Chapter 5

LIST DECODABILITY OF RANDOM LINEAR CONCATENATED CODES

5.1 Introduction

In Chapter 2, we saw that for any fixed alphabet of size $q \geqslant 2$ there exist codes of rate R that can be list decoded up to $H_q^{-1}(1-R-\varepsilon)$ fraction of errors with list of size $O(1/\varepsilon)$. For linear codes one can show a similar result with lists of size $q^{O(1/\varepsilon)}$. These results are shown by choosing the code at random. However, as we saw in Chapter 4 the explicit constructions of codes over finite alphabets are nowhere close to achieving list-decoding capacity.

The linear codes in Chapter 4 are based on code concatenation. A natural question to ask is whether linear codes based on code concatenation can get us to list-decoding capacity for fixed alphabets.

In this chapter, we answer the question above in the affirmative. In particular, in Section 5.4 we show that if the outer code is random linear code and the inner codes are also (independent) random linear codes, then the resulting concatenated codes can get to within ε of the list-decoding capacity with list of constant size depending on ε only. In Section 5.5, we also show a similar result when the outer code is the folded Reed-Solomon code from Chapter 3. However, we can only show the latter result with polynomial-sized lists.

The way to interpret the results in this chapter is the following. We exhibit an ensemble of random linear codes with more structure than general random (linear) codes that achieve the list-decoding capacity. This structure gives rise to the hope of being able to list decode such a random ensemble of codes up to the list-decoding capacity. Furthermore, for designing explicit codes that meet the list-decoding capacity, one can concentrate on concatenated codes. Another corollary of our result is that we need fewer random bits to construct a code that achieves the list-decoding capacity. In particular, a general random linear code requires number of random bits that grows quadratically with the block length. On the other hand, random concatenated codes with outer codes as folded Reed-Solomon code require number of random bits that grows quasi-linearly with the block length.

The results in this chapter (and their proofs) are inspired by the following results due to Blokh and Zyablov [19] and Thommesen [102]. Blokh and Zybalov show that random concatenated linear binary codes (where both the outer and inner codes are chosen uniformly at random) have with high probability the same minimum distance as general random linear codes. Thommesen shows a similar result when the outer code is the Reed-Solomon code. The rate versus distance tradeoff achieved by random linear codes satisfies the so called Gilbert-Varshamov (GV) bound. However, like list decodability of binary codes, explicit codes that achieve the GV bound are not known. Coming up with such explicit constructions is one of the biggest open questions in coding theory.

5.2 Preliminaries

We will consider outer codes that are defined over $\mathbb{F}_Q$, where $Q = q^k$ for some fixed $q \geqslant 2$. The outer code will have rate and block length of R and N respectively. The outer code C_{out} will either be a random linear code over $\mathbb{F}_Q$ or the folded Reed-Solomon code from Chapter 3. In the case when C_{out} is random, we will pick C_{out} by selecting $K = RN$ vectors uniformly at random from $\mathbb{F}_Q^N$ to form the rows of the generator matrix. For every position $1 \leqslant i \leqslant N$, we will choose an inner code C_{in}^i to be a random linear code over $\mathbb{F}_q$ of block length n and rate $r = k/n$. In particular, we will work with the corresponding generator matrices $\mathbf{G}_i$, where every $\mathbf{G}_i$ is a random $k \times n$ matrix over $\mathbb{F}_q$. All the generator matrices $\mathbf{G}_i$ (as well as the generator matrix for C_{out}, when we choose a random C_{out}) are chosen independently. This fact will be used crucially in our proofs.

Given the outer code C_{out} and the inner codes C_{in}^i, the resulting concatenated code $C = C_{out} \circ (C_{in}^1, \ldots, C_{in}^N)$ is constructed as follows.[1] For every codeword $\mathbf{u} = (\mathbf{u}_1, \ldots, \mathbf{u}_N) \in C_{out}$, the following codeword is in C:

$$\mathbf{uG} \overset{def}{=} (\mathbf{u}_1\mathbf{G}_1, \mathbf{u}_2\mathbf{G}_2, \ldots, \mathbf{u}_N\mathbf{G}_N),$$

where the operations are over $\mathbb{F}_q$.

We will need the following notions of the weight of a vector. Given a vector $\mathbf{v} \in \mathbb{F}_q^{nN}$, its Hamming weight is denoted by $wt(\mathbf{v})$. Given a vector $\mathbf{y} = (y_1, \ldots, y_N) \in (\mathbb{F}_q^n)^N$ and a subset $S \subseteq [N]$, we will use $wt_S(\mathbf{y})$ to denote the Hamming weight over $\mathbb{F}_q$ of the subvector $(y_i)_{i \in S}$. Note that $wt(\mathbf{y}) = wt_{[N]}(y)$.

We will need the following lemma due to Thommesen, which is stated in a slightly different form in [102]. For the sake of completeness we also present its proof.

Lemma 5.1 ([102]). *Given a fixed outer code C_{out} of block length N and an ensemble of random inner linear codes of block length n given by generator matrices $\mathbf{G}_1, \ldots, \mathbf{G}_N$ the following is true. Let $\mathbf{y} \in \mathbb{F}_q^{nN}$. For any codeword $\mathbf{u} \in C_{out}$, any non-empty subset $S \subseteq [N]$ such that $\mathbf{u}_i \neq 0$ for all $i \in S$ and any integer $h \leqslant n|S| \cdot \left(1 - \frac{1}{q}\right)$:*

$$\Pr[wt_S(\mathbf{uG} - \mathbf{y}) \leqslant h] \leqslant q^{-n|S|\left(1 - H_q\left(\frac{h}{n|S|}\right)\right)},$$

where the probability is taken over the random choices of $\mathbf{G}_1, \ldots, \mathbf{G}_N$.

[1] Note that this is a slightly general form of code concatenation that is considered in Chapter 4. We did consider the current generalization briefly in Section 4.4.

Proof. Let $|S| = s$ and w.l.o.g. assume that $S = [s]$. As the choices for $\mathbf{G}_1, \ldots, \mathbf{G}_N$ are made independently, it is enough to show that the claimed probability holds for the random choices for $\mathbf{G}_1, \ldots, \mathbf{G}_s$. For any $1 \leqslant i \leqslant s$ and any $y \in \mathbb{F}_q^n$, since $\mathbf{u}_i \neq 0$, we have $\Pr_{\mathbf{G}_i}[\mathbf{u}_i\mathbf{G}_i = y] = q^{-n}$. Further, these probabilities are independent for every i. Thus, for any $\mathbf{y} = \langle y_1, \ldots, y_s\rangle \in (\mathbb{F}_q^n)^s$, $\Pr_{\mathbf{G}_1,\ldots,\mathbf{G}_s}[\mathbf{u}_i\mathbf{G}_i = y_i \text{ for every } 1 \leqslant i \leqslant s] = q^{-ns}$. This implies that:

$$\Pr_{\mathbf{G}_1,\ldots,\mathbf{G}_s}[wt_S(\mathbf{uG} - \mathbf{y}) \leqslant h] = q^{-ns}\sum_{j=0}^{h}\binom{ns}{j}(q-1)^j.$$

The claimed result follows from the Proposition 2.1. □

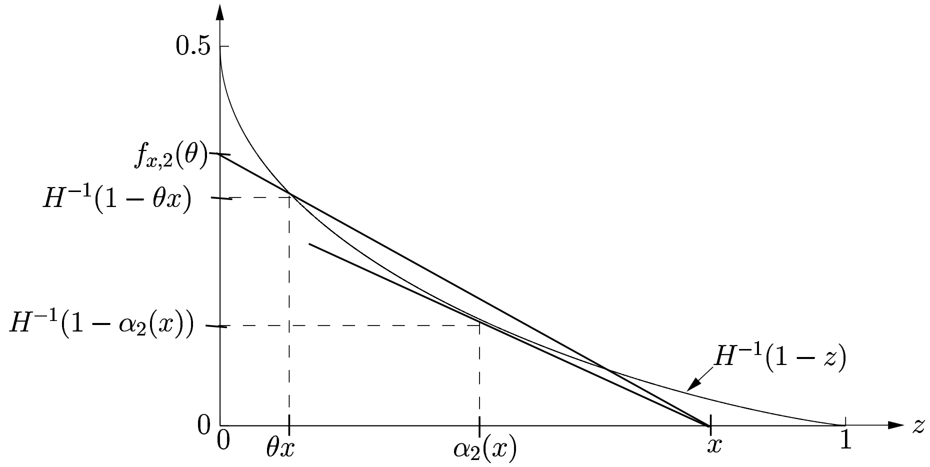

Figure 5.1: Geometric interpretations of functions $\alpha_2(\cdot)$ and $f_{x,2}(\cdot)$.

For $0 \leqslant z \leqslant 1$ define

$$\alpha_q(z) = 1 - H_q(1 - q^{z-1}). \tag{5.1}$$

We will need the following property of the function above.

Lemma 5.2. *Let $q \geqslant 2$ be an integer. For every $0 \leqslant z \leqslant 1$,*

$$\alpha_q(z) \leqslant z.$$

Proof. The proof follows from the subsequent sequence of relations:

$$\begin{aligned}\alpha_q(z) &= 1 - H_q(1 - q^{z-1}) \\ &= 1 - (1 - q^{z-1})\log_q(q-1) + (1 - q^{z-1})\log_q(1 - q^{z-1}) + q^{z-1}(z-1) \\ &= zq^{z-1} + (1 - q^{z-1})\left(1 - \log_q\left(\frac{q-1}{1-q^{z-1}}\right)\right) \\ &\leqslant z,\end{aligned}$$

where the last inequality follows from the facts that $q^{z-1} \leqslant 1$ and $1 - q^{z-1} \leqslant 1 - 1/q$, which implies that $\log_q\left(\frac{q-1}{1-q^{z-1}}\right) \geqslant 1$. □

We will consider the following function

$$f_{x,q}(\theta) = (1-\theta)^{-1} \cdot H_q^{-1}(1 - \theta x),$$

where $0 \leqslant \theta, x \leqslant 1$. We will need the following property of this function.[2]

Lemma 5.3 ([102]). *Let $q \geqslant 2$ be an integer. For any $x \geqslant 0$ and $0 \leqslant y \leqslant \alpha_q(x)/x$,*

$$\min_{0 \leqslant \theta \leqslant y} f_{x,q}(\theta) = (1-y)^{-1} H_q^{-1}(1 - xy).$$

Proof. The proof follows from the subsequent geometric interpretations of $f_{x,q}(\cdot)$ and $\alpha_q(\cdot)$. See Figure 5.1 for a pictorial illustration of the arguments used in this proof (for $q = 2$).

First, we claim that for any $0 \leqslant z_0 \leqslant 1$, $\alpha_q(z)$ satisfies the following property: the line segment between $(\alpha_q(z_0), H_q^{-1}(1 - \alpha_q(z_0)))$ and $(z_0, 0)$ is tangent to the curve $H_q^{-1}(1-z)$ at $\alpha_q(z_0)$.

Thus, we need to show that

$$\frac{-H_q^{-1}(1 - \alpha_q(z_0))}{z_0 - \alpha_q(z_0)} = (H_q^{-1})'(1 - \alpha_q(z_0)). \tag{5.2}$$

One can check that $(H_q^{-1})'(1-x) = \frac{-1}{H_q'(H_q^{-1}(1-x))} = \frac{-1}{\log_q(q-1) - \log_q(H_q^{-1}(1-x)) + \log_q(1 - H_q^{-1}(1-x))}$. Now,

$$\begin{aligned}z_0 - \alpha_q(z_0) =\ & z_0 - 1 + (1 - q^{z_0-1})\log_q(q-1) - (1 - q^{z_0-1})\log_q(1 - q^{z_0-1}) \\ & - q^{z_0-1}(z_0 - 1) \\ =\ & (1 - q^{z_0-1}) \cdot \big(\log_q(q-1) - \log_q(1 - q^{z_0-1}) + z_0 - 1\big) \\ =\ & H_q^{-1}(1 - \alpha_q(z_0)) \cdot \big(\log_q(q-1) - \log_q(H_q^{-1}(1 - \alpha_q(z_0))) \\ & + \log_q(1 - H_q^{-1}(1 - \alpha_q(z_0)))\big) \\ =\ & \frac{-H_q^{-1}(1 - \alpha_q(z_0))}{(H_q^{-1})'(1 - \alpha_q(z_0))},\end{aligned}$$

[2] Lemma 5.3 was proven in [102] for the $q = 2$ case. Here we present the straightforward extension of the result for general q.

which proves (5.2) (where we have used the expression for $\alpha_q(z)$ and $(H_q^{-1})'(1-z)$ and the fact that $1-q^{z-1} = H_q^{-1}(1-\alpha_q(z))$).

We now claim that $f_{x,q}(\theta)$ is the intercept of the line segment through $(x,0)$ and $(\theta x, H_q^{-1}(1-\theta x))$ on the "y-axis." Indeed, the "y-coordinate" increases by $H_q^{-1}(1-\theta x)$ in the line segment from x to θx. Thus, when the line segment crosses the "y-axis", it would cross at an intercept of $1/(1-\theta)$ times the "gain" going from x to θx. The lemma follows from the fact that the function $H_q^{-1}(1-r)$ is a decreasing (strictly) convex function of r and thus, the minimum of $f_{x,q}(\theta)$ would occur at $\theta = y$ provided $yx \leqslant \alpha_q(x)$. □

5.3 Overview of the Proof Techniques

In this section, we will highlight the main ideas in our proofs. Our proofs are inspired by Thommesen's proof of the following result [102]. Binary linear concatenated codes with an outer Reed-Solomon code and independently and randomly chosen inner codes meet the Gilbert-Varshamov bound[3]. Given that our proof builds on the proof of Thommesen, we start out by reviewing the main ideas in his proof.

The outer code C_{out} in [102] is a Reed-Solomon code of length N and rate R (over $\mathbb{F}_Q$) and the inner codes (over $\mathbb{F}_q$ such that $Q = q^k$ for some $k \geqslant 1$) are generated by N randomly chosen $k \times n$ generator matrices $\mathbf{G} = (\mathbf{G}_1, \ldots, \mathbf{G}_N)$, where $r = k/n$. Note that since the final code will be linear, to show that with high probability the concatenated code will have distance close to $H^{-1}(1-rR)$, it is enough to show that the probability of the Hamming weight of $\mathbf{uG}$ over $\mathbb{F}_q$ being at most $(H^{-1}(1-rR)-\varepsilon)nN$ (for some Reed-Solomon codeword $\mathbf{u} = (\mathbf{u}_1, \ldots, \mathbf{u}_N)$), is small. Let us now concentrate on a fixed codeword $\mathbf{u} \in C_{out}$. Now note that if for some $1 \leqslant i \leqslant N$, $\mathbf{u}_i = 0$, then for every choice of $\mathbf{G}_i$, $\mathbf{u}_i\mathbf{G}_i = 0$. Thus, only the non-zero symbols of $\mathbf{u}$ contribute to $wt(\mathbf{uG})$. Further, for a non-zero $\mathbf{u}_i$, $\mathbf{u}_i\mathbf{G}_i$ takes all the values in $\mathbb{F}_q^n$ with equal probability over the random choices of $\mathbf{G}_i$. Also for two different non-zero positions $i_1 \neq i_2$ in $\mathbf{u}$, the random variables $\mathbf{u}_{i_1}\mathbf{G}_{i_1}$ and $\mathbf{u}_{i_2}\mathbf{G}_{i_2}$ are *independent* (as the choices for $\mathbf{G}_{i_1}$ and $\mathbf{G}_{i_2}$ are independent). This implies that $\mathbf{uG}$ takes each of the possible $q^{n\cdot wt(\mathbf{u})}$ values in $\mathbb{F}_q^{nN}$ with the same probability. Thus, the total probability that $\mathbf{uG}$ has a Hamming weight of at most h is $\sum_{w=0}^{h}\binom{n\cdot wt(\mathbf{u})}{w}q^{-n\cdot wt(\mathbf{u})} \leqslant q^{-n\cdot wt(\mathbf{u})\left(1-H\left(\frac{h}{n\cdot wt(\mathbf{u})}\right)\right)}$. The rest of the argument follows by doing a careful union bound of this probability for all non zero codewords in C_{out} (using the known weight distribution of Reed-Solomon codes[4]).

Let us now try to extend the idea above to show a similar result for list decoding of a code similar to the one above (the inner codes are the same but we might change the outer

[3] A binary code of rate $\mathcal{R}$ satisfies the Gilbert-Varshamov bound if it has relative distance at least $H^{-1}(1-\mathcal{R})$.

[4] In fact, the argument works just as well for any code that has a weight distribution that is close to that of the Reed-Solomon code. In particular, it also works for folded Reed-Solomon codes– we alluded to this fact in Section 4.4.

code). We want to show that for any Hamming ball of radius at most $h = (H^{-1}(1-rR) - \varepsilon)nN$ has at most L codewords from the concatenated code C (assuming we want to show that L is the worst case list size). To show this let us look at a set of $L+1$ codewords from C and try to prove that the probability that all of them lie within some ball of radius h is small. Let $\mathbf{u}^1, \ldots, \mathbf{u}^{L+1}$ be the corresponding codewords in C_{out}. As a warm up, let us try and show this for a Hamming ball centered around $\mathbf{0}$. Thus, we need to show that all of the $L+1$ codewords $\mathbf{u}^1\mathbf{G}, \ldots, \mathbf{u}^{L+1}\mathbf{G}$ have Hamming weight at most h. Note that $L = 0$ reverts back to the setup of Thommesen, that is, any fixed codeword has weight at most h with small probability. However, we need all the codewords to have small weight. Extending Thommesen's proof would be straightforward if the random variables corresponding to each of $\mathbf{u}^i\mathbf{G}$ having small weight were independent. In particular, if we can show that for every position $1 \leqslant i \leqslant N$, all the non-zero symbols in $\{\mathbf{u}_i^1, \mathbf{u}_i^2, \ldots, \mathbf{u}_i^{L+1}\}$ are linearly independent[5] over $\mathbb{F}_q$ then the generalization of Thommesen's proof is immediate.

Unfortunately, the notion of independence discussed above does *not* hold for every $L+1$ tuple of codewords from C_{out}. A fairly common trick to get independence when dealing with linear codes is to look at messages that are linearly independent. It turns out that if C_{out} is a random linear code over $\mathbb{F}_Q$ then we have a good approximation of the the notion of independence above. Specifically, we show that with very high probability for a linearly independent (over $\mathbb{F}_Q$) set of messages[6] $\mathbf{m}^1, \ldots, \mathbf{m}^{L+1}$, the set of codewords $\mathbf{u}^1 = C_{out}(\mathbf{m}^1), \ldots, \mathbf{u}^N = C_{out}(\mathbf{m}^N)$ have the following approximate independence property. For most of the positions $1 \leqslant i \leqslant N$, most of the non-zero symbols in $\{\mathbf{u}_i^1, \ldots, \mathbf{u}_i^N\}$ are linearly independent over $\mathbb{F}_q$. It turns out that this approximate notion of independence is enough for Thommesen's proof to go through. Generalizing this argument to the case when the Hamming ball is centered around an arbitrary vector from $\mathbb{F}_q^{nN}$ is straightforward.

We remark that the notion above crucially uses the fact that the outer code is a random linear code. However, the argument is bit more tricky when C_{out} is fixed to be (say) the Reed-Solomon code. Now even if the messages $\mathbf{m}^1, \ldots, \mathbf{m}^{L+1}$ are linearly independent it is not clear that the corresponding codewords will satisfy the notion of independence in the above paragraph. Interestingly, we can show that this notion of independence is equivalent to showing good list recoverability properties for C_{out}. Reed-Solomon codes are however not known to have optimal list recoverability (which is what is required in our case). In fact, the results in Chapter 6 show that this is *impossible* for Reed-Solomon codes in general. However, as we saw in Chapter 3, folded Reed-Solomon codes *do* have optimal list recoverability and we exploit this fact in this chapter.

[5] Recall that $\mathbb{F}_{q^k}$ is isomorphic to $\mathbb{F}_q^k$ and hence, we can think of the symbols in $\mathbb{F}_Q$ as vectors over $\mathbb{F}_q$.

[6] Again any set of $L+1$ messages need not be linearly independent. However, it is easy to see that some subset of $J = \lceil \log_Q(L+1) \rceil$ of messages are indeed linearly independent. Hence, we can continue the argument by replacing $L+1$ with J.

5.4 List Decodability of Random Concatenated Codes

In this section, we will look at the list decodability of concatenated codes when both the outer code and the inner codes are (independent) random linear codes.

The following is the main result of this section.

Theorem 5.1. *Let q be a prime power and let $0 < r < 1$ be an arbitrary rational. Let $0 < \varepsilon < \alpha_q(r)$ be an arbitrary real, where $\alpha_q(r)$ is as defined in (5.1), and $0 < R \leqslant (\alpha_q(r) - \varepsilon)/r$ be a rational. Then the following holds for large enough integers n, N such that there exist integers k and K that satisfy $k = rn$ and $K = RN$. Let C_{out} be a random linear code over $\mathbb{F}_{q^k}$ that is generated by a random $K \times N$ matrix over $\mathbb{F}_{q^k}$. Let $C_{in}^1, \ldots, C_{in}^N$ be random linear codes over $\mathbb{F}_q$, where C_{in}^i is generated by a random $k \times n$ matrix $\mathbf{G}_i$ and the random choices for $C_{out}, \mathbf{G}_1, \ldots, \mathbf{G}_N$ are all independent. Then the concatenated code $C = C_{out} \circ (C_{in}^1, \ldots, C_{in}^N)$ is a $\left(H_q^{-1}(1 - Rr) - \varepsilon, q^{O\left(\frac{rn}{\varepsilon^2(1-R)}\right)}\right)$-list decodable code with probability at least $1 - q^{-\Omega(nN)}$ over the choices of $C_{out}, \mathbf{G}_1, \ldots, \mathbf{G}_N$. Further, with high probability, C has rate rR.*

In the rest of this section, we will prove the above theorem.

Define $Q = q^k$. Let L be the worst-case list size that we are shooting for (we will fix its value at the end). The first observation is that any $L{+}1$-tuple of messages $(\mathbf{m}^1, \ldots, \mathbf{m}^{L+1}) \in (\mathbb{F}_Q^K)^{L+1}$ contains at least $J = \lceil \log_Q(L+1) \rceil$ many messages that are linearly independent over $\mathbb{F}_Q$. Thus, to prove the theorem it suffices to show that with high probability, no Hamming ball over $\mathbb{F}_q^{nN}$ of radius $(H_q^{-1}(1 - rR) - \varepsilon)nN$ contains a J-tuple of codewords $(C(\mathbf{m}^1), \ldots, C(\mathbf{m}^J))$, where $\mathbf{m}^1, \ldots, \mathbf{m}^J$ are linearly independent over $\mathbb{F}_Q$.

Define $\rho = H_q^{-1}(1 - Rr) - \varepsilon$. For every J-tuple of linearly independent messages $(\mathbf{m}^1, \ldots, \mathbf{m}^J) \in (\mathbb{F}_Q^K)^J$ and received word $\mathbf{y} \in \mathbb{F}_q^{nN}$, define an indicator random variable $\mathbf{I}(\mathbf{y}, \mathbf{m}^1, \ldots, \mathbf{m}^J)$ as follows. $\mathbf{I}(\mathbf{y}, \mathbf{m}^1, \ldots, \mathbf{m}^J) = 1$ if and only if for every $1 \leqslant j \leqslant J$, $wt(C(\mathbf{m}^j) - \mathbf{y}) \leqslant \rho nN$. That is, it captures the bad event that we want to avoid. Define

$$X_C = \sum_{\mathbf{y} \in \mathbb{F}_q^{nN}} \sum_{(\mathbf{m}^1, \ldots, \mathbf{m}^J) \in \mathrm{Ind}(Q,K,J)} \mathbf{I}(\mathbf{y}, \mathbf{m}^1, \ldots, \mathbf{m}^J)$$

where $\mathrm{Ind}(Q, K, J)$ denotes the collection of subsets of $\mathbb{F}_Q$-linearly independent vectors from $\mathbb{F}_Q^K$ of size J. We want to show that with high probability $X_C = 0$. By Markov's inequality, the theorem would follow if we can show that:

$$\mathbb{E}[X_C] = \sum_{\mathbf{y} \in \mathbb{F}_q^{nN}} \sum_{(\mathbf{m}^1, \ldots, \mathbf{m}^J) \in \mathrm{Ind}(Q,K,J)} \mathbb{E}[\mathbf{I}(\mathbf{y}, \mathbf{m}^1, \ldots, \mathbf{m}^J)] \text{ is } q^{-\Omega(nN)}. \tag{5.3}$$

Note that the number of distinct possibilities for $\mathbf{y}, \mathbf{m}^1, \ldots, \mathbf{m}^J$ is upper bounded by $q^{nN} \cdot Q^{RNJ} = q^{nN(1+rRJ)}$. Fix some arbitrary choice of $\mathbf{y}, \mathbf{m}^1, \ldots, \mathbf{m}^J$. To prove (5.3), we will show that

$$q^{nN(1+rRJ)} \cdot \mathbb{E}[\mathbf{I}(\mathbf{y}, \mathbf{m}^1, \ldots, \mathbf{m}^J)] \text{ is } q^{-\Omega(nN)}. \tag{5.4}$$

Before we proceed, we need some more notation. Given vectors $\mathbf{u}^1, \ldots, \mathbf{u}^J \in \mathbb{F}_Q^N$, we define $\mathbf{Z}(\mathbf{u}^1, \ldots, \mathbf{u}^J) = (Z_1, \ldots, Z_N)$ as follows. For every $1 \leqslant i \leqslant N$, $Z_i \subseteq [J]$ denotes the largest subset such that the elements $(u_i^j)_{j \in Z_i}$ are linearly independent over $\mathbb{F}_q$ (in case of a tie choose the lexically first such set), where $\mathbf{u}^j = (u_1^j, \ldots, u_N^j)$. A subset of $\mathbb{F}_Q$ is linearly independent over $\mathbb{F}_q$ if its elements, when viewed as vectors from $\mathbb{F}_q^k$ (recall that $\mathbb{F}_{q^k}$ is isomorphic to $\mathbb{F}_q^k$) are linearly independent over $\mathbb{F}_q$. If $u_i^j \in Z_i$ then we will call u_i^j a *good* symbol. Note that a good symbol is always non-zero. We will also define another partition of all the good symbols, $\mathbf{T}(\mathbf{u}^1, \ldots, \mathbf{u}^J) = (T_1, \ldots, T_J)$ by setting $T_j = \{i | j \in Z_i\}$ for $1 \leqslant j \leqslant J$.

Since $\mathbf{m}^1, \ldots, \mathbf{m}^J$ are linearly independent over $\mathbb{F}_Q$, the corresponding codewords in C_{out} are distributed uniformly in $\mathbb{F}_Q^N$. In other words, for any fixed $(\mathbf{u}^1, \ldots, \mathbf{u}^J) \in (\mathbb{F}_Q^N)^J$,

$$\Pr_{C_{out}}\left[\bigwedge_{j=1}^{J} C_{out}(\mathbf{m}^j) = \mathbf{u}^j\right] = Q^{-NJ} = q^{-rnNJ}. \tag{5.5}$$

Recall that we denote the (random) generator matrices for the inner code C_{in}^i by $\mathbf{G}_i$ for every $1 \leqslant i \leqslant N$. Also note that every $(\mathbf{u}^1, \ldots, \mathbf{u}^J) \in (\mathbb{F}_Q^N)^J$ has a unique $\mathbf{Z}(\mathbf{u}^1, \ldots, \mathbf{u}^J)$. In other words, the 2^{NJ} choices of $\mathbf{Z}$ partition the tuples in $(\mathbb{F}_Q^N)^J$.

Let $h = \rho n N$. Consider the following calculation (where the dependence of $\mathbf{Z}$ and $\mathbf{T}$ on $\mathbf{u}^1, \ldots, \mathbf{u}^J$ have been suppressed for clarity):

$$\mathbb{E}[\mathbf{I}(\mathbf{y}, \mathbf{m}^1, \ldots, \mathbf{m}^J)] = \sum_{(\mathbf{u}^1,\ldots,\mathbf{u}^J)\in(\mathbb{F}_Q^N)^J} \Pr_{\mathbf{G}=(\mathbf{G}_1,\ldots,\mathbf{G}_N)}\left[\bigwedge_{j=1}^{J} wt(\mathbf{u}^j\mathbf{G} - \mathbf{y}) \leqslant h\right] \tag{5.6}$$

$$\cdot \Pr_{C_{out}}\left[\bigwedge_{j=1}^{J} C_{out}(\mathbf{m}^j) = \mathbf{u}^j\right]$$

$$= q^{-rnNJ} \sum_{(\mathbf{u}^1,\ldots,\mathbf{u}^J)\in(\mathbb{F}_Q^N)^J} \Pr_{\mathbf{G}=(\mathbf{G}_1,\ldots,\mathbf{G}_N)}\left[\bigwedge_{j=1}^{J} wt(\mathbf{u}^j\mathbf{G} - \mathbf{y}) \leqslant h\right] \tag{5.7}$$

$$\leqslant q^{-rnNJ} \sum_{(\mathbf{u}^1,\ldots,\mathbf{u}^J)\in(\mathbb{F}_Q^N)^J} \Pr_{\mathbf{G}=(\mathbf{G}_1,\ldots,\mathbf{G}_N)}\left[\bigwedge_{j=1}^{J} wt_{T_j}(\mathbf{u}^j\mathbf{G} - \mathbf{y}) \leqslant h\right] \tag{5.8}$$

$$= q^{-rnNJ} \sum_{(\mathbf{u}^1,\ldots,\mathbf{u}^J)\in(\mathbb{F}_Q^N)^J} \prod_{j=1}^{J} \Pr_{\mathbf{G}}\left[wt_{T_j}(\mathbf{u}^j\mathbf{G} - \mathbf{y}) \leqslant h\right] \tag{5.9}$$

In the above (5.6) follows from the fact that the (random) choices for C_{out} and $\mathbf{G} = (\mathbf{G}_1, \ldots, \mathbf{G}_N)$ are all independent. (5.7) follows from (5.5). (5.8) follows from the simple fact that for every $\mathbf{y} \in (\mathbb{F}_q^n)^N$ and $T \subseteq [N]$, $wt_T(\mathbf{y}) \leqslant wt(\mathbf{y})$. (5.9) follows from the subsequent argument. By definition of conditional probability, $\Pr_{\mathbf{G}}\left[\bigwedge_{j=1}^{J} wt_{T_j}(\mathbf{u}^j\mathbf{G}-\mathbf{y}) \leqslant h\right]$ is the same as

$$\Pr_{\mathbf{G}}\left[wt_{T_J}(\mathbf{u}^J\mathbf{G}-\mathbf{y}) \leqslant h | \bigwedge_{j=1}^{J-1} wt_{T_j}(\mathbf{u}^j\mathbf{G}-\mathbf{y}) \leqslant h\right] \cdot \Pr_{\mathbf{G}}\left[\bigwedge_{j=1}^{J-1} wt_{T_j}(\mathbf{u}^j\mathbf{G}-\mathbf{y}) \leqslant h\right].$$

Now as all symbols corresponding to T_J are good symbols, for every $i \in T_J$, the value of $\mathbf{u}_i^J\mathbf{G}_i$ is independent of the values of $\{\mathbf{u}_i^1\mathbf{G}_i, \ldots, \mathbf{u}_i^{J-1}\mathbf{G}_i\}$. Further since each of $\mathbf{G}_1, \ldots, \mathbf{G}_N$ are chosen independently (at random), the event $wt_{T_J}(\mathbf{u}^J\mathbf{G}-\mathbf{y}) \leqslant h$ is independent of the event $\bigwedge_{j=1}^{J-1} wt_{T_j}(\mathbf{u}^j\mathbf{G}-\mathbf{y}) \leqslant h$. Thus, $\Pr_{\mathbf{G}}\left[\bigwedge_{j=1}^{J} wt_{T_j}(\mathbf{u}^j\mathbf{G}-\mathbf{y}) \leqslant h\right]$ is

$$\Pr_{\mathbf{G}}\left[wt_{T_J}(\mathbf{u}^J\mathbf{G}-\mathbf{y}) \leqslant h\right] \cdot \Pr_{\mathbf{G}}\left[\bigwedge_{j=1}^{J-1} wt_{T_j}(\mathbf{u}^j\mathbf{G}-\mathbf{y}) \leqslant h\right].$$

Inductively applying the argument above gives (5.9).

Further,

$$\begin{aligned}
\mathbb{E}[\mathbf{I}(\mathbf{y}, \mathbf{m}^1, \ldots, \mathbf{m}^J)] &= \sum_{(\mathbf{u}^1,\ldots,\mathbf{u}^J)\in(\mathbb{F}_Q^N)^J} \prod_{j=1}^{J} q^{-rnN} \cdot \Pr_{\mathbf{G}}\left[wt_{T_j}(\mathbf{u}^j\mathbf{G}-\mathbf{y}) \leqslant h\right] && (5.10)\\
&= \sum_{(d_1,\ldots,d_J)\in\{0,\ldots,N\}^J} \sum_{\substack{(\mathbf{u}^1,\ldots,\mathbf{u}^J)\in(\mathbb{F}_Q^N)^J,\\ (|T_1|=d_1,\ldots,|T_J|=d_J)}} \prod_{j=1}^{J} \frac{\Pr_{\mathbf{G}}\left[wt_{T_j}(\mathbf{u}^j\mathbf{G}-\mathbf{y}) \leqslant h\right]}{q^{rnN}} && (5.11)\\
&\leqslant \sum_{\substack{(d_1,\ldots,d_J)\\ \in\{0,\ldots,N\}^J}} q^{JN+(rn+J)\sum_{j=1}^{J} d_j} \prod_{\substack{j=1,\\ |T_j|=d_j}}^{J} \frac{\Pr_{\mathbf{G}}\left[wt_{T_j}(\mathbf{u}^j\mathbf{G}-\mathbf{y}) \leqslant h\right]}{q^{rnN}} && (5.12)\\
&= \sum_{(d_1,\ldots,d_J)\in\{0,\ldots,N\}^J} \prod_{\substack{j=1,\\ |T_j|=d_j}}^{J} \frac{\Pr_{\mathbf{G}}\left[wt_{T_j}(\mathbf{u}^j\mathbf{G}-\mathbf{y}) \leqslant h\right]}{q^{n\left(-r(d_j-N)-\frac{Jd_j}{n}-\frac{N}{n}\right)}} && (5.13)
\end{aligned}$$

In the above (5.10), (5.11), (5.13) follow from rearranging and grouping the summands. (5.12) uses the following argument. Given a fixed $\mathbf{Z} = (Z_1, \ldots, Z_N)$, the number of tuples $(\mathbf{u}^1, \ldots, \mathbf{u}^J)$ such that $\mathbf{Z}(\mathbf{u}^1, \ldots, \mathbf{u}^J) = \mathbf{Z}$ is at most $U = \prod_{i=1}^{N} q^{|Z_i|k} \cdot q^{|Z_i|(J-|Z_i|)}$, where the $q^{|Z_i|k}$ is an upper bound on the number of $|Z_i|$ linearly independent vectors from $\mathbb{F}_q^k$ and $q^{|Z_i|(J-|Z_i|)}$ follows from the fact that every bad symbol $\{u_i^j\}_{j\notin Z_i}$ has to take a value that is a linear combination of the symbols $\{u_i^j\}_{j \in Z_i}$. Now $U \leqslant \prod_{i=1}^N q^{|Z_i|(k+J)} = q^{(k+J)\sum_{i=1}^N |Z_i|} = q^{(k+J)\sum_{j=1}^J |T_j|}$. Finally, there are $2^{JN} \leqslant q^{JN}$ distinct choices for $\mathbf{Z}$.

(5.13) implies the following

$$q^{nN(1+rRJ)} \cdot \mathbb{E}[\mathbf{I}(\mathbf{y}, \mathbf{m}^1, \ldots, \mathbf{m}^J)] \leqslant \sum_{(d_1,\ldots,d_J) \in \{0,\ldots,N\}^J} \prod_{j=1}^J E_j$$

where

$$E_j = q^{-n\left(-r(d_j - N(1-R)) - \frac{N}{J} - \frac{Jd_j}{n} - \frac{N}{n}\right)} \cdot \Pr_{\mathbf{G}}\left[wt_{T_j}(\mathbf{u}_j\mathbf{G} - \mathbf{y}) \leqslant h\right].$$

We now proceed to upper bound E_j by $q^{-\Omega(nN)}$ for every $1 \leqslant j \leqslant J$. Note that this will imply the claimed result as there are at most $(N+1)^J = q^{o(nN)}$ choices for different values of d_j's.

We first start with the case when $d_j < d^*$, where

$$d^* = N(1 - R - \gamma),$$

for some parameter $0 < \gamma < 1 - R$ to be defined later. In this case we use the fact that $\Pr_{\mathbf{G}}\left[wt_{T_j}(\mathbf{u}_j\mathbf{G} - \mathbf{y}) \leqslant h\right] \leqslant 1$. Thus, we would be done if we can show that

$$\frac{1}{N}\left(r\left(d_j - N(1-R)\right) + \frac{N}{J} + \frac{Jd_i}{n} + \frac{N}{n}\right) \leqslant -\delta' < 0,$$

for some $\delta' > 0$ that we will choose soon. The above would be satisfied if

$$\frac{d_j}{N} < (1-R) - \frac{1}{r}\left(\frac{1}{J} + \frac{Jd_j}{nN} + \frac{1}{n}\right) - \frac{\delta'}{r},$$

which is satisfied if we choose $\gamma > \frac{2}{r}\left(\frac{1}{J} + \frac{Jd_j}{nN} + \frac{1}{n}\right) + \frac{\delta'}{r}$ as $d_j < d^*$. Note that if $n > 2J\left(\frac{Jd_j}{N} + 1\right)$ and if we set $\delta' = \frac{1}{J}$, it is enough to choose $\gamma = \frac{4}{Jr}$.

We now turn our attention to the case when $d_j \geqslant d^*$. The arguments are very similar to the ones employed by Thommesen in the proof of his main theorem in [102]. In this case, by Lemma 5.1 we have

$$E_j \leqslant q^{-nd_j\left(1 - H_q\left(\frac{h}{nd_j}\right) - r\left(1 - \frac{N(1-R)}{d_j}\right) - \frac{N}{d_jJ} - \frac{J}{n} - \frac{N}{nd_j}\right)}.$$

The above implies that we can show that E_j is $q^{-\Omega(nN(1-R-\gamma))}$ provided we show that for every $d^* \leqslant d \leqslant N$,

$$h/(nd) \leqslant H_q^{-1}\left(1 - r\left(1 - \frac{N(1-R)}{d}\right) - \frac{N}{dJ} - \frac{J}{n} - \frac{N}{nd}\right) - \delta,$$

for $\delta = \varepsilon/3$. Now if $n \geqslant 2J^2$, then both $\frac{J}{n} \leqslant \frac{N}{2Jd}$ and $\frac{N}{nd} \leqslant \frac{N}{2Jd}$. In other words, $\frac{J}{n} + \frac{N}{nd} \leqslant \frac{N}{Jd}$. Using the fact that H_q^{-1} is increasing, the above is satisfied if

$$h/(nd) \leqslant H_q^{-1}\left(1 - r\left(1 - \frac{N(1-R-\gamma)}{d}\right) - \frac{2N}{dJ}\right) - \delta,$$

By Lemma 5.4, as long as $J \geqslant 4c'_q/(\delta^2(1-R))$ (and the conditions on γ are satisfied), the above can be satisfied by picking

$$h/(nN) = H_q^{-1}(1 - rR) - 3\delta = \rho,$$

as required. We now verify that the conditions on γ in Lemma 5.4 are satisfied by our choice of $\gamma = \frac{4}{Jr}$. Note that if we choose $J = 4c'_q/(\delta^2(1-R))$, we will have $\gamma = \frac{\delta^2(1-R)}{c'_q r}$. Now, as $R < 1$, we also have $\gamma \leqslant \delta^2/(rc'_q)$. Finally, we show that $\gamma \leqslant (1-R)/2$. Indeed

$$\gamma = \frac{\delta^2(1-R)}{c'_q r} = \frac{\varepsilon^2(1-R)}{9c'_q r} \leqslant \frac{\varepsilon(1-R)}{9r} \leqslant \frac{\alpha_q(r)(1-R)}{9r} < \frac{1-R}{2},$$

where the first inequality follows from the facts that $c'_q \geqslant 1$ and $\varepsilon \leqslant 1$. The second inequality follows from the assumption on ε. The third inequality follows from Lemma 5.2.

Note that $J = O\left(\frac{1}{(1-R)\varepsilon^2}\right)$, which implies $L = Q^{O(1/((1-R)\varepsilon^2)}$ as claimed in the statement of the theorem.

We still need to argue that with high probability the rate of the code $C = C_{out} \circ (C_{in}^1, \ldots, C_{in}^N)$ is rR. One way to argue this would be to show that with high probability all of the generator matrices have full rank. However, this is not the case: in fact, with some non-negligible probability at least one of them will not have full rank. However, we claim that with high probability C has distance > 0. Note that as C is a linear code, this implies that for every distinct pair of messages $\mathbf{m}^1 \neq \mathbf{m}^2 \in \mathbb{F}_Q^K$ are mapped to distinct codewords, which implies that C has q^{rnRN} codewords, as desired. We now briefly argue why C has distance > 0. The proof above in fact implies that with high probability C has distance about $H_q^{-1}(1 - rR)nN$. It is easy to see that to show that C has distance at least h, it is enough to show that with high probability $\sum_{\mathbf{m} \in \mathbb{F}_Q^K} \mathbf{I}(\mathbf{0}, \mathbf{m}) = 0$. Note that this is a special case of our proof, with $J = 1$ and $\mathbf{y} = \mathbf{0}$ and hence, with probability at least $1 - q^{\Omega(nN)}$, the code C has large distance. The proof is complete.

Remark 5.1. *In a typical use of concatenated codes, the block lengths of the inner and outer codes satisfy $n = \Theta(\log N)$, in which case the concatenated code of Theorem 5.1 is list decodable with lists of size $N^{O\left(\varepsilon^{-2}(1-R)^{-1}\right)}$. However, the proof of Theorem 5.1 also works with smaller n. In particular as long as n is at least $3J^2$, the proof of Theorem 5.1 goes through. Thus, with n in $\Theta(J^2)$, one can get concatenated codes that are list decodable up to the list-decoding capacity with lists of size $q^{O\left(\varepsilon^{-6}(1-R)^{-3}\right)}$.*

Lemma 5.4. *Let q be a prime power, and $1 \leqslant n \leqslant N$ be integers. Let $0 < r, R < 1$ be rationals and $\delta > 0$ be a real such that $R \leqslant (\alpha_q(r) - \delta)/r$ and $\delta \leqslant \alpha_q(r)$, where $\alpha_q(r)$ is as defined in (5.1). Let $\gamma > 0$ be a real such that $\gamma \leqslant \min\left(\frac{1-R}{2}, \frac{\delta^2}{c'_q r}\right)$, where c'_q is the constant that depends only on q from Lemma 2.4. Then for all integers $J \geqslant \frac{4c'_q}{\delta^2(1-R)}$ and $h \leqslant (H_q^{-1}(1-rR) - 2\delta)nN$ the following is satisfied. For every integer $(1-R-\gamma)N \leqslant d \leqslant N$,*

$$\frac{h}{nd} \leqslant H_q^{-1}\left(1 - r\left(1 - \frac{N(1-R-\gamma)}{d}\right) - \frac{2N}{Jd}\right). \tag{5.14}$$

Proof. Using the fact H_q^{-1} is an increasing function, (5.14) is satisfied if the following is true (where $d^* = (1-R-\gamma)N$):

$$\frac{h}{nN} \leqslant \min_{d^* \leqslant d \leqslant N}\left\{\left(\frac{d}{N}\right) \cdot H_q^{-1}\left(1 - r\left(1 - \frac{N(1-R-\gamma)}{d}\right) - \frac{2N}{d^* J}\right)\right\}.$$

Define a new variable $\theta = 1 - N(1-R-\gamma)/d$. Note that as $d^* = (1-R-\gamma)N \leqslant d \leqslant N$, $0 \leqslant \theta \leqslant R + \gamma$. Also $d/N = (1-R-\gamma)(1-\theta)^{-1}$. Thus, the above inequality would be satisfied if

$$\frac{h}{nN} \leqslant (1-R-\gamma) \min_{0 \leqslant \theta \leqslant R+\gamma}\left\{(1-\theta)^{-1} H_q^{-1}\left(1 - r\theta - \frac{2}{(1-R-\gamma)J}\right)\right\}.$$

Again using the fact that H_q^{-1} is an increasing function along with the fact that $\gamma \leqslant (1-R)/2$, we get that the above is satisfied if

$$\frac{h}{nN} \leqslant (1-R-\gamma) \min_{0 \leqslant \theta \leqslant R+\gamma}\left\{(1-\theta)^{-1} H_q^{-1}\left(1 - r\theta - \frac{4}{(1-R)J}\right)\right\}.$$

By Lemma 2.4, if $J \geqslant \frac{4c'_q}{\delta^2(1-R)}$, then[7] $H_q^{-1}\left(1 - r\theta - \frac{4}{(1-R)J}\right) \geqslant H_q^{-1}(1-r\theta) - \delta$. Since for every $0 \leqslant \theta \leqslant R+\gamma$, $(1-R-\gamma)(1-\theta)^{-1}\delta \leqslant \delta$, the above equation would be satisfied if

$$\frac{h}{nN} \leqslant (1-R-\gamma) \min_{0 < \theta \leqslant R+\gamma} f_{r,q}(\theta) - \delta.$$

Note that the assumptions $\gamma \leqslant \delta^2/(rc'_q) \leqslant \delta/r$ (as $\delta \leqslant 1$ and $c'_q \geqslant 1$) and $R \leqslant (\alpha_q(r) - \delta)/r$, we have $R + \gamma \leqslant \alpha_q(r)/r$. Thus, by using Lemma 5.3 we get that $(1-R-\gamma)\min_{0<\theta\leqslant R+\gamma} f_{r,q}(\theta) = H_q^{-1}(1 - rR - r\gamma)$. By Lemma 2.4, the facts that $\gamma \leqslant \delta^2/(rc'_q)$ and H_q^{-1} is increasing, we have $H_q^{-1}(1-rR-r\gamma) \geqslant H_q^{-1}(1-rR) - \delta$. This implies that (5.14) is satisfied if $h/(nN) \leqslant H_q^{-1}(1-rR) - 2\delta$, as desired. □

[7] We also use the fact that H_q^{-1} is increasing.

5.5 Using Folded Reed-Solomon Code as Outer Code

In this section, we will prove a result similar to Theorem 5.1, with the outer code being the folded Reed-Solomon code from Chapter 3. The proof will make crucial use of the list recoverability of folded Reed-Solomon codes. Before we begin we will need the following definition and results.

5.5.1 Preliminaries

We will need the following notion of independence.

Definition 5.1 (Independent tuples). *Let C be a code of block length N and rate R defined over $\mathbb{F}_{q^k}$. Let $J \geqslant 1$ and $0 \leqslant d_1, \ldots, d_J \leqslant N$ be integers. Let $\mathbf{d} = \langle d_1, \ldots, d_J \rangle$. An ordered tuple of codewords $(c^1, \ldots, c^J)$, $c^j \in C$ is said to be $(\mathbf{d}, \mathbb{F}_q)$-independent if the following holds. $d_1 = wt(c^1)$ and for every $1 < j \leqslant J$, d_j is the number of positions i such that c_i^j is $\mathbb{F}_q$-independent of the vectors $\{c_i^1, \ldots, c_i^{j-1}\}$, where $c^\ell = (c_1^\ell, \ldots, c_N^\ell)$.*

Note that for any tuple of codewords $(c^1, \ldots, c^J)$ there exists a unique $\mathbf{d}$ such that it is $(\mathbf{d}, \mathbb{F}_q)$-independent.

The next result will be crucial in our proof.

Lemma 5.5. *Let C be a folded Reed-Solomon code of block length N that is defined over $\mathbb{F}_Q$ with $Q = q^k$ as guaranteed by Theorem 3.6. For any L-tuple of codewords from C, where $L \geqslant J \cdot (N/\varepsilon^2)^{O\left(\varepsilon^{-1} J \log(q/R)\right)}$ (where $\varepsilon > 0$ is same as the one in Theorem 3.6), there exists a sub-tuple of J codewords such that the J-tuple is $(\mathbf{d}, \mathbb{F}_q)$-independent, where $\mathbf{d} = \langle d_1, \ldots, d_J \rangle$ such that for every $1 \leqslant j \leqslant J$, $d_j \geqslant (1 - R - \varepsilon)N$.*

Proof. The proof is constructive. In particular, given an L-tuple of codewords, we will construct a J sub-tuple with the required property. The correctness of the procedure will hinge on the list recoverability of the folded Reed-Solomon code as guaranteed by Theorem 3.6.

We will construct the final sub-tuple iteratively. In the first step, pick any non-zero codeword in the L-tuple– call it c^1. Note that as C has distance $(1 - R)N$ (and $\mathbf{0} \in C$), c^1 is non-zero in at least $d_1 \geqslant (1 - R)N > (1 - R - \varepsilon)N$ many places. Note that c^1 is vacuously independent of the "previous" codewords in these positions. Now, say that the procedure has chosen codewords $c^1, \ldots, c^s$ such that the tuple is $(\mathbf{d}', \mathbb{F}_q)$-independent for $\mathbf{d}' = \langle d_1, \ldots, d_s \rangle$, where for every $1 \leqslant j \leqslant s$, $d_j \geqslant (1 - R - \varepsilon)N$. For every $1 \leqslant i \leqslant N$, define S_i to be the $\mathbb{F}_q$-span of the vectors $\{c_i^1, \ldots, c_i^s\}$ in $\mathbb{F}_q^k$. Note that $|S_i| \leqslant q^s$. Call $c = (c_1, \ldots, c_N) \in C$ to be a *bad* codeword, if there does not exist any $d_{s+1} \geqslant (1 - R - \varepsilon)N$ such that $(c^1, \ldots, c^s, c)$ is $(\mathbf{d}, \mathbb{F}_q)$-independent for $\mathbf{d} = \langle d_1, \ldots, d_{s+1} \rangle$. In other words, c is a bad codeword if and only if some $T \subset [N]$ with $|T| = (R + \varepsilon)N$ satisfies $c_i \in S_i$ for every $i \in T$. Put differently, c satisfies the condition of being in the output list for list recovering C with input $S_1, \ldots, S_N$ and agreement fraction $R + \varepsilon$. Thus, by Theorem 3.6, the number of such bad codewords is $U = (N/\varepsilon^2)^{O\left(\varepsilon^{-1} s \log(q/R)\right)} \leqslant (N/\varepsilon^2)^{O\left(\varepsilon^{-1} J \log(q/R)\right)}$, where J is the number of steps for which this greedy procedure can be applied. Thus, as long as at each step there are strictly more than U codewords from the original L-tuple of codewords left, we can continue this greedy procedure. Note that we can continue this procedure J times, as long as $J \leqslant L/U$. The proof is complete. □

Finally, we will need a bound on the number of independent tuples for folded Reed-Solomon codes.

Lemma 5.6. *Let C be a folded Reed-Solomon code of block length N and rate $0 < R < 1$ that is defined over $\mathbb{F}_Q$, where $Q = q^k$. Let $J \geqslant 1$ and $0 \leqslant d_1, \ldots, d_J \leqslant N$ be integers and define $\mathbf{d} = \langle d_1, \ldots, d_J \rangle$. Then the number of $(\mathbf{d}, \mathbb{F}_q)$-independent tuples in C is at most*

$$q^{NJ(J+1)} \prod_{j=1}^{J} Q^{\max(d_j - N(1-R)+1, 0)}.$$

Proof. Given a tuple $(c^1, \ldots, c^J)$ that is $(\mathbf{d}, \mathbb{F}_q)$-independent, define $T_j \subseteq [N]$ with $|T_j| = d_j$, for $1 \leqslant j \leqslant J$ to be the set of positions i, where c_i^j is linearly independent of the values $\{c_i^1, \ldots, c_i^{j-1}\}$. We will estimate the number of $(\mathbf{d}, \mathbb{F}_q)$-independent tuples by first estimating a bound U_j on the number of choices for the j^{th} codeword in the tuple (given a fixed choice of the first $j-1$ codewords). To complete the proof, we will show that

$$U_j \leqslant q^{N(J+1)} \cdot Q^{\max(d_j - N(1-R)+1, 0)}.$$

A codeword $c \in C$ can be the j^{th} codeword in the tuple in the following way. Now for every position in $[N] \setminus T_j$, c can take at most $q^{j-1} \leqslant q^J$ values (as in these position the value has to lie in the $\mathbb{F}_q$ span of the values of the first $j-1$ codewords in that position). Since C is folded Reed-Solomon, once we fix the values at positions in $[N] \setminus T_j$, the codeword will be completely determined once any $\max(RN - (N - d_j) + 1, 0) = \max(d_j - N(1-R) + 1, 0)$ positions in T_j are chosen (w.l.o.g. assume that they are the "first" so many positions). The number of choices for T_j is $\binom{N}{d_j} \leqslant 2^N \leqslant q^N$. Thus, we have

$$U_j \leqslant q^N \cdot (q^J)^{N-d_j} \cdot Q^{\max(d_j - N(1-R)+1, 0)} \leqslant q^{N(J+1)} \cdot Q^{\max(d_j - N(1-R)+1), 0)},$$

as desired. □

5.5.2 The Main Result

We will now prove the following result.

Theorem 5.2. *Let q be a prime power and let $0 < r < 1$ be an arbitrary rational. Let $0 < \varepsilon < \alpha_q(r)$ an arbitrary real, where $\alpha_q(r)$ is as defined in (5.1), and $0 < R \leqslant (\alpha_q(r) - \varepsilon)/r$ be a rational. Then the following holds for large enough integers n, N such that there exist integers k and K that satisfy $k = rn$ and $K = RN$. Let C_{out} be a folded Reed-Solomon* *code over $\mathbb{F}_{q^k}$ of block length N and rate R. Let $C_{in}^1, \ldots, C_{in}^N$ be random linear codes over $\mathbb{F}_q$, where C_{in}^i is generated by a random $k \times n$ matrix $\mathbf{G}_i$ over $\mathbb{F}_q$ and the random choices for $\mathbf{G}_1, \ldots, \mathbf{G}_N$ are all independent. Then the concatenated code $C = C_{out} \circ (C_{in}^1, \ldots, C_{in}^N)$ is a $\left(H_q^{-1}(1 - Rr) - \varepsilon, \left(\frac{N}{\varepsilon^2}\right)^{O\left(\varepsilon^{-4}(1-R)^{-2}\log(1/R)\right)}\right)$-list decodable code with probability at least $1 - q^{-\Omega(nN)}$ over the choices of $\mathbf{G}_1, \ldots, \mathbf{G}_N$. Further, with high probability, C has rate rR.*

In the rest of this section, we will prove the above theorem.

Define $Q = q^k$. Let L be the worst-case list size that we are shooting for (we will fix its value at the end). By Lemma 5.5, any $L+1$-tuple of C_{out} codewords $(\mathbf{u}^0, \ldots, \mathbf{u}^L) \in (C_{out})^{L+1}$ contains at least $J = \left\lfloor (L+1)/(N/\gamma^2)^{O\left(\gamma^{-1}J\log(q/R)\right)} \right\rfloor$ codewords that form an $(\mathbf{d}, \mathbb{F}_q)$-independent tuple, for some $\mathbf{d} = \langle d_1, \ldots, d_J \rangle$, with $d_j \geqslant (1 - R - \gamma)N$ (we will specify γ, $0 < \gamma < 1 - R$, later). Thus, to prove the theorem it suffices to show that with high probability, no Hamming ball in $\mathbb{F}_q^{nN}$ of radius $(H_q^{-1}(1 - rR) - \varepsilon)nN$ contains a J-tuple of codewords $(\mathbf{u}^1\mathbf{G}, \ldots, \mathbf{u}^J\mathbf{G})$, where $(\mathbf{u}^1, \ldots, \mathbf{u}^J)$ is a J-tuple of folded Reed-Solomon codewords that is $(\mathbf{d}, \mathbb{F}_q)$-independent. For the rest of the proof, we will call a J-tuple of C_{out} codewords $(\mathbf{u}^1, \ldots, \mathbf{u}^J)$ a *good* tuple if it is $(\mathbf{d}, \mathbb{F}_q)$-independent for some $\mathbf{d} = \langle d_1, \ldots, d_J \rangle$, where $d_j \geqslant (1 - R - \gamma)N$ for every $1 \leqslant j \leqslant J$.

Define $\rho = H_q^{-1}(1 - Rr) - \varepsilon$. For every good J-tuple of C_{out} codewords $(\mathbf{u}^1, \ldots, \mathbf{u}^J)$ and received word $\mathbf{y} \in \mathbb{F}_q^{nN}$, define an indicator variable $\mathbf{I}(\mathbf{y}, \mathbf{u}^1, \ldots, \mathbf{u}^J)$ as follows. $\mathbf{I}(\mathbf{y}, \mathbf{u}^1, \ldots, \mathbf{u}^J) = 1$ if and only if for every $1 \leqslant j \leqslant J$, $wt(\mathbf{u}^j\mathbf{G} - \mathbf{y}) \leqslant \rho nN$. That is, it captures the bad event that we want to avoid. Define

$$X_C = \sum_{\mathbf{y} \in \mathbb{F}_q^{nN}} \sum_{\text{good } (\mathbf{u}^1, \ldots, \mathbf{u}^J) \in (C_{out})^J} \mathbf{I}(\mathbf{y}, \mathbf{u}^1, \ldots, \mathbf{u}^J).$$

We want to show that with high probability $X_C = 0$. By Markov's inequality, the theorem would follow if we can show that:

$$\mathbb{E}[X_C] = \sum_{\mathbf{y} \in \mathbb{F}_q^{nN}} \sum_{\text{good } (\mathbf{u}^1, \ldots, \mathbf{u}^J) \in (C_{out})^J} \mathbb{E}[\mathbf{I}(\mathbf{y}, \mathbf{u}^1, \ldots, \mathbf{u}^J)] \leqslant q^{-\Omega(nN)}. \tag{5.15}$$

Before we proceed, we need a final bit of notation. For a good tuple $(\mathbf{u}^1, \ldots, \mathbf{u}^J)$ and every $1 \leqslant j \leqslant J$, define $T_j(\mathbf{u}^1, \ldots, \mathbf{u}^J) \subseteq [N]$ to be the set of positions i such that $\mathbf{u}_i^j$ is $\mathbb{F}_q$-independent of the set $\{\mathbf{u}_i^1, \ldots, \mathbf{u}_i^{j-1}\}$. Note that since the tuple is good, $|T_j(\mathbf{u}^1, \ldots, \mathbf{u}^J)| \geqslant (1 - R - \gamma)N$.

Let $h = \rho nN$. Consider the following sequence of inequalities (where below we have suppressed the dependence of T_j on $(\mathbf{u}^1, \ldots, \mathbf{u}^J)$ for clarity):

$$\mathbb{E}[X_C] = \sum_{\mathbf{y} \in \mathbb{F}_q^{nN}} \sum_{\text{good } (\mathbf{u}^1, \ldots, \mathbf{u}^J) \in (C_{out})^J} \Pr_{\mathbf{G} = (\mathbf{G}_1, \ldots, \mathbf{G}_N)} \left[\bigwedge_{j=1}^{J} wt(\mathbf{u}^j\mathbf{G} - \mathbf{y}) \leqslant h \right] \tag{5.16}$$

$$\leqslant \sum_{\mathbf{y}\in\mathbb{F}_q^{nN}} \sum_{\text{good } (\mathbf{u}^1,\ldots,\mathbf{u}^J)\in(C_{out})^J} \Pr_{\mathbf{G}=(\mathbf{G}_1,\ldots,\mathbf{G}_N)}\left[\bigwedge_{j=1}^{J} wt_{T_j}(\mathbf{u}^j\mathbf{G}-\mathbf{y})\leqslant h\right] \tag{5.17}$$

$$= \sum_{\mathbf{y}\in\mathbb{F}_q^{nN}} \sum_{\text{good } (\mathbf{u}^1,\ldots,\mathbf{u}^J)\in(C_{out})^J} \prod_{j=1}^{J} \Pr_{\mathbf{G}}\left[wt_{T_j}(\mathbf{u}^i\mathbf{G}-\mathbf{y})\leqslant h\right] \tag{5.18}$$

$$\leqslant \sum_{\mathbf{y}\in\mathbb{F}_q^{nN}} \sum_{\text{good } (\mathbf{u}^1,\ldots,\mathbf{u}^J)\in(C_{out})^J} \prod_{j=1}^{J} q^{-n|T_j|\left(1-H_q\left(\frac{h}{n|T_j|}\right)\right)} \tag{5.19}$$

$$= \sum_{\mathbf{y}\in\mathbb{F}_q^{nN}} \sum_{\substack{(d_1,\ldots,d_J)\\ \in\{(1-R-\gamma)N,\ldots,N\}^J}} \sum_{\substack{\text{good } (\mathbf{u}^1,\ldots,\mathbf{u}^J)\in(C_{out})^J,\\ (|T_1|=d_1,\ldots,|T_J|=d_J)}} \prod_{j=1}^{J} q^{-nd_j\left(1-H_q\left(\frac{h}{nd_j}\right)\right)} \tag{5.20}$$

$$\leqslant \sum_{\substack{(d_1,\ldots,d_J)\\ \in\{(1-R-\gamma)N,\ldots,N\}^J}} q^{nN}\cdot q^{NJ(J+1)} \prod_{j=1}^{J} Q^{\max(d_j-(1-R)N+1,0)} \prod_{j=1}^{J} q^{-nd_j\left(1-H_q\left(\frac{h}{nd_j}\right)\right)} \tag{5.21}$$

$$\leqslant \sum_{\substack{(d_1,\ldots,d_J)\\ \in\{(1-R-\gamma)N,\ldots,N\}^J}} q^{nN}\cdot q^{NJ(J+1)} \prod_{j=1}^{J} Q^{d_j-(1-R-\gamma)N} \prod_{j=1}^{J} q^{-nd_j\left(1-H_q\left(\frac{h}{nd_j}\right)\right)} \tag{5.22}$$

$$= \sum_{\substack{(d_1,\ldots,d_J)\\ \in\{(1-R-\gamma)N,\ldots,N\}^J}} \prod_{j=1}^{J} q^{-nd_j\left(1-H_q\left(\frac{h}{nd_j}\right)-r\left(1-\frac{(1-R-\gamma)N}{d_j}\right)-\frac{N}{Jd_j}-\frac{N(J+1)}{nd_j}\right)}. \tag{5.23}$$

In the above (5.16) follows from the definition of the indicator variable. (5.17) follows from the simple fact that for every vector $\mathbf{u}$ of length N and every $T\subseteq[N]$, $wt_T(\mathbf{u})\leqslant wt(\mathbf{u})$. (5.18) follows by an argument similar to the one used to argue (5.9) from (5.8) in the proof of Theorem 5.1. Basically, we need to write out the probability as a product of conditional probabilities (with $j=J$ "taken out" first) and then using the facts that the tuple $(\mathbf{u}^1,\ldots,\mathbf{u}^J)$ is good and the choices for $\mathbf{G}_1,\ldots,\mathbf{G}_N$ are independent.[8] (5.19) follows from Lemma 5.1. (5.20) follows from rearranging the summand and using the fact that the tuple is good (and hence $d_j\geqslant(1-R-\gamma)N$). (5.21) follows from the fact that there are q^{nN} choices[9] for $\mathbf{y}$ and Lemma 5.6. (5.22) follows from the fact that

[8] In Theorem 5.1, the tuple of codewords were not ordered while they are ordered here. However, it is easy to check that the argument in Theorem 5.1 also works for ordered tuples as long as the induction is applied in the right order.

[9] As the final code C will be linear, it is sufficient to only look at received words that have Hamming weight at most ρnN. However, this gives a negligible improvement to the final result and hence, we just bound the number of choices for $\mathbf{y}$ by q^{nN}.

$d_j - (1-R)N + 1 \leqslant d_j - (1-R-\gamma)N$ (for $N \geqslant 1/\gamma$) and that $d_j \geqslant (1-R-\gamma)N$. (5.23) follows by rearranging the terms.

Now, as long as $n \geqslant J(J+1)$, we have $\frac{N(J+1)}{nd} \leqslant \frac{N}{Jd}$. (5.23) will imply (5.15) (along with the fact that H_q^{-1} is increasing) if we can show that for every $(1-R-\gamma)N \leqslant d \leqslant N$,

$$\frac{h}{nd} \leqslant H_q^{-1}\left(1 - r\left(1 - \frac{(1-R-\gamma)N}{d}\right) - \frac{2N}{Jd}\right) - \delta,$$

for $\delta = \varepsilon/3$. Thus, by Lemma 5.4 (and using the arguments used in the proof of Theorem 5.1 to show that the conditions of Lemma 5.4 are satisfied), we can select J in $\Theta\left(\frac{1}{\varepsilon^2(1-R)}\right)$ (and γ in $\Theta(\varepsilon^2(1-R)/r)$), and pick

$$h/(nN) = H_q^{-1}(1-rR) - \varepsilon = \rho,$$

as desired. This along with Lemma 5.5, implies that we can set

$$L = (N/\varepsilon^2)^{O\left(\varepsilon^{-4}(1-R)^{-2}\log(q/R)\right)},$$

as required.

Using arguments similar to those in the proof of Theorem 5.1, one can show that the code $C_{out} \circ (C_{in}^1, \ldots, C_{in}^N)$ with high probability has rate rR.

Remark 5.2. *The idea of using list recoverability to argue independence can also be used to prove Theorem 5.1. That is, first show that with good probability, a random linear outer code will have good list recoverability. Then the argument in this section can be used to prove Theorem 5.1. However, this gives worse parameters than the proof presented in Section 5.4. In particular, by a straightforward application of the probabilistic method, one can show that a random linear code of rate R over $\mathbb{F}_Q$ is $(R+\gamma, \ell, Q^{\ell/\gamma})$-list recoverable [49, Sec 9.3.2]. In proof of Theorem 5.2, ℓ is roughly q^J, where J is roughly $1/\varepsilon^2$. Thus, if we used the arguments in the proof of Theorem 5.2, we would be able to prove Theorem 5.1 but with lists of size of $Q^{q^{O\left(\varepsilon^{-2}(1-R)^{-1}\right)}}$, which is worse than the list size of $Q^{O\left(\varepsilon^{-2}(1-R)^{-1}\right)}$ guaranteed by Theorem 5.1.*

5.6 Bibliographic Notes and Open Questions

The material presented in this chapter appears in [61].

Theorem 5.1 in some sense generalizes the following result of Blokh and Zyablov [19]. Blokh and Zyablov show that the concatenated code where both the outer and inner codes are chosen to be random linear codes with high probability lies on the Gilbert-Varshamov bound of relative minimum distance is (at least) $H_q^{-1}(1-R)$ for rate R.

The arguments used in this chapter also generalize Thommesen's proof that concatenated codes obtained by setting Reed-Solomon codes as outer codes and independent random inner code lie on the Gilbert-Varshamov bound [102]. In particular by using $\mathbf{y}$ to be the all zero vector and $J = L = 1$ in proof of Theorem 5.2, one can recover Thommesen's proof. Note that when $J = 1$, a codeword $\mathbf{c}$ is $(\langle w \rangle, \mathbb{F}_q)$-independent if $wt(\mathbf{w}) = w$. Thus, the proof of Thommesen only required a knowledge of the weight distribution of the Reed-Solomon code. However, for our purposes, we need a stronger form of independence in the proof of Theorem 5.2 for which we used the strong list-recoverability property of folded Reed-Solomon codes.

Theorem 5.2 leads to the following intriguing possibility.

Open Question 5.1. *Can one list decode the concatenated codes from Theorem 5.2 up to the fraction of errors for which Theorem 5.2 guarantees it to be list decodable (with high probability)?*

Current list-decoding algorithms for concatenated codes work in two stages. In the first stage, the inner code(s) are list decoded and in the second stage the outer code is list recovered (for example see Chapter 4). In particular, the fact that in these algorithms the first phase is oblivious to the outer codes seems to be a bottleneck. Somehow "merging" the two stages might lead to a positive resolution of the question above.