03683072: Error Correcting Codes. October 2017 – Lecture 1

First Definitions and Basic Codes

Amnon Ta-Shma and Dean Doron

1 Error Correcting Codes Basics

Definition 1. *An* $(n,K,d)_q$ code *is a subset of* $\mathbb{F}_q^n$ *of size* K *where every two distinct codewords have Hamming distance at least* d*.* d *is called the code's* distance*. An* $[n,k,d]_q$ code *is a* linear *subspace of* $\mathbb{F}_q^n$ *of dimension* k *that is an* $(n,2^k,d)_q$ *code.*

Claim 2. *Let* $\mathcal{C}$ *be a linear code of distance* d*. Then:*

1. *d equals the minimum weight of the nonzero codewords of $\mathcal{C}$.*
2. *$\mathcal{C}$ can detect $d-1$ errors and correct $\left\lfloor \frac{d-1}{2} \right\rfloor$ errors.*

The two most common ways to present an $[n,k]_q$ code are with either an $n \times k$ generating matrix or an $n \times (n-k)$ parity-check matrix.

Definition 3. *A* generator matrix *for an* $[n,k]_q$ *code* $\mathcal{C}$ *is any matrix over* $\mathbb{F}_q$ *whose columns form a basis for* $\mathcal{C}$*. That is,* G *is a generating matrix for* $\mathcal{C}$ *if* $\mathsf{Image}(G) = \left\{Gx \mid x \in \mathbb{F}_q^k\right\} = \mathcal{C}$*.*

Definition 4. *An* $n \times (n-k)$ *matrix* H *over* $\mathbb{F}_q$ *is a* parity-check matrix *of an* $[n,k]_q$ *code* $\mathcal{C}$ *if* $\left\{y \in \mathbb{F}_q^n \mid H^T y = 0\right\} = \mathcal{C}$*.*

Claim 5. *Let* $\mathcal{C}$ *be an* $[n,k]_q$ *code with a generating matrix* G*.* H *is a parity-check matrix for* G *iff* $H^T G = 0$ *(0 here is the zero matrix) and* $rank(H) = n-k$*.*

In particular, if G is of the form $\begin{pmatrix} I_k \\ A \end{pmatrix}$ then $H = \begin{pmatrix} -A^T \\ I_{n-k} \end{pmatrix}$ is a parity-check matrix for $\mathcal{C}$.

Claim 6. *Let* $\mathcal{C}$ *be an* $[n,k]_q$ *code with a generating matrix* G *and parity-check matrix* H*. Then the (minimum) distance of* $\mathcal{C}$ *is the minimum number* d *such that every* $d-1$ *rows of* H *are linearly independent while there exist* d *rows that are linearly dependent.*

Clearly, H is a generator matrix of *some* code. This code is called the *dual* of $\mathcal{C}$ and denoted $\mathcal{C}^\perp$. Stated differently:

Definition 7. *Let* $\mathcal{C}$ *be an* $[n,k]_q$ *code. The* dual code *of* $\mathcal{C}$ *is* $\mathcal{C}^\perp = \left\{y \in \mathbb{F}_q^n \mid \langle x,c \rangle = 0 \text{ for all } c \in \mathbb{C}\right\}$*.*

Notice that $\mathcal{C}^\perp$ is an $[n,n-k]_q$ code.

1.1 The Hamming Code

Our first example is the Hamming $[7,4]_2$ code. It encodes four bits into seven bits by adding three parity bits. Thus, it can detect and correct single-bit errors. A generating matrix is:

$$G = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 \end{pmatrix}$$

And a parity-check matrix is:

$$H = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

The parity-check matrix of a *general* Hamming code over $\mathbb{F}_q$ is constructed by listing all rows of length r that are pairwise independent over $\mathbb{F}_q$. For $\mathbb{F}_2^r$, this corresponds to simply listing all nonzero binary vectors of length r. This clearly gives a $[2^r-1, 2^r-r-1, 3]_2$ code.

Write $d = 2e+1$. An $[n,k,d]$ code $\mathcal{C}$ is said to be *perfect* if for every possible word w there is a unique codeword $c \in \mathcal{C}$ in which at most e letters of c differ from the corresponding bits of w. That is, if $\sum_{i=0}^{e} \binom{n}{i}(q-1)^i = q^{n-k}$. Clearly, the binary $[n = 2^r-1, k = 2^r-r-1, d = 3]_2$ Hamming code satisfies

$$1 + n = 2^r = 2^{n-k},$$

so it is perfect.

For every code, perfect or not, linear or not, $q^k \sum_{i=0}^{e} \binom{n}{i}(q-1)^i \leq q^n$, because the balls of radius e around codewords must be disjoint (why?). In a perfect code these balls perfectly cover the whole space $\mathbb{F}_q^n$. Thus, these codes (if exist) are perfectly optimal and have the maximum possible number of codewords among all codes of length n and distance d.

To decode the $[n = 2^r-1, k = 2^r-r-1, 3]_2$ Hamming code we introduce the notion of *syndrome* decoding. A syndrome of $y \in \mathbb{F}_2^n$ is simply $s = H^T y$, so the codewords of $\mathcal{C}$ are precisely the vectors whose syndrome is 0. Also, $c_1 - c_2 \in \mathcal{C}$ if and only if $H^T c_1 = H^T c_2$. Consider the following procedure for the Hamming code. Given a received word $y \in \mathbb{F}_2^n$:

1. Compute $s = H^T y \in \mathbb{F}_2^{n-k}$. If $s = 0$, y is a codeword and return y.

2. Otherwise, there must be $e \in \mathbb{F}_2^n$ of weight 1 such that $s = H^T e$ (why?). If we let $e_j \in \mathbb{F}_2^n$ denote the vector with 1 at the j'th coordinate and zero otherwise ($1 \leq j \leq n$), then $H^T e_j$ is the j'th column of H^T. As the columns of H^T are all distinct, we compare the syndrome s to the columns of H^T, and find the column j where they are equal. We then output $c = y - e_j$.

If we enumerate the $n = 2^r - 1$ rows of H (i.e., the columns of H^T) such that they count in binary from 1 to n, then the value of s in binary immediately tells how to correct the error: value 0 means no error, and value j $(1 \leq j \leq n)$ means $c = y - e_j$.

Decoding here is especially easy as we only need to correct one error. Decoding general codes is much more challenging.

1.2 The Hadamard Code

The Hadamard code is a binary code that maps messages of length k to codewords of length 2^k in the following way. For a message $x \in \{0,1\}^k$, every coordinate $y \in \{0,1\}^k$ is computed by $\langle x, y \rangle$ (modulo 2), so

$$\mathsf{Had}(x) = \left(\langle x, y \rangle\right)_{y \in \{0,1\}^k}.$$

Check that the Hadamard code is linear.

We claim the distance of the Hadamard code is 2^{k-1}. Furthermore, we claim *every* nonzero codeword has weight *exactly* 2^{k-1}. To see this, let $x \in \{0,1\}^k$ be a nonzero message. Then, the weight of $\mathsf{Had}(x)$ is given by $\Pr_y[\langle x, y \rangle = 1]$. Prove that this value is exactly $\frac{1}{2}$.

We remark that because there are so few codewords, decoding can be done in polynomial time by going brute force over all the $2^k = n$ codewords.

To summarize, the Hadamard code is $[n = 2^k, k, 2^{k-1}]_2$ code. Thus, the Hadamard code has high distance (relative distance=distance/length=1/2) but very low dimension, compared with the $[n = 2^r - 1, k = 2^r - r - 1, d = 3]_2$ Hamming code that has very high dimension but very low distance. What we shall seek next are codes that lie in between, for example codes with constant relative distance (i.e., distance/length) and constant relative rate (which is, dimension/length).

2 Basic Algebra

1. Groups, Fields.
2. $\mathbb{F}_2$, $\mathbb{F}_p$, $\mathbb{F}_7^\star$, $\mathbb{F}_p^\star$.
3. Rings, Euclidean rings, Unique factorization domains.
4. The ring of polynomials. GCD. Division with reminder.
5. $\mathbb{F}_{p^k} \cong \mathbb{F}_p[x] \bmod E(x)$ for an irreducible E of degree k.
6. Generators.
7. $\mathbb{F}_4$, $\mathbb{F}_4^\star$, $\mathbb{F}_8$, $\mathbb{F}_8^\star$.
8. For $p \in \mathbb{F}[x]$, $p(a) = 0$ implies $x - a \mid p$.
9. For $p \in \mathbb{F}[x]$ of degree k, p has at most k roots in $\mathbb{F}$.

3 The Reed-Solomon Code

Fix a field $\mathbb{F}_q$ of size q with a generator α of $\mathbb{F}_q^\star$. The code $\mathsf{RS} : \mathbb{F}_q^k \to \mathbb{F}_q^n$ corresponds to evaluating all degree $k-1$ polynomial (whose coefficients are given to us as the input message) on all nonzero field elements.[1] That is, $n = q-1$ and

$$\mathsf{RS}(a_0, \ldots, a_{k-1}) \;=\; \left(p_a(\alpha^0), p_a(\alpha^1), \ldots, p_a(\alpha^{n-1})\right),$$

where $p_a(x) = \sum_{i=0}^{k-1} a_i x^i$.

Every nonzero polynomial of degree $k-1$ can have at most $k-1$ zeros in $\mathbb{F}_q$, so the weight of every nonzero codeword is at least $d = n-(k-1) = n-k+1$. Thus, the RS code is an $[n, k, n-k+1]_q$ code.

By inspection, the $n \times k$ generating matrix is given by

$$G \;=\; \begin{pmatrix} (\alpha^0)^0 & (\alpha^0)^1 & \cdots & (\alpha^0)^{k-1} \\ (\alpha^1)^0 & (\alpha^1)^1 & \cdots & (\alpha^1)^{k-1} \\ \vdots & \vdots & \vdots & \vdots \\ (\alpha^{n-1})^0 & (\alpha^{n-2})^1 & \cdots & (\alpha^{n-1})^{k-1} \end{pmatrix}$$

so for $0 \le i \le n-1$ and $0 \le j \le k-1$, $G[i,j] = \alpha^{ij}$. Now, consider the $n \times (n-k)$ matrix

$$H \;=\; \begin{pmatrix} (\alpha^0)^1 & (\alpha^0)^2 & \cdots & (\alpha^0)^{n-k} \\ (\alpha^1)^1 & (\alpha^1)^2 & \cdots & (\alpha^1)^{n-k} \\ \vdots & \vdots & \vdots & \vdots \\ (\alpha^{n-1})^1 & (\alpha^{n-1})^2 & \cdots & (\alpha^{n-1})^{n-k} \end{pmatrix}$$

so for $0 \le i \le n-1$ and $1 \le j \le n-k$, $H[i,j] = \alpha^{ij}$. H is a Vandermonde matrix with the right dimensions, so in order to prove that H is indeed the parity-check matrix of the RS code, it is sufficient to prove that $H^T G = 0_{(n-k)\times k}$. We have that

$$(H^T G)[a,b] = \sum_{k=0}^{n-1} H[k,a]G[k,b] = \sum_{k=0}^{n-1} \left(\alpha^{a+b}\right)^k .$$

a ranges from 1 to $n-k$ and b ranges from 0 to $k-1$, so $1 \le a+b \le n-1$ and the above sums to zero.

Remark 8. *An $[n,k,d]_q$ code that satisfies $d = n-k+1$ (like the RS code) is called an MDS (maximum distance separable) code. Such codes meet the Singleton bound $d \le n-k+1$ which we will see later, and thus have an optimal number of codewords for the given n and d.*

Remark 9. *With 0 as an additional evaluation point, the code becomes an $[q,k,q-k+1]_q$ code. We encourage the reader to find natural generating and parity check matrices for it. Also verify that a dual of such a code is of the same type.*

[1] It is also common to take the zero element as well, however, for reasons that will become clear soon we will not use the zero element as an evaluation point.

4 The Reed-Muller code

We will follow the references in the syllabus (Sudan's Lecture 4). For calculating the distance, we will use the following lemmas:

For a field size larger than the degree, we will use

Lemma 10 (Schwartz-Zippel)**.** *A nonzero polynomial $f \in \mathbb{F}_q[x_1, \ldots, x_n]$ of total degree d is zero on at most $\frac{d}{q}$ fraction of the points in $\mathbb{F}_q^n$.*

For a field size smaller than the degree, we will use:

Lemma 11 ([2], Theorem 19 of Lecture 2.5)**.** *If $f \in \mathbb{F}_q[x_1, \ldots, x_n]$ has individual degree at mot $\ell < q$ in each variable and has total degree $d = \ell k + r$ with $r < \ell$, then it is nonzero on at least $\left(1 - \frac{\ell}{q}\right)^k \left(1 - \frac{r}{q}\right)$ fraction of points in $\mathbb{F}_q^n$.*

5 The Hermitian Code

Up until now, our code's domains were the q-ary hypercube. We can instead pick a subset $S \subseteq \mathbb{F}_q^m$ with some nice *geometric* properties.

We start with the trace and norm functions.

Definition 12. *The functions* $\mathrm{Tr}, \mathrm{N} : \mathbb{F}_{q^r} \to \mathbb{F}_q$ *are given by* $\mathrm{Tr}(x) = \sum_{i=0}^{r-1} x^{q^i}$ *and* $\mathrm{N}(x) = \prod_{i=0}^{r-1} x^{q^i}$.

The image of both functions is in $\mathbb{F}_q$ (this follows easily $\mathbb{F}_q = \{\alpha \in \mathbb{F}_{q^r} \mid \alpha^q = \alpha\}$, and using $\alpha^{q^r} = \alpha$ for every $\alpha \in \mathbb{F}_{q^r}$. Check!). It is easy to check that the trace is additive whereas the norm is multiplicative. Also,

Claim 13. $\mathrm{Tr} : {\mathbb{F}_q}^r \to \mathbb{F}_q$ *is a perfect q^{r-1}-to-1 map, and* $\mathrm{N} : (\mathbb{F}_q^r)^* \to \mathbb{F}_q$ *is a perfect $\sum_{i=0}^{r-1} q^i$-to-1 map.*

The Hermitian code lies on the curve

$$S = \left\{(x, y) \in \mathbb{F}_{q^2}^2 \mid \mathrm{Tr}(x) = \mathrm{N}(y)\right\} = \left\{(x, y) \in \mathbb{F}_{q^2} \mid x^q + x = y^{q+1}\right\}$$

and its message space is given by bivariate polynomials over $\mathbb{F}_{q^2}$ with total degree at most $r \leq q$, so $k = \binom{r+2}{2} \geq \frac{r^2}{2}$. That is, similar to a RM code, we evaluate low-degree, bi-variate polynomials but this time on a curve.

We shall determine the length of the code, $|S|$, and the distance of the code.

Claim 14. $|S| = q^3$.

Proof. For every choice of y (there are q^2 such choices), let $\gamma = \mathrm{N}(y)$. Then, there are q choices of $x \in \mathbb{F}_{q^2}$ satisfying $\mathrm{Tr}(x) = \gamma$, as Tr is a perfect q-to-1 map. □

To determine the distance, we will use Bézout's theorem.

Theorem 15 (Bézout, see e.g., Chapter 7 of [1])**.** *If $\mathbb{F}$ is a field and $f, g \in \mathbb{F}[x, y]$ have no common factors over the algebraic closure, then*

$$|\{(\alpha, \beta) \mid f(\alpha, \beta) = g(\alpha, \beta) = 0\}| \;\leq\; \deg(f) \cdot \deg(g).$$

Lemma 16. *The distance d of the code is at least $q^3 - r(q + 1)$.*

Proof. Let $f \in \mathbb{F}_{q^2}[x, y]$ be of degree at most r. Define $g(x, y) = \mathrm{Tr}(x) - \mathrm{N}(y)$, so S is the zeros of g. Since $\deg(g) = q + 1 > r$ and g is irreducible, g and f have no common factors. By Bézout's theorem, the number of zero evaluation points (that is, (α, β) so that $f(\alpha, \beta) = g(\alpha, \beta) = 0$) is at most $\deg(f) \cdot \deg(g) \leq r(q + 1)$, so $d \geq n - r(q + 1)$. □

In conclusion, the code is a

$$\left[q^3, \frac{r^2}{2}, q^3 - r(q + 1)\right]_{q^2}$$

code.

References

[1] H. Hilton. *Plane algebraic curves.* The Clarendon press.

[2] Madhu Sudan. Algorithmic introduction to coding theory – lecture notes, September 2001.