CS 229r Essential Coding Theory Feb 26, 2020

Lecture 9

Instructor: Madhu Sudan *Scribe: Abishrant Panday*

1 Non Algebraic Geometry Codes

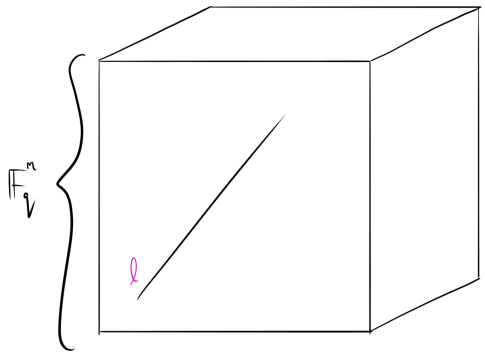
Most of the error correcting codes we've considered so far have very large alphabet size, with none getting down to constant sized alphabets. We want to utilize the algebraic approach (evaluating functions on points), but with this approach, a long code requires lots of points, which requires us to have a growing number of variables over a constant-sized field.

1.1 Multivariate Polynomials

As an aside, before getting into this we note that repetition codes are bad because we have to look at two locations but only get out one symbol's worth of information.

Consider multivariate polynomials with $r < q$ and m large. To this end, take an $m-$dimensional box with $q-$points along each axis. Here, our encoding of a message consists of taking some polynomial and evaluating on the entire space. However, this contains quite a few redundancies. For example, consider lines

$$l = \{\alpha + t\beta | t \in \mathbb{F}_q, \alpha, \beta \in \mathbb{F}_q^m\}$$



Look at some polynomial $P \in \mathbb{F}_q[x_1, \ldots, x_m]$ with $\deg(p) \leq r$ and consider what happens when we restrict to our line l. We get $P|_l = f(t)$ where f is a polynomial of degree $\leq r$, since we're simply doing a linear substitution of each variable by a linear function of t. Then, if we know the value of f at $r+1$ locations on the line, P is determined and all other points are redundant. Similarly, if we had $r = \frac{q}{2}$, then our redundancy would be half of the line l, since it wouldn't give any extra information. We thus want a way to avoid these local redundancies brought about by multivariate polynomials.

The goal with algebraic geometry codes is thus to find some dense-enough curve in this space that doesn't intersect any one line, circle, etc. often, which will reduce the amount of local redundancy.

2 The Algebraic Geometry Approach

We want to find a nice subset $S \subseteq \mathbb{F}_q^n$ such that low degree polynomials or some other class of functions

1. maintain "global" distance
2. eliminate "local" redundancies

2.1 Goppa's Idea

Let S be zeroes of multivariate polynomials

$$P_1(x_1, \ldots, x_m), \ldots, P_{m-1}(x_1, \ldots, x_m)$$

Which forms an $m-1$ dimensional manifold. Then, we have that

$$S = \{\alpha | P_1(\alpha) = \cdots = P_{m-1}(\alpha) = 0\}$$

Ideally, we want $|S|$ to be large (to grow with m and ideally with $\exp(m)$) and for there to be a nice set of functions on S that form a linear space and have few zeroes on S.

2.2 A Simple Example

Consider an alphabet $\mathbb{F}_{q^2}$ where q is prime. Let $S \subseteq \mathbb{F}_{q^2} \times \mathbb{F}_{q^2}$. We want some polynomial equation with lots of zeroes but not too high of a degree. Define

$$S = \{(\alpha, \beta) | \alpha, \beta \in \mathbb{F}_{q^2}, Tr(\alpha) = N(\beta)\}$$

Here, we have that the trace is the sum of Galois conjugates of z: $Tr(z) = z + z^q$ and the norm is the product of Galois conjugates of z: $N(z) = z^{q+1}$, both of which we can view as functions on $\mathbb{F}_{q^2}$:

$$Tr, N : \mathbb{F}_{q^2} \to \mathbb{F}_{q^2}$$

Exercise 1. *Show that* $\forall \alpha \in \mathbb{F}_{q^2}$, $Tr(\alpha)^q = Tr(\alpha)$ *and* $N(\alpha)^q = N(\alpha)$. *In addition, show that* $Tr(\alpha)^q = Tr(\alpha) \mod x^q - x$ *and* $N(\alpha)^q = N(\alpha) \mod x^q - x$.

From this, we see that both the trace and norm are maps to $\mathbb{F}_q$:

$$Tr, N : \mathbb{F}_{q^2} \to \mathbb{F}_q$$

Exercise 2. *Show that* $Tr(x + y) = Tr(x) + Tr(y)$ *and* $N(xy) = N(x)N(y)$.

Putting all of this together, we get that the trace and the norm satisfy:

$$Tr : q \to 1 \text{ map}$$

$$N : q + 1 \to 1 \text{ map on nonzero elements}$$

Based on this, we see that the size of S must satisfy $|S| = q^3$. To see this, note that if we fix a β, we get some value of $N(\beta)$, which is mapped to by q α and vice versa.

2.3 General Family of Functions

Fix r and consider all polynomials of degree $r \leq q$. Suppose $\deg(P) \leq r$. How large is the set

$$\{(\alpha, \beta) | \alpha, \beta \in SP(\alpha, \beta) = 0\}$$

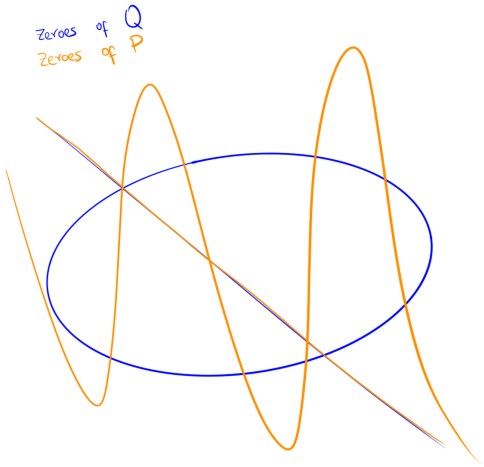
First, we note that $\dim(\text{polynomials of degree} \leq r) = \binom{r+1}{2}$. Define $Q(x, y) \triangleq Tr(x) - N(y)$. Thus, we can rewrite our set as

$$|\{(\alpha, \beta) | \alpha, \beta \in SP(\alpha, \beta) = 0\}| = |\{(\alpha, \beta) \in \mathbb{F}_{q^2}^2 | P(\alpha, \beta) = Q(\alpha, \beta) = 0\}|$$

This is a pretty common problem in algebra/geometry asking how many common zeros a set of polynomials can have.

Theorem 3 (Bezout's Theorem (in plane))**.** *For two plane algebraic curves P, Q with $\deg(P) \leq D_1$ and $\deg(Q) \leq D_2$ and $P, Q \in \mathbb{F}[x, y]$ then either P, Q have a common factor or the number of common zeroes $\leq D_1 \cdot D_2$.*

To see why we have a common factor issue with Bezout's Theorem, consider $P(x, y) = A(x, y) \cdot (x + y)$ and $Q(x, y) = B(x, y) \cdot (x + y)$. If we have P, Q as shown below, then we would have an infinite number of common zeroes.



Exercise 4. *Show that in our case P, Q have no common factors.*

If there are no common factors, then the number of common zeroes is at most $r(q+1)$, giving us the bound

$$|\{(\alpha, \beta) \in \mathbb{F}_{q^2}^2 | P(\alpha, \beta) = Q(\alpha, \beta) = 0\}| \leq r(q+1)$$

In order to compare different codes along the same 'axis,' we can concatenate with the best binary code and consider that error correcting code. In our case, as seen in [Ben-Aroya and Ta-Shma], concatenating with the appropriate binary codes gives a code of distance $(\frac{1}{2} - \varepsilon)n$ and dimension k with

$$n = O\left(\frac{k^{5/4}}{\varepsilon^{5/2}}\right)$$

In contrast, doing Reed-Solonom + binary code would give us $n = \frac{k}{\varepsilon^2}$. The best code that exists is, of course, $\frac{k}{\varepsilon^2}$.

2.4 Tsfasman-Vladut-Zink's Construction

Their construction came up with polynomials $P_1, \ldots, P_{m-1}$ and a basis of functions f that gave a resulting code of

$$\left[n, k, (n-k) - \frac{n}{q-1}\right]_{q^2}$$

The best we could get is $[n, k, n-k]$ over any alphabet. In the original construction of Tsfasman, Vladut, and Zink it wasn't clear that you could encode, decode from these codes efficiently.

2.4.1 Comparison With Random Codes

If we took codes over some $q-$ary alphabet randomly or greedily, etc. we would get a code with distance $[n, k, (n-k) - \frac{n}{\log(q)}]_{q^2}$. When q gets large enough, we thus have that our code is better than random codes. In literature, we have the magic number $q \geq 49$, where this beats the GV bound.

2.5 A New Family of Curves

The family of curves suggested by [Garcia, Stichtenoth] constructs $P_1, \ldots, P_{m-1}$ over $\mathbb{F}_{q^2}$ where

$$P_i(x_i, x_{i+1}) = N(x_i) - Tr(x_i)Tr(x_{i+1})$$

Exercise 5. *Prove that the number of zeroes is $q^{m+1} + o(q^{m+1})$.*

We can extract a sequence of functions $f_1, \ldots, f_i, \ldots$ not necessarily linearly independent that map from the zeroes of the set of polynomials P_i with the following properties:

1. $\dim(span(f_1, \ldots, f_i)) \leq i - g$ where $g = \frac{n}{q-1}$ (g refers to the genus of the curve).
2. The number of zeroes of $f \in span(f_1, \ldots f_n)$ is at most i.
3. $f_i \cdot f_j \in span(f_1, \ldots, f_{i+j})$.

Based on the first two properties, we can construct a $[n, k, (n-k) - \frac{n}{q-1}]_q$ code.

3 Reed-Solomon Decoding

Fix some $\mathbb{F}_q, n \leq q, k \leq n$ and a sequence of n distinct points $(\alpha_1, \ldots, \alpha_n) \in \mathbb{F}_q$ that we evaluate the message polynomial on. Define the message polynomial $M(x) \triangleq \sum_{i<k} c_i x^i$. If we send the sequence $(M(\alpha_1), \ldots, M(\alpha_n))$ over a noisy channel, we receive $(\beta_1, \ldots, \beta_n)$. We're promised that the number of errors is small:

$$\{i : \beta_i \neq M(\alpha_i)\} \leq e$$

More formally, we're given

$$q, n, k, e, \{\alpha_1, \beta_i\}_{i=1,\ldots,n}$$

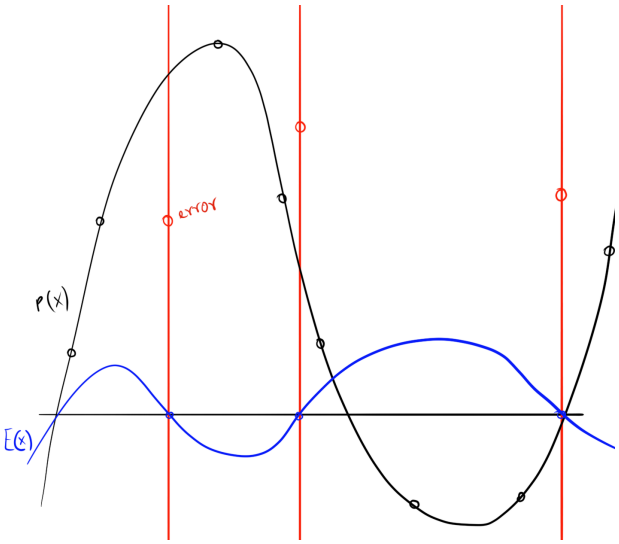
And we want to find M (if one exists) such that $\deg(M) < k$ and $\{i : \beta_i \neq M(\alpha_i)\} \leq e$. In particular, consider a bunch of points in $2-d$ space. We want to find a high-degree polynomial p that passes through a good number of these points. The points it doesn't pass through are errors. We care about scenarios where $\deg(p)$ is linear in n and the fraction of errors is also linear in n. Thus, you can't enumerate all possible polynomials, errors without an exponential search. Algorithms for this decoding include ones by Peterson '60, Berlekamp-Massey '70s, and Berlekamp-Welch '86.

3.1 Peterson's Algorithm

We define an *Error-Locator Polynomial* in x which is zero whenever there is an error:

$$E(x) = \Pi(x - \alpha_i), i \in S \triangleq \{i : \beta_i \neq M(\alpha_i)\}$$

We can visualize the problem with the following diagram:



We note that E satisfies $E(\alpha_i)M(\alpha_i) = E(\alpha_i)\beta_i$ To see this, note that in general $M(\alpha_i) = \beta_i$ and when $M(\alpha_i) \neq \beta_i$ then $E(\alpha_i) = 0$.

Define $N(x) = E(x)M(x)$. Thus, we have that

$$N(\alpha_i) = E(\alpha_i)M(\alpha_i) = E(\alpha_i)\beta_i$$

$$\deg(E) \leq e, \deg(N) \leq k + e$$

This is now a linear system on coefficients of N, E.

Claim 6. *A solution exists.*

This is true by construction.

Claim 7. *If we find a solution (N', E'), then $\frac{N'}{E'} = \frac{N}{E} = M$.*

Proof. The statement is equivalent to saying $N'(x)E(x) = N(x)E'(x)$. To see this, note that we have

$$E(\alpha_i)\beta_i = N(\alpha_i), E'(\alpha_i)\beta_i = N'(\alpha_i) \Longrightarrow N'E\beta_i = NE'\beta_i$$

We can divide through for $\beta_i \neq 0$ and if it equals zero then $N(\alpha_i) = 0$ by definition.

We now consider the number if i such that $N'(\alpha_i)E(\alpha_i) = N(\alpha_i)E'(\alpha_i)$:

$$|\{i|N'(\alpha_i)E(\alpha_i) = N(\alpha_i)E'(\alpha_i)\}|$$

This set has size $\geq k + 2e$. Moreover, the degree of $N'E$ and NE' are polynomials of degree $\leq k + 2e$. Thus, as long as $n = k + 2e$, we have enough points to make the two sides agree. This is true by our assumption above on k, e. □