IERG 6120 Coding Theory for Storage Systems Lecture 5 - 27/09/2016

Reed-Solomon Code and Its Application

Lecturer: Kenneth Shum *Scribe: Xishi Wang*

1 Equivalence of Codes

Two linear codes are said to be *equivalent* if one of them can be obtained from the other by means of a sequence of transformations of the following types:

(i) a permutation of the positions of the code;

(ii) multiplication of symbols in a fixed position by a non-zero scalar in F.

Note that these transformations can be applied to all code symbols.

2 Reed-Solomon Code

In RS code each symbol in the codewords is the evaluation of a polynomial at one point α, namely,

$$f(\alpha) = \begin{bmatrix} c_0 & c_1 & c_2 & \cdots & c_{k-1} \end{bmatrix} \begin{bmatrix} 1 \\ \alpha \\ \alpha^2 \\ \vdots \\ \alpha^{k-1} \end{bmatrix}.$$

The whole codeword is given by n such evaluations at distinct points $\alpha_1, \cdots, \alpha_n$,

$$\begin{bmatrix} f(\alpha_1) & f(\alpha_2) & \cdots & f(\alpha_n) \end{bmatrix} = \begin{bmatrix} c_0 & c_1 & c_2 & \cdots & c_{k-1} \end{bmatrix} \begin{bmatrix} 1 & 1 & \cdots & 1 \\ \alpha_1 & \alpha_2 & \cdots & \alpha_n \\ \alpha_1^2 & \alpha_2^2 & \cdots & \alpha_n^2 \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_1^{k-1} & \alpha_2^{k-1} & \cdots & \alpha_n^{k-1} \end{bmatrix} = \mathbf{c}^T G.$$

G is the generator matrix of $\mathrm{RS}_q(n, k, d)$ code. The order of writing down the field elements α_1 to α_n is not important as far as the code structure is concerned, as any permutation of the α_i's give an equivalent code.

The generator matrix G is a Vandermonde matrix, which is of the form $[a_j^i]_{i=0,\ldots,k-1}^{j=1,\ldots,n}$. The following theorem concerns the determinant of a Vandermonde matrix.

Theorem 1. *The determinant of a Vandermonde matrix is given by:*

$$V_n = \begin{vmatrix} 1 & 1 & \cdots & 1 \\ a_1 & a_2 & \cdots & a_n \\ a_1^2 & a_2^2 & \cdots & a_n^2 \\ \vdots & \vdots & \ddots & \vdots \\ a_1^{n-1} & a_2^{n-1} & \cdots & a_n^{n-1} \end{vmatrix} = \prod_{j>i} (a_j - a_i)$$

Proof If $\exists i \neq j$ such that $a_j = a_i$, the determinant is zero and hence the theorem holds.
If all a_i's are distinct, we proof the theorem by induction. For $n = 1$ and $n = 2$, one can easily check the theorem holds. Assume that the theorem is true for $n = k \geq 2$. For $n = k + 1$, the determinant is given by

$$V_{k+1} = \begin{vmatrix} 1 & 1 & \cdots & 1 \\ x & a_2 & \cdots & a_{k+1} \\ x^2 & a_2^2 & \cdots & a_{k+1}^2 \\ \vdots & \vdots & \ddots & \vdots \\ x^k & a_2^k & \cdots & a_{k+1}^k \end{vmatrix}.$$

Expand V_{k+1} in terms of the first column and the result is a polynomial $f(x)$ in x whose degree is no greater than k. If we substitute $a_i, 2 \leq i \leq n$ for x, the value of the determinant will be 0. Since a_i are distinct, it follows that

$$f(a_2) = f(a_3) = \ldots = f(a_{k+1}) = 0.$$

Therefore,

$$f(x) = C(x - a_2)(x - a_3) \cdots (x - a_k)(x - a_{k+1}).$$

As the degree of $f(x)$ is no greater than k, it follows that C is independent of x. From the expansion of determinant, the coefficient of x^k is

$$(-1)^{k-1} \begin{vmatrix} 1 & 1 & \cdots & 1 \\ a_2 & a_3 & \cdots & a_{k+1} \\ \vdots & \ddots & \vdots & \vdots \\ a_2^{k-1} & a_3^{k-1} & \cdots & a_{k+1}^{k-1} \end{vmatrix},$$

which, by induction hypothesis when $n = k$, is equal to

$$\prod_{2 \leq i < j \leq k+1} (a_j - a_i).$$

Hence, we have

$$f(x) = (x - a_2)(x - a_3) \cdots (x - a_k)(x - a_{k+1})(-1)^{k-1} \prod_{2 \leq i < j \leq k+1} (a_j - a_i).$$

So the theorem holds for $n = k + 1$. Therefore,

$$V_n = (a_2 - a_1)(a_3 - a_1) \cdots (a_k - a_1)(a_{k+1} - a_1) \prod_{2 \leq i < j \leq n} (a_j - a_i) = \prod_{1 \leq i < j \leq n} (a_j - a_i).$$

□

Notice that when a_i are distinct, the determinant of any $k \times k$ sub-matrix of a $k \times n$ Vandermonde matrix is non-zero. Therefore, we have the following corollary.

Corollary 2. *Any $k \times k$ sub-matrix of a $k \times n$ Vandermonde matrix is invertible if all a_i are distinct.*

3 Application

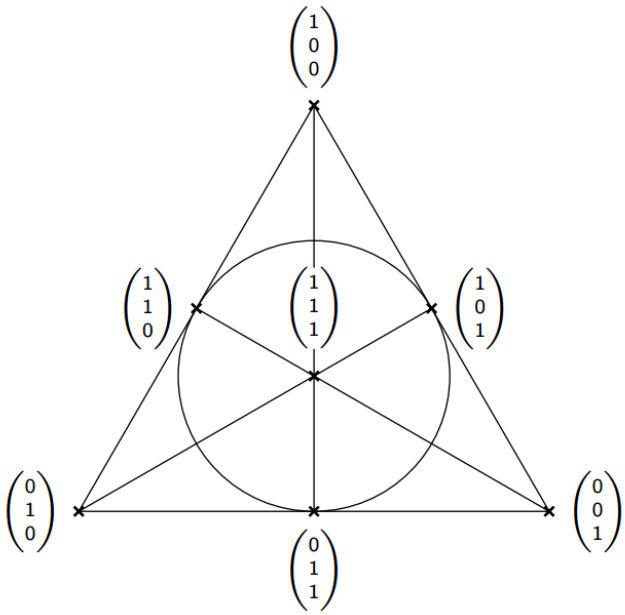
Reed-Solomon Code can be applied to disk storage. Suppose we have n disks. In the i-th disk, a symbol $f(\alpha_i)$ of RS code is stored. If we pick any k disks $i_1, i_2, \ldots, i_k$, we have

$$\begin{bmatrix} f(\alpha_{i_1}) & f(\alpha_{i_2}) & \cdots & f(\alpha_{i_k}) \end{bmatrix} = \begin{bmatrix} c_0 & c_1 & c_2 & \cdots & c_{k-1} \end{bmatrix} \begin{bmatrix} 1 & 1 & \cdots & 1 \\ \alpha_{i_1} & \alpha_{i_2} & \cdots & \alpha_{i_k} \\ \alpha_{i_1}^2 & \alpha_{i_2}^2 & \cdots & \alpha_{i_k}^2 \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_{i_1}^{k-1} & \alpha_{i_2}^{k-1} & \cdots & \alpha_{i_k}^{k-1} \end{bmatrix}$$

The $k \times k$ matrix is invertible and hence one can recover n nodes by any k nodes, namely, RS code satisfies (n, k)-*recovery property*. However, to recover any one node, RS code needs to query k nodes, which is not desirable for distributed storage systems. This motivates the the construction of code such that it can recover one erasure symbol from a small number of other symbols. A *locally repairable code* is thus defined as a code in which any code symbol is a function of r other code symbols, where r is an integer less than the dimension of the code.
An example of binary code with locality $r = 2$ is the $(7, 3, 4)_2$ simplex code. The generator matrix G is given by

$$G = \begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix}.$$

The locality $r = 2$ can be shown in a Fano plane. We can compute any point by adding two other points in the same line.



4 LRC by Modifying RS Code

In this section, we give an example of locally repairable code by modifying Reed-Solomon code. The construction hinges on the property that the function $f(x) = x^3$ takes on only two values on the non-zero element in $GF(7)$. We can obtain the codewords of the LRC by evaluating a polynomial of the form $c_0 + c_1 x + c_3 x^3$ on the non-zero elements of $GF(7)$. By appropriately permuting the components of the code, we can write the generator matrix as

$$G = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 2 & 4 & 3 & 6 & 5 \\ 1 & 1 & 1 & 6 & 6 & 6 \end{bmatrix}.$$

And the parity-check matrix is

$$H = \begin{bmatrix} 2 & 4 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 & 4 & 1 \\ 1 & 1 & 1 & 6 & 6 & 6 \end{bmatrix}.$$

Observe that H has non-zero element in the first three symbols of the first row and in the last three symbols of the second row. Hence this code has locality 2. To recover one erasure, accessing two other symbols is sufficient. To recover two erasures, access four other symbols. In summary, this code has $n = 6, k = 3, r = 2$. Since each codeword is obtained by evaluating a polynomial of degree at most 3, there are at most three zeros in a codeword. Hence the and minimum distance is $d = 6 - 3 = 3$.

Exercises: 1. Let p be a prime and s be a positive integer. Let $GF(p)$ be a finite field of size p. Consider the set of nonzero vectors (x, y, z) in $GF(p)^3$. There are $p^3 - 1$ such vectors. Define an equivalence relation $\sim$ on them by declaring that $(x, y, z) \sim (x', y', z')$ if there exists a non-zero $\lambda \in GF(p)$ such that

$$(x, y, z) = \lambda \cdot (x', y', z').$$

In other words, two nonzero vectors are said to be equivalent if they are in the same direction. Each equivalence class contains $p - 1$ vectors. Arbitrarily pick a representative from each equivalence class, and form a $3 \times (\frac{p^3-1}{p-1})$ matrix. Take this matrix as the generator of a code. Determine the locality and minimum distance of this code. (When $p = 2$, this is equal to the $(7, 3, 4)_2$ simplex code.) (The choice of representative of each equivalence class is not very important, because a different set of representatives will give an equivalent code.)

2. Construct a locally repairable code of length $n = 10$ over $GF(11)$, that has minimum distance $d = 5$ and locality $r = 4$. Write down a generator matrix of your code. Explain why your code has minimum distance 5 and locality 4.

Hint: The polynomial $f(x) = x^5$ takes two values on the non-zero elements of $GF(11)$,

$$x^5 = \begin{cases} 1 \bmod 11 & \text{if } x = 1, 3, 4, 5, 9 \\ 10 \bmod 11 & \text{if } x = 2, 6, 7, 8, 10. \end{cases}$$

Obtain the codewords by evaluating polynomials of the form

$$c_0 + c_1 x + c_2 x^2 + c_3 x^3 + c_5 x^5$$

on the non-zero elements of $GF(11)$.