Near-Optimal List-Recovery of Linear Code Families

Ray Li
Department of Mathematics and Computer Science, Santa Clara University, CA, USA

Nikhil Shagrithaya
Department of EECS, University of Michigan, Ann Arbor, MI, USA

Abstract

We prove several results on linear codes achieving list-recovery capacity. We show that random linear codes achieve list-recovery capacity with constant output list size (independent of the alphabet size and length). That is, over alphabets of size at least $\ell^{\Omega(1/\varepsilon)}$, random linear codes of rate R are $(1-R-\varepsilon, \ell, (\ell/\varepsilon)^{O(\ell/\varepsilon)})$-list-recoverable for all $R \in (0,1)$ and ℓ. Together with a result of Levi, Mosheiff, and Shagrithaya, this implies that randomly punctured Reed–Solomon codes also achieve list-recovery capacity. We also prove that our output list size is near-optimal among *all* linear codes: all $(1-R-\varepsilon, \ell, L)$-list-recoverable linear codes must have $L \geq \ell^{\Omega(R/\varepsilon)}$.

Our simple upper bound combines the Zyablov-Pinsker argument with recent bounds from Kopparty, Ron-Zewi, Saraf, Wootters, and Tamo on the maximum intersection of a "list-recovery ball" and a low-dimensional subspace with large distance. Our lower bound is inspired by a recent lower bound of Chen and Zhang.

2012 ACM Subject Classification Mathematics of computing → Coding theory; Theory of computation → Pseudorandomness and derandomization; Mathematics of computing → Combinatorics

Keywords and phrases Error-Correcting Codes, Randomness, List-Recovery, Reed-Solomon Codes, Random Linear Codes

Digital Object Identifier 10.4230/LIPIcs...1

Funding *Ray Li*: Research supported by NSF grant CCF-2347371.
Nikhil Shagrithaya: Research supported in part by NSF grants CCF-2236931 and CCF-2107345.

1 Introduction

In this work, we study list-recovery for random linear codes and random Reed–Solomon codes, proving near-optimal upper and lower bounds.

An *(error correcting) code* $\mathcal{C}$ is a subset of Σ^n for an alphabet Σ, which, in this work, is always $\mathbb{F}_q$ for some prime power q. We study *linear* codes, which are subspaces of $\mathbb{F}_q^n$. We want codes to be large, meaning they have large *rate* $R = \left(\log_q |\mathcal{C}|\right)/n$. We also want codes to tolerate more errors. In the standard unique decoding setting, tolerating many errors means that, for any vector $z \in \mathbb{F}_q^n$, there is at most one *codeword* $c \in \mathcal{C}$ that agrees with z on many coordinates.

We study a generalization of the unique-decoding problem known as list-recovery. In list-recovery, we want that, for any $\ell \times \ell \times \cdots \times \ell$ combinatorial rectangle, there are few codewords c that "agree" with this rectangle on many coordinates. Formally, a code $\mathcal{C}$ is (ρ, ℓ, L)*-list-recoverable* if for any sets $S_1, \ldots, S_n \subset \mathbb{F}_q$ of size $|S_i| = \ell$, there are at most L codewords $c_1, \ldots, c_L \in \mathcal{C}$ such that $c_i \in S_i$ for at least a $(1-\rho)n$ fraction of the coordinates. The special case of $(\rho, 1, 1)$ list-recoverability is the standard unique-decoding setting, and the special case of $(\rho, 1, L)$ list-recoverability is known as *list-decodability*.

List-recovery has motivations in coding theory, complexity theory, and algorithms. In coding theory, list-recovery has been used as a tool to obtain efficient list-decoding algorithms [25, 22, 28, 35, 31]. Also, list-recoverable random-linear codes — which we study in this work

— are used as a building block in other coding constructions [18, 32]. In complexity theory, list-recoverable codes find applications in constructions of other pseudorandom objects such as extractors [49] and condensers [27]. In algorithms, they are also useful primitives in group testing [33, 41] sparse recovery [12], and streaming algorithms [37, 8].

The list-recovery capacity theorem states that $\rho = 1 - R$ is the optimal tradeoff between the error radius ρ and the code rate R. (see e.g., [23, 43]). That is, below capacity $\rho < 1 - R$, there exist $(p, \ell, O_\ell(1))$-list-recoverable codes of rate R, and above capacity $\rho > 1 - R$, any (ρ, ℓ, L)-list-recoverable code must have exponential list size $L \geq q^{\Omega(n)}$. The existence holds because uniformly random codes of rate R (over sufficiently large alphabets $q \geq \ell^{\Omega(1/\varepsilon)}$) are $(1 - R - \varepsilon, \ell, O(\ell/\varepsilon))$-list-recoverable with high probability.

We wish to understand what kinds of codes achieve list-recovery capacity. A number of explicit code constructions are known to achieve list-recovery capacity, including Folded Reed–Solomon codes, Multiplicity codes, Folded Algebraic–Geometry codes. Additional techniques — subspace evasive sets, subspace designs, and expander techniques — can be used to improve the output list-size L and alphabet size q [22, 35, 28, 29, 30, 32, 31, 19, 9, 36, 50] (see Table 1 in [36], see also [48, 5] for even tighter list size bounds in the special case of list-decoding).

Still, several fundamental questions remain open.

1. First, how list-recoverable is a random linear code? A random linear code is a random subspace of $\mathbb{F}_q^n$. All explicit constructions are based on linear codes (though many are only linear over a subfield), so it is natural to wonder about list-recovery of a "typical" linear code. As list-recovery is a pseudorandom property, this question also addresses the deeper geometric question of "how similar is a random subspace to a random set over $\mathbb{F}_q^n$?", which is well-studied in the more specific context of list-decoding [52, 10, 17, 7, 51, 44, 45, 46, 39, 20, 1].
2. Second, how list-recoverable are Reed–Solomon codes? The above constructions all generalize the Reed–Solomon code, the most fundamental polynomial evaluation code. Can Reed–Solomon codes themselves achieve list-recovery capacity? Given recent progress that showed the special case that Reed–Solomon codes achieve list-*decoding* capacity [4, 15, 1], this general case of list-recovery has been an obvious and tantalizing open question.
3. Lastly, is there a fundamental separation between linear and nonlinear codes for list-recovery? On one hand, there is no apparent separation for the special case of list-decoding, where random linear codes are list-decodable to capacity with list-size $O(1/\varepsilon)$ [17, 51, 39, 20], just like uniformly random codes. On the other hand, uniformly random codes are list-recoverable with list size $O(\ell/\varepsilon)$, but all known linear constructions require output list size at least $\ell^{\Omega(1/\varepsilon)}$, and this lower bound has been proven in various specific settings [20, 38, 5].

We answer all three questions. We show that random linear codes are list-recoverable to capacity with provably near-optimal output list size. By a recent result of [38], this implies that randomly punctured Reed–Solomon codes are list-recoverable to capacity with near-optimal output list size. Lastly, we prove a fundamental separation between linear and non-linear codes by showing that *all* linear codes of rate R must have list-size at least $L \geq \ell^{\Omega(R/\varepsilon)}$.

1.1 Our results

We now state our results in the context of prior work.

Citation	Radius ρ	input list size	output list size
[52, 16]	$1-R-\varepsilon$	ℓ	$q^{O(\ell/\varepsilon)}$
[46]	$1-R-\varepsilon$	ℓ	$q^{O(\log^2(\ell/\varepsilon))}$
This work	$1-R-\varepsilon$	ℓ	$\left(\frac{\ell}{\varepsilon}\right)^{O(\ell/\varepsilon)}$

Table 1 List-recovery of Random Linear codes

1.1.0.1 List recovery for Random Linear Codes.

Several known arguments show that random linear codes achieve list-recovery capacity. A random linear code is a code generated by a uniformly random generator matrix $\mathbf{G} \in \mathbb{F}_q^{n\times k}$. First, the Zyablov-Pinsker argument [52] adapted to list-recovery shows that random linear codes of rate R over alphabet $q \geq \ell^{\Omega(1/\varepsilon)}$ are $(1-R-\varepsilon, \ell, q^{O(\ell/\varepsilon)})$-list-recoverable (see, for example [16, Lemma 9.6]). Rudra and Wootters [46] improved the output list size to $q^{O(\log^2(\ell/\varepsilon))}$, showing random linear codes of rate R over alphabet $q \geq \ell^{\Omega(1/\varepsilon)}$ are $(1-R-\varepsilon, \ell, q^{O(\log^2(\ell/\varepsilon))})$-list-recoverable. We improve the output list size to be independent of the alphabet size q.

▶ **Theorem 1** (Theorem 8, Informal). *For all $R, \varepsilon \in (0,1)$, and $q \geq \ell^{\Omega(1/\varepsilon)}$ a random linear code of rate R is $\left(1-R-\varepsilon, \ell, \left(\frac{\ell}{\varepsilon}\right)^{O(\ell/\varepsilon)}\right)$-list-recoverable with high probability.*

Our list size improves on the prior bounds when $q \geq \ell^{\Omega(\ell/\varepsilon)}$, which covers most alphabet sizes ($q \geq \ell^{\Omega(1/\varepsilon)}$ is needed to achieve list-recovery capacity), and the improvement is more significant when q is larger. This improvement to an alphabet-independent output list size is critical for Theorem 2 below (see Remark 3). As we show in Theorem 5, this output list size is near optimal among all linear codes.

Our proof is simple, combining the Zyablov–Pinsker [52] argument with recent analyses of the list-recovery of explicit constructions like Folded Reed–Solomon codes. In particular, the Zyablov–Pinsker argument [52] shows that a random linear code can be list-recovered so that, with high probability the output list always lies in a subspace of dimension at most $O(\ell/\varepsilon)$. Naively, this implies an output list size bound of $q^{O(\ell/\varepsilon)}$. However, recent analyses of list-recovering explicit codes [36, 50] showed that subspaces of dimension D with good distance — random linear codes are well known to have good distance with high probability — can have at most $(\ell/\varepsilon)^{O(D)}$ points inside an ℓ-list-recovery ball, thus giving our improved output list size. We also show that we get the best possible output list size for our proof technique, in the sense that, for any linear code, there are output lists that span a subspace of dimension at least $\Omega(\ell/\varepsilon)$ (see Proposition 13).

We believe that the simplicity of our proof is a strong indicator that we have found the right way to approach the problem, which had previously resisted various other proof techniques.

1.1.0.2 List recovery for Random Reed-Solomon Codes.

Reed–Solomon codes [42] are the most fundamental evaluation codes. A Reed–Solomon code is given by n evaluation points $\alpha_1, \alpha_2, \ldots, \alpha_n$ in a finite field $\mathbb{F}_q$, and a degree $k < n$, and is defined as

$$\mathrm{RS}_{n,k}(\alpha_1, \ldots, \alpha_n) := \{(f(\alpha_1), \ldots, f(\alpha_n)) \mid f \in \mathbb{F}_q[x],\ \deg f < k\}\,.$$

List-decoding and list-recovery of Reed–Solomon codes are well-studied questions. The seminal Guruswami–Sudan [25] algorithm showed that Reed–Solomon codes are list-decodable and list-recoverable up to the *Johnson radius* $1-\sqrt{R\ell}$ [34, 26]. Since then, there has been much interest in determining whether Reed–Solomon codes are list-decodable and list-recoverable beyond the Johnson bound, and perhaps even up to capacity $\rho = 1-R$ (the capacity is $1-R$ for both list-decoding and list-recovery). Initially, there was evidence against this possibility [21, 6, 2], suggesting that Reed–Solomon codes could not be list-decoded or list-recovered much beyond the Johnson bound. Since then, an exciting line of work has shown, to contrary, that Reed–Solomon codes can beat the Johnson bound for list-decoding [44, 47, 11, 13, 14, 4, 15, 1], and, in fact, can be list-decoded up to capacity [4, 15, 1]. All of these works studied *randomly punctured* Reed–Solomon codes, where $\alpha_1, \ldots, \alpha_n$ are chosen at random from a larger field q.

Despite the exciting progress for list-decoding, there has been comparatively little progress on list-recovery. Lund and Potukuchi [40] and Guo, Li, Shangguan, Tamo, and Wootters [14] proved that (randomly punctured) Reed–Solomon codes are list-recoverable beyond the Johnson bound in the low-rate regime: [40] shows (ρ, ℓ, L)-list-recovery for $\rho \le 1-1/\sqrt{2}$, $L = O(\ell)$ and rate $\Omega\left(\frac{1}{\sqrt{\ell}\log q}\right)$, and [14] shows $\left(\Omega\left(\frac{\varepsilon}{\sqrt{\ell}\log(1/\varepsilon)}\right), \ell, O(\ell/\varepsilon)\right)$-list-recovery for rate $1-\varepsilon$ Reed–Solomon codes. Both improve on the Johnson radius of $O\left(\frac{1}{\ell}\right)$ in the low rate setting.

In [38], Levi, Mosheiff and Shagrithaya showed that random Reed–Solomon codes and random linear codes are *locally equivalent*, meaning that both random code families achieve identical rate thresholds for all "local properties", which include (the complements of) list-decoding and list-recovery. Thus, our result for list-recovery of random linear codes transfers to random Reed–Solomon codes as well.

▶ **Theorem 2** (Theorem 11, Informal). *For all $R, \varepsilon \in (0,1)$, a randomly punctured Reed–Solomon code of length n over alphabet size $q = n \cdot (\ell/\varepsilon)^{(\ell/\varepsilon)^{O(\ell/\varepsilon)}}$, of rate R is $\left(1-R-\varepsilon, \ell, \left(\frac{\ell}{\varepsilon}\right)^{O(\ell/\varepsilon)}\right)$-list-recoverable with high probability.*

We note that the problem of determining optimal list sizes for random Reed–Solomon codes across all rates $R \in (0,1)$ has proven resistant to a variety of previous approaches. The simplicity of our proof suggests that the method presented here may offer a promising direction for completely resolving this question.

▶ Remark 3. We note that in order to use the equivalence result from [38], it is crucial that the upper bound on the list size be independent of the alphabet size, as guaranteed by Theorem 1. Hence, previous results on list size cannot be used with the equivalence result.

▶ Remark 4. A fruitful line of work [22, 35, 28, 9, 36, 50] has culminated in output list sizes of $O\left(\frac{\ell}{\varepsilon}\right)^{O(\log(\ell)/\varepsilon)}$ for various explicit list-recoverable codes such as Folded Reed–Solomon codes and Multiplicity codes. This list size is better than our list size of $(\frac{\ell}{\varepsilon})^{O(\ell/\varepsilon)}$ by roughly a factor of $\ell/\log(\ell)$ in the exponent. However, our results are still interesting because, as described above, the list-recovery of random linear codes and Reed–Solomon codes are fundamental questions, and also because our results yield linear codes for list-recovery and use much smaller alphabet sizes.

1.1.0.3 Lower bounds for list-recovery.

We now discuss impossibility results for list-recovery. An early impossibility result of Guruswami and Rudra in [21] showed that, in the setting of zero-error list-recovery ($\rho = 0$),

many full length Reed–Solomon codes of rate R require $R \leq 1/\ell$ in order to have poly n output list size, so many full length Reed–Solomon codes cannot be list-recovered beyond the Johnson bound — note this does not contradict Theorem 2, as we consider randomly punctured, as opposed to full length ($n = q$) codes. More recently it was shown that achieving list-recovery capacity requires exponential list size $\ell^{\Omega(1/\varepsilon)}$ for particular codes: random linear codes in the high-rate zero-error ($\rho = 0$) regime [20], random linear codes in general parameter settings [38], and for Reed–Solomon codes, Folded Reed–Solomon codes, and Multiplicity codes in general parameter settings [5].

Inspired by the lower bound in [5], we show that *any* linear code list-recoverable to capacity must have output list size at least $\ell^{\Omega(R/\varepsilon)}$.

▶ **Theorem 5** (Theorem 12, Informal)**.** *Over any field, any linear code of rate R that is $(1 - R - \varepsilon, \ell, L)$ list-recoverable must satisfy $L \geq \ell^{\Omega(R/\varepsilon)}$.*

One takeaway from Theorem 5 is that our list sizes of $(\ell/\varepsilon)^{O(\ell/\varepsilon)}$ in Theorem 1 and Theorem 2 are near-optimal. Additionally, Doron and Wootters [8] asked whether there were explicit list-recoverable codes with, among other desired guarantees, output list size $L = O(\ell)$. Our result shows this is not possible with *any* linear code. Lastly, our lower bound shows separation between non-linear and linear codes for list-recovery, which is perhaps surprising given that no such separation exists for list-decoding.

We point out that, for list-decoding ($\ell = 1$), our lower bound is trivial ($L \geq 1$), so it does not contradict the recent results that random linear codes, randomly punctured Reed–Solomon codes, and randomly punctured Algebraic-Geometry codes achieve list-decoding capacity with output list size $O(1/\varepsilon)$ [4, 15, 1, 3].

2 Preliminaries

For a prime power q, let $\mathbb{F}_q$ be the finite field of order q. Let $[n]$ denote the set $\{1, \ldots, n\}$. For a given vector space V, let $\mathcal{L}(V)$ denote the set of all subspaces of V. For a given set S, let 2^S denote the power set of S. For a vector v, let $v[i]$ denote its ith entry.

A code $\mathcal{C} \subseteq \mathbb{F}_q^n$ is said to be linear if it is a linear subspace, and said to have rate $R \in (0, 1)$ if $R = dim(\mathcal{C})/n$. We say $\mathcal{C}$ has relative distance $\delta \in (0, 1)$ if $\forall c \in \mathcal{C}, wt(c) \geq \delta \cdot n$, where $wt(c)$ denotes the number of non-zero entries in the codeword c. A matrix $\mathbf{G} \in \mathbb{F}_q^{n \times Rn}$ containing linearly independent columns is said to be the *generator matrix* of $\mathcal{C}$ if every codeword $c \in \mathcal{C}$ can be constructed using some linear combinations of the columns in $\mathbf{G}$. $\mathcal{C}$ is said to *contain* a set of vectors $s_1, \ldots s_b \in \mathbb{F}_q^n$ if $s_i \in \mathcal{C}$ for every $i \in [b]$.

For a vector $x \in \mathbb{F}_q^n$ and sets $S_1, \ldots, S_n \subseteq \mathbb{F}_q$, the *agreement set* $\text{agr}(x, S_1, \ldots, S_n)$ is defined as:

$$\text{agr}(x, S_1, \ldots, S_n) \coloneqq \{i \in [n] \mid x[i] \in S_i\}.$$

A *ρ-radius ℓ-list-recovery ball* $B(\rho, S_1 \times \cdots \times S_n)$ is given by input lists $S_1, \ldots, S_n \subseteq \mathbb{F}_q$ of size ℓ, and is defined to be

$$B(\rho, S_1 \times \cdots \times S_n) = \left\{x \in \mathbb{F}_q^n : \text{agr}(x, S_1 \times \cdots \times S_n) \geq (1 - \rho)n\right\}. \tag{1}$$

We can alternatively define (ρ, ℓ, L)-list-recovery using the above definition: a code $\mathcal{C} \subseteq \mathbb{F}_q^n$ is (ρ, ℓ, L)-list-recoverable if every ρ-radius ℓ-list-recovery ball B contains at most L codewords.

For $0 < R < 1$, a *random linear code* (RLC) of rate R is a linear code whose generator matrix $\mathbf{G} \in \mathbb{F}_q^{n \times Rn}$ is a matrix whose entries are chosen uniformly at random from $\mathbb{F}_q$, independently of one another. For $\alpha_1, \ldots, \alpha_n \in \mathbb{F}_q$, we use $\mathsf{RS}(\alpha_1, \ldots, \alpha_n; Rn)$ to denote

the Reed–Solomon (RS) code of rate R obtained by evaluating polynomials of degree $< Rn$ on evaluation points $\alpha_1, \ldots, \alpha_n \in \mathbb{F}_q$. We say this is a *random RS code* if the evaluation points have been chosen uniformly at random and independently of one another[1].

2.1 Local Coordinate-Wise Linear (LCL) Properties

We now introduce the machinery in [38] that connects random linear codes to (randomly punctured) Reed–Solomon codes. A *code property* $\mathcal{P}_n$ for codes of block length n in $\mathbb{F}_q^n$ is simply a family of codes in $\mathbb{F}_q^n$. We say that a code $\mathcal{C}_n \subseteq \mathbb{F}_q^n$ *satisfies* $\mathcal{P}_n$ if $\mathcal{C}_n \in \mathcal{P}_n$. Denoting $\mathcal{P} \coloneqq \{\mathcal{P}_n\}_{n\in\mathbb{N}}$, we say that an infinite family of codes $\mathcal{C}_n \coloneqq \{\mathcal{C}_n\}_{n\in\mathbb{N}}$ *satisfies* $\mathcal{P}$ if $\mathcal{C}_n \in \mathcal{P}_n$ for every $n \in \mathbb{N}$. In this paper, we focus on local, monotone-increasing code properties. A local code property, informally speaking, is defined by the inclusion of some bad set. A *monotone-increasing* code property is one for which the following is true: if $\mathcal{C}$ satisfies $\mathcal{P}$, then every $\mathcal{C}'$ for which $\mathcal{C}' \supseteq \mathcal{C}$ holds, also satisfies $\mathcal{P}$. An example of local, monotone-increasing code property is the complement of (ρ, L)-list-decodability, defined as the family of all codes that contain at least one set of $L+1$ distinct vectors, all lying within a Hamming ball of radius ρ.

For a locality parameter $b \in \mathbb{N}$, an ordered tuple of subspaces $\mathcal{V} = (\mathcal{V}_1, \ldots, \mathcal{V}_n)$, where $\mathcal{V}_i \in \mathcal{L}(\mathbb{F}_q^b)$ for each $i \in [n]$ is defined to be a *b-local profile*. Note that $\mathcal{V} \in \mathcal{L}(\mathbb{F}_q^b)^n$. We say that a matrix $A \in \mathbb{F}_q^{n\times b}$ *is contained in* $\mathcal{V}$ if the ith row of A belongs to $\mathcal{V}_i$, for all i. A code $\mathcal{C} \subseteq \mathbb{F}_q^n$ is said to *contain* $\mathcal{V}$ if

(a) there exists a matrix $A \in \mathbb{F}_q^{n\times b}$ *with distinct columns* such that the set of columns of A is contained in $\mathcal{C}$, and

(b) A is contained in $\mathcal{V}$.

For a family of b-local profiles $\{\mathcal{F}_n\}_{n\in\mathbb{N}}$, where $\mathcal{F}_n \subseteq \mathcal{L}\left(\mathbb{F}_q^b\right)^n$, we define a *$b$-local coordinate wise linear* (b-LCL) property $\mathcal{P} \coloneqq \{\mathcal{P}_n\}_{n\in\mathbb{N}}$ as follows:

$$\mathcal{P}_n = \left\{\mathcal{C} \subseteq \mathbb{F}_q^n \mid \exists \mathcal{V} \in \mathcal{F}_n \text{ such that } \mathcal{C} \text{ contains } \mathcal{V}\right\}.$$

The complement of (ρ, ℓ, L)-list-recoverability is a $(L+1)$-LCL property. This is proven in [38, Proposition 2.2], but we provide a justification in this paragraph. Every bad set of vectors lying within a given ρ-radius ℓ-list-recovery ball agrees with some input lists $S_1, \ldots, S_n \subseteq \mathbb{F}_q$ at a lot of coordinates. This implies that the vectors agree with one another at a lot of coordinates as well, and once we arrange the bad vector sets as columns in a matrix of dimension $n \times (L+1)$, we can specify these agreements as linear constraints on the rows of such matrices. Formally, the property is defined by a family of $(L+1)$-local profiles that we now describe. For every $n \in \mathbb{N}$, we define $\mathcal{F}_n$ by describing the $(L+1)$-local profiles $\mathcal{V} \in \mathcal{L}(\mathbb{F}_q^{L+1})^n$ that constitute it. Let $S_{(1-\rho)} \subseteq \left(2^{[n]}\right)^{L+1}$ denote the collection of all $L+1$-length tuples where each element is a subset of $[n]$ of size exactly $(1-\rho)n$. Furthermore, let $M_{[\ell]} \coloneqq [\ell]^{n\times(L+1)}$ denote the set of all matrices of dimension $n \times (L+1)$ having elements in $[\ell]$ [2]. Then, for every $s = (s_1, \ldots, s_{(L+1)}) \in S_{(1-\rho)}$, every $M \in M_{[\ell]}$, define $\mathcal{V}(s, M) = (\mathcal{V}_1, \ldots, \mathcal{V}_n)$ such that for every $i \in [n]$,

$$\mathcal{V}_i \coloneqq \left\{r \in \mathbb{F}_q^{L+1} \mid \forall j,k \in [L+1], r[j] = r[k] \text{ if } (i \in s_j) \wedge (i \in s_k) \wedge (M[i,j] = M[i,k])\right\}.$$

[1] This is different from the usual model for random RS codes, where it is required that the random evaluation points be distinct. However, it can be shown that both models behave similarly (refer to [38], Appendix A for details).

[2] Even though $S_{(1-\rho)}$ and $M_{[\ell]}$ depend on n, we have suppressed this dependence in the notation for sake of clarity.

Note that each $\mathcal{V}_i$ is a subspace of $\mathbb{F}_q^{(L+1)}$, and therefore $\mathcal{V}(s, M)$ is a valid linear profile. We can now define the associated family of linear profiles for the complement of (ρ, ℓ, L)-list-recoverability:

$$\mathcal{F}_n := \left\{ \mathcal{V} \in \mathcal{L}\left(\mathbb{F}_q^{L+1}\right)^n \mid \exists s \in S_{(1-\rho)}, M \in M_{[\ell]}, \text{ such that } \mathcal{V} = \mathcal{V}(s, M) \right\}.$$

Observe that

$$|\mathcal{F}_{\mathcal{P}}| \leq \left|S_{(1-\rho)}\right| \cdot \left|M_{[\ell]}\right| \leq \binom{n}{\rho n}^{(L+1)} \cdot \ell^{(L+1)n}. \tag{2}$$

In the same work, the authors also prove a threshold theorem for random linear codes (RLCs) in relation to all LCL properties, and moreover, gave a complete characterization of the rate threshold. Informally, the theorem says that RLCs of a sufficiently large alphabet exhibit a sharp threshold phenomenon for all LCL properties. That is, for every LCL property $\mathcal{P}$, there exists a rate threshold $R_{\mathcal{P}}$ such that RLCs of rate $R_{\mathcal{P}} - \varepsilon$ satisfy $\mathcal{P}$ with high probability, and RLCs of rate $R_{\mathcal{P}} + \varepsilon$ **do not** satisfy $\mathcal{P}$ with high probability.

▶ **Theorem 6** ([38], Theorem 3.1)**.** *Let $\mathcal{P}$ be a b-LCL property of codes in $\mathbb{F}_q^n$ and let $\mathcal{F} \subseteq \mathcal{L}\left(\mathbb{F}_q^b\right)^n$ be a corresponding family of profiles. Let $\mathcal{C} \subseteq \mathbb{F}_q^n$ be an RLC of rate R. Then, there is some threshold rate $R_{\mathcal{P}}$ for which the following holds.*

1. *If $R \geq R_{\mathcal{P}} + \varepsilon$ then $\mathbf{Pr}[\mathcal{C}$ satisfies $\mathcal{P}] \geq 1 - q^{-\varepsilon n + b^2}$.*
2. *If $R \leq R_{\mathcal{P}} - \varepsilon$ then $\mathbf{Pr}[\mathcal{C}$ satisfies $\mathcal{P}] \leq |\mathcal{F}| \cdot q^{-\varepsilon n + b^2}$.*
3. *In particular, if $R \leq R_{\mathcal{P}} - \varepsilon$ and $q \geq 2^{\frac{2\log_2 |\mathcal{F}|}{\varepsilon n}}$ then $\mathbf{Pr}[\mathcal{C}$ satisfies $\mathcal{P}] \leq q^{-\frac{\varepsilon n}{2} + b^2}$.*

The concept of LCL properties allows for "transfer type" theorems between random linear codes and random RS codes. In more detail, for every reasonable LCL property (that is, for every LCL property whose corresponding family of profiles is large), the rate thresholds for random linear codes and random RS codes are the same. That is, any rate threshold proved for LCL properties of RLCs also applies for random RS codes, and vice versa. For our purposes, we only require one part of this result, which we formally state below.

▶ **Theorem 7** ([38], Theorem 3.10 (part 1) (Threshold theorem for RS codes))**.** *Let $\mathcal{P}$ be a b-LCL property of codes in $\mathbb{F}_q^n$, with associated local profile family $\mathcal{F}_{\mathcal{P}} \subseteq \mathcal{L}\left(\mathbb{F}_q^b\right)^n$ and (random linear code) threshold rate $R_{\mathcal{P}}$. Let $0 < R' < 1$ and let $\mathcal{C} = \mathsf{RS}_{\mathbb{F}_q}(\alpha_1, \ldots, \alpha_n; R'n)$, and $\alpha_1, \ldots, \alpha_n$ are sampled independently and uniformly from $\mathbb{F}_q$. Assume that $q > R'nb$. Fix $\varepsilon' n \geq 2b(b+1)$. If $R' \leq R_{\mathcal{P}} - \varepsilon'$, then*

$$\mathbf{Pr}[\mathcal{C} \text{ satisfies } \mathcal{P}] \leq (2^b - 1) \cdot \left(\frac{(4b)^{4b} R' n}{\varepsilon' q}\right)^{\frac{\varepsilon' n}{2b}} \cdot |\mathcal{F}_{\mathcal{P}}|. \tag{3}$$

3 List-recovery of Random Linear Codes

In this section, we prove Theorem 1, that random linear codes achieve list-recovery capacity with constant output list size. Formally, we show the following.

▶ **Theorem 8.** *Fix $0 < R < 1$, $\varepsilon > 0$ so that $(1 - R - \varepsilon) > 0$, $\ell \in \mathbb{N}$, and let q be a prime power such that $q \geq \max\left(\ell^{\frac{8R}{\varepsilon}+6}, \ell \cdot 2^{4/\varepsilon}\right)$. Let $\mathcal{C} \subseteq \mathbb{F}_q^n$ be an RLC of rate R. Then with probability at least $1 - 2q^{-\frac{\varepsilon n}{8}}$, $\mathcal{C}$ is $((1 - R - \varepsilon), \ell, L)$-list-recoverable with L satisfying $L \leq \left(\frac{2\ell}{\varepsilon}\right)^{2\ell/\varepsilon}$.*

The theorem follows as a consequence of two lemmas. We first state both lemmas, and then give the proof of Theorem 8 using them. The first lemma essentially states that any low dimensional subspace with good distance has few points in a list-recovery ball. This lemma appears in [36, 50]; we state the version from [50, Lemma 3.1].

▶ **Lemma 9** ([50], see also [36]). *For $\varepsilon > 0$ and $\ell \in \mathbb{N}$, let $\mathcal{C} \subseteq \mathbb{F}_q^n$ be a linear code with relative distance $\delta > \varepsilon/2$ that is $(\delta - \frac{\varepsilon}{2}, \ell, L)$-list-recoverable. Assume further that any output list is contained in a subspace $V \subseteq \mathcal{C}$ of dimension r. Then the output list size $L \leq \left(\frac{2\ell}{\varepsilon}\right)^r$.*

The second lemma uses the Zyablov–Pinsker argument [52], showing that a random linear code does not have too many linearly independent codewords within a list-recovery ball.

▶ **Lemma 10.** *Fix $0 < R < 1$, $\varepsilon > 0$ so that $(1 - R - \varepsilon) > 0$, $\ell \in \mathbb{N}$, and let q be a prime power such that $q \geq \max\left(\ell^{\frac{8R}{\varepsilon}+6}, \ell \cdot 2^{4/\varepsilon}\right)$. Let $\mathcal{C} \subseteq \mathbb{F}_q^n$ be an RLC of rate R. Then with probability at least $1 - q^{-\frac{\varepsilon n L}{8}}$, for every input lists $\mathcal{S}_1, \ldots, \mathcal{S}_n$ of size ℓ, the maximal linearly independent subset of $\mathcal{C}$ within the $(1 - R - \varepsilon)$ radius ℓ-list-recovery ball $B\left((1 - R - \varepsilon), S_1 \times \cdots \times S_n\right)$ has size less than $2\ell/\varepsilon$.*

Proof of Lemma 10. Denote $\rho := 1 - R - \varepsilon$ and $L := 2\ell/\varepsilon$. We also assume q is a multiple of ℓ for simplicity of exposition, and note that the result holds in the general case as well. We show that an RLC "avoids" all bad configurations with high probability. A *bad configuration* is a set V of linearly independent vectors of size L such that there exist input lists $\mathcal{S}_1, \ldots, \mathcal{S}_n$ of size ℓ, so that $V \subseteq B\left(\rho, S_1 \times \cdots \times S_n\right)$. We say that $\mathcal{C}$ contains a bad configuration V if for every $v \in V$, v is also in $\mathcal{C}$. If this condition is not satisfied, then we say that $\mathcal{C}$ does not contain V. It is easy to see that if $\mathcal{C}$ contains no bad configurations, then the maximal linearly independent subset of $\mathcal{C}$ within any ℓ-list-recovery ball of radius ρ has size less than $2\ell/\varepsilon$. Therefore we show that the probability of $\mathcal{C}$ containing a bad configuration is low.

Fix input lists $\mathcal{S}_1, \ldots, \mathcal{S}_n$ of size ℓ, and let $B := B\left(\rho, S_1 \times \cdots \times S_n\right)$ be the corresponding ρ radius ℓ-list-recovery ball. The size of B is $\binom{n}{\rho n} \cdot \ell^{(1-\rho)n} \cdot q^{\rho n}$. The probability that a particular configuration is bad is equal to the probability of the encodings of some L linearly independent messages being inside B simultaneously, which is $\left(\frac{|B|}{q^n}\right)^L$. By a union bound over at most $q^{n\ell}$ possible input lists and all L-sized linearly independent subsets of the message vectors in $\mathbb{F}_q^{Rn}$ (there are at most q^{RnL} such subsets), we have

$$
\begin{aligned}
\mathop{\mathbf{Pr}}_{\mathcal{C}}[\mathcal{C} \text{ contains a bad configuration}] &\leq \left(\frac{\binom{n}{\rho n} \cdot \ell^{(1-\rho)n} \cdot q^{\rho n}}{q^n}\right)^L \cdot q^{n\ell} \cdot q^{RnL} \\
&= \left(\left(\frac{\ell}{q}\right)^n \cdot \binom{n}{\rho n} \cdot \left(\frac{q}{\ell}\right)^{\rho n}\right)^L \cdot q^{n\ell} \cdot q^{RnL} \\
&\leq \left(\left(\frac{\ell}{q}\right)^n \cdot (q/\ell)^{H_{q/\ell}(\rho)n}\right)^L \cdot q^{n\ell} \cdot q^{RnL} \\
&\leq \left((q/\ell)^{-(1-H_{q/\ell}(\rho))n}\right)^L \cdot q^{n\ell} \cdot q^{RnL} \\
&\leq \left((q/\ell)^{-\left(R+\frac{3\varepsilon}{4}\right)n}\right)^L \cdot q^{n\ell} \cdot q^{RnL} \qquad (4) \\
&= \ell^{(R+\frac{3\varepsilon}{4})nL} \cdot q^{-\left(R+\frac{3\varepsilon}{4}\right)nL} \cdot q^{\frac{\varepsilon n L}{2}} \cdot q^{RnL} \\
&= \ell^{(R+\frac{3\varepsilon}{4})nL} \cdot q^{-\frac{\varepsilon n L}{4}} \\
&\leq q^{-\frac{\varepsilon n L}{8}}.
\end{aligned}
$$

In Equation 4, we used $q \geq \ell \cdot 2^{4/\varepsilon}$, and for the last inequality, we used $q \geq \ell^{\frac{8R}{\varepsilon}+6}$. This implies that the probability with which $\mathcal{C}$ does not contain any bad configuration is at least $1 - q^{-\frac{\varepsilon n L}{8}}$. ◀

We now prove Theorem 8.

Proof of Theorem 8. Denote $\rho := 1 - R - \varepsilon$. Denote by E_1 the event that for a RLC $\mathcal{C}$ of rate R, the maximal linearly independent subset of $\mathcal{C}$ within every $(1 - R - \varepsilon)$ radius ℓ-list-recovery ball has size less than $2\ell/\varepsilon$. By Lemma 10, we know that E_1 happens with probability at least $1 - q^{-\frac{\varepsilon n L}{8}}$. Let E_2 denote the event that a rate R RLC $\mathcal{C}$ has distance at least $1 - R - \frac{\varepsilon}{2}$. By the Gilbert-Varshamov bound (see [24], Section 4.2), and because of the fact that $q \geq \ell \cdot 2^{4/\varepsilon} \geq 2^{4/\varepsilon}$, E_2 happens with probability at least $1 - q^{-\frac{\varepsilon n}{2}}$. Therefore we have

$$\mathop{\mathbf{Pr}}_{\mathcal{C}}[E_1 \wedge E_2] \geq 1 - q^{-\frac{\varepsilon n L}{8}} - q^{-\frac{\varepsilon n}{2}} \geq 1 - 2 \cdot q^{-\frac{\varepsilon n}{8}}$$

When E_1 and E_2 occur simultaneously, the assumptions of Lemma 9 are satisfied by $\mathcal{C}$ with $r = 2\ell/\varepsilon$ and $\delta = 1 - R - \frac{\varepsilon}{2}$, and therefore we see that

$$\mathop{\mathbf{Pr}}_{\mathcal{C}}\left[\bigwedge_{B} |\mathcal{C} \cap B| \leq \left(\frac{2\ell}{\varepsilon}\right)^{2\ell/\varepsilon}\right] \geq 1 - 2 \cdot q^{-\frac{\varepsilon n}{8}}$$

where B is ranging over all $(1 - R - \varepsilon)$ radius ℓ-list-recovery balls, and we are done. ◀

4 List-Recovery of Reed–Solomon codes

In this section, we will prove the following result, which says that random Reed–Solomon codes are list-recoverable to capacity with constant output list size. The proof combines Theorem 8 from the previous section with Theorem 6, the equivalence theorem from [38].

▶ **Corollary 11.** *Fix $0 < R < 1$, $\varepsilon > 0$ so that $(1 - R - \varepsilon) > 0$, $\ell \in \mathbb{N}$. Fix a constant $\varepsilon' > 0$ such that $\varepsilon' < R$, and denote $L := \lfloor \left(\frac{2\ell}{\varepsilon}\right)^{2\ell/\varepsilon} \rfloor$. Let $\eta > 0$ be a constant, and let q be a prime power satisfying $q > \frac{(4(L+1))^{4(L+1)} Rn}{\varepsilon'} \cdot 2^{\frac{((\log \ell + 2)(L+1) + \eta) \cdot 2(L+1)}{\varepsilon'}}$. Then, a random RS code of rate $R - \varepsilon'$ over $\mathbb{F}_q^n$ is $(1 - R - \varepsilon, \ell, L)$-list-recoverable with probability at least $1 - 2^{-\eta n}$.*

Proof of Corollary 11. Denote $L := \lfloor \left(\frac{2\ell}{\varepsilon}\right)^{2\ell/\varepsilon} \rfloor$ and $b := L + 1$. Let $\mathcal{P}$ be the b-LCL property of **not** being $(1 - R - \varepsilon, \ell, L)$-list-recoverable, and let $R_{\mathcal{P}}$ be the corresponding (random linear code) threshold rate. By Theorem 6, part 1 [38], we know that if $\mathcal{C}$ is an RLC of rate R, then the following holds for every constant $\varepsilon^* > 0$:

$$\mathbf{Pr}[\mathcal{C} \text{ satisfies } \mathcal{P}] < 1 - q^{\varepsilon^* n + b^2} \implies R < R_{\mathcal{P}} + \varepsilon^*$$

According to Theorem 8, a rate R RLC (having a sufficiently large alphabet size) satisfies $\mathcal{P}$ only with probability at most $2q^{-\frac{\varepsilon n}{8}} < 1 - q^{\varepsilon^* n + b^2}$. Therefore, $R < R_{\mathcal{P}} + \varepsilon^*$ for every $\varepsilon^* > 0$, and so $R \leq R_{\mathcal{P}}$.

We will now work with random RS codes having rate slightly less than R. Define $R' := R - \varepsilon' \leq R_{\mathcal{P}} - \varepsilon'$, take n to be large enough so that $\varepsilon' n \geq 2b(b+1)$. Note that $q > Rnb > R'nb$. Define $\mathcal{C} = \mathsf{RS}_{\mathbb{F}_q}(\alpha_1, \ldots, \alpha_n; R'n)$, where $\alpha_1, \ldots, \alpha_n$ are sampled independently and uniformly from $\mathbb{F}_q$. Upon denoting $\mathcal{F}_{\mathcal{P}}$ to be the local profile family associated with property $\mathcal{P}$, we see that the hypothesis of Theorem 7 [38] is satisfied, and therefore, Equation 3 is satisfied.

Recall that we calculated an upper bound for $|\mathcal{F}_{\mathcal{P}}|$ in Equation 2, and so $|\mathcal{F}_{\mathcal{P}}| \leq \binom{n}{(1-R-\varepsilon)n}^b \cdot \ell^{bn}$. Substituting this bound on $|\mathcal{F}_{\mathcal{P}}|$ in Equation 3,

$$\begin{aligned}\mathbf{Pr}[\mathcal{C} \text{ satisfies } \mathcal{P}] &\leq (2^b - 1) \cdot \left(\frac{(4b)^{4b} R' n}{\varepsilon' q}\right)^{\frac{\varepsilon' n}{2b}} \cdot |\mathcal{F}_{\mathcal{P}}| \\ &\leq \left(\frac{(4b)^{4b} R' n}{\varepsilon' q}\right)^{\frac{\varepsilon' n}{2b}} \cdot \binom{n}{(1-R-\varepsilon)n}^b \cdot \ell^{bn} \\ &\leq \left(\frac{(4(L+1))^{4(L+1)} R' n}{\varepsilon' q}\right)^{\frac{\varepsilon' n}{2(L+1)}} \cdot 2^{(H_2(1-R-\varepsilon)+1)\cdot(L+1)n}.\end{aligned}$$

Because $q > \frac{(4(L+1))^{4(L+1)} R' n}{\varepsilon'} \cdot 2^{\frac{((\log \ell + 2)(L+1)+\eta)\cdot 2(L+1)}{\varepsilon'}}$, we see that $\mathbf{Pr}[\mathcal{C} \text{ satisfies } \mathcal{P}] \leq 2^{-\eta n}$. Thus, $\mathcal{C}$ is $(1-R-\varepsilon, \ell, L)$-list-recoverable with probability at least $1 - 2^{-\eta n}$. ◀

5 Any linear code needs output list-size $\ell^{\Omega(R/\epsilon)}$

We now prove our lower bounds for list-recovery, that any linear code list-recoverable to capacity needs output list size $\ell^{\Omega(R/\varepsilon)}$.

▶ **Theorem 12.** *Let $R, \varepsilon \in (0,1)$, ℓ be a positive integer, and $n \geq n_0(\ell, R, \varepsilon)$ be sufficiently large. Let $\mathcal{C} \subseteq \mathbb{F}^n$ be a linear code of rate R. If $\mathcal{C}$ is $(1-R-\varepsilon, \ell, L)$-list-recoverable, then $L > \ell^{\lfloor R/\varepsilon \rfloor}$.*

Proof. Let $k := Rn$ be the dimension of the code. Let $k' = \left\lceil \frac{\varepsilon}{R} \cdot k \right\rceil$. Let $m = \left\lfloor \frac{k-1}{k'+1} \right\rfloor$. By Gaussian elimination and permuting rows and columns, we may, without loss of generality write the generator matrix of $\mathcal{C}$ as

$$\mathbf{G} = \left[\begin{array}{ccccc} 1 & & & & \\ & 1 & & & \\ & & \ddots & & \\ & & & \ddots & \\ & & & & 1 \\ \hline * & * & \cdots & \cdots & * \end{array}\right] \tag{5}$$

where each $*$ is a length $n-k$ vector. For $i \in [k]$, let $v_i \in \mathbb{F}^n$ denote the columns. By rank-nullity, there exist vectors $w_0, \ldots, w_{m-1} \in \mathbb{F}^n$ such that w_i is a linear combination of $v_{i\cdot(k'+1)+1}, \ldots, v_{(i+1)\cdot(k'+1)}$ such that w_i is not supported on indices $k+1, \ldots, k+k'$ (there are $k'+1$ vectors and k' indices). Now let $w_m = v_k$. Restricted to indices in $[k+k']$, vectors $w_0, \ldots, w_m$ have pairwise disjoint supports: within indices $[k+k']$, for $i = 0, \ldots, m-1$, vector w_i is supported on $i \cdot (k'+1) + 1, \ldots, (i+1) \cdot (k'+1) \leq k-1$, and vector w_m is supported on $k, \ldots, k+k'$.

Now fix ℓ arbitrary distinct values $\beta_1, \ldots, \beta_\ell \in \mathbb{F}_q$. Consider the output list $\mathcal{L} = \{\sum_{i=0}^m \beta_{r_i} w_i : r_i \in [\ell]\}$ to be all linear combinations of w_i with coefficients from $\beta_1, \ldots, \beta_\ell$. The fact that the vectors $w_0, \ldots, w_m$ have pairwise disjoint supports on $[k+k']$ implies (i) the vectors $w_0, \ldots, w_m$ are linearly independent, and so all vectors in $\mathcal{L}$ are distinct, and (ii) the vectors in $\mathcal{L}$ can only take on one of ℓ values at any index in $[k+k']$. Therefore, we can choose input lists $S_1, \ldots, S_{k+k'}$, each of size ℓ such that all codewords in $\mathcal{L}$ agree with all of $S_1, \ldots, S_{k+k'}$.

Choosing the rest of the input lists arbitrarily, we see that if this code is (ρ, ℓ, L) list-recoverable with radius $\rho = (n-k-k')/n < 1-R-\varepsilon$, then the list size satisfies $L \geq \ell^{m+1} \geq \ell^{\lfloor R/\varepsilon \rfloor}$.[3]

◀

We also show that our Zyablov-Pinsker type argument in Theorem 1 (Lemma 10) is tight, in the sense that any linear code must have $\Omega(\ell/\varepsilon)$ linearly independent codewords in a list-recovery ball.

▶ **Proposition 13.** *Let $R, \varepsilon \in (0,1)$, ℓ be a positive integer, and $n \geq n_0(\ell, R, \varepsilon)$ be sufficiently large. Let $\mathcal{C}$ be a linear code of rate R. Then there exists a $(1-R-\varepsilon)$ radius ℓ-list-recovery ball B that contains at least $\lceil (1-R)\ell/\varepsilon \rceil - 1$ linearly independent elements of $\mathcal{C}$.*

Proof. Upon writing the generator matrix of $\mathcal{C}$ in the same form as described above in the proof of Theorem 12, consider the first $m := \lceil (1-R)\ell/\varepsilon \rceil - 1 < (1-R)\ell/\varepsilon$ columns of the generator matrix. Denote these linearly independent column vectors by $v_1, \ldots, v_m$. Create input lists $S_1, \ldots, S_k$ each of size ℓ that contain 0 and 1, but are otherwise arbitrary. Then create lists $S_{k+1}, \ldots, S_n$ of size ℓ, each containing elements that are evenly distributed so that they agree equally with each of $v_1, \ldots, v_m$. Thus, each of $v_1, \ldots, v_m$ agrees with $S_1, \ldots, S_n$ on the first k, and on at least $\lfloor \frac{\ell}{m} \cdot (n-k) \rfloor > \varepsilon n$ of the remaining coordinates. Therefore, these vectors lie inside a $(1-R-\varepsilon)n$-radius ℓ-list-recovery ball around $S_1, \ldots, S_n$, as desired. ◀

6 Concluding remarks

We showed that random linear codes and Reed–Solomon codes are list-recoverable to capacity with near-optimal output-list size. Several open questions remain.

1. What is the optimal output-list size for random linear codes and Reed–Solomon codes? There is a gap between our upper bound of $(\frac{\ell}{\varepsilon})^{O(\ell/\varepsilon)}$ and the lower bound of $\ell^{\Omega(1/\varepsilon)}$. We surmise that the correct answer is closer to the lower bound.
2. As asked by Doron and Wootters [8], are there *explicit* list-recoverable codes with output list size $L = O_\varepsilon(\ell)$? (and, even better, over alphabet size $q = \mathrm{poly}(\ell)$). We showed (Theorem 5) that any such code must be nonlinear.
3. Our alphabet size for list-recovering Reed–Solomon codes (Theorem 2) is optimal in that it is linear in n, but the constant is double-exponential in ℓ/ε. By contrast, for list-decoding, the best known alphabet size for achieving capacity has an exponential-type constant, $2^{\mathrm{poly}(1/\varepsilon)} \cdot n$ [1]. Can our alphabet size be improved?

7 Acknowledgments

The authors would like to thank Yeyuan Chen and Zihan Zhang for pointing out a mistake in Theorem 12 in an earlier version of the paper.

[3] $m+1 = \left\lfloor \frac{k+k'}{k'+1} \right\rfloor \geq \left\lfloor \frac{k+\frac{\varepsilon}{R}k}{\frac{\varepsilon}{R}k+2} \right\rfloor \geq \left\lfloor \frac{R}{\varepsilon} \right\rfloor$, where we used that $k > 2R^2/\varepsilon^2$.

References

1 Omar Alrabiah, Venkatesan Guruswami, and Ray Li. Randomly punctured reed-solomon codes achieve list-decoding capacity over linear-sized fields. In Bojan Mohar, Igor Shinkar, and Ryan O'Donnell, editors, *Proceedings of the 56th Annual ACM Symposium on Theory of Computing, STOC 2024, Vancouver, BC, Canada, June 24-28, 2024*, pages 1458–1469. ACM, 2024. doi:10.1145/3618260.3649634.

2 Eli Ben-Sasson, Swastik Kopparty, and Jaikumar Radhakrishnan. Subspace Polynomials and Limits to List Decoding of Reed-Solomon Codes. *IEEE Trans. Inform. Theory*, 56(1):113–120, January 2010. doi:10.1109/TIT.2009.2034780.

3 Joshua Brakensiek, Manik Dhar, Sivakanth Gopi, and Zihan Zhang. AG codes achieve list decoding capacity over contant-sized fields. *CoRR*, abs/2310.12898, 2023. URL: https://doi.org/10.48550/arXiv.2310.12898, arXiv:2310.12898, doi:10.48550/ARXIV.2310.12898.

4 Joshua Brakensiek, Sivakanth Gopi, and Visu Makam. Generic reed-solomon codes achieve list-decoding capacity. In Barna Saha and Rocco A. Servedio, editors, *Proceedings of the 55th Annual ACM Symposium on Theory of Computing, STOC 2023, Orlando, FL, USA, June 20-23, 2023*, pages 1488–1501. ACM, 2023. doi:10.1145/3564246.3585128.

5 Yeyuan Chen and Zihan Zhang. Explicit folded reed-solomon and multiplicity codes achieve relaxed generalized singleton bounds. In Michal Koucký and Nikhil Bansal, editors, *Proceedings of the 57th Annual ACM Symposium on Theory of Computing, STOC 2025, Prague, Czechia, June 23-27, 2025*, pages 1–12. ACM, 2025. doi:10.1145/3717823.3718114.

6 Qi Cheng and Daqing Wan. On the List and Bounded Distance Decodability of Reed-Solomon Codes. *SIAM J. Comput.*, 37(1):195–209, April 2007. Place: Philadelphia, PA, USA Publisher: Society for Industrial and Applied Mathematics. URL: http://dx.doi.org/10.1137/S0097539705447335, doi:10.1137/S0097539705447335.

7 Mahdi Cheraghchi, Venkatesan Guruswami, and Ameya Velingker. Restricted isometry of Fourier matrices and list decodability of random linear codes. 42(5):1888–1914.

8 Dean Doron and Mary Wootters. High-Probability List-Recovery, and Applications to Heavy Hitters. In *Electron. Colloquium Comput. Complex.*, volume 27, page 162, 2020.

9 Zeev Dvir and Shachar Lovett. Subspace evasive sets. In *Proceedings of the forty-fourth annual ACM symposium on Theory of computing*, pages 351–358, 2012.

10 Peter Elias. Error-correcting codes for list decoding. *IEEE Transactions on Information Theory*, 37(1):5–12, 1991. Publisher: IEEE.

11 Asaf Ferber, Matthew Kwan, and Lisa Sauermann. List-decodability with large radius for Reed-Solomon codes. *IEEE Transactions on Information Theory*, 68(6):3823–3828, 2022. Publisher: IEEE.

12 Anna C Gilbert, Hung Q Ngo, Ely Porat, Atri Rudra, and Martin J Strauss. l2/l2-foreach sparse recovery with low risk. In *International Colloquium on Automata, Languages, and Programming*, pages 461–472. Springer, 2013.

13 Eitan Goldberg, Chong Shangguan, and Itzhak Tamo. Singleton-type bounds for list-decoding and list-recovery, and related results. In *2022 IEEE International Symposium on Information Theory (ISIT)*, pages 2565–2570. IEEE, 2022.

14 Zeyu Guo, Ray Li, Chong Shangguan, Itzhak Tamo, and Mary Wootters. Improved list-decodability and list-recoverability of Reed-Solomon codes via tree packings. In *2021 IEEE 62nd Annual Symposium on Foundations of Computer Science (FOCS)*, pages 708–719. IEEE, 2022.

15 Zeyu Guo and Zihan Zhang. Randomly punctured reed-solomon codes achieve the list decoding capacity over polynomial-size alphabets. In *64th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2023, Santa Cruz, CA, USA, November 6-9, 2023*, pages 164–176. IEEE, 2023. doi:10.1109/FOCS57990.2023.00019.

16 Venkatesan Guruswami. *List Decoding of Error-Correcting Codes (Winning Thesis of the 2002 ACM Doctoral Dissertation Competition)*, volume 3282 of *Lecture Notes in Computer Science*. Springer, 2004. URL: https://doi.org/10.1007/b104335, doi:10.1007/B104335.

17 Venkatesan Guruswami, Johan Håstad, and Swastik Kopparty. On the List-Decodability of Random Linear Codes. *IEEE Trans. Inform. Theory*, 57(2):718–725, February 2011. doi:10.1109/TIT.2010.2095170.

18 Venkatesan Guruswami and Piotr Indyk. Expander-based constructions of efficiently decodable codes. In *Proceedings 42nd IEEE Symposium on Foundations of Computer Science*, pages 658–667. IEEE, 2001.

19 Venkatesan Guruswami and Swastik Kopparty. Explicit subspace designs. *Combinatorica*, 36(2):161–185, 2016. Publisher: Springer.

20 Venkatesan Guruswami, Ray Li, Jonathan Mosheiff, Nicolas Resch, Shashwat Silas, and Mary Wootters. Bounds for list-decoding and list-recovery of random linear codes. *IEEE Transactions on Information Theory*, 68(2):923–939, 2021. Publisher: IEEE.

21 Venkatesan Guruswami and Atri Rudra. Limits to List Decoding Reed–Solomon Codes. *IEEE Trans. Inform. Theory*, 52(8):3642–3649, August 2006. Place: Piscataway, NJ, USA Publisher: IEEE Press. doi:10.1109/TIT.2006.878164.

22 Venkatesan Guruswami and Atri Rudra. Explicit codes achieving list decoding capacity: Error-correction with optimal redundancy. *IEEE Transactions on Information Theory*, 54(1):135–150, 2008. Publisher: IEEE.

23 Venkatesan Guruswami, Atri Rudra, and Madhu Sudan. Essential coding theory. *Draft available at http://cse.buffalo.edu/faculty/atri/courses/coding-theory/book/*, 2019.

24 Venkatesan Guruswami, Atri Rudra, and Madhu Sudan. Essential coding theory. *Draft available at https://cse.buffalo.edu/faculty/atri/courses/coding-theory/book/*, 2022.

25 Venkatesan Guruswami and Madhu Sudan. Improved Decoding of Reed–Solomon and Algebraic-Geometry Codes. *IEEE Transactions on Information Theory*, 45(6):1757–1767, 1999. doi:10.1109/18.782097.

26 Venkatesan Guruswami and Madhu Sudan. Extensions to the Johnson bound. *Manuscript, February*, 2001.

27 Venkatesan Guruswami, Christopher Umans, and Salil P. Vadhan. Unbalanced expanders and randomness extractors from parvaresh-vardy codes. *J. ACM*, 56(4):20:1–20:34, 2009. doi:10.1145/1538902.1538904.

28 Venkatesan Guruswami and Carol Wang. Linear-algebraic list decoding for variants of Reed–Solomon codes. *IEEE Transactions on Information Theory*, 59(6):3257–3268, 2013. Publisher: IEEE.

29 Venkatesan Guruswami and Chaoping Xing. Folded codes from function field towers and improved optimal rate list decoding. In *Proceedings of the forty-fourth annual ACM symposium on Theory of computing*, pages 339–350, 2012.

30 Venkatesan Guruswami and Chaoping Xing. List decoding Reed-Solomon, Algebraic-Geometric, and Gabidulin subcodes up to the Singleton bound. In *Proceedings of the forty-fifth annual ACM symposium on Theory of computing*, pages 843–852, 2013.

31 Brett Hemenway, Noga Ron-Zewi, and Mary Wootters. Local list recovery of high-rate tensor codes and applications. *SIAM Journal on Computing*, 49(4):FOCS17–157, 2019. Publisher: SIAM.

32 Brett Hemenway and Mary Wootters. Linear-time list recovery of high-rate expander codes. *Information and Computation*, 261:202–218, 2018. Publisher: Elsevier.

33 Piotr Indyk, Hung Q Ngo, and Atri Rudra. Efficiently decodable non-adaptive group testing. In *Proceedings of the twenty-first annual ACM-SIAM symposium on Discrete Algorithms*, pages 1126–1142. SIAM, 2010.

34 Selmer Johnson. A new upper bound for error-correcting codes. *IRE Transactions on Information Theory*, 8(3):203–207, 1962. Publisher: IEEE.

35 Swastik Kopparty. List-decoding multiplicity codes. *Theory of Computing*, 11(1):149–182, 2015. Publisher: Theory of Computing Exchange.

36 Swastik Kopparty, Noga Ron-Zewi, Shubhangi Saraf Saraf, and Mary Wootters. Improved decoding of folded Reed-Solomon and multiplicity codes. In *2018 IEEE 59th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 212–223. IEEE, 2018.

37 Kasper Green Larsen, Jelani Nelson, Huy L Nguyen, and Mikkel Thorup. Heavy hitters via cluster-preserving clustering. *Communications of the ACM*, 62(8):95–100, 2019.

38 Matan Levi, Jonathan Mosheiff, and Nikhil Shagrithaya. Random Reed-Solomon Codes and Random Linear Codes are Locally Equivalent, November 2024. arXiv:2406.02238. URL: http://arxiv.org/abs/2406.02238, doi:10.48550/arXiv.2406.02238.

39 Ray Li and Mary Wootters. Improved list-decodability of random linear binary codes. *IEEE Transactions on Information Theory*, 67(3):1522–1536, 2020. Publisher: IEEE.

40 Ben Lund and Aditya Potukuchi. On the List Recoverability of Randomly Punctured Codes. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX/RANDOM 2020)*, volume 176, pages 30:1–30:11, 2020.

41 Hung Q Ngo, Ely Porat, and Atri Rudra. Efficiently decodable error-correcting list disjunct matrices and applications. In *Automata, Languages and Programming: 38th International Colloquium, ICALP 2011, Zurich, Switzerland, July 4-8, 2011, Proceedings, Part I 38*, pages 557–568. Springer, 2011.

42 I. S. Reed and G. Solomon. Polynomial codes over certain finite fields. *Journal of the Society for Industrial and Applied Mathematics*, 8(2):300–304, 1960. arXiv:https://doi.org/10.1137/0108018, doi:10.1137/0108018.

43 Nicolas Resch. List-decodable codes:(randomized) constructions and applications. 2020.

44 Atri Rudra and Mary Wootters. Every list-decodable code for high noise has abundant near-optimal rate puncturings. In David B. Shmoys, editor, *Symposium on Theory of Computing, STOC 2014, New York, NY, USA, May 31 - June 03, 2014*, pages 764–773. ACM, 2014. doi:10.1145/2591796.2591797.

45 Atri Rudra and Mary Wootters. It'll probably work out: Improved list-decoding through random operations. In Tim Roughgarden, editor, *Proceedings of the 2015 Conference on Innovations in Theoretical Computer Science, ITCS 2015, Rehovot, Israel, January 11-13, 2015*, pages 287–296. ACM, 2015. doi:10.1145/2688073.2688092.

46 Atri Rudra and Mary Wootters. Average-radius list-recoverability of random linear codes. In *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 644–662. SIAM, 2018.

47 Chong Shangguan and Itzhak Tamo. Combinatorial list-decoding of Reed-Solomon codes beyond the Johnson radius. In *Proceedings of the 52nd Annual ACM Symposium on Theory of Computing*, STOC 2020, pages 538–551, 2020.

48 Shashank Srivastava. Improved list size for folded reed-solomon codes. In Yossi Azar and Debmalya Panigrahi, editors, *Proceedings of the 2025 Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2025, New Orleans, LA, USA, January 12-15, 2025*, pages 2040–2050. SIAM, 2025. doi:10.1137/1.9781611978322.64.

49 Amnon Ta-Shma and David Zuckerman. Extractor codes. *IEEE Trans. Inf. Theory*, 50(12):3015–3025, 2004. doi:10.1109/TIT.2004.838377.

50 Itzhak Tamo. Tighter List-Size Bounds for List-Decoding and Recovery of Folded Reed-Solomon and Multiplicity Codes, December 2023. arXiv:2312.17097. URL: http://arxiv.org/abs/2312.17097.

51 Mary Wootters. On the List Decodability of Random Linear Codes with Large Error Rates. In *Proceedings of the Forty-fifth Annual ACM Symposium on Theory of Computing*, STOC '13, pages 853–860, New York, NY, USA, 2013. ACM. event-place: Palo Alto, California, USA. URL: http://doi.acm.org/10.1145/2488608.2488716, doi:10.1145/2488608.2488716.

52 Victor Vasilievich Zyablov and Mark Semenovich Pinsker. List concatenated decoding. *Problemy Peredachi Informatsii*, 17(4):29–33, 1981. Publisher: Russian Academy of Sciences, Branch of Informatics, Computer Equipment and ....