computational complexity

DERANDOMIZED PARALLEL REPETITION VIA STRUCTURED PCPS

Irit Dinur and Or Meir

Abstract. A PCP is a proof system for NP in which the proof can be checked by a probabilistic verifier. The verifier is only allowed to read a very small portion of the proof and in return is allowed to err with some bounded probability. The probability that the verifier accepts a proof of a false claim is called the soundness error and is an important parameter of a PCP system that one seeks to minimize. Constructing PCPs with subconstant soundness error and, at the same time, a minimal number of queries into the proof (namely two) is especially important due to applications for inapproximability.

In this work, we construct such PCP verifiers, i.e., PCPs that make only two queries and have subconstant soundness error. Our construction can be viewed as a combinatorial alternative to the "manifold vs. point" construction, which is the basis for all constructions in the literature for this parameter range. The "manifold vs. point" PCP is based on a low-degree test, while our construction is based on a direct product test. We also extend our construction to yield a decodable PCP (dPCP) with the same parameters. By plugging in this dPCP into the scheme of Dinur and Harsha (FOCS 2009), one gets an alternative construction of the result of Moshkovitz and Raz (FOCS 2008), namely a construction of two-query PCPs with small soundness error and small alphabet size. Our construction of a PCP is based on extending the derandomized direct product test of Impagliazzo, Kabanets, and Wigderson (STOC 09) to a derandomized parallel repetition theorem. More accurately, our PCP construction is obtained in two steps. We first prove a derandomized parallel repetition theorem for specially structured PCPs. Then, we show that any PCP can be transformed into one that has the required structure, by embedding it on a de-Bruijn graph.

🕮 Birkhäuser

1. Introduction

The PCP theorem (Arora *et al.* 1998; Arora & Safra 1998) says that every language in NP can be verified by a polynomial time verifier that queries proof of polynomial length in a constant number of locations. The verifier is guaranteed to always accept a correct proof and to accept a proof of a false claim with bounded probability (called the *soundness error*). Following the proof of the PCP theorem, research has been directed toward strengthening the PCP theorem in terms of the important parameters, such as the proof length, the number of queries, the soundness error, and the randomness complexity of the verifier.

In parallel, there is a line of work attempting to expand the variety of techniques at our disposal for constructing PCPs. Here, the aim is to gain a deeper and more intuitive understanding of why PCP theorems hold. One of the threads in this direction is replacing algebraic constructions by combinatorial ones. This is motivated by the intuition that algebra is not an essential component of PCPs; indeed, the definition of PCPs involves no algebra at all. Of course, one may also hope that the discovery of new techniques may lead to new results.

For the "basic" PCP theorem (Arora *et al.* 1998; Arora & Safra 1998), there have been alternative combinatorial proofs (Dinur 2007; Dinur & Reingold 2006). It is still a challenge to match stronger PCP theorems with combinatorial constructions. Such is the work of the second author (Meir 2009) on PCPs with efficient verifiers. In this paper, we seek to do so for PCPs in the small soundness error regime.

In this work, we give a new construction of a PCP with subconstant soundness error and two queries. This setting is particularly important for inapproximability, as will be discussed shortly below. In addition, our PCP maintains the polynomial proof length and logarithmic randomness complexity of the original PCP theorem of Arora *et al.* (1998), Arora & Safra (1998). Formally, we prove

THEOREM 1.1 (Two-query PCP with small soundness). *There exists a constant $\kappa > 0$ such that for every function $\varepsilon : \mathbb{N} \to (0,1)$ satisfying $1/n^\kappa \leq \varepsilon(n) \leq 1/\mathrm{poly}\log n$ the following holds: Every language $L \in \mathbf{NP}$ has a two-query PCP system with perfect completeness, soundness error ε, alphabet size $2^{1/\mathrm{poly}(\varepsilon)}$, proof length $\mathrm{poly}(n)$, and randomness complexity $O(\log n)$. Furthermore, the verifier in this PCP system makes only 'projection' queries.*

This theorem matches the parameters of the folklore "manifold vs. point" construction, which has been the only construction in the literature for this parameter range. The technical heart of that construction is a subconstant error low-degree test (Arora & Sudan 2003; Raz & Safra 1997), see full details in Moshkovitz & Raz (2008).

Our proof of Theorem 1.1 is based on the elegant derandomized direct product test of Impagliazzo *et al.* (2009). In a nutshell, our construction is based on applying this test to obtain a "derandomized parallel repetition theorem". While it is not clear how to do this for an arbitrary PCP, it turns out to be possible for PCPs with certain structure. We show how to convert any PCP to a PCP with the required structure and then prove a "derandomized parallel repetition theorem" for such PCPs, thereby getting Theorem 1.1. The derandomized parallel repetition theorem relies on a reduction from the derandomized direct product test of Impagliazzo *et al.* (2009).

The Moshkovitz–Raz construction.   Recently, Moshkovitz & Raz (2008) constructed even stronger PCPs. Specifically, they managed to remove the limitation $\varepsilon(n) \leq 1/\mathrm{poly}\log n$ from Theorem 1.1, thus allowing any function $\varepsilon(n) \geq 1/n^\kappa$. This allows constructing PCPs with subconstant error and *any* alphabet size smaller than $2^{\mathrm{poly}\log n}$, at the expense of a suitable increase in the soundness error. Being able to reduce the alphabet size has strong consequences for inapproximability, see Moshkovitz & Raz (2008) for details. The technique of Moshkovitz & Raz (2008) (as explained in the later simplification of Dinur & Harsha (2009)) is essentially based on the composition of certain PCP constructions.

In fact, their main building block is the "manifold vs. point" construction mentioned above.

Our construction can be extended to yield a so-called decodable PCP (Dinur & Harsha 2009), which is an object slightly stronger than a PCP. This can be plugged into the scheme of Dinur & Harsha (2009) to give a nearly[1] combinatorial proof of the following result of Moshkovitz & Raz (2008). Namely,

THEOREM 1.2 (Moshkovitz & Raz 2008). *There exists a constant $\kappa > 0$ such that for every function $\varepsilon(n) \geq 1/n^\kappa$, the following holds: Every language $L \in$ **NP** has a two-query PCP system with perfect completeness, soundness error ε, alphabet size at most $2^{1/\text{poly}(\varepsilon)}$, proof length* poly (n), *and randomness complexity $O(\log n)$. Furthermore, the verifier in this PCP system makes only 'projection' queries.*

We note that the result of Moshkovitz & Raz (2008) is in fact even stronger than claimed above since their verifier has almost-linear proof length (specifically $n^{1+o(1)}$) and has randomness complexity of only $(1 + o(1)) \log n$ random bits, see also Remark 6.27.

Organization of the introduction. In the following four sections, we outline the background and main ideas of this work. We start by describing the parallel repetition technique in general and its relation with direct product tests. We proceed to describe our technique of derandomized parallel repetition. We then describe our notion of "PCPs with linear structure", to which the derandomized parallel repetition is applied.

After the foregoing outline, we discuss relevant works and possible future directions and describe the organization of this work.

Parallel repetition and direct products. A natural approach to reducing the soundness error of a PCP verifier is by running it several times independently and accepting only if all runs accept. This is called *sequential repetition*. Obviously, if the verifier is invoked k times, the soundness error drops exponentially

---

[1]It is debatable whether our use of "linear structure" disqualifies the result from being considered purely combinatorial.

in k. However, the total number of queries made into the proof grows k-fold, and in particular, it is greater than 2. Since our focus is on constructing PCPs that make only two queries, we cannot afford sequential repetition.

In order to decrease the soundness error while maintaining the query complexity, one may use *parallel repetition*. For the rest of this discussion, we consider only PCPs that use only two queries. Let us briefly recall what parallel repetition means in this context. As in the case of sequential repetition, one starts out with a PCP with constant soundness error and then amplifies the rejection probability by repetition of the verifier. However, in order to save on queries, the prover is expected to give the k-wise direct product encoding of the original proof. Formally, if $\pi : [n] \to \Sigma$ describes the original proof, then its direct product encoding, denoted by $\pi^{\otimes k}$, is the function $\pi^{\otimes k} : [n]^k \to \Sigma^k$ defined by

$$\pi^{\otimes k}(x_1, \ldots, x_k) = (\pi(x_1), \ldots, \pi(x_k)).$$

The new verifier will simulate the original verifier on k independent runs but will read only *two* symbols from the new proof, which together contain answers to k independent runs of the original verifier.

Of course, there is no a priori guarantee that the given proof is a direct product encoding $\pi^{\otimes k}$ of any underlying proof π, as intended in the construction. This is the main difficulty proving the celebrated parallel repetition due to Raz (1998) that shows that the soundness error does go down exponentially with k.

One may try to circumvent the difficulty analyzing the parallel repetition theorem by augmenting it with a direct product test. That is, making the verifier *test* that the given proof Π is a direct product encoding of some string π and only then running the original parallel repetition verifier. This can sometimes be done without even incurring extra queries. Motivated by this approach, Goldreich & Safra (2000) suggested and studied the following question:

DP testing: Given a function $F : [n]^k \to \Sigma^k$ test that it is close to $f^{\otimes k}$ for some $f : [n] \to \Sigma$.

Let us now describe a two-query direct product test. From now on, let us make the simplifying assumption that the function

$F : [n]^k \to \Sigma^k$ to be tested is given as a function of k-sized subsets rather than tuples, meaning that $F(x_1, \ldots, x_k)$ is the same for any permutation of $x_1, \ldots, x_k$. The test chooses two random k-subsets $B_1, B_2 \in \binom{[n]}{k}$ that intersect on a subset $A = B_1 \cap B_2$ of a certain prescribed size and accept if and only if $F(B_1)_{|A} = F(B_2)_{|A}$. This test was analyzed further in several works, see Dinur & Goldenberg (2008), Dinur & Reingold (2006), Goldreich & Safra (2000), Impagliazzo *et al.* (2009).

REMARK 1.3. *An expert reader may note that the above direct product test is not a projection test, while we need a projection test for Theorem 1.1. Indeed, in our actual proof, we use a variant of the above direct product test which is a projection test (see Section 2.1 for details).*

Derandomized direct product testing.   Recall that our goal is to construct PCPs with subconstant soundness error. Note, however, that since the parallel repetition increases the proof length exponentially in k (and the randomness of the verifier grows k-fold), one can only afford to make a constant number of repetitions if one wishes to maintain polynomial proof length and logarithmic randomness complexity. On the other hand, obtaining subconstant soundness error requires a super-constant number of repetitions.

This leads to the derandomization question, addressed already 15 years ago (Feige & Kilian 1995). Can one recycle randomness of the verifier in the parallel repetition scheme without losing too much in soundness error?

Motivated by this question, Impagliazzo *et al.* (2009) introduced a method for analyzing the direct product test that allowed them to derandomize it. Namely, they exhibited a relatively small collection of subsets $\mathcal{K} \subset \binom{[n]}{k}$ and considered the restriction of the direct product encoding $f^{\otimes k}$ to this collection. They then showed that this form of derandomized direct product can be tested using the above test. The collection $\mathcal{K}$ is as follows: identify $[n]$ with a vector space $\mathbb{F}^m$, let $k = |\mathbb{F}|^d$ for constant d, and let $\mathcal{K}$ be the set of all d-dimensional linear subspaces.

A natural next step is to use the derandomized direct product of Impagliazzo *et al.* (2009) to obtain a derandomized parallel repetition theorem. Recall that the parallel repetition verifier works by simulating k independent invocations of the original verifier on π and querying the (supposed) direct product Π on the resulting k-tuples of queries. However, in the derandomized setting, the k-tuples of queries generated by the verifier may fall outside $\mathcal{K}$. This is the main difficulty that we address in this work.

This is where the structure of the PCP comes to our aid. We show that for PCPs with a certain linear structure, the k-tuples of queries can be made in a way that is compatible with the derandomized direct product test of Impagliazzo *et al.* (2009). More specifically, the k-tuples of queries always belong to the collection $\mathcal{K}$ and are distributed like queries of the derandomized direct product test. This allows us to prove a derandomized parallel repetition theorem for the particular case of PCPs with linear structure. Our main theorem is proved by constructing PCPs with linear structure (discussed next) and applying the derandomized parallel repetition theorem.

PCPs with linear structure. We turn to discuss PCPs with linear structure. The *underlying graph structure* of a two-query PCP is a graph defined as follows. The vertices are the proof coordinates, and the edges correspond to all possible query pairs of the verifier. (See also Section 2.3). We say that a graph has *linear structure* if the vertices can be identified with a vector space $\mathbb{F}^m$ and the edges, which clearly can be viewed as a subset of $\mathbb{F}^{2m}$, form a linear subspace of $\mathbb{F}^{2m}$ (see also Definition 3.1). A two-query PCP has linear structure if its underlying graph has linear structure.

As mentioned above, an additional contribution of this work is the construction of PCPs with linear structure. That is, we prove the following result.

THEOREM 1.4 (PCPs with linear structure). *Every language $L \in$* **NP** *has a two-query PCP system with a linear structure which has perfect completeness, soundness error $1 - 1/\text{poly} \log n$, constant alphabet size, proof length $\text{poly}\,(n)$, and randomness complexity $O(\log n)$.*

We believe that Theorem 1.4 is interesting in its own right: For known PCPs, the underlying graph structure is quite difficult to describe, mostly due to the fact that PCP constructions are invariably based on composition. In principle, however, the fact that a PCP is a "complex" object need not prevent the underlying graph from being simple. In analogy, certain Ramanujan expanders (Lubotzky *et al.* 1988) are Cayley graphs that are very easy to describe, even if the proof of their expansion is not quite so easy. It is therefore interesting to study whether there exist PCPs with simple underlying graphs.

Philosophically, the more structured the PCP, the stronger is the implied statement about the class NP, and the easier it is to exploit for applications. Indeed, the structure of a PCP system has been used in several previous works. For example, Khot (2006) constructs a PCP with quasi-random structure in order to establish the hardness of minimum bisection. Dinur (2007) imposes an expansion structure on a PCP to obtain amplification.

We prove Theorem 1.4 by embedding a given PCP into the de Bruijn graph and relying on the algebraic structure of this graph. We remark that the de Bruijn graph has been used in the constructions of PCPs before, e.g., Polishchuk & Spielman (1994), Babai *et al.* (1991), in similar contexts. We believe that structured PCPs are an object worthy of further study. One may view their applicability toward proving Theorem 1.1 as supporting evidence. An interesting question which we leave open is whether Theorem 1.4 can be strengthened so as to get *constant* soundness error. By simply plugging such a PCP into our derandomized parallel repetition theorem, one would get a direct proof of the aforementioned result of Moshkovitz & Raz (2008), *without* using two-query composition.

REMARK 1.5. *Our notion of PCPs with linear structure should not be confused with the notion of "linear PCPPs" that appeared in the literature before (see Ben-Sasson* et al. *(2009) and the related "linear inner verifier" of Goldreich & Safra (2000)). A linear PCPP is, roughly, a PCP system for checking the membership of a vector in a given linear subspace, in which the proof is required to be a linear function of the aforementioned vector. This requirement is unrelated to our definition, which does not restrict the claim to be*

verified or the proof and, on the other hand, restricts the query structure of the PCP verifier.

Decodable PCPs.   We extend our results to also yield a new construction of *decodable PCPs* (dPCPs). A dPCP gives a way to encode NP witnesses, so that a verifier (called a decoder in this context) is able to both locally test their validity and locally decode bits from the encoded NP witness. Decodable PCPs[2] were introduced in Dinur & Harsha (2009) toward simplifying and modularizing the work of Moshkovitz & Raz (2008) on two-query PCPs with small soundness. In Dinur & Harsha (2009), the result of Moshkovitz & Raz (2008) was reproved assuming the existence of two building blocks, a PCP and a dPCP, which were used as a black box. Until this work, there has been only one known construction of a dPCP, based on the manifold vs. point construction. In this work, we give a new construction of a dPCP which is obtained by applying derandomized parallel repetition in an analogous way to Theorem 1.1. We prove

THEOREM 1.6 (dPCP, informal version). *There exists a two-query PCP decoder with perfect completeness, soundness error $1/\mathrm{poly}\log n$, list size $\mathrm{poly}\log n$, proof alphabet $2^{\mathrm{poly}\log n}$, proof length $\mathrm{poly}\,(n)$, and randomness complexity $O(\log n)$.*

The notion of dPCPs is described in detail in Section 6 and, in particular, in Section 6.2. Theorem 1.6 is stated and proved in Section 6.4 based on two main lemmas, which are proved in Section 7 and Section 8.

In order to prove this theorem, we generalize each of the steps of the proof of Theorem 1.1. First, we construct a dPCP with linear structure but with relatively high soundness error in an analogous way to our proof of Theorem 1.4 (PCPs with linear structure). Next, we apply derandomized parallel repetition to get the desired dPCP. The two steps are described in Section 7 and Section 8 respectively.

---

[2]Decodable PCPs generalize the notion of "locally decode/reject codes" of Moshkovitz & Raz (2008) and the even earlier notion of "LDF readers" of Dinur *et al.* (1999).

An additional contribution of this work is an extension of the definitions of Dinur & Harsha (2009), of dPCPs that work with low soundness error, to one that works with high soundness error. This is necessary because plugging in a higher value for the soundness error parameter into the existing definition of Dinur & Harsha (2009) turns out to be useless. Instead, we give a variant which we call uniquely decodable PCPs (udPCPs). We show that udPCPs are in fact equivalent to PCPs of proximity (PCPPs). This allows us to rely on known constructions of PCPPs (Ben-Sasson *et al.* 2006; Dinur & Reingold 2006) as our starting point. For more details, see Section 6.2.

Together, Theorem 1.1 and Theorem 1.6 imply Theorem 1.2 (the Moshkovitz & Raz 2008 result). This is sketched in Section 6.5.

REMARK 1.7. *In fact, Theorem 1.6 can be proved for any soundness error $\varepsilon(n)$ satisfying $1/n^\kappa \leq \varepsilon(n) \leq 1/\mathrm{poly}\log n$ (for some constant $\kappa > 0$. As in Theorem 1.1, the alphabet size in such case is $2^{1/\mathrm{poly}(\varepsilon)}$, and furthermore, the list size becomes $1/\mathrm{poly}(\varepsilon)$. However, in this paper, we only prove Theorem 1.6 for $\varepsilon(n) = 1/\mathrm{poly}\log n$, since this is all we need to prove Theorem 1.2 (the Moshkovitz & Raz 2008 result).*

Related work and future directions.   Our final construction of a two-query PCP has exponential relation between the alphabet size and the error probability (that is, $|\Sigma| = 2^{1/\mathrm{poly}(\varepsilon)}$). In general, one can hope for a polynomial relation, and this is the so-called sliding scale conjecture of Bellare *et al.* (1993). Our approach is inherently limited to an exponential relation both because of a lower bound on direct product testing from Dinur & Goldenberg (2008) and, more generally, because of the following lower bound of Feige & Kilian (1995) on parallel repetition of games. Feige and Kilian prove that for every PCP system and $k = O(\log n)$ invocations of the original verifier, if one insists on the parallel repetition using only $O(\log n)$ random bits, then the soundness error must be at least $1/\mathrm{poly}\log n$ (and not $1/\mathrm{poly}(n)$ as one might hope). For the choice of $k = O(\log n)$, our work matches the Feige & Kilian (1995) lower bound by exhibiting a derandomized parallel

repetition theorem, albeit only for PCPs with linear structure, that achieves a matching upper bound of $1/\text{poly} \log n$ on the soundness error.

Nevertheless, for three queries, we are in a completely different ball-game, and no lower bound is known. It would be interesting to find a derandomized direct product test with three queries with lower soundness error and to try and adapt it to a PCP. We note that there are "algebraic" constructions (Raz & Safra 1997; Dinur et al. 1999) that make only three queries and have much better relationship between the error and the alphabet size.

It has already been mentioned that while our result matches the soundness error and alphabet size of the Moshkovitz & Raz (2008) result, it does not attain nearly-linear proof length. Improving our result in this respect is another interesting direction.

Structure of the paper.   The paper has two main parts: the first part is concerned with proving the main result for PCPs and the second part generalizes this result to dPCPs.

- ○ **Part 1.** The structure of the proof is "top to bottom". Our main theorem for PCPs is based on two main steps: (i) embedding a PCP into a PCP with linear structure and (ii) a derandomized parallel repetition theorem for such PCPs. We begin, in Section 3, by stating the two main lemmas corresponding to the two steps above and then proving the main theorem, assuming correctness of the lemmas. We then proceed to prove each main lemma. In Section 4, we show how to embed a PCP into one with linear structure (by routing it on a de Bruijn like graph). In Section 5, we prove the "derandomized parallel repetition" theorem for PCPs with linear structure. This is done by reduction to the derandomized direct product test of Impagliazzo et al. (2009). More accurately, our analysis relies on a specialized variant of this test which we call an S-test, which is analyzed in Section 9.

- ○ **Part 2.** The second part of the paper adapts our PCP construction to a dPCP. In Section 6, we discuss and define dPCPs and prove Theorem 1.6. We also show how to use this

theorem to derive the Moshkovitz & Raz (2008) result (Theorem 1.2) as a corollary. The two main steps in the proof of Theorem 1.6 are described in Section 7 and Section 8 and are analogous to the two main steps of proving Theorem 1.1.

○ Finally, we analyze the specialized direct product test (called the S-test) in Section 9, based on the work of Impagliazzo *et al.* (2009).

2. Preliminaries

Let $g : U \to \Sigma$ be an arbitrary function, and let $A \subset U$ be a subset. We denote by $g_{|A}$ the restriction of g (as a function) to A. We also use the following convention.

NOTATION 2.1. *Given two functions $f, g : U \to \Sigma$, we denote $f \stackrel{\alpha}{\approx} g$ ($f \stackrel{\alpha}{\not\approx} g$) to mean that they differ on at most (more than) α fraction of the elements of U.*

We refer to a d-dimensional linear subspace of an underlying vector space simply as a *d-subspace*. For two linear subspaces A_1 and A_2, the standard notation $A_1 + A_2$ denotes the smallest linear subspace containing both of them. We say that A_1, A_2 are *independent* if and only if $A_1 \cap A_2 = \{0\}$. If A_1 and A_2 are disjoint, the standard notation $A_1 \oplus A_2$ is used to denotes $A_1 + A_2$.

Let $G = (V, E)$ be a directed graph. For each edge $e \in E$, we denote by left (e) and right (e) the left and right endpoints of e, respectively. That is, if we view the edge $e \in E$ as a pair in $V \times V$, then left (e) and right (e) are the first and second elements of the pair e, respectively. Given a set of edges $E_0 \subseteq E$, we denote by left (E_0) and right(E_0) the set of left endpoints and right endpoints of the edges in E_0, respectively.

2.1. Direct Product Testing Impagliazzo *et al.* (2009).
Let us briefly describe the setting in which we use the derandomized direct product test of Impagliazzo *et al.* (2009). In Impagliazzo *et al.* (2009) the main derandomized direct product test is a

> 1. Choose a uniformly distributed d_1-subspace $B \subseteq \mathbb{F}^m$.
>
> 2. Choose a uniformly distributed d_0-subspace $A \subseteq B$.
>
> 3. Accept if and only if $\Pi(B)_{|A} = \Pi(A)$.

Figure 2.1: The P-test.

so-called V-test. We consider a variation of this test that appears in Impagliazzo *et al.* (2009), Section 6.3, to which we refer as the "P-test" (P for projection).

Given a string $\pi \in \Sigma^\ell$, we define its (derandomized) P-direct product Π as follows: We identify $[\ell]$ with $\mathbb{F}^m$, where $\mathbb{F}$ is a finite field and $m \in \mathbb{N}$, and think of π as an assignment that maps the points in $\mathbb{F}^m$ to Σ. We also fix $d_0 < d_1 \in \mathbb{N}$. Now, we define Π to be the assignment that assigns each d_0- and d_1-subspace W of $\mathbb{F}^m$ to the function $\pi_{|W} : W \to \Sigma$ (recall that $\pi_{|W}$ is the restriction of π to W).

We now consider the task of testing whether a given assignment Π is the P-direct product of some string $\pi : \mathbb{F}^m \to \Sigma$. In those settings, we are given an assignment to subspaces, i.e., a function Π that on input a d_0-subspace $A \subset \mathbb{F}^m$ (respectively, d_1-subspace $B \subset \mathbb{F}^m$), answers with a function $a : A \to \Sigma$ (respectively, $b : \mathbb{F}^m \to \Sigma$). We wish to test whether Π is a P-direct product of some $\pi : \mathbb{F}^m \to \Sigma$, and to this end, we invoke the P-test, described in Figure 2.1.

It is easy to see that if Π is a P-direct product, then the P-test always accepts. Furthermore, it can be shown that if Π is "far" from being a P-direct product, then the P-test rejects with high probability. Formally, we have the following result.

THEOREM 2.2 *(Soundness of the P-test of Impagliazzo et al.* 2009*). There exists a universal constant $h \in \mathbb{N}$ such that the following holds: Let $\varepsilon \geq h \cdot d_0 \cdot |\mathbb{F}|^{-d_0/h}, \alpha \stackrel{\text{def}}{=} h \cdot d_0 \cdot |\mathbb{F}|^{-d_0/h}$. Assume that $d_1 \geq h \cdot d_0, m \geq h \cdot d_1$. Suppose that an assignment Π passes the P-test with probability at least ε. Then, there exists an assignment π such that*

$$\Pr\left[\Pi\left(B\right)_{|A}=\Pi\left(A\right) \text{ and } \Pi\left(B\right)\overset{\alpha}{\approx}\pi_{|B} \text{ and } \Pi\left(A\right)\overset{\alpha}{\approx}\pi_{|A}\right]=\Omega(\varepsilon^4),$$
(2.3)

where the probability is over A, B chosen as in the P-test.

Theorem 2.2 can be proved by adapting the analysis of Impagliazzo *et al.* (2009) (in particular, Sections 3.4 and 4) to the setting of the P-test while relying on a lemma of Impagliazzo *et al.* (2009). For completeness, the proof is given in Section 9.3.

Working with randomized assignments. As observed by Impagliazzo *et al.* (2009), Theorem 2.2 works in even stronger settings. Suppose that Π is a randomized function, i.e., a function of both its input and some additional randomness. Then, Theorem 2.2 still holds for Π, where the probability in (2.3) is both over the choice of A and B *and over the internal randomness of* Π. We will rely on this fact in a crucial way in this work.

2.2. Sampling Tools. The following is the standard definition of a sampler, stated in the terminology of graphs, see e.g., Impagliazzo *et al.* (2008).

DEFINITION 2.4 (Sampler Graph). *A bipartite graph* $G = (L, R, E)$ *is said to be an* (ε, δ)-*sampler if, for every function* $f : L \to [0, 1]$, *there are at most* $\delta|R|$ *vertices* $u \in R$ *for which*

$$\left|\mathbb{E}_{v\in N(u)}[f(v)] - \mathbb{E}_{v\in L}[f(v)]\right| > \varepsilon.$$

Observe that if G *is an* (ε, δ)-*sampler, and if* $F \subset L$, *then by considering the function* $f \equiv 1_F$, *we get that there are at most* $\delta|R|$ *vertices* $u \in R$ *for which*

$$\left|\Pr_{v\in N(u)}[v \in F] - \Pr_{v\in L}[v \in F]\right| > \varepsilon.$$

The following lemma is stated in Impagliazzo *et al.* (2009), Lemma 2.2, and is proved implicitly in Impagliazzo *et al.* (2008), Lemma 2.9. For completeness, we include its proof.

LEMMA 2.5 (Subspace-point sampler of Impagliazzo *et al.* 2008).
*Let $d' < d$ be natural numbers, let V be a linear space over a
finite field $\mathbb{F}$, and let W be a fixed d'-subspace of V. Let G be the
bipartite graph whose left vertices are all points of V and whose
right vertices are all d-subspaces of V that contain W. We place
an edge between a d-subspace X and $x \in V$ if and only if $x \in X$.
Then, G is an $(\tau + \frac{1}{|\mathbb{F}|^{d-d'}}, \frac{1}{|\mathbb{F}|^{d-d'-2}\cdot\tau^2})$-sampler for every $\tau > 0$.*

PROOF.   Fix a function $f : V \to [0,1]$. We show that for a uni-
formly distributed d-subspace $X \subseteq V$ that contains W, it holds
with probability at least $1 - \frac{1}{|\mathbb{F}|^{d-d'-2}\cdot\tau^2}$ that

$$|\mathbb{E}_{x \in X}[f(x)] - \mathbb{E}_{v \in V}[f(v)]| \leq \tau + \frac{1}{|\mathbb{F}|^{d-d'}}.$$

Let $\overline{W}$ be a fixed subspace of V for which $V = W \oplus \overline{W}$. Let
$f_W : \overline{W} \to [0,1]$ be the function that maps each vector $\overline{w}$ of $\overline{W}$
to $\mathbb{E}_{v \in \overline{w}+W}[f(v)]$, and observe that $\mathbb{E}_{v \in V}[f(v)] = \mathbb{E}_{\overline{w} \in \overline{W}}[f_W(\overline{w})]$.
Furthermore, observe that every d-subspace X that contains W
can be written as $X = W \oplus U$ where U is a $(d - d')$-subspace of $\overline{W}$
and, moreover, that $\mathbb{E}_{x \in X}[f(x)] = \mathbb{E}_{u \in U}[f_W(u)]$. Thus, it suffices
to prove that for a uniformly distributed $(d - d')$-subspace U of
$\overline{W}$, it holds with probability at least $1 - \frac{1}{|\mathbb{F}|^{d-d'-2}\cdot\tau^2}$ that

$$(2.6) \qquad |\mathbb{E}_{u \in U}[f_W(u)] - \mathbb{E}_{\overline{w} \in \overline{W}}[f_W(\overline{w})]| \leq \tau + \frac{1}{|\mathbb{F}|^{d-d'}}.$$

To that end, let U be a uniformly distributed $(d - d')$-subspace of
$\overline{W}$. Let S_1 be a uniformly distributed set of $Q \overset{\text{def}}{=} \frac{|\mathbb{F}|^{d-d'}-1}{|\mathbb{F}|-1}$ vectors
of U such that every two vectors in S_1 are linearly independent.[3]
For every $\alpha \in \mathbb{F}^*$, let S_α be the set obtained by multiplying every
vector in S_1 by α. Observe that all the sets S_α have the property
that every two vectors in S_α are linearly independent and that the

---

[3]Such a set can be sampled, for example, by iteratively choosing a uni-
formly distributed vector of U that is linearly independent from each of the
previously chosen vectors individually. It is not hard to see that such a process
will halt after choosing $Q \overset{\text{def}}{=} \frac{|\mathbb{F}|^{d-d'}-1}{|\mathbb{F}|-1}$ vectors.

sets S_α form a partition of $U \backslash \{0\}$. We will show that for every $\alpha \in \mathbb{F}^*$, it holds with probability at least $1 - \frac{1}{|\mathbb{F}|^{d-d'-1} \cdot \tau^2}$ that

$$|\mathbb{E}_{u \in S_\alpha} [f_W(u)] - \mathbb{E}_{\overline{w} \in \overline{W}} [f_W(\overline{w})]| \leq \tau,$$

and the required result will follow by taking the union bound over all $\alpha \in \mathbb{F}^*$ and by noting that the vector 0 contributes at most $\frac{1}{|\mathbb{F}|^{d-d'}}$ to the difference in Inequality (2.6).

Fix $\alpha \in \mathbb{F}^*$, and let $s_1, \ldots, s_Q$ be the vectors in S_α. It is a known fact that $s_1, \ldots, s_Q$ are pairwise independent and uniformly distributed vectors of $\overline{W}$ (over the random choice of U). This implies that $f_W(s_1), \ldots, f_W(s_Q)$ are pairwise independent random variables with expectation $\mathbb{E}_{\overline{w} \in \overline{W}} [f_W(\overline{w})]$ and therefore by the Chebyshev inequality it follows that

$$\Pr \left[\left| \frac{1}{Q} \sum_{i=1}^{Q} f_W(s_i) - \mathbb{E}_{\overline{w} \in \overline{W}} [f_W(\overline{w})] \right| > \tau \right] \leq \frac{1}{Q \cdot \tau^2} \leq \frac{1}{|\mathbb{F}|^{d-d'-1} \cdot \tau^2},$$

as required. $\square$

2.3. Constraint Graphs and PCPs. As discussed in the introduction, the focus of this work is on claims that can be verified by reading a small number of symbols of the proof. A PCP system for a language L is an oracle machine M, called a verifier, that has oracle access to a proof π over an alphabet Σ. The verifier M reads the input x, tosses r coins, makes at most q "oracle" queries into π, and then accepts or rejects. If x is in the language, then it is required that M accepts with probability 1 for some π, and otherwise, it is required that M accepts with probability at most ε for every π. More formally:

DEFINITION 2.7. *Let $r, q : \mathbb{N} \to \mathbb{N}$, and let Σ be a function that maps the natural numbers to finite alphabets. A $(r,q)_\Sigma$-PCP verifier M is a probabilistic polynomial time oracle machine that when given input $x \in \{0,1\}^*$, tosses at most $r(|x|)$ coins, makes at most $q(|x|)$ non-adaptive queries to an oracle that is a string over $\Sigma(|x|)$, and outputs either "accept" or "reject". We refer to r, q, and Σ as the randomness complexity, query complexity, and proof alphabet of the verifier, respectively.*

REMARK 2.8. *Note that for an $(r, q)_\Sigma$-PCP verifier M and an input x, we can assume without loss of generality that the oracle is a string of length at most $2^{r(|x|)} \cdot q(|x|)$, since this is the maximal number of different queries that M can make. Hence, it is unnecessary to keep track of the proof length of the verifier.*

DEFINITION 2.9. *Let r, q, and Σ be as in Definition 2.7, let $L \subseteq \{0, 1\}^*$ and let $\varepsilon : \mathbb{N} \to [0, 1)$. We say that $L \in \mathbf{PCP}_{\varepsilon, \Sigma}[r, q]$ if there exists an $(r, q)_\Sigma$-PCP verifier M that satisfies the following requirements:*

○ **Completeness:** *For every $x \in L$, there exists $\pi \in \Sigma(|x|)^*$ such that $\Pr[M^\pi(x) \text{ accepts}] = 1$.*

○ **Soundness:** *For every $x \notin L$ and for every $\pi \in \Sigma(|x|)^*$, it holds that $\Pr[M^\pi(x) \text{ accepts}] \leq \varepsilon(|x|)$.*

One possible formulation of the PCP theorem is as follows.

THEOREM 2.10 *(PCP Theorem Arora & Safra 1998; Arora et al. 1998). There exist universal constant $\varepsilon \in (0, 1)$ and a finite alphabet Σ such that $\mathbf{NP} \subseteq \mathbf{PCP}_{\varepsilon, \Sigma}[O(\log n), 2]$.*

PCPs that have query complexity 2 correspond to graphs in a natural way: Consider the action of an $(r, 2)_\Sigma$-verifier M on some fixed string x, and let $r \stackrel{\text{def}}{=} r(|x|), \Sigma \stackrel{\text{def}}{=} \Sigma(|x|)$. The verifier M is given access to some proof string π of length ℓ and may make 2^r possible tests on this string, where each such test consists of making two queries to π and deciding according to the answers. We now view the action of M as a graph in the following way. We consider the graph G whose vertices are the coordinates in $[\ell]$ and that has an edge for each possible test of the verifier M. The endpoints of an edge e of G are the coordinates that are queried by M in the test that corresponds to e. We also associate an edge e with a constraint $c_e \in \Sigma \times \Sigma$, which contains all the pairs of answers that make M accept when performing the test that corresponds to e. We think of π as an assignment that assigns the vertices of G values in Σ, and say that π *satisfies* an edge (u, v) if $(\pi(u), \pi(v)) \in c_{(u,v)}$. If $x \in L$, then it is required that there exists some assignment π

that satisfies all the edges of G, and otherwise, it is required that every assignment satisfies at most ε fraction of the edges. This correspondence is called the FGLSS correspondence (Feige *et al.* 1996). We turn to state it formally:

DEFINITION 2.11 (Constraint graph).   *A (directed) constraint graph is a directed graph $G = (V, E)$ together with an alphabet Σ, and, for each edge $(u, v) \in E$, a binary constraint $c_{u,v} \subseteq \Sigma \times \Sigma$. The size of G is the number of edges of G. The graph is said to have* projection constraints *if it is bipartite with all the edges directed from the left to the right, and every constraint $c_{u,v}$ has an associated function $f_{u,v} : \Sigma \to \Sigma$ such that $c_{u,v}$ is satisfied by (a, b) if and only if $f_{u,v}(a) = b$.*

Given an assignment $\pi : V \to \Sigma$, we define

$$\mathrm{SAT}(G, \pi) = \Pr_{(u,v) \in E}[(\pi(u), \pi(v)) \in c_{u,v}] \quad \text{and}$$

$$\mathrm{SAT}(G) = \max_{\pi}(\mathrm{SAT}(G, \pi)).$$

We also denote $\mathrm{UNSAT}(G, \pi) = 1 - \mathrm{SAT}(G, \pi)$ *and similarly* $\mathrm{UNSAT}(G) = 1 - \mathrm{SAT}(G)$.

REMARK 2.12. *Note that Definition 2.11 uses directed graphs, while the common definition of constraint graphs refers to undirected graphs.*

REMARK 2.13. *Note that if the graph G has projection constraints, then this is simply a label cover instance with projection constraints (Arora & Lund 1996).*

PROPOSITION 2.14 (FGLSS correspondence, Feige *et al.* 1996). *The following two statements are equivalent:*

○ $L \in \mathbf{PCP}_{\varepsilon, \Sigma}[r, 2]$.

○ *There exists a polynomial time algorithm that transforms strings $x \in \{0, 1\}^*$ to constraint graphs G_x of size $2^{r(|x|)}$ with alphabet $\Sigma(|x|)$ such that: (1) if $x \in L$, then $\mathrm{SAT}(G_x) = 1$ and (2) if $x \notin L$, then $\mathrm{SAT}(G_x) \leq \varepsilon$.*

Given a PCP system for L, we refer to the corresponding family of graphs $\{G_x\}$ where x ranges over all possible instances as its underlying graph family. *If the graphs $\{G_x\}$ have projection constraints then we say that the PCP system has the* projection property.

Using the FGLSS correspondence, we can rephrase the PCP theorem in the terminology of constraint graphs:

THEOREM 2.15 (PCP Theorem for constraint graphs).   *There exist universal constant $\varepsilon \in (0,1)$ and a finite alphabet Σ such that for every language $L \in$ **NP** the following holds: There exists a polynomial time reduction that on input $x \in \{0,1\}^*$, outputs a constraint graph G_x such that if $x \in L$ then $\mathrm{SAT}(G_x) = 1$ and otherwise $\mathrm{SAT}(G_x) \leq \varepsilon$.*

REMARK 2.16. *The connection between PCPs and approximation problems (such as Proposition 2.14) was discovered by Feige et al. (1996). However, the precise correspondence between PCPs and constraint graphs that is given in Proposition 2.14 was only stated for the first time by Arora et al. (1998). Still, in the rest of this paper, we refer to Proposition 2.14 as the FGLSS correspondence.*

REMARK 2.17. *Note the tight relationship between the randomness complexity of the PCP and the size of the corresponding constraint graphs. In particular, observe that PCP verifiers with randomness complexity $O(\log n)$ correspond to constraint graphs of polynomial size. This relationship is one of the main reasons for the study of the randomness complexity of PCP verifiers.*

Moreover, recall that the work of Moshkovitz & Raz (2008) constructs PCPs that are very randomness efficient, i.e., have randomness complexity $(1 + o(1)) \log n$ (see also Remark 6.27). This randomness efficiency is translated into constraints graphs of almost-linear size, namely $n^{1+o(1)}$.

2.4. Basic Facts About Random Subspaces.   In this section, we present two useful propositions about random subspaces. The following proposition says that a uniformly distributed subspace is independent from every fixed subspace with high probability.

PROPOSITION 2.18. *Let $d, d' \in \mathbb{N}$ such that $d > 2d'$, and let V be a d-dimensional space. Let W_1 be a uniformly distributed d'-subspace of V, and let W_2 be a fixed d'-subspace of V. Then,*

$$\Pr[W_1 \cap W_2 = \{0\}] \geq 1 - 2 \cdot d'/\left|\mathbb{F}\right|^{d-2 \cdot d'}.$$

PROOF. Suppose that W_1 is chosen by choosing random basis vectors $v_1, \ldots, v_{d'}$ one after the other. It is easy to see that $W_1 \cap W_2 \neq \{0\}$ only if $v_i \in \mathrm{span}\left(W_2 \cup \{v_1, \ldots, v_{i-1}\}\right)$ for some $i \in [d']$. For each fixed i, the vector v_i is uniformly distributed in $V \backslash \mathrm{span}\{v_1, \ldots, v_{i-1}\}$, and therefore, the probability that $v_i \in \mathrm{span}\left(W_2 \cup \{v_1, \ldots, v_{i-1}\}\right)$ for a fixed i is at most

$$\frac{\left|\mathrm{span}\left(W_2 \cup \{v_1, \ldots, v_{i-1}\}\right)\right|}{\left|V \backslash \mathrm{span}\{v_1, \ldots, v_{i-1}\}\right|} = \frac{\left|\mathbb{F}\right|^{d'+i-1}}{\left|\mathbb{F}\right|^d - \left|\mathbb{F}\right|^{i-1}}$$

$$\leq \frac{2 \cdot \left|\mathbb{F}\right|^{d'+i-1}}{\left|\mathbb{F}\right|^d}$$

$$\leq \frac{2 \cdot \left|\mathbb{F}\right|^{2 \cdot d'-1}}{\left|\mathbb{F}\right|^d}$$

(2.19) $$\leq \frac{2}{\left|\mathbb{F}\right|^{d-2 \cdot d'}},$$

where Inequality (2.19) can be observed by noting that $\left|\mathbb{F}\right|^{i-1} \leq \left|\mathbb{F}\right|^{d-1} \leq \frac{1}{2} \cdot \left|\mathbb{F}\right|^d$. By the union bound, the probability that this event occurs for some $i \in [d']$ is at most $\frac{2 \cdot d'}{\left|\mathbb{F}\right|^{d-2 \cdot d'}}$. It follows that the probability that $W_1 \cap W_2 \neq \{0\}$ is at most $\frac{2 \cdot d'}{\left|\mathbb{F}\right|^{d-2 \cdot d'}}$ as required. $\square$

The following proposition says that the span of d' uniformly distributed vectors is with high probability a uniformly distributed d'-subspace.

PROPOSITION 2.20. *Let V be a d-dimensional space over a finite field $\mathbb{F}$, let $w_1, \ldots, w_{d'}$ be independent and uniformly distributed vectors of V, and let $W = \mathrm{span}\{w_1, \ldots, w_{d'}\}$. Then, with probability at least $1 - d'/\left|\mathbb{F}\right|^{d-d'}$, it holds that $\dim W = d'$. Furthermore, conditioned on the latter event, W is a uniformly distributed d'-subspace of V.*

PROOF.   The fact that $\dim W = d'$ with probability at least $1 - d'/\left|\mathbb{F}\right|^{d-d'}$ can be proved in essentially the same way as Proposition 2.18. To see that conditioned on the latter event it holds that the subspace W is uniformly distributed, observe that since $w_1, \ldots, w_{d'}$ were originally chosen to be uniformly distributed, all the possible d'-sets of linearly independent vectors have the same probability to occur. □

Finally, the following proposition shows the equivalence of two different ways of choosing subspaces $A_1, A_2 \subseteq B$ where A_1 and A_2 are independent.

PROPOSITION. *Let V be a linear space over a finite field $\mathbb{F}$, and let $d_0, d_1 \in \mathbb{N}$ be such that $d_0 < d_1 < \dim V$. The following two distributions over d_0-subspaces A_1, A_2 and a d_1-subspace B are the same:*

(i) *Choose B to be a uniformly distributed d_1-subspace of V, and then choose A_1 and A_2 to be two uniformly distributed and independent d_0-subspaces of B.*

(ii) *Choose A_1 and A_2 to be two uniformly distributed and independent d_0-subspaces of V, and then choose B to be a uniformly distributed d_1-subspace of V that contains A_1 and A_2.*

PROOF.   Observe that choosing A_1, A_2, B under the first distribution amounts to choosing d_1 uniformly distributed and linearly independent vectors in V (those vectors will serve as the basis of B) and then choosing two disjoint subsets of those vectors to serve as the basis of A_1 and as the basis of A_2. On the other hand, choosing A_1, A_2, and B under the second distribution amounts to choosing d_0 uniformly distributed and linearly independent vectors in V to serve as the basis of A_1, then choosing another d_0 uniformly distributed and linearly independent vectors in V to serve as the basis of A_2 while making sure that this basis is also linearly independent from the basis of A_1, and then completing the basis of A_1 and the basis of A_2 to a basis of B. It is easy to see that those two distributions over a set of d_1 vectors and its two disjoint subsets are identical. □

2.5. Similarity of Distributions.   In this section, we introduce a notion of "similarity of distributions", which we will use in the second part of the paper. Let X_1 and X_2 be two random variables that take values from a set $\mathcal{X}$, and let $\gamma \in (0, 1]$. We say that X_1 and X_2 are γ-similar if for every $x \in \mathcal{X}$ it holds that

$$\gamma \cdot \Pr\left[X_1 = x\right] \leq \Pr\left[X_2 = x\right] \leq \frac{1}{\gamma} \cdot \Pr\left[X_1 = x\right].$$

Note that if X_1 and X_2 are γ-similar, then actually it holds for every $S \subseteq \mathcal{X}$ that

$$\gamma \cdot \Pr\left[X_1 \in S\right] \leq \Pr\left[X_2 \in S\right] \leq \frac{1}{\gamma} \cdot \Pr\left[X_1 \in S\right],$$

The following claim says roughly that if f is a randomized function, then the random variable $f(X_1)$ is γ-similar to $f(X_2)$.

CLAIM 2.21. *Let X_1 and X_2 be two random variables that take values from a set $\mathcal{X}$ that are γ-similar. Let Y_1 and Y_2 be two random variables that take values from a set $\mathcal{Y}$ such that for every $x \in \mathcal{X}, y \in \mathcal{Y}$, it holds that*

$$\Pr\left[Y_1 = y | X_1 = x\right] = \Pr\left[Y_2 = y | X_2 = x\right].$$

Then, the variables Y_1, Y_2 are γ-similar.

PROOF.   It holds that

$$\begin{aligned}
\Pr\left[Y_1 = y\right] &= \sum_{x \in \mathcal{X}} \Pr\left[Y_1 = y | X_1 = x\right] \cdot \Pr\left[X_1 = x\right] \\
&= \sum_{x \in \mathcal{X}} \Pr\left[Y_2 = y | X_2 = x\right] \cdot \Pr\left[X_1 = x\right] \\
&\geq \sum_{x \in \mathcal{X}} \Pr\left[Y_2 = y | X_2 = x\right] \cdot \gamma \cdot \Pr\left[X_2 = x\right] \\
&= \gamma \cdot \Pr\left[Y_2 = y\right].
\end{aligned}$$

Similarly, it can be proved that $\Pr\left[Y_1 = y\right] \leq \frac{1}{\gamma} \cdot \Pr\left[Y_2 = y\right]$.   □

2.6. Expanders. Expanders are graphs with certain properties that make them extremely useful for many applications in theoretical computer science. Below, we give a definition of expanders that suits our needs.

DEFINITION 2.22. *Let $G = (V, E)$ be a d-regular graph. Let $E\left(S, \overline{S}\right)$ be the set of edges from a subset $S \subseteq V$ to its complement. We say that G has edge expansion h if for every $S \subseteq V$ such that $|S| \leq |V|/2$, it holds that*

$$\left|E(S, \overline{S})\right| \geq h \cdot d_0 \cdot |S|.$$

A useful fact is that there exist constant degree expanders over any number of vertices:

FACT 2.23. *There exist $d_0 \in \mathbb{N}$ and $h_0 > 0$ such that there exists a polynomial time constructable family $\{G_n\}_{n \in \mathbb{N}}$ of d_0-regular graphs G_n on n vertices that have edge expansion h_0 (such graphs are called expanders).*

3. Main Theorem

In this section, we prove our main PCP theorem (Theorem 1.1), which asserts the existence of two-query PCPs with soundness error $\varepsilon(n)$ for any function $1/n^{\kappa} \leq \varepsilon(n) \leq 1/\text{poly}\log n$. To that end, we use the PCP theorem for graphs (Theorem 2.15) to reduce the problem of deciding membership of a string x in the language L to the problem of checking the satisfiability of a constraint graph with constant soundness error. We then show that every constraint graph can be transformed into one that has "linear structure", defined shortly below. This is done in Lemma 3.3, which directly proves Theorem 1.4 (the existence of PCPs with linear structure). Finally, in Lemma 3.4, we prove a derandomized parallel repetition theorem for constraint graphs with linear structure. Theorem 1.1 follows by combining the two lemmas. We begin by defining the notion of a graph with linear structure.

DEFINITION 3.1 (Linear Structure). *We say that a directed graph G has a linear structure if it satisfies the following conditions:*

(i) The vertices of G can be identified with the linear space $\mathbb{F}^m$, where $\mathbb{F}$ is a finite field and $m \in \mathbb{N}$.

(ii) We identify the set of pairs of vertices $(\mathbb{F}^m)^2$ with the linear space $\mathbb{F}^{2m}$. Using this identification, the edges E of G are required to form a linear subspace of $\mathbb{F}^{2m}$.

(iii) We require that $\text{left}(E) = \text{right}(E) = \mathbb{F}^m$. In other words, this means that every vertex of G is both the left endpoint of some edge and the right point of some edge.

REMARK 3.2. *We mention that although it is not required by Definition 3.1, a graph with linear structure must be regular, i.e., all the vertices in the graph have the same in-degree and out-degree. This is a straightforward corollary of Items (ii) and (iii) of the definition.*

The following lemmas are proved in Section 4 and Section 5, respectively.

LEMMA 3.3 (Linear Structure Embedding). *There exists a polynomial time procedure that satisfies the following requirements:*

○ **Input:**

    – *A constraint graph G of size n over alphabet Σ.*

    – *A finite field $\mathbb{F}$ of size q.*

○ **Output:** *A constraint graph $G' = (\mathbb{F}^m, E')$ such that the following holds:*

    – *G' has a linear structure.*

    – *The size of G' is at most $O(q^2 \cdot n)$.*

    – *G' has alphabet $\Sigma^{O(\log_q(n))}$.*

    – *If G is satisfiable then G' is satisfiable.*

    – *If $\text{UNSAT}(G) \geq \rho$ then $\text{UNSAT}(G') \geq \Omega\left(\frac{1}{q \cdot \log_q(n)} \cdot \rho\right)$.*

LEMMA 3.4 (Derandomized Parallel Repetition). *There exist a universal constant h and a polynomial time procedure that satisfy the following requirements:*

∘ **Input:**

- *A finite field $\mathbb{F}$ of size q*

- *A constraint graph $G = (\mathbb{F}^m, E)$ over alphabet Σ that has a linear structure.*

- *A parameter $d_0 \in \mathbb{N}$ such that $d_0 < m/h^2$. This parameter will determine the dimension of linear subspaces used in the derandomized parallel repetition and thus together with q will determine the number of repetitions used in the derandomized parallel repetition.*

- *A parameter $\rho \in (0, 1)$ such that $\rho \geq h \cdot d_0 \cdot q^{-d_0/h}$. Intuitively, the parameter ρ should be chosen such that $1 - \rho$ is an upper bound on the soundness error of G.*

∘ **Output:** *A constraint graph G' such that the following holds:*

- *G' has size $n^{O(d_0)}$.*

- *G' has alphabet $\Sigma^{q^{O(d_0)}}$.*

- *If G is satisfiable then G' is satisfiable.*

- *If $\mathrm{SAT}(G) < 1 - \rho$ then $\mathrm{SAT}(G') < h \cdot d_0 \cdot q^{-d_0/h}$.*

- *G' has the projection property.*

We turn to prove the main theorem from the above lemmas.

THEOREM (1.1, restated). *There exists a constant $\kappa > 0$ such that for every function $\varepsilon : \mathbb{N} \to (0, 1)$ satisfying $1/n^\kappa \leq \varepsilon(n) \leq 1/\mathrm{poly}\log n$ the following holds: Every language $L \in \mathbf{NP}$ has a two-query PCP system with perfect completeness, soundness error ε, alphabet size $2^{1/\mathrm{poly}(\varepsilon)}$, proof length $\mathrm{poly}(n)$, and randomness complexity $O(\log n)$. Furthermore, the verifier in this PCP system makes only 'projection' queries.*

PROOF.   Let $\kappa > 0$ be a constant to be chosen later, and let $\varepsilon : \mathbb{N} \to (0,1)$ be a function satisfying $1/n^\kappa \leq \varepsilon(n) \leq 1/\text{poly} \log n$. Fix a language $L \in \mathbf{NP}$. We show that L has a two-query PCP system with perfect completeness, soundness error $\varepsilon(n)$, and alphabet size $2^{1/\text{poly}(\varepsilon)}$, which has the projection property. By the FGLSS correspondence (Proposition 2.14), it suffices to show a polynomial time procedure that on input $x \in \{0,1\}^*$, outputs a constraint graph G' of size $\text{poly}\,(n)$ such that the following holds: If $x \in L$, then G' is satisfiable (i.e., $\text{SAT}(G') = 1$), and if $x \notin L$, then $\text{SAT}(G') \leq \varepsilon(n)$. The procedure begins by transforming x, using the PCP theorem for constraint graphs (Theorem 2.15), to a constraint graph G of size $n = \text{poly}\,|x|$ such that if $x \in L$, then $\text{SAT}(G) = 1$ and if $x \notin L$, then $\text{SAT}(G) \leq \varepsilon_0$, where $\varepsilon_0 \in (0,1)$ is a universal constant that does not depend on x. Let $n = \text{poly}\,(|x|)$ be the size of G, and let $\rho_0 = 1 - \varepsilon_0$.

Next, the procedure sets $\mathbb{F}$ to be the smallest field of size at least $1/\left(\varepsilon(n)\right)^c$ for some constant $c > 1$ to be determined later, and sets $q = |\mathbb{F}|$. Note that $q \geq \text{poly} \log n$. The procedure now invokes Lemma 3.3 (linear structure embedding) on input G and $\mathbb{F}$, thus obtaining a new constraint graph G_1. Note that by Lemma 3.3, if $\text{UNSAT}(G) \geq \rho_0$, then $\rho_1 \stackrel{\text{def}}{=} \text{UNSAT}(G_1) \geq \Omega\left(\frac{1}{q \cdot \log_q(n)} \cdot \rho_0\right)$.

Finally, the procedure sets d_0 to be an arbitrary constant such that $\rho_1 \geq h \cdot d_0 \cdot q^{-d_0/h}$. Note that this is indeed possible, since $\log_q(1/\rho_1)$ is a constant that depends only on ρ (here, we use the fact that $q \geq \text{poly} \log n$). Finally, the procedure invokes Lemma 3.4 (derandomized parallel repetition) on input $G_1, \mathbb{F}, \rho_1$, and d_0, and outputs the resulting constraint graph G'. We note that we use here the assumption that $\varepsilon(n) \geq n^\kappa$ and choose κ to be sufficiently small, in order to guarantee that q, m, and d_0 satisfy the requirements of Lemma 3.4.

It remains to analyze the parameters of G'. It is not hard to see that G' has size $n^{O(d_0)}$ and alphabet $\Sigma^{q^{O(d_0)}} = \Sigma^{1/\text{poly}(\varepsilon)}$. Furthermore, if $\text{UNSAT}(G) \geq \rho$, then $\text{UNSAT}(G_1) \geq \rho_1$. Therefore, by Lemma 3.4 and by the choice of d_0, it holds that $\text{SAT}(G') \leq O(1/q^{\Omega(1)})$. Since $q = 1/\left(\varepsilon(n)\right)^c$, it holds for sufficiently large c that $\text{SAT}(G') \leq \varepsilon(n)$, as required. $\square$

REMARK 3.5. *Recall that* Moshkovitz & Raz (2008) *prove a stronger version of the main theorem, saying that for every soundness error $\varepsilon(n) > n^\kappa$, not necessarily upper bounded by $1/\mathrm{poly}\log n$, it holds that* **NP** *has a PCP system with soundness ε and alphabet size $\exp(1/\mathrm{poly}(\varepsilon))$ (Theorem 1.2). If one could prove a stronger version of* Lemma 3.3 *(Linear Structure Embedding) in which the soundness of G' is $\rho/\mathrm{poly}(q)$ and the alphabet size is $|\Sigma|^{\mathrm{poly}(q)}$, then the stronger* Theorem 1.2 *would follow using the same proof as above, without using a composition technique as in* Dinur & Harsha (2009); Moshkovitz & Raz (2008), *by choosing q to be sufficiently small.*

REMARK 3.6. *The reduction described in* Theorem 1.1 *yields graphs of polynomial size, but not of nearly-linear size as in* Moshkovitz & Raz (2008) *(see* Remark 2.8*). In fact, the construction of graphs with linear structure (*Lemma 3.3*) is nearly-linear size (taking an instance of size n to an instance of size $q^2 \cdot n$). The part that incurs a polynomial and not nearly-linear blow-up is the derandomized parallel repetition (*Lemma 3.4*) that relies on the derandomized direct product. It is possible that a more efficient derandomized direct product may lead to a nearly-linear size construction in total.*

4. PCPs with Linear Structure

In this section, we prove Lemma 3.3 (linear structure embedding), which implies Theorem 1.4 (the existence of PCPs with linear structure) by combining it with the PCP theorem (Theorem 2.15). The lemma which says that every constraint graph can be transformed into one that has linear structure. To this end, we use a family of structured graphs called de-Bruijn graphs. We show that de-Bruijn graphs have linear structure and that every constraint graph can be embedded in some sense on a de-Bruijn graph. This embedding technique is a variant of a technique introduced by Babai *et al.* (1991) and Polishchuk & Spielman (1994) for embedding circuits on de-Bruijn graphs. We begin by defining de-Bruijn graphs.

DEFINITION 4.1. *Let Λ be a finite alphabet and let $m \in \mathbb{N}$. The de Bruijn graph $\mathcal{DB}_{\Lambda,m}$ is the directed graph whose vertices set is Λ^m such that each vertex $(\alpha_1, \ldots, \alpha_m) \in \Lambda^m$ has outgoing edges to all the vertices of the form $(\alpha_2, \ldots, \alpha_m, \beta)$ for $\beta \in \Lambda$.*

REMARK 4.2. *We note that previous works used a slightly different notion, the "wrapped de Bruijn graph", which is a layered graph in which the edges between layers are connected as in the de Bruijn graph. Also, we note that previous works fixed Λ to be the binary alphabet, while we use a general alphabet.*

Lemma 3.3 follows easily from the following two propositions. Proposition 4.3 says that de Bruijn graphs have linear structure. Proposition 4.4 says that any constraint graph can be embedded on a de Bruijn graph.

PROPOSITION 4.3. *Let $\mathbb{F}$ be a finite field and let $m \in \mathbb{N}$. Then, the de Bruijn graph $\mathcal{DB}_{\mathbb{F},m}$ has linear structure.*

PROOF. Items i and iii of the definition of linear structure (Definition 3.1) follow immediately from the definition of de Bruijn graphs. To see that Item ii holds, observe that in order for a tuple in $\mathbb{F}^{2m}$ to be an edge of $\mathcal{DB}_{\mathbb{F},m}$, it only needs to satisfy equality constraints, which are in turn linear constraints. Thus, the set of edges of $\mathcal{DB}_{\mathbb{F},m}$ form a linear subspace of $\mathbb{F}^{2m}$. □

PROPOSITION 4.4 (Embedding on de-Bruijn graphs). *There exists a polynomial time procedure that satisfies the following requirements:*

○ **Input:**

   – *A constraint graph G of size n over alphabet Σ.*
   – *A finite alphabet Λ.*
   – *A natural number m such that $|\Lambda|^m \geq 2 \cdot n$*

○ **Output:** *A constraint graph G' such that the following holds:*

   – *The underlying graph of G' is the de Bruijn graph $\mathcal{DB}_{\Lambda,m}$.*

- *The size of G' is $|\Lambda|^{m+1}$.*
- *G' has alphabet $\Sigma^{O(m)}$.*
- *If G is satisfiable, then G' is satisfiable.*
- *If $\mathrm{UNSAT}\,(G) \geq \rho$ then $\mathrm{UNSAT}\,(G') \geq \Omega\left(\frac{n}{|\Lambda|^{m+1}\cdot m} \cdot \rho\right).$*

Lemma 3.3 (linear structure embedding) is obtained by invoking Proposition 4.4 with $\Lambda = \mathbb{F}, m = \lceil \log_q (2 \cdot n) \rceil$ and combining it with Proposition 4.3. The rest of this section is devoted to proving Proposition 4.4 and is organized as follows: In Section 4.1, we give the required background on the routing properties of de Bruijn graphs. Then, in Section 4.2, we give an outline of the proof of Proposition 4.4. Finally, we give the full proof of the proposition in Section 4.3.

4.1. de Bruijn Graphs as Routing Networks. The crucial property of the de Bruijn graphs that we use is that the de Bruijn graph is a permutation routing network. To explain the intuition that underlies this notion, let us think of the vertices of the de Bruijn graph as computers in a network, such that two computers can communicate if and only if they are connected by an edge. Furthermore, sending a message from a computer to its neighbor takes one unit of time. Suppose that each computer in the network wishes to send a message to some other computer in the network, and furthermore, each computer needs to receive a message from exactly one computer (that is, the mapping from source computers to target computers is a permutation). Then, the routing property of the de Bruijn network says that we can find paths in the network that have the following properties:

1. Each path corresponds to a message that needs to be sent and goes from the message's source computer to its target computer.

2. If all the messages are sent simultaneously along their corresponding paths, then at each unit of time, each computer processes exactly one message. By "processing", we mean that the computer receives the message from one of its neighbors and sends it to one of its neighbors.

3. The paths are of length exactly $2 \cdot m$. This means that if all
   the messages are sent simultaneously along their corresponding
   paths, then after $2 \cdot m$ units of time all the messages will reach
   their destination.

Formally, this property can be stated as follows.

FACT 4.5. *Let $\mathcal{DB}_{\Lambda,m}$ be a de-Brujin graph. Then, given a permu-*
tation μ on the vertices of $\mathcal{DB}_{\Lambda,m}$, one can find a set of undirected
paths of length $l = 2m$ which connect each vertex v to $\mu(v)$ and
which have the following property: For every $j \in [l]$, each vertex v
is the j-th vertex of exactly one path. Furthermore, finding the
paths can be done in time that is polynomial in the size of $\mathcal{DB}_{\Lambda,m}$.

Fact 4.5 is proved in Leighton (1992) for the special case of $\Lambda = \{0, 1\}$. The proof of the general case essentially follows the original
proof, except that the looping algorithm of Beneš is replaced with
the decomposition of d-regular graphs to d perfect matchings. For
completeness, we give the proof of the general case in Section 4.4.

REMARK 4.6. *Note that the paths mentioned in Fact 4.5 are undi-*
rected. That is, if a vertex u appears immediately after a vertex v
in path, then either (u, v) or (v, u) are edges of $\mathcal{DB}_{\Lambda,m}$.

4.2. Proof Overview.   Suppose we are given as input a con-
straint graph G which we want to embed on $\mathcal{DB} = \mathcal{DB}_{\Lambda,m}$. Recall
that the size of G is at most $|\Lambda|^m$, so we may identify the vertices
of G with some of the vertices of $\mathcal{DB}$.

Handling degree 1   As a warm-up, assume that G has degree 1,
i.e., G is a perfect matching. In this case, we construct G' as fol-
lows. We choose the alphabet of G' to be Σ^l for $l \stackrel{\text{def}}{=} 2m$. Fix any
assignment π to G. We describe how to construct a corresponding
assignment π' to G'. We think of the vertices of G as computers,
such that each vertex v wants to send the value $\pi(v)$ as a message
to its unique neighbor in G. Using the routing property of the
de Bruijn graph, we find paths for routing those messages along
the edges of G'. Recall that if all the messages are sent simulta-
neously along those paths, then every computer has to deal with

one packet at each unit of time, for l units of time. We now define the assignment π' to assign each vertex v of G' a tuple in Σ^l whose j-th element is the message with which v deals at the j-th unit of time.

We define the constraints of G' such that they verify that the routing is done correctly. That is, if the computer u is supposed to send a message to a vertex v between the j-th unit of time and the $(j+1)$-th unit of time, then the constraint of the edge between u and v checks that $\pi'(u)_j = \pi'(v)_{j+1}$. Furthermore, for each edge (u, v) of G, the constraints of G' check that the values $\pi'(v)_l$ and $\pi'(v)_1$ satisfy the edge (u, v). This condition should hold because if π' was constructed correctly according to π, then $\pi'(v)_l = \pi(u)$ and $\pi'(v)_1 = \pi(v)$. It should be clear that the constraints of G' "simulate" the constraints of G. We discuss the exact behavior of the soundness error in the detailed proof.

Handling arbitrary degree graphs Using the expander replacement technique of Papadimitriou & Yannakakis (1991), we may assume that G is d-regular for some universal constant d. The d-regularity of G implies that the edges of G can be partitioned to d disjoint perfect matchings $\mu_1, \ldots, \mu_d$ in polynomial time (see, e.g., Cameron 1998, Proposition 18.1.2). Now, we set the alphabet of G' to be $\left(\Sigma^l\right)^d$ and handle each of the matchings μ_i as before, each time using a "different part" of the alphabet symbols. In other words, the alphabet of G' consists of d-tuples of Σ^l, and so the constraints used to handle each matching μ_i will refer to the i-th coordinates in those tuples. Finally, for vertex v, its constraints will also check that the message it sends in each of the d paths is the same. In other words, if $\pi'(v) = (\sigma_1, \ldots, \sigma_d) \in \left(\Sigma^l\right)^d$, then the constraints will check that $(\sigma_1)_1 = \cdots = (\sigma_d)_1$. As before, the constraints of resulting graph G' "simulate" the constraints of the original graph G.

REMARK 4.7. *Observe that the foregoing proof used only the routing property of de Bruijn graphs and will work for any graph that satisfies this property. In other words, Proposition 4.4 (embedding on de-Bruijn graphs) holds for any graph for which Fact 4.5 holds.*

4.3. Detailed Proof. We use the following version of the expander-replacement technique of Papadimitriou & Yannakakis (1991).

LEMMA 4.8 (Dinur 2007, Lemma 3.2). *There exist universal constants $c, d \in \mathbb{N}$ and a polynomial time procedure that when given as input a constraint graph G of size n outputs a constraint graph G' of size $2 \cdot d \cdot n$ over alphabet Σ such that the following holds:*

○ *G' has $2 \cdot n$ vertices and is d-regular.*

○ *If G is satisfiable, then so is G'.*

○ *If $\mathrm{UNSAT}(G) \geq \rho$ then $\mathrm{UNSAT}(G') \geq \rho/c$.*

We turn to proving Proposition 4.4 (embedding on de-Bruijn graphs). When given as input a constraint graph G, a finite alphabet Λ, and a natural number m such that $|\Lambda^m| \geq 2 \cdot n$, the procedure of Proposition 4.4 acts as follows. The procedure begins by invoking Lemma 4.8 on G, resulting in a d-regular constraint graph G_1 over $2 \cdot n$ vertices. Then, the vertices of G_1 are identified with a subset of the vertices of $\mathcal{DB} = \mathcal{DB}_{\Lambda,m}$ (note that this is possible since $|\Lambda^m| \geq 2 \cdot n$).

Next, the procedure partitions the edges of G_1 to d disjoint perfect matchings and views those matchings as permutations $\mu_1, \ldots, \mu_d$ on the vertices of $\mathcal{DB}$ in the following way: Given a vertex v of $\mathcal{DB}$, if v is identified with a vertex of G_1, then μ_i maps v to its unique neighbor in G via the i-th matching, and otherwise μ_i maps v to itself. The procedure then applies Fact 4.5 to each permutation μ_i resulting in a set of paths $\mathcal{P}_i$ of length $l \stackrel{\mathrm{def}}{=} 2m$. Let $\mathcal{P} = \bigcup \mathcal{P}_i$.

Finally, the procedure constructs G' in the following way. We set the alphabet of G' to be $\Sigma^{l \cdot d}$, viewed as $(\Sigma^l)^d$. If $\sigma \in (\Sigma^l)^d$, and we denote $\sigma = (\sigma_1, \ldots, \sigma_d)$, then we denote by $\sigma_{i,j}$ the element $(\sigma_i)_j \in \Sigma$. To define the constraints of G', let us consider their action on an assignment π' of G'. An edge (u, v) of $\mathcal{DB}'$ is associated with the constraint that accepts if and only if all the following conditions hold:

1. For every $i \in [d]$, the values $\left(\pi'(u)_{i,l}, \pi'(u)_{i,1}\right)$ satisfy the edge $\left(\mu_i^{-1}(u), u\right)$ of G.

2. It holds that $\pi'(u)_{1,1} = \cdots = \pi'(u)_{d,1}$ and that $\pi'(v)_{1,1} = \cdots = \pi'(v)_{d,1}$.

3. For every $i \in [d]$ and $j \in [l-1]$ such that u and v are the j-th and $(j+1)$-th vertices of a path in $p \in \mathcal{P}_i$ respectively, it holds that $\pi'(u)_{i,j} \neq \pi'(v)_{i,j+1}$.

4. Same as Condition 3, but when v is the j-th vertex of p and u is its $(j+1)$-th vertex.

The size of G' is indeed $|\Lambda|^{m+1}$, since the graph is $|\Lambda|$-regular and contains $|\Lambda|^m$ vertices. Furthermore, if G is satisfiable, then so is G': The satisfiability of G implies the satisfiability of G_1, so there exists a satisfying assignment π_1 for G_1. We construct a satisfying assignment π' from π_1 by assigning each vertex v of G' a value $\pi'(v)$, such that for each $i \in [d]$, if v is the j-th vertex of a path $p \in \mathcal{P}_i$ that connects the vertices u and $\mu_i(u)$, then we set $\pi'(v)_{i,j} = \pi_1(u)$. Note that this is well defined, since every vertex is the j-th vertex of exactly one path in $\mathcal{P}_i$.

It remains to analyze the soundness of G'. Suppose that $\mathrm{UNSAT}(G) \geq \rho$. Then, by Lemma 4.8, it holds that $\mathrm{UNSAT}(G_1) \geq \rho/c$. Let π' be an assignment to G' that minimizes the fraction of violated edges of G'. Without loss of generality, we may assume that for every vertex v of the $\mathcal{DB}$, it holds that $\pi'(v)_{1,1} = \cdots = \pi'(v)_{d,1}$: If there is a vertex v that does not match this condition, all of the edges attached to v are violated, and therefore, we can modify the $\pi'(v)$ to match this condition without increasing the fraction of violated edges of π'. Define an assignment π_1 to G_1 by setting $\pi_1(v) = \pi'(v)_{1,1}$ (when v is viewed as a vertex of $\mathcal{DB}$).

Since $\mathrm{UNSAT}(G_1) \geq \rho/c$, it holds that π_1 violates at least ρ/c fraction of the edges of G_1 or, in other words, π_1 violates at least $\rho \cdot 2 \cdot n \cdot d/c$ edges of G_1. Thus, there must exist a permutation μ_i such that π_1 violates at least $\rho \cdot 2 \cdot n/c$ edges of G_1 of the form $(u, \mu_i(u))$. Fix such an edge $(u, \mu_i(u))$ and consider the corresponding path $p \in \mathcal{P}_i$. Observe that π' must violate at least

one of the edges of p: To see it, note that if π' would satisfy all the edges on p, then it would imply that $\pi'\left(\mu_i(u)\right)_{i,l} = \pi_1(u)$ and that $\pi'\left(\mu_i(u)\right)_{i,1} = \pi_1(\mu_i(u))$, but the last two values violate the edge $(u, \mu_i(u))$ of G_1, and therefore, π' must violate the last edge of p—contradiction. It follows that for each of the $\rho \cdot 2 \cdot n/c$ edges of the matching μ_i that are violated by π_1, it holds that π' violates at least one edge of their corresponding path. By averaging, there must exist $j \in [l]$ such that for at least $\rho \cdot 2 \cdot n/c \cdot l$ edges of the matching μ_i, it holds that π' violates the j-th edge of their corresponding path.

Now, by the definition of the paths in $\mathcal{P}_i$, no edge of G' can be the j-th edge of two distinct paths in $\mathcal{P}_i$, and therefore, it follows that there at least $\rho \cdot 2 \cdot n/c \cdot l$ edges of G' are violated by π'. Finally, there are $|\Lambda|^{m+1}$ edges in G', and this implies that π' violates a fraction of the edges of G' that is at least

$$\frac{\rho \cdot 2 \cdot n/c \cdot l}{|\Lambda|^{m+1}} = \Omega\left(\frac{n}{|\Lambda|^{m+1} \cdot l} \cdot \rho\right),$$

as required. $\square$

4.4. Routing on de Bruijn Graphs.   In this section, we prove the routing property of de Bruijn graph given in Fact 4.5. Recall the following.

DEFINITION (4.1, restated).   *Let Λ be a finite alphabet and let $m \in \mathbb{N}$. The de Bruijn graph $\mathcal{DB}_{\Lambda,m}$ is the directed graph whose vertices set is Λ^m such that each vertex $(\alpha_1, \ldots, \alpha_t) \in \Lambda^m$ has outgoing edges to all the vertices of the form $(\alpha_2, \ldots, \alpha_t, \beta)$ for $\beta \in \Lambda$.*

FACT (4.5, restated).   *Let $\mathcal{DB}_{\Lambda,m}$ be a de-Bruijn graph. Then, given a permutation μ on the vertices of $\mathcal{DB}_{\Lambda,m}$ one can find a set of undirected paths of length $l = 2m$ which connect each vertex v to $\mu(v)$ and which have the following property: For every $j \in [l]$, each vertex v is the j-th vertex of exactly one path. Furthermore, finding the paths can be done in time that is polynomial in the size of $\mathcal{DB}_{\Lambda,m}$.*

We actually prove the following slightly stronger result, which says that if the permutation μ acts only on the i last coordinates of its input, then the routing can be done in only $2i$ steps.

CLAIM 4.9. *Let $\mathcal{DB}_{\Lambda,m}$ be a de-Bruijn graph and let $i \in [m]$. Then, given a permutation μ on Λ^i, one can find a set of undirected paths of length $2 \cdot i$ that connect each vertex $(\alpha_1, \ldots, \alpha_m)$ of $\mathcal{DB}_{\Lambda,m}$ to the vertex*

$$(\alpha_1, \ldots, \alpha_{m-i}, \mu(\alpha_{m-i+1}, \ldots, \alpha_m)),$$

and that have the following two property: For every $j \in [l]$, each vertex v is the j-th vertex of exactly one path. Furthermore, finding the paths can be done in time that is polynomial in the size of $\mathcal{DB}_{\Lambda,m}$.

The proof works by induction on i. For $i = 0$, the claim is obvious. Assume that the claim holds for some $0 \leq i < m$. We prove that the claim holds for $i + 1$. Let $\mathcal{DB} = \mathcal{DB}_{\Lambda,m}$, and let μ be a permutation on Λ^{i+1}. For convenience, let us define the action of μ on each $(\alpha_1, \ldots, \alpha_m) \in \mathbb{F}^m$ as $\mu(\alpha_1, \ldots, \alpha_m) = (\alpha_1, \ldots, \alpha_{m-i-1}, \mu(\alpha_{m-i}, \ldots, \alpha_m))$.

Let G be the directed graph whose vertices are the set Λ^m and whose edges are all the pairs of the form $(v, \mu(v))$. Let G' be the graph that is obtained from G by contracting each $|\Lambda|$ vertices of G that agree on their last coordinate to one vertex. Clearly, every vertex in G' has in-degree and out-degree exactly $|\Lambda|$, and each edge of G' corresponds to an edge of G. Furthermore, observe that the vertices of G' can be identified with the vertices of Λ^{m-1}.

The $|\Lambda|$-regularity of G implies that the edges of G' can be partitioned to $|\Lambda|$ perfect matchings $\{G'_\sigma\}_{\sigma \in \Lambda}$ in polynomial time (see, e.g., Cameron 1998, Proposition 18.1.2). Fix a matching G'_σ, and consider an edge e' in G'_σ. Observe that if e is coming out of a vertex $(\alpha_1, \ldots, \alpha_{m-1})$ of G', then it must enter a vertex of the form $(\alpha_1, \ldots, \alpha_{m-i}, \alpha'_{m-i+1}, \ldots, \alpha'_{m-1})$. Thus, we can define a permutation ν_σ on Λ^i that maps $(\alpha_{m-i}, \ldots, \alpha_{m-1})$ to $(\alpha'_{m-i}, \ldots, \alpha'_{m-1})$ for each such edge e'(since G'_σ is a perfect matching, this is well defined). We now invoke the induction hypothesis on the graph

$\mathcal{DB} = \mathcal{DB}_{\Lambda,m}$ to find a set of paths $\mathcal{P}_\sigma$ of length $2i$ for each permutation ν_σ.

We construct the required paths for μ as follows. Let $v = (\alpha_1, \ldots, \alpha_m) \in \Lambda^m$, and suppose that $\mu(\alpha_{m-i}, \ldots, \alpha_m) = (\alpha'_{m-i}, \ldots, \alpha'_m)$. We wish to construct a path p in $\mathcal{DB}$ that connects v to $\mu(v)$. The edge $(v, \mu(v))$ corresponds to some edge e' in G', so let G'_β be the matching to which e' belongs. We turn to construct the path p: The first edge in the path p connects $v = (\alpha_1, \ldots, \alpha_m)$ to the vertex $(\beta, \alpha_1, \ldots, \alpha_{m-1})$. The next $2i$ edges of p will be the edges of the path in $\mathcal{P}_\beta$ that connects $(\beta, \alpha_1, \ldots, \alpha_{m-1})$ to $(\beta, \alpha_1, \ldots, \alpha_{m-i-1}, \alpha'_{m-i}, \ldots, \alpha'_{m-1})$. Finally, the last edge of p will go from the vertex $(\beta, \alpha_1, \ldots, \alpha_{m-i-1}, \alpha'_{m-i}, \ldots, \alpha'_{m-1})$ to the vertex

$$(\alpha_1, \ldots, \alpha_{m-i-1}, \alpha'_{m-i}, \ldots, \alpha'_m) = \mu(v).$$

Observe that p indeed connects v to $\mu(v)$ and is of length $2 \cdot (i+1)$.

It remains to show that for each $j \in [2i + 2]$, it holds that every vertex v is the j-th vertex of exactly one path. The cases of $j = 1$ and $j = 2 \cdot i + 2$ are trivial. We analyze the case of $j = 2$, and the rest of cases will follow from the induction hypothesis. Let $u = (\beta, \alpha_1, \ldots, \alpha_{m-1}) \in \Lambda^m$. We show that u is the second vertex of a unique path p by constructing p. Let e' be the unique edge of G' that comes out of the vertex $(\alpha_1, \ldots, \alpha_{m-1})$ and that belongs to the matching G'_β. The edge e' of G' corresponds to some unique edge $(v, \mu(v))$ of G. Now, by construction, the only path p such that u is the second vertex of p is the path that connects v to $\mu(v)$. The required result follows. $\qquad\square$

5. Derandomized Parallel Repetition of Constraint Graphs with Linear Structure

In this section, we prove Lemma 3.4, restated below, by implementing a form of derandomized parallel repetition on graphs that have linear structure.

LEMMA 5.1 (3.4, restated). *There exist a universal constant h and a polynomial time procedure that satisfy the following requirements:*

○ **Input:**

– A finite field $\mathbb{F}$ of size q

– A constraint graph $G = (\mathbb{F}^m, E)$ over alphabet Σ that has a linear structure.

– A parameter $d_0 \in \mathbb{N}$ such that $d_0 < m/h^2$. This parameter will determine the dimension of linear subspaces used in the derandomized parallel repetition and thus together with q will determine the number of repetitions used in the derandomized parallel repetition.

– A parameter $\rho \in (0,1)$ such that $\rho \geq h \cdot d_0 \cdot q^{-d_0/h}$. Intuitively, the parameter ρ should be chosen such that $1 - \rho$ is an upper bound on the soundness error of G.

○ **Output:** A constraint graph G' such that the following holds:

– G' has size $n^{O(d_0)}$.

– G' has alphabet $\Sigma^{q^{O(d_0)}}$.

– If G is satisfiable then G' is satisfiable.

– If $\mathrm{SAT}\,(G) < 1 - \rho$ then $\mathrm{SAT}\,(G') < h \cdot d_0 \cdot q^{-d_0/h}$.

– G' has the projection property.

The basic idea of the proof is as follows. G' contains two kinds of vertices: the first kind corresponds to small subspaces of the vertices space $\mathbb{F}^m$ and of the other kind corresponds to small subspaces of the edges space E, where in both cases "small subspaces" mean $O(d_0)$-dimensional subspaces. A satisfying assignment Π to G' is expected to be constructed in the following way: Take a satisfying assignment π to G. For each vertex of G' which is a subspace A of vertices, the assignment Π should assign A to $\pi_{|A}$. For each vertex of G' which is a subspace F of edges, the assignment Π should assign F to $\pi_{|\mathrm{left}(F) \cup \mathrm{right}(F)}$.

The edges of G' are constructed so as to simulate a test on Π to which we refer as the "E-test" and act roughly as follows (see Figure 5.1 for the actual test): Choose a random subspace F of edges and a random subspace A of endpoints of F, and accept if

and only if the labeling of the endpoints of the edges in F by $\Pi\,(F)$ satisfies the edges and is consistent with the labeling of the vertices of A by $\Pi\,(A)$.

The intuition that underlies the soundness analysis of G' is the following: The E-test performs some form of a "derandomized direct product test" on Π—if we compare it to the P-test (Figure 2.1), then the pair (A, F) here is analogous to the pair (A, B) there. Therefore, if $\Pi\,(F)$ is consistent with $\Pi\,(A)$, the labeling $\Pi\,(F)$ should be roughly consistent with some assignment π to G. Therefore, by checking that the labeling $\Pi\,(F)$ satisfies the edges in F, the E-test checks that π satisfies many edges of π in parallel. In this sense, the E-test can be thought as a form of "derandomized parallel repetition".

The rest of this section is organized as follows. In Section 5.1, we provide a formal description of the construction of G' and analyze all its parameters except for the soundness. In order to analyze the soundness of G', we introduce in Section 5.2 a specialized direct product test. Finally, in Section 5.3, we analyze the soundness of G' by reducing it to the analysis of the specialized direct product test.

NOTATION 5.2. *Given a function* $f : U \rightarrow \Sigma$ *and two subsets* $S, T \subseteq U$, *we denote by* $f_{|(S,T)}$ *the pair of functions* $\left(f_{|S}, f_{|T}\right)$.

NOTATION 5.3. *Recall that in* Notation 2.1 *we denoted the notation* $f \overset{\alpha}{\approx} g$ *(*$f \overset{\alpha}{\not\approx} g$*) to mean that* f *and* g *differ on at most (more than)* α *fraction of the elements of* U. *We now extend this notation to pairs of functions. Given two pairs of functions* $f_1, f_2 : U \rightarrow \Sigma$ *and* $g_1, g_2 : V \rightarrow \Sigma$, *we denote by* $(f_1, g_1) \overset{\alpha}{\approx} (f_2, g_2)$ *the fact that both* $f_1 \overset{\alpha}{\approx} f_2$ *and* $g_1 \overset{\alpha}{\approx} g_2$, *and otherwise we denote* $(f_1, g_1) \overset{\alpha}{\not\approx} (f_2, g_2)$.

5.1. The Construction of G'.   We begin by describing the construction of G'. Let $G = (\mathbb{F}^m, E)$ be the given constraint graph, let d_0 be the parameter from Lemma 3.4, and let $d_1 = h \cdot d_0$ where h is the universal constant from Lemma 3.4 to be chosen later. The graph G' is bipartite. The right vertices of G' are identified with all the $2d_0$-subspaces of $\mathbb{F}^m$ (the vertex space of G). The left

1. Let F_L and F_R be random d_1-subspaces of E, and let

$$B_L \overset{\text{def}}{=} \text{left}\,(F_L)\,, \quad B_R \overset{\text{def}}{=} \text{right}\,(F_R)\,, \quad F \overset{\text{def}}{=} F_L + F_R.$$

   F_L and F_R are chosen to be uniformly and independently distributed d_1-subspaces of E conditioned on $\dim(F) = 2d_1$, $\dim\,(B_L) = d_1$, $\dim\,(B_R) = d_1$, and $B_L \cap B_R = \{0\}$.

2. Let A_L and A_R be uniformly distributed d_0-subspaces of B_L and B_R respectively, and let
$$A \overset{\text{def}}{=} A_L + A_R.$$

3. Accept if and only if $\Pi\,(F)_{|(A_L,A_R)} = \Pi\,(A)_{|(A_L,A_R)}$ and the assignment $\Pi\,(F)$ satisfies the edges in F.
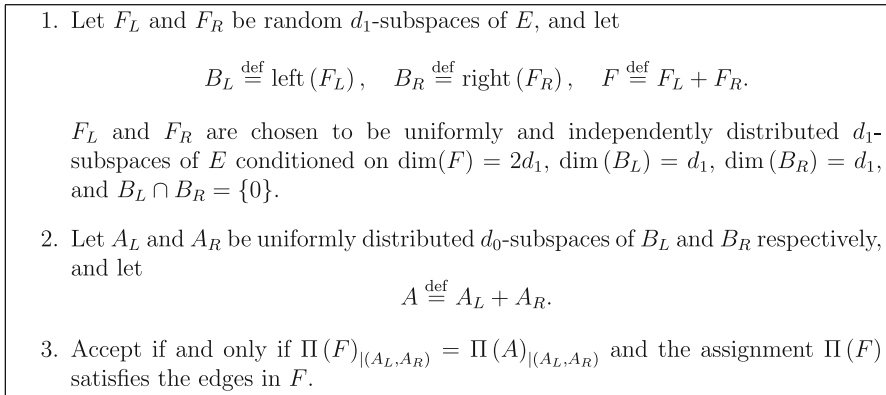
Figure 5.1: The E-test.

vertices of G' are identified with all the $2d_1$-subspaces of the edge space E of G. An assignment Π to G' should label each $2d_0$-subspace A of $\mathbb{F}^m$ with a function from A to Σ, and each $2d_1$-subspace F of E with a function that maps the endpoints of the edges in F to Σ. The edges of G' are constructed such that they simulate the action of the "E-test" described in Figure 5.1.

The completeness of G' is clear. It is also clear that G' has projection constraints. Let us verify the size and alphabet size of G'. The size of G' is at most the number of $2d_1$-subspaces of E multiplied by the number of $2d_0$-subspaces of $\mathbb{F}^m$, which is $|E|^{2d_1} \cdot |\mathbb{F}^m|^{2d_0}$. It holds that $d_0 < d_1$, and furthermore, the linear structure of G' implies that $\dim E \geq m$ (by Item iii of Definition 3.1), so it follows that $|\mathbb{F}^m|^{2d_0} \leq |E|^{2d_1}$ and thus $|E|^{2d_1} \cdot |\mathbb{F}^m|^{2d_0} \leq |E|^{4d_1}$. Finally, observe that the size of G is $n = |E|$, so it follows that the size of G' is at most $n^{4d_1} = n^{O(d_0)}$, as required.

For the alphabet size, recall that an edges subspace F is labeled by a function that maps the endpoints of the edges to Σ. Such a function can be represented by a string in $\Sigma^{2 \cdot q^{2 \cdot d_1}}$, since each $2d_1$-subspace F contains q^{2d_1} edges and each has two endpoints. It can be observed similarly that the labels assigned by Π to $2d_0$-subspaces A of $\mathbb{F}^m$ can be represented by strings in $\Sigma^{2 \cdot q^{2 \cdot d_1}}$. The alphabet of G' is therefore $\Sigma^{2 \cdot q^{2 \cdot d_1}} = \Sigma^{q^{O(d_0)}}$, as required.

5.2. The Specialized Direct Product Test. In order to analyze the soundness of the E-test, we introduce a variant of the direct

---

1. Choose uniformly distributed pair of independent d_1-subspaces B_1, B_2 of $\mathbb{F}^m$.

2. Choose uniformly distributed pair of d_0-subspaces $A_1 \subseteq B_1$, $A_2 \subseteq B_2$.

3. Accept if and only if $\Pi\left(B_1, B_2\right)_{|(A_1, A_2)} = \Pi\left(A_1 + A_2\right)_{|(A_1, A_2)}$.

Figure 5.2: The S-test.

product test of Impagliazzo *et al.* (2009) that is specialized to our needs. We refer to this variant as the *specialized direct product test*, abbreviated the "S-test".

Given an string $\pi : \mathbb{F}^m \to \Sigma$, we define its *S-direct product* Π (with respect to $d_0, d_1 \in \mathbb{N}$) as follows: Π assigns each $2d_0$-subspace $A \subseteq \mathbb{F}^m$ the function $\pi_{|A}$, and assigns each pair of independent d_1-subspaces (B_1, B_2) the pair of functions $\pi_{|(B_1, B_2)}$.

We turn to consider the task of testing whether a given assignment Π is the S-direct product of some string $\pi : \mathbb{F}^m \to \Sigma$. In our settings, we are given an assignment Π that assigns each $2d_0$-subspace A to a function $a : A \to \Sigma$ and each pair of independent d_1-subspaces (B_1, B_2) to a pair of functions $b_1 : B_1 \to \Sigma, b_2 : B_2 \to \Sigma$. We wish to check whether Π is a S-direct product of some $\pi : \mathbb{F}^m \to \Sigma$. To this end, we invoke the S-test, described in Figure 5.2.

It is easy to see that if Π is a S-direct product, then the S-test always accepts. Furthermore, it can be shown that if Π is "far" from being a S-direct product, then the S-test rejects with high probability. As in the P-test, this holds even if Π is a randomized assignment. Formally, we have the following result.

THEOREM 5.4 (the soundness of the S-test). *There exist universal constants $h', c \in \mathbb{N}$ such that the following holds: Let $d_0 \in \mathbb{N}, d_1 \geq h' \cdot d_0$, and $m \geq h' \cdot d_1$, and let $\varepsilon \geq h' \cdot d_0 \cdot q^{-d_0/h'}, \alpha \stackrel{\text{def}}{=} h' \cdot d_0 \cdot q^{-d_0/h'}$. Suppose that a (possibly randomized) assignment Π passes the S-test with probability at least ε. Then there exists an assignment $\pi : \mathbb{F}^m \to \Sigma$ for which the following holds. Let B_1, B_2 be uniformly distributed and independent d_1-subspaces of $\mathbb{F}^m$, let A_1 and A_2 be uniformly distributed d_0-subspaces of B_1 and B_2, respectively, and*

denote $A = A_1 + A_2$. Then:

$$\Pr\left[\Pi\left(B_1, B_2\right)_{|(A_1, A_2)} = \Pi\left(A\right)_{|(A_1, A_2)} \text{ and } \Pi\left(B_1, B_2\right) \overset{\alpha}{\approx} \pi_{|(B_1, B_2)}\right]$$
$$= \Omega\left(\varepsilon^c\right).$$
(5.5)

We defer the proof of Theorem 5.4 to Section 9.

REMARK 5.6. *Note that Equation (5.5) only says that Π is close to the S-direct product of π on pairs (B_1, B_2), and not necessarily on $2d_0$-subspaces A. In fact, it could be also proved that Π is close to the S-direct product of π on the $2d_0$-subspaces, but this is unnecessary for our purposes.*

5.3. The Soundness of the Derandomized Parallel Repetition.   In this section, we prove the soundness of G': namely, that if $\text{SAT}(G) < 1 - \rho$, then

$$\text{SAT}(G') \leq \varepsilon \overset{\text{def}}{=} h \cdot d_0 \cdot q^{-d_0/h},$$

where h is the universal constant from Lemma 3.4 (derandomized parallel repetition). We will choose h to be sufficiently large such that the various inequalities in the following proof will hold. To this end, we note that throughout all the following proof, increasing the choice of h does not break any of our assumptions on h, so we can always choose a larger h to satisfy the required inequalities.

Let h' and c be the universal constants whose existence is guaranteed by Theorem 5.4 (the soundness of the S-test), and let α denote the corresponding value from Theorem 5.4. We will choose the constant h to be at least h'.

Let Π be an assignment to G'. Let us denote by $\mathcal{T}$ the event in which the E-test accepts Π. With a slight abuse of notation, for a subspace $F \subseteq E$ and an assignment $\pi : \mathbb{F}^m \to \Sigma$, we denote by $\Pi(F) \overset{\alpha}{\approx} \pi$ the claim that for at least $1 - \alpha$ fraction of the edges e of F, it holds that $\Pi(F)$ is consistent with π on both the endpoints of e, and otherwise, we denote $\Pi(F) \overset{\alpha}{\not\approx} \pi$. Our proof is based on two steps:

○ We will show (in Proposition 5.7 below) that if the test accepts with probability ε, then it is "because" Π is consistent with some underlying assignment $\pi : \mathbb{F}^m \to \Sigma$. This is done essentially by observing that the E-test "contains" an S-test and reducing to the analysis of the S-test.

○ On the other hand, we will show (in Proposition 5.8 below) that for every assignment $\pi : \mathbb{F}^m \to \Sigma$ the probability that the test accepts while being consistent with π is negligible. This is done roughly as follows: Any fixed assignment π is rejected by at least ρ fraction of G's edges. Furthermore, the subspace F queried by the test is approximately a uniformly distributed subspace of E and hence a good sampler of E. It follows F must contain $\approx \rho$ fraction of edges of G that reject π, and therefore, $\Pi(F)$ must be inconsistent with π.

The conclusions of each of the foregoing two steps clearly contradict each other; we therefore conclude that the E-test accepts with probability less than ε. We now state the two said propositions, which formalize the foregoing two steps, and which are proved in Section 5.3.1 and Section 5.3.2, respectively.

PROPOSITION 5.7. *There exists* $\varepsilon_0 = \Omega(\varepsilon^c)$ *such that the following holds: If* $\Pr[\mathcal{T}] \geq \varepsilon$, *then there exists an assignment* $\pi : \mathbb{F}^m \to \Sigma$ *such that*

$$\Pr\left[\mathcal{T} \text{ and } \Pi(F) \stackrel{4\cdot\alpha}{\approx} \pi\right] \geq \varepsilon_0.$$

PROPOSITION 5.8. *Let* ε *be as in Proposition 5.7. Then, for every assignment* $\pi : \mathbb{F}^m \to \Sigma$ *it holds that* $\Pr\left[\mathcal{T} \text{ and } \Pi(F) \stackrel{4\cdot\alpha}{\approx} \pi\right] < \varepsilon_0$.

Clearly, the two propositions together imply that $\Pr[\mathcal{T}] \leq \varepsilon$, as required.

Before turning to the proofs of Proposition 5.7 and Proposition 5.8, let us state a useful claim that says that if we take a random d-subspace of edges and project it to its left endpoints (respectively, right endpoints), we get a random d-subspace of vertices with high probability.

CLAIM 5.9. *Let $d \in \mathbb{N}$ and let E_a be a uniformly distributed d-subspace of E. Then, $\Pr\left[\dim\left(\mathrm{left}\left(E_a\right)\right) = d\right] \geq 1 - d/q^{m-d}$, and conditioned on $\dim\left(\mathrm{left}\left(E_a\right)\right) = d$, it holds that $\mathrm{left}\left(E_a\right)$ is a uniformly distributed d-subspace of $\mathbb{F}^m$. The same holds for $\mathrm{right}\left(E_a\right)$.*

More generally, let E_b be a fixed subspace of E such that $\dim\left(E_b\right) > d$ and $\dim\left(\mathrm{left}\left(E_b\right)\right) = D > d$. Let E_a be a uniformly distributed d-subspace of E_b. Then, $\Pr\left[\dim\left(\mathrm{left}\left(E_a\right)\right) = d\right] \geq 1 - d/q^{D-d}$, and conditioned on $\dim\left(\mathrm{left}\left(E_a\right)\right) = d$, it holds that $\mathrm{left}\left(E_a\right)$ is a uniformly distributed d-subspace of $\mathrm{left}\left(E_b\right)$. Again, the same holds for $\mathrm{right}\left(E_a\right)$.

We defer the proof of to Section 5.4.

5.3.1. Proof of Proposition 5.7. Suppose that $\Pr\left[\mathcal{T}\right] \geq \varepsilon$. We prove Proposition 5.7 by arguing that the E-test contains an "implicit S-test" and applying Theorem 5.4 (the soundness of the S-test).

Observe that, without loss of generality, we may assume that for every edge-subspace F such that $\Pi\left(F\right)$ violates one of the edges in F, it holds that $\Pi\left(F\right)_{(A_L, A_R)} \neq \Pi\left(A\right)_{(A_L, A_R)}$ for any choice of A_L and A_R. The reason is that for every such F, we can modify $\Pi\left(F\right)$ such that it assigns symbols outside of the alphabet Σ of G, so $\Pi\left(F\right)$ will always disagree with $\Pi\left(A\right)$. Note that this modification indeed does not change the acceptance probability of Π. This assumption that we make on Π implies, in particular, that the event $\mathcal{T}$ is equivalent to the event $\Pi\left(F\right)_{(A_L, A_R)} \neq \Pi\left(A\right)_{(A_L, A_R)}$, and this equivalence is used in the following analysis.

We turn back to the proof of Proposition 5.7. We begin the proof by extending Π to pairs of independent d_1-subspaces of $\mathbb{F}^m$ in a randomized manner as follows: Given a pair of independent d_1-subspaces B_1 and B_2, we choose F_1 and F_2 to be uniformly distributed and independent d_1-subspaces of E such that $\mathrm{left}\left(F_1\right) = B_1$ and $\mathrm{right}\left(F_2\right) = B_2$, and set $\Pi\left(B_1, B_2\right) = \Pi\left(F_1 + F_2\right)_{|(B_1, B_2)}$.

Now, observe that the probability that the E-test accepts equals to the probability that the S-test accepts the extended Π. The reason is that the subspaces B_L, B_R, A_L, A_R of the E-test are distributed like the subspaces B_1, B_2, A_1, A_2 of the S-test. It thus

follows the E-test performs in a way an S-test on the extended assignment Π.

Next, we note that by choosing h to be sufficiently large, the foregoing "implicit S-test" matches the requirements of Theorem 5.4 (the soundness of the S-test), and we can thus apply this theorem. It follows that there exists an assignment $\pi : \mathbb{F}^m \to \Sigma$ such that

$$\Pr\left[\Pi\left(B_L, B_R\right)_{(A_L,A_R)} = \Pi\left(A\right)_{|(A_L,A_R)} \text{ and } \Pi\left(B_L, B_R\right) \overset{\alpha}{\approx} \pi_{(B_L,B_R)}\right]$$
$$\geq \Omega\left(\varepsilon^c\right).$$
(5.10)

By using the equivalence between the event $\mathcal{T}$ and the event $\Pi\left(F\right)_{(A_L,A_R)} \neq \Pi\left(A\right)_{(A_L,A_R)}$, it follows that Inequality (5.10) is equivalent to the inequality

$$(5.11) \quad \Pr\left[\mathcal{T} \text{ and } \Pi\left(F\right)_{|(B_L,B_R)} \overset{\alpha}{\approx} \pi_{|(B_L,B_R)}\right] \geq \Omega\left(\varepsilon^c\right).$$

We turn to show that

$$\Pr\left[\mathcal{T} \text{ and } \Pi\left(F\right) \overset{4\alpha}{\approx} \pi\right] \geq \Omega\left(\varepsilon^c\right).$$

We will prove that if F is such that $\Pi\left(F\right) \overset{4\alpha}{\not\approx} \pi$, then for a random choice of B_L, B_R conditioned on F, it is highly unlikely that Inequality (5.11) still holds. Formally, we will prove the following.

CLAIM 5.12. *For every fixed $2d_0$-subspace F_0 of E such that $\Pi\left(F_0\right) \overset{4\alpha}{\not\approx} \pi$, it holds that*

$$\Pr\left[\Pi\left(F\right)_{|(B_L,B_R)} \overset{\alpha}{\approx} \pi_{|(B_L,B_R)} \middle| F = F_0\right] \leq 1/\left(q^{d_1-2} \cdot \alpha^2\right).$$

We defer the proof of Claim 5.12 to the end of this section. Claim 5.12 immediately implies the following.

COROLLARY 5.13. *It holds that*

$$\Pr\left[\Pi\left(F\right)_{(B_L,B_R)} \overset{\alpha}{\approx} \pi_{|(B_L,B_R)} \middle| \Pi\left(F\right) \overset{4\alpha}{\not\approx} \pi\right] \leq 1/\left(q^{d_1-2} \cdot (\alpha/2)^2\right).$$

By combining Corollary 5.13 with Inequality (5.11) and by choosing h to be sufficiently large, it follows that

$$\Pr\left[\mathcal{T} \text{ and } \Pi\left(F\right)_{|(B_L,B_R)} \overset{\alpha}{\approx} \pi_{|(B_L,B_R)} \text{ and } \Pi\left(F\right) \overset{4\alpha}{\approx} \pi\right] \geq \Omega\left(\varepsilon^c\right).$$

This implies that

$$\Pr\left[\mathcal{T} \text{ and } \Pi\left(F\right) \overset{4\alpha}{\approx} \pi\right] \geq \Omega\left(\varepsilon^c\right).$$

Setting ε_0 to be the latter lower bound finishes the proof. $\square$

PROOF (Proof of Claim 5.12.).   Observe that the assumption $\Pi\left(F_0\right) \overset{4\alpha}{\not\approx} \pi$ implies that one of the following holds

$$\Pi\left(F_0\right)_{|\text{left}(F_0)} \overset{2\alpha}{\not\approx} \pi_{|\text{left}(F_0)},$$

$$\Pi\left(F_0\right)_{|\text{right}(F_0)} \overset{2\alpha}{\not\approx} \pi_{|\text{right}(F_0)}.$$

Without loss of generality, assume that the first holds. Now, when conditioning on $F = F_0$, it holds that F_L is a uniformly distributed d_1-subspace of F_0 satisfying $\dim\left(\text{left}\left(F_L\right)\right) = d_1$. By Claim 5.9 (with $E_b = F_0$ and $E_a = F_L$), under the conditioning on $\dim\left(\text{left}\left(F_L\right)\right) = d_1$, it holds that $B_L \overset{\text{def}}{=} \text{left}\left(F_L\right)$ is a uniformly distributed d_1-subspace of $\text{left}\left(F_0\right)$. Therefore, by Lemma 2.5 (subspace-point sampler), the event $\Pi\left(F\right)_{|B_L} \overset{\alpha}{\not\approx} \pi_{|B_L}$ occurs with probability at least

$$1 - 1/\left(q^{d_1-2} \cdot \left(\alpha - q^{-d_1}\right)^2\right) \geq 1 - 1/\left(q^{d_1-2} \cdot (\alpha/2)^2\right),$$

as required. $\square$

5.3.2. Proof of Proposition 5.8. Fix an assignment $\pi : \mathbb{F}^m \to \Sigma$. By assumption, it holds that $\text{SAT}\left(G\right) < 1 - \rho$, and therefore, π must violate a set E^* of edges of G of density at least ρ. Below, we will show that at least $\rho/2$ fraction of the edges in F are in E^* with probability greater than $1 - \varepsilon_0$. Now, observe that $\Pi\left(F\right)$ cannot satisfy the edges of F and at the same time be consistent with π on the edges in E^*, and hence whenever the latter event occurs,

it either holds that the E-test fails or that $\Pi(F) \overset{\rho/2}{\not\approx} \pi$. However, for sufficiently large choice of h, it holds that $\rho/2 > 4 \cdot \alpha$, and therefore, the probability that the E-test passes and at the same time it holds that $\Pi(F) \overset{4 \cdot \alpha}{\approx} \pi$ is less than ε_0, as required.

It remains to show that

$$\Pr\left[\frac{|F \cap E^*|}{|F|} \geq \rho/2\right] > 1 - \varepsilon_0.$$

We prove the above inequality by showing that F is close to being a uniformly distributed $2d_1$-subspace of E, and then applying Lemma 2.5 (subspace-point sampler). To this end, let F'_L and F'_R be uniformly distributed d_1-subspaces of F, and let $F' = F'_L + F'_R$. Let us denote by $\mathcal{E}_1$ the event in which $\dim(F') = 2d_1$ and by $\mathcal{E}_2$ the event in which $\text{left}(F'_L)$ and $\text{right}(F'_R)$ are independent and are of dimension d_1. Observe that conditioned on $\mathcal{E}_1$ and $\mathcal{E}_2$, the subspace F' is distributed exactly like the subspace F. It therefore holds that

$$\begin{aligned}
\Pr\left[\frac{|F \cap E^*|}{|F|} \geq \rho/2\right] &= \Pr\left[\frac{|F' \cap E^*|}{|F'|} \geq \rho/2 \,\middle|\, \mathcal{E}_1 \text{ and } \mathcal{E}_2\right] \\
&\geq \Pr\left[\frac{|F' \cap E^*|}{|F'|} \geq \rho/2 \text{ and } \mathcal{E}_2 \,\middle|\, \mathcal{E}_1\right] \\
&\geq \Pr\left[\frac{|F' \cap E^*|}{|F'|} \geq \rho/2 \,\middle|\, \mathcal{E}_1\right] - \Pr\left[\neg\mathcal{E}_2 | \mathcal{E}_1\right] \\
&\geq \Pr\left[\frac{|F' \cap E^*|}{|F'|} \geq \rho/2 \,\middle|\, \mathcal{E}_1\right] - \frac{\Pr\left[\neg\mathcal{E}_2\right]}{\Pr\left[\mathcal{E}_1\right]}.
\end{aligned}$$

Now, observe that conditioned on $\mathcal{E}_1$, the subspace F' is a uniformly distributed $2d_1$-subspace of E. Thus, by Lemma 2.5 (subspace-point sampler), it holds that

$$\begin{aligned}
\Pr\left[\frac{|F' \cap E^*|}{|F'|} \geq \rho/2 \,\middle|\, \mathcal{E}_1\right] &\geq 1 - 1/q^{2d_1-2} \cdot \left(\rho/2 - q^{-2d_1}\right)^2 \\
&\geq 1 - 1/q^{2d_1-2} \cdot \left(\rho/3\right)^2.
\end{aligned}$$

Moreover, by Proposition 2.18, it holds that

$$\begin{aligned}
\Pr\left[\mathcal{E}_1\right] &\geq 1 - 2d_1/q^{\dim E - 2d_1} \\
&\geq 1 - 2d_1/q^{m - 2d_1} \\
&\geq \frac{1}{2}.
\end{aligned}$$

Finally, we upper bound $\Pr\left[\neg\mathcal{E}_2\right]$ by showing that $\Pr\left[\mathcal{E}_2\right] \geq 1 - 4d_1/q^{m - 2 \cdot d_1}$. By Claim 5.9 (with $E_b = E$ and $E_a = F'_L, F'_R$), it holds that $\dim\left(\text{left}\left(F'_L\right)\right) = \dim\left(\text{right}\left(F'_R\right)\right) = d_1$ with probability at least $1 - 2 \cdot d_1/q^{m - d_1}$. Furthermore, conditioned on the latter event, it holds that $\text{left}\left(F'_L\right)$ and $\text{right}\left(F'_R\right)$ are uniformly distributed d_1-subspaces of $\mathbb{F}^m$, and it is also easy to see that those subspaces are independent. By Proposition 2.18, this implies that conditioned on $\dim\left(\text{left}\left(F'_L\right)\right) = \dim\left(\text{right}\left(F'_R\right)\right) = d_1$ the subspaces $\text{left}\left(F'_L\right)$ and $\text{right}\left(F'_R\right)$ are independent with probability at least $1 - 2d_1/q^{m - 2 \cdot d_1}$, and hence $\Pr\left[\mathcal{E}_2\right] \geq 1 - 4d_1/q^{m - 2 \cdot d_1}$ as required.

We conclude that

$$\begin{aligned}
\Pr\left[\frac{|F \cap E^*|}{|F|} \geq \rho/2\right] &\geq \Pr\left[\frac{|F' \cap E^*|}{|F'|} \geq \rho/2 \,\middle|\, \mathcal{E}_1\right] - \frac{\Pr\left[\neg\mathcal{E}_2\right]}{\Pr\left[\mathcal{E}_1\right]} \\
&\geq 1 - 1/q^{2 \cdot d_1 - 2} \cdot (\rho/3)^2 - \frac{4 \cdot d_1/q^{m - 2 \cdot d_1}}{1/2} \\
&= 1 - 1/q^{2 \cdot d_1 - 2} \cdot (\rho/3)^2 - 8 \cdot d_1/q^{m - 2 \cdot d_1} \\
&> 1 - \varepsilon_0,
\end{aligned}$$

where the last inequality holds for sufficiently large choice of h. This concludes the proof. $\qquad\square$

5.4. Proof of Claim 5.9. In this section, we prove Claim 5.9, restated below. Recall that $G = \left(\mathbb{F}^m, E\right)$ is a graph with linear structure and, in particular, E is a linear subspace of edges.

CLAIM (5.9, restated). *Let $d \in \mathbb{N}$ and let E_a be a uniformly distributed d-subspace of E. Then, $\Pr\left[\dim\left(\text{left}\left(E_a\right)\right) = d\right] \geq 1 - d/q^{m-d}$, and conditioned on $\dim\left(\text{left}\left(E_a\right)\right) = d$, it holds that $\text{left}\left(E_a\right)$ is a uniformly distributed d-subspace of $\mathbb{F}^m$. The same holds for $\text{right}\left(E_a\right)$.*

More generally, let E_b be a fixed subspace of E such that $\dim(E_b) > d$ and $\dim(\text{left}(E_b)) > d$. Let E_a be a uniformly distributed d-subspace of E_b. Then, $\Pr[\dim(\text{left}(E_a)) = d] \geq 1 - d/q^{\dim(\text{left}(E_b)) - d}$, and conditioned on $\dim(\text{left}(E_a)) = d$, it holds that $\text{left}(E_a)$ is a uniformly distributed d-subspace of $\text{left}(E_b)$. Again, the same holds for $\text{right}(E_a)$.

PROOF.   We prove the proposition only for special case in which $E_b = E$ and only for $\text{left}(E_a)$. The proof of the general case and of the case of $\text{right}(E_a)$ is analogous. Let $e_1, \ldots, e_d$ be independent and uniformly distributed vectors of E, and let $E'_a = \text{span}\{e_1, \ldots, e_d\}$. We prove Proposition 5.9 by showing that E_a is distributed similarly to E'_a and analyzing the distribution of E'_a.

Observe that by Proposition 2.20, it holds that conditioned on $\dim(E'_a) = d$, the subspace E'_a is a uniformly distributed d-subspace of E. It therefore holds that

$$
\begin{aligned}
\Pr[\dim(\text{left}(E_a)) = d] &= \Pr[\dim(\text{left}(E'_a)) = d \mid \dim(E'_a) = d] \\
&\geq \Pr[\dim(\text{left}(E'_a)) = d \text{ and } \dim(E'_a) = d] \\
&= \Pr[\dim(\text{left}(E'_a)) = d],
\end{aligned}
$$

where the last equality holds since clearly $\dim(\text{left}(E'_a)) = d$ implies $\dim(E'_a) = d$. Now, since $\text{left}(\cdot)$ is a linear function, it holds that $\text{left}(e_1), \ldots \text{left}(e_d)$ are independent and uniformly distributed vectors of $\text{left}(E) = \mathbb{F}^m$, and therefore by Proposition 2.20, it holds that $\Pr[\dim(\text{left}(E'_a)) = d] \geq 1 - d/q^{m-d}$. It thus follows that $\Pr[\dim(\text{left}(E_a)) = d] \geq 1 - d/q^{m-d}$, as required.

It remains to show that conditioned on $\Pr[\dim(\text{left}(E_a)) = d]$, it holds that $\text{left}(E_a)$ is a uniformly distributed d-subspace of $\mathbb{F}^m$. To see it, observe that for every fixed d-subspace D of $\mathbb{F}^m$, it holds that

$$
\begin{aligned}
&\Pr[\text{left}(E_a) = D \mid \dim(\text{left}(E_a)) = d] \\
&= \Pr[\text{left}(E'_a) = D \mid \dim(E'_a) = d \text{ and } \dim(\text{left}(E'_a)) = d] \\
&= \Pr[\text{left}(E'_a) = D \mid \dim(\text{left}(E'_a)) = d],
\end{aligned}
$$

where the first equality again holds since conditioned on $\dim(E'_a) = d$ it holds that E'_a is a uniformly distributed d-subspace, and

the second equality again holds since $\dim\left(\text{left}\left(E'_a\right)\right) = d$ implies $\dim\left(E'_a\right) = d$. Now, it holds that $\text{left}\left(E'_a\right)$ is the span of d uniformly distributed vectors of $\mathbb{F}^m$, and therefore by Proposition 2.20, it holds that conditioned on $\dim\left(\text{left}\left(E'_a\right)\right) = d$ the subspace $\text{left}\left(E'_a\right)$ is a uniformly distributed d-subspace of $\text{left}\left(E_b\right)$. This implies that the probability

$$\Pr\left[\text{left}\left(E'_a\right) = D\,\middle|\,\dim\left(\text{left}\left(E'_a\right)\right) = d\right]$$

is the same for all possible choices of D, and therefore the probability

$$\Pr\left[\text{left}\left(E_a\right) = D\,\middle|\,\dim\left(\text{left}\left(E_a\right)\right) = d\right]$$

is the same for all possible choices of D, as required.　　　□

6. Decodable PCPs

The PCP theorem says that CIRCUITSAT has a proof system in which the (randomized) verifier reads only $O(1)$ bits from the proof. In known constructions, this proof is invariably an *encoding* of a satisfying assignment to the input circuit. Although this is not stipulated by the classical definition of a PCP, the fact that a PCP is really an encoding of a 'standard' NP witness is sometimes useful. Various attempts to capture this behavior gave rise to such objects as PCPs of proximity (PCPPs, Ben-Sasson *et al.* 2006) or assignment testers (Dinur & Reingold 2006) and more recently to decodable PCPs (dPCPs, Dinur & Harsha 2009).

Application: alphabet reduction through composition. The notion of dPCPs is useful for reducing the alphabet size of PCPs with small soundness error via composition. They were introduced in Dinur & Harsha (2009) in an attempt to simplify and modularize the construction of Moshkovitz & Raz (2008). Indeed, this notion is a refinement of Moshkovitz & Raz (2008)'s so-called locally decode or reject codes (LDRCs) that allowed Dinur & Harsha (2009) to prove a generic two-query composition theorem. This theorem allows one to improve parameters of a PCP using any dPCP. The only known construction of a dPCP (until

this work) is the so-called manifold vs. point construction. In the next sections, we give a new construction of a dPCP by adapting the work of the previous sections to a dPCP. Our dPCP can then be plugged into the composition scheme of Dinur & Harsha (2009) to reprove the result of Moshkovitz & Raz (2008). We sketch this in Section 6.5.

Decodable PCPs and PCPs of proximity (PCPPs).   We can define dPCPs for any NP language but we focus on the language CircuitSat since it suffices for our purposes. A dPCP system for CircuitSat is a proof system in which the satisfying assignments of the input circuit are *encoded* into a special "dPCP" format. These encodings can then be both locally verified and locally decoded in a probabilistic manner. In other words, the verifier is given an input circuit as well as oracle access to a proof string and is able to simultaneously check that the given string is a valid encoding of a *satisfying* assignment, as well as to decode a random symbol in that assignment. The formal definition is given in Section 6.2.

dPCPs are closely related to PCPs of proximity (Ben-Sasson *et al.* 2006) or assignment testers (Dinur & Reingold 2006) (to be defined shortly below). In fact, dPCPs were first defined in the context of low soundness error to overcome inherent limitations of PCPPs in this parameter range. In this work, we extend the definition of a dPCP also to the high soundness error range (i.e., matching the parameter range of PCPPs). We call these uniquely decodable PCPs (udPCPs) as opposed to list decodable dPCPs. It is natural to consider such an object in our context since our approach is to reduce the error by parallel repetition. Thus, we start with a dPCP with relatively high error and then reduce the error. Uniquely decodable PCPs turn out to be roughly equivalent to PCPPs in the sense that any PCPP can be used to construct a udPCP and vice versa. In retrospect, we find the notion of udPCPs (and dPCPs) just as natural as that of PCPPs. In fact, many known constructions of PCPPs work by implicitly constructing a udPCP and then adding comparison checks.

As mentioned above, our main goal in Section 6, Section 7, and Section 8 is to give a new construction of dPCPs with low

soundness error (Theorem 1.6). Our construction of dPCPs with low soundness error follows the same steps as our construction of PCPs with low soundness error: In the first step, we construct a dPCP with high soundness error (that is, a udPCP). In the second step, we apply derandomized parallel repetition to the foregoing udPCP to reduce its soundness error to a subconstant function.

In the following subsections, we recall the definitions of PCPPs (Section 6.1) and define udPCPs (Section 6.2). We then prove the equivalence of PCPPs and udPCPs. Next, we state two lemmas that capture the two main steps in constructing dPCPs. This is followed by a proof of Theorem 1.6 (construction of dPCPs). Finally, we sketch a proof of Theorem 1.2 (the Moshkovitz & Raz (2008) result) based on Theorem 1.6.

6.1. Recalling the Definition of PCPPs.   PCPs of proximity (PCPPs) were defined simultaneously in Ben-Sasson *et al.* (2006) and in Dinur & Reingold (2006) under the name assignment testers. PCPPs allow the verifier to check not only that a given circuit is satisfiable but also that a given assignment is (close to being) satisfying. They were introduced for various motivations, and in particular, they facilitate composition of PCPs that is important for constructing PCPs with reasonable parameters.

Intuitively, a PCP verifier for CIRCUITSAT is an oracle machine V that is given as input a circuit $\varphi : \{0,1\}^t \to \{0,1\}$ and is also given oracle access to an assignment x to φ and a proof π. The verifier V is required to verify that x is close to a satisfying assignment of φ, and to do so by making only few queries to x and π. For technical reasons, it is often preferable to define V in a different way. In this definition, instead of requiring that V makes few queries to a oracle and decides according to the answers it gets, we require that V outputs explicitly the queries it intends to make and the predicate ψ it intends to apply to the answers it gets. The advantage of this definition is that it allows us to measure the complexity of the predicate ψ. The formal definitions of PCPP are given below.

DEFINITION 6.1 (PCPP verifier).   *A PCPP verifier for* CIRCUITSAT *is a probabilistic polynomial time algorithm V that*

on input circuit $\varphi : \{0,1\}^t \to \{0,1\}$ of size n tosses $r(n)$ coins and generates

   (i) $q = q(n)$ queries $I = (i_1, \ldots, i_q)$ in $[t + \ell]$ (where $\ell = \ell(n)$ and the queries are viewed as coordinates of a string in $\{0,1\}^{t+\ell}$).

  (ii) A circuit $\psi : \{0,1\}^q \to \{0,1\}$ of size at most $s(n)$.

We shall refer to $r(n), q(n), \ell(n),$ and $s(n)$ as the randomness complexity, query complexity, proof length, and decision complexity respectively.

DEFINITION 6.2 (PCPPs). Let $V, r(n), q(n), \ell(n),$ and $s(n)$, be as in Definition 6.1, and let $\rho : \mathbb{N} \to (0, 1]$. We say that V is a PCPP system for CIRCUITSAT$_{\{0,1\}}$ with rejection ratio ρ if the following holds for every circuit $\varphi : \{0,1\}^t \to \{0,1\}$ of size n:

○ **Completeness:** For every satisfying assignment x for φ, there exists a proof string $\pi_x \in \{0,1\}^\ell$ such that

$$\Pr_{I,\psi} \left[\psi \left((x \circ \pi_x)_{|I} \right) = 1 \right] = 1,$$

where I and ψ are the (random) output of $V(\varphi)$.

○ **Soundness:** For every $x \in \{0,1\}^t$ that is ε-far from a satisfying assignment to φ and every proof string $\pi \in \{0,1\}^\ell$, the following holds:

$$\Pr_{I,\psi} \left[\psi \left((x \circ \pi)_{|I} \right) = 0 \right] \geq \rho \cdot \varepsilon.$$

The starting point for our construction of a dPCP is the fact that NP has PCPPs with reasonable parameters:

THEOREM 6.3 (Ben-Sasson *et al.* 2006; Dinur & Reingold 2006). CIRCUITSAT$_{\{0,1\}}$ has a PCPP system with randomness complexity $O(\log n)$, query complexity $O(1)$, proof length $\text{poly}(n)$, decision complexity $O(1)$, and rejection ratio $\Omega(1)$.

REMARK 6.4. *The PCPPs described in* Definition 6.2 *are known in the literature as "strong PCPPs". Here, the term "strong" means that the rejection probability is linearly related to the distance ε of x from a satisfying assignment. In particular, this implies that even if ε is small (but non-zero), then the PCPP rejects with non-zero probability.*

An alternative definition of PCPPs, known as "weak PCPPs", requires only that every assignment $x \in \{0,1\}^t$ that is very far from a satisfying assignment will be rejected with high probability, while $x's$ that are close to a satisfying assignment may be accepted with probability 1.

6.2. The Definition of Decodable PCPs. Decodable PCPs (dPCPs) were defined in the work of Dinur & Harsha (2009) in order to overcome certain limitations of PCPPs.[4] As mentioned above, the definition of Dinur & Harsha (2009) is only useful if the soundness error is indeed very low. Below, we recall the definition of Dinur & Harsha (2009) and suggest an alternative definition for the case where the soundness error is high. This alternative definition will be useful later in the construction of decodable PCPs with low soundness error.

6.2.1. Recalling the Definition of Dinur & Harsha (2009). Intuitively, a PCP decoder for CIRCUITSAT is an oracle machine D that is given as input a circuit φ and is also given oracle access to a "proof" π that is supposed to be the encoding of some *satisfying* assignment x to φ. The PCP decoder D is required to decode a uniformly distributed coordinate k of the assignment x by making only few queries to π. It could also be the case that the proof π is too corrupted for the decoding to be possible, in which case D is allowed to output a special failure symbol $\bot$. Thus, we say that D has made an error only if it outputs a symbol other than x_k and $\bot$. We refer to the probability of the latter event as the "decoding error of D" and would like it to be minimal. We do

---

[4]In particular, using arguments in the spirit of Ben-Sasson *et al.* (2009), it is easy to prove that a PCPP that has low soundness error must make at least three queries. Hence, PCPPs can not be used to construct two-query PCPs with low soundness error.

note, however, that if π is not corrupted, then D is not allowed to output $\perp$.

It turns out that if we wish the decoding error of D to be very small, we need to relax the foregoing definition and allow the PCP decoder D to perform "list-decoding". That is, instead of requiring that there would be a single assignment x that is decoded by D, we only require that there exists a *short* list of assignments $x^1, \ldots, x^L$ such that the decoder outputs either $\perp$ or one of the symbols $x_k^1, \ldots, x_k^L$ with very high probability. Of course, this is meaningless if the assignments are binary strings, and therefore, we extend the definition of CIRCUITSAT to circuits whose inputs are symbols from some large alphabet Γ.

We turn to give the formal definitions of (list-)decodable PCPs. As in the case of PCPPs, instead of letting the decoder make the queries and process the answers directly, we require the decoder to output the queries and a circuit ψ that given the answers to the queries outputs the decoded value.

NOTATION 6.5. *Let Σ and Γ be finite alphabets, and let $f : \Gamma^k \to \Sigma^n$ be a function. We say that a circuit C computes f if it takes as input a binary string of length $k \cdot \lceil \log |\Gamma| \rceil$ and outputs a binary string of length $n \cdot \lceil \log |\Sigma| \rceil$ that represent the input in Γ^k and the output in Γ^n in the natural way. We will usually omit the function f and simply refer to the circuit $C : \Gamma^k \to \Sigma^n$. We will also view the circuit C as taking as input k symbols in Γ and outputs n symbols in Σ. Given a circuit $\varphi : \Gamma^t \to \{0,1\}$, an assignment $x \in \Gamma^t$ for φ is said to satisfy φ if $\varphi(x)$, and otherwise, it is said to be* unsatisfying.

DEFINITION 6.6 (PCP decoders, similar to Dinur & Harsha 2009, Definition 3.1). *Let $r, q, s, \ell : \mathbb{N} \to \mathbb{N}$, and let Γ, Σ be functions that map each $n \in \mathbb{N}$ to some finite alphabet. A PCP decoder for CIRCUITSAT$_\Gamma$ over proof alphabet Σ is a probabilistic polynomial time algorithm D that for every $n \in \mathbb{N}$ acts as follows. Let $\Gamma = \Gamma(n), \Sigma = \Sigma(n), \ell = \ell(n)$. When given as input an input circuit $\varphi : \Gamma^t \to \{0,1\}$ of size n and an index $k \in [t]$, the PCP decoder D tosses $r(n)$ coins and generates*

(i) *A sequence of queries $I = \left(i_1, \ldots, i_{q(n)}\right)$ in $[\ell]$ (where the queries are viewed as coordinates of a proof string in Γ^ℓ).*

(ii) *A circuit $\psi : \Sigma^{q(n)} \to \Gamma \cup \{\bot\}$ of size at most $s(n)$.*

We shall refer to the functions $r(n), q(n), \ell(n)$, and $s(n)$ as the randomness complexity, query complexity, proof length, and decoding complexity respectively. Without loss of generality, we have $\ell(n) = 2^{r(n)} \cdot q(n) \cdot t$.

DEFINITION 6.7 (List Decodable PCPs, similar to Dinur & Harsha 2009). *Let D, Γ, Σ, and ℓ be as in Definition 6.6, and $L : \mathbb{N} \to \mathbb{N}$ and $\varepsilon : \mathbb{N} \to [0, 1]$. We say that a PCP decoder D with the foregoing parameters is a* (list) decodable PCP system *for* CIRCUITSAT$_\Gamma$ *(abbreviated ldPCP) with list size $L = L(n)$, soundness error $\varepsilon = \varepsilon(n)$ if the following holds for every circuit $\varphi : \Gamma^t \to \{0, 1\}$ of size n:*

∘ **Completeness:** *For every $x \in \Gamma^t$ such that $\varphi(x) = 1$, there exists a proof string $\pi_x \in \Sigma^\ell$ such that*

$$\Pr_{k;I,\psi} \left[\psi\left(\pi_{x|I}\right) = x_k \right] = 1,$$

where k is uniformly distributed in $[t]$ and I and ψ are the (random) output of $D(\varphi, k)$.

∘ **Soundness:** *For every proof string $\pi \in \Sigma^\ell$, there exist a (possibly empty) list of satisfying assignments $x^1, \ldots, x^L \in \Gamma^t$ for φ such that*

$$\Pr_{k;I,\psi} \left[\psi\left(\pi_{|I}\right) \notin \left\{ x_k^1, \ldots, x_k^L, \bot \right\} \right] \leq \varepsilon,$$

where k, I, ψ are as before.

6.2.2. Uniquely Decodable PCPs. We turn to discuss our suggested definition for dPCPs for the case of high soundness error. If the soundness error is high, then we can actually require the PCP decoder to decode a unique assignment, instead of decoding a list of assignments. Thus, we refer to dPCPs with high soundness error as "uniquely decodable PCPs" (udPCPs).

The straightforward definition for udPCPs would be to take the foregoing definition of ldPCPs and set ε to be large and L to be 1. However, this definition turns out to be useless for our purposes. To see why, recall that our ultimate goal is to construct dPCPs with *low error* by first constructing dPCPs with *high error* and then decreasing their error using derandomized parallel repetition. However, if we define udPCPs using the above straightforward definition, then it is not even clear that *sequential repetition* decreases their error.[5]

We therefore use the following alternative definition for udPCP. We now require that if the proof π is such that the PCP decoder D errs with high probability, then D detects that there is an error with at least proportional probability. In other words, we require that the probability that D outputs $\perp$ is related to the probability that D errs. Observe that such PCP decoders can indeed be improved by sequential repetition: If the proof π is erroneous and we invoke the PCP decoder D many times, then the probability that D detects the error and outputs $\perp$ improves. Below, we give the formal definition.

DEFINITION 6.8. *Let* D, Γ, Σ, *and* ℓ *be as in* Definition 6.6. *Let* $\varphi : \Gamma^t \to \{0,1\}$ *be a circuit of size* n, *let* x *be an assignment to* φ, *and let* $\pi \in \Sigma^{\ell(n)}$ *be a proof for* D. *We define the* decoding error *of* D *on* π *with respect to* x *as the probability*

$$\Pr_{k; I, \psi} \left[\psi \left(\pi_{|I} \right) \notin \{x_k, \perp\} \right],$$

where k, I, ψ *are as in* Definition 6.7. *We define the* decoding error *of* D *on* π *as the minimal decoding error of* D *on* π *with respect to an assignment* x' *for* φ, *over all possible assignments* x' *to* φ.

DEFINITION 6.9 (Uniquely Decodable PCPs). *Let* D, Γ, Σ, *and* ℓ *be as in* Definition 6.6, *and let* $\rho : \mathbb{N} \to [0,1]$. *We say that the PCP*

---

[5]The problem in performing sequential repetition for such definition of udPCPs is that we must invoke the PCP decoder on a *uniformly distributed and independent index* k in each invocation, and it is not clear how to use invocations for different indices k in order to decrease the error.

decoder D is a (uniquely) decodable PCP system for $\textsc{CircuitSat}_\Gamma$ (abbreviated udPCP) with rejection ratio ρ if for every circuit $\varphi : \Gamma^t \to \{0,1\}$ of size n the PCP decoder D satisfies the completeness requirement of Definition 6.7 and furthermore satisfies the following requirement:

○ **Soundness:** For every proof string $\pi \in \Sigma^\ell$, if D has decoding error ε on π, then

$$\Pr_{k;I,\psi} \left[\psi \left(\pi_{|I} \right) = \bot \right] \geq \rho(n) \cdot \varepsilon,$$

where k, I, ψ are as in Definition 6.7.

REMARK 6.10. *We could have also defined the decoding error of D on π with respect to x as the probability $\Pr_{k;I,\psi} \left[\psi \left(\pi_{|I} \right) \neq x_k \right]$. This definition may be more natural, but it is more convenient to work with the current definition.*

REMARK 6.11. *Note that the soundness requirement in our definition of udPCPs is similar to the soundness requirement of PCPPs and, in particular, to definition of soundness of strong PCPPs (see Remark 6.4). We could also use a definition that is analogous to the definition of a weak PCPP. Specifically, we could have required only that when the decoding error is very large, the decoder rejects with high probability. However, our definition is stronger, and since we can satisfy it, we prefer to work with it. It is also more convenient to work with this definition throughout this work.*

We next argue that every PCPP implies a udPCP.

PROPOSITION 6.12. *Let V be a PCPP system for $\textsc{CircuitSat}_{\{0,1\}}$ with randomness complexity $r(n)$, query complexity $q(n)$, proof length $\ell(n)$, decision complexity $s(n)$, and rejection ratio $\rho(n)$. Then, for every $u : \mathbb{N} \to \mathbb{N}$, there exists a udPCP for $\textsc{CircuitSat}_{\{0,1\}^{u(n)}}$ with proof alphabet $\{0,1\}$, randomness complexity $r(n)$, query complexity $q(n) + u(n)$, proof length $n + \ell(n)$, decoding complexity $s(n) + O(u(n))$, and rejection ratio $\rho(n)/u(n)$.*

PROOF.   Let $u : \mathbb{N} \to \mathbb{N}$ and denote $u = u(n)$. For every circuit
$\varphi : (\{0,1\}^u)^t \to \{0,1\}$ of size n and satisfying assignment x for φ,
we define the corresponding proof string for D to be $x \circ \pi_x$, where
π_x is the proof string of V for x when x is treated as a binary
string.

Fix a circuit $\varphi : (\{0,1\}^u)^t \to \{0,1\}$ and $k \in [t]$, and let $x' \in \{0,1\}^{u \cdot t}, \pi \in \{0,1\}^\ell$. On input (φ, k) and oracle access to a proof
$x' \circ \pi$, the decoder D first emulates the verifier V on φ with oracle
access to $x' \circ \pi_x$. If V rejects, then D outputs $\bot$. Otherwise, D
queries the coordinates

$$u \cdot (k-1) + 1, \ldots, u \cdot k$$

of x and outputs the tuple of answers as the symbol in $\{0,1\}^u$ that
it is ought to decode.

It should be clear that D satisfies the completeness require-
ments and has the correct randomness complexity, query complex-
ity, proof length, and decoding complexity.

It remains to analyze the rejection ratio of D. Let π' be a proof
string for D and assume that $\pi' = x \circ \pi$ where $x \in \{0,1\}^{u \cdot t}$ and
$\pi \in \{0,1\}^\ell$. Let x_0 be the satisfying assignment of φ that is nearest
to x when viewed as a binary string. Let ε be the relative distance
between x and x_0 when viewed as strings over the alphabet $\{0,1\}^u$.
Clearly, the decoding error of D on $x \circ \pi$ with respect to x_0 is ε
and is an upper bound on the decoding error of D. Furthermore,
the relative distance between x and x_0 as binary strings is at least
ε/u. Thus, the emulation of V rejects $x \circ \pi$ with probability at
least $\rho(n) \cdot \varepsilon/u$, and this is also the rejection probability of D, as
required.   □

REMARK 6.13. *One could also prove Proposition 6.12 without a
loss of a factor of u in the rejection ratio ρ using error correcting
codes.*

REMARK 6.14. *It is not hard to see that the converse of
Proposition 6.12 also holds. Namely, given a udPCP, it is easy to
construct from it a PCPP. Roughly, given a udPCP D, construct
a PCPP verifier that when given oracle access to $x \circ \pi$, invokes D*

with oracle access to π on a uniformly distributed k and verifies that the output of D equals x_k.

REMARK 6.15. *Our definition of udPCPs (Definition 6.9) bears some similarities to the notion of relaxed locally decodable codes (Ben-Sasson et al. 2006), which are also constructed using PCPPs. However, the notions are fundamentally different. The most important difference between the notions is that while the decoder of a relaxed LDC should decode any possible message, the decoder of a udPCP is required to decode only satisfying assignments of a given circuit. This makes udPCPs significantly more powerful and in fact makes them equivalent to PCPPs. A secondary difference is that when a udPCP is given oracle access to a corrupted oracle, it can output $\perp$ with any probability, while a relaxed LDC is required to output x_k (instead of $\perp$) with some given probability.*

6.3. Decoding Graphs.

6.3.1. The Definition of Decoding Graphs. Recall that in the first part of the paper, we often found it more convenient to work with constraint graphs instead of working with PCPs. We now define the notion of "decoding graphs", which will serve as the graph analog of decoding PCPs just as constraint graphs serve as the graph analog of PCPs.

DEFINITION 6.16 (Decoding graphs). *A (directed) decoding graph is a directed graph $G = (V, E)$ that is augmented with the following objects:*

(i) *A circuit $\varphi : \Gamma^t \to \{0, 1\}$, to which we refer as the* input circuit. *Here Γ denotes some finite alphabet.*

(ii) *A finite alphabet Σ, to which we refer as the* alphabet *of G.*

(iii) *For each edge $e \in E$, an index $k_e \in [t]$, and a circuit $\psi_e : \Sigma \times \Sigma \to \Gamma \cup \{\perp\}$. We say that e is* associated *with k_e and ψ_e. For $k \in [t]$, we denote by E_k the set of edges associated with k.*

The size of G is the number of edges of G. We say that G has decoding complexity s if all the circuits are of size at most s. It is required that G satisfies the following property:

○ **Completeness:** *For every satisfying assignment $x \in \Gamma^t$ to φ, there exists an assignment $\pi_x : V \to \Sigma$ to G such that the following holds. For every edge (u, v) that is associated with an index $k = k_{(u,v)}$ and a circuit $\psi = \psi_{(u,v)}$, it holds that $\psi(\pi(u), \pi(v)) = x_k$.*

NOTATION 6.17. *We will use the following terminology regarding constraint graphs: Let $G = (V, E)$ be a decoding graph with input circuit $\varphi : \Gamma^t \to \{0, 1\}$ alphabet Σ.*

(i) *Let $(u, v) \in E$ and $\psi = \psi_{(u,v)}$ be and edge its associated circuit, and let $\pi : V \to \Sigma$ be an assignment to G. If ψ outputs $\perp$ on input $(\pi(u), \pi(v))$, then we say that (u, v) rejects π (or that π violates (u, v)), and otherwise, we say that (u, v) accepts π (or that π satisfies (u, v)).*

(ii) *Let $(u, v), \psi$, and π be as before, let $k = k_{(u,v)}$ be the index associated with (u, v), and let x be an assignment to φ. We say that (u, v) fails to decode x if $\psi(\pi(u), \pi(v)) \notin \{x_k, \perp\}$. When x is clear from the context, we will omit it and we will also say that (u, v) errs or that (u, v) decodes correctly (if (u, v) does not err). Note that outputting $\perp$ is not considered to be failure.*

(iii) *We say that G has the* projection property *if for every circuit $\psi_{(u,v)}$ has an associated function $f_{(u,v)} : \Sigma \to \Sigma$ such that $\psi_{(u,v)}(a, b) \neq \perp$ if and only if $f_{(u,v)}(a) = b$.*

(iv) *We refer to the quantity $\log\left(\max_{k \in [t]} |E_k|\right)$ as the randomness complexity of G, since it upper bounds the number of bits required to choose a uniformly distributed edge that is associated with a particular index.*

We turn to define soundness properties of decoding graphs. As in the case of decodable PCPs, we have two definitions, one for the case of high soundness error (unique decoding) and another for the case of low soundness error (list-decoding).

DEFINITION 6.18. *Let* $G = (V, E), \Sigma, \Gamma, \varphi$ *be as before, and let* $\pi : V \to \Sigma$ *be an assignment to* G.

○ **Unique decoding soundness:** *For every satisfying assignment* $x \in \Gamma^t$ *to* φ, *we define the* decoding error *of* G *on* π *with respect to* x *as the probability*

$$\Pr_{k \in [t], (u,v) \in E_k} \left[\psi_{(u,v)} \left(\pi(u), \pi(v) \right) \notin \{x_k, \bot\} \right],$$

where k *is uniformly distributed in* $[t]$ *and* (u, v) *is uniformly distributed in* E_k. *Note that the edge* (u, v) *is chosen according to the decoding distribution of* G.
We define the decoding error *of* G *on* π *as the minimal decoding error of* G *on* π *with respect to any satisfying assignment of* φ. *Now, we say that* G *has* rejection ratio ρ *if for every assignment* π *to* G, *if* G *has decoding error* ε *on* π, *then it holds that*

$$\Pr_{k \in [t], (u,v) \in E_k} \left[\psi_{(u,v)} \left(\pi(u), \pi(v) \right) = \bot \right] \geq \rho \cdot \varepsilon,$$

where k *and* (u, v) *are chosen as before.*

○ **List decoding soundness:** *We say that* G *is* list-decoding *with list size* L *and soundness error* ε *if for every assignment* π *to* G, *there exists a (possibly empty) list of satisfying assignments* $x^1, \ldots, x^L \in \Gamma^k$ *for* φ *such that*

$$\Pr_{k \in [t], (u,v) \in E_k} \left[\psi_{(u,v)} \left(\pi(u), \pi(v) \right) \notin \{x_k^1, \ldots, x_k^L, \bot\} \right] \leq \varepsilon,$$

where k *and* (u, v) *are chosen as before*

The following proposition gives the correspondence between decoding PCPs and decoding graphs, in analogy to the correspondence between PCPs and constraint graphs.

PROPOSITION 6.19. *Let* $r, s, \ell, \rho, \Gamma, \Sigma$ *be as in Definition 6.9. The following two statements are equivalent:*

○ CIRCUITSAT$_\Gamma$ *has a udPCP with query complexity 2, randomness complexity* r, *decoding complexity* s, *proof length* ℓ, *proof alphabet* Σ, *and rejection ratio* ρ.

○ *There exists a polynomial time transformation that transforms a circuit $\varphi : \Gamma^t \to \{0,1\}$ of size n to a decoding graph $G = (V, E)$ with $\ell(n)$ vertices, randomness complexity $r(n)$, decoding complexity $s(n)$, proof alphabet $\Sigma(n)$, and rejection ratio $\rho(n)$.*

A similar equivalence holds for ldPCPs and list-decoding graphs.

6.3.2. Additional Properties of Decoding Graphs. Recall that when discussing constraint graphs, we were interested in the probability that a uniformly distributed edge of the graph is satisfied by a given assignment. As can be seen in Definition 6.18, when discussing decoding graphs, we are interested in a different distribution over the edges, defined below.

DEFINITION 6.20. *The decoding distribution $\mathcal{D}_G$ of a decoding graph $G = (V, E)$ is the distribution over the edges of G that corresponds to the following way for picking a random edge of G: Choose $k \in [t]$ uniformly at randoms and then choose an edge uniformly at random from E_k.*

It is usually inconvenient to analyze the decoding distribution of the graphs we work with. However, we will work only with graphs whose decoding distribution is similar to the uniform distribution over the edges (where similarity is defined as in Section 2.5). The following definition aims to capture this property, which allows us to analyze the uniform distribution instead of the decoding distribution.

DEFINITION 6.21. *We say that a decoding graph $G = (V, E)$ has smoothness γ if its decoding distribution is γ-similar to the uniform distribution over E.*

The following proposition gives a comfortable way for calculating the smoothness of a decoding graph. Intuitively, observe that if all the sets E_k are of the same size, then the decoding distribution is identical to the uniform distribution. We now observe that if the sizes of the sets E_k are close to each other, then the decoding distribution is similar to the uniform distribution.

PROPOSITION 6.22 (Smoothness criterion). *A decoding graph G with edge-set E has smoothness γ if and only if for every $k \in [t]$, the number of edges that are associated with k is between $\gamma \cdot \frac{|E|}{t}$ and $\frac{1}{\gamma} \cdot \frac{|E|}{t}$.*

PROOF.   Observe that if there are m_k edges associated with $k \in [t]$, then the probability for such an edge to be chosen under the decoding distribution is $\frac{1}{t} \cdot \frac{1}{m_k}$ while the corresponding probability under the uniform distribution is $\frac{1}{|E|}$. Now apply the definition of similarity of distributions. $\square$

We will often want our decoding graphs to be regular or at least have bounded degree. The precise definition follows.

DEFINITION 6.23. *We say that a decoding graph G has degree bound $d \in \mathbb{N}$ if all the in-degrees and all out-degrees of the vertices in G are bounded by d. We say that it is d-regular if every vertex has exactly d incoming edges and exactly d outgoing edges.*

6.3.3. General udPCPs and Decoding Graphs.   Proposition Proposition 6.19 gave us only a correspondence between decoding graphs and udPCPs that makes exactly two queries. The next proposition shows that in fact any udPCP, even if it uses more than two queries, gives rise to a procedure that transforms circuits to decoding graphs with related parameters and unique decoding soundness. A nice property of this procedure is that it generates decoding graphs that are regular and have smoothness 1, which will be useful later in this work.

PROPOSITION 6.24. *Let $\Gamma, \Sigma, r(n), q(n), \ell(n), s(n)$, and $\rho(n)$ be as in Definition 6.9, and let h_0 and d_0 be the constants from Fact 2.23. If there exists a udPCP D for $\mathrm{CIRCUITSAT}_\Gamma$ with the foregoing parameters, then there exists a polynomial time procedure that acts as follows. When given a circuit $\varphi : \Gamma^t \rightarrow \{0, 1\}$ of size n, the procedure outputs a corresponding vertex-decoding graph $G = (V, E)$ with randomness complexity $r(n) + \log(d_0 \cdot q(n))$, alphabet $\Sigma^{q(n)}$, decoding complexity $s(n) + \mathrm{poly} \log |\Sigma(n)|$, and rejection ratio $\Omega\left(\rho(n) / \left(q(n)\right)^2\right)$. Furthermore, G is $(q(n) \cdot d_0)$-regular, and has $t \cdot 2^{r(n)}$ vertices and smoothness 1.*

PROOF (Proof sketch).   The proof is a variant of a well-known technique for reducing the query complexity of a PCP verifier to 2, and its full details are provided in Section 6.6. The graph G is constructed roughly as follows: The graph G has a vertex for every possible invocation of the decoder D. Each such vertex v is expected to be labeled with the answers that D receives to its queries on the corresponding invocation, and the edges that are connected to v check that those answers are not rejected by D. The edges of G also verify that the labels of the different vertices are consistent with each other, and in order to save in the number of edges, we choose the consistency checks according to an expander.

Observe that since a vertex should be labeled with all the answers that D gets to its queries on this particular invocation, we can use those labels to perform decoding. In particular, given that an edge (u, v) accepts, the value that it decodes can be decided based only on the label of u. This property will be useful in Section 7 (see Definition 7.1 for details).   □

6.4. Our construction of dPCPs, Theorem 1.6.   In this section, we state and prove Theorem 1.6.

THEOREM (1.6, dPCP, restated formally). *For every function* Γ *that maps natural numbers to finite alphabets such that* $|\Gamma(n)| \le 2^{\text{poly}\log n}$ *the following holds.   There exists an ldPCP D for* CIRCUITSAT$_\Gamma$ *with query complexity 2, proof alphabet* $2^{\text{poly}\log n}$, *randomness complexity* $O(\log n)$, *soundness error* $1/\log^{\Omega(1)} n$, *and list size* poly log n. *Furthermore, D has the projection property (see Notation 6.17, Item iii).*

We prove this theorem analogously to the proof of Theorem 1.1, which asserts the existence of two-query PCPs with soundness error $1/\text{poly}\log n$. Our starting point is a known construction of a PCPP, stated here as Theorem 6.3 which is then reduced to a transformation mapping circuits to decoding graphs. We then have two main steps. The first is to equip the decoding graphs with linear structure, as formulated in Lemma 6.25. The second step is to reduce the error by derandomized parallel repetition, as stated in Lemma 6.26. Theorem 1.6 follows by combining the two lemmas which we state next,

LEMMA 6.25 (Linear Structure Embedding for udPCPs). *There exists a polynomial time procedure that satisfies the following requirements:*

○ **Input:**

  – *A decoding graph G of size n for input circuit $\varphi : \Gamma^t \to \{0,1\}$ with alphabet Σ, rejection ratio ρ, decoding complexity s, and smoothness γ.*

  – *A finite field $\mathbb{F}$ of size q such that $q \geq 4 \cdot d_0^2$, where d_0 is the constant from Fact 2.23.*

○ **Output:** *A decoding graph $G' = (\mathbb{F}^m, E')$ for φ such that the following holds:*

  – *G' has a linear structure.*

  – *The size of G' is at most $O\left(q \cdot n/\gamma\right)$.*

  – *G' has alphabet $\Sigma^{O(\log_q(n/\gamma))}$.*

  – *G' has rejection ratio $\Omega\left(\rho/q^2 \cdot \log_q(n/\gamma)\right)$.*

  – *G' has decision complexity $s + \text{poly}\left(\log_q\left(n/\gamma\right), \log|\Gamma|\right)$.*

  – *G' has smoothness $\Omega\left(1/q\right)$.*

LEMMA 6.26 (Derandomized Parallel Repetition for dPCPs). *There exist a universal constant h and a polynomial time procedure that satisfy the following requirements:*

○ **Input:**

  – *A finite field $\mathbb{F}$ of size q.*

  – *A decoding graph $G = (\mathbb{F}^m, E)$ of size n for input circuit $\varphi : \Gamma^t \to \{0,1\}$ with linear structure, alphabet Σ, rejection ratio ρ, decision complexity s, and smoothness γ.*

  – *The rejection ratio ρ of G.*

  – *A parameter $d_0 \in \mathbb{N}$ such that $d_0 < m/h^2$ and $\rho \geq h \cdot d_0 \cdot q^{-d_0/h}/\gamma$.*

○ **Output:** *A decoding graph G' for φ such that the following holds:*

  – *G' has size $n^{O(d_0)}$.*

  – *G' has alphabet $\Sigma^{q^{O(d_0)}}$.*

  – *G' is list-decoding with soundness error $\varepsilon \stackrel{\text{def}}{=} h \cdot d_0 \cdot q^{-d_0/h}/\gamma$ and list size $L \stackrel{\text{def}}{=} q^{O(d_0)}$.*

  – *G' has the projection property.*

  – *G' has decoding complexity $q^{O(d_0)} \cdot (s + \operatorname{poly} \log |\Sigma|)$.*

We now turn to prove Theorem 1.6.

PROOF.   Let V be a PCPP verifier for CIRCUITSAT as in Theorem 6.3. By Proposition 6.12, this implies a udPCP for CIRCUITSAT with similar parameters. Next, by Proposition 6.24, we get a polynomial time transformation taking a circuit $\varphi :$ $\{0,1\}^n \rightarrow \{0,1\}$ into a vertex-decoding graph. The graph G has the following parameters. The randomness complexity is $r(n) = O(\log n)$, the decoding complexity, rejection ratio, and constant proof alphabet are constant, and the smoothness is 1.

We choose $\mathbb{F}$ to be the smallest finite field of size at least $\log n$ and set $\mathbb{F}$ to be the finite field of size q. We now invoke Lemma 6.25 (linear structure embedding for udPCPs) on input G and $\mathbb{F}$ and obtain a new vertex-decoding graph G_1 with linear structure and parameters:

○ The size of G_1 is at most $O(q \cdot n)$.

○ G_1 has alphabet size $2^{O(\log_q(n))}$.

○ G_1 has rejection ratio $\rho_1 \stackrel{\text{def}}{=} \Omega\left(\rho/q^2 \cdot \log_q(n)\right)$.

○ G_1 has decision complexity $\operatorname{poly}(\log_q n)$.

○ G_1 has smoothness $\gamma_1 = \Omega\left(\frac{1}{q}\right)$.

Finally, we set d_0 to be an arbitrary constant such that $\rho_1 \geq h \cdot d_0 \cdot q^{-d_0/h}/\gamma_1$. Note that this is indeed possible, since $\log_q(1/\rho_1)$ is a

constant that depends only on ρ. Finally, we invoke Lemma 6.26 (derandomized parallel repetition for dPCPs) on input $G_1, \mathbb{F}, \rho_1$, and d_0 and denote by G' the output decoding graph. The transformation taking the initial input φ into G' (via intermediate steps G and G_1) is equivalent, by Proposition 6.19, to a dPCP with the claimed parameters. $\square$

6.5. Proof of the Result of Moshkovitz & Raz (2008), Theorem 1.2. Our Theorem 1.1 asserts the existence of a two-query PCP with soundness error $1/\text{poly} \log n$ and alphabet size $|\Sigma| = 2^{\text{poly} \log n}$. In this section, we will sketch a proof of Theorem 1.2 in which the alphabet size $|\Sigma|$ can be any value smaller than $2^{\text{poly} \log n}$ while maintaining the relation of $\varepsilon \leq 1/\text{poly}(\log |\Sigma|)$.

THEOREM (**1.2**, restated, Moshkovitz & Raz 2008). *For any function* $\varepsilon(n) \geq 1/\text{poly} \log n$, *the class* **NP** *has a two-query PCP verifier with perfect completeness, soundness error at most* ε *over alphabet* Σ *of size at most* $|\Sigma| \leq 2^{1/\text{poly}(\varepsilon)}$.

Our proof of Theorem 1.2 relies on the scheme of Dinur & Harsha (2009) who showed a generic way to compose a PCP with a dPCP and then proved Theorem 1.2 by repeating the composition step, assuming the existence of two building blocks: a PCP and a dPCP. We plug in our constructions of a PCP (Theorem 1.1) and of a dPCP (Theorem 1.6) into the composition scheme of Dinur & Harsha (2009) and obtain a new construction of the verifier of Theorem 1.2 that does not rely on low-degree polynomials.

REMARK 6.27. *An important feature of the theorem of Moshkovitz & Raz (2008) asserts that the verifier is randomness efficient, i.e., it uses only* $(1 + o(1)) \log n$ *random bits rather than* $O(\log n)$ *random bits. This is equivalent to constructing constraint graphs of almost-linear size rather than polynomial size (see Remark 2.17). Using the composition scheme of Dinur & Harsha (2009), the outcome will be randomness efficient as long as the PCP verifier at the outermost level of composition is randomness efficient. It does not, for example, depend on whether the dPCP is randomness efficient.*

However, since our PCP verifier from Theorem 1.1 is not randomness efficient, we can only get this additional feature by relying at the outermost level on a PCP verifier as in Moshkovitz & Raz (2008). The dPCP can still be based on our Theorem 1.6. Alternatively, if we also base the outermost PCP on Theorem 1.1, we get a polynomial size construction, but not a "randomness efficient" one. It is also conceivable that the construction of Theorem 1.1 can be improved to yield a randomness efficient PCP, and we leave this for future work.

In order to state the generic composition theorem of Dinur & Harsha (2009), let us first define the *decision complexity* of a PCP verifier. Roughly speaking, a PCP verifier has decision complexity $s(n)$ if every constraint in the underlying constraint graph can be computed by a circuit of size at most $s(n)$.[6] This definition is analogous to the definition of the decoding complexity of a PCP decoder. It is easy to see that the PCP verifier (from Theorem 1.1) has decision complexity poly $\log n$ in the same way that the dPCP decoder (from Theorem 1.6) was shown to have decoding complexity poly $\log n$.

We turn to state the composition theorem of Dinur & Harsha (2009). As in all composition theorems in the literature, the goal of this theorem is to take an "outer verifier" (in this case, a PCP verifier), which has a large alphabet, and reduce its alphabet size by composing it with an "inner verifier" (in this case, a PCP decoder). The gain is obtained from the fact that the inner verifier is invoked on a claim of size $s(n) \ll n$ and thus can have a much smaller alphabet than the outer verifier. The result of the composition is a verifier that has the alphabet size roughly as of the inner verifier and can still be invoked on a claim of size n. However, the composed verifier accumulates soundness error from the invocations of both the outer verifier and the inner verifier, and thus, the composition does not come "for free".

THEOREM 6.28 (Paraphrasing Dinur & Harsha 2009). *Let V and D be a PCP verifier and a PCP decoder as follows:*

---

[6]More precisely, the verifier should be able to compute this circuit based on its input and its randomness.

(i) *Let V be a two-query PCP verifier for **NP** with perfect completeness, soundness error $\Delta(n)$, alphabet size $|\Sigma(n)|$, and decision complexity $s(n)$. Assume further that the PCP verifier makes projection queries.*

(ii) *Let D be a two-query PCP decoder for CIRCUITSAT_Γ for some $\Gamma(n)$. Assume D has perfect completeness, soundness error $\delta(n)$, list size $L(n)$, and alphabet size $|\sigma(n)|$.*

If both V and D have the projection property, then there is a PCP verifier $V \circledast D$ with the following properties. $V \circledast D$ invokes D on inputs of length at most $s(n)$. $V \circledast D$ has perfect completeness, soundness error $O(\delta(s(n)) + L(s(n))\Delta(n))$, alphabet size $|\sigma(s(n))|^{\text{poly}(L(s(n))/\delta(s(n)))}$, and $V \circledast D$ has the projection property.

As discussed above, the main gain from this theorem is that the alphabet size of $V \circledast D$ is much smaller than that of V. Let us see how this is useful. Suppose we take V, D from Theorem 1.1 and Theorem 1.6. We have $\Sigma(n) \leq 2^{\text{poly} \log n}, s(n) = \text{poly} \log n$, and $\sigma(n) \leq 2^{\text{poly} \log n}$. Thus, $\sigma(s(n)) = 2^{\text{poly} \log \log(n)}$. Similarly $L(s(n)) \leq \text{poly} \log \log n$ and $\delta(s(n)) = 1/\text{poly} \log \log n$. This results in alphabet size of $2^{\text{poly} \log \log(n)}$ and soundness error of $1/\text{poly} \log \log n$. By composing this verifier again with D (yielding $(V \circledast D) \circledast D$), one can inductively obtain a PCP verifier with soundness error $1/\text{poly} \log^{(i)} n$ for any i and corresponding alphabet size $|\Sigma| = 2^{1/\text{poly}(\epsilon)}$. To get *any* alphabet size $|\Sigma|$, one must do careful padding and we do not go into these details.

The composition theorem (Theorem 6.28) is stated here in the two-query terminology (rather than in the terminology of "robust" PCPs). Let us now give a brief outline of how to obtain this version from the version of Dinur & Harsha (2009):

1. *From two-query to robust:* Use Lemma 2.5 of Dinur & Harsha (2009) to deduce the existence of a robust PCP rV and a robust dPCP rD with parameters related to V and D. In particular, the number of accepting views for rD is bounded by $|\sigma|$.

2. *Composition:* Apply Theorem 4.2 of Dinur & Harsha (2009) with parameter $\varepsilon = \delta/L \geq |\sigma|^{\Omega(1)}$. Deduce a new robust PCP

$rV \circledast rD$ with parameters as follows. The soundness error is $\delta + L\Delta + 4L\varepsilon = O(\delta + L\Delta)$. The number of accepting views is at most $|\sigma|^{4/\varepsilon^4}$ (this follows from inspecting the proof, but not directly from the theorem statement).

3. *Back to two queries:* Again use Lemma 2.5 to move back to a two-query PCP. The new alphabet size is at most the number of accepting views of $rV \circledast rD$ which is at most $|\sigma(s(n))|^{4/\varepsilon^4} = |\sigma|^{(L/\delta)^{O(1)}}$ as claimed. $\square$

6.6. Proof of Proposition 6.24.   In this section, we prove Proposition 6.24, restated below.

PROPOSITION (6.24, restated).   *Let $\Gamma, \Sigma, r(n), q(n), \ell(n), s(n)$, and $\rho(n)$ be as in Definition 6.9, and let h_0 and d_0 be the constants from Fact 2.23.   If there exists a udPCP D for CIRCUITSAT$_\Gamma$ with the foregoing parameters, then there exists a polynomial time procedure that acts as follows.   When given a circuit $\varphi : \Gamma^t \to \{0,1\}$ of size n, the procedure outputs a corresponding decoding graph $G = (V,E)$ $q(n) \cdot d_0 \cdot t \cdot 2^{r(n)}$ with randomness complexity $r(n) + \log(d_0 \cdot q(n))$, alphabet $\Sigma^{q(n)}$, decoding complexity $s(n) + \text{poly} \log |\Sigma(n)|$, and rejection ratio $\Omega\left(\rho(n)/(q(n))^2\right)$. Furthermore, G is $(q(n) \cdot d_0)$-regular and has $t \cdot 2^{r(n)}$ vertices and smoothness 1.*

Fix $n \in \mathbb{N}$ and let $r = r(n), q = q(n), \ell = \ell(n), \Sigma = \Sigma(n)$, and $s = s(n)$. We describe the output of the procedure on fixed circuit $\varphi : \Gamma^t \to \{0,1\}$ of size n. The procedure outputs a decoding graph G defined as follows:

○ The vertices set of G is the set $[t] \cdot \{0,1\}^r$, whose elements are identified with all the pairs (k, ω) where $k \in [t]$ is an index to be decoded and ω is a sequence of coin tosses of D on input (φ, k). We denote by $I_{(k,\omega)}$ and $\psi_{(k,\omega)}$ are the queries tuple and circuit that are output by D on input (φ, k) and coin tosses ω.

○ The alphabet of G is Σ^q.

○ The edges of G are constructed as follows. For every $i \in [\ell]$, we let C_i be the set of pairs (k, ω) such that on $I_{(k,\omega)}$ contains i.

For each $i \in [\ell]$, we consider the expander $G_{|C_i|}$ over $|C_i|$ vertices from Fact 2.23 and identify its vertices with the elements of C_i. Now, for each undirected edge of $G_{|C_i|}$, we put two directed edges between the corresponding vertices in C_i, one edge per direction.

○ If an edge is coming out from a vertex (k, ω), then it is associated with the index k.

○ The circuits ψ_e associated with the edges are constructed as follows. Let e be an edge going from (k_1, ω_1) to (k_2, ω_2), and let ψ_e be the associated circuit. Suppose that (k_1, ω_1) and (k_2, ω_2) belong to C_i, so there exist $j_1, j_2 \in [q]$ such that $(I_{(k_1,\omega_1)})_{j_1} = (I_{(k_2,\omega_2)})_{j_2} = i$. Now, the circuit ψ_e is given as input two tuples $a, b \in \Sigma^q$, outputs $\perp$ if $a_{j_1} \neq b_{j_2}$, and otherwise outputs $\psi_{(k_1,\omega_1)}(a)$.

Let ℓ' and n' denote the numbers of vertices and edges of G. It is easy to see that the decoding graph G has the correct size, randomness complexity, alphabet, decoding complexity, and number of vertices and also that it is $q \cdot d_0$-regular. To see that it has smoothness 1, consider an edge (u, v) that is chosen under the decoding distribution and observe that

○ u is uniformly distributed among the vertices of G.

○ Conditioned on the choice of u, the edge (u, v) is uniformly distributed among the edges of u.

Combining the two above observations with the regularity of G implies that the decoding distribution of G is the uniform distribution over the edges.

We turn to show the completeness of G. Let x be a satisfying assignment for φ, and let $\pi = \pi_x$ be the corresponding proof string for D. We define an assignment Π to the vertices of G by defining $\Pi_{(k,\omega)}$ to be $\pi_{|I_{(k,\omega)}}$. It should be clear that this choice of Π satisfies the requirements.

It remains to analyze the rejection ratio of G. Let Π be an assignment to G. For each vertex (k, ω), if for some $j \in [q]$ it holds that $(I_{(j,\omega)})_j = i$, then we refer to $(\Pi_{(k,\omega)})_j$ as the opinion of (k, ω) on i and also as the j-th opinion of (k, ω). Let π be the proof

string for D defined by setting π_i to be the most popular opinion of a vertex of G on i. Suppose that D has decoding error ε on π and let x be the satisfying assignment to φ that achieves this decoding error. Let ε' be the decoding error of G on Π with respect to x. We show that at least $\frac{\rho}{q} \cdot \varepsilon'$ fraction of the edges of G reject Π, and this will establish the rejection ratio of G.

Let η be the fraction of vertices of G that have an opinion that is inconsistent with π. Clearly, $\varepsilon' \leq \varepsilon + \eta$: To see it, note that for at least $1 - \varepsilon - \eta$ of the vertices (k, ω) of G, it holds that all the opinions of (k, ω) are consistent with π and that D does not err on proof string π and on (k, ω) (i.e., $\psi_{(k,\omega)}\left(\pi_{|I_{(j,\omega)}}\right) \in \{\bot, x_k\}$). Then, observe that all the outgoing edges of such a vertex (k, ω) do not err.

Let k be uniformly distributed over $[t]$. We consider two possible cases. First, consider the case in which $\eta \leq \rho \cdot \varepsilon/2$. By the soundness of D, it holds that D rejects π with probability at least $\rho \cdot \varepsilon$. Thus, at least $\rho \cdot \varepsilon$ fraction of the vertices (k, ω) of G, it holds that D rejects π on (k, ω). This implies that at least $(\rho \cdot \varepsilon - \eta)$ fraction of the vertices (k, ω) of G, it holds that both D rejects π on (k, ω) *and* all the opinions of (k, ω) are consistent with π, in which case all the outgoing edges of (k, ω) reject Π. It follows that the fraction of edges of G that reject Π is at least

$$\rho \cdot \varepsilon - \eta \geq \rho \cdot \varepsilon/2 \geq \frac{1}{2} \cdot \eta + \frac{\rho}{4} \cdot \varepsilon \geq \frac{\rho}{4}(\eta + \varepsilon) \geq \frac{\rho}{4} \cdot \varepsilon',$$

as required.

We turn to consider the case in which $\eta \geq \rho \cdot \varepsilon/2$. By averaging, there exists some $j \in [q]$ such that for at least η/q fraction of the vertices (k, ω) of G it holds that the j-th opinion of (k, ω) is inconsistent with π. For every $i \in [\ell]$, denote by S_i the set of vertices of C_i whose j-th opinion is an opinion on i that is inconsistent with π_i, and observe that

$$\frac{1}{\ell'} \cdot \sum_{i=1}^{\ell} |S_i| \geq \frac{\eta}{q}.$$

Fix $i \in [\ell]$ and denote $\overline{S}_i = C_i \backslash S_i$, and note that since π_i is the plurality vote, it holds that $|S_i| \leq |C_i|/2$. Now, observe that every

edge that goes from S_i to $\overline{S}_i$ or vice versa must reject Π. By the edge expansion of $G_{|C_i|}$, the number of such edges is at least $h_0 \cdot d_0 \cdot |S|$. Since this holds for every $i \in [\ell]$, it follows that the fraction of edges of G that reject Π is at least

$$
\begin{aligned}
\frac{1}{n'} \cdot \sum_{i=1}^{\ell} h_0 \cdot d_0 \cdot |S_i| &= \frac{1}{q \cdot d_0 \cdot \ell'} \cdot \sum_{i=1}^{\ell} h_0 \cdot d_0 \cdot |S_i| \\
&= \frac{h_0}{q \cdot \ell'} \cdot \sum_{i=1}^{\ell} |S_i| \\
&\geq \frac{h_0}{q} \cdot \frac{\eta}{q} \\
&\geq \frac{h_0}{2 \cdot q^2} \cdot \rho \cdot \varepsilon,
\end{aligned}
$$

where the first equality follows since G is $(q \cdot d_0)$-regular. The required result follows. $\square$

7. Decoding PCPs with Linear Structure

In this section, we prove Lemma 6.25, i.e., that every decoding graph G can be embedded on a graph that has linear structure. The heart of the proof is very similar to the proof of the corresponding lemma for constraint graphs (Lemma 3.3) with few adaptations to the setting of decoding graphs. Two important differences are the following:

1. Recall that we prove Lemma 3.3 by embedding the constraint graph G on a de Bruijn graph $\mathcal{DB}$ and that this is done by identifying the vertices of G with the vertices of $\mathcal{DB}$. Furthermore, recall that if $\mathcal{DB}$ has more vertices than G, then some of the vertices of $\mathcal{DB}$ are not identified with vertices of G, and thus we place only trivial constraints on those vertices.
   This construction does not work for decoding graphs. The reason is that in the setting of decoding graphs, every edge needs to be able to decode some index $k \in [t]$. Furthermore, every edge that fails to decode must contribute to the fraction of rejecting edges. Thus, we cannot have many trivial edges.

In order to resolve this issue, we prove a proposition that allows us to ensure that G has exactly the same number of vertices as in $\mathcal{DB}$, see Proposition 7.4 below.

We note that Item 1 is *not* caused by the fact we chose a strong definition of udPCP and not a weak one (see Remark 6.11). Even if we used a weak definition of udPCP, requiring edges to reject only if the decoding error is above some threshold, we still could not use dummy vertices and edges in the embedding, as this would cause the aforementioned threshold to be too large for our purposes.

2. Recall that in the embedding of constraint graphs on de Bruijn graphs we used the expander-replacement technique (Lemma 4.8) to make sure that the graph G has small degree. Since such a lemma was not proved for decoding graphs in previous works, we have to prove it on our own. This is done in Proposition 7.3 below.

The rest of this section is organized as follows. In Section 7.1, we prove the aforementioned Proposition 7.3 and Proposition 7.4. Then, in Section 7.2, we prove Lemma 6.25.

7.1. Auxiliary Propositions. In this section, we prove Proposition 7.3 and Proposition 7.4 mentioned above. In order to state those two propositions, we need to define a special kind of decoding graphs, called "vertex-decoding graphs". The reason is that we only know how to prove Proposition 7.4 for vertex-decoding graphs. Fortunately, we can convert any decoding graph to a vertex-decoding one using Proposition 7.3.

We move to define the notion of vertex-decoding graphs. Intuitively, a decoding graph is vertex-decoding if the value that an edge (u, v) decodes depends only on the labeling of u, while the labeling of v only affects on whether the edge accepts or rejects. The formal definition follows.

DEFINITION 7.1 (Vertex-decoding graphs). *We say that a decoding graph G is a* vertex-decoding graph *if it has the following properties:*

(i) *For every edge (u, v) of G and its associated circuit $\psi = \psi_{(u,v)}$,
    there exists a function $f : \Sigma \to \Gamma$ that satisfies the follow-
    ing: For every assignment π to the vertices of G for which
    $\psi(\pi(u), \pi(v)) \neq \bot$, it holds that $\psi(\pi(u), \pi(v)) = f(\pi(u))$.*

(ii) *Every vertex has at least one outgoing edge. In other words,
     every vertex is capable of decoding at least one index $k \in [t]$.*

REMARK 7.2. *While the property of a graph being vertex-decod-
ing is reminiscent of the projection property, there are two impor-
tant differences. First, note that Item i in Definition 7.1 is weaker
than the projection property, since it only requires that $\pi(u)$ deter-
mines the decoded value and not necessarily $\pi(v)$. Second, note
that Item ii is not required by the projection property and is actu-
ally violated by the known constructions of graphs that have the
projection property.*

We turn to prove Proposition 7.3 and Proposition 7.4. We begin
with Proposition 7.3, which says that we can always reduce the
degree of decoding graphs while paying only a moderate cost in
the parameters. As mentioned above, the proposition also trans-
forms the decoding graph into a vertex-decoding graph.

PROPOSITION 7.3. *Let d_0 be the constant from Fact 2.23, and let
$d = 2d_0$. There exists a polynomial time procedure that acts as
follows:*

○ **Input**: *A decoding graph G of size n for input circuit $\varphi : \Gamma^t \to
  \{0, 1\}$ with alphabet Σ, rejection ratio ρ, decoding complexity s,
  and smoothness γ.*

○ **Output:** *A d-regular vertex-decoding graph G' of size at most
  $d \cdot n/\gamma$ for input circuit φ, alphabet Σ^2, rejection ratio $\Omega(\rho)$,
  decoding complexity $s + \text{poly} \log |\Sigma|$, and smoothness 1. Fur-
  thermore, G' has at most n/γ vertices.*

PROOF (Proof sketch).   We apply the same construction as in the
proof of Proposition 6.24. Let $\varphi : \Gamma^t \to \{0, 1\}$ be the input circuit
of G. The key observation is that G corresponds to a decoder D

that acts on φ such that D has query complexity 2, randomness complexity $\log(n/t \cdot \gamma)$, proof alphabet Σ, rejection ratio ρ, and decoding complexity s. The reason for the foregoing randomness complexity is that by the smoothness of G and by the smoothness criterion of Proposition 6.22, it holds that for every $k \in [t]$, there are at most $n/t \cdot \gamma$ edges that are associated with k, and therefore, choosing a uniformly distributed edge that is associated with G requires $\log(n/(t \cdot \gamma))$ uniformly distributed bits. Now, by applying the construction of the proof of Proposition 6.24 to the decoder D, we obtain a graph G' that satisfies the requirements. The fact that G' is vertex-decoding can be observed by examining the construction of Proposition 6.24 (see also the second paragraph in the above proof sketch of Proposition 6.24). $\square$

We next prove Proposition 7.4, which says that we can increases the number of vertices of a vertex-decoding graph to any size we wish, while paying only a small cost in the parameters. This proposition will be used to ensure that the number of vertices of a decoding graph G is equal to the number of vertices of the de Bruijn graph on which we want to embed G.

PROPOSITION 7.4. *There exists a polynomial time procedure that acts as follows:*

○ **Input:**

  – *A vertex-decoding graph G of size n for input circuit $\varphi : \Gamma^t \to \{0,1\}$ with ℓ vertices, alphabet Σ, rejection ratio ρ, decoding complexity s, degree bound d, and smoothness γ.*

  – *A number $\ell' \in \mathbb{N}$ such that $\ell' \geq \ell$ (given in unary).*

○ **Output:** *Let $c \overset{\text{def}}{=} \lfloor \frac{\ell'}{\ell} \rfloor$ and let d_0 and h_0 be the constants from Fact 2.23. The procedure outputs a vertex-decoding graph G' of size at most $2 \cdot (c+1) \cdot d_0 \cdot n$ for input circuit φ that has exactly ℓ' vertices and also has alphabet Σ, output size $s + \text{poly} \log |\Sigma|$, rejection ratio $\Omega(\gamma^2 \cdot \rho/d^2)$, degree bound $2 \cdot d_0 \cdot d$, and smoothness $\frac{1}{2} \cdot \gamma$.*

Furthermore, if G is d-regular, then G' is $(2 \cdot d_0 \cdot d)$-regular and has rejection ratio $\Omega(\gamma^2 \cdot \rho)$.

PROOF.   [Proof sketch]The basic idea of the proof is as follows. Given the graph G, we construct the graph G' by replacing each vertex v of G with multiple copies of v, such that the total number of vertices becomes ℓ' as required. Each copy of v will be connected to the same edges as the original v. An assignment to G' will be required to assign the same value to all the copies of v: Clearly, if an assignment π' to G' assigns the same value to the copies of each vertex v of G, then in a way π' "behaves" like an assignment to G, and we can use the soundness of G to establish the soundness of G' with respect to π'. In order to verify that the copies of a vertex v are assigned the same value, we will put equality constraints between the copies of v. In order to save edges, the equality constraints are placed according to the edges of an expander, and the analysis goes exactly as in the proof of Proposition 6.24. We use the fact that G is vertex decoding in order to allow the equality constraints to decode values even though they can use only the labeling of a single vertex of G. The rest of this proof consists of the technical details of this construction and is provided in Section 7.3.

$\square$

7.2. Embedding Decoding Graphs on de Bruijn Graphs. In this section, we prove the following proposition, which implies Lemma 6.25 (linear structure embedding for udPCPs) and is analogous to Proposition 4.4 (embedding of constraint graphs on de-Bruijn graphs). The proof follows the steps of Proposition 4.4 with the few adaptations to the setting of decoding graphs. For intuition and a high-level explanation of the proof, we refer the reader to Section 4 and, in particular, to Section 4.2.

PROPOSITION 7.5 (Embedding Decoding Graphs on de-Bruijn Graphs). *Let d_0 be the constant of Fact 2.23. There exists a polynomial time procedure that satisfies the following requirements:*

∘ **Input:**

– *A decoding graph G of size n for an input circuit $\varphi : \Gamma^t \rightarrow \{0,1\}$ with alphabet Σ, rejection ratio ρ, decoding complexity s, and smoothness γ.*

- A finite alphabet Λ such that $|\Lambda| \geq 4 \cdot d_0^2$.
- A natural number m such that $|\Lambda|^m \geq 2 \cdot d_0 \cdot n/\gamma$.

○ **Output:** A decoding graph G' for φ such that the following holds:

  - The underlying graph of G' is the de Bruijn graph $\mathcal{DB}_{\Lambda,m}$.
  - The size of G' is $|\Lambda|^{m+1}$.
  - G' has alphabet $\Sigma^{O(m)}$.
  - G' has rejection ratio $\Omega\left(\rho/|\Lambda|^2 \cdot m\right)$.
  - G' has smoothness at least $\gamma' \stackrel{\text{def}}{=} \Omega\left(\frac{1}{|\Lambda|}\right)$.
  - G' has decision complexity $s + \text{poly}\left(m, \log|\Sigma|\right)$.

Let G, Λ, and m be as in Proposition 7.5, and let $\varphi : \Gamma^t \to \{0, 1\}$ be the input circuit of G. On input G, Λ, and m, the procedure acts as follows. The procedure first constructs a vertex-decoding graph G_1 by applying to G the procedure of Proposition 7.3 and then applying to the resulting graph the procedure of Proposition 7.4 with $\ell' = |\Lambda|^m$. It can be verified that G_1 is a vertex-decoding graph for input circuit φ with exactly $|\Lambda|^m$ vertices, alphabet $\Sigma_1 \stackrel{\text{def}}{=} \Sigma^2$, rejection ratio $\rho_1 = \Omega(\rho)$, decoding complexity $s + \text{poly} \log |\Sigma|$, and smoothness at least $\frac{1}{2}$. Furthermore, G_1 is d-regular for $d = 4 \cdot d_0^2 \leq |\Lambda|$ and is of size $d \cdot |\Lambda|^m$.

Then, the procedure identifies the vertices of G_1 with the vertices of $\mathcal{DB} = \mathcal{DB}_{\Lambda,m}$, partitions the edges of G_1 to d matchings $\mu_1, \ldots, \mu_d$, and views those matchings as permutations on the vertices of $\mathcal{DB}$. We apply Fact 4.5 to each permutation μ_i resulting in a set of paths $\mathcal{P}_i$ of length $l \stackrel{\text{def}}{=} 2m$. Let $\mathcal{P} = \bigcup \mathcal{P}_i$.

Next, the procedure constructs G' in the following way. The alphabet of G' is set to be $\Sigma_1^{l \cdot d}$, viewed as $\left(\Sigma_1^l\right)^d$. If $\sigma \in \left(\Sigma_1^l\right)^d$, and $\sigma = (\sigma_1, \ldots, \sigma_d)$, we denote by $\sigma_{i,j}$ the element $(\sigma_i)_j \in \Sigma_1$. It remains to describe how to associate each edge e of G' with an index $k_e \in [k]$ and with a circuit ψ_e. To this end, we first describe in which cases a circuit ψ_e accepts and then describe

how the index k_e is chosen and what is the output of ψ_e when it accepts.

The conditions in which ψ_e accepts.   Fix an edge $e' = (u, v)$ of G', and let ψ_e be the circuit associated with e. The circuit ψ_e accepts in exactly the same cases in which the constraint that corresponds to e in the proof of Proposition 4.4 (for constraint graphs) accepts. That is, the circuit ψ_e accepts if and only if all of the following conditions hold:

1. For every $i \in [d]$, the values $\left(\pi'(u)_{i,l}, \pi'(u)_{i,1} \right)$ satisfy the edge $\left(\mu_i^{-1}(u), u \right)$ of G.

2. It holds that $\pi'(u)_{1,1} = \cdots = \pi'(u)_{d,1}$ and that $\pi'(v)_{1,1} = \cdots = \pi'(v)_{d,1}$.

3. For every $i \in [d]$ and $j \in [l-1]$ such that u and v are the j-th and $(j+1)$-th vertices of a path in $p \in \mathcal{P}_i$ respectively, it holds that $\pi'(u)_{i,j} \neq \pi'(v)_{i,j+1}$.

4. Same as Condition 3, but when v is the j-th vertex of p and u is its $(j+1)$-th vertex.

The choice of k_e and the output of ψ_e.   Fix a vertex u of G'. We describe the way we assign indices k_e to the outgoing edges of u and the output of the circuits ψ_e. We begin by associating each of the $|\Lambda|$ outgoing edges of u in G' with one of the d outgoing edges of u in G_1. This association is done in a "balanced" way—that is, each outgoing edge of u in G_1 is associated with either $\lfloor |\Lambda| / d \rfloor$ or $\lceil |\Lambda| / d \rceil$ edges of u in G'.

Now, let e' be an outgoing edge of u in G', and suppose that it is associated with an outgoing edge e_1 of u in G_1 and that e_1 belongs to the matching μ_i. Let k_{e_1} and ψ_{e_1} be the index and circuit associated with e_1. Recall that since G_1 is vertex-decoding, there exists a function $f_{e_1} : \Sigma_1 \to \Gamma$ such that whenever $\psi_{e_1}(a, b) \neq \bot$, it holds that $\psi_{e_1}(a, b) = f_{e_1}(a)$. We associate e' with the index k_{e_1} and with the circuit $\psi_{e'}$ that is defined for every $a', b' \in \left(\Sigma_1^l \right)^d$ for

which $\psi_{e'}(a, b) \neq \bot$ by

$$\psi_{e'}(a', b') = f_{e_1}\left((a')_{1,1}\right).$$

Note that $\psi_{e'}$ is indeed well defined, since the cases in which $\psi_{e'}$ outputs $\bot$ were defined above.

The parameters of G'.   The size and alphabet of G' are immediate, and the completeness of G' can be established in the same way as in Proposition 4.4 (embedding of constraint graphs on de-Bruijn graphs). It can also be verified that G' has smoothness at least $\gamma' = \frac{1}{2 \cdot |\Lambda|}$ using the smoothness criterion (Proposition 6.22) and a straightforward calculation.

It remains to analyze the rejection ratio of G'. Let π' be an assignment to G' that minimizes the ratio between the probability that a random edge of G' rejects π' (under the decoding distribution) to the decoding error of G' on π'. As in the proof of Proposition 4.4, we may assume that for every vertex u of $\mathcal{DB}$, it holds that $\pi'(u)_{1,1} = \cdots = \pi'(u)_{d,1}$, since otherwise we may modify π' to such an assignment that satisfies this property without increasing the rejection probability or decreasing the decoding error. Let π_1 be the assignment to G_1 defined by $\pi_1(u) = \pi'(u)_{1,1}$. Let ε be the decoding error of G_1 on π_1, and let x be the assignment to φ that achieves this decoding error. Let ε' be the decoding error of G' on π' with respect to x. We show that the rejection probability of G' on π' is at least $\Omega\left(\gamma' \cdot \rho_1 \cdot \varepsilon' / |\Lambda| \cdot m\right)$, and this will yield the required rejection ratio.

Observe that by the smoothness of G_1 (resp. G'), the fraction of edges of G_1 (resp. G') that fail to decode x on π_1 (resp. π') is at least $\varepsilon_0 \overset{\text{def}}{=} \frac{1}{2} \cdot \varepsilon$ (resp. $\varepsilon_0' = \gamma' \cdot \varepsilon'$). Furthermore, the fraction of edges of G_1 that reject π_1 is at least $\rho_1 \cdot \varepsilon_0$. This implies, using the same argument as in the proof of Proposition 4.4, that the fraction of edges of G' that reject π' is at least $\Omega\left(\rho_1 \cdot \varepsilon_0 / |\Lambda| \cdot m\right)$.

We finish the proof by relating ε_0' with ε_0. To this end, observe that for every edge $e' = (u, v)$ of G' and its associated edge e_1 of G_1, the edge e' fails to decode x on π' (i.e., $\psi_{e'}(\pi'(u)) \notin \{x_{k_{e'}}, \bot\}$) only if e_1 fails to decode x on π_1 (i.e., $\psi_{e_1}(\pi_1(u)) \notin \{x_{k_{e_1}}, \bot\}$).

Furthermore, each edge e_1 of G_1 corresponds to either $\lfloor |\Lambda|/d \rfloor$ or $\lceil |\Lambda|/d \rceil$ edges in G'. It can be verified by a straightforward calculation that this implies that $\varepsilon_0' \leq 2 \cdot \varepsilon_0$. It now follows that the fraction of edges of G' that reject π' is at least

$$\Omega\left(\frac{\rho_1 \cdot \varepsilon_0}{|\Lambda| \cdot m}\right) \geq \Omega\left(\frac{\rho_1 \cdot \varepsilon_0'}{|\Lambda| \cdot m}\right)$$

$$\geq \Omega\left(\frac{\rho_1 \cdot \gamma'}{|\Lambda| \cdot m} \cdot \varepsilon'\right)$$

$$= \Omega\left(\frac{\rho}{|\Lambda|^2 \cdot m} \cdot \varepsilon'\right).$$

The required rejection ratio follows.   □

7.3. Proof of Proposition 7.4. In this section, we prove Proposition 7.4, restated below.

PROPOSITION (7.4, restated). *There exists a polynomial time procedure that acts as follows:*

○ **Input:**

- *A vertex-decoding graph G of size n for input circuit $\varphi : \Gamma^t \to \{0,1\}$ with ℓ vertices, alphabet Σ, rejection ratio ρ, decoding complexity s, degree bound d, and smoothness γ.*
- *A number $\ell' \in \mathbb{N}$ such that $\ell' \geq \ell$ (given in unary).*

○ **Output:** *Let $c \stackrel{\text{def}}{=} \lfloor \frac{\ell'}{\ell} \rfloor$ and let d_0 and h_0 be the constants from Fact 2.23. The procedure outputs a vertex-decoding graph G' of size at most $2 \cdot (c+1) \cdot d_0 \cdot n$ for input circuit φ that has exactly ℓ' vertices and also has alphabet Σ, output size $s + \text{poly} \log |\Sigma|$, rejection ratio $\Omega(\gamma^2 \cdot \rho/d^2)$, degree bound $2 \cdot d_0 \cdot d$, and smoothness $\frac{1}{2} \cdot \gamma$.*

Furthermore, if G is d-regular, then G' is $(2 \cdot d_0 \cdot d)$-regular and has rejection ratio $\Omega(\gamma^2 \cdot \rho)$.

Let $G = (V, E), \varphi, \ell$, and ℓ' be as in the proposition and let $z = \ell' \mod \ell$. We construct G' as follows. Choose an arbitrary set

$T \subseteq V$ of size z. The vertices of G' consist of a set C_v of vertices for each $v \in V$, where $|C_v| = c + 1$ if $v \in T$ and $|C_v| = c$ otherwise. Observe that G' indeed has ℓ' vertices. For each $v \in V$, let us denote $C_v = \{v_1, \ldots, v_{|C_v|}\}$. The edges of G' are defined as follows:

1. For each edge (u, v) of G and for each $l \in [c]$, the graph G' has d_0 edges (u_l, v_l) that are associated with the same index $k_{(u,v)}$ and circuit $\psi_{(u,v)}$ as the edge (u, v) of G. We call such edges "G-edges".

2. For each edge (u, v) for which $u \in T$, the graph G' contains the following "trivial" edges: Let $jk = k_{(u,v)}$ and $\psi = \psi_{(u,v)}$ be the index and circuit associated with (u, v). Recall that since G is vertex-decoding, there exists a function $f : \Sigma \to \Gamma$ such that for every $a, b \in \Sigma$ on which $\psi(a, b) \neq \bot$, it holds that $\psi(a, b) = f(a)$. Let $\psi' : \Sigma^2 \to \Gamma \cup \{\bot\}$ be the circuit that for every input $(a, b) \in \Sigma^2$ outputs $f(a)$. The graph G' contains d_0 edges (u_{c+1}, u_{c+1}) that are associated with the index k and with the circuit ψ'.

3. For each edge (u, v) of G, the graph G' contains the following edges, which correspond to "equality constraints": Let $k = k_{(u,v)}$ and $\psi = \psi_{(u,v)}$ be the index and circuit associated with (u, v), and let $f : \Sigma \to \Gamma$ as in Item 2. Let ψ' be the circuit that on input $(a, b) \in \Sigma^2$ outputs $\bot$ if $a \neq b$ and outputs $f(a)$ otherwise. We now identify the vertices of C_u with the vertices of the expander $G_{|C_u|}$ from Fact 2.23, and for every (undirected) edge of $G_{|C_u|}$, we put two directed edges between the corresponding vertices of C_u, where the directed edges are associated with the index k and with the circuit ψ'. We call such edges "consistency edges" of u.

Let n' be the size of G'. It is easy to see that G' has the correct size, alphabet, decoding complexity, and degree bound and also that G' satisfies the completeness requirement. It can also be verified that G' has smoothness $\left(1 - \frac{1}{c+1}\right) \cdot \gamma \geq \frac{1}{2} \cdot \gamma$ using the smoothness criterion (Proposition 6.22) and a straightforward calculation.

It remains to analyze the rejection ratio of G'. Let π' be an assignment to the vertices of G', and let π be the corresponding

plurality assignment to G. That is, π is the assignment that assigns each vertex v of G,the most popular value among the values that π' assigns to vertices in C_v. Suppose that G has decoding error ε on π and let $x \in \Gamma^t$ be an assignment that attains this decoding error. Let ε' be the decoding error of G' on π' with respect to x. We will show that G' rejects π' with probability at least $\frac{h_0 \cdot \gamma^2}{64} \cdot \rho \cdot \varepsilon'$ under the decoding distribution, and this clearly suffices since ε' is an upper bound on the decoding error of G'. To this end, we will analyze the decoding error and rejection probability of G' under the uniform distribution on the edges and then use the smoothness of G' to derive conclusions on the decoding distribution.

By the smoothness of G', the probability that a uniformly distributed edge of G' fails to decode x on π' is at least $\varepsilon'_1 \overset{\text{def}}{=} \frac{1}{2} \cdot \gamma \cdot \varepsilon'$. Furthermore, a uniformly distributed edge of G fails to decode x on π with probability at least $\varepsilon_1 \overset{\text{def}}{=} \gamma \cdot \varepsilon$ and rejects with probability at least $\rho \cdot \varepsilon_1 = \gamma \cdot \rho \cdot \varepsilon$. Let η be the fraction of vertices of G' on which π' is inconsistent with π. We begin the analysis by expressing ε'_1 in terms of ε_1 and η.

Let F be the set of edges of G that fail to decode x on π, let F' be the set of edges of G' that fail to decode x on π', and let S' be the set of vertices of G' on which π' is inconsistent with plurality assignment π, so $\eta \overset{\text{def}}{=} |S'| / \ell'$. An edge $e' = (u, v)$ of G' is in F' if and only if e' corresponds to some $e \in F$ or if u is in S' (note that since G' is vertex-decoding, we need not consider the case where v is in S'). Now, every edge in F has $d_0 \cdot c$ corresponding G-edges in G', and every vertex in S' has at most $2 \cdot d_0 \cdot d$ outgoing edges. Thus, it holds that

$$|F'| \le d_0 \cdot c \cdot |F| + 2 \cdot d_0 \cdot d \cdot |S'| \,.$$

Observe that since every vertex of G has at least one outgoing edge (since G is vertex-decoding), it holds that every vertex in G' has at least $2 \cdot d_0$ outgoing edges, and therefore $n' \ge 2 \cdot d_0 \cdot \ell'$. It follows that

$$\varepsilon'_1 = \frac{|F'|}{n'}$$
$$\le \frac{d_0 \cdot c \cdot |F| + 2 \cdot d_0 \cdot d \cdot |S'|}{n'}$$

$$\leq \frac{d_0 \cdot c \cdot |F|}{2 \cdot d_0 \cdot c \cdot n} + \frac{2 \cdot d_0 \cdot d \cdot |S'|}{2 \cdot d_0 \cdot \ell'}$$

(7.6)
$$\leq \varepsilon_1 + d \cdot \eta.$$

Observe that the last inequality implies that if η is small compared to ε_1', then ε_1 must be large and vice versa. We turn to consider each of the cases separately.

The case where η is small.   First, consider the case where $\eta \leq \rho \cdot \varepsilon_1'/16 \cdot d$. In this case, we argue that π' is roughly consistent with π, and therefore, the action of G' on π' is similar to the action of G on π. In particular, we argue that the fraction of edges of G' that reject π' must be related to the fraction of edges of G that reject π, which is at least $\rho \cdot \varepsilon_1$. However, since by Inequality **??**, it holds that ε_1 is large compared to ε_1', it will follow that the fraction of edges of G' that reject π' is roughly $\rho \cdot \varepsilon_1'$, as required.

More formally, it holds that the fraction of edges touching S' (both incoming and outgoing) is at most

$$\frac{2 \cdot d_0 \cdot d \cdot |S'|}{n'} = \frac{2 \cdot d_0 \cdot d \cdot \eta \cdot \ell'}{n'}$$

$$\text{(Since } n' \geq 2 \cdot d_0 \cdot \ell') \leq \frac{2 \cdot d_0 \cdot d \cdot \eta}{2 \cdot d_0}$$

$$\text{(By assumption on } \eta) \leq \frac{d_0 \cdot d \cdot \rho \cdot \varepsilon_1'}{d_0 \cdot 16d}$$

$$= \frac{\rho \cdot \varepsilon_1'}{16}.$$

On the other hand, it holds that the size of F (the set of edges of G that reject π) is at least $\rho \cdot \varepsilon_1 \cdot n$. Each such edge has at least $d_0 \cdot c$ corresponding G-edges in G', and since $n' \leq 2 \cdot d_0 \cdot (c+1) \cdot n$, it follows that the fraction of edges of G' that correspond to edges in F is at least $\left(\frac{d_0 \cdot c \cdot |F|}{2 \cdot d_0 \cdot (c+1) \cdot n} \right) \geq \rho \cdot \varepsilon_1/4$. Furthermore, it holds that

$$\varepsilon_1 \geq \varepsilon_1' - d \cdot \eta \geq \varepsilon_1' - \rho \cdot \varepsilon_1'/16 \geq \varepsilon_1'/2.$$

So in fact, the fraction of edges in G' that correspond to edges in F is at least $\rho \cdot \varepsilon_1/4 \geq \rho \cdot \varepsilon_1'/8$. This implies that the fraction of edges

of G' that both correspond to edges in F and whose endpoints are consistent with π is at least $\rho \cdot \varepsilon_1'/8 - \rho \cdot \varepsilon_1'/16 \geq \rho \cdot \varepsilon_1'/16$. Since all of these edges reject π', it follows that the fraction of edges of G' that reject π' is at least $\rho \cdot \varepsilon_1'/16 \geq \rho \cdot \frac{1}{2} \cdot \gamma \cdot \varepsilon'/16 \geq \gamma \cdot \rho \cdot \varepsilon'/32$. This implies that the rejection probability of π' under the decoding distribution of G' is at least $\Omega\left(\gamma^2 \cdot \rho \cdot \varepsilon'\right)$. as required.

The case where η is large.   We turn to consider the case where $\eta \geq \rho \cdot \varepsilon_1'/16 \cdot d$. In this case, the assignment π' is quite inconsistent with π, and we argue that a significant fraction of the consistency edges reject π'. More formally, using similar considerations as in the proof of Proposition 6.24, every set C_v contributes at least $h_0 \cdot d_0 \cdot |S' \cap C_v|$ rejecting consistency edges. Thus, there are at least $h_0 \cdot d_0 \cdot |S'|$ rejecting edges. This implies that the fraction of rejecting edges is at least

$$
\begin{aligned}
\frac{h_0 \cdot d_0 \cdot |S'|}{n'} &\geq \frac{h_0 \cdot d_0 \cdot |S'|}{2 \cdot d_0 \cdot d \cdot \ell'} \\
&= \frac{h_0}{2 \cdot d} \cdot \eta \\
&\geq \frac{h_0}{32 \cdot d^2} \cdot \rho \cdot \varepsilon_1' \\
&\geq \frac{h_0}{32 \cdot d^2} \cdot \rho \cdot \frac{1}{2} \cdot \gamma \cdot \varepsilon' \\
&\geq \frac{h_0 \cdot \gamma}{64 \cdot d^2} \cdot \rho \cdot \varepsilon',
\end{aligned}
$$

which implies that the rejection probability under the decoding distribution is at least $\Omega\left(\gamma^2 \cdot \rho \cdot \varepsilon'/d^2\right)$, as required.

The "furthermore" part.   For the "furthermore" part of the lemma, first observe that it is easy to see from the definition of G' that if G is d-regular, then G' is $(2 \cdot d_0 \cdot d)$-regular. For the rejection ratio part, note that in the foregoing analysis, we lose a $1/d$ factor in two places:

1. We lose a factor of $1/d$ in the proof of Inequality (7.6), where our upper bound on the number of edges that go out of S is $2 \cdot d_0 \cdot d \cdot |S|$, while our lower bound on n' is only $2 \cdot d_0 \cdot \ell'$.

However, if G is d-regular, then G' is $(2 \cdot d_0 \cdot d)$-regular, and thus, the lower bound on n' can be improved to $2 \cdot d_0 \cdot d \cdot \ell'$. This implies that Inequality (7.6) becomes $\varepsilon'_1 \leq \varepsilon_1 + \eta$.

As a result, the case of "small η" can be extended to all the cases where $\eta \leq \rho \cdot \varepsilon'_1/16$, and in the case of "large η", we can assume that $\eta \geq \rho \cdot \varepsilon'_1/16$. This saves a factor of $1/d$ in the case of "large η".

2. We lose a factor of $1/d$ in the case of "large η", since the lower bound on the number of rejecting consistency edges for a set C_v is only $h_0 \cdot d_0 \cdot |S \cap C_v|$, while the upper bound on the number of consistency edges in the graph is $d_0 \cdot d \cdot n$. However, if G is d-regular, then the foregoing lower bound can be improved to $h_0 \cdot d_0 \cdot d \cdot |S \cap C_v|$, regaining the factor of $1/d$. □

8. Derandomized Parallel Repetition of Decoding Graphs with Linear Structure

In this section, we prove Lemma 6.26 (derandomized parallel repetition for dPCPs), restated below.

LEMMA (6.26, restated). *There exist a universal constant h and a polynomial time procedure that satisfy the following requirements:*

○ **Input:**

 – *A finite field $\mathbb{F}$ of size q.*

 – *A decoding graph $G = (\mathbb{F}^m, E)$ of size n for input circuit $\varphi : \Gamma^t \to \{0, 1\}$ with linear structure, alphabet Σ, rejection ratio ρ, decision complexity s, and smoothness γ.*

 – *The rejection ratio ρ of G.*

 – *A parameter $d_0 \in \mathbb{N}$ such that $d_0 < m/h^2$ and $\rho \geq h \cdot d_0 \cdot q^{-d_0/h}/\gamma$.*

○ **Output:** *A decoding graph G' for φ such that the following holds:*

 – *G' has size $n^{O(d_0)}$.*

 – *G' has alphabet $\Sigma^{q^{O(d_0)}}$.*

- G' is list-decoding with soundness error $\varepsilon \overset{\text{def}}{=} h \cdot d_0 \cdot q^{-d_0/h}/\gamma$ and list size $L \overset{\text{def}}{=} q^{O(d_0)}$.

- G' has the projection property.

- G' has decoding complexity $q^{O(d_0)} \cdot (s + \text{poly} \log |\Sigma|)$.

The proof follows the proof of the corresponding lemma for constraint graphs (Lemma 3.4), with the following modification: Recall that the proof of Lemma 3.4 described the graph G' by describing a *verification* procedure (the E-test, Figure 5.1). More-over, recall that the E-test works by choosing a random subspace F of edges and verifying that the edges in F are satisfied by the assignment $\Pi(F)$.

In order to describe the graph G' of Lemma 6.26, we describe a *decoding* procedure (the E-decoder, see Figure 8.1 below). The E-decoder is constructed by changing the E-test as follows. When-ever the E-decoder is required to decode an index $k \in [t]$, the E-decoder chooses a random edge e that is associated with k and then chooses the subspace F to be a random subspace that con-tains e. The E-decoder then checks, as before, that the edges in F are satisfied by the assignment $\Pi(F)$. If one of the edges in F is unsatisfied, then the E-decoder rejects. If all the edges in F are satisfied, then the E-decoder decodes the index k by invoking the circuit ψ_e associated with e on input $\Pi(F)_{|e}$.

The intuition that underlies the construction of the E-decoder is as follows. Just as in the proof of Lemma 3.4, we argue that the E-decoder contains an implicit S-test, and therefore, the assign-ment Π needs to be roughly consistent with some assignment π to G in order to be accepted. We now consider two cases:

1. If G has high decoding error on π, then by the soundness of G, it holds that many of the edges of G reject π. By the sampling property of F, there are many edges in F that reject π, and therefore, the E-decoder must reject with high probability.

2. If G has low decoding error on π, then due to the sampling property of F, only few of the edges in F err. In particular, since e is distributed like a random edge of F, it only errs with

---

1. Suppose that we are required to decode an index $k \in [t]$. Let $e = (u, v)$ be a uniformly distributed edge of G that is associated with k, and let ψ_e be its associated circuit.

2. Let F_L and F_R to be random d_1-subspaces of E, and let

$$B_L \stackrel{\text{def}}{=} \text{left}(F_L), \quad B_R \stackrel{\text{def}}{=} \text{right}(F_R), \quad F \stackrel{\text{def}}{=} F_L + F_R.$$

   F_L and F_R are chosen to be uniformly and independently distributed d_1-subspaces of E conditioned on $e \in F$, $\dim(F) = 2d_1$, $\dim(B_L) = d_1$, $\dim(B_R) = d_1$, and $B_L \cap B_R = \{0\}$.

3. Let A_L and A_R be uniformly distributed d_0-subspaces of B_L and B_R respectively, and let

$$A \stackrel{\text{def}}{=} A_L + A_R.$$

4. If either $\Pi(F)_{|(A_L, A_R)} \neq \Pi(A)_{|(A_L, A_R)}$ or the assignment $\Pi(F)$ is rejected by of the edges in F, output $\perp$.

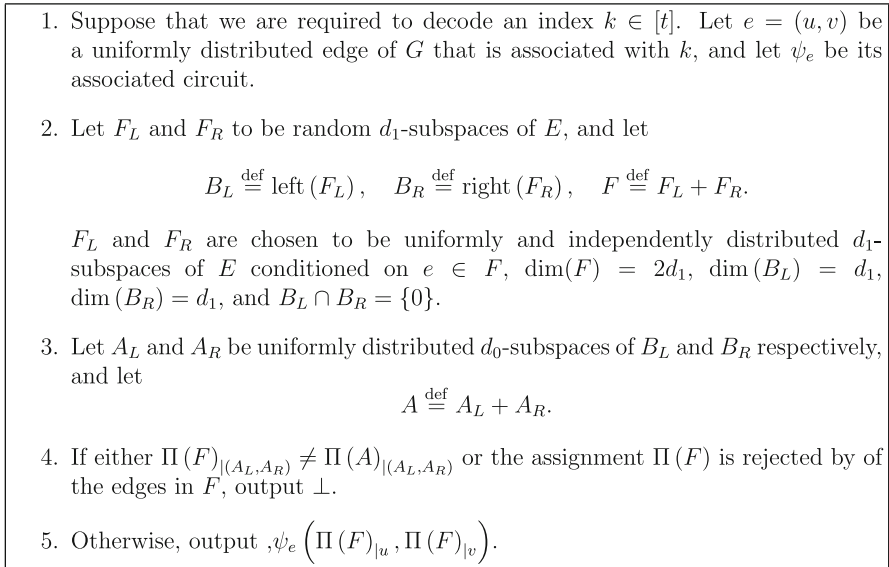5. Otherwise, output $,\psi_e\left(\Pi(F)_{|u}, \Pi(F)_{|v}\right)$.

Figure 8.1: The E-decoder.

low probability. Thus, in this case, the E-decoder decodes correctly with high probability.

Thus, in both cases, the soundness error of the E-decoder is small.

8.1. The Construction of G' and its Parameters. The decoding graph G' is constructed as follows. Let $G = (\mathbb{F}^m, E)$ and d_0 be as in Lemma 6.26 (derandomized parallel repetition for dPCPs), and let $d_1 = h \cdot d_0$ where h is the universal constant from Lemma 6.26 to be chosen later. As in the proof of the corresponding lemma for constraint graphs (Lemma 3.4), the graph G' is bipartite, the right vertices of G' are the $2d_0$-subspaces of $\mathbb{F}^m$ (the vertex-space of G), and the left vertices of G' are the $2d_1$-subspaces of the edge space E of G. An assignment Π to G' should label each $2d_0$-subspace A of $\mathbb{F}^m$ with a function from A to Σ and each $2d_1$-subspace F of E with a function that maps the endpoints of the edges in F to Σ. The edges of G' are constructed such that they simulate the action of the "E-decoder" described in Figure 8.1.

The completeness, size, and alphabet size of G' can be verified in the same way as it was done in the proof of Lemma 3.4, and so is the fact that G' has the projection property. It remains to analyze the soundness of G', which is done in the following section.

We turn to prove that G' is list-decoding with $\varepsilon = h \cdot d_0 \cdot q^{-d_0/h}/\gamma$ and list size $L = q^{O(d_0)}$. Let Π be an assignment to G'. That is, we prove that there exists a (possible empty) list of satisfying assignments $x^1, \ldots, x^L \in \Gamma^t$ to the input circuit φ such that when given as input a uniformly distributed index $k \in [t]$, the probability that the output of the E-decoder is not in $\{x_k^1, \ldots, x_k^L, \bot\}$ is at most ε.

Consider the distribution on the edges of G' that results from letting the edge e of the E-decoder be chosen according to *the uniform distribution on the edges of G* instead of the decoding distribution of G. We will refer to the above distribution as the *G-uniform distribution of G'*. It is straightforward to show that the G-uniform distribution and decoding distribution of G' are γ-similar, by applying Claim 2.21 with X_1 and X_2 being the choices of e according to the G-uniform distribution and the decoding distribution, and Y_1 and Y_2 being the G-uniform distribution and decoding distribution of G', respectively. In the following proof, all the probability expressions are *not over the decoding distribution* of G', but rather over the *G-uniform distribution of G'*. We will later use the similarity between the distributions to argue that G' has small soundness error with respect to its decoding distribution.

NOTATION 8.1. *We denote by $\mathcal{D}$ the random variable that equals to the output of the E-decoder. As in the proof of Lemma 3.4 (derandomized parallel repetition for constraint graphs), we denote by $\mathcal{T}$ the event in which the E-decoder accepts Π, so $\mathcal{T}$ is the event $\mathcal{D} \neq \bot$. Moreover, as in the proof of Lemma 3.4, for an assignment $\pi : \mathbb{F}^m \to \Sigma$, we denote by $\Pi(F) \overset{\alpha}{\approx} \pi$ the claim that for at least $1 - \alpha$ fraction of the edges e of F it holds that $\Pi(F)$ is consistent with π on both the endpoints of e, and otherwise, we denote $\Pi(F) \overset{\alpha}{\not\approx} \pi$.*

Our proof proceeds in two steps. We first show that there exists a (possible empty) assignments $\pi^1, \ldots, \pi^L : \mathbb{F}^m \to \Sigma$ such that whenever the E-decoder accepts Π, it almost always does so while being roughly consistent with one of the assignments $\pi^1, \ldots, \pi^L$. We can then choose the assignments $x^1, \ldots, x^L$ to be the assignments that minimize the decoding error of $\pi^1, \ldots, \pi^L$, respectively. Next, we show that whenever Π is roughly consistent with π^i, the E-decoder either rejects Π with high probability (if π^i has high decoding error) or decodes x^i successfully with high probability (if π^i has low decoding error). Thus, the overall probability that the E-decoder fails is small.

The above strategy is made formal in the following three propositions. Let h' and c be the universal constants defined in The-orem 8.7 below, and let $\alpha \stackrel{\text{def}}{=} h' \cdot d_0 \cdot q^{-d_0/h'}$. Let $\varepsilon_0 \stackrel{\text{def}}{=} \varepsilon \cdot \gamma/3 = h \cdot d_0 \cdot q^{-d_0/h}/3$ and let $L = O\left(1/\varepsilon_0^c\right)$.

PROPOSITION 8.2. *There exists a (possibly empty) list of assignments $\pi^1, \ldots, \pi^L : \mathbb{F}^m \to \Sigma$ such that*

$$\Pr\left[\mathcal{T} \text{ and } \not\exists i \in [L] \text{ s.t. } \Pi(F) \stackrel{4 \cdot \alpha}{\approx} \pi^i\right] < 2 \cdot \varepsilon_0.$$

PROPOSITION 8.3. *For every assignment $\pi : \mathbb{F}^m \to \Sigma$ on which G has decoding error at least $\varepsilon_0/2L$, it holds that $\Pr\left[\mathcal{T} \text{ and } \Pi(F) \stackrel{4 \cdot \alpha}{\approx} \pi\right] < \varepsilon_0/L$.*

PROPOSITION 8.4. *For every assignment $\pi : \mathbb{F}^m \to \Sigma$ on which G has decoding error less than $\varepsilon_0/2L$ with respect to a satisfying assignment x to the input circuit φ, it holds that*

$$\Pr\left[\mathcal{D} \neq x_k \text{ and } \Pi(F) \stackrel{4 \cdot \alpha}{\approx} \pi\right] < \varepsilon_0/L,$$

where k is the index on which the E-decoder is invoked.

Proposition 8.2 and Proposition 8.4 are proved in Section 8.2.1 and Section 8.2.2, respectively. Proposition 8.3 can be proved in the same way as Proposition 5.8, by noting that due to the soundness of G, at least $\rho \cdot \varepsilon_0/2L$ of the edges of G reject π.

We now prove that G' is (L, ε)-list decoding using Proposition 8.2, Proposition 8.3, and Proposition 8.4. Let $\pi^1, \ldots, \pi^L$ be the assignments from Proposition 8.2. For each $i \in [L]$, let x^i be the assignment to φ that attains the decoding error of π^i. The decoding error of G' on Π under the G-uniform distribution of G' is as follows.

$$\Pr\left[\mathcal{D} \notin \left\{x_k^1, \ldots, x_k^L, \bot\right\}\right]$$

$$\leq \sum_{i=1}^{L} \Pr\left[\mathcal{D} \notin \left\{x_k^1, \ldots, x_k^L, \bot\right\} \text{ and } \Pi(F) \overset{4 \cdot \alpha}{\approx} \pi^i\right]$$

$$+ \Pr\left[\mathcal{D} \notin \left\{x_k^1, \ldots, x_k^L, \bot\right\} \text{ and } \not\exists i \in [L] \text{ s.t. } \Pi(F) \overset{4 \cdot \alpha}{\approx} \pi^i\right]$$

$$\leq \sum_{i=1}^{L} \Pr\left[\mathcal{D} \notin \left\{x_k^i, \bot\right\} \text{ and } \Pi(F) \overset{4 \cdot \alpha}{\approx} \pi^i\right]$$

$$(8.5) \quad + \Pr\left[\mathcal{T} \text{ and } \not\exists i \in [L] \text{ s.t. } \Pi(F) \overset{4 \cdot \alpha}{\approx} \pi^i\right]$$

$$\leq \sum_{i=1}^{L} \varepsilon_0 / L + 2 \cdot \varepsilon_0$$

$$= 3 \cdot \varepsilon_0,$$

$$(8.6)$$

where Inequality (8.6) follows from Proposition 8.2 and Proposition 8.4. Finally, since the G-uniform distribution of G' and the decoding distribution of G' are γ-similar, it follows that the decoding error of G' on Π under the decoding distribution of G' is at most $3 \cdot \varepsilon_0 / \gamma = \varepsilon$, as required. $\qquad \square$

8.2.1. Proof of Proposition 8.2.
Recall that in order to analyze the soundness of the E-test in Proposition 5.7, we argued that the E-test contains an "implicit S-test" and then relied on a theorem regarding the soundness of the S-test (Theorem 5.4). The aforementioned theorem said that if the S-test accepts an assignment Π with some probability, then there exists an assignment π such that with some (smaller) probability, the S-test accepts Π while being consistent with the S-direct product of π. This can be thought as a "unique decoding" theorem that decodes π from Π.

In order to prove Proposition 8.2 for the E-decoder, we use a
similar argument, but this time we use a "list-decoding" theorem
for the S-test. The following theorem says that there exists a short
list of assignments $\pi_1, \ldots, \pi_L$, such that it is *almost always* the case
that if the S-test accepts Π, it does so while being consistent with
the S-direct product of one of the assignments $\pi_1, \ldots, \pi_L$.

THEOREM 8.7 (List-decoding soundness of the S-test).   *There
exist universal constants* $h', c \in \mathbb{N}$ *such that for every* $d_0 \in
\mathbb{N}, d_1 \geq h' \cdot d_0$, *and* $m \geq h' \cdot d_1$, *the following holds: Let*
$\varepsilon \geq h' \cdot d_0 \cdot q^{-d_0/h'}, \alpha \overset{\text{def}}{=} h' \cdot d_0 \cdot q^{-d_0/h'}$. *Let* Π *be a (possibly
randomized) assignment to* $2d_0$-*subspaces of* $\mathbb{F}^m$ *and to pairs of*
d_1-*subspaces of* $\mathbb{F}^m$. *Then, there exists a (possibly empty) list of*
$L = O\left(1/\varepsilon^c\right)$ *assignments* $\pi^1, \ldots, \pi^L : \mathbb{F}^m \to \Sigma$ *such that*

$$\Pr\left[\Pi\left(B_1, B_2\right)_{|(A_1, A_2)} = \Pi\left(A\right)_{|(A_1, A_2)}\right.$$
$$\left. \text{and} \quad \nexists i \in [L] \text{ s.t. } \Pi\left(B_1, B_2\right) \overset{\alpha}{\approx} \pi^i_{|(B_1, B_2)}\right] < \varepsilon.$$

Theorem 8.7 is proved in Section 9.

We turn to prove Proposition 8.2 based on Theorem 8.7. As
in the proof of Proposition 5.7, we begin by extending Π to pairs
of independent d_1-subspaces of $\mathbb{F}^m$ in a randomized manner as fol-
lows: Given a pair of independent d_1-subspaces B_1 and B_2, we
choose F_1 and F_2 to be uniformly distributed and independent
d_1-subspaces of E such that $\text{left}\left(F_1\right) = B_1$ and $\text{right}\left(F_2\right) = B_2$,
and set $\Pi\left(B_1, B_2\right) = \Pi\left(F_1 + F_2\right)_{|(B_1, B_2)}$.

Again as in the proof of Proposition 5.7, we observe that the
probability that the E-decoder accepts equals to the probability
that the S-test accepts the extended Π. The reason is that the
subspaces B_L, B_R, A_L and A_R of the E-decoder are distributed like
the subspaces B_1, B_2, A_1 and A_2 of the S-test. By choosing h to be
at least the constant h', we can invoke Theorem 8.7 (list-decoding
soundness of the S-test) and conclude that there exists a list of
$L = O\left(1/\varepsilon^c\right)$ assignments $\pi^1, \ldots, \pi^L : \mathbb{F}^m \to \Sigma$ such that for sub-

spaces B_1, B_2, A_1, A_2 as in the S-test it holds that

$$\Pr\left[\Pi(B_1, B_2)_{|(A_1, A_2)} = \Pi(A)_{|(A_1, A_2)} \right.$$
$$\left. \text{and} \quad \nexists i \in [L] \text{ s.t. } \Pi(B_1, B_2) \overset{\alpha}{\approx} \pi^i_{|(B_1, B_2)} \right] < \varepsilon_0.$$

The latter inequality is equivalent to the following inequality:

$$\Pr\left[\Pi(F)_{|(B_L, B_R)} = \Pi(A)_{|(A_1, A_2)} \right.$$
$$\left. \text{and} \quad \nexists i \in [L] \text{ s.t. } \Pi(F)_{|(B_L, B_R)} \overset{\alpha}{\approx} \pi^i_{|(B_L, B_R)} \right] < \varepsilon_0,$$

which in turn implies the inequality

$$\Pr\left[\mathcal{T} \text{ and } \nexists i \in [L] \text{ s.t. } \Pi(F)_{|(B_L, B_R)} \overset{\alpha}{\approx} \pi^i_{|(B_L, B_R)} \right] < \varepsilon_0.$$
(8.8)

In the rest of this section, we show that this implies that

$$(8.9) \qquad \Pr\left[\mathcal{T} \text{ and } \nexists i \in [L] \text{ s.t. } \Pi(F) \overset{4 \cdot \alpha}{\approx} \pi^i \right] < 2 \cdot \varepsilon_0.$$

To this end, we use Claim 5.12, which was proved in Section 5.3.1 and is restated below.

CLAIM (5.12, restated). *For every fixed $2d_0$-subspace F_0 of E such that $\Pi(F_0) \overset{4\alpha}{\not\approx} \pi$, it holds that*

$$\Pr\left[\Pi(F)_{|(B_L, B_R)} \overset{\alpha}{\approx} \pi_{|(B_L, B_R)} \,\middle|\, F = F_0 \right] \le 1/\left(q^{d_1 - 2} \cdot \alpha^2 \right).$$

Claim 5.12 implies immediately the following corollary.

COROLLARY 8.10. *For every $i \in [L]$, it holds that*

$$\Pr\left[\Pi(F)_{|\left(B_L, B_R\right)} \overset{\alpha}{\approx} \pi_{i|\left(B_L, B_R\right)} \,\middle|\, \nexists j \in [L] \text{ s.t.} \right.$$
$$\left. \Pi(F) \overset{4 \cdot \alpha}{\approx} \pi^j \right] < 1/\left(q^{d_1 - 2} \cdot \alpha^2 \right).$$

In order to prove Inequality (8.9), we first show that

$$\Pr\left[\mathcal{T} \text{ and } \not\exists i \in [L] \text{ s.t.} \Pi(F)_{|(B_L,B_R)} \stackrel{\alpha}{\approx} \pi^i_{|(B_L,B_R)}\right|$$

$$(8.11) \qquad \not\exists i \in [L] \text{ s.t. } \Pi(F) \stackrel{4\cdot\alpha}{\approx} \pi^i\right] \geq \frac{1}{2}.$$

To show it, we prove an upper bound on the complement event, that is, we prove that

$$\Pr\left[\mathcal{T} \text{ and } \exists i \in [L] \text{ s.t. } \Pi(F)_{|(B_L,B_R)} \stackrel{\alpha}{\approx} \pi^i_{|(B_L,B_R)}\right|$$

$$\not\exists i \in [L] \text{ s.t. } \Pi(F) \stackrel{4\cdot\alpha}{\approx} \pi^i\right] \leq \frac{1}{2}.$$

To see the latter inequality, observe that the right end side is upper bounded by

$$\sum_{i\in[L]} \Pr\left[\Pi(F)_{|(B_L,B_R)} \stackrel{\alpha}{\approx} \pi^i_{|(B_L,B_R)}\right| \not\exists j \in [L] \text{ s.t. } \Pi(F) \stackrel{4\cdot\alpha}{\approx} \pi^j\right]$$

$$\leq \sum_{i\in[L]} 1/\left(q^{d_1-2} \cdot \alpha^2\right)$$

$$= L \cdot / \left(q^{d_1-2} \cdot \alpha^2\right)$$

$$= O\left(1/\varepsilon_0^c \cdot \left(q^{d_1-2} \cdot \alpha^2\right)\right)$$

$$\leq \frac{1}{2}.$$

where the first inequality follows from Corollary 8.10, and the second inequality follows for sufficiently large choice of h. Now, it holds that

$$\Pr\left[\mathcal{T} \text{ and } \not\exists i \in [L] \text{ s.t. } \Pi(F)_{|(B_L,B_R)} \stackrel{\alpha}{\approx} \pi^i_{|(B_L,B_R)}\right.$$

$$\text{and } \not\exists i \in [L] \text{ s.t. } \Pi(F) \stackrel{4\cdot\alpha}{\approx} \pi^i\right]$$

is upper bounded by

$$\Pr\left[\mathcal{T} \text{ and } \not\exists i \in [L] \text{ s.t. } \Pi(F)_{|(B_L,B_R)} \stackrel{\alpha}{\approx} \pi^i_{|(B_L,B_R)}\right] < \varepsilon_0.$$

On the other hand, by writing the probability in (8.12) in conditional form and applying Inequality (8.11), we obtain that the probability in (8.12) is at least

$$\frac{1}{2} \cdot \Pr\left[\mathcal{T} \text{ and } \not\exists i \in [L] \text{ s.t. } \Pi(F) \overset{4 \cdot \alpha}{\approx} \pi^i\right].$$

By combining the two last bounds, we obtain that

$$\Pr\left[\mathcal{T} \text{ and } \not\exists i \in [L] \text{ s.t. } \Pi(F) \overset{4 \cdot \alpha}{\approx} \pi^i\right] < 2 \cdot \varepsilon_0,$$

as required. $\qquad\square$

8.2.2. Proof of Proposition 8.4. Fix an assignment $\pi : \mathbb{F}^m \to \Sigma$ on which G has decoding error less than $\varepsilon_0/2L$ with respect to a satisfying assignment x of the input circuit φ. We prove that $\Pr\left[D \neq x_k \text{ and } \Pi(F) \overset{4 \cdot \alpha}{\approx} \pi\right] < \varepsilon_0/L$ Let us denote by $\mathcal{E}_1$ the event in which $\Pi(F) \overset{4 \cdot \alpha}{\approx} \pi$ and by $\mathcal{E}_2$ the event in which F contains less than $\varepsilon_0/3L$ fraction of edges on which G fails to decode x on π. We will prove that

$$\Pr\left[\mathcal{D} \neq x_k \text{ and } \mathcal{E}_1\right] = \Pr\left[\mathcal{D} \neq x_k \text{ and } \Pi(F) \overset{4 \cdot \alpha}{\approx} \pi\right] < \varepsilon_0/L.$$

It holds that

$$\begin{aligned}\Pr\left[\mathcal{D} \neq x_k \text{ and } \mathcal{E}_1\right] = {} &\Pr\left[\mathcal{D} \neq x_k \text{ and } \mathcal{E}_1 \text{ and } \mathcal{E}_2\right] \\ &+ \Pr\left[\psi(a,b) \neq x_k \text{ and } \mathcal{E}_1 \text{ and } \neg\mathcal{E}_2\right].\end{aligned}$$

We upper bound both terms on the right-hand side. The second term is clearly upper bounded by $\Pr[\neg\mathcal{E}_2]$. The latter probability can be shown to be at most $O\left(L^2/q^{2 \cdot d_1 - 2} \cdot \varepsilon_0^2 + \cdot d_1/q^{m-2 \cdot d_1}\right)$, using the fact that F samples well the edges of G and more specifically using an argument similar to the one used in the proof of Proposition 5.8. For sufficiently large choice of h, the latter expression is upper bounded by $\varepsilon/3L$.

We turn to upper bound the probability $\Pr\left[\mathcal{D} \neq x_j \text{ and } \mathcal{E}_1 \text{ and } \mathcal{E}_2\right]$. This probability is upper bounded by the probability $\Pr\left[\mathcal{D} \neq x_j | \mathcal{E}_1 \text{ and } \mathcal{E}_2\right]$. Now, let F_0 be any $2d_1$-subspace of E

such that $\Pi(F_0) \overset{4\cdot\alpha}{\approx} \pi_i$ and such that the fraction of edges of F_0 that fail to decode x on π is at most $2\varepsilon_0/3L$. Let us consider the probability $\Pr[\mathcal{D} \neq x_j | F = F_0]$. Observe that conditioned on the choice $F = F_0$, the edge e chosen by the E-test is uniformly distributed among the edges of F. Observe that e fails to decode x only if one of the endpoints of e is inconsistent with π or if e is one of the edges in F that fail to decode x on π. The probability of the first case is at most $4\cdot\alpha \leq \varepsilon_0/3L$ (where the latter inequality holds for sufficiently large choice of h), and the probability of the second case is at most $\varepsilon_0/3L$. It therefore holds that

$$\Pr[\mathcal{D} \neq x_k \text{ and } \mathcal{E}_1 \text{ and } \mathcal{E}_2] \leq \Pr[\mathcal{D} \neq x_j | F = F_0]$$
$$\leq \varepsilon_0/3L + \varepsilon_0/3L \leq 2\varepsilon_0/3L.$$

All in all, it holds that $\Pr[\mathcal{D} \neq x_k \text{ and } \mathcal{E}_1]$ is at most $2\varepsilon_0/3L + 3 \cdot \varepsilon_0/3L = \varepsilon_0/L$, as required.

9. The Analysis of the Specialized Direct Product Test

In this section, we provide the analysis of the S-test and prove Theorem 5.4 and Theorem 8.7, which are the theorems on the soundness of the S-test that are used in Section 5.3.1 and Section 8.2.1, respectively. The proof proceeds in two steps. First, in Section 9.1, we define and analyze an intermediate direct product test, which we call the P^2-test. Then, in Section Section 9.2, we reduce the analysis of the S-test to that of the P^2-test. We provide the analysis of the P-test (Theorem 2.2) in Section 9.3.

For the rest of this section, we let $\mathbb{F}$ be a finite field of size q and let $d_0, d_1 \in \mathbb{N}$.

9.1. The P^2-test. In this section, we define and analyze the P^2-test. Informally, the P^2-test consists of two P-tests that are performed simultaneously. Details follow.

Given two strings $\pi_1, \pi_2 : \mathbb{F}^m \to \Sigma$, we define their P^2-*direct product* Π (with respect to $d_0, d_1 \in \mathbb{N}$) as follows: Π assigns each pair of d_0-subspaces (A_1, A_2) to the pair of functions $(\pi_{1|A_1}, \pi_{2|A_2})$ and assigns each pair of d_1-subspaces (B_1, B_2) to the pair of functions $(\pi_{1|B_1}, \pi_{2|B_2})$. We consider the task of testing whether a

---

1. Choose two uniformly distributed d_1-subspaces B_1, B_2 of $\mathbb{F}^m$.

2. Choose two uniformly distributed d_0-subspaces $A_1 \subseteq B_1, A_2 \subseteq B_2$.

3. Accept if and only if $\Pi\left(B_1, B_2\right)_{|(A_1, A_2)} = \Pi\left(A_1, A_2\right)$.

Figure 9.1: The P^2-test.

given assignment Π is the P^2-direct product of some pair of strings $\pi_1, \pi_2 : \mathbb{F}^m \to \Sigma$. That is, we are given an assignment Π , and in order to check whether Π is a P^2-direct product, we invoke the P^2-test, described in Figure 9.1.

It is easy to see that if Π is a P^2-direct product, then the P^2-test always accepts. Again, it can be shown that if Π is "far" from being a P^2-direct product, then the P^2-test rejects with high probability and that this holds even if Π is a randomized assignment. Formally, we have the following result.

THEOREM 9.1 (Soundness of the P^2-test). *There exist universal constants* $h, c \in \mathbb{N}$ *such that the following holds: Let* $\varepsilon \geq h \cdot d_0 \cdot q^{-d_0/h}, \alpha \stackrel{\text{def}}{=} h \cdot d_0 \cdot q^{-d_0/h}$. *Assume that* $d_1 \geq h \cdot d_0, m \geq h \cdot d_1$. *Suppose that an assignment* Π *passes the* P^2-test *with probability at least* ε. *Then, there exist two assignments* π_1 *and* π_2 *to* $\mathbb{F}^m$ *such that for* $B_1, B_2, A_1,$ *and* A_2, *distributed as in the* P^2-test, *it holds that*

$$\Pr\left[\Pi\left(B_1, B_2\right)_{|(A_1, A_2)} = \Pi\left(A_1, A_2\right) \text{ and } \Pi\left(A_1, A_2\right) \stackrel{\alpha}{\approx} \left(\pi_{1|A_1}, \pi_{2|A_2}\right)\right.$$
$$\left. \text{ and } \Pi\left(B_1, B_2\right) \stackrel{\alpha}{\approx} \left(\pi_{1|B_1}, \pi_{2|B_2}\right)\right] \geq \Omega\left(\varepsilon^c\right).$$

In the rest of this section, we prove Theorem 9.1. We denote by $\mathcal{P}$ the event in which the P^2-test accepts, that is, that $\Pi\left(B_1, B_2\right)_{|(A_1, A_2)} = \Pi\left(A_1, A_2\right)$. The core of the proof is the following lemma:

LEMMA 9.2. *There exist universal constants* $h', c' \in \mathbb{N}$ *such that the following holds: Let* $\varepsilon \geq h' \cdot d_0 \cdot q^{-d_0/h'}, \alpha' \stackrel{\text{def}}{=} h' \cdot d_0 \cdot q^{-d_0/h'}$. *Assume that* $d_1 \geq h' \cdot d_0, m \geq h' \cdot d_1$. *If* Π *passes the* P^2-test *with probability at least* ε, *then there exists an assignment* $\pi_2 : \mathbb{F}^m \to \Sigma$

such that

$$\Pr\left[\mathcal{P} \text{ and } \Pi\,(A_1, A_2)_{|A_2} \overset{\alpha'}{\approx} \pi_{2|A_2} \text{ and } (B_1, B_2)_{|B_2} \overset{\alpha'}{\approx} \pi_{2|B_2}\right]$$
$$\geq \Omega(\varepsilon^{c'}),$$

and symmetrically, there exists a function $\pi_1 : \mathbb{F}^m \to \Sigma$ *such that*

$$\Pr\left[\mathcal{P} \text{ and } \Pi\,(A_1, A_2)_{|A_1} \overset{\alpha'}{\approx} \pi_{1|A_1} \text{ and } (B_1, B_2)_{|B_1} \overset{\alpha'}{\approx} \pi_{1|B_1}\right]$$
$$\geq \Omega(\varepsilon^{c'}).$$

We prove Lemma 9.2 in Section 9.1.1. We turn to derive Theorem 9.1 from Lemma 9.2.

PROOF (Proof of Theorem 9.1.).   The following proof is for the case where Π is not randomized, but it can be easily extended to the case where Π is randomized (see Remark 9.4 for details). We will choose h to be larger than the constant h' of Lemma 9.2, so we can apply this lemma. Let $\pi_2 : \mathbb{F}^m \to \Sigma$ be the assignment guaranteed by Lemma 9.2, and let Π' be an assignment that is obtained from Π as follows:

1. For every pair (A_1, A_2) for which $\Pi\,(A_1, A_2)_{|A_2} \overset{\alpha'}{\approx} \pi_{2|A_2}$, set $\Pi'\,(A_1, A_2) = \Pi\,(A_1, A_2)$.

2. For every other pair (A_1, A_2), set $\Pi'\,(A_1, A_2) = \bot$, where $\bot$ is some special value on which the test never accepts.

3. Set the pairs (B_1, B_2) similarly.

The probability ε' that the assignment Π' passes the P^2-test is at least $\Omega(\varepsilon^{c'})$ by the definition of π_2. By choosing h to be sufficiently larger than the corresponding constants of Lemma 9.2, we can make sure that ε' satisfies the requirements of Lemma 9.2. Therefore, we can deduce by Lemma 9.2 that there exists an assignment $\pi_1 : \mathbb{F}^m \to \Sigma$ such that

$$\Pr\left[\mathcal{P} \text{ and } \Pi'\,(A_1, A_2)_{|A_1} \overset{\alpha'}{\approx} \pi_{1|A_1} \text{ and } \Pi'\,(B_1, B_2)_{|B_1} \overset{\alpha'}{\approx} \pi_{1|B_1}\right]$$
$$\geq \Omega((\varepsilon')^{c'}) = \Omega(\varepsilon^{(c')^2}).$$

We now choose $c = (c')^2$. Since the test never accepts when Π' answers $\perp$, we deduce that

$$
\Pr\left[\mathcal{P} \text{ and } \Pi(A_1, A_2) \stackrel{\alpha'}{\approx} \left(\pi_{1|A_1}, \pi_{2|A_2}\right) \text{ and} \right.
$$
$$
\left. \Pi\left(B_1, B_2\right) \stackrel{\alpha'}{\approx} \left(\pi_{1|B_1}, \pi_{2|B_2}\right)\right] \geq \Omega(\varepsilon^c).
$$

Choosing h such that $\alpha \geq \alpha'$ completes the proof. $\square$

REMARK 9.3. *Technically speaking, our use of the special value $\perp$ requires formal justification, since when defining the P^2-test and stating Lemma 9.2, we did not allow the use of such a special symbol. To this end, we observe that the use of $\perp$ can be implemented as follows: Let $\Sigma' = \Sigma \cup \{\perp_A, \perp_B\}$, where $\perp_A, \perp_B$ are symbols outside Σ. We first observe that Lemma 9.2 works just as well if we replace the alphabet Σ with the modified alphabet Σ', since Lemma 9.2 is oblivious to the choice of the alphabet. Now, whenever we wish to set $\Pi'(A_1, A_2) = \perp$ in the proof of Theorem 9.1, we actually set $\Pi'(A_1, A_2)$ to be the pair of functions that map all the vectors of A_1 and A_2, respectively, to the symbol $\perp_A$. We deal with the case of $\Pi'(B_1, B_2) = \perp$ similarly, this time using the symbol $\perp_B$. It remains to observe that when assigning $\Pi'(A_1, A_2)$ this way, the P^2-test will always reject $\Pi'(A_1, A_2)$, since the assignment Π' never assigns pairs (B_1, B_2) with the symbol $\perp_A$. The same holds for the case of $\Pi'(B_1, B_2) = \perp$.*

REMARK 9.4. *If Π is randomized, then the definition of Π' in the foregoing proof should be slightly changed to consider the internal randomness of Π. That is, we define Π' to be a randomized assignment and obtain it from Π as follows. For every pair (A_1, A_2) and every internal randomness ω of Π, let us denote by (a_1, a_2) the output of Π on (A_1, A_2) and randomness ω. We define the output of Π' on (A_1, A_2) and randomness ω to be (a_1, a_2) if $a_2 \stackrel{\alpha'}{\approx} \pi_{2|A_2}$ and define it to be $\perp$ otherwise. The definition for pairs (B_1, B_2) is again similar.*

9.1.1. The Proof of Lemma 9.2.   We prove Lemma 9.2 only for the assignment π_2, and the conclusion π_1 can be proved analogously. The proof proceeds in three steps. First, we rely on Theorem 2.2 (soundness of the P-test) to find for each pair of A_1, B_1 a direct product function that agrees (on average) with a good fraction of $\Pi(A_1, \cdot)$ and $\Pi(B_1, \cdot)$. Then, we show that for each A_1 separately, the number of distinct such functions is bounded. Next, we show that there is a single function π such that the probability that the test accepts and $\Pi(A_1, A_2)_{|A_2} \approx \pi_{|A_2}$ is non-negligible (A priori there could have been a different π for each A_1). Finally, we extend the latter result for d_1-subspaces B_1, B_2. Let h_1 be the universal constant whose existence is guaranteed in Theorem 2.2, and let α_1 be the corresponding value from Theorem 2.2.

Step 1.   Consider the bipartite graph corresponding to the P-test, that is, the graph whose left vertices are d_0-subspaces and whose right vertices are d_1-subspaces, and such that a d_0-subspace A_1 is connected to a d_1-subspace B_1 by an edge if and only if $A_1 \subseteq B_1$. We label an edge (A_1, B_1) by $\pi : \mathbb{F}^m \to \Sigma$ if

$$\Pr_{A_2, B_2} \left[\mathcal{P} \text{ and } \Pi(B_1, B_2)_{|B_2} \overset{\alpha_1}{\approx} \pi_{|B_2} \text{ and } \Pi(A_1, A_2)_{|A_2} \overset{\alpha_1}{\approx} \pi_{|A_2} \right]$$
$$\geq \Omega\left(\varepsilon^4\right).$$

If no such π exists, then do not label the edge.

Fix A_1, B_1. We will choose the universal constant h' to be at least $2 \cdot h_1$. If the probability of passing the P^2-test conditioned on A_1, B_1 is at least $\varepsilon/2$, then we claim that the edge is labeled. Indeed, define an assignment $\Pi_{(A_1, B_1)}$ by

$$\Pi_{(A_1, B_1)}(A_2) = \Pi(A_1, A_2)_{|A_2} \text{ and } \Pi_{(A_1, B_1)}(B_2) = \Pi(B_1, B_2)_{|B_2}.$$

If $\Pi_{(A_1, B_1)}$ passes the P-test with probability at least $\varepsilon/2$, then by Theorem 2.2 (soundness of the P-test), there is an assignment π as needed (since $h' \geq 2 \cdot h_1$).

Furthermore, observe that by averaging at least $\varepsilon/2$ of the edges (A_1, B_1) have conditional success at least $\varepsilon/2$, so (A_1, B_1) is labeled.

Step 2. Fix B_1 and let $L(B_1)$ be the labels on edges touching B_1. Consider the following "pruning" process: arbitrarily choose a label $\pi \in L(B_1)$ and remove all elements in $L(B_1)$ that are within relative Hamming distance $3\alpha_1$ of π. Repeat until no more labels can be removed. Let $L'(B_1)$ denote the remaining set of labels. The set $L'(B_1)$ has the following properties

○ Every pair of labels in $L'(B_1)$ are at least $3\alpha_1$ apart and

○ Every $f \in L(B_1)$ is $3\alpha_1$-close to some label in $L'(B_1)$.

We prove that $|L'(B_1)| \leq O(1/\varepsilon^4)$, using an argument in the spirit of the Johnson bound: Suppose $L'(B_1) = \{\pi_1, \pi_2, \ldots\}$ is non-empty. For every $\pi_i \neq \pi_j \in L'(B)$, let us denote

$$p_i \overset{\text{def}}{=} \Pr_{B_2}\left[\Pi\left(B_1, B_2\right)_{|B_2} \overset{\alpha_1}{\approx} \pi_{i|B_2}\right]$$

$$p_{i,j} = \Pr_{B_2}\left[\Pi\left(B_1, B_2\right)_{|B_2} \overset{\alpha_1}{\approx} \pi_{i|B_2} \text{ and } \Pi\left(B_1, B_2\right)_{|B_2} \overset{\alpha_1}{\approx} \pi_{j|B_2}\right].$$

By the definition of the labels π_i, we know that for some universal constant η, it holds that $p_i \geq \eta \cdot \varepsilon^4$ for every π_i. We upper bound the fractions $p_{i,j}$: We know that for every $\pi_i \neq \pi_j$, it holds that $\pi_i \overset{3\cdot\alpha_1}{\not\approx} \pi_j$. It follows that

$$p_{i,j} \leq \Pr_{B_2}\left[\pi_{i|B_2} \overset{2\cdot\alpha_1}{\approx} \pi_{j|B_2}\right]$$

$$\leq 1/\left(q^{d_1 - 2} \cdot \left(\alpha_1 - q^{-d_1}\right)^2\right)$$

$$\leq \frac{1}{2} \cdot \eta^2 \cdot \varepsilon^8,$$

where the second inequality follows by Lemma 2.5 (subspace-point sampler) and the third inequality holds for sufficiently large choice of h'. Now, by the inclusion–exclusion principle that

$$\sum_i p_i - \sum_{i \neq j} p_{i,j} \leq 1$$

$$|L'(B_1)| \cdot \left(\eta \cdot \varepsilon^4\right) - \frac{1}{2}|L'(B_1)|^2 \cdot \left(\frac{1}{2} \cdot \eta^2 \cdot \varepsilon^8\right) \leq 1.$$

The last inequality immediately implies that $|L'(B_1)| \leq 2/(\eta \cdot \varepsilon^4) = O(1/\varepsilon^4)$.

We define $L(A_1)$ similarly and prune it to $L'(A_1)$. Imagine now choosing a random $\pi_{A_1} \in L'(A)$ for each A_1 and a random $\pi_{B_1} \in L'(B_1)$ for each B_1. An edge (A_1, B_1) is called alive if it is labeled by a function π that is $3\alpha'$-close to both π_{A_1} and π_{B_1}. We expect at least $1/|L'(A)||L'(B)| = \Omega(\varepsilon^8)$ fraction of edges to be alive. Fix a choice of π_{A_1} and π_{B_1} for each A_1 and B_1 in a way that attains this expectation.

Step 3. Let $\mathcal{D}_1$ be the distribution of choosing a random d_1-subspace B_1 and two neighbors A_1, A_1' of it in the graph. Let $\mathcal{D}_2$ be the distribution of choosing two d_0-spaces A_1, A_1' independently and a random B_1 that is a common neighbor of them in the graph. The statistical distance between $\mathcal{D}_1$ and $\mathcal{D}_2$ is small:

CLAIM 9.5. *For every $\kappa \in \mathbb{N}$, if the constant h' is sufficiently large, then the distributions $\mathcal{D}_1$ and $\mathcal{D}_2$ are δ-close for $\delta < \varepsilon^{24}/\kappa$.*

We defer the proof of this claim to Section 9.1.2. Now choose a random triplet A_1, A_1', B_1 according to $\mathcal{D}_1$. We lower bound the probability that both edges (A_1, B_1) and (A_1', B_1) are alive. This certainly holds if (i) $\Omega(\varepsilon^8)$ fraction of the edges adjacent to B are alive and (ii) both edges (A_1, B_1) and (A_1', B_1) are alive. Part (i) holds with probability $\Omega(\varepsilon^8)$ and conditioned on this, and Part (ii) holds with probability at least $\Omega(\varepsilon^{16})$. Altogether

$$\Pr_{(B_1,A_1,A_1')\sim\mathcal{D}_1}[(A_1, B_1), (A_1', B_1) \text{ are both alive}] = \Omega(\varepsilon^{24}).$$

Finally, if we let δ be the statistical distance of $\mathcal{D}_1$ and $\mathcal{D}_2$ and apply Claim 9.5 with sufficiently large choices of κ and h', then we have that

$$\Pr_{(B_1,A_1,A_1')\sim\mathcal{D}_2}[(A_1, B_1), (A_1', B_1) \text{ are both alive}]$$
$$\geq \Omega(\varepsilon^{24}) - \delta = \Omega(\varepsilon^{24}).$$

Now fix A_1 such that the above holds when conditioning on A_1. This means that for at least $\Omega(\varepsilon^{24})$ fraction of the d_0-subspaces A_1',

there exists a d_1-subspace B_1 such that both the edges (A_1, B_1) and (A_1', B_1) are alive. For each such A_1', it holds that the label of (A_1', B_1) is $3\alpha_1$-close to π_{B_1}, which in turn is $3\alpha_1$-close to the label of the edge (A_1, B_1), which is $3\alpha_1$-close to π_{A_1}. Thus, the label of (A_1', B_1) is $9\alpha_1$-close to π_{A_1}. Let us denote by $\pi_{(A_1', B_1)}$ the label of the edge (A_1', B_1). Recall that by the definition of $\pi_{(A_1', B_1)}$, it holds that

$$(9.6) \quad \Pr_{A_2, B_2} \left[\mathcal{P} \text{ and } \Pi(A_1', A_2)_{|A_2} \overset{\alpha_1}{\approx} \pi_{(A_1', B_1)|A_2} \right] \geq \Omega\left(\varepsilon^4\right).$$

Since $\pi_{(A_1', B_1)} \overset{9 \cdot \alpha_1}{\approx} \pi_A$, it holds by Lemma 2.5 (subspace-point sampler) that for a uniformly distributed d_0-subspace A_2:

$$\Pr_{A_2} \left[\pi_{(A_1', B_1)|A_2} \overset{10 \cdot \alpha_1}{\not\approx} \pi_{A_1|A_2} \right] \leq \frac{1}{q^{d_0 - 2} \cdot \left(\alpha_1 - q^{-d_0}\right)^2}.$$

The latter expression can be made smaller than any constant times ε^4 by choosing h' to be sufficiently large. By subtracting that expression from Inequality (9.6), we obtain that

$$\Pr_{A_2, B_2} \Big[\mathcal{P} \text{ and } \Pi(A_1', A_2)_{|A_2} \overset{\alpha_1}{\approx} \pi_{(A_1', B_1)|A_2} \text{ and }$$
$$\pi_{(A_1', B_1)|A_2} \overset{10 \cdot \alpha_1}{\approx} \pi_{A_1|A_2} \Big] \geq \Omega\left(\varepsilon^4\right).$$

By letting $\pi_2 = \pi_{A_1}$ and choosing $c' = 28$, we have by the triangle inequality

$$\Pr_{A_1', A_2} \left[\mathcal{P} \text{ and } \Pi(A_1', A_2)_{|A_2} \overset{11 \cdot \alpha_1}{\approx} \pi_{2|A_2} \right] \geq \Omega(\varepsilon^{24}) \cdot \Omega\left(\varepsilon^4\right) = \Omega(\varepsilon^{c'}).$$
$$(9.7)$$

Step 4. It remains to show that the assignment Π agrees with π_2 on a non-negligible fraction of the B's. To this end, we observe that

$$\Pr \left[\mathcal{P} \text{ and } \Pi(A_1, A_2)_{|A_2} \overset{11 \cdot \alpha_1}{\approx} \pi_{2|A_2} \,\middle|\, \Pi(B_1, B_2)_{|B_2} \overset{12 \cdot \alpha_1}{\not\approx} \pi_{2|B_2} \right]$$
$$(9.8) \qquad \leq \frac{1}{q^{d_0 - 2} \cdot \left(\alpha_1/2\right)^2}.$$

To see it, note that it suffices to prove that

$$\Pr\left[\Pi\left(B_1, B_2\right)_{|A_2} \overset{11\cdot\alpha_1}{\approx} \pi_{2|A_2} \,\middle|\, \Pi\left(B_1, B_2\right)_{|B_2} \overset{12\cdot\alpha_1}{\not\approx} \pi_{2|B_2}\right]$$

$$\leq \frac{1}{q^{d_0-2}\cdot\left(\alpha_1 - q^{-d_0}\right)^2} \leq \frac{1}{q^{d_0-2}\cdot\left(\alpha_1/2\right)^2}.$$

The latter inequality is an immediate corollary of Lemma 2.5 (subspace-point sampler).

Now, by choosing h' to be sufficiently large, so that the upper bound in Inequality (9.8) is sufficiently smaller than $\varepsilon^{c'}$, and by combining Inequality (9.7) with Inequality (9.8), we obtain that

$$\Pr\left[\mathcal{P} \text{ and } \Pi\left(A_1, A_2\right)_{|A_2} \overset{11\cdot\alpha_1}{\approx} \pi_{2|A_2} \text{ and } \Pi\left(B_1, B_2\right)_{|B_2} \overset{12\cdot\alpha_1}{\approx} \pi_{2|B_2}\right]$$

$$\geq \Omega(\varepsilon^{c'}).$$

By setting h' such that $\alpha' \geq 12 \cdot \alpha_1$, this concludes the proof of Lemma 9.2. □

9.1.2. Proofs of Auxiliary Claim.

PROOF (Proof of Claim 9.5.).   Fix $\kappa \in \mathbb{N}$. In order to prove the claim, consider the event J which holds if and only if A and A' are independent. We argue that

$$\mathcal{D}_1 \overset{\delta/2}{\approx} \mathcal{D}_1|J = \mathcal{D}_2|J \overset{\delta/2}{\approx} D_2.$$

The fact that $\mathcal{D}_1|J = \mathcal{D}_2|J$ is exactly Proposition 1. We show that $\mathcal{D}_1 \overset{\delta/2}{\approx} \mathcal{D}_1|J$ and $\mathcal{D}_2 \overset{\delta/2}{\approx} \mathcal{D}_2|J$. The statistical distance between $\mathcal{D}_1$ and $\mathcal{D}_1|J$ (respectively, $\mathcal{D}_2$ and $\mathcal{D}_2|J$) is exactly the probability that the event J does not occur under $\mathcal{D}_1$ (respectively $\mathcal{D}_2$). It follows immediately from Proposition 2.18 that $\Pr_{\mathcal{D}_1}\left[\neg J\right] \leq 2 \cdot d_0/q^{d_1-2\cdot d_0}$ and $\Pr_{\mathcal{D}_2}\left[\neg J\right] \leq 2 \cdot d_0/q^{m-2\cdot d_0}$. Both the latter expressions can indeed be made smaller than ε^{24}/κ by choosing sufficiently large h', as required. □

9.2. The Proof of Theorem 5.4 and Theorem 8.7.   In the rest of this section, we prove Theorem 5.4 and Theorem 8.7.

THEOREM (5.4, the soundness of the S-test, restated). *There exists a universal constants $h, c \in \mathbb{N}$ such that the following holds: Let $\varepsilon \geq h \cdot d_0 \cdot q^{-d_0/h}, \alpha \overset{\text{def}}{=} h \cdot d_0 \cdot q^{-d_0/h}$. Assume that $d_1 \geq h \cdot d_0, m \geq h \cdot d_1$. Suppose that a (possible randomized) assignment Π passes the S-test with probability at least ε. There exists an assignment $\pi : \mathbb{F}^m \to \Sigma$ for which the following holds. Let B_1, B_2 be uniformly distributed and independent d_1-subspaces of $\mathbb{F}^m$, and let A_1 and A_2 be uniformly distributed d_0-subspaces of B_1 and B_2 respectively, and denote $A = A_1 + A_2$. Then:*

$$\Pr\left[\Pi\left(B_1, B_2\right)_{|(A_1,A_2)} = \Pi\left(A\right)_{|(A_1,A_2)} \text{ and } \Pi\left(B_1, B_2\right) \overset{\alpha}{\approx} \pi_{|(B_1,B_2)}\right]$$
$$= \Omega\left(\varepsilon^c\right).$$

REMARK 9.9. *Note that in the foregoing restatement of Theorem 5.4, we denote the first universal constant by h, while in its original statement it was denoted by h'.*

The intuition that underlies the proof is the following. Consider an adversary the chooses the proof Π. Since the S-test essentially contains a P^2-test, the adversary must choose the assignment Π such that for random d_0-subspaces A_1 and A_2, the assignment $\Pi\left(A_1 + A_2\right)_{|(A_1,A_2)}$ is consistent with two assignments π_1 and π_2 on A_1 and A_2, respectively. On the other hand, given the sum $A_1 + A_2$, the adversary cannot deduce the choices of A_1 and A_2, and therefore, he must label both A_1 and A_2 with the same assignment in order to make the S-test accept. We conclude that π_1 and π_2 must be essentially the same. Details follow.

Let h' be the universal constant whose existence guaranteed in Theorem 9.1 (soundness of the P^2-test), and let α' be the corresponding value from Theorem 9.1. We choose c to be the same constant as in Theorem 9.1 and will choose the universal constant h to be at least h'.

Fix an assignment Π that passes the S-test with probability at least ε. We define a new assignment Π' that assigns values to pairs of d_0-subspaces and to pairs of d_1-subspaces of $\mathbb{F}^m$ (not necessarily independent) by choosing $\Pi'\left(B_1, B_2\right)$ (respectively, $\Pi'\left(A_1, A_2\right)$) to be equal to $\Pi\left(B_1, B_2\right)$ (respectively, $\Pi\left(A_1 + A_2\right)$) if B_1 and B_2

(respectively, A_1 and A_2) are independent and choosing Π' to be arbitrary otherwise. Observe that the assignment Π' passes the P^2-test whenever B_1 and B_2 are independent and Π passes the S-test. Furthermore, the probability that two uniformly distributed d_1-subspaces B_1 and B_2 of $\mathbb{F}^m$ are not independent is at most $d_1/q^{m-2\cdot d_1}$ by Proposition 2.18, and therefore, Π' passes the P^2-test with probability at least $\varepsilon - d_1/q^{m-2\cdot d_1}$. For a sufficiently large choice of h, the latter probability is at least $\Omega(\varepsilon)$, and also matches the requirements of Theorem 9.1 (soundness of the P^2-test), so we can apply this theorem. It follows that there exist assignments $\pi_1, \pi_2 : \mathbb{F}^m \to \Sigma$ such that for uniformly distributed (not necessarily independent) $B_1, B_2, A_1 \subseteq B_1, A_2 \subseteq B_2$ it holds that

$$\Pr\left[\Pi'(B_1, B_2)_{|(A_1, A_2)} = \Pi'(A_1, A_2)\right.$$

$$\text{and } \Pi'(A_1, A_2) \overset{\alpha'}{\approx} \left(\pi_{1|A_1}, \pi_{2|A_2}\right)$$

$$\left.\text{and } \Pi'(B_1, B_2) \overset{\alpha'}{\approx} \left(\pi_{1|B_1}, \pi_{2|B_2}\right)\right]$$

(9.10)
$$= \Omega(\varepsilon^c).$$

The probability that B_1 and B_2 are not independent is at most $d_1/q^{m-2\cdot d_1}$, and the latter expression can be made smaller than any constant factor times ε^c by choosing h to be sufficiently large. Thus, Inequality (9.10) also holds for uniformly distributed *independent* B_1 and B_2. We now argue that

CLAIM 9.11. *For sufficiently large choice of h, it holds that* $\pi_1 \overset{5\cdot\alpha'}{\approx} \pi_2$.

We defer the proof of Claim 9.11 to the end of this section. We turn to prove the theorem. By Inequality (9.10), it holds for uniformly distributed and independent d_1-subspaces B_1 and B_2 of $\mathbb{F}^m$ that

$$\Pr\left[\Pi'(B_1, B_2)_{|(A_1, A_2)} = \Pi'(A_1, A_2) \text{ and}\right.$$

$$\left.\Pi(B_1, B_2) \overset{\alpha'}{\approx} \left(\pi_{1|B_1}, \pi_{2|B_2}\right)\right] \geq \Omega(\varepsilon^c).$$

By Claim 9.11, it holds that $\pi_1 \overset{5 \cdot \alpha'}{\approx} \pi_2$. Since B_2 is a uniformly distributed d_1-subspace of $\mathbb{F}^m$, this implies by Lemma 2.5 (subspace-point sampler) that

$$\Pr\left[\pi_{1|B_2} \overset{6 \cdot \alpha'}{\approx} \pi_{2|B_2}\right] \geq 1 - \frac{1}{q^{d_1-2} \cdot (\alpha' - q^{-d_1})^2} \geq 1 - \frac{1}{q^{d_1-2} \cdot (\alpha'/2)^2}.$$

We conclude that

$$\Pr\left[\Pi'(B_1, B_2)_{|(A_1, A_2)} = \Pi'(A_1, A_2) \text{ and } \Pi(B_1, B_2) \overset{7 \cdot \alpha'}{\approx} \left(\pi_{1|B_1}, \pi_{1|B_2}\right)\right]$$

$$\geq \Pr\left[\Pi'(B_1, B_2)_{|(A_1, A_2)} = \Pi'(A_1, A_2) \text{ and } \Pi(B_1, B_2) \overset{\alpha'}{\approx} \left(\pi_{1|B_1}, \pi_{2|B_2}\right)\right.$$

$$\left. \text{and } \pi_{1|B_2} \overset{6 \cdot \alpha'}{\approx} \pi_{2|B_2}\right]$$

$$= \Omega\left(\varepsilon^c\right) - \frac{1}{q^{d_1-2} \cdot (\alpha'/2)^2}$$

$$= \Omega\left(\varepsilon^c\right),$$

where the last equality holds for sufficiently large choice of h. The theorem now follows by defining $\pi = \pi_1$ and setting h to be sufficiently large such that $\alpha = 7 \cdot \alpha'$. □

PROOF (Proof of Claim 9.11.).   For the sake of contradiction, assume that $\pi_1 \overset{5 \cdot \alpha'}{\not\approx} \pi_2$. Let A be a uniformly distributed $2 \cdot d_0$-subspace A of $\mathbb{F}^m$ and let A_1 and A_2 be uniformly distributed and independent d_0-subspaces of A. By Lemma 2.5, it holds that

$$\Pr\left[\pi_{1|A} \overset{4 \cdot \alpha'}{\not\approx} \pi_{2|A}\right] \geq 1 - \frac{1}{q^{2 \cdot d_0 - 2} \cdot (\alpha' - q^{-2d_0})^2}$$

$$\geq 1 - \frac{1}{q^{2 \cdot d_0 - 2} \cdot (\alpha'/2)^2}.$$

If $\pi_{1|A} \overset{4 \cdot \alpha'}{\not\approx} \pi_{2|A}$ then by the triangle inequality, it either holds that $\Pi(A) \overset{2 \cdot \alpha'}{\not\approx} \pi_{1|A}$ or that $\Pi(A) \overset{2 \cdot \alpha'}{\not\approx} \pi_{2|A}$. Since A_1 is a uniformly distributed d_0-subspace of A, it holds by Lemma 2.5 (subspace-point sampler) that

$$\Pr\left[\Pi(A)_{|A_1} \overset{\alpha'}{\not\approx} \pi_{1|A_1} \,\middle|\, \Pi(A) \overset{2 \cdot \alpha'}{\not\approx} \pi_{1|A}\right] \geq 1 - \frac{1}{q^{2 \cdot d_0 - 2} \cdot (\alpha'/2)^2}.$$

A similar claim can be made for π_2 and A_2. Now, if either $\Pi(A)_{|A_1} \overset{\alpha'}{\not\approx} \pi_{1|A_1}$ or $\Pi(A)_{|A_2} \overset{\alpha'}{\not\approx} \pi_{2|A_2}$, then by definition it holds that $\Pi(A)_{|(A_1,A_2)} \overset{\alpha'}{\not\approx} (\pi_{1|A_1}, \pi_{2|A_2})$. We conclude that

$$\Pr\left[\Pi(A)_{|(A_1,A_2)} \overset{\alpha'}{\not\approx} (\pi_{1|A_1}, \pi_{2|A_2}) \,\middle|\, \pi_{1|A} \overset{4\cdot\alpha'}{\not\approx} \pi_{2|A}\right]$$

$$\geq 1 - \frac{1}{q^{2\cdot d_0 - 2} \cdot (\alpha'/2)^2},$$

and therefore by lifting the conditioning and substituting $A = A_1 + A_2$, we obtain that for a uniformly distributed and independent d_0-subspaces A_1 and A_2 of $\mathbb{F}^m$, it holds that

$$\Pr\left[\Pi(A_1 + A_2)_{|(A_1,A_2)} \overset{\alpha'}{\approx} (\pi_{1|A_1}, \pi_{2|A_2})\right] \leq \frac{2}{q^{2\cdot d_0 - 2} \cdot (\alpha'/2)^2}.$$

On the other hand, by the definition of Π', Inequality (9.10) implies that for uniformly distributed and independent d_0-subspaces A_1 and A_2 of $\mathbb{F}^m$, it holds that

$$\Pr\left[\Pi(A_1 + A_2)_{|(A_1,A_2)} \overset{\alpha'}{\approx} (\pi_{1|A_1}, \pi_{2|A_2})\right] \geq \Omega(\varepsilon^c).$$

By choosing h to be sufficiently large, the latter lower bound can be made larger than $2/\left(q^{2\cdot d_0 - 2} \cdot (\alpha')^2\right)$, and this is a contradiction.
$\square$

THEOREM 9.12 (8.7, list-decoding soundness of the S-test, restated). *There exist universal constants $h, c \in \mathbb{N}$ such that for every $d_0 \in \mathbb{N}, d_1 \geq h \cdot d_0$, and $m \geq h \cdot d_1$, the following holds: Let $\varepsilon \geq h \cdot d_0 \cdot q^{-d_0/h}, \alpha \overset{\text{def}}{=} h \cdot d_0 \cdot q^{-d_0/h}$. Let Π be a (possibly randomized) assignment to $2d_0$-subspaces of $\mathbb{F}^m$ and to pairs of d_1-subspaces of $\mathbb{F}^m$. Then, there exists a (possibly empty) list of $L = O(1/\varepsilon^c)$ assignments $\pi^1, \ldots, \pi^L : \mathbb{F}^m \to \Sigma$ such that*

$$\Pr\left[\Pi(B_1, B_2)_{|(A_1,A_2)} = \Pi(A)_{|(A_1,A_2)}\right.$$

$$\left. \text{and } \nexists i \in [L] \text{ s.t. } \Pi(B_1, B_2) \overset{\alpha}{\approx} \pi^i_{|(B_1,B_2)}\right] < \varepsilon.$$

REMARK 9.13. *Note that in the foregoing restatement of Theorem 8.7, we denote the first universal constant by h, while in its original statement it was denoted by h'.*

The basic idea of the proof is as follows. We apply Theorem 5.4 to Π, thus "decoding" from it an assignment π^1. We then remove from Π the places at which it roughly agrees with π^1, resulting in an assignment Π^2. If the assignment Π^2 is accepted by the S-test with probability less than ε, then we are finished—the required list of assignments in this case consists only of π^1. Otherwise, the assignment Π^2 is accepted by the S-test with probability at least ε, and we can therefore "decode" a second assignment π^2 from Π^2. Next, we remove from Π^2 the places at which it roughly agrees with π^2, resulting in an assignment Π^3. We proceed in this manner, each time obtaining new assignments Π^i and π^i, until the conclusion of Theorem 8.7 holds.

 We prove Theorem 8.7 only for non-randomized assignments Π, but the proof can easily be extended to randomized assignments, see Remark 9.18 for details. We choose the constants h and c to be the same as in Theorem 5.4. If the S-test accepts Π with probability less than ε, then the theorem holds vacuously. We thus assume that the S-test accepts Π with probability at least ε. We show that for $L = O\left(1/\varepsilon^c\right)$, there exist assignments $\pi^1, \ldots, \pi^L : \mathbb{F}^m \to \Sigma$ such that

$$\Pr\left[\Pi\left(B_1, B_2\right)_{|(A_1, A_2)} = \Pi\left(A\right)_{|(A_1, A_2)}\right]$$
$$- \Pr\left[\Pi\left(B_1, B_2\right)_{|(A_1, A_2)} = \Pi\left(A\right)_{|(A_1, A_2)}\right.$$
$$\left.\text{and } \exists i \in [L] : \Pi\left(B_1, B_2\right) \overset{\alpha}{\approx} \pi^i_{|(B_1, B_2)}\right]$$

(9.14) $\leq \varepsilon.$

We construct the assignments $\pi^1, \ldots, \pi^L$ as follows. We begin by applying Theorem 5.4 to Π, obtaining the assignment π^1, and set $\Pi^1 \overset{\text{def}}{=} \Pi$. Then, for each $i \geq 1$, we define an assignment Π^{i+1} as follows.

1. For every pair of d_1-subspaces B_1, B_2 such that $\Pi^i\left(B_1, B_2\right) \overset{\alpha}{\approx} \pi^i_{|(B_1, B_2)}$, we set $\Pi^{i+1}\left(B_1, B_2\right) = \perp$, where $\perp$ is a special symbol

that the test always rejects. This is our formal way of "removing" $\Pi^i (B_1, B_2)$.

2. For every pair of d_1-subspaces B_1, B_2 such that $\Pi^i (B_1, B_2) \overset{\alpha}{\not\approx} \pi^i_{|(B_1,B_2)}$, we set $\Pi^{i+1} (B_1, B_2) = \Pi^i (B_1, B_2)$.

3. For every $2d_0$-subspace A, we set $\Pi^{i+1} (A) = \Pi^i (A)$.

Now, observe that

$$\Pr \left[\Pi^{i+1} (B_1, B_2)_{|(A_1,A_2)} = \Pi^{i+1} (A)_{|(A_1,A_2)} \right]$$

$$= \Pr \left[\Pi^i (B_1, B_2)_{|(A_1,A_2)} = \Pi^i (A)_{|(A_1,A_2)} \right]$$

$$(9.15) \quad - \Pr \left[\Pi^i (B_1, B_2)_{|(A_1,A_2)} = \Pi^i (A)_{|(A_1,A_2)} \wedge \Pi^i (B_1, B_2) \overset{\alpha}{\approx} \pi^i_{|(B_1,B_2)} \right],$$

since we must have $\Pi^{i+1} (B_1, B_2)_{|(A_1,A_2)} \neq \Pi^{i+1} (A)_{|(A_1,A_2)}$ whenever $\Pi^{i+1} (B_1, B_2)_{|(A_1,A_2)} = \perp$, and the latter occurs whenever $\Pi^i (B_1, B_2) \overset{\alpha}{\approx} \pi^i_{|(B_1,B_2)}$. If

$$\Pr \left[\Pi^{i+1} (B_1, B_2)_{|(A_1,A_2)} = \Pi^{i+1} (A)_{|(A_1,A_2)} \right] < k\varepsilon$$

then we set $L = i$ and finish the construction. Otherwise, we construct π^{i+1} by applying Theorem 5.4 to the assignment Π^{i+1} and setting π^{i+1} to be the resulting assignment.

It is easy to prove by induction that for every $i \in [L]$, it holds that

$$\Pr \left[\Pi^{i+1} (B_1, B_2)_{|(A_1,A_2)} = \Pi^{i+1} (A)_{|(A_1,A_2)} \right]$$

$$= \Pr_{A \subseteq B} \left[\Pi (B_1, B_2)_{|(A_1,A_2)} = \Pi (A)_{|(A_1,A_2)} \right]$$

$$- \Pr_{A \subseteq B} \left[\Pi (B_1, B_2)_{|(A_1,A_2)} = \Pi (A)_{|(A_1,A_2)} \right.$$

$$(9.16) \qquad \left. \text{and } \exists i \in [L] : \Pi_i (B_1, B_2) \overset{\alpha}{\approx} \pi^i_{|(B_1,B_2)} \right].$$

The proof of the Equality (9.16) goes essentially by summing over the probabilities of events of the form

$$\Pi^i (B_1, B_2)_{|(A_1,A_2)} = \Pi^i (A)_{|(A_1,A_2)}$$

and $\Pi^i (B_1, B_2) \overset{\alpha}{\approx} \pi^j_{|(B_1,B_2)}$ and $\not\exists j < i$ s.t. $\Pi^j (B_1, B_2) \overset{\alpha}{\approx} \pi^j_{|(B_1,B_2)}$,

for different values of i.

Finally, by combining Equality (9.16) with the fact that

$$\Pr\left[\Pi^{L+1}(B_1, B_2)_{|(A_1,A_2)} = \Pi^{L+1}(A)_{|(A_1,A_2)}\right] < \varepsilon,$$

it follows that the assignments $\pi^1, \ldots, \pi^L$ satisfy Inequality (9.14). To see that $L = O(1/\varepsilon^c)$, observe that for each i we have that

$$\Pr\left[\Pi^i(B_1, B_2)_{|(A_1,A_2)} = \Pi^i(A)_{|(A_1,A_2)} \text{ and } \Pi^i(B_1, B_2) \overset{\alpha}{\approx} \pi^i_{|(B_1,B_2)}\right]$$
$$= \Omega(\varepsilon^c).$$

By Equality (9.15), this implies that the acceptance probability of Π^{i+1} is smaller than the acceptance probability of Π^i by at least ε^c, and therefore, that the number of iterations can be at most $O(1/\varepsilon^c)$, as required.

REMARK 9.17. *As in the proof of Theorem 9.1 (soundness of the P^2-test), the use of the special symbol $\perp$ requires formal justification. This can be done as explained in Remark 9.3.*

REMARK 9.18. *As in the proof of Theorem 9.1 (soundness of the P^2-test), if Π is randomized, then for each i the definition of Π^{i+1} should be slightly changed to consider the internal randomness of Π^i. That is, we define Π^{i+1} to be a randomized assignment and obtain it from Π as follows. For every pair (B_1, B_2) and every internal randomness ω of Π^i, let us denote by (b_1, b_2) the output of Π_i on (B_1, B_2) and randomness ω. We define the output of Π^{i+1} on (B_1, B_2) and randomness ω to be $\perp$ if $(b_1, b_2) \overset{\alpha'}{\approx} \pi^i_{|(B_1,B_2)}$ and define it to be (b_1, b_2) otherwise. The definition for $2d_0$-spaces A can be changed similarly to include the internal randomness of Π^i.* $\square$

9.3. Proof of Theorem 2.2, Soundness of the P-test. In this section, we prove Theorem 2.2, restated below, by adapting the analysis of Impagliazzo *et al.* (2009) (in particular, Sections 3.4 and 4) to the setting of the P-test, while relying on a lemma of Impagliazzo *et al.* (2009). Let $\mathbb{F}$ be a finite field of size q, let $m, d_0, d_1 \in \mathbb{N}$, and consider a (possible randomized) assignment Π that assigns values to d_0- and d_1-subspaces of $\mathbb{F}^m$.

THEOREM 9.19 (2.2, soundness of the P-test, restated). *There exists a universal constant $h \in \mathbb{N}$ such that the following holds: Let $\varepsilon \geq h \cdot d_0 \cdot q^{-d_0/h}, \alpha \stackrel{\text{def}}{=} h \cdot d_0 \cdot q^{-d_0/h}$. Assume that $d_1 \geq h \cdot d_0, m \geq h \cdot d_1$. Suppose that an assignment Π passes the P-test with probability at least ε. Then, there exists an assignment π such that*

$$\Pr\left[\Pi(B)_{|A} = \Pi(A) \text{ and } \Pi(B) \stackrel{\alpha}{\approx} \pi_{|B} \text{ and } \Pi(A) \stackrel{\alpha}{\approx} \pi_{|A}\right] = \Omega(\varepsilon^4),$$

where the probability is over A, B chosen as in the P-test.

We begin by recalling the required preliminaries from Impagliazzo *et al.* (2009) and then turn to prove Theorem 2.2.

DEFINITION 9.20 (Good). *Let A be a d_0-subspace of $\mathbb{F}^m$ and let $\varepsilon \in (0,1)$. We say that A is ε-good (with respect to an assignment Π) if for a uniformly distributed d_1-dimensional subspace B that contains A, it holds that*

$$\Pr\left[\Pi(B)_{|A} = \Pi(A)\right] \geq \varepsilon,$$

where the randomness is over the choice of B and over the randomness of Π.

DEFINITION 9.21 (Plurality function). *Let A be a d_0-subspace of $\mathbb{F}^m$. We denote by $\pi_A : \mathbb{F}^m \to \Sigma$ the plurality function of A (with respect to Π). In other words, for every $x \in \mathbb{F}^m$, we define $\pi_A(x)$ to be the value $v \in \Sigma$ that maximizes*

$$\Pr_{B \supseteq A}\left[\Pi(B)_{|x} = v \,\middle|\, \Pi(B)_{|A} = \Pi(A)\right],$$

where B is a uniformly distributed d_1-dimensional subspace that contains A.

DEFINITION 9.22 (DP-consistent). *Let A be a d_0-subspace of $\mathbb{F}^m$ and let $\alpha, \gamma \in (0,1)$. We say that A is $(\varepsilon, \alpha, \gamma)$-direct product consistent (abbreviated $(\varepsilon, \alpha, \gamma)$ -DP-consistent) if A is ε-good and it holds that*

$$\Pr_{B \supseteq A}\left[\Pi(B) \stackrel{\alpha}{\approx} \pi_{A|B} \,\middle|\, \Pi(B)_{|A} = \Pi(A)\right] \geq 1 - \gamma.$$

The following lemma is a direct corollary of the proof of Impagliazzo *et al.* (2009), Lemmas 4.2 and 4.4.

LEMMA 9.23. *There exists a universal constant $h_0 \in \mathbb{N}$ such that the following holds: Let $\varepsilon \geq h_0 \cdot q^{-(d_1/h_0 - d_0)}$ and $\alpha, \gamma \in (0, 1)$. The probability that a uniformly distributed A is ε-good but not $(\varepsilon, \alpha, \gamma)$-DP-consistent is at most $O\big(1/\big(\alpha \cdot \gamma \cdot \varepsilon^2 \cdot q^{d_0 - 2}\big)\big)$.*

Proof of Theorem 2.2.   We will choose the universal constant h to be larger than h_0 (where h_0 is the constant from Lemma 9.23). Assume that the P-test accepts with probability at least ε as in the statement of the theorem. Let $\varepsilon_1 = \frac{1}{3} \cdot \varepsilon$ and $\gamma_1 = \varepsilon_1^3/h$. Choose $\alpha_1 = O\big(1/\varepsilon_1^3 \cdot \gamma_1 \cdot q^{d_0 - 2}\big)$ such that the probability in Lemma 9.23 that A is ε_1-good but not $(\varepsilon_1, \alpha_1, \gamma_1)$-DP-consistent is at most ε_1, which is indeed possible for sufficiently large choice of h. We will later choose $\alpha = O(\alpha_1)$, by choosing again h to be sufficiently large.

   We consider the following sequence of events. Let A_1, A_2 denote random d_0-subspaces, and let B denote a random d_1-subspace and define events $\mathcal{S}_1, \mathcal{S}_2, \mathcal{S}_3$ as follows:

1. $\mathcal{S}_1(A_1, A_2, B)$: A_1 and A_2 are $(\varepsilon_1, \alpha_1, \gamma_1)$-DP-consistent and $\Pi(B)_{|A_1} = \Pi(A_1)$, $\Pi(B)_{|A_2} = \Pi(A_2)$.

2. $\mathcal{S}_2(A_1, A_2, B)$: The event $\mathcal{S}_1(A_1, A_2, B)$ occurs and $\pi_{A_1|B} \overset{2\alpha_1}{\approx} \pi_{A_2|B}$ (recall that π_{A_1} and π_{A_2} are the plurality assignments of A_1 and A_2 respectively).

3. $\mathcal{S}_3(A_1, A_2)$: A_1 and A_2 are $(\varepsilon_1, \alpha_1, \gamma_1)$-DP-consistent and $\pi_{A_1} \overset{3\alpha_1}{\approx} \pi_{A_2}$.

In the next three claims, we choose A_1, A_2, and B according to the following distribution: choose A_1 and A_2 to be uniformly distributed and independent d_0-spaces A_1, A_2 and a choose B to be a uniformly distributed d_1-subspace that contains them. We show that the probability of events $\mathcal{S}_1, \mathcal{S}_2, \mathcal{S}_3$ under this distribution is non-negligible.

CLAIM 9.24. $Pr[\mathcal{S}_1] \geq \Omega(\varepsilon_1^3)$.

PROOF.   Let B' be a uniformly distributed d_1-subspace of $\mathbb{F}^m$ and let A' be a d_0-uniformly distributed subspace of B'. We begin by lower bounding the probability

$$\Pr\left[\Pi(B')_{|A'} = \Pi(A') \text{ and } A' \text{ is } (\varepsilon_1, \alpha_1, \gamma_1)\text{-DP-consistent}\right].$$
(9.25)

To this end, let us denote by $\mathcal{P}$ the event that $\Pi(B')_{|A'} = \Pi(A')$, by $\mathcal{D}$ the event that A' is $(\varepsilon_1, \alpha_1, \gamma_1)$-DP-consistent, and by $\mathcal{G}$ the event that A' is ε_1-good. Observe that $\Pr[\mathcal{P} \text{ and } \neg\mathcal{G}] \leq \Pr[\mathcal{P}|\neg\mathcal{G}] \leq \varepsilon_1$. Furthermore, A' is a uniformly distributed d_0-subspace of $\mathbb{F}^m$, and thus by Lemma 9.23 and our choice of α_1, it holds that $\Pr[\mathcal{G} \text{ and } \neg\mathcal{D}] \leq \varepsilon_1$. Finally, it holds that the probability in (9.25) is

$$\begin{aligned}
\Pr[\mathcal{P} \text{ and } \mathcal{D}] &\geq \Pr[\mathcal{P} \text{ and } \mathcal{G} \text{ and } \mathcal{D}] \\
&= \Pr[\mathcal{P} \text{ and } \mathcal{G}] - \Pr[\mathcal{P} \text{ and } \mathcal{G} \text{ and } \neg\mathcal{D}] \\
&= \Pr[\mathcal{P}] - \Pr[\mathcal{P} \text{ and } \neg\mathcal{G}] - \Pr[\mathcal{P} \text{ and } \mathcal{G} \text{ and } \neg\mathcal{D}] \\
&\geq \Pr[\mathcal{P}] - \Pr[\mathcal{P} \text{ and } \neg\mathcal{G}] - \Pr[\mathcal{G} \text{ and } \neg\mathcal{D}] \\
&\geq \varepsilon - \varepsilon_1 - \varepsilon_1 \\
&\geq \varepsilon_1.
\end{aligned}$$

So the probability in (9.25) is at least ε_1. By averaging, this implies that for $\Omega(\varepsilon_1)$ fraction of the d_1-subspaces B', it holds that at least $\Omega(\varepsilon_1)$ fraction of the d_0-subspaces A' of B' are $(\varepsilon_1, \alpha_1, \gamma_1)$-DP-consistent and satisfy $\Pi(B')_{|A'} = \Pi(A')$.

Now, observe that by Proposition 1, the distribution over A_1, A_2, B is equivalent to choosing B to be a uniformly distributed d_1-subspace of $\mathbb{F}^m$ and then choosing A_1 and A_2 to be independent uniformly distributed d_0-subspaces of B. With probability at least $\Omega(\varepsilon_1)$ it holds for B that at least $\Omega(\varepsilon_1)$ fraction of the d_0-subspaces A of B are $(\varepsilon_1, \alpha_1, \gamma_1)$-DP-consistent and satisfy $\Pi(B)_{|A} = \Pi(A)$. We condition on the latter event, and claim that under this conditioning the event $\mathcal{S}_1(A_1, A_2, B)$ occurs with probability at least $\Omega(\varepsilon_1^2)$. To see it, consider two uniformly distributed (*not necessarily independent*) d_0-subspaces A_1' and A_2' of B. Then, by our

conditioning, it holds that $\mathcal{S}_1(A'_1, A'_2, B)$ occurs with probability at least $\Omega\left(\varepsilon_1^2\right)$. Furthermore, by Proposition 2.18 it holds with probability at least $1 - 2 \cdot d_0/q^{d_1 - 2 \cdot d_0}$ that A'_1 and A'_2 are independent. It therefore follows under the foregoing conditioning on B that

$$
\begin{aligned}
\Pr\left[\mathcal{S}_1(A_1, A_2, B)\right] &= \Pr\left[\mathcal{S}_1(A'_1, A'_2, B)\mid A'_1, A'_2 \text{ are disjoint}\right] \\
&\geq \Pr\left[\mathcal{S}_1(A'_1, A'_2, B) \text{ and } A'_1, A'_2 \text{ are disjoint}\right] \\
&\geq \Pr\left[\mathcal{S}_1(A'_1, A'_2, B)\right] - \Pr\left[A'_1, A'_2 \text{ are disjoint}\right] \\
&\geq \Omega\left(\varepsilon_1^2\right) - 2 \cdot d_0/q^{d_1 - 2 \cdot d_0} \\
&\geq \Omega\left(\varepsilon_1^2\right),
\end{aligned}
$$

where the last inequality holds for sufficiently large h. Lifting the conditioning on B, we get that for a uniformly distributed d_1-subspace B of $\mathbb{F}^m$ and two independent uniformly distributed d_0-subspaces A_1 and A_2 of B, it holds with probability at least $\Omega\left(\varepsilon_1^3\right)$ that both A_1 and A_2 are $(\varepsilon_1, \alpha_1, \gamma_1)$-DP-consistent and that $\Pi\left(B\right)_{|A_1} = \Pi\left(A_1\right), \Pi\left(B\right)_{|A_2} = \Pi\left(A_2\right)$, as required.　□

CLAIM 9.26. $\Pr[\mathcal{S}_2] \geq \Omega\left(\varepsilon_1^3\right)$.

PROOF.　Let $\mathcal{E}_1$ be the event in which A_1 is $(\varepsilon_1, \alpha_1, \gamma_1)$-DP-consistent, $\Pi\left(B\right)_{|A_1} = \Pi\left(A_1\right)$ and $\Pi\left(B\right) \overset{\alpha_1}{\not\approx} \pi_{A_1|B}$, and let $\mathcal{E}_2$ be the corresponding event for A_2. We begin by noting that the probabilities of both $\mathcal{E}_1$ and $\mathcal{E}_2$ are upper bounded by γ_1. To see it for $\mathcal{E}_1$, note that conditioned on A_1 being $(\varepsilon_1, \alpha_1, \gamma_1)$-DP-consistent and on $\Pi\left(B\right)_{|A_1} = \Pi\left(A_1\right)$ it holds that B is a uniformly distributed d_1-subspace satisfying $\Pi\left(B\right)_{|A_1} = \Pi\left(A_1\right)$, and therefore, it holds that $\Pi\left(B\right) \overset{\alpha_1}{\not\approx} \pi_{A_1|B}$ with probability at most γ_1 (by the DP-consistency of A_1). The probability of $\mathcal{E}_2$ can be upper bounded similarly.

It now follows by Claim 9.24 that

$$
\begin{aligned}
\Pr\left[\mathcal{S}_2\right] &= \Pr\left[\mathcal{S}_1 \text{ and } \pi_{A_1|B} \overset{2\alpha_1}{\approx} \pi_{A_2|B}\right] \\
&\geq \Pr\left[\mathcal{S}_1 \text{ and } \neg\mathcal{E}_1 \text{ and } \neg\mathcal{E}_2\right] \\
&\geq \Pr\left[\mathcal{S}_1\right] - \Pr\left[\mathcal{E}_1\right] - \Pr\left[\mathcal{E}_2\right] \\
&\geq \Omega\left(\varepsilon_1^3\right) - 2 \cdot \gamma_1 \\
&\geq \Omega\left(\varepsilon_1^3\right),
\end{aligned}
$$

where the last inequality holds for sufficiently large choice of h. The required result follows. $\qquad\square$

CLAIM 9.27. $\Pr[\mathcal{S}_3] \geq \Omega(\varepsilon_1^3)$.

PROOF.   Let us say that A_1 and A_2 are "agree on a random B" if both A_1 and A_2 are $(\varepsilon_1, \alpha_1, \gamma_1)$-DP-consistent and $\Pr_{B \supset A_1, A_2}\left[\pi_{A_1|B} \overset{2 \cdot \alpha_1}{\approx} \pi_{A_2|B}\right] \geq \Omega(\varepsilon_1^3)$. By Claim 9.26 and by averaging, we know that with probability at least $\Omega(\varepsilon_1^3)$ it holds that A_1 and A_2 agree on a random B. We show that for every A_1 and A_2 that are $(\varepsilon_1, \alpha_1, \gamma_1)$-DP-consistent such that $\pi_{A_1} \overset{3 \cdot \alpha_1}{\not\approx} \pi_{A_2}$ it holds that A_1 and A_2 do not agree on a random B. This will imply that if A_1 and A_2 agree on a random B then it must hold that $\pi_{A_1} \overset{3 \cdot \alpha_1}{\approx} \pi_{A_2}$. Since we know that the probability of A_1 and A_2 to agree on a random B is at least $\Omega(\varepsilon_1^3)$ the required result will follow.

Fix A_1 and A_2 to be any $(\varepsilon_1, \alpha_1, \gamma_1)$-DP-consistent independent d_0-subspaces such that $\pi_{A_1} \overset{3 \cdot \alpha_1}{\not\approx} \pi_{A_2}$. Now, by Lemma 2.5 (subspace-point sampler) and by sufficiently large choice of h, the probability that a uniformly distributed d_1-subspace B that contains A_1 and A_2 contains at most $2 \cdot \alpha_1 \leq 3 \cdot \alpha_1 - 1/q^{d_0-2} - 1/q^{d_1-2 \cdot d_0}$ fraction of coordinates on which π_{A_1} and π_{A_2} disagree is at most $1/\left(q^{d_1-4 \cdot d_0-6}\right)$, and the latter expression can be made smaller than any constant factor times ε_1^3. Thus, it holds that $\Pr_{B \supset A_1, A_2}\left[\pi_{A_1|B} \overset{2 \cdot \alpha_1}{\approx} \pi_{A_2|B}\right]$ can be made sufficiently small such that A_1 and A_2 do not agree on a random B, as required. $\qquad\square$

We now find a global assignment π and show that it agrees with Π on many B's, and then on many A's.

CLAIM 9.28. *There exists an assignment* $\pi : \mathbb{F}^m \to \Sigma$ *such that* $\Pr_B[\Pi(B) \overset{5 \cdot \alpha_1}{\approx} \pi_{|B}$ *and* $\Pi(B)_{|A} = \Pi(A)] \geq \Omega(\varepsilon_1^4)$.

PROOF.   By Claim 9.27 and by averaging, we get that for at least $\Omega(\varepsilon_1^3)$ fraction of the d_0-subspaces A_1, it holds that A_1 is

$(\varepsilon_1, \alpha_1, \gamma_1)$-DP-consistent and

$$\Pr_{A_2 : A_2 \cap A_1 = \{0\}} \left[A_2 \text{ is } (\varepsilon_1, \alpha_1, \gamma_1)\text{-DP-consistent and } \pi_{A_1} \overset{3 \cdot \alpha_1}{\approx} \pi_{A_2} \right]$$
$$\geq \Omega\left(\varepsilon_1^3\right).$$

Fix such d_0-subspace A_1, and set $\pi = \pi_{A_1}$. Consider choosing a uniformly distributed d_0-space A_2 and a uniformly distributed d_1-space $B \supset A_2$. We show that $\Pi(B) \overset{5 \cdot \alpha_1}{\approx} \pi_{|B}$ with probability at least $\Omega\left(\varepsilon_1^4\right)$.

Let us denote by $\mathcal{D}$ the event in which A_2 is independent from A_1, by $\mathcal{P}$ the event in which $\Pi(B)_{|A_2} = \Pi(A_2)$, and by $\mathcal{C}$ the event in which A_2 is $(\varepsilon_1, \alpha_1, \gamma_1)$-DP-consistent and $\pi_{A_1} \overset{3 \cdot \alpha_1}{\approx} \pi_{A_2}$.

By Proposition Proposition 2.18, it holds that $\Pr[\mathcal{D}] \geq 1 - 2 \cdot d_0/q^{m-2 \cdot d_0} \geq \frac{1}{2}$ (where the second inequality holds for sufficiently large h). Furthermore, conditioned on $\mathcal{D}$, the subspace A_2 is a uniformly distributed d_0-subspace of $\mathbb{F}^m$ *that is independent from A_1*, and thus by the choice of A_1, it holds that $\Pr[\mathcal{C}|\mathcal{D}] \geq \Omega\left(\varepsilon_1^3\right)$. Lifting the conditioning, it follows that $\Pr[\mathcal{C}] \geq \Omega\left(\varepsilon_1^3\right)$. Next, observe that B is distributed uniformly over the d_1-subspaces that contain A_2, and thus (since in particular A_2 is ε_1-good) $\Pr[\mathcal{P}|\mathcal{C}] \geq \varepsilon_1$. It therefore holds that $\Pr[\mathcal{C} \text{ and } \mathcal{P}] \geq \Omega\left(\varepsilon_1^4\right)$

Now, let us condition on the events $\mathcal{C}$ and $\mathcal{P}$. By Lemma 2.5 (subspace-point sampler) and for sufficiently large h, it holds with probability at least $1 - 1/\left(q^{d_1 - 3 \cdot d_0 - 6}\right) \geq \frac{3}{4}$ that B contains at most $4 \cdot \alpha_1 \geq 3\alpha_1 + 1/q^{d_0 - 2} + 1/q^{d_1 - 2 \cdot d_0}$ fraction of coordinates on which π_{A_1} and π_{A_2} disagree. Furthermore, by the DP-consistency of A_2 and for sufficiently large choice of h, it holds with probability at least $1 - \gamma_1 \geq \frac{3}{4}$ that $\Pi(B) \overset{\alpha_1}{\approx} \pi_{A_2|B}$. By the union bound and the triangle inequality, it follows that with probability at least $\frac{1}{2}$, it holds that $\Pi(B)$ disagrees with $\pi_{A_1|B}$ on at most $5 \cdot \alpha_1$ fraction of the coordinates. Lifting the conditioning on $\mathcal{C}$ and $\mathcal{P}$, we obtain that with probability at least $\Omega\left(\varepsilon_1^4\right)$, it holds that $\Pi(B) \overset{5 \cdot \alpha_1}{\approx} \pi_{A_1|B}$ and $\Pi(B) = \Pi(A)$ as required. $\square$

Finally, we turn to prove the theorem. Let π be the assignment whose existence is guaranteed by the previous claim. Let us denote by $\mathcal{P}$ the event in which $\Pi(B)_{|A} = \Pi(A)$ (i.e., the P-test accepts

A and B), by $\mathcal{E}_1$ the event in which $\Pi(B) \overset{5 \cdot \alpha_1}{\approx} \pi_{|B}$, by $\mathcal{E}_2$ the event in which $\Pi(A) \overset{6 \cdot \alpha_1}{\approx} \pi_{|A}$, and by $\mathcal{E}_3$ the event in which $\Pi(B)_{|A} \overset{6 \cdot \alpha_1}{\approx} \pi_{|A}$. Using this notation, it suffices to prove that

$$\Pr\left[\mathcal{P} \text{ and } \mathcal{E}_1 \text{ and } \mathcal{E}_2\right] = \Omega\left(\varepsilon_1^4\right).$$

By the definition of π, it holds that

$$\Pr\left[\mathcal{P} \text{ and } \mathcal{E}_1\right] = \Omega\left(\varepsilon_1^4\right).$$

The subspace A is a uniformly distributed d_0-subspace of B, and therefore, it holds by Lemma 2.5 (subspace-point sampler) that

$$\Pr\left[\neg\mathcal{E}_3 \,|\, \mathcal{E}_1\right] = O\left(1/q^{d_0/2-2}\right).$$

This implies that

$$
\begin{aligned}
\Pr\left[\mathcal{P} \text{ and } \mathcal{E}_1 \text{ and } \mathcal{E}_3\right] &= \Pr\left[\mathcal{P} \text{ and } \mathcal{E}_1\right] - \Pr\left[\mathcal{P} \text{ and } \mathcal{E}_1 \text{ and } \neg\mathcal{E}_3\right] \\
&\geq \Pr\left[\mathcal{P} \text{ and } \mathcal{E}_1\right] - \Pr\left[\neg\mathcal{E}_3 | \mathcal{E}_1\right] \\
&= \Omega\left(\varepsilon_1^4\right) - O\left(1/q^{d_0/2-2}\right) \\
&= \Omega\left(\varepsilon_1^4\right),
\end{aligned}
$$

where the last inequality holds for sufficiently large h. Now, observe that whenever both the events $\mathcal{P}$ and $\mathcal{E}_3$ occur, the event $\mathcal{E}_2$ also occurs. It follows that

$$\Pr\left[\mathcal{P} \text{ and } \mathcal{E}_1 \text{ and } \mathcal{E}_2\right] \geq \Pr\left[\mathcal{P} \text{ and } \mathcal{E}_1 \text{ and } \mathcal{E}_3\right] = \Omega\left(\varepsilon_1^4\right),$$

as required. $\square$

Acknowledgements

of the Israel Academy of Sciences and Humanities. An extended abstract of this paper has appeared as Dinur & Meir (2010).

Open Access   This article is distributed under the terms of the Creative Commons Attribution Noncommercial License which permits any noncommercial use, distribution, and reproduction in any medium, provided the original author(s) and source are credited.

References

Sanjeev Arora & Carsten Lund (1996). *Hardness of Approximations*. PW Publishing.

Sanjeev Arora, Carsten Lund, Rajeev Motwani, Madhu Sudan & Mario Szegedy (1998). Proof verification and Intractability of Approximation Problems. *Journal of ACM* **45**(3), 501–555. Preliminary version in FOCS 1992.

Sanjeev Arora & Shmuel Safra (1998). Probabilistic Checkable Proofs: A New Characterization of NP. *Journal of ACM volume* **45**(1), 70–122. Preliminary version in FOCS 1992.

Sanjeev Arora & Madhu Sudan (2003). Improved Low-Degree Testing and its Applications. *Combinatorica* **23**(3), 365–426.

László Babai, Lance Fortnow, Leonid A. Levin & Mario Szegedy (1991). Checking Computations in Polylogarithmic Time. In *STOC*, 21–31.

Mihir Bellare, Shafi Goldwasser, Carsten Lund & Alexander Russell (1993). Efficient probabilistically checkable proofs and applications to approximations. In *STOC*, 294–304.

Eli Ben-Sasson, Oded Goldreich, Prahladh Harsha, Madhu Sudan & Salil Vadhan (2006). Robust PCPs of Proximity, Shorter PCPs and Applications to Coding. *SIAM Journal of Computing* **36**(4), 120–134.

Eli Ben-Sasson, Prahladh Harsha, Oded Lachish & Arie Matsliah (2009). Sound 3-Query PCPPs Are Long. *TOCT* **1**(2).

Peter J. Cameron (1998). *Combinatorics: Topics, Techniques, Algorithms*. Cambridge University Press, Cambridge CB2 2RU, MA, USA.

Irit Dinur (2007). The PCP Theorem by Gap Amplification. *Journal of ACM* **54**(3), 241–250. Preliminary version in STOC 2006.

Irit Dinur, Eldar Fischer, Guy Kindler, Ran Raz & Shmuel Safra (1999). PCP Characterizations of NP: Towards a Polynomially-Small Error-Probability. In *STOC*, 29–40.

Irit Dinur & Elazar Goldenberg (2008). Locally Testing Direct Product in the Low Error Range. In *FOCS*, 613–622.

Irit Dinur & Praladh Harsha (2009). Composition of low-error 2-query PCPs using decodable PCPs. In *FOCS*.

Irit Dinur & Or Meir (2010). Derandomized Parallel Repetition via Structured PCPs. In *CCC*.

Irit Dinur & Omer Reingold (2006). Assignment testers: Towards combinatorial proof of the PCP theorem. *SIAM Journal of Computing* **36**(4), 155–164.

Uriel Feige, Shafi Goldwasser, László Lovász, Shmuel Safra & Mario Szegedy (1996). Interactive Proofs and the Hardness of Approximating Cliques. *J. ACM* **43**(2), 268–292.

Uriel Feige & Joe Kilian (1995). Impossibility results for recycling random bits in two-prover proof systems. In *STOC*, 457–468.

Oded Goldreich & Shmuel Safra (2000). A Combinatorial Consistency Lemma with Application to Proving the PCP Theorem. *SIAM J. Comput.* **29**(4), 1132–1154.

Russell Impagliazzo, Ragesh Jaiswal, Valentine Kabanets & Avi Wigderson (2008). Uniform direct product theorems: simplified, optimized, and derandomized. In *STOC*, 579–588.

Russell Impagliazzo, Valentine Kabanets & Avi Wigderson (2009). New direct-product testers and 2-query PCPs. In *STOC*, 131–140.

Subhash Khot (2006). Ruling Out PTAS for Graph Min-Bisection, Dense k-Subgraph, and Bipartite Clique. *SIAM J. Comput.* **36**(4), 1025–1071.

F. Thomson Leighton (1992). *Introduction to parallel algorithms and architectures: array, trees, hypercubes.* Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

Alexander Lubotzky, R. Phillips & P. Sarnak (1988). Ramanujan graphs. *Combinatorica* **8**(3), 261–277.

Or Meir (2009). Combinatorial PCPs with efficient verifiers. In *FOCS*.

Dana Moshkovitz & Ran Raz (2008). Two Query PCP with Sub-Constant Error. In *FOCS*. Full version is available as ECCC TR08-071.

Christos H. Papadimitriou & Mihalis Yannakakis (1991). Optimization, Approximation, and Complexity Classes. *J. Comput. Syst. Sci.* **43**(3), 425–440.

Alexander Polishchuk & Daniel A. Spielman (1994). Nearly-linear size holographic proofs. In *STOC*, 194–203.

Ran Raz (1998). A Parallel Repetition Theorem. *SIAM J. Comput.* **27**(3), 763–803.

Ran Raz & Shmuel Safra (1997). A Sub-Constant Error-Probability Low-Degree Test, and a Sub-Constant Error-Probability PCP Characterization of NP. In *STOC*, 475–484.

Irit Dinur
Weizmann Institute of Science,
76100 Rehovot, Israel.
irit.dinur@weizmann.ac.il

Or Meir
Weizmann Institute of Science,
76100 Rehovot, Israel.
or.meir@weizmann.ac.il