Electronic Colloquium on Computational Complexity, Report No. 117 (2022)

Error Correcting Codes that Achieve BSC Capacity Against Channels that are Poly-Size Circuits

Ronen Shaltiel[*] Jad Silbak[†]

August 23, 2022

Abstract

Guruswami and Smith (J. ACM 2016) considered codes for channels that are poly-size circuits which modify at most a p-fraction of the bits of the codeword. This class of channels is significantly stronger than Shannon's binary symmetric channel (BSC), but weaker than Hamming's channels which are computationally unbounded. Guruswami and Smith gave an explicit Monte-Carlo construction of codes with optimal rate of $R(p) = 1-H(p)$ that achieve *list-decoding* in this scenario. Here, "explicit Monte-Carlo" means that both encoding and decoding algorithms run in polynomial time. However, the encoding and decoding algorithms also receive a uniformly chosen string of polynomial length (which is chosen and published, once and for all, in a pre-processing stage) and their correctness is guaranteed w.h.p. over this random choice. Guruswami and Smith asked whether it is possible to obtain *uniquely decodable* codes for poly-size channels with rate that beats the Gilbert-Varshamov bound $R^{GV}(p) = 1 - H(2p)$. We give an affirmative answer, Specifically:

- For every $0 \leq p < \frac{1}{4}$, we give an explicit Monte-Carlo construction of *uniquely-decodable* codes with optimal rate $R(p) = 1 - H(p)$. This matches the rate achieved by Guruswami and Smith for the easier task of *list-decoding*, and also matches the capacity of binary symmetric channels. Moreover, this rate is strictly larger than that of codes for the standard coding scenario (namely, uniquely-decodable codes for Hamming channels).
- Even ignoring explicitness, our result implies a characterization of the capacity of poly-size channels, which was not previously understood.

Our technique builds on the earlier list-decodable codes of Guruswami and Smith, achieving *unique-decoding* by extending and modifying the construction so that we can identify the correct message in the list. For this purpose we use ideas from coding theory and pseudorandomness, specifically:

- We construct codes for binary symmetric channels that beat the Gilbert-Varshamov bound, and are "evasive" in the sense that a poly-size circuit that receives a *random* (or actually *pseudorandom*) string, cannot find a codeword within relative distance $2p$. This notion of evasiveness is inspired by the recent work of Shaltiel and Silbak (STOC 2021) on codes for space bounded channels.
- We develop a methodology (that is inspired by proofs of t-wise independent tail inequalities, and may be of independent interest) to analyze random codes, in scenarios where the success of the channel is measured in an additional random experiment (as in the evasiveness experiment above).
- We introduce a new notion of "small-set non-malleable codes" that is tailored for our application, and may be of independent interest.

[*] Department of Computer Science. University of Haifa, Email: ronen@cs.haifa.ac.il. Supported by ISF grant 1628/17.

[†] School of Computer Science, Tel Aviv University. Email: jadsilbak@mail.tau.ac.il. Supported by ISF grant 1628/17, ISF grant 666/19, and ERC starting grant 638121.

ISSN 1433-8092

Contents

1 Introduction

1.1 Codes and channels

Coding theory studies transmission of messages using noisy channels. In this paper we are interested in binary codes, and prefer to focus on decoding properties of a code, rather than combinatorial properties like minimal distance. More specifically, given a family $\mathcal{C}$ of (possibly randomized) functions $C : \{0,1\}^n \to \{0,1\}^n$ (which we call "channels") the goal is to design a code (namely, a pair $(\mathrm{Enc}, \mathrm{Dec})$ of an encoding map $\mathrm{Enc} : \{0,1\}^k \to \{0,1\}^n$ and a decoding map $\mathrm{Dec} : \{0,1\}^n \to \{0,1\}^k$) such that for every message $m \in \{0,1\}^k$ and every channel $C \in \mathcal{C}$, decoding is successful, namely:

$$\mathrm{Dec}(\mathrm{Enc}(m) \oplus C(\mathrm{Enc}(m))) = m.$$

The rate of a code is $R = \frac{k}{n}$. For a family $\mathcal{C}$ of channels, we use $R(\mathcal{C})$ to denote the capacity of the family, which is the best possible rate of a code for this family.[1] For a family $\mathcal{C}$ of channels, there are two main goals:

1. Determine the capacity $R(\mathcal{C})$.
2. Construct explicit codes (namely codes with poly-time encoding and decoding algorithms).

Let us review some coding scenarios and channel families. In all examples below $0 \leq p < \frac{1}{2}$ is a parameter.

Binary symmetric channels. A binary symmetric channel (denoted by BSC_p) is the randomized function that ignores its input and produces n i.i.d. random bits, where each of them is one with probability p. This is a special case of an extensively studied class of randomized channels (often referred to as "Shannon channels"). A celebrated theorem of Shannon shows that $R(\mathrm{BSC}_p) = 1 - H(p)$.[2] Later work on code concatenation (due to Forney [For65]) produced codes with explicit and even linear time algorithms [GI05].

Hamming channels. The class of Hamming channels (denoted by Ham_p) is the class of all functions such that for every input x, the relative Hamming weight of $C(x)$ is at most p.[3] (This corresponds to a channel that flips at most a p fraction of the bits). This class is probably the most studied class of channels, and yet, its capacity $R(\mathrm{Ham}_p)$ is not precisely understood. It is known that $R(\mathrm{Ham}_p) = 0$ for $p \geq \frac{1}{4}$, and that for $0 < p < \frac{1}{4}$, $R(\mathrm{Ham}_p) < 1 - H(p)$.[4] The Gilbert-Varshamov bound shows that $R(\mathrm{Ham}_p) \geq R^{GV}(p) = 1 - H(2p)$, but explicit codes with this rate are unknown. Recently, there has been progress on explicit codes with rate that is close to the Gilbert-Varshamov bound for p approaching $\frac{1}{4}$ [TS17, JST21, BD22].

List-decoding. In the relaxed goal of list-decoding, the decoding map is allowed to output a list of $L = O(1)$ messages, and decoding is considered successful if $\mathrm{Dec}(\mathrm{Enc}(m) \oplus C(\mathrm{Enc}(m))) \ni m$. Unlike the case of unique decoding, the list decoding capacity of Hamming channels (denoted by $R^{\mathrm{List}}(\mathrm{Ham}_p)$) is known

[1] More formally, $R(\mathcal{C})$ is the largest number R such that for every $\epsilon > 0$, there exist infinitely many n, for which there exists a code for $\mathcal{C}$, with rate at least $R - \epsilon$. We mostly use the term "rate" for a specific (family of) codes, and "capacity" for a class of channels, but these terms are interchangeable in this paper.

[2] Here $H(p) = p \cdot \log(1/p) + (1-p) \cdot \log(1/(1-p))$ is Shannon's entropy function.

[3] The relative Hamming weight of a string $z \in \{0,1\}^n$ is $wt(z) = \frac{|\{i \in [n] : z_i \neq 0\}|}{n}$.

[4] This follows because by the Elias-Bassalygo bound, which states that $R(\mathrm{Ham}_p) < R_{\text{Elias-Bassalygo}}(p)$ where the latter is strictly smaller than $1 - H(p)$. We remark that the Elias-Bassalygo bound gives a stronger result, and that later work by McEliece, Rodemich, Rumsey and Welch [MRRW77] improves this bound in some ranges. We state the bound $R < 1 - H(p)$ to stress that $R(\mathrm{Ham}_p) < R(\mathrm{BSC}_p) = 1 - H(p)$.

to be $R^{\text{List}}(\text{Ham}_p) = 1 - H(p)$,[5] which allows positive rate even for $\frac{1}{4} \leq p < \frac{1}{2}$ (in contrast to unique decoding). Explicit constructions of such codes are unknown.

Intermediate classes of channels. It is natural to consider intermediate classes of channels that lie between binary symmetric channels and Hamming channels. One such example was studied by Guruswami and Smith [GS16] that considered the class of additive channels. This class (denoted by Add_p) contains all *constant* functions $C \in \text{Ham}_p$. This means that an additive channel $C : \{0,1\}^n \to \{0,1\}^n$ has a predetermined noise vector $e \in \{0,1\}^n$ of Hamming weight at most p, and the channel C uses this noise vector *regardless* of its input. In particular, the channel does not choose the noise vector as a function of the transmitted codeword.

It turns out that with the standard definition of codes, every code for additive channels is also a code for Hamming channels.[6] In order to take advantage of restricted families of channels, one needs to consider a different coding scenario. Several such scenarios were considered in the literature. In this paper, we follow the approach of Guruswami and Smith [GS16] and consider *stochastic codes*.

Stochastic codes. These are codes where the encoding algorithm is randomized, and decoding only needs to succeed with high probability. More precisely, an encoding map of a *stochastic code*, is a function $\text{Enc} : \{0,1\}^k \times \{0,1\}^d \to \{0,1\}^n$ and it is required that for every $m \in \{0,1\}^k$, and every channel C in the considered class:

$$\Pr_{S \leftarrow U_d}[\text{Dec}(\text{Enc}(m,S) \oplus C(\text{Enc}(m,S)) = m] \geq 1 - \nu,$$

where ν is an error parameter. (A precise formal definition is given in Definition 3.8). Note that the decoding algorithm *does not* need to receive S, and so, these codes can be used in the standard coding communication scenario. The rate of a stochastic code is $R = \frac{k}{n}$. Stochastic codes do not give an improvement in capacity in the case of Hamming channels (as it is easy to show that a stochastic code for Hamming channels yields a standard code with the same rate) but they do allow improved capacities for other classes.

Additive channels. Recall that an additive channel $C \in \text{Add}_p$ is a channel that ignores the transmitted codeword, and always uses a predetermined noise vector of Hamming weight at most p. Guruswami and Smith [GS16] showed that the *stochastic capacity*, $R^{\text{Stoc}}(\text{Add}_p) = 1 - H(p)$,[7] while also providing explicit encoding and decoding algorithms for the stochastic code.

Online channels with small space. The class of space-bounded channels (denoted by Spc_p^s) is the class of all $C \in \text{Ham}_p$, where C reads its input in one pass, using space s, and produces its i'th output bit before reading the $(i+1)$'th bit. Guruswami and Smith [GS16] showed that $R^{\text{Stoc}}(\text{Spc}_p^{\log n}) = 0$ for $p > \frac{1}{4}$.[8] Shaltiel and Silbak [SS21b] (building on earlier work [GS16, SS21a, KSS19] that considered list-decoding) showed that $R^{\text{Stoc}}(\text{Spc}_p^s)$ is $1 - H(p)$ for $s = n^{1-o(1)}$ and $0 < p < \frac{1}{4}$, while also providing explicit codes for $s = n^{\Theta(1)}$. Chen, Jaggi and Langberg [CJL15], and Dey, Jaggi, Langberg, and Sarwate [DJLS13], considered Causal channels (in which $s = n$, and there are no space restrictions). These works determined

[5] This formally means that for every $\epsilon > 0$, there exists a constant L_ϵ such that there are infinitely many n, for which there exists a code with rate $R = 1 - H(p) - \epsilon$ and a list-decodable code with list size L_ϵ.

[6] This follows as if there is a message $m \in \{0,1\}^k$ and a channel $C \in \text{Ham}_p$ such that $\text{Dec}(\text{Enc}(m) \oplus C(\text{Enc}(m))) \neq m$, then the channel $C'(x) = C(\text{Enc}(m))$ is a channel in Add_p on which decoding is not successful.

[7] This formally means that for every $\epsilon > 0$, and for infinitely many n, there is a stochastic code with rate $R = 1 - H(p) - \epsilon$ for Add_p that achieves success probability $1 - o(1)$. In fact, the success probability achieved by [GS16] against additive channels is much better.

[8] More formally, Guruswami and Smith [GS16] showed that for every constant $p > \frac{1}{4}$, there does not exist a family of stochastic codes with positive rate against $\text{Spc}_p^{\log n}$, that achieves success probability $1 - o(1)$.

the capacity $R^{\mathrm{Stoc}}(\mathrm{Spc}_p^n)$, and showed that this capacity is $1 - H(p)$ for $p \leq p_0 \approx 0.0804$, and is strictly smaller than $1 - H(p)$ for $p > p_0$. These bounds are achieved by non-explicit constructions.

Poly-size channels. In this paper we will consider the class of channels that can be implemented by polynomial size circuits, that includes channels that can be implemented in polynomial time. The class Ckt_p^s is the class of all $C \in \mathrm{Ham}_p$ such that C is a circuit of size s. We will focus on the case that $s = n^c$ for a constant c, and call this class poly-size channels. Guruswami and Smith gave an "explicit Monte-Carlo construction" of stochastic codes with rate $1 - H(p)$ that are *list-decodable* for $\mathrm{Ckt}_p^{n^c}$. We define and discuss the notion of "Monte-Carlo constructions" in the next section.

We summarize all theses surveyed results (as well as our new results) in Table 1.

Table 1: Summary of surveyed known results. The results of this paper appear in bold text

Channel	Decoding	Stochastic?	Range	Rate	Explicit?	Reference
BSC_p	Unique	No	$0 \leq p < \frac{1}{2}$	$R = 1 - H(p)$	Yes	[For65]
Ham_p	Unique	No	$0 \leq p < \frac{1}{4}$	$R < 1 - H(p)$	N/A	[MRRW77]
Ham_p	Unique	No	$\frac{1}{4} \leq p < \frac{1}{2}$	$R = 0$	N/A	Plotkin bound
Ham_p	List	No	$0 \leq p < \frac{1}{2}$	$R = 1 - H(p)$	No	Easy
Add_p	Unique	Yes	$0 \leq p < \frac{1}{2}$	$R = 1 - H(p)$	Yes	[GS16]
Spc_p^n	Unique	Yes	$0 \leq p < 0.0804$	$R = 1 - H(p)$	No	[CJL15]
Spc_p^n	Unique	Yes	$0.0804 < p < \frac{1}{4}$	$R < 1 - H(p)$	No	[DJLS13]
$\mathrm{Spc}_p^{n^{\Theta(1)}}$	List	Yes	$0 \leq p < \frac{1}{2}$	$R = 1 - H(p)$	Yes	[KSS19]
$\mathrm{Spc}_p^{n^{\Theta(1)}}$	Unique	Yes	$0 \leq p < \frac{1}{4}$	$R = 1 - H(p)$	Yes	[SS21b]
$\mathrm{Spc}_p^{n^{1-o(1)}}$	Unique	Yes	$0 \leq p < \frac{1}{4}$	$R = 1 - H(p)$	No	[SS21b]
$\mathrm{Ckt}_p^{O(n)}$	Unique	Yes	$\frac{1}{4} < p < \frac{1}{2}$	$R = 0$	N/A	[GS16]
$\mathrm{Ckt}_p^{n^c}$	List	Yes	$0 \leq p < \frac{1}{2}$	$R = 1 - H(p)$	Monte-Carlo	[GS16]
$\mathrm{Ckt}_p^{n^c}$	List	Yes	$0 \leq p < \frac{1}{2}$	$R = 1 - H(p)$	explicit, under hardness assumptions	[SS21a]
$\mathbf{Ckt_p^{n^c}}$	**Unique**	**Yes**	$\mathbf{0 \leq p < \frac{1}{4}}$	$\mathbf{R = 1 - H(p)}$	**Monte-Carlo**	**Here**

1.2 Explicit Monte-Carlo constructions

A *Monte-Carlo construction* (with q bits of Monte-Carlo randomness) is a pair $(\mathrm{Enc}, \mathrm{Dec})$ of maps, such that in addition to their usual inputs Enc and Dec also receive a string $y \in \{0,1\}^q$. (More specifically, a *Monte-Carlo code* is a pair of maps $\mathrm{Enc} : \{0,1\}^q \times \{0,1\}^k \to \{0,1\}^n$ and $\mathrm{Dec} : \{0,1\}^q \times \{0,1\}^n \to \{0,1\}^k$, and a *Monte-Carlo stochastic code* is a pair of maps: $\mathrm{Enc} : \{0,1\}^q \times \{0,1\}^k \times \{0,1\}^d \to \{0,1\}^n$ and $\mathrm{Dec} : \{0,1\}^q \times \{0,1\}^n \to \{0,1\}^k$).

Let $(\mathrm{Enc}_y, \mathrm{Dec}_y)$ denote the maps obtained with fixed strings $y \in \{0,1\}^q$, as first input. We say that the construction has *Monte-Carlo error* $\eta = \eta(n)$, if with probability $1 - \eta$ over choosing $y \leftarrow \{0,1\}^q$, the obtained code $(\mathrm{Enc}_y, \mathrm{Dec}_y)$ have the desired property (which in our case is that $(\mathrm{Enc}_y, \mathrm{Dec}_y)$ is a stochastic code for $\mathrm{Ckt}_p^{n^c}$). A Monte-Carlo code is *explicit* if $q = q(n)$ is polynomial, and Enc and Dec run in time polynomial in n.[9]

[9] An alternative definition is that an explicit Monte-Carlo construction is a randomized algorithm A that on input n, runs in time $\mathrm{poly}(n)$, and with probability $1 - \eta(n)$ produces circuits $\mathrm{Enc}, \mathrm{Dec}$ of polynomial size with the desired property.

A more formal definition is given in Definition 3.8.

Explicit Monte-Carlo constructions vs. Random codes. We stress that a random code does not yield an explicit Monte-Carlo construction. One reason is that sampling a random code requires $\exp(n)$ random bits. A more acute reason is that even if one somehow reduces the number of random bits to polynomial, it is not clear how to implement a polynomial time decoding algorithm. A good example to keep in mind is a random linear code. Random linear codes can be sampled using polynomially many random bits, and have polynomial time encoding algorithms. Nevertheless, they are not explicit Monte-Carlo constructions, because we do not know how (and don't think it is possible to) couple them with polynomial time decoding algorithms.

Necessity of Monte-Carlo constructions in the case of poly-size channels. Explicit constructions of stochastic codes against a class $\mathcal{C}$ immediately imply lower bounds against the class $\mathcal{C}$ (unless they also apply for Hamming channels). See [SS21a] for a precise formulation of this statement and a discussion.

Given our current inability to prove circuit lower bounds, we cannot expect to obtain unconditional explicit constructions of stochastic codes for $\mathrm{Ckt}_p^{n^c}$, and the best we can hope for is explicit Monte-Carlo constructions (as is achieved in the aforementioned result of Guruswami and Smith [GS16], and in this paper). An alternative approach taken by Shaltiel and Silbak [SS21a] (that we discuss later in Section 2.6) is to base explicit constructions on complexity theoretic hardness assumptions.

1.3 Our results

1.3.1 An explicit Monte-Carlo construction of stochastic codes with optimal rate for $\mathrm{Ckt}_p^{n^c}$

We give an explicit Monte-Carlo construction of a *uniquely decodable* stochastic code for $\mathrm{Ckt}_p^{n^c}$ with optimal rate of $R = 1 - H(p)$.

Theorem 1.1 (Explicit Monte-Carlo construction of stochastic codes with optimal rate for $\mathrm{Ckt}_p^{n^c}$)**.** *For every constants $0 \le p < \frac{1}{4}$, $\epsilon > 0$, and $c > 1$, there is an explicit Monte-Carlo construction of a stochastic code with rate $1 - H(p) - \epsilon$ for $\mathrm{Ckt}_p^{n^c}$, achieving success probability $1 - \frac{1}{n^c}$.*

Theorem 1.1 is restated in a more general way that includes more precise parameters in Theorem 7.1.

Guruswami and Smith [GS16] showed that stochastic codes for $\mathrm{Ckt}_p^{O(n)}$ (in fact even for a weaker class) cannot have positive rate for $p > \frac{1}{4}$. This means that Theorem 1.1 achieves the best possible capacity for every choice of p, and in particular, we obtain a characterization of the capacity $R^{\mathrm{Stoc}}(\mathrm{Ckt}_p^{n^c})$ of stochastic codes for $\mathrm{Ckt}_p^{n^c}$, showing that:

$$R^{\mathrm{Stoc}}(\mathrm{Ckt}_p^{n^c}) = \begin{cases} 1 - H(p) & 0 \le p < 1/4 \\ 0 & p > 1/4. \end{cases}$$

Note that this determines $R^{\mathrm{Stoc}}(\mathrm{Ckt}_p^{n^c})$ for every p except $p = \frac{1}{4}$.[10] A curious consequence is that the capacity function is not continuous at $p = \frac{1}{4}$, as $1 - H(\frac{1}{4}) > 0$.

[10] The aforementioned result of Guruswami and Smith [GS16] (that rules out stochastic codes for $\mathrm{Ckt}_p^{O(n)}$ with positive rate, and success probability $1 - o(1)$, for $p > \frac{1}{4}$) can be extended to the case that $p = \frac{1}{4}$, and show that there does not exist a stochastic code for $\mathrm{Ckt}_p^{O(n)}$ with positive rate, and success probability $1 - O(\frac{1}{n})$. It is open whether this lower bound can be extended to rule out stochastic codes with success probability $1 - o(1)$. We omit the details to the full version.

Comparison to previous work. Prior to this work it was not known whether codes for $\mathrm{Ckt}_p^{n^c}$ can beat the Gilbert-Varshamov bound $R^{GV} = 1 - H(2p)$, and this was raised as an open problem by Guruswami and Smith [GS16]. We completely resolve this problem, while achieving the best possible rate (matching the rate of codes for BSC_p, or list-decoding for Ham_p) while also achieving an explicit Monte-Carlo construction. Our codes achieve the same rate, and success probability as those given by Guruswami and Smith [GS16] for the harder task of *unique-decoding* whereas the codes of Guruswami and Smith only achieve *list-decoding*.

The running time of Enc and Dec in Theorem 1.1 is polynomial in n, for a larger polynomial than n^c. This is also the case in the list-decodable codes of Guruswami and Smith [GS16]. See Section 2.6 for a discussion.

The capacity of $\mathrm{Ckt}_p^{n^c}$ is larger than that of Ham_p. The codes of Theorem 1.1 have rate that is superior to that of codes for the standard model of decoding against Hamming channels (regardless of the issue of explicitness). More precisely, while the capacity of codes for Ham_p is not understood, it is known (by the aforementioned Elias-Bassalygo bound, and improvements in [MRRW77]) that this capacity is strictly smaller than $1 - H(p)$. We show that it is possible to achieve rate $1 - H(p)$ for $\mathrm{Ckt}_p^{n^c}$,

Note that this is in contrast to the case of *list-decodable* stochastic codes for $\mathrm{Ckt}_p^{n^c}$ considered by Guruswami and Smith [GS16], where the rate of the (Monte-Carlo constructed) stochastic codes, matches the rate of list-decoding for Ham_p, rather than beating it. Indeed, in the case of Guruswami and Smith [GS16], the advantage of codes for $\mathrm{Ckt}_p^{n^c}$ over codes for Ham_p is in explicitness, rather than in "combinatorial supremacy".

Perspective. Our results immediately extend to channels that are randomized poly-size circuits, as these can be viewed as a convex combination of channels from $\mathrm{Ckt}_p^{n^c}$. It seems that all classes of channels that are studied in the Shannon literature are captured by Ckt_p^s for $s = O(n)$. On a more philosophical level, it is hard to imagine a "physical channel" that is not implementable by a poly-size or even linear size circuit.

1.3.2 An analysis of random stochastic codes

Prior to this work it was unknown whether stochastic codes for $\mathrm{Ckt}_p^{n^c}$ with rate that beats the Gilbert-Varshamov bound exist. In particular, it was not known whether a random stochastic code achieves this. This is an intriguing question, even though, now, by Theorem 1.1 there exist such codes with rate $1 - H(p)$.

In this paper, we develop new techniques to argue about random stochastic codes, and related scenarios. This technique (that is inspired from ideas used in proofs of t-wise independent tail inequalities) is outlined in Section 2, and presented in Section 4. We can use this technique to give a direct proof that a random stochastic code with rate $1 - H(p)$ is w.h.p. good for $\mathrm{Ckt}_p^{n^c}$, or in fact for any class $\mathcal{C}$ which contains $2^{2^{\alpha n}}$ channels, where $\alpha > 0$ is a constant.

Theorem 1.2 (Random stochastic codes that decode against small families)**.** *For every constants $0 \le p < \frac{1}{4}$ and $\epsilon > 0$, there exist constants $\alpha > 0$ and $c_d > 0$, such that for $R = 1 - H(p) - \epsilon$, and for every sufficiently large n, and every class $\mathcal{C} \subseteq \mathrm{Ham}_p$ of size at most $2^{2^{\alpha \cdot n}}$, a function $\mathrm{Enc} : \{0,1\}^{Rn} \times \{0,1\}^{c_d \cdot n} \to \{0,1\}^n$ chosen uniformly from all such functions, coupled with a function $\mathrm{Dec} : \{0,1\}^n \to \{0,1\}^{Rn}$ that applies maximum likelihood decoding, is with probability $1 - 2^{-2^{\alpha \cdot n}}$ a code for $\mathcal{C}$ achieving success probability $1 - 2^{-\alpha \cdot n}$.*

Theorem 1.2 is restated in a more precise way that includes more precise parameters in Theorem 8.1.

2 Overview of the technique

In this section we give an overview of the main ideas that we use. For this purpose we will allow ourselves to be informal, and not entirely precise. The later technical sections do not build on the content of this section, and the reader can skip to the technical section if they wish.

Our overall approach builds on the *list-decodable* stochastic codes for $\mathrm{Ckt}_p^{n^c}$ by Guruswami and Smith. We would like to modify and enhance the construction so that we can identify the correct message in the output list, and achieve *unique decoding*. For this purpose, we introduce several ideas that we explain below.

2.1 Codes for BSC_p that are also evasive for $\mathrm{Ckt}_p^{n^c}$

Following Guruswami and Smith [GS16] we will use a code $(\mathrm{Enc}_{\mathrm{BSC}}, \mathrm{Dec}_{\mathrm{BSC}})$ for BSC_p as a component in the construction of stochastic codes, and the final stochastic code will inherit the rate of the BSC code. Shaltiel and Silbak [SS21b] introduced a notion of "evasive codes" that plays a major part in their construction of stochastic codes for space bounded channels. We will now present a related notion that is tailored for our intended application where we are interested in poly-size channels.

Evasiveness of codes. We will be interested in (standard) codes $(\mathrm{Enc}_{\mathrm{BSC}}, \mathrm{Dec}_{\mathrm{BSC}})$ for BSC_p that have an additional evasiveness property against $\mathrm{Ckt}_p^{n^c}$. More specifically, we would like to modify the decoding algorithm $\mathrm{Dec}_{\mathrm{BSC}}$ into a new decoding algorithm $\mathrm{Dec}'_{\mathrm{BSC}}$, where $\mathrm{Dec}'_{\mathrm{BSC}}$ is allowed to output $fail$, so that:

- $(\mathrm{Enc}_{\mathrm{BSC}}, \mathrm{Dec}'_{\mathrm{BSC}})$ still form a code for BSC_p, meaning that, for every $m \in \{0,1\}^k$, for $z = \mathrm{Enc}_{\mathrm{BSC}}(m)$, we have that w.h.p. $\mathrm{Dec}'_{\mathrm{BSC}}(z \oplus \mathrm{BSC}_p) = m$.
- We say that the code is *ρ-evasive* for a class $\mathcal{C}$, if when a uniform $Z \leftarrow U_n$ is corrupted by a channel $C \in \mathcal{C}$, then w.h.p. Dec' fails. Namely, $\Pr_{Z \leftarrow U_n}[\mathrm{Dec}'_{\mathrm{BSC}}(Z \oplus C(Z)) = fail] \geq 1 - \rho$.

A more formal definition appears in Section 5.1. We stress that in the second requirement we are interested in a scenario in which C receives $Z \leftarrow U_n$ (rather than a codeword $z = \mathrm{Enc}(m)$).

An easy modification of $\mathrm{Dec}_{\mathrm{BSC}}$ is to make it fail if it decodes to a codeword that has distance slightly larger than p from the received corrupted word. This obviously satisfies the first item. Using this modification, codes for BSC_p with rate that doesn't beat the Gilbert-Varshamov bound (namely, rate $R < R^{GV}(p) = 1 - H(2p)$) can be easily made evasive even against the class Ham_p of computationally unbounded channels.[11]

Shaltiel and Silbak [SS21b] gave an explicit construction of codes for BSC_p that beat the Gilbert-Varshamov bound, and are evasive for small space channels. This constructions rely on the weakness of small space channels, and roughly show that any "well behaved" code for BSC_p can be made evasive for small space channels.

We need to construct codes for BSC_p, that are evasive for $\mathrm{Ckt}_p^{n^c}$, which is a significantly more powerful class (for which we have no lower bounds). By the discussion in Section 1.2 we cannot expect to have unconditional constructions, unless they are also evasive for Ham_p. We will therefore require a very different approach from that of Shaltiel and Silbak [SS21b].[12]

[11] More precisely, any code with rate $R \leq 1 - H(2p) - \epsilon$ is immediately $2^{-\epsilon n}$-evasive. This is because a uniform $Z \leftarrow U_n$ is likely to have relative Hamming distance larger than $2p$ from any codeword, and so, by the triangle inequality, a channel $C \in \mathrm{Ham}_p$, cannot corrupt a p fraction of the bits, so that the corrupted codeword is within relative Hamming distance p from a codeword.

[12] Furthermore, in the case that $\mathrm{Dec}_{\mathrm{BSC}}$ is the maximum likelihood decoder, and runs in polynomial time, evasiveness for poly-size circuits implies that no poly-size circuit can find a codeword that is within relative Hamming distance $2p$ from a random string.

We first give an explicit construction of codes for BSC_p with rate $R^*(p) > R^{GV}(p) = 1 - H(2p)$ that are evasive not only for $\mathrm{Ckt}_p^{n^c}$, but in fact for the computationally unbounded class Ham_p. We find this quite surprising, as it is easy to see that random codes with rate that beats the Gilbert-Varshamov bound (namely, $R > R^{GV} = 1 - H(2p)$) are not evasive against Ham_p (See Section 5 for details). This means that in the construction given in Section 5.3, we will need to use codes that are very different than random codes.
We also give an explicit Monte-Carlo construction (which achieves optimal rate of $1 - H(p)$) for $\mathrm{Ckt}_p^{n^c}$, that is stated below:

Informal Theorem 2.1 (Monte-Carlo codes for BSC_p with optimal rate, that are evasive for $\mathrm{Ckt}_p^{n^C}$)**.** *For every constants $p < \frac{1}{4}$, $c > 1$, and $\epsilon > 0$, there is an explicit Monte-Carlo construction of codes for* BSC_p *with rate $1 - H(p) - \epsilon$ that is evasive against* $\mathrm{Ckt}_p^{n^c}$.

As we are shooting for a Monte-Carlo construction of stochastic codes for $\mathrm{Ckt}_p^{n^c}$, we can afford to use the Monte-Carlo construction of Theorem 2.1. In Section 2.6 we discuss the possibility of constructing explicit (rather than Monte-Carlo) stochastic codes based on hardness assumptions, and the aforementioned explicit construction of evasive codes with rate $R^*(p)$ can be viewed as a first step towards this goal.

An overview of the proof of Theorem 2.1 appears in Section 2.4. For this proof, we develop new machinery to analyze random codes, based on an intuition that is borrowed from techniques used to prove t-wise independent tail inequalities. We believe that this machinery is of independent interest, and we also use it as part of our Monte-Carlo construction of "small-set non-malleable codes" that we describe next.

2.2 Small Set Non-malleable codes

In this paper we introduce a new notion of stochastic codes that is related to the well studied notion of non-malleable codes (defined by Dziembowski, Pietrzak and Wichs [DPW18]) but is defined using a different methodology and tailored to handle a related, but somewhat different scenario that comes up in our setting.

Background on non-malleable codes. Loosely speaking, non-malleable codes are designed to work against channels $C : \{0,1\}^n \to \{0,1\}^n$, which when given a codeword z, produce a "corrupted word" $C(z)$. However, in contrast to earlier settings that we discussed, there will be no bound on the number of errors that C can induce. This means that C can erase the codeword and replace it with another string, and we cannot expect the decoding algorithm to produce the original message m. Instead, the goal is to show that C is unable to make the decoding algorithm produce a message $\bar{m} \neq m$ that is related to m.

The new notion of SS-non-malleability, and comparison to earlier notions. We will be interested in the scenario where the message M is chosen uniformly from $\{0,1\}^k$, and will require that for every small circuit C, one can guess in advance a small set H_C of messages, such that C is unlikely to make the decoding algorithm produce a message that is neither in H_C nor the original message M.

Moreover, we will require that this holds even if the adversary also receives additional information about M, in the form of some specific function $\psi(M)$. (It is instructive to think of the case that $\psi(M) = M$, giving the adversary the ability to decode "for free").[13] This means that unlike the notion of non-malleable codes defined by Dziembowski, Pietrzak and Wichs [DPW18], this notion does not imply the inability of the adversary to *decode*. Instead, it intuitively relies upon the inability of the adversary to *encode*.[14]

[13] A definition with a similar "small set" flavor (but with a "worst-case" choice of message, and without allowing C to obtain additional information in the form of $\psi(M)$) was defined by Faust et al. [FMVW16] where it was called "bounded malleability" and used as an intermediate notion, in order to produce the standard notion of non-malleability.

[14] More specifically, while this definition has the same intuition as the standard definition given by Dziembowski, Pietrzak and

Definition of SS-non-malleable codes. More specifically, let $\mathrm{Enc}: \{0,1\}^k \times \{0,1\}^d \to \{0,1\}^n$ and $\mathrm{Dec}: \{0,1\}^n \to \{0,1\}^k$ be a stochastic code, and let $\psi(m)$ be some function that outputs strings. We say that $(\mathrm{Enc}, \mathrm{Dec})$ is an *SS-non-malleable code* with respect to ψ, *set size* h and *error* ρ, against a circuit $C: \{0,1\}^n \to \{0,1\}^n$ if there exists a set H_C of size at most h such that:

$$\Pr_{\substack{M \leftarrow \{0,1\}^k \\ S \leftarrow \{0,1\}^d}} [\mathrm{Dec}\left(C(\mathrm{Enc}(M,S), \psi(M))\right) \notin H_C \cup \{M\}] \le \rho.$$

Note that this definition is quite compact, and does not mention "simulators" or "indistinguishability". This makes it easy to use, and to argue about.

More importantly, this definition is tailored for our intended application, in which the adversary will receive information $\psi(M)$ about M that we do not completely control, and can potentially reveal a lot of information about M. We remark that in our actual application we use a more general definition (that also allows C to view many encodings of the same message M, in addition to $\psi(M)$). This comes up because in our application the adversary receives this additional information. The precise formal definition and some additional discussion, appear in Section 6.

We will be interested in the adversaries that are poly-size circuits. There are several known constructions of non-malleable codes for this class [DPW18, CG16, FMVW16, BDK+19, DKP21, BDL22]. Some of these constructions rely on complexity theoretic and/or cryptographic assumptions, and others are Monte-Carlo. We cannot use these constructions as they do not seem to have the properties that we require. More specifically, we require SS-non-malleability, as well as some additional properties listed below.

Additional properties that are required in our intended application. In our intended application (of constructing stochastic codes with optimal rate for $\mathrm{Ckt}_p^{n^c}$) we will require stochastic codes that in addition to being SS-non-malleable for poly-size circuits, also decode against channels in Ham_p, and moreover, we will need them to have certain additional pseudorandom properties. It seems very hard to construct such stochastic codes (even assuming cryptographic assumptions). Fortunately, in our intended application, these SS-non-malleable codes will be used to encode short strings of logarithmic length, and this will make it easier to give a Monte-Carlo construction of such codes that runs in polynomial time (by choosing a random stochastic code, which requires a polynomial number of random bits). This is stated in the next theorem.

Informal Theorem 2.2 (A Monte-Carlo construction of SS-non-malleable codes)**.** *For every constants $p < \frac{1}{4}$ and $c > 1$, there exist constants $c_d, c_b \ge 1$, such that a random stochastic code (namely, a random encoding map* $\mathrm{Enc}: \{0,1\}^{\log N} \times \{0,1\}^{c_d \cdot \log N} \to \{0,1\}^{c_b \cdot \log N}$ *which can be sampled using* $\mathrm{poly}(N)$ *random bits) is w.h.p. both a stochastic code for* Ham_p *and SS-non-malleable for circuits of size N^c (when coupled with maximum likelihood decoding up to distance p).*

Wichs [DPW18], it is on one hand weaker, as it is only designed to guarantee security if M is chosen uniformly from $\{0,1\}^k$, whereas the standard definition guarantees security on any distribution. Moreover, even in the case of a uniform M, the small set condition seems weaker than what is guaranteed by the standard definition (as it allows the decoded message to be relate to M if it is in H_C). On the other hand, the definition given here is stronger in the sense that security is guaranteed even if the adversary receives additional information $\psi(M)$ about M that may allow it to "decode" and obtain M. We also remark that an adversary that can compute the encoding function, can break SS-non-malleability (regardless of the function ψ) by replacing the codeword with an encoding of a uniform message M' that is independent of M, and trivially, M' is unlikely to fall in any small set. This demonstrates that the notion of SS-non-malleability that we define, does not allow the adversary to encode the code, whereas the standard notion does not seem to prevent the adversary from computing the encoding, as can be seen by the construction of Dachman-Soled, Komargodski and Pass [DKP21] which (under cryptographic assumptions) give a non-malleable code in which the adversary *is able* to compute the encoding function).

Theorem 2.2 is stated more formally as Theorem 6.5 in Section 6. We stress that it is not obvious that a random code is non-malleable or SS-non-malleable. The seminal paper of Dziembowski, Pietrzak and Wichs [DPW18] had an analysis of a randomized construction, which was later improved by Faust et al. [FMVW16] and Cheraghchi and Guruswami [CG16]. However, we need the additional security against adversaries that see many encodings, and receive additional information ψ, as well as additional decoding properties against Hamming channels and pseudorandomness properties. Consequently, we cannot use the previous analyses.[15] An intriguing open problem is to obtain such codes explicitly (rather than Monte-Carlo) under complexity theoretic or cryptographic hardness assumptions. Together with our explicit construction of evasive codes, this will give a construction of stochastic codes for $\mathrm{Ckt}_p^{n^c}$ with rate $R^*(p)$ that is explicit (rather than Monte-Carlo) assuming these assumptions, see Section 2.6 for a discussion.

2.3 Stochastic codes that are uniquely decodable for $\mathrm{Ckt}_p^{n^c}$

At this point, we are ready to explain how to use the components that we already discussed (SS-non-malleable codes, and evasive codes for BSC_p) to obtain stochastic codes for $\mathrm{Ckt}_p^{n^c}$. We start with a brief overview of the list-decodable stochastic codes of Guruswami and Smith in Section 2.3.1, and in Section 2.3.2 we explain how to identify the correct candidate in the list, and obtain *unique decoding*.

2.3.1 A brief overview of the list-decodable stochastic codes of Guruswami and Smith

Guruswami and Smith [GS16] use an approach that "boosts" a (standard) code for BSC_p, so that it works for stronger channels like $\mathrm{Ckt}_p^{n^c}$, albeit, in an easier scenario than that of stochastic codes, called "codes with shared private randomness".

Codes with shared private randomness. This is a pair of maps $\mathrm{Enc}_{\mathrm{spr}} : \{0,1\}^k \times \{0,1\}^d \to \{0,1\}^n$ and $\mathrm{Dec}_{\mathrm{spr}} : \{0,1\}^n \times \{0,1\}^d \to \{0,1\}^k$ such that for every message $m \in \{0,1\}^k$, and for every $C \in \mathrm{Ckt}_p^{n_c}$, let $Z = \mathrm{Enc}_{\mathrm{spr}}(m, S)$ be the codeword, and $\bar{Z} = Z \oplus C(Z)$ be the "received word". It is guaranteed that:

$$\Pr_{S \leftarrow U_d}[\mathrm{Dec}_{\mathrm{spr}}(\bar{Z}, S) = m] \geq 1 - \nu.$$

Note that unlike the scenario of stochastic codes, in this scenario of shared private randomness, the decoding algorithm, *does receive* the randomness S chosen by the encoding procedure. This gives the decoding algorithm a huge advantage over the case of stochastic codes (where the decoding does not receive S).

Indeed, it is much easier to construct codes for shared private randomness than stochastic codes, and such a construction for $\mathrm{Ckt}_p^{n^c}$ was given by Lipton [Lip94], and extended by Smith [Smi07], and Guruswami and Smith [GS16]. In this paper we use a variant of this construction (which we need to modify, as in our application it will be crucial that $d = O(\log n)$). We explain this construction in Section 2.5.

Converting codes with shared randomness to stochastic codes for $\mathrm{Ckt}_p^{n^c}$. In order to convert a code with shared private randomness $(\mathrm{Enc}_{\mathrm{spr}}, \mathrm{Dec}_{\mathrm{spr}})$ for $\mathrm{Ckt}_p^{n^c}$, into a stochastic code $(\mathrm{Enc}, \mathrm{Dec})$ for $\mathrm{Ckt}_p^{n^c}$, Guruswami and Smith suggested to "embed" the seed S into the codeword in a way that will allow Dec to "list-decode" a small list of candidates for S, even after the codeword is corrupted by a channel in $\mathrm{Ckt}_p^{n^c}$.

[15] We remark that some of the aforementioned earlier works also showed how to derandomize the sampling of the random stochastic code, using t-wise independence, resulting in a Monte-Carlo construction. We stress that in our methodology, we do not rely on t-wise independence, but rather on techniques that were developed to *prove* t-wise independent tail inequalities. These two issue are orthogonal, and our technique seems different than that used in previous work.

This step will rely on a constant rate code $(\mathrm{Enc}_{\mathrm{ctrl}}, \mathrm{Dec}_{\mathrm{ctrl}})$ called a "control code" that decodes messages of length $d = O(\log n)$ (so that it can be applied to encode S) into codewords of length $b = O(\log n)$.

We now explain how Guruswami and Smith [GS16] used a code for shared randomness for $\mathrm{Ckt}_p^{n^c}$, and a control code (with certain properties to be discussed later) to construct a stochastic code that is *list-decodable* for $\mathrm{Ckt}_p^{n^c}$. The stochastic encoding algorithm Enc will receive an additional random seed $I \in [n]$ (in addition to the message $m \in \{0,1\}^k$, and seed $S \in \{0,1\}^d$ that are inputs to $\mathrm{Enc}_{\mathrm{spr}}$) and will start by preparing $Z = \mathrm{Enc}_{\mathrm{spr}}(m, S)$. It will then replace the b bits $Z_I, \ldots, Z_{I+b-1}$ of Z with $\mathrm{Enc}_{\mathrm{ctrl}}(S)$.

Guruswami and Smith [GS16] showed how to match Enc with a list-decoding algorithm Dec that outputs a list of candidate messages such that w.h.p. one of them is the original message. Loosely speaking, this is done as follows:[16] When Dec receives a "corrupted codeword" $\bar{Z} = Z \oplus C(Z)$, it will first prepare a list of candidate control strings $\bar{S}_1, \ldots, \bar{S}_n$ for the seed S, by setting $\bar{S}_i = \mathrm{Dec}_{\mathrm{ctrl}}(\bar{Z}_i, \ldots, \bar{Z}_{i+b-1})$. It will then output n candidate messages $\bar{M}_1, \ldots, \bar{M}_n$ by setting $\bar{M}_i = \mathrm{Dec}_{\mathrm{spr}}(\bar{Z}, \bar{S}_i)$.

Loosely speaking, Guruswami and Smith [GS16] showed that if the control code decodes from Ham_p (and has certain additional pseudorandomness properties that we will not discuss in this overview, and prevent the channel from identifying I) then w.h.p there exists $i^* \in [n]$ such that $\bar{S}_{i^*} = S$, so that $\bar{M}_{i^*} = \mathrm{Dec}_{\mathrm{spr}}(\bar{Z}, \bar{S}_i) = m$, and the correct message appears in the list of candidates.

2.3.2 Using evasiveness and SS-non-malleability to find the correct candidate in the list

Summing up the overview given above (of the list-decodable stochastic codes for $\mathrm{Ckt}_p^{n^c}$) when a channel $C \in \mathrm{Ckt}_p^{n^c}$ corrupts the codeword $Z = \mathrm{Enc}(m; (S, I))$ into $\bar{Z} = Z \oplus C(Z)$:

- C cannot prevent Dec from having S appear in the list of candidate control strings $\bar{S}_1, \ldots, \bar{S}_n$, and so the original message will appear in the list of candidate messages $\bar{M}_1, \ldots, \bar{M}_n$.
- However, C can affect the candidate control strings $\bar{S}_1, \ldots, \bar{S}_n$, and inject candidates to the list. (Note for example, that C can place the fixed string $\mathrm{Enc}_{\mathrm{ctrl}}(\bar{s})$ for any fixed $\bar{s} \in \{0,1\}^d$ that he chooses, as a substring of the corrupted codeword $\bar{Z}$, which will cause $\bar{s}$ to appear in the list of candidates).

We want to trim the list, and discard incorrect candidates. We will be interested in the following questions:

1. Can a channel $C \in \mathrm{Ckt}_p^{n^c}$ arrange it so that some of the candidates $\bar{S}_i$ are related to the correct control string S, and yet different from it?
2. What happens when $\mathrm{Dec}_{\mathrm{spr}}(\bar{Z}, \bar{S}_i)$ is applied on a corrupted codeword $\bar{Z}$, using an *incorrect* seed $\bar{S}_i \neq S$, and is unrelated to S?

Jumping ahead, we mention that we will use SS-non-malleability to handle the first case, and evasiveness to handle the second. We first address the first question. It turns out that in the construction of Guruswami and Smith [GS16], a channel $C \in \mathrm{Ckt}_p^{n^c}$ can control many of the candidate control strings $\bar{S}_1, \ldots, \bar{S}_n$. Moreover, C can arrange things so that many of these candidates are different than S, and yet correlated with S.

[16] The description that we give here is an over-simplification that is used to give the flavor of the construction of [GS16], and motivate the ingredients and ideas that we introduce in order to extend it so that it uniquely-decodes, rather than list-decodes. The actual construction of Guruswami and Smith is more involved, and we use the same approach (with some modifications) in Section 7.2.

Using SS-non-malleable control codes. The first step in our plan is to require additional "non-malleability" properties from the control code $(\mathrm{Enc}_{\mathrm{ctrl}}, \mathrm{Dec}_{\mathrm{ctrl}})$. Using the SS-non-malleable codes of Theorem 2.2, prevents C from injecting candidate control strings that are different than S, except for candidates in a small set H_C that is fixed in advance. We will use the stochastic codes from Theorem 2.2 as the control code $(\mathrm{Enc}_{\mathrm{ctrl}}, \mathrm{Dec}_{\mathrm{ctrl}})$, and they indeed decode from Ham_p and are SS-non-malleable for poly-size circuits.

We remark that Guruswami and Smith, also used random stochastic codes on logarithmic length strings to obtain their control codes, and so, we are also guaranteed that these codes have the additional pseudorandomness required for the application. See Section 6 for a specification of these properties.

Note that in our scenario, in addition to a control encoding of S, the channel C also receives additional information about S in the form of $\psi(S) = \mathrm{Enc}_{\mathrm{spr}}(m, S)$. We need to argue that receiving $\psi(S)$ does not help C to break the SS-non-malleability. Indeed, the definition of SS-non-malleability is precisely tailored so that for every poly-size circuit C, there exists a small set H_C such that when C receives $\mathrm{Enc}_{\mathrm{ctrl}}(S)$ and $\psi(S)$, it is unlikely to make $\mathrm{Dec}_{\mathrm{ctrl}}$ produce $\bar{S} \notin H_C \cup \{S\}$.

Handling the case that $\mathbf{\bar{S}_i \in H_C}$. We now have that for every channel $C \in \mathrm{Ckt}_p^{n^c}$, there exists a small set H_C of control strings, such that w.h.p. every element $\bar{S}_i \neq S$ in the list of candidates is in H_C. Our plan is to arrange things so that for every fixed control string $\bar{s} \in \{0,1\}^d$ (which in particular does not vary with S) when running $\mathrm{Dec}_{\mathrm{spr}}(\bar{Z}, \bar{s})$ (on fixed $\bar{s}$) we can detect that $\bar{s}$ is "incorrect", and discard this candidate.

Once this is achieved, by a union bound over the few $\bar{s} \in H_C$, we have that w.h.p. all incorrect control candidates are discarded, and only the correct candidate S, which decodes to the correct message m survives.

Summing up this discussion, we would like to address the scenario considered in the second question above: what happens when $\mathrm{Dec}_{\mathrm{spr}}(\bar{Z}, \bar{s})$ is applied on a corrupted codeword, with a *fixed* string $\bar{s}$.

Using evasive BSC codes to discard incorrect candidates. Loosely speaking, the construction of the code for shared private randomness, $\mathrm{Enc}_{\mathrm{spr}}(m, S)$ encodes m using a code for $\mathrm{Enc}_{\mathrm{BSC}}(m)$, and then xors the codeword with a string $W = G(S)$, where G is a pseudorandom generator for circuits of size n^c. (For completeness, we will review the construction of $\mathrm{Enc}_{\mathrm{spr}}$, which is more complicated than this oversimplified description, in Section 2.5). The decoding algorithm $\mathrm{Dec}_{\mathrm{spr}}(\bar{Z}, S)$ "reverses this operation". More specifically, when obtaining a corrupted message $\bar{Z} = Z \oplus C(Z)$, it computes $W = G(S)$, and applies $\mathrm{Dec}_{\mathrm{BSC}}(\bar{Z} \oplus W)$. This is done so that the two operations of "masking with W"', cancel each other out when $\mathrm{Enc}_{\mathrm{spr}}$ and $\mathrm{Dec}_{\mathrm{spr}}$ are applied with the same seed S.

For the sake of intuition, let us pretend that W is truly random, rather than just pseudorandom. In the scenario that we are considering, where $\mathrm{Dec}_{\mathrm{spr}}$ is applied with a fixed $\bar{s}$ (which is obviously independent of S), W is not masked out by the constant $G(\bar{s})$. In fact, $W \oplus G(\bar{s})$ is uniformly distributed. It follows that $\mathrm{Dec}_{\mathrm{spr}}(\bar{Z}, \bar{s})$ is distributed like $\mathrm{Dec}_{\mathrm{BSC}}(W \oplus C(W))$. This is precisely the scenario considered in the evasiveness experiment!

Consequently, if we use the code $(\mathrm{Enc}_{\mathrm{BSC}}, \mathrm{Dec}'_{\mathrm{BSC}})$ from Theorem 2.3, then we are guaranteed that w.h.p. $\mathrm{Dec}_{\mathrm{BSC}}$ outputs $fail$, which means that $\mathrm{Dec}_{\mathrm{spr}}(\bar{Z}, \bar{s})$ outputs $fail$, and the candidate $\bar{s}$ is discarded.

Summing up, together, SS-non-malleability and evasiveness allow us to discard all incorrect control candidates (SS-non-malleability discards candidates that are related to S, and evasiveness discards the candidates in H_C). We are therefore left with only the correct control candidate, and the decoding algorithm will return the unique (and correct) message m. (We stress that in this informal overview we are ignoring some additional technical issues, and a full proof appears in Section 7).

2.4 A Monte-Carlo construction of evasive codes for BSC_p with rate $1 - H(p)$

In this section we prove Theorem 2.1. Our plan is to use code concatenation, where the inner code encodes messages of length $O(\log n)$ and is a code for BSC_p. It is standard that this construction gives a BSC_p code, for a suitable outer code (in fact, as we are not picky with respect to decoding success probability, we can use the identity function as an outer code). We will be able to argue that the concatenated code is evasive for $\mathrm{Ckt}_p^{n^c}$ if the inner code is. (This argument appears in Section 5.4, and we will not review it here). As we are shooting for a Monte-Carlo construction, it will be sufficient to analyze a *random* inner code (as such a code can be sampled using $\mathrm{poly}(n)$ random bits). Consequently, it is sufficient to show that:[17]

Informal Theorem 2.3 (A Monte-Carlo construction of inner codes that are evasive)**.** *For every constants $p < \frac{1}{4}$, $c > 1$, and $\epsilon > 0$, there exists a constant $e > 1$, such that setting $R = (1 - H(p) - \epsilon)$, a uniformly chosen* $\mathrm{Enc} : \{0,1\}^{R \cdot e \cdot \log N} \to \{0,1\}^{e \cdot \log N}$ *(that can be sampled using* $\mathrm{poly}(N)$ *random bits) is w.h.p. evasive for* $\mathrm{Ckt}_p^{N^c}$ *(when coupled with maximum likelihood decoding up to distance p).*

In order to prove Theorem 2.3, we develop a general methodology that we describe in detail in Section 4, and is also used to prove Theorem 2.2 and the results on random stochastic codes presented in Section 1.3.2.

2.4.1 A methodology inspired by proofs of t-wise independent inequalities

We will explain our methodology in the context of proving Theorem 2.3. (A detailed formal statemet, and discussion of this method appears in Section 4). We will consider a random code with the required parameters. Namely, let $n = e \cdot \log N$ and $k = R \cdot n$, so that we are aiming for a code $\mathrm{Enc} : \{0,1\}^k \to \{0,1\}^n$. Let $K = 2^k$, and $\mathcal{X} = (\{0,1\}^n)^K$, so that elements $x \in \mathcal{X}$ can be viewed as "encoding maps" by $\mathrm{Enc}(m) = x_m$. We associate an "encoding map" $x \in \mathcal{X}$ with a decoding map Dec_x which on input $\bar{z} \in \{0,1\}^n$, finds the closest codeword (that is the index $j \in [k]$, that minimizes the relative Hamming distance $\delta(x_j, \bar{z})$), and outputs fail if this distance is larger than p.

With this notation, choosing a random code, corresponds to the experiment $x \leftarrow \mathcal{X}$. Once the code x is selected, we are interested in the evasiveness experiment, in which $z \leftarrow U_n$ is selected, and a channel $C \in \mathrm{Ckt}_p^{N^c}$ wins if $\mathrm{Dec}_x(z \oplus C(z))$ does not fail. Let us define a function $W^C(x,z)$ which outputs 1 iff "C wins on z with the code x", namely if there exists $j \in [K]$ such that $\delta(x_j, z \oplus C(z)) \le p$. This notation is set up so that in order to prove that the code is ρ-evasive for $\mathrm{Ckt}_p^{N^c}$, we need to prove that:

$$\Pr_{X \leftarrow \mathcal{X}}\left[\exists C \in \mathrm{Ckt}_p^{N^c} \text{ s.t. } \Pr_{Z \leftarrow U_n}[W^C(X,Z) = 1] > \rho\right] \text{ is small.}$$

Note that for every constant $\alpha > 0$, the number of channels in $\mathrm{Ckt}_p^{N^c}$ is bounded by $2^{N^{2c}} = 2^{2^{\frac{2c}{e} \cdot n}} \le 2^{2^{\alpha \cdot n}}$ for a sufficiently large constant e. By a union bound over all $C \in \mathrm{Ckt}_p^{N^c}$, it will be sufficient to show that for every $C \in \mathrm{Ckt}_p^{N^c}$,

$$\Pr_{X \leftarrow \mathcal{X}}\left[\Pr_{Z \leftarrow U_n}[W^C(X,Z) = 1] > \rho\right] < 2^{-2^{\alpha \cdot n}}.$$

[17] In Theorem 2.3 we discuss only evasiveness and do not mention decoding against BSC_p, this is because by Shannon's theorem, the code of Theorem 2.3 can be "trimmed" to yield a code for BSC_p. More precisely, Shannon's proof shows that while the random code Enc chosen in Theorem 2.3 is unlikely to be a code for BSC_p, it is likely that after removing at most half of the codewords, the obtained code is good for BSC_p. Note that removing codewords does not harm the evasiveness property, and is insignificant in terms of rate. Moreover, in time $\mathrm{poly}(N)$, one can identify a set of "offending codewords" that needs to be removed, and so, the trimming step, can be viewed as an explicit Monte-Carlo construction. This argument appears in Section 5.

Let $W^C_z(X) = W^C(X,z)$ so that we can view the 2^n random variables W^C_z, as random variables over choosing $X \leftarrow \mathcal{X}$. With this notation we have that:

$$\Pr_{X \leftarrow \mathcal{X}}\left[\Pr_{Z \leftarrow U_n}[W^C(X,Z) = 1] > \rho\right] = \Pr_{X \leftarrow \mathcal{X}}\left[\frac{1}{2^n} \cdot \sum_{z \in \{0,1\}^n} W^C_z > \rho\right], \qquad (1)$$

which now looks like a tail inequality for the sum of the 2^n random variables $(W^C_z)_{z \in \{0,1\}^n}$. If these random variables were independent, then we would be able to argue (by the Chernoff bound) that the probability that the sum "deviates from the expectation" is indeed doubly exponentially small.

These random variables are dependent, and not even pairwise independent. Nevertheless, by utilizing ideas from t-wise independent tail inequalities, we show that, in order to show that (1) is smaller than $2^{-2^{\alpha \cdot n}}$, it is sufficient to show that for every $q < 2^{\alpha \cdot n}$,

$$\Pr_{\substack{X \leftarrow \mathcal{X}, Z \leftarrow \{0,1\}^n \\ Z_1,\ldots,Z_q \overset{\text{wor}}{\leftarrow} \{0,1\}^n}}[W^C(X,Z) = 1 | W^C(X,Z_1) = \ldots = W^C(X,Z_q) = 1] \le \frac{\rho}{3}, \qquad (2)$$

where the notation $Z_1,\ldots,Z_q \overset{\text{wor}}{\leftarrow} \{0,1\}^n$ means that $Z_1,\ldots,Z_q \leftarrow \{0,1\}^n$ without replacement. The precise statement appears as Lemma 4.1. Intuitively, (2) is a way to account for the dependance of the random variables $(W^C_z)_{z \in \{0,1\}^n}$ as it identifies a sufficient condition for a tail inequality.

This means that we have reduced the task of achieving doubly exponentially small probabilities for (1), to obtaining a moderate bound of $\rho/3$, on a scenario where we are interested in a version of the original experiment where $Z \leftarrow \{0,1\}^n$, and an independent $X \leftarrow \mathcal{X}$ is chosen, conditioned on the event $E = \left\{W^C(X,Z_1) = \ldots, W^C(Z,Z_q) = 1\right\}$ (namely conditioned that C wins on all $Z_1,\ldots,Z_q \overset{\text{wor}}{\leftarrow} \{0,1\}^n$).

The hope is that $(X|E)$ can be shown to be "roughly the same" as X. Indeed, it possible to show that having won on $Z_1,\ldots,Z_q \overset{\text{wor}}{\leftarrow} \{0,1\}^n$, gives C only limited information about X. Loosely speaking, it allows C to learn the locations of $q \le 2^{\alpha \cdot n}$ codewords. (This is because every time C wins on Z_i, we might as well reveal to C, the index j, and the value X_j for the codeword that was decoded by Dec on $Z_j \oplus C(Z_j)$). This means that conditioning on having won on $Z_1,\ldots,Z_q$, essentially fixes q variables of $X_1,\ldots,X_K$, leaving the others uniform. (Here we are somewhat oversimplifying, and the formal proof is given in Section 5.6).

We can now prove (2) as follows: When a fresh $Z \leftarrow \{0,1\}^n$ is chosen, then for each one of the $q < 2^{\alpha \cdot n}$ codewords that were revealed to C, the probability that $Z \leftarrow \{0,1\}^n$ has relative distance $\le 2p$ from the revealed codeword is at most $2^{-(1-H(2p)) \cdot n}$, which for $p < \frac{1}{4}$, is sufficiently small, so that by a union bound, over all the $q < 2^{\alpha \cdot n}$ revealed codewords, we have that the probability that all the revealed codewords have relative Hamming distance at least $2p$ to Z, is large.

This means that the location of these codewords (that was revealed to C) is unhelpful to C, as $C \in \mathrm{Ham}_p$ cannot corrupt Z in a p fraction of positions, and "push it" to within distance p to one of these codewords.

It follows that in order to win, C needs to push the received word Z to within relative Hamming distance p to some codeword that was not revealed. To handle this case, we show that w.h.p. $Z \oplus C(Z)$ is not within relative Hamming distance p to one of the unrevealed codewords. Recall, that an unrevealed codeword, X_j is still uniformly distributed and independent of Z, and so the probability that $\delta(X_j, Z \oplus C(Z)) \le p$, is at most $2^{-(1-H(p)) \cdot n}$. There are at most $K = 2^{(1-H(p)-\epsilon) \cdot n}$ unrevealed codewords, and by a union bound, the probability that one of them is within distance p to $Z \oplus C(Z)$ is at most $2^{-\epsilon \cdot n}$.

Overall, we have shown that (1) holds for $\rho = 2^{-\alpha \cdot n}$ for a sufficiently small constant $\alpha > 0$, and this completes the proof. See Section 5.6 for a complete formal proof (rather than an oversimplified overview).

2.5 An overview of earlier codes for shared private randomness against $\mathrm{Ckt}_p^{n^c}$.

For completeness, we now explain how the construction of codes with shared private randomness $(\mathrm{Enc}_{\mathrm{spr}}, \mathrm{Dec}_{\mathrm{spr}})$ works. Recall that we want to transform a code $(\mathrm{Enc}_{\mathrm{BSC}}, \mathrm{Dec}_{\mathrm{BSC}})$ with rate R_{BSC} for BSC_p, into a code with shared private randomness $(\mathrm{Enc}_{\mathrm{spr}}, \mathrm{Dec}_{\mathrm{spr}})$ for $\mathrm{Ckt}_p^{n^c}$ that inherits the same rate, and can therefore achieve rate $R = 1 - H(p)$. The construction presented below emerged out of the works of Lipton [Lip94] and Smith [Smi07], and is essentially similar to the way it was used by Guruswami and Smith [GS16].[18]

The construction of the code for shared private randomness $(\mathrm{Enc}_{\mathrm{spr}}, \mathrm{Dec}_{\mathrm{spr}})$ will rely on a PRG G for circuits of size n^c that uses a seed of length $d = O(\log n)$.[19] On input message $m \in \{0,1\}^{k=R_{\mathrm{BSC}}\cdot n}$ and randomness $S \in \{0,1\}^d$, $\mathrm{Enc}_{\mathrm{spr}}(m,S)$ acts as follows:

- Apply the PRG on S to generate a pseudorandom string $G(S)$, and split it into two parts $W \in \{0,1\}^n$, and $\Pi \in \{0,1\}^{\log n! = \Theta(n\cdot\log n)}$. The string Π can be interpreted as a permutation $\Pi : [n] \to [n]$.
- Encode m using the BSC code to obtain $x = \mathrm{Enc}_{\mathrm{BSC}}(m)$.
- Permute the indices of x using Π^{-1}. That is, generate the string $Y \in \{0,1\}^n$ where $Y_i = x_{\Pi^{-1}(i)}$.
- Output $Z = Y \oplus W$. (That is, mask Y using W).

In the scenario of shared private randomness, the decoding algorithm $\mathrm{Dec}_{\mathrm{spr}}$ receives a corrupted codeword $\bar{Z} = Z \oplus C(Z)$, where $C \in \mathrm{Ckt}_p^{n^c}$, and the seed S used by $\mathrm{Enc}_{\mathrm{spr}}$. As $\mathrm{Dec}_{\mathrm{spr}}$ receive S, it can compute W, Π, and undo the operations performed by the decoding. Specifically, $\mathrm{Dec}_{\mathrm{spr}}(m,S)$ acts as follows:

- Unmask $\bar{Z}$ using W: Namely, compute $\bar{Y} = \bar{Z} \oplus W$.
- Undo the permutation Π, namely compute $\bar{X} \in \{0,1\}^n$, by $\bar{X}_i = \bar{Y}_{\Pi(i)}$.
- Decode the BSC code and output $\bar{M} = \mathrm{Dec}_{\mathrm{BSC}}(\bar{X})$.

Let us imagine that W and Π are uniform and independent (rather than just pseudorandom). The rationale behind this construction is the following:

- Let $C \in \mathit{Add}_p$ be an additive channel. Namely, $C(z) = e$ for a fixed noise vector $e \in \{0,1\}^n$ with relative Hamming weight $\leq p$. When $\mathrm{Dec}_{\mathrm{spr}}$ decodes, it permutes the indices of e, according to Π, generating a noise distribution that is essentially BSC_p. This means that when $\mathrm{Dec}_{\mathrm{spr}}$ applies $\mathrm{Dec}_{\mathrm{BSC}}$ on the "received message" $\bar{Z} = Z \oplus C(Z) = Z \oplus e$, it decodes correctly to the message m.[20]
- Let $C \in \mathrm{Ckt}_p^{n^c}$ be a poly-size channel. The encoded string Z is masked with W, and completely masks out m and Π. This intuitively means that the noise $E = C(Z)$ that C induces is independent of m and Π, intuitively "reducing" C to an additive channel. Using the previous item. This can be used to argue that when $\mathrm{Dec}_{\mathrm{spr}}$ applies $\mathrm{Dec}_{\mathrm{BSC}}$ on the "received message" $\bar{Z} = Z \oplus C(Z) = Z \oplus E$, it correctly decodes to the message m. The precise argument appears in Section 7.3.

[18] There are some minor differences, as it is crucial for our purposes that $d = O(\log n)$, which prevents us from using "t-wise independent permutations" that were used in previous work. This creates some complications in the analysis, see Section 7.2 for more details.

[19] Such PRGs can be constructed under complexity theoretic hardness assumptions as shown by Impagliazzo and Wigderson [IW97], but as we are shooting for a Monte-Carlo construction, we note that a random function G that stretches $O(\log n)$ bits into n^c bits, can be sampled using $\mathrm{poly}(n)$ bits, and is w.h.p. a PRG for circuits of size n^c. In other words, there is a Monte-Carlo construction of PRGs for circuits of size n^c.

[20] The noise distribution obtained by permuting e using a uniform permutation Π is not identical to BSC_p. Therefore, for this argument to go through, we will need that $(\mathrm{Enc}_{\mathrm{BSC}}, \mathrm{Dec}_{\mathrm{BSC}})$ decodes against this slightly different noise model, which we call "decoding from noise induced by a random permutation". A precise definition of this model appears in Section 5. It turns out that codes designed for BSC often apply also for this more general model, and in particular, there are explicit constructions of codes with rate $1 - H(p)$ for this noise model.

2.6 Open problems

Stochastic codes for channels that are more powerful than the encoding and decoding algorithms. Our results (as well as all aforementioned previous work) critically relies on the fact that the channel is not sufficiently strong to simulate the decoding algorithm. In particular, this means that when considering $\mathrm{Ckt}_p^{n^c}$, we use decoding algorithms that run in time n^d where $d > c$. Is this necessary? Can we obtain codes in which encoding and decoding run in time n^d for $\mathrm{Ckt}_p^{n^c}$ with $c > d$, and rate that beats (or even matches) the Gilbert-Varshamov bound? Such codes seem to imply cryptographic assumptions. Can we construct such codes based on cryptographic assumptions?

Explicit constructions of stochastic codes for classes $\mathcal{C}$ for which lower bounds are known. Our construction of stochastic codes for $\mathrm{Ckt}_p^{n^c}$ is Monte-Carlo. As explained in Section 1.2, because the rate of our stochastic codes is better than the best rate possible for Hamming channels, it follows that matching the rate that we obtain with an explicit construction, implies circuit lower bounds.

Our approach can be seen as giving a "recipe" for constructing stochastic codes with optimal rate for a class $\mathcal{C}$. Loosely speaking, we show that this can be done if one has PRGs for $\mathcal{C}$, SS-non-malleable codes for $\mathcal{C}$, and codes for BSC_p that are evasive for $\mathcal{C}$, and achieve optimal rate.

We would like to use this recipe to give explicit (rather than Monte-Carlo) constructions of stochastic codes with optimal rate against intermediate classes $\mathcal{C}$ for which we have lower bounds. One intriguing such example is AC^0, namely the class of poly-size, constant depth circuits.

Explicit constructions for $\mathrm{Ckt}_p^{n^c}$ under hardness assumptions. Another possible avenue is to give conditional explicit constructions of stochastic codes for $\mathrm{Ckt}_p^{n^c}$, *assuming* circuit lower bounds. One example of this direction was given by Shaltiel and Silbak [SS21a]. Following the Monte-Carlo construction of list-decodable stochastic codes for $\mathrm{Ckt}_p^{n^c}$ of Guruswami and Smith [GS16], Shaltiel and Silbak [SS21a] showed how to achieve the same goal with an explicit (rather than Monte-Carlo) construction under a complexity theoretic hardness assumption. Specifically, they assumed the "Imapgliazzo-Wigderson assumption" which was used by Impagliazzo and Wigderson [IW97] to prove that BPP=P.

An intriguing open problem is whether an explicit construction of codes with the parameters of Theorem 1.1 (or even just beating the Gilbert-Varshamov bound) can be based on complexity theoretic and/or cryptograhic assumptions. We remark that our techniques can be used to beat the Gilbert-Varshamov bound if one can replace the Monte-Carlo construction of Theorem 2.2 with an explicit construction based on hardness assumptions. This is because, as we explain in Section 2.1, we can obtain codes for BSC_p that are evasive against Ham_p (and in particular, against $\mathrm{Ckt}_p^{n^c}$) that are explicit, and have rate that beat the Gilbert-Varshamov bound. This is an intriguing open problem.

There are recent explicit constructions of non-malleable codes for poly-size circuits by Dachman-Soled, Komargodski and pass [DKP21], and Ball, Dachman-Soled and Loss [BDL22]. These constructions rely on strong assumptions. It is an intriguing open problem whether these techniques can yield non-malleable codes with the many additional properties that we need in our application.

Codes for BSC_p that are evasive for Ham_p. We give explicit constructions of codes for BSC_p that are evasive for Ham_p, and have rate that beats the Gilbert-Varshamov bound. More specifically, we obtain rate:

$$R^*(p) = (1 - H(p)) \cdot \frac{1-4p}{1-2p}.$$

Is it possible to achieve rate $1 - H(p)$ in this scenario? This is especially interesting in light of the previous open problem.

2.7 Organization of this paper

In Section 3 we give some preliminaries, and ingredients that we use from earlier work. In Section 4 we outline the methodology for analyzing random codes that will be used in later sections. In Section 5 we define evasive codes, and present our construction of codes for BSC_p that are also evasive. The analysis of the construction relies on the methodology of Section 4. In Section 6 we define the notion of small-set non-malleable codes, and give a Monte-Carlo construction of such codes against poly-size circuits on logarithmic length strings. The analysis of this construction relies on the methodology of Section 4. In Section 7 we present our main result, giving a Monte-Carlo construction of stochastic codes with optimal rate for $\mathrm{Ckt}_p^{n^c}$. This construction relies on the evasive codes of Section 5, and the small set non-malleable codes of Section 6. Finally, in Section 8 we present the results stated in Section 1.3.2 concerning random stochastic codes, showing that they achieve optimal rate against families with few channels. This result does not rely on Sections 5,6 and 7, and only relies on the methodology explained in Section 4.

3 Preliminaries, and ingredients used in the construction

In this section we give formal definitions of the notions and ingredients used in the construction. We also cite previous results from coding theory and pseudorandomness that are used in the construction.

General notation. We use $[n]$ to denote $\{1,\dots,n\}$. We sometimes use the notation $O_\lambda(\cdot)$ to emphasize that the constant hidden in the $O(\cdot)$ notation may depend on λ.

Tuples and distinct tuples. For a set A and an integer b, we use A^b to denote the set of tuples $(a_1,\dots,a_b)$ such that $a_1,\dots,a_b \in A$. We use $A^b(\mathrm{ds})$ to denote the set of such distinct tuples, namely, tuples $(a_1,\dots,a_b)$ such that for every $i \neq j \in [b]$, $a_i \neq a_j$. For a set B, we use A^B to denote the set of tuples $(a_i)_{i\in B}$, such that for every i, $a_i \in A$. We use $A^B(\mathrm{ds})$ to denote the set of such distinct tuples, namely, tuples $(a_i)_{i\in B}$ such that for every $i \neq j \in B$, $a_i \neq a_j$.

Probability distributions. We use U_n to define the uniform distribution on n bits. The ***statistical distance*** between two distributions P,Q over Ω is $\max_{A\subseteq\Omega}|P(A)-Q(A)|$. Given a distribution P, we use $X \leftarrow P$ to denote the experiment in which the random variable X is chosen according to P. For a set A, we use $X \leftarrow A$ to denote the experiment in which X is chosen uniformly from A. We use $X_1,\dots,X_n \leftarrow A$ to denote the experiment in which n variables are chosen uniformly from A with replacement, and $X_1,\dots,X_n \overset{\mathrm{wor}}{\leftarrow} A$ to denote the experiment where the n variables are chosen without replacement.

Shannon's entropy function. We use $H(p)$ to denote the ***Shannon binary entropy*** function: $H(p) = p\cdot\log(1/p) + (1-p)\cdot\log(1/(1-p))$. It is standard that the derivative $H'(p)$ in the interval $(0,\frac{1}{2})$ is decreasing, and satisfies $H'(p) \le \log\frac{1}{p}$. This implies that for $0 < p < \frac{1}{2}$, if $p(1+\delta) < \frac{1}{2}$ then:

$$H(p(1+\delta)) \le H(p) + \delta\cdot p\cdot H'(p) \le H(p) + \delta. \tag{3}$$

We will also rely on the standard fact that $H(\frac{1}{2}-\epsilon) = 1 - O(\epsilon^2)$.

Hamming distance and weight. The ***Hamming weight*** of $x \in [q]^n$ is $\mathit{WT}(x) = |\{i : x_i \neq 0\}|$. The ***relative Hamming weight*** of x is $\mathit{wt}(x) = \frac{\mathit{WT}(x)}{n}$. The ***Hamming distance*** between $x,y \in [q]^n$ is $\Delta(x,y) = |\{i : x_i \neq y_i\}|$. The ***relative Hamming distance*** between $x,y \in [q]^n$ is $\delta(x,y) = \frac{\Delta(x,y)}{n}$.

Channels. Let BSC_p denote the distribution over n bits, where bits are i.i.d., and each bit has probability p to be one. We will sometimes abuse the notation and think of BSC_p as a probabilistic procedure that on input $z \in \{0,1\}^n$, produces the distribution BSC_p. Let Ham_p denote the class of functions $C : \{0,1\}^n \to \{0,1\}^n$ such that for every $z \in \{0,1\}^n$, $wt(C(z)) \leq p$. Let Ckt_p^s be the class of all functions in Ham_p that can be computed by size s circuits.

3.1 Permuting strings

We will use a permutation $\pi : [n] \to [n]$ to "reorder" the bits of a string $x \in \{0,1\}^n$: The i'th bit in the rearranged string will be the $\pi(i)$'th bit in x. This is captured in the definition below.

Definition 3.1 (Permuting strings)**.** *Given a string $x \in \{0,1\}^n$ and a permutation $\pi : [n] \to [n]$. Let $\pi(x)$ denote the string $x' \in \{0,1\}^n$ with $x'_i = x_{\pi(i)}$.*

3.2 Pseudorandom generators

We need the following standard definition of pseudorandom distributions and generators.

Definition 3.2 (Pseudorandom generators)**.** *A distribution X on n bits is ϵ-**pseudorandom** for a class $\mathcal{C}$ of functions from n bits to one bit, if for every $C \in \mathcal{C}$, $|\Pr[C(X) = 1] - \Pr[C(U_n)] = 1]| \leq \epsilon$. A function $G : \{0,1\}^d \to \{0,1\}^n$ is an ϵ-**PRG** for $\mathcal{C}$ if $G(U_d)$ is ϵ-pseudorandom for $\mathcal{C}$.*

We also record the standard fact that (by a union bound) a uniformly chosen function $G : \{0,1\}^{O(\log m)} \to \{0,1\}^n$, is likely to be a PRG against any class $\mathcal{C}$ of at most 2^m functions.

Proposition 3.3 (A Monte-Carlo construction of PRGs)**.** *Let $\mathcal{C}$ be a class of at most 2^m functions from n bits to one bit. There exists a constant c such that for $d = O(\log m + \log(1/\epsilon) + \log\log(1/\gamma))$, if we consider the experiment $B_1, \ldots, B_{2^d} \leftarrow \{0,1\}^n$, and define the function $G : \{0,1\}^d \to \{0,1\}^n$ by $G(s) = B_s$ (where on the right hand side we view s as a binary number between 1 and 2^d) then:*

$$\Pr_{B_1,\ldots,B_{2^d} \leftarrow \{0,1\}^n}[G \textit{ is an } \epsilon\textit{-PRG for } \mathcal{C}] \geq 1 - \gamma.$$

3.3 Pseudorandomly chosen permutations

Let $\mathrm{UniPerm}_n$ denote the uniform distribution on the set of permutation $\pi : [n] \to [n]$. We will omit n when it is clear from the context. Fix some poly-time computable function function $F : \{0,1\}^{n^2} \to S_n$ such that $F(U_{n^2})$ is a 2^{-n}-close to $\mathrm{UniPerm}_n$. (Note that since $|S_n| = n!$ is not a power of 2, there has to be some statistical error, but using a seed sufficiently larger than $\log n!$, the error can be made small).

We will be interested in the distribution $F(G(U_d))$ where G is a function (that will later be chosen to be a PRG).

Definition 3.4 (Pseudorandomly-chosen permutation)**.** *Given a function $G : \{0,1\}^d \to \{0,1\}^{n^2}$ we define: $\pi^G : \{0,1\}^d \times [n] \to [n]$ by: $\pi^G(s,i) = (F(s))(i)$ (namely, applying the permutation $\pi = F(s)$ on i). We use π_s^G to denote the permutation π, defined by $\pi(i) = \pi^G(s,i)$. We omit G when it is clear from the context.*

We use the term "pseudorandonly-chosen permutation" to differentiate this notion from the cryptographic notion of "pseudorandom permutation" (which is different).

3.4 Averaging Samplers

The reader is referred to Goldreich's survey [Gol97] on averaging samplers.

Definition 3.5 (Averaging Samplers). *A function* $\mathrm{Samp} : \{0,1\}^n \to (\{0,1\}^m)^t$ *is an* (ϵ, δ)-**Sampler** *if for every* $f : \{0,1\}^m \to [0,1]$,

$$\Pr_{(z_1,\ldots,z_t)\leftarrow \mathrm{Samp}(U_n)}[|\frac{1}{t}\sum_{i\in[t]} f(z_i) - \frac{1}{2^m}\sum_{x\in\{0,1\}^m} f(x)| > \epsilon] \leq \delta.$$

A sampler has **distinct samples** *if for every* $x \in \{0,1\}^n$*, the* t *elements in* $\mathrm{Samp}(x)$ *are distinct.*

Zuckerman [Zuc97] showed that extractors can be viewed as samplers. Moreover, "strong extractors" translate into samplers with distinct samples. Using this connection, and the competitive extractor constructions of Guruswami, Umans and Vadhan [GUV07], we obtain the following sampler. (In fact, the construction of [GUV07] translates into a sampler with much better parameters than the one cited here).

Theorem 3.6. *For every constant* $c_1 \geq 1$ *there exists a constant* c_2*, such that for every sufficiently large* m*, and* $2^{m^{0.1}} \leq t(m) \leq 2^m$*, there is a* $(\frac{1}{m^{c_1}}, \frac{1}{m^{c_2}})$*-sampler with distinct samples* $\mathrm{Samp} : \{0,1\}^{c_2\cdot m} \to (\{0,1\}^m)^{t(m)}$*. Furthermore,* Samp *is computable in time* $t(m) \cdot \mathrm{poly}(c_2 \cdot \log m)$.

We remark that previous work in this area [GS16, SS21a, KSS19, SS21b] used "expander based samplers", rather than "extractor based ones". We need a sampler with shorter seed than what was used in previous work, and this is why we choose this sampler.

3.5 Error-Correcting Codes

In this section we give formal definitions of some of the various notions of error correcting codes used in this paper. We will also introduce less standard definitions in the next sections.

A code is a pair $(\mathrm{Enc}, \mathrm{Dec})$ of encoding and decoding maps, and different notions are obtained by considering the requirements on the decoding algorithm.

3.5.1 Standard notions of error correcting codes

We start by giving definitions of error correcting codes that covers the standard cases of Hamming channels and binary symmetric channels.

Definition 3.7 (Codes for Shannon and Hamming channels). *Let* k, n *be parameters and let* $\mathrm{Enc} : \{0,1\}^k \to \{0,1\}^n$ *and* $\mathrm{Dec} : \{0,1\}^n \to \{0,1\}^k$ *be functions. We say that* $(\mathrm{Enc}, \mathrm{Dec})$*:*

- **decodes from** t **errors***, if for every* $m \in \{0,1\}^k$ *and every* $v \in \{0,1\}^n$ *with* $\Delta(\mathrm{Enc}(m), v) \leq t$*,* $\mathrm{Dec}(v) = m$*.*
- **decodes from a distribution** P*, with success probability* $1 - \nu$*, if* P *is a distribution over* $\{0,1\}^n$*,* $0 \leq \nu \leq 1$*, and for every* $m \in \{0,1\}^k$*,* $\Pr_{e\leftarrow P}[\mathrm{Dec}(\mathrm{Enc}(m) \oplus e) = m] \geq 1 - \nu$*.*

The rate of the code is the ratio of the message length and output length of Enc*, where both lengths are measured in bits. That is the rate* $R = \frac{k}{n}$*.*

The code is **explicit** *if both encoding and decoding run in polynomial time. (Naturally, this makes sense only for a family of encoding and decoding functions with varying block length* n*, message length* $k(n)$*).*

The notion of "decoding from errors" corresponds to *Hamming channels*, where the decoding algorithm needs to decode (or list-decode) from a certain distance. We remark that it is standard that a code is decodable from t errors if and only if the Hamming distance between every two codewords is at least $2t+1$.

The notion of decoding from P covers the case of *BSC channels*, where P is taken to be the distribution BSC_p of n i.i.d. bits where each bit has probability p to be one.

3.5.2 Stochastic Codes for a class of channels

In this section we give a precise formal definition of the notion of stochastic codes for a class of channels (that was already explained in the introduction).

Definition 3.8 (Stochastic codes for channels)**.** *Let k, n, d be parameters and let* $\mathrm{Enc} : \{0,1\}^k \times \{0,1\}^d \to \{0,1\}^n$*, and* $\mathrm{Dec} : \{0,1\}^n \to \{0,1\}^k$ *be functions. Let $\mathcal{C}$ be a class of functions from n bits to n bits. We say that* $(\mathrm{Enc}, \mathrm{Dec})$ *is a stochastic code for "channel class" $\mathcal{C}$, with success probability $1-\nu$, if for every $m \in \{0,1\}^k$ and every $C \in \mathcal{C}$, setting $X = \mathrm{Enc}(m, U_d)$, we have that*

$$\Pr[\mathrm{Dec}(X \oplus C(X)) = m] \geq 1 - \nu.$$

The rate of the code is the ratio of the message length and output length of Enc*, where both lengths are measured in bits. That is the rate $R = \frac{k}{n}$.*

The code is **explicit** *if both encoding and decoding run in polynomial time. (Naturally, this makes sense only for a family of encoding and decoding functions with varying block length n, message length $k(n)$ and randomness $d(n)$).*

A **Monte-Carlo** *stochastic code with success $1-\nu$ for a class $\mathcal{C}$, that uses q bits of Monte-Carlo randomness, with Monte-Carlo error $\eta > 0$, is a pair of functions* $\mathrm{Enc} : \{0,1\}^q \times \{0,1\}^k \times \{0,1\}^d \to \{0,1\}^n$ *and* $\mathrm{Dec} : \{0,1\}^q \times \{0,1\}^n \to \{0,1\}^k$*, such that with probability $1-\eta$ over choosing $y \leftarrow U_q$, the pair of functions* $\mathrm{Enc}_y(m,s) = \mathrm{Enc}(y,m,s)$ *and* $\mathrm{Dec}_y(v) = \mathrm{Dec}(y,v)$ *form a stochastic-code for $\mathcal{C}$ with success $1-\nu$.*

A Monte-carlo stochastic code is **explicit** *if* $\mathrm{Enc}, \mathrm{Dec}$ *run in time polynomial in n, and q is a polynomial in n. (Naturally, this makes sense only for a family of encoding and decoding functions with varying block length n, message length $k(n)$, seed length $d(n)$ and Monte-Carlo randomness length $q(n)$).*[21]

3.6 t-wise independent tail inequalities

We will be interested in obtaining tail inequalities on a random variable which is the sum of n independent indicator random variables that are roughly t-wise independent.

The following Lemma was proven in [KSS19].

Lemma 3.9 (tail bounds for variables that are roughly t-wise independent)**.** *Let $X_1, ..., X_n$ be binary random variables, such that for every set of distinct t indices $i_1, \cdots, i_t \in [n]$, $\Pr[X_{i_1} = \ldots = X_{i_t} = 1] \leq \mu^t$. If $0 < \delta \leq 1$ and $t \leq \frac{\delta \cdot \mu \cdot n}{2}$ then*

$$Pr[\sum_{j=1}^{n} X_j \geq (1+\delta) \cdot \mu \cdot n] \leq e^{-\Omega(\delta t)}$$

[21] An alternative view of Monte-Carlo constructions (that is sometimes preferable) is that a Monte-Carlo construction is a randomized algorithm that tosses $q(n)$ coins, and produces circuits Enc, Dec, such that with probability $1-\eta(n)$, the obtained circuits have the required property.

4 A methodology inspired by proofs of t-wise independent inequalities

In this paper we develop a methodology to show that random codes (or sometimes random stochastic codes) are "good" in the sense that no channel C from a family $\mathcal{C}$ that contains "few" channels, can win in a some random experiment, that occurs *after* the code is constructed. (Here, "few" is typically $2^{2^{\alpha \cdot n}} \ll 2^{2^n}$ for a small constant $\alpha > 0$. Note that this is quite large. For example, $\mathrm{Ckt}_p^{n^c}$ is of much smaller size, as the number of channels in $\mathrm{Ckt}_p^{n^c}$ is bounded by $2^{n^{2c}}$).

Examples are the scenarios that were considered in Section 1 and Section 2, specifically:

Stochastic codes for $\mathcal{C}$: In this scenario, once a stochastic code is chosen, the random experiment is that some message m is encoded using a random seed S, and $C \in \mathcal{C}$ wins, if it can corrupt the codeword, so that decoding does not produce the message m.

SS-Non-Malleable codes for $\mathcal{C}$: In this scenario, once a stochastic code is chosen, the random experiment is that a random message M is encoded using a random seed S, and $C \in \mathcal{C}$ wins, if it can corrupt the codeword, so that decoding produces a message that is not in a small set H_C (where H_C is determined in advance as a function of C).

Codes for BSC_p that are evasive for $\mathcal{C}$: In this scenario, once a (standard) code for BSC_p is chosen, the random experiment is that a random $Z \leftarrow U_n$ is chosen, and $C \in \mathcal{C}$ wins, if it can corrupt Z, so that the decoding will not fail.

In this paper we develop a technique to analyze random codes in this scenario, which is a key ingredient in obtaining our results, and may be of independent interest.

4.1 Overview of the setting of random codes with an additional random experiment

When choosing a random code for the scenarios described above, there are two experiments taking place:

- First a random code is selected. Let us denote this experiment as choosing a code $x \leftarrow \mathcal{X}$. The precise definition of $\mathcal{X}$ depends on the application. It will typically be the set of all codes (or stochastic codes) of a certain rate.
- Once $x \leftarrow \mathcal{X}$, is chosen and fixed, another random experiment takes place. In this experiment, a new random variable $z \leftarrow \mathcal{Z}$ is selected. (The choice of $\mathcal{Z}$ will depend on which scenario is considered). In each scenario, there is a game, and the goal is to show that w.h.p. over $x \leftarrow \mathcal{X}$ every $C \in \mathcal{C}$ has small probability to win for $z \leftarrow \mathcal{Z}$.

In other words, we would like to show statements of the form:

$$\Pr_{x \leftarrow \mathcal{X}}\left[\exists C \in \mathcal{C} \text{ s.t. } \Pr_{z \leftarrow \mathcal{Z}}[\text{C wins with "too large" probability}]\right] \text{ is small.}$$

The natural approach is to obtain this by a union bound over all $C \in \mathcal{C}$, and this will follow if we show that for every $C \in \mathcal{C}$:

$$\Pr_{x \leftarrow \mathcal{X}}\left[\Pr_{z \leftarrow \mathcal{Z}}[\text{C wins with "too large" probability}]\right] \text{ is significantly smaller than } \frac{1}{|\mathcal{C}|}.$$

For a fixed $C \in \mathcal{C}$ we will define a function $W(x,z)$ which answers one if C wins on x, z. Let $W_z(X)$ be the random variable (over the choice of $X \leftarrow \mathcal{X}$) that answers $W(X,z)$. This is done so that if we define

$$W_{\text{avg}}(X) = \frac{1}{|\mathcal{Z}|} \sum_{z \in \mathcal{Z}} W_z(X),$$

What we need to prove looks like a tail inequality for the sum of the random variables W_z. More specifically, we need to prove that:

$$\Pr_{X \leftarrow \mathcal{X}} \left[W_{\text{avg}} = \frac{1}{|\mathcal{Z}|} \sum_{z \in \mathcal{Z}} W_z \text{ is “too large”} \right] \text{ is significantly smaller than } \tfrac{1}{|\mathcal{C}|}.$$

Recall that we are interested in cases where $\mathcal{C}$ is of size doubly expoential in n, and $\mathcal{Z}$ is of size exponential in n. This is exactly the setting in which tail bounds for independent variables (like the Chernoff bound) perform best, and it is standard to get such a bound if the random variables W_z are independent.

Unfortunately, in the settings that we will consider, these variables will not be independent or even pairwise independent.

4.2 Formal statement of the methodology and a recipe for applications

Continuing the earlier discussion, we now state a “tail inequality” that will be useful in all three scenarios above, and may be of independent interest.

Lemma 4.1. *Consider a probability space defined by some random variable X over a set $\mathcal{X}$, and let $\mathcal{Z}$ be a set of size n. Let $W : \mathcal{X} \times \mathcal{Z} \to \{0,1\}$ be some function, and let*

$$W_{\text{avg}}(x) = \frac{1}{|\mathcal{Z}|} \cdot \sum_{z \in \mathcal{Z}} W(x,z).$$

For every real numbers $0 \le \mu \le 1$, $\delta > 0$, and every integer $0 \le t \le \frac{\delta \cdot \mu \cdot n}{2}$, if for every integer $0 \le q < t$,

$$\Pr_{\substack{X; Z \leftarrow \mathcal{Z} \\ Z_1, \ldots, Z_q \overset{\text{wor}}{\leftarrow} \mathcal{Z}}} [W(X,Z) = 1 \mid W(X,Z_1) = \ldots = W(X,Z_q) = 1] \le \mu,$$

then

$$\Pr_X [W_{\text{avg}}(X) > (1+\delta) \cdot \mu] \le e^{-\frac{\delta \cdot t}{2}}.$$

Lemma 4.1 reduces the task of showing a tail-bound to that of analyzing a case in which $X \leftarrow \mathcal{X}$ and $Z \leftarrow \mathcal{Z}$ are chosen independently, and we ask whether C wins in this game, *conditioned* on the event that C already won q games on random $Z_1, \ldots, Z_q \overset{\text{wor}}{\leftarrow} \mathcal{Z}$ (namely, on $Z_1, \ldots, Z_q$ that were chosen from $\mathcal{Z}$ without replacement).

The advantage is that while X is affected by this conditioning, Z is not, and remains independent of X. This means, that in order to use this lemma, we only need to understand how X is affected, when conditioned on C having won on q independently chosen $Z_1, \ldots, Z_q \overset{\text{wor}}{\leftarrow} \mathcal{Z}$.

A general recipe for applying Lemma 4.1. In the three scenarios mentioned above, we will be able (with some considerable effort) to show that this conditioning does not affect the "code" X too much, under this conditioning. This will allow us to analyze this single game, and by Lemma 4.1, we will obtain the bounds in the three scenarios discussed above.

More precisely, in each one of the scenarios described above, we will show that the event:

$$T = \{W(X, Z_1) = \ldots = W(X, Z_q) = 1\}$$

can be expressed as a disjoint union of events, namely:

$$T = \bigcup_i E_i$$

This means that it is sufficient to bound:

$$\Pr_{\substack{X;Z \leftarrow \mathcal{Z} \\ Z_1,\ldots,Z_q \overset{\text{wor}}{\leftarrow} \mathcal{Z}}} [W(X, Z) = 1 \mid E_i] \le \mu,$$

for every event E_i. We will then show that every event E_i is "simple" in the sense that the distribution of X, in the experiment

$$\begin{matrix} X \leftarrow \mathcal{X} \\ Z_1, \ldots, Z_q \overset{\text{wor}}{\leftarrow} \mathcal{Z} \end{matrix} \mid E_i.$$

is not that different than the initial distribution of X, and "many" codewords, are still "somewhat random" conditioned on E_i. This means that in order to complete the proof, all we have to do is show that C is unlikely to win in an experiment, where X is chosen from the conditioned distribution (which is very similar to a random code) and $Z \leftarrow \mathcal{Z}$ is chosen independently.

Applications of the approach in this paper. This approach is used in Section 5 to analyze random codes, and show that they are evasive. (An overview of this argument, which is the simplest of the three, appears in Section 2.4.1). The approach is also used in Section 6 to analyze random stochastic codes and show that they are SS-non-malleable. The approach is used in Section 8 to show that random stochastic codes with optimal rate, decode against small classes of channels.

In some cases, we will be able to leverage these random codes into explicit Monte-Carlo constructions, by using them as "inner codes" on short blocks in some form of "code concatenation". See Section 2 for an overview.

4.3 Proof of Lemma 4.1

The proof of Lemma 4.1 uses ideas that are employed in proving t-wise independent tail inequalities [BR94, SSS95, DHRS07, KSS19], see e.g., Lemma 3.9. In the proof that we give below, we generalize a folklore combinatorial approach to t-wise independent tail inequalities (in contrast to some earlier proofs for t-wise independent tail inequalities that generalize Chebichev's in equality to higher moments). This approach is also somewhat related to the approach used in the alternative proof of Chernoff's bound given by Impagliazzo and Kabanets [IK10].

We will first prove the following lemma (which discusses t games) and will then deduce Lemma 4.1.

Lemma 4.2. *Consider a probability space defined by some random variable X over a set $\mathcal{X}$, and let $\mathcal{Z}$ be a set of size n. Let $W : \mathcal{X} \times \mathcal{Z} \to \{0,1\}$ be some function, and let*

$$W_{\mathrm{avg}}(x) = \frac{1}{|\mathcal{Z}|} \cdot \sum_{z \in \mathcal{Z}} W(x,z).$$

For every real numbers $0 \le \mu \le 1$, every $\delta > 0$ and every integer $t \le \frac{\delta \cdot \mu \cdot n}{2}$, if

$$\Pr_{X; Z_1,\dots,Z_t \overset{\mathrm{wor}}{\leftarrow} \mathcal{Z}}[W(X,Z_1) = \ldots = W(X,Z_t) = 1] \le \mu^t,$$

then

$$\Pr_X[W_{\mathrm{avg}}(X) > (1+\delta)\cdot\mu] \le e^{-\frac{\delta \cdot t}{2}}.$$

Proof. For every $x \in \mathcal{X}$ we define:

- $n_x = |\{z \in \mathcal{Z} : W(x,z) = 1\}|$.
- $\bar{n}_x = |\{z_1,\dots,z_t \in \mathcal{Z} :\ z_1,\dots,z_t$ are distinct, and $W(x,z_1),\dots,W(x,z_t) = 1\}|$.
- $\hat{n}_x = \frac{\bar{n}_x}{\binom{n}{t}}$.

This definition is tailored so that:

$$\hat{n}_x = \Pr_{X; Z_1,\dots,Z_t \overset{\mathrm{wor}}{\leftarrow} \mathcal{Z}}[W(X,Z_1) = \ldots = W(X,Z_t) = 1 \mid X = x].$$

Let

$$A = \{x \in \mathcal{X} : W_{\mathrm{avg}}(x) \ge (1+\delta)\cdot\mu\},$$

and let

$$\ell = (1+\delta)\cdot\mu\cdot n.$$

By definition, for every $x \in A$, we have that $n_x \ge \ell$. It follows that for every $x \in A$,

$$\begin{aligned}
\bar{n}_x &\ge \binom{\ell}{t} \\
&= \frac{\ell\cdot(\ell-1)\cdot\ldots\cdot(\ell-t+1)}{t!} \\
&\ge \frac{(\ell-t)^t}{t!} \\
&\ge \frac{\left((1+\frac{\delta}{2})\cdot\mu\cdot n\right)^t}{t!},
\end{aligned}$$

where the last inequality is by the requirement on t. Therefore, for every $x \in A$:

$$\begin{aligned}
\hat{n}_x &= \frac{\bar{n}_x}{\binom{n}{t}} \\
&\ge \frac{\frac{\left((1+\frac{\delta}{2})\cdot\mu\cdot n\right)^t}{t!}}{\frac{n\cdot(n-1)\cdot\ldots\cdot(n-t+1)}{t!}} \\
&\ge \frac{\left((1+\frac{\delta}{2})\cdot\mu\cdot n\right)^t}{n^t} \\
&\ge ((1+\frac{\delta}{2})\cdot\mu)^t
\end{aligned}$$

We have that:

$$
\begin{aligned}
\mu^t &\geq \Pr_{X;Z_1,\ldots,Z_t \overset{\text{wor}}{\leftarrow} \mathcal{Z}}[W(X,Z_1) = \ldots = W(X,Z_t) = 1] \\
&= \sum_{x \in \mathcal{X}} \Pr_{X;Z_1,\ldots,Z_t \overset{\text{wor}}{\leftarrow} \mathcal{Z}}[W(X,Z_1) = \ldots = W(X,Z_t) = 1 \mid X = x] \cdot \Pr[X = x] \\
&= \sum_{x \in \mathcal{X}} \hat{n}_x \cdot \Pr[X = x] \\
&\geq \sum_{x \in A} \hat{n}_x \cdot \Pr[X = x] \\
&\geq ((1 + \frac{\delta}{2}) \cdot \mu)^t \cdot \sum_{x \in A} \Pr[X = x] \\
&\geq ((1 + \frac{\delta}{2}) \cdot \mu)^t \cdot \Pr[X \in A] \\
&\geq ((1 + \frac{\delta}{2}) \cdot \mu)^t \cdot \Pr_X[W_{\text{avg}}(X) > (1+\delta) \cdot \mu]
\end{aligned}
$$

We conclude that:

$$
\begin{aligned}
\Pr_X[W_{\text{avg}}(X) > (1+\delta) \cdot \mu] &\leq \frac{\mu^t}{((1+\frac{\delta}{2}) \cdot \mu)^t} \\
&\leq \frac{1}{(1+\frac{\delta}{2})^t} \\
&\leq e^{-\frac{\delta \cdot t}{2}}
\end{aligned}
$$

□

We now derive Lemma 4.1 from Lemma 4.2.

Proof. (of Lemma 4.1) Lemma 4.1 follows by using Lemma 4.2. More specifically, we show that the conditions on Lemma 4.1 immediately imply the conditions of Lemma 4.2. Let $A_i = \{W(X,Z_i) = 1\}$.

$$
\Pr_{X;Z_1,\ldots,Z_t \overset{\text{wor}}{\leftarrow} \mathcal{Z}}[A_1 \cap \ldots \cap A_t] = \prod_{0 \leq q < t} \Pr_{X;Z_1,\ldots,Z_t \overset{\text{wor}}{\leftarrow} \mathcal{Z}}[A_{q+1} \mid A_1 \cap \ldots \cap A_q].
$$

Thus, it is sufficient to show that for every $0 \leq q < t$,

$$
p_q = \Pr_{X;Z_1,\ldots,Z_t \overset{\text{wor}}{\leftarrow} \mathcal{Z}}[A_{q+1} \mid A_1 \cap \ldots \cap A_q] \leq \mu.
$$

We note that the event above does not consider $Z_{q+2}, \ldots, Z_t$ and therefore:

$$
\begin{aligned}
p_q &= \Pr_{X;Z_1,\ldots,Z_{q+1} \overset{\text{wor}}{\leftarrow} \mathcal{Z}}[A_{q+1} \mid A_1 \cap \ldots \cap A_q] \\
&\leq \Pr_{\substack{X;Z_1,\ldots,Z_q \overset{\text{wor}}{\leftarrow} \mathcal{Z} \\ Z_{q+1} \leftarrow \mathcal{Z}}}[A_{q+1} \mid A_1 \cap \ldots \cap A_q],
\end{aligned}
$$

where the last inequality is because choosing Z_{q+1} independently of $Z_1, \ldots, Z_q$ only increases the probability. This concludes the proof. □

5 Evasive codes

In this section, we aim to construct explicit codes with the following properties:

- Decoding from BSC_p for every $0 \le p < \frac{1}{4}$. (In fact, we will need a related, but stronger property of decoding from "errors induced by a random permutation" that we will explain in Section 5.2).
- Large as possible rate $R(p)$. We aim to beat the Gilbert Varshamov bound of $R^{GV}(p) = 1 - H(2p)$, and the best we can hope for is $R(p) = 1 - H(p)$.
- "Evasiveness" against $\mathrm{Ckt}_p^{n^c}$.

We give formal definitions of what we mean by "evasiveness" in Section 5.1 and what we mean by codes for errors induced by random permutations in Section 5.2. Let us start by explaining the big picture.

Motivation and background to evasive codes. The notion of evasiveness that we give below, is essentially similar to that considered by Shaltiel and Silbak [SS21b] (A major difference is that we will be interested in the case where channels are poly-size circuits, rather than space bounded channels). Shaltiel and Silbak [SS21b] introduced an approach to convert a *list-decodable* stochastic code against space bounded channels [GS16, SS21a, KSS19] into a *uniquely-decodable* stochastic code against space bounded channels. As part of this approach one requires a (standard, non-stochastic) code with decoding from BSC_p (actually, from "errors induced by a random permutation") that is also "evasive" against space bounded channels. The rate of the final stochastic code that is constructed using this approach is inherited from the rate achieved for the (standard code) for BSC_p.

Using this approach (and building upon previous constructions of list-decodable stochastic codes for space bounded channels [GS16, KSS19]) Shaltiel and Silbak [SS21b] constructed stochastic codes against space bounded channels that match the rate $R^{\mathrm{BSC}}(p) = 1 - H(p)$ of binary symmetric channels. A component in this construction is a construction of a (standard) code with rate $R = 1 - H(p)$ that decodes from BSC_p and is also "evasive" against space bounded channels.

Codes that are evasive against poly-size circuits. We would like to imitate the overall approach of [SS21b] replacing space bounded channels with poly-size circuits. We will construct (standard) codes that decode from BSC_p, and are evasive against *poly-size circuits*, rather than *space bounded channels*. We stress that unlike space bounded channels, for which we have lower bounds, poly-size circuits are quite powerful, and we will need very different techniques. The main results of this section are:

- We give an explicit construction of such codes with rate
$$R^*(p) = (1 - H(p)) \cdot \frac{1 - 4p}{1 - 2p}.$$
This rate is larger than the Gilbert-Varshamov bound $R^{GV}(p) = 1 - H(2p)$ for every $0 < p < \frac{1}{4}$.
- We give an explicit Monte-Carlo construction of such codes with rate $R(p) = 1 - H(p)$. This matches the rate $R^{\mathrm{BSC}}(p) = 1 - H(p)$ of binary symmetric channels.

Organization of this section. In Section 5.1 we give a formal definition of evasiveness. In Section 5.2 we give a formal definition of codes for errors induced by random permutations. In Sections 5.3 and 5.4 we present some results on evasive codes, including the two main results above (stated as Theorem 5.5 and Theorem 5.7). In the remainder of the section we prove the two theorems.

5.1 Definition of evasiveness

We start by giving an informal definition of the evasiveness property (we will state things more precisely below). This description is similar to the one given in Section 2.1.

The evasiveness experiment: Given $\mathrm{Enc} : \{0,1\}^k \to \{0,1\}^n$ and $\mathrm{Dec} : \{0,1\}^n \to \{0,1\}^k \cup \{fail\}$, and a channel C: rather than giving the channel C a codeword of Enc to corrupts, we will be interested in the behavior of the channel and decoding algorithm on a ***uniformly chosen*** string. Specifically, we consider the following experiment:

- A uniform $Z \leftarrow U_n$ is chosen.
- The "received word" $V = Z \oplus C(Z)$ is obtained when the channel C "corrupts" Z.
- We apply $\mathrm{Dec}(V)$ and will say that the code $(\mathrm{Enc}, \mathrm{Dec})$ is *evasive* if the probability that $\mathrm{Dec}(V) \neq fail$ is small.

This is stated formally in the definition below.

Definition 5.1 (Evasive codes)**.** *Let* $\mathrm{Enc} : \{0,1\}^k \to \{0,1\}^n$ *and* $\mathrm{Dec} : \{0,1\}^n \to \{0,1\}^k \cup \{fail\}$*. We say that* $(\mathrm{Enc}, \mathrm{Dec})$ *are* ρ*-evasive for a function* $C : \{0,1\}^n \to \{0,1\}^n$ *if:*

$$\Pr_{Z \leftarrow U_n}[\mathrm{Dec}(Z \oplus C(Z)) \neq fail] \leq \rho.$$

We say that $(\mathrm{Enc}, \mathrm{Dec})$ *are* ρ*-evasive for a class* $\mathcal{C}$ *if* $(\mathrm{Enc}, \mathrm{Dec})$ *are* ρ*-evasive for every* C *in* $\mathcal{C}$*.*

Naturally, evasiveness is only interesting when coupled with some additional decoding properties of the code, like in our case decoding from BSC_p. When considering codes for BSC_p we will assume that the decoding algorithm Dec changes its answer from $m \in \{0,1\}^k$ to $fail$ on a "received word" $v \in \{0,1\}^n$, if $\delta(\mathrm{Enc}(m), z)$ is slightly larger than p. This is w.l.o.g. as a BSC_p channel has exponentially small probability to corrupt $\mathrm{Enc}(m)$ to such a z.

5.2 Decoding from errors induced by a random permutation

As is the case of earlier work on codes for bounded channels [GS16, SS21a, KSS19, SS21b] we will actually be interested in evasive codes that decode not just from BSC_p, but in a related (and more general) setup that we now explain. The definition below uses the notion of permuting strings from Definition 3.1.

Definition 5.2 (Noise induced by a distribution on permutations)**.** *Let* Π *be a distribution over permutations* $\pi : [n] \to [n]$*. For every* $e \in \{0,1\}^n$*, we can consider the "noise distribution"* $\Pi(e)$*, and let* Perm_p^{Π} *denote the class of all such distributions over all choices of* $e \in \{0,1\}^n$ *such that* $wt(e) \leq p$*.*

We say that a pair $(\mathrm{Enc}, \mathrm{Dec})$ *decodes from* Perm_p^{Π} *if it decodes from every distribution in* Perm_p^{Π}*.*

Recall that we use $\mathrm{UniPerm}_n$ to denote the uniform distribution on permutations on $[n]$ and omit n when it is clear from the context. Note that for every $e \in \{0,1\}^n$ such that $wt(e) = p$, the distribution $\mathrm{UniPerm}(e)$ is "somewhat similar" to BSC_p. More formally, individual bits of $\mathrm{UniPerm}(e)$ are distributed like individual bits of BSC_p, and while bits of $\mathrm{UniPerm}(e)$ are not independent, the correlation between "not too many" of them is "small". By the same rationale the distributions in $\mathrm{Perm}_p^{\mathrm{UniPerm}}$ are "somewhat similar" to the distributions in $\left\{\mathrm{BSC}_{p'} : p' \leq p\right\}$.

This similarity can be used to show that codes designed to decode from BSC_p, often also decode from $\mathrm{Perm}_p^{\mathrm{UniPerm}}$. We will be relying on such a code construction, which we now state. These constructions rely on good "standard codes" with high rate (specifically, previous work relied on the codes of [GI05, KMRS17]).

Theorem 5.3 ([Smi07, GS16, SS21a, KSS19])**.** *Let $R(p) = 1 - H(p)$. For every constant $0 \leq p < \frac{1}{4}$, and every sufficiently small constant $\epsilon > 0$, there exist infinitely many n, and functions $\mathrm{Enc} : \{0,1\}^{k(n)=(R(p)-\epsilon)\cdot n} \to \{0,1\}^n$, $\mathrm{Dec} : \{0,1\}^n \to \{0,1\}^{k(n)}$ such that:*

- $(\mathrm{Enc}, \mathrm{Dec})$ *decode from* $\mathrm{Perm}_p^{\mathrm{UniPerm}}$ *with success probability* $1 - 2^{-\Omega(n^{0.1})}$.
- $(\mathrm{Enc}, \mathrm{Dec})$ *are explicit.*
- $(\mathrm{Enc}, \mathrm{Dec})$ *has "monotone decoding" meaning that for every $m \in \{0,1\}^{k(n)}$ and every $e, e' \in \{0,1\}^n$ such that for every $i \in [n]$, $e_i \leq e'_i$,*

$$\mathrm{Dec}(\mathrm{Enc}(m) \oplus e') = m \Rightarrow \mathrm{Dec}(\mathrm{Enc}(m) \oplus e) = m.$$

These constructions are based on concatenating an outer code with rate roughly $1 - \epsilon/2$ (that decodes from few errors) with a random inner code of rate $1 - H(p) - \epsilon/2$ that decodes from BSC_p. The monotonicity property (that is not stated explicitly in these results) is a byproduct of this concatenation approach. We also remark that the result is stated for infinitely many n, but this subset of integers is very dense, and depends on the density of block lengths n achieved by existing constructions of the outer codes.

Remark 5.4 (Uniform, t-wise, and pseudorandomly chosen permutations)**.** *The previous work on stochastic codes for bounded channels [Smi07, GS16, SS21a, KSS19, SS21b] considered distributions Π that are "almost t-wise independent" rather than completely uniform. In fact, Theorem 5.3 is proven for almost t-wise independent permutations (for a sufficiently large $t < n$) but obviously follows for a uniform permutation.*

The advantage of t-wise independent permutations on $[n]$ is that they can be sampled using only $d = O(t \cdot \log n)$ random bits [KNR09], which for $t = \mathrm{polylog}(n)$ allows $d = \mathrm{polylog}(n) \ll n$. This is in contrast to a random permutation that requires $d = \Omega(n \log n)$ random bits.

The approach of Guruswami and Smith [GS16] and later work [SS21a, KSS19, SS21b] critically relies on distributions over permutation that can be sampled using only $d = o(n)$ random bits. (Loosely speaking, this is because the "seed" used to sample the permutation is "appended" to the encoding, and we don't want the rate to substantially decrease).

In our setting, the parameters will be even tighter, and it will be crucial that $d = O(\log n)$. This means that we cannot afford "almost t-wise independent permutations" for $t = \omega(1)$.

Instead, we will make use of the fact that in our setting, we have access to PRGs with exponential stretch against poly-size circuits. We will "derandomize" the random permutation by using a PRG G to stretch $d = O(\log n)$ bits into $\Theta(n \log n)$ bits that are used to sample a uniform permutation, as explained in Section 3.3. This means that we will be interested in decoding from $\mathrm{Perm}_p^{\pi^G_{U_d}}$, where π^G is the family of permutations defined in Section 3.3.

By taking a PRG that fools $\mathrm{Dec}_{\mathrm{BSC}}$, we will be able to argue that decoding from $\mathrm{Perm}_p^{\mathrm{UniPerm}}$ implies decoding from $\mathrm{Perm}_p^{\pi^G_{U_d}}$, paying an additional error factor in the success probability.

5.3 An explicit evasive code for Ham_p that beats the Gilbert-Varshamov bound

Given $0 \leq p < \frac{1}{4}$, and a constant $c > 1$, we would like to construct codes with the largest possible rate that decode from $\mathrm{Perm}_p^{\mathrm{UniPerm}}$, and are evasive against $\mathrm{Ckt}_p^{n^c}$.

Somewhat surprisingly, we will explicitly construct codes with rate that improves on the Gilbert-Varshamov bound, that are evasive not only for the computationally bounded class of $\mathrm{Ckt}_p^{n^c}$, but in fact, to the unbounded class Ham_p of Hamming channels!

Theorem 5.5 (Explicit evasive codes for Ham_p that beat the Gilbert-Varshamov bound)**.** *Let*

$$R^*(p) = (1 - H(p)) \cdot \frac{1-4p}{1-2p}.$$

For every constant $0 < p < \frac{1}{4}$, and every sufficiently small constant $\epsilon > 0$, there exists a constant $\beta > 0$ and infinitely many n, such that there are functions $\mathrm{Enc} : \{0,1\}^{k(n)=(R^*(p)-\epsilon)\cdot n} \to \{0,1\}^n$, $\mathrm{Dec} : \{0,1\}^n \to \{0,1\}^{k(n)} \cup \{fail\}$ that satisfy:

- $(\mathrm{Enc}, \mathrm{Dec})$ *are explicit.*
- $(\mathrm{Enc}, \mathrm{Dec})$ *decode from* $\mathrm{Perm}_p^{\mathrm{UniPerm}}$ *with success probability* $1 - 2^{-n^{0.09}}$.
- $(\mathrm{Enc}, \mathrm{Dec})$ *are $2^{-\beta\cdot n}$-evasive for* Ham_p*.*
- $R^*(p) > R^{GV}(p) = 1 - H(2p)$.

The proof of Theorem 5.5 appears in Section 5.5.

Random codes that beat the GV bound are not evasive for Ham_p. We now explain why we think Theorem 5.5 is surprising. It is easy to see that any code with rate $R(p) < R^{GV}(p) = 1 - H(2p)$ (namely a code that does not beat the Gilbert Varshamov bound) is evasive against the class Ham_p (of unbounded channels).[22] It is also easy to see that a random code with rate that beats the Gilbert-Varshamov bound is unlikely to be evasive against Ham_p, which means that random codes do not satisfy the properties in Theorem 5.5. This is stated formally below.

Proposition 5.6 (Random codes with rate that beats the GV-bound are not evasive for Ham_p)**.** *Let $R(p) > R^{GV}(p) = 1 - H(2p)$. For every $0 < p < \frac{1}{4}$, and for every sufficiently large n, if we choose a uniform function $\mathrm{Enc} : \{0,1\}^{R(p)\cdot n} \to \{0,1\}^n$, then with probability $1 - e^{-2^{\Omega(n)}}$ it holds that for every $z \in \{0,1\}^n$, there exists $m \in \{0,1\}^{R(p)\cdot n}$ such that $\delta(\mathrm{Enc}(m), z) \le 2p$.*

Proof. Consider the experiment in which a random code with rate $R(p)$ is chosen. For every $z \in \{0,1\}^n$, the probability that for every codeword $c \in \{0,1\}^{R\cdot n}$, z is not within relative distance $2p$ to c, is (by independence of the Rn codewords) at most $(1-v)^{2^{Rn}}$ where v is the volume of a Hamming ball of relative radius $2p$. As $v \ge 2^{(1-H(2p)-o(1))\cdot n}$ we have that this is bounded as follows:

$$(1-v)^{2^{R\cdot n}} \le (1 - 2^{-(1-H(2p)-o(1)\cdot n)})^{R\cdot n} \le e^{-2^{-(1-H(2p)-o(1))\cdot n} \cdot 2^{R\cdot n}} \le e^{-2^{\alpha\cdot n}},$$

For some constant $\alpha > 0$. Therefore, by a union bound on the 2^n choices for z, we have that with probability $1 - e^{-2^{\alpha'\cdot n}}$ for some $\alpha' > 0$, every $z \in \{0,1\}^n$ has a codeword c that is within relative Hamming distance $2p$. □

[22] This is because with high probability a random word Z has relative distance larger than $2p$ from any codeword. This means that if (an unbounded channel) examines Z and induces p relative errors, the corrupted word is still not within distance p to a codeword, and therefore will be rejected by a decoding algorithm that decodes from p relative errors.

Interpretation of Proposition 5.6. Proposition 5.6 says that if we choose a random encoding map Enc, and pair it with the "maximum likelihood decoder" (that on input $v \in \{0,1\}^n$, returns the message $m \in \{0,1\}^k$, such that $\delta' = \delta(\mathrm{Enc}(m), v)$ is minimal if $\delta' \leq p$, and rejects otherwise) then it is likely that a channel C in Ham_p can win in the evasiveness game with probability 1, because it is likely that for every $z \in \{0,1\}^n$, there exists an $m \in \{0,1\}^k$ such that $\delta(\mathrm{Enc}(m), v) \leq 2p$, and the channel C can corrupt v to cover "half the distance" towards $\mathrm{Enc}(m)$, so that the decoding will cover the other half, and decode to m.

This in particular means that in order to prove Theorem 5.5 we will need to use a code that "looks very different from" a random code.

A hybrid between BSC codes and constant maps. It turns out that once we are open to this possibility of dealing with Ham_p rather than $\mathrm{Ckt}_p^{n^c}$, it is not difficult to construct such codes. The intuition is that an encoding map, that maps all messages to the same codeword is obviously evasive, but does not decode from BSC_p. We will use codes that are a hybrid of codes for BSC_p and constant encoding maps. Specifically, we will take the code of Theorem 5.3 and append $\alpha \cdot n$ zeros to each codeword, where α is a carefully chosen constant so that:

- The rate decreases, but still beats the Gilbert-Varshamov bound.
- A random Z is likely to have relative Hamming distance at least $2p$ from all codewords (simply because all codewords have a suffix of αn zeros).

This proof appears in Section 5.5, in which we show that such an α exists, and that this construction decodes from $\mathrm{Perm}_p^{\mathrm{UniPerm}}$.

5.4 An explicit Monte-Carlo construction of evasive codes for $\mathrm{Ckt}_p^{n^c}$ with BSC rate

While the rate $R^*(p)$ that we achieve in Theorem 5.5 beats the Gilbert-Varshamov bound, it does not match the optimal rate $R^{\mathrm{BSC}}(p) = 1 - H(p)$ (achieved for binary symmetric channels).

A natural question is whether there exist codes that decode from BSC_p, are evasive, and have rate $R^{\mathrm{BSC}}(p) = 1 - H(p)$, if we do not insist on evasiveness against Hamming channels and settle for evasiveness for poly-size circuits. The next theorem shows that such codes exist, and furthermore, gives an explicit Monte-Carlo construction. The next theorem is a formal restatement of Theorem 2.1.

Theorem 5.7. *For every constant $0 < p < \frac{1}{4}$, every sufficiently small constant $\epsilon > 0$, and every constant $c > 1$, there exists a constant $D > 1$ such that for $R = 1 - H(p) - \epsilon$, there is a randomized algorithm M running in time N^D, such that for every sufficiently large N, when M is given input N, with probability $1 - \frac{1}{N^c}$ it produces circuits $\mathrm{Enc} : \{0,1\}^{RN} \to \{0,1\}^N$ and $\mathrm{Dec} : \{0,1\}^N \to \{0,1\}^{RN} \cup \{fail\}$ of size N^D, such that:*

- $(\mathrm{Enc}, \mathrm{Dec})$ *decode from* BSC_p *with success* $1 - \frac{1}{N^c}$.
- $(\mathrm{Enc}, \mathrm{Dec})$ *decode from* $\mathrm{Perm}_p^{\mathrm{UniPerm}}$ *with success* $1 - \frac{1}{N^c}$.
- $(\mathrm{Enc}, \mathrm{Dec})$ *are* $\frac{1}{N^c}$*-evasive for* $\mathrm{Ckt}_p^{N^c}$.

The proof of Theorem 5.7 relies on the proof of Theorem 5.9 (that is stated next) and appears in Section 5.8. (An overview of this argument appears in Section 2.4).

A weakness of Theorem 5.7 is that the evasiveness error, decoding error, and Monte-Carlo error are only polynomially small, whereas we can expect (and obtain in the case of Theorem 5.5) errors that are exponentially small.

This does not matter in our intended application of stochastic codes for $\mathrm{Ckt}_p^{n^c}$, as there are currently other bottlenecks, that prevent us from achieving sub-polynomial error in that setting. Nevertheless, if these bottlenecks were to be removed, it would be beneficial to improve this aspect of Theorem 5.7.

Jumping ahead, the code in Theorem 5.7 will be a concatenated code, where the inner code will be a random code on strings of logarithmic length. We will therefore be interested in the evasiveness of random codes against small circuits.

Random codes are evasive for small circuits. We can show that a random code Enc (paired with maximum likelihood decoding) achieves evasiveness for small circuits, and the other desired properties. Recall that in contrast, we have already seen in Proposition 5.6 that random codes are not evasive for Ham_p.

Theorem 5.8 (Existence of evasive codes with optimal rate)**.** *For every constants $0 \leq p < \frac{1}{4}$ and every sufficiently small constant $\epsilon > 0$, there exists a constant $\alpha > 0$ such that for $R = 1 - H(p) - \epsilon$, and for every sufficiently large n, there exist $\mathrm{Enc} : \{0,1\}^{Rn} \to \{0,1\}^n$, and $\mathrm{Dec} : \{0,1\}^n \to \{0,1\}^{Rn} \cup \{fail\}$ such that:*

- $(\mathrm{Enc}, \mathrm{Dec})$ *is $2^{-\alpha \cdot n}$-evasive for* $\mathrm{Ckt}_p^{2^{\alpha \cdot n}}$.
- $(\mathrm{Enc}, \mathrm{Dec})$ *decode from* BSC_p *with success* $1 - 2^{-\alpha \cdot n}$.

We stress that the proof of Theorem 5.8 does not follow by a straightforward union bound, and relies on the methodology presented in Section 4

A random code does not yield a Monte-Carlo construction (as explained in Section 1.2). Nevertheless, in the theorem below, we present a generalized version of Theorem 5.8 that is stated as an exponential time Monte-Carlo construction. This point of view will be helpful, as we plan to use this construction as an inner code (on blocks of logarithmic length) as part of the explicit Monte-Carlo construction of Theorem 5.7.

Theorem 5.9 (Exponential time Monte-Carlo construction)**.** *There exists a universal constant d, such that for every constant $0 \leq p < \frac{1}{4}$ and every sufficiently small constant $\epsilon > 0$, there exists a constant $\alpha > 0$, and a randomized algorithm M running in time $2^{d \cdot n}$ such that for $R = 1 - H(p) - \epsilon$, and for every sufficiently large n, when M is given input n, with probability $1 - 2^{-\alpha \cdot n}$ it produces circuits* $\mathrm{Enc} : \{0,1\}^{Rn} \to \{0,1\}^n$, *and* $\mathrm{Dec} : \{0,1\}^n \to \{0,1\}^{Rn} \cup \{fail\}$, *of size $2^{d \cdot n}$ such that:*

- $(\mathrm{Enc}, \mathrm{Dec})$ *is $2^{-\alpha \cdot n}$-evasive for* $\mathrm{Ckt}_p^{2^{\alpha \cdot n}}$.
- $(\mathrm{Enc}, \mathrm{Dec})$ *decode from* BSC_p *with success* $1 - 2^{-\alpha \cdot n}$.

Furthermore, on input $v \in \{0,1\}^n$, Dec *is a "maximum likelihod decoder up to distance p", namely:*

- *Finds $m \in \{0,1\}^k$, such that $\delta(\mathrm{Enc}(m), v)$ is minimal (breaking ties arbitrarily).*
- *If $\delta(\mathrm{Enc}(m), v) \leq p$, outputs m, and otherwise outputs $fail$.*

A high level overview of this argument appears in Section 2.4. The proofs of Theorem 5.8 and Theorem 5.9 appear in Section 5.6. The proof of Theorem 5.7 will use Theorem 5.9 and is given in Section 5.8.

5.5 Proof of Theorem 5.5

In this section we prove Theorem 5.5. We will use the intuition and simple construction outlined in Section 5.3. We will make use of Theorem 5.3. More specifically, when given parameters $0 < p < \frac{1}{4}$ and $\epsilon > 0$, we set $\delta > 0$ for a sufficiently small constant that we will set later, and set $p' = (1 + \delta) \cdot p$. We apply Theorem

5.3 using p' and a sufficiently small constant $\epsilon' > 0$. Let n' be an integer on which Theorem 5.3 applies, and let $\mathrm{Enc}' : \{0,1\}^k \to \{0,1\}^{n'}$ and $\mathrm{Dec}' : \{0,1\}^{n'} \to \{0,1\}^k$ be the encoding and decoding maps that are guaranteed by the Theorem. In particular, we have that $k = (1 - H(p) - \epsilon') \cdot n'$, and $(\mathrm{Enc}', \mathrm{Dec}')$ decode from $\mathrm{Perm}_p^{\mathrm{UniPerm}_{n'}}$ with success $1 - 2^{-\Omega((n')^{0.1})}$.

Let $\alpha > 0$ be some constant that we will choose later. Let $n'' = \alpha \cdot n'$ and let $n = n' + n'' = (1+\alpha) \cdot n'$. Given a string $v \in \{0,1\}^n$, we will use v' to denote the first n' bits of v and v'' to denote the last n'' bits of v. We define a map $\mathrm{Enc} : \{0,1\}^k \to \{0,1\}^n$ as follows:

$$\mathrm{Enc}(m) = \mathrm{Enc}'(m) \circ 0^{n''}.$$

We define $\mathrm{Dec} : \{0,1\}^n \to \{0,1\}^k \cup \{fail\}$ as follows:

- On input $v \in \{0,1\}^n$, Dec operates as follows:
  - Compute $m = \mathrm{Dec}'(v')$, and $\bar{v} = \mathrm{Enc}(m)$.
  - If $\delta(v, \bar{v}) > p'$ output $fail$.
  - Otherwise, output m.

It is obvious that $(\mathrm{Enc}, \mathrm{Dec})$ are explicit. We will now verify the other requirements.

Claim 5.10. *The construction satisfies:*

- $(\mathrm{Enc}, \mathrm{Dec})$ *is* $2^{-\beta \cdot n}$*-evasive for* Ham_p*, where* $\beta > 0$ *is a constant that depends on p and* ϵ*.*
- *The rate of* Enc *is at least* $R^*(p) - \epsilon = (1 - H(p)) \cdot \frac{1-4p}{1-2p} - \epsilon$*.*

Proof. Let $C : \{0,1\}^n \to \{0,1\}^n$ be a channel in Ham_p. We will show that for every $m \in \{0,1\}^k$,

$$\Pr_{Z \leftarrow U_n}[\delta(\mathrm{Enc}(m), Z) > p + p'] \leq 2^{-\beta \cdot n},$$

for some constant $\beta > 0$ that we choose later. This is sufficient, because $\delta(Z, Z \oplus C(Z)) \leq p$, and if the event above occurs, then by the triangle inequality, $\delta(\mathrm{Enc}(m), Z \oplus C(Z)) > p'$, and $\mathrm{Dec}(Z) = fail$. By construction we have that for every $m \in \{0,1\}^k$ and $z \in \{0,1\}^n$:

$$\delta(\mathrm{Enc}(m), z) = \frac{n' \cdot \delta(\mathrm{Enc}'(m), z') + n'' \cdot wt(z'')}{n}.$$

We will analyze the two summands separately: For the first part, we show that:

$$\begin{aligned}\Pr_{Z \leftarrow U_n}[\exists m \in \{0,1\}^k : \text{ s.t } \delta(\mathrm{Enc}'(m), Z') \leq p'] &\leq \sum_{m \in \{0,1\}^k} \Pr_{Z \leftarrow U_n}[\delta(\mathrm{Enc}'(m), Z') \leq p'] \\ &\leq 2^k \cdot 2^{-(1-H(p')) \cdot n'}. \\ &\leq 2^{(1-H(p')-\epsilon') \cdot n' - (1-H(p') \cdot n'} \\ &\leq 2^{-\epsilon' \cdot n'}.\end{aligned}$$

For the second part, we observe that as the n'' indices of Z'' are chosen uniformly, by a Chernoff bound, for every constant $\eta > 0$,

$$\Pr_{Z \leftarrow U_n}[wt(Z'') \leq \frac{1}{2} - \eta] \leq e^{-\frac{\eta^2 \cdot n''}{3}}.$$

Therefore, by a union bound, with probability at least $1-2^{-\epsilon' \cdot n'}-e^{-\frac{\eta^2 \cdot n''}{3}}$ over choosing $Z \leftarrow U_n$, we have that $wt(Z'') > \frac{1}{2}-\frac{\eta}{2}$ and for every $m \in \{0,1\}^k$, $\delta(\mathrm{Enc}'(m), Z') > p'$. If both events occur then for every $m \in \{0,1\}^k$:

$$\delta(\mathrm{Enc}(m), Z) > \frac{n' \cdot p' + n'' \cdot (\frac{1}{2}-\eta)}{n} = \frac{p' + \alpha \cdot (\frac{1}{2}-\eta)}{1+\alpha}.$$

The last quantity is equal to $p+p' = (2+\delta)\cdot p$ for:

$$\alpha = \frac{2p \cdot (3+\delta) - \eta}{1-2\cdot(2+\delta)\cdot p} > 0,$$

where the last inequality is because we are allowed to assume that $p < \frac{1}{4}$ and $\delta, \eta > 0$ are sufficiently small, so that $2 \cdot (2+\delta) \cdot p < 1$. This gives that the rate of Enc is

$$\frac{k}{n} = \frac{k}{(1+\alpha)\cdot n'} = (1-H(p')-\epsilon') \cdot \frac{1}{1+\alpha} = (1-H(p')-\epsilon') \cdot \frac{1-4\cdot(4+\delta)\cdot p}{1+\eta-2\cdot(1+\delta)\cdot p}.$$

By continuity, we can take $\delta > 0$ and $\eta > 0$ to be sufficiently small constants so that:

$$\frac{1-4\cdot(4+\delta)\cdot p}{1+\eta-2\cdot(1+\delta)\cdot p} \geq \frac{1-4p}{1-2p} - \frac{\epsilon}{2}.$$

Using Equation (3) from Section 3, we get that:

$$H(p') = H((1+\delta)\cdot p) \leq H(p) + \delta.$$

which gives that:

$$1-H(p')-\epsilon' \geq 1-H(p)-\epsilon'-\delta \geq 1-H(p)-\frac{\epsilon}{2},$$

For sufficiently small constants $\delta > 0$ and $\epsilon' > 0$. Putting everything together, we have that for sufficiently small $\epsilon' > 0$ and $\delta > 0$ we get that the rate of Enc is at least

$$(1-H(p)-\frac{\epsilon}{2}) \cdot (\frac{1-4p}{1-2p}-\frac{\epsilon}{2}) \geq (1-H(p)) \cdot \frac{1-4p}{1-2p} - \epsilon,$$

as required. Finally, we note that we have bounded the evasiveness error ρ by:

$$2^{-\epsilon' \cdot n'} + e^{-\frac{\eta^2 \cdot n''}{3}} \leq 2^{-\beta \cdot n},$$

for a sufficiently small constant $\beta > 0$, that we can choose as a function of p and ϵ. □

Claim 5.11. $(\mathrm{Enc}, \mathrm{Dec})$ *decode from* $\mathrm{Perm}_p^{\mathrm{UniPerm}}$ *with success probability* $1-2^{-n^{0.09}}$.

Proof. We need to show that for every $e \in \{0,1\}^n$ with $wt(e) \leq p$, Dec decodes from $\mathrm{UniPerm}(e)$. The monotonicity property of $(\mathrm{Enc}', \mathrm{Dec}')$ immediately gives monotonicity also for $(\mathrm{Enc}, \mathrm{Dec})$, which implies that it is sufficient to restrict our attention to $e \in \{0,1\}^n$ with $wt(e) = p$.[23]

[23] Here we are implicitly assuming that pn is an integer, which doesn't necessarily applies. However, if pn is not an integer, we will replace p by the smallest $p^* \leq p$ such that $p^* \cdot n$ is an integer, and for large enough n, this difference is negligible.

We will consider the experiment $\pi \leftarrow \mathrm{UniPerm}_n$, and note that in this experiment, π^{-1} is also a uniform permutation. For every $i \in [n']$, let $X_i = \pi(e)_i$, and let $X = \sum_{i \in [n']} X_i$ be the weight of $\pi(e)'$. We claim that by Lemma 3.9, taking $t = \frac{\delta \cdot p \cdot n'}{2}$.

$$\Pr_{\pi \leftarrow \mathrm{UniPerm}}[X > (1+\delta)p \cdot n'] \leq e^{-\frac{\delta \cdot t}{2}} \leq e^{-\frac{\delta^2 \cdot p \cdot n'}{2}}.$$

Let $\ell = (1+\delta)p \cdot n' = p' \cdot n'$, so that $p' = \ell / n'$, and let $E = \{X \leq \ell\}$.

Consider the distribution π conditioned on the event E, and let us denote this distribution by σ. We note that by considering different choices for the n'' indices j such that $n' < \sigma(j) \leq n$, the distribution σ can be expressed as a convex combination of distributions τ in which:

- For every $n' < i \leq n$, $\tau^{-1}(i)$ is fixed.
- This means that there are fixed indices $j_1 < \ldots < j_{n'}$ such that for every $g \in [n']$, $\tau(j_g) \in [n']$.
- The n' bit long string $e_{j_1}, \ldots, e_{j_{n'}}$ has Hamming weight at most ℓ, meaning that it has relative Hamming weight at most $p' = \ell/n'$.
- The random variables $\tau(j_1), \ldots, \tau(j_{n'})$ are distributed like n' random values in $[n']$ that are chosen without replacement.

Each one of the elements τ in the convex combination can be viewed as a permutation on $[n']$, by identifying $\{j_1, \ldots, j_{n'}\}$ with $[n']$. Furthermore, this permutation is distributed like $\mathrm{UniPerm}_{n'}$ (that is, like a uniform permutation on $[n']$). It follows that $\tau(e)'$ is a distribution in $\mathrm{Perm}_{p'}^{\mathrm{UniPerm}_{n'}}$. For every such τ, we have that Dec' decodes from $\tau(e)'$ with success $1 - 2^{-\Omega((n')^{0.1})}$. We conclude that:

$$\Pr_{\pi \leftarrow \mathrm{UniPerm}_n}[\mathrm{Dec}(\mathrm{Enc}(m) \oplus \pi(e)) = m \mid E] \geq 1 - 2^{-\Omega((n')^{0.1})}.$$

Putting things together we conclude that:

$$\Pr_{\pi \leftarrow \mathrm{UniPerm}_n}[\mathrm{Dec}(\mathrm{Enc}(m) \oplus \pi(e)) = m] \geq 1 - 2^{-\Omega((n')^{0.1})} - e^{-\frac{\delta^2 \cdot p \cdot n'}{2}} \geq 1 - 2^{-n^{0.09}},$$

for sufficiently large n. □

Finally, it is not difficult to verify that for every $0 < p < \frac{1}{4}$, $R^*(p) > 1 - H(2p)$. This concludes the proof of Theorem 5.5.

5.6 Random codes with rate $1 - H(p)$ are evasive for $\mathrm{Ckt}_p^{2^{O(n)}}$

In this section we prove Theorem 5.8 and Theorem 5.9. We first show that a random code with rate approaching $1 - H(p)$ is w.h.p. evasive against any class $\mathcal{C} \subseteq Ham_p$ of $2^{2^{O(n)}}$ channels, and in particular for $\mathrm{Ckt}_p^{n^c}$ or even circuits of almost exponential size. This is stated in the next lemma.

Lemma 5.12 (Random codes are evasive for small classes)**.** *For every constants $0 \leq p < \frac{1}{4}$ and $\epsilon > 0$, there exists a constant $\alpha > 0$, such that for $R = 1 - H(p) - \epsilon$, and for every sufficiently large n, the following holds: Let $\mathcal{C} \subseteq \mathrm{Ham}_p$ be a class of functions that contains at most $2^{2^{\alpha \cdot n}}$ functions, and let* $\mathrm{Enc} : \{0,1\}^{Rn} \to \{0,1\}^n$ *be chosen uniformly from all such functions. Let* $\mathrm{Dec} : \{0,1\}^n \to \{0,1\}^{Rn} \cup \{fail\}$*, be the map that on input $v \in \{0,1\}^n$:*

- *Finds $m \in \{0,1\}^k$, such that $\delta(\mathrm{Enc}(m), v)$ is minimal (breaking ties arbitrarily).*
- *If $\delta(\mathrm{Enc}(m), v) \leq p$, outputs m, and otherwise outputs $fail$.*

With probability $1 - 2^{-2^{\alpha \cdot n}}$, $(\mathrm{Enc}, \mathrm{Dec})$ is $2^{-\alpha \cdot n}$-evasive for $\mathcal{C}$.

A high level overview of this argument is given in Section 2.4. The formal proof is given in Section 5.7. Before proving Lemma 5.12, we observe that Theorem 5.8 follows from Lemma 5.12.

Proof of Theorem 5.8. We will apply Lemma 5.12 using $p' = p + \epsilon'$ for a constant $\epsilon' > 0$ that is sufficiently small as a function of p and ϵ, so that $H(p') \leq H(p) + \epsilon$. This gives that when applying Lemma 5.12, we obtain a code with rate $1 - H(p) - 2\epsilon$, which is just as good for our purposes.

Shannon showed that if one chooses a random code Enc (as is done in Lemma 5.12) then, while the obtained code is not necessarily likely to decode from $\mathrm{BSC}_{p'}$, it does hold that except for probability $2^{-\Omega_{\epsilon,p}(n)}$, it is possible to remove half of the codewords, and obtain a code that does decode from BSC_p using maximum likelihood decoding algorithm Dec. Therefore, following this removal we obtain a code that is decodable from $\mathrm{BSC}_{p'}$ (and in particular from BSC_p) with success probability $1 - 2^{-\Omega_{\epsilon,p}(n)}$.

The reason that we took some slack, choosing $p' = p + \epsilon'$ is to guarantee that with probability $1 - 2^{-\Omega_{\epsilon,p}(n)}$ over the choice of $e \leftarrow \mathrm{BSC}_p$, we have that $wt(e) \leq p'$. This guarantees that it is unlikely that Dec will reject a codeword that was corrupted by BSC_p.

Removing codewords from a code cannot harm the evasiveness property, therefore, by a union bound, it is possible to obtain a code that is both decodable from BSC_p and $2^{-\alpha \cdot n}$-evasive against $\mathcal{C}$. This proves Theorem 5.8.

Extending the proof to prove Theorem 5.9. For Theorem 5.9 we observe that given a candidate encoding map $\mathrm{Enc} : \{0,1\}^k \to \{0,1\}^n$, and a message $m \in \{0,1\}^k$, in time 2^n, one can go over all choices of noise from BSC_p and compute the probability that Dec decodes correctly. By going over all 2^k choices of messages, we can check which messages we want to remove from the code Enc in time $2^{O(n)}$. This means that in time $2^{O(n)}$ we can trim the encoding map Enc to one that is decodable from BSC_p with success $1 - 2^{-\Omega_{\epsilon,p}(n)}$, giving a Monte-Carlo construction that runs in exponential time, uses $2^{O(n)}$ random bits, and has Monte-Carlo error $2^{-\Omega_{\epsilon,p}(n)}$ as required.

5.7 Proof of Lemma 5.12

We will use the methodology explained in Section 4, and start with some notation.

5.7.1 Preperations for the methodology of Section 4

The setup: Let $0 \geq p < \frac{1}{4}$ be a constant, let $\epsilon > 0$ be a sufficiently small constant, and let n be an integer (that we are allowed to assume that is sufficiently large). Let $R = 1 - H(p) - \epsilon$, $k = Rn$ and $K = 2^k$. We will use the following notation for the probability space of choosing Enc.

Experiment expr**: A random code.**

- Let $\mathcal{X} = (\{0,1\}^n)^K$.
- Let $X \leftarrow \mathcal{X}$ be a uniform element $X = (X_1, \ldots, X_K)$ from $\mathcal{X}$.
- we identify $j \in [K]$ with elements $j \in \{0,1\}^k$.
- Let $\mathrm{Enc}(j) = X_j$.

5.7.2 The game of a channel C

Following the recipe in Section 4 we now define a game for a channel C.

Definition 5.13 (Circuit C wins)**.** *Given a $C \in \mathcal{C}$, $x \in \mathcal{X}$, and $z \in \{0,1\}^n$, we say that C* **wins on** *x, z if*

$$\exists j \in [K] : \delta(z \oplus C(z), x_j) \leq p.$$

To make the notation easier, we define $W^C(x,z) = 1$ iff C wins on x, z.

Note when considering the code Enc defined by x, if $\mathrm{Dec}(z \oplus C(z)) \neq fail$ then C wins on x, z. This means that whenever C is able to make Dec decode, C wins.

Let $\alpha > 0$ be a sufficiently small constants that we will choose later. We will shoot for "evasiveness error" $\rho = 2^{-\alpha \cdot n}$. Definition 5.13 is made so that the task of proving Lemma 5.12 reduces to the task of proving that:

$$\Pr_{X \leftarrow \mathcal{X}}\left[\exists C \in \mathcal{C} \text{ s.t. } \Pr_{Z \leftarrow \{0,1\}^n}\left[W^C(X,Z) = 1\right] > 2^{-\alpha \cdot n}\right] < 2^{-2^{\alpha \cdot n}}.$$

This will follow by a union bound if we can prove that for every for every one of the $2^{2^{\alpha \cdot n}}$ choices of $C \in \mathcal{C}$:

$$\Pr_{X \leftarrow \mathcal{X}}\left[\Pr_{Z \leftarrow \{0,1\}^n}\left[W^C(X,Z) = 1\right] > 2^{-\alpha \cdot n}\right] < \frac{2^{-2^{\alpha \cdot n}}}{2^{2^{\alpha \cdot n}}}.$$

Let $\mathcal{Z} = \{0,1\}^n$ and let:

$$W_{\mathrm{avg}}(x) = \frac{1}{|\mathcal{Z}|} \cdot \sum_{z \in \mathcal{Z}} W^C(x,z),$$

as is done in Lemma 4.1. We would like to use Lemma 4.1. For that purpose we choose $t = 2^{2\alpha \cdot n}$, so that it is sufficient to prove that:

$$\Pr_{X \leftarrow \mathcal{X}}[W_{\mathrm{avg}}(X) > 2^{-\alpha \cdot n}] \leq e^{-t}. \tag{4}$$

We will use Lemma 4.1 choosing $\delta = 2$, and $\mu = \frac{2^{-\alpha \cdot n}}{3}$ (so that $(1+\delta) \cdot \mu = 2^{-\alpha \cdot n}$), We need to verify that we meet the condition on t in Lemma 4.1, and indeed:

$$t = 2^{2 \cdot \alpha \cdot n} \leq \frac{\mu \cdot \delta \cdot 2^n}{2},$$

by taking the constant $\alpha > 0$ to be sufficiently small. Using Lemma 4.1, it follows that in order to prove that (4) holds, it is sufficient to prove the following claim:

Claim 5.14. *For every integer $0 \leq q < t$,*

$$\Pr_{\substack{X; Z \leftarrow \mathcal{Z} \\ Z_1, \ldots, Z_q \overset{\mathrm{wor}}{\leftarrow} \mathcal{Z}}}[W(X,Z) = 1 \mid W(X,Z_1) = \ldots = W(X,Z_q) = 1] \leq \mu,$$

We therefore focus our attention on proving Claim 5.14, and the lemma will follow once we prove Claim 5.14. We prove Claim 5.14 in Section 5.7.3.

5.7.3 Using the recipe of Section 4: proof of Claim 5.14

Fix some $0 \leq q < t$. We will now show that we can express the event $\{W(X, Z_1) = \ldots = W(X, Z_q) = 1\}$ as a disjoint union of "simple events".

More specifically, the event $\{W(X, Z_1) = \ldots = W(X, Z_q) = 1\}$ can be viewed as a subset $T \subseteq \mathcal{X} \times \mathcal{Z}^q$ by setting:

$$T = \{(x; z_1, \ldots, z_q) : W(x, z_1) = \ldots = W(x, z_q) = 1\}\,.$$

We will now show that T can be expressed as a disjoint union of "simple events". Loosely speaking, a simple event E is a subset $E \subseteq T$ in which $z_1, \ldots, z_q$, as well as several x_j's are fixed, in a very specific way.

Definition 5.15 (Simple event)**.** *For every choice of:*

- $z_1, \ldots, z_q \in \mathcal{Z} = \{0,1\}^n$.
- *A function* $h : [q] \to \{0,1\}^n$
- $j_1, \ldots, j_q \in [K]$.

We define a set $E \subseteq \mathcal{X} \times \mathcal{Z}^q$ *(called the* **simple event** *induced by* $z_1, \ldots, z_q$*,* h *and* $j_1, \ldots, j_q$*). The event* E *is defined by:*

$$E = D \times \{(z_1, \ldots, z_q)\}\,,$$

where D *is the set of all* $x \in \mathcal{X}$ *such that for every* $g \in [q]$*:*

- $x_{j_g} = h(g)$.
- $\delta(x_{j_g}, z_g \oplus C(z_g)) \leq p$.
- *For every* $u < j_g$*,* $\delta(x_u, z_g \oplus C(z_g)) > p$.

We will say that a simple event E *is nontrivial if*

$$\Pr_{X \leftarrow \mathcal{X}, Z_1, \ldots, Z_q \overset{\text{wor}}{\leftarrow} \mathcal{Z}}[(X, Z_1, \ldots, Z_q) \in E] > 0.$$

Experiment expr_2**: conditioning on a simple event.** We will be interested in the distribution obtained by conditioning the distribution $(X \leftarrow \mathcal{X}, Z_1, \ldots, Z_q \overset{\text{wor}}{\leftarrow} \mathcal{Z})$ on $\{(X, Z_1, \ldots, Z_q) \in E\}$ for a nontrivial simple event E induced by $z_1, \ldots, z_q$, h and $j_1, \ldots, j_q$. Let us denote this experiment by $\mathrm{expr}_2(z_1, \ldots, z_q; h; j_1, \ldots, j_q)$. We observe that for $(X, Z_1, \ldots, Z_q) \leftarrow \mathrm{expr}_2(z_1, \ldots, z_q; h; j_1, \ldots, j_q)$ we have that for every $g \in [q]$:

- Z_g is fixed to z_g.
- X_{j_g} is fixed to $x_{j_q} = h(g)$, such that $\delta(x_{j_g}, z_g \oplus C(z_g)) \leq p$ (which means that $W(x, z_g) = 1$ and C wins on x, z_g).
- j_g is the smallest index u for which $\delta(X_u, z_g \oplus C(z_g)) \leq p$, or in other words, that for every $u < j_g$, $\Pr[\delta(X_u, z_g \oplus C(z_g)) > p] = 1$.

This means that $X \leftarrow \mathrm{expr}_2(z_1, \ldots, z_q; h; j_1, \ldots, j_q)$ is distributed as follows:

- For $u \in \{j_1, \ldots, j_q\}$, X_u is fixed to a value $h(g)$ in the Hamming ball of radius p around $z_g \oplus C(z_g)$.

- For $u \notin \{j_1, \ldots, j_q\}$, X_u is uniformly distributed over a subset of $\{0,1\}^n$, which is of size at least $2^n - q \cdot 2^{H(p) \cdot n}$. This is because X_u is uniform given the restriction that it does not belong to a Hamming ball of relative radius p around some $z_g \oplus C(z_g)$ where the corresponding index $j_g > u$.

  There are at most q choices for g, and each one rules out a Hamming ball of relative radius p, which is a set of size at most $2^{H(p) \cdot n}$.
- $(X_j)_{j \in [K] \setminus \{j_1, \ldots, j_g\}}$ are independent. (We will actually not use this property in the proof below, but we list it anyway to make things more clear.)

We can also conclude that:

- Every simple event E, satisfies $E \subseteq T = \{W(X, Z_1) = \ldots = W(X, Z_q) = 1\}$. This is because by definition on every $(x, z_1, \ldots, z_q) \in E$, $W(x, z_1) = \ldots = W(x, z_q) = 1$.
- Every two simple events are either equal or disjoint. This is because in order for two simple events to have a non-empty intersection, the two events must agree on $z_1, \ldots, z_q$. They also must agree on $j_1, \ldots, j_q$, because if they don't agree on some j^g, then one of the two simple events will use a larger j_g, enforcing that for all $u < j_g$, $\delta(x_u, z_g \oplus C(z_j)) > p$, and this cannot occur on the other simple event. Once they agree on $j_1, \ldots, j_q$, they must also (by definition) agree on h.

The discussion above implies that:

- The event $T = \{W(X, Z_1) = \ldots = W(X, Z_q) = 1\}$ is a disjoint union of nontrivial simple events.
- In order to show that:

$$\Pr_{\substack{X \leftarrow \mathcal{X}, Z \leftarrow \mathcal{Z} \\ Z_1, \ldots, Z_q \overset{\text{wor}}{\leftarrow} \mathcal{Z}}} [W(X, Z) = 1 \mid W(X, Z_1) = \ldots = W(X, Z_q) = 1] \leq \mu,$$

  it is sufficient to show that for every choice of nontrivial simple event E that is induced by some $z_1, \ldots, z_q$, h and $j_1, \ldots, j_q$:

$$\Pr_{\substack{X \leftarrow \mathcal{X}, Z \leftarrow \mathcal{Z} \\ Z_1, \ldots, Z_q \overset{\text{wor}}{\leftarrow} \mathcal{Z}}} [W(X, Z) = 1 \mid (X, Z_1, \ldots, Z_q) \in E] \leq \mu,$$

- Simplifying the expression above, it is sufficient to show the following for every choice of nontrivial simple event E that is induced by some $z_1, \ldots, z_q$, h and $j_1, \ldots, j_q$:

$$\Pr_{\substack{X \leftarrow \mathrm{expr}_2(z_1, \ldots, z_q; h; j_1, \ldots, j_q) \\ Z \leftarrow \mathcal{Z}}} [W(X, Z) = 1] \leq \mu. \tag{5}$$

In the remainder of the proof, we will prove that (5) holds. We will fix some nontrivial simple event E that is induced by some $z_1, \ldots, z_q$, h and $j_1, \ldots, j_q$, and to avoid clutter, we will define:

$$\mathrm{expr}_2 = \mathrm{expr}_2(z_1, \ldots, z_q; h; j_1, \ldots, j_q).$$

We are interested in bounding:

$$\Pr_{\substack{X \leftarrow \mathrm{expr}_2 \\ Z \leftarrow \mathcal{Z}}} [W(X, Z) = 1] = \Pr_{\substack{X \leftarrow \mathrm{expr}_2 \\ Z \leftarrow \mathcal{Z}}} [\exists j : \delta(X_j, Z \oplus C(Z)) \leq p].$$

The two next claims bound this probability for a specific j, depending on whether $j \in \{j_1, \ldots, j_q\}$ or $j \notin \{j_1, \ldots, j_q\}$.

Claim 5.16. *For every $g \in [q]$,*

$$\Pr_{\substack{X\leftarrow \mathrm{expr}_2\\ Z\leftarrow \mathcal{Z}}}[\delta(X_{j_g}, Z \oplus C(Z)) \le p] \le 2^{-(1-H(2p))\cdot n}$$

Proof. Recall that for $X \leftarrow \mathrm{expr}_2$, X_{j_g} is fixed to $h(g)$. It follows that:

$$\begin{aligned}\Pr_{\substack{X\leftarrow \mathrm{expr}_2\\ Z\leftarrow \mathcal{Z}}}[\delta(X_{j_g}, Z \oplus C(Z)) \le p] &= \Pr_{Z\leftarrow \mathcal{Z}}[\delta(h(g), Z \oplus C(Z)) \le p] \\ &\le \Pr_{Z\leftarrow \mathcal{Z}}[\delta(h(g), Z) \le 2 \cdot p] \\ &\le 2^{-(1-H(2p))\cdot n},\end{aligned}$$

where the second line follows because using the fact that $C \in \mathcal{C} \subseteq \mathrm{Ham}_p$, we have that $\delta(Z, Z \oplus C(Z)) \le p$ and therefore, by the triangle inequality we have that if $\delta(h(g), Z \oplus C(Z)) \le p$ then $\delta(h(g), Z) \le 2p$. □

Claim 5.17. *For every $j \in [K] \setminus \{j_1, \ldots, j_q\}$,*

$$\Pr_{\substack{X\leftarrow \mathrm{expr}_2\\ Z\leftarrow \mathcal{Z}}}[\delta(X_{j_g}, Z \oplus C(Z)) \le p] \le 2 \cdot 2^{-(1-H(p))\cdot n}$$

Proof. Recall that for $X \leftarrow \mathrm{expr}_2$, and $j \in [K] \setminus \{j_1, \ldots, j_q\}$, we have that X_j is uniform over a set of size at least $2^n - q \cdot 2^{H(p)\cdot n}$. It follows that:

$$\begin{aligned}\Pr_{\substack{X\leftarrow \mathrm{expr}_2\\ Z\leftarrow \mathcal{Z}}}[\delta(X_j, Z \oplus C(Z)) \le p] &\le \max_{z\in\mathcal{Z}}\left(\Pr_{X\leftarrow \mathrm{expr}_2}[\delta(X_j, z \oplus C(z)) \le p]\right) \\ &\le \frac{2^{H(p)\cdot n}}{2^n - q \cdot 2^{H(p)\cdot n}} \\ &\le 2 \cdot 2^{-(1-H(p))\cdot n},\end{aligned}$$

where the first line follows because X, Z are independent, and the last line follows because $q \le t = 2^{2\cdot\alpha\cdot n}$, $p < \frac{1}{4}$, and we can take $\alpha > 0$ to be sufficiently small, so that $t \cdot 2^{H(p)\cdot n} \le 2^{n-1}$. □

We are finally ready to prove Claim 5.14.

$$\begin{aligned}\Pr_{\substack{X\leftarrow \mathrm{expr}_2\\ Z\leftarrow \mathcal{Z}}}[W(X,Z) = 1] &= \Pr_{\substack{X\leftarrow \mathrm{expr}_2\\ Z\leftarrow \mathcal{Z}}}[\exists j : \delta(X_j, Z \oplus C(Z)) \le p] \\ &\le \sum_{j\in[K]} \Pr_{\substack{X\leftarrow \mathrm{expr}_2\\ Z\leftarrow \mathcal{Z}}}[\delta(X_j, Z \oplus C(Z)) \le p] \\ &\le \sum_{j\in\{j_1,\ldots,j_q\}} \Pr_{\substack{X\leftarrow \mathrm{expr}_2\\ Z\leftarrow \mathcal{Z}}}[\delta(X_j, Z \oplus C(Z)) \le p] \\ &\qquad + \sum_{j\in[K]\setminus\{j_1,\ldots,j_q\}} \Pr_{\substack{X\leftarrow \mathrm{expr}_2\\ Z\leftarrow \mathcal{Z}}}[\delta(X_j, Z \oplus C(Z)) \le p] \\ &\le q \cdot 2^{-(1-H(2p))\cdot n} + K \cdot 2 \cdot 2^{-(1-H(p))\cdot n} \\ &\le q \cdot 2^{-O(\epsilon^2)\cdot n} + 2 \cdot 2^{-\epsilon\cdot n} \\ &\le 2^{2\cdot\alpha\cdot n - O(\epsilon^2\cdot n)} + 2^{-\epsilon\cdot n+1} \\ &\le \mu,\end{aligned}$$

where the fourth line follows using Claim 5.16 and Claim 5.17, the fifth line follows because we can assume that ϵ is sufficiently small so that $p+\epsilon < \frac{1}{4}$, which gives that $2p < \frac{1}{2} - 2\epsilon$, and $H(\frac{1}{2} - 2\epsilon) \le 1 - O(\epsilon^2)$, the fifth line also uses the choice of $K = 2^{(1-H(p)-\epsilon)\cdot n}$, the penultimate line follows because $q < t = 2^{2\cdot\alpha\cdot n}$, and the final line follows because we can take $\alpha > 0$ to be sufficiently small as a function of ϵ, and we have chosen $\mu = \frac{2^{-\alpha\cdot n}}{3}$.

5.8 Proof of Theorem 5.7

Intuition for the proof. In this section we prove we prove Theorem 5.7. As in many constructions of codes for BSC_p, we will use code concatenation. As an inner code, we will use Theorem 5.9 over blocks of length $O(\log n)$ (where its encoding and decoding run in polynomial time). As an outer code we use the identity map (as in our setting, we will not need to recover from errors in the outer code).

The main issue is to argue that the code is evasive. This is because the channel $C \in \mathrm{Ckt}_p^{n^c}$, does not have to "evenly allocate the number of errors" on the different inner blocks. Moreover, it can choose the error pattern on different blocks in a correlated way, that depends on the entire input string. In the proof that is given below, we will need to address this problem. This is treated formally in the proof of Claim 5.19 below.

Let $c_s = c+2$ and $p_{\mathrm{BSC}} = p \cdot (1 + \frac{\epsilon}{10})$. We will apply Theorem 5.9 that gives a Monte-Carlo construction using the constants p_{BSC} and $\frac{\epsilon}{10}$. Theorem 5.9 guarantees the existence of a constant $\alpha > 0$. Let $e > 1$ be a sufficiently large constant. At this point, we will require that $e \ge c_s/\alpha$, and we will make additional requirements that e is sufficiently large, later. Let N be sufficiently large for Theorem 5.9, and let $n = e \cdot \log N$, so that

$$2^{\alpha\cdot n} \ge N^{c_s}.$$

The algorithm M that we construct will run the randomized algorithm from Theorem 5.9 using n. By Theorem 5.9, this takes time $2^{d\cdot n}$, and we take the constant D to be sufficiently large so that $2^{d \log n} \le N^D$, so that the randomized algorithm runs in time N^D, and produces circuits $\mathrm{Enc}_{\mathrm{in}} : \{0,1\}^{R_{\mathrm{BSC}}\cdot n} \to \{0,1\}^n$ and circuits $\mathrm{Dec}_{\mathrm{in}} : \{0,1\}^n \to \{0,1\}^{R_{\mathrm{BSC}}\cdot n}$ of size N^d, for

$$R_{\mathrm{BSC}} = 1 - H(p_{\mathrm{BSC}}) - \frac{\epsilon}{10} \ge R = 1 - H(p) - \epsilon,$$

where the inequality follows using (3) from Section 3. We now define the encoding and decoding maps $\mathrm{Enc} : \{0,1\}^{RN} \to \{0,1\}^N$ and $\mathrm{Dec} : \{0,1\}^N \to \{0,1\}^{Rn} \cup \{fail\}$ as follows: Let $\ell = N/n$. We divide strings m of length $k = RN$ into ℓ blocks of length Rn, and use m_i to be the i'th block. We also divide strings v of length N into ℓ blocks of length n, and use v_i to be the i'th block. We define:

- $\mathrm{Enc}(m) = \mathrm{Enc}_{\mathrm{in}}(m_1), \ldots, \mathrm{Enc}_{\mathrm{in}}(m_\ell)$.
- $\mathrm{Dec}(v)$ applies $\mathrm{Dec}_{\mathrm{in}}(v_i)$ for every $i \in [\ell]$. If all of them do not fail, then $\mathrm{Dec}(v) = \mathrm{Dec}_{\mathrm{in}}(v_1), \ldots, \mathrm{Dec}_{\mathrm{in}}(v_k)$, and otherwise, $\mathrm{Dec}(v) = fail$.

We can choose D to be sufficiently large so that $\mathrm{Enc}, \mathrm{Dec}$ are of size N^D.

Claim 5.18.

- $(\mathrm{Enc}, \mathrm{Dec})$ *are decodable from* BSC_p *with success* $1 - \frac{1}{N^c}$.
- $(\mathrm{Enc}, \mathrm{Dec})$ *are decodable from* $\mathrm{Perm}_p^{\mathrm{UniPerm}}$ *with success* $1 - \frac{1}{N^c}$.

Proof. By the properties of $\mathrm{Dec}_{\mathrm{in}}$ guaranteed in Theorem 5.9, we have that for every $m \in \{0,1\}^k$, and $e \in \{0,1\}^n$, if for every $i \in [n]$, $wt(e_i) \leq p_{\mathrm{BSC}}$, then $\mathrm{Dec}_{\mathrm{in}}(\mathrm{Enc}(m_i) \oplus e_i) = m_i$, which gives that $\mathrm{Dec}(\mathrm{Enc}(m \oplus e)) = m$. Thus, it is sufficient to prove that for $E \leftarrow \mathrm{BSC}_p$ or $E \leftarrow \mathrm{Perm}_p^{\mathtt{UniPerm}}$, for every $i \in [n]$,

$$\Pr[wt(E_i) > p_{\mathrm{BSC}}] \leq \frac{1}{N^{c+1}}.$$

so that:

$$\Pr[\exists i : wt(E_i) > p_{\mathrm{BSC}}] \leq \sum_{i \in [n]} \Pr[wt(E_i) > p_{\mathrm{BSC}}] \leq \frac{N}{N^{c+1}} = \frac{1}{N^c}.$$

In both cases (namely, both when $E \leftarrow \mathrm{BSC}_p$ and $E \leftarrow \mathrm{Perm}_p^{\mathtt{UniPerm}}$), we have that:

$$\Pr[wt(E_i) > p \cdot (1 + \frac{\epsilon}{10})] \leq e^{-\Omega(\epsilon^2 \cdot p \cdot n)} \leq \frac{1}{N^{c+1}},$$

if we take the constant e to be sufficiently large. For BSC_p this follows by a Chernoff bound. For $\mathrm{Perm}_p^{\mathtt{UniPerm}}$ this follows by Lemma 3.9, taking $\delta = \frac{\epsilon}{10}$, $\mu = p$, $t = \delta \cdot \mu \cdot n/2$ and considering the random variables $X_1, \ldots, X_n$, where X_j is the j'th bit of E_i. □

It remains to show that $(\mathrm{Enc}, \mathrm{Dec})$ are $\frac{1}{N^c}$-evasive for $\mathrm{Ckt}_p^{N^c}$. Let $C : \{0,1\}^N \to \{0,1\}^N$ be a circuit of size N^c. We will consider the experiment where $Z \leftarrow U_N$ is chosen uniformly. For every $i \in [\ell]$, and let

$$W_i = \{\mathrm{Dec}_{\mathrm{in}}(Z_i \oplus C(Z)_i) \neq fail\}.$$

In words, W_i is the event that $\mathrm{Dec}_{\mathrm{in}}$ does not fail on the i'th block. Let $W = \cap_{i \in [\ell]} W_i$ be the event that Dec does not fail. Let $B_i = \{wt(C(Z))_i > p\}$, and S_i be the complement, namely $S_i = \{wt(C(Z))_i \leq p\}$.

Assume for the purpose of contradiction that $\Pr[W] > N^{-c}$. We will iteratively apply the following claim, starting with $i = 1$, and $\gamma = \frac{1}{N^c}$:

Claim 5.19. *For every $z_1, \ldots, z_{i-1} \in \{0,1\}^n$, if*

$$\Pr[W \cap B_1 \cap \ldots \cap B_{i-1} \mid Z_1 = z_1, \ldots, Z_{i-1} = z_{i-1}] > \gamma,$$

then there exists $z_i \in \{0,1\}^n$, such that:

$$\Pr[W \cap B_1 \cap \ldots \cap B_i \mid Z_1 = z_1, \ldots, Z_i = z_i] > \gamma - N^{-c_s}.$$

Proof.

$$\begin{aligned}\gamma &< \Pr[W \cap B_1 \cap \ldots \cap B_{i-1} \mid Z_1 = z_1, \ldots, Z_{i-1} = z_{i-1}] \\ &= \Pr[W \cap B_1 \cap \ldots \cap B_{i-1} \cap B_i \mid Z_1 = z_1, \ldots, Z_{i-1} = z_{i-1}] \\ &+ \Pr[W \cap B_1 \cap \ldots \cap B_{i-1} \cap S_i \mid Z_1 = z_1, \ldots, Z_{i-1} = z_{i-1}]\end{aligned}$$

We have that:

$$\begin{aligned}\Pr[W \cap B_1 \cap \ldots \cap B_{i-1} \cap S_i \mid Z_1 = z_1, \ldots, Z_{i-1} = z_{i-1}] &\leq \Pr[W_i \cap S_i \mid Z_1 = z_1, \ldots, Z_{i-1} = z_{i-1}] \\ &\leq N^{-c_s},\end{aligned}$$

where the last inequality follows because Z_i is uniform over $\{0,1\}^n$, and we can consider a circuit $C'(z_i)$ which is hardwired with $z_1, \dots, z_{i-1}$, and also with the best choices for $z_{i+1}, \dots, z_\ell$, and $C'(z_i) = C(z_1, \dots, z_\ell)_i$. As such a circuit cannot break the evasiveness of $\mathrm{Enc}_{\mathrm{in}}, \mathrm{Dec}_{\mathrm{in}}$ when modifying only a p fraction of the output bits (which happens in S_i), we have that:

$$\Pr[W_i \cap S_i \mid Z_1 = z_1, \dots, Z_{i-1} = z_{i-1}] \leq \Pr_{Z_i \leftarrow U_n}[\mathrm{Dec}(Z_i \oplus C'(Z_i)) \neq \mathit{fail} \cap \mathit{wt}(C'(Z_i)) \leq p] \leq N^{-c_s}.$$

Combining the inequalities, we get that:

$$P := \Pr[W \cap B_1 \cap \ldots \cap B_i \mid Z_1 = z_1, \dots, Z_{i-1} = z_{i-1}] > \gamma - N^{-c_s}.$$

We can conclude that:

$$P = \sum_{z_i \in \{0,1\}^n} \Pr[W \cap B_1 \cap \ldots \cap B_i \mid Z_1 = z_1, \dots, Z_i = z_i] \cdot \Pr[Z_i = z_i],$$

and therefore, by averaging, there exists $z_i \in \{0,1\}^n$, such that:

$$\Pr[W \cap B_1 \cap \ldots \cap B_i \mid Z_1 = z_1, \dots, Z_i = z_i] \geq P > \gamma - N^{-c_s}.$$

□

After applying the claim ℓ times, we conclude that:

$$\Pr[W \cap B_1 \cap \ldots \cap B_\ell \mid Z_1 = z_1, \dots, Z_\ell = z_\ell] \geq N^{-c} - \ell \cdot N^{-c_s} > N^{-c} - N^{-(c_s - 1)} > 0.$$

This is a contradiction, because $C \in \mathrm{Ham}_p$ outputs a string of hamming weight at most p, and therefore $\Pr[B_1 \cap \ldots \cap B_\ell] = 0$, because in the latter event

$$\mathit{wt}(C(Z)) = \frac{1}{\ell} \cdot \sum_{i \in [\ell]} \mathit{wt}(C(Z)_i) > p.$$

6 Small set non-malleable codes

In this section we give the definition of SS-non-malleable codes (which we discussed in Section 2.2). In Section 6.1 we define SS-non-malleable codes and compare them to the standard definition of non-malleable codes given by Dziembowski, Pietrzak and Wichs [DPW18]. In Section 6.2 we state the main theorem of this section (Theorem 6.5 which is the formal restatement of Theorem 2.2) that gives an explicit Monte-Carlo construction of SS-non-malleable codes on logarithmic length strings, for circuits of polynomial size, with certain additional decoding and pseudorandomness properties. In remainder of the section is devoted to the proof of Theorem 6.5. The proof is quite involved, and uses amongst other ideas, the methodology developed in Section 4.

6.1 Definition of small set non-malleable codes

In this paper we introduce a new notion of stochastic codes. This notion is a variant in the framework of non-malleable codes that was defined by the seminal work of Dziembowski, Pietrzak and Wichs [DPW18]. Non-malleable codes consider a scenario where the channel C is not necessarily in Ham_p, and there is no bound on the number of errors that it can induce. The goal is to guarantee that no channel C from some specific channel class $\mathcal{C}$ can corrupt the codeword, so that the decoder outputs a message that is related with the original message. This is defined using notions of "simulators" and "indistinguishability" that are inspired from cryptographic definitions. The reader is referred to [DPW18] for a precise definition and a discussion.

In this paper we introduce a different definition which we call "small set non-malleable codes" that is tailored for our application, and may be of independent interest.

We will use these SS-non-malleable codes to encode seeds S of our final stochastic code, and therefore, it will be more convenient in this section to denote the message of the stochastic code by s, and the randomness by r. We start with a simple case which we call "weakly SS-non-malleability" (that introduces some of the modifications and is strictly weaker than the standard definition of non-malleable codes). The full-fledged definition adds additional components to this weak notion.[24]

Definition 6.1 (weakly SS-non-malleable codes)**.** *A stochastic code (that is a pair of maps $Enc : \{0,1\}^k \times \{0,1\}^d \to \{0,1\}^n$, and $\mathrm{Dec} : \{0,1\}^n \to \{0,1\}^k \cup \{fail\}$) is* **weakly SS-non-malleable** *for a class $\mathcal{C}$, with set size h and error ρ, if for every $C \in \mathcal{C}$, there exists a set $H_C \subseteq \{0,1\}^k$, with $|H_C| \leq h$, such that:*

$$\Pr_{\substack{S \leftarrow \{0,1\}^k \\ R \leftarrow \{0,1\}^d}} [\mathrm{Dec}(C(\mathrm{Enc}(S,R))) \notin H_C \cup \{S\} \cup \{fail\}] \leq \rho.$$

This definition requires that every $C \in \mathcal{C}$, there exists a *small set* of messages, that can be guessed in advance, such that it is unlikely that by modifying $\mathrm{Enc}(S,R)$, the channel C can lead Dec to decode a message $\bar{S}$ that is neither S nor in H_C.

This intuitively means that for $S \leftarrow \{0,1\}^k$, while it is possible for $\bar{S} = \mathrm{Dec}(C(\mathrm{Enc}(S,R)))$ to be correlated with S, this correlation is limited, as $\Pr[\bar{S} \in H_C] \geq 1 - \rho$.

This discussion shows that this notion is somewhat different than full fledged non-malleability, and allows some correlation to occur. We remark that a very similar notion termed "bounded-malleability" was defined by Faust et al. [FMVW16] where it was used as an intermediate notion on route to achieving standrad non-malleability.

In our application we will use a stronger form of SS-non-malleability, in which we will allow C to receive many encodings of S, and more crucially, some information $\psi(S)$ (for a specific function ψ) about S. We now define this notion.

Definition 6.2 (SS-non-malleable codes)**.** *Let v be an integer, and ψ be a function that on input $s \in \{0,1\}^k$ returns a string. A stochastic code (that is a pair of maps $Enc : \{0,1\}^k \times \{0,1\}^d \to \{0,1\}^n$, and $\mathrm{Dec} : \{0,1\}^n \to \{0,1\}^k \cup \{fail\}$) is (v,ψ)*-**SS-non-malleable** *for a class $\mathcal{C}$, with set size h and error ρ, if for every $C \in \mathcal{C}$, there exists a set $H_C \subseteq \{0,1\}^k$, with $|H_C| \leq h$, such that:*

$$\Pr_{\substack{S \leftarrow \{0,1\}^k \\ R_1,\ldots R_v \leftarrow \{0,1\}^d}} [\mathrm{Dec}\left(C(\psi(s), \mathrm{Enc}(S,R_1), \ldots, \mathrm{Enc}(S,R_v))\right) \notin H_C \cup \{S\} \cup \{fail\}] \leq \rho.$$

[24] A technical comment is that in contrast to the setting of decoding against channels, in which the channel C, produces an "error pattern" $e = C(z)$ and the "corrupted codeword" is $z \oplus C(Z)$, in this setting it is more natural that C produces the "corrupted codeword" directly, namely that $C(z)$ *is* the "corrupted codeword", and this is the choice made below, following previous work in this area.

For simplicity let's first consider the case that $v = 1$, so that Dec gets to see a single instance of $\mathrm{Enc}(S, R)$ as in Definition 6.1. Note that in Definition 6.2, C also receives some information about S in the form of the function $\psi(S)$. It is instructive to think about the case that $\psi(S) = S$, which means that C has no uncertainty about S (or intuitively, that C can decode). As we require that C cannot produce $\bar{S} \notin H_C$, even in this case, this intuitively should follow because C cannot encode.

We believe that this captures a different intuition about non-malleability. Moreover, this will be crucial for our application in which C will be given information $\psi(S)$ about S, that we do not entirely control, and may give C the ability to know quite a bit about S.

In addition, in our application we allow C to see many encodings of S (using independent random strings), and Definition 6.2 guarantees that this does not help C to break the security of the code.

We will be interested in SS-non-malleable codes against poly-size circuits, and will also require codes that have additional decoding properties (against Ham_p) and pseudorandomness properties, that we list in Section 6.2.

Ball et al [BDK$^+$19], Dachman-Soled, Komargodski and Pass [DKP21], and Ball, Dachman-Soled and Loss [BDL22] gave constructions of non-malleable codes (with the standard definition) against poly-size circuits. These constructions are explicit and rely on complexity theoretic and/or cryptographic assumptions. Faust et al. [FMVW16] and Cheraghchi and Guruswami [CG16] gave Monte-Carlo explicit constructions of non-malleable codes. We cannot use any of these codes, as they don't seem to have the added security against a function ψ that we require in SS-non-Malleability, and do not seem to have the additional "decoding from errors" properties that we also require in Theorem 6.5.

6.2 A Monte-Carlo construction of SS-non-malleable control codes on logarithmic strings

We will be interested in a scenario where there are constants $c_k, c_d, c_b \geq 1$, and the SS-non-malleable codes will be used to encode messages of length $k = c_k \cdot \log N$ to strings of length $n = c_b \cdot \log N$, using $d = c_d \cdot \log N$ random bits. Here, N will be the length of codewords in our final construction of stochastic codes for $\mathrm{Ckt}_p^{N^c}$, and $\mathcal{C}$ will be the class of circuits of size N^c (note that this size is exponential in the length of codewords).

We will also want these codes to have several additional properties:

- We want that the decoding algorithm Dec (in addition to satisfying SS-non-malleability with set size $h = N^{O(1)}$, $v = N$, and some specific function ψ) will also have the property it decodes against Ham_p, (for p that is as close as we want to $\frac{1}{4}$) with success probability one. This means that for every s, r, if $\mathrm{Enc}(s, r)$ is modified in pn positions, then Dec decodes correctly to s.

  This can be obtained if we can choose a constant $\beta > 0$, and guarantee that any two codewords of the code have relative distance $\frac{1}{2} - \beta$.

  (We are not aware of stochastic codes in the literature which are both non-malleable and can decode in the Hamming sense. This is partly, because very different techniques are used to construct codes with the different properties).

- We want that the encoding algorithm Enc (in addition to the aforementioned SS-non-malleability property) will have the following pseudorandomness property: For every $s \in \{0,1\}^{c_k \cdot \log N}$, $\mathrm{Enc}(s, U_d)$ is $\frac{1}{N^{c_\epsilon}}$-pseudorandom for circuits of size N^{c_ϵ}, where N^{c_ϵ} is significantly larger than both h and N^c.

As we want codes on messages of logarithmic length, a random stochastic code $\mathrm{Enc} : \{0,1\}^{c_k \cdot \log N} \times \{0,1\}^{c_d \cdot \log N} \to \{0,1\}^{c_b \cdot \log N}$ takes $\mathrm{poly}(N)$ random bits to sample, and we can obtain a Monte-Carlo construction if we can analyze such random stochastic codes and show that they are small set non-malleable, in addition to the other properties. While we do want all these properties, in this construction, rate is unimportant, and any positive constant rate will do.[25]

Let us define the notion of minimum distance for a stochastic code that was discusses above, and associate it with a suitable maximum likelihood decoder.

Definition 6.3 (minimum distance of a sequence)**.** *The minimum distance of a sequence* $a = (a_1, \ldots, a_m) \in \{0,1\}^b$ *is*

$$\min_{i,i' \in [m] \textit{ s.t. } i \neq i'} \delta(a_i, a_{i'}).$$

The minimum distance of a map $\mathrm{Enc} : \{0,1\}^k \times \{0,1\}^d \to \{0,1\}^n$*, is the minimum distance of the sequence* $(\mathrm{Enc}(s,r))_{s \in \{0,1\}^k, r \in \{0,1\}^d}$.

In the definition above, we measure distance also between pairs (s,r), (s,r') for the same s. This is done to keep the definition simple, and this is an overkill for the next maximum likelihood decoder, which is set up to decode up to p which is half the minimum distance.

Definition 6.4 (Maximum likelihood decoder)**.** *Given a function* $\mathrm{Enc} : \{0,1\}^k \times \{0,1\}^d \to \{0,1\}^b$*, and a parameter* $p > 0$*, the* p**-ML-Decoder** *associated with* Enc*, is the function* $\mathrm{Dec} : \{0,1\}^b \to \{0,1\}^k \cup \{fail\}$ *defined as follows: On input* $z \in \{0,1\}^b$ *do the following:*

- *For every pair* $(s,r) \in \{0,1\}^k \times \{0,1\}^d$*, compute* $\delta_{(s,r)} = \delta(z, \mathrm{Enc}(s,r))$.
- *Let* $\delta' = \min_{(s,r) \in \{0,1\}^k \times \{0,1\}^d} \delta_{(s,r)}$*. If* $\delta' > p$ *output* $fail$.
- *Let* $A = \left\{ s \in \{0,1\}^k : \exists r \in \{0,1\}^d \textit{ s.t. } \delta_{(s,r)} = \delta' \right\}$.
- *Output the element* $s \in A$ *that is minimal according to some order on* $\{0,1\}^k$ *(say lexicographic order).*

Note that once we have chosen and fixed a stochastic code $\mathrm{Enc} : \{0,1\}^{c_k \cdot \log N} \times \{0,1\}^{d \cdot \log N} \to \{0,1\}^{c_b \cdot \log N}$ the ML-dcoder can be implemented in time polynomial in N. This means to get a Monte-Carlop construction we can sample a random stochastic code and pair it with an ML-decoder. We therefore need to show that such a pair of maps, satisfies the three required properties w.h.p. We are finally ready to state the Monte-Carlo construction.

Theorem 6.5 (A Monte-Carlo construction of SS-non-malleable codes on logaithmic strings)**.** *For every constants* $c_s, c_\rho \geq 1$ *there exists a constant* $c_H \geq 1$ *such that for every constants* $c_k, c_\gamma, c_\epsilon \geq 1$ *and* $\beta > 0$*, there exist constant* $c_d, c_b \geq 1$ *such that every sufficiently large integer* N *the following holds: For every choice of* $1 \leq v \leq N$ *and* $\psi : \{0,1\}^{c_k \cdot \log N} \to \{0,1\}^{N^3}$*, a function* $\mathrm{Enc} : \{0,1\}^{c_k \cdot \log N} \times \{0,1\}^{c_d \cdot \log N} \to \{0,1\}^{c_b \cdot \log N}$ *that is chosen uniformly from all function from* $\{0,1\}^{c_k \cdot \log N} \times \{0,1\}^{c_d \cdot \log N}$ *to* $\{0,1\}^{c_b \cdot \log N}$*, satisfies the three properties below with probability at least* $1 - N^{-c_\gamma}$.

Distance and decoding: Enc *has minimum distance* $\frac{1}{2} - \beta$*. This means that for* $p = \frac{\frac{1}{2} - \beta}{2}$*, the p-ML-decoder* Dec *associated with* Enc *decodes for* Ham_p *with success probability one.*

SS-non-Malleability. $(\mathrm{Enc}, \mathrm{Dec})$ *is* (v, ψ)*-SS-non-malleable for circuits of size* N^{c_s} *with set size* N^{c_H} *and error* N^{-c_ρ}.

[25] Cheraghchi and Guruswami [CG16] analyzed random stochastic codes, and showed that they are w.h.p. non-malleable according to the standard definition. Once again, we cannot use this analysis because we require additional properties of SS-non-malleability with respect to a function ψ, as well as decoding from errors.

Pseudorandomness. *For every $s \in \{0,1\}^{c_k \cdot \log N}$, the distribution $\mathrm{Enc}(s, U_{c_d \cdot \log N})$ is N^{-c_ϵ}-pseudorandom for circuits of size N^{c_ϵ}.*

The Proof of Theorem 6.5 appears in Section 6.3. The proof uses the methodology explained in Section 4, which is different than the approach used by Cheraghchi and Guruswami [CG16] that analyzed random stochastic codes with the standard notion of non-malleability.

Interpretation of Theorem 6.5 as an explicit Monte-Carlo construction. In Section 7 we will interpret Theorem 6.5 as a Monte-Carlo construction of a stochastic code with the properties listed above, that runs in time $\mathrm{poly}(N)$, uses $\mathrm{poly}(N)$-Monte-Carlo randomness, and has Monte-Carlo error N^{-c_γ}.

Remark 6.6 (Exponentially small Monte-Carlo error in Theorem 6.5)**.** *We remark that (using repetition) the probability of failure in Theorem 6.5 can be reduced from N^{-c_γ} to exponentially small in N. See Remark 6.15 for precise details.*

6.3 Proof of Theorem 6.5

We first set up some notation for the probability space of choosing a random stochastic code Enc. We will analyze the three different properties separately. Showing that a random stochastic code achieves the distance property and the pseudorandomness property follows by a standard union bound. The main technical difficulty is in the SS-non-Malleability property. Here, we will use the methodology explained in Section 4, and it will also be helpful to imagine that the experiment in which Enc is chosen, is done in two steps: First, a random subset of "potential codewords" is chosen from $\{0,1\}^{c_b \cdot \log N}$ and fixed, and then each pair $(s,r) \in \{0,1\}^{c_k \cdot \log N} \times \{0,1\}^{c_d \cdot \log N}$ will select a codeword $\mathrm{Enc}(S,R)$ from the potential codewords (without repetition). This yields the same distribution, but is beneficial as we can imagine that an adversarial channel circuit C is aware of the choice made in the first step, but not of the choice made in the second step. This will make it easier to analyze the behavior of such a circuit. Details follow:

The setup: Let $c_s, c_\rho \geq 1$ be some constants. We will later choose a constant $c_H \geq 1$, such that for every constants $c_k, c_\gamma, c_\epsilon \geq 1$ and $\beta > 0$, we will show that there exists choices of constants $c_d, c_b \geq 1$ that satisfy the conclusion of the Theorem 6.5.

For that purpose, we fix an integer N, and throughout the proof we will be allowed to assume that N is sufficiently large relative to the constants above. We also fix $1 \leq v \leq N$ and $\psi : \{0,1\}^{c_k \cdot \log N} \to \{0,1\}^{N^3}$. We will show that if the constants are carefully chosen, then a random choice of Enc satisfies the conclusion of the theorem.

We will use the following notation for the probability space of choosing Enc.

Experiment expr_1: A random code.

- We set $\mathcal{J} = \{0,1\}^{c_k \cdot \log N} \times \{0,1\}^{c_d \cdot \log N}$.
- We set $\ell = |\mathcal{J}| = N^{c_k} \cdot N^{c_d}$.
- We consider the following experiment (which we will denote by expr_1).
  - Let $B \in (\{0,1\}^{c_b \cdot \log N})^{\mathcal{J}}$ be a random variable $B = (B_j)_{j \in \mathcal{J}} \leftarrow (\{0,1\}^{c_b \cdot \log N})^{\mathcal{J}}$. (Namely, B consists of ℓ independent, uniformly chosen $B_j \leftarrow \{0,1\}^{c_b \cdot \log N}$).
  - Let $\mathrm{Enc}(s,r) = B_{s,r}$.

Indeed this gives that if $\mathrm{Enc} : \{0,1\}^{c_k \cdot \log N} \times \{0,1\}^{c_d \cdot \log N} \to \{0,1\}^{c_d \cdot \log N}$ is chosen according to expr_1 then it is uniformly from all function from $\{0,1\}^{c_k \cdot \log N} \times \{0,1\}^{c_d \cdot \log N}$ to $\{0,1\}^{c_d \cdot \log N}$.

It is a standard argument to see that Enc chosen in expr_1 satisfies the distance and pseudorandomness properties (we will handle this later). The main difficulty is to show that Enc satisfies the SS-non-malleability property. For this purpose it will be helpful to imagine that Enc is chosen by a different experiment.

Experiment expr_2: A slightly different experiment.

- Let $c_A \geq c_k + c_d$ be a constant that will be chosen later, and let $\ell_A = N^{c_A} \geq N^{c_k} \cdot N^{c_d} = \ell$.
- We consider the following experiment (which we will denote by expr_2).
  - Let $\mathcal{A} = (\{0,1\}^{c_b \cdot \log N})^{\ell_A}$. Let $A = (A_1, \ldots, A_{\ell_A}) \in \mathcal{A}$ be a random variable where $A \leftarrow \mathcal{A}$. (Namely, A consists of ℓ_A random variables $A_1, \ldots, A_{\ell_A} \leftarrow \{0,1\}^{c_b \cdot \log N}$) chosen with replacement).
  - Let $\mathcal{X} = [\ell_A]^{\mathcal{J}}(\mathrm{ds})$. (That is $\mathcal{X}$ is the set of ℓ-tuples $(X_j)_{j \in \mathcal{J}}$ where all elements X_j are distinct). Let $X = (X_j)_{j \in \mathcal{J}} \in \mathcal{X}$ be a random variable $X \leftarrow \mathcal{X}$. (Namely X consists of ℓ random variables $(X_j)_{j \in \mathcal{J}} \overset{\mathrm{wor}}{\leftarrow} [\ell_A]$ chosen without replacement).
  - For every $j \in \mathcal{J}$, let $B_j = A_{X_j}$, and let $B = (B_j)_{j \in \mathcal{J}}$.
  - Let $\mathrm{Enc}(s,r) = B_{s,r}$.

Note that the distribution of B is identical in the two experiments (this can be seen as for each fixing $x \in \mathcal{X}$ of X, the distribution $(B \mid X = x)$ for $B \leftarrow \mathrm{expr}_2$ is identical to $B \leftarrow \mathrm{expr}_1$). Therefore, the distribution of Enc is identical in expr_1 and expr_2.

Intuition for the SS-non-malleability property. The advantage of expr_2 is that it allows us to analyze Enc as follows: We will first use a standard argument to show that we can choose the parameters so that w.h.p $a \leftarrow \mathcal{A}$ satisfies the distance property (that is that for $i \neq i' \in [\ell_A]$, $\delta(a_i, a_{i'}) > \frac{1}{2} - \beta$). We will now focus on the case that $a \leftarrow \mathcal{A}$ is already chosen, fixed, and satisfies the distance property, and so expr_2 amounts to choosing $X \leftarrow \mathcal{X}$.

For such a fixing of $A = a$, we have that for every $z \in \{0,1\}^b$, there is at most one $i \in [\ell_A]$ such that $\delta(z, a_i) \leq p = \frac{\frac{1}{2} - \beta}{2}$. However, whether or not a_i will be a codeword of Enc (meaning that there exists $j \in \mathcal{J}$ such that $X_j = i$) has not yet been determined, and has small probability.

Our goal is to show that for every circuit C of size N^{c_S}, the probability that C breaks the SS-non-malleability property of Enc is exponentially small (say smaller than $2^{-N^{2 \cdot c_S}}$) so that we can do a union bound over all such circuits.

Having fixed $A = a$, allows us to imagine that when a circuit C tries to break the SS-non-malleability property of Enc, it already knows a, but it does not know X. Intuitively, when such a circuit outputs some $z \in \{0,1\}^b$, then we can without loss of generality assume that $z = a_i$ for $i \in [\ell_A]$. This is because if $z \notin \{a_1, \ldots, a_{\ell_A}\}$ then when performing ML-decoding, if the decoding will not fail, then the codeword that will be considered is the unique $b \in \{a_1, \ldots, a_{\ell_A}\}$ that is within distance p of z. (Note however, that the output of $\mathrm{Dec}(z)$ is not yet determined as at this point we do not know which $s \in \{0,1\}^k$ (and also whether there will exists such an s that) will have an $r \in \{0,1\}^d$ such that $a_{X_{(s,r)}} = b$).

Given a circuit C that tries to break the SS-non-malleability property of Enc, we will consider the distribution:

$$B^C = C(\psi(S), a_{Y_1}, \ldots, a_{Y_v}) \text{ for } S \leftarrow \{0,1\}^{c_k \cdot \log N}, Y_1, \ldots, Y_v \leftarrow [\ell_A]$$

Intuitively, C can choose between two strategies:

- There are elements b in the support of B^C which are "heavy", meaning that $\Pr[B^c = b] \geq 1/N^{c_H}$ for some constant c_H.
- All $b \in [\ell_a]$ are "light".

If C implements the first strategy, and assume for simplicity that all the elements in the support of B^C are heavy, then the number of heavy b is obviously bounded by the polynomial N^{c_H}. Intuitively, this should allow us to mark these elements as "belonging to H_C" so that by definition of SS-non-malleability, C cannot win when outputting these elements. (A technicality that we ignore here is that H_C is supposed to be a set of "messages" $s \in \{0,1\}^k$ rather than a set of "codewords). This intuitively means that in some sense, we can deal with heavy elements.

If C implements the second strategy, then intuitively, C is unlikely to win with large probability. This is because, in order to win with sufficiently large probability, it intuitively must be the case that polynomially many of the b's in the support of B^C end up being codewords of Enc (meaning that there exists $(s,r) \in \mathcal{J}$ such that $a_{X_{s,r}} = b$). Intuitively, the probability of polynomially many b's ending up as codewords is exponentially small. (Here we are supposed to benefit from the observation that having fixed a, the events that C succeeds for different a's are intuitively somewhat independent).

Overall, we can hope to show that the probability that C breaks the SS-non-malleability property is exponentially small (no matter what strategy it uses).

There are technical difficulties when trying to convert this high level intuition into a proof.

One difficulty is that we will need to argue that the distribution B^C (which was defined in terms of selecting a codeword from $a_1, \ldots, a_{[\ell_A]}$ is a good approximation to the distribution of selecting a codeword from $(a_{x_j})_{j \leftarrow \mathcal{J}}$ for a typical choice of $x \leftarrow X$ (which is the distribution on which the circuit C is evaluated).

Another difficulty is how to argue that the events that come up in the intuition above are "sufficiently independent" so that we can obtain exponentially small probabilities for events that are conjunctions of polynomially many events that are "intuitively somewhat independent" but not formally independent.

The actual argument handles both difficulties by using the methodology explained in Section 4.

Back to the formal proof. Our plan is to show that when Enc is chosen at random, then for each of the three properties in Theorem 6.5, the probability that it does not hold is small. The theorem will then follow by a union bound.

As explained earlier, the main difficulty is handling the SS-non-malleability property, so we will start by implementing the strategy outlined in the intuition above. More specifically, we assume that Enc is chosen as in experiment expr_2 (which consists of independent choices of $A \leftarrow \mathcal{A}$, $X \leftarrow \mathcal{X}$).

Definition 6.7 (good a)**.** *We say that $a \in \mathcal{A}$ is* good *if a has minimal distance larger than $\frac{1}{2} - \beta$.*

We start by observing that if the parameters are chosen correctly, then the probability that A does not have minimum distance larger than $\frac{1}{2} - \beta$ is very small.

Lemma 6.8 (Distance property for A)**.** *For every choice of constants $c_\gamma, c_A \geq 1$ and $\beta > 0$, if c_b is sufficiently large as a function of c_γ, c_A and β, then for every sufficiently large N,*

$$\Pr_{A \leftarrow \mathcal{A}}[A \textit{ does not have minimum distance larger than } \tfrac{1}{2} - \beta] \leq N^{-10 \cdot c_\gamma}.$$

Proof. The proof of Lemma 6.8 is a completely standard union bound. More specifically, for every $i \neq i' \in [\ell_A]$, using the fact that $H(\frac{1}{2} - \beta) = 1 - O(\beta^2)$ we have that:

$$\begin{aligned}\Pr_{A \leftarrow \mathcal{A}}[\delta(A_i, A_{i'}) \leq \frac{1}{2} - \beta] &\leq 2^{-(1-H(\frac{1}{2}-\beta))\cdot c_b \cdot \log N} \\ &\leq 2^{-(1-(1-O(\beta^2)))\cdot c_b \cdot \log N}] \\ &\leq 2^{-\Omega(\beta^2)\cdot c_b \cdot \log N} \\ &= N^{-\Omega(c_b \cdot \beta^2)}.\end{aligned}$$

The number of pairs $i \neq i' \in [\ell_A]$ is bounded by $\ell_A^2 = N^{2 \cdot c_A}$. By a union bound,

$$\Pr_{A \leftarrow \mathcal{A}}[A \text{ does not have minimum distance larger than } \tfrac{1}{2} - \beta] \leq N^{2\cdot c_A} \cdot N^{-\Omega(c_b \cdot \beta^2)},$$

which is smaller than $N^{-10 \cdot c_\gamma}$ if c_b is sufficiently large as a function of c_γ, c_A and β. □

Before proceeding with handling the SS-non-Malleability property, we note that as the elements of X are distinct, Lemma 6.8 immediately implies the distance property in Theorem 6.5.

Corollary 6.9 (Distance property)**.** *Under the conditions of Lemma 6.8,*

$$\Pr_{\mathrm{Enc} \leftarrow \mathrm{expr}_2}[\mathrm{Enc} \textit{ does not satisfy the distance property}] \leq N^{-10 \cdot c_\gamma}.$$

The main technical lemma in the proof of Theorem 6.5 is the following:

Lemma 6.10. *If the following conditions on the constants $c_s, c_\rho, c_H, c_k, c_d, c_A$ are met:*

- *c_H is sufficiently large as a function of c_s and c_ρ.*
- *c_d is sufficiently large as a function of c_s and c_ρ.*
- *c_A is sufficiently large as a function of c_k, c_d, c_s and c_ρ.*

Then for every sufficiently large N, and for every good $a \in \mathcal{A}$,

$$\Pr_{\mathrm{Enc} \leftarrow \mathrm{expr}_2}[\mathrm{Enc} \textit{ does not satisfy the SS-non-malleability property} \mid A = a] \leq N^{-10 \cdot c_\gamma}.$$

Remark 6.11. *In fact, the probability is exponentially small in N^{c_s}, and this is why the lemma does not mention the constant c_γ.*

The main innovation in the proof of Theorem 6.5 is the proof of Lemma 6.10. This proof appears in Section 6.4.

An immediate consequence of lemma 6.8 and Lemma 6.10 is that if the conditions in both lemmas hold then Enc is likely to satisfy the SS-non-malleability property. This is stated in the next corollary:

Corollary 6.12 (SS non-malleability property)**.** *Under the conditions of Lemma 6.8 and Lemma 6.10,*

$$\Pr_{\mathrm{Enc} \leftarrow \mathrm{expr}_2}[\mathrm{Enc} \textit{ does not satisfy the the SS-non-malleability property}] \leq 2 \cdot N^{-10 \cdot c_\gamma}.$$

It is standard to show that for an appropriate choice of parameters, $\mathrm{Enc} \leftarrow \mathrm{expr}_1$ is likely to have the pseudorandomness property. This is stated below:

Lemma 6.13 (Pseudorandomness property)**.** *For every choice of constants $c_k, c_\epsilon \geq 1$, if c_d is sufficiently large as a function of c_k, c_ϵ, then for every sufficiently large N,*

$$\Pr_{\mathrm{Enc} \leftarrow \mathrm{expr}_1}[\mathrm{Enc} \textit{ does not satisfy the the Pseudorandomness property}] \leq N^{-10 \cdot c_\gamma}.$$

Proof. The proof of Lemma 6.13 is a completely standard union bound. Here it is more convenient to use expr_1. More specifically, for every circuit C of size N^{c_ϵ} and $s \in \{0,1\}^{c_k \cdot \log N}$, the probability over choosing $B \leftarrow \mathrm{expr}_1$ that C distinguishes the output of the function $G(r) = \mathrm{Enc}(s,r)$ from uniform with advantage N^{-c_ϵ}, can be expressed as the probability that then fraction of $(B_{s,r})_{r \in \{0,1\}^{c_d \log N}}$ such that $C(B_{s,r}) = 1$ deviates from $\Pr[C(U_{c_b \cdot \log N}) = 1]$ by N^{-c_ϵ}. Using the independence of $(B_{s,r})_{r \in \{0,1\}^{c_d \log N}}$, by a Chernoff bound, this probability is upper bounded by $2^{-\Omega(\frac{N^{c_d}}{N^{2 \cdot c_\epsilon}})}$. Therefore by a union bound over the at most $2^{N^{c_\epsilon}}$ circuits of size N^{c_ϵ} and N^{c_k} choices of $s \in \{0,1\}^{c_k \cdot \log N}$, the probability that the pseudorandomness property does not hold is at most:

$$2^{N^{c_\epsilon}} \cdot N^{c_k} \cdot 2^{-\Omega(\frac{N^{c_d}}{N^{2 \cdot c_\epsilon}})},$$

which is smaller than $N^{-10 \cdot c_\gamma}$ for every large enough N, if c_d is sufficiently large as a function of c_k, c_ϵ. □

Remark 6.14. *In fact, the probability is exponentially small in N^{c_d}, and this is why the lemma does not mention the constant c_γ.*

In order to conclude the proof of Theorem 6.5 we need to show how to choose the parameters to meet all the conditions in the lemmas above. This is done next.

Choosing the constants to meet all requirements. We are given $c_s, c_\rho \geq 1$. We first choose c_H to be sufficiently large as instructed by the first item of Lemma 6.10. We are then given constants $c_k, c_\gamma, c_\epsilon \geq 1$ and $\beta > 0$. We now choose c_d to be sufficiently large as instructed by the second item of Lemma 6.10, and by Lemma 6.13. We now choose the constant c_A to be sufficiently large as instructed in the third item of Lemma 6.10. Finally, we choose c_b to be sufficiently large as instructed in Lemma 6.8.

Using this process, we meet all the requirements in Lemma 6.8, Lemma 6.10 and Lemma 6.13. Therefore, by a union bound, it follows that the probability that a random Enc fails to meet all the three properties is at most $3 \cdot N^{-10 \cdot c_\gamma} \leq N^{-c_\gamma}$. This concludes the proof of Theorem 6.5.

Remark 6.15 (Exponentially small probability of failure in Theorem 6.5)**.** *The only reason that we did not get an exponentially small probability of failure is that in Lemma 6.8 the probability of failure is not exponentially small. (This is in contrast to Lemma 6.10 and Lemma 6.13 where we do get exponentially small probability).*

We remark that given $a = (a_1, \ldots, a_{\ell_A})$ for $a_1, \ldots, a_{\ell_A} \in \{0,1\}^{c_b \cdot \log N}$, one can test in time polynomial in N, whether a is good. Therefore, if we choose a polynomial number of candidates $A^1, \ldots, A^{p(N)} \leftarrow \mathcal{A}$ and $X^1, \ldots, X^{t(n)} \leftarrow \mathcal{X}$, then with probability $1 - 2^{-t(N)}$, there will exist an $i \in [t(n)]$ such that A^i is good, and if we choose Enc *in* expr_2 *using A^i, X^i then the probability that we will not obtain a code that satisfies all three properties is exponentially small.*

6.4 Proof of Lemma 6.10

In this section we prove Lemma 6.10. Fix some good $a \in \mathcal{A}$, and recall that Lemma 6.10 considers the experiment expr_2 conditioned on the event $\{A = a\}$. In this experiment, once A is fixed, the only remaining random variable is $X \leftarrow \mathcal{X}$. Therefore, in this experiment for every $(s,r) \in \mathcal{J}$, $\mathrm{Enc}(s,r) = a_{X_{s,r}}$.

6.4.1 Preparations for the methodology of Section 4

With this notation, in order to prove Lemma 6.10 we need to prove that:

Lemma 6.16. *With probability at least $1 - N^{10 \cdot c_\gamma}$, over $X \leftarrow \mathcal{X}$, for every circuit C of size N^{c_s} there exists a set $H_C \subseteq \{0,1\}^{c_k \cdot \log N}$ with $|H_C| \leq N^{c_H}$ such that*

$$\Pr_{\substack{S \leftarrow \{0,1\}^{c_k \cdot \log N} \\ R_1,\ldots,R_v \leftarrow \{0,1\}^{c_d \cdot \log N}}} [\mathrm{Dec}(C(\psi(S), a_{X_{S,R_1}}, \ldots, a_{X_{S,R_v}})) \notin H_C \cup \{S\} \cup \{fail\}] < N^{-c_\rho}.$$

We now explain how to define the set H_C for a given circuit C.

Defining a set H_C. Note that we are allowed to define H_C as a function of C (but also as a function of X that was chosen in the experiment $X \leftarrow \mathcal{X}$). We start by defining a circuit C_a for every circuit C. Loosely speaking, C_a receives the same input as C, and after applying C, it outputs the unique $i \in [\ell_A]$ such that the output of C is within distance p to a_i.

Definition 6.17 (The circuit C_a)**.** *Let $C(w, e_1, \ldots, e_v)$ be a size N^{c_s} circuit that receives as input $w \in \{0,1\}^{N^3}$ and $e_1, \ldots, e_v \in \{0,1\}^{c_b \cdot \log N}$. For every good $a \in \mathcal{A}$, we define the function $C_a(w, e_1, \ldots, e_v)$ as follows: if there exist $i \in [\ell_A]$ such that $\delta(C(w, e_1, \ldots, e_v), a_i) \leq p$ then C_a outputs i (and note that if such an i exists, then it is unique because a has minimum distance larger than $\frac{1}{2} - \beta = 2p$). If such an i does not exist then C_a outputs $fail$.*

We now use the circuit C_a to define a small set $T_C \subseteq [\ell_A]$.

Definition 6.18 (The set T_C)**.** *For every circuit C of size N^{c_s} we define:*

$$T_C = \left\{ i \in [\ell_A] : \Pr_{\substack{Y_1,\ldots,Y_v \overset{\text{wor}}{\leftarrow} [\ell_A] \\ S \leftarrow \{0,1\}^{c_k \cdot \log N}}} [C_a(\psi(S), a_{Y_1}, \ldots, a_{Y_v}) = i] \geq \frac{1}{N^{c_H}} \right\}.$$

Note that by definition, we immediately have that:

Claim 6.19. *For every circuit C of size N^{c_s}, $|T_C| \leq N^{c_H}$.*

A crucial observation to keep in mind is that T_C is defined in terms of C and a, but not in terms of X. Loosely speaking, this means that before selecting $X \leftarrow \mathcal{X}$, we can define a fixed set T_C that contains all the indices of "codewords" which C is likely to output in an experiment that is intuitively related to the one that is considered when C tries to break the SS-non-malleability property with a specific choice of $x \in \mathcal{X}$. The point is that we use the same experiment in the definition of T_C regardless of the specific choice of x. This intuitively means that we can identify "likely" outputs of C before knowing which $x \leftarrow \mathcal{X}$ will be chosen.

Once $x \leftarrow \mathcal{X}$ is chosen, we can use T_C (which is a set of indices to a) to define a set $H_{C,x}$ of $s \in \{0,1\}^k$, which are "likely to be decoded" following corruption by C.

Definition 6.20 (The set $H_{C,x}$)**.** *For every circuit C of size N^{c_s}, and $x \in \mathcal{X}$ we define:*

$$H_{C,x} = \left\{ s \in \{0,1\}^{c_k \cdot \log N} : \exists r \in \{0,1\}^{c_d \cdot \log N} \text{ s.t. } x_{s,r} \in T_C \right\}.$$

We now observe that H_C is small.

Claim 6.21. *For every circuit C of size N^{c_s}, and $x \in \mathcal{X}$, $|H_{C,x}| \leq |T_C| \leq N^{c_H}$.*

Proof. We have that elements in $x \in \mathcal{X}$ are of the form $(x_j)_{j \in \mathcal{J}}$ where the x_j's are distinct. It follows that for every $i \in T_C$, there is at most one $j = (s, r)$ such that $x_j = i$. It follows that for every $i \in T_C$ there is at most one $s \in \{0,1\}^{c_k \cdot \log N}$ such that there exists an $r \in \{0,1\}^{d \cdot \log N}$ such that $x_{s,r} = i$. □

6.4.2 The game of a channel C

Continuing with the methodology of Section 4, we will define a game for the circuit C. Here we use the intuition (explained earlier) that it is helpful to imagine that C is aware of the choice of $a \in \mathcal{A}$, but not of $x \in \mathcal{X}$.

Let C be a circuit of size N^{c_s}. For fixed $x \in \mathcal{X}$, $s \in \{0,1\}^{c_k \cdot \log N}$ and $r_1, \ldots, r_v \in \{0,1\}^{c_d \cdot \log N}$, we will be interested in whether C breaks the SS-non-malleability property when the code Enc is defined as in expr_2 (that is $\mathrm{Enc}(j) = a_{x_j}$) and $s, r_1, \ldots, r_v$ are the elements chosen in the inner SS-non-malleability experiment.

Definition 6.22 (Circuit C wins game)**.** *For every circuit C of size N^{c_s}, $x \in \mathcal{X}$, $s \in \{0,1\}^{c_k \cdot \log N}$ and $r_1, \ldots, r_v \in \{0,1\}^{c_d \cdot \log N}$, we say that C* **wins** *on $x, s, (r_1, \ldots, r_v)$ if there exists $(s', r') \in \mathcal{J}$ such that $s' \neq s$ and,*

$$x_{s',r'} = C_a(\psi(s), a_{x_{s,r_1}}, \ldots, a_{x_{s,r_v}}) \notin T_C$$

The next claim justifies the definition above, by showing that whenever C breaks the SS-non-malleability property, it wins in the game defined above.

Claim 6.23. *For every circuit C of size N^{c_s}, $x \in \mathcal{X}$, $s \in \{0,1\}^{c_k \cdot \log N}$ and $(r_1, \ldots, r_v) \in (\{0,1\}^{c_d \cdot \log N})^v$,*

- *If:* $\mathrm{Dec}(C(\psi(s), a_{x_{s,r_1}}, \ldots, a_{x_{s,r_v}})) \notin H_{C,x} \cup \{s\} \cup \{fail\}$.
- *Then: C wins on $x, s, (r_1, \ldots, r_v)$.*

Proof. Assume for contradiction that C doesn't win on $x, s, (r_1, \ldots, r_v)$, and let $i = C_a(\psi(s), a_{x_{s,r_1}}, \ldots, a_{x_{s,r_v}})$. We will show that

$$\mathrm{Dec}(C(\psi(s), a_{x_{s,r_1}}, \ldots, a_{x_{s,r_v}})) \in H_{C,x} \cup \{s\} \cup \{fail\}\,.$$

If $i = fail$ then for every $i' \in [\ell_A]$,

$$\delta(C(\psi(s), a_{x_{s,r_1}}, \ldots, a_{x_{s,r_v}}), a_{i'}) > p,$$

meaning that (by the definition of ML-decoding) $\mathrm{Dec}(C(\psi(s), a_{x_{s,r_1}}, \ldots, a_{x_{s,r_v}}) = fail$, and we are done.

We can therefore assume that $i \neq fail$. If there does not exists $(s', r') \in \mathcal{J}$ such that $x_{s',r'} = i$, then (again by the definition of ML-decoding) $\mathrm{Dec}(C(\psi(s), a_{x_{s,r_1}}, \ldots, a_{x_{s,r_v}}) = fail$, and we are done.

If there exists a pair $(s', r') \in \mathcal{J}$ such that $i = x_{s',r'}$, then this pair is unique, and using the fact that a has minimum distance larger than $2p$ it follows that:

$$\delta(a_{x_{s',r'}}, C(\psi(s), a_{x_{s,r_1}}, \ldots, a_{x_{s,r_v}})) \leq p.$$

It follows that:

$$\mathrm{Dec}(C(\psi(s), a_{x_{s,r_1}}, \ldots, a_{x_{s,r_v}}) = s'.$$

- If $s' = s$ then $\mathrm{Dec}(C(\psi(s), a_{x_{s,r_1}}, \ldots, a_{x_{s,r_v}}) = s$ and we are done.

- If $s' \neq s$ then as we have that C does not win on $x, s, (r_1, \ldots, r_v)$, then it must be that $x_{s',r'} \in T_C$. By the definition of $H_{C,x}$, this gives that $s' \in H_{C,x}$ and we are done.

□

This means that in order to bound the probability that C breaks the SS-non-malleability property, it is sufficient to bound the probability that C wins. More precisely, in order to prove Lemma 6.16, it is sufficient to prove:

$$\Pr_{X \leftarrow \mathcal{X}} \left[\exists C \text{ of size } N^{c_s} \text{ s.t. } \Pr_{\substack{S \leftarrow \{0,1\}^{c_k \cdot \log N} \\ R_1,\ldots,R_v \leftarrow \{0,1\}^{c_d \cdot \log N}}} [C \text{ wins on } X, S, (R_1, \ldots, R_v)] > N^{-c_\rho} \right] < N^{-10 \cdot c_\gamma}.$$

As the number of circuit of size N^{c_s} is smaller than $2^{N^{2 \cdot c_s}}$, and because we are allowed to assume that N is sufficiently large, this will immediately follow by a union bound, if we prove that for every circuit C of size N^{c_s},

$$\Pr_{X \leftarrow \mathcal{X}} \left[\Pr_{\substack{S \leftarrow \{0,1\}^{c_k \cdot \log N} \\ R_1,\ldots,R_v \leftarrow \{0,1\}^{c_d \cdot \log N}}} [C \text{ wins on } X, S, (R_1, \ldots, R_v)] > N^{-c_\rho} \right] < 2^{-N^{3 \cdot c_s}}.$$

Let $\mathcal{Z} = \{0,1\}^{c_k \cdot \log N} \times (\{0,1\}^{c_d \cdot \log N})^v$, so that elements $z \in \mathcal{Z}$ are of the form $z = (s, (r_1, \ldots, r_v))$.

Let $W^C : \mathcal{X} \times \mathcal{Z} \to \{0,1\}$ be the function defined by $W^C(x,z) = 1$ iff C wins on x, z. We also define the function $W^C_{\text{avg}} : \mathcal{X} \to [0,1]$ by:

$$W^C_{\text{avg}}(x) = \frac{1}{|\mathcal{Z}|} \cdot \sum_{z \in \mathcal{Z}} W^C(x,z).$$

With this notation it is sufficient to prove that:

Lemma 6.24. *Under the requirements in Lemma 6.10, for every circuit C of size N^{c_s},*

$$\Pr_{X \leftarrow \mathcal{X}} \left[W^C_{\text{avg}}(X) > N^{-c_\rho} \right] < 2^{-N^{3 \cdot c_s}}.$$

This is exactly the setting that is considered in Section 4 and handled by Lemma 4.1.

6.4.3 Using the recipe of Section 4: proof of Lemma 6.24

We will apply the recipe explained in Section 4. Let us recall the setup. We have fixed a good $a \in \mathcal{A}$, and a circuit C of size N^{c_s}. We need to prove that under the requirements in Lemma 6.10,

$$\Pr_{X \leftarrow \mathcal{X}} \left[W^C_{\text{avg}}(X) > N^{-c_\rho} \right] < 2^{-N^{3 \cdot sc}}.$$

We will use Lemma 4.1 with parameters $t = N^{10 \cdot c_s}$, $\mu = N^{-c_\rho}/3$ and $\delta = 2$. To meet the condition of the lemma, we have to verify that

$$t = N^{10 \cdot c_s} \leq \frac{\delta \cdot \mu \cdot |\mathcal{Z}|}{2} = \frac{N^{-c_\rho}}{3} \cdot N^{c_k + v \cdot c_d},$$

which follows as we are assuming that c_d is sufficiently large as a function of c_s and c_ρ as part of the assumptions of Lemma 6.10. We conclude that to prove Lemma 6.24 it is sufficient to prove that for every $0 \le q < t$,

$$\Pr_{\substack{X \leftarrow \mathcal{X}, Z \leftarrow \mathcal{Z} \\ Z_1,\ldots,Z_q \overset{\text{wor}}{\leftarrow} \mathcal{Z}}} [W(X,Z) = 1 \mid W(X,Z_1) = \ldots = W(X,Z_q) = 1] \le \mu.$$

Fix some $0 \le q < t$. We will now show that we can express the event $\{W(X,Z_1) = \ldots = W(X,Z_q) = 1\}$ as a disjoint union of "simple events".

More specifically, the event $\{W(X,Z_1) = \ldots = W(X,Z_q) = 1\}$ can be viewed as a subset $T \subseteq \mathcal{X} \times \mathcal{Z}^q$ by setting:

$$T = \{(x; z_1, \ldots, z_q) : W(x,z_1) = \ldots = W(x,z_q) = 1\}.$$

We will now show that T can be expressed as a disjoint union of "simple events". We now define what we mean by "simple events". We start with some notation.

Definition 6.25. *For $z^1, \ldots, z^q \in \mathcal{Z}$ such that for every $g \in [q]$, $z^g = (s^g, r_1^g, \ldots, r_v^g)$, we define $J(z^1, \ldots, z^q) = \{(s,r) \in \mathcal{J} : \exists g \in [q], i \in [v] : \ s = s^g, r = r_i^g\}$*

Loosely speaking, a simple event E is a subset $E \subseteq T$ in which $z^1, \ldots, z^q$, as well as several x_j's are fixed, in a very specific way.

Definition 6.26 (Simple event)**.** *For every choice of:*

- *$z^1, \ldots, z^q \in \mathcal{Z}$, such that for every $g \in [q]$, $z^g = (s^g, r_1^g, \ldots, r_v^g)$.*
- *A function $h : J(z^1, \ldots, z^t) \to [\ell_A]$*
- *$j^1, \ldots, j^q \in \mathcal{J}$ such that for each $g \in [q]$, $j^g = (s', r')$ for $s' \neq s^g$.*

We define a set $E \subseteq \mathcal{X} \times \mathcal{Z}^q$ (called the **simple event** *induced by $z^1, \ldots, z^q$, h and $j^1, \ldots, j^q$). The event E is defined by:*

$$E = D \times \left\{(z^1, \ldots, z^q)\right\},$$

where D is the set of all $x \in \mathcal{X}$ such that:

- *For every $j \in J(z^1, \ldots, z^q)$, $x_j = h(j)$.*
- *For every $g \in [q]$, $x_{j^g} = C_a(\psi(s^g), a_{h(s^g, r_1^g)}, \ldots, a_{h(s^g, r_v^g)})$.*
- *$x_{j^g} \notin T_C$.*

We will say that a simple event E is nontrivial if

$$\Pr_{X \leftarrow \mathcal{X}, Z_1,\ldots,Z_q \overset{\text{wor}}{\leftarrow} \mathcal{Z}} [(X, Z_1, \ldots, Z_q) \in E] > 0.$$

Experiment expr_3**: conditioning on a simple event.** We will be interested in the distribution obtained by conditioning the distribution $(X \leftarrow \mathcal{X}, Z_1, \ldots, Z_q \overset{\text{wor}}{\leftarrow} \mathcal{Z})$ on $\{(X, Z_1, \ldots, Z_q) \in E\}$ for a nontrivial simple event E induced by $z^1, \ldots, z^q$, h and $j^1, \ldots, j^q$. Let us denote this experiment by $\text{expr}_3(z^1, \ldots, z^q; h; j^1, \ldots, j^q)$. We observe that for $(X, Z_1, \ldots, Z_q) \leftarrow \text{expr}_3(z^1, \ldots, z^q; h; j^1, \ldots, j^q)$ we have that:

- For every $g \in [q]$, $Z_g = (S^g, R_1^g, \ldots, R_v^g)$ is fixed to $z^g = (s^g, r_1^g, \ldots, r_v^g)$.
- For every $j \in J(z^1, \ldots, z^q)$, X_j is fixed to $h(j)$.

- For every $g \in [q]$, X_{j^g} is fixed to $C_a(\psi(s^g), a_{h(s^g,r_1^g)}, \ldots, a_{h(s^g,r_v^g)})$ that is not in T_C, and furthermore,

$$\begin{aligned} X_{j^q} &= C_a(\psi(s^g), a_{h(s^g,r_1^g)}, \ldots, a_{h(s^g,r_v^g)}) \\ &= C_a(\psi(S^g), a_{X_{S^g,R_1^g}}, \ldots, a_{X_{S^g,R_v^g}}). \end{aligned}$$

- We also have that every simple event E, satsifies $E \subseteq T = \{W(X, Z_1) = \ldots = W(X, Z_q) = 1\}$. This is because for every $g \in [q]$, in the event E, we have that for every $g \in [q]$:

$$\begin{aligned} C_a(\psi(s^g), a_{x_{s^g,r_1^g}}, \ldots, a_{x_{s^g,r_v^g}}) &= C_a(\psi(s^g), a_{h(s^g,r_1^g)}, \ldots, a_{h(s^g,r_v^g)}) \\ &= x_{j^g} \notin T_C \cup \left\{x_{s^g,r} : r \in \{0,1\}^{c_d \cdot \log N}\right\} \cup \{fail\}, \end{aligned}$$

  where the last inequality uses the requirements that $x_{j^g} \notin T_C$, and that $j^g = (s', r')$ for $s' \neq s^g$, which implies that $x_{j^g} \notin \cup \left\{x_{s^g,r} : r \in \{0,1\}^{c_d \cdot \log N}\right\}$.

  This means that for every $(x, z^1, \ldots, z^q) \in E$, C wins on $(x, s^g, r_1^g, \ldots, r_v^g)$ for every $g \in [q]$.

- By definition, every element in $(x, z^1, \ldots, z^q) \in T$ belongs to some simple event. This is because if C wins on $(x, z^1), \ldots, (x, z^q)$, then there exist $j^1, \ldots, j^q$ such that for every $g \in [q]$,

  - $x_{j^q} = C_a(\psi(s^g), a_{x_{(s^g,r_1^g)}}, \ldots, a_{x_{(s^g,r_v^g)}}) \notin T_C$.
  - $j^g = (s', r')$ for $s' \neq s^g$.

- By definition, every two simple events are either equal or disjoint. This is because in order for two simple events to have a non-empty intersection, the two events must agree on $z_1, \ldots, z_q$. Using the fact that the elements of $x \in \mathcal{X}$ are disjoint, it follows that the two events must also agree on h and on $j^1, \ldots, j^q$. Thus, they must coincide.

The discussion above implies that:

- The event $T = \{W(X, Z_1) = \ldots = W(X, Z_q) = 1\}$ is a disjoint union of nontrivial simple events.
- In order to show that:

$$\Pr_{\substack{X \leftarrow \mathcal{X}, Z \leftarrow \mathcal{Z} \\ Z_1, \ldots, Z_q \overset{\text{wor}}{\leftarrow} \mathcal{Z}}} [W(X, Z) = 1 \mid W(X, Z_1) = \ldots = W(X, Z_q) = 1] \leq \mu,$$

  it is sufficient to show that for every choice of nontrivial simple event E that is induced by some $z^1, \ldots, z^q$, h and $j^1, \ldots, j^q$:

$$\Pr_{\substack{X \leftarrow \mathcal{X}, Z \leftarrow \mathcal{Z} \\ Z_1, \ldots, Z_q \overset{\text{wor}}{\leftarrow} \mathcal{Z}}} [W(X, Z) = 1 \mid (X, Z_1, \ldots, Z_q) \in E] \leq \mu,$$

- Simplifying the expression above, and using our choice of $\mu = \frac{N^{-c_\rho}}{3}$ it is sufficient to show the following for every choice of nontrivial simple event E that is induced by some $z^1, \ldots, z^q$, h and $j^1, \ldots, j^q$:

$$\Pr_{\substack{X \leftarrow \mathrm{expr}_3(z^1, \ldots, z^q; h; j^1, \ldots, j^q) \\ Z \leftarrow \mathcal{Z}}} [W(X, Z) = 1] \leq \frac{N^{-c_\rho}}{3}. \tag{6}$$

In the remainder of the proof, we will prove that (6) holds. We will fix some nontrivial simple event E that is induced by some $z^1, \dots, z^q$, h and $j^1, \dots, j^q$, and to avoid clutter, we will define:

$$\mathrm{expr}_3 = \mathrm{expr}_3(z^1, \dots, z^q; h; j^1, \dots, j^q).$$

In the next definition and claim, we express the left hand side of (6) more specifically.

Definition 6.27. *For every $(s', r') \in \mathcal{J}$, let*

$$p_{s',r'} = \Pr_{\substack{X \leftarrow \mathrm{expr}_3 \\ (S, R_1, \dots, R_v) \leftarrow \mathcal{Z}}} [\left\{ C_a(\psi(S), a_{X_{S,R_1}}, \dots, a_{X_{S,R_v}}) = X_{s',r'} \right\} \cap \left\{ X_{s',r'} \notin T_C \right\} \cap \left\{ S \neq s' \right\}].$$

Claim 6.28.

$$\Pr_{\substack{X \leftarrow \mathrm{expr}_3 \\ Z \leftarrow \mathcal{Z}}} [W(X, Z) = 1] = \sum_{(s',r') \in \mathcal{J}} p_{s',r'}.$$

Proof. Let $P = \Pr_{\substack{X \leftarrow \mathrm{expr}_3 \\ Z \leftarrow \mathcal{Z}}} [W(X, Z) = 1]$. We have that:

$$\begin{aligned}
P &= \Pr_{\substack{X \leftarrow \mathrm{expr}_3 \\ (S, R_1, \dots, R_v) \leftarrow \mathcal{Z}}} [W(X, (S, R_1, \dots R_v)) = 1] \\
&= \Pr_{\substack{X \leftarrow \mathrm{expr}_3 \\ (S, R_1, \dots, R_v) \leftarrow \mathcal{Z}}} [C \text{ wins on } X, S, (R_1, \dots, R_v)] \\
&= \Pr_{\substack{X \leftarrow \mathrm{expr}_3 \\ (S, R_1, \dots, R_v) \leftarrow \mathcal{Z}}} [\exists (s', r') \in \mathcal{J} \text{ s.t. } S \neq s' \text{ and } X_{s',r'} = C_a(\psi(S), a_{X_{S,R_1}}, \dots, a_{X_{S,R_v}}) \text{ and } X_{s',r'} \notin T_C] \\
&= \sum_{(s',r') \in \mathcal{J}} \Pr_{\substack{X \leftarrow \mathrm{expr}_3 \\ (S, R_1, \dots, R_v) \leftarrow \mathcal{Z}}} [S \neq s' \text{ and } X_{s',r'} = C_a(\psi(S), a_{X_{S,R_1}}, \dots, a_{X_{S,R_v}}) \text{ and } X_{s',r'} \notin T_C] \\
&= \sum_{(s',r') \in \mathcal{J}} \Pr_{\substack{X \leftarrow \mathrm{expr}_3 \\ (S, R_1, \dots, R_v) \leftarrow \mathcal{Z}}} [\left\{ C_a(\psi(S), a_{X_{S,R_1}}, \dots, a_{X_{S,R_v}}) = X_{s',r'} \right\} \cap \left\{ X_{s',r'} \notin T_C \right\} \cap \left\{ S \neq s' \right\}] \\
&= \sum_{(s',r') \in \mathcal{J}} p_{s',r'}.
\end{aligned}$$

□

We use the following notation.

Definition 6.29. *Let* $\mathrm{Fixed} = J(z^1, \dots, z^q) \cup \left\{ j^1, \dots, j^q \right\}$. *Let* $\mathrm{NotFixed} = \mathcal{J} \setminus \mathrm{Fixed}$.

We note that when choosing $X \leftarrow \mathrm{expr}_3$, for $j \in \mathrm{Fixed}$, the variable X_j is fixed, while $(X_j)_{j \in \mathrm{NotFixed}}$ is distributed like $|\,\mathrm{NotFixed}\,| = \ell - |\,\mathrm{Fixed}\,|$ variables that are chosen from a large set without replacement.

Claim 6.30.

- $|\,\mathrm{Fixed}\,| \leq q \cdot (v + 1)$.
- *There exists* $h' : \mathrm{Fixed} \to [\ell_A]$ *such that for every* $j \in \mathrm{Fixed}$*,*

$$\Pr_{X \leftarrow \mathrm{expr}_3} [X_j = h'(j)] = 1.$$

- *Let* $L = [\ell_A] \setminus \{h'(j) : j \in \mathrm{Fixed}\}$*, and note that* $|L| = \ell_a - |\,\mathrm{Fixed}\,| \geq \ell_a - q \cdot (v+1)$.

  In the experiment $X \leftarrow \mathrm{expr}_3$*, the random variable* $(X_j)_{j \in \mathrm{NotFixed}}$ *is distributed like* $\ell - |\,\mathrm{Fixed}\,|$ *variables chosen from* L *without replacement.*

Proof. (of Claim 6.30) The first item follows because by definition: $|J(z_1, \ldots, z_q)| \leq q \cdot v$, and so $|\,\mathrm{Fixed}\,| \leq q \cdot v + v = q \cdot (v+1)$. The second item follows by the definition of a simple event. For the third item, we recal that $X \leftarrow \mathcal{X}$ is distributed like $\ell = |\mathcal{J}|$ variables chosen from $[\ell_A]$ without replacement. After conditioning on the event E, the value of X_j for $j \in \mathrm{Fixed}$ have been fixed to $h'(j)$. This means that the remaining values can no longer take the values in $\{h'(j) : j \in \mathrm{Fixed}\}$, but are free to take the remaining values. □

We will now show an upper bound on $p_{s',r'}$. The bounds will depend on whether or not $(s', r') \in \mathrm{Fixed}$.

Claim 6.31. *If* $(s', r') \in \mathrm{NotFixed}$ *then* $p_{s',r'} \leq \frac{1}{\ell_A - (q+1)\cdot(v+1)} \leq \frac{1}{N^{c_A} - N^{10 \cdot c_s + 2}}$.

Proof. Let $(s', r') \in \mathrm{NotFixed}$. This intuitively means that $X_{s',r'}$ has not been fixed, and is therefore unlikely to be predicted by C_a.

More formally, we first note that for every $(s, r_1, \ldots, r_v) \in \mathcal{Z}$ such that $s \neq s'$, and every v distinct values $x_{s,r_1}, \ldots, x_{s,r_v} \in [\ell_A]$, the distribution

$$X' = (X \leftarrow \mathrm{expr}_3 \mid X_{s,r_1} = x_{s,r_1}, \ldots, X_{s,r_v} = x_{s,r_v})$$

satisfies that $X'_{s',r'}$ is uniformly distributed over $L \setminus \{x_{s,r_1}, \ldots, x_{s,r_v}\}$, which is a set of size at least $|L| - v$ (which by Claim 6.30 is at least $\ell_A - q \cdot (v+1) - v \geq \ell_A - (q+1) \cdot (v+1)$). It follows that:

$$\Pr_{X \leftarrow \mathrm{expr}_3}[C_a(\psi(s), a_{X_{s,r_1}}, \ldots, a_{X_{s,r_v}}) = X_{s',r'}] \leq \frac{1}{\ell_A - (q+1)\cdot(v+1)},$$

(This is because the expression of the left hand side is fixed).

We can now conclude that for random variables $S \neq s'$ and $R_1, \ldots, R_v$ that are chosen independently of X, the inequality above also holds. More specifically, that:

$$\Pr_{\substack{X \leftarrow \mathrm{expr}_3 \\ (S, R_1, \ldots, R_v) \leftarrow \mathcal{Z}}}[C_a(\psi(S), a_{X_{S,R_1}}, \ldots, a_{X_{S,R_v}}) = X_{s',r'} \mid S \neq s'] \leq \frac{1}{\ell_A - (q+1)\cdot(v+1)}. \tag{7}$$

This follows because having conditioned on $\{S \neq s'\}$ (and using the fact that X and S are independent) the probability above is a convex combination of the probabilities considered in the previous inequality.

We are now ready to prove the bound.

$$\begin{aligned}
p_{s',r'} &= \Pr_{\substack{X \leftarrow \mathrm{expr}_3 \\ (S, R_1, \ldots, R_v) \leftarrow \mathcal{Z}}}[\Big\{C_a(\psi(S), a_{X_{S,R_1}}, \ldots, a_{X_{S,R_v}}) = X_{s',r'}\Big\} \cap \Big\{X_{s',r'} \notin T_C\Big\} \cap \Big\{S \neq s'\Big\}] \\
&\leq \Pr_{\substack{X \leftarrow \mathrm{expr}_3 \\ (S, R_1, \ldots, R_v) \leftarrow \mathcal{Z}}}[\Big\{C_a(\psi(S), a_{X_{S,R_1}}, \ldots, a_{X_{S,R_v}}) = X_{s',r'}\Big\} \cap \Big\{S \neq s'\Big\}] \\
&\leq \Pr_{\substack{X \leftarrow \mathrm{expr}_3 \\ (S, R_1, \ldots, R_v) \leftarrow \mathcal{Z}}}[C_a(\psi(S), a_{X_{S,R_1}}, \ldots, a_{X_{S,R_v}}) = X_{s',r'} \mid S \neq s'] \\
&\leq \frac{1}{\ell_A - (q+1)\cdot(v+1)} \\
&\leq \frac{1}{N^{c_A} - N^{10 \cdot c_s + 2}}
\end{aligned}$$

where the penultimate inequality follows from (7), and the final inequality follows for large enough N because $\ell_A = N^{c_A}$, $q+1 \leq t = N^{10\cdot c_s}$ and $v \leq N$. $\square$

Claim 6.32. *If* $(s', r') \in \mathrm{Fixed}$ *then* $p_{s',r'} \leq \frac{1}{N^{c_H}} + \frac{N^{10\cdot c_s+4}}{N^{c_k+c_d}}$.

Proof. Let $(s', r') \in \mathrm{Fixed}$. By Claim 6.30 this means that for $X \leftarrow \mathrm{expr}_3$, $X_{s',r'}$ is fixed to some value $x_{s',r'} \in [\ell_A]$.

$$\begin{aligned}
p_{s',r'} &= \Pr_{\substack{X\leftarrow \mathrm{expr}_3\\(S,R_1,\ldots,R_v)\leftarrow\mathcal{Z}}}[\Big\{C_a(\psi(S), a_{X_{S,R_1}},\ldots,a_{X_{S,R_v}}) = X_{s',r'}\Big\} \cap \big\{X_{s',r'} \notin T_C\big\} \cap \big\{S \neq s'\big\}]\\
&\leq \Pr_{\substack{X\leftarrow \mathrm{expr}_3\\(S,R_1,\ldots,R_v)\leftarrow\mathcal{Z}}}[\Big\{C_a(\psi(S), a_{X_{S,R_1}},\ldots,a_{X_{S,R_v}}) = X_{s',r'}\Big\} \cap \big\{X_{s',r'} \notin T_C\big\}]\\
&\leq \Pr_{\substack{X\leftarrow \mathrm{expr}_3\\(S,R_1,\ldots,R_v)\leftarrow\mathcal{Z}}}[\Big\{C_a(\psi(S), a_{X_{S,R_1}},\ldots,a_{X_{S,R_v}}) = x_{s',r'}\Big\} \cap \big\{x_{s',r'} \notin T_C\big\}],\\
&\leq \Pr_{\substack{X\leftarrow \mathrm{expr}_3\\(S,R_1,\ldots,R_v)\leftarrow\mathcal{Z}}}[C_a(\psi(S), a_{X_{S,R_1}},\ldots,a_{X_{S,R_v}}) = x_{s',r'}],
\end{aligned}$$

where we can assume that $x_{s',r'} \notin T_C$ (as otherwise the probability is zero). We recall that by the definition of T_C (Definition 6.18), we have that:

$$\Pr_{\substack{Y_1,\ldots,Y_v \overset{\mathrm{wor}}{\leftarrow}[\ell_A]\\ S\leftarrow\{0,1\}^{c_k\cdot\log N}}}\Big[C_a(\psi(S), a_{Y_1},\ldots,a_{Y_v}) = x_{s',r'}\Big] < \frac{1}{N^{c_H}}.$$

The difference between the two considered probabilities is that:

- In the first one the indices of a are $X_{S,R_1},\ldots,X_{S,R_v}$ where $X \leftarrow \mathrm{expr}_3$, $(S,R_1,\ldots,R_v) \leftarrow \mathcal{Z}$. Let us denote this distribution by $Q_1,\ldots,Q_v$.
- In the second one the indices of a are $Y_1,\ldots,Y_v \overset{\mathrm{wor}}{\leftarrow} [\ell_A]$.

Let $\eta \geq 0$ be the statistical distance between $Q_1,\ldots,Q_v$ and $Y_1,\ldots,Y_v$. It follows that:

$$\begin{aligned}
p_{s',r'} &\leq \Pr_{\substack{X\leftarrow \mathrm{expr}_3\\(S,R_1,\ldots,R_v)\leftarrow\mathcal{Z}}}[C_a(\psi(S), a_{X_{S,R_1}},\ldots,a_{X_{S,R_v}}) = x_{s',r'}]\\
&\leq \Pr_{\substack{Y_1,\ldots,Y_v \overset{\mathrm{wor}}{\leftarrow}[\ell_A]\\ S\leftarrow\{0,1\}^{c_k\cdot\log N}}}\Big[C_a(\psi(S), a_{Y_1},\ldots,a_{Y_v}) = x_{s',r'}\Big] + \eta\\
&< \frac{1}{N^{c_H}} + \eta.
\end{aligned}$$

It remains to bound η. For this purpose, we will introduce a hybrid distribution $Y'_1,\ldots,Y'_v \overset{\mathrm{wor}}{\leftarrow} [L]$.

- Let η_1 be the statistical distance between $Q_1,\ldots,Q_v$ and $Y'_1,\ldots,Y'_v$.
- Let η_2 be the statistical distance between $Y'_1,\ldots,Y'_v$ and $Y_1,\ldots,Y_v$.

To bound η_1 we notice that for every choice of $(s, r_1, \ldots, r_v) \in \mathcal{Z}$ such that for every $i \in [v]$, $(s, r_i) \in \mathrm{NotFixed}$, by Claim 6.30, the distribution of $X_{s,r_1}, \ldots, X_{s,r_v}$ for $X \leftarrow \mathrm{expr}_3$ is identical to $Y'_1, \ldots, Y'_v$. This means that:

$$\begin{aligned}
\eta_1 &\leq \Pr_{(S,R_1,\ldots,R_v)\leftarrow\mathcal{Z}}[\exists i \in [v] \text{ s.t. } (S, R_i) \in \mathrm{Fixed}] \\
&\leq \sum_{i\in[v]} \Pr_{(S,R_1,\ldots,R_v)\leftarrow\mathcal{Z}}[(S, R_i) \in \mathrm{Fixed}] \\
&\leq v \cdot \frac{|\,\mathrm{Fixed}\,|}{\ell} \\
&\leq \frac{N^{10\cdot c_s+3}}{N^{c_k+c_d}},
\end{aligned}$$

where the last inequality follows because $v \leq N$, $|\,\mathrm{Fixed}\,| \leq q \cdot (v + 1) \leq t \cdot N^2 \leq N^{10\cdot c_s+2}$, and $\ell = N^{c_k+c_d}$.

To bound η_2 we observe that both distributions $Y'_1, \ldots, Y'_v$ and $Y_1, \ldots, Y_v$ are chosen without replacement but from different sets. The former chooses from $L \subseteq [\ell_A]$ (which by Lemma 6.30 is of size at least $\ell_A - q \cdot (v+1)$) and the latter chooses from $[\ell_A]$. Therefore the statistical distance η_2 is bounded as follows:

$$\begin{aligned}
\eta_2 &\leq \Pr_{Y_1,\ldots,Y_v \overset{\mathrm{wor}}{\leftarrow} [\ell_A]}[\exists i \in [v] : Y_i \in [\ell_A] \setminus L] \\
&\leq \sum_{i\in[v]} \Pr_{Y_1,\ldots,Y_v \overset{\mathrm{wor}}{\leftarrow} [\ell_A]}[Y_i \in [\ell_A] \setminus L] \\
&\leq v \cdot \frac{q \cdot (v + 1)}{\ell_A} \\
&\leq \frac{N^{10\cdot c_s+3}}{N^{c_A}},
\end{aligned}$$

where the last inequality follows because $v \leq N$, $q \leq t \leq N^{10\cdot c_s}$ and $\ell_A = N^{c_A}$. Overall, we get that:

$$\eta \leq \eta_1 + \eta_2 \leq \frac{N^{10\cdot c_s+3}}{N^{c_k+c_d}} + \frac{N^{10\cdot c_s+3}}{N^{c_A}} \leq \frac{N^{10\cdot c_s+4}}{N^{c_k+c_d}},$$

where the last inequality is using the requirement that $N^{c_A} = \ell_A \geq \ell = N^{c_k+c_d}$. □

Putting everything together, we get that:

$$\begin{aligned}
\Pr_{\substack{X\leftarrow\mathrm{expr}_3 \\ Z\leftarrow\mathcal{Z}}}[W(X, Z) = 1] &= \sum_{(s',r')\in\mathcal{J}} p_{s',r'} \\
&= \sum_{(s',r')\in\mathrm{NotFixed}} p_{s',r'} + \sum_{(s',r')\in\mathrm{Fixed}} p_{s',r'} \\
&\leq \ell \cdot \frac{1}{N^{c_A} - N^{10\cdot c_s+2}} + q \cdot (v + 1) \cdot (\frac{1}{N^{c_H}} + \frac{N^{10\cdot c_s+4}}{N^{c_k+c_d}}) \\
&\leq \frac{N^{c_k+c_d}}{N^{c_A} - N^{10\cdot c_s+2}} + N^{10\cdot c_s+2} \cdot (\frac{1}{N^{c_H}} + \frac{N^{10\cdot c_s+4}}{N^{c_k+c_d}}) \\
&\leq \frac{N^{-c_\rho}}{3},
\end{aligned}$$

where the first inequality follows from Claim 6.31 and Claim 6.32, the penultimate inequality follows because $\ell = N^{c_k + c_d}$, $v \leq N$ and $q \leq t = N^{10 \cdot c_s}$, and the final inequality follows from the requirements of Lemma 6.10.

7 A Monte-Carlo construction of stochastic codes for poly-size circuits

In this section we prove our main result, providing an explicit Monte-Carlo construction of stochastic codes for poly-size circuits, proving Theorem 1.1. We start by restating the theorem in a more precise way:

Theorem 7.1 (Explicit stochastic codes for poly-size channels)**.** *For every constants $p < \frac{1}{4}$, $c > 1$, and for every sufficiently small constant $\epsilon > 0$, there exists a constant d, such that for every sufficiently large N, there is a Monte-Carlo stochastic code for $\mathrm{Ckt}_p^{N^c}$ with:*

- *Rate $R \geq 1 - H(p) - \epsilon$.*
- *Success probability $1 - \frac{1}{N^c}$.*
- *N^d bits of Monte-Carlo randomness.*
- *Monte-Carlo error $\frac{1}{N^c}$.*

Furthermore, the Monte-Carlo construction is explicit.

Recall, that a Monte-Carlo explicit construction means that there is a pre-processing stage in which a uniform string y of polynomial length N^d is chosen, and hardwired (once and for all) to the encoding and decoding algorithms $(\mathrm{Enc}, \mathrm{Dec})$. It is guaranteed that with probability $1 - \frac{1}{N^c}$ over the choice of y, the algorithms $(\mathrm{Enc}, \mathrm{Dec})$ form a stochastic code for $\mathrm{Ckt}_p^{N^c}$ with success $1 - \frac{1}{N^c}$. Furthermore, Enc and Dec run in time N^d.

In the remainder of the section we prove Theorem 7.1 (recall that an overview of this proof is given in Section 2). In Section 7.1 we present our construction. In Section 7.2 we compare our construction to those used in previous work. The proof is given in Section 7.3.

7.1 The construction

In this section we present our construction of stochastic codes for bounded channels. The construction is detailed in Four figures: Figure 1 lists parameters, Figure 2 lists ingredients that we use, Figure 3 describes the encoding algorithm, and Figure 4 describes the decoding algorithm.

We start with some notation and definitions. We remark that an intuitive explanation of the construction appears in Section 2.

Partitioning codewords into control blocks and data blocks. The construction will think of codewords $c \in \{0,1\}^N$ as being composed of $n = n_{\mathrm{ctrl}} + n_{\mathrm{data}}$ blocks of length $b = N/n$. We specify the precise choices of $n, b, n_{\mathrm{ctrl}}, n_{\mathrm{data}}$ in Figure 1.

We now set up some notation. Given a subset $I \subseteq [n]$ of n_{ctrl} distinct indices, we can decompose c into its data part $c_{\mathrm{data}} \in \{0,1\}^{N_{\mathrm{data}} = n_{\mathrm{data}} \cdot b}$ and its control part $c_{\mathrm{ctrl}} \in \{0,1\}^{N_{\mathrm{ctrl}} = n_{\mathrm{ctrl}} \cdot b}$. Similarly, given strings c_{data} and c_{ctrl} we can prepare the codeword c (which we denote by $(c_{\mathrm{data}}, c_{\mathrm{ctrl}})^I$ by the reverse operation. This is stated formally in the definition below.

Figure 1: Parameters for stochastic code

In this figure we make some of the parameter choices for the construction of Theorem 7.1.

We are given constants:

- $0 < p < \frac{1}{4}$ - The fraction of errors we need to recover from.
- $\epsilon > 0$ - We will construct a stochastic code Enc with output length N, and rate $R \geq 1 - H(p) - \epsilon$. We assume that $\epsilon > 0$ is sufficiently small in terms of p.
- c - We are aiming to construct a stochastic code for circuits of size N^c with success $1 - \nu$ for $\nu = \frac{1}{N^c}$.

Other parameters that we choose:

- N - The length (in bits) of the codeword. Throughout, we assume that N is sufficiently large, and that other parameters are chosen as a function of N.
- Let $b = c_b \cdot \log N$, where c_b is a constant that we choose later in Figure 2.
- Let $n = N/b$. We split the N output bits of the codeword to n blocks of length b.
- Let $n_{\text{ctrl}} = n^{0.1}$ be the number of "control blocks", and $n_{\text{data}} = n - n_{\text{ctrl}}$ be the number of "data blocks".
- Let $N_{\text{ctrl}} = b \cdot n_{\text{ctrl}}$ and $N_{\text{data}} = b \cdot n_{\text{data}}$. (Note that: $n = n_{\text{ctrl}} + n_{\text{data}}$, $N = N_{\text{ctrl}} + N_{\text{data}}$).
- Let $c_0 > 1$ be a sufficiently large universal constant that we will choose in the proof of Theorem 7.1.

We use these choices to choose ingredients (a sampler, a PRG, a pseudorandomly chosen permutation, a control code, and a BSC code) that will be used in the construction. This choice is made in Figure 2.

Definition 7.2 (Data and control portion of a codeword)**.** *We view strings $c \in \{0,1\}^N$ as composed of n blocks of length $b = N/n$, so that $c \in (\{0,1\}^b)^n$, and c_i denotes the b bit long i'th block of c.*

Let $I = \{i_1, \ldots, i_{n_{\text{ctrl}}}\} \subseteq [n]$ be a subset of indices of size n_{ctrl}.

- *Given strings $c_{\text{data}} \in \{0,1\}^{N_{\text{data}}}$ and $c_{\text{ctrl}} \in \{0,1\}^{N_{\text{ctrl}}}$ we define an N bit string c denoted by $(c_{\text{data}}, c_{\text{ctrl}})^I$ as follows: We think of $c_{\text{data}}, c_{\text{ctrl}}, c$ as being composed of blocks of length b (that is $c_{\text{data}} \in (\{0,1\}^b)^{n_{\text{data}}}$, $c_{\text{ctrl}} \in (\{0,1\}^b)^{n_{\text{ctrl}}}$ and $c \in (\{0,1\}^b)^n$). We enumerate the indices in $[n] \setminus I$ by $j_1, \ldots, j_{n_{\text{data}}}$ and set $c_\ell = \begin{cases} (c_{\text{ctrl}})_k & \text{if } \ell = i_k \text{ for some } k; \\ (c_{\text{data}})_k & \text{if } \ell = j_k \text{ for some } k \end{cases}$*
- *Given a string $c \in \{0,1\}^N$ (which we think of as $c \in (\{0,1\}^b)^n$) we define strings $c^I_{\text{data}}, c^I_{\text{ctrl}}$ by $c^I_{\text{ctrl}} = c|_I$ and $c^I_{\text{data}} = c|_{[n]\setminus I}$, (namely the strings restricted to the indices in I, $[n] \setminus I$, respectively).*

We omit the superscript I when it is clear from the context.

7.2 Comparison of the construction to earlier work

Our construction builds on ideas from earlier work by Guruswami and Smith [GS16]. It also incorporates ideas that were introduced later in [SS21a, KSS19, SS21b], as well as several key new ideas.

Loosely speaking, the construction imitates the list-decodable codes of Guruswami and Smith [GS16] and then applies additional ideas to trim the list to a single candidate. We have already described our high level approach for trimming the list in Section 2. Our approach of using evasiveness for discarding incorrect candidates $\bar{S}_i$ that are uncorrelated with S is inspired by the recent work of Shaltiel and Silbak [SS21b] on codes for small space channels. However, in [SS21b] channels are significantly weaker, which leads

Figure 2: Ingredients for stochastic code

In Figure 1 we specified parameters that are used by the construction. More specifically, when given constants p, ϵ, c, and a specified codeword length N, we have chosen parameters $n, b, n_{\text{ctrl}}, n_{\text{data}}, N_{\text{ctrl}}, N_{\text{data}}$ that were chosen as a function of previous choices, and of constants $c_b, c_0 \geq 1$, that were not yet specified. In this figure, we will specify the ingredients that will be used in our construction, and choose c_b. The constant c_0 will be chosen in the proof.

Parameter choices for applying Theorem 6.5: We choose a constant $c_s = c + c_0$ and will apply Theorem 6.5 with c_s and $c_\rho = c_s$ to obtain a constant c_H. At this point, Theorem 6.5 allows us to choose additional constants $c_\gamma, c_k, c_\epsilon \geq 1$ and $\beta > 0$. We choose: $\beta = \frac{\frac{1}{4}-p}{8}$, $c_\gamma = c_s = c + c_0$. We will choose c_k and c_ϵ later on, and continue with the consequences of Theorem 6.5 after this choice is made.

Ingredients:

BSC code: Let $p_{\text{BSC}} = p \cdot (1 + \frac{\epsilon}{10})$. We apply Theorem 5.7 using $p_{\text{BSC}}, \epsilon/3, N_{\text{data}}, c_s + c_H + c_0$ as choices for p, ϵ, N, c, respectively, to obtain a code. Theorem 5.7 gives an explicit Monte-Carlo construction of codes, and we will use part of the polynomially many random bits of our Monte-Carlo construction to obtain an encoding map $\text{Enc}_{\text{BSC}} : \{0,1\}^{R_{\text{BSC}} \cdot N_{\text{data}}} \to \{0,1\}^{N_{\text{data}}}$, and a decoding map $\text{Dec}_{\text{BSC}} : \{0,1\}^{N_{\text{data}}} \to \{0,1\}^{R_{\text{BSC}} \cdot N_{\text{data}}} \cup \{fail\}$, for $R_{\text{BSC}} = 1 - H(p_{\text{BSC}}) - \epsilon/3$. We can guarantee that with probability $1 - \frac{1}{10 \cdot N^c}$, the obtained code satisfies the properties listed in Theorem 5.7. Theorem 5.7 also provides a constant D, such that the encoding and decoding run in time N^D. We choose $c_\epsilon = c_H + c_s + D + c_0$.

Averaging Sampler: We use Theorem 3.6 to obtain a $(\frac{1}{n^{c_s}}, \frac{1}{n^{c_s}})$-sampler with distinct samples $\text{Samp} : \{0,1\}^{d_{\text{samp}}} \to [n]^{n_{\text{ctrl}}}$. We indeed meet the condition that the number of samples $n_{\text{ctrl}} = n^{0.1} \geq 2^{(\log n)^{0.1}}$. By Theorem 3.6 we have an explicit construction with seed length $d_{\text{samp}} = O(\log n) = O(\log N)$ (where the hidden constat depends on earlier choices of constants).

PRG against circuits: We will use Proposition 3.3 to obtain a PRG. Theorem 3.3 gives an explicit Monte-Carlo construction of PRGs, and we will use part of the polynomially many random bits of our Monte-Carlo construction to obtain a function $G : \{0,1\}^{d_{\text{PRG}}} \to \{0,1\}^{N^2}$. We can guarantee that with probability $1 - \frac{1}{10 \cdot N^c}$, the obtained function G is a $\frac{1}{N^{c_\epsilon}}$-PRG against circuits of size N^{c_ϵ}, with $d_{\text{PRG}} = O(\log N)$ (where the hidden constat depends on earlier choices of constants). We will sometimes view G as a function that outputs only N_{data} bits by truncating the output to length N_{data}.

Pseudorandomly chosen permutation: Let $\pi^G : \{0,1\}^{d_{\text{PRG}}} \times [N_{\text{data}}] \to [N_{\text{data}}]$ be the function defined in Definition 3.4.

Length of control string: Let $\ell' = \max(d_{\text{PRG}}, d_{\text{samp}})$. Let $\ell = 3 \cdot \ell'$ and let $c_k \geq 1$ be a sufficiently large constant so that $\ell \leq c_k \cdot \log n$. This means that we can w.l.o.g. have that $d_{\text{samp}} = d_{\text{PRG}} = \ell'$. This choice is made so that a "control string" (that will consist of three seeds for G, π^G and Samp) have length $\ell = c_k \cdot \log N$.

SS-non-malleable code: Having chosen c_ϵ and c_k, we can now continue choosing parameters in the application of Theorem 6.5. We can continue the application of Theorem 6.5 and conclude that by Theorem 6.5 there exist constants c_d, c_b such that we can obtain a code $\text{Enc}_{\text{ctrl}} : \{0,1\}^{c_k \cdot \log N} \times \{0,1\}^{c_d \cdot \log N} \to \{0,1\}^{c_b \cdot \log N}$ that satisfies the properties guaranteed in Theorem 6.5. In particular, we will choose the parameter v to be n_{ctrl}, and the function $\psi(s) = (\text{Samp}(s_{\text{samp}}), \pi^{-1}_{s_\pi}, G(s_{\text{PRG}}))$. Note that Theorem 6.5 only gives an explicit Monte-Carlo construction of the code $(\text{Enc}_{\text{ctrl}}, \text{Dec}_{\text{ctrl}})$. However, we are shooting for a Monte-Carlo construction and we will use part of the polynomially many random bits of our Monte-Carlo construction to obtain the code $(\text{Enc}_{\text{ctrl}}, \text{Dec}_{\text{ctrl}})$. We can guarantee that with probability $1 - N^{-(c_\gamma)} \geq 1 - \frac{1}{10 \cdot N^c}$, the obtained code satisfies the properties listed above. Finally, we set $d = c_d \cdot \log N$.

Figure 3: Encoding algorithm for stochastic code

Preperations: In this figure we use the parameters and ingredients of Figures 1 and Figure 2 to define the stochastic encoding map. The encoding map will encode messages of length $R_{\mathrm{BSC}} \cdot N_{\mathrm{data}}$, and will output words of length N. This gives that it will have rate $R = \frac{R_{\mathrm{BSC}} \cdot N_{\mathrm{data}}}{N}$. The stochastic encoding map and will use a seed of length $\ell + n_{\mathrm{ctrl}} \cdot d$.

Definition: We define $\mathrm{Enc} : \{0,1\}^{R \cdot N} \times \{0,1\}^{\ell + n_{\mathrm{ctrl}} \cdot d} \to \{0,1\}^N$ as follows:

Input:

- A message $m \in \{0,1\}^{R_{\mathrm{BSC}} \cdot N_{\mathrm{data}}}$.
- A "random coin" for the stochastic encoding that consists of: a string $s = (s_{\mathrm{samp}}, s_\pi, s_{\mathrm{PRG}})$ and $r_1, \ldots, r_{n_{\mathrm{ctrl}}} \in \{0,1\}^d$, where $s_{\mathrm{samp}}, s_\pi, s_{\mathrm{PRG}} \in \{0,1\}^{\ell'}$ so that $s \in \{0,1\}^\ell$,

Output: A codeword $c = \mathrm{Enc}(m; (s, r_1, \ldots, r_{n_{\mathrm{ctrl}}}))$ of length N.

Operation:

Determine control blocks: Apply $\mathrm{Samp}(s_{\mathrm{samp}})$ to generate *control* $= \{i_1, \ldots, i_{n_{\mathrm{ctrl}}}\} \subseteq [n]$. These blocks will be called "control blocks", and the remaining n_{data} blocks will be called "data blocks".

Prepare data part: We prepare a string c_{data} of length N_{data} as follows:

- Encode m by $x = \mathrm{Enc}_{\mathrm{BSC}}(m)$.
- Generate an N_{data} bit string y by reordering the N_{data} bits of the encoding using the (inverse of) the permutation $\pi_{s_\pi}(\cdot) = \pi^G(s_\pi, \cdot)$. More precisely, $y = \pi^{-1}_{s_\pi}(x) = \pi^{-1}_{s_\pi}(\mathrm{Enc}_{\mathrm{BSC}}(m))$.
- Mask y using PRG. That is, $c_{\mathrm{data}} = y \oplus G(s_{\mathrm{PRG}}) = \pi^{-1}_{s_\pi}(\mathrm{Enc}_{\mathrm{BSC}}(m)) \oplus G(s_{\mathrm{PRG}})$. (Here we truncate the output of G to length N_{data}).

Prepare control part: We prepare a string c_{ctrl} of length N_{ctrl} (which we view as n_{ctrl} blocks of length b) as follows:

- $(c_{\mathrm{ctrl}})_j = \mathrm{Enc}_{\mathrm{ctrl}}(s, r_j)$.

Merge data and control parts: We prepare the final output codeword $c \in \{0,1\}^N$ by merging c_{data} and c_{ctrl}. That is, $c = (c_{\mathrm{data}}, c_{\mathrm{ctrl}})^{\mathit{control}}$.

to different considerations, as channels are sufficiently strong to run decoding algorithms for some of the components.

The use of SS-non-malleability and evasiveness will dictate that we need to modify the "list-decoding part" of the construction. Let us point where our construction deviates from that of Guruswami and Smith [GS16].

Using SS-non-malleable codes: We use the SS-non-malleable codes of Theorem 6.5 as control codes. This will be crucial for discarding candidate control strings $\bar{S}_i$ that are correlated with S (as we have already explained in detail in Section 2). This use comes at a cost. We are forced to use control strings of length $\ell = O(\log n)$. This is much shorter than what was used by previous work [GS16, SS21a, KSS19, SS21b]. Consequently, we have to modify the construction in order to accomodate this requirement.

- We can no longer use polylog(n)-wise independent permutations, as sampling such a permutation requires $\omega(\log n)$ bits. We will instead use the pseudorandom generator G to sample a permutation that is pseudorandomly chosen, but does not have the information theoretic proper-

Figure 4: Decoding algorithm for stochastic code

Definition: We define $\mathrm{Dec} : \{0,1\}^N \to \{0,1\}^{R \cdot N} \cup \{fail\}$ as follows:

Input: A "received word" $\bar{v} \in \{0,1\}^N$.

Output: A message $\bar{m} \in \{0,1\}^{R \cdot N}$ or $fail$.

Operation of decoding algorithm: On input $\bar{v} \in \{0,1\}^N$:

Compute control candidates: For $i \in [n]$, let $\bar{s}_i = \mathrm{Dec}_{\mathrm{ctrl}}(\bar{v}_i)$. (Here $\bar{v}_i$ is the i'th block of $\bar{v}$). Let *candidates* $= \{\bar{s}_i : i \in [n]\}$.

Compute valid candidates: We say that $\bar{s}$ is ***successful***, if when computing the procedure *DecodeUsingCandidate*$(\bar{s})$ (that is defined below) we obtain $\bar{m}(\bar{s}) \neq fail$.

Output message: If there exists a single $s^* \in$ *candidates* that is successful, output $\bar{m}(s^*)$. Otherwise, output $fail$.

Internal procedure *DecodeUsingCandidate*: On input $\bar{s} \in \{0,1\}^\ell$ (which we think of as a candidate for the control string) this procedure works as follows:

Determine control blocks: Apply $\mathrm{Samp}(\bar{s}_{\mathrm{samp}})$ to generate $\overline{\mathit{control}} = \{\bar{i}_1, \ldots, \bar{i}_{n_{\mathrm{ctrl}}}\}$. Compute $\bar{v}_{\mathrm{data}} = \bar{v}_{\mathrm{data}}^{\overline{\mathit{control}}}$.

Unmask PRG: Compute $\bar{y} = \bar{v}_{\mathrm{data}} \oplus G(\bar{s}_{\mathrm{PRG}})$. (Here we truncate the output of G to length N_{data}).

Reverse permutation: Let $\bar{x}$ be the N_{data} bit string obtained by "undoing" the permutation. More precisely, let $\pi_{\bar{s}_\pi}(\cdot) = \pi^G(\bar{s}_\pi, \cdot)$, and let $\bar{x} = \pi_{\bar{s}_\pi}(\bar{y}) = \pi_{\bar{s}_\pi}(\bar{v}_{\mathrm{data}} \oplus G(\bar{s}_{\mathrm{PRG}}))$.

Decode data: Compute $\bar{m} = \mathrm{Dec}_{\mathrm{BSC}}(\bar{x})$.

output: We use $\bar{m}(\bar{s})$ to denote the answer $\bar{m}$ when *DecodeUsingCandidate* is applied on $\bar{s}$.

ties of polylog(n)-wise independent permutations. See discussion in Section 5.2. This creates some complications, as we will need to balance the running times/sizes of pseudorandom components carefully, so that each one is pseudorandom against the next one.

- We also need to use a sampler with improved parameters, and a tighter analysis.
- We are not able to use the control code as an inner code in a "bigger code" as was done in [GS16, SS21a]. This pushes us to change the number of control blocks n_{ctrl} to be much smaller than the number of control blocks used in earlier works, and requires more care.

Using codes for BSC_p that are evasive for $\mathrm{Ckt}_p^{n^c}$: We use the evasive codes of Theorem 5.7. This is crucial for discarding control strings $\bar{S}_i$ that are not correlated with S (as we have already explained in detail in Section 2). This use comes at a cost, as the decoding of this code is less efficient than codes that were used earlier, and we need to be careful and arrange that the pseudorandomness properties of G and $\mathrm{Enc}_{\mathrm{ctrl}}$ are against circuits which are sufficiently large to compute this decoding algorithm.

More specifically, the running time of the encoding and decoding algorithms $\mathrm{Enc}_{\mathrm{BSC}}, \mathrm{Dec}_{\mathrm{BSC}}$ is N^D, where D is not a universal constant, but rather a constant that depends on other parameters (See Figure 2). This will create complications, as we will want certain pseudorandomness properties to hold with respect to circuits of size N^D, and will have to make sure that we choose constants and parameters in an order that allows this.

7.3 Proof of Theorem 7.1

This section is devoted to proving Theorem 7.1, and show the correctness of the main construction.

The setup: Throughout the remainder of the section, we fix the setup of Theorem 7.1. Specifically, let $0 \leq p < \frac{1}{4}$, $c \geq 1$ be constants, and let $\epsilon > 0$ be a sufficiently small constant. We use these choices to set up the parameters and ingredients as explained in Figure 1 and Figure 2. The choices in Figure 1 allow us to choose a universal constant $c_0 > 1$ that we will choose later. Let N be sufficiently large. We set $\nu = \frac{1}{N^c}$ to be the required error parameter.

We are using three ingredients that are obtained by a Monte-Carlo construction. More specifically, as explained in Figure 2, we use a Monte-Carlo construction to choose $(\mathrm{Enc}_{\mathrm{BSC}}, \mathrm{Dec}_{\mathrm{BSC}})$, G and $(\mathrm{Enc}_{\mathrm{ctrl}}, \mathrm{Dec}_{\mathrm{ctrl}})$. All Monte-Carlo constructions are explicit and have Monte-Carlo error that is smaller than $\frac{1}{10 \cdot N^c}$. Therefore, by a union bound, we can assume that following their choices, the code $(\mathrm{Enc}_{\mathrm{BSC}}, \mathrm{Dec}_{\mathrm{BSC}})$, the function G and the code $(\mathrm{Enc}_{\mathrm{ctrl}}, \mathrm{Dec}_{\mathrm{ctrl}})$ satisfy the properties listed in Figure 2.

Let $\mathrm{Enc} : \{0,1\}^{RN} \times \{0,1\}^{\ell + n_{\mathrm{ctrl}} \cdot d} \to \{0,1\}^N$ and $\mathrm{Dec} : \{0,1\}^N \to \{0,1\}^{RN} \cup \{fail\}$ be the functions specified in Figures 3 and Figure 4 using the ingredients and parameter choices in Figure 1 and Figure 2. Overall, by construction, the algorithms $(\mathrm{Enc}, \mathrm{Dec})$ run in time polynomial in n.

Bounding The Rate. By Figure 3, the rate R of Enc is given by:

$$\begin{aligned} R &= \frac{R_{\mathrm{BSC}} \cdot N_{\mathrm{data}}}{N} \\ &= \frac{(1 - H(p_{\mathrm{BSC}}) - \frac{\epsilon}{3}) \cdot (1 - \frac{n_{\mathrm{ctrl}}}{n}) \cdot N}{N} \\ &\geq (1 - H(p) - \frac{\epsilon}{10} - \frac{\epsilon}{3}) \cdot (1 - \frac{\epsilon}{10}) \\ &\geq 1 - H(p) - \epsilon, \end{aligned}$$

where the third line follows because $n_{\mathrm{ctrl}} = n^{0.1}$, and using Equation (3) from Section 3, we have that:

$$H(p_{\mathrm{BSC}}) = H(p \cdot (1 + \frac{\epsilon}{10})) \leq H(p) + \frac{\epsilon}{10}.$$

Road map for arguing the correctness of decoding. The main part in proving Theorem 7.1 is showing that the decoding algorithm is correct. The remainder of this section is devoted to this proof, and in this section we give a roadmap of this proof.

The setup:

- Let $m \in \{0,1\}^{RN}$ be a message.
- Let $C : \{0,1\}^N \to \{0,1\}^N$ be a channel in $\mathrm{Ckt}_p^{N^c}$

We will keep these choices of m, C fixed throughout this section.

We need to show that w.h.p. the message m is decoded correctly when applying encoding, channel and decoding. We will refer to this experiment as the encoding/decoding experiment, and will denote it by $\mathrm{expr}^{\mathrm{ed}}(m, C)$. In this experiment, $S \in \{0,1\}^\ell$ and $R \in (\{0,1\}^d)^{n_{\mathrm{ctrl}}}$ are chosen uniformly at random. $Z = \mathrm{Enc}(m; (S, R))$ is the codeword, $E = C(Z)$ is the error pattern chosen by the channel, $\bar{V} = Z \oplus E$ is the received word given to the decoding, and $\bar{M} = \mathrm{Dec}(\bar{V})$ is the message returned by the decoding. We use the convention that capital letters denote the random variables associated with small letters used in the construction, and a complete specification of experiment $\mathrm{expr}^{\mathrm{ed}}(m, C)$ is given in Figure 5.

Figure 5: The encoding/decoding experiment $\mathrm{expr}^{\mathrm{ed}}(m, C)$.

Parameters: A message $m \in \{0,1\}^{RN}$ and a channel $C \in \mathrm{Ckt}_p^{n^c}$.

Encoding phase: Choose uniformly at random $S \in \{0,1\}^{\ell}$ and $R \in (\{0,1\}^d)^{n_{\mathrm{ctrl}}}$, and let $Z = \mathrm{Enc}(m, S, R)$. More specifically, divide S into three parts of length $\ell' = \ell/3$, so that $S = (S_{\mathrm{samp}}, S_{\mathrm{PRG}}, S_{\pi})$ and perform the following:

- $\mathit{CONTROL} = \{I_1 < \ldots < I_{n_{\mathrm{ctrl}}}\} = \mathrm{Samp}(S_{\mathrm{samp}})$.
- We denote the elements of $[n] \setminus \mathit{CONTROL}$ by $\{W_1, \ldots, W_{n_{\mathrm{data}}}\}$.
- $x = \mathrm{Enc}_{\mathrm{BSC}}(m)$
- $Y = \pi_{S_\pi}^{-1}(x)$.
- $Z \in \{0,1\}^N$ is defined as follows:
  - $Z_{\mathrm{data}}^{\mathit{CONTROL}} = Y \oplus G(S_{\mathrm{PRG}})$.
  - $Z_{\mathrm{ctrl}}^{\mathit{CONTROL}}$ is defined as follows: for every $j \in [n_{\mathrm{ctrl}}]$, $Z_{I_j} = \mathrm{Enc}_{\mathrm{ctrl}}(S, R_j)$.

Channel phase: Let $E = C(Z)$ and $\bar{V} = Z \oplus E$.

Decoding phase: Let $\bar{M} = \mathrm{Dec}(\bar{V})$. More specifically:

Compute candidates:

- For every $i \in [n]$, let $\bar{S}_i = \mathrm{Dec}_{\mathrm{ctrl}}(\bar{V}_i)$.
- Let $\mathit{CANDIDATES} = \left\{\bar{S}_i : i \in [n]\right\}$.

Decode using candidates: For every $\bar{s} \in \mathit{CANDIDATES}$, compute $\mathit{DecodeUsingCandidate}(\bar{s})$, more specifically:

- Let $\overline{\mathit{CONTROL}}(\bar{s}) = \mathrm{Samp}(\bar{s}_{\mathrm{samp}})$ and compute $\bar{V}_{\mathrm{data}}(\bar{s}) = \bar{V}(\bar{s})_{\mathrm{data}}^{\overline{\mathit{CONTROL}}}$.
- Let $\bar{Y}(\bar{s}) = \bar{V}_{\mathrm{data}}(\bar{s}) \oplus G(\bar{s}_{\mathrm{PRG}})$.
- Let $\bar{X}(\bar{s}) = \pi_{\bar{s}_\pi}(\bar{Y}(\bar{s}))$.
- Let $\bar{M}(\bar{s}) = \mathrm{Dec}_{\mathrm{BSC}}(\bar{X}(\bar{s}))$.

Compute valid candidates: For every $\bar{s} \in \mathit{CANDIDATES}$, determine whether $\bar{s}$ is successful, that is, if $\bar{M}(\bar{s}) \neq \mathit{fail}$.

Output message: If there exists a single $\bar{s} \in \mathit{CANDIDATES}$ that is successful, we denote it by S^* and the final output is $\bar{M} = \bar{M}(S^*)$, otherwise we set $S^* = \mathit{fail}$ and $\bar{M} = \mathit{fail}$.

In order to complete the proof of Theorem 7.1 we need to show that the probability that the decoded message $\bar{M}$ is equal to m is large. More precisely, that:

$$\Pr_{\mathrm{expr}^{\mathrm{ed}}(m,C)}[\bar{M} = m] \geq 1 - \nu. \tag{8}$$

Recall that in the experiment, every candidate control string $\bar{s} \in \mathit{CANDIDATES}$ is used to produce a candidate message $\bar{M}(\bar{s})$.

The correct control string is one of the candidates. We first claim that w.h.p. the correct control string S is in *CANDIDATES* and that when decoding using this candidate we obtain the correct message m. (Loosely speaking, the earlier work of [GS16, SS21a] that obtained list-decoding, stopped here, and outputted the list of messages $\left\{\bar{M}(\bar{s}) : \bar{s} \in \mathit{CANDIDATES}\right\}$). The next lemma is stating that this list indeed contains the original message m.

Lemma 7.3 (The correct control string is one of the candidates)**.**

$$\Pr_{\mathrm{expr}^{\mathrm{ed}}(m,C)}[S \in \textit{CANDIDATES and } \bar{M}(S) = m] \geq 1 - \nu/2.$$

Loosely speaking, this is supposed to follow by the correctness of the list-decoding algorithm of [GS16] (with modifications in [SS21a, KSS19]) which guarantees that the correct candidate control string appears in the list *CANDIDATES*, and that when decoding with this candidate, the original message m is obtained. We explain the technique of previous work [GS16, SS21a, KSS19, SS21b] and prove Lemma 7.3 in Section 7.3.3. The argument is very similar to that used in the aforementioned previous work, but one has to be more careful, because of the reasons explained in Section 7.2.

All incorrect candidates are unsuccessful. The main contribution of this paper is that we achieve *unique decoding*. That is, we will show that w.h.p. only the candidate S is successful. Meaning that our decoding algorithm has that w.h.p. $S^* = S$ (and we identify the correct candidate). This is formally stated in the next lemma.

Lemma 7.4 (Only the correct candidate survives)**.**

$$\Pr_{\mathrm{expr}^{\mathrm{ed}}(m,C)}[S^* = S] \geq 1 - \nu/2.$$

Together, Lemmata 7.3 and 7.4 imply that with probability at least $1 - \nu$, we have that $S^* = S$ and $\bar{M} = \bar{M}(S^*) = \bar{M}(S) = m$. This means that (8) holds, and the correct message is decoded with probability $1 - \nu$, concluding the proof of Theorem 7.1.

The argument for proving Lemma 7.4. We now explain how to prove Lemma 7.4. We will first use the SS-non-malleability of $\mathrm{Enc}_{\mathrm{ctrl}}$ to prove the following lemma:

Lemma 7.5 (Using SS-non-malleability)**.** *There exists a set $H \subseteq \{0,1\}^{c_k \cdot \log N}$ of size at most N^{c_H+1} such that:*

$$\Pr_{\mathrm{expr}^{\mathrm{ed}}(m,C)}[\exists i \in [n] : \bar{S}_i \notin H \cup \{S\} \cup \{fail\}] < N^{-(c_s-1)}.$$

The proof of Lemma 7.5 appears in Section 7.3.1, and implements the intuition regarding SS-non-malleability explained in Section 2.3.

Next we will use the evasiveness property of Dec to show that for every fixed $\bar{s} \in \{0,1\}^{c_k \cdot \log N}$, the probability that the procedure *DecodeUsingCandidate*$(\bar{s})$ will not fail is small, where the probability is in $\mathrm{expr}^{\mathrm{ed}}(m, C)$.

Lemma 7.6 (Using evasiveness)**.** *For every fixed $\bar{s} \in \{0,1\}^{c_k \cdot \log N}$,*

$$\Pr_{\mathrm{expr}^{\mathrm{ed}}(m,C)}[\textit{DecodeUsingCandidate}(\bar{s}) \neq fail] \leq N^{-(c_H+c+c_0)}.$$

The proof of Lemma 7.6 appears in Section 7.3.2 and implements the intuition regarding evasiveness explained in Section 2.3.

Putting the two lemmas together, we conclude that it is unlikely that there exists $i \in [n]$ such that $\bar{S}_i$ is successful, and $\bar{S}_i \neq S$. This is done fortmally in the next claim.

Claim 7.7.

$$\Pr_{\mathrm{expr}^{\mathrm{ed}}(m,C)}[\exists i \in [n] : \textit{DecodeUsingCandidate}(\bar{S}_i) \neq fail,\ \textit{and}\ \bar{S}_i \neq S] < \nu/2.$$

Proof. Let:

$$P = \Pr_{\mathrm{expr}^{\mathrm{ed}}(m,C)}[\exists i \in [n] : \textit{DecodeUsingCandidate}(\bar{S}_i) \neq fail \text{ and } \bar{S}_i \neq S].$$

We have that:

$$\begin{aligned} P &\leq \Pr_{\mathrm{expr}^{\mathrm{ed}}(m,C)}[\exists i \in [n] : \bar{S}_i \notin H \cup \{S\} \cup \{fail\}] + \Pr_{\mathrm{expr}^{\mathrm{ed}}(m,C)}[\exists \bar{s} \in H : \textit{DecodeUsingCandidate}(\bar{s}) \neq fail] \\ &\leq N^{-(c_s-1)} + \sum_{\bar{s} \in H} \Pr_{\mathrm{expr}^{\mathrm{ed}}(m,C)}[\textit{DecodeUsingCandidate}(\bar{s}) \neq fail] \\ &\leq N^{-(c_s-1)} + N^{c_H+1} \cdot N^{-(c_H+c+c_0)} \\ &\leq \frac{\nu}{2}, \end{aligned}$$

where the second inequality is by Lemma 7.5, the third inequality is by Lemma 7.6, and the final inequality follows because $\nu = N^{-c}$, and we have chosen $c_s = c + c_0$, and we can choose c_0 to be sufficiently large so that the inequality holds. □

We are now ready to prove Lemma 7.4.

Proof. (of Lemma 7.4) By Lemma 7.3 we have that:

$$\Pr_{\mathrm{expr}^{\mathrm{ed}}(m,C)}[\exists i : \bar{S}_i = S \text{ and } \textit{DecodeUsingCandidate}(\bar{S}_i) = m \neq fail] \geq 1 - \nu/2.$$

Combining this with Claim 7.7 we have that except with probability ν, we have that in $\mathrm{expr}^{\mathrm{ed}}(m, C)$ the two events below occur:

- $\big\{\exists i \in [n] : \bar{S}_i = S$ and $\bar{S}_i$ is successful$\big\}$.
- $\big\{\forall i \in [n] :$ either $\bar{S}_i = S$ or $\bar{S}_i$ is not suceessful$\big\}$.

When these two events occur, we have that there is a unique $\bar{s} \in$ *CANDIDATES* that is successful, and $S^* = \bar{s} = S$. □

This concludes the proof of Theorem 7.1. It remains to prove the lemmas that appeared inside the proof, and this is done in the next sections.

7.3.1 Using SS-non-malleability: Proof of Lemma 7.5

We need to define a set $H \subseteq \{0,1\}^{c_k \cdot \log N}$ of size N^{c_H+1}. We will use the channel circuit C to define n adversaries $C^1, \ldots, C^n$ for the SS-non-malleability property of $\mathrm{Enc}_{\mathrm{ctrl}}$. Recall, that any such adversary C^i to the SS-non-malleability is expecting to receive input of the form:

$$\psi(S), \mathrm{Enc}_{\mathrm{ctrl}}(S, R_1), \ldots, \mathrm{Enc}_{\mathrm{ctrl}}(S, R_v),$$

for a uniformly chosen $S \leftarrow \{0,1\}^{c_k \cdot \log N}$. Intuitively, such an adversary is trying to produce a string that will be decoded by $\mathrm{Dec}_{\mathrm{ctrl}}$ to a string $S' \neq S$ that is "correlated" with S. In our setting, we have chosen $v = n_{\mathrm{ctrl}}$ and

$$\psi(s) = (\mathrm{Samp}(s_{\mathrm{samp}}), \pi_{s_\pi}^{-1}, G(s_{\mathrm{PRG}})).$$

We now define the adversary C^i:

Definition 7.8. *For every $i \in [n]$, we define a function C^i as follows:*

Input: *The adversary C^i receives:*

- n_{ctrl} *distinct indices $\{i_1, \dots, i_{n_{\mathrm{ctrl}}}\} \subseteq [n]$. (This is supposed to be $\mathrm{Samp}(S_{\mathrm{samp}})$).*
- N_{data} *distinct indices $\{j_1, \dots, j_{N_{\mathrm{data}}}\} \subseteq [N_{\mathrm{data}}]$. (This is supposed to be a description of the permutation $\pi_{S_\pi}^{-1}$, which is given by $j_i = \pi_{s_\pi}^{-1}(i)$).*
- *A string $g \in \{0,1\}^{N_{\mathrm{data}}}$. (This is supposed to be $G(S_{\mathrm{PRG}})$).*
- $v = n_{\mathrm{ctrl}}$ *strings $e_1, \dots, e_v$. (These are supposed to be $\mathrm{Enc}_{\mathrm{ctrl}}(S, R_1), \dots, \mathrm{Enc}_{\mathrm{ctrl}}(S, R_v)$).*

Operation: *The adversary C^i is hardwired with the fixed message m, the fixed string $\mathrm{Enc}_{\mathrm{BSC}}(m)$, and the circuit C. It will act as follows:*

Simulate an encoding: *Note that if the inputs to C^i are as intended, then C^i can prepare $z = \mathrm{Enc}(m; (S, R_1, \dots, R_n))$ by following the same procedure that* Enc *uses. (A key observation is that C^i does not need to compute $\mathrm{Enc}_{\mathrm{ctrl}}$, π_G,* Samp, *G or $\mathrm{Enc}_{\mathrm{BSC}}$. Therefore, simulating the encoding can be computed by a circuit of size N^{c_1} for a universal constant c_1).*

Simulate the channel: *The adversary will compute $e = C(z)$ and $\bar{v} = z \oplus e$. (This is what the channel C does on the encoding, and if the inputs to C^i are as intended, then this is a simulation of $\bar{V}$ from $\mathrm{expr}^{\mathrm{ed}}(m, C)$).*

Output the i'th block: *Output $\bar{v}_i$. (If the inputs to C^i are as intended, then this is a simulation of $\bar{V}_i$ from $\mathrm{expr}^{\mathrm{ed}}(m, C)$).*

We record the following obvious properties of the adversary C^i:

Claim 7.9. *There exists a universal constant c_1 such that For every $i \in [n]$, C^i is a circuit of size $N^c + N^{c_1}$, and furthermore:*

- *Let $S, R_1, \dots, R_v$ be chosen as in $\mathrm{expr}^{\mathrm{ed}}(m, C)$. For every $i \in [n]$, if C^i receives the input:*

$$\psi(S), \mathrm{Enc}_{\mathrm{ctrl}}(S, R_1), \dots, \mathrm{Enc}_{\mathrm{ctrl}}(S, R_v),$$

then it outputs $\bar{V}_i$.

- *In particular, it follows that the decoded control block in the decoding process in $\mathrm{expr}^{\mathrm{ed}}(m, C)$ is:*

$$\bar{S}_i = \mathrm{Dec}_{\mathrm{ctrl}}(C^i(\psi(S), \mathrm{Enc}_{\mathrm{ctrl}}(S, R_1), \dots, \mathrm{Enc}_{\mathrm{ctrl}}(S, R_v))).$$

Recall that we have chosen $c_s = c + c_0$. We will choose the constant c_0 to be sufficiently large so that the size of C^i is bounded by N^{c_s}. This means that using the SS-non-malleability property of the code $\mathrm{Enc}_{\mathrm{ctrl}}$ we have that for every $i \in [n]$, there exists a set $H_{C^i} \subseteq \{0,1\}^{c_k \cdot \log N}$ with $|H_{C^i}| \leq N^{c_H}$, such that

$$\Pr_{\substack{S \leftarrow \{0,1\}^{c_k \cdot \log N} \\ R_1, \dots, R_v \leftarrow \{0,1\}^{c_d \cdot \log N}}} [\mathrm{Dec}(C^i(\psi(S), \mathrm{Enc}_{\mathrm{ctrl}}(S, R_1), \dots, \mathrm{Enc}_{\mathrm{ctrl}}(S, R_v))) \notin H_{C^i} \cup \{S\} \cup \{fail\}] < N^{-c_\rho},$$

and recall that we have chosen $c_\rho = c_s$. By Claim 7.9, we get that for every $i \in [n]$:

$$\Pr_{\mathrm{expr}^{\mathrm{ed}}(m,C)}[\bar{S}_i \notin H_{C^i} \cup \{S\} \cup \{fail\}] < N^{-c_s}.$$

We define:

$$H = \bigcup_{i \in [n]} H_{C^i}.$$

It follows that $|H| \leq n \cdot N^{c_H} \leq N^{c_H+1}$, and by a union bound over the $n \leq N$ choices for $i \in [n]$, we get that:

$$\begin{aligned}\Pr_{\mathrm{expr}^{\mathrm{ed}}(m,C)}[\exists i \in [n] : \bar{S}_i \notin H \cup \{S\} \cup \{fail\}] &\leq \sum_{i \in [n]} \Pr_{\mathrm{expr}^{\mathrm{ed}}(m,C)}[\bar{S}_i \notin H_{C^i} \cup \{S\} \cup \{fail\}] \\ &< n \cdot N^{-c_s} \\ &\leq N^{-(c_s-1)},\end{aligned}$$

and Lemma 7.5 follows.

7.3.2 Using evasiveness: Proof of Lemma 7.6

Fix some $\bar{s} \in \{0,1\}^{c_k \cdot \log N}$. We are interested in the operation of *DecodeUsingCandidate*$(\bar{s})$. The output of *DecodeUsingCandidate*$(\bar{s})$ can be viewed as the following function of the "codeword" z as follows:

Definition 7.10 (The function $C_{\bar{s}}$). *We define a function $C_{\bar{s}} : \{0,1\}^N \to \{0,1\}$. On input $z \in \{0,1\}^N$, $C_{\bar{s}}(z)$ acts as follows:*

Simulate channel: *Compute $e = C(z)$, and $\bar{v} = z \oplus C(z)$.*

Simulate *DecodeUsingCandidate*$(\bar{s})$: *Simulate the operation of* DecodeUsingCandidate$(\bar{s})$ *on $\bar{v}$. More precisely, perform the operations "Determine control blocks", "Unmask PRG" "Reverese Permutation" and "Decode data", precisely as described in the description of* DecodeUsingCandidate *that appears in Figure 4. If the procedure* DecodeUsingCandidate *outputs $fail$, then output 1, and otherwise output zero.*

This definition is made so that:

$$\Pr_{\mathrm{expr}^{\mathrm{ed}}(m,C)}[\textit{DecodeUsingCandidate}(\bar{s}) = fail] = \Pr_{\mathrm{expr}^{\mathrm{ed}}(m,C)}[C_{\bar{s}}(Z) = 1]. \tag{9}$$

An important observation is that $C_{\bar{s}}$ does not need to compute $\mathrm{Dec}_{\mathrm{ctrl}}$. We also point out that as $\bar{s}$ is a fixed string, an implementation of $C_{\bar{s}}$ by a circuit, can be hardwired with $\mathrm{Samp}(\bar{s}_{\mathrm{samp}})$, $G(\bar{s}_{\mathrm{PRG}})$ and $\pi_{\bar{s}_\pi}$. This means that $C_{\bar{s}}$ does not need to compute the functions Samp and G. The only nontrivial computation performed by $C_{\bar{s}}$ is applying C and applying the algorithm $\mathrm{Dec}_{\mathrm{BSC}}$. The former is computable by a size N^c circuit, and the latter is computable by a circuit of size N^D (where D was chosen in figure 2). We have made sure to choose $c_\epsilon \geq D + c + c_0$, where c_0 is a sufficiently large constant. We can conclude that by choosing c_0 to be sufficiently large:

Claim 7.11. *$C_{\bar{s}}$ is a circuit of size at most $N^{c_\epsilon - 3}$.*

We will also need the following claim:

Lemma 7.12 (Pseudorandomness of encoding). *For every message $m \in \{0,1\}^{RN}$, sampler seed $s_{\text{samp}} \in \{0,1\}^{\ell'}$ and permutation seed $s_\pi \in \{0,1\}^{\ell'}$, let $V = \text{Enc}(m; (s_\pi, s_{samp}, S_{PRG}, R_1, \cdots, R_{n_{ctrl}}))$ be a random variable (defined over the probability space where $S_{PRG}, R_1, \cdots, R_{n_{ctrl}}$ are chosen uniformly and independently). V is $N^{-(c_\epsilon - 3)}$-pseudorandom for circuits of size $N^{c_\epsilon - 3}$.*

Proof. We assume for contradiction that there exists a circuit D of size $N^{c_\epsilon - 3}$

$$|\Pr[D(V) = 1] - \Pr[D(U_N) = 1]| > N^{-(c_\epsilon - 3)}.$$

We will prove that one of the following holds:

- There exists a size N^{c_ϵ} circuit $C : \{0,1\}^{N_{\text{data}}} \to \{0,1\}$ such that:

$$|\Pr[C(G(S_{\text{PRG}})) = 1] - \Pr[C(U_{N_{\text{data}}}) = 1]| > N^{-c_\epsilon}.$$

- There exists $z' \in \{0,1\}^\ell$ and a size N^{c_ϵ} circuit $C : \{0,1\}^b \to \{0,1\}$, such that:

$$|\Pr[C(\text{Enc}_{\text{ctrl}}(z', U_d)) = 1] - \Pr[C(U_b) = 1]| > N^{-c_\epsilon}.$$

This suffices, as the lemma follows by the pseudorandomness properties of the G and Enc_{ctrl}.

We now prove that one of the two items hold. We partition V into $V = (V_{\text{data}}, V_{\text{ctrl}})^{\text{Samp}(s_{\text{samp}})}$. We have that D distinguishes $V = (V_{\text{data}}, V_{\text{ctrl}})$ from $U_N = (U_{\text{data}}, U_{\text{ctrl}})$ with probability greater than $N^{-(c_\epsilon - 3)}$, we do a hybrid argument and consider the hybrid distribution $H = (V_{\text{data}}, U_{\text{ctrl}})$. It follows that:

- Either D distinguishes H from U_N with probability $N^{-(c_\epsilon - 3)}/2$,
- or, D distinguishes H from V with probability $N^{-(c_\epsilon - 3)}/2$.

In the first case, we have that V_{data} and U_{ctrl} are independent, and an averaging argument gives that there exists a fixed value v'_{ctrl}, such that D distinguishes $(U_{\text{data}}, v'_{\text{ctrl}})$ from $(V_{\text{data}}, v'_{\text{ctrl}})$ with probability $N^{-(c_\epsilon - 3)}/2$. This gives that there exists a size $N^{c_\epsilon - 3}$ circuit $C : \{0,1\}^{N_{\text{data}}} \to \{0,1\}$ such that the first item holds.

In the second case, we have that m and s_π are fixed and therefore the string $y = \pi_{s_\pi}^{-1}(\text{Enc}_{\text{BSC}}(m))$ used in the encoding algorithm is also fixed. The encoding algorithm computes the data part by xoring y with $G(S_{\text{PRG}})$ and therefore $V_{\text{data}} = G(S_{\text{PRG}}) \oplus y$. By an averaging argument, there exists a fixing s'_{PRG} such that D distinguishes $((G(s'_{\text{PRG}}) \oplus y), U_{\text{ctrl}})$ from $(((G(s'_{\text{PRG}}) \oplus y), V_{\text{ctrl}})|S_{\text{PRG}} = s'_{\text{PRG}})$ with probability $N^{-(c_\epsilon - 3)}/2$.

We get that there exists a size $N^{c_\epsilon - 3}$ circuit $C' : \{0,1\}^{n_{\text{ctrl}} \cdot d} \to \{0,1\}$ such that D' distinguishes U_{ctrl} from $V'_{\text{ctrl}} = (V_{\text{ctrl}}|S_{PRG} = s'_{\text{PRG}})$.

Recall that the encoding procedure prepares the j'th block of the control part c_{ctrl}, by $\text{Enc}_{\text{ctrl}}(s, r_j)$.

Having fixed $S_{\text{PRG}} = s'_{\text{PRG}}$ the only random variables that remain unfixed in V'_{ctrl} are $R_1, \ldots, R_{n_{\text{ctrl}}}$. This means that there exists $s' \in \{0,1\}^\ell$ such that $(V'_{\text{ctrl}})_j = \text{Enc}_{\text{ctrl}}(s', R_j)$ and in particular, the n_{ctrl} blocks are independent. We have that D' distinguishes V'_{ctrl} from U_{ctrl} with probability $N^{-(c_\epsilon - 3)}/2$, and by a standard hybrid argument, there exists a circuit C of size $N^{c_\epsilon - 3}$ such that C distinguishes $(V'_{\text{ctrl}})_j = \text{Enc}_{\text{ctrl}}(s', R_j)$ from uniform with probability $\frac{N^{-(c_\epsilon - 3)}}{2 \cdot n_{\text{ctrl}}} \geq N^{-c_\epsilon}$ and the second item follows. □

In particular the encoding Z in $\mathrm{expr}^{\mathrm{ed}}(m, C)$ in which S_{samp}, S_π are chosen uniformly, is also pseudorandom. This implies that:

$$|\Pr_{\mathrm{expr}^{\mathrm{ed}}(m,C)}[C_{\bar{s}}(Z) = 1] - \Pr[C_{\bar{s}}(U_N) = 1]| \leq N^{-(c_\epsilon - 3)}. \tag{10}$$

The input string $z \in \{0,1\}^n$ to $C_{\bar{s}}$ can be divided into its control and data parts, using $I = \mathrm{Samp}(\bar{s}_{\mathrm{samp}})$. Recall that we use z^I_{ctrl} and z^I_{data} for these two parts, and omit I when it is clear from the context. We will now consider a version of $C_{\bar{s}}$ in which the control part is already fixed, and only the data part is given as input.

Definition 7.13. *Let $I = \mathrm{Samp}(\bar{s}_{\mathrm{samp}})$. For every $w \in \{0,1\}^{N_{\mathrm{ctrl}}}$, we define $C_{\bar{s},w} : \{0,1\}^{N_{\mathrm{data}}} \to \{0,1\}$ by:*

$$C_{\bar{s},w}(x) = C((w, x)^I).$$

This definition is made so that for every $y \in \{0,1\}^N$, $C_{\bar{s}}(y) = C_{\bar{s},y_{\mathrm{ctrl}}}(y_{\mathrm{data}})$. Note that for $z \in \{0,1\}^{N_{\mathrm{data}}}$, we can view the operation of $C_{\bar{s},w}(z)$ as follows:

- In the “simulate channel” step, z is modified by a size N^c channel

  $$C_w(z) := C((w, z)^I)_{\mathrm{data}}$$

  to produce $\bar{v}_{\mathrm{data}}$.

  Note that C_w is a channel that induces pN errors, and the size of C_w is bounded by $N^c + N^{c_1}$ for a universal constant c_1. Therefore, as $c_s = c + c_0$ for a a sufficiently large constant, we have that the size of C_w is bounded by c_s.

- in the “simulate *DecodeUsingCandidate*($\bar{s}$)” step, we perform the following:
  - Unmask the PRG: $\bar{y} = \bar{v}_{\mathrm{data}} \oplus g$, for the fixed string $g = G(\bar{s}_{\mathrm{PRG}})$.
  - Reverse the permutation: $\bar{x} = \pi(\bar{y})$ for the fixed permutation $\pi = \pi_{\bar{s}_\pi}$.
  - Decode the BSC code: $\bar{m} = \mathrm{Dec}_{\mathrm{BSC}}(\bar{x})$.

This intuitively means that in the experiment $C_{\bar{s},w}(Z)$, where $Z \leftarrow U_{N_{\mathrm{data}}}$, we come very close to reproducing the evasiveness experiment of Section 5. More precisely, we can imagine a version of the evasiveness experiment as follows:

- A channel C_w (defined above) which is a circuit of size N^{c_s} that induces at most pN errors, is applied on $Z \leftarrow U_{N_{\mathrm{data}}}$ to produce the corrupted word $\bar{V}_{\mathrm{data}} = Z \oplus C_w(Z)$.
- The decoding algorithm $\mathrm{Dec}_{\mathrm{BSC}}$ is applied on the corrupted word, and we check whether $\mathrm{Dec}_{\mathrm{BSC}}$ outputs $fail$.

The only difference between the latter experiment and our scenario, is that in our scenario there are two intermediate steps between the two operations in the evasiveness experiment: The corrupted word $\bar{V}_{\mathrm{data}}$ is xored with a fixed string g, and permuted using a fixed permutation π. More precisely, $\mathrm{Dec}_{\mathrm{BSC}}$ is applied on $\pi(\bar{V}_{\mathrm{data}} \oplus g)$ rather than on $\bar{V}_{\mathrm{data}}$.

Intuitively, this difference is immaterial, because the two actions considered above (xoring with a fixed string, and permuting with a fixed distribution) preserve the uniform distribution $U_{N_{\mathrm{data}}}$, and therefore can be ignored. A formal way to see this is that these two operations can be “hardwired” into the circuit C_w as follows:

- We define $Z' = \pi(Z \oplus g)$. Note that for $Z \leftarrow U_{N_{\text{data}}}$, Z' is also distributed uniformly over $\{0,1\}^{N_{\text{data}}}$.
- Fix some $w \in \{0,1\}^{N_{\text{ctrl}}}$, and let us denote C_w by D, in order to avoid clutter, and not carry w in the notation. We define $D' : \{0,1\}^{N_{\text{data}}} \to \{0,1\}^{N_{\text{data}}}$ as follows:

$$D'(z') = \pi(D(\pi^{-1}(z') \oplus g)).$$

We are assuming that $C \in \text{Ckt}_p^{N^c}$, which gives that D outputs strings with Hamming weight at most pN. Note that by definition, D' also outputs strings with Hamming weight at most pN. The definition of Z' and D' is made so that:

$$\begin{aligned} Z' \oplus D'(Z') &= \pi(Z \oplus g) \oplus \pi(D(\pi^{-1}(\pi(Z \oplus g)) \oplus g)) \\ &= \pi(Z \oplus g) \oplus \pi(D(Z \oplus g \oplus g)) \\ &= \pi(Z \oplus g) \oplus \pi(D(Z)) \\ &= \pi(Z \oplus D(Z) \oplus g) \end{aligned}$$

In other words, we can imagine that the uniform random variable chosen in the evasiveness experiment is Z' (which is also uniformly distributed over $\{0,1\}^{N_{\text{data}}}$), and that the channel is D'. We have that D' is a circuit of size N^{c_s} which induces at most pN errors. Let us express pN as $p' \cdot N_{\text{data}}$ (because the fraction of errors is measured as a precentage of the length).

$$pN = (p \cdot \frac{N}{N_{\text{data}}}) \cdot N_{\text{data}} = p \cdot (1 + \frac{n_{\text{ctrl}}}{n}) \cdot N_{\text{data}} \le p \cdot (1 + \frac{\epsilon}{10}) N_{\text{data}} = p_{BSC} \cdot N_{\text{data}}.$$

It follows that $D' \in \text{Ckt}_{p_{\text{BSC}}}^{N^{c_s}}$ is a circuit that cannot break the evasiveness of $(\text{Enc}_{\text{BSC}}, \text{Dec}_{\text{BSC}})$.

The distribution $\bar{V}' = Z' \oplus D'(Z')$ is precisely the distribution $\pi(Z \oplus D(Z) \oplus g)$ that is obtained in the experiment $D(Z)$.

Therefore, by the choices made in Figure 2, using Theorem 5.7, that guarantees the evasiveness of $(\text{Enc}_{\text{BSC}}, \text{Dec}_{\text{BSC}})$ against $\text{Ckt}_{p_{\text{BSC}}}^{N^{c_s}}$, we can conclude that for every choice of $w \in \{0,1\}^{N_{\text{ctrl}}}$.

$$\begin{aligned} \Pr_{Z \leftarrow U_{N_{\text{data}}}}[C_{\bar{s},w}(Z) = 1] &= \Pr_{Z \leftarrow U_{N_{\text{data}}}}[\text{Dec}_{\text{BSC}}(\pi(Z \oplus D(Z) \oplus g)) \neq \mathit{fail}] \\ &\le \Pr_{Z \leftarrow U_{N_{\text{data}}}, Z' = \pi(Z \oplus g)}[\text{Dec}_{\text{BSC}}(Z' \oplus D'(Z')) \neq \mathit{fail}] \\ &\le N_{\text{data}}^{-(c_s + c_H + c_0)}, \end{aligned}$$

where the first inequality follows by the definition of $C_{\bar{s},w}$. The second inequality follows by the previous discussion, and the third inequality follows by Theorem 5.7, and by the choices made in Figure 2. We are finally ready to prove Lemma 7.6.

$$\begin{aligned} \Pr_{\text{expr}^{\text{ed}}(m,C)}[\mathit{DecodeUsingCandidate}(\bar{s}) = \mathit{fail}] &= \Pr_{\text{expr}^{\text{ed}}(m,C)}[C_{\bar{s}}(Z) = 1] \\ &\le \Pr[C_{\bar{s}}(U_N) = 1] + N^{-(c_\epsilon - 3)} \\ &= \Pr_{W \leftarrow U_{N_{\text{ctrl}}}, Z \leftarrow U_{N_{\text{data}}}}[C_{\bar{s},W}(Z) = 1] + N^{-(c_\epsilon - 3)} \\ &\le \max_{w \leftarrow \{0,1\}^{N_{\text{ctrl}}}} \left(\Pr_{Z \leftarrow U_{N_{\text{data}}}}[C_{\bar{s},w}(Z) = 1] \right) + N^{-(c_\epsilon - 3)} \\ &\le N_{\text{data}}^{-(c_s + c_H + c_0)} + N^{-(c_\epsilon - 3)} \\ &\le N^{-(c_H + c + c_0)}, \end{aligned}$$

for large enough N, where the first line follows by (9), the second line follows by (10), the third line follows by the definition of $C_{\bar{s},w}$, and the last line by the choice of c_ϵ, and because $c_s > c$, using that $N_{\text{data}} \geq N/2$.

7.3.3 The correct control string is one of the candidates: Proof of Lemma 7.3

The proof of Lemma 7.3 is very similar to the corresponding proofs in earlier works [GS16, SS21a, KSS19] that achieved list-decoding, rather than unique deocding. We cannot directly cite these proofs, as our construction has some differences compared to the aforementioned previous work (and these differences are necessary to make the unique-decoding go through). Nevertheless, the argument that we present here is essentially identical to previous work, modulu these modifications.

Definition 7.14 (Milestone function)**.** *We define a function $A : \{0,1\}^{\ell'} \times \{0,1\}^{\ell'} \times \{0,1\}^N \to \{0,1\}$ as follows: On inputs $s_{\text{samp}}, s_\pi \in \{0,1\}^{\ell'}$ and $e \in \{0,1\}^n$, $A(s_{\text{samp}}, s_\pi, e)$ outputs one iff there exists $i \in I = \text{Samp}(s_{\text{samp}})$ such that wt$(e_i) \leq p + \frac{1}{N^{c_s}}$, and*

$$\text{Dec}_{\text{BSC}}(\text{Enc}_{\text{BSC}}(m) \oplus \pi_{s_\pi}(e^I_{\text{data}})) = m.$$

We define $A_{s_{\text{samp}},s_\pi}(e) = A(s_{\text{samp}}, s_\pi, e)$.

A is designed so that in the experiment $\text{expr}^{\text{ed}}(m, C)$:

- $A(S_{\text{samp}}, S_\pi, C(Z))$ checks whether there exists a control block $i \in I = \text{Samp}(S_{\text{samp}})$ on which the i'th block $E_i = C(Z)_i$ of the error vector $E = C(Z)$, has low weight. If this happens then:
$$\bar{S}_i = \text{Dec}_{\text{ctrl}}(\bar{V}_i) = \text{Dec}_{\text{ctrl}}(Z_i \oplus C(Z)_i) = \text{Dec}_{\text{ctrl}}(\text{Enc}_{\text{ctrl}}(S, R_i) \oplus E_i),$$
will be correctly decoded to S. (This is because Dec_{ctrl} can decode from $\frac{\frac{1}{2}-\beta}{2} \geq p + \frac{1}{N^{c_s}}$ errors).
- Furthermore, that when continuing the decoding in $\text{expr}^{\text{ed}}(m, C)$ using the candidate $\bar{S}_i = S$, the original message m will be decoded. (Note that the noise pattern that is applied to Enc_{BSC} for fixed e, during the decoding by Dec_{BSC} is indeed $\pi_{s_\pi}(e)$).

By this discussion we have that:

$$\Pr_{\text{expr}^{\text{ed}}(m,C)}[S \in \textit{CANDIDATES} \text{ and } \bar{M}(S) = m] \geq \Pr_{\text{expr}^{\text{ed}}(m,C)}[A(S_{\text{samp}}, S_\pi, C(Z)) = 1]. \tag{11}$$

On the other hand, the construction of Enc and Dec was set up so that, decoding is successful against additive channels. This is stated next.

Lemma 7.15. *For every $e \in \{0,1\}^N$ with wt$(e) \leq p$,*

$$\Pr_{S_{\text{samp}},S_\pi \leftarrow \{0,1\}^{\ell'}}[A(S_{\text{samp}}, S_\pi, e) = 1] \geq 1 - N^{-c_s} - N^{-(c_\epsilon - 1)}.$$

Proof. We will now analyze what happens in this scenario, where $e \in \{0,1\}^N$ is fixed, and S_{samp}, S_π are uniform.

To prove the lemma, we make the following definition. For every $i \in [n]$, let $f(i) = wt(e_i)$. By the properties of the Sampler, we have that when choosing $S_{\text{samp}} \leftarrow \{0,1\}^{\ell'}$, and taking $\{i_1, \ldots, i_{n_{\text{ctrl}}}\} = \text{Samp}(S_{\text{samp}})$, the probability that

$$\frac{1}{n_{\text{ctrl}}} \cdot \sum_{j \in [n_{\text{ctrl}}]} f(i_j) > p + \frac{1}{N^{c_s}}$$

is at most $\frac{1}{N^{c_s}}$. (This follows by the choice of parameters of the Sampler Samp in Figure 1 using Theorem 3.6). When this event occurs, there must exist j such that $f(i_j) = \mathit{wt}(e_{i_j}) \leq p + \frac{1}{N^{c_s}}$, and the first condition in the definition of A is met.

We now show that the second condition is met w.h.p. over the choice of $S_\pi \leftarrow \{0,1\}^{\ell'}$. This is essentially the scenario for which $\mathrm{Dec}_{\mathrm{BSC}}$ was designed (see Theorem 5.5 and Figure 1) of decoding from a pseudorandomly chosen permutation. More precisely, the weight of e is pN, and even if all the pN ones in e end up in e^I_{data} then the fraction of ones in the data part of e is at most:

$$\frac{pN}{N_{\text{data}}} = p \cdot \frac{1}{1 - \frac{n_{\text{ctrl}}}{n}} \leq p \cdot (1 + \frac{\epsilon}{10}) = p_{\mathrm{BSC}},$$

where the last inequality holds for large enough N, by our choice that $n_{\text{ctrl}} = n^{0.1} \geq N^{0.05}$. In Figure 2, we have set up the code $(\mathrm{Enc}_{\mathrm{BSC}}, \mathrm{Dec}_{\mathrm{BSC}})$ to decode from $\mathrm{Perm}_{p_{\mathrm{BSC}}}^{\mathrm{UniPerm}_{N_{\text{data}}}}$, and because π_{S_π} is pseudorandomly chosen (with error N^{-c_ϵ}), rather than uniformly chosen, we get decoding error

$$2^{-\Omega(n^{0.09})} + N^{-c_\epsilon} \leq N^{-(c_\epsilon - 1)},$$

for sufficiently large N. Here, we make use of the fact that G fools circuits of size N^{c_ϵ}, and we have chosen $c_\epsilon \geq D + c_0$, This gives that for every fixing s_{Samp} of S_{Samp}, $\mathrm{Dec}_{\mathrm{BSC}}$ (which is of size N^D) does not distinguish between uniform and pseudorandomly chosen permutations.

Overall, by a union bound, we have that:

$$\Pr_{Z_{\text{samp}}, Z_\pi \leftarrow \{0,1\}^{\ell'}} [A(S_{\text{samp}}, S_\pi, e) = 1] \geq 1 - N^{-c_s} - N^{-(c_\epsilon - 1)},$$

as required. □

We would like to argue that Lemma 7.15 implies a bound in the scenario considered in (11). For this purpose, we observe that for every fixed values $s_{\text{samp}}, s_\pi \in \{0,1\}^{\ell'}$:

- The function $T(e) = A_{s_{\text{samp}}, s_\pi}(C(e))$ can be computed by a circuit of size $N^{c_\epsilon - 3}$. This is because for fixed s_{samp}, s_π, such a circuit can be hardwired with $\mathrm{Samp}(s_{\text{samp}})$ and π_{s_π}, and the only nontrivial operations that such a circuit needs to compute are C and $\mathrm{Dec}_{\mathrm{BSC}}$ which can be computed by circuits of size N^c and N^D (respectively). We have made sure to choose $c_\epsilon \geq c + D + c_0$ for a sufficiently large constant c_0, so that the total size of T can be bounded by $c_\epsilon - 3$.
- The distribution $Z \leftarrow \mathrm{expr}^{\mathrm{ed}}(m, C)$ conditioned on the event $\{S_{\text{samp}} = s_{\text{samp}}, S_\pi = s_\pi\}$ is $N^{-(c_\epsilon - 3)}$-pseudorandom against circuits of size $N^{c_\epsilon - 3}$. (We have already argued that this is the case in Lemma 7.12).

It follows that for every fixed values $s_{\text{samp}}, s_\pi \in \{0,1\}^{\ell'}$:

$$|\Pr_{\mathrm{expr}^{\mathrm{ed}}(m,C)} [A_{s_{\text{samp}}, s_\pi}(C(Z)) = 1 \mid S_{\text{samp}} = s_{\text{samp}}, S_\pi = s_\pi] - \Pr[A_{s_{\text{samp}}, s_\pi}(C(U_N)) = 1]| \leq N^{-(c_\epsilon - 3)}. \tag{12}$$

We are now ready to prove Lemma 7.3. Let:

$$P = \Pr_{\mathrm{expr}^{\mathrm{ed}}(m,C)} [S \in \mathit{CANDIDATES} \text{ and } \bar{M}(S) = m].$$

$$\begin{aligned}
P &\geq \Pr_{\mathrm{expr}^{\mathrm{ed}}(m,C)}[A_{S_{\mathrm{samp}},S_\pi}(C(Z))=1] \\
&= \mathbb{E}_{s_{\mathrm{samp}},s_\pi \leftarrow \{0,1\}^{\ell'}}\left[\Pr_{\mathrm{expr}^{\mathrm{ed}}(m,C)}[A_{S_{\mathrm{samp}},S_\pi}(C(Z))=1 \mid S_{\mathrm{samp}}=s_{\mathrm{samp}}, S_\pi = s_\pi]\right] \\
&\geq \mathbb{E}_{s_{\mathrm{samp}},s_\pi \leftarrow \{0,1\}^{\ell'}}\left[\Pr_{\mathrm{expr}^{\mathrm{ed}}(m,C)}[A_{S_{\mathrm{samp}},S_\pi}(C(U_n))=1 \mid S_{\mathrm{samp}}=s_{\mathrm{samp}}, S_\pi = s_\pi] - N^{-(c_\epsilon-3)}\right] \\
&= \mathbb{E}_{s_{\mathrm{samp}},s_\pi \leftarrow \{0,1\}^{\ell'}}\left[\Pr[A_{s_{\mathrm{samp}},s_\pi}(C(U_n))=1]\right] - N^{-(c_\epsilon-3)} \\
&= \Pr_{S_{\mathrm{samp}},S_\pi \leftarrow \{0,1\}^{\ell'}}[A(S_{\mathrm{samp}},S_\pi,C(U_n))=1] - N^{-(c_\epsilon-3)} \\
&\geq 1 - N^{-c_s} - N^{-(c_\epsilon-1)} - N^{-(c_\epsilon-3)} \\
&\geq 1-\nu/2,
\end{aligned}$$

where the first line is by (11), the third line is by (12), the fifth line is by Lemma 7.15, and the final inequality is because $\nu = N^{-c}$ and we can choose the constant c_0 so that c_s and $c_\epsilon - 3$ are larger than c.

8 An analysis of random stochastic codes

In this section we prove Theorem 1.2, showing that a random stochastic code with rate approaching $1-H(p)$ is good against any class $\mathcal{C} \subseteq \mathrm{Ham}_p$ which contains "few" channels (where "few" means $2^{2^{O(n)}}$). One example is the class $\mathrm{Ckt}_p^{n^c}$ (or even $\mathrm{Ckt}_p^{2^{O(n)}}$).

We stress that this is not obvious, and was not known before. We also stress that this result by itself does not imply an explicit (or Monte-Carlo explicit) construction. See Section 1.2 for a discussion on the difference between random constructions and explicit Monte-Carlo construction. We start by restating Theorem 1.2 in a more precise way.

Theorem 8.1 (Random stochastic codes that decode against small families)**.** *For every constants $0 \leq p < \frac{1}{4}$ and $\epsilon > 0$, there exist constants $\alpha > 0$ and $c_d > 0$, such that for $R = 1 - H(p) - \epsilon$, and for every sufficiently large n, the following holds: Let $\mathcal{C} \subseteq \mathrm{Ham}_p$ be a class of functions that contains at most $2^{2^{\alpha \cdot n}}$ functions, and let $\mathrm{Enc} : \{0,1\}^{Rn} \times \{0,1\}^{c_d \cdot n} \to \{0,1\}^n$ be chosen uniformly from all such functions. Let $\mathrm{Dec} : \{0,1\}^n \to \{0,1\}^{Rn}$, be the map that on input $v \in \{0,1\}^n$, finds $m \in \{0,1\}^k$, and $s \in \{0,1\}^d$, such that $\delta(v, \mathrm{Enc}(m,s))$ is minimal (breaking ties arbitrarily). With probability $1 - 2^{-2^{\alpha \cdot n}}$ over the choice of Enc, $(\mathrm{Enc}, \mathrm{Dec})$ is a code for $\mathcal{C}$ with success probability $1 - 2^{-\alpha \cdot n}$.*

8.1 Proof of Theorem 8.1

We will use the methodology explained in Section 4. This proof has some a similar flavor to the proof given in Section 5.6 that random codes are evasive.

8.1.1 Preparations for the methodology of Section 4

The setup: Let $0 < p < \frac{1}{4}$ be a constant, let $\epsilon > 0$ be a sufficiently small constant, and let n be an integer (that we are allowed to assume that is sufficiently large).

Some notation: Let $R = 1 - H(p) - \epsilon$, $k = Rn$, $d = c_d \cdot n$ for a constant c_d that we will choose later on. Let $K = 2^k$, $D = 2^D$ and $L = K \cdot D$.

Throughout this proof we will identify strings $m \in \{0,1\}^k$ with numbers $m \in [K]$, and strings $s \in \{0,1\}^d$ with numbers $s \in [D]$. Let $\mathcal{J} = \{0,1\}^k \times \{0,1\}^d$. For $j \in \mathcal{J}$, we will use the notation (j^1, j^2) to denote its two parts, and will identify j with numbers in $[L]$ in the obvious way. Let $\mathcal{J}^m = \left\{j \in \mathcal{J} : j^1 = m\right\}$.

We will use the following notation for the probability space of choosing Enc.

Experiment expr**: A random code.**

- Let $\mathcal{X} = (\{0,1\}^n)^L(\mathrm{ds})$. Namely the set of all L distinct tuples $(x_1, \ldots, x_L) \in \{0,1\}^n$.
- Let $X \leftarrow \mathcal{X}$ be a uniform element $X = (X_1, \ldots, X_L)$ from $\mathcal{X}$.
- Let $\mathrm{Enc}(j) = X_j$, namely $\mathrm{Enc}(m,s) = X_{(m,s)}$.

8.1.2 The game of a channel C

Following the recipe in Section 4 we now define a game for a channel C.

Definition 8.2 (The game of a channel C)**.** *Given $C \in \mathcal{C}$, $x \in \mathcal{X}$, $m \in \{0,1\}^k$ and $j \in \mathcal{J}^m$, we say that C* **wins on** x, m, j *if*

$$\exists \bar{j} \neq j : \delta(x_j \oplus C(x_j), x_{\bar{j}}) \leq p.$$

To make the notation easier, we define $W^{C,m}(x,j) = 1$ iff C wins on x, m, j.

Note when considering the code Enc defined by x, if $\mathrm{Dec}(x_j \oplus C(x_j)) \neq m$ then C wins on x, m, j. This means that whenever C is able to make Dec decode incorrectly, when corrupting some x_j which is an encoding of s, then C wins. (Note that we could have also required that $\bar{j} \notin \mathcal{J}^m$, but we don't bother to do so).

Let $\alpha > 0$ be a sufficiently small constants that we will choose later. We will shoot for "decoding error" $\nu = 2^{-\alpha \cdot n}$. Definition 8.2 is made so that the task of proving Theorem 8.1 reduces to the task of proving that:

$$\Pr_{X \leftarrow \mathcal{X}}\left[\exists C \exists m : C \in \mathcal{C}, m \in \{0,1\}^k \text{ s.t. } \Pr_{j \leftarrow \mathcal{J}^m}\left[W^{C,m}(X,j) = 1\right] > 2^{-\alpha \cdot n}\right] < 2^{-2^{\alpha \cdot n}}.$$

This will follow by a union bound if we can prove that for every one of the $2^{2^{\alpha \cdot n}}$ choices of $C \in \mathcal{C}$ and every one of the 2^k choices of $m \in \{0,1\}^k$:

$$\Pr_{X \leftarrow \mathcal{X}}\left[\Pr_{j \leftarrow \mathcal{J}^m}\left[W^{C,m}(X,j) = 1\right] > 2^{-\alpha \cdot n}\right] < \frac{2^{-2^{\alpha \cdot n}}}{2^{2^{\alpha \cdot n}} \cdot 2^k} = 2^{-(k + 2 \cdot 2^{\alpha n})}.$$

Let us fix some choice of $C \in \mathcal{C}$ and $m \in \{0,1\}^k$. Let $\mathcal{Z} = \mathcal{J}^m$ and let:

$$W_{\mathrm{avg}}(x) = \frac{1}{|\mathcal{Z}|} \cdot \sum_{z \in \mathcal{Z}} W^{C,m}(x,z),$$

as is done in Lemma 4.1. We would like to use Lemma 4.1. For that purpose we choose $t = 2^{2\alpha \cdot n}$, so that (using the fact that $k \leq n$) it is sufficient to prove that:

$$\Pr_{X \leftarrow \mathcal{X}}[W_{\mathrm{avg}}(X) > 2^{-\alpha \cdot n}] \leq e^{-t}. \tag{13}$$

We will use Lemma 4.1 choosing $\delta = 2$, and $\mu = \frac{2^{-\alpha \cdot n}}{3}$ (so that $(1+\delta) \cdot \mu = 2^{-\alpha \cdot n}$). We need to verify that we meet the condition on t in Lemma 4.1, and indeed we will make sure that:

$$t = 2^{2 \cdot \alpha \cdot n} \leq \frac{\mu \cdot \delta \cdot D}{2} = \frac{2^{(c_d - \alpha) \cdot n}}{3},$$

by choosing the constant $\alpha > 0$ to be smaller than say $c_d/5$. Using Lemma 4.1, it follows that in order to prove that (13) holds, it is sufficient to prove the following claim:

Claim 8.3. *For every integer* $0 \leq q < t$,

$$\Pr_{\substack{X;Z \leftarrow \mathcal{Z} \\ Z_1,\ldots,Z_q \overset{\text{wor}}{\leftarrow} \mathcal{Z}}} [W(X,Z) = 1 \mid W(X,Z_1) = \ldots = W(X,Z_q) = 1] \leq \mu,$$

We therefore focus our attention on proving Claim 8.3, and the lemma will follow once we prove Claim 8.3.

8.1.3 Using the recipe of Section 4: proof of Claim 8.3

Fix some $0 \leq q < t$. We will now show that we can express the event $\{W(X,Z_1) = \ldots = W(X,Z_q) = 1\}$ as a disjoint union of "simple events".

More specifically, the event $\{W(X,Z_1) = \ldots = W(X,Z_q) = 1\}$ can be viewed as a subset $T \subseteq \mathcal{X} \times \mathcal{Z}^q$ by setting:

$$T = \{(x; z_1, \ldots, z_q) : W(x,z_1) = \ldots = W(x,z_q) = 1\}.$$

We will now show that T can be expressed as a disjoint union of "simple events". Loosely speaking, a simple event E is a subset $E \subseteq T$ in which $z_1, \ldots, z_q$, as well as several x_j's are fixed, in a very specific way.

Definition 8.4 (Simple event)**.** *For every choice of:*

- $z_1, \ldots, z_q \in \mathcal{Z} = \mathcal{J}^m$.
- *A function* $h : [q] \to \{0,1\}^n$
- $j_1, \ldots, j_q \in \mathcal{J}$ *such that for every* $g \in [q]$, $z_g \neq j_g$.

We define a set $E \subseteq \mathcal{X} \times \mathcal{Z}^q$ *(called the* **simple event** *induced by* $z_1, \ldots, z_q$, h *and* $j_1, \ldots, j_q$*). The event* E *is defined by:*

$$E = D \times \{(z_1, \ldots, z_q)\},$$

where D *is the set of all* $x \in \mathcal{X}$ *such that for every* $g \in [q]$*:*

- $x_{j_g} = h(g)$.
- $\delta(x_{j_g}, x_{z_g} \oplus C(x_{z_g})) \leq p$.
- *For every* $u < j_g$, $\delta(x_u, x_{z_g} \oplus C(x_{z_g})) > p$. *(Recall that we identify* $z_g \in \mathcal{J}$ *with a number in* $[L]$.

We will say that a simple event E *is nontrivial if*

$$\Pr_{X \leftarrow \mathcal{X}, Z_1,\ldots,Z_q \overset{\text{wor}}{\leftarrow} \mathcal{Z}} [(X, Z_1, \ldots, Z_q) \in E] > 0.$$

Experiment expr_2**: conditioning on a simple event.** We will be interested in the distribution obtained by conditioning the distribution $(X \leftarrow \mathcal{X}, Z_1, \ldots, Z_q \overset{\mathrm{wor}}{\leftarrow} \mathcal{Z})$ on $\{(X, Z_1, \ldots, Z_q) \in E\}$ for a nontrivial simple event E induced by $z_1, \ldots, z_q, h$ and $j_1, \ldots, j_q$. Let us denote this experiment by $\mathrm{expr}_2(z_1, \ldots, z_q; h; j_1, \ldots, j_q)$. We observe that for $(X, Z_1, \ldots, Z_q) \leftarrow \mathrm{expr}_2(z_1, \ldots, z_q; h; j_1, \ldots, j_q)$ we have that for every $g \in [q]$:

- Z_g is fixed to z_g and $z_g \neq j_g$.
- X_{j_g} is fixed to $x_{j_q} = h(g)$, such that $\delta(x_{j_g}, x_{z_g} \oplus C(x_{z_g})) \leq p$ (which means that $W(x, z_g) = 1$ and C wins on x, m, z_g).
- j_g is the smallest index u for which $\delta(x_{j_g}, x_{z_g} \oplus C(x_{z_g})) \leq p$, or in other words, that for every $u < j_g$, $\Pr[\delta(x_u, x_{z_g} \oplus C(x_{z_g})) > p] = 1$.

This means that $X \leftarrow \mathrm{expr}_2(z_1, \ldots, z_q; h; j_1, \ldots, j_q)$ is distributed as follows:

- For $u \in \{j_1, \ldots, j_q\}$, X_u is fixed to a value $h(g)$ in the Hamming ball of radius p around $X_{z_g} \oplus C(X_{z_g})$.
- For $u \notin \{j_1, \ldots, j_q\}$, X_u is uniformly distributed over a subset of $\{0,1\}^n$, which is of size at least $2^n - q \cdot 2^{H(p) \cdot n} - q$. This is because X_u is uniform given the following two restrictions:
  - X_u does not belong to a Hamming ball of relative radius p around some $X_{z_g} \oplus X_{z_g}$ where the corresponding index $j_g > u$.

    There are at most q choices for g, and each one rules out a Hamming ball of relative radius p, which is a set of size at most $2^{H(p) \cdot n}$.
  - X_u cannot take the values of $x_{j_1}, \ldots, x_{j_q}$. This means that at most an additional q values are not available.

  Overall out of the initial 2^n values, at most $q \cdot 2^{H(p) \cdot n} + q$ are no longer available, but X_u is free to take any of the remaining values.
- Furthermore, if $u \notin \{j_1, \ldots, j_q\}$ and $j \neq u$, we have that for every possible fixing $a \in \{0,1\}^n$ for X_j, the distribution $(X_u | X_j = a)$ is uniform over a set of size at least $2^n - q \cdot 2^{H(p) \cdot n} - q - 1$.

  This is because, the condition that $\{X_j = a\}$ can at worst, affect X_u in that one more slot (the value a) is not available.

We can also conclude that:

- Every simple event E, satisfies $E \subseteq T = \{W(X, Z_1) = \ldots = W(X, Z_q) = 1\}$. This is because by definition on every $(x, z_1, \ldots, z_q) \in E$, $W(x, z_1) = \ldots = W(x, z_q) = 1$.
- Every two simple events are either equal or disjoint. This is because in order for two simple events to have a non-empty intersection, the two events must agree on $z_1, \ldots, z_q$. They also must agree on $j_1, \ldots, j_q$, because if they don't agree on some j^g, then one of the two simple events will use a larger j_g, enforcing that for all $u < j_g$, $\delta(x_u, x_{z_g} \oplus C(x_{z_j})) > p$, and this cannot occur on the other simple event. Once they agree on $j_1, \ldots, j_q$, they must also (by definition) agree on h.

The discussion above implies that:

- The event $T = \{W(X, Z_1) = \ldots = W(X, Z_q) = 1\}$ is a disjoint union of nontrivial simple events.

- In order to show that:

$$\Pr_{\substack{X \leftarrow \mathcal{X}, Z \leftarrow \mathcal{Z} \\ Z_1,\ldots,Z_q \overset{\text{wor}}{\leftarrow} \mathcal{Z}}}[W(X,Z)=1 \mid W(X,Z_1)=\ldots=W(X,Z_q)=1] \le \mu,$$

  it is sufficient to show that for every choice of nontrivial simple event E that is induced by some $z_1,\ldots,z_q$, h and $j_1,\ldots,j_q$:

$$\Pr_{\substack{X \leftarrow \mathcal{X}, Z \leftarrow \mathcal{Z} \\ Z_1,\ldots,Z_q \overset{\text{wor}}{\leftarrow} \mathcal{Z}}}[W(X,Z)=1 \mid (X,Z_1,\ldots,Z_q) \in E] \le \mu,$$

- Simplifying the expression above, it is sufficient to show the following for every choice of nontrivial simple event E that is induced by some $z_1,\ldots,z_q$, h and $j_1,\ldots,j_q$:

$$\Pr_{\substack{X \leftarrow \mathrm{expr}_2(z_1,\ldots,z_q;h;j_1,\ldots,j_q) \\ Z \leftarrow \mathcal{Z}}}[W(X,Z)=1] \le \mu. \tag{14}$$

In the remainder of the proof, we will prove that (14) holds. We will fix some nontrivial simple event E that is induced by some $z_1,\ldots,z_q$, h and $j_1,\ldots,j_q$, and to avoid clutter, we will define:

$$\mathrm{expr}_2 = \mathrm{expr}_2(z_1,\ldots,z_q;h;j_1,\ldots,j_q).$$

We are interested in bounding:

$$\Pr_{\substack{X \leftarrow \mathrm{expr}_2 \\ Z \leftarrow \mathcal{Z}}}[W(X,Z)=1] = \Pr_{\substack{X \leftarrow \mathrm{expr}_2 \\ Z \leftarrow \mathcal{Z}}}[\exists j \ne Z : \delta(X_j, X_Z \oplus C(X_Z)) \le p].$$

The two next claims bound this probability for a specific j, depending on whether $j \in \{j_1,\ldots,j_q\}$ or $j \notin \{j_1,\ldots,j_q\}$.

Claim 8.5. *For every $g \in [q]$,*

$$\Pr_{\substack{X \leftarrow \mathrm{expr}_2 \\ Z \leftarrow \mathcal{Z}}}[\delta(X_{j_g}, X_Z \oplus C(X_Z)) \le p \textit{ and } Z \ne j_g] \le 2 \cdot 2^{-(1-H(2p))\cdot n} + 2^{-(c_d - 2\alpha)n}$$

Proof. Recall that for $X \leftarrow \mathrm{expr}_2$, X_{j_g} is fixed to $h(g)$. Let

$$P = \Pr_{\substack{X \leftarrow \mathrm{expr}_2 \\ Z \leftarrow \mathcal{Z}}}[\delta(X_{j_g}, X_Z \oplus C(X_Z)) \le p \text{ and } Z \ne j_g].$$

It follows that:

$$\begin{aligned}
P &= \Pr_{\substack{X\leftarrow \mathrm{expr}_2\\ Z\leftarrow \mathcal{Z}}}[\delta(h(g), X_Z \oplus C(X_Z)) \le p \text{ and } Z \ne j_g] \\
&= \sum_{z\in \mathcal{Z}\setminus\{j_g\}} \Pr_{\substack{X\leftarrow \mathrm{expr}_2\\ Z\leftarrow \mathcal{Z}}}[\delta(h(g), X_Z \oplus C(X_Z)) \le p \text{ and } Z = z] \\
&= \sum_{z\in \mathcal{Z}\setminus\{j_g\}} \Pr_{\substack{X\leftarrow \mathrm{expr}_2\\ Z\leftarrow \mathcal{Z}}}[\delta(h(g), X_z \oplus C(X_z)) \le p \mid Z = z] \cdot \Pr_{Z\leftarrow \mathcal{Z}}[Z=z] \\
&= \sum_{z\in \mathcal{Z}\setminus\{j_g\}} \Pr_{X\leftarrow \mathrm{expr}_2}[\delta(h(g), X_z \oplus C(X_z)) \le p] \cdot \Pr_{Z\leftarrow \mathcal{Z}}[Z=z] \\
&\le \sum_{z\in \mathcal{Z}\setminus\{j_1,\dots,j_q\}} \Pr_{X\leftarrow \mathrm{expr}_2}[\delta(h(g), X_z \oplus C(X_z)) \le p] \cdot \Pr_{Z\leftarrow \mathcal{Z}}[Z=z] + \frac{q}{D} \\
&\le \sum_{z\in \mathcal{Z}\setminus\{j_1,\dots,j_q\}} \Pr_{X\leftarrow \mathrm{expr}_2}[\delta(h(g), X_z) \le 2\cdot p] \cdot \Pr_{Z\leftarrow \mathcal{Z}}[Z=z] + \frac{q}{D} \\
&\le \max_{z\in \mathcal{Z}\setminus\{j_1,\dots,j_q\}} \left(\Pr_{X\leftarrow \mathrm{expr}_2}[\delta(h(g), X_z) \le 2\cdot p] \right) + \frac{t}{2^{c_d\cdot n}} \\
&\le \frac{2^{H(2p)\cdot n}}{2^n - q\cdot 2^{H(p)\cdot n - q - 1}} + 2^{2\cdot\alpha n - c_d\cdot n} \\
&\le 2\cdot 2^{-(1-H(2p))\cdot n} + 2^{-(c_d - 2\alpha)n},
\end{aligned}$$

where the fourth line follows because X and Z are independent, the fifth line follows because Z is uniform over $\mathcal{Z}$ which is of size D, the sixth line follows because using the fact that $C \in \mathcal{C} \subseteq \mathrm{Ham}_p$, we have that $\delta(X_z, X_z \oplus C(X_z)) \le p$, and then the sixth line follows by the triangle inequality, that is, that $\delta(h(g), X_z \oplus C(X_z)) \le p$ implies $\delta(h(g), X_z) \le 2p$. Finally, the penultimate line follows because for $z \in \mathcal{J} \setminus \{j_1,\dots,j_q\}$, we have that X_z is uniform over a set of size at least $2^n - q\cdot 2^{H(p)\cdot n} - q - 1$. □

Claim 8.6. *For every $j \in \mathcal{J} \setminus \{j_1,\dots,j_q\}$,*

$$\Pr_{\substack{X\leftarrow \mathrm{expr}_2\\ Z\leftarrow \mathcal{Z}}}[\delta(X_j, X_Z \oplus C(X_Z)) \le p \textit{ and } Z \ne j] \le 2\cdot 2^{-(1-H(p))\cdot n}$$

Proof. Recall that for $X \leftarrow \mathrm{expr}_2$, and $j \in \mathcal{J} \setminus \{j_1,\dots,j_q\}$, we have that X_j is uniform over a set of size at least $2^n - q\cdot 2^{H(p)\cdot n} - q - 1$. Furthermore, this is still the case even after conditioning on the event $\{X_{j'} = a\}$ for every $j' \ne j$ and $a \in \{0,1\}^n$. It follows that:

$$\begin{aligned}
\Pr_{\substack{X\leftarrow \mathrm{expr}_2\\ Z\leftarrow \mathcal{Z}}}[\delta(X_j, X_Z \oplus C(X_Z)) \le p \text{ and } Z \ne j] &\le \max_{z\in\mathcal{Z}\setminus\{j\}} \left(\Pr_{X\leftarrow \mathrm{expr}_2}[\delta(X_j, X_Z \oplus C(X_Z)) \le p \mid Z = z] \right) \\
&\le \max_{z\in\mathcal{Z}\setminus\{j\}} \left(\Pr_{X\leftarrow \mathrm{expr}_2}[\delta(X_j, X_z \oplus C(X_z)) \le p] \right) \\
&\le \frac{2^{H(p)\cdot n}}{2^n - q\cdot 2^{H(p)\cdot n} - q - 1} \\
&\le 2\cdot 2^{-(1-H(p))\cdot n},
\end{aligned}$$

where the second line follows because X, Z are independent, and the last line follows because $q \le t = 2^{2\cdot\alpha\cdot n}$, $p < \frac{1}{4}$, and we can take $\alpha > 0$ to be sufficiently small, so that $t\cdot 2^{H(p)\cdot n} + t \le 2^{n-1}$. □

We are finally ready to prove Claim 8.3.

$$\begin{aligned}
\Pr_{\substack{X\leftarrow \mathrm{expr}_2 \\ Z\leftarrow \mathcal{Z}}}[W(X,Z)=1] &= \Pr_{\substack{X\leftarrow \mathrm{expr}_2 \\ Z\leftarrow \mathcal{Z}}}[\exists j \neq Z : \delta(X_j, X_Z \oplus C(X_Z)) \leq p] \\
&\leq \sum_{j\in\mathcal{J}} \Pr_{\substack{X\leftarrow \mathrm{expr}_2 \\ Z\leftarrow \mathcal{Z}}}[\delta(X_j, X_Z \oplus C(X_Z)) \leq p \text{ and } Z \neq j] \\
&\leq \sum_{j\in\{j_1,\dots,j_q\}} \Pr_{\substack{X\leftarrow \mathrm{expr}_2 \\ Z\leftarrow \mathcal{Z}}}[\delta(X_j, X_Z \oplus C(X_Z)) \leq p \text{ and } Z \neq j] \\
&\quad + \sum_{j\in\mathcal{J}\setminus\{j_1,\dots,j_q\}} \Pr_{\substack{X\leftarrow \mathrm{expr}_2 \\ Z\leftarrow \mathcal{Z}}}[\delta(X_j, X_Z \oplus C(X_Z)) \leq p \text{ and } Z \neq j] \\
&\leq q \cdot \left(2 \cdot 2^{-(1-H(2p))\cdot n} + 2^{-(c_d-2\alpha)n}\right) + L \cdot 2 \cdot 2^{-(1-H(p))\cdot n} \\
&\leq q \cdot \left(2 \cdot 2^{-O(\epsilon^2)\cdot n} + 2^{-(c_d-2\alpha)n}\right) + 2 \cdot 2^{-(\epsilon-c_d)\cdot n} \\
&\leq 2^{2\cdot\alpha\cdot n} \cdot \left(2 \cdot 2^{-O(\epsilon^2)\cdot n} + 2^{-(c_d-2\alpha)n}\right) + 2 \cdot 2^{-(\epsilon-c_d)\cdot n} \\
&\leq \mu,
\end{aligned}$$

where the fourth line follows using Claim 8.5 and Claim 8.6, the fifth line follows because we can choose ϵ to be sufficiently small so that $p+\epsilon < \frac{1}{4}$, which gives that $2p < \frac{1}{2} - 2\epsilon$, and $H(\frac{1}{2} - 2\epsilon) \leq 1 - O(\epsilon^2)$, the fifth line also uses that $L = K \cdot D = 2^{(1-H(p)-\epsilon)\cdot n + c_d \cdot n}$, the penultimate line follows because $q < t = 2^{2\cdot\alpha\cdot n}$, and the final line follows because $\mu = \frac{2^{-\alpha\cdot n}}{3}$, and given p and ϵ, we are free to choose $\alpha > 0$ and $c_d > 0$ with the requirements that $\alpha > 0$ is sufficiently small, and that $c_d \geq 10 \cdot \alpha$. We can fulfil these requirements by choosing c_d to be sufficiently small so that $\epsilon - c_d \geq \alpha$, and then choosing $\alpha > 0$ to be sufficiently small so that $10 \cdot \alpha \leq c_d$ and $\epsilon^2 \geq 2\alpha$.

Remark 8.7 (Random codes with rate $1 - H(p) - o(1)$ for $p = \frac{1}{2} - o(1)$)**.** *It can be seen by this proof, that it was not crucial that p and ϵ are constants if one allows α not to be a constant. The computation above shows that one can take for exdample, $p = \frac{1}{2} - o(1)$, and $\epsilon = o(1)$, as long as $p + \epsilon < \frac{1}{4}$ and $\alpha = o(1)$ is chosen to be sufficiently small so that the argument above can be repeated. Loosely speaking, it will be sufficient that the volume of a Hamming ball of radius $\frac{1}{2} - \epsilon$ is sufficiently small compared to $2^{(1-O(\alpha))\cdot n}$. We omit the details.*

Acknowledgements

We thank anonymous referees for helpful comments and suggestions.

References

[BD22] G. Blanc and D. Doron. New near-linear time decodable codes closer to the GV bound. *Electron. Colloquium Comput. Complex.*, TR22-027, 2022.

[BDK$^+$19] M. Ball, D. Dachman-Soled, M. Kulkarni, H. Lin, and T. Malkin. Non-malleable codes against bounded polynomial time tampering. In *Advances in Cryptology - EUROCRYPT 2019 - 38th*

Annual International Conference on the Theory and Applications of Cryptographic Techniques, volume 11476 of *Lecture Notes in Computer Science*, pages 501–530. Springer, 2019.

[BDL22] M. Ball, D. Dachman-Soled, and J. Loss. (nondeterministic) hardness vs. non-malleability. *Electron. Colloquium Comput. Complex.*, TR22-010, 2022.

[BR94] M. Bellare and J. Rompel. Randomness-efficient oblivious sampling. In *35th Annual Symposium on Foundations of Computer Science*, pages 276–287, 1994.

[CG16] M. Cheraghchi and V. Guruswami. Capacity of non-malleable codes. *IEEE Trans. Inf. Theory*, 62(3):1097–1118, 2016.

[CJL15] Z. Chen, S. Jaggi, and M. Langberg. A characterization of the capacity of online (causal) binary channels. In *Proceedings of the forty-seventh annual ACM symposium on Theory of computing*, pages 287–296, 2015.

[DHRS07] Y. Ding, D. Harnik, A. Rosen, and R. Shaltiel. Constant-round oblivious transfer in the bounded storage model. *J. Cryptology*, 20(2):165–202, 2007.

[DJLS13] B. K. Dey, S. Jaggi, M. Langberg, and A. D. Sarwate. Upper bounds on the capacity of binary channels with causal adversaries. *IEEE Transactions on Information Theory*, 59(6):3753–3763, 2013.

[DKP21] D. Dachman-Soled, I. Komargodski, and R. Pass. Non-malleable codes for bounded parallel-time tampering. In *Advances in Cryptology - CRYPTO 2021 - 41st Annual International Cryptology Conference, CRYPTO 2021*, volume 12827 of *Lecture Notes in Computer Science*, pages 535–565. Springer, 2021.

[DPW18] S. Dziembowski, K. Pietrzak, and D. Wichs. Non-malleable codes. *J. ACM*, 65(4):20:1–20:32, 2018.

[FMVW16] S. Faust, P. Mukherjee, D. Venturi, and D. Wichs. Efficient non-malleable codes and key derivation for poly-size tampering circuits. *IEEE Trans. Inf. Theory*, 62(12):7179–7194, 2016.

[For65] G. D. Forney. *Concatenated codes.* PhD thesis, Massachusetts Institute of Technology, 1965.

[GI05] V. Guruswami and P. Indyk. Linear-time encodable/decodable codes with near-optimal rate. *IEEE Transactions on Information Theory*, 51(10):3393–3400, 2005.

[Gol97] O. Goldreich. A sample of samplers - a computational perspective on sampling (survey). *Electronic Colloquium on Computational Complexity (ECCC)*, 4(20), 1997.

[GS16] V. Guruswami and A. Smith. Optimal rate code constructions for computationally simple channels. *Journal of the ACM (JACM)*, 63(4):35, 2016.

[GUV07] V. Guruswami, C. Umans, and S. P. Vadhan. Unbalanced expanders and randomness extractors from parvaresh-vardy codes. In *CCC*, pages 96–108, 2007.

[IK10] R. Impagliazzo and V. Kabanets. Constructive proofs of concentration bounds. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, 13th International Workshop, APPROX 2010, and 14th International Workshop, RANDOM 2010*, volume 6302 of *Lecture Notes in Computer Science*, pages 617–631, 2010.

[IW97] R. Impagliazzo and A. Wigderson. $P = BPP$ if E requires exponential circuits: Derandomizing the XOR lemma. In *STOC*, pages 220–229, 1997.

[JST21] F. G. Jeronimo, S. Srivastava, and M. Tulsiani. Near-linear time decoding of ta-shma's codes via splittable regularity. In *STOC '21: 53rd Annual ACM SIGACT Symposium on Theory of Computing*, pages 1527–1536, 2021.

[KMRS17] S. Kopparty, O. Meir, N. Ron-Zewi, and S. Saraf. High-rate locally correctable and locally testable codes with sub-polynomial query complexity. *Journal of the ACM (JACM)*, 64(2):11, 2017.

[KNR09] E. Kaplan, M. Naor, and O. Reingold. Derandomized constructions of *k*-wise (almost) independent permutations. *Algorithmica*, 55(1):113–133, 2009.

[KSS19] S. Kopparty, R. Shaltiel, and J. Silbak. Quasilinear time list-decodable codes for space bounded channels. *To appear in the 60th Annual Symposium on Foundations of Computer Science (FOCS)*, 2019.

[Lip94] R. J. Lipton. A new approach to information theory. In *11th Annual Symposium on Theoretical Aspects of Computer Science*, pages 699–708, 1994.

[MRRW77] R. McEliece, E. Rodemich, H. Rumsey, and L. Welch. New upper bounds on the rate of a code via the delsarte-macwilliams inequalities. *IEEE Transactions on Information Theory*, 23(2):157–166, 1977.

[Smi07] A. D. Smith. Scrambling adversarial errors using few random bits, optimal information reconciliation, and better private codes. In *Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA*, pages 395–404, 2007.

[SS21a] R. Shaltiel and J. Silbak. Explicit list-decodable codes with optimal rate for computationally bounded channels. *Comput. Complex.*, 30(1):3, 2021.

[SS21b] R. Shaltiel and J. Silbak. Explicit uniquely decodable codes for space bounded channels that achieve list-decoding capacity. In *STOC '21: 53rd Annual ACM SIGACT Symposium on Theory of Computing*, pages 1516–1526, 2021.

[SSS95] J. P. Schmidt, A. Siegel, and A. Srinivasan. Chernoff-hoeffding bounds for applications with limited independence. *SIAM J. Discrete Math.*, 8(2):223–250, 1995.

[TS17] A. Ta-Shma. Explicit, almost optimal, epsilon-balanced codes. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing, STOC*, pages 238–251, 2017.

[Zuc97] David Zuckerman. Randomness-optimal oblivious sampling. *Random Struct. Algorithms*, 11(4):345–367, 1997.

ECCC ISSN 1433-8092

https://eccc.weizmann.ac.il