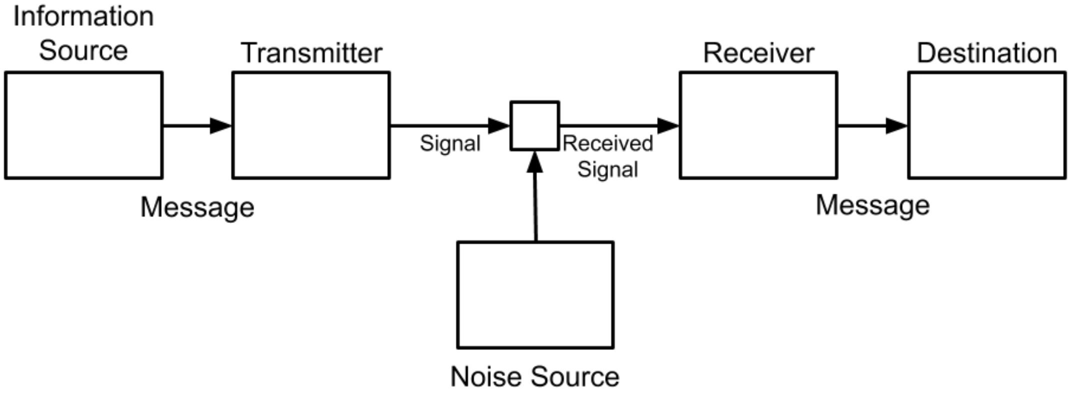
ERROR CORRECTING CODES

ERIK FELVINCZI
EULER CIRCLE

1. Introduction

All information processed in systems is encoded in symbols to which meanings and values are assigned. A mathematical model of information processing was formulated by Claude Shannon in 1948, describing how a sender communicates over a channel with a receiver. This model describes discrete, continuous, and mixed communication, and for all of them it is essential, that the message is reproduced at the receiver, even when perturbation happens due to noise in the transmission channel. Shannon showed that information can be transmitted with arbitrarily low rate of error at the receiver.



For discrete communication, where messages are selected from a finite set, the introduction of error-correcting codes by Richard Hamming in 1950 addressed the errors caused by a noise source in the channel, leading to more reliable systems. Error-correcting codes are being used especially in data storage, to avoid data corruption.

2. Mathematical Models of Discrete Communication

If we follow a message on its path from the source to the receiver, the transmition of information is a stochastic process, as there is a sequence of discrete symbols chosen from a finite set at the source, with a transmission governed by the probabilities of errors in the channel. This is a discrete Markov chain.

Definition 2.1. A *discrete Markov chain* is a discrete stochastic process with the following properties:

- There exist a finite number of possible states of a system, $S_1, S_2, \ldots, S_n$.
- There is a set of transition probabilities $p_i(j)$ as the probability of state S_i transitioning to S_j.

Suppose a set of events with probabilities of occurrence $p_1, p_2, \ldots, p_n$. The uncertainty of the outcome can be represented with a function H with the following properties:

- H is continuous in the p_i.
- If all p_i are equal, then $p_i = \frac{1}{n}$, and H is a monotonic increasing function of n.

This shows that with equally likely events there are more uncertainties when there are more events, or an increase in the probability of errors as the string length of the transmitted information is increasing.

A message m of a finite set $\mathcal{M}$, called the message space, can be encoded into binary for transmission with symbols from a finite set $\Sigma = \{0, 1\}$. If n is the length of the transmitted sequence, then Σ^n is the space of transmitted words over the channel, called ambient space. Thus the encoding function E is an injective map from the message space to the ambient space, $E : \mathcal{M} \longrightarrow \Sigma^n$. When the sender communicates a message m, the image of the encoding function $\{E(m) | m \in \mathcal{M}\}$ is the error correcting code. We define a group structure to the ambient space Σ with the operator $+$ such that $\langle a_1, \ldots, a_n\rangle + \langle b_1, \ldots, b_n\rangle = \langle a_1 + b_1, \ldots, a_n + b_n\rangle$. If an error $\eta \in \Sigma^n$ is produced in the channel, the received signal becomes $y = E(m) + \eta$, and the receiver has to use a decoding function $D : \Sigma^n \longrightarrow \mathcal{M}$ to recover the message $m = D(y)$.

Question 2.2. *What are the best encoding and decoding functions if the channel inserts errors $\mathcal{P} : \Sigma^n \longrightarrow [0, 1]$?*

$$\max_{E,D}\{E_{m\in\mathcal{M}}[P_{\eta\in\mathcal{P}}[D(E(m) + \eta) = \mathbf{m}]]\}.$$

3. Single Error Detecting Code: Parity Check

Hamming defines systematic codes as those having n binary digits, with m digits for the message, and k digits for error detection and correction: $n = k + m$. These systematic codes have a redundancy R as the ratio of the number of digits used, to the minimum number necessary to transmit the message: $R = \frac{n}{m}$. A single error detecting code, also called parity check, has n binary digits with the first $n - 1$ being used for the message, while in the n-th position we place a 0 or a 1 so that n will have an even number of 1's. This is single error detecting, since any single error leaves an odd number of 1's.

Example. The decimal number 6 will be encoded as 110 plus an additional 0, as there are two 1's. The receiver will be able to determine whether or not the message contains errors, which for 1100 is not the case. If however, the second bit has an error, and 1000 is transmitted, it is evident that there is an odd number of 1's, and thus the message must contain an error.

The presence of errors passes unnoticed however, for an even number of errors, as the change in the number of 1's cancels out, meaning the n-th bit is still valid. It may also occur that there is an error in the n-th bit, in which case the correct message is transferred, but interpreted to be wrong by the decoder.

For parity check encoding the redundancy is

$$R = \frac{n}{m} \text{ with } m = \frac{n}{n-1} = 1 + \frac{1}{n-1}.$$

In order to gain low redundancy, n should increase. This, however, as shown for monotonic increasing functions, increases the probability of errors, which may lead to double errors that will pass undetected. This parity check could only be used for odd parity, while it is noticeable that an even number of errors is not detected. As $n \to \infty$, $R \to 1$, but the probability of errors also increases.

4. Single Error Correcting Code

4.1. **(3,1) Repetition codes.** Let m be any arbitrary four bits string in the form a, b, c, d $\in$ {1,0}. Assume that during transmission there is a probability of a single error occuring. We could triple every bit and transmit *aaabbbcccddd*.

For example, instead of $n = 1011$, we will send $n = 111000111111$. If, say, 110000111111 is received, than the third bit has error and the corrected string is 111000111111. In this case $n = 12$ with m = 4 and the redundancy is R = n/m = 3 which is very high and not desirable.

4.2. **The Hamming Code.** Assuming a string of four bits ($m = 4$), we transmit a code $n = 7$ such that $m = abcd$ and $n = abcdefg$ with

$$\begin{aligned} e &:= a + b + c \\ f &:= a + b + d \\ g &:= a + c + d, \end{aligned}$$

where := refers to the residual class of the sum of three bits, modulo 2. For example for $m = 1011$, $n = 1011001$. With linear algebra it will be proved that the above code makes it possible to correct any single-bit error.

Definition 4.1. Σ is a finite set called the alphabet, such that

$$\Sigma^n = \{w = a_1, \ldots, a_n : a_1, \ldots, a_n \in \Sigma\}$$

with a word w being an arbitrary finite sequence of letters in the alphabet.

Definition 4.2. A code of length n over an alphabet Σ is an arbitrary subset $C \subset \Sigma^n$

For example, for the Hamming code, $\Sigma = \{0, 1\}$ and $n = 7$, where C is the set of all 7-bit words that can be generated from all 4-bit words using the method described above, $|C| = 2^4 = 16$:
$C = \{0000000, 0001011, 0010101, 0011110, 0101101, 0110011, 0111000, 1001100, 1010010, 1011001, 1100001, 1101010, 1110100, 1111111\}$. This code has the property that every two of its words differ in at least three bits.

Definition 4.3. The Hamming Distance of two words $u, v \in \Sigma^n$ is

$$d(u, v) := |\{i : u_i \neq v_i, i = 1, 2, \ldots, n\}|$$

where u_i is the ith letter of the word u.

For example, $d(1011, 1101) = 2$ as there are two bits in v which are not equal to the corresponding bits in u.

A code C corrects t errors if $\forall u \in \Sigma^n$, there is at most one $v \in C$ such that $d(u, v) \leq t$. The minimum distance of a code C is defined as

$$d(C) := \min\{d(u, v) : u, v \in C, u \neq v\}.$$

Lemma 4.4. *A code C corrects t errors if and only if $d(C) \geq 2t+1$.*

Proof. Suppose $d(C) \geq 2t+1$. We transmit a message u, and v is received, with $\leq t$ errors having occurred, meaning $d(u,v) \leq t$. Suppose w is a message with $d(w,v) \leq t$. Then by the triangle inequality, $d(u,w) \leq d(u,v) + d(v,w) \leq 2t$, meaning $w = u$. Thus u is the closest message to v. □

4.3. **Linear Codes.**

Definition 4.5. A linear code C is a subspace of Σ^n with the properties:

- $(0, \ldots, 0) \in C$;
- If $u, v \in C$ then $u + v \in C$;
- If $u \in C$ and $k \in \Sigma$, then $ku \in C$.

For every linear code C,

$$d(C) = \min\{d(0,w) : w \in C, w \neq 0\}.$$

According to linear algebra, we can specify a linear subspace by a basis or by linear equations. We consider the former case first:

We specify C by a $k \times n$ generator matrix, where $k \coloneqq dim(C)$, and whose rows are vectors of some basis of C. We construct a generator matrix of the form $G = (I_k|H)$, where I_k denotes the identity matrix of dimensions $k \times k$. H can be determined from the construction of a (7,4) Hamming code:

Lemma 4.6. *The generator matrix of a (7,4) Hamming code is* $\begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{pmatrix}$.

Proof. We recall the construction of a Hamming code. Say we wish to encode a 4-bit message $u = b_1, b_2, b_3, b_4$. The three added parity bits are

$$\begin{aligned} p_1 &= b_1 + b_2 + b_3 \\ p_2 &= b_2 + b_3 + b_4 \\ p_3 &= b_1 + b_2 + b_4. \end{aligned}$$

This produces the encoded message $b_1, b_2, b_3, b_4, p_1, p_2, p_3$.
The first part of G can be constructed from the following equation:

$$\begin{pmatrix} b_1 & b_2 & b_3 & b_4 \end{pmatrix} \begin{pmatrix} I_4 & P \end{pmatrix} = \begin{pmatrix} b_1 & b_2 & b_3 & b_4 & - & - & - \end{pmatrix}.$$

This holds for arbitrary P, such that the empty values denoted by - can be chosen to be the parity bits.
Using the formulas for the values of the parity bits, we obtain the following:

$$\begin{pmatrix} p_1 & p_2 & p_3 \end{pmatrix} = \begin{pmatrix} b_1 & b_2 & b_3 & b_4 \end{pmatrix} \begin{pmatrix} 1 & 0 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix}.$$

Where H is the 4×3 matrix, proving the initial claim. □

For example, if we wish to encode the message (1011), we can use the generator matrix to obtain the Hamming code

$$\begin{pmatrix}1 & 0 & 1 & 1\end{pmatrix}\begin{pmatrix}1 & 0 & 0 & 0 & 1 & 0 & 1\\ 0 & 1 & 0 & 0 & 1 & 1 & 1\\ 0 & 0 & 1 & 0 & 1 & 1 & 0\\ 0 & 0 & 0 & 1 & 0 & 1 & 1\end{pmatrix} = \begin{pmatrix}1 & 0 & 1 & 1 & 0 & 0 & 0\end{pmatrix},$$

which does in fact produce the correct encoding.

We now specify a linear subspace using linear equations, the latter case:

A linear code C is given as the set of solutions of a system of linear equations in the form of $P \cdot E(u) = 0$ where P is the parity check matrix of the code C and $E(u)$ is an encoded message.

Definition 4.7. The parity check matrix $P := (-H^T|I_{n-k})$.

We compute the parity check matrix of the previous example.

$$H^T = \begin{pmatrix}1 & 1 & 1 & 0\\ 0 & 1 & 1 & 1\\ 1 & 1 & 0 & 1\end{pmatrix}$$

This is equal to $-H^T$.

$$\begin{aligned} I_{n-k} &= I_{7-4} \\ &= \begin{pmatrix}1 & 0 & 0\\ 0 & 1 & 0\\ 0 & 0 & 1\end{pmatrix}\end{aligned}$$

It follows that

$$P = \begin{pmatrix}1 & 1 & 1 & 0 & 1 & 0 & 0\\ 0 & 1 & 1 & 1 & 0 & 1 & 0\\ 1 & 1 & 0 & 1 & 0 & 0 & 1\end{pmatrix}$$

We prove this parity check matrix is correct for the above example, using the predefined formula $P \cdot E(u) = 0$.

$$\begin{pmatrix}1 & 1 & 1 & 0 & 1 & 0 & 0\\ 0 & 1 & 1 & 1 & 0 & 1 & 0\\ 1 & 1 & 0 & 1 & 0 & 0 & 1\end{pmatrix}\begin{pmatrix}1\\0\\1\\1\\0\\0\\0\end{pmatrix} = \begin{pmatrix}0\\0\\0\end{pmatrix},$$

which proves the claim for this case. We can generalize this construction.

Definition 4.8. A generalized Hamming code is a linear code over Σ_2 of length $n := 2^l - 1$ with a parity check matrix P whose columns are all nonzero vectors from Σ_2^l, where l is a parameter $l \geq 2$.

Proposition 4.9. *The generalized Hamming code C has $d(C) = 3$ and thus it corrects one error.*

Proof. To show that $d(C) \geq 3$, it is sufficient to prove that every nonzero $E(u) \in C$ has ≥ 3 nonzero values. We proceed by proving that a Hamming distance of 1 or 2 can not be obtained for a code $E(u) \in \Sigma^n$, i.e. $P \cdot E(u) \neq 0$ when $E(n)$ contains one or two 1s. We analyze the two cases.

Case 1 ($D(C) = 1$): It is impossible for the encoded message to contain only one 1, since each of the four bits in the initial message contribute to ≥ 2 of the parity bits, meaning they appear ≥ 3 times.

Case 2 ($D(C) = 2$): With a similar argument, each initial bit appears once in the message, and twice (or three times) in the parity bits, thus making it impossible for there to be two 1s. □

4.3.1. *Decoding a generalized Hamming code.* When an encoded message v is transmitted, a message v' is received, such that if an error t has occurred, then $v' = v + t_i$ for some $i \in \{1, 2, \ldots, n\}$ where t_i has 1 at position i and 0 elsewhere, while if an error hasn't occurred, $v' = v$. Using the parity check matrix, when $v' = v$, we obtain $Pv' = 0$, while when $v' = v + t_i$, $Pv' = Pv + Pt_i = Pt_i$, the i-th column of the parity check matrix. Thus for ≤ 1 errors, we can determine whether or not an error has occurred, and the position of the incorrect bit.

The main advantage of the generalized Hamming code is that it is able to determine whether or not an error is present, and locate which bit contains the error, allowing for a correction to obtain the transmitted message.

References

[1] R. W. Hamming, *Error detecting and error correcting codes*, The Bell system technical journal, 29 (1950), pp. 147–160.

[2] J. Matoušek, *Thirty-three miniatures: Mathematical and Algorithmic applications of Linear Algebra*, American Mathematical Society Providence, RI, 2010.

[3] C. E. Shannon, *A mathematical theory of communication bell system technical journal, vol. 27*, July and October, (1948).

[4] M. Sudan, *Coding theory: Tutorial & survey*, in Proceedings 42nd IEEE Symposium on Foundations of Computer Science, IEEE, 2001, pp. 36–53.