6.440 Essential Coding Theory Mar 3, 2008

Lecture 8

Lecturer: Madhu Sudan *Scribe: Dongwoon Bai*

1 Overview: Algebraic Geometry Codes

- Motivation: q-ary aysymptotics
- Introduction
- Construction

In this lecture, we will not give detailed proofs, nor even detailed constructions. We will just give the rough ideas of how these codes are working and how theses codes are invented mostly because these materials are far out of the scope of this course. Nevertheless, we want to introduce at least the basic concepts for these families of codes.

It is known that BCH codes are interesting because it shows that the Hamming bounds are tight for some codes. However, in today's lecture, we will talk about the codes over moderate size of alphabet not as large as that of RS codes nor as small as that of BCH codes which is binary, but somewhere between.

2 q-ary Aysymptotics: Bounds for q-ary Codes

We introduce the following relatively simple but effective Plotkin bound for codes over q-ary alphabet. For binary, it tells us that $R = 0$ for $\delta > 1/2$.

Bound 1 (Plotkin) *For q-ary codes, $R + \left(\frac{q}{q-1}\delta\right) \leq 1$.*

Suppose you are given $0 < R, \delta < 1$. Does a family of q-ary codes of rate R and distance δ exist? We know that the following bounds hold:

1. **Singleton:** $R + \delta \leq 1$.
2. **Plotkin:** $R + \delta < 1$.

The Plotkin bound give us the lower bound for q for given R, δ, i.e. binary and ternary alphabets cannot achieve $R = 0.6$ and $\delta = 0.3$. From the fact that

$$R + \delta + \Omega\left(\frac{1}{q}\right) \leq 1, \tag{1}$$

we have the necessary condition

$$q = \Omega\left(\frac{1}{1-(R+\delta)}\right). \tag{2}$$

What will be the sufficiency condition for q versus $\frac{1}{1-(R+\delta)}$? Is it linear? We don't know yet. Anyway, in this consideration, the RS codes are ruled out because they have fixed $q = n$ for given n.

Consider the following greedy construction. There exist $(n, k, d)_q$ codes with

$$q^k \mathrm{Vol}_q(n, d) \geq q^n. \tag{3}$$

The ball is in Σ^n, where $|\Sigma| = q$. Thus, the volume is given by

$$\mathrm{Vol}_q(n, d) = |\{x \in \Sigma^n | \Delta(x, 0) \leq d\}| \approx q^{H_q(\frac{d}{n})n}, \tag{4}$$

where

$$H_q(\delta) = \delta \log_q \frac{q-1}{\delta} + (1-\delta)\log_q \frac{1}{1-\delta}. \tag{5}$$

This implies

$$R + H_q(\delta) \geq 1. \tag{6}$$

Let us check the asymptotic of it as q increases. We can see that

$$\lim_{q\to\infty} H_q(\delta) = \lim_{q\to\infty} \delta \log_q(q-1) = \delta \tag{7}$$

This implies the following convergence:

$$H_q(\delta) = \delta + O\left(\frac{1}{\log q}\right). \tag{8}$$

Then, it suffices to have

$$q = 2^{\Omega\left(\frac{1}{1-(R+\delta)}\right)}. \tag{9}$$

3 Introduction to Algebraic Geometry (AG) Codes

3.1 AG Codes Bounds

AG codes are known to achieve

$$R + \delta \geq 1 - \frac{1}{\sqrt{q}-1} \tag{10}$$

if q a square and prime power. This is much better than greedy construction because now q only need to grow as fast as some polynomial of the gap $1/(1-(R+\delta))$ not exponential of it.

Suppose that we want to $[n,k,d]_q$ code with $k+d=n$. We can construct it with $q=n$. However, greedy construction needs $q \sim \exp(n)$. This implies that we can construct more efficient codes by using some algebraic properties.

3.2 Bivariate Polynomials

We try to follow the RS code approach but reduce the size of q for AG code constuction.

We encode message using a bivariate polynomial as follows:

$$\text{Message Space} = \{Q(x,y) \text{ over } \mathbb{F}_q | \deg x \leq l,\ \deg y \leq l\}. \tag{11}$$

The input message is embedded in the coefficients of such polynomials and we can get the encoded message by evaluating the polynomial for every $(x,y) \in \mathbb{F}_q^2$. This gives us $n=q^2$ and roughly $k=l^2$. Moreover, we have roughly q^{l^2} of such polynomials.

Let us check the distance of such a code. Consider the maximum number of zeros that these polynomials can have. We can construct polynomial with l zeros in x and l zeros in y, given by

$$\prod_{i=1}^{l}(x-\alpha_i) \times \prod_{i=1}^{l}(y-\beta_i), \tag{12}$$

where $\alpha_i \neq \alpha_j$ and $\beta_i \neq \beta_j$ for $i \neq j$. This polynomial has $q^2-(q-l)^2$ zeros, and thus $(q-l)^2$ nonzeros. Therefore, this construction gives us

$$[q^2, l^2, (q-l)^2]_q, \tag{13}$$

compared with

$$[q^2, l^2, q^2-l^2]_{q^2} \tag{14}$$

for RS codes.

Where is this distance gap coming from? For RS codes, whenever we are evaluating polynomial, we are getting the extra piece of information. However, for this codes, there is some redundancy. For example, assume that we already know that the polynomial is zero at $l+1$ points in a line (either the same x or y). Then, we already know that the polynomial becomes zero anywhere in the line.

4 Constuctions

4.1 Goppa's Cosntuction

The main idea is to evaluate bivariate (or multivariate) polynomials over $S \subsetneq \mathbb{F}_q \times \mathbb{F}_q$. By carefully choosing it, we hope we could improve the rate while the distance does not suffer. Then, can we pick S random? If we pick S randomly, we will encounter the union bound and this will lead us GV bound. It might work but we don't know yet. Thus, S should have some algebraic structure. Let us pick $R(x, y)$ and choose

$$S = \{(\alpha, \beta) \in \mathbb{F}_q^2 | R(\alpha, \beta) = 0\}. \tag{15}$$

Let us think about some bad examples.

- $R(x, y) = ax + by + c$: This is a code over q but the length reduces to $n = |S| = q$ as well. Thus, just the same as RS codes.
- $R(x, y) = ax^2 + bxy + cy^2 + d$: For any given x, we have 2 values of y for the solution, and thus $n = |S| \leq 2q$. It just gives us something similar to two sets of RS codes.

Choosing the low degree polynomial of R is not a good idea because if we fix one of x, y, we have only limited number of values for the other variable. Even higher degree polynomials can be bad as follows:

- $R(x, y) = \prod(x - \alpha) \times \prod(y - \beta)$: This is bad because it evaluates all the points in the lines, which we have been trying to avoid.

Therefore, we have to pick R such that it has moderately high degrees, and it is irreducible or at least it does not have any factors of $(x - \alpha)$ and $(y - \beta)$. Goppa has given some nice explicit constructions based on R in 1978.

4.2 Codes from Tsafasman, Vladuts, and Zink

In 1982, [Tsafasman, Vladuts, and Zink] gave constructions of codes as follows.

Theorem 2 *If q is an even power of prime, there exist $[n, k, d]_q$ codes satisfying*

$$n \geq k + d - \frac{n}{\sqrt{q} - 1}. \tag{16}$$

We introduce the following illustrative examples.

- $q = 13$
- $[19, 6, 13]_{13}$: This code and can be compared with RS code $[19, 6, 13]_{19}$
- $R(x, y) = y^2 - 2(x - 1)x(x + 1)$: It is not factored easily.
- Basis $= \{1, x, x^2, x^3, y, xy\}$: The terms having y^2 are excluded because $y^2 = 2(x - 1)x(x + 1)$ in S. y^2 with $1, x, x^2, x^3$ will yield redundancy.

We claim that if

$$Q(x, y) = a_0 + a_1 x + a_2 x^2 + a_3 x^3 + b_1 y + b_2 xy \tag{17}$$

shares 7 zeros with $R(x, y)$, then

$$a_0 = a_1 x = a_2 = a_3 = b_1 = b_2 = 0, \tag{18}$$

which can be proved using the following theorem.

Theorem 3 (Bezout) *Suppose that $R(x, y), Q(x, y)$ are polynomials satisfying $deg(R) \leq D_1$, $deg(Q) \leq D_2$. If they share more than $D_1 \cdot D_2$ zeroes, they must share a common factors.*

Using the Bezout's theorem, we can prove our original claim.

Generally, how can we build more codes? For example, how can we pick R? How can we analyze the distance? There are deep theories answering these questions but the solutions are out of scope of this course.

4.3 Hermitian Codes

These codes are over $\mathbb{F}_{q^2}$ i.e. square of some prime power. There are two functions with nice properties, which map from $\mathbb{F}_{q^2}$ to $\mathbb{F}_q$.

- Trace function: $Tr(y) = y^q + y$
- Norm function: $N(x) = x^{q+1}$

At this point, they just map $\mathbb{F}_{q^2} \to \mathbb{F}_{q^2}$. However, we can use the following fact.

Fact 4 *There exists a unique set $T \subset \mathbb{F}_{q^2}$, which is $\mathbb{F}_q = \{\alpha \in \mathbb{F}_{q^2} |$ roots of $x^q - x = 0\}$*

T consists of the elements of smaller field $\mathbb{F}_q$. For trace and norm functions,

$$[Tr(y)]^q = (y^q + y)^q = y^{q \cdot q} + y^q = y + y^q = Tr(y), \tag{19}$$

$$[N(x)]^q = (x^{q+1})^q = x^{q^2+q} = x^{1+q} = N(x), \tag{20}$$

and thus they map $\mathbb{F}_{q^2} \to \mathbb{F}_q$. Then, Hermitian R is given by

$$R(x, y) = Tr(y) - N(x). \tag{21}$$

Moreover, we have the following properties. For any $\gamma \in \mathbb{F}_q - \{0\}$, there exist $(q+1)$ of α such that $N(\alpha) = \alpha^{q+1} = \gamma$. For any $\gamma \in \mathbb{F}_q$, there exist q of β such that $Tr(\beta) = \beta^q + \beta = \gamma$. These imply that $|S|$ for $S = \{(\alpha, \beta) | N(\alpha) = Tr(\beta)\}$, we have $|S| = q^3$. If we pick message space to be the set of degree q polynomials in (x, y), we can get a code of $\left[q^3, \binom{q+2}{q}, q^3 - q(q+1)\right]_{q^2}$, where the distance can be shown using the Bezout's theorem.

4.4 Garcia-Stichtnoth Construction

Consider $(m+1)$-variate functions such that

$$S = \left\{ (x_1, ..., x_{m+1}) \middle| Tr(x_2) = \frac{N(x_1)}{Tr(x_1)}, ..., Tr(x_{i+1}) = \frac{N(x_i)}{Tr(x_i)}, ... \right\} \subsetneq \mathbb{F}_q^{m+1}, \tag{22}$$

where $Tr(x_{i+1}) = N(x_i)/Tr(x_i)$ is a bivariate equation relating x_i to x_{i+1} and meaningful only if $Tr(x_i) \neq 0$ and we can easily see that $Tr(x_i) \neq 0 \Rightarrow N(x_i) \neq 0$. Thus, we can see that

$$|S| \geq (q^2 - q)q^m, \tag{23}$$

because we choose initial x_1 so that $Tr(x_1) \neq 0$ and for any x_i, we have q choices of x_{i+1}. We take the basis as the set of polynomials in $x_1, ..., x_{m+1}$ of degree q in each. There is $(m+1)$-variate Bezout theorem which says that if we have $(m+1)$ polynomials, the number of their common zeros is small. Applying this, we can get the distance of this code.

5 Questions

We don't have any intuitive explanation why we can have infinite families satisfying:

$$R + \delta \geq 1 - \frac{1}{\sqrt{q}}. \tag{24}$$

We don't know whether there exist any codes that can achieve

$$R + \delta \geq 1 - \frac{1}{\text{ploy}(q)}. \tag{25}$$

We also have some open questions about the Plotkin bound

$$R + \delta \geq 1 - \frac{1}{\sqrt{q}}. \tag{26}$$

Can it be improved? What does it imply to binary codes. Can binary codes be concatenated to make interesting codes?