Combinatorial Limitations of Average-radius List Decoding *

Venkatesan Guruswami** and Srivatsan Narayanan

Computer Science Department
Carnegie Mellon University
{venkatg,srivatsa}@cs.cmu.edu

Abstract. We study certain combinatorial aspects of list-decoding, motivated by the exponential gap between the known upper bound (of $O(1/\gamma)$) and lower bound (of $\Omega_p(\log(1/\gamma))$) for the list-size needed to list decode up to error fraction p with rate γ away from capacity, i.e., $1-h(p)-\gamma$ (here $p \in (0, \frac{1}{2})$ and $\gamma > 0$). Our main result is the following:

> We prove that in any binary code $C \subseteq \{0,1\}^n$ of rate $1-h(p)-\gamma$, there must exist a set $\mathcal{L} \subset C$ of $\Omega_p(1/\sqrt{\gamma})$ codewords such that the average distance of the points in $\mathcal{L}$ from their centroid is at most pn. In other words, there must exist $\Omega_p(1/\sqrt{\gamma})$ codewords with low "average radius." The standard notion of list-decoding corresponds to working with the *maximum* distance of a collection of codewords from a center instead of *average* distance. The average-radius form is in itself quite natural; for instance, the classical Johnson bound in fact implies average-radius list-decodability.

The remaining results concern the standard notion of list-decoding, and help clarify the current state of affairs regarding combinatorial bounds for list-decoding:

- We give a short simple proof, over all fixed alphabets, of the above-mentioned $\Omega_p(\log(1/\gamma))$ lower bound. Earlier, this bound followed from a complicated, more general result of Blinovsky.
- We show that one *cannot* improve the $\Omega_p(\log(1/\gamma))$ lower bound via techniques based on identifying the zero-rate regime for list-decoding of constant-weight codes. On a positive note, our $\Omega_p(1/\sqrt{\gamma})$ lower bound for average-radius list-decoding circumvents this barrier.
- We exhibit a "reverse connection" between the existence of constant-weight and general codes for list-decoding, showing that the best possible list-size, as a function of the gap γ of the rate to the capacity limit, is the same up to constant factors for both constant-weight codes (whose weight is bounded away from p) and general codes.
- We give simple second moment based proofs that w.h.p. a list-size of $\Omega_p(1/\gamma)$ is needed for list-decoding *random* codes from errors as well as erasures. For *random linear* codes, the corresponding list-size bounds are $\Omega_p(1/\gamma)$ for errors and $\exp(\Omega_p(1/\gamma))$ for erasures.

* Full version can be found at http://eccc.hpi-web.de/report/2012/017/

** Research supported in part by NSF grant CCF 0953155 and a Packard Fellowship.

1 Introduction

The list-decoding problem for an error-correcting code $C \subseteq \Sigma^n$ consists of finding the set of all codewords of C with Hamming distance at most pn from an input string $y \in \Sigma^n$. Though it was originally introduced in early work of Elias and Wozencraft [6, 15] in the context of estimating the decoding error probability for random error models, recently the main interest in list-decoding has been for adversarial error models. List decoding enables correcting up to a factor two more worst-case errors compared to algorithms that are always restricted to output a unique answer, and this potential has even been realized algorithmically [10, 8].

In this work, we are interested in some fundamental combinatorial questions concerning list-decoding, which highlight the important tradeoffs in this model. Fix $p \in (0, \frac{1}{2})$ and a positive integer L. We say that a binary code $C \subseteq \{0,1\}^n$ is (p, L) list-decodable if every Hamming ball of radius pn has *less than* L codewords. Here, p corresponds to the error-fraction and L to the list-size needed by the error-correction algorithm. Note that (p, L) list-decodability imposes a sparsity requirement on the distribution of codewords in the Hamming space. A natural combinatorial question that arises in this context is to place bounds on the largest size of a code meeting this requirement. In particular, an outstanding open question is to characterize the maximum rate (defined to be the limiting ratio $\frac{1}{n} \log |C|$ as $n \to \infty$) of a (p, L) list-decodable code.

By a simple volume packing argument, it can be shown that a (p, L) list-decodable code has rate at most $1 - h(p) + o(1)$. (Throughout, for $z \in [0, \frac{1}{2}]$, we use $h(z)$ to denote the binary entropy function at z.) Indeed, picking a random center x, the Hamming ball $\mathbf{B}(x, pn)$ contains at least $|C| \cdot \binom{n}{pn} 2^{-n}$ codewords in expectation. Bounding this by $(L-1)$, we get the claim. On the positive side, in the limit of large L, the rate of a (p, L) list-decodable code approaches the optimal $1 - h(p)$. More precisely, for any $\gamma > 0$, there exists a $(p, 1/\gamma)$ list-decodable code of rate at least $1 - h(p) - \gamma$. In fact, a random code of rate $1 - h(p) - \gamma$ is $(p, 1/\gamma)$ list-decodable w.h.p. [16, 7], and a similar result holds for random linear codes (with list-size $O_p(1/\gamma)$) [9]. In other words, a dense random packing of $2^{(1-h(p)-\gamma)n}$ Hamming balls of radius pn (and therefore volume $\approx 2^{h(p)n}$ each) is "near-perfect" w.h.p. in the sense that no point is covered by more than $O_p(1/\gamma)$ balls.

The determination of the best asymptotic code rate of binary (p, L) list-decodable codes as p, L are held fixed and the block length grows is wide open for every choice of $p \in (0, \frac{1}{2})$ and integer $L \geqslant 1$. However, we *do* know that for each fixed $p \in (0, \frac{1}{2})$, this rate approaches $1 - h(p)$ in the limit as $L \to \infty$. To understand this rate of convergence as a function of list-size L, following [9], let us define $L_{p,\gamma}$ to be the minimum integer L such that there exist (p, L) list-decodable codes of rate $1 - h(p) - \gamma$ for infinitely many block lengths n (the quantity γ is the "gap" to "list-decoding capacity"). In [1], Blinovsky showed that a (p, L) list-decodable code has rate at most $1 - h(p) - 2^{-\Theta_p(L)}$. In particular, this implies that for any finite L, a (p, L) list-decodable code has rate strictly below the optimal $1 - h(p)$. Stated in terms of $L_{p,\gamma}$, his result implies the corollary $L_{p,\gamma} \geqslant \Omega_p(\log(1/\gamma))$ for rates γ-close to capacity. We provide a short and simple

proof of this corollary in Section 4. Our proof works almost as easily over non-binary alphabets. (Blinovsky's subsequent proof for the non-binary case in [3, 4] involved substantial technical effort. However, his results also give non-trivial bounds for every finite L, as opposed to just the growth rate of $L_{p,\gamma}$.)

Observe the exponential gap (in terms of the dependence on γ) between the $O(1/\gamma)$ upper bound and $\Omega_p(\log(1/\gamma))$ lower bounds on the quantity $L_{p,\gamma}$. Despite being a basic and fundamental question about sphere packings in the Hamming space and its direct relevance to list-decoding, there has been no progress on narrowing this asymptotic gap in the 25 years since the works of Zyablov-Pinsker [16] and Blinovsky [1]. This is the motivating challenge driving this work.

1.1 Prior work on list-size lower bounds

We now discuss some lower bounds (besides Blinovsky's general lower bound) on list-size that have been obtained in restricted cases.

Rudra shows that the $O_p(1/\gamma)$ bound obtained via the probabilistic method for random codes is, in fact, tight up to constant factors [14]. Formally, there exists $L = \Omega_p(1/\gamma)$ such that a random code of rate $1 - h(p) - \gamma$ is *not* (p, L) list-decodable w.h.p. His proof uses near-capacity-achieving codes for the binary symmetric channel, the existence of which is promised by Shannon's theorem, followed by a second moment argument. We give a simpler proof of this result via a more direct use of the second moment method. This has the advantage that it works uniformly for random general as well as random linear codes, and for channels that introduce errors as well as erasures.

Guruswami and Vadhan [12] consider the problem of establishing list-size bounds when the channel may corrupt close to half the bits, that is, when $p = \frac{1}{2} - \varepsilon$, and more generally $p = 1 - 1/q - \varepsilon$ for codes over an alphabet of size q. (Note that decoding is impossible if the channel could corrupt up to a half fraction of bits.) They show that there exists $c > 0$ such that for all $\varepsilon > 0$ and all block lengths n, any $(\frac{1}{2} - \varepsilon, c/\varepsilon^2)$ list-decodable code contains $O_\varepsilon(1)$ codewords. For p bounded away from $\frac{1}{2}$ (or $1 - 1/q$ in the q-ary case), their methods do not yield any nontrivial list-size lower bound as a function of gap γ to list-decoding capacity.

1.2 Our main results

We have already mentioned our new proof of the $\Omega(\log(1/\gamma))$ list-size lower bound for list-decoding general codes, and the asymptotically optimal list-size lower bound for random (and random linear) codes.

Our main result concerns an average-radius variant of list-decoding. This variant was implicitly used in [1, 12] en route their list-size lower bounds for standard list-decoding. In this work, we formally abstract this notion of *average-radius list-decodability*: a code is (p, L) *average-radius list-decodable* if for every L codewords, the *average* distance of their centroid from the L codewords exceeds pn. Note that this is a stronger requirement than (p, L) list-decodability where only the *maximum* distance from any center point to the L codewords must exceed pn.

We are able to prove nearly tight bounds on the achievable rate of a (p, L) average-radius list-decodable code. To state our result formally, denote by $L^{\mathsf{avg}}_{p,\gamma}$ the minimum L such that there exists a (p, L) average-radius list-decodable code family of rate $1-h(p)-\gamma$. A simple random coding argument shows that a random code of $1 - h(p) - \gamma$ is $(p, 1/\gamma)$ average-radius list-decodable (matching the list-decodability of random codes). That is, $L^{\mathsf{avg}}_{p,\gamma} \leqslant 1/\gamma$. Our main technical result is a lower bound on the list-size that is polynomially related to the upper bound, namely $L^{\mathsf{avg}}_{p,\gamma} \geqslant \Omega_p(\gamma^{-1/2})$.

We remark that the classical Johnson bound in coding theory in fact proves the average-radius list-decodability of codes with good minimum distance — namely, a binary code of relative distance δ is $(J(\delta - \delta/L), L)$ average-radius list-decodable, where $J(z) = (1 - \sqrt{1-2z})/2$ for $z \in [0, \frac{1}{2}]$. (This follows from a direct inspection of the proof of the Johnson bound [11].) Also, one can show that if a binary code is $(\frac{1}{2} - 2^i\varepsilon, O(1/(2^{2i}\varepsilon^2))$ list-decodable for all $i = 0, 1, 2, \ldots$, then it is also $(\frac{1}{2} - 2\varepsilon, O(1/\varepsilon^2))$ average-radius list-decodable [5]. This shows that at least in the high noise regime, there is some reduction between these notions. Further, a suitable soft version of average-radius list-decodability can be used to construct matrices with a certain restricted isometry property [5]. For these reasons, we feel that average-radius list-decodability is a natural notion to study, even beyond treating it as a vehicle to understand (standard) list-decoding.

1.3 Our other results

We also prove several other results that clarify the landscape of combinatorial limitations of list-decodable codes. Many results showing rate limitations in coding theory proceed via a typical approach in which they pass to a constant weight $\lambda \in (p, \frac{1}{2}]$; i.e., they restrict the codewords to be of weight exactly λn. They show that under this restriction, a code with the stated properties must have a constant number of codewords (that is, asymptotically *zero rate*). Mapping this bound back to the unrestricted setting one gets a rate upper bound of $1 - h(\lambda) + o(1)$ for the original problem. For instance, the Elias-Bassalygo bound for rate R vs. relative distance δ is of this nature (here λ is picked to be the Johnson radius for list-decoding for codes of relative distance δ).

The above is also the approach taken in Blinovsky's work [1] as well as that of [12]. We show that such an approach does not and *cannot* give any bound better than Blinovsky's $\Omega_p(\log(1/\gamma))$ bound for $L_{p,\gamma}$. More precisely, for any $\lambda \geqslant p + 2^{-b_p L}$ for some $b_p > 0$, we show that there exists a (p, L) (average-radius) list-decodable code of rate $\Omega_{p,L}(1)$. Thus in order to improve the lower bound, we *must* be able to handle codes of strictly positive rate, and cannot deduce the bound by pinning down the zero-rate regime of constant-weight codes. This perhaps points to why improvements to Blinovsky's bounds have been difficult. On a positive note, we remark that we *are* able to effect such a proof for average-radius list-decoding (some details follow next).

To describe the method underlying our list-size lower bound for average-radius list-decoding, it is convenient to express the statement as an upper bound on rate in terms of list-size L. Note that a list-size lower bound of $L \geqslant \Omega_p(1/\sqrt{\gamma})$ for (p, L) average-radius list-decodable codes of rate $1 - h(p) - \gamma$ amounts to

proving an upper bound of $1 - h(p) - \Omega_p(1/L^2)$ on the rate of (p, L) average-radius list-decodable codes. Our proof of such an upper bound proceeds by first showing a rate upper bound of $h(\lambda) - h(p) - a_p/L^2$ for such codes when the codewords are all restricted to have weight λn, for a suitable choice of λ, namely $\lambda = p + a'_p/L$. To map this back to the original setting (with no weight restrictions on codewords), one simply notes that every (p, L) average-radius list-decodable code of rate R contains as a subcode, a translate of a constant λn-weight subcode of rate $R - (1 - h(\lambda))$. (The second step uses a well-known argument.)

Generally speaking, by passing to a constant-weight subcode, one can translate combinatorial results on limitations of constant-weight codes to results showing limitations for the case of general codes. But this leaves open the possibility that the problem of showing limitations of constant-weight codes may be harder than the corresponding problem for general codes, or worse still, have a different answer making it impossible to solve the problem for general codes via the methodology of passing to constant-weight codes. We show that for the problem of list-decoding this is fortunately not the case, and there is, in fact, a "reverse connection" of the following form: A rate upper bound of $1 - h(p) - \gamma$ for (p, L) list-decodable codes implies a rate upper bound of $h(\lambda) - h(p) - \left(\frac{\lambda - p}{\frac{1}{2} - p}\right)\gamma$ for (p, L) list-decodable codes whose codewords must all have Hamming weight λn. A similar claim holds also for average-radius list-decodability, though we don't state it formally.

1.4 Our proof techniques

Our proofs in this paper employ variants of the standard probabilistic method. We show an extremely simple probabilistic argument that yields a $\Omega_p(\log(1/\gamma))$ bound on the list-size of a standard list-decodable code; we emphasize that this is qualitatively the tightest known bound in this regime.

For the "average-radius list-decoding" problem that we introduce, we are able to improve this list-size bound to $\Omega_p(1/\sqrt{\gamma})$. The proof is based on the idea that instead of picking the "bad list-decoding center" x uniformly at random, one can try to pick it randomly very close to a special codeword c^*, and this still gives similar guarantees on the number of near-by codewords. Now since the quantity of interest is the average radius, including this close-by codeword in the list gives enough savings for us. In order to estimate the probability that a typical codeword c belongs to the list around x, we write this probability explicitly as a function of the Hamming distance between c^* and c, which is then lower bounded using properties of hypergeometric distributions and Taylor approximations for the binary entropy function.

For limitations of list-decoding random codes, we define a random variable W that counts the number of "violations" of the list-decoding property of the code. We then show that W has a exponentially large mean, around which it is concentrated w.h.p. This yields that the code cannot be list-decodable with high probability, for suitable values of rate and list-size parameters. We skip the formal statement of these results and their proofs in this version (due to space restrictions); these can be found in the full version.

1.5 Organization

We define some useful notation and the formal notion of average-radius list-decodability in Section 2. Our main list-size lower bound for average-radius list-decoding appears in Section 3. We give our short proof of Blinovsky's lower bounds for binary and general alphabets in Section 4. Our results about the zero-error rate regime for constant-weight codes and the connection between list-decoding bounds for general codes and constant-weight codes appear in Section 5. For reasons of space, many of the proofs, and all results on list-size lower bounds for random codes, are skipped and can be found in the full version.

2 Notation and Preliminaries

2.1 List decoding

We recall some standard terminology regarding error-correcting codes. Let $[n]$ denote the index set $\{1, 2, \ldots, n\}$. For $q \geqslant 2$, let $[q]$ denote the set $\{0, 1, \ldots, q-1\}$. A *q-ary code* refers to any subset $C \subseteq [q]^n$, where n is the *blocklength* of C. We will mainly focus on the special case of binary codes corresponding to $q = 2$. The rate $R = R(C)$ is defined to be $\frac{\log |C|}{n \log q}$. For $x \in [q]^n$ and $S \subseteq [n]$, the restriction of x to the coordinates in S is denoted $x|_S$. Let $\mathrm{Supp}(x) := \{i \in [n] \ : \ x_i \neq 0\}$. A *subcode* of C is a subset C' of C. We say that C is a *constant-weight code* with weight $w \in [0, n]$, if all its codewords have weight exactly w. (Such codes are studied in Section 5.)

For $x, y \in [q]^n$, define the *Hamming distance* between x and y, denoted $d(x, y)$, to be the number of coordinates in which x and y differ. The *(Hamming) weight* of x, denoted $\mathrm{wt}(x)$, is $d(\mathbf{0}, x)$, where $\mathbf{0}$ is the vector in $[q]^n$ with zeroes in all coordinates. The *(Hamming) ball* of radius r centered at x, denoted $\mathbf{B}(x, r)$, is the set $\{y \in [q]^n \ : \ d(x, y) \leqslant r\}$. In this paper, we need the following nonstandard measure of distance of a (small) "list" $\mathcal{L}$ of vectors from a "center" x: For a nonempty $\mathcal{L} \subseteq [q]^n$, define

$$D_{\mathsf{max}}(x, \mathcal{L}) := \max\{d(x, y) \ : \ y \in \mathcal{L}\},$$

and

$$D_{\mathsf{avg}}(x, \mathcal{L}) := \mathbf{E}_{y \in \mathcal{L}} \left[d(x, y) \right] = \frac{1}{|\mathcal{L}|} \sum_{y \in \mathcal{L}} d(x, y).$$

We formalize the error recovery capability of the code using list-decoding.

Definition 1. *Fix $0 < p < \frac{1}{2}$ and a positive integer L. Let C be a q-ary code with blocklength n.*

1. *C is said to be* (p, L) list-decodable *if for all $x \in [q]^n$, $\mathbf{B}(x, pn)$ contains at most $L - 1$ codewords from C. Equivalently, for any x and any list $\mathcal{L} \subseteq C$ of size at least L, we have $D_{\mathsf{max}}(x, \mathcal{L}) > pn$.*
2. *C is said to be* (p, L) average-radius list-decodable *if for any x and $\mathcal{L}$ as in Item 1, we have $D_{\mathsf{avg}}(x, \mathcal{L}) > pn$.*

For constant-weight codes, it is convenient to augment the notation with the weight parameter:

Definition 2. *Let p, L, q, n, C be as in Definition 1, and let $0 < \lambda \leqslant \frac{1}{2}$. C is said to be $(\lambda; p, L)$* (average-radius) list-decodable *if C is (p, L) (average-radius) list-decodable, and every codeword in C has weight exactly λn.*

We remark that the list-decodability property is standard in literature. Moreover, while the notion of (p, L) *average-radius* list-decodability is formally introduced by this paper, it is already implicit in [1, 2, 12].

Since the max-distance of a list from a center always dominates its average distance, every (p, L) average-radius list-decodable code is also (p, L) list-decodable. That is, average-radius list-decodability is a syntactically stronger property than its standard counterpart, and hence any limitation we establish for the standard list-decodable codes also carries over for average-radius list-decodability.

Following (and extending) the notation in [9], we make the following definitions to quantify the tradeoffs in the different parameters of a code: the rate R, the error-correction radius p, list-size L, and the weight λ of the codewords (for constant-weight codes). Further, for general codes (without the constant-weight restriction), it is usually more convenient to replace the rate R by the parameter $\gamma := 1 - h(p) - R$; this measures the "gap" to the "limiting rate" or the "capacity" of $1 - h(p)$ for $(p, O(1))$ list-decodable codes.

Fix $p, \lambda \in (0, \frac{1}{2}]$ such that $p < \lambda$, $R \in (0, 1)$, and a positive integer L.

Definition 3. *1. Say that the triple $(p, L; R)$ is* achievable for list-decodable codes *if there exist (p, L) list-decodable codes of rate R for infinitely many lengths n.*

Define $R_{p,L}$ to be the supremum over R such that $(p, L; R)$ is achievable for list-decodable codes, and $\gamma_{p,L} := 1 - h(p) - R_{p,L}$. Also define $L_{p,\gamma}$ to be the least integer L such that $(p, L; 1 - h(p) - \gamma)$ is achievable.

*2. (***For constant weight codes.)** *Say that the 4-tuple $(\lambda; p, L; R)$ is achievable if there exists $(\lambda; p, L)$ list-decodable codes of rate R. Define $R_{p,L}(\lambda)$ to be the supremum rate R for which the 4-tuple $(\lambda; p, L; R)$ is achievable.*

We can also define analogous quantities for average-radius list-decoding (denoted by a superscript avg), but to prevent notational clutter, we will not explicitly do so. Throughout this paper, p is treated as a fixed constant in $(0, \frac{1}{2})$, and we will not attempt to optimize the dependence of our bounds on p.

2.2 Standard distributions and functions

In this paper, we use 'log' for logarithms to base 2 and 'ln' for natural logarithms. Also, to avoid cumbersome notation, we often denote b^z by $\exp_b(z)$. Standard asymptotic notation (big O, little o, and big Omega) is employed liberally in this paper; when subscripted by a parameter (typically p), the notation hides a constant depending arbitrarily on the parameter.

Our proofs also make repeated use of *hypergeometric distributions*, which we review here for the sake of completeness, as well as to set the notation. Suppose a set contains n objects, exactly $m < n$ of which are $m$arked, and suppose we $s$ample $s < n$ objects uniformly at random from the set *without replacement*. Then the random variable T counting the number of marked objects in the sample follows the hypergeometric distribution with parameters (n, m, s). A simple counting argument shows that, for $t \leqslant \min\{m, s\}$,

$$\mathbf{Pr}[T = t] = \frac{\binom{m}{t}\binom{n-m}{s-t}}{\binom{n}{s}}.$$

We will denote the above expression by $f(n, m, s, t)$. By convention, $f(n, m, s, t)$ is set to 0 if $n < \max\{m, s\}$ or $t > \min\{m, s\}$. Hypergeometric distributions satisfy a useful symmetry property:

Lemma 1. *For all integers n, m, s with $n \geqslant \max\{m, s\}$, the hypergeometric distribution with parameters (n, m, s) is identical to that with parameters (n, s, m). That is, for all t, we have $f(n, m, s, t) = f(n, s, m, t)$.*

Throughout this paper, we are especially concerned with the asymptotic behaviour of binomial coefficients, which is characterized in terms of the *binary entropy function*, defined as $h(z) := -z \log z - (1 - z) \log(1 - z)$. We will use the following standard estimate without proof: For $z \in (0, 1)$ and $n \to \infty$, if zn is an integer, then

$$\exp_2(h(z)n - o(n)) \leqslant \binom{n}{zn} \leqslant \sum_{i=0}^{zn} \binom{n}{i} \leqslant \exp_2(h(z)n).$$

3 Bounds for average-radius list-decodability

In this section, we bound the rate of a (p, L) average-radius list-decodable code as:

$$1 - h(p) - \frac{1}{L} - o(1) \leqslant R \leqslant 1 - h(p) - \frac{a_p}{L^2} + o(1),$$

where a_p is a constant depending only on p. (Here p is a fixed constant bounded away from 0 and $\frac{1}{2}$.) Note that, ignoring the dependence on p, the corresponding upper and lower bounds on $\gamma := 1 - h(p) - R$ are polynomially related.

We first state the rate lower bound.

Theorem 1. *Fix $p \in (0, \frac{1}{2})$ and a positive integer L. Then, for all $\varepsilon > 0$ and all sufficiently large lengths n, there exists a (p, L) average-radius list-decodable code of rate at least $1 - h(p) - 1/L - \varepsilon$.*

In fact, a random code of the above rate has the desired property w.h.p. This calculation is routine and omitted here, and can be found in the full version.

We now show an upper bound of $1 - h(p) - a_p/L^2$ on the rate of a (p, L) average-radius list-decodable code. As stated in the Introduction, the main idea

behind the construction is that instead of picking the "bad list decoding center" x uniformly at random, we pick it randomly *very close to a designated codeword* c^* (which itself is a uniformly random element from C). Now as long as we are guaranteed to find a list of $L-1$ other codewords near x, we can include c^* in our list to lower the average radius of the list.

However formalizing the above intuition into a proof is nontrivial, since our restriction of the center x to be very close to c^* introduces statistical dependencies while analyzing the number of codewords near x. We are able to control these dependencies, but this requires some heavy calculations involving the entropy function and hypergeometric distribution.

We are now ready to state our main result establishing a rate upper bound for (p, L) average-radius list-decodable codes. In fact, the bulk of the work is to show an analogous upper bound for the special case of a constant-weight code C, i.e., all codewords have weight exactly λn, for some $\lambda \in (p, \frac{1}{2}]$. We can then map this bound for general codes using a standard argument (given in Lemma 2).

Theorem 2 (Main theorem). *Fix $p \in (0, \frac{1}{2})$, and let L be a sufficiently large positive integer. Then there exist $a_p, a'_p > 0$ (depending only on p) such that the following holds (for sufficiently large lengths n):*

1. *If C is a (p, L) average-radius list-decodable code, then C has rate at most $1 - h(p) - a_p/L^2 + o(1)$.*
2. *For $\lambda := p + a'_p/L$, if C is a $(\lambda; p, L)$ average-radius list-decodable code, then C has rate at most $h(\lambda) - h(p) - a_p/L^2 + o(1)$.*

As already mentioned in Section 1.3, the second claim readily implies the first via the following well-known argument (a partial converse to this statement for list-decoding will be given in Section 5):

Lemma 2. *Let $\lambda \in (p, \frac{1}{2}]$ be such that λn is an integer. If C is a (p, L) average-radius list-decodable code of rate $R = 1 - h(p) - \gamma$, then there exists a $(\lambda; p, L)$ average-radius list-decodable code C' of rate at least $R' - o(1)$, where $R' := h(\lambda) - h(p) - \gamma$.*

Proof: For a random center x, let $C'(x)$ be the subcode of C containing the codewords c with $d(x, c) = \lambda n$. The expected size of $C'(x)$ at least $|C| \cdot \binom{n}{\lambda n} 2^{-n}$, which, for the assumed value of R, is $\exp_2(R'n - o(n))$; thus for some x, $C'(x)$ has rate at least $R' - o(1)$. The claim follows by translating $C'(x)$ by $-x$. □

Before we proceed to the proof of (the first part of) Theorem 2, we will establish the following folklore result, whose proof illustrates our idea in a simple case.

Lemma 3 (A warm-up lemma). *Fix p, λ so that $p < \lambda \leqslant \frac{1}{2}$. Then, if C is a $(\lambda; p, L)$ list-decodable code, then C has rate at most $h(\lambda) - h(p) + o(1)$.*

Proof: The main idea behind the proof is that a random center of a *particular weight* (carefully chosen) is close to a large number of codewords in expectation. Pick a random subset $S \subseteq [n]$ of coordinates of size αn, with $\alpha := (\lambda - p)/(1 - 2p)$, and let $\overline{S} = [n] \setminus S$. (The motivation for this choice of α will be clear shortly.) Define the center x be the indicator vector of S, so that $\mathrm{Supp}(x) = S$.

Consider the set $\mathcal{L}$ of codewords $c \in C$ such that $\mathrm{wt}(c|_S) \geqslant (1-p)\alpha n$; this is our candidate bad list of codewords. Then each $c \in \mathcal{L}$ is close to c:

$$d(x,c) = (\alpha n - \mathrm{wt}(c|_S)) + \mathrm{wt}(c|_{\overline{S}}) \leqslant \alpha p n + (\lambda - \alpha(1-p))n = (\lambda - \alpha(1-2p))n,$$

which equals pn for the given choice of α. Hence the size of $\mathcal{L}$ is a lower bound on the list-size of the code.

We complete the proof by computing $\mathbf{E}\,|\mathcal{L}|$. For any fixed $c \in C$, the random variable $\mathrm{wt}(c|_S)$ follows the hypergeometric distribution with parameters $(n, \lambda n, \alpha n)$, , which is identical to the hypergeometric distribution with parameters $(n, \alpha n, \lambda n)$ (see Lemma 1). Hence the probability that c is included in the list $\mathcal{L}$ is at least

$$f(n, \alpha n, \lambda n, \alpha(1-p)n) := \frac{\binom{\alpha n}{(1-p)\alpha n}\binom{(1-\alpha)n}{(\lambda-\alpha(1-p))n}}{\binom{n}{\lambda n}} = \frac{\binom{\alpha n}{p\alpha n}\binom{(1-\alpha)n}{p(1-\alpha)n}}{\binom{n}{\lambda n}},$$

where the second step holds because of the identity $\lambda - (1-p)\alpha = p(1-\alpha)$, which holds for our particular choice of α. As $n \to \infty$, this is equal to

$$\exp_2\left(\alpha n h(p) + (1-\alpha)n h(p) - h(\lambda)n - o(n)\right) = \exp_2((h(p) - h(\lambda) - o(1))n).$$

Thus, by linearity of expectations, the expected size of $\mathcal{L}$ is at least $|C| \cdot \exp_2((h(p) - h(\lambda) - o(1))n)$. On the other hand, the (p, L) list-decodability of C says that $|\mathcal{L}| \leqslant L$ (with probability 1). Comparing these lower and upper bounds on $\mathbf{E}\,|\mathcal{L}|$ yields the claim. □

Proof of Theorem 2 (part 2): At a high level, we proceed as in the proof of Lemma 3, but in addition to the bad list $\mathcal{L}$ of codewords, we will a special codeword $c^* \in C$ such that $d(x, c^*)$ is *much smaller than the codewords in* $\mathcal{L}$. Then defining $\mathcal{L}^*$ to consist of c^* and $(L-1)$ other codewords from $\mathcal{L}$, we see that the *average* distance of $\mathcal{L}^*$ is much smaller than before, thus enabling us to obtain an improved rate bound.

We now provide the details. Pick a uniformly random codeword $c^* \in C$. Let $S \subseteq [n]$ be a random subset of $\mathrm{Supp}(c^*)$ of size βn, where the parameter β is chosen appropriately later[1] (this plays the role of α in Lemma 3). Also, let x be the indicator vector of S.

As before, consider the set $\mathcal{L}$ of codewords $c \in C$ such that $\mathrm{wt}(c|_S) \geqslant (1-p)|S|$. For a fixed $c \in C$, the random variable $\mathrm{wt}(c|_S)$ follows the hypergeometric distribution with parameters $(\lambda n, (\lambda - \delta)n, \beta n)$, where $\delta = \delta(c^*, c)$ is defined by $d(c^*, c) := 2\delta n$. (Observe that the normalization ensures that $0 \leqslant \delta \leqslant \lambda$ for all pairs $c^*, c \in C$.) To see this, notice that we are sampling βn coordinates from $\mathrm{Supp}(c^*)$ without replacement, and that $\mathrm{wt}(c|_S)$ simply counts the number of coordinates picked from $\mathrm{Supp}(c^*) \cap \mathrm{Supp}(c)$ (the size of this intersection is exactly $(\lambda - \delta)n$). Thus, conditioned on c^*, the probability that a fixed $c \in C$ is

[1] The reader might find it helpful to think of β as $O(1/L)$; roughly speaking, this translates to a rate upper bound of $h(\lambda) - h(p) - \Omega(\beta/L)$.

included in $\mathcal{L}$ is

$$Q(\delta) := \sum_{w=(1-p)\beta n}^{\beta n} f(\lambda n, (\lambda-\delta)n, \beta n, w). \tag{1}$$

By linearity of expectations, and taking expectations over c^*, the expected size of $\mathcal{L}$ can be written as $\mathbf{E}_{c^* \in C}\left[\sum_{c \in C} Q(\delta(c^*, c))\right] = |C| \cdot \mathbf{E}\ Q(\delta(c^*, c))$, where *both* c^* and c are picked uniformly at random from C. The bulk of the work lies in obtaining a lower bound on this expectation, which we state below.

Claim. For $A_1 := (1-p)\log\left(\frac{1-p}{\lambda}\right) + p\log\left(\frac{p}{1-\lambda}\right)$ and $A_2 = \frac{5}{p^2}$, we have

$$\mathbf{E}\ Q(\delta(c^*, c)) \geqslant \exp_2\left(-(A_1\beta + A_2\beta^2 + o(1))n\right).$$

Proof Sketch: A standard application of the Cauchy-Schwarz inequality shows that $\mathbf{E}\ \delta \leqslant \lambda(1-\lambda)$, and hence Markov's inequality implies that

$$\delta \leqslant \lambda(1-\lambda) + \frac{1}{n} = \lambda(1-\lambda) + o(1)$$

with probability at least $1/n$. Moreover, since $Q(\delta)$ is a monotone decreasing function of δ, we have

$$\mathbf{E}\ Q(\delta) \geqslant \frac{1}{n} \cdot Q(\lambda(1-\lambda) + o(1)).$$

The rest of the proof is technical and involves lower bounding the right hand side using properties of binomial coefficients and Taylor approximations for the binary entropy function. Due to lack of space, we skip the detailed calculations, which can be found in the full version. □

Therefore, as before, if the code C has rate $A_1\beta + A_2\beta^2 + o(1)$ (for a suitable $o(1)$ term), the list $\mathcal{L}$ has size at least L in expectation. Fix some choice of c^* and S such that $|\mathcal{L}| \geqslant L$. Let $\mathcal{L}^*$ be any list containing c^* and $L-1$ other codewords from $\mathcal{L}$; we are interested in $D_{\mathsf{avg}}(x, \mathcal{L}^*)$. Clearly, $d(x, c^*) = (\lambda - \beta)n$. On the other hand, for $c \in \mathcal{L}^* \setminus \{c^*\}$, we can bound its distance from x as: $d(x, c) \leqslant \beta pn + (\lambda - \beta(1-p))n = (\lambda - \beta(1-2p))n$, where the two terms are respectively the contribution by S and $[n] \setminus S$. Averaging these L distances, we get that

$$D_{\mathsf{avg}}(x, \mathcal{L}^*) \leqslant (\lambda - \beta(1 - 2p + 2p/L))\, n.$$

Now, we pick β so that this expression is at most pn; i.e., set

$$\beta := \frac{\lambda - p}{1 - 2p + 2p/L}. \tag{2}$$

(Compare with the choice of α in Lemma 3.) For this choice of β, the list $\mathcal{L}^*$ violates the average-radius list-decodability property of C.

Thus the rate of a (p, L) average-radius list-decodable code is upper bounded by $R \leqslant A_1\beta + A_2\beta^2 + o(1)$, where β is given by (2). Further technical manipulations brings this to the following more convenient form: If $L > \frac{2p}{1-2p}$, then

$$R \leqslant (h(\lambda) - h(p)) - \frac{B_1(\lambda - p)}{L} + B_2(\lambda - p)^2 + o(1).$$

for some constants B_1 and B_2 depending only on p. (See the full version for a detailed calculation.) Setting $\lambda := p + B_1/(2B_2L)$, the rate is upper bounded by $R \leqslant h(\lambda) - h(p) - B_1^2/(4B_2L^2) + o(1)$. □

4 Bounds for (standard) list-decodability

In this section, we consider the rate vs. list-size tradeoff for the traditional list-decodability notion. For the special case when the fraction of errors is close to $\frac{1}{2}$, [12] showed that any code family of growing size correcting up to $\frac{1}{2} - \varepsilon$ fraction of errors must have a list-size $\Omega(1/\varepsilon^2)$, which is optimal up to constant factors. When p is bounded away from $1/2$, Blinovsky [1, 3] gives the best known bounds on the rate of a (p, L) list-decodable code. His results imply (see [14] for the calculations) that any (p, L) list-decodable code of rate $1 - h(p) - \gamma$ has list-size L at least $\Omega_p(\log(1/\gamma))$. We give a short and simple proof of this latter claim in this section.

Theorem 3 ([1, 3]).

1. *Suppose C is $(\lambda; p, L)$ list-decodable code with $\lambda = p + \frac{1}{2}p^L$. Then $|C| \leqslant 2L^2/p$, independent of its blocklength n. (In particular, the rate approaches 0 as $n \to \infty$.)*
2. *Any (p, L) list-decodable code has rate at most $1 - h(p) - \Omega_p(p^L)$.*

Proof: For the first part, assume for the sake of contradiction that $|C| > 2L^2/p$. Pick a random L-tuple of codewords (without replacement) $\mathcal{L} = \{c_1, c_2, \ldots, c_L\}$, and let S be the set of indices $i \in [n]$ such that each $c_j \in \mathcal{L}$ has 1 in the ith coordinate. Define x to be the indicator vector of S. Note that $d(x, c_j) = \lambda n - \mathrm{wt}(x) = \lambda n - |S|$, so that $\mathbf{E}\ D_{\mathsf{max}}(x, \mathcal{L}) = \lambda n - \mathbf{E}\ |S|$. Thus to obtain a contradiction, it suffices to show that $\mathbf{E}\ |S| \geqslant \lambda - p = \frac{1}{2}p^L$.

Let $M := |C|$ be the total number of codewords in C, and let M_i be the number of codewords of C with 1 in the i^{th} position. Then the probability that $i \in S$ is equal to $g(M_i)/\binom{M}{L}$, where the function $g : \mathbb{R}^{\geqslant 0} \to \mathbb{R}^{\geqslant 0}$ is defined by $g(z) := \binom{\max\{z, L-1\}}{L}$. By standard closure properties of convex functions, g is convex on $\mathbb{R}$. (Specifically, $z \mapsto \max\{z, L-1\}$ is convex over $\mathbb{R}$, and restricted to its image (i.e., the interval $[L-1, \infty)$), the function $z \mapsto \binom{z}{L}$ is convex. Hence their composition, namely g, is convex as well.)

We are now ready to bound $\mathbf{E}\ |S|$:

$$\frac{1}{n}\mathbf{E}\ |S| \overset{(a)}{=} \frac{1}{\binom{M}{L}} \cdot \frac{1}{n}\sum_{i=1}^{n} g(M_i) \overset{(b)}{\geqslant} \frac{1}{\binom{M}{L}} \cdot g\left(\frac{1}{n}\sum_{i=1}^{n} M_i\right) = \frac{g(\lambda M)}{\binom{M}{L}} \overset{(c)}{=} \frac{\binom{\lambda M}{L}}{\binom{M}{L}}.$$

Here we have used (a) the linearity of expectations, (b) Jensen's inequality, and (c) the fact that $\lambda M \geqslant 2L^2 \geqslant L - 1$. We complete the proof using a straightforward approximation of the binomial coefficients.

$$\mathbf{E}\,|S| \geqslant \frac{(\lambda M - L)^L}{M^L} = \lambda^L \left(1 - \frac{L}{\lambda M}\right)^L \geqslant \lambda^L \left(1 - \frac{L^2}{\lambda M}\right) \geqslant \frac{1}{2}\lambda^L \geqslant \frac{1}{2}p^L.$$

For the second part, by Lemma 2, the rate of a *general* (p, L) list-decodable code is upper bounded by $1 - h\left(p + \frac{1}{2}p^L\right) + o(1)$, which can be shown to be at most $1 - h(p) - \frac{1}{4}(1 - 2p) \cdot p^L + o(1)$. □

The above method can be adapted for q-ary codes with an additional trick:

Theorem 4. *1. Suppose C is a q-ary $(\lambda; p, L)$ list-decodable code with $\lambda = p + \frac{1}{2L}p^L$. Then $|C| \leqslant 2L^2/\lambda$.*
2. Suppose C is a q-ary (p, L) list-decodable code. Then there exists a constant $b = b_{q,p} > 0$ such that the rate of C is at most $1 - h_q(p) - 2^{-bL}$.

Before we provide a proof of this theorem, we will state a convenient lemma due to Erdös. (See Section 2.1 of [13] for reference.) This result was implicitly established in our proof of Theorem 3, so we will omit a formal proof.

Lemma 4 (Erdös 1964). *Suppose $\mathcal{A}$ is a set system over the ground set $[n]$, such that each $A \in \mathcal{A}$ has size at least λn. Then if $|\mathcal{A}| \geqslant 2L^2/\lambda$, then there exist distinct $A_1, A_2, \ldots, A_L$ in $\mathcal{A}$ such that $\bigcap_{i=1}^{L} A_i$ has size at least $\frac{1}{2}n\lambda^L$.*

Proof of Theorem 4: As in Theorem 3, the second part follows from the first. To prove the first claim, assume towards a contradiction that $|C| > 2L^2/\lambda$. Consider the family of sets $\mathcal{A} := \{\mathrm{Supp}(c) : c \in C\}$. By Lemma 4, there exists an L-tuple $\{c_1, c_2, \ldots, c_L\}$ of codewords such that the intersection of their support, say S, has size at least $\frac{1}{2}n\lambda^L \geqslant \frac{1}{2}np^L$. Arbitrarily partition the coordinates in S into L parts, say $S_1, \ldots, S_L$, of almost equal size $\frac{1}{2L}p^L \cdot n$.

Now consider the center x such that x agrees with c_j on all coordinates $i \in S_j$. For $i \notin S$, set x_i to be zero. Then, clearly, $d(x, c_j) \leqslant \lambda n - \frac{1}{2L}p^L \cdot n = pn$. Thus the list $\{c_1, \ldots, c_L\}$ contradicts the (p, L) list-decodability of C. □

5 Constant-weight vs. General codes

In this section, we will understand the rate vs. list-size trade-offs for constant-weight codes, that is, codes with every codeword having weight λn, where $\lambda \in (p, \frac{1}{2}]$ is a parameter. (Setting $\lambda = \frac{1}{2}$ roughly corresponds to arbitrary codes having no weight restrictions.) As observed earlier, a typical approach in coding theory to establish rate upper bounds is to study the problem under the above constant-weight restriction. One then proceeds to show a strong negative result of the flavor that a code with the stated properties must have a constant size (and in particular *zero* rate). For instance, the first part of Theorem 3 above is of this form. Finally, mapping this bound to arbitrary codes, one obtains a rate upper bound of $1 - h(\lambda)$ for the original problem. (Note that Lemma 2 provides a particular formal example of the last step.)

In particular, Blinovsky's rate upper bound (Theorem 3) of $1-h(p)-2^{-O(L)}$ for (p, L) list-decodable codes follows this approach. (For notational ease, we suppress the dependence on p in the O and Ω notations in this informal discussion.) More precisely, he proves that, under the weight-λ restriction, such code must have zero rate for all $\lambda \leqslant p + 2^{-b_p L}$ for some $b_p < \infty$. One may then imagine improving the rate upper bound to $1 - h(p) - L^{-O(1)}$ *simply by* establishing the latter result for correspondingly higher values of λ (i.e., up to $p + L^{-O(1)}$). We show that this approach cannot work by establishing that (average-radius) list-decodable codes of positive (though possibly small) rates exist as long as $\lambda - p \geqslant 2^{-O(L)}$. Thus Blinovsky's result identifies the correct *zero-rate regime* for the list-decoding problem; in particular, his bound is also the best possible if we restrict ourselves to this approach. In this context, it is also worth noting that for average-radius list-decodable codes, Theorem 2 already provides a better rate upper bound than what the zero-rate regime suggests, thus indicating that the 'zero-rate regime barrier' is not an inherent obstacle, but more a limitation of the current proof techniques.

In the opposite direction, we show that the task of establishing rate upper bounds for constant weight codes is not significantly harder than the general problem. Formally, we state that that if the "gap to list-decoding capacity" for general codes is γ, then the gap to capacity for weight-λn codes is *at least* $\left(\frac{\lambda-p}{\frac{1}{2}-p}\right)\gamma$. Stated differently, if our goal is to establish a $L^{-O(1)}$ lower bound on the gap γ, then we do not lose by first passing to a suitable λ (that is not too close to p).

5.1 Zero-rate regime

Theorem 5. *Fix $p \in (0, \frac{1}{2})$, and set $b = b_p := \frac{1}{2}\left(\frac{1}{2} - p\right)^2$. Then for all sufficiently large L, there exists a $(\lambda; p, L)$ average-radius list-decodable code of rate at least $R - o(1)$, with $p \leqslant \lambda \leqslant p + 5e^{-bL}$ and $R := \min\{e^{-2bL}, e^{-bL}/(6L)\} = \Omega_{p,L}(1)$.*

Proof Sketch: We only provide a sketch of the proof here; see the full version for the complete proof. We obtain this result by random coding followed by expurgation. Set $\varepsilon := e^{-bL}$ and $\lambda' := p + 4\varepsilon$. Consider a random code C of size 2^{Rn} such that each coordinate of each codeword in C is independently set to 1 with probability λ' and to 0 with probability $1 - \lambda'$. For our choice of parameters, we can show that w.h.p., C satisfies the following properties: (a) C is (p, L) average-radius list-decodable; and (b) every codeword in C has weight in the range $(\lambda' \pm \varepsilon)n$. In particular, the maximum weight of any codeword is at most $(p + 5\varepsilon)n$.

Now, pick any C satisfying these two properties, and let C_w denote the subcode of C consisting of the weight-w codewords. Then, if we define $w_0 = \lambda n$ to be the "most popular" weight, then the code C_{w_0} satisfies all our requirements. Note that the final step incurs only a $o(1)$ loss in the rate, since by the pigeonhole principle, the resulting code has size at least $2^{Rn}/(n+1) = \exp_2(Rn - o(n))$. □

5.2 A reverse connection between constant-weight and arbitrary codes

Lemma 5. *Fix p, λ such that $0 < p < \lambda < \frac{1}{2}$. Then in the notation of Definition 3, if $\gamma := 1 - h(p) - R_{p,L}$, then*

$$h(\lambda) - h(p) - \gamma \leqslant R_{p,L}(\lambda) \leqslant h(\lambda) - h(p) - \left(\frac{\lambda - p}{\frac{1}{2} - p}\right)\gamma.$$

Proof: The left inequality is essentially the content of Lemma 2; we show the second inequality here. The manipulations in this proof are of a similar flavor to those in Lemma 3, but the exact details are different.

Suppose C is a $(\lambda; p, L)$ list-decodable code of blocklength n and rate R, such that each codeword in C has weight exactly λn. Pick a random subset $S \subseteq [n]$ of coordinates of size $\alpha_2 n$, with $\alpha_2 := (\lambda - p)/(\frac{1}{2} - p)$, and let $\overline{S} := [n] \setminus S$. (Interestingly, our setting of α_2 differs from the parameter α employed in the proof of Lemma 3 only by a factor of 2. The motivation for this choice of α_2 will become clear shortly.) Consider the subcode C' consisting of codewords $c \in C$ such that $\mathrm{wt}(c|_S) \geqslant \alpha_2 n/2$. For our choice of α_2, one can verify that if $c \in C'$, then c has weight at most $p(1-\alpha_2)n = p|\overline{S}|$ when restricted to $\overline{S}$.

Our key insight now is that the code $C'|_S := \{c|_S \ : \ c \in C'\}$ (of blocklength $\alpha_2 n$) is (p, L) list-decodable. Suppose not. Then there exists a center $x' \in \{0,1\}^S$ and a size-L list $\mathcal{L} \subseteq C$ such that $d(x', c|_S) \leqslant p\alpha_2 n$ for all $c \in \mathcal{L}$. Now, extend x' to $x \in \{0,1\}^n$ such that x agrees with x' on (the coordinates in) S and is zero on the remaining coordinates. Then $\mathcal{L}$ violates the (p, L) list-decodability of C, since for every $c \in \mathcal{L}$,

$$d(x, c) = d(x', c|_S) + \mathrm{wt}(c|_{\overline{S}}) \leqslant p\alpha_2 n + p(1-\alpha_2)n = pn.$$

Hence $C'|_S$ must be (p, L) list-decodable as well. For a fixed $c \in C$, the random variable $\mathrm{wt}(c|_S)$ follows the hypergeometric distribution with parameters $(n, \lambda n, \alpha_2 n)$, which is identical to the hypergeometric distribution with parameters $(n, \alpha_2 n, \lambda n)$. Hence, the probability that c is included in C' is at least

$$\begin{aligned} f(n, \alpha_2 n, \lambda n, \alpha_2 n/2) &= \frac{\binom{\alpha_2 n}{\alpha_2 n/2}\binom{(1-\alpha_2)n}{(\lambda - \alpha_2/2)n}}{\binom{n}{\lambda n}} \\ &\overset{(*)}{=} \frac{\binom{\alpha_2 n}{\alpha_2 n/2}\binom{(1-\alpha_2)n}{p(1-\alpha_2)n}}{\binom{n}{\lambda n}} \\ &\geqslant \exp_2\left(\alpha_2 n + h(p)(1-\alpha_2)n - h(\lambda)n - o(n)\right). \end{aligned}$$

In the step marked $(*)$, we have used the the identity $\lambda - \alpha_2/2 = p(1-\alpha_2)$, which holds for our particular choice of α_2. Thus, summing this over all $c \in C$, the expected size of $C'|_S$ is at least

$$\exp_2\left(Rn + \alpha_2 n + h(p)(1-\alpha_2)n - h(\lambda)n - o(n)\right).$$

On the other hand, since $C'|_S$ is (p, L) list-decodable, the hypothesis of the lemma implies that its size is at most $\exp_2((1 - h(p) - \gamma)\alpha_2 n)$ with probability

1. (It is crucial for our purposes that the blocklength of C' is $\alpha_2 n$, which is significantly smaller than n.) Comparing the upper and lower bound on the expected size of $C'|_S$, we get $R + \alpha_2 + (1-\alpha_2)h(p) - h(\lambda) \leqslant (1 - h(p) - \gamma)\alpha_2$, which can be rearranged to give the desired bound $R \leqslant h(\lambda) - h(p) - \alpha_2\gamma$. □

References

1. V. M. Blinovsky. Bounds for codes in the case of list decoding of finite volume. *Problems of Information Transmission*, 22(1):7–19, 1986.
2. V. M. Blinovsky. *Asymptotic Combinatorial Coding Theory*. Kluwer Academic Publishers, Boston, 1997.
3. V. M. Blinovsky. Code bounds for multiple packings over a nonbinary finite alphabet. *Problems of Information Transmission*, 41(1):23–32, 2005.
4. V. M. Blinovsky. On the convexity of one coding-theory function. *Problems of Information Transmission*, 44(1):34–39, 2008.
5. M. Cheraghchi and V. Guruswami. Restricted isometry via list decoding. Work in progress, 2012.
6. P. Elias. List decoding for noisy channels. *Technical Report 335, Research Laboratory of Electronics, MIT*, 1957.
7. P. Elias. Error-correcting codes for list decoding. *IEEE Transactions on Information Theory*, 37:5–12, 1991.
8. V. Guruswami. Linear-algebraic list decoding of folded Reed-Solomon codes. In *Proceedings of the 26th IEEE Conference on Computational Complexity*, pages 77–85, June 2011.
9. V. Guruswami, J. Håstad, and S. Kopparty. On the list-decodability of random linear codes. *IEEE Transactions on Information Theory*, 57(2):718–725, 2011.
10. V. Guruswami and A. Rudra. Explicit codes achieving list decoding capacity: Error-correction up to the Singleton bound. *IEEE Transactions on Information Theory*, 54(1):135–150, January 2008.
11. V. Guruswami and M. Sudan. Extensions to the Johnson bound. *Unpublished manuscript*, 2001. Available at http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.145.9405.
12. V. Guruswami and S. P. Vadhan. A lower bound on list size for list decoding. *IEEE Transactions on Information Theory*, 56(11):5681–5688, 2010.
13. S. Jukna. *Extremal Combinatorics: with applications in Computer Science*. Springer, 2001.
14. A. Rudra. Limits to list decoding of random codes. *IEEE Transactions on Information Theory*, 57(3):1398–1408, 2011.
15. J. M. Wozencraft. List Decoding. *Quarterly Progress Report, Research Laboratory of Electronics, MIT*, 48:90–95, 1958.
16. V. V. Zyablov and M. S. Pinsker. List cascade decoding. *Problems of Information Transmission*, 17(4):29–34, 1981 (in Russian); pp. 236-240 (in English), 1982.