-capacity.

[17] "Claude E. Shannon — Information Theory Society." *Www.itsoc.org*, www.itsoc.org/about/shannon.

[18] Shannon, C. E. "A Mathematical Theory of Communication." *Bell System Technical Journal*, vol. 27, no. 4, Oct. 1948, pp. 623–656, 10.1002/j.1538-7305.1948.tb00917.x.

and then defining the mutual information between inputs and outputs and later maximise it with respect to the arbitrary input distribution. The capacity of a channel is basically the rate at which error free communication is possible over that channel.

The noisy channel theorem[19] main point was to prove that we can get arbitrarily low error rates close to a channel capacity (see figure 2). Our goal and use of the noisy channel theorem is all about keeping low error probability rates without getting beyond the channel capacity, we should try too keep our values in the blue area, as close as possible to 0. This means that we will have low error in between the channel capacity.



C= Capacity

R= rate of transmission

N= number of times you use the channel

Figure 2
Author: Hand drawn by me

The proof of the Noisy channel coding theorem goes beyond this paper goal we can hence summarise the relevant part as follows: For the BSC (later explained) with flip probability f whose capacity is $C = 1 - H_2(f)$[20] for any $\varepsilon > 0$ and $R < C$ for large enough N (number of times you use the channel)

[19] Slot, Lucas, and Sebastian Zur. Shannon's Noisy-Channel Coding Theorem. 2015.

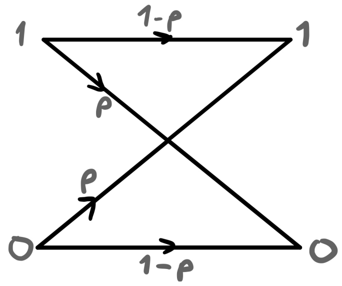
[20] "Lecture 8: Noisy Channel Coding (III): The Noisy-Channel Coding Theorem." *Www.youtube.com*, www.youtube.com/watch?v=KSV8KnF38bs&t=1061s. Accessed 4 Nov. 2022.

Exists a code of length N and a rate of $\geq R$ and a decoder such that the probability of block error is $< \varepsilon$ which in simpler terms states that exist a code that can lay on the blue area which has low error probabilities.[21]

Essentially the channel coding theorem defines a channel capacity at which we can produce a certain probability of noise, it is essential for mathematicians and scientist to identify the channel capacity of their satellite in order to correctly develop methods which detect and correct errors.

Binary Symmetric Channel

Figure 3



Author: Hand drawn by me

Table 1-1

y= output	$P(y=0 \mid x=0) = P(y=1 \mid x=1) = 1-p$
x= input	$P(y=0 \mid x=1) = P(y=1 \mid x=0) = p$

The Binary symmetric channel (BSC),[22] is the most common theoretical communication channel in information theory, we will mainly use it to determine a method that will find the probability of success of sending a 0 and receiving a 0 and vice versa.

[21] "Lecture 8: Noisy Channel Coding (III): The Noisy-Channel Coding Theorem." *Www.youtube.com*, www.youtube.com/watch?v=KSV8KnF38bs&t=1061s. Accessed 4 Nov. 2022.

[22] "Binary Symmetric Channel - an Overview | ScienceDirect Topics." *Www.sciencedirect.com*, www.sciencedirect.com/topics/mathematics/binary-symmetric-channel.

It is fundamental to understand how error probabilities work before diving deeper in the functioning of the algorithm itself as determining the probability will lead to an understanding of how much data could be possibly in error and what percent of it could be corrected (referring to the satellite system). Understanding how much error could occur in deep space communication could reveal to be fundamental for the project itself. [23]

In the previous table (Table 1) using simple probability functions we have determined the probabilities of flipping or not flipping a bit. Our goal here is to develop a method to find probabilities of error and success that will later be applied to our system.

We can better understand the probability of error with an example.

If we want to determine the probability of error in our system we must use some sort of binomial distribution[24]. Let's say I'm sending 10000 files of data to space with a probability of flip of 0.1 using the BSC we can deduce how many files will be found in error

$p = 0.1$ Data

N= 10000

We therefore will use a binomial distribution[25] as we only have a fail or success. In the distribution we know by definition that:

$mean = np$ Mean is the number of trials times the probability of error

$variance = np(1-p)$ Variance= mean times probability of success

$\sigma = \sqrt{vairance}$ Standard deviation

$$mean = 10000 \times 0.1 = 1000$$

$$variance = 10000 \times 0.1 \times (1 - 0.1) = 900$$

[23] Dizon, Reiner. Efficient Image Coding and Transmission in Deep Space Communication Communication. 2018.

[24] "1.3.6.6.18. Binomial Distribution." *Www.itl.nist.gov*, www.itl.nist.gov/div898/handbook/eda/section3/eda366i.htm.

[25]

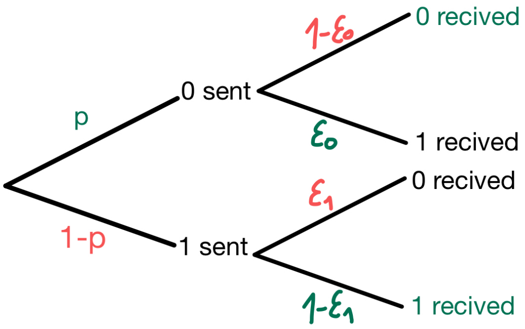
$$\sigma = \sqrt{900} = 30$$

Therefore we will have 1000 ± 30 bits flipped

This method is a simple way to calculate the probability of error in a BSC, it might hence be useful to calculate probabilities of errors in simple systems.

The BSC is at the base of the world communication system and could be applied to our model to create an effective system of interconnection where we can calculate probabilities and understand their relevance. In spatial communications or in general relevant messaging, it's of fundamental importance to know the error probability in a system in order to be prepared to receive an x amount of error and understand if the algorithm will be able to correct it.[26]

Figure 4



Author: Hand drawn by me

Looking at figure 4

$$0 \text{ received} = \text{success}$$

$$1 \text{ received} = \text{success}$$

$$P(success) = p(1-\varepsilon_0) + \varepsilon_1(1-p)$$

$$\therefore$$

$$p(0) \times p(succ\,|\,0) + p(1) \times p(succ\,|\,1)$$

[26] Nikolov, Dimitar, et al. Estimating Error-Probability and Its Application for Optimizing Roll-Back Recovery with Checkpointing. 2010.

Therefore, to calculate the probability of success of any given binary message we can use our method to break down the different probabilities. Let's take as an example the message 1011 where we assume that each transmission bit has the same probabilistic function. All the bits error are independent between each other.

$P(1011 \rightarrow 1011)$	Probability that we send 1011 and we receive 1011
$P(1 \rightarrow 1 \cap 0 \rightarrow 0 \cap 1 \rightarrow 1 \cap 1 \rightarrow 1)$	Probability of transmission of each bit in order of the message
$P(1 \rightarrow 1)P(0 \rightarrow 0)P(1 \rightarrow 1)P(1 \rightarrow 1)$	The events are independent and we can simplify the function in a way that we know each probability
$(1-\varepsilon_0)(1-\varepsilon_1)^3$	Using our diagram we can deduce the probabilities for each bit

We have hence deduced a method which solves the probability of success of any binary message and could be used in our channel to detect probabilities of errors according to the channel capacity. Understanding the channel which in our case will just be the environment in which data is transferred is fundamental not only to understand how errors might occur but also to develop efficient and directed algorithms that aim specific types of errors.

Prerequisites for Correction codes

Basic Binary

To completely understand how spatial error detection works we first need to take a step back and be able to understand the binary language[27].

If we think of communication in the world today, we might conclude that there is not a true common language that every single individual can speak and understand. Even if English is the most used method to communicate between cultures it's still not spoken and understood in several countries.

Computers though have their own language which is common for every model and company which produced it, and unlike humans they are able to communicate with only yes or no's or as we better know them with 0's and 1's. Furthermore they are not only able to "talk" to each other but also create images, calculate on excel, show movies, messages, and replay audio, this list could continue forever as many as the features and functions of a computer can be.

Binary is a number system that only uses base 2 number hence being only 2 digits (0,1), it might hence be simple and intuitive to represent 0 as 000 or 1 as 001[28]. While expressing greater numbers we have to move our final 1 from right to left. For example if we have to represent 2 in binary we will write it as 010 while 3 will be shown as 011 and 4 will be written as 100 and so on.

[27] *Study.com*, 2022, study.com/learn/lesson/binary-language-explained.html#:~:text=Binary%20code%20is%20a%20system. Accessed 15 Sept. 2022.

[28] "Binary Explained in 01100100 Seconds." *Www.youtube.com*, www.youtube.com/watch?v=zDNaUi2cjv4. Accessed 15 Sept. 2022.

The main difference between binary and our numerical system is that, the last uses base 10 numbers, while binary base 2 numbers making calculations only between 1s and 0s.

Taking 234 as an example

2	3	4
100	10	1
10^2	10^1	10^0

From this reasoning we understand that

$$2(10^2) + 3(10^1) + 4(10^0) = 234$$

In binary instead we change the value of the columns, in a 3 digit number the column furthest to the right will be 2^0, the second column will be 2^1 while the third will be 2^0 and so on.[29]

0	0	0	0
8	4	2	1
2^3	2^2	2^1	2^0

Redundancy in mathematical terms

Error correction works under the concept of redundancy, even if deep space error correction algorithms use sophisticated types of redundancy that we will later analyse , I thought that understanding the concept at a basic level could reveal to be extremely useful for following explanations.

[29] freeCodeCamp.org. "Binary Definition." *FreeCodeCamp.org*, FreeCodeCamp.org, 26 Apr. 2021, https://www.freecodecamp.org/news/binary-definition/#:~:text=In%20computer%20science%20and%20mathematics,based%20on%20of%20Boolean%20Algebra.

Computer redundancy is usually described as the repetition of a part or the whole data[30]. This leads to our first and most basic method to add redundancy in our system which is by using repetition coding. There are different methods of parity coding but for the scope of this paper we will cover only the most relevant:

Simple repetition code: The most intuitive method in which we could add redundancy to a message will simply be by repeating the whole message a second time. For example if the source will want to emit 0101, the BSC could repeat 0101 another time

0101**0101**

This method is being named as the simple repetition code[31]. This technique is only able to detect errors and correct only single bit errors.

Triplar modular redundancy: The second way we could add redundancy to a message sent by the source would be just implementing the simple repetition code for another time, hence repeating the message for three times. By taking the same example as before, if the source is willing to send 0101 the triple modular redundancy will add 0101 for 2 other times, creating a message that will just look like this:

0101**01010101**

The main improvement is that we can both detect and correct single and double errors in the message as there are three parts which can be compared. In this case we have 2:3 information to redundancy when we usually only want a ratio of at most 1:2[32] therefore increasing the information over our channel capacity.

[30] "What Is Redundancy? - Definition from WhatIs.com." *WhatIs.com*, www.techtarget.com/whatis/definition/redundancy#:~:text=Redundancy%20is%20a%20system%20design.

[31] "Repetition Code - an Overview | ScienceDirect Topics." *Www.sciencedirect.com*, www.sciencedirect.com/topics/engineering/repetition-code. Accessed 15 Sept. 2022.

[32] https://www.researchgate.net/figure/Redundancy-ratio-and-number-of-redundant-records-a-refers-to-redundancy-ratio-of_fig4_322779627

Repetition coding, even if useful in basic algorithms is way too simple for detecting and correcting errors in spacial communications, which is what we need for our investigation. We will hence introduce the concept of hamming codes and how they relate with the noisy channel coding theorem to create a error correction algorithm for more sophisticated systems.

Hamming codes

Ideation

The hamming codes[33] are at the base of the functioning of every satellite system or general sophisticated technology which requires some type of error detection and correction[34]. It is helpful to understand their functioning that will help us develop more sophisticated systems.

Just after the publication of the noisy channel coding theorem[35] in 1948 by Claude Shannon which proved that "efficient codes" could possibly exist, many researchers started searching for self correcting coding algorithms[36] which could both detect and correct errors.

Richard Hamming[37] which at that time worked at Bell Laboratories, shortly after the publication of the theory, started developing a system that could both detect and correct errors following the noisy channel coding theorem after thinking in improving the current algorithms that could only detect the error.

Hamming (7,4 code)

The previous examples of error correcting codes worked on the concept of repetition which provided a ratio of true data over parity data of at least 1:3 where 1 was the message data and 3 was message data + redundancy. The Hamming (7,4)[38] takes a message of 4 bits and forms a 7 codeword as we include some type of parity, meaning that we will get an overall efficiency of 4/7 = 57.1 % as we must divide the message bits by the total number of bits sent.

Let's say we have a binary message to transmit of:

[33] "What Is Hamming Code and How Does It Work?" *WhatIs.com*, www.techtarget.com/whatis/definition/Hamming-code.

[34] Dizon, Reiner. Efficient Image Coding and Transmission in Deep Space Efficient Image Coding and Transmission in Deep Space Communication Communication. 2018.

[35] Slot, Lucas, and Sebastian Zur. Shannon's Noisy-Channel Coding Theorem. 2015.

[36] "What Is Self Correcting Code?" *Quora*, www.quora.com/What-is-self-correcting-code. Accessed 16 Sept. 2022.

[37] "Computer Pioneers - Richard Wesley Hamming." *History.computer.org*, history.computer.org/pioneers/hamming.html.

[38] Nikhil, Nishant. "Introduction to Information Theory -Hamming(7,4) Code." *Medium*, 3 June 2017, becominghuman.ai/introduction-to-information-theory-hamming-7-4-code-b341bd01d955. Accessed 25 May 2021.

10010000110011100100111000110010

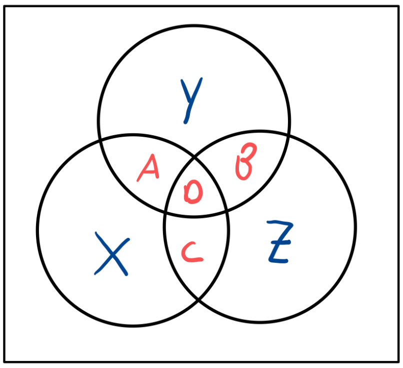
The first step is to divide it in 4 bit chunks as follows to create the 4 message datas

1001; 0000; 1100; 1110; 0100; 1110; 0011; 0010

Now having an indefinite amount of groups of 4 bits we must start to add parity bits to our message. For consistency let's name each data bit as $ABCD$, and each group of parity bits xyz. For clarity our first data bit block: 1000 will be labeled as $A_1B_1C_1D_1$. Our goal is to calculate xyz parities related to each block.

To fully understand the concept it will be useful to visualise the code with a Venn diagram

Figure 4



Author: Hand drawn by me

Looking at the diagram we can notice that our parity position y has the associated data bits ABD, we hence will find the value of the y parity bit using the "exclusive or" function as our y parity bit serves as a even parity for the positions ABD. The Xor (Exclusive or)[39] is the same concept as finding the even parity bit and will give the same result as will later be represented

The Even Parity bit, will count all the 1s in the data.

- If the number is odd the parity bit is set to 1 making the total numbers (including parity) of 1s even

[39] "The Power of XOR - Gary Explains." *Www.youtube.com*, www.youtube.com/watch?v=3Kvv7VEM3uc&t=156s. Accessed 14 Nov. 2022.

- If the number is even the parity bit is set to 0 keeping the number of 1s even

Xor calculations can be represented as simple addition with a slight variation.

$$0 \oplus 0 = 0$$

$$0 \oplus 1 = 1$$

$$1 \oplus 0 = 1$$

$$1 \oplus 1 = 0$$

In our example we can notice that the first 3 operations give the same result of a simple addition but the last one, which addition should give 2 as answer with our Xor[40] it results in a 0. This is because Xor operations work under the concept of even or odd.

The Xor operations can be seen as a simple addition where if the answer is even we get a 0 as a result while if the answer is odd we will receive a 1.

Therefore the Y parity bit can be written as follows by referring back to our diagram:

$$Y = A \oplus B \oplus D$$

Which in our example

$$A_1B_1C_1D_1 = 1001$$

$$\therefore$$

$$Y = 1 \oplus 0 \oplus 1 = 1$$

As the answer of 1 +0+1 is even we receive a 0

For our following parity bits:

$$X = A \oplus D \oplus C$$

$$X = 1 \oplus 0 \oplus 1 = 0$$

[40] "Xor Function - Google Search." *Www.google.com*, www.google.com/search?client=safari&rls=en&q=xor+function&ie=UTF-8&oe=UTF-8. Accessed 7 Nov. 2022.

As the answer of 1 +0+1 is even we receive a 0

Finally

$$Z = B \oplus D \oplus C$$

$$Z = 0 \oplus 1 \oplus 0 = 1$$

As the answer of 0 +1+0 is odd we receive a 1

$$\therefore$$

$$x = 0; y = 0; z = 1$$

Our first block message will be

$$A_1B_1C_1D_1xyz = 1001001$$

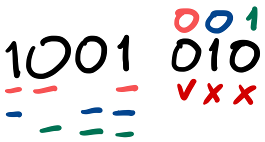
As we have developed a technique to create a 7,4 Hamming code, lets take an example which could show us how to detect error using this algorithm.

A signal is sent from space to earth, the receiver gets the binary data 1001 010 where the first chunk is the data while the final part is the parity. From our previous working we know

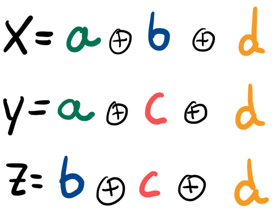
$$X = A \oplus D \oplus C$$

$$Y = A \oplus B \oplus D$$

$$Z = B \oplus D \oplus C$$



We can notice that our yz parity bits are incorrect.



We know that Y and Z equations are incorrect, by looking at those two we notice they have a C and D bit in common. Meaning either C or D is incorrect. We can notice that the X equation also contains the D bit. For this reason we deduce the C bit is the incorrect one and we therefore flip it.

As shown with the previous example the hamming code 7,4 enables us to solve multiple bit errors using simple calculations. This can reveal to be extremely useful for systems that require low computational capacity[41] but still need some sort of error correction codes. The efficiency of error detection is also acceptable as our codes doesn't require excessive channel capacity.

For our specific problem (deep space communication), massive data are usually sent through long distances with relatively high noise (background radiation[42]). This means that hamming codes are not sufficiently effective in correcting data in space. For this reason Nasa developed a system of error correction codes named as Cyclic Reed Solomon codes[43] which were used in the Voyager satellite programme[44] back in 1977. Understanding the functioning of these codes is the goal of the paper as they will solve the problem described in our research question.

[41] Williams, Lawrence. "Hamming Code: Error Detection and Correction with Examples." *Www.guru99.com*, 2020, www.guru99.com/hamming-code-error-correction-example.html.

[42] "Cosmic Microwave Background (CMB) Radiation." *Www.esa.int*, www.esa.int/Science_Exploration/Space_Science/Herschel/Cosmic_Microwave_Background_CMB_radiation.

[43] https://www.cs.tau.ac.il/~amnon/Classes/2017-ECC/Lectures/Lecture2.pdf

[44] "Voyager - Mission Overview." *Nasa.gov*, 2012, voyager.jpl.nasa.gov/mission/.

Cyclic Codes

We have previously discussed simple Error Correction codes such as the Hamming codes. This algorithm calculated a 3 bit parity set to a 4 bit message data creating a 7 bit codeword, we have hence named it Hamming (7,4) as 4 is the message data while 7 is the whole codeword including parity.

We have understood the functioning of this algorithm using simple equations and Exclusive or functions but its important to know that this is not the only way to calculate the parity bits in the Hamming code. The matrix and polynomial check are 2 other methods used in error correction codes to visualise and develop a solution,[45] yet we have used the simple version of these 3 methods as hamming codes can be easily explained and developed with the Xor function. This method has its limitation when we want to develop more complicated algorithms to be able to solve errors with a higher degree of accuracy, for example in our case of deep space communication. For this reason the Reed Solomon code[46] is used with polynomials to detect its parity bits. The Nasa decided to use polynomial error correction codes for its deep space communication error correction mainly for its high efficiency and reliability[47].

[45] "How to Send a Self-Correcting Message (Hamming Codes)." *Www.youtube.com*, www.youtube.com/watch?v=X8jsijhllIA&t=679s. Accessed 14 Nov. 2022.

[46] Ta-Shma, Amnon, and Dean Doron. Polynomial Codes and Cyclic Codes.

[47] Geisel, William A. "Tutorial on Reed-Solomon Error Correction Coding." *Ntrs.nasa.gov*, 1 Aug. 1990, ntrs.nasa.gov/citations/19900019023. Accessed 14 Nov. 2022.

Modular Arithmetic

Before diving deep into the functioning of the algorithm itself we must understand an important mathematical concept behind the algorithm itself: modular arithmetic[48]. Modular arithmetic finds its roots on congruencies and division, where in simple terms we can say that two numbers are congruent to each other if, when divided by a number p obtain the same remainder

$$a \equiv b \qquad modp$$

When we say that a is congruent to b with mod p we basically mean that a and b have the same reminder when divided by p. Let's take an example

$$10 \equiv 14(mod4)$$

$$10/4 = 2(r\ = 2)$$

$$14/4 = r(r\ = 2)$$

Therefore when we divide 10 by 4 we get as result 2 with a remainder of 2, if we divide 14 by 2 we will get 3 with a reminder of 2. We can hence say that 10 is congruent to 14 mod 4.

This will reveal to be useful while developing Lagrange interpolations which are a fundamental aspect for the Reed Solomon cyclic code.

Lagrange interpolation

As we previously discussed Reed Solomon codes work under the concept of polynomials, our goal is to create a polynomial function for the code we want to correct. The Lagrange Interpolation methods helps us to find a polynomial function given a set of specific points. It gives a polynomial n-1 with n number of coordinates.

[48] "What Is Modular Arithmetic - Introduction to Modular Arithmetic - Cryptography - Lesson 2." *Www.youtube.com*, www.youtube.com/watch?v=Eg6CTCu8iio&t=129s. Accessed 14 Nov. 2022.

Let's consider 3 points as our example: $(1,2);(3,2);(4,-1)$ our goal is to find a two degree polynomial (quadratic) equation that passes through these points.

The Lagrange interpolation first considers n different polynomials where n is the number of points given. The first polynomial $l_1(x)$ will have a value of y=1 at the first data point while a value of y=0 at any other point, for instance in our example the first polynomial is equal to 1 at x=1 but equals to 0 at x=3 and x=4. Secondly, the second polynomial $l_2(x)$ will have a value of y=1 at the second data point while y=0 at every other. In our example it equals 0 at x=1, equals 1 at x=3 and equals 0 again at x=4. For last the third polynomial $l_3(x)$ will be the same thing as the previous equation only that y=1 at the last data point.

Our first step will be to work out the specific functions for these specific equations, it's fundamental to consider that we are also using modular arithmetic At this point we can construct a function for each of the polynomials using our knowledge on quadratic functions.

$$l_1(x) = (x-3)(x-4)$$

$$l_1(1) = (1-3)(1-4) = 6$$

$$l_1(x) = 1/6(x-3)(x-4)$$

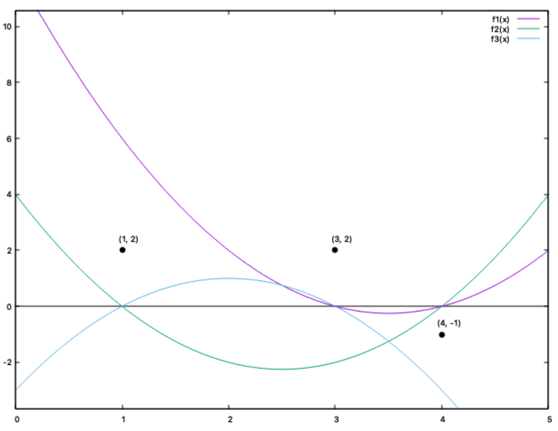
We know the intersection points on the x-axis and can deduce the following equations using the intercept form. To complete our equation we must satisfy that the first function must be y=1 at x=1. At x=1 we get 6, we hence divide by 6 to match the first point to y=1.

$$l_2(x) = (x-1)(x-4)$$

$$l_3(x) = (x-1)(x-3)$$

The same procedure should be done for all other equations

Figure 5



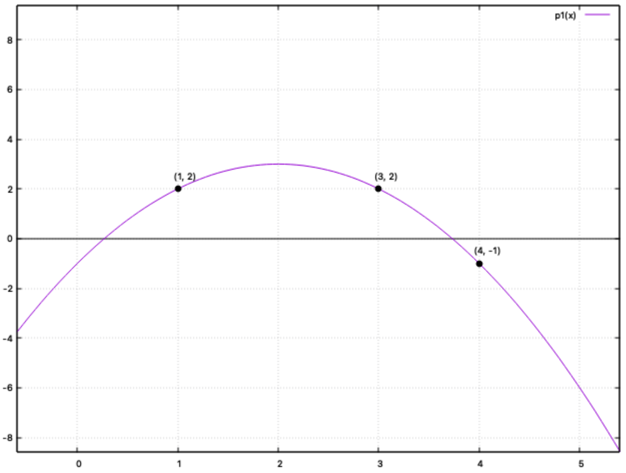
Author: Personally plotted using GnuPlot

To get our final polynomial we should multiply each Lagrange polynomial for it's y value and add them all together.

$$p(x) = (2 \times l_1(x)) + (2 \times l_2(x)) + (-1 \times l_3(x))$$
$$(2 \times 1/6(x-3)(x-4) + (2 \times -(x-1)(x-4)) + ((x-1)(x-3)) = -x^2 + 4x + 1$$

$$p(x) = -x^2 + 4x - 1$$

Figure 6



Author: Personally plotted using Gnuplot

We have therefore deduced a polynomial equation that passed through the points we had previously mentioned. As seen in the working the Lagrange equation provides us a simple method to calculate polynomial functions for given data points, this will be useful in the Reed Solomon codes as they will create a path for the sent and parity data, as will be described in the following section. Lagrange interpolation is hence a fundamental concept for the development of deep space communication algorithms.

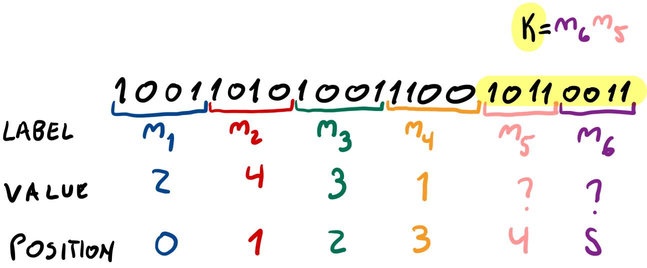
Reed-Solomon Codes

Having some knowledge on modular arithmetic and Lagrange interpolation we can finally develop an advanced algorithm to detect errors for deep space communication devices.

The system will divide our data in "packs" which represent sets of binary data, at this point we want to associate some type of value and label to the pack in order to recall it's data. This is done using complicated computational systems that go beyond the scope of the paper, we will hence associate some random values to the data packs. We will then determine a k value which will be based on error probabilities to find the maximum data packs that could be lost in our transfer.

These steps are shown in figure 7.

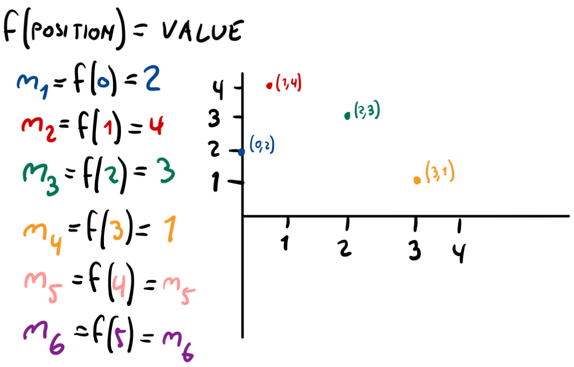
Figure 7



Author: Hand drawn by me

We arrived to the point in which we have created 6 data packs which have been labeled, assigned a value and a position number. Reed Solomon codes work under the concept of developing a polynomial equation that represents our data, hence if a specific pack will be lost during transmission it's data could be recovered by analysing its position on the graph and reading its value on the x axis. This could be better understood if we represent our example with functions and points that can be plotted on a graph, as shown in Figure 8

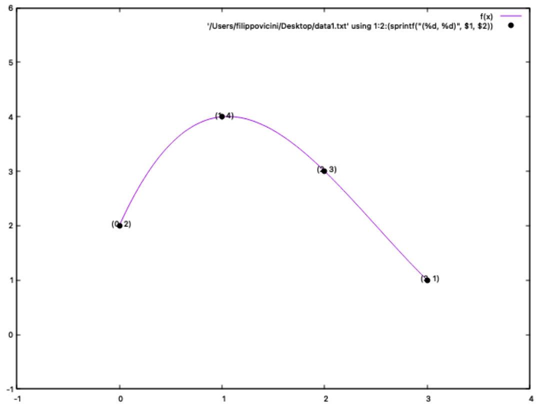
Figure 8



Author: Hand drawn by me

Our goal is to therefore create a polynomial function using these points, for this reason the Lagrange interpolation is being used in reed Solomon codes as it creates a polynomial function that passes through each of pack points. The equation representing our sample points is shown in figure 9

Figure 9



Author: Plotted by me using GnuPlot

Reed Solomon codes functions gets exponentially more complicated, yet the goal of this paper has been to develop an understanding of basic error correction codes to later extend our knowledge on more complicated systems specifically used to detect deep space communication errors.

Conclusion

The aim of this paper has been to develop an understanding on Error correction algorithms. The guiding line of my whole research was my investigation question which, to be answered, required knowledge in different aspects of information theory. I hence tried to develop a paper which works as a sort of staircase where the first steps are the foundation blocks to later fully understand how to correct errors in deep space communication.

I was aware of the fact that these types of codes get extremely complicated when discussing each single component and function, but my goal was to get a general overview of how these types of codes function. Basically each explanation until the Cyclic Code section is made to create a background knowledge on ECC's so that a potential reader will be able to fully understand the Reed Solomon codes.

Therefore at the question: To what extent is error detection helpful when sending data from a satellite to the earth?

I can now clearly state that much of the progress we have done in exploring space would have not been possible without ECC's, Reed Solomon codes, the most common spacial error detection system is an essential aspect of our everyday technological life. As it ensures clear communication between our devices and the satellites above our heads.

Even if in the near future this won't be helpful in any kind of way I am happy to state that I had fun writing most of this paper, I never would have thought of how interesting a simple and ignored topic could become. Even if the new things learned won't be useful by itself I'm sure that the the struggles of writing such a paper in concise and comprehensible English will help me in the near future.

Bibliography

Pamuditha, Isuru. “Hamming Code Generation & Correction (with Explanations Using c Codes).” *Medium*, 12 May 2020, medium.com/swlh/hamming-code-generation-correction-with-explanations-using-c-codes-38e700493280.

“How to Send a Self-Correcting Message (Hamming Codes).” *Www.youtube.com*, www.youtube.com/watch?v=X8jsijhllIA&t=421s. Accessed 19 Sept. 2022.

“Hamming Code in Computer Network - GeeksforGeeks.” *GeeksforGeeks*, 26 Dec. 2017, www.geeksforgeeks.org/hamming-code-in-computer-network/.

“5. Conditions That Affect CDs and DVDs • CLIR.” CLIR, www.clir.org/pubs/reports/pub121/sec5/.

“COS 126: Assignments (Fall 2019) - Hamming Codes.” Www.cs.princeton.edu, www.cs.princeton.edu/courses/archive/fall19/cos126/assignments/hamming/index.html. Accessed 19 Sept. 2022.

“Hamming Code in Computer Network.” GeeksforGeeks, 26 Dec. 2017, www.geeksforgeeks.org/hamming-code-in-computer-network/?ref=lbp. Accessed 19 Sept. 2022.

GeeksforGeeks. “Error Detection in Computer Networks.” GeeksforGeeks, 9 Jan. 2016, www.geeksforgeeks.org/error-detection-in-computer-networks/.

“Hamming Code Implementation in C/C++.” GeeksforGeeks, 25 Oct. 2020, geeksforgeeks.org/hamming-code-implementation-in-c-cpp/?ref=rp. Accessed 19 Sept. 2022.

“Hamming Code.” Wikipedia, 15 Sept. 2022, en.wikipedia.org/wiki/Hamming_code#General_algorithm. Accessed 19 Sept. 2022.

“Coding Theory.” Wikipedia, 12 Feb. 2022, en.wikipedia.org/wiki/Coding_theory.

“Cyclic Redundancy Check.” Wikipedia, 11 Mar. 2021, en.wikipedia.org/wiki/Cyclic_redundancy_check#:~:text=A%20cyclic%20redundancy%20check%20(CRC.

“What Is Algorithm for Computing the CRC?” Www.tutorialspoint.com, www.tutorialspoint.com/what-is-algorithm-for-computing-the-crc#:~:text=Algorithm%20for%20Encoding%20using%20CRC&text=The%20block%20xrM. Accessed 19 Sept. 2022.

“Error Detection and Correction.” Wikipedia, 25 July 2022, en.wikipedia.org/wiki/Error_detection_and_correction#Types_of_error_correction. Accessed 19 Sept. 2022.

“Week 0 - CS50.” Cs50.Harvard.edu, cs50.harvard.edu/college/2022/spring/weeks/0/. Accessed 19 Sept. 2022.

“Binary Symmetric Channel - an Overview | ScienceDirect Topics.” Www.sciencedirect.com, www.sciencedirect.com/topics/mathematics/binary-symmetric-channel. Accessed 19 Sept. 2022.

“What Is Bit (Binary Digit) in Computing?” WhatIs.com, www.techtarget.com/whatis/definition/bit-binary-digit#:~:text=A%20bit%20(binary%20digit)%20is.

“What Is Bit (Binary Digit) in Computing?” WhatIs.com, www.techtarget.com/whatis/definition/bit-binary-digit#:~:text=A%20bit%20(binary%20digit)%20is.

“Definition of Redundancy - Gartner Information Technology Glossary.” Gartner, www.gartner.com/en/information-technology/glossary/redundancy. Accessed 19 Sept. 2022.

“Definition of Redundancy - Gartner Information Technology Glossary.” Gartner, www.gartner.com/en/information-technology/glossary/redundancy. Accessed 19 Sept. 2022.

“Learn Binary Code: 5-Bit Binary Code Challenge | Alphabet Code, Binary Code, Writing Code.” Pinterest, www.pinterest.it/pin/68750331799380379/. Accessed 19 Sept. 2022.

Krištofík, Štefan, and Elena Gramatová. “Redundancy Algorithm for Embedded Memories with Block-Based Architecture.” IEEE Xplore, 1 Apr. 2013, ieeexplore.ieee.org/document/6549832. Accessed 4 Nov. 2022.

“Lecture 1: Introduction to Information Theory.” Www.youtube.com, www.youtube.com/watch?v=BCiZc0n6COY&list=PLruBu5BI5n4aFpG32iMbdWoRVAA-Vcso6&index=1. Accessed 4 Nov. 2022.

“What Is Modular Arithmetic - Introduction to Modular Arithmetic - Cryptography - Lesson 2.” Www.youtube.com, www.youtube.com/watch?v=Eg6CTCu8iio&t=129s. Accessed 14 Nov. 2022.
Ta-Shma, Amnon, and Dean Doron. Polynomial Codes and Cyclic Codes.

Geisel, William A. “Tutorial on Reed-Solomon Error Correction Coding.” Ntrs.nasa.gov, 1 Aug. 1990, ntrs.nasa.gov/citations/19900019023. Accessed 14 Nov. 2022.

“Cosmic Microwave Background (CMB) Radiation.” Www.esa.int, www.esa.int/Science_Exploration/Space_Science/Herschel/Cosmic_Microwave_Background_CMB_radiation.

Ta-Shma, Amnon, and Dean Doron. Polynomial Codes and Cyclic Codes.

“The Power of XOR - Gary Explains.” Www.youtube.com, www.youtube.com/watch?v=3Kvv7VEM3uc&t=156s. Accessed 14 Nov. 2022.

Dizon, Reiner. Efficient Image Coding and Transmission in Deep Space Efficient Image Coding and Transmission in Deep Space Communication Communication. 2018.

Slot, Lucas, and Sebastian Zur. Shannon’s Noisy-Channel Coding Theorem. 2015.

freeCodeCamp.org. “Binary Definition.” FreeCodeCamp.org, FreeCodeCamp.org, 26 Apr. 2021, https://www.freecodecamp.org/news/binary-definition/

“Binary Explained in 01100100 Seconds.” Www.youtube.com, www.youtube.com/watch?v=zDNaUi2cjv4. Accessed 15 Sept. 2022.

“1.3.6.6.18. Binomial Distribution.” Www.itl.nist.gov, www.itl.nist.gov/div898/handbook/eda/section3/eda366i.htm.

“How to Send a Self-Correcting Message (Hamming Codes).” Www.youtube.com, www.youtube.com/watch?v=X8jsijhllIA&t=679s. Accessed 14 Nov. 2022.