15:853: Algorithms in the Real World
Fall 2019

Lecture 4

September 12 2019

Lecturer: Rashmi Vinayak *Scribe: Yuwei Qin*

4.1 Polynomials over $\mathbb{F}_p$ (slide 14)

Notation for set of polynomial $\mathbb{F}_q[x]$:

- x is the variable;
- coefficient are from field $\mathbb{F}_q$;
- degree of polynomial is the maximum power of variable in the polynomial;
- please note the value for x does NOT have to be in $\mathbb{F}_q$.

An example of $\mathbb{F}_5[x]$: $f(x) = 3x^4 + 1x^3 + 4x^2 + 3$, where the coefficients are from $\mathbb{F}_5 = \{0, 1, 2, 3, 4\}$.

Question: For the set of polynomials $\mathbb{F}_q[x]$ with polynomial addition (+) and polynomial multiplication (*) as the binary operators, do they form a field?
Answer: no, because it may not have a multiplicative inverse.

Next we will see that by replacing the regular polynomial addition and multiplication operators into modulo addition and multiplication with respect to irreducible polynomial (slide 16), we can build a finite field (slide 17).

4.2 Galois Fields (slide 17)

Let us consider $\mathbb{F}_q[x]$: set of ALL polynomials where the coefficients are from $\mathbb{F}_q$, and its order is infinite because we can have polynomials of infinite degrees.
Now we will use modulo arithmetic to make it a finite field. This procedure is similar to the case where we go from set of all integers to finite field $\mathbb{Z}_q$ given q being prime (slide 13).
This time, instead of taking modulo with respect to a prime number, we will take modulo with respect to the irreducible polynomials. The two operators for the field are the polynomial addition and multiplication with modulo in respect to the irreducible polynomials $p(x)$:

$$+ \bmod p(x)$$
$$* \bmod p(x)$$

Question: How many elements are there in this finite field?
Answer: Let us assume $deg(p(x)) = n$, then the field $\mathbb{F}_q[x] \bmod p(x)$ has all polynomials of degree $n-1$, hence the possible number of coefficients will be n. For each coefficient, it can only take p possible values. Altogether, the number of elements in the finite field $\mathbb{F}_q[x] \bmod p(x)$ equals q^n.

For example, if $deg(p(x)) = 3$, the general form of polynomials for $\mathbb{F}_q[x] \bmod p(x)$ is

$$ax^2 + bx + c$$

The degree of the above polynomial is $3 - 1 = 2$, and the number of coefficients in this case is 3: a, b, and c. Each coefficient can take q possible values from set $\mathbb{F}_q$. So the total number of elements in this field is q^3.

4.3 Euclid's Algorithm (slide 22)

Question: How can we use the Euclid's algorithm and the extended Euclid's algorithm to find inverse of a: $a^{-1} \bmod n$ with n being prime?
Answer: The goal here is to find x such that $ax = 1$.
Let us start with a prime number n and apply the extended Euclid's algorithm: choose x and y such that

$$ax + ny = gcd(a, n) \tag{4.1}$$

Since the operation is over modulo n, 2nd term on the left (ny) becomes 0. For prime number n, its gcd with anything is 1:

$$gcd(a, n) = 1$$

Equation (4.1) becomes:

$$ax = 1$$

Therefore, we find the inverse of a under modulo n.

4.4 Generator matrix of RS code (slide 31)

Question: What is the generator matrix?
Answer: To find the generator matrix G_{RS}, let us consider the coding process for a message:

$$m = [a_{k-1},\ \cdots,\ a_1,\ a_0]$$

where $a_i \in GF(q^r)$. The message can also be represented as polynomial of degree k-1:

$$P(x) = a_{k-1}x^{k-1} + \cdots + a_1x + a_0$$

To get the code word, let us evaluate the polynomial at n distinct points: α_1, α_2, $\cdots$, α_n. The code word becomes:

$$\begin{aligned} P(\alpha_1) &= a_{k-1}\alpha_1^{k-1} + \cdots + a_1\alpha_1 + a_0 \\ P(\alpha_2) &= a_{k-1}\alpha_2^{k-1} + \cdots + a_1\alpha_2 + a_0 \\ &\vdots \\ P(\alpha_n) &= a_{k-1}\alpha_n^{k-1} + \cdots + a_1\alpha_n + a_0 \end{aligned}$$

If we write the above equations in the matrix form:

$$\begin{bmatrix} a_{k-1} & \cdots & a_1 & a_0 \end{bmatrix} \begin{bmatrix} \alpha_1^{k-1} & \alpha_2^{k-1} & \dots & \alpha_n^{k-1} \\ \vdots & \vdots & \vdots & \vdots \\ \alpha_1 & \alpha_2 & \dots & \alpha_n \\ 1 & 1 & \dots & 1 \end{bmatrix} = \begin{bmatrix} P(\alpha_1) & P(\alpha_2) & \cdots & P(\alpha_n) \end{bmatrix}$$

It is clear that the generator matrix for RS code is

$$G_{RS} = \begin{bmatrix} \alpha_1^{k-1} & \alpha_2^{k-1} & \dots & \alpha_n^{k-1} \\ \vdots & \vdots & \vdots & \vdots \\ \alpha_1 & \alpha_2 & \dots & \alpha_n \\ 1 & 1 & \dots & 1 \end{bmatrix}$$

In practice, it is common to flip the above generator matrix and write it in the form of the Vandermonde matrix:

$$\begin{bmatrix} 1 & 1 & \dots & 1 \\ \alpha_1 & \alpha_2 & \dots & \alpha_n \\ \vdots & \vdots & \vdots & \vdots \\ \alpha_1^{k-1} & \alpha_2^{k-1} & \dots & \alpha_n^{k-1} \end{bmatrix}$$