List error-correction with information-theoretically minimal redundancy

Venkatesan Guruswami

Carnegie Mellon University

(Spring'14 @ Microsoft Research New England)

2014 European School of Information Theory

Tallinn, Estonia

April 17, 2014

Codes

Error-correcting code $C \subseteq \Sigma^N$
with encoding map $E : \mathcal{M} \to \Sigma^N$ $\quad$ (Image$(E) = C$)

- $\mathcal{M}$ = message space; $\quad \Sigma$ = alphabet; $\quad N$ = block length.
- To communicate *message* m, send **codeword** $E(m) \in C$.

Codes

Error-correcting code $C \subseteq \Sigma^N$
with encoding map $E : \mathcal{M} \to \Sigma^N$ $\quad$ ($\text{Image}(E) = C$)

- $\mathcal{M}$ = message space; $\quad \Sigma$ = alphabet; $\quad N$ = block length.
- To communicate *message* m, send **codeword** $E(m) \in C$.

Rate $R = \frac{\log |\mathcal{M}|}{N \log |\Sigma|}$. $(\in [0, 1])$

- Ratio of # information bits communicated to # transmitted bits
- Identify messages $\mathcal{M} \simeq \Sigma^{RN}$; $\quad |C| = |\Sigma|^{RN}$.
- Proportion of redundant bits $= 1 - R$

Error correction

We'll be interested in correcting **worst-case** (adversarial) errors.

- arbitrary corruption of up to τN symbols ($\tau =$ error fraction)
- Both error locations and error values worst-case
- We count *symbol errors*, not bit errors.

Refer to τ as "decoding radius" (or error-correction radius)

Error correction

We'll be interested in correcting **worst-case** (adversarial) errors.

- arbitrary corruption of up to τN symbols ($\tau =$ error fraction)
- Both error locations and error values worst-case
- We count *symbol errors*, not bit errors.

Refer to τ as "decoding radius" (or error-correction radius)

Decoding problem for code $C \subset \Sigma^N$ up to radius τ:

Input: "Noisy received word" $\mathbf{y} \in \Sigma^N$

Output: Codeword $\mathbf{c} \in C$ such that the Hamming distance $\Delta(\mathbf{c}, \mathbf{y}) \leqslant \tau N$.

Rate vs. decoding radius

Goal

Would like *both* R and τ to be large (and alphabet Σ to be small).

(Think of $R, \tau \in (0, 1)$ as fixed, and block length $N \to \infty$.)

Conflicting goals: correcting more errors requires more redundancy (lower rate).

Rate vs. error-correction radius

A trivial information-theoretic limit: $\tau \leqslant 1 - R$

- $|\mathcal{M}| = |\Sigma|^{RN} \implies$ need at least RN correct symbols from Σ to have any hope of meaningfully recovering message.
- Need *redundancy* $\geqslant$ *target error fraction.*

Question

Could we hope to approach such a nice trade-off?

Unique decoding

- $|C| = |\Sigma|^{RN} \implies$ some two codewords $c_1 \neq c_2 \in C$ agree in first $RN - 1$ positions, i.e., differ in $\leqslant (1 - R)N + 1$ positions. (Singleton Bound)
- So when $\tau \geqslant (1 - R)/2$, can't unambigiously recover correct codeword (for worst-case errors).

Unique decoding

- $|C| = |\Sigma|^{RN} \Longrightarrow$ some two codewords $c_1 \neq c_2 \in C$ agree in first $RN - 1$ positions, i.e., differ in $\leqslant (1 - R)N + 1$ positions. (Singleton Bound)
- So when $\tau \geqslant (1 - R)/2$, can't unambigiously recover correct codeword (for worst-case errors).
- "Unique decoding" for error fraction $\tau \approx (1 - R)/2$ achieved by Reed-Solomon (or similar) codes.
  - Note: This is over large alphabets
- For larger τ, resort to **list decoding**.

List decoding

List decoding code $C \subset \Sigma^N$ up to radius τ:

Input: Noisy received word $\mathbf{y} \in \Sigma^N$

Output: A list of **all** codewords $\mathbf{c} \in C$ such that the Hamming distance $\Delta(\mathbf{c}, \mathbf{y}) \leqslant \tau N$.

List decoding

List decoding code $C \subset \Sigma^N$ up to radius τ:

Input: Noisy received word $\mathbf{y} \in \Sigma^N$

Output: A list of **all** codewords $\mathbf{c} \in C$ such that the Hamming distance $\Delta(\mathbf{c}, \mathbf{y}) \leqslant \tau N$.

Comments:

1. Code must guarantee that list is *small* for *every* $\mathbf{y}$
2. Need to find the list in $\mathrm{poly}(N)$ time, exploiting code structure.

A combinatorial definition

Definition (List decodability)

A code $C \subset \Sigma^N$ is said to be **(τ, ℓ)-list decodable** if for $\forall \mathbf{y} \in \Sigma^N$, there are $\leqslant \ell$ codewords of C within Hamming distance τN of $\mathbf{y}$.

Such a code offers potential for correcting τ fraction worst-case errors up to ambiguity ("list-size") ℓ.

The model of list decoding

But how useful is a list anyway?

1. List size > 1 typically a rare event (and we don't need to model channel stochastics precisely!)
2. In worst-case, better than decoding failure
   - Could use context/side information (or pick closest codeword) to disambiguate
3. Extensions such as list recovery & soft decoding very useful
   - decoding concatenated codes
   - practical use of channel reliability information
4. Versatile primitive
   - codes for computationally limited channels
5. Many applications beyond coding theory
   - eg. in complexity theory and cryptography
   - list decoding fits the bill as the right notion

The potential of list decoding

A code $C \subset \Sigma^N$ is said to be (τ, ℓ)-**list decodable** if for $\forall \mathbf{y} \in \Sigma^N$, there are $\leqslant \ell$ codewords of C within Hamming distance τN of y.

The potential of list decoding

A code $C \subset \Sigma^N$ is said to be (τ, ℓ)-**list decodable** if for $\forall \mathbf{y} \in \Sigma^N$, there are $\leqslant \ell$ codewords of C within Hamming distance τN of y.

Theorem (Non-constructive, via random coding)

For all $q \geqslant 2$, $\varepsilon > 0$ and $p \in (0, 1 - 1/q)$, there **exists** *a $(p, 1/\varepsilon)$-list decodable code of rate $1 - h_q(p) - \varepsilon$ over alphabet size q.*

The potential of list decoding

A code $C \subset \Sigma^N$ is said to be (τ, ℓ)-**list decodable** if for $\forall \mathbf{y} \in \Sigma^N$, there are $\leqslant \ell$ codewords of C within Hamming distance τN of y.

Theorem (Non-constructive, via random coding)

For all $q \geqslant 2$, $\varepsilon > 0$ and $p \in (0, 1 - 1/q)$, there **exists** *a $(p, 1/\varepsilon)$-list decodable code of rate $1 - h_q(p) - \varepsilon$ over alphabet size q.*

- Binary codes: Approach "Shannon capacity" of BSC_p for worst-case errors ("bridge" between Shannon & Hamming)

The potential of list decoding

A code $C \subset \Sigma^N$ is said to be (τ, ℓ)-**list decodable** if for $\forall \mathbf{y} \in \Sigma^N$, there are $\leqslant \ell$ codewords of C within Hamming distance τN of y.

Theorem (Non-constructive, via random coding)

For all $q \geqslant 2$, $\varepsilon > 0$ and $p \in (0, 1 - 1/q)$, there **exists** *a $(p, 1/\varepsilon)$-list decodable code of rate $1 - h_q(p) - \varepsilon$ over alphabet size q.*

- Binary codes: Approach "Shannon capacity" of BSC_p for worst-case errors ("bridge" between Shannon & Hamming)
- Large q: $(1 - R - \varepsilon, 1/\varepsilon)$-list decodable code over alphabet size $\exp(O(1/\varepsilon))$.

The potential of list decoding

A code $C \subset \Sigma^N$ is said to be (τ, ℓ)-**list decodable** if for $\forall \mathbf{y} \in \Sigma^N$, there are $\leqslant \ell$ codewords of C within Hamming distance τN of y.

Theorem (Non-constructive, via random coding)

For all $q \geqslant 2$, $\varepsilon > 0$ and $p \in (0, 1 - 1/q)$, there **exists** *a $(p, 1/\varepsilon)$-list decodable code of rate $1 - h_q(p) - \varepsilon$ over alphabet size q.*

- Binary codes: Approach "Shannon capacity" of BSC_p for worst-case errors ("bridge" between Shannon & Hamming)
- Large q: $(1 - R - \varepsilon, 1/\varepsilon)$-list decodable code over alphabet size $\exp(O(1/\varepsilon))$.

  $\Rightarrow$ **List decoding offers the potential to approach the $\tau = 1 - R$ limit with small list-size ℓ**

Random coding argument

Theorem

For all $q \geqslant 2$, $\varepsilon > 0$ and $p \in (0, 1 - 1/q)$, there **exists** *a $(p, 1/\varepsilon)$-list decodable code of rate $1 - h_q(p) - \varepsilon$ over alphabet size q.*
($h_q(x) = x \log_q(q-1) - x \log_q x - (1-x)\log_q(1-x)$ is q-ary entropy function)

Proof sketch.

Let $R = 1 - h_q(p) - \varepsilon$ and $\ell = \frac{1}{\varepsilon} + 1$.
Pick q^{Rn} codewords at random from $\{1, 2, \ldots, q\}^n$.
Prob. that code is not $(p, \ell - 1)$-list decodable is at most

$$q^n \cdot q^{Rn\ell} \cdot \left(\frac{q^{h_q(p)n}}{q^n}\right)^{\ell}$$

Random coding argument

Theorem

For all $q \geqslant 2$, $\varepsilon > 0$ and $p \in (0, 1 - 1/q)$, there **exists** *a $(p, 1/\varepsilon)$-list decodable code of rate $1 - h_q(p) - \varepsilon$ over alphabet size q.*
($h_q(x) = x \log_q(q-1) - x \log_q x - (1-x)\log_q(1-x)$ is q-ary entropy function)

Proof sketch.

Let $R = 1 - h_q(p) - \varepsilon$ and $\ell = \frac{1}{\varepsilon} + 1$.
Pick q^{Rn} codewords at random from $\{1, 2, \ldots, q\}^n$.
Prob. that code is not $(p, \ell - 1)$-list decodable is at most

$$q^n \cdot q^{Rn\ell} \cdot \left(\frac{q^{h_q(p)n}}{q^n} \right)^{\ell} \quad = q^{n(1+\ell(R+h_q(p)-1))} \quad = q^{n(1-\varepsilon\ell)} = q^{-\varepsilon n} \quad \square$$

Explicit list decoding

Challenges: Realize this constructively

1. List decode error fraction τ with an *explicit binary* code of rate $\approx 1 - h(\tau)$
2. List decode error fraction $\tau = 1 - R - \varepsilon$ with an *explicit* code of rate R

Explicit list decoding

Challenges: Realize this constructively

1. List decode error fraction τ with an *explicit binary* code of rate $\approx 1 - h(\tau)$
2. List decode error fraction $\tau = 1 - R - \varepsilon$ with an *explicit* code of rate R

The goal for binary codes is wide open.
But the second challenge over large alphabets has been met:

Explicit list decoding

Challenges: Realize this constructively

1. List decode error fraction τ with an *explicit binary* code of rate $\approx 1 - h(\tau)$
2. List decode error fraction $\tau = 1 - R - \varepsilon$ with an *explicit* code of rate R

The goal for binary codes is wide open.
But the second challenge over large alphabets has been met:

Theorem (G.-Rudra'08)

For all $R \in (0,1)$ and $\varepsilon > 0$, explicit codes ("folded Reed-Solomon") of rate R with efficient list decoding up to radius $\tau = 1 - R - \varepsilon$.

Plus, subsequent improvements to other parameters (alphabet size, list-size).

Rest of the talk

1. (List) decoding Reed-Solomon codes (see Powerpoint slides)
2. Folded Reed-Solomon codes: Linear-algebraic list decoding
3. Subspace-evasive pre-coding
    - Extensions to algebraic-geometric & rank-metric codes
4. Concluding remarks, Open challenges

Talk plan

1. (List) decoding Reed-Solomon codes
2. **Folded Reed-Solomon codes: Linear-algebraic list decoding**
3. Subspace-evasive pre-coding
   - Extensions to algebraic-geometric & rank-metric codes
4. Concluding remarks, Open challenges

Folded Reed-Solomon codes

Definition (Reed-Solomon codes)

Messages = polynomials $f \in \mathbb{F}_q[X]$ of degree $< k$. Encoding:

$$f \mapsto (f(1), f(\gamma), f(\gamma^2), \ldots, f(\gamma^{n-1}))$$

where γ is a primitive element of $\mathbb{F}_q$ (and $n < q$).

Rate $= k/n$; alphabet size $= q$.

Folded Reed-Solomon codes

Definition (Reed-Solomon codes)

Messages = polynomials $f \in \mathbb{F}_q[X]$ of degree $< k$. Encoding:

$$f \mapsto (f(1), f(\gamma), f(\gamma^2), \ldots, f(\gamma^{n-1}))$$

where γ is a primitive element of $\mathbb{F}_q$ (and $n < q$).

Rate $= k/n$; alphabet size $= q$.

Definition (m-Folded Reed-Solomon codes)

Same rate; alphabet size q^m; block length $= n/m$

$$f \mapsto \left(\begin{bmatrix} f(1) \\ f(\gamma) \\ \vdots \\ f(\gamma^{m-1}) \end{bmatrix}, \begin{bmatrix} f(\gamma^m) \\ f(\gamma^{m+1}) \\ \vdots \\ f(\gamma^{2m-1}) \end{bmatrix}, \ldots, \begin{bmatrix} f(\gamma^{n-m}) \\ f(\gamma^{n-m+1}) \\ \vdots \\ f(\gamma^{n-1}) \end{bmatrix} \right) .$$

Folded Reed-Solomon list decoding

Theorem (G.-Rudra; based on root-finding in extension fields, building on Parvaresh-Vardy)

For any s, $1 \leqslant s \leqslant m$, the m-folded RS code can be list decoding from error fraction $\tau \approx 1 - \left(\frac{mR}{m-s+1}\right)^{s/(s+1)}$ with list-size q^s.

- $s = m = 1$ is the $1 - \sqrt{R}$ bound for RS codes.
- Picking $s \approx 1/\varepsilon$, $m \approx 1/\varepsilon^2$, $\tau \geqslant 1 - R - \varepsilon$.

Folded Reed-Solomon list decoding

Theorem (G.-Rudra; based on root-finding in extension fields, building on Parvaresh-Vardy)

For any s, $1 \leqslant s \leqslant m$, the m-folded RS code can be list decoding from error fraction $\tau \approx 1 - \left(\frac{mR}{m-s+1}\right)^{s/(s+1)}$ with list-size q^s.

- $s = m = 1$ is the $1 - \sqrt{R}$ bound for RS codes.
- Picking $s \approx 1/\varepsilon$, $m \approx 1/\varepsilon^2$, $\tau \geqslant 1 - R - \varepsilon$.

Theorem (Linear-algebra approach (G.-Wang'13))

For any s, $1 \leqslant s \leqslant m$, the m-folded RS code can be list decoded from error fraction $\tau = \frac{s}{s+1}\left(1 - \frac{mR}{m-s+1}\right)$ with list-size q^{s-1}.

- $s = m = 1$: $(1 - R)/2$ unique decoding bound for RS codes.
- Picking $s \approx 1/\varepsilon$, $m \approx 1/\varepsilon^2$, again $\tau \geqslant 1 - R - \varepsilon$.

Folded Reed-Solomon list decoding

- Following Reed-Solomon list decoder, two steps: (i) interpolation, and (ii) solution/root finding.

Folded Reed-Solomon list decoding

- Following Reed-Solomon list decoder, two steps:
  (i) interpolation, and (ii) solution/root finding.
- For folded codes, *multivariate* interpolation is used.
  In linear-algebraic version, interpolate a polynomial of following form [Vadhan] (for some $s \in \{1, 2, \ldots, m\}$)

$$A_0(X) + A_1(X)Y_1 + A_2(X)Y_2 + \cdots + A_s(X)Y_s$$

Folded Reed-Solomon list decoding

- Following Reed-Solomon list decoder, two steps: (i) interpolation, and (ii) solution/root finding.
- For folded codes, *multivariate* interpolation is used. In linear-algebraic version, interpolate a polynomial of following form [Vadhan] (for some $s \in \{1, 2, \dots, m\}$)

$$A_0(X) + A_1(X)Y_1 + A_2(X)Y_2 + \cdots + A_s(X)Y_s$$

- Algebraic crux is to find all degree k solutions $f \in \mathbb{F}_q[X]$ to

$$A_0(X) + A_1(X)f(X) + A_2(X)f(\gamma X) + \cdots + A_s(X)f(\gamma^{s-1}X) = 0$$

Folded Reed-Solomon list decoding

- Following Reed-Solomon list decoder, two steps:
  (i) interpolation, and (ii) solution/root finding.
- For folded codes, *multivariate* interpolation is used.
  In linear-algebraic version, interpolate a polynomial of following form [Vadhan] (for some $s \in \{1, 2, \ldots, m\}$)

$$A_0(X) + A_1(X)Y_1 + A_2(X)Y_2 + \cdots + A_s(X)Y_s$$

- Algebraic crux is to find all degree k solutions $f \in \mathbb{F}_q[X]$ to

$$A_0(X) + A_1(X)f(X) + A_2(X)f(\gamma X) + \cdots + A_s(X)f(\gamma^{s-1}X) = 0$$

- Next: details of these steps
  - $s = 1$ corresponds to *unique* decoding: $f(X) = -A_0(X)/A_1(X)$.

Interpolation

$$\left(\begin{bmatrix} f(1) \\ f(\gamma) \\ \vdots \\ f(\gamma^{m-1}) \end{bmatrix}, \begin{bmatrix} f(\gamma^m) \\ f(\gamma^{m+1}) \\ \vdots \\ f(\gamma^{2m-1}) \end{bmatrix}, \ldots \right) \rightsquigarrow \left(\begin{bmatrix} y_0 \\ y_1 \\ \vdots \\ y_{m-1} \end{bmatrix}, \ldots, \begin{bmatrix} y_{n-m} \\ y_{n-m+1} \\ \vdots \\ y_{n-1} \end{bmatrix} \right)$$

Find $A_0, A_1, \ldots, A_s \in \mathbb{F}_q[X]$ such that
$Q(X, Y_1, \cdots, Y_s) = A_0(X) + A_1(X)Y_1 + \cdots + A_s(X)Y_s$ satisfies

$$Q(\gamma^i, y_i, y_{i+1}, \ldots, y_{i+s-1}) = 0 \quad \forall\ i, i \mod m \in \{0, 1, \ldots, m-s\} .$$

Interpolation

$$\left(\begin{bmatrix} f(1) \\ f(\gamma) \\ \vdots \\ f(\gamma^{m-1}) \end{bmatrix}, \begin{bmatrix} f(\gamma^m) \\ f(\gamma^{m+1}) \\ \vdots \\ f(\gamma^{2m-1}) \end{bmatrix}, \ldots \right) \rightsquigarrow \left(\begin{bmatrix} y_0 \\ y_1 \\ \vdots \\ y_{m-1} \end{bmatrix}, \ldots, \begin{bmatrix} y_{n-m} \\ y_{n-m+1} \\ \vdots \\ y_{n-1} \end{bmatrix} \right)$$

Find $A_0, A_1, \ldots, A_s \in \mathbb{F}_q[X]$ such that
$Q(X, Y_1, \cdots, Y_s) = A_0(X) + A_1(X)Y_1 + \cdots + A_s(X)Y_s$ satisfies

$$Q(\gamma^i, y_i, y_{i+1}, \ldots, y_{i+s-1}) = 0 \quad \forall\ i, i \mod m \in \{0, 1, \ldots, m-s\} .$$

- Restrict $\deg(A_0) < D + k$, $\deg(A_j) \leqslant D$ for $1 \leqslant j \leqslant s$.

Interpolation

$$\left(\begin{bmatrix} f(1) \\ f(\gamma) \\ \vdots \\ f(\gamma^{m-1}) \end{bmatrix}, \begin{bmatrix} f(\gamma^m) \\ f(\gamma^{m+1}) \\ \vdots \\ f(\gamma^{2m-1}) \end{bmatrix}, \ldots \right) \rightsquigarrow \left(\begin{bmatrix} y_0 \\ y_1 \\ \vdots \\ y_{m-1} \end{bmatrix}, \ldots, \begin{bmatrix} y_{n-m} \\ y_{n-m+1} \\ \vdots \\ y_{n-1} \end{bmatrix} \right)$$

Find $A_0, A_1, \ldots, A_s \in \mathbb{F}_q[X]$ such that
$Q(X, Y_1, \cdots, Y_s) = A_0(X) + A_1(X)Y_1 + \cdots + A_s(X)Y_s$ satisfies

$$Q(\gamma^i, y_i, y_{i+1}, \ldots, y_{i+s-1}) = 0 \quad \forall\ i, i \mod m \in \{0, 1, \ldots, m-s\} \ .$$

- Restrict $\deg(A_0) < D + k$, $\deg(A_j) \leqslant D$ for $1 \leqslant j \leqslant s$.
- $> (s+1)D + k$ degrees of freedom/unknowns
- $n' := N(m - s + 1)$ constraints ($N = n/m$ is block length of folded code)

Linear interpolation step

Received word $\left(\begin{bmatrix} y_0 \\ y_1 \\ \vdots \\ y_{m-1} \end{bmatrix}, \ldots, \begin{bmatrix} y_{n-m} \\ y_{n-m+1} \\ \vdots \\ y_{n-1} \end{bmatrix} \right)$

When $D = (n' - k)/(s+1)$, can find $A_0, A_1, \ldots, A_s \in \mathbb{F}_q[X]$, not all zero, such that
$Q(X, Y_1, \ldots, Y_s) = A_0(X) + A_1(X)Y_1 + \cdots + A_s(X)Y_s$ satisfies

1. $Q(\gamma^i, y_i, y_{i+1}, \ldots, y_{i+s-1}) = 0$ for $i \mod m \leqslant m - s$.
2. For any degree $< k$ polynomial f, $Q(X, f(X), f(\gamma X), \ldots, f(\gamma^{s-1}X))$ has degree $< D + k = (n' + sk)/(s+1)$

Second fact follows from degree restrictions on A_i's.

Algebraic handle on message polynomials

Lemma

If $t \geqslant \frac{n'+sk}{(m-s+1)(s+1)}$ *values of* $j \in \{0, 1, \ldots, N-1\}$ *satisfy* $f(\gamma^{jm}), f(\gamma^{jm+1}, \ldots, f(\gamma^{jm+m-1})) = (y_{jm}, \ldots, y_{jm+m-1})$*, then* $A_0(X) + A_1(X)f(X) + A_2(X)f(\gamma X) + \cdots + A_s(X)f(\gamma^{s-1}X) = 0$.

Algebraic handle on message polynomials

Lemma

If $t \geqslant \frac{n'+sk}{(m-s+1)(s+1)}$ *values of* $j \in \{0,1,\ldots,N-1\}$ *satisfy* $f(\gamma^{jm}), f(\gamma^{jm+1}, \ldots, f(\gamma^{jm+m-1})) = (y_{jm}, \ldots, y_{jm+m-1})$*, then* $A_0(X) + A_1(X)f(X) + A_2(X)f(\gamma X) + \cdots + A_s(X)f(\gamma^{s-1}X) = 0$.

$$\left(\begin{bmatrix} f(1) \\ f(\gamma) \\ \vdots \\ f(\gamma^{m-1}) \end{bmatrix}, \ldots \right) \rightsquigarrow \left(\begin{bmatrix} y_0 \\ y_1 \\ \vdots \\ y_{m-1} \end{bmatrix}, \ldots, \begin{bmatrix} y_{n-m} \\ y_{n-m+1} \\ \vdots \\ y_{n-1} \end{bmatrix} \right)$$

Key Fact: If codeword and **y** agree on t columns, then $(f(\gamma^i), f(\gamma^{i+1}), \ldots, f(\gamma^{i+s-1})) = (y_i, y_{i+1}, \ldots, y_{i+s-1})$ for at least $(m-s+1)t$ values of i.

The decoding radius

$N = n/m$ is block length of m-folded code.
$t = (1-\tau)N$ is the number of correct columns.

Decoding condition is $(1-\tau)N \geqslant \frac{N(m-s+1)+sk}{s+1)(m-s+1)}$.

The decoding radius

$N = n/m$ is block length of m-folded code.
$t = (1-\tau)N$ is the number of correct columns.

Decoding condition is $(1-\tau)N \geqslant \frac{N(m-s+1)+sk}{s+1)(m-s+1)}$.

Since degree $k = R \cdot n = R \cdot Nm$, above is met for

$$\tau \leqslant \frac{s}{s+1}\left(1 - \frac{m}{m-s+1}R\right) .$$

The decoding radius

$N = n/m$ is block length of m-folded code.
$t = (1-\tau)N$ is the number of correct columns.

Decoding condition is $(1-\tau)N \geqslant \frac{N(m-s+1)+sk}{s+1)(m-s+1)}$.

Since degree $k = R \cdot n = R \cdot Nm$, above is met for

$$\tau \leqslant \frac{s}{s+1}\left(1 - \frac{m}{m-s+1}R\right) \; .$$

- Error fraction approches $\frac{s}{s+1}(1-R)$ for large $m \gg s$.
- Can achieve $\tau = 1 - R - \varepsilon$ by taking $s \gtrsim 1/\varepsilon$ and $m \gtrsim 1/\varepsilon^2$.

Recovering list of messages

Following interpolation step, algebraic crux is to find all degree $< k$ solutions $f \in \mathbb{F}_q[X]$ to the equation

$$A_0(X) + A_1(X)f(X) + A_2(X)f(\gamma X) + \cdots + A_s(X)f(\gamma^{s-1}X) = 0$$

Recovering list of messages

Following interpolation step, algebraic crux is to find all degree $< k$ solutions $f \in \mathbb{F}_q[X]$ to the equation

$$A_0(X) + A_1(X)f(X) + A_2(X)f(\gamma X) + \cdots + A_s(X)f(\gamma^{s-1}X) = 0$$

[G.'11] Observe that the above is an $\mathbb{F}_q$-linear system (in the coefficients of f)

- So we can solve for f and pin down possibilities to an affine subspace!
- To control list size, need to bound dimension of solution space.

Solving for f

Illustrate with $s = 2$

$$A_0(X) + A_1(X)f(X) + A_2(X)f(\gamma X) = 0 \qquad (\clubsuit)$$

Let $A_i(X) = a_{i0} + a_{i1}X + a_{i2}X^2 + \cdots$ (wlog, not all $a_{i0} = 0$), and let $f = f_0 + f_1X + \cdots, +f_{k-1}X^{k-1}$.

Solving for f

Illustrate with $s = 2$

$$A_0(X) + A_1(X)f(X) + A_2(X)f(\gamma X) = 0 \qquad (\clubsuit)$$

Let $A_i(X) = a_{i0} + a_{i1}X + a_{i2}X^2 + \cdots$ (wlog, not all $a_{i0} = 0$), and let $f = f_0 + f_1 X + \cdots, + f_{k-1}X^{k-1}$.

($\clubsuit$) is the lower-triangular linear system:

$$\begin{aligned}
a_{00} + (a_{10} + a_{20}) \cdot f_0 &= 0 \\
a_{01} + (\cdots) \cdot f_0 + (a_{10} + a_{20}\gamma) \cdot f_1 &= 0 \\
a_{02} + (\cdots) \cdot f_0 + (\cdots) \cdot f_1 + (a_{10} + a_{20}\gamma^2) \cdot f_2 &= 0 \\
&\vdots
\end{aligned}$$

Solving for f

Illustrate with $s = 2$

$$A_0(X) + A_1(X)f(X) + A_2(X)f(\gamma X) = 0 \qquad (\clubsuit)$$

Let $A_i(X) = a_{i0} + a_{i1}X + a_{i2}X^2 + \cdots$ (wlog, not all $a_{i0} = 0$), and let $f = f_0 + f_1 X + \cdots, + f_{k-1}X^{k-1}$.

($\clubsuit$) is the lower-triangular linear system:

$$\begin{aligned} a_{00} + (a_{10} + a_{20}) \cdot f_0 &= 0 \\ a_{01} + (\cdots) \cdot f_0 + (a_{10} + a_{20}\gamma) \cdot f_1 &= 0 \\ a_{02} + (\cdots) \cdot f_0 + (\cdots) \cdot f_1 + (a_{10} + a_{20}\gamma^2) \cdot f_2 &= 0 \\ &\vdots \end{aligned}$$

At most one i s.t. $a_{10} + a_{20}\gamma^i = 0 \Longrightarrow$ soln. space dimension $\leqslant 1$.

Solving for f

Illustrate with $s = 2$

$$A_0(X) + A_1(X)f(X) + A_2(X)f(\gamma X) = 0 \qquad (\clubsuit)$$

Let $A_i(X) = a_{i0} + a_{i1}X + a_{i2}X^2 + \cdots$ (wlog, not all $a_{i0} = 0$), and let $f = f_0 + f_1 X + \cdots, +f_{k-1}X^{k-1}$.

$(\clubsuit)$ is the lower-triangular linear system:

$$\begin{aligned} a_{00} + (a_{10} + a_{20}) \cdot f_0 &= 0 \\ a_{01} + (\cdots) \cdot f_0 + (a_{10} + a_{20}\gamma) \cdot f_1 &= 0 \\ a_{02} + (\cdots) \cdot f_0 + (\cdots) \cdot f_1 + (a_{10} + a_{20}\gamma^2) \cdot f_2 &= 0 \\ &\vdots \end{aligned}$$

At most one i s.t. $a_{10} + a_{20}\gamma^i = 0 \implies$ soln. space dimension $\leqslant 1$.
For general s, solns. lie in dim. $\leqslant s-1$ subspace ($\therefore$ list size $\leqslant q^{s-1}$)

Summary

Folded RS decoding

For folded RS code of rate R, can list decode up to radius $\approx \frac{s}{s+1}(1-R)$ pinning down candidate messages to an affine subspace of dimension $\leqslant s-1$.

List size bound is q^{s-1}, or $q^{\Omega(1/\varepsilon)}$ when $s \approx 1/\varepsilon$.

Decoding complexity also similar, dominated by sifting through the $s-1$-dimensional subspace for close-by codewords.

Also $q > N$ (inherent to Reed-Solomon)

Analogous results for "derivative/multiplicity codes" [G.-Wang] [Kopparty]

Derivative/multiplicity codes

Definition (Order-m Derivative codes)

$a_1, a_2, \ldots, a_n$ distinct elements of $\mathbb{F}_q$, $\mathrm{char}(\mathbb{F}_q) > k$. Message $f \in \mathbb{F}_q[X]_{<k}$ is mapped to codeword

$$\left(\begin{bmatrix} f(a_1) \\ f'(a_1) \\ \vdots \\ f^{(m-1)}(a_1) \end{bmatrix}, \begin{bmatrix} f(a_2) \\ f'(a_2) \\ \vdots \\ f^{(m-1)}(a_2) \end{bmatrix}, \ldots, \begin{bmatrix} f(a_n) \\ f'(a_n) \\ \vdots \\ f^{(m-1)}(a_n) \end{bmatrix} \right) .$$

Alphabet size q^m; block length $= n$; rate $R = k/(nm)$

For large $m \approx 1/\varepsilon^2$, can be list decoded from $1 - R - \varepsilon$ error fraction.

Summary

Optimal rate list decoding

Explicit (folded Reed-Solomon, or derivative) codes of rate R list-decodable up to error fraction $1 - R - \varepsilon$.

- Alphabet size $> N^{1/\varepsilon^2}$, and list-size $N^{1/\varepsilon}$.

Summary

Optimal rate list decoding

Explicit (folded Reed-Solomon, or derivative) codes of rate R list-decodable up to error fraction $1 - R - \varepsilon$.

- Alphabet size $> N^{1/\varepsilon^2}$, and list-size $N^{1/\varepsilon}$.

Algorithm also gives soft decodability.

Using this in a concatenation scheme followed by expander graph based symbol redistribution:

$\Rightarrow$ reduce alphabet size $\exp(1/\varepsilon^4)$ (*independent of block length*)

  - $\exp(1/\varepsilon)$ is a lower bound on alphabet size.

- But decoding complexity and list-size high (inherited from outer folded RS code)

Talk plan

1. (List) decoding Reed-Solomon codes
2. Folded Reed-Solomon codes: Linear-algebraic list decoding
3. **Subspace-evasive pre-coding**
    - Extensions to algebraic-geometric & rank-metric codes
4. Concluding remarks, Open challenges

Pre-coding idea

In linear-algebraic list decoding, the list of candidate messages are contained within a s-dimensional subspace.

Simple yet influential idea: *Instead of all degree k polys as messages, only allow a carefully chosen subset which doesn't intersect any low-dimensional subspace too much.*

Pre-coding idea

In linear-algebraic list decoding, the list of candidate messages are contained within a s-dimensional subspace.

Simple yet influential idea: *Instead of all degree k polys as messages, only allow a carefully chosen subset which doesn't intersect any low-dimensional subspace too much.*

Subspace-evasive sets

A subset $S \subset \mathbb{F}_q^k$ is said to be (s, ℓ)-subspace evasive if for *all* s-dimensional subspaces W of $\mathbb{F}_q^k$, $|S \cap W| \leqslant \ell$.

Pre-coding idea

In linear-algebraic list decoding, the list of candidate messages are contained within a s-dimensional subspace.

Simple yet influential idea: *Instead of all degree k polys as messages, only allow a carefully chosen subset which doesn't intersect any low-dimensional subspace too much.*

Subspace-evasive sets

A subset $S \subset \mathbb{F}_q^k$ is said to be (s, ℓ)-subspace evasive if for *all* s-dimensional subspaces W of $\mathbb{F}_q^k$, $|S \cap W| \leqslant \ell$.

Observation: Restricting (coefficients of) message polynomials to belong to such a subspace-evasive set brings down list size to ℓ.

But how much does this cost in terms of rate?

Subspace-evasive sets

Natural notion (in pseudorandomness, geometry).

Considered in work on bipartite Ramsey problem [Pudlák-Rödl'05]

Subspace-evasive sets

Natural notion (in pseudorandomness, geometry).

Considered in work on bipartite Ramsey problem [Pudlák-Rödl'05]

Easy application of probabilistic method gives:

Lemma

A random subset of $\mathbb{F}_q^k$ of size $q^{(1-\varepsilon)k}$ is $(s, O(s/\varepsilon))$-subspace evasive w.h.p. (for $s \lesssim \varepsilon k$).

Factor $(1-\varepsilon)$ loss in rate suffices for significant pruning of the solution subspaces!

How to represent and encode into the subspace-evasive set?

Good subcodes of folded RS code

Prob. method works even for $O(s/\varepsilon)$-*wise independent subsets*, which admit compact representation and efficient encoding.

Via a pseudorandom construction of subspace-evasive sets, can get

- Monte Carlo consruction of a subcode of folded RS codes with list size $O(1/\varepsilon)$ (matching existential random coding bound!)

Upshot

Monte Carlo construction of efficiently $(1 - R - \varepsilon, O(1/\varepsilon))$-list decodable subcodes of folded Reed-Solomon codes.

Explicit construction?

Explicit subspace-evasive sets

Theorem (Dvir-Lovett'12)

Explicit *construction of a* $(s, (s/\varepsilon)^{O(s)})$*-subspace evasive subset of* $\mathbb{F}_q^k$ *of size* $q^{(1-\varepsilon)k}$.

Approach: An algberaic variety cut out by s polynomial equations such that the intersection with **every** s-dimensional affine space is a zero-dimensional variety. (Intersection size bound via *Bézout's theorem.*)

Explicit subspace-evasive sets

Theorem (Dvir-Lovett'12)

Explicit *construction of a* $(s, (s/\varepsilon)^{O(s)})$*-subspace evasive subset of* $\mathbb{F}_q^k$ *of size* $q^{(1-\varepsilon)k}$.

Approach: An algberaic variety cut out by s polynomial equations such that the intersection with **every** s-dimensional affine space is a zero-dimensional variety. (Intersection size bound via *Bézout's theorem.*)

Upshot

Explicit construction of efficiently $(1 - R - \varepsilon, \exp(\tilde{O}(1/\varepsilon)))$-list decodable codes.

Explicit subspace-evasive sets

Theorem (Dvir-Lovett'12)

Explicit *construction of a* $(s, (s/\varepsilon)^{O(s)})$*-subspace evasive subset of* $\mathbb{F}_q^k$ *of size* $q^{(1-\varepsilon)k}$.

Approach: An algberaic variety cut out by s polynomial equations such that the intersection with **every** s-dimensional affine space is a zero-dimensional variety. (Intersection size bound via *Bézout's theorem.*)

Upshot

Explicit construction of efficiently $(1 - R - \varepsilon, \exp(\tilde{O}(1/\varepsilon)))$-list decodable codes.

Beautiful challenge: Explicit construction of subspace evasive set with $\exp(o(s))$ intersection bound?

Summary

Optimal rate list decoding

Subcodes of folded Reed-Solomon codes with rate R, list decoding radius $1 - R - \varepsilon$, list-size constant depending only on ε.

But the alphabet size in $N^{\Omega(1/\varepsilon^2)}$.

- Extensions to *algebraic-geometric codes* (Garcia-Stichtenoth) gives alphabet size $\approx \exp(O(1/\varepsilon^2))$. [G.-Xing'12,'13]

Summary

Optimal rate list decoding

Subcodes of folded Reed-Solomon codes with rate R, list decoding radius $1 - R - \varepsilon$, list-size constant depending only on ε.

But the alphabet size in $N^{\Omega(1/\varepsilon^2)}$.

- Extensions to *algebraic-geometric codes* (Garcia-Stichtenoth) gives alphabet size $\approx \exp(O(1/\varepsilon^2))$. [G.-Xing'12,'13]
- Can also construct **explicit rank-metric codes** of rate R efficiently list-decodable up to $1 - R - \varepsilon$ fraction of rank-metric errors. [G.-Xing'13], [G.-Wang'14]

Summary

Optimal rate list decoding

Subcodes of folded Reed-Solomon codes with rate R, list decoding radius $1 - R - \varepsilon$, list-size constant depending only on ε.

But the alphabet size in $N^{\Omega(1/\varepsilon^2)}$.

- Extensions to *algebraic-geometric codes* (Garcia-Stichtenoth) gives alphabet size $\approx \exp(O(1/\varepsilon^2))$. [G.-Xing'12,'13]
- Can also construct **explicit rank-metric codes** of rate R efficiently list-decodable up to $1 - R - \varepsilon$ fraction of rank-metric errors. [G.-Xing'13], [G.-Wang'14]
- Based on subspace designs (a variant of subspace-evasive sets)

Next: Illustrate emergence of the subspace design notion in decoding Reed-Solomon codes themselves.

Reed-Solomon codes again

Definition (RS codes with evaluation points in a subfield)

Messages = polynomials $f \in \mathbb{F}_{q^m}[X]$ of degree $< k$. Encoding:

$$f \mapsto (f(\alpha_1), f(\alpha_2), \cdots, f(\alpha_q))$$

where α_i are all the elements of $\mathbb{F}_q$.

Rate $R = k/q$; block length $= q$; alphabet size $= q^m$.

Reed-Solomon codes again

Definition (RS codes with evaluation points in a subfield)

Messages = polynomials $f \in \mathbb{F}_{q^m}[X]$ of degree $< k$. Encoding:

$$f \mapsto (f(\alpha_1), f(\alpha_2), \cdots, f(\alpha_q))$$

where α_i are all the elements of $\mathbb{F}_q$.

Rate $R = k/q$; block length $= q$; alphabet size $= q^m$.

- Being a RS code, we can list decode above up to radius $\tau = 1 - \sqrt{R}$ [G.-Sudan]
- [G.-Xing] A *subcode* of above code can be efficiently list decoded up to radius $1 - R - \varepsilon$ when $m \approx 1/\varepsilon^2$.

Frobenius

For $f \in \mathbb{F}_{q^m}[X]$ equal to $f_0 + f_1 X + \cdots + f_{k-1} X^{k-1}$, define the polynomial $f^\sigma \in \mathbb{F}_{q^m}[X]$ as

$$f_0^q + f_1^q X + f_2^q X^2 + \cdots + f_{k-1}^q X^{k-1}.$$

Key fact

For $\alpha \in \mathbb{F}_q$, $f^\sigma(\alpha) = f(\alpha)^q$.

Frobenius

For $f \in \mathbb{F}_{q^m}[X]$ equal to $f_0 + f_1 X + \cdots + f_{k-1}X^{k-1}$, define the polynomial $f^\sigma \in \mathbb{F}_{q^m}[X]$ as

$$f_0^q + f_1^q X + f_2^q X^2 + \cdots + f_{k-1}^q X^{k-1}.$$

Key fact

For $\alpha \in \mathbb{F}_q$, $f^\sigma(\alpha) = f(\alpha)^q$.

Proof.

$$f(\alpha)^q = \left(\sum_{j=0}^{k-1} f_j \alpha^{j-1}\right)^q = \sum_{j=0}^{k-1} f_j^q \alpha^{(j-1)q}$$
$$= \sum_{j=0}^{k-1} f_j^q \alpha^{j-1} = f^\sigma(\alpha).$$

□

Decoding idea

One can "manufacture" evaluations of f^σ on $\mathbb{F}_q$ given those of f.

- In folded RS case, evaluations of $f(\gamma X)$ given those of $f(X)$.

Decoding idea

One can "manufacture" evaluations of f^σ on $\mathbb{F}_q$ given those of f.

- In folded RS case, evaluations of $f(\gamma X)$ given those of $f(X)$.

Similar multivariate interpolation approach can decode up to radius $\frac{s}{s+1}(1-R)$, pinning down message polynomials f to solutions of

$$A_0(X) + A_1(X)f(X) + A_2(X)f^\sigma(X) + \cdots + A_s(X)f^{\sigma^{s-1}}(X) = 0$$

Decoding idea

One can "manufacture" evaluations of f^σ on $\mathbb{F}_q$ given those of f.

- In folded RS case, evaluations of $f(\gamma X)$ given those of $f(X)$.

Similar multivariate interpolation approach can decode up to radius $\frac{s}{s+1}(1-R)$, pinning down message polynomials f to solutions of

$$A_0(X) + A_1(X)f(X) + A_2(X)f^\sigma(X) + \cdots + A_s(X)f^{\sigma^{s-1}}(X) = 0$$

Again, solutions $f \in \mathbb{F}_{q^m}[X]$ form an $\mathbb{F}_q$-affine subspace!

- Can show $(s-1)k$ bound on overall dimension (non-trivially smaller than mk, but still too large).

Periodic subspaces

The solution subspace has additional s-periodic structure:

- $\exists$ a subspace $W \subset \mathbb{F}_{q^m}$ of dimension $< s$ such that f_j belongs to a coset of W (that only depends on $f_0, f_1, \ldots, f_{j-1}$).

Key idea: Exploit fact that each f_j is in coset of the **same** subspace W.

- Restrict $f_j \in H_j$ for subspaces H_j that are "well spread out" (so they don't intersect W too often)

Subspace designs

Definition (Subspace design (G.-Xing'13))

A collection of subspaces $H_0, H_2, \ldots, H_{k-1} \subset \mathbb{F}_q^m$ is an (s,d)-*subspace design* if $\forall$ s-dimensional subspaces $W \subset \mathbb{F}_q^m$,

$$\sum_{j=0}^{k-1} \dim(H_j \cap W) \leqslant d.$$

Subspace designs

Definition (Subspace design (G.-Xing'13))

A collection of subspaces $H_0, H_2, \ldots, H_{k-1} \subset \mathbb{F}_q^m$ is an (s, d)*-subspace design* if $\forall$ s-dimensional subspaces $W \subset \mathbb{F}_q^m$,

$$\sum_{j=0}^{k-1} \dim(H_j \cap W) \leqslant d.$$

Theorem

For H_j's from an (s, d)-subspace design, intersection of an s-periodic subspace with $H_0 \times H_1 \times \cdots \times H_{k-1}$ is an affine space of dimension at most d.

<u>Note</u>: Now even the pruning is linear-algebraic! (impose additional linear constants $f_j \in H_j$ on top of interpolation equation)

Constructing subspace designs

Spreads

Explicit construction of a large collection $\{H_j\}$ of $m/2$-dimensional subspaces of $\mathbb{F}_q^m$ are known such that $H_j \cap H_{j'} = \{0\}$ for $j \neq j'$.

- These give a (s, s)-subspace design.

Constructing subspace designs

Spreads

Explicit construction of a large collection $\{H_j\}$ of $m/2$-dimensional subspaces of $\mathbb{F}_q^m$ are known such that $H_j \cap H_{j'} = \{0\}$ for $j \neq j'$.

- These give a (s, s)-subspace design.
- But factor $1/2$ loss in rate.

We need a design with subspaces of dimension $(1 - \varepsilon)m$.

Subspace designs of large dimension

Lemma (Probabilistic method)

For dimension $(1-\varepsilon)m$*, random collection of size* $\approx q^{\varepsilon m}$ *is an* $(s, s/\varepsilon)$*-subspace design w.h.p.*

Subspace designs of large dimension

Lemma (Probabilistic method)

For dimension $(1-\varepsilon)m$, random collection of size $\approx q^{\varepsilon m}$ is an $(s, s/\varepsilon)$-subspace design w.h.p.

Theorem (G.-Kopparty'13)

Explict construction of $(s, s/\varepsilon)$-subspace design of size q (with subspaces of dimension $(1-\varepsilon)m$.
Also explicit $(s, s^2/\varepsilon)$-subspace design of larger size $q^{\varepsilon m/s}$.

Yields explicit subcodes of these RS codes that are list-decodable up to radius $1 - R - \varepsilon$.

Similar idea works for Gabidulin codes in rank-metric setting.

List-decodable subcodes of RS codes

Upshot

There is an $\mathbb{F}_q$-**linear** subcode of RS code over $\mathbb{F}_{q^m}$ with evaluation points in $\mathbb{F}_q$ that is decodable up to optimal radius $(1 - R - \varepsilon)$.

(list contained in a subspace of dimension $1/\varepsilon^2$)

Subspace design construction

Curiously, the subspace design construction is itself based on (variants of) Reed-Solomon codes.

Recall, we want many dimension $(1-\varepsilon)m$ subspaces $H_1, \ldots, H_M$ of $\mathbb{F}_q^m$ such that for every W, $\dim(W) = s$, the number of H_i's such that $H_i \cap W \neq \{0\}$ is small.

Subspace design construction

Curiously, the subspace design construction is itself based on (variants of) Reed-Solomon codes.

Recall, we want many dimension $(1-\varepsilon)m$ subspaces $H_1, \ldots, H_M$ of $\mathbb{F}_q^m$ such that for every W, $\dim(W) = s$, the number of H_i's such that $H_i \cap W \neq \{0\}$ is small.

Baby case: $s = 1$.

- Identity $\mathbb{F}_q^m$ with $\mathbb{F}_q[X]_{<m}$ (degree $< m$ polynomials over $\mathbb{F}_q$).
- For $a \in \mathbb{F}_q$, define $H_a = \{f \in \mathbb{F}_q[X]_{<m} \mid \mathrm{mult}(f, a) \geqslant \varepsilon m\}$ (which has dimension $(1-\varepsilon)m$).

Subspace design construction

Curiously, the subspace design construction is itself based on (variants of) Reed-Solomon codes.

Recall, we want many dimension $(1-\varepsilon)m$ subspaces $H_1, \ldots, H_M$ of $\mathbb{F}_q^m$ such that for every W, $\dim(W) = s$, the number of H_i's such that $H_i \cap W \neq \{0\}$ is small.

Baby case: $s = 1$.

- Identity $\mathbb{F}_q^m$ with $\mathbb{F}_q[X]_{<m}$ (degree $< m$ polynomials over $\mathbb{F}_q$).
- For $a \in \mathbb{F}_q$, define $H_a = \{f \in \mathbb{F}_q[X]_{<m} \mid \mathrm{mult}(f, a) \geqslant \varepsilon m\}$ (which has dimension $(1-\varepsilon)m$).
- Let $W = \mathrm{span}(\{g\})$ for some nonzero $g \in \mathbb{F}_q[X]_{<m}$.
- $W \cap H_a \neq \{0\}$ iff g has $\geqslant \varepsilon m$ zeroes at a.
  Happens at most $1/\varepsilon$ times!

Subspace design construction

Same construction works for larger dimensional subspaces W.

If $W = \mathrm{span}(g_1, g_2, \ldots, g_s)$, proof via *Wronskian*:

$$\begin{vmatrix} g_1(X) & g_2(X) & \cdots & g_s(X) \\ g_1'(X) & g_2'(X) & \cdots & g_s'(X) \\ \vdots & \vdots & \ddots & \\ g_1^{(s-1)}(X) & g_2^{(s-1)}(X) & \cdots & g_s^{(s-1)}(X) \end{vmatrix} .$$

This is based on "derivative codes"; requires large characteristic.

Better construction via folded Reed-Solomon codes.

Talk plan

1. (List) decoding Reed-Solomon codes
2. Folded Reed-Solomon codes: Linear-algebraic list decoding
3. Subspace-evasive pre-coding
   - Extensions to algebraic-geometric & rank-metric codes
4. **Concluding remarks, Open challenges**

Wrap-up

- Variants of Reed-Solomon codes enable list decoding up to radius approaching optimal $1 - R$ bound with rate R.
- *Linear-algebraic approach* pins down candidates to a low-dimensional (or structured) subspace.

Wrap-up

- Variants of Reed-Solomon codes enable list decoding up to radius approaching optimal $1 - R$ bound with rate R.
- *Linear-algebraic approach* pins down candidates to a low-dimensional (or structured) subspace.
- Decoding approach versatile and applies to variants of
  - Algebraic-geometric codes (achieving constant alphabet size)
  - Gabidulin codes (optimal radius decoding in rank metric)
  - Koetter-Kschischang subspace codes

Wrap-up

- Variants of Reed-Solomon codes enable list decoding up to radius approaching optimal $1 - R$ bound with rate R.
- *Linear-algebraic approach* pins down candidates to a low-dimensional (or structured) subspace.
- Decoding approach versatile and applies to variants of
    - Algebraic-geometric codes (achieving constant alphabet size)
    - Gabidulin codes (optimal radius decoding in rank metric)
    - Koetter-Kschischang subspace codes
- Reduce list size by pruning subspace of candidate messages using (variants of) *subspace-evasive sets* or *subspace designs*.

Open Problems

Large alphabet list decoding quite well understood.
But many interesting challenges remain.

Open Problems

Large alphabet list decoding quite well understood.
But many interesting challenges remain.

- List decoding capability of Reed-Solomon codes itself?
- Explicit optimal rate **binary** list-decodable codes?
  - Tackle case of erasures (list decoding up to $1 - R - \varepsilon$ erasure fraction with rate R)?

Open Problems

Large alphabet list decoding quite well understood.
But many interesting challenges remain.

- List decoding capability of Reed-Solomon codes itself?
- Explicit optimal rate **binary** list-decodable codes?
  - Tackle case of erasures (list decoding up to $1 - R - \varepsilon$ erasure fraction with rate R)?
- List decoding Gabidulin codes beyond half the distance?
  - Certain variants list decodable up to almost the distance
- Combinatorial bounds for list decoding
  - (p, L)-list decodable binary code of rate $1 - h(p) - \varepsilon$:
    What's the smallest possible list-size $L = L(\varepsilon)$?
    We have $\log(1/\varepsilon) \lesssim L(\varepsilon) \lesssim 1/\varepsilon$