Efficient List-Decoding with Constant Alphabet and List Sizes

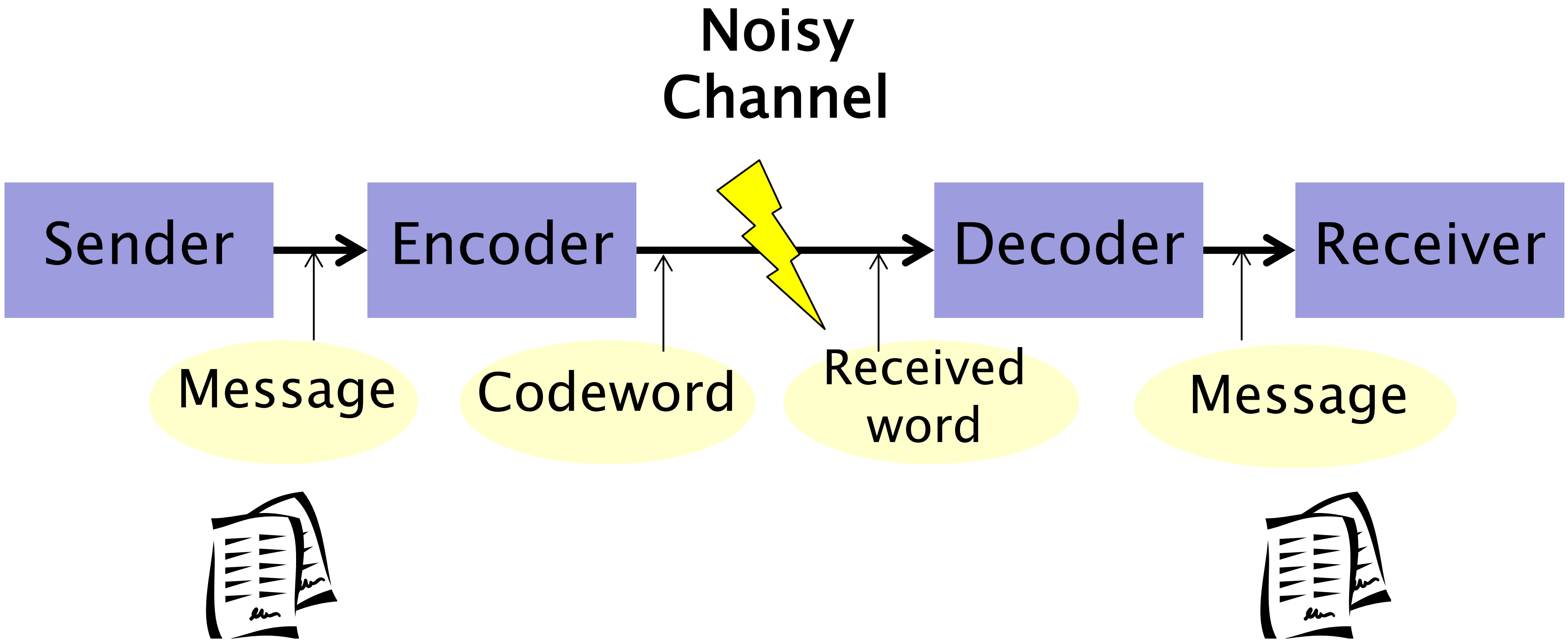
Zeyu Guo

University of Haifa

Joint work with Noga Ron-Zewi

Error-Correcting Codes

[Hamming, Shannon '40s]



Error-Correcting Codes

Code: $C: \Sigma^k \to \Sigma^n$, maps messages to codewords

- Alphabet Σ, message length k, codeword length n

Rate: $R = \frac{k}{n} \sim$ redundancy in encoding

Minimum distance:

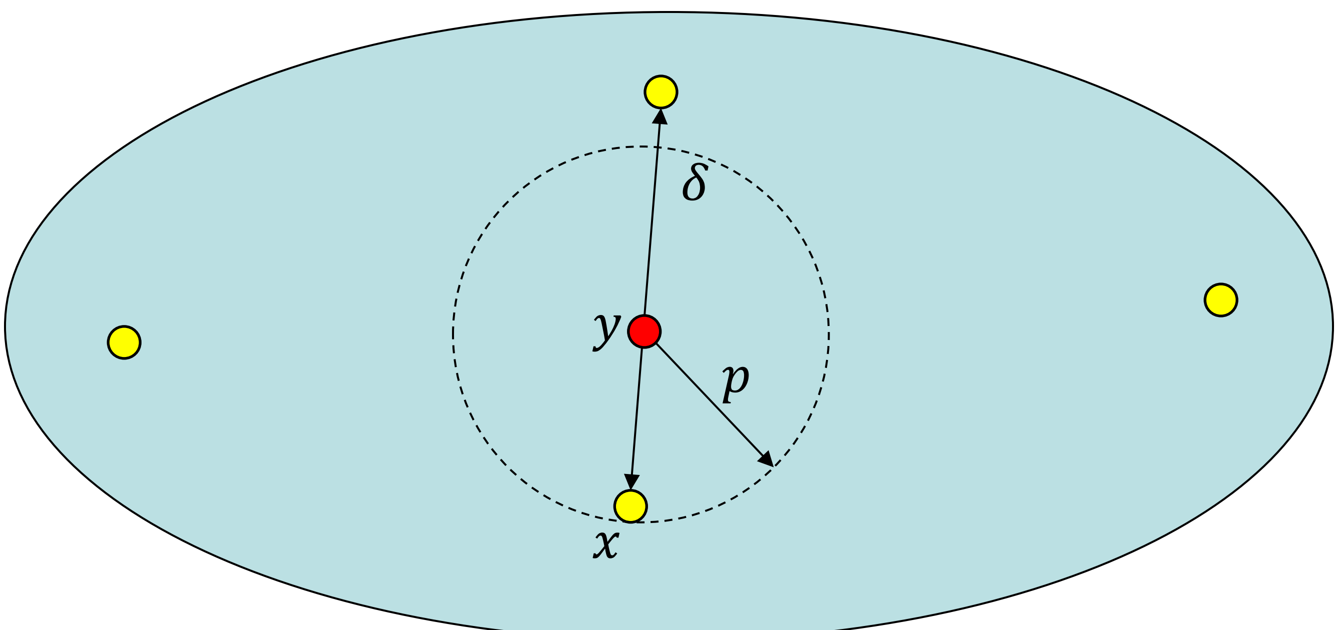
- For $x, y \in \Sigma^n$, the (relative) distance $\mathrm{dist}(x, y)$ is the fraction of coordinates where x and y differ
- The (relative) minimum distance of C is

$$\delta = \min(\mathrm{dist}(x, y) : x, y \in C, x \neq y)$$

Unique Decoding

Unique decoding:

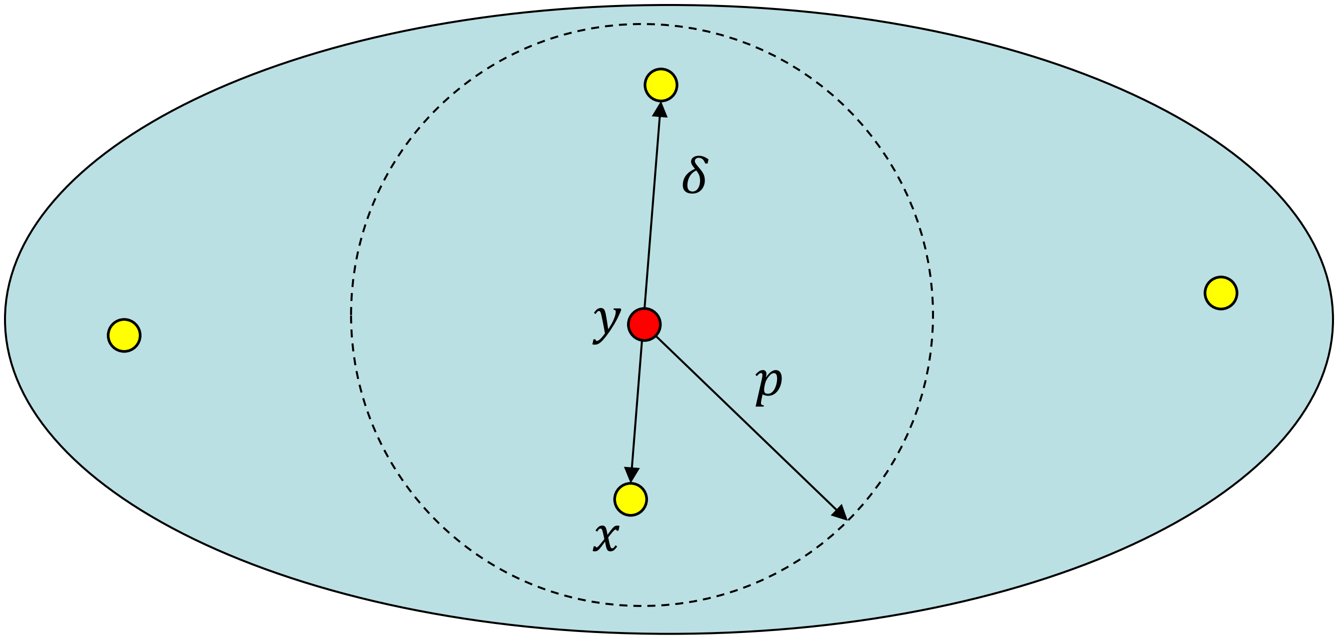
- Adversary corrupts p fraction of coordinates
- Given $y \in \Sigma^k$, decoder finds the unique $x \in C$ such that $\mathrm{dist}(x, y) \le p$
- Must have $p \le \delta/2$



List Decoding

List decoding [Elias, Wozencraft '50s]:

- Adversary corrupts p fraction of coordinates
- Given $y \in \Sigma^k$, decoder finds a short list of $x \in C$ such that $\mathrm{dist}(x, y) \leq p$ for every x in the list
- Can correct more than $\delta/2$ fraction of errors



Advantages of List Decoding

In coding theory:

- Bridge between Hamming Channel & Shannon Channel
- Sometimes can use context / side information even if list size >1

In TCS:

Define the (relative) agreement $\mathrm{agr}(x, y) = 1 - \mathrm{dist}(x, y)$

For unique decoding, $\mathrm{dist}(x, y) \leq \delta/2 \leq 1/2$

- Can recover x from y only if $\mathrm{agr}(x, y) \geq 1/2$

For list decoding, can have $\mathrm{dist}(x, y)$ close to 1

- Can find a short list of candidates x from y even if $\mathrm{agr}(x, y)$ is small

Advantages of List Decoding

TCS apps:

1. Cryptography: Hard-core predicates [Goldreich-Levin'89]

2. Learning:
   - Boolean functions [Goldreich-Levin'89]
   - Decision trees [Kushilevitz-Mansour'91]
   - CNFs / DNFs [Jackson'94]

3. Complexity theory:
   - Average-to-worst-case reductions [Lipton'89, Cai-Pavan-Sivakumar'99, Goldreich-Rubinfeld-Sudan'99]
   - Derandomization / Construction of PRGs [Babai-Fortnow-Nisan-Wigderson'93, Sudan-Trevisan-Vadhan'99]

List Decodable Codes

- Ideally, we want a code of rate R that is
  - list decodable up to radius $\approx 1 - R$
  - with small list size and small alphabet size
  - explicit and efficiently list decodable

List decoding capacity

List Decoding Capacity Theorem:

For $R, \varepsilon \in (0,1)$, there exist *(non-explicit)* codes of rate R that are list decodable up to radius $1 - R - \varepsilon$ with list size $O(1/\varepsilon)$ and alphabet size $2^{O(1/\varepsilon)}$

Are there explicit capacity-achieving list decodable codes with similar parameters?

Reed–Solomon codes

- A Reed–Solomon code over F_q is given by the encoding map $F_q^k \to F_q^n$ defined by

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

where f is a univariate polynomial of degree $< k$ and $\alpha_1, \alpha_2, \cdots, \alpha_n \in F_q$ are n distinct evaluation points

- Guruswami and Sudan proved that RS codes are efficiently list decodable up to radius $1 - \sqrt{R}$ (known as the Johnson bound) [Sudan'97, Guruswami-Sudan'99]
- Most RS codes are list decodable beyond the Johnson bound [Rudra-Wootters'14, GLSTW'20]
- However, it is not known if RS codes can achieve the capacity $1 - R$

Folded Reed–Solomon codes

- Folded Reed–Solomon codes [Guruswami–Rudra'05] are the first explicit codes that achieve the rate $1 - R - \varepsilon$
- It is obtained by combining $m = O(1/\varepsilon^2)$ symbols into one

$$f \mapsto \left(\begin{matrix} f(\alpha_1) \\ f(\gamma\alpha_1) \\ f(\gamma^2\alpha_1) \end{matrix}, \begin{matrix} f(\alpha_2) \\ f(\gamma\alpha_2) \\ f(\gamma^2\alpha_2) \end{matrix}, \cdots, \begin{matrix} f(\alpha_n) \\ f(\gamma\alpha_n) \\ f(\gamma^2\alpha_n) \end{matrix} \right)$$

- Alphabet size $n^{O(1/\varepsilon^2)}$
- List size $(1/\varepsilon)^{O(1/\varepsilon)}$ [Kopparty–Ron-Zewi–Saraf–Wootters'18]

Other Constructions

- Subcodes of AG codes [Guruswami-Xing'13]
  - Alphabet size $2^{\tilde{O}(1/\varepsilon^2)}$
  - List size $2^{\mathrm{poly}(1/\varepsilon)} \cdot 2^{2^{2^{O(\log^* n)}}}$

- Multi-level concatenation of FRS codes + expander-based amplification [Kopparty-Ron-Zewi-Saraf-Wootters'18]
  - Alphabet size $2^{\mathrm{poly}(1/\varepsilon)}$
  - List size $2^{2^{2^{2^{O(1/\varepsilon)}}}}$
  - Encoding time $2^{\mathrm{poly}(1/\varepsilon)} \cdot \mathrm{poly}(n)$

Our Result

There exist codes $C: \Sigma^k \to \Sigma^n$ of rate R that are list decodable up to radius $1 - R - \varepsilon$ with list size $2^{\mathrm{poly}(1/\varepsilon)}$ and alphabet size $2^{\tilde{O}(1/\varepsilon^2)}$. Moreover:

- The encoding time is $\mathrm{poly}(n, 1/\varepsilon)$
- The list is contained in a subspace of dimension $\mathrm{poly}(1/\varepsilon)$, whose basis can be found in time $\mathrm{poly}(n, 1/\varepsilon)$
- Outputting the list takes time $2^{\mathrm{poly}(1/\varepsilon)} \cdot \mathrm{poly}(n)$

➢ Our proof heavily depends on [Guruswami-Xing'13]. One key new idea is the use of BTT subspaces.

BTT matrix/subspace

- A (k, m, r) **block-triangular-Toeplitz (BTT) matrix** over F is a $km \times kr$ full rank matrix over F that is both block-lower-triangular and block-Toeplitz as a $k \times k$ block matrix

$$k = 4: \quad \begin{bmatrix} M_1 & 0 & 0 & 0 \\ M_2 & M_1 & 0 & 0 \\ M_3 & M_2 & M_1 & 0 \\ M_4 & M_3 & M_2 & M_1 \end{bmatrix}$$

- A subspace of F^{km} is a (k, m, r) **BTT subspace** if it is the image of $v \mapsto Mv$

Overview of Our Construction

- Let $\Sigma = F_q^m$ where $q = \mathrm{poly}(1/\varepsilon)$ and $m = O(1/\varepsilon^2)$

1) We first construct a list decodable code $C': \Sigma^k \to \Sigma^n$ such that the list of candidate messages is contained in a small BTT subspace of $\Sigma^k \cong F_q^{km}$
2) Then we construct an explicit subspace $W \subseteq F_q^{km}$ of low codimension that evades any BTT subspace

- The final code is obtained by restricting the message space of C' to W, which reduces the list size to constant

RS code with subfield evaluations

- The code C' is a "AG code with subfield evaluations" [Guruswami-Xing'13]
- We explain the idea using "RS codes with subfield evaluations"
- An RS code over F_q is defined by the encoding map

$$f \mapsto C_f := (f(\alpha_1), f(\alpha_2), \cdots, f(\alpha_n))$$

where $\alpha_1, \alpha_2, \cdots, \alpha_n \in F_q$ are n distinct evaluation points

- An "RS code with subfield evaluations" is simply an RS code over an extension field F_{q^m} with $\alpha_1, \alpha_2, \cdots, \alpha_n \in F_q$

Finding the BTT subspace

- Given y, we want to find a BTT subspace containing the list of all f satisfying $\mathrm{dist}(C_f, y) \leq 1 - R - \varepsilon$
- We can find a low degree multivariate polynomial $Q(Y_1, Y_2, \cdots, Y_s)$ over $F_{q^m}[X]$ such that

$$Q^*(f) := Q\left(f, f^q, \cdots, f^{q^{s-1}}\right) = 0$$

- As $Q^*(f) \in F_{q^m}[X]$, we get a collection of equations by equating the coefficients of $Q^*(f)$ with zero
- This system of linear equations is represented by a BTT matrix M
- So f is contained in $\ker(M)$ whose basis can be found efficiently
- Finally, we show that the kernel of a BTT matrix is a BTT subspace, so $\ker(M)$ is a BTT subspace

Algebraic-Geometric Codes

- For Reed-Solomon codes, we need $q \geq n$ to get n evaluation points
- To make the alphabet size independent of n, we use AG codes (with subfield evaluations)
- AG codes are generalizations of Reed-Solomon codes, where lines are generalized by algebraic curves
- Can have arbitrarily many # evaluation points over F_q for fixed q by using more and more complicated algebraic curves
- We use explicit curves from the Garcia-Stichtenoth tower [Garcia-Stichtenoth'96], following [Guruswami-Xing'13]

Algebraic-Geometric Codes

- Two properties used for RS codes:
  1) Let V be the space of degree-d polynomials. Then any nonzero $f \in V$ has at most d zeros
  2) Dimension of V is $d + 1$
- They are generalized for AG codes
  1) There is an analogous space V for "degree-d polynomials" (called a Riemann-Roch space), and any nonzero $f \in V$ has at most d zeros
  2) Dimension of V is in $[d - g + 1, d + 1]$, where $g \geq 0$ is called the genus
- In the GS tower, there is a good upper bound for g

BTT Evasive Subspace

- A (k, m, r, s) **BTT evasive subspace** is a subspace $W \subseteq F_q^{km}$ such that for any (k, m, r) BTT subspace V,

$$\dim(V \cap W) \leq s$$

Theorem: [GR'20]

> There exists an explicit $(k, m, \varepsilon m, s)$ BTT evasive subspace $W \subseteq F_q^{km}$ of codimension $O(\varepsilon km)$, where $s = \mathrm{poly}(1/\varepsilon)$

- Restricting the message space $\Sigma^k \cong F_q^{km}$ to W reduces the list size to $q^s = 2^{\tilde{O}(1/\varepsilon^2)}$, and yields the desired code

Periodic Subspace

- Periodic subspaces are relaxations of BTT subspaces

$$\begin{bmatrix} M_1 & 0 & 0 & 0 \\ M_2 & M_1 & 0 & 0 \\ M_3 & M_2 & M_1 & 0 \\ M_4 & M_3 & M_2 & M_1 \end{bmatrix} \qquad \begin{bmatrix} M_1 & 0 & 0 & 0 \\ ? & M_1 & 0 & 0 \\ ? & ? & M_1 & 0 \\ ? & ? & ? & M_1 \end{bmatrix}$$

BTT matrix | periodic matrix

- A (k, m, r, s) periodic evasive subspace is also a (k, m, r, s) BTT evasive subspace

Theorem: [Guruswami-Kopparty'13] (based on subspace designs)

For $k \leq q^{O(\varepsilon m/r)}$, there exists an explicit (k, m, r, s) periodic evasive subspace of codimension $O(\varepsilon k m)$, where $s = O(1/\varepsilon^2)$

- However, this yields $(k, m, \varepsilon m, s)$ BTT evasive subspace only for $k = \text{poly}(1/\varepsilon)$ which is too small

Composition

Composition Lemma: [Guruswami–Xing'13]

Let W be (k, m, r, s) periodic evasive	inner subspace
Let W' be (k', km, s, s') periodic evasive.	outer subspace
Then $W \circ W' := W^k \cap W'$ is $(k'k, m, r, s')$ periodic evasive	

- One can use [Guruswami–Kopparty'13] to construct an outer subspace, so that it remains to construct an inner subspace
- This reduces k to $k' = O(\log k)$, but increase s to $\mathrm{poly}(s)$
- [Guruswami–Xing'13] applied composition $O(\log^* n)$ times
  - List size $2^{\mathrm{poly}(1/\varepsilon)} \cdot 2^{2^{2^{O(\log^* n)}}}$

Better Construction

Ideas:

- We observe that if W is BTT evasive, then $W \circ W'$ is also BTT evasive
- Apply composition twice to reduce k to

$$k' = O(\log\log k)$$

- Use brute-force search to find a good non-explicit inner BTT evasive subspace
- Existence of such a BTT evasive subspace follows from the probabilistic method
  - It is crucial to use BTT evasiveness — there are too many period subspaces

Summary

- We first construct an AG code with subfield evaluations
- Then we construct a BTT evasive subspace W and restrict the message space to W to obtain the final code
- W is constructed using repeated composition of periodic evasive subspaces [Guruswami-Kopparty'13] and an inner subspace found by brute-force search
- The "repeated composition" structure also appears elsewhere in coding theory and TCS
  - Construction of asymptotically good codes
  - First proof of the PCP theorem

Open Problems & Directions

- Reduce our list size $2^{\mathrm{poly}(1/\varepsilon)}$ to $O(1/\varepsilon)$ or even subexponential in $1/\varepsilon$
  - For explicit codes, best known bound is $(1/\varepsilon)^{O(1/\varepsilon)}$ for FRS codes
- For an absolute constant q, achieve the list decoding capacity $h_q^{-1}(1-R)$ over a q–ary alphabet
- Are our methods useful for constructing other pseudorandom objects?
  - E.g., lossless dimension expanders [Guruswami-Resch-Xing'18]?