When Do Low-Rate Concatenated Codes Approach The Gilbert–Varshamov Bound?

Dean Doron
Ben-Gurion University of the Negev, Beersheba, Israel

Jonathan Mosheiff
Ben-Gurion University of the Negev, Beersheba, Israel

Mary Wootters
Stanford University, CA, USA

Abstract

The *Gilbert–Varshamov* (GV) bound is a classical existential result in coding theory. It implies that a random linear binary code of rate ε^2 has relative distance at least $\frac{1}{2} - O(\varepsilon)$ with high probability. However, it is a major challenge to construct *explicit* codes with similar parameters.

One hope to derandomize the Gilbert–Varshamov construction is with code concatenation: We begin with a (hopefully explicit) outer code $\mathcal{C}_{\text{out}}$ over a large alphabet, and concatenate that with a small binary random linear code $\mathcal{C}_{\text{in}}$. It is known that when we use *independent* small codes for each coordinate, then the result lies on the GV bound with high probability, but this still uses a lot of randomness. In this paper, we consider the question of whether code concatenation with a *single* random linear inner code $\mathcal{C}_{\text{in}}$ can lie on the GV bound; and if so what conditions on $\mathcal{C}_{\text{out}}$ are sufficient for this.

We show that first, there *do* exist linear outer codes $\mathcal{C}_{\text{out}}$ that are "good" for concatenation in this sense (in fact, *most* linear codes codes are good). We also provide two sufficient conditions for $\mathcal{C}_{\text{out}}$, so that if $\mathcal{C}_{\text{out}}$ satisfies these, $\mathcal{C}_{\text{out}} \circ \mathcal{C}_{\text{in}}$ will likely lie on the GV bound. We hope that these conditions may inspire future work towards constructing explicit codes $\mathcal{C}_{\text{out}}$.

2012 ACM Subject Classification Theory of computation → Error-correcting codes

Keywords and phrases Error-correcting codes, Concatenated codes, Derandomization, Gilbert-Varshamov bound

Digital Object Identifier 10.4230/LIPIcs.APPROX/RANDOM.2024.53

Category RANDOM

Related Version *Full Version*: https://arxiv.org/abs/2405.08584

Funding *Dean Doron*: Supported in part by NSF-BSF grant #2022644.
Jonathan Mosheiff: Supported by an Alon Fellowship.
Mary Wootters: Partially supported by NSF grants CCF-2231157 and CCF-2133154.

Acknowledgements We thank Amnon Ta-Shma for helpful and interesting discussions, and collaboration at the beginning of this work. We thank Arya Mazumdar for pointing out [4] and for helping us understand its implications. This work was done partly while the authors were visiting the Simons Institute for the Theory of Computing.

1 Introduction

An *error correcting code* (or just a *code*) is a subset $\mathcal{C} \subseteq \Sigma^n$, for some alphabet Σ. We think of a code $\mathcal{C}$ being used to encode messages in Σ^k for $k = \log_{|\Sigma|} |\mathcal{C}|$. That is, for any $m \in \Sigma^k$, we can identify m with a codeword $\mathcal{C}(m) \in \mathcal{C}$.[1] The idea is that encoding m into the

[1] Here and throughout the paper, we will abuse notation and use $\mathcal{C}$ both as the code itself (a subset of Σ^n) and also as an encoding map $\mathcal{C}\colon \Sigma^k \to \Sigma^n$.

codeword $\mathcal{C}(m)$ will introduce redundancy that can later be used to correct errors. In this work we focus on *linear* codes $\mathcal{C}$, which are codes where $\Sigma = \mathbb{F}$ is a finite field and $\mathcal{C} \subseteq \mathbb{F}^n$ is a linear subspace of $\mathbb{F}^n$.

Two important properties of error correcting codes are the *rate* R and the *relative distance* δ. For a code $\mathcal{C} \subseteq \Sigma^n$, the rate is defined as $R = \frac{\log_{|\Sigma|} |\mathcal{C}|}{n} = \frac{k}{n}$, and it quantifies how large the code is. The rate is between 0 and 1, and typically we want it to be as close to 1 as possible; this means that the encoding map does not introduce much redundancy. The (relative) distance of $\mathcal{C} \subseteq \Sigma^n$ is defined as $\delta = \frac{1}{n} \min_{c \neq c' \in \mathcal{C}} \Delta(c, c')$, where $\Delta(\cdot, \cdot)$ is Hamming distance. Again, the relative distance is between 0 and 1, and again we typically want it to be as close to 1 as possible; this means that the code can correct many worst-case errors.

These two quantities – rate and distance – are in tension. The larger the rate is, the smaller the distance must be. For binary codes (that is, codes where $\Sigma = \mathbb{F}_2$), it is a major open question to pin down the best trade-off possible between rate and distance. However, we know that good trade-offs are possible: The best known possibility result in general is the *Gilbert–Varshamov* (GV) bound (Theorem 2.1 in the full version).

In this paper we focus on *low rate* codes. In this parameter regime, the GV bound implies that there *exist* binary linear codes with relative distance $\frac{1-\varepsilon}{2}$ and rate $\Omega(\varepsilon^2)$, for small $\varepsilon > 0$. In fact, Varshamov's proof shows that a random binary linear code achieves this with high probability.

Constructing such codes explicitly, hopefully accompanied by an efficient decoding algorithm, has been subject to extensive and fruitful research in the past decades (e.g., [24, 2, 3, 6, 11, 28, 7]), with several exciting breakthroughs in recent years. These breakthroughs include explicit constructions of codes with distance $\delta = \frac{1-\varepsilon}{2}$ and rate $R = \Omega(\varepsilon^{2+o(1)})$, even with efficient algorithms (see Section 1.1). However, there are still open questions. For example, we do not know how to attain $\delta = \frac{1-\varepsilon}{2}$ and $R = \Omega(\varepsilon^2)$ (without any $o(1)$ term) explicitly, and we do not have explicit constructions approaching the GV bound with rates bounded away from zero. Motivated by these questions, we consider *concatenated codes*, possibly with some randomness, which we discuss next.

Concatenated Codes, and Our Question

A natural candidate for explicit (for low randomness) codes on the GV bound are *concatenated linear codes.* These codes are built out of two ingredients: a (hopefully explicit) linear outer code $\mathcal{C}_{\text{out}} \subseteq \mathbb{F}_q^n$ with dimension k for some large q; and a smaller inner binary linear code $\mathcal{C}_{\text{in}} \subseteq \mathbb{F}_2^{n_0}$, with dimension $k_0 = \log_2 q$. We define the concatenated code $\mathcal{C} = \mathcal{C}_{\text{out}} \circ \mathcal{C}_{\text{in}} \subseteq \mathbb{F}_2^{n_0 \cdot n}$ by first encoding a message $m \in \mathbb{F}_q^k$ (which can also be thought of as $m \in \mathbb{F}_2^{k_0 \cdot k}$) with $\mathcal{C}_{\text{out}}$. Then, we encode each symbol of the resulting codeword using $\mathcal{C}_{\text{in}}$. That is, for a message m,

$$\mathcal{C}(m) = (\mathcal{C}_{\text{in}}(\mathcal{C}_{\text{out}}(m)_1), \mathcal{C}_{\text{in}}(\mathcal{C}_{\text{out}}(m)_2), \cdots, \mathcal{C}_{\text{in}}(\mathcal{C}_{\text{out}}(m)_n)) \in \mathbb{F}_2^{n_0 \cdot n}.$$

It is not hard to see that the rate of $\mathcal{C}$ is the product of the rates of $\mathcal{C}_{\text{in}}$ and $\mathcal{C}_{\text{out}}$, and that the distance of $\mathcal{C}$ is *at least* the product of the distances of $\mathcal{C}_{\text{in}}$ and $\mathcal{C}_{\text{out}}$.

The natural approach to constructing a good concatenated code is to choose $\mathcal{C}_{\text{out}}$ and $\mathcal{C}_{\text{in}}$ with the best known trade-offs: Since $\mathcal{C}_{\text{out}}$ is over a large alphabet, we know explicit constructions of codes with optimal rate-distance trade-off[2]; and if n_0 is sufficiently small, we can find a $\mathcal{C}_{\text{in}}$ on the GV bound either deterministically by brute force or else with low randomness, depending on the size of n_0.

[2] For codes over large alphabets, the best possible trade-off is the *Singleton bound*, or $R = 1 - \delta$. This is achievable, for example, by Reed–Solomon codes.

However, in general this approach will not achieve the GV bound. If we do not assume any additional properties of $\mathcal{C}_{\text{out}}$ and $\mathcal{C}_{\text{in}}$, and simply use the concatenation properties, then setting the parameters so that $\mathcal{C} = \mathcal{C}_{\text{out}} \circ \mathcal{C}_{\text{in}}$ has distance $\frac{1-\varepsilon}{2}$, the rate of $\mathcal{C}$ will be at most roughly ε^3. This is known as the *Zyablov bound* [31] (see also [14]). As we discuss more in Section 1.1, concatenation has been a popular approach to obtain fully explicit codes with good rate-distance trade-offs, but none of these constructions are known to beat the Zyablov bound.

Instead of using a *single* inner code, several works have focused on a related construction originally due to Thommesen [29], which uses multiple inner codes. More precisely, this construction uses i.i.d. random linear inner codes for each coordinate. It can be shown [29] that the resulting code does lie on the GV bound with high probability, and if $\mathcal{C}_{\text{out}}$ is chosen appropriately there are even efficient decoding algorithms for it [10, 27, 15]. However, this approach relies heavily on the fact that the inner codes are independent, and as a result uses a lot of randomness.

This state of affairs motivates the following question (also asked in the title of this paper):

▶ **Question 1.** *Are there concatenated linear codes $\mathcal{C}_{\text{out}} \circ \mathcal{C}_{\text{in}}$ (with a* single *random linear inner code $\mathcal{C}_{\text{in}}$) that meet the GV bound with high probability over $\mathcal{C}_{\text{in}}$?*[3] *If so, are there sufficient conditions on $\mathcal{C}_{\text{out}}$ that will guarantee this?*

In this paper, we show that *yes*, there are concatenated codes that meet the GV bound, and we also give two sufficient conditions on $\mathcal{C}_{\text{out}}$ for this to hold. Our existential result is non-constructive, but it is our hope that our sufficient conditions will lead to explicit constructions of appropriate $\mathcal{C}_{\text{out}}$-s, which would lead to explicit (or at least pseudo-random, depending on the alphabet size of $\mathcal{C}_{\text{out}}$) concatenated codes on the GV bound.

▶ Remark 1 (Motivation for Question 1). Above, we have motivated Question 1 as an avenue towards explicit or pseudo-random binary codes on the GV bound, and indeed this is our original motivation. But we point out that Question 1 is also interesting in its own right. Concatenated codes are a classical construction, going back to the 1960's [9], and have been used in many different settings over the decades. It seems like a fundamental question to understand when these codes can attain the GV bound.

▶ Remark 2 (Focus on Linear Codes). In Question 1 and in this paper, we focus on *linear* codes. This is because if we used, say, a uniformly random non-linear code as the inner code, it would require exponentially more randomness than a random linear inner code, so this does not seem like a hopeful avenue for derandomization. We note however that the question is much easier for non-linear codes. For example, suppose that $\mathcal{C}_{\text{out}}$ is a Reed–Solomon code of rate ε so that each symbol is additionally tagged with its evaluation point: that is, the symbol corresponding to $\alpha \in \mathbb{F}_q$ is $(\alpha, f(\alpha)) \in \mathbb{F}_q^2$. For the inner code, we use a completely random (non-linear) code of rate ε. Then since all of the symbols in each outer codeword are different by construction, each codeword is essentially uniformly random, and it is not hard to show that the result is close to the GV bound in the sense that a code of rate $O(\varepsilon^2)$ will have distance $1/2 - O(\varepsilon)$ with high probability. This same argument will not work when $\mathcal{C}_{\text{in}}$ is linear, since the different symbols of codewords of $\mathcal{C}_{\text{out}}$ will still have $\mathbb{F}_2$-linear relationships.

[3] Of course, if the length of either the inner code or the outer code is 1, this question reduces to the non-concatenated setting; we are interested in parameter regimes where n_0 is non-trivial.

Our Contributions

Our main results are:

1. **Existence of concatenated codes on the GV bound.** We answer the first part of Question 1: there *are* concatenated codes $\mathcal{C}_{\text{out}} \circ \mathcal{C}_{\text{in}}$ that achieve the GV bound, in a wide variety of parameter regimes. In particular, we show that *most* codes $\mathcal{C}_{\text{out}}$ are actually good:

   ▶ **Theorem 3** (Informal; Theorem 4.2 in the full version)**.** *Suppose that $\mathcal{C}_{\text{out}} \subseteq \mathbb{F}_q^n$ and $\mathcal{C}_{\text{in}} \subseteq \mathbb{F}_2^{n_0}$ are random linear codes of rate ε, so that $q \geq 2^{\Omega(\varepsilon^{-3})}$. Then $\mathcal{C} = \mathcal{C}_{\text{out}} \circ \mathcal{C}_{\text{in}}$ has rate ε^2, and with high probability, the relative distance of $\mathcal{C}$ is at least $1/2 - O(\varepsilon)$.*

   While Theorem 3 seems intuitive (in the sense that a random linear code lies on the GV bound with high probability, so why not concatenated random linear codes?), to the best of our knowledge it has not appeared in the literature before, and the proof was not obvious (to us).[4] One challenge is that a codeword $c \in \mathcal{C}_{\text{out}} \circ \mathcal{C}_{\text{in}}$ is not uniformly random in $\mathbb{F}_2^N$. In particular, the natural strategy of "show that each non-zero codeword has high weight with high probability and union bound" that is used to establish the Gilbert–Varshamov bound will not work in this setting, as we do not have enough concentration.

2. **Sufficient conditions for $\mathcal{C}_{\text{out}}$.** Our existence result above uses a random linear code as the outer code, which does not help in the quest for explicit constructions. However, our proof techniques inspire two sufficient conditions on $\mathcal{C}_{\text{out}}$. That is, if $\mathcal{C}_{\text{out}}$ satisfies these conditions, then $\mathcal{C}_{\text{out}} \circ \mathcal{C}_{\text{in}}$ will meet the GV bound with high probability when $\mathcal{C}_{\text{in}}$ is a random linear code. Our hope is that formalizing these will lead to explicit constructions in the future.

   We give an overview and intuition for our two sufficient conditions here. We note that both conditions are only sufficient when the alphabet size q for $\mathcal{C}_{\text{out}}$ is suitably large (exponential in $1/\operatorname{poly}(\varepsilon)$); see Theorems 5.1 and 6.2 in the full version for details.

   - **Sufficient Condition 1: A soft-decoding-like condition on $\mathcal{C}_{\text{out}}^{\perp}$.** Our first sufficient condition, formalized in Theorem 5.1 in the full version, is a soft-list-decoding-like condition on $\mathcal{C}_{\text{out}}^{\perp}$. More precisely, we define a distribution $\mathcal{D}$[5] on the alphabet $\mathbb{F}_q$; the condition is that

     $$\Pr_{x \sim \mathcal{D}^n}[x \in \mathcal{C}_{\text{out}}^{\perp} \setminus \{0\}] \leq \frac{1}{q^k}(1 + \Delta) \tag{1}$$

     for some small Δ. Note that $1/q^k$ is the probability that a completely random vector is in $\mathcal{C}_{\text{out}}^{\perp}$, so this condition is saying that if the coordinates of x are drawn i.i.d. from the same distribution $\mathcal{D}$, then x not much more likely to be in $\mathcal{C}^{\perp}$ than in a uniformly random vector. We show that if this holds, then $\mathcal{C}_{\text{out}} \circ \mathcal{C}_{\text{in}}$ lies on the GV bound with high probability over the choice of a random linear inner code $\mathcal{C}_{\text{in}}$.

     It's not hard to see (Remark 8 in the full version) that this condition holds in expectation for a random linear code $\mathcal{C}_{\text{out}}$, and in particular there exist linear codes $\mathcal{C}_{\text{out}}$ that have this property.

---

[4] We note that earlier work by Barg, Justesen and Thomessen [4] also addresses random linear outer codes concatenated with an arbitrary (fixed) inner code, using very different techniques than we do. They do not explicitly state a statement like Theorem 3 above, though it is plausible that their techniques could be used to prove something similar. We discuss their techniques and the relationship to our work in Section 1.1.

[5] The distribution $\mathcal{D}$ is intuitively defined as follows. Let $\mathcal{C}_{\text{in}}$ be the inner code, and suppose that it has a generator matrix $G_0 \in \mathbb{F}_2^{n_0 \times k_0}$. Then to sample from $\mathcal{D}$, we take a random sparse linear combination of the rows of G_0 (over $\mathbb{F}_2$), and interpret the result in $\mathbb{F}_2^{k_0}$ as an element of $\mathbb{F}_q$, which we return.

This condition is reminiscent of $\mathcal{C}_{\text{out}}^{\perp}$ being list-decodable from soft information (e.g., [20]). In soft-list-decoding, one typically gets a distribution $\mathcal{D}_i$ for each $i \in [n]$, interpreted as giving "soft information" about the i'th symbol. If one can show that a vector drawn from $\mathcal{D}_1 \times \cdots \times \mathcal{D}_n$ is unlikely to be in the code, this implies that there are not too many codewords that are likely given the soft information we hare received. However, there are several differences between existing work on soft list-decoding and our work, notably that our distribution $\mathcal{D}$ is a particular one and is the same for all i, and also there are some differences in the parameter settings.

This condition can also be seen as a soft form of *list-recovery*, where we have the same list in each coordinate.[6] In more detail, if the support of $\mathcal{D}$ is concentrated on a small set S (which ours is for reasonable settings of n_0, ε, see Remark 7 in the full version), then the condition in Theorem 5.1 is related to asking that the number of codewords that lie in the combinatorial rectangle given by $S \times S \times \cdots \times S$ is about what it should be. Unfortunately, the definition of "small" here does not seem to be small enough for existing constructions of list-recoverable codes (for example folded RS codes or multiplicity codes) to yield any results.

- **Sufficient Condition 2: $\mathcal{C}_{\text{out}}$ has good min-entropy.** Our second sufficient condition, formalized in Theorem 6.2, requires the codewords of $\mathcal{C}_{\text{out}}$ to be "smooth", meaning, roughly, that every nonzero codeword has a fairly uniform distribution of symbols from $\mathbb{F}_q$. To illustrate why a smoothness condition is desirable, let us consider two extreme cases.

  The bad extreme is when there exists a codeword c that is supported on very few symbols, say even on a single symbol. If $c = (\sigma, \sigma, \ldots, \sigma)$ for some $\sigma \in \mathbb{F}_q$, then the relative weight of $c \circ \mathcal{C}_{\text{in}}$, for a random binary inner code $\mathcal{C}_{\text{in}}$ of rate ε, might be $\frac{1}{2} - \Omega(\sqrt{\varepsilon})$, much worse than the $\frac{1}{2} - O(\varepsilon)$ that we would want for the GV bound.

  The good (possibly unrealistic) extreme is where each nonzero codeword of $\mathcal{C}_{\text{out}}$ has a symbol distribution that is *uniform* over $\mathbb{F}_q$. In this case it is not hard to see that $\mathcal{C}_{\text{out}} \circ \mathcal{C}_{\text{in}}$ will be close to the GV bound with high probability over a random linear code $\mathcal{C}_{\text{in}}$. (For this, all we need is that $\mathcal{C}_{\text{in}}$ has about the "right" weight distribution, which a random linear code will have with high probability).

  The natural question is thus *how smooth* the codewords of $\mathcal{C}_{\text{out}}$ should be in order for $\mathcal{C}$ to have distance $\frac{1}{2} - O(\varepsilon)$. In Section 6 in the full version, we quantify this by the *smooth min-entropy* of the codewords' empirical distributions on symbols. We show in Theorem 6.2 that if this smooth min-entropy is large enough for all $c \in \mathcal{C}_{\text{out}}$, then $\mathcal{C} = \mathcal{C}_{\text{out}} \circ \mathcal{C}_{\text{in}}$ is likely to lie near the GV bound when $\mathcal{C}_{\text{in}}$ is a random linear binary code.

  How large is "large enough"? For this informal discussion, we give one example of the parameter settings from Theorem 6.2: It is enough for every non-zero codeword $c \in \mathcal{C}_{\text{out}}$ to have a symbol distribution that has $\Theta(\varepsilon n)$ copies of the same symbol (say, the zero symbol), while the remaining symbols in c are uniformly distributed over a set of size only $q^{1-\varepsilon}$. By some metrics this is still a fairly "spiky" distribution, but it is "smooth enough" for our purposes.

  Note that while our soft-decoding-like condition considers $\mathcal{C}_{\text{out}}^{\perp}$, our smooth min-entropy condition here considers $\mathcal{C}_{\text{out}}$ itself.

[6] Informally, a code $\mathcal{C} \subseteq \Sigma^n$ is said to be list-recoverable if for any small sets $S_1, \ldots, S_n \subseteq \Sigma$, there are not too many codewords $c \in \mathcal{C}$ so that $c_i \in S_i$ for many values of i.

1.1 Related Work

Explicit Concatenated Codes

Concatenation (with a single inner code) has been a common approach to obtain explicit codes close to the GV bound. Here we mention a few such places this comes up. Choosing $\mathcal{C}_{\text{out}}$ to be the Reed–Solomon code, and $\mathcal{C}_{\text{in}}$ to be the Hadamard code, gets a code of length $O(k^2/\varepsilon^2)$ for any dimension k [3], and replacing Reed–Solomon with the Hermitian code gets length $O((k/\varepsilon)^{5/4})$ [6]. Choosing a different AG code for $\mathcal{C}_{\text{out}}$ can result in non-vanishing rate and in fact approach rate ε^3 (see [28]). Moreover, concatenating Reed–Solomon with the Wozencraft ensemble gives the *Justesen* code [19], having constant relative rate and constant relative distance. Note that none of these concatenation-based constructions thus far have beat the Zyablov bound.

Concatenated Codes with Random Linear $\mathcal{C}_{\text{out}}$

Relevant to Theorem 3, [4] studies a random linear code $\mathcal{C}_{\text{out}}$ concatenated with a *fixed* inner code $\mathcal{C}_{\text{in}}$. (See also [5], which applies the same techniques for an application in compressive sensing). The work [4] derives bounds on the distance of $\mathcal{C}_{\text{out}} \circ \mathcal{C}_{\text{in}}$ in terms of (moments of) the weight distribution of $\mathcal{C}_{\text{in}}$. These bounds imply that $\mathcal{C}_{\text{out}} \circ \mathcal{C}_{\text{in}}$ approaches the GV bound in some cases, but doesn't seem to immediately imply Theorem 3.

Before discussing their techniques more, we note that the biggest difference between [4] and our work is that their question is about the behavior of random linear codes, and so naturally their approach crucially uses the fact that $\mathcal{C}_{\text{out}}$ is random. In contrast, the motivation for our work is to find deterministic sufficient conditions on $\mathcal{C}_{\text{out}}$, and we invoke a random linear outer code as a proof of concept that our approach is realizable.

Next, we briefly describe the techniques and implications of [4], relative to Theorem 3. The key result of [4] is an expression of the limiting trade-off between the rate R and the distance δ of $\mathcal{C}_{\text{out}} \circ \mathcal{C}_{\text{in}}$, in terms of the function $\phi(\tau) = \ln \mathbb{E}_X[e^{\tau X}]$, where X is the weight of a random codeword from $\mathcal{C}_{\text{in}}$ and where $\tau \leq 0$ parameterizes the trade-off.[7] They show that this trade-off meets the GV bound when $\mathcal{C}_{\text{in}}$ is the identity (trivial) code, and investigate how it behaves when $\mathcal{C}_{\text{in}}$ is a non-trivial code. Towards this, one can use their trade-off to work out the Taylor series for R around $\delta = 1/2$. It is not hard to see that under mild conditions on $\mathcal{C}_{\text{in}}$, the first two terms of this Taylor expansion vanish and hence we obtain $R = \Theta(\varepsilon^2) + O_{\mathcal{C}_{\text{in}}}(\varepsilon^3)$ when $\delta = 1/2 - \varepsilon$, where the $O_{\mathcal{C}_{\text{in}}}(\cdot)$ notation hides constants that depend on $\mathcal{C}_{\text{in}}$. This implies that if n_0 is a constant, independent even of ε, then $\mathcal{C}_{\text{out}} \circ \mathcal{C}_{\text{in}}$ approaches the GV bound. However, if n_0 is growing relative to ε (which it is in our case, as we take $\mathcal{C}_{\text{in}}$ to have rate ε), then the "constant" terms hiding in the $O_{\mathcal{C}_{\text{in}}}(\varepsilon^3)$ term may depend on n_0, which in turn may depend on ε. It seems plausible that when $\mathcal{C}_{\text{in}}$ is a random linear code, this dependence is mild[8] and something like Theorem 3 could be established with these techniques, but to the best of our knowledge such a proof has not appeared in the literature and does not seem to follow immediately.

---

[7] In more detail, this trade-off is given by $R = \frac{1}{n_0 \ln(2)}(\tau\phi'(\tau) - \phi(\tau))$ and $\delta = \frac{\phi'(\tau)}{n_0}$, for $\tau \leq 0$.

[8] In particular, as pointed out in [4], the first $d^\perp - 1$ terms of the Taylor series will agree with the GV bound, where $d^\perp$ is the dual distance of $\mathcal{C}_{\text{in}}$, which for a random linear code $\mathcal{C}_{\text{in}}$ is quite large.

Non-Concatenation-Based Explicit Constructions

As mentioned above, there have been several breakthroughs in the past few years obtaining explicit constructions of binary codes near the GV bound, and even efficient algorithms for them. In a breakthrough result, Ta-Shma [28] constructed *explicit* linear codes of relative distance $\frac{1-\varepsilon}{2}$ having rate $\varepsilon^{2+o(1)}$. Ta-Shma's codes are also ε-*balanced*, i.e., $\Delta(x,y) \in \left[\frac{1-\varepsilon}{2}, \frac{1+\varepsilon}{2}\right]$, and thus give rise to explicit ε-biased sample spaces, which are ubiquitous in pseudorandomness and derandomization. Works that followed gave efficient *decoding* of Ta-Shma codes and their variants [1, 16, 17, 26, 18] (see also [7] for a different, randomized, construction that slightly improves upon the rate of [28], and admits efficient decoding). We note that these codes are graph-based, and do not in general have a concatenated structure.

Results with Multiple i.i.d. Inner Codes

Thommesen showed that when the outer code is a Reed–Solomon code, and it is concatenated with n different random linear codes, one for each coordinate, chosen independently, then the resulting code lies on the GV bound with high probability [29]. Guruswami and Indyk devised efficient decoding algorithms for these codes, based on *list-recoverability* of the outer code [10]. That work used a Reed–Solomon code as the outer code, which is list-recoverable up to the Johnson bound. Later, Rudra [27] observed that the parameters could be improved by swapping out the Reed–Solomon code for a code that can be list-recovered up to capacity, for example a Folded Reed–Solomon code. Later work obtained nearly-linear-time decoding algorithms by swapping out the outer code for a capacity-achieving list-recoverable code with near-linear-time list-recovery algorithms [15, 21]. Codes with multiple i.i.d. inner codes have also been studied in [32, 8].

We also mention the work of Guruswami and Rudra [13], who show that the same construction (a list-recoverable code concatenated with n different i.i.d. random linear codes) is list-*decodable* up to capacity with high probability. In the results [10, 27, 15, 21] mentioned above, list-recovery of the outer code was needed for *algorithms*, not the combinatorial result (which follows already from [29]). In contrast, in [13], the list-recoverability of the outer code is needed for the combinatorial result itself. In that sense, the flavor is similar to our sufficient condition in Section 5 in the full version, although the techniques are very different, and in our work we only use one inner code.

Further Low-randomness Constructions of Binary Codes on GV Bound

If one's goal is to explicitly construct a binary code that achieves that GV bound, at least two types of partial results may be considered as subgoals. In the first class of results, one seeks explicit codes whose rate vs. distance tradeoff is as close to the GV bound as possible. This includes the works discussed in the first two paragraphs of Section 1.1 above. A second path is to seek codes that fully attain the GV bound, and strive to minimize the amount of randomness used in their construction.

Varshamov's classic result [30] is that a random linear code likely achieves the GV bound. Constructing such a code of length n and rate R requires sampling either a random generating matrix or a random parity-check matrix, and thus $O\big(\min\{R, 1-R\} \cdot n^2\big)$ random bits are needed. Two classical elementary constructions – the Wozencraft ensemble [22] and the random Toeplitz Matrix construction (e.g., [14, Exercise 4.6]) – are able to reduce the needed randomness to $O(n)$.

So far, no codes achieving the GV bound using $o(n)$ randomness are known. Moreover, there is a certain natural obstacle, which we now describe, that needs to be tackled before sublinear randomness can be achieved. Say that a random code $\mathcal{C} \subseteq \mathbb{F}_2^n$ is *uniform* if

every $x \in \mathbb{F}_2^n \setminus \{0\}$ appears in the code with the same probability, namely, $p_{R,n} = \frac{2^{Rn}-1}{2^n-1}$. It is not hard to prove via a union bound that a uniform linear code achieves the GV bound with high probability (this is exactly Varshamov's observation). To the best of our knowledge, every known GV-bound construction to date, including the linear randomness constructions mentioned above, is uniform. Unfortunately, a uniform code ensemble with sublinear randomness cannot exist as long as R is bounded away from 1. Indeed, to have events that occur with probability $p_{R,n}$, at least $\log_2 \frac{1}{p_{R,n}} \approx (1-R)n$ random bits are required. Therefore, a code construction obtaining the GV bound with sublinear randomness would have to do so without being uniform (see also [23, Section 5]). We have hope that our sufficient conditions in Theorems 5.1 and 6.2 could be attained by non-uniform codes. For example, as discussed above, the soft-decoding-like condition of Theorem 5.1 is reminiscent of results on soft-list-decoding and soft-list-recovery, which in different parameter regimes can even be achieved by deterministic codes.

A related line of work [12, 25, 23] attempts to construct codes that enjoy a broad class of desirable combinatorial properties similar to those of random linear codes using as little randomness as possible. Such properties include not just the GV bound, but also list decodability up to the *Elias bound* (see [23]), list recoverability, and, more generally, *local similarity* (see [23, Definition 2.14]) to a random linear code.

1.2 Technical Overview

In this section we give an overview of the main technical ideas. This section also serves as an outline of the full version of the paper.

Section 3: A moment-based framework

In Section 3, we set up a framework that will be useful for the results in Section 4 and Section 5. We describe this approach here.

Suppose that we are trying to encode a message $m \in \mathbb{F}_q^k$ with our concatenated code $\mathcal{C} = \mathcal{C}_{\text{out}} \circ \mathcal{C}_{\text{in}}$, to obtain $\mathcal{C}(m) = w \in \mathbb{F}_2^{n \cdot n_0}$. Each symbol of w is indexed by some $\alpha \in [n]$ and some $\beta \in [n_0]$; this symbol is equal to

$$(\mathcal{C}_{\text{in}}(\mathcal{C}_{\text{out}}(m)_\alpha))_\beta = \langle \mathcal{C}_{\text{out}}(m)_\alpha, b_\beta \rangle,$$

where b_β is the β'th row for a generator matrix $G_0 \in \mathbb{F}_2^{n_0 \times k_0}$ for $\mathcal{C}_{\text{in}}$, and where the $\langle \cdot, \cdot \rangle$ notation denotes the dot product over $\mathbb{F}_2$. This motivates the definition of a variable $X_m \in \mathbb{R}$ defined by

$$X_m = \sum_{\alpha \in [n]} \sum_{\beta \in [n_0]} (-1)^{\langle \mathcal{C}_{\text{out}}(m)_\alpha, b_\beta \rangle}.$$

Indeed, X_m is the bias of $w = \mathcal{C}(m)$; the weight of w is at least $\frac{1}{2} - O(\varepsilon N)$ if and only if X_m is at most $O(\varepsilon N)$. Thus, to show that the code $\mathcal{C}$ has distance at least $\frac{1}{2} - O(\varepsilon N)$, it suffices to show that

$$\max_{m \in \mathbb{F}_q^k \setminus \{0\}} X_m = O(\varepsilon N).$$

Our strategy will be to consider a large moment of X_m over the choice of a random nonzero message m:

$$\mathbb{E}_{m \sim \mathbb{F}_q^k \setminus \{0\}}[X_m^r]$$

for some appropriate r. If we can show that this is smaller than $(c\varepsilon N)^r/q^k$, then Markov's inequality will imply that

$$\Pr_{m\sim\mathbb{F}_q^k\setminus\{0\}}[X_m \geq c\varepsilon N] \leq \frac{\mathbb{E}_{m\sim\mathbb{F}_q^k\setminus\{0\}}[X_m^r]}{(c\varepsilon N)^r} < \frac{1}{q^k},$$

and in particular that there are no messages m so that $X_m \geq c\varepsilon N$.

In Lemma 3.3, we take a Fourier transform in order to re-write $\mathbb{E}[X_m^r]$ as a quantity involving $\mathcal{C}_{\text{out}}^\perp$. This quantity can be thought of as follows. For every integer-valued matrix[9] $V \in \mathbb{Z}_{\geq 0}^{n_0\times n}$ with entries that sum to r, we consider a vector $g_V \in \mathbb{F}_q^n$ defined by considering the matrix $G_0^T \cdot V \in \mathbb{F}_2^{k_0\times n}$ and then treating it as a vector $g_V \in \mathbb{F}_q^n$ by identifying each of the columns in $\mathbb{F}_2^{k_0}$ with elements of $\mathbb{F}_q$. Then the quantity in Lemma 3.3 has to do with the number of these vectors g_V that are in $\mathcal{C}_{\text{out}}^\perp$. The exact expression doesn't matter too much for this informal discussion; instead we explain below how we use this re-writing to prove Theorem 4.2 and Theorem 5.1.

Section 4: Most codes $\mathcal{C}_{\text{out}}$ are good

Theorem 4.2 informally says that if $\mathcal{C}_{\text{out}}$ is a random linear code, then with high probability $\mathcal{C}_{\text{out}} \circ \mathcal{C}_{\text{in}}$ is near the GV bound. In the proof, we use our framework from Section 3, and show that with high probability over $\mathcal{C}_{\text{out}}$, the moment $\mathbb{E}_m[X_m^r]$ is small for an appropriate r. To do this, we need to count the number of matrices V described above that are likely to land in $\mathcal{C}_{\text{out}}^\perp$. Since $\mathcal{C}_{\text{out}}$ is a random linear code, so is $\mathcal{C}_{\text{out}}^\perp$, and so the probability of any particular *non-zero* g_V landing in it is small (about $1/q^k$), while of course the probability that 0 is contained in $\mathcal{C}_{\text{out}}^\perp$ is 1. Thus, the challenge is understanding how many g_V-s are actually zero. There are two ways that a matrix V as described above could lead to $g_V = 0$: Either $V = 0 \bmod 2$, or else V is non-zero mod 2 but $G_0^T V = 0$. The first case can be counted straightforwardly. For the second, we leverage the weight distribution that the inner code $\mathcal{C}_{\text{in}}$ is likely to have. We note that this is the only place (in any of our arguments) that we need $\mathcal{C}_{\text{in}}$ to be a random linear code: We just need it to have approximately the "right" weight distribution.

Section 5: A soft-decoding-like sufficient condition

The expression that we get for $\mathbb{E}_m[X_r^m]$ in Lemma 3.3 directly inspires our soft-decoding-like sufficient condition in Theorem 5.1. One can view the task of counting the matrices V so that $g_V \in \mathcal{C}_{\text{out}}^\perp$ as choosing a random V and asking about the probability that $g_V \in \mathcal{C}_{\text{out}}^\perp$. If the columns of V were independent, then this would be the same as choosing the coordinates of g_V i.i.d. from some distribution $\mathcal{D}$. Thus we would get a requirement on $\Pr_{x\sim\mathcal{D}^n}[x \in \mathcal{C}_{\text{out}}^\perp]$, similar to the condition in Equation (1) that we end up with.

Of course, the coordinates are not independent (because the total weight of V is fixed to be r), but this can be solved. In more detail, we choose r to be a Poisson random variable, which in this setting makes the columns of V independent. One hiccup is that the "Poisson-ized" distribution turns out to be meaningfully different than the original distribution, in the sense that it is much more likely that $g_V = 0$ in the Poisson-ized version. This means that the "natural" soft-decoding-like condition that one would get out of this is not realizable: The probability that $g_V \in \mathcal{C}_{\text{out}}^\perp$ is much bigger than we want it to be, for *any*

[9] In the actual quantity, the entries of this matrix are ordered, and we denote it $\mathcal{V}$ instead of V; we ignore the ordering in this discussion for simplicity.

$\mathcal{C}_{\text{out}}$, just because g_V is too likely to be zero. Fortunately, this seems to be the only obstacle: as in Equation (1), we separate out the $g_V = 0$ term (using the analysis from Section 4) to arrive at a condition that *is* realizable. We explain why the condition is realizable – that is, why there exists a $\mathcal{C}_{\text{out}}$ that meets it – in Remark 8.

Section 6: A smoothness condition on $\mathcal{C}_{\text{out}}$

For our second sufficient condition, we depart from our moment-based framework and work from first principles. Our main theorem in Section 6 is Theorem 6.2, which informally says that if the elements of $\mathcal{C}_{\text{out}}$ have "smooth" enough distributions of symbols, in the sense that they each have large enough min-entropy, that $\mathcal{C} = \mathcal{C}_{\text{out}} \circ \mathcal{C}_{\text{in}}$ will lie near the GV bound with high probability. The basic idea is to consider a *worst-case* assignment of symbols in $\mathbb{F}_q$ to codewords in $\mathcal{C}_{\text{in}}$; this assignment need not be linear and can depend on a particular codeword $c \in \mathcal{C}_{\text{out}}$. Such a worst-case assignment would simply assign the lowest-weight codewords in $\mathcal{C}_{\text{in}}$ to the most frequent symbols in a codeword $c \in \mathcal{C}_{\text{out}}$. Using the weight distribution that $\mathcal{C}_{\text{in}}$ is likely to have, along with the min-entropy assumption, we can show that this worst-case assignment will *still* result in codewords $w \in \mathcal{C}$ of weight at least $\frac{1}{2} - O(\varepsilon)$.

We note that, unlike our sufficient condition from Section 5, we don't have a proof of feasibility for our smoothness condition. That is, as far as we know, there may not be any linear code $\mathcal{C}_{\text{out}}$ that is smooth in this sense. However, as a proof of concept we mention in Remark 9 that a random linear code will have a similar property with high probability. Moreover, we find it plausible that codewords of *algebraically structured* codes (say, Folded Reed–Solomon codes, Folded Multiplicity, or even large sub-codes of plain Reed–Solomon codes), would satisfy this property, even if a random code does not.

References

1 Vedat Levi Alev, Fernando Granha Jeronimo, Dylan Quintana, Shashank Srivastava, and Madhur Tulsiani. List decoding of direct sum codes. In *Proceedings of the 31st Symposium on Discrete Algorithms (SODA 2020)*, pages 1412–1425. ACM-SIAM, 2020.

2 Noga Alon, Jehoshua Bruck, Joseph Naor, Moni Naor, and Ron M. Roth. Construction of asymptotically good low-rate error-correcting codes through pseudo-random graphs. *Information Theory, IEEE Transactions on*, 38(2):509–516, 1992.

3 Noga Alon, Oded Goldreich, Johan Håstad, and René Peralta. Simple constructions of almost k-wise independent random variables. *Random Structures & Algorithms*, 3(3):289–304, 1992.

4 Alexander Barg, Jørn Justesen, and Christian Thommesen. Concatenated codes with fixed inner code and random outer code. *IEEE Transactions on Information Theory*, 47(1):361–365, 2001.

5 Alexander Barg and Arya Mazumdar. Small ensembles of sampling matrices constructed from coding theory. In *2010 IEEE International Symposium on Information Theory*, pages 1963–1967. IEEE, 2010.

6 Avraham Ben-Aroya and Amnon Ta-Shma. Constructing small-bias sets from algebraic-geometric codes. *Theory of Computing*, 9(5):253–272, 2013.

7 Guy Blanc and Dean Doron. New near-linear time decodable codes closer to the GV bound. In *Proceedings of the 37th Computational Complexity Conference (CCC 2022)*. Schloss-Dagstuhl-Leibniz Zentrum für Informatik, 2022.

8 È L Blokh and Victor Vasilievich Zyablov. Existence of linear concatenated binary codes with optimal correcting properties. *Problemy Peredachi Informatsii*, 9(4):3–10, 1973.

9 G. David Forney. Concatenated codes. Technical Report 440, Research Laboratory of Electronics, MIT, 1965.

10 Venkatesan Guruswami and Piotr Indyk. Efficiently decodable codes meeting gilbert-varshamov bound for low rates. In *Proceedings of the 15th Symposium on Discrete Algorithms (SODA 2004)*, pages 756–757. ACM-SIAM, 2004.

11 Venkatesan Guruswami and Piotr Indyk. Linear-time encodable/decodable codes with near-optimal rate. *IEEE Transactions on Information Theory*, 51(10):3393–3400, 2005.

12 Venkatesan Guruswami and Jonathan Mosheiff. Punctured Low-Bias Codes Behave Like Random Linear Codes. In *Proceedings of the 63rd Annual Symposium on Foundations of Computer Science (FOCS 2022)*, pages 36–45. IEEE, 2022.

13 Venkatesan Guruswami and Atri Rudra. The existence of concatenated codes list-decodable up to the hamming bound. *IEEE Transactions on information theory*, 56(10):5195–5206, 2010.

14 Venkatesan Guruswami, Atri Rudra, and Madhu Sudan. Essential coding theory. URL: `http://www.cse.buffalo.edu/faculty/atri/courses/coding-theory/book`.

15 Brett Hemenway, Noga Ron-Zewi, and Mary Wootters. Local list recovery of high-rate tensor codes and applications. *SIAM Journal on Computing*, pages FOCS17–157, 2019.

16 Fernando Granha Jeronimo, Dylan Quintana, Shashank Srivastava, and Madhur Tulsiani. Unique decoding of explicit ε-balanced codes near the Gilbert–Varshamov bound. In *Proceedings of the 61st Annual Symposium on Foundations of Computer Science (FOCS 2020)*, pages 434–445. IEEE, 2020.

17 Fernando Granha Jeronimo, Shashank Srivastava, and Madhur Tulsiani. Near-linear time decoding of Ta-Shma's codes via splittable regularity. In *Proceedings of the 53rdth Annual Symposium on Theory of Computing (STOC 2021)*, pages 1527–1536. ACM, 2021.

18 Fernando Granha Jeronimo, Shashank Srivastava, and Madhur Tulsiani. List decoding of tanner and expander amplified codes from distance certificates. In *Proceedings of the 64th Annual Symposium on Foundations of Computer Science (FOCS 2023)*, pages 1682–1693. IEEE, 2023.

19 Jørn Justesen. Class of constructive asymptotically good algebraic codes. *IEEE Transactions on Information Theory*, 18(5):652–656, 1972.

20 Ralf Koetter and Alexander Vardy. Algebraic soft-decision decoding of reed-solomon codes. *IEEE Transactions on Information Theory*, 49(11):2809–2825, 2003.

21 Swastik Kopparty, Nicolas Resch, Noga Ron-Zewi, Shubhangi Saraf, and Shashwat Silas. On list recovery of high-rate tensor codes. *IEEE Transactions on Information Theory*, 67(1):296–316, 2020.

22 James L. Massey. Threshold decoding. Technical Report 410, Research Laboratory of Electronics, MIT, 1963.

23 Jonathan Mosheiff, Nicolas Resch, Kuo Shang, and Chen Yuan. Randomness-efficient constructions of capacity-achieving list-decodable codes. *arXiv preprint*, 2024.

24 Joseph Naor and Moni Naor. Small-bias probability spaces: Efficient constructions and applications. *SIAM Journal on Computing*, 22(4):838–856, 1993.

25 Aaron (Louie) Putterman and Edward Pyne. Pseudorandom Linear Codes Are List-Decodable to Capacity. In *Proceedings of the 15th Innovations in Theoretical Computer Science Conference (ITCS 2024)*, pages 90:1–90:21. Schloss-Dagstuhl-Leibniz Zentrum für Informatik, 2024.

26 Silas Richelson and Sourya Roy. Gilbert and Varshamov meet Johnson: List-decoding explicit nearly-optimal binary codes. In *Proceedings of the 64th Annual Symposium on Foundations of Computer Science (FOCS 2023)*, pages 194–205. IEEE, 2023.

27 Atri Rudra. *List decoding and property testing of error-correcting codes*. University of Washington, 2007.

28 Amnon Ta-Shma. Explicit, almost optimal, ε-balanced codes. In *Proceedings of the 49th Annual Symposium on Theory of Computing (STOC 2017)*, pages 238–251. ACM, 2017.

29 Christian Thommesen. The existence of binary linear concatenated codes with Reed–Solomon outer codes which asymptotically meet the Gilbert–Varshamov bound. *IEEE Transactions on Information Theory*, 29(6):850–853, 1983.

30 Rom Rubenovich Varshamov. Estimate of the number of signals in error correcting codes. *Docklady Akad. Nauk, SSSR*, 117:739–741, 1957.

31 Victor Vasilievich Zyablov. An estimate of the complexity of constructing binary linear cascade codes. *Problemy Peredachi Informatsii*, 7(1):5–13, 1971.

32 Victor Vasilievich Zyablov. An estimate of the complexity of constructing binary linear cascade codes. *Problemy Peredachi Informatsii*, 7(1):5–13, 1971.