Reed Solomon Code

What can a Quantum Computer compute?

Kumar Arpit

July 2021

Error Correction and Detection

- Error detection and correction or error control are techniques that enable reliable delivery of digital data over unreliable communication channels. Many communication channels are subject to channel noise, and thus errors may be introduced during transmission from the source to a receiver. Error detection techniques allow detecting such errors, while error correction enables reconstruction of the original data in many cases often without retransmission.

Error Correction and Detection

- Error detection and correction or error control are techniques that enable reliable delivery of digital data over unreliable communication channels. Many communication channels are subject to channel noise, and thus errors may be introduced during transmission from the source to a receiver. Error detection techniques allow detecting such errors, while error correction enables reconstruction of the original data in many cases often without retransmission.
- Error Correcting Codes: The central idea is the sender encodes the message with redundant information in the form of an ECC. The redundancy allows the receiver to detect a limited number of errors that may occur anywhere in the message, and often to correct these errors without retransmission.

Error Correction and Detection

- Error detection and correction or error control are techniques that enable reliable delivery of digital data over unreliable communication channels. Many communication channels are subject to channel noise, and thus errors may be introduced during transmission from the source to a receiver. Error detection techniques allow detecting such errors, while error correction enables reconstruction of the original data in many cases often without retransmission.
- Error Correcting Codes: The central idea is the sender encodes the message with redundant information in the form of an ECC. The redundancy allows the receiver to detect a limited number of errors that may occur anywhere in the message, and often to correct these errors without retransmission.
- Example: We can play a CD having scratches with fidelity. In fact, the ECC used for this case is Reed-Solomon Code.

Some Definitions

- A finite field or a Galois field is a finite set which is a field.
  A finite field of order q exists if and only if q is a prime power p^k (where p is a prime number and k is a positive integer).

Some Definitions

- A finite field or a Galois field is a finite set which is a field.
  A finite field of order q exists if and only if q is a prime power p^k (where p is a prime number and k is a positive integer).
- In communication theory, a code word is an element of a standardized code or protocol. Each code word is assembled in accordance with the specific rules of the code and assigned a unique meaning.

Some Definitions

- A finite field or a Galois field is a finite set which is a field.
  A finite field of order q exists if and only if q is a prime power p^k (where p is a prime number and k is a positive integer).
- In communication theory, a code word is an element of a standardized code or protocol. Each code word is assembled in accordance with the specific rules of the code and assigned a unique meaning.
- The code rate of a forward error correction code is the proportion of the data-stream that is useful (non-redundant). That is, if the code rate is k/n for every k bits of useful information, the coder generates a total of n bits of data, of which $n - k$ are redundant.

Some Definitions(Contd...)

- The Hamming distance,nor simply distance, between two words of the same length, w_1 and w_2 , denoted $D(w_1, w_2)$, is the number of places in which the components of w_1 and w_2 differ. The weight of a word is the Hamming distance between it and the zero word.
  The distance between any two distinct words is non-negative, and $D(w_1, w_2) = 0$ if and only if $w_1 = w_2$ .Clearly, $D(w_1, w_2) = D(w_2, w_1)$
  Also, it can be shown with induction that it satisfies the triangle inequality. Together this shows that Hamming distance is a metric.

Some Definitions(Contd...)

- The Hamming distance,nor simply distance, between two words of the same length, w_1 and w_2 , denoted $D(w_1, w_2)$, is the number of places in which the components of w_1 and w_2 differ. The weight of a word is the Hamming distance between it and the zero word.
  The distance between any two distinct words is non-negative, and $D(w_1, w_2) = 0$ if and only if $w_1 = w_2$ .Clearly, $D(w_1, w_2) = D(w_2, w_1)$ Also, it can be shown with induction that it satisfies the triangle inequality. Together this shows that Hamming distance is a metric.
- The distance d of the linear code is the minimum weight of its nonzero codewords, or equivalently, the minimum distance between distinct codewords.

Some Definitions(Contd...)

- The Hamming distance,nor simply distance, between two words of the same length, w_1 and w_2 , denoted $D(w_1, w_2)$, is the number of places in which the components of w_1 and w_2 differ. The weight of a word is the Hamming distance between it and the zero word.
  The distance between any two distinct words is non-negative, and $D(w_1, w_2) = 0$ if and only if $w_1 = w_2$ .Clearly, $D(w_1, w_2) = D(w_2, w_1)$ Also, it can be shown with induction that it satisfies the triangle inequality. Together this shows that Hamming distance is a metric.
- The distance d of the linear code is the minimum weight of its nonzero codewords, or equivalently, the minimum distance between distinct codewords.
- A generator matrix is a matrix whose rows form a basis for a linear code. The codewords are all of the linear combinations of the rows of this matrix, that is, the linear code is the row space of its generator matrix.It is not unique.

Linear codes

- In coding theory, a linear code is an error-correcting code for which any linear combination of codewords is also a codeword. A linear code of length n, dimension k, and distance d is called an [n,k,d] code.

Linear codes

- In coding theory, a linear code is an error-correcting code for which any linear combination of codewords is also a codeword. A linear code of length n, dimension k, and distance d is called an [n,k,d] code.
- A linear code of length n and rank k is a linear subspace C with dimension k of the vector space $\mathbb{F}_q^n$ where $\mathbb{F}_q$ is the finite field with q elements. Such a code is called a q-ary code. The vectors in C are called codewords. The size of a code is the number of codewords and equals q_k.

Linear codes

- In coding theory, a linear code is an error-correcting code for which any linear combination of codewords is also a codeword. A linear code of length n, dimension k, and distance d is called an [n,k,d] code.
- A linear code of length n and rank k is a linear subspace C with dimension k of the vector space $\mathbb{F}_q^n$ where $\mathbb{F}_q$ is the finite field with q elements. Such a code is called a q-ary code. The vectors in C are called codewords. The size of a code is the number of codewords and equals q_k.
- They are broadly divided into two types:
  - Block Codes
  - Convolutional Codes

Linear codes(Contd...)

- If C is a linear block code, then we define $C^{\perp}$. as the dual code of C where:

$$C^{\perp} = \{v \in F^n | v \cdot c = 0, \forall c \in C\}$$

A parity check matrix H of a code C is an n x (n - k) matrix whose rows generate the orthogonal complement of C. The rows of H generate the null space of the generator matrix G, and H is also agenerator matrix for the code $C^{\perp}$

Linear codes(Contd...)

- If C is a linear block code, then we define $C^{\perp}$. as the dual code of C where:

$$C^{\perp} = \{v \in F^n | v \cdot c = 0, \forall c \in C\}$$

A parity check matrix H of a code C is an n x (n - k) matrix whose rows generate the orthogonal complement of C. The rows of H generate the null space of the generator matrix G, and H is also agenerator matrix for the code $C^{\perp}$

- Let's imagine that the codewords in a code are points in a three-dimensional space and draw a spher eof radius r = d/2 around each point, we will have set of spheres that may touch but they will not intersect. Now, consider that we receive a codeword that has just been transmitted. We take its value and mark its coordinate in this imaginary three-dimensional space. If less than d/2 errors occurred in transmission, then the coordinate will still be in the sphere belonging to the correct codeword.

Singleton Bound

The Singleton bound establishes a useful relationship between n, k, and the minimum distance:
The minimum distance, d, of an (n,k) code satisfies the following inequality: $d \leq n - (k - l)$.

Singleton Bound

The Singleton bound establishes a useful relationship between n, k, and the minimum distance:
The minimum distance, d, of an (n,k) code satisfies the following inequality: $d \leq n - (k - l)$.
Proof:
Project all the codewords on the first (k - 1) coordinates. Since there are q^k different codewords, by the pigeon-hole principle at least two of them should agree on these (k-1) coordinates. But these then disagree on at most the remaining n-(k - 1) coordinates.And hence the minimum distance of the code C is $d \leq n - (k - 1)$.

Singleton Bound

The Singleton bound establishes a useful relationship between n, k, and the minimum distance:
The minimum distance, d, of an (n,k) code satisfies the following inequality: $d \leq n - (k - l)$.
Proof:
Project all the codewords on the first (k - 1) coordinates. Since there are q^k different codewords, by the pigeon-hole principle at least two of them should agree on these (k-1) coordinates. But these then disagree on at most the remaining n-(k - 1) coordinates.And hence the minimum distance of the code C is $d \leq n - (k - 1)$.
(Note that we are not guaranteed the existence of a code that satisfies the Singleton bound with equality for all n and k) An (n, k) code is maximum distance separable (MDS) if it satisfies the Singleton bound with equality:$d = n - (k - 1)$.

Basic Method for Using an ECC

Basic Method for Using an ECC

1. We are given some data to encode as some vector u of k elements from a field F; $u \in F^k$

Basic Method for Using an ECC

1. We are given some data to encode as some vector u of k elements from a field $F; u \in F^k$
2. We then encode this vector u by mapping it to a codeword $v \in F^n$. This can be accomplished by multiplying by a generator matrix G. For example, $v = uG$.

Basic Method for Using an ECC

1. We are given some data to encode as some vector u of k elements from a field F; $u \in F^k$
2. We then encode this vector u by mapping it to a codeword $v \in F^n$. This can be accomplished by multiplying by a generator matrix G. For example, $v = uG$.
3. We transmit the codeword v ; errors may occur during transmission.

Basic Method for Using an ECC

1. We are given some data to encode as some vector u of k elements from a field F; $u \in F^k$
2. We then encode this vector u by mapping it to a codeword $v \in F^n$. This can be accomplished by multiplying by a generator matrix G. For example, $v = uG$.
3. We transmit the codeword v ; errors may occur during transmission.
4. We receive a codeword w = v + e where $e \in F^n$ contains the errors that occurred during transmission.

Basic Method for Using an ECC

1. We are given some data to encode as some vector u of k elements from a field F; $u \in F^k$
2. We then encode this vector u by mapping it to a codeword $v \in F^n$. This can be accomplished by multiplying by a generator matrix G. For example, $v = uG$.
3. We transmit the codeword v ; errors may occur during transmission.
4. We receive a codeword w = v + e where $e \in F^n$ contains the errors that occurred during transmission.
5. **We attempt to recover v given w by finding the closest codeword $c \in F^n$ to the vector w.**

Basic Method for Using an ECC

1. We are given some data to encode as some vector u of k elements from a field$F; u \in F^k$
2. We then encode this vector u by mapping it to a codeword $v \in F^n$. This can be accomplished by multiplying by a generator matrix G. For example, $v = uG$.
3. We transmit the codeword v ; errors may occur during transmission.
4. We receive a codeword w = v + e where $e \in F^n$ contains the errors that occurred during transmission.
5. **We attempt to recover v given w by finding the closest codeword $c \in F^n$ to the vector w.**
6. The original data is determined by finding the $d \in F^k$such that c = dG.

Reed Solomon Codes

- Reed Solomon Code is ideal for situation where majority of errors occurs ij burst.
- For any parameters n, k , and $d = n - k + 1$ with $(1 \leq k \leq n)$ and a finite field F_q, there exists a (n, k, d) Reed-Solomon (RS) code over F_q so long as $n \leq q + 1$. Since $n \leq q + 1$, codes over the binary field F_2 are limited to length three and are not particularly interesting, so we usually consider non-binary RS codes over the field F_q where q can be quite large. In channels where it is convenient to transmit binary m-tuples, we let $q = 2^m$, with F_q forming an extension field of F_2 .
- The standard (or punctured) code is found by setting $n = q - 1$.

RS codes as Evaluation codes

Using a k x n generator matrix like the following is one method of performing the mapping from F_q^k to F_q^n.

$$\begin{bmatrix} 1 & 1 & 1 & 1 & \dots & \dots & \dots & 1 \\ 1 & \alpha & \alpha^2 & \alpha^3 & \dots & \dots & \dots & \alpha^{n-1} \\ 1 & \alpha^2 & \alpha^4 & \alpha^6 & \dots & \dots & \dots & \alpha^{n-2} \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 1 & \alpha^{k-1} & \alpha^{2k-2} & \alpha^{3k-3} & \dots & \dots & \dots & \alpha^{n-k+1} \end{bmatrix}$$

Where, α is a primitive element in the field F_q

RS codes as Evaluation codes

Using a k x n generator matrix like the following is one method of performing the mapping from F_q^k to F_q^n.

$$\begin{bmatrix} 1 & 1 & 1 & 1 & \dots & \dots & \dots & 1 \\ 1 & \alpha & \alpha^2 & \alpha^3 & \dots & \dots & \dots & \alpha^{n-1} \\ 1 & \alpha^2 & \alpha^4 & \alpha^6 & \dots & \dots & \dots & \alpha^{n-2} \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 1 & \alpha^{k-1} & \alpha^{2k-2} & \alpha^{3k-3} & \dots & \dots & \dots & \alpha^{n-k+1} \end{bmatrix}$$

Where, α is a primitive element in the field F_q

In field theory, a primitive element of a finite field GF(q) is a generator of the multiplicative group of the field. In other words,$\alpha \in$ GF(q) is called a primitive element if it is a primitive $(q-1)$th root of unity in GF(q); this means that each non-zero element of GF(q) can be written as α^i for some integer i.

RS codes as Evaluation codes(Contd...)

- The benefit of using this particular generator matrix is that we can easily compute the matrix multiplication by means of simple polynomial evaluation at the field elements.
  By noticing that we may obtain the ith row of G by evaluating the polynomial x^i at the field elements α_j, we arrive at a clear description of the codewords independent of the generator matrix:

$$RS[n = q-1, k, d] = \{(J(\alpha^0), f(\alpha^1), f(\alpha^2), ..., f(\alpha^{n-l})) : f \in F_q[z] and degf(z) < k\}$$

RS codes as Evaluation codes(Contd...)

- The benefit of using this particular generator matrix is that we can easily compute the matrix multiplication by means of simple polynomial evaluation at the field elements.
  By noticing that we may obtain the ith row of G by evaluating the polynomial x^i at the field elements α_j, we arrive at a clear description of the codewords independent of the generator matrix:

  $$RS[n = q-1, k, d] = \{(J(\alpha^0), f(\alpha^1), f(\alpha^2), ..., f(\alpha^{n-l})) : f \in F_q[z] and degf(z) < k\}$$

- There is a unique polynomial f(z) corresponding to each element of F_q^k, but the field elements at which we evaluate the polynomials will remain fixed.

RS codes as Evaluation codes(Contd...)

- Theorem: The q_k n-tuples generated by the mappings f(z) $\rightarrow$ $\{f(\alpha^i), 0 \leq j < n\}$ form a linear $(n = q - 1, k, d = n - k + 1)$ MDS code over F_q, where α is· a primitive element in F_q and the polynomials$f(z) \in F_q[z]$ have degree less than k.

RS codes as Evaluation codes(Contd...)

- Proof:
  The code is linear because the sum of the codewords corresponding to two polynomials f(z) and h(z) is the codeword corresponding to the polynomial $f(z) + h(z)$, and the multiple of the codeword corresponding to f(z) by $\beta \in F_q$ is the codeword corresponding to the polynomial βf(z).
  A codeword has a zero symbol in the coordinate corresponding to β_i iff $f\beta_i)$=0; ie, if and only if β_i is a solution of the equation $f(z) = 0$. By the fundamental theorem of algebra, if $f(z) \neq 0$, then since $degf(z) < k - 1$, this equation can have at most k-1 roots in F_q, therefore a nonzero codeword can have at most k-1 symbols equal to zero, so its weight is at least n-k+1. Since the code is linear, this implies that its minimum distance is at least $d \geq n - k + 1$. But by the Singleton bound, $d \leq n - k + 1$; thus $d = n - k + l$.

An Example

Example. Let's look at a standard (punctured) RS code where $n = q - 1$. Let $q = 4$, then $n = 3$, and we'll let $k = 2$. Since RS codes are MDS, $d = n - k + 1 = 2$, and so we have a $(3, 2, 2)$ RS code. We will be working over the field $\mathbb{F}_4$. Recall that addition and multiplication in this field can be defined by the following tables.

$+$	0	1	α	α^2
0	0	1	α	α^2
1	1	0	α^2	α
α	α	α^2	0	1
α^2	α^2	α	1	0

$*$	0	1	α	α^2
0	0	0	0	0
1	0	1	α	α^2
α	0	α	α^2	1
α^2	0	α^2	1	α

Taking vectors of length $k = 2$, the following mapping yields codewords of length n.

$$RS[3, 2, 2] : f(z) = f_0 + f_1 z \mapsto (f(\alpha^0), f(\alpha^1), f(\alpha^2))$$

where $\alpha \in \mathbb{F}_4$ is primitive, $f(z) \in \mathbb{F}_4[z]$ and $\deg f(z) < 2$. The following is equivalent:

$$(f_0, f_1) \mapsto (f_0 + f_1\alpha^0, f_0 + f_1\alpha^1, f_0 + f_1\alpha^2)$$

Using the two generators $g_0 = (1, 1, 1)$ and $g_1 = (1, \alpha, \alpha^2)$, which are a basis for the vector space we wish to create, we can build the following generator matrix.

$$G = \begin{bmatrix} 1 & 1 & 1 \\ 1 & \alpha & \alpha^2 \end{bmatrix}$$

This code has 16 codewords since each of f_0 and f_1 can range over the four elements of $\mathbb{F}_4$. By multiplying the possible input vectors by the generator matrix we arrive at the following coding map.

An Example(Contd...)

$$
\begin{array}{llllll}
(0, & 0) & \mapsto & (0, & 0, & 0) \\
(0, & 1) & \mapsto & (1, & \alpha, & \alpha^2) \\
(0, & \alpha) & \mapsto & (\alpha, & \alpha^2, & 1) \\
(0, & \alpha^2) & \mapsto & (\alpha^2, & 1, & \alpha) \\
(1, & 0) & \mapsto & (1, & 1, & 1) \\
(1, & 1) & \mapsto & (0, & \alpha^2, & \alpha) \\
(1, & \alpha) & \mapsto & (\alpha^2, & \alpha, & 0) \\
(1, & \alpha^2) & \mapsto & (\alpha, & 0, & \alpha^2)
\end{array}
\qquad
\begin{array}{llllll}
(\alpha, & 0) & \mapsto & (\alpha, & \alpha, & \alpha) \\
(\alpha, & 1) & \mapsto & (\alpha^2, & 0, & 1) \\
(\alpha, & \alpha) & \mapsto & (0, & 1, & \alpha^2) \\
(\alpha, & \alpha^2) & \mapsto & (1, & \alpha^2, & 0) \\
(\alpha^2, & 0) & \mapsto & (\alpha^2, & \alpha^2, & \alpha^2) \\
(\alpha^2, & 1) & \mapsto & (\alpha, & 1, & 0) \\
(\alpha^2, & \alpha) & \mapsto & (1, & 0, & \alpha) \\
(\alpha^2, & \alpha^2) & \mapsto & (0, & \alpha, & 1)
\end{array}
$$

This map tells us how to map every input vector into a codeword which we can transmit.

Guruswami-Sudan Algorithm

Continued in Notes ::> _ <::