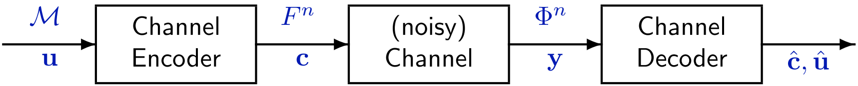
Review of Basic Coding Theory

Gadiel Seroussi

October 12, 2022

Channel Coding



Discrete probabilistic channel: (F, Φ, Prob)

- F: finite *input alphabet*, Φ: finite *output alphabet*
- Prob: conditional probability distribution

$$\text{Prob}\{\, \mathbf{y} \text{ received} \mid \mathbf{x} \text{ transmitted} \,\} \quad \mathbf{x} \in F^m, \quad \mathbf{y} \in \Phi^m, \quad m \geq 1$$

- $\mathbf{u}$: *message word* $\in \mathcal{M}$, set of M possible messages
- $\mathbf{c} \in F^n$: *codeword*, $\mathcal{E} : \mathbf{u} \xrightarrow{1-1} \mathbf{c}$ *encoding*
- $\mathcal{C} = \{\mathcal{E}(\mathbf{u}) \mid \mathbf{u} \in \mathcal{M}\}$ *code*
- $\mathbf{y} \in \Phi^n$: *received word*
- $\hat{\mathbf{c}}, \hat{\mathbf{u}}$: *decoded codeword, message word*, $\mathbf{y} \longrightarrow \hat{\mathbf{c}} \;\; (\longrightarrow \hat{\mathbf{u}})$ *decoding*

Code Parameters

$\mathcal{C} \;=\; \mathcal{E}(\mathcal{M}) \;\subseteq\; F^n, \quad |\mathcal{C}| = M$

- n: *code length*
- $k = \log_{|F|} M = \log_{|F|} |\mathcal{C}|$: *code dimension*
- $R = \dfrac{k}{n}$: *code rate* ≤ 1
- $r = n - k$: *code redundancy*
- We call $\mathcal{C}$ an (n, M) (*block*) *code* over F

The Hamming Metric

- *Hamming distance*

  For single-letters $x, y \in F$: $\mathsf{d}(x,y) = \begin{cases} 0, & x = y, \\ 1, & x \neq y. \end{cases}$

  For vectors $\mathbf{x}, \mathbf{y} \in F^n$: $\mathsf{d}(\mathbf{x},\mathbf{y}) = \sum_{j=0}^{n-1} \mathsf{d}(x_j, y_j)$

  number of locations where the vectors differ

- The Hamming distance defines a *metric*:
  - $\mathsf{d}(\mathbf{x},\mathbf{y}) \geq 0$, with equality if and only if $\mathbf{x} = \mathbf{y}$
  - Symmetry $\mathsf{d}(\mathbf{x},\mathbf{y}) = \mathsf{d}(\mathbf{y},\mathbf{x})$
  - Triangle inequality: $\mathsf{d}(\mathbf{x},\mathbf{y}) \leq \mathsf{d}(\mathbf{x},\mathbf{z}) + \mathsf{d}(\mathbf{z},\mathbf{y})$
- *Hamming weight* $\mathsf{wt}(\mathbf{e}) = \mathsf{d}(\mathbf{e}, \mathbf{0})$ *number of nonzero entries*
- When F is an abelian group, $\mathsf{d}(\mathbf{x},\mathbf{y}) = \mathsf{wt}(\mathbf{x} - \mathbf{y})$

Minimum Distance

- Let $\mathcal{C}$ be an (n, M) code over F, $M > 1$

$$d = \min_{\mathbf{c}_1, \mathbf{c}_2 \in \mathcal{C} \,:\, \mathbf{c}_1 \neq \mathbf{c}_2} \mathsf{d}(\mathbf{c}_1, \mathbf{c}_2)$$

  is called the *minimum distance* of $\mathcal{C}$
- We say that $\mathcal{C}$ is an (n, M, d) code.

Decoding

- $\mathcal{C} : (n, M, d)$ over F, used on channel $S = (F, \Phi, \text{Prob})$
- A *decoder* for $\mathcal{C}$ on S is a function

$$\mathcal{D} : \Phi^n \longrightarrow \mathcal{C}.$$

- *Decoding error probability* of $\mathcal{D}$ is

$$P_{\text{err}} = \max_{\mathbf{c} \in \mathcal{C}} P_{\text{err}}(\mathbf{c}) \ ,$$

where

$$P_{\text{err}}(\mathbf{c}) = \sum_{\mathbf{y} \,:\, \mathcal{D}(\mathbf{y}) \neq \mathbf{c}} \text{Prob}\{ \, \mathbf{y} \text{ received} \mid \mathbf{c} \text{ transmitted} \, \} \ .$$

goal: *find encoders (codes) and decoders that make P_{err} small*

Maximum Likelihood and Maximum a Posteriori Decoding

- $\mathcal{C} : (n, M, d)$, channel $S : (F, \Phi, \text{Prob})$.
- *Maximum likelihood decoder (MLD)*:

$$\mathcal{D}_{\text{MLD}}(\mathbf{y}) = \arg\max_{\mathbf{c}\in\mathcal{C}} \text{Prob}\{\ \mathbf{y} \text{ received} \mid \mathbf{c} \text{ transmitted}\ \},\ \forall \mathbf{y} \in \Phi^n$$

With a fixed tie resolution policy, $\mathcal{D}_{\text{MLD}}$ is well-defined for $\mathcal{C}$ and S.

- *Maximum a posteriori (MAP) decoder*:

$$\mathcal{D}_{\text{MAP}}(\mathbf{y}) = \arg\max_{\mathbf{c}\in\mathcal{C}} \text{Prob}\{\ \mathbf{c} \text{ transmitted} \mid \mathbf{y} \text{ received}\ \},\ \ \forall \mathbf{y} \in \Phi^n$$

But,

$$\begin{aligned} & \text{Prob}\{\,\mathbf{c} \text{ transmitted} \mid \mathbf{y} \text{ received}\,\} \\ & \quad = \quad \text{Prob}\{\,\mathbf{y} \text{ received} \mid \mathbf{c} \text{ transmitted}\,\} \cdot \frac{\text{Prob}\{\,\mathbf{c} \text{ transmitted}\,\}}{\text{Prob}\{\,\mathbf{y} \text{ received}\,\}} \end{aligned}$$

$\Longrightarrow$ MLD and MAP are the same when $\mathbf{c}$ is *uniformly distributed*

MLD on the BSC

- $\mathcal{C} : (n, M, d)$, channel $\mathsf{BSC}(p)$

$$\begin{aligned} &\mathsf{Prob}\{\,\mathbf{y}\ \mathsf{received}|\ \mathbf{c}\ \mathsf{transmitted}\,\} \\ &= \prod_{j=1}^{n} \mathsf{Prob}\{\,y_j\ \mathsf{received}\ |\ c_j\ \mathsf{transmitted}\,\} \\ &= p^{\mathsf{d}(\mathbf{y},\mathbf{c})}(1-p)^{n-\mathsf{d}(\mathbf{y},\mathbf{c})} = (1-p)^n \cdot \left(\frac{p}{1-p}\right)^{\mathsf{d}(\mathbf{y},\mathbf{c})}, \end{aligned}$$

where $\mathsf{d}(\mathbf{y}, \mathbf{c})$ is the Hamming distance. Since $p/(1-p) < 1$ for $p < 1/2$, for all $\mathbf{y} \in F_2^n$ we have

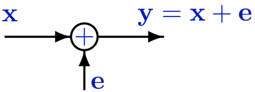
$$\mathcal{D}_{\mathrm{MLD}}(\mathbf{y}) = \arg\min_{\mathbf{c}\in\mathcal{C}} \mathsf{d}(\mathbf{y}, \mathbf{c})$$

$$\boxed{\mathcal{D}_{\mathrm{MLD}} = \textit{nearest-codeword decoder}}$$

- True also for $\mathsf{QSC}(p)$ whenever $p < 1 - 1/q$

Error Correction

$$\mathbf{e} = [0\ldots0,\ \boxed{e_{i_1}},0\ldots0,\ \boxed{e_{i_2}},0\ldots0,\ \boxed{e_{i_t}},0\ldots0]$$



$i_1, i_2, \ldots, i_t$:	*error locations*	$e_{i_1}, e_{i_2}, \ldots, e_{i_t}$:	*error values* $(\neq 0)$

Full error correction: the task of recovering all $\{i_j\}$ and $\{e_{i_j}\}$ given $\mathbf{y}$

Theorem

Let $\mathcal{C}$ be an (n, M, d) code over F. There is a decoder $\mathcal{D} : F^n \to \mathcal{C}$ that recovers correctly every pattern of up to $\lfloor (d-1)/2 \rfloor$ errors for every channel $S = (F, F, \mathsf{Prob})$.

Linear Codes

- Assume $\mathbb{F}$ is a *finite field*
- $\mathcal{C} : (n, M, d)$ over $\mathbb{F}$ is called a *linear code* if $\mathcal{C}$ is a *linear sub-space* of $\mathbb{F}^n$ over $\mathbb{F}$
  - $\mathbf{c}_1, \mathbf{c}_2 \in \mathcal{C},\ a_1, a_2 \in \mathbb{F} \quad \Rightarrow \quad a_1\mathbf{c}_1 + a_2\mathbf{c}_2 \in \mathcal{C}$
- A linear code $\mathcal{C}$ has $M = q^k$ codewords, where $k = \log_q M$ is the dimension of $\mathcal{C}$ as a linear space over $\mathbb{F}$
- $r = n - k$ is the redundancy of $\mathcal{C}$, $R = k/n$ its rate
- We use the notation $[n, k, d]$ to denote the parameters of a linear code
- A *generator* matrix for a linear code $\mathcal{C}$ is a $k \times n$ matrix G whose rows form a basis of $\mathcal{C}$.

Minimum Weight

- For an $[n, k, d]$ code $\mathcal{C}$,

$$\mathbf{c}_1, \mathbf{c}_2 \in \mathcal{C} \implies \mathbf{c}_1 - \mathbf{c}_2 \in \mathcal{C}\,, \text{ and } \mathsf{d}(\mathbf{c}_1, \mathbf{c}_2) = \mathsf{wt}(\mathbf{c}_1 - \mathbf{c}_2)\,.$$

  Therefore,

$$d = \min_{\mathbf{c}_1, \mathbf{c}_2 \in \mathcal{C}\,:\,\mathbf{c}_1 \neq \mathbf{c}_2} \mathsf{d}(\mathbf{c}_1, \mathbf{c}_2) = \min_{\mathbf{c}_1, \mathbf{c}_2 \in \mathcal{C}\,:\,\mathbf{c}_1 \neq \mathbf{c}_2} \mathsf{wt}(\mathbf{c}_1 - \mathbf{c}_2) = \min_{\mathbf{c} \in \mathcal{C} \setminus \{\mathbf{0}\}} \mathsf{wt}(\mathbf{c})\,.$$

  $\Rightarrow$ *minimum distance is the same as minimum weight for linear codes*

  - Recall also that $\mathbf{0} \in \mathcal{C}$ and $\mathsf{d}(\mathbf{c}, \mathbf{0}) = \mathsf{wt}(\mathbf{c})$

Encoding Linear Codes

- Since $\mathrm{rank}(G) = k$, the map $\mathcal{E} : \mathbb{F}^k \to \mathcal{C}$ defined by

$$\mathcal{E} : \mathbf{u} \mapsto \mathbf{u}G$$

  is 1-1, and can serve as an encoding mechanism for $\mathcal{C}$.

- Applying elementary row operations and possibly reordering coordinates, we can bring G to the form

$$G = \left(\; I_k \; \middle| \; A \; \right) \qquad \textit{systematic generator matrix,}$$

  where I_k is a $k \times k$ identity matrix, and A is a $k \times (n-k)$ matrix.

$$\mathbf{u} \mapsto \mathbf{u}G = (\, \mathbf{u} \mid \mathbf{u}A \,) \qquad \textit{systematic encoding.}$$

- In a systematic encoding, the *information symbols* from $\mathbf{u}$ are transmitted 'as is,' and $n-k$ *check symbols* (or *redundancy* symbols, or *parity* symbols) are appended.

Parity Check Matrix

- Let $\mathcal{C} : [n, k, d]$. A *parity-check matrix (PCM)* of $\mathcal{C}$ is an $r \times n$ matrix H such that for all $\mathbf{c} \in \mathbb{F}^n$,

$$\mathbf{c} \in \mathcal{C} \qquad \Longleftrightarrow \qquad H\mathbf{c}^T = \mathbf{0}\,.$$

- For a generator matrix G of $\mathcal{C}$, we have

$$HG^T = 0 \Rightarrow GH^T = 0, \;\; \textbf{and} \;\; \dim\ker(G) = n - \mathrm{rank}(G) = n - k = r$$

- If $G = (\ I_k \mid A\ )$, then $H = (-A^T \mid I_{n-k}\ )$ is a (systematic) parity-check matrix.

Cosets and Syndromes

- Let $\mathbf{y} \in \mathbb{F}^n$. The *syndrome* of $\mathbf{y}$ (with respect to a PCM H of $\mathcal{C}$) is defined by

$$\mathbf{s} = H\mathbf{y}^T \in \mathbb{F}^{n-k}.$$

The set

$$\mathbf{y} + \mathcal{C} = \{\mathbf{y} + \mathbf{c} \ : \ \mathbf{c} \in \mathcal{C}\}$$

is a *coset* of $\mathcal{C}$ (as an additive subgroup) in $\mathbb{F}^n$.

- If $\mathbf{y}_1 \in \mathbf{y} + \mathcal{C}$, then

$$\mathbf{y}_1 - \mathbf{y} \in \mathcal{C} \implies H(\mathbf{y}_1 - \mathbf{y})^T = \mathbf{0} \implies H\mathbf{y}_1^T = H\mathbf{y}^T$$

$\implies$ *The syndrome is invariant for all* $\mathbf{y}_1 \in \mathbf{y} + \mathcal{C}$.

- Let $F = F_q$. Given a PCM H, there is a 1-1 correspondence between the q^{n-k} cosets of $\mathcal{C}$ in $\mathbb{F}^n$ and the q^{n-k} possible syndrome values (H is full-rank $\implies$ all values are attained).

Syndrome Decoding of Linear Codes

- $\mathbf{c} \in \mathcal{C}$ is sent and $\mathbf{y} = \mathbf{c} + \mathbf{e}$ is received on an additive channel
- $\mathbf{y}$ and $\mathbf{e}$ are in the same coset of $\mathcal{C}$
- Nearest-neighbor decoding of $\mathbf{y}$ calls for finding the closest codeword $\mathbf{c}$ to $\mathbf{y}$ $\implies$ find a vector $\mathbf{e}$ of *lowest weight* in $\mathbf{y}+\mathcal{C}$: a *coset leader*.
  - *coset leaders need not be unique* (when are they?)
- Decoding algorithm: upon receiving $\mathbf{y}$
  - compute the syndrome $\mathbf{s} = H\mathbf{y}^T$
  - find a coset leader $\mathbf{e}$ in the coset corresponding to $\mathbf{s}$
  - decode $\mathbf{y}$ into $\hat{\mathbf{c}} = \mathbf{y} - \mathbf{e}$
- If $n - k$ is (very) small, a table containing one leader per coset can be pre-computed. The table is indexed by $\mathbf{s}$.
- In general, however, syndrome decoding appears exponential in $n - k$. In fact, it has been shown to be NP-hard.

The Singleton Bound

- The *Singleton bound*.

Theorem (Singleton bound)

For any (n, M, d) *code over an alphabet of size* q,

$$d \leq n - (\log_q M) + 1 \ .$$

- Singleton bound for linear codes

Theorem (Singleton bound for linear codes)

For any linear $[n, k, d]$ *code over* $GF(q)$,

$$d \leq n - k + 1 \ .$$

- $\mathcal{C} : (n, M, d)$ (or, if linear, $\mathcal{C} : [n, k, d]$) is called *maximum distance separable (MDS)* if it meets the Singleton bound, namely $d = n - (\log_q M) + 1 \quad (d = n - k + 1)$.

The Sphere-Packing Bound

The *sphere* of center $\mathbf{c}$ and radius t in $\mathbb{F}_q^n$ is the set of vectors at Hamming distance t or less from $\mathbf{c}$. Its *volume* (cardinality) is

$$V_q(n,t) = \sum_{i=0}^{t} \binom{n}{i} (q-1)^i .$$

Theorem (The sphere-packing (SP) bound)

For any (n, M, d) code over $\mathbb{F}_q$,

$$M \cdot V_q(n, \lfloor (d-1)/2 \rfloor) \le q^n .$$

Proof. Spheres of radius $t = \lfloor (d-1)/2 \rfloor$ centered at codewords must be disjoint. $\square$

For a linear $[n,k,d]$ code, the bound becomes $V_q(n, \lfloor (d-1)/2 \rfloor) \le q^{n-k}$.
For $q = 2$,

$$\sum_{i=0}^{\lfloor (d-1)/2 \rfloor} \binom{n}{i} \le 2^{n-k}$$

The Gilbert-Varshamov bound

The Singleton and SP bounds set *necessary* conditions on the parameters of a code. The following is a *sufficient* condition:

Theorem (The Gilbert-Varshamov (GV) bound)

There exists an $[n, k, d]$ *code over the field* $\mathbb{F}_q$ *whenever*

$$V_q(n-1, d-2) < q^{n-k}.$$

Theorem

Let

$$\rho = \frac{q^k - 1}{q - 1} \cdot \frac{V_q(n, d-1)}{q^n}\ .$$

Then, a random $[n, k]$ *code has minimum distance* d *with* Prob $\geq 1 - \rho$.

Lots of codes are near the GV bound. But it's very hard to find them!

Asymptotic Bounds

- **Definition:** *relative distance* $\delta = d/n$
- We are interested in the behavior of δ and $R = (\log_q M)/n$ as $n \to \infty$.
- Singleton bound: $d \leq n - \lceil \log_q M \rceil + 1 \quad \Longrightarrow \quad \boxed{\; R \leq 1 - \delta + o(1)}$
- For the SP and GV bounds, we need estimates for $V_q(n,t)$
- **Definition:** *symmetric q-ary entropy function* $\mathsf{H}_q : [0,1] \to [0,1]$

$$\mathsf{H}_q(x) = -x \log_q x - (1-x) \log_q (1-x) + x \log_q (q-1) \;,$$

  - $\mathsf{H}_q(0) = 0,\ \mathsf{H}_q(1) = \log_q(q-1)$, strictly $\cap$-convex, $\max = 1$ at $x = 1 - 1/q$
  - coincides with $\mathsf{H}(x)$ when $q = 2$

Asymptotic Bounds (II)

Lemma. *For $0 \le t/n \le 1 - (1/q)$,*

$$V_q(n,t) = \sum_{i=0}^{t} \binom{n}{i} (q-1)^i \le q^{n\mathsf{H}_q(t/n)} .$$

Lemma. *For integers $0 \le t \le n$,*

$$V_q(n,t) \ge \binom{n}{t} (q-1)^t \ge \frac{1}{\sqrt{8t(1-(t/n))}} \cdot q^{n\mathsf{H}_q(t/n)} .$$

Theorem (Asymptotic SP bound)

For every $(n, q^{nR}, \delta n)$ code over $\mathbb{F}_q$,

$$R \le 1 - \mathsf{H}_q(\delta/2) + o(1) .$$

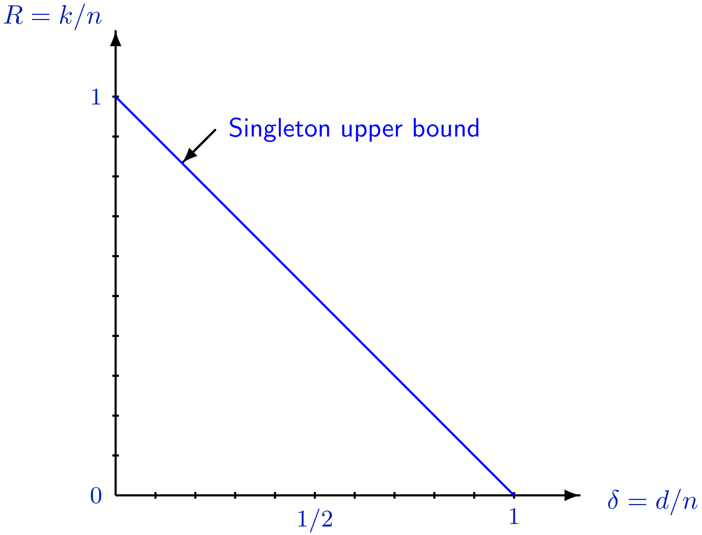
Theorem (Asymptotic GV bound)

Let n, nR, δn be positive integers such that $\delta \in (0, 1-(1/q)]$ and

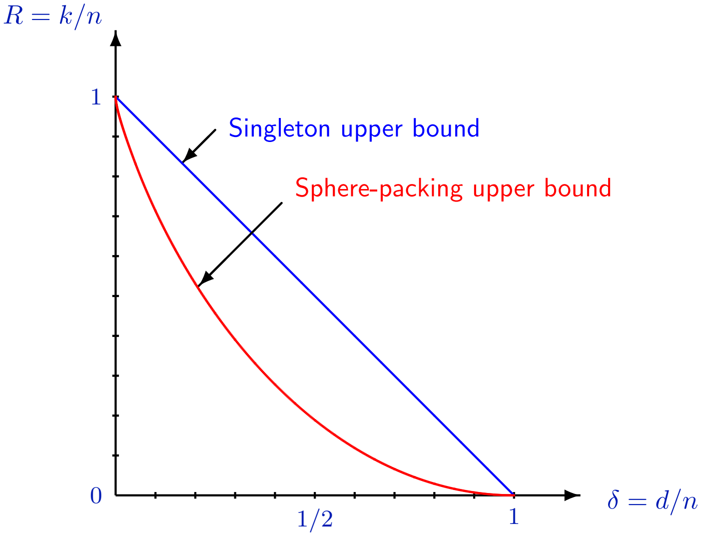
$$R \le 1 - \mathsf{H}_q(\delta) .$$

Then, there exists a linear $[n, nR, \ge \delta n]$ code over Fq.

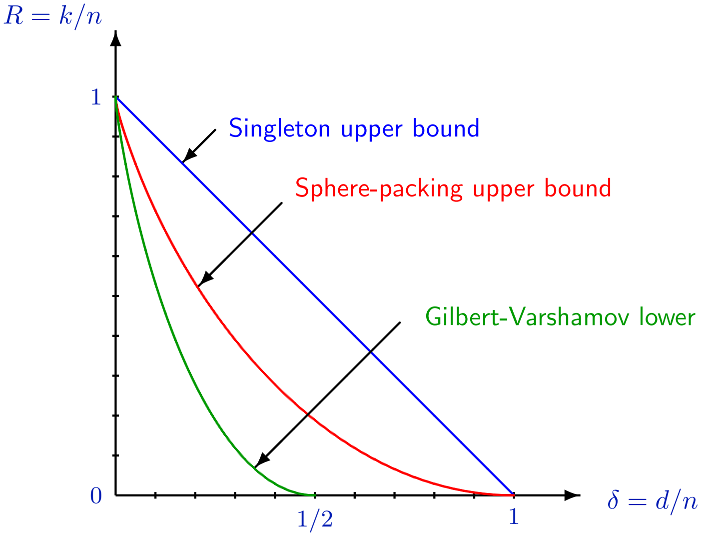
Plot of Asymptotic Bounds



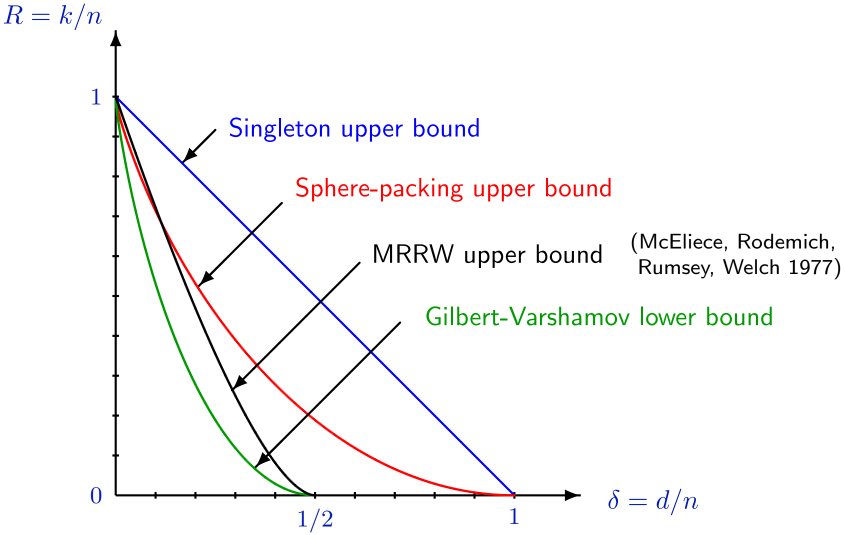
Plot of Asymptotic Bounds



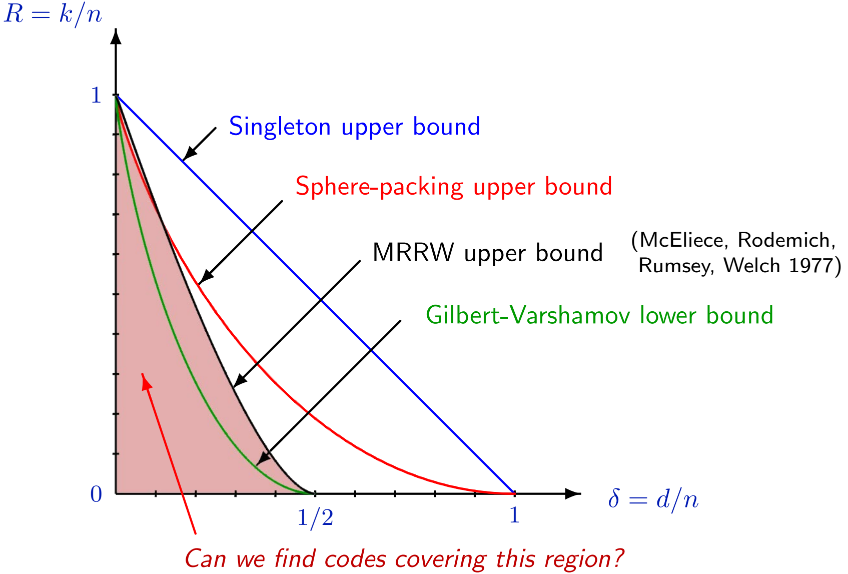
Plot of Asymptotic Bounds



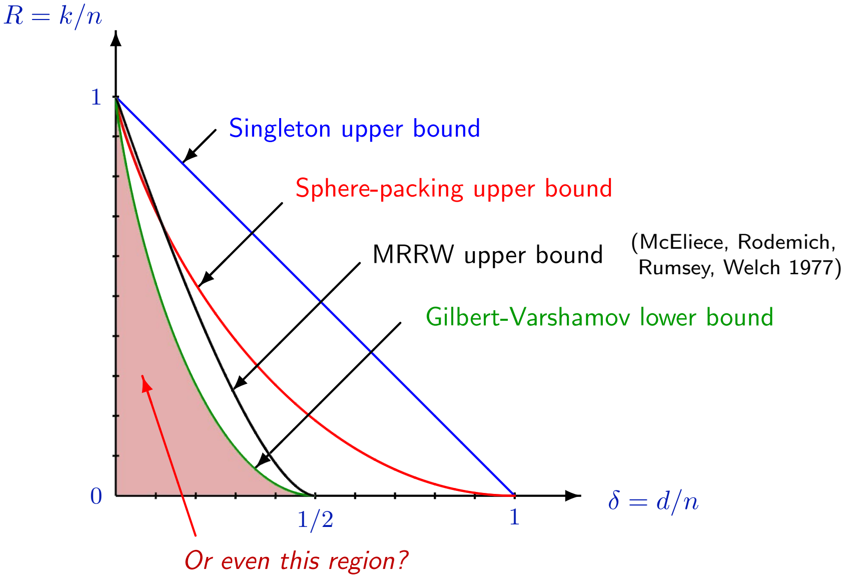
Plot of Asymptotic Bounds



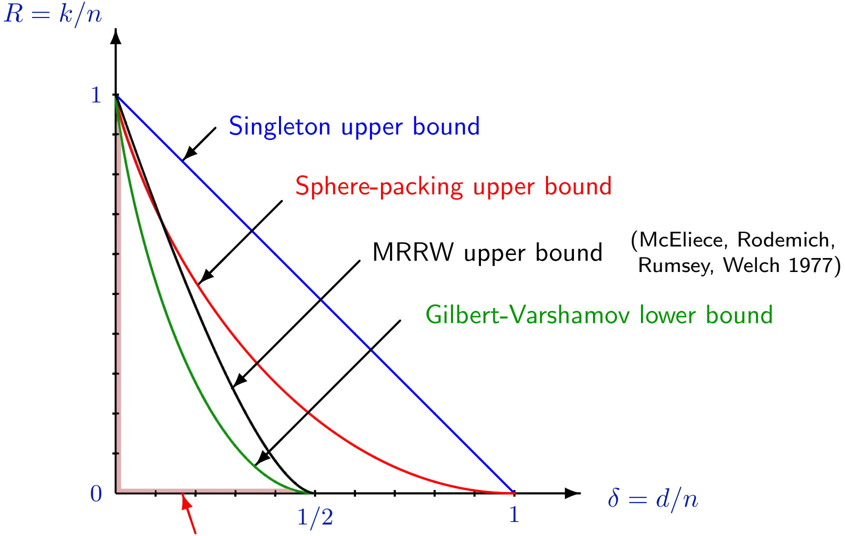
Plot of Asymptotic Bounds



Plot of Asymptotic Bounds



Plot of Asymptotic Bounds



So far, we have only seen codes on these lines!

What we lose for decoding only up to (d-1)/2

- Hamming (sphere packing) bound

$$R \leq 1 - H(\delta/2) + o(1)$$

- Assume binary symmetric channel of parameter p.
  Channel capacity: $C = 1 - H(p)$
  $\Rightarrow$ with R arbitrarily close to $1 - H(p)$, can correct typical patterns of weight np with probability 1
  $\Rightarrow$ "equivalent minimum distance" $\approx 2np$
  $\Rightarrow$ $\delta \approx 2p$
  $\Rightarrow$ can achieve virtually zero-error communication with $R \approx 1 - H(\delta/2)$

Hamming bound curve

Generalized Reed-Solomon Codes

- Let $\alpha_1, \alpha_2, \ldots, \alpha_n$, $n < q$, be distinct nonzero elements of $\mathbb{F}_q$, and let $v_1, v_2, \ldots, v_n$ be *nonzero* elements of $\mathbb{F}_q$ (not necessarily distinct). A *generalized Reed-Solomon (GRS)* code is a linear $[n, k, d]$ code $\mathcal{C}_{\mathrm{GRS}}$ with PCM

$$H_{\mathrm{GRS}} = \begin{pmatrix} 1 & 1 & \ldots & 1 \\ \alpha_1 & \alpha_2 & \ldots & \alpha_n \\ \alpha_1^2 & \alpha_2^2 & \ldots & \alpha_n^2 \\ \vdots & \vdots & \vdots & \vdots \\ \alpha_1^{n-k-1} & \alpha_2^{n-k-1} & \ldots & \alpha_n^{n-k-1} \end{pmatrix} \begin{pmatrix} v_1 & & & \\ & v_2 & & 0 \\ 0 & & \ddots & \\ & & & v_n \end{pmatrix} .$$

α_j: *code locators* (distinct), v_j: *column multipliers* ($\neq 0$)

Generalized Reed-Solomon Codes

- Let $\alpha_1, \alpha_2, \ldots, \alpha_n$, $n < q$, be distinct nonzero elements of $\mathbb{F}_q$, and let $v_1, v_2, \ldots, v_n$ be *nonzero* elements of $\mathbb{F}_q$ (not necessarily distinct). A *generalized Reed-Solomon (GRS)* code is a linear $[n, k, d]$ code $\mathcal{C}_{\mathrm{GRS}}$ with PCM

$$H_{\mathrm{GRS}} = \begin{pmatrix} 1 & 1 & \ldots & 1 \\ \alpha_1 & \alpha_2 & \ldots & \alpha_n \\ \alpha_1^2 & \alpha_2^2 & \ldots & \alpha_n^2 \\ \vdots & \vdots & \vdots & \vdots \\ \alpha_1^{n-k-1} & \alpha_2^{n-k-1} & \ldots & \alpha_n^{n-k-1} \end{pmatrix} \begin{pmatrix} v_1 & & & \\ & v_2 & & 0 \\ 0 & & \ddots & \\ & & & v_n \end{pmatrix}.$$

α_j: *code locators* (distinct), v_j: *column multipliers* ($\neq 0$)

Theorem

$\mathcal{C}_{\mathrm{GRS}}$ is an MDS code, namely, $d = n - k + 1$.

Theorem

The dual of a GRS code is a GRS code.

GRS Encoding as Polynomial Evaluation

- For $\mathbf{u} = (u_0\, u_1\, \ldots\, u_{k-1})$, let $u(x) = u_0 + u_1 x + u_2 x^2 + \cdots + u_{k-1}x^{k-1}$. Then,

$$\mathbf{c} = \mathbf{u}G_{\text{GRS}} = \mathbf{u} \cdot \begin{pmatrix} 1 & 1 & \ldots & 1 \\ \alpha_1 & \alpha_2 & \ldots & \alpha_n \\ \alpha_1^2 & \alpha_2^2 & \ldots & \alpha_n^2 \\ \vdots & \vdots & \vdots & \vdots \\ \alpha_1^{k-1} & \alpha_2^{k-1} & \ldots & \alpha_n^{k-1} \end{pmatrix} \begin{pmatrix} v'_1 & & & 0 \\ & v'_2 & & \\ 0 & & \ddots & \\ & & & v'_n \end{pmatrix}$$

$$= [\, v'_1 u(\alpha_1) \;\; v'_2 u(\alpha_2) \;\; \ldots \;\; v'_n u(\alpha_n)\,]$$

- Minimum distance now follows from the fact that a polynomial of degree $\leq k-1$ cannot have more than $k-1$ roots in $\mathbb{F}_q \implies \mathrm{wt}(\mathbf{c}) \geq n-k+1$.
- Decoding as *noisy interpolation*: reconstruct $u(x)$ from $(k+2t)$ noisy evaluations $u(\alpha_1)+e_1,\, u(\alpha_2)+e_2, \ldots, u(\alpha_{k+2t})+e_{k+2t}$, possible if at most t evaluations are corrupted.

Conventional Reed-Solomon Codes

- *Conventional Reed-Solomon (RS)* code: GRS code with $n|(q-1)$, $\alpha \in \mathbb{F}^*$ with $\mathcal{O}(\alpha) = n$,

$$\begin{array}{rcll} \alpha_j & = & \alpha^{j-1} , & 1 \le j \le n, \\ v_j & = & \alpha^{b(j-1)} , & 1 \le j \le n \,. \end{array}$$

- *Canonical PCM* of a RS code is given by

$$H_{\mathrm{RS}} = \begin{pmatrix} 1 & \alpha^b & \ldots & \alpha^{(n-1)b} \\ 1 & \alpha^{b+1} & \ldots & \alpha^{(n-1)(b+1)} \\ \vdots & \vdots & \vdots & \vdots \\ 1 & \alpha^{b+d-2} & \ldots & \alpha^{(n-1)(b+d-2)} \end{pmatrix} \quad (\#\ \text{rows} = d{-}1 = n{-}k)$$

- $\mathbf{c} \in \mathcal{C}_{\mathrm{RS}} \iff H_{\mathrm{RS}}\mathbf{c}^T = \mathbf{0} \iff c(\alpha^\ell) = 0,\ \ell = b, b{+}1, \ldots, b{+}d{-}2$.
- $\alpha^b, \alpha^{b+1}, \ldots, \alpha^{b+d-2}$: *roots* of $\mathcal{C}_{\mathrm{RS}}$
- $g(x) = (x - \alpha^b)(x - \alpha^{b+1}) \cdots (x - \alpha^{b+d-2})$: *generator polynomial* of $\mathcal{C}_{\mathrm{RS}}$

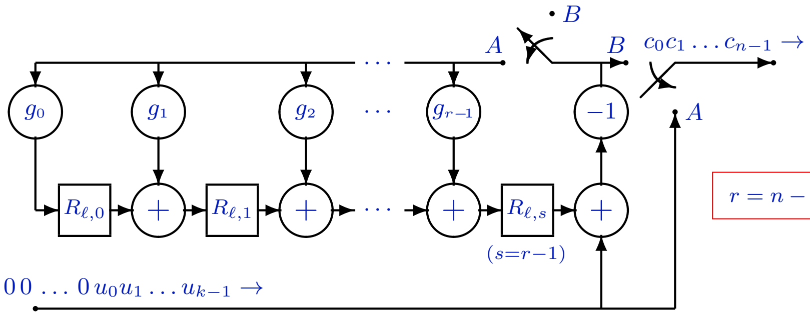
Systematic Encoding of RS Codes

- For $u(x) \in \mathbb{F}_q[x]_k$, let $r_u(x)$ be the unique polynomial in $\mathbb{F}_q[x]_{n-k}$ such that
$$r_u(x) \equiv x^{n-k}u(x) \mod g(x)$$

- Clearly, $x^{n-k}u(x) - r_u(x) \in \mathcal{C}_{\mathrm{RS}}$
- The mapping $\mathcal{E}_{\mathrm{RS}} : u(x) \mapsto x^{n-k}u(x) - r_u(x)$ is a *linear, systematic* encoding for $\mathcal{C}_{\mathrm{RS}}$

$$\begin{array}{rcccccccl}
[& u_{k-1} & u_{k-2} & \dots & u_0 & 0 & 0 & \dots & 0 \quad] \\
-[& 0 & 0 & \dots & 0 & r_{n-k-1} & r_{n-k-2} & \dots & r_0 \quad] \\
\hline
[& c_{n-1} & c_{n-2} & \dots & c_{n-k} & c_{n-k-1} & c_{n-k-2} & \dots & c_0 \quad]
\end{array}$$

Systematic Encoding Circuit



$\boxed{r = n - k}$

Switches:

- at A for k cycles
- at B for $r=n-k$ cycles

Register contents:

$$R_\ell(x) = \sum_{i=0}^{r-1} R_{\ell,i} x^i, \quad 0 \le \ell < k$$

with initial condition

$$R_0(x) = 0$$

Decoding Generalized Reed-Solomon Codes

- We consider $\mathcal{C}_{\mathrm{GRS}}$ over $\mathbb{F}_q$ with PCM

$$H_{\mathrm{GRS}} = \begin{pmatrix} 1 & 1 & \dots & 1 \\ \alpha_1 & \alpha_2 & \dots & \alpha_n \\ \alpha_1^2 & \alpha_2^2 & \dots & \alpha_n^2 \\ \vdots & \vdots & \vdots & \vdots \\ \alpha_1^{d-2} & \alpha_2^{d-2} & \dots & \alpha_n^{d-2} \end{pmatrix} \begin{pmatrix} v_1 & & & 0 \\ & v_2 & & \\ 0 & & \ddots & \\ & & & v_n \end{pmatrix}$$

  with $\alpha_1, \alpha_2, \dots, \alpha_n \in \mathbb{F}_q^*$ distinct, and $v_1, v_2, \dots, v_n \in \mathbb{F}_q^*$
- Codeword $\mathbf{c}$ transmitted, word $\mathbf{y}$ received, with error vector

$$\mathbf{e} = (e_1 \; e_2 \; \dots \; e_n) = \mathbf{y} - \mathbf{c}$$

- $J = \{\kappa \; : \; e_\kappa \neq 0\}$ set of *error locations*
- We describe an algorithm that correctly decodes $\mathbf{y}$ to $\mathbf{c}$, under the assumption $|J| \leq \frac{1}{2}(d-1)$.

Syndrome Computation

- First step of the decoding algorithm

$$\mathbf{S} = \begin{pmatrix} S_0 \\ S_1 \\ \vdots \\ S_{d-2} \end{pmatrix} = H_{\mathrm{GRS}} \mathbf{y}^T = H_{\mathrm{GRS}} \mathbf{e}^T$$

$$S_\ell = \sum_{j=1}^{n} y_j v_j \alpha_j^\ell = \sum_{j=1}^{n} e_j v_j \alpha_j^\ell = \sum_{j \in J} e_j v_j \alpha_j^\ell \,, \quad \ell = 0, 1, \ldots, d-2$$

Example. For RS codes, we have $\alpha_j = \alpha^{j-1}$ and $v_j = \alpha^{b(j-1)}$, so

$$S_\ell = \sum_{j=1}^{n} y_j \alpha^{(j-1)(b+\ell)} = y(\alpha^{b+\ell}) \,, \quad \ell = 0, 1, \ldots, d-2 \,.$$

- *Syndrome polynomial*:

$$S(x) = \sum_{\ell=0}^{d-2} S_\ell x^\ell = \sum_{\ell=0}^{d-2} x^\ell \sum_{j \in J} e_j v_j \alpha_j^\ell = \sum_{j \in J} e_j v_j \sum_{\ell=0}^{d-2} (\alpha_j x)^\ell \,.$$

A Congruence for the Syndrome Polynomial

$$S(x) = \sum_{j\in J} e_j v_j \sum_{\ell=0}^{d-2} (\alpha_j x)^\ell \,.$$

- We have

$$\sum_{\ell=0}^{d-2} (\alpha_j x)^\ell \equiv \frac{1}{1-\alpha_j x} \pmod{x^{d-1}}$$

$$\Longrightarrow \quad \boxed{S(x) \equiv \sum_{j\in J} \frac{e_j v_j}{1-\alpha_j x} \pmod{x^{d-1}}} \qquad \left(\textstyle\sum_\phi \Box = 0\right)$$

More Auxiliary Polynomials

- *Error locator polynomial (ELP)*

$$\Lambda(x) = \prod_{j \in J} (1 - \alpha_j x) \qquad \left(\prod_{\phi} \square \triangleq 1 \right)$$

- *Error evaluator polynomial (EEP)*

$$\Gamma(x) = \sum_{j \in J} e_j v_j \prod_{m \in J \setminus \{j\}} (1 - \alpha_m x)$$

- $\Lambda(\alpha_\kappa^{-1}) = 0 \quad \Longleftrightarrow \quad \kappa \in J$ *roots of EEP point to error locations*
- $\Gamma(\alpha_\kappa^{-1}) = e_\kappa v_\kappa \prod_{m \in J \setminus \{\kappa\}} (1 - \alpha_m \alpha_\kappa^{-1}) \neq 0$

$$\Longrightarrow \quad \boxed{\gcd(\Lambda(x), \Gamma(x)) = 1}$$

- The degrees of ELP and EEP satisfy

$$\deg \Lambda = |J| \quad \text{and} \quad \deg \Gamma < |J|$$

Of course, we don't know $\Lambda(x)$, $\Gamma(x)$: our goal is to find them

Key Equation of GRS Decoding

Since $|J| \leq \frac{1}{2}(d-1)$, we have

(1) $\boxed{\deg \Lambda \leq \frac{1}{2}(d-1)}$ and (2) $\boxed{\deg \Gamma < \frac{1}{2}(d-1)}$

The ELP and the EEP are related by

$$\Gamma(x) = \sum_{j \in J} e_j v_j \prod_{m \in J \setminus \{j\}} (1 - \alpha_m x) = \sum_{j \in J} e_j v_j \frac{\Lambda(x)}{1 - \alpha_j x} = \Lambda(x) \sum_{j \in J} \frac{e_j v_j}{1 - \alpha_j x}$$

$$\Longrightarrow (3) \quad \boxed{\Lambda(x) S(x) \equiv \Gamma(x) \pmod{x^{d-1}}}$$

(1)+(2)+(3): *key equation of GRS decoding*

- (3) is a set of $d-1$ *linear* equations in the coefficients of Λ and Γ
- $\lfloor \frac{1}{2}(d-1) \rfloor$ equations depend only on Λ (corresponding to x^i, $i \geq \frac{1}{2}(d-1)$)
- we can solve for Λ, find its root set J, then solve *linear* equations for e_j
- straightforward solution leads to $O(d^3)$ algorithm — we'll present an $O(d^2)$ one

Solving the Key Equation

- Apply the Euclidean algorithm with
  $a(x) = x^{d-1}$ and $b(x) = S(x)$, to produce
  $\Lambda(x) = c \cdot t_h(x)$ and $\Gamma(x) = c \cdot r_h(x)$
  [the key equation guarantees conditions (C1)–(C3)].
  How do we find h —the stopping index?

Theorem

The solution to the key equation is unique up to a scalar constant, and it is obtained with the Euclidean algorithm by stopping at the unique index h such that

$$\deg r_h < \tfrac{1}{2}(d-1) \leq \deg r_{h-1}$$

Finding the Error Values

- *Formal derivatives* in finite fields: $\left[\sum_{i=0}^{s} a_i x^i\right]' = \sum_{i=1}^{s} i a_i x^{i-1}$
  $(a(x)b(x))' = a'(x)b(x) + a(x)b'(x)$ (not surprising)
- For the ELP, we have

$$\Lambda(x) = \prod_{j \in J} (1 - \alpha_j x) \quad \Longrightarrow \quad \Lambda'(x) = \sum_{j \in J} (-\alpha_j) \prod_{m \in J \setminus \{j\}} (1 - \alpha_m x),$$

and, for $\kappa \in J$,

$$\Lambda'(\alpha_\kappa^{-1}) = -\alpha_\kappa \prod_{m \in J \setminus \{\kappa\}} (1 - \alpha_m \alpha_\kappa^{-1}),$$

$$\Gamma(\alpha_\kappa^{-1}) = e_\kappa v_\kappa \prod_{m \in J \setminus \{\kappa\}} (1 - \alpha_m \alpha_\kappa^{-1})$$

- Therefore, for all error locations $\kappa \in J$, we obtain

$$\boxed{e_\kappa = -\frac{\alpha_\kappa}{v_\kappa} \cdot \frac{\Gamma(\alpha_\kappa^{-1})}{\Lambda'(\alpha_\kappa^{-1})}}$$

Forney's algorithm for error values

Summary of GRS Decoding

Input: received word $(y_1\ y_2\ \ldots\ y_n) \in \mathbb{F}_q^n$.
Output: error vector $(e_1\ e_2\ \ldots\ e_n) \in \mathbb{F}_q^n$.

1. *Syndrome computation:* Compute the polynomial $S(x) = \sum_{\ell=0}^{d-2} S_\ell x^\ell$ by
$$S_\ell = \sum_{j=1}^{n} y_j v_j \alpha_j^\ell\ , \quad \ell = 0, 1, \ldots, d-2\ .$$
2. *Solving the key equation:* Apply Euclid's algorithm to $a(x) \leftarrow x^{d-1}$ and $b(x) \leftarrow S(x)$ to produce $\Lambda(x) \leftarrow t_h(x)$ and $\Gamma(x) \leftarrow r_h(x)$, where h is the smallest index i for which $\deg r_i < \frac{1}{2}(d-1)$.
3. *Forney's algorithm:* Compute the error locations and values by
$$e_j = \begin{cases} -\dfrac{\alpha_j}{v_j} \cdot \dfrac{\Gamma(\alpha_j^{-1})}{\Lambda'(\alpha_j^{-1})} & \text{if } \Lambda(\alpha_j^{-1}) = 0 \\ 0 & \text{otherwise} \end{cases}\ , \quad j = 1, 2, \ldots, n\ .$$

Complexity: 1. $O(dn)$ 2. $O\big((|J|+1)\, d\big)$ 3. $O\big((|J|+1)n\big)$

Back to main