A Crash Course on Coding Theory

Madhu Sudan

MIT

Disclaimer

This is an opinionated survey of coding theory, unbiased by actual reading of papers.

Trivial Constructions

(Think binary, then generalize)

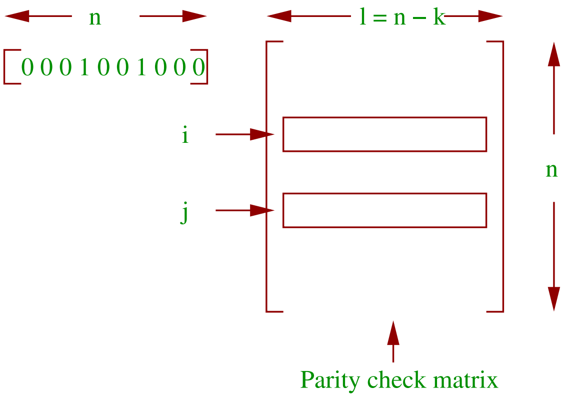
- Trivial code:
  - E is the identity function.
  - Has $n = k, d = 1$.
  - Generalizes to all alphabets!
- Parity code:
  - Append parity of all k bits to message.
  - Gives $n = k + 1, d = 2$.
  - More generally, append sum of the first k letters.

Meet Singleton bound: $k + d \leq n + 1$.

Hamming code

- Historically first (approximately).
- For any l, $[n = (q^l - 1)/(q - 1), n - l, d = 3]_q$ code.
- Rows of parity check matrix H:
  - All non-zero vectors of length l,
  - Scalar multiples removed (say by fixing first non-zero entry to 1).
- Since any two rows of H are linearly independent, distance is greater than 2.

Hamming code (contd).



Hadamard code

$[n = q^l, k = l, d = q^l - q^{l-1}]_q$ code.

(Roughly, the dual of the Hamming code.)

Construction:

- Message: $m = \langle m_1, \ldots, m_k \rangle$
  associated with
  $M(x_1, \ldots, x_k) = \sum_{i=1}^{k} m_i x_i$.
- Encoding: $E(m) = \langle M(x) \rangle_{x \in \Sigma^k}$.
- Distance = Why?

Polynomials over finite fields

Some facts (Fix size of field to q).

- Non-zero deg. $\leq l$ poly. has $\leq l$ zeroes.
  (alt'ly, zero on $\leq l/q$ fraction of inputs.)
- Deg $\leq l$ polys = vector space of dim. $l+1$.
- Non-zero deg. $\leq l$, m-variate, poly.
  zero on $\leq l/q$ fraction of inputs.
- $l < q \Rightarrow$ Deg. $\leq l$, m-variate, polys
  = dim. $\binom{m+l}{l}$ vector space.

Poly facts (contd.)

- Non-zero deg. $\leq l$, m-var., poly. zero on
  $\leq 1 - q^{-(l/(q-1))}$ fraction of inputs.
- Vector space of dimension $\geq \binom{m}{l}$.
- Actual dimension = # of ordered partitions of l into integers from $\{0, \ldots, q-1\}$.

Hadamard codes (contd).

- Codewords are evaluations of degree 1 polynomials over $\mathbb{F}_q$.
- May agree in at most $1/q$ fraction of indices.
- $\Rightarrow$ Distance $\geq q^l - q^{l-1}$.

Reed-Solomon Codes

Reed-Solomon Codes:
$[n, k, n-k+1]_q$ code for $q \geq n$.

- Fix distinct $x_0, \ldots, x_{n-1} \in \Sigma$.
- Message: Coefficients of polynomials $\langle m_0, \ldots, m_{k-1} \rangle \approx M(x) = \sum_{i=0}^{k-1} m_i x^i$
- Encoding: Evaluations of polynomials $\langle M(x_0), \ldots, M(x_{n-1}) \rangle$
- Distance follows from fact on univariate polynomials.

Aside

Standard textbooks give very different presentation of RS codes.

Next few slides explain the connection.

(May skip till "End of Aside")

Primitive roots and Canonical RS code

Defn: w is a primitive nth root of unity if:
$w^n = 1$ but $w^i \neq 1, \quad 1 \leq i < n$.

Fact: $0 < k < n \;\Rightarrow$

$$1 + w^k + (w^k)^2 + \cdots + (w^k)^{n-1} = 0.$$

Proof:
$w^n = 1 \;\Rightarrow\; (w^k)^n = 1 \;\Rightarrow\; (w^k)^n - 1 = 0$
$\Rightarrow\; (w^k - 1) = 0$
or $(1 + w^k + \cdots (w^k)^{n-1}) = 0$.

Defn: Canonical RS code:
Set $x_i = w^i$, $0 \leq i \leq n-1$.

Dual of Canonical RS code

Exploring the dual:
$\langle c_0, \ldots, c_{n-1} \rangle \in \mathrm{RS}^\perp$

$$\Updownarrow$$

$$\sum_{i=0}^{n-1} c_i M(w^i) = 0, \forall \text{ deg. } \leq k-1 \text{ poly. } M$$

$$\Updownarrow$$

$$\textstyle\sum_{i=0}^{n-1} c_i (w^i)^j = 0, \quad \forall 0 \leq j \leq k-1$$

$$\Updownarrow$$

$$(x - w^j) \mid C(x) \triangleq \textstyle\sum_{i=0}^{n-1} c_i x^i$$

Dual of RS code (contd).

- Let $G(x) = \prod_{j=0}^{k-1}(x - w^j)$
- Message: Coeffs of polys
  $\langle d_0, \ldots, d_{n-k-1} \rangle \approx D(x) = \sum_i d_i x^i$
- Encoding: Coeffs of poly
  multiplied with $G(x)$.
  Let $C(x) = G(x) \cdot D(x) = \sum_i c_i x^i$.
  Encoding $= \langle c_0, \ldots, c_{n-1} \rangle$

Alternate description of Dual

Thm: $\mathrm{RS}^\perp = \left\{ \begin{array}{r} \langle M(w^i) \rangle_{i=0}^{n-1} \mid M(0) = 0, \\ \deg(M) \leq n-k \end{array} \right\}$

Proof: Need to verify

$$(x - w^l) \mid \textstyle\sum_{i=0}^{n-1} M(w^i) x^i, \quad 0 \leq l \leq k-1$$

$$\Leftrightarrow \textstyle\sum_{i=0}^{n-1} M(w^i)(w^l)^i$$

$$\Leftrightarrow \textstyle\sum_{i=0}^{n-1} ((w^i)^j)(w^l)^i, \quad 1 \leq j \leq n-k$$

$$\Leftrightarrow \textstyle\sum_{i=0}^{n-1} ((w^i)^j)(w^l)^i$$

$$\Leftrightarrow \textstyle\sum_{i=0}^{n-1} (w^{j+l})^i, \quad 1 \leq l + j \leq n-1$$

Follows from primitivity.

End of Aside

Conclusion: Reed-Solomon codes are (almost) their own duals.

Statement of great importance.

Or as Levin would say ....

Magnitude of importance very large.
Unfortunately sign is negative.

Reed-Muller Codes

Codes based on multiv. polynomials.

$\#$ variables $= m$; degree $\leq r$.

Coding theory favorite: $q = 2$, $[n, k, d]_2$ code

$$n = q^m; k = \left(\binom{m}{r}\right); d = q^{m-r}$$

Complexity th. favorite: $q > r$, $[n, k, d]_q$ code

$$n = q^m; k = \binom{m+r}{r}; d = q^m - rq^{m-1}$$

Latter version:

Larger alphabet; larger distance.

Can also take indiv. degree bounded polys.

Random linear codes

Pick $c_1, \ldots, c_k \in_R \Sigma^n$ and let

$$G = \begin{bmatrix} - & - & c_1 & - & - \\ - & - & c_2 & - & - \\ & & \vdots & & \\ - & - & c_k & - & - \end{bmatrix}$$

Analysis (of Distance):

- For fixed $\langle \alpha_1, \ldots, \alpha_k \rangle \neq \vec{0}$
  $\Pr\left[\alpha G \in B(\vec{0}, d)\right] \leq q^{(H_q(d/n)-1)n}$.
- Thus
  $\Pr\left[\exists \alpha \text{ s.t.} \alpha G \in B(\vec{0}, d)\right] \leq q^{k+(H_q(d/n)-1)n}$.
- Thus if $k/n < 1 - H_q(d/n)$
  then code is $[n, k, d]_q$ code.

Hamming Balls

- Recall $B(\vec{x}, r)$ ball of radius r around $\vec{x}$.
- $V(n, r, q) =$ "volume" of $B(\cdot, r)$ in Σ^n.
- Let $H_q(p)$ be q-ary entropy function.

$$H_q(p) = p \log_q \left(\frac{q-1}{p}\right) + (1-p) \log_q \left(\frac{1}{1-p}\right)$$

Fact:

$$V(n, pn, q) \approx q^{H_q(p)n}$$

Summary

- Reed-Solomon codes are great, but alphabet is too large.
- Hadamard codes are exponentially large but have great distance.
- Random codes are great.
  Achieve $k/n, d/n > 0$ over binary alphabet.

  But non-constructive; non-verifiable;
  non-decodable.

Operations on codes

Can produce codes from other codes by some basic operations.

- Puncturing:
  Throw away column of generator matrix.
  $[n,k,d]_q \quad \rightarrow \quad [n-1,k,d-1]_q$
  Asymptotically weaker.
  (Every linear code is punctured Hadamard code.)

- Pasting:
  Adjoin generators of codes of same dim.
  to get longer code.
  $[n_1,k,d_1]_q \mid [n_2,k,d_2]_q$
  $\rightarrow \quad [n_1+n_2,k,d_1+d_2]_q$
  Asymptotically weaker.

Direct Products

- $[n_1,k_1,d_1]_q \otimes [n_2,k_2,d_2]_q$
  $\rightarrow \quad [n_1n_2,k_1k_2,d_1d_2]_q$

- Let R generate $[n_1,k_1,d_1]$ code.
  Let C generate $[n_2,k_2,d_2]$ code.

- Codewords of $R \otimes C$ are $n_1 \times n_2$ matrices:
  $\left\{C^T X R \mid X \in \Sigma^{k_1 \times k_2}\right\}$

- Columns of tensor are codewords of C.
  Rows of tensor are codewords of R.

- Asymptotically weakening.

Example: tensor product of RS codes, gives bivariate polynomials of degree $k_1 - 1$ in x and $k_2 - 1$ in y.

Concatenation of codes [Forney]

$[n_1,k_1,d_1]_{q^{k_2}} \circ [n_2,k_2,d_2]_q$
$\rightarrow \quad [n_1n_2,k_1k_2,d_1d_2]_q.$

- Compare with Tensor Products!

- Terminology: First code is "outer code"
  Second code is "inner code".

- Encoding:
  Encode message with outer encoder.
  Then encode each letter w. inner code.

- Linearity achieved with care. Outer alphabet must be properly extended from inner alphabet.

Example: RS $\circ$ Hadamard

- Fix k_2.

- Let $n_1 = 2^{k_2}, k_1 = \cdot 5 n_1, q = 2$.

- RS outer code: $[n_1, \cdot 5 n_1, \cdot 5 n_1]_{n_1}$

- Hadamard inner code: $[n_1, k_2, \cdot 5 n_1]_2$

- Concatenate code: $[n_1^2, \cdot 5 k_2 n_1, \cdot 25 n_1^2]_2$

- Let $n = n_1^2$, Use $k_2 = \cdot 5 \log_2 n$
  $[n, \cdot 25\sqrt{n}\log_2 n, \cdot 25 n]_2$ code

- Constant distance, poly rate!
  Good for many complexity th. applications.

Forney Codes

- Concat. RS codes with random linear code.
- At each level code has constant $d/n, k/n$.
- Concat. code has constants for both ratios.

Thm: (Unflattering version). Asymptotically good code can be found in quasi-polynomial time.

Thm: (Flattering version). By using 2 levels of concatenation, asymptotically good code can be found in nearly linear time, with polylog space.

(Not the end of story.)

Justesen Codes

(More "explicit" codes; Nice idea; Exposition due to Zuckerman)

Suppose: Can explicitly describe sample space containing n_1 codes such that all but ϵ fraction of the codes are $[n_2, k_2, d_2]_q$ codes.

Then concatenate codes as follows:

- Encode message using $[n_1, k_1, d_1]_{q^{k_2}}$ code.
- Encode ith letter of result using ith code from sample space.
- Result is a $[n_1 n_2, k_1 k_2, (d_1 - \epsilon n_1) d_2]_q$ code.

Can get asymptotically good code!

Justesen's sample space

- The Wozencraft ensemble.
- Let $n_1 = q^{k_2}$.
- Let $F = GF(q^{k_2})$.
- Message for inner code: $x \in F$.
- α-th code maps $x \mapsto \langle x, \alpha x \rangle$.
- For most α, get $[2k_2, k_2, H_q^{-1}(\frac{1}{2})(2k_2)]_q$ code.
- (I.e., most codes, as good as random code!)

Wozencraft ensemble (contd).

(Ignoring subscript on k_2 below.)

α is d-bad if α-th code not $[2k, k, d]_q$ code.

Claim: # of bad α's is at most

$$V(2k, d, q) \approx q^{H_q(d/(2k)) \cdot (2k)}.$$

Proof:

- If $\alpha_1 \neq \alpha_2$ then intersection of corr. codes is the 0-vector.
- Each bad code must have non-zero vector in $B(\vec{0}, d)$. These must be distinct.
- Thus, at most $V(2k, d, q)$ bad codes.

Further pointers

- Weldon codes: $x \mapsto \langle x, \alpha x, \alpha^2 x, \ldots \rangle$.
- Gets distance arbitrarily close to $1 - \frac{1}{q}$.
- Alternate route: Can apply Zuckerman exposition with 2-level concatenation and random linear codes.
- Sugiyama et al. papers: Get better rates than Weldon.