proof relax relax

6.895 Essential Coding Theory	October 27, 2004
Lecture 14	
Lecturer: Madhu Sudan	*Scribe: Alexey Spiridonov*

1 Motivation

Today's lecture's goal is to describe, informally, a family of codes that achieve better-than-random performance. This lecture does not describe these codes either accurately or completely; much extra reading is necessary to understand this material. While they are also incomplete, it may be helpful to refer to Jonathan Kelner's notes from 2002 [1].

In a previous lecture, we have shown that the Gilbert-Varshamov bound (obtained by a random construction) guarantees the existence of a family of codes such that

$$\delta = \left(1 - O(\frac{1}{\log_2 q})\right) - R.$$

The Singleton bound, on the other hand, tells us that every family of codes is limited by $\delta \le 1 - R$. Therefore, the above construction approaches the optimum as q grows. However, Plotkin's bound restricts the rate at which any family of codes can approach the Singleton bound:

$$R + \frac{q}{q-1}\delta \le 1 \Rightarrow R + \delta \lesssim 1 - \frac{1}{q}.$$

The Gilbert-Varshamov construction requires an alphabet exponentially large in n in order for the distance to have be $\frac{1}{O(n)}$ away from Singleton. (unlike Plotkin, which permits a linear dependence). The nonconstructive nature of the Gilbert-Varshamov bound aside, this slow convergence makes it impractical to get codes that are very close to the Singleton bound.

The codes in this lecture are $[n, k, (1 - \frac{1}{\sqrt{q}-1})n - k]_q$ codes; now, for $\frac{1}{O(n)}$ convergence, we only need an alphabet quadratic in n; this is much better. It is an open problem whether the "defect" in the distance can be improved to $\frac{1}{q}$ to make Plotkin's bound tight.

2 Algebraic Geometry Codes

For the rest of the lecture, we will be working over the field $\mathbb{F}_q$, with q a prime power. The codes we'll construct derive from Reed-Solomon (RS) codes, and share the essential mechanics. Our message space will be a subspace $\mathcal{M}$ of the m-variable polynomials of $\mathbb{F}_q$: $\mathcal{M} \subseteq \mathbb{F}[x_1, \ldots, x_m]$ (subspace means that $\mathcal{M}$ is closed under addition and multiplications by scalars from the field). We will encode a polynomial by evaluating it at a subset S of points of $\mathbb{F}_q$ (rather than the whole space, as in RS codes).

In order to get an efficient code, we want to make $\dim \mathcal{M}$ relatively larger, $|S|$ relatively smaller, and we are also looking to maximize the code's distance. Since it's a linear code, rather than look for the minimum number of places two polynomials' encodings differ, we can get the distance from the maximum number of zeros any $p \in \mathcal{M}$ can have on S.

These codes are called "alebgraic geometry codes" because the sets S we will pick will be *varieties,* objects that algebraic geometry studies. A *variety* $V(p_1, \ldots p_k) \subseteq \mathbb{F}_q^m$ is the set of common zeros of the polynomials p_i:

$$V(p_1, \ldots, p_k) = \{x \in \mathbb{F}_q^m : p_i(x) = 0 \forall 1 \le i \le k\}.$$

3 A Concrete Example

We will make a $[19, 6, 13]_{13}$ code. With a Reed-Solomon code, we would be able to get $[19, 6, 14]_{19}$ which gets us 1 extra distance at the expense of a larger alphabet. Neither is unambiguously better – it's a tradeoff.

First off, we choose $q = 13$ and $m = 2$. We let $\mathcal{M} = \text{span}\{1, x, y, x^2, xy, y^2\} \subset \mathbb{F}_{13}[x, y]$ be the 2-variable monomials of degree ≤ 2. Take $S = V(R)$, the set of zeros of the polynomial

$$R(x, y) = y^2 - 2(x - 1)x(x + 1).$$

Since the domain of the polynomial has just $|\mathbb{F}_{13}^2| = 169$ points, one can easily see by enumeration that the $|S| = 19$. There are ways to speed up the computation of these roots, but in most applications, the search space is small, and the search only needs to be done once.

So, in our code, $k = \dim \mathcal{M} = 6$ and $n = |S| = 19$. It remains to find the minimum distance d. It is achieved by a non-zero polynomial $p \in \mathcal{M}$ which has the most roots in S. In other words, we want to find $d = n - \max |V(p, R)|$: the maximum number of zeros that $p \in \mathcal{M}$ can have in common with R without being identically zero. Since we want d maximized, a scenario we want to avoid having all zeros of p fall in S. This would happen, for instance, if p divided R. Luckily, R doesn't factor (is irreducible). Intuitively, if $0 \neq f$ were to match a lot of zeros of R, it would need to approximate it well. However, polynomials in f have degree at most 2, while R has degree 3, so this shouldn't happen if $p \nmid R$. The following theorem makes this intuition precise.

Theorem 3.1 (Bezout). *Let $A(x, y)$ be a polynomial of degree d_A, and $B(x, y)$ be a polynomial of degree d_B (both nonzero). If $|V(A, B)| \geq d_A d_B$, then A and B have a nontrivial common factor.*

Proof. We only sketch an idea for a proof. This might not be the best approach for turning it into a formal argument; it's main intention is to give a flavor of the subject. We want to compute a quantity called the *resultant* polynomial with respect to y:

$$r(x) = p(x, y)A(x, y) + q(x, y)B(x, y),$$

where p and q are polynomials in x, y chosen so that r does not depend on y. We can pick find such an $r(x) \not\equiv 0$ of degree $\leq d_A d_B$, unless A and B have a common factor containing y. Now, if we pretend all common roots of A and B have distinct x values, then there can't be more than $d_A d_B$ of them (otherwise the resultant would be zero, which implies that both original polynomials are zero); from this, it can be concluded that A and B cannot have zeros with $d_A d_B$ distinct x coordinates. Since A and B don't have a common factor, one should always be able to find a change of coordinates ($u = ax + by$, $v = cx + dy$) to get all roots to have distinct coordinates. Thus, we would obtain the required bound on $V(A, B)$. $\square$

From the theorem, it follows that no $p \in \mathcal{M}$ has more than 6 roots in S, so the minimum distance is $d = 19 - 6 = 3$. Thus, we have the promised $[19, 6, 13]_{13}$ code.

4 General Construction

The essential ideas about constructing algebraic geometry (AG) codes in the previous sections are as follows:

1. We work over a vector space $\mathbb{F}_q^m$ with q a prime power. Take our message space $\mathcal{M}$ to be a subspace of the m-variable polynomials on $\mathbb{F}_q^m$. Then, our message words can be viewed as elements of $\mathbb{F}_q^{\dim \mathcal{M}}$ (coordinates of $v \in \mathcal{M}$ in some basis), giving $k = \dim \mathcal{M}$.

2. To send $p \in \mathcal{M}$, we send its the result of its evaluation on a special subset of $\mathbb{F}_q^m$, a variety $S = V(R_1, \ldots, R_k)$. Then, $n = |S|$.

3. We can find the minimum distance by an algebraic method using resultants. Our selections of $\mathcal{M}$ and S are made to get the best distance we can.

Constructions for the first family of such codes came riding on discoveries in algebraic geometry and number theory, and were very complex (due to Tsafsman, Vladuts, and Zink). Thereafter, Garcia and Stichtenoth made a considerably simplified construction. We present a derivative of this construction described by Pellikaan, Stichtenoth, and Torres. Shum recently discovered an $\tilde{O}(n^2)$[1] for both a encoding and decoding algorithm. [2]

For this construction, we will take $q = p^2$, p prime. We will use two functions $\mathbb{F}_q \to \mathbb{F}_p$: the trace Tr and the norm N, defined as follows:

$$N(x) = x^{p+1}; \operatorname{Tr} x = x^p + x.$$

The fact that both of these functions map from $\mathbb{F}_q$ to the subfield $\mathbb{F}_p$ is a simple exercise. In addition, the trace is linear; this implies that it is a "$p \to 1$" map. In other words, the inverse image of any point $x \in \mathbb{F}_p$ consists of exactly p points.

To construct $S = S_m$, label the coordinates of $\mathbb{F}_q^m$ as $x_1, \ldots, x_m$. Let $R_i(x_i, x_{i+1}) = \operatorname{Tr} x_{i+1} - \frac{N(x_i)}{Tr(x_i)}$[3], and then we can define $S_m = V(R_1, \ldots, R_{m-1})$. Next, we inductively derive a lower bound for S in terms of q and m. The claim is that $|S_i| \geq p^i(p-1)$. Instead of dealing with S', we will work with a subset defined so:

$$S_{i+1} \supseteq \{\mathbf{x} \in \mathbb{F}_q^m : \operatorname{Tr} x_j \neq 0, \operatorname{Tr} x_{j+1} = \frac{N(x_j)}{\operatorname{Tr} x_j} \forall 1 \leq j \leq i+1\} = S_i'.$$

In the base case, $m = 1$, we just have the requirement $\operatorname{Tr} x_1 \neq 0$; since the trace is $p \to 1$, the size of the inverse image of 0 is p, leaving $p(p-1)$. Compared to S_i, S_{i+1} has an extra variable x_{i+1}, and the constraints $\operatorname{Tr} x_{i+1} \neq 0$ and $\operatorname{Tr} x_{i+1} = \frac{N(x_i)}{\operatorname{Tr} x_i}$. For any fixed assignment of $(x_1, \ldots, x_i)$, there are p values of x_{i+1} that satisfy both of the constraints. By induction, the claim follows.

One might wonder why we obtained a lower bound rather than an upper bound (since we said we're trying to minimize $|S|$, relatively). However, given this lower bound, we will now produce the maximal message size and distance to match, and everything will work.

We will pick some polynomials as a basis for $\mathcal{M}$ (although we won't specify what they are exactly). Call this basis $p_1, \ldots, p_k \in \mathbb{F}_q[x_1, \ldots, x_m]$; we will take them so that they have some helpful properties. We will associate with every $p \in \mathbb{F}_q[x_1, \ldots, x_m]$ an order $ord_S(p)$ with respect to S that behaves like a degree of a polynomial. In particular,

$$\begin{aligned} ord_S(P+Q) &\leq \max(ord_S(P), ord_S(Q)) \\ ord_S(PQ) &= ord_S(P) + ord_S(Q) \\ \#\text{zeros of } P \text{ on } S &\leq ord_S(P), \text{ if } P \neq 0. \end{aligned}$$

A possible construction for this order is to take the resultant of P together with all $R(x_i, x_{i+1})$.

I believe the following few paragraphs contain some incorrect statements.

The order permits us to make the p_i quite structured. For instance, if we have p and q linearly independent, of the same order, one can clearly[4] replace them by p', q' (having the same span) such that $ord_S(p') = ord_S(p)$ and $ord_S(q') < ord_S(q)$. Thus, we can take our basis elements in increasing order, and the lower bound on $|S| = n' \geq p^m(p-1)$ permits us to take them so that

$$ord_S(p_i) \leq i + p^m.$$

[1] $\tilde{O}(n^2) = O(n^2 \log^{O(1)} n)$ is just a way to neglect a constant log power.

[2] Thanks are also due to Sergey Yekhanin for his help in understanding this construction.

[3] It's not clear that this is a polynomial, but we treat it as such for the purposes of the construction.

[4] If one thinks of polynomial degrees, this statement is trivial. Since our orders are defined to behave exactly the same, so is this.

So, the maximum total order t we could end up reaching if we take a basis of k elements is $k + p^m$.

... somehow make conclusion that we can get $[n', k, n' - t]_q$ code.

Note that although our n' isn't well-defined, it has a lower bound, so we can throw out some coordinates to get a $[n, k, n - t]_q$ with $n = p^m(p - 1)$. What is t going to be?

5 Decoding

This is very brief summary of the approach to decoding such codes. It turns out that the abstract Welch-Berlekamp construction can be adapted to this case. The key idea is to use a prefix $p_1, \ldots .p_j$ of the basis sequence as the error-locator function. The properties of the order allow us to proceed smoothly through the construction.

References

[1] Jonathan Kelner, Algebraic Geometry Codes: Class notes for 6.897, 2002. Available at `http://theory.lcs.mit.edu/~madhu/FT02/scribe/lect07.pdf`.